" LPSTRESS " โปรแกรมการออกแบบแผ่นพื้นคอนกรีตอัดแรง

" LPSTRESS " PRESTRESS CONCRETE SLAB DESIGN PROGRAM

โดย

นายสมพงศ์ คงกระพันธ์ รหัสประจำตัว 35104443

นายสยาม กันเกตุ รหัสประจำตัว 35104449

วัน เดือน ปี 1 ต.ค. 2540

เลขทะเบียน 037193

เลขเรียกหนังสือ T 38288 ส2ค ฝ.

โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมการก่อสร้าง

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

" LPSTRESS " PRESTRESS CONCRETE SLAB DESIGN PROGRAM

MR. SOMPONG KONGKRAPUN

MR. SIAM KUNKATE

A SPECIAL PROJECT SUBMITTED IN PARTIAL

FULLFILLMENT

OF THE REQUIREMENT FOR THE BACHELOR DEGREE OF

CONSTRUTION TECHNOLOGY DEPARTMENT

KING'S MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG

1995

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ใบรับรองโครงงานพิเศษ

หัวข้อโครงงานพิเศษ " LPSTRESS " โปรแกรมการออกแบบแผ่นพื้นคอนกรีตอัดแรง

" LPSTRESS " PRESTRESS CONCRETE SLAB DESIGN PROGRAM

นักศึกษา นาย สมพงศ์ คงกระพันธ์

นาย สยาม กันเกตุ

หลักสูตร วิศวกรรมศาสตรบัณฑิต

สาขา วิศวกรรมการก่อสร้าง

ภาควิชา วิศวกรรมโยธา

อาจารย์ที่ปรึกษา อาจารย์ วิบูลย์ วุฒิญาณ

คณะกรรมการตรวจสอบโครงงานพิเศษ	ลายมือชื่อ
1. อาจารย์ ศรีกริช หิรัญมาศ	
2. อาจารย์ วิบูลย์ วุฒิญาณ	
3. อาจารย์ สมชาย สำลีรางค์กูล	
4. อาจารย์ สมเกียรติ ขวัญพฤกษ์	

ภาควิชาวิศวกรรมโยธารับรองแล้ว

(นายอำนวย พานิชกุลพงศ์)

หัวหน้าภาควิชาวิศวกรรมโยธา

วันที่ เดือน พ.ศ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ " LPSTRESS " โปรแกรมการออกแบบแผ่นพื้นคอนกรีตอัดแรง

" LPSTRESS " PRESTRESS CONCRETE SLAB DESIGN PROGRAM

นักศึกษา นายสมพงศ์ คงกระพันธ์ รหัสประจำตัว 35104443

นายสยาม กันเกตุ รหัสประจำตัว 35104449

อาจารย์ที่ปรึกษา อาจารย์วิบูลย์ วุฒิญาณ

ระดับการศึกษา วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมการก่อสร้าง

ภาควิชา วิศวกรรมโยธา คณะวิศวกรรมศาสตร์

ปีการศึกษา 2538

บทคัดย่อ

โครงงานพิเศษนี้จัดทำขึ้นเพื่อนำ คอมพิวเตอร์เข้ามาช่วยอำนวยความสะดวกในการออกแบบแผ่นพื้นคอนกรีตอัดแรง โดยการเขียนโปรแกรมขึ้นมาใช้งานในคอมพิวเตอร์ โดยใช้ภาษา Visual BASIC ซึ่งสามารถทำงานบนระบบปฏิบัติการ Windows และมีการพัฒนาด้าน GUI (Graphical User Interface) ทำให้โปรแกรมน่าใช้ และการออกแบบแผ่นพื้นคอนกรีตอัดแรง โดยการเขียนโปรแกรมขึ้นมาใช้งานในคอมพิวเตอร์ โดยใช้ภาษา Visual BASIC ซึ่งสามารถทำงานบนระบบปฏิบัติการ Windows และมีการพัฒนาด้าน GUI (Graphical User Interface) ทำให้โปรแกรมน่าใช้ และสะดวกรวดเร็วมากขึ้นหลักการทำงานคือติดต่อกับผู้ใช้ รับรายละเอียดต่าง ๆ ที่ใช้ในการออกแบบแล้วนำมาคำนวณโครงสร้างด้วยโปรแกรมคำนวณ การออกแบบใช้ข้อกำหนดตามมาตรฐาน ACI และ ว.ส.ท. ตามความเหมาะสม การวิเคราะห์โมเมนต์ใช้ทฤษฎีโครงเปรียบเทียบ (Equivalent Frame) ซึ่งเป็นการวิเคราะห์แบบ Elastic ที่มีสมมุติฐานที่แน่นอนใกล้เคียงกับความเป็นจริง ส่วนการวิเคราะห์เหล็กเสริมใช้ทฤษฎีกำลังประลัยซึ่งจะทำให้เกิดความประหยัด

ABSTRACT

This special project's aim is to create a Computer Aided Programe in designing of prestress structures. The program simplifies the complex process in designing of these structures. Equivalent frame method is used in the analysis part since it provides more accurate results than the normal empirical method. Member design part is based on strength design method following American Concrete Institute and Engineering Institute of Thailand Codes. Visual Basic language is used to privide better graphical user interface(GUI). The program runs on Windows 3.1 or higher.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จลงได้ด้วยดี ก็เนื่องจากได้รับความช่วยเหลือและคำแนะนำในการทำงานจากบุคคลต่างๆ ดังนั้นคณะผู้จัดทำจึงขอแสดงความขอบคุณ

อาจารย์วิบูลย์	วุฒิญาณ	อาจารย์ที่ปรึกษา
อาจารย์สมเกียรติ	ขวัญพฤกษ์	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์อำนวย	พานิชกุลพงศ์	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์ศรีกริช	หิรัญมาศ	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์สมชาย	สำลีรางค์กูล	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์ศิริวัฒน์	ไชยชนะ	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์จักรพงษ์	พงษ์เพ็ง	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์สุพจน์	ศรีนิล	อาจารย์ภาควิชาวิศวกรรมโยธา
อาจารย์คมสันต์	มาลีศรี	อาจารย์ภาควิชาวิศวกรรมโยธา

เพื่อนๆ และ พี่ๆ ทุกคนที่ให้ความช่วยเหลือในการทำโครงงานนี้

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ หน้า

ตอนที่ 1

บทที่ 1 บทนำ 1

บทที่ 2 วัสดุสำหรับคอนกรีตอัดแรง 20

คอนกรีต 20

เหล็กเสริม 26

อุกรณ์ยึดกำลังสูง 30

บทที่ 3 การสูญเสียแรงอัด 32

แรงสูญเสียเนื่องจากแรงเสียดทาน 34

แรงสูญเสียเนื่องจากการเคลื่อนตัวในอุปกรณ์ยึด 37

แรงสูญเสียเนื่องจากการหดตัวแบบอิลาสติก 38

แรงสูญเสียเนื่องจากการคืบ 39

แรงสูญเสียเนื่องจากการหดตัว 39

แรงสูญเสียเนื่องจากการล้าตัวของเหล็กเสริม 40

การเปรียบเทียบแรงสูญเสียทั้งหมด 43

บทที่ 4 การวิเคราะห์และการออกแบบคาน 46

การวิเคราะห์หน่วยแรงเนื่องจากการดัด 46

หน่วยแรงบนหน้าตัดคาน 46

หน่วยแรงปลอดภัยในคอนกรีตและเหล็กเสริม 49

การออกแบบหาแรงอัดโดยวิธีน้ำหนักสมดุล 52

การวางลวดเหล็กอัดแรง 52

บทที่ 5 ขั้นตอนการออกแบบ 55

บทที่ 6 การวิเคราะห์โครงสร้างโดยวิธี Equivalent Frame Method 57

บทที่ 7 คานต่อเนื่องและโครงสร้าง Indetermenate 61

การวิเคราะห์หน่วยแรงในคานต่อเนื่อง 72

กำลังประลัยและการถ่ายเทโมเมนต์ดัด 73

โมเมนต์แตกร้าว 76

บทที่ 8 การเขียนและการใช้โปรแกรม 77

ตอนที่ 2 ขั้นตอนการทำงานของโปรแกรม 79

ลำดับขั้นตอนการทำงานของโปรแกรม LPSTRESS 80

ตอนที่ 3 คู่มือการใช้โปรแกรมสำหรับออกแบบคอนกรีตอัดแรง 83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

ตอนที่ 4 ตัวอย่างการออกแบบโดยใช้โปรแกรม LPSTRESS 113

ตอนที่ 5 บทสรุปและวิจัย 124

ภาคผนวก

PROGRAM LISTING

FLOWCHART

บรรณานุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป หน้า

ตอนที่ 1

รูปที่ 1.1 วิธีการอัดแรงแบบดึงเหล็กก่อน 4

รูปที่ 1.2 วิธีการอัดแรงแบบดึงเหล็กทีหลัง 5

รูปที่ 1.3 การก่อสร้างพื้นสองขนาดแรง 6

รูปที่ 2.1 22

รูปที่ 2.2 22

รูปที่ 2.3 การเปลี่ยนแปลงของสัมประสิทธิ์การคืบกับเวลา 24

รูปที่ 2.4 การเปลี่ยนแปลงของสัมประสิทธิ์การหดตัวกับเวลา 26

รูปที่ 2.5 ความสัมพันธ์ระหว่างหน่วยแรงและความเครียดของเหล็กกำลังสูง 30

รูปที่ 3.1 แรงดึงและแรงเสียดทานบน Element ของคาน 34

รูปที่ 3.2 ฟังก์ชันของแนวเหล็กเสริมแบบ พาราโบลา 36

รูปที่ 3.3 การล้าตัวด้วยลวดกำลังสูงและลวดตีเกรียว 41

รูปที่ 4.1 คานคอนกรีตอัดแรงช่วงเดียว 47

รูปที่ 4.2 แสดงความสัมพันธ์ของช่วงอายุคอนกรีตอัดแรงและการแอ่นตัว 53

รูปที่ 4.3 แบบที่แนะนำสำหรับการกระจายเหล็กเสริมในคอนกรีตอัดแรง 54

รูปที่ 6.1 57

รูปที่ 6.2 59

รูปที่ 6.3 59

รูปที่ 6.4 60

รูปที่ 7.1 62

รูปที่ 7.2 การก่อสร้างแบบต่อเนื่องทั้งหมดและต่อเนื่องบางส่วน 63

รูปที่ 7.3 แนวโค้งของเหล็กเสริม 63

รูปที่ 7.4 วิธีการดึงเหล็กในคานต่อเนื่อง 64

รูปที่ 7.5 ผลของแรงอัดคานช่วงเดียวธรรมดา 65

รูปที่ 7.6 66

รูปที่ 7.7 ELEMENT ของเหล็กเสริม 70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ) หน้า

รูปที่ 7.8 ฟังก์ชัน พาราโบลา 71

รูปที่ 7.9 การกระจายหน่วยแรงบนหน้าตัดคานคอนกรีตอัดแรง 76

รูปที่ 7.10 76

รูปที่ 8.1 การเขียนโปรแกรมแบบธรรมดากับแบบ Event drive 78

ตอนที่ 2

ตอนที่ 3

รูปที่ 1 แสดง Icon ของโปรแกรม LPSTRESS 86

รูปที่ 2 แสดงหน้าจอการแนะนำโปรแกรม 87

รูปที่ 3 แสดงส่วนต่าง ๆ บนหน้าจอของตัวโปรแกรม 88

รูปที่ 4 แสดงหน้าจอซึ่งมี Pull down menu File และ menu ย่อยต่าง ๆ 90

รูปที่ 5 แสดงหน้าจอซึ่งมี Pull down menu Input และ menu ย่อยต่าง ๆ 91

รูปที่ 6 แสดงหน้าจอซึ่งมี Pull down menu Design และ menu ย่อยต่าง ๆ 92

รูปที่ 7 แสดงหน้าจอซึ่งมี Pull down menu Result และ menu ย่อยต่าง ๆ 93

รูปที่ 8 แสดงหน้าจอซึ่งมี Pull down menu Save Settingและmenu ย่อยต่าง ๆ 94

รูปที่ 9 แสดงหน้าจอซึ่งมี Pull down menu Optionและmenu ย่อยต่าง ๆ 95

รูปที่ 10 แสดงหน้าจอซึ่งมี Pull down menu ABOUT และ menu ย่อยต่าง ๆ 96

รูปที่ 11 แสดงหน้าจอแสดงการ Click System Tool 97

รูปที่ 12 แสดงหน้าจอซึ่งมีหน้าต่าง Output ของการออกแบบ 99

รูปที่ 13 สดงหน้าจอซึ่งมีหน้าต่าง Output ของ Slab to Design 100

รูปที่ 14 แสดง Shear Force Diagram 101

รูปที่ 15 แสดง Profile ของการวางลวดเหล็กเสริม 102

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ) หน้า

รูปที่ 16 แสดง Output รายการคำนวณ 102
รูปที่ 17 แสดงหน้าจอซึ่ง Click Data Tool 103
รูปที่ 18 แสดงหน้าจอซึ่งมี Panal การใส่ข้อมูลของพื้น 104
รูปที่ 19 แสดงหน้าจอซึ่งมี Panal แสดง Design Strip 107
รูปที่ 20 แสดงหน้าจอซึ่งมี Panal การใส่ข้อมูลของแรง
กระทำบนพื้น 109
รูปที่ 21 แสดงหน้าจอซึ่งมี Panal การใส่ข้อมูลทั่วไป 110
รูปที่ 22 แสดงหน้าจอซึ่งมี Panal การใส่ข้อมูล Node Data 111
รูปที่ 23 แสดงหน้าจอซึ่งมี Panal การใส่ข้อมูล Information 112

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง หน้า

ตอนที่ 1

ตารางที่ 1-1 น้ำหนักบรรทุกจรตามข้อบัญญัติกรุงเทพมหานคร 8

ตารางที่ 1-2 น้ำหนักบรรทุกจรตามมาตรฐาน ANSI 9

ตารางที่ 1-3 ค่าโหลดแฟกเตอร์ตามมาตรฐาน SCI 11

ตารางที่ 1-4 ค่าแฟกเตอร์ 13

ตารางที่ 1-5 16

ตารางที่ 2-1 คุณสมบัติของลวดกำลังสูงตามมาตรฐาน ASTM A421 27

ตารางที่ 2-2 คุณสมบัติของลวดสะแตรนด์ตามมาตรฐาน ASTM A416 28

ตารางที่ 2-3 คุณสมบัติของลวดท่อนเหล็กโลหะผสม 29

ตารางที่ 3-1 การประมาณแรงสูญเสียตามมาตรฐาน AASHTO 33

ตารางที่ 3-2 การประมาณแรงสูญเสียสำหรับงานดึงเหล็กทีหลัง 33

ตารางที่ 3-3 ค่าสัมประสิทธิ์ และ K 35

ตารางที 3-4 ค่าแฟกเตอร์ Ksh 40

ตารางที่ 3-5 ค่าแฟกเตอร์ Kre และ J 42

ตารางที่ 3-6 ค่าแฟกเตอร์ C 42

ตารางที่ 3-7 ค่าหน่วยแรงสูญเสียสูงสุด 43

ตารางที่ 3-8 AASHTO lumpsum loss 45

ตารางที่ 3-9 Approximate Prestress loss value for posttensioning 45

ตารางที่ 4-1 หน่วยแรงปลอดภัยในคอนกรีต 50

ตารางที่ 4-2 หน่วยแรงปลอดภัยในเหล็กเสริมกำลังสูง 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 1

เนื้อหาสำหรับการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ประวัติงานคอนกรีตอัดแรง

หลักการอัดแรงโครงสร้างเพื่อปรับปรุงพฤติกรรมบางอย่างขององค์อาคารได้เป็นที่ทราบกันเป็นเวลานานหลายร้อยปี แต่การนำหลักการมาประยุกต์กับงานคอนกรีตได้เริ่มต้นเมื่อร้อยปีที่แล้ว ในปี ค.ศ. 1886 , P.H. Jackson วิศวกรแห่งเมือง ซานฟรานซีสโก สหรัฐอเมริกาได้จดทะเบียนการก่อสร้างแผ่นพื้นคอนกรีตโดยการขันท่อนเหล็กยึดพื้นคอนกรีตเข้าด้วยกัน ในปีค.ศ. 1888, C.E.W. Doehring ได้จดทะเบียนการก่อสร้างแผ่นพื้นคอนกรีตโดยการอัดแรงก่อนการรองรับน้ำหนักบรรทุกในประเทศเยอรมัน อย่างไรก็ตามวิธีก่อสร้างแผ่นพื้นคอนกรีตอัดแรงในระยะนั้นไม่ใคร่ประสบความสำเร็จมากนักทั้งนี้เพราะแรงอัดแผ่นพื้นเกิดขึ้นเนื่องจากการขันท่อนเหล็กกล้าละมุนในตรึง เมื่อเวลาผ่านไปนาน ๆ แผ่นพื้นคอนกรีตหดตัวเนื่องจากการคืบและการหดตัวเนื่องจากการสูญเสียความชื้น ทำให้แรงอัดแผ่นพื้นก่อนเหลือเพียงเล็กน้อย ถึงแม้ว่า C.R. Steiner ในปี ค.ศ. 1908 จะใช้วิธีขันท่อนเหล็กให้แน่นใหม่หลังจากที่คอนกรีตได้หดตัวไปบ้างแล้ว และค่าก่อสร้างวิธีนี้ก็สูงมากกว่าที่จะก่อสร้างพื้นคอนกรีตธรรมดา จนกระทั่งปี ค.ศ. 1925 R.E. Dill ใช้ท่อนเหล็กกำลังสูงเสริมคอนกรีต ในการก่อสร้างเหล็กจะถูกทาด้วยสารป้องกันการยึดเหนี่ยวก่อนการเทคอนกรีต เมื่อคอนกรีตแข็งตัวแล้วจึงอัดแรงเข้าสู่คอนกรีตโดยการขันน็อตยึดกับคาน วิธีนี้คือการก่อสร้างคอนกรีตอัดแรงก็ไม่แพร่หลายนัก

การพัฒนาคอนกรีตอัดแรงสมัยใหม่ได้ริเริ่มในประเทศฝรั่งเศส โดย E. Freyssinet ในปี ค.ศ. 1928 โดยเริ่มใช้ลวดเหล็กซึ่งมีกำลังประลัยถึง 175000 กก/ซม2 ในการผลิตคอนกรีตอัดแรง เราลองพิจารณาว่าลวดเหล็กกำลังสูงนี้ถูกดึงจนเกิดหน่วยแรงประมาณ 10000 กก/ซม2 เนื่องจากโมดูลัสของความยืดหยุ่นของลวดกำลังสูงเป็น 10000/2000000 เท่ากับ 0.005 เมื่อคอนกรีตเกิดการหดตัวคิดเป็นความเครียด 0.001 จะเห็นว่าเหลือหน่วยแรงอัดก่อนอยู่เท่ากับ 0.004x 2000000 เท่ากับ8000 กก/ซม2 ซึ่งเท่ากับ 80 เปอร์เซ็นต์ของแรงอัดก่อนแรกเริ่ม หลังจากปี ค.ศ. 1930 ได้มีการพัฒนาวิธีอัดแรงแบบดึงเหล็กก่อน ในประเทศเยอรมันโดย E. Hoyer รวมถึงการพัฒนาแม่แรงไฮดรอลิคที่ใช้ดึงลวดและอุปกรณ์ยึดลวดในประเทศฝรั่งเศสและประเทศเบลเยี่ยม และการก่อสร้างโดยใช้คอนกรีตอัดแรงได้เป็นที่แพร่หลายตั้งแต่หลังสงครามโลกครั้งที่สองเป็นต้นมา

ในประเทศไทยได้เริ่มมีการก่อสร้างสะพานคอนกรีตอัดแรงในช่วงหลัง พ.ศ. 2505 โดยมีการผลิตงานสะพานคอนกรีตอัดแรงช่วงยาวต่าง ๆ กันตั้งแต่ 5.00เมตร จนถึง 60เมตร ในระยะหลังได้มีโครงงานก่อสร้างเสาเข็มคอนกรีตอัดแรง พื้นอาคารคอนกรีตอัดแรงสำเร็จรูปตลอดจนการก่อสร้างแบบพื้นไร้คานแบบหล่อในที่ และดึงเหล็กทีหลังหลายโครงการ การก่อสร้างโดยใช้คอนกรีตอัดแรงเป็นที่นิยมมาก เพราะประหยัดเวลาและค่าก่อสร้าง เพราะองค์อาคารน้ำหนักน้อยเมื่อเทียบกับคอนกรีตเสริมเหล็กก่อนสามารถผลิตได้เป็นจำนวนมากในเวลาสั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการอัดแรงสำหรับคอนกรีตอัดแรง

โครงสร้างคอนกรีตอัดแรงสามารถแบ่งตามวิธีการอัดแรงระหว่างการก่อสร้างได้ดังนี้

การอัดแรงโดยใช้แม่แรง

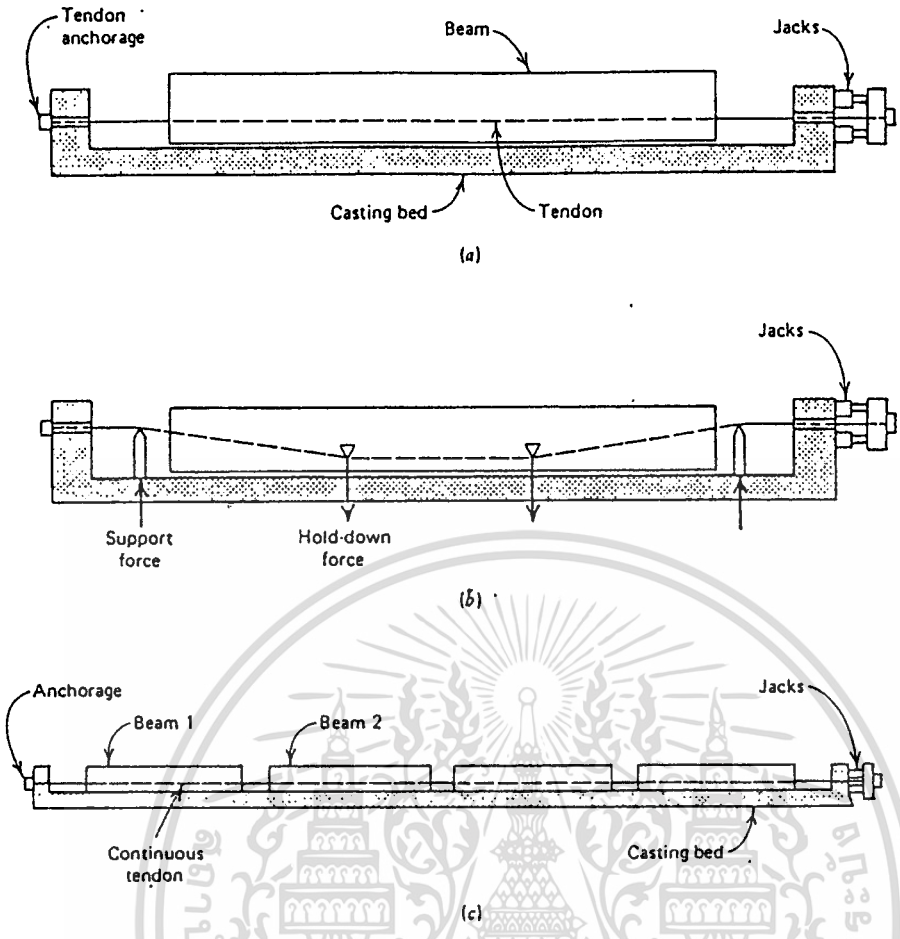
วิธีนี้อาจเรียกว่าการอัดแรงภายนอก การอัดแรงกระทำได้โดยการใช้แม่แรงวางระหว่างปลายคานทั้งสองข้างกับอบัทเมนต์และทำการปั๊มแม่แรงอัดตามที่ต้องการ วิธีนี้ใช้ในการก่อสร้างเขื่อนหรือโครงสร้างพิเศษอื่น ๆ ซึ่งต้องมีการก่อสร้างวิธีนี้ไม่ใคร่เป็นที่นิยมมากนักเพราะการสูญเสียแรงอัดเกิดขึ้นมากกว่าวิธีอัดแรงภายในเนื่องจากคอนกรีตอยู่ในสภาวะความเครียดคงที่ทำให้ต้องมีการเพิ่มแรงอัดภายหลังเป็นครั้งคราว

การอัดแรงแบบดึงเหล็กก่อน

วิธีนี้เรียกว่าการอัดแรงภายใน เป็นวิธีการก่อสร้างคอนกรีตอัดแรงที่ใช้แพร่หลายในปัจจุบันเพราะการก่อสร้างทำได้รวดเร็ว วิธีนี้เหมาะสมสำหรับการผลิตในโรงงานซึ่งมีการก่อสร้างอบัทเมนต์ถาวร การอัดแรงแบบดึงเหล็กก่อนหมายถึงเราดึงเหล็กก่อนการเทคอนกรีต ขั้นตอนก่อสร้างเริ่มต้นจากการเตรียมแบบหล่อคอนกรีต จากนั้นจึงขึงลวดกำลังสูงหรือลวดเหล็กตีเกลียวกำลังสูงหรือสะแตรนตามแนวเหล็กที่ออกแบบไว้ ลวดเหล็กนี้มีปลายยึดกับอบัทเมนต์โดยใช้อุปกรณ์ยึด ในกรณีที่แนวของลวดประกอบด้วยเส้นตรงหักมุมจะต้องออกแบบอุปกรณ์สำหรับยึดเส้นลวดด้วย ลวดเหล็กกำลังสูงถูกดึงให้ตึงจนได้แรงในเส้นลวดเท่ากับจำนวนที่ต้องการ ทั้งนี้จะต้องตรวจสอบความดันของน้ำมันไฮดรอลิคและระยะยืดของเส้นลวดด้วย เมื่อทำการดึงเหล็กแล้วจึงเทคอนกรีตลงในแบบหล่อ ตามปกติคอนกรีตที่ใช้จะเป็นคอนกรีตกำลังสูงกว่าคอนกรีตในงานคอนกรีตเสริมเหล็ก เมื่อคอนกรีตแข็งตัวแล้วก็ทำการบ่มต่อไปอีก 3-7 วัน จนกำลังของคอนกรีตเพิ่มมากพอ จึงทำการอัดแรงคอนกรีตดยการตัดลวด เนื่องจากลวดมีการยึดเหนี่ยวกับคอนกรีตตลอดเส้น เมื่อตัดลวดแรงดึงในเหล็กจะถ่ายเป็นแรงอัดในคอนกรีตโดยผ่านแรงยึดเหนี่ยวที่บริเวณปลายคาน หลังจากนั้นองค์อาคารก็สามารถขนส่งไปยังสถานที่ก่อสร้างได้

การก่อสร้างแบบดึงเหล็กก่อนเหมาะสำหรับการผลิตเป็นจำนวนมากในเวลาเดียวกันในระบบผลิตแบบนี้โครงสร้างฐานอบัทเมนต์จะมีระยะห่างกันมากกว่า 100 เมตร ซึ่งฐานหล่อคานคอนกรีตจะมีความยาวพอที่จะหล่อองค์อาคารได้หลายท่อนพร้อม ๆ กัน ระบบผลิตนี้นิยมใช้สำหรับการก่อสร้างเสาเข็ม คานสะพาน หรือพื้นสำเร็จรูปเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



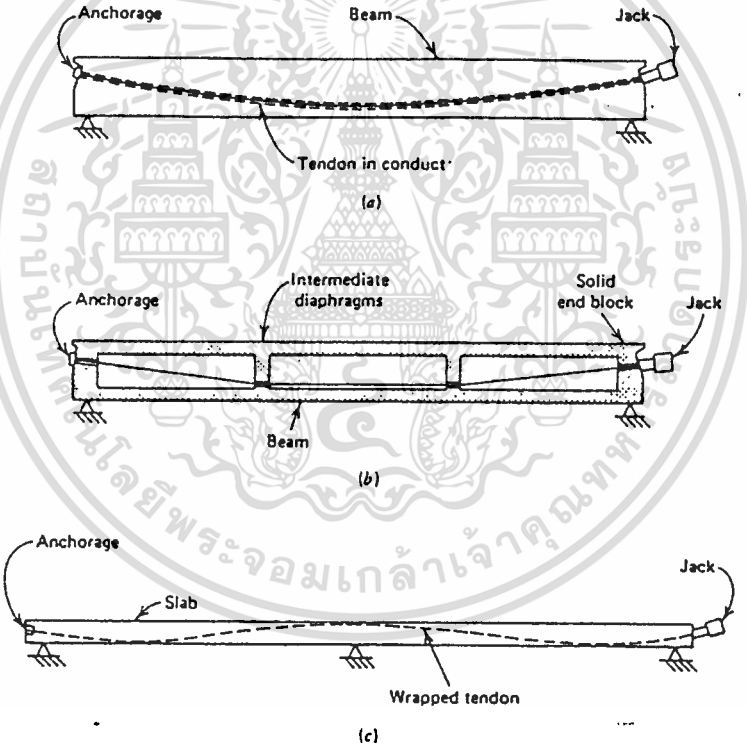
รูปที่ 1.1 วิธีอัดแรงแบบดึงเหล็กก่อน

การอัดแรงแบบดึงเหล็กทีหลัง

วิธีนี้เป็นการอัดแรงภายในอีกวิธหนึ่งซึ่งใช้กันแพร่หลายเช่นเดียวกัน สำหรับโครงสร้างซึ่งมีขนาดใหญ่ซึ่งไม่สะดวกในการขนส่ง หรือสำหรับสถานที่ก่อสร้างซึ่งอยู่ห่างไกลโรงงานผลิตมาก ท่อคอนดุทและลวดเหล็กกำลังสูงจะถูกร้อยภายในแบบหล่อองค์อาคารคอนกรีตพร้อมอุปกรณ์แองคอเรจทั้งสองข้าง จากนั้นจึงทำการเทคอนกรีต เมื่อคอนกรีตแข็งตัวและได้รับการบ่มจนมีกำลังสูงเพียงพอจึงทำการอัดแรงโดยใช้แม่แรงไฮดรอลิคดึงเหล็กให้ยืดออกโดยยันกับคานคอนกรีตเอง และลวดเหล็กจะขึงตึงระหว่างปลายทั้งสองข้างโดยการยึดของตัวลิ่มในอุปกรณ์แองคอเรจ เราสามารถจะทำให้ลวดเหล็กยึดติดกับเนื้อคอนกรีตตลอดความยาวของคานได้โดยการอัดมอร์ต้าในท่อคอนดุท เรียกว่าboned tendon คานซึ่งมี bonded tendon ที่ไม่ได้รับการอัดมอร์ต้าในคอนดุท เรียกว่า unbonded tendon จะมีพฤติกรรมในช่วง overload ดีกว่าคานซึ่งมี unbonded tendon ตามปกติ tendon หนึ่งประกอบด้วยลวดกำลังสูงหรือ strand หลายเส้นร้อยอยู่ในคอนดุทท่อเดียวกันและอุปกรณ์แองคอเรจที่ปลายคานจะมีจำนวนรูสำหรับยึดลวดเท่ากับจำนวนลวดหรือ strand ในคอนดุท รูปที่ 1.3แสดงวิธีการก่อสร้างแบบดึงเหล็กทีหลังสำหรับคานช่วงเดี่ยวธรรมดาและคานต่อเนื่อง

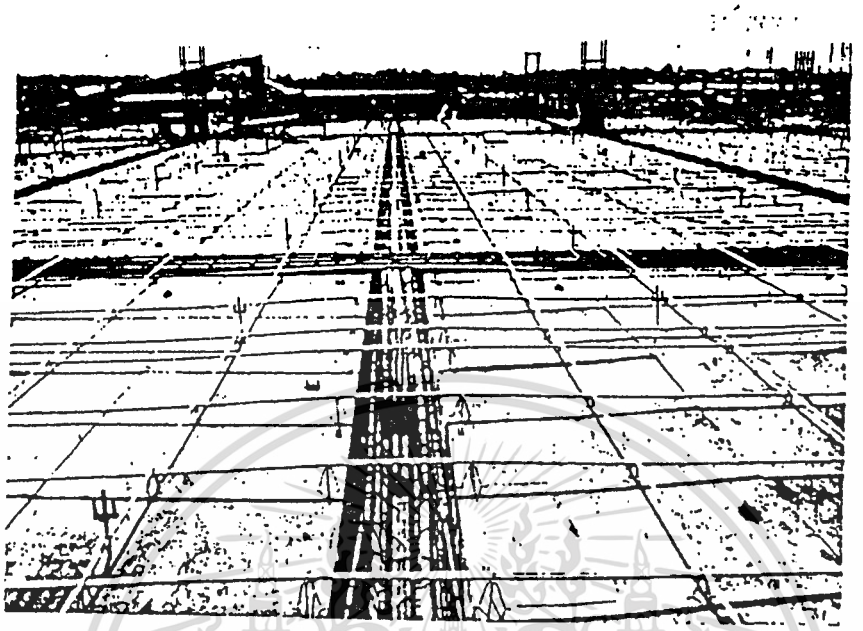
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการก่อสร้างระบบพื้นไร้คานอัดแรง หรือพื้นสองทางอัดแรงวิศวกรมักจะออกแบบเหล็กอัดแรงเป็นแบบ unbonded tendon เนื่องจากค่าใช้จ่ายในการอัดมอร์ต้า และราคาคอนดุทจะเป็นมูลค่าสูง tendon ในพื้นปกติจะประกอบด้วยเหล็ก strand เดี่ยววางห่างกัน 40-50 ซม. เหล็ก strand นี้จะทายางแอสฟัลต์หรือพันด้วยกระดาษแอสฟัลต์ ซึ่งทำหน้าที่ป้องกันเหล็กยึดกับคอนกรีตขณะที่ดึง tendon รูปที่ 1.4 แสดงการเตรียม tendon ในการก่อสร้างพื้นสองทางอัดแรงซึ่งเราจะเห็นการวาง tendonซึ่งพันด้วยกระดาษแอสฟัลต์ ในพื้นคอนกรีตซึ่งประกอบด้วย unbonded tendon นี้เราจะต้องเสริมเหล็กกล้าละมุนเพื่อประโยชน์ในการกระจายรอยร้าวในช่วงน้ำหนักประลัยด้วยซึ่งจะได้กล่าวต่อไป



รูปที่ 1.2 วิธีการอัดแรงแบบดึงเหล็กทีหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.3 การก่อสร้างพื้นสองทางอัดแรง

การอัดแรงบางส่วน

หลักการออกแบบคอนกรีตอัดแรงในระยะแรกมักจะพิจารณาระดับการอัดแรงเต็มที่ กล่าวคือที่ระดับน้ำหนักบรรทุกปกติหน่วยแรงที่เกิดขึ้นเป็นเฉพาะหน่วยแรงอัด นโยบายการอัดแรงที่มีข้อเสียคือแรงอัดคานมีขนาดใหญ่ ทำให้ขนาดขององค์อาคารเพิ่มขึ้นเพื่อหน่วยแรงอัดจะต้องไม่สูงเกินไปด้วย นอกจากนี้คานอาจจะเกิดการโค้งงอ เนื่องจากการอัดแรงมากเกินไปด้วย ในการใช้งานคอนกรีตที่อัดแรงเต็มที่อาจจะเกิดหน่วยแรงดึงบ้าง เช่นคานสะพานเมื่อมีน้ำหนักจรสูงมากซึ่งเกิดขึ้นเป็นครั้งคราวจะทำให้เกิดหน่วยแรงดึงขึ้นเป็นผลให้เกิดรอยแตกร้าว รอยร้าวนี้จะปิดสนิทเมื่อน้ำหนักบรรทุกผ่านเลยไป ในเวลาต่อมาหลักการออกแบบอนุญาตให้องค์อาคารเกิดหน่วยแรงดึงได้บ้าง นโยบายนี้ทำให้ขนาดของแรงอัดคานลดลงซึ่งเราเรียกว่า การอัดแรงบางส่วน โดยควบคุมให้หน่วยแรงดึงไม่สูงเกินไป หลักการอัดแรงบางส่วนนี้ทำให้ขนาดขององค์อาคารคอนกรีตอัดแรงลดลงได้ นอกจากนี้พฤติกรรมการแอ่นตัวก็ดีขึ้น ในบางกรณีข้อบัญญัติกำหนดให้ใช้เหล็กกล้าละมุนต้านทานหน่วยแรงดึงที่เกิดขึ้น ดังนั้นการอัดแรงบางส่วนจึงเป็นมาตรการกลางระหว่างคอนกรีต ซึ่งอัดแรงเต็มที่และคอนกรีตเสริมเหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้ำหนักบรรทุกบนโครงสร้าง

ในการออกแบบโครงสร้างคอนกรีตอัดแรง วิศวกรต้องคำนวนน้ำหนักขององค์อาคาร และน้ำหนักจรที่กระทำต่อโครงสร้างคอนกรีตเสริมเหล็กหรือโครงสร้างเหล็ก น้ำหนักขององค์อาคารคอนกรีต คำนวนจากน้ำหนักจำเพาะของคอนกรีตเท่ากับ 2400 กก/ลบม. ตามปกติเราต้องกำหนดขนาดหน้าตัดองค์อาคารซึ่งมีความแข็งแรงที่จะต้านทานน้ำหนักบรรทุกจร หลังจากการวิเคราะห์โครงสร้างจำเป็นต้องทบทวนขนาดหน้าตัดใหม่ให้เหมาะสม เกี่ยวกับน้ำหนักบรรทุกจรวิศวกรจะต้องพิจารณาประเภทและลักษณะการใช้สอยของโครงสร้าง น้ำหนักบรรทุกจรนี้กำหนด ในข้อบัญญัติและมาตรฐานการออกแบบต่าง ๆ ของ ANSI มาตรฐานของ AASHTO ข้อบัญญัติกรุงเทพมหานครเป็นต้น น้ำหนักบรรทุกจรและน้ำหนักขององค์อาคารรวมกัน เรียกว่าน้ำหนักบรรทุกปกติซึ่งเป็นน้ำหนักบรรทุกเกณฑ์ปกติสูงสุดที่คาดว่าจะกระทำต่อโครงสร้างขณะใดขณะหนึ่งตลอดอายุของโครงสร้าง

ความปลอดภัยของโครงสร้าง

น้ำหนักบรรทุกปกติเนื่องจากน้ำหนักองค์อาคารและน้ำหนักบรรทุกจร ทำให้เกิดแรงภายในบนส่วนต่าง ๆ ของโครงสร้างและหน่วยแรงในวัสดุ ในการออกแบบส่วนต่าง ๆ ของโครงสร้างจะต้องมีความแข็งแรงเพียงพอที่จะต้านทานแรงกระทำประลัยบนโครงสร้างจนถึงระดับหนึ่งทำให้วัสดุต้านทานหน่วยแรงที่เกิดขึ้นจนถึงกำลังสูงสุด เนื่องจากขนาดของแรงกระทำประลัยขึ้นอยู่กับโอกาสของการเกิดแรงนั้น ตามมาตรฐานการออกแบบ เช่น มาตรฐานการออกแบบอาคารของ ACI จึงกำหนดระดับโหลดแฟกเตอร์ ,U ซึ่งเป็นอัตราส่วนของแรงกระทำประลัยต่อแรงกระทำปกติบนโครงสร้าง

ค่ากำลังประลัยของคอนกรีตหรือเหล็กกำลังสูงมีความแปรปรวนเกิดขึ้น เนื่องจากเหตุผลหลายประการ ได้แก่ความแปรปรวนของการวัด และลักษณะการทดสอบ เพื่อพิจารณาถึงกำลังประลัยที่น่าเชื่อถือได้ มาตรฐานการออกแบบจึงกำหนดค่าแฟคเตอร์ ϕ เป็นเปอร์เซ็นต์ของค่ากำลังประลัยที่ระบุค่า แฟกเตอร์ ϕ หรือ Capacity reduction factor มีค่าต่างกันตามลักษณะการต้านทานแรงกระทำประลัยขององค์อาคาร

ในการตรวจสอบความปลอดภัยของโครงสร้าง เราวิเคราะห์โครงสร้างเพื่อหาโมเมนต์ดัด แรงเฉือน หรือแรงตามแนวแกนภายใต้แรงกระทำประลัย แรงภายในที่เกิดนี้จะต้องน้อยกว่าหรือเท่ากับกำลังประลัยของคอนกรีตคูณด้วยค่าแฟกเตอร์ ϕ กำลังประลัยขององค์อาคารนี้วิเคราะห์จากค่ากำลังประลัยของคอนกรีตและของเหล็ก โดยใช้ทฤษฎีพฤติกรรมขององค์อาคารที่ถูกต้อง ดังนั้นถ้า M_U , V_U และ N_U เป็นแรงภายในประลัยที่เกิดขึ้นในองค์อาคารและ M_n ,V_n และ N_n เป็นกำลังประลัยขององค์อาคารในเชิงการดัด การเฉือน และการรับแรงตามแนวแกน เราจะตรวจสอบว่า $M_U <= M_n$; $V_U <= \phi V_n$ และ $N_u <= \phi N_n$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1-1 น้ำหนักบรรทุกจรตามข้อบัญญัติกรุงเทพมหานคร

	ประเภทการใช้อาคาร	น้ำหนักบรรทุกเป็นกิโลกรัมต่อตารางเมตร
1.	หลังคา	50
2.	กันสาดหรือหลังคาคอนกรีต	100
3.	ที่พักอาศัย โรงเรียนอนุบาล ห้องน้ำ ห้องส้วม	150
4.	ห้องแถว ตึกแถว อาคารชุด หอพัก โรงแรม และห้องคนไข้พิเศษของโรงพยาบาล	200
5.	สำนักงาน ธนาคาร	250
6.	(ก) อาคารพาณิชย์ ส่วนของห้องแถว ตึกแถวที่ใช้เพื่อการพาณิชย์ มหาวิทยาลัย วิทยาลัย และโรงเรียน	300
	(ข) ห้องโถง บันได ช่องทางเดิน ของอาคารชุด หอพัก โรงแรม โรงพยาบาล สำนักงาน และธนาคาร	300
7.	(ก) ตลาด ห้างสรรพสินค้า หอประชุม โรงมหรสพ ภัตตาคาร ห้องประชุม ห้องอ่านหนังสือในหอสมุด ที่จอดหรือเก็บรถยนต์นั่ง	400
	(ข) ห้องโถง บันได ช่องทางเดิน ของอาคารพาณิชย์ มหาวิทยาลัย วิทยาลัย และโรงเรียน	400
8.	(ก) คลังสินค้า โรงกีฬา พิพิทธภัณฑ์ อัฒจันทร์ โรงงานอุตสาหกรรม โรงพิมพ์ ห้องเก็บเอกสาร และพัสดุ	500
	(ข) ห้องโถง บันได ช่องทางเดิน ของตลาด ห้างสรรพสินค้า หอประชุม โรงมหรสพ ภัตตาคาร และหอสมุด	500
9.	ห้องเก็บหนังสือของหอสมุด	600
10.	ที่จอดหรือเก็บรถยนต์บรรทุกเปล่าและรถอื่น ๆ	800

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1-2 น้ำหนักบรรทุกจรตามมาตรฐาน ANSI

Occupancy or Use	Live Load psf	Live Load kN/m^2
Apartments (see Residential)		
Armories and drill rooms	150	7.2
Assembly halls and other places of assembly:		
Fixed seats	60	2.9
Movable seats	100	4.8
Balcony (exterior)	100	4.8
Bowling alleys, poolrooms, and similar recreational areas	75	3.6
Corridors:		
First floor	100	4.8
Other floors, same as occupancy served except as indicated		
Dance halls	100	4.8
Dining rooms and restaurants	100	4.8
Dwellings (see Residential)		
Garages (passenger cars)	100	4.8
Floors shall be designed to carry 150 percent of the maximum wheel load anywhere on the floor.		
Grandstands (see Reviewing stands)		
Gymnasiums, main floors, and balconies	100	4.8
Hospitals:		
Operating rooms	60	2.9
Private rooms	40	1.9
Wards	40	1.9
Hotels (see Residential)		
Libraries:		
Reading rooms	60	2.9
Stack rooms	150	7.2
Manufacturing	125	6.0
Marquees	75	3.6
Office buildings:		
Offices	80	3.8
Lobbies	100	4.8
Penal institutions:		
Cell blocks	40	1.9
Corridors	100	4.8
Residential:		
Multifamily houses:		
Private apartments	40	1.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1-2 น้ำหนักบรรทุกจรตามมาตรฐาน ANSI (ต่อ)

Occupancy of Use	Live Load psf	Live Load kN/m^2
Public rooms	100	4.8
Corridors	60	2.9
Dwellings:		
First floor	40	1.9
Second floor and habitable attics	30	1.4
Uninhabitable attics	20	1.0
Hotels:		
Guest rooms	40	1.9
Public rooms	100	4.8
Corridors serving public rooms	100	4.8
Public corridors	60	2.9
Private corridors	40	1.9
Reviewing stands and bleachers	100	4.8
Schools:		
Classrooms	40	1.9
Corridors	100	4.8
Sidewalks, vehicular driveways, and yards subject to trucking	250	12.0
Skating rinks	100	4.8
Stairs, fire escapes, and exitways	100	4.8
Storage warehouse:		
Light	125	6.0
Heavy	250	12.0
Stores:		
Retail:		
First floor, rooms	100	4.8
Upper floors	75	3.6
Wholesale	125	6.0
Theaters:		
Aisles, corridors, and lobbies	100	4.8
Orchestra floors	60	2.9
Balconies	60	2.9
Stage floors	150	7.2
Yards and terraces, pedestrians	100	4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1-3 ค่าโหลดแฟกเตอร์ตามมาตรฐาน ACI

---

1. The required strength U to resist dead load D and live load L shall be at least equal to

$$U = 1.4D + 1.7L \qquad \text{(ACI 9-1)}$$

2. If resistance to structural effects of a specified wind load W are included in design, the following combinations of D, L, and W shall be investigated to determine the greatest required strength U:

$$U = 0.75(1.4D + 1.7L + 1.7W) \qquad \text{ACI 9-2)}$$

where load combinaitons shall include both full value and zero value of L to determine the more severe condition, and

$$U = 0.9D + 1.3W \qquad \text{(ACI 9-3)}$$

but for any combination of D, L, and W the required strength U shall not be less than Eq. (ACI 9-1).

3. If resistance to specified earthquake loads or forces E are included in design, load combinations of Section 2 shall apply, except that 1.1E shall be substituted for W.

4. If resistance to lateral earth pressure H is included in design, the required strength U shall be at least equal to

$$U = 1.4D + 1.7L + 1.7H \qquad \text{(ACI 9-4)}$$

and where D or L reduce the effect of H, the following combinations of D, L, and H shall be investigated to determine the greatest required strength U:

D opposing H:	$U = 0.9D + 1.7L + 1.7H$	(ACI 9-5)
L opposing H:	$U = 1.4D + 1.7H$	(ACI 9-6)
D and L opposing H:	$U = 0.9D + 1.7H$	(ACI 9-7)

but for any combination of D, L, and H the required strength U shall not be less than Eq. (ACI 9-1).

5. If resistance to lateral liquid pressure F is included in design, the load combination of Section 4 shall apply, except that 1.4F shall be substituted for 1.7H. Vertical liquid pressure shall be considered as dead load D, with due regard to variation in liquid depth.

6. If resistance to impact effects are included in design, such effects shall be included with live load L.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1-3 ค่าโหลดแฟกเตอร์ตามมาตรฐาน ACI(ต่อ)

7. Where structural effects T of differential settlement, creep, shrinkage, or temperature change may be significant in design, the required strength U shall be at least equal to $U = 0.75(1.4D + 1.4T + 1.7L)$ (ACI 9-8) but the required strength U shall not be less than $U = 1.4(D + T)$ (ACI 9-9) Estimations of differential settlement, creep, shrinkage, or temperature change shall be based on a realistic assessment of such effects occurring in service.

[a] Adapted with permission of the American Concrete Institute from ACI Building Code 318-77.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1-4 ค่าแฟกเตอร์ ϕ

Kine of Stress	Strength Reduction Factor ø
Flexural, with or without axial tension	0.90
Axial tension	0.90
Axial compression, with or without flexure:	
Members with spiral reinforcement	0.75
Other reinforced members	0.70
except that for low values of axial load, ø may be increased in accordance with the following:	
For members in which f_y does not exceed 60,000 psi, with symmetrical reinforcement, and with $(h - d' - d_s)/h$ not less than 0.70, ø may be increased linearly to 0.90 as ø P_n decreases from $0.10f'_c A_g$ to zero.	
For other reinforced members, ø may be increased linearly to 0.90 as øP_n decreases from $0.10f'_c A_g$ or ø P_{nb} whichever is smaller, to zero.	
Shear and torsion	0.85
Bearing on concrete	0.70
Flexure in plain concrete	0.65

[a] Adapted with permission of the American Concrete Institute from ACI Building Code 318-77.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นคอนกรีตอัดแรงกับเศรษฐศาสตร์

(ECONOMIC OF POST TENSION SLAB)

งานพื้นคอนกรีตอัดแรงเกี่ยวกับทางด้านเศรษฐกิจเป็นอย่างมาก รวมทั้งมีผลถึงด้านสถาปัตยกรรมที่ทำให้อาคารมีความสวยงามก่อสร้างได้ง่ายได้ตามรูปแบบที่ต้องการถ้าจะพิจารณาในด้านพฤติกรรมของโครงสร้างแล้วโครงสร้างคอนกรีตอัดแรงจะดีกว่าโครงสร้างอื่น ๆ ที่ใช้กันอยู่ในปัจจุบันนี้

ความประหยัด

1) <u>ทางตรง</u> หากความยาวช่วงเสาและรูปแบบของอาคารมีความเหมาะสมและระบบพื้นคอนกรีตอัดแรงที่ได้รับการออกแบบโดยคำนึงถึงปัจจัยข้อพิจารณาต่าง ๆ โดยถูกต้องอาจมีราคาประหยัดกว่าระบบพื้นคอนกรีตเสริมเหล็กถึง 10 เปอร์เซ็นต์ พิจารณาจากค่าก่อสร้างของระบบพื้นทั้งหมดรวมค่าวัสดุและค่าแรงทั้งนี้ยังไม่รวมถึงผลเนื่องจากเวลาที่ใช้ในการทำงานน้อยลงไป

2)<u>ทางอ้อม</u>

ประหยัดไม้แบบเพราะเป็นไม้แบบท้องเรียบ ซึ่งมีรูปแบบง่าย ๆ ทำให้การทำงานตั้งแบบ ถอดแบบทำได้ง่ายและรวดเร็วไม้แบบเสียหายน้อยสามารถนำกลับมาใช้ได้หลายครั้ง นอกจากนั้นเนื่องจากการถอดไม้แบบสามารถทำได้ทันทีเมื่อทำการดึงลวดอัดแรงเสร็จเรียบร้อย(ประมาณ 4-5 วันหลังจากการเทคอนกรีต)ทำให้สามารถหมุนเวียนไม้แบบมาใช้ได้เร็วขึ้น ซึ่งทำให้ไม่จำเป็นต้องใช้ไม้แบบเป็นจำนวนมาก

ประหยัดโครงสร้างอื่น ๆ เนื่องจากอาคารที่ใช้ระบบพื้นคอนกรีตอัดแรงมีความสูงของอาคารน้อยกว่าอาคารที่ใช้ระบบอื่น ๆ ที่ใช้จำนวนชั้นเท่า ๆ กัน ทำให้ความยาวของเสาของอาคารสั้นลงระบบท่อสั้นลงพื้นที่ของผนังอาคารน้อยลง ซึ่งนอกจากนั้นการที่อาคารมีความสูงลดลงจะทำให้ประหยัดระบบวัสดุตกแต่งเช่น กระเบื้อง กระจก และอื่น ๆ นอกจากนั้นการที่อาคารมีความสูงลดลงจะทำให้ประหยัดระบบโครงสร้างที่ใช้รับแรงลม เช่น เสา ผนัง เป็นต้น

ประหยัดค่าโสหุ้ยการก่อสร้างเนื่องจากสามารถก่อสร้างได้เร็วขึ้น

ความรวดเร็วของการก่อสร้างระบบพื้นคอนกรีตอัดแรงได้พิสูจน์ให้เห็นในหลาย ๆ โครงการว่าสามารถทำการก่อสร้างได้อย่างรวดเร็ว เมื่อเปรียบเทียบกับระบบพื้นอื่น ๆ ทั้งนี้เนื่องจากการทำงานในขั้นตอนต่าง ๆ มีความง่าย เช่นการตั้งไม้แบบการวางเหล็กเสริมธรรมดา นอกจากนี้การถอดแบบสามารถทำได้รวดเร็วเมื่อเปรียบเทียบกับระบบพื้นอื่น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นคอนกรีตอัดแรงกับงานสถาปัตยกรรม

1)ความคล่องตัวของพื้นที่ใช้สอยสถาปัตยกรรมสมัยใหม่เน้นความจำเป็นของการใช้ที่ว่างที่มีความกว้างขวางขึ้นโดยจะต้องมีความอิสระในการจัดแบ่งพื้นที่ภายใน เพื่อให้เหมาะสมกับการใช้ประโยชน์ของพื้นที่ระบบพื้นคอนกรีตอัดแรงสามารถให้ช่วงเสาได้กว้างกว่าทำให้มีจำนวนเสาน้อย และพื้นที่ว่างมากขึ้น การก่อกำแพงสามารถก่อบนพื้นได้โดยไม่ต้องมีคานรองรับ ซึ่งสามารถให้อิสระนการจัดแบ่งพื้นที่ และการเปลี่ยนแปลงในอนาคตก่อให้เกิดความคล่องตัวในการใช้พื้นที่

2)ความสะดวกของงานระบบพื้นคอนกรีตอัดแรงเป็นแผ่นพื้นท้องเรียบโดยไม่มีคานทำให้การเดินท่อสำหรับงานระบบต่าง ๆ เช่น ไฟฟ้า ประปา ระบบปรับอากาศ ทำได้สะดวก

3)ลดความสูงของอาคารพื้นคอนกรีตอัดแรงหนาเพียง 1/40 - 1/45 ของความยาวช่วงเสาและไม่ต้องมีคานซึ่งมีผลให้ลดความสูงของชั้นได้ตั้งแต่ 30- 60 ซม.ดังนั้นเมื่อเปรียบเทียบกับอาคารคอนกรีตเสริมเหล็ก อาคารคอนกรีตอัดแรงจะมีความสูงของอาคารน้อยกว่าเมื่อมีจำนวนชั้นเท่ากัน และในกรณีที่ความสูงเท่ากันอาคารที่ใช้พื้นคอนกรีตอัดแรงจะทำให้จำนวนชั้นและพื้นที่ใช้สอยมากกว่าซึ่งจะให้ผลตอบแทนการลงทุนในที่ดินสูงกว่าโดยเฉพาะอย่างยิ่งสำหรับอาคารที่ก่อสร้างในพื้นที่ที่มีการจำกัดความสูงอาคาร

พื้นคอนกรีตเสริมเหล็กกับพฤติกรรมโครงสร้าง

<u>ด้านโครงสร้าง</u>

1)ลดการแอ่นตัวของแผ่นพื้นการอัดแรงในพื้นคอนกรีตเสริมเหล็กนั้นนอกจากจะทำให้เกิดหน่วยแรงอัดในแผ่นพื้นแล้วการวางลวดอัดแรงที่โค้งเป็นรูปพาราโบลาตามโพรไฟล์ที่กำหนดไว้จะทำให้เกิดการยกตัวซึ่งมีทิศทางตรงข้ามกับน้ำหนักบรรทุกคงที่และน้ำหนักบรรทุกจรโดยทั่วไปแรงยกตัวจะมีขนาดน้อยกว่าน้ำหนักบรรทุกเล็กน้อยทำให้สภาพที่ไม่มีน้ำหนักบรรทุกจรเต็มที่นั้น แผ่นพื้นจะมีการแอ่นตัวน้อยมากหรือแทบไม่มีเลย ส่วนในสภาพที่พื้นรับน้ำหนักบรรทุกเต็มที่นั้น หน่วยแรงอัดในพื้นจะช่วยให้การแอ่นตัวน้อยลง

2)ทนต่อการแตกร้าวได้ดีขึ้น การที่แผ่นพื้นอยู่ภายใต้หน่วยแรงอัดและมีการแอ่นตัวน้อยทำให้โอกาสที่ จะเกิดรอยร้าวในการทำงานเป็นไปได้ยาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ทำซ้ำ ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3)สภาพการใช้งานดีในกรณีที่มีการบรรทุกน้ำหนักเกินมาก ๆ พื้นอาจจะเกิดรอยร้าวขึ้นแต่เมื่อเอาน้ำหนักบรรทุกที่เกินออก รอยร้าวจะปิด พื้นจะมีสภาพเกือบไม่เปลี่ยนแปลงและจะสามารถรับน้ำหนักบรรทุกได้เหมือนเดิม

4)หน่วยแรงในลวดอัดแรงมีการเปลี่ยนแปลงน้อยในกรณีที่น้ำหนักบรรทุกมีการเปลี่ยนแปลงเนื่องจากการสั่น

5)โครงสร้างมีความเหนียว พื้นคอนกรีตอัดแรงสามารถรับแรงเนื่องจากแผ่นดินไหวได้ดีโดยมีค่า การแอ่นตัวที่ขอบที่มีการพังทลายสูงกว่าโครงสร้างคอนกรีตเสริมเหล็กถึง 50 เปอร์เซ็นต์

6)ความต้านทานไฟไหม้ของพื้นคอนกรีตอัดแรงในการออกแบบได้คำนึงถึงความต้านทานไฟไหม้ของพื้นคอนกรีตอัดแรงที่จะทนทานต่อไฟไหม้ที่กำหนดไว้ใน Post tensioning Institute (PT) ได้รายงานไว้ดังนี้

ตารางที่ 1-5

ความต้านทานไฟ (ชั่วโมง)	ระยะหุ้มไม่ต่ำกว่า (มม.)	ความหนาพื้นไม่ต่ำกว่า (มม.)
0.5	10-15	60-90
1.0	15-19	70-100
1.5	19-25	90-125
2.0	19-30	110-127
3.0	25-40	150-160
4.0	32-50	150-177

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเปรียบเทียบระหว่างพื้นคอนกรีตอัดแรงกับพื้นหล่อกับที่ชนิดไร้คานและหล่อบนพื้นคาน

อาคารคอนกรีตอัดแรง

-กำลังคอนกรีตที่ใช้สูงกว่า โครงสร้างมีความทนทานมากกว่า

-พื้นคอนกรีตบางกว่า

-น้ำหนักคอนกรีตน้อยกว่า

-น้ำหนักลงบนเสาเข็มน้อยกว่าทำให้ประหยัดค่าก่อสร้างฐานราก

-ลดความสูงระหว่างชั้น ส่วนสูงเท่ากันได้จำนวนชั้นมากกว่า

-ก่อสร้างได้รวดเร็วกว่า

-ราคาค่าก่อสร้างถูกกว่า เพราะปริมาณเหล็กเสริมและคอนกรีตน้อยกว่า

-ทำให้เปิดใช้อาคารได้เร็วขึ้น ทำให้ได้กำไรเรื่องดอกเบี้ยของค่าก่อสร้าง

-การแอ่นตัวน้อยกว่าหรือลดการแอ่นตัวของโครงสร้าง

-เนื่องจากโครงสร้างไม่ยอมให้มีการแตกร้าวทำให้เกิด corrosion ยากกว่า

-มีความทึบน้ำมากกว่าทำให้น้ำซึมผ่านได้ยาก

-พฤติกรรมการรับแรงเฉือนสูงกว่า

-ช่วงระยะห่างของเสาสามารถออกแบบให้มีช่วงยาวและมีความผอมมากกว่า

-โครงสร้างมีความเหนียวสูงกว่า

-ในกรณีมีน้ำหนักบรรทุกเกินถ้าเอาน้ำหนักส่วนเกินนั้นออกจะทำให้โครงสร้างเข้าสู่สภาพเดิมได้ตามปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การป้องกันไฟไหม้ได้นานใกล้เคียงกันทั้งนี้ขึ้นอยู่กับระยะคอนกรีตหุ้มตามมาตรฐานกำหนด

-การซ่อมแซมเปิดช่องเจาะสามารถทำได้

-การเดินระบบ ไฟฟ้า สุขาภิบาลทำได้ง่ายกว่า

-สามารถก่อกำแพง ผนังกั้นห้องได้ทุกตำแหน่ง

อาคารคอนกรีตหล่อกับที่

-กำลังคอนกรีตที่ใช้ต่ำกว่า ความคงทนน้อยกว่า

-พื้นคอนกรีตหนากว่า

-น้ำหนักโครงสร้างมากกว่า

-น้ำหนักลงบนเสาเข็มมากกว่า ทำให้ค่าก่อสร้างฐานรากสูงกว่า

-ความสูงของอาคารมากกว่า ทำให้ได้จำนวนชั้นของอาคารน้อยกว่า

-การก่อสร้างใช้เวลานานกว่า

-ราคาค่าก่อสร้างมากกว่า เพราะใช้ปริมาณคอนกรีต และเหล็กเสริมมากกว่า

-เปิดใช้อาคารได้ช้ากว่า ทำให้ดอกเบี้ยค่าก่อสร้างสูงกว่า

-การแอ่นตัวของโครงสร้างสูงกว่า

-เกิด corrosion สูงกว่าทำให้เกิดสนิมในเหล็กเสริมได้ง่าย

-มีความทึบน้ำน้อยกว่า ทำให้น้ำซึมผ่านได้ง่ายกว่า

-พฤติกรรมการรับแรงเฉือนน้อยกว่า

-ช่วงความยาวสั้นกว่าและผอมได้น้อยกว่า

-โครงสร้างมีความเหนียวน้อยกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ในกรณีที่น้ำหนักบรรทุกเกิน เมื่อเอาน้ำหนักออกแล้วโครงสร้างจะอยู่ในสภาพเดิม เหมือนเมื่อขณะมีน้ำหนักบรรทุกเกินคือโครงสร้างจะไม่กลับสู่สภาพปกติ

-การซ่อมแซม เจาะช่องเปิด สามารถทำได้โดยสะดวก

-การเดินท่อระบบสุขาภิบาล ไฟฟ้าทำได้ยากกว่า

-การก่อกำแพงผนังกั้นห้อง สามารถทำได้เฉพาะก่อบนคานเท่านั้น หรือก่อบนพื้นได้ถ้าพื้นมีความหนามาก ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
037195

บทที่ 2

วัสดุสำหรับงานคอนกรีตอัดแรง

คอนกรีต

คอนกรีตที่ใช้กับงานคอนกรีตอัดแรงจำเป็นต้องเป็นคอนกรีตกำลังสูง มีกำลังประลัยของทรงกระบอกคอนกรีตขนาด 15 ซม. x 30 ซม. ที่อายุ 28 วัน ตั้งแต่ 350-600 กก/ซม.2

ทั้งนี้เพราะคอนกรีตจะต้องต้านทานหน่วยแรงที่สูงมากที่บริเวณปลายคาน ในขณะที่มีการถ่ายแรงอัดแรกเริ่มเข้าสู่องค์อาคาร นอกจากนี้การใช้คอนกรีตกำลังสูงจะทำให้เกิดการคืบและการหดตัวน้อยลง

คอนกรีตเป็นวัสดุประกอบมีส่วนผสมสำคัญคือปอร์ตแลนด์ซีเมนต์ มวลหยาบ มวลละเอียดและน้ำซีเมนต์ที่ใช้ปกติเป็นไปตามมาตรฐาน ASTM C 150 ชนิดที่ 1 เป็นปอร์ตแลนด์ซีเมนต์สำหรับงานโครงสร้างทั่วไปซึ่งรวมถึงโครงสร้างคอนกรีตเสริมเหล็กด้วยในระบบการผลิตขององค์อาคารแบบอัดแรงก่อนมักจะต้องอัดแรงในเวลาที่เร็วที่สุดเพื่อเหตุผลทางด้านเศรษฐศาสตร์ การออกแบบส่วนผสมคอนกรีตจึงมักใช้ปอร์ตแลนด์ซีเมนต์ชนิดที่ 3 ซึ่งเป็นซีเมนต์ที่ทำให้คอนกรีตมีกำลังสูงในเวลาสั้นหลังจากที่คอนกรีตแข็งตัว กำลังของคอนกรีตที่ใช้ซีเมนต์ชนิดที่ 13 ที่มีอายุ 7 วัน จะเกือบเท่าคอนกรีตที่ใช้ซีเมนต์ชนิดที่ 1 มีอายุ 28วัน

การที่จะออกแบบส่วนผสมคอนกรีตให้มีกำลังมากกว่า 350 กก/ซม.2 จะต้องควบคุมอัตราส่วนน้ำต่อซีเมนต์ไม่เกิน 0.45 โดยน้ำหนัก เพื่อสะดวกในการเทส่วนยุบตัวจะมีค่าระหว่าง 5 ซม. ถึง 10 ซม. ปริมาณซีเมนต์ที่ใช้จะเท่ากับ 400 กก.ต่อคอนกรีต 1 ลูกบาสก์เมตร เนื่องจากความเครียดเนื่องจากการคืบและการหดตัวที่มีค่ามากขึ้นตามปริมาณของซีเมนต์ดังนั้น การใช้ส่วนผสมคอนกรีตที่แห้งและใช้ซีเมนต์น้อยลง และเพิ่มสารผสมเพิ่ม จะช่วยลดปริมาณการคืบและการหดตัว

การควบคุมคุณภาพของคอนกรีตกำลังสูง ตลอดจนขั้นตอนการเทจะต้องเข้มงวดและควบคุมสม่ำเสมอในสนาม ทั้งนี้เนื่องจากส่วนต่าง ๆ ขององค์อาคารคอนกรีตอัดแรงจะเกิดหน่วยแรงในช่วงอัดแรงและช่วงใช้งานปกติ เช่นหน้าตัดที่กึ่งกลางคานในขณะถ่ายแรงอัดผิวล่างของคานจะเกิดหน่วยแรงอัดผิวบนจะเกิดหน่วยแรงดึง แต่ในขณะใช้งานเมื่อน้ำหนักบรรทุกปรกติกระทำเต็มที่หน่วยแรงดึงจะเกิดขึ้นที่ผิวล่าง โดยที่ผิวบนจะเกิดหน่วยแรงอัด สำหรับส่วนปลายขององค์อาคารจะเกิดหน่วยแรงกดหรือหน่วยแรงยึดเกาะสูงขณะถ่ายแรง ดังนั้นการควบคุมคุณภาพในการก่อสร้างงานคอนกรีตอัดแรงจะต้องพิถีพิถันกว่างานคอนกรีตเสริมเหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสัมพันธ์ระหว่างหน่วยแรงและความเครียด

จากรูปที่ 2.1 ก แสดงกราฟความสัมพันธ์ระหว่างหน่วยแรงและความเครียดของคอนกรีตที่อายุ 28 วันซึ่งมีกำลังประลัยต่าง ๆ กันเนื่องจากการอัดในแนวแกน จะเห็นว่าในช่วงที่หน่วยแรงมีค่าน้อยกว่าประมาณ $0.5fc'$ โดยที่ fc' คือกำลังประลัยของคอนกรีตทรงกระบอก ความสัมพันธ์ระหว่างหน่วยแรงและความเครียดเกือบจะเป็นเส้นตรง เมื่อคอนกรีตมีกำลังสูงขึ้นความชันของกราฟจะเพิ่มขึ้น ค่าความเครียดที่จุดสูงสุดของกราฟมีค่าประมาณ 0.002 และอัตราการลดหน่วยแรงเมื่อความเครียดมากกว่า 0.002 จะเร็วขึ้นเมื่อคอนกรีตมีกำลังสูงขึ้น ในรูปที่ แสดงกราฟหน่วยแรงและความเครียดของคอนกรีตที่มีกำลังประลัยค่าหนึ่ง ในช่วงหน่วยแรงน้อย ๆ ความชันของเส้นตรง OA เรียกว่าโมดูลัสความยืดหยุ่นของคอนกรีต(Secant modulus of elasticity) , Ec ค่าโมดูลัสนี้เปลี่ยนแปลงไปตามน้ำหนักจำเพาะของคอนกรีต สำหรับคอนกรีตมวลปกติสามารถเขียนเป็นฟังค์ชั่นได้ดังนี้

$$Ec = 0.043\, w^{1.5} \sqrt{fc'} \quad Mpa \quad \text{----------A}$$

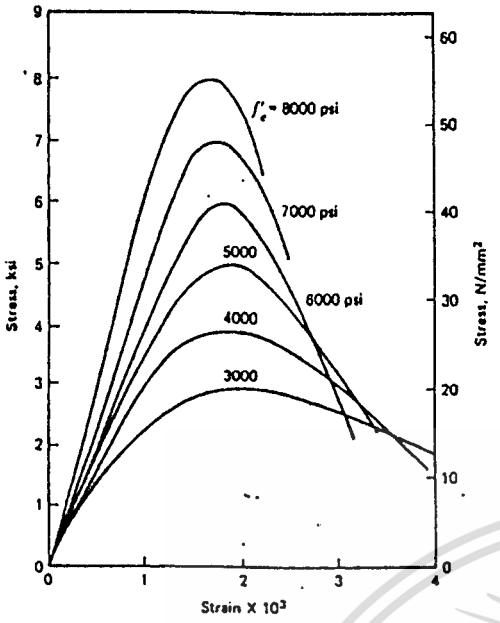
โดยที่ w คือ น้ำหนักจำเพาะของคอนกรีตเป็นกิโลนิวตันต่อลูกบาสก์เมตร และ fc' มีหน่วยเป็นกิโลนิวตันต่อตารางมิลลิเมตร สำหรับคอนกรีตมวลปกติ w มีค่าเท่ากับ 2400 กิโลนิวตันต่อลูกบาสก์เมตรซึ่งสามารถเขียนสูตรใหม่ได้เป็น

$$Ec = 4730\sqrt{fc'} \quad \text{----------B}$$

โมดูลัสความยืดหยุ่นของคอนกรีตเมื่อทดสอบโดยการดึงก่อนที่จะเกิดการแตกร้าวถือว่ามีค่าเท่ากับเมื่อทดสอบโดยการอัด ค่าโมดูลัสนี้คำนวนจากการยืดตัวต่อความยาวหลายเท่าของเส้นผ่าศูนย์กลางของแท่งตัวอย่าง กำลังประลัยของคอนกรีตในเชิงการดึงมีค่าน้อยมากและขึ้นอยู่กับวิธีทดลอง กำลังดึงประลัยที่ได้จากการทดสอบการดัดของคานคอนกรีตล้วนรูสี่เหลี่ยมเรียกว่า โมดูลัสของการหักซึ่งมีค่าเท่ากับ $0.62\sqrt{fc'}$

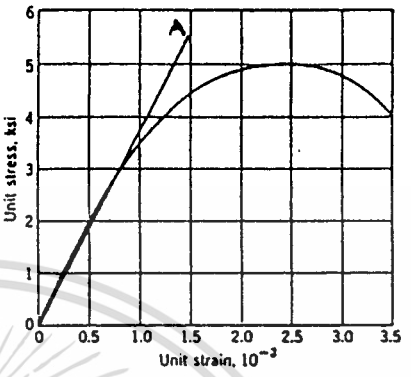
เมื่อคอนกรีตหน่วยแรงอัดกระทำในทิศทางหนึ่ง จะเกิดความเครียดในทิศทางของการอัด อัตราส่วนของความเครียดด้านข้างกับความเครียดในแนวแกน เรียกว่า อัตราส่วนปัวซอง มีค่าเปลี่ยนแปลงระหว่าง 0.15 - 0.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

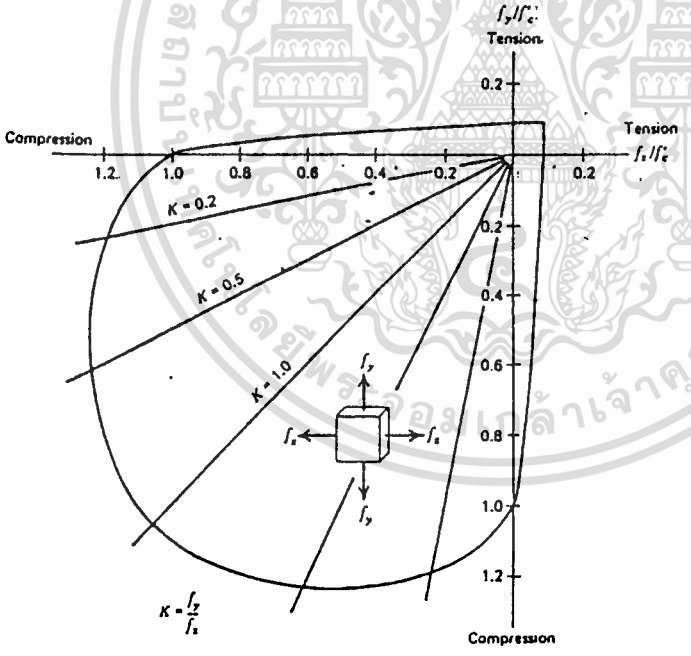


◀ ก. หน่วยแรงและความเครียดของคอนกรีต

▼ ข. อิลาสติกโมดูลัส



รูปที่ 2.1



รูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำลังของคอนกรีตภายใต้หน่วยแรงสองทิศทาง

ในส่วนใดส่วนหนึ่งของโครงสร้างคอนกรีตอิลิเมนต์ของคอนกรีตอาจมีหน่วยแรงตามแนวแกนและหน่วยแรงเฉือนกระทำในระนาบ ในบางกรณีพิเศษเช่นในเขื่อนคอนกรีตหน่วยแรงที่เกิดขึ้นจะเป็นหน่วยแรงในสภาวะหน่วยแรงระนาบ โดยการวิเคราะห์หน่วยแรงระนาบโดยการแตกหน่วยแรงเข้าสู่แกนหลัก จะได้หน่วยแรงมุขสำคัญสองค่าคือค่าที่มากที่สุดและค่าที่น้อยที่สุด ค่าหน่วยแรงมุขสำคัญในสองทิศทางนี้สามารถเขียนได้บนระนาบในรูปที่ จากการศึกษาของ Kufer, Hilsdorf และRusch จะได้เอ็นวิล็อปแสดงความสัมพันธ์ของหน่วยแรงมุขสำคัญทั้งสองทิศทางกำลังอัดในทิศทางหนึ่งจะสูงกว่ากำลังอัดประลัยในทิศทางเดียว จะเห็นได้ว่าในบริเวณที่มีการอัดสองทิศทาง บนกราฟนี้ค่าหน่วยแรงมุขสำคัญจะทำให้เป็นค่าไร้มิติโดยการหารด้วยค่ากำลังประลัยในทิศทางเดียว จะเห็นได้ว่าในบริเวณที่เกิดการอัดสองทิศทางกำลังอัดในทิศทางหนึ่งจะสูงกว่ากำลังอัดประลัยในทิศทาเดียวประมาณ 20 เปอร์เซ็นต์ ในกรณีที่หน่วยแรงมุขสำคัญในทิศทางตั้งฉากมีค่าประมาณ 0.5 - 0.6 ของกำลังอัดทิศทางเดียวในบริเวณที่คอนกรีตเกิดหน่วยแรงดึงทั้งสองทิศทาง กำลังดึงของคอนกรีตในทิศทางหนึ่งจะไม่มีผลต่อกำลังดึงของคอนกรีตในอีกทิศทางหนึ่งในบริเวณที่คอนกรีตมีหน่วยแรงอัดในทิศทางหนึ่ง และหน่วยแรงดึงในอีกทิศทางหนึ่งกราฟความเกี่ยวพันจะเป็นเส้นตรง ดังนั้นหน่วยแรงดึงที่เกิดขึ้นในทิศทางตั้งฉากเพียงเล็กน้อยจะลดกำลังอัดในทิศทางแนวแกนเป็นอันมาก

การเปลี่ยนรูปของคอนกรีตเป็นฟังค์ชั่นของเวลา

การเปลี่ยนรูปของคอนกรีตเนื่องจากการคืบและการหดตัวมีผลทำให้เกิดการสูญเสียแรงอัดในองค์อาคารคอนกรีตอัดแรง นอกจากนี้ทำให้ค่าความโค้งเพิ่มขึ้นเป็นฟังค์ชั่นของเวลาทำให้การแอ่นตัวขององค์อาคารเพิ่มมากขึ้น

การคืบ

การคืบเป็นคุณสมบัติของคอนกรีตหรือของวัสดุหลายชนิดที่เกิดการเปลี่ยนรูปเพิ่มขึ้นเนื่องจากแรงที่กระทำชนิดถาวรบนคอนกรีต เมื่อคอนกรีตมีแรงกระทำจะเกิดการเปลี่ยนรูปทันทีทันใด หลังจากนั้นจะเกิดการคืบโดยมีอัตราการเพิ่มของความเครียดรวดเร็วในเวลาแรกเริ่ม และอัตราการคืบจะค่อย ๆ ลดลงจนถึงค่าคงที่ค่าหนึ่งเมื่อเวลาผ่านไปนาน ๆ ความเครียดเนื่องจากการคืบขึ้นอยู่กับองค์ประกอบหลายประการได้แก่ ส่วนประกอบของคอนกรีต ความชื้นสัมพัทธ์สภาพของการบ่มและอายุของคอนกรีตขณะที่แรงกระทำ จากการวิจัยพบว่าความเครียดเนื่องจากการคืบ ,Cu ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Cu = \varepsilon_{cu} / \varepsilon_{ci} \qquad \text{----------C}$$

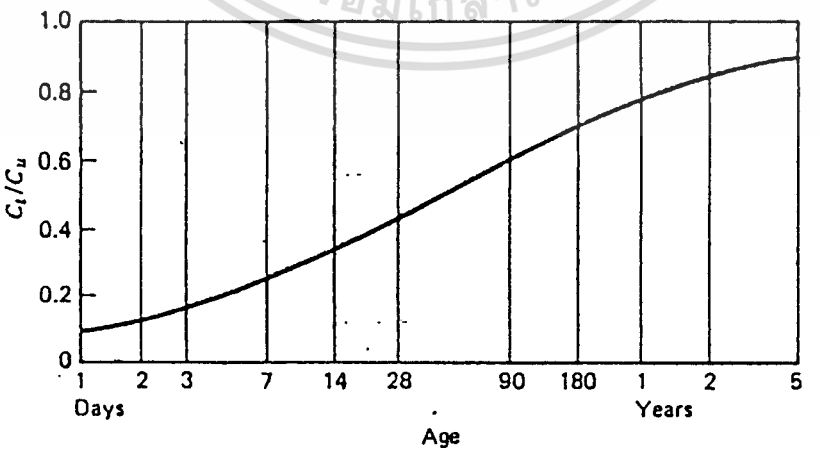
โดยที่ ε_{cu} คือความเครียดที่เพิ่มขึ้นทั้งหมดเนื่องจากการคืบ และ ε_{ci} คือความเครียดแรกเริ่มซึ่งเกิดขึ้นทันทีที่มีหน่วยแรงกระทำบนคอนกรีตในบางครั้งปริมาณการคืบระบุโดยค่าการคืบจำเพาะδ_uซึ่งเท่ากับอัตราส่วนของความเครียดเนื่องจากการคืบทั้งหมดต่อหน่วยแรงที่กระทำต่อคอนกรีตนั่นคือ

$$\delta_u = \varepsilon_{ci} / f_c \qquad \text{----------D}$$

ในการหาค่าสัมประสิทธิ์การคืบขณะใด ๆ Ct Brason ได้ทำการวิจัยการคืบของแท่งคอนกรีตตัวอย่างภายใต้สภาพมาตรฐานกล่าวคือคอนกรีตมีค่าการยุบตัวไม่มากกว่า 10 ซม. มีความลึกไม่มากกว่า 15 ซม. ภานใต้ความชื้นสัมพัทธ์ 40 เปอร์เซ็นต์ คอนกรีตมีอายุ 7 วัน และ 3 วัน สำหรับลักษณะการบ่มแบบใช้งานและไอน้ำตามลำดับได้เสนอว่าสัมประสิทธิ์การคืบขณะใด ๆ เป็นฟังค์ชั่นของสัมประสิทธิ์การคืบทั้งหมดดังนี้

$$ct = \frac{t^{0.60}}{10 + t^{0.60}} cu \qquad \text{----------E}$$

โดยที่ t เป็นเวลานับตั้งแต่มีแรงกระทำมีหน่วยแรงกระทำมีหน่วยเป็นวัน ฟังค์ชั่นแสดงเป็นกราฟดังในรูปที่ 2.3



รูปที่ 2.3 การเปลี่ยนแปลงของสัมประสิทธิ์การคืบกับเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่สภาพของคอนกรีตต่างจากสภาพมาตรฐาน ค่าสัมประสิทธิ์การคืบจะต้องคูณด้วยแฟกเตอร์ปรับค่าเนื่องจากอายุของคอนกรีตดังนี้คือ

$\alpha_1 = 1.25t_{age}^{-0.118}$ สำหรับคอนกรีตบ่มด้วยน้ำ

$\alpha_2 = 1.13t_{age}^{-0.095}$ สำหรับคอนกรีตบ่มด้วยไอน้ำ

และแฟกเตอร์ปรับค่าเนื่องจากความชื้นสัมพัทธ์ที่สูงกว่า 40 เปอร์เซ็นต์ คือ

$$\alpha_3 = 1.27 - 0.0067H$$

โดยที่ tage เป็นอายุของคอนกรีตมีหน่วยเป็นวันและ H เป็นค่าความชื้นสัมพัทธ์เป็นเปอร์เซ็นต์

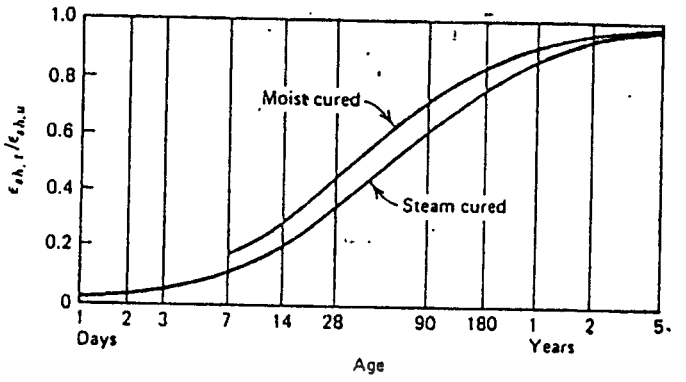
การหดตัว

การหดตัวคือการเปลี่ยนรูปของคอนกรีตเนื่องจากการสูญเสียความชื้นอิสระในมวลคอนกรีตซึ่งเป็นฟังค์ชั่นของเวลาการเปลี่ยนรูปจะเกิดขึ้น ถึงแม้ว่าคอนกรีตจะเกิดขึ้นถึงแม้ว่าคอนกรีตไม่คอนกรีตไม่มีหน่วยแรงกระทำ การสูญเสียความชื้นจะมีอัตราเร็วในช่วงแรกเริ่มและค่อย ๆ ลดลงจนปริมาณของคอนกรีตคงที่เมื่อเวลาผ่านไปนาน ๆ ความเครียดการหดตัวทั้งหมดของคอนกรีต ε_{sh}^{*} เท่ากับ 0.0008 สำหรับคอนกรีตบ่มด้วยน้ำ และเท่ากับ 0.00073 สำหรับคอนกรีตบ่มด้วยไอน้ำ อัตราการหดตัวขึ้นอยู่กับองค์ประกอบต่าง ๆ ได้แก่ความชื้นสัมพัทธ์ของบรรยากาศ อุณหภูมิ ขนาด และรูปร่างขององค์อาคารสำหรับในสภาพมาตรฐาน Branson ได้เสนอฟังค์ชั่นของความเครียดการหดตัวขณะใด ๆ ε_{sh} หาได้จาก

$\varepsilon_{sh} = \frac{t}{35+t}\varepsilon_{sh}^{*}$ สำหรับคอนกรีตบ่มด้วยน้ำ

$\varepsilon_{sh} = \frac{t}{55+t}\varepsilon_{sh}^{*}$ สำหรับคอนกรีตบ่มด้วยไอน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การเปลี่ยนแปลงของสัมประสิทธิ์การหดตัวกับเวลา

ในกรณีที่ความชื้นสัมพัทธ์ของบรรยากาศมากกว่า 40 เปอร์เซ็นต์ ค่าสัมประสิทธิ์การหดตัวจะต้องปรับด้วยค่าแฟกเตอร์ต่อไปนี้

$$\alpha_3 = 1.4 - 0.010H \qquad 40 \le H \le 80\%$$
$$\alpha_4 = 3.00 - 0.030H \qquad 80 \le H \le 100\%$$

เหล็กเสริม

เหล็กเสริมที่ใช้งานคอนกรีตอัดแรงเป็นเหล็กกำลังสูงมีอยู่ 3 ชนิด คือ ลวดเหล็กกำลังสูง ลวดสะแตรนด์และท่อนเหล็กกำลังสูง การใช้เหล็กกำลังสูงเหมาะสมสำหรับคอนกรีตอัดแรง เพราะทำให้แรงอัดองค์อาคารเมื่อหักการสูญเสียแรงอัดแล้วมีค่ามากพอที่หักล้างแรงกระทำภายนอก

ลวดเหล็กกำลังสูง

ลวดเหล็กกำลังสูงผลิตตามมาตรฐาน ASTM Specification A421 : Uncoated Stress-Relieved Wire for Prestress Concrete การผลิตกระทำโดยการนำแท่งเหล็กกล้ามาหลอมด้วยความร้อนและรีดเป็นท่อนเหล็ก เมื่อท่อนเหล็กเย็นลงจึงนำมารีดเย็นผ่านลูกกลิ้งหลาย ๆ ครั้ง จนกระทั่งได้ลวดขนาดที่ต้องการขบวนการรีดเย็นเนื้อเหล็กเกิดโคลด์เวิร์ค ทำให้โครงสร้างของผลึกเปลี่ยนแปลงไป เป็นผลทำให้กำลังของลวดเหล็กเพิ่มขึ้น ลวดเหล็กหลังจากการรีดเย็นแล้วนำมาทำให้ร้อนแล้วปล่อยให้เย็นตัวช้า ๆ เพื่อลดหน่วยแรงภายใน ซึ่งทำให้ความเหนียวของลวดเพิ่มขึ้น ลวดกำลังสูงนี้แบ่งตามรูปร่างมีสองชนิดคือ ชนิดปลายยึดแบบกลม(Type BA) และชนิดปลายยึดแบบลิ่ม(Type WA) และมีเส้นผ่านศูนย์กลาง 4 ขนาดดังแสดงในตารางที่ 2-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2-1 คุณสมบัติของลวดกำลังสูงตามมาตรฐาน ASTM A421

Nominal Diameter in.(mm)	Minimum Tensile Strength psi (N/mm^2)		Minimum Stress at 1% Extension psi (N/mm^2)	
	Type BA	Type WA	Type BA	Type WA
0.192(4.88)	a	250,000(1725)	a	200,000(1380)
0.196(4.98)	240,000(1655)	250,000(1725)	192,000(1325)	200,000(1380)
0.250(6.35)	240,000(1655)	240,000(1655)	192,000(1325)	192,000(1325)
0.276(7.01)	a	235,000(1622)	a	188,000(1295)

[a] These sizes are not commonly furnished in Type BA wire.

ลวดกำลังสูงใช้สำหรับการก่อสร้างแบบดึงเหล็กทีหลังและแบบดึงเหล็กก่อน ในกรณีแรก tendon เส้นหนึ่งจะประกอบด้วยลวดหลาย ๆ เส้นวางเป็นแนวขนานกับภายในท่อ คอนดุทกำลังดึงประลัยของลวดกำลังสูงมีค่าระหว่าง 1620 - 1725 นิวตันต่อ ตารางมิลลิเมตรหน่วยแรงดึงที่ความเครียด 0.01 มีค่าระหว่าง 1295 - 1380 นิวตันต่อตารางมิลลิเมตร

ลวดสะแตรนด์

ลวดสะแตรนด์สำหรับงานคอนกรีตอัดแรงผลิตตามมาตรฐาน ASTM Specification A416 : Unbonded Seven - Wire Stress Relieved Strand for Prestressed Concrete ประกอบด้วยลวดกำลังสูง 7 เส้น โดยนำลวด 6 เส้นพันรอบเส้นที่ 7 ซึ่งเป็นแกนกลาง ระยะพิทช์ของการพันเปลี่ยนแปลงระหว่างค่า 12 เท่า ถึง 16 เท่าของเส้นผ่านศูนย์กลางลวดสะแตรนด์ที่ผลิตมีอยู่สองเกรดคือ เกรด 250 และเกรด 270 และมีเส้นผ่านศูนย์กลางของกลุ่มลวดตั้งแต่ 6.35 มม ถึง15.24 มม. ดังแสดงในตารางที่ ค่ากำลังประลัยของลวดสะแตรนด์เป็น 1725 นิวตันต่อตารางมิลลิเมตร สำหรับเกรด 250 และ เท่ากับ 1860 นิวตันต่อตารางมิลลิเมตร สำหรับเกรด 270

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2-2 คุณสมบัติลวดสะแตรนด์ตามมาตรฐาน ASTM A416

Nominal Diameter in.(mm)	Breaking Strength lb (kN)	Nominal Area of Strand $in.^2$ (mm^2)	Minimum Load at 1% Extension lb (kN)
	Grade 250		
0.250(6.35)	9000(40.0)	0.036(23.22)	7650(34.0)
0.313(7.94)	14,500(64.5)	0.058(37.42)	12,300(54.7)
0.375(9.53)	20,000(89.0)	0.080(51.61)	17,000(75.6)
0.438(11.11)	27,000(120.1)	0.108(69.68)	23,000(102.3)
0.500(12.70)	36,000(160.1)	0.144(92.90)	30,600(136.2)
0.600(15.24)	54,000(240.2)	0.216(139.35)	45,900(204.2)
	Grade 270		
0.375(9.53)	23,000(102.3)	0.085(54.84)	19,550(87.0)
0.438(11.11)	31,000(137.9)	0.115(74.19)	26,350(117.2)
0.500(12.70)	41,300(183.7)	0.153(98.71)	35,100(156.1)
0.600(15.24)	58,600(260.7)	0.217(140.00)	49,800(221.5)

ท่อนเหล็กโลหะผสม

ในการผลิตท่อนเหล็กโลหะผสมกำลังสูงส่วนผสมของเหล็กจะเติมธาตุบางชนิด เช่น แมงกานิส ซิลิกอน และโครเมี่ยม นอกจากนี้กำลังดึงของท่อนเหล็กเกิดจากกระบวนการโคลเวิร์คด้วย นอกจากนี้กำลังดึงของท่อนเหล็กเกิดจากขบวนการโคลเวิร์คด้วย นอกจากนี้ความเหนียวของท่อนเหล็กทำให้เพิ่มขึ้นโดยขบวนการเพิ่มความร้ อน ท่อนเหล็กโลหะผสมผลิตตามมาตรฐาน ASTM Specification A 722 : Uncoated High Strength Steel Bar for Prestressing Concrete ขนดของท่อนเหล็กมีตั้งแต่ 12.70 มม. ถึง 34.93 มม. ท่อนโลหะผสมมีสองเกรด คือเกรด 145 มีกำลังประลัย 1000 นิวตันต่อตารางมิลลิเมตร และเกรด 160 ซึ่งกำลังดึงประลัยเป็น 1100 นิวตันต่อตารางมิลลิเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2-3 คุณสมบัติของท่อนเหล็กโลหะผสม

Nominal Diameter in.(mm)	Nominal Area of Bar in^2 (mm^2)	Breaking Strength lb(kN)	Minimum Load at 0.7% Extension lb (kN)
		Grade 145	
½(12.70)	0.196(127)	28,000(125)	25,000(111)
⅜ (15.88)	0.307(198)	45,000(200)	40,000(178)
½(19.05)	0.442(285)	64,000(285)	58,000(258)
⅞(22.23)	0.601(388)	87,000(387)	78,000(347)
1(25.40)	0.785(507)	114,000(507)	102,000(454)
1⅛(28.58)	0.994(642)	144,000(641)	129,000(574)
1¼(31.75)	1.227(792)	178,000(792)	160,000(712)
1⅜ (34.93)	1.485(958)	215,000(957)	193,000(859)
		Grade 160	
½(12.70)	0.196(127)	31,000(138)	27,000(120)
⅜ (15.88)	0.307(198)	49,000(218)	43,000(191)
½(19.05)	0.442(285)	71,000(316)	62,000(276)
⅞(22.23)	0.601(388)	96,000(427)	84,000(374)
1(25.40)	0.785(507)	126,000(561)	110,000(490)
1⅛(28.58)	0.994(642)	159,000(708)	139,000(619)
1¼(31.75)	1.227(792)	196,000(872)	172,000(765)
1⅜ (34.93)	1.485(958)	238,000(1059)	208,000(926)

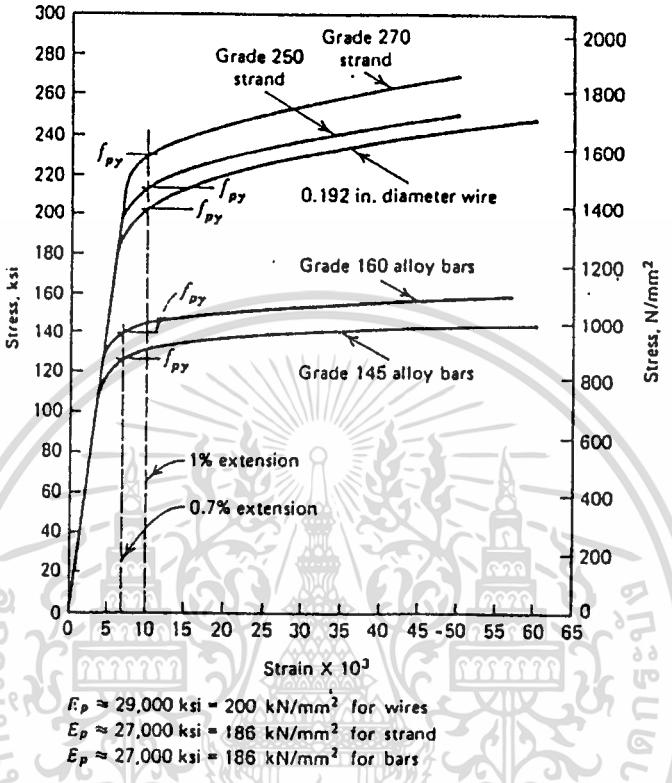
ในประเทศไทยเหล็กเสริมสำหรับงานคอนกรีตอัดแรงนิยมใช้ชนิดลวดกำลังสูง และลวดสะแตรนด์ ขนาดของลวดกำลังสูงมีเส้นผ่าศูนย์กลาง 4 มม., 5 มม.,6 มม.และ 7 มม.

คุณสมบัติของหน่วยแรงและความเครียดของเหล็กกำลังสูง

คุณสมบัติทางกลของเหล็กกำลังสูงแสดงในรูปของไดอะแกรมหน่วยแรงและความเครียดในรูปที่ จะเห็นว่าเหล็กกำลังสูงทั้งสามชนิดไม่มีจุดคลาก อัตราการเพิ่มของหน่วยแรงค่อย ๆ ลดลงเมื่อความเครียดเพิ่มขึ้นมากกว่าความเครียดที่พิกัดเส้นตรงจนกระทั่งถึงค่ากำลังประลัย ค่ากำลังคลากหาได้จากการกำหนดค่าความเครียด เช่นความเครียดที่ 0.010 สำหรับลวดกำลังสูงหรือลวดสะแตรนด์ยารดเดนนิ่ง และความเครียดที่จุดแตกหักอยู่ในช่วง 0.05 – 0.06 ซึ่งน้อยกว่าค่าดังกล่าวของเหล็กกล้าละมุนมาก โมดูลัสความยืดหยุ่นของลวดกำลังสูงมีค่าเท่ากับค่าดังกล่าวของเหล็กกล้าละมุนคือ 200,000 นิวตันต่อตารางมิลลิเมตร โมดูลัสในลวดสะแตรนด์เป็น 186,000 นิวตันต่อตารางมิลลิเมตรมีค่าใกล้เคียงกับค่าดังกล่าวของท่อนเหล็กโลหะผสม ในลวดสะแตรนด์ขณะทดลอง โดยการดึงเส้นลวดที่พันอยู่รอบเส้นแกนจะพยายามยืดออก

เอกสารนี้มีลิขสิทธิ์ ห้ามนำไปใช้เพื่อการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งนี้เนื่องจากการเยื้องศูนย์ของแรงดึงเทียบกับจุดเซ็นทรอยของเส้นลวดดังกล่าวทำให้การยืดตัวของลวดทั้งกลุ่มมากกว่าค่าดังกล่าวของลวดเดี่ยว ส่วนท่อนเหล็กโลหะผสมมีค่าโมดูลัสต่ำเนื่องจากการผสมท่อนโลหะอื่น ๆ ในส่วนผสมของเหล็ก



รูปที่ 2-5 ความสัมพันธ์ระหว่างหน่วยแรงและความเครียดของเหล็กกำลังสูง

นอกจากเหล็กกำลังสูงแล้ว องค์อาคารคอนกรีตอัดแรงมีการเสริมเหล็กกล้าละมุนเพื่อต้านทานหน่วยแรงเฉือน หน่วยแรงดึงบางส่วนที่เกิดขึ้นในองค์อาคารชนิดอัดแรงบางว่วนและหน่วยแรงดึงระเบิดที่เกิดขึ้นที่ปลาย เหล็กกล้าละมุนนี้เป็นเกรดเดียวกับที่ใช้ในงานคอนกรีตเสริมเหล็กซึ่งผลิตตามมาตรฐาน ASTM Specification A15 ,A16 และ A305

อุปกรณ์ยึดเหล็กกำลังสูง

ในระบบการก่อสร้างคอนกรีตอัดแรงแบบดึงก่อน เหล็กกำลังสูงจะถูกดึงระหว่างฐานอบัทเมนต์ โดยใช้อุปกรณ์แบบปล่อยตัวเร็ว อุปกรณ์นี้ใช้ยึดเหล็กโดยระบบลิ่มประกอบด้วยท่อสั้นซึ่งผิวด้านในมีลึกษณะเป็นกรวย มีแหวนยึดที่ปลายฟันเพื่อให้ตัวฟันเคลื่อนที่ไปพร้อม ๆ กันในท่อสั้น และมีสปริงสำหรับตัวฟันกับฝาเกลียวที่ปลาย

อุปกรณ์ยึดในระบบการก่อสร้างแบบดึงเหล็กหลังทำหน้าที่ยึดลวดกำลังสูง หรือลวดสะแตรนด์กับปลายคานคอนกรีตโดยตัวอุปกรณ์ยึดประกอบด้วยรูสำหรับลวดสะแตรนด์ผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีลักษณะเป็นแผ่นเหล็กหนา เพื่อต้านทานแรงปฏิกิริยาโดยการกด ลวดสะแตรนด์เมื่อถูกดึงจะยึดติดกับอุแกรณ์ยึด โดยใช้ระบบลิ่มเช่นเดียวกับกรณีการดึงเหล็กก่อน หรือโดยการใช้เครื่องย้ำปลายลวดให้เป็นหัวหมุดย้ำ ซึ่งจะยึดลวดติดกับอุปกรณ์ยึดโดยการกด ปัจจุบันมีบริษัทที่ผลิตอุปกรณ์ยึดหลายบริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การสูญเสียแรงอัด

การพัฒนาคอนกรีตอัดแรงในระยะเริ่มแรกไม่ประสบความสำเร็จมากนักเนื่องจากแรงดึงในการดึงเหล็กกล้าละมุนมีค่าไม่สูงมากนัก แรงดึงนี้จะสูญเสียไปเมื่อเวลาผ่านไประยะเวลาหนึ่ง การประยุกต์ใช้คอนกรีตอัดแรงเป็นที่แพร่หลาย เมื่อมีการใช้เหล็กกำลังสูงมาอัดแรง แรงดึงประสิทธิผลหักด้วนแรงสูญเสียมีค่ามากพอที่จะต้านทานหน่วยแรงที่กระทำ การออกแบบองค์อาคารคอนกรีตอัดแรงในภาวะน้ำหนักบรรทุกปกติจะต้องทำการวิเคราะห์การสูญเสียแรงอัดว่ามีค่าเป็นกี่เปอร์เซ็นต์ของแรงอัดแรกเริ่ม การสูญเสียแรงอัดแบ่งออกเป็นสองประเภท คือการสูญเสียแรงอัดทันทีทันใด และการสูญเสียแรงอัดตามเวลา

ในองค์อาคารคอนกรีตอัดแรงแบบดึงเหล็กที่หลังแรงดึงที่ปลายด้านแม่แรงเรียกว่าแรงอัดแม่แรง Pj ขณะที่ถ่ายแรงอัดคาน จะเกิดการสูญเสียแรงอัดเนื่องจากความเสียดทานเนื่องจากการเคลื่อนตัวของลิ่มในอุปกรณ์ยึด และเนื่องจากการหดตัวแบบอีลาสติก แรงอัดคานเมื่อหักการสูญเสียแรงอัดในขั้นตอนนี้เรียกว่า แรงอัดแรกเริ่ม เมื่อเวลาผ่านไปนาน ๆ แรงอัดคานจะลดลงเนื่องจากการหดตัวของคอนกรีต ซึ่งเป็นผลมาจากการคืบและการสูญเสียความชื้น นอกจากนี้การสูญเสียแรงอัดยังเกิดขึ้นเนื่องจากการล้าตัวของเหล็กกำลังสูงเอง อัตราการสูญเสียแรงอัดตามเวลานี้จะมีค่ามากในตอนแรก เมื่อเวลาผ่านไปการสูญเสียแรงอัดตามเวลาจะน้อยลง จนกระทั่งแรงอัดประสิทธิผลมีค่าคงที่ เมื่อเวลาผ่านไปหลาย ๆ ปี ในองค์อาคารคอนกรีตอัดแรงแบบดึงเหล็กก่อนจะไม่มีการสูญเสียแรงอัด เนื่องจากความเสียดทานและเนื่องจากการเคลื่อนตัวของลิ่มในอุปกรณ์ยึด เพราะการสูญเสียเนื่องจากสาเหตุทั้งสองเกิดขึ้นก่อนการตัดลวด การสูญเสียแรงอัดทันทีทันใดของคานแบบดึงเหล็กก่อนเกิดขึ้นเนื่องจากการหดตัวแบบอิลาสติก ซึ่งเกิดขึ้นขณะทำการตัดลวดเพื่อปล่อยแรง การสูญเสียแรงอัดตามเวลาเกิดขึ้นเนื่องจากสาเหตุเดียวกันกับคานแบบดึงเหล็กที่หลัง สูญเสียทั้งหมดทำได้โดยการคำนวนแรงสูญเสียเนื่องจากสาเหตุต่าง ๆ ทีละสาเหตุแล้วนำมารวมกันการวิเคราะห์ละเอียดนี้คอนข้างจะใช้เวลามากแต่เป็นสิ่งจำเป็นสำหรับโครงสร้างพิเศษ ในกรณีที่ องค์อาคารที่มีขนาดและช่วงยาวปกติผู้ออกแบบสามารถประมาณแรงสูญเสียทั้งหมดคร่าว ๆ ได้ถ้าทราบชนิดขององค์อาคารและวิธีดึงเหล็กมาตรฐาน AASHTO แนะนำค่าแรงสูญเสียทั้งหมดในตารางที่ 3-1 และข้อกำหนดของ Postensioning Institute ประมาณค่าแรงสูญเสียทั้งหมดสำหรับแผ่นพื้นและคานในตารางที่3-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3-1 การประมาณแรงสูญเสียตามมาตรฐาน AASHTO

ชนิดของเหล็กเสริม	แรงสูญเสียทั้งหมด	
	f_c' = 2706 N/mm2	f_c' = 34.5 N/mm2
ลวดตีเกลียวแบบดึงก่อน	221 N/mm2	310 N/mm2
ลวดตีเกลียวแบบดึงทีหลัง (ไม่รวมแรงเสียดทาน)		228 N/mm2
ท่อนเหล็ก	152 N/mm2	159 N/mm2

ตารางที่ 3-2 การประมาณแรงสูญเสียสำหรับงานดึงเหล็กทีหลัง (ไม่รวมแรงเสียดทาน)

ชนิดของเหล็กเสริมแบบดึงทีหลัง	แรงสูญเสียทั้งหมด	
	แผ่นพื้น	คานหรือคานย่อย
ลวดตีเกลียวเกรด 270 K	207 N/mm2	241 N/mm2
ลวดเกรด 240 K	138 N/mm2	172 N/mm2
ท่อนเหล็ก		

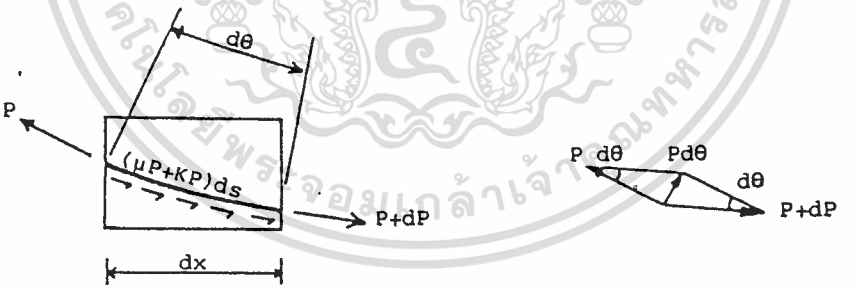
ค่าแรงสูญเสียตามตารางทั้งสองไม่รวมแรงสูญเสียเนื่องจากแรงเสียดทาน และใช้ได้ในกรณีที่ ค่ากำลังประลัยของคอนกรีต ระดับแรงอัดคานและสภาพบรรยากาศแวดล้อมอยู่ในเกณฑ์ปกติ นอกจากนี้ตารางที่ 3.1 ยังคงใช้ได้สำหรับคอนกรีตที่มีกำลังประลัยแตกต่างจากค่าในตารางเท่ากับ 3.5 N/mm2 ในหัวข้อต่อไปนี้จะกล่าวถึงวิธีคำนวนแรงสูญเสียเนื่องจากสาเหตุโดยละเอียด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงสูญเสียเนื่องจากแรงเสียดทาน

ในคานแบบดึงเหล็กทีหลัง แรงสูญเสียเนื่องจากแรงเสียดทานเกิดขึ้นจากสาเหตุสองประการ ประการแรกท่อคอนดุทที่เราใส่ไว้ในคานคอนกรีตจะมีแนวโค้ง ขณะดึงเหล็กผิวของเหล็กเสริมจะสัมผัสกับผนังด้านในของท่อคอนดุททำให้เกิดแรงเสียดทานตลอดความยาวของเหล็กเสริม แรงเสียดทานนี้เรียกว่าผลของความโค้ง นอกจากนี้แนวของท่อคอนดุทอาจเบี่ยงเบนไปจากที่ออกแบบไว้ทั้งทางแนวดิ่งและแนวระดับในระหว่างการก่อสร้าง ดังนั้นขณะดึงเหล็กจะมีแรงเสียดทานเกิดขึ้นเนื่องจากการคดของท่อ ทิศทางของแรงเสียดทานนี้จะมีทิศตรงข้ามกับการเคลื่อนที่ของเหล็กเสริม ทำให้แรงดึงที่บริเวณกลางคานน้อยกว่าแรงดึงที่ปลายด้านติดกับแม่แรง ในการออกแบบแนวโค้งผู้ออกแบบจะต้องทราบแรงอัดคานเมื่อหักด้วยแรงเสียดทานแล้ว ว่าจะเพียงพอที่จะต้านทานโมเมนต์เนื่องจากแรงภายนอกหรือไม่ หรือเพื่อประโยชน์ในการควบคุมหน่วยแรงดึงที่ปลายคานของเหล็กเสริม ถ้ามีการดึงเหล็กเกินพิกัดเผื่อแรงเสียดทาน

พิจารณาแรงที่เกิดขึ้นบนเหล็กเสริมในอิลิเมนต์คานในรูปที่ 3.1 ที่ปลายข้างซ้ายใกล้กับปลายคานมีแรงดึง P กระทำ และที่ระยะห่างออกไป ds แรงดึงเปลี่ยนแปลงไปมีค่าเป็น P + dP



รูปที่ 3.1 แรงดึงและแรงเสียดทานบนอิลิเมนต์ของคาน

และการเปลี่ยนแปลงของมุมเป็น$d\theta$ จากรูปสี่เหลี่ยมด้านขานของแรง แรงกดตั้งฉากที่กระทำบนคอนกรีตเท่ากับ P$d\theta$ ถ้า μ เป็นสัมประสิทธิ์ของแรงเสียดทานระหว่างเหล็กเสริมกับผิวท่อคอนดุท แรงเสียดทานที่เกิดขึ้นบนเหล็กเสริมเท่ากับμP$d\theta$ ค่าเฉลี่ยของแรงเสียดทานต่อหนึ่งหน่วยความยาว μP เท่ากับ μ P$d\theta$ /ds เนื่องจากการคดของท่อคอนดุท ถ้า K เป็นสัมประสิทธิ์ ของการคดซึ่งหมายถึงแรงเสียดทานต่อแรงดึงหนึ่งหน่วยบนความยาวของเหล็กเสริมหนึ่งหน่วยบนความยาวของเหล็กเสริม หนึ่งหน่วยดังนั้นแรงเสียดทานทั้งหมดที่เกิดขึ้นบนเหล็กเสริมซึ่งยาว ds เท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$dF = (\mu P + KP) ds \qquad \text{----------A}$$

ตารางที่ 3-3 แสดงค่า μ และ K ที่ใช้กันทั่ว ๆ ไป สำหรับท่อคอนดุทโลหะ พิจารณาสมดุลของแรงในแนวตามยาวของเหล็กเสริม จะได้

ตารางที่ 3-3 ค่าสัมประสิทธิ์ μ และ K

ชนิดของเหล็กเสริม	ช่วงพิกัดค่าสัมประสิทธิ์		ค่าสัมประสิทธิ์ที่ใช้ออกแบบ	
	K*	μ	K*	μ
ลวดกำลังสูง	0.0005-0.0030	0.15-0.35	0.0015	0.25
ท่อนเหล็กกำลังสูง	0.0001-0.0005	0.08-0.30	0.0003	0.20
ลวดตีเกลียวอาบรังสี	0.0005-0.0020	0.15-0.30	0.0015	0.25
*ต่อความยาว 1 ฟุต				

$$P + dP + (\mu p + KP)ds - P = 0 \qquad \text{----------B}$$

แทนค่า pds เท่ากับ $Pd\theta$ และหารทุกเทอมด้วย P จะได้

$$dP / P + Pd\theta + Kds = 0 \qquad \text{----------C}$$

ทำการอินทิเกรดทุกเทอมจากขีดจำกัดล่างถึงขีดจำกัดบน จะได้

$$\int_{P0}^{P} \frac{dP}{P} + \int_{0}^{\alpha} \mu d\theta + \int_{0}^{s} Kds = 0 \qquad \text{----------D}$$

$$ln(P_X / P_0) + \mu\alpha + Ks = 0$$
$$P_x = P_0 e^{-(\mu\alpha + Ks)} \approx P_0 e^{-(\mu\alpha + Kx)} \qquad \text{----------E}$$

สำหรับ $\mu\alpha + Ks$ ที่มีค่าน้อยกว่า 0.30 เราสามารถวิเคราะห์ค่า Px ในสมการที่ ----E ให้ง่ายขึ้นโดยการกระจายฟังค์ชั่นเป็นอนุกรม และใช้สองเทอมแรกของอนุกรมจะได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Px = Po (1 - (\mu\alpha + Kx)) \qquad \text{----------F}$$

โดยที่ Po = แรงดึงในเหล็กเสริมที่ปลายคานด้านติดกับแม่แรง

Px = แรงดึงในเหล็กเสริมที่หน้าตัดระยะ x นับจากปลายคานด้านติดกับแม่แรง

α = มุมระหว่างจุดตัดของเส้นสัมผัสที่ลากจากหน้าตัดที่ปลายคานและหน้าตัดที่ระยะ x

s = ความยาวตามผิวโค้งระหว่างหน้าตัดที่ปลายคานกับหน้าตัดที่ระยะ x

x = ความยาวในแนวราบระหว่างหน้าตัดที่ปลายคานและหน้าตัดที่ระยะ x

ในกรณีที่เหล็กเสริมเป็นเส้นโค้งแบบพาราโบราดังรูปที่ 3.4 เราจะหาค่ามุมระหว่างจุดตัดของเส้นสัมผัสที่ปลายคานทั้งสองข้างได้ดังนี้



รูปที่ 3.2 ฟังค์ชั่นของแนวเหล็กเสริมแบบพาราโบลา

จาก $\frac{e_c - e_x}{(\frac{l}{2} - x^2)^2} = \frac{e_c}{(\frac{l}{2})^2}$ ----------G

$$ln = \frac{4e_c}{l^2}(e_x - x^2)$$ ----------H

ความชันของเหล็กเสริมที่ระยะ x $\theta(x) = \frac{de_x}{dx}$

$= \frac{4e_c}{l^2}(1-2x)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมของเหล็กเสริมที่ปลาย , x = 0 ; $\theta(0) = \frac{4e_c}{l}$

$= 2\theta(0) = \frac{8e_c}{l}$

----------I

สำหรับคานต่อเนื่องช่วงยาวซึ่งมีแนวเหล็กโค้งตามรูปไดอะแกรมของโมเมนต์ดัด แรงสูญเสียเนื่องจากแรงเสียดทานอาจมีค่ามาก แรงดึงประสิทธิ์ผลที่หน้าตัดภายในคานวัดจากปลายด้านแม่แรงจะมีค่าน้อยกว่าหน้าตัดที่ปลายคานมาก ทำให้หน่วยแรงมากกว่าหน่วยแรงปลอดภัยในกรณีเราอาจจะดึงเหล็กเสริมที่ปลายด้านแม่แรงเพิ่มขึ้นเกินพิกัดเพื่อหักล้างแรงสูญเสียบางส่วนหรือทั้งหมด มาตรฐาน ACI ยอมให้หน่วยแรงดึงเกินพิกัดไม่สูงกว่า 0.8fpu หรือ 0.94fpy แล้วแต่ว่าค่าไหนจะน้อยกว่ากัน การดึงเหล็กเกินพิกัดอาจจะกระทำเพื่อหักล้างแรงสูญเสียเนื่องจากการเคลื่อนตัวในอุปกรณ์ยึดประไยชน์อีกประการหนึ่งคือผู้ออกแบบอาจพิจารณาดึงเหล็กที่ปลายคานทั้งสองข้าง ซึ่งจะลดแรงเสียดทานลงครึ่งหนึ่ง หรืออาจจะออกแบบแนวเหล็กเสริมบริเวณส่วนรับแรงดึงให้มีความโค้งน้อยลง โดยออกแบบเหล็กเสริมในบริเวณส่วนที่รับแรงอัดประกอบกัน

แรงสูญเสียเนื่องจากการเคลื่อนตัวในอุปกรณ์ยึด

ในคานแบบดึงเหล็กทีหลัง การถ่ายแรงอัดคานสู่คอนกรีตกระทำโดยผ่านอุปกรณ์ยึดซึ่งฝังไว้ที่ปลายคาน ภายใต้แรงอัดอุปกรณ์ยึดจะเกิดการเสียรูปซึ่งเรียกว่าการเซ็ตตัวและเป็นสาเหตุให้เกิดการสูญเสียในคานการเสียรูปมีค่ามากน้อยขึ้นอยู่กับชนิดของอุปกรณ์ยึด ในอุปกรณ์ยึดที่มีรูปกรวย ตัวลิ่มผ่าซีกจะเคลื่อนที่ประมาณ 3 มม. ก่อนที่ฟันลิ่มจะยึดเหล็กแน่น สำหรับอุปกรณ์ยึดชนิดแผ่นกด การเสียรูปมีค่าประมาณ 0.75 มม.

ค่าแรงสูญเสียคำนวนได้เมื่อทราบค่าเซ็ทตัว , Δ_a ของอุปกรณ์ยึด ถ้าเหล็กเสริมยาว l ความเครียดการหดตัวเท่ากับ $\Delta\varepsilon_s = \Delta_a / l$ ดังนั้นหน่วยแรงสูญเสียเท่ากับ

$$ANC = \Delta f_s = E_S \Delta_a / l$$ ----------J

ดังนั้นเราจะเห็นว่าแรงสูญเสียจะมีค่ามากเมื่อคานมีความยาวน้อยดังนั้นผู้ออกแบบอาจพิจารณาดึงเหล็กเกินพิกัดในกรณีเดียวกับในกรณีแรงเสียดทาน

ในคานแบบดึงเหล็กก่อน การเซ็ทตัวของอุปกรณ์ยึดจะเกิดขึ้นก่อนการปล่อยแรงจึงไม่มีการสูญเสียแรงอัดเนื่องจากสาเหตุนี้ อย่างไรก็ตามการดึงเหล็กติดกับฐานยึดในโรงงานมักจะดึงเผื่อระยะเซ็ทตัวที่เกิดขึ้น

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงสูญเสียเนื่องจากการหดตัวแบบอิลาสติค

ในคานแบบดึงเหล็กก่อนขณะตัดลวดเหล็กเพื่อถ่ายแรงอัดสู่คาน คานจะหดตัวทันทีภายใต้แรงอัดคานและน้ำหนักของคานเอง เนื่องจากคานหดตัวเหล็กเสริมจะหดตัวตามไปด้วย โดยมีการเปลี่ยนแปลงความเครียดของเหล็กเท่ากับการเปลี่ยนแปลงความเครียดของคอนกรีตที่ระดับจุดศูนย์กลางถ่วงของเหล็กเสริมนั่นคือ

$$\Delta\varepsilon_s = \Delta\varepsilon_{cs} = f_{cs} / E_c$$

$$ES = E_S \Delta\varepsilon_S = \frac{E_S}{E_{ci}} f_{cs} = nf_{cs} \qquad \text{----------K,L}$$

โดยที่ ES = หน่วยแรงสูญเสียเนื่องจากการหดตัวแบบอิลาสติค

$$f_{cs} = \frac{P_0}{A} + \frac{P_0 e}{I} - \frac{M_g e}{I}$$

E_{ci} = โมดูลัสความยืดหยุ่นของคอนกรีตขณะปล่อยแรง ซึ่งคอนกรีตมีกำลังสูงสุดเท่ากับ f_{ci}'

ค่า Po เป็นแรงอัดแรกเริ่มซึ่งหักแรงสูญเสียเนื่องจากการหดตัวแบบอิลาสติกในการวิเคราะห์ต้องสมมุติแรงสูญเสียนี้ก่อนซึ่งโดยปกติจะให้แรงอัดเป็น Po = 0.9Pi

การวิเคราะห์แรงสูญเสียเนื่องจากการหดตัวแบบอิลาสติคตามปกติจะคำนวนที่หน้าตัดซึ่งเกอดโมเมนต์ดัดมากที่สุด

การสูญเสียแรงอัดเนื่องจากการหดตัวแบบอิลาสติคสำหรับ tendon แต่ละเส้นสามารถเขียนเป็นสูตรตามข้อแนะนำของ ACI - ASCE ได้ดังนี้

$$ES = K_{ES} \frac{E_S}{E_{ci}} f_{cs} \qquad \text{----------M}$$

โดยที่

ES = เป็นหน่วยแรงสูญเสียเนื่องจากการหดตัวแบบอิลาสติค (กก/ซม.2)

Kes = เท่ากับ 1.0 สำหรับคานแบบดึงเหล็กก่อน

Kes = เท่ากับ 0.5 สำหรับคานแบบดึงเหล็กทีหลัง

แรงสูญเสียซึ่งเป็นฟังค์ชั่นของเวลา

แรงสูญเสียซึ่งเป็นฟังค์ชั่นของเวลาประกอบด้วยแรงสูญเสียเนื่องจากการคืบและการหดตัวของคอนกรีตและเนื่องจากการล้าตัวของเหล็กเสริม แรงสูญเสียเนื่องจากสาเหตุทั้งสามนี้มีความสัมพันธ์ซึ่งกันและกัน และอัตราการเกิดแรงสูญเสียจะสูงในช่วงเวลาแรก และลดน้อยลงเมื่อเวลาผ่านไปนาน ๆ หลาย ๆ ปี แรงสูญเสียตามเวลานี้มีค่าประมาณ 15-20 เปอร์เซ็นต์ ของแรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัดแรกเริ่มขณะปล่อยแรงแรงอัดแรกเริ่มหักด้วยแรงสูญเสียเป็นแรงอัดประสิทธิผลซึ่งก่อให้เกิดหน่วยแรงต้านหน่วยแรงเนื่องจากน้ำหนักบรรทุกขณะใช้งาน

วิธีวิเคราะห์แรงสูญเสียตามเวลามีหลายวิธีแต่วิธีที่สะดวกและง่ายต่อการคำนวณคือวิธีของ ACI - ASCE ซึ่งประมาณแรงสูญเสียเนื่องจากสาเหตุทีละสาเหตุโดยไม่ต้องแบ่งช่วงเวลาของการเกิดการเสียรูปในการคำนวณซึ่งมีรายละเอียดดังต่อไปนี้

แรงสูญเสียเนื่องจากการคืบ

การคืบคือการเสียรูปของคอนกรีตเมื่อมีแรงกระทำถาวรบนคอนกรีต ถ้า ε_{ci} คือความเครียดที่เกิดขึ้นทันทีขณะที่แรงกระทำและ ε_{cu} คือความเครียดทั้งหมดเนื่องจากการคืบสัมประสิทธิ์ความเครียด $c_c = \varepsilon_{cu} / \varepsilon_{ci}$ มีค่าประมาณ 3 ถึง 5 ในคานคอนกรีตอัดแรงผลของการคืบมีค่าน้อยลง เพราะความเครียดขณะปล่อยแรง จะมีค่าลดลงเนื่องจากความเครียดคืนเมื่อแรงถาวรกระทำ ACI -ASCE แนะนำสูตรการคำนวนแรงสูญเสียเนื่องจากการคืบดังนี้

$$CR = K_{cr} \frac{E_s}{E_s} (f_{cs} - f_{cds}) \qquad \text{----------N}$$

โดยที่ CR = หน่วยแรงสูญเสียเนื่องจากการคืบ

K_{cr} = 2.0 สำหรับคานแบบดึงเหล็กก่อน

K_{cr} = 1.6 สำหรับคานแบบดึงเหล็กทีหลัง

fcds = หน่วยแรงในคอนกรีตที่ระดับจุดศูนย์ถ่วงของเหล็กเสริมเนื่องจากแรงถาวรทั้งหมดซึ่งกระทำภายหลังขณะปล่อยแรง

Ec = โมดูลัสความยืดหยุ่นของคอนกรีตที่อายุ 28 วัน

สำหรับคอนกรีตทรายมวลเบาใช้ค่า K_{cr} ลดลง 20 % และใช้สมการ --N ได้ในกรณีที่คานมีการยึดเกาะ (bonded)

แรงสูญเสียเนื่องจากการหดตัว

ความเครียดการหดตัวเป็นผลเนื่องจากการสูญเสียความชื้น มีความสัมพันธ์กับองค์ประกอบหลายประการ แต่ที่สำคัญคืออัตราส่วนปริมาตร ต่อพื้นที่ผิวขององค์อาคาร และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความชื้นสัมพัทธ์นอกจากนี้สำหรับคานคอนกรีตอัดแรงระยะเวลาตั้งแต่หลังบ่มคอนกรีตถึงปล่อยแรงมีผลต่อขนาดของแรงสูญเสีย ค่าเฉลี่ยของความเครียดการหดตัวทั้งหมดเป็น 5.50×10^{-4} ซึ่งประมาณ 80 % ของความเครียดทั้งหมดเกิดในปีแรกของอายุคอนกรีต เนื่องจากอัตราส่วนปริมาตรต่อพื้นที่ผิว , V/S และความชื้นสัมพัทธ์ของบรรยากาศ H ค่าความเครียดการหดตัวประสิทธิผลจะได้

$$\varepsilon_{sh} = 5.5 \times 10^{-4} \alpha(1 - 0.024V/S) \text{----------O}$$

โดยที่ $\alpha = 1.5 - 0.015H$

หรือ $$\varepsilon_{sh} = 8.2 \times 10^{-6}(1 - 0.024V/S)(100 - H) \text{-----P}$$

ACI - ASCE แนะนำสูตรการคำนวนแรงสูญเสียเนื่องจากการหดตัวดังนี้

$$SH = 8.2X10^{-6} K_{sh} E_s (1 - 0.024V/S)(100 - H) \text{----------Q}$$

โดยที่ K_{sh} เป็นค่าเปอร์เซ็นต์ของความเครียดการหดตัวประสิทธิผล ขึ้นอยู่กับจำนวนวันซึ่งนับจากเวลาที่เสร็จการบ่มจนถึงเวลาปล่อย ดังแสดงในตารางที่ และใช้สำหรับคานแบบดึงเหล็กทีหลังในกรณีคานแบบดึงเหล็กก่อนซึ่งการตัดลวดมักกระทำเมื่ออายุคอนกรีตน้อย ๆ ดังนั้นจะใช้ K_{sh} = 1.00

ตารางที่ 3-4 ค่าแฟกเตอร์ K_{sh}

ระยะเวลาหลังการบ่มถึงขณะปล่อยแรง (วัน)	1	3	5	7	10	20	30
Ksh	0.92	0.85	0.80	0.77	0.73	0.64	0.58

แรงสูญเสียเนื่องจากการล้าตัวของเหล็กเสริม

เมื่อเราทำการดึงเหล็กกำลังสูงภายใต้ความเครียดค่าหนึ่งเป็นระยะเวลานาน แรงดึงในเส้นลวดจะลดลง ปรากฏการณ์นี้เรียกว่า การล้าตัว ขนาดของแรงดึงที่ลดลงนี้ขึ้นอยู่กับระดับของหน่วยแรงดึงแรกเริ่มและระยะเวลาในการดึงลวดกราฟค่าอัตราส่วนหน่วยแรงดึงแรกเริ่มเป็นฟังก์ชั่นของเวลาแสดงในรูปที่ 3.4 สำหรับลวดกำลังสูงและลวดตีเกลียวแบบ stress - relieved ฟังก์ชั่นของอัตราส่วน fse / fsi กับเวลาเป็น

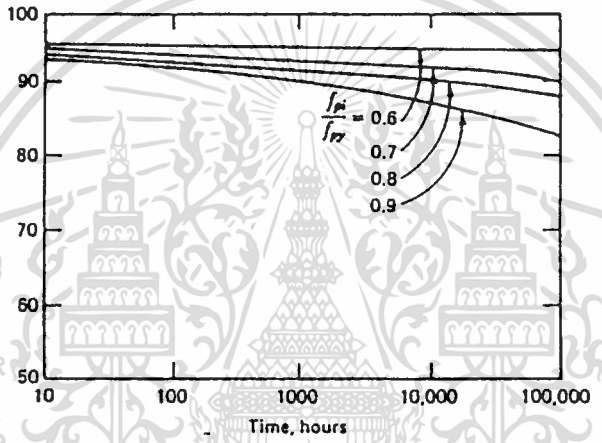
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{f_{se}}{f_{si}} = 1 - \frac{\log t}{10}\left(\frac{f_{si}}{f_{sy}} - 0.55\right) \quad \text{----------R}$$

ฟังค์ชั่นนี้ได้มาจากผลของการทดลองโดย Magura กับคณะ ในสมการที่ ----R

t = เวลาเป็นชั่วโมงนับจากเวลาที่ดึงเหล็กเสร็จ

และ $\frac{f_{si}}{f_{sy}}$ มีค่ามากกว่า 0.55



รูปที่ 3.3 การล้าตัวในลวดกำลังสูงและลวดตีเกลียว

จากรูปที่ 3.7 แรงสูญเสียเมื่อหน่วยแรงดึงแรกเริ่มมีค่าสูงเมื่อเทียบกับ f_{sy} จะมีค่ามากสำหรับลวดชนิด stress - relieved ในกรณีที่ ต้องการลดแรงสูญเสียเนื่องจากการล้าตัว ผู้ออกแบบอาจจะเลือกใช้ลวดชนิด low relaxation ซึ่งจะเกิดแรงสูญเสียน้อยกว่าสำหรับหน่วยแรงดึงเท่ากันอย่างไรก็ตามลวดชนิดหลังก็มีราคาสูงกว่า

ในคอนกรีตอัดแรง แรงสูญเสียเนื่องจากการล้าตัวจะน้อยกว่าในการทดสอบ ทั้งนี้เป็นเพราะความเครียดในเหล็กเสริมเปลี่ยนแปลงไปตามเวลาเนื่องจากการคืบและการหดตัว ACI - ASCE แนะนำสูตรการคำนวนแรงสูญเสียดังนี้

$$RE = \{K_{re} - J(SH + CR + ES)\}C \quad \text{----------S}$$

โดยค่าแฟกเตอร์ K_{re}, J *and* C แสดงในตารางที่3.5และ3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3-5 ค่าแฟกเตอร์ Kre และ J

Type of tendon[a]	K_{re} (N/mm^2)	J
270 Grade stress-relieved strand or wire	137.9	0.15
250 Grade stress-relieved strand or wire	127.7	0.14
240 or 235 Grade stress-relieved wire	121.4	0.13
270 Grade low-relaxation strand	34.5	0.040
250 Grade low-relaxation wire	31.9	0.037
240 or 235 Grade low-relaxation wire	30.3	0.035
145 or 160 Grade stress-relieved bar	41.4	0.05

[a] In accordance with ASTM A416-74, ASTM A421-76, or ASTM A722-75.

ตารางที่ 3-6 ค่าแฟกเตอร์ C

f_{pi}/f_{pu}	Stress relieved strand or wire	Stress-relieved bar or low-relaxation strand or wire
0.80		1.28
0.79		1.22
0.78		1.16
0.77		1.11
0.76		1.05
0.75	1.45	1.00
0.74	1.36	0.95
0.73	1.27	0.90
0.72	1.18	0.85
0.71	1.09	0.80
0.70	1.00	0.75
0.69	0.94	0.70
0.68	0.89	0.66
0.67	0.83	0.61
0.66	0.78	0.57
0.65	0.73	0.53
0.64	0.68	0.49
0.63	0.63	0.45
0.62	0.58	0.41
0.61	0.53	0.37
0.60	0.49	0.33

จากสมการที่ 3.18 จะเห็นว่าเทอมที่สองมีค่าเป็นลบ ทั้งนี้เป็นเพราะว่าแรงสูญเสียเนื่องจากแบบอิลาสติก การคืบและการหดตัวทำให้แรงดึงในเส้นลวดลดลง ทำให้แรงสูญเสียเนื่องจาก

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การล้าตัวลดลงด้วยค่า f_{si} ในตาราง เป็นแรงดึงที่แม่แรง f_{sj} หักด้วยหน่วยแรงสูญเสียเนื่องจากแรงเสียดทานและการเคลื่อนตัวของอุปกรณ์ยึด

การเปรียบเทียบแรงสูญเสียทั้งหมด

แรงสูญเสียทันทีทันใดเนื่องจากการหดตัวแบบอิลาสติคและแรงสูญเสียตามเวลา เนื่องจากสาเหตุต่าง ๆ มีค่าไม่เท่ากันในคานแบบดึงเหล็กก่อนและแบบดึงเหล็กทีหลังเพื่อเป็นการเปรียบเทียบขนาดของแรงสูญเสียในคานทั้งสองชนิดที่มีคุณสมบัติของคอนกรีตและเหล็กเสริม และการบ่มในอากาศอยู่ในเกณฑ์เฉลี่ย LIN แนะนำค่าเฉลี่ยของแรงสูญเสียดังนี้

	คานแบบดึงเหล็กก่อน	คานแบบดึงเหล็กทีหลัง
ES	3%	1%
CR	6%	5%
SH	7%	6%
RE	8%	8%
รวม	25%	20%

ตารางเปรียบเทียบนี้ยังไม่รวมแรงสูญเสียเนื่องจากแรงเสียดทานและการเซ็ตตัวของอุปกรณ์ยึดในการพิจารณาค่าแรงสูญเสียทั้งหมด ACI - ASCE แนะนำค่าสูงสุดของหน่วยแรงสูญเสียสำหรับเหล็กเสริมชนิดลวดดีเกลียวดังในตาราง3.7

ชนิดของลวดดีเกลียว	หน่วยแรงสูญเสีย (นิวตัน/มม.^{2})	
	คอนกรีตน้ำหนักปกติ	คอนกรีตน้ำหนักเบา
ชนิด stress - relieved	345	380
ชนิด low - relaxation	276	311

ตารางที่ 3-7 ค่าหน่วยแรงสูญเสียสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีวิเคราะห์แรงสูญเสียในคานที่สะดวกอีกวิธีหนึ่งของ PCI ซึ่งมีเหตุผลต่าง ๆ ในการวิเคราะห์แบบประมาณเช่นเดียวกัน

การหาแรงสูญเสียโดยการประมาณ(Lump Sum Estimate For Prestress loss)

แบบแผนสำหรับการออกแบบโครงสร้างคอนกรีตอัดแรงถูกพัฒนาโดย ACI - ASCE Committee 423 ได้รวมถึงวิธีการประมาณค่าการสูญเสียแบบ Lumpsum ไว้ด้วยได้แนะนำถึงการสูญเสียทั้งหมดอันเนื่องมาจากการหดตัว ,การคืบ ,การสูญเสียความชื้น ,การล้า(แต่ไม่รวมถึงการสูญเสียเนื่องจากการเกิดแรงเสียดทานและการเคลื่อนตัวของห้วยึด)ในคอนกรีตธรรมดา 240 Mpa สำหรับคานคอนกรีตอัดแรงก่อนและ 170 Mpa สำหรับคอนกรีตอัดแรงทีหลัง ค่าเหล่านี้ได้รับการยอมรับอย่างกว้างขวางและได้บรรจุเป็นมาตรฐานในปี 1963 (ACI) และมาตรฐาน AASHTO สำหรับสะพานและทางด่วนที่ออกแบบโดยคอนกรีตอัดแรงวิธีนี้ออกแบบไว้ตั้งแต่ปี 1975 ปัจจุบันยังใช้งานอยู่โดยไม่มีปัญหา ใช้การประมาณค่าแบบLumpsum แทบทั้งสิ้น แต่มาตรฐาน ACI 1977 ที่ใช้อ้างอิงนั้นค่าเหล่านี้ปัจจุบันไม่สามารถใช้ครอบคลุมทุกสถานการณ์

ทั้งสองมาตรฐานได้ถูกทดแทนด้วยค่าที่ได้มีการปรับปรุงให้ใช้ในกรณีที่มีปัญหาเดิมได้ ในตารางแสดงค่าที่ปรับปรุงโดย AASHTO สำหรับคอนกรีตอัดแรง การล้านอกเหนือที่ประมาณในปี 1938 และได้เปลี่ยนแปลงค่าให้เหมาะสมยิ่งขึ้น ค่า f_c' ที่ใช้มาตรฐาน AASHTO ใช้ 3.5 Mpa เปลี่ยนแปลงไปอาจจะมากขึ้นหรือน้อยลงกว่าที่ให้มา PTI ได้ให้มาตรฐานไว้ในตารางที่ ในคู่มือPostensioning ค่าที่ปรับปรุงใช้สำหรับอุตสาหกรรม และสำหรับโครงการที่ค่าการสูญเสียไม่ถูกควบคุมไว้

ในกรณีที่ค่าความเค้นมีค่าต่ำมากหรือการให้แรงอัดสูงเกินไป แห้ง หรือเปียกเกินไปก็ให้มีการปรับแปลงค่าไปได้จากค่าเหล่านี้ ค่าการสูญเสียเนื่องจากความเสียดทานไม่ได้รวมไว้ในตารางทั้งสอง

การใช้งานLumpsum loss ที่ใช้ในกรณีที่เป็นสภาวะเฉลี่ยซึ่งรายละเอียดเกี่ยวกับที่มาของการประมาณไม่จำเป็นต้องทราบ แต่สำหรับการออกแบบที่ต้องการความถูกต้องแม่นยำอาจไม่เหมาะสมในบางสถานการณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3-8 AASSHTO Lumpsum Loss

Type of Prestressing Steel	Total Loss	
	f_c' = 28 Mpa	f_c' = 35 Mpa
Pretensioning Strand		310 Mpa
Posttensioning		
wire or strand	220 Mpa	230 Mpa
Bars	150 Mpa	160 Mpa

loss due to friction loss exclude Friction loss should be complete accordingly to Section 6-5

ตารางที่ 3-9 Approximate Prestress loss Value for Posttensioning

Posttensioning	Prestress Loss psi	
Tendon Materials	Slabs	Beam and Joint
Stress relieved 270 Strand and stress relieved 240 wire	210 Mpa	240 Mpa
Bars	140 Mpa	170 Mpa

Loss due to friction include Average values of concrete strength, Prestress level and exposure condition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การวิเคราะห์และการออกแบบคาน

การวิเคราะห์หน่วยแรงเนื่องจากการดัด

สำหรับการวิเคราะห์หมายความถึงการคำนวณหน่วยแรงที่เกิดขึ้นเนื่องแรงภายในที่กระทำต่อหน้าตัดคานเนื่องจากน้ำหนักบรรทุกภายนอกและแรงอัดคาน โดยสมมุติว่าทราบขนาดต่าง ๆ ของหน้าตัดคานคอนกรีต พื้นที่และตำแหน่งของเหล็กกำลังสูงหน่วยแรงที่เกิดขึ้นนี้จะมีการตรวจสอบกับหน่วยแรงปลอดภัยที่กำหนดให้ใช้ตามข้อกำหนดต่าง ๆ ในทางตรงกันข้ามสำหรับการออกแบบหน้าตัดคานผู้ออกแบบทราบแรงกระทำภายนอก คุณสมบัติของวัสดุ และหน่วยแรงปลอดภัย ดังนั้นเราจำเป็นต้องกำหนดหน้าตัดคานและพื้นที่เหล็กเสริมจากประสบการณ์จากนั้นจึงทำการคำนวณตรวจสอบหน่วยแรงที่เกิดขึ้น ถ้าหน่วยแรงมากกว่าหน่วยแรงปลอดภัยจะต้องออกแบบหน้าตัดใหม่และตรวจสอบหน่วยแรงจนกว่าจะปลอดภัยจะเห็นว่าการวิเคราะห์เป็นส่วนหนึ่งของการออกแบบ ในที่นี้จะกล่าวถึงรายระเอียดการวิเคราะห์เนื่องจากการดัดต่อไป

หน่วยแรงในแนวแกนที่เกิดขึ้นบนหน้าตัดมีสองชนิดคือหน่วยแรงดึงและหน่วยแรงอัดในการออกแบบคอนกรีตอัดแรงเนื่องจากคอนกรีตมีความสามารถในการรับแรงอัดได้ดี แต่ด้านทานแรงดึงได้ไม่มากนัก ดังนั้นผู้ออกแบบจะต้องระมัดระวังไม่ให้หน่วยแรงดึงมากเกินไป การกำหนดเครื่องหมายกำกับหน่วยแรงจึงกำหนดหน่วยแรงอัดมีเครื่องหมานเป็นบวกและหน่วยแรงดึงมีเครื่องหมายเป็นลบ

การวิเคราะห์หน่วยแรงบนหน้าตัดหมายถึงหน่วยแรงที่เกิดขึ้นเนื่องจากน้ำหนักบรรทุกภายนอกและแรงอัดคานที่ระดับปกติ คานคอนกรีตอัดแรงมีคุณสมบัติเป็นคานอิลาสติคในช่วงน้ำหนักบรรทุกปกติการแจกแจงหน่วยแรงเนื่องจากแรงภายในต่าง ๆ ที่กระทำบนหน้าตัดเป็นเส้นตรงดังนั้นการวิเคราะห์หน่วยแรงลัพธ์จึงใช้หลักการซุปเปอร์โพซิชัน

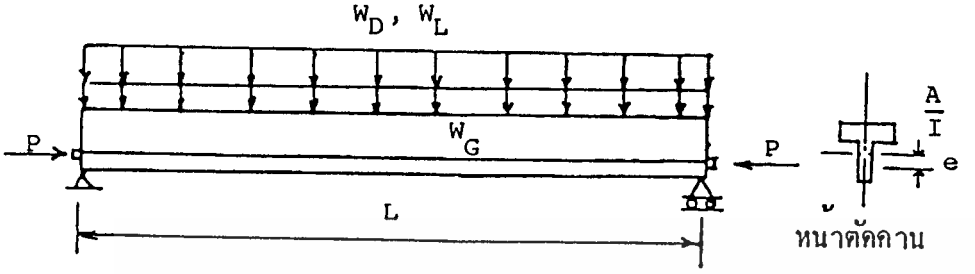
หน่วยแรงบนหน้าตัดคาน

การวิเคราะห์หน่วยแรงในคานจะต้องพิจารณาผลของแรงภายในต่าง ๆ ที่กระทำต่อหน้าตัด โดยพิจารณาถึงลำดับขั้นของแรงกระทำ ในทางปฏิบัติหน่วยแรงที่เกิดขึ้นจะมีค่าวิกฤติที่จุดรองรับและที่หน้าตัดกึ่งกลางคาน ซึ่งต้องพิจารณาที่สองช่วงเวลา คือขั้นตอนขณะปล่อยแรงและขั้นตอนแรงกระทำปกติ

พิจารณาคานช่วงเดี่ยวธรรมดาเป็นคานที่มีหน้าตัดคงที่สม่ำเสมอหน้าตัดคานมีพื้นที่เท่ากับ A และ I เป็นโมเมนต์ของความเฉื่อยรอบแกนหลักซึ่งผ่านจุดเซ็นทรอยด์ หน้าตัดคานเป็นรูปสมมาตรรอบแกนรองซึ่งอยู่ในระนาบดิ่ง ขณะปล่อยแรงอัดคานมีแรงอัดคานแรกเริ่มเท่ากับ P ซึ่งมีค่าคงที่ตลอดความยาวคานในกรณีที่เหล็กไม่ยึดกับคอนกรีตแนวของเหล็กกำลังสูงมีระยะเยื้องศูนย์ e เทียบกับแกนหลักของหน้าตัดแรงบรรทุกคานประกอบด้วยน้ำหนักคาน w_g ,

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้ำหนักตายอื่น ๆ wd,และน้ำหนักบรรทุกจร wl แรงเหล่านี้เป็นแรงกระทำปกติและสมมุติให้ wd และ wl มีค่าคงที่สม่ำเสมอเช่นเดียวกับ wg



รูปที่ 4.1 คานคอนกรีตอัดแรงช่วงเดียว

ขั้นตอนขณะปล่อยแรง

ในช่วงเวลาขณะปล่อยแรงคานมีแรงอัดคานแรกเริ่ม P กระทำพร้อมกับโมเมนต์ดัดเนื่องจากแรง wg ซึ่งมีการแจกแจงเป็นรูปพาราโบลามีค่าเท่ากับศูนย์และเท่ากับ Mg ที่จุดรองรับและที่หน้าตัดกึ่งกลางคานตามลำดับ เนื่องจากแรง P โดยการย้ายแกนจะได้แรงอัดในแนวศูนย์กระทำพร้อมกับโมเมนต์ดัด Pe รอบแกนหลักของหน้าตัดหน่วยแรงบนหน้าตัดใด ๆ ของคานเท่ากับ

$$ft = \frac{P}{A} - \frac{Pec1}{I}; \quad fb = \frac{P}{A} + \frac{Pec2}{I}$$

โดยที่ ft และ fb เป็นหน่วยแรงที่ผิวบนและผิวล่างของหน้าตัดตามลำดับ , c1 เป็นระยะจากแกนหลักถึงผิวบนและ c2 เป็นระยะจากแกนหลักถึงผิวล่างของหน้าตัด

ผลของโมเมนต์ดัด Mg ที่หน้าตัดกึ่งกลางคานทำให้แรงที่ผิวบนและผิวล่างของหน้าตัดเป็น $\Delta ft = +\frac{Mgc1}{I}$ และ $\Delta fb = -\frac{Mgc2}{I}$ ดังนั้นโดยการซุปเปอร์โพซิชันหน่วยแรงบนหน้าตัดที่กึ่งกลางคานจะเท่ากับ

$$ft = \frac{P}{A} - \frac{Pec1}{I} + \frac{Mgc1}{I}$$

$$fb = \frac{P}{A} + \frac{Pec2}{I} - \frac{Mgc2}{I}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และหน่วยแรงบนหน้าตัดที่จุรองรับคานจะเท่ากับ

$$ft = \frac{P}{A} - \frac{Pec1}{I}$$

$$fb = \frac{P}{A} + \frac{Pec2}{I}$$

ตามปกติในขั้นตอนขณะปล่อยแรง ผิวบนของหน้าตัดมีแนวโน้มจะเกิดหน่วยแรงดึงและที่ผิวล่างของหน้าตัดจะเกิดหน่วยแรงอัด ในช่วงเวลาปล่อยแรงหน่วยแรงในคานจะมีค่าวิกฤติช่วงเวลาหนึ่งเนื่องจากแรงอัดคานมีค่าสูงกว่าขณะใด ๆ ตลอดอายุของคานซึ่งเกิดขึ้นขณะที่โมเมนต์ดัดที่กระทำมีเพียง Mg จึงมีค่าไม่มากนัก

ขั้นตอนแรงกระทำปกติ

สำหรับขั้นตอนแรงกระทำปกติซึ่งอาจจะเป็นระยะเวลานานนับจากช่วงเวลาขณะปล่อยแรง จึงสมมุติว่าการสูญเสียแรงอัดเนื่องจากการคืบการหดตัวของคอนกรีตและการล้าของเหล็กได้เกิดขึ้นเสร็จแล้วแรงอัดคานประสิทธิผลในคานมีค่าเท่ากับแฟกเตอร์ประสิทธิผลคูณด้วยค่าแรงอัดแรกเริ่ม นอกจากนี้โมเมนต์ดัดที่กระทำต่อคานทั้งหมดเกิดเนื่องจากน้ำหนักคานและน้ำหนักบรรทุกอื่น ๆ ถ้า Md และ Ml เป็นโมเมนต์ดัดที่หน้าตัดกึ่งกลางคานเนื่องจาก wd และ wl ดังนั้นหน่วยแรงที่ผิวบนและผิวล่างของหน้าตัดเป็น $\Delta ft = +(Md + Ml)/I$ และ $\Delta fb = -(Md + Ml)/I$ หน่วยแรงบนหน้าตัดที่กึ่งกลางคานเท่ากับ

$$ft = \frac{\eta P}{A} - \frac{\eta Pec1}{I} + \frac{Mgc1}{I} + \frac{(Md + Ml)c1}{L}$$

$$fb = \frac{\eta P}{A} + \frac{\eta Pec2}{I} - \frac{Mgc2}{I} - \frac{(Md + Ml)c2}{I}$$

และหน่วยแรงบนหน้าตัดที่จุดรองรับคานจะเท่ากับ

$$ft = \frac{\eta P}{A} - \frac{\eta Pc1}{I}$$

$$fb = \frac{\eta P}{A} + \frac{\eta Pec2}{I}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนแรงกระทำปกติผิวบนของหน้าตัดมีแนวโน้มที่จะเกิดแรงอัด และผิวล่างของหน้าตัดจะเกิดหน่วยแรงดึง ทั้งนี้เนื่องจากแรงเนื่องจากโมเมนต์ดัดทั้งหมดมีค่ามากกว่าหน่วยแรงเนื่องจากแรงอัดคานประสิทธิผล

ในการวิเคราะห์หน่วยแรงบนหน้าตัดขณะปล่อยแรงและขณะน้ำหนักบรรทุกปกติกระทำคานอยู่ในช่วงอิลาสติค คานคอนกรีตอัดแรงยังไม่เกิดการแตกร้าวดังนั้นทั้งคอนกรีตและเหล็กจะต้านทานหน่วยแรงได้ การวิเคราะห์หน่วยแรงบนหน้าตัดให้ถูกต้องตามทฤษฎีจริง ๆ จะต้องคำนึงถึงพื้นที่หน้าตัดของเหล็ก และ พื้นที่ของคอนกรีตส่วนที่ถูกเหล็กเสริมแทนที่วิธีคำนวณจะทำได้สะดวกโดยใช้หลักการเปลี่ยนพื้นที่เหล็กให้เป็นพื้นที่คอนกรีตโดยคูณด้วยอัตราส่วนโมดูลัสของความยืดหยุ่นของเหล็กต่อคอนกรีต และคำนวณคุณสมบัติของพื้นที่หน้าตัดรวมอย่างไรก็ตามหน่วยแรงที่คำนวณได้ก็จะแตกต่างน้อยมากจากการคำนวณโดยใช้คุณสมบัติของหน้าตัดคอนกรีตในคานคอนกรีตอัดแรงแบบดึงเหล็กก่อน ทั้งนี้เป็นเพราะพื้นที่เหล็กเสริมกำลังสูงในหน้าตัดมีค่าน้อยเมื่อเทียบกับพื้นที่คอนกรีต ในคานแบบดึงเหล็กทีหลังก่อนที่จะทำการอัดมอร์ต้า พื้นที่ท่อคอนดุทอาจจะมีขนาดใหญ่ กรณีผู้ออกแบบอาจจะต้องคำนวณคุณสมบัติของหน้าตัดโดยการหักพื้นที่ท่อคอนดุท

หน่วยแรงปลอดภัยในคอนกรีตและเหล็กเสริม

ในการออกแบบคานคอนกรีตอัดแรง หน่วยแรงในคอนกรีตและเหล็กเสริมขณะถ่ายแรงและขณะน้ำหนักบรรทุกปกติกระทำจะต้องไม่มากกว่าหน่วยแรงปลอดภัย ซึ่งกำหนดโดยข้อบัญญัติของ ACI สำหรับโครงสร้างอาคารหรือข้อบัญญัติของ AASHTO สำหรับโครงสร้างสะพาน ตารางที่ 4-1 แสดงหน่วยแรงปลอดภัยในคอนกรีต และตารางที่ 4-2 แสดงหน่วยแรงปลอดภัยในเหล็กเสริมตามข้อบัญญัติของ ACI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-1 หน่วยแรงปลอดภัยในคอนกรีต

	MPa
1. Stresses immediately after prestress transfer (before prestress losses) shall not exceed the following:	
(a) Extreme fiber stress in compression	$0.60f'_{ci}$
(b) Extreme fiber stress in tension except as permitted in c	$-0.25\sqrt{f'_{ci}}$
(c) Extreme fiber stress in tension at ends of simply supported members Where computed tensile stresses exceed these values, bonded auxiliary reinforcement (nonprestressed or prestressed) shall be provided in the tensile zone to resist the total tensile force in the concrete computed with the assumption of an uncracked section.	$-0.50\sqrt{f'_{ci}}$
2. Stresses at service loads (after allowance for all prestress losses) shall not exceed the following:	
(a) Extreme fiber stress in compression	$0.45f'_{c}$
(b) Extreme fiber stress in tension in precompressed tensile zone	$-0.50\sqrt{f'_{c}}$
(c) Extreme fiber stress in tension in precompressed tensile zone of members (except two-way slab systems) where analysis based on transformed cracked sections and on bilinear moment-deflection relationships shows that immediate and long-time deflections comply with requirements stated elsewhere in the code	$-\sqrt{f'_{c}}$

3. The permissible stresses of sections 1 and 2 may be exceeded if whown by test or analysis that performance will not be impaired.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-2 หน่วยแรงปลอดภัยในเหล็กเสริมกำลังสูง

Tensile stress in prestressing tendons shall not exceed the following:	
1. Due to tendon jacking force whichever is smaller, but not greater than the maximum value recommended by the manufacturer of the prestressing tendons or anchorages	$0.80f_{pu}$ or $0.94f_{py}$
2. Pretensioning tendons immediately after prestress transfer	$0.70f_{pu}$
3. Post-tensioning tendons immediately after tendon anchorage	$0.70f_{pu}$

ค่าหน่วยแรงปลอดภัยในคอนกรีตขณะถ่ายแรงจะเท่ากับแฟคเตอร์คูณด้วยค่า fci' หรือ fci' ซึ่งเป็นกำลังของคอนกรีตขณะถ่ายแรง นอกจากนี้หน่วยแรงอัดปลอดภัยขณะใช้งานตามข้อกำหนดของ AASHTO เท่ากับ 0.4fc' สำหรับงานสะพาน ซึ่งมีความปลอดภัยสูงกว่าค่า 0.45fc'ในงานอาคาร และในงานสะพานข้อบัญญัติจะไม่อนุญาติให้เกิดหน่วยแรงดึงที่ผิวล่างของหน้าตัดและจะต้องออกแบบเหล็กเสริมธรรมดาด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบหาแรงอัดโดยวิธีน้ำหนักสมดุลย์

ความสัมพันธ์ของหลักการของคอนกรีตอัดแรงกับหลักการน้ำหนักสมดุลย์

หลักการของคอนกรีตอัดแรงทั้งสามข้อที่ได้กล่าวไปแล้วข้างต้นซึ่งได้แก่ข้อแรกโดยถือว่าคอนกรีตอัดแรงเป็นวัสดุที่มีความยืดหยุ่นจึงสามารถออกแบบโดยใช้ทฤษฎีความเค้นยืดหยุ่นได้ ส่วนข้อที่สองพิจารณาคอนกรีตอัดแรงเป็นคอนกรีตเสริมเหล็กและโดยที่คิดที่คุณสมบัติรับกำลังประลัยจึงใช้หลักการของทฤษฎีกำลังประลัยได้ ส่วนหลักการข้อที่สามโดยพิจารณาคอนกรีตอัดแรงให้โครงสร้างรับน้ำหนักสมดุลในช่วงแรกก่อน ซึ่งนั่นก็คือหลักการของวิธีน้ำหนักสมดุลย์นั่นเอง ซึ่งหลักการนี้ง่ายต่อการออกแบบและมีเนื้อหาเกินกว่าหลักการสองข้อข้างต้น ในการออกแบบโครงสร้างอินดิเทอมิเนต หลักการของวิธีน้ำหนักสมดุลย์จะมีประโยชน์มากทั้งในแง่ของการคำนวนและในแง่ของภาพรวมและมุมมอง

การที่จะทำความเข้าใจกับหลักการวิธีน้ำหนักสมดุลย์ให้สัมพันธ์กับหลักการอีกสองข้อต้องศึกษาพฤติกรรมของคอนกรีตอัดแรงตลอดช่วงอายุของชิ้นส่วนนั้น ๆ ภายใต้สภาวะยืดหยุ่น จากรูปจะแสดงรายละเอียดของความสัมพันธ์ระหว่างน้ำหนักกระทำและการแอ่นตัวของชิ้นส่วนคานช่วงเดียว จะมีจุดวิกฤตต่าง ๆ หลายจุดในช่วงอายุ

1. จุดที่ ไม่เกิดการแอ่นตัวความเค้นตลอดหน้าตัดจะเท่ากันเป็นรูปสี่เหลี่ยม
2. จุดที่ไม่เกิดแรงดึงรูปหน่วยแรงอัดจะเป็นรูปสามเหลี่ยมที่มีค่าเป็นศูนย์ที่ลวดรับแรงดึงด้านล่างในกรณีของคานช่วงเดียว
3. จุดที่เกิดการแตกร้าว ซึ่งจะเกิดขึ้นเมื่อลวดรับแรงดึงถึงค่าโมดูลัสแตกร้าว
4. จุดที่เกิดการครากเกิดขึ้นที่ลวดเหล็กมีค่าความเค้นเกินจุดคราก
5. จุดที่เกิดการพังทลายคือจุดที่ชิ้นส่วนรับน้ำหนักสูงที่สุดที่สามารถรับได้

จากรูปน้ำหนักที่สภาวะต่าง ๆเปลี่ยนแปลงไปอธิบายได้ดังนี้

1. น้ำหนักเกอเดอร์ (GL)
2. น้ำหนักคาน (DL)
3. น้ำหนักใช้งาน เกิดจากน้ำหนักคานรวมกับน้ำหนักจร (DL + LL)
4. น้ำหนักขณะเริ่มถึงจุดครากได้จากการคูณค่าแฟกเตอร์ความปลอดภัยกับค่าน้ำหนักน้อยที่สุดที่ทำให้ถึงจุดคราก(k1(DL + LL)
5. น้ำหนักขณะพังทลายได้จากการคูณค่าแฟกเตอร์ความปลอดภัยกับน้ำหนักน้อยที่สุดที่ทำให้เกิดการพังทลาย(k2(DL + LL)

หลักการของความเค้นจะใช้ในช่วงที่ตรงกับ (DL + LL) ซึ่งเป็นจุดที่ไม่เกิดแรงดึง ส่วนการออกแบบโดยทฤษฎีประลัยก็อยู่ในช่วงที่มีน้ำหนัก k2(DL + LL)ซึ่งเป็นช่วงที่ถึงจุดประลัย ส่วนหลักการของวิธีน้ำหนักสมดุลย์จะใช้ในช่วงที่มีน้ำหนักเท่ากับ DL + k3LL (k3 มีค่าเท่ากับ 0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

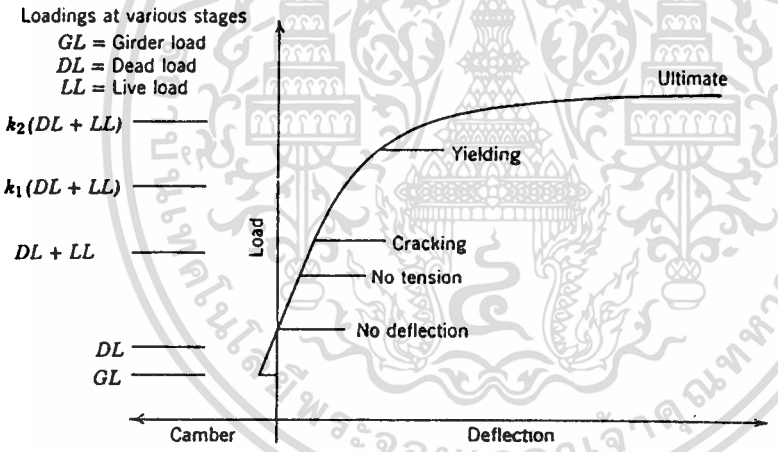
ซึ่งเป็นจุดที่ไม่เกิดการแอ่นตัว ในการออกแบบที่เหมาะสมมักใช้ค่าที่น้อยกว่าน้ำหนักคานจริงเล็กน้อยจะให้ค่าเหมาะสม ซึ่งในการออกแบบด้วยโปรแกรมนี้จะใช้ค่า 0.8 เท่าของน้ำหนักคาน

ดังนั้นเราจึงสามารถเปรียบเทียบความสัมพันธ์ระหว่างน้ำหนักที่กระทำกับลำดับขั้นตอนของพฤติกรรมของคานที่เกิดขึ้น

	Applied Loadings	Stages of Beam Behavior
1.	DL + k3LL	No deflection
2.	DL + LL	No tension or allowable tension
3.	k2(DL + LL)	Ultimate

จากรูปแสดงการสมดุลแรงกระจายที่กระจายแรงไปตามเส้นลวดโค้งซึ่งจะยกขึ้นส่วนขึ้นโดยน้ำหนักสมดุลย์ wb วิ่งได้จาก

$$wb = \frac{8Fh}{L^2}$$



รูปที่ 4.2 แสดงความสัมพันธ์ของช่วงอายุของคอนกรีตอัดแรงและการแอ่นตัว

หากว่ามีน้ำหนักกระจายลงมา(รวมน้ำหนักของคานด้วย)จะสามารถดุลน้ำหนักนั้นด้วย wb ซึ่งไม่ทำให้เกิดการโก่งในคาน ในคานก็จะอยู่ภายใต้สภาวะถูกอัดโดยหน่วยแรงอัดกระจาย

$$f = F/Ac$$

ถ้ามีน้ำหนักอื่น ๆ นอกเหนือจากน้ำหนัก wb กระทำแล้ว นำน้ำหนักดังกล่าวมาวิเคราะห์หาโมเมนต์และจะได้หน่วยแรงอัดที่เกิดจากน้ำหนักดังกล่าว $f = Mc/I$

แต่ถ้าในกรณีของคานยื่นแล้วสามารถหาระยะยกของลวดได้จาก $h = \frac{wL^2}{2F}$

และคานที่ต่อเนื่องกับคานยื่นยะมีระยะยกที่ช่วงกลางคาน $h1 = \frac{wL_1^2}{8F}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวางลวดเหล็กอัดแรง (Distribution of tendons in two way slab)

จำนวนเหล็กสามารถหาได้จากวิธีน้ำหนักสมดุล ในแต่ละ Stripทั้งสองทิศทาง สำหรับการวางเหล็ก หลังจากออกแบบแล้วเสร็จ ก็ถึงขั้นตอนการเขียนแบบเพื่อให้ผู้ทำงานเข้าใจ และทำงานได้อย่างถูกต้อง ข้อแนะนำสำหรับการวางลวดอัดแรงนั้นมีอยู่ประมาณ 4 แบบ ทั้งนี้จะเลือกแบบใดนั้นขึ้นอยู่กับความเหมาะสมหรือความสะดวกในการทำงานซึ่งก็ขึ้นอยู่กับวิจารณญาณของวิศวกรแต่ละคน

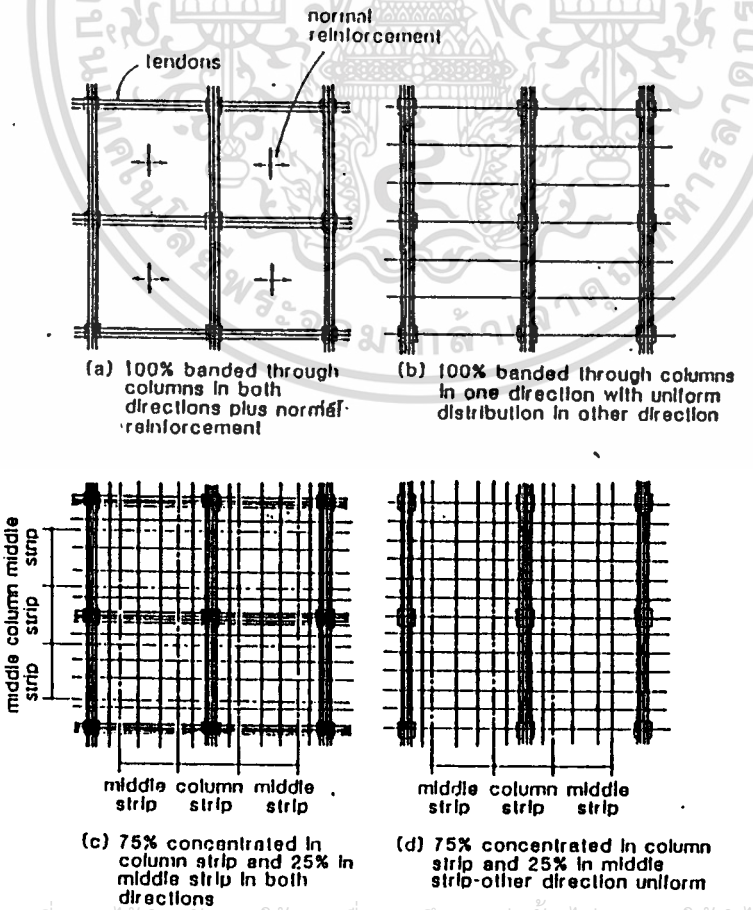
ระยะห่างของลวดนั้น ACI และนำไว้ดังนี้

	Bonded system	Unbonded system
ระยะสูงสุด	8 เท่าของความหนาพื้น	6 เท่าของความหนาพื้น

เหล็กเสริมกันร้าวที่หัวเสาในแต่ละทิศทางใช้ As = 0.0075bh

ห่างจากเสา 1.5h ระยะห่างไม่เกิน 12 นิ้ว

แต่ต้องไม่น้อยกว่า 4 เส้น



รูปที่ 4.3 แบบที่แนะนำสำหรับการกระจายเหล็กเสริมในคอนกรีตอัดแรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งนี้ ห้ามมิให้ลอกเลียนแบบ ดัดแปลง เผยแพร่ หรือนำไปใช้ในรูปแบบอื่นใดที่มีการนำไปใช้

บทที่ 5

สรุปขั้นตอนการคำนวณออกแบบ (DESIGN STEP)

ขั้นตอน	หลักการ	หมายเหตุ
1. หาความหนาพื้น (Slab Thickness)	จากอัตราส่วน l/t	จากข้อแนะนำ
2. ตรวจสอบแรงเฉือนแป้องต้น (Primary Shear Check)	Vinternal 1.20Vt Vedeg 1.50Vt Vcorner 1.30Vt	ถ้าหน่วยแรงเฉือนเกินให้เพิ่มความหนาของพื้นหรือเพิ่ม CAP หัวเสา หรือเพิ่ม DROP PANEL
3. หาระยะจุดสูงสุดของระยะ drape(a)	จากระยะสูงสุดและต่ำสุดของลวด โดยคิดการหุ้มของคอนกรีตจากอัตราการป้องกันไฟ	Covering 30 mm. หรือมากกว่า จากศูนย์กลางลวด Unboned = 8 mm Bonded = 12 mm
4. หาแรงอัดในคอนกรีต (P) ต่อหน่วยความกว้าง	Balancing Method	จะได้ค่า P = Wbl^2/(8a)
5. หาจำนวนลวดอัดแรงต่อหน่วยความกว้าง No. of Strand/m	จาก P/Feff	Feff = 0.6FyAps 11,000 kg 110 kn
6. วิเคราะห์โครงสร้าง Analysis Structure	โดยวิธี equivalent frame หรือวิธีอื่นๆ ที่เหมาะสม	
7. ตรวจสอบการสูญเสียแรงอัด	-Elastic shortening -Creep -Shrinkage -Relaxation -Friction and wobble -Anchorage seating set	เพื่อหาแรงอัดในลวดเมื่อใช้งานนานๆ
8. ตรวจสอบหน่วยแรงอัด Check stress at transfer and working load	f = P/A +-Pe/Z+-M/Z	ตรวจสอบหน่วยแรงอัดทั้งผิวบนและล่างของแผ่นพื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

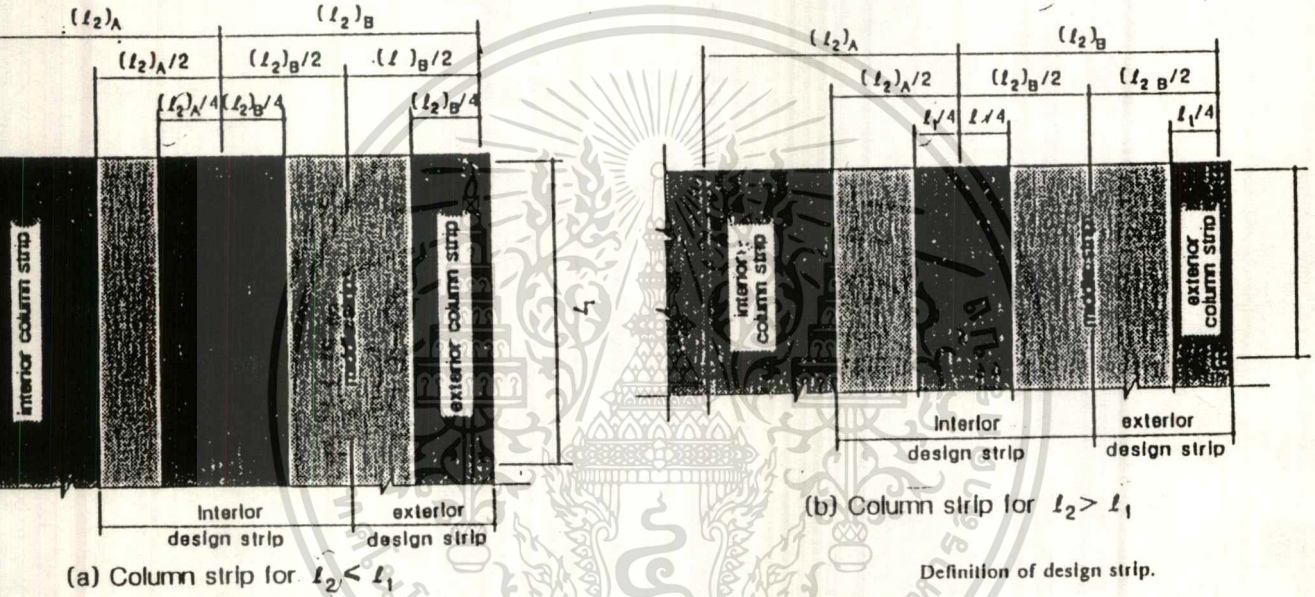
9. Check Ultimate Strength	Mu= 1.4MDL+1.7MLL+1.4MS(Secondary)	-วสท. -ACI
10. ตรวจสอบแรงเฉือน Check shear	Vpunching Vbeam Vshear+Moment	ถ้าแรงเฉือนที่ได้มากกว่าที่คอนกรีตจะรับได้จำเป็นออกแบบเหล็กรับแรงเฉือน หรือเพิ่มความหนาพื้น
11. ตรวจสอบการแอ่นตัว Check deflection		deflection l/360 l/125(พื้นยื่น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การวิเคราะห์โครงสร้างโดยวิธี Equivalent Frame Method

การวิเคราะห์โครงสร้างโดยวิธี Equivalent Frame Method เป็นวิธีที่นิยมใช้กันมากในการวิเคราะห์โมเมนต์ในพื้น Post tension โดยโครงสร้างที่วิเคราะห์จะถูกแบ่งออกเป็นโครงสร้างย่อย ๆ ดังรูป



รูปที่ 6.1

1)หา slab stiffness โมเมนต์อินเนเชีย ของพื้นที่บริเวณพื้นที่ไม่คิดผ่านเสา

$$Is = \frac{1}{2} L2h^3$$

โดยที่ h = ความหนาพื้น

L2 = ความกว้างของหน้าตัดพื้น

ส่วนโมเมนต์อินนิเชียของหน้าตัดเสา จะได้โมเมนต์อินเนเชียจากวิธีการของ Column Analogy ซึ่งเป็นวิธีการที่ยุ่งยากมาก ฉะนั้นค่าที่ใกล้เคียงโดยวิธีประมาณ

$$I1 = Is(1 - \frac{C2}{L2})^2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ C2 = ความกว้างของหน้าตัดเสา

เมื่อได้ค่าโมเมนต์อินเนเชียของหน้าตัดพื้นแล้วคำนวนหา Equivalent slab stiffness ซึ่งมีวิธีแนะนำไว้โดยประมาณดังนี้

$$Ks = 4I1\frac{Ecs}{(L1-\frac{C1}{2})}$$

โดยที่ Ks = equivalent slab stiffness

Ecs = concrete slab 's modulus of elasticity

L1 = ความยาวระหว่างศูนย์กลางเสา

C1 = ความลึกของหน้าตัดเสา

2) หา Column Stiffness

หาค่าโมเมนต์อินเนเชีย ของเสาระหว่างชั้นของพื้นโดยคำนวณจากหน้าตัดเสาปกติ

$$Ic = \frac{1}{12}C2C1^3$$

ส่วนโมเมนต์อินเนเชียของเสาที่อยู่ช่วงเดียวกับพื้นมีค่า มาก ฉะนั้นจะด้ค่าของ Column stiffness โดยประมาณ

$$Kc = 4Ec\frac{Ic}{(Lc-2h1)}$$

โดยที่ Kc = Column Stiffness

Ec = Concrete Column 's modulus of elasticity

Lc = ความสูงระหว่างชั้นของเสา

h1 = ความหนาของพื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3)จากรูปจะแสดงส่วนพื้นและเสามีพฤติกรรมรับแรงบิดร่วมกันฉะนั้นค่า equivalent stiffness ของเสาจะได้ดังนี้

$$1/Kec = 1/Kc + 1/Kt$$

โดยที่ Kec = equivalent column stiffness

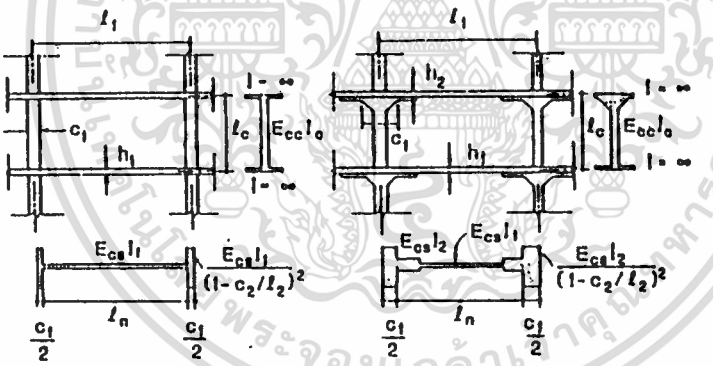
Kt = tension stiffness attached member

ซึ่ง $$Kt = \frac{9Ecs\,C}{\left[L2\left(1-\frac{C2}{L2}\right)^3\right]}$$

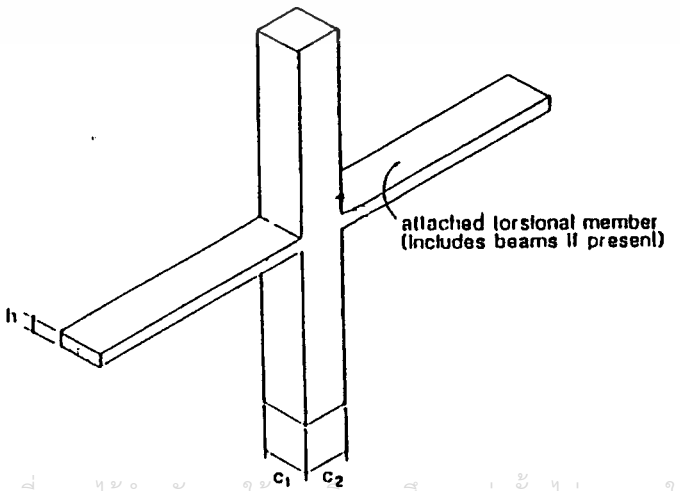
โดยที่ $$C = \Sigma\left(1-0.63\frac{x}{y}\right)x^3\frac{y}{3}$$

x = ความหนาของพื้น = h

y = ความลึกของเสา = C1



รูปที่ 6.2 Determining the slab and column stiffnesses.



รูปที่ 6.3 Column with attached torsional members.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) หาค่า Distribution factor

ค่า Distribution factor ที่เกิดขึ้นในแต่ละชุดของโครงสร้าง

DF(ext) = Ks/(Ks + $\sum$ Kec) สำหรับเสาช่วงนอก

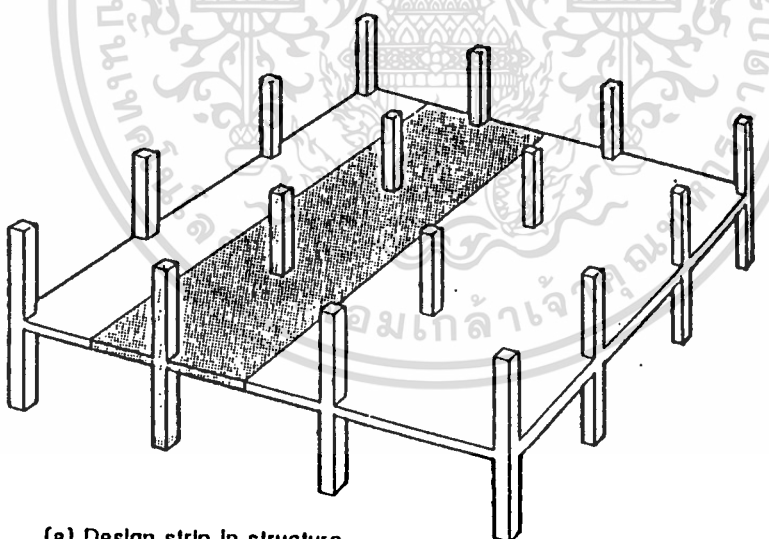
DF(int) = Ks/($\sum$Ks + Kec) สำหรับเสาช่วงใน

5) หาค่า Carry Over Factor

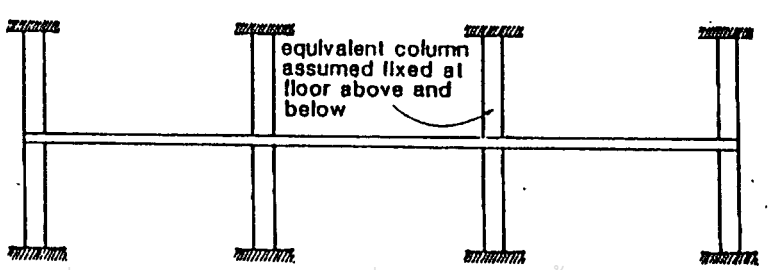
สำหรับพื้น Flat plate ใช้ค่าโดยประมาณ 0.50 สำหรับ Flat Slab ที่มี drop panel ให้ใช้ค่าประมาณจากตาราง

6) การวิเคราะห์หาค่าโมเมนต์จากการกระจายโมเมนต์

นำค่า carry over factors ,distribution factors ,fixed-end moment ที่หามาได้จากขั้นตอนก่อนหน้านี้มาทำการกระจายตามขั้นตอน การกระจายโมเมนต์ที่สร้างขึ้น เพื่อทำการวิเคราะห์หาโมเมนต์ โดยค่า carry over moment รอบแรกนั้นได้จากการกระจายผลรวมโมเมนต์ของ carry over ด้านตรงข้ามที่ได้ในรอบก่อน หน้านั้น ทำ 3 รอบจากนั้นนำผลรวมโมเมนต์จาก fixed -end moment และ carry over moment ที่ได้มารวมกันในแต่ละ joint จากนั้นก็ทำการกระจายโมเมนต์ที่รวมโดยมีค่าที่เป็นลบแล้วนำกลับมาบวกกับผลรวมโมเมนต์ที่ได้ในตอนแรกก็จะได้ผลรวมในแต่ละ Joint



(a) Design strip in structure



(b) Idealized two-dimensional frame for design strip

รูปที่ 6.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

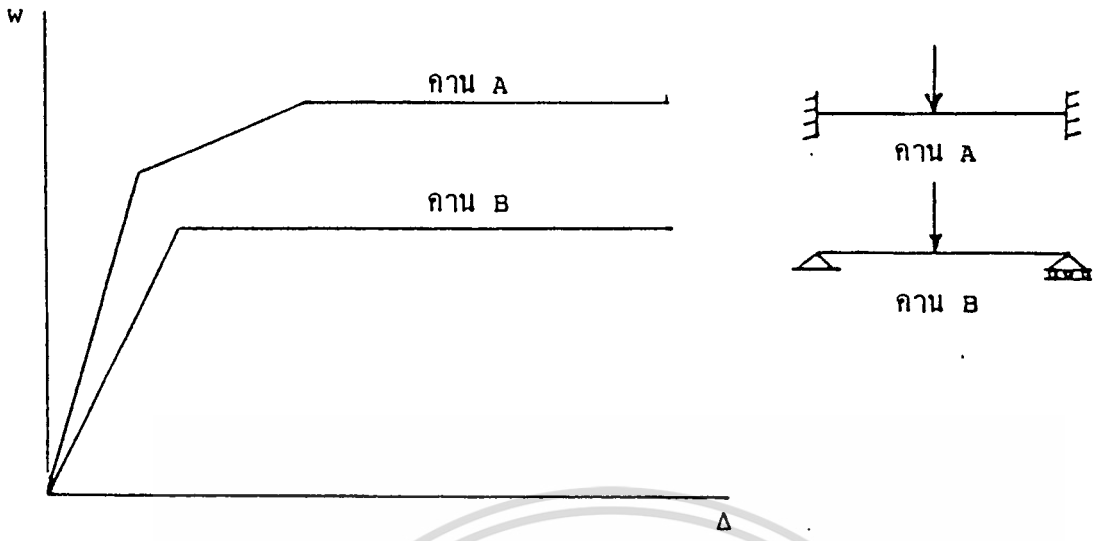
คานต่อเนื่องและโครงสร้างอินดิเทอร์มิเนท

เราได้ศึกษาพฤติกรรมการวิเคราะห์และออกแบบคานคอนกรีตอัดแรงช่วงธรรมดามาแล้ว ซึ่งโมเมนต์ดัดและแรงเฉือนสามารถหาได้จากสมการสมดุลย์คานชนิดนี้เหมาะสำหรับคานช่วงสั้นหรือมีความยาวปานกลาง เช่นในกรณีที่คานสะพานซึ่งมีช่วงความยาวประมาณ 20 ถึง 25 เมตร รับน้ำหนักบรรทุก SH 20-44 จะมีความลึกประมาณ 1.70 เมตร ในกรณีที่ช่วงคานยาวมากกว่านี้ การออกแบบคานเป็นชนิดช่วงเดี่ยวธรรมดาจะไม่ประหยัดเพราะคานจะมีน้ำหนักและความลึกมาก ทำให้แรงอัดคานต้องเพิ่มมากขึ้นและมีปัญหาเกี่ยวกับค่าแอ่นตัวในช่วงใช้งานมากเกินไป และอาจเป็นตัวควบคุมการออกแบบถึงแม้ว่าหน่วยแรงจะไม่เกินหน่วยแรงปลอดภัยและกำลังประลัยของคานพอเพียง ดังนั้นสำหรับคานสะพานช่วงยาวมาก ๆ การออกแบบคานต่อเนื่องมีความเหมาะสม เพราะสามารถประหยัดประหยัดปริมาณคอนกรีตและเหล็กเสริมได้มาก อนึ่งในโครงสร้างอาคารคานซึ่งรองรับพื้นที่หล่อในที่จะเป็นส่วนหนึ่งของโครงสร้างอินดิเทอร์มิเนท ซึ่งการวิเคราะห์โมเมนต์ดัดจะต้องพิจารณาการแจกแจงของโมเมนต์ที่จุดต่อเนื่องของคาน โมเมนต์ดัดนี้เป็นผลต่อเนื่องมาจากแรงกระทำภายนอกและผลของการเยื้องศูนย์ของแรงอัดคาน

พฤติกรรมที่ต่างกันของคานที่ต่อเนื่องเทียบกับคานช่วงเดี่ยวธรรมดาอีกประการหนึ่งก็คือความสามารถในการรับแรงกระทำเพิ่มขึ้นเนื่องจากการถ่ายเทแจกแจงโมเมนต์ดัดที่จุดต่อเนื่องภายหลังที่หน้าตัดหนึ่งเกิดพลาสติคฮิงค์ พฤติกรรมนี้จะเกิดขึ้นได้เมื่อคานมีความเหนียวมาก ทำให้หน้าตัดคานหมุนตัวได้สูงขณะต้านทานโมเมนต์ดัดที่คงที่หรือเพิ่มขึ้นในรูปที่ 7-1 เปรียบเทียบความสามารถในการรับน้ำหนักบรรทุกของคานช่วงเดี่ยวธรรมดา และคานที่มีฐานรองรับยึดแน่น คานทั้งสองมีขนาดหน้าตัดและเหล็กเสริมเท่ากัน คานที่ปลายยึดแน่นรับน้ำหนักได้มากกว่าเนื่องจากการถ่ายเทแจกแจงโมเมนต์ดัด แต่ในคานช่วงเดี่ยาธรรมดาไม่มีการถ่ายเทโมเมนต์ดัดนี้

การก่อสร้างคานต่อเนื่องแบ่งเป็นประเภทใหญ่ ๆ ได้สองชนิดคือการก่อสร้างแบบต่อเนื่องทั้งหมด การก่อสร้างแบบต่อเนื่องบางส่วน การก่อสร้างแบบต่อเนื่องทั้งหมดจะต้องทำนั่งร้านค้ำยัน เพื่อหล่อคอนกรีตคานในที่และอัดแรงให้คานแบบดึงเหล็กทีหลังในการก่อสร้างต่อเนื่องแบบบางส่วน คานที่หล่อไว้ก่อนจะถูกนำมาวางบนตอม่อเป็นลักษณะคานช่วงเดี่ยวธรรมดา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

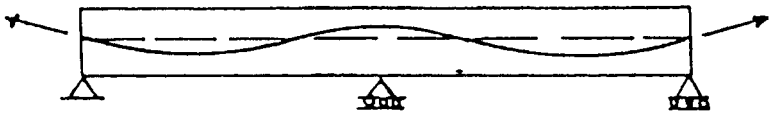


รูปที่ 7-1 พฤติกรรมคานช่วงเดี่ยวธรรมดาและคานฐานยึด

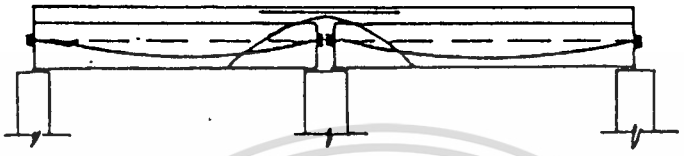
หลังจากเทคอนกรีตพื้นจนแข็งตัวดีแล้ว จึงทำการอัดแรงคานที่บริเวณจุดต่อเชื่อมในสนามโดยการดึงเหล็กในท่อคอนดุทรอบจุดต่อเชื่อมซึ่งเตรียมไว้ก่อน แรงภายนอกซึ่งกระทำหลังจากนั้นจะต้านทานโดยระบบคานต่อเนื่อง

ปัญหาหนึ่งซึ่งมักจะเกิดขึ้นในคานต่อเนื่องก็คือ มีการสูญเสียแรงอัดคานเนื่องจากแรงเสียดทานสูง ทั้งนี้เนื่องจากคานต่อเนื่องตั้งแต่สองถึงสามช่วงขึ้นไป เหล็กเสริมมีความยาวมาก ซึ่งจะทำให้เกิดแรงเสียดทานเนื่องจากความคดของเหล็กเสริมประการหนึ่ง อีกประการหนึ่งเป็นแรงเสียดทานเนื่องจากความโค้งของเหล็กเสริมตลอดความยาวของคาน ซึ่งตามปกติจะมีค่ามากกว่าผลของความคด กล่าวคือในการออกแบบแนวโค้งของเหล็กเสริมจะมีรูปร่างคล้ายไดอะแกรมของโมเมนต์ดัดทั้งนี้เพื่อให้โมเมนต์ดัดเนื่องจากแรงกระทำภายนอก ตามที่ทราบมาแล้วว่าในช่วงใช้งานแรงดึงในเหล็กเสริมมีค่าเกือบคงที่ตลอดความยาวคาน ดังนั้นโดยหลักของแรงคู่ควบถ้าระยะเยื้องศูนย์ของเหล็กเสริมเท่ากับโมเมนต์ดัดเนื่องจากแรงกระทำ หารด้วยแรงดึงในเหล็กที่ทุก ๆ หน้าตัดของคานตลอดความยาว ผลของการหักล้างโมเมนต์ดัดจะเกิดขึ้นพอดี เราเรียกแนวโค้งของเหล็กเสริมนี้ว่า แนวคอนคอแดนต์ แต่ในทางปฏิบัติเราจะออกแบบเหล็กเสริมให้ระยะเยื้องศูนย์มีค่ามากที่สุดเพื่อให้แรงอัดคานหรือแรงดึงในเหล็กมีค่าน้อยที่สุดเพื่อเหตุผลทางด้านการปฏิบัติ ดังนั้นแนวเหล็กเสริมจะเบี่ยงเบนจากแนวคอนคอแดนต์ ประกอบกับแนวโค้งบริเวณจุดรองรับภายในจะมีลักษณะเป็นเส้นโค้งเรียบทำให้แนวโค้งจริงของเหล็กเสริมประกอบด้วยส่วนของเส้นโค้งที่มีเครื่องหมายกลับกันหลาย ๆ ส่วน ผลรวมของมุมโค้งตลอดความยาวของเหล็กเสริมจะมีค่ามากทำให้แรงเสียดทานเนื่องจากความโค้งมีค่ามากไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก. คานต่อเนื่องทั้งหมด

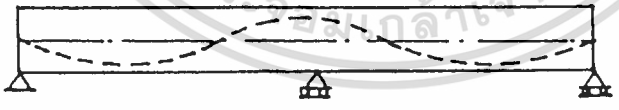


ข. คานต่อเนื่องบางส่วน

รูปที่7-2 การก่อสร้างแบบต่อเนื่องทั้งหมดและต่อเนื่องบางส่วน



ก. แนวเหล็กเสริมคอนคอรแดนท์

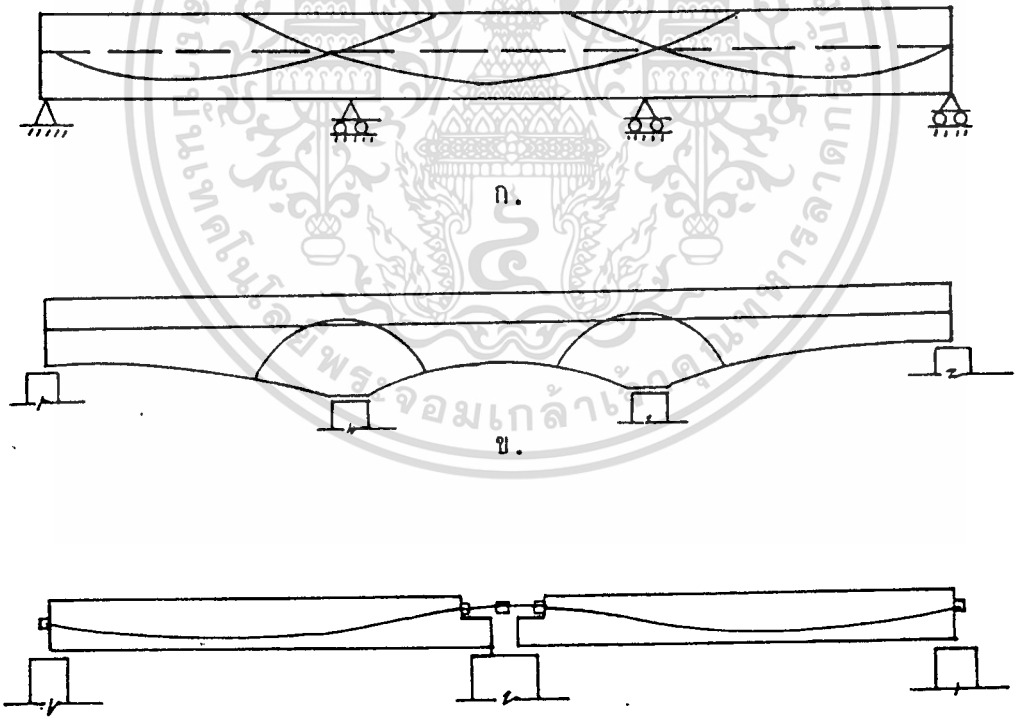


ข. แนวเหล็กเสริมเชิงปฏิบัติ

รูปที่7-3 แนวโค้งของเหล็กเสริม

ในกรณีที่แรงเสียดทานในเหล็กเสริมมีค่ามากเกินไป ผู้ออกแบบจะต้องพิจารณาการออกแบบรายระเอียดและวิธีก่อสร้างเพื่อทำให้แรงเสียดทานมีค่าน้อยที่สุด เราอาจพิจารณาอัดแรงโดยใช้แนวเหล็กเสริมที่เหลื่อมกันเหนือจุดรองรับภายใน เพื่อลดความโค้งของมุมคัดกลับ วิธีนี้จะทำการดึงเหล็กจากด้านบนของแนวเหล็กเสริมต่าง ๆ โดยจะต้องออกแบบรายละเอียดการฝังอุปกรณ์ยึด ซึ่งจะหล่อคอนกรีตปิดหลังจากปล่อยแรงอัดแล้ว รูปที่9-4ข การเปลี่ยนแปลงของระยะเยื้องศูนย์อาจทำได้โดยการออกแบบเป็นคานที่มีความลึกไม่สม่ำเสมอ และใช้แนวเหล็กเสริมเป็นเส้น

ตรง ที่จุดรองรับภายในเราทำการดึงเหล็กโดยใช้แนวโค้งช่วงสั้นเพื่อปรับระยะเยื้องศูนย์ให้มีเครื่องหมายกลับจากระยะเยื้องศูนย์ให้มีเครื่องหมายกลับจากระยะเยื้องศูนย์ที่บริเวณกลางคาน รูปที่ 7-4ค แสดงการดึงเหล็กในการก่อสร้างแบบต่อเนื่องบางส่วน ในที่นี้เราดึงเหล็กในคานช่วงที่หนึ่งในล็อกอุปกรณ์ยึดที่ปลายทั้งสองข้าง หลังจากนั้นจึงจึงทำการต่อเหล็กเสริมโดยใช้อุปกรณ์ต่อเหล็ก แล้วจึงใช้แม่แรงดึงดึงเหล็กจากปลายขวามือของคานช่วงที่สอง เมื่อดึงเหล็กเสร็จแล้วจึงเสริมเหล็กพื้นและหล่อคอนกรีตให้กลายเป็นคานต่อเนื่อง การก่อสร้างคานต่อเนื่องสามช่วงหรือหลาย ๆ ช่วงก็มีลำดับขั้นตอนก่อสร้างทำนองเดียวกันวิธีนี้จะต้องทิ้งอุปกรณ์ต่อเหล็กไว้ในคานอย่างถาวร แต่เป็นการลดแรงเสียดทานเพราะทำการดึงเหล็กในคานทีละช่วง การลดแรงเสียดทานอีกประการหนึ่งทำได้โดยการเลือกใช้ท่อคอนดุทที่มีสัมประสิทธิ์ของแรงเสียดทานน้อย วิธีนี้อาจจะได้ผลไม่มากนัก ผู้ออกแบบควรออกแบบคานแต่ละช่วงเป็นคานชนิดดึงเหล็กก่อน โดยภายในคานจะฝังท่อคอนดุทไว้สำหรับดึงเหล็กเพื่อทำให้เป็นคานต่อเนื่องในภายหลัง



รูปที่ 7-4 วิธีการดึงเหล็กในคานต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลของแรงอัดคานในคานต่อเนื่อง

การวิเคราะห์หน่วยแรงในช่วงปล่อยแรงและช่วงปล่อยแรงและช่วงใช้งานของคานต่อเนื่องใช้ทฤษฎีอิลาสติก ซึ่งในที่นี้จะพิจารณาเฉพาะกรณีการก่อสร้างระบบต่อเนื่องทั้งหมดก่อนอื่นจะขอกล่าวถึงสมมุติฐานต่าง ๆ ดังต่อไปนี้

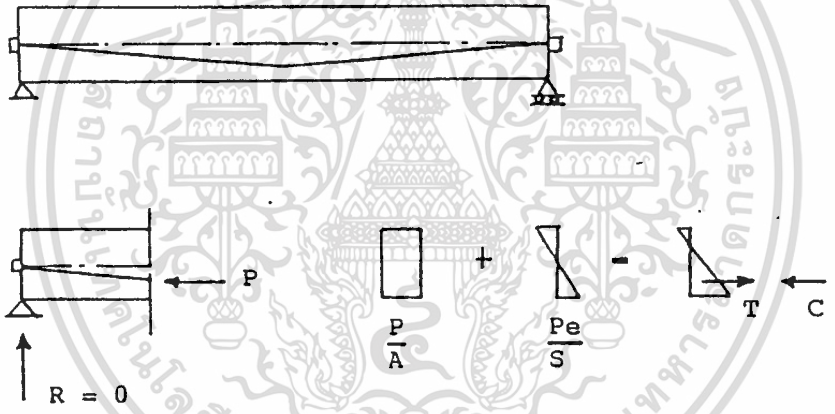
ก. คานมีหน้าตัดเหมือนกันทุกประการตลอดความยาวของคาน

ข. แรงอัดคานมีค่าคงที่ตลอดความยาวของคาน

ค ใช้คุณสมบัติของหน้าตัดรวม

ง. จุดรองรับของคานไม่มีการทรุดตัว

พิจารณาคานคอนกรีตอัดแรงช่วงเดี่ยวธรรมดาภายใต้การกระทำของแรงอัดคานอย่างเดียวมีระยะเยื้องศูนย์ดังรูปที่ 7-5 ตามหลักการของแรงคู่ควบที่หน้าตัดใด ๆ ในคาน

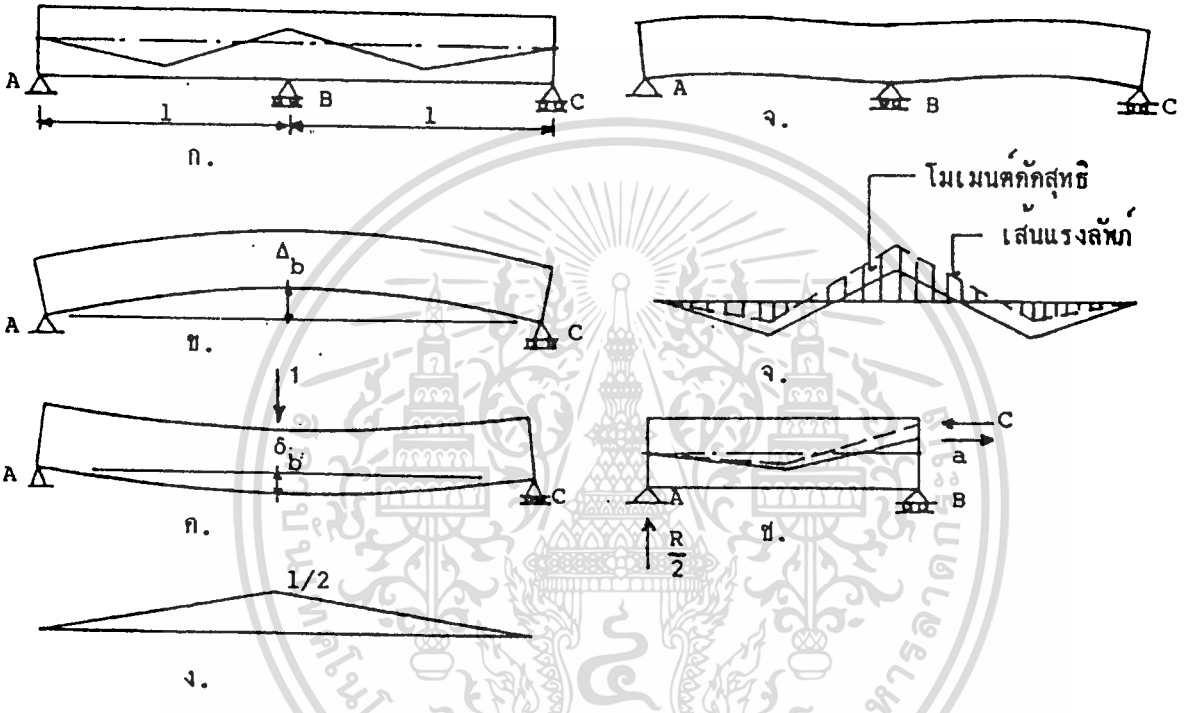


รูปที่7-5 ผลของแรงอัดคานช่วงเดี่ยวธรรมดา

ตำแหน่งของแรงอัดลัพธ์ในคอนกรีตจะตรงกับตำแหน่งของแรงดึงในเหล็กเสริม นั่นคือจะไม่มีโมเมนต์ดัดของแรงคู่ควบเนื่องจากแขนของแรงคู่ควบเป็นศูนย์ หรืออาจกล่าวได้ว่าไม่มีโมเมนต์รองของแรงอัดคานนั่นคือจะไม่มีแรงปฏิกิริยาที่จุดรองรับ ดังนั้นถ้าเราสมมุติว่าน้ำหนักคานมีค่าเป็นศูนย์ คานจะสามารถลอยอยู่เฉย ๆ ได้ภายใต้แรงอัดคานอย่างเดียว

ซึ่งข้อสรุปนั้นไม่จำเป็นต้องเป็นจริงในคานต่อเนื่องอาจกล่าวได้ว่าคานซึ่งมีแรงอัดคานกระทำอย่างเดียว ในขณะที่คานที่มีแนวเหล็กไม่เป็นแนวคอนคอแคนท์ จะเกิดแรงปฏิกิริยาที่จุดรองรับเนื่องจากแรงอัดคานพิจารณาคานต่อเนื่องสองช่วงดังรูปที่ 7-6 ตามหลักการวิเคราะห์เฟลกซิบิลิตี้ ขั้นแรกเราสมมุติเอาจุดรองรับภายในออกไปทำให้คานต่อเนื่องกลายเป็นคานช่วงเดียวธรรมดา โมเมนต์ดัดเนื่องจากแรงอัดคานคูณกับระยะเยื้องศูนย์หรือที่เรียกว่าโมเมนต์หลักทำให้คานโก่งขึ้นดังในรูป7-6ข คานจะแอ่นตัวลงและระยะแอ่นตัวที่จุดกึ่งกลางคานเท่ากับ Δb

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเราให้แรงหนึ่งหน่วยกระทำลงที่จุด Bของคานช่วงเดี่ยวธรรมดา รูปที่ 7-6ค คานจะแอ่นตัวลงและระยะแอ่นตัวที่จุดกึ่งกลางคานเท่ากับ δ_b รูปที่7-6 ง แสดงไดอะแกรมของโมเมนต์ดัดเนื่องจากแรงหนึ่งหน่วย เนื่องจากคานในรูปที่ 7-6ก มีการแอ่นตัวที่จุดรองรับภายในเป็นศูนย์ ดังนั้นโดยการเขียนสมการคอมพาทติบิลิตี้ ของการแอ่นตัวที่จุด B การคำนวนหาแรงปฏิกิริยาที่จุด B ได้ดังนี้

$$\Delta b - R\delta_b = 0$$

$$R = \Delta b / \delta_b$$

โดยที่ R = แรงปฏิกิกริยาที่จุด B

จากหลักการวิเคราะห์โครงสร้างวิธีแรงหนึ่งหน่วยค่าแอ่นตัวที่จุด B เท่ากับ

$$\Delta b = \int_0^{2l} (Pe)m\frac{dx}{EI}$$

และ

$$\delta_b = \int_0^{2l} m^2\frac{dx}{EI}$$

ดังนั้น

$$R = \int_0^{2l} (Pe)m\frac{dx}{EI} \Big/ \int_0^{2l} m^2\frac{dx}{EI}$$

โดยที่ P = แรงอัดคานในที่นี้เป็นค่าคงที่

e = ระยะเยื้องศูนย์ที่หน้าตัดใด ๆ ในคาน

m = โมเมนต์ดัดที่หน้าตัดใด ๆ ตามรูปที่ เนื่องจากแรงกระทำหนึ่งหน่วย

E = โมดูลัสความยืดหยุ่นของคาน

I = โมเมนต์ความเฉื่อยของหน้าตัด

การอินทิเกรดกระทำจากซ้ายไปขวาตลอดความยาว 2l ของคานสองช่วง

เมื่อทราบแรงปฏิกิริยา R ที่จุดรองรับ B เราสามารถคำนวนแรงปฏิกิริยาที่จุด A และ C ได้ ในที่นี้เท่ากับ R/2 แรงปฏิกิริยานี้เกิดขึ้นเนื่องจากแรงอัดคานเพราะแนวเหล็กเสริมในรูป 7-6 ก ไม่เป็นแนวคอนคอร์แดนท์ ตามหลักการแรงคู่ควบของการอัดแรง จะมีการเคลื่อนตัวของแนวแรงลัพท์ C จากแนวแรงดึง T ในเหล็กเสริมที่ทุก ๆ หน้าตัดของคาน โดยระยะเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวเป็นสัดส่วนกับค่าคอร์ดิเนทของโมเมนต์ไดอะแกรมในรูปที่ 7-6 ง คูณด้วยค่า R เช่นที่หน้าตัด B มีโมเมนต์ของแรงคู่ควบเป็น RI/2 ดังนั้นจากรูปที่7-6 ช

$$C \times a = RI/2$$

จะได้ $$a = RI/2C$$

ที่หน้าตัด x วัดจากปลายซ้ายมือระยะเคลื่อนตัวจะเท่ากับ

$$a_x = \frac{x}{l}a = \frac{x}{l}(\frac{Rl}{2C})$$

เราเรียกการเคลื่อนตัวเช่นนี้ว่าลิเนียร์ทรานฟอร์เมชั่น

รูปที่ 7-6 ฉ แสดงกราฟการแจกแจงของแนวจุดศูนย์ถ่วงของเหล็กเสริมและแนวแรงลัพธ์ C ตลอดความยาวของคาน จะได้ว่าค่าคอร์ดิเนทของแรง C ที่หน้าตัดใด ๆ เมื่อคูณกับแรงลัพธ์จะได้ค่าโมเมนต์ดัดสุทธิ ซึ่งแสดงด้วยบริเวณที่แรเงาในรูป

การวิเคราะห์โมเมนต์รองและโมเมนต์สุทธิโดยวิธีเฟลกซิบิลิตี้ โดยการเลือกแรงปฏิกิริยาที่จุดรองรับภายในเป็นรีดันแดนท์ มีความยุ่งยากมากขึ้นสำหรับคานต่อเนื่องที่มีช่วงหลาย ๆ ช่วง ในกรณีดังกล่าวการวิเคราะห์โดยวิธีแจกแจงโมเมนต์จะมีความสะดวกกว่า วิธีมีขั้นตอนดังนี้

1.จากแนวเหล็กเสริมในคานหาไดอะแกรมของโมเมนต์หลักในคานทุก ๆ ช่วง

2.วิเคราะห์หาไดอะแกรมของแรงกระทำซึ่งเป็นผลของโมเมนต์หลัก

3.สมมุติจุดรองรับเป็นชนิดปลายยึดเนื่องจากแรงกระทำในข้อสอง

4.วิเคราะห์การแจกแจงของโมเมนต์ปลายยึดจะได้โมเมนต์ดัดสุทธิในคาน

5.จากการหักโมเมนต์หลักจากโมเมนต์ดัดสุทธิจะได้โมเมนต์รอง ทำการคำนวนหาแรงปฏิกิริยาที่จุดรองรับต่าง ๆ

ลิเนียร์ทรานฟอร์เมชั่น

ในคานต่อเนื่องผลของแรงอัดคานก่อให้เกิดแรงปฏิกิริยาที่จุดรองรับต่าง ๆ แนวแรงอัดลัพธ์จะเคลื่อนตัวไปจากแนวเหล็กเสริม โดยระยะห่างระหว่างแนวสองแนวนี้มีค่าเป็นศูนย์ที่ปลายสุดของคานและมีค่ามากที่สุดที่จุดรองรับภายในช่วงระยะห่างนี้จะเปลี่ยนแปลงเป็นสัดส่วนโดยตรงกับระยะทางวัดจากปลายคาน ลักษณะการเคลื่อนตัวนี้เรียกว่า ลิเนียร์ทรานฟอร์เมชั่น คุณสมบัตินี้มีประโยชน์สำหรับการออกแบบแนวเหล็กเสริมหลาย ๆ แนวในคานเดียวกันซึ่งมีรูปร่างเป็นเส้นกราฟคล้ายกัน จะให้แนวแรงอัดลัพธ์เส้นเดียวกัน ในนัยกลับกันเราอาจกล่าวได้ว่าในคานต่อเนื่อง แนวแรงอัดลัพธ์มีตำแหน่งคงที่ในขณะที่แนวเหล็กเสริมที่มีลักษณะกราฟคล้ายกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถจะเคลื่อนตัวเหนือจุดรองรับภายในไปอยู่ในแนวต่าง ๆ เท่ากับผลคูณของแรงอัดคานระยะเยื้องเบนระหว่างแนวแรงอัดลัพธ์และแนวเหล็กเสริมที่หน้าตัดนั้น ดังนั้นโมเมนต์รองเนื่องจากแนวเหล็กเสริมหลาย ๆ แนวจะมีค่าไม่เท่ากัน

จะเห็นได้ว่าแรงกระทำภายในมีค่าเท่ากัน ดังนั้นหลังจากการแจกแจงโมเมนต์ปลายยึดจะได้ไดอะแกรมโมเมนต์สุทธิ

แนวคอนคอแดนท์

ในคานคอนกรีตอัดแรงแนวคอนคอร์แดนท์ของเหล็กเสริม คือแนวโค้งของเหล็กเสริมที่ทับกันกับแนวแรงลัพธ์ในคานพอดี หมายความว่าถ้าแนวโค้งของเหล็กเสริมเป็นแนวคอนคอร์แดนท์โมเมนต์รองเนื่องจากแรงอัดคานมีค่าเป็นศูนย์ที่ทุก ๆ หน้าตัดของคาน และคานจะไม่มีแรงปฏิกิริยาที่จุดรองรับ ดังนั้นในคานช่วงเดี่ยวธรรมดาแนวเหล็กเสริมทุกแนวเป็นแนวคอนคอร์แดนท์ แต่ในคานต่อเนื่องโดยทั่วไปแนวเหล็กเสริมไม่เป็นแนวคอนคอร์แดนท์ ดังนั้นแนวแรงลัพธ์จะเคลื่อนตัวจากแนวเหล็กตามลักษณะลิเนียร์ทรานฟอร์เมชั่น แนวคอนคอรแดนท์มีประโยชน์แต่เพียงการวิเคราะห์หน่วยแรงในคานต่อเนื่องทำได้สะดวกแต่ในการออกแบบผู้ออกแบบไม่จำเป็นต้องเลือกแนวเหล็กเสริมเป็นแนวคอนคอร์แดนท์ แต่ควรออกแบบแนวเหล็กเสริมให้มีระยะเยื้องศูนย์มากที่สุดที่เป็นไปได้ทั้งภายในช่วงและเหนือที่รองรับ โดยตรวจสอบให้แนวแรงลัพธ์มีตำแหน่งที่เหมาะสม

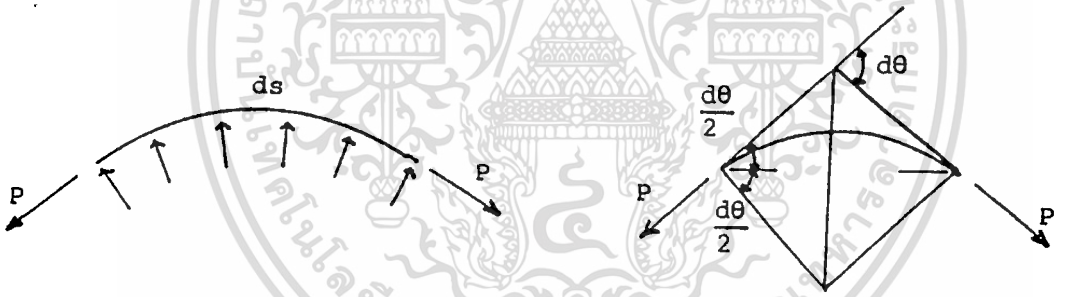
อนึ่งในการกำหนดแนวคอนคอร์แดนท์ เราอาจหาจากรูปร่างของโมเมนต์ไดอะแกรมของแรงกระทำใด ๆ บนคานต่อเนื่อง ซึ่งมีจุดรองรับที่ไม่มีการทรุดตัว กล่าวคือระยะจากจุดศูนย์ถ่วงถึงแนวคอนคอร์แดนท์ ที่หน้าตัดใด ๆ จะเท่ากับค่าโมเมนต์ดัดเนื่องจากแรงกระทำบนคานที่จุดนั้น ซึ่งเขียนด้วยมาตราส่วนอะไรก็ได้ ดังนั้นจะเห็นว่าแนวคอนคอร์แดนท์จะมีหลาย ๆ แนวขึ้นอยู่กับจำนวนรูปแบบของแรงกระทำบนคาน เหตุผลก็คือเปรียบเทียบโมเมนต์ไดอะแกรมกรณีที่คานซึ่งมีจุดรองรับไม่ทรุดตัวมีแรงกระทำตามขวาง กับกรณีที่คานมีแรงอัดกระทำในแนวเหล็กเสริมซึ่งคล้ายกับรูปร่างของโมเมนต์ไดอะแกรมของกรณีแรก โมเมนต์ไดอะแกรมทั้งสองกรณีจะคล้ายคลึงกันดังนั้นคานจึงไม่เกิดการแอ่นตัว และไม่มีแรงปฏิกิริยาที่จุดรองรับ นั่นคือแนวเหล็กเสริมเป็นแนวคอนคอร์แดนท์ สังเกตว่าแนวแรงอัดลัพธ์ใด ๆ ในคานต่อเนื่องก็คือแนวคอนคอร์แดนท์นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแรงอัดคานเป็นแรงกระทำด้านข้าง

ในการวิเคราะห์หาโมเมนต์ดัดสุทธิเนื่องจากแรงอัดคานโดยวิธีแจกแจงโมเมนต์ เราเริ่มต้นจากการเขียนไดอะแกรมของโมเมนต์หลัก จากนั้นจึงหาไดอะแกรมของแรงเฉือนและแรงกระทำด้านข้างโดยใช้ความสัมพันธ์ของแรงภายในตามหลักกลศาสตร์ของวัสดุ กล่าวคือ $V = dM/dx$ และ $w = -dV/dx$ โดยที่ M , V และw คือโมเมนต์ดัด แรงเฉือนและแรงกระทำสม่ำเสมอที่หน้าตัดใด ๆ ของคานการวิเคราะห์ดังกล่าวมีขั้นตอนกลับกันกับหลักกลศาสตร์ ซึ่งเราเริ่มต้นจากฟังค์ชั่นของแรงกระทำไปหาแรงเฉือนและโมเมนต์ดัด ในกรณีทั่วไปแนวเหล็กเสริมประกอบด้วยส่วนของเส้นโค้งหลาย ๆ ช่วงต่อกัน เราสามารถจะหาฟังค์ชั่นของแรงดันสม่ำเสมอด้านข้างที่กระทำต่อคานเนื่องจากแรงดึงในเหล็กเสริมได้ดังนี้

พิจารณาอิลิเมนต์ของเหล็กเสริมซึ่งเป็นเส้นโค้งยาว ds มีแรงดึง P กระทำที่ปลายทั้งสองข้างดังรูปที่ 7.7 จากรูปสี่เหลี่ยมด้านขนานของแรง



รูปที่ 7.7 อิลิเมนต์ของเหล็กเสริม

$$dQ = 2P\sin d\theta/2 \approx 2Pd\theta/2 = Pd\theta$$

ดังนั้นแรงดันสม่ำเสมอด้านข้าง, w เท่ากับ

$$w = dQ/ds = Pd\theta/ds = P/R$$

โดยที่ R = รัศมีความโค้ง

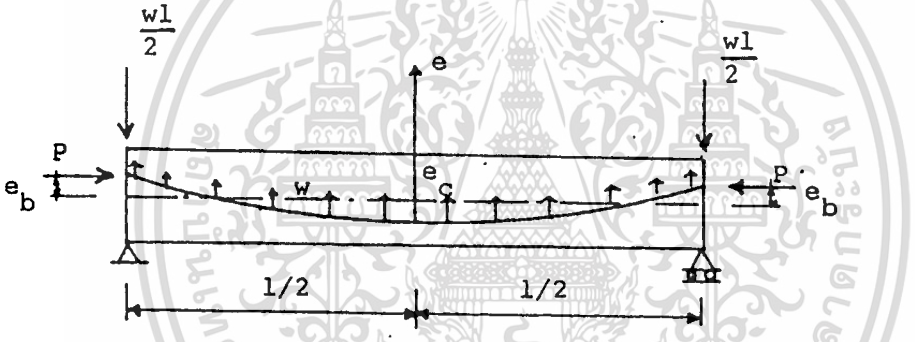
เนื่องจากแนวโค้งของเหล็กเสริมเป็นกราฟที่มีความชันน้อย จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{1}{R} \approx \frac{d^2y}{dx^2}$$

ดังนั้น $$w = \frac{Pd^2y}{dx^2} \qquad \text{----------A}$$

ในกรณีทั่วไปถ้าเราทราบฟังค์ชั่นของแนวเหล็กเสริมเช่นในรูปที่ 7.8 แนวเหล็กเสริมมีฟังค์ชั่นเป็นรูปพาราโบลาดังนี้



รูปที่ 7.8 ฟังค์ชั่นพาราโบลา

$$e(x) = \frac{4(eb - ec)x^2}{l^2} + ec$$

จะได้ $$\frac{de}{dx} = \frac{eb - ec}{2}(8x)$$

และ $$\frac{d^2e}{dx^2} = \frac{8(eb - ec)}{l^2}$$

จากสมการที่ A จะได้แรงดันด้านข้างดังนี้

$$w = \frac{8P}{l^2}(eb - ec) = \frac{8P}{l^2}h \qquad \text{----------B}$$

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับแนวเหล็กเสริมซึ่งประกอบด้วยกราฟพาราโบลาดัดกลับเช่นในคานต่อเนื่องช่วงนอกและช่วงใน Khaxhaturian ได้วิเคราะห์รูปแบบของฟังก์ชั่นในเทอมของสัมประสิทธิ์ต่าง ๆ ซึ่งกำหนดรูปร่างของกราฟและแสดงไว้ในตารางที่ 7-4 นอกจากนี้อาศัยสมการที่ A ซึ่งจะให้ลักษณะของไดอะแกรมของแรงดันด้านข้างบนช่วงคาน และจากหลักการของกลศาสตร์โครงสร้าง เราสามารถเขียนฟังก์ชั่นของโมเมนต์ปลายยึดได้ดังในตารางที่ 7-2 เมื่อทราบค่าโมเมนต์ปลายยึดและแฟกเตอร์ การแจกแจงโมเมนต์ก็สามารถทำการแจกแจงโมเมนต์ปลายยึดเพื่อวิเคราะห์หาโมเมนต์ดัดสุทธิได้ต่อไป

การวิเคราะห์หน่วยแรงในคานต่อเนื่อง

ผู้ออกแบบจะต้องตรวจสอบหน่วยแรงขณะปล่อยแรงและขณะใช้งานให้มีค่าไม่เกินหน่วยแรงปลอดภัย โดยจะต้องตรวจสอบหน้าตัดภายในช่วง และหน้าตัดที่จุดรองรับภายในซึ่งมีโมเมนต์บวกหรือโมเมนต์ลบที่มากที่สุดกระทำตามลำดับ เนื่องจากการแจกแจงของหน่วยแรงเป็นเส้นตรงจนถึงจุดที่คานเริ่มแตกร้าว ดังนั้นการวิเคราะห์หน่วยแรงจึงใช้ทฤษฎีอิลาสติคและหลักการซุปเปอร์โพซิชั่น โดยจะต้องคำนึงถึงวิธี และขั้นตอนการก่อสร้างและลักษณะของแรงภายนอกต่าง ๆ ที่อาจเกิดขึ้นพร้อมกัน สำหรับโมเมนต์เนื่องจากแรงอัดคานที่ใช้ในการคำนวนหน่วยแรงเป็นโมเมนต์ดัดสุทธิ ซึ่งเท่ากับผลคูณของแรงอัดคานกับระยะเยื้องศูนย์ของแรงอัดลัพธ์ที่หน้าตัดที่พิจารณา ดังนั้นเราอาจเขียนสมการของหน่วยแรงที่ขอบหน้าตัด ซึ่งโมเมนต์ดัดภายนอกมีเครื่องหมายบวกได้ดังนี้

ขณะปล่อยแรง

$$f_t = \frac{P}{A} - \frac{Pe^*}{S1} + \frac{M_g}{S1}$$

$$f_b = \frac{P}{A} + \frac{Pe^*}{S2} - \frac{M_g}{S2}$$

ขณะใช้งาน

$$f_t = \frac{\eta P}{A} - \frac{\eta Pe^*}{S1} + \frac{M_g}{S1} + \left(\frac{M_d + M_l}{S1}\right)$$

$$f_b = \frac{\eta P}{A} + \frac{\eta Pe^*}{S2} - \frac{M_g}{S2} - \left(\frac{M_d + M_l}{S2}\right)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำลังประลัยและการถ่ายเทโมเมนต์ดัด

ในการออกแบบคานต่อเนื่องเราจะต้องตรวจสอบกำลังดัดประลัยของหน้าตัดต่าง ๆ เช่นเดียวกับคานช่วงเดี่ยวธรรมดา โดยกำลังดัดประลัยของแรงคู่ควบภายในจะต้องมากกว่าหรือเท่ากับโมเมนต์ดัดประลัยที่กระทำต่อหน้าตัด ขณะที่หน้าตัดพัฒนากำลังประลัยความเครียดที่ขอบหน้าตัดมีค่าเท่ากับ 0.003 และการแจกแจงความเครียดในส่วนที่รับแรงอัดไม่เป็นเส้นตรงมาตรฐาน ACI ใช้ผลการศึกษาของ Whitney ให้การแจกแจงหน่วยแรงอัดมีค่าคงที่เท่ากับ $0.85f_c'$ และมีความลึกเท่ากับ a แรงอัดที่สภาวะประลัยมีค่าเท่ากับแรงดึง ซึ่งในกรณีที่หน้าตัดเสริมด้วยเหล็กกำลังสูงและเหล็กเสริมธรรมดา สมการสมดุลย์ของแรงภายในจะเท่ากับ

$$0.85 f_c' A_C = A_{ps} f_{ps} + A_S f_y$$

โดยที่ Ac = พื้นที่ของหน่วยแรงอัดในกรณีหน้าตัดรูปสี่เหลี่ยมมีค่าเท่ากับผลคูณของความกว้างของคานกับระยะ a

Aps = พื้นที่ของเหล็กเสริมกำลังสูง

As = พื้นที่ของเหล็กเสริมธรรมดา

fps = หน่วยแรงดึงในคอนกรีตที่สภาวะประลัย

fy = หน่วยแรงดึงที่จุดคลากของเหล็กเสริม

โดยการคิดโมเมนต์รอบจุดศูนย์ถ่วงของแรงอัด เราสามารถคำนวนกำลังดัดประลัยของหน้าตัดได้เป็น

$$\overline{M}_u = A_{ps} f_{ps} (d - \overline{Z}) + A_S f_y (d - \overline{Z})$$

โดยที่ dp = ระยะจากผิวคานถึงจุดศูนย์ถ่วงของเหล็กกำลังสูง

d = ระยะจากผิวคานถึงจุดศูนย์ถ่วงของเหล็กธรรมดา

$\overline{Z}$ = ระยะจากผิวคานถึงจุดศูนย์ถ่วงของพื้นที่รับแรงอัด

กำลังดัดประลัยจะคูณด้วยแฟคเตอร์ลดกำลัง ϕ ซึ่งมีค่าเท่ากับ 0.9 สำหรับการดัด

เมื่อคานต่อเนื่องมีแรงประลัยกระทำ คอนกรีตและเหล็กเสริมจะมีพฤติกรรมอยู่ในช่วงอินอิลาสติคที่หน้าตัดต่าง ๆ ซึ่งมีหน่วยแรงดึงมากกว่ากำลังดึงมากกว่ากำลังดึงของคอนกรีตการวิเคราะห์หาแรงกระทำประลัยจะต้องพิจารณาการตอบสนองในช่วงอินอิลาสติค การถ่ายเทแจกแจงโมเมนต์ดัดจากหน้าตัดที่พัฒนากำลังดัดหน้าตัดแรก ไปยังหน้าตัดอื่นจนกระทั่งคานเปลี่ยนเป็นกลไกหมุนการที่คานจะมีพฤติกรรมแบบนี้จะต้องมีความหนียวมากเพียงพอ และการวิเคราะห์ที่ถูกต้องตามทฤษฎีก็มีความสลับซับซ้อนมากเพราะบริเวณที่อยู่ในช่วงพลาสติคแผ่กระจายตามความยาวของคาน การวิเคราะห์แบบขีดจำกัดสมมุติหน้าตัดที่พัฒนาพลาสติคฮิคส์เป็นจุดและส่วนอื่นๆ ของคานมีพฤติกรรมอยู่ในช่วงอิลาสติค ตัวอย่างของการวิเคราะห์แบบขีดจำกัดได้แก่พฤติกรรมของคานปลายยึดในรูปที่7-1

ลิขสิทธิ์ของมหาวิทยาลัยเชียงใหม่ การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการออกแบบวิธีกำลังประลัยตามมาตรฐาน ACI เราทำการวิเคราะห์หาโมเมนต์คัด เนื่องจากแรงกระทำประลัยโดยทฤษฎีอิลาสติค แรงกระทำประลัยเท่ากับแรงกระทำใช้งานคูณด้วยโหลดแฟคเตอร์ของแรงจรหรือแรงตายตัวแล้วแต่กรณี ในประเด็นที่คานต่อเนื่องสามารถรับแรงได้เพิ่มขึ้นเนื่องจากการถ่ายเทแจกแจงโมเมนต์คัด มาตรฐาน ACI พิจารณาให้มีการถ่ายเทแจกแจงโมเมนต์คัดได้บางส่วนสำหรับคานที่หน้าตัดสามารถหมุนตัวได้มาก คุณสมบัติประการนี้ของหน้าตัดเกี่ยวพันกับดัชนีต่อไปนี้ของเหล็กเสริม

$$\omega = \frac{\rho f_Y}{f_c'}$$

$$\omega' = \frac{\rho_p f_{ps}}{f_c'}$$

$$\omega' = \frac{\rho_p f_{ps}}{f_c'}$$

โดยที่ ρ, ρ', ρ_p เป็นอัตราส่วนเหล็กเสริมธรรมดารับแรงดึง,อัตราส่วนเหล็กเสริมธรรมดารับแรงอัดและอัตราส่วนเหล็กเสริมกำลังสูงรับแรงดึง $(\rho = A_S / bd; \rho' = A_S' / bd; \rho_p = A_{ps} / bd)$ มาตรฐาน ACI กล่าวว่าในคานซึ่งหน้าตัดมีค่าดัชนี

$$\omega + \omega_p + \omega' < 0.2$$

เราสามารถปรับค่าโมเมนต์คัดเครื่องหมายลบเนื่องจากแรงกระทำประลัยที่จุดรองรับต่าง ๆ ของคานต่อเนื่องซึ่งคำนวนโดยทฤษฎีอิลาสติคขณะที่เกิดพลาสติคฮิกส์อันแรกให้เพิ่มขึ้นหรือลดลงได้เท่ากับ

$$P = 20\left(1 - \frac{\omega + \omega_p - \omega'}{0.30}\right)\%$$

โดยค่าโมเมนต์คัดลบที่ปรับค่าแล้วนี้ จะต้องใช้ในการคำนวนค่าโมเมนต์คัดหน้าตัดอื่น ๆ ในคานซึ่งมีแรงกระทำประลัยอันเดียวกัน นอกจากข้อจำกัดในสมการที่ คานจะต้องเสริมเหล็กธรรมดามีปริมาณมากพอเพื่อควบคุมการกระจายของรอยร้าวเนื่องจากการคัดที่สภาวะประลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการวิเคราะห์โมเมนต์ดัดที่สภาวะประลัย มีสิ่งหนึ่งที่ควรกล่าวถึงคือโมเมนต์รองเนื่องจากแรงอัดคาน เนื่องจากเราใช้ทฤษฎีอิลาสติคในการคำนวนหาโมเมนต์ดัดเนื่องจากแรงกระทำภายนอก ในกรณีที่เหล็กเสริมไม่เป็นแนวกอนคอร์แดนต์ คานจะมีโมเมนต์รองเกิดขึ้น สำหรับพฤติกรรมในช่วงอิลาสติค แรงอัดคานเพิ่มขึ้นน้อยมากเนื่องจากโมเมนต์ดัดดังนั้นเราใช้แรงอัดคานประสิทธิผลในการวิเคราห์หาโมเมนต์รอง นั่นคือใช้โหลดแฟคเตอร์ของแรงอัดคานเป็น 1.0 ถึงแม้ว่า โหลดแฟกเตอร์สำหรับแรงกระทำภายนอกจะเป็น1.4 สำหรับแรงตายตัว และ 1.7 สำหรับแรงกระทำจร โดยสรุปสำหรับตรวจสอบกำลังดัดประลัยเราสามารถเขียนเป็นสมการได้ดังนี้

$$\phi \bar{M}_u > 1.4(M_g + M_d) + 1.7M_l + \text{sec}\,ondarymoment$$

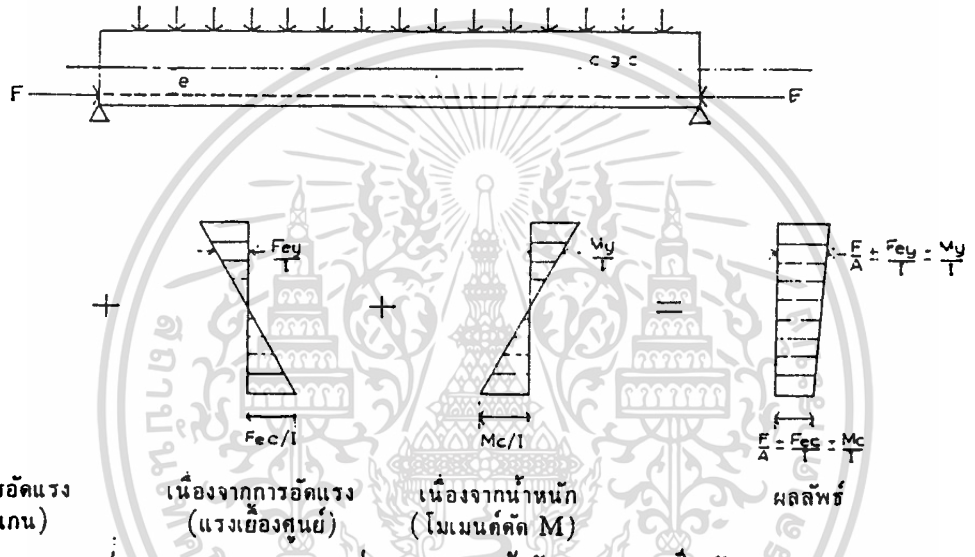


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมเมนต์ดัดแตกร้าว

โมเมนต์ดัดที่เริ่มกระทำให้เกิดรอยแตกร้าวในคานคอนกรีตอัดแรงจะเรียกว่า โมเมนต์ดัดแตกร้าว (Cracking Moment) รอยแตกร้าวนี้จะเกิดขึ้นเมื่อหน่วยแรงดึงที่ผิวนอกสุดของคอนกรีตมีค่าถึงโมดูลัสของการแตกร้าว (Modulus of Rupture) คือ $1.99\sqrt{f_c'}$

พิจารณาคานคอนกรีตอัดแรงรับน้ำหนักเหมือนในรูปที่ 7.10 ถ้าหากน้ำหนักที่กระทำเป็นน้ำหนักที่ทำให้เกิดการแตกร้าว (Cracking Load) หรือคานรับโมเมนต์ดัดแตกร้าว Mcr หน่วยแรงดึงที่เกิดขึ้นจะมีค่าเท่ากับ ตามรูปที่ 7.9



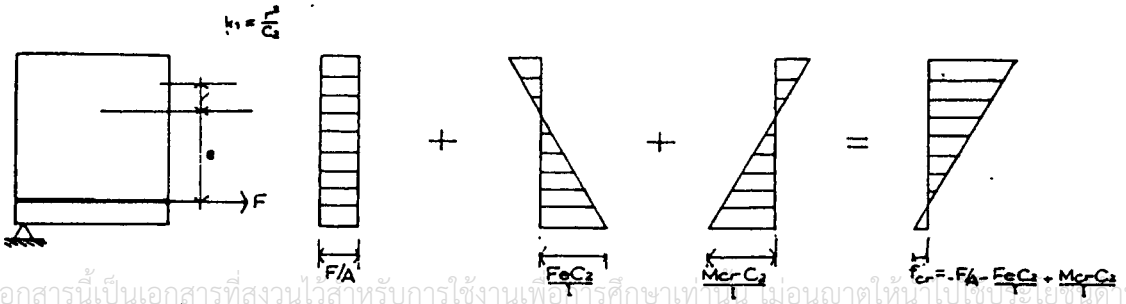
รูปที่ 7.9 การกระจายหน่วยแรงบนหน้าตัดคานคอนกรีตอัดแรง

จะได้

$$-\frac{F}{A}-\frac{Fec2}{I}+\frac{M_{cr}c2}{I}=f_{cr}$$

$$\frac{M_{cr}c2}{I}=\frac{Fec2}{I}+\frac{F}{A}+f_{cr}$$

$$M_{cr}=Fe+\frac{FI}{Ac2}+f_{cr}\frac{I}{c2}$$



รูปที่ 7.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

การเขียนและการใช้โปรแกรม

หลักการของการเขียนและใช้งานรวมทั้งสภาพแวดล้อมของ VISUAL BASIC

หลังจากกระบวนการติดตั้ง Visual BASIC แล้วจะมีกรุ๊ปวินโดว์ของ Visual BASIC 3.0 สำหรับการเรียก Visual BASIC มาทำงาน และไอคอนสำหรับแสดงข้อมูล ข่าวสารต่าง ๆ เช่น Visual Design Guide เป็น Help สำหรับแนะนำการออกแบบส่วนติดต่อกับผู้ใช้ หรือ User INTERFACE , Help ของ Visual BASIC ที่ทำให้ผู้ใช้อ่าน Help ได้โดยไม่ต้องรัน Visual BASIC ไว้มากมาย มีทั้งคำถามคำตอบซึ่งอาจจะเป็นปัญหาที่ผู้ใช้พบและต้องการแก้ไข รวมทั้ง Custom Control Help ที่เป็น Help เฉพาะของ Custom Control ที่มีใช้ โดยอธิบายถึงคุณสมบัติและความสามารถของคอนโทรนแต่ละตัวอย่างละเอียด

ก่อนที่จะกล่าวถึงสภาพแวดล้อมในการพัฒนาของ Visual BASIC จะขอกล่าวถึงหลักการของการโปรแกรมเชิงภาพ หรือ Visual Programming เสียก่อนว่ามีลักษณะอย่างไร แตกต่างจากการเขียนโปรแกรมปกติและเกี่ยวข้องกับการทำงาน แบบ Even Driven ซึ่งเป็นรูปแบบการทำงานของ Microsroft Windows อย่างไร

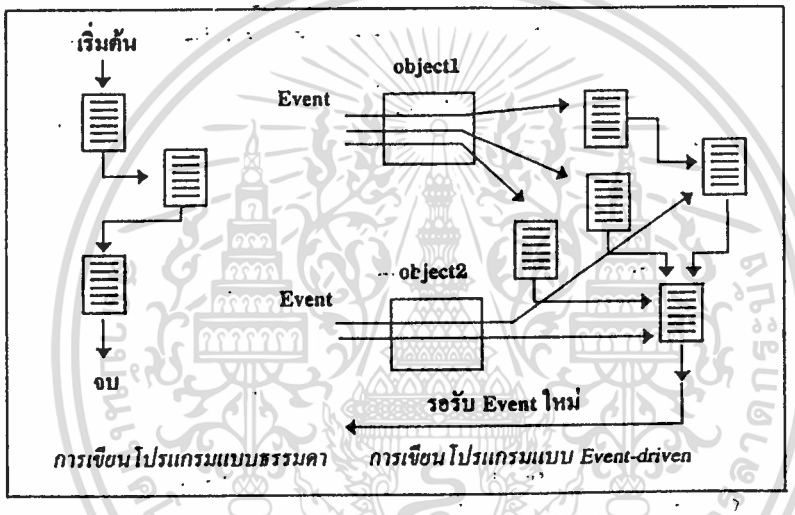
หลักการโปรแกรมเชิงภาพของ Visual BASIC

ใน Visual BASIC นั้นการพัฒนาและการเขียนโปรแกรมจะเป็นไปในอีกรูปแบบหนึ่งกล่าวคือ ในการเขียนโปรแกรมแบบเดิมนั้นต้องมานั่งออกแบบหน้าจอ ระบุตำแหน่งการแสดงผลคิดหาขั้นตอนการทำงาน และอื่น ๆจากนั้นจึงทำการเขียนโปรแกรม โปรแกรมที่ได้จะอธิบายและสั่งงานคอมพิวเตอร์เป็นลำดับไป แต่ใน Visual BASIC จะใช้หลักของภาพและการมองเห็น โดยเริ่มจากออกแบบวินโดว์หรือที่ใน Visual BASIC เรียกว่า ฟอร์ม ในฟอร์มประกอบด้วยสิ่งต่าง ๆ ที่เราจะทำงานด้วย หรือเรียกว่าเป็น Object เช่นข้อความ Scroll Bar หรือปุ่ม (Button) เมื่อกำหนดสิ่งเหล่านี้ครบตามความต้องการแล้วจึงระบุว่าองค์ประกอบแต่ละอย่างจะทำงานอย่างไร โดยเขียนโปรแกรมย่อย ๆ ปะเข้าไปกับObject เหล่านั้น ที่ต้องทำแบบนี้ก็เพราะว่าการทำงานใน วินโดว์เป็นแบบที่เรียกว่า Event-Driven คิดขึ้นกับเหตุการณ์ การเขียนโปรแกรมแบบเดิมคือ สั่งการตามลำดับจะยุ่งยากมาก หรือบางกรณีอาจทำไม่ได้เลย เพราะอย่าลืมว่าในขณะใดขณะหนึ่งนั้น ในระบบไม่ใช่จะมีเพียงแอพพลิเคชั่นที่ทำงานในขณะนั้นทั้งหมดไปพร้อม ๆ กัน เนื่องจากเป็นระบบแบบ Multitesking หรือรันหลาย ๆ งานได้ในเวลาเดียวกันในขณะที่โปรแกรมแสดงหน้าจอสำหรับรับ อินพุท ผู้ใช้อาจพิมพ์ข้อมูลเข้าไปแต่โดยดี หรืออาจอยู่ไม่สุข ใช้เมาส์เลื่อนไปคลิคตรงโน้นตรงนี้ได้โดยอิสระ ทำให้ยากที่จะเขียนโปรแกรมธรรมดา ให้คอยดักเส้นทางการทำงาน ในการรับอินพุทว่าจะเกิดอะไรขึ้นตรงไหนได้จึงใช้รูปแบบการโปรแกรมในลักษณะ Event-Driven ดังกล่าวซึ่ง Object แต่ละตัวก็จะมีเหตุการณ์เกิดขึ้นกับมันได้หลายอย่าง ถ้า

เอกสารนี้สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสนใจเฉพาะเหตุการณ์ใดก็เขียนโปรแกรมสั่งงานให้คอยดักหรือทำงานเฉพาะเหตุการณ์นั้น ๆ ได้ เช่นถ้าสิ่งที่เราสนใจเป็นปุ่มควบคุมและเราต้องการให้ทำงานเมื่อคลิ๊กหรือดับเบิลคลิ๊กก็ระบุว่าหากมีการคลิ๊กที่ปุ่มควบคุมนี้โปรแกรมจะต้องทำอย่างไร หรือถ้ามีการดับเบิลคลิ๊กจะทำอย่างไร ส่วนเหตุการณ์อื่น ๆ ที่ไม่ได้ระบุจะไม่มีผลต่อObject นั้น

จากแนวทางนี้จะเห็นว่าลักษณะเป็นธรรมชาติมากคือกำหนดหน้าตาของจอภาพหรือส่วนติดต่อกับผู้ใช้และระบุว่าถ้าเกิดเหตุการณ์อย่างนี้กับสิ่งนี้จะต้องทำอย่างไร ซึ่งโปรแกรมที่เราจะเขียนกันนี้ก็คือส่วนที่จะบอกว่าจะต้องทำอย่างไรนั่นเอง



รูปที่ 8-1 การเขียนโปรแกรมแบบธรรมดา กับแบบEvent-Driven

นอกจาก Object จะมีการตอบสนองต่อเหตุการณ์ต่าง ๆ มีกำหนดแล้ว ยังมีอีกเรื่องที่จะขอกล่าวคือทุกObject จะมีลักษณะหรือคุณสมบัติของตัวเองเช่น ช่องรับข้อความจะมีชื่อ,ข้อความในนั้น,ความสูง,สีโดยเราสามารถอ้างถึงหรือเปลี่ยนคุณสมบัติเหล่านี้ได้ขณะที่โปรแกรมทำงานอยู่ เป็นต้นว่าหากไม่มีการป้อนข้อมูลจะแสดงด้วยสีหนึ่งหรืออาจไม่ต้องแสดงบนจอภาพเลย

โดยสรุปแล้วรูปแบบของหลักการใน Visual BASIC ก็คือเริ่มจากออกแบบจอภาพ และเขียนโปรแกรมสำหรับแต่ละ Event ปะเข้าไปยัง Object ต่าง ๆให้ทำงานตามเหตุการณ์ที่เกิดขึ้น โดยทุก Object จะมีคุณสมบัติเฉพาะที่สามารถเปลี่ยนแปลงได้ของตัวเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

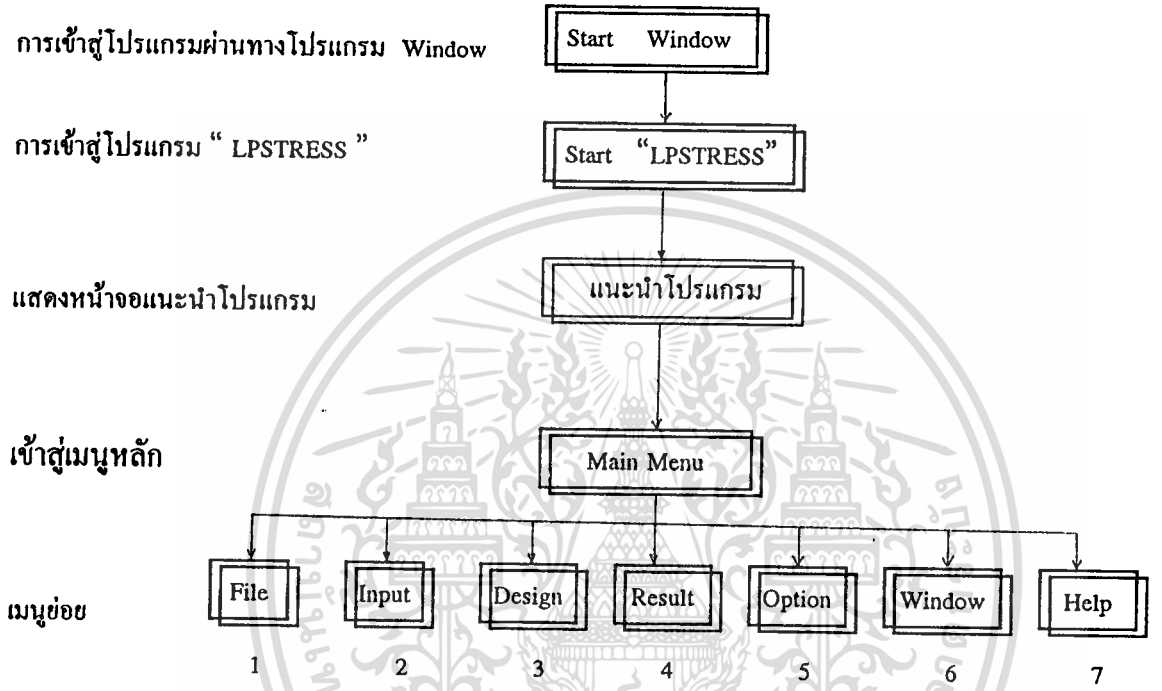
ตอนที่ 2

ขั้นตอนการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นตอนการใช้งานของโปรแกรม " LPSTRESS"

โปรแกรม "LPSTRESS" เป็นโปรแกรมที่สร้างขึ้นให้ใช้งานได้ง่าย โดยบนหน้าจอของเครื่องคอมพิวเตอร์จะปรากฏทั้งหน้าจอในส่วนของการใส่ข้อมูลและหน้าจอในส่วนของการดูผลลัพธ์ในเวลาเดียวกัน โดยไม่ต้องเปลี่ยนหน้าจอเรื่อย ๆไปที่ละหน้าจอจนทำให้ลืมหรือการทำงานไม่สะดวก ฉะนั้นผู้ใช้จึงทำงานได้ภายในจอภาพเดียว ซึ่งจะเป็นไปในลักษณะของโปรแกรมที่อยู่บน Windows ทั้งหลายนั่นเอง



1. เมนูFile



การเปิดโครงงานใหม่



การเปิดโครงงานเดิม



การบันทึกทันที



การบันทึกโดยกำหนดชื่อFile



การพิมพ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใส่ข้อมูลต่าง ๆ ที่ไม่มีผลต่อการออกแบบ → Information

3.เริ่มทำการ Execute → Design

เลือกเพื่อให้เริ่ม Execute → Prestress

4.แสดงผลการออกแบบ → Result

การแสดงผลการวางลวด → Prestress → Profile Tendon

แสดงรายการการวางลวด → List Profile

แสดงผลทางรายการคำนวณ → List Computation

แสดงผลรูปร่างของโครงงาน → Geometric

แสดงผล Shear Diagram → Shear Diagram

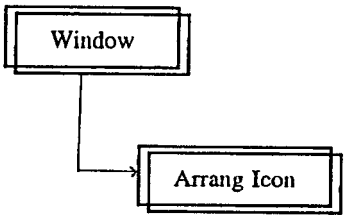
แสดงผล Moment Diagram → Moment Diagram

5.เมนู Option → Option

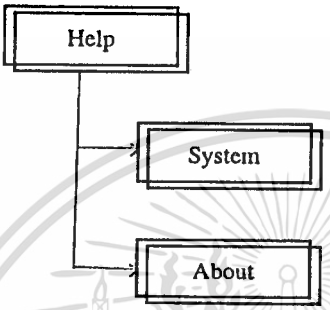
การเก็บค่าการแสดงหน้าจอไว้ → Save Setting

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.เมนู Window



7.เมนู Help



แสดงทรัพยากรของเครื่อง

แสดงหน้าจอแนะนำโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 3

คู่มือการใช้โปรแกรมสำหรับออกแบบคอนกรีตอัดแรง “ LPSTRESS”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้โปรแกรม LPSTRESS

(Manual of LPSTRESS)



โดย

นายสมพงศ์ คงกระพันธ์ 35104443

นายสยาม กันเกตุ 35104449

คู่มือการใช้โปรแกรมฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมการก่อสร้าง

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือประกอบการใช้โปรแกรม LPSTRESS

(Manual of LPSTRESS)

บทนำ

โปรแกรม LPSTRESS เป็นซอฟแวร์สำเร็จรูปสำหรับคำนวณและออกแบบแผ่นพื้นคอนกรีตอัดแรง โดยพัฒนาโปรแกรมจากภาษา Visual BASIC Version 3.0 แล้วทำการ Compile เป็น Execute file (ไฟล์ที่มีนามสกุลเป็น exe .exe) ซึ่งสามารถทำงานได้เหมือน Execute file ทั่วไป โดยไม่จำเป็นต้องใช้โปรแกรม Visual BASIC

อย่างไรก็ตาม Execute file ที่พัฒนาจาก Visual BASIC จะต้องใช้ windows ในการทำงานดังนั้น เครื่องคอมพิวเตอร์ที่สามารถจะใช้โปรแกรมนี้ได้ จะต้องมีโปรแกรม windows ด้วย (ซึ่งเครื่องโดยทั่วไปจะมีโปรแกรม windows อยู่แล้ว) ในกรณีที่ต้องการปรับปรุงโปรแกรม จะต้องใช้ source code และใช้ Visual BASIC ตั้งแต่ Version 3.0 ขึ้นไปในการพัฒนาโปรแกรม

การออกแบบโปรแกรม LPSTRESS นี้เน้นการใช้งานง่ายแบบโปรแกรม windows ทั่วไป โปรแกรมเน้นความถูกต้องในการคำนวณ ผู้พัฒนาโปรแกรมได้พัฒนาโปรแกรมได้ทำการออกแบบโปรแกรม LPSTRESS ให้สามารถพิมพ์ผลการคำนวณ (summary) และรายละเอียดในการคำนวณ (Detail) ได้ เพื่อความสะดวกของผู้ใช้ทั่วไปในการออกแบบในประเทศไทย คือหน่วยเมตริก (Metric Units)

การใช้โปรแกรมในการออกแบบแผ่นพื้นคอนกรีตอัดแรงนั้นผู้ใช้โปรแกรมควรมีความรู้ในการวิเคราะห์และออกแบบแผ่นพื้น จะทำให้สามารถใช้โปรแกรมได้ง่ายและเข้าใจโครงสร้างของโปรแกรมได้ดีขึ้น ฉะนั้นผู้ใช้โปรแกรมจึงควรที่จะศึกษาการออกแบบควบคู่ไปกับการเรียนรู้การใช้โปรแกรมจากคู่มือการใช้งานเล่มนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

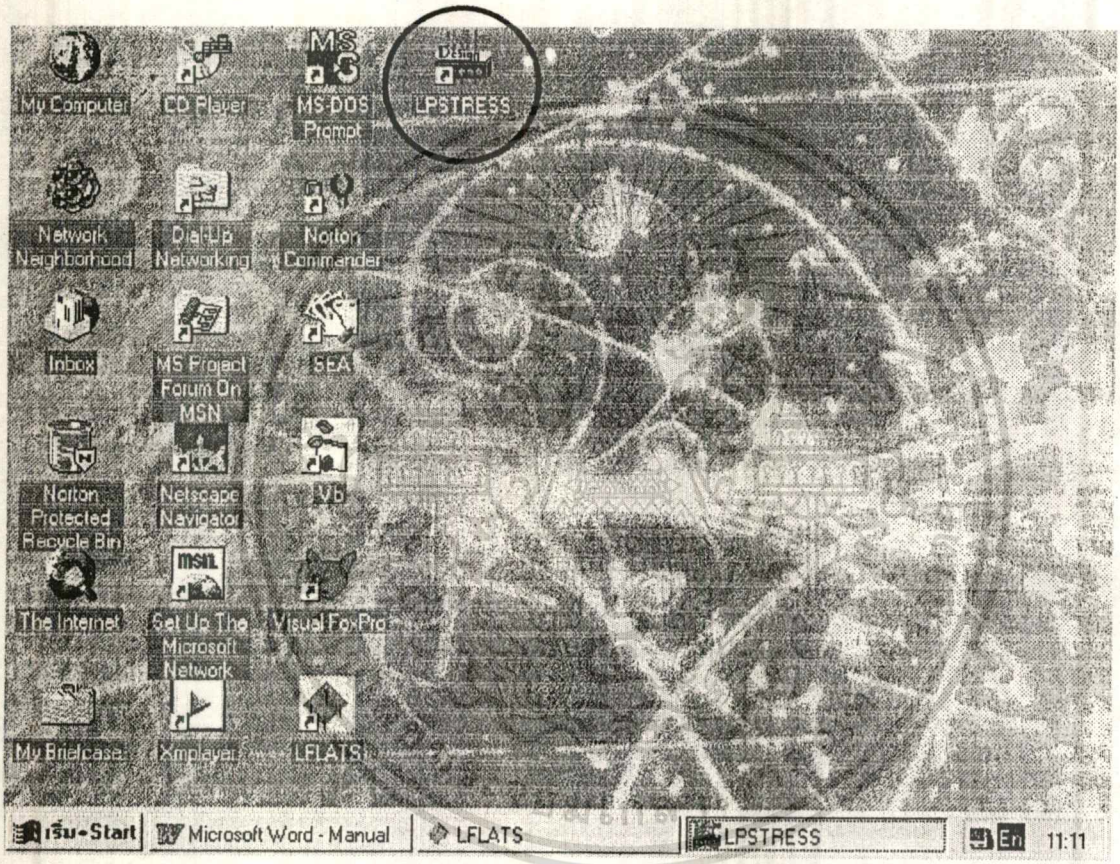
1. การเข้าสู่โปรแกรม

เนื่องจากโปรแกรม LPSTRESS เป็นโปรแกรมที่ run บนโปรแกรม windows ฉะนั้นการเข้าสู่โปรแกรมจึงต้องผ่าน windows

c:\win

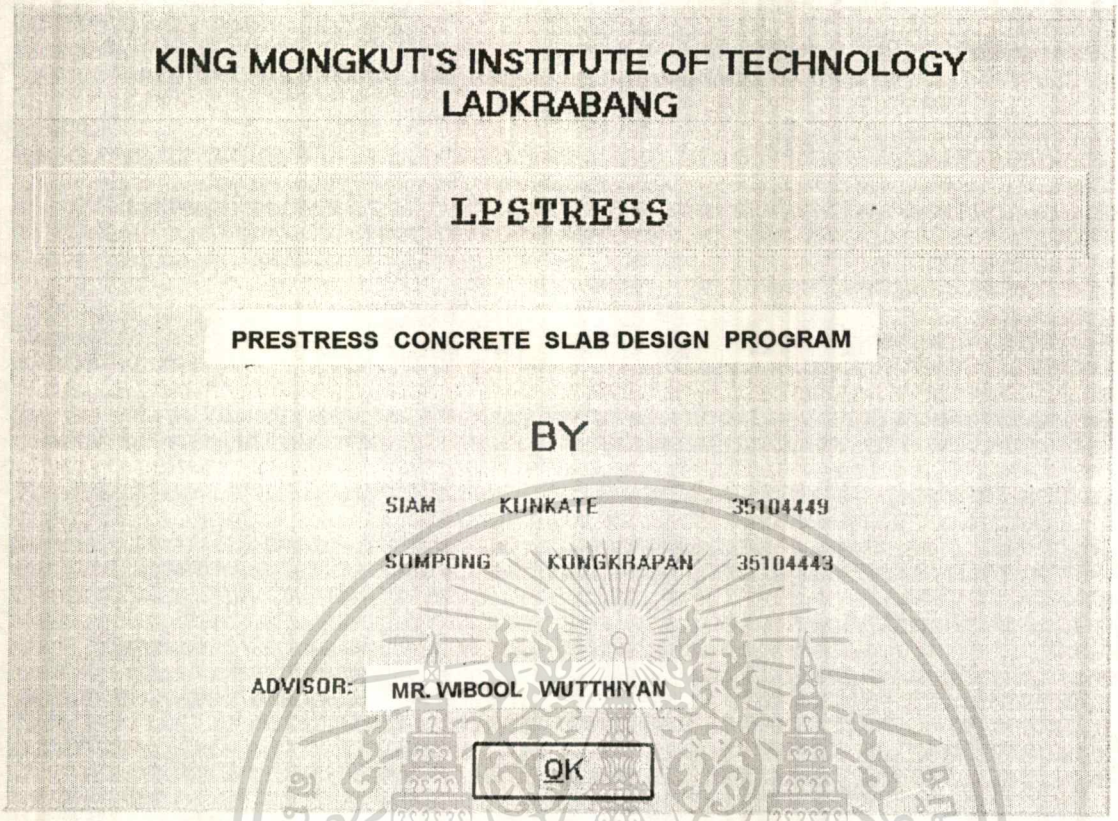
การเข้าสู่โปรแกรมสามารถเข้าได้โดยใช้การใช้ Icon ที่ได้

ก่อนเข้าสู่โปรแกรมจะมี Icon ของโปรแกรม ปรากฏดังรูป



รูปที่1 แสดงไอคอนของโปรแกรม LPSTRESS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 เป็นหน้าจอที่แสดงการแนะนำโปรแกรมการออกแบบเมื่อเริ่มต้นใช้โปรแกรม

อุปกรณ์ที่ใช้และซอฟแวร์ที่ต้องใช้ควบคู่กับโปรแกรม LPSTRESS

1.ไมโครคอมพิวเตอร์ควรจะเป็น 486 DX2-66 ขึ้นไปและควรมี RAM อย่างน้อย 4 M byte เพื่อความรวดเร็วในการคำนวณ แนะนำให้ใช้ 8 M byte

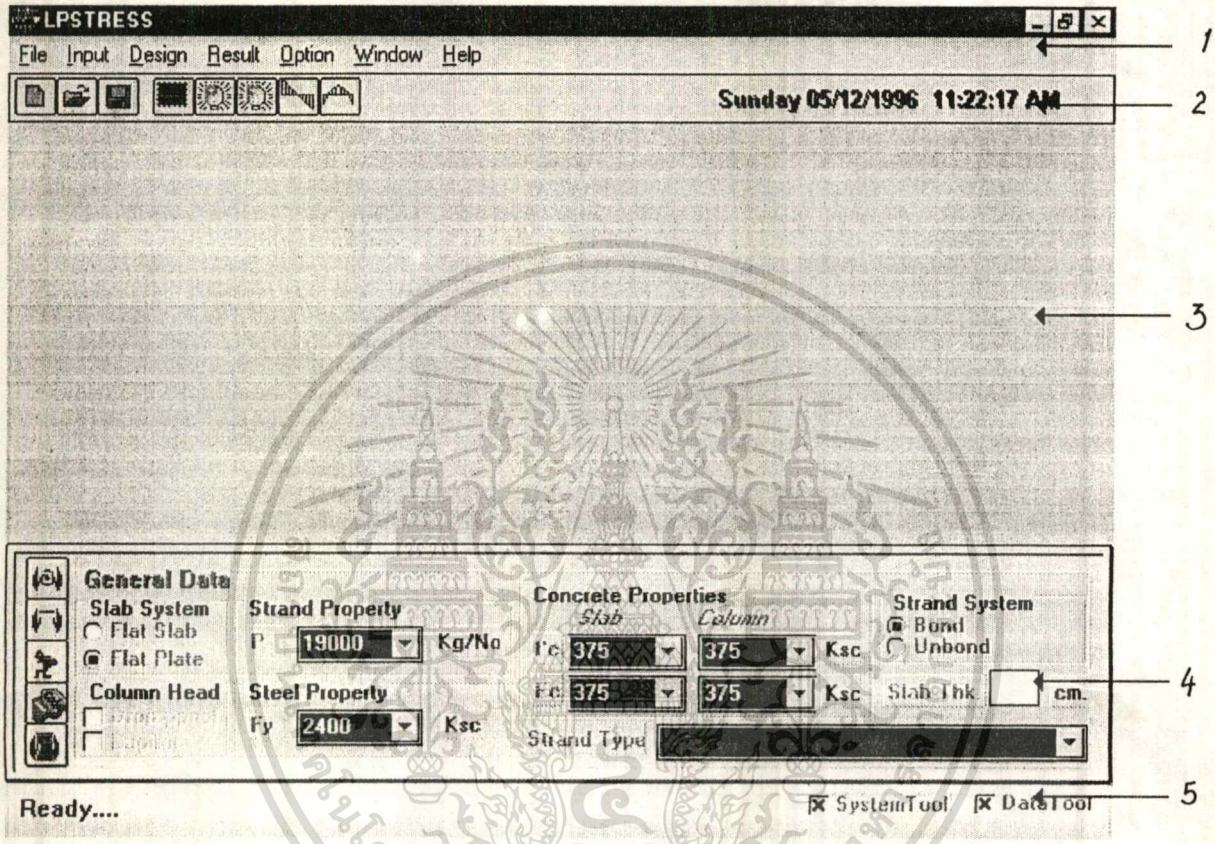
2. Mouse
3. จอสี (โปรแกรมนี้ออกแบบมาสำหรับจอสี)
4. โปรแกรม Windows ควรเป็นรุ่น 3.11 ขึ้นไป
5. เครื่องพิมพ์

ลักษณะโดยรวมของหน้าจอแบ่งเป็น 5 ส่วนใหญ่ ๆ ดังนี้คือ

1. Pull Down Menu ประกอบด้วยเมนู 7 เมนู แต่ละเมนูจะประกอบด้วยเมนูย่อย ๆ ซึ่งจะอธิบายในภายหลัง
2. Icon Menu เป็นเมนูที่เป็นภาพที่เป็นภาพทำให้โปรแกรมใช้ได้สะดวกขึ้น
3. หน้าจอแสดง Output จะแสดงผลการคำนวณได้แก่ โมเมนต์ แรงเฉือน และอื่น ๆ ซึ่งจะแสดงเป็นกราฟิค หรือเป็นรายการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. หน้าจอแสดงส่วนที่ต้องใส่ข้อมูล ซึ่งจะมีอยู่หลายหน้าจอ สามารถจะเลือกโดยใช้ Icon ที่เป็นรูปภาพที่สื่อความหมายทำให้โปรแกรมใช้ได้สะดวกขึ้น
5. Status Bar เป็นส่วนที่อำนวยความสะดวกในการบอกสถานะของโปรแกรมในขณะนั้น



รูปที่ 3 แสดงส่วนต่าง ๆ บนหน้าจอของตัวโปรแกรม

ส่วนประกอบของโปรแกรมโดยละเอียด

1. Pull Down Menu ประกอบด้วยเมนู 7 เมนู

1.Menu file (Alt+F) เป็นเมนูที่จัดการเกี่ยวกับ file จะมีเมนูย่อย ๆ ให้เลือกจัดการกับ file ดังนี้

1.1 New (F1) เป็นการเปิด File ใหม่ทุกครั้งที่ต้องการทำโครงการใหม่

1.2 Open (F3) เป็นการเปิด file ที่มีอยู่แล้วขึ้นมาแก้ไข

1.3 Save (F2) เป็นการบันทึกข้อมูลที่กำลังทำงานอยู่โดยใช้ชื่อเดิม การบันทึกจะทำได้เมื่อ file ถูก New หรือ Open เท่านั้น

1.4 Save as เป็นการบันทึกข้อมูลที่กำลังทำงานอยู่โดยบันทึกโดยมีการตั้งชื่อใหม่ การบันทึกต้องมีการ New หรือ Open file เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 Print เป็นเมนูที่ใช้ในการพิมพ์ ประกอบด้วยเมนูย่อยดังนี้

1.5.1 Pre-stress เป็นการพิมพ์ผลของการออกแบบในส่วนของแผ่นพื้นคอนกรีตอัดแรงได้แก่

1.5.1.1 Profile strand (Ctrl+P) เป็นการพิมพ์กราฟิคแนวการวางลวดในแผ่นพื้น

1.5.1.2 List Profile (Ctrl+L) เป็นการพิมพ์รายการของแนวการวางลวดในแผ่นพื้นคอนกรีตอัดแรงซึ่งจะสัมพันธ์กับในผลของ Profile strand

1.5.1.3 List Computation (Ctrl+C)เป็นการพิมผลของการคำนวณทั้งหมดออกมาเป็นรายการคำนวณ

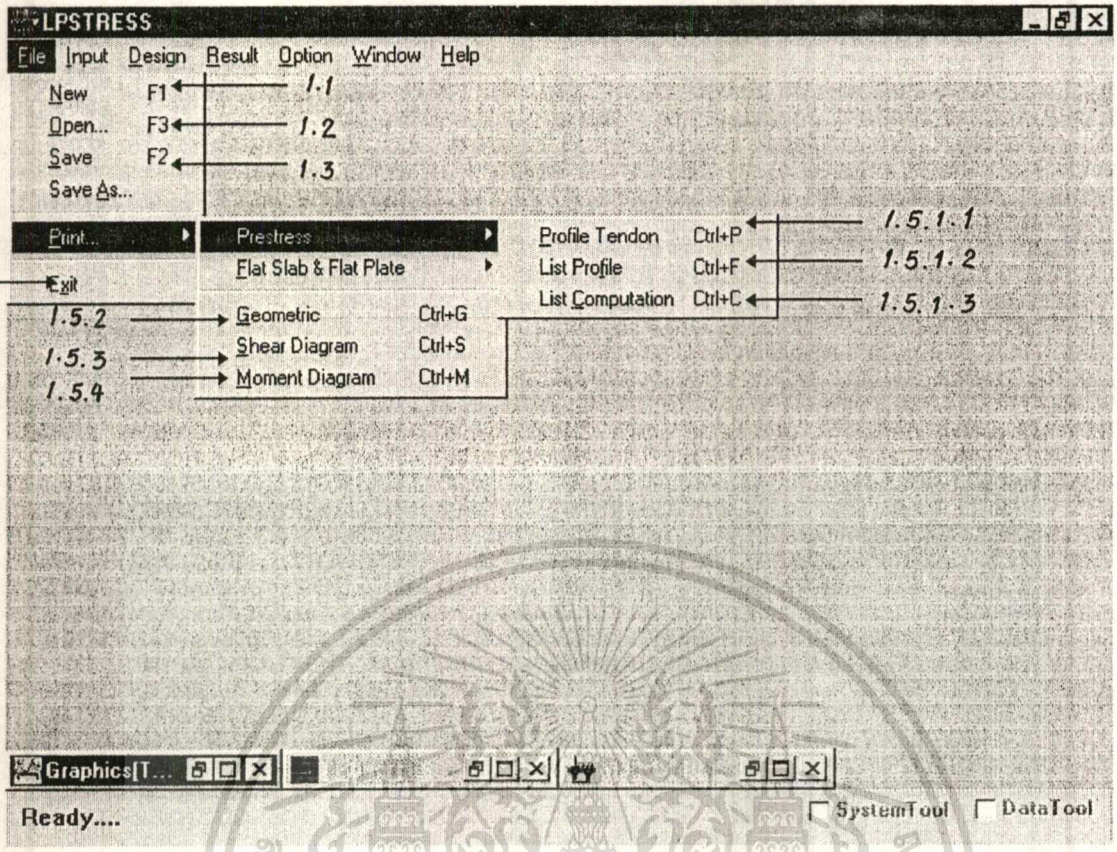
1.5.2 Geometric (Ctrl+G) เป็นการพิมพ์ผลของการคำนวณในส่วนของกราฟิคแสดงรูปร่าง ขนาด ตำแหน่งของแผ่นพื้น

1.5.3 Shear Diagram (Ctrl+S) เป็นการพิมพ์ผลการคำนวณแรงเฉือนทางด้านกราฟิค

1.5.4 Moment Diagram (Ctrl+M) เป็นการพิมพ์ผลการคำนวณโมเมนต์ทางด้านกราฟิก

1.6 Exit เพื่อออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4 แสดงหน้าจอซึ่งแสดง Pull down Menu File และเมนูย่อยต่าง ๆ

2.Menu Input (Alt+I) เป็นเมนูสำหรับใส่ข้อมูลที่ใช้ในการ Design

2.1 Node (F4) สำหรับเปิด Panel ของการใส่ข้อมูลของ คุณสมบัติของเสาที่ใช้ในการออกแบบพื้น (อธิบายใน การใส่ข้อมูล Node)

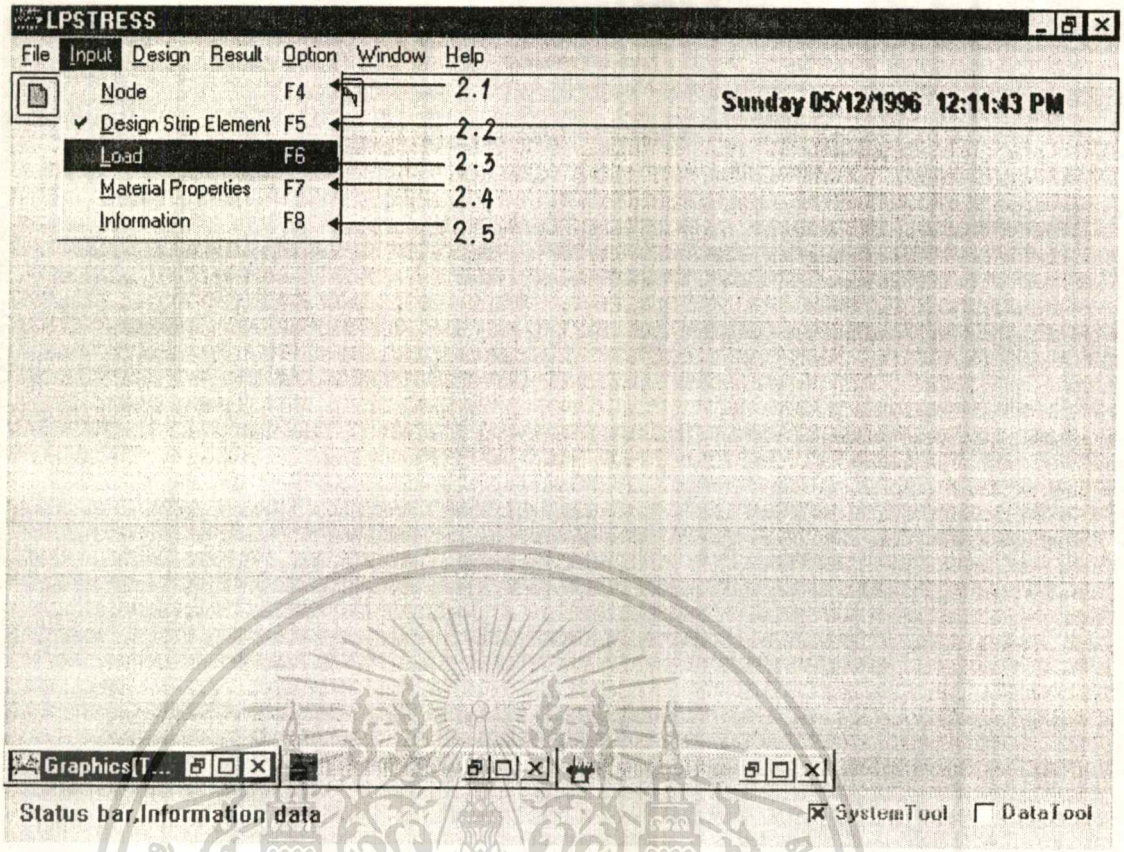
2.2 Design Strip Element (F5) สำหรับเปิด Panel แสดง Design Strip ในแนวแกน X และ แกน Y

2.3 Load (F6) สำหรับเปิด Panel สำหรับใส่ข้อมูลของ แรงกระทำกับพื้น

2.4 Material Properties (F7) สำหรับเปิด Panel สำหรับ ใส่ข้อมูลของวัสดุ

2.5 Information (F8) สำหรับเปิด Panel สำหรับ Information ซึ่งเป็นข้อมูลซึ่งไม่มีผลต่อการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



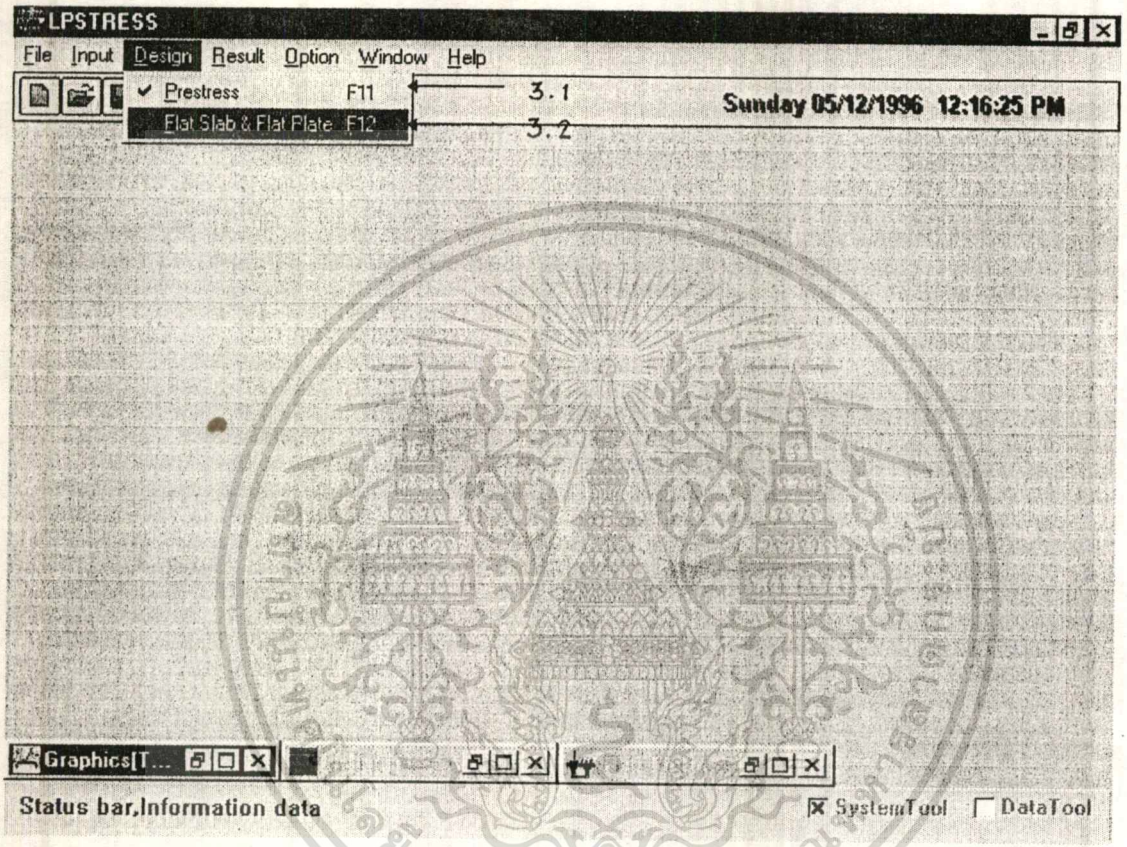
รูปที่ 5 แสดงหน้าจอซึ่งแสดง Pull down Menu Input และเมนูย่อยต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.Menu Design

3.1 Pre stress (F11) ใช้ในการออกแบบพื้นรูปแบบ Pre-stressed

3.2 Flat Slab & Flat Plate (F12) ใช้ในการออกแบบพื้นรูปแบบ Flat Slab & Flat Plate



รูปที่ 6 แสดงหน้าจอซึ่งแสดง Pull down Menu Design และเมนูย่อยต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.Menu Result

จะแสดงผลของของการ Design ในหน้าจอดังหัวข้อ 3

(แสดงรายละเอียดในจอแสดงผลอีกครั้งใน หัวข้อ 3)

4.1 Pre stress

4.1.1 Profile Strand (Ctrl+K) สำหรับแสดงลักษณะการวางลวดอัดแรง จะ Active เมื่อ Design Pre-stressed ก่อน

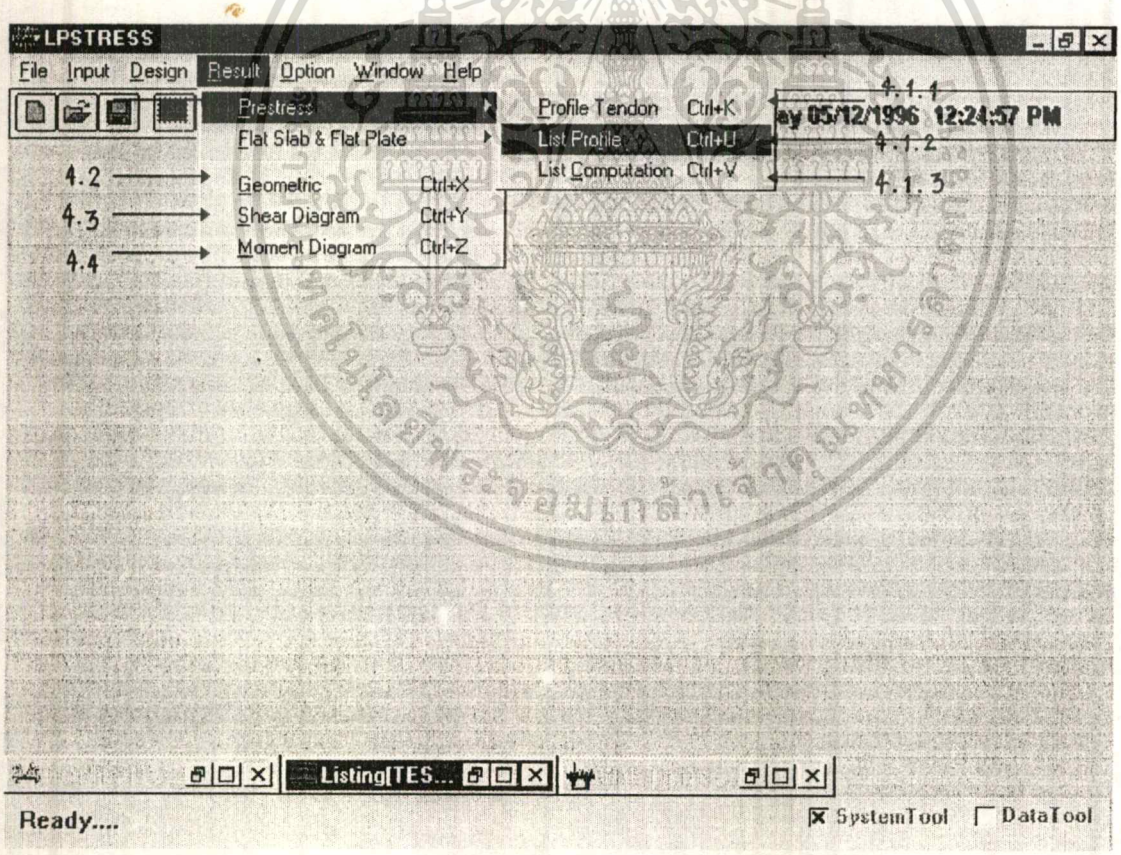
4.1.2 List Profile (Ctrl+U) สำหรับแสดงผลของระยะความสูงของการวางลวดเป็นระยะจากกลางพื้น

4.1.3 List Computation (Ctrl+V) สำหรับแสดงรายการคำนวณ

4.2 Geometric (Ctrl+X) แสดงรูป Graphics ของลักษณะของแผ่นพื้น

4.3 Shear Diagram (Ctrl+Y)

4.4 Moment Diagram (Ctrl+Z)

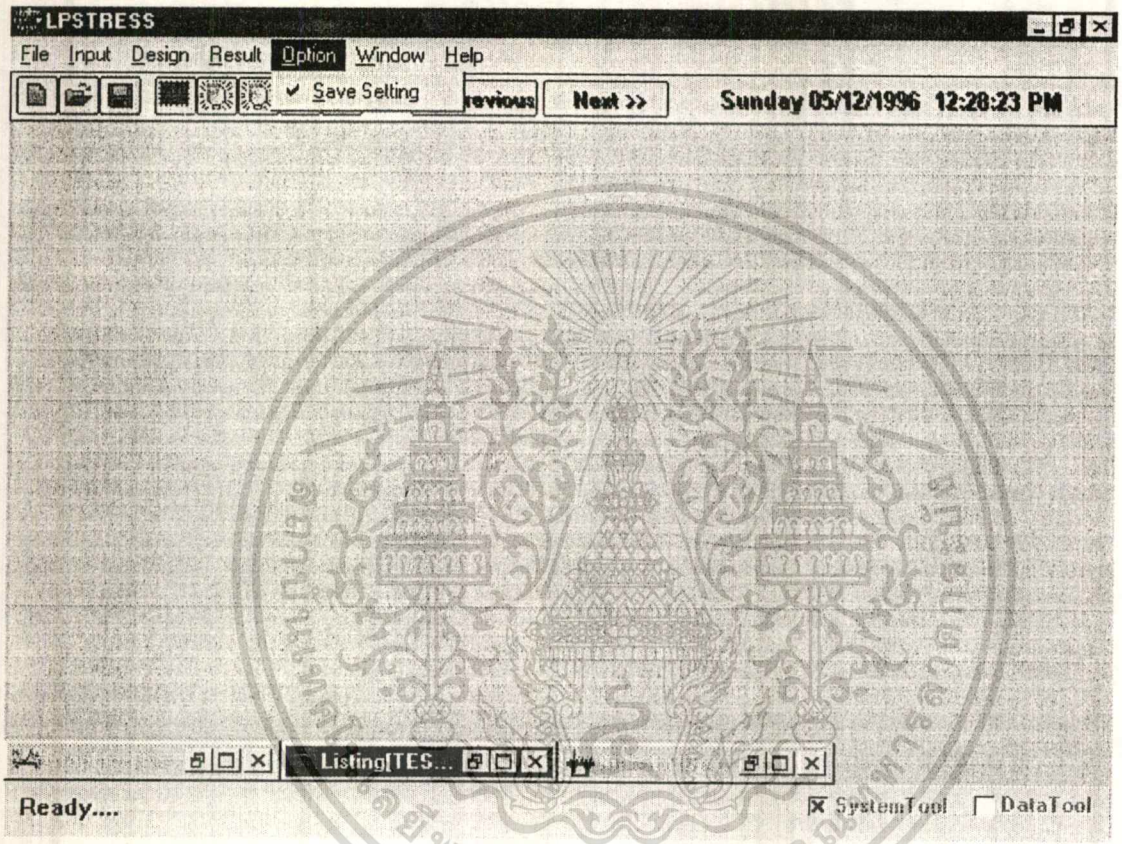


รูปที่ 7 แสดงหน้าจอซึ่งแสดง Pull down Menu Result และเมนูย่อยต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.Menu Option

5.1 Save setting เพื่อ เก็บค่าไว้ว่าแสดง Icon Menu หรือ Panel Input data ในครั้งต่อไปเพื่อแสดงหรือไม่แสดงเมื่อเข้าโปรแกรม



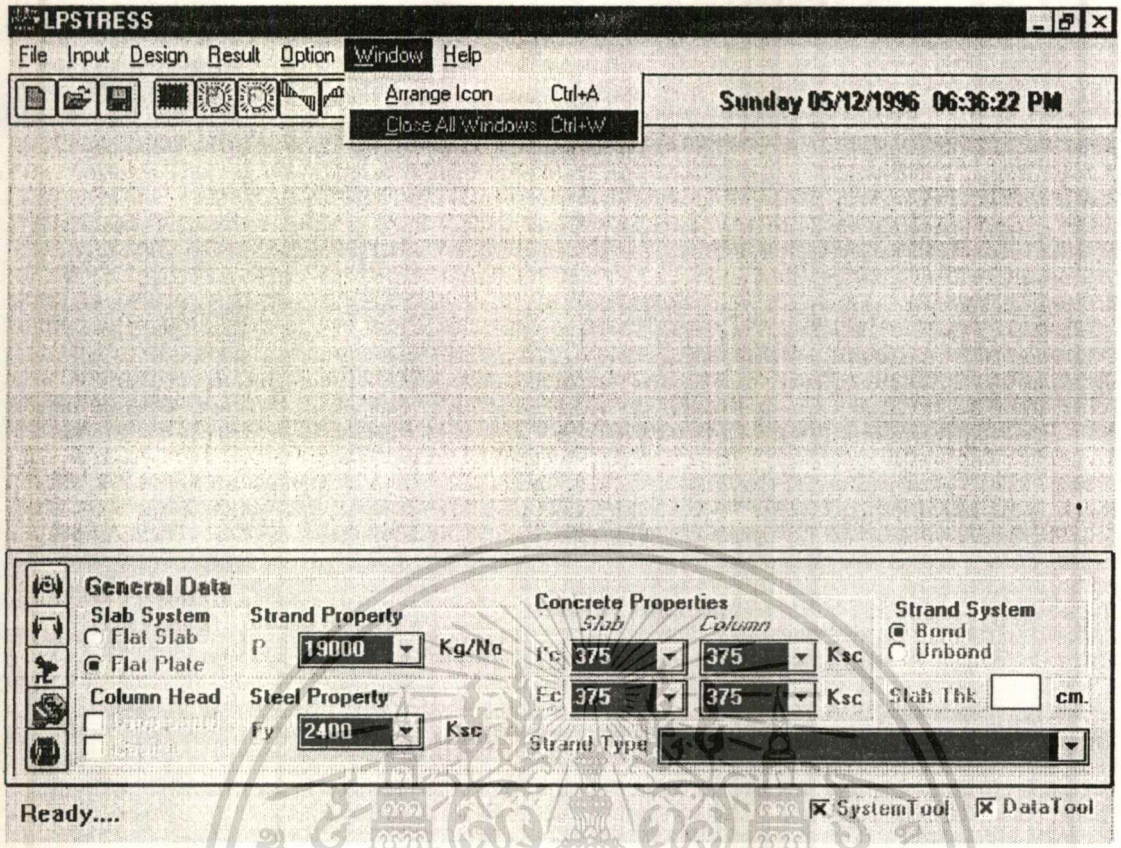
รูปที่ 8 แสดงหน้าจอซึ่งแสดง Pull down Menu Save setting และเมนูย่อยต่าง ๆ

6.Menu Window

6.1 Arrange Icon (Ctrl+A) การเรียง Icon ให้เป็นระเบียบ

6.2 Close All Windows (Ctrl+W) การปิดหน้าต่าง Output 3 หน้าต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



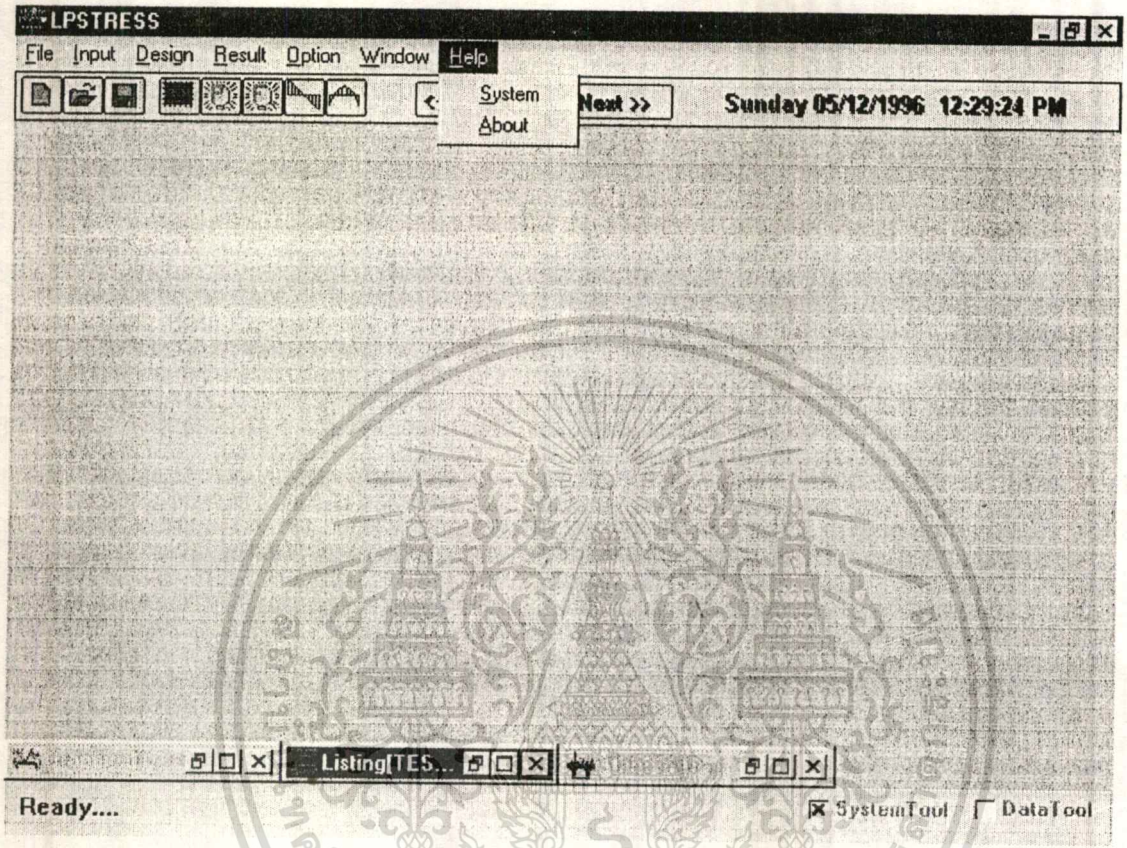
รูปที่ 9 แสดงหน้าจอซึ่งแสดง Pull down Menu Option และเมนูย่อยต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.Menu Help

7.1 System แสดงว่าทรัพยากรของเครื่องนั้นเป็นอย่างไร

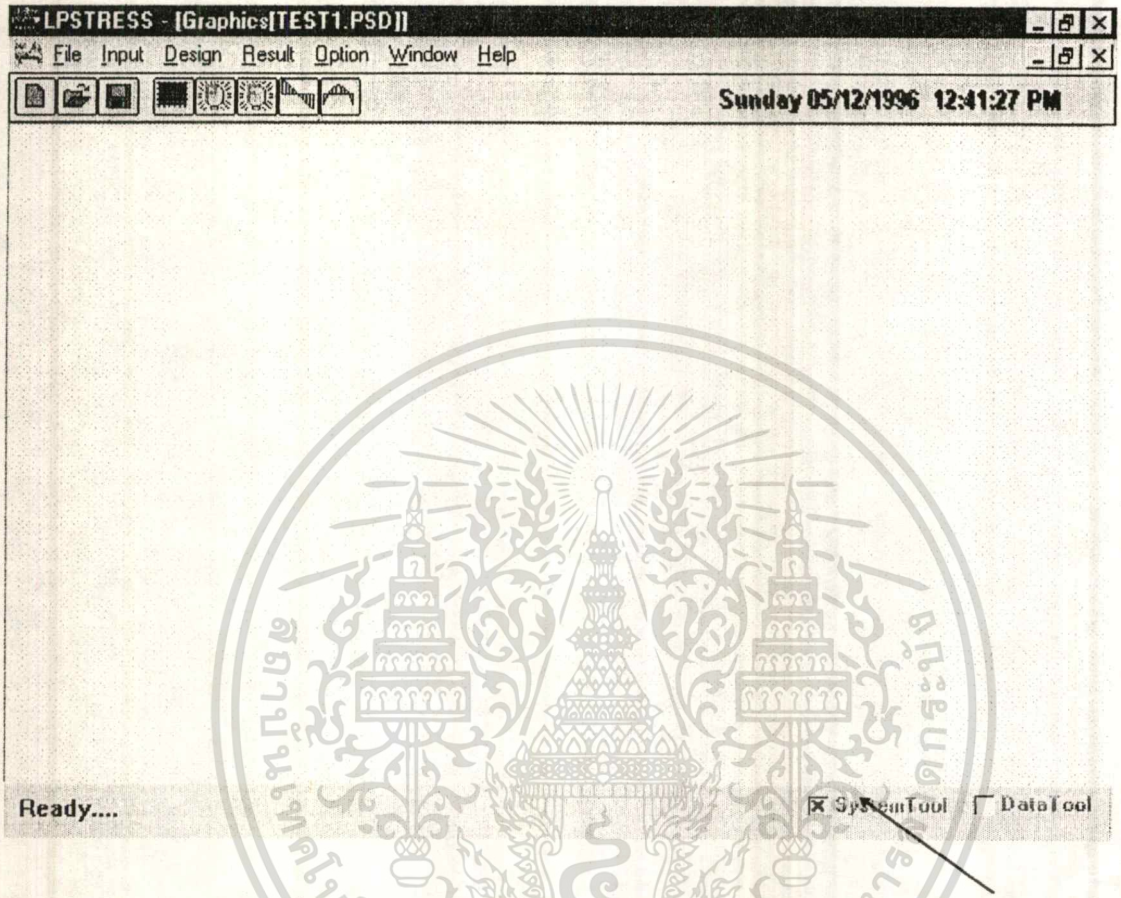
7.2 About แสดงหน้าจอเหมือนกับหน้าจอในรูปที่ 2



รูปที่ 10 แสดงหน้าจอซึ่งแสดง Pull down Menu About และเมนูย่อยต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.Icon Menu เป็นเมนูที่เป็นภาพที่เป็นภาพทำให้โปรแกรมใช้ได้สะดวกขึ้น แต่ต้องมีการ Check ที่ System Tool ใน Status bar ก่อน



รูปที่ 11 แสดงหน้าจอซึ่งแสดง การคลิก System Tool

 เป็น Icon Menu สำหรับเปิดไฟล์ใหม่ขึ้นมาทำ

 เป็น Icon Menu สำหรับเปิดไฟล์ที่มีอยู่ขึ้นมาทำการแก้ไขหรือทำงานต่อ

 เป็น Icon Menu สำหรับบันทึกข้อมูลโดยอัตโนมัติ

 เป็น Icon Menu สำหรับ แสดงรูปของพื้นที่ใช้ในการ Design แต่ต้องแน่ใจว่าใส่ข้อมูลของพื้นนั้นถูกต้อง จะ Active เมื่อมีการเปิด ไฟล์ใหม่ หรือเก่าขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

 เป็น Icon Menu สำหรับ Design Pre-stressed จะ Active เมื่อมีการเปิดไฟล์ใหม่ หรือ เก่าขึ้นมา

 เป็น Icon Menu สำหรับ Design Flat Slab & Flat Plate จะ Active เมื่อมีการเปิด ไฟล์ใหม่ หรือ เก่าขึ้นมา

 เป็น Icon Menu สำหรับ แสดง Shear Force Diagram จะ Active เมื่อมีการ Design อย่างใด อย่างหนึ่ง

 เป็น Icon Menu สำหรับ แสดง Bending Moment Diagram จะ Active เมื่อมีการ Design อย่างใด อย่างหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.หน้าจอแสดง Output จะแสดงผลการคำนวณได้แก่ โมเมนต์ แรงเฉือน และอื่น ๆ ซึ่งจะแสดงเป็นกราฟิค หรือเป็นรายการ

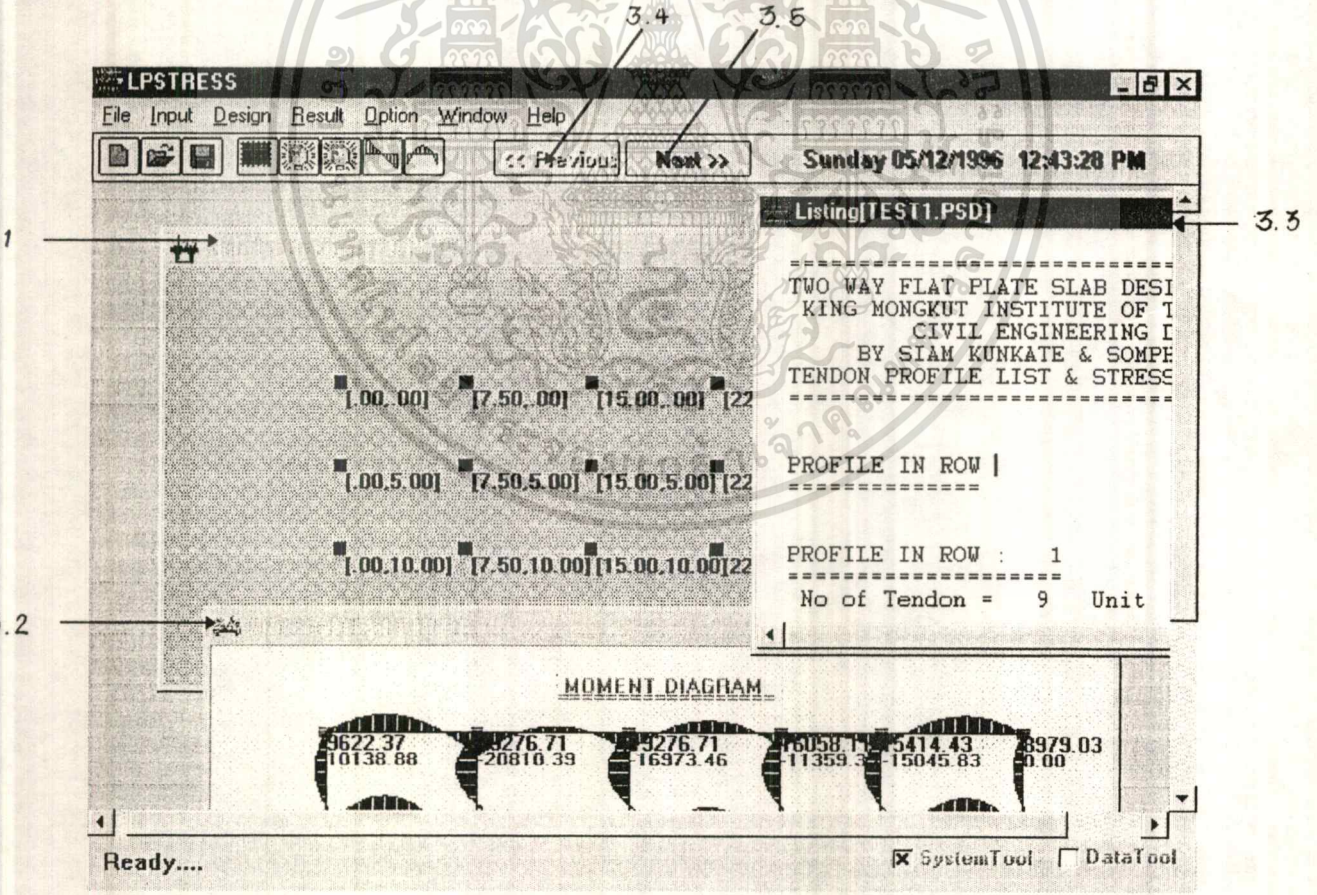
3.1 Slab To Design เป็นหน้าต่าง แสดงรูปของพื้นตอน Load ข้อมูลขึ้นมา จุดสีดำแสดงตำแหน่งเสา และในวงเล็บเป็น Coordinate ของเสา โดยแกน X วัดจากซ้ายไปขวา แกน Y วัดจากบนลงล่าง

3.2 Graphics เป็นหน้าต่าง แสดง Graphics ของ Shear Diagram ,Moment Diagram ,Profile of Tendon

3.3 Listing เป็นหน้าต่าง แสดง ผลที่เป็น Text ของการ Design

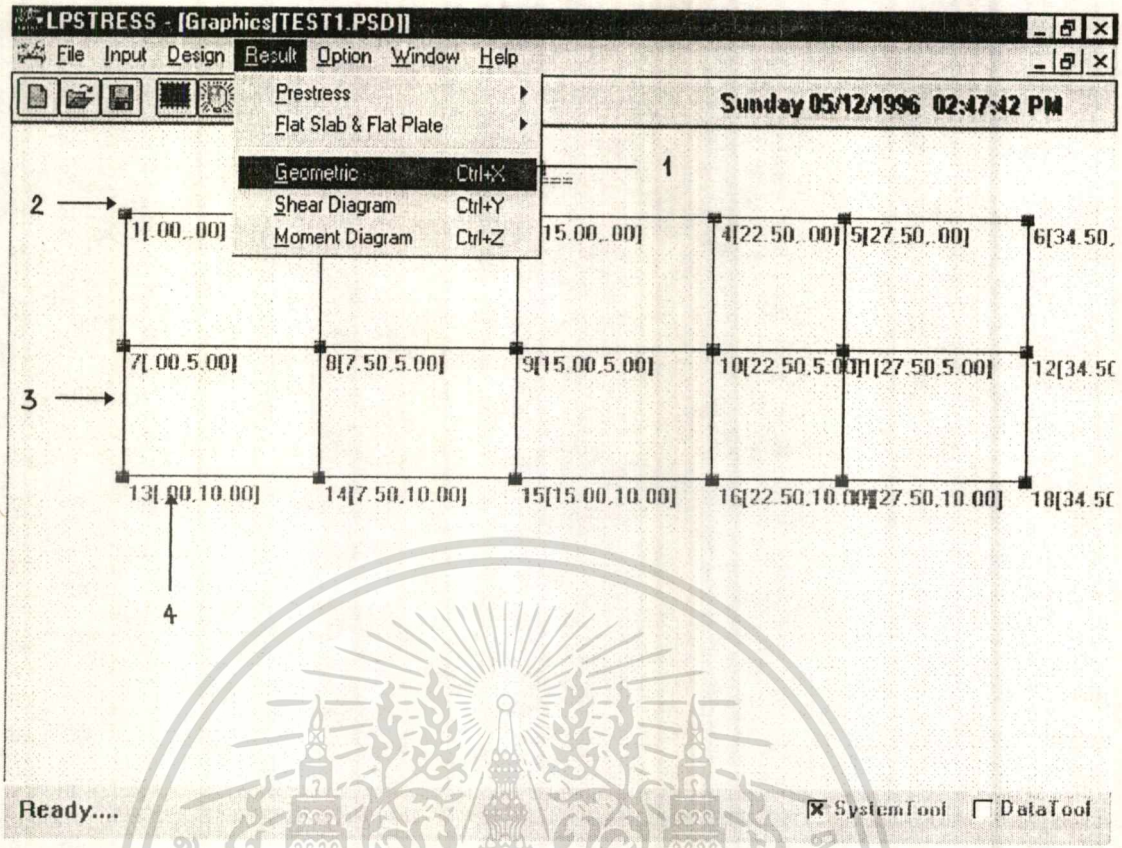
3.4 ปุ่ม Previous จะปรากฏเมื่อหน้าต่าง Listing ปรากฏเท่านั้น สำหรับเปิด Listing ย้อนกลับไปหน้าที่ผ่านมาแล้วของ Listing เมื่อ Listing มีหลายหน้า

3.5 ปุ่ม Next จะปรากฏเมื่อหน้าต่าง Listing ปรากฏเท่านั้น สำหรับเปิด Listing ไปหน้าต่อไปของ Listing เมื่อ Listing มีหลายหน้า



รูปที่ 12 แสดงหน้าจอซึ่งแสดง หน้าต่าง output ของการ Design

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

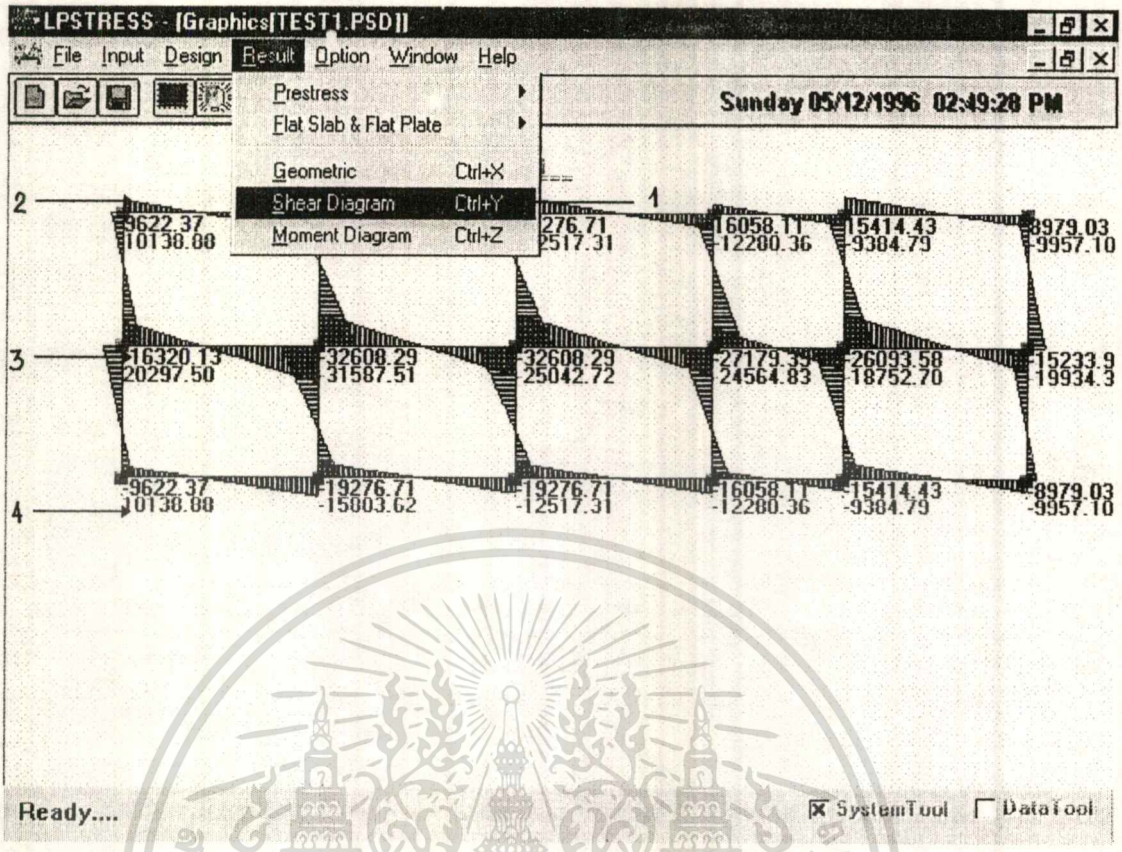


รูปที่ 13 แสดงหน้าจอซึ่งแสดง หน้าต่าง output ของ Slab To Design

อธิบายเพิ่มเติม

1. เมื่อทำการ Design แล้วคลิกที่ เมนูหมายเลข 1 จะแสดงOutput เป็นรูปแบบของพื้นที่ออกแบบ
2. แสดงตำแหน่งเสา
3. แสดง Element ที่เชื่อมต่อระหว่างเสาสองเสา
4. ตัวเลขหน้าวงเล็บบอกหมายเลขเสา ตัวเลขในวงเล็บบอก Co-ordinate x,y ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

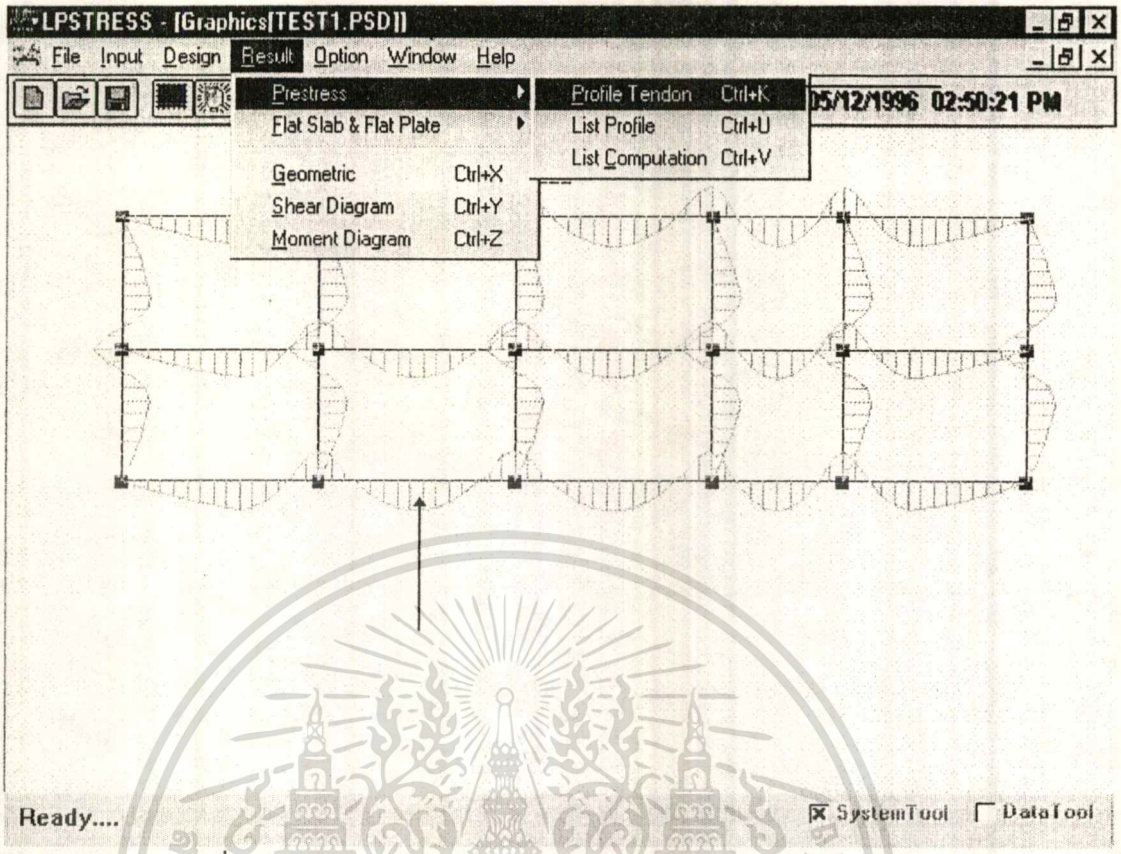


รูปที่ 14 แสดง Shear Force Diagram

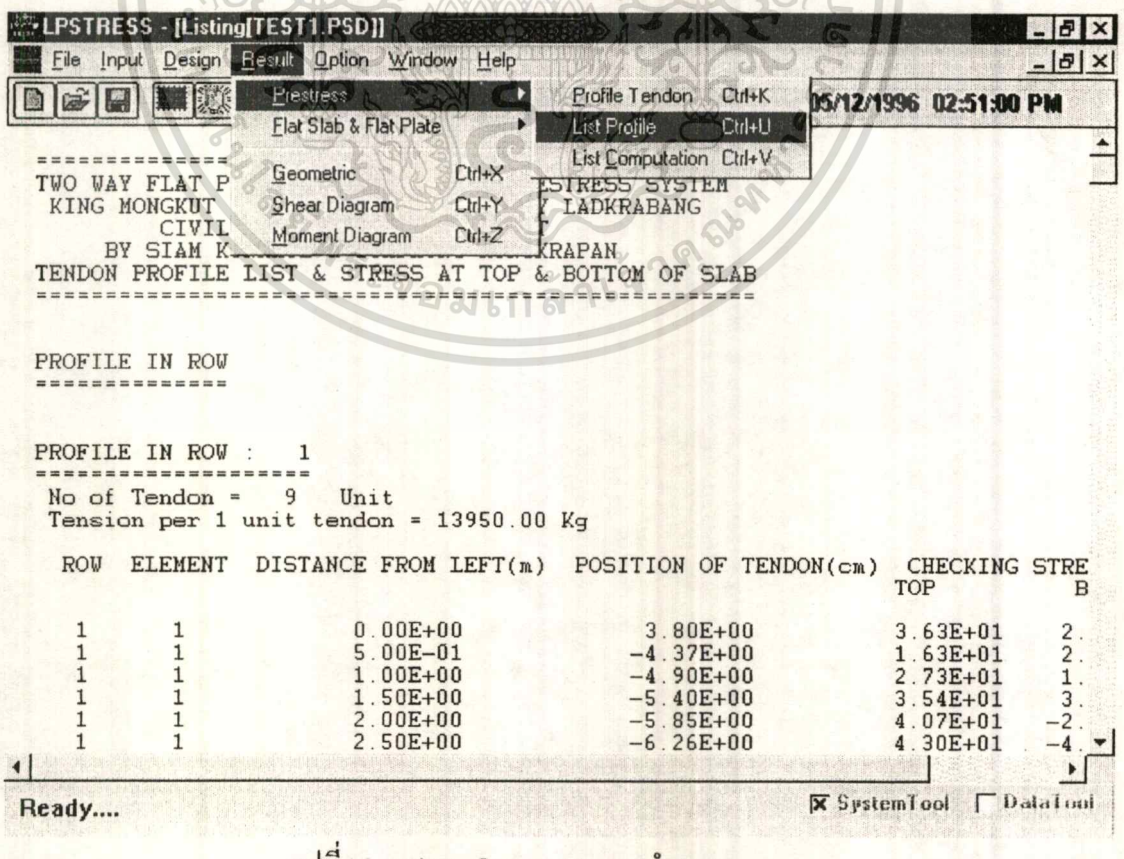
อธิบายเพิ่มเติม

1. เมื่อทำการ Design แล้วคลิกที่ เมนูหมายเลข 1 จะแสดง Shear Force Diagram
2. Shear Force Diagram
3. ตัวเลขสีน้ำเงิน เป็นค่าแรงเฉือนที่เกิดขึ้นที่ Element ในแนวแกน X
4. ตัวเลขสีแดง เป็นค่าแรงเฉือนที่เกิดขึ้นที่ Element ในแนวแกน Y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



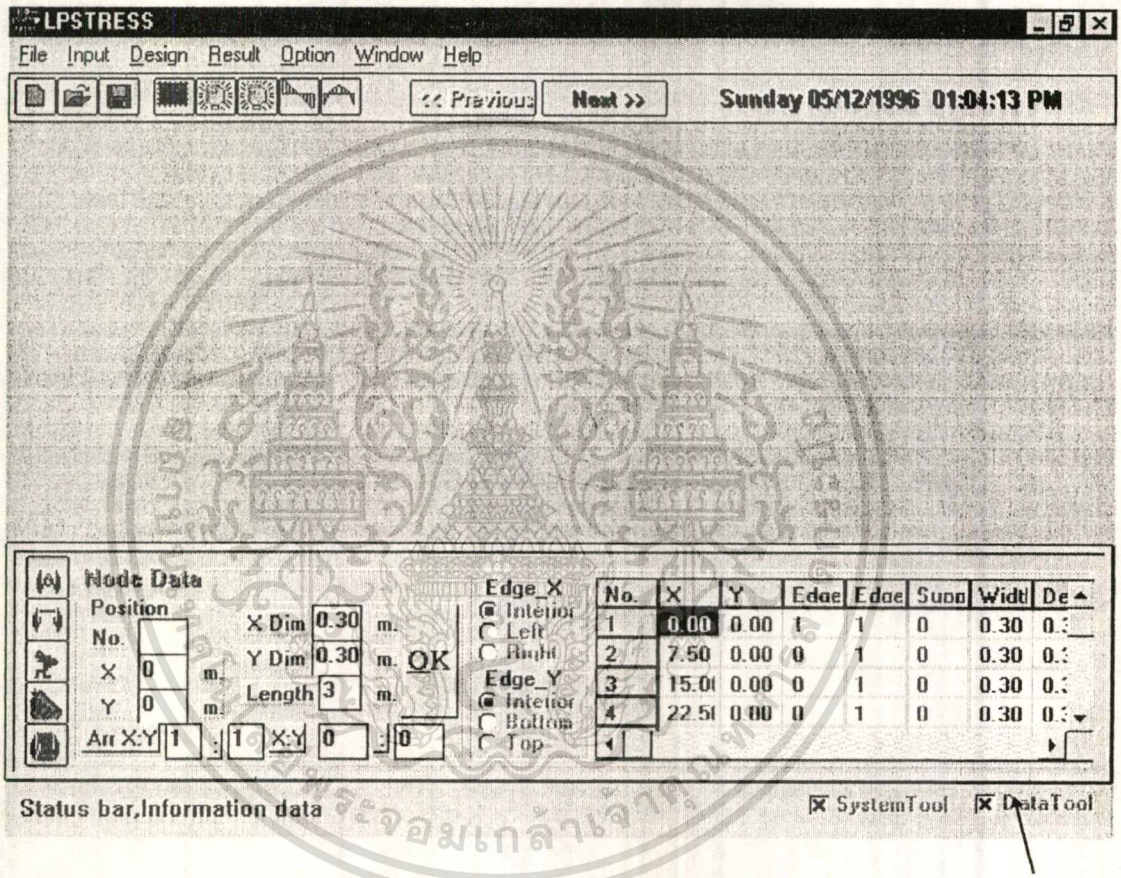
รูปที่ 15 แสดง Profile ของการวางลวดเหล็กเสริม



รูปที่ 16 แสดง Output รายการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

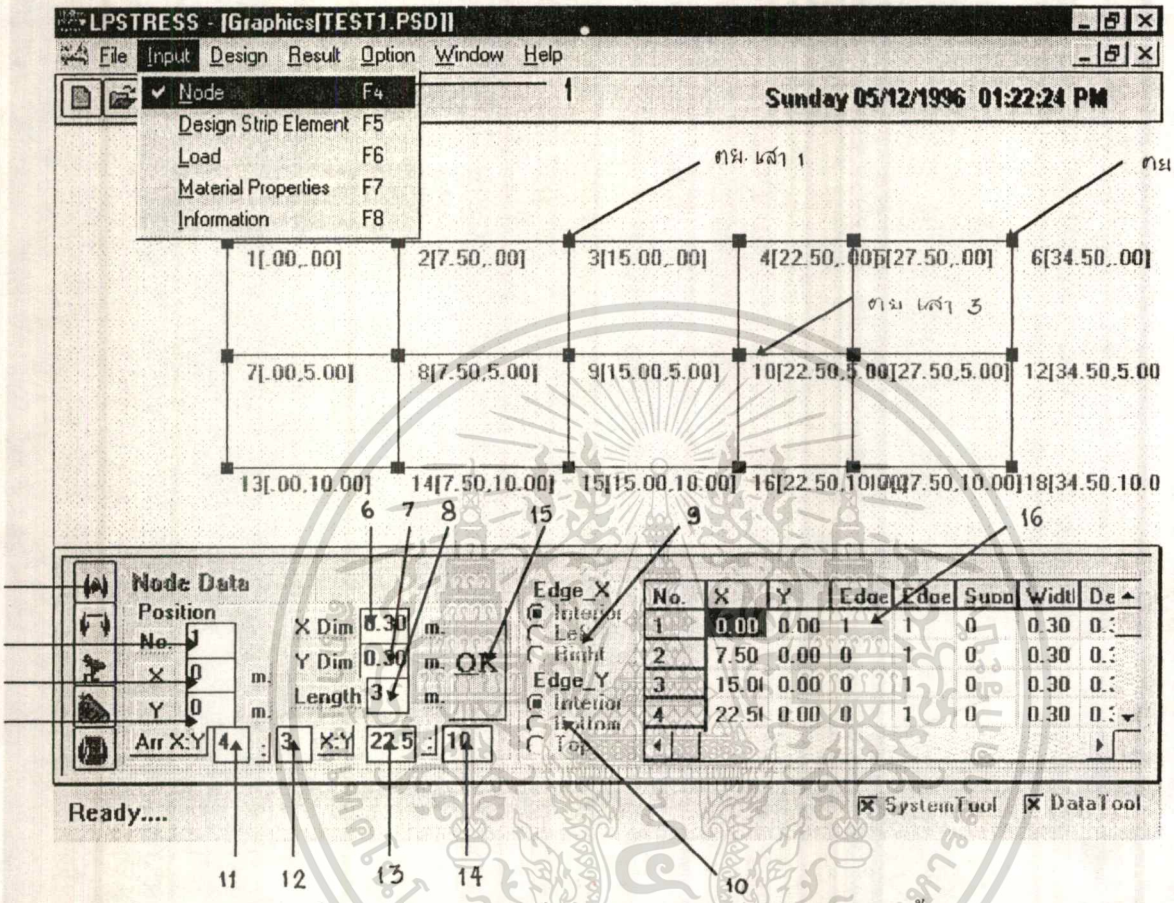
4.หน้าจอแสดงส่วนที่ต้องใส่ข้อมูล ซึ่งจะมีอยู่หลายหน้าจอ สามารถจะเลือกโดยใช้ Icon ที่เป็นรูปภาพที่สื่อความหมายทำให้โปรแกรมใช้ได้สะดวกขึ้น แต่ต้อง Check ที่ Data Tool ใน Status Bar ก่อน จะปรากฏ Panel ของการใส่ข้อมูลในการ Design ปรากฏขึ้นมา



รูปที่ 17 แสดงหน้าจอซึ่งแสดงการคลิก Data Tool

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 Node Data



รูปที่ 18 แสดงหน้าจอซึ่งแสดง Panel การใส่ข้อมูลของพื้น

อธิบายการใส่ข้อมูลตามรูปตามตัวเลขที่ปรากฏ

1,2 คลิกที่นี่จะมี Panel ใส่ข้อมูลปรากฏดังรูป

3 หมายเลขของเสา เป็นจำนวนเต็มบวกและต้องเรียงเช่น ข้อมูลเสามี 8 ต้น เมื่อต้องการเพิ่มเสาอีกก็ต้องกรอกข้อมูลเลข 9 เป็นเสาที่ 9

4. Coordinate ของเสาในแกน X

5. Coordinate ของเสาในแกน Y

6. ขนาดของเสาในแกน X

7. ขนาดของเสาในแกน Y

8. ความยาวของเสา

9. ลักษณะความเป็นขอบในแกน X ของเสา มี 3 แบบ คือ Interior,Left,Right เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเสาที่ 1 Edge X เป็น Interior

ตัวอย่างเสาที่ 2 Edge X เป็น Right

ตัวอย่างเสาที่ 3 Edge X เป็น Interior

10 ลักษณะความเป็นขอบในแกน Y ของเสา มี 3 แบบ คือ Interior,Top,Bottom เช่น

ตัวอย่างเสาที่ 1 Edge Y เป็น Top

ตัวอย่างเสาที่ 2 Edge Y เป็น Top

ตัวอย่างเสาที่ 3 Edge Y เป็น Interior

การใส่ข้อมูลเสาเป็นชุด

11 จำนวน Column ของเสาในแกน X ที่เรียงเป็น Array

12 จำนวน Row ของเสาในแกน Y ที่เรียงเป็น Array

13 Coordinate ในแกน X ของเสาในด้านขวาล่าง

14 Coordinate ในแกน Y ของเสาในด้านขวาล่าง

ตัวอย่างการใส่ข้อมูลเสาเป็นชุด

ดังในรูปที่ 13 สมมติว่าระยะห่างของทุกเสาเป็น 5 ทั้งหมด

จะใส่ No เป็น 1

X เป็น 0

Y เป็น 0

ArrX:Y แกน X เป็น 6 (มีเสาอยู่ 6 Column)

แกน Y เป็น 3 (มีเสาอยู่ 3 Row)

X:Y แกน X เป็น 25 (0+5*5=25)

แกน Y เป็น 15 (0+3*5=15)

ถ้าต้องการใส่ข้อมูล 1 Node ArrX:Y เป็น 1,1 X:Y เป็น 0,0

15 ปุ่ม OK กดเมื่อกรอกข้อมูลเรียบร้อยแล้ว

16 เป็นตารางแสดงข้อมูล Node ที่ใส่เรียบร้อยแล้ว

ช่อง 1 เป็นหมายเลขของ Node

ช่อง 2 เป็น Coordinate แกน X

ช่อง 3 เป็น Coordinate แกน Y

ช่อง 4 เป็นสถานะของขอบแกน X

0 เป็น Interior

1 เป็น Left

2 เป็น Right

ช่อง 5 เป็นสถานะของขอบแกน Y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0 เป็น Interior

1 เป็น Top

2 เป็น Bottom

ช่อง 6 ชนิดของฐานรองรับเป็น 0 แสดงว่ามีเสา

ช่อง 7 ขนาดเสาแกน X

ช่อง 8 ขนาดเสาแกน Y

การแก้ไขข้อมูล Node ที่ใส่ไปแล้ว

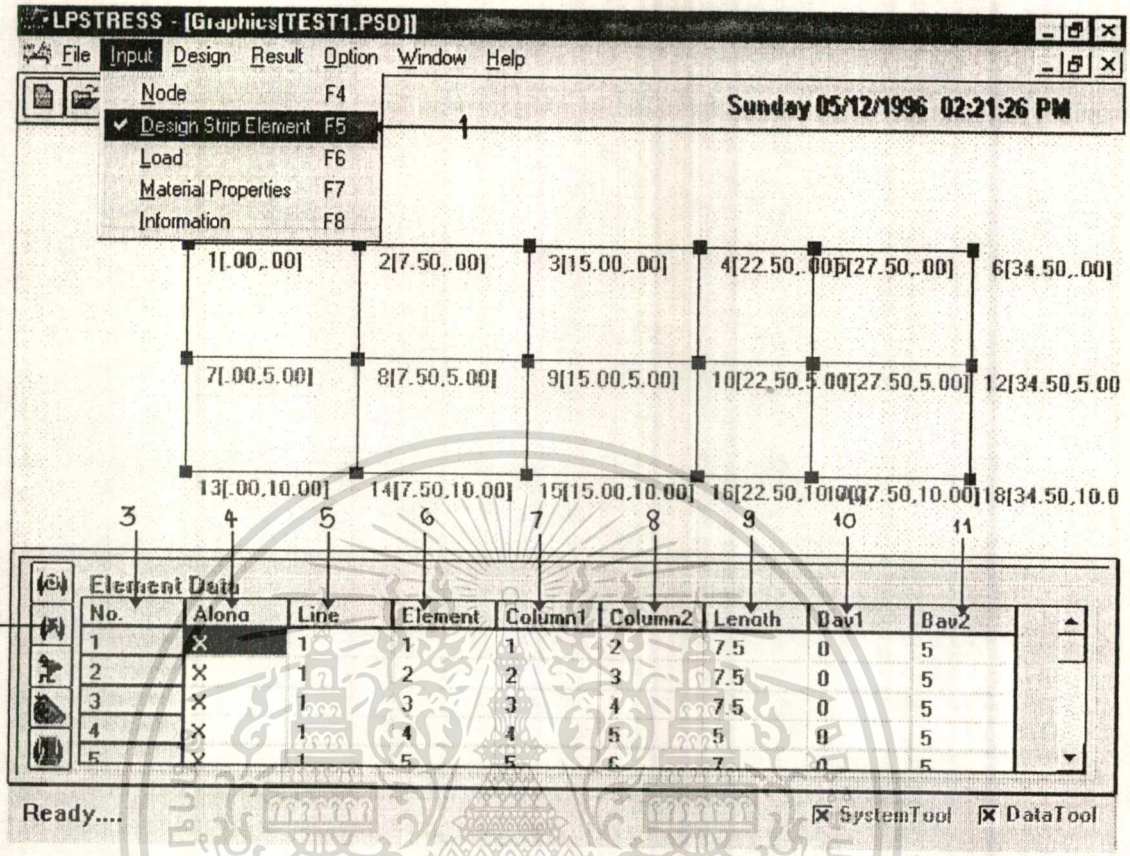
1 ลบโดยการเลือกที่ Node ที่ต้องการลบแล้วกด Back space

2 การแก้ไขข้อมูล โดยการเลือกที่ Node ที่ต้องการแก้ไขแล้ว ข้อมูลทั้งหมดจะปรากฏที่ช่องใส่ข้อมูล แล้วไปแก้ไขที่ช่องใส่ข้อมูลแล้วกด OK การแก้ไขก็เสร็จสมบูรณ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 Element Data



รูปที่ 19 แสดงหน้าจอซึ่งแสดง Panel แสดง Design strip

อธิบายการส่ข้อมูลตามรูปตามตัวเลขที่ปรากฏ

1,2 คลิกที่นี่จะปรากฏ Panel ที่ใช้ในการแสดง Design strip Element จะปรากฏข้อมูลเมื่อมีการ Design อย่างใดอย่างหนึ่งแล้ว

3 No of Element ของ Element ที่อยู่แกน x,y

4 Along เป็นตัวบอกว่าเป็น Strip ในแกน X หรือ Y

5 Line เป็นตัวแสดง Line ของเสา

6 Element เป็นตัวแสดงหมายเลขของElement ใน Line นั้น

7 Column 1 เชื่อมระหว่าง Element

8 Column 2 เชื่อมระหว่าง Element

9 Length ความยาวของ Element

10 Bay 1 ความกว้างของช่วงพื้น

ถ้าเป็น Element ทางแกน x ก็คือความกว้างช่วงพื้นข้างบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเป็น Element ทางแกน y ก็คือความกว้างช่วงพื้นด้านซ้าย

11 Bay 2 ความกว้างของช่วงพื้น

ถ้าเป็น Element ทางแกน x ก็คือความกว้างช่วงพื้นด้านล่าง

ถ้าเป็น Element ทางแกน y ก็คือความกว้างช่วงพื้นด้านขวา

Along แสดงว่าอยู่ในแนวแกน x หรือ y

Line แสดง Line ของเสา

Column 1 เชื่อม Element

Column 2 เชื่อม Element

Length ความยาว

Bay 1 ความกว้างของช่วงพื้น ถ้าเป็น element ทางแกน x ก็คือช่วงพื้นข้างบน

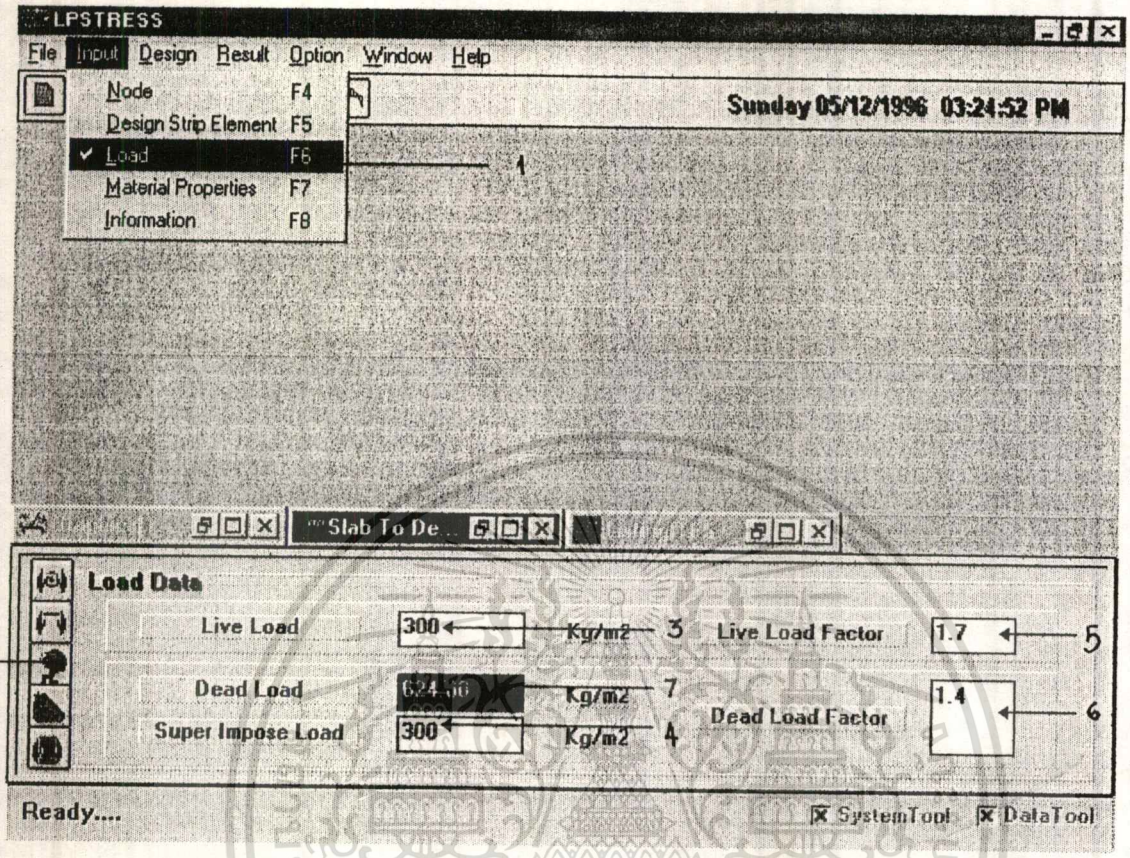
ถ้าเป็น element ทางแกน y ก็คือช่วงพื้นด้านซ้าย

Bay 2 ความกว้างของช่วงพื้น ถ้าเป็น element ทางแกน x ก็คือช่วงพื้นข้างล่าง

ถ้าเป็น element ทางแกนy ก็คือช่วงพื้นด้านขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 Load Data



รูปที่ 20 แสดงหน้าจอซึ่งแสดง Panel การใส่ข้อมูลของแรงกระทำบนพื้นและ Factor อธิบายรูปเพิ่มเติม

1,2 เมื่อคลิกที่นี่ จะแสดง Panel ดังแสดงในรูป

3. ใส่ข้อมูลน้ำหนักจร

4. ใส่ข้อมูล Super Impose Load ใช้สำหรับการออกแบบคอนกรีตอัดแรงเท่านั้น

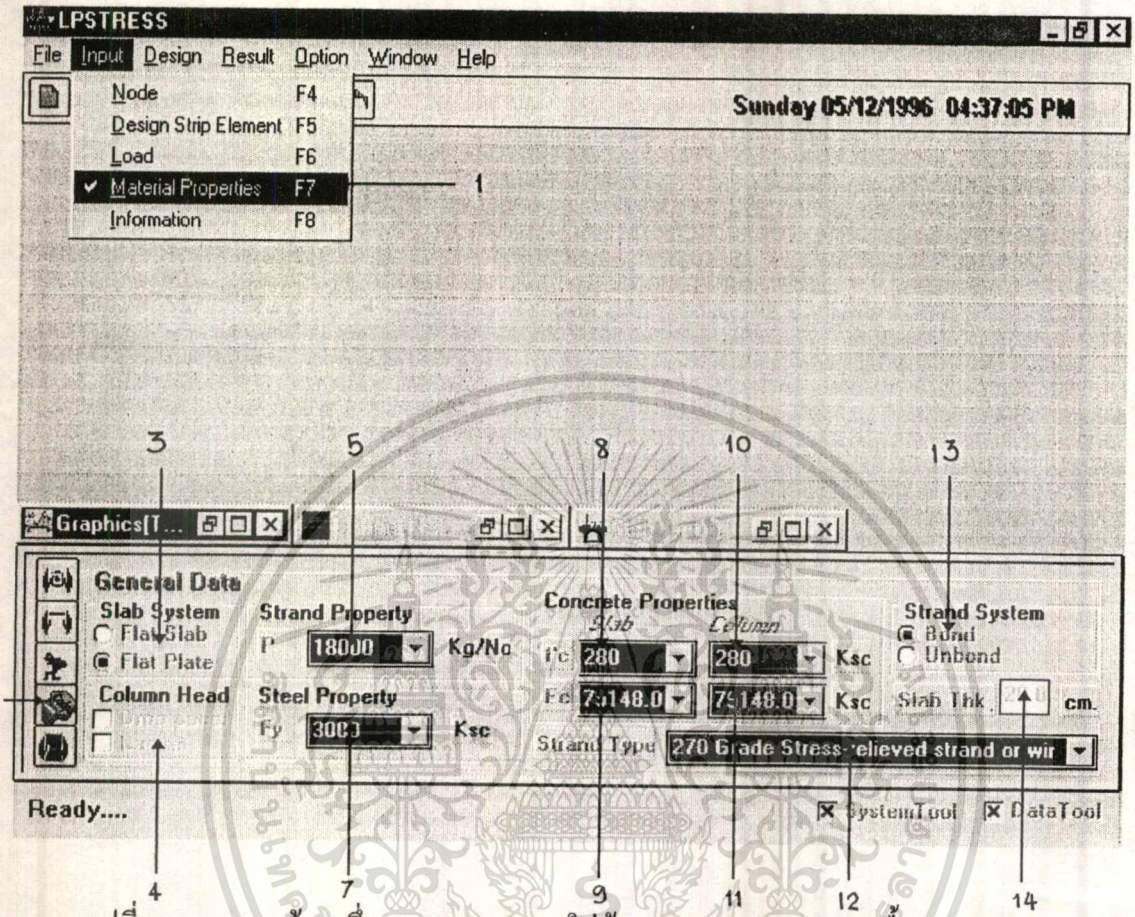
5. ใส่ข้อมูล Load Factor สำหรับน้ำหนักจร (ตามมาตรฐาน ACI Default ไว้ที่ 1.7)

6. ใส่ข้อมูล Load Factor สำหรับน้ำหนักตาย (ตามมาตรฐาน ACI Default ไว้ที่ 1.4)

7. เป็น Dead Load ที่เป็น Output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 General Data



รูปที่ 21 แสดงหน้าจอซึ่งแสดง Panel การใส่ข้อมูลของแรงกระทำบนพื้นและ Factor

อธิบายรูปเพิ่มเติม

1,2 คลิกที่นี่จะปรากฏ Panel ใส่ข้อมูลของวัสดุ

3 เลือกรูปแบบพื้นเป็น Flat Plate หรือ Flat Slab ถ้าเลือก Flat Slab จะ Enable ช่องเลือก Column Head

4 เลือกรูปแบบชนิดหัวเสา

5 ใส่ข้อมูลของ Yield Strength ของTendon ต่อเส้น

7 ใส่ข้อมูลของ Yield Strength ของ Reinforce Steel

8 f'c ของ Concrete พื้น

9 Ec ของ Concrete พื้น

10 f'c ของ Concrete เสา

11 Ec ของ Concrete เสา

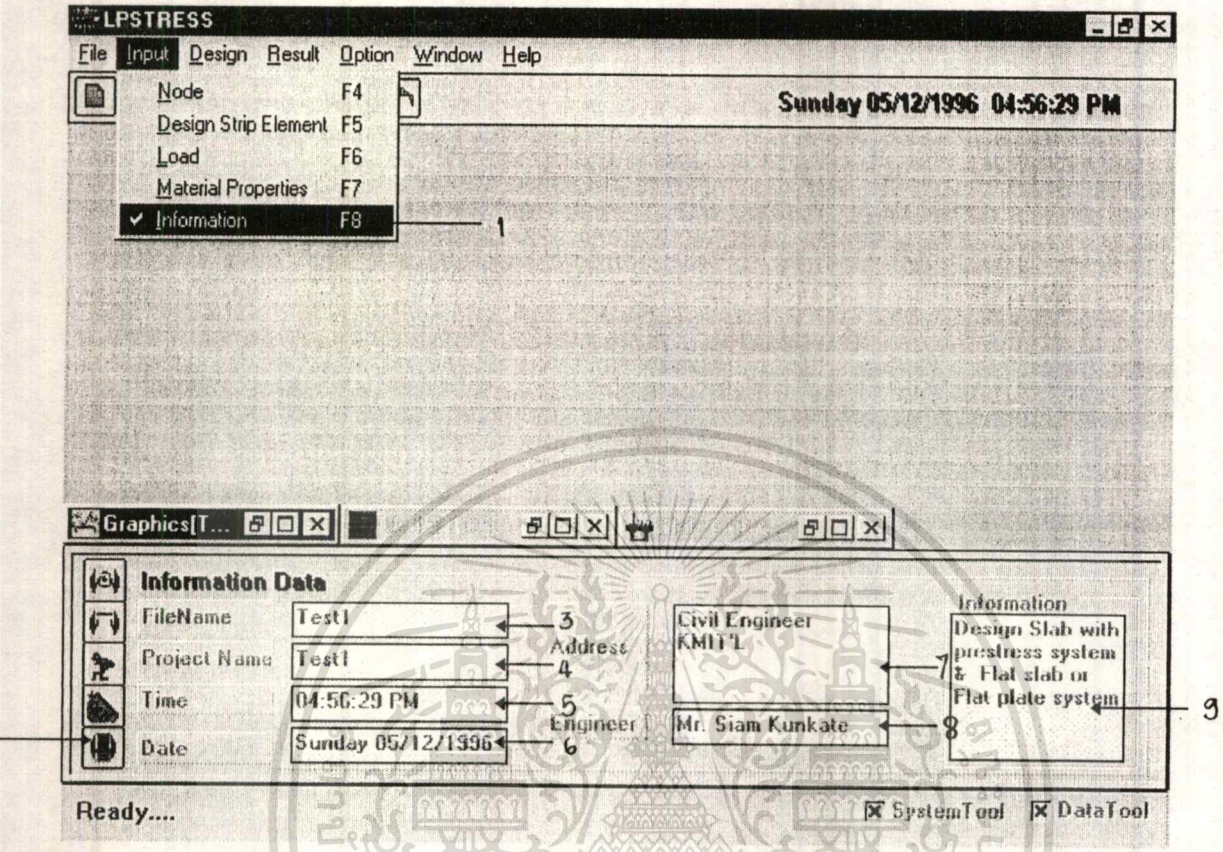
12 ชนิดของเส้นลวด Tendon

13 เลือกระบบการวางลวด

14 ความหนาของพื้น เป็น Output ที่เปลี่ยนแปลงตามที่ Design

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 Information Data



รูปที่ 22 แสดงหน้าจอซึ่งแสดง Panel การใส่ข้อมูล Information

อธิบายรูปเพิ่มเติม

1,2 คลิกที่นี่จะปรากฏ Panel ใส่ข้อมูล Information

3 ใส่ข้อมูลเป็นการตั้งชื่อไฟล์

4. ใส่ชื่อ โครงงาน

5. เป็นการบอกเวลาที่ใช้โปรแกรมออกแบบแผ่นพื้น

6. เป็นการบอกวัน เดือน ปี ที่ทำการออกแบบโครงงาน

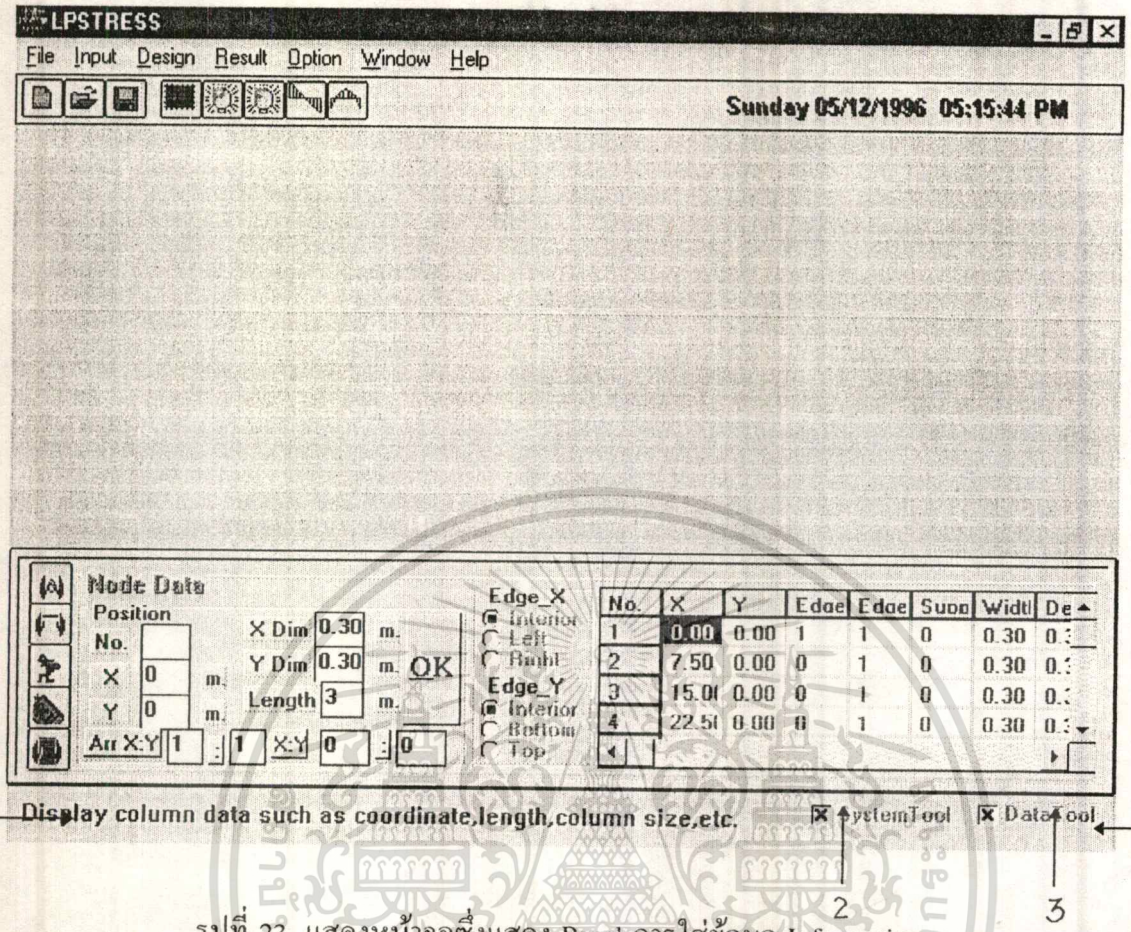
7. เป็นการใส่ที่อยู่ตามที่ผู้ใช้ต้องการ

8. เป็นการใส่ชื่อผู้ใช้โปรแกรม

9. เป็นข้อมูลเพิ่มเติมบางประการที่ผู้ใช้ต้องการระบุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.Status Bar เป็นส่วนที่อำนวยความสะดวกในการบอกสถานะของโปรแกรมในขณะนั้น



รูปที่ 23 แสดงหน้าจอซึ่งแสดง Panel การใส่ข้อมูล Information

อธิบายรูปเพิ่มเติม

1 แสดง Status Bar

2 System Tool เป็นส่วนที่ใช้แสดงสถานะของโปรแกรมขณะที่ทำงานอยู่ว่าแสดง Menu Icon ที่เป็นรูปภาพปรากฏอยู่ด้านบนไว้เพื่ออำนวยความสะดวกในการทำงาน

3 Data Tool เป็นส่วนที่ใช้แสดงสถานะของโปรแกรมขณะที่ทำงานอยู่ว่าแสดง Panel สำหรับใส่ข้อมูล

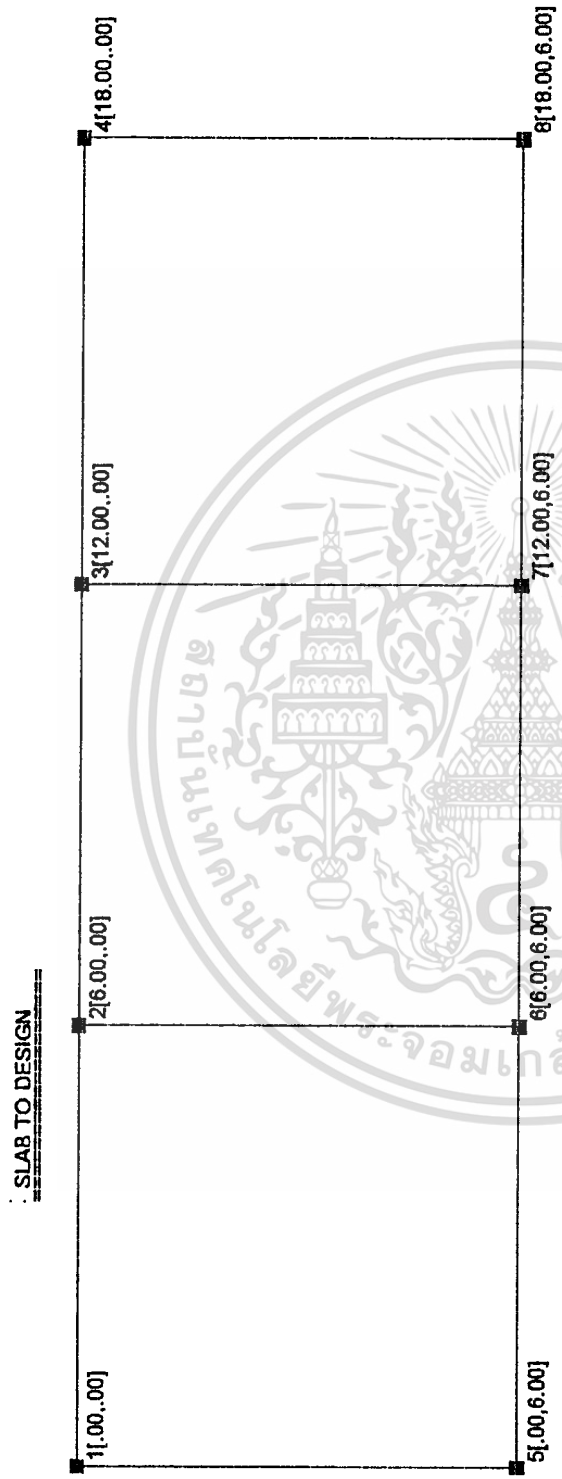
4 นอกจากนี้จะมีข้อความที่แสดงสถานะของโปรแกรมว่าขณะนี้โปรแกรมกำลังแสดงส่วนใดของโปรแกรมอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

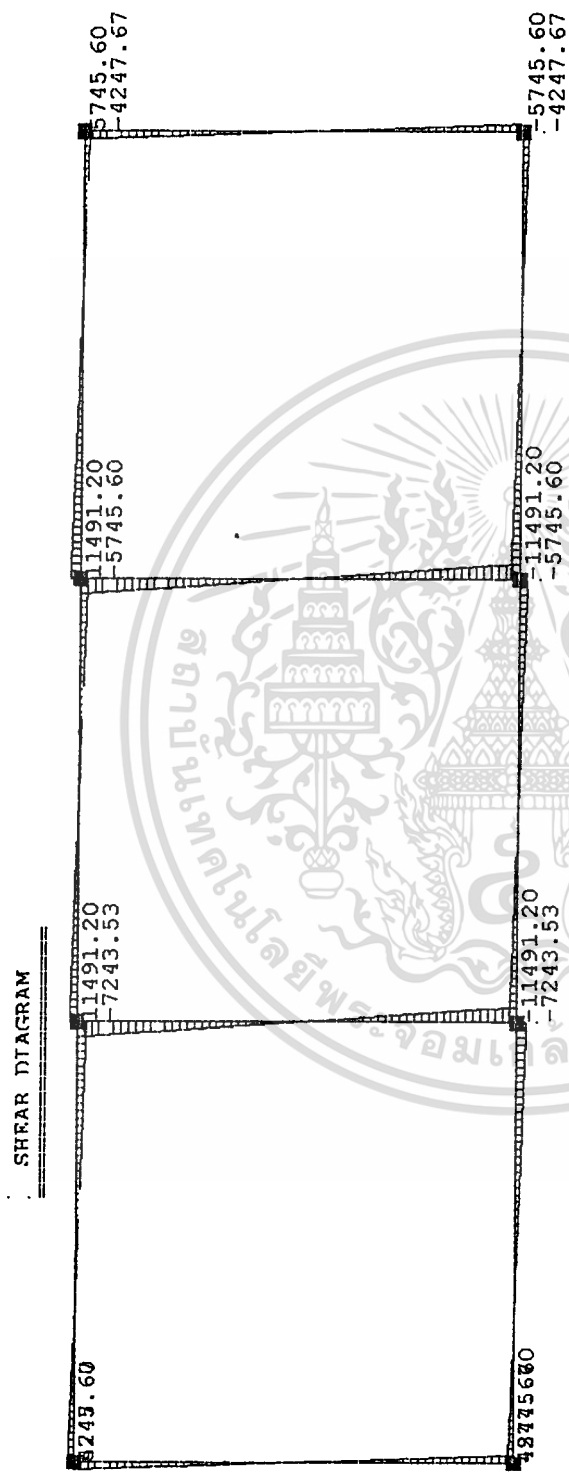
ตอนที่ 4

ตัวอย่างการออกแบบโดยใช้โปรแกรม " LPSTRESS"

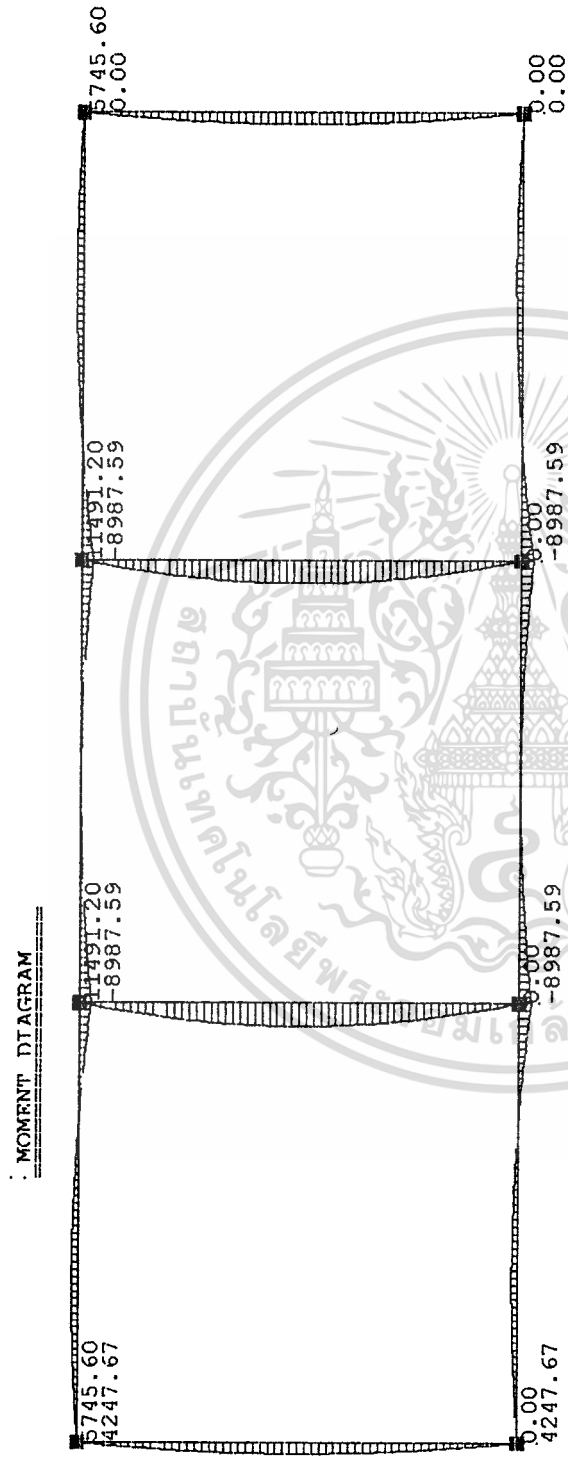
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



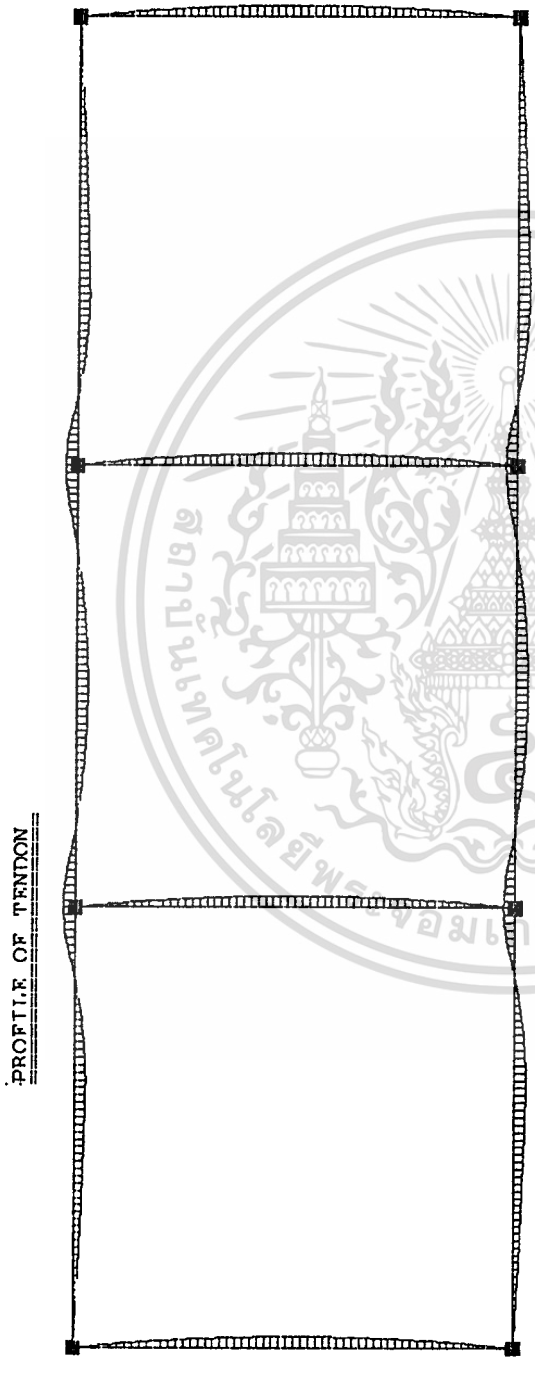
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM
KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG
CIVIL ENGINEERING DEPARTMENT
BY SIAM KUNKATE & SOMPHONG KONGKRAPAN
TENDON PROFILE LIST & STRESS AT TOP & BOTTOM OF SLAB

PROFILE IN ROW

PROFILE IN ROW : 1

No of Tendon = 14 Unit
Tension per 1 unit tendon = 13958.00 Kg

ROW	ELEMENT	DISTANCE FROM LEFT(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
1	1	0.00E+00	1.30E+00	4.03E+01	2.02E+01
1	1	5.00E-01	-1.54E+00	2.80E+01	4.05E+01
1	1	1.00E+00	-1.76E+00	3.34E+01	3.51E+01
1	1	1.50E+00	-1.96E+00	3.64E+01	3.21E+01
1	1	2.00E+00	-2.14E+00	3.69E+01	3.16E+01
1	1	2.50E+00	-2.30E+00	3.50E+01	3.35E+01
1	1	3.00E+00	-2.44E+00	3.07E+01	3.78E+01
1	1	3.50E+00	-2.55E+00	2.40E+01	4.45E+01
1	1	4.00E+00	-2.50E+00	1.64E+01	5.21E+01
1	1	4.50E+00	-2.11E+00	9.87E+00	5.87E+01
1	1	5.00E+00	-1.42E+00	3.93E+00	6.46E+01
1	1	5.50E+00	2.15E+00	2.65E+01	4.21E+01
1	1	6.00E+00	2.60E+00	1.26E+01	5.59E+01
1	2	0.00E+00	2.60E+00	1.26E+01	5.59E+01
1	2	5.00E-01	2.24E+00	2.33E+01	4.52E+01
1	2	1.00E+00	1.44E+00	2.66E+01	4.19E+01
1	2	1.50E+00	-1.79E+00	9.40E-01	6.76E+01
1	2	2.00E+00	-2.24E+00	2.69E+00	6.58E+01
1	2	2.50E+00	-2.51E+00	3.74E+00	6.48E+01
1	2	3.00E+00	-2.60E+00	4.09E+00	6.44E+01
1	2	3.50E+00	-2.51E+00	3.71E+00	6.48E+01
1	2	4.00E+00	-2.24E+00	2.69E+00	6.58E+01
1	2	4.50E+00	-1.79E+00	9.40E-01	6.76E+01
1	2	5.00E+00	1.44E+00	2.66E+01	4.19E+01
1	2	5.50E+00	2.24E+00	2.33E+01	4.52E+01
1	2	6.00E+00	2.60E+00	1.26E+01	5.59E+01
1	3	0.00E+00	2.60E+00	1.26E+01	5.59E+01
1	3	5.00E-01	2.15E+00	2.65E+01	4.21E+01
1	3	1.00E+00	-1.42E+00	3.93E+00	6.46E+01
1	3	1.50E+00	-2.11E+00	9.87E+00	5.87E+01
1	3	2.00E+00	-2.50E+00	1.64E+01	5.21E+01
1	3	2.50E+00	-2.55E+00	2.40E+01	4.45E+01
1	3	3.00E+00	-2.44E+00	3.07E+01	3.78E+01
1	3	3.50E+00	-2.30E+00	3.50E+01	3.35E+01
1	3	4.00E+00	-2.14E+00	3.69E+01	3.16E+01
1	3	4.50E+00	-1.96E+00	3.64E+01	3.21E+01
1	3	5.00E+00	-1.76E+00	3.34E+01	3.51E+01
1	3	5.50E+00	-1.54E+00	2.80E+01	4.05E+01
1	3	6.00E+00	1.30E+00	4.03E+01	2.02E+01

PROFILE IN ROW : 2

No of Tendon = 14 Unit
Tension per 1 unit tendon = 13958.00 Kg

ROW	ELEMENT	DISTANCE FROM LEFT(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
2	1	0.00E+00	1.30E+00	4.03E+01	2.02E+01
2	1	5.00E-01	-1.54E+00	2.80E+01	4.05E+01
2	1	1.00E+00	-1.76E+00	3.34E+01	3.51E+01
2	1	1.50E+00	-1.96E+00	3.64E+01	3.21E+01
2	1	2.00E+00	-2.14E+00	3.69E+01	3.16E+01
2	1	2.50E+00	-2.30E+00	3.50E+01	3.35E+01
2	1	3.00E+00	-2.44E+00	3.07E+01	3.78E+01
2	1	3.50E+00	-2.55E+00	2.40E+01	4.45E+01
2	1	4.00E+00	-2.50E+00	1.64E+01	5.21E+01
2	1	4.50E+00	-2.11E+00	9.87E+00	5.87E+01
2	1	5.00E+00	-1.42E+00	3.93E+00	6.46E+01
2	1	5.50E+00	2.15E+00	2.65E+01	4.21E+01
2	1	6.00E+00	2.60E+00	1.26E+01	5.59E+01
2	2	0.00E+00	2.60E+00	1.26E+01	5.59E+01
2	2	5.00E-01	2.24E+00	2.33E+01	4.52E+01
2	2	1.00E+00	1.44E+00	2.66E+01	4.19E+01
2	2	1.50E+00	-1.79E+00	9.40E-01	6.76E+01
2	2	2.00E+00	-2.24E+00	2.69E+00	6.58E+01
2	2	2.50E+00	-2.51E+00	3.74E+00	6.48E+01
2	2	3.00E+00	-2.60E+00	4.09E+00	6.44E+01
2	2	3.50E+00	-2.51E+00	3.74E+00	6.48E+01
2	2	4.00E+00	-2.24E+00	2.69E+00	6.58E+01
2	2	4.50E+00	-1.79E+00	9.40E-01	6.76E+01
2	2	5.00E+00	1.44E+00	2.66E+01	4.19E+01
2	2	5.50E+00	2.24E+00	2.33E+01	4.52E+01
2	2	6.00E+00	2.60E+00	1.26E+01	5.59E+01
2	3	0.00E+00	2.60E+00	1.26E+01	5.59E+01
2	3	5.00E-01	2.15E+00	2.65E+01	4.21E+01
2	3	1.00E+00	-1.42E+00	3.93E+00	6.46E+01
2	3	1.50E+00	-2.11E+00	9.87E+00	5.87E+01
2	3	2.00E+00	-2.50E+00	1.64E+01	5.21E+01
2	3	2.50E+00	-2.55E+00	2.40E+01	4.45E+01
2	3	3.00E+00	-2.44E+00	3.07E+01	3.78E+01
2	3	3.50E+00	-2.30E+00	3.50E+01	3.35E+01
2	3	4.00E+00	-2.14E+00	3.69E+01	3.16E+01
2	3	4.50E+00	-1.96E+00	3.64E+01	3.21E+01
2	3	5.00E+00	-1.76E+00	3.34E+01	3.51E+01
2	3	5.50E+00	-1.54E+00	2.80E+01	4.05E+01
2	3	6.00E+00	1.30E+00	4.03E+01	2.02E+01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารนี้ทุกครั้งที่มีการนำไปใช้

PROFILE IN COLUMN : 1

No of tendon = 14 Unit
Tension per 1 unit tendon = 13950.00 Kg

COLUMN	ELEMENT	DISTANCE FROM TOP(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
1	1	6.50E+00	1.30E+00	4.03E+01	2.02E+01
1	1	6.50E+00	-1.70E+00	3.05E+01	3.00E+01
1	1	6.50E+00	-2.02E+00	3.09E+01	2.96E+01
1	1	6.50E+00	-2.27E+00	4.55E+01	2.31E+01
1	1	6.50E+00	-2.46E+00	5.01E+01	1.84E+01
1	1	6.50E+00	-2.56E+00	5.29E+01	1.56E+01
1	1	6.50E+00	-2.60E+00	5.39E+01	1.46E+01
1	1	6.50E+00	-2.56E+00	5.29E+01	1.56E+01
1	1	6.50E+00	-2.46E+00	5.01E+01	1.84E+01
1	1	6.50E+00	-2.27E+00	4.55E+01	2.31E+01
1	1	6.50E+00	-2.02E+00	3.09E+01	2.96E+01
1	1	6.50E+00	-1.70E+00	3.05E+01	3.00E+01
1	1	6.50E+00	1.30E+00	4.03E+01	2.02E+01

PROFILE IN COLUMN : 2

No of tendon = 28 Unit
Tension per 1 unit tendon = 13950.00 Kg

COLUMN	ELEMENT	DISTANCE FROM TOP(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
2	1	6.50E+00	1.30E+00	9.67E+01	4.04E+01
2	1	6.50E+00	-1.70E+00	6.10E+01	7.61E+01
2	1	6.50E+00	-2.02E+00	7.70E+01	5.92E+01
2	1	6.50E+00	-2.27E+00	9.09E+01	4.61E+01
2	1	6.50E+00	-2.46E+00	1.00E+02	3.68E+01
2	1	6.50E+00	-2.56E+00	1.06E+02	3.12E+01
2	1	6.50E+00	-2.60E+00	1.08E+02	2.93E+01
2	1	6.50E+00	-2.56E+00	1.06E+02	3.12E+01
2	1	6.50E+00	-2.46E+00	1.00E+02	3.68E+01
2	1	6.50E+00	-2.27E+00	9.09E+01	4.61E+01
2	1	6.50E+00	-2.02E+00	7.70E+01	5.92E+01
2	1	6.50E+00	-1.70E+00	6.10E+01	7.61E+01
2	1	6.50E+00	1.30E+00	9.67E+01	4.04E+01

PROFILE IN COLUMN : 3

No of tendon = 28 Unit
Tension per 1 unit tendon = 13950.00 Kg

COLUMN	ELEMENT	DISTANCE FROM TOP(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
3	1	6.50E+00	1.30E+00	9.67E+01	4.04E+01
3	1	6.50E+00	-1.70E+00	6.10E+01	7.61E+01
3	1	6.50E+00	-2.02E+00	7.70E+01	5.92E+01
3	1	6.50E+00	-2.27E+00	9.09E+01	4.61E+01
3	1	6.50E+00	-2.46E+00	1.00E+02	3.68E+01
3	1	6.50E+00	-2.56E+00	1.06E+02	3.12E+01
3	1	6.50E+00	-2.60E+00	1.08E+02	2.93E+01
3	1	6.50E+00	-2.56E+00	1.06E+02	3.12E+01
3	1	6.50E+00	-2.46E+00	1.00E+02	3.68E+01
3	1	6.50E+00	-2.27E+00	9.09E+01	4.61E+01
3	1	6.50E+00	-2.02E+00	7.70E+01	5.92E+01
3	1	6.50E+00	-1.70E+00	6.10E+01	7.61E+01
3	1	6.50E+00	1.30E+00	9.67E+01	4.04E+01

PROFILE IN COLUMN : 4

No of tendon = 14 Unit
Tension per 1 unit tendon = 13950.00 Kg

COLUMN	ELEMENT	DISTANCE FROM TOP(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
4	1	6.50E+00	1.30E+00	4.03E+01	2.02E+01
4	1	6.50E+00	-1.70E+00	3.05E+01	3.00E+01
4	1	6.50E+00	-2.02E+00	3.09E+01	2.96E+01
4	1	6.50E+00	-2.27E+00	4.55E+01	2.31E+01
4	1	6.50E+00	-2.46E+00	5.01E+01	1.84E+01
4	1	6.50E+00	-2.56E+00	5.29E+01	1.56E+01
4	1	6.50E+00	-2.60E+00	5.39E+01	1.46E+01
4	1	6.50E+00	-2.56E+00	5.29E+01	1.56E+01
4	1	6.50E+00	-2.46E+00	5.01E+01	1.84E+01
4	1	6.50E+00	-2.27E+00	4.55E+01	2.31E+01
4	1	6.50E+00	-2.02E+00	3.09E+01	2.96E+01
4	1	6.50E+00	-1.70E+00	3.05E+01	3.00E+01
4	1	6.50E+00	1.30E+00	4.03E+01	2.02E+01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
------------------------------------------------------
TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM
 KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG
        CIVIL ENGINEERING DEPARTMENT
     BY SIAM KUNKATE & SOMPHONG KONGKRAPAN
          LISTING COMPUTATION CHECKING
------------------------------------------------------


    LIST OF PRE-STRESSED DESIGN
Date         : Saturday 05/18/1996
Time         : 09:20:52 PM
File Name    : Untitled
Project      : Untitled
Engineer     : Mr. Siam Kunkate
Address      : Civil Engineer KMIT'L
Information  : Design Slab with prestress system &  Flat slab or Flat plate system
Length Unit  : Meter(m)
Load Unit    : Kilogram(Kg)
fc' (Slab)   : 280.00 ksc
fc' (Column): 280.00 ksc
fy (Strand) : 19000.00 ksc
Ec (Slab)       : 79148.04 ksc
Ec (Column)     : 79148.04 ksc
Slab Thickness : 19.00 cm
Slab Covering  : 8.20 cm
Drape          : 2.60 cm

COLUMN DATA
-----------

COLUMN NO.  CO_X(m)      CO_Y(m)      LENGTH(m)

      1     0.00E+0      0.00E+0      3.00E+0
      2     6.00E+0      0.00E+0      3.00E+0
      3     1.20E+1      0.00E+0      3.00E+0
      4     1.80E+1      0.00E+0      3.00E+0
      5     0.00E+0      6.00E+0      3.00E+0
      6     6.00E+0      6.00E+0      3.00E+0
      7     1.20E+1      6.00E+0      3.00E+0
      8     1.80E+1      6.00E+0      3.00E+0
-------------------------------------------------


DESIGN STRIP IN ROW
-------------------

 ROW   ELEMENT   START-COL  END-COL  LENGTH(m)   TOP-BAY(m) BOTTOM-BAY(m)

   1       1         1         2     6.00E+0     0.00E+0      6.00E+0
   1       2         2         3     6.00E+0     0.00E+0      6.00E+0
   1       3         3         4     6.00E+0     0.00E+0      6.00E+0
   2       1         5         6     6.00E+0     6.00E+0      0.00E+0
   2       2         6         7     6.00E+0     6.00E+0      0.00E+0
   2       3         7         8     6.00E+0     6.00E+0      0.00E+0
-------------------------------------------------------------------


DESIGN STRIP IN COLUMN
----------------------

COLUMN ELEMENT   START-COL  END-COL  LENGTH(m)   TOP-BAY(m) BOTTOM-BAY(m)

   1       1         1         5     6.00E+0     0.00E+0      6.00E+0
   2       1         2         6     6.00E+0     6.00E+0      6.00E+0
   3       1         3         7     6.00E+0     6.00E+0      6.00E+0
   4       1         4         8     6.00E+0     6.00E+0      0.00E+0
-------------------------------------------------------------------


SLAB SYSTEM : Flat Plate

Load factor :::
Wu = 1.4DL1.7LL
Unfactored Dead Load            = 456.00 kg/m^2
Unfactored Live Load            = 0.00 kg/m^2
Unfactored Supper imposed load = 0.00 kg/m^2

SHEAR CHECKING AT COLUMN,DROP PANEL,COLUMN CAPITAL AND BEAM ACTION SHEAR
------------------------------------------------------------------------

COLUMN                          PUNCHING SHEAR                                  BEAM-ACTION-SHEAR
  NO.          COLUMN          DROP-PANEL          COLUMN-CAPITAL
       Vc(kg)      Vu(kg)      Vc(kg)      Vu(kg)      Vc(kg)      Vu(kg)      Vc(kg)      Vu(kg)

   1  1.36E+04    5.26E+03    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.36E+03
   2  2.14E+04    1.06E+04    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.54E+03
   3  2.14E+04    1.06E+04    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.54E+03
   4  1.36E+04    5.26E+03    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.36E+03
   5  1.36E+04    5.26E+03    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.36E+03
   6  2.14E+04    1.06E+04    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.54E+03
   7  2.14E+04    1.06E+04    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.54E+03
   8  1.36E+04    5.26E+03    0.00E+00    0.00E+00    0.00E+00    0.00E+00    9.58E+03    1.36E+03
-------------------------------------------------------------------------------------------------


------------------------------------------------------
CHECK CRACKING MOMENT & ULTIMATE MOMENT & DEFLECTION
------------------------------------------------------


 ROW :   1
 ---------
 No of tendon              -  14  Unit
 Tension per 1 unit tendon - 13950.00 Kg
 Mcr - 10480.15 kg-m
 Ma  - 6329.17 kg-m
 Mu  - 8987.59 Kg-m
CHECK DEFLECTION AT AFTER TENSIONED
 Deflection(Actual)     - 1.29E-02 m
 Deflection(Allowable) - 1.67E-02 m
CHECK DEFLECTION AT USED
 Deflection(Actual)     - 0.00E+00 m
 Deflection(Allowable) - 1.67E-02 m
-----------------------------------------------------


 ROW :   2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ทำซ้ำ ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 ----------
 No of tendon                 =  14  Unit
 Tension per 1 unit tendon = 13958.00 Kg
 Mcr = 10488.15 kg-m
 Ma  = 6329.17 kg-m
 Mu  = 8987.59 Kg-m
CHECK DEFLECTION AT AFTER TENSIONED
 Deflection(Actual)     = 1.29E-03 m
 Deflection(Allowable) = 1.67E-02 m
CHECK DEFLECTION AT USED
 Deflection(Actual)     = 0.00E+00 m
 Deflection(Allowable) = 1.67E-02 m
-----------------------------------------

 COLUMN :   1
 ------------
 No of tendon                 =  14  Unit
 Tension per 1 unit tendon = 13958.00 Kg
 Mcr = 10488.15 kg-m
 Ma  = 6329.17 Kg-m
 Mu  = 8618.48 Kg-m
CHECK DEFLECTION AT USED
 Deflection(Actual)     = 1.29E-03 m
 Deflection(Allowable) = 1.67E-02 m
CHECK DEFLECTION AT AFTER TENSIONED
 Deflection(Actual)     = 0.00E+00 m
 Deflection(Allowable) = 1.67E-02 m
-----------------------------------------

 COLUMN :   2
 ------------
 No of tendon                 =  20  Unit
 Tension per 1 unit tendon = 13958.00 Kg
 Mcr = 20960.30 kg-m
 Ma  = 12650.33 Kg-m
 Mu  = 17236.00 Kg-m
CHECK DEFLECTION AT USED
 Deflection(Actual)     = 1.29E-03 m
 Deflection(Allowable) = 1.67E-02 m
CHECK DEFLECTION AT AFTER TENSIONED
 Deflection(Actual)     = 0.00E+00 m
 Deflection(Allowable) = 1.67E-02 m
-----------------------------------------

-------------------------------------------------
CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD
-------------------------------------------------
```

ROW	ELEMENT	DISTANCE FROM LEFT(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
1	1	0.00E+00	1.30E+00	4.83E+01	2.02E+01
1	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
1	1	1.00E+00	-1.76E+00	2.02E+01	4.83E+01
1	1	1.50E+00	-1.96E+00	2.97E+01	3.88E+01
1	1	2.00E+00	-2.14E+00	2.95E+01	3.90E+01
1	1	2.50E+00	-2.30E+00	2.77E+01	4.08E+01
1	1	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
1	1	3.50E+00	-2.55E+00	1.90E+01	4.95E+01
1	1	4.00E+00	-2.50E+00	1.38E+01	5.47E+01
1	1	4.50E+00	-2.11E+00	1.83E+01	5.02E+01
1	1	5.00E+00	-1.42E+00	8.28E+00	6.03E+01
1	1	5.50E+00	2.15E+00	3.53E+01	3.32E+01
1	1	6.00E+00	2.60E+00	2.60E+01	4.17E+01
1	2	0.00E+00	2.60E+00	2.60E+01	4.17E+01
1	2	5.00E-01	2.24E+00	3.33E+01	3.52E+01
1	2	1.00E+00	1.44E+00	3.33E+01	3.53E+01
1	2	1.50E+00	-1.79E+00	4.94E+00	6.36E+01
1	2	2.00E+00	-2.74E+00	4.79E+00	6.37E+01
1	2	2.50E+00	-2.51E+00	4.70E+00	6.38E+01
1	2	3.00E+00	-2.68E+00	4.67E+00	6.39E+01
1	2	3.50E+00	-2.51E+00	4.78E+00	6.38E+01
1	2	4.00E+00	-2.24E+00	4.79E+00	6.37E+01
1	2	4.50E+00	-1.79E+00	4.94E+00	6.36E+01
1	2	5.00E+00	1.44E+00	3.33E+01	3.53E+01
1	2	5.50E+00	2.24E+00	3.33E+01	3.52E+01
1	2	6.00E+00	2.60E+00	2.60E+01	4.17E+01
1	3	0.00E+00	2.60E+00	2.60E+01	4.17E+01
1	3	5.00E-01	2.15E+00	3.53E+01	3.32E+01
1	3	1.00E+00	-1.42E+00	8.28E+00	6.03E+01
1	3	1.50E+00	-2.11E+00	1.83E+01	5.02E+01
1	3	2.00E+00	-2.50E+00	1.38E+01	5.47E+01
1	3	2.50E+00	-2.55E+00	1.90E+01	4.95E+01
1	3	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
1	3	3.50E+00	-2.30E+00	2.77E+01	4.08E+01
1	3	4.00E+00	-2.14E+00	2.95E+01	3.90E+01
1	3	4.50E+00	-1.96E+00	2.97E+01	3.88E+01
1	3	5.00E+00	-1.76E+00	2.02E+01	4.83E+01
1	3	5.50E+00	-1.54E+00	2.50E+01	4.35E+01
1	3	6.00E+00	1.30E+00	4.83E+01	2.02E+01

ROW	ELEMENT	DISTANCE FROM LEFT(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
2	1	0.00E+00	1.30E+00	4.83E+01	2.02E+01
2	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
2	1	1.00E+00	-1.76E+00	2.02E+01	4.83E+01
2	1	1.50E+00	-1.96E+00	2.97E+01	3.88E+01
2	1	2.00E+00	-2.14E+00	2.95E+01	3.90E+01
2	1	2.50E+00	-2.30E+00	2.77E+01	4.08E+01
2	1	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
2	1	3.50E+00	-2.55E+00	1.90E+01	4.95E+01
2	1	4.00E+00	-2.50E+00	1.38E+01	5.47E+01
2	1	4.50E+00	-2.11E+00	1.83E+01	5.02E+01
2	1	5.00E+00	-1.42E+00	8.28E+00	6.03E+01
2	1	5.50E+00	2.15E+00	3.53E+01	3.32E+01
2	1	6.00E+00	2.60E+00	2.60E+01	4.17E+01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2	2	0.00E+00	2.60E+00	2.60E+01	4.17E+01
2	2	5.00E-01	2.24E+00	3.33E+01	3.52E+01
2	2	1.00E+00	1.44E+00	3.33E+01	3.53E+01
2	2	1.50E+00	-1.79E+00	4.94E+00	6.36E+01
2	2	2.00E+00	-2.24E+00	4.79E+00	6.37E+01
2	2	2.50E+00	-2.51E+00	4.70E+00	6.38E+01
2	2	3.00E+00	-2.60E+00	4.67E+00	6.39E+01
2	2	3.50E+00	-2.51E+00	4.70E+00	6.38E+01
2	2	4.00E+00	-2.24E+00	4.79E+00	6.37E+01
2	2	4.50E+00	-1.79E+00	4.94E+00	6.36E+01
2	2	5.00E+00	1.44E+00	3.33E+01	3.53E+01
2	2	5.50E+00	2.24E+00	3.33E+01	3.52E+01
2	2	6.00E+00	2.60E+00	2.60E+01	4.17E+01
2	3	0.00E+00	2.60E+00	2.60E+01	4.17E+01
2	3	5.00E-01	2.15E+00	3.53E+01	3.32E+01
2	3	1.00E+00	-1.47E+00	8.20E+00	6.03E+01
2	3	1.50E+00	-2.11E+00	1.03E+01	5.02E+01
2	3	2.00E+00	-2.50E+00	1.30E+01	5.47E+01
2	3	2.50E+00	-2.55E+00	1.90E+01	4.95E+01
2	3	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
2	3	3.50E+00	-2.30E+00	2.77E+01	4.00E+01
2	3	4.00E+00	-2.14E+00	2.95E+01	3.90E+01
2	3	4.50E+00	-1.96E+00	2.97E+01	3.80E+01
2	3	5.00E+00	-1.76E+00	2.82E+01	4.03E+01
2	3	5.50E+00	-1.54E+00	2.50E+01	4.35E+01
2	3	6.00E+00	1.30E+00	4.03E+01	2.02E+01

COLUMN ELEMENT DISTANCE FROM TOP(m) POSITION OF TENDON(cm) CHECKING STRESS(kg/cm^2)

				TOP	BOTTOM
1	1	0.00E+00	1.30E+00	4.03E+01	2.02E+01
1	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
1	1	1.00E+00	-1.76E+00	3.41E+01	3.44E+01
1	1	1.50E+00	-1.96E+00	3.06E+01	2.99E+01
1	1	2.00E+00	-2.14E+00	4.14E+01	2.71E+01
1	1	2.50E+00	-2.30E+00	4.25E+01	2.60E+01
1	1	3.00E+00	-2.44E+00	4.20E+01	2.65E+01
1	1	3.50E+00	-2.55E+00	3.90E+01	2.87E+01
1	1	4.00E+00	-2.50E+00	3.75E+01	3.10E+01
1	1	4.50E+00	-2.11E+00	3.70E+01	3.15E+01
1	1	5.00E+00	-1.47E+00	3.78E+01	3.07E+01
1	1	5.50E+00	2.15E+00	6.79E+01	5.94E-01
1	1	6.00E+00	2.60E+00	6.24E+01	6.13E+00

COLUMN ELEMENT DISTANCE FROM TOP(m) POSITION OF TENDON(cm) CHECKING STRESS(kg/cm^2)

				TOP	BOTTOM
2	1	0.00E+00	1.30E+00	4.03E+01	2.02E+01
2	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
2	1	1.00E+00	-1.76E+00	3.41E+01	3.44E+01
2	1	1.50E+00	-1.96E+00	3.06E+01	2.99E+01
2	1	2.00E+00	-2.14E+00	4.14E+01	2.71E+01
2	1	2.50E+00	-2.30E+00	4.25E+01	2.60E+01
2	1	3.00E+00	-2.44E+00	4.20E+01	2.65E+01
2	1	3.50E+00	-2.55E+00	3.90E+01	2.87E+01
2	1	4.00E+00	-2.50E+00	3.75E+01	3.10E+01
2	1	4.50E+00	-2.11E+00	3.70E+01	3.15E+01
2	1	5.00E+00	-1.47E+00	3.78E+01	3.07E+01
2	1	5.50E+00	2.15E+00	6.79E+01	5.94E-01
2	1	6.00E+00	2.60E+00	6.24E+01	6.13E+00

CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD

ROW ELEMENT DISTANCE FROM LEFT(m) POSITION OF TENDON(cm) CHECKING STRESS(kg/cm^2)

				TOP	BOTTOM
1	1	0.00E+00	1.30E+00	4.03E+01	2.02E+01
1	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
1	1	1.00E+00	-1.76E+00	2.02E+01	4.03E+01
1	1	1.50E+00	-1.96E+00	2.97E+01	3.80E+01
1	1	2.00E+00	-2.14E+00	2.95E+01	3.90E+01
1	1	2.50E+00	-2.30E+00	2.77E+01	4.00E+01
1	1	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
1	1	3.50E+00	-2.55E+00	1.90E+01	4.95E+01
1	1	4.00E+00	-2.50E+00	1.30E+01	5.47E+01
1	1	4.50E+00	-2.11E+00	1.03E+01	5.02E+01
1	1	5.00E+00	-1.47E+00	8.20E+00	6.03E+01
1	1	5.50E+00	2.15E+00	3.53E+01	3.32E+01
1	1	6.00E+00	2.60E+00	2.60E+01	4.17E+01
1	2	0.00E+00	2.60E+00	2.60E+01	4.17E+01
1	2	5.00E-01	2.24E+00	3.33E+01	3.52E+01
1	2	1.00E+00	1.44E+00	3.33E+01	3.53E+01
1	2	1.50E+00	-1.79E+00	4.94E+00	6.36E+01
1	2	2.00E+00	-2.24E+00	4.79E+00	6.37E+01
1	2	2.50E+00	-2.51E+00	4.70E+00	6.38E+01
1	2	3.00E+00	-2.60E+00	4.67E+00	6.39E+01
1	2	3.50E+00	-2.51E+00	4.70E+00	6.38E+01
1	2	4.00E+00	-2.24E+00	4.79E+00	6.37E+01
1	2	4.50E+00	-1.79E+00	4.94E+00	6.36E+01
1	2	5.00E+00	1.44E+00	3.33E+01	3.53E+01
1	2	5.50E+00	2.24E+00	3.33E+01	3.52E+01
1	2	6.00E+00	2.60E+00	2.60E+01	4.17E+01
1	3	0.00E+00	2.60E+00	2.60E+01	4.17E+01
1	3	5.00E-01	2.15E+00	3.53E+01	3.32E+01
1	3	1.00E+00	-1.47E+00	8.20E+00	6.03E+01
1	3	1.50E+00	-2.11E+00	1.03E+01	5.02E+01
1	3	2.00E+00	-2.50E+00	1.30E+01	5.47E+01
1	3	2.50E+00	-2.55E+00	1.90E+01	4.95E+01
1	3	3.00E+00	-2.44E+00	2.42E+01	4.43E+01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารนี้ทุกครั้งที่มีการนำไปใช้

ROW	ELEMENT	DISTANCE FROM LEFT(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
1	3	3.50E+00	-2.30E+00	2.77E+01	4.08E+01
1	3	4.00E+00	-2.14E+00	2.95E+01	3.90E+01
1	3	4.50E+00	-1.96E+00	2.97E+01	3.88E+01
1	3	5.00E+00	-1.76E+00	2.82E+01	4.03E+01
1	3	5.50E+00	-1.54E+00	2.50E+01	4.35E+01
1	3	6.00E+00	1.30E+00	4.83E+01	2.02E+01

ROW	ELEMENT	DISTANCE FROM LEFT(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
2	1	0.00E+00	1.30E+00	4.83E+01	2.02E+01
2	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
2	1	1.00E+00	-1.76E+00	2.82E+01	4.03E+01
2	1	1.50E+00	-1.96E+00	2.97E+01	3.88E+01
2	1	2.00E+00	-2.14E+00	2.95E+01	3.90E+01
2	1	2.50E+00	-2.30E+00	2.77E+01	4.08E+01
2	1	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
2	1	3.50E+00	-2.55E+00	1.90E+01	4.95E+01
2	1	4.00E+00	-2.50E+00	1.38E+01	5.47E+01
2	1	4.50E+00	-2.11E+00	1.03E+01	5.82E+01
2	1	5.00E+00	-1.42E+00	8.20E+00	6.03E+01
2	1	5.50E+00	2.15E+00	3.53E+01	3.32E+01
2	1	6.00E+00	2.60E+00	2.60E+01	4.17E+01
2	2	0.00E+00	2.60E+00	2.60E+01	4.17E+01
2	2	5.00E-01	2.24E+00	3.33E+01	3.52E+01
2	2	1.00E+00	1.44E+00	3.33E+01	3.53E+01
2	2	1.50E+00	-1.79E+00	4.94E+00	6.36E+01
2	2	2.00E+00	-2.24E+00	4.79E+00	6.37E+01
2	2	2.50E+00	-2.51E+00	4.70E+00	6.38E+01
2	2	3.00E+00	-2.60E+00	4.67E+00	6.39E+01
2	2	3.50E+00	-2.51E+00	4.70E+00	6.38E+01
2	2	4.00E+00	-2.24E+00	4.79E+00	6.37E+01
2	2	4.50E+00	-1.79E+00	4.94E+00	6.36E+01
2	2	5.00E+00	1.44E+00	3.33E+01	3.53E+01
2	2	5.50E+00	2.24E+00	3.33E+01	3.52E+01
2	2	6.00E+00	2.60E+00	2.60E+01	4.17E+01
2	3	0.00E+00	2.60E+00	2.60E+01	4.17E+01
2	3	5.00E-01	2.15E+00	3.53E+01	3.32E+01
2	3	1.00E+00	-1.42E+00	8.20E+00	6.03E+01
2	3	1.50E+00	-2.11E+00	1.03E+01	5.82E+01
2	3	2.00E+00	-2.50E+00	1.38E+01	5.47E+01
2	3	2.50E+00	-2.55E+00	1.90E+01	4.95E+01
2	3	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
2	3	3.50E+00	-2.30E+00	2.77E+01	4.08E+01
2	3	4.00E+00	-2.14E+00	2.95E+01	3.90E+01
2	3	4.50E+00	-1.96E+00	2.97E+01	3.88E+01
2	3	5.00E+00	-1.76E+00	2.82E+01	4.03E+01
2	3	5.50E+00	-1.54E+00	2.50E+01	4.35E+01
2	3	6.00E+00	1.30E+00	4.83E+01	2.02E+01

COLUMN	ELEMENT	DISTANCE FROM TOP(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
1	1	0.00E+00	1.30E+00	4.83E+01	2.02E+01
1	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
1	1	1.00E+00	-1.76E+00	2.82E+01	4.03E+01
1	1	1.50E+00	-1.96E+00	2.97E+01	3.88E+01
1	1	2.00E+00	-2.14E+00	2.95E+01	3.90E+01
1	1	2.50E+00	-2.30E+00	2.77E+01	4.08E+01
1	1	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
1	1	3.50E+00	-2.55E+00	1.90E+01	4.95E+01
1	1	4.00E+00	-2.50E+00	1.38E+01	5.47E+01
1	1	4.50E+00	-2.11E+00	1.03E+01	5.82E+01
1	1	5.00E+00	-1.42E+00	8.20E+00	6.03E+01
1	1	5.50E+00	2.15E+00	3.53E+01	3.32E+01
1	1	6.00E+00	2.60E+00	2.60E+01	4.17E+01

COLUMN	ELEMENT	DISTANCE FROM TOP(m)	POSITION OF TENDON(cm)	CHECKING STRESS(kg/cm^2) TOP	BOTTOM
2	1	0.00E+00	1.30E+00	4.83E+01	2.02E+01
2	1	5.00E-01	-1.54E+00	2.50E+01	4.35E+01
2	1	1.00E+00	-1.76E+00	2.82E+01	4.03E+01
2	1	1.50E+00	-1.96E+00	2.97E+01	3.88E+01
2	1	2.00E+00	-2.14E+00	2.95E+01	3.90E+01
2	1	2.50E+00	-2.30E+00	2.77E+01	4.08E+01
2	1	3.00E+00	-2.44E+00	2.42E+01	4.43E+01
2	1	3.50E+00	-2.55E+00	1.90E+01	4.95E+01
2	1	4.00E+00	-2.50E+00	1.38E+01	5.47E+01
2	1	4.50E+00	-2.11E+00	1.03E+01	5.82E+01
2	1	5.00E+00	-1.42E+00	8.20E+00	6.03E+01
2	1	5.50E+00	2.15E+00	3.53E+01	3.32E+01
2	1	6.00E+00	2.60E+00	2.60E+01	4.17E+01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 5

บทสรุปและวิจารณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทสรุปและวิจารณ์

เนื่องจากการออกแบบพื้นระบบแผ่นพื้นคอนกรีตอัดแรงเป็นงานที่ค่อนข้างจะซับซ้อนมาก นอกจากนี้ผู้ที่มีความรู้เชี่ยวชาญงานด้านนี้ยังมีอยู่น้อย กลุ่มโครงงานได้วางวัตถุประสงค์ในการทำโครงงานครั้งนี้ เพื่อรวบรวมข้อมูลในการออกแบบที่ถูกต้องนำมาเขียนเป็นโปรแกรมคอมพิวเตอร์เพื่อให้การออกแบบสามารถทำได้ง่ายและสะดวกขึ้น โดยอาศัยโปรแกรมที่มีประสิทธิภาพและมีความยืดหยุ่นในการทำงาน ดังนั้นทางกลุ่มโครงงานจึงจำกัดขอบเขตการทำงานของโปรแกรมเพื่อความเหมาะสมในการทำงาน โดยมีการคำนึงถึงการออกแบบในรูปแบบการใช้งานเพื่อลดความยุ่งยากในการใช้งานจริง

เนื่องจากการวิเคราะห์โดยวิธีโครงเปรียบเทียบถือว่าเป็นการวิเคราะห์โครงข้อแข็งที่มีความต่อเนื่องดังนั้นแผ่นต้องมีความต่อเนื่องอย่างน้อย ยสามช่วงติดต่อกันในแต่ละทิศทางซึ่งเป็นข้อจำกัดของโปรแกรม

ระยะช่วงพื้นที่ออกแบบควรจะมีค่าแตกต่างกันไม่มาก เพราะส่วนใหญ่แผ่นพื้นคอนกรีตอัดแรงจะมีช่วงพื้นที่กว้างและรับน้ำหนักมากอยู่แล้วให้การออกแบบได้ผลลัพธ์ที่ถูกต้องมากที่สุด

นอกจากนี้ปัญหาในการเขียนโปรแกรมการคำนวณ เป็นเรื่องของการออกแบบทางโยธาซึ่งการออกแบบส่วนใหญ่ใช้การตัดสินใจของวิศวกรในการเลือกว่าต้องการจะออกแบบแบบใดยกตัวอย่างเช่นการเลือกเหล็กเสริม โดยปกติแล้วผู้ออกแบบจะต้องเป็นผู้ตัดสินใจว่าจะต้องใช้เหล็กขนาดใดและระยะห่างเท่าใดจึงจะเหมาะสม นอกจากปัญหาด้านการคำนวนแล้วยังมีปัญหาของตัวภาษาที่ใช้เขียนด้วย เนื่องจากภาษา Visual BASIC นั้นยังไม่เป็นที่แพร่หลายมากนักจึงมีแหล่งข้อมูลให้ศึกษาน้อยและต้องใช้เวลาศึกษานานพอสมควร เนื่องจากเวลาในการทำโครงงานนี้มีจำกัดเป็นอย่างมากจึงทำให้ไม่สามารถทำการวางแผนการเขียนโปรแกรมให้เป็นไปโดยมีระบบและระเบียบตามที่ควรจะเป็น ทำให้การเขียนโปรแกรมเป็นไปโดยยากลำบาก

ปัญหาที่พบอีกอย่างก็คือการตรวจสอบและแก้ไขโปรแกรมนั้นเป็นไปด้วยความยากลำบากเนื่องจากไม่สามารถตรวจสอบโปรแกรมทีละขั้นตอนได้ และจากการที่ตัวโปรแกรมมีขนาดใหญ่มีตัวแปรมาก ทำให้ใช้หน่วยความจำสูง

จะเห็นได้ว่าเราสามารถพัฒนาโปรแกรมวิเคราะห์โครงสร้างบนระบบปฏิบัติการ Windows ได้และถ้าหากต้องการโครงสร้างที่มีความละเอียดในการคำนวณที่มากกว่านี้ก็สามารถนำโครงงานนี้ไปพัฒนาขั้นตอนในการคำนวณได้ แต่ก็คงต้องใช้เวลาและความสามารถในการศึกษาเรื่องการเขียนโปรแกรมด้วยภาษา Visual BASIC เพิ่มเติม นอกจากนี้ผู้ทำการออกแบบต้องมีพื้นฐานความรู้และประสบการณ์ในการออกแบบพอสมควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของโปรแกรมนี้คือสามารถแสดงผลเป็นกราฟิก และรายการแสดงผล ซึ่งทำให้สามารถเข้าใจพฤติกรรมของแผ่นพื้นและเห็นภาพเป็นรูปธรรมมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROGRAM LISTING



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMABOUT.FRM - 1

```
Sub cmdOK_Click ()
    frmAbout.Visible = False
    Unload frmAbout
End Sub
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMABOUT.FRM - 1

```
VERSION 2.00
Begin Form frmAbout
   AutoRedraw      =   -1  'True
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   1  'Fixed Single
   Caption         =   "About Program"
   ClientHeight    =   6930
   ClientLeft      =   1665
   ClientTop       =   735
   ClientWidth     =   9570
   ClipControls    =   0   'False
   ControlBox      =   0   'False
   Height          =   7335
   Left            =   1605
   LinkTopic       =   "Form1"
   ScaleHeight     =   6930
   ScaleWidth      =   9570
   Top             =   390
   Width           =   9690
   Begin SSPanel Panel3D5
      BevelOuter       =   0  'None
      Caption          =   "PATCHARAWALAI MAITA 35104"
      Font3D           =   3  'Inset w/light shading
      ForeColor        =   &H00FF0000&
      Height           =   255
      Left             =   240
      TabIndex         =   8
      Top              =   5160
      Width            =   9135
   End
   Begin SSPanel Panel3D4
      BevelOuter       =   0  'None
      Caption          =   "ADVISOR: MR. WIBOOL VOOTIYAN"
      Font3D           =   3  'Inset w/light shading
      ForeColor        =   &H00FF0000&
      Height           =   495
      Left             =   240
      TabIndex         =   7
      Top              =   5640
      Width            =   9255
   End
   Begin SSPanel Panel3D3
      BevelOuter       =   0  'None
      Caption          =   "SOMPONG KONGKRAPAN  35104443"
      Font3D           =   3  'Inset w/light shading
      ForeColor        =   &H00FF0000&
      Height           =   255
      Left             =   240
      TabIndex         =   6
      Top              =   4680
      Width            =   9135
   End
   Begin SSPanel Panel3D2
      BevelOuter       =   0  'None
      Caption          =   "SIAM KUNKATE 35104449"
      Font3D           =   3  'Inset w/light shading
      ForeColor        =   &H00FF0000&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMABOUT.FRM - 2

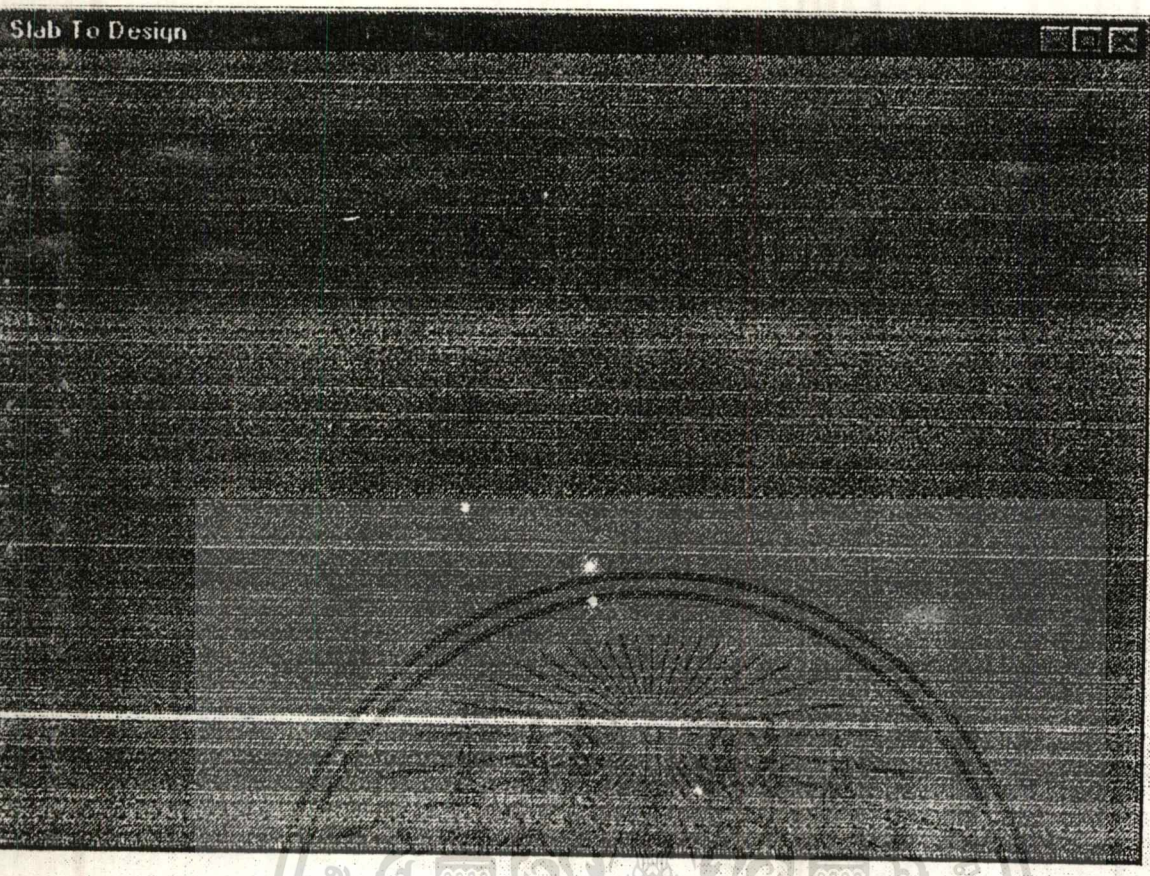
```
      Height          =   495
      Left            =   240
      TabIndex        =   5
      Top             =   4080
      Width           =   9135
   End
   Begin SSPanel Panel3D1
      BevelOuter      =   0  'None
      Caption         =   "DIRECTED BY"
      Font3D          =   3  'Inset w/light shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   495
      Index           =   3
      Left            =   360
      TabIndex        =   4
      Top             =   3480
      Width           =   8895
   End
   Begin SSPanel Panel3D1
      BevelOuter      =   0  'None
      Caption         =   "SLAB DESIGN IN PRE-STRESSED && FLAT SLAB AND FLA
      Font3D          =   3  'Inset w/light shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   12
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   975
      Index           =   2
      Left            =   240
      TabIndex        =   3
      Top             =   2400
      Width           =   8895
   End
   Begin SSPanel Panel3D1
      BevelOuter      =   0  'None
      Caption         =   "PFPS"
      Font3D          =   3  'Inset w/light shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Courier"
      FontSize        =   18.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H000000FF&
      Height          =   735
      Index           =   1
      Left            =   240
      TabIndex        =   2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMABOUT.FRM - 3

```
      Top             =   1440
      Width           =   8895
   End
   Begin SSPanel Panel3D1
      BevelOuter      =   0  'None
      Caption         =   "KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABAN
      Font3D          =   3  'Inset w/light shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   14.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   975
      Index           =   0
      Left            =   360
      TabIndex        =   1
      Top             =   240
      Width           =   8895
   End
   Begin SSCommand cmdOK
      Caption         =   "&OK"
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   12
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   495
      Left            =   4080
      Picture         =   (none)
      TabIndex        =   0
      Top             =   6360
      Width           =   1575
   End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMDATAN.FRM - 1

```
VERSION 2.00
Begin Form frmDataInfo
   AutoRedraw      =   -1  'True
   BackColor       =   &H0080FF80&
   Caption         =   "Slab To Design"
   ClientHeight    =   4485
   ClientLeft      =   2070
   ClientTop       =   2325
   ClientWidth     =   7140
   Height          =   4890
   Icon            =   (Icon)
   Left            =   2010
   LinkTopic       =   "frmDataName"
   MDIChild        =   -1  'True
   ScaleHeight     =   4485
   ScaleWidth      =   7140
   Top             =   1980
   Width           =   7260
   WindowState     =   2  'Maximized
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMDATAN.FRM - 1

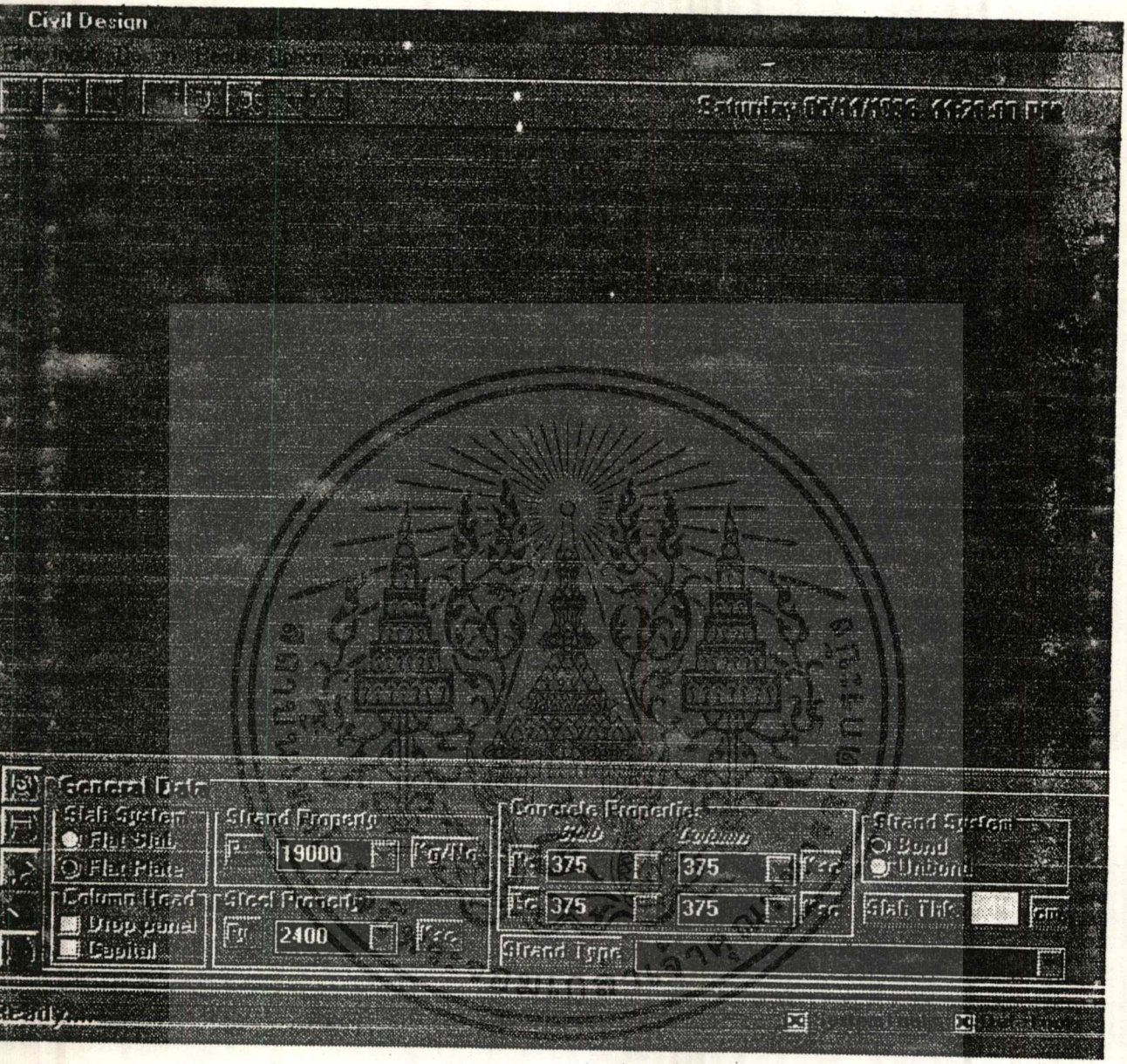
```
Sub Form_QueryUnload (Cancel As Integer, UnloadMode As Integer)
    Ready_ExitProject
End Sub
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"LPSTRESS" Present by Sompong Kongkrapun & Siam Kunkate

Mr. Wibool Wutthiyan : ADVISOR



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FRMMAIN.FRM - 1

VERSION 2.00
Begin MDIForm frmMain
   Caption         =   "Civil Design"
   ClientHeight    =   2655
   ClientLeft      =   -90
   ClientTop       =   3915
   ClientWidth     =   9600
   Height          =   3345
   Icon            =   (Icon)
   Left            =   -150
   LinkTopic       =   "frmMain"
   Top             =   3285
   Width           =   9720
   WindowState     =   2  'Maximized
   Begin SSPanel pnlTextInputData
      Align           =   2  'Align Bottom
      BevelInner      =   1  'Inset
      ForeColor       =   &H000000FF&
      Height          =   2070
      Left            =   0
      Outline         =   -1  'True
      ShadowColor     =   1  'Black
      TabIndex        =   12
      Top             =   -330
      Width           =   9600
      Begin SSFrame fmeTextInputData
         Caption         =   "Node Data"
         Font3D          =   2  'Raised w/heavy shading
         FontBold        =   -1  'True
         FontItalic      =   0   'False
         FontName        =   "MS Sans Serif"
         FontSize        =   9.75
         FontStrikethru  =   0   'False
         FontUnderline   =   0   'False
         ForeColor       =   &H000080000&
         Height          =   1815
         Index           =   0
         Left            =   9480
         ShadowStyle     =   1  'Raised
         TabIndex        =   102
         Top             =   300
         Visible         =   0   'False
         Width           =   8955
         Begin SSPanel pnlLengthUnit
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "m."
            Font3D          =   1  'Raised w/light shading
            ForeColor       =   &H00000080&
            Height          =   315
            Index           =   8
            Left            =   2580
            TabIndex        =   109
            Top             =   1020
            Width           =   315
         End
         Begin TextBox txtNodeLength
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 2

```
      Height          =   315
      Left            =   2160
      TabIndex        =   110
      Text            =   "3"
      Top             =   1020
      Width           =   375
   End
   Begin SSPanel pnlColumnLength
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Length"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Left            =   1440
      TabIndex        =   108
      Top             =   1020
      Width           =   735
   End
   Begin SSPanel pnlLengthUnit
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "m."
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Index           =   0
      Left            =   2580
      TabIndex        =   112
      Top             =   720
      Width           =   315
   End
   Begin SSPanel pnlLengthUnit
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "m."
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Index           =   4
      Left            =   2580
      TabIndex        =   111
      Top             =   420
      Width           =   315
   End
   Begin TextBox txtNodeDepth
      Alignment       =   2  'Center
      Height          =   315
      Left            =   2100
      TabIndex        =   115
      Text            =   "0.30"
      Top             =   720
      Width           =   435
   End
   Begin TextBox txtNodeWidth
      Alignment       =   2  'Center
      Height          =   315
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 3

```
      Left            =   2100
      TabIndex        =   116
      Text            =   "0.30"
      Top             =   420
      Width           =   435
   End
   Begin SSPanel Panel3D3
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Y Dim"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Left            =   1440
      TabIndex        =   113
      Top             =   720
      Width           =   675
   End
   Begin SSPanel Panel3D2
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "X Dim"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Left            =   1440
      TabIndex        =   114
      Top             =   420
      Width           =   675
   End
   Begin TextBox txtGrid
      BorderStyle     =   0  'None
      Height          =   435
      Left            =   5040
      TabIndex        =   143
      Top             =   2280
      Visible         =   0   'False
      Width           =   1335
   End
   Begin SSFrame fmeEdgeYIndex
      Caption         =   "Edge_Y"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   855
      Left            =   3480
      TabIndex        =   139
      Top             =   900
      Width           =   1035
      Begin SSOption optEdgeY
         Caption          =   "Interior"
         Font3D           =   3  'Inset w/light shading
         ForeColor        =   &H00808000&
         Height           =   210
         Index            =   0
         Left             =   60
         TabIndex         =   142
         Top              =   240
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 4

```
            Value           =   -1  'True
            Width           =   915
         End
         Begin SSOption optEdgeY
            Caption         =   "Top"
            Font3D          =   3  'Inset w/light shading
            ForeColor       =   &H00808000&
            Height          =   210
            Index           =   1
            Left            =   60
            TabIndex        =   141
            TabStop         =   0   'False
            Top             =   600
            Width           =   915
         End
         Begin SSOption optEdgeY
            Caption         =   "Bottom"
            Font3D          =   3  'Inset w/light shading
            ForeColor       =   &H00808000&
            Height          =   210
            Index           =   2
            Left            =   60
            TabIndex        =   140
            TabStop         =   0   'False
            Top             =   420
            Width           =   915
         End
      End
      Begin SSFrame fmeNode
         Caption         =   "Position"
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00FF0000&
         Height          =   1455
         Left            =   60
         TabIndex        =   117
         Top             =   300
         Width           =   3375
         Begin SSPanel pnlGen
            Caption         =   "Arr X:Y"
            Font3D          =   1  'Raised w/light shading
            ForeColor       =   &H00000080&
            Height          =   255
            Index           =   0
            Left            =   60
            ShadowColor     =   1  'Black
            TabIndex        =   127
            Top             =   1140
            Width           =   675
         End
         Begin TextBox txtNode
            Alignment       =   2  'Center
            Height          =   315
            Index           =   4
            Left            =   780
            TabIndex        =   129
            Text            =   "1"
            Top             =   1140
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 5

```
      Width           =   315
   End
   Begin CommandButton cmdNodeOK
      BackColor       =   &H00FF00FF&
      Caption         =   "&OK"
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Serif"
      FontSize        =   12
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   975
      Left            =   2820
      TabIndex        =   121
      Top             =   120
      Width           =   495
   End
   Begin SSPanel pnlNodeNo
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "No."
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H000000080&
      Height          =   315
      Left            =   60
      TabIndex        =   138
      Top             =   240
      Width           =   435
   End
   Begin SSPanel pnlNodeCo_X
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "X"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H000000080&
      Height          =   315
      Left            =   60
      TabIndex        =   137
      Top             =   540
      Width           =   435
   End
   Begin SSPanel pnlNodeCo_Y
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Y"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00004080&
      Height          =   315
      Left            =   60
      TabIndex        =   136
      Top             =   840
      Width           =   435
   End
   Begin SSPanel pnlLengthUnit
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "m."
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 6

```
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Index           =   6
      Left            =   1020
      TabIndex        =   131
      Top             =   540
      Width           =   315
   End
   Begin SSPanel pnlLengthUnit
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "m."
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   315
      Index           =   7
      Left            =   1020
      TabIndex        =   130
      Top             =   840
      Width           =   315
   End
   Begin TextBox txtNode
      Alignment       =   2  'Center
      Height          =   315
      Index           =   5
      Left            =   1320
      TabIndex        =   128
      Text            =   "1"
      Top             =   1140
      Width           =   315
   End
   Begin SSPanel pnlComment
      Caption         =   ":"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   255
      Index           =   0
      Left            =   1140
      ShadowColor     =   1  'Black
      TabIndex        =   126
      Top             =   1200
      Width           =   135
   End
   Begin SSPanel pnlComment
      Caption         =   "X:Y"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000080&
      Height          =   255
      Index           =   1
      Left            =   1680
      ShadowColor     =   1  'Black
      TabIndex        =   125
      Top             =   1140
      Width           =   315
   End
   Begin TextBox txtNode
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 7

```
      Alignment       =   2  'Center
      Height          =   315
      Index           =   6
      Left            =   2100
      TabIndex        =   124
      Text            =   "0"
      Top             =   1140
      Width           =   435
   End
   Begin SSPanel pnlComment
      Caption         =   ":"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   255
      Index           =   2
      Left            =   2580
      ShadowColor     =   1  'Black
      TabIndex        =   123
      Top             =   1140
      Width           =   135
   End
   Begin TextBox txtNode
      Alignment       =   2  'Center
      Height          =   315
      Index           =   7
      Left            =   2760
      TabIndex        =   122
      Text            =   "0"
      Top             =   1140
      Width           =   435
   End
   Begin TextBox txtNode
      Alignment       =   2  'Center
      Height          =   315
      Index           =   1
      Left            =   540
      TabIndex        =   120
      Top             =   240
      Width           =   435
   End
   Begin TextBox txtNode
      Alignment       =   2  'Center
      Height          =   315
      Index           =   2
      Left            =   540
      TabIndex        =   119
      Text            =   "0"
      Top             =   540
      Width           =   435
   End
   Begin TextBox txtNode
      Alignment       =   2  'Center
      Height          =   315
      HideSelection   =   0   'False
      Index           =   3
      Left            =   540
      TabIndex        =   118
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 8

```
      Text            =   "0"
      Top             =   840
      Width           =   435
   End
   Begin SSPanel pnlNodeSupport
      BackColor       =   &H00FFFF00&
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Support Type"
      ForeColor       =   &H00FF0000&
      Height          =   315
      Left            =   480
      TabIndex        =   132
      Top             =   600
      Visible         =   0   'False
      Width           =   1275
   End
   Begin SSOption optNodeSupport
      Caption         =   "Pin"
      ForeColor       =   &H00808000&
      Height          =   210
      Index           =   0
      Left            =   540
      TabIndex        =   135
      Top             =   300
      Value           =   -1  'True
      Visible         =   0   'False
      Width           =   795
   End
   Begin SSOption optNodeSupport
      Caption         =   "Fixed"
      ForeColor       =   &H00808000&
      Height          =   210
      Index           =   1
      Left            =   600
      TabIndex        =   134
      TabStop         =   0   'False
      Top             =   300
      Visible         =   0   'False
      Width           =   735
   End
   Begin SSOption optNodeSupport
      Caption         =   "Free"
      ForeColor       =   &H00808000&
      Height          =   210
      Index           =   2
      Left            =   600
      TabIndex        =   133
      TabStop         =   0   'False
      Top             =   300
      Visible         =   0   'False
      Width           =   735
   End
End
Begin Grid grdNode
   BackColor       =   &H00FFFFFF&
   Cols            =   8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 9

```
      ForeColor       =   &H000000FF&
      GridLineWidth   =   2
      Height          =   1575
      Left            =   4560
      Rows            =   100
      TabIndex        =   107
      Top             =   180
      Width           =   4335
   End
   Begin SSFrame Frame3D1
      Caption         =   "Edge_X"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   795
      Left            =   3480
      TabIndex        =   103
      Top             =   120
      Width           =   1035
      Begin SSOption optEdgeX
         Caption         =   "Interior"
         Font3D          =   3  'Inset w/light shading
         ForeColor       =   &H00808000&
         Height          =   210
         Index           =   0
         Left            =   60
         TabIndex        =   106
         Top             =   240
         Value           =   -1  'True
         Width           =   915
      End
      Begin SSOption optEdgeX
         Caption         =   "Left"
         Font3D          =   3  'Inset w/light shading
         ForeColor       =   &H00808000&
         Height          =   210
         Index           =   1
         Left            =   60
         TabIndex        =   105
         TabStop         =   0   'False
         Top             =   420
         Width           =   735
      End
      Begin SSOption optEdgeX
         Caption         =   "Right"
         Font3D          =   3  'Inset w/light shading
         ForeColor       =   &H00808000&
         Height          =   210
         Index           =   2
         Left            =   60
         TabIndex        =   104
         TabStop         =   0   'False
         Top             =   600
         Width           =   795
      End
   End
End
Begin SSFrame fmeTextInputData
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 10

```
   Caption          =   "Load Data"
   Font3D           =   2  'Raised w/heavy shading
   FontBold         =   -1  'True
   FontItalic       =   0   'False
   FontName         =   "MS Sans Serif"
   FontSize         =   9.75
   FontStrikethru   =   0   'False
   FontUnderline    =   0   'False
   ForeColor        =   &H00FF0000&
   Height           =   1815
   Index            =   1
   Left             =   9360
   ShadowStyle      =   1  'Raised
   TabIndex         =   70
   Top              =   540
   Visible          =   0   'False
   Width            =   8835
   Begin SSPanel pnlLoadFactor
      BevelInner       =   2  'Raised
      BevelOuter       =   0  'None
      Caption          =   "Dead Load Factor"
      Font3D           =   3  'Inset w/light shading
      ForeColor        =   &H00FF0000&
      Height           =   315
      Index            =   1
      Left             =   5340
      TabIndex         =   157
      Top              =   1140
      Width            =   1935
   End
   Begin TextBox txtLLFactor
      Height           =   315
      Left             =   7440
      TabIndex         =   152
      Text             =   "1.7"
      Top              =   420
      Width            =   735
   End
   Begin SSPanel pnlLoadFactor
      BevelInner       =   2  'Raised
      BevelOuter       =   0  'None
      Caption          =   "Live Load Factor"
      Font3D           =   3  'Inset w/light shading
      ForeColor        =   &H00FF0000&
      Height           =   315
      Index            =   0
      Left             =   5340
      TabIndex         =   154
      Top              =   420
      Width            =   1935
   End
   Begin SSPanel pnlLoadUnit
      BevelInner       =   2  'Raised
      BevelOuter       =   0  'None
      Caption          =   "Kg/m2"
      Font3D           =   1  'Raised w/light shading
      ForeColor        =   &H00800000&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 11

```
      Height          =   315
      Index           =   5
      Left            =   4200
      TabIndex        =   150
      Top             =   420
      Width           =   675
   End
   Begin SSPanel pnlLL
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Live Load"
      Font3D          =   3  'Inset w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   315
      Index           =   1
      Left            =   540
      TabIndex        =   149
      Top             =   420
      Width           =   1995
   End
   Begin TextBox txtLL
      Height          =   315
      Left            =   2820
      TabIndex        =   147
      Text            =   "300"
      Top             =   420
      Width           =   1095
   End
   Begin SSPanel pnlLoadUnit
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Kg/m2"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00800000&
      Height          =   315
      Index           =   4
      Left            =   4200
      TabIndex        =   71
      Top             =   1320
      Width           =   675
   End
   Begin TextBox txtSL
      Height          =   315
      Left            =   2820
      TabIndex        =   73
      Text            =   "300"
      Top             =   1320
      Width           =   1095
   End
   Begin TextBox txtDLFactor
      Alignment       =   2  'Center
      Height          =   645
      Left            =   7440
      TabIndex        =   153
      Text            =   "1.4"
      Top             =   960
      Width           =   735
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 12

```
End
Begin SSPanel pnlLoadUnit
   BevelInner      =   2  'Raised
   BevelOuter      =   0  'None
   Caption         =   "Kg/m2"
   Font3D          =   1  'Raised w/light shading
   ForeColor       =   &H00800000&
   Height          =   315
   Index           =   6
   Left            =   4200
   TabIndex        =   151
   Top             =   960
   Width           =   675
End
Begin TextBox txtDL
   BackColor       =   &H00000000&
   Enabled         =   0   'False
   ForeColor       =   &H00FFFFFF&
   Height          =   315
   Left            =   2820
   TabIndex        =   146
   Text            =   "0"
   Top             =   960
   Width           =   1095
End
Begin SSPanel Panel3D4
   BevelInner      =   2  'Raised
   BevelOuter      =   0  'None
   Caption         =   "Super Impose Load"
   Font3D          =   3  'Inset w/light shading
   ForeColor       =   &H00FF0000&
   Height          =   315
   Left            =   540
   TabIndex        =   72
   Top             =   1320
   Width           =   1995
End
Begin SSPanel pnlDL
   BevelInner      =   2  'Raised
   BevelOuter      =   0  'None
   Caption         =   "Dead Load"
   Font3D          =   3  'Inset w/light shading
   ForeColor       =   &H00FF0000&
   Height          =   315
   Index           =   1
   Left            =   540
   TabIndex        =   148
   Top             =   960
   Width           =   1995
End
Begin SSPanel Panel3D6
   BevelInner      =   2  'Raised
   BevelOuter      =   0  'None
   ForeColor       =   &H00FF0000&
   Height          =   555
   Left            =   240
   TabIndex        =   156
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 13

```
      Top             =   300
      Width           =   8355
   End
   Begin SSPanel Panel3D5
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      ForeColor       =   &H00FF0000&
      Height          =   915
      Left            =   240
      TabIndex        =   155
      Top             =   840
      Width           =   8355
   End
   Begin SSFrame fmeLoadType
      Caption         =   "Type"
      ForeColor       =   &H00800080&
      Height          =   1095
      Left            =   7980
      TabIndex        =   97
      Top             =   1500
      Visible         =   0   'False
      Width           =   1275
      Begin SSOption optLoadType
         Caption      =   "Point"
         ForeColor    =   &H00808000&
         Height       =   210
         Index        =   0
         Left         =   60
         TabIndex     =   101
         TabStop      =   0   'False
         Top          =   300
         Width        =   735
      End
      Begin SSOption optLoadType
         Caption      =   "Uniform"
         ForeColor    =   &H00808000&
         Height       =   210
         Index        =   1
         Left         =   60
         TabIndex     =   100
         Top          =   480
         Value        =   -1  'True
         Width        =   975
      End
      Begin SSOption optLoadType
         Caption      =   "Trapizodal"
         ForeColor    =   &H00808000&
         Height       =   210
         Index        =   2
         Left         =   60
         TabIndex     =   99
         TabStop      =   0   'False
         Top          =   660
         Width        =   1155
      End
      Begin SSOption optLoadType
         Caption      =   "Moment "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 14

```
            ForeColor       =   &H00808000&
            Height          =   210
            Index           =   3
            Left            =   60
            TabIndex        =   98
            TabStop         =   0   'False
            Top             =   840
            Width           =   1035
         End
      End
      Begin SSFrame fmeLoadPosition
         Caption         =   "Position"
         ForeColor       =   &H00800080&
         Height          =   1695
         Left            =   8040
         TabIndex        =   84
         Top             =   1560
         Visible         =   0   'False
         Width           =   1635
         Begin SSPanel pnlLoadX1
            BackColor       =   &H00FFFF00&
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "X1 :"
            ForeColor       =   &H00FF00FF&
            Height          =   315
            Left            =   60
            TabIndex        =   96
            Top             =   240
            Width           =   495
         End
         Begin SSPanel pnlLoadY2
            BackColor       =   &H00FFFF00&
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "Y2 :"
            ForeColor       =   &H00FF00FF&
            Height          =   315
            Left            =   60
            TabIndex        =   95
            Top             =   1260
            Width           =   495
         End
         Begin SSPanel pnlLoadX2
            BackColor       =   &H00FFFF00&
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "X2 :"
            ForeColor       =   &H00FF00FF&
            Height          =   315
            Left            =   60
            TabIndex        =   94
            Top             =   900
            Width           =   495
         End
         Begin SSPanel pnlLoadY1
            BackColor       =   &H00FFFF00&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 15

```
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Y1 :"
      ForeColor       =   &H00FF00FF&
      Height          =   315
      Left            =   60
      TabIndex        =   93
      Top             =   540
      Width           =   495
   End
   Begin TextBox txtLoadX1
      Height          =   315
      Left            =   540
      TabIndex        =   92
      Top             =   240
      Width           =   675
   End
   Begin TextBox txtLoadY1
      Height          =   315
      Left            =   540
      TabIndex        =   91
      Top             =   600
      Width           =   675
   End
   Begin TextBox txtLoadX2
      Height          =   315
      Left            =   540
      TabIndex        =   90
      Top             =   960
      Width           =   675
   End
   Begin TextBox txtLoadY2
      Height          =   315
      Left            =   540
      TabIndex        =   89
      Top             =   1320
      Width           =   675
   End
   Begin SSPanel pnlLengthUnit
      BackColor       =   &H00FFFF00&
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "m."
      ForeColor       =   &H000000FF&
      Height          =   315
      Index           =   1
      Left            =   1260
      TabIndex        =   88
      Top             =   240
      Width           =   315
   End
   Begin SSPanel pnlLengthUnit
      BackColor       =   &H00FFFF00&
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "m."
      ForeColor       =   &H000000FF&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 16

```
            Height          =   315
            Index           =   2
            Left            =   1260
            TabIndex        =   87
            Top             =   600
            Width           =   315
         End
         Begin SSPanel pnlLengthUnit
            BackColor       =   &H00FFFF00&
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "m."
            ForeColor       =   &H000000FF&
            Height          =   315
            Index           =   3
            Left            =   1260
            TabIndex        =   86
            Top             =   960
            Width           =   315
         End
         Begin SSPanel pnlSlabFc_U
            BackColor       =   &H00FFFF00&
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "m."
            ForeColor       =   &H000000FF&
            Height          =   315
            Index           =   4
            Left            =   1260
            TabIndex        =   85
            Top             =   1320
            Width           =   315
         End
      End
      Begin SSFrame fmeLoadMagnitude
         Caption         =   "Magnitude"
         ForeColor       =   &H00008000&
         Height          =   1455
         Left            =   7080
         TabIndex        =   77
         Top             =   1620
         Visible         =   0  'False
         Width           =   2055
         Begin TextBox txtLoadEnd
            Height          =   315
            Left            =   1980
            TabIndex        =   82
            Top             =   -60
            Visible         =   0   'False
            Width           =   735
         End
         Begin SSPanel pnlLoadw1
            BackColor       =   &H00FFFF00&
            BevelInner      =   2  'Raised
            BevelOuter      =   0  'None
            Caption         =   "w1"
            ForeColor       =   &H00FF00FF&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 17

```
      Height          =   315
      Left            =   2040
      TabIndex        =   81
      Top             =   0
      Visible         =   0   'False
      Width           =   375
   End
   Begin SSPanel pnlw2
      BackColor       =   &H00FFFF00&
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "w2"
      ForeColor       =   &H00FF00FF&
      Height          =   315
      Left            =   -60
      TabIndex        =   80
      Top             =   1320
      Visible         =   0   'False
      Width           =   375
   End
   Begin SSPanel pnlLoadUnit
      BackColor       =   &H00FFFF00&
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Kg"
      ForeColor       =   &H00FF00FF&
      Height          =   315
      Index           =   1
      Left            =   1980
      TabIndex        =   78
      Top             =   -120
      Visible         =   0   'False
      Width           =   435
   End
   Begin TextBox txtLoadStart
      Height          =   315
      Left            =   1920
      TabIndex        =   83
      Top             =   -120
      Visible         =   0   'False
      Width           =   735
   End
   Begin SSPanel pnlLoadUnit
      BackColor       =   &H00FFFF00&
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Kg"
      ForeColor       =   &H00FF00FF&
      Height          =   315
      Index           =   0
      Left            =   -300
      TabIndex        =   79
      Top             =   1260
      Visible         =   0   'False
      Width           =   435
   End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 18

```
      Begin Grid grdLoad
         Cols            =   8
         ForeColor       =   &H00FF0000&
         Height          =   1575
         Left            =   8760
         Rows            =   100
         TabIndex        =   76
         Top             =   1740
         Visible         =   0   'False
         Width           =   3975
      End
      Begin SSPanel pnlLoadNo
         BackColor       =   &H00FFFF00&
         BevelInner      =   2  'Raised
         Caption         =   "No."
         ForeColor       =   &H00FF00FF&
         Height          =   375
         Left            =   8520
         TabIndex        =   75
         Top             =   1500
         Visible         =   0   'False
         Width           =   555
      End
      Begin TextBox txtLoadNo
         Height          =   315
         Left            =   8340
         TabIndex        =   74
         Top             =   1380
         Visible         =   0   'False
         Width           =   675
      End
   End
   Begin SSFrame fmeTextInputData
      Caption         =   "Element Data"
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00808000&
      Height          =   1815
      Index           =   2
      Left            =   9240
      ShadowStyle     =   1  'Raised
      TabIndex        =   68
      Top             =   540
      Visible         =   0   'False
      Width           =   8835
      Begin Grid grdElement
         Cols            =   9
         ForeColor       =   &H000000FF&
         GridLineWidth   =   2
         Height          =   1455
         Left            =   60
         Rows            =   5000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 19

```
      TabIndex        =   69
      Top             =   300
      Width           =   8715
   End
End
Begin SSFrame fmeTextInputData
   Caption         =   "Information Data"
   Font3D          =   2  'Raised w/heavy shading
   FontBold        =   -1  'True
   FontItalic      =   0   'False
   FontName        =   "MS Sans Serif"
   FontSize        =   9.75
   FontStrikethru  =   0   'False
   FontUnderline   =   0   'False
   ForeColor       =   &H00FF0000&
   Height          =   1815
   Index           =   4
   Left            =   9000
   ShadowStyle     =   1  'Raised
   TabIndex        =   53
   Top             =   480
   Visible         =   0   'False
   Width           =   8955
   Begin SSPanel pnlFileName
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "FileName"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   67
      Top             =   300
      Width           =   1275
   End
   Begin TextBox txtFileName
      BackColor       =   &H00E0FFFF&
      ForeColor       =   &H00008000&
      Height          =   315
      HideSelection   =   0   'False
      Left            =   1440
      TabIndex        =   66
      Top             =   360
      Width           =   1875
   End
   Begin SSPanel pnlProjectName
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Project Name"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   65
      Top             =   660
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 20

```
      Width           =   1275
   End
   Begin TextBox txtProjectName
      BackColor       =   &H00E0FFFF&
      ForeColor       =   &H00008000&
      Height          =   315
      HideSelection   =   0   'False
      Left            =   1440
      TabIndex        =   64
      Top             =   720
      Width           =   1875
   End
   Begin SSPanel pnlTime
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Time"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   63
      Top             =   1020
      Width           =   1275
   End
   Begin TextBox txtTime
      BackColor       =   &H00E0FFFF&
      ForeColor       =   &H00008000&
      Height          =   315
      HideSelection   =   0   'False
      Left            =   1440
      TabIndex        =   62
      Top             =   1080
      Width           =   1875
   End
   Begin SSPanel pnlDate
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Date"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   61
      Top             =   1440
      Width           =   1275
   End
   Begin TextBox txtDate
      BackColor       =   &H00E0FFFF&
      ForeColor       =   &H00008000&
      Height          =   315
      HideSelection   =   0   'False
      Left            =   1440
      TabIndex        =   60
      Top             =   1440
      Width           =   1875
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 21

```
    End
    Begin SSFrame fmeInfomation
       Caption          =   "&Information"
       Font3D           =   1  'Raised w/light shading
       ForeColor        =   &H00808000&
       Height           =   1575
       Left             =   7080
       TabIndex         =   58
       Top              =   180
       Width            =   1695
       Begin TextBox txtInformation
          BackColor        =   &H00E0FFFF&
          ForeColor        =   &H00008000&
          Height           =   1275
          HideSelection    =   0   'False
          Left             =   60
          MultiLine        =   -1  'True
          TabIndex         =   59
          Top              =   240
          Width            =   1515
       End
    End
    Begin Timer tmrTime
       Interval         =   1000
       Left             =   7920
       Top              =   600
    End
    Begin SSPanel pnlAddress
       Alignment        =   1  'Left Justify - MIDDLE
       BevelInner       =   2  'Raised
       BevelOuter       =   0  'None
       Caption          =   "Address"
       Font3D           =   1  'Raised w/light shading
       ForeColor        =   &H00808000&
       Height           =   855
       Left             =   3600
       TabIndex         =   57
       Top              =   300
       Width            =   975
    End
    Begin TextBox txtAddress
       BackColor        =   &H00E0FFFF&
       ForeColor        =   &H00008000&
       Height           =   855
       HideSelection    =   0   'False
       Left             =   4740
       MultiLine        =   -1  'True
       TabIndex         =   56
       Top              =   360
       Width            =   1875
    End
    Begin SSPanel pnlEngineer
       Alignment        =   1  'Left Justify - MIDDLE
       BevelInner       =   2  'Raised
       BevelOuter       =   0  'None
       Caption          =   "Engineer"
       Font3D           =   1  'Raised w/light shading
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 22

```
         ForeColor        =   &H00808000&
         Height           =   375
         Left             =   3600
         TabIndex         =   55
         Top              =   1200
         Width            =   975
      End
      Begin TextBox txtEngineer
         BackColor        =   &H00E0FFFF&
         ForeColor        =   &H00008000&
         Height           =   315
         HideSelection    =   0   'False
         Left             =   4740
         TabIndex         =   54
         Top              =   1260
         Width            =   1875
      End
   End
   Begin SSPanel pnlSelectInput
      BevelWidth       =   0
      BorderWidth      =   1
      ForeColor        =   &H000000FF&
      Height           =   1815
      Left             =   120
      RoundedCorners   =   0   'False
      TabIndex         =   52
      Top              =   120
      Width            =   435
      Begin SSRibbon gphSelectInput
         AutoSize         =   0  'None
         Height           =   375
         Index            =   0
         Left             =   60
         PictureDisabled  =   (none)
         PictureDn        =   (Bitmap)
         PictureUp        =   (Bitmap)
         Top              =   0
         Width            =   375
      End
      Begin SSRibbon gphSelectInput
         AutoSize         =   0  'None
         Height           =   375
         Index            =   1
         Left             =   60
         PictureDisabled  =   (none)
         PictureDn        =   (Bitmap)
         PictureUp        =   (Bitmap)
         Top              =   720
         Width            =   375
      End
      Begin SSRibbon gphSelectInput
         AutoSize         =   0  'None
         Height           =   375
         Index            =   2
         Left             =   60
         PictureDisabled  =   (none)
         PictureDn        =   (Bitmap)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 23

```
            PictureUp          =   (Bitmap)
            Top                =   360
            Width              =   375
         End
         Begin SSRibbon gphSelectInput
            AutoSize           =   0  'None
            Height             =   375
            Index              =   3
            Left               =   60
            PictureDisabled    =   (none)
            PictureDn          =   (Bitmap)
            PictureUp          =   (Bitmap)
            Top                =   1080
            Width              =   375
         End
         Begin SSRibbon gphSelectInput
            AutoSize           =   0  'None
            Height             =   375
            Index              =   4
            Left               =   60
            PictureDisabled    =   (none)
            PictureDn          =   (Bitmap)
            PictureUp          =   (Bitmap)
            Top                =   1440
            Width              =   375
         End
      End
      Begin SSFrame fmeTextInputData
         Caption            =   "General Data"
         Font3D             =   2  'Raised w/heavy shading
         FontBold           =   -1  'True
         FontItalic         =   0  'False
         FontName           =   "MS Sans Serif"
         FontSize           =   9.75
         FontStrikethru     =   0  'False
         FontUnderline      =   0  'False
         ForeColor          =   &H00FF00FF&
         Height             =   1815
         Index              =   3
         Left               =   9120
         ShadowStyle        =   1  'Raised
         TabIndex           =   13
         Top                =   180
         Visible            =   0  'False
         Width              =   8955
         Begin SSFrame fmeStrand
            Caption            =   "Strand Property"
            Font3D             =   1  'Raised w/light shading
            ForeColor          =   &H00004000&
            Height             =   735
            Left               =   1440
            TabIndex           =   14
            Top                =   300
            Width              =   2415
            Begin ComboBox cmbSFs_
               BackColor          =   &H00FF0000&
               ForeColor          =   &H00FFFFFF&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 24

```
      Height          =   315
      Left            =   540
      TabIndex        =   17
      Text            =   "19000"
      Top             =   300
      Width           =   1095
   End
   Begin SSPanel pnlFyStrand
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "P"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   16
      Top             =   240
      Width           =   435
   End
   Begin SSPanel pnlFs_U
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Kg/No."
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   375
      Index           =   1
      Left            =   1680
      TabIndex        =   15
      Top             =   240
      Width           =   675
   End
End
Begin SSFrame fmeSlabSystem
   Caption         =   "Slab System"
   Font3D          =   1  'Raised w/light shading
   ForeColor       =   &H00004000&
   Height          =   735
   Left            =   60
   TabIndex        =   49
   Top             =   300
   Width           =   1335
   Begin SSOption optSystem
      Caption         =   "Flat Plate"
      Font3D          =   3  'Inset w/light shading
      ForeColor       =   &H00008000&
      Height          =   210
      Index           =   0
      Left            =   60
      TabIndex        =   51
      Top             =   480
      Value           =   -1  'True
      Width           =   1095
   End
   Begin SSOption optSystem
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 25

```
            Caption         =   "Flat Slab"
            Font3D          =   3  'Inset w/light shading
            ForeColor       =   &H00008000&
            Height          =   210
            Index           =   1
            Left            =   60
            TabIndex        =   50
            TabStop         =   0   'False
            Top             =   240
            Width           =   1095
         End
      End
      Begin SSFrame fmeDropOrCapital
         Caption         =   "Column Head"
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00004000&
         Height          =   735
         Left            =   60
         TabIndex        =   46
         Top             =   1020
         Width           =   1335
         Begin SSCheck chkCHead
            Caption         =   "Capital"
            Enabled         =   0   'False
            Font3D          =   3  'Inset w/light shading
            ForeColor       =   &H00008000&
            Height          =   195
            Index           =   1
            Left            =   60
            TabIndex        =   48
            Top             =   480
            Width           =   1095
         End
         Begin SSCheck chkCHead
            Caption         =   "Drop panel"
            Enabled         =   0   'False
            Font3D          =   3  'Inset w/light shading
            ForeColor       =   &H00008000&
            Height          =   240
            Index           =   0
            Left            =   60
            TabIndex        =   47
            Top             =   240
            Width           =   1215
         End
      End
      Begin SSPanel pnlThick_U
         BevelInner      =   2  'Raised
         BevelOuter      =   0  'None
         Caption         =   "cm."
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00FF0000&
         Height          =   375
         Index           =   1
         Left            =   8460
         TabIndex        =   45
         Top             =   960
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 26

```
      Width           =   435
   End
   Begin TextBox txtThick
      BackColor       =   &H00FFFFFF&
      Enabled         =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   315
      Left            =   7980
      TabIndex        =   44
      Top             =   960
      Width           =   435
   End
   Begin SSPanel pnlThick
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Slab Thk."
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   7020
      TabIndex        =   43
      Top             =   960
      Width           =   975
   End
   Begin SSFrame fmeReinforce
      Caption         =   "Steel Property"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00400000&
      Height          =   735
      Left            =   1440
      TabIndex        =   39
      Top             =   1020
      Width           =   2415
      Begin SSPanel pnlFs_U
         Alignment       =   1  'Left Justify - MIDDLE
         BevelInner      =   2  'Raised
         BevelOuter      =   0  'None
         Caption         =   "Ksc"
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00FF0000&
         Height          =   375
         Index           =   0
         Left            =   1740
         TabIndex        =   42
         Top             =   240
         Width           =   495
      End
      Begin SSPanel pnlFs_
         Alignment       =   1  'Left Justify - MIDDLE
         BevelInner      =   2  'Raised
         BevelOuter      =   0  'None
         Caption         =   "Fy"
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00808000&
         Height          =   375
         Left            =   60
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 27

```
      TabIndex        =   41
      Top             =   210
      Width           =   435
   End
   Begin ComboBox cmbFs_
      BackColor       =   &H00FF0000&
      ForeColor       =   &H00FFFFFF&
      Height          =   315
      Left            =   540
      TabIndex        =   40
      Text            =   "2400"
      Top             =   300
      Width           =   1095
   End
End
Begin SSFrame fmeSlab
   Caption         =   "Concrete Properties"
   Font3D          =   1  'Raised w/light shading
   ForeColor       =   &H00004000&
   Height          =   1215
   Left            =   3900
   TabIndex        =   29
   Top             =   180
   Width           =   3075
   Begin SSPanel pnlMP
      Alignment       =   1  'Left Justify - MIDDLE
      BevelOuter      =   0  'None
      Caption         =   "  Slab          Column"
      Font3D          =   3  'Inset w/light shading
      FontBold        =   -1  'True
      FontItalic      =   -1  'True
      FontName        =   "MS Sans Serif"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H000080000&
      Height          =   255
      Left            =   360
      TabIndex        =   38
      Top             =   225
      Width           =   1875
   End
   Begin SSPanel pnlSlabEcU
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Ksc"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   390
      Left            =   2580
      TabIndex        =   37
      Top             =   780
      Width           =   435
   End
   Begin SSPanel pnlSlabFc_U
      Alignment       =   1  'Left Justify - MIDDLE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 28

```
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Ksc"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   375
      Index           =   0
      Left            =   2580
      TabIndex        =   36
      Top             =   420
      Width           =   435
   End
   Begin SSPanel pnlSlabModulus
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "Ec"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   35
      Top             =   780
      Width           =   375
   End
   Begin SSPanel pnlSlabFc_
      Alignment       =   1  'Left Justify - MIDDLE
      BevelInner      =   2  'Raised
      BevelOuter      =   0  'None
      Caption         =   "f'c"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00808000&
      Height          =   375
      Left            =   60
      TabIndex        =   34
      Top             =   420
      Width           =   375
   End
   Begin ComboBox cmbSlabFc_
      BackColor       =   &H00FF0000&
      ForeColor       =   &H00FFFFFF&
      Height          =   315
      Left            =   420
      TabIndex        =   33
      Text            =   "375"
      Top             =   480
      Width           =   1035
   End
   Begin ComboBox cmbColumnFc_
      BackColor       =   &H00FF0000&
      ForeColor       =   &H00FFFFFF&
      Height          =   315
      Left            =   1560
      TabIndex        =   32
      Text            =   "375"
      Top             =   480
      Width           =   1035
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 29

```
        End
        Begin ComboBox cmbSlabEc
           BackColor       =   &H00FF0000&
           ForeColor       =   &H00FFFFFF&
           Height          =   315
           Left            =   420
           TabIndex        =   31
           Text            =   "375"
           Top             =   840
           Width           =   1035
        End
        Begin ComboBox cmbColumnEc
           BackColor       =   &H00FF0000&
           ForeColor       =   &H00FFFFFF&
           Height          =   315
           Left            =   1560
           TabIndex        =   30
           Text            =   "375"
           Top             =   840
           Width           =   1035
        End
     End
     Begin SSFrame fmeLenghtUnit
        Caption         =   "Length Unit"
        Enabled         =   0   'False
        ForeColor       =   &H00800080&
        Height          =   735
        Left            =   1440
        TabIndex        =   26
        Top             =   240
        Visible         =   0   'False
        Width           =   1275
        Begin SSOption optLengthUnit
           Caption         =   "Meter"
           ForeColor       =   &H00008000&
           Height          =   210
           Index           =   4
           Left            =   60
           TabIndex        =   28
           Top             =   240
           Value           =   -1  'True
           Width           =   1155
        End
        Begin SSOption optLengthUnit
           Caption         =   "Centimeter"
           ForeColor       =   &H00008000&
           Height          =   210
           Index           =   5
           Left            =   60
           TabIndex        =   27
           TabStop         =   0   'False
           Top             =   480
           Width           =   1155
        End
     End
     Begin SSFrame fmeLoadUnit
        Caption         =   "Load Unit"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 30

```
      Enabled         =   0   'False
      ForeColor       =   &H00800080&
      Height          =   735
      Left            =   2760
      TabIndex        =   23
      Top             =   240
      Visible         =   0   'False
      Width           =   1095
      Begin SSOption optLoadUnit
         Caption          =   "Kilogram"
         ForeColor        =   &H00008000&
         Height           =   210
         Index            =   4
         Left             =   60
         TabIndex         =   25
         Top              =   240
         Value            =   -1  'True
         Width            =   975
      End
      Begin SSOption optLoadUnit
         Caption          =   "Newton"
         ForeColor        =   &H00008000&
         Height           =   210
         Index            =   5
         Left             =   60
         TabIndex         =   24
         TabStop          =   0   'False
         Top              =   480
         Width            =   975
      End
   End
   Begin SSFrame fmeStrandSystem
      Caption          =   "Strand System"
      Font3D           =   1  'Raised w/light shading
      ForeColor        =   &H00400000&
      Height           =   675
      Left             =   7020
      TabIndex         =   20
      Top              =   240
      Width            =   1815
      Begin SSOption optBond
         Caption          =   "Unbond"
         Font3D           =   3  'Inset w/light shading
         ForeColor        =   &H00008000&
         Height           =   210
         Index            =   2
         Left             =   60
         TabIndex         =   22
         TabStop          =   0   'False
         Top              =   420
         Width            =   1155
      End
      Begin SSOption optBond
         Caption          =   "Bond"
         Font3D           =   3  'Inset w/light shading
         ForeColor        =   &H00008000&
         Height           =   210
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 31

```
            Index           =   3
            Left            =   60
            TabIndex        =   21
            Top             =   240
            Value           =   -1  'True
            Width           =   1155
         End
      End
      Begin SSPanel Panel3D1
         Alignment       =   1  'Left Justify - MIDDLE
         BevelInner      =   2  'Raised
         BevelOuter      =   0  'None
         Caption         =   "Strand Type"
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00808000&
         Height          =   375
         Left            =   3900
         TabIndex        =   19
         Top             =   1380
         Width           =   1155
      End
      Begin ComboBox cmbStrand
         BackColor       =   &H00FF0000&
         ForeColor       =   &H00FFFFFF&
         Height          =   315
         Left            =   5100
         TabIndex        =   18
         Top             =   1440
         Width           =   3735
      End
   End
End
Begin SSPanel pnlWork
   Align           =   2  'Align Bottom
   Alignment       =   0  'Left Justify - TOP
   ForeColor       =   &H000000FF&
   Height          =   435
   Left            =   0
   Outline         =   -1  'True
   TabIndex        =   9
   Top             =   1740
   Visible         =   0   'False
   Width           =   9600
   Begin TextBox txtDummy
      Height          =   315
      Left            =   3300
      TabIndex        =   11
      Text            =   "txtDummy"
      Top             =   420
      Visible         =   0   'False
      Width           =   210
   End
   Begin CommonDialog cmdlgSystem
      Color           =   &H00C0C0C0&
      DefaultExt      =   ".PSD"
      DialogTitle     =   "Open"
      Filename        =   "Untitled"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 32

```
      Filter          =   "Prestress data files (*.psd)|*.psd All files
      Left            =   7200
      Top             =   0
   End
   Begin Image imgWorkTool1
      Height          =   330
      Left            =   60
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgWorkTool2
      Height          =   330
      Left            =   420
      Picture         =   (Bitmap)
      Top             =   60
      Width           =   360
   End
   Begin Image imgWorkTool3
      Height          =   330
      Left            =   780
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgWorkTool4
      Height          =   330
      Left            =   1140
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgWorkTool5
      Height          =   330
      Left            =   1860
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgWorkTool6
      Height          =   330
      Left            =   1500
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
End
Begin SSPanel pnlStatus
   Align           =   2   'Align Bottom
   Alignment       =   1   'Left Justify - MIDDLE
   Caption         =   "  Ready...."
   Font3D          =   3   'Inset w/light shading
   FontBold        =   -1  'True
   FontItalic      =   0   'False
   FontName        =   "MS Sans Serif"
   FontSize        =   9.75
   FontStrikethru  =   0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 33

```
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   480
      Left            =   0
      RoundedCorners  =   0   'False
      TabIndex        =   1
      Top             =   2175
      Width           =   9600
      Begin SSCheck chkTextInputData
         Caption          =   "DataTool"
         ForeColor        =   &H00008000&
         Height           =   210
         Left             =   8400
         TabIndex         =   8
         Top              =   60
         Width            =   1155
      End
      Begin SSCheck chkInput
         Caption          =   "WorkTool"
         Enabled          =   0   'False
         ForeColor        =   &H00008000&
         Height           =   270
         Left             =   6960
         TabIndex         =   7
         Top              =   240
         Visible          =   0   'False
         Width            =   1215
      End
      Begin SSCheck chkSystem
         Caption          =   "SystemTool"
         ForeColor        =   &H00008000&
         Height           =   240
         Left             =   6960
         TabIndex         =   6
         Top              =   60
         Width            =   1335
      End
      Begin MhState kstInsert
         Autosize         =   -1  'True
         Height           =   420
         Left             =   8760
         Style            =   2  'Insert State
         TabIndex         =   5
         TimerInterval    =   100
         Top              =   30
         Value            =   0   'False
         Visible          =   0   'False
         Width            =   420
      End
      Begin MhState kstNumLock
         Autosize         =   -1  'True
         Height           =   420
         Left             =   8340
         Style            =   1  'Num Lock
         TabIndex         =   4
         TimerInterval    =   100
         Top              =   30
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 34

```
      Value          =   0   'False
      Visible        =   0   'False
      Width          =   420
   End
   Begin MhState kstCapLock
      Autosize       =   -1  'True
      Height         =   420
      Left           =   7920
      TabIndex       =   3
      TimerInterval  =   100
      Top            =   30
      Value          =   0   'False
      Visible        =   0   'False
      Width          =   420
   End
   Begin MhState kstScrollLock
      Autosize       =   -1  'True
      Height         =   420
      Left           =   9180
      Style          =   3   'Scroll Lock
      TabIndex       =   2
      TimerInterval  =   100
      Top            =   60
      Value          =   0   'False
      Visible        =   0   'False
      Width          =   420
   End
End
Begin SSPanel pnlMenu
   Align          =   1   'Align Top
   Alignment      =   4   'Right Justify - MIDDLE
   ForeColor      =   &H00FF0000&
   Height         =   435
   Left           =   0
   Outline        =   -1  'True
   RoundedCorners =   0   'False
   TabIndex       =   0
   Top            =   0
   Width          =   9600
   Begin SSCommand cmdPrevious
      BevelWidth     =   1
      Caption        =   "<< Previous"
      Font3D         =   2   'Raised w/heavy shading
      ForeColor      =   &H00FF0000&
      Height         =   315
      Left           =   3480
      Picture        =   (none)
      TabIndex       =   145
      Top            =   60
      Visible        =   0   'False
      Width          =   1095
   End
   Begin SSCommand cmdNext
      BevelWidth     =   1
      Caption        =   "Next >>"
      Font3D         =   2   'Raised w/heavy shading
      ForeColor      =   &H00FF0000&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 35

```
      Height          =   315
      Left            =   4620
      Picture         =   (none)
      TabIndex        =   144
      Top             =   60
      Visible         =   0   'False
      Width           =   1095
   End
   Begin SSPanel pnlT
      BevelOuter      =   0  'None
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   315
      Left            =   5880
      TabIndex        =   10
      Top             =   60
      Width           =   3615
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   6
      Left            =   2340
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   5
      Left            =   1260
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   7
      Left            =   2700
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   4
      Left            =   1980
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 36

```
      Height          =   330
      Index           =   1
      Left            =   420
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   3
      Left            =   1620
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   2
      Left            =   780
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
   Begin Image imgTool1
      Height          =   330
      Index           =   0
      Left            =   60
      Picture         =   (Bitmap)
      Top             =   45
      Width           =   360
   End
End
Begin Menu mnuFile
   Caption         =   "&File"
   Begin Menu mnuFile_New
      Caption         =   "&New"
      Shortcut        =   {F1}
   End
   Begin Menu mnuFile_Open
      Caption         =   "&Open..."
      Shortcut        =   {F3}
   End
   Begin Menu mnuFile_Save
      Caption         =   "&Save"
      Enabled         =   0   'False
      Shortcut        =   {F2}
   End
   Begin Menu mnuFile_SaveAs
      Caption         =   "Save &As..."
   End
   Begin Menu mnuFile__1
      Caption         =   "-"
   End
   Begin Menu mnuFile_Print
      Caption         =   "&Print..."
      Begin Menu mnuFile_Print_Prestress
         Caption         =   "Prestress"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 37

```
        Begin Menu mnuFile_Print_Prestress_Profile
           Caption          =   "&Profile Tendon"
           Enabled          =   0   'False
           Shortcut         =   ^P
        End
        Begin Menu mnuFile_Print_
           Caption          =   "&Layout Tendon"
           Enabled          =   0   'False
           Shortcut         =   ^L
           Visible          =   0   'False
        End
        Begin Menu mnuFile_Print_Prestress_LProfile
           Caption          =   "List Pro&file"
           Enabled          =   0   'False
           Shortcut         =   ^F
        End
        Begin Menu mnuFile_Print_Prestress_LCompute
           Caption          =   "List &Computation "
           Enabled          =   0   'False
           Shortcut         =   ^C
        End
     End
     Begin Menu mnuFile_Print_Flat
        Caption          =   "&Flat Slab && Flat Plate"
        Begin Menu mnuFile_Print_Flat_LCompute
           Caption          =   "List &Computation"
           Enabled          =   0   'False
           Shortcut         =   ^R
        End
        Begin Menu mnuFile_Print_Flat_Reinforce
           Caption          =   "List &Reinforcement"
           Enabled          =   0   'False
           Visible          =   0   'False
        End
     End
     Begin Menu mnuFile_Print__
        Caption          =   "-"
     End
     Begin Menu mnuFile_Print_Geometric
        Caption          =   "&Geometric"
        Enabled          =   0   'False
        Shortcut         =   ^G
     End
     Begin Menu mnuFile_Print_Shear
        Caption          =   "&Shear Diagram"
        Enabled          =   0   'False
        Shortcut         =   ^S
     End
     Begin Menu mnuFile_Print_Moment
        Caption          =   "&Moment Diagram"
        Enabled          =   0   'False
        Shortcut         =   ^M
     End
  End
  Begin Menu mnuFile__2
     Caption          =   "-"
  End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 38

```
      Begin Menu mnuFile_Exit
         Caption         =   "E&xit"
      End
   End
   Begin Menu mnuInput
      Caption         =   "&Input"
      Begin Menu mnuInput_Node
         Caption         =   "&Node"
         Enabled         =   0   'False
         Shortcut        =   {F4}
      End
      Begin Menu mnuInput_Element
         Caption         =   "&Design Strip Element"
         Enabled         =   0   'False
         Shortcut        =   {F5}
      End
      Begin Menu mnuInput_Load
         Caption         =   "&Load "
         Enabled         =   0   'False
         Shortcut        =   {F6}
      End
      Begin Menu mnuInput_Material
         Caption         =   "&Material Properties"
         Enabled         =   0   'False
         Shortcut        =   {F7}
      End
      Begin Menu mnuInput_Information
         Caption         =   "&Information"
         Enabled         =   0   'False
         Shortcut        =   {F8}
      End
   End
   Begin Menu mnuDesign
      Caption         =   "&Design"
      Begin Menu mnuDesign_Prestress
         Caption         =   "&Prestress"
         Enabled         =   0   'False
         Shortcut        =   {F11}
      End
      Begin Menu mnuDesign_Flat
         Caption         =   "&Flat Slab && Flat Plate"
         Enabled         =   0   'False
         Shortcut        =   {F12}
      End
   End
   Begin Menu mnuResult
      Caption         =   "&Result"
      Begin Menu mnuResult_Prestress
         Caption         =   "&Prestress"
         Begin Menu mnuResult_Prestress_Profile
            Caption         =   "&Profile Tendon"
            Enabled         =   0   'False
            Shortcut        =   ^K
         End
         Begin Menu mnuResult_Prestress_Layout
            Caption         =   "&Layout Tendon"
            Enabled         =   0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 39

```
            Shortcut        =   ^O
            Visible         =   0   'False
         End
         Begin Menu mnuResult_Prestress_LProfile
            Caption         =   "List Pro&file"
            Enabled         =   0   'False
            Shortcut        =   ^U
         End
         Begin Menu mnuResult_Prestress_LCompute
            Caption         =   "List &Computation"
            Enabled         =   0   'False
            Shortcut        =   ^V
         End
      End
      Begin Menu mnuResult_Flat
         Caption         =   "&Flat Slab && Flat Plate"
         Begin Menu mnuResult_Flat_LCompute
            Caption         =   "List &Computation"
            Enabled         =   0   'False
            Shortcut        =   ^Q
         End
         Begin Menu mnuResult_Flat_LReinforce
            Caption         =   "List &Reinforcement"
            Enabled         =   0   'False
            Visible         =   0   'False
         End
      End
      Begin Menu mnuResult__
         Caption         =   "-"
      End
      Begin Menu mnuResult_Geometric
         Caption         =   "&Geometric"
         Enabled         =   0   'False
         Shortcut        =   ^X
      End
      Begin Menu mnuResult_Shear
         Caption         =   "&Shear Diagram"
         Enabled         =   0   'False
         Shortcut        =   ^Y
      End
      Begin Menu mnuResult_Moment
         Caption         =   "&Moment Diagram"
         Enabled         =   0   'False
         Shortcut        =   ^Z
      End
   End
   Begin Menu mnuOption
      Caption         =   "&Option"
      Begin Menu mnuOption_SaveSetting
         Caption         =   "&Save Setting "
         Checked         =   -1  'True
      End
   End
   Begin Menu mnuWindow
      Caption         =   "&Window"
      Begin Menu mnuWindow_ArrangeIcon
         Caption         =   "&Arrange Icon"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 40

```
         Shortcut        =   ^A
      End
      Begin Menu mnuWindow_Close
         Caption         =   "&Close All Windows"
         Enabled         =   0   'False
         Shortcut        =   ^W
      End
   End
   Begin Menu mnuHelp
      Caption         =   "&Help"
      Begin Menu mnuHelp_System
         Caption         =   "&System"
      End
      Begin Menu mnuHelp_About
         Caption         =   "&About"
      End
   End
End
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 1

```
Sub Init_Screen_Resize ()
    Dim LoadWidth As Integer
    Dim NodeWidth As Integer
    Dim kstWidth As Integer
    Const kstTop = 30
    Const kstRight = 30
    LoadWidth = (grdLoad.Width - 300) \ grdLoad.Cols
    NodeWidth = (grdNode.Width - 300) \ grdNode.Cols
    kstWidth = kstCapLock.Width
    For I = 0 To 4
     fmeTextInputData(I).Width = frmMain.Width - 860
    Next I
    frmMain.kstCapLock.Top = kstTop
    frmMain.kstNumLock.Top = kstTop
    frmMain.kstInsert.Top = kstTop
    frmMain.kstScrollLock.Top = kstTop
    frmMain.kstScrollLock.Left = frmMain.ScaleWidth - kstRight - kstWidth
    frmMain.kstInsert.Left = frmMain.ScaleWidth - kstRight - (2 * kstWidth
)
    frmMain.kstNumLock.Left = frmMain.ScaleWidth - kstRight - (3 * kstWidt
h)
    frmMain.kstCapLock.Left = frmMain.ScaleWidth - kstRight - (4 * kstWidt
h)
    'grdNode
    For I = 0 To grdNode.Cols - 1
     grdNode.ColWidth(I) = NodeWidth
    Next I
    'grdLoad
    For I = 0 To grdLoad.Cols - 1
     grdLoad.ColWidth(I) = LoadWidth
    Next I
End Sub

Sub Init_ValueMain ()
    'Toolbar1
    Icon1(0, 0) = "new-up.bmp"
    Icon1(0, 1) = "new-mds.bmp"
    Icon1(1, 0) = "open-up.bmp"
    Icon1(1, 1) = "open-mds.bmp"
    Icon1(2, 0) = "save-up.bmp"
    Icon1(2, 1) = "save-mds.bmp"
    Icon1(3, 0) = "light-up.bmp"
    Icon1(3, 1) = "light-dn.bmp"
    Icon1(4, 0) = "lightfup.bmp"
    Icon1(4, 1) = "lightfdn.bmp"
    Icon1(5, 0) = "grd01-up.bmp"
    Icon1(5, 1) = "grd01-dw.bmp"
    Icon1(6, 0) = "Cals&mu.bmp"
    Icon1(6, 1) = "Cals&md.bmp"
    Icon1(7, 0) = "CalM&mu.bmp"
    Icon1(7, 1) = "CalM&md.bmp"

    'Toolbar2
    Icon2(0, 0) = "new-up.bmp"
    Icon2(0, 1) = "new-mds.bmp"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 2

```
    Icon2(1, 0) = "grd01-up.bmp"
    Icon2(1, 1) = "grd01-dw.bmp"
    Icon2(2, 0) = "save-up.bmp"
    Icon2(2, 1) = "save-mds.bmp"
    Icon2(3, 0) = "save-up.bmp"
    Icon2(3, 1) = "save-mds.bmp"
    Icon2(4, 0) = "save-up.bmp"
    Icon2(4, 1) = "save-mds.bmp"
    Icon2(6, 0) = "Cals&mu.bmp"
    Icon2(6, 1) = "Cals&md.bmp"
    'ToolSelect
    'init menu
    Initial_Menu 1
End Sub

Sub mnuFile_Print_Prestress_Layout_Click ()
    Print_Layout
End Sub

Sub chkCHead_Click (Index As Integer, Value As Integer)
    If chkCHead(0).Value Then
     GeneralData(CProj).SlabIndex = 1
    End If
    If chkCHead(1).Value Then
     GeneralData(CProj).SlabIndex = 2
    End If
    If chkCHead(0).Value And chkCHead(1).Value Then
     GeneralData(CProj).SlabIndex = 3
    End If
    If Not (chkCHead(0).Value) And Not (chkCHead(1).Value) Then
     GeneralData(CProj).SlabIndex = 0
    End If
End Sub

Sub chkInput_Click (Value As Integer)
    'pnlWork.Top = 435
    'pnlWork.Visible = Value
    frmMain.Arrange 3
End Sub

Sub chkSystem_Click (Value As Integer)
    pnlMenu.Top = 0
    pnlMenu.Visible = Value
    frmMain.Arrange 3
End Sub

Sub chkTextInputData_Click (Value As Integer)
    pnlTextInputData.Visible = Value
    frmMain.Arrange 3
End Sub

Sub cmbColumnEc_Change ()
    GeneralData(CProj).CEc = CSng(cmbColumnEc.Text)
End Sub

Sub cmbColumnFc__Change ()
    If cmbSlabFc_.Text <> "" Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 3

```
        GeneralData(CProj).CFc_ = CSng(cmbColumnFc_.Text)
        cmbColumnEc.Text = Fc_ToEc(CSng(cmbColumnFc_.Text))
      End If
  End Sub

  Sub cmbColumnFc__Click ()
      If cmbColumnFc_.Text <> "" Then
       GeneralData(CProj).CFc_ = CSng(cmbColumnFc_.Text)
       cmbColumnEc.Text = Fc_ToEc(CSng(cmbColumnFc_.Text))
      End If
  End Sub

  Sub cmbFs__Change ()
      If cmbFs_.Text <> "" Then
       GeneralData(CProj).RFs_ = CSng(cmbFs_.Text)
      End If
  End Sub

  Sub cmbFs__Click ()
      If cmbFs_.Text <> "" Then
       GeneralData(CProj).RFs_ = CSng(cmbFs_.Text)
      End If
  End Sub

  Sub cmbSFs__Change ()
      If cmbFs_.Text <> "" Then
       GeneralData(CProj).SFs_ = CSng(cmbSFs_.Text)
      End If
  End Sub

  Sub cmbSFs__Click ()
      If cmbFs_.Text <> "" Then
       GeneralData(CProj).SFs_ = CSng(cmbSFs_.Text)
      End If
  End Sub

  Sub cmbSlabEc_Change ()
      GeneralData(CProj).SEc = CSng(cmbSlabEc.Text)
  End Sub

  Sub cmbSlabEc_Click ()
      GeneralData(CProj).SEc = CSng(cmbSlabEc.Text)
  End Sub

  Sub cmbSlabFc__Change ()
      If cmbSlabFc_.Text <> "" Then
       GeneralData(CProj).SFc_ = CSng(cmbSlabFc_.Text)
       cmbSlabEc.Text = Fc_ToEc(CSng(cmbSlabFc_.Text))
      End If
  End Sub

  Sub cmbSlabFc__Click ()
      If cmbSlabFc_.Text <> "" Then
       GeneralData(CProj).SFc_ = CSng(cmbSlabFc_.Text)
       cmbSlabEc.Text = Fc_ToEc(CSng(cmbSlabFc_.Text))
      End If
  End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 4

```
Sub cmbSlabFc__DragOver (Source As Control, X As Single, Y As Single, Stat
e As Integer)
    'pnlStatus.Caption = "  concrete ultimate strength of slab ~210-400 ks
c"
End Sub

Sub cmbStrand_Click ()
    StrandIndex(CProj) = cmbStrand.ListIndex
    pnlStatus.Caption = "  Select Tendon Type"
End Sub

Sub cmdNext_Click ()
    If CurrentPage < TextModule Then
     frmList(CProj).txtListCompute(CurrentPage).Visible = False
     CurrentPage = CurrentPage + 1
     frmList(CProj).txtListCompute(CurrentPage).Visible = True
     If CurrentPage = TextModule Then
      frmMain.cmdNext.Enabled = False
     Else
      frmMain.cmdNext.Enabled = True
     End If
     If CurrentPage = 1 Then
      frmMain.cmdPrevious.Enabled = False
     Else
      frmMain.cmdPrevious.Enabled = True
     End If
    End If
End Sub

Sub cmdNodeOK_Click ()
    Input_NodeData
End Sub

Sub cmdNodeOK_MouseMove (Button As Integer, Shift As Integer, X As Single,
 Y As Single)
    pnlStatus.Caption = "  Click when data ready"
End Sub

Sub cmdPrevious_Click ()
    If CurrentPage > 1 Then
     frmList(CProj).txtListCompute(CurrentPage).Visible = False
     CurrentPage = CurrentPage - 1
     frmList(CProj).txtListCompute(CurrentPage).Visible = True
     If CurrentPage = TextModule Then
      frmMain.cmdNext.Enabled = False
     Else
      frmMain.cmdNext.Enabled = True
     End If
     If CurrentPage = 1 Then
      frmMain.cmdPrevious.Enabled = False
     Else
      frmMain.cmdPrevious.Enabled = True
     End If
    End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 5

```
Sub fmeDropOrCapital_MouseMove (Button As Integer, Shift As Integer, X As
Single, Y As Single)
    pnlStatus.Caption = "  Select type of column,Drop panel or capital or
both"
End Sub

Sub fmeEdgeYIndex_MouseMove (Button As Integer, Shift As Integer, X As Sin
gle, Y As Single)
    pnlStatus.Caption = "  Edge note,in Y axis edge in top or bottom or no
 edge(Interior)"
End Sub

Sub fmeNode_MouseMove (Button As Integer, Shift As Integer, X As Single, Y
 As Single)
    pnlStatus.Caption = "  Input Column Data"
End Sub

Sub fmeReinforce_MouseMove (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    pnlStatus.Caption = "  Yeild strength of reinforce steel"
End Sub

Sub fmeSlab_MouseMove (Button As Integer, Shift As Integer, X As Single, Y
 As Single)
    pnlStatus.Caption = "  Properties of concrete in slab and column"
End Sub

Sub fmeSlabSystem_MouseMove (Button As Integer, Shift As Integer, X As Sin
gle, Y As Single)
    pnlStatus.Caption = "  Slab system,Select flat plate or flat slab"
End Sub

Sub fmeStrand_MouseMove (Button As Integer, Shift As Integer, X As Single,
 Y As Single)
    pnlStatus.Caption = "  Breaking strength of 1 unit strand"
End Sub

Sub fmeStrandSystem_DragDrop (Source As Control, X As Single, Y As Single)
    pnlStatus.Caption = "  Method to Construct ,Bond or Unbond"
End Sub

Sub fmeTextInputData_MouseMove (Index As Integer, Button As Integer, Shift
 As Integer, X As Single, Y As Single)
    pnlStatus.Caption = "  Ready...."
End Sub

Sub Frame3D1_MouseMove (Button As Integer, Shift As Integer, X As Single,
Y As Single)
    pnlStatus.Caption = "  Edge note,in X axis edge in left or right or no
 edge(Interior)"
End Sub

Sub gphSelectInput_Click (Index As Integer, Value As Integer)
    fmeTextInputData(InputTextIndex).Visible = False
    InputTextIndex = Index
    fmeTextInputData(InputTextIndex).Visible = gphSelectInput(InputTextInd
ex).Value
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 6

```
    '''' Set to False
    mnuInput_Node.Checked = False
    mnuInput_Load.Checked = False
    mnuInput_Material.Checked = False
    mnuInput_Information.Checked = False
    mnuInput_Element.Checked = False

    Select Case Index
    Case 0: mnuInput_Node.Checked = gphSelectInput(InputTextIndex).Value
    Case 1: mnuInput_Load.Checked = gphSelectInput(InputTextIndex).Value
    Case 2: mnuInput_Element.Checked = gphSelectInput(InputTextIndex).Valu
e
    Case 3: mnuInput_Material.Checked = gphSelectInput(InputTextIndex).Val
ue
    Case 4: mnuInput_Information.Checked = gphSelectInput(InputTextIndex).
Value
    End Select
    'If InputTextIndex = 0 Then
    ' grdNode.SetFocus
    'End If
End Sub

Sub grdElement_MouseMove (Button As Integer, Shift As Integer, X As Single
, Y As Single)
    pnlStatus.Caption = "  Element data for design,Separate in line and li
ne element"
End Sub

Sub grdLoad_Click ()
   'Dim a, b, c As Integer
   'If ((frmMain.grdLoad.Row > 1) And (frmMain.grdLoad.Row < SLoadNum(CPro
j))) Then
    ' For I = 1 To 2
    '  frmMain.grdLoad.Row = frmMain.grdLoad.Row: frmMain.grdLoad.Col = I
    '  frmMain.txtLoad(I) = frmMain.grdLoad.Text
    ' Next I
    '  frmMain.grdLoad.Col = 3
    '  a = CInt(frmMain.grdLoad.Text)
    '  frmMain.grdLoad.Col = 4
    '' b = CInt(frmMain.grdLoad.Text)
     ' frmMain.grdLoad.Col = 5
    '  c = CInt(frmMain.grdLoad.Text)
    ' frmMain.optLoadSupport(a).Value = True
    ' frmMain.optNodeSupport(b).Value = True
    ' frmMain.optNodeSupport(c).Value = True
    'End If
End Sub

Sub grdNode_Click ()
    Dim a, b, c As Integer
    If ((frmMain.grdNode.Row >= 1) And (frmMain.grdNode.Row <= SNodeNum(CP
roj))) Then
       frmMain.grdNode.Row = frmMain.grdNode.Row
       frmMain.grdNode.Col = 0
       frmMain.txtNode(1) = frmMain.grdNode.Text
       frmMain.grdNode.Col = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 7

```
      frmMain.txtNode(2) = frmMain.grdNode.Text
      frmMain.grdNode.Col = 2
      frmMain.txtNode(3) = frmMain.grdNode.Text
      frmMain.grdNode.Col = 3
      b = CInt(frmMain.grdNode.Text)
      frmMain.grdNode.Col = 4
      c = CInt(frmMain.grdNode.Text)
      frmMain.grdNode.Col = 5
      a = CInt(frmMain.grdNode.Text)
     frmMain.optNodeSupport(a).Value = True
     frmMain.optEdgeX(b).Value = True
     frmMain.optEdgeY(c).Value = True
     frmMain.txtNode(4).Text = "1"
     frmMain.txtNode(5).Text = "1"
    End If
End Sub

Sub grdNode_DblClick ()
    'If (grdNode.Row > 0) And (grdNode.Col > 0) Then
    ' grdNode_KeyPress 13
    'End If
End Sub

Sub grdNode_KeyPress (KeyAscii As Integer)
    'Dim Char As String
    'Select Case KeyAscii
    'Case 27: 'Escape
    'Case 9: 'Tab
    ' If Shifting Then
    '  If grdNode.Col > 1 Then
    '   grdNode.Col = grdNode.Col - 1
    '  End If
    ' Else
    '  If grdNode.Col < (grdNode.Cols - 1) Then
    '   grdNode.Col = grdNode.Col + 1
    '  End If
    ' End If
    ' UnSelect
    'Case Else:
    ' If KeyAscii = 13 Then
    '  txtGrid.Text = grdNode.Text
    '  txtGrid.SelStart = Len(txtGrid.Text)
    ' Else
    '  Char = Chr$(KeyAscii)
    '  txtGrid.Text = Char
    '  txtGrid.SelStart = 1
    ' End If
    ' ShowTextBox
    ' KeyAscii = 0
    'End Select
    If KeyAscii = 8 Then
     If ((frmMain.grdNode.Row >= 1) And (frmMain.grdNode.Row <= SNodeNum(C
Proj))) Then
      SNodeNum(CProj) = SNodeNum(CProj) - 1
      For I = frmMain.grdNode.Row To SNodeNum(CProj)
       Column(CProj, I) = Column(CProj, I + 1)
      Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 8

```
      End If
     End If
     Show_Data
End Sub

Sub grdNode_MouseDown (Button As Integer, Shift As Integer, X As Single, Y
 As Single)
     txtGrid.Visible = False
End Sub

Sub grdNode_MouseMove (Button As Integer, Shift As Integer, X As Single, Y
 As Single)
     pnlStatus.Caption = "  Display column data such as coordinate,length,c
olumn size,etc."
End Sub

Sub imgTool1_MouseDown (Index As Integer, Button As Integer, Shift As Inte
ger, X As Single, Y As Single)
     imgTool1(Index).Picture = LoadPicture(DirApp + "picico\" + Icon1(Index
, 1))
     IconClick1 = True
End Sub

Sub imgTool1_MouseMove (Index As Integer, Button As Integer, Shift As Inte
ger, X As Single, Y As Single)
     If Not (X >= 0 And X <= 360 And Y >= 0 And Y <= 330) Then
      IconClick1 = False
      imgTool1(Index).Picture = LoadPicture(DirApp + "picico\" + Icon1(Inde
x, 1))
     End If
End Sub

Sub imgTool1_MouseUp (Index As Integer, Button As Integer, Shift As Intege
r, X As Single, Y As Single)
     If X >= 0 And X <= 360 And Y >= 0 And Y <= 330 And IconClick1 Then
      Select Case Index
       Case 0:
        New_File
       Case 1:
        Open_File
       Case 2:
        Save_File
       Case 3:
        pnlStatus.Caption = "  Design processing...."
        Screen.MousePointer = 11
        If SNodeNum(CProj) <> 0 Then
         Design_PSlab
        ElseIf SNodeNum(CProj) = 0 Then
         MsgBox "Data Error. Please input column data.", 0 + 48, "Data Erro
r."
        End If
        Screen.MousePointer = 0
        pnlStatus.Caption = "  Ready...."
       Case 4:
        pnlStatus.Caption = "  Design processing...."
        Screen.MousePointer = 11
        If SNodeNum(CProj) <> 0 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 9

```
        Design_FSlab
       ElseIf SNodeNum(CProj) = 0 Then
        MsgBox "Data Error. Please input column data.", 0 + 48, "Data Erro
r."
       End If
       Screen.MousePointer = 0
       pnlStatus.Caption = "  Ready...."
      Case 5:
       Screen.MousePointer = 11
       frmMain.pnlStatus.Caption = "  Generating slab form...."
       If SNodeNum(CProj) <> 0 Then
        Screen_Geometric
       ElseIf SNodeNum(CProj) = 0 Then
        MsgBox "Data Error. Please input column data.", 0 + 48, "Data Erro
r."
       End If
       frmMain.pnlStatus.Caption = "  Ready...."
       Screen.MousePointer = 0
     Case 6:
       Screen.MousePointer = 11
       frmMain.pnlStatus.Caption = "  Generating slab form...."
       If SNodeNum(CProj) <> 0 Then
        Screen_Shear
       ElseIf SNodeNum(CProj) = 0 Then
        MsgBox "Data Error. Please input column data.", 0 + 48, "Data Erro
r."
       End If
       frmMain.pnlStatus.Caption = "  Ready...."
       Screen.MousePointer = 0
     Case 7:
       Screen.MousePointer = 11
       frmMain.pnlStatus.Caption = "  Generating slab form...."
       If SNodeNum(CProj) <> 0 Then
        Screen_Moment
       ElseIf SNodeNum(CProj) = 0 Then
        MsgBox "Data Error. Please input column data.", 0 + 48, "Data Erro
r."
       End If
       frmMain.pnlStatus.Caption = "  Ready...."
       Screen.MousePointer = 0
     End Select
    End If
    imgTool1(Index).Picture = LoadPicture(DirApp + "picico\" + Icon1(Index
, 0))
End Sub

Sub imgWorkTool1_DragDrop (Source As Control, X As Single, Y As Single)
     imgWorkTool1.Picture = LoadPicture(DirApp + "picico\" + Icon2(Index,
0))
     Unload frmWorkTool1
End Sub

Sub imgWorkTool1_MouseDown (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    imgWorkTool1.Picture = LoadPicture(DirApp + "picico\" + Icon2(0, 1))
    imgWorkTool1.Drag 1
    Dim ToolNum As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 10

```
    ToolNum = 0
    If frmMain.chkSystem Then ToolNum = ToolNum + 1
    If frmMain.chkInput Then ToolNum = ToolNum + 1
    'frmMain.Panel3D1.Visible = True
    frmWorkTool1.Top = frmMain.Top + 620 + frmMain.pnlMenu.Height * ToolNu
m'frmMain.Height - (frmMain.SCaleHeight * Screen.TwipsPerPixelY) + (frmMai
n.pnlMenu.Height * ToolNum)
    frmWorkTool1.Left = frmMain.Left + frmMain.imgWorkTool1.Left + 80' (fr
mMain.Width / 2)
    Load frmWorkTool1
    frmWorkTool1.Show
    frmWorkTool1.SetFocus
End Sub

Sub imgWorkTool2_DragDrop (Source As Control, X As Single, Y As Single)
     imgWorkTool2.Picture = LoadPicture(DirApp + "picico\" + Icon2(1, 0))
     Unload frmWorkTool2
End Sub

Sub imgWorkTool2_MouseDown (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    imgWorkTool2.Picture = LoadPicture(DirApp + "picico\" + Icon2(1, 1))
    imgWorkTool2.Drag 1
    Dim ToolNum As Integer
    ToolNum = 0
    If frmMain.chkSystem Then ToolNum = ToolNum + 1
    If frmMain.chkInput Then ToolNum = ToolNum + 1
    'frmMain.Panel3D1.Visible = True
    frmWorkTool2.Top = frmMain.Top + 620 + frmMain.pnlMenu.Height * ToolNu
m'frmMain.Height - (frmMain.SCaleHeight * Screen.TwipsPerPixelY) + (frmMai
n.pnlMenu.Height * ToolNum)
    frmWorkTool2.Left = frmMain.Left + frmMain.imgWorkTool2.Left + 60' (fr
mMain.Width / 2)
    'Work Tool2 ,Column Dimension
    Load frmWorkTool2
    frmWorkTool2.Show
    frmWorkTool2.SetFocus
End Sub

Sub imgWorkTool3_MouseDown (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    imgTool1(Index).Picture = LoadPicture(DirApp + "picico\" + Icon1(Index
,  1))
    IconClick1 = True
End Sub

Sub imgWorkTool3_MouseMove (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    If Not (X >= 0 And X <= 360 And Y >= 0 And Y <= 330) Then
     IconClick1 = False
     imgTool1(Index).Picture = LoadPicture(DirApp + "picico\" + Icon1(Inde
x, 1))
    End If
End Sub

Sub imgWorkTool5_MouseDown (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 11

```
    imgWorkTool5.Picture = LoadPicture(DirApp + "picico\" + Icon1(4, 1))
    IconClick1 = True
End Sub

Sub imgWorkTool5_MouseMove (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    If Not (X >= 0 And X <= 360 And Y >= 0 And Y <= 330) Then
     IconClick1 = False
     imgWorkTool5.Picture = LoadPicture(DirApp + "picico\" + Icon1(4, 1))
    End If
End Sub

Sub imgWorkTool5_MouseUp (Button As Integer, Shift As Integer, X As Single
, Y As Single)
    If X >= 0 And X <= 360 And Y >= 0 And Y <= 330 And IconClick1 Then
     Screen.MousePointer = 11
     pnlStatus.Caption = "Wait...."
     If FileIndex <> -1 Then
      BCal
     Else
      Screen.MousePointer = 0
      Beep
      MsgBox "Load data first.", 0 + 48, "Data error."
     End If
     pnlStatus.Caption = "Ready...."
     Screen.MousePointer = 0
    End If
    imgWorkTool5.Picture = LoadPicture(DirApp + "picico\" + Icon1(4, 0))
End Sub

Sub imgWorkTool6_MouseDown (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    imgWorkTool6.Picture = LoadPicture(DirApp + "picico\" + Icon2(6, 1))
    IconClick1 = True
End Sub

Sub imgWorkTool6_MouseMove (Button As Integer, Shift As Integer, X As Sing
le, Y As Single)
    If Not (X >= 0 And X <= 360 And Y >= 0 And Y <= 330) Then
     IconClick1 = False
     imgWorkTool6.Picture = LoadPicture(DirApp + "picico\" + Icon2(6, 1))
    End If
End Sub

Sub imgWorkTool6_MouseUp (Button As Integer, Shift As Integer, X As Single
, Y As Single)
    If X >= 0 And X <= 360 And Y >= 0 And Y <= 330 And IconClick1 Then
     Screen.MousePointer = 11
     pnlStatus.Caption = "Wait...."
     If FileIndex <> -1 Then
      Load frmInputDesign
      frmInputDesign.Top = (Screen.Height - frmInputDesign.Height) / 2
      frmInputDesign.Left = (Screen.Width - frmInputDesign.Width) / 2
      frmInputDesign.Show MODEL
     Else
      Screen.MousePointer = 0
      Beep
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 12

```
      MsgBox "Load data first.", 0 + 48, "Data error."
     End If
     pnlStatus.Caption = "Ready...."
     Screen.MousePointer = 0
    End If
    imgWorkTool6.Picture = LoadPicture(DirApp + "picico\" + Icon2(6, 0))
End Sub

Sub MDIForm_DragDrop (Source As Control, X As Single, Y As Single)
    'Unload frmWorkTool2 'try
End Sub

Sub MDIForm_Load ()
    Init_DataValue
    Init_ValueMain
End Sub

Sub MDIForm_Resize ()
    If WindowState <> 1 Then
     Init_Screen_Resize
    End If
End Sub

Sub mnuDesign_Flat_Click ()
    mnuDesign_Prestress.Checked = False
    mnuDesign_Flat.Checked = True
    pnlStatus.Caption = "  Design processing...."
    Screen.MousePointer = 11
    If SNodeNum(CProj) <> 0 Then
     Design_FSlab
     Initial_Menu 4
    ElseIf SNodeNum(CProj) = 0 Then
     MsgBox "Data Error. Please input column data.", 0 + 48, "Data Error."
    End If
    Screen.MousePointer = 0
    pnlStatus.Caption = "  Ready...."
End Sub

Sub mnuDesign_Prestress_Click ()
    mnuDesign_Flat.Checked = False
    mnuDesign_Prestress.Checked = True
    pnlStatus.Caption = "  Design processing...."
    Screen.MousePointer = 11
    If SNodeNum(CProj) <> 0 Then
     Design_PSlab
     Initial_Menu 3
    ElseIf SNodeNum(CProj) = 0 Then
     MsgBox "Data Error. Please input column data.", 0 + 48, "Data Error."
    End If
    Screen.MousePointer = 0
    pnlStatus.Caption = "  Ready...."
End Sub

Sub mnuFile_Exit_Click ()
    Ready_Exit
    End
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 13

```
Sub mnuFile_New_Click ()
    New_File
End Sub

Sub mnuFile_Open_Click ()
    Open_File
End Sub

Sub mnuFile_Print_Flat_LCompute_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Listing Computation For Flate slab & Fl
at Plate system...."
    Print_FlatListCompute
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuFile_Print_Geometric_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Printing Slab...."
    Print_Geometric
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuFile_Print_Moment_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Printing Moment Diagram...."
    Print_Moment
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuFile_Print_Prestress_LCompute_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Listing Computation for Prestress-Tensi
oned slab...."
    Print_ListCompute
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0

End Sub

Sub mnuFile_Print_Prestress_LProfile_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Listing Profile of Tendon for Prestress
-Tensioned slab...."
    Print_ListProfile
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0

End Sub

Sub mnuFile_Print_Prestress_Profile_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Printing Profile of Tendon...."
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 14

```
    Print_Profile
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0

End Sub

Sub mnuFile_Print_Shear_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Printing Shear Diagram...."
    Print_Shear
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuFile_Save_Click ()
    Save_File
End Sub

Sub mnuFile_SaveAs_Click ()
    SaveAs_File
End Sub

Sub mnuHelp_About_Click ()
    frmMain.pnlStatus.Caption = "  About Program...."
    frmAbout.Top = (Screen.Height - frmAbout.Height) \ 2
    frmAbout.Left = (Screen.Width - frmAbout.Width) \ 2
    Load frmAbout
    frmAbout.Show MODAL
    frmMain.pnlStatus.Caption = "  Ready...."
    'Print_Try
End Sub

Sub mnuHelp_System_Click ()
    frmSysInfo.Top = (Screen.Height - frmSysInfo.Height) / 2
    frmSysInfo.Left = (Screen.Width - frmSysInfo.Width) / 2
    frmSysInfo.Show MODAL
    'Screen_Try
End Sub

Sub mnuInput_Element_Click ()
    frmMain.mnuInput_Node.Checked = False
    frmMain.mnuInput_Element.Checked = False
    frmMain.mnuInput_Information.Checked = False
    frmMain.mnuInput_Material.Checked = False
    frmMain.mnuInput_Load.Checked = False
    frmMain.gphSelectInput(2).Value = True
End Sub

Sub mnuInput_Information_Click ()
    frmMain.mnuInput_Node.Checked = False
    frmMain.mnuInput_Element.Checked = False
    frmMain.mnuInput_Information.Checked = False
    frmMain.mnuInput_Material.Checked = False
    frmMain.mnuInput_Load.Checked = False
    frmMain.gphSelectInput(4).Value = True
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 15

```
Sub mnuInput_Load_Click ()
    frmMain.mnuInput_Node.Checked = False
    frmMain.mnuInput_Element.Checked = False
    frmMain.mnuInput_Information.Checked = False
    frmMain.mnuInput_Material.Checked = False
    frmMain.mnuInput_Load.Checked = False
    frmMain.gphSelectInput(1).Value = True
End Sub

Sub mnuInput_Material_Click ()
    frmMain.mnuInput_Node.Checked = False
    frmMain.mnuInput_Element.Checked = False
    frmMain.mnuInput_Information.Checked = False
    frmMain.mnuInput_Material.Checked = False
    frmMain.mnuInput_Load.Checked = False
    frmMain.gphSelectInput(3).Value = True
End Sub

Sub mnuInput_Node_Click ()
    frmMain.mnuInput_Node.Checked = False
    frmMain.mnuInput_Element.Checked = False
    frmMain.mnuInput_Information.Checked = False
    frmMain.mnuInput_Material.Checked = False
    frmMain.mnuInput_Load.Checked = False
    frmMain.gphSelectInput(0).Value = True
End Sub

Sub mnuOption_SaveSetting_Click ()
    If mnuOption_saveSetting.Checked Then
     mnuOption_saveSetting.Checked = False
    Else
     mnuOption_saveSetting.Checked = True
    End If
End Sub

Sub mnuResult_Flat_LCompute_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Listing computing...."
    Screen_FlatListCompute
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuResult_Geometric_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Generating slab form...."
    Screen_Geometric
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuResult_Moment_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Generating moment diagram...."
    Screen_Moment
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 16

```
End Sub

Sub mnuResult_Prestress_LCompute_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Listing computing...."
    Screen_ListCompute
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuResult_Prestress_LProfile_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Listing profile...."
    Screen_ListProfile
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuResult_Prestress_Profile_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Generating strand profile...."
    Screen_Profile
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuResult_Shear_Click ()
    Screen.MousePointer = 11
    frmMain.pnlStatus.Caption = "  Generating shear diagram...."
    Screen_Shear
    frmMain.pnlStatus.Caption = "  Ready...."
    Screen.MousePointer = 0
End Sub

Sub mnuWindow_ArrangeIcon_Click ()
    frmMain.Arrange 3
End Sub

Sub mnuWindow_Close_Click ()
    Unload frmList(CProj)
    Unload frmData(CProj)
    Unload frmOutGraph(CProj)
    Initial_Menu 1
End Sub

Sub optBond_Click (Index As Integer, Value As Integer)
    If Value Then
     StrandIndex(CProj) = Index
    End If
End Sub

Sub optSystem_Click (Index As Integer, Value As Integer)
    Select Case Index
    Case 0:
     If Value Then
      GeneralData(CProj).SlabIndex = Index
      chkCHead(0).Value = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 17

```
      chkCHead(0).Enabled = False
      chkCHead(1).Value = False
      chkCHead(1).Enabled = False
     End If
    Case 1:
     If Value Then
      If chkCHead(0).Value Then
       GeneralData(CProj).SlabIndex = 1
      End If
      If chkCHead(1).Value Then
       GeneralData(CProj).SlabIndex = 2
      End If
      If chkCHead(0).Value And chkCHead(1).Value Then
       GeneralData(CProj).SlabIndex = 3
      End If
      If Not (chkCHead(0).Value) And Not (chkCHead(1).Value
       GeneralData(CProj).SlabIndex = 0
      End If
      chkCHead(0).Enabled = True
      chkCHead(1).Enabled = True
     End If
    End Select
End Sub

Sub optSystem_MouseMove (Index As Integer, Button As Intege
eger, X As Single, Y As Single)
    If Index = 1 Then pnlStatus.Caption = "  Flat slab,slab
l or capital or both"
    If Index = 0 Then pnlStatus.Caption = "  Flat plate,sla
panel and capital"
End Sub

Sub Panel3D1_MouseMove (Button As Integer, Shift As Integer
Y As Single)
    pnlStatus.Caption = "  Select Tendon Type"
End Sub

Sub pnlStatus_MouseMove (Button As Integer, Shift As Intege
 Y As Single)
    pnlStatus.Caption = "  Status bar,Information data"
End Sub

Sub pnlTextInputData_MouseMove (Button As Integer, Shift As
Single, Y As Single)
    pnlStatus.Caption = "  Ready...."
End Sub

Sub pnlThick_MouseMove (Button As Integer, Shift As Integer
Y As Single)
   pnlStatus.Caption = "  Slab Thickness [Output]"
End Sub

Sub pnlThick_U_MouseMove (Index As Integer, Button As Integ
teger, X As Single, Y As Single)
    pnlStatus.Caption = "  Slab Thickness [Output]"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 18

```
Sub tmrTime_Timer ()

    txtTime.Text = Format$(Now, "hh:mm:ss AM/PM")
    pnlT.Caption = Format$(Now, "dddd mm/dd/yyyy  hh:mm:ss AM/PM  ")
End Sub

Sub txtAddress_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 125, txtAddress
    End If
End Sub

Sub txtAddress_LostFocus ()
'    Input_Data 125, txtAddress
End Sub

Sub txtAddress_MouseMove (Button As Integer, Shift As Integer, X As Single
, Y As Single)
    pnlStatus.Caption = "  Information data "
End Sub

Sub txtDate_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 124, txtDate
    End If
End Sub

Sub txtDate_LostFocus ()
'    Input_Data 124, txtDate
End Sub

Sub txtDate_MouseMove (Button As Integer, Shift As Integer, X As Single, Y
 As Single)
    pnlStatus.Caption = "  Information data "
End Sub

Sub txtDL_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     DeadLoad(CProj) = CSng(txtDL.Text)
     txtLL.SetFocus
    End If
End Sub

Sub txtDL_MouseMove (Button As Integer, Shift As Integer, X As Single, Y A
s Single)
    pnlStatus.Caption = "  Actual Dead load at complete design"
End Sub

Sub txtDLFactor_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      DeadCo = CSng(txtDLFactor.Text)
      txtLL.SetFocus
    End If
End Sub

Sub txtEngineer_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 19

```
     Input_Data 126, txtEngineer
    End If
End Sub

Sub txtEngineer_LostFocus ()
'    Input_Data 126, txtEngineer
End Sub

Sub txtEngineer_MouseMove (Button As Integer, Shift As Integer, X As Singl
e, Y As Single)
    pnlStatus.Caption = "  Information data "
End Sub

Sub txtFileName_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 121, txtFileName
    End If
End Sub

Sub txtFileName_LostFocus ()
    'Input_Data 121, txtFileName
End Sub

Sub txtFileName_MouseMove (Button As Integer, Shift As Integer, X As Singl
e, Y As Single)
    pnlStatus.Caption = "  Information data "
End Sub

Sub txtGrid_KeyDown (KeyCode As Integer, Shift As Integer)
    If Shift = 1 Then
     Shifting = True
    Else
     Shifting = False
    End If
    Select Case KeyCode
    Case 38: 'up
     txtGrid_KeyPress 13
     SendKeys "{UP}"
    Case 40: 'down
     txtGrid_KeyPress 13
     SendKeys "{DOWN}"
    End Select
End Sub

Sub txtGrid_KeyPress (KeyAscii As Integer)
    Select Case KeyAscii
    Case 13, 9:
     grdNode.Text = txtGrid.Text
     txtGrid.Visible = False
     grdNode.SetFocus
     If (KeyAscii = 9) And (grdNode.Col < (grdNode.Cols - 1)) Then
      If grdNode.Col > 1 Then
       grdNode.Col = grdNode.Col - 1
      End If
     Else
      If grdNode.Col < (grdNode.Cols - 1) Then
       grdNode.Col = grdNode.Col + 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 20

```
      End If
     End If
     UnSelect
    Case 27:
     KeyAscii = 0
     txtGrid.Visible = False
     grdNode.SetFocus
    End Select
End Sub

Sub txtInformation_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 127, txtInformation
    End If
End Sub

Sub txtInformation_LostFocus ()
'    Input_Data 127, txtInformation
End Sub

Sub txtInformation_MouseMove (Button As Integer,
ngle, Y As Single)
    pnlStatus.Caption = "  Information data "
End Sub

Sub txtLL_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     LiveLoad(CProj) = CSng(txtLL.Text)
     txtSL.SetFocus
    End If
End Sub

Sub txtLL_MouseMove (Button As Integer, Shift As
s Single)
    pnlStatus.Caption = "  Live Load"
End Sub

Sub txtLLFactor_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     LiveCo = CSng(txtLLFactor.Text)
     txtLL.SetFocus
    End If

End Sub

Sub txtLoadEnd_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 66, txtLoadEnd
    End If
End Sub

Sub txtLoadEnd_LostFocus ()
    Input_Data 66, txtLoadEnd
End Sub

Sub txtLoadStart_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 21

```
     Input_Data 65, txtLoadStart
    End If
End Sub

Sub txtLoadStart_LostFocus ()
    Input_Data 65, txtLoadStart
End Sub

Sub txtLoadX1_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 61, txtLoadX1
    End If
End Sub

Sub txtLoadX1_LostFocus ()
    Input_Data 61, txtLoadX1
End Sub

Sub txtLoadX2_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 63, txtLoadX2
    End If
End Sub

Sub txtLoadX2_LostFocus ()
    Input_Data 63, txtLoadX2
End Sub

Sub txtLoadY1_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 62, txtLoadY1
    End If
End Sub

Sub txtLoadY1_LostFocus ()
    Input_Data 62, txtLoadY1
End Sub

Sub txtLoadY2_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 64, txtLoadY2
    End If
End Sub

Sub txtLoadY2_LostFocus ()
    Input_Data 64, txtLoadY2
End Sub

Sub txtNode_Change (Index As Integer)
    'Dim N As Integer
    'Dim R As Integer
    'If txtNode(1).Text <> "" Then
    ' Select Case Index
    ' Case 1:
    '  On Error GoTo ErrorData2
    '  N = CInt(txtNode(Index).Text)
    '  R = Check_Error(1, (txtNode(Index).Text))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FRMMAIN.FRM - 22

```
      '  If R <> 0 Then
      '   Error_Msg R
      '  End If
      '  cmdNodeOK.Enabled = True
      '  If (N > 0 And N < SNodeNum(CProj)) Then
      '   txtNode(2).Text = Column(CProj, N).Co_X
      '   txtNode(3).Text = Column(CProj, N).Co_Y
      '   txtNode(4).Text = 1
      '   txtNode(5).Text = 1
      '   txtNode(6).Text = 0
      '   txtNode(7).Text = 0
      '   'CurrentNode(CProj) = N
      '  Else
      '   txtNode(2).Text = 0
      '   txtNode(3).Text = 0
      '   txtNode(4).Text = 1
      '   txtNode(5).Text = 1
      '   txtNode(6).Text = 0
      '   txtNode(7).Text = 0
      '  End If
      ' End Select
      'End If
      Exit Sub
ErrorData2: MsgBox "Error : " & Format$(Error), 0 + 48, "Data Error"
            frmMain.txtNode(Index).SetFocus
            Exit Sub
End Sub

Sub txtNode_KeyPress (Index As Integer, KeyAscii As Integer)
    If (KeyAscii = 13) And (Index < 7) Then
     txtNode(Index + 1).SetFocus
    End If
    Exit Sub
End Sub

Sub txtNode_MouseMove (Index As Integer, Button As Integer, Shift As Integ
er, X As Single, Y As Single)
    Select Case Index
    Case 1:
     pnlStatus.Caption = "  Column number must be in order,[Must be intege
r]"
    Case 2:
     pnlStatus.Caption = "  Coordinate in X axis from left to right"
    Case 3:
     pnlStatus.Caption = "  Coordinate in Y axis from top to bottom"
    Case 4:
     pnlStatus.Caption = "  Array of column in X axis ,1 is not array"
    Case 5:
     pnlStatus.Caption = "  Array of column in Y axis ,1 is not array"
    Case 6:
     pnlStatus.Caption = "  Coordinate of oposite corner column in X axis
"
    Case 7:
     pnlStatus.Caption = "  Coordinate of oposite corner column in Y axis
"
    End Select
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRINT.BAS - 1

```
Sub txtProjectName_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 122, txtProjectName
    End If
End Sub

Sub txtProjectName_MouseMove (Button As Integer, Shift As Integer, X As Si
ngle, Y As Single)
    pnlStatus.Caption = "  Information data "
End Sub

Sub txtSL_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     SuperLoad(CProj) = CSng(txtSL.Text)
     'txtDummy.SetFocus
    End If
End Sub

Sub txtSL_MouseMove (Button As Integer, Shift As Integer, X As Single, Y A
s Single)
    pnlStatus.Caption = "  Super imposed dead load use in prestress design
"
End Sub

Sub txtThick_MouseMove (Button As Integer, Shift As Integer, X As Single,
Y As Single)
      pnlStatus.Caption = "  Slab Thickness [Output]"
End Sub

Sub txtTime_KeyPress (KeyAscii As Integer)
    If KeyAscii = 13 Then
     Input_Data 123, txtTime
    End If
End Sub

Sub txtTime_MouseMove (Button As Integer, Shift As Integer, X As Single, Y
 As Single)
    pnlStatus.Caption = "  Information data "
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"LPSTRESS" Present by Sompong Kongkrapun & Siam Kunkate

Mr. Wibool Wutthiyan : ADVISOR



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LISTCOM.FRM - 1

VERSION 2.00
Begin Form frmListInfo
   AutoRedraw      =   -1  'True
   Caption         =   "Listing "
   ClientHeight    =   3690
   ClientLeft      =   1845
   ClientTop       =   1800
   ClientWidth     =   7770
   FontBold        =   0   'False
   FontItalic      =   0   'False
   FontName        =   "MS Sans Serif"
   FontSize        =   8.25
   FontStrikethru  =   0   'False
   FontUnderline   =   0   'False
   Height          =   4095
   Icon            =   (Icon)
   Left            =   1785
   LinkTopic       =   "Form1"
   MDIChild        =   -1  'True
   ScaleHeight     =   3690
   ScaleWidth      =   7770
   Top             =   1455
   Width           =   7890
   WindowState     =   2  'Maximized
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Courier"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   18
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   17
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   10
      Left            =   -60
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 2

```
      MultiLine        =   -1  'True
      ScrollBars       =   3  'Both
      TabIndex         =   9
      TabStop          =   0   'False
      Top              =   0
      Width            =   9615
   End
   Begin TextBox txtListCompute
      BackColor        =   &H00FFFFFF&
      BorderStyle      =   0  'None
      FontBold         =   0   'False
      FontItalic       =   0   'False
      FontName         =   "Arial"
      FontSize         =   8.25
      FontStrikethru   =   0   'False
      FontUnderline    =   0   'False
      ForeColor        =   &H00000000&
      Height           =   6915
      Index            =   7
      Left             =   -60
      MultiLine        =   -1  'True
      ScrollBars       =   3  'Both
      TabIndex         =   6
      TabStop          =   0   'False
      Top              =   0
      Width            =   9615
   End
   Begin TextBox txtListCompute
      BackColor        =   &H00FFFFFF&
      BorderStyle      =   0  'None
      FontBold         =   0   'False
      FontItalic       =   0   'False
      FontName         =   "Arial"
      FontSize         =   8.25
      FontStrikethru   =   0   'False
      FontUnderline    =   0   'False
      ForeColor        =   &H00000000&
      Height           =   6915
      Index            =   9
      Left             =   -60
      MultiLine        =   -1  'True
      ScrollBars       =   3  'Both
      TabIndex         =   8
      TabStop          =   0   'False
      Top              =   0
      Width            =   9615
   End
   Begin TextBox txtListCompute
      BackColor        =   &H00FFFFFF&
      BorderStyle      =   0  'None
      FontBold         =   0   'False
      FontItalic       =   0   'False
      FontName         =   "Arial"
      FontSize         =   8.25
      FontStrikethru   =   0   'False
      FontUnderline    =   0   'False
      ForeColor        =   &H00000000&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 3

```
      Height          =   6915
      Index           =   12
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   11
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   11
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   10
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   6
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   5
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 4

```
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   15
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3   'Both
      TabIndex        =   14
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0   'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   5
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3   'Both
      TabIndex        =   4
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0   'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   4
      Left            =   60
      MultiLine       =   -1  'True
      ScrollBars      =   3   'Both
      TabIndex        =   3
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0   'None
      FontBold        =   0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 5

```
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   3
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3   'Both
      TabIndex        =   2
      TabStop         =   0   'False
      Top             =   -60
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0   'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   19
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3   'Both
      TabIndex        =   18
      TabStop         =   0   'False
      Top             =   -60
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0   'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   20
      Left            =   -60
      MultiLine       =   -1  'True
      ScrollBars      =   3   'Both
      TabIndex        =   19
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 6

```
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   17
      Left            =   420
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   16
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   1
      Left            =   -60
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   1
      TabStop         =   0   'False
      Top             =   60
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   2
      Left            =   -60
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   0
      TabStop         =   0   'False
      Top             =   0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 7

```
    Width           =   9615
End
Begin TextBox txtListCompute
   BackColor        =   &H00FFFFFF&
   BorderStyle      =   0  'None
   FontBold         =   0   'False
   FontItalic       =   0   'False
   FontName         =   "Arial"
   FontSize         =   8.25
   FontStrikethru   =   0   'False
   FontUnderline    =   0   'False
   ForeColor        =   &H00000000&
   Height           =   6915
   Index            =   14
   Left             =   0
   MultiLine        =   -1  'True
   ScrollBars       =   3  'Both
   TabIndex         =   13
   TabStop          =   0   'False
   Top              =   0
   Width            =   9615
End
Begin TextBox txtListCompute
   BackColor        =   &H00FFFFFF&
   BorderStyle      =   0  'None
   FontBold         =   0   'False
   FontItalic       =   0   'False
   FontName         =   "Arial"
   FontSize         =   8.25
   FontStrikethru   =   0   'False
   FontUnderline    =   0   'False
   ForeColor        =   &H00000000&
   Height           =   6915
   Index            =   13
   Left             =   0
   MultiLine        =   -1  'True
   ScrollBars       =   3  'Both
   TabIndex         =   12
   TabStop          =   0   'False
   Top              =   0
   Width            =   9615
End
Begin TextBox txtListCompute
   BackColor        =   &H00FFFFFF&
   BorderStyle      =   0  'None
   FontBold         =   0   'False
   FontItalic       =   0   'False
   FontName         =   "Arial"
   FontSize         =   8.25
   FontStrikethru   =   0   'False
   FontUnderline    =   0   'False
   ForeColor        =   &H00000000&
   Height           =   6915
   Index            =   8
   Left             =   -60
   MultiLine        =   -1  'True
   ScrollBars       =   3  'Both
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 8

```
      TabIndex        =   7
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
   Begin TextBox txtListCompute
      BackColor       =   &H00FFFFFF&
      BorderStyle     =   0  'None
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "Arial"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00000000&
      Height          =   6915
      Index           =   16
      Left            =   0
      MultiLine       =   -1  'True
      ScrollBars      =   3  'Both
      TabIndex        =   15
      TabStop         =   0   'False
      Top             =   0
      Width           =   9615
   End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LISTCOM.FRM - 1

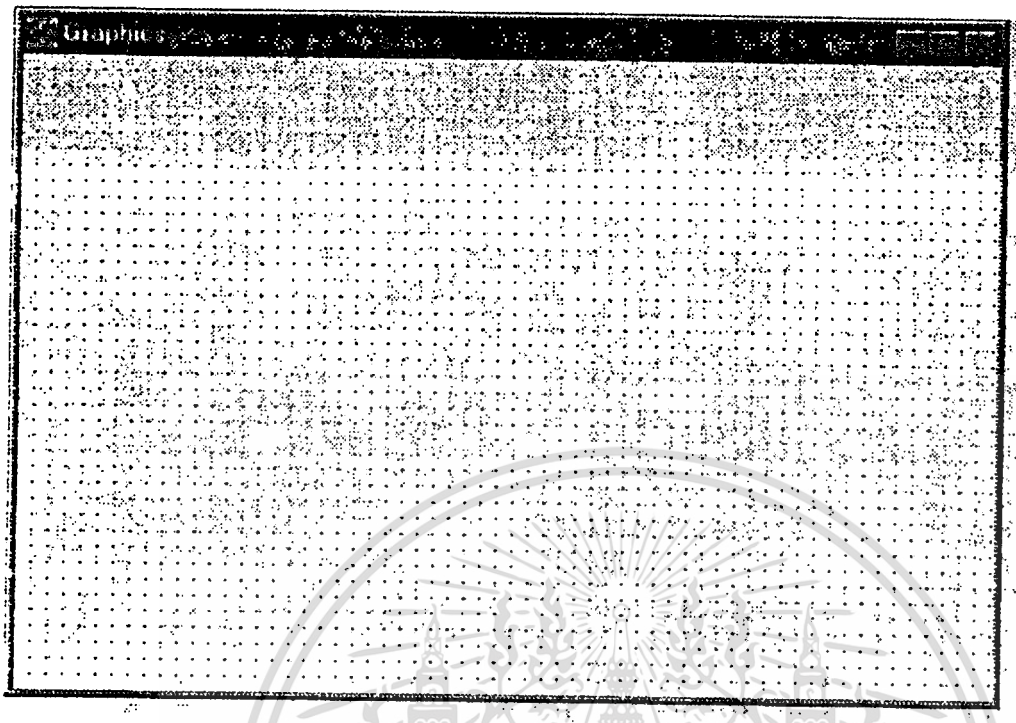
```
Sub Form_Activate ()
    frmMain.cmdPrevious.Visible = True
    frmMain.cmdNext.Visible = True
    If TextModule = 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = False
    End If
End Sub

Sub Form_Deactivate ()
    frmMain.cmdPrevious.Visible = False
    frmMain.cmdNext.Visible = False
End Sub

Sub Form_Load ()
    For I = 1 To 20
     frmList(1).txtListCompute(I).FontName = "Courier"
     frmList(1).txtListCompute(I).FontSize = 6
    Next I
End Sub

Sub Form_Resize ()
    For I = 1 To 20
    frmList(CProj).txtListCompute(I).Width = frmList(CProj).ScaleWidth
    frmList(CProj).txtListCompute(I).Height = frmList(CProj).ScaleHeight
    Next I
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTGRAPH.FRM - 1

```
VERSION 2.00
Begin Form frmOutGraphInfo
   AutoRedraw      =   -1  'True
   BackColor       =   &H00E0FFFF&
   Caption         =   "Graphics"
   ClientHeight    =   4005
   ClientLeft      =   2475
   ClientTop       =   1920
   ClientWidth     =   5250
   ForeColor       =   &H00FF00FF&
   Height          =   4410
   Icon            =   (Icon)
   Left            =   2415
   LinkTopic       =   "Form1"
   MDIChild        =   -1  'True
   ScaleHeight     =   4005
   ScaleWidth      =   5250
   Top             =   1575
   Width           =   5370
   WindowState     =   2  'Maximized
   Begin Line lneSxX
      BorderColor     =   &H00FF0000&
      Visible         =   0   'False
      X1              =   360
      X2              =   9420
      Y1              =   1680
      Y2              =   1680
   End
   Begin Line lneSxY
      BorderColor     =   &H00FF0000&
      Visible         =   0   'False
      X1              =   360
      X2              =   360
      Y1              =   0
      Y2              =   2820
   End
   Begin Line Line2
      BorderColor     =   &H00FFFF00&
      X1              =   0
      X2              =   0
      Y1              =   0
      Y2              =   6840
   End
   Begin Line Line1
      BorderColor     =   &H00FFFF00&
      X1              =   0
      X2              =   9480
      Y1              =   0
      Y2              =   0
   End
   Begin Line lneMxX
      BorderColor     =   &H00FF0000&
      Visible         =   0   'False
      X1              =   360
      X2              =   9420
      Y1              =   3900
      Y2              =   3900
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTGRAPH.FRM - 2

```
    End
    Begin Line lneMxY
       BorderColor      =   &H00FF0000&
       Visible          =   0   'False
       X1               =   360
       X2               =   360
       Y1               =   2760
       Y2               =   6420
    End
End
```



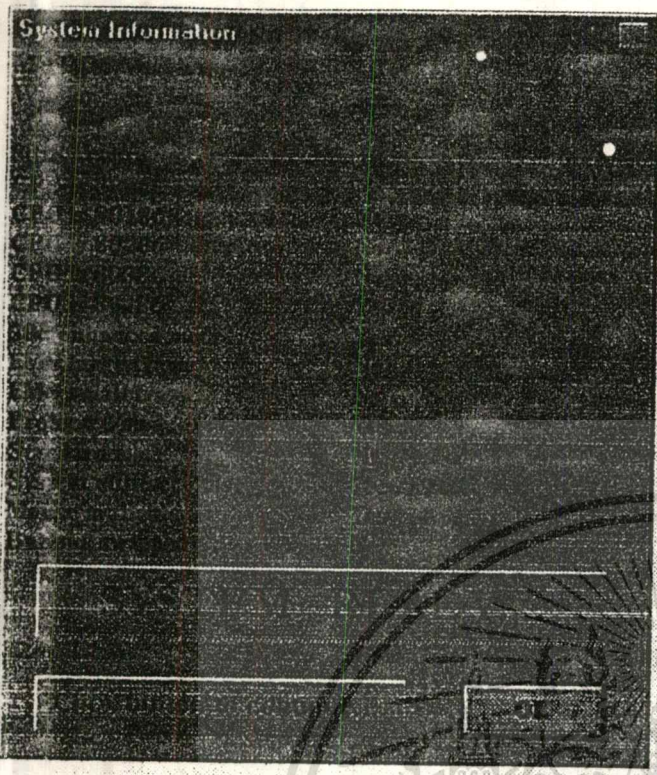
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTGRAPH.FRM - 1

```
Sub Form_Activate ()
    'For I = 0 To FileIndex
    ' If frmMain.ActiveForm.Caption = frmData(I).Caption
    '  Ready_Activate (I)
    ' End If
    'Next I
End Sub
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SYSINFO.FRM - 1

```
VERSION 2.00
Begin Form frmSysInfo
   AutoRedraw      =   -1  'True
   BackColor       =   &H00FF80FF&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "System Information"
   ClientHeight    =   5340
   ClientLeft      =   1575
   ClientTop       =   1485
   ClientWidth     =   4800
   FillColor       =   &H0000FF00&
   FillStyle       =   0  'Solid
   FontBold        =   -1  'True
   FontItalic      =   0   'False
   FontName        =   "Times New Roman"
   FontSize        =   9.75
   FontStrikethru  =   0   'False
   FontUnderline   =   0   'False
   ForeColor       =   &H00FF0000&
   Height          =   5745
   Icon            =   (Icon)
   Left            =   1515
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   Picture         =   (Bitmap)
   ScaleHeight     =   5340
   ScaleWidth      =   4800
   Top             =   1140
   Width           =   4920
   Begin SSPanel pnlMessage
      Alignment       =   1  'Left Justify - MIDDLE
      BackColor       =   &H00FFFF00&
      Caption         =   "  Click button to continue...."
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "System"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H000000FF&
      Height          =   435
      Left            =   240
      TabIndex        =   0
      Top             =   4740
      Width           =   2775
   End
   Begin SSPanel pnlSysStatus
      BackColor       =   &H00FFFF00&
      Caption         =   "SYSTEM   READY...."
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Garamond"
      FontSize        =   18
      FontStrikethru  =   0   'False.
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SYSINFO.FRM - 2

      Height          =   555
      Left            =   240
      TabIndex        =   1
      Top             =   3900
      Width           =   4275
   End
   Begin CommandButton cmdOK
      Caption         =   "OK"
      Default         =   -1  'True
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Terminal"
      FontSize        =   9
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   435
      Left            =   3420
      TabIndex        =   2
      Top             =   4740
      Width           =   1095
   End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SYSINFO.FRM - 1

```
Sub cmdOK_Click ()
    frmSysInfo.Visible = False
    Unload frmSysInfo
    'If SystemPass Then
    ' Load frmMain
    ' frmMain.Visible = True
    'Else
    ' End
    'End If
End Sub

Sub Form_Load ()
    Check_System
    'If SystemPass Then
    ' pnlSysStatus.Caption = "SYSTEM READY...."
    'Else
    ' pnlSysStatus.Caption = "SYSTEM NOT READY...."
    'End If
End Sub
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 1

''''Type''''
Type SpecialData
    FileName As String
    ProjectData As String * 100
    TimeData As String * 20
    DateData As String * 20
    AddressData As String * 100
    Engineer As String * 30
    InformData As String
End Type
Type GeneralType
    SlabOrBeam As Integer
    SlabIndex As Integer  'flat plat or flat slab
    CHeadIndex As Integer 'capital or panel or both or none
    LengthUnitIndex As Integer'length unit
    LoadUnitIndex As Integer  'load unit

    SFc_ As Single 'fc' of slab concrete
    SFs_ As Single 'fs' of strand steel
    RFs_ As Single 'fs' of reinforce steel
    CFc_ As Single 'fc' of column concrete
    SEc As Single  'E of slab concrete
    SEs As Single  'E of strand steel
    REs As Single  'E of reinforce steel
    CEc As Single  'E of column concrete
End Type
Type NodeType
    Co_X As Single
    Co_Y As Single
    End_XFlag As Integer
    End_YFlag As Integer
    SupportType As Integer
End Type
Type ColumnType
    Type As Integer
    Co_X As Single
    Co_Y As Single
    X_Flag As Integer
    Y_Flag As Integer
    c1 As Single
    c2 As Single
    DropX As Single
    DropY As Single
    DropThick As Single
    CapitalX As Single
    CapitalY As Single
    Length As Single
    SupportType As Integer
End Type
Type ElementType
    Start As Integer
    End As Integer
    CStart As Integer
    CEnd As Integer
    L1 As Single
    L2 As Single
    Length As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 2

End Type
Type SectionType
    Type As Integer
    b As Single
    d As Single
End Type
Type LoadType
    Type As Integer
    ElementRef As Integer
    Start As Integer
    End As Integer
    StartLoad As Single
    EndLoad As Single
End Type
Type SLoadType
    Type As Integer
    ElementRef As Integer
    Start As Integer
    End As Integer
    'X1 As Single
    'Y1 As Single
    'X2 As Single
    'Y2 As Single
    StartLoad As Single
    EndLoad As Single
End Type
Type ResultType
    Line_X As Integer
    Line_Y As Integer
    StartMoment As Single
    EndMoment As Single
    StartShear As Single
    EndShear As Single
    EndReaction As Single
End Type
'''


''''DECLARE FUNCTION CalL''''
Declare Function GetVersion Lib "Kernel" () As Long
Declare Function GetFreeSpace Lib "Kernel" (ByVal wFlags As Integer) As Lo
ng
Declare Function GetFreeSystemResources Lib "User" (ByVal fuSysResource As
 Integer) As Integer
Declare Function GetWinFlags Lib "Kernel" () As Long
Declare Function GetCurrentTime Lib "User" () As Long
Declare Function GetSingleClickTime Lib "User" () As Integer
Declare Function GetSystemMetrics Lib "User" (ByVal nIndex As Integer) As
Integer


''''DEFINED VARIABLES'''

'General Graphic variable
Global Icon1(10, 3) As String * 12
Global Icon2(10, 3) As String * 12
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 3

```
Global ColumnColor As Double
'Global LineX As New frmOutGraphInfoLine1

'General Tool variable

Global SelectTool(5, 10) As Integer
Global InputTextIndex As Integer
Global InputTextIndexOpen As Integer

'Constant
Global Const MaxFileNum = 1
Global frmData(MaxFileNum) As New frmDataInfo
Global frmOutGraph(MaxFileNum) As New frmOutGraphInfo
Global frmOutText(MaxFileNum) As New frmOutTextInfo
Global frmList(MaxFileNum) As New frmListInfo

'System Variable
Global FFName As String
Global SaveTag As Integer
Global DirApp As String
Global FileIndex As Integer
Global FileNames(MaxFileNum) As String
Global Colors(47) As Long
Global SystemPass As Integer
Global Pass As Integer
Global DragEnter As Integer
Global IconClick1 As Integer
Global IconClick As Integer
Global TagChange(MaxFileNum) As Integer
Global CProj As Integer
Global Num As Integer
Global Const KEY_CR = &HD
Global Const KEY_LF = &HA
Global TextModule As Integer
Global CurrentPage As Integer
Global PrintCount As Long
Global TextToPrint As String
Global TextToScreen As String
Global TextNum As Integer
'General Data For Cal
Global SlabType As Integer
Global CalIndex As Integer
Global Const s1 = " "
Global Const s2 = "  "
Global Const s3 = "   "
Global Const s4 = "    "
Global Const s5 = "     "
Global Const s6 = "      "
Global Const s7 = "       "
Global Const s8 = "        "
Global Const s9 = "         "
Global Const s10 = "          "
Global Const s15 = "               "

''''Data In File *.PSD
Global SData(MaxFileNum) As SpecialData 'Data of project
Global Node(MaxFileNum, 1000) As NodeType'Data of specific Beam
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 4

```
Global Element(MaxFileNum, 1000) As ElementType'Data of specific Beam
Global LoadData(MaxFileNum, 1000) As LoadType'Data of specific Beam
Global Section(MaxFileNum, 100) As SectionType'Data of specific Beam
Global Result(MaxFileNum, 10000) As ResultType
Global ML(MaxFileNum, 100) As Single
Global MD(MaxFileNum, 100) As Single
Global MR(MaxFileNum, 100) As Single
Global RL(MaxFileNum, 100) As Single
Global RR(MaxFileNum, 100) As Single
Global COFL(MaxFileNum, 100) As Single
Global COFR(MaxFileNum, 100) As Single
Global FEMCL(MaxFileNum, 100) As Single
Global FEMCR(MaxFileNum, 100) As Single
Global GeneralData(MaxFileNum) As GeneralType
Global KfrictionLength(MaxFileNum) As Single
Global UfrictionAngle(MaxFileNum) As Single
Global NodeNum(MaxFileNum) As Integer 'Specific Node Number
Global LoadNum(MaxFileNum) As Integer 'Specific Load Number
Global ElementNum(MaxFileNum) As Integer 'Specific Element Number
Global SNodeNum(MaxFileNum) As Integer
Global SLoadNum(MaxFileNum) As Integer
Global SElementNum(MaxFileNum) As Integer
Global StrandIndex(MaxFileNum) As Integer
Global EElement(MaxFileNum, 10000) As ElementType
Global SElement(MaxFileNum, 1, 10000) As ElementType'General data of slab
Global EleColRow(MaxFileNum, 1, 100, 100) As ElementType
Global SNode(MaxFileNum, 10000) As NodeType'General data of slab
Global Column(MaxFileNum, 10000) As ColumnType
'Global Load(MaxFileNum,1000) As SLoadType
Global SLoad(MaxFileNum, 1, 100, 100) As LoadType
Global DeadLoad(MaxFileNum) As Single
Global LiveLoad(MaxFileNum) As Single
Global SuperLoad(MaxFileNum) As Single
Global SRow(MaxFileNum) As Integer
Global SCol(MaxFileNum) As Integer
Global RowEle(100) As Integer
Global ColEle(100) As Integer
Global ElementXNum(MaxFileNum) As Integer
Global ElementYNum(MaxFileNum) As Integer
Global Thick(MaxFileNum) As Single
Global d(MaxFileNum) As Single
Global Emax(MaxFileNum) As Single
Global DeadCo As Single
Global LiveCo As Single



'Try
Global ColNum(MaxFileNum) As Integer
Global RowNum(MaxFileNum) As Integer
Global CurrentNode(MaxFileNum) As Integer
Global CurrentLoad(MaxFileNum) As Integer
'Global SNodeNum(MaxFileNum)  As Integer


''''DEFINED CONSTANTS WITH PROGRAM''''
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 5

```
Global Const SHADOW_COLOR = 8421504
Global Const BRIGHT_COLOR = 16777215
Global Const BEAM_INDEX = 0
Global Const SLAB_INDEX = 1



''''DEFINED CONSTANTS WITH SYSTEM CODE''''
''''''''''''''''''''''''''''''
' Visual Basic global constant file. This file can be loaded
' into a code module.
'
' Some constants are commented out because they have
' duplicates (e.g., NONE appears several places).
'
' If you are updating a Visual Basic application written with
' an older version, you should replace your global constants
' with the constants in this file.
'
''''''''''''''''''''''''''''''

' General

' Clipboard formats
Global Const CF_LINK = &HBF00
Global Const CF_TEXT = 1
Global Const CF_BITMAP = 2
Global Const CF_METAFILE = 3
Global Const CF_DIB = 8
Global Const CF_PALETTE = 9

' DragOver
Global Const ENTER = 0
Global Const LEAVE = 1
Global Const OVER = 2

' Drag (controls)
Global Const CANCEL = 0
Global Const BEGIN_DRAG = 1
Global Const END_DRAG = 2

' Show parameters
Global Const MODAL = 1
Global Const MODELESS = 0

' Arrange Method
' for MDI Forms
Global Const CASCADE = 0
Global Const TILE_HORIZONTAL = 1
Global Const TILE_VERTICal = 2
Global Const ARRANGE_ICONS = 3

'ZOrder Method
Global Const BRINGTOFRONT = 0
Global Const SENDTOBACK = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 6

```
' Key Codes
Global Const KEY_LBUTTON = &H1
Global Const KEY_RBUTTON = &H2
Global Const KEY_CANCEL = &H3
Global Const KEY_MBUTTON = &H4      ' NOT contiguous with L & RBUTTON
Global Const KEY_BACK = &H8
Global Const KEY_TAB = &H9
Global Const KEY_CLEAR = &HC
Global Const KEY_RETURN = &HD
Global Const KEY_SHIFT = &H10
Global Const KEY_CONTROL = &H11
Global Const KEY_MENU = &H12
Global Const KEY_PAUSE = &H13
Global Const KEY_CAPITAL = &H14
Global Const KEY_ESCAPE = &H1B
Global Const KEY_SPACE = &H20
Global Const KEY_PRIOR = &H21
Global Const KEY_NEXT = &H22
Global Const KEY_END = &H23
Global Const KEY_HOME = &H24
Global Const KEY_LEFT = &H25
Global Const KEY_UP = &H26
Global Const KEY_RIGHT = &H27
Global Const KEY_DOWN = &H28
Global Const KEY_SELECT = &H29
Global Const KEY_PRINT = &H2A
Global Const KEY_EXECUTE = &H2B
Global Const KEY_SNAPSHOT = &H2C
Global Const KEY_INSERT = &H2D
Global Const KEY_DELETE = &H2E
Global Const KEY_HELP = &H2F

' KEY_A thru KEY_Z are the same as their ASCII equivalents: 'A' thru 'Z'
' KEY_0 thru KEY_9 are the same as their ASCII equivalents: '0' thru '9'

Global Const KEY_NUMPAD0 = &H60
Global Const KEY_NUMPAD1 = &H61
Global Const KEY_NUMPAD2 = &H62
Global Const KEY_NUMPAD3 = &H63
Global Const KEY_NUMPAD4 = &H64
Global Const KEY_NUMPAD5 = &H65
Global Const KEY_NUMPAD6 = &H66
Global Const KEY_NUMPAD7 = &H67
Global Const KEY_NUMPAD8 = &H68
Global Const KEY_NUMPAD9 = &H69
Global Const KEY_MULTIPLY = &H6A
Global Const KEY_ADD = &H6B
Global Const KEY_SEPARATOR = &H6C
Global Const KEY_SUBTRACT = &H6D
Global Const KEY_DECIMAL = &H6E
Global Const KEY_DIVIDE = &H6F
Global Const KEY_F1 = &H70
Global Const KEY_F2 = &H71
Global Const KEY_F3 = &H72
Global Const KEY_F4 = &H73
Global Const KEY_F5 = &H74
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 7

```
Global Const KEY_F6 = &H75
Global Const KEY_F7 = &H76
Global Const KEY_F8 = &H77
Global Const KEY_F9 = &H78
Global Const KEY_F10 = &H79
Global Const KEY_F11 = &H7A
Global Const KEY_F12 = &H7B
Global Const KEY_F13 = &H7C
Global Const KEY_F14 = &H7D
Global Const KEY_F15 = &H7E
Global Const KEY_F16 = &H7F

Global Const KEY_NUMLOCK = &H90

' Variant VarType tags

Global Const V_EMPTY = 0
Global Const V_NULL = 1
Global Const V_INTEGER = 2
Global Const V_LONG = 3
Global Const V_Single = 4
Global Const V_DOUBLE = 5
Global Const V_CURRENCY = 6
Global Const V_DATE = 7
Global Const V_STRING = 8


' Event Parameters

' ErrNum (LinkError)
Global Const WRONG_FORMAT = 1
Global Const DDE_SOURCE_CLOSED = 6
Global Const TOO_MANY_LINKS = 7
Global Const DATA_TRANSFER_FAILED = 8

' QueryUnload
Global Const FORM_CONTROLMENU = 0
Global Const FORM_CODE = 1
Global Const APP_WINDOWS = 2
Global Const APP_TASKMANAGER = 3
Global Const FORM_MDIFORM = 4

' Properties

' Colors
Global Const BLACK = &H0&
Global Const RED = &HFF&
Global Const GREEN = &HFF00&
Global Const YELLOW = &HFFFF&
Global Const BLUE = &HFF0000
Global Const MAGENTA = &HFF00FF
Global Const CYAN = &HFFFF00
Global Const WHITE = &HFFFFFF

' System Colors
Global Const SCROLL_BARS = &H80000000          ' Scroll-bars gray area.
Global Const DESKTOP = &H80000001              ' Desktop.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 8

```
Global Const ACTIVE_TITLE_BAR = &H80000002       ' Active window caption.
Global Const INACTIVE_TITLE_BAR = &H80000003     ' Inactive window caption.
Global Const MENU_BAR = &H80000004               ' Menu background.
Global Const WINDOW_BACKGROUND = &H80000005      ' Window background.
Global Const WINDOW_FRAME = &H80000006           ' Window frame.
Global Const MENU_TEXT = &H80000007              ' Text in menus.
Global Const WINDOW_TEXT = &H80000008            ' Text in windows.
Global Const TITLE_BAR_TEXT = &H80000009         ' Text in caption, size bo
x, scroll-bar arrow box..
Global Const ACTIVE_BORDER = &H8000000A          ' Active window border.
Global Const INACTIVE_BORDER = &H8000000B        ' Inactive window border.
Global Const APPLICATION_WORKSPACE = &H8000000C ' Background color of mult
iple document interface (MDI) applications.
Global Const HIGHLIGHT = &H8000000D              ' Items selected item in a
 control.
Global Const HIGHLIGHT_TEXT = &H8000000E         ' Text of item selected in
 a control.
Global Const BUTTON_FACE = &H8000000F            ' Face shading on command
buttons.
Global Const BUTTON_SHADOW = &H80CC0010          ' Edge shading on command
buttons.
Global Const GRAY_TEXT = &H80000011              ' Grayed (disabled) text.
 This color is set to 0 if the current display driver does not support a s
olid gray color.
Global Const BUTTON_TEXT = &H80000012            ' Text on push buttons.

' Enumerated Types

' Align (picture box)
Global Const NONE = 0
Global Const ALIGN_TOP = 1
Global Const ALIGN_BOTTOM = 2

' Alignment
Global Const LEFT_JUSTIFY = 0   ' 0 - Left Justify
Global Const RIGHT_JUSTIFY = 1  ' 1 - Right Justify
Global Const CENTER = 2          ' 2 - Center

' BorderStyle (form)
'Global Const NONE = 0           ' 0 - None
Global Const FIXED_Single = 1    ' 1 - Fixed Single
Global Const SIZABLE = 2         ' 2 - Sizable (Forms only)
Global Const FIXED_DOUBLE = 3    ' 3 - Fixed Double (Forms only)

' BorderStyle (Shape and Line)
'Global Const TRANSPARENT = 0    '0 - Transparent
'Global Const SOLID = 1          '1 - Solid
'Global Const DASH = 2          ' 2 - Dash
'Global Const DOT = 3           ' 3 - Dot
'Global Const DASH_DOT = 4      ' 4 - Dash-Dot
'Global Const DASH_DOT_DOT = 5 ' 5 - Dash-Dot-Dot
'Global Const INSIDE_SOLID = 6 ' 6 - Inside Solid

' MousePointer
Global Const DEFAULT = 0         ' 0 - Default
Global Const ARROW = 1           ' 1 - Arrow
Global Const CROSSHAIR = 2       ' 2 - Cross
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 9

```
Global Const IBEAM = 3            ' 3 - I-Beam
Global Const ICON_POINTER = 4     ' 4 - Icon
Global Const SIZE_POINTER = 5     ' 5 - Size
Global Const SIZE_NE_SW = 6       ' 6 - Size NE SW
Global Const SIZE_N_S = 7         ' 7 - Size N S
Global Const SIZE_NW_SE = 8       ' 8 - Size NW SE
Global Const SIZE_W_E = 9         ' 9 - Size W E
Global Const UP_ARROW = 10        ' 10 - Up Arrow
Global Const HOURGLASS = 11       ' 11 - Hourglass
Global Const NO_DROP = 12         ' 12 - No drop

' DragMode
Global Const MANUAL = 0     ' 0 - Manual
Global Const AUTOMATIC = 1 ' 1 - Automatic

' DrawMode
Global Const BLACKNESS = 1        ' 1 - Blackness
Global Const NOT_MERGE_PEN = 2    ' 2 - Not Merge Pen
Global Const MASK_NOT_PEN = 3     ' 3 - Mask Not Pen
Global Const NOT_COPY_PEN = 4     ' 4 - Not Copy Pen
Global Const MASK_PEN_NOT = 5     ' 5 - Mask Pen Not
Global Const INVERT = 6           ' 6 - Invert
Global Const XOR_PEN = 7          ' 7 - Xor Pen
Global Const NOT_MASK_PEN = 8     ' 8 - Not Mask Pen
Global Const MASK_PEN = 9         ' 9 - Mask Pen
Global Const NOT_XOR_PEN = 10     ' 10 - Not Xor Pen
Global Const NOP = 11             ' 11 - Nop
Global Const MERGE_NOT_PEN = 12 ' 12 - Merge Not Pen
Global Const COPY_PEN = 13        ' 13 - Copy Pen
Global Const MERGE_PEN_NOT = 14 ' 14 - Merge Pen Not
Global Const MERGE_PEN = 15       ' 15 - Merge Pen
Global Const WHITENESS = 16       ' 16 - Whiteness

' DrawStyle
Global Const SOLID = 0          ' 0 - Solid
Global Const DASH = 1           ' 1 - Dash
Global Const DOT = 2            ' 2 - Dot
Global Const DASH_DOT = 3       ' 3 - Dash-Dot
Global Const DASH_DOT_DOT = 4 ' 4 - Dash-Dot-Dot
Global Const INVISIBLE = 5      ' 5 - Invisible
Global Const INSIDE_SOLID = 6 ' 6 - Inside Solid

' FillStyle
' Global Const SOLID = 0            ' 0 - Solid
Global Const TRANSPARENT = 1        ' 1 - Transparent
Global Const HORIZONTAL_LINE = 2    ' 2 - Horizontal Line
Global Const VERTICal_LINE = 3      ' 3 - VertiCal Line
Global Const UPWARD_DIAGONAL = 4    ' 4 - Upward Diagonal
Global Const DOWNWARD_DIAGONAL = 5 ' 5 - Downward Diagonal
Global Const CROSS = 6              ' 6 - Cross
Global Const DIAGONAL_CROSS = 7     ' 7 - Diagonal Cross

' LinkMode (forms and controls)
' Global Const NONE = 0          ' 0 - None
Global Const LINK_SOURCE = 1     ' 1 - Source (forms only)
Global Const LINK_AUTOMATIC = 1 ' 1 - Automatic (controls only)
Global Const LINK_MANUAL = 2     ' 2 - Manual (controls only)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 10

Global Const LINK_NOTIFY = 3     ' 3 - Notify (controls only)

' LinkMode (kept for VB1.0 compatibility, use new constants instead)
Global Const HOT = 1     ' 1 - Hot (controls only)
Global Const SERVER = 1  ' 1 - Server (forms only)
Global Const COLD = 2    ' 2 - Cold (controls only)


' SCaleMode
Global Const USER = 0         ' 0 - User
Global Const TWIPS = 1        ' 1 - Twip
Global Const POINTS = 2       ' 2 - Point
Global Const PIXELS = 3       ' 3 - Pixel
Global Const CHARACTERS = 4   ' 4 - Character
Global Const INCHES = 5       ' 5 - Inch
Global Const MILLIMETERS = 6  ' 6 - Millimeter
Global Const CENTIMETERS = 7  ' 7 - Centimeter

' ScrollBar
' Global Const NONE     = 0 ' 0 - None
Global Const HORIZONTAL = 1 ' 1 - Horizontal
Global Const VERTICal = 2    ' 2 - VertiCal
Global Const BOTH = 3        ' 3 - Both

' Shape
Global Const SHAPE_RECTANGLE = 0
Global Const SHAPE_SQUARE = 1
Global Const SHAPE_OVAL = 2
Global Const SHAPE_CIRCLE = 3
Global Const SHAPE_ROUNDED_RECTANGLE = 4
Global Const SHAPE_ROUNDED_SQUARE = 5

' WindowState
Global Const NORMAL = 0     ' 0 - Normal
Global Const MINIMIZED = 1  ' 1 - Minimized
Global Const MAXIMIZED = 2  ' 2 - Maximized

' Check Value
Global Const UNCHECKED = 0  ' 0 - Unchecked
Global Const CHECKED = 1    ' 1 - Checked
Global Const GRAYED = 2     ' 2 - Grayed

' Shift parameter masks
Global Const SHIFT_MASK = 1
Global Const CTRL_MASK = 2
Global Const ALT_MASK = 4

' Button parameter masks
Global Const LEFT_BUTTON = 1
Global Const RIGHT_BUTTON = 2
Global Const MIDDLE_BUTTON = 4

' Function Parameters
' MsgBox parameters
Global Const MB_OK = 0                    ' OK button only
Global Const MB_OKCANCEL = 1              ' OK and Cancel buttons
Global Const MB_ABORTRETRYIGNORE = 2      ' Abort, Retry, and Ignore buttons
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 11

```
Global Const MB_YESNOCANCEL = 3          ' Yes, No, and Cancel buttons
Global Const MB_YESNO = 4                ' Yes and No buttons
Global Const MB_RETRYCANCEL = 5          ' Retry and Cancel buttons

Global Const MB_ICONSTOP = 16            ' CritiCal message
Global Const MB_ICONQUESTION = 32        ' Warning query
Global Const MB_ICONEXCLAMATION = 48     ' Warning message
Global Const MB_ICONINFORMATION = 64     ' Information message

Global Const MB_APPLMODAL = 0            ' Application Modal Message Box
Global Const MB_DEFBUTTON1 = 0           ' First button is default
Global Const MB_DEFBUTTON2 = 256         ' Second button is default
Global Const MB_DEFBUTTON3 = 512         ' Third button is default
Global Const MB_SYSTEMMODAL = 4096       'System Modal

' MsgBox return values
Global Const IDOK = 1                    ' OK button pressed
Global Const IDCANCEL = 2                ' Cancel button pressed
Global Const IDABORT = 3                 ' Abort button pressed
Global Const IDRETRY = 4                 ' Retry button pressed
Global Const IDIGNORE = 5                ' Ignore button pressed
Global Const IDYES = 6                   ' Yes button pressed
Global Const IDNO = 7                    ' No button pressed

' SetAttr, Dir, GetAttr functions
Global Const ATTR_NORMAL = 0
Global Const ATTR_READONLY = 1
Global Const ATTR_HIDDEN = 2
Global Const ATTR_SYSTEM = 4
Global Const ATTR_VOLUME = 8
Global Const ATTR_DIRECTORY = 16
Global Const ATTR_ARCHIVE = 32

'Grid
'ColAlignment,FixedAlignment Properties
Global Const GRID_ALIGNLEFT = 0
Global Const GRID_ALIGNRIGHT = 1
Global Const GRID_ALIGNCENTER = 2

'Fillstyle Property
Global Const GRID_SINGLE = 0
Global Const GRID_REPEAT = 1


'Data control
'Error event Response arguments
Global Const DATA_ERRCONTINUE = 0
Global Const DATA_ERRDISPLAY = 1

'Editmode property values
Global Const DATA_EDITNONE = 0
Global Const DATA_EDITMODE = 1
Global Const DATA_EDITADD = 2

' Options property values
Global Const DATA_DENYWRITE = &H1
Global Const DATA_DENYREAD = &H2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 12

```
Global Const DATA_READONLY = &H4
Global Const DATA_APPENDONLY = &H8
Global Const DATA_INCONSISTENT = &H10
Global Const DATA_CONSISTENT = &H20
Global Const DATA_SQLPASSTHROUGH = &H40

'Validate event Action arguments
Global Const DATA_ACTIONCANCEL = 0
Global Const DATA_ACTIONMOVEFIRST = 1
Global Const DATA_ACTIONMOVEPREVIOUS = 2
Global Const DATA_ACTIONMOVENEXT = 3
Global Const DATA_ACTIONMOVELAST = 4
Global Const DATA_ACTIONADDNEW = 5
Global Const DATA_ACTIONUPDATE = 6
Global Const DATA_ACTIONDELETE = 7
Global Const DATA_ACTIONFIND = 8
Global Const DATA_ACTIONBOOKMARK = 9
Global Const DATA_ACTIONCLOSE = 10
Global Const DATA_ACTIONUNLOAD = 11


'OLE Client Control
'Actions
Global Const OLE_CREATE_EMBED = 0
Global Const OLE_CREATE_NEW = 0                 'from ole1 control
Global Const OLE_CREATE_LINK = 1
Global Const OLE_CREATE_FROM_FILE = 1           'from ole1 control
Global Const OLE_COPY = 4
Global Const OLE_PASTE = 5
Global Const OLE_UPDATE = 6
Global Const OLE_ACTIVATE = 7
Global Const OLE_CLOSE = 9
Global Const OLE_DELETE = 10
Global Const OLE_SAVE_TO_FILE = 11
Global Const OLE_READ_FROM_FILE = 12
Global Const OLE_INSERT_OBJ_DLG = 14
Global Const OLE_PASTE_SPECIAL_DLG = 15
Global Const OLE_FETCH_VERBS = 17
Global Const OLE_SAVE_TO_OLE1FILE = 18

'OLEType
Global Const OLE_LINKED = 0
Global Const OLE_EMBEDDED = 1
Global Const OLE_NONE = 3

'OLETypeAllowed
Global Const OLE_EITHER = 2

'UpdateOptions
Global Const OLE_AUTOMATIC = 0
Global Const OLE_FROZEN = 1
Global Const OLE_MANUAL = 2

'AutoActivate modes
'Note that OLE_ACTIVATE_GETFOCUS only applies to objects that
'support "inside-out" activation.  See related Verb notes below.
Global Const OLE_ACTIVATE_MANUAL = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 13

Global Const OLE_ACTIVATE_GETFOCUS = 1
Global Const OLE_ACTIVATE_SingleCLICK = 2

'SizeModes
Global Const OLE_SIZE_CLIP = 0
Global Const OLE_SIZE_STRETCH = 1
Global Const OLE_SIZE_AUTOSIZE = 2

'DisplayTypes
Global Const OLE_DISPLAY_CONTENT = 0
Global Const OLE_DISPLAY_ICON = 1

'Update Event Constants
Global Const OLE_CHANGED = 0
Global Const OLE_SAVED = 1
Global Const OLE_CLOSED = 2
Global Const OLE_RENAMED = 3

'Special Verb Values
Global Const VERB_PRIMARY = 0
Global Const VERB_SHOW = -1
Global Const VERB_OPEN = -2
Global Const VERB_HIDE = -3
Global Const VERB_INPLACEUIACTIVATE = -4
Global Const VERB_INPLACEACTIVATE = -5
'The last two verbs are for objects that support "inside-out" activation,
'meaning they can be edited in-place, and that they support being left
'in-place-active even when the input focus moves to another control or for
m.
'These objects actually have 2 levels of being active.  "InPlace Active"
'means that the object is ready for the user to click inside it and start
'working with it.  "In-Place UI-Active" means that, in addition, if the ob
ject
'has any other UI associated with it, such as floating palette windows,
'that those windows are visible and ready for use.  Any number of objects
'can be "In-Place Active" at a time, although only one can be
'"InPlace UI-Active".

'You can cause an object to move to either one of states programmatiCally
by
'setting the Verb property to the appropriate verb and setting
'Action=OLE_ACTIVATE.

'Also, if you set AutoActivate = OLE_ACTIVATE_GETFOCUS, the server will
'automatiCally be put into "InPlace UI-Active" state when the user clicks
'on or tabs into the control.

'VerbFlag Bit Masks
Global Const VERBFLAG_GRAYED = &H1
Global Const VERBFLAG_DISABLED = &H2
Global Const VERBFLAG_CHECKED = &H8
Global Const VERBFLAG_SEPARATOR = &H800

'MiscFlag Bits - Or these together as desired for special behaviors

'MEMSTORAGE causes the control to use memory to store the object while
'           it is loaded.  This is faster than the default (disk-tempfile)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 14

```
'
'           but can consume a lot of memory for objects whose data takes
'           up a lot of space, such as the bitmap for a paint program.
Global Const OLE_MISCFLAG_MEMSTORAGE = &H1

'DISABLEINPLACE overrides the control's default behavior of allowing
'           in-place activation for objects that support it.  If you
'           are having problems activating an object inplace, you can
'           force it to always activate in a separate window by setting th
is
'           bit
Global Const OLE_MISCFLAG_DISABLEINPLACE = &H2

'Common Dialog Control
'Action Property
Global Const DLG_FILE_OPEN = 1
Global Const DLG_FILE_SAVE = 2
Global Const DLG_COLOR = 3
Global Const DLG_FONT = 4
Global Const DLG_PRINT = 5
Global Const DLG_HELP = 6

'File Open/Save Dialog Flags
Global Const OFN_READONLY = &H1&
Global Const OFN_OVERWRITEPROMPT = &H2&
Global Const OFN_HIDEREADONLY = &H4&
Global Const OFN_NOCHANGEDIR = &H8&
Global Const OFN_SHOWHELP = &H10&
Global Const OFN_NOVALIDATE = &H100&
Global Const OFN_ALLOWMULTISELECT = &H200&
Global Const OFN_EXTENSIONDIFFERENT = &H400&
Global Const OFN_PATHMUSTEXIST = &H800&
Global Const OFN_FILEMUSTEXIST = &H1000&
Global Const OFN_CREATEPROMPT = &H2000&
Global Const OFN_SHAREAWARE = &H4000&
Global Const OFN_NOREADONLYRETURN = &H8000&

'Color Dialog Flags
Global Const CC_RGBINIT = &H1&
Global Const CC_FULLOPEN = &H2&
Global Const CC_PREVENTFULLOPEN = &H4&
Global Const CC_SHOWHELP = &H8&

'Fonts Dialog Flags
Global Const CF_SCREENFONTS = &H1&
Global Const CF_PRINTERFONTS = &H2&
Global Const CF_BOTH = &H3&
Global Const CF_SHOWHELP = &H4&
Global Const CF_INITTOLOGFONTSTRUCT = &H40&
Global Const CF_USESTYLE = &H80&
Global Const CF_EFFECTS = &H100&
Global Const CF_APPLY = &H200&
Global Const CF_ANSIONLY = &H400&
Global Const CF_NOVECTORFONTS = &H800&
Global Const CF_NOSIMULATIONS = &H1000&
Global Const CF_LIMITSIZE = &H2000&
Global Const CF_FIXEDPITCHONLY = &H4000&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 15

```
Global Const CF_WYSIWYG = &H8000&          'must also have CF_SCREENFONTS &
 CF_PRINTERFONTS
Global Const CF_FORCEFONTEXIST = &H10000
Global Const CF_SCalABLEONLY = &H20000
Global Const CF_TTONLY = &H40000
Global Const CF_NOFACESEL = &H80000
Global Const CF_NOSTYLESEL = &H100000
Global Const CF_NOSIZESEL = &H200000

'Printer Dialog Flags
Global Const PD_ALLPAGES = &H0&
Global Const PD_SELECTION = &H1&
Global Const PD_PAGENUMS = &H2&
Global Const PD_NOSELECTION = &H4&
Global Const PD_NOPAGENUMS = &H8&
Global Const PD_COLLATE = &H10&
Global Const PD_PRINTTOFILE = &H20&
Global Const PD_PRINTSETUP = &H40&
Global Const PD_NOWARNING = &H80&
Global Const PD_RETURNDC = &H100&
Global Const PD_RETURNIC = &H200&
Global Const PD_RETURNDEFAULT = &H400&
Global Const PD_SHOWHELP = &H800&
Global Const PD_USEDEVMODECOPIES = &H40000
Global Const PD_DISABLEPRINTTOFILE = &H80000
Global Const PD_HIDEPRINTTOFILE = &H100000

'Help Constants
Global Const HELP_CONTEXT = &H1            'Display topic in ulTopic
Global Const HELP_QUIT = &H2               'Terminate help
Global Const HELP_INDEX = &H3              'Display index
Global Const HELP_CONTENTS = &H3
Global Const HELP_HELPONHELP = &H4         'Display help on using help
Global Const HELP_SETINDEX = &H5           'Set the current Index for multi
 index help
Global Const HELP_SETCONTENTS = &H5
Global Const HELP_CONTEXTPOPUP = &H8
Global Const HELP_FORCEFILE = &H9
Global Const HELP_KEY = &H101              'Display topic for keyword in of
fabData
Global Const HELP_COMMAND = &H102
Global Const HELP_PARTIALKEY = &H105       'Call the search engine in winhe
lp

'Error Constants
Global Const CDERR_DIALOGFAILURE = -32768

Global Const CDERR_GENERALCODES = &H7FFF
Global Const CDERR_STRUCTSIZE = &H7FFE
Global Const CDERR_INITIALIZATION = &H7FFD
Global Const CDERR_NOTEMPLATE = &H7FFC
Global Const CDERR_NOHINSTANCE = &H7FFB
Global Const CDERR_LOADSTRFAILURE = &H7FFA
Global Const CDERR_FINDRESFAILURE = &H7FF9
Global Const CDERR_LOADRESFAILURE = &H7FF8
Global Const CDERR_LOCKRESFAILURE = &H7FF7
Global Const CDERR_MEMALLOCFAILURE = &H7FF6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 16

Global Const CDERR_MEMLOCKFAILURE = &H7FF5
Global Const CDERR_NOHOOK = &H7FF4

'Added for CMDIALOG.VBX
Global Const CDERR_CANCEL = &H7FF3
Global Const CDERR_NODLL = &H7FF2
Global Const CDERR_ERRPROC = &H7FF1
Global Const CDERR_ALLOC = &H7FF0
Global Const CDERR_HELP = &H7FEF

Global Const PDERR_PRINTERCODES = &H6FFF
Global Const PDERR_SETUPFAILURE = &H6FFE
Global Const PDERR_PARSEFAILURE = &H6FFD
Global Const PDERR_RETDEFFAILURE = &H6FFC
Global Const PDERR_LOADDRVFAILURE = &H6FFB
Global Const PDERR_GETDEVMODEFAIL = &H6FFA
Global Const PDERR_INITFAILURE = &H6FF9
Global Const PDERR_NODEVICES = &H6FF8
Global Const PDERR_NODEFAULTPRN = &H6FF7
Global Const PDERR_DNDMMISMATCH = &H6FF6
Global Const PDERR_CREATEICFAILURE = &H6FF5
Global Const PDERR_PRINTERNOTFOUND = &H6FF4

Global Const CFERR_CHOOSEFONTCODES = &H5FFF
Global Const CFERR_NOFONTS = &H5FFE

Global Const FNERR_FILENAMECODES = &H4FFF
Global Const FNERR_SUBCLASSFAILURE = &H4FFE
Global Const FNERR_INVALIDFILENAME = &H4FFD
Global Const FNERR_BUFFERTOOSMALL = &H4FFC

Global Const FRERR_FINDREPLACECODES = &H3FFF
Global Const CCERR_CHOOSECOLORCODES = &H2FFF


'---------------------------------------------------------
'       Table of Contents for Visual Basic Professional
'
'        1.  3-D Controls
'            (Frame/Panel/Option/Check/Command/Group Push)
'        2.  Animated Button
'        3.  Gauge Control
'        4.  Graph Control Section
'        5.  Key Status Control
'        6.  Spin Button
'        7.  MCI Control (Multimedia)
'        8.  Masked Edit Control
'        9.  Comm Control
'        10. Outline Control
'---------------------------------------------------------


'-----------------------------------------------------------------
'3D Controls
'-----------------------------------------------------------------
'Alignment (Check Box)
Global Const SSCB_TEXT_RIGHT = 0          '0 - Text to the right
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 17

```
Global Const SSCB_TEXT_LEFT = 1          '1 - Text to the left

'Alignment (Option Button)
Global Const SSOB_TEXT_RIGHT = 0         '0 - Text to the right
Global Const SSOB_TEXT_LEFT = 1          '1 - Text to the left

'Alignment (Frame)
Global Const SSFR_LEFT_JUSTIFY = 0       '0 - Left justify text
Global Const SSFR_RIGHT_JUSTIFY = 1      '1 - Right justify text
Global Const SSFR_CENTER = 2             '2 - Center text

'Alignment (Panel)
Global Const SSPN_LEFT_TOP = 0           '0 - Text to left and top
Global Const SSPN_LEFT_MIDDLE = 1        '1 - Text to left and middle
Global Const SSPN_LEFT_BOTTOM = 2        '2 - Text to left and bottom
Global Const SSPN_RIGHT_TOP = 3          '3 - Text to right and top
Global Const SSPN_RIGHT_MIDDLE = 4       '4 - Text to right and middle
Global Const SSPN_RIGHT_BOTTOM = 5       '5 - Text to right and bottom
Global Const SSPN_CENTER_TOP = 6         '6 - Text to center and top
Global Const SSPN_CENTER_MIDDLE = 7      '7 - Text to center and middle
Global Const SSPN_CENTER_BOTTOM = 8      '8 - Text to center and bottom

'Autosize (Command Button)
Global Const SS_AUTOSIZE_NONE = 0          '0 - No Autosizing
Global Const SSPB_AUTOSIZE_PICTOBUT = 1    '0 - Autosize Picture to Button
Global Const SSPB_AUTOSIZE_BUTTOPIC = 2    '0 - Autosize Button to Picture

'Autosize (Ribbon Button)
'Global Const SS_AUTOSIZE_NONE      = 0   '0 - No Autosizing
Global Const SSRI_AUTOSIZE_PICTOBUT = 1    '0 - Autosize Picture to Button
Global Const SSRI_AUTOSIZE_BUTTOPIC = 2    '0 - Autosize Button to Picture

'Autosize (Panel)
'Global Const SS_AUTOSIZE_NONE    = 0     '0 - No Autosizing
Global Const SSPN_AUTOSIZE_WIDTH = 1       '1 - Autosize Panel width to Capt
ion
Global Const SSPN_AUTOSIZE_HEIGHT = 2      '2 - Autosize Panel height to Cap
tion
Global Const SSPN_AUTOSIZE_CHILD = 3       '3 - Autosize Child to Panel

'BevelInner (Panel)
Global Const SS_BEVELINNER_NONE = 0        '0 - No Inner Bevel
Global Const SS_BEVELINNER_INSET = 1       '1 - Inset Inner Bevel
Global Const SS_BEVELINNER_RAISED = 2      '2 - Raised Inner Bevel

'BevelOuter (Panel)
Global Const SS_BEVELOUTER_NONE = 0        '0 - No Outer Bevel
Global Const SS_BEVELOUTER_INSET = 1       '1 - Inset Outer Bevel
Global Const SS_BEVELOUTER_RAISED = 2      '2 - Raised Outer Bevel

'FloodType (Panel)
Global Const SS_FLOODTYPE_NONE = 0         '0 - No flood
Global Const SS_FLOODTYPE_L_TO_R = 1       '1 - Left to light
Global Const SS_FLOODTYPE_R_TO_L = 2       '2 - Right to left
Global Const SS_FLOODTYPE_T_TO_B = 3       '3 - Top to bottom
Global Const SS_FLOODTYPE_B_TO_T = 4       '4 - Bottom to top
Global Const SS_FLOODTYPE_CIRCLE = 5       '5 - Widening circle
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 18

```
'Font3D (Panel, Command Button, Option Button, Check Box, Frame)
Global Const SS_FONT3D_NONE = 0           '0 - No 3-D text
Global Const SS_FONT3D_RAISED_LIGHT = 1   '1 - Raised with light shading
Global Const SS_FONT3D_RAISED_HEAVY = 2   '2 - Raised with heavy shading
Global Const SS_FONT3D_INSET_LIGHT = 3    '3 - Inset with light shading
Global Const SS_FONT3D_INSET_HEAVY = 4    '4 - Inset with heavy shading

'PictureDnChange (Ribbon Button)
Global Const SS_PICDN_NOCHANGE = 0        '0 - Use 'Up'bitmap with no chang
e
Global Const SS_PICDN_DITHER = 1          '1 - Dither 'Up'bitmap
Global Const SS_PICDN_INVERT = 2          '2 - Invert 'Up'bitmap

'ShadowColor (Panel, Frame)
Global Const SS_SHADOW_DARKGREY = 0       '0 - Dark grey shadow
Global Const SS_SHADOW_BLACK = 1          '1 - Black shadow

'ShadowStyle (Frame)
Global Const SS_SHADOW_INSET = 0          '0 - Shadow inset
Global Const SS_SHADOW_RAISED = 1         '1 - Shadow raised


'---------------------------------------
'Animated Button
'---------------------------------------
'Cycle property
Global Const ANI_ANIMATED = 0
Global Const ANI_MULTISTATE = 1
Global Const ANI_TWO_STATE = 2

'Click Filter property
Global Const ANI_ANYWHERE = 0
Global Const ANI_IMAGE_AND_TEXT = 1
Global Const ANI_IMAGE = 2
Global Const ANI_TEXT = 3

'PicDrawMode Property
Global Const ANI_XPOS_YPOS = 0
Global Const ANI_AUTOSIZE = 1
Global Const ANI_STRETCH = 2

'SpecialOp Property
Global Const ANI_CLICK = 1

'TextPosition Property
Global Const ANI_CENTER = 0
Global Const ANI_LEFT = 1
Global Const ANI_RIGHT = 2
Global Const ANI_BOTTON = 3
Global Const ANI_TOP = 4


'---------------------------------------
'GAUGE
'---------------------------------------
'Style Property
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 19

```
Global Const GAUGE_HORIZ = 0
Global Const GAUGE_VERT = 1
Global Const GAUGE_SEMI = 2
Global Const GAUGE_FULL = 3


'----------------------------------------
'Graph Control
'----------------------------------------
'General
Global Const G_NONE = 0
Global Const G_DEFAULT = 0

Global Const G_OFF = 0
Global Const G_ON = 1

Global Const G_MONO = 0
Global Const G_COLOR = 1

'Graph Types
Global Const G_PIE2D = 1
Global Const G_PIE3D = 2
Global Const G_BAR2D = 3
Global Const G_BAR3D = 4
Global Const G_GANTT = 5
Global Const G_LINE = 6
Global Const G_LOGLIN = 7
Global Const G_AREA = 8
Global Const G_SCATTER = 9
Global Const G_POLAR = 10
Global Const G_HLC = 11

'Colors
Global Const G_BLACK = 0
Global Const G_BLUE = 1
Global Const G_GREEN = 2
Global Const G_CYAN = 3
Global Const G_RED = 4
Global Const G_MAGENTA = 5
Global Const G_BROWN = 6
Global Const G_LIGHT_GRAY = 7
Global Const G_DARK_GRAY = 8
Global Const G_LIGHT_BLUE = 9
Global Const G_LIGHT_GREEN = 10
Global Const G_LIGHT_CYAN = 11
Global Const G_LIGHT_RED = 12
Global Const G_LIGHT_MAGENTA = 13
Global Const G_YELLOW = 14
Global Const G_WHITE = 15
Global Const G_AUTOBW = 16

'Patterns
Global Const G_SOLID = 0
Global Const G_HOLLOW = 1
Global Const G_HATCH1 = 2
Global Const G_HATCH2 = 3
Global Const G_HATCH3 = 4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 20

Global Const G_HATCH4 = 5
Global Const G_HATCH5 = 6
Global Const G_HATCH6 = 7
Global Const G_BITMAP1 = 16
Global Const G_BITMAP2 = 17
Global Const G_BITMAP3 = 18
Global Const G_BITMAP4 = 19
Global Const G_BITMAP5 = 20
Global Const G_BITMAP6 = 21
Global Const G_BITMAP7 = 22
Global Const G_BITMAP8 = 23
Global Const G_BITMAP9 = 24
Global Const G_BITMAP10 = 25
Global Const G_BITMAP11 = 26
Global Const G_BITMAP12 = 27
Global Const G_BITMAP13 = 28
Global Const G_BITMAP14 = 29
Global Const G_BITMAP15 = 30
Global Const G_BITMAP16 = 31

'Symbols
Global Const G_CROSS_PLUS = 0
Global Const G_CROSS_TIMES = 1
Global Const G_TRIANGLE_UP = 2
Global Const G_SOLID_TRIANGLE_UP = 3
Global Const G_TRIANGLE_DOWN = 4
Global Const G_SOLID_TRIANGLE_DOWN = 5
Global Const G_SQUARE = 6
Global Const G_SOLID_SQUARE = 7
Global Const G_DIAMOND = 8
Global Const G_SOLID_DIAMOND = 9

'Line Styles
'Global Const G_SOLID = 0
Global Const G_DASH = 1
Global Const G_DOT = 2
Global Const G_DASHDOT = 3
Global Const G_DASHDOTDOT = 4

'Grids
Global Const G_HORIZONTAL = 1
Global Const G_VERTICal = 2

'Statistics
Global Const G_MEAN = 1
Global Const G_MIN_MAX = 2
Global Const G_STD_DEV = 4
Global Const G_BEST_FIT = 8

'Data Arrays
Global Const G_GRAPH_DATA = 1
Global Const G_COLOR_DATA = 2
Global Const G_EXTRA_DATA = 3
Global Const G_LABEL_TEXT = 4
Global Const G_LEGEND_TEXT = 5
Global Const G_PATTERN_DATA = 6
Global Const G_SYMBOL_DATA = 7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 21

Global Const G_XPOS_DATA = 8
Global Const G_ALL_DATA = 9

'Draw Mode
Global Const G_NO_ACTION = 0
Global Const G_CLEAR = 1
Global Const G_DRAW = 2
Global Const G_BLIT = 3
Global Const G_COPY = 4
Global Const G_PRINT = 5
Global Const G_WRITE = 6

'Print Options
Global Const G_BORDER = 2

'Pie Chart Options                '
Global Const G_NO_LINES = 1
Global Const G_COLORED = 2
Global Const G_PERCENTS = 4

'Bar Chart Options                '
'Global Const G_HORIZONTAL = 1
Global Const G_STACKED = 2
Global Const G_PERCENTAGE = 4
Global Const G_Z_CLUSTERED = 6

'Gantt Chart Options              '
Global Const G_SPACED_BARS = 1

'Line/Polar Chart Options         '
Global Const G_SYMBOLS = 1
Global Const G_STICKS = 2
Global Const G_LINES = 4

'Area Chart Options               '
Global Const G_ABSOLUTE = 1
Global Const G_PERCENT = 2

'HLC Chart Options                '
Global Const G_NO_CLOSE = 1
Global Const G_NO_HIGH_LOW = 2


'---------------------------------------
'Key Status Control
'---------------------------------------
'Style
Global Const KEYSTAT_CAPSLOCK = 0
Global Const KEYSTAT_NUMLOCK = 1
Global Const KEYSTAT_INSERT = 2
Global Const KEYSTAT_SCROLLLOCK = 3


'---------------------------------------
'MCI Control (Multimedia)
'---------------------------------------
'NOTE:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PRESTRES.GBL - 22

'Please use the updated Multimedia constants
'in the WINMMSYS.TXT file from the \VB\WINAPI
'subdirectory.

'Mode Property
'Global Const MCI_MODE_NOT_OPEN = 11
'Global Const MCI_MODE_STOP = 12
'Global Const MCI_MODE_PLAY = 13
'Global Const MCI_MODE_RECORD = 14
'Global Const MCI_MODE_SEEK = 15
'Global Const MCI_MODE_PAUSE = 16
'Global Const MCI_MODE_READY = 17

'NotifyValue Property
'Global Const MCI_NOTIFY_SUCCESSFUL = 1
'Global Const MCI_NOTIFY_SUPERSEDED = 2
'Global Const MCI_ABORTED = 4
'Global Const MCI_FAILURE = 8

'Orientation Property
'Global Const MCI_ORIENT_HORZ = 0
'Global Const MCI_ORIENT_VERT = 1

'RecordMode Porperty
'Global Const MCI_RECORD_INSERT = 0
'Global Const MCI_RECORD_OVERWRITE = 1

'TimeFormat Property
'Global Const MCI_FORMAT_MILLISECONDS = 0
'Global Const MCI_FORMAT_HMS = 1
'Global Const MCI_FORMAT_MSF = 2
'Global Const MCI_FORMAT_FRAMES = 3
'Global Const MCI_FORMAT_SMPTE_24 = 4
'Global Const MCI_FORMAT_SMPTE_25 = 5
'Global Const MCI_FORMAT_SMPTE_30 = 6
'Global Const MCI_FORMAT_SMPTE_30DROP = 7
'Global Const MCI_FORMAT_BYTES = 8
'Global Const MCI_FORMAT_SAMPLES = 9
'Global Const MCI_FORMAT_TMSF = 10


'------------------------------------
'Spin Button
'------------------------------------
'SpinOrientation
Global Const SPIN_VERTICal = 0
Global Const SPIN_HORIZONTAL = 1


'------------------------------------
'Masked Edit Control
'------------------------------------
'ClipMode
Global Const ME_INCLIT = 0
Global Const ME_EXCLIT = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 23

```
'---------------------------------------
'Comm Control
'---------------------------------------
'Handshaking
Global Const MSCOMM_HANDSHAKE_NONE = 0
Global Const MSCOMM_HANDSHAKE_XONXOFF = 1
Global Const MSCOMM_HANDSHAKE_RTS = 2
Global Const MSCOMM_HANDSHAKE_RTSXONXOFF = 3

'Event constants
Global Const MSCOMM_EV_SEND = 1
Global Const MSCOMM_EV_RECEIVE = 2
Global Const MSCOMM_EV_CTS = 3
Global Const MSCOMM_EV_DSR = 4
Global Const MSCOMM_EV_CD = 5
Global Const MSCOMM_EV_RING = 6
Global Const MSCOMM_EV_EOF = 7

'Error code constants
Global Const MSCOMM_ER_BREAK = 1001
Global Const MSCOMM_ER_CTSTO = 1002
Global Const MSCOMM_ER_DSRTO = 1003
Global Const MSCOMM_ER_FRAME = 1004
Global Const MSCOMM_ER_OVERRUN = 1006
Global Const MSCOMM_ER_CDTO = 1007
Global Const MSCOMM_ER_RXOVER = 1008
Global Const MSCOMM_ER_RXPARITY = 1009
Global Const MSCOMM_ER_TXFULL = 1010


'---------------------------------------
' MAPI SESSION CONTROL CONSTANTS
'---------------------------------------
'Action
Global Const SESSION_SIGNON = 1
Global Const SESSION_SIGNOFF = 2


'---------------------------------------
' MAPI MESSAGE CONTROL CONSTANTS
'---------------------------------------
'Action
Global Const MESSAGE_FETCH = 1                  ' Load all messages from messag
e store
Global Const MESSAGE_SENDDLG = 2                ' Send mail bring up default ma
pi dialog
Global Const MESSAGE_SEND = 3                   ' Send mail without default map
i dialog
Global Const MESSAGE_SAVEMSG = 4                ' Save message in the compose b
uffer
Global Const MESSAGE_COPY = 5                   ' Copy current message to compo
se buffer
Global Const MESSAGE_COMPOSE = 6                ' Initialize compose buffer (pr
evious
                                                ' data is lost
Global Const MESSAGE_REPLY = 7                  ' Fill Compose buffer as REPLY
Global Const MESSAGE_REPLYALL = 8               ' Fill Compose buffer as REPLY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 24

```
ALL
Global Const MESSAGE_FORWARD = 9            ' Fill Compose buffer as FORWAR
D
Global Const MESSAGE_DELETE = 10            ' Delete current message
Global Const MESSAGE_SHOWADBOOK = 11        ' Show Address book
Global Const MESSAGE_SHOWDETAILS = 12       ' Show details of the current r
ecipient
Global Const MESSAGE_RESOLVENAME = 13       ' Resolve the display name of t
he recipient
Global Const RECIPIENT_DELETE = 14            ' Fill Compose buffer as FOR
WARD
Global Const ATTACHMENT_DELETE = 15           ' Delete current message


'---------------------------------------
'  ERROR CONSTANT DECLARATIONS (MAPI CONTROLS)
'---------------------------------------
Global Const SUCCESS_SUCCESS = 32000
Global Const MAPI_USER_ABORT = 32001
Global Const MAPI_E_FAILURE = 32002
Global Const MAPI_E_LOGIN_FAILURE = 32003
Global Const MAPI_E_DISK_FULL = 32004
Global Const MAPI_E_INSUFFICIENT_MEMORY = 32005
Global Const MAPI_E_ACCESS_DENIED = 32006
Global Const MAPI_E_TOO_MANY_SESSIONS = 32008
Global Const MAPI_E_TOO_MANY_FILES = 32009
Global Const MAPI_E_TOO_MANY_RECIPIENTS = 32010
Global Const MAPI_E_ATTACHMENT_NOT_FOUND = 32011
Global Const MAPI_E_ATTACHMENT_OPEN_FAILURE = 32012
Global Const MAPI_E_ATTACHMENT_WRITE_FAILURE = 32013
Global Const MAPI_E_UNKNOWN_RECIPIENT = 32014
Global Const MAPI_E_BAD_RECIPTYPE = 32015
Global Const MAPI_E_NO_MESSAGES = 32016
Global Const MAPI_E_INVALID_MESSAGE = 32017
Global Const MAPI_E_TEXT_TOO_LARGE = 32018
Global Const MAPI_E_INVALID_SESSION = 32019
Global Const MAPI_E_TYPE_NOT_SUPPORTED = 32020
Global Const MAPI_E_AMBIGUOUS_RECIPIENT = 32021
Global Const MAPI_E_MESSAGE_IN_USE = 32022
Global Const MAPI_E_NETWORK_FAILURE = 32023
Global Const MAPI_E_INVALID_EDITFIELDS = 32024
Global Const MAPI_E_INVALID_RECIPS = 32025
Global Const MAPI_E_NOT_SUPPORTED = 32026

Global Const CONTROL_E_SESSION_EXISTS = 32050
Global Const CONTROL_E_INVALID_BUFFER = 32051
Global Const CONTROL_E_INVALID_READ_BUFFER_ACTION = 32052
Global Const CONTROL_E_NO_SESSION = 32053
Global Const CONTROL_E_INVALID_RECIPIENT = 32054
Global Const CONTROL_E_INVALID_COMPOSE_BUFFER_ACTION = 32055
Global Const CONTROL_E_FAILURE = 32056
Global Const CONTROL_E_NO_RECIPIENTS = 32057
Global Const CONTROL_E_NO_ATTACHMENTS = 32058


'---------------------------------------
'  MISCELLANEOUS GLOBAL CONSTANT DECLARATIONS (MAPI CONTROLS)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 25

```
'-----------------------------------------
Global Const RECIPTYPE_ORIG = 0
Global Const RECIPTYPE_TO = 1
Global Const RECIPTYPE_CC = 2
Global Const RECIPTYPE_BCC = 3

Global Const ATTACHTYPE_DATA = 0
Global Const ATTACHTYPE_EOLE = 1
Global Const ATTACHTYPE_SOLE = 2


'-------------------------------------------------
'  Outline
'-------------------------------------------------
' PictureType
Global Const MSOUTLINE_PICTURE_CLOSED = 0
Global Const MSOUTLINE_PICTURE_OPEN = 1
Global Const MSOUTLINE_PICTURE_LEAF = 2

'Outline Control Error Constants
Global Const MSOUTLINE_BADPICFORMAT = 32000
Global Const MSOUTLINE_BADINDENTATION = 32001
Global Const MSOUTLINE_MEM = 32002
Global Const MSOUTLINE_PARENTNOTEXPANDED = 32003

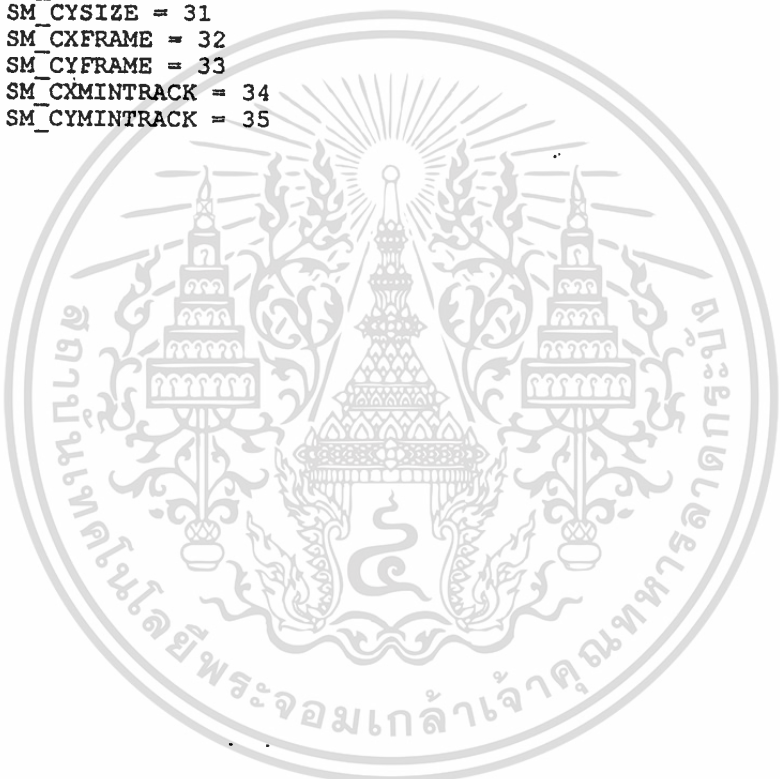

''''WinFlag''''
Global Const WF_PMODE = &H1
Global Const WF_CPU286 = &H2
Global Const WF_CPU386 = &H4
Global Const WF_CPU486 = &H8
Global Const WF_STANDARD = &H10
Global Const WF_WIN286 = &H10
Global Const WF_ENHANCED = &H20
Global Const WF_WIN386 = &H20
Global Const WF_CPU086 = &H40
Global Const WF_CPU186 = &H80
Global Const WF_LARGEFRAME = &H100
Global Const WF_SMALLFRAME = &H200
Global Const WF_80x87 = &H400

''''SystemMetric''''
Global Const SM_CXSCREEN = 0
Global Const SM_CYSCREEN = 1
Global Const SM_CXVSCROLL = 2
Global Const SM_CYHSCROLL = 3
Global Const SM_CYCAPTION = 4
Global Const SM_CXBORDER = 5
Global Const SM_CYBORDER = 6
Global Const SM_CXDLGFRAME = 7
Global Const SM_CYDLGFRAME = 8
Global Const SM_CYVTHUMB = 9
Global Const SM_CXHTHUMB = 10
Global Const SM_CXICON = 11
Global Const SM_CYICON = 12
Global Const SM_CXCURSOR = 13
Global Const SM_CYCURSOR = 14
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRESTRES.GBL - 26

```
Global Const SM_CYMENU = 15
Global Const SM_CXFULLSCREEN = 16
Global Const SM_CYFULLSCREEN = 17
Global Const SM_CYKANJIWINDOW = 18
Global Const SM_MOUSEPRESENT = 19
Global Const SM_CYVSCROLL = 20
Global Const SM_CXHSCROLL = 21
Global Const SM_DEBUG = 22
Global Const SM_SWAPBUTTON = 23
Global Const SM_RESERVED1 = 24
Global Const SM_RESERVED2 = 25
Global Const SM_RESERVED3 = 26
Global Const SM_RESERVED4 = 27
Global Const SM_CXMIN = 28
Global Const SM_CYMIN = 29
Global Const SM_CXSIZE = 30
Global Const SM_CYSIZE = 31
Global Const SM_CXFRAME = 32
Global Const SM_CYFRAME = 33
Global Const SM_CXMINTRACK = 34
Global Const SM_CYMINTRACK = 35
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 1

```
Function Find_Bottom (BottomY As Integer, SC As Integer, XY As Integer) As
 Integer
    Dim MinY As Single
    Dim MaxY As Single
    Dim v As Integer
    Dim I As Integer
    Dim Found As Integer
    v = False
    Found = False
    MinY = Column(CProj, BottomY).Co_Y
    For I = 1 To SNodeNum(CProj)
     If (Column(CProj, I).Co_Y > MinY) And In_Length(XY, I, Column(CProj,
SC).Co_X - Ec, Column(CProj, SC).Co_X + Ec) Then 'find column inlength and
 more than column
      If Not v Then
       MaxY = Column(CProj, I).Co_Y
       Find_Bottom = I
       v = True
       Found = True
      End If
      If Column(CProj, I).Co_Y <= MaxY Then 'find nearest column
       MaxY = Column(CProj, I).Co_Y
       Find_Bottom = I
       Found = True
      End If
     Else
      If Not Found Then
       Find_Bottom = 0        'Find_Bottom = 0 is Not Founded
      End If
     End If
    Next I
End Function

Function FInd_Left (LeftX As Integer, SC As Integer, XY As Integer) As Int
eger
    Dim MinX As Single
    Dim MaxX As Single
    Dim v As Integer
    Dim I As Integer
    Dim Found As Integer
    v = False
    Found = False
    MinX = Column(CProj, LeftX).Co_X
    For I = 1 To SNodeNum(CProj)
     If (Column(CProj, I).Co_X > MinX) And In_Length(XY, I, Column(CProj,
SC).Co_Y - Ec, Column(CProj, SC).Co_Y + Ec) Then
      If Not v Then
       MaxX = Column(CProj, I).Co_X
       FInd_Left = I
       v = True
       Found = True
      End If
      If Column(CProj, I).Co_X <= MaxX Then
       MaxX = Column(CProj, I).Co_X
       FInd_Left = I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 2

```
        Found = True
       End If
      Else
       If Not Found Then
        FInd_Left = 0
       End If
      End If
     Next I
End Function

Function Find_Right (RightX As Integer, SC As Integer, XY As Integer) As I
nteger
     Dim MinX As Single
     Dim MaxX As Single
     Dim I  As Integer
     Dim v As Integer
     Dim Found As Integer
     v = False
     Found = False
     MaxX = Column(CProj, RightX).Co_X
     For I = 1 To SNodeNum(CProj)
      If (Column(CProj, I).Co_X < MaxX) And In_Length(XY, I, Column(CProj,
SC).Co_Y - Ec, Column(CProj, SC).Co_Y + Ec) Then
       If Not v Then
        MinX = Column(CProj, I).Co_X
        Find_Right = I
        Found = True
        v = True
       End If
       If Column(CProj, I).Co_X >= MinX Then
        MinX = Column(CProj, I).Co_X
        Find_Right = I
        Found = True
       End If
      Else
       If Not Found Then
         Find_Right = 0
        End If
      End If
     Next I
End Function

Function Find_Top (BottomY As Integer, SC As Integer, XY As Integer) As In
teger
     Dim MinY As Single
     Dim MaxY As Single
     Dim v As Integer
     Dim I As Integer
     Dim Found As Integer
     v = False
     Found = False
     MaxY = Column(CProj, BottomY).Co_Y
     For I = 1 To SNodeNum(CProj)
      If (Column(CProj, I).Co_Y < MaxY) And In_Length(XY, I, Column(CProj,
SC).Co_X - Ec, Column(CProj, SC).Co_X + Ec) Then
       If Not v Then
        MinY = Column(CProj, I).Co_Y
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 3

```
        Find_Top = I
        Found = True
        v = True
       End If
       If Column(CProj, I).Co_Y >= MinY Then
        MinY = Column(CProj, I).Co_Y
        Find_Top = I
        Found = True
       End If
      Else
       If Not Found Then
        Find_Top = 0
       End If
      End If
     Next I
End Function

Function Find_X (X As Integer) As Integer
    Dim MinX As Single
    Dim v As Integer
    v = False
    If X = 0 Then
     For I = 1 To SNodeNum(CProj)
      If Column(CProj, I).Y_Flag = 1 Then
       If Not v Then
        MaxX = Column(CProj, I).Co_X
        Find_X = I
        v = True
       End If
       If Column(CProj, I).Co_X < MaxX Then
        Find_X = I
       End If
      End If
     Next I
    ElseIf X > 0 Then
     MinX = Column(CProj, X).Co_X
     For I = 1 To SNodeNum(CProj)
      If Column(CProj, I).Y_Flag = 1 Then
       If Column(CProj, I).Co_X > MinX Then
        If Not v Then
         MaxX = Column(CProj, I).Co_X
         Find_X = I
         v = True
        End If
        If Column(CProj, I).Co_X < MaxX Then
         MaxX = Column(CProj, I).Co_X
         Find_X = I
        End If
       End If
      End If
     Next I
    End If
End Function

Function Find_Y (Y As Integer) As Integer
    Dim MinY As Single
    Dim v As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 4

```
    v = False
    If Y = 0 Then
     For I = 1 To SNodeNum(CProj)
      If Column(CProj, I).X_Flag = 1 Then
       If Not v Then
        MaxY = Column(CProj, I).Co_Y
        Find_Y = I
        v = True
       End If
       If Column(CProj, I).Co_Y < MaxY Then
        Find_Y = I
       End If
      End If
     Next I
    ElseIf Y > 0 Then
     MinY = Column(CProj, Y).Co_Y
     For I = 1 To SNodeNum(CProj)
      If Column(CProj, I).X_Flag = 1 Then
       If Column(CProj, I).Co_Y > MinY Then
        If Not v Then
         MaxY = Column(CProj, I).Co_Y
         Find_Y = I
         v = True
        End If
        If Column(CProj, I).Co_Y < MaxY Then
         MaxY = Column(CProj, I).Co_Y
         Find_Y = I
        End If
       End If
      End If
     Next I
    End If
End Function

Function In_Length (XY, I As Integer, Min As Single, Max As Single) As Int
eger
    If XY Then
     If ((Column(CProj, I).Co_X >= Min) And (Column(CProj, I).Co_X <= Max)
) Then
      In_Length = True
     Else
      In_Length = False
     End If
    ElseIf Not (XY) Then
     If ((Column(CProj, I).Co_Y >= Min) And (Column(CProj, I).Co_Y <= Max)
) Then
      In_Length = True
     Else
      In_Length = False
     End If
    End If
End Function

Function Max (a As Single, b As Single)
    Max = a
    If b > a Then
     Max = b
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 5

```
    End If
End Function

Sub Show_Element ()
    Dim axis As Integer
    Dim N As Integer
    N = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      N = N + 1
      frmMain.grdElement.Row = N
      frmMain.grdElement.Col = 1: frmMain.grdElement.Text = "X"
      frmMain.grdElement.Col = 2: frmMain.grdElement.Text = Format$(I)
      frmMain.grdElement.Col = 3: frmMain.grdElement.Text = Format$(J)
      frmMain.grdElement.Col = 4: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 0, I, J).Start)
      frmMain.grdElement.Col = 5: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 0, I, J).End)
      frmMain.grdElement.Col = 6: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 0, I, J).Length)
      frmMain.grdElement.Col = 7: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 0, I, J).L1)
      frmMain.grdElement.Col = 8: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 0, I, J).L2)
     Next J
    Next I
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      N = N + 1
      frmMain.grdElement.Row = N
      frmMain.grdElement.Col = 1: frmMain.grdElement.Text = "Y"
      frmMain.grdElement.Col = 2: frmMain.grdElement.Text = Format$(I)
      frmMain.grdElement.Col = 3: frmMain.grdElement.Text = Format$(J)
      frmMain.grdElement.Col = 4: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 1, I, J).Start)
      frmMain.grdElement.Col = 5: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 1, I, J).End)
      frmMain.grdElement.Col = 6: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 1, I, J).Length)
      frmMain.grdElement.Col = 7: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 1, I, J).L1)
      frmMain.grdElement.Col = 8: frmMain.grdElement.Text = Format$(EleCol
Row(CProj, 1, I, J).L2)
     Next J
    Next I
End Sub

Sub Trans_BElement (ColOrRow As Integer, ByVal ColRow As Integer) ' ColOrR
ow = True Cal Column Dir
    Const YE = 1
    Const XE = 0
    If ColOrRow Then
     For I = 1 To ColEle(ColRow)
      Element(CProj, I) = EleColRow(CProj, YE, ColRow, I)
      LoadData(CProj, I) = SLoad(CProj, YE, ColRow, I)
      LoadData(CProj, I).ElementRef = I
      Node(CProj, I).Co_X = Column(CProj, Element(CProj, I).Start).Co_X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 6

```
        Node(CProj, I).Co_Y = Column(CProj, Element(CProj, I).Start).Co_Y
        Node(CProj, I).SupportType = Column(CProj, Element(CProj, I).Start).
SupportType
        Element(CProj, I).CStart = EleColRow(CProj, YE, ColRow, I).Start
        Element(CProj, I).Start = I
        Node(CProj, I + 1).Co_X = Column(CProj, Element(CProj, I).End).Co_X
        Node(CProj, I + 1).Co_Y = Column(CProj, Element(CProj, I).End).Co_Y
        Node(CProj, I + 1).SupportType = Column(CProj, Element(CProj, I).End
).SupportType
        Element(CProj, I).CEnd = EleColRow(CProj, YE, ColRow, I).End
        Element(CProj, I).End = I + 1
      Next I
      NodeNum(CProj) = ColEle(ColRow) + 1
      ElementNum(CProj) = ColEle(ColRow)
      LoadNum(CProj) = ColEle(ColRow)
    ElseIf Not ColOrRow Then
      For I = 1 To RowEle(ColRow)
        Element(CProj, I) = EleColRow(CProj, XE, ColRow, I)
        LoadData(CProj, I) = SLoad(CProj, XE, ColRow, I)
        LoadData(CProj, I).ElementRef = I
        Node(CProj, I).Co_X = Column(CProj, Element(CProj, I).Start).Co_X
        Node(CProj, I).Co_Y = Column(CProj, Element(CProj, I).Start).Co_Y
        Node(CProj, I).SupportType = Column(CProj, Element(CProj, I).Start).
SupportType
        Element(CProj, I).CStart = EleColRow(CProj, XE, ColRow, I).Start
        Element(CProj, I).Start = I
        Node(CProj, I + 1).Co_X = Column(CProj, Element(CProj, I).End).Co_X
        Node(CProj, I + 1).Co_Y = Column(CProj, Element(CProj, I).End).Co_Y
        Node(CProj, I + 1).SupportType = Column(CProj, Element(CProj, I).End
).SupportType
        Element(CProj, I).CEnd = EleColRow(CProj, XE, ColRow, I).End
        Element(CProj, I).End = I + 1
      Next I
      NodeNum(CProj) = RowEle(ColRow) + 1
      ElementNum(CProj) = RowEle(ColRow)
      LoadNum(CProj) = RowEle(ColRow)
    End If
End Sub

Sub Trans_Element ()
    Dim I As Integer
    Dim J  As Integer
    Dim K As Integer
    Dim EleCol As Integer
    Dim EleRow As Integer
    Dim MinX As Integer
    Dim MinY As Integer
    Dim L1 As Single
    Dim L2 As Single
    Dim SColColumn As Integer
    Dim SRowColumn As Integer
    Const Ec = 3   'ecentric of column equal 3 meters
    Const XE = 0     'Code for X axis
    Const YE = 1     'Code for Y axis
    SCol(CProj) = 0          'Number of Col
    EleCol = 0      '1-100 is Column Number
    SColColumn = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    '''' Find Element in Column & Row ,EleColRow
    For I = 1 To SNodeNum(CProj)

    '''' In Y axis
     If ((Column(CProj, I).Y_Flag = 1) And (Find_X(SColColumn) = I)) Then
 'Check Next Right Column
      SColColumn = I
      MinY = I
      SCol(CProj) = SCol(CProj) + 1
      EleCol = 0
      For J = 1 To SNodeNum(CProj)
       If In_Length(True, J, Column(CProj, SColColumn).Co_X - Ec, Column(C
Proj, SColColumn).Co_X + Ec) And (Find_Bottom(MinY, SColColumn, True) = J)
 Then 'Check Next Element
        EleCol = EleCol + 1
        EleColRow(CProj, YE, SCol(CProj), EleCol).Start = MinY
        EleColRow(CProj, YE, SCol(CProj), EleCol).End = J
        EleColRow(CProj, YE, SCol(CProj), EleCol).Length = Column(CProj, J
).Co_Y - Column(CProj, MinY).Co_Y
        MinY = J
        J = 1  '************************
       End If
      Next J
      ColEle(SCol(CProj)) = EleCol
      I = 1 '***********************
     End If
    Next I

    '''' In  Y axis
    SRow(CProj) = 0
    EleRow = 0
    SRowColumn = 0
    For I = 1 To SNodeNum(CProj)
     If ((Column(CProj, I).X_Flag = 1) And (Find_Y(SRowColumn) = I)) Then
 'Check Next Top Row
      SRowColumn = I
      MinX = I
      SRow(CProj) = SRow(CProj) + 1
      EleRow = 0
      For J = 1 To SNodeNum(CProj)
       If In_Length(False, J, Column(CProj, I).Co_Y - Ec, Column(CProj, I)
.Co_Y + Ec) And (FInd_Left(MinX, SRowColumn, False) = J) Then    'Check Ne
xt Row Element
        EleRow = EleRow + 1
        EleColRow(CProj, XE, SRow(CProj), EleRow).Start = MinX
        EleColRow(CProj, XE, SRow(CProj), EleRow).End = J
        EleColRow(CProj, XE, SRow(CProj), EleRow).Length = Column(CProj, J
).Co_X - Column(CProj, MinX).Co_X
        MinX = J
        J = 1 '*************************
       End If
      Next J
      RowEle(SRow(CProj)) = EleRow
      I = 1 '******************************
     End If
    Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 8

```
    '''' Transform To Element Only Y axis ,SElement
    ElementYNum(CProj) = 0
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      ElementYNum(CProj) = ElementYNum(CProj) + 1
      SElement(CProj, YE, ElementYNum(CProj)) = EleColRow(CProj, YE, I, J)
     Next J
    Next I

    '''' Transform To Element Only X axis ,SElement
    ElementXNum(CProj) = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      ElementXNum(CProj) = ElementXNum(CProj) + 1
      SElement(CProj, XE, ElementXNum(CProj)) = EleColRow(CProj, XE, I, J)
     Next J
    Next I

    '''' Find L1,L2 in Row
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      L1 = 0
      L2 = 0
      For K = 1 To ElementYNum(CProj)
       If (EleColRow(CProj, XE, I, J).Start = SElement(CProj, YE, K).End)
Or (EleColRow(CProj, XE, I, J).End = SElement(CProj, YE, K).End) Then
        If L1 < SElement(CProj, YE, K).Length Then
         L1 = SElement(CProj, YE, K).Length
        End If
       End If
       If (EleColRow(CProj, XE, I, J).Start = SElement(CProj, YE, K).Start
) Or (EleColRow(CProj, XE, I, J).End = SElement(CProj, YE, K).Start) Then
        If L2 < SElement(CProj, YE, K).Length Then
         L2 = SElement(CProj, YE, K).Length
        End If
       End If
      Next K
      EleColRow(CProj, XE, I, J).L1 = L1
      EleColRow(CProj, XE, I, J).L2 = L2
     Next J
    Next I

    '''' Find L1,L2 in Column
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      L1 = 0
      L2 = 0
      For K = 1 To ElementXNum(CProj)
       If (EleColRow(CProj, YE, I, J).Start = SElement(CProj, XE, K).End)
Or (EleColRow(CProj, YE, I, J).End = SElement(CProj, XE, K).End) Then
        If L1 < SElement(CProj, XE, K).Length Then
         L1 = SElement(CProj, XE, K).Length
        End If
       End If
       If (EleColRow(CProj, YE, I, J).Start = SElement(CProj, XE, K).Start
) Or (EleColRow(CProj, YE, I, J).End = SElement(CProj, XE, K).Start) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 9

```
        If L2 < SElement(CProj, XE, K).Length Then
         L2 = SElement(CProj, XE, K).Length
       End If
      End If
     Next K
     EleColRow(CProj, YE, I, J).L1 = L1
     EleColRow(CProj, YE, I, J).L2 = L2
    Next J
   Next I

   '''' Transform To All Element in project,EElement
   For I = 1 To ElementYNum(CProj)
    EElement(CProj, I) = SElement(CProj, 1, I)
   Next I
   For I = 1 To ElementXNum(CProj)
    EElement(CProj, ElementYNum(CProj) + I) = SElement(CProj, 0, I)
   Next I

   '''' Replace L1,L2
   ElementYNum(CProj) = 0
   For I = 1 To SCol(CProj)
    For J = 1 To ColEle(I)
     ElementYNum(CProj) = ElementYNum(CProj) + 1
     SElement(CProj, YE, ElementYNum(CProj)) = EleColRow(CProj, YE, I, J)
    Next J
   Next I

   '''' Transform To Element Only X axis ,SElement
   ElementXNum(CProj) = 0
   For I = 1 To SRow(CProj)
    For J = 1 To RowEle(I)
     ElementXNum(CProj) = ElementXNum(CProj) + 1
     SElement(CProj, XE, ElementXNum(CProj)) = EleColRow(CProj, XE, I, J)
    Next J
   Next I
   Show_Element
   '''' Try output element conjunction
   'Load frmOutText(CProj)
   'frmOutText(CProj).Show
   'frmOutText(CProj).Cls
   'For J = 1 To (ElementXNum(CProj) + ElementYNum(CProj))
   ' CStart = EElement(CProj, J).Start
   ' CEnd = EElement(CProj, J).End
   ' frmOutText(CProj).Print "Start : End " + Format$(CStart) + " : " + F
ormat$(CEnd)
   'Next J
   Show_DataScreenE
End Sub

Sub Trans_UniformLoad (SUniform As Single)
   Const XE = 0
   Const YE = 1
   For I = 1 To SCol(CProj)
    For J = 1 To ColEle(I)
     SLoad(CProj, YE, I, J).Type = 2
     SLoad(CProj, YE, I, J).Start = 0
     SLoad(CProj, YE, I, J).End = EleColRow(CProj, YE, I, J).Length
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSDAT.BAS - 10

```
        SLoad(CProj, YE, I, J).ElementRef = J
        SLoad(CProj, YE, I, J).StartLoad = (1 / 2) * SUniform * (EleColRow(C
Proj, YE, I, J).L2 + EleColRow(CProj, YE, I, J).L1)
      Next J
     Next I
     For I = 1 To SRow(CProj)
      For J = 1 To RowEle(I)
       SLoad(CProj, XE, I, J).Type = 2
       SLoad(CProj, XE, I, J).Start = 0
       SLoad(CProj, XE, I, J).End = EleColRow(CProj, XE, I, J).Length
       SLoad(CProj, XE, I, J).ElementRef = J
       SLoad(CProj, XE, I, J).StartLoad = (1 / 2) * SUniform * (EleColRow(C
Proj, XE, I, J).L2 + EleColRow(CProj, XE, I, J).L1)
      Next J
     Next I
End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 1

```
Sub BCal ()
    Dim I As Integer
    Cal_BEndMoment
    For I = 1 To ElementNum(CProj)
     RL(CProj, I) = Cal_BEndReaction(I, ML(CProj, I), MR(CProj, I), True)
     RR(CProj, I) = Cal_BEndReaction(I, ML(CProj, I), MR(CProj, I), False)
    Next I
End Sub

Function Cal_BBeta1 (emax As Single, eb As Single) As Single
    Cal_BBeta1 = emax / eb
End Function

Function Cal_BBeta2 (emax As Single, eb As Single) As Single
    Cal_BBeta2 = emax / eb
End Function

Function Cal_BCOF (E As Integer, LeftOrRight As Integer) As Single
    Dim N As Integer
    If LeftOrRight Then
     Cal_BCOF = (1 / 2)
     N = Element(CProj, E).Start
    ElseIf Not (LeftOrRight) Then
     Cal_BCOF = (1 / 2)
     N = Element(CProj, E).End
    End If
    If Node(CProj, N).SupportType = 3 Then
     Cal_BCOF = 0
    End If
End Function

Function Cal_BDeflection () As Single

End Function

Function Cal_BDF (E As Integer, StartOrEnd As Integer) As Single
    Dim I As Integer
    Dim SumDF As Single
    SumDF = 0
   . If StartOrEnd Then
     For I = 0 To ElementNum(CProj)
      If (Element(CProj, E).Start = Element(CProj, I).Start) Or (Element(C
Proj, E).Start = Element(CProj, I).End) Then
       SumDF = SumDF + Cal_BK(I)
      End If
     Next I
     If Not (SumDF = 0) Then
      Cal_BDF = (Cal_BK(E) / SumDF)
     Else
      Cal_BDF = 1
     End If
     If Node(CProj, Element(CProj, E).Start).SupportType = 2 Then
      Cal_BDF = 0
     End If
    ElseIf Not (StartOrEnd) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 2

```
      For I = 0 To ElementNum(CProj)
       If (Element(CProj, E).End = Element(CProj, I).Start) Or (Element(CPr
oj, E).End = Element(CProj, I).End) Then
        SumDF = SumDF + Cal_BK(I)
       End If
      Next I
      If Not (SumDF = 0) Then
      Cal_BDF = (Cal_BK(E) / SumDF)
      Else
      Cal_BDF = 1
      End If
      If Node(CProj, Element(CProj, E).End).SupportType = 2 Then
       Cal_BDF = 0
      End If
    End If
End Function

Sub Cal_BEndMoment ()
    Const ErrorCheck = .01
    Dim ErrorValue As Single
    Dim I As Integer
    ReDim FEML(ElementNum(CProj)) As Single
    ReDim FEMR(ElementNum(CProj))  As Single
    ReDim DFL(ElementNum(CProj)) As Single
    ReDim DFR(ElementNum(CProj)) As Single
    'ReDim ML(CProj,ElementNum(CProj)) As Single
    'ReDim MR(CProj,ElementNum(CProj)) As Single
    'ReDim COFL(CProj,ElementNum(CProj)) As Single
    'ReDim COFR(CProj,ElementNum(CProj)) As Single
    ReDim COML(ElementNum(CProj)) As Single
    ReDim COMR(ElementNum(CProj))  As Single
    ReDim DML(ElementNum(CProj)) As Single
    ReDim DMR(ElementNum(CProj))  As Single
    ReDim MNode(NodeNum(CProj)) As Single
    For I = 1 To ElementNum(CProj)
     FEML(I) = Cal_BFEM(I, True)
     FEMR(I) = Cal_BFEM(I, False)
     DFL(I) = Cal_BDF(I, True)
     DFR(I) = Cal_BDF(I, False)
     COFL(CProj, I) = Cal_BCOF(I, True)
     COFR(CProj, I) = Cal_BCOF(I, False)
     ML(CProj, I) = FEML(I)
     MR(CProj, I) = FEMR(I)
     DML(I) = 0
     DMR(I) = 0
     COML(I) = 0
     COMR(I) = 0
    Next I
    'Distribution Moment
    Do
    ErrorValue = 0
      'Sum DM
     For I = 1 To NodeNum(CProj)
      MNode(I) = 0
      For J = 1 To ElementNum(CProj)
       If (Element(CProj, J).Start) = I Then
        MNode(I) = (MNode(I) + ML(CProj, J))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 3

```
        ElseIf (Element(CProj, J).End = I) Then
         MNode(I) = (MNode(I) + MR(CProj, J))
        End If
       Next J
      Next I
       'DM
      For I = 1 To NodeNum(CProj)
       For J = 1 To ElementNum(CProj)
        If (Element(CProj, J).Start = I) Then
         DML(J) = (((-1) * MNode(I)) * DFL(J))
        ElseIf (Element(CProj, J).End = I) Then
         DMR(J) = (((-1) * MNode(I)) * DFR(J))
        End If
       Next J
      Next I
      For I = 1 To ElementNum(CProj)
       If Abs(ErrorValue) < Abs(DML(I)) Then
        ErrorValue = DML(I)
       End If
       If Abs(ErrorValue) < Abs(DMR(I)) Then
        ErrorValue = DMR(I)
       End If
      Next I
      'Carry over moment
      For I = 1 To ElementNum(CProj)
       COML(I) = DMR(I) * COFL(CProj, I)
       COMR(I) = DML(I) * COFR(CProj, I)
      Next I
       'Sum Moment
      If (Abs(ErrorValue) < ErrorCheck) Then
       For I = 1 To ElementNum(CProj)
        ML(CProj, I) = ML(CProj, I) + DML(I)
        MR(CProj, I) = MR(CProj, I) + DMR(I)
       Next I
      Else
       For I = 1 To ElementNum(CProj)
        ML(CProj, I) = ML(CProj, I) + DML(I) + COML(I)
        MR(CProj, I) = MR(CProj, I) + DMR(I) + COMR(I)
       Next I
      End If
     Loop Until (Abs(ErrorValue) < ErrorCheck)
     Load frmOutText(CProj)
     frmOutText(CProj).Show
     frmOutText(CProj).Cls
     For I = 1 To ElementNum(CProj)
      frmOutText(CProj).Print "Fixed end moment Left element" + Format$(I)
+ " = " + Format$(Cal_BFEM(I, True), "0.00")
      frmOutText(CProj).Print "Fixed end moment Right element" + Format$(I)
 + " = " + Format$(Cal_BFEM(I, False), "0.00")
      frmOutText(CProj).Print "DF Factor Left " + Format$(DFL(I), "0.000")
      frmOutText(CProj).Print "DF Factor Right " + Format$(DFR(I), "0.000")
      frmOutText(CProj).Print "End Moment Left " + Format$(ML(CProj, I), "0
.00")
      frmOutText(CProj).Print "End Moment Right " + Format$(MR(CProj, I), "
0.00")
     Next I
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 4

```
Function Cal_BEndReaction (E As Integer, ML As Single, MR As Single, LeftO
rRight As Integer) As Single
    Dim RL As Single
    Dim RR As Single
    Dim M As Single
    Dim p As Single
    Dim W As Single
    Dim W1 As Single
    Dim W2 As Single
    Dim a As Single
    Dim b As Single
    Dim L As Single
    RL = 0
    RR = 0
    If LeftOrRight Then
     For I = 1 To LoadNum(CProj)
      If LoadData(CProj, I).ElementRef = E Then
       If Node(CProj, Element(CProj, E).Start).SupportType = 3 Then
        Select Case LoadData(CProj, I).Type
        Case 1:
         a = LoadData(CProj, I).Start
         p = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
         RL = RL + p
        Case 2:
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W1 = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
         RL = RL + (W1 * (b - a))
        Case 3:
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W1 = LoadData(CProj, I).StartLoad
         W2 = LoadData(CProj, I).EndLoad
         L = Element(CProj, E).Length
         RL = RL + ((1 / 2) * (W1 + W2) * (b - a))
        Case 4:
         L = Element(CProj, E).Length
         M = LoadData(CProj, I).StartLoad
         RL = RL
        Case 5:
        End Select
       Else
        Select Case LoadData(CProj, I).Type
        Case 1:
         a = LoadData(CProj, I).Start
         p = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
         RL = RL + ((p * (L - a)) / L)
        Case 2:
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W1 = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
         RL = RL + ((W * (b - a) * (((b - a) / 2) + L - b)) / L)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 5

```
        Case 3:
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W1 = LoadData(CProj, I).StartLoad
         W2 = LoadData(CProj, I).EndLoad
         L = Element(CProj, E).Length
         RL = RL + ((((1 / 2) * (W1 + W2) * (b - a)) * (L - b + ((W1 + ((1
 / 3) * W2)) / (W1 + W2)))) / L)
        Case 4:
         L = Element(CProj, E).Length
         M = LoadData(CProj, I).StartLoad
         RL = RL + ((M / L) * (-1))
        Case 5:
        End Select
       End If
      End If
     Next I
     L = Element(CProj, E).Length
     If Not (Node(CProj, Element(CProj, E).End).SupportType = 3) Then
      RL = RL + (((ML + MR) / L) * (-1))
     End If
     Cal_BEndReaction = RL
    ElseIf Not (LeftOrRight) Then
     For I = 1 To LoadNum(CProj)
      If LoadData(CProj, I).ElementRef = E Then
       If Node(CProj, Element(CProj, E).End).SupportType = 3 Then
        Select Case LoadData(CProj, I).Type
        Case 1:
         a = LoadData(CProj, I).Start
         p = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
         RR = RR + p
        Case 2:
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W1 = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
         RR = RR + (W1 * (b - a))
        Case 3:
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W1 = LoadData(CProj, I).StartLoad
         W2 = LoadData(CProj, I).EndLoad
         L = Element(CProj, E).Length
         RR = RR + ((1 / 2) * (W1 + W2) * (b - a))
        Case 4:
         L = Element(CProj, E).Length
         M = LoadData(CProj, I).StartLoad
         RR = RR + 0
        Case 5:
        End Select
       Else
        Select Case LoadData(CProj, I).Type
        Case 1:
         a = LoadData(CProj, I).Start
         p = LoadData(CProj, I).StartLoad
         L = Element(CProj, E).Length
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 6

```
          RR = RR + ((p * a) / L)
         Case 2:
          a = LoadData(CProj, I).Start
          b = LoadData(CProj, I).End
          W1 = LoadData(CProj, I).StartLoad
          L = Element(CProj, E).Length
          RR = RR + ((W * (b - a) * (a + ((b - a) / 2))) / L)
         Case 3:
          a = LoadData(CProj, I).Start
          b = LoadData(CProj, I).End
          W1 = LoadData(CProj, I).StartLoad
          W2 = LoadData(CProj, I).EndLoad
          L = Element(CProj, E).Length
          RR = RR + ((((1 / 2) * (W1 + W2) * (b - a)) * (a + ((W2 + ((1 / 3
) * W1)) / (W1 + W2)))) / L)
         Case 4:
          L = Element(CProj, E).Length
          M = LoadData(CProj, I).StartLoad
          RR = RR + (M / L)
          Case 5:
         End Select
        End If
       End If
      Next I
      L = Element(CProj, E).Length
      If Not (Node(CProj, Element(CProj, E).End).SupportType <> 3) Then
       RR = RR + ((ML + MR) / L)
      End If
      Cal_BEndReaction = RR
     End If
End Function

Function Cal_BFEM (E As Integer, LeftOrRight As Integer) As Single
     Dim I As Integer
     Dim a As Single
     Dim b As Single
     Dim C As Single
     Dim L As Single
     Dim W As Single
     Dim W1 As Single
     Dim W2 As Single
     Dim M As Single
     Dim FEML As Single
     Dim FEMR As Single
     FEML = 0
     FEMR = 0
     If LeftOrRight Then
      For I = 1 To LoadNum(CProj)
       If LoadData(CProj, I).ElementRef = E Then
        If Node(CProj, Element(CProj, E).Start).SupportType = 3 Then
         Select Case LoadData(CProj, I).Type
         Case 1: 'Point Load
          W = LoadData(CProj, I).StartLoad
          a = LoadData(CProj, I).Start
          b = Element(CProj, E).Length - LoadData(CProj, I).Start
          L = Element(CProj, E).Length
          FEML = FEML - (W * a)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 7

```
        Case 2: 'Uniform Load
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         L = Element(CProj, E).Length
         M = L - ((a + b) / 2)
         W = (LoadData(CProj, I).StartLoad) * (b - a)
         p = b - a
         FEML = FEML - (W * ((a + b) / 2))
        Case 3: 'Triangle Load
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         W = (LoadData(CProj, I).StartLoad) * (b - a)
         W1 = (1 / 2) * (LoadData(CProj, I).EndLoad - LoadData(CProj, I).S
tartLoad) * (b - a)
         FEML = FEML - (W * ((a + b) / 2))
         FEML = FEML - (W1 * (((2 * b) + a) / 3))
        Case 4: 'Moment at Point
         a = LoadData(CProj, I).Start
         b = Element(CProj, E).Length - LoadData(CProj, I).End
         L = Element(CProj, E).Length
         M = LoadData(CProj, I).StartLoad
         FEML = FEML + -M
        Case 5: 'Function Load
        End Select
       Else
        Select Case LoadData(CProj, I).Type
        Case 1: 'Point Load
         W = LoadData(CProj, I).StartLoad
         a = LoadData(CProj, I).Start
         b = Element(CProj, E).Length - LoadData(CProj, I).Start
         L = Element(CProj, E).Length
         FEML = FEML + (((W * a * (b ^ 2)) / (L ^ 2)) * (-1))
         'FEMR = FEMR + ((W * (a ^ 2) * b) / (L ^ 2))
        Case 2: 'Uniform Load
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         L = Element(CProj, E).Length
          M = L - ((a + b) / 2)
         W = (LoadData(CProj, I).StartLoad) * (b - a)
         p = b - a
         FEML = FEML - ((((W / (24 * L)) * ((((24 * (M ^ 3)) / L) - ((6 *
b * (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (4 * (p ^ 2)) - (24 * (M ^ 2))))
 * (-1))))
         'FEMR = FEMR + (W / (24 * L)) * ((((24 * (m ^ 3)) / L) - ((6 * b
* (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (2 * (p ^ 2)) - (48 * (m ^ 2)) + (
24 * m * L)))
        Case 3: 'Triangle Load
         a = LoadData(CProj, I).Start
         b = LoadData(CProj, I).End
         L = Element(CProj, E).Length
         M = L - ((a + b) / 2)
         N = L - ((a + (2 * b)) / 3)
         W = (LoadData(CProj, I).StartLoad) * (b - a)
         W1 = (1 / 2) * (LoadData(CProj, I).EndLoad - LoadData(CProj, I).S
tartLoad) * (b - a)
         p = b - a
         FEML = FEML + (((W / (24 * L)) * ((((24 * (M ^ 3)) / L) - ((6 * b
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 8

```
 * (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (4 * (p ^ 2)) - (24 * (M ^ 2)))))
 * (-1))
          FEML = FEML + (((W1 / L) * (((N ^ 3) / L) + ((p ^ 2) / 9) + ((51
* (p ^ 3)) / (810 * L)) - (((1 / 6) * (p ^ 2) * b) / L) - (N ^ 2)))) * (-1
)
          'FEMR = FEMR + (W / (24 * L)) * ((((24 * (m ^ 3)) / L) - ((6 * b
* (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (2 * (p ^ 2)) - (48 * (m ^ 2)) + (
24 * m * L))) + ((W1 / L) * (((n ^ 3) / L) - ((p ^ 2) / 18) + ((51 * (p ^
3)) / (810 * L)) - (((1 / 6) * (p ^ 2) * b) / L) - (2 * (n ^ 2)) + (n * L)
))
          Case 4: 'Moment at Point
          a = LoadData(CProj, I).Start
          b = Element(CProj, E).Length - LoadData(CProj, I).End
          L = Element(CProj, E).Length
          M = LoadData(CProj, I).StartLoad
          FEML = FEML + (((M * b) * ((3 * a) - L)) / (L ^ 2))
          'FEMR = FEMR + ((m * a) * ((3 * b) - L)) / (L ^ 2)
         Case 5: 'Function Load
         End Select
        End If
       End If
      Next I
      Cal_BFEM = FEML
     ElseIf Not (LeftOrRight) Then
      For I = 1 To LoadNum(CProj)
       If LoadData(CProj, I).ElementRef = E Then
        If Node(CProj, Element(CProj, E).End).SupportType = 3 Then
         Select Case LoadData(CProj, I).Type
         Case 1: 'Point Load
          W = LoadData(CProj, I).StartLoad
          a = Element(CProj, E).Length - LoadData(CProj, I).Start
          L = Element(CProj, E).Length
          FEML = FEML + (W * a)
         Case 2: 'Uniform Load
          a = Element(CProj, E).Length - LoadData(CProj, I).End
          b = Element(CProj, E).Length - LoadData(CProj, I).Start
          L = Element(CProj, E).Length
          M = L - ((a + b) / 2)
          W = (LoadData(CProj, I).StartLoad) * (b - a)
          FEMR = FEMR + (W * ((a + b) / 2))
         Case 3: 'Triangle Load
          a = Element(CProj, E).Length - LoadData(CProj, I).End
          b = Element(CProj, E).Length - LoadData(CProj, I).Start
          L = Element(CProj, E).Length
          W = (LoadData(CProj, I).EndLoad) * (b - a)
          W1 = (1 / 2) * (LoadData(CProj, I).StartLoad - LoadData(CProj, I)
.EndLoad) * (b - a)
          FEMR = FEMR - (W * ((a + b) / 2))
          FEMR = FEMR - (W1 * (((2 * b) + a) / 3))
         Case 4: 'Moment at Point
          a = Element(CProj, E).Length - LoadData(CProj, I).Start
          M = LoadData(CProj, I).StartLoad
          FEMR = FEMR - M
         Case 5: 'Function Load
         End Select
        Else
         Select Case LoadData(CProj, I).Type
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 9

```
          Case 1: 'Point Load
           W = LoadData(CProj, I).StartLoad
           a = LoadData(CProj, I).Start
           b = Element(CProj, E).Length - LoadData(CProj, I).Start
           L = Element(CProj, E).Length
           'FEML = FEML + ((W * a * (b ^ 2)) / (L ^ 2)) * (-1)
           FEMR = FEMR + ((W * (a ^ 2) * b) / (L ^ 2))
          Case 2: 'Uniform Load
           a = LoadData(CProj, I).Start
           b = LoadData(CProj, I).End
           L = Element(CProj, E).Length
           M = L - ((a + b) / 2)
           W = (LoadData(CProj, I).StartLoad) * (b - a)
           p = b - a
           'FEML = FEML - (W / (24 * L)) * ((((24 * (m ^ 3)) / L) - ((6 * b
* (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (4 * (p ^ 2)) - (24 * (m ^ 2))))
           FEMR = FEMR + ((W / (24 * L)) * ((((24 * (M ^ 3)) / L) - ((6 * b
* (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (2 * (p ^ 2)) - (48 * (M ^ 2)) + (
24 * M * L))))
          Case 3: 'Triangle Load
           a = LoadData(CProj, I).Start
           b = LoadData(CProj, I).End
           L = Element(CProj, E).Length
           M = L - ((a + b) / 2)
           N = L - ((a + (2 * b)) / 3)
           W = (LoadData(CProj, I).StartLoad) * (b - a)
           W1 = (1 / 2) * (LoadData(CProj, I).EndLoad - LoadData(CProj, I).S
tartLoad) * (b - a)
           p = b - a
           'FEML = FEML - (W / (24 * L)) * ((((24 * (m ^ 3)) / L) - ((6 * b
* (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (4 * (p ^ 2)) - (24 * (m ^ 2)))) +
 ((W1 / L) * (((n ^ 3) / L) + ((p ^ 2) / 9) + ((51 * (p ^ 3)) / (810 * L))
 - (((1 / 6) * (p ^ 2) * b) / L) - (n ^ 2)))
           FEMR = FEMR + (W / (24 * L)) * ((((24 * (M ^ 3)) / L) - ((6 * b *
 (p ^ 2)) / L) + ((3 * (p ^ 3)) / L) + (2 * (p ^ 2)) - (48 * (M ^ 2)) + (2
4 * M * L)))
           FEMR = FEMR + ((W1 / L) * (((N ^ 3) / L) - ((p ^ 2) / 18) + ((51
* (p ^ 3)) / (810 * L)) - (((1 / 6) * (p ^ 2) * b) / L) - (2 * (N ^ 2)) +
(N * L)))
          Case 4: 'Moment at Point
           a = LoadData(CProj, I).Start
           b = Element(CProj, E).Length - LoadData(CProj, I).End
           L = Element(CProj, E).Length
           M = LoadData(CProj, I).StartLoad
           'FEML = FEML + ((m * b) * ((3 * a) - L)) / (L ^ 2)
           FEMR = FEMR + (((M * a) * ((3 * b) - L)) / (L ^ 2))
          Case 5: 'Function Load
          End Select
         End If
       End If
      Next I
      Cal_BFEM = FEMR
     End If
End Function

Function Cal_BK (E As Integer) As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 10

```
    Dim Ec As Single
    Ec = GeneralData(CProj).SEc
    Cal_BK = ((4 * Ec * Cal_BSecondInertia(E)) / Element(CProj, E).Length)
    If (Node(CProj, Element(CProj, E).Start).SupportType = 3) Or (Node(CPr
oj, Element(CProj, E).End).SupportType = 3) Then
     Cal_BK = 0
    End If
End Function

Function Cal_BMoment (E As Integer, x As Single, ML As Single, MR As Singl
e) As Single
    Dim M As Single
    Dim p As Single
    Dim W As Single
    Dim W1 As Single
    Dim W2 As Single
    Dim a As Single
    Dim b As Single
    Dim L As Single
    Dim Mx As Single
    Dim Wx As Single
    'RL = 0
    'RR = 0
    Mx = 0
    For I = 1 To LoadNum(CProj)
     If LoadData(CProj, I).ElementRef = E Then
      Select Case LoadData(CProj, I).Type
      Case 1:
       a = LoadData(CProj, I).Start
       p = LoadData(CProj, I).StartLoad
       L = Element(CProj, E).Length
       If x <= a Then
        Mx = Mx + (((p * (L - a)) / L) * x)
       End If
       If x > a Then
        Mx = Mx + (((p * a) / L) * (L - x))
       End If
      Case 2:
       a = LoadData(CProj, I).Start
       b = LoadData(CProj, I).End
       W = LoadData(CProj, I).StartLoad
       L = Element(CProj, E).Length
       If x <= a Then
        Mx = Mx + (((W * (b - a) * (L - b + ((b - a) / 2))) / L) * x)
       End If
       If (x > a) And (x < b) Then
        Mx = Mx + ((((W * (b - a) * (L - b + ((b - a) / 2))) / L) * x) - (
W * (x - a) * (1 / 2) * (x - a)))
       End If
       If x >= b Then
        Mx = Mx + (((W * (b - a) * (a + ((b - a) / 2))) / L) * (L - x))
       End If
      Case 3:
       a = LoadData(CProj, I).Start
       b = LoadData(CProj, I).End
       W1 = LoadData(CProj, I).StartLoad
       W2 = LoadData(CProj, I).EndLoad
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 11

```
        L = Element(CProj, E).Length
        Wx = (W1 + (((W2 - W1) * (x - a)) / (b - a)))
        If x <= a Then
         Mx = Mx + (((((1 / 2) * (W1 + W2) * (b - a)) * (L - b + (((W1 + ((
1 / 3) * W2)) / (W1 + W2))) * (b - a))) / L) * x)
        ElseIf (x > a) And (x < b) Then
         Mx = Mx + (((((1 / 2) * (W1 + W2) * (b - a)) * (L - b + (((W1 + ((
1 / 3) * W2)) / (W1 + W2))) * (b - a))) / L) * x)
         Mx = Mx - ((1 / 2) * (W1 + Wx) * (b - a) * (((W1 + ((1 / 3) * W2))
 / (W1 + W2))) * (x - a))
        ElseIf (x > b) Then
         Mx = Mx + (((((1 / 2) * (W1 + W2) * (b - a)) * (a + (((W2 + ((1 /
3) * W1)) / (W1 + W2))) * (b - a))) / L) * (L - x))
        End If
       Case 4:
        a = LoadData(CProj, I).Start
        M = LoadData(CProj, I).StartLoad
        L = Element(CProj, E).Length
        If x <= a Then
         Mx = Mx - ((M / L) * x)
        ElseIf x > a Then
         Mx = Mx + ((M / L) * (L - x))
        End If
       Case 5:
       End Select
      End If
     Next I
     L = Element(CProj, E).Length
     Mx = Mx - (((ML + MR) / L) * x) + ML
     Cal_BMoment = Mx
End Function

Function Cal_BSecondInertia (E As Integer) As Single
     Dim b As Single
     Dim d As Single
     Select Case Section(CProj, E).Type
     Case 1:
      b = Section(CProj, E).b
      d = Section(CProj, E).d
      Cal_BSecondInertia = (1 / 12) * b * (d ^ 3)
     Case 2:
     Case 3:
     Case 4:
     End Select
End Function


Function Cal_BShear (E As Integer, x As Single) As Single
     Dim M As Single
     Dim p As Single
     Dim W As Single
     Dim W1 As Single
     Dim W2 As Single
     Dim Wx As Single
     Dim a As Single
     Dim b As Single
     Dim L As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 12

```
    Dim V As Single
    V = 0
    For I = 1 To LoadNum(CProj)
     If LoadData(CProj, I).ElementRef = E Then
      Select Case LoadData(CProj, I).Type
      Case 1:
       a = LoadData(CProj, I).Start
       p = LoadData(CProj, I).StartLoad
       L = Element(CProj, E).Length
       If x <= a Then
        V = V + ((p * (L - a)) / L)
       End If
       If x > a Then
        V = V - ((p * a) / L)
       End If
      Case 2:
       a = LoadData(CProj, I).Start
       b = LoadData(CProj, I).End
       W = LoadData(CProj, I).StartLoad
       L = Element(CProj, E).Length
       If x <= a Then
        V = V + (W * (b - a) * (L - b + ((b - a) / 2)) / L)
       End If
       If (x > a) And (x < b) Then
        V = V + ((W * (b - a) * (L - b + ((b - a) / 2)) / L) - (W * (x - a
)))
       End If
       If x >= b Then
       V = V - ((W * (b - a) * (a + ((b - a) / 2))) / L)
       End If
      Case 3:
       a = LoadData(CProj, I).Start
       b = LoadData(CProj, I).End
       W1 = LoadData(CProj, I).StartLoad
       W2 = LoadData(CProj, I).EndLoad
       Wx = W1 + (((W2 - W1) * (x - a)) / (b - a))
       L = Element(CProj, E).Length
       If x <= a Then
        V = V + ((1 / 2) * (W1 + W2) * (b - a) * (L - b + ((b - a) * (W1 +
((1 / 3) * W2)) / (W1 + W2))) / L)
       End If
       If (x > a) And (x < b) Then
        V = V + ((((1 / 2) * (W1 + W2) * (b - a) * (L - b + ((b - a) * ((W
1 + ((1 / 3) * W2)) / (W1 + W2))))) / L) - ((1 / 2) * (Wx + W1) * (x - a))
)
       End If
       If (x >= b) Then
        V = V + (((1 / 2) * (W1 + W2) * (b - a) * (L - b + (W1 + ((1 / 3)
* W2)) / (W1 + W2)) / L) - ((1 / 2) * (b - a) * (W1 + W2)))
       End If
      Case 4:
       a = LoadData(CProj, I).Start
       M = LoadData(CProj, I).StartLoad
       L = Element(CProj, E).Length
       If x <= a Then
        V = V - (M / L)
       End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTE.BAS - 13

```
        If x > a Then
         V = V - (M / L)
        End If
       Case 5:
       End Select
      End If
     Next I
     Cal_BShear = V - ((ML(CProj, E) + MR(CProj, E)) / Element(CProj, E).Le
ngth)
End Function

Function Fc_ToEc (fc_) As Single
    Fc_ToEc = 4730 * Sqr(fc_)
End Function
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CRTDATA.BAS - 1

```
Sub Action_Icon (Index As Integer, Enabled As Integer)
    Select Case Enabled
    Case True:
     Select Case Index
     Case 0:
      frmMain.imgTool1(Index).Enabled = True
     Case 1:
      frmMain.imgTool1(Index).Enabled = True
     Case 2:
      frmMain.imgTool1(Index).Enabled = True
     Case 3:
      frmMain.imgTool1(Index).Enabled = True
     Case 4:
      frmMain.imgTool1(Index).Enabled = True
     Case 5:
      frmMain.imgTool1(Index).Enabled = True
     Case 6:
      frmMain.imgTool1(Index).Enabled = True
     Case 7:
      frmMain.imgTool1(Index).Enabled = True
     End Select
    Case False:
     Select Case Index
     Case 0:
      frmMain.imgTool1(Index).Enabled = False
     Case 1:
      frmMain.imgTool1(Index).Enabled = False
     Case 2:
      frmMain.imgTool1(Index).Enabled = False
     Case 3:
      frmMain.imgTool1(Index).Enabled = False
     Case 4:
      frmMain.imgTool1(Index).Enabled = False
     Case 5:
      frmMain.imgTool1(Index).Enabled = False
     Case 6:
      frmMain.imgTool1(Index).Enabled = False
     Case 7:
      frmMain.imgTool1(Index).Enabled = False
     End Select
    End Select
End Sub

Sub Do_Tool2 (ColNum As Integer, RowNum As Integer)
    Const Span = 5
    Const TwipsPerMeter = 100
    Const I_X = 1000
    Const I_Y = 1000
    For I = 1 To ColNum
     For J = 1 To RowNum
      Column(CProj, (I - 1) * J + J).Co_X = Span * (I - 1) * TwipsPerMeter
      Column(CProj, (I - 1) * J + J).Co_Y = Span * (J - 1) * TwipsPerMeter
      'frmDataInfo(CProj).Line (I_X + 0 + Column(CProj, (I - 1) * J + J).C
o_X, I_Y + 0 + Column(CProj, (I - 1) * J + J).Co_Y)-(I_X + 50 + Column(CPr
oj, (I - 1) * J + J).Co_X, I_Y + 50 + Column(CProj, (I - 1) * J + J).Co_Y)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CRTDATA.BAS - 2

```
, ColumnColor, BF
     Next J
    Next I
End Sub

Sub Draw_SNode (ArrX As Integer, ArrY As Integer, Node As Integer)
    Const I_X = 1000
    Const I_Y = 1000
    For I = 0 To ArrY - 1
     For J = 0 To ArrX - 1
      'frmData(CProj).Line (I_X + 0 + Column(CProj, Current + I * J + J).C
o_X, I_Y + 0 + Column(CProj, I * J + J).Co_Y)-(I_X + 50 + Column(CProj, I
* J + J).Co_X, I_Y + 50 + Column(CProj, I * J + J).Co_Y), ColumnColor, BF
     Next J
    Next I
    Exit Sub
End Sub

Sub Initial_Menu (Index As Integer)
    Select Case Index
    Case 1: 'initial in enter program
     'file
     frmMain.mnuFile_New.Enabled = True
     frmMain.mnuFile_Open.Enabled = True
     frmMain.mnuFile_Save.Enabled = False
     frmMain.mnuFile_SaveAs.Enabled = False
     frmMain.mnuFile_Print_Prestress_Profile.Enabled = False
     frmMain.mnuFile_Print_Prestress_LProfile.Enabled = False
     frmMain.mnuFile_Print_Prestress_LCompute.Enabled = False
     frmMain.mnuFile_Print_Flat_LCompute.Enabled = False
     frmMain.mnuFile_Print_Geometric.Enabled = False
     frmMain.mnuFile_Print_Shear.Enabled = False
     frmMain.mnuFile_Print_Moment.Enabled = False
     'input
     frmMain.mnuInput_Node.Enabled = False
     frmMain.mnuInput_Element.Enabled = False
     frmMain.mnuInput_Load.Enabled = False
     frmMain.mnuInput_Material.Enabled = False
     frmMain.mnuInput_Information = False
     'Design
     frmMain.mnuDesign_Prestress.Enabled = False
     frmMain.mnuDesign_Flat.Enabled = False
    'Result
     frmMain.mnuResult_Prestress_Profile.Enabled = False
     frmMain.mnuResult_Prestress_LProfile.Enabled = False
     frmMain.mnuResult_Prestress_LCompute.Enabled = False
     frmMain.mnuResult_Flat_LCompute.Enabled = False
     frmMain.mnuResult_Geometric.Enabled = False
     frmMain.mnuResult_Shear.Enabled = False
     frmMain.mnuResult_Moment.Enabled = False
     'Option
     frmMain.mnuOption_SaveSetting.Enabled = True
     'Window
     frmMain.mnuWindow_ArrangeIcon.Enabled = True
     frmMain.mnuWindow_Close.Enabled = False
     'Help
     frmMain.mnuHelp_System.Enabled = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    frmMain.mnuHelp_About.Enabled = True
    ''menu icon
    Action_Icon 0, True
    Action_Icon 1, True
    Action_Icon 2, False
    Action_Icon 3, False
    Action_Icon 4, False
    Action_Icon 5, False
    Action_Icon 6, False
    Action_Icon 7, False
  Case 2: 'Open or new file
    'file
    frmMain.mnuFile_New.Enabled = True
    frmMain.mnuFile_Open.Enabled = True
    frmMain.mnuFile_Save.Enabled = True
    frmMain.mnuFile_SaveAs.Enabled = True
    frmMain.mnuFile_Print_Prestress_Profile.Enabled = False
    frmMain.mnuFile_Print_Prestress_LProfile.Enabled = False
    frmMain.mnuFile_Print_Prestress_LCompute.Enabled = False
    frmMain.mnuFile_Print_Flat_LCompute.Enabled = False
    frmMain.mnuFile_Print_Geometric.Enabled = False
    frmMain.mnuFile_Print_Shear.Enabled = False
    frmMain.mnuFile_Print_Moment.Enabled = False
    'input
    frmMain.mnuInput_Node.Enabled = True
    frmMain.mnuInput_Element.Enabled = True
    frmMain.mnuInput_Load.Enabled = True
    frmMain.mnuInput_Material.Enabled = True
    frmMain.mnuInput_Information = True
    'Design
    frmMain.mnuDesign_Prestress.Enabled = True
    frmMain.mnuDesign_Flat.Enabled = True
  'Result
    frmMain.mnuResult_Prestress_Profile.Enabled = False
    frmMain.mnuResult_Prestress_LProfile.Enabled = False
    frmMain.mnuResult_Prestress_LCompute.Enabled = False
    frmMain.mnuResult_Flat_LCompute.Enabled = False
    frmMain.mnuResult_Geometric.Enabled = False
    frmMain.mnuResult_Shear.Enabled = False
    frmMain.mnuResult_Moment.Enabled = False
    'Option
    frmMain.mnuOption_SaveSetting.Enabled = True
    'Window
    frmMain.mnuWindow_ArrangeIcon.Enabled = True
    frmMain.mnuWindow_Close.Enabled = True
    'Help
    frmMain.mnuHelp_System.Enabled = True
    frmMain.mnuHelp_About.Enabled = True
    'menu icon
    Action_Icon 0, True
    Action_Icon 1, True
    Action_Icon 2, True
    Action_Icon 3, True
    Action_Icon 4, True
    Action_Icon 5, True
    Action_Icon 6, False
    Action_Icon 7, False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CRTDATA.BAS - 4

```
    Case 3: 'Prestress design
     'file
     frmMain.mnuFile_New.Enabled = True
     frmMain.mnuFile_Open.Enabled = True
     frmMain.mnuFile_Save.Enabled = True
     frmMain.mnuFile_SaveAs.Enabled = True
     frmMain.mnuFile_Print_Prestress_Profile.Enabled = True
     frmMain.mnuFile_Print_Prestress_LProfile.Enabled = True
     frmMain.mnuFile_Print_Prestress_LCompute.Enabled = True
     frmMain.mnuFile_Print_Flat_LCompute.Enabled = False
     frmMain.mnuFile_Print_Geometric.Enabled = True
     frmMain.mnuFile_Print_Shear.Enabled = True
     frmMain.mnuFile_Print_Moment.Enabled = True
     'input
     frmMain.mnuInput_Node.Enabled = True
     frmMain.mnuInput_Element.Enabled = True
     frmMain.mnuInput_Load.Enabled = True
     frmMain.mnuInput_Material.Enabled = True
     frmMain.mnuInput_Information = True
     'Design
     frmMain.mnuDesign_Prestress.Enabled = True
     frmMain.mnuDesign_Flat.Enabled = True
    'Result
     frmMain.mnuResult_Prestress_Profile.Enabled = True
     frmMain.mnuResult_Prestress_LProfile.Enabled = True
     frmMain.mnuResult_Prestress_LCompute.Enabled = True
     frmMain.mnuResult_Flat_LCompute.Enabled = False
     frmMain.mnuResult_Geometric.Enabled = True
     frmMain.mnuResult_Shear.Enabled = True
     frmMain.mnuResult_Moment.Enabled = True
     'Option
     frmMain.mnuOption_SaveSetting.Enabled = True
     'Window
     frmMain.mnuWindow_ArrangeIcon.Enabled = True
     frmMain.mnuWindow_Close.Enabled = True
     'Help
     frmMain.mnuHelp_System.Enabled = True
     frmMain.mnuHelp_About.Enabled = True
     'menu icon
     Action_Icon 0, True
     Action_Icon 1, True
     Action_Icon 2, True
     Action_Icon 3, True
     Action_Icon 4, True
     Action_Icon 5, True
     Action_Icon 6, True
     Action_Icon 7, True
    Case 4: 'Flat design
     'file
     frmMain.mnuFile_New.Enabled = True
     frmMain.mnuFile_Open.Enabled = True
     frmMain.mnuFile_Save.Enabled = True
     frmMain.mnuFile_SaveAs.Enabled = True
     frmMain.mnuFile_Print_Prestress_Profile.Enabled = False
     frmMain.mnuFile_Print_Prestress_LProfile.Enabled = False
     frmMain.mnuFile_Print_Prestress_LCompute.Enabled = False
     frmMain.mnuFile_Print_Flat_LCompute.Enabled = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CRTDATA.BAS - 5

```
        frmMain.mnuFile_Print_Geometric.Enabled = True
        frmMain.mnuFile_Print_Shear.Enabled = True
        frmMain.mnuFile_Print_Moment.Enabled = True
        'input
        frmMain.mnuInput_Node.Enabled = True
        frmMain.mnuInput_Element.Enabled = True
        frmMain.mnuInput_Load.Enabled = True
        frmMain.mnuInput_Material.Enabled = True
        frmMain.mnuInput_Information = True
        'Design
        frmMain.mnuDesign_Prestress.Enabled = True
        frmMain.mnuDesign_Flat.Enabled = True
      'Result
        frmMain.mnuResult_Prestress_Profile.Enabled = False
        frmMain.mnuResult_Prestress_LProfile.Enabled = False
        frmMain.mnuResult_Prestress_LCompute.Enabled = False
        frmMain.mnuResult_Flat_LCompute.Enabled = True
        frmMain.mnuResult_Geometric.Enabled = True
        frmMain.mnuResult_Shear.Enabled = True
        frmMain.mnuResult_Moment.Enabled = True
        'Option
        frmMain.mnuOption_SaveSetting.Enabled = True
        'Window
        frmMain.mnuWindow_ArrangeIcon.Enabled = True
        frmMain.mnuWindow_Close.Enabled = True
        'Help
        frmMain.mnuHelp_System.Enabled = True
        frmMain.mnuHelp_About.Enabled = True
        'menu icon
        Action_Icon 0, True
        Action_Icon 1, True
        Action_Icon 2, True
        Action_Icon 3, True
        Action_Icon 4, True
        Action_Icon 5, True
        Action_Icon 6, True
        Action_Icon 7, True
    End Select
End Sub

Sub Show_DataScreen ()
    Dim TPMX As Integer
    Dim TPMY As Integer
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer

    frmData(CProj).Cls
    Col = 50
    InitX = 1500
    InitY = 1000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CRTDATA.BAS - 6

```
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I
    TPMX = (frmData(CProj).Width - 3000) \ (XMax - XMin)
    TPMY = (frmData(CProj).Height - 2000) \ (YMax - YMin)
    If TPMX < TPMY Then
     TPMY = TPMX
    Else
     TPMX = TPMY
    End If
    For I = 1 To SNodeNum(CProj)
     frmData(CProj).Line (((Column(CProj, I).Co_X - MinX) * TPMX) - Col +
InitX, ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY)-(((Column(CP
roj, I).Co_X - MinX) * TPMX) + Col + InitX, ((Column(CProj, I).Co_Y - MinY
) * TPMY) + Col + InitY), ColumnColor, BF
     frmData(CProj).Print "[" + Format$(Column(CProj, I).Co_X, "#.00") + "
," + Format$(Column(CProj, I).Co_Y, "#.00") + "]"
    Next I
End Sub

Sub Show_DataScreenE ()
    Dim TPMX As Integer
    Dim TPMY As Integer
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer

    ElementCOlor = 30
    ColumnColor = 34

    frmOutGraph(CProj).Cls
    Col = 50
    InitX = 1500
    InitY = 1000
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CRTDATA.BAS - 7

```
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I
    TPMX = (frmOutGraph(CProj).Width - 3000) \ (XMax - XMin)
    TPMY = (frmOutGraph(CProj).Height - 2000) \ (YMax - YMin)
    If TPMX < TPMY Then
     TPMY = TPMX
    Else
     TPMX = TPMY
    End If
    For I = 1 To SNodeNum(CProj)
     frmOutGraph(CProj).Line (((Column(CProj, I).Co_X - MinX) * TPMX) - Co
l + InitX, ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY)-(((Colum
n(CProj, I).Co_X - MinX) * TPMX) + Col + InitX, ((Column(CProj, I).Co_Y -
MinY) * TPMY) + Col + InitY), ColumnColor, BF
     frmOutGraph(CProj).Print Format$(I, "@@@") + "[" + Format$(Column(CPr
oj, I).Co_X, "#.00") + "," + Format$(Column(CProj, I).Co_Y, "#.00") + "]"
    Next I
    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementCOlor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementCOlor)
     Next J
    Next I
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 1

```
Function Cal_Fb0 (Column As ColumnType, Index As Integer) As Single
    Dim Cal_Fb0t As Single
    Dim c1 As Single
    Dim c2 As Single
    Dim ed As Single
    Dim ed1 As Single
    Dim ed2 As Single
    ed1 = d(CProj)
    ed2 = Column.DropThick + ed1
    ed = ed1
    If Index = 0 Then 'none
     c1 = Column.c1
     c2 = Column.c2
     ed = ed1
    ElseIf Index = 1 Then 'Drop panel
     c1 = Column.DropX
     c2 = Column.DropY
     ed = ed1
    ElseIf Index = 2 Then 'Drop Capital
     c1 = Column.CapitalX
     c2 = Column.CapitalY
     ed = ed2
    End If
    'Interior
    If (Column.X_Flag + Column.Y_Flag) = 0 Then
     Cal_Fb0t = (c1 + ed + c2 + ed) * 2
    'Corner
    ElseIf (Column.X_Flag <> 0) And (Column.Y_Flag <> 0) Then
     Cal_Fb0t = (c1 + ed + c2)
    'Edge Y
    ElseIf (Column.X_Flag = 0) And ((Column.Y_Flag = 1) Or (Column.Y_Flag
= 2)) Then
     Cal_Fb0t = (c1 + ed + (2 * c2) + ed)
    'Edge X
    ElseIf (Column.Y_Flag = 0) And ((Column.X_Flag = 1) Or (Column.X_Flag
= 2)) Then
     Cal_Fb0t = ((c1 * 2) + ed + c2 + ed)
    End If
    Cal_Fb0 = Cal_Fb0t
End Function

Function Cal_FColumnK (CN As Integer, XY As Integer, h As Single) As Singl
e  'h is thickness
    Dim Kec As Single
    Dim Kt As Single
    Dim Kcc As Single
    Dim Kc As Single
    Dim a As Single
    Dim b As Single
    Dim R As Single
    Dim s As Single
    Dim c As Single
    Dim d As Single
    Dim Y_ As Single
    Dim X2 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 2

```
    Dim CC1 As Single   'length of column as x axis
    Dim CC2 As Single   'length of column as y axis
    Dim CDx As Single   'Drop x length
    Dim CDy As Single   'Drop y Length
    Dim CDt As Single   'Drop Thick
    Dim CCx As Single   'Capital x Length
    Dim CCy As Single   'Capital y Length
    Dim CCL As Single 'Column length
    Dim Cs As Single
    Dim Cd As Single
    Dim C3 As Single
    Dim C4 As Single
    Dim C5 As Single
    Dim C6 As Single
    Dim C7 As Single
    Dim FA As Single
    Dim FB As Single
    Dim FR As Single
    Dim FC As Single
    Dim FD As Single
    Dim CR As Single
    ReDim SColRow(MaxFileNum) As Integer
    lc = Column(CProj, CN).Length
    If XY Then
     SColRow(CProj) = ElementXNum(CProj)
    Else
     SColRow(CProj) = ElementYNum(CProj)
    End If
    For I = 1 To SColRow(CProj) 'find transverse length
     If (SElement(CProj, XY * (-1), I).Start = Element(CProj, E).CStart) O
r (SElement(CProj, XY * (-1), E).End = Element(CProj, E).CStart) Then
      L2 = (SElement(CProj, XY * (-1), I).Length / 2)
     End If
    Next I
    If XY Then 'Along x axis
     CC1 = Column(CProj, CN).c1
     CC2 = Column(CProj, CN).c2
     CDx = Column(CProj, CN).DropX
     CDy = Column(CProj, CN).DropY
     CDt = Column(CProj, CN).DropThick
     CCx = Column(CProj, CN).CapitalX
     CCy = Column(CProj, CN).CapitalY
     CCL = Column(CProj, CN).Length
     a = ((h + CDt) / 2) / lc
     b = 0
     R = 0
     s = 0
     c = 0
     d = (1 - a)
     FA = 0
     FB = 1
     FR = 1
     FC = 1
     FD = 0
     C3 = (((-FB) * a * a * (3 - (2 * a)) * (1 - FA)) - ((b * b) * (3 - (2
 * b)) * (1 - FB)) + ((R ^ 2) * (3 - (2 * R)) * (1 - FR)) - ((s ^ 2) * (3
 - (2 * s)) * (1 - FR)) + ((c ^ 2) * (3 - (2 * c)) * (1 - FC)) + ((FC * d *
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 3

```
 d) * (3 - (2 * d)) * (1 - FD)) + (FC * FD)) / 6
     C4 = (((-FB * (a ^ 3)) * (1 - FA)) - ((b ^ 3) * (1 - FB)) + ((R ^ 3)
* (1 - FR)) - ((s ^ 3) * (1 - FR)) + ((c ^ 3) * (1 - FC)) + ((FC * (d ^ 3)
) * (1 - FD)) + (FC * FD)) / 3
     C5 = ((((-FB) * (a ^ 3)) * (4 - (3 * a)) * (1 - FA)) - ((b ^ 3) * (4
- (3 * b)) * (1 - FB)) + ((R ^ 3) * (4 - (3 * R)) * (1 - FR)) - ((s ^ 3) *
 (4 - (3 * s)) * (1 - FR)) + ((c ^ 3) * (4 - (3 * c)) * (1 - FC)) + ((FC *
 (d ^ 3)) * (4 - (3 * d)) * (1 - FD)) + (FC * FD)) / 24
     C6 = ((((-FB) * a) * (3 - (3 * a) + (a ^ 2)) * (1 - FA)) - (b * (3 -
(3 * b) + (b ^ 2)) * (1 - FB)) + (R * (3 - (3 * R) + (R ^ 2)) * (1 - FR))
- (s * (3 - (3 * s) + (s ^ 2)) * (1 - FR)) + (c * (3 - (3 * c) + (c ^ 2))
* (1 - FC)) + ((FC * d) * (3 - (3 * d) + (d ^ 2)) * (1 - FD)) + (FC * FD))
 / 3
     C7 = (((-FB) * (a ^ 2) * (6 - (8 * a) + (3 * (a ^ 2))) * (1 - FA)) -
((b ^ 2) * (6 - (8 * b) + (3 * (b ^ 2))) * (1 - FB)) + ((R ^ 2) * (6 - (8
* R) + (3 * (R ^ 2))) * (1 - FR)) - ((s ^ 2) * (6 - (8 * s) + (3 * (s ^ 2)
)) * (1 - FR)) + ((c ^ 2) * (6 - (8 * c) + (3 * (c ^ 2))) * (1 - FC)) + ((
FC * (d ^ 2)) * (6 - (8 * d) + (3 * (d ^ 2))) * (1 - FD)) + (FC * FD)) / 2
4
     CR = (C6 * C4) - (C3 * C3)
     Kcc = C4 / CR
     Kc = ((Kcc * GeneralData(CProj).CEc) * ((CC2 * (CC1 ^ 3)) / 12)) / lc
     '*************************************************************
     'to be continued tomorrow
     If frmMain.optSystem(1).Value Then
      Cs = ((1 - (.63 * (h / CC1))) * (((h ^ 3) * CC1) / 3))
      Cd = (1 - (.63 * ((h + CDt) / CC1))) * ((((h + CDt) ^ 3) * CC1) / 3)
      X2 = (L2 - CDy) / 2
      Kt = (GeneralData(CProj).CEc / (((L2 * (1 - ((CC2 / L2) ^ 5))) / (20
 * Cd))) + (((L2 * ((1 - (X2 / L2)) ^ 5)) / 20) * ((1 / Cs) - (1 / Cd))))
     Else
      Kt = ((9 * GeneralData(CProj).CEc * ((1 - (.63 * (h / CC1))) * (((h
^ 3) * CC1) / 3))) / (L2 * ((1 - (CC2 / L2)) ^ 3)))
     End If
    End If
    If Not (XY) Then 'Along Y axis
     CC1 = Column(CProj, CN).c1
     CC2 = Column(CProj, CN).c2
     CDx = Column(CProj, CN).DropX
     CDy = Column(CProj, CN).DropY
     CDt = Column(CProj, CN).DropThick
     CCx = Column(CProj, CN).CapitalX
     CCy = Column(CProj, CN).CapitalY
     CCL = Column(CProj, CN).Length
     a = ((h + CDt) / 2) / lc
     b = 0
     R = 0
     s = 0
     c = 0
     d = (1 - a)
     FA = 0
     FB = 1
     FR = 1
     FC = 1
     FD = 0
     C3 = (((-FB) * a * a * (3 - (2 * a)) * (1 - FA)) - ((b * b) * (3 - (2
 * b)) * (1 - FB)) + ((R ^ 2) * (3 - (2 * R)) * (1 - FR)) - ((s ^ 2) * (3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 4

```
- (2 * s)) * (1 - FR)) + ((c ^ 2) * (3 - (2 * c)) * (1 - FC)) + ((FC * d *
 d) * (3 - (2 * d)) * (1 - FD)) + (FC * FD)) / 6
     C4 = (((-FB * (a ^ 3)) * (1 - FA)) - ((b ^ 3) * (1 - FB)) + ((R ^ 3)
* (1 - FR)) - ((s ^ 3) * (1 - FR)) + ((c ^ 3) * (1 - FC)) + ((FC * (d ^ 3)
) * (1 - FD)) + (FC * FD)) / 3
     C5 = ((((-FB) * (a ^ 3)) * (4 - (3 * a)) * (1 - FA)) - ((b ^ 3) * (4
- (3 * b)) * (1 - FB)) + ((R ^ 3) * (4 - (3 * R)) * (1 - FR)) - ((s ^ 3) *
 (4 - (3 * s)) * (1 - FR)) + ((c ^ 3) * (4 - (3 * c)) * (1 - FC)) + ((FC *
 (d ^ 3)) * (4 - (3 * d)) * (1 - FD)) + (FC * FD)) / 24
     C6 = ((((-FB) * a) * (3 - (3 * a) + (a ^ 2)) * (1 - FA)) - (b * (3 -
(3 * b) + (b ^ 2)) * (1 - FB)) + (R * (3 - (3 * R) + (R ^ 2)) * (1 - FR))
- (s * (3 - (3 * s) + (s ^ 2)) * (1 - FR)) + (c * (3 - (3 * c) + (c ^ 2))
* (1 - FC)) + ((FC * d) * (3 - (3 * d) + (d ^ 2)) * (1 - FD)) + (FC * FD))
 / 3
     C7 = (((-FB) * (a ^ 2) * (6 - (8 * a) + (3 * (a ^ 2))) * (1 - FA)) -
((b ^ 2) * (6 - (8 * b) + (3 * (b ^ 2))) * (1 - FB)) + ((R ^ 2) * (6 - (8
* R) + (3 * (R ^ 2))) * (1 - FR)) - ((s ^ 2) * (6 - (8 * s) + (3 * (s ^ 2)
)) * (1 - FR)) + ((c ^ 2) * (6 - (8 * c) + (3 * (c ^ 2))) * (1 - FC)) + ((
FC * (d ^ 2)) * (6 - (8 * d) + (3 * (d ^ 2))) * (1 - FD)) + (FC * FD)) / 2
4
     CR = (C6 * C4) - (C3 * C3)
     Kcc = C4 / CR
     Kc = ((Kcc * GeneralData(CProj).CEc) * ((CC2 * (CC1 ^ 3)) / 12)) / lc
     If frmMain.optSystem(0).Value = True Then
      Cs = ((1 - (.63 * (h / CC2))) * (((h ^ 3) * CC2) / 3))
      Cd = (1 - (.63 * ((h + CDt) / CC2))) * ((((h + CDt) ^ 3) * CC2) / 9)
      X2 = (L2 - CDx) / 2
      Kt = (GeneralData(CProj).CEc / (((L2 * (1 - ((CC1 / L2) ^ 5))) / (20
 * Cd))) + (((L2 * ((1 - (X2 / L2)) ^ 5)) / 20) * ((1 / Cs) - (1 / Cd))))
     Else
      Kt = ((9 * GeneralData(CProj).CEc * ((1 - (.63 * (h / CC2))) * (((h
^ 3) * CC2) / 3))) / (L2 * ((1 - (CC1 / L2)) ^ 3)))
     End If
    End If
    Kec = ((4 * Kc * Kt) / (2 * Kt + 2 * Kc))
    Cal_FColumnK = Kec
End Function

Function Cal_FColumnShear (N As Integer, Index As Integer, d As Single, Wu
 As Single) 'Check Dia & Punch Shear at Column
    Dim b0 As Single
    Dim L1 As Single
    Dim L2 As Single
    Dim lf As Single
    Dim XNum As Integer
    Dim YNum As Integer
    Dim c1 As Single 'Column Width ,Length in x axis
    Dim c2 As Single 'Column Thick ,Length in y axis
    Dim X_Flag As Integer
    Dim Y_Flag As Integer
    Dim EdgeIndex As Integer
    c1 = Column(CProj, N).c1
    c2 = Column(CProj, N).c2
    X_Flag = Column(CProj, N).X_Flag
    Y_Flag = Column(CProj, N).Y_Flag
    L1 = 0
    L2 = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 5

```
    lf = 0
    XNum = 0
    YNum = 0
    Select Case Index
    Case 1: 'Punch
     L1 = 0
     L2 = 0
     lf = 0
     XNum = 0
     YNum = 0
     For I = 1 To ElementXNum(CProj)
      'Span Then Conjunc That Column in X axis Must Less Or Equal 2 Span
      If (SElement(CProj, 0, I).Start = N) Or (SElement(CProj, 0, I).End =
 N) Then
       L1 = L1 + (SElement(CProj, 0, I).Length / 2)
       XNum = XNum + 1
      End If
     Next I
      'Span Then Conjunc That Column in Y axis Must Less Or Equal 2 Span
     For I = 1 To ElementYNum(CProj)
      If (SElement(CProj, 1, I).Start = N) Or (SElement(CProj, 1, I).End =
 N) Then
       L2 = L2 + (SElement(CProj, 1, I).Length / 2)
       YNum = YNum + 1
      End If
     Next I
     If (((X_Flag = 1) Or (X_Flag = 2)) And ((Y_Flag = 1) Or (Y_Flag = 2))
) Then
      EdgeIndex = 1
     ElseIf ((((X_Flag = 1) Or (X_Flag = 2)) And (Y_Flag = 0)) Or ((X_Flag
 = 0) And ((Y_Flag = 1) Or (Y_Flag = 2)))) Then
      EdgeIndex = 2
     ElseIf ((X_Flag = 0) And (Y_Flag = 0)) Then
      EdgeIndex = 3
     End If
     Select Case EdgeIndex
     Case 1: 'punch corner
      vn = (Wu * ((L1 * L2) - ((c1 + (d / 2)) * (c2 + d / 2)))) / .85
     Case 2: 'punch edge
      vn = (Wu * ((L1 * L2) - ((c1 + (XNum * d / 2)) * (c2 + (YNum * d / 2
))))) / .85
     Case 3: 'punch interior
      vn = (Wu * ((L1 * L2) - (c1 + d) * (c2 + d))) / .85
     End Select
    Case 2: 'Dia
     lf = 0
     XNum = 0
     YNum = 0
     For I = 1 To ElementXNum(CProj) + ElementYNum(CProj)
      If ((EElement(CProj, I).Start = N) Or (EElement(CProj, I).End = N))
And (lf < EElement(CProj, I).Length) Then
       lf = (EElement(CProj, I).Length / 2)
       XNum = 1
      Else
       'XNum = 0
      End If
      If ((EElement(CProj, I).Start = N) Or (EElement(CProj, I).End = N))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 6

```
And (lf < EElement(CProj, I).Length) Then
        lf = (EElement(CProj, I).Length / 2)
        YNum = 1
       Else
        'YNum = 0
       End If
      Next I
      If (((X_Flag = 1) Or (X_Flag = 2)) And ((Y_Flag = 1) Or (Y_Flag = 2))
) Then
       EdgeIndex = 1
      ElseIf ((((X_Flag = 1) Or (X_Flag = 2)) And (Y_Flag = 0)) Or ((X_Flag
 = 0) And ((Y_Flag = 1) Or (Y_Flag = 2)))) Then
       EdgeIndex = 2
      ElseIf ((X_Flag = 0) And (Y_Flag = 0)) Then
       EdgeIndex = 3
      End If
      Select Case EdgeIndex     'Check Shear For 1 m strip
      Case 1: 'dia corner
       vn = (Wu * (lf - d - (XNum * c1) - (YNum * c2))) / .85
      Case 2: 'dia edge
       vn = (Wu * (lf - d - (XNum * (c1 / 2)) - (YNum * (c2 / 2)))) / .85
      Case 3: 'dia interior
       vn = (Wu * (lf - d - (XNum * c1 / 2) - (YNum * c2 / 2))) / .85
      End Select
     End Select
     Cal_FColumnShear = vn
End Function


Function Cal_FCover (Thk As Single) As Single
     Dim Cover As Single
     If ((Thk > 0) And (Thk <= 12)) Then
      Cover = .025
     End If
     If ((Thk > 12) And (Thk <= 20)) Then
      Cover = .03
     End If
     If ((Thk > 20) And (Thk <= 40)) Then
      Cover = .04
     End If
     If (Thk > 40) Then
      Cover = .05
     End If
     Cal_FCover = Cover
End Function

Function Cal_FDF (E As Integer, XY As Integer, LR As Integer) As Single
     Dim I As Integer
     Dim SumDF As Single
     SumDF = 0
     If LR Then
      For I = 1 To ElementNum(CProj)
       If (Element(CProj, E).Start = Element(CProj, I).Start) Or (Element(C
Proj, E).Start = Element(CProj, I).End) Then
        SumDF = SumDF + Cal_FK(I, XY)
       End If
      Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 7

```
    ElseIf Not (LR) Then
     For I = 1 To ElementNum(CProj)
      If (Element(CProj, E).End = Element(CProj, I).Start) Or (Element(CPr
oj, E).End = Element(CProj, I).End) Then
       SumDF = SumDF + Cal_FK(I, XY)
      End If
     Next I
    End If
    If Not (SumDF = 0) Then
     Cal_FDF = (Cal_FK(E, XY) / SumDF)
    Else
     Cal_FDF = 1
    End If
End Function

Sub Cal_FDropCapital (CN As Integer, h As Single)  'h is Thickness
    Dim Thick As Single
    Dim Lmax As Single 'Now Lmax is Maximum Length of Span
                       'For Adapt Lmax is Maximum Length That conjunc The
Column
    Dim d As Single     'if true is find lmax in x axis or y axis around c
olumn
    Lmax = 0
    For I = 1 To ElementXNum(CProj) + ElementYNum(CProj)
     If ((EElement(CProj, I).Start = CN) Or (EElement(CProj, I).End = CN))
 And (EElement(CProj, I).Length > Lmax) Then
      Lmax = EElement(CProj, I).Length
     End If
    Next I
    Column(CProj, CN).DropThick = .375 * h
    Column(CProj, CN).DropX = Lmax / 3
    Column(CProj, CN).DropY = Lmax / 3
    Column(CProj, CN).CapitalX = .225 * Lmax
    Column(CProj, CN).CapitalY = .225 * Lmax
    If frmMain.chkCHead(0).Value Then
     Column(CProj, CN).Type = 1
    End If
    If frmMain.chkCHead(1).Value Then
     Column(CProj, CN).Type = 2
    End If
    If frmMain.chkCHead(0).Value And frmMain.chkCHead(1).Value Then
     Column(CProj, CN).Type = 3
    End If
    If frmMain.optSystem(0).Value Then
     Column(CProj, CN).Type = 0
    End If
    If Column(CProj, CN).Type = 3 Then
     Column(CProj, CN).DropThick = .25 * h
    End If
    Select Case Column(CProj, CN).Type
    Case 0:
     Column(CProj, CN).DropThick = 0
     Column(CProj, CN).DropX = 0
     Column(CProj, CN).DropY = 0
     Column(CProj, CN).CapitalX = 0
     Column(CProj, CN).CapitalY = 0
    Case 1:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 8

```
      Column(CProj, CN).CapitalX = 0
      Column(CProj, CN).CapitalY = 0
     Case 2:
      Column(CProj, CN).DropThick = 0
      Column(CProj, CN).DropX = 0
      Column(CProj, CN).DropY = 0
     Case 3:
     End Select
End Sub

Sub Cal_FEndMoment (ColOrRow As Integer, ColRow As Integer)
    Dim Pattern As Integer
    Dim axis As Integer
    Const ErrorCheck = 1
    Dim ErrorValue As Single
    Dim I As Integer
    If ColOrRow Then
     axis = 0
    ElseIf Not ColOrRow Then
     axis = 1
    End If
    ReDim FEML(ElementNum(CProj)) As Single
    ReDim FEMR(ElementNum(CProj)) As Single
    ReDim DFL(ElementNum(CProj)) As Single
    ReDim DFR(ElementNum(CProj)) As Single
    ReDim COML(ElementNum(CProj)) As Single
    ReDim COMR(ElementNum(CProj))  As Single
    ReDim DML(ElementNum(CProj)) As Single
    ReDim DMR(ElementNum(CProj)) As Single
    ReDim MNode(NodeNum(CProj)) As Single
    ReDim MMaxL(ElementNum(CProj)) As Single
    ReDim MMaxR(ElementNum(CProj)) As Single
    For I = 1 To ElementNum(CProj)
     MMaxL(I) = 0
     MMaxR(I) = 0
    Next I
  For Pattern = 1 To 3
   For I = 1 To ElementNum(CProj)
    R = Cal_FK(I, Not (ColOrRow))
    FEML(I) = Cal_FFEM(axis, I, True, Pattern)
    FEMR(I) = Cal_FFEM(axis, I, False, Pattern)
    DFL(I) = Cal_FDF(I, Not (ColOrRow), True)
    DFR(I) = Cal_FDF(I, Not (ColOrRow), False)
    COFL(CProj, I) = COFL(CProj, I)
    COFR(CProj, I) = COFR(CProj, I)
    ML(CProj, I) = FEML(I)
    MR(CProj, I) = FEMR(I)
    DML(I) = 0
    DMR(I) = 0
    COML(I) = 0
    COMR(I) = 0
  Next I
  'Distribution Moment
  Do
  ErrorValue = 0
    'Sum DM
   For I = 1 To NodeNum(CProj)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 9

```
        MNode(I) = 0
        For J = 1 To ElementNum(CProj)
         If (Element(CProj, J).Start) = I Then
          MNode(I) = (MNode(I) + ML(CProj, J))
         ElseIf (Element(CProj, J).End = I) Then
          MNode(I) = (MNode(I) + MR(CProj, J))
         End If
        Next J
       Next I
        'DM
       For I = 1 To NodeNum(CProj)
        For J = 1 To ElementNum(CProj)
         If (Element(CProj, J).Start = I) Then
          DML(J) = (((-1) * MNode(I)) * DFL(J))
         ElseIf (Element(CProj, J).End = I) Then
          DMR(J) = (((-1) * MNode(I)) * DFR(J))
         End If
        Next J
       Next I
       For I = 1 To ElementNum(CProj)
        If Abs(ErrorValue) < Abs(DML(I)) Then
         ErrorValue = DML(I)
        End If
        If Abs(ErrorValue) < Abs(DMR(I)) Then
         ErrorValue = DMR(I)
        End If
       Next I
       'Carry over moment
       For I = 1 To ElementNum(CProj)
        COML(I) = DMR(I) * COFL(CProj, I)
        COMR(I) = DML(I) * COFR(CProj, I)
       Next I
        'Sum Moment
       If (Abs(ErrorValue) < ErrorCheck) Then
        For I = 1 To ElementNum(CProj)
         ML(CProj, I) = ML(CProj, I) + DML(I)
         MR(CProj, I) = MR(CProj, I) + DMR(I)
        Next I
       Else
        For I = 1 To ElementNum(CProj)
         ML(CProj, I) = ML(CProj, I) + DML(I) + COML(I)
         MR(CProj, I) = MR(CProj, I) + DMR(I) + COMR(I)
        Next I
       End If
      Loop Until (Abs(ErrorValue) < ErrorCheck)
      For I = 1 To ElementNum(CProj)
       If Abs(MMaxL(I)) < Abs(ML(CProj, I)) Then
        MMaxL(I) = ML(CProj, I)
       End If
       If Abs(MMaxR(I)) < Abs(MR(CProj, I)) Then
        MMaxR(I) = MR(CProj, I)
       End If
      MD(CProj, I) = (1 / 8) * ((DeadCo * DeadLoad(CProj)) + (1.8 * LiveLoa
d(CProj))) * ((Element(CProj, I).L1 + Element(CProj, I).L2) / 2) * (Elemen
t(CProj, I).Length) ^ 2
     Next I
    Next Pattern
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 10

```
    For I = 1 To ElementNum(CProj)
     ML(CProj, I) = MMaxL(I)
     MR(CProj, I) = MMaxR(I)
     MD(CProj, I) = MD(CProj, I) - (ML(CProj, I) + MR(CProj, I)) / 2
    Next I

     '''' Show End Moment at Programmed
     'Load frmOutText(CProj)
     'frmOutText(CProj).Show
     'frmOutText(CProj).Cls
     'For I = 1 To ElementNum(CProj)
      ' frmOutText(CProj).Print "Fixed end moment Left element" + Format$(I)
  + " = " + Format$(FEML(I), "0.00")
      ' frmOutText(CProj).Print "Fixed end moment Right element" + Format$(I
 ) + " = " + Format$(FEMR(I), "0.00")
      ' frmOutText(CProj).Print "DF Factor Left " + Format$(DFL(I), "0.000")
      ' frmOutText(CProj).Print "DF Factor Right " + Format$(DFR(I), "0.000"
 )
      ' frmOutText(CProj).Print "End Moment Left " + Format$(ML(CProj, I), "
 0.00")
      ' frmOutText(CProj).Print "End Moment Right " + Format$(MR(CProj, I),
 "0.00")
     'Next I
End Sub

Function Cal_FFEM (axis As Integer, E As Integer, LeftOrRight As Integer,
Pattern As Integer) As Single
    Dim L1 As Single
    Dim L2 As Single
    Dim L3 As Single
    Dim Wu As Single
    Dim N As Integer
    Wu = Cal_FWu(E, LeftOrRight, Pattern)
    L1 = Element(CProj, E).Length
    L2 = ((Element(CProj, E).L1 / 2) + (Element(CProj, E).L2 / 2))
    If LeftOrRight Then
     Cal_FFEM = (-1) * FEMCL(CProj, E) * Wu * L2 * L1 ^ 2
    ElseIf Not LeftOrRight Then
     Cal_FFEM = FEMCL(CProj, E) * Wu * L2 * L1 ^ 2
    End If
End Function

Function Cal_FK (E As Integer, XY As Integer) As Single 'Element of Slab s
tiffness
    Dim EC As Single
    Dim Ks As Single
    Dim ka As Single
    Dim kb As Single
    Dim L1 As Single
    Dim L2 As Single
    Dim a As Single
    Dim b As Single
    Dim R As Single
    Dim s As Single
    Dim c As Single
    Dim d As Single
    Dim I1 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 11

```
    Dim I2 As Single
    Dim I3 As Single
    Dim I4 As Single
    Dim I5 As Single
    Dim I6 As Single
    Dim Y_ As Single
    Dim CC1 As Single
    Dim CC2 As Single
    Dim CDx As Single
    Dim CDy As Single
    Dim CDt As Single
    Dim CCx As Single
    Dim CCy As Single
    Dim CCL As Single
    Dim c1 As Single
    Dim c2 As Single
    Dim C3 As Single
    Dim C4 As Single
    Dim C5 As Single
    Dim C6 As Single
    Dim C7 As Single
    Dim FA As Single
    Dim FB As Single
    Dim FR As Single
    Dim FC As Single
    Dim FD As Single
    Dim CR As Single
    Dim h As Single 'Thickness
    Dim NodeA As Integer
    Dim NodeB As Integer
    Dim ElementNum As Integer
    h = Thick(CProj)
    EC = GeneralData(CProj).CEc
    L1 = Element(CProj, E).Length
    L2 = ((Element(CProj, E).L1 / 2) + (Element(CProj, E).L2 / 2))
    NodeA = Element(CProj, E).CStart
    NodeB = Element(CProj, E).CEnd
    If XY Then
     a = (Column(CProj, NodeA).c1 / 2) / L1
     b = (Column(CProj, NodeA).CapitalX / 2) / L1
     R = (Column(CProj, NodeA).DropX / 2) / L1
     s = (L1 - (Column(CProj, NodeB).DropX / 2)) / L1
     c = (L1 - (Column(CProj, NodeB).CapitalX / 2)) / L1
     d = (L1 - (Column(CProj, NodeB).c1 / 2)) / L1
     CC1 = Column(CProj, NodeA).c1
     CC2 = Column(CProj, NodeA).c2
     CDx = Column(CProj, NodeA).DropX
     CDy = Column(CProj, NodeA).DropY
     CDt = Column(CProj, NodeA).DropThick
     CCx = Column(CProj, NodeA).CapitalX
     CCy = Column(CProj, NodeA).CapitalY
     CCL = Column(CProj, NodeA).Length
     Y_ = ((h * L2) * ((h / 2) + CDt) + (CDy * (CDt ^ 2) * .5)) / ((h * L2
) + (CDy * CDt))
     I6 = (L2 * (h ^ 3)) / 12
     I1 = I6 + ((h * L2) * (((h / 2) - Y_) ^ 2)) + ((CDy * (CDt ^ 3)) / 12
) + (CDy * CDt * (Y_ - (CDt / 2)))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 12

```
      I2 = I1 / ((1 - (CC2 / L2)) ^ 2)
      I4 = I2
      CC1 = Column(CProj, NodeB).c1
      CC2 = Column(CProj, NodeB).c2
      CDx = Column(CProj, NodeB).DropX
      CDy = Column(CProj, NodeB).DropY
      CDt = Column(CProj, NodeB).DropThick
      CCx = Column(CProj, NodeB).CapitalX
      CCy = Column(CProj, NodeB).CapitalY
      CCL = Column(CProj, NodeB).Length
      Y_ = ((h * L2) * ((h / 2) + CDt) + (CDy * (CDt ^ 2) * .5)) / ((h * L2
) + (CDy * CDt))
      I1 = I6 + ((h * L2) * (((h / 2) - Y_) ^ 2)) + ((CDy * (CDt ^ 3)) / 12
) + (CDy * CDt * (Y_ - (CDt / 2)))
      I3 = I1 / ((1 - (CC2 / L2)) ^ 2)
      I5 = I3
      FA = (1 - (Column(CProj, NodeA).c2 / L2)) ^ 2
      FB = I1 / I2
      FR = I1 / I6
      FC = I1 / I3
      FD = (1 - (Column(CProj, NodeB).c2 / L2)) ^ 2
      C3 = (((-FB) * a * a * (3 - (2 * a)) * (1 - FA)) - ((b * b) * (3 - (2
 * b)) * (1 - FB)) + ((R ^ 2) * (3 - (2 * R)) * (1 - FR)) - ((s ^ 2) * (3
 - (2 * s)) * (1 - FR)) + ((c ^ 2) * (3 - (2 * c)) * (1 - FC)) + ((FC * d *
 d) * (3 - (2 * d)) * (1 - FD)) + (FC * FD)) / 6
      C4 = (((-FB * (a ^ 3)) * (1 - FA)) - ((b ^ 3) * (1 - FB)) + ((R ^ 3)
* (1 - FR)) - ((s ^ 3) * (1 - FR)) + ((c ^ 3) * (1 - FC)) + ((FC * (d ^ 3)
) * (1 - FD)) + (FC * FD)) / 3
      C5 = ((((-FB) * (a ^ 3)) * (4 - (3 * a)) * (1 - FA)) - ((b ^ 3) * (4
 - (3 * b)) * (1 - FB)) + ((R ^ 3) * (4 - (3 * R)) * (1 - FR)) - ((s ^ 3) *
 (4 - (3 * s)) * (1 - FR)) + ((c ^ 3) * (4 - (3 * c)) * (1 - FC)) + ((FC *
 (d ^ 3)) * (4 - (3 * d)) * (1 - FD)) + (FC * FD)) / 24
      C6 = ((((-FB) * a) * (3 - (3 * a) + (a ^ 2)) * (1 - FA)) - (b * (3 -
(3 * b) + (b ^ 2)) * (1 - FB)) + (R * (3 - (3 * R) + (R ^ 2)) * (1 - FR))
 - (s * (3 - (3 * s) + (s ^ 2)) * (1 - FR)) + (c * (3 - (3 * c) + (c ^ 2))
* (1 - FC)) + ((FC * d) * (3 - (3 * d) + (d ^ 2)) * (1 - FD)) + (FC * FD))
 / 3
      C7 = (((-FB) * (a ^ 2) * (6 - (8 * a) + (3 * (a ^ 2))) * (1 - FA)) -
((b ^ 2) * (6 - (8 * b) + (3 * (b ^ 2))) * (1 - FB)) + ((R ^ 2) * (6 - (8
* R) + (3 * (R ^ 2))) * (1 - FR)) - ((s ^ 2) * (6 - (8 * s) + (3 * (s ^ 2)
)) * (1 - FR)) + ((c ^ 2) * (6 - (8 * c) + (3 * (c ^ 2))) * (1 - FC)) + ((
FC * (d ^ 2)) * (6 - (8 * d) + (3 * (d ^ 2))) * (1 - FD)) + (FC * FD)) / 2
4
      CR = (C6 * C4) - (C3 * C3)
      COFL(CProj, E) = C3 / C4
      COFR(CProj, E) = C3 / C6
      ka = C4 / CR
      kb = C6 / CR
      FEMCL(CProj, E) = ((C7 * C4) - (C5 * C3)) / CR
      FEMCR(CProj, E) = ((C6 * C5) - (C7 * C3)) / CR
      Ks = ((ka * (GeneralData(CProj).SEc) * (L2 * (h ^ 3))) / 12) / L1
    End If
    If Not (XY) Then
      a = (Column(CProj, NodeA).c2 / 2) / L1
      b = (Column(CProj, NodeA).CapitalY / 2) / L1
      R = (Column(CProj, NodeA).DropY / 2) / L1
      s = (L1 - (Column(CProj, NodeB).DropY / 2)) / L1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 13

```
        c = (L1 - (Column(CProj, NodeB).CapitalY / 2)) / L1
        d = (L1 - (Column(CProj, NodeB).c2 / 2)) / L1
        CC1 = Column(CProj, NodeA).c2
        CC2 = Column(CProj, NodeA).c1
        CDx = Column(CProj, NodeA).DropY
        CDy = Column(CProj, NodeA).DropX
        CDt = Column(CProj, NodeA).DropThick
        CCx = Column(CProj, NodeA).CapitalY
        CCy = Column(CProj, NodeA).CapitalX
        CCL = Column(CProj, NodeA).Length
        Y_ = ((h * L2) * ((h / 2) + CDt) + (CDy * (CDt ^ 2) * .5)) / ((h * L2
) + (CDy * CDt))
        I6 = (L2 * (h ^ 3)) / 12
        I1 = I6 + ((h * L2) * (((h / 2) - Y_) ^ 2)) + ((CDy * (CDt ^ 3)) / 12
) + (CDy * CDt * (Y_ - (CDt / 2)))
        I2 = I1 / ((1 - (CC2 / L2)) ^ 2)
        I4 = I2
        CC1 = Column(CProj, NodeB).c2
        CC2 = Column(CProj, NodeB).c1
        CDx = Column(CProj, NodeB).DropY
        CDy = Column(CProj, NodeB).DropX
        CDt = Column(CProj, NodeB).DropThick
        CCx = Column(CProj, NodeB).CapitalY
        CCy = Column(CProj, NodeB).CapitalX
        CCL = Column(CProj, NodeB).Length
        Y_ = ((h * L2) * ((h / 2) + CDt) + (CDy * (CDt ^ 2) * .5)) / ((h * L2
) + (CDy * CDt))
        I1 = I6 + ((h * L2) * (((h / 2) - Y_) ^ 2)) + ((CDy * (CDt ^ 3)) / 12
) + (CDy * CDt * (Y_ - (CDt / 2)))
        I3 = I1 / ((1 - (CC2 / L2)) ^ 2)
        I5 = I3
        FA = (1 - (Column(CProj, NodeA).c1 / L2)) ^ 2
        FB = I1 / I2
        FR = I1 / I6
        FC = I1 / I3
        FD = (1 - (Column(CProj, NodeB).c1 / L2)) ^ 2
        C3 = (((-FB) * a * a * (3 - (2 * a)) * (1 - FA)) - ((b * b) * (3 - (2
 * b)) * (1 - FB)) + ((R ^ 2) * (3 - (2 * R)) * (1 - FR)) - ((s ^ 2) * (3
 - (2 * s)) * (1 - FR)) + ((c ^ 2) * (3 - (2 * c)) * (1 - FC)) + ((FC * d *
 d) * (3 - (2 * d)) * (1 - FD)) + (FC * FD)) / 6
        C4 = (((-FB * (a ^ 3)) * (1 - FA)) - ((b ^ 3) * (1 - FB)) + ((R ^ 3)
* (1 - FR)) - ((s ^ 3) * (1 - FR)) + ((c ^ 3) * (1 - FC)) + ((FC * (d ^ 3)
) * (1 - FD)) + (FC * FD)) / 3
        C5 = ((((-FB) * (a ^ 3)) * (4 - (3 * a)) * (1 - FA)) - ((b ^ 3) * (4
 - (3 * b)) * (1 - FB)) + ((R ^ 3) * (4 - (3 * R)) * (1 - FR)) - ((s ^ 3) *
 (4 - (3 * s)) * (1 - FR)) + ((c ^ 3) * (4 - (3 * c)) * (1 - FC)) + ((FC *
 (d ^ 3)) * (4 - (3 * d)) * (1 - FD)) + (FC * FD)) / 24
        C6 = ((((-FB) * a) * (3 - (3 * a) + (a ^ 2)) * (1 - FA)) - (b * (3 -
(3 * b) + (b ^ 2)) * (1 - FB)) + (R * (3 - (3 * R) + (R ^ 2)) * (1 - FR))
 - (s * (3 - (3 * s) + (s ^ 2)) * (1 - FR)) + (c * (3 - (3 * c) + (c ^ 2))
 * (1 - FC)) + ((FC * d) * (3 - (3 * d) + (d ^ 2)) * (1 - FD)) + (FC * FD))
 / 3
        C7 = (((-FB) * (a ^ 2) * (6 - (8 * a) + (3 * (a ^ 2))) * (1 - FA)) -
((b ^ 2) * (6 - (8 * b) + (3 * (b ^ 2))) * (1 - FB)) + ((R ^ 2) * (6 - (8
* R) + (3 * (R ^ 2))) * (1 - FR)) - ((s ^ 2) * (6 - (8 * s) + (3 * (s ^ 2)
)) * (1 - FR)) + ((c ^ 2) * (6 - (8 * c) + (3 * (c ^ 2))) * (1 - FC)) + ((
FC * (d ^ 2)) * (6 - (8 * d) + (3 * (d ^ 2))) * (1 - FD)) + (FC * FD)) / 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 14

```
4
      CR = (C6 * C4) - (C3 * C3)
      COFL(CProj, E) = C3 / C4
      COFR(CProj, E) = C3 / C6
      ka = C4 / CR
      kb = C6 / CR
      FEMCL(CProj, E) = ((C7 * C4) - (C5 * C3)) / CR
      FEMCR(CProj, E) = ((C6 * C5) - (C7 * C3)) / CR
      Ks = ((ka * (GeneralData(CProj).SEc) * (L2 * (h ^ 3))) / 12) / L1
    End If
    Cal_FK = Ks
End Function



Function Cal_FMoment (Index As Integer, ElementNum As Integer, Element As
ElementType) As Single
    Dim w As Single
    Dim L1 As Single
    Dim L2 As Single
    Select Case Index
    Case 1: 'Left

    Case 2: 'Mid
    Case 3: 'Right
    End Select
End Function

Sub Cal_FReinforce ()

End Sub

Function Cal_FThick () As Single
    Dim Lmax As Single
    Dim Thick As Single
    Dim c As Single
    Dim E As Integer
    Lmax = 0
    For I = 1 To ElementXNum(CProj) 'find lmax in x axis
     If SElement(CProj, 0, I).Length > Lmax Then
      Lmax = SElement(CProj, 0, I).Length
      c = Column(CProj, SElement(CProj, 0, I).Start).c1
     End If
    Next I
    For I = 1 To ElementYNum(CProj) 'find lmax in y axis
     If SElement(CProj, 1, I).Length > Lmax Then
      Lmax = SElement(CProj, 1, I).Length
      c = Column(CProj, SElement(CProj, 1, I).Start).c2
     End If
    Next I
    h = (((Lmax - c) * (.8 + ((2.844 * GeneralData(CProj).RFs_) / 40000)))
/ 36)
    Select Case GeneralData(CProj).SlabIndex 'thick for flat plate or flat
slab
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 15

```
    Case 0:
     h = h
    Case 1, 2, 3:
     h = 1.1 * h
    End Select
    Dim x As Integer
    h = h * 100
    x = Int(h)
    If h - x > 0 Then
     h = x + 1  'ÈÑ´¢ÖéÊ
    ElseIf h - x = 0 Then
     h = x
    End If
    h = h / 100
    Cal_FThick = h
End Function

Function Cal_FWu (E As Integer, LeftOrRight As Integer, Pattern As Integer
) As Single 'Check By Load Pattern 3 Case
    Dim Wu As Single
    Select Case Pattern
    Case 0:
     Wu = DeadLoad(CProj)
    Case 1:
     Wu = 1.8 * LiveLoad(CProj) + DeadCo * DeadLoad(CProj)
    Case 2:
     If (E Mod 2) = 1 Then
      Wu = DeadCo * DeadLoad(CProj) + .75 * LiveCo * LiveLoad(CProj)
     ElseIf (E Mod 2) = 0 Then
      Wu = DeadCo * DeadLoad(CProj)
     End If
    Case 3:
     If (E Mod 2) = 1 Then
      Wu = DeadCo * DeadLoad(CProj)
     ElseIf (E Mod 2) = 0 Then
      Wu = DeadCo * DeadLoad(CProj) + .75 * LiveCo * LiveLoad(CProj)
     End If
    Case 4:
     Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj) + DeadCo * S
uperLoad(CProj)
    Case 5:
     Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    End Select
    Cal_FWu = Wu
End Function


Function Check_FFShear (Diag As Integer, b0 As Single, d As Single, N As I
nteger) As Single
    Dim Vc As Single 'Diagonal check diag  = True
    Dim vc1 As Single
    Dim vc2 As Single
    Dim vc3 As Single
    Dim Bs As Single
    Dim AAs As Single
    Dim c1 As Single
    Dim c2 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 16

```
    Dim X_Flag As Integer
    Dim Y_Flag As Integer
    c1 = Column(CProj, N).c1
    c2 = Column(CProj, N).c2
    X_Flag = Column(CProj, N).X_Flag
    Y_Flag = Column(CProj, N).Y_Flag
    Dim EdgeIndex As Integer
     If (((X_Flag = 1) Or (X_Flag = 2)) And ((Y_Flag = 1) Or (Y_Flag = 2))
) Then
      EdgeIndex = 1       'Corner
     ElseIf ((((X_Flag = 1) Or (X_Flag = 2)) And (Y_Flag = 0)) Or ((X_Flag
 = 0) And ((Y_Flag = 1) Or (Y_Flag = 2)))) Then
      EdgeIndex = 2       'Edge
     ElseIf ((X_Flag = 0) And (Y_Flag = 0)) Then
      EdgeIndex = 3       'Interior
     End If
    Select Case EdgeIndex
    Case 1: 'Corner
     AAs = 20
    Case 2: 'Edge
     AAs = 30
    Case 3: 'Interior
     AAs = 40
    End Select
    Bs = (c1 / c2)
    If Bs < 1 Then
     Bs = (Bs ^ -1)
    End If
    If Diag Then
     Vc = (.53) * Sqr(GeneralData(CProj).SFc_) * b0 * d
     Vc = Vc * 10000
    End If
    'punch checking
    vc1 = (.53 + (1.06 / Bs)) * Sqr(GeneralData(CProj).SFc_) * b0 * d
    vc2 = .265 * ((AAs / (d * b0)) + 2) * Sqr(GeneralData(CProj).SFc_) * b
0 * d
    vc3 = 1.06 * Sqr(GeneralData(CProj).SFc_) * b0 * d
    If Not (Diag) Then 'Find minimum vc
     Vc = vc1
     If vc2 < Vc Then
      Vc = vc2
     End If
     If vc3 < Vc Then
      Vc = vc3
     End If
     Vc = Vc * 10000
    End If
    Check_FFShear = Vc
End Function

Function Check_FShear (Diag As Integer, vn As Single, b0 As Single, d As S
ingle, N As Integer) As Integer 'Unit Shear That Checked If True Check Pas
s,If False Check Not Pass
    Dim Vc As Single 'Diagonal check diag  = True
    Dim vc1 As Single
    Dim vc2 As Single
    Dim vc3 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FCOMPUTE.BAS - 17

```
    Dim Bs As Single
    Dim AAs As Single
    Dim c1 As Single
    Dim c2 As Single
    Dim X_Flag As Integer
    Dim Y_Flag As Integer
    c1 = Column(CProj, N).c1
    c2 = Column(CProj, N).c2
    X_Flag = Column(CProj, N).X_Flag
    Y_Flag = Column(CProj, N).Y_Flag
    Dim EdgeIndex As Integer
     If (((X_Flag = 1) Or (X_Flag = 2)) And ((Y_Flag = 1) Or (Y_Flag = 2))
) Then
      EdgeIndex = 1       'Corner
     ElseIf ((((X_Flag = 1) Or (X_Flag = 2)) And (Y_Flag = 0)) Or ((X_Flag
 = 0) And ((Y_Flag = 1) Or (Y_Flag = 2)))) Then
      EdgeIndex = 2       'Edge
     ElseIf ((X_Flag = 0) And (Y_Flag = 0)) Then
      EdgeIndex = 3       'Interior
     End If
    Select Case EdgeIndex
    Case 1: 'Corner
     AAs = 20
    Case 2: 'Edge
     AAs = 30
    Case 3: 'Interior
     AAs = 40
    End Select
    Bs = (c1 / c2)
    If Bs < 1 Then
     Bs = (Bs ^ -1)
    End If
    If Diag Then
     Vc = (.53) * Sqr(GeneralData(CProj).SFc_) * b0 * d
     Vc = Vc * 10000
    End If
    'punch checking
    vc1 = (.53 + (1.06 / Bs)) * Sqr(GeneralData(CProj).SFc_) * b0 * d
    vc2 = .265 * (((AAs * d) / b0) + 2) * Sqr(GeneralData(CProj).SFc_) * b
0 * d
    vc3 = 1.06 * Sqr(GeneralData(CProj).SFc_) * b0 * d
    If Not (Diag) Then 'Find minimum vc
     Vc = vc1
     If vc2 < Vc Then
      Vc = vc2
     End If
     If vc3 < Vc Then
      Vc = vc3
     End If
     Vc = Vc * 10000
    End If
    If Vc >= vn Then
     Check_FShear = True
    Else
     Check_FShear = False
    End If
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 1

```
Sub Access_Data ()
    'Load
    LiveCo = CSng(frmMain.txtLLFactor.Text)
    DeadCo = CSng(frmMain.txtDLFactor.Text)
    LiveLoad(CProj) = CSng(frmMain.txtLL.Text)
    SuperLoad(CProj) = CSng(frmMain.txtSL.Text)
End Sub

Sub Cal_BSelectSection (Mt As Single, Mg As Single, ML As Single)
    Dim h As Single
    Const k = 1.6
    h = k * Sqr(Mt)
    Do

    Loop Until ((Mg / Mt) > .3)
End Sub

Function Cal_FrictionLoss (al As Single, L As Single)
    Cal_FrictionLoss = Format((al * UfrictionAngle(CProj) + L * KfrictionL
ength(CProj)) * 100, "###.##")
End Function

Function Cal_LossOfTime (SType As Integer, PType As Integer, TimeApply As
Integer, fcpi As Single, fcpa As Single, fc As Single, fcds As Single, Vol
ume As Single, Surface As Single, fpi As Single)
    'Import data is
    '1. SType = StrandType
    '2. fcpi
    '3. fcpa
    '4. fc
    '5. fcds
    '6. TimeApply = Time apply load
    Dim Esl As Single
    Dim Kes As Single
    Dim Es As Single
    Dim Eci As Single
    Dim SH As Single
    Dim Ksh As Single
    Dim CR As Single
    Dim RH As Single
    Dim Kcir As Single
    Dim fcir As Single
    Dim fg As Single
    Dim RE As Single
    Dim Kre As Single
    Dim c As Single
    'Compute require data
    Eci = 4730 * (fci ^ (1 / 2))    'fci determine from dead load only
    Ec = 4730 * (GeneralData(CProj).SFc_ ^ (1 / 2))
    Es = GeneralData(CProj).SEs
    fcir = (Kcir * fcpi) * fg
    fpu = GeneralData(CProj).SFs_
    RH = 70

    Select Case TimeApply
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 2

```
    Case 1: Ksh = .92
    Case 3: Ksh = .85
    Case 5: Ksh = .8
    Case 7: Ksh = .77
    Case 10: Ksh = .73
    Case 20: Ksh = .64
    Case 30: Ksh = .58
    Case 60: Ksh = .45
    End Select
    Select Case PType
    Case 0: 'Post-tensioned member
     Kcr = 1.6
     Kes = .5
     Kcir = 1
    Case 1: 'Pre-tensioned member
     Kcr = 1.6
     Kes = .5
     Kcir = 1
    End Select
    Select Case StrandIndex(CProj)
    Case 0:
     Kre = 137.9
     J = .15
    Case 1:
     Kre = 127.7
     J = .14
    Case 2:
     Kre = 121.4
     J = .13
    Case 3:
     Kre = 34.5
     J = .04
    Case 4:
     Kre = 31.9
     J = .037
    Case 5:
     Kre = 30.3
     J = .035
    Case 6:
     Kre = 41.4
     J = .05
    End Select
    Select Case StrandIndex(CProj) 'specify by strand input
    Case 0, 1, 2:
     Select Case Format((fpi / fpu), ".##")
     Case .8:
     Case .79:
     Case .78:
     Case .77:
     Case .76:
     Case .75: c = 1.45
     Case .74: c = 1.36
     Case .73: c = 1.27
     Case .72: c = 1.18
     Case .71: c = 1.09
     Case .7: c = 1
     Case .69: c = .94
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 3

```
      Case .68: c = .89
      Case .67: c = .83
      Case .66: c = .78
      Case .65: c = .73
      Case .64: c = .68
      Case .63: c = .63
      Case .62: c = .58
      Case .61: c = .53
      Case .6: c = .49
      End Select
     Case 3, 4, 5, 6:
      Select Case Format((fpi / fpu), ".##")
      Case .8: c = 1.38
      Case .79: c = 1.22
      Case .78: c = 1.16
      Case .77: c = 1.11
      Case .76: c = 1.05
      Case .75: c = 1
      Case .74: c = .95
      Case .73: c = .9
      Case .72: c = .85
      Case .71: c = .8
      Case .7: c = .75
      Case .69: c = .7
      Case .68: c = .66
      Case .67: c = .61
      Case .66: c = .57
      Case .65: c = .53
      Case .64: c = .49
      Case .63: c = .45
      Case .62: c = .41
      Case .61: c = .37
      Case .6: c = .33
      End Select
    End Select
    'Elastic Shortening of concrete
    Select Case SType
    Case 0: 'BOND
     Esl = Kes * (Es / Eci) * fcir 'fcir = (Kcir*fcpi)-fg,Kcir = 1
    Case 1: 'UNBOND
     Esl = Kes * (Es / Eci) * fc  'fc = Average stress ,Kes = 0.5
    End Select
    'Shrinkage of concrete (SH)
    SH = ((8.2) / 1000000) * Ksh * Es * (1 - .024 * (Volume / Surface)) *
(100 - 70)  '100-RH ~ 30
    'Creep of concrete (CR)
    Select Case SType
    Case 0: 'BOND
     CR = Kcr * (Es / Ec) * fcpa 'fcpa = stress average between end & At p
oint
    Case 1: 'UNBOND
     CR = Kcr * (Es / Ec) * (fcir - fcds) 'fcds = stress from Dead Load (D
.L.)
    End Select
    'Relaxation (RE)
    RE = (Kre - (J * (SH + CR + Esl))) * c
    Cal_LossOfTime = Esl + SH + CR + RE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 4

```
End Function

Function Cal_Profile (ByVal COR As Integer, ByVal ColRow As Integer, ByVal
 E As Integer, ByVal X As Single, ByVal Beta1 As Single, ByVal Beta2 As Si
ngle, ByVal emax As Single) As Single
    Dim L As Single
    Dim eb As Single
    Dim eu
    Dim Alpha As Single
    Dim Alpha1 As Single
    Dim Alpha2 As Single
    Dim Beta As Single
    Dim ElementNum As Integer
    Dim LeftEnd As Integer
    Dim RightEnd As Integer
    Dim LeftST As Integer
    Dim RightST As Integer
    Beta = -.3
    Alpha = .6
    Alpha1 = .1
    Alpha2 = .1
    Beta1 = -1
    Beta2 = -1
    If COR Then
     L = EleColRow(CProj, 1, ColRow, E).Length
     ElementNum = ColEle(ColRow)
     LeftEnd = Column(CProj, EleColRow(CProj, 1, ColRow, E).Start).Y_Flag
     LeftST = Column(CProj, EleColRow(CProj, 1, ColRow, E).Start).SupportT
ype
     RightEnd = Column(CProj, EleColRow(CProj, 1, ColRow, E).End).Y_Flag
     RightST = Column(CProj, EleColRow(CProj, 1, ColRow, E).End).SupportTy
pe
    ElseIf Not COR Then
     L = EleColRow(CProj, 0, ColRow, E).Length
     ElementNum = RowEle(ColRow)
     LeftEnd = Column(CProj, EleColRow(CProj, 0, ColRow, E).Start).X_Flag
     LeftST = Column(CProj, EleColRow(CProj, 0, ColRow, E).Start).SupportT
ype
     RightEnd = Column(CProj, EleColRow(CProj, 0, ColRow, E).End).X_Flag
     RightST = Column(CProj, EleColRow(CProj, 0, ColRow, E).Start).Support
Type
    End If
    'Alpha = Emax / L
    eb = emax
    If ((LeftEnd = 1) And (RightEnd = 0) And (LeftST = 0) And (RightST = 0
)) Then 'Left end hinge
     If (X >= 0 And X <= Alpha * L) Then
      eu = -((((Beta1 - Beta) * eb) * (X ^ 2)) / (Alpha * L ^ 2)) + ((2 *
(Beta1 - Beta) * eb * X) / (Alpha * L)) + Beta * eb
     ElseIf (X > Alpha * L And X <= (1 - Alpha1) * L) Then
      eu = (((1 - Beta1) * eb * X ^ 2) / ((1 - Alpha) * (1 - Alpha - Alpha
1) * (L ^ 2))) - ((2 * Alpha * (1 - Beta1) * eb * X) / ((1 - Alpha) * (1 -
 Alpha - Alpha1) * L)) + (((1 - Beta1) * Alpha ^ 2 * eb) / ((1 - Alpha) *
(1 - Alpha - Alpha1))) + (Beta1 * eb)
     ElseIf (X > (1 - Alpha1) * L And (X <= L)) Then
      eu = -(((1 - Beta1) * eb * X ^ 2) / (Alpha1 * (1 - Alpha) * (L ^ 2))
) + ((2 * (1 - Beta1) * eb * X) / (Alpha1 * (1 - Alpha) * L)) - (((1 - Bet
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 5

```
a1) * eb) / (Alpha1 * (1 - Alpha))) + eb
      End If
     ElseIf ((LeftEnd = 0) And (RightEnd = 0) And (LeftST = 0) And (RightST
 = 0)) Then 'Interior span
      X = X - L / 2
      If (X >= (-L / 2) And X <= -((1 / 2) - Alpha2) * L) Then
       eu = -((2 * (1 - Beta2) * eb * X ^ 2) / (Alpha2 * L ^ 2)) - ((2 * (1
 - Beta2) * eb * X) / (Alpha2 * L)) - (((1 - Beta2) * eb) / (2 * Alpha2))
+ eb
      ElseIf (X > -((1 / 2) - Alpha2) * L And X <= ((1 / 2) - Alpha2) * L)
Then
       eu = ((2 * (1 - Beta2) * eb * X ^ 2) / ((.5 - Alpha2) * L ^ 2)) + (B
eta2 * eb)
      ElseIf (X > ((1 / 2) - Alpha2) * L) And X <= (L / 2) Then
       eu = -((2 * (1 - Beta2) * eb * X ^ 2) / (Alpha2 * L ^ 2)) + ((2 * (1
 - Beta2) * eb * X) / (Alpha2 * L)) - (((1 - Beta2) * eb) / (2 * Alpha2))
+ eb
      End If
     ElseIf ((LeftEnd = 0) And (RightEnd = 2) And (LeftST = 0) And (RightST
 = 0)) Then   'Right end hinge
      X = L - X
      If (X >= 0 And X <= Alpha * L) Then
       eu = -((((Beta1 - Beta) * eb) * ((X) ^ 2)) / (Alpha * L ^ 2)) + ((2
* (Beta1 - Beta) * eb * (X)) / (Alpha * L))'+Beta*eb
      ElseIf (X > Alpha * L And X <= (1 - Alpha1) * L) Then
       eu = (((1 - Beta1) * eb * (X) ^ 2) / ((1 - Alpha) * (1 - Alpha - Alp
ha1) * (L ^ 2))) - ((2 * Alpha * (1 - Beta1) * eb * (X)) / ((1 - Alpha) *
(1 - Alpha - Alpha1) * L)) + (((1 - Beta1) * Alpha ^ 2 * eb) / ((1 - Alpha
) * (1 - Alpha - Alpha1))) + (Beta1 * eb)
      ElseIf (X > (1 - Alpha1) * L) And X <= L Then
       eu = -(((1 - Beta1) * eb * (X) ^ 2) / (Alpha1 * (1 - Alpha) * (L ^ 2
))) + ((2 * (1 - Beta1) * eb * (X)) / (Alpha1 * (1 - Alpha) * L)) - (((1 -
 Beta1) * eb) / (Alpha1 * (1 - Alpha))) + eb
      End If
     ElseIf ((LeftEnd = 1) And (RightEnd = 2) And (LeftST = 0) And (RightST
 = 0)) Then   '1 span Left hinge,Right hinge
      X = X - (L / 2)
      eu = ((4 * (eb) * X ^ 2) / L ^ 2) - eb
     End If
     Cal_Profile = eu
End Function

Function Cal_Wb () As Single
     Cal_Wb = .8 * DeadLoad(CProj)
End Function

Function Check_CrackMoment (f As Single, b As Single) As Single
     Dim I As Single
     Dim fcr As Single
     Dim E As Single
     Dim MCR As Single
     Dim c2 As Single
     Dim A As Single
     fcr = 1.99 * GeneralData(CProj).SFc_
     I = (1 / 12) * b * Thick(CProj) ^ 3
     E = emax(CProj)
     c2 = Thick(CProj) / 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 6

```
    A = b * Thick(CProj)
    MCR = (f * E) + ((f * I) / (A * c2)) + ((fcr * I) / c2)
    Check_CrackMoment = 1.2 * MCR
End Function

Function Check_MUlF (P As Single, b As Single, fse As Single) As Single
    Ap1 = P / (GeneralData(CProj).SFs_ * .775)' area per 1 strand
    Pp = Ap1 / (.775 * b * d(CProj))
    fps = fse + 69 + Fc_ / (100 * Pp)
    Check_MUlF = fps
End Function

Sub Design_PSlab ()
    Dim DiagIndex As Integer
    Dim RNum As Integer
    Dim WNum As Integer
    Dim Cov As Single
    Dim Lmax As Single
    Dim L As Single
    Dim Wu As Single 'Kg/m^2
    Dim b0 As Single
    Dim I As Integer
    Dim J As Integer
    Dim eb As Single      'find complete before
    Dim Drape As Single
    Dim Wb As Single
    Dim Alpha As Single
    Const Alpha1 = .1
    Const Alpha2 = .1
    Dim Beta1 As Single
    Dim Beta2 As Single
    Dim xemax As Single   'find complete before
    Const Beta = 0
    Dim Vc0 As Single
    Dim Vn0 As Single
    Dim Vc1 As Single
    Dim Vn1 As Single
    Dim Vc2 As Single
    Dim Vn2 As Single
    Dim Vc3 As Single
    Dim Vn3 As Single
    Dim de As Single

    'M design
    Dim MDL As Single
    Dim MDD As Single
    Dim MDR As Single
  ' 'M design at column strip
  ' Dim MCL As Single
  ' Dim MCD As Single
  ' Dim Mcr As Single
  Access_Data

    Trans_Element

    'Find Lmax to find primary thickness
    Lmax = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 7

```
    For I = 1 To SRow(CProj)
     If Lmax < SElement(CProj, 0, I).Length Then
      Lmax = SElement(CProj, 0, I).Length
     End If
    Next I
    For I = 1 To SCol(CProj)
     If Lmax < SElement(CProj, 1, I).Length Then
      Lmax = SElement(CProj, 1, I).Length
     End If
    Next I
    Thick(CProj) = Cal_Thick(Lmax, GeneralData(CProj).SlabIndex)
    DeadLoad(CProj) = Thick(CProj) * 2400
    frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.000")
    frmMain.txtDL.Text = Format$(Thick(CProj) * 2400, "0.00")
    Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
    'Cov = Cover(Thk) 'Flat
    If StrandIndex(CProj) = 0 Then      '0 is bond,1 is unbond
     Cov = .03 + .012
    Else
     Cov = .03 + .008
    End If


    Trans_UniformLoad Wu  'Find Load Each Element
    d(CProj) = Thick(CProj) - Cov 'Try d Initial

    ''''Display
    frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.000")
    frmMain.txtDL.Text = Format$(Thick(CProj) * 2400, "0.00")
    For I = 1 To SNodeNum(CProj)
     Cal_FDropCapital I, Thick(CProj)
    Next I

    ''''
    DiagIndex = 1    'Check Punch
    For I = 1 To SNodeNum(CProj) 'Test Shear ,Find For Slab Thickness,Chec
k For All Column
     DiagIndex = 1
     If Column(CProj, I).SupportType <> 2 Then 'If Not Free,Mean That is C
olumn support
      Select Case GeneralData(CProj).SlabIndex
      Case 0:
       de = d(CProj)
       b0 = Cal_Fb0(Column(CProj, I), 0)
       Do While Not (Check_FShear(False, Cal_FColumnShear(I, DiagIndex, de
, Wu), b0, de, I))    'If Not pass Revise Thick ,False Is Diagonal Check
        Thick(CProj) = Thick(CProj) + .01

        frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00")'Displa
y in txtThick ,Thick
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in
txtDL ,Dead Load
        Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
        d(CProj) = Thick(CProj) - Cov
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 8

```
        Loop
       Case 1:
        Cal_FDropCapital I, Thick(CProj)
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 1)'Check Drop panel
        Do While Not (Check_FShear(False, Cal_FColumnShear(I, DiagIndex, d(
CProj), Wu), b0, d(CProj), I))   'If Not pass Revise Thick ,False Is Diago
nal Check
         Thick(CProj) = Thick(CProj) + .01
         frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00")'Displa
y in txtThick ,Thick
         DeadLoad(CProj) = Thick(CProj) * 2400
         frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in
txtDL ,Dead Load
         Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
         d(CProj) = Thick(CProj) - Cov
         Cal_FDropCapital I, Thick(CProj)
         de = d(CProj)
         b0 = Cal_Fb0(Column(CProj, I), 1)'Check Drop panel
        Loop
       Case 2:
        Cal_FDropCapital I, Thick(CProj)
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 2)'Check  Capital
        Do While Not (Check_FShear(False, Cal_FColumnShear(I, DiagIndex, d(
CProj), Wu), b0, d(CProj), I))   'If Not pass Revise Thick ,False Is Diago
nal Check
         Thick(CProj) = Thick(CProj) + .01
         frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00")'Displa
y in txtThick ,Thick
         DeadLoad(CProj) = Thick(CProj) * 2400
         frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in
txtDL ,Dead Load
         Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
         d(CProj) = Thick(CProj) - Cov
         Cal_FDropCapital I, Thick(CProj)
         de = d(CProj)
         b0 = Cal_Fb0(Column(CProj, I), 2)'Check  Capital
        Loop
       Case 3:
        Cal_FDropCapital I, Thick(CProj)
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 1)'Check Drop panel
        Do While Not (Check_FShear(False, Cal_FColumnShear(I, DiagIndex, d(
CProj), Wu), b0, d(CProj), I))   'If Not pass Revise Thick ,False Is Diago
nal Check
         Thick(CProj) = Thick(CProj) + .01
         frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00") 'Displ
ay in txtThick ,Thick
         DeadLoad(CProj) = Thick(CProj) * 2400
         frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in
txtDL ,Dead Load
         Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
         d(CProj) = Thick(CProj) - Cov
         Cal_FDropCapital I, Thick(CProj)
         de = d(CProj)
         b0 = Cal_Fb0(Column(CProj, I), 1)'Check Drop panel
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 9

```
        Loop
        Cal_FDropCapital I, Thick(CProj)
        de = d(CProj) + Column(CProj, I).DropThick
        b0 = Cal_Fb0(Column(CProj, I), 2)'Check capital
        Do While Not (Check_FShear(False, Cal_FColumnShear(I, DiagIndex, d(
CProj), Wu), b0, d(CProj), I))   'If Not pass Revise Thick ,False Is Diago
nal Check
         Thick(CProj) = Thick(CProj) + .01
         frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00")'Displa
y in txtThick ,Thick
         DeadLoad(CProj) = Thick(CProj) * 2400
         frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in
txtDL ,Dead Load
         Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
         d(CProj) = Thick(CProj) - Cov
         Cal_FDropCapital I, Thick(CProj)
         de = d(CProj) + Column(CProj, I).DropThick
         b0 = Cal_Fb0(Column(CProj, I), 2)'Check capital
        Loop
       End Select
      End If
     'check diagonal in column
      DiagIndex = 2 'Check Diagonal
      If Column(CProj, I).SupportType <> 2 Then 'If Not Free,Mean That is C
olumn support
       b0 = 1
       Do While Not (Check_FShear(True, Cal_FColumnShear(I, DiagIndex, d(CP
roj), Wu), b0, d(CProj), I))   'If Not pass Revise Thick ,False Is Diagona
l Check
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00")'Display
 in txtThick ,Thick
         DeadLoad(CProj) = Thick(CProj) * 2400
         frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in
txtDL ,Dead Load
        Wu = (DeadCo * DeadLoad(CProj)) + (LiveCo * LiveLoad(CProj))
        d(CProj) = Thick(CProj) - Cov
       Loop
      End If
     Next I


     Revise_Slab
     ''''Find Thick Complete
     DeadLoad(CProj) = Thick(CProj) * 2400
     frmMain.txtThick.Text = Format$(Thick(CProj) * 100, "0.00")'Display in
txtThick ,Thick
     frmMain.txtDL.Text = Format$(DeadLoad(CProj), "0.00") 'Display in txtD
L ,Dead Load

     '''' 'Find Drop And Capital
     For I = 1 To SNodeNum(CProj)
      Cal_FDropCapital I, Thick(CProj)
     Next I
     emax(CProj) = (Thick(CProj) - 2 * (Thick(CProj) - d(CProj))) / 2

     Trans_UniformLoad Wu
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 10

```
    ''''Write Data output in File
    WNum = 0
    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
 As #20 Len = 4
     Put #20, WNum + 1, Thick(CProj)
     Put #20, WNum + 2, Cov
     WNum = WNum + 2
     '''' Puncing Shear
     DiagIndex = 1     'Check Punch
     For I = 1 To SNodeNum(CProj) 'Test Shear ,Find For Slab Thickness,Che
ck For All Column
      DiagIndex = 1'Check Punch
      If Column(CProj, I).SupportType <> 2 Then 'If Not Free,Mean That is
Column support
       Select Case GeneralData(CProj).SlabIndex
       Case 0:
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 0)
        Vc0 = Check_FFShear(False, b0, de, I) 'If Not pass Revise Thick ,F
alse Is Diagonal Check
        Vn0 = Cal_FColumnShear(I, DiagIndex, de, Wu)
        Vc1 = 0
        Vn1 = 0
        Vc2 = 0
        Vn2 = 0
       Case 1:
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 1)'Check Drop panel
        Vc0 = 0
        Vn0 = 0
        Vc1 = Check_FFShear(False, b0, d(CProj), I)
        Vn1 = Cal_FColumnShear(I, DiagIndex, d(CProj), Wu)
        Vc2 = 0
        Vn2 = 0
        Vc3 = 0
        Vn3 = 0
       Case 2:
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 2)'Check  Capital
        Vc0 = 0
        Vn0 = 0
        Vc1 = 0
        Vn1 = 0
        Vc2 = Check_FFShear(False, b0, d(CProj), I)
        Vn2 = Cal_FColumnShear(I, DiagIndex, d(CProj), Wu)
        Vc3 = 0
        Vn3 = 0
       Case 3:
        Vc0 = 0
        Vn0 = 0
        de = d(CProj)
        b0 = Cal_Fb0(Column(CProj, I), 1)'Check Drop panel
        Vc1 = Check_FFShear(False, b0, d(CProj), I)
        Vn1 = Cal_FColumnShear(I, DiagIndex, d(CProj), Wu)
        de = d(CProj) + Column(CProj, I).DropThick
        b0 = Cal_Fb0(Column(CProj, I), 2)'Check capital
        Vc2 = Check_FFShear(False, b0, d(CProj), I)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 11

```
        Vn2 = Cal_FColumnShear(I, DiagIndex, d(CProj), Wu)
       End Select
      End If
      'check diagonal in column
      DiagIndex = 2 'Check Diagonal
      If Column(CProj, I).SupportType <> 2 Then 'If Not Free,Mean That is
Column support
       b0 = 1
       Vc3 = (Check_FFShear(True, b0, d(CProj), I))   'If Not pass Revise
Thick ,False Is Diagonal Check
       Vn3 = Cal_FColumnShear(I, DiagIndex, d(CProj), Wu)
      End If
      Put #20, WNum + 1, Vc0
      Put #20, WNum + 2, Vn0
      Put #20, WNum + 3, Vc1
      Put #20, WNum + 4, Vn1
      Put #20, WNum + 5, Vc2
      Put #20, WNum + 6, Vn2
      Put #20, WNum + 7, Vc3
      Put #20, WNum + 8, Vn3
      WNum = WNum + 8
     Next I


    '''******* End Moment at Ultimate load 3 case *******
     '''' Write end moment & mid span moment and transfer to design moment
 at edge of column
     For I = 1 To SRow(CProj)
      Trans_BElement False, I  'False is calculate in x axis
      Cal_FEndMoment False, I
      For J = 1 To RowEle(I)
       Put #20, WNum + 1, ML(CProj, J)
       Put #20, WNum + 2, MD(CProj, J)
       Put #20, WNum + 3, MR(CProj, J)
       WNum = WNum + 3
      Next J
     Next I
     For I = 1 To SCol(CProj)
      Trans_BElement True, I   'False is calculate in x axis
      Cal_FEndMoment True, I
      For J = 1 To ColEle(I)
       Put #20, WNum + 1, ML(CProj, J)
       Put #20, WNum + 2, MD(CProj, J)
       Put #20, WNum + 3, MR(CProj, J)
       WNum = WNum + 3
      Next J
     Next I

    '''******* End Moment Elastic at working deadload *******
     '''' Write end moment & mid span moment and transfer to design moment
at edge of column
     For I = 1 To SRow(CProj)
      Trans_BElement False, I  'False is calculate in x axis
      Cal_ElasticEndMoment False, I
      For J = 1 To RowEle(I)
       Put #20, WNum + 1, ML(CProj, J)
       Put #20, WNum + 2, MD(CProj, J)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 12

```
        Put #20, WNum + 3, MR(CProj, J)
        WNum = WNum + 3
       Next J
      Next I
      For I = 1 To SCol(CProj)
       Trans_BElement True, I   'False is calculate in x axis
       Cal_ElasticEndMoment True, I
       For J = 1 To ColEle(I)
        Put #20, WNum + 1, ML(CProj, J)
        Put #20, WNum + 2, MD(CProj, J)
        Put #20, WNum + 3, MR(CProj, J)
        WNum = WNum + 3
       Next J
      Next I

    '''****** End Moment service at working all Load  *******
     '''' Write end moment & mid span moment and transfer to design moment
at edge of column
     For I = 1 To SRow(CProj)
      Trans_BElement False, I  'False is calculate in x axis
      Cal_ElasticEndMoment False, I
      For J = 1 To RowEle(I)
       Put #20, WNum + 1, ML(CProj, J)
       Put #20, WNum + 2, MD(CProj, J)
       Put #20, WNum + 3, MR(CProj, J)
       WNum = WNum + 3
      Next J
     Next I
     For I = 1 To SCol(CProj)
      Trans_BElement True, I   'False is calculate in x axis
      Cal_ElasticEndMoment True, I
      For J = 1 To ColEle(I)
       Put #20, WNum + 1, ML(CProj, J)
       Put #20, WNum + 2, MD(CProj, J)
       Put #20, WNum + 3, MR(CProj, J)
       WNum = WNum + 3
      Next J
     Next I


    '''****** End Moment Ultimate at Ultimate all Load  *******
     '''' Write end moment & mid span moment and transfer to design moment
at edge of column
     For I = 1 To SRow(CProj)
      Trans_BElement False, I  'False is calculate in x axis
      Cal_ElasticEndMoment False, I
      For J = 1 To RowEle(I)
       Put #20, WNum + 1, ML(CProj, J)
       Put #20, WNum + 2, MD(CProj, J)
       Put #20, WNum + 3, MR(CProj, J)
       WNum = WNum + 3
      Next J
     Next I
     For I = 1 To SCol(CProj)
      Trans_BElement True, I   'False is calculate in x axis
      Cal_ElasticEndMoment True, I
      For J = 1 To ColEle(I)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 13

```
        Put #20, WNum + 1, ML(CProj, J)
        Put #20, WNum + 2, MD(CProj, J)
        Put #20, WNum + 3, MR(CProj, J)
        WNum = WNum + 3
      Next J
     Next I
    Close #20
    Initial_Menu 3
End Sub

Function Find_AsStrand (L As Single, b As Single) As Integer 'Number of st
rand
    Dim Ps As Single
    Dim P As Single
    Dim Wb As Single
    Dim A As Single
    A = Thick(CProj) - (Thick(CProj) - d(CProj)) * 2
    Ps = .775 * GeneralData(CProj).SFs_
    Wb = .8 * DeadLoad(CProj) * b
    P = (Wb * L ^ 2) / (8 * A)
    Find_AsStrand = (-1) * Int((P / Ps) * (-1))
End Function

Sub Find_AsUl (b As Single, M As Single, TR As Single, BR As Single)
     Const Phi = .9 'For Moment
    Dim pmin As Single
    Dim pb As Single
    Dim P As Single
    Dim Pc As Single  'for compress steel
    Dim pt As Single
    Dim Asr As Single 'area of balance steel
    Dim at As Single  'area of tension steel
    Dim Ac As Single  'area of compress steel
    Dim Rn As Single
    Dim Mn As Single
    Dim Fc_ As Single
    Dim Fy As Single
    Dim fs_ As Single
    Dim T_ As Single
    Dim Wu As Single
    Dim Mu As Single
    Dim Mc As Single
    Dim Cov As Single
    Dim B1 As Single
    If b = 0 Then
     TR = 0
     BR = 0
     Exit Sub
    End If

    '''' Init Data
    T_ = d(CProj)
    Mu = Abs(M)
    Cov = Thick(CProj) - d(CProj)
    Fc_ = GeneralData(CProj).SFc_
    If Fc_ < 280 Then
     B1 = .85
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 14

```
    ElseIf ((Fc_ >= 280) And (Fc_ <= 560)) Then
     B1 = .85 - ((.05 * (Fc_ - 280)) / 70)
    ElseIf Fc_ > 560 Then
     B1 = .65
    End If
    Fy = GeneralData(CProj).RFs_
    pmin = 14 / Fy
    pb = (.85 * B1 * Fc_ * 6117) / (Fy * (6117 + Fy))
    P = .75 * pb
    Rn = P * Fy * (1 - ((.5 * P * Fy) / (.85 * Fc_)))
    Mn = Rn * b * T_ ^ 2
    Asr = b * T_ * P
    If Mu < Phi * Mn Then
     Ra = Mu / (Phi * b * T_ ^ 2)  'in kg/m^2
     pt = (.85 * Fc_ / Fy) * (1 - Sqr(1 - ((2 * Ra) / (.85 * Fc_))))'in m^
2
     If pt < pmin Then
      pt = pmin
     End If
     at = b * T_ * pt
     Ac = 0
    ElseIf Mu > Phi * Mn Then
     Mc = Mu - (Phi * Mn)
     Ac = Mc / (Phi * Fy * (T_ - Cov))
     at = Asr + Ac
     Pc = Ac / (b * T_)
     If (P - Pc) < ((.85 * B1 * Fc_ * Cov * 6117) / (Fy * T_ * (6117 - Fy)
)) Then
      fs_ = 6117 - (((6117 + Fy) * Cov) / T_)
      Ac = (Mc / (Phi * fs_ * (T_ - Cov)))
      at = Asr + Ac
     End If
    End If
    BR = at
    TR = Ac
    If M < 0 Then
     Swap BR, TR
    End If
End Sub


Sub Find_Stress (P As Single, M As Single, E As Single, b As Single, Top A
s Single, Bottom As Single)
    I = (1 / 12) * b * Thick(CProj) ^ 3
    c = Thick(CProj) / 2
    If E < 0 Then
     E = E - c + (Thick(CProj) - d(CProj))
    ElseIf E >= 0 Then
     E = E + c - (Thick(CProj) - d(CProj))
    End If
    Top = (P / (b * Thick(CProj))) + ((M * c) / I) + ((P * E * c) / I)
    Bottom = (P / (b * Thick(CProj))) - ((M * c) / I) - ((P * E * c) / I)
End Sub

Sub Revise_Slab ()
    Dim I As Integer
    Dim J As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 15

```
    Dim Wu As Single
'''''' find end moment to display design moment in graphics
    '''******* End Moment at Ultimate load 3 case *******
     '''' Write end moment & mid span moment and transfer to design moment
 at edge of column
     For I = 1 To SRow(CProj)
      Trans_BElement False, I  'False is calculate in x axis
      Cal_FEndMoment False, I
     Next I
     For I = 1 To SCol(CProj)
      Trans_BElement True, I   'False is calculate in x axis
      Cal_FEndMoment True, I
     Next I

   Close #20
   '''''''''check stress
   Dim Lmax As Single
   Dim XMax As Integer
   Dim XMin As Integer
   Dim YMax As Integer
   Dim YMin As Integer
   Dim InitX As Integer
   Dim InitY As Integer
   Dim Col As Integer
   Dim CStart As Integer
   Dim CEnd As Integer
   Dim TPMX As Single
   Dim TPMY As Single
   Dim TPP As Single
   Dim TPM As Single
   Dim AddSect As Single
   Dim X1 As Single
   Dim Y1 As Single
   Dim X2 As Single
   Dim Y2 As Single
   Dim X As Single
   Dim Y As Single
   'Dim TextNo As Integer


   Dim Init1 As Integer
   Dim Init2 As Integer
   Dim PF As Single
   Dim M As Single
   Dim ColumnColor As Integer
   Dim ElementColor As Integer
   Dim ProfileColor As Integer
   Dim Bbeam As Single
   Dim RNum1 As Integer
   Dim RNum2 As Integer
   Dim RNum3 As Integer

   Dim TextNo As Integer
   ReDim Text(100) As String
   TextNo = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 16

```
    Printer.FontName = "Arial"
    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
As #20 Len = 4

    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    'frmOutGraph(CProj).Cls
    Col = 50      'Column Width in Twips
    AddSect = .5 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    'TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m =
TPMX Twips for X axis
    'TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
 TPMY Twips for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips       '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01

    'Dim Text As String
    Dim Top As Single
    Dim Bottom As Single
    Dim P As Single
    CR = Chr$(13)
    LF = Chr$(10)
    TB = Chr$(9)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    '''' Draw Profile in X axis
    Init1 = InitX
    Init2 = InitY
    Dim Mu As Single
    Dim Mn As Single
    Dim Ms As Single
    Dim AllTEnsion As Single
    Dim AllCOmpress As Single
    Dim CheckStress As Single
    AllCOmpress = .75 * GeneralData(CProj).SFc_
    AllTEnsion = 2.65 * Sqr(GeneralData(CProj).SFc_)

Revise5:
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_FEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Select Case Top / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Top / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))
        GoTo Revise5
       End If
       Select Case Bottom / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Bottom / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))

        GoTo Revise5
      End If
     Next X
    Next J
   Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 18

```
Revise6:    '''' Draw Profile in Y axis
    For I = 1 To SCol(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_FEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     For J = 1 To ColEle(I)
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Select Case Top / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Top / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))
        GoTo Revise6
       End If
       Select Case Bottom / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Bottom / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))

        GoTo Revise6
       End If
      Next Y
     Next J
    Next I
    Close #20


    ''''end moment
    Dim MWL As Single
    Dim MWR As Single
    Dim MSL As Single
    Dim MSR As Single
    Dim MUL As Single
    Dim MUR As Single

    CR = Chr$(13)
    LF = Chr$(10)
    Const ucm = " cm"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 19

```
    Const um = " m"
    Const ukg = " kg"
    Const uksc = " ksc"
    Const uksm = " kg/m^2"
    Dim CNo As Integer

    AddSect = .5

    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
 As #20 Len = 4

    Wu = DeadLoad(CProj)
    AllCOmpress = .75 * GeneralData(CProj).SFc_
    AllTEnsion = 2.65 * Sqr(GeneralData(CProj).SFc_)

    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
Revise1:
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ElasticEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Select Case Top / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Top / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))
        GoTo Revise1
       End If
       Select Case Bottom / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Bottom / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 20

```
        GoTo Revise1
       End If
      Next X
     Next J
    Next I
    'Text2 flexeral stress in Column
Revise2:
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_ElasticEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     For J = 1 To ColEle(I)
      For X = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 1, I, J).L2 / 2, Top, Bottom
       Select Case Top / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Top / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))

        GoTo Revise2
       End If
       Select Case Bottom / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Bottom / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))

        GoTo Revise2
      End If
     Next X
    Next J
   Next I
   AddSect = .5

   '**************************'
   ''''check at SURVICE load
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PCOMPUTE.BAS - 21

```
    '**************************'
    AllCOmpress = .75 * GeneralData(CProj).SFc_
    AllTEnsion = 2.65 * Sqr(GeneralData(CProj).SFc_)

Revise3:
    Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    'Text3 flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ServiceEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Select Case Top / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Top / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))

        GoTo Revise3
       End If
       Select Case Bottom / 10000
       Case Is < 0
        CheckStress = AllTEnsion
       Case Is >= 0
        CheckStress = AllCOmpress
       End Select
       If Abs(Bottom / 10000) > CheckStress Then
        Thick(CProj) = Thick(CProj) + .01
        frmMain.txtThick.Text = Format$(Thick(CProj) * 100)
        DeadLoad(CProj) = Thick(CProj) * 2400
        frmMain.txtDL.Text = Format$(DeadLoad(CProj))

        GoTo Revise3
       End If
      Next X
     Next J
    Next I
    'Text3 flexeral stress in Column
Revise4:
    Lmax = 0
    Bbeam = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Cal_ElasticEndMoment (ColOrRow As Integer, ColRow As Integer) 'deadloa
d tranfer
    Dim Pattern As Integer
    Dim axis As Integer
    Const ErrorCheck = 1
    Dim ErrorValue As Single
    Dim I As Integer
    If ColOrRow Then
     axis = 0
    ElseIf Not ColOrRow Then
     axis = 1
    End If
    ReDim FEML(ElementNum(CProj)) As Single
    ReDim FEMR(ElementNum(CProj)) As Single
    ReDim DFL(ElementNum(CProj)) As Single
    ReDim DFR(ElementNum(CProj)) As Single
    ReDim COML(ElementNum(CProj)) As Single
    ReDim COMR(ElementNum(CProj))  As Single
    ReDim DML(ElementNum(CProj)) As Single
    ReDim DMR(ElementNum(CProj)) As Single
    ReDim MNode(NodeNum(CProj)) As Single
    ReDim MMaxL(ElementNum(CProj)) As Single
    ReDim MMaxR(ElementNum(CProj)) As Single
    For I = 1 To ElementNum(CProj)
     MMaxL(I) = 0
     MMaxR(I) = 0
   Next I
   Pattern = 0
   For I = 1 To ElementNum(CProj)
    R = Cal_FK(I, Not (ColOrRow))
    FEML(I) = Cal_FFEM(axis, I, True, Pattern)
    FEMR(I) = Cal_FFEM(axis, I, False, Pattern)
    DFL(I) = Cal_FDF(I, Not (ColOrRow), True)
    DFR(I) = Cal_FDF(I, Not (ColOrRow), False)
    COFL(CProj, I) = COFL(CProj, I)
    COFR(CProj, I) = COFR(CProj, I)
    ML(CProj, I) = FEML(I)
    MR(CProj, I) = FEMR(I)
    DML(I) = 0
    DMR(I) = 0
    COML(I) = 0
    COMR(I) = 0
  Next I
  'Distribution Moment
  Do
  ErrorValue = 0
    'Sum DM
   For I = 1 To NodeNum(CProj)
    MNode(I) = 0
    For J = 1 To ElementNum(CProj)
     If (Element(CProj, J).Start) = I Then
      MNode(I) = (MNode(I) + ML(CProj, J))
     ElseIf (Element(CProj, J).End = I) Then
      MNode(I) = (MNode(I) + MR(CProj, J))
     End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 2

```
      Next J
     Next I
      'DM
     For I = 1 To NodeNum(CProj)
      For J = 1 To ElementNum(CProj)
       If (Element(CProj, J).Start = I) Then
        DML(J) = (((-1) * MNode(I)) * DFL(J))
       ElseIf (Element(CProj, J).End = I) Then
        DMR(J) = (((-1) * MNode(I)) * DFR(J))
       End If
      Next J
     Next I
     For I = 1 To ElementNum(CProj)
      If Abs(ErrorValue) < Abs(DML(I)) Then
       ErrorValue = DML(I)
      End If
      If Abs(ErrorValue) < Abs(DMR(I)) Then
       ErrorValue = DMR(I)
      End If
     Next I
     'Carry over moment
     For I = 1 To ElementNum(CProj)
      COML(I) = DMR(I) * COFL(CProj, I)
      COMR(I) = DML(I) * COFR(CProj, I)
     Next I
      'Sum Moment
     If (Abs(ErrorValue) < ErrorCheck) Then
      For I = 1 To ElementNum(CProj)
       ML(CProj, I) = ML(CProj, I) + DML(I)
       MR(CProj, I) = MR(CProj, I) + DMR(I)
      Next I
     Else
      For I = 1 To ElementNum(CProj)
       ML(CProj, I) = ML(CProj, I) + DML(I) + COML(I)
       MR(CProj, I) = MR(CProj, I) + DMR(I) + COMR(I)
      Next I
     End If
    Loop Until (Abs(ErrorValue) < ErrorCheck)
    For I = 1 To ElementNum(CProj)
     If Abs(MMaxL(I)) < Abs(ML(CProj, I)) Then
      MMaxL(I) = ML(CProj, I)
     End If
     If Abs(MMaxR(I)) < Abs(MR(CProj, I)) Then
      MMaxR(I) = MR(CProj, I)
     End If
     MD(CProj, I) = (1 / 8) * ((DeadCo * DeadLoad(CProj)) + (1.8 * LiveLoa
d(CProj))) * ((Element(CProj, I).L1 + Element(CProj, I).L2) / 2) * (Elemen
t(CProj, I).Length) ^ 2
    Next I
   For I = 1 To ElementNum(CProj)
    ML(CProj, I) = MMaxL(I)
    MR(CProj, I) = MMaxR(I)
    MD(CProj, I) = MD(CProj, I) - (ML(CProj, I) + MR(CProj, I)) / 2
   Next I

    '''' Show End Moment at Programmed
    'Load frmOutText(CProj)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 3

```
    'frmOutText(CProj).Show
    'frmOutText(CProj).Cls
    'For I = 1 To ElementNum(CProj)
    ' frmOutText(CProj).Print "Fixed end moment Left element" + Format$(I)
 + " = " + Format$(FEML(I), "0.00")
    ' frmOutText(CProj).Print "Fixed end moment Right element" + Format$(I
) + " = " + Format$(FEMR(I), "0.00")
    ' frmOutText(CProj).Print "DF Factor Left " + Format$(DFL(I), "0.000")
    ' frmOutText(CProj).Print "DF Factor Right " + Format$(DFR(I), "0.000"
)
    ' frmOutText(CProj).Print "End Moment Left " + Format$(ML(CProj, I), "
0.00")
    ' frmOutText(CProj).Print "End Moment Right " + Format$(MR(CProj, I),
"0.00")
    'Next I
End Sub

Sub Cal_ServiceEndMoment (ColOrRow As Integer, ColRow As Integer) 'at serv
ice load all load with no factor
    Dim Pattern As Integer
    Dim axis As Integer
    Const ErrorCheck = 1
    Dim ErrorValue As Single
    Dim I As Integer
    If ColOrRow Then
     axis = 0
    ElseIf Not ColOrRow Then
     axis = 1
    End If
    ReDim FEML(ElementNum(CProj)) As Single
    ReDim FEMR(ElementNum(CProj)) As Single
    ReDim DFL(ElementNum(CProj)) As Single
    ReDim DFR(ElementNum(CProj)) As Single
    ReDim COML(ElementNum(CProj)) As Single
    ReDim COMR(ElementNum(CProj)) As Single
    ReDim DML(ElementNum(CProj)) As Single
    ReDim DMR(ElementNum(CProj)) As Single
    ReDim MNode(NodeNum(CProj)) As Single
    ReDim MMaxL(ElementNum(CProj)) As Single
    ReDim MMaxR(ElementNum(CProj)) As Single
    For I = 1 To ElementNum(CProj)
     MMaxL(I) = 0
     MMaxR(I) = 0
    Next I
    Pattern = 5
    For I = 1 To ElementNum(CProj)
     R = Cal_FK(I, Not (ColOrRow))
     FEML(I) = Cal_FFEM(axis, I, True, Pattern)
     FEMR(I) = Cal_FFEM(axis, I, False, Pattern)
     DFL(I) = Cal_FDF(I, Not (ColOrRow), True)
     DFR(I) = Cal_FDF(I, Not (ColOrRow), False)
     COFL(CProj, I) = COFL(CProj, I)
     COFR(CProj, I) = COFR(CProj, I)
     ML(CProj, I) = FEML(I)
     MR(CProj, I) = FEMR(I)
     DML(I) = 0
     DMR(I) = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 4

```
     COML(I) = 0
     COMR(I) = 0
    Next I
    'Distribution Moment
    Do
    ErrorValue = 0
      'Sum DM
     For I = 1 To NodeNum(CProj)
      MNode(I) = 0
      For J = 1 To ElementNum(CProj)
       If (Element(CProj, J).Start) = I Then
        MNode(I) = (MNode(I) + ML(CProj, J))
       ElseIf (Element(CProj, J).End = I) Then
        MNode(I) = (MNode(I) + MR(CProj, J))
       End If
      Next J
     Next I
      'DM
     For I = 1 To NodeNum(CProj)
      For J = 1 To ElementNum(CProj)
       If (Element(CProj, J).Start = I) Then
        DML(J) = (((-1) * MNode(I)) * DFL(J))
       ElseIf (Element(CProj, J).End = I) Then
        DMR(J) = (((-1) * MNode(I)) * DFR(J))
       End If
      Next J
     Next I
     For I = 1 To ElementNum(CProj)
      If Abs(ErrorValue) < Abs(DML(I)) Then
       ErrorValue = DML(I)
      End If
      If Abs(ErrorValue) < Abs(DMR(I)) Then
       ErrorValue = DMR(I)
      End If
     Next I
     'Carry over moment
     For I = 1 To ElementNum(CProj)
      COML(I) = DMR(I) * COFL(CProj, I)
      COMR(I) = DML(I) * COFR(CProj, I)
     Next I
      'Sum Moment
     If (Abs(ErrorValue) < ErrorCheck) Then
      For I = 1 To ElementNum(CProj)
       ML(CProj, I) = ML(CProj, I) + DML(I)
       MR(CProj, I) = MR(CProj, I) + DMR(I)
      Next I
     Else
      For I = 1 To ElementNum(CProj)
       ML(CProj, I) = ML(CProj, I) + DML(I) + COML(I)
       MR(CProj, I) = MR(CProj, I) + DMR(I) + COMR(I)
      Next I
     End If
    Loop Until (Abs(ErrorValue) < ErrorCheck)
    For I = 1 To ElementNum(CProj)
     If Abs(MMaxL(I)) < Abs(ML(CProj, I)) Then
      MMaxL(I) = ML(CProj, I)
     End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 5

```
      If Abs(MMaxR(I)) < Abs(MR(CProj, I)) Then
       MMaxR(I) = MR(CProj, I)
      End If
      MD(CProj, I) = (1 / 8) * ((DeadCo * DeadLoad(CProj)) + (1.8 * LiveLoa
d(CProj))) * ((Element(CProj, I).L1 + Element(CProj, I).L2) / 2) * (Elemen
t(CProj, I).Length) ^ 2
     Next I
    For I = 1 To ElementNum(CProj)
     ML(CProj, I) = MMaxL(I)
     MR(CProj, I) = MMaxR(I)
     MD(CProj, I) = MD(CProj, I) - (ML(CProj, I) + MR(CProj, I)) / 2
    Next I
End Sub

Sub Cal_UltimateEndMoment (ColOrRow As Integer, ColRow As Integer) 'Ultima
te all load with factor
    Dim Pattern As Integer
    Dim axis As Integer
    Const ErrorCheck = 1
    Dim ErrorValue As Single
    Dim I As Integer
    If ColOrRow Then
     axis = 0
    ElseIf Not ColOrRow Then
     axis = 1
    End If
    ReDim FEML(ElementNum(CProj)) As Single
    ReDim FEMR(ElementNum(CProj)) As Single
    ReDim DFL(ElementNum(CProj)) As Single
    ReDim DFR(ElementNum(CProj)) As Single
    ReDim COML(ElementNum(CProj)) As Single
    ReDim COMR(ElementNum(CProj)) As Single
    ReDim DML(ElementNum(CProj)) As Single
    ReDim DMR(ElementNum(CProj)) As Single
    ReDim MNode(NodeNum(CProj)) As Single
    ReDim MMaxL(ElementNum(CProj)) As Single
    ReDim MMaxR(ElementNum(CProj)) As Single
    For I = 1 To ElementNum(CProj)
     MMaxL(I) = 0
     MMaxR(I) = 0
    Next I
    Pattern = 4
    For I = 1 To ElementNum(CProj)
     R = Cal_FK(I, Not (ColOrRow))
     FEML(I) = Cal_FFEM(axis, I, True, Pattern)
     FEMR(I) = Cal_FFEM(axis, I, False, Pattern)
     DFL(I) = Cal_FDF(I, Not (ColOrRow), True)
     DFR(I) = Cal_FDF(I, Not (ColOrRow), False)
     COFL(CProj, I) = COFL(CProj, I)
     COFR(CProj, I) = COFR(CProj, I)
     ML(CProj, I) = FEML(I)
     MR(CProj, I) = FEMR(I)
     DML(I) = 0
     DMR(I) = 0
     COML(I) = 0
     COMR(I) = 0
    Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 6

```
'Distribution Moment
Do
ErrorValue = 0
  'Sum DM
 For I = 1 To NodeNum(CProj)
  MNode(I) = 0
  For J = 1 To ElementNum(CProj)
   If (Element(CProj, J).Start) = I Then
    MNode(I) = (MNode(I) + ML(CProj, J))
   ElseIf (Element(CProj, J).End = I) Then
    MNode(I) = (MNode(I) + MR(CProj, J))
   End If
  Next J
 Next I
  'DM
 For I = 1 To NodeNum(CProj)
  For J = 1 To ElementNum(CProj)
   If (Element(CProj, J).Start = I) Then
    DML(J) = (((-1) * MNode(I)) * DFL(J))
   ElseIf (Element(CProj, J).End = I) Then
    DMR(J) = (((-1) * MNode(I)) * DFR(J))
   End If
  Next J
 Next I
 For I = 1 To ElementNum(CProj)
  If Abs(ErrorValue) < Abs(DML(I)) Then
   ErrorValue = DML(I)
  End If
  If Abs(ErrorValue) < Abs(DMR(I)) Then
   ErrorValue = DMR(I)
  End If
 Next I
 'Carry over moment
 For I = 1 To ElementNum(CProj)
  COML(I) = DMR(I) * COFL(CProj, I)
  COMR(I) = DML(I) * COFR(CProj, I)
 Next I
  'Sum Moment
 If (Abs(ErrorValue) < ErrorCheck) Then
  For I = 1 To ElementNum(CProj)
   ML(CProj, I) = ML(CProj, I) + DML(I)
   MR(CProj, I) = MR(CProj, I) + DMR(I)
  Next I
 Else
  For I = 1 To ElementNum(CProj)
   ML(CProj, I) = ML(CProj, I) + DML(I) + COML(I)
   MR(CProj, I) = MR(CProj, I) + DMR(I) + COMR(I)
  Next I
 End If
Loop Until (Abs(ErrorValue) < ErrorCheck)
For I = 1 To ElementNum(CProj)
 If Abs(MMaxL(I)) < Abs(ML(CProj, I)) Then
  MMaxL(I) = ML(CProj, I)
 End If
 If Abs(MMaxR(I)) < Abs(MR(CProj, I)) Then
  MMaxR(I) = MR(CProj, I)
 End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 7

```
     MD(CProj, I) = (1 / 8) * ((DeadCo * DeadLoad(CProj)) + (1.8 * LiveLoa
d(CProj))) * ((Element(CProj, I).L1 + Element(CProj, I).L2) / 2) * (Elemen
t(CProj, I).Length) ^ 2
    Next I
   For I = 1 To ElementNum(CProj)
    ML(CProj, I) = MMaxL(I)
    MR(CProj, I) = MMaxR(I)
    'MD(CProj, I) = MD(CProj, I) - (ML(CProj, I) + MR(CProj, I)) / 2 'try
   Next I
End Sub

Function Check_Deflection (COR As Integer, LNo As Integer) As Single
    Dim L As Single
    Dim La As Single
    If COR Then
     L = EleColRow(CProj, 1, LNo, 1).Length / 360
     For I = 1 To ColEle(LNo)
      La = EleColRow(CProj, 1, LNo, I).Length / 360
      If L < La Then
       L = La
      End If
     Next I
    ElseIf Not (COR) Then
     L = EleColRow(CProj, 0, LNo, 1).Length / 360
     For I = 1 To ColEle(LNo)
      La = EleColRow(CProj, 0, LNo, I).Length / 360
      If L < La Then
       L = La
      End If
     Next I
    End If
    Check_Deflection = L
End Function

Function Check_MnMax (s As Integer, b As Single) As Single
    Dim a As Single
    Dim Mn As Single
    Dim ss As Single
    Dim ds As Single
    ds = (Thick(CProj) - .025) * 100    'cm
    ss = .004 * b * (Thick(CProj) / 2) * 10000 'cm^2
    a = ((s * GeneralData(CProj).SFs_ * .775) + (GeneralData(CProj).RFs_ *
 ss)) / (.85 * b * 100 * GeneralData(CProj).SFc_) 'cm
    Mn = (s * GeneralData(CProj).SFs_ * .775 * (100 * Thick(CProj) - 2 * (
Thick(CProj) * 100 - 100 * d(CProj)) - (a / 2))) + (ss * GeneralData(CProj
).RFs_ * (ds - (a / 2)))'kg-m
    Check_MnMax = (.9 * Mn) / 100
End Function

Sub Find_BL (COR As Integer, ColRow As Integer, Lmax As Single, Bbeam As S
ingle)
    Lmax = 0
    Bbeam = 0
    If COR Then
     For I = 1 To ColEle(ColRow)
      If Lmax < EleColRow(CProj, 1, ColRow, I).Length Then
       Lmax = EleColRow(CProj, 1, ColRow, I).Length
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 8

```
      End If
      If Bbeam < EleColRow(CProj, 1, ColRow, I).L1 / 2 + EleColRow(CProj,
1, ColRow, I).L2 / 2 Then
       Bbeam = EleColRow(CProj, 1, ColRow, I).L1 / 2 + EleColRow(CProj, 1,
 ColRow, I).L2 / 2
      End If
     Next I
    ElseIf Not (COR) Then
     For I = 1 To RowEle(ColRow)
      If Lmax < EleColRow(CProj, 0, ColRow, I).Length Then
       Lmax = EleColRow(CProj, 0, ColRow, I).Length
      End If
      If Bbeam < EleColRow(CProj, 0, ColRow, I).L1 / 2 + EleColRow(CProj,
0, ColRow, I).L2 / 2 Then
       Bbeam = EleColRow(CProj, 0, ColRow, I).L1 / 2 + EleColRow(CProj, 0,
 ColRow, I).L2 / 2
      End If
     Next I
    End If
End Sub

Function Find_MaxDeflection (COR As Integer, LNo As Integer, EIndex As Sin
gle) As Single
    Dim W As Single
    Dim Def As Single
    Dim DefT As Single
    Dim L As Single
    Dim b As Single
    Dim Ini As Single
    Def = 0
    If COR Then
     For I = 1 To ColEle(LNo)
      b = (EleColRow(CProj, 1, LNo, I).L1 / 2) + (EleColRow(CProj, 1, LNo,
I).L2 / 2)
      L = EleColRow(CProj, 1, LNo, I).Length
      If EIndex = 1 Then 'after tension
       W = (DeadLoad(CProj) + SuperLoad(CProj)) * b
      ElseIf EIndex = 2 Then 'during use
       W = (LiveLoad(CProj) + SuperLoad(CProj)) * b
      End If
      Ini = (1 / 12) * b * Thick(CProj) ^ 3
      If I = 1 Or I = ColEle(LNo) Then
       DefT = (3.8 * W * L ^ 4) / (384 * GeneralData(CProj).SEc * Ini)
      Else
       DefT = (2.6 * W * L ^ 4) / (384 * GeneralData(CProj).SEc * Ini)
      End If
      If Def < DefT Then
       Def = DefT
      End If
     Next I
    ElseIf Not COR Then
     For I = 1 To RowEle(LNo)
      b = (EleColRow(CProj, 0, LNo, I).L1 / 2) + (EleColRow(CProj, 0, LNo,
I).L2 / 2)
      L = EleColRow(CProj, 0, LNo, I).Length
      If EIndex = 1 Then 'after tension
       W = (DeadLoad(CProj) + SuperLoad(CProj)) * b
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 9

```
        ElseIf EIndex = 2 Then 'during use
         W = (LiveLoad(CProj) + SuperLoad(CProj)) * b
        End If
        Ini = (1 / 12) * b * Thick(CProj) ^ 3
        If I = 1 Or I = RowEle(LNo) Then
         DefT = (3.8 * W * L ^ 4) / (384 * GeneralData(CProj).SEc * Ini)
        Else
         DefT = (2.6 * W * L ^ 4) / (384 * GeneralData(CProj).SEc * Ini)
        End If
        If Def < DefT Then
         Def = DefT
        End If
       Next I
      End If
      Find_MaxDeflection = Def / 100000'unit in meter
End Function


Function Find_MuMax (COR As Integer, LNo As Integer) As Single
      Dim MMax As Single
      If COR Then
       MMax = Abs(ML(CProj, 1))
       For I = 1 To ColEle(LNo)
        If MMax < Abs(ML(CProj, I)) Then
         MMax = Abs(ML(CProj, I))
        End If
        If MMax < Abs(MR(CProj, I)) Then
         MMax = Abs(MR(CProj, I))
        End If
        If MMax < Abs(MD(CProj, I)) Then
         MMax = Abs(MD(CProj, I))
        End If
       Next I
      ElseIf Not COR Then
       MMax = Abs(ML(CProj, 1))
       For I = 1 To RowEle(LNo)
        If MMax < Abs(ML(CProj, I)) Then
         MMax = Abs(ML(CProj, I))
        End If
        If MMax < Abs(MR(CProj, I)) Then
         MMax = Abs(MR(CProj, I))
        End If
        If MMax < Abs(MD(CProj, I)) Then
         MMax = Abs(MD(CProj, I))
        End If
       Next I
      End If
      Find_MuMax = Abs(MMax)
End Function

Sub Graph_BMD (AddSect As Single)
      Dim SumL As Single
      Dim MMax As Single
      Dim MMin As Single
      Dim StartPoint As Single
      Dim EndPoint As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 10

```
    Dim Mx As Single
    Dim I As Integer
    Dim J As Single
    Dim LineColor As Integer
    Dim TwipsPerMeter As Integer
    LineColor = 14
    'find best dimension
    SumL = 0
    For I = 1 To ElementNum(CProj)
     SumL = SumL + Element(CProj, I).Length
    Next I
    TwipsPerMeter = (frmOutGraph(CProj).ScaleWidth - 1500) / SumL
    'Find M Max
    MMax = 0
    MMin = 0
    For I = 1 To ElementNum(CProj)
     For J = 0 To Element(CProj, I).Length Step AddSect
      Mx = Cal_BMoment(I, J, ML(CProj, I), MR(CProj, I))
      If Mx > MMax Then MMax = Mx
      If Mx < MMin Then MMin = Mx
     Next J
    Next I
    'find best amplitude
    TwipsPerMoment = ((frmOutGraph(CProj).ScaleHeight / 2) - 1500) / (MMax
 - MMin)
    frmOutGraph(CProj).PSet (frmOutGraph(CProj).lneMxX.X1, frmOutGraph(CPr
oj).lneMxX.Y1), Colors(LineColor)
    For I = 1 To ElementNum(CProj)
     For J = 0 To Element(CProj, I).Length Step AddSect
      Mx = Cal_BMoment(I, J, ML(CProj, I), MR(CProj, I)) * TwipsPerMoment
      frmOutGraph(CProj).Line -(frmOutGraph(CProj).lneMxX.X1 + (((Node(CPr
oj, Element(CProj, I).Start).Co_X) + J) * TwipsPerMeter), frmOutGraph(CPro
j).lneMxX.Y1 - Mx), Colors(LineColor)
      frmOutGraph(CProj).Line (frmOutGraph(CProj).lneMxX.X1 + (((Node(CPro
j, Element(CProj, I).Start).Co_X) + J) * TwipsPerMeter), frmOutGraph(CProj
).lneMxX.Y1)-(frmOutGraph(CProj).lneMxX.X1 + (((Node(CProj, Element(CProj,
 I).Start).Co_X) + J) * TwipsPerMeter), frmOutGraph(CProj).lneMxX.Y1 - Mx)
, Colors(LineColor)
     Next J
    Next I
End Sub

Sub Graph_Profile (COR As Integer, ColRow As Integer, AddSect As Single, T
PMX As Integer, TPP As Integer, InitX As Integer, InitY As Integer) '1 Lin
e Column Or Row
    Dim SumL As Single
    Dim Thick As Single
    Dim StartPoint As Single
    Dim EndPoint As Single
    Dim PF As Single
    Dim I As Integer
    Dim J As Single
    Dim LineColor As Integer
    Dim X As Single
    Dim Y As Single
    Dim X1 As Integer
    Dim Y1 As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 11

```
    Dim X2 As Integer
    Dim Y2 As Integer
    Dim e As Integer
    Dim ElementNum As Integer

    '''' Init Value
    LineColor = 37
    SumL = 0
    Trans_Element 'Design Column I
    Trans_BElement COR, ColRow 'Trans To Beam Element at ColRow Column Or
Row
    'TPP = 2000 / 30
    'If COR Then
    ' ElementNum = ColEle(ColRow)
    'ElseIf Not COR Then
    ' ElementNum = RowEle(ColRow)
    'End If
    'For I = 1 To ElementNum
    ' SumL = SumL + Element(CProj, I).Length
    'Next I
    'SumL = 0
    'TPM = frmOutGraph(CProj).Width \ SumL
    SumL = 0
    If COR Then
     For I = 1 To ColEle(ColRow)
      For J = 0 To Element(CProj, I).Length Step AddSect
       'PF = Cal_Profile(I, J, .1, .1, emax(CProj))
       X1 = InitX
       Y1 = InitY - ((SumL + J) * TPM)
       X2 = InitX - (PF * TPP)
       Y2 = InitY - ((SumL + J) * TPM)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(LineColor)
      Next J
      SumL = SumL + Element(CProj, I).Length
     Next I
    ElseIf Not COR Then
     For I = 1 To RowEle(ColRow)
      For J = 0 To Element(CProj, I).Length Step AddSect
       'PF = Cal_Profile(I, J, .1, .1, emax(CProj))
       X1 = InitX + ((SumL + J) * TPM)
       Y1 = InitY
       X2 = InitX + ((SumL + J) * TPM)
       Y2 = InitY - (PF * TPP)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(LineColor)
      Next J
      SumL = SumL + Element(CProj, I).Length
     Next I
    End If
'Sub Graph_BMD (AddSect As Single)
  '  Dim SumL As Single
 '   Dim MMax As Single
'    Dim MMin As Single
   'Dim StartPoint As Single
  ' Dim EndPoint As Single
  '  Dim Mx As Single
 '   Dim I As Integer
'    Dim J As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 12

```
    'Dim LineColor As Integer
    ' Dim TwipsPerMeter As Integer
   '  LineColor = 14
  '   'find best dimension
 '     SumL = 0
     'For I = 1 To ElementNum(CProj)
    '   SumL = SumL + Element(CProj, I).Length
   '   Next I
  '    TwipsPerMeter = (frmOutGraph(CProj).ScaleWidth - 1500) / SumL
 '     'Find M Max
    'MMax = 0
   ' MMin = 0
  '   For I = 1 To ElementNum(CProj)
 '     For J = 0 To Element(CProj, I).Length Step AddSect
'       Mx = Cal_BMoment(I, J, ML(CProj, I), MR(CProj, I))
   '    If Mx > MMax Then MMax = Mx
  '     If Mx < MMin Then MMin = Mx
 '     Next J
'     Next I
    ''find best amplitude
   ' TwipsPerMoment = ((frmOutGraph(CProj).ScaleHeight / 2) - 1500) / (MMa
x - MMin)
   '  frmOutGraph(CProj).PSet (frmOutGraph(CProj).lneMxX.X1, frmOutGraph(CP
roj).lneMxX.Y1), Colors(LineColor)
  '   For I = 1 To ElementNum(CProj)
'      For J = 0 To Element(CProj, I).Length Step AddSect
     ' Mx = Cal_BMoment(I, J, ML(CProj, I), MR(CProj, I)) * TwipsPerMoment
    '   frmOutGraph(CProj).Line -(frmOutGraph(CProj).lneMxX.X1 + (((Node(CP
roj, Element(CProj, I).Start).Co_X) + J) * TwipsPerMeter), frmOutGraph(CPr
oj).lneMxX.Y1 - Mx), Colors(LineColor)
   '    frmOutGraph(CProj).Line (frmOutGraph(CProj).lneMxX.X1 + (((Node(CPr
oj, Element(CProj, I).Start).Co_X) + J) * TwipsPerMeter), frmOutGraph(CPro
j).lneMxX.Y1)-(frmOutGraph(CProj).lneMxX.X1 + (((Node(CProj, Element(CProj
, I).Start).Co_X) + J) * TwipsPerMeter), frmOutGraph(CProj).lneMxX.Y1 - Mx
), Colors(LineColor)
  '    Next J
 '    Next I
'End Sub

End Sub

Sub Graph_SFD (AddSect As Single)
   . Dim SumL As Single
    Dim VMax As Single
    Dim VMin As Single
    Dim StartPoint As Single
    Dim EndPoint As Single
    Dim Vx As Single
    Dim v As Single
    Dim I As Integer
    Dim J As Single
    Dim LineColor As Integer
    Dim TwipsPerMeter As Integer
    LineColor = 34
    SumL = 0
    For I = 1 To ElementNum(CProj)
     SumL = SumL + Element(CProj, I).Length
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 13

```
    Next I
    TwipsPerMeter = (frmOutGraph(CProj).ScaleWidth - 1500) / SumL
    'Find V Max & Min
    VMax = 0
    VMin = 0
    For I = 1 To ElementNum(CProj)
     For J = 0 To Element(CProj, I).Length Step AddSect
      Vx = Cal_BShear(I, J)
      If Vx > VMax Then VMax = Vx
      If Vx < VMin Then VMin = Vx
     Next J
    Next I
    'find best amplitude
    TwipsPerShear = ((frmOutGraph(CProj).ScaleHeight / 2) - 1500) / (VMax
- VMin)
    frmOutGraph(CProj).PSet (frmOutGraph(CProj).lneSxX.X1, frmOutGraph(CPr
oj).lneSxX.Y1), Colors(LineColor)
    For I = 1 To ElementNum(CProj)
     For J = 0 To Element(CProj, I).Length Step AddSect
      Vx = Cal_BShear(I, J) * TwipsPerShear
      frmOutGraph(CProj).Line -(frmOutGraph(CProj).lneSxX.X1 + ((Node(CPro
j, Element(CProj, I).Start).Co_X + J) * TwipsPerMeter), frmOutGraph(CProj)
.lneSxX.Y1 - Vx), Colors(LineColor)
      frmOutGraph(CProj).Line (frmOutGraph(CProj).lneSxX.X1 + ((Node(CProj
, Element(CProj, I).Start).Co_X + J) * TwipsPerMeter), frmOutGraph(CProj).
lneSxX.Y1)-(frmOutGraph(CProj).lneSxX.X1 + ((Node(CProj, Element(CProj, I)
.Start).Co_X + J) * TwipsPerMeter), frmOutGraph(CProj).lneSxX.Y1 - Vx), Co
lors(LineColor)
     Next J
    Next I
End Sub

Sub List_Compute ()

End Sub

Sub List_DataInput ()

End Sub

Sub List_EndMoment ()

End Sub

Sub List_Loss (SType As Integer, PType As Integer, TimeApply As Integer, f
cpi As Single, fcpa As Single, fc As Single, fcds As Single, Volume As Sin
gle, Surface As Single)
    'Import data is
    '1. SType = StrandType
    '2. fcpi
    '3. fcpa
    '4. fc
    '5. fcds
    '6. TimeApply = Time apply load
    Dim Pxl As Single
    Dim Al As Single '
    Dim L As Single '
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 14

```
    Dim Esl As Single
    Dim Kes As Single
    Dim Es As Single
    Dim Eci As Single
    Dim SH As Single
    Dim Ksh As Single
    Dim CR As Single
    Dim RH As Single
    Dim Kcir As Single
    Dim fcir As Single
    Dim fg As Single
    Dim RE As Single
    Dim Kre As Single
    'Compute require data
    Select Case TimeApply
    Case 1: Ksh = .92
    Case 3: Ksh = .85
    Case 5: Ksh = .8
    Case 7: Ksh = .77
    Case 10: Ksh = .73
    Case 20: Ksh = .64
    Case 30: Ksh = .58
    Case 60: Ksh = .45
    End Select
    Select Case PType
    Case 0: 'Post-tensioned member
     Kcr = 1.6
     Kes = .5
     Kcir = 1
    Case 1: 'Pre-tensioned member
     Kcr = 1.6
     Kes = .5
     Kcir = 1
    End Select
    Eci = 4730 * (fci ^ (1 / 2))     'fci determine from dead load only
    Ec = 4730 * (GeneralData(CProj).SFc_ ^ (1 / 2))
    Es = GeneralData(CProj).SEs
    fcir = (Kcir * fcpi) - fg
    RH = 70
    'Friction Loss
    Pxl = (Al * UfrictionAngle(CProj) + L * KfrictionLength(CProj))
    'Elastic Shortening of concrete
    Select Case SType
    Case 0: 'BOND
     Esl = Kes * (Es / Eci) * fcir 'fcir = (Kcir*fcpi)-fg,Kcir = 1
    Case 1: 'UNBOND
     Esl = Kes * (Es / Eci) * fc  'fc = Average stress ,Kes = 0.5
    End Select
    'Shrinkage of concrete (SH)
    SH = ((8.2) / 1000000) * Ksh * Es * (1 - .024 * (Volume / Surface)) *
(100 - 70)  '100-RH ~ 30
    'Creep of concrete (CR)
    Select Case SType
    Case 0: 'BOND
     CR = Kcr * (Es / Ec) * fcpa 'fcpa = stress average between end & At p
oint
    Case 1: 'UNBOND
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
OUTPUT.BAS - 15

      CR = Kcr * (Es / Ec) * (fcir - fcds) 'fcds = stress from Dead Load (D
.L.)
    End Select
    'Relaxation (RE)
    RE = (Kre - (J * (SH + CR + Esl))) * c
End Sub

Sub List_NodeReaction ()

End Sub

Sub List_Profile ()

End Sub

Sub Print_Geometric ()
    frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
    frmMain.cmdlgSystem.CancelError = True
    On Error GoTo DlgErrorGeometric
    frmMain.cmdlgSystem.Action = 5

    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer


    ''''
    XMax = Column(CProj, 1).Co_X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 16

```
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    '''' Init Value
    'frmOutGraph(CProj).Cls
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Printer.PSet (3000, 250)
    Printer.Print "  SLAB TO DESIGN  "
    Printer.PSet (3000, 350)
    Printer.Print "=================="

    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (Printer.Width - (InitX * 2)) \ (XMax - XMin)'1m = TPMX Twips f
or X axis
    TPMY = (Printer.Height - (InitY * 2)) \ (YMax - YMin)'1m = TPMY Twips
for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If
    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 17

```
        Printer.Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
        Printer.Print Format$(I, "0") + "[" + Format$(Column(CProj, I).Co_X,
"#.00") + "," + Format$(Column(CProj, I).Co_Y, "#.00") + "]"
      Next I

      '''' Draw Element Conjunction in X axis
      For I = 1 To SRow(CProj)
       For J = 1 To RowEle(I)
        CStart = EleColRow(CProj, 0, I, J).Start
        CEnd = EleColRow(CProj, 0, I, J).End
        X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
        Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
        X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
        Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
        Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)

       Next J
      Next I

      '''' Draw Element Conjunction in Y axis
      For I = 1 To SCol(CProj)
       For J = 1 To ColEle(I)
        CStart = EleColRow(CProj, 1, I, J).Start
        CEnd = EleColRow(CProj, 1, I, J).End
        X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
        Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
        X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
        Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
        Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
       Next J
      Next I
      Printer.EndDoc
      Exit Sub
DlgErrorGeometric: Exit Sub
End Sub

Sub Print_Layout ()
      frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
      frmMain.cmdlgSystem.CancelError = True
      On Error GoTo DlgErrorLayOut
      frmMain.cmdlgSystem.Action = 5

    Dim XMax As Integer
      Dim XMin As Integer
      Dim YMax As Integer
      Dim YMin As Integer
      Dim InitX As Integer
      Dim InitY As Integer
      Dim Col As Integer
      Dim CStart As Integer
      Dim CEnd As Integer
      Dim TPMX As Single
      Dim TPMY As Single
      Dim TPP As Single
      Dim TPM As Single
      Dim AddSect As Single
      Dim X1 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 18

```
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim I As Integer
    Dim J As Integer
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer

    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    'Printer.Cls
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 2000 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (Printer.Width - (InitX * 2)) \ (XMax - XMin)'1m = TPMX Twips f
or X axis
    TPMY = (Printer.Height - (InitY * 2)) \ (YMax - YMin)'1m = TPMY Twips
for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 19

```
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If

    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     Printer.Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Profile in X axis
    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     Trans_BElement False, I
     Cal_FEndMoment False, I
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      Printer.PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 20

```
        PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
        X1 = InitX + (X * TPMX)
        Y1 = InitY
        X2 = InitX + (X * TPMX)
        Y2 = InitY - (M * TPM)
        Printer.Line -(X2, Y2), Colors(MomentColor)
        Printer.Line (X1, Y1)-(X2, Y2), Colors(MomentColor)

        X1 = InitX + (X * TPMX)
        Y1 = InitY
        X2 = InitX + (X * TPMX)
        Y2 = InitY - (PF * TPP)
        'Printer.Line -(X2, Y2), Colors(ProfileColor)
        'Printer.Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
      Next X
     Next J
    Next I

    '''' Draw Profile in Y axis
    For I = 1 To SCol(CProj)
     Trans_BElement True, I
     Cal_FEndMoment True, I
     For J = 1 To ColEle(I)
      InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
 * TPMX + Init1
      InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
 * TPMY + Init2
      Printer.PSet (InitX, InitY), ProfileColor
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (M * TPM)
       Y2 = InitY + (Y * TPMY)
       Printer.Line -(X2, Y2), Colors(MomentColor)
       Printer.Line (X1, Y1)-(X2, Y2), Colors(MomentColor)

       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (PF * TPP)
       Y2 = InitY + (Y * TPMY)
       'Printer.Line -(X2, Y2), Colors(ProfileColor)
       'Printer.Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
      Next Y
     Next J '820438
    Next I
    Printer.EndDoc
    Exit Sub
DlgErrorLayOut: Exit Sub
End Sub

Sub Print_Moment ()
    frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
    frmMain.cmdlgSystem.CancelError = True
    On Error GoTo DlgErrorMoment
    frmMain.cmdlgSystem.Action = 5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 21

```
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer


    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    'Printer.Cls
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Printer.PSet (3000, 250)
    Printer.Print " MOMENT DIAGRAM   "
    Printer.PSet (3000, 350)
    Printer.Print "=================="

    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (Printer.Width - (InitX * 2)) \ (XMax - XMin)'1m = TPMX Twips f
or X axis
    TPMY = (Printer.Height - (InitY * 2)) \ (YMax - YMin)'1m = TPMY Twips
for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips ..      '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If


    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     Printer.Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 23

```
      For J = 1 To ColEle(I)
       CStart = EleColRow(CProj, 1, I, J).Start
       CEnd = EleColRow(CProj, 1, I, J).End
       X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
       Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
       X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
       Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
       Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
      Next J
     Next I

     '''' Draw Moment in X axis
     Init1 = InitX
     Init2 = InitY
     For I = 1 To SRow(CProj)
      Trans_BElement False, I
      Cal_FEndMoment False, I
      For J = 1 To RowEle(I)
       InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
       InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
       Printer.PSet (InitX, InitY), ProfileColor
       If J = 1 Then
        TX1 = InitX
        TY1 = InitY + 150
        Printer.PSet (TX1, TY1)
        Printer.Print Format$(Cal_BShear(J, 0), "0.00")
       End If
       Printer.PSet (InitX, InitY), ProfileColor
       For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
        M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
        X1 = InitX + (X * TPMX)
        Y1 = InitY
        X2 = InitX + (X * TPMX)
        Y2 = InitY - (M * TPM)
        Printer.Line -(X2, Y2), Colors(MomentColor)
        Printer.Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
       Next X
       TX1 = InitX + EleColRow(CProj, 0, I, J).Length * TPMX
       TY1 = InitY + 150
       Printer.PSet (TX1, TY1)
       Printer.Print Format$(MR(CProj, J) * (-1), "0.00")
      Next J
     Next I

     '''' Draw Moment in Y axis
     For I = 1 To SCol(CProj)
      Trans_BElement True, I
      Cal_FEndMoment True, I
      For J = 1 To ColEle(I)
       InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
* TPMX + Init1
       InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
* TPMY + Init2
       Printer.PSet (InitX, InitY), ProfileColor
       If J = 1 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 24

```
        TX1 = InitX
        TY1 = InitY
        Printer.PSet (TX1, TY1)
        Printer.Print Format$(Cal_BShear(J, 0), "0.00")
       End If
       Printer.PSet (InitX, InitY), ProfileColor
       For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
        M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
        X1 = InitX
        Y1 = InitY + (Y * TPMY)
        X2 = InitX - (M * TPM)
        Y2 = InitY + (Y * TPMY)
        Printer.Line -(X2, Y2), Colors(MomentColor)
        Printer.Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
       Next Y
       TX1 = InitX
       TY1 = InitY + EleColRow(CProj, 1, I, J).Length * TPMY
       Printer.PSet (TX1, TY1)
       Printer.Print Format$(MR(CProj, J) * (-1), "0.00")
      Next J '820438
     Next I
     Printer.EndDoc
     Exit Sub
DlgErrorMoment: Exit Sub
End Sub

Sub Print_Shear ()
     frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
     frmMain.cmdlgSystem.CancelError = True
     On Error GoTo DlgErrorShear
     frmMain.cmdlgSystem.Action = 5

     Dim XMax As Integer
     Dim XMin As Integer
     Dim YMax As Integer
     Dim YMin As Integer
     Dim InitX As Integer
     Dim InitY As Integer
     Dim Col As Integer
     Dim CStart As Integer
     Dim CEnd As Integer
     Dim TPMX As Single
     Dim TPMY As Single
     Dim TPP As Single
     Dim TPM As Single
     Dim AddSect As Single
     Dim X1 As Single
     Dim Y1 As Single
     Dim X2 As Single
     Dim Y2 As Single
     Dim X As Single
     Dim Y As Single
     Dim I As Integer
     Dim J As Integer

     Dim Init1 As Integer
     Dim Init2 As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 25

```
    Dim PF As Single
    Dim v As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer



    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    'frmOutGraph(CProj).Cls
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Printer.PSet (3000, 250)
    Printer.Print "  SHEAR DIAGRAM   "
    Printer.PSet (3000, 400)
    Printer.Print "=================="

    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (Printer.Width - (InitX * 2)) \ (XMax - XMin)'1m = TPMX Twips f
or X axis
    TPMY = (Printer.Height - (InitY * 2)) \ (YMax - YMin)'1m = TPMY Twips
for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 26

```
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If

    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     Printer.Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Shear in X axis
    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     Trans_BElement False, I
     Cal_FEndMoment False, I
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      Printer.PSet (InitX, InitY), ProfileColor
      If J = 1 Then
       TX1 = InitX
       TY1 = InitY
       Printer.PSet (TX1, TY1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 27

```
        Printer.Print Format$(Cal_BShear(J, 0), "0.00")
      End If
      Printer.PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       v = Cal_BShear(J, X)
       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (v * TPM)
       Printer.Line -(X2, Y2), Colors(MomentColor)
       Printer.Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
      Next X
      TX1 = InitX + EleColRow(CProj, 0, I, J).Length * TPMX
      TY1 = InitY + 150
      Printer.PSet (TX1, TY1)
      Printer.Print Format$(Cal_BShear(J, EleColRow(CProj, 0, I, J).Length
), "0.00")
     Next J
    Next I

    '''' Draw Shear in Y axis
    For I = 1 To SCol(CProj)
     Trans_BElement True, I
     Cal_FEndMoment True, I
     For J = 1 To ColEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      Printer.PSet (InitX, InitY), ProfileColor
      If J = 1 Then
       TX1 = InitX
       TY1 = InitY
       Printer.PSet (TX1, TY1)
       Printer.Print Format$(Cal_BShear(J, 0), "0.00")
      End If

      Printer.PSet (InitX, InitY), ProfileColor
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       v = Cal_BShear(J, Y)
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (v * TPM)
       Y2 = InitY + (Y * TPMY)
       Printer.Line -(X2, Y2), Colors(MomentColor)
       Printer.Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
      Next Y
      TX1 = InitX
      TY1 = InitY + EleColRow(CProj, 1, I, J).Length * TPMY
      Printer.PSet (TX1, TY1)

      Printer.Print Format$(Cal_BShear(J, EleColRow(CProj, 1, I, J).Length
), "0.00")
     Next J '820438
    Next I
    Printer.EndDoc
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 28

```
    Exit Sub
DlgErrorShear: Exit Sub
End Sub

Sub Print_Text (Title As Integer, Text As String)
    Dim CH As String * 1
    Dim CR As String * 1
    Dim LF As String * 1
    Dim TB As String * 1
    Dim TitleText As String
    CR = Chr$(13)
    LF = Chr$(10)
    TB = Chr$(9)
    Printer.FontName = "Courier"
    Printer.FontBold = True
    Printer.FontSize = 7
    Text = "  " + Text 'tab from front 2 space
    TitleText = ""
    Select Case Title
    Case 0: 'Unneccessary title
     TitleText = ""
    Case 1: 'for Column
     TitleText = TitleText + CR + LF
     TitleText = TitleText + "COLUMN NO.  CO_X(m)     CO_Y(m)     LENGTH(m
)    " + CR + LF
     TitleText = TitleText + CR + LF
    Case 2: 'for row element
     TitleText = TitleText + CR + LF
     TitleText = TitleText + " ROW   ELEMENT   START-COL  END-COL  LENGTH(
m)    TOP-BAY(m) BOTTOM-BAY(m)" + CR + LF
     TitleText = TitleText + CR + LF
    Case 3: 'for column element
     TitleText = TitleText + CR + LF
     TitleText = TitleText + "COLUMN ELEMENT   START-COL  END-COL  LENGTH(
m)    TOP-BAY(m) BOTTOM-BAY(m)" + CR + LF
     TitleText = TitleText + CR + LF
    Case 4: 'for shear check
     TitleText = TitleText + CR + LF
     TitleText = TitleText + "COLUMN" + s10 + s10 + s5 + "PUNCHING SHEAR"
+ s10 + s10 + s9 + "BEAM-ACTION-SHEAR" + CR + LF
     TitleText = TitleText + s2 + "NO." + s9 + "COLUMN" + s3 + s10 + "DROP
-PANEL" + s1 + s10 + "COLUMN-CAPITAL" + CR + LF
     TitleText = TitleText + s8 + "Vc(kg)" + s5 + "Vn(kg)" + s5 + "Vc(kg)"
 + s5 + "Vn(kg)" + s5 + "Vc(kg)" + s5 + "Vn(kg)" + s5 + "Vc(kg)" + s5 + "V
n(kg)" + CR + LF
     TitleText = TitleText + CR + LF
    Case 5: 'for drop panel & capital dimension
     TitleText = TitleText + CR + LF
     TitleText = TitleText + s2 + "COLUMN" + s2 + "DROP WIDTH(m)" + s2 + "
DROP LENGTH(m)" + s2 + "DROP THICK(m)" + s2 + "CAPITAL WIDTH(m)" + s2 + "C
APITAL LENGTH(m)" + CR + LF
     TitleText = TitleText + CR + LF
    Case 6: 'Check flexeral stress in row
     TitleText = TitleText + CR + LF
     TitleText = TitleText + s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE
FROM LEFT(m)" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/c
m^2)" + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 29

```
      TitleText = TitleText + "
                   TOP" + s10 + "BOTTOM" + CR + LF
      TitleText = TitleText + CR + LF
     Case 7: 'Check flexeral stress in column
      TitleText = TitleText + CR + LF
      TitleText = TitleText + "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FR
OM TOP(m) " + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^
2)" + CR + LF
      TitleText = TitleText + "
                   TOP" + s10 + "BOTTOM" + CR + LF
      TitleText = TitleText + CR + LF
     Case 8: 'Reinforcement for flat plate slab in row
      TitleText = TitleText + CR + LF
      TitleText = TitleText + "ROW" + s3 + "ELEMENT" + s10 + s1 + "SECTION"
 + s10 + s3 + "DESIGN M(kg-m)" + s3 + "As(cm^2)" + CR + LF
      TitleText = TitleText + CR + LF
     Case 9: 'Reinforcement for flat plate slab in column
      TitleText = TitleText + CR + LF
      TitleText = TitleText + "COLUMN" + s1 + "ELEMENT" + s10 + s1 + "SECTI
ON" + s10 + s3 + "DESIGN M(kg-m)" + s3 + "As(cm^2)" + CR + LF
      TitleText = TitleText + CR + LF
     End Select

     'Printer.CurrentY = 500
     If Printer.CurrentY > (Printer.Height - 1500) Then
      Printer.NewPage
      Printer.CurrentY = 1000
      Text = TitleText + Text
     End If
     Printer.Print Text;
End Sub

Sub Print_Try ()
     frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
     frmMain.cmdlgSystem.CancelError = True
     On Error GoTo DlgError
     frmMain.cmdlgSystem.Action = 5

     Dim XMax As Integer
     Dim XMin As Integer
     Dim YMax As Integer
     Dim YMin As Integer
     Dim InitX As Integer
     Dim InitY As Integer
     Dim Col As Integer
     Dim CStart As Integer
     Dim CEnd As Integer
     Dim TPMX As Single
     Dim TPMY As Single
     Dim TPP As Single
     Dim TPM As Single
     Dim AddSect As Single
     Dim X1 As Single
     Dim Y1 As Single
     Dim X2 As Single
     Dim Y2 As Single
     Dim X As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 30

```
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    ''''end moment
    Dim MWL As Single
    Dim MWR As Single
    Dim MSL As Single
    Dim MSR As Single
    Dim MUL As Single
    Dim MUR As Single

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    'Dim Text As String
    Dim CR As String * 1
    Dim LF As String * 1
    Dim TB As String * 1
    CR = Chr$(13)
    LF = Chr$(10)
    TB = Chr$(9)
    Const ucm = " cm"
    Const um = " m"
    Const ukg = " kg"
    Const uksc = " ksc"
    Const uksm = " kg/m^2"
    Dim CNo As Integer
    Dim RNum As Integer

    'shear
    'Column
    Dim Vc0 As Single
    Dim Vn0 As Single
    'Drop
    Dim Vc1 As Single
    Dim Vn1 As Single
    'Capital
    Dim Vc2 As Single
    Dim Vn2 As Single
    'Diagnal
    Dim Vc3 As Single
    Dim Vn3 As Single

    ''''M design
    Dim MDL As Single
    Dim MDD As Single
    Dim MDR As Single
    'M design at column strip
    Dim MCL As Single
    Dim MCD As Single
    Dim MCR As Single
    'M design at middle strip
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 31

```
    Dim MML1 As Single
    Dim MMD1 As Single
    Dim MMR1 As Single
    Dim MML2 As Single
    Dim MMD2 As Single
    Dim MMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim MLB1  As Single
    Dim MMB1  As Single
    Dim MRB1  As Single
    Dim MLT1  As Single
    Dim MMT1  As Single
    Dim MRT1  As Single
    'Half middle strip 2
    Dim MLB2  As Single
    Dim MMB2  As Single
    Dim MRB2  As Single
    Dim MLT2  As Single
    Dim MMT2  As Single
    Dim MRT2  As Single
    'column strip
    Dim CLB  As Single
    Dim CMB  As Single
    Dim CRB  As Single
    Dim CLT  As Single
    Dim CMT  As Single
    Dim CRT  As Single

    ''''Reinforcement design
    Dim RMDL As Single
    Dim RMDD As Single
    Dim RMDR As Single
    'M design at column strip
    Dim RMCL As Single
    Dim RMCD As Single
    Dim RMCR As Single
    'M design at middle strip
    Dim RMML1 As Single
    Dim RMMD1 As Single
    Dim RMMR1 As Single
    Dim RMML2 As Single
    Dim RMMD2 As Single
    Dim RMMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim RMLB1  As Single
    Dim RMMB1  As Single
    Dim RMRB1  As Single
    Dim RMLT1  As Single
    Dim RMMT1  As Single
    Dim RMRT1  As Single
    'Half middle strip 2
    Dim RMLB2  As Single
    Dim RMMB2  As Single
    Dim RMRB2  As Single
    Dim RMLT2  As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 32

```
    Dim RMMT2  As Single
    Dim RMRT2  As Single
    'column strip
    Dim RCLB  As Single
    Dim RCMB  As Single
    Dim RCRB  As Single
    Dim RCLT  As Single
    Dim RCMT  As Single
    Dim RCRT  As Single


    AddSect = .5


    '''' Init
    frmList(CProj).Cls

    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
 As #20 Len = 4
    '''' Information
    Print_Text 0, "    List of Flat Slab & Flat Plate Design" + CR + LF
    Print_Text 0, "Data      " + TB + "  : " + SData(CProj).DateData + "
::: Time      : " + SData(CProj).TimeData + CR + LF
    Print_Text 0, "File Name " + TB + "  : " + SData(CProj).FileName + CR
+ LF
    Print_Text 0, "Project   " + TB + "  : " + SData(CProj).ProjectData +
CR + LF
    Print_Text 0, "Engineer  " + TB + "  : " + SData(CProj).Engineer + CR
+ LF
    Print_Text 0, "Address   " + TB + "  : " + SData(CProj).AddressData +
CR + LF
    Print_Text 0, "Information" + TB + " : " + SData(CProj).InformData + C
R + LF

    '''' Data for Design
    Print_Text 0, "Length Unit   " + TB + " : Meter(m)  ::: Load Unit  :
Kilogram(Kg)" + CR + LF
    Print_Text 0, "fc' (Slab)    " + TB + " : " + Format$(GeneralData(CPro
j).SFc_, ".00") + uksc + CR + LF
    Print_Text 0, "fc' (Column)  " + TB + " : " + Format$(GeneralData(CPro
j).CFc_, ".00") + uksc + CR + LF
    Print_Text 0, "fy (Strand)   " + TB + " : " + Format$(GeneralData(CPro
j).SFs_, ".00") + uksc + CR + LF
    Select Case StrandIndex(CProj):
    Case 0:
    Case 1:
    Case 2:
    Case 3:
    Case 4:
    Case 5:
    Case 6:
    Case 7:
    End Select
    'Sum_Text Text(TextNo),TextNo, "Es (Strand)     : " + Format$(Es, ".00"
) + CR + LF
    Print_Text 0, "Ec (Slab)       : " + Format$(GeneralData(CProj).SEc, ".
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 33

```
00") + uksc + CR + LF
    Print_Text 0, "Ec (Column)     : " + Format$(GeneralData(CProj).CEc, ".
00") + uksc + CR + LF
    Print_Text 0, "Slab Thickness : " + Format$(100 * Thick(CProj), ".00")
 + ucm + CR + LF
    Print_Text 0, "Slab Covering  : " + Format$((Thick(CProj) - d(CProj))
* 100, ".00") + ucm + CR + LF
    Print_Text 0, "Drape          : " + Format$((Thick(CProj) - 2 * (Thick
(CProj) - d(CProj))) * 100, ".00") + ucm + CR + LF


    '''' Column Data
    'frmList(CProj).FontSize = 12
    frmList(CProj).FontBold = True
    Print_Text 0, "Column Data" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "Column  No." + TB + "Co_X(m)" + TB + TB + "Co_Y(m)" + T
B + "SupportType" + TB + " Length(m) " + CR + LF
    Print_Text 0, CR + LF
    CNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CNo = CNo + 1
      Co_X = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X
      Co_Y = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y
      Support = Column(CProj, EleColRow(CProj, 0, I, J).Start).SupportType
      Length = Column(CProj, EleColRow(CProj, 0, I, J).Start).Length
      Print_Text 1, Format$(CNo, "@@@") + TB + TB + Format$(Co_X, "0.00E+0
0") + TB + TB + Format$(Co_Y, "0.00E+00") + TB + TB + Format$(Support, "@@
") + TB + Format$(Length, "0.00E+00") + CR + LF
      If J = RowEle(I) Then
       CNo = CNo + 1
       Co_X = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_X
       Co_Y = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_Y
       Support = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Supp
ortType
       Length = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Lengt
h
       Print_Text 1, Format$(CNo, "@@@") + TB + TB + Format$(Co_X, "0.00E+
00") + TB + TB + Format$(Co_Y, "0.00E+00") + TB + TB + Format$(Support, "@
@") + TB + Format$(Length, "0.00E+00") + CR + LF
      End If
     Next J
    Next I

    '''' Element  Data
    Dim LCol As Integer
    Dim RCol As Integer
    Dim LBay As Single
    Dim RBay As Single
    'Row
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "Row Element Data" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, " Row  :Element  No.:Start Column: End  Column:   Length
 : Top  Bay   : Bottom Bay  " + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 34

```
    Print_Text 0, "                                                    (m)
      (m)            (m)   " + CR + LF
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      LCol = EleColRow(CProj, 0, I, J).Start
      RCol = EleColRow(CProj, 0, I, J).End
      Length = EleColRow(CProj, 0, I, J).Length
      LBay = EleColRow(CProj, 0, I, J).L1
      RBay = EleColRow(CProj, 0, I, J).L2
      Print_Text 1, Format$(I, "@@@@@@@@") + Format$(J, "@@@@@@@@@@@@@@@@@@
@@") + Format$(LCol, "@@@@@@@@@@@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@@@@@@
@@@@") + Format$(Length, "@@@@@@@@@@@.00") + Format$(LBay, "@@@@@@@@@@@.00
") + Format$(RBay, "@@@@@@@@@@@.00") + CR + LF
     Next J
    Next I
    'Column
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "Row Element Data" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "Column:Element  No.:Start Column: End  Column:   Length
  : Left Bay   : Right  Bay  " + CR + LF
    Print_Text 0, CR + LF
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      LCol = EleColRow(CProj, 1, I, J).Start
      RCol = EleColRow(CProj, 1, I, J).End
      Length = EleColRow(CProj, 1, I, J).Length
      LBay = EleColRow(CProj, 1, I, J).L1
      RBay = EleColRow(CProj, 1, I, J).L2
      Print_Text 2, Format$(I, "@@@@@@@@") + Format$(J, "@@@@@@@@@@@@@@@@@@
@@") + Format$(LCol, "@@@@@@@@@@@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@@@@@@
@@@@") + Format$(Length, "@@@@@@@@@@@.00") + Format$(LBay, "@@@@@@@@@@@.00
") + Format$(RBay, "@@@@@@@@@@@.00") + CR + LF
     Next J
    Next I
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Dim SSystem As String * 50
    Select Case GeneralData(CProj).SlabIndex
    Case 0:
     SSystem = "Flat Plate"
    Case 1:
     SSystem = "Flat Slab With Drop Panel"
    Case 2:
     SSystem = "Flat Slab With Column Capital"
    Case 3:
     SSystem = "Flat Slab With Column Drop Panel And Column Capital"
    End Select
    Print_Text 0, "SLAB SYSTEM : " + SSystem + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "Load factor ::: " + CR + LF
    Print_Text 0, "Wu = " + Format$(DeadCo, "0.0") + "DL" + Format$(LiveCo
, "0.0") + "LL" + CR + LF
    Print_Text 0, "Unfactored Dead Load          = " + Format$(DeadLoad(C
Proj), "0.00") + uksm + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 35

```
    Print_Text 0, "Unfactored Live Load           = " + Format$(LiveLoad(C
Proj), "0.00") + uksm + CR + LF
    Print_Text 0, "Unfactored Supper imposed load = " + Format$(SuperLoad(
CProj), "0.00") + uksm + CR + LF
    Print_Text 0, ""

    Print_Text 0, "Column " + TB + "      Punching at Column    " + TB + TB
 + "         Punching at Drop Panel    " + TB + "        Punching at Capit
al        " + TB + "              Diagonal shear          " + CR + LF
    Print_Text 0, " No.    " + Chr$(9) + "  Vc(kg)   " + Chr$(9) + Chr$(9) +
 " Vn(kg)   " + Chr$(9) + Chr$(9) + " Vc(kg)   " + Chr$(9) + Chr$(9) + " Vn(
kg) " + Chr$(9) + Chr$(9) + " Vc(kg) " + Chr$(9) + Chr$(9) + "  Vn(kg)   "
+ Chr$(9) + Chr$(9) + "  Vc(kg)   " + Chr$(9) + Chr$(9) + "  Vn(kg)   " + Ch
r$(9) + CR + LF
    Print_Text 0, CR + LF
    RNum = 2
    RIden = 2
    Dim ColumnNo As Integer
    ColumnNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      ColumnNo = ColumnNo + 1
      CNo = EleColRow(CProj, 0, I, J).Start
      If Column(CProj, CNo).SupportType <> 2 Then 'If Not Free,Mean That i
s Column support
       RIden = RNum + ((CNo - 1) * 8)
       Get #20, RIden + 1, Vc0
       Get #20, RIden + 2, Vn0
       Get #20, RIden + 3, Vc1
       Get #20, RIden + 4, Vn1
       Get #20, RIden + 5, Vc2
       Get #20, RIden + 6, Vn2
       Get #20, RIden + 7, Vc3
       Get #20, RIden + 8, Vn3
       Print_Text 3, Format$(ColumnNo, "@@@@") + Chr$(9) + Format$(Vc0, "0
.0000E+00") + Chr$(9) + Format$(Vn0, "0.0000E+00") + Chr$(9) + Format$(Vc1
, "0.0000E+00") + Chr$(9) + Format$(Vn1, "0.0000E+00") + Chr$(9) + Format$
(Vc2, "0.0000E+00") + Chr$(9) + Format$(Vn2, "0.0000E+00") + Chr$(9) + For
mat$(Vc3, "0.0000E+00") + Chr$(9) + Format$(Vn3, "0.0000E+00") + CR + LF
      End If
      If J = RowEle(I) Then
       ColumnNo = ColumnNo + 1
       CNo = EleColRow(CProj, 0, I, J).End
       If Column(CProj, CNo).SupportType <> 2 Then
        RIden = RNum + ((CNo - 1) * 8)
        Get #20, RIden + 1, Vc0
        Get #20, RIden + 2, Vn0
        Get #20, RIden + 3, Vc1
        Get #20, RIden + 4, Vn1
        Get #20, RIden + 5, Vc2
        Get #20, RIden + 6, Vn2
        Get #20, RIden + 7, Vc3
        Get #20, RIden + 8, Vn3
        Print_Text 3, Format$(ColumnNo, "@@@@") + Chr$(9) + Format$(Vc0, "
0.0000E+00") + Chr$(9) + Format$(Vn0, "0.0000E+00") + Chr$(9) + Format$(Vc
1, "0.0000E+00") + Chr$(9) + Format$(Vn1, "0.0000E+00") + Chr$(9) + Format
$(Vc2, "0.0000E+00") + Chr$(9) + Format$(Vn2, "0.0000E+00") + Chr$(9) + Fo
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 36

```
rmat$(Vc3, "0.0000E+00") + Chr$(9) + Format$(Vn3, "0.0000E+00") + CR + LF

       End If
      End If
     Next J
    Next I

    'Drop panel & Capital
    CNo = 0
    ColumnNo = 0
    Print_Text 0, CR + LF
    Print_Text 0, "Dimension of drop & column capital (m)"
    Print_Text 0, CR + LF
    Print_Text 0, " No. " + TB + "Drop Width" + TB + "Drop Length" + TB +
"Drop Thick" + TB + "Capital Width" + TB + "Capital Length" + TB + CR + LF
    Print_Text 0, CR + LF
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CNo = CNo + 1
      ColumnNo = EleColRow(CProj, 0, I, J).Start
      DropX = Column(CProj, ColumnNo).DropX
      DropY = Column(CProj, ColumnNo).DropY
      DropThick = Column(CProj, ColumnNo).DropThick
      CapitalX = Column(CProj, ColumnNo).CapitalX
      CapitalY = Column(CProj, ColumnNo).CapitalY
      Print_Text 4, Format$(CNo, "@@@@") + Chr$(9) + Format$(DropX, "0.000
0E+00") + Chr$(9) + Format$(DropY, "0.0000E+00") + Chr$(9) + Format$(DropT
hick, "0.0000E+00") + Chr$(9) + Format$(CapitalX, "0.0000E+00") + Chr$(9)
+ Format$(CapitalY, "0.0000E+00") + CR + LF
      If J = RowEle(I) Then
       CNo = CNo + 1
       ColumnNo = EleColRow(CProj, 0, I, RowEle(I)).End
       DropX = Column(CProj, ColumnNo).DropX
       DropY = Column(CProj, ColumnNo).DropY
       DropThick = Column(CProj, ColumnNo).DropThick
       CapitalX = Column(CProj, ColumnNo).CapitalX
       CapitalY = Column(CProj, ColumnNo).CapitalY
       Print_Text 4, Format$(CNo, "@@@@") + Chr$(9) + Format$(DropX, "0.00
00E+00") + Chr$(9) + Format$(DropY, "0.0000E+00") + Chr$(9) + Format$(Drop
Thick, "0.0000E+00") + Chr$(9) + Format$(CapitalX, "0.0000E+00") + Chr$(9)
 + Format$(CapitalY, "0.0000E+00") + CR + LF
      End If
     Next J
    Next I
    'declare
    Dim Lmax As Single
    Dim Bbeam As Single
    Dim P As Single
    Dim Top As Single
    Dim Bottom As Single
    Dim Text1 As String
    Dim Text2 As String
    Dim Text3 As String
    Dim Text4 As String


    Text1 = ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 37

```
    ''''check Cracking Moment & Deflection
    Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_UltimateEndMoment False, I
     Mu = Find_MuMax(False, I)
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, " ROW : " + Format$(I, "@@@") + CR + LF
     Print_Text 0, " No of tendon = " + Format$(Find_AsStrand(Lmax, Bbeam)
, "@@@") + TB + "Unit" + CR + LF
     Print_Text 0, " Tension per 1 unit tendon = " + Format$(GeneralData(C
Proj).SFs_, "0.00") + " Kg" + CR + LF
     Print_Text 0, " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), "0.00"
) + " kg-m" + CR + LF
     Print_Text 0, " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
     Print_Text 0, " Deflection(Actual)    = " + Format$(Find_MaxDeflectio
n(False, I, 1), "0.00E+00") + um + CR + LF
     Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
False, I), "0.00E+00") + um + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT USED" + CR + LF
     Print_Text 0, " Deflection(Actual)    = " + Format$(Find_MaxDeflectio
n(False, I, 2), "0.00E+00") + um + CR + LF
     Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
False, I), "0.00E+00") + um + CR + LF
     Print_Text 0, "=================================================
=================================" + CR + LF
    Next I

    'Text1flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_UltimateEndMoment True, I
     Mu = Find_MuMax(True, I)
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, " COLUMN : " + Format$(I, "@@@") + CR + LF
     Print_Text 0, " No of tendon = " + Format$(Find_AsStrand(Lmax, Bbeam)
, "@@@") + TB + "Unit" + CR + LF
     Print_Text 0, " Tension per 1 unit tendon = " + Format$(GeneralData(C
Proj).SFs_, ".00") + " Kg" + CR + LF
     Print_Text 0, " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), ".00")
 + " kg-m" + CR + LF
     Print_Text 0, " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT USED" + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 38

```
      Print_Text 0, " Deflection(Actual)     = " + Format$(Find_MaxDeflectio
n(True, I, 1), "0.00E+00") + um + CR + LF
      Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
True, I), "0.00E+00") + um + CR + LF
      Print_Text 0, "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
      Print_Text 0, " Deflection(Actual)     = " + Format$(Find_MaxDeflectio
n(True, I, 2), "0.00E+00") + um + CR + LF
      Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
True, I), "0.00E+00") + um + CR + LF
      Print_Text 0, "=================================================
=====================================" + CR + LF
    Next I

    '''''at working load
    Text2 = ""
    Wu = DeadLoad(CProj)
    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ElasticEndMoment False, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, "Row No.  " + "  Element No." + TB + "Distance From Lef
t" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR +
LF
     Print_Text 0, "          " + TB + TB + "                " + TB + TB + TB + TB
 + TB + TB + TB + TB + "Top" + TB + "Bottom" + CR + LF
     Print_Text 0, CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Print_Text 5, Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") +
 TB + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00
") + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 1
0000, "0.00E+00") + CR + LF
     Next X
      Print_Text 0, "=================================================
=====================================" + CR + LF
    Next J
   Next I

   'Text2 flexeral stress in Column
   Lmax = 0
   Bbeam = 0
   Mu = 0
   For I = 1 To SRow(CProj)
    Trans_UniformLoad Wu
    Trans_BElement True, I
    Cal_ElasticEndMoment True, I
    Find_BL True, I, Lmax, Bbeam
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 39

```
      P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
      Print_Text 0, CR + LF
      Print_Text 0, "Column No." + "  Element No." + TB + "Distance From Le
ft" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR +
 LF
      Print_Text 0, "        " + TB + TB + "            " + TB + TB + TB + TB
 + TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
      For J = 1 To RowEle(I)
       For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
        M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
        PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
        Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
        Print_Text 6, Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") +
 TB + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00
") + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 1
0000, "0.00E+00") + CR + LF
       Next X
       Print_Text 0, "=============================================================
=======================================" + CR + LF
     Next J
    Next I

    AddSect = .5
    ''''check at transfer load
    Text3 = ""
    Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    'Text3 flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ServiceEndMoment False, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, "Row No.    " + "  Element No." + TB + "Distance From L
eft" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR
+ LF
     Print_Text 0, "        " + TB + TB + "            " + TB + TB + TB + TB.
 + TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
     Print_Text 0, CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Print_Text 0, Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") +
 TB + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00
") + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 1
0000, "0.00E+00") + CR + LF
      Next X
      Print_Text 0, "=============================================================
=======================================" + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 40

```
      Next J
     Next I

     'Text3 flexeral stress in Column
     Lmax = 0
     Bbeam = 0
     Mu = 0
     For I = 1 To SRow(CProj)
      Trans_UniformLoad Wu
      Trans_BElement True, I
      Cal_ServiceEndMoment False, I
      Find_BL True, I, Lmax, Bbeam
      P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
      Print_Text 0, CR + LF
      Print_Text 0, "Column No." + "  Element No." + TB + "Distance From Le
ft" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR +
 LF
      Print_Text 0, "          " + TB + TB + "              " + TB + TB + TB + TB
 + TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
      Print_Text 0, CR + LF
      For J = 1 To RowEle(I)
       For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
        M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
        PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
        Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
        Print_Text 0, Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") +
 TB + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00
") + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 1
0000, "0.00E+00") + CR + LF
       Next X
       Print_Text 0, "=============================================================
=======================================" + CR + LF
      Next J
     Next I
     Printer.EndDoc
     Close #20
     Exit Sub
DlgError: Exit Sub
End Sub

Sub Screen_Geometric ()
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 41

```
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer


    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    frmOutGraph(CProj).Cls
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    frmOutGraph(CProj).PSet (3000, 250)
    frmOutGraph(CProj).Print "  SLAB TO DESIGN  "
    frmOutGraph(CProj).PSet (3000, 350)
    frmOutGraph(CProj).Print "=================="

    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 42

```
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m = T
PMX Twips for X axis
    TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
TPMY Twips for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips       '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If
    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
     frmOutGraph(CProj).Print Format$(I, "0") + "[" + Format$(Column(CProj
, I).Co_X, "#.00") + "," + Format$(Column(CProj, I).Co_Y, "#.00") + "]"
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)

     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I
End Sub

Sub Screen_Layout ()
    Dim XMax As Integer
    Dim XMin As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 43

```
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer

    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    frmOutGraph(CProj).Cls
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 44

```
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m = T
PMX Twips for X axis
    TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
TPMY Twips for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If


    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 45

```
    '''' Draw Profile in X axis
    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     Trans_BElement False, I
     Cal_FEndMoment False, I
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (M * TPM)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(MomentColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(MomentColor)

       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (PF * TPP)
       'frmOutGraph(CProj).Line -(X2, Y2), Colors(ProfileColor)
       'frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
      Next X
     Next J
    Next I

    '''' Draw Profile in Y axis
    For I = 1 To SCol(CProj)
     Trans_BElement True, I
     Cal_FEndMoment True, I
     For J = 1 To ColEle(I)
      InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
* TPMX + Init1
      InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
* TPMY + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (M * TPM)
       Y2 = InitY + (Y * TPMY)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(MomentColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(MomentColor)

       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (PF * TPP)
       Y2 = InitY + (Y * TPMY)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 46

```
        'frmOutGraph(CProj).Line -(X2, Y2), Colors(ProfileColor)
        'frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
      Next Y
     Next J '820438
    Next I
End Sub

Sub Screen_Moment ()
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer



    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 47

```
      If YMin > Column(CProj, I).Co_Y Then
       YMin = Column(CProj, I).Co_Y
      End If
     Next I

     ''''  Init Value
     frmOutGraph(CProj).Cls
     Col = 50      'Column Width in Twips
     AddSect = .1 'Additional Section
     InitX = 1000 'Left Edge in Twips
     InitY = 750 'Top Edge in Twips
     ColumnColor = 34
     ElementColor = 30
     ProfileColor = 27
     MomentColor = 17
     frmOutGraph(CProj).PSet (3000, 250)
     frmOutGraph(CProj).Print " MOMENT DIAGRAM    "
     frmOutGraph(CProj).PSet (3000, 350)
     frmOutGraph(CProj).Print "=================="

     Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

     ''''
     Trans_UniformLoad (Wu)'Trans Load

     ''''
     TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m = T
PMX Twips for X axis
     TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
TPMY Twips for Y axis
     TPP = 2000 / .3  'profile 30 cm use 2000 Twips       '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
     TPM = .01
     If TPMX < TPMY Then
      TPMY = TPMX
     ElseIf TPMX >= TPMY Then
      TPMX = TPMY
     End If


     '''' Draw Column
     For I = 1 To SNodeNum(CProj)
      X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
      Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
      X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
      Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
     Next I

     '''' Draw Element Conjunction in X axis
     For I = 1 To SRow(CProj)
      For J = 1 To RowEle(I)
       CStart = EleColRow(CProj, 0, I, J).Start
       CEnd = EleColRow(CProj, 0, I, J).End
       X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
       Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
       X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 48

```
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Moment in X axis
    frmOutGraph(CProj).ForeColor = Colors(34)
    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     Trans_BElement False, I
     Cal_FEndMoment False, I
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      If J = 1 Then
       TX1 = InitX
       TY1 = InitY + 150
       frmOutGraph(CProj).PSet (TX1, TY1)
       frmOutGraph(CProj).Print Format$(Cal_BShear(J, 0), "0.00")
      End If
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (M * TPM)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(MomentColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
      Next X
      TX1 = InitX + EleColRow(CProj, 0, I, J).Length * TPMX
      TY1 = InitY + 150
      frmOutGraph(CProj).PSet (TX1, TY1)
      frmOutGraph(CProj).Print Format$(MR(CProj, J) * (-1), "0.00")
     Next J
    Next I

    '''' Draw Moment in Y axis
    frmOutGraph(CProj).ForeColor = Colors(39)
    For I = 1 To SCol(CProj)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 49

```
      Trans_BElement True, I
      Cal_FEndMoment True, I
      For J = 1 To ColEle(I)
       InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
 * TPMX + Init1
       InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
 * TPMY + Init2
       frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
       If J = 1 Then
        TX1 = InitX
        TY1 = InitY
        frmOutGraph(CProj).PSet (TX1, TY1)
        frmOutGraph(CProj).Print Format$(Cal_BShear(J, 0), "0.00")
       End If
       frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
       For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
        M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
        X1 = InitX
        Y1 = InitY + (Y * TPMY)
        X2 = InitX - (M * TPM)
        Y2 = InitY + (Y * TPMY)
        frmOutGraph(CProj).Line -(X2, Y2), Colors(MomentColor)
        frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
       Next Y
       TX1 = InitX
       TY1 = InitY + EleColRow(CProj, 1, I, J).Length * TPMY
       frmOutGraph(CProj).PSet (TX1, TY1)
       frmOutGraph(CProj).Print Format$(MR(CProj, J) * (-1), "0.00")
      Next J '820438
     Next I
     frmOutGraph(CProj).ForeColor = Colors(34)
End Sub

Sub Screen_Profile ()
     Dim XMax As Integer
     Dim XMin As Integer
     Dim YMax As Integer
     Dim YMin As Integer
     Dim InitX As Integer
     Dim InitY As Integer
     Dim Col As Integer
     Dim CStart As Integer
     Dim CEnd As Integer
     Dim TPMX As Single
     Dim TPMY As Single
     Dim TPP As Single
     Dim TPM As Single
     Dim AddSect As Single
     Dim X1 As Single
     Dim Y1 As Single
     Dim X2 As Single
     Dim Y2 As Single
     Dim X As Single
     Dim Y As Single
     Dim I As Integer
     Dim J As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 50

```
    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer

    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    frmOutGraph(CProj).Cls
    Col = 50     'Column Width in Twips
    AddSect = .5 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m = T
PMX Twips for X axis
    TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
TPMY Twips for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 51

```
    'Head
    frmOutGraph(CProj).PSet (3000, 250)
    frmOutGraph(CProj).Print "PROFILE OF TENDON"
    frmOutGraph(CProj).PSet (3000, 350)
    frmOutGraph(CProj).Print "=================="
    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I


    Open DirApp + "profile.swp" For Random Access Read Write Lock Read Wri
te As #21 Len = 4
    Dim WNum As Integer
    Dim L As Single
    WNum = 0
    AddSect = .5

    '''' Draw Profile in X axis

    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 52

```
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (PF * TPP)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(ProfileColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
       'frmOutGraph(CProj).PSet (X1, Y1)
       'frmOutGraph(CProj).Print Format$(Y, "@@.0")
       'frmOutGraph(CProj).PSet (X2, Y2)
       'frmOutGraph(CProj).Print Format$(PF * 100, "00.0")
       'frmOutGraph(CProj).PSet (X2, Y2)


      Next X
     Next J
    Next I

    '''' Draw Profile in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
 * TPMX + Init1
      InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
 * TPMY + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (PF * TPP)
       Y2 = InitY + (Y * TPMY)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(ProfileColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
       'frmOutGraph(CProj).PSet (X1, Y1)
       'frmOutGraph(CProj).Print Format$(Y, "@@.0")
       'frmOutGraph(CProj).PSet (X2, Y2)
       'frmOutGraph(CProj).Print Format$(PF * 100, "00.0")
       'frmOutGraph(CProj).PSet (X2, Y2)
      Next Y
     Next J '820438
    Next I
    Close #21
End Sub

Sub Screen_Shear ()
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 53

```
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim v As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim TX1 As Single
    Dim TY1 As Single


    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    frmOutGraph(CProj).Cls
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 54

```
    MomentColor = 17
    frmOutGraph(CProj).PSet (3000, 250)
    frmOutGraph(CProj).Print "  SHEAR DIAGRAM   "
    frmOutGraph(CProj).PSet (3000, 350)
    frmOutGraph(CProj).Print "=================="

    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m = T
PMX Twips for X axis
    TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
TPMY Twips for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If

    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 55

```
     Next J
    Next I

    '''' Draw Shear in X axis
    frmOutGraph(CProj).ForeColor = Colors(34)
    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     Trans_BElement False, I
     Cal_FEndMoment False, I
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      If J = 1 Then
       TX1 = InitX
       TY1 = InitY + 150
       frmOutGraph(CProj).PSet (TX1, TY1)
       frmOutGraph(CProj).Print Format$(Cal_BShear(J, 0), "0.00")
      End If
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       v = Cal_BShear(J, X)
       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (v * TPM)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(MomentColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
      Next X
      TX1 = InitX + EleColRow(CProj, 0, I, J).Length * TPMX
      TY1 = InitY + 150
      frmOutGraph(CProj).PSet (TX1, TY1)
      frmOutGraph(CProj).Print Format$(Cal_BShear(J, EleColRow(CProj, 0, I
, J).Length), "0.00")
     Next J
    Next I

    frmOutGraph(CProj).ForeColor = Colors(39)
    '''' Draw Shear in Y axis
    For I = 1 To SCol(CProj)
     Trans_BElement True, I
     Cal_FEndMoment True, I
     For J = 1 To ColEle(I)
      InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
* TPMX + Init1
      InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
* TPMY + Init2
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      If J = 1 Then
       TX1 = InitX
       TY1 = InitY
       frmOutGraph(CProj).PSet (TX1, TY1)
       frmOutGraph(CProj).Print Format$(Cal_BShear(J, 0), "0.00")
      End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 56

```
      frmOutGraph(CProj).PSet (InitX, InitY), ProfileColor
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       v = Cal_BShear(J, Y)
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (v * TPM)
       Y2 = InitY + (Y * TPMY)
       frmOutGraph(CProj).Line -(X2, Y2), Colors(MomentColor)
       frmOutGraph(CProj).Line (X1, Y1)-(X2, Y2), Colors(MomentColor)
      Next Y
      TX1 = InitX
      TY1 = InitY + EleColRow(CProj, 1, I, J).Length * TPMY
      frmOutGraph(CProj).PSet (TX1, TY1)
      frmOutGraph(CProj).Print Format$(Cal_BShear(J, EleColRow(CProj, 1, I
, J).Length), "0.00")
     Next J '820438
    Next I
    frmOutGraph(CProj).ForeColor = Colors(34)
End Sub

Sub Screen_Text (Text As String)
    frmList(CProj).FontName = "Courier"
    frmList(CProj).FontSize = 7
    TextToScreen = TextToScreen + "  " + Text
    If Len(TextToScreen) > 30000 Then
     TextNum = TextNum + 1
     frmList(CProj).txtListCompute(TextNum).Text = TextToScreen
     TextToScreen = ""
    End If
End Sub

Sub Screen_Try ()
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    ''''end moment
    Dim MWL As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 57

```
    Dim MWR As Single
    Dim MSL As Single
    Dim MSR As Single
    Dim MUL As Single
    Dim MUR As Single

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim CR As String * 1
    Dim LF As String * 1
    CR = Chr$(13)
    LF = Chr$(10)
    Const ucm = " cm"
    Const um = " m"
    Const ukg = " kg"
    Const uksc = " ksc"
    Const uksm = " kg/m^2"
    Dim CNo As Integer
    Dim RNum As Integer

    'shear
    'Column
    Dim Vc0 As Single
    Dim Vn0 As Single
    'Drop
    Dim Vc1 As Single
    Dim Vn1 As Single
    'Capital
    Dim Vc2 As Single
    Dim Vn2 As Single
    'Diagnal
    Dim Vc3 As Single
    Dim Vn3 As Single

    ''''M design
    Dim MDL As Single
    Dim MDD As Single
    Dim MDR As Single
    'M design at column strip
    Dim MCL As Single
    Dim MCD As Single
    Dim MCR As Single
    'M design at middle strip
    Dim MML1 As Single
    Dim MMD1 As Single
    Dim MMR1 As Single
    Dim MML2 As Single
    Dim MMD2 As Single
    Dim MMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 58

```
    Dim MLB1  As Single
    Dim MMB1  As Single
    Dim MRB1  As Single
    Dim MLT1  As Single
    Dim MMT1  As Single
    Dim MRT1  As Single
    'Half middle strip 2
    Dim MLB2  As Single
    Dim MMB2  As Single
    Dim MRB2  As Single
    Dim MLT2  As Single
    Dim MMT2  As Single
    Dim MRT2  As Single
    'column strip
    Dim CLB  As Single
    Dim CMB  As Single
    Dim CRB  As Single
    Dim CLT  As Single
    Dim CMT  As Single
    Dim CRT  As Single

    ''''Reinforcement design
    Dim RMDL As Single
    Dim RMDD As Single
    Dim RMDR As Single
    'M design at column strip
    Dim RMCL As Single
    Dim RMCD As Single
    Dim RMCR As Single
    'M design at middle strip
    Dim RMML1 As Single
    Dim RMMD1 As Single
    Dim RMMR1 As Single
    Dim RMML2 As Single
    Dim RMMD2 As Single
    Dim RMMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim RMLB1  As Single
    Dim RMMB1  As Single
    Dim RMRB1  As Single
    Dim RMLT1  As Single
    Dim RMMT1  As Single
    Dim RMRT1  As Single
    'Half middle strip 2
    Dim RMLB2  As Single
    Dim RMMB2  As Single
    Dim RMRB2  As Single
    Dim RMLT2  As Single
    Dim RMMT2  As Single
    Dim RMRT2  As Single
    'column strip
    Dim RCLB  As Single
    Dim RCMB  As Single
    Dim RCRB  As Single
    Dim RCLT  As Single
    Dim RCMT  As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 59

```
    Dim RCRT  As Single

    'Dim TextNo As Integer
    'Dim Text0 As String
    'ReDim Text(100) As String
    'TextNo = 1

    Dim TextNo As Integer
    ReDim Text(100) As String
    TextNo = 1


    TB = Chr$(9)

    AddSect = .5


    '''' Init
    frmList(CProj).Cls
    Text0 = ""

    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
 As #20 Len = 4
    '''' Information
    Screen_Text CR + LF
    Screen_Text "=================================================" + C
R + LF
    Screen_Text "TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM  " + C
R + LF
    Screen_Text " KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG    " + C
R + LF
    Screen_Text "          CIVIL ENGINEERING DEPARTMENT              " + C
R + LF
    Screen_Text "     BY SIAM KUNKATE & SOMPHONG KONGKRAPAN          " + C
R + LF
    Screen_Text "          LISTING COMPUTATION CHECKING              " + C
R + LF
    Screen_Text "=================================================" + C
R + LF
    Screen_Text CR + LF
    Screen_Text CR + LF

    Screen_Text "    List of Flat Slab & Flat Plate Design" + CR + LF
    Screen_Text "Data       " + TB + "  : " + SData(CProj).DateData + "  ::
: Time      : " + SData(CProj).TimeData + CR + LF
    Screen_Text "File Name " + TB + "  : " + SData(CProj).FileName + CR +
LF
    Screen_Text "Project   " + TB + "  : " + SData(CProj).ProjectData + CR
 + LF
    Screen_Text "Engineer  " + TB + "  : " + SData(CProj).Engineer + CR +
LF
    Screen_Text "Address   " + TB + "  : " + SData(CProj).AddressData + CR
 + LF
    Screen_Text "Information" + TB + " : " + SData(CProj).InformData + CR
+ LF

    '''' Data for Design
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 60

```
    Screen_Text "Length Unit    " + TB + " : Meter(m)  :::  Load Unit  : Ki
logram(Kg)" + CR + LF
    Screen_Text "fc' (Slab)     " + TB + " : " + Format$(GeneralData(CProj)
.SFc_, ".00") + uksc + CR + LF
    Screen_Text "fc' (Column)   " + TB + " : " + Format$(GeneralData(CProj)
.CFc_, ".00") + uksc + CR + LF
    Screen_Text "fy (Strand)    " + TB + " : " + Format$(GeneralData(CProj)
.SFs_, ".00") + uksc + CR + LF
    Select Case StrandIndex(CProj):
    Case 0:
    Case 1:
    Case 2:
    Case 3:
    Case 4:
    Case 5:
    Case 6:
    Case 7:
    End Select
    'Screen_Text "Es (Strand)     : " + Format$(Es, ".00") + CR + LF
    Screen_Text "Ec (Slab)       : " + Format$(GeneralData(CProj).SEc, ".00
") + uksc + CR + LF
    Screen_Text "Ec (Column)     : " + Format$(GeneralData(CProj).CEc, ".00
") + uksc + CR + LF
    Screen_Text "Slab Thickness : " + Format$(100 * Thick(CProj), ".00") +
 ucm + CR + LF
    Screen_Text "Slab Covering  : " + Format$((Thick(CProj) - d(CProj)) *
100, ".00") + ucm + CR + LF
    Screen_Text "Drape          : " + Format$((Thick(CProj) - 2 * (Thick(C
Proj) - d(CProj))) * 100, ".00") + ucm + CR + LF


    '''' Column Data
    frmList(CProj).FontSize = 7
    frmList(CProj).FontBold = True
    Screen_Text "Column Data" + CR + LF
    Screen_Text CR + LF
    Screen_Text "Column  No." + TB + "Co_X(m)" + TB + TB + "Co_Y(m)" + TB
+ "SupportType" + TB + " Length(m) " + CR + LF
    Screen_Text CR + LF
    CNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CNo = CNo + 1
      Co_X = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X
      Co_Y = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y
      Support = Column(CProj, EleColRow(CProj, 0, I, J).Start).SupportType
      Length = Column(CProj, EleColRow(CProj, 0, I, J).Start).Length
      Screen_Text Format$(CNo, "@@@") + TB + TB + Format$(Co_X, "0.00E+00"
) + TB + TB + Format$(Co_Y, "0.00E+00") + TB + TB + Format$(Support, "@@")
 + TB + Format$(Length, "0.00E+00") + CR + LF
      If J = RowEle(I) Then
       CNo = CNo + 1
       Co_X = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_X
       Co_Y = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_Y
       Support = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Supp
ortType
       Length = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Lengt
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 61

```
h
        Screen_Text Format$(CNo, "@@@") + TB + TB + Format$(Co_X, "0.00E+00
") + TB + TB + Format$(Co_Y, "0.00E+00") + TB + TB + Format$(Support, "@@"
) + TB + Format$(Length, "0.00E+00") + CR + LF
       End If
      Next J
     Next I

     '''' Element  Data
     Dim LCol As Integer
     Dim RCol As Integer
     Dim LBay As Single
     Dim RBay As Single
     'Row
     Screen_Text CR + LF
     Screen_Text CR + LF
     Screen_Text "Row Element Data" + CR + LF
     Screen_Text CR + LF
     Screen_Text " Row  :Element  No.:Start Column: End  Column:   Length
: Top  Bay   : Bottom Bay  " + CR + LF
     Screen_Text "                                                      (m)
     (m)           (m)   " + CR + LF
     For I = 1 To SRow(CProj)
      For J = 1 To RowEle(I)
       LCol = EleColRow(CProj, 0, I, J).Start
       RCol = EleColRow(CProj, 0, I, J).End
       Length = EleColRow(CProj, 0, I, J).Length
       LBay = EleColRow(CProj, 0, I, J).L1
       RBay = EleColRow(CProj, 0, I, J).L2
       Screen_Text Format$(I, "@@@@@@@@") + Format$(J, "@@@@@@@@@@@@@@@@@@@
") + Format$(LCol, "@@@@@@@@@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@@@@@@@@
@@") + Format$(Length, "@@@@@@@@@@@.00") + Format$(LBay, "@@@@@@@@@@@.00")
 + Format$(RBay, "@@@@@@@@@@@.00") + CR + LF
      Next J
     Next I
     'Column
     Screen_Text CR + LF
     Screen_Text CR + LF
     Screen_Text "Row Element Data" + CR + LF
     Screen_Text CR + LF
     Screen_Text "Column:Element  No.:Start Column: End  Column:   Length
: Left Bay   : Right  Bay  " + CR + LF
     Screen_Text "                                                      (m)
     (m)           (m)   " + CR + LF
     For I = 1 To SCol(CProj)
      For J = 1 To ColEle(I)
       LCol = EleColRow(CProj, 1, I, J).Start
       RCol = EleColRow(CProj, 1, I, J).End
       Length = EleColRow(CProj, 1, I, J).Length
       LBay = EleColRow(CProj, 1, I, J).L1
       RBay = EleColRow(CProj, 1, I, J).L2
       Screen_Text Format$(I, "@@@@@@@@") + Format$(J, "@@@@@@@@@@@@@@@@@@@
") + Format$(LCol, "@@@@@@@@@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@@@@@@@@
@@") + Format$(Length, "@@@@@@@@@@@.00") + Format$(LBay, "@@@@@@@@@@@.00")
 + Format$(RBay, "@@@@@@@@@@@.00") + CR + LF
      Next J
     Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 62

```
    Screen_Text CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF
    Dim SSystem As String * 50
    Select Case GeneralData(CProj).SlabIndex
    Case 0:
     SSystem = "Flat Plate"
    Case 1:
     SSystem = "Flat Slab With Drop Panel"
    Case 2:
     SSystem = "Flat Slab With Column Capital"
    Case 3:
     SSystem = "Flat Slab With Column Drop Panel And Column Capital"
    End Select
    Screen_Text "SLAB SYSTEM : " + SSystem + CR + LF
    Screen_Text CR + LF
    Screen_Text "Load factor ::: " + CR + LF
    Screen_Text "Wu = " + Format$(DeadCo, "0.0") + "DL" + Format$(LiveCo,
"0.0") + "LL" + CR + LF
    Screen_Text "Unfactored Dead Load           = " + Format$(DeadLoad(CPr
oj), "0.00") + uksm + CR + LF
    Screen_Text "Unfactored Live Load           = " + Format$(LiveLoad(CPr
oj), "0.00") + uksm + CR + LF
    Screen_Text "Unfactored Supper imposed load = " + Format$(SuperLoad(CP
roj), "0.00") + uksm + CR + LF
    Screen_Text ""

    Screen_Text "Column " + Chr$(9) + "      Punching at Column    " + Chr$
(9) + Chr$(9) + "          Punching at Drop Panel    " + Chr$(9) + "
 Punching at Capital       " + Chr$(9) + "              Diagonal shear
      " + CR + LF
    Screen_Text " No.   " + Chr$(9) + "  Vc(kg)  " + Chr$(9) + Chr$(9) + "
 Vn(kg)  " + Chr$(9) + Chr$(9) + " Vc(kg)  " + Chr$(9) + Chr$(9) + " Vn(kg
) " + Chr$(9) + Chr$(9) + " Vc(kg) " + Chr$(9) + Chr$(9) + "  Vn(kg)  " +
Chr$(9) + Chr$(9) + "  Vc(kg)  " + Chr$(9) + Chr$(9) + "  Vn(kg)  " + Chr$
(9) + CR + LF
    Screen_Text CR + LF
    RNum = 2
    RIden = 2
    Dim ColumnNo As Integer
    ColumnNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      ColumnNo = ColumnNo + 1
      CNo = EleColRow(CProj, 0, I, J).Start
      If Column(CProj, CNo).SupportType <> 2 Then 'If Not Free,Mean That i
s Column support
       RIden = RNum + ((CNo - 1) * 8)
       Get #20, RIden + 1, Vc0
       Get #20, RIden + 2, Vn0
       Get #20, RIden + 3, Vc1
       Get #20, RIden + 4, Vn1
       Get #20, RIden + 5, Vc2
       Get #20, RIden + 6, Vn2
       Get #20, RIden + 7, Vc3
       Get #20, RIden + 8, Vn3
       Screen_Text Format$(ColumnNo, "@@@@") + Chr$(9) + Format$(Vc0, "0.0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 63

```
000E+00") + Chr$(9) + Format$(Vn0, "0.0000E+00") + Chr$(9) + Format$(Vc1,
"0.0000E+00") + Chr$(9) + Format$(Vn1, "0.0000E+00") + Chr$(9) + Format$(V
c2, "0.0000E+00") + Chr$(9) + Format$(Vn2, "0.0000E+00") + Chr$(9) + Forma
t$(Vc3, "0.0000E+00") + Chr$(9) + Format$(Vn3, "0.0000E+00") + CR + LF
       End If
       If J = RowEle(I) Then
        ColumnNo = ColumnNo + 1
        CNo = EleColRow(CProj, 0, I, J).End
        If Column(CProj, CNo).SupportType <> 2 Then
         RIden = RNum + ((CNo - 1) * 8)
         Get #20, RIden + 1, Vc0
         Get #20, RIden + 2, Vn0
         Get #20, RIden + 3, Vc1
         Get #20, RIden + 4, Vn1
         Get #20, RIden + 5, Vc2
         Get #20, RIden + 6, Vn2
         Get #20, RIden + 7, Vc3
         Get #20, RIden + 8, Vn3
         Screen_Text Format$(ColumnNo, "@@@@") + Chr$(9) + Format$(Vc0, "0.
0000E+00") + Chr$(9) + Format$(Vn0, "0.0000E+00") + Chr$(9) + Format$(Vc1,
 "0.0000E+00") + Chr$(9) + Format$(Vn1, "0.0000E+00") + Chr$(9) + Format$(
Vc2, "0.0000E+00") + Chr$(9) + Format$(Vn2, "0.0000E+00") + Chr$(9) + Form
at$(Vc3, "0.0000E+00") + Chr$(9) + Format$(Vn3, "0.0000E+00") + CR + LF
        End If
       End If
      Next J
     Next I
     'Drop panel & Capital
     CNo = 0
     ColumnNo = 0
     Screen_Text CR + LF
     Screen_Text "Dimension of drop & column capital"
     Screen_Text CR + LF
     Screen_Text " No. " + Chr$(9) + "Drop Width" + Chr$(9) + "Drop Length"
 + Chr$(9) + "Drop Thick" + Chr$(9) + "Capital Width" + Chr$(9) + "Capital
 Length" + Chr$(9) + CR + LF
     Screen_Text CR + LF
     For I = 1 To SRow(CProj)
      For J = 1 To RowEle(I)
       CNo = CNo + 1
       ColumnNo = EleColRow(CProj, 0, I, J).Start
       DropX = Column(CProj, ColumnNo).DropX
       DropY = Column(CProj, ColumnNo).DropY
       DropThick = Column(CProj, ColumnNo).DropThick
       CapitalX = Column(CProj, ColumnNo).CapitalX
       CapitalY = Column(CProj, ColumnNo).CapitalY
       Screen_Text Format$(CNo, "@@@@") + Chr$(9) + Format$(DropX, "0.0000E
+00") + Chr$(9) + Format$(DropY, "0.0000E+00") + Chr$(9) + Format$(DropThi
ck, "0.0000E+00") + Chr$(9) + Format$(CapitalX, "0.0000E+00") + Chr$(9) +
Format$(CapitalY, "0.0000E+00") + CR + LF
       If J = RowEle(I) Then
        CNo = CNo + 1
        ColumnNo = EleColRow(CProj, 0, I, RowEle(I)).End
        DropX = Column(CProj, ColumnNo).DropX
        DropY = Column(CProj, ColumnNo).DropY
        DropThick = Column(CProj, ColumnNo).DropThick
        CapitalX = Column(CProj, ColumnNo).CapitalX
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 64

```
        CapitalY = Column(CProj, ColumnNo).CapitalY
        Screen_Text Format$(CNo, "@@@@") + Chr$(9) + Format$(DropX, "0.0000
E+00") + Chr$(9) + Format$(DropY, "0.0000E+00") + Chr$(9) + Format$(DropTh
ick, "0.0000E+00") + Chr$(9) + Format$(CapitalX, "0.0000E+00") + Chr$(9) +
 Format$(CapitalY, "0.0000E+00") + CR + LF
       End If
      Next J
     Next I
     'declare
     Dim Lmax As Single
     Dim Bbeam As Single
     Dim P As Single
     Dim Top As Single
     Dim Bottom As Single
     TB = Chr$(9)


     Text1 = ""
     Screen_Text CR + LF
     Screen_Text "==========================================================" + C
R + LF
     Screen_Text "CHECK CRACKING MOMENT & ULTIMATE MOMENT & DEFLECTION" + C
R + LF
     Screen_Text "==========================================================" + C
R + LF
     Screen_Text CR + LF
     Screen_Text CR + LF
     ''''check Cracking Moment & Deflection
    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj) + DeadCo * Su
perLoad(CProj)
     'text flexeral stress in row
     Lmax = 0
     Bbeam = 0
     Mu = 0
     For I = 1 To SRow(CProj)
      Trans_UniformLoad Wu
      Trans_BElement False, I
      Cal_UltimateEndMoment False, I
      Mu = Find_MuMax(False, I)
      Find_BL False, I, Lmax, Bbeam
      P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
      Screen_Text CR + LF
      Screen_Text " ROW : " + Format$(I, "@@@") + CR + LF
      Screen_Text " No of tendon = " + Format$(Find_AsStrand(Lmax, Bbeam),
"@@@") + TB + "Unit" + CR + LF
      Screen_Text " Tension per 1 unit tendon = " + Format$(.775 * GeneralD
ata(CProj).SFs_, "0.00") + " Kg" + CR + LF
      Screen_Text " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), "0.00")
+ " kg-m" + CR + LF
      Screen_Text " Mn  = " + Format$(Check_MnMax(Find_AsStrand(Lmax, Bbeam
), Bbeam), "0.00") + " kg-m" + CR + LF
      Screen_Text " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
      Screen_Text "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
      Screen_Text " Deflection(Actual)     = " + Format$(Find_MaxDeflection(
False, I, 1), "0.00E+00") + um + CR + LF
      Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Fa
lse, I), "0.00E+00") + um + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 65

```
     Screen_Text "CHECK DEFLECTION AT USED" + CR + LF
     Screen_Text " Deflection(Actual)     = " + Format$(Find_MaxDeflection(
False, I, 2), "0.00E+00") + um + CR + LF
     Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Fa
lse, I), "0.00E+00") + um + CR + LF
     Screen_Text "==========================================================
==================================" + CR + LF
    Next I

    'Textlflexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_UltimateEndMoment True, I
     Mu = Find_MuMax(True, I)
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text " COLUMN : " + Format$(I, "@@@") + CR + LF
     Screen_Text " No of tendon = " + Format$(Find_AsStrand(Lmax, Bbeam),
"@@@") + TB + "Unit" + CR + LF
     Screen_Text " Tension per 1 unit tendon = " + Format$(.775 * GeneralD
ata(CProj).SFs_, ".00") + " Kg" + CR + LF
     Screen_Text " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), ".00") +
 " kg-m" + CR + LF
     Screen_Text " Mn  = " + Format$(Check_MnMax(Find_AsStrand(Lmax, Bbeam
), Bbeam), ".00") + " Kg-m" + CR + LF
     Screen_Text " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
     Screen_Text "CHECK DEFLECTION AT USED" + CR + LF
     Screen_Text " Deflection(Actual)     = " + Format$(Find_MaxDeflection(
True, I, 1), "0.00E+00") + um + CR + LF
     Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Tr
ue, I), "0.00E+00") + um + CR + LF
     Screen_Text "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
     Screen_Text " Deflection(Actual)     = " + Format$(Find_MaxDeflection(
True, I, 2), "0.00E+00") + um + CR + LF
     Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Tr
ue, I), "0.00E+00") + um + CR + LF
     Screen_Text "==========================================================
==================================" + CR + LF
    Next I

    '''''at TRANSFER load
    Text2 = ""
    Screen_Text CR + LF
    Screen_Text "=====================================================" +
CR + LF
    Screen_Text "CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD" +
CR + LF
    Screen_Text "=====================================================" +
CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 66

```
    Wu = DeadLoad(CProj)
    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ElasticEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text "Row No.  " + "  Element No." + TB + "Distance From Left"
 + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR + LF
     Screen_Text "       " + TB + TB + "           " + TB + TB + TB + TB +
 TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
     Screen_Text CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") + T
B + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00")
 + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 100
00, "0.00E+00") + CR + LF
      Next X
      Screen_Text "=================================================
==================================" + CR + LF
     Next J
    Next I

    'Text2 flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_ElasticEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text "Column No." + "  Element No." + TB + "Distance From Left
" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR + L
F
     Screen_Text "       " + TB + TB + "           " + TB + TB + TB + TB +
 TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") + T
B + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00")
 + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 67

```
00, "0.00E+00") + CR + LF
      Next X
      Screen_Text "=============================================================
=====================================" + CR + LF
     Next J
    Next I

    AddSect = .5
    ''''check at SURVICE load
    Text3 = ""
    Screen_Text CR + LF
    Screen_Text "=========================================================" +
CR + LF
    Screen_Text "CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD" +
CR + LF
    Screen_Text "=========================================================" +
CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF

    Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    'Text3 flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ServiceEndMoment False, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text "Row No.    " + "  Element No." + TB + "Distance From Lef
t" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR +
LF
     Screen_Text "          " + TB + TB + "                " + TB + TB + TB + TB +
 TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
     Screen_Text CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") + T
B + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00")
 + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 100
00, "0.00E+00") + CR + LF
      Next X
      Screen_Text "=============================================================
=====================================" + CR + LF
     Next J
    Next I

    'Text3 flexeral stress in Column
    Lmax = 0
    Bbeam = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

.OUTPUT.BAS - 68

```
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_ServiceEndMoment False, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text "Column No." + "  Element No." + TB + "Distance From Left
" + TB + " Position of tendon" + TB + TB + TB + "Checking Stress" + CR + L
F
     Screen_Text "        " + TB + TB + "            " + TB + TB + TB + TB +
 TB + TB + TB + "Top" + TB + TB + "Bottom" + CR + LF
     Screen_Text CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@@") + TB + TB + Format$(J, "@@@@@") + T
B + TB + Format$(X, "0.00E+00") + TB + TB + Format$(PF * 100, "0.000E+00")
 + TB + TB + Format$(Top / 10000, "0.000E+00") + TB + Format$(Bottom / 100
00, "0.00E+00") + CR + LF
      Next X
      Screen_Text "=============================================================
=====================================" + CR + LF
     Next J
    Next I

    For I = 1 To 20
     frmList(CProj).txtListCompute(I).Top = 0
     frmList(CProj).txtListCompute(I).Left = 0
     frmList(CProj).txtListCompute(I).Width = frmList(CProj).ScaleWidth
     frmList(CProj).txtListCompute(I).Height = frmList(CProj).ScaleHeight
     frmList(CProj).txtListCompute(I).Visible = False
    Next I
    frmList(CProj).txtListCompute(TextNum + 1).Text = TextToScreen
    TextToScreen = ""
    TextModule = TextNum + 1
    TextNum = 0
    CurrentPage = 1
    frmList(CProj).txtListCompute(1).Visible = True
    frmMain.cmdPrevious.Visible = True
    frmMain.cmdNext.Visible = True
    If TextModule = 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = False
    ElseIf TextModule > 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = True
    End If
    Close #20
End Sub

Sub Sum_Text (Text0 As String, TextNo As Integer, TextAdd As String)
    On Error GoTo ErrorText
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUT.BAS - 69

```
    Text0 = Text0 + "   " + TextAdd
    If Len(Text0) > 30000 Then
     TextNo = TextNo + 1
    End If

    Exit Sub
ErrorText: For I = 1 To TextNo
            frmList(CProj).txtListCompute(I).Text = Text0
           Next I
           Exit Sub
End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SYSTEMSL.BAS - 1

```
Option Explicit


Function Cal_Thick (L As Single, Index As Integer) As Single
    Dim T As Single
    Dim tb As Single
    Dim Lb As Single'beam

    Select Case Index
    Case 0: 'Flat Plate
     T = L / 42
    Case 1: 'Panel
     T = L / 45
    Case 2: 'Capital
     T = L / 42
    Case 3: 'Both
     T = L / 42
    Case 4: 'Band Beam
     T = L / 52
     tb = Lb / 28
    End Select
    Dim x As Integer
    T = T * 100
    x = Int(T)
    If T - x > 0 Then
     T = x + 1  '»Ñ´¢Öé¹
    ElseIf T - x = 0 Then
     T = x
    End If
    T = T / 100

    Cal_Thick = T
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 1

```
Sub Print_ListCompute ()
    frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
    frmMain.cmdlgSystem.CancelError = True
    On Error GoTo DlgErrorListCOmpute
    frmMain.cmdlgSystem.Action = 5

    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    ''''end moment
    Dim MWL As Single
    Dim MWR As Single
    Dim MSL As Single
    Dim MSR As Single
    Dim MUL As Single
    Dim MUR As Single

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim CR As String * 1
    Dim LF As String * 1
    CR = Chr$(13)
    LF = Chr$(10)
    Const ucm = " cm"
    Const um = " m"
    Const ukg = " kg"
    Const uksc = " ksc"
    Const uksm = " kg/m^2"
    Dim CNo As Integer
    Dim RNum As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    'shear
    'Column
    Dim Vc0 As Single
    Dim Vn0 As Single
    'Drop
    Dim Vc1 As Single
    Dim Vn1 As Single
    'Capital
    Dim Vc2 As Single
    Dim Vn2 As Single
    'Diagnal
    Dim Vc3 As Single
    Dim Vn3 As Single

    ''''M design
    Dim MDL As Single
    Dim MDD As Single
    Dim MDR As Single
    'M design at column strip
    Dim MCL As Single
    Dim MCD As Single
    Dim MCR As Single
    'M design at middle strip
    Dim MML1 As Single
    Dim MMD1 As Single
    Dim MMR1 As Single
    Dim MML2 As Single
    Dim MMD2 As Single
    Dim MMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim MLB1  As Single
    Dim MMB1  As Single
    Dim MRB1  As Single
    Dim MLT1  As Single
    Dim MMT1  As Single
    Dim MRT1  As Single
    'Half middle strip 2
    Dim MLB2  As Single
    Dim MMB2  As Single
    Dim MRB2  As Single
    Dim MLT2  As Single
    Dim MMT2  As Single
    Dim MRT2  As Single
    'column strip
    Dim CLB  As Single
    Dim CMB  As Single
    Dim CRB  As Single
    Dim CLT  As Single
    Dim CMT  As Single
    Dim CRT  As Single

    ''''Reinforcement design
    Dim RMDL As Single
    Dim RMDD As Single
    Dim RMDR As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 3

```
    'M design at column strip
    Dim RMCL As Single
    Dim RMCD As Single
    Dim RMCR As Single
    'M design at middle strip
    Dim RMML1 As Single
    Dim RMMD1 As Single
    Dim RMMR1 As Single
    Dim RMML2 As Single
    Dim RMMD2 As Single
    Dim RMMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim RMLB1  As Single
    Dim RMMB1  As Single
    Dim RMRB1  As Single
    Dim RMLT1  As Single
    Dim RMMT1  As Single
    Dim RMRT1  As Single
    'Half middle strip 2
    Dim RMLB2  As Single
    Dim RMMB2  As Single
    Dim RMRB2  As Single
    Dim RMLT2  As Single
    Dim RMMT2  As Single
    Dim RMRT2  As Single
    'column strip
    Dim RCLB  As Single
    Dim RCMB  As Single
    Dim RCRB  As Single
    Dim RCLT  As Single
    Dim RCMT  As Single
    Dim RCRT  As Single

    'Dim TextNo As Integer
    'Dim Text0 As String
    'ReDim Text(100) As String
    'TextNo = 1

    Dim TextNo As Integer
    ReDim Text(100) As String
    TextNo = 1


    TB = Chr$(9)

    AddSect = .5


    '''' Init
    frmList(CProj).Cls

    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
 As #20 Len = 4
    '''' Information
    Print_Text 0, CR + LF
    Print_Text 0, "=====================================================" +
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 4

```
 CR + LF
    Print_Text 0, "TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM  " +
 CR + LF
    Print_Text 0, " KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG    " +
 CR + LF
    Print_Text 0, "          CIVIL ENGINEERING DEPARTMENT              " +
 CR + LF
    Print_Text 0, "     BY SIAM KUNKATE & SOMPHONG KONGKRAPAN          " +
 CR + LF
    Print_Text 0, "           LISTING COMPUTATION CHECKING             " +
 CR + LF
    Print_Text 0, "==================================================" +
 CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF

    Print_Text 0, "    LIST OF PRE-STRESSED DESIGN          " + CR + LF
    Print_Text 0, "Date         : " + Format$(Now, "dddd mm/dd/yyyy ") + CR
 + LF
    Print_Text 0, "Time         : " + Format$(Now, "hh:mm:ss AM/PM") + CR +
 LF
    Print_Text 0, "File Name    : " + SData(CProj).FileName + CR + LF
    Print_Text 0, "Project      : " + SData(CProj).ProjectData + CR + LF
    Print_Text 0, "Engineer     : " + SData(CProj).Engineer + CR + LF
    Print_Text 0, "Address      : " + SData(CProj).AddressData + CR + LF
    Print_Text 0, "Information  : " + SData(CProj).InformData + CR + LF

    '''' Data for Design
    Print_Text 0, "Length Unit : Meter(m)" + CR + LF
    Print_Text 0, "Load Unit   : Kilogram(Kg)" + CR + LF
    Print_Text 0, "fc' (Slab)  : " + Format$(GeneralData(CProj).SFc_, ".00
") + uksc + CR + LF
    Print_Text 0, "fc' (Column): " + Format$(GeneralData(CProj).CFc_, ".00
") + uksc + CR + LF
    Print_Text 0, "fy (Strand) : " + Format$(GeneralData(CProj).SFs_, ".00
") + uksc + CR + LF
    Select Case StrandIndex(CProj):
    Case 0:
    Case 1:
    Case 2:
    Case 3:
    Case 4:
    Case 5:
    Case 6:
    Case 7:
    End Select
    'Print_Text 0, "Es (Strand)     : " + Format$(Es, ".00") + CR + LF
    Print_Text 0, "Ec (Slab)       : " + Format$(GeneralData(CProj).SEc, ".
00") + uksc + CR + LF
    Print_Text 0, "Ec (Column)     : " + Format$(GeneralData(CProj).CEc, ".
00") + uksc + CR + LF
    Print_Text 0, "Slab Thickness : " + Format$(100 * Thick(CProj), ".00")
 + ucm + CR + LF
    Print_Text 0, "Slab Covering  : " + Format$((Thick(CProj) - d(CProj))
* 100, ".00") + ucm + CR + LF
    Print_Text 0, "Drape          : " + Format$((Thick(CProj) - 2 * (Thick
(CProj) - d(CProj))) * 100, ".00") + ucm + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 5

```
    '''' Column Data
    'frmList(CProj).FontSize = 12
    frmList(CProj).FontBold = True
    Print_Text 0, CR + LF
    Print_Text 0, "COLUMN DATA" + CR + LF
    Print_Text 0, "===========" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "COLUMN NO.  CO_X(m)      CO_Y(m)      LENGTH(m)    " + CR
+ LF
    Print_Text 0, CR + LF
    CNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CNo = CNo + 1
      Co_X = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X
      Co_Y = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y
      Support = Column(CProj, EleColRow(CProj, 0, I, J).Start).SupportType
      Length = Column(CProj, EleColRow(CProj, 0, I, J).Start).Length
      Print_Text 1, Format$(CNo, "@@@@@@@@") + s5 + Format$(Co_X, "0.00E+0"
) + s5 + Format$(Co_Y, "0.00E+0") + s5 + Format$(Length, "0.00E+0") + CR +
 LF
      If J = RowEle(I) Then
       CNo = CNo + 1
       Co_X = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_X
       Co_Y = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_Y
       Support = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Supp
ortType
       Length = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Lengt
h
       Print_Text 1, Format$(CNo, "@@@@@@@@") + s5 + Format$(Co_X, "0.00E+0
") + s5 + Format$(Co_Y, "0.00E+0") + s5 + Format$(Length, "0.00E+0") + CR
+ LF
      End If
     Next J
    Next I
    Print_Text 0, "=============================================" + CR + L
F


    '''' Element  Data
    Dim LCol As Integer
    Dim RCol As Integer
    Dim LBay As Single
    Dim RBay As Single
    'Row
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "DESIGN STRIP IN ROW" + CR + LF
    Print_Text 0, "===================" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, " ROW   ELEMENT   START-COL  END-COL  LENGTH(m)   TOP-BA
Y(m) BOTTOM-BAY(m)" + CR + LF
    Print_Text 0, CR + LF
    For I = 1 To SRow(CProj)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 6

```
      For J = 1 To RowEle(I)
       LCol = EleColRow(CProj, 0, I, J).Start
       RCol = EleColRow(CProj, 0, I, J).End
       Length = EleColRow(CProj, 0, I, J).Length
       LBay = EleColRow(CProj, 0, I, J).L1
       RBay = EleColRow(CProj, 0, I, J).L2
       Print_Text 2, Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + Format$(
LCol, "@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@") + s5 + Format$(Length, "
0.00E+0") + s5 + Format$(LBay, "0.00E+0") + s5 + s2 + Format$(RBay, "0.00E
+0") + CR + LF
      Next J
     Next I
     Print_Text 0, "=============================================================
==================" + CR + LF
     'Column
     Print_Text 0, CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, "DESIGN STRIP IN COLUMN" + CR + LF
     Print_Text 0, "======================" + CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, "COLUMN ELEMENT   START-COL  END-COL  LENGTH(m)   TOP-BA
Y(m) BOTTOM-BAY(m)" + CR + LF
     Print_Text 0, CR + LF
     For I = 1 To SCol(CProj)
      For J = 1 To ColEle(I)
       LCol = EleColRow(CProj, 1, I, J).Start
       RCol = EleColRow(CProj, 1, I, J).End
       Length = EleColRow(CProj, 1, I, J).Length
       LBay = EleColRow(CProj, 1, I, J).L1
       RBay = EleColRow(CProj, 1, I, J).L2
       Print_Text 3, Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + Format$(
LCol, "@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@") + s5 + Format$(Length, "
0.00E+0") + s5 + Format$(LBay, "0.00E+0") + s5 + s2 + Format$(RBay, "0.00E
+0") + CR + LF
      Next J
     Next I
     Print_Text 0, "=============================================================
==================" + CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, CR + LF
     Dim SSystem As String * 50
     Select Case GeneralData(CProj).SlabIndex
     Case 0:
      SSystem = "Flat Plate"
     Case 1:
      SSystem = "Flat Slab With Drop Panel"
     Case 2:
      SSystem = "Flat Slab With Column Capital"
     Case 3:
      SSystem = "Flat Slab With Column Drop Panel And Column Capital"
     End Select
     Print_Text 0, "SLAB SYSTEM : " + SSystem + CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, "Load factor ::: " + CR + LF
     Print_Text 0, "Wu = " + Format$(DeadCo, "0.0") + "DL" + Format$(LiveCo
, "0.0") + "LL" + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 7

```
    Print_Text 0, "Unfactored Dead Load           = " + Format$(DeadLoad(C
Proj), "0.00") + uksm + CR + LF
    Print_Text 0, "Unfactored Live Load           = " + Format$(LiveLoad(C
Proj), "0.00") + uksm + CR + LF
    Print_Text 0, "Unfactored Supper imposed load = " + Format$(SuperLoad(
CProj), "0.00") + uksm + CR + LF

    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "SHEAR CHECKING AT COLUMN,DROP PANEL,COLUMN CAPITAL AND
BEAM ACTION SHEAR" + CR + LF
    Print_Text 0, "=============================================================
================" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "COLUMN" + s10 + s10 + s5 + "PUNCHING SHEAR" + s10 + s10
 + s9 + "BEAM-ACTION-SHEAR" + CR + LF
    Print_Text 0, s2 + "NO." + s9 + "COLUMN" + s3 + s10 + "DROP-PANEL" + s
1 + s10 + "COLUMN-CAPITAL" + CR + LF
    Print_Text 0, s8 + "Vc(kg)" + s5 + "Vn(kg)" + s5 + "Vc(kg)" + s5 + "Vn
(kg)" + s5 + "Vc(kg)" + s5 + "Vn(kg)" + s5 + "Vc(kg)" + s5 + "Vn(kg)" + CR
 + LF
    Print_Text 0, CR + LF
    RNum = 2
    RIden = 2
    Dim ColumnNo As Integer
    ColumnNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      ColumnNo = ColumnNo + 1
      CNo = EleColRow(CProj, 0, I, J).Start
      If Column(CProj, CNo).SupportType <> 2 Then 'If Not Free,Mean That i
s Column support
       RIden = RNum + ((CNo - 1) * 8)
       Get #20, RIden + 1, Vc0
       Get #20, RIden + 2, Vn0
       Get #20, RIden + 3, Vc1
       Get #20, RIden + 4, Vn1
       Get #20, RIden + 5, Vc2
       Get #20, RIden + 6, Vn2
       Get #20, RIden + 7, Vc3
       Get #20, RIden + 8, Vn3
       Print_Text 4, Format$(ColumnNo, "@@@@") + s3 + Format$(Vc0, "0.00E+
00") + s3 + Format$(Vn0, "0.00E+00") + s3 + Format$(Vc1, "0.00E+00") + s3
+ Format$(Vn1, "0.00E+00") + s3 + Format$(Vc2, "0.00E+00") + s3 + Format$(
Vn2, "0.00E+00") + s3 + Format$(Vc3, "0.00E+00") + s3 + Format$(Vn3, "0.00
E+00") + CR + LF
      End If
      If J = RowEle(I) Then
       ColumnNo = ColumnNo + 1
       CNo = EleColRow(CProj, 0, I, J).End
       If Column(CProj, CNo).SupportType <> 2 Then
        RIden = RNum + ((CNo - 1) * 8)
        Get #20, RIden + 1, Vc0
        Get #20, RIden + 2, Vn0
        Get #20, RIden + 3, Vc1
        Get #20, RIden + 4, Vn1
        Get #20, RIden + 5, Vc2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 8

```
        Get #20, RIden + 6, Vn2
        Get #20, RIden + 7, Vc3
        Get #20, RIden + 8, Vn3
        Print_Text 4, Format$(ColumnNo, "@@@@") + s3 + Format$(Vc0, "0.00E
+00") + s3 + Format$(Vn0, "0.00E+00") + s3 + Format$(Vc1, "0.00E+00") + s3
 + Format$(Vn1, "0.00E+00") + s3 + Format$(Vc2, "0.00E+00") + s3 + Format$
(Vn2, "0.00E+00") + s3 + Format$(Vc3, "0.00E+00") + s3 + Format$(Vn3, "0.0
0E+00") + CR + LF

       End If
      End If
     Next J
    Next I
    Print_Text 0, "=================================================================
=====================================" + CR + LF
    'Drop panel & Capital
   If GeneralData(CProj).SlabIndex <> 0 Then
    CNo = 0
    ColumnNo = 0
    Print_Text 0, CR + LF
    Print_Text 0, "=================================================================
===========" + CR + LF
    Print_Text 0, "                    DIMENSION OF DROP PANEL & COLUMN CAPITAL
          " + CR + LF
    Print_Text 0, "
          " + CR + LF
    Print_Text 0, "=================================================================
===========" + CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, s2 + "COLUMN" + s2 + "DROP WIDTH(m)" + s2 + "DROP LENGTH
(m)" + s2 + "DROP THICK(m)" + s2 + "CAPITAL WIDTH(m)" + s2 + "CAPITAL LENG
TH(m)" + CR + LF
    Print_Text 0, CR + LF
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CNo = CNo + 1
      ColumnNo = EleColRow(CProj, 0, I, J).Start
      DropX = Column(CProj, ColumnNo).DropX
      DropY = Column(CProj, ColumnNo).DropY
      DropThick = Column(CProj, ColumnNo).DropThick
      CapitalX = Column(CProj, ColumnNo).CapitalX
      CapitalY = Column(CProj, ColumnNo).CapitalY
      Print_Text 5, Format$(CNo, "@@@@@@") + s6 + Format$(DropX, "0.00E+00
") + s7 + Format$(DropY, "0.00E+00") + s8 + Format$(DropThick, "0.00E+00")
 + s9 + Format$(CapitalX, "0.00E+00") + s10 + Format$(CapitalY, "0.00E+00"
) + CR + LF
      If J = RowEle(I) Then
       CNo = CNo + 1
       ColumnNo = EleColRow(CProj, 0, I, RowEle(I)).End
       DropX = Column(CProj, ColumnNo).DropX
       DropY = Column(CProj, ColumnNo).DropY
       DropThick = Column(CProj, ColumnNo).DropThick
       CapitalX = Column(CProj, ColumnNo).CapitalX
       CapitalY = Column(CProj, ColumnNo).CapitalY
       Print_Text 5, Format$(CNo, "@@@@@@") + s6 + Format$(DropX, "0.00E+0
0") + s7 + Format$(DropY, "0.00E+00") + s8 + Format$(DropThick, "0.00E+00"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 9

```
) + s9 + Format$(CapitalX, "0.00E+00") + s10 + Format$(CapitalY, "0.00E+00
") + CR + LF
      End If
     Next J
    Next I
    Print_Text 0, "=================================================================
==================================" + CR + LF
   ElseIf GeneralData(CProj).SlabIndex = 0 Then

   End If
    'declare
    Dim Lmax As Single
    Dim Bbeam As Single
    Dim P As Single
    Dim Top As Single
    Dim Bottom As Single
    TB = Chr$(9)


    Text1 = ""
    Print_Text 0, CR + LF
    Print_Text 0, "=================================================" +
 CR + LF
    Print_Text 0, "CHECK CRACKING MOMENT & ULTIMATE MOMENT & DEFLECTION" +
 CR + LF
    Print_Text 0, "=================================================" +
 CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    ''''check Cracking Moment & Deflection
    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj) + DeadCo * Su
perLoad(CProj)
    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_UltimateEndMoment False, I
     Mu = Find_MuMax(False, I)
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, " ROW : " + Format$(I, "@@@") + CR + LF
     Print_Text 0, " ==========" + CR + LF
     Print_Text 0, " No of tendon              = " + Format$(Find_AsStrand
(Lmax, Bbeam), "@@@") + TB + "Unit" + CR + LF
     Print_Text 0, " Tension per 1 unit tendon = " + Format$(.775 * Genera
lData(CProj).SFs_, "0.00") + " Kg" + CR + LF
     Print_Text 0, " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), "0.00"
) + " kg-m" + CR + LF
     Print_Text 0, " Mn  = " + Format$(Check_MnMax(Find_AsStrand(Lmax, Bbe
am), Bbeam), "0.00") + " kg-m" + CR + LF
     Print_Text 0, " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
     Print_Text 0, " Deflection(Actual)    = " + Format$(Find_MaxDeflectio
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 10

```
n(False, I, 1), "0.00E+00") + um + CR + LF
     Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
False, I), "0.00E+00") + um + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT USED" + CR + LF
     Print_Text 0, " Deflection(Actual)    = " + Format$(Find_MaxDeflectio
n(False, I, 2), "0.00E+00") + um + CR + LF
     Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
False, I), "0.00E+00") + um + CR + LF
     Print_Text 0, "=========================================" + CR + LF
    Next I

    'Text1flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_UltimateEndMoment True, I
     Mu = Find_MuMax(True, I)
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, " COLUMN : " + Format$(I, "@@@") + CR + LF
     Print_Text 0, " ============" + CR + LF
     Print_Text 0, " No of tendon                 = " + Format$(Find_AsStrand
(Lmax, Bbeam), "@@@") + TB + "Unit" + CR + LF
     Print_Text 0, " Tension per 1 unit tendon = " + Format$(.775 * Genera
lData(CProj).SFs_, ".00") + " Kg" + CR + LF
     Print_Text 0, " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), ".00")
 + " kg-m" + CR + LF
     Print_Text 0, " Mn  = " + Format$(Check_MnMax(Find_AsStrand(Lmax, Bbe
am), Bbeam), ".00") + " Kg-m" + CR + LF
     Print_Text 0, " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT USED" + CR + LF
     Print_Text 0, " Deflection(Actual)    = " + Format$(Find_MaxDeflectio
n(True, I, 1), "0.00E+00") + um + CR + LF
     Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
True, I), "0.00E+00") + um + CR + LF
     Print_Text 0, "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
     Print_Text 0, " Deflection(Actual)    = " + Format$(Find_MaxDeflectio
n(True, I, 2), "0.00E+00") + um + CR + LF
     Print_Text 0, " Deflection(Allowable) = " + Format$(Check_Deflection(
True, I), "0.00E+00") + um + CR + LF
     Print_Text 0, "=========================================" + CR + LF
    Next I

    '''''at TRANSFER load
    Text2 = ""
    Print_Text 0, CR + LF
    Print_Text 0, "======================================================"
+ CR + LF
    Print_Text 0, "CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD"
+ CR + LF
    Print_Text 0, "======================================================"
+ CR + LF
    Print_Text 0, CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"LPSTRESS" Present by Sompong Kongkrapun & Siam Kunkate

Mr. Wibool Wutthiyan : ADVISOR

OUTPUTP.BAS - 11

```
    Print_Text 0, CR + LF

    Wu = DeadLoad(CProj)
    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ElasticEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE FROM LEFT(
m)" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR
 + LF
     Print_Text 0, "
        TOP" + s10 + "BOTTOM" + CR + LF
     Print_Text 0, CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Print_Text 6, Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + s10 + s2
 + Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E
+00") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bo
ttom / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Print_Text 0, "==========================================================
==============================" + CR + LF
    Next J
   Next I
   Print_Text 0, "==========================================================
==============================" + CR + LF
   'Text2 flexeral stress in Column
   Lmax = 0
   Bbeam = 0
   Mu = 0
   For I = 1 To SRow(CProj)
    Trans_UniformLoad Wu
    Trans_BElement True, I
    Cal_ElasticEndMoment True, I
    Find_BL True, I, Lmax, Bbeam
    P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
    Print_Text 0, CR + LF
    Print_Text 0, "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FROM TOP(m)
" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR +
LF
    Print_Text 0, "
       TOP" + s10 + "BOTTOM" + CR + LF
    Print_Text 0, CR + LF
    For J = 1 To ColEle(I)
     For X = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
      M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 12

```
        PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
        Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 1, I, J).L2 / 2, Top, Bottom
        Print_Text 7, Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + s10 + s2
 + Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E
+00") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bo
ttom / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Print_Text 0, "=================================================
==============================" + CR + LF
     Next J
    Next I
    Print_Text 0, "=================================================
==============================" + CR + LF
    AddSect = .5
    ''''check at SURVICE load
    Text3 = ""
    Print_Text 0, CR + LF
    Print_Text 0, "======================================================"
+ CR + LF
    Print_Text 0, "CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD"
+ CR + LF
    Print_Text 0, "======================================================"
+ CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF

    Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    'Text3 flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ServiceEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE FROM LEFT(
m)" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR
 + LF
     Print_Text 0, "
        TOP" + s10 + "BOTTOM" + CR + LF
     Print_Text 0, CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
      Print_Text 6, Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + s10 + s2
+ Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+
00") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bot
tom / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Print_Text 0, "=================================================
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 13

```
==============================" + CR + LF
     Next J
    Next I
    Print_Text 0, "=============================================================
============================" + CR + LF
    'Text3 flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_ServiceEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FROM TOP(m)
" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR +
 LF
     Print_Text 0, "
        TOP" + s10 + "BOTTOM" + CR + LF
     Print_Text 0, CR + LF
     For J = 1 To ColEle(I)
      For X = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 1, I, J).L2 / 2, Top, Bottom
      Print_Text 7, Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + s10 + s2
+ Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+
00") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bot
tom / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Print_Text 0, "=============================================================
==============================" + CR + LF
     Next J
    Next I
    Print_Text 0, "=============================================================
============================" + CR + LF
    Printer.EndDoc
    Close #20
    Exit Sub
DlgErrorListCOmpute: Exit Sub
End Sub

Sub Print_ListProfile ()
    frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
    frmMain.cmdlgSystem.CancelError = True
    On Error GoTo DlgErrorListProfile
    frmMain.cmdlgSystem.Action = 5

    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 14

```
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    'Dim TextNo As Integer



    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim Lmax As Single
    Dim Bbeam As Single
    Dim RNum1 As Integer
    Dim RNum2 As Integer
    Dim RNum3 As Integer

    Dim TextNo As Integer
    ReDim Text(100) As String
    TextNo = 1

    Printer.FontName = "Arial"
    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
As #20 Len = 4

    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 14

```
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    'Dim TextNo As Integer



    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim Lmax As Single
    Dim Bbeam As Single
    Dim RNum1 As Integer
    Dim RNum2 As Integer
    Dim RNum3 As Integer

    Dim TextNo As Integer
    ReDim Text(100) As String
    TextNo = 1

    Printer.FontName = "Arial"
    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
As #20 Len = 4

    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If
If XMin > Column(CProj, I).Co_X Then
 XMin = Column(CProj, I).Co_X
End If
If YMax < Column(CProj, I).Co_Y Then
 YMax = Column(CProj, I).Co_Y
End If
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 15

```
      If YMin > Column(CProj, I).Co_Y Then
       YMin = Column(CProj, I).Co_Y
      End If
     Next I

     ''''  Init Value
     'frmOutGraph(CProj).Cls
     Col = 50      'Column Width in Twips
     AddSect = .5 'Additional Section
     InitX = 1000 'Left Edge in Twips
     InitY = 750 'Top Edge in Twips
     ColumnColor = 34
     ElementColor = 30
     ProfileColor = 27
     MomentColor = 17
     Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

     ''''
     Trans_UniformLoad (Wu)'Trans Load

     ''''
     'TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m =
TPMX Twips for X axis
     'TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
 TPMY Twips for Y axis
     TPP = 2000 / .3  'profile 30 cm use 2000 Twips       '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
     TPM = .01

     'Dim Text As String
     Dim Top As Single
     Dim Bottom As Single
     Dim P As Single
     CR = Chr$(13)
     LF = Chr$(10)
     TB = Chr$(9)



     '''' Draw Profile in X axis
     Init1 = InitX
     Init2 = InitY
     Dim Mu As Single
     Dim Mn As Single
     Dim Ms As Single

     Print_Text 0, CR + LF
     Print_Text 0, "=====================================================" +
CR + LF
     Print Text 0, "TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM  " +
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CR + LF
   Print_Text 0, " KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG     " +
CR + LF
   Print_Text 0, "          CIVIL ENGINEERING DEPARTMENT               " +
CR + LF
   Print_Text 0, "      BY SIAM KUNKATE & SOMPHONG KONGKRAPAN          " +
CR + LF
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 16

```
    Print_Text 0, "TENDON PROFILE LIST & STRESS AT TOP & BOTTOM OF SLAB" +
 CR + LF
    Print_Text 0, "=================================================" +
 CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF


    Print_Text 0, "PROFILE IN ROW " + CR + LF
    Print_Text 0, "==============" + CR + LF
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_FEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Print_Text 0, CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, "PROFILE IN ROW : " + Format$(I, "@@@") + CR + LF
     Print_Text 0, "====================" + CR + LF
     Print_Text 0, " No of Tendon = " + Format$(Find_AsStrand(Lmax, Bbeam)
, "@@@") + TB + "Unit" + CR + LF
     Print_Text 0, " Tension per 1 unit tendon = " + Format$(.775 * Genera
lData(CProj).SFs_, "0.00") + " Kg" + CR + LF
     Print_Text 0, CR + LF
     Print_Text 0, s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE FROM LEFT(
m)" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR
 + LF
     Print_Text 0, "
        TOP" + s10 + "BOTTOM" + CR + LF
     Print_Text 0, CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Print_Text 6, Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2
 + Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E
+00") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bo
ttom / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Print_Text 0, "=================================================
==========================" + CR + LF
     Next J
    Next I
    Print_Text 0, "=================================================
==========================" + CR + LF

    '''' Draw Profile in Y axis
    Print_Text 0, CR + LF
    Print_Text 0, CR + LF
    Print_Text 0, "PROFILE IN COLUMN " + CR + LF
    Print_Text 0, "=================" + CR + LF
    For I = 1 To SCol(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 17

```
      Cal_FEndMoment True, I
      Find_BL True, I, Lmax, Bbeam
      P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
      Print_Text 0, CR + LF
      Print_Text 0, CR + LF
      Print_Text 0, "PROFILE IN COLUMN : " + Format$(I, "@@@") + CR + LF
      Print_Text 0, "======================" + CR + LF
      Print_Text 0, " No of tendon = " + Format$(Find_AsStrand(Lmax, Bbeam)
, "@@@") + TB + "Unit" + CR + LF
      Print_Text 0, " Tension per 1 unit tendon = " + Format$(.775 * Genera
lData(CProj).SFs_, "0.00") + " Kg" + CR + LF
      Print_Text 0, CR + LF
      Print_Text 0, CR + LF
      Print_Text 0, "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FROM TOP(m)
" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR +
 LF
      Print_Text 0, "
         TOP" + s10 + "BOTTOM" + CR + LF
      Print_Text 0, CR + LF
      For J = 1 To ColEle(I)
       For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
        M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
        PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
        Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
        Print_Text 7, Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2
 + Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E
+00") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bo
ttom / 10000, " 0.00E+00;-0.00E+00") + CR + LF
       Next Y
       Print_Text 0, "==========================================================
==================================" + CR + LF
     Next J
    Next I
    Print_Text 0, "==========================================================
============================" + CR + LF
    Close #20
    Printer.EndDoc
    Exit Sub
DlgErrorListProfile: Exit Sub
End Sub

Sub Print_Profile ()
    frmMain.cmdlgSystem.Flags = PD_DISABLEPRINTTOFILE
    frmMain.cmdlgSystem.CancelError = True
    On Error GoTo DlgErrorProfile
    frmMain.cmdlgSystem.Action = 5

    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 18

```
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer


    ''''
    XMax = Column(CProj, 1).Co_X
    XMin = Column(CProj, 1).Co_X
    YMax = Column(CProj, 1).Co_Y
    YMin = Column(CProj, 1).Co_Y
    For I = 1 To SNodeNum(CProj)
     If XMax < Column(CProj, I).Co_X Then
      XMax = Column(CProj, I).Co_X
     End If
     If XMin > Column(CProj, I).Co_X Then
      XMin = Column(CProj, I).Co_X
     End If
     If YMax < Column(CProj, I).Co_Y Then
      YMax = Column(CProj, I).Co_Y
     End If
     If YMin > Column(CProj, I).Co_Y Then
      YMin = Column(CProj, I).Co_Y
     End If
    Next I

    ''''  Init Value
    Col = 50      'Column Width in Twips
    AddSect = .1 'Additional Section
    InitX = 1000 'Left Edge in Twips
    InitY = 750 'Top Edge in Twips
    ColumnColor = 34
    ElementColor = 30
    ProfileColor = 27
    MomentColor = 17
    Printer.PSet (3000, 250)
    Printer.Print "PROFILE OF TENDON "
    Printer.PSet (3000, 350)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 19

```
    Printer.Print "=================="

    Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

    ''''
    Trans_UniformLoad (Wu)'Trans Load

    ''''
    TPMX = (Printer.Width - (InitX * 2)) \ (XMax - XMin)'1m = TPMX Twips f
or X axis
    TPMY = (Printer.Height - (InitY * 2)) \ (YMax - YMin)'1m = TPMY Twips
for Y axis
    TPP = 2000 / .3  'profile 30 cm use 2000 Twips        '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
    TPM = .01
    If TPMX < TPMY Then
     TPMY = TPMX
    ElseIf TPMX >= TPMY Then
     TPMX = TPMY
    End If


    '''' Draw Column
    For I = 1 To SNodeNum(CProj)
     X1 = ((Column(CProj, I).Co_X - MinX) * TPMX) - Col + InitX
     Y1 = ((Column(CProj, I).Co_Y - MinY) * TPMY) - Col + InitY
     X2 = ((Column(CProj, I).Co_X - MinX) * TPMX) + Col + InitX
     Y2 = ((Column(CProj, I).Co_Y - MinY) * TPMY) + Col + InitY
     Printer.Line (X1, Y1)-(X2, Y2), Colors(ColumnColor), BF
    Next I

    '''' Draw Element Conjunction in X axis
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CStart = EleColRow(CProj, 0, I, J).Start
      CEnd = EleColRow(CProj, 0, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I

    '''' Draw Element Conjunction in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      CStart = EleColRow(CProj, 1, I, J).Start
      CEnd = EleColRow(CProj, 1, I, J).End
      X1 = ((Column(CProj, CStart).Co_X - MinX) * TPMX) + InitX
      Y1 = ((Column(CProj, CStart).Co_Y - MinY) * TPMY) + InitY
      X2 = ((Column(CProj, CEnd).Co_X - MinX) * TPMX) + InitX
      Y2 = ((Column(CProj, CEnd).Co_Y - MinY) * TPMY) + InitY
      Printer.Line (X1, Y1)-(X2, Y2), Colors(ElementColor)
     Next J
    Next I
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 20

```
    Open DirApp + "profile.swp" For Random Access Read Write Lock Read Wri
te As #21 Len = 4
    Dim WNum As Integer
    Dim L As Single
    WNum = 0
    AddSect = .1

    '''' Draw Profile in X axis
    Init1 = InitX
    Init2 = InitY
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      InitX = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X - MinX
) * TPMX) + Init1
      InitY = ((Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y - MinY
) * TPMY) + Init2
      Printer.PSet (InitX, InitY), ProfileColor
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       X1 = InitX + (X * TPMX)
       Y1 = InitY
       X2 = InitX + (X * TPMX)
       Y2 = InitY - (PF * TPP)
       Printer.Line -(X2, Y2), Colors(ProfileColor)
       Printer.Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
      Next X
     Next J
    Next I

    '''' Draw Profile in Y axis
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      InitX = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_X - MinX)
 * TPMX + Init1
      InitY = (Column(CProj, EleColRow(CProj, 1, I, J).Start).Co_Y - MinY)
 * TPMY + Init2
      Printer.PSet (InitX, InitY), ProfileColor
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       X1 = InitX
       Y1 = InitY + (Y * TPMY)
       X2 = InitX - (PF * TPP)
       Y2 = InitY + (Y * TPMY)
       Printer.Line -(X2, Y2), Colors(ProfileColor)
       Printer.Line (X1, Y1)-(X2, Y2), Colors(ProfileColor)
      Next Y
     Next J '820438
    Next I
    Close #21
    Printer.EndDoc
    Exit Sub
DlgErrorProfile: Exit Sub
End Sub


Sub Screen_ListCompute ()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 21

```
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    ''''end moment
    Dim MWL As Single
    Dim MWR As Single
    Dim MSL As Single
    Dim MSR As Single
    Dim MUL As Single
    Dim MUR As Single

    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim CR As String * 1
    Dim LF As String * 1
    CR = Chr$(13)
    LF = Chr$(10)
    Const ucm = " cm"
    Const um = " m"
    Const ukg = " kg"
    Const uksc = " ksc"
    Const uksm = " kg/m^2"
    Dim CNo As Integer
    Dim RNum As Integer

    'shear
    'Column
    Dim Vc0 As Single
    Dim Vn0 As Single
    'Drop
    Dim Vc1 As Single
    Dim Vn1 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 22

```
    'Capital
    Dim Vc2 As Single
    Dim Vn2 As Single
    'Diagnal
    Dim Vc3 As Single
    Dim Vn3 As Single

    ''''M design
    Dim MDL As Single
    Dim MDD As Single
    Dim MDR As Single
    'M design at column strip
    Dim MCL As Single
    Dim MCD As Single
    Dim MCR As Single
    'M design at middle strip
    Dim MML1 As Single
    Dim MMD1 As Single
    Dim MMR1 As Single
    Dim MML2 As Single
    Dim MMD2 As Single
    Dim MMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim MLB1  As Single
    Dim MMB1  As Single
    Dim MRB1  As Single
    Dim MLT1  As Single
    Dim MMT1  As Single
    Dim MRT1  As Single
    'Half middle strip 2
    Dim MLB2  As Single
    Dim MMB2  As Single
    Dim MRB2  As Single
    Dim MLT2  As Single
    Dim MMT2  As Single
    Dim MRT2  As Single
    'column strip
    Dim CLB  As Single
    Dim CMB  As Single
    Dim CRB  As Single
    Dim CLT  As Single
    Dim CMT  As Single
    Dim CRT  As Single

    ''''Reinforcement design
    Dim RMDL As Single
    Dim RMDD As Single
    Dim RMDR As Single
    'M design at column strip
    Dim RMCL As Single
    Dim RMCD As Single
    Dim RMCR As Single
    'M design at middle strip
    Dim RMML1 As Single
    Dim RMMD1 As Single
    Dim RMMR1 As Single
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 23

```
    Dim RMML2 As Single
    Dim RMMD2 As Single
    Dim RMMR2 As Single
    ''''Area of reinforce steel
    'Half middle strip 1
    Dim RMLB1  As Single
    Dim RMMB1  As Single
    Dim RMRB1  As Single
    Dim RMLT1  As Single
    Dim RMMT1  As Single
    Dim RMRT1  As Single
    'Half middle strip 2
    Dim RMLB2  As Single
    Dim RMMB2  As Single
    Dim RMRB2  As Single
    Dim RMLT2  As Single
    Dim RMMT2  As Single
    Dim RMRT2  As Single
    'column strip
    Dim RCLB  As Single
    Dim RCMB  As Single
    Dim RCRB  As Single
    Dim RCLT  As Single
    Dim RCMT  As Single
    Dim RCRT  As Single

    'Dim TextNo As Integer
    'Dim Text0 As String
    'ReDim Text(100) As String
    'TextNo = 1

    Dim TextNo As Integer
    ReDim Text(100) As String
    TextNo = 1


    TB = Chr$(9)

    AddSect = .5


    '''' Init
    frmList(CProj).Cls

    Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
 As #20 Len = 4
    '''' Information
    Screen_Text CR + LF
    Screen_Text "=======================================================" + C
R + LF
    Screen_Text "TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM  " + C
R + LF
    Screen_Text " KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG    " + C
R + LF
    Screen_Text "          CIVIL ENGINEERING DEPARTMENT              " + C
R + LF
    Screen_Text "     BY SIAM KUNKATE & SOMPHONG KONGKRAPAN          " + C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 24

```
R + LF
    Screen_Text "          LISTING COMPUTATION CHECKING                " + C
R + LF
    Screen_Text "=================================================" + C
R + LF
    Screen_Text CR + LF
    Screen_Text CR + LF

    Screen_Text "    LIST OF PRE-STRESSED DESIGN          " + CR + LF
    Screen_Text "Date         : " + Format$(Now, "dddd mm/dd/yyyy ") + CR +
 LF
    Screen_Text "Time         : " + Format$(Now, "hh:mm:ss AM/PM") + CR + L
F
    Screen_Text "File Name    : " + SData(CProj).FileName + CR + LF
    Screen_Text "Project      : " + SData(CProj).ProjectData + CR + LF
    Screen_Text "Engineer     : " + SData(CProj).Engineer + CR + LF
    Screen_Text "Address      : " + SData(CProj).AddressData + CR + LF
    Screen_Text "Information  : " + SData(CProj).InformData + CR + LF

    '''' Data for Design
    Screen_Text "Length Unit  : Meter(m)" + CR + LF
    Screen_Text "Load Unit    : Kilogram(Kg)" + CR + LF
    Screen_Text "fc' (Slab)   : " + Format$(GeneralData(CProj).SFc_, ".00")
 + uksc + CR + LF
    Screen_Text "fc' (Column) : " + Format$(GeneralData(CProj).CFc_, ".00")
 + uksc + CR + LF
    Screen_Text "fy (Strand)  : " + Format$(GeneralData(CProj).SFs_, ".00")
 + uksc + CR + LF
    Select Case StrandIndex(CProj):
    Case 0:
    Case 1:
    Case 2:
    Case 3:
    Case 4:
    Case 5:
    Case 6:
    Case 7:
    End Select
    'Screen_Text "Es (Strand)     : " + Format$(Es, ".00") + CR + LF
    Screen_Text "Ec (Slab)       : " + Format$(GeneralData(CProj).SEc, ".00
") + uksc + CR + LF
    Screen_Text "Ec (Column)     : " + Format$(GeneralData(CProj).CEc, ".00
") + uksc + CR + LF
    Screen_Text "Slab Thickness : " + Format$(100 * Thick(CProj), ".00") +
 ucm + CR + LF
    Screen_Text "Slab Covering  : " + Format$((Thick(CProj) - d(CProj)) *
100, ".00") + ucm + CR + LF
    Screen_Text "Drape          : " + Format$((Thick(CProj) - 2 * (Thick(C
Proj) - d(CProj))) * 100, ".00") + ucm + CR + LF


    '''' Column Data
    frmList(CProj).FontSize = 7
    frmList(CProj).FontBold = True
    Screen_Text CR + LF
    Screen_Text "COLUMN DATA" + CR + LF
    Screen_Text "===========" + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 25

```
    Screen_Text CR + LF
    Screen_Text "COLUMN NO.  CO_X(m)      CO_Y(m)      LENGTH(m)    " + CR +
LF
    Screen_Text CR + LF
    CNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      CNo = CNo + 1
      Co_X = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_X
      Co_Y = Column(CProj, EleColRow(CProj, 0, I, J).Start).Co_Y
      Support = Column(CProj, EleColRow(CProj, 0, I, J).Start).SupportType
      Length = Column(CProj, EleColRow(CProj, 0, I, J).Start).Length
      Screen_Text Format$(CNo, "@@@@@@@@") + s5 + Format$(Co_X, "0.00E+0")
+ s5 + Format$(Co_Y, "0.00E+0") + s5 + Format$(Length, "0.00E+0") + CR + L
F
      If J = RowEle(I) Then
       CNo = CNo + 1
       Co_X = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_X
       Co_Y = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Co_Y
       Support = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Supp
ortType
       Length = Column(CProj, EleColRow(CProj, 0, I, RowEle(I)).End).Lengt
h
       Screen_Text Format$(CNo, "@@@@@@@@") + s5 + Format$(Co_X, "0.00E+0")
 + s5 + Format$(Co_Y, "0.00E+0") + s5 + Format$(Length, "0.00E+0") + CR +
LF
      End If
     Next J
    Next I
    Screen_Text "=============================================" + CR + LF



    '''' Element  Data
    Dim LCol As Integer
    Dim RCol As Integer
    Dim LBay As Single
    Dim RBay As Single
    'Row
    Screen_Text CR + LF
    Screen_Text CR + LF
    Screen_Text "DESIGN STRIP IN ROW" + CR + LF
    Screen_Text "====================" + CR + LF
    Screen_Text CR + LF
    Screen_Text " ROW   ELEMENT   START-COL  END-COL  LENGTH(m)   TOP-BAY(
m) BOTTOM-BAY(m)" + CR + LF
    Screen_Text CR + LF
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      LCol = EleColRow(CProj, 0, I, J).Start
      RCol = EleColRow(CProj, 0, I, J).End
      Length = EleColRow(CProj, 0, I, J).Length
      LBay = EleColRow(CProj, 0, I, J).L1
      RBay = EleColRow(CProj, 0, I, J).L2
      Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + Format$(LC
ol, "@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@") + s5 + Format$(Length, "0.
00E+0") + s5 + Format$(LBay, "0.00E+0") + s5 + s2 + Format$(RBay, "0.00E+0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 26

```
") + CR + LF
     Next J
    Next I
    Screen_Text "============================================================
================" + CR + LF
    'Column
    Screen_Text CR + LF
    Screen_Text CR + LF
    Screen_Text "DESIGN STRIP IN COLUMN" + CR + LF
    Screen_Text "======================" + CR + LF
    Screen_Text CR + LF
    Screen_Text "COLUMN ELEMENT   START-COL  END-COL  LENGTH(m)   TOP-BAY(
m) BOTTOM-BAY(m)" + CR + LF
    Screen_Text CR + LF
    For I = 1 To SCol(CProj)
     For J = 1 To ColEle(I)
      LCol = EleColRow(CProj, 1, I, J).Start
      RCol = EleColRow(CProj, 1, I, J).End
      Length = EleColRow(CProj, 1, I, J).Length
      LBay = EleColRow(CProj, 1, I, J).L1
      RBay = EleColRow(CProj, 1, I, J).L2
      Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + Format$(LC
ol, "@@@@@@@@@@") + Format$(RCol, "@@@@@@@@@@") + s5 + Format$(Length, "0.
00E+0") + s5 + Format$(LBay, "0.00E+0") + s5 + s2 + Format$(RBay, "0.00E+0
") + CR + LF
     Next J
    Next I
    Screen_Text "============================================================
================" + CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF
    Dim SSystem As String * 50
    Select Case GeneralData(CProj).SlabIndex
    Case 0:
     SSystem = "Flat Plate"
    Case 1:
     SSystem = "Flat Slab With Drop Panel"
    Case 2:
     SSystem = "Flat Slab With Column Capital"
    Case 3:
     SSystem = "Flat Slab With Column Drop Panel And Column Capital"
    End Select
    Screen_Text "SLAB SYSTEM : " + SSystem + CR + LF
    Screen_Text CR + LF
    Screen_Text "Load factor ::: " + CR + LF
    Screen_Text "Wu = " + Format$(DeadCo, "0.0") + "DL" + Format$(LiveCo,
"0.0") + "LL" + CR + LF
    Screen_Text "Unfactored Dead Load           = " + Format$(DeadLoad(CPr
oj), "0.00") + uksm + CR + LF
    Screen_Text "Unfactored Live Load           = " + Format$(LiveLoad(CPr
oj), "0.00") + uksm + CR + LF
    Screen_Text "Unfactored Supper imposed load = " + Format$(SuperLoad(CP
roj), "0.00") + uksm + CR + LF

    Screen_Text CR + LF
    Screen_Text CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 27

```
    Screen_Text "SHEAR CHECKING AT COLUMN,DROP PANEL,COLUMN CAPITAL AND BE
AM ACTION SHEAR" + CR + LF
    Screen_Text "============================================================
==============" + CR + LF
    Screen_Text CR + LF
    Screen_Text "COLUMN" + s10 + s10 + s5 + "PUNCHING SHEAR" + s10 + s10 +
 s9 + "BEAM-ACTION-SHEAR" + CR + LF
    Screen_Text s2 + "NO." + s9 + "COLUMN" + s3 + s10 + "DROP-PANEL" + s1
+ s10 + "COLUMN-CAPITAL" + CR + LF
    Screen_Text s8 + "Vc(kg)" + s5 + "Vn(kg)" + s5 + "Vc(kg)" + s5 + "Vn(k
g)" + s5 + "Vc(kg)" + s5 + "Vn(kg)" + s5 + "Vc(kg)" + s5 + "Vn(kg)" + CR +
 LF
    Screen_Text CR + LF
    RNum = 2
    RIden = 2
    Dim ColumnNo As Integer
    ColumnNo = 0
    For I = 1 To SRow(CProj)
     For J = 1 To RowEle(I)
      ColumnNo = ColumnNo + 1
      CNo = EleColRow(CProj, 0, I, J).Start
      If Column(CProj, CNo).SupportType <> 2 Then 'If Not Free,Mean That i
s Column support
       RIden = RNum + ((CNo - 1) * 8)
       Get #20, RIden + 1, Vc0
       Get #20, RIden + 2, Vn0
       Get #20, RIden + 3, Vc1
       Get #20, RIden + 4, Vn1
       Get #20, RIden + 5, Vc2
       Get #20, RIden + 6, Vn2
       Get #20, RIden + 7, Vc3
       Get #20, RIden + 8, Vn3
       Screen_Text Format$(ColumnNo, "@@@@") + s3 + Format$(Vc0, "0.00E+00
") + s3 + Format$(Vn0, "0.00E+00") + s3 + Format$(Vc1, "0.00E+00") + s3 +
Format$(Vn1, "0.00E+00") + s3 + Format$(Vc2, "0.00E+00") + s3 + Format$(Vn
2, "0.00E+00") + s3 + Format$(Vc3, "0.00E+00") + s3 + Format$(Vn3, "0.00E+
00") + CR + LF
      End If
      If J = RowEle(I) Then
       ColumnNo = ColumnNo + 1
       CNo = EleColRow(CProj, 0, I, J).End
       If Column(CProj, CNo).SupportType <> 2 Then
        RIden = RNum + ((CNo - 1) * 8)
        Get #20, RIden + 1, Vc0
        Get #20, RIden + 2, Vn0
        Get #20, RIden + 3, Vc1
        Get #20, RIden + 4, Vn1
        Get #20, RIden + 5, Vc2
        Get #20, RIden + 6, Vn2
        Get #20, RIden + 7, Vc3
        Get #20, RIden + 8, Vn3
        Screen_Text Format$(ColumnNo, "@@@@") + s3 + Format$(Vc0, "0.00E+0
0") + s3 + Format$(Vn0, "0.00E+00") + s3 + Format$(Vc1, "0.00E+00") + s3 +
 Format$(Vn1, "0.00E+00") + s3 + Format$(Vc2, "0.00E+00") + s3 + Format$(V
n2, "0.00E+00") + s3 + Format$(Vc3, "0.00E+00") + s3 + Format$(Vn3, "0.00E
+00") + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 28

```
        End If
       End If
      Next J
     Next I
     Screen_Text "============================================================
==================================" + CR + LF
     'Drop panel & Capital
    If GeneralData(CProj).SlabIndex <> 0 Then
     CNo = 0
     ColumnNo = 0
     Screen_Text CR + LF
     Screen_Text "============================================================
=========" + CR + LF
     Screen_Text "                DIMENSION OF DROP PANEL & COLUMN CAPITAL
          " + CR + LF
     Screen_Text "
          " + CR + LF
     Screen_Text "============================================================
=========" + CR + LF
     Screen_Text CR + LF
     Screen_Text CR + LF
     Screen_Text s2 + "COLUMN" + s2 + "DROP WIDTH(m)" + s2 + "DROP LENGTH(m
)" + s2 + "DROP THICK(m)" + s2 + "CAPITAL WIDTH(m)" + s2 + "CAPITAL LENGTH
(m)" + CR + LF
     Screen_Text CR + LF
     For I = 1 To SRow(CProj)
      For J = 1 To RowEle(I)
       CNo = CNo + 1
       ColumnNo = EleColRow(CProj, 0, I, J).Start
       DropX = Column(CProj, ColumnNo).DropX
       DropY = Column(CProj, ColumnNo).DropY
       DropThick = Column(CProj, ColumnNo).DropThick
       CapitalX = Column(CProj, ColumnNo).CapitalX
       CapitalY = Column(CProj, ColumnNo).CapitalY
       Screen_Text Format$(CNo, "@@@@@@") + s6 + Format$(DropX, "0.00E+00")
 + s7 + Format$(DropY, "0.00E+00") + s8 + Format$(DropThick, "0.00E+00") +
 s9 + Format$(CapitalX, "0.00E+00") + s10 + Format$(CapitalY, "0.00E+00")
+ CR + LF
       If J = RowEle(I) Then
        CNo = CNo + 1
        ColumnNo = EleColRow(CProj, 0, I, RowEle(I)).End
        DropX = Column(CProj, ColumnNo).DropX
        DropY = Column(CProj, ColumnNo).DropY
        DropThick = Column(CProj, ColumnNo).DropThick
        CapitalX = Column(CProj, ColumnNo).CapitalX
        CapitalY = Column(CProj, ColumnNo).CapitalY
        Screen_Text Format$(CNo, "@@@@@@") + s6 + Format$(DropX, "0.00E+00"
) + s7 + Format$(DropY, "0.00E+00") + s8 + Format$(DropThick, "0.00E+00")
+ s9 + Format$(CapitalX, "0.00E+00") + s10 + Format$(CapitalY, "0.00E+00")
 + CR + LF
       End If
      Next J
     Next I
     Screen_Text "============================================================
==================================" + CR + LF
    ElseIf GeneralData(CProj).SlabIndex = 0 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 29

```
    End If
     'declare
     Dim Lmax As Single
     Dim Bbeam As Single
     Dim P As Single
     Dim Top As Single
     Dim Bottom As Single
     TB = Chr$(9)


     Text1 = ""
     Screen_Text CR + LF
     Screen_Text "=====================================================" + C
R + LF
     Screen_Text "CHECK CRACKING MOMENT & ULTIMATE MOMENT & DEFLECTION" + C
R + LF
     Screen_Text "=====================================================" + C
R + LF
     Screen_Text CR + LF
     Screen_Text CR + LF
     ''''check Cracking Moment & Deflection
     Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj) + DeadCo * Su
perLoad(CProj)
     'text flexeral stress in row
     Lmax = 0
     Bbeam = 0
     Mu = 0
     For I = 1 To SRow(CProj)
      Trans_UniformLoad Wu
      Trans_BElement False, I
      Cal_UltimateEndMoment False, I
      Mu = Find_MuMax(False, I)
      Find_BL False, I, Lmax, Bbeam
      P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
      Screen_Text CR + LF
      Screen_Text " ROW : " + Format$(I, "@@@") + CR + LF
      Screen_Text " =========" + CR + LF
      Screen_Text " No of tendon                    = " + Format$(Find_AsStrand(L
max, Bbeam), "@@@") + TB + "Unit" + CR + LF
      Screen_Text " Tension per 1 unit tendon = " + Format$(.775 * GeneralD
ata(CProj).SFs_, "0.00") + " Kg" + CR + LF
      Screen_Text " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), "0.00")
+ " kg-m" + CR + LF
      Screen_Text " Mn  = " + Format$(Check_MnMax(Find_AsStrand(Lmax, Bbeam
), Bbeam), "0.00") + " kg-m" + CR + LF
      Screen_Text " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
      Screen_Text "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
      Screen_Text " Deflection(Actual)     = " + Format$(Find_MaxDeflection(
False, I, 1), "0.00E+00") + um + CR + LF
      Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Fa
lse, I), "0.00E+00") + um + CR + LF
      Screen_Text "CHECK DEFLECTION AT USED" + CR + LF
      Screen_Text " Deflection(Actual)     = " + Format$(Find_MaxDeflection(
False, I, 2), "0.00E+00") + um + CR + LF
      Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Fa
lse, I), "0.00E+00") + um + CR + LF
      Screen_Text "=========================================" + CR + LF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 30

```
    Next I

    'Text1flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_UltimateEndMoment True, I
     Mu = Find_MuMax(True, I)
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text " COLUMN : " + Format$(I, "@@@") + CR + LF
     Screen_Text " ============" + CR + LF
     Screen_Text " No of tendon                    = " + Format$(Find_AsStrand(L
max, Bbeam), "@@@") + TB + "Unit" + CR + LF
     Screen_Text " Tension per 1 unit tendon = " + Format$(.775 * GeneralD
ata(CProj).SFs_, ".00") + " Kg" + CR + LF
     Screen_Text " Mcr = " + Format$(Check_CrackMoment(P, Bbeam), ".00") +
 " kg-m" + CR + LF
     Screen_Text " Mn  = " + Format$(Check_MnMax(Find_AsStrand(Lmax, Bbeam
), Bbeam), ".00") + " Kg-m" + CR + LF
     Screen_Text " Mu  = " + Format$(Mu, ".00") + " Kg-m" + CR + LF
     Screen_Text "CHECK DEFLECTION AT USED" + CR + LF
     Screen_Text " Deflection(Actual)    = " + Format$(Find_MaxDeflection(
True, I, 1), "0.00E+00") + um + CR + LF
     Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Tr
ue, I), "0.00E+00") + um + CR + LF
     Screen_Text "CHECK DEFLECTION AT AFTER TENSIONED" + CR + LF
     Screen_Text " Deflection(Actual)    = " + Format$(Find_MaxDeflection(
True, I, 2), "0.00E+00") + um + CR + LF
     Screen_Text " Deflection(Allowable) = " + Format$(Check_Deflection(Tr
ue, I), "0.00E+00") + um + CR + LF
     Screen_Text "=============================================" + CR + LF
    Next I

    '''''at TRANSFER load
    Text2 = ""
    Screen_Text CR + LF
    Screen_Text "=====================================================" +
CR + LF
    Screen_Text "CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD" +
CR + LF
    Screen_Text "=====================================================" +
CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF

    Wu = DeadLoad(CProj)
    'text flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 31

```
      Trans_BElement False, I
      Cal_ElasticEndMoment False, I
      Find_BL False, I, Lmax, Bbeam
      P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
      Screen_Text CR + LF
      Screen_Text s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE FROM LEFT(m)
" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR +
 LF
      Screen_Text "
       TOP" + s10 + "BOTTOM" + CR + LF
      Screen_Text CR + LF
      For J = 1 To RowEle(I)
       For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
        M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
        PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
        Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
        Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2 +
 Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+0
0") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bott
om / 10000, " 0.00E+00;-0.00E+00") + CR + LF
       Next X
       Screen_Text "==========================================================
============================" + CR + LF
     Next J
    Next I
    Screen_Text "==========================================================
========================" + CR + LF
    'Text2 flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_ElasticEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FROM TOP(m) "
+ s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR + L
F
     Screen_Text "
      TOP" + s10 + "BOTTOM" + CR + LF
     Screen_Text CR + LF
     For J = 1 To ColEle(I)
      For X = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 1, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2 +
 Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+0
0") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bott
om / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Screen_Text "==========================================================
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 32

```
==========================" + CR + LF
     Next J
    Next I
    Screen_Text "============================================================
==========================" + CR + LF
    AddSect = .5
    ''''check at SURVICE load
    Text3 = ""
    Screen_Text CR + LF
    Screen_Text "============================================================" +
CR + LF
    Screen_Text "CHECK STRESS AT TOP & BOTTOM OF SLAB AT TRANSFER LOAD" +
CR + LF
    Screen_Text "============================================================" +
CR + LF
    Screen_Text CR + LF
    Screen_Text CR + LF

    Wu = DeadLoad(CProj) + LiveLoad(CProj) + SuperLoad(CProj)
    'Text3 flexeral stress in row
    Lmax = 0
    Bbeam = 0
    Mu = 0
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_ServiceEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE FROM LEFT(m)
" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR +
 LF
     Screen_Text "
      TOP" + s10 + "BOTTOM" + CR + LF
     Screen_Text CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
      Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@@") + s10 + s2 +
Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+00
") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Botto
m / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Screen_Text "============================================================
==========================" + CR + LF
     Next J
    Next I
    Screen_Text "============================================================
==========================" + CR + LF
    'Text3 flexeral stress in Column
    Lmax = 0
    Bbeam = 0
    Mu = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 33

```
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_ServiceEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FROM TOP(m) "
+ s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR + L
F
     Screen_Text "
      TOP" + s10 + "BOTTOM" + CR + LF
     Screen_Text CR + LF
     For J = 1 To ColEle(I)
      For X = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 1, I, J).L2 / 2, Top, Bottom
      Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2 +
Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+00
") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Botto
m / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Screen_Text "=============================================================
============================" + CR + LF
     Next J
    Next I
    Screen_Text "=============================================================
============================" + CR + LF
    For I = 1 To 20
     frmList(CProj).txtListCompute(I).Top = 0
     frmList(CProj).txtListCompute(I).Left = 0
     frmList(CProj).txtListCompute(I).Width = frmList(CProj).ScaleWidth
     frmList(CProj).txtListCompute(I).Height = frmList(CProj).ScaleHeight
     frmList(CProj).txtListCompute(I).Visible = False
    Next I
    frmList(CProj).txtListCompute(TextNum + 1).Text = TextToScreen
    TextToScreen = ""
    TextModule = TextNum + 1
    TextNum = 0
    CurrentPage = 1
    frmList(CProj).txtListCompute(1).Visible = True
    frmMain.cmdPrevious.Visible = True
    frmMain.cmdNext.Visible = True
    If TextModule = 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = False
    ElseIf TextModule > 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = True
    End If

    Close #20
End Sub

Sub Screen_ListProfile ()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 34

```
    TextToScreen = ""
    Dim XMax As Integer
    Dim XMin As Integer
    Dim YMax As Integer
    Dim YMin As Integer
    Dim InitX As Integer
    Dim InitY As Integer
    Dim Col As Integer
    Dim CStart As Integer
    Dim CEnd As Integer
    Dim TPMX As Single
    Dim TPMY As Single
    Dim TPP As Single
    Dim TPM As Single
    Dim AddSect As Single
    Dim X1 As Single
    Dim Y1 As Single
    Dim X2 As Single
    Dim Y2 As Single
    Dim X As Single
    Dim Y As Single
    Dim I As Integer
    Dim J As Integer
    'Dim TextNo As Integer



    Dim Init1 As Integer
    Dim Init2 As Integer
    Dim PF As Single
    Dim M As Single
    Dim Wu As Single
    Dim ColumnColor As Integer
    Dim ElementColor As Integer
    Dim ProfileColor As Integer
    Dim Lmax As Single
    Dim Bbeam As Single
    Dim RNum1 As Integer
    Dim RNum2 As Integer
    Dim RNum3 As Integer

   Dim TextNo As Integer
   ReDim Text(100) As String
   TextNo = 1

   'Printer.FontName = "Arial"
   Open DirApp + "tempp.swp" For Random Access Read Write Lock Read Write
As #20 Len = 4

   ''''
   XMax = Column(CProj, 1).Co_X
   XMin = Column(CProj, 1).Co_X
   YMax = Column(CProj, 1).Co_Y
   YMin = Column(CProj, 1).Co_Y
   For I = 1 To SNodeNum(CProj)
    If XMax < Column(CProj, I).Co_X Then
     XMax = Column(CProj, I).Co_X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 35

```
      End If
      If XMin > Column(CProj, I).Co_X Then
       XMin = Column(CProj, I).Co_X
      End If
      If YMax < Column(CProj, I).Co_Y Then
       YMax = Column(CProj, I).Co_Y
      End If
      If YMin > Column(CProj, I).Co_Y Then
       YMin = Column(CProj, I).Co_Y
      End If
     Next I

     ''''  Init Value
     'frmOutGraph(CProj).Cls
     Col = 50      'Column Width in Twips
     AddSect = .5 'Additional Section
     InitX = 1000 'Left Edge in Twips
     InitY = 750 'Top Edge in Twips
     ColumnColor = 34
     ElementColor = 30
     ProfileColor = 27
     MomentColor = 17
     Wu = DeadCo * DeadLoad(CProj) + LiveCo * LiveLoad(CProj)

     ''''
     Trans_UniformLoad (Wu)'Trans Load

     ''''
     'TPMX = (frmOutGraph(CProj).Width - (InitX * 2)) \ (XMax - XMin)'1m =
TPMX Twips for X axis
     'TPMY = (frmOutGraph(CProj).Height - (InitY * 2)) \ (YMax - YMin)'1m =
 TPMY Twips for Y axis
     TPP = 2000 / .3  'profile 30 cm use 2000 Twips       '1m =TPP Twips Fo
r Profile Value ,Maximum = Maximum Thickness/2
     TPM = .01

     'Dim Text As String
     Dim Top As Single
     Dim Bottom As Single
     Dim P As Single
     CR = Chr$(13)
     LF = Chr$(10)
     TB = Chr$(9)



     '''' Draw Profile in X axis
     Init1 = InitX
     Init2 = InitY
     Dim Mu As Single
     Dim Mn As Single
     Dim Ms As Single

     Screen_Text CR + LF
     Screen_Text "==================================================" + C
R + LF
     Screen_Text "TWO WAY FLAT PLATE SLAB DESIGN IN PRESTRESS SYSTEM  " + C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 36

```
R + LF
    Screen_Text " KING MONGKUT INSTITUTE OF TECHNOLOGY LADKRABANG     " + C
R + LF
    Screen_Text "          CIVIL ENGINEERING DEPARTMENT                " + C
R + LF
    Screen_Text "     BY SIAM KUNKATE & SOMPHONG KONGKRAPAN            " + C
R + LF
    Screen_Text "TENDON PROFILE LIST & STRESS AT TOP & BOTTOM OF SLAB" + C
R + LF
    Screen_Text "=================================================" + C
R + LF
    Screen_Text CR + LF
    Screen_Text CR + LF


    Screen_Text "PROFILE IN ROW " + CR + LF
    Screen_Text "==============" + CR + LF
    For I = 1 To SRow(CProj)
     Trans_UniformLoad Wu
     Trans_BElement False, I
     Cal_FEndMoment False, I
     Find_BL False, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text CR + LF
     Screen_Text "PROFILE IN ROW : " + Format$(I, "@@@") + CR + LF
     Screen_Text "====================" + CR + LF
     Screen_Text " No of Tendon = " + Format$(Find_AsStrand(Lmax, Bbeam),
"@@@") + TB + "Unit" + CR + LF
     Screen_Text " Tension per 1 unit tendon = " + Format$(.775 * GeneralD
ata(CProj).SFs_, "0.00") + " Kg" + CR + LF
     Screen_Text CR + LF
     Screen_Text s2 + "ROW" + s2 + "ELEMENT" + s2 + "DISTANCE FROM LEFT(m)
" + s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR +
 LF
     Screen_Text "
      TOP" + s10 + "BOTTOM" + CR + LF
     Screen_Text CR + LF
     For J = 1 To RowEle(I)
      For X = 0 To EleColRow(CProj, 0, I, J).Length Step AddSect
       M = Cal_BMoment(J, X, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(False, I, J, X, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 0, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2 +
 Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+0
0") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bott
om / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next X
      Screen_Text "=================================================
============================" + CR + LF
     Next J
    Next I
    Screen_Text "=================================================
==========================" + CR + LF

    '''' Draw Profile in Y axis
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 37

```
    Screen_Text CR + LF
    Screen_Text CR + LF
    Screen_Text "PROFILE IN COLUMN " + CR + LF
    Screen_Text "==================" + CR + LF
    For I = 1 To SCol(CProj)
     Trans_UniformLoad Wu
     Trans_BElement True, I
     Cal_FEndMoment True, I
     Find_BL True, I, Lmax, Bbeam
     P = .775 * GeneralData(CProj).SFs_ * Find_AsStrand(Lmax, Bbeam)
     Screen_Text CR + LF
     Screen_Text CR + LF
     Screen_Text "PROFILE IN COLUMN : " + Format$(I, "@@@") + CR + LF
     Screen_Text "=======================" + CR + LF
     Screen_Text " No of tendon = " + Format$(Find_AsStrand(Lmax, Bbeam),
"@@@") + TB + "Unit" + CR + LF
     Screen_Text " Tension per 1 unit tendon = " + Format$(.775 * GeneralD
ata(CProj).SFs_, "0.00") + " Kg" + CR + LF
     Screen_Text CR + LF
     Screen_Text CR + LF
     Screen_Text "COLUMN" + s1 + "ELEMENT" + s2 + "DISTANCE FROM TOP(m) "
+ s2 + "POSITION OF TENDON(cm)" + s2 + "CHECKING STRESS(kg/cm^2)" + CR + L
F
     Screen_Text "
      TOP" + s10 + "BOTTOM" + CR + LF
     Screen_Text CR + LF
     For J = 1 To ColEle(I)
      For Y = 0 To EleColRow(CProj, 1, I, J).Length Step AddSect
       M = Cal_BMoment(J, Y, ML(CProj, J), MR(CProj, J))
       PF = Cal_Profile(True, I, J, Y, 1, 1, emax(CProj))
       Find_Stress P, M, PF, EleColRow(CProj, 1, I, J).L1 / 2 + EleColRow(
CProj, 0, I, J).L2 / 2, Top, Bottom
       Screen_Text Format$(I, "@@@@") + Format$(J, "@@@@@@@") + s10 + s2 +
 Format$(X, "0.00E+00") + s10 + s2 + Format$(PF * 100, " 0.00E+00;-0.00E+0
0") + s9 + Format$(Top / 10000, " 0.00E+00;-0.00E+00") + s3 + Format$(Bott
om / 10000, " 0.00E+00;-0.00E+00") + CR + LF
      Next Y
      Screen_Text "=============================================================
============================" + CR + LF
     Next J
    Next I
    Screen_Text "=============================================================
========================" + CR + LF
    Close #20

    For I = 1 To 20
     frmList(CProj).txtListCompute(I).Top = 0
     frmList(CProj).txtListCompute(I).Left = 0
     frmList(CProj).txtListCompute(I).Width = frmList(CProj).ScaleWidth
     frmList(CProj).txtListCompute(I).Height = frmList(CProj).ScaleHeight
     frmList(CProj).txtListCompute(I).Visible = False
    Next I
    frmList(CProj).txtListCompute(TextNum + 1).Text = TextToScreen
    TextToScreen = ""
    TextModule = TextNum + 1
    TextNum = 0
    CurrentPage = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTPUTP.BAS - 38

```
    frmList(CProj).txtListCompute(1).Visible = True
    frmMain.cmdPrevious.Visible = True
    frmMain.cmdNext.Visible = True
    If TextModule = 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = False
    ElseIf TextModule > 1 Then
     frmMain.cmdPrevious.Enabled = False
     frmMain.cmdNext.Enabled = True
    End If

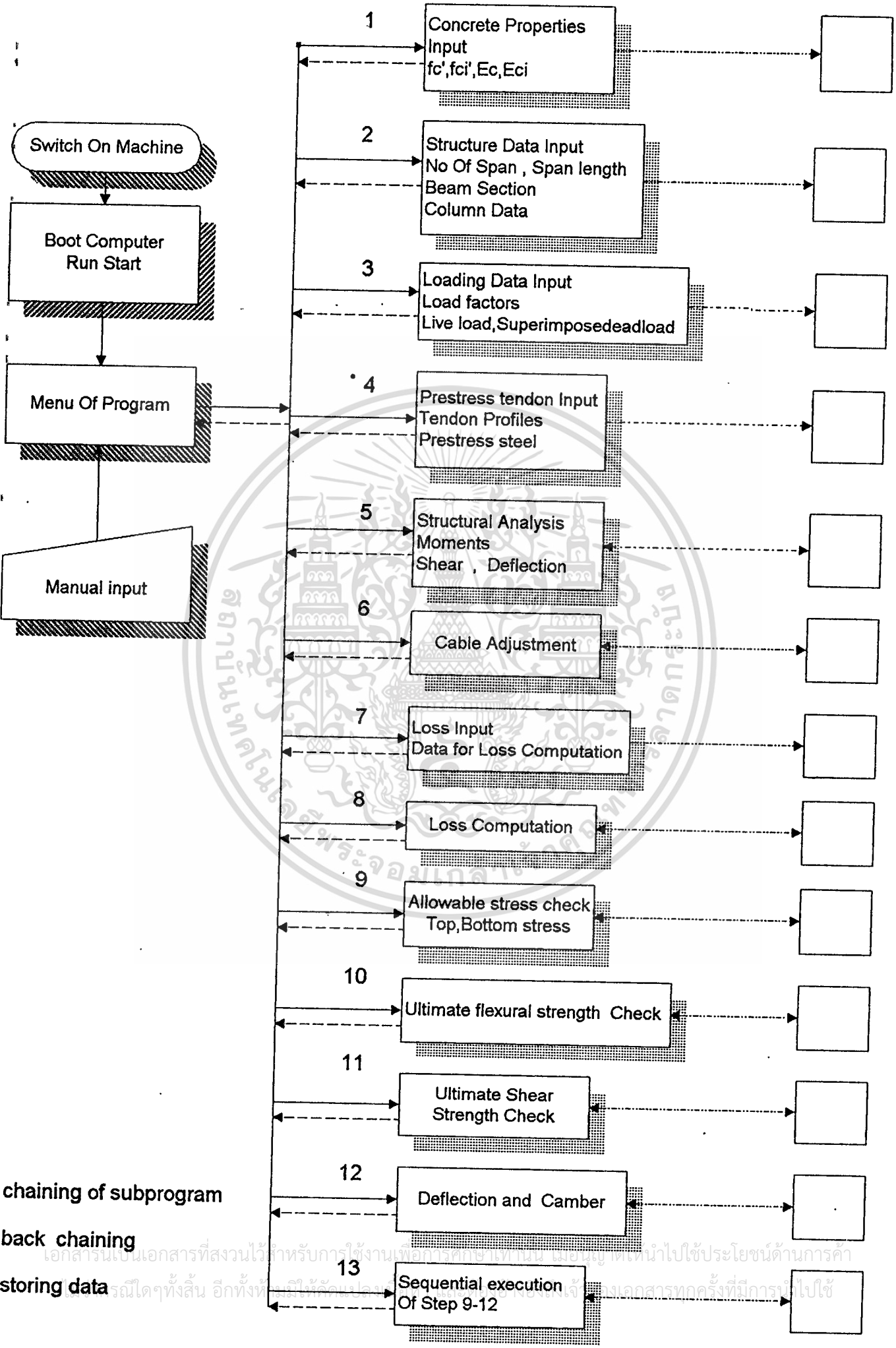
End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART

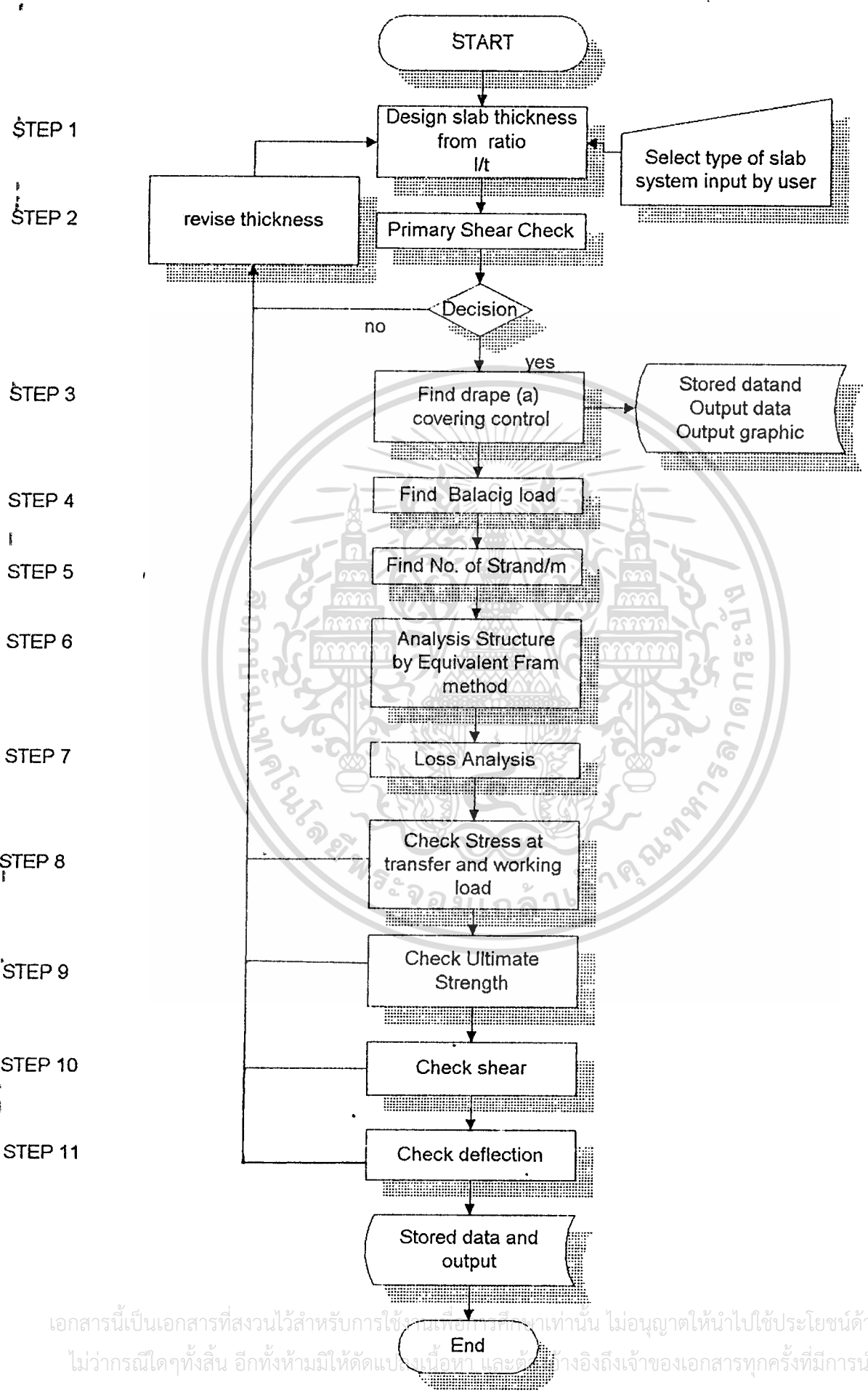
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

User Initiated Execute Steps (Upon selection of Program)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

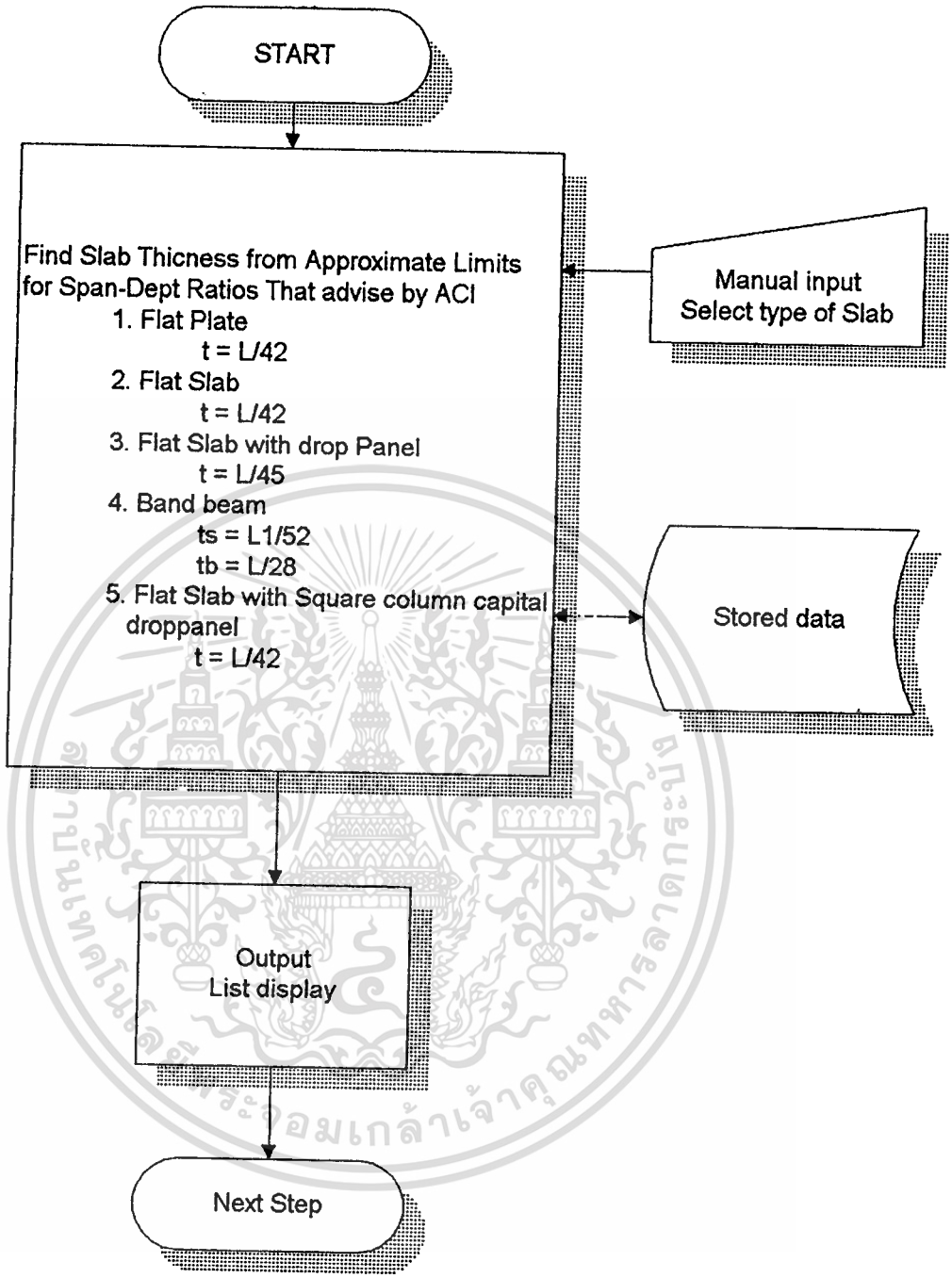
DESIGN STEP MAIN FLOW CHART



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR THICKNESS DESIGN

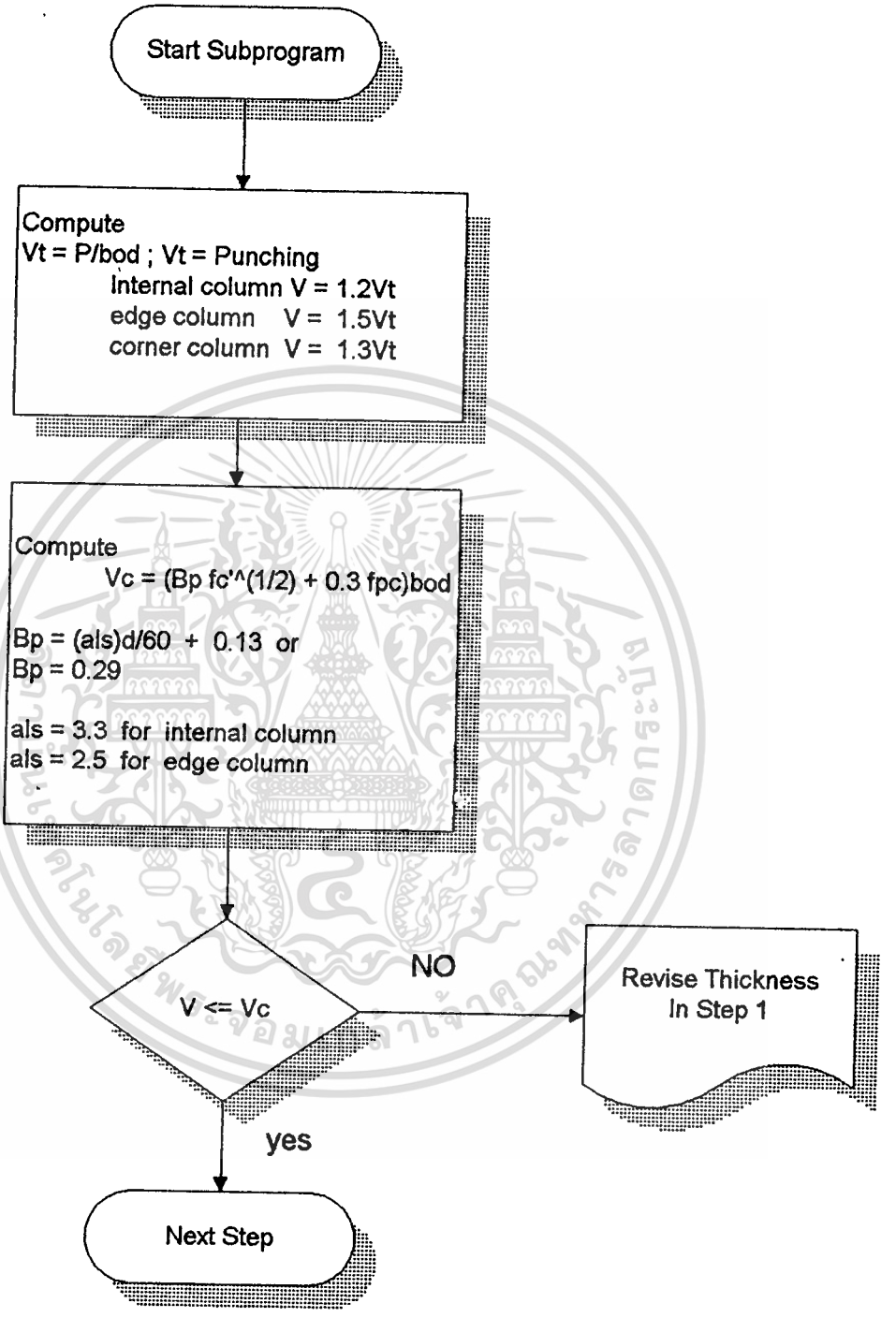
STEP 1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

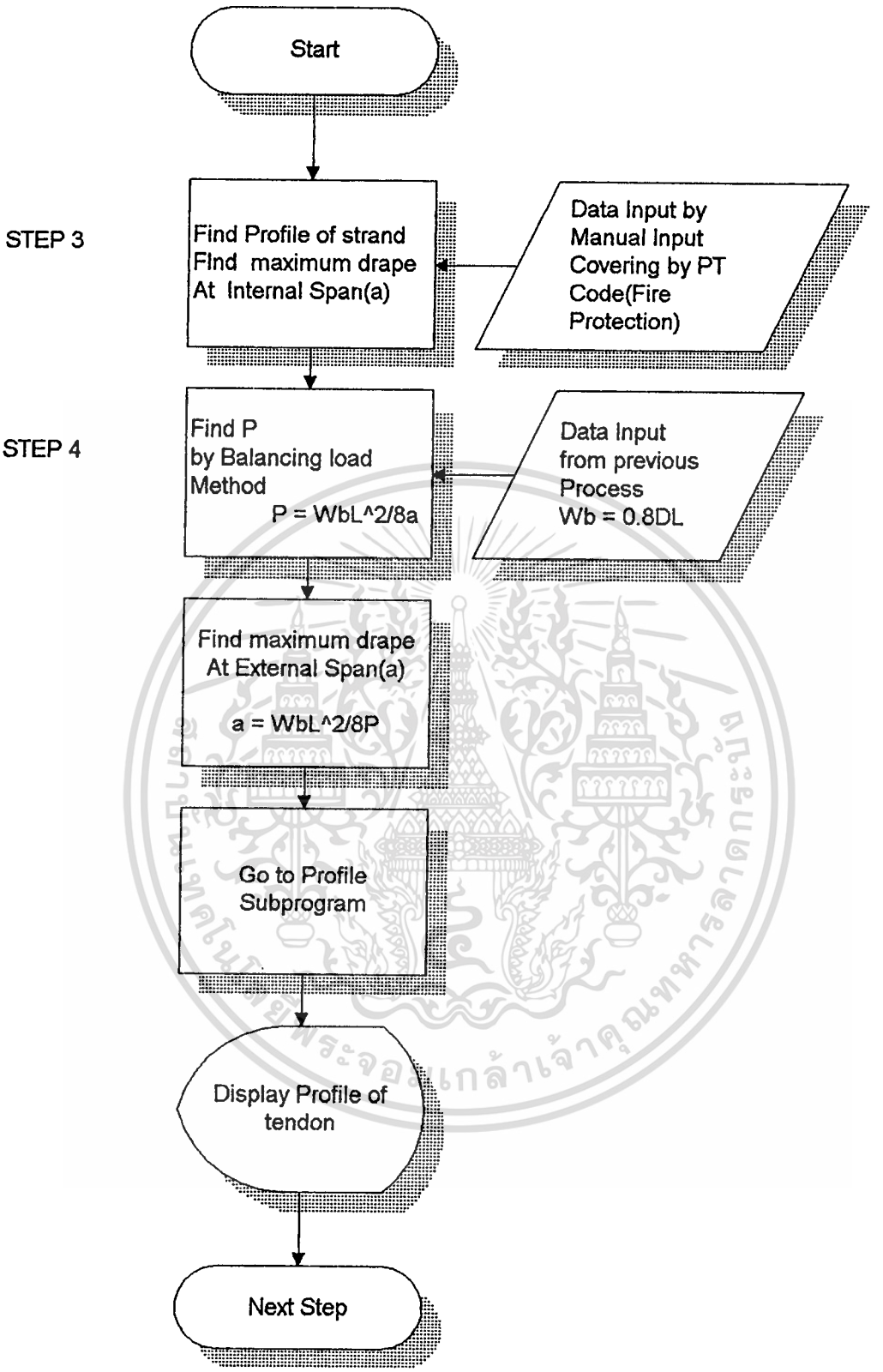
FLOWCHART FOR PRIMARY SHEAR CHECK

STEP 2



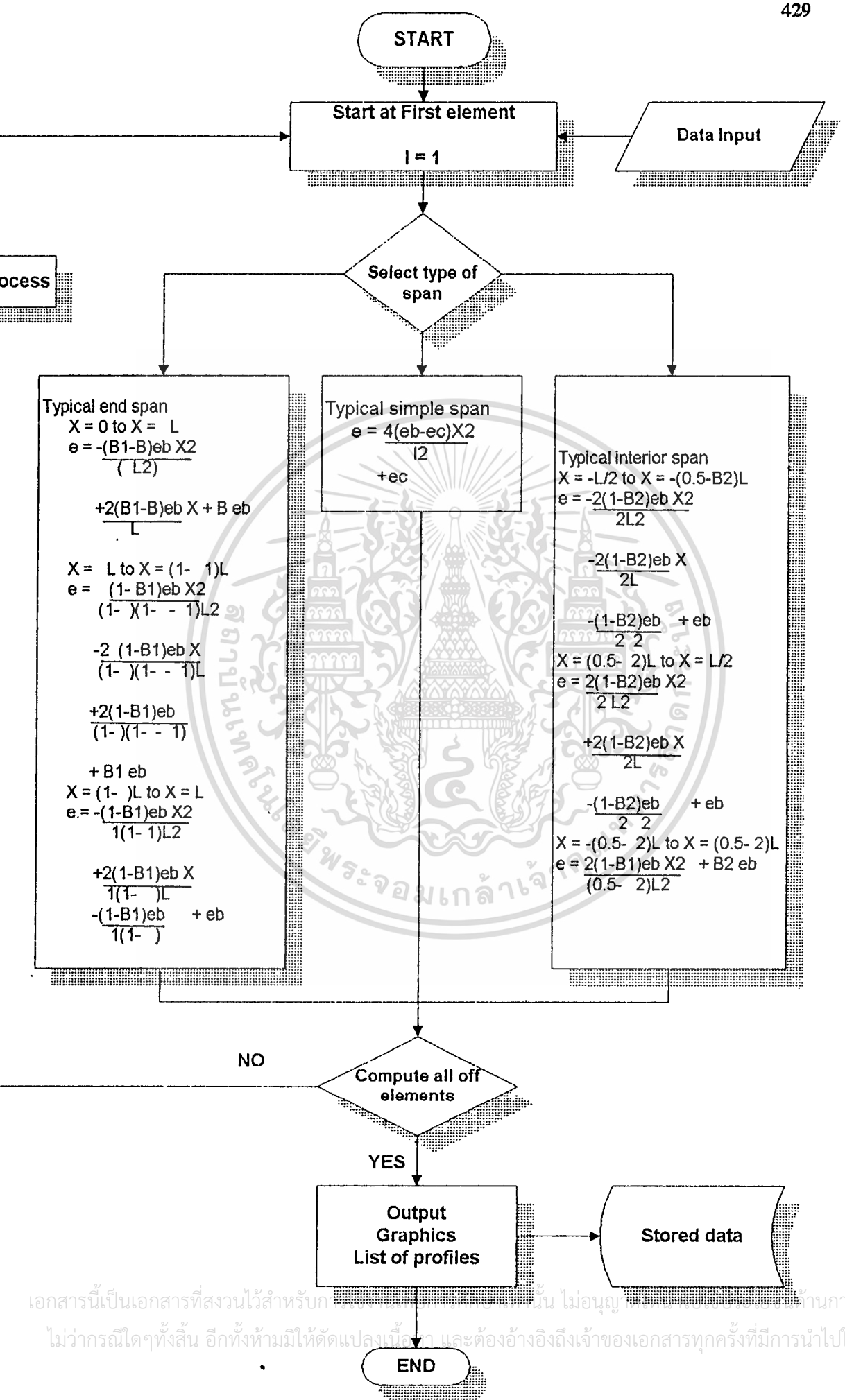
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR PROFILE OF STRAND



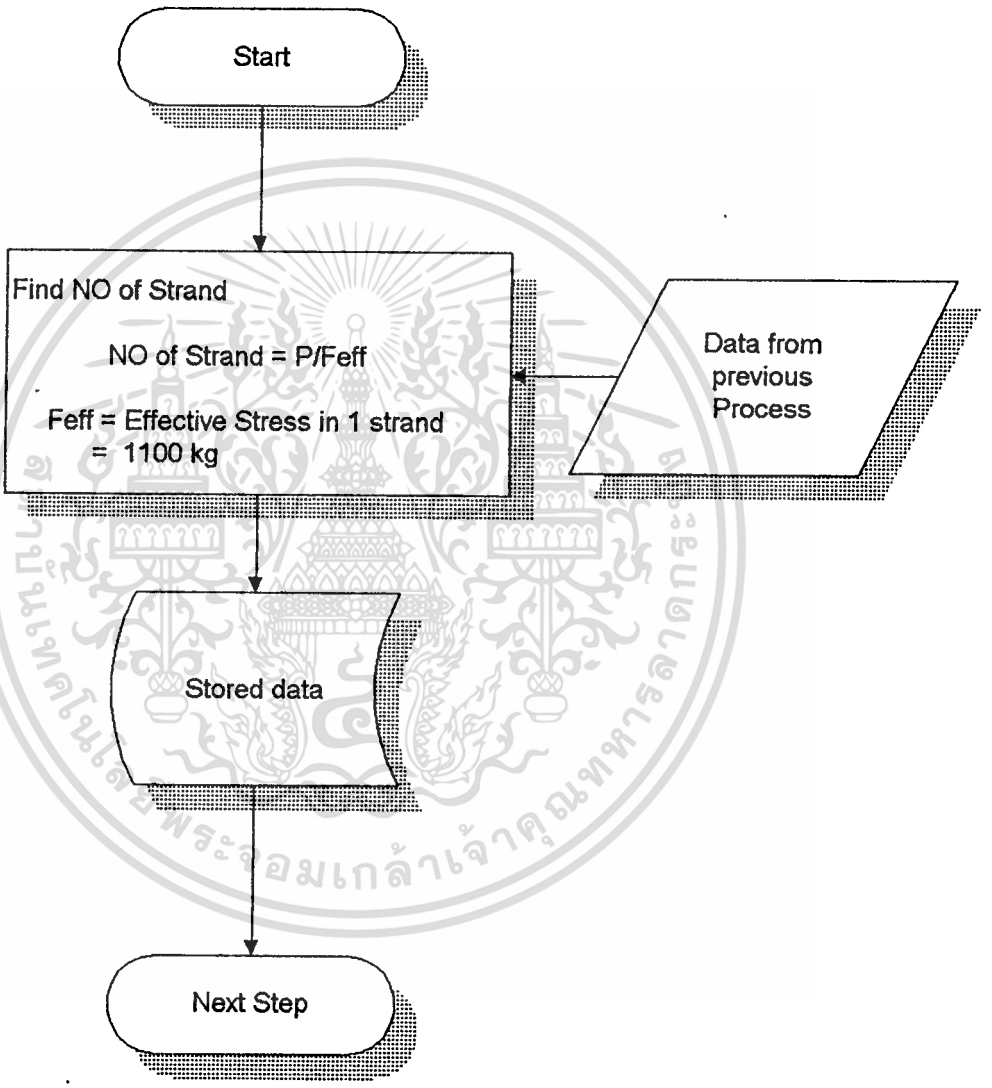
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOW CHART OF PROFILE OF TENDON



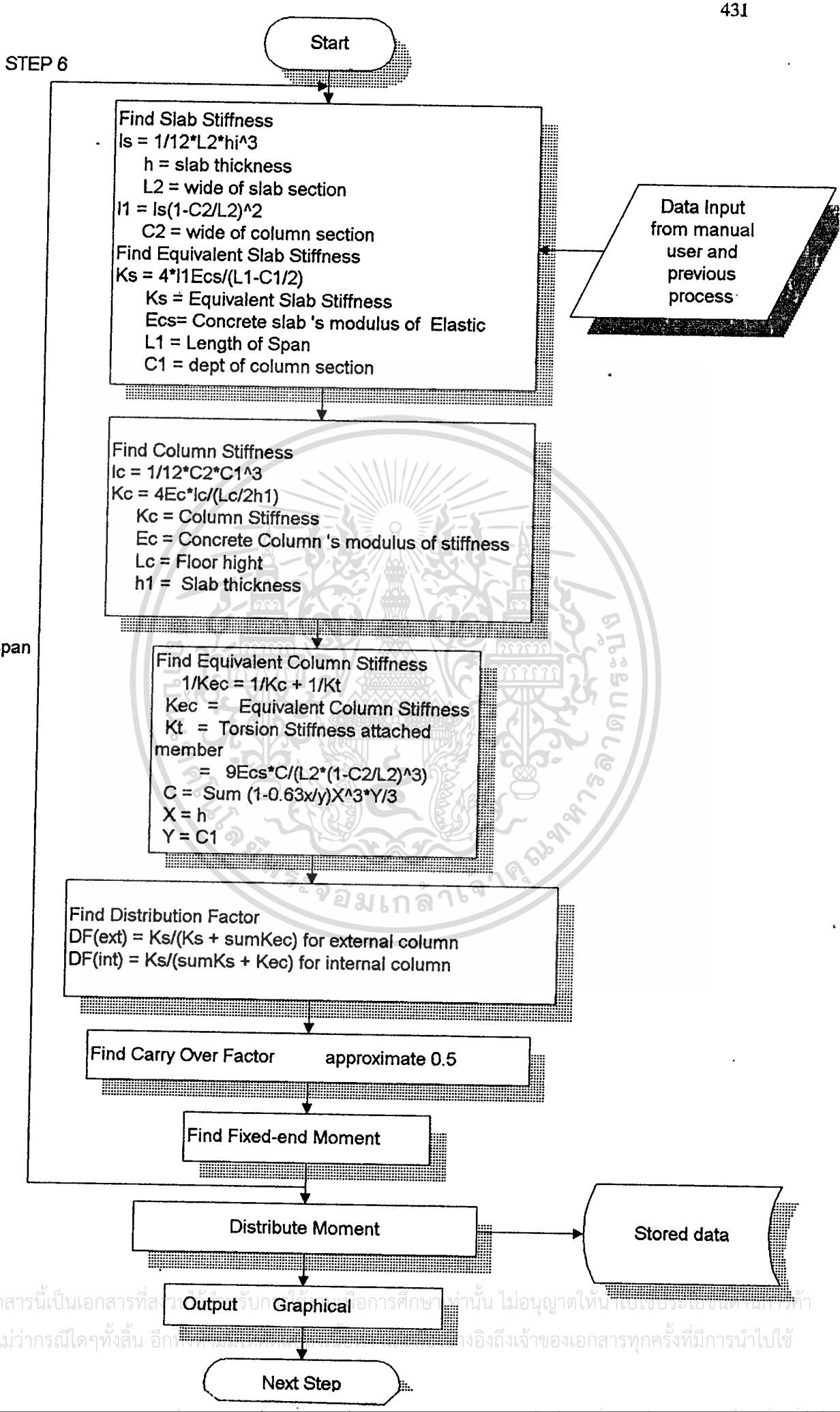
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STEP 5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

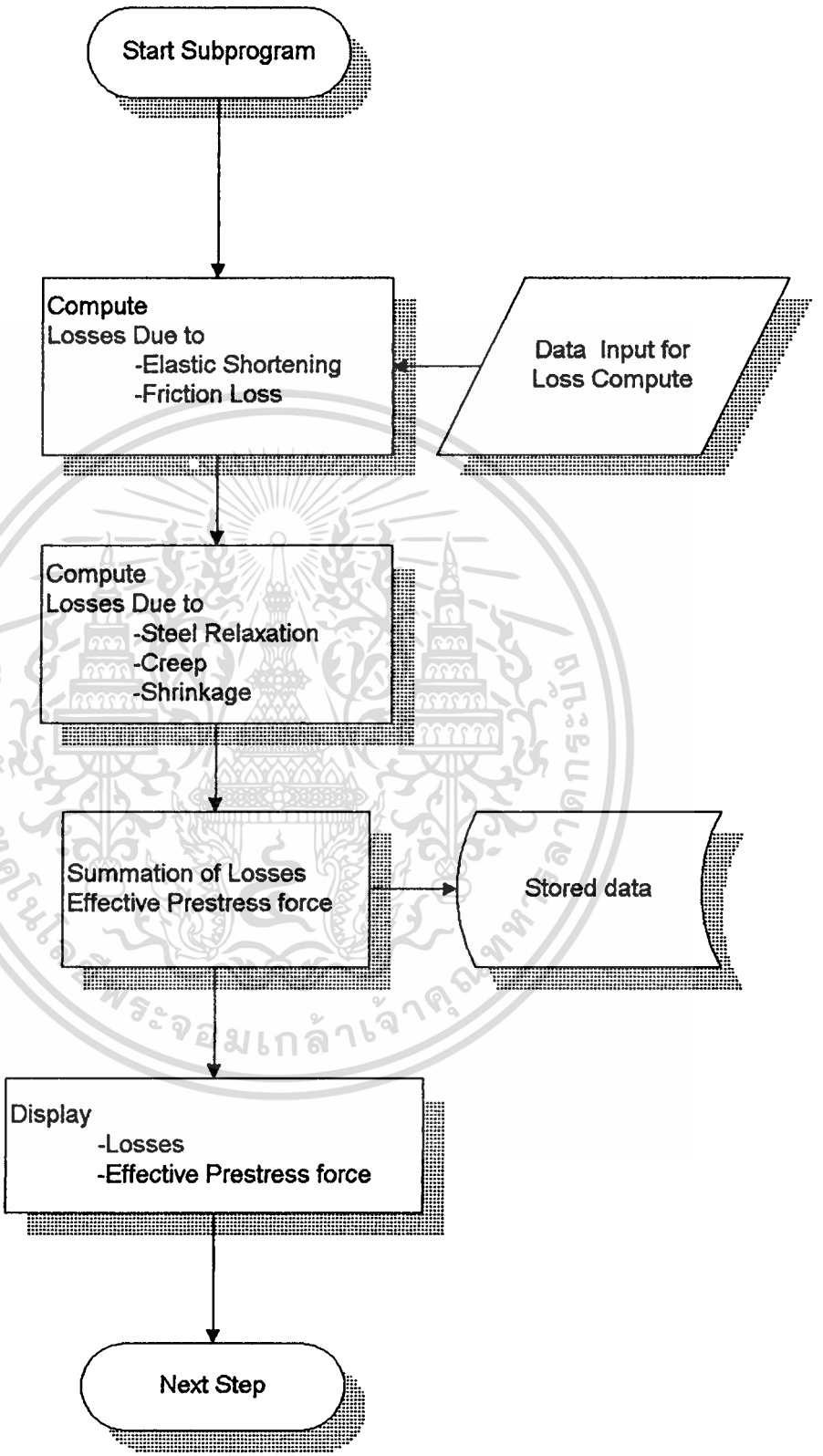
FLOWCHART FOR ANALYSIS MOMENT BY EQUIVALENT FRAME METHOD



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

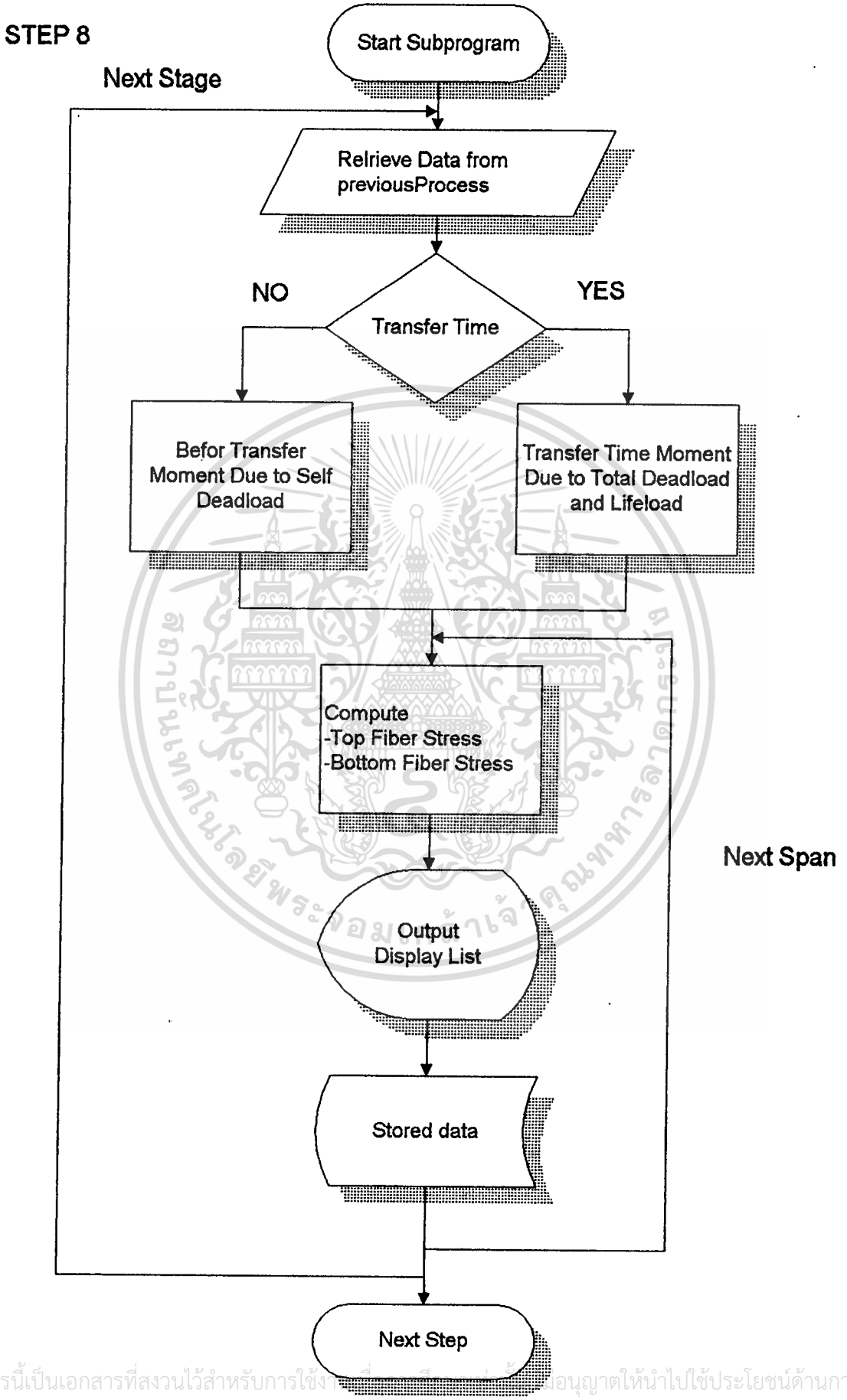
FLOECHART FOR LOSS COMPUTE

STEP 7



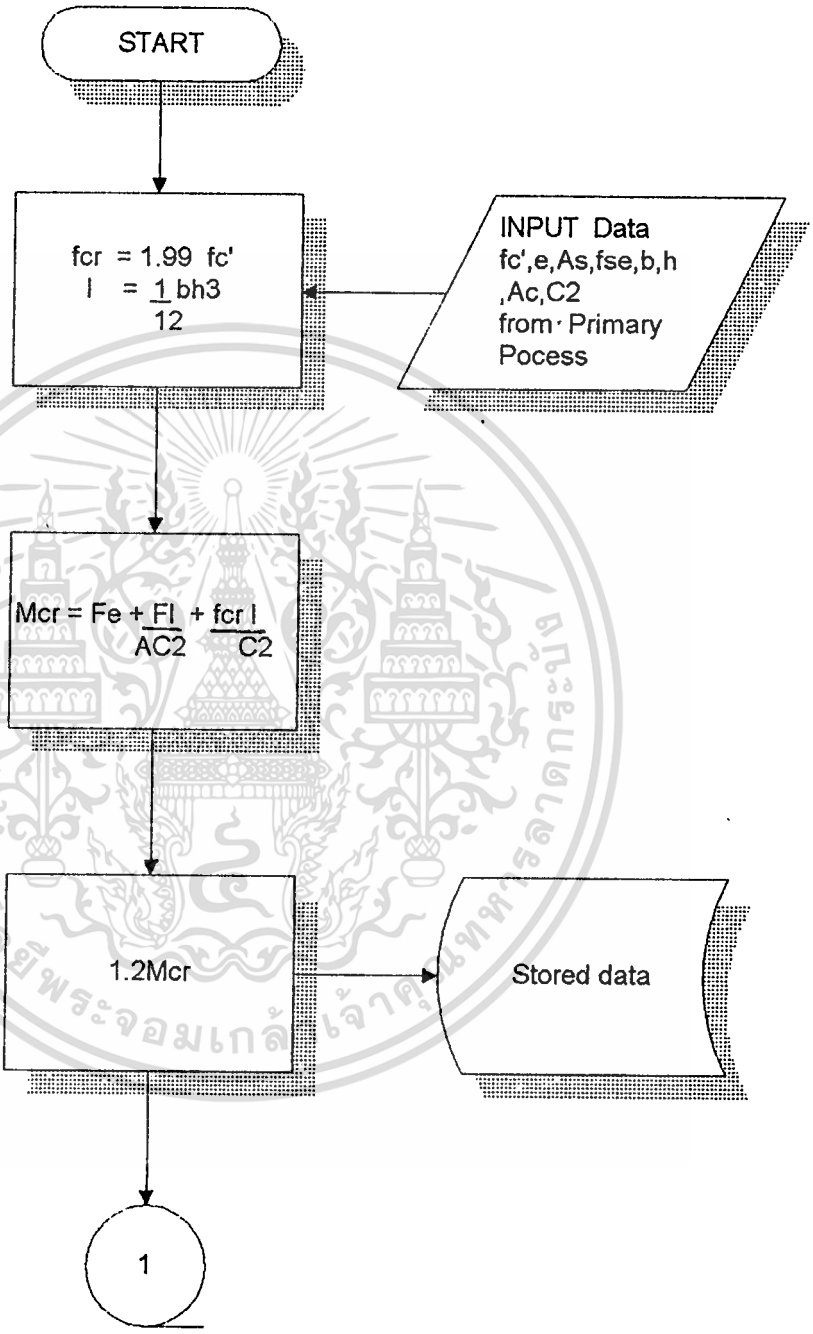
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR STRESS CHECK



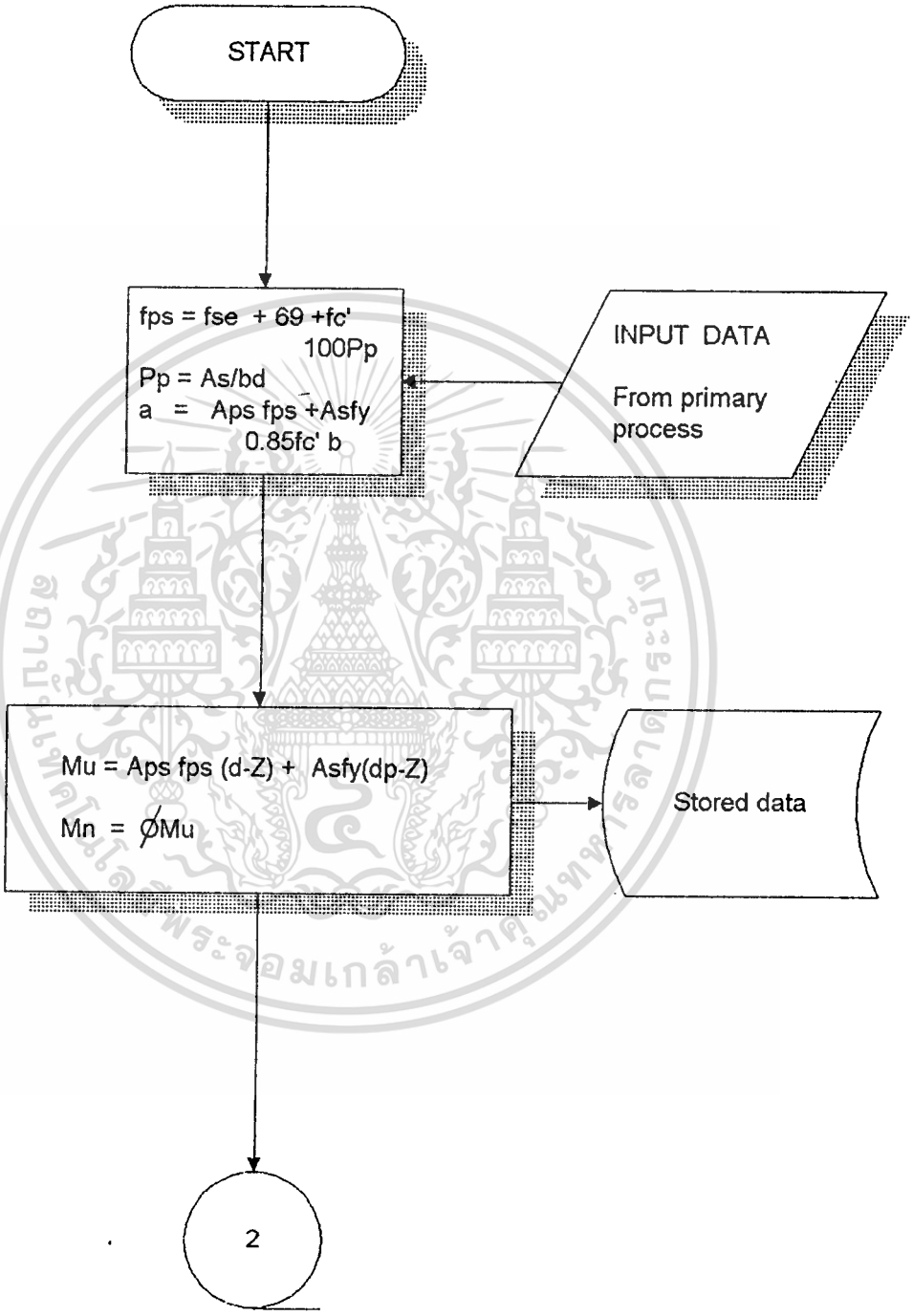
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน[illegible]อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR "CRACKING MOMENT CHECK"



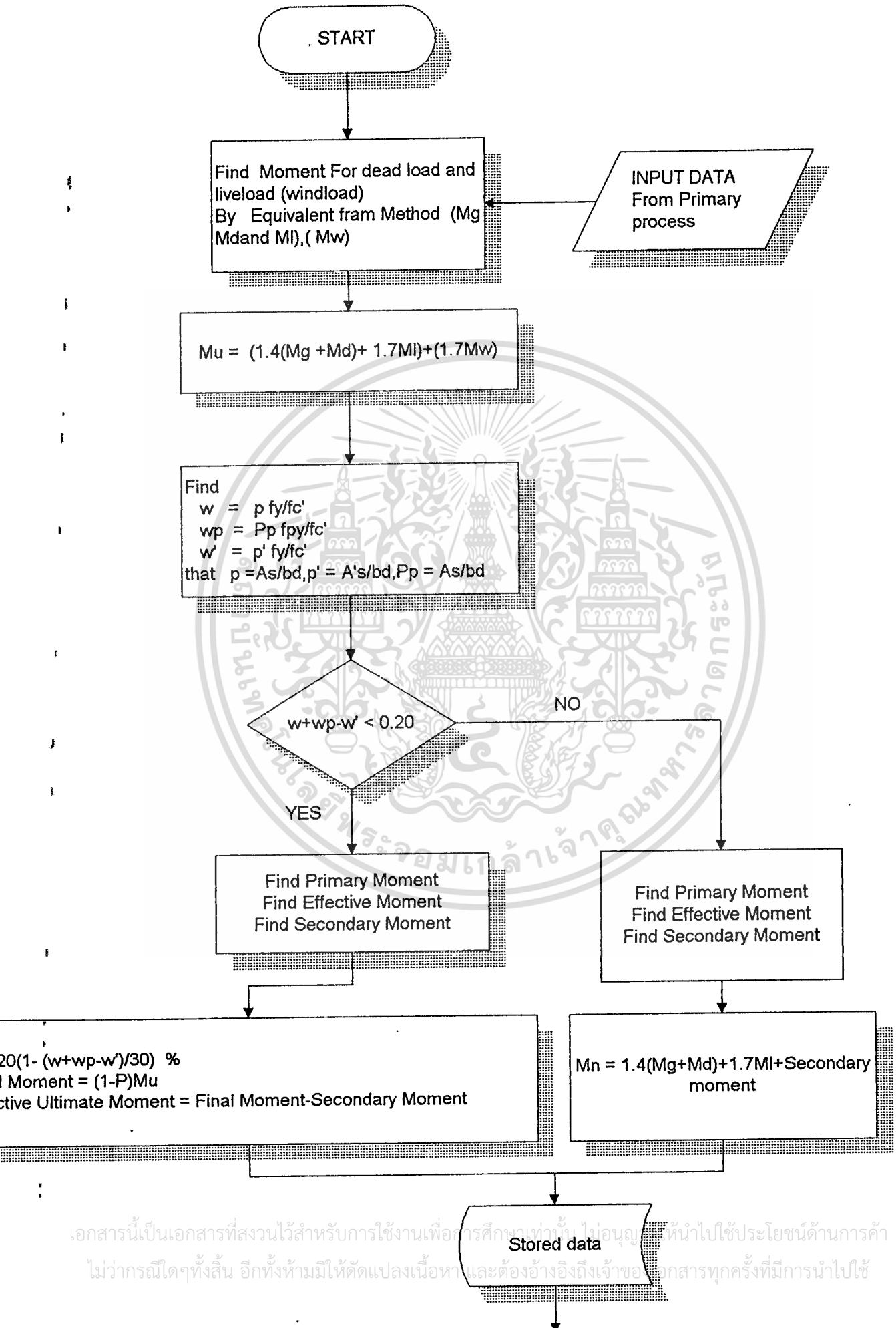
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR "ULTIMATE DESIGN"



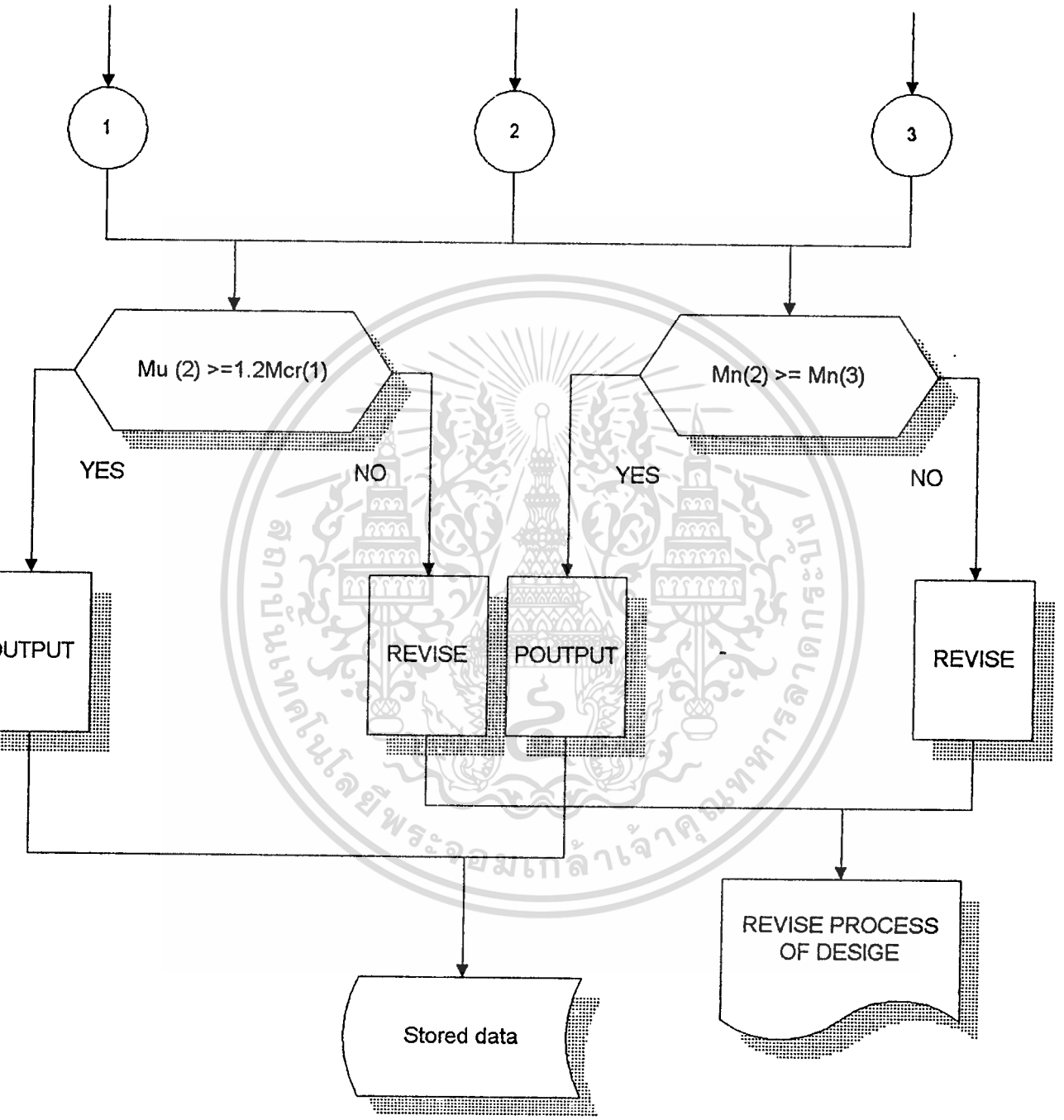
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR "ELASTIC MOMENT CHECK"



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

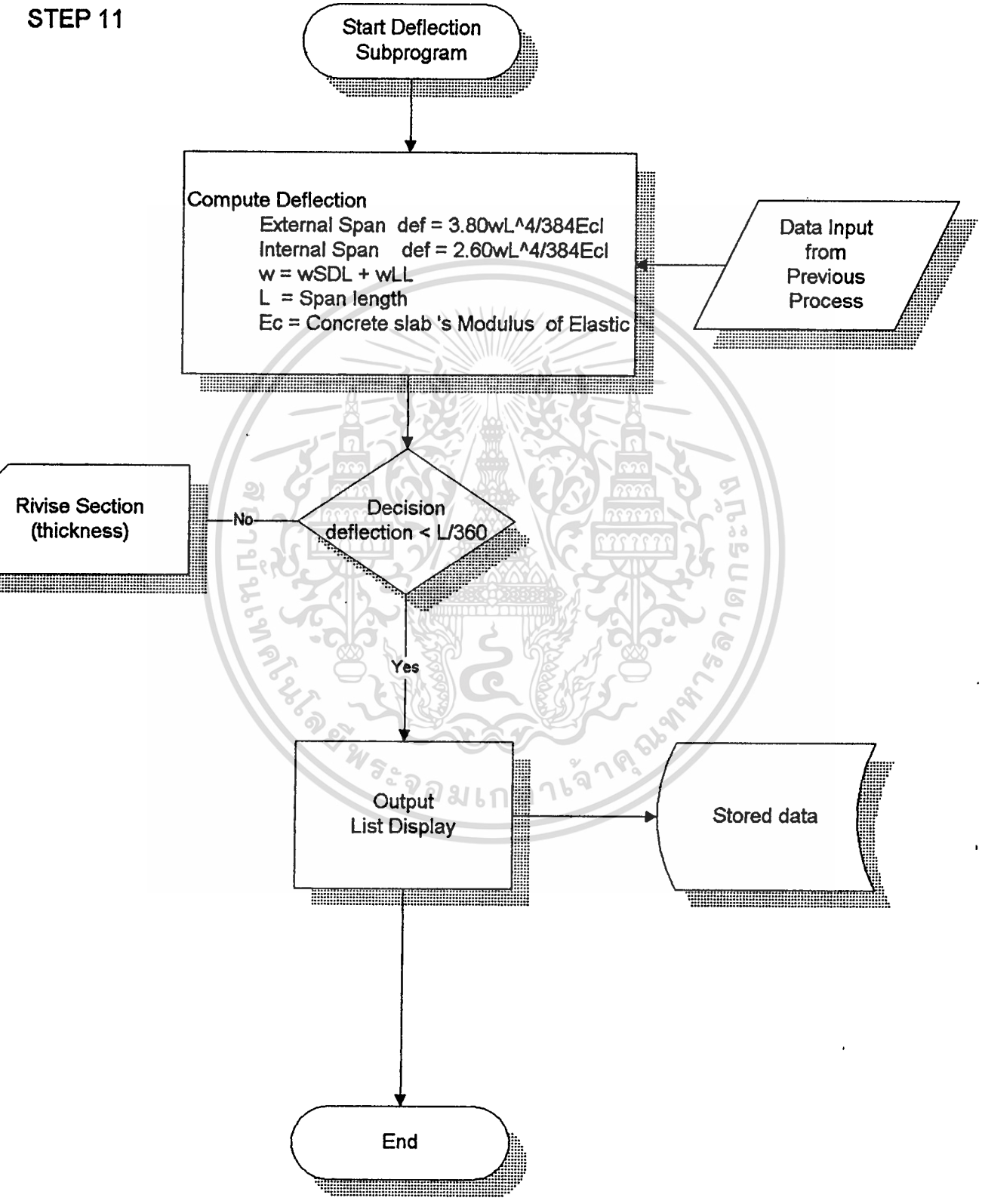
FLOWCHART FOR "PREPARATION MOMENT"



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART FOR DEFLECTION CHECK

STEP 11



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. "Requirements for Reinforced Concrete with Design Applications", Note on ACI 318-77 Building Code, Portland Cement Association, Second Edition, part 21
2. Design of Prestressed Concrete Structures Third Edition
3. Paul F. Rice and Edward S.Hoffman, "Structural Desing Guide To The ACI Building Code", Third Edition, chapter 5,6
4. Arthur H. Nilson, George Winter, "Design Of Concrete Structures", McGraw-Hill, Eleventh Edition, page 472-485
6. การออกแบบคอนกรีตอัดแรง โดย ดร. ต่อกุล กาญจนาลัย
7. อำนวย พานิชกุล, สมนึก กุลประภา, วินิต ช่อวิเชียร "การวิเคราะห์โครงสร้าง" พิมพ์ครั้งที่ 6, พ.ศ. 2535
9. "มาตรฐานสำหรับอาคารคอนกรีตอัดแรง", คณะกรรมการสาขาวิศวกรรมโยธา 2533-34, ตุลาคม 2534, พิมพ์ครั้งที่ 2, พฤษภาคม 2536
10. สนั่น เจริญผ่า และ วินิต ช่อวิเชียร, "คอนกรีตเสริมเหล็ก", ฉบับแก้ไขเพิ่มเติม พิมพ์ครั้งที่ 8, มิถุนายน 2537, บทที่ 10, 11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้