การใช้คอมพิวเตอร์ช่วยในการออกแบบภาพ 3 มิติ โดยใช้โปรแกรม AutoCAD Release 12

3D GRAPHIC DESIGN USE AUTOCAD RELEASE 12

โดย

นางสาว มัลลิกา ชาติไทย

นางสาว สุวรรณา กาลธิยานันท์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033286

ปริญญานิพนธ์ทางวิศวกรรมคอมพิวเตอร์

ภาคเรียนที่ 2 ปีการศึกษา 2536

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การใช้คอมพิวเตอร์ช่วยในการออกแบบภาพ 3 มิติ

โดยใช้ **AutoCAD Release 12**

ผู้จัดทำ 1.นางสาว มัลลิกา ชาติไทย

2.นางสาว สุวรรณา กาลธิยานันท์

.....อาจารย์ที่ปรึกษา

(ดร. วรวัฒน์ ลิ้มโภคา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน : การใช้คอมพิวเตอร์ช่วยในการออกแบบภาพ 3 มิติ โดยใช้โปรแกรม AutoCAD Release 12

: 3D Graphic Design Use AutoCAD Release 12

ผู้จัดทำ : นางสาว มัลลิกา ชาติไทย

: นางสาว สุวรรณา กาลธิยานันท์

อาจารย์ที่ปรึกษา : ดร. วรวัฒน์ ลิ้มโภคา

บทคัดย่อ

โครงงานนี้มีวัตถุประสงค์เพื่อนำโปรแกรมออโตแคด รีลีส 12 มาประยุกต์ใช้ในการสร้างภาพ 3 มิติจากภาพ 2 มิติ ซึ่งประกอบด้วย การสร้างภาพไอโซเมติกของวัตถุ 3 มิติจากภาพฉาย 3 ด้าน โดยให้ผู้ใช้ทำการเขียนภาพทั้ง3ด้านแล้วโปรแกรมจะทำการคำนวณเพื่อสร้างภาพวัตถุ 3 มิติ, การสร้างภาพทรงกรวยและภาพคลี่ของทรงกรวยนั้น, และสุดท้ายเป็นการสร้างภาพบ้าน 3มิติ จากแบบแปลนบ้าน 2 มิติ โดยผู้ใช้สามารถทำการออกแบบแปลนบ้านในระนาบ 2 มิติตามความต้องการ และสามารถกำหนดมุมมองของบ้านในระนาบ 3 มิติได้ ซึ่งการประยุกต์นี้จะใช้ โปรแกรมออโตลิปส์เป็นเครื่องมือช่วยในการประยุกต์โปรแกรมออโตแคด รีลีส 12 นี้

ในการนำโปรแกรมออโตแคดมาช่วยในการสร้างภาพ 3 มิตินี้จะทำให้เกิดความสะดวกในการใช้งานและได้ภาพที่เหมือนจริงเนื่องจากโปรแกรมออโตแคดมีคำสั่งในการสร้างภาพ 3 มิติ และ การใช้ระบบเมนูที่มีความสะดวกในการใช้งาน จึงสามารถใช้โปรแกรมออโตลิปส์ควบคุมการใช้งานได้ดี

ABSTRACT

An object of this project is using AutoCAD Release 12 to build three dimension Modeling from two dimension view and show isometric drawing . User must input some value for program to produce product. To construct cone and development of it. Finally construct house in three dimension from 2 dimension house plan ,user can design house in two dimension plan and program produce house in three dimension to show.

Using AutoCAD to produce three dimension drawing is facility to produce product that like real object because AutoCAD has many command and utility that AutoLISP can control it.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
1.1 ประวัติความเป็นมาของออโตแคด	1
1.2 ลักษณะของออโตแคด รีลีส 12	2
1.3 เนื้อหาโดยย่อของแต่ละบท	6
บทที่ 2 ทฤษฎีและหลักการเบื้องต้น	7
2.1 คำสั่งที่ใช้ทั่วไป	7
2.2 การป้อนคำสั่งให้กับออโตแคด	35
2.3 เครื่องมือช่วยในการเขียน	35
2.4 การใช้คีย์บอร์ดในการตรออิง	38
2.5 การใช้เมาส์ในการตรออิง	39
2.6 การใช้เทปเบล็ตในการตรออิง	39
2.7 การรับตรออิงมาจากสแกนเนอร์	42
2.8 การแสดงตรออิงออกจากจอภาพ	42
2.9 การแสดงภาพโดยเทคนิคสไลด์	43
2.10 การแสดงภาพโดยเทคนิคสคลิปไฟล์	44
2.11 การสร้างสกรีนเมนู	45
2.12 การสร้างป๊อปอัพเมนู	47
2.13 การสร้างเมนูรูปภาพ	48
2.14 การกำหนดรูปแบบการระบาย	49
2.15 การใช้คำสั่งการควบคุมแสงและการสร้างฉาก	50
2.16 ออโตลิปส์	51
บทที่ 3 การคำนวณและการสร้าง	74
บทที่ 4 การทดลองและผลการทดลอง	106
4.1 การสร้างภาพวัตถุ 3 มิติจากภาพ 2 มิติ	106
4.2 การทดลองโปรแกรมสร้างทรงกรวย 3 มิต และภาพคลี่ทรงกรวย	112
4.3 ผลการทดลองโปรแกรมการสร้างบ้าน 3 มิติ จากแบบแปลน 2 มิติ	112
บทที่ 5 บทวิจารณ์ และ สรุป	141
ภาคผนวก	
สรุปคำสั่งที่ใช้ในออโตแคด	1
ตัวแปรระบบในออโตแคด	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
รูปที่1 แสดง Flowchart of program bulid 3D-Modeling from orthographic 2D	76
รูปที่ 2 แสดงการสร้าง 4-viewport	78
รูปที่ 3 แสดงการ extrude object in layer "top"	80
รูปที่ 4 แสดงการ extrude object in layer "front"	80
รูปที่ 5 แสดงการ extrude object in layer "side"	82
รูปที่ 6 แสดงการ extrude object in each layer และทำการ intersect 3 object ให้เป็น object in layer "9"	82
รูปที่ 7 แสดงการ subtract object in layer "0" with object in layer "9"	83
รูปที่ 8 แสดง Flowchart of program make 3D-cone and bulid Development of 3D-cone	85
รูปที่ 9 แสดงรายละเอียดของ Algorithm ของโปรแกรม การสร้างรูปกรวย 3 มิติและนำมาสร้างภาพคลี่	87
รูปที่ 10 แสดงขอบเขตของบ้านและตัวแปรต่างๆที่เกี่ยวข้อง	89
รูปที่ 11 แสดงจุดต่างๆของขอบเขตในการวาด สร้างพื้น และ พรมปูพื้นของชั้นที่ 1 และทำการสอดแทรก บันไดเข้ามาในแปลน	90
รูปที่ 12 แสดงการวาดเสาให้แปลนบ้าน	91
รูปที่ 13 แสดงจุดต่างๆของขอบเขตในการวาด สร้างพื้น และพรมปูพื้นของชั้นที่ 2 และทำการสอดแทรก บันไดเข้ามาในแปลน	91
รูปที่ 14 แสดงจุดต่างๆที่ใช้ในการสร้างพื้น ของชั้นที่ 3 ถึงชั้นที่ n และทำการสอดแทรก บันไดเข้ามาในแปลน	92
รูปที่ 15 แสดงรูปแบบของหลังคา 5 ชนิด	93
รูปที 16 แสดงตัวอยางภาพที่วาดลงบน TOP VIEW	107
รูปที 17 แสดงตัวอยางภาพที่วาดลงบน FTONT VIEW	107
รูปที 18 แสดงตัวอยางภาพที่วาดลงบน SIDE	107
รูปที 19 แสดงตัวอยางภาพที่ไดจากการเลือกเมนู COMPLETE	107
รูปที่ 20 แสดงตัวอย่างภาพที่วาดลงบน Top View (เมนู TOP2)	109
รูปที่ 21 แสดงตัวอย่างภาพที่วาดลงบน Front View (เมนู FRONT2)	109

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป หน้า

รูปที่ 22 แสดงตัวอย่างภาพที่วาดลงบน Side View (เมนู SIDE2) 109
รูปที่ 23 แสดงภาพที่ได้หลังจากเลือกเมนู COMPLETE2 109
รูปที่ 24 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER 110
รูปที่ 25 แสดงรูปภาพที่ได้จากการทำคำสั่ง 3DCONE ซึ่งได้ลบเส้นที่ถูกบังทิ้งแล้ว 110
รูปที่ 26 แสดงรูปภาพของภาพคลี่ของทรงกรวยที่ได้จากการทำคำสั่ง CONEPROJ 111
รูปที่ 27 แสดงค่าต่าง ๆของภาพคลี่หลังจากใช้คำสั่ง LIST 111
รูปที่ 28 แสดงไดอะลอกบล็อกที่ได้จากการเลือกเมนู House[Begin] 113
รูปที่ 29 แสดงรูปภาพที่ได้หลังจากการเลือกเมนู House[Step1] 113
รูปที่ 30 แสดงรูปภาพที่ได้หลังการเติมประตูเข้าไปในแบบแปลน 116
รูปที่ 31 แสดงรูปภาพที่ได้หลังจากการเติมหน้าต่างเข้าไปในแบบแปลน 116
รูปที่ 32 แสดงรูปภาพที่ได้หลังจากการเติมกำแพงเข้าไปในแบบแปลน 117
รูปที่ 33 แสดงรูปตัวบ้าน 3 มิติหลังจากเติมกำแพงประตูและหน้าต่างแล้ว 117
รูปที่ 34 แสดงรูปภาพที่ได้จากการเติมเฟอร์นิเจอร์ต่าง ๆเข้าไปในแบบแปลน 119
รูปที่ 35 แสดงรูปภาพที่ได้จากการเติมทางเดินเท้าเข้าไปในแบบแปลน 119
รูปที่ 36 แสดงรูปภาพที่ได้จากการเติมระเบียงเข้าไปในแบบแปลน 121
รูปที่ 37 แสดงรูปตัวบ้าน 3 มิติหลังจากเติมเฟอร์นิเจอร์,ทางเดินเท้า,และระเบียง แล้ว 121
รูปที่ 38 แสดงรูปภาพที่ได้จากการเติมต้นไม้เข้าไปในแบบแปลน 122
รูปที่ 39 แสดงรูปตัวบ้าน 3 มิติหลังจากเติมต้นไม้เข้าไปในแบบแปลนแล้ว 123
รูปที่ 40 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 0,0,0 124
รูปที่ 41 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 0,1,0 124
รูปที่ 42 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 1,0,1 125
รูปที่ 43 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ -1,-1,1 125
รูปที่ 44 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 1,1,1 126
รูปที่ 45 แสดงรูปตัวบ้าน 3 มิติหลังจากทำการเปิดกำแพงและหลังคาออก 126
รูปที่ 46 แสดงการกำหนดจุดมุมมองบริเวณชั้นที่ 1 บริเวณห้องครัว 128
รูปที่ 47 แสดงผลลัพธ์ที่จากการกำหนดจุดมุมมองของชั้นที่ 1 128
รูปที่ 48 แสดงการกำหนดจุดมุมมองบริเวณชั้นที่ 2 บริเวณห้องนอน 129
รูปที่ 49 แสดงผลลัพธ์ที่จากการกำหนดจุดมุมมองของชั้นที่ 2 129
รูปที่ 50 แสดงการกำหนดจุดมุมมองบริเวณชั้นที่ 3 บริเวณบันได 130
รูปที่ 51 แสดงผลลัพธ์ที่จากการกำหนดจุดมุมมองของชั้นที่ 3 130
รูปที่ 52 แสดงตำแหน่งของผู้ใช้บริเวณสนามนอกบ้านที่มีมุมมองมายังตัวบ้าน 131
รูปที่ 53 แสดงรูปมุมมองที่ได้จากรูปที่ 52 131
รูปที่ 54 แสดงมุมมองอื่น ๆ หลังจากเลือกเมนู View[Target2] 133

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป หน้า

รูปที่ 55 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2] 134
รูปที่ 56 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2] 135
รูปที่ 57 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2] 136
รูปที่ 58 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2] 137
รูปที่ 59 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER 138
รูปที่ 60 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER 139
รูปที่ 61 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER 140

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 แสดงการใช้คีย์บอร์ด	38
ตารางที่ 2.2 Keyboard with Dialogue Box	38
ตารางที่ 2.3 แสดงอักขระพิเศษที่ใช้ในการควบคุมเมนู	45
ตารางที่ 2.4 แสดงรหัสและความหมายการกำหนดดีไวซ์	46
ตารางที่ 2.5 แสดงรายละเอียดตัวแปรและรูปแบบในการ shade	48
ตารางที่ 2.6 แสดงรหัสและความหมายของการควบคุม	58
ตารางที่ 2.7 แสดงความสัมพันธ์ของฟังก์ชัน และรหัสที่มีผลในการควบคุม	59
ตารางที่ 2.8 แสดงรหัสควบคุมและความหมาย	59
ตารางที่ 2.9 แสดงชนิดของการอินพุตที่ได้จาก grread	60
ตารางที่ 2.10 แสดง unitcode และรูปแบบที่ใช้ใน rtos	63
ตารางที่ 2.11 แสดงรหัสสำหรับฟังก์ชัน open	64
ตารางที่ 2.12 แสดงรหัสกลุ่มต่างๆที่ใช้เป็นรายการของ ตัวกรองในฟังก์ชัน ssget	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

ออโตแคด(AutoCAD) เป็นโปรแกรมประเภทแคด(CAD) ที่ได้รับความนิยมสูงสุดทั่วโลก ทั้งนี้เพราะเป็นโปรแกรมที่มีโครงสร้างของโปรแกรมระบบเปิด คือเป็นโปรแกรมที่เปิดเผยรายละเอียดให้ทราบกันทั่วโลก ดังนั้นผู้ใช้จึงสามารถใช้โปรแกรมในทางเขียนแบบได้ทั่วๆไป ไม่จำเพาะเจาะจง ผู้ใช้สามารถปรับโปรแกรมได้ตามต้องการ เพื่อให้เหมาะสมกับประเภทของงานและสะดวกในการใช้งานยิ่งขึ้น นอกจากใช้ในงานเขียนแบบแล้วยังสามารถนำไปใช้งานด้านอื่นๆได้ เช่น พรีเซนเตรชันได้อีกด้วย

1.1 ประวัติความเป็นมาของออโตแคด

เริ่มมาจากรีลีส(Release) 1 ออกเมื่อเดือนธันวาคม 1982 โดยครั้งแรกออกเวอร์ชัน 1.00 ที่ทำงานบนแพลตฟอร์ม CP/M ต่อมาได้ออกเวอร์ชัน 1.10 ที่ทำงานบนแพลตฟอร์ม DOS และได้มีการพัฒนาโปรแกรมมาเรื่อยจนถึงรีลีส 7 เวอร์ชัน 2.50 ได้มีการเปลี่ยนแปลงมากขึ้น โดยมีคำสั่งเพิ่มขึ้นมากกว่า 70 คำสั่ง จะเน้นในด้านความสะดวกในการเขียนแบบ

ต่อมาได้มีการพัฒนาขึ้นอีกในรีลีส9โดยเน้นทางด้านการใช้งานง่ายขึ้นได้มียูสเซอร์อินเตอร์เฟซเพิ่มขึ้นมา สามารถเรียกใช้คำสั่งได้จากพลูดาวน์(Pull Down),ไดอะลอกบล็อก(Dialogue Box)และไอคอน(Icon)ในเดือนตุลาคม1988 ก็ได้ออกรีลีส10 ที่ได้เพิ่มความสามารถในการเขียนแบบ 3 มิติ โดยได้ใช้หลักการเขียนภาพในระนาบต่างๆ ด้วย User CoOrdinate System (UCS) ส่วนการเขียนในระนาบ 2 มิติ ใช้ World Co-Ordinate system (WCS) และใน รีลีส10 ยังได้ออกเวอร์ชันที่ทำงานบนแพลตฟอร์มทั้งของเครื่องระดับ 386 และ SunSPARC

ปลายปี 2533 ได้ออกรีลีส 11 ขึ้นมาโดยได้มีการพัฒนาทางด้านการแสดงผลทางจอภาพและการพล็อตแบบได้ 2ลักษณะคือ แบบ Model Space และแบบ Paper Space สำหรับการเซ็ตสีสามารถที่จะทำได้ในโปรแกรม ในการเขียนภาพ 3 มิติก็มี AME ขึ้นมาช่วยให้เขียนภาพ 3 มิติในลักษณะ Solid model ทำให้การสร้างภาพ 3 มิติที่ซับซ้อนได้ง่ายขึ้น ซึ่งแตกต่างจาก รีลีส 10 ที่เป็นการเขียนภาพ 3มิติในลักษณะพื้นผิว(surface)และในรีลีสนี้ก็ได้ออกเวอร์ชันบนหลายแพลตฟอร์มทั้งระดับพีซี และเวิร์กสเตชัน และเมื่อกลางปี 2535 ได้มีการพัฒนาโปรแกรม ออโตแคด ขึ้นมาอีกมากมาย จัดได้ว่าเป็นการเปลี่ยนแปลงครั้งสำคัญที่สุด โดยได้ออกรีลีส12 ซึ่งมีทั้งเวอร์ชันบนแพลตฟอร์ม DOS และ SunSPARC ส่วนแพลตฟอร์มอื่นๆ คือ MS Windows,DECstation ,SGI Iris Indigo แอปเปิล แมคอินทอช เอชพี 9000/700 และไอบีเอ็ม RS6000 กำลังอยู่ในระหว่างการพัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ลักษณะของออโตแคด รีลีส12 (AutoCAD Release12 Feature)

1.2.1 ความเข้ากันได้กับออโตแคด รีลีส 11

ออโตแคด รีลีส 12 ได้ถูกออกแบบให้มีความเข้ากันได้ (Compatible) กับรีลีส 11 โดยที่สามารถส่งดรออิงไฟล์ (Drawing file) ไปมาจากรีลีส 11 ไปยังรีลีส 12 ไปยังรีลีส 11 ได้ ซึ่งช่วยให้เกิดความสะดวกกับผู้ใช้ที่ทำงานกับดรออิง (Drawing) จากหน่วยงานอื่นที่ยังใช้รีลีส11 อยู่ได้ หรือไม่สามารถปรับปรุง(upgrade) เป็นรีลีส 12 ได้ทั้งหมด

1.2.2 เพิ่มความเร็วในการแสดงภาพ โดยการออกแบบลดโอเวอร์เฮด (Overhead) ในการงานที่ไม่จำเป็น ทำให้สามารถ

-Pan และ Zoom ด้วยความเร็วสูงด้วยการใช้ 32 บิท เวกเตอร์ สเปท(Vector Space) ทำให้การรีเจน (Regen) น้อยที่สุดด คำสั่ง Pan จะได้ประโยชน์จากการนี้ด้วยเช่นกัน

-ในดรออิงขนาดใหญ่ การเลือกวัตถุจะเร็วขึ้นมากซึ่งเป็นผลมาจากการใช้อัลกอริทึม (Algorithm) แบบใหม่ เวลาที่ใช้ในการเลือกวัตถุ การ snap การแสดงภาพผลที่ได้ จะไม่ต้องเรียงลำดับก่อน หลังตามการสร้างวัตถุนั้นอีกต่อไป

-ทำการซ่อนบังเส้นด้วยความเร็วสูง

โดยใช้อัลกอริทึม แบบใหม่และความจำแบบเสมือน ทำให้การซ่อนบังเส้นเร็วขึ้น 5-100 เท่ากว่ารุ่นก่อนและยังแสดงเปอร์เซนต์ของการซ่อนบังเส้นที่ทำไปแล้วให้ทราบด้วย

1.2.3 เพิ่มประสิทธิภาพการติดต่อผู้ใช้แบบกราฟฟิค ออโตแคด รีลีส 12 ใช้กราฟิกยูเซอร์อินเตอร์เฟซ (Graphic User Interface (GUI)) ที่เป็นมาตราฐานในโปรแกรมที่ทำงานด้วยกราฟิกยูเซอร์อินเตอร์เฟซ อยู่แล้วคือ วินโดว์(Window) ทำให้เรียนรู้และใช้งานได้ง่าย การเพิ่มประสิทธิภาพที่สำคัญมีดังนี้

-ยกเลิกการใช้เมนเมนู (Main menu)

เมนเมนูที่มีแต่ข้อความถูกแทนที่โดยไดอะลอกบล็อก (Dialogue Boxes) การเรียกโปรแกรมจะเข้าสู่ดรออิงอิดิเตอร์ (Drawing Editor) โดยตรงทันทีและเริ่มต้นดรออิงใหม่ให้โดยอัตโนมัติโดยไม่ต้องตั้งชื่อก่อน

-ไดอะลอกบล็อก,ป็อปอัพเมนู (Pop up menu),พลูดาวน์เมนู (Pull down menu) แบบใหม่

-พลูดาวน์เมนู สามารถแสดงเมนูย่อยพร้อมกันกับเมนูหลักได้

-ป็อปอัพเมนู ที่สามารถแสดงที่ตำแหน่งของเคอร์เซอร์ได้ ทำให้เรียกเมนูที่ใช้บ่อยได้ทันที เช่น ออปเจ็กสแนปเมนู (Object Snap Menu)

-มีไดอะลอกบล็อก แบบใหม่จำนวนมาก เพื่อใช้แทนที่การเลือกออปชั่น (Options)แบบธรรมดา

-ไดอะลอกบล็อก ที่สามารถโปรแกรมได้โดยใช้ ออโตลิปส์ (AutoLISP) และ ADSและยังมีภาษาใหม่คือ ภาษาการควบคุมไดอะลอกบล็อก (Dialogue Control Langues (DCL)) ที่สามารถจะใช้สร้างไดอะลอกบล็อก อันใหม่ได้

-คำสั่งการทำงานกับไฟล์แบบใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-เปิดไฟล์ (Open) ใช้เลือกชื่อดรออิงจากไดอะลอกบล็อกแบบใหม่

-บันทึก (Save) ใช้เซฟดรออิง (Save drawing) ปัจจุบันเป็นชื่อใหม่ และเข้าดรออิงใหม่นั้นเพื่อทำงานต่อได้ทันที

-คิวเซฟ (Qsave) ใช้เซฟดรออิง ปัจจุบันโดยไม่ต้องการให้ถามชื่อไฟล์

-กำหนดรูปแบบ (Configure) ทำให้สามารถทำการกำหนดรูปแบบของออโตแคดใหม่ (Reconfigure AutoCAD) ได้โดยไม่ต้องออกจาก ดรออิงอิดิเตอร์

1.2.4 เพิ่มประสิทธิภาพการเลือกวัตถุ

-ขั้นตอนการเลือกวัตถุแบบใหม่

-เปลี่ยนขั้นตอนการเลือกใหม่จาก เรียกคำสั่งแล้วเลือกวัตถุเป็น เลือกวัตถุก่อน แล้วจึงเรียกคำสั่ง ทำให้มีความยืดหยุ่นในการใช้งานดีกว่ามาก

-Entity Grip เป็นลักษณะ (Feature) ใหม่อย่างแท้จริง ซึ่งจะอนุญาตให้เราทำการ

-Stretch,move,copy,rotate,scale และ mirror วัตถุได้ โดยไม่จำเป็นต้องเข้าคำสั่งในการแก้ไข

-สามารถ Snap ไปบนจุดอ้างอิงหลักได้ทันที เช่น endpoint,midpoint,quadrant,centers

-ทำให้สามารถปรับขนาดของมิติ (Dimensions) ได้ทันที

-เพิ่มวิธีการเลือกวัตถุแบบใหม่ที่มีประสิทธิภาพสูงขึ้นมาก คือ polygon,fence,หรือเลือกทุกๆวัตถุในดรออิงที่ไม่ได้ถูก freeze

-การแยกวัตถุที่ต้องการ (Entity Filtering) เป็นโบนัสรูทีน (Bonus routine)ทำให้สามารถแยกวัตถุที่ต้องการได้อย่างรวดเร็ว และสะดวกในการใช้งาน โดยสามารถเลือกวัตถุได้ดังนี้

-เลือกตามชนิดของวัตถุ (entity type)

-เลือกตามเลเยอร์ (Layer)

-เลือกตามข้อกำหนดที่ต้องการ โดยสามารถมีข้อกำหนดในการเลือกรวมกันได้หลายข้อ

1.2.5 การเพิ่มประสิทธิภาพที่เห็นได้อย่างชัดเจน

-ปรับสเกล (Scale) ของแบบของเส้น (Linetype) ในโมเดลสเปซ (Model space) โดยอัตโนมัติ ทำให้สามารถกำหนดสเกล (fixed scale) ในเปเปอร์สเปซ (Paper space) ได้ ทำให้การใช้ Viewports ที่มีสเกลการ Zoom ที่ต่างกันไม่มีผลกระทบกับสเกลของแบบของเส้น

-การใส่ลวดลายแบบใหม่

-สามารถใส่ลวดลายในพื้นที่ว่างได้ทันที โดยไม่ต้องสร้างเส้นล้อมรอบก่อน

-สามารถพรีวิว (Preview) ดูก่อนได้ว่าลายแน่น หรือชิดกันเกินไปหรือไม่ก่อนใส่จริง ทำให้ไม่ต้องเสียเวลาทำใหม่ แบบในรุ่นก่อน

-แบบของเส้น ที่ไม่ต่อเนื่องสามารถคำนวณให้ผ่านจุดต่อของ Polyline ได้ ทำให้ได้เส้น Polyline ที่สวยกว่าและไม่ขาดตรงรอยต่อ

-Dimension Dragging สามารถเปลี่ยนตำแหน่งการจัดมิติได้ด้วยการลาก(Dragging) ทำให้สามารถจัดตำแหน่งของมิติใหม่ได้ง่าย

-วัตถุที่อยู่ในบล็อก (Block) และ Xrefs สามารถเลือกได้ เพื่อทำการใส่ dimension

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-Reference Dimensions เพิ่มการล้อมข้อความของ dimension ด้วยกรอบสี่เหลี่ยม ได้ซึ่งใช้มากในงานวิศวกรรม

1.2.6 เพิ่มประสิทธิภาพในการเขียนแบบ

-สามารถอ้างอิง External Reference Drawing ได้ แม้ในขณะที่ Referencedrawing นั้นถูกแก้ไขอยู่

-สามารถ lock layer ได้ดังนี้

-Layers สามารถ Locked ไว้ชั่วคราวได้

-วัตถุบน layers ที่ถูก locked จะไม่ถูกแก้ไข หรือ ลบ แต่สามารถที่จะเห็นได้ และสามารถทำ Object snap ได้ตามปกติ

-ตั้งเวลาบันทึกข้อมูล โดยอัตโนมัติ เพื่อป้องกันข้อมูลสูญหายโดยอุบัติเหตุ

-ควบคุมคำสั่ง Dview โดยใช้อุปกรณ์ชี้ (mouse) ทั้งหมด ทำให้การสร้างมุมมองทำได้สะดวกมาก

-การเพิ่มการกำหนดให้โดยอัตโนมัติ (Defaults) แบบใหม่เข้าไปในคำสั่งเขียนแบบ เช่น Circle,Fillet,Arc

-Zoom window เป็นการกำหนดให้โดยอัตโนมัติ การใช้ Zoom แล้วเลือกจุดสองจุด จะเป็นการ Zoom Window ทันที

1.2.7 เครื่องมือและภาษาในการพัฒนา

-ออโตลิปส์ ทำการโหลดและทำงานได้เร็วกว่า สามารถทำงานได้จากคอมมานด์พร้อม (Command prompt)

-Real mode ADS สามารถใช้คอมไพล์เลอร์(Compiler) ที่มีราคาไม่แพงและหาได้ง่าย คือ เทอร์โบซี (Turbo C),ไมโครซอฟท์ซี (Microsoft C) อีกทั้งยังทำให้ไม่ต้องใช้เครื่องมือ (Tools) เพิ่มเติมหลายตัว ดังที่ใช้ในคอมไพล์เลอร์แบบ Protected mode ADS

1.2.8 ปรับปรุงวิธีการพล็อต(Plot) ใหม่ทั้งหมด

-สามารถพล็อตโดยไม่ต้องออกจากตรออิงอิดิเตอร์

-สามารถปรับให้มี สี,แบบของเส้น,ความเร็วปากกา ได้ถึง 255 แบบ

-สามารถตั้งการกำหนดรูปแบบของเครื่องพล็อต (plotter configurations) ได้หลายแบบพร้อมกัน และใช้สลับไปมาได้ โดยไม่ต้องตั้งใหม่ทุกครั้ง

-ADI ดีไวซ์ไดเวอร์(Device Driver) ชุดใหม่กว่า 20 ตัว ซึ่งใช้กับมาตราฐานกระดาษได้หลายแบบ คือ ANSI,DIN,ISO

-สามารถทำพล็อตพรีวิว (Plot preview) ดูก่อนได้ ก่อนที่จะพล็อตจริง ทำให้ไม่เสียกระดาษในการลองพล็อต

1.2.9 ลักษณะเพิ่มประสิทธิภาพในการใช้งานเฉพาะด้าน

-GIS (Geographic Information Systems)

-multi-point digitizer calibration

-AEC และพรีเซนเตชันกราฟิก (Presentation Graphics)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-รวม Shading & Rendering ซึ่งเป็นส่วนหนึ่งในออโตเซด (AutoShade) ไว้ภายในเป็นมาตราฐานสามารถใช้ได้ทันที สำหรับการกำหนดวัสดุให้กับพื้นผิวโมเดล (models) ยังคงต้องทำในออโตเซด

-การออกแบบทางเครื่องกล (Mechanical Design) & การวาด(Drafting)

-รวมรีเจียสโมเดลเลอร์ (Region Modeler) ไว้เป็นลักษณะมาตราฐาน ทำให้สามารถสร้างรูปสองมิติได้อย่างรวดเร็ว และเปลี่ยนเป็นโซลิดโมเดล (Solid model) 3 มิติในภายหลัง

-สนับสนุนโพสท์สคลิป (PostScript Support)

-สามารถอ่าน แสดงผล และพิมพ์ไฟล์โพสท์สคลิป ได้โดยตรง

-นำลวดลายมาตราฐานของโพสท์สคลิป มาใช้ในดรออิงได้

-นำตัวอักษรแบบโพสท์สคลิปไทป์ 1 มาใช้ในดรออิงได้

1.2.10 แอดวานซ์โมเดลลิงเอกซ์เทนชัน (Advance Modeling Extension) R2.1 มีการเพิ่มประสิทธิภาพดังนี้

-ใช้ความละเอียดของข้อมูลสูงเป็นแบบดับเบิลพรีเซนเดชันโฟล์ดิงพอยน์ (Doble-precision floating-point) ซึ่งมีความละเอียดแม่นยำเท่ากับซอฟท์แวร์บนเวอร์กสเดชัน (Workstations)

-อินเตอร์เฟอร์เรนซ์เช็คกิง (Interference Checking)สามารถตรวจสอบการทับกัน หรือชนกันของโซลิดโมเดล ใน 3D Space ได้ ในการใช้ออกแบบแอสเซมบลี้โมเดล(Assembly models)

-สามารถตัดส่วน (Section) ได้โดยตรง โดยกำหนดระนาบชั่วคราวเพื่อตัดโซลิดโมเดลเป็นสองชิ้น

-สามารถทำแอลินแมนท์ (Alignment) ตัวโซลิดโมเดล เพื่อประกอบโซลิดโมเดลเข้าด้วยกันเพื่อจำลองการประกอบชิ้นงาน

-สามารถใช้ออโตลิปส์ เพื่อสร้างแอปปลิเคชัน (Applications) ที่ทำงานกับโซลิดโมเดลได้ (ปกติจะต้องใช้ API ซึ่งเป็น ADS)

1.2.11 เพิ่มความสามารถในการติดต่อกับฐานข้อมูล

-ภาษาเอสคิวแอล (SQL :Structural Query Language) ทำให้สามารถเชื่อมข้อมูลภาพของออโตแคด เข้ากับข้อมูลภายนอกของโปรแกรมฐานข้อมูลมาตราฐานขนาดใหญ่ได้ เช่น ออราเคิล (Oracle),ดีเบส(dBASE),อินเฟอร์มิก(Informix),พาราดร็อก(Paradox)

-ค้นหาส่วนของดรออิงที่ต้องการโดยใช้เอสคิวแอล ซึ่งจะทำการ ไฮไลท์(High-light)กระพริบเป็นเส้นประบนหน้าจอให้

-สามารถเชื่อมข้อมูลแบบ 2 ทิศทาง(2 way)ได้อย่างสมบูรณ์ เช่นการเปลี่ยนบางฟิลด์ (Field) ของเรคคอร์ด (Record) หนึ่งบนฐานข้อมูล (Database) จะทำให้ดรออิง ส่วนที่สัมพันธ์กันเปลี่ยนคุณสมบัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 เนื้อหาโดยย่อแต่ละบท

บทที่2 จะกล่าวถึงทฤษฎีและการหลักการเบื้องต้นของออโตแคด รีลีส12 ซึ่งได้แก่ คำสั่งที่ใช้ทั่วไปในลักษณะต่างๆ การใช้อุปกรณ์การเชื่อมต่อเพื่อทำการอินพุตหรือเอาท์พุตดรออิง เช่น คีย์บอร์ด,เมาส์ ฯลฯ ,การสร้างไสด์และสคลิปเพื่อความสะดวกในการใช้งาน ,การshade วัตถุที่สร้างขึ้นมา และการใช้ออโตลิปส์ ซึ่งเป็นภาษาที่ใช้ทำการประยุกต์การใช้งานของออโตแคดให้มีประสิทธิภาพมากขึ้น

บทที่ 3 เป็นการอธิบายการคำนวณและการสร้างโปรแกรมที่ทำการสร้างไอโซเมติก 3 มิติจากภาพฉาย 2 มิติทั้ง 3 ด้าน,โปรแกรมการสร้างภาพทรงกรวยและภาพคลี่ของทรงกรวยนั้น,และโปรแกมการสร้างบ้าน 3 มิติจากแบบแปลนในระนาบ 2 มิติ

บทที่ 4 เป็นการอธิบายการใช้งานและผลที่เกิดขึ้นจากการทดลอง ของโปรแกรมที่ได้ทำการสร้างขึ้นจากโปรแกรมออโตลิปส์ ที่กล่าวมาแล้ว

บทที่ 5 เป็นการสรุปและวิจารณ์โครงงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการเบื้องต้น

2.1 คำสั่งในออโตแคด

2.1.1 คำสั่งที่ใช้ทั่วไป

-คำสั่ง HELP หรือ ?

ทำหน้าที่ แสดงคำอธิบายคำสั่งของออโตแคด ที่ต้องการอย่างสั้นๆ

ทางเลือก ให้ระบุชื่อคำสั่งที่ต้องการลงไป ในกรณีใช้แบบทรานสพาเรนท์(Transparent) คำสั่งนี้จะแสดงคำอธิบายของคำสั่งที่ใช้อยู่ทันที

-คำสั่ง END

ทำหน้าที่ เก็บข้อมูลที่เขียนลงดิสค์และออกจากดรออิงอิดิเตอร์ กลับไปที่เมนเมนู

-คำสั่ง ENDSV

ทำหน้าที่ เหมือนคำสั่ง END แต่จะเก็บภาพสุดท้ายก่อนเก็บข้อมูล (Vector File) เอาไว้ ด้วยคำสั่งนี้มีผลต่อเมื่อไม่มีการแก้ไขดรออิง เลยในครั้งนั้น หากมีการแก้ไขใดๆ ก่อนใช้คำสั่งนี้ ผลลัพธ์จะเหมือนคำสั่ง END ปกติ

-คำสั่ง QUIT

ทำหน้าที่ ออกจากดรออิงอิดิเตอร์ และกลับไปที่เมนเมนู โดยไม่เก็บข้อมูลที่แก้ไขลงดิสค์

ทางเลือก จะออกไปได้โดยการตอบ Y (Yes) เท่านั้น

-คำสั่ง SAVE

ทำหน้าที่ เก็บข้อมูลลงดิสค์โดยไม่ออกจากดรออิงอิดิเตอร์

ทางเลือก ระบุชื่อไฟล์ที่ต้องการเก็บข้อมูล

-คำสั่ง STATUS

ทำหน้าที่ แสดงรายละเอียดของสถานะต่างๆของโปรแกรมในขณะนั้น

-คำสั่ง LIMITS

ทำหน้าที่ กำหนดขนาดของพื้นที่ที่ใช้เขียน

ทางเลือก <Value> -กำหนดค่าโคออร์ดิเนตของมุมล่างซ้ายและมุมบนขวาของพื้นที่ที่ต้องการ

ON -ไม่ยอมให้มีการเขียนนอกเหนือขอบเขตที่กำหนดไว้

OFF -ยอมให้มีการเขียนนอกเหนือขอบเขตที่กำหนดได้

-คำสั่ง UNITS

ทำหน้าที่ กำหนดหน่วยของการวัดระยะทางและมุมที่ใช้ในดรออิง

-คำสั่ง MENU

ทำหน้าที่ กำหนดชื่อของเมนูไฟล์ (Menu file) (xxx.MNU) ที่ต้องการจะโหลดเข้ามาใช้

-คำสั่ง FILES

ทำหน้าที่ เข้าสู่รายการหน้าที่เกี่ยวกับการจัดการไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-คำสั่ง TIME

ทำหน้าที่ แสดงเวลาปัจจุบัน และเวลาที่เกี่ยวข้องกับดรออิง ที่กำลังใช้งานอยู่

ทางเลือก Display -แสดงรายการเวลาปัจจุบันและเวลาที่ใช้ในดรออิงนี้

ON -เริ่มให้มีการนับเวลาขณะใช้ดรออิงอยู่

OFF -หยุดการนับเวลา

RESET -ปรับเวลาที่นับให้เป็น 0

-คำสั่ง 'SETVAR

ทำหน้าที่ เปลี่ยนแปลงค่าตัวแปรต่าง ๆของโปรแกรม โดยเฉพาะตัวแปรบางตัวไม่สามารถกำหนดได้ด้วยคำสั่งมาตราฐานของโปรแกรม

ทางเลือก ใส่ชื่อตัวแปรที่ต้องการเปลี่ยนแปลง

-คำสั่ง SHELL หรือ SH

ทำหน้าที่ ออกจากโปรแกรมชั่วคราวไปที่ดอส (DOS)

ทางเลือก กลับเข้ามาสู่โปรแกรมที่ต้องการเปลี่ยนแปลง

-คำสั่ง RENAME

ทำหน้าที่ เปลี่ยนแปลงชื่อของชนิดวัตถุที่มีอยู่

ทางเลือก ระบุชนิดของวัตถุที่ต้องการเปลี่ยนชื่อ มีดังนี้

/Block/LAyer/LType/Style/View

-คำสั่ง PURGE

ทำหน้าที่ ลบข้อมูลของชนิดวัตถุที่ไม่ได้ใช้ออกไป ต้องใช้เป็นคำสั่งแรกที่เริ่มเข้ามาในดรออิง เท่านั้น

ทางเลือก ระบุวัตถุชนิดที่ต้องการลบออกไป มีดังนี้

Block/LAyer/LType/SHapes/STyle/All

-คำสั่ง PLOT

ทำหน้าที่ ส่งข้อมูลของดรออิงไปยังเครื่องเขียน

-คำสั่ง PRPLOT

ทำหน้าที่ ส่งข้อมูลของดรออิง ไปยังเครื่องพิมพ์แบบดอดเมตริกซ์

-คำสั่ง UNDEFINE

ทำหน้าที่ ยกเลิกคำสั่งในออโตแคด บางคำสั่งชั่วคราว โดยการระบุชื่อคำสั่งที่ต้องการยกเลิกลงไป หลังจากนั้นหากมีการเรียกใช้คำสั่งที่ยกเลิกนี้ โปรแกรมจะแสดงข้อความว่าไม่รู้จักคำสั่งดังกล่าว

-คำสั่ง REDEFINE

ทำหน้าที่ เรียกคืนคำสั่งที่ถูกยกเลิกจากคำสั่ง UNDEFINE กลับคืนมา ให้ใช้งานได้เหมือนเดิม โดยการระบุชื่อคำสั่งที่ต้องการลงไปเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 คำสั่งที่ใช้ในการเขียน

<point>

-คำสั่ง LINE

ทำหน้าที่ เขียนเส้นตรง

ทางเลือก <point> -กำหนดจุดที่จะลากเส้นไปถึง

U <Undo> -ยกเลิกเส้นส่วนสุดท้ายที่เขียน

C <Close> -ลากเส้นจากจุดสุดท้ายไปยังจุดเริ่มต้น

-คำสั่ง POINT

ทำหน้าที่ เขียนจุด

ทางเลือก จุดที่ถูกเขียนอาจมองเห็นยาก เราสามารถกำหนดเครื่องหมายบนจุดนั้นๆให้มองเห็นง่ายขึ้น เช่น กากบาท วงกลม เป็นต้น (เครื่องหมายนี้ไม่เกี่ยวกับการพล็อต) โดยที่ PDMODE จะควบคุมเครื่องหมายที่จะเกิดขึ้น โดยมีค่าต่างๆดังนี้

0 -แสดงเป็นจุดแบบปกติ

1 -ไม่แสดงอะไรเลย

2 -แสดงเป็นเครื่องหมายกากบาท เหมือนสัญลักษณ์การกำหนจุด(blip)

3 -แสดงเป็นกากบาทไขว้ 45 องศา

4 -แสดงเป็นเส้นขีดตั้ง

และหากรวมค่าตัวเลขต่อไปนี้เข้าไปด้วย จะแสดงเครื่องหมายดังนี้

32 -เพิ่มวงกลมรอบจุดที่กำหนด

64 -เพิ่มสี่เหลี่ยมรอบจุดที่กำหนด

96 -เพิ่มทั้งวงกลมและสี่เหลี่ยมรอบจุดที่กำหนด

ส่วนค่าของ PDSIZE จะเป็นค่าที่ใช้ควบคุมขนาดของเครื่องหมายที่เกิดจากตัวแปร PDMODE โดยที่หากเราให้ค่าบวก ตัวเลขนั้นจะเป็นค่าเปอร์เซ็นต์สัดส่วนกับจอภาพ ในกรณีนี้ขนาดของเครื่องหมายไม่เปลี่ยนแปลงไม่ว่าจะแสดงในขอบเขตใดก็ตาม

-คำสั่ง CIRCLE

ทำหน้าที่ เขียนวงกลม

ทางเลือก center point & radius หรือ diameter

-กำหนดจุดศูนย์กลางและรัศมีหรือเส้นผ่านศูนย์กลาง

2P -กำหนดจุด 2 จุดบนปลายเส้นผ่านศูนย์กลาง

3P -กำหนดจุด 3 จุดบนเส้นรอบวง

TTR -กำหนดวัตถุที่จะให้วงกลมไปสัมผัส 2 จุด และรัศมีของวงกลมที่จะเขียน

-คำสั่ง ARC

ทำหน้าที่ เขียนส่วนโค้งของวงกลม

ทางเลือก ต้องให้ข้อมูล 3 ตัวในการเขียน โดยมีการเขียนหลายวิธีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033286

-กำหนดจุด 3 จุดบนส่วนโค้ง
-กำหนดจุดตั้งต้น จุดศูนย์กลาง และจุดสิ้นสุดของส่วนโค้ง
-กำหนดจุดตั้งต้น จุดศูนย์กลาง และมุมภายในของส่วนโค้ง
-กำหนดจุดตั้งต้น จุดศูนย์กลาง และความยาวของคอร์ด
-กำหนดจุดตั้งต้น จุดสิ้นสุด และรัศมีของส่วนโค้ง
-กำหนดจุดตั้งต้น จุดสิ้นสุด และมุมภายในของส่วนโค้ง
-กำหนดจุดตั้งต้น จุดสิ้นสุด และทิศทางตั้งต้น
-การต่อเชื่อมระหว่างเส้นตรงและส่วนโค้ง

-คำสั่ง TRACE
ทำหน้าที่ เขียนเส้นตรงที่มีความหนาของเส้น
Trace width -กำหนดความกว้างของเส้นที่จะเขียน
<point> -หลังจากกำหนดความหนาแล้วจะถามถึงจุดที่จะลากเส้นไป เหมือนกับคำสั่ง LINE แต่จะเขียนเส้นในส่วน หลังจากที่กำหนดตำแหน่งของส่วนถัดไปแล้วเพราะโปรแกรมจะรอเพื่อปาดมุมของเส้น(เส้นมีความหนา) เพื่อให้สัมพันธ์กับตำแหน่งถัดไป

-คำสั่ง SOLID
ทำหน้าที่ เขียนรูปหลายเหลี่ยมตัน ต้องการอย่างน้อย 3 ตำแหน่ง ในการเขียนรูปสามเหลี่ยม และ 4 จุดสำหรับสี่เหลี่ยม การเขียนรูปสี่เหลี่ยมตัน จุดที่ 1 และ 4 ต้องอยู่ในมุมตรงข้ามกันเสมอ

-คำสั่ง TEXT
ทำหน้าที่ เขียนตัวอักษรในตำแหน่งที่กำหนด
ทางเลือก <point> -กำหนดจุดเริ่มต้นของข้อความ
A<Align> -กำหนดขอบเขตที่จะเขียนข้อความ
C<Center>-กำหนดจุดกึ่งกลางของข้อความ
F<Fit> -กำหนดขอบเขตที่จะเขียนข้อความและความสูงของอักษร
M<Middle>-กำหนดจุดกึ่งกลางของข้อความ(กึ่งกลางความสูงด้วย)
R<Right> -กำหนดจุดปลายของข้อความ
S<Style> -เลือก Style ของอักษรที่จะใช้

-คำสั่ง STYLE
ทำหน้าที่ สร้างแบบ (Style) อักษรใหม่จากฟอนต์ไฟล์ (Font File) (xxx.SHX)ที่เลือก
ทางเลือก ? -แสดงรายชื่อ Style ที่มีอยู่ในขณะนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<name> -สร้าง Style ขึ้นมาใหม่ในชื่อที่กำหนดสำหรับความสูงของอักษรนั้น หากกำหนดความสูงไว้ในขณะสร้างแบบความสูงนั้นจะถูกใช้ตลอด (ปรับไม่ได้) หากกำหนดความสูงให้ เท่ากับ 0 เมื่อเรียกใช้แบบนี้จะสามารถกำหนดความสูงได้

-คำสั่ง PLINE

ทำหน้าที่ เขียนเส้น polyline ซึ่งมีคุณสมบัติเหมือน LINE,TRACE,ARC ปนกัน แต่ละครั้งที่เขียนจะถือเป็นชิ้นส่วน 1 ชิ้นเท่านั้น ซึ่งประกอบไปด้วยชิ้นส่วนเล็กๆต่อกัน เรียกว่าเซกเมนต์(segment) หรือเวอร์เทกซ์(vertex) มีความคล่องตัว ในการปรับเปลี่ยนมาก แต่ใช้เวลาในการสร้างภาพนานกว่าวัตถุชนิดอื่นๆ จึงควรใช้เฉพาะที่จำเป็น

คุณสมบัติของ Pline

1. อาจจะลากเป็นเส้นขีด/จุดได้
2. อาจจะมีความกว้างเท่ากัน (เหมือนกัน Trace) หรือ เรียวได้
3. polyline กว้างๆอาจจะทำวงกลมเหมือนขนมโดนัทได้
4. อาจจะเอาเส้นตรงและส่วนโค้งมาต่อกันเป็นรูปเหลี่ยมได้
5. Polyline อาจจะแก้ไข นำมาต่อกันได้
6. สามารถจะทำรูปมนหรือตัวมุมได้
7. สามารถทำส่วนโค้งให้พอดีกับส่วนอื่นได้
8. สามารถหาพื้นที่และเส้นรอบรูปของ Polyline ได้

ทางเลือก มีอยู่ 2 โหมด(mode) คือ เส้นตรงและเส้นโค้ง เมื่อเริ่มต้นจะเป็นโหมดของเส้นตรง

โหมดเส้นตรง มีดังนี้ คือ

A<Arc> -เปลี่ยนไปโหมดเส้นโค้ง
C<Close> -เขียนเส้นตรงจากเซกเมนต์สุดท้ายไปยังจุดเริ่มต้น
H<Halfwidth> -กำหนดระยะครึ่งหนึ่งของความกว้างเส้น (จากกึ่งกลาง)
L<Legth> -กำหนดความยาวของเซกเมนต์นั้น
U<Undo> -ยกเลิกเซกเมนต์สุดท้ายที่เขียน
W<Width> -กำหนดความกว้างของเส้น(เหมือนคำสั่ง TRACE)
<point> -กำหนดจุดปลายของเซกเมนต์นั้น

โหมดเส้นโค้ง มีดังนี้ คือ

A<Angle> -กำหนดมุมภายในของส่วนโค้ง
CE<Center> -กำหนดจุดศูนย์กลางของส่วนโค้ง
CL<Close> -เขียนเส้นโค้งจากเซกเมนต์สุดท้ายไปยังจุดเริ่มต้น
D<Direction> -กำหนดทิศทางของส่วนโค้งจากจุดเริ่มต้น
H<Halfwidth> -กำหนดระยะครึ่งหนึ่งของความกว้างเส้น
L<Line> -กลับไปโหมดเส้นตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

R<Radius> -กำหนดรัศมีของส่วนโค้ง

S<Second point> -กำหนดจุดบนส่วนโค้ง

U<Undo> -ยกเลิกความกว้างของเส้น

W<Width> -กำหนดความกว้างของเส้น

<point> -กำหนดจุดปลายของส่วนโค้ง

-คำสั่ง POLYGON

ทำหน้าที่ เขียนรูปหลายเหลี่ยมด้านเท่า (เป็น POLYLINE แบบหนึ่ง)

ทางเลือก หลังจากกำหนดจำนวนด้ายของรูปหลายเหลี่ยมแล้ว(ตั้งแต่ 3 ด้านจนถึง 1024 ด้าน)

<point> -กำหนดจุดศูนย์กลางของรูปหลายเหลี่ยม และมีทางเลือกต่อดังนี้

I<Inscribed> -เขียนรูปหลายเหลี่ยมอยู่ภายในรัศมีของวงกลม

C<Circumscribed> -เขียนรูปครอบรัศมีของวงกลม

E<Edge> -จะกำหนดความยาว และทิศทางของด้านใดด้านหนึ่งโดยการกำหนดจุดเริ่มต้นและจุดปลายเส้น

-คำสั่ง DOUGHNUT หรือ DONUT

ทำหน้าที่ เขียนวงกลมตันหรือมีรูตรงกลาง (เป็นคำสั่ง POLYLINE แบบหนึ่ง)

ทางเลือก กำหนดรัศมีของวงกลมในและภายนอก หากรัศมีภายในเป็น 0 จะเป็นการเขียนวงกลมตัน

-คำสั่ง ELLIPSE

ทำหน้าที่ เขียนรูปวงรี (เป็น POLYLINE แบบหนึ่ง)

ทางเลือก 1-กำหนดจุดปลาย 2 จุดบนแกนใดแกนหนึ่ง กับอีกจุดหนึ่งบนแกนที่เหลือ

2-กำหนดจุดกึ่งกลาง และจุดปลายข้างหนึ่งของแต่ละแกน

3-กำหนดจุดปลาย 2 จุดบนแกนใดแกนหนึ่งและมุมหมุน(rotate)

4-กำหนดจุดกึ่งกลาง และจุดปลายของแกนใดแกนหนึ่ง และมุมหมุน

หมายเหตุ ในกรณีที่อยู่ในโหมด ของไอโซแพน(Isoplane) จะมีทางเลือกเพิ่มเติม คือ

I<Isocircle> -การเขียนวงกลมในภาพไอโซเมตริก (isometric)

-คำสั่ง LOAD

ทำหน้าที่ ดึงข้อมูลของจากเชฟไฟล์ (Shape file) (xxx.SHX) เข้ามาในดรออิง

ทางเลือก ? -แสดงรายชื่อเชฟ (Shape) ที่มีอยู่ในขณะนั้น

<name> -ระบุชื่อไฟล์ที่ต้องการดึงข้อมูลมา

-คำสั่ง SHAPE

ทำหน้าที่ เรียกเอาข้อมูลของเชฟที่ถูกโหลด(Load) แล้วมาใช้

ทางเลือก ? -แสดงรายชื่อเชฟที่เรียกใช้ได้

<name> -ระบุชื่อเชฟที่ต้องการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 คำสั่งที่ใช้ในการแก้ไขเปลี่ยนแปลง

คำสั่งเกี่ยวกับการแก้ไข้เปลี่ยนแปลง(Editor command) ส่วนของภาพหรือวัตถุ (Entity or Object) ส่วนใหญ่จะให้เราระบุว่าต้องการนำส่วนของภาพใดมาใช้บ้าง ในการแก้ไขเปลี่ยนแปลงในครั้งนี้(select object) หลังจากเลือกแล้ว จึงจะเป็นขั้นตอนของการทำงานในคำสั่งนั้นๆ ต่อไป

-คำสั่ง ERASE

ทำหน้าที่ ลบวัตถุที่เลือกทั้งชิ้น

-คำสั่ง OOPS

ทำหน้าที่ เรียกวัตถุที่ถูกลบไปโดยคำสั่ง ERASE (ในครั้งล่าสุด) กลับคืนมาใหม่

-คำสั่ง BREAK

ทำหน้าที่ การลบเพียงบางส่วนของวัตถุ

ทางเลือก ในการเลือกวัตถุ จะถือจุดที่ใช้ชี้วัตถุนั้นเป็นจุดแรกของการตัดทันที และจะถามถึงจุดที่ 2 ทันที หากต้องการกำหนดจุดแรกใหม่ ให้ใส่ "F" แทนการเลือกจุด โปรแกรมจะกลับไปถามถึงจุดแรกใหม่ การ Break เส้นโค้ง จะลบส่วนโค้งจากจุดที่ 1 ทวนเข็มนาฬิกาไปยังจุดที่ 2 เสมอ

-คำสั่ง MOVE

ทำหน้าที่ เคลื่อนย้ายตำแหน่งของวัตถุที่ระบุ

-คำสั่ง COPY

ทำหน้าที่ ลอกแบบวัตถุที่ระบุไปในตำแหน่งที่ต้องการ

ทางเลือก <point> -กำหนด base point

M<Multiple> -กำหนดการลอกแบบซนิดซ้ำหลายครั้ง

-คำสั่ง ROTATE

ทำหน้าที่ หมุนวัตถุที่ระบุโดยรอบจุดหมุนที่กำหนด

ทางเลือก หลังจากกำหนดจุดหมุนแล้ว

<angle> -ระบุมุมที่ต้องการให้หมุนไปจากตำแหน่งเดิม

R<Reference> -กำหนดค่ามุมเดิมของวัตถุ และมุมใหม่ที่ต้องการ

-คำสั่ง SCALE

ทำหน้าที่ เปลี่ยนแปลงมาตราส่วนของวัตถุที่ระบุ

ทางเลือก หลังจากกำหนดจุดอ้างอิง (base point) แล้ว

<Scale Factor> -กำหนดจำนวนเท่าของการเปลี่ยนแปลง

R<Reference> -กำหนดค่าขนาดของเดิมและค่าใหม่ที่ต้องการ

-คำสั่ง MIROR

ทำหน้าที่ การ copy แบบกระจกเงา วัตถุที่ได้ออกมาจะกลับซ้ายเป็นขวา

-การกำหนดระนาบของการสะท้อน(mirror line) จะอยู่ในแนวใดก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-โดยปกติ text จะถูกพลิกกลับด้วย เหมือนมองดูในกระจกเช่นกัน หากต้องการให้ text แสดงออกมาเหมือนปกติ ก็สามารถทำได้ โดยการกำหนดค่าตัวแปรชื่อ MIRRTEXT ให้มีค่าเท่ากับ 0

-คำสั่ง STRETCH

ทำหน้าที่ เปลี่ยนแปลงตำแหน่งบางส่วนของชิ้นวัตถุ เช่น ยืดหรือหดความยาวของเส้น การเลือกวัตถุในการใช้คำสั่งนี้ ต้องใช้โหมดของวินโดว์ หรือครอสซิง (crossing) เท่านั้น สำหรับการใช้โหมดของวินโดว์การใช้คำสั่งนี้จะให้ผลเหมือนกับคำสั่ง MOVE

-คำสั่ง ARRAY

ทำหน้าที่ การลอกแบบซ้ำ โดยมีตำแหน่งในการลอกแบบเป็นไปอย่างมีระเบียบ

ทางเลือก R<Rectangular> -แบบเป็นแถวและคอลัมน์ โดยกำหนดจำนวนแถว คอลัมน์ และ ระยะห่าง ซึ่งหากค่าระยะเป็นบวก การ copy จะ copy ไปทางขวาของวัตถุ(สำหรับคอลัมน์)และขึ้นข้างบน(สำหรับแถว) และกระทำตรงกันข้ามหากใส่ค่าลบ

P<Polar> -แบบวงกลม

-คำสั่ง CHANGE

ทำหน้าที่ เปลี่ยนแปลงคุณสมบัติ หรือตำแหน่งบางส่วนของวัตถุ

ทางเลือก <point> -จุดที่ต้องการให้วัตถุที่เลือกเปลี่ยนตำแหน่ง (บางส่วน) ไปยังจุดนั้น

P<Propoties> -การเปลี่ยนแปลงคุณสมบัติของวัตถุ ดังนี้

C<Color> -สีของวัตถุ

E<Elevation> -ระดับที่เขียนวัตถุนั้นๆ

LA<Layer> -layer ที่เก็บวัตถุนั้น

LT<Ltype> -ชนิดของลายเส้นที่ใช้

T<Thickness> -ความหนา หรือความสูงของวัตถุ

เราสามารถกำหนดสีและชนิดของลายเส้นนั้นได้อย่างไม่จำกัดไม่ว่าจะอยู่ใน layer เดียวกันก็ตาม แต่เพื่อป้องกันความสับสนควรที่จะใช้สีและลายเส้นที่เหมือนกันในแต่ละ layer

-คำสั่ง TRIM

ทำหน้าที่ ใช้สำหรับตัดส่วนเกินของวัตถุที่ไม่ต้องการออก โดยมีขอบเขตของการตัดออกที่แน่นอน

ทางเลือก <Cutting Edge> -ขอบเขตที่เป็นตัวกั้นเขตการตัดวัตถุ

<Select Object>-วัตถุที่จะถูกตัดออกตามขอบเขตที่กำหนด โดยส่วนที่ถูกชี้จะเป็นส่วนที่ถูกตัดออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-คำสั่ง EXTEND

ทำหน้าที่ ยืดวัตถุที่ต้องการไปชนขอบเขตที่กำหนด

ทางเลือก <Select boundary> -กำหนดขอบเขตที่เป็นตัวหยุด

<Select object> -วัตถุที่จะยืดออกไปชนขอบเขตที่เลือก

-คำสั่ง FILLET

ทำหน้าที่ ตัดมุมระหว่างวัตถุ 2 ชิ้น

ทางเลือก <point> -กำหนดส่วนของภาพ 2 ชิ้นที่จะตัดมุมมน

P<Polyline> -ตัดมุมมนให้กับ polyline ทั้งชิ้น

R<Radius> -กำหนดรัศมีในการตัดมุมมน

-คำสั่ง CHAMFER

ทำหน้าที่ ตัดมุมเหลี่ยมของวัตถุ 2 ชิ้น

ทางเลือก <point> -เลือกวัตถุที่จะตัดมุมเหลี่ยม 2 ชิ้น(ระยะที่ 1 จะใช้ตัดชิ้นแรกที่เลือก และระยะที่ 2 สำหรับชิ้นที่ 2)

P<Polyline>-ตัดมุมเหลี่ยม polyline ทั้งชิ้น

D<Distance>-กำหนดระยะการตัดมุม 2 ระยะ

-คำสั่ง OFFSET

ทำหน้าที่ สร้างเส้นขนานกับวัตถุที่ระบุ ด้วยระยะห่างที่กำหนด

ทางเลือก <Distance> -ระยะห่างที่ต้องการ

T<Through> -ต้องการระบุระยะห่างโดยการชี้ด้วยเคอร์เซอร์

-คำสั่ง DIVIDE

ทำหน้าที่ แบ่งส่วนของภาพที่ระบุออกเป็นส่วนๆเท่ากัน ตามจำนวนที่กำหนด

ทางเลือก <Number> -จำนวนส่วนของภาพที่ต้องการ

B<Block>-ต้องการใส่ชิ้นวัตถุประเภทบล็อก(Block)ลงตามจุดที่แบ่ง

การแบ่งนี้ไม่ได้ตัดวัตถุออกเป็นส่วนๆเพียงแต่กำหนดจุด (point) ลงไปเท่านั้น ซึ่งโดยปกติจะมองไม่เห็น เพราะจะซ้อนทับอยู่กับวัตถุที่เลือก นอกจากจะกำหนด PDMODE

-คำสั่ง MEASURE

ทำหน้าที่ แบ่งชิ้นวัตถุที่ B<Block> -ต้องการใส่ชิ้นวัตถุประเภทบล็อกลงตามจุดที่แบ่ง เหมือนกับ

divide คือไม่ได้ตัดแบ่งชิ้นวัตถุที่เลือกออกเป็นส่วนๆเพียงแต่เขียนจุดลงไปตามระยะที่แบ่งเท่านั้น

-คำสั่ง PEDIT

ทำหน้าที่ แก้ไขดัดแปลงส่วนของภาพประเภท Polyline

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ 2D polyline

ทางเลือก C<Close> -ลากเส้นจากเซกเมนต์สุดท้ายไปยังจุดเริ่มต้นสำหรับ polyline ที่เปิดอยู่ (open pline)

O<Open> -ลบเซกเมนต์สุดท้ายออกสำหรับ polyline ที่ปิดอยู่ (Close pline).

J<Join> -ผนวกเส้นชนิดต่างๆ หรือ polyline หลายๆชิ้น ให้เป็น polyline ชิ้นเดียวกัน (เส้นต้องต่อกัน)

W<Width> -แก้ไขความกว้างของ Polyline ทั้งชิ้น

E<Edit vertex>-เข้าสู่ โหมดของการแก้ไขแต่ละเซกเมนต์

F<Fit curve> -ปรับเซกเมนต์ต่างๆให้มีความโค้งที่สัมพันธ์กัน

D<Decurve> -ปรับเซกเมนต์ต่างๆที่โค้งให้เป็นเส้นตรง

U<Undo> -ยกเลิกการแก้ไขถอยหลังไปทีละขั้น(ได้จนถึงจุดแรกที่เริ่มแก้ไข)

X(eXit> -ออกจากการแก้ไข

อิดิตเวอร์เทกซ์โหมด(Edit vertex mode) เมื่อเข้ามาอยู่ในโหมดนี้จะมีการเขียนกากบาทลงบนเซกเมนต์แรกของ polyline เครื่องหมายนี้เป็นเครื่องหมายที่แสดงถึงตำแหน่งของเซกเมนต์ที่ จะใช้ในการแก้ไขด้วยคำสั่งใดๆ หลังจากเลือกคำสั่งย่อยในการแก้ไขแล้ว โปรแกรมจะแสดงข้อความว่า "Next/Previous/Go/eXit/<N>" ซึ่งหมายถึงการให้กำหนดระยะ(Range)ของเซกเมนต์ในการแก้ไขโดยมี

ความหมายดังนี้

Next -เลื่อนกากบาทไปข้างหน้าอีก 1 เซกเมนต์

Previous -เลื่อนกากบาทถอยหลัง 1 เซกเมนต์

Go -ดำเนินการตามคำสั่ง

eXit -ยกเลิกการทำงาน และออกจากคำสั่งย่อย

หากไม่มีข้อความต่างๆเหล่านี้ หมายความว่า คำสั่งย่อยนั้นๆใช้กับการแก้ไขทีละเซกเมนต์เท่านั้น คำสั่งย่อยต่างๆประกอบไปด้วย

Next/Previous -เลื่อนกากบาทไปข้างหน้า หรือถอยหลังทีละ 1 เซกเมนต์

B<Break> -ตัดแบ่ง polyline ออกเป็น 2 ชิ้น หากมีการกำหนดระยะในการ break แล้ว เซกเมนต์ในช่วงระยะที่กำหนดจะถูกลบออกไปด้วย

I<Insert> -เพิ่มเซกเมนต์ใหม่เข้าไป

M<Move> -ย้ายตำแหน่งจุดต่อของเซกเมนต์ไปยังที่ใหม่

R<Regen> -กำหนดให้มีการสร้างภาพใหม่สำหรับแสดงผลที่เกิดจากการแก้ไขความหนาของแต่ละเซกเมนต์

S<Straighten> -เปลี่ยนแปลงเซกเมนต์ที่อยู่ระหว่างระยะให้เป็นเซกเมนต์ตรงชิ้นเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T<Tangent> -กำหนดทิศทางของแต่ละเซกเมนต์ เอาไว้ เมื่อมีการสั่ง fit curve จะมีการใช้ทิศทางที่เคยกำหนดไว้ในการวาดเส้นโค้ง

W<Width> -แก้ไขความหนาของเส้นแต่ละเซกเมนต์

x<eXit> -ออกจากคำสั่งย่อยการแก้ไขเซกเมนต์

สำหรับ 3D-polyline

ทางเลือก C<Close> -ลากเส้นจากเซกเมนต์ สุดท้ายไปยังจุดเริ่มต้น สำหรับ polyline ที่เปิดอยู่

D<Decurve> -ปรับเส้น polyline จากการปรับโค้ง(spling) กลับมาเป็นตัวต้นฉบับ

E<Edit vertex>-การแก้ไขแต่ละเซกเมนต์ ซึ่งมีทางเลือกย่อยตามรายละเอียดข้างท้าย

O<Open> -ยกเลิกเส้นสุดท้ายของ polyline ที่ปิดอยู่

S<Spling> -การปรับโค้งให้กับเส้น polyline

U<Undo> -ยกเลิกการทำงานในขั้นตอนสุดท้าย

X<eXit> -ออกจากคำสั่ง PEDIT

โหมดการแก้ไขแต่ละเซกเมนต์ <Edit vertex>

B<Break> -กำหนดจุดเริ่มของการตัดแบ่งเส้น polyline ออกจากกัน

G<Go> -สั่งให้ดำเนินการทำการตัดหรือทำให้เป็นเส้นตรงจากทางเลือกย่อยอื่นๆ

I<Insert> -เพิ่มเติมเซกเมนต์ เข้าไปใหม่

M<Move> -ย้ายตำแหน่งของเวอร์เทกซ์ ที่กำหนด

N<Next> -เลื่อนกากบาทไปยังเวอร์เทกซ์ ถัดไป

P<Previous> -เลื่อนกากบาทไปยังเวอร์เทกซ์ ก่อนหน้า

R<Regen> -Regenerate เส้น polyline ใหม่

S<Straighten>-กำหนดจุดเริ่มต้นของการปรับเซกเมนต์ต่างๆให้เป็นเส้นตรงเพียง 1 เซกเมนต์

X<eXit> -ออกจาก อิดิทเวอร์เทกซ์(Edit vertex)

สำหรับ 3D MESH

ทางเลือก D<Desmooth> -ยกเลิกการปรับความโค้งกลับไปสู่ต้นฉบับ

E<Edit> -เข้าสู่โหมดการแก้ไขแต่ละเวอร์เทกซ์

M<M-direction> -เปิดหรือปิดพื้นผิวด้านแกน M

N<N-direction> -เปิดหรือปิดพื้นผิวด้านแกน N

S<Smooth> -ปรับความโค้งให้กับพื้นผิว

U<Undo> -ยกเลิกการทำงานขั้นตอนสุดท้าย

X<eXit> -ออกจากคำสั่ง PEDIT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

mode การแก้ไขแต่ละเซกเมนต์

D<Down> -เลื่อนตำแหน่งกากบาทลง (ในแนวแกน M)

L<Left> -เลื่อนตำแหน่งกากบาทไปทางซ้าย(ในแนวแกน N)

N<Next> -เลื่อนตำแหน่งกากบาทไปยังเวอร์เทกซ์ ถัดไป

P<Previous> -เลื่อนตำแหน่งกากบาทไปยังเวอร์เทกซ์ก่อนหน้านี้

R<Right> -เลื่อนตำแหน่งกากบาทไปทางขวา(ในแนวแกน N)

RE<REgen> -กำหนดให้สร้างพื้นผิวใหม่ตามการแก้ไขที่เกิดขึ้น

U<Undo> -ยกเลิกการทำงานขั้นตอนสุดท้าย

X<eXit> -ออกจากอิดิทเวอร์เทกซ์

-คำสั่ง EXPLODE

ทำหน้าที่ ใช้สำหรับแยกวัตถุประเภทบล็อก หรือ Polyline ให้กลายเป็นส่วนของภาพของ line หรือ arc

-การแตกตัวจะทำได้ทีละระดับเท่านั้น ยกตัวอย่างเช่น Polyline ที่ทำเป็นบล็อก เมื่อใช้คำสั่งนี้ บล็อกจะแตกตัวออกเป็น polyline และเมื่อใช้คำสั่งนี้อีกครั้งหนึ่ง Polyline นั้นจึงจะกลายเป็น line หรือ arc

-ข้อมูลของความหนาและ tangent ของ polyline จะถูกยกเลิก

-บล็อก ที่ insert เข้ามาโดยมีการเปลี่ยนแปลง scale จะใช้คำสั่งนี้ไม่ได้

-คำสั่ง MULTIPLE

ทำหน้าที่ เป็นคำสั่งที่ใช้ร่วมกับคำสั่งอื่นๆ เพื่อให้เกิดการทำคำสั่งนั้นซ้ำไปเรื่อยๆ จนกว่าจะมีการกด Ctrl C เท่านั้น

ทางเลือก ป้อนคำสั่งที่ต้องการซ้ำ ในรูปแบบดังนี้

MULTLPLE <command name>

-คำสั่ง U

ทำหน้าที่ ยกเลิกการทำงานในคำสั่งที่ผ่านมา ทีละคำสั่ง

-คำสั่ง UNDO

ทำหน้าที่ ยกเลิกการทำงานที่ผ่านมา

ทางเลือก <number> -จำนวนคำสั่งที่จะยกเลิกย้อนหลังไป

A<Auto> -กำหนดให้มีการยกเลิกกลุ่มคำที่สั่งจากเมนูโดยการใช้คำสั่ง U เพียงครั้งเดียวหรือไม่

B<Back> -การยกเลิกคำสั่งย้อนหลังไปจนถึงจุดสุดท้ายที่มาร์ค (mark) เอาไว้

C<Control> -การควบคุมการทำงานของ undo มักจะใช้ในกรณีที่มีปัญหาเกี่ยวกับเนื้อที่ของดิสค์ในการเก็บข้อมูล เพราะการทำงาน ของคำสั่ง UNDO นั้นจะใช้เนื้อที่ในดิสค์เพิ่มขึ้นด้วยโดยมีทางเลือกดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A<All> -กำหนดให้มีความสามารถของ undo เต็มที่

N<None> -ยกเลิกความสามารถของ undo ทั้งหมด

O<One> -กำหนดให้คำสั่ง UNDO และ U สามารถย้อนหลังได้เพียงคำสั่งเดียวเท่านั้น

E<End> -ระบุจุดสิ้นสุดของกลุ่มคำสั่ง(group) ที่ต้องการ

G<Group> -ระบุจุดเริ่มต้นของกลุ่มคำสั่ง

การยกเลิกคำสั่งย้อนหลังด้วยคำสั่ง U หากพบรหัส undo end แล้ว จะย้อนหลังไปจนถึงรหัส undo group ทันที มิใช่ยกเลิกเพียงคำสั่งเดียว

M<Mark> -ระบุเครื่องหมาย สำหรับการตรวจสอบการยกเลิกคำสั่งย้อนหลังด้วย undo back เมื่อพบรหัสนี้ การยกเลิกคำสั่งย้อนหลังจะหยุดลง

-คำสั่ง REDO

ทำหน้าที่ ยกเลิกคำสั่ง U หรือ UNDO ที่ผ่านมา (ให้ทำคำสั่งที่ยกเลิกไปนั่นเอง)

หมายเหตุ ต้องใช้หลังคำสั่ง UNDO หรือ U ทันที

-คำสั่ง CHPROP

ทำหน้าที่ ใช้สำหรับการแก้ไขคุณสมบัติของวัตถุที่เขียนโดยเฉพาะ (เป็นหน้าที่ส่วนหนึ่งของคำสั่ง CHANGE)

ทางเลือก C -สี

LA -LAyer ที่เก็บวัตถุนั้น

LT -ชนิดของลายเส้นที่ใช้

T -ความหนา หรือ ความสูงของวัตถุ

2.1.4 คำสั่งที่ใช้ในการสอบถามข้อมูล

-คำสั่ง LIST

ทำหน้าที่ แสดงข้อมูลรายละเอียดและประเภทของชิ้นวัตถุที่เลือก

-คำสั่ง DBLIST

ทำหน้าที่ แสดงข้อมูล รายละเอียด และประเภทของชิ้นวัตถุทั้งหมดที่มีอยู่ในดรออิงนี้

-คำสั่ง DIST

ทำหน้าที่ วัดระยะทาง และมุมระหว่างจุดที่กำหนด

-คำสั่ง ID

ทำหน้าที่ แสดงค่าโคออร์ดิเนตของจุดที่ระบุ

-คำสั่ง AREA

ทำหน้าที่ คำนวณพื้นที่ที่ระบุ

ทางเลือก <point> -กำหนดจุดตามเส้นรอบรูปที่จะวัดพื้นที่

E<Entity> -วัดพื้นที่ของชิ้นส่วนวงกลม ส่วนโค้ง หรือ polyline ที่เลือก

A<Add> -เข้าสู่โหมดการเพิ่มเติมพื้นที่ที่วัดไว้ครั้งล่าสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

S<Subtract>-เข้าสู่โหมดของการหักออก (หักออกจากพื้นที่ครั้งล่าสุด)

2.1.5 คำสั่งที่ใช้ในการแสดงภาพ

-คำสั่ง 'ZOOM

ทำหน้าที่ การขยายหรือย่อภาพที่แสดงตามที่ระบุ

ทางเลือก A<All> -ดูพื้นที่ที่เขียนทั้งหมด (เท่า limits หรือเท่าขนาดของชิ้นงานที่เขียน แล้วแต่อย่างไหนจะมากกว่า)

E<Extent> -ขยาย หรือย่อชิ้นงานทั้งหมดให้เต็มจอภาพ

W<Window> -ขยายบริเวณที่กำหนดออกมาเต็มจอภาพ

C<Center> -กำหนดจุดกึ่งกลางของจอภาพใหม่และความสูง โดยบอกเป็นจำนวนดรออิงยูนิต (drawing unit) หรือ จำนวนเท่าของความเปลี่ยนแปลง(<n>X)

L<Left> -กำหนดจุดมุมล่างซ้ายของจอภาพใหม่และความสูง โดยบอกเป็นจำนวนดรออิงยูนิต หรือจำนวนเท่าของความเปลี่ยนแปลง (<n>X)

P<Previous> -เรียกภาพที่เคยแสดงมาก่อนย้อนหลังไปทีละภาพ(เก็บไว้ได้ 5 ภาพย้อนหลัง)

D<Dynamic> -คล้าย zoom window แต่จะแสดงภาพทั้งหมดออกมา และให้กำหนดเฟรม (frame) ที่ต้องการให้แสดงออกมาเต็มจอภาพ

<n>X -จำนวนเท่าของการเปลี่ยนแปลงจากภาพที่แสดงอยู่

-คำสั่ง PAN

ทำหน้าที่ กวาดจอภาพไปแสดงในบริเวณข้างเคียง

-คำสั่ง 'VIEW

ทำหน้าที่ เรียกดูหรือตั้งชื่อบริเวณของชิ้นงานที่ต้องการ เพื่อความสะดวกในการแสดงภาพในส่วนที่ต้องการได้โดยสะดวกกว่าการใช้คำสั่ง zoom โดยเฉพาะกับบริเวณที่มีการเรียกดูอยู่บ่อยๆ

ทางเลือก ? -ขอดูรายชื่อภาพที่เคยตั้งชื่อเอาไว้

D<Delete> -ลบชื่อภาพที่มีอยู่ออกไป

R<Restore> -เรียกชื่อภาพที่เคยตั้งชื่อเอาไว้ออกมาแสดง

S<Save> -เก็บภาพที่แสดงอยู่ในขณะนั้น ในชื่อที่กำหนด

W<Window> -เก็บภาพที่แสดงอยู่เพียงบางส่วน ในชื่อที่กำหนด

-คำสั่ง 'REDRAW

ทำหน้าที่ สำหรับล้างเครื่องหมายกากบาท(blipmode) ที่เกิดจากการกำหนดตำแหน่ง

-คำสั่ง REGEN

ทำหน้าที่ การกำหนดให้โปรแกรมทำการคำนวณภาพขึ้นมาใหม่ จากข้อมูลที่เก็บเอาไว้ในฐานข้อมูลของโปรแกรม คำสั่งบางอย่างในโปรแกรมจะไม่แสดงผลลัพธ์โดยทันทีจะต้องใช้คำสั่งนี้เสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-คำสั่ง REGENAUTO

ทำหน้าที่ กำหนดให้มีการคำนวณสร้างภาพใหม่จากฐานข้อมูลโดยอัตโนมัติในทันทีที่มีการเปลี่ยนแปลงสถานภาพของวัตถุภายในตรออิง

-คำสั่ง FILL

ทำหน้าที่ กำหนดให้มีการระบายพื้นที่ตัน หรือเส้นที่มีความหนา หรือไม่มี

-คำสั่ง BLIPMODE

ทำหน้าที่ กำหนดให้มีการเขียนกากบาทชั่วคราวลงบนจอภาพ ในขณะที่ทำการกำหนดตำแหน่งเพื่อความสะดวกในการมองเห็น เครื่องหมายเหล่านี้จะถูกล้างออกไปด้วยคำสั่ง REDRAW หรือคำสั่ง REGEN

-คำสั่ง QTEXT

ทำหน้าที่ กำหนดให้วัตถุเป็น TEXT แสดงเฉพาะเส้นกรอบเท่าขนาดของ text หรือแสดงรายละเอียดของตัวอักษรทั้งหมด

-คำสั่ง DRAGMODE

ทำหน้าที่ กำหนดสถานะของการลาก (drag) (การสร้างภาพที่จะเกิดขึ้นจากคำสั่งใดๆ ในขณะที่เลื่อนเคอร์เซอร์)

ทางเลือก ON -กำหนดให้มีการลากในจุดที่มีการสั่งโดยผู้ใช้

OFF -ยกเลิกการลากทุกกรณี

A<Auto> -กำหนดให้มีการลากโดยอัตโนมัติ ทันทีที่ทำให้

-คำสั่ง VIEWRES

ทำหน้าที่ กำหนดความละเอียดในการแสดงผลของส่วนโค้งบนจอภาพ และการมีระบบ fast zoom หรือไม่มี ระบบ fast zoom จะใช้หน่วยความจำส่วนหนึ่งในการเก็บภาพของตรออิงใน การแสดงภาพแต่ละครั้ง โดยเฉพาะการเปลี่ยนบริเวณแสดงผล โปรแกรมจะใช้การ regen ซึ่งใช้เวลานานกว่ากันมาก circle zoom percent เป็นการควบคุมความละเอียดของส่วนโค้งที่จะแสดงโดยการแบ่งวงกลมออกเป็นส่วนๆ(sector) แต่ละส่วนคือเส้นตรงสั้นๆ หากมีค่าน้อยหมายถึงจำนวนของเซกเตอร์ (sector) ที่มีจะน้อยไปด้วย ซึ่งจะทำให้ภาพที่แสดงออกมาหยาบแต่มีความเร็วในการแสดงผลสูง (ค่าเท่ากับ 100 จะเป็นค่าปกติของโปรแกรม)

-คำสั่ง LAYER

ทำหน้าที่ การควบคุมสถานะต่างๆของแผ่นงานแต่ละชั้น (layer)

คุณสมบัติของชั้น

1. ชื่อของชั้น (layer name) คือ ชื่อที่เราใช้อ้างถึงในคำสั่ง ชื่อจะต้องยาวไม่เกิน 31 ตัวอักษร อาจจะเป็นตัวหนังสือ ตัวเลข เครื่องหมาย $ ขีดเส้นใต้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สถานะ(state) ถ้าเรากำหนดให้สถานะเป็น OFF วัตถุที่เขียนอยู่ในชั้นนั้นๆ จะไม่ปรากฏบนจอและไม่เขียนในเครื่องพล็อต แต่ไม่ว่าจะ ON หรือ OFF สิ่งที่เราเขียนลงไปจะยังคงถูกเก็บอยู่ในแฟ้มข้อมูลนั้น

3. การแช่แข็ง/การละลาย (Freeze/Thaw) ในการกำหนดสถานะ OFF ให้กับชิ้นงานต่างๆ โปรแกรมยังคงต้องคำนวณตำแหน่งการแสดงผลอยู่ดี เพียงแต่ไม่เขียนลงบนจอภาพ สำหรับสถานะการแช่แข็ง นั้น โปรแกรมจะไม่สนใจหรือไม่คำนวณข้อมูลบนชั้นนั้นๆเลย แต่หากต้องการให้แสดงข้อมูลออกมาอีกครั้งต้องใช้วิธีการละลาย อย่างเดียวเท่านั้น

4. หมายเลขสี (Color number) ปกติจะเป็นสีหมายเลข 7 คือ สีขาว เราสามารถกำหนดได้ตั้งแต่ 1 ถึง 255 สำหรับอุปกรณ์แสดงผล บางชนิด

5. ชื่อของชนิดของเส้น (Linetype name) คือ ชื่อที่กำหนดเป็นเส้นขีดจุด สำหรับส่วนของภาพที่เขียนด้วยเส้น ส่วนโค้ง วงกลม และ polyline ชื่ออาจจะยาวได้ถึง 31 ตัวอักษร ชั้นต่างกันอาจจะมีชื่อซ้ำกันได้ ปกติจะใช้เส้นต่อเนื่อง (continuous) คือ เส้นทึบธรรมดา

ทางเลือก

?	-แสดงรายชื่อของ layer ที่ต้องการ
S<Set>	-กำหนด layer ที่จะเรียกมาเป็นแผ่นใช้งาน (current layer)
N<New>	-สร้าง layer ขึ้นมาใหม่
M<Make>	-สร้าง layer ขึ้นมาใหม่และกำหนดให้เป็นแผ่นใช้งาน
ON	-กำหนด layer ที่จะให้แสดงผลทางจอภาพ
OFF	-กำหนด layer ที่จะไม่ให้แสดงออกทางจอภาพ
C<Color>	-กำหนดสีให้กับ layer ที่ระบุ
L<Ltype>	-กำหนดชนิดของลายเส้นให้กับ layer ที่ระบุ
F<Freeze>	-คล้ายกับ off แต่กำหนดให้ไม่มีการสร้างภาพบน layerที่ระบุ
T<Thaw>	-ยกเลิกการแช่แข็ง ใน layer ที่ระบุ

-คำสั่ง COLOR

ทำหน้าที่ กำหนดสีให้กับชิ้นวัตถุที่เลือก

-คำสั่ง LINETYPE

ทำหน้าที่ เลือกใช้หรือสร้างชนิดของเส้นสายที่ระบุ

ทางเลือก	?	-เรียกดูรายชื่อชนิดของเส้นสายที่มีอยู่
	C<Create>	-การสร้างเส้นสายขึ้นมาใหม่
	L<Load>	-การดึงข้อมูลเส้นสายที่มีอยู่ในดิสค์เข้ามาในดรออิง
	S<Set>	-การกำหนดชิ้นวัตถุที่ระบุให้ใช้ linetype ที่กำหนด

-คำสั่ง LTSCALE

ทำหน้าที่ กำหนดขนาดมาตราส่วนของเส้นสายที่ใช้ ให้เหมาะสมกับการแสดงผล หรือการส่งชิ้นงานไปพล็อต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การเปลี่ยนแปลงระบบโคออร์ดิเนต (Coordinate (UCS))

การควบคุม UCSICON หากสังเกตให้ดีในขณะที่อยู่ดรออิงอิดิตเตอร์ จะเห็นว่าที่มุมล่างซ้ายของจอภาพ จะมีสัญลักษณ์แสดงระบบโคออร์ดิเนต (Coordinate) อยู่เรียกว่า UCSICON จะเป็นตัวบอกจุดกำเนิต (Origin) (0,0) ทิศทางแกน X และแกน Y ว่าอยู่ที่ใด อักษร W ที่อยู่ใน UCSICON จะบอกให้ทราบว่ากำลังใช้ระบบโคออร์ดิเนต WCS(World Coordinate System) อยู่

การควบคุม UCSICON ทำได้ดังนี้

ทำหน้าที่ กำหนดให้มีการแสดงเครื่องหมายบอกทิศทางของโคออร์ดิเนต

ทางเลือก ON -กำหนดให้ UCSICON แสดงผล

OFF -กำหนดให้ปิด UCSICON ไม่แสดงผล

All -แสดง UCSICON ทุกๆ views ports

Noorigin -แสดง UCSICON ที่มุมล่างซ้ายของ viewport

ORigin -แสดง UCSICON ที่จุดกำเนิด (0,0,0) ไม่ว่าจะย้าย

จุดกำเนิดไปอยู่ที่ใดก็ตามUCSICONจะไปแสดงที่จุดนั้นแต่ถ้าเนื้อที่ไม่พอแสดงผล UCSICON จะไปอยู่ที่มุมล่างซ้ายของ viewport การควบคุม UCSICON นี้ไม่มีผลต่อการเปลี่ยนแปลงระบบโคออร์ดิเนต แต่จะเป็นเพียงตัวบอกสถานะ ของระบบโคออร์ดิเนตเท่านั้น เช่น ในกรณีที่กำหนด UCSICON จะไปแสดงผลที่จุดกำเนิดใหม่ เพื่อให้ผู้ใช้ทราบว่าขณะนี้จุดกำเนิดอยู่ตำแหน่งใด

-คำสั่ง UCS (User Coordinate System)

ทางเลือก

Origin -ย้ายจุดกำเนิด (0,0,0) ไปอยู่ตำแหน่งอื่นโดยอ้างอิงจากจุดกำเนิดปัจจุบัน

ZAxis -ย้ายจุดกำเนิด (0,0,0) ไปอยู่ตำแหน่งอื่น และสามารถกำหนดทิศทางของ แกน Z ได้ด้วย

3point -ย้ายจุดกำเนิด (0,0,0) ไปอยู่ตำแหน่งอื่น และสามารถกำหนดทิศทางของแกน X และแกน Y ได้

Entity -กำหนดระบบโคออร์ดิเนต ตามวัตถุที่เลือก โดยระนาบ X-Y ของ UCS จะขนานกับระนาบ X_Y ของวัตถุที่เลือก

View -กำหนดให้ระนาบ X-Y ของระบบโคออร์ดิเนตขนานกับจอ (ระนาบ X-Y ตั้งฉากกับแนวการมอง) แต่จุดกำเนิดไม่เปลี่ยนแปลง

X/Y/Z -ทำการหมุนระบบโคออร์ดิเนตรอบแกน X,Y หรือ Z ตามที่เรากำหนด

Prev -กลับไปยังระบบโคออร์ดิเนต ที่เพิ่งผ่านมา (Previous)

Restore -นำระบบโคออร์ดิเนตที่ได้ Saveไว้มาใช้

Save -เก็บระบบโคออร์ดิเนตปัจจุบันไว้เพื่อเรียกมาใช้ภายหลังได้ด้วย คำสั่ง RESTORE

Del -ลบระบบโคออร์ดิเนต ที่ได้ save ไว้

? แสดงรายชื่อระบบโคออร์ดิเนต ที่ได้ save ไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<World> กลับไปใช้ระบบโคออร์ดิเนต ตาม World Coordinate System

-การแบ่งจอภาพออกเป็นส่วนๆ (Viewports)

เราสามารถแบ่งการแสดงผลบนจอภาพออกเป็นส่วนๆ ซึ่งในแต่ละส่วนเรียกว่า viewport ซึ่งจะช่วยให้การมองภาพเป็นไปได้หลายๆวิว(view) ในเวลาเดียวกัน และ การควบคุมการแสดงผลจะแยกเป็นอิสระต่อกันในแต่ละ viewport

หลังจากที่แบ่งจอภาพออกเป็นหลาย viewports แล้ว จะมี viewport ใช้งานอยู่ viewport เดียว ซึ่งถ้ามีการสั่งให้ Zoom หรือ grid หรือ อื่นๆ จะมีผลเฉพาะ viewport ใช้งานนี้เท่านั้น เรียกว่า current viewport การเปลี่ยนให้ viewport อื่นๆเป็น viewport ปัจจุบันสามารถทำได้โดยเลื่อนอุปกรณ์ชี้ให้เคอร์เซอร์ออกไปที่ viewport อื่นจากใยกากบาทจะเปลี่ยนเป็นลูกศรแทนจากนั้นกดปุ่มเลือก(pick) ที่อุปกรณ์ชี้ จากลูกศรจะเปลี่ยนเป็นใยกากบาทแสดงว่าเราได้เปลี่ยนviewport ปัจจุบันแล้วใน 1 จอภาพ จะมี viewport ปัจจุบันได้เพียง 1 viewport เท่านั้น

ทางเลือก

Save	-เก็บสถานะของการแบ่งจอภาพขณะนั้นไว้ เพื่อที่จะเรียกใช้ในภายหลังได้
Restore	-นำสถานะของการแบ่งจอภาพที่ได้ save ไว้มาใช้งาน
Delete	-ลบสถานะของการแบ่งจอภาพที่ได้ save ไว้ทิ้งไป
Join	-ทำการเชื่อม Viewports 2 viewports เป็น 1 ส่วน
SIngle	-ปรับให้มีการแสดงผล 1 viewport ตามปกติ
?	-แสดงรายละเอียดของ viewport ต่างๆที่ได้ save ไว้
2	-แบ่ง current viewport เป็น 2 viewports
3	-แบ่ง current viewport เป็น 3 viewports
4	-แบ่ง current viewport เป็น 4 viewports

-ตัวเลือกเพื่อกำหนดลักษณะการแบ่งมีดังนี้

Horizontal	-แบ่งออกตามแนวนอน
Vertical	-แบ่งออกตามแนวดิ่ง
Above	-แบ่งแล้วให้ viewport ช่องใหญ่อยู่ส่วนบน
Below	-แบ่งแล้วให้ viewport ช่องใหญ่อยู่ด้านล่าง
Left	-แบ่งแล้วให้ viewport ช่องใหญ่อยู่ด้านซ้าย
Right	-แบ่งแล้วให้ viewport ช่องใหญ่อยู่ด้านขวา

-คำสั่ง PLAN

ทำหน้าที่ กำหนดให้พลิกระนาบของ UCS ปัจจุบันขึ้นมาระนาบกับจอภาพ (แกน X อยู่ในแนวนอน แกน Y อยู่ในแนวตั้ง และแกน Z พุ่งออกมาจากจอภาพ)

2.1.6 คำสั่งที่ใช้ช่วยในการเขียน

-คำสั่ง SNAP

ทำหน้าที่ กำหนดการ snap หรือกระโดดของเคอร์เซอร์อย่างมีระเบียบ

ทางเลือก <n> -กำหนดค่าของการ snap ทั้งแกน x และแกน y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ON -กำหนดให้มีการ snap ของเคอร์เซอร์ตามค่าที่ระบุ

OFF -ยกเลิกการ snap ของเคอร์เซอร์ที่ระบุ

A<Aspect> -การกำหนดระยะการ snap ในแกน X และ Y ที่ไม่เท่ากัน

R<Rotate> -การหมุนแกนของการ snap ไปในแนวอื่น

S<Style> -กำหนดลักษณะของการ snap มี 2 ลักษณะดังนี้

S<Standard> -แบบปกติ

I<Isometric> -แบบ isometric สำหรับการเขียนภาพ Isometric

-คำสั่ง GRID

ทำหน้าที่ แสดงจุดอ้างอิงบนจอภาพ ด้วยระยะที่กำหนด

ทางเลือก <n> -ระยะของ grid ที่ต้องการ ทั้งแกน x และ y

ON -กำหนดให้มีการแสดง grid

OFF -ยกเลิกการแสดง grid

S<Snap>-กำหนดให้ระยะ grid เปลี่ยนแปลงค่า Snap

A<Aspect>-กำหนดระยะ grid ในแกน x และ y ที่ไม่เท่ากัน

หมายเหตุ ลักษณะของ grid จะเปลี่ยนไปตามลักษณะของ Snap style ด้วย

-คำสั่ง AXIS

ทำหน้าที่ กำหนดให้มีขีดแบ่งระยะที่ขอบของจอภาพ เพื่อช่วยในการมอง

ทางเลือก <n> -กำหนดระยะห่างของขีดแบ่ง

ON -กำหนดให้แสดงขีดแบ่งระยะ

OFF -ยกเลิกการแสดงขีดแบ่งระยะ

S<Snap> -กำหนดให้ขีดแบ่งระยะปรับตามค่า Snap

A<Aspect>-กำหนดขีดแบ่งระยะในแกน X และ แกน Y ที่ไม่เท่ากัน

-คำสั่ง ORTHO

ทำหน้าที่ กำหนดหรือไม่กำหนดการเลือกตำแหน่งแบบตั้งฉากกับจุดที่ผ่านมา โดยปกติเส้นนอนจะอยู่ในแกน 0 องศาและ 90 องศาสำหรับแกนตั้ง ทั้งนี้แนวแกนจะขึ้น อยู่กับการกำหนดค่า snap rotate ด้วย

-คำสั่ง ISOPLANE

ทำหน้าที่ ช่วยในการเขียนรูป Isometric ให้สะดวกขึ้น โดยทำให้ลักษณะการเคลื่อนที่ของเคอร์เซอร์เปลี่ยนแปลงไป เพื่อให้สะดวกในการเขียนในระนาบต่างๆ

ทางเลือก <Enter> -เป็นการเปลี่ยนระนาบที่อยู่ไปยังระนาบถัดไปในลักษณะ Left ...Top...Right... สลับกันไปตามลำดับ

L<Left> -เลือกระนาบทางซ้าย

T<Top> -เลือกระนาบบน

R<Right> -เลือกระนาบขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ ในการใช้งานยังมี control key อีกตัวหนึ่งคือ Ctrl E สำหรับใช้ในการเปลี่ยนระนาบ เหมือนการกด Enter ในคำสั่ง ISOPLANE แต่สะดวกกว่าโดยที่เราสามารถเปลี่ยนระนาบนี้ได้ตลอดเวลาที่ต้องการ ไม่ว่าจะอยู่ในคำสั่งใดๆ

-คำสั่ง OSNAP (Object Snap)

ทำหน้าที่ กำหนดลักษณะการ snap ของเคอร์เซอร์ไปยังจุดต่างๆ ตามเงื่อนไขที่ระบุ ที่จะให้โปรแกรมทำงานในโหมด ที่ระบุทันทีที่มีการกำหนดจุดใดๆ โดยในโหมดของ osnap นี้จะสร้างกรอบสี่เหลี่ยม ขึ้นมาที่เคอร์เซอร์ การกำหนดจุดเพียงแต่เลือก mode ของการ snap และเลื่อนเคอร์เซอร์ไปยังบริเวณที่ต้องการ ไม่จำเป็นต้องกำหนดให้ตรงกับจุดที่ต้องการพอดี เพียงแต่ให้จุดที่ต้องการอยู่ภายในกรอบสี่เหลี่ยมเท่านั้น

ทางเลือก

NEA<Nearest>	-จุดที่ใกล้กับจุดตัดของเคอร์เซอร์ที่สุด
END<ENDpoint>	-จุดปลายของเส้น
MID<MIDpoint>	-จุดกึ่งกลางของเส้น
CEN<CENter>	-จุดศูนย์กลางของวงกลมที่เลือก
NOD<NODe>	-snap ไปยังชิ้นวัตถุที่เป็นจุด
QUA<QUAdrant>	-snap ไปยังจุด 0,90,180,270 องศาบนเส้นรอบวงของวงกลมหรือ ส่วนโค้งที่เลือก
INT<Intersection>	-จุดตัดของเส้น
INS<INSert>	-จุดอ้างอิงของ บล็อก
PER<Perpendicular>	-จุดที่อยู่ในแนวตั้งฉากกับจุดที่ผ่านมา
TAN<TANgent>	-จุดสัมผัสกับส่วนโค้งที่กำหนด
QUI<QUIck>	-ใช้ร่วมกับ mode อื่นๆ โดยจะเลือกเอาจุดแรกที่พบว่าถูกต้องตามเงื่อนไข ซึ่งโดยปกติจะมีการตรวจสอบภายในกรอบเสียก่อน แล้วจึงเลือกจุดที่ตรงเงื่อนไขที่ใกล้จุดตัดของเคอร์เซอร์ที่สุด

-คำสั่ง APERTURE

ทำหน้าที่ กำหนดขนาดของกรอบสี่เหลี่ยมที่จะปรากฏในการเรียก Object snap

ทางเลือก กำหนดขนาดของวงกรอบ ตั้งแต่ 1 ถึง 50 จุดบนจอภาพ

-คำสั่ง DDRMODES

ทำหน้าที่ กำหนดตัวแปรเกี่ยวกับดรออิงเอด (Drawing Aids) ด้วยแบบฟอร์มบนจอภาพ

-คำสั่ง DDEMODES

ทำหน้าที่ กำหนดตัวแปรที่เกี่ยวกับการสร้างวัตถุด้วยแบบฟอร์มบนจอภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 คำสั่งเกี่ยวกับบล็อก และ แอตทิบิว (Attribute)

-คำสั่ง BLOCK

ทำหน้าที่ เก็บดรออิง ที่ระบุในลักษณะของเทปเบล็ต (template) เพื่อความสะดวกในการเรียกใช้ในภายหลัง ไม่ว่าวัตถุที่เลือกจะมีกี่ชิ้นส่วนก็ตาม เมื่อถูกสร้างให้เป็นบล็อกแล้วจะถือเป็นวัตถุชิ้นเดียวกัน และโดยปกติจะแก้ไขไม่ได้ ยกเว้นในขณะที่เรียก บล็อก นั้นกลับมา จะใส่เครื่องหมายดอกจัน นำหน้าชื่อ บล็อก ที่ระบุ หรือการใช้คำสั่ง EXPLODE

ทางเลือก ? -ดูรายชื่อของ บล็อก ที่มีอยู่แล้วใน ดรออิง นั้นๆ

<name> -ระบุชื่อ บล็อก ที่จะตั้งชื่อให้วัตถุที่เลือก

-คำสั่ง INSERT

ทำหน้าที่ การเรียกบล็อก ที่เคยสร้างไว้มาใช้

ทางเลือก ? -ดูรายชื่อของ บล็อก ที่มีอยู่แล้วใน ดรออิง นั้นๆ

<name> -ระบุชื่อของ บล็อก ที่มีอยู่มาใช้

<name>= -กำหนดให้มีการเรียกข้อมูลของ บล็อก นั้นๆเข้ามาใหม่ ในกรณีที่มีการแก้ไข บล็อก นั้นๆ บล็อก ที่มีอยู่แล้วจะยังไม่แก้ไขตามจนกว่าจะใช้วิธีนี้

-คำสั่ง WBLOCK

ทำหน้าที่ เหมือนคำสั่ง BLOCK แต่จะเก็บข้อมูลของ บล็อก นั้นๆลงดิสค์ด้วยโดยถามชื่อไฟล์ที่จะเก็บข้อมูลเหล่านี้ไว้

ทางเลือก หลังจากกำหนดชื่อไฟล์แล้ว จะถามถึงชื่อ Block โดยมีทางเลือกดังนี้

<name> -นำเอาข้อมูลของ บล็อก ที่สร้างไว้ ไปเก็บไว้ในไฟล์ที่ระบุ

= -นำเอาข้อมูลของ บล็อก ที่มีชื่อเหมือนชื่อไฟล์ไปเก็บไว้

* -เป็นการเก็บข้อมูลของ ดรออิง ทั้งหมด เหมือนคำสั่ง SAVE แต่ยกเว้นข้อมูลของ บล็อก ที่ไม่มีการใช้งานใน ดรออิง

<Enter> -เพื่อเลือกวัตถุที่ต้องการเก็บเป็น บล็อก และเก็บข้อมูลเหล่านั้นลงดิสค์ในชื่อไฟล์ที่ระบุ

-คำสั่ง MINSERT

ทำหน้าที่ เหมือนคำสั่ง INSERT รวมกับคำสั่ง ARRAY โดยที่ในการ insert เข้ามาจะให้ผู้ใช้ระบุถึงจำนวนแถวและคอลัมน์ที่ต้องการด้วย

-คำสั่ง ATTDEF (Attribute Define)

ทำหน้าที่ สร้างรูปแบบของข้อมูลที่เป็น text เพื่อประกอบกับวัตถุที่จะสร้างเป็น บล็อก และเพื่อความสะดวกในการรวบรวม แยกแยะ จัดระเบียบข้อมูลของ บล็อก ที่มีการใส่ลงไปใน ดรออิงได้สะดวกขึ้น

ทางเลือก ลักษณะของข้อมูลมีทางเลือกดังนี้

I<Invisible> -กำหนดให้แสดงข้อมูล หรือไม่แสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C<Constant> -กำหนดข้อมูลเป็นค่าคงที่ตายตัว

V<Verify> -กำหนดให้มีการถามย้ำถึงข้อมูลที่ใส่เข้าไป

P<Preset> -กำหนดข้อมูลไว้ล่วงหน้าโดยไม่มีการถามในขณะ Insert แต่สามารถแก้ไขได้ภายหลัง

-คำสั่ง ATTDISP (Attribute Display)

ทำหน้าที่ กำหนดสถานะการแสดงข้อมูลประเภท แอตทิบิว

ทางเลือก N<Normal> -ตามที่กำหนดในขณะ define (Invisible or not)

ON -กำหนดให้แสดงข้อมูลออกมา

OFF -กำหนดให้ไม่ต้องแสดงข้อมูลออกมา

-คำสั่ง ATTEDIT (Attribute Edit)

ทำหน้าที่ แก้ไขข้อมูลต่างๆของ แอตทิบิว ที่ระบุ

ทางเลือก วิธีการแก้ไขข้อมูลทำได้ดังนี้

-แก้ข้อมูลเฉพาะ แอตทิบิว ที่เลือก ทีละตัว

-แก้ไขข้อมูลของ แอตทิบิว ที่ระบุ พร้อมกันทุกตัวที่มีอยู่ใน ดรออิง

การแก้ไขข้อมูลมีทางเลือกดังนี้

V<Value> -แก้ไขตัวข้อมูลที่บรรจุอยู่

P<Position> -แก้ไขตำแหน่งที่แสดง แอตทิบิว นั้นๆ

H<Height> -แก้ไขความสูงของตัวอักษร

A<Angle> -แก้ไขมุมของการแสดงตัวอักษร

L<Layer> -แก้ไข Layer ที่บรรจุ แอตทิบิว นั้นๆ

C<Color> -แก้ไขสีของตัวอักษร

N<Next> -เลื่อนไปยังข้อมูลตัวต่อไปของ แอตทิบิว ที่เลือกมาแก้ไข

-คำสั่ง ATTEXT (Attribute Text)

ทำหน้าที่ สกัดเฉพาะข้อมูลของ แอตทิบิว ไปเก็บไว้ในไฟล์ที่ระบุ

ทางเลือก รูปแบบของข้อมูลที่สกัดออกไปได้มีดังนี้

C<CDF> -ข้อมูลแบบคั่นด้วยเครื่องหมายคอมม่า (comma)

S<SDF> -ข้อมูลที่กำหนดความยาวของข้อมูลแต่ละตัวไว้อย่างแน่นอน ส่วนมากโปรแกรมด้านฐานข้อมูล จะอ่านรูปแบบนี้ได้

D<DXF> -เป็นรูปแบบของโปรแกรมเอง เรียกว่า Drawing interchange file format

E<Entity> -กำหนด แอตทิบิว ที่ต้องการสกัดข้อมูลไป

Template File-เป็นไฟล์ที่เก็บรูปแบบ สำหรับการสกัดข้อมูล SDF,CDF เพื่อที่โปรแกรมจะได้ใช้เป็นตัวอย่างในการสกัดข้อมูลออกไปในรูปแบบที่ต้องการ

Extrace File -เป็นไฟล์ที่เก็บข้อมูลที่ถูกสกัดออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-คำสั่ง DDATTE

ทำหน้าที่ สำหรับการแก้ไขข้อมูล แอตทิบิว ของบล็อกที่เลือก ผ่านทางไดอะลอกบล็อกที่จะปรากฏขึ้นมาบนจอภาพแทนการแก้ไขข้อมูลที่คอมมานด์ไลน์(command line)

2.1.8 คำสั่งเกี่ยวกับการให้เส้นบอกขนาดและการเขียนลวดลาย

-คำสั่ง DIM

ทำหน้าที่ เข้าสู่โหมดของการให้เส้นวัดระยะแบบต่างๆ

ทางเลือก การวัดระยะทางตรงมีทางเลือกดังนี้

HOR<Horizontal> -ให้เส้นวัดระยะแนวนอน

VER<Vertical> -ให้เส้นวัดระยะแนวตั้ง

ALI<Aligned> -ให้เส้นวัดระยะตามแนวของวัตถุที่วัด

ROT<Rotated> -ให้เส้นวัดระยะตามแนวที่กำหนด

BAS<Baseline> -การให้เส้นวัดระยะต่อเนื่องกับระยะที่ผ่านมาโดยวัดระยะจากจุดเริ่มต้นของเส้นววัดระยะที่ผ่านมา

CON<Continue> -การให้เส้นวัดระยะต่อเนื่องกับระยะที่ผ่านมา โดยวัดระยะต่อจากจุดปลายของเส้นวัดระยะที่ผ่านมา

การวัดระยะกับส่วนโค้งมีทางเลือกดังนี้

ANG<Angular> -วัดมุมระหว่างเส้น 2 เส้นที่ระบุ

DIA<Diameter> -วัดเส้นผ่านศูนย์กลางของวงกลม หรือส่วนโค้งที่ระบุ

RAD<Radius> -วัดรัศมีของวงกลม หรือส่วนโค้งที่ระบุ

คำสั่งอำนวยความสะดวกต่างๆ มีดังนี้

CEN<Center> -เขียนเครื่องหมายกากบาทที่จุดศูนย์กลางของวงกลมหรือส่วนโค้ง

LEA<Leader> -เขียนเส้นตรง เหมือนคำสั่ง LINE แต่ที่จุดเริ่มต้นของเส้นจะมีลูกศรติดอยู่ด้วย

EXIT -ออกจากโหมด ของการให้ dimension

REDRAW -เหมือนคำสั่ง REDRAW ปกติ

STATUS -แสดงค่าปัจจุบันของตัวแปรต่างๆ ที่ใช้อยู่ในการให้ dimension

STYLE -เปลี่ยนสไตล์(style) ของอักษรที่ใช้อยู่ในการให้ dimension

UNDO -ยกเลิกการให้ระยะครั้งที่ผ่านมา ย้อนหลังไปทีละคำสั่ง

-คำสั่ง HATCH

ทำหน้าที่ เขียนลวดลายที่ระบุลงบนพื้นที่ที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางเลือก ? -เลือกรายการชื่อลวดลายมาตราฐานที่มีอยู่ใน PATTREN FILE
<name> -ระบุเฉพาะชื่อลวดลายที่ต้องการใช้(จะแลเงาด้วยวิธีการแบบปกติ)
<name>,style-ระบุชื่อของลวดลายและวิธีการเขียน ซึ่งมีรูปแบบดังนี้
N<Normal> -แบบปกติ
O<Outermost>-เขียนลวดลายเฉพาะขอบเขตนอกสุดของวัตถุที่เลือก
I<Ignore> -เขียนลวดลายไปทั้งชิ้น โดยไม่สนใจลวดลายภายใน
U<User style>-กำหนดรูปแบบการเขียนลวดลายเอง(ด้วยเส้นตรงที่เรา
สามารถกำหนดระยะห่างของเส้น และมุม รวมทั้งมีเส้น
ไขว้ด้วยหรือไม่)

2.1.9 คำสั่งเกี่ยวกับการควบคุมโปรแกรม

-คำสั่ง SCRIPT

ทำหน้าที่ สั่งให้โปรแกรมทำงานตามคำสั่งต่างๆที่บรรจุอยู่ในสคลิปไฟล์ (script file) โดยอัตโนมัติ

ทางเลือก ระบุชื่อของสคลิปไฟล์ที่ต้องการ

-คำสั่ง DELAY

ทำหน้าที่ หน่วงเวลาการทำงานของโปรแกรม โดยมากมักใช้กับสคลิปไฟล์ เพื่อให้หน่วงเวลาการแสดงภาพต่างๆตามเวลาที่ระบุ

ทางเลือก ระบุรายการหน่วงเวลาเป็นตัวเลข มีค่าประมาณ 1000 ต่อ 1 วินาที

-คำสั่ง RESUME

ทำหน้าที่ สั่งให้มีการทำงานในสคลิปไฟล์ต่อเนื่องจากจุดที่หยุด (โดยอัตโนมัต หากมีข้อผิดพลาดใดๆ โปรแกรมจะหยุดและกลับไปที่คอมมานด์พร็อมต์)

-คำสั่ง RSCRIPT

ทำหน้าที่ กำหนดให้มีการทำงานในสคลิปไฟล์ ที่เพิ่งเรียกใช้อีกครั้ง หากใส่คำสั่งนี้ไว้ในตอนท้ายของสคลิปไฟล์ จะทำให้เกิดการแสดงแบบต่อเนื่องไม่หยุด(จนกว่าจะมีการกด Ctrl C เท่านั้น)

-คำสั่ง 'GRAPHSCR

ทำหน้าที่ สั่งให้เปลี่ยนการแสดงผลบนจอภาพเป็นจอภาพกราฟิก (graphic screen)

-คำสั่ง 'TEXTSCR

ทำหน้าที่ สั่งให้เปลี่ยนการแสดงผลบนจอภาพเป็นจอภาพตัวอักษณ (text screen)

-คำสั่ง MSLIDE

ทำหน้าที่ การเก็บภาพที่แสดงอยู่บนจอภาพเป็นไฟล์

ทางเลือก ให้ชื่อสไลด์ไฟล์ (slide file :xxx.SLD) ที่จะเก็บ

-คำสั่ง VSLID

ทำหน้าที่ เรียกเอาภาพที่เก็บไว้ด้วยคำสั่ง MSLID มาแสดงบนจอภาพ

ทางเลือก ให้ชื่อสไลด์ไฟล์ (xxx.SLD) ที่จะเรียกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.10 คำสั่งเกี่ยวกับเครื่องมือชี้

-คำสั่ง SKETCH

ทำหน้าที่ เขียนเส้นสายแบบฟรีแฮนด์ (free hand) ตามการเคลื่อนไหวของ pointing device ที่มีอยู่

ทางเลือก ก่อนอื่นต้องระบุความยาวของเส้นแต่ละเส้นที่จะเกิดขึ้น ในขณะที่เลื่อนเคอร์เซอร์ว่าจะมีความยาวเท่าใด จึงจะบันทึกเส้นนั้นๆลงไป เช่น 0.2 unit หมายความว่าเมื่อเลื่อนเคอร์เซอร์ไปเป็นระยะทาง 0.2 unit ในทิศทางใดก็ตาม โปรแกรมจะเขียนเส้นตรงลงไป 1 เส้น และเมื่อเลื่อนก 0.2 unit ก็จะบันทึกลงไปอีก 1 เส้น(ต่อเนื่องกัน) เป็นเช่นนี้ไปเรื่อยๆ

-คำสั่ง TABLET

ทำหน้าที่ ควบคุมการทำงานของเทปเบล็ต (tablet) หรือ ดิจิไตเซอร์(digitizer)

ทางเลือก CAL<Calibrate> -ปรับการอ่านระยะของเทปเบล็ต ให้ถูกต้องกับระยะจริง

ON -กำหนดให้มีการใช้เทปเบล็ตแบบมีมาตราส่วนตามการปรับ(CAL)

OFF -กำหนดให้เทปเบล็ต ทำหน้าที่เป็นตัวชี้เฉยๆ

CFG<Configure>-การกำหนดพื้นที่และจำนวนของบริเวณที่จะใช้เป็นบริเวณรายการคำสั่งบนเทปเบล็ต

2.1.11 คำสั่งเกี่ยวกับรูป 3 มิติ

-คำสั่ง ELEV

ทำหน้าที่ กำหนดระดับของการเขียน

ทางเลือก <elevation> -ระดับของการเขียน

<thickness> -ความหนา หรือความสูงของวัตถุที่เขียน (ค่าในแกน Z)

-การมองภาพ 3 มิติ

ภาพ 3 มิติที่เราได้สร้างขึ้นนั้นสามารถมองในมุมต่างๆได้ตามความต้องการ ซึ่งการมองภาพ 3 มิติแบ่งออกเป็น 2 ลักษณะคือ

1. parallel Projection (มองขนานแบบ isometric)
2. Perspective Projection

ใน ออโตแคด ต่ำกว่า รีลิส 10 ลงมา จะมองภาพ 3 มิติได้แบบเดียวเท่านั้น คือ แบบขนาน (parallel projection) โดยใช้คำสั่ง VPOINT (VIEW POINT) แต่ถ้า รีลิส 10 ขึ้นไป จะมองภาพแบบ perspective ได้โดยใช้คำสั่ง DVIEW (DYNAMIC VIEW)

การใช้ VPOINT (view point) VPOINT เป็นคำสั่งที่ทำให้เกิดการมองภาพแบบ3 มิติแบบขนานที่เก่าแก่คำสั่งหนึ่ง เพราะมีตั้งแต่รุ่นแรกๆในขณะที่กำหนดมุมมอง 3 มิตินี้ภาพทีบนจอจะไม่เคลื่อนไหว จนกว่าจะกำหนดมุมมองเรียบร้อยแล้ว จึงจะปรากฏภาพ 3 มิติตามมุมมองจุดใหม่บนจอ ถ้าต้องการให้ภาพมีการเคลื่อนไหวไปด้วย ขณะที่กำลังเลือกมุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-คำสั่ง VPOINT

ทำหน้าที่ กำหนดจุดมุมมองที่จะมองวัตถุในภาพ 3 มิติ

ทางเลือก 1 -กำหนดค่าโคออร์ดิเนตทั้ง 3 (x,y,z) ลงไปเลย

2 -หากกด Enter เฉยๆ โปรแกรมจะให้เลือกมุมมองโดยการหมุนแกน3 มิติด้วยเคอร์เซอร์ไปยังมุมที่ต้องการ

R<Rotate> -การหมุนดูภาพ 3 มิติ โดยการให้ค่ามุม 2 ตัว คือ

1. มุมหมุนในแกน z (เหมือนคำสั่ง ROTATE)
2. มุมที่ทำกับระนาบ x-y

<point> -การชี้จุดมองจากระนาบ 2 มิติ ที่เขียนอยู่โดยตรง ซึ่งจะเป็นการมองที่ระดับที่เขียนเท่ากับ 0 (มองเห็นเฉพาะค่าในแกน z) โดยจุดที่ชี้ค- +@ ทิศ ทางที่มองเข้าไปหาจุดกำเนิดของชิ้นวัตถุ

-การใช้ Dview (dynamic view)

นอกจากการแสดงภาพ 3 มิติในลักษณะ parallel injection ในรีลีส 10 ยังสามารถแสดงภาพ 3 มิติในลักษณะ perspective ได้โดยใช้คำสั่ง Dview ช่วยให้ภาพที่ปรากฏมีความรู้สึกที่ดีขึ้น

การใช้ DVIEW จะคล้ายกับการมองภาพผ่านช่องมองภาพของกล้องถ่ายภาพแบบปรับเลนซ์ซูมได้โดยที่จุดมองหรือสายตาอยู่ที่ camaraสามารถปรับขนาดของเลนส์ที่ใช้ให้มีมุมกว้างมุมแคบได้

ขีดจำกัดของ DVIEW คือ เราจะต้องกำหนดว่าส่วนของภาพที่จะให้แสดงใน DVIEW มีอะไรบ้าง หากภาพมีความสลับซับซ้อนมาก การแสดงภาพในขณะที่เราเคลื่อนไหวไปมองมุมต่างๆจะช้าลง ต่างกับ VPOINT จะแสดงภาพใน ดรออิง ทั้งหมดออกมาในลักษณะ 3 มิติและขณะอยู่ใน DVIEW ไม่สามารถ แก้ไขหรือเพิ่มเติมส่วนของภาพได้

ทางเลือก

CAmera -เป็นการกำหนดตำแหน่งหรือมุมมองหรือตำแหน่งของกล้อง โดยที่เป้าหมายอยู่กับที่

TArget -เป็นการกำหนดเป้าหมายในการมอง โดยจุดมองหรือตำแหน่งของกล้องอยู่กับที่คล้ายกับการแพน (pan) กล้องนั่นเอง

Distance -กำหนดระยะห่างระหว่างเป้าหมาย และตำแหน่งกล้อง ในขณะที่เข้าสู่คำสั่ง DVIEW การแสดงภาพยังเป็นลักษณะ parallel projection อยู่ ต่อ เมื่อมีการกำหนดระยะทาง (distance)แล้วการแสดงภาพจึงจะเป็นลักษณะ perspective

POints -กำหนดตำแหน่งของกล้อง และเป้าหมายในการมอง

PAn -เลื่อนภาพที่แสดงอยู่ใน DVIEW ให้ไปอยู่ตำแหน่งต่างๆของจอคล้าย pan โหมดปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Zoom -กำหนดขนาดของเลนส์ที่ใช้ในการมอง เช่นเดียวกับการถ่ายภาพ โดยที่ค่า ที่กำหนดขึ้นโดยอัตโนมัติใน ออโตแคด ตั้งไว้ให้ใช้ก่อนคือ เลนส์ขนาด 50 มิลลิเมตร หากการแสดงภาพยังเป็นแบบ parallel projection การ zoom ก็จะเป็นการย่อ-ขยายภาพนั่นเอง

TWist -เป็นการหมุนภาพ โดยแนวแกนการหมุนอยู่ที่แนวการมองจากกล้องไปยังเป้าหมายคล้ายกับการเอียงกล้องขณะถ่ายภาพ

CLip -กำหนดให้มีการขลิบภาพไม่ให้แสดงออกมาบนจอ โดยสามารถขลิบจากด้านหน้า หรือ ขลิบจากด้านหลังได้

Hide -ทำการลบเส้นที่ถูกบังให้การแสดงภาพแบบ 3 มิติออกไป

Off -ยกเลิกการแสดงภาพแบบ perspective กลับไปที่ parallel เช่นเดิม ตัวเลือกนี้ตรงข้ามกับ distance

Undo -ยกเลิกการทำงานที่ได้ทำมาก่อนหน้านี้ 1 ครั้ง

eXit -ออกจากคำสั่ง DVIEW

-คำสั่ง HIDE

ทำหน้าที่ ลบเส้นด้านหลังที่ถูกบัง (hidden line) ในการมองภาพ 3 มิติ

-คำสั่ง 3DLINE

ทำหน้าที่ เขียนเส้นตรงเหมือนคำสั่ง LINE แต่สามารถกำหนดค่าโคออร์ดิเนตได้ทั้ง 3 แกน คือ x,y,z (สำหรับคำสั่ง LINE ค่า z จะขึ้นอยู่กับค่าของระดับที่เขียนหรือ elevation ที่ระบุ)

ทางเลือก เราสามารถที่จะใส่ค่าของทั้ง 3 แกนลงไปได้เลยด้วยแป้นพิมพ์ แต่สำหรับการใช้การเลื่อนเคอร์เซอร์แล้ว คำสั่งนี้จะรับค่า elevation ปัจจุบัน ไปเป็นค่า z โดยอัตโนมัติ ยกเว้นจะมีการกำหนดให้โปรแกรมรู้ว่าจะให้ค่าเฉพาะใน แกนใดบ้าง เช่น

.xy -จะกำหนดเฉพาะค่าในแกน x และ y ก่อน และเมื่อกำหนดแล้วโปรแกรมจะขึ้นข้อความว่า "need z" หมายความว่าต้องการให้กำหนดค่า z ลงไปอีกการกำหนดจุดนั้นๆจึงจะสมบูรณ์

.xz -จะกำหนดเฉพาะค่าในแกน x และ z ก่อน หลังจากนั้นจึงจะให้ค่าในแกน y ที่เหลือต่อไป

-คำสั่ง 3DFACE

ทำหน้าที่ เขียนระนาบทึบเหมือนคำสั่ง SOLID แต่สามารถกำหนดโคออร์ดิเนตได้ทั้ง 3 แกนแบบเดียวกับคำสั่ง 3DLINE

ทางเลือก มีวิธีการกำหนดค่าโคออร์ดิเนตเหมือนกับการใช้คำสั่ง 3DLINE

-คำสั่ง 3DMESH

ทำหน้าที่ ใช้สำหรับการสร้างพื้นผิว 3 มิติ โดยแบ่งเป็นตารางที่ขนาดที่กำหนด(MxN) และ โคออร์ดิเนต ของจุดตัดของตารางที่เหล่านั้น(ความสูง หรือ ค่า z ด้วย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-คำสั่ง 3DPOLY

ทำหน้าที่ ใช้สำหรับการเขียน polyline แบบ 3 มิติ(กำหนดค่า z ในแต่ละ vertex ได้)

ทางเลือก C<Close> -ต่อเชื่อมเส้นจากจุดสุดท้ายไปยังจุดแรกที่เขียน

U<Undo> -ยกเลิกเส้นสุดท้ายที่กำหนดลงไป

RETURN -ออกจากคำสั่ง 3 DPOLY

-คำสั่ง EDGESURF

ทำหน้าที่ ใช้สำหรับการสร้างพื้นผิว 3 มิติ โดยกำหนดเส้นขอบจำนวน 4 เส้น(ที่ต่อเนื่องกัน)

-คำสั่ง REVSURF

ทำหน้าที่ ใช้สำหรับการสร้างพื้นผิวรอบแกนที่กำหนด โดยการกำหนดรูปร่างหรือ profile ของพื้นผิวนั้นๆ และจำนวนองศาที่จะกวาดเป็นพื้นผิวนั้นๆ

-คำสั่ง RULESURF

ทำหน้าที่ ใช้สำหรับการสร้างพื้นผิวระหว่างเส้นสาย 2 เส้นที่เลือกเป็น profile

-คำสั่ง TABSURF

ทำหน้าที่ ใช้สำหรับการสร้างพื้นผิวตาม profile ที่เลือกไปในทิศทางและความยาวที่กำหนด

2.1.12 คำสั่งเกี่ยวกับการแลกเปลี่ยนไฟล์

-คำสั่ง DXFOUT

ทำหน้าที่ การเก็บข้อมูลของชิ้นงานในรหัส ASCII ด้วยรูปแบบมาตราฐาน เพื่อความสะดวกในการนำข้อมูลของโปรแกรมไปใช้กับโปรแกรมอื่น

-คำสั่ง DXFIN

ทำหน้าที่ การนำข้อมูลรหัส ASCII ที่ถูกเก็บในรูปแบบมาตราฐานของ data interchange file เข้ามาในโปรแกรม

-คำสั่ง DXBIN

ทำหน้าที่ การนำข้อมูลในรูปแบบของรหัส binary ซึ่งโปรแกรมบางชนิดจะเก็บข้อมูลในรูปแบบดังกล่าว และเราสามารถดึงข้อมูลเหล่านั้นมาใช้ได้ด้วยคำสั่งนี้

-คำสั่ง IGESOUT

ทำหน้าที่ การเก็บข้อมูลของ ดรออิง ในรูปแบบของ IGES(Initial graphics exchange standrad) ซึ่งเป็นรูปแบบมาตราฐานแบบหนึ่งของโปรแกรมประเภทกราฟิกหลายโปรแกรม

-คำสั่ง IGESIN

ทำหน้าที่ การนำเอาข้อมูลที่เก็บในรูปแบบของ IGES มาใช้ในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2การป้อนคำสั่งให้กับ ออโตแคด สามารถทำได้ 3 วิธี ดังนี้ คือ

2.2.1 จากแป้นพิมพ์ ในส่วนของ command line จะมีข้อความว่า "command:" ซึ่งหมายความว่าโปรแกรมพร้อมที่จะรับคำสั่งใดๆ การสั่งให้ทำงานใดๆก็ตามสามารถทำได้โดยการพิมพ์คำสั่งนั้นๆลงไป ตามด้วยการกด Enter หรือ Space bar หากที่ command line ในบรรทัดสุดท้ายยังเป็นข้อความอื่นนอกเหนือจากคำว่า "command:" แสดงว่าโปรแกรมยังทำงานค้างอยู่และไม่พร้อมที่จะรับคำสั่งอื่นๆ หากต้องการยกเลิกสถานะที่ค้างอยู่ให้กด Ctrl C หลังจากนั้น โปรแกรมก็พร้อมที่จะรับคำสั่งต่อไปได้

2.2.2 จากสกรีนเมนู (Screen menu) เราสามารถใช้คำสั่งต่างๆได้โดยการเลือกจากสกรีนเมนูทางขวามือของจอภาพ โดยการกดปุ่ม Ins สำหรับการใช้แป้นพิมพ์เป็นตัวชี้ . หากใช้ตัวชี้อื่นๆ เราจะใช้สกรีนเมนูได้โดยอัตโนมัติทันทีที่เลื่อนเคอร์เซอร์ไปพ้น Graphic area โปรแกรมจะแสดงแถบสีที่ เรียกว่า inverse bar ที่บริเวณ สกรีนเมนูและสามารถเลื่อนไปยังคำสั่งที่ต้องการได้โดยใช้ Direction key ขึ้นหรือลง เมื่อแถบ inverse ปรากฏบนคำสั่งที่ต้องการก็ให้กด Enter จะเป็นการเลือกคำสั่งนั้นมาใช้

2.2.3ป็อปอัพเมนู(Pop Up Menu) เมื่อเลื่อนเคอร์เซอร์มาที่ส่วนบนของจอภาพStatus line จะเปลี่ยนเป็นป็อปอัพเมนู โดยคำสั่งในหมวดต่างๆจะแสดงอยู่ในบรรทัดดังกล่าวเราสามารถเลื่อนเคอร์เซอร์ไปยังหมวดคำสั่งที่ต้องการแล้วกดปุ่มบนอุปกรณ์ชี้ คำสั่งในหมวดนั้นๆจะปรากฏขึ้นมาบนจอภาพทันที แล้วเราก็เลือกคำสั่งที่ต้องการโดยเลื่อนเคอร์เซอร์ลงมายังคำสั่งที่ต้องการ

2.2.4 การซ้ำคำสั่งเดิม

การใช้คำสั่งซ้ำกับคำสั่งที่ผ่านมา สามารถย่นย่อระยะเวลาลงได้บ้าง โดยการกด Enter หรือ Space bar ลงไป(หลังคำว่า "command:") โปรแกรมจะซ้ำคำสั่งเดิมที่เพิ่งใช้มาโดยอัตโนมัติ สำหรับในบางคำสั่ง การใช้วิธีดังกล่าวนี้ โปรแกรมอาจจะตัดคำถามบางข้อในตอนต้นของคำสั่งนั้นๆ ออกไปเพื่อลดขั้นตอนและเพิ่มความเร็วในการทำงาน สำหรับในกรณีที่ต้องการใส่ข้อมูลลงไปใหม่ในคำถามต่างๆที่ละเอาไว้ จะต้องใช้วิธีการพิมพ์ลงไปใหม่จากแป้นพิมพ์หรือเลือกจาก Screen Menu เท่านั้น

2.3เครื่องมือที่ช่วยในการเขียน

ในการเลื่อนเคอร์เซอร์ไปยังตำแหน่งที่เราต้องการนั้น บางครั้งอาจจะมองตำแหน่งเหล่านั้นยากและค่อนข้างลำบากในการกำหนดจุดที่แน่นอน สังเกตได้จากตัวเลขที่มุมบนขวาของจอภาพ(ค่าของโคออร์ดิเนตที่จุดตัดของเคอร์เซอร์จะเปลี่ยนแปลงไปในขณะที่เลื่อนเคอร์เซอร์หากตัวเลขไม่เปลี่ยนแปลงให้กด F6)ในโปรแกรมนี้จึงคำสั่งที่เป็นเครื่องมือช่วยในการเขียน (drawing aided) เข้ามาช่วยดังนี้

2.3.1 SNAP เป็นการตั้งระยะการกระโดด(Step size) ของเคอร์เซอร์ โดยปกติการตั้งsnap ในครั้งแรกจะให้ค่าเป็น 1 หากมีการตั้ง snap ที่มุมบนซ้ายของจอ จะมีคำว่า snap หรือ S อยู่ การตั้ง snap ไว้จะทำให้การกำหนดจุดเที่ยงตรงขึ้นและสะดวกขึ้น ในการใช้งานจริงจึงควรตั้ง snap เสมอ แล้วใช้วิธีเปลี่ยนค่า snap ไปเรื่อยๆตามลักษณะที่จะเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 ORTH เป็นการกำหนดให้มีการเขียนเส้นเป็นแบบตั้งฉากกันเท่านั้น โดยปกติจะเป็นเส้นนอน (0 องศา) และเส้นตั้ง(90 องศา) ยกเว้นที่มีการตั้ง snap rotate เส้นนอนก็จะเอียงไปตามค่ามุม rotate ส่วนอีกเส้นจะตั้งฉากกับเส้นนี้ mode นี้จะช่วยให้สะดวกขึ้นในการเขียนรูปเหลี่ยมที่มีเส้นตั้งฉากกัน

2.3.3 GRID เป็นคำสั่งที่ทำให้เกิดจุดบนจอเป็นระยะตามที่จะกำหนด หากกำหนดให้มีความละเอียดของ grid มากเกินไป จะมีข้อความว่า "grid too dense to display " หมายความว่าค่า grid ที่กำหนดมีค่าน้อยเกินไป (ถี่เกินไปที่จะแสดงผลออกมาได้) คำสั่งนี้เป็นคำสั่งที่ช่วยในเรื่องของการแสดงผลบนจอภาพเพื่อช่วยในการมองเท่านั้น และจุดเหล่านี้จะไม่ถูกพล็อตออกมาด้วย ถ้ากำหนดให้ grid = 0 จะเป็นการกำหนดให้ระยะ grid ปรับไปตามค่าของ snap

2.3.4 OSNAP (OBJECT SNAP) เป็นคำสั่งที่ช่วยให้การเลือก หรือการกำหนดตำแหน่งของเคอร์เซอร์สะดวกขึ้น โดยเฉพาะในกรณีที่ snap mode เป็น off หรือจุดที่ต้องการไม่ลงตัวกับตำแหน่งการ snap โดยแทนที่การเลือกตำแหน่งจะเป็นเพียงจุดตัดของเคอร์เซอร์ก็จะกลายเป็นกรอบสี่เหลี่ยมที่เรียกว่า aperture แทน การเลือกเพียงแต่ให้จุดที่ต้องการอยู่ภายในกรอบก็เพียงพอ ทั้งนี้จะต้องบอกลักษณะของการ snap ไปยังจุดนั้นๆ ให้กับโปรแกรมด้วย

วิธีการเข้าสู่โหมด ของ osnap ทำได้โดยการพิมพ์โหมดต่างๆ ที่ต้องการลงไปในขณะที่โปรแกรมถามถึงค่าโคออร์ดิเนต (ไม่ว่าคำสั่งใดก็ตาม) เช่น end หมายถึงจุดปลายของเส้น mid หมายถึงจุดกึ่งกลางของเส้น เป็นต้น หลังจากนั้นที่จุดตัดของเคอร์เซอร์จะมีกรอบสี่เหลี่ยมติดอยู่แล้วจึงเลื่อนเคอร์เซอร์ไปยังบริเวณที่ต้องการต่อไป

ด้วยวิธีการดังกล่าวจะเพิ่มความเร็วในการทำงานเป็นอย่างมาก สิ่งที่เราต้องการทำงานเพิ่มเติมจากเดิมก็เพียงแต่พิจารณาว่า จุดที่เราต้องการเป็นจุดหมายลักษณะใด (เช่น ปลายเส้น กึ่งกลางเส้นจุดตัด เป็นต้น) แล้วกำหนดโหมดของการ snap ลงไปเท่านั้น จะทำให้การกำหนดจุดรวดเร็วขึ้นและลดความคลาดเคลื่อนลง เพราะในการมองผ่านจอภาพการกำหนดบางครั้งมองดูว่าตรงแล้วแต่ที่จริงยังไม่ตรงก็มี ซึ่งถ้า Zoom เข้าไปใกล้ๆ ก็จะเห็นได้ชัดเจน

เราสามารถกำหนดลักษณะของการ snap ไว้ทำงานโดยอัตโนมัติได้เลย เมื่อมีการถามถึงโคออร์ดิเนตโดยการใช้คำสั่ง OSNAP สามารถมีได้หลายลักษณะพร้อมกัน โดยการป้อนโหมดต่างๆ เรียงเข้าไปและคั่นด้วยเครื่องหมาย , การทำงานจะทำงานในเงื่อนไขของโหมด แรกก่อนเป็นสำคัญ หากไม่พบลักษณะของโหมดแรกจึงจะพิจารณาในโหมดถัดไปเรื่อยๆจนครบ

2.3.5 DDRMODES,DDEMODES,DDLMODES เป็นคำสั่งสำหรับการตั้งค่าตัวแปรในคำสั่งต่างๆ บางคำสั่ง โดยที่เมื่อเรียกคำสั่งเหล่านี้แล้วจอภาพจะปรากฏลักษณะของแบบฟอร์มให้เรากรอกข้อมูล โดยการเลื่อนเคอร์เซอร์ที่กลายเป็นลูกศรไปยังข้อที่ต้องการ หากเราเลื่อนเคอร์เซอร์ไปตรงกับหัวข้อใดแล้ว โปรแกรมจะแสดงถึงจุดที่ทำงานในข้อเหล่านั้นเป็นลักษณะของ inverse video (แถบขาวตัวดำ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับช่องที่เป็นค่าตัวเลขหรือข้อความเมื่อเราเลื่อนเคอร์เซอร์ไปที่ช่องดังกล่าวเราก็สามารถที่จะแก้ไขค่าในช่องนั้นๆได้ โดยการพิมพ์ค่าหรือข้อความที่ต้องการลงไปและเมื่อกดปุ่มเลือก pick ที่ mouse โปรแกรมจะให้ข้อความว่า Cancel หรือ OK ต่อท้ายช่องนั้นๆ ซึ่งเป็นการให้เรารับรอง (OK) ค่าที่แก้ไขนั้น หรือ จะยกเลิก ก็ได้ วิธีการก็ทำได้โดยการชี้ไปที่ข้อความที่เราต้องการและกดปุ่มเลือก

สำหรับช่องสี่เหลี่ยมที่แสดงเครื่องหมายขีดถูกจะมีอยู่ 2 ลักษณะด้วยกัน คือ แบบกำหนดค่าเป็น on หรือ off ในลักษณะนี้เมื่อเราเลื่อนเคอร์เซอร์ไปยังช่องที่ว่างอยู่ และกดปุ่มเลือกที่ช่องนั้นจะถูกเขียนเป็นเครื่องหมายขีดถูกลงไป ซึ่งแสดงถึงการตั้งสถานะให้ตัวแปรนั้นๆเป็น ON และเมื่อกดอีกครั้งหนึ่งก็จะหายไป(หมายถึง off) สลับกันไป ส่วนอีกลักษณะหนึ่งเป็นการให้เลือกทางใดทางหนึ่งในกรณีนี้ในกลุ่มคำสั่งนั้นจะมีช่องที่มีเครื่องหมายขีดถูกเพียงช่องเดียวหมายถึงใช้ทางเลือกนั้นอยู่หากเราต้องการใช้ทางเลือกอื่นก็เพียงแต่เลื่อนเคอร์เซอร์ไปยังช่องที่เราเลือกโดยอัตโนมัติการออกจากโหมดต่างๆของคำสั่งนี้จะทำได้โดยการเลื่อนเคอร์เซอร์ไปที่ช่อง OK หรือ Cancel แล้วกดปุ่มเลือก

DDRMODES (Drawing aids) การตั้งค่าตัวแปรของเครื่องมือช่วยในการเขียนต่างๆได้แก่ snap ,grid,axis,ortho,blipmode base และ isoplane

DDEMODES (entity creation) การกำหนด layer ที่จะใช้เขียนชิ้นวัตถุต่างๆ รวมไปถึงการกำหนดสี(color) ชนิดของเส้นสาย (ltype) ระดับที่เขียน (elevation) ความหนาของเส้น (thickness)

DDLMODES (modify layer) การแก้ไขสถานะต่างๆของ layer ที่มีอยู่ไม่ว่าจะเป็นการเปิด การ freeze การกำหนดสี ชนิดของเส้นสาย รวมไปถึงการกำหนด layer ปัจจุบัน หรือ current layer และการสร้าง layer ขึ้นใหม่ (new layer)

-ข้อดีของคำสั่งเหล่านี้ คือ

นอกจากจะเปิดโอกาสให้เราเรียกใช้ได้ตลอดเวลาไม่ว่าจะค้างอยู่ในคำสั่งใดก็ตามเพราะว่าเป็นคำสั่งพวก Transparent แล้ว ในแต่ละ mode ที่ขึ้นแบบฟอร์มขึ้นมาบนจอภาพ เราจะสามารถตั้งค่าของตัวแปรต่างๆ ของหลายๆคำสั่งได้พร้อมๆกัน โดยไม่ต้องเรียกคำสั่งเหล่านี้มาแก้ไขเป็นเรื่องๆไปนอกจากนี้เรายังสามารถแก้ไขค่าต่างๆกลับไปกลับมาจนพอใจเสียก่อน แล้วจึงสั่งให้โปรแกรมทำงานตามที่เรากำหนดรวดเดียวในภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การใช้คีย์ คีย์บอร์ด(Keyboard) ในการดรออิง

การใช้คีย์บอร์ด ในการดรออิงนั้นมีคีย์(key) ที่สำคัญดังนี้

ตารางที่2.1 แสดงการใช้คีย์บอร์ด

AutoCAD	function On keyboard
CTRL	Ctrl
FLIP SCREEN	F1 or Ctrl-D
TOGGLE COORDINATES	F6 or Ctrl-G
TOGGLE GRID	F7 or Ctrl-O
TOGGLE SNAP	F9 or Ctlr-B
TOGGLE TABLET	F10 or Ctrl-T
MENU CURSOR	Ins
SCREEN CURSOR	Home
ABORT CURSOR	End
FAST CURSOR	PgUp
SLOW CURSOR	PgDn
CANCEL	Ctrl-C
TOGGLE ISO-PLANE	Ctrl-E

ตารางที่2.2 Keyboard with Dialogue Box

Dialogue	function On keyboard
Pick button	Enter or Ctrl-M
Change focus in dialogue	Tab or Ctrl-I
Cancel dialogue box	Esc or Ctrl-[or Ctrl-C

ข้อดี

1.การใช้คีย์บอร์ดouhเหมาะกับผู้ที่เริ่มใช้โปรแกรมออโตแคด ใหม่ๆเนื่องจากยังไม่ชินกับการหาตำแหน่งของcommandบนเมนู

2.เหมาะใช้ควบคู่กับเมาส์ (mouse)ซึ่งจะได้ใช้มือซ้ายช่วยในการเรียกคอมมานด์ต่างๆ ควบคู่ไปกับเมาส์ซึ่งช่วยเพิ่มความเร็วในการ ดรออิง มากขึ้นอีก เพราะคอมมานด์ที่ใช้ส่วนมากจะมีรหัสคำสั่งย่อ ของมันเอง เช่นคอมมานด์ CIRCLR ก็ใช้ C แทน,คอมมานด์ POLYLINE ก็ใช้ PLแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสีย

1.ช้ามากเมื่อมีการเคลื่อนย้ายเคอร์เซอร์ ในการดรออิง ถึงแม้นจะมีคำสั่งช่วยเพิ่มระยะทางstep)ในการเคลื่อนย้ายเคอร์เซอร์ ก็ตาม

2.5 การใช้เมาส์ในการดรออิง

Mouse มีปุ่มอยู่3 ปุ่มซึ่งมีหน้าที่ดังนี้

LEFT-BOTTOM MOUSE - single select object ,
window select object,
start point select

RIGHT-BOTTOM MOUSE - Last command select,
end object select,
crossing select object

MIDDLE-BOTTOM MOUSE - เรียก Object Snap

ข้อดี

1.มีความสะดวกมากในการเคลื่อนย้ายตำแหน่งเคอร์เซอร์

2.เพิ่มประสิทธิภาพนการเรียก object snap ช่วยเพิ่มความแม่นยำในการเลือกจุดที่ต้องการ ทำให้ดรออิง ออกมานั้นเส้นมีการต่อเนื่องกัน ไม่ขาดจากกัน

ข้อเสีย

1.เนื่องจากในบางครั้งผู้เขียนมีความสะเพร่าในการเขียน โดยไม่ใช้ object snap ช่วยในการเขียน โดยใช้วิธีเล็งตำแหน่งเอาเอง ทำให้ ดรออิง ที่ออกมานั้นเส้นไม่ต่อเนื่องกัน ซึ่งถ้ามองทางจอภาพ อาจดูเหมือนว่าต่อกันก็ตามแต่เมื่อพล็อตดรออิง ออกมาแล้วจะพบว่าเส้นนั้นไม่ต่อกัน

2.ในการเคลื่อนย้ายเมาส์นั้น บางครั้งผู้เขียนอาจลืมกดปุ่ม snap-ON(F9) ทำให้ภาพที่ออกมานั้นมีระยะทางไม่ถูกต้อง

3.สำหรับผู้ที่เพิ่งเริ่มใช้จะไม่มีความชำนาญในการหาตำแหน่งของคอมมานด์บนเมนู จึงควรใช้ คีย์บอร์ด ช่วยในการเรียกคอมมานด์

4.ผู้ใช้ส่วนมากในระยะเริ่มแรกจะไม่รู้ว่ามีมือซ้ายเอาไว้ทำอะไร เนื่องจากมือขวาจับเมาส์ใช้งานอยู่มือเดียว ดังนั้นควรใช้มือซ้ายช่วยในการกดคีย์บอร์ด เรียก code command เพื่อเพิ่มความสะดวกได้

2.5 การใช้ เทปเบล็ตในการดรออิง

เทปเบล็ตคือ อุปกรณ์ที่ใช้ในการชี้ภาพบนจอ คล้ายๆกับเมาส์ แต่เทปเบล็ตสามารถกำหนดโคออร์ดิเนตด้วยการชี้ไปที่แผ่นของตัวเทปเบล็ต เองได้

เราสามารถจะสร้างเมนูของคำสั่งไว้บน เทปเบล็ตได้ (เรียกว่า Tablet Menu) โดยตัวเมนูนี้จะอยู่บนส่วนใดของเทปเบล็ต ก็ได้ และสามารถมีได้ถึง 4 เมนู ในเมนูเราจะแบ่งช่อง แต่ละช่องมีคำสั่ง 1 คำสั่ง เมื่อต้องการใช้คำสั่งใดก็เลื่อนตัวชี้ที่มีสายใยไปที่ช่องนั้น แล้วกดปุ่มบนตัวชี้ คำสั่งที่ตรงกับที่เขียนไว้ในช่องนั้นก็จะออกมาทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมนูบนตัวชี้

ตัวชี้ของ เทปเบล็ต หรือบน เมาส์ จะมีปุ่มกดอยู่หลายปุ่ม ปุ่มๆหนึ่งจะใช้เป็นตัวหยิบ(pick) คือ สำหรับการกำหนดจุด ปุ่มอื่นๆจะเป็นปุ่มสำหรับเมนู (Button Menu)

การทำสำเนาของแบบแปลนบนกระดาษ

ปกติ เทปเบล็ต จะทำหน้าที่เป็นตัวชี้บนจอภาพ กล่าวคือ เมื่อเลื่อนตัวชี้ไปทางไหนเส้นสายใยบนจอภาพจะเลื่อนตาม ตำแหน่งของตัวชี้บน Tablet จะไม่เกี่ยวกับโคออร์ดิเนตบนจอภาพแต่ประการใด

ในกรณีที่เราต้องการจะให้ตำแหน่งบนเทปเบล็ต สัมพันธ์โดยตรงกับโคออร์ดิเนตบนจอภาพ อย่างเช่นจะทำสำเนา(Copy) รูปที่ถูกสเกลที่วางไว้บน เทปเบล็ตลงบนจอภาพ โดยให้โคออร์ดิเนตของรูปตรงกับโคออร์ดิเนตบนจอภาพนั้น เราจะต้องเข้าไปใน Tablet Menu ด้วยการใช้คำสั่ง TABLET

คำสั่ง TABLET

มีทางเลือกต่างๆ ดังนี้

ON ให้ Tablet Mode อยู่ในสภาพ ON แต่ทั้งนี้จะต้องมีการปรับค่า ด้วย "CAL" เสียก่อน

OFF ให้ Tablet Mode เป็น OFF

CAL สำหรับปรับค่า (Calibrate) ของเทปเบล็ตกับแบบแปลนที่วางอยู่บนเทปเบล็ต

CFG สำหรับสร้างหรือเปลี่ยนแปลงเมนูบนเทปเบล็ต (Tablet Menu)

-คำสั่ง TABLET CAL

เป็นคำสั่งที่ให้ Tablet Mode เป็น ON และปรับค่าตัวเทปเบล็ตดังนี้

Command : TABLET

Option (ON/OFF/CAL/CFG) : CAL

Calibrate tablet for use ...

Digitize first known point : (ชี้ไปที่จุดที่รู้ค่าโคออร์ดิเนต)

Enter coordinates for first point :

(บอกค่าโคออร์ดิเนต เช่น 1,1)

Digitize second known point :

(ชี้ไปที่จุดอีกจุดหนึ่งที่รู้ค่าโคออร์ดิเนต)

Enter coordinates for second point L:

(บอกค่าโคออร์ดิเนต เช่น 20,10)

จุดสองจุดที่รู้โคออร์ดิเนตนั้นไม่จำเป็นจะต้องเป็นแกน จะเป็นจุดไหนก็ได้ที่ไม่ใช่จุดเดียวกัน หลังจากที่ปรับค่าแล้ว เราสามารถจะใช้คำสั่งของ ออโตแคด ได้ตามปกติ

ข้อควรระวัง ขอบเขตของแบบแปลนยังใช้อยู่ ใช้คำสั่ง LIMITS เพื่อให้แน่ใจว่าแบบแปลนที่เขียนอยู่ในขอบเขต ถ้าเราเลื่อนแบบแปลนที่เขียนไป ถ้าจะเขียนต่อจะต้องทำการปรับค่าใหม่

-คำสั่ง TABLET OFF เป็นคำสั่งที่ทำให้ออกจาก Tablet Mode

Command : TABLET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Option (ON/OFF/CAL/CFG) : OFF

- -คำสั่ง TABLET ON

เป็นคำสั่งที่ให้เข้า Tablet Mode

Command : TABLET

Option (ON/OFF/CAL/CFG) : ON

ถ้ายังไม่ปรับค่า จะเข้า Tablet Mode ไม่ได้ จะต้องทำการปรับค่าเสียก่อน อาจจะใช้ Ctrl T หรือ F10 เป็นสวิตช์ เปลี่ยนไปมาระหว่าง ON กับ OFF ก็ได้

- คำสั่ง TABLET CFG

เป็นคำสั่งสำหรับสร้างเมนูบนพื้นกระดานของเทปเบล็ต เราเรียกเมนูนี้ว่าเทปเบล็ตเมนู(Tablet Menu) เราสามารถจะสร้างเมนูแบบนี้ได้ถึง 4 พื้นที่ แต่ละพื้นที่จะแบ่งเป็นช่องเล็กๆ แต่ละช่องจะจัดไว้สำหรับคำสั่ง 1 คำสั่ง เมื่อเราต้องการออกคำสั่งใดแก่เครื่องแทนที่เราจะพิมพ์คำสั่งโดยตรงด้วยแป้นพิมพ์ เราอาจใช้ตัวชี้เลื่อนไปที่ช่องที่มีคำสั่งอยู่ แล้วกดปุ่ม 'PICK' ที่อยู่บนตัวชี้นั้นจะออกมาเอง ลองมาสร้าง Tablet Menu เพียง 1 พื้นที่ดู ดังนี้

Command : TABLET

Option (ON/OFF/CAL/CFG) : CFG

Enter number of tablet menus desired (0-4) : 1

Digitizing upper left corner of menu area 1;

(เลื่อนตัวชี้ไปที่จุดมุมบนซ้ายของพื้นที่ที่จะให้เป็นเมนู)

Digitizing lower left corner of menu area 1;

(เลื่อนตัวชี้ไปที่จุดมุมล่างซ้ายของพื้นที่ที่จะให้เป็นเมนู)

Digitizing lower right corner of menu area 1;

(เลื่อนตัวชี้ไปที่จุดมุมล่างขวาของพื้นที่ที่จะเป็นเมนู)

Enter the number of column for menu area 1 : 2

(เราจะแบ่งเป็น 2 คอลัมน์)

Enter the number of rows for menu area 1 : 10

(จะมี 10 แถว)

ข้อดี

1.มีความสะดวกในเคลื่อนย้ายเคอร์เซอร์

2.สามารถหาตำแหน่งของคอมมานด์ได้ดีกว่าใช้เมาส์เนื่องจาก สามารถกำหนดตำแหน่งต่างๆบนเทปเบล็ตให้ตรงกับตำแหน่งของจอภาพได้ และสามารถกำหนดตำแหน่งของคอมมานด์ ที่ใช้บ่อยๆให้อยู่บนเทปเบล็ต ได้เองโดยใช้โปรแกรมออโตลิปส์ ช่วย ทำให้ไม่ต้องเสียเวลาในการวนหาคอมมานด์บนเมน

3.ช่วยในการกำหนดตำแหน่งของภาพได้ดีกว่าเมาส์มากเพราะมีลักษณะคล้ายปากกาซึ่งจับถนัดมือกว่ามากและ เหมาะในการวาดภาพที่มความคดโค้ง หรือดภาพที่ลอกจากต้นแบบได้ดี โดยอาศัยcommand SKETCH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสีย

1.เกะกะพื้นที่ในการวางเพราะเทปเบล็ต มีขนาดใหญ่มาก ทำให้ไม่สะดวกเมื่อใช้ในพื้นที่ๆจำกัด

2.สำหรับผู้ที่ยังไม่ชำนาญ ในการดรออิง ถ้าจอภาพ มีขนาดเล็กจะทำให้ผู้ใช้ต้องเมื่อยคให้การเงยขึ้นลงระหว่างการดูภาพบนจอภาพ และการดูเมนูบนตัวเทปเบล็ต

2.7 การรับ ดรออิง มาจากสแกนเนอร์ (scanner)

เนื่องจากอุปกรณ์ เทปเบล็ตมีราคาแพงมาก ร่วมทั้งความไม่สะดวกในเรื่องสถานที่เราจึงอาศัยความสามารถของโปรแกรมออโตแคด คือความสามารถในการรับ Drawing fileจากโปรแกรมกราฟิกทั่วไป และทำการimport เข้ามาใช้กับโปรแกรม ออโตแคด ได้ และโปรแกรม ออโตแคด เองก็ยังสามารถส่งDrawing file ออกไป(Export) เพื่อใช้กับโปรแกรมอื่นได้เช่นเดียวกันทั้งนี้เนื่องจากออโตแคดเป็นโปรแกรมที่ใช้ในการ draftโดยเฉพาะ

โดยอาศัยคำสั่ง IMAGE IMPORT/EXPORT ซึ่งใช้ได้กับ drawing graphic file *.EPS, *.DXF, *.DXB, *.ISP

2.8 การแสดง ดรออิง ออกทาง จอภาพ

ลักษณะของจอภาพ รแสดงข้อมูลต่างๆใน ออโตแคด นั้นจะแสดงอยู่บนจอภาพ ซึ่งมีชื่อเรียกเป็น 2 ชนิด คือ Text screen คือ ลักษณะของจอภาพที่แสดงข้อมูลในรูปของตัวหนังสือ และจอภาพกราฟิก (Drawing editor screen) คือ ลักษณะของจอภาพที่แสดงข้อมูลในรูปของรูปภาพในการใช้จอภาพเพียงจอเดียว เราจะใช้ F1 ใช้ในการปรับไปมาระหว่างจอภาพทั้ง 2 นี้

สำหรับจอภาพกราฟิก จะเป็นจอภาพที่ใช้ในการเขียนภาพลงไป ซึ่งเรียกอีกอย่างหนึ่งว่า Drawing Editor ซึ่งจะถูกแบ่งออกเป็นส่วนๆ ดังนี้ คือ

1. Graphic Area คือบริเวณที่ใช้เขียนรูปหรือมองเห็นภาพที่เขียน บน Graphic area นี้จะมีกากบาทเกิดขึ้นบนจอ เรียกว่า เคอร์เซอร์ ซึ่งเป็นเครื่องหมายบอกตำแหน่งของการเขียน (Coordinate หรือ Point) ซึ่งสามารถเลื่อนไปมาได้ ตำแหน่งที่ใช้กำหนด คือ ตำแหน่งจุดตัดของกากบาทนั่นเอง

2. Command Line มีอยู่ 3 บรรทัด เป็นบริเวณที่แสดงข้อความ คำสั่ง คำถามต่างๆที่เกิดขึ้น ซึ่งข้อความเหล่านี้จะปรากฏในจอภาพตัวอักษร ด้วย แต่แสดงให้เห็นถึง 24 บรรทัด

3. Menu Area เป็นบริเวณที่แสดงรายการคำสั่งที่สามารถเลือกใช้ได้ แทนการพิมพ์ลงไปที่ Command line

4. Status Line เป็นบรรทัดที่แสดงสถานะต่างๆ ของโปรแกรม โดยมีการแสดงรายละเอียดต่างๆ ดังนี้

LAYER xxx เป็นชื่อของแผ่นงาน(layer) ที่กำลังเขียนลงไป xxx หมายถึงชื่อของแผ่นงานนั้นๆ

O(Ortho),S(Snap) เป็นสถานะของการกำหนดตำแหน่งโคออร์ดิเนตในการเขียนรูปโดยที่หากมีข้อความขึ้นมาจะแสดงว่า สถานะเหล่านั้นถูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดอยู่(ON) และไม่ได้กำหนดอยู่ (OFF) ในกรณีที่ไม่ข้อความเหล่านี้

X,Y เป็นตำแหน่งโคออร์ดิเนตของเคอร์เซอร์ในขณะนั้น

2.8การพล็อต

การพล็อตดรออิง ด้วย ออโตแคด ด้วยเครื่องพล็อต (Plotter) เครื่องพล็อตนั้นละเอียดและแน่นอนสามารถพล็อตบนกระดาษใหญ่และพล็อตได้หลายๆสี

การprint ออกทางเครื่องพิมพ์(Printer) นั้นความละเอียดค่อนข้างจำกัดและกระดาษได้แผ่นเล็กprintได้สีเดียว แต่จะให้ผลเร็วกว่า (ถ้าใช้Laser printer จะช่วยเพิ่มความเร็ว และความคมชัดของภาพได้อย่างมาก)เราอาจจะให้มีการพล็อตได้ด้วยการเลือกจากเมนเมนู หรือจากดรออิงอิดิตเตอร์ ในการเลือกนั้นเราควรที่เลือกจากเมนเมนู และเลือกตัวเลือกในเมนเมนูให้เหมาะสมกับเครื่องและแบบของเครื่องที่เราใช้ในการพล็อตด้วย

ในการพล็อตชิ้นงานที่เขียนไว้สามารถทำได้ด้วยคำสั่ง PLOT ซึ่งอยู่ใน Drawing Editor และเมื่อเรียกใช้คำสั่งนี้จะมีไดอะลอกบล็อก ขึ้นมาเพื่อให้เราทำการกำหนดรูปแบบที่ต้องการพล็อต ได้ เช่น การกำหนดขนาดของชิ้นงานที่เขียนขึ้น เป็นต้น

2.9 การแสดงภาพโดยเทคนิค สไลด์

ใน ออโตแคด มีคำสั่งที่สามารถนำภาพในส่วนของ graphic area ขณะนั้นไปเก็บไว้เป็นไฟล์ เพื่อที่จะนำภาพนั้นกลับมาแสดงบนจอได้ในภายหลัง ซึ่งภาพที่เก็บไว้เป็นไฟล์เรียกว่า สไลด์ไฟล์ (ลักษณะคล้ายกับการบันทึกจอภาพ (save screen)ในโปรแกรมอื่น)

ก่อนที่จะทำการสร้าง สไลด์ จะต้องจัดภาพบนจอบริเวณ graphic area ให้ได้ตามต้องการเสียก่อน หลังจากนั้นสร้าง สไลด์ โดยใช้คำสั่ง MSLIDE(MAKE SLIDE) ทำการเก็บภาพลงในไฟล์จะได้ไฟล์ที่มีนามสกุล .SLD เราสามารถนำภาพ สไลด์ ที่มีอยู่แล้วขึ้นมาแสดงบนจอโดยใช้คำสั่ง VSLIDE (view slide) ได้

ภาพสไลด์ ที่ปรากฏจะทับภาพบนจอภาพขณะนั้นเป็นการชั่วคราว และจะหายไปเพื่อคืนจอภาพตามปกติด้วย redraw เราไม่สามารถนำภาพ สไลด์ ที่เก็บไว้นั้นมาแก้ไขได้ เนื่องจากข้อมูลที่เก็บอยู่ในรูปของ Display vector ที่จะทำให้การแสดงภาพบนจอเป็นไปอย่างไรหากต้องการแก้ไข สไลด์ จะต้องแก้ไขภาพที่เป็นต้นกำเนิด สไลด์ จากนั้นใช้คำสั่ง MSLIDE สร้าง สไลด์ ใหม่

-การสร้าง Slide Library File

จากการสร้าง สไลด์ ด้วย MSLIDE ที่ผ่านมา จะพบว่า 1 สไลด์ จะมีไฟล์เกิดไฟล์(.SLD) เพื่อลดจำนวน สไลด์ไฟล์ ในดิสค์ลง

ลำดับขั้นการสร้าง Slide Library File

1. สร้าง สไลด์ ต่างๆที่ต้องการ โดยเขียนภาพบนจอแล้วใช้คำสั่ง MSLIDE

2. รวบรวมชื่อ สไลด์ ต่างๆ ที่จะเก็บเข้าใน Library ทำเป็น ไฟล์ขอ้มูล โดยใช้โปรแกรมต่างๆ เช่น EDLIN ,SIDEKICK,Turbo หรือใช้คำสั่งของ DOS ก็ได้

3. ทำการคอมไพล์ให้เกิด slide Library File ขั้นตอนนี้จะต้องปฏิบัติขณะอยู่ที่คอมไพล์ หากใครไม่มี SLIDELIB.EXE นี้ไม่สามารถสร้าง Library File ได้ รูปแบบใช้ไฟล์ SLIDELIB.EXE มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้นำเอกสารนี้ไปเผยแพร่ต่อ ในรูปแบบอื่นๆ ไม่ว่ากรณีใดๆ

SLIDELIB LIBFILE < SLDLIST

Text File ที่รวบรวมชื่อ สไลด์ ต่างๆไว้

ชื่อ Slide Library File ที่ต้องการ

-การเรียก สไลด์ จาก Slide Library File

คำสั่งที่ใช้สำหรับเรียก สไลด์ ใน Library File ก็ใช้ VSLIDE เช่นเดียวกัน แต่การระบุชื่อ Library File และ ชื่อ สไลด์ ที่ต้องการมีรูปแบบดังนี้

LIBRARY FILE (SLIDE NAME)

ชื่อ สไลด์ที่ต้องการ

ชื่อ Library File ที่บรรจุ สไลด์ ไว้

2.10 การแสดงภาพโดยเทคนิค สคลิปไฟล์

-การใช้คลิป

เราสามารถที่จะนำภาพ สไลด์ มาแสดงแบบอัตโนมัติในลักษณะวนฉายไปเรื่อยๆได้ คล้ายกับการแสดงภาพกราฟิกในงานโชว์คอมพิวเตอร์หรืองานโชว์อื่นๆ , โดยการนำกลุ่มคำสั่งแสดงภาพสไลด์ ต่างๆ ที่ต้องการฉาย มาเขียนเป็น สคลิปไฟล์ (.SCR) แล้วเรียก สคลิปไฟล์ นั้นมาใช้งาน โดยใช้คำสั่ง SCRIPT การนำกลุ่มคำสั่งมาเขียนเป็น สคลิปไฟล์จะคล้ายๆกับการนำกลุ่มคำสั่งเขียนเป็น Menu File แต่ สคลิปไฟล์ ไม่สามารถที่จะหยุดรอรับข้อมูลได้

-ขั้นตอนการสร้าง สคลิปไฟล์ มีดังนี้

1. ทำการสร้าง สไลด์ ของภาพแต่ละขั้นตอน

2. สร้าง สคลิปไฟล์ File ชื่อ MYSCRIPT.SCR โดยใช้ Text Editor เขียนกลุ่มคำสั่งที่จะแสดง การควบคุมให้ สไลด์ แต่ละภาพหยุดแสดงให้ดูชั่วขณะ เราควบคุมได้โดยใช้ Delay ซึ่งหน่วยของเวลาที่ใช้เป็นมิลลิวินาที ส่วนเครื่องหมาย * ที่นำหน้าชื่อ สไลด์ หมายถึงให้ ออโตแคด โหลดภาพสไลด์ นั้นเข้ามาก่อน (preloads) เป็นการเพิ่มความเร็วของการแสดงสไลด์ สุดท้าย RSCRIPT จะควบคุมให้เริ่มทำงานใหม่ตั้งแต่ต้น สคลิปไฟล์ ถ้าหากต้องการให้แสดง 1 รอบแล้วเลิก ก็ไม่ต้องมี RSCRIPT

-การเรียกใช้งานสคลิปไฟล์ ก็จะใช้คำสั่ง SCRIPT ตามด้วยชื่อ สคลิปไฟล์ (.SCR) ไม่ต้องใส่นามสกุล .SCR ลงไป ออโตแคด จะเริ่มปฏิบัติตามคำสั่งที่อยู่ในบรรทัดที่ 1 ก่อน แล้วทำต่อไปเรื่อยๆจนหมดไฟล์ หากต้องการหยุดให้กด Ctlr C หรือ Backspace เครื่องจะกลับไปที่ Command รอรับคำสั่งต่อไป ในกรณีที่ต้องการให้ทำงานต่อจากภาพที่หยุด ให้ใช้คำสั่ง RESUME ออโตแคด จะเริ่มทำบรรทัดคำสั่งที่ต่อท้ายจากบรรทัดที่สั่งให้หยุด

2.11 การสร้างสกรีนเมนู(Screen menu)

สกรีนเมนูเป็นเมนูคำสั่งที่อยู่ข้างขวามือของจอภาพ ซึ่งออโตแคดจะมีสกรีนเมนูมาตราฐานอยู่ในไฟล์ ACAD.MNU หรือ ACAD.MNX ซึ่งไฟล์นี้จะได้มาจากการคอมไฟล์(Compileing) ไฟล์ ACAD.MNU (ไฟล์ที่อยู่ในรูปของไฟล์ตัวอักษร)

เราสามารถสร้างสกรีนเมนูได้โดยเก็บข้อมูลเมนูไฟล์ให้อยู่ในรูปของไฟล์ข้อมูล โดยใช้เทกซ์อิดิตเตอร์ต่างๆ และตั้งชื่อไฟล์ให้มีนามสกุล.MNU เมื่อเราต้องการนำสกรีนเมนูที่สร้างนั้นขึ้นมาใช้เรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถทำได้โดย ใช้คำสั่ง MENU แล้วเลือกไฟล์ที่เก็บสกรีนเมนูที่สร้างไว้ ถ้าเป็นเมนูใหม่ ออโตแคดจะทำการคอมไพล์ใหม่จะได้ **.MNX แล้วก็จะสามารถใช้สกรีนเมนูที่สร้างไว้นั้นได้

โครงสร้างของสกรีนเมนู ประกอบด้วย

1.ชื่อกลุ่มคำสั่ง (Macro Labels) เป็นชื่อของกลุ่มคำสั่ง(Macro)เช่น Zoom A, Zoom E ฯลฯ ชื่อของกลุ่มคำสั่งนี้จะไปปรากฏบนจอภาพเพื่อให้ผู้ใช้ได้เลือกใช้งานได้ ข้อความที่เป็นเลเบล(Label) นี้จะอยู่ในเครื่องหมาย [](Square brackets) ความยาวของอักษรที่จะบรรจุลงในเครื่องหมาย []เพื่อเป็นเลเบลนี้สูงสุด 8 ตัวอักษรและสามารถเว้นวรรคได้ ชื่อของกลุ่มคำสั่งนี้เป็นเพียงฉลากข้อความสำหรับแต่ละคำสั่งเท่านั้น อาจจะไม่สำคัญมากนัก เพียงแต่ให้สื่อความหมายได้เท่านั้น ส่วนการเลือกกลุ่มคำสั่งจะต้องใช้อุปกรณ์ชี้หรือใช้คีย์ลูกศรเลื่อนไปเลือกเท่านั้น ไม่สามารถพิมพ์ชื่อกลุ่มคำสั่งลงไปได้

2.คำสั่งปฏิบัติการ(Macro Command) เป็นคำสั่งหรือกลุ่มคำสั่งเพื่อให้ออโตแคดไปปฏิบัติ ข้อความที่อยู่หลังเครื่องหมาย] (right square bracket)ออโตแคดจะถือว่าเป็นกลุ่มคำสั่งที่จะให้ปฏิบัติจะต้องพิมพ์ดีดเครื่องหมาย]เลย ห้ามเว้นวรรค

3.ส่วนขยายของคำสั่ง (Parameter) เป็นส่วนขยายของคำสั่งต่างๆ ถ้ามี เช่น การ Zoom มีหลายลักษณะ ถ้าต้องการ Zoom all ส่วนขยายของคำสั่ง Zoom คือ A บางคำสั่งอาจมีพารามิเตอร์หรือส่วนขยายนี้หลายตัว บางคำสั่งอาจไม่มีเลยก็ได้ สำหรับส่วนขยายจะต้องอยู่ห่างจากคำสั่งอย่างน้อย 1 ตัวอักษรโดยเว้นช่องว่างไว้

การควบคุมให้เมนูหยุดรับข้อมูล ถ้าต้องการให้มีการหยุดรับข้อมูลช่วงใด จะต้องใช้เครื่องหมาย \ (back slashes) แทรกไว้ที่ช่วงนั้น และเครื่องหมาย ; (Semi colon) มีความหมายเป็นการกด Enter

-อักขระพิเศาที่ใช้ควบคุมเมนู(Special Characters)

ตารางที่2.3 แสดงอักขระพิเศษที่ใช้ในการควบคุมเมนู

อักขระ	หน้าที่
\	หยุดรอรับข้อมูลจากอุปกรณ์ชี้หรือคีย์บอร์ด
+	ให้บรรทัดถัดไปเป็นส่วนที่ต่อกับเครื่องหมาย +
^D	ปิด/เปิด โคออร์ดิเนต read out
^G	ปิด/เปิด grid

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่2.3 (ต่อ)

อักขระ	หน้าที่
^C	ยกเลิกคำสั่ง (Cancel)
^P	ควบคุม MENU ECHO
^T	ควบคุม Tablet
^B	ปิด/เปิด snap
;	มีค่าเป็นการกด Enter
^M	มีค่าเป็นการกด Enter
[]	เริ่มต้นและสิ้นสุดข้อความที่เป็นเลเบล
^E	ควบคุม Isoplan
^H	มีค่าเท่ากับกดแป้นถอยหลัง(back space)
^O	ปิด/เปิด ORTHO MODE
^Q	ควบคุมการส่งข้อมูลไปเครื่องพิมพ์
^X	ลบข้อมูลในบัฟเฟอร์ (buffer)

-การสร้างสกรีนเมนูแบบหลายหน้า

ใน 1 หน้าหรือ 1 สกรีนสามารถมีเมนูคำสั่งได้ 21 รายการสำหรับจอ EGA และ monochrome ถ้าเป็นจอ VGA จะได้ 26 รายการ แต่ถ้าใช้จอชนิด ADI (AutodeskDriver Interface)จะได้ถึง 34 บรรทัด หากมีรายการมากกว่านี้จะต้องแบ่งรายการเมนูออกเป็นหน้า โดยที่แต่ละหน้าจะต้องมีรายการเมนูไม่เกินความสามารถของจอ เมื่อแบ่งรายการเมนูออกเป็นหลายๆหน้าแล้ว จะต้องมีการกำหนดชื่อหน้าแต่ละหน้าไว้ด้วย เพื่อที่จะเรียกใช้ต่อไป

-รูปแบบการกำหนดชื่อหน้า มีดังนี้

**Pagename [Line]

บรรทัดที่เริ่มแสดงเมนู

ชื่อหน้าของรายการเมนู

-การเรียกเมนูหน้าอื่นมาใช้งาน

การกำหนดให้ออโตแคดนำเมนูหน้าต่างๆขึ้นมาปรากฏบนจอ มีรูปแบบกำหนดดังนี้

$S=Pagename

โดยปกติถ้าไม่มีการกำหนด Line (บรรทัดที่เริ่มแสดงเมนู)ที่ชื่อเมนูแต่ละหน้าออโตแคดจะเริ่มแสดงรายการตั้งแต่บรรทัดที่ 1 เสมอ แต่ถ้ามีการกำหนด Line เมื่อเมนูหน้านั้นถูกเรียกใช้งานออโตแคด จะเริ่มแสดงตั้งแต่บรรทัดที่กำหนดไว้ใน Line เป็นต้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 การสร้างป๊อปอัพเมนู (Poop-up Menu)

ป๊อปอัพเมนูในออโตแคด จะปรากฏก็ต่อเมื่อเราเลื่อนอุปกรณ์ชี้ให้เคอร์เซอร์ขึ้นไปส่วนบนสุดของจอภาพบริเวณ Status line ซึ่งเดิมสถานะเลเยอร์ และโคออร์ดิเนต ก็จะกลายเป็นเมนูทันที หากต้องการให้ป๊อปอัพเมนูหายไปก็เลื่อนอุปกรณ์ชี้ให้เคอร์เซอร์ลงมาจากStatus line เท่านั้นเอง สถานะเลเยอร์ และโคออร์ดิเนต ก็จะปรากฏตามเดิม จะเห็นว่าต่างกับสกรีนเมนู เนื่องจากป๊อปอัพเมนูมีการเปิดเมื่อใช้งานและถูกปิดเมื่อเลิกใช้งาน ทำให้ประหยัดเนื้อที่บนจอภาพ

การสร้างป๊อปอัพเมนูทำได้เช่นเดียวกับสกรีนเมนู โดยการเขียนป๊อปอัพเมนูในเทกซ์อิดิตเตอร์ แล้วเก็บไว้เป็นไฟล์ตัวอักษรในนามสกุล .MNU

จะเห็นว่าไฟล์ที่เก็บข้อมูลเกี่ยวกับเมนูไม่ว่าจะเป็นสกรีนเมนูหรือป๊อปอัพเมนูจะอยู่ในนามสกุล .MNU เหมือนกันอธิบายได้ว่าเมนูทุกชนิด จะรวมอยู่ไฟล์เดียวกัน แต่จะมีรหัสแยกว่าส่วนไหนคือ สกรีนเมนู ส่วนไหนคือป๊อปอัพเมนู เรียกว่า การกำหนดดีไวซ์ (device) ซึ่งมีรูปแบบดังนี้

***Device

ตารางที่2.4 แสดงรหัสและความหมายการกำหนดดีไวซ์

รหัสกำหนด	ความหมาย
***BUTTONS	หน้าที่ของปุ่มบนอุปกรณ์ชี้
***AUX1	หน้าที่ของปุ่มอุปกรณ์ต่าง ๆบนอุปกรณ์ช่วย
***POP1	ป๊อปอัพเมนู คอลัมน์ที่ 1
***POP2	ป๊อปอัพเมนู คอลัมน์ที่ 2
***POP3	ป๊อปอัพเมนู คอลัมน์ที่ 3
***ICON	รายการเมนูรูปภาพ(Icon Menu)
***SCREEN	รายการสกรีนเมนู
***TABLET1	เมนูบน Tablet ส่วนที่ 1
***TABLET2	เมนูบน Tablet ส่วนที่ 2

-การสร้างป๊อปอัพเมนูหลายหน้าใน 1 คอลัมน์

ในกรณีที่รายการเมนูใน 1 คอลัมน์ มีมากเกินกว่าที่จะบรรจุลงใน 1 หน้า เราจำเป็นที่จะต้องแบ่งออกเป็นหลาย ๆหน้า เช่นเดียวกับสกรีนเมนู โดยใช้ ** นำหน้าชื่อ page

ส่วนการที่จะให้หน้าใดของป๊อปอัพเมนู มาปรากฏให้ใช้รูปแบบดังนี้

$Pn=PAGENAME $Pn=*

โดยที่ n คือหมายเลขคอลัมน์ของป๊อปอัพเมนู คอลัมน์ที่ 1 ถึง 10

PAGENAME คือชื่อหน้าของป๊อปอัพเมนูที่ต้องการให้ปรากฏ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 การสร้างเมนูรูปภาพ (Icon menu)

เมนูรูปภาพ คือ รายการเมนูรูปภาพซึ่งต่างกับ ป๊อปอัพเมนูและสกรีนเมนู ซึ่งเห็นเป็นตัวอักษร สำหรับการเลือกรายการต่างๆของเมนูรูปภาพสามารถทำได้ โดยใช้อุปกรณ์ชี้เลื่อนตัวชี้ไปยังกรอบสี่เหลี่ยมเล็กที่อยู่ด้านซ้ายของรายการที่เราต้องการเลือกแล้วกดปุ่มเลือกที่อุปกรณ์ชี้ เมนูรูปภาพในออโตแคด จะใช้กลุ่มของสไลด์ มาเป็นกราฟิกเลเบล (graphic labels) ต่างกับเลเบลในป๊อปอัพเมนูหรือสกรีนเมนู ซึ่งจะใช้กลุ่มของตัวอักษรเป็นเลเบลสำหรับผู้ที่จะสร้างเมนูรูปภาพจะต้องสร้างสไลด์และสไลด์ไลบารีไฟล์ ได้ก่อน

-การกำหนดกราฟิกเลเบลและเทกซ์เลเบล

สำหรับเมนูรูปภาพนี้สามารถกำหนดให้เป็นได้ทั้งกราฟิก โดยนำเอาภาพจากสไลด์มาเป็นกราฟิกเลเบล และกำหนดให้เป็นเลเบลแบบอักษรได้

รูปแบบการกำหนดกราฟิกเลเบล มีดังนี้

[LIBRARY FILE (SLIDE NAME)]

ชื่อสไลด์ที่ต้องการให้เป็นกราฟิกเลเบล

ชื่อไลบารีไฟล์ที่เก็บสไลด์ไว้

รูปแบบการกำหนดเทกซ์เลเบล มีดังนี้

[_TEXT]

ข้อความที่ต้องการให้เป็นเลเบล

เว้นช่องว่างไว้ 1 อักษร

-การเรียกเมนูรูปภาพ

โดยลำพังเมนูรูปภาพเองไม่สามารถเปิดแสดงเมนูออกมาได้ จะต้องอาศัยเมนูอื่น มาช่วยกระตุ้นถึงจะเริ่มเปิดแสดงได้ การกระตุ้นจะจำเป็นสำหรับการเริ่มเปิดเมนูรูปภาพครั้งแรกเท่านั้น หลังจากที่เมนูรูปภาพปรากฏขึ้นแล้ว จึงจะสามารถกระตุ้นตัวเองให้เปิดเมนูหน้าต่อไปหรือหน้าอื่นๆ หรือแม้กระทั่งไปกระตุ้นให้เปิดเมนูหน้าต่างๆของป๊อปอัพเมนู,สกรีนเมนูก็ได้

-รูปแบบการกระตุ้นให้เปิดเมนูรูปภาพ

$I=Pagename $I=*

Pagename คือ เมนูรูปภาพ หน้าที่จะเปิด

-ลำดับการสร้างเมนูรูปภาพ มีดังนี้

1.สร้างสไลด์ที่จะให้เป็นกราฟิกเลเบลในเมนูรูปภาพ โดยเขียนภาพที่ต้องการบนจอแล้วใช้คำสั่งMSLIDEทำการเก็บภาพขณะนั้นเป็นสไลด์ลงในไฟล์การสร้างสไลด์นี้จำเป็นเฉพาะไอคอนที่มีเลเบลเป็นรูปภาพเท่านั้น

2.รวบรวมชื่อ สไลด์ ต่างๆที่ได้จากข้อที่ 1 เก็บไว้ในรูปของเทกซ์ไฟล์ มีข้อแม้ว่า 1 ชื่อสไลด์จะเป็น 1 บรรทัดและจะต้องใส่เฉพาะชื่อเท่านั้น นามสกุล .SLD ไม่ต้อง ห้ามเว้นบรรทัดหรือกดแป้นเคาะหลังชื่อสไลด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.ทำการสร้างสไลด์ไลบารีไฟล์ โดยมีรูปแบบดังนี้

SLIDELIB LIBRARYNAME < FILELIST

ชื่อไฟล์ที่ได้จากข้อ 2

ชื่อไลบารีไฟล์ที่จะตั้งขึ้น

4.สร้างเมนูไฟล์ โดยมีหลักเช่นเดียวกับสกรีนเมนูหรือป๊อปอัพเมนู

2.14 การกำหนดรูปแบบการระบาย (Shade) รูปแบบในการระบายภาพshade มีอยู่ 4 ลักษณะด้วยกัน ซึ่งรูปแบบดังกล่าวจะถูกควบคุมด้วยตัวแปรระบบ(system variable)ที่ชื่อ shadedge ดังมีรายละเอียดดังต่อไปนี้

ตารางที่2.5 แสดงรายละเอียดตัวแปรและรูปแบบในการ shade

ค่าของตัวแปร shadedge	ความหมาย	รูปแบบในการระบาย shade
0	256-col	ระบายshadeตามปกติมีการแจกแจงสีได้ 256 สี
1	256-edg	ระบาย shade เช่นเดียวกับตัวแปรที่มีค่าเป็นศูนย์แต่จะมีเส้นตัดขอบระหว่างรอยต่อของแต่ละพื้นผิว
2	Hidden	สร้างภาพโดยไม่มีการระบาย shade แต่จะลบเส้นที่ถูกบังทิ้งไป
3	Filled	ระบายภาพ shade ให้ทึบทั้งหมดโดยมีความเข้มเท่ากัน คล้ายกับการใช้คำสั่ง SOLID แต่จะมีเส้นขอบรอยต่อระหว่างพื้นผิว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15 การใช้คำสั่งการควบคุมแสงและการสร้างฉาก (Light&Scene)

2.15.1 การให้แสงด้วย Light

ในการ Render วัตถุใดๆนั้นเราสามารถที่จะปรับปริมาณของแสงที่จะใช้ได้ ซึ่งจะมีแสงอยู่ 2 ชนิดคือ

1. Point Source ลักษณะของแสงชนิดนี้จะแผ่กระจายทุกทิศทางรอบตัว เช่น แสงจากหลอดไฟแบบไส้ แสงส่องจากดวงอาทิตย์ ฯลฯ

2. Directed Light หรือแสงส่องเป็นลำ ลักษณะของแสงชนิดนี้จะส่องแสงออกมาเป็นลำไม่แผ่กระจายรอบตัวเหมือนแบบแรก เช่น แสงจากสปอร์ตไลต์ แสงจากไฟแฟลช ฯลฯ

ทางเลือก	New	-การเพิ่มชนิดของแสงแบบใหม่ในดรออิงปัจจุบัน
	Modify	-การเปลี่ยนชนิดของแสงในดรออิง
	Delete	-การลบแสงจากดรออิง
	Pick	-การเลือกแสงเพื่อแก้ไขหรือลบออกจากดรออิง

2.15.2 การสร้างฉากด้วย Scene

หลังจากกำหนดมุมมอง และจัดแสงแล้ว เราจะต้องเก็บข้อมูลภาพแสงของกล้องแต่ละตัว รวมทั้งแสงที่ใช้เรียกว่า Scene (ฉากในการถ่ายภาพ)

ทางเลือก	New	-เพิ่มฉากใหม่ในดรออิงปัจจุบัน
	Modify	-การเปลี่ยนฉาก โดยสามารถเปลี่ยนมุมมอง และการเพิ่มหรือลบ แสงออกจากฉาก
	Delete	-การลบฉากจากดรออิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.16 ออโตลิปส์(AutoLISP)

2.16.1 ทำไมถึงใช้ ออโตลิปส์

เราเลือก ออโตลิปส์ เป็น AutoCAD application Interface Language เนื่องมาจาก

1. ลิปส์ (LISP)เป็นภาษาในการโปรแกรม(Programming language) ที่ง่ายต่อการเรียนรู้และนำไปใช้งาน

2. ลิปส์ ถูกเลือกเพื่อใช้ในการวิจัยและพัฒนา ปัญญาประดิษฐ์ และ ระบบผู้เชี่ยวชาญ

3. มี Syntax ที่ง่าย และ ลิปส์ interpreter ง่ายต่อการนำไปใช้ และใช้หน่วยความจำ (Memory)น้อย

4. ตัวแปลภาษาลิปส์(LISP interpreter) เหมาะในการใช้กับ Unstructured interaction (ซึ่งเป็นลักษณะของ design process)

5. ลิปส์ เก่งในการทำงานกับเซท (set) ของวัตถุ ที่เหมือนๆกัน ซึ่งมีขนาดต่างๆ(ซึ่งจะพบในข้อมูลประเภทต่างๆของระบบแคด (CAD-system) เช่น AutoCAD manipulate)

การเขียนโปรแกรมภาษา ออโตลิปส์ นั้นจะทำภายใต้การทำงานของออโตแคด ถ้ามองกันแบบผิวเผินจะเป็นการสร้างและจัดฐานข้อมูลภายในโปรแกรมง่ายๆ

2.16.2 การใช้งาน ออโตลิปส์ ภายใน ออโตแคด

1. ใช้สร้างฟังก์ชัน (Function) ต่างๆไว้เรียกใช้
2. ใช้สร้างเมนูไฟล์ (Menu File)
3. ใช้สร้างสคลิปไฟล์
4. ใช้สร้างมาโคร (Macro) ของคอมมานด์ ต่างๆ
5. ใช้ในการออกแบบแบบของการระบาย (Hatch pattern)
6. ใช้ในการออกแบบแบบของเส้น (Line pattern)
7. ใช้ในการออกแบบแบบของตัวอักษร (Font pattern)

การใช้โปรแกรมออโตลิปส์เป็นการเพิ่มความสามารถในการทำงานของออโตแคดจริงๆ แล้วการเรียนสร้างรูปประโยคคำสั่งพื้นฐานของออโตลิปส์ก็เพื่อให้เราสามารถเรียกใช้ฟังก์ชันภายในนั้นมาเขียนโปรแกรมของเราเอง ประโยคคำสั่งของออโตลิปส์จะอยู่ในรูปมาตรฐานของนิพจน์ของมันเราอาจจะค้นหาฟังก์ชัน ออโตลิปส์ เพิ่มเติมได้จากคู่มือคำสั่งของออโตแคด ในหัวข้อของคำสั่งภาษาออโตลิปส์การเรียนรู้ฟังก์ชันใหม่ๆเพิ่มเติมจะเป็นการเพิ่มพูนประสิทธิภาพในงานการเขียนโปรแกรมของเราเอง

ภาษาออโตลิปส์จะมีความคล้ายคลึงกับภาษาคอมมอนท์ลิปส์ (commonLISP) ซึ่งเป็นโปรแกรมที่มีใช้งานกันมานานแล้วจนถึงทุกวันนี้ ภาษาออโตลิปส์เป็นส่วนหนึ่งของภาษาคอมมอนท์ลิปส์ ที่ได้มีการเพิ่มเติมและเปลี่ยนแปลงให้เข้าโปรแกรมออโตแคด

ข้อดีของภาษาลิปส์ คือ มีประโยคคำสั่งพื้นฐานที่ง่ายต่อการเรียนรู้และการเขียนโปรแกรม ดังนั้นจึงจัดได้ว่าภาษาออโตลิปส์ เป็นส่วนย่อย (subset) ของภาษา คอมมอนท์ลิปส์ และยังง่ายมากต่อการเรียนรู้

เอกสารนี้จัดทำขึ้นเพื่อสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.16.3 โครงสร้างของ ออโตลิปส

ออโตลิปส์เป็นภาษาคอมพิวเตอร์ที่เขียนได้ง่ายๆไม่ต้องมีการประกาศหรือกำหนดชนิดของตัวแปรที่ต้นโปรแกรม เช่นเดียวกับภาษาเบสิก (BASIC) ซึ่งต่างกับภาษาซีหรือปาสคาล(Pascal)ที่เข้มงวดเรื่องตัวแปรมาก จะต้องมีการประกาศใช้ตัวแปรก่อนจึงจะใช้ตัวแปรนั้นได้ ดังนั้นผู้ที่จะเขียนโปรแกรมออโตลิปส์ ไม่จำเป็นต้องเข้าใจเรื่องโครงสร้างและชนิดของตัวแปรมากนัก ก็สามารถเขียนโปรแกรมออโตลิปส์ ได้

ลักษณะการใช้งาน ออโตลิปส์ แบ่งเป็น 2 ลักษณะคือ

1. ไดเรกโหมด (Direct Mode) เป็นการติดต่อกับ ออโตลิปส์ โดยตรงขณะที่อยู่คอมมานด์พร้อมต์คล้ายภาษาเบสิค ที่เขียนโปรแกรมโดยไม่ได้กำหนดหมายเลขบรรทัด

2. โปรแกรมโหมด (Programmed Mode) เป็นการเขียนโปรแกรม ออโตลิปส์ ให้อยู่ในรูปของไฟล์ตัวอักษร ก่อนแล้วเรียกเข้ามาในหน่วยความจำ จากนั้นจึงสั่งรันอีกครั้งหนึ่ง จะเห็นว่า แบบไดเรกโหมด จะใช้ได้ครั้งเดียว แต่แบบโปรแกรมโหมดสามารถเรียกมาใช้ในภายหลังได้ นอกจากนั้นยังสามารถที่จะปรับปรุงแก้ไขโปรแกรมบางส่วนได้อีกด้วย การเก็บโปรแกรมอยู่ในรูปของไฟล์ตัวอักษรนี้ นามสกุลจะต้องเป็น .LSP เท่านั้น ส่วนการ เรียกโปรแกรมเข้ามาในหน่วยความจำใช้คำสั่ง LOAD

2.16.4 ส่วนประกอบของโปรแกรม

ในโปรแกรมออโตลิปส์ แต่ละโปรแกรมจะประกอบด้วยนิพจน์หลายๆนิพจน์ (expression) เขียนต่อกันไปเป็นบล็อก ความหมายของนิพจน์คือ การรวมเอาฟังก์ชันตัวแปร (variable) หรือค่าคงที่ (constant) มาสัมพันธ์กันให้ได้ความหมายหรือผลลัพธ์ใหม่ขึ้นมา

องค์ประกอบของโปรแกรม ออโตลิปส์

ในโปรแกรม ออโตลิปส์ ไม่ว่าจะเป็น ไดเรกโหมด หรือ โปรแกรมโหมดจะมีองค์ประกอบดังนี้

1. ค่าคงที่และตัวแปร (Constants and Variable)

ค่าคงที่คือ ค่าที่มีค่าคงที่ตลอดเวลาโดยไม่มีการเปลี่ยนแปลงขณะที่โปรแกรมกำลังดำเนินการประมวลผล ซึ่งค่าคงที่จะถูกกำหนดโดยผู้สร้างโปรแกรม เช่น 5,5.8 เป็นต้น

ตัวแปร คือ สัญลักษณ์ที่ผู้เขียนกำหนดขึ้น เพื่อใช้สำหรับเก็บข้อมูลในการประมวลผล ซึ่งตัวแปรเหล่านี้สามารถเปลี่ยนแปลงได้ตามคำสั่งในการประมวลผลของโปรแกรม

-กฎเกณฑ์การใช้ตัวแปร

การตั้งชื่อตัวแปรหรือการใช้ตัวแปรในออโตลิปส์ ค่อนข้างสะดวกคือ ไม่ต้องมีการประกาศชนิดตัวแปรไว้ที่ต้นโปรแกรมต่างกับภาษาปาสคาล หรือซี ที่ต้องประกาศไว้ก่อน มิฉะนั้นจะใช้ตัวแปรไม่ได้ อย่างไรก็ตามการตั้งชื่อตัวแปรใน ออโตลิปส์ จะมีกฎเกณฑ์ดังต่อไปนี้

1. อักษรแรกของชื่อตัวแปรจะเป็นตัวเลขหรืออักษร A ถึง Z ก็ได้ และ อักษรตัวใหญ่ (Upper case) หรืออักษรตัวเล็ก (Lower case) จะมีค่าเหมือนกัน เช่น เดียวกับภาษาปาสคาล แต่มีข้อแม้ว่าจะตั้งชื่อตัวแปรเป็นตัวเลขหมดไม่ได้ จะต้องมีอักษรร่วมอยู่ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ความยาวของชื่อตัวแปรได้ถึง 100 อักษรจะเป็นตัวเล็กหรือตัวใหญ่ก็ได้ ถ้าเกินจากนี้แล้วเครื่องจะแจ้งให้ทราบว่า "Bad Argument type" เกี่ยวกับความยาว ของชื่อตัวแปรควรจะทำให้สั้นที่สุดเท่าที่จะทำได้ เพราะจะเป็นการช่วยประหยัดเนื้อที่หน่วย ความจำในระบบด้วย

3. ห้ามใช้อักษรพิเศษดังต่อไปนี้ เป็นส่วนหนึ่งของชื่อตัวแปรโดยเด็ดขาดคือ .'";() ช่องว่างหรือเว้นวรรค *=><+-/?`!^{}หรือคีย์พิเศษอื่นๆ ยกเว้นเครื่องหมาย_ (ขีด) สามารถใช้ได้

4. ชื่อตัวแปรในออโตลิปส์ สามารถใช้ชื่อซ้ำกับชื่อตัวแปรในระบบใน ออโตแคด ได้เนื่องจากมีการแยกกันเป็นอิสระ

5. ห้ามใช้ชื่อตัวแปรซ้ำกับชื่อฟังก์ชันในออโตลิปส์ ทั้งที่เป็นฟังก์ชันมาตรฐานและฟังก์ชันที่ผู้ใช้กำหนดขึ้นเอง

-ชนิดของตัวแปร

ตัวแปรต่างๆนี้ไม่จำเป็นต้องกำหนดชนิดก่อนใช้งานขึ้นอยู่กับว่าเรานำค่าชนิดใดเข้าไปใส่ให้ตัวแปร ชนิดของตัวแปรมีดังต่อไปนี้

1.จำนวนเต็ม(Integer) คือข้อมูลตัวเลขที่มีค่าเป็นจำนวนเต็ม ทั้งบวกและลบที่ไม่มีทศนิยม ในระบบออโตแคด มีการใช้ค่าแบบจำนวนเต็มเก็บไว้เป็นตัวแปรระบบเพื่อแสดงสถานะปิด/เปิด เช่น ถ้า snapmode = 0 จะมีค่าเป็นปิด และมีค่าเป็น 1 จะกลายเป็นเปิดใช้ snap ค่าตัวเลขจำนวนเต็ม ที่จะเก็บไว้ในตัวแปร จะมีค่าอยู่ในช่วง -32768 ถึง +32768 การกำหนดตัวแปรเช่น

(Setq BC 15) หรือ (Setq TIME 289)

2. จำนวนจริง(Real)คือข้อมูลตัวเลขทั้งค่าบวกและค่าลบที่มีทศนิยมด้วย เช่น -5.3 ,4.8 ฯลฯ การกำหนดค่าตัวเลขแบบจำนวนจริง ให้กับตัวแปรในกรณีที่ค่าน้อยกว่า 1 เช่น 0.85 จะขึ้นต้นด้วยจุดทศนิยมไม่ได้ เช่น

(Setq AB .85)

เครื่องจะไม่รับและเตือนให้ทราบว่า "error:INVALID dotted pair" ทางที่ถูกคือต้องขึ้นต้นด้วยเลขศูนย์เป็น (Setq AB 0.85) รูปแบบที่สามารถกำหนดได้อีกลักษณะคือ แบบวิทยาศาสตร์ (Scientific)สำหรับตัวเลขที่มีค่ามากๆหรือน้อยๆ เช่น

(Setq A 1.4E+8)

(Setq B 2.5E-4)

3. รายการ(List) มีลักษณะคล้ายตัวแปรชุด หรือเป็นเซต โดยที่ตัวแปร 1 ตัวอาจมีข้อมูลอยู่หลายตัว และข้อมูลของตัวแปรรายการนี้ จะอยู่ภายใต้เครื่องหมายวงเล็บเปิด และ วงเล็บปิด

ใน 1 ชุดของตัวแปรแบบรายการ จะมีข้อมูลแบบใดอยู่ก็ได้ เช่น จำนวนเต็ม,จำนวนจริง,สตริงค์แต่มีข้อแม้ว่าจะอยู่ภายใต้เครื่องหมายวงเล็บเปิดและปิด ตัวอย่างของตัวแปรแบบรายการนี้ได้แก่ ตัวแปรที่แทนค่า โคออร์ดิเนต ซึ่งจะต้องมีค่า X,Y,Z อยู่ใน 1 ตัวแปร เช่น (25,10,25) จะมีลักษณะการกำหนดดังนี้

(Setq p1 (list 25 10 25)) จะได้ p1 = (25,10,25)

(Setq A2 (list "AB" 2)) จะได้ A2 = ("AB" 2)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. สตริงค์(Strings) เป็นตัวแปรที่เก็บอักขระต่างๆหรือข้อความ การกำหนดจะต้องเขียนข้อความภายใต้เครื่องหมายคำพูด "" (double quatotion mark) เช่น กำหนดให้ข้อความ TABLE เก็บไว้ในตัวแปร TXT1 และ CHAIR เก็บไว้ในตัวแปร TXT2

นอกจากนี้อักขระพิเศษต่างๆ เช่น <>,.{} สามารถนำไปเป็นส่วนหนึ่งของข้อมูลแบบ สตริงค์ ได้

5. ตัวบ่งชี้ไฟล์(File Descriptor) เป็นตัวแปรที่เก็บหมายเลขของไฟล์ไว้ เช่น <File: #50e0>เนื่องจากออโตลิปส์ มีฟังก์ชันที่เกี่ยวกับการอ่าน-เขียนไฟล์ ในกรณีที่ออโตลิปส์ต้องการที่จะติดต่อกับไฟล์จะใช้ตัวแปรตัวบ่งชี้ไฟลนี้เป็นตัวกำหนด ซึ่งต่างกับภาษาอื่นๆที่ใช้ชื่อนามสกุลไฟล์เป็นตัวกำหนด เช่น ต้องการเปิดไฟล์ชื่อ CHTR. TXT เพื่อทำการอ่าน (read) จะต้องกำหนดตัวแปรตัวบ่งชี้ไฟล์ก่อนดังนี้

(Setq Nofile (open "CHTR.TXT" "r"))

หลังจากที่กำหนดตัวแปรดังกล่าวแล้ว Nofile จะกลายเป็นตัวแปรแบบ ตัวบ่งชี้ไฟล์ ซึ่งใช้สำหรับอ้างถึงไฟล์ CHTR.TXT ในการอ่านต่อไป

6. ชื่อของส่วนประกอบ(Entity Names) เป็นตัวแปรที่ใช้เก็บชื่อของวัตถุหรือส่วนประกอบ (entity) ใน ออโตแคด ข้อมูลจะอยู่ในรูปของรหัส เช่น <Entity name: 60000A14> ซึ่งทุกๆวัตถุ หรือ ส่วนประกอบจะมีชื่อ(อยู่ในรูปรหัส) ประจำอยู่ เช่น ต้องการกำหนดให้ชื่อของส่วนประกอบ ที่ได้สร้างไว้ล่าสุดเก็บไว้ในตัวแปรชื่อ lent ทำได้ดังนี้

(Setq lent (entlast))

7. ชุดของตัวเลือก(Selection Sets) เป็นตัวแปรที่ใช้เก็บกลุ่มของวัตถุ ที่เลือกไว้ เช่น <Selection Set:1> ใน ออโตลิปส์ เราสามารถกำหนดให้ วัตถุหลายๆวัตถุเข้าไปอยู่ในตัวแปรแบบ ชุดของตัวเลือก หลังจากนั้นเมื่อมีการเรียกใช้งาน ตอนที่เครื่องถาม Selection object: เราสามารถตอบชื่อตัวแปรแบบชุดของตัวเลือก นี้ลงไปแทนการเลือกวัตถุทีละ วัตถุได้

-การนำค่าจากตัวแปรมาใช้งาน

ตัวแปรที่ได้กำหนดค่าไว้แล้วไม่ว่าจะเป็นจำนวนเต็ม,จำนวนจริง,สตริงค์หรือรายการก็ตาม ตัวแปรนั้นจะยังคงมีค่าเดิมอยู่จนกว่าจะมีการกำหนดค่าใหม่ลง หรือ QUIT ออกจากออโตแคดค่าตัวแปรต่างๆที่ได้กำหนดไว้สามารถเรียกออกมาใช้ได้โดยใช้เครื่องหมาย ! (exclamation point) นำหน้าชื่อตัวแปรดังนี้

!ชื่อตัวแปร

2. ฟังก์ชันและอาร์กิวเมนต์(Function and Arguments)

ฟังก์ชันเป็นเสมือนตัวดำเนินการค่าคงที่หรือตัวแปรเพื่อให้ได้ผลลัพธ์ใหม่ขึ้นมาตามวิธีดำเนินการ ได้แก่ ฟังก์ชัน +,-,*,/ ฯลฯ

อาร์กิวเมนต์ คือ ข้อมูลอาจจะเป็นค่าคงที่หรือตัวแปรที่จะส่งผ่านเข้าไปดำเนินการตามฟังก์ชันที่กำหนด เช่น

(/ 15 5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/ คือ ฟังก์ชัน

15 5 คือ อาร์กิวเมนต์

ในบางฟังก์ชันจำเป็นต้องมีอาร์กิวเมนต์บางฟังก์ชันอาจไม่จำเป็นต้องมีก็ได้ ฟังก์ชันแบ่งออกได้เป็น 2 ประเภทคือ

1. ฟังก์ชันมาตรฐานที่มีใน ออโตลิปส์ (Standard function)

-ฟังก์ชันคณิตศาสตร์

ในออโตลิปส์มีฟังก์ชันคณิตศาสตร์ไว้คำนวณหลายฟังก์ชันซึ่งมีรูปแบบการใช้งานดังนี้

(Function Arg1 Arg2)

Arg1,Arg2 คือ อาร์กิวเมนต์อาจจะเป็นค่าคงที่หรือตัวแปร หรือเป็นนิพจน์ก็ได้ที่ต้องการผ่านค่าเข้าไปปฏิบัติในฟังก์ชัน

-ฟังก์ชันมาตรฐานต่าง ๆที่สำคัญมีดังต่อไปนี้

ฟังก์ชันคณิตศาสตร์

ฟังก์ชัน	หน้าที่
(+ Arg1 Arg2...)	ทำการรวมผล Arg1 Arg2...
(- Arg1 Arg2...)	หาผลลบของ Arg1 ลบด้วย Arg2...
(Arg1 Arg2...)*	หาผลคูณ Arg1 คูณด้วย Arg2...
(/ Arg1 Arg2...)	หาผลหาร Arg1 หารด้วย Arg2...
(max Arg1 Arg2...)	หาค่าสูงสุดของอาร์กิวเมนต์ที่ระบุ
(min Arg1 Arg2...)	หาค่าต่ำสุดของอาร์กิวเมนต์ที่ระบุ
(rem Arg1 Arg2...)	หาค่าเศษเหลือของผลหาร Arg1 หารด้วย Arg2
(1+ Arg)	หาค่า Arg บวก 1
(1- Arg)	หาค่า Arg ลบ 1
(abs Arg)	หาค่าแอบโซลูตของ Arg (ค่าบวก)
(exp n)	หาค่า e ยกกำลัง n (e^n)
(expt Arg n)	หาค่า Arg ยกกำลัง n (Arg^n)
(fix real)	เปลี่ยนจากจำนวนจริงให้เป็นจำนวนเต็ม
(float integer)	เปลี่ยนจากจำนวนเต็มให้เป็นจำนวนจริง
(log Arg)	หาค่า log ฐาน e ของ Arg (loge Arg)
(sqrt Arg)	หาค่ารากที่ 2 (square root) ของ Arg
(gcd Arg1 Arg2)	หาค่า ตัวหารร่วมมากของ Arg1 และ Arg2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันที่กระทำทางบิต

ฟังก์ชัน	หน้าที่
(logand int1 int2...)	ทำการ and ทางบิตของ int1 และ int2...
(logior int1 int2...)	ทำการ or ทางบิตของ int1 int2...
(lsh int bits)	ทำการเลื่อนบิตทุกๆบิตของ int ไปทางซ้ายจำนวน bits
(~int)	หาค่า logic not ของ int

ฟังก์ชันตรีโกณมิติ

ฟังก์ชัน	หน้าที่
(sin arg)	หาค่า sine ของมุม arg หน่วย radian
(cos arg)	หาค่า cosine ของมุม arg หน่วย radian
(atan arg1 arg2)	หาค่า arctan ของ arg1 หรือ arg1/arg2
PI	ให้ค่า PI

-ฟังก์ชันที่ใช้กับข้อมูลแบบรายการ

สำหรับตัวแปรแบบ list หรือตัวแปรแบบชุด เช่น พวก โคออร์ดิเนต จะมีค่า X, Y,Z รวมอยู่ด้วยเป็น 1 ชุด เราสามารถใช้ฟังก์ชันใน ออโตลิปส์ แยกเอาค่า X,Y หรือ Z ออกมาจากตัวแปรหรือข้อมูลแบบรายการได้

ในข้อมูลแบบรายการจะมีข้อมูลหลายๆตัวรวมอยู่เป็นชุดข้อมูลแต่ละตัวเรียกว่า element แต่ละ element ก็จะมีตำแหน่งเรียก โดยนับเริ่มตั้งแต่ ตำแหน่งที่ 0 ตำแหน่งที่ 1 ตำแหน่งที่2 ไปเรื่อยๆจนกว่าจะหมดข้อมูลดังนี้

(1 "ABC" 2 7 9)

1	element 0
"ABC"	element 1
2	element 2
7	element 3
9	element 4

ฟังก์ชันเกี่ยวกับข้อมูลแบบรายการมีดังนี้

ฟังก์ชัน	หน้าที่
(car list)	นำค่า element ที่ 0 ออกมาจากรายการ
(cdr list)	ตัดค่า element ที่ 0 ทิ้งไป
(cadr list)	นำค่า element ที่ 1 ออกมา
(cddr list)	ตัดค่า element ที่ 0 และ 1 ทิ้งไป
(caddr list)	นำค่า element ที่ 2 ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน	หน้าที่
(cdddr list)	ตัดค่า element ที่ 0,1 และ 2 ทิ้งไป
(cadddr list)	นำค่า element ที่ 3 ออกมา
(cddddr list)	ตัดค่า element ที่ 0,1,2 และ 3 ทิ้งไป
(nth n list)	นำค่า element ที่ n ออกมาจากรายการ
(last list)	นำค่า element สุดท้ายออกมาจากรายการ
(reverse list)	สลับตำแหน่ง element ทั้งหมด ลักษณะพลิกกลับ
(length list)	นับจำนวน element ในรายการ
(append list1 list2)	นำข้อมูลใน list2 ไปต่อท้าย list1 ให้เป็น list 1 ชุด
(cons list1 list2)	นำข้อมูล list1 ไปเป็น element ที่ 0 รวมกับ list2
(assoc item alist)	พิจารณาค่าของ alist โดยจะค้นหารายการที่มีองค์ประกอบ ตัวแรกเป็นค่า item ที่กำหนด แล้วให้ผลลัพธ์เป็นรายการที่บรรจุค่า item นั้น
(foreach name list expression)	เป็นฟังก์ชันที่ทำการอ่านองค์ประกอบจากรายการหนึ่งและส่งพวกมันไปแบบหนึ่งต่อหนึ่งให้กับตัวแปรตัวหนึ่งซึ่งถูกนำไปใช้ในรูปแบบของนิพจน์ นิพจน์จะถูกประเมินผลซ้ำ ๆ กันโดยในแต่ละครั้งจะแทนค่าตัวแปรด้วยองค์ประกอบทีละตัวของรายการ

-ฟังก์ชันอินพุต(Input) เอาต์พุต(Output)ข้อมูล

ฟังก์ชันสำหรับอินพุต

สามารถรับข้อมูลได้ทั้งอุปกรณ์ชี้ (pointing device) และ แป้นพิมพ์ (keyboard) ซึ่งมีฟังก์ชันต่าง ๆดังต่อไปนี้

ฟังก์ชัน **getint**

รูปแบบ *(getint promptstring)*

เป็นฟังก์ชันที่ใช้สำหรับรอรับค่าจำนวนเต็มจากแป้นพิมพ์ โดยมีข้อความปรากฏบนจอที่ command line ร่วมกับการรอรับค่า คือ promptstring และ จะรอจนกว่าผู้ใช้กด Enter หรือแป้นเคาะ (space bar) ถือว่าจบการทำงานฟังก์ชัน getint

สำหรับ promptstring จะมีหรือไม่มีก็ได้ แต่ถ้ามีจะต้องอยู่ในเครื่องหมายคำพูด(" ") เสมอ

ฟังก์ชัน **getreal**

รูปแบบ *(getreal promptstring)*

เป็นฟังก์ชันที่สามารถรับค่าได้ทั้งจำนวนเต็มและจำนวนจริง แต่ค่าที่ได้หลังจากจบฟังก์ชันแล้วจะเป็นจำนวนจริง เช่น เราใส่เลข 3 ลงไป เมื่อจบฟังก์ชันแล้วเครื่องจะจัดให้เป็นจำนวนจริง โดยเติมทศนิยมให้กลายเป็น 3.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน **getstring**

รูปแบบ *(getstring T promptstring)*

เป็นฟังก์ชันใช้สำหรับรอรับข้อมูลอักษรโดยมีข้อความปรากฏขณะรอรับข้อมูลคือ promptstring ส่วนตัวเลือก T จะมีหรือไม่มีก็ได้ ถ้าไม่มี การใส่ข้อมูลอักษรเข้าไป หากกดแป้นเว้นวรรคจะกลายเป็นการจบการป้อนข้อมูล หรืออีกนัยหนึ่ง คือเป็นการใช้แป้นเคาะแทน Enter ได้นั่นเอง แต่ถ้ามี การป้อนข้อมูลเมื่อเสร็จแล้วจะต้องกด Enter ข้อมูลนั้นถึงจะเข้าเครื่อง

สรุป มีตัวเลือก T ต้องกด Enter เมื่อป้อนข้อมูลเสร็จ

ไม่มีตัวเลือก T กด Enter หรือ แป้นเคาะก็ได้เมื่อป้อนข้อมูลเสร็จ

ฟังก์ชัน **getkword**

รูปแบบ *(getkword promptstring)*

เป็นฟังก์ชันใช้สำหรับรับข้อมูลที่เป็นรหัส (keyword) โดยปกติแล้วขณะที่ใช้ออโตแคดมักจะได้เห็นการใช้ getkword บ่อย ๆ เช่นคำสั่ง ARC

สำหรับการใช้ getkword จะต้องใช้ร่วมกับฟังก์ชัน initget เพื่อกำหนดอักษรที่จะเป็น keyword เสียก่อน โดยอักษรที่จะเป็น keyword จะต้องใช้อักษรตัวใหญ่เสมอ

ฟังก์ชัน **getangle**

รูปแบบ *(getangle basepoint promptstring)*

เป็นฟังก์ชันสำหรับรับค่ามุมหน่วยเป็นองศาจากแป้นพิมพ์ หรือจะใช้อุปกรณ์ชี้เป็นตัวป้อนค่ามุมก็ได้ หลังจากที่ผ่านค่าองศาให้กับฟังก์ชันแล้ว ค่าที่ได้จะมีหน่วยเป็นเรเดียน (radian) อัตโนมัติ

สำหรับ basepoint เป็นจุดอ้างอิงที่จะวัดมุม เราจะกำหนดหรือไม่กำหนดก็ได้ ถ้ากำหนดก็จะมีผลเฉพาะการป้อนค่ามุมด้วยอุปกรณ์ชี้เท่านั้น

ฟังก์ชัน **getorient**

รูปแบบ *(getorient basepoint promptstring)* เป็นฟังก์ชันที่มีรูปแบบการใช้ระบุ

เป็นฟังก์ชันที่มีรูปแบบการใช้งานเหมือน getangle ทุกประการ แตกต่างกันที่ getangle จะอ้างอิงตาม ระบบมุมที่เซตไว้ด้วยคำสั่ง UNITS แต่ getorient จะอ้างอิงตามระบบมาตรฐานเท่านั้น

ฟังก์ชัน **getdist**

รูปแบบ *(getdist basepoint promptstring)*

เป็นฟังก์ชันที่ใช้สำหรับรอรับค่าระยะห่าง (distance) โดยการใช้แป้นพิมพ์ป้อนค่าเป็นตัวเลขลงไป หรือใช้อุปกรณ์ชี้เป็นตัวกำหนดก็ได้ สำหรับ basepoint เป็นจุดอ้างอิง เช่นเดียวกับ getangle จะกำหนดหรือไม่กำหนดก็ได้ ถ้ามีการกำหนด basepoint และ ใช้อุปกรณ์ชี้เป็นตัวกำหนดระยะค่าที่ได้คือ ระยะห่างจากจุด basepoint ถึงจุดที่เคอร์ เซอร์อยู่ แต่ถ้าไม่ได้กำหนด basepoint เครื่องจะใช้จุดที่กำหนดครั้งแรกเป็น basepoint แทน

ฟังก์ชัน **getpoint**

รูปแบบ *(getpoint basepoint promptstring)*

เป็นฟังก์ชันที่ใช้สำหรับรอรับค่า โคออร์ดิเนต โดยป้อนผ่านแป้นพิมพ์หรือใช้อุปกรณ์ชี้ก็ได้ หากป้อนผ่านแป้นพิมพ์จะต้องใช้เครื่องหมาย,คั่นระหว่างค่า X,Y และ Z ด้วยและ การป้อนค่าจะต้อง

เอกสารนี้เป็นเอ... ไม่ว่า...

ป้อนอย่างน้อย X และ Y หากไม่ใส่ค่า Z ลงไป ออโตลิปส์ จะถือว่าค่า Z เป็นศูนย์นั่นเอง หากใช้อุปกรณ์ชี้กำหนด โคออร์ดิเนต ที่ได้คือตำแหน่งเคอร์เซอร์บนจอขณะ pick point นั่นเอง

ส่วน basepoint จะกำหนดหรือไม่กำหนดก็ได้ ถ้ากำหนดจะมีเส้นใยเกิดขึ้นระหว่าง basepoint ถึงตำแหน่งเคอร์เซอร์ปัจจุบัน ค่าที่ได้จาก getpoint จะมีค่าเป็น โคออร์ดิเนต X,Y,Z (มีข้อมูลเป็นแบบ list)

ฟังก์ชัน **getcorner**

รูปแบบ *(getcorner basepoint prompstring)*

เป็นฟังก์ชันที่ใช้สำหรับรอรับค่า โคออร์ดิเนต คล้ายกับ getpoint แต่ getcorner จะต้องมีการกำหนด basepoint มิฉะนั้นจะเกิด error ขึ้น

เมื่อฟังก์ชัน getcorner ทำงานจะมีกรอบวินโดว์เกิดขึ้นคล้ายกับการ select object: เป็นแบบวินโดว์โดยที่ basepoint จะเป็นมุมแรก และตำแหน่งปัจจุบันของเคอร์เซอร์จะเป็นมุมที่เหลือ โดยปกติมักจะใช้ฟังก์ชัน · getpoint ก่อนเพื่อกำหนดโคออร์ดิเนต ให้เป็น basepoint จากนั้นจึงจะใช้ฟังก์ชัน getcorner กำหนดมุมที่เหลือ

-การใช้ **Initget** ควบคุมการรับข้อมูล

ขณะที่ใช้ฟังก์ชันรอรับข้อมูล หากผู้ใช้กด Enter ผ่านไปโดยไม่ใส่ข้อมูลใดๆ โปรแกรมจะถือว่ามีค่าเป็น nil คือไม่มีค่านั่นเอง ลักษณะนี้บางครั้งอาจส่งผลให้โปรแกรม error เช่น รอรับค่าเข้าไปเป็นตัวหาร เมื่อไม่มีค่าก็จะเกิด error ขึ้น เป็นต้น

เราสามารถใช้ฟังก์ชัน initget ควบคู่กับการใช้ฟังก์ชันอินพุตข้อมูล เพื่อควบคุมลักษณะของข้อมูลที่จะอินพุตเข้ามาได้ เช่น ข้อมูลที่เข้ามาจะต้องไม่เป็นศูนย์ หากผู้ใช้ใส่ค่าเป็นศูนย์มาให้กลับไปรอรับข้อมูลใหม่ เป็นต้น

รูปแบบ *(initget code stringinput)*

code เป็นตัวควบคุมลักษณะการอินพุตข้อมูล ส่วน stringinput ใช้สำหรับกำหนดอักษรที่จะเป็น keyword ซึ่งในฟังก์ชัน initget จะกำหนดอาร์กิวเมนต์ตัวใดตัวหนึ่ง คือ code หรือ stringinput หรือจะกำหนดทั้ง 2 ตัวก็ได้

ตารางที่2.6 แสดงรหัส (code) และความหมายของการควบคุม

รหัส	ความหมาย
1	ไม่รับข้อมูลที่ไม่มีค่า (nil)
2	ไม่รับข้อมูลที่มีค่าเป็นศูนย์
4	ไม่รับข้อมูลที่มีค่าเป็นลบ
8	ไม่ต้องเช็ค Limits หากมีการใช้ Limits on
16	ส่งผ่านค่าโคออร์ดิเนต แบบ 3 มิติ
32	เส้นใยของการใช้ getdist getpoint และกรอบวินโดว์ของ getcorner จะเป็นเส้นประแทนเส้นเต็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่ต้องการควบคุมหลายลักษณะให้นำค่าของรหัส มาบวกกันแล้วเป็นรหัสใหม่ได้ เช่น ต้องการควบคุมไม่ให้รับข้อมูลที่เป็นศูนย์และค่าลบ ให้นำค่า 2+4 กลายเป็น รหัสใหม่คือ 6

สำหรับรหัสใน initget จะควบคุมฟังก์ชันอินพุตข้อมูลได้เป็นบางรายการเท่านั้น ดังแสดงในตารางที่2.7

ตารางที่ 2.7 แสดงความสัมพันธ์ของฟังก์ชัน และ รหัสที่มีผลในการควบคุม

ฟังก์ชัน	รหัส
getint	1,2,4
getreal	1,2,4
getdist	1,2,4,16,32
getangle	1,2,32
getorient	1,2,32
getpoint	1,8,16,32
getcorner	1,8,16,32
getkword	1
getstring	ไม่มี

ฟังก์ชันสำหรับเอาต์พุต

ฟังก์ชัน	หน้าที่
Prin1	เอาต์พุตข้อมูลลงบริเวณ command line หรือไฟล์ที่กำหนด
PrinC	เช่นเดียวกับ Prin1 แต่จะปฏิบัติตามรหัสควบคุมด้วย
Print	เช่นเดียวกับ Prin1 แต่จะขึ้นบรรทัดใหม่ให้
Prompt	เขียนข้อมูลลงบริเวณcommandlineอย่างเดียวและปฏิบัติตามรหัสควบคุม

ตารางที่2.8 แสดงรหัสควบคุมและความหมาย

รหัส	ความหมาย
\n	ให้ขึ้นบรรทัดใหม่ก่อนพิมพ์ข้อมูล (new line)
\t	ให้เว้นระยะ tab
\e	ให้ความหมายเป็น escape
\r	ให้ความหมาย return
\007	กำเนิดเสียง beep
\nnn	รหัสควบคุมตามเลขฐาน 8 nnn

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน **Print 1 ,PrinC,Print**

รูปแบบ *(Prin1 variable/string/Expression filedescriptor)*

ถ้าไม่กำหนด ตัวบ่งชี้ไฟล์จะถือว่าเป็นการเอาต์พุตข้อมูลบริเวณ commnad line ส่วน variable/string/Expression คือ ข้อมูลที่จะเอาต์พุตออกไปจะเป็นข้อมูลแบบใดก็ได้ แต่ถ้าเป็นแบบสตริงค์จะต้องอยู่ในเครื่องหมายคำพูด

ฟังก์ชัน **Prompt**

รูปแบบ *Prompt variable/string*

prompt เป็นการเขียนข้อมูลลงบริเวณ command line คล้ายกับ Prin1,PrinC, Print แต่อาร์กิวเมนต์จะต้องเป็นตัวแปรที่กำหนดค่าไว้แล้วหรือเป็นสตริงค์เท่านั้น จะเป็นนิพจน์เช่น (Prompt (* 2 3)) ไม่ได้ และไม่สามารถให้เอาต์พุตลงไฟล์ได้ แต่สามารถปฏิบัติรหัสควบคุมได้ โดยปกตินิยมใช้สำหรับเขียนข้อความพร้อมต์เพื่อให้คอมพิวเตอร์ติดต่อกับผู้ใช้งาน

ฟังก์ชัน Reading และ writing แบบพิเศษ

จากฟังก์ชันอินพุต,เอาต์พุตข้อมูลที่ผ่านมา แต่ละฟังก์ชันจะมีหน้าที่ไปคนละอย่าง เช่น getreal รับค่าจำนวนจริง getpoint รับค่า โคออร์ดิเนต ขณะที่ใช้ getreal เราจะใช้อุปกรณ์ชี้บอกตำแหน่ง โคออร์ดิเนต ไม่ได้ แต่ ออโตลิปส์ มีฟังก์ชันพิเศษอีกฟังก์ชันหนึ่ง คือ grread ซึ่งสามารถรับค่าอินพุตได้จากอุปกรณ์ทุกชนิด

ฟังก์ชัน **grread**

รูปแบบ *(grread dynamic)*

โดยที่ไดนามิกจะกำหนดเป็น T สำหรับการอ่านค่าแบบไดนามิก หรือจะไม่กำหนดก็ได้

ตารางที่2.9 แสดงชนิดของการอินพุตที่ได้จาก grread

ค่าที่ได้	ความหมาย
(2 ASCII)	เป็นค่าที่ได้จากการกดคีย์ที่แป้นพิมพ์ โดยจะได้รหัส ASCII ของคีย์ที่กด
(3 coordinate)	ค่าที่ได้จากการอ่านตำแหน่ง coodinate เมื่อชี้ด้วยอุปกรณ์หรือแป้นพิมพ์
(4 menu_cell)	ค่าที่ได้จากการเลือกตำแหน่งเมนูบน สกรีนเมนู โดยตำแหน่งเมนูบนสุด คือ ตำแหน่งศูนย์
(5 coordinate)	อ่านค่าโคออร์ดิเนตของเคอร์เซอร์ทุกขณะที่เลื่อนไป (dynamic) ได้จากการใช้ฟังก์ชัน (grread T)
(6 button)	เป็นค่าที่ได้จากการกดปุ่ม botton ที่อุปกรณ์ชี้โดยที่ botton คือ ตำแหน่งของปุ่มที่ถูกกด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.9 (ต่อ)

ค่าที่ได้	ความหมาย
(7-10 Tablet menu)	เป็นค่าที่อ่านได้จากตำแหน่งของ เทปเบล็ตเมนูโดยที่ 7 คือเทปเบล็ตเมนูกลุ่ม 1 8 คือเทปเบล็ตเมนูกลุ่มที่ 2 ไปเรื่อยๆจนถึง 10
(11 Aux)	เป็นค่าที่ได้จากการเลือกตำแหน่งที่อุปกรณ์ Aux

ฟังก์ชัน **grtext**

รูปแบบ *(grtext location string)*

ทำหน้าที่คล้ายๆกับ prin1,print หรือ prompt ที่ได้กล่าวมาแล้ว แต่ขีดความสามารถจะสูงกว่าคือ เขียนข้อความที่ตำแหน่งต่างๆได้ เช่น โคออร์ดิเนต readout (บริเวณแสดง โคออร์ดิเนต) status line (บริเวณแสดง layer) หรือตำแหน่งบนสกรีนเมนู โดยการระบุ location เป็นรหัสเลขจำนวนเต็ม ถ้าต้องการแสดงบริ เวณ status line ใช้ location เป็น -1 บริเวณ โคออร์ดิเนต readout มีค่า โคออร์ดิเนต เป็น -2 ส่วนสกรีนเมนู จะเริ่มจากศูนย์เป็นต้นไป จนถึง 20

การใช้ **Distance** วัดระยะทาง

distance เป็นฟังก์ชันที่ใช้สำหรับวัดระยะทางจากจุด 2 จุด คล้ายกับคำสั่ง DIST ในออโตแคด มีรูปแบบดังนี้

(distance coordinate1 coordinate2)

หน่วยที่ได้จะขึ้นอยู่กับ units ที่ได้ตั้งไว้

การใช้ **Polar** กำหนดจุด

polar เป็นฟังก์ชันที่ใช้กำหนดจุดแบบสัมพัทธ์ (relative point) ซึ่งเป็นการกำหนดจุดใหม่โดยการอ้างอิงจุดร่วมกับมุมและระยะ มีรูปแบบดังนี้

(Polar point angle distance)

โดยที่ point คือ จุดที่ใช้เป็นจุดอ้างอิง

angle คือ มุมที่จุดใหม่กระทำต่อจุดอ้างอิงหน่วยเป็น radian

distance คือ ระยะห่างระหว่างจุดอ้างอิงและจุดที่จะกำหนดใหม่

การใช้ **Angle** วัดค่ามุม

ฟังก์ชัน angle เป็นฟังก์ชันที่ใช้สำหรับวัดค่ามุม โดยกำหนดจุด 2 จุด มีรูปแบบดังนี้

(angle coordinate1 coordinate2)

หน่วยที่ได้จะเป็นเรเดียนลักษณะการวัดมุมจะอ้างอิงจากวงกลมในหน่วยมุมเรเดียน

-ฟังก์ชันเปลี่ยนแปลงชนิดข้อมูล

ฟังก์ชันเปลี่ยนแปลงชนิดข้อมูลใน ออโตลิปส์ ได้แก่ การเปลี่ยนจากจำนวนจริงไปเป็นสตริงค์ เปลี่ยนจากค่ามุมเป็นสตริงค์เป็นต้น ซึ่งมีฟังก์ชันต่างๆดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน **Atoi**

รูปแบบ *(Atoi string)*

ทำหน้าที่ เปลี่ยนข้อมูลจากชนิดสตริงค์เป็นจำนวนเต็มซึ่งจะเปลี่ยนได้เฉพาะข้อมูล สตริงค์ที่มีส่วนประกอบเป็นตัวเลขเท่านั้น วัตถุประสงค์ที่เปลี่ยนเพื่อที่จะให้มีการคำนวณได้ เนื่องจากตัวเลขที่อยู่ในรูปของสตริงค์ไม่สามารถนำไปคำนวณได้ จะต้องทำการเปลี่ยนเสียก่อน

ฟังก์ชัน **Atof**

รูปแบบ *(Atof string)*

มีลักษณะการใช้งานคล้ายกับ Atoi แต่จะเปลี่ยนข้อมูล สตริงค์ไปเป็นแบบจำนวนจริง แทน

ฟังก์ชัน **Ascii**

รูปแบบ *(Ascii char)*

ทำหน้าที่เปลี่ยนจากอักษร char ให้เป็นรหัส ASCII

ฟังก์ชัน **Angtos**

รูปแบบ *(Angtos real)*

ทำหน้าที่ เปลี่ยนค่าของมุมในหน่วยเรเดียนให้เป็นสตริงค์ที่มีค่าเป็นองศา โดยอัตโนมัติ ซึ่งโดยปกติหากใช้ฟังก์ชัน getangle รับค่ามุมเข้าไป ถึงแม้เราจะใส่หน่วยเป็นองศาก็ตาม เมื่อเข้าระบบแล้ว ออโตลิปส์ จะเปลี่ยนเป็นหน่วยเรเดียนทันที

ฟังก์ชัน **Chr**

รูปแบบ *(chr integer)*

ทำหน้าที่ เปลี่ยนจากเลขจำนวนเต็ม (หรือรหัส ASCII) ให้เป็นอักษรตรงข้ามกับฟังก์ชัน ASCII

ฟังก์ชัน **Itoa**

รูปแบบ *(Itoa integer)*

ทำหน้าที่ เปลี่ยนเลขจำนวนเต็มไปเป็นสตริงค์ซึ่งตรงข้ามกับฟังก์ชัน Atoi

ฟังก์ชัน **Rtos**

รูปแบบ *(Rtos real/integer unitcode precision)*

ทำหน้าที่ เปลี่ยนข้อมูลจากชนิดจำนวนเต็มหรือจำนวนจริง ให้เป็นชนิดสตริงค์โดยหน่วยที่แสดงกำหนดโดย unitcode และจำนวนตำแหน่งทศนิยมที่แสดงกำหนดโดย precision ซึ่ง unitcode และ precision อาจไม่กำหนดก็ได้ หากไม่กำหนด ออโตลิปส์ จะถือว่าให้แสดงตามรูปแบบที่กำหนดไว้ตามคำสั่ง UNITS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่2.10 แสดง unitcode และรูปแบบที่ใช้ใน rtos

รหัส	รูปแบบที่ได้	ตัวอย่าง
1	Scientific	4.5800E+01
2	Decimal	45.8000
3	Feet and decimal inches	3' -9.80000"
4	Feet and inches	3' -9 13/16"
5	Fractional units	45 13/16

ฟังก์ชัน **Fix**

รูปแบบ *(Fix real)*

เป็นฟังก์ชันที่ทำหน้าที่เปลี่ยนจากเลขจำนวนจริง ให้เป็นชนิดจำนวนเต็ม

ฟังก์ชัน **Float**

รูปแบบ *(Float real)*

มีลักษณะการใช้งานเช่นเดียวกับ Fix แต่จะเปลี่ยนจากข้อมูลชนิด จำนวนเต็มให้เป็นจำนวนจริง

-ฟังก์ชันเกี่ยวกับ String

ออโตลิปส์มีฟังก์ชันเกี่ยวกับการรวมข้อความหลายๆ ข้อความให้เป็นข้อความเดียว การตัดข้อความ การเปลี่ยนชนิดของข้อมูลจากจำนวนจริงให้เป็นชนิด สตริงค์และอื่นๆอีกหลายฟังก์ชัน

ฟังก์ชันเกี่ยวกับสตริงค์มีหน้าที่จัดการข้อมูลแบบ สตริงค์(หรือที่เรียกว่าข้อความนั่นเอง) เช่น รวมข้อมูล ตัดแยกข้อมูล เป็นต้น

ฟังก์ชันเกี่ยวกับ สตริงค์มีดังนี้

ฟังก์ชัน **Strlen**

รูปแบบ *(Strlen str)*

ทำหน้าที่ นับจำนวนอักษรในข้อมูล str ซึ่งจะต้องเป็นแบบ สตริงค์เท่านั้น อาจจะเป็นตัวแปรหรือค่าคงที่ก็ได้

ฟังก์ชัน **Substr**

รูปแบบ *(Substr str start length)*

ทำหน้าที่ แยกข้อความจาก str ซึ่งเป็นแบบสตริงค์โดยเริ่มที่ตำแหน่ง start ความยาวที่นำออกมาจำนวนเท่ากับ length

ฟังก์ชัน **Strcat**

รูปแบบ *(Strcat str1 str2)*

ทำหน้าที่ รวมข้อความ str1 str2 เข้าเป็น string เดียวกัน ฟังก์ชันนี้ค่อนข้างใช้บ่อย เนื่องจากฟังก์ชัน output เช่น princ จะเอาต์พุทข้อมูลได้ครั้งละ 1 ข้อมูลเท่านั้น หากต้องการใช้ princ output ข้อมูลหลายๆข้อมูลพร้อมๆกัน จะต้องทำการรวมให้เป็น 1 ข้อมูลเสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน **Strcase**

รูปแบบ *(Strcase str code)*

ทำหน้าที่ เปลี่ยนอักษร str ให้เป็นอักษรตัวเล็ก ถ้ารหัส (code) เป็น T และ เปลี่ยนให้เป็นอักษรใหญ่ทั้งหมดถ้ารหัสเป็น F ในกรณีที่ใช้รหัสเป็น F เราอาจจะไม่กำหนดตัว F ลงไปก็ได้ จะให้ผลลัพธ์เช่นเดียวกัน

-ฟังก์ชันเกี่ยวกับไฟล์

ฟังก์ชัน **Open**

รูปแบบ *(open filename code)*

เป็นฟังก์ชันที่ทำหน้าที่เปิดไฟล์ filename สำหรับอ่าน/เขียนตามที่กำหนดไว้ใน รหัส

ตารางที่2.11 แสดงรหัสสำหรับฟังก์ชัน open

รหัส	ความหมาย	หน้าที่
"r"	read	เปิดไฟล์สำหรับอ่านข้อมูลอย่างเดียวเท่านั้น
"w"	write	เปิดไฟล์สำหรับเขียนข้อมูลในกรณีที่มีไฟล์อยู่แล้ว จะทำการเขียนทับ แต่ถ้ายังไม่มีไฟล์อยู่จะเป็นการสร้างไฟล์ใหม่ สำหรับเขียนข้อมูล
"a"	append	เปิดไฟล์สำหรับเพิ่มเติมข้อมูลที่มีอยู่แล้วในไฟล์ ในกรณีที่ยังไม่มีไฟล์อยู่ก็จะเป็นการสร้างไฟล์ใหม่

ข้อควรระวัง จากการใช้งานพบว่า รหัสที่จะกำหนดให้อ่าน/เขียนไฟล์นี้ จะต้องใช้อักษรเล็กเท่านั้น ห้ามใช้อักษรใหญ่เด็ดขาด

ฟังก์ชัน **Read-line**

รูปแบบ *(read-line file_descriptor)*

เป็นฟังก์ชันที่ใช้สำหรับอ่านข้อมูลจากไฟล์ file_descriptor มา 1 บรรทัด หากใช้ read-line ภายหลังจากที่เพิ่งกำหนด open เป็นครั้งแรก ออโตลิปส์ จะทำการ อ่านข้อมูลจากบรรทัดแรกของไฟล์ จากนั้นจะไปรออยู่บรรทัดถัดไปเพื่อรอการใช้ read-line ต่อ ลักษณะการทำงานจะเป็นเช่นนี้ไปเรื่อยๆจนกว่าจะหมดไฟล์หรือสั่งปิดไฟล์

ฟังก์ชัน **write-line**

รูปแบบ *(write-line string file_descriptor)*

เป็นฟังก์ชันที่ใช้สำหรับเขียนข้อมูลลงไฟล์ file_descriptor ลักษณะการเขียนจะเริ่มเขียนจากบรรทัดแรกของไฟล์ไปเรื่อยๆจนกว่าจะมีการปิดไฟล์

ฟังก์ชัน **Read-char**

รูปแบบ *(Read-char file_descriptor)*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นฟังก์ชันที่ใช้อ่านตัวอักขระทีละตัวจากคีย์บอร์ด แต่ถ้ามีอาร์กิวเมนต์ของตัวบ่งชี้ไฟล์ แล้วมันจะอ่านตัวอักขระจากไฟล์ที่บ่งชี้ไว้ ค่าของผลลัพธ์จะอยู่ในรูปของรหัสแอสกี

ฟังก์ชัน **Write-char**

รูปแบบ *(Write-char integer file_descriptor)*

เขียนอักขระทีละหนึ่งตัวออกมาที่พร็อมต์บนหน้าจอหรือไฟล์ที่เปิดไว้ อาร์กิวเมนต์ของตัวอักขระ คือตัวเลขที่แสดงรหัสของตัวอักขระแอสกี

ฟังก์ชัน Close

รูปแบบ *(close file_descriptor)*

เป็นฟังก์ชันที่ทำหน้าที่ปิดไฟล์ที่เปิดด้วย open เป็นการสิ้นสุดการอ่าน/เขียนข้อมูลในไฟล์ ทุกๆครั้งที่มีการเปิดไฟล์ จะต้องมีการปิดไฟล์เมื่อเลิกใช้ด้วย มิฉะนั้นข้อมูลในไฟล์ที่ถูกเปิดนั้นอาจสูญหายไปได้

-ฟังก์ชันที่ใช้ในการตัดสินใจ

ฟังก์ชัน **If**

รูปแบบ *(If (condition) (then_expression)*
(else_expression)
)

ลักษณะการทำงานของฟังก์ชันนี้ คือ เมื่อโปรแกรมดำเนินงานมาถึงฟังก์ชันนี้ ก็จะทำการทดสอบเงื่อนไขใน condition ถ้าเป็นจริง (T) ก็จะไปทำใน then_expression ต่อ แต่ถ้าเป็นเท็จ (nil) ก็จะกระโดดข้ามไปทำใน else_expression

สำหรับบางกรณีอาจไม่จำเป็นต้องมี else_expression ก็ได้ หากการทดสอบเงื่อนไขใน condition เป็นจริง (T) ก็จะทำงานใน then_expression แต่ถ้าเป็นเท็จ (nil) จะกระโดดข้ามไปทำนิพจน์ในบรรทัดต่อไป

การใช้ฟังก์ชัน Progn รวมหลาย ๆนิพจน์เข้าด้วยกัน

จากการใช้งานฟังก์ชัน If จะเห็นว่าเมื่อเปรียบเทียบเงื่อนไขแล้ว จะปฏิบัติ then_expression หรือ else_expression ได้อย่างละหนึ่งนิพจน์เท่านั้น ในกรณีที่ต้องการใช้ then_expression หรือ else_expression ประกอบไปด้วยหลาย ๆนิพจน์ จะต้องใช้ฟังก์ชัน Progn ช่วยมีรูปแบบดังนี้

(Progn
(Expression1)
(Expression2)
.
.
.
)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน **Cond**

รูปแบบ *(Cond*
((Condition1) (Expression1) (Expression2)...)
((Condition2) (Expression1) (Expression2)...)
.
.
.
(else_expression)
)

เมื่อโปรแกรมดำเนินการมาถึงฟังก์ชัน Cond จะเข้าไปตรวจสอบ condition1 ก่อน หากเป็นจริง (T) ก็จะปฏิบัตินิพจน์ที่อยู่หลัง condition1 นั้นจนครบ แล้วออกจากฟังก์ชัน cond ไป แต่ถ้าเป็นเท็จ (nil) ก็จะเข้าไปทดสอบ condition2 ต่อไป หากเป็นเท็จอีกก็จะเข้าไปทดสอบ condition อื่นๆต่อไป จนหมด หากไม่มี condition ใดเป็นจริง ก็จะเข้าไปทำในส่วนของ else_expression สำหรับ else_expression นี้จะกำหนดหรือไม่กำหนดก็ได้ขึ้นอยู่กับลักษณะงาน

ฟังก์ชัน **While**

รูปแบบ *(While (Condition)*
(Expression1)
(Expression2)
.
.
.
)

ลักษณะการทำงาน เมื่อโปรแกรมดำเนินมาถึงฟังก์ชัน while ก็จะตรวจสอบ condition ก่อนหากเป็นจริง (T) ก็จะปฏิบัตินิพจน์ที่ระบุภายใต้ฟังก์ชัน while แต่ถ้าเป็นเท็จ (nil) ก็จะหยุดออกจากฟังก์ชัน while ไปทำนิพจน์อื่นๆต่อไป

ฟังก์ชัน **Repeat**

รูปแบบ *(Repeat loop*
(Expression1)
(Expression2)
.
.
.
)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับฟังก์ชันนี้มีลักษณะการทำงานคล้ายๆกับฟังก์ชัน while แต่การกำหนดจำนวนรอบที่จะวนซ้ำ (loop) จะใช้อาร์กิวเมนต์ที่มีค่าตัวเลขเป็น integer เป็นตัวกำหนด โดยที่ loop คือจำนวนรอบที่จะทำการวนซ้ำ การกำหนด loop อาจจะใช้นิพจน์ ค่าคงที่ หรือตัวแปรก็ได้ แต่ต้องเป็นข้อมูลชนิด integer

-ฟังก์ชันที่ใช้ในการเข้าถึงฐานข้อมูลแบบแปลน

ฟังก์ชัน **entnext**

รูปแบบ *(entnext ชื่อของส่วนประกอบ)*

ถ้าใช้โดยไม่มีอาร์กิวเมนต์ Entnext จะให้ผลลัพธ์เป็นชื่อส่วนประกอบจริงของส่วนประกอบตัวแรกในฐานข้อมูล แต่ถ้าอาร์กิวเมนต์เป็นชื่อของส่วนประกอบแล้ว Entnext จะให้ผลลัพธ์เป็นส่วนประกอบย่อยตัวแรกของชื่อส่วนประกอบส่วนประกอบย่อยเป็นส่วนประกอบที่บรรจุอยู่ในส่วนประกอบที่ซับซ้อน เช่น จุดมุมของเส้น polyline หรือ attribute ของ block

ฟังก์ชัน **entlast**

รูปแบบ *(entlast)*

เป็นฟังก์ชันที่จะให้ผลลัพธ์เป็นชื่อส่วนประกอบของส่วนประกอบตัวสุดท้ายที่เพิ่มเข้าไปในฐานข้อมูลของแบบแปลน

ฟังก์ชัน **entsel**

รูปแบบ *(entsel prompt)*

จะแสดง prompt ให้ผู้ใช้เลือกส่วนประกอบแล้วให้ผลลัพธ์เป็นรายการขององค์ประกอบตัวแรก ซึ่งเป็นชื่อของส่วนประกอบ และองค์ประกอบตัวที่สอง ซึ่งเป็นตำแหน่งจุดที่ใช้เลือกส่วนประกอบนั้นๆ เราสามารถใส่ prompt เข้าไปด้วยได้ แต่ถ้าไม่มีการใส่ prompt แล้วจะมีการใส่ prompt ของ "Select object" ให้เองโดยอัตโนมัติ

ฟังก์ชัน **entdel**

รูปแบบ *(entdel ชื่อของส่วนประกอบ)*

ทำการลบชื่อของส่วนประกอบแต่ถ้าส่วนประกอบเดิมถูกลบในขบวนการแก้ไขแบบแปลนแล้วชื่อของส่วนประกอบนั้นจะถูกเรียกกลับมา

ฟังก์ชัน **entget**

รูปแบบ *(entget ชื่อของส่วนประกอบ)*

ให้ผลลัพธ์เป็นรายการคุณสมบัติของชื่อของส่วนประกอบ

ฟังก์ชัน **entmod**

รูปแบบ *(entmod รายการคุณสมบัติ)*

จะทำการปรับปรุงฐานข้อมูลของแบบแปลนในส่วนที่ใช้เก็บชื่อของส่วนประกอบที่ปรากฏในรายการคุณสมบัติ ชื่อของส่วนประกอบเป็นรายการย่อยที่มีรหัสกลุ่ม -1 ของรายการคุณสมบัติ

ฟังก์ชัน **entupd**

รูปแบบ *(entupd ชื่อของส่วนประกอบ)*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะทำการปรับปรุงจุดมุมของ Polyline และ Attribute ของ Block ซึ่งได้เปลี่ยนแปลงโดยการใช้ Entmod ให้เป็นรูปแบบใหม่ตามที่ตั้งไว้

-ฟังก์ชันต่างๆสำหรับการจัดการกับชุดของตัวเลือก

ฟังก์ชัน **ssadd**

รูปแบบ *(ssadd ชื่อของส่วนประกอบ ชุดของตัวเลือก)*

ทำหน้าที่ในการสร้างชุดของตัวเลือก ถ้าใช้โดยไม่มีอาร์กิวเมนต์แล้วมันจะสร้างชุดของตัวเลือกที่ไม่มีส่วนประกอบ แต่ถ้ามีเฉพาะชื่อของส่วนประกอบเป็นอาร์กิวเมนต์ แล้วมันจะสร้างชุดของตัวเลือก ซึ่งบรรจุส่วนประกอบนั้นภายใน แต่ถ้าชื่อของส่วนประกอบ และชื่อชุดของตัวเลือกถูกกำหนดให้ แล้วส่วนประกอบจะถูกเพิ่มเข้าไปในชุดของตัวเลือกนั้น

ฟังก์ชัน **ssdel**

รูปแบบ *(ssdel ชื่อของส่วนประกอบ ชุดของตัวเลือก)*

ทำหน้าที่ ลบส่วนประกอบออกจากชุดของตัวเลือก และจะให้ผลลัพธ์เป็นชื่อของชุดตัวเลือก ถ้าส่วนประกอบนั้นไม่ได้เป็นสมาชิกตัวหนึ่งของชุดตัวเลือกนั้น แล้วมันจะให้ผลลัพธ์เป็นค่า nil

ฟังก์ชัน **sslength**

รูปแบบ *(sslength ชุดของตัวเลือก)*

จะให้ผลลัพธ์เป็นจำนวนของส่วนประกอบต่างๆในชุดของตัวเลือก

ฟังก์ชัน **ssmemb**

รูปแบบ *(ssmemb ชื่อส่วนประกอบ ชุดของตัวเลือก)*

ทำหน้าที่ตรวจสอบดูว่า ส่วนประกอบหนึ่งเป็นสมาชิกของชุดตัวเลือกหรือไม่ ถ้ามันเป็นแล้ว ssmemb จะให้ผลลัพธ์เป็นชื่อของชุดตัวเลือก แต่ถ้าไม่ ssmemb จะให้ผลลัพธ์เป็นค่า nil

ฟังก์ชัน **ssname**

รูปแบบ *(ssname ชุดของตัวเลือก เลขชี้ส่วนประกอบ)*

จะให้ผลลัพธ์เป็นชื่อส่วนประกอบของส่วนประกอบเดี่ยวในชุดของตัวเลือกหนึ่ง อาร์กิวเมนต์ตัวที่สองของ ssname จะเป็นหมายเลขของส่วนประกอบภายในชุดของตัวเลือกหมายเลขส่วนประกอบต่างๆจะเริ่มต้นที่ 0 และไปเรื่อยๆจนถึงจำนวนทั้งหมดของส่วนประกอบในชุดของตัวเลือกลบด้วยหนึ่ง

ฟังก์ชัน **ssget**

รูปแบบ *(ssget Argument)*

ฟังก์ชัน ssget ย่อมาจาก Select set get คือทำการรอรับค่าชุดของตัวเลือก เมื่อฟังก์ชันถูกใช้โดยไม่มีส่วนของอาร์กิวเมนต์ในนิพจน์แล้วจะปรากฏพร้อมต์ "Select objects:" และผู้ใช้สามารถเลือกวัตถุทีละชิ้น หรือกลุ่มของวัตถุได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้อาร์กิวเมนต์ในฟังก์ชัน ssget มีรูปแบบดังนี้

(ssget "P")	เลือกชุดของตัวเลือกครั้งสุดท้าย
(ssget "L")	เลือกชิ้นส่วนของวัตถุที่วาดเป็นครั้งล่าสุด
(ssget "W" '(1 1) '(4 4))	เลือกชิ้นส่วนวัตถุด้วยวิธีกรอบของวินโดว์โดยให้มุมของมันเป็นตำแหน่งของโคออร์ดิเนต 1,1 และ 4,4
(ssget "C" '(3 3) '(8 8))	เลือกชิ้นส่วนวัตถุด้วยวิธีกรอบของครอสซิงโดยให้มุมของมัน เป็นตำแหน่งโคออร์ดิเนต 3,3 และ 8,8
(ssget '(3 4))	เลือกชิ้นส่วนของวัตถุที่ตำแหน่งโคออร์ดิเนต 3,4
(ssget "X" code)	เลือกชิ้นส่วนของวัตถุตามคุณสมบัติกรหัสกำหนด

ตารางที่2.12 แสดงรหัสกลุ่มต่างๆที่ใช้เป็นรายการของตัวกรองในฟังก์ชัน ssget

รหัสกลุ่ม	คุณสมบัติร่วม
0	ชนิดของส่วนประกอบจริง
2	ชื่อของ Block
6	ชื่อของ Linetype
7	ชื่อของรูปแบบตัวอักขระ
8	ชื่อของ Layer
38	การแบ่งระดับ (Elevation)
39	ความหนา (Thickness)
62	หมายเลขสี 0 ตาม block ,256 ตาม layer

2. ฟังก์ชันที่ผู้ใช้กำหนดขึ้นเอง (User defined function)

-เรียกใช้งานได้เช่นเดียวกับคอมมานด์

-เรียกใช้งานโดยฟังก์ชัน

3.เครื่องหมายดำเนินการ ใน ออโตลิปส์ มีเครื่องหมายดำเนินการเช่นเดียวกับ ภาษาคอมพิวเตอร์อื่นๆ ที่ช่วยในการสร้างเงื่อนไขสำหรับตัดสินใจ แบ่งออกได้ดังนี้

-เครื่องหมายเปรียบเทียบ (relational operator เครื่องหมาย เปรียบเทียบ ใน ออโตลิปส์ เรียกอีกอย่างว่า predicates

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องหมายดำเนินการและความหมาย

เครื่องหมายดำเนินการ	ความหมาย
<	น้อยกว่า
>	มากกว่า
<=	น้อยกว่าหรือเท่ากับ
>=	มากกว่าหรือเท่ากับ
=	เท่ากับ
/=	ไม่เท่ากับ
eq	เท่ากันอย่างแท้จริง
equal	ท่ากับ (ใช้กับตัวแปรแบบรายการ)
listP	ตัวแปรเป็นแบบรายการหรือไม่
minUSP	จำนวนมีค่าเป็นลบหรือไม่
numberP	ตัวแปรเป็นแบบ จำนวนจริงหรือจำนวนเต็มหรือไม่
ZeroP	ตัวแปรมีค่าเป็นศูนย์หรือไม่

-เครื่องหมายทางตรรก (logical operators) เครื่องหมายทางตรรกที่ใช้อยู่มี 3 ชนิด คือ

-**AND** ใช้เปรียบเทียบสถานะทางตรรก จะให้ค่าเป็นจริง(True) ก็ต่อเมื่อทุกๆ สถานะมีค่าเป็นจริง

-**OR** ใช้เปรียบเทียบสถานะ ถ้ามีสถานะใดสถานะหนึ่งเป็นจริงจะให้ค่าเป็น จริงได้

-**NOT** ใช้สำหรับเปลี่ยนสถานะทางตรรกจาก nil(เป็นเท็จ(false)) ให้เป็น T(true) และจาก T(true) ให้เป็น nil(false)

4. นิพจน์ นิพจน์เปรียบเสมือนชุดคำสั่ง ซึ่งเป็นการนำเอาฟังก์ชัน อาร์กิวเมนต์ หรือ เครื่องหมายดำเนินการมาสัมพันธ์กันเพื่อให้ได้ผลลัพธ์ใหม่ เช่น

(+ A 30) เรียกว่า นิพจน์ ซึ่งประกอบด้วย ฟังก์ชัน + และมีอาร์กิวเมนต์คือ A 30

ในโปรแกรม ออโตลิปส์ แต่ละโปรแกรมจะประกอบด้วยนิพจน์หลายๆนิพจน์ เขียนต่อกันไปเป็นบล็อก หรืออาจจะเป็นหลายๆบรรทัดเรียงกันไป อย่างไรก็ตามนิพจน์มีกฎเกณฑ์ดังนี้

1.ทุกๆนิพจน์ไม่ว่าจะมีอาร์กิวเมนต์หรือไม่ก็ตาม จะต้องขึ้นต้นด้วยเครื่องหมายวงเล็บเปิดก่อน และปิดท้ายด้วยเครื่องหมายวงเล็บปิด (opening and closing parentheses)

2.จำนวนเครื่องหมายวงเล็บเปิดจะต้องเท่ากับจำนวนเครื่องหมายวงเล็บปิดในกรณีที่มีนิพจน์ซ้อนอยู่ในนิพจน์ เช่น

(Setq B2 (+ 5 A))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกอบด้วย นิพจน์ (+ 5 A)และนิพจน์ (Setq B2 (+ 5 A))

ในกรณีที่เราใส่เครื่องหมายวงเล็บปิดไม่ครบออโตลิปส์จะยังไม่ทำงานจะรอจนกว่าผู้ใช้งานใส่ให้จนครบ

3.ในกรณีที่มีนิพจน์ซ้อนกันเป็น 1 นิพจน์ ออโตลิปส์จะทำวงเล็บในสุดก่อนแล้วขยับออกมาทำวงเล็บนอกๆต่อไป

4. จะต้องเว้นช่องว่าง 1 อักษรระหว่างฟังก์ชันและอาร์กิวเมนต์สำหรับทุกๆนิพจน์

5. นิพจน์ที่มีการประมวลผลหรือรับข้อมูลจากภายนอก เช่น คีย์บอร์ด หรือ เมาส์ หรือ อุปกรณ์อื่นๆ อาจมีค่าหรือไม่มีค่าก็ได้ ถ้าไม่มีค่าถือว่าเป็น nil(false) หรือถ้ามีค่าจะเป็น T(true)

2.16.5 การสร้างโปรแกรม ออโตลิปส์

ออโตลิปส์ ถือว่าโปรแกรมภาษา ออโตลิส์ ที่เขียนขึ้นคือ ฟังก์ชันที่ผู้ใช้สร้างไว้ใช้งาน โดยที่ออโตลิปส์ มีฟังก์ชันที่ใช้สำหรับการเขียนโปรแกรม ออโตลิปส์ (หรือ ออโตลิปส์ function นั่นเอง) คือ **DEFUN** ซึ่งย่อมาจาก DEfine Function ลักษณะ การกำหนดมีรูปแบบดังนี้

(DEFUN C:FuncNAME (Arguments)
(Expression)
(Expression)
.
.
.
)

สังเกตได้ว่า จำนวนวงเล็บเปิดและวงเล็บปิดจะสมดุลกัน ใน ออโตลิปส์ จะถือว่าการใช้อักษรใหญ่หรืออักษรเล็กมีค่าเหมือนกัน เพราะฉะนั้น **DEFUN** อาจจะใช้ตัวเล็กหรือตัวอักษรเล็กผสมอักษรตัวใหญ่ก็ได้

FuncNAME คือ ชื่อฟังก์ชันที่เราสร้างขึ้น จะใช้อักษรเล็กหรือใหญ่ก็ได้ ชื่อฟังก์ชันที่ตั้งนี้ค่อนข้างสำคัญ จะต้องไม่ซ้ำกับชื่อฟังก์ชันที่มีอยู่แล้ว และหลังจากตั้งชื่อแล้วจะต้องจำให้ได้ เพื่อที่จะได้เรียกใช้ในภายหลัง

C: เป็นตัวเลือก จะมีหรือไม่มีก็ได้ ความหมายนั้นไม่ได้หมายถึงไดรฟ์ C แต่เป็นตัวกำหนดว่า ถ้ามีแล้ว ฟังก์ชันนั้นสามารถเรียกใช้ได้ เช่นเดียวกับคำสั่ง ออโตแคด โดยทั่วไปแต่ถ้าไม่มีแล้วการเรียกใช้จะต้องเรียกในลักษณะฟังก์ชันหรือกำหนดให้ฟังก์ชันอื่นมาเรียกใช้อีกต่อหนึ่ง โปรดอย่าลืมว่า C: ไม่ใช่ไดรฟ์ เพราะฉะนั้นเรากำหนดเป็น A: หรือ B: ไม่ได้เด็ดขาด

Arguments เป็นรายการตัวแปรที่จะใช้ในฟังก์ชันอาจจะกำหนดหรือไม่กำหนดก็ได้ คล้ายกับการเขียนโปรแกรมภาษา C

Expression คือรายการคำสั่งหรือหน้าที่ที่ต้องการให้ ออโตลิปส์ ปฏิบัติ กฎเกณฑ์ ของนิพจน์ได้กล่าวมาแล้ว

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างโปรแกรมหรือฟังก์ชันทำได้ 2 ลักษณะคือ

-สร้างขณะที่อยู่ Command:(direct mode)

-สร้างเป็นโปรแกรมเก็บไว้ในไฟล์ (programmed mode)

หากต้องการนำโปรแกรม ออโตลิปส์ ไว้ใช้ครั้งต่อไปๆ ควรสร้างเก็บไว้ในไฟล์ เพราะการสร้างขณะอยู่ที่ Command: เมื่อ QUIT หรือ ไฟดับ โปรแกรมหรือฟังก์ชันที่สร้างไว้จะหายไป

การสร้างโปรแกรม ออโตลิปส์ เก็บไว้ในไฟล์

วิธีนี้เป็นวิธีที่ดีกว่าเนื่องจากเราสามารถที่จะนำโปรแกรม ออโตลิปส์ ที่เก็บไว้ในไฟล์มาแก้ไขได้ หากทดลองใช้แล้วปรากฏว่ามีข้อผิดพลาด หรือถ้าต้องการปรับปรุงให้โปรแกรมมีประสิทธิภาพเพิ่มขึ้น นอกจากนี้ยังสามารถเรียกใช้ได้หลายๆครั้งอีกด้วย

การสร้างสามารถทำได้โดยใชเท็กอิดิตเตอร์ (text editor) เขียนโปรแกรมให้อยู่ในรูปของไฟล์ตัวอักษร ที่มีนามสกุลเป็น .LSP text editor ที่เหมาะสมที่สุดในตอนนี้คือ EDLIN ของดอส

2.16.6 การเรียกใช้โปรแกรม ออโตลิปส์

ก่อนที่จะใช้โปรแกรม ออโตลิปส์ ที่เก็บไว้ในไฟล์ จะต้องทำการเรียกโปรแกรมเข้ามาในหน่วยความจำก่อน เรียกว่า โหลดโปรแกรม ซึ่ง ออโตลิปส์ มีฟังก์ชัน load ทำหน้าที่เรียกโปรแกรมเข้ามาในหน่วยความจำ รูปแบบการใช้งานมีดังนี้

(LOAD "filename")

filename เป็นชื่อโปรแกรมหรือชื่อไฟล์โปรแกรม แต่ไม่ต้องรวมนามสกุล .LSP เข้าไป ในกรณีที่ต้องการเรียกโปรแกรมจากไดเรกทอรีอื่น จะต้องมีการบอก path ด้วย แต่มีข้อแม้ว่าจะต้องใช้เครื่องหมาย / หรือ \\ แทนเครื่องหมาย \ ที่ใช้ตามปกติ เช่น ต้องการโหลดจาก B:\LSP\TEST.LSP ให้ใช้รูปแบบ

(LOAD "B"/LSP/TEST") หรือ (LOAD "B:\\LSP\\TEST")

การเรียกโปรแกรม ออโตลิปส์ โดยอัตโนมัติขณะที่เข้าสู่ ออโตแคด

ในกรณีที่ต้องการให้เครื่องโหลดโปรแกรมออโตลิปส์ทุกครั้งขณะที่เริ่มเข้าสู่ออโตแคด ทำได้โดยเขียนโปรแกรม แล้วกำหนดให้ชื่อไฟล์เป็น ACAD.LSP โดยปกติขณะที่เข้าสู่ ออโตแคด เครื่องจะทำการตรวจสอบก่อนว่า มีไฟล์ ACAD.LSP ในไดเรกทอรีระบบหรือไดเรกทอรีปัจจุบัน (current directory) หรือไม่ ถ้ามีก็จะโหลดเข้าไปในหน่วยความจำโดยอัตโนมัติ และขณะที่โหลดจะมีข้อความบอกว่า

"Loading acad.lsp..."

2.16.7 การเพิ่มความเร็วในการรัน ออโตลิปส์ File

จากการรันโปรแกรมออโตลิปส์ ที่ผ่านมาคงเห็นว่าขณะที่ทำโปรแกรมก็จะมีข้อความแสดงขั้นตอนการปฏิบัติคำสั่งขึ้นที่บนจอบริเวณ command line ด้วยซึ่งลักษณะนี้เป็นส่วนหนึ่งที่ทำให้การรันโปรแกรมล่าช้าลงไป ยิ่งถ้าโปรแกรมที่ยาวๆ จะเห็นได้ชัดทีเดียว วิธีที่จะปิดไม่ให้มีข้อความขึ้นมาขณะรัน เรากำหนดที่ตัวแปรระบบที่ชื่อว่า **CMDECHO** (Command Echo) ให้มีค่าเป็นศูนย์ อาจจะใช้คำสั่ง SETVAR ขณะที่อยู่ Command: หรือ จะใช้นิพจน์ *(Setvar "CMDECHO" 0)* ของออโตลิปส์ก็ได้ ซึ่งวิธีนี้นิยมกว่าโดยการ เขียนไว้ที่ต้นๆโปรแกรม

เอกสารนี้เป็น ... ไม่ว่า ... ของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณ และการสร้าง

ความสามารถของโปรแกรมออโตลิป(AutoLISP) โดยใช้ออโตแคดเป็นตัวอินเตอร์พรีทเตอร์(Interpreter)นั้น คือสามารถทำการเขียนโปรแกรมออโตลิปเรียกใช้คำสั่งต่างๆของออโตแคดได้ และยังทำการแสดงผลของโปรแกรมโดยใช้การเรียกเมนูบนออโตแคดได้ ซึ่งจะทำการเขียนโปรแกรมออโตลิปเป็นโปรแกรมย่อยต่างๆ ตามที่ได้กำหนดไว้ในเมนู ซึ่งเมนูนี้เราก็สามารถสร้างได้จากการเขียนโปรแกรมออโตลิปเช่นกัน

ส่วนประกอบของโครงงานนี้จะแบ่งออกเป็นหลายส่วนด้วยกันคือ

1.ส่วนของโปรแกรม ACAD.LSP ซึ่งเป็นโปรแกรมที่เก็บคำสั่งต่างๆในเมนูที่สร้างขึ้นมาสามารถแยกออกเป็น 3 โปรแกรมหลักใหญ่ คือ

1.C:TOP () :เป็นโปรแกรมเริ่มต้น ของโปรแกรมการสร้างภาพวัตถุ3มิติจากภาพฉาย3ด้าน โดยจะทำการเรียกใช้โปรแกรมย่อยอื่นๆ ซึ่งทำการเรียกใช้ได้จากการเรียกเมนูในออโตแคด

2.C:3DCONE () :เป็นโปรแกรมเริ่มต้น ของโปรแกรมการสร้างภาพทรงกรวย3มิติ และ นำมาสร้างภาพคลี่ของทรงกรวยที่สร้างมาแล้ว

3.C:BEGIN () :เป็นโปรแกรมเริ่มต้น ของโปรแกรมการสร้างภาพบ้าน3มิติจากภาพแบบแปลน 2 มิติ ซึ่งผู้ใช้สามารถกำหนด และ ตกแต่งส่วนต่างๆของบ้านได้เอง และ สามารถดูผลของการสร้างได้ จากแนวระนาบ3มิติตามมุมมองที่ต้องการได้

2.ส่วนของโปรแกรม ACAD.MNU เป็นโปรแกรมที่สร้างเมนูที่ใช้บนออโตแแคดเพื่อสนับสนุนการเรียกใช้คำสั่งในโปรแกรมACAD.LSP โดยไม่ต้องทำการจำคำสั่งต่างๆในโปรแกรมACAD.LSP เนื่องจากสามารถดูคำสั่งได้จากเมนู โดยประกอบด้วยส่วนย่อยต่างๆ คือ การสร้างscreen menu , การสร้าง popup menu , การสร้าง icon menu เป็นต้น

3.ส่วนของโปรแกรมDIALOG1.DCL เป็นโปรแกรมการสร้างไดอะลอกบล็อกที่เรียกใช้โดยโปรแกรม ACAD.LSP ในส่วนของโปรแกรมการสร้างบ้าน เพื่อเป็นการป้อนค่าเริ่มต้นต่างๆ โดยถูกเรียกใช้โดยออโตแคด ซึ่งเพิ่งเริ่มมีในออโตแคดลีริส12เท่านั้น เพราะช่วยสร้างความสะดวกในการเลือกตัวเลือกต่างๆได้สะดวกกว่าให้ผู้ใช้พิมพ์ข้อมูลผ่านทาง command line

4.ส่วนของโปรแกรม DIALOG2.DCL เป็นโปรแกรมการสร้างไดอะลอกบล็อกที่เรียกใช้โดยโปรแกรม ACAD.LSP ในส่วนของโปรแกรมการสร้างบ้าน เพื่อแสดงรายละเอียดของโครงงานนี้แบบย่อ

5.ส่วนของไฟล์นามสกุล .DWG เป็น ไฟล์ที่เก็บรูปเฟอร์นิเจอร์ต่างๆที่จะใช้ร่วมกับการตกแต่งบ้าน ซึ่งถูกเรียกใช้ออกมาจากโปรแกรม ACAD.LSP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.ส่วนของไฟล์นามสกุล .SLB เป็นไลบรารีไฟล์ ที่เก็บภาพไสลด์ของเฟอร์นิเจิร์ต่างๆ เพื่อทำการเรียกใช้โดยโปรแกรม ACAD.MNU ในส่วนของการสร้าง icon menu เพื่อให้ผู้ใช้ได้สามารถเห็นรูปเฟอร์นิเจอร์ต่างๆ ได้สะดวกก่อนทำการเลือกใช้

ในส่วนของโปรแกรม ACAD.LSP มีหลักการสร้างและคำนวณดังนี้ แบ่งย่อยออกเป็นส่วนต่างๆคือ

I.หลักการของโปรแกรม การสร้างภาพวัตถุ3มิติจากภาพฉาย3ด้าน ส่วนประกอบของโปรแกรมแยกเป็น 13ส่วน

1.C:TOP ()

:ทำการเซ็ทค่าตัวแปรต่างๆของออโตแคด เพื่อให้สนับสนุนการทำงานของโปรแกรม

:ทำการเซ็ทสิ่งแวดล้อมต่างๆเพื่อสนับสนุนการวาดภาพ

:ทำการสร้าง 4 จอภาพ ซึ่งประกอบด้วย

-จอแสดงภาพของด้าน Top view

-จอแสดงภาพของด้าน Front view

-จอแสดงภาพของด้าน Side view

-จอแสดงภาพไอโซเมตริก(Isometric) ของวัตถุที่สร้างจากภาพฉาย 3 ด้าน (ดูรูปที่ 2 ประกอบ)

-เซ็ทมุมมองตามแนวเวคเตอร์ (-1,-1,1) ให้กับหน้าจอไอโซเมตริก และเซ็ทค่าสิ่งแวดล้อมต่างๆ

-ทำการสร้างชั้นของภาพ และ สีประจำชั้นของภาพนั้น ชั้นของภาพประกอบด้วย top สีเหลือง, front สีแดง, side สีฟ้า, 9 สีเขียว, 0 สีเขียว

-เซ็ทให้จอภาพปัจจุบันอยู่ที่หน้าจอ top และทำการเซ็ทมุมมองตามแนวเวคเตอร์(0,0,1)

เซ็ทสิ่งแวดล้อมต่างๆให้เหมาะสมกับการวาดภาพ top view เซ็ทให้ชั้นภาพปัจจุบันคือชั้น top

/* หลังจากทำโปรแกรมส่วนนี้แล้วให้ผู้ใช้ทำการวาดภาพฉายของ Top view เฉพาะกรอบนอกของภาพเท่านั้น ด้วยคำสั่ง pline(เป็นคำสั่งในโปรแกรมออโตแแคด) */

/* ดังนั้นรูปทั้งหมดที่วาดจะเป็นวัตถุของชั้น top ทั้งหมด */

/* ในการใช้คำสั่ง pline ของโปรแกรมออโตแคดมาวาดภาพฉายใดๆนั้น จะต้องวาดให้ครบวง นั่นคือ จุดเริ่มต้นและจุดจบของเส้นต้องเป็นจุดเดียวกัน */

2.C:FRONT ()

-ทำการเซ็ทจอภาพปัจจุบันเป็นจอ front view และทำการเซ็ทมุมมองตามแนวเวคเตอร์ (0,-1,0)

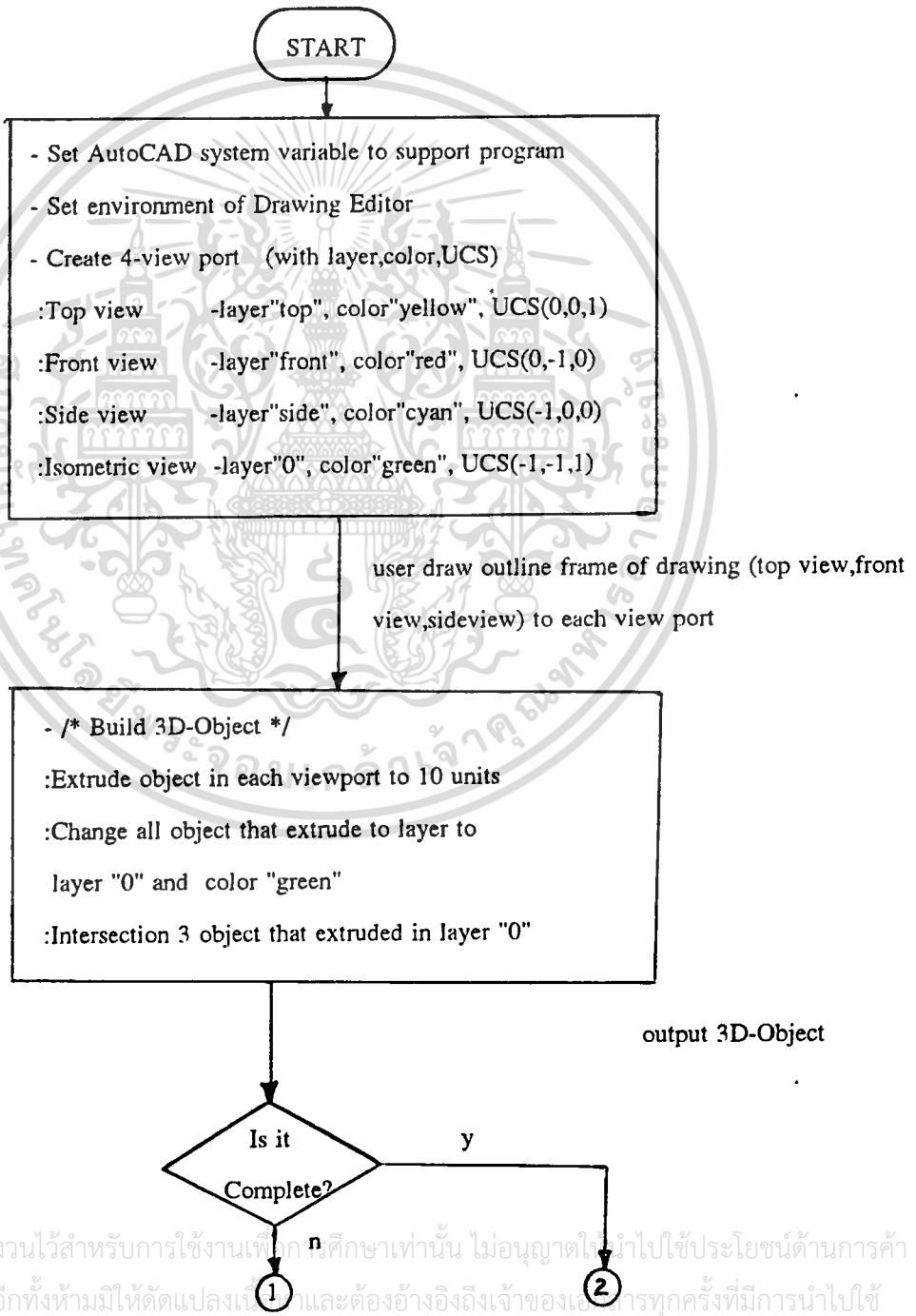
เซ็ทสิ่งแวดล้อมต่างๆให้เหมาะสมกับการวาดภาพ front view เซ็ทให้ชั้นภาพปัจจุบันคือชั้น front

/* หลังจากทำโปรแกรมส่วนนี้แล้วให้ผู้ใช้ทำการวาดภาพฉายของFront view เฉพาะกรอบนอกของภาพเท่านั้น ด้วยคำสั่ง pline(เป็นคำสั่งในโปรแกรมออโตแแคด) */

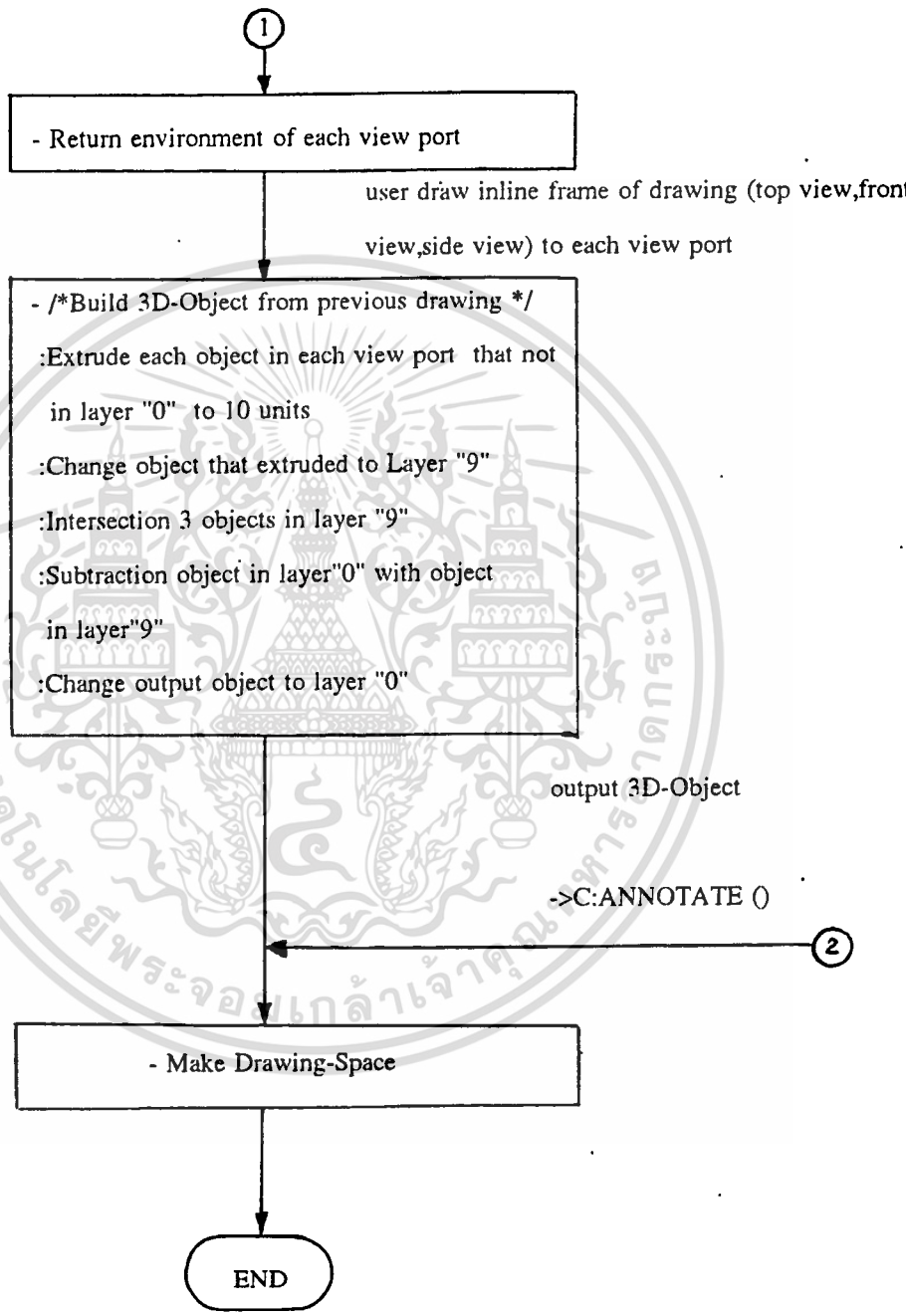
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flowchart

(Program bulid 3D-Modeling from orthographic 2D)



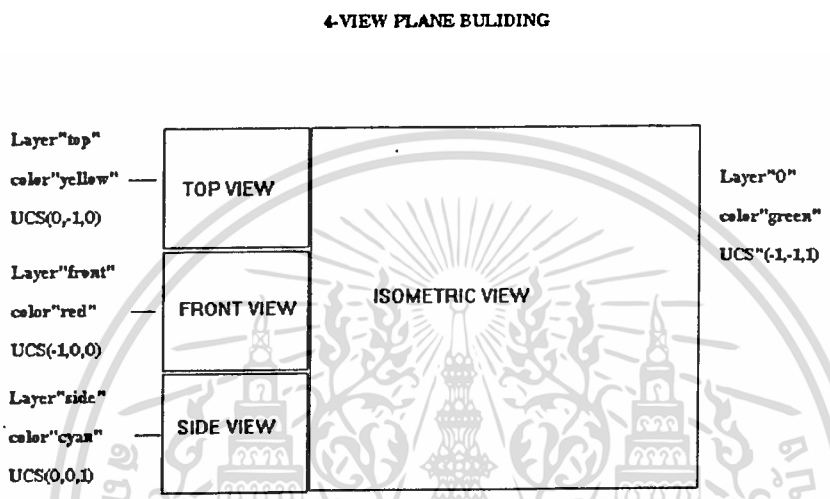
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่1. Flowchart of program build 3D-Modeling from orthographic 2D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายขั้นตอนการทำงานของโปรแกรมการสร้างวัตถุ 3 มิติ จากภาพฉาย 3 view



รูปที่2. แสดงการสร้าง 4-viewport โดย

-รูปซ้ายบน ใช้ในการวาดภาพฉาย Top view

-รูปซ้ายกลาง ใช้ในการวาดภาพฉาย Front view

-รูปซ้ายล่าง ใช้ในการวาดภาพฉาย Side view

-รูปด้านขวา ใช้ในการแสดงภาพวัตถุ 3 มิติ ที่ได้สร้างมาจากภาพฉายในแต่ละมุมมอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ดังนั้นรูปทั้งหมดที่วาดจะเป็นวัตถุของชั้น front ทั้งหมด */

3.C:SIDE ()

-ทำการเซ็ทจอภาพปัจจุบันเป็นจอ side view และทำการเซ็ทมุมมองตามแนวเวคเตอร์ (-1,0,0)

เซ็ทสิ่งแวดล้อมต่างๆให้เหมาะสมกับการวาดภาพ side view เซ็ทให้ชั้นภาพปัจจุบันคือชั้น side

/* หลังจากทำโปรแกรมส่วนนี้แล้วให้ผู้ใช้ทำการวาดภาพฉายของFront view เฉพาะกรอบนอกของภาพเท่านั้น ด้วยคำสั่ง pline(เป็นคำสั่งในโปรแกรมออโตแแคด) */

/* ดังนั้นรูปทั้งหมดที่วาดจะเป็นวัตถุของชั้น front ทั้งหมด */

4.C:COMPLETE ()

-ทำการสร้างภาพไอโซเมตริกของวัตถุที่สร้างมาจากภาพฉายทั้ง 3 ภาพนั้น โดยมีหลักการคือ

.เซ็ทสถานะหน้าจอปัจจุบันคือ isometric view

.ทำการ extrude ภาพฉายทั้งสามด้านนั้น ขึ้นมาด้านละ 10 หน่วย (เนื่องจากเรากำหนดขอบเขตของภาพมากที่สุดเพียง 10X10หน่วย)

โดยอาศัยหลักการเรียกวัตถุทีละชั้นจากโปรแกรมย่อย Getlayer

.ทำการเปลี่ยนคุณสมบัติของวัตถุทั้งหมดให้มาเป็นวัตถุในชั้น "0"

.ทำการ intersect วัตถุทั้งหมดที่อยู่ในชั้นภาพ "0" ให้เป็นวัตถุชิ้นเดียวกัน จะได้วัตถุใหม่อยู่ในชั้นภาพ "0"

(ดูรูปที่ 3 - 5 ประกอบ)

/* สอบถามผู้ใช้ว่ารายละเอียดของภาพฉายทั้ง 3 ด้านมีครบหรือยัง ถ้าครบแล้วก็เสร็จสิ้นการทำงาน ถ้ายังไม่ครบก็ให้ไปทำโปรแกรม C:TOP2 () ต่อ*/

5.C:TOP2 ()

-เซ็ทให้จอภาพปัจจุบันอยู่ที่หน้าจอ top เรียกสถานะที่ใช้กับหน้าจอtopออกมา และเซ็ทให้ชั้นภาพปัจจุบันคือ top

/* หลังจากทำโปรแกรมส่วนนี้แล้วให้ผู้ใช้ทำการวาดภาพฉายของ Top view เฉพาะรายละเอียดภายในของภาพฉายนั้นด้วยคำสั่ง pline(เป็นคำสั่งในโปรแกรมออโตแแคด) */

/*โดยจะต้องวาดให้ครบวง นั่นคือ จุดเริ่มต้นและจุดจบของเส้นต้องเป็นจุดเดียวกัน */

/* ดังนั้นรูปทั้งหมดที่วาดจะเป็นวัตถุของชั้น top ทั้งหมด */

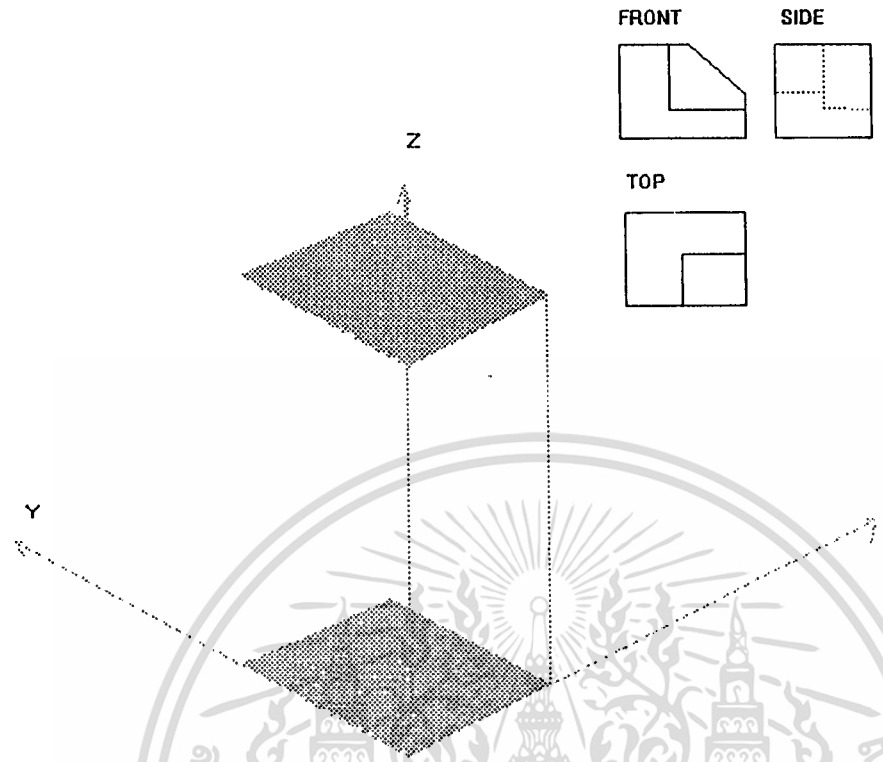
6.C:FRONT2 ()

-เซ็ทให้จอภาพปัจจุบันอยู่ที่หน้าจอfront เรียกสถานะที่ใช้กับหน้าจอfrontออกมา และเซ็ทให้ชั้นภาพปัจจุบันคือ front

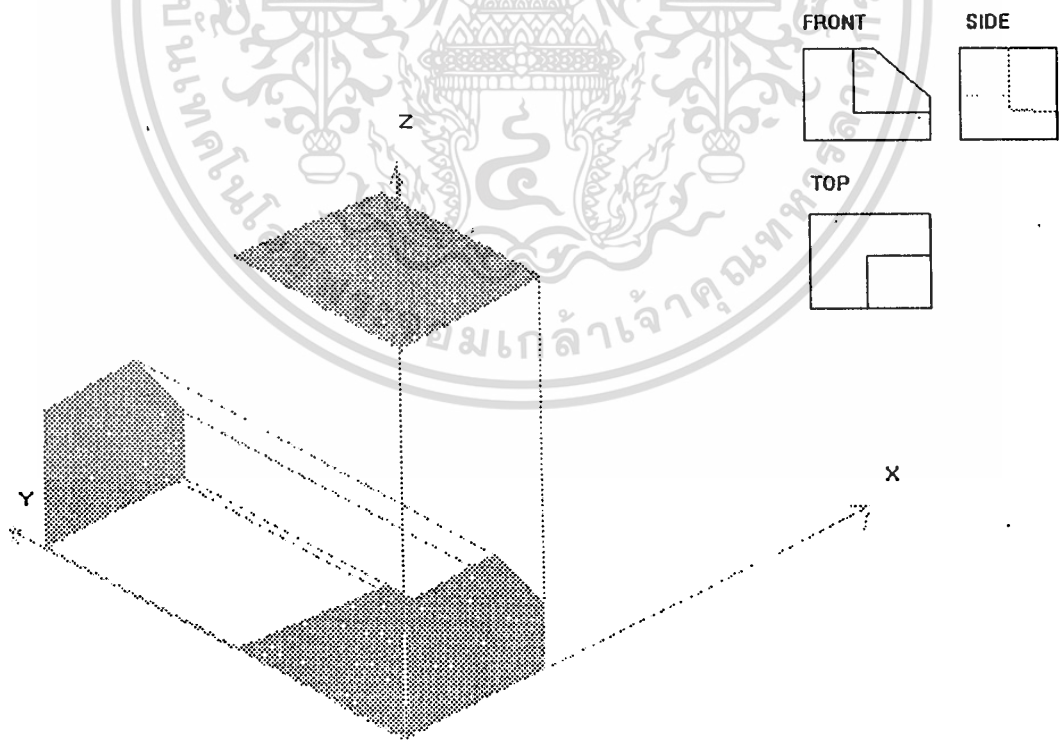
/* หลังจากทำโปรแกรมส่วนนี้แล้วให้ผู้ใช้ทำการวาดภาพฉายของ Front view เฉพาะรายละเอียดภายในของภาพฉายนั้นด้วยคำสั่ง pline(เป็นคำสั่งในโปรแกรมออโตแแคด) */

/*โดยจะต้องวาดให้ครบวง นั่นคือ จุดเริ่มต้นและจุดจบของเส้นต้องเป็นจุดเดียวกัน */

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.แสดงการextrude object in layer "top"



รูปที่4.แสดงการextrude object in layer "front"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ดังนั้นรูปทั้งหมดที่วาดจะเป็นวัตถุของชั้น front ทั้งหมด */

7.C:SIDE2 ()

-เซ็ทให้จอภาพปัจจุบันอยู่ที่หน้าจอside เรียกสถานะที่ใช้กับหน้าจอsideออกมา และเซ็ทให้ชั้นภาพปัจจุบันคือ side

/* หลังจากทำโปรแกรมส่วนนี้แล้วให้ผู้ใช้ทำการวาดภาพฉายของ Side view เฉพาะรายละเอียดภายในของภาพฉายนั้นด้วยคำสั่ง pline(เป็นคำสั่งในโปรแกรมออโตแแคด) */

/*โดยจะต้องวาดให้ครบวง นั่นคือ จุดเริ่มต้นและจุดจบของเส้นต้องเป็นจุดเดียวกัน */

/* ดังนั้นรูปทั้งหมดที่วาดจะเป็นวัตถุของชั้น side ทั้งหมด */

8.C:COMPLET2 ()

-ทำการสร้างภาพไอโซเมตริกของวัตถุที่สร้างมาจากภาพฉายทั้ง 3 ภาพที่วาดจากโปรแกรมย่อย TOP2 ,FRONT2และ SIDE2 โดยมีหลักการคือ

.เซ็ทสถานะหน้าจอปัจจุบันคือ isometric view

.ทำการ extrude ภาพฉายทั้งสามด้านนั้น ขึ้นมาด้านละ 10 หน่วย (เนื่องจากเรากำหนดขอบเขตของภาพมากที่สุดเพียง 10X10หน่วย)

โดยอาศัยหลักการเรียกวัตถุทีละชั้นจากโปรแกรมย่อย Getlayer

.ทำการเปลี่ยนคุณสมบัติของวัตถุทั้งหมดให้มาเป็นวัตถุในชั้น "9"

.ทำการ intersect วัตถุทั้งหมดที่อยู่ในชั้นภาพ "9" ให้เป็นวัตถุชิ้นเดียวกัน จะได้วัตถุใหม่อยู่ในชั้นภาพ "9"

.ทำการsubtract วัตถุที่อยู่ในชั้นภาพ"0" ด้วยวัตถุที่อยู่ในชั้นภาพ"9" จะได้วัตถุที่อยู่ในชั้นภาพ "0"

(ดูรูปที่ 6-7 ประกอบ)

/* สอบถามผู้ใช้ว่ารายละเอียดของภาพฉายทั้ง 3 ด้านมีครบหรือยัง ถ้าครบแล้วก็เสร็จสิ้นการทำงาน ถ้ายังไม่ครบก็ให้ไปทำโปรแกรม C:TOP2 () ต่อ*/

9.GETLAYER

-เป็นโปรแกรมย่อยที่ถูกเรียกจากโปรแกรมย่อยอันอื่น

-ทำหน้าที่เก็บวัตถุที่อยู่ในชั้นภาพที่ต้องการเป็นกลุ่มเดียวกันเพื่อนำไปใช้ในโปรแกรมอื่นๆ

10.C:PAPER ()

-ทำการสร้างแบบแปลนที่ประกอบด้วยภาพฉายทั้งสามด้าน และ ภาพของวัตถุ3มิติที่ได้ทำการสร้างโดยโปรแกรมมาแล้ว โดยมีหลักการคือ

.ทำการสร้างบล็อกภาพของวัตถุในแต่ละหน้าจอเก็บเป็นไฟล์ไว้

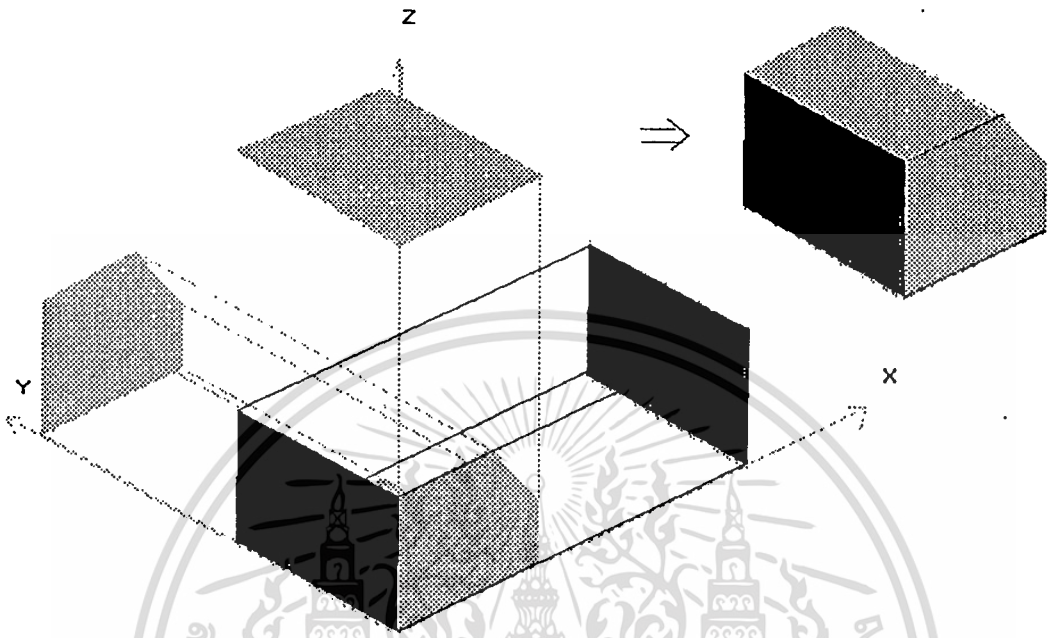
.และมาทำการสอดแทรกเข้าไปในแบบแปลนตามจุดที่กำหนดไว้

.เขียนส่วนประกอบต่างๆของแบบแปลนให้ครบ

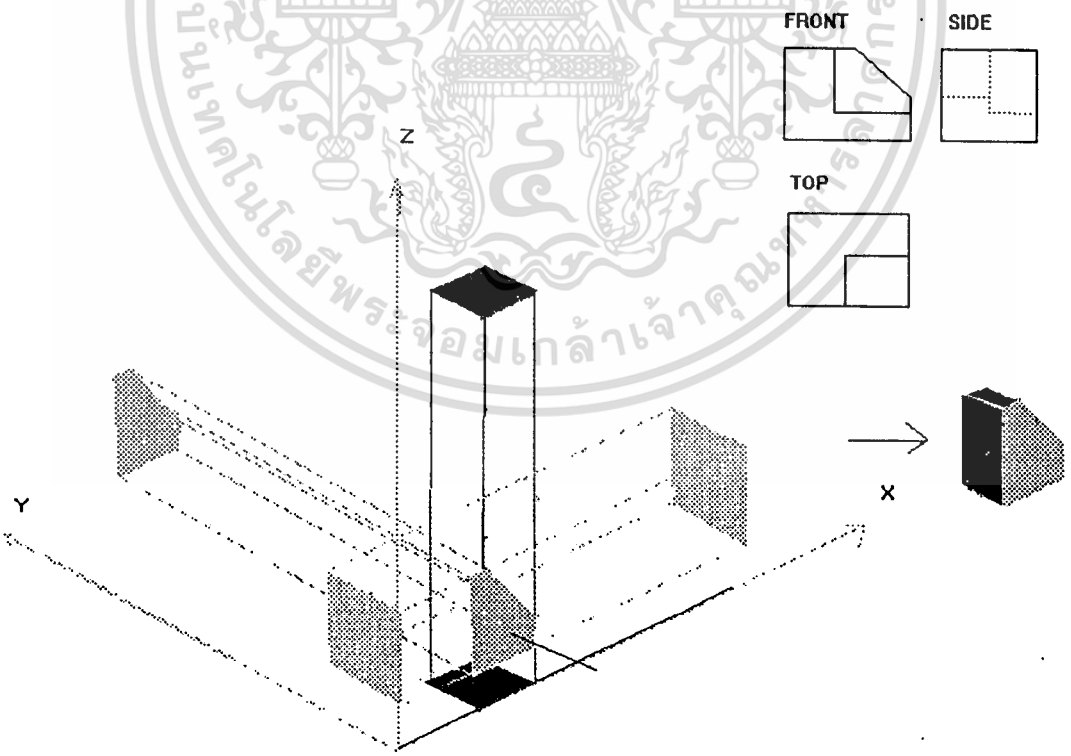
11.C:RETURN ()

-ทำการกลับไปสู่หน้าจอการสร้างภาพ3มิติ ในขณะที่อยู่ในหน้าจอการสร้างแบบแปลน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

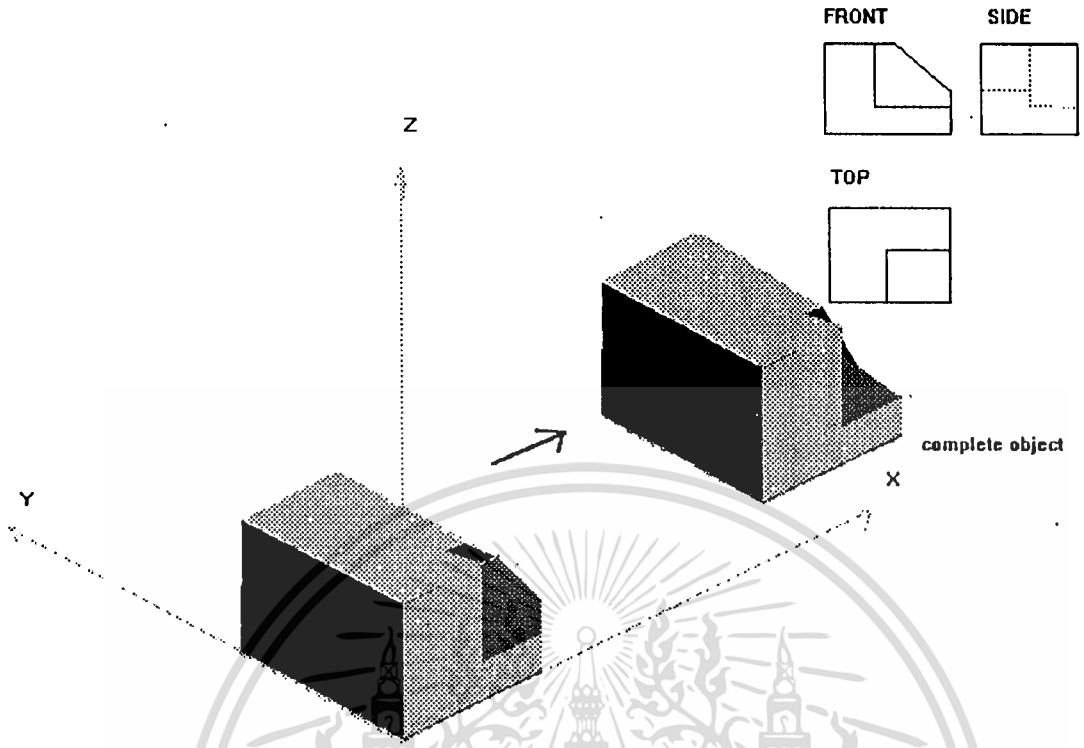


รูปที่5. แสดงการ extrude object in layer "side" และ ทำการintersect 3 object เป็น object layer "0"



รูปที่6.แสดงการextrude object in each layer และทำการ intersect 3 objects ให้เป็นobject in layer "9"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่7. แสดงการ subtract object in layer "0" with object in layer "9"
จะได้ภาพวัตถุ 3มิติที่สมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12.C:RETURNP ()

-ทำการกลับไปสู่หน้าจอการสร้างแบบแปลน ในขณะที่อยู่ในหน้าจอการสร้างภาพ3มิติ

13.C:ANOTATE ()

-ทำการหมุนภาพวัตถุ3มิติที่ได้ทำการสร้างเสร็จแล้ว โดยอาศัยคำสั่ง DVIEW ของโปรแกรมออโตแคดมาประยุกต์ใช้

II.หลักการของโปรแกรม การสร้างภาพทรงกรวย3มิติ และ สร้างภาพคลื่ออกมา

ส่วนประกอบของโปรแกรมแยกเป็น 2 ส่วน

1.C:3DCONE ()

-เป็นการสร้างภาพทรงกรวย 3 มิติ โดยมีหลักการคือ

-ทำการตั้งค่าตัวแปรของระบบให้เหมาะสม

-ทำการรับค่าตัวแปรจากผู้ใช้ซึ่งประกอบด้วย

.cen :เก็บค่าจุดศูนย์กลางของรูปทรงกรวย

.max-rad :เก็บค่ารัศมีของฐานของทรงกรวย

.min-rad :เก็บค่ารัศมีของยอดของทรงกรวย

.h :เก็บค่าความสูงของกรวย

.num-seg :เก็บค่าความละเอียดในการวาด

-ทำการสร้างภาพทรงกรวยโดยมี 2 เงื่อนไขคือ

1.เมื่อรัศมีของกรวยเป็น 0 มีหลักการคือ

-คำนวณหาจุด 3 จุด และ มุม$ (ดูรูปที่ 9 ประกอบ) คือ

จุด cen = cen

จุด pt2 = (polar(0.0) , cen , max-rad)

จุด pt3 = (polar($) , cen , max-rad) โดย

$ = 2*pi*(360/num-seg) / 360

2.เมื่อรัศมีของกรวยมากกว่า 0 มีหลักการคือ

-คำนวณหาจุด4 จุด และ มุม$ (ดูรูปที่ 9 ประกอบ)

จุด pt1 = (polar(0.0) , cen , min-rad)

จุด pt2 = (polar(0.0) , cen , max-rad)

จุด pt3 = (polar($) , cen , max-rad)

จุด pt4 = (polar ($) , cen , min-rad) โดย

$ = 2*pi*(360/num-seg) / 360

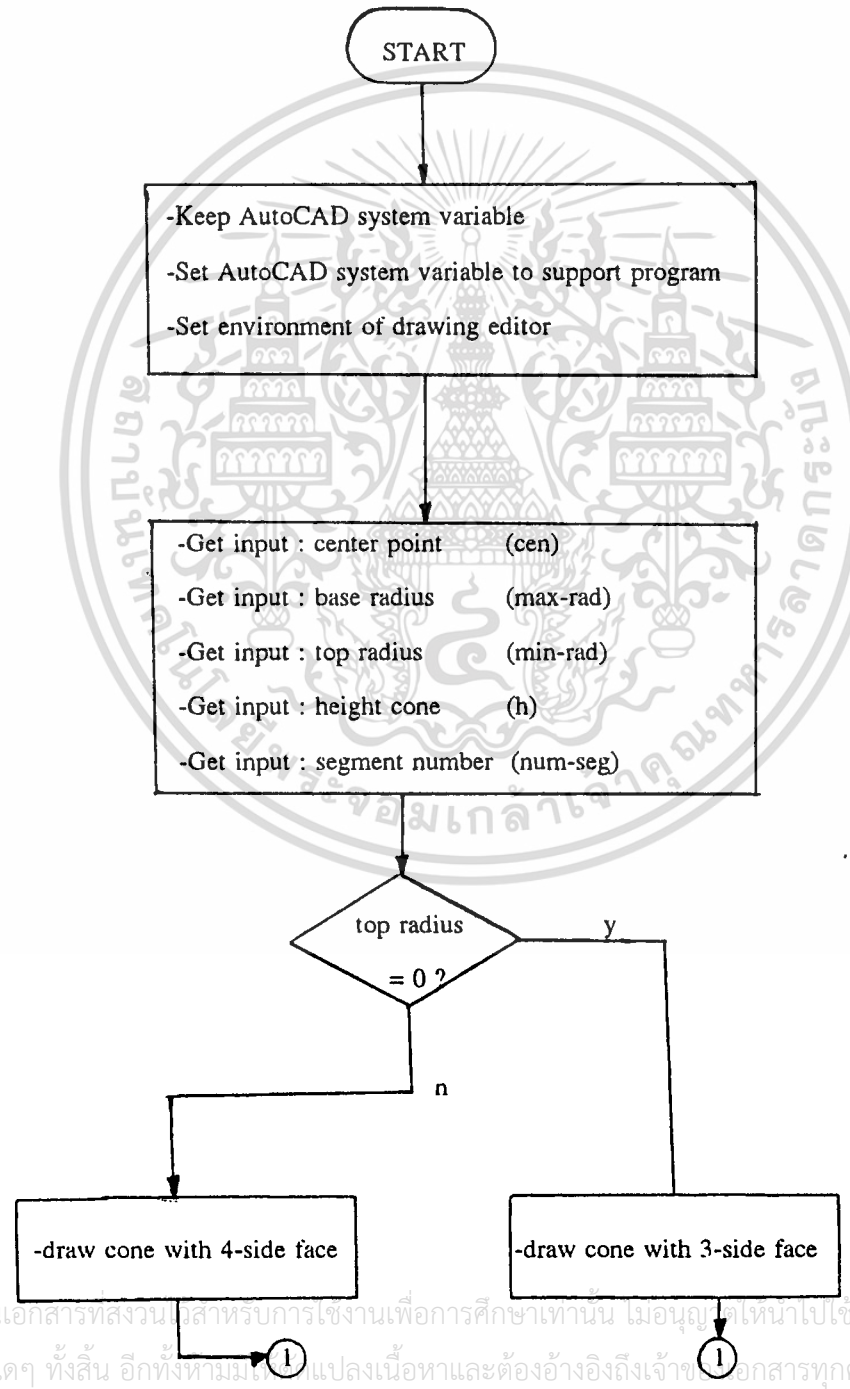
-เมื่อได้จุดต่าง ๆที่ต้องการแล้วก็ทำการสร้าง พื้นผิว เชื่อมจุดเหล่านี้

-ทำการคัดลอก พื้นผิวเหล่านี้ไปในแนวแกน Z เท่ากับจำนวน num-seg จะได้ครบวงกรวย พอดี

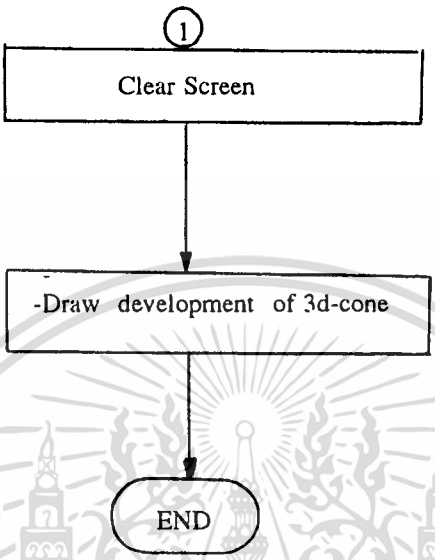
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flowchart

(Build 3d-cone and make development of 3d-cone)



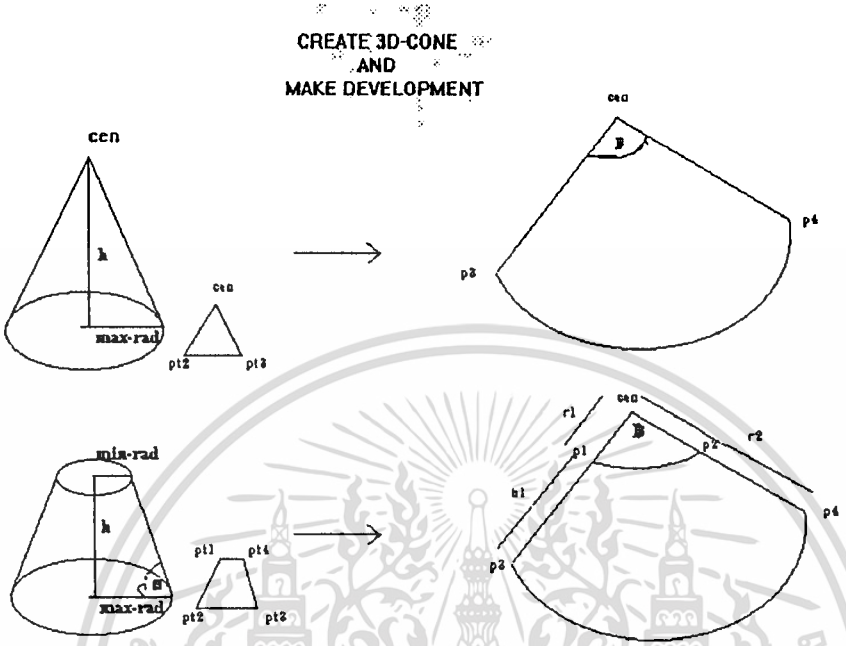
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่8. Flowchart of program make 3D-cone and build Development of 3D-cone

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายขั้นตอนการทำงานของโปรแกรมการสร้างรูปกรวย 3 มิติ
และนำมาสร้างภาพคลี่



:3-side face build 3d-cone
pt2 = polar(0.0), cen, max-rad
pt3 = polar($), cen, max-rad
$ = 2*Pi*(360/num-seg)/ 360

:4-side face build 3d-face
pt1 = polar(0.0), cen, min-rad
pt2 = polar(0.0), cen, max-rad
pt3 = polar($), cen, max-rad
pt4 = polar($), cen, min-rad

-ทำการสร้าง face เชื่อมจุดเหล่านี้
-สร้างarray of object รอบ แกน Z
โดยมีจำนวนเท่ากับ num-seg

ทั้ง 3-side face และ 4-side face สร้างภาพคลี่โดยใช้
Algorithm เดียวกันคือ

Θ = arctan(h / (max-rad-min-rad))

h1 = (h) * (max-rad - min-rad)
r1 = min-rad / (cosΘ)
r2 = r1+h1
Θ = (2*Pi*min-rad) / r1

p1 = polar(0), cen, r1
p2 = polar(Θ), cen, r1
p3 = polar(0), cen, r2
p4 = polar(Θ), cen, r2

และทำการสร้าง arc-p1,p2 กับ arc-p3,p4
และลากเส้นเชื่อม p1-p3 กับ p2-p4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่9. แสดงรายละเอียดของAlgorithmของโปรแกรมการสร้างรูปกรวย3มิติ และ นำมาสร้างภาพคลี่

2.C:CONEPRJ ()

-ทำการสร้างภาพคลี่ ของภาพทรงกรวยที่สร้างจากโปรแกรม 3DCONE มีหลักการคือ (ดูรูปที่ 9 ประกอบ)

1.คำนวณหาจุดปลายของมุมต่างๆ โดย

จุด p1 = (polar(o) , cen , r1) โดย

o = arctan(h / (max-rad - min-rad))

r1 = min-rad / cos(o)

จุด p2 = (polar(฿) , cen , r1) โดย

฿ = (2 * pi * min-rad) / r1

r1 = min-rad / cos(o)

จุด p3 = (polar(o) , cen , r2) โดย

r2 = r1 + h1

r1 = min-rad / cos(o)

h1 = h + max-rad - min-rad

o = arctan(h / (max-rad - min-rad))

จุด p4 = (polar(฿) , cen , r2) โดย

฿ = (2 * pi * min-rad) / r1

r2 = r1 + h1

r1 = min-rad / cos(o)

h1 = h + max-rad - min-rad

-และทำการสร้างภาพคลี่โดยทำการลากเส้นเชื่อมจุด p1-p3 และ p2-p4 และทำการสร้าง arcระหว่างp1-p2และ arcระหว่าง p3-p4

III.หลักการของโปรแกรม การสร้างภาพบ้าน3มิติ จากภาพแบบแปลน 2 มิติ

:โดยผู้ใช้สามารถทำการตกแต่งแบบแปลนบ้านในแปลน 2 มิติด้วยตัวเอง และ ทำการมองดูบ้านที่ได้สร้างในมุมมองต่างๆ ในระบบ3 มิติได้

ส่วนประกอบของโปรแกรมแยกเป็น 5 ส่วน

1.c:begin ( )

-ทำการตั้งค่าตัวแปรต่างๆของโปรแกรมออโตแคดให้เหมาะสม

-ทำการตั้งระบบ และ สิ่งแวดล้อมต่างๆให้เหมาะสมกับการวาด

-กำหนดตัวแปรคงที่ซึ่งประกอบด้วย

ar = 10 ;ความกว้างของสนามรอบๆบ้าน (หน่วยเป็น เมตร)

fl = 0.1 ;ความหนาของพื้นบ้าน (หน่วยเป็น เมตร)

hi = 2.3 ;ความสูงของกำแพง (หน่วยเป็น เมตร)

post = 0.2 ;ความกว้างของเสา (หน่วยเป็น เมตร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

:ตัวแปรคงที่เหล่านี้สามารถเปลี่ยนแปลงได้ แต่ตัวแปร hi และ fl จะต้องแปรเปลี่ยนไปตามบล็อกของรูปบันได (ไฟล์ stair1.dwg และ stair2.dwg)

-เรียกโปรแกรมย่อย show เพื่อทำการเปิดไดอะล็อกบล็อก ให้ผู้ใช้ป้อนค่าให้ตัวแปร

wid ;ความกว้างของบ้าน (หน่วยเป็น เมตร)

len ;ความยาวของบ้าน (หน่วยเป็น เมตร)

class ;จำนวนชั้นของบ้าน (หน่วยเป็น ชั้น)

manh ;ความสูงของผู้ใช้ (หน่วยเป็น เมตร)

-สั่งให้ผู้ใช้เรียกเมนู House[Step1] เพื่อทำการให้ผู้ใช้เลือกสีของกำแพงบ้าน ,สีของพื้นบ้าน และ รูปแบบของหลังคา

โดยเก็บไว้ในตัวแปร

wallcl ;สีของกำแพงบ้าน

floorcl ;สีของพื้นบ้าน

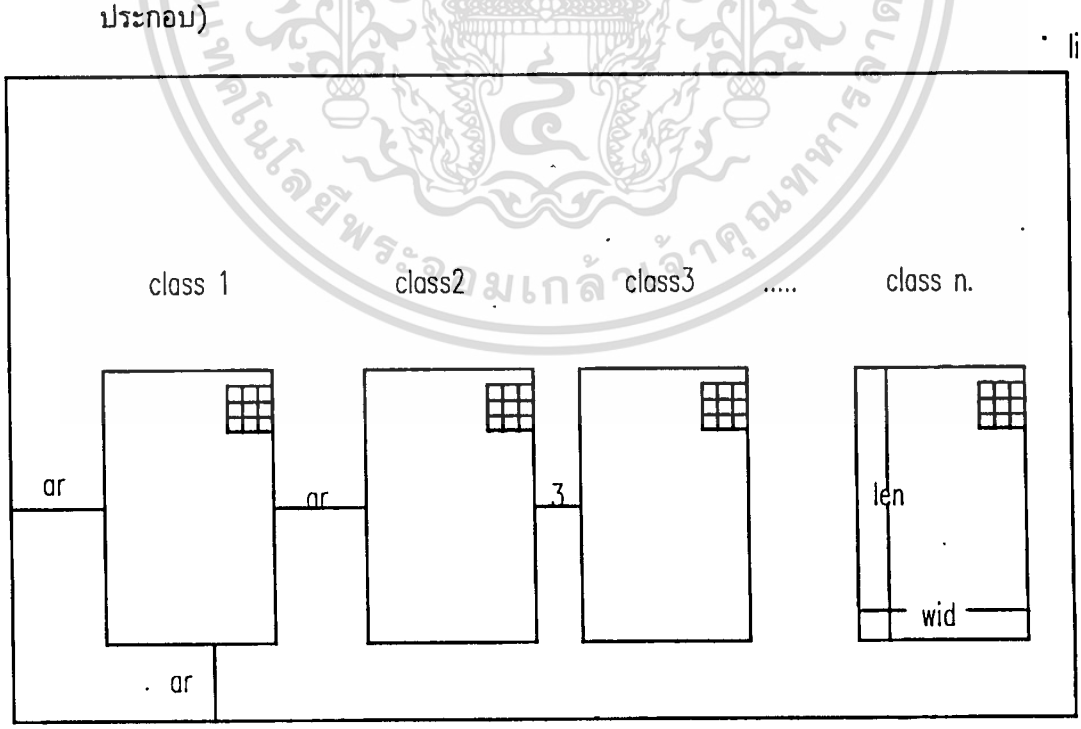
roof ;รูปแบบของหลังคา

/*ถ้าเป็นแบบแปลนที่ได้ทำการสร้างมาไว้แล้วก็ไม่ต้องทำ ขั้นตอนนี้*/

-หลังจากนั้นจะทำการเรียกโปรแกรมย่อย C:FLOOR () เพื่อทำการสร้างส่วนประกอบของบ้านต่อไป

2.C:FLOOR ()

-ทำการสร้างพื้นบ้าน , เสา , บันได และ หลังคาของบ้าน มีหลักการคือ (ดูรูปที่ 10 ประกอบ)



รูปที่10. แสดงขอบเขตของบ้าน และตัวแปรต่างๆที่เกี่ยวข้อง

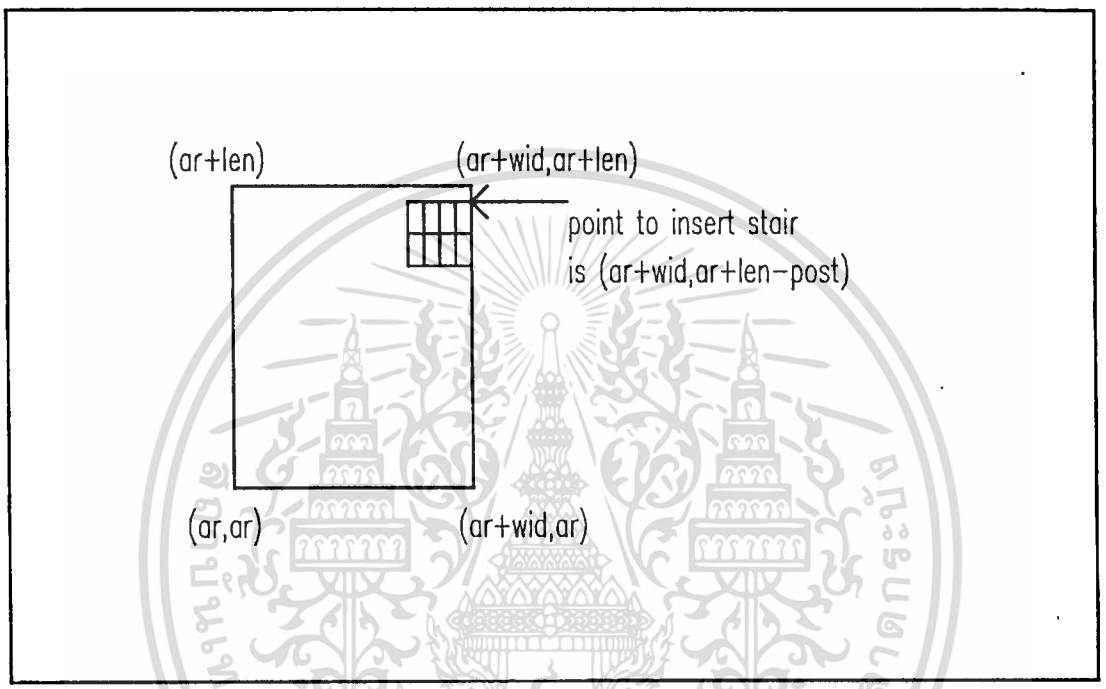
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.คำนวณหาขอบเขตของภาพจากสูตร

limit = { (2*ar) + (class*wid) + (3*(class - 1)) , (len + (2*ar)) }

2.สร้างชั้นภาพต่างๆ สำหรับเก็บวัตถุประเภทต่างๆ

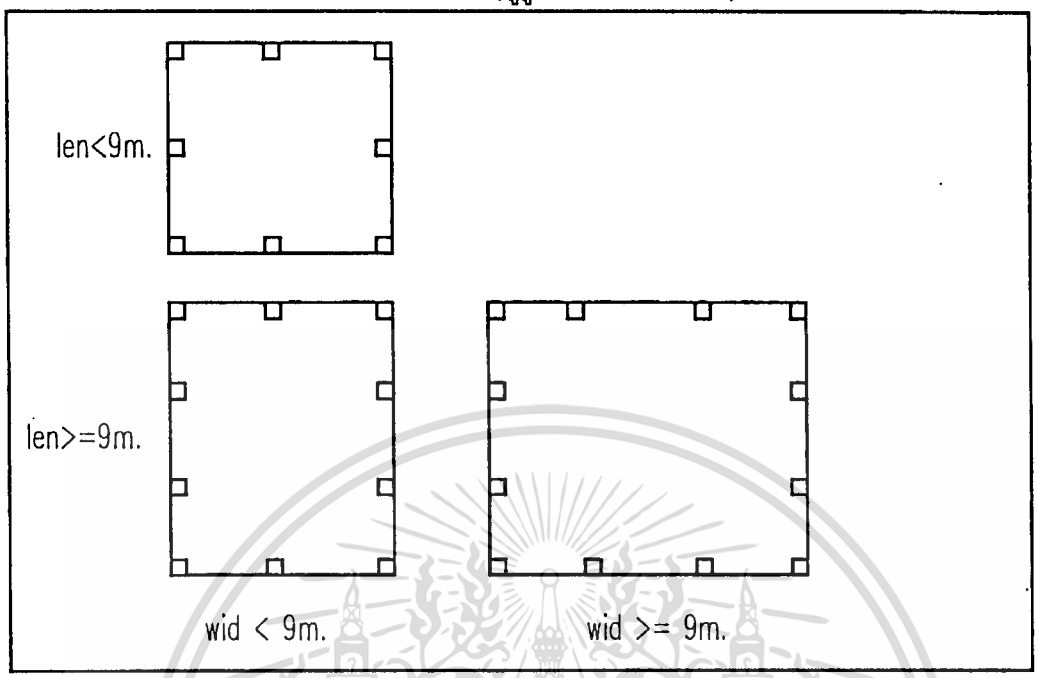
3.สร้างพื้น และ พรมปูพื้นของ ชั้นที่ 1 และทำการสอดแทรกบันไดเข้ามาในแปลน (ดูรูปที่ 11 ประกอบ)



รูปที่11. แสดงจุดต่างๆของขอบเขตในการวาด สร้างพื้น และ พรมปูพื้นของ ชั้นที่ 1 และทำการสอดแทรกบันไดเข้ามาในแปลน

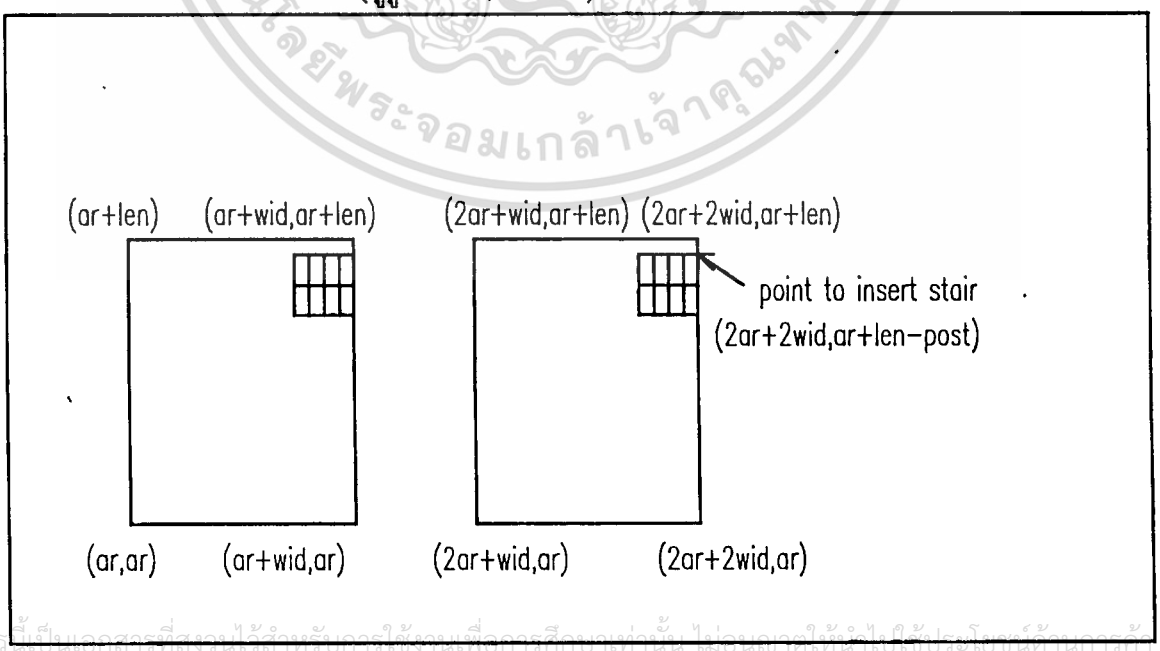
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.ทำการวาดเสาของบ้าน โดยมีหลักการคือ (ดูรูปที่ 12 ประกอบ)



รูปที่12. แสดงการวาดเสาให้แปลนบ้าน โดยถ้าความยาวของด้านกว้าง หรือ ด้านยาวน้อยกว่า 9 เมตร จะทำการวาดเสา ด้านละ 3 ต้น แต่ถ้าความยาวของด้านกว้าง หรือ ด้านยาวมากกว่า 9 เมตร จะทำการวาดเสา ด้านละ 4 ต้น

5.ถ้าจำนวนชั้นของบ้านเท่ากับ 2 ชั้น สร้างพื้น และ พรมปูพื้นของ ชั้นที่ 2 และทำการสอดแทรกบันไดเข้ามาในแปลน (ดูรูปที่ 13 ประกอบ)



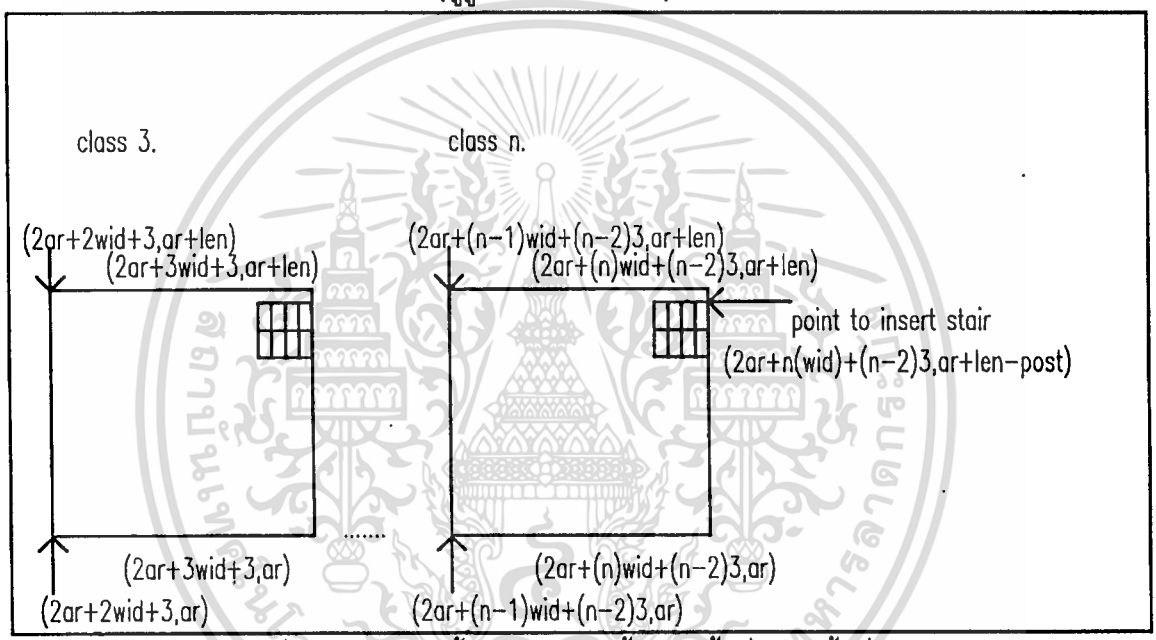
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่13. แสดงจุดต่างๆของขอบเขตในการวาด สร้างพื้น และ พรมปูพื้นของ ชั้นที่ 2 และทำการสอดแทรกบันไดเข้ามาในแปลน

6.ทำการวาดเสาของบ้าน โดยมีหลักการคือ (ดูรูปที่ 12 ประกอบ)

โดยถ้าความยาวของด้านกว้าง หรือ ด้านยาวน้อยกว่า 9 เมตร จะทำการวาดเสา ด้านละ 3 ต้น แต่ถ้าความยาวของด้านกว้าง หรือ ด้านยาวมากกว่า 9 เมตร จะทำการวาดเสา ด้านละ 4 ต้น

7.ถ้าจำนวนชั้นของบ้านมากกว่า2ชั้น สร้างพื้น และ พรมปูพื้นของ ชั้นที่ 3 ถึงชั้นที่ n และทำการสอดแทรกบันไดเข้ามาในแปลน (ดูรูปที่ 14 ประกอบ)



รูปที่14. แสดงจุดต่างๆที่ใช้ในการสร้างพื้น และ พรมปูพื้นของ ชั้นที่ 3 ถึงชั้นที่ n และทำการสอดแทรกบันไดเข้ามาในแปลน

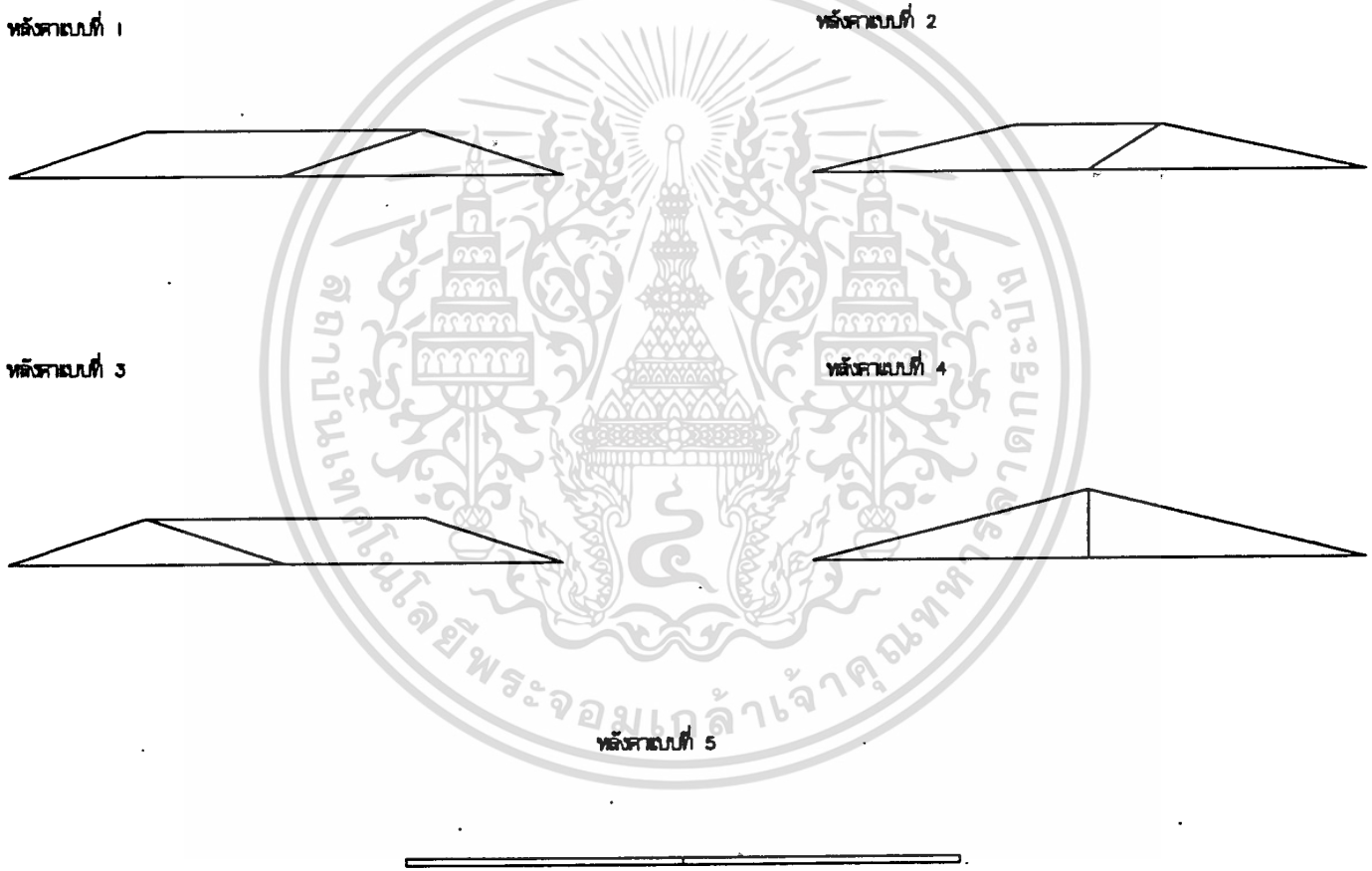
8.ทำการวาดเสาของบ้าน โดยมีหลักการคือ (ดูรูปที่ 12 ประกอบ)

โดยถ้าความยาวของด้านกว้าง หรือ ด้านยาวน้อยกว่า 9 เมตร จะทำการวาดเสา ด้านละ 3 ต้น แต่ถ้าความยาวของด้านกว้าง หรือ ด้านยาวมากกว่า 9 เมตร จะทำการวาดเสา ด้านละ 4 ต้น

9.ทำการวาดตัวอักษรประกอบแปลน แสดงชั้นของบ้าน, ทิศทาง, ความกว้าง-ความยาว-จำนวนชั้นของบ้าน(เพื่อบอกแก่ผู้ใช้ในคราวต่อไปเมื่อนำแบบแปลนนี้กลับมาสร้างอีกครั้งภายหลังทำการบันทึกไปแล้ว เพื่อให้ผู้ใช้เลือกคำสั่ง BEGIN จากเมนู เพื่อเก็บค่าเริ่มต้นที่จำเป็นในการคำนวณ)

10.ทำการสร้างหลังคา โดยเลือกเงื่อนไขที่ตรงกับตัวแปร roof ที่ผู้ใช้ได้เลือกไว้แล้วจาก icon menu มีรูปแบบต่างๆ ตามรูปที่ 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่15. แสดงรูปแบบของหลังคา 5 ชนิด

2.C:VIEWHOUSE ()

-ทำการสร้างตัวบ้าน 3มิติ ทั้งหลังให้ผู้ใช้ดู มีหลักการคือ

-ปิดชั้นภาพ txt ที่เก็บภาพเกี่ยวกับข้อความต่างๆ

-ทำการเคลื่อนย้ายแบบแปลนจากชั้นที่อยู่เหนือกว่ามาวางไว้บนชั้นที่อยู่ต่ำกว่า ที่ทำเช่นนี้ได้เพราะในขณะที่ผู้ใช้ทำการตกแต่งบ้านในแบบแปลน2มิตินั้น จริงๆแล้วได้มีการสร้างอุปกรณ์ประกอบต่างๆที่เป็น3มิติเข้าไปพร้อมกันด้วย แต่มุมมองของเวคเตอร์ (0,0,1) ทำให้ผู้ใช้มองเห็นเป็นระนาบ 2มิติ ซึ่งมีหลักการในการเคลื่อนย้ายอุปกรณ์ต่างๆในแต่ละชั้นดังนี้

1.ถ้าจำนวนชั้นเท่ากับ 1 ชั้น ก็ทำการเปลี่ยนเวคเตอร์การมองเป็น (-1,-1,0.3) ได้เลย และผู้ใช้ก็สามารถทำการขยายรูปได้ตามต้องการ

2.ถ้าจำนวนชั้นเท่ากับ 2 ชั้น ให้ทำการเคลื่อนย้ายอุปกรณ์ในชั้นที่ 2 มาวางบนชั้นที่ 1โดยทำการเลือกอุปกรณ์ทั้งหมดในขอบเขต

lim1 = { (2*ar) + wid- 0.1) , (ar- 0.1) }

lim2 = { (2*ar) + (* 26 wid) + 0.1) , (ar + len + 0.1) }

;ที่ต้องมีการเพิ่ม/ลดค่า 0.1 เนื่องจากต้องเลือกระเบียงบ้านด้วย

และทำการย้ายวัตถุที่เลือกจากจุดฐานคือ

basep = { (2*ar) + wid , ar , 0 }

และจุดเป้าหมายคือ

secp = { ar , ar , (fl + hi) }

ต้องทำการเคลื่อนย้ายในระบบ 3มิติจึงมีค่าแกน Z ร่วมด้วย

และทำการเปลี่ยนเวคเตอร์การมองเป็น (-1,-1,0.3) ผู้ใช้ก็สามารถทำการขยายรูปได้ตามต้องการ

3.ถ้าจำนวนชั้นมากกว่า 2 ชั้น ให้ทำการเคลื่อนย้ายอุปกรณ์ในชั้นที่ n มาวางบนชั้นที่ n-1 ไปเรื่อยๆจนถึงการวางอุปกรณ์จากชั้นที่2 ถึง ชั้นที่ n บนชั้นที่ 1โดยทำการเลือกอุปกรณ์ทั้งหมดในขอบเขตต่างๆ มีหลักการคำนวณคือ

lim1 = { (2*ar) + (n-2)*3 + (n-1)*wid - 0.1) , (ar - 0.1) }

lim2 = { (2*ar) + (n-2)*3 +(n)*wid) + 0.1) , (ar + len + 0.1) }

;ที่ต้องมีการเพิ่ม/ลดค่า 0.1 เนื่องจากต้องเลือกระเบียงบ้านด้วย

และทำการย้ายวัตถุที่เลือกจากจุดฐานคือ

basep = { (2*ar) + (n-2)*3 + (n-1)* wid , ar , 0 }

และจุดเป้าหมายคือ

secp = { ar , ar , (n-1)*fl + (n-1)*hi }

ต้องทำการเคลื่อนย้ายในระบบ 3มิติจึงมีค่าแกน Z ร่วมด้วย

และทำการเปลี่ยนเวคเตอร์การมองเป็น (-1,-1,0.3) ผู้ใช้ก็สามารถทำการขยายรูปได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.อุปรณ์ต่างๆที่ใช้ในการเติมแบบแปลนประกอบด้วย

1.กำแพง อยู่ในโปรแกรมย่อย

C:WALLN - เติมกำแพงโดยจุดเริ่มต้นอยู่ทางทิศเหนือของบ้าน

C:WALLS - เติมกำแพงโดยจุดเริ่มต้นอยู่ทางทิศใต้ของบ้าน

C:WALLE - เติมกำแพงโดยจุดเริ่มต้นอยู่ทางทิศตะวันออกของบ้าน

C:WALLW - เติมกำแพงโดยจุดเริ่มต้นอยู่ทางทิศตะวันตกของบ้าน

มีหลักการคือทำการวาดรูปสีเหลี่ยมพื้นผ้าที่มีความหนาเท่ากับความสูงของกำแพง (hi), กว้างเท่ากับความหนาของกำแพง(กำหนดไว้ 0.1เมตร) ,ความยาวและทิศทางของกำแพงตามที่ผู้ใช้กำหนดเอง

2.ต้นไม้ อยู่ในโปรแกรมย่อย C:TREE

มีหลักการคือ เมื่อผู้ใช้เลือกแบบของต้นไม้ได้เลือกจาก icon menu แล้วก็ทำการสอดแทรกรูปของต้นไม้ชนิดนั้นลงในตำแหน่งที่ผู้ใช้เลือก โดยรูปของต้นไม้นั้นได้สร้างเก็บเป็น บล็อกไฟล์ไว้แล้ว

3.ประตู

มีหลักการคือ เมื่อผู้ใช้เลือกแบบของประตูโดยเลือกจาก icon menu แล้ว ก็ทำการสอดแทรกรูปของประตูชนิดนั้นลงในตำแหน่งที่ผู้ใช้เลือก โดยรูปของประตูนั้นได้สร้างเก็บเป็นบล็อกไฟล์ไว้แล้ว ประกอบด้วยโปรแกรมย่อยดังนี้

C:DOORN - เติมประตูโดยทำการหันให้หน้าประตูไปทางทิศเหนือของบ้าน

C:DOORS - เติมประตูโดยทำการหันให้หน้าประตูไปทางทิศใต้ของบ้าน

C:DOORE - เติมประตูโดยทำการหันให้หน้าประตูไปทางทิศตะวันออกของบ้าน

C:DOORW- เติมประตูโดยทำการหันให้หน้าประตูไปทางทิศตะวันตกของบ้าน

4.หน้าต่าง

มีหลักการคือ เมื่อผู้ใช้เลือกแบบของหน้าต่างโดยเลือกจาก icon menu แล้ว ก็ทำการสอดแทรกรูปของหน้าต่างชนิดนั้นลงในตำแหน่งที่ผู้ใช้เลือก โดยรูปของหน้าต่างนั้นได้สร้างเก็บเป็นบล็อกไฟล์ไว้แล้ว ประกอบด้วยโปรแกรมย่อยดังนี้

C:WINDOWN - เติมหน้าต่างโดยทำการหันให้ด้านหน้าไปทางทิศเหนือของบ้าน

C:WINDOWS - เติมหน้าต่างโดยทำการหันให้ด้านหน้าไปทางทิศใต้ของบ้าน

C:WINDOWE - เติมหน้าต่างโดยทำการหันให้ด้านหน้าไปทางทิศตะวันออกของบ้าน

C:WINDOWW- เติมหน้าต่างโดยทำการหันให้ด้านหน้าไปทางทิศตะวันตกของบ้าน

5.เฟอร์นิเจอร์อื่นๆ อยู่ในโปรแกรมย่อย C:FUR

มีหลักการคือ เมื่อผู้ใช้เลือกแบบของเฟอร์นิเจอร์โดยเลือกจาก icon menu แล้ว ก็ทำการสอดแทรกรูปของเฟอร์นิเจอร์ชนิดนั้นลงในตำแหน่งที่ผู้ใช้เลือก โดยรูปของเฟอร์นิเจอร์นั้นได้สร้างเก็บเป็นบล็อกไฟล์ไว้แล้ว โดยเมื่อสอดแทรกรูปเข้ามาผู้ใช้สามารถทำการเคลื่อนย้ายเฟอร์นิเจอร์นั้นได้เลย

6.ระเบียง อยู่ในโปรแกรมย่อย C:VELANDA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีหลักการคือ ให้ผู้ใช้ทำการกำหนดตำแหน่งต่าง ๆ คือ

first = จุดเริ่มต้นของระเบียง

second = จุดจบของระเบียง

wide = จุดที่ระเบียงกว้างออกจากตัวบ้าน

โดยเราสามารถคำนวณว่าเป็นการสร้างระเบียงออกจากตัวบ้านในทิศทางใดได้ดังนี้

ทิศเหนือ -> first(y) = second(y) และ first(y) < wide(y)

ทิศใต้ -> first(y) = second(y) และ first(y) > wide(y)

ทิศตะวันออก -> first(x) = second(x) และ first(x) < wide(x)

ทิศตะวันตก -> first(x) = second(x) และ first(x) > wide(x)

หลังจากนั้นทำการวาดรูประเบียงลงบนแปลนตามขนาดที่คำนวณได้

7.ทางเดินเท้า อยู่ในโปรแกรมย่อย C:PATH ()

มีหลักการคือให้ผู้ใช้กำหนดจุดเริ่มต้น และ จุดสิ้นสุดของทางเดินเท้า และทำการวาดรูปสี่เหลี่ยมที่ระยะทางและทิศทางตามที่ผู้ใช้กำหนด และขนาดความกว้างเท่ากับที่ผู้ใช้กำหนด โดยความหนาของทางเดินกำหนดเท่ากับ 0.1เมตร

4.การเปิด หรือ การปิดประตู, หน้าต่าง ,หลังคา ,กำแพง

อาศัยหลักการปิดเปิดชั้นของภาพ ประกอบด้วยโปรแกรมย่อยคือ

C:OPENDOOR -ทำการ on layer "door"

C:OPENWINDOW -ทำการ on layer "window"

C:OPENWALL -ทำการ on layer "wall"

C:OPENROOF -ทำการ on layer "roof"

C:CLOSEDOOR -ทำการ off layer "door"

C:CLOSEWINDOW -ทำการ off layer "window"

C:CLOSEWALL -ทำการ off layer "wall"

C:CLOSEROOF -ทำการ off layer "roof"

5.C:SEE ()

เป็นโปรแกรมส่วนที่ช่วยให้ผู้ใช้สามารถมองเห็นมุมมองต่าง ๆของบ้านในระนาบ3มิติ ซึ่งเป็นส่วนของบ้านที่ผู้ใช้ได้ทำการวาดในแบบแปลน มีหลักการดังนี้

1.ทำการเซ็ทระบบมุมมอง เป็นมุมมองระนาบ2มิติของแปลน

2.จะใช้คำสั่ง DVIEW ในเงื่อนไข POINT ซึ่งเป็นคำสั่งในออโตแคด แต่ต้องการตัวแปร 2 ชนิดคือ ตำแหน่งของเป้าหมาย และ ตำแหน่งของผู้ใช้ จึงให้ผู้ใช้ป้อนค่าตัวแปรที่จำเป็นในการคำนวณคือ

cls = ชั้นที่ผู้ใช้ต้องการมอง

ang1 = ระยะทางจากพื้นถึงเป้าหมายที่ต้องการมอง

man = ตำแหน่งที่อยู่ของผู้ใช้ในแบบแปลน (x,y,z)

manh = ความสูงของผู้ใช้ (ได้จากโปรแกรมส่วน C:BEGIN ())

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

target = ตำแหน่งของเป้าหมายที่ต้องการมอง

โดยมีเงื่อนไขของขอบเขตว่าถ้าตำแหน่งของผู้ใช้อยู่ในบริเวณ สนามรอบๆชั้นที่1 ซึ่งเช็คจากเงื่อนไข i.man(x) < (ar -1.5)

ii.man(x) > (ar + wid + 1.5) and
man(x) < (2*ar)+ wid-1.5)

iii.man(y) < (ar - 1.5)

iv.man(y) > (ar + len +1.5)

แล้วกำหนดว่า ตำแหน่งของผู้ใช้คือ { man(x) , man(y) , manh }

ถ้าผู้ใช้อยู่ในบริเวณชั้นต่างๆของบ้านจะได้ ตำแหน่งของผู้ใช้คือ

mh = { man(x) , man(y) ,(fl+ manh) }

ส่วนตำแหน่งของเป้าหมายคือ { target(x),target(y),target(z) }

3.ใช้หลักการของโปรแกรมย่อย C:VIEWHOUSE () เพื่อสร้างบ้านเป็นระบบ3มิติ และใช้คำสั่ง DVIEW ของออโตแคดโดยเขียนเป็นภาษาออโตลิปดังนี้

(command "DVIEW" "" "Points" tar pos "Distance" d "Zoom" "30" ""))

ซึ่งคำสั่ง DVIEW นี้เปรียบเสมือนการใช้กล้องถ่ายรูปมองไปในทิศทางต่างๆ อธิบายได้ดังนี้

command	หมายถึง การเรียกคำสั่งในออโตแคด
..	หมายถึง การเลือกวัตถุทั้งหมด
"Point"	หมายถึง การเลือกทางเลือกpoint คือ เลือกจุดเป้าหมายในการมอง และตำแหน่งที่อยู่ของกล้อง
tar	หมายถึง ตำแหน่งของเป้าหมาย
pos	หมายถึง ตำแหน่งของกล้อง
"Distance"	หมายถึง การเลือกทางเลือก ขนาดของระยะทางระหว่างกล้องกับเป้าหมาย
d	หมายถึง ระยะทางระหว่าง ตำแหน่งของกล้อง และตำแหน่งของเป้าหมาย
"Zoom"	หมายถึง การเลือกทางเลือก ความยาวเลนซ์ของกล้อง
"30"	หมายถึง เลือกขนาดเลนซ์ของกล้องเท่ากับ 30 มิลลิเมตร
""	หมายถึง การจบคำสั่ง DVIEW

5.การหมุนภาพมุมมองที่ผู้ใช้ได้ทำการเลือกจากโปรแกรมย่อย C:SEE () ประกอบด้วยโปรแกรมย่อยดังนี้

1.C:TWISTR () เป็นการหมุนภาพ ซึ่งคล้ายกับการหมุนตัวกล้องถ่ายรูปในแนวระนาบ X-Z โดยการใช้คำสั่ง

(command "DVIEW" "" "twist" x "")

โดยที่เปลี่ยนค่า x จาก มุม 0 องศา ถึง มุม 360องศา

โดยที่ twist คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของมุมที่กล้องหมุน โดยวัดจากแนวแกน Z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.C:CAMERA2 () เป็นการหมุนภาพ ซึ่งคล้ายกับการมองผ่านกล้องในขณะที่เดินหมุนไปรอบๆเป้าหมายที่ต้องการ ในแนวแกน Z โดยการใช้คำสั่ง

(command "DVIEW" "" "camera" "" x "")

โดยที่เปลี่ยนค่า x จาก มุม -180 องศา ถึง มุม 180องศา

โดยที่ camera คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของมุมที่วัดจากแกน X

3.C:TARGET2 () เป็นการหมุนภาพ ซึ่งคล้ายกับการมองผ่านกล้องในขณะที่หมุนกล้องไปรอบๆตามแนวแกน Z โดยการใช้คำสั่ง

(command "DVIEW" "" "target" "" x "")

โดยที่เปลี่ยนค่า x จาก มุม -180 องศา ถึง มุม 180องศา

โดยที่ target คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของมุมที่วัดจากแกน X

4.C:DIST1 () เป็นการมองภาพผ่านกล้องโดยเปลี่ยนแปลงระยะทางระหว่างตัวผู้ใช้ กับเป้าหมายน้อยลงเรื่อยๆ โดยใช้คำสั่ง

(command "DVIEW" "" "distance" x "")

โดยที่เปลี่ยนค่า x จาก มากไปหาน้อย โดยค่าเริ่มต้นเท่ากับระยทางระหว่างเป้าหมายถึงตำแหน่งผู้ใช้

โดยที่ distance คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของระยะทางระหว่างเป้าหมายถึงตำแหน่งที่ผู้ใช้อยู่

5.C:DIST2 () เป็นการมองภาพผ่านกล้องโดยเปลี่ยนแปลงระยะทางระหว่างตำแหน่งผู้ใช้กับเป้าหมายมากขึ้นเรื่อยๆ โดยใช้คำสั่ง

(command "DVIEW" "" "distance" x "")

โดยที่เปลี่ยนค่า x จาก น้อยไปมาก โดยค่าเริ่มต้นเท่ากับระยะทางระหว่างเป้าหมายถึง ตำแหน่งผู้ใช้

โดยที่ distance คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของระยะทางระหว่างเป้าหมายถึงตำแหน่งที่ผู้ใช้อยู่

6.C:ZOOMIN () เป็นการมองภาพผ่านกล้องโดยเปลี่ยนแปลงขนาดของเลนซ์กล้องให้มีค่ามากขึ้นเรื่อยๆ โดยใช้คำสั่ง

(command "DVIEW" "" "zoom" x "")

โดยที่เปลี่ยนค่า x จาก น้อยไปมาก โดยค่าเริ่มต้นเท่ากับ 30 มิลลิเมตร

โดยที่ zoom คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของขนาดเลนซ์ของกล้องถ่ายรูป

7.C:ZOOMOUT () เป็นการมองภาพผ่านกล้องโดยเปลี่ยนแปลงขนาดของเลนซ์กล้องให้มีค่าน้อยลงเรื่อยๆ โดยใช้คำสั่ง

(command "DVIEW" "" "zoom" x "")

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่เปลี่ยนค่า x จาก มากไปน้อย โดยค่าเริ่มต้นเท่ากับ 30 มิลลิเมตร

โดยที่ zoom คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของขนาดเลนซ์ของกล้องถ่ายรูป

8. C:CAMERA1 () เป็นการหมุนภาพ ซึ่งคล้ายกับการมองผ่านกล้องในขณะที่เปลี่ยนตำแหน่งกล้องโดยการหมุนขึ้นและลง โดยมีเป้าหมายเป็นจุดศูนย์กลางในแนวแกน X โดยการใช้คำสั่ง

(command "DVIEW" "" "camera" x "" "")

โดยที่เปลี่ยนค่า x จาก มุมระหว่างเป้าหมายกับตำแหน่งผู้ใช้ ถึง มุม 90องศา และ จากมุม 90 องศาถึงมุมระหว่างเป้าหมายกับตำแหน่งผู้ใช้

โดยที่ camera คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของมุมที่วัดจากระนาบ XY

9.C:CAMERA11() เป็นการหมุนภาพ ซึ่งคล้ายกับการมองผ่านกล้องในขณะที่เปลี่ยนตำแหน่งกล้องให้หมุนลงและขึ้น โดยมีเป้าหมายเป็นจุดศูนย์กลางในแนวแกน X โดยการใช้คำสั่ง

(command "DVIEW" "" "camera" x "" "")

โดยที่เปลี่ยนค่า x จาก มุมระหว่างเป้าหมายกับตำแหน่งผู้ใช้ ถึง มุม -90องศา และ จากมุม -90 องศาถึงมุมระหว่างเป้าหมายกับตำแหน่งผู้ใช้

โดยที่ camera คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของมุมที่วัดจากระนาบ XY

10. C:TARGET1 () เป็นการหมุนภาพ ซึ่งคล้ายกับการมองผ่านกล้องในขณะที่ทำการหมุนกล้องอยู่กับที่ในแนวแกนX โดยการใช้คำสั่ง

(command "DVIEW" "" "target" x "" "")

โดยที่เปลี่ยนค่า x จาก มุมระหว่างเป้าหมาย กับระนาบ XY

โดยที่ target คือ ทางเลือกในการหมุนภาพของคำสั่ง DVIEW

x คือ ค่าของมุมที่วัดจากระนาบ XY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์ส่วนประกอบที่ใช้ร่วมกันกับโปรแกรม ACAD.LSP
เก็บไฟล์เหล่านี้ไว้ในไดเรคตอรี C:\ACAD

Install program AutoCAD Release 12 in All files options

ACAD.LSP - เก็บโปรแกรมหลักที่นำไปใช้รันบนโปรแกรม AutoCAD
ACAD.MNU - เมนูไฟล์ที่ใช้คู่กับโปรแกรมหลัก
ACAD.MNX - เมนูคอมไพล์ไฟล์
ACADR12.BAT - แบทไฟล์ ที่ใช้เรียกโปรแกรม AutoCAD
ACADR12.MNU - เมนูไฟล์ของโปรแกรม AutoCAD
ACADR12.MNX - เมนูคอมไพล์ไฟล์
DIALOG1.DCL - ไฟล์สร้างไดอะลอกบ็อก ถูกเรียกใช้จากโปรแกรม ACAD.LSP
DIALOG2.DCL - ไฟล์สร้างไดอะลอกบ็อก ถูกเรียกใช้จากโปรแกรม ACAD.LSP

บล็อกไฟล์เก็บรูปอุปกรณ์ต่าง ๆที่ใช้ร่วมกับโปรแกรม ACAD.LSP
ควรสร้างไดเรคตอรี(directory) C:\ACAD\BLOCK เอาไว้เก็บไฟล์เหล่านี้
(รวม 98 ไฟล์)

B6AR1.DWG	BAR2.DWG	BAR3.DWG	BAR4.DWG
BAR5.DWG	BED21.DWG	BED22.DWG	BOILER.DWG
BOTTLE.DWG	BOWL.DWG	CABINET1.DWG	CABINET2.DWG
CABINET3.DWG	CABINET4.DWG	CABINET5.DWG	CABINET6.DWG
CABINET7.DWG	CABINET8.DWG	CHAIR_1.DWG	CHAIR_2.DWG
COUNTER1.DWG	D1_N.DWG	D2_N.DWG	D3_N.DWG
DISH.DWG	DOOR1_N.DWG	DOOR2_N.DWG	DOOR3_N.DWG
DOOR4_N.DWG	DRAWER1.DWG	DRAWER2.DWG	GLASS1.DWG
GLASS2.DWG	KIT1.DWG	KIT2.DWG	LOG1.DWG
LOG2.DWG	LOG3.DWG	MANTLE1.DWG	MANTLE2.DWG
MANTLE3.DWG	OVEN.DWG	PLACE.DWG	POT.DWG
SHELF1.DW3G	SHELF10.DWG	SHELF11.DWG	SHELF12.DWG
SHELF13.DWG	SHELF2.DWG	SHELF3.DWG	SHELF4.DWG
SHELF5.DWG	SHELF6.DWG	SHELF7.DWG	SHELF8.DWG
SHELF9.DWG	SINK1.DWG	SINK2.DWG	SINK3.DWG
SINK4.DWG	SINK5.DWG	SLIDE1.DWG	SLIDE2.DWG
STOVE1.DWG	STOVE2.DWG	TABLE1.DWG	TABLE10.DWG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE11.DWG	TABLE12.DWG	TABLE13.DWG	TABLE2.DWG
TABLE3.DWG	TABLE4.DWG	TABLE5.DWG	TABLE6.DWG
TABLE7.DWG	TABLE8.DWG	TABLE9.DWG	TOIL1.DWG
TOIL2.DWG	TOIL3.DWG	TREE1.DWG	TREE2.DWG
TREE3.DWG	WARDROB1.DWG	WARDROB2.DWG	WARDROB3.DWG
WARDROB4.DWG	WARDROB5.DWG	WARDROB6.DWG	WARDROB7.DWG
WARDROB8.DWG	WINDOW1.DWG	WINDOW2.DWG	WINDOW3.DWG

ไลบรารีไฟล์เก็บสไลด์ไฟล์ของรูปอุปกรณ์ต่างๆที่ใช้ร่วมกับโปรแกรม ACAD.LSP
ควรสร้างไดเรคตอรี C:\ACAD\SLIDE เอาไว้เก็บไฟล์เหล่านี้
(รวม 9 ไฟล์)

BED.SLB	CAB.SLB	DOOR1.SLB	DOO2.SLB
KIT.SLB	SHELF.SLB	STEP1.SLB	TAB.SLB
WIN.SLB			

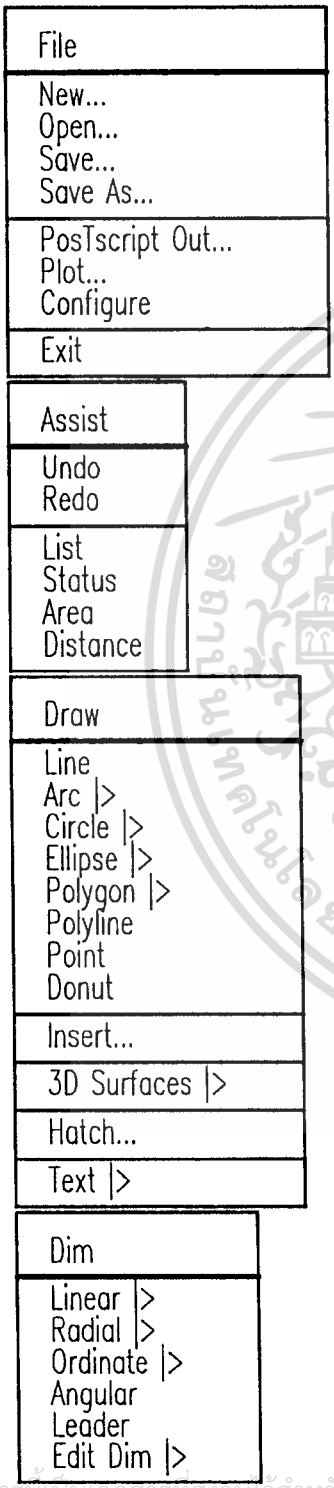
:ภายใน ACADR12.BAT ให้ทำการสร้างพาท(Path) ให้เข้าถึงไดเรคตอรี BLOCK และ SLIDE ด้วย

```
SET ACAD= C:\ACAD;C:\ACAD\SLIDE;C:\ACAD\BLOCK
```

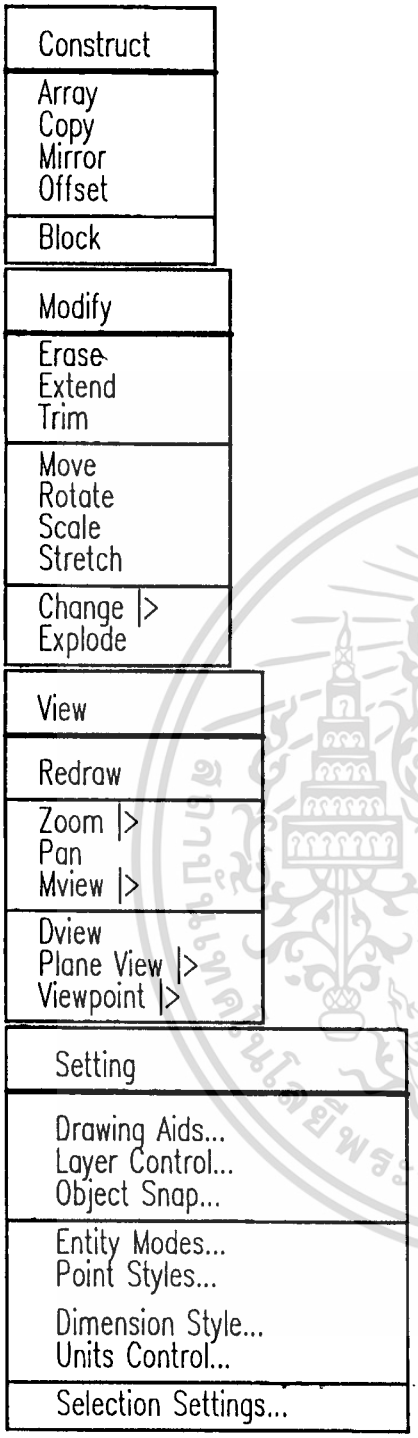
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโปรแกรม ACAD.MNU เป็นการใช้ออโตลิปช่วยในการสร้างเมนู เพื่อใช้งานบนออโตแคด ซึ่งลักษณะของเมนูที่ได้ พร้อมคำสั่งประจำเมนูนั้นๆแสดงจากคำสั่งหลัง ^C^C

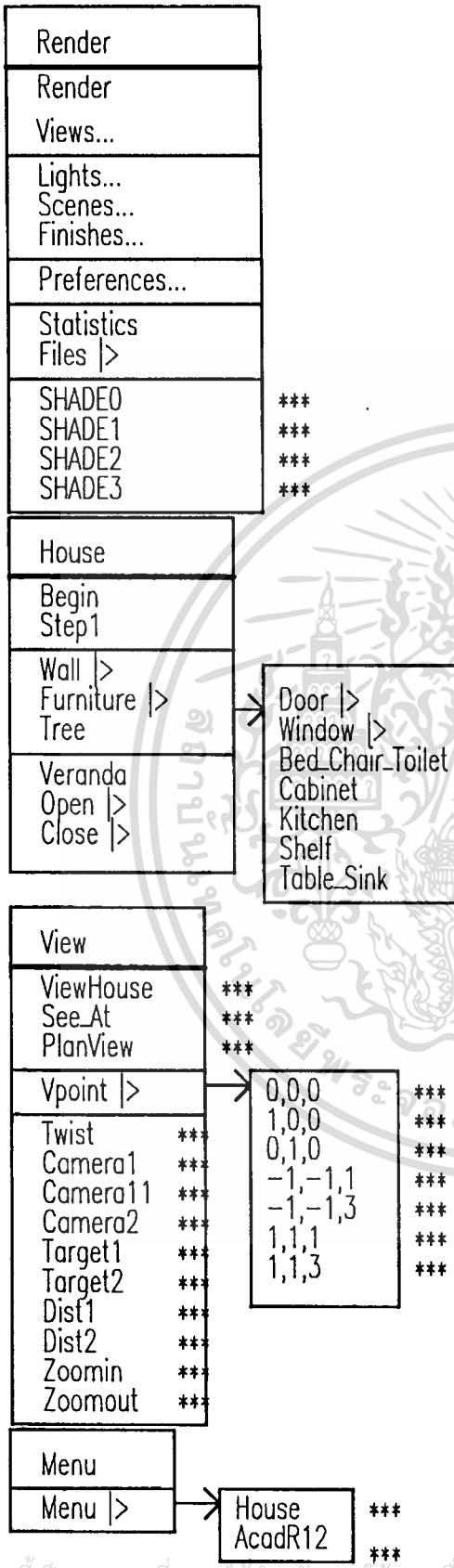
/* เมนูที่มี*** แสดงว่าเป็นส่วนที่เรียกใช้คำสั่งที่สร้างจากโปรแกรม ACAD.LSP ซึ่งเป็น ส่วนหนึ่งของโครงงานนี้ นอกนั้นจะเป็นคำสั่งมาตรฐานของออโตแคดอยู่แล้ว */



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3D	
Top	***
Front	***
Side	***
Complete	***
Top2	***
Front2	***
Side2	***
Complete2	***
Paper	***
Restore	***
ModelPlan	***
PaperPlan	***
Anotate	***
3D Cone	***
ConeProj	***

นอกเหนือจากนี้ยังมีส่วนที่สร้าง icon menu ซึ่งส่วนของโปรแกรมจะอยู่หลังคำว่า ***icon icon menu ที่ถูกสร้างขึ้นนอกดหนือจากเมนูมาตรฐานของออโตแคดคือ

1. ไอคอนเมนู tree ประกอบด้วย แบบของต้นไม้ 3 แบบ
2. ไอคอนเมนู step4 ประกอบด้วย แบบสีของกำแพง 23 สี
3. ไอคอนเมนู step5 ประกอบด้วย แบบสีของพื้น และ หลังคา 23 สี
4. ไอคอนเมนู step6 ประกอบด้วย แบบของหลังคา 5 แบบ
5. ไอคอนเมนู D_NORTH ,D_SOUTH,D_EAST,และ D_WEST ประกอบด้วยแบบของประตู 7 แบบ2
6. ไอคอนเมนู W_NORTH,W_SOUTH,W_EAST,และ W_WEST ประกอบด้วยแบบของหน้าต่าง 3 แบบ
7. ไอคอนเมนู CABINET ประกอบด้วยแบบของ โต๊ะลิ้นชัก,หิ้ง,ตู้ 19 แบบ
8. ไอคอนเมนู TABLE&SINK ประกอบด้วยแบบของ โต๊ะและชั้นวางของ 19 แบบ
9. ไอคอนเมนู SHELF ประกอบด้วยแบบของ ชั้นวาง,โต๊ะรับแขก 17 แบบ
10. ไอคอนเมนู KITCHEN ประกอบด้วยแบบของ อุปกรณ์ภายในครัว 20 แบบ
11. ไอคอนเมนู BED&CHAIR&TOILET ประกอบด้วยแบบของ อุปกรณ์ในห้องนอน,ห้องน้ำ, ห้องรับแขก 7 แบบ

ดูตัวอย่างอุปกรณ์ได้จากรูปที่ .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

เนื่องจากโครงงานนี้มีโปรแกรมที่สามารถแบ่งออกเป็น 3 ส่วนใหญ่ คือ

1. โปรแกรมสร้างวัตถุ 3 มิติจากภาพฉาย 2 มิติ 3 ด้านคือ ภาพฉายด้านบน,ภาพฉายด้านหน้า,และภาพฉายด้านข้าง
2. โปรแกรมสร้างทรงกรวย 3 มิติและนำมาสร้างภาพคลี่ของทรงกรวยนั้นได้
3. โปรแกรมสร้างบ้าน 3 มิติจากแปลนบ้าน 2 มิติ

การทดลองเริ่มด้วยทำการเปิดโปรแกรมออโตแคดขึ้นมา ซึ่งในขณะที่เครื่องคอมพิวเตอร์โหลดโปรแกรมออโตแคดนี้จะทำการโหลดโปรแกรม ACAD.LSP และ ACAD.MNU ขึ้นมาให้โดยอัตโนมัติ ซึ่งเป็นส่วนของโปรแกรมที่ทำการเขียนภาษาลิปส์เก็บไว้แล้ว

4.1 ผลการทดลองโปรแกรมสร้างภาพ 3 มิติ (Isometric Drawing) จากภาพ 2 มิติทั้ง 3 View(Top,Front,Side) ได้โดยใช้ AutoLISP

4.1 เข้าสู่โปรแกรมนี้ โดยเลือกเมนู 3D[Top] ใน Pop-up menu ที่หน้าจอจะมีการเปลี่ยนแปลงเกิดขึ้นและจะมีข้อความต่อไปนี้ปรากฏข้อความ "*** TOP VIEW *** When complete pick menu Front "

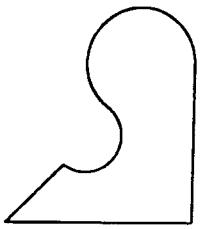
ทำการวาดภาพ 2 มิติใน View ต่างๆทั้ง 3 View โดยจะต้องทำการวาดเฉพาะเส้นรอบรูปของภาพ 2 มิติก่อน ซึ่งต้องใช้คำสั่ง Polyline 1 loop

4.2. เมื่อวาดภาพ Top View เสร็จแล้วตัวอย่างแสดงดังรูปที่ 16 ให้ Userเลือกเมนู 3D[Front] ใน Pop-up menu ซึ่งจะมีการเปลี่ยนแปลงขึ้นที่ View นี้ User จึงทำการวาดภาพ Front View ลงไปได้ และที่ Status line จะมีข้อความต่อไปนี้ปรากฏ "*** FORNT VIEW *** When complete press menu Side "

4.3.เมื่อวาดภาพFront View แล้วตัวอย่างแสดงดังรูปที่ 17 ให้ทำการเลือกเมนู 3D[Side] ใน Pop-up menu ซึ่งจะมีการเปลี่ยนแปลงขึ้นที่ View นี้เช่นกัน User จึงทำการวาดภาพ Side View ลงไปได้ ตัวอย่างแสดงดังรูปที่ 18 และที่ Status line จะมีข้อความต่อไปนี้ปรากฏ "***SIDE VIEW*** When complete pick menu Complete "

4.4.เมื่อวาดภาพครบทั้ง 3 View แล้วให้เลื่อนตำแหน่งเคอร์เซอร์ไป Pick ที่เมนู3D[Complete] ที่ Pop-up menu หลังจากนี้โปรแกรมจะทำการสร้างภาพ 3 มิติที่ประกอบขึ้นจากภาพ 2 มิติทั้ง 3 View ที่วาดไปแล้วนั้น ดังแสดงในรูปที่ 19 เมื่อได้ภาพ 3 มิติแล้วที่ Command line จะมีข้อความต่อไปนี้ปรากฏ " There are any thing wrong in any view (y/n): " ซึ่งข้อความนี้จะมีความหมายว่าภาพ 3 มิติเกิดขึ้นนี้เป็นภาพที่ต้องการหรือไม่ ซึ่งถ้าหากภาพ 3 มิติที่ได้ยังมีรายละเอียดภาพอีก เช่น เป็นภาพที่ถูกเจาะก็จะต้องทำการเพิ่มรายละเอียดของภาพ 2 มิติ ซึ่ง User

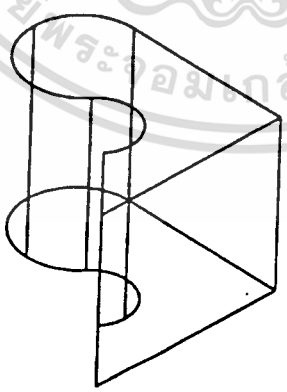
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 16 แสดงตัวอย่างภาพที่วาดลงบน Top View (เมนู TOP)

รูปที่ 17 แสดงตัวอย่างภาพที่วาดลงบน Front View (เมนู FRONT)

รูปที่ 18 แสดงตัวอย่างภาพที่วาดลงบน Side View (เมนู SIDE)



รูปที่ 19 แสดงภาพที่ได้หลังจากเลือกเมนู COMPLETE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องตอบว่า "y" และที่ Status line จะมีข้อความต่อไปนี้ปรากฏ " Pick menu Top2 " หลังจากนี้จะเป็นการวาดภาพรายละเอียด ในแต่ละวิว โดย User จะต้องรู้ว่าส่วนไหนของภาพจะเป็นส่วนที่ถูกเจาะ ซึ่ง User ต้องพื้นฐานทางด้านการเขียนแบบมาบ้าง โดยวาดทีละส่วนที่ถูกเจาะเท่า นั้น (1 loop)

4.5.ให้ User เลือกเมนู3D[Top2] ใน Pop-up menu แล้วทำการวาดรายละเอียดเหล่านั้น ตัวอย่างแสดงดังรูปที่ 20 ส่วนที่ Status line จะมีข้อความต่อไปนี้ปรากฏ "***TOP VIEW *** when complete pick menu Front2 "

4.6.ให้ User ทำการเลือกเมนู 3D[Front2] ใน Pop-up menu แล้วทำการวาดรายละเอียดเหล่านั้นตัวอย่างแสดงดังรูปที่ 21 ส่วนที่ Status line จะมีข้อความต่อไปนี้ปรากฏ " ***FRONT VIEW *** when complete pick menu Side2 "

4.7.ให้ เลือกเมนู 3D[Side2] ใน Pop-up menu แล้วทำการวาดรายละเอียดเหล่านั้น ตัวอย่างแสดงดังรูปที่ 22 ส่วนที่ Status line จะมีข้อความต่อไปนี้ปรากฏ "*** SIDE VIEW *** when complete pick menu Complet2 "

4.8.เมื่อทำการแก้ไขภาพทั้งหมดแล้วให้ผู้ใช้เลือกเมนู 3D[Complet2] ที่ Pop-up menu ในขั้นตอนนี้โปรแกรมจะทำการเปลี่ยนแปลงภาพ 3 มิติจากภาพ 2 มิติที่ทำการเปลี่ยนแปลงแล้ว ผลที่ได้แสดงดังรูปที่ 23 ส่วนที่ Command line จะมีข้อความปรากฏคือ "There are any thing wrong in any view (y/n):" ซึ่งข้อความนี้จะถาม User เช่นเดียวกับในขั้นตอนที่ 4 ถ้าหากในภาพ 3 มิติที่ต้องการมีส่วนที่ถูกเจาะอีกก็ให้ตอบ "y" และทำเช่นเดียวกับขั้นตอนที่ 6-8 ถ้าภาพที่ได้เป็นภาพที่ต้องการแล้วให้ตอบ "n"

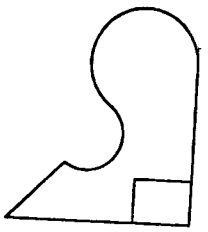
ซึ่งขั้นตอนต่อไปจะเป็นการทำให้ภาพ 3 มิติที่ได้เป็น Drawing Space ซึ่งจะต้องทำตามขั้นตอนต่อไปนี้ หลังจากตอบ "n" แล้วที่ Status line จะมีข้อความต่อไปนี้ปรากฏ " Pick left top view and press menu TopB "

4.9.ให้ ผู้ใช้ทำการเลือกเมนู 3D[Paper] จาก Pop-up menu หลังจากนั้นเครื่องจะทำการสร้างรูปของแบบแปลนซึ่งประกอบด้วย ภาพฉายด้านบน, ภาพฉายด้านข้าง ,ภาพฉายด้านหน้า และภาพไอโซเมตริกที่ได้ ดังแสดงในรูปที่ 24 Status line จะมีข้อความ ต่อไปนี้ปรากฏ "Return to 3D model by pressmenu return " หมายความว่า ถ้าหาก User ต้องการให้หน้าจอกลับไปปรากฏที่หน้าจอเดิมให้เลือกเมนู 3D[Return] ที่ Pop-up menu

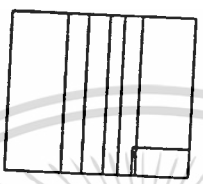
4.10 ถ้าต้องการให้ได้ชิ้นงานออกมาให้ใช้คำสั่ง Plot เพื่อ Plot Drawing space ให้ออกมาเป็นชิ้นงานได้

4.11.ถ้าหากต้องการเห็นภาพ 3 มิติที่ได้มีการ Anotate เกิดขึ้นก็สามารถทำได้โดยการเลือกเมนู 3D[Anotate] ที่ Pop-up menu

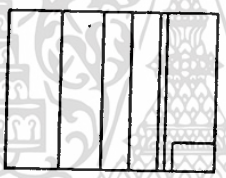
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



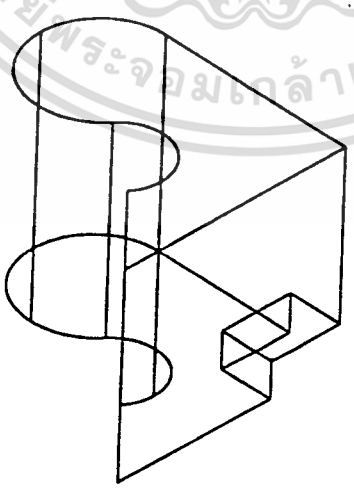
รูปที่ 20 แสดงตัวอย่างภาพที่วาดลงบน Top View (เมนู TOP2)



รูปที่ 21 แสดงตัวอย่างภาพที่วาดลงบน Front View (เมนู FRONT2)

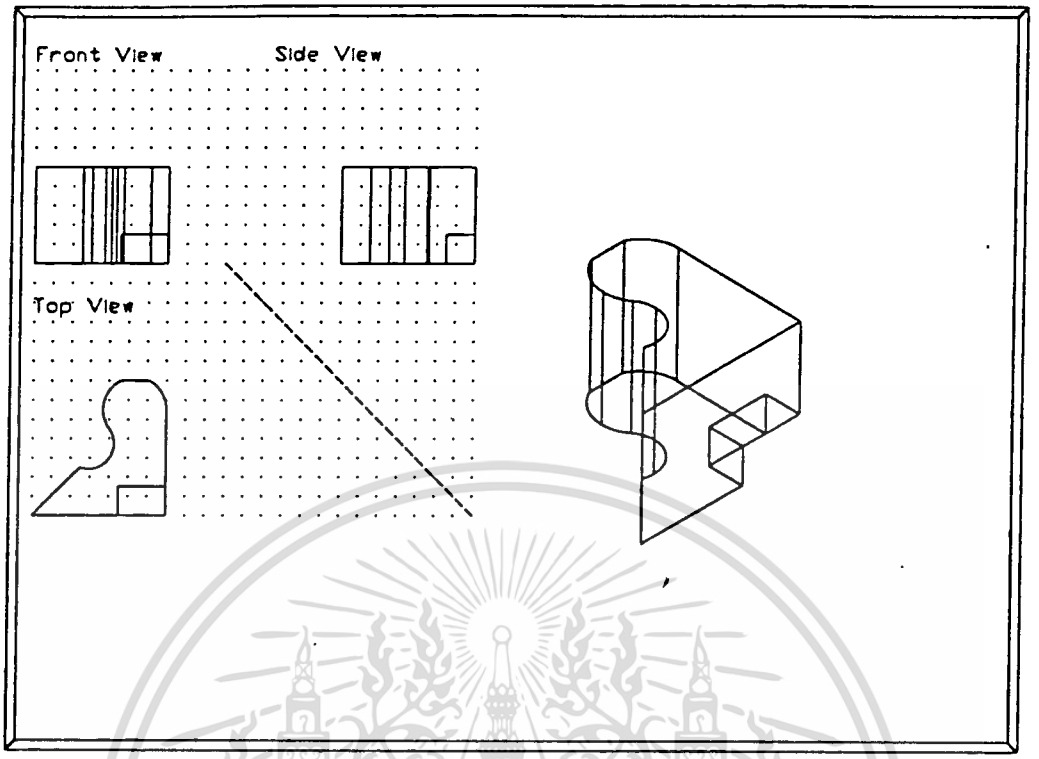


รูปที่ 22 แสดงตัวอย่างภาพที่วาดลงบน Side View (เมนู SIDE2)

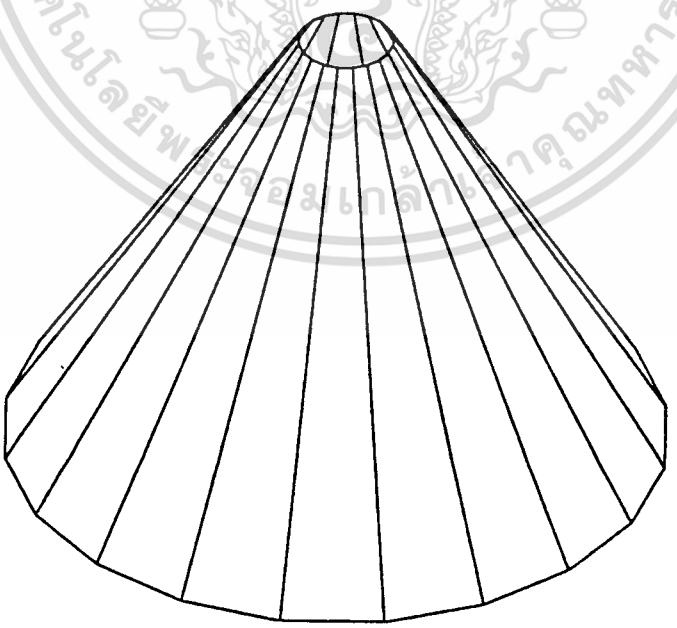


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 23 แสดงภาพที่ได้หลังจากเลือกเมนู COMPLETE2

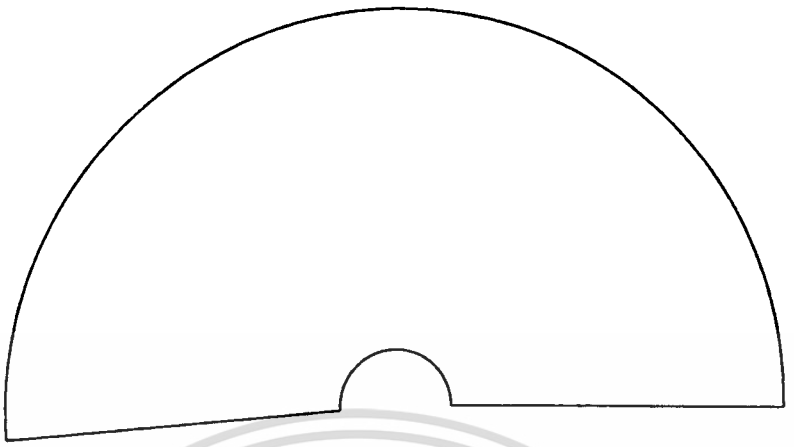


รูปที่ 24 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER

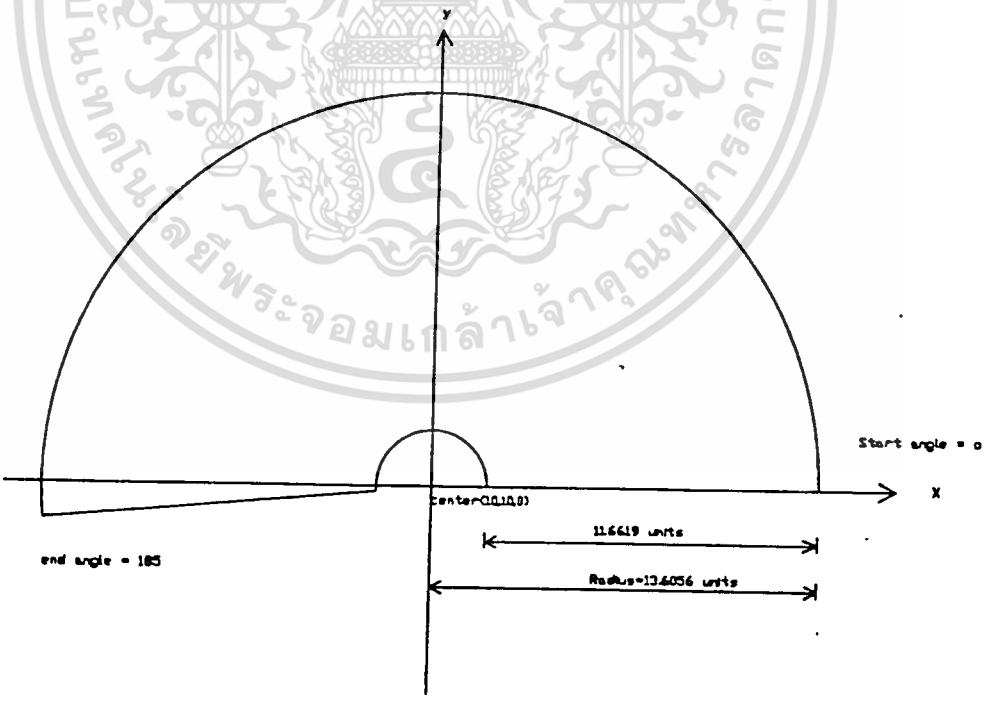


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 25 แสดงรูปภาพที่ได้จากการทำคำสั่ง 3DCONE ซึ่งได้ลบเส้นที่ถูกบังทิ้งแล้ว



รูปที่ 26 แสดงรูปภาพของภาพคลี่ของทรงกรวยที่ได้จากการทำคำสั่ง CONEPROJ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 27 แสดงค่าต่างๆของภาพคลี่หลังจากใช้คำสั่ง LIST ตรวจสอบ

4.2 การทดลองโปรแกรมสร้างทรงกรวย 3 มิติและภาพคลี่ทรงกรวยนั้น

4.2.1 เลือกเมนู File[New...] เพื่อทำการเปิดดรออิง อิดิตเตอร์ใหม่

4.2.2 เลือกเมนู 3D[3D Cone] เพื่อทำการเรียกคำสั่ง 3D Cone ซึ่งเขียนไว้ในโปรแกรม ACAD.LSP เครื่องจะแสดง

command:3dcone

เครื่องจะถามว่า

Center point : 10,10 (ให้ผู้ใช้ใช้เมาส์เลือกโคออร์ดิเนต 10,10) เป็นตัวอย่างและเครื่องจะถามว่า

Base Radius : 10,17 (ให้ผู้ใช้เลือกโคออร์ดิเนต 10,17)เป็นตัวอย่าง แล้วเครื่องจะถามว่า

Top Radius : 10,11 (ให้ผู้ใช้เลือกโคออร์ดิเนต 10,11)เป็นตัวอย่าง แล้วเครื่องจะถามว่า

High of Cone : 10 (ให้ผู้ใช้พิมพ์ค่า 10)เป็นตัวอย่างแล้วเครื่องจะถามว่า

Number of Segment(15) : 20 (ให้ผู้ใช้พิมพ์ค่า 20)เป็นตัวอย่าง

หลังจากนั้นเครื่องจะทำการคำนวณแล้วสร้างรูปทรงกรวยแสดงออกมาทางหน้าจะ ดังรูปที่ 25 เราสามารถทำการเลือกเมนู Render[Shade2] เพื่อทำการซ่อนเส้นที่ถูกบังได้

4.2.3 เลือกเมนู 3D[ConeProj] เพื่อทำการเรียกคำสั่ง Coneproj ซึ่งเขียนไว้ในโปรแกรม ACAD.LSP เครื่องจะทำการคำนวณและสร้างภาพคลี่ของทรงกรวยที่เพิ่งสร้างเสร็จนั้นออกมาดังรูปที่ 26

เราสามารถทำการหาจุดต่างๆของภาพคลี่นี้ได้ โดยใช้คำสั่ง LIST ซึ่งเป็นคำสั่งมาตรฐานของออโตแคด ดังแสดงในรูปที่ 27 เราสามารถทดลองพล็อตรูปออกมาประกอบเป็นรูปทรงกรวยได้จริง

4.3 ผลการทดลองโปรแกรมการสร้างบ้าน 3 มิติจากแบบแปลน 2 มิติ

โปรแกรมนี้จะให้ผู้ใช้ทำการสร้างแบบแปลนบ้านในระนาบ 2 มิติด้วยตนเองและทำการตกแต่งเฟอร์นิเจอร์ต่างๆตามความต้องการในตำแหน่งต่างๆได้ หลังจากนั้นสามารถมองมุมต่างๆของบ้านในระนาบ 3 มิติได้ โดยเลือกมุมมองจากแบบแปลน 2 มิตินั้น

ขั้นตอนการทดลองและผลการทดลอง

4.3.1 เลือกเมนู File[New...] เพื่อทำการเปิดดรออิง อิดิตเตอร์ใหม่

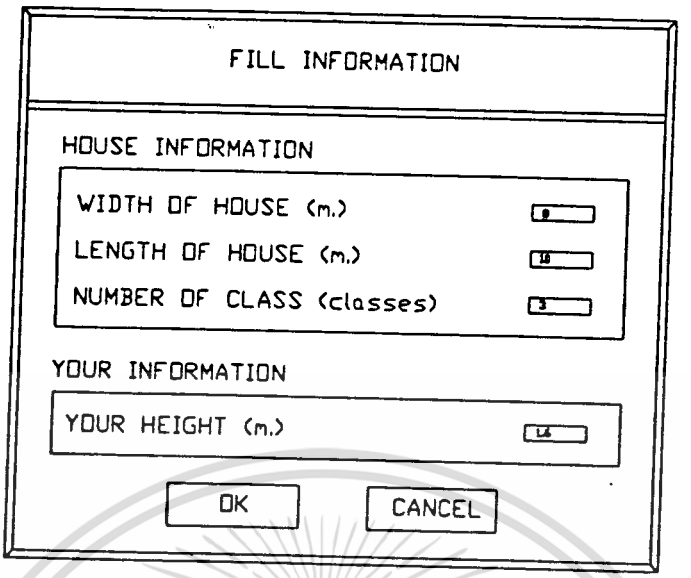
หรือ อาจเลือกเมนู File[Open...] เพื่อทำการเรียกไฟล์ของภาพแปลนบ้านที่ได้ทำการรสร้างเก็บไว้ ก็ได้ 4.3.2 เลือกเมนู House[Begin] เครื่องจะแสดงไดอะลอกบล็อก ดังรูปที่ 28 ทำการป้อนค่าในไดอะลอกบล็อกนั้นตามตัวอย่าง คือ

Wide of House(m.) : 8 (กด Tab)

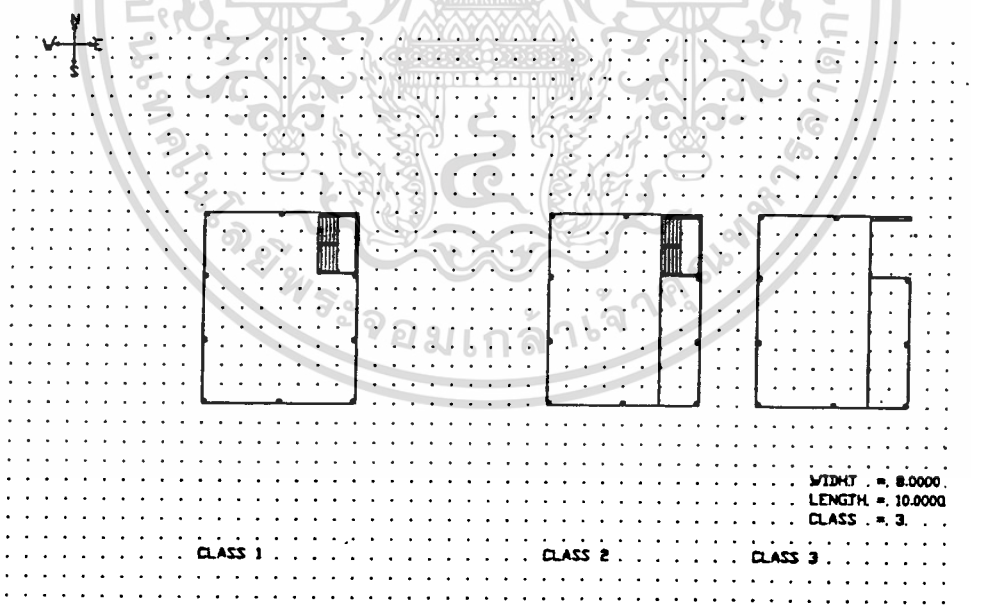
Lenght of House(m.) : 10 (กด Tab)

Number of Class(Classes) : 3 (กด Tab)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 28 แสดงไดอะลอกบล็อกที่ได้จากการเลือกเมนู House[Begin]



รูปที่ 29 แสดงรูปภาพที่ได้หลังจากการเลือกเมนู House[Step1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Your high(m.) : 1.6 (กด Tab) มายังปุ่ม OK

กด Space Bar ทำการเลือกปุ่ม OK หลังจากนั้นเครื่องจะแสดงข้อความ

" IF NEW HOUSE... PICK MENU [STEP1] TO CHOOSE COLOR "

/* หมายความว่าถ้าผู้ใช้ ใช้คำสั่ง Open เปิดภาพของแบบแปลนที่ได้ถูกสร้างมาก่อนแล้ว ก็ไม่ต้องเลือกเมนู นี้ */

4.3.3 เลือกเมนู House[Step1] เครื่องจะแสดงไดอะล็อกบล็อก SELECT COLOR OF WALL (ซึ่งเป็นส่วนของเมนูรูปภาพ ที่ได้เขียนไว้บนโปรแกรม ACAD.MNU) ให้ผู้ใช้เลือกสีของกำแพงที่ต้องการโดยการใช้เมาส์เลือกรูปของสีนั้นตามด้วยการกดปุ่ม OK หรืออาจจะทำการเลือกโดยการกดเมาส์ที่รูปของสีนั้นอย่างรวดเร็ว 2 ครั้ง(Double Click)

หลังจากนั้นเครื่องจะแสดงไดอะลอกบล็อก SELECT COLOR OF FLOOR & ROOF (ซึ่งเป็นส่วนของเมนูรูปภาพที่ได้เขียนไว้บนโปรแกรม ACAD.MNU) ให้ผู้ใช้ทำการเลือกสีของพื้นที่ต้องการ

หลังจากนั้นเครื่องจะแสดงไดอะบล็อก SELECT STYLE OF ROOF (ซึ่งเป็นส่วนของเมนูรูปภาพที่ได้ทำการเขียนไว้บนโปรแกรม ACAD.MNU) ให้ผู้ใช้ทำการเลือกแบบของหลังคาที่ต้องการ

หลังจากนั้นเครื่องจะทำการคำนวณ และสร้างแบบแปลน 2 มิติ ซึ่งประกอบด้วย พื้นบ้าน,เสาบ้าน,บันได,หลังคาบ้าน (ซึ่งชั้นภาพของหลังคาบ้านนี้ได้ถูกปิดอยู่จึงมองไม่เห็น),และข้อความแสดงชั้นของบ้านและขนาดของบ้าน ดังแสดงในรูปที่ 29

4.3.4 ผู้ใช้สามารถดูตัวบ้านที่ยังไม่ได้ประกอบเฟอร์นิเจอร์ใด ๆ ได้โดยการเลือกเมนู View[View House] เครื่องจะทำการคำนวณและแสดงตัวบ้านในระนาบ 3 มิติ ให้ผู้ใช้เลือกขนาดการมองตามต้องการ โดยการใช้เมาส์ลากและกด

4.3.5 ทำการกลับสู่แบบแปลน 2 มิติ โดยการเลือกเมนู View[PlanView] แบบแปลนบ้าน 2 มิติจะปรากฏดังเดิม

4.3.6 ทำการเดิมเฟอร์นิเจอร์ต่าง ๆ ของบ้านตามลำดับดังนี้

/* ในระหว่างที่ผู้ใช้ทำการเดิมเฟอร์นิเจอร์เหล่านี้ ผู้ใช้สามารถเลือกเมนู House[Open->] [Grid] และ

House[Close->] [Grid] เพื่อทำการเปิดและปิดตำแหน่งอ้างอิงต่าง ๆได้ */

1. ประตู

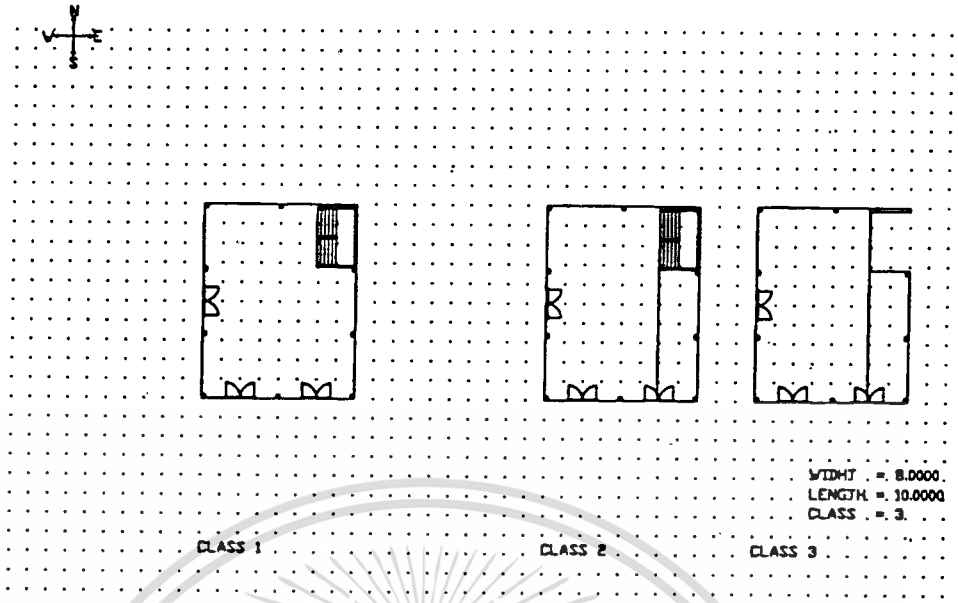
ทำการเลือกเมนู House[Furniture ->] [Door ->] [North] เครื่องจะแสดงไดอะลอกบล็อก SELECT DOOR STYLE (ซึ่งเป็นส่วนของเมนูรูปภาพที่ทำการเขียนไว้บนโปรแกรม ACAD.MNU) ให้ผู้ใช้ทำการเลือกแบบของประตูตามที่ต้องการ หลังจากนั้นเครื่องจะถามว่า

"POSITION...(upper right) :" ให้ผู้ใช้เลือกตำแหน่งมุมขวาบนของตำแหน่งของประตูที่ต้องการ

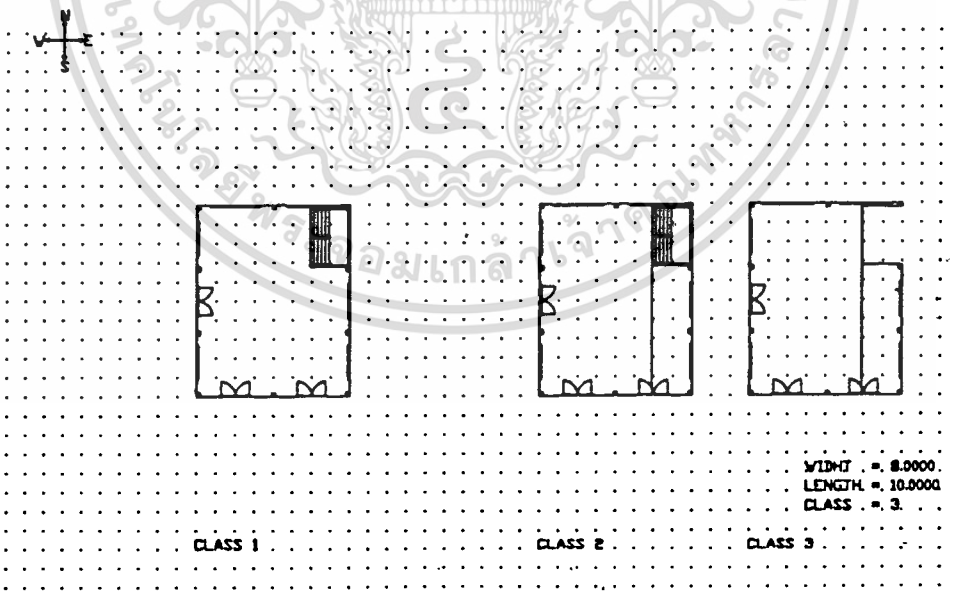
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วางโดยมีจุดสอดแทรกของประตูอยู่ที่มุมขวาบนของประตู
และผู้ใช้สามารถทำการเพิ่มประตูได้โดยการกดเมนู
House[Furniture ->] [Door ->] [South]
House[Furniture ->] [Door ->] [East]
House[Furniture ->] [Door ->] [West] ดังแสดงในรูปที่ 30
2. หน้าต่าง
ทำการเลือกเมนู House[Furniture ->] [Window ->] [North] เครื่องจะแสดงไดอะลอกบล็อก SELECT WINDOW STYLE (ซึ่งเป็นส่วนของเมนูรูปภาพที่ทำการเขียนไว้บนโปรแกรม ACAD.MNU) ให้ผู้ใช้ทำการเลือกแบบของหน้าต่างตามที่ต้องการ หลังจากนั้นเครื่องจะถามว่า "POSITION...(upper right) :" ให้ผู้ใช้เลือกตำแหน่งมุมขวาบนของตำแหน่งของหน้าต่างที่ต้องการวางโดยมีจุดสอดแทรกของหน้าต่างอยู่ที่มุมขวาบนของหน้าต่าง
และผู้ใช้สามารถทำการเพิ่มหน้าต่างได้โดยการกดเมนู
House[Furniture ->] [Window ->] [South]
House[Furniture ->] [Window ->] [East]
House[Furniture ->] [Window ->] [West] ดังแสดงในรูปที่ 31
3. กำแพง
ทำการเลือกเมนู House[Wall ->] [North] หลังจากนั้นเครื่องจะถามว่า
"START POINT OF WALL :" ให้ผู้ใช้ใช้เมาส์ทำการเลือกตำแหน่งจุดเริ่มต้นของกำแพง หลังจากนั้นเครื่องจะถาม "END POINT OF WALL :" ให้ผู้ใช้ทำการใช้เมาส์ทำการเลือกตำแหน่งจุดสิ้นสุดของกำแพง หลังจากนั้นเครื่องจะสร้างเส้นตรงแสดงส่วนของกำแพงให้ ผู้ใช้สามารถทำการเพิ่มกำแพงเข้าไปในแบบแปลนได้ โดยเลือกเมนูดังต่อไปนี้คือ House[Wall ->] [South]
House[Wall ->] [East]
House[Wall ->] [West] ดังแสดงในรูปที่ 32
ทดลองตรวจสอบตัวบ้านที่ได้ทำการเดิมกำแพงประตูและหน้าต่างแล้ว โดยเลือกเมนู View[ViewHouse] จะได้ผลแสดงดังรูปที่ 33
ทำการกลับสู่แบบแปลน 2 มิติ โดยเลือกเมนู View[PlanView]
4. เฟอร์นิเจอร์อื่นๆ
ทำการเพิ่มเฟอร์นิเจอร์ประเภทต่างๆ ได้โดยการเลือกเมนูดังต่อไปนี้
House[Furniture->] [Bed_Chair_Toilet] :เพื่อทำการเลือกรูปแบบของ เตียง,เก้าอี้,และเครื่องสุขภัณฑ์ในห้องน้ำ
House[Furniture->] [Cabinet] : เพื่อทำการเลือกรูปแบบของตู้ต่างๆ
House[Furniture->] [Kitchen] : เพื่อทำการเลือกรูปแบบของครัวและอุปกรณ์ภายในครัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

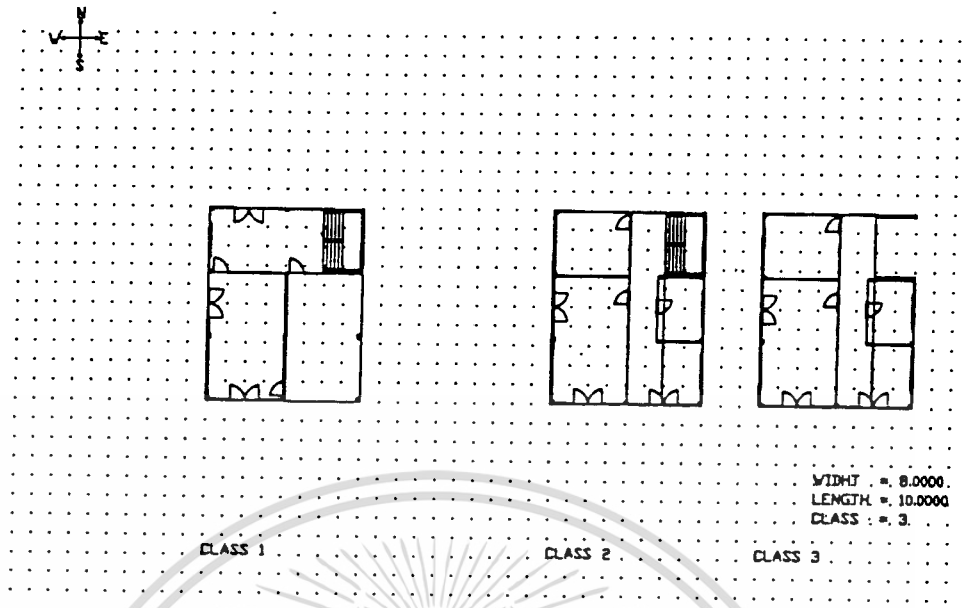


รูปที่ 30 แสดงรูปภาพที่ได้หลังการเติมประตูเข้าไปในแบบแปลน



รูปที่ 31 แสดงรูปภาพที่ได้หลังจากการเติมหน้าต่างเข้าไปในแบบแปลน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 32 แสดงรูปภาพที่ได้หลังจากการเติมกำแพงเข้าไปในแบบแปลน

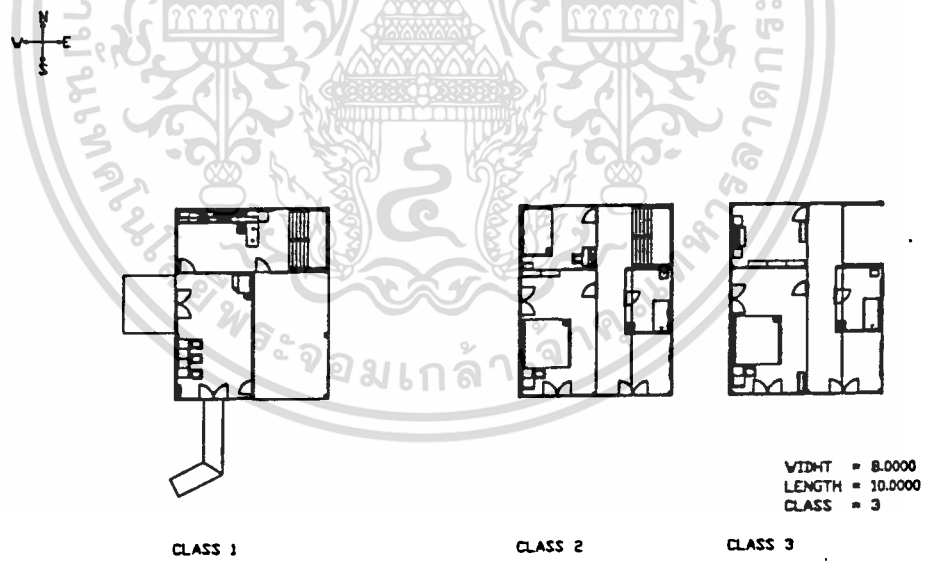
รูปที่ 33 แสดงรูปตัวบ้าน 3 มิติหลังจากเติมกำแพงประตูและหน้าต่างแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

House[Furniture->] [Shelf] : เพื่อทำการเลือกรูปแบบของชั้นแบบต่างๆ
House[Furniture->] [Table_Sink] : เพื่อทำการเลือกรูปแบบของโต๊ะและที่ล้างจาน
โดยเมื่อผู้ใช้เลือกเมนูเหล่านี้แล้วเครื่องจะแสดงไดอะลอกบล็อกของเมนูรูปภาพต่างๆ (ซึ่งได้ทำการเขียนไว้ในโปรแกรม ACAD.MNU แล้ว) หลังจากนั้นเครื่องจะถามว่า
"DIRECTION OF FURNITURE (N,S,E,or W) : ให้ผู้ใช้ทำการป้อนตัวอักษร n เพื่อบอกตำแหน่งทิศทางการวางของเฟอร์นิเจอร์ให้ทางด้านหลังของเฟอร์นิเจอร์หันไปทางทิศเหนือ
ให้ผู้ใช้ทำการป้อนตัวอักษร s เพื่อบอกตำแหน่งทิศทางการวางของเฟอร์นิเจอร์ให้ทางด้านหลังของเฟอร์นิเจอร์หันไปทางทิศใต้
ให้ผู้ใช้ทำการป้อนตัวอักษร e เพื่อบอกตำแหน่งทิศทางการวางของเฟอร์นิเจอร์ให้ทางด้านหลังของเฟอร์นิเจอร์หันไปทางทิศตะวันออก ให้ผู้ใช้ทำการป้อนตัวอักษร w เพื่อบอกตำแหน่งทิศทางการวางของเฟอร์นิเจอร์ให้ทางด้านหลังของเฟอร์นิเจอร์หันไปทางทิศตะวันตก
หลังจากนั้นเครื่องจะถามว่า
" POSITION OF FURNITURE :" ให้ผู้ใช้ใช้เมาส์เลือกตำแหน่งที่ตั้งเฟอร์นิเจอร์ตามต้องการโดยเมื่อทำการเลือกจุดแล้ว ตำแหน่งของเฟอร์นิเจอร์จะยังไม่ถูกกำหนดตายตัว ผู้ใช้สามารถเคลื่อนย้ายตำแหน่งเฟอรร์นิเจอร์ได้ เมื่อได้ตำแหน่งที่ต้องการแล้วให้ผู้ใช้กดเมาส์หนึ่งครั้ง ดังแสดงในรูปที่ 34
5. ทางเดินเท้า
ให้ผู้ใช้ทำการเลือกเมนู House[Path] หลังจากนั้นเครื่องจะถามว่า
"WIDE OF PATH(m.) :" ให้ผู้ใช้ใส่ค่าความกว้างของทางเดินเท้าที่ต้องการ เช่น 1
หลังจากนั้นเครื่องจะถามว่า
"START POINT :" ให้ผู้ใช้ใช้เมาส์เลือกตำแหน่งจุดเริ่มต้นของทางเดินเท้าที่ต้องการ หลังจากนั้นเครื่องจะถามว่า "ANOTHER POINT OF LINE "ให้ผู้ใช้ทำการเลือกตำแหน่งต่อไปของทางเดินเท้าที่ต้องการ หลังจากนั้นเครื่องจะถามว่า "ANOTHER POINT OF LINE "ต่อไปเรื่อยๆ ผู้ใช้สามารถเปลี่ยนทิศทางของทางเดินเท้านี้ได้ เมื่อถึงจุดสิ้นสุดของทางเดินเท้าให้ผู้ใช้กด "ENTER" เครื่องจะทำการคำนวณและสร้างทางเดินเท้า ตัวอย่างแสดงดังรูปที่ 35
6. ระเบียง
ให้ผู้ใช้เลือกเมนู House[Veranda] เครื่องจะถามว่า
"START POINT OF VELANDA :" ให้ผู้ใช้ทำการเลือกจุดปลายข้างหนึ่งของระเบียง ซึ่งมีตำแหน่งติดกับตัวบ้าน หลังจากนั้นเครื่องจะถามว่า
"END POINT OF VELANDA :" ให้ผู้ใช้ทำการเลือกจุดปลายอีกข้างหนึ่งของระเบียง ซึ่งมีตำแหน่งติดกับตัวบ้าน หลังจากนั้นเครื่องจะถามว่า
"WIDTH POINT OF VELANDA :" ให้ผู้ใช้ทำการเลือกตำแหน่งจุดปลายของระเบียง ซึ่งอยู่ห่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 34 แสดงรูปภาพที่ได้จากการเติมเฟอร์นิเจอร์ต่างๆเข้าไปในแบบแปลน



รูปที่ 35 แสดงรูปภาพที่ได้จากการเติมทางเดินเท้าเข้าไปในแบบแปลน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออกไปจากตัวบ้าน หลังจากนั้นเครื่องจะทำการสร้างส่วนที่เป็นระเบียงให้ ดังแสดงในรูปที่ 36
ให้ผู้ใช้ตรวจสอบดูรูปตัวบ้าน โดยเลือกเมนู View[ViiewHouse] ผลที่ได้ แสดงดังรูปที่ 37
ให้ผู้ใช้เลือกเมนู View[PlanView] เพื่อกลับสู่แบบแปลน 2 มิติ

7. ต้นไม้

ทำการเลือกเมนู House[Tree] เครื่องจะแสดงไดอะลอกบล็อกของเมนูรูปภาพของต้นไม้ ให้ผู้ใช้ทำการเลือกรูปต้นไม้ที่ต้องการ หลังจากนั้นเครื่องจะถามว่า

"POSITION OF TREE :"ให้ผู้ใช้กำหนดตำแหน่งของต้นไม้ภายนอกรอบๆบริเวณชั้นที่ 1 ตัวอย่างดังแสดงดังรูปที่ 38

ให้ผู้ใช้ตรวจสอบดูรูปตัวบ้าน โดยเลือกเมนู View[ViiewHouse] ผลที่ได้ แสดงดังรูปที่ 39

/*ในระหว่างที่ผู้ใช้อยู่ในแบบแปลน 2 มิตินั้น ผู้ใช้สามารถทำการเคลื่อนย้ายอุปกรณ์ต่างๆได้ โดยใช้คำสั่ง Move ซึ่งเป็นคำสั่งมาตรฐานของออโตแคด */

4.3.7 เมื่อผู้ใช้เลือกเมนู View[ViewHouse] ผู้ใช้สามารถเปลี่ยนมุมมองในแนวอื่นๆได้ โดยเลือกเมนู

View[Vpoint->] [0,0,0] ดังแสดงในรูปที่ 40

View[Vpoint->] [1,0,0] ดังแสดงในรูปที่ 41

View[Vpoint->] [0,1,0] ดังแสดงในรูปที่ 42

View[Vpoint->] [-1,-1,1] ดังแสดงในรูปที่ 43

View[Vpoint->] [1,1,1] ดังแสดงในรูปที่ 44

4.3.8 ถ้าผู้ใช้ต้องการมองตัวบ้านโดยปราศจากกำแพง หรือหลังคาบัง ผู้ใช้สามาถทำการเลือกเมนู View[Open->] [Wall] เพื่อทำการเปิดกำแพงออก

View[Open->] [Roof] เพื่อทำการเปิดหลังคาออก ดังแสดงในรูปที่ 45

หลังจากนั้นผู้ใช้สามารถ ปิดกำแพงและหลังคาตามเดิมได้ โดยเลือกจากเมนู

View[Close->] [Wall] และ เมนู View[Close->] [Roof]

เลือกเมนู View[PlanView] เพื่อกลับสู่แบบแปลน 2 มิติ

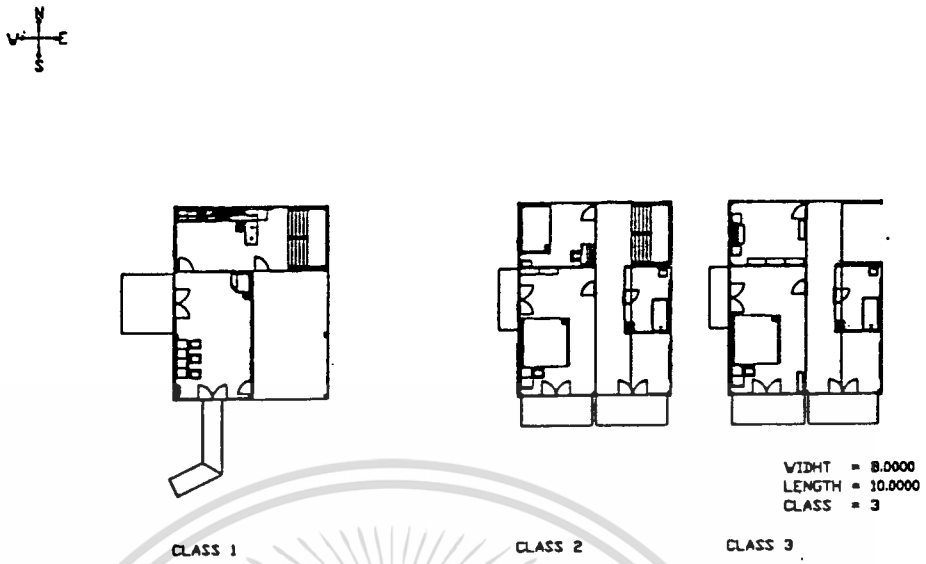
4.3.9 ผู้ใช้สามารถทำการดูมุมมองต่างๆของบริเวณบ้านที่ตำแหน่งและชั้นต่างๆได้ โดยการเลือกเมนู View[See_At] ซึ่งก่อนที่ผู้ใช้จะเลือกเมนูนี้ได้ จะต้องแน่ใจว่าขณะนั้นอยู่ในแบบแปลน 3 มิติอยู่

หลังจากนั้นเครื่องจะถามว่า

"WHAT CLASS YOU STAND :"ให้ผู้ใช้ใส่ค่าชั้นที่เป็นตำแหน่งที่ผู้ใช้อยู่ เช่น 1 แสดงว่า ขณะนั้นผู้ใช้อยู่บนชั้นที่ 1 หลังจากนั้นเครื่องจะถามว่า

"DISTANCE FROM FLOOR TO TARGET POINT: " ให้ผู้ใช้ทำการใส่ค่าของระยะทางระหว่างพื้นชั้นที่ผู้ใช้เลือก กับเป้าหมายที่ผู้ใช้ต้องการมอง เช่น 1 หมายถึง ผู้ใช้ต้องการมองวัตถุอยู่ห่าง

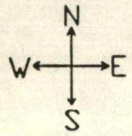
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

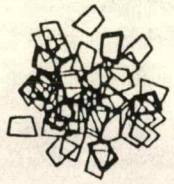


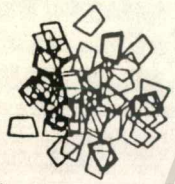
รูปที่ 36 แสดงรูปภาพที่ได้จากการเดิมระเบียงเข้าไปในแบบแปลน

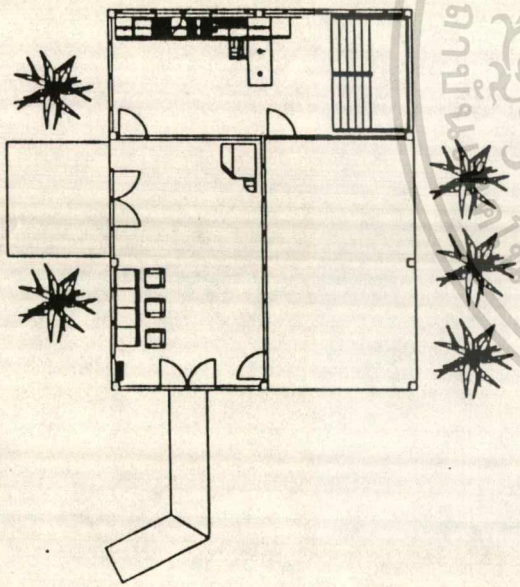
รูปที่ 37 แสดงรูปตัวบ้าน 3 มิติหลังจากเดิมเฟอร์นิเจอร์,ทางเดินเท้า,และระเบียง แล้ว

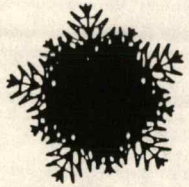
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

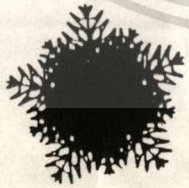


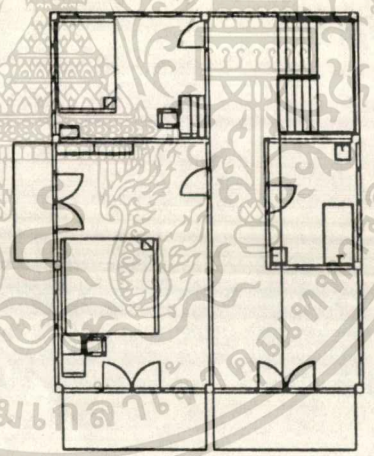


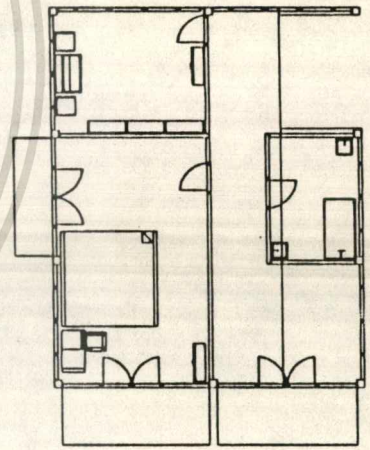












WIDHT = 8.0000
LENGTH = 10.0000
CLASS = 3

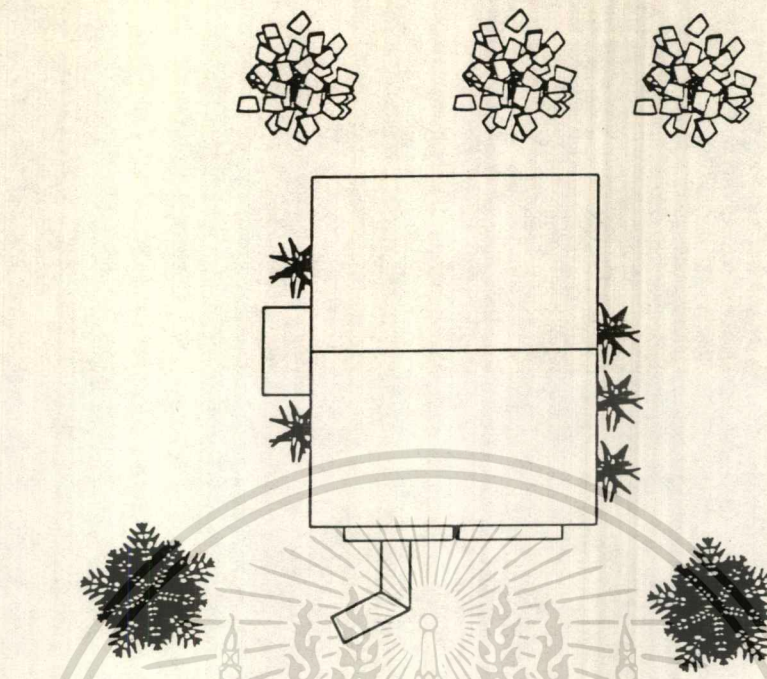
CLASS 1

CLASS 2

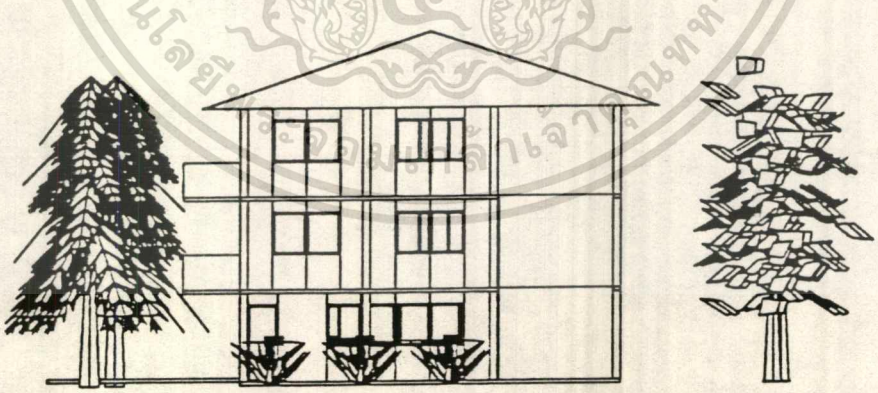
CLASS 3

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย
ห้องสมุดไฮเดียพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

รูปที่ 39 แสดงรูปตัวบ้าน 3 มิติหลังจากเติมต้นไม้เข้าไปในแบบแปลนแล้ว

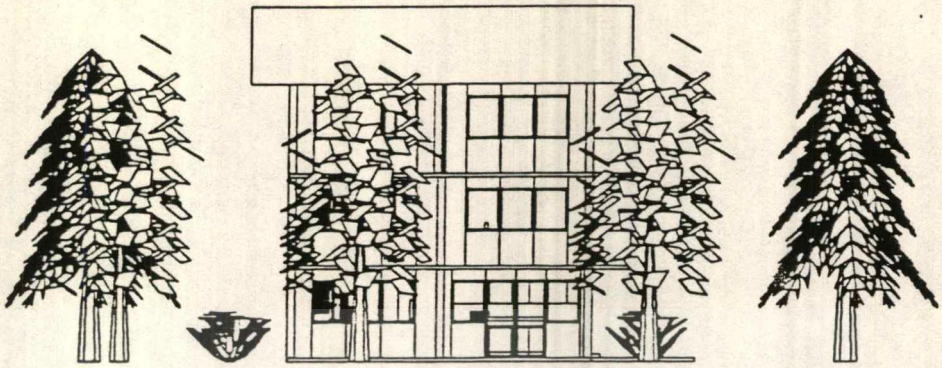


รูปที่ 40 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 0,0,0



รูปที่ 41 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 0,1,0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

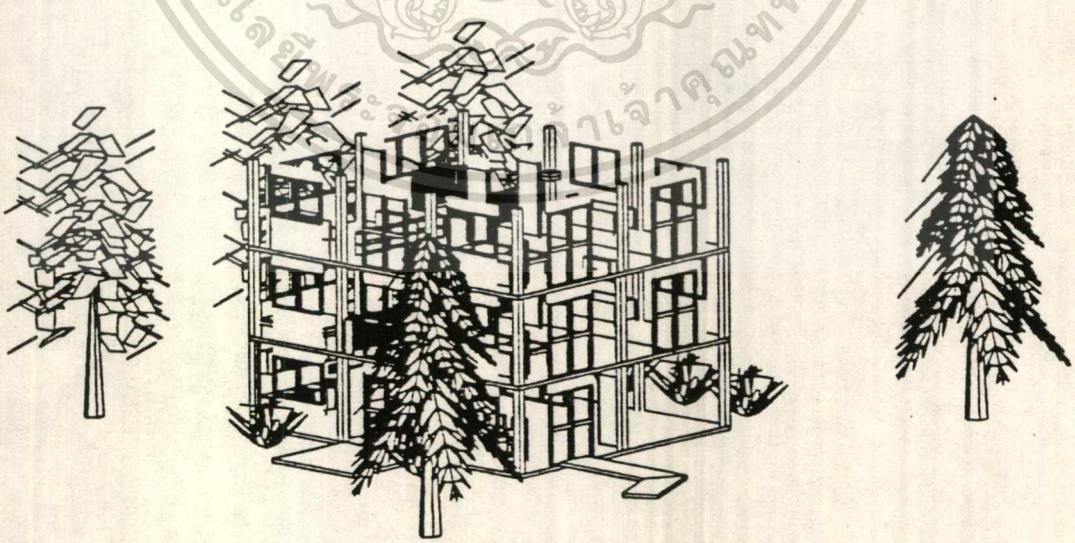


รูปที่ 42 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 1,0,1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 43 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ -1,-1,1

รูปที่ 44 แสดงรูปภาพของการมองบ้านในแนวเวกเตอร์ 1,1,1

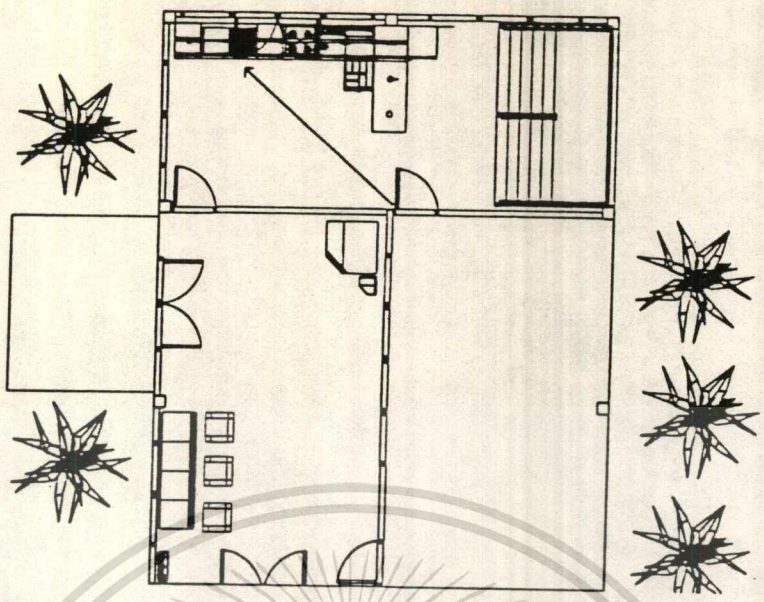


รูปที่ 45 แสดงรูปตัวบ้าน 3 มิติหลังจากทำการเปิดกำแพงและหลังคาออก

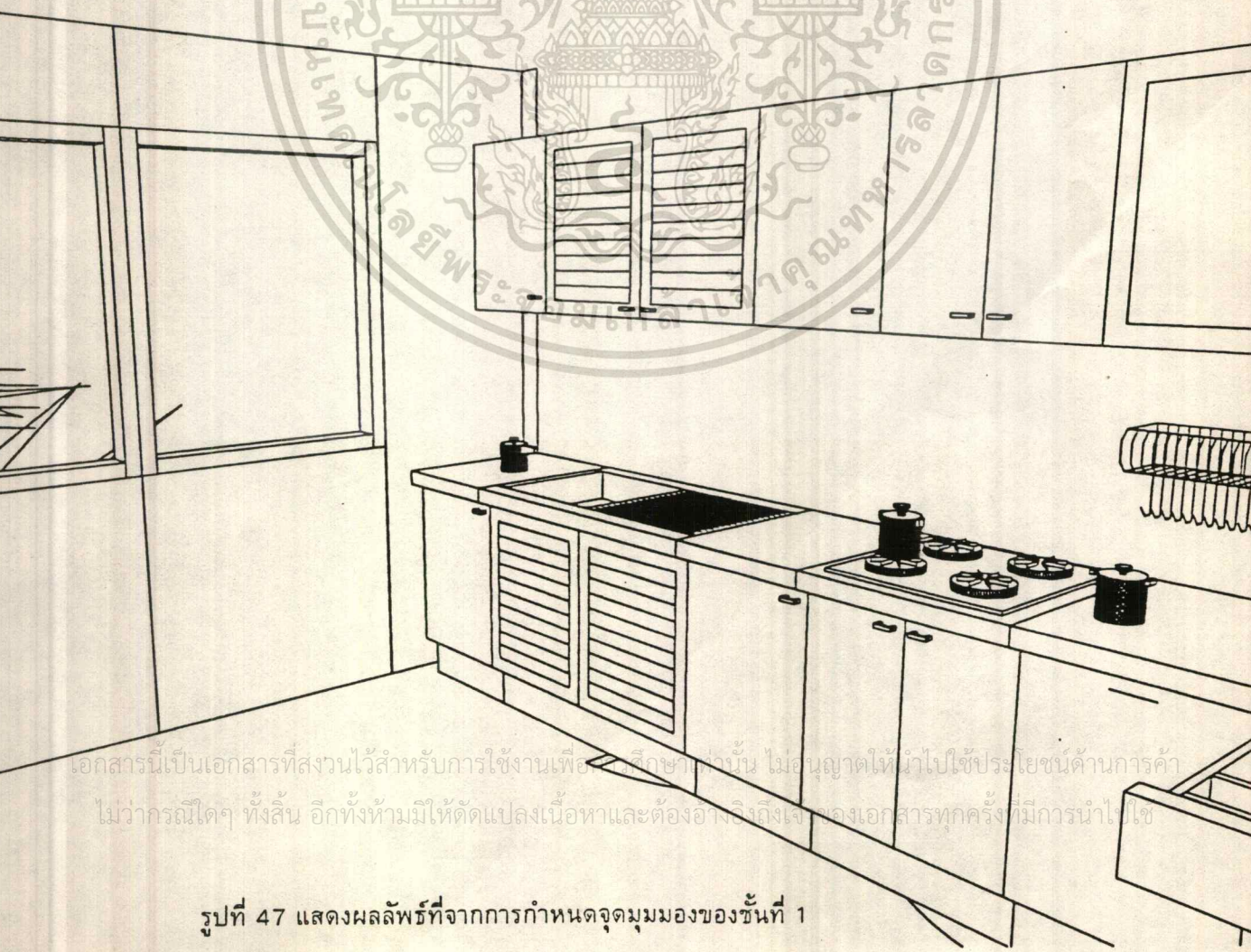
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากพื้นชั้นที่ 1 ขึ้นไปเป็นระยะทาง 1 เมตร หลังจากนั้น เครื่องจะถามว่า
"WHERE IS YOUR POSITION :" ให้ผู้ใช้ใช้เมาส์เลือกตำแหน่งที่ผู้ใช้ต้องการยืนอยู่ในบริเวณชั้นที่ผู้ใช้เลือก หลังจากนั้นเครื่องจะถามว่า
"WHERE S TARGET POINT :" ให้ผู้ใช้ใช้เมาส์เลือกตำแหน่งเป้าหมายในการมอง .
ตัวอย่างดังรูปที่ 46 เป็นการกำหนดว่า ผู้ใช้ต้องการดูมุมมองของห้องครัวบริเวณชั้นที่ 1 โดยผู้ใช้อยู่บริเวณประตูมองไปทางห้องครัว ซึ่งเป็นเป้าหมาย ซึ่งอยู่ห่างจากพื้นดิน 1 เมตร หลังจากนั้นเครื่องจะทำการคำนวณและแสดงผลรับ ของมุมมองที่ผู้ใช้เลือกดังแสดงในรูปที่ 47
ตัวอย่างต่อๆไปดังเช่น การกำหนดมุมมองในรูปที่ 48 โดยกำหนดตำแหน่งชั้นที่ผู้ใช้อยู่คือ ชั้นที่ 2 บริเวณห้องนอน โดยเป้าหมายอยู่ห่างจากพื้นห้องขึ้นไป 0.8 เมตร จะมองเห็นภาพดังรูปที่ 49
และการกำหนดมุมมองในรูปที่ 50 โดยกำหนดตำแหน่งชั้นที่ผู้ใช้อยู่คือ ชั้นที่ 3 บริเวณบันได โดยเป้าหมายอยู่ที่พื้น (หมายถึงอยู่ห่างจากพื้นชั้นที่ 3 ศูนย์เมตร)จะมองเห็นภาพดังแสดงในรูปที่ 51
/* ผู้ใช้สามารถอยู่บริเวณสนามรอบบ้านได้ โดยกำหนดให้ชั้นที่ผู้ใช้ยืนมีค่าเท่ากับ 0 และผู้ใช้สามารถมองเป้าหมายที่อยู่ต่ำกว่าพื้นได้ โดยกาป้อนค่าติดลบ เช่น -1 เมตร ซึ่งผลที่ได้จะมองเห็นพื้นนั่นเอง */
4.3.10 เมื่อผู้ใช้ทำการเลือกมุมมองที่ต้องการเลือกแล้วนั้น จากการเลือกเมนู View[See_At] ผู้ใช้สามารถหมุนดูวัตถุรอบๆตัว หรือหมุนภาพที่มองเห็นได้ โดยเลือกจากเมนู
View[Twist]
View[Camera1]
View[Camera11]
View[Camera2]
View[Target1]
View[Target2]
View[Dist1]
View[Dist2]
View[Zoomin]
View[Zoomout]
ตัวอย่าง เช่น เมื่อผู้ใช้ทำการเลือกเมนู View[See_At] โดยเลือกตำแหน่งของผู้ใช้อยู่ที่สนามหญ้า (ชั้นที่อยู่คือ 0) และมองไปยังที่เป้าหมาย ซึ่งอยู่ห่างจากพื้นดินขึ้นมา 1 เมตร ดังแสดงในรูปที่ 52
ผลลัพธ์ของมุมมองแสดงดังรูปที่ 53 หลังจากนั้นให้ผู้ใช้เลือกเมนู View[Target2] ซึ่งหมายถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

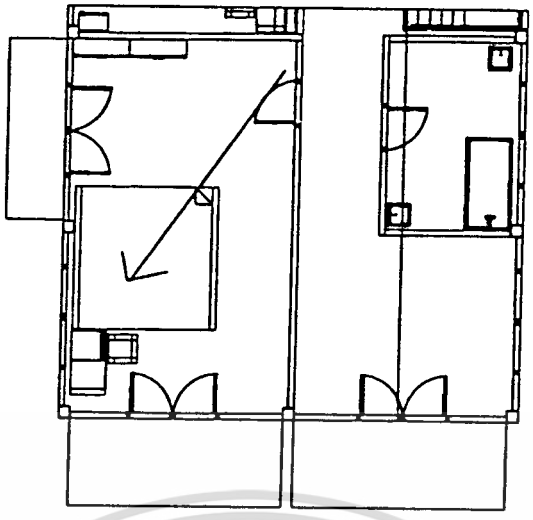


รูปที่ 46 แสดงการกำหนดจุดมุมมองบริเวณชั้นที่ 1 บริเวณห้องครัว

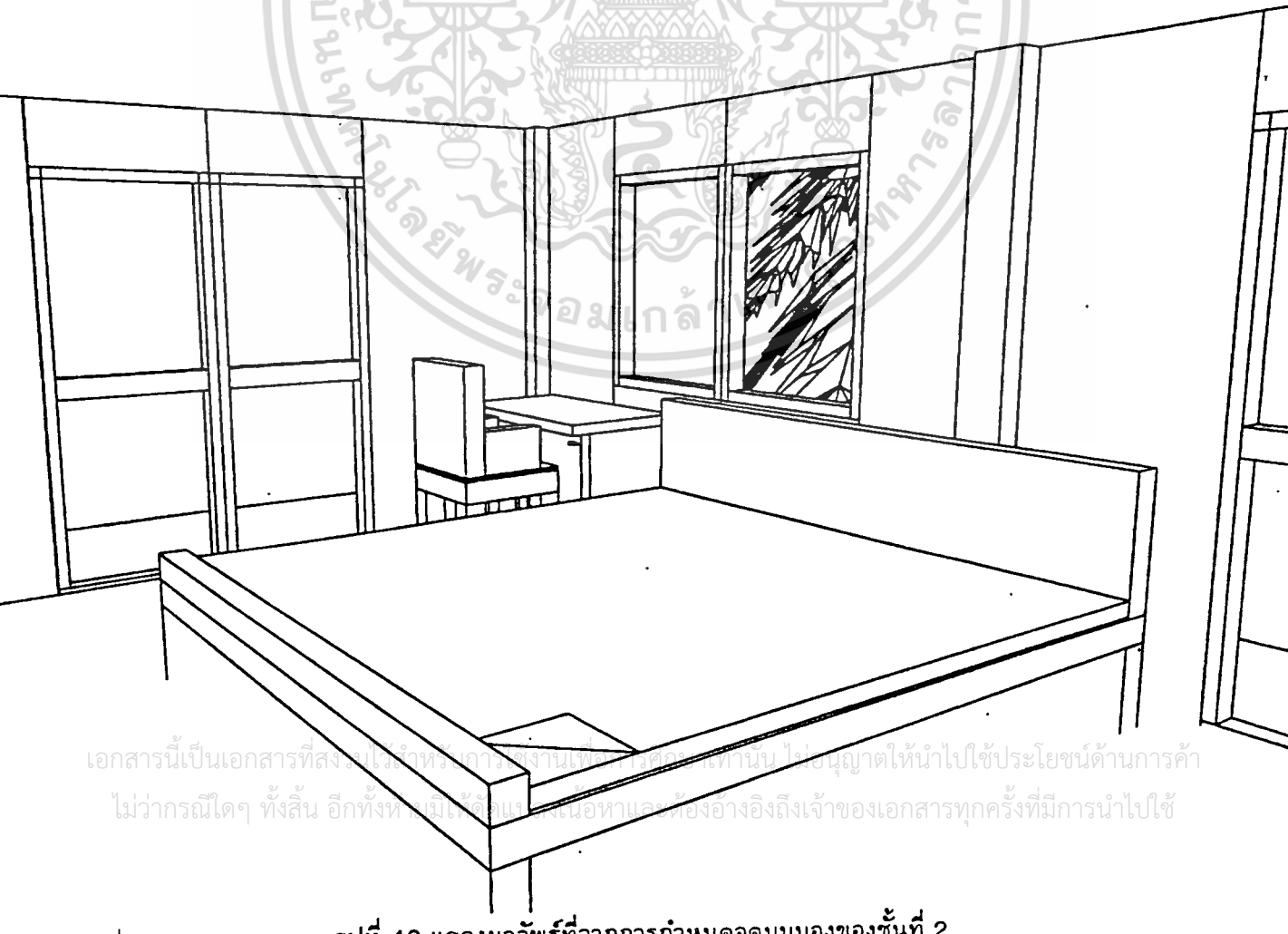


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 47 แสดงผลลัพธ์ที่จากการกำหนดจุดมุมมองของชั้นที่ 1

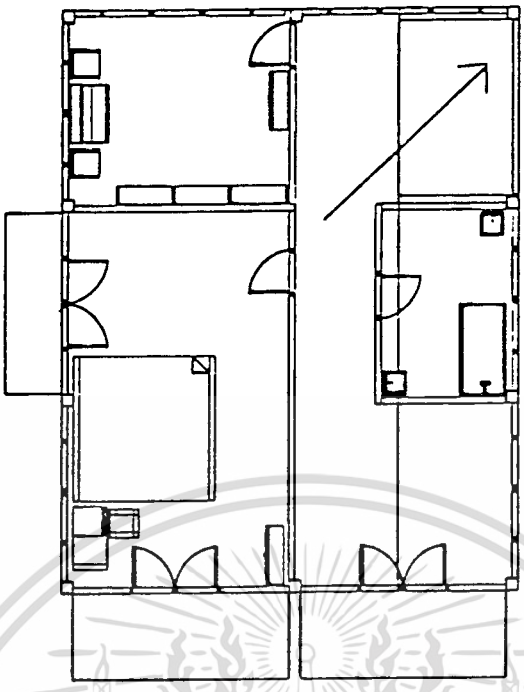


รูปที่ 48 แสดงการกำหนดจุดมุมมองบริเวณชั้นที่ 2 บริเวณห้องนอน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

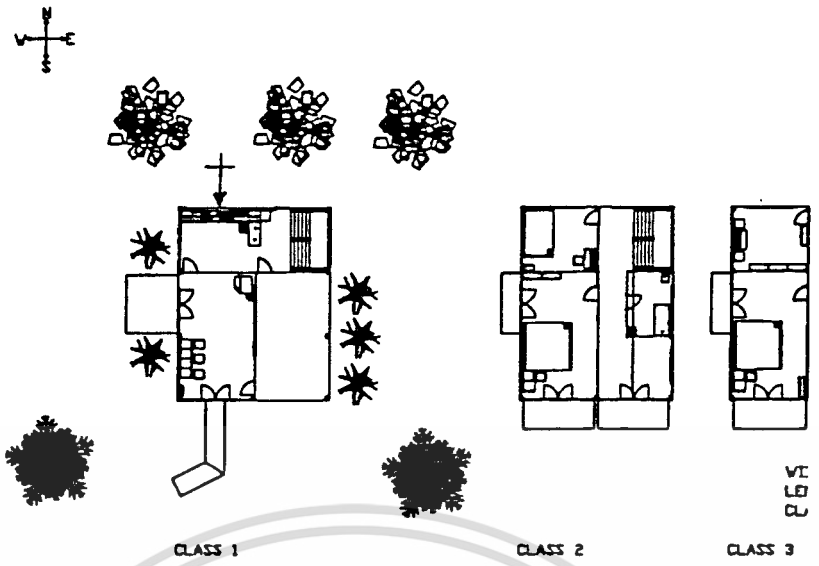
รูปที่ 49 แสดงผลลัพธ์ที่จากการกำหนดจุดมุมมองของชั้นที่ 2



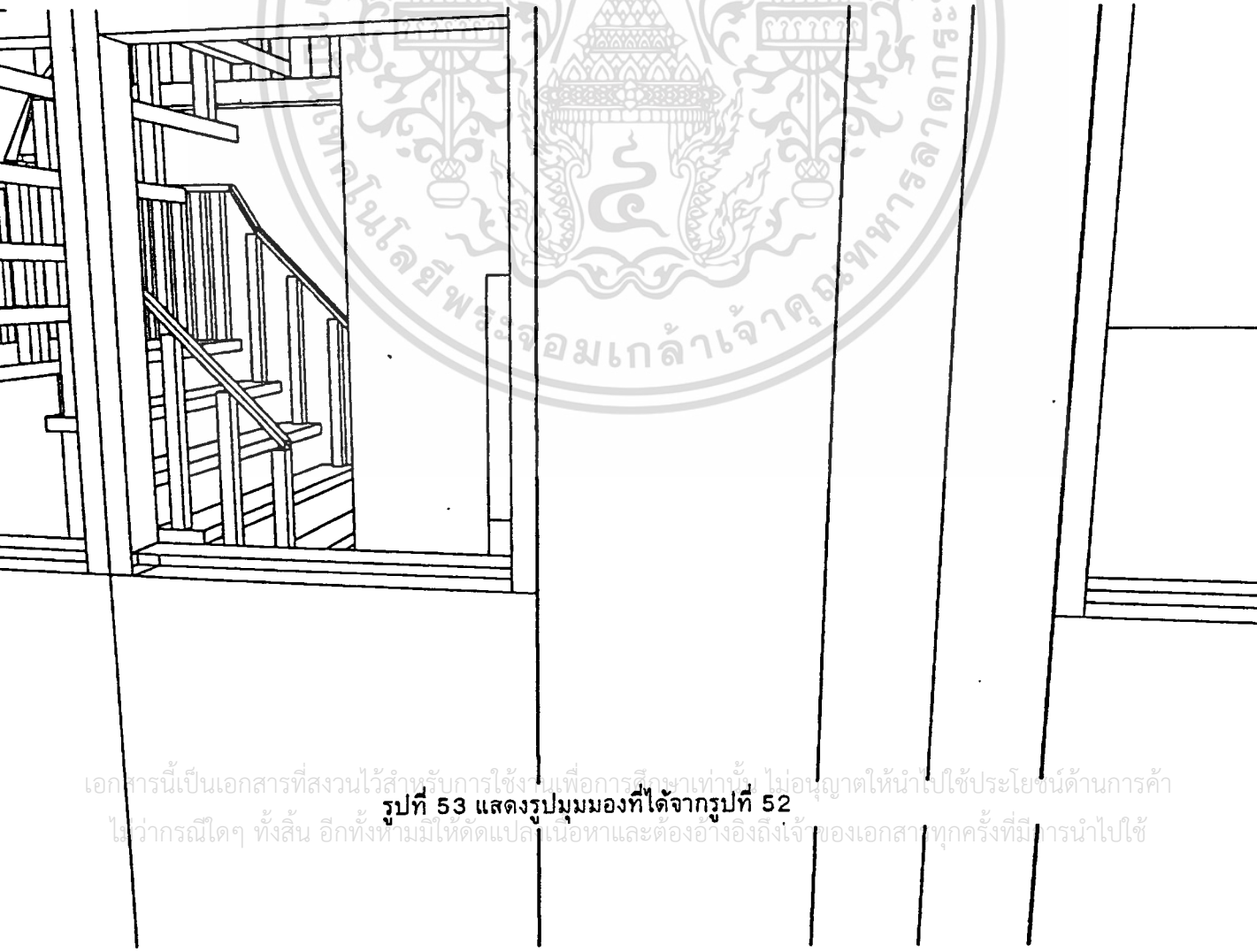
รูปที่ 50 แสดงการกำหนดจุดมุมมองบริเวณชั้นที่ 3 บริเวณบันได

รูปที่ 51 แสดงผลลัพธ์ที่จากการกำหนดจุดมุมมองของชั้นที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้



รูปที่ 52 แสดงตำแหน่งของผู้ใช้บริเวณสนามนอกบ้านที่มีมุมมองมายังตัวบ้าน



รูปที่ 53 แสดงรูปมุมมองที่ได้จากรูปที่ 52

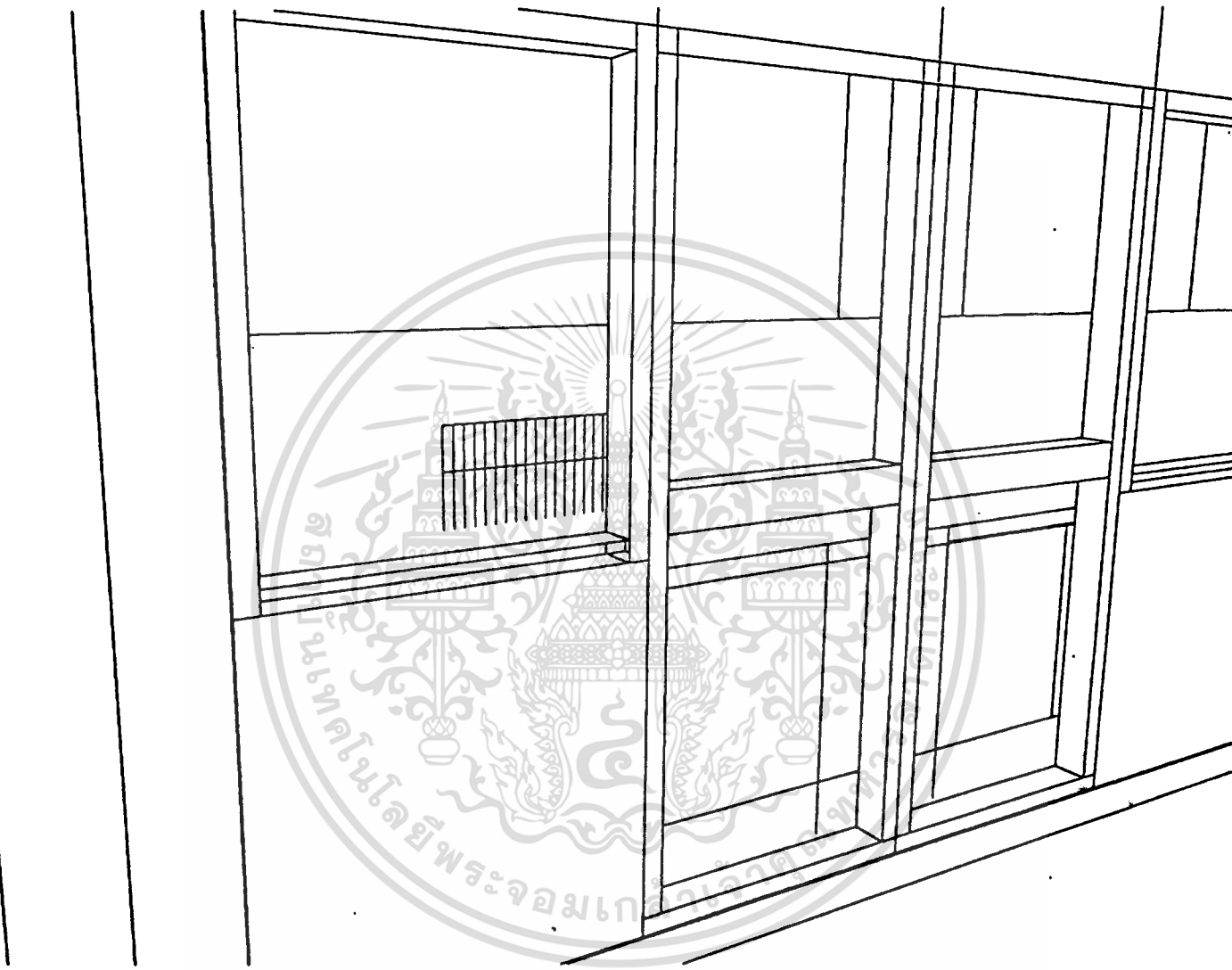
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหมุนมองดูสิ่งต่าง ๆ รอบตัวของผู้ใช้เอง ตัวอย่างของภาพที่มองเห็น แสดงดังรูปที่ 54 ถึง รูปที่ 58 เป็นต้น

เมื่อผู้ใช้ต้องการออกจากโปรแกรมออโตแคด และต้องการแบบแปลนที่สร้างขึ้นนี้ เก็บไว้ใช้ในคราวต่อไป ควรตรวจสอบดูก่อนว่าภาพที่ปรากฏเป็นแบบแปลนระนาบ 2 มิติ ซึ่งถ้าไม่ใช่ ให้ทำการเลือกเมนู

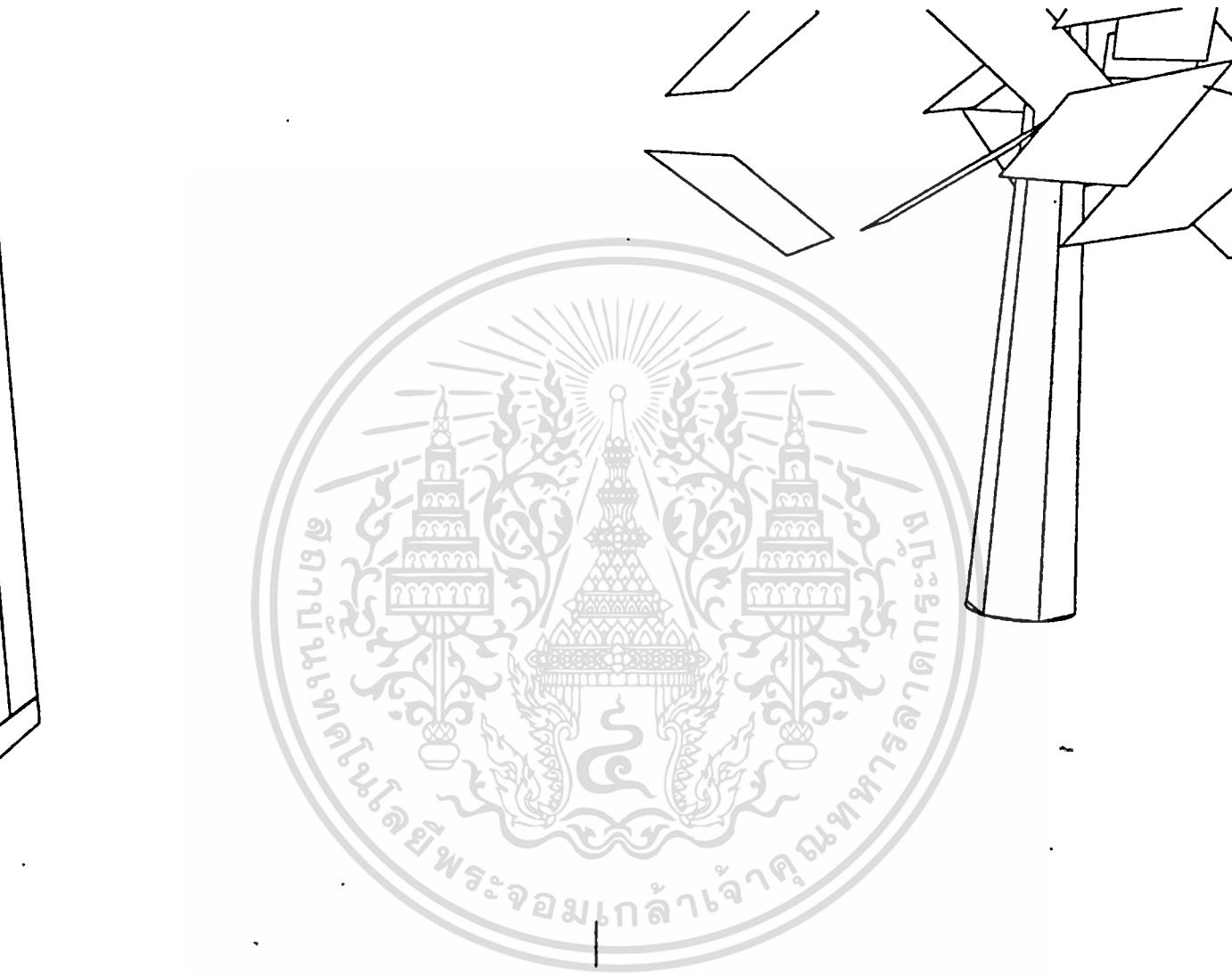
view[PlanView] ก่อนเพื่อให้ภาพคืนสู่แบบแปลน 2 มิติ หลังจากนั้นจึงทำการเลือกเมนู File[Save] เพื่อบันทึกภาพเก็บไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



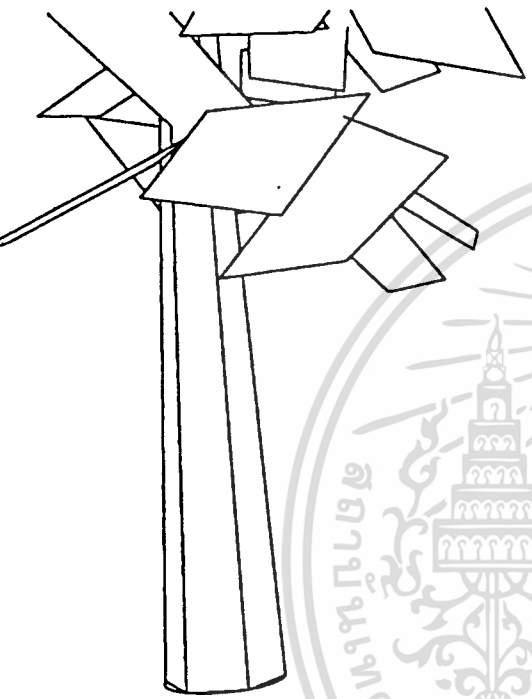
รูปที่ 54 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



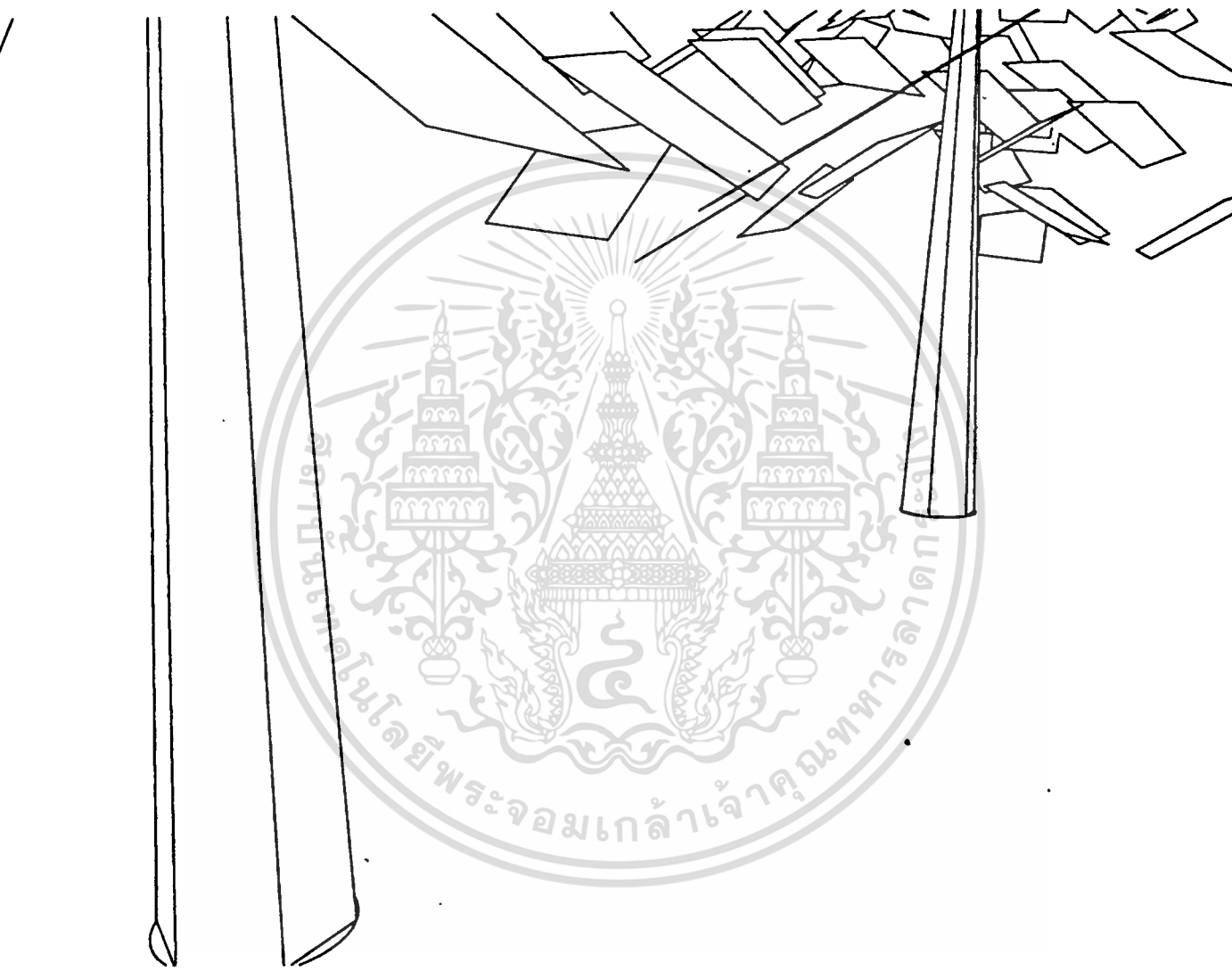
รูปที่ 55 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 56 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

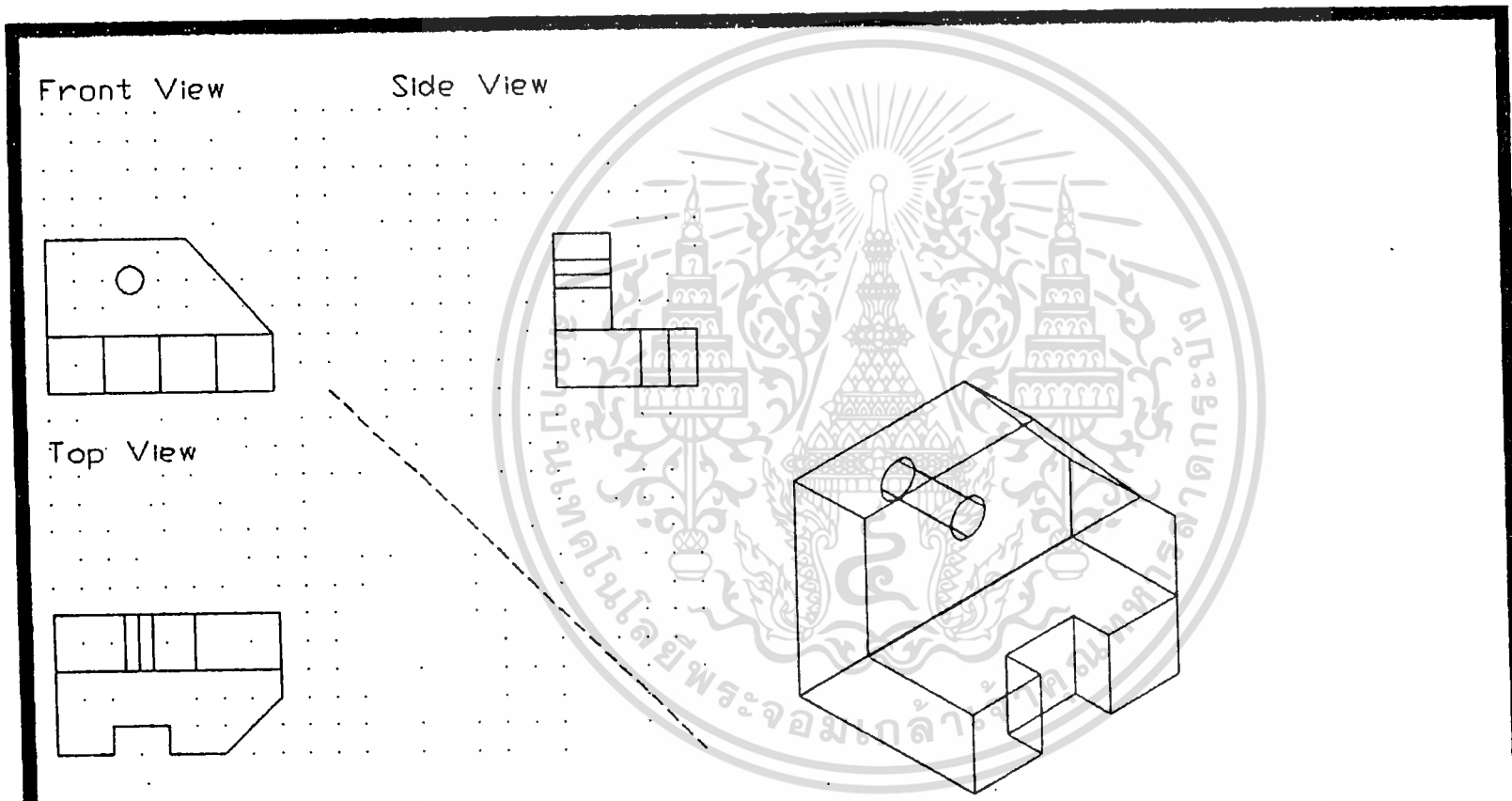


รูปที่ 57 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2]

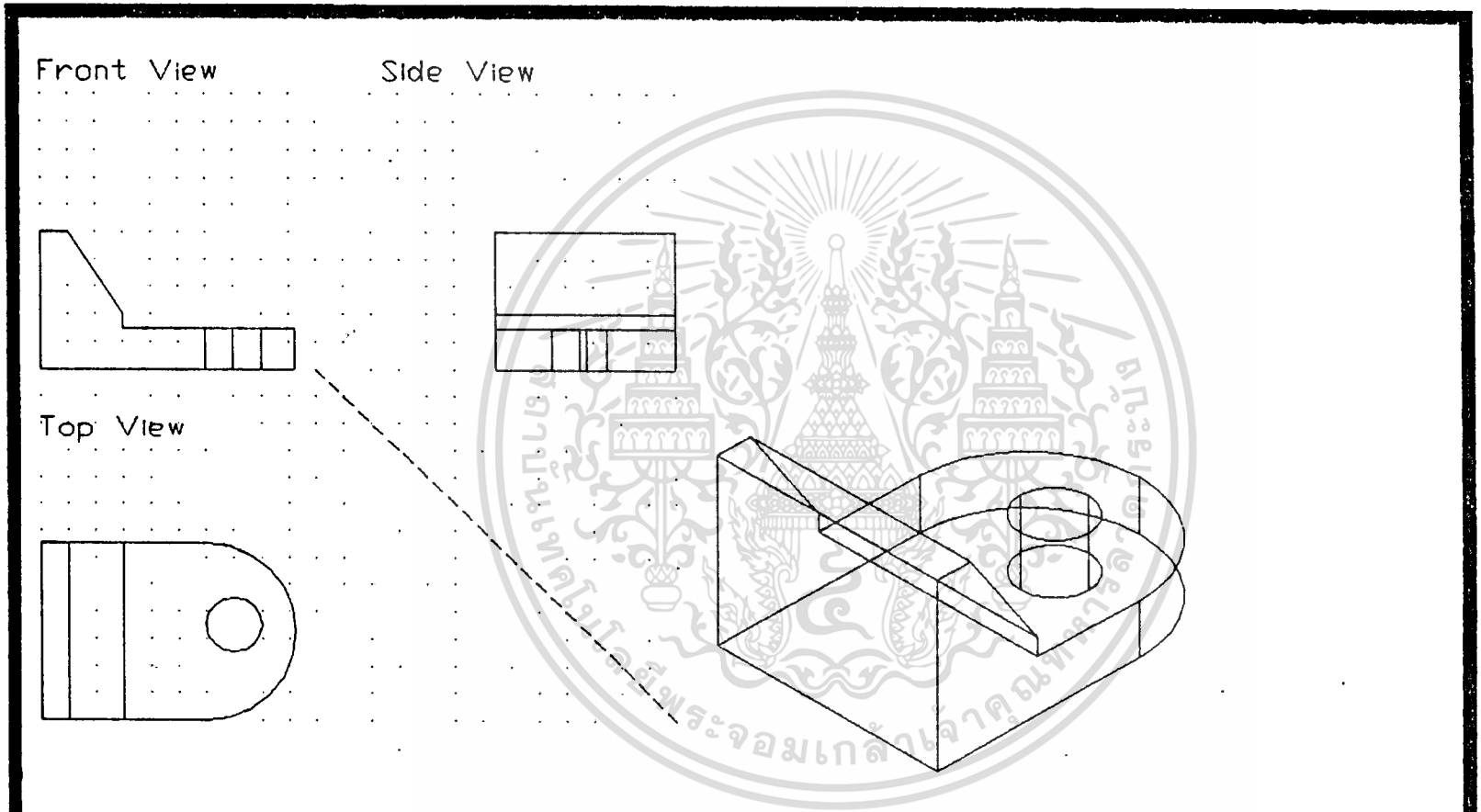
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 58 แสดงมุมมองอื่นๆ หลังจากเลือกเมนู View[Target2]

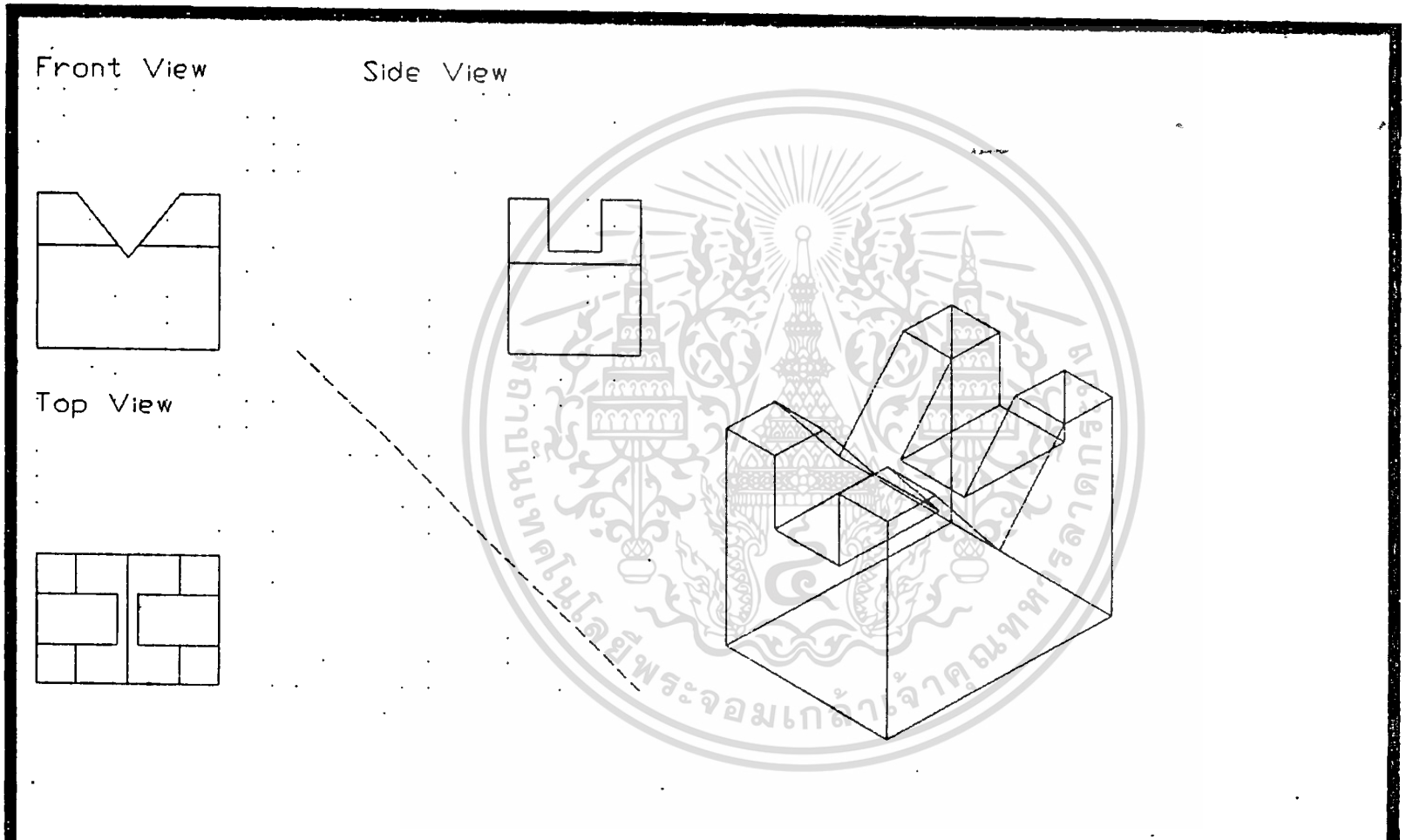
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 59 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER



รูปที่ 60 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER



รูปที่ 6-1 แสดงภาพที่ได้หลังจากเลือกเมนู PAPER

บทที่ 5

บทวิจารณ์ และ สรุป

สรุปข้อจำกัดต่าง ๆของโปรแกรม

1.ต้องทำการติดตั้งโปรแกรมAutoCAD Release 12 ไว้ในฮารด์ดิสด์อยู่แล้ว โดยเลือกทางเลือกต่าง ๆต้องทำการจัดตั้งให้ตรงตามอุปกรณ์ฮารด์แวร์ที่ใช้ เช่น ชนิดของจอมอนิเตอร์ ,อุปกรณ์ชี้ เช่น เมาส์ , อุปกรณ์พล็อตรูป เป็นต้น และควรเลือกทางเลือกโปรแกรมส่วนประกอบทั้งหมด(Install all program)ซึ่งเป็นทางเลือกหนึ่งขณะติดตั้งโปรแกรม ซึ่งโปรแกรมAutoCAD Release 12 นี้ต้องการเนื้อที่ฮารด์ดิสด์ประมาณ 25 เมกะไบด์ ทั้งนี้เนื่องจากโปรแกรมที่สร้างคือ ACAD.LSP นี้เป็นโปรแกรมออโตลิปส ซึ่งต้องอาศัยโปรแกรมออโตแคดเพื่อเป็นตัวอินเตอร์พลีทเตอร์ จึงจะสามารถทำงานได้

2.ทำการติดตั้งโปรแกรมต่าง ๆของโครงงานนี้ ซึ่งแสดงไว้ในบทที่ 1 โดยใช้เนื้อที่ฮารด์ดิสด์ประมาณ 5 เมกะไบด์ โดยต้องสร้างSubdirectoty ตามตัวอย่างด้วย และต้องทำการแก้ไขแบตไฟล์ (ACADR12.BAT) ดังนี้

```
SET ACAD=C:\ACAD;C:\ACAD\SUPPORT;C:\ACAD\FONTS;
        C:\ACAD\ADS;C:\ACAD\BLOCK;C:\ACAD\SLIDE;
```

คือต้องตั้งทางเดินของการอ่านโปรแกรมของออโตแคดให้สามารถอ่านพบไฟล์ที่เราติดตั้งไว้ได้

```
SET ACADCFG=C:\ACAD
SET ACADDRV=C:\ACAD\DRV
C:\ACAD\ACAD
```

3.ไม่สามารถทำใช้โปรแกรมนี้บน ออโตแคต รีลิสที่ต่ำกว่ารีลิส12 ได้ เนื่องจาก โปรแกรมที่สร้างต้องอาศัย ยูทิลิตี และ คำสั่งมาตรฐานต่าง ๆของโปรแกรมออโตแคดรีลิส12 ซึ่งรีลิสที่ต่ำกว่านี้ไม่มีใช้ เช่น การสร้างไดอะลอกบล็อก , การใช้คำสั่งประเภท Modeling เป็นต้น

4.ผู้ใช้งานโปรแกรมนี้ ควรมีความรู้ในการเขียนภาพบนโปรแกรมออโตแคดบ้าง ทั้งนี้เพื่อความสะดวกในการใช้คำสั่งต่าง ๆของออโตแคดในการวาดรูป เช่น คำสั่ง PLINE ,MOVE, ERASE, UNDO, ZOOM เป็นต้น

5.ในการพล็อตรูปต่าง ๆที่ได้จากการทำงานของโปรแกรมนั้น เนื่องจากภาพที่ได้ส่วนมากจะเป็นภาพ3มิติ ซึ่งออโตแคดจะแสดงเส้นต่าง ๆภายในรูปทั้งหมดทำให้ผู้ใช้เห็นภาพเส้นต่าง ๆซึ่งเป็นเส้นที่ต้องถูกบังด้วย ถ้าผู้ใช้ต้องการมองภาพที่เหมือนจริงให้เลือกเมนู Render[SHADE2] เพื่อทำการลบเส้นที่ถูกบังเหล่านั้นออก ซึ่งจะได้ภาพที่เหมือนจริงมากขึ้น

แต่ในการพล็อตรูป ถ้าผู้ใช้ต้องการได้ภาพที่มองไม่เห็นเส้นที่ถูกบังนี้ ต้องเลือกทางเลือก Hide จากคำสั่ง PLOT และต้องเลือกอุปกรณ์พล็อตประเภทที่อ่านไฟล์ PostScrip ได้เช่น เลเซอร์พรินเตอร์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.ข้อจำกัดโปรแกรมสร้าง 3D Modeling นี้ยังมีข้อจำกัด คือ

6.1. User จะต้องมีความรู้เรื่องหลักการเขียนแบบมาบ้างแล้ว จึงจะสามารถใช้โปรแกรมนี้ได้อย่างถูกต้อง

6.2. ในภาพที่มีความซับซ้อน User จะต้องเป็นผู้ตัดสินใจในการเลือกที่จะเลือกวาดส่วนต่างๆของ Drawing เช่น ต้องรู้ว่าส่วนใดเป็นส่วนที่วัตถุถูกเจาะโดยดูจาก 3 view นั้น

6.3.ยังมีบางภาพที่ก่อให้เกิดความยุ่งยากในการวาด เช่น ภาพท่อที่มีส่วนประกอบอื่นๆ

6.4.มีบางภาพที่ยังไม่สามารถทำได้ เช่น วัตถุที่ถูกเฉือนตามแนวมุมเอียงของทั้ง 3 Plan

6.5. วัตถุที่มีผิวโค้งจะได้ภาพที่ยังไม่สมจริง ต้องทำการ Render จึงจะสมบูรณ์

7. ข้อจำกัดของโปรแกรมการสร้าง Cone

7.1 สามารถทำงานภายใต้โปรแกรม ออโตแคด ที่ต่ำกว่า รีลีส 12 ได้

7.2 เนื่องจากภาพของทรงกรวยที่ถูกสร้างขึ้นนั้น ผู้ใช้จะต้องกำหนดความละเอียดของภาพโดยเลือกจากขั้นตอนที่โปรแกรมถามว่า

" Number of Segment (15) : ถ้าผู้ใช้จะต้องการได้รูปที่ละเอียดควรเลือก ตัวเลขที่มีค่าสูงกว่า 15 ตัวเลขยิ่งมากภาพจะละเอียดขึ้น แต่ก็ทำให้เปลืองเนื้อที่ disk ในการจัดเก็บรูปมากขึ้น เช่นกัน

7.3 ภาพคลี่ที่โปรแกรมสร้างขึ้นนั้น ถ้าผู้ใช้ต้องการ plot ออกมาเป็น paper ผู้ใช้จะต้องกำหนดขนาดของภาพให้เหมาะสม เช่นการกำหนดสเกลระหว่าง รูปแบบกับการ plot จริง และควรจะดูภาพว่าเลยออกนอกกระดาษ หรือไม่ โดยทำการกดปุ่ม preview ในคำสั่ง plot นั้น

7.4 ก่อนสร้างรูปทรงกรวยควรมีการ New... ก่อนทุกครั้ง

8. ข้อกำหนดโปรแกรมการสร้างบ้าน 3 มิติจากแบบแปลน 2 มิติ

8.1 ต้องทำงานบน ออโตแคด รีลีส 12 ขึ้นไปเท่านั้น เนื่องจากมีการสร้างไดอะลอกบล็อก

8.2 ในระยะแรกของการแต่งเติม เฟอร์นิเจอร์ ผู้ใช้อาจยังไม่ชินกับการวางตำแหน่งของวัตถุ เช่น กำแพง,ประตู,หน้าต่าง เป็นต้น เนื่องจากจะมีข้อกำหนดของทิศทางการวางว่าเป็น ทิศ N,S,E,W เสมอ

8.3 ในการวางตำแหน่งของอุปกรณ์ใดๆ ถ้าผู้ใช้ต้องการรู้ตำแหน่งของแบบแปลนสามารถทำได้ โดยการกด ^G เพื่อให้ออโตแคด แสดง Grid Line ออกมา ซึ่งได้กำหนดไว้ว่า แต่ละจุดของ Grid ห่างกัน 1 เมตร และสามารถทำการ Zoom ได้โดยใช้คำสั่ง Zoom

8.4 ตำแหน่งของบันได ไม่สามารถเคลื่อนย้ายได้ เนื่องจากบ้าน แต่ละชั้นจะมีบันไดอยู่ด้วย บริเวณบันไดนั้น พื้นจะต้องถูกเจาะ ดังนั้นจึงไม่สามารถเปลี่ยนตำแหน่งได้

8.5 จำนวนชั้นของบ้าน,ขนาดของบ้าน ผู้ใช้สามารถกำหนดได้เลย เช่น บ้านสามารถมีจำนวนชั้นได้ n ชั้นเลย แต่รูปภาพแปลนที่ปรากฏก็จะเล็กตามไปด้วยจำนวนชั้นที่เพิ่มขึ้น

8.6 ในขณะใช้คำสั่งแบบแปลนกลายเป็น บ้าน 3 มิติ หรือ จากบ้าน 3 มิติ เป็น 2 มิติ หรือในการหมุนภาพ 3 มิตินั้น ถ้าภาพที่มีความซับซ้อนมากเท่าไร ผลของภาพที่ออกมาก็จะช้าตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.7 ภาพบ้าน 3 มิติที่ปรากฏออกมา จะแสดงเป็นเส้นทุกเส้นที่ปรากฏอยู่ จึงทำให้มองไม่ชัดเจน ผู้ใช้ควรใช้คำสั่ง Shade2 เพื่อลบเส้นที่ถูกบังออก หรือ Shade0,Shade1,Shade3 เพื่อทำการระบายสีภาพบ้านแบบต่างๆ

สรุป

1.ประโยชน์ของโปรแกรมภาพ 3 มิติ(Isometric Drawing) จากภาพฉาย 2 มิติที่ให้มาทั้งหมด 3ด้าน คือ Top,Front,Side ทำให้เกิดความง่ายและความสะดวกในการเขียนแบบวิศวกรรม และยังสร้างความเพลิดเพลินในการเขียนแบบ รวมทั้งสามารถสร้างเป็น Drawing Space ของภาพ 3 view และ 3D modeling ที่ได้และสามารถทำการ Plot ต่อไป

2.ประโยชน์ของโปรแกรมการสร้างภาพทรงกรวย และ ภาพคลี่ของทรงกรวยที่สร้างสามารถนำมาทดลองสร้างแบบสำหรับทำอุปกรณ์ทดลองต่างๆ ที่มีทรงกรวยเป็นองค์ประกอบ และยังสามารถใช้เป็นตัวอย่างของโปรแกรมที่จะใช้สร้างภาพคลี่ของรูปทรงอื่นได่อไปได้

3.ประโยชน์ของโปรแกรมการสร้างบ้านจากแบบแปลนบ้าน2มิติ ช่วยให้ผู้ใช้สามารถทดลองดูตัวบ้านที่จะสร้าง โดยยังไม่ต้องทำการสร้างบ้านจริง โดยทำการออกแบบบนแบบแปลน 2 มิติ และทดลองใส่เฟอร์นิเจอร์ต่างๆ ได้เลย และทำการดูมุมมองต่างๆของบ้านตามทิศทางที่กำหนดบนแบบแปลนนั้นได้ ซึ่งสามารถนำไปประยุกต์ใช้ในการออกแบบโรงงาน หรือ สถานที่ต่างๆได้

จึงสามารถสรุปได้ว่า การที่นำโปรแกรมออโตแคดมาใช้งานนั้น เพราะโปรแกรมออโตแคดนี้มีความสามารถในการวาดรูปเป็นอย่างมาก และ สามารถทำใช้งานได้ในหลายๆด้าน ไม่ใช่ว่าโปรแกรมออโตแแคดจะนำมาใช้ในการเขียนแบบเท่านั้น โดยจะทำการประยุกต์เขียนโปรแกรมออโตลิปเข้าไปควบคุมการทำงานของออโตแคดอีกทีหนึ่ง ซึ่งสามารถนำมาใช้ได้พร้อมๆกับการทำงานบนออโตแคด ซึ่งสร้างความสะดวกแก่ผู้ใช้อย่างมาก และยังสามารถประยุกต์เขียนโปรแกรมในด้านอื่นๆได้ ดังตัวอย่างของโปรแกรมในโครงงานครั้งนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปคำสั่งที่ใช้ใน AutoCAD

APERTURE	ควบคุมขนาดของเป้า
ARC	เขียนส่วนโค้งของวงกลม
AREA	หาพื้นที่และเส้นรอบรูปของรูปเหลี่ยม
ARRAY	ทำสำเนาของรูปที่เลือกแบบเป็นแถวหรือเป็นวงกลม
ATTDEF	สร้างข่าวสารในรูปของ TEXT ใน ATTRIBUTE
ATTDISP	ควบคุมการมองเห็นของ ATTRIBUTE
ATTEDIT	ตรวจแก้ ATTRIBUTE
ATTEXT	เรียกข้อมูลของ ATTRIBUTE มาจากภาพ
BASE	กำหนดจุดกำเนิดของการสอดแทรกไปที่อีกภาพหนึ่ง
BLIPMODE	ให้มีหรือไม่ให้มีจุดบนจอภาพขณะเขียนแบบ
BLOCK	จัดกลุ่มสิ่งที่ซับซ้อนรวมเข้าเป็นส่วนของภาพเพียงส่วนเดียว
BREAK	ลบส่วนของวัตถุหรือแยกเป็นสองส่วน
CHAMFER	ลบเหลี่ยมเส้นสองเส้นที่ตัดกัน
CHANGE	เปลี่ยนคุณสมบัติหรือตำแหน่งของวัตถุ
CIRCLE	เขียนวงกลม
COLOR	กำหนดสีให้แก่วัตถุที่เขียน
COPY	ทำสำเนาของวัตถุ
DBLIST	แสดงรายการฐานข้อมูลทุกๆส่วนของภาพ
DELAY	หน่วงเวลาการทำงานของคำสั่งต่อไปตามระยะเวลาที่กำหนด
DIM	ให้ Dimension แก่ภาพ
DIMI	เขียนเส้นบอกขนาดได้เส้นเดียว แล้วกลับไปรับคำสั่งอื่น
DIST	หาระยะทางระหว่างจุดสองจุด
DIVIDE	แบ่งวัตถุให้เป็นส่วนๆเท่าๆกันตามแต่จะกำหนดว่ากี่ส่วน
DOUGHNUT	เขียนรูปวงแหวน โดยมีเส้นผ่านศูนย์กลางวงในและวงนอกตามที่กำหนด
DONUT	เหมือนคำสั่ง DOUGHNUT
DRAGMODE	กำหนดให้ว่าจะมีการเขียนรูปตามไปขณะที่สายใยเคลื่อนที่หรือไม่
DTEXT	ขณะเขียนตัวอักษรจะมีตัวอักษรขึ้นเมื่อพิมพ์ลงไป
DXBIN	คำสั่งพิเศษสำหรับโปรแกรม เช่น CAD/Camera
DXFIN	โหลด Drawing Interchange File
DXFOUT	เขียน Drawing Interchange File
ELEV	ตั้งระดับและความหนาของส่วนของภาพที่จะเขียนเมื่อทำเป็นภาพ 3 มิติ
ELLIPSE	เขียนรูป Ellipse

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

END	บันทึก Drawing file ที่กำลังเขียนอยู่แล้วออกจาก Drawing Editor
Erase	ลบส่วนของภาพออกจากแบบ
EXPLODE	แยก Block หรือ Polyline ที่เขียนไว้ให้เป็นส่วนๆ
EXTEND	ตอความยาวของเส้นตรง ส่วนโค้ง หรือ Polyline ไปจนจดกับวัตถุอื่น
FILES	เรียกเมนูของ File Utilities มาใช้
FILL	ควบคุมให้วัสดุทึบโดยอัตโนมัติ
FILLET	สร้างเส้นโค้งต่อเส้นตรงสองเส้นตามรัศมีที่กำหนด
'GRAPHSCR	พลิกให้แสดงภาพกราฟิกในระบบจอเดียว ใช้ในการเขียน Command Script และ เมนู
GRID	แสดงจุดเป็นเส้นกริดบนจอภาพตามระยะที่กำหนด
HATCH	เขียนเส้นระบาย(Cross-Hatch) และเส้นลวดลาย
'HELP	แสดงรายการของคำสั่งที่ใช้อยู่ และแนะนำการป้อนข้อมูล
HIDE	สร้างภาพ 3 มิติขึ้นใหม่โดยลบเส้นที่ถูกบังออกไป
ID	แสดงโคออร์ดิเนตของจุดที่กำหนด
IGESIN	โหลด IGES Interchange File
IGESOUT	เขียน IGES Interchange File
INSERT	สอดแทรกลำเนาของวัตถุที่เขียนมาแล้ว เข้าไปในภาพที่กำลังเขียนอยู่
ISOPLANE	เลือกระนาบ(Plane) ของเส้นกริดที่เป็น Isometric เพื่อให้เป็นระนาบที่ใช้งานในการเขียนแบบ
LAYER	สร้างชั้นของภาพที่กำหนดชื่อไว้แล้ว และให้สีกับแบบของเส้นในชั้นเหล่านั้น
LIMITS	เปลี่ยนขอบเขตของภาพและตรวจสอบว่าภาพอยู่ในขอบเขตหรือไม่
LINE	เขียนเส้นตรง
LINETYPE	กำหนดแบบของเส้น(เส้นประ) และโหลดมาจาก Library
LIST	แสดงรายการที่เป็นฐานข้อมูลสำหรับวัตถุที่กำหนด
LOAD	Load File ของ Shapes ที่ผู้ใช้เป็นผู้กำหนดเพื่อเอามาใช้ในคำสั่ง SHAPE
LTSCALE	กำหนดตัวคูณของมาตราส่วน (Scaling Factor) ที่จะมาใช้กับแบบของเส้นในภาพ
MEASURE	จะทำเครื่องหมายไว้บนวัตถุห่างกันเท่าที่จะกำหนด
MENU	จะโหลดไฟล์ของคำสั่งใน Drawing Editor เข้าไปไว้ในพื้นที่ที่จัดไว้
MINSERT	ทำการสอดแทรก BLOCK หลายๆรูปเป็น Array แบบสี่เหลี่ยม
MIRROR	สร้างรูปกลับข้างตามแกนที่ผู้ใช้กำหนด
MOVE	เลื่อนส่วนของภาพไปไว้ในตำแหน่งอื่น
MSLIDE	สร้าง Slide File สำหรับแสดง
OFFSET	เป็นการสร้างเส้นหรือส่วนโค้งที่ขนานกับเส้นหรือส่วนโค้งเดิม
OOPS	กู้ภาพที่ถูกลบให้คืนกลับมา
ORHTO	บังคับให้เส้นที่เขียนเป็นเส้นราบกับเส้นดิ่งเท่านั้น
OSNAP	ทำให้สามารถกำหนดให้จุดอยู่ในตำแหน่งบนอีกวัตถุหนึ่งพอดีตามที่ต้องการ
PAN	เลื่อนจอให้แสดงภาพตามตำแหน่งต่างๆของภาพ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามมิให้ผู้ใดนำไปใช้ประโยชน์ด้านการค้า เว้นแต่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่ว่าทางตรงหรือทางอ้อม และห้ามมิให้ดัดแปลงเนื้อหาของเอกสารนี้ก่อนที่จะได้รับอนุญาตให้นำไปใช้

PEDIT	ตรวจแก้ Polyline
PLINE	ลากเส้นต่อเส้นหรือส่วนของส่วนโค้งด้วยความกว้าง และส่วนเรียวตามที่ต้องการ
PLOT	พล็อตแบบแปลนด้วยเครื่องพล็อต
POINT	เขียนจุด ๆหนึ่ง
POLYGON	เป็นการสร้างรูปหลายเหลี่ยมตามแต่จะกำหนดจำนวนด้าน
PRPLOT	พิมพ์ภาพด้วยเครื่องพิมพ์
PURGE	ลบบล็อก แบบตัวอักษร(Text Style) ชั้น หรือแบบของเส้นที่ไม่ต้องการออกจากภาพ
QTEXT	ทำเป็นกรอบแสดงว่ามีตัวอักษรอยู่โดยไม่ต้องเขียนตัวอักษร
QUIT	ออกจาก Drawing Editor เข้าสู่ Main Menu ของ AutoCAD โดยไม่ได้บันทึกส่วนที่แก้ไขครั้งหลัง
REDO	หลังจากที่ใช้คำสั่ง U หรือ UNDO แล้วคำสั่งนี้จะกลับไปเขียนรูปใหม่
REDRAW	เขียนรูปใหม่บนจอภาพ
REGEN	คำนวณและสร้างรูปใหม่บนจอภาพ
REGENAUTO	สามารถบังคับให้มีการสร้างรูปบนจอภาพขึ้นมาใหม่โดยอัตโนมัติเมื่อรับคำสั่งอื่น
RENAME	เปลี่ยนชื่อที่เกี่ยวกับแบบของตัวอักษร Named View ชั้น แบบ ของเส้น และบล็อก
'RESUME	เมื่อมีการใช้คำสั่งขัดจังหวะระหว่างการทำ SCRIPT คำสั่งนี้จะกลับไปทำต่อ
ROTATE	จะหมุนวัตถุที่มีอยู่
RSCRIPT	ตั้งต้นทำตาม Command Script ตั้งแต่แรก
SAVE	บันทึกภาพที่แก้ไขไว้โดยที่ไม่ต้องออกจาก Drawing Editor
SCALE	เปลี่ยนขนาดของวัตถุที่มีอยู่
SCRIPT	สั่งให้ทำตาม Command Script
SELECT	เลือกวัตถุไว้สำหรับใช้ทำงานในภายหลัง
'SETVAR	สำหรับตรวจดูและเปลี่ยนค่าในตัวแปรในระบบ
SH	สำหรับออกจากโปรแกรมไปที่ DOS ชั่วคราว
SHAPE	เขียนรูปร่าง(Shape) ตามที่ได้กำหนดไว้แล้ว
SHELL	ออกจาก AutoCAD ไปทำงานในโปรแกรมอื่น
SKETCH	ร่างรูปด้วยมือ
SNAP	กำหนดให้จุดที่เขียนลงจัวหวะพอดีตามที่กำหนด
SOLID	เขียนรูปเหลี่ยมโดยระบายเต็ม
STATUS	แสดงสถิติต่าง ๆของรูปที่กำลังเขียนอยู่
STRETCH	เลื่อนส่วนของภาพออกไป โดยไม่กระทบกระเทือนส่วนที่เกี่ยวข้องส่วนอื่น
STYLE	สร้างรูปแบบตัวอักษร พร้อมกับเลือกชนิดตัวอักษรและอื่น ๆ
TABLET	เทียบ Digitizing Tablet กับโคออร์ดิเนตของกระดาษเขียนแบบเพื่อจะได้ Copy รูปได้อย่างถูกต้อง
TEXT	เขียนตัวอักษรขนาดต่าง ๆตามแบบที่เลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEXTSCR	พลิกไปให้จอภาพแสดงตัวอักษร ใช้ใน Command Script และเมนู
TIME	จะแสดงเวลาปัจจุบัน เวลาที่สร้างภาพ
TRACE	เขียนเส้นทึบตามความกว้างที่กำหนด
TRIM	ลบบางส่วนของภาพตามขอบเขตที่กำหนด
U	ลบสิ่งที่ได้กระทำก่อนหน้านี้หลาย ๆคำสั่งสุดท้าย
UNDO	ลบล้างสิ่งที่กระทำในคำสั่งสุดท้าย
UNITS	เลือกการแสดงรูปแบบและความละเอียดของโคออร์ดิเนตและมุม
VIEW	บันทึกภาพที่แสดงอยู่ (เป็น View ที่มีชื่อ) หรือเรียกรูปที่บันทึกไว้ออกมาแสดง
VIEWRES	จะทำให้สามารถควบคุมความละเอียดและความเร็ว ในการเขียนวงกลมหรือส่วนโค้งของวงกลมได้
VPOINT	เลือกมุมมองสำหรับภาพ 3 มิติ
VSLIDE	แสดง Slide File ที่สร้างไว้
WBLOCK	เขียนส่วนของภาพลงใน Disk File
ZOOM	ขยายหรือลดภาพที่แสดง

หมายเหตุ คำสั่งที่มีเครื่องหมายคำพูดเดี่ยวนำ (') เรียกว่า Transparent Command เป็นคำสั่งที่ใช้ได้แม้จะอยู่ในคำสั่งอื่นอยู่ เวลาใช้คำสั่งไม่ต้องให้เครื่องหมายคำพูดลงไปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปรระบบในออโตแคด

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
ACADPREFIX	"C:\ACAD\"			RO*	เก็บ directory path ของระบบออโตแคด
ACADVER	"10"			RO	บอกรุ่นของออโตแคด
AFLAGS	0		ATTDEF		เก็บค่าของแอตทิบิว 1=invisible,2=constant 4=verify,8=preset
ANGBASE	0	3 นาฬิกา	UNITS		ทิศทางของมุม 0 องศา โดยยึดจากมุมมาตรฐานเป็นหลัก
ANGDIR	0	ทวนเข็ม	UNITS		ทิศทางการวัดมุมเป็นบวก1=ตามเข็มเป็นบวก ,0=ทวนเข็มเป็นบวก
APERTRUE	10	10	APERTRUE	CFG**	ขนาดของ APERTURE หน่วยเป็น pixels
AREA	0		AREA,LIST	RO	ขนาดของ area ที่ได้คำนวณไว้ล่าสุด
ATTDIA	0	ไม่มี			ควบคุม attribute dialogue box สำหรับการ insert 0=ไม่มี dialogue box,1=มี dialogue box
ATTMODE	1	Normal	ATTDISP		ควบคุมการแสดงแอตทิบิว 0=Off,1=Normal,2=On
ATTREQ	1	มี			ควบคุมพร็อมต์ของแอตทิบิว 0=ไม่มีพร็อมต์จะใช้ค่าที่กำหนดให้, 1=มีพร็อมต์
AUNITS	0	Decimal	UNITS		หน่วยของมุมที่ใช้ 0=Decimal,1=Degrees, 2=grad,3=rad,4=survey

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
DIMALTF	25.40		DIMALTF		แฟกเตอร์ตัวคูณของ alternate units เช่น ขนาดปกติเป็น 1 ขนาด alternate units จะได้เป็น 2.54 เป็นต้น
DIMAPOST	" "	ไม่มี	DIMAPOST	RO	ข้อความต่อท้ายตัวเลขบอกขนาดของ alternate units
DIMASO	1	ON	DIMASO		ควบคุมการปรับขนาดบอกระยะเมื่อมีการแก้ไขขนาดของภาพที่กำหนดขนาดแล้ว 0=OFF,1=ON
DIMASZ	0.18		DIMASZ		ขนาดของลูกศรบอกขนาด
DIMBLK	" "		DIMBLK	RO	ชื่อ block ที่จะกำหนดให้เขียนแทนหัวลูกศรบอกขนาด
DIMBLK1	" "		DIMBLK1	RO	ชื่อ block ที่จะกำหนดให้เขียนแทนหัวลูกศรบอกขนาดในปลายที่1เมื่อ DIMSAH = ON
DIMBLK2	" "		DIMBLK2	RO	ชื่อ block ที่จะกำหนดให้เขียนแทนหัวลูกศรบอกขนาดในปลายที่2เมื่อ . DIMSAH = ON
DIMCEN	0.09		DIMCEN		ขนาดของเส้นกากบาทที่จุดศูนย์กลาง
DIMDLE	0.00		DIMDLE		ขนาดของเส้นบอกขนาดที่ยาวเกินออกมา เมื่อ DIMTSZ มากกว่า 0
DIMDLI	0.38		DIMDLI		ระยะห่างระหว่างเส้นบอกขนาดเมื่อกำหนดเป็น BAS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
AUPREC	0	0	UNITS		ตำแหน่งทศนิยมในการแสดงค่ามุม
AXISMODE		OFF	AXIS		ควบคุมการแสดง axis 0=OFF,1=ON
AXUNIT	0.0,0.0		AXIS		ระยะ X,Y ของ axis
BACKZ	0		DVIEW	RO	ระยะ back clippint สำหรับการมองภาพ 3 มิติ
BLIPMODE	1	มี	BLIPMODE		ควบคุมกากบาทบนจอ 0=ไม่มี,1=มี
CECOLOR	"BYALAYER"		COLOR	RO	สีปัจจุบันที่ได้เซตไว้ด้วยคำสั่ง COLOR
CELTYPE	"BYLAYER"		LINETYPE	RO	ชนิดของเส้นที่ใช้ปัจจุบัน
CHAMFERA	0		CHAMFER		ระยะ chamfer ที่ 1
CHAMFERB	0		CHAMFER		ระยะ chamfer ที่ 2
CLAYER	"0"		LAYER	RO	layer ที่ใช้ปัจจุบัน
CMDECHO	1	ECHO			แสดงรายละเอียดคำสั่งขณะรันโปรแกรมออโตลิปส์ 0=ไม่แสดง,1=แสดง
CORDS	0	OFF	[^D],F6		ควบคุมลักษณะการแสดงโคออร์ดิเนต 0=กำหนดจุดก่อนจึงจะแสดงค่าปัจจุบัน, 1=แสดงค่าปัจจุบันตลอดเวลา,2=แสดงรูปแบบระยะมากกว่ามุม
CVPORT	1		VPORTS		หมายเลขของ current view port
DATE	2447498.61		TIME	RO	Julian time
DIMALT	0	OFF	DIMALT		ควบคุมการเปิดปิด alternate units 1=ON, 0=OFF
DIMALTD	2		DIMALTD		จำนวนตำแหน่งทศนิยมของ alternate units

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
DIMEXE	0.18		DIMEXE		ระยะของเส้นช่วยบอกขนาดที่ยาวออกไปเกินจากเส้นบอกขนาด
DIMEXO	0.0625		DIMEXO		ระยะห่างจากจุดที่ pick ไปถึงปลายเส้นช่วยบอกขนาด
DIMLFAC	1.00	ปกติ	DIMLFAC		ตัวคูณระยะที่ได้จากการวัดขนาดจริง
DIMLIM	0	OFF	DIMLIM		บอกค่าพิกัดช+,- 0=OFF, 1=ON
DIMPOST	" "	ไม่มี	DIMPOST	RO	ข้อความที่ต่อท้ายเลขบอกขนาดปกติ
DIMRND	0.00	ไม่มี	DIMRND		กำหนดการปัดค่าตัวเลขบอกขนาดเพื่อให้อ่านค่าโดยประมาณได้เช่น ปัดครั้งละ 0.5,0.25 ฯลฯ
DIMSAH	0	OFF			ควบคุมการแยกสัญลักษณ์ที่จะเขียนแทนหัวลูกศร (DIMBLK1 และDIMBLK2) 0=OFF,1=ON
DIMSCALE	1.00		DIMSCALE		แฟกเตอร์ตัวคูณค่าต่าง ๆ เกี่ยวกับการบอกค่าที่ได้เซตไว้
DIMSE1	0	OFF	DIMSE1		ยกเลิกเส้นช่วยบอกขนาดเส้นที่1 0=OFF,1=ON
DIMSE2	0	OFF	DIMSE2		ยกเลิกเส้นช่วยบอกขนาดเส้นที่2 0=OFF,1=ON
DIMSHO	0	OFF	DIMSHO		แสดงเลขบอกขนาดที่เปลี่ยนแปลงขณะแก้ไขขนาด
DIMSOXD	0	OFF	DIMSOXD		ยกเลิกเส้นบอกขนาดที่อยู่ด้านนอกเส้นช่วยบอกขนาดเมื่อ DIMTIX=ON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
DIMTAD	0	OFF	DIMTAD		กำหนดให้เขียนตัวเลขบอกขนาดเหนือเส้นบอกขนาดในแนวนอน
DIMTIH	1	ON	DIMTIH		กำหนดให้เขียนตัวเลขบอกขนาดในแนวนอน เมื่อตัวเลขอยู่ด้านใน0=OFF(เขียนตามแนวเส้นบอกขนาด),1=ON (เขียนแนวนอน)
DIMTIX	0	OFF	DIMTIX		บังคับให้เขียนเลขบอกขนาดภายในเส้นช่วยบอกขนาด 0=OFF,1=ON
DIMTM	0.00		DIMTM		ค่าพิกัดสำหรับค่าลบ เมื่อ DIMLIM เป็น ON
DIMTOFL	0	OFF	DIMTOFL		กำหนดให้เขียนเส้นบอกขนาดภายในเส้นช่วยบอกขนาดไม่ว่าจะมีระยะแคบแค่ไหนก็ตาม 0=OFF,1=ON
DIMTOH	1	ON			กำหนดให้เขียนตัวเลขบอกขนาดในแนวนอน เมื่อตัวเลขอยู่ภายนอก 0=OFF(เขียนตามแนวเส้นบอกขนาด),1=ON(เขียนตามแนวนอน)
DIMTOL	0	OFF	DIMTOL		กำหนดให้มีการบอกขนาดเป็นพิกัดบวกลบ 0=OFF, 1=ON
DIMTP	0.00		DIMTP		ค่าพิกัดสำหรับค่าบวกเมื่อ DIMLIM เป็น ON
DIMTSZ	0.00	หัวลูกศร	DIMTSZ		ความยาวของเส้นทแยงที่ใช้แทนลูกศร ถ้ามีค่าเป็นศูนย์จะใช้เป็นหัวลูกศร
DIMTVP	0.00		DIMTVP		กำหนดระยะห่างระหว่างเลขบอกขนาดกับเส้นบอกขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
DIMTXT	0.18		DIMTXT		กำหนดขนาดความสูงของตัวเลขบอกขนาด
DIMZIN	0		DIMZIN		ควบคุมการแสดง/ไม่แสดงเลขศูนย์สำหรับหน่วยนิ้ว 0=เลขศูนย์สำหรับนิ้ว,ฟุต จะไม่เขียน 1=เลขศูนย์สำหรับนิ้ว,ฟุต จะเขียนตามปกติ 2=เลขศูนย์นิ้ว จะไม่เขียน 3=เลขศูนย์ฟุต จะไม่เขียน
DISTANCE	0.00		DIST	RO	ค่าระยะ distance ครั้งสุดท้ายที่ได้หาไว้
DTAGMODE	2	AUTO	DRAGMODE		ควบคุมการ DRAGGING 0=OFF,1=ON,2=AUTO
DRAGP1	10		SETVAR	CFG	หน่วงเวลาที่จะเริ่มdragภาพได้(ค่ามากจะใช้เวลามาก)
DRAGP2	25		SETVAR	CFG	ความถี่ที่ใช้ในการสร้างภาพขณะDragging(ค่ามากจะdrag ได้ช้าลง)
DWGNAME				RO	ชื่อ drawing ที่อยู่ขณะนั้น
DWGREFIX				RO	ชื่อ path ของ drawing ที่ใช้อยู่ปัจจุบัน
ELEVATION	0.00		ELEV		ระดับ elevation ปัจจุบัน
EXPERT	0	ปกติ	SETVAR		ยกเลิกข้อความที่จะถาม (prompt)0=ไม่มีการยกเลิกข้อความใดๆทั้งสิ้น 1=ยกเลิก prompt about regen และ really want to turn the current light off 2=ยกเลิก 1 และ block already defined และ drawing with this name already exists.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
					3=ยกเลิก 2และข้อความเตือนเกี่ยวกับ linetype 4=ยกเลิก3และข้อความเตือนเกี่ยวกับการsave UCS VPORTS
EXTMAX				RO	ขอบเขตของ drawing extents มุมขวาบน
EXTMIN				RO	ขอบเขตของ drawing extents มุมซ้ายล่าง
FILLETRAD	0.00		FILLET		ค่ารัศมีที่ใช้ในการมนมุมด้วย fillet
FILLMODE	1	ON	FILL		ระบายทึบเส้นที่มีความหนา (trace) 0=OFF,1=ON
FLATLAND	0	R.10	SETVAR		ควบคุมฟังก์ชันการทำงานในลักษณะ 3 มิติ
FRONTZ	0.00		DVIEW	RO	ระยะ front clipping ในคำสั่ง DVIEW
GRIDMODE	0	OFF	GRID		ควบคุมการปิดเปิด grid 0=OFF,1=ON
GRIDUNIT	0,0		GRID		ระยะห่างของแนว x,y ของ grid
HANDLES	0	OFF	HANDLES	RO	ควบคุมการจัดเก็บส่วนของภาพในลักษณะฐานข้อมูล 0=OFF,1=ON
HIGHLIGHT	1	ON	SETVAR		ควบคุมให้มีการ mark กับส่วนของภาพที่เลือก 0=OFF,1=ON
INSBASE	0,0		BASE		กำหนดจุด insert basepoint ของ drawing ปัจจุบัน
LASTANGLE	0			RO	ค่ามุมสุดท้ายที่ได้เขียน arc, line

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
LASTPOINT	0,0				ตำแหน่งx,y ล่าสุดที่ได้กำหนดจุด
LASTPT3D	0,0,0				ตำแหน่งx,y,z ล่าสุดที่ได้กำหนดจุด
LENSLENGTH	50.00		DVIEW	RO	ขนาดของเลนส์ที่ใช้ในการแสดงภาพ perspective หน่วยเป็นมิลลิเมตร
LIMCHECK	0	OFF	LIMITS		มีการตรวจสอบ LIMITS 0=OFF,1=ON
LIMMAX	12.0,9.0		LIMITS		ขอบเขต limits มุมขวาบน
LIMMIN	0,0		LIMITS		ขอบเขต limits มุมซ้ายล่าง
LTSCALE	1.00		LTSCALE		แฟกเตอร์ตัวคูณสำหรับสเกลของลายเส้น(linetype)
LUNITS	2	DEC.	UNITS		หน่วยความยาว 1=scientific 2=decimal,3=engineering 4=architec,5=fractional
LUPREC	4	4 ตำแหน่ง	UNITS		จำนวนตำแหน่งจุดทศนิยม
MENUECHO	0	ปกติ	SETVAR		ควบคุมการแสดงคำสั่งเมื่อเลือกใช้เมนู 0=ปกติ(แสดงทุกอย่าง) 1=ไม่แสดงส่วนที่เป็นข้อมูลป้อนเข้าเครื่อง 2=ไม่แสดงคำถามแต่แสดงข้อมูลที่ป้อนเข้าไป 3=ไม่แสดงส่วนที่เป็นคำถามและส่วนที่เป็นข้อมูล
MENUNAME	"ACAD"		MENU	RO	ชื่อ Menu File ที่ใช้อยู่
MIRRTEXT	1	กลับด้าน	SETVAR		ควบคุมการเขียนอักษรเมื่อถูกใช้คำสั่ง MIRROR 0=ไม่เขียนกลับด้าน 1=เขียนกลับด้านเมื่อถูก MIRROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
ORTHOMODE	0	OFF	[^O],F8		ควบคุมให้ลากเส้นแนวดิ่งแนวนอน0=OFF,1=ON
OSMODE	0	NONE	OSNAP		ควบคุม osnap 0=NONE, 1=END POINT,2=MID POINT,4=CENTER, 8=NODE,16=QUARDANT ,32=INTERSECTION,64= INSERT POINT,128= PERPENDICULAR,256= TANGENT,512= NEAREST,1024=QUICK
PDMODE	0	POINT	SETVAR		รูปแบบของการวางจุดด้วย point
PDSIZE	0.00		SETVAR		ควบคุมขนาดของการวางจุดด้วย point
PERIMETER	0.00		AREA		ความยาวรอบรูปที่ได้คำนวณไว้
PICKBOX	3	OFF ON	SETVAR	CFG	ขนาดของกรอบสี่เหลี่ยมใช้สำหรับ select object หน่วยเป็น pixels
POPUPS	1			RO	แสดงสถานะของระบบ AutoCAD User Interface 0 เท่ากับไม่สนับสนุน,1=ใช้ได้
QTEXTMODE	0		QTEXT		ควบคุมQTEXT 0=OFF, 1=ON
REGENMODE	1	ON	REGENAUTO		0=REGENAUTO OFF, 1=REGENAUTO ON
SCREENSIZE				RO	ขนาดความละเอียดของจอ x,y หน่วยเป็น pixels
SKETCHINC	0.1		SKETCH		ควบคุมระยะ sketch record

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
SKPOLY	0	LINE	SETVAR		ควบคุมการใช้แบบเส้นเมื่อใช้คำสั่ง SKETCH 0=LINE, 1=POLYLINE
SNAPANG	0		SNAP		ควบคุมมุมในการหมุน snap
SNAPBASE	0,0		SNAP		จุด basepoint ในการหมุน snap
SNAPISOPAIR	0	LEFT	SNAP,[^E]		ควบคุม isometric plane0=LEFT,1=TOP,2=RIGHT
SNAPMODE	0	OFF	[^B],F9		ควบคุมการปิดเปิด snap 0=OFF,1=ON
SNAPSTYL	0	STANDARD	SNAP		ควบคุมรูปแบบของsnap 0=STANDRD, 1=ISOMETRIC
SNAPUNIT	1,1		SNAP		ระยะห่าง snap x,y
SPLFRAME	0		SETVAR		ควบคุมการแสดงภาพโครงร่าง spline 0=OFF,1=ON
SPLINESEGS	8		SETVAR		ควบคุมจำนวนเซกเมนต์ในแต่ละ spline
SPLINETYPE	6	CUBIC	SETVAR		ควบคุมรูปแบบของเส้นโค้งที่ได้จาก Pedit SPline 5=quadratic B-spline, 6=Cubic B-spline
SURFTAB1	6		SETVAR		จำนวนเซกเมนต์ในการสร้างภาพ 3 มิติทิศทาง M
SURFTAB2	6		SETVAR		จำนวนเซกเมนต์ในการสร้างภาพ 3 มิติทิศทาง M
SURFTYPE	6	CUBIC	SETVAR		ควบคุมรูปแบบของการ smooth พื้นผิวรูป 3 มิติจาก Pedit 5=quadratic B-spline,8=Bizier
SURFU	6		SETVAR		จำนวนเซกเมนต์หลังจาก smooth ภาพ 3 มิติทิศทางM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
SURFV	6		SETVAR		จำนวนเซกเมนต์ทิศทาง N หลังจาก smooth ภาพ 3 มิติ
TARGET	0,0,0		DVIEW	RO	ตำแหน่งโคออร์ดิเนตของเป้าหมายในการมอง
TDCREATE			TIME	RO	วันที่และเวลาที่สร้างดรออิง
TDINDWG			TIME	RO	เวลาทั้งหมดที่ใช้ในการสร้าง/แก้ไขดรออิง
TDUPDATE			TIME	RO	วันที่และเวลาที่ได้ save ดรออิงครั้งสุดท้าย
TDUSRTIMER			TIME	RO	ช่วงเวลาที่ใช้ในการแก้ไข
TEMPREFIX	" "			RO	บอกถึง directory path ที่ใช้เป็นที่เก็บไฟล์ชั่วคราว
TEXTEVAL	0		SETVAR		ควบคุมการตีความหมายของข้อความ(,!ที่ป้อนเข้ามา 0=เป็น text ธรรมดา, 1=เป็นออโตลิปส์
TEXTSIZE	0.2				ความสูงของอักษรที่ใช้สำหรับคำสั่งเขียนอักขระ
TEXTSTYLE	"STANDARD"		TEXT, STYLE	RO	รูปแบบอักษรที่ใช้
THICKNESS	0		ELEV		ค่า thickness ปัจจุบันที่ใช้ในการเขียนภาพ 3 มิติ
TRACEWID	0.05		TRACE		ความกว้างของเส้นที่มีความหนา trace
UCSFOLLOW	0	OFF	SETVAR		ปรับ plane view ตาม UCS ปัจจุบัน 0=OFF,1=ON
UCSICON	1		UCSICON		ควบคุมการแสดงUCSICON 0=OFF,1=ON,2=ORIGIN
UCSNAME	" "		UCS	RO	ชื่อของ UCS ปัจจุบัน
UCSORG	0,0,0		UCS	RO	จุด Origin ของ UCSปัจจุบัน
UCSXDIR	1,0,0		UCS	RO	ทิศทางของแกน x UCS ปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
UCSYDIR	0,1,0		UCS	RO	ทิศทางของแกน y UCS ปัจจุบัน
USERI1-5	0				เป็นตัวแปรสำหรับเก็บค่าจำนวนจริงของUSERI1-USERI5
USERR1-5	0.00				เป็นตัวแปรสำหรับเก็บค่าจำนวนจริงของ USERR1-USERR5
VIEWCTR			ZOOM, PAN, VIEW	RO	จุดศูนย์กลาง x,y สำหรับ view ปัจจุบัน
VIEWDIR	0,0,1		DVIEW	RO	ทิศทางการมองของ view ปัจจุบัน
VIEWMODE	0		DVIEW	RO	ควบคุมการแสดงผลในโหมด DVIEW 1=PERSPECTIVE ,2=FRONT CLIPPING,4=BACK CLIPPING,8=UCS FOLLOW ON,16=FRONT CLIPPING ตำแหน่ง EYE
VIEWSIZE	9.00		ZOOM	RO	ความสูงของ current view โดยวัดจากระยะในแนวแกน Y บนสุดถึงล่างสุด หน่วยที่ได้ขึ้นอยู่กับ units ที่ตั้งไว้
VIEWTWIST	0		DVIEW	RO	มุมในการหมุน(twist view)
VPOINTX	0.00		VPOINT	RO	โคออร์ดิเนต x ของ vpoint
VPOINTY	0.00		VPOINT	RO	โคออร์ดิเนต y ของ vpoint
VPOINTZ	1		VPOINT	RO	โคออร์ดิเนต z ของ vpoint
VSMAX				RO	แสดงตำแหน่งโคออร์ดิเนตมุมขวาบนของ VIRTUAL SCREEN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตัวแปร	ค่าที่กำหนดโดยอัตโนมัติ	ความหมาย	คำสั่งที่เกี่ยวข้อง	สถานะ	รายละเอียด
VSMIN				RO	แสดงตำแหน่งโคออร์ดิเนตมุมล่างซ้ายของ VIRTUAL SCREEN (ขอบเขตของภาพ 3 มิติที่มีอยู่จริงคล้าย EXTMAX และ EXTMIN)
WORLDUCS	1		UCS	RO	แสดงสถานะของ UCS 0= ไม่ใช่ WCS,1= เป็นWCS
WORLDVIEW	1		DVIEW, UCS		ตำแหน่งโคออร์ดิเนตของมุมมอง DVIEW หรือ VPOINT อ้างอิงจาก WCS หรือUCS 0=อ้างอิงจาก UCS,1= อ้างอิงจาก WCS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำโครงงานนี้สำเร็จลุล่วงไปด้วยดีก็เนื่องมาจากหน่วยงานของสถาบันและบุคคลหลายๆฝ่าย ได้แก่ สำนักวิจัยและบริการคอมพิวเตอร์ ที่ให้ความอนุเคราะห์ในการเข้าไปใช้เครื่องคอมพิวเตอร์ ที่สามารถ Support โปรแกรม AutoCAD Release 12 ได้ ซึ่งได้แก่ เครื่อง workstation เพื่อใช้ในการประยุกต์โปรแกรมออโตแคด และให้ความอนุเคราะห์เกี่ยวกับอุปกรณ์ต่างๆ และความสะดวกในการดำเนินงานโครงงานในครั้งนี้ เช่น การใช้เครื่อง Laser printer รวมทั้งเจ้าหน้าที่และพี่ๆที่ดูแลสำนักวิจัย ที่คอยช่วยเหลือและให้คำแนะนำในการให้งานออโตแคดบนระบบยูนิกซ์ ที่มีประสิทธิภาพในการใช้งานมากขึ้น และพี่ๆปริญญาโทที่อนุญาตให้ใช้เครื่องคอมพิวเตอร์ PC ที่มีไว้ให้สำหรับนักศึกษาปริญญาโทใช้

บุคคลผู้ซึ่งมีความสำคัญต่อการทำโครงงานนี้ คือ ท่านอาจารย์ที่ปรึกษา คือ ดร. วรวัฒน์ ลิ้มโภคา ที่คอยให้คำแนะนำและคอยช่วยเหลือในด้านการติดต่อการใช้เครื่อง Computer ทั้งที่เป็น PC และ Workstation และได้ช่วยจัดหาคู่มือที่ใช้สำหรับการศึกษาในครั้งนี้ รวมถึงคอยให้คำปรึกษาแก่คณะผู้จัดทำเป็นอย่างดี ซึ่งคณะผู้จัดทำขอกราบขอบพระคุณทุกท่านที่กล่าวมาเป็นอย่างสูง มา ณ ที่นี้ด้วย

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. พต.ประพัฒน์ อุทโยภาศ และ ฐิติพัฒน์ ประทานทรัพย์,"AutoCAD ฉบับพิสดาร Release 10 ", บริษัท ซีเอ็ดยูเคชัน จำกัด,352 หน้า,2534
2. อนิรุต ลิ่วหาทอง,"พื้นฐานการโปรแกรม AutoLISP,บริษัท ซีเอ็ดยูเคชัน จำกัด,368 หน้า,2536
3. กอบเกียรติ สระอุบล,"Advanced AutoCAD Release 10",บริษัท ซีเอ็ดยูเคชัน จำกัด,400 หน้า, 2535
4. กอบเกียรติ สระอุบล,"การใช้โปรแกรม AutoShade AutoFix",บริษัทซีเอ็ดยูเคชัน จำกัด, 247 หน้า,2535
5. พรศักดิ์ อรรถวานิช,"เขียนแบบวิศวกรรม 1",ภาควิชาเครื่องกล คณะวิศวกรรมศาสตร์ สจล., 239 หน้า,2532
6. Autodesk,Inc. ,"AutoCAD Release 12 Interface Installation , AND Performance Guide", 278 p.,1992.
7. Autodesk,Inc. ,"AutoCAD Release 12 ,Tutorial", 278 p.,1992.
8. Autodesk,Inc. ,"AutoCAD Reference Manual",684 p.,1992.
9. Autodesk,Inc. ,"AutoCAD Customization Manual ",334 p.,1992 .
10. Autodesk,Inc. ,"AutoCAD Extras Manual ",788 p.,1992.
11. Autodesk,Inc. ,"AutoCAD Render Reference Manual ",168 p.,1992 .
12 Autodesk,Inc. ,"AutoLISP Programmer 's Reference Manual ",249 p.,1992 .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้