



คอมพิวเตอร์กับผลผลิตในโรงงาน

COMPUTER INTEGRATED MANUFACTURING



โดย

นางสาวภิญณี คุงกะเกษกริน 821281

นางสาว อรวรรณ องบรรณจิต 821421

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร ปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาคอมพิวเตอร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มี

032710

บทคัดย่อ

Computer Integrated Manufacturing (CIM) เป็นการนำคอมพิวเตอร์มา ใช้ในขบวนการผลิตของ โรงงาน ลักษณะของระบบนี้เป็นการเก็บรวบรวมข้อมูลสถานะต่าง ๆ ของเครื่องจักรในขณะปัจจุบัน เพื่อสามารถนำข้อมูลนั้น มาช่วยในการตัดสินใจของฝ่ายจัดการได้ทันตามความต้องการ การรวมทั้งเก็บรวบรวมข้อมูลในอดีตเพื่อใช้ในการพิจารณากระบวนการผลิตในระยะยาวต่อไป การอ่านข้อมูลจากเครื่องจักรเข้ามาสู่ระบบ คอมพิวเตอร์นั้น จะใช้ Programmable Logic Controller (PLC) เป็นตัวติดต่อระหว่างเครื่องจักรและ คอมพิวเตอร์ ข้อมูลที่อ่านได้ เช่น ผลผลิตที่ผลิตได้ในแต่ละกะการทำงาน ความเร็วของเครื่องจักร จะถูกนำมาประมวล ผลในรูปแบบต่าง ๆ ตามความต้องการของฝ่ายจัดการ การพัฒนาระบบนี้กระทำบน Local area network (LAN) เพื่อให้ได้ข้อมูลใกล้เคียงกับการผลิตในปัจจุบันที่สุด คือ สถานะของเครื่องจักรขณะทำงาน ถูกนำมาแสดงผล บนจอคอมพิวเตอร์ในแต่ละ work station ภายในออฟฟิศ

ABSTRACT

Computer Integrated Manufacturing (CIM) is a system which assists production in plants. This system obtains information from machines at the moment time so that it is used by production managers for immediate decisions. And keep information in the past to be considered in developing productivity in the future. We can obtain data from machines by Programmable Logic Controller (PLC). The data is read by PLC for instance product in each shift, speed of machine they are processed in the specified type that managers need. This system is developed on Local Area Network (LAN) to get information approach production in the present. Status of working machine is demonstrated on screen of computer in each workstation at office.

คำนำ

รายงานฉบับนี้เป็น ประเด็นที่เกี่ยวกับ การนำคอมพิวเตอร์มาใช้ในขบวนการผลิตภายในโรงงานมีจจุบันได้มีการนำคอมพิวเตอร์มาใช้ในโรงงานอย่างกว้างขวางเพราะความสะดวกรวดเร็วและมีความถูกต้องกว่าใช้คนทำ ดังนั้นจึงมีความคิดที่จะนำคอมพิวเตอร์มาใช้ควบคุมผลผลิตในโรงงาน แทนการใช้คนจดบันทึกอีกทั้งยังสามารถนำข้อมูลเหล่านั้นมาประมวลผล และ เก็บรวบรวมเป็นข้อมูลในอดี้อีกด้วย

รายงานนี้ เป็นส่วนหนึ่งของวิชา Project II ภาคเรียนที่ 2 ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2535

คณะผู้จัดทำ

สารบัญ

เนื้อหา	หน้าที่
FOXPRO/LAN V 2.0	1
การสร้างและแสดงฐานข้อมูล	3
การค้นหาข้อมูล	4
การ Index ใน Foxpro 2.0	5
การจัดการ Window	5
การทำรายงานออกทางเครื่องพิมพ์	7
การรับข้อมูลและการแสดงผลข้อมูล	7
WATCOM C	15
PLC	22
การทำงานในช่วง Realtime	25
การนำเสนอข้อมูล	26
DATABASE	29
Flowchart การทำงานของโปรแกรม	30

FOXPRO/LAN V 2.0

Foxpro/LAN version 2.0 เป็นโปรแกรมที่พัฒนาโดยบริษัท Fox software เป็นโปรแกรมที่ทำงานได้รวดเร็ว มีสถานะ แวดล้อมสำหรับการพัฒนาโปรแกรมแบบผู้ใช้หลายคนที่ยอดเยียม และมีถึงอำนาจความสะดวกมากมาย

ระบบที่ต้องการ

Foxpro/LAN 2.0 สามารถใช้ได้ทั้งระบบผู้ใช้เดี่ยว (Single user) และระบบเครือข่าย (Local Area Network)

ซึ่งต้อง การอุปกรณ์ดังนี้

SingleUser	LAN
1. ใช้กับเครื่องไมโครคอมพิวเตอร์ IBM PC/XT AT PS/2, compag despro ,Portable II Portable III หรือ 386 หรือ เครื่องที่ compatible	1. ใช้กับเครื่องไมโครคอมพิวเตอร์ IBM PC/XT AT PS/2, compag despro ,Portable II Portable III หรือ 386 หรือ เครื่องที่ compatible.
2. มีหน่วยความจำอย่างน้อย 512 kbyte ถ้าเป็นแบบ Entend ต้องมีหน่วยความจำอย่างน้อย 1.5 MB	2. บนเครื่อง Work station ต้องมีหน่วยความจำอย่างน้อย 512 KB
3. มี Disk Drive อย่างน้อย 1 ตัว และมี harddisk	3. บน work ststion ไม่จำเป็น ต้องมี Disk Drive หรือ Harddisk
4. สามารถใช้ Mouse ได้	4. สามารถใช้ Mouse ได้
5. ใช้ระบบ PC DOS หรือ MS DOS หรือ MS DOS Version 2.10 ขึ้นไปหรือ OS/2 Version 1.0 หรือสูงกว่า	5. ใช้ระบบ PC DOS หรือ MS DOS Version 3.10 ขึ้นไปหรือ OS/2 Version 1.0 หรือสูงกว่า และระบบจัดการบน LAN ที่สามารถใช้ได้
6. สามารถใช้หน่วยความจำพิเศษต่างๆ ได้ เช่น AST rampage, intel above หรือ lim 4.0 ems ได้	6. สามารถใช้หน่วยความจำพิเศษต่างๆ ได้ เช่น AST rampage, intel above หรือ lim 4.0 ems ได้
7. จอภาพแบบ Monochrome หรือ จอสี แบบ EGA, CGA, VGA	7. จอภาพแบบ Monochrome หรือ จอสี แบบ EGA, CGA, VGA
8. เครื่องพิมพ์ (Printer) ขนาด จำนวน 80 ตัวอักษร/บรรทัดหรือมากกว่าหรือ เครื่องที่ทัด เทียมกัน	8. เครื่องพิมพ์ (Printer) ขนาด จำนวน 80 ตัวอักษร/บรรทัดหรือมากกว่าหรือ เครื่องที่ทัด เทียมกัน

ขีดความสามารถของ FOXPRO/LAN 2.0

Database & Index	Standard	Extend
จำนวนสูงสุดของ Record/file	1 พันล้าน	1 พันล้าน
จำนวนสูงสุดของตัวอักษร/ Record	4,000	4,000
จำนวนสูงสุดของ field/record	255	255
จำนวนสูงสุดของ file ที่เปิด/ครั้ง	25	25
จำนวนสูงสุดของตัวอักษร/Field	254	254
จำนวนสูงสุดของความยาว Key index.idx	100	100
จำนวนสูงสุดของความยาว Key index.cdx	254	254
จำนวนสูงสุดของ file index ที่เปิด/file	ไม่จำกัด	ไม่จำกัด
จำนวนสูงสุดของ file index ที่เปิดใช้ทั้งหมด	ไม่จำกัด	ไม่จำกัด
Field Characteristics		
จำนวนหลักสูงสุดของตัวเลข/field	20	20
ความยาวสูงสุดของชื่อ field	10	10
นัยสำคัญของการคำนวณทางตัวเลข	16	16
Memory Variables & Arrays		
ค่าปกติของตัวแปรหน่วยความจำ	256	256
จำนวนสูงสุดของตัวแปรหน่วยความจำ	3,600	3,600
จำนวนสูงสุดของตัวแปรแบบ array	3,600	6,500
จำนวนสูงสุดของสมาชิกใน array	3,600	6,500
Program & Procedure		
จำนวนบรรทัดคำสั่งสูงสุดของ File Program	ไม่จำกัด	ไม่จำกัด
ขนาดสูงสุดของ Program Compile Module	64 KBytes	64 KBytes
จำนวนคำสั่ง Do ที่ซ้อนกันได้	32	32
จำนวนคำสั่ง READ ที่ซ้อนกันได้	5	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนสูงสุดคำสั่งที่เป็นโครงสร้าง	64	64
Report Writer Capacity		
จำนวนสูงสุดขององค์ประกอบในรายงาน	ไม่จำกัด	ไม่จำกัด
จำนวนบรรทัดสูงสุดในการกำหนดรายงาน	255	255
จำนวนสูงสุดของการจัดกลุ่มข้อมูล	20	20
Window Support		
จำนวนสูงสุดของ Window ที่เปิดใช้งาน	ไม่จำกัด	ไม่จำกัด
จำนวนสูงสุดของ Browse Window	25	25
Miscellaneous Capacities		
จำนวนสูงสุดของตัวอักษร/ตัวแปรแบบ String	64 KBytes	2 Gbytes
จำนวนสูงสุดของอักษร/บรรทัดคำสั่ง	2,048	2,048
จำนวนสูงสุดของ File ที่เปิด	99	จำกัดที่ Dos
จำนวนสูงสุดของการกด Keyboard/Macro	1,024	1,024
จำนวน Field สูงสุด/คำสั่ง Select(SQL)	255	255
Color Support		
จำนวนของ Scheme/Color Set	24	24
จำนวนสูงสุดของ Color Set	ไม่จำกัด	ไม่จำกัด
จำนวนสูงสุดของสีต่อ Color Screen	10	10
จำนวน Scheme ที่ให้ผู้ใช้เปลี่ยนสีได้	8	8

การสร้างและแสดงพื้นฐานข้อมูล

การสร้างพื้นฐานข้อมูล FoxPro ขอมให้สร้างพื้นฐานข้อมูลได้ 2 วิธี

- 1 ป้อนคำสั่งที่รองรับคำสั่ง (Command Window) โดยการพิมพ์คำสั่งในรูปแบบคำสั่ง CREATE

(filename)

- 2 ใช้คำสั่งโดยการเปิดหน้าต่าง <File> จากรายการหลัก <NEW> <DATABASE> <<OK>>

ในการตั้งชื่อ

Field มีกฎการตั้งชื่อดังนี้

- ใช้ตัวหนังสือ A-Z,a-z หรือตัวเลข (0-9) และเครื่องหมาย "_" (Underscore)
- ห้ามใช้ Space Bar ในการตั้งชื่อ
- ชื่อ Field มีความยาวได้ 10 ตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประเภทของ Field ใน Foxpro มีให้ใช้ดังนี้

Field

- Character Field ใช้เก็บข้อมูลประเภทตัวอักษร ซึ่งไม่ได้ใช้ในการคำนวณสามารถกำหนดความกว้างประเภทนี้ได้ 256 ตัวอักษร

- Date Field ใช้เก็บวันที่ที่มีรูปแบบของอเมริกันเป็น mm/dd/yy

และมีทศ

- Numeric Field ใช้เก็บข้อมูลที่เป็นจำนวนที่ใช้ในการคำนวณสามารถเก็บจำนวนได้ 20 หลัก
นิยม ที่มีความละเอียด 15 หลัก และยอมรับจำนวนลบได้

- Floating Point มักอยู่ในรูปแบบของจำนวนเลขที่คูณกับ 10 ยกกำลัง

- Logical Field ใช้เก็บข้อมูลทางตรรกศาสตร์คือเป็นจริงกับเท็จเพียง หลักเดียวเท่านั้น

- Memo Field ใช้สำหรับบรรทัดข้อความที่ยาวมากๆ ไม่จำกัดความยาว

- Picture Field

การแสดงผลพื้นฐานข้อมูล ด้วยวิธี Browse

คำสั่ง Browse เป็นการสั่งให้ Foxpro แสดงข้อมูลใน Database ที่กำลัง Active และสามารถแสดงได้พร้อมกันที่ เดียวหลาย Database โดยคำสั่ง Browse มี Option ให้ใช้ดังนี้

- Field List เป็นการระบุชื่อ Field ที่ต้องการให้แสดง ถ้าไม่ระบุจะแสดงหมดทุก Field

และ

- Field Control เป็นการกำหนดลักษณะพิเศษของแต่ละ Field ที่แสดงเช่น Read-only, Validated, formatted, size

- FOR condition การแสดงจำนวน record จะแสดงตามเงื่อนไข FOR

Expression นี้

- WIDTH expression จำกัดจำนวนของ Character ที่แสดงใน Browse Window ถ้าไม่ได้กำหนดใน

ความกว้างจะทำกับขนาดของ Fields ที่กำหนดไว้ในตอนสร้าง DBF

- FREEZE field กำหนดให้ Field เพียง 1 Field เท่านั้นที่สามารถจะทำการแก้ไขได้

- LOCK expression กำหนดจำนวนของ Fields ที่อยู่ใน Browse ทางด้านซ้ายมือ

- NOCLEAR กำหนดให้ยังคงแสดงข้อมูลใน Browse หลังจากที่ย่อจากการ Browse แล้ว

- COLOR SCHEME | COLOR color pair list กำหนดสีใน Browse

- NOAPPEND กำหนดไม่ให้มีการ Append Record ใหม่

- NODELETE กำหนดไม่ให้ Delete Record

- NOEDIT กำหนดไม่ให้แก้ไขข้อมูลใน Data Base

การค้นหาข้อมูล

สำหรับการค้นหาข้อมูลใน Foxpro สามารถทำได้หลายวิธีคือ

1. ทำการค้นหาด้วยคำสั่งที่มีอยู่ใน Menu เป็นวิธีสำหรับคนที่จำคำสั่งไม่ได้

2. โดยใช้คำสั่งของ SQL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. โดยการพิมพ์คำสั่งลงใน Command Window ซึ่งคำสั่งสำหรับการค้นหาข้อมูล
ที่ Foxpro มีให้ใช้มีดังนี้คือ

- CONTINUE ให้ทำการค้นหาใน Record ถัดไปตามคำสั่ง LOCATE
- GOTO เลื่อน File Pointer ไปที่ Record Number ที่ต้องการ
- LOCATE ค้นหาข้อมูลสำหรับ Record แรกที่ตรงกับเงื่อนไข
- SEEK ค้นหาข้อมูลสำหรับ Record แรกที่ตรงกับเงื่อนไข สำหรับ Database ที่ทำการ INDEX แล้ว

การ INDEX ใน FoxPro 2.0

ก่อนหน้านี้ FoxPro จะมีการสร้าง INDEX File ได้แบบเดียว เมื่อสร้างแล้วจะเป็น INDEX File ที่เป็น .IDX.
แต่ใน เวอร์ชัน 2.0 สร้าง INDEX File ได้ 2 แบบคือ แบบ Single INDEX และแบบ Compound INDEX จุดเด่นของ
Compound

INDEX คือ สามารถสร้าง INDEX File ให้มีคีย์สำหรับควบคุมข้อมูลได้หลาย ๆ คีย์ อยู่ใน INDEX File
เดียวกันนอกจากนี้ ยังมีการกำหนดลักษณะพิเศษของ INDEX File ได้ อีก 3 แบบคือ

1. Compact จุดเด่นของ INDEX File แบบนี้คือ กินเนื้อที่ของ LIST น้อยมาก โดยใช้เนื้อที่เพียง 1/6 ของ
INDEX File
2. Structural INDEX File แบบนี้จะมีชื่อเหมือนกับ DATA BASE File ที่เป็นคู่กันอยู่และเมื่อมีการเปิด
DATA BASE File INDEX File ก็จะถูกเปิดขึ้นโดยอัตโนมัติ
3. Non Compact สำหรับแบบนี้เป็นแบบเดิมที่ไม่กระชับ เพราะถ้าต้องการใช้กับโปรแกรมที่เขียนด้วย
FoxPro รุ่น ก่อน ๆ จะต้องสร้าง INDEX File ด้วยการระบุให้เป็นแบบ NON Compact ถึงจะใช้ร่วมกันได้

การจัดการ MENU

การสร้าง MENU มีคำสั่งที่อำนวยความสะดวก และมีทางเลือกได้มากมาย คำสั่ง DEFINE MENU สำหรับ
MENU BAR และ DEFINE POPUP สำหรับ POPUP MENU และยังสามารถกำหนด HOT KEY ได้ด้วยคำสั่งต่าง ๆ
สำหรับใช้ใน การเขียนโปรแกรมทำ MENU มีมาก และง่ายแต่ Foxpro มีวิธีสร้าง MENU ที่ง่ายกว่านั้นอีกคือ MENU
เพียงผู้ใช้ออกแบบ ใน window Menu Builder แล้วทำการ Generate ก็สามารถนำ Menu
มาใช้ได้โดยไม่ต้องเขียนโปรแกรมเอง นอกจากนี้ยังสามารถนำ System Menu มาใช้ได้ด้วย

การจัดการ WINDOW

โปรแกรมเมอร์สามารถสร้าง WINDOW เชิงอักษร (TEXT Oriented WINDOW)
ซึ่งสามารถกำหนดคุณสมบัติต่าง ๆ ของ WINDOW ได้เช่น

FLOATINO FLOAT กำหนดให้สามารถที่จะเคลื่อนย้าย WINDOW ได้หรือไม่

SHADOW กำหนดให้ WINDOW ที่สร้างมีเงาทำให้ดูคล้ายกับ WINDOW ลอยขึ้นมาจาก BACKGROUND ด้านหลังได้

ZOOMINOZOOM กำหนดให้ผู้ใช้สามารถที่จะขยาย WINDOW ที่สร้างขึ้นเป็น Full Screen

COLOR สามารถกำหนดสีของส่วนต่าง ๆ ของ WINDOW และ Process ที่อยู่ใน WINDOW นั้นได้

TITLE กำหนดข้อความที่จะแสดงตรงส่วนบนของ WINDOW เพื่อแสดงความสัมพันธ์ของ Process ที่มีอยู่ใน WINDOW ต่าง ๆ

FOOTER กำหนดข้อความที่จะแสดงตรงส่วนล่างของ WINDOW ซึ่งมีการใช้งานคล้ายกับ TITLE

IN WINDOW กำหนดให้ WINDOW ที่สร้างขึ้นเป็น CHILD ของ PARENT WINDOW ได้

DOUBLEIPANELINONEISYSTEM WINDOW ที่สร้างขึ้นสามารถที่จะกำหนดลักษณะของขอบได้เป็น 4

แบบ

CLOSEINOCLOSE กำหนดให้ผู้ใช้สามารถที่จะปิด WINDOW ได้หรือไม่ได้

GROWINOGROW กำหนดให้ผู้ใช้สามารถที่จะย่อหรือขยาย WINDOW

MINIMIZE กำหนดให้ผู้ใช้สามารถที่จะเปลี่ยน WINDOW ให้เป็นแบบ ICON ได้

การนำเข้าและส่งออกข้อมูล

เป็นการนำข้อมูลจาก โปรแกรมอื่นที่ใช้งานอยู่ มาใช้ใน FoxPro 2.0 ตัวอย่างเช่น File ข้อมูลในโปรแกรม

LOTUS 123 VERSION 1-A, 2.x และ 3.x, SYMPHONY FRAME WORK II, Microsoft Multiplan, Excel VERSION 2, Paradox, เพิ่มข้อมูลแบบ DIF (DATA INTERCHANGE FORMAT), เพิ่มข้อมูลแบบ SYLK (SYMBOLIC LINK INTERCHANGE FORMAT)

SCREEN BUILDER

ในการออกแบบหน้าจอที่ติดต่อกับผู้ใช้ ไม่ว่าจะเป็นการรับข้อมูลหรือการแสดงผลนั้น FoxPro 2.0 มี Utility ที่ช่วย งานทางด้านนี้ โดยที่โปรแกรมเมอร์สามารถออกแบบหน้าจอว่าต้องการให้ออกมา ในรูปแบบใดเก็บไว้ใน File.scx โดยไม่ จำเป็นต้องเขียนเป็นโปรแกรม เมื่อได้หน้าจอตามที่ต้องการแล้วจึงนำเอา File.scx ไป Generate ให้เป็น Program File จึง สามารถรันได้ Object ต่างๆที่ Screen Builder มีให้ออกแบบได้คือ

- Text สามารถเขียนข้อความต่างๆ บนหน้าจอได้ตามต้องการ
- Fields สามารถที่นำ Database File มาร่วมในการออกแบบ Screen ได้
- Window Programmer สามารถสร้าง Window ได้
- Push Buttons เป็นการสร้างส่วนที่ให้ Interact กับ User โดยให้ User เลือกตอบได้ เช่น << Ok >> ,

< Cancel >

- Radio Buttons เป็นการเลือก Choice โดยสามารถที่จะเลือกได้ครั้ง 1 ข้อ

- Check Box เป็นการเลือก Choice โดยสามารถที่จะเลือกได้ครั้งละหลายข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Popup เป็นการเลือก Choice โดยสามารถที่จะเลือกได้ครั้งละ 1 ข้อแบบเดียวกับ Radio Button แต่มีลักษณะ เหมือนกับ Popup ทั่วไป

- Scrolling List เป็นการแสดงข้อความในขอบเขตที่กำหนด ข้อความอาจจะเป็นตัวแปรแบบ Array หรือข้อมูลใน DATA BASE โดยถ้าไม่สามารถที่จะแสดงข้อความทั้งหมดในขอบเขตที่กำหนดได้ ก็จะมี Scroll Bar ให้ผู้ใช้เห็น

การทำรายงานออกทางเครื่องพิมพ์

การทำรายงานออกทางเครื่องพิมพ์นั้น Foxpro/LAN V2.0 มีวิธีทำได้ 2 วิธีคือ

1. โดยใช้ Report generator ของ Foxpro วิธีนี้สามารถทำได้ง่ายและสะดวก เพราะ Foxpro สามารถให้ Set ค่าและ ข้อกำหนดต่างๆที่ต้องการได้โดยง่ายเช่น จำนวนบรรทัดต่อหน้า ต้องการให้ Printer Eject ก่อนพิมพ์ข้อมูลหรือหลังพิมพ์ข้อมูล หรือไม่ Eject เลย Top margin, Bottom margin เป็นต้น การออกแบบรายงานก็ทำได้สะดวก คือ Report generator จะแบ่งรายงานออกเป็น 3 ส่วนคือ PgHead Detail และ PgFoot

- PgHead และ PgFoot คือส่วนที่จะพิมพ์เป็นหัวกระดาษ และท้ายกระดาษของทุกๆ หน้า
- Detail เป็นส่วนที่รายละเอียดของรายงานส่วนนี้อาจเป็น Variable หรือ Database Fields ก็ได้

เมื่อออกแบบเสร็จ แล้ว Foxpro จะเป็น File ของรายงาน เป็น .FRX เมื่อต้องการใช้ใน Program ก็เพียงแค่เรียกใช้ File นี้เท่านั้น

2. โดยการเขียน Program ขึ้นเอง ซึ่งใน Foxpro ก็มีคำสั่งต่างๆ ให้ใช้เช่น

- \I\ Outputs lines to printer
- ? แสดงค่าของ Expression บน line ถัดไป
- ?? แสดงค่าของ Expression บน line เดียวกัน
- @...SAY ส่ง Output ที่เป็นรูปแบบที่ต้องการออกไปที่ Screen หรือ Printer
- DEFINE BOX เขียน BOX รอบๆตัว text
- Eject ส่ง Form feed ไปให้เครื่องพิมพ์
- PCOL() Return Column ปัจจุบันของเครื่องพิมพ์
- PROW() Return Row ปัจจุบันของเครื่องพิมพ์
- SET MARGIN กำหนดขอบซ้ายของเครื่องพิมพ์

การรับ Input และการแสดง Output

ใน foxpro มีคำสั่งสำหรับใช้ในการแสดงข้อความที่ต้องการออกทางจอภาพ และสามารถควบคุมตำแหน่งที่จะแสดง ได้นั้นคือคำสั่ง SAY และในการรับข้อมูลจากผู้ใช้จะ ใช้คำสั่ง GET ร่วมกับ READ INPUT, ACCEPT

- SAY ,GET มีรูปแบบการใช้ดังนี้คือ

@ <row,column>

[SAY <expC1>

[PICTURE <expC2>]

[FUNCTION <fcodes1>]

[COLOR SCHEME <exprN1>]

|COLOR <color pair list1>]]

[GET <var>

[PICTURE <expC2>]

[FUNCTION <fcodes2>]

[DEFAULT <expr2>]

[ENABLE | DISABLE]

[MESSAGE <expC3>]

[[OPEN] WINDOW <window name>]

[RANGE [<expr3>

[, <expr4>]

[SIZE <expN2>,<expN3>]

[VALID <expL1> | <expN4>]

[WHEN <expL2>]

[COLOR SCHEME <expN5>

| COLOR <color pair list>]

OPTION PICTURE มีรายละเอียดดังนี้

A รับข้อมูลเฉพาะตัวอักษรเท่านั้น

L รับข้อมูลเฉพาะ Logic เท่านั้น

N รับข้อมูลเฉพาะตัวอักษรและตัวเลขเท่านั้น

X รับข้อมูลทุกชนิด

Y รับข้อมูลที่เป็น Logic, Y, y, N, n เท่านั้น และจะเปลี่ยน y และ n ให้เป็น Y และ N ตามลำดับ

9 ถ้าเป็นข้อมูลชนิด Character จะรับข้อมูลที่เป็นตัวเลขเท่านั้น ถ้าเป็นข้อมูลชนิด Numeric

จะรับข้อมูลเป็น ตัวเลขและเครื่องหมายบวกหรือลบเท่านั้น

รับข้อมูลที่เป็นตัวเลข, Blank และเครื่องหมายบวกหรือลบ เท่านั้น

! จะเปลี่ยนอักษรแบบ Lower case ไปเป็น Uper case

ร แสดงสัญลักษณ์ที่ถูกเซทไว้ในคำสั่ง SET CURRENCY

* แสดงเครื่องหมาย * หน้าค่าที่เป็น Numeric กำหนดตำแหน่งของเครื่องหมายจุดทศนิยม , ใส่เครื่องหมาย , ในตัวเลขที่อยู่ทางด้านซ้ายมือของจุดทศนิยม

OPTION FUNCTION

- A รับข้อมูลเฉพาะตัวอักษรเท่านั้น
- B กำหนดให้ข้อมูลที่เป็น Numeric อยู่ชิดด้านซ้าย
- C กำหนดให้มี CR (Credit) อยู่หลังจากตัวเลขที่เป็นบวก
- D กำหนดให้ใช้รูปแบบการกรอกวันที่ ตามแบบที่เซทไว้ในคำสั่ง SET DATE format
- E เขียนวันที่ตามแบบของ European Type
- I แสดงตัวอักษรให้อยู่กึ่งกลาง Field
- J แสดงตัวอักษรให้อยู่ชิดขวาของ Field
- T คัด Blank ที่อยู่ข้างหน้าและข้างหลังข้อมูลที่อยู่ใน Field

Color Table by Color Pair

Color Schemes

User Wind	Scheme 1
User Menus	Scheme 2
Menu Bar	Scheme 3
Menu Pops	Scheme 4
Dialogs	Scheme 5
Dlog Pops	Scheme 6
Alert	Scheme 7
Windows	Scheme 8
Wind Pops	Scheme 9
Browse	Scheme 10
Report	Scheme 11
Alert Pops	Scheme 12

User Winds (Scheme 1)

Color Pair 1 SAY field

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032719

- Color Pair 2 GET field
- Color Pair 3 Border
- Color Pair 4 Title, active
- Color Pair 5 Title, idle & message
- Color Pair 6 Selected item
- Color Pair 7 Clock, Hot keys
- Color Pair 8 Shadow
- Color Pair 9 Enabled ctrl.
- Color Pair 10 Disabled ctrl.

User Menus (Scheme 2)

- Color Pair 1 Disabled
- Color Pair 2 Enabled opt.
- Color Pair 3 Border
- Color Pair 4 Menu titles
- Color Pair 5 Message
- Color Pair 6 Selected opt.
- Color Pair 7 Hot keys
- Color Pair 8 Shadow
- Color Pair 9 Enabled ctrl.
- Color Pair 10 Disabled ctrl.

User Menus (Scheme 3)

- Color Pair 1 Disabled pads
- Color Pair 2 Enabled pads
- Color Pair 3 Border
- Color Pair 4 Title
- Color Pair 5 Message
- Color Pair 6 Selected pad
- Color Pair 7 Hot keys
- Color Pair 8 Shadow
- Color Pair 9 Enabled ctrl.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Color Pair 10 Disabled ctrl.

Menu Pops (Scheme 4)

Color Pair 1 Disabled opt.

Color Pair 2 Enabled opt.

Color Pair 3 Border

Color Pair 4 Title

Color Pair 5 Message

Color Pair 6 Selected opt.

Color Pair 7 Hot keys

Color Pair 8 Shadow

Color Pair 9 Enabled ctrl.

Color Pair 10 Disabled ctrl.

Dialogs (Scheme 5)

Color Pair 1 Normal text

Color Pair 2 Text box

Color Pair 3 Border

Color Pair 4 Title

Color Pair 5 Message

Color Pair 6 Selected opt.

Color Pair 7 Hot keys

Color Pair 8 Shadow

Color Pair 9 Enabled ctrl.

Color Pair 10 Disabled ctrl.

Dialog Pops (Scheme 6)

Color Pair 1 Disabled opt.

Color Pair 2 Enabled opt.

Color Pair 3 Border*

Color Pair 4 Title

Color Pair 5 Message

Color Pair 6	Selected item
Color Pair 7	Hot keys
Color Pair 8	Shadow
Color Pair 9	Enabled ctrl.
Color Pair 10	Disabled ctrl.

Alert (Scheme 7)

Color Pair 1	Normal text
Color Pair 2	Text box
Color Pair 3	Border
Color Pair 4	Title
Color Pair 5	Message
Color Pair 6	Selected item
Color Pair 7	Hot keys
Color Pair 8	Shadow
Color Pair 9	Enabled ctrl.
Color Pair 10	Disabled ctrl.

Windows (Scheme 8)

Color Pair 1	Normal text***
Color Pair 2	Text box***
Color Pair 3	Border
Color Pair 4	Title, active
Color Pair 5	Title, idle
Color Pair 6	Selected text
Color Pair 7	Hot keys
Color Pair 8	Shadow
Color Pair 9	Enabled ctrl.
Color Pair 10	Disabled ctrl.

Wind Pops (Scheme 9)

Color Pair 1	Disabled opt.
--------------	---------------

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Color Pair 2	Enabled opt.
Color Pair 3	Border*
Color Pair 4	Title
Color Pair 5	Message
Color Pair 6	Selected opt.
Color Pair 7	Hot keys
Color Pair 8	Shadow
Color Pair 9	Enabled ctrl.
Color Pair 10	Disabled ctrl.

Browse (Scheme 10)

Color Pair 1	Other records
Color Pair 2	Curr. field
Color Pair 3	Border*
Color Pair 4	Title, active
Color Pair 5	Title, idle
Color Pair 6	Selected text
Color Pair 7	Curr. record**
Color Pair 8	Shadow
Color Pair 9	Enabled ctrl.
Color Pair 10	Disabled ctrl.

Report (Scheme 11)

Color Pair 1	Text & B full
Color Pair 2	Report field
Color Pair 3	Border
Color Pair 4	Title, active
Color Pair 5	Title, idle
Color Pair 6	Selected item
Color Pair 7	Band A, empty
Color Pair 8	Shadow
Color Pair 9	Band A, full
Color Pair 10	Band B, empty

Alert Pops (Scheme 12)

Color Pair 1	Disabled opt.
Color Pair 2	Enabled opt.
Color Pair 3	Border*
Color Pair 4	Title
Color Pair 5	Message
Color Pair 6	Selected opt.
Color Pair 7	Hot keys
Color Pair 8	Shadow
Color Pair 9	Enabled ctrl.
Color Pair 10	Disabled ctrl.

EXE COMPILER

การเขียนโปรแกรมใน FoxPro 2.0 โปรแกรมที่เขียนขึ้นนั้นจะสามารถ Compile ให้เป็น Executable File คือ สามารถเรียกใช้ที่ DOS โดยไม่ต้อง RUN FoxPro ก่อน ซึ่งสามารถเลือกสร้างได้ 3 แบบคือ 1. Compact Executable File คือ โปรแกรม Execute File แบบกระชับ ขนาดของ File จะเล็กมาก แต่ต้องใช้ร่วมกับ Library File ที่มีขนาด 1.2 Mbyte 2. Stand Alone Executable File คือ โปรแกรมที่เป็น .EXE สามารถทำงานด้วยตัวเองอย่างอิสระ แต่ File ชนิดนี้มีขนาดใหญ่ มาก โดยจะเริ่มต้นที่ขนาด 656 Kbyte 3. Stand Alone Executable File Extended

การล็อก File โดยอัตโนมัติ

Foxpro/LAN รองรับการทำงานแบบใช้งานพร้อมกัน โดยการล็อก File และ Record ซึ่งก่อให้เกิด exclusive write คือ Foxpro จะทำการล็อก Record หรือ File ให้ถ้าผู้ใช้ต้องการเขียนข้อมูล ดังนั้นผู้ใช้อื่นๆ จะมีสิทธิ์เพียงอ่าน File หรือ Record นั้นเท่านั้น ทั้งนี้เพื่อป้องกันการเขียนข้อมูลลงในตำแหน่งเดียวกันพร้อมกัน Foxpro/LAN ยังสามารถทำการล็อก File หรือ Record แบบอัตโนมัติทันทีที่คำสั่งบางคำสั่งถูก เรียกใช้และจะคลายล็อกเองเมื่อสิ้นสุดคำสั่ง และผู้ใช้ยังสามารถล็อก File หรือ Record เองได้โดยใช้คำสั่ง FLOCK() และ RLOCK() ตามลำดับ และสามารถล็อกหลาย Record ในเวลาเดียวกัน โดยระบุจำนวน Record ในฟังก์ชัน RLOCK()

การป้องกันการเกิด Deadlock

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การป้องกันการเกิด Deadlock ทำได้โดยการใช้คำสั่ง SET REPROCESS โดยปกติแล้วถ้ามีการพยายามล็อก File ที่ถูกล็อกอยู่ก่อนแล้ว Foxpro จะแจ้ง Error Message ให้ทราบและสามารถบอกให้ Foxpro ทำการล็อกซ้ำ โดยอัตโนมัติ ในจำนวนครั้งที่ระบุ หรือในช่วงเวลาที่กำหนด

WATCOM C

API Distribution kit

การพัฒนา ระบบ CIM นี้ เนื่องจากเราใช้ Foxpro เป็นตัวจัดการฐานข้อมูล ดังนั้นเราจึงต้องมีขั้นตอนการติดต่อเพื่อนำข้อมูลที่อ่านจาก PLC ซึ่งเขียนโดยภาษาซีเข้ามาเก็บใน database ที่จัดการโดย Foxpro

การติดต่อของ Foxpro กับภาษาซี

การที่ Foxpro จะติดต่อกับภาษาซีได้นั้น เราใช้ Application ที่เรียกว่า API (Application Program Interface) API ถูกใช้ในด้านเพื่อให้ user ทำงานด้านเฉพาะที่ต้องการโดยไม่จำเป็นต้องทำงานโดยทั่วไปใน Package ชนิดนั้น

ลักษณะของ API

ในส่วนของ API จะเป็นรูปแบบของ re-useable code สำหรับ Application ของเรา concept ในการทำงานคือ เป็นการเรียก external code จากภายใน Package ซึ่ง code เหล่านี้เป็นรูปแบบของ .BIN (Binary file) แต่ละ ส่วนของ code จะเป็นการทำงานเฉพาะส่วน ตัวอย่างเช่น การแปลงตัวเลขเป็นสตริง

การทำงานของ API จะสร้าง Stack หรือ queue ของพารามิเตอร์ และส่งค่า address ใน memory ของ stack ที่เป็น จุดเริ่มต้นของ API function ซึ่ง address นี้ถูกเรียกว่า Pointer อีกนัยหนึ่งเราถือว่า pointer เป็น Mailbox ซึ่งเป็นตัวบอกให้ Software ไปดูค่าที่ตำแหน่งนั้นใน memory สำหรับข้อมูลโดยทั่วไปใน stack ประกอบด้วย

- จำนวนของ Parameter
- Pointer (Mailbox) ของแต่ละพารามิเตอร์ใน stack

API ใช้ Stack ในการบอกว่ามี Parameter ที่ตัวในการส่งผ่านไป รวมทั้งบอกชนิดของข้อมูลความยาวและฟังก์ชัน ที่จำเป็นต่างๆ นั่นคือการทำงานของ API สามารถทำงานในด้าน

1. เรียกโปรแกรมซึ่งสามารถผ่านค่าไปให้ได้
2. สามารถ return ค่ากลับมายังโปรแกรมที่เรียก
3. สามารถในการเปลี่ยนค่า ซึ่งผ่านไปให้ API routine

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ API

Foxpro 2.0 API นั้นแตกต่างจากตัวแปรภาษาชนิดอื่น API จำเป็นต้องใช้ Function ของ Foxpro ในการติดต่อกับ External function ในการสร้าง External function มันจะถูก Compile และ Link เพื่อสร้างเป็น .EXE แต่สำหรับ Foxpro คือ .PLB ซึ่งเป็นชื่อเฉพาะของ Library Foxpro และจะถูก Run โดย Foxpro เมื่อใดก็ตามที่มีการเรียกใช้งาน ซึ่งในการ Run ไฟล์นี้ Foxpro จำเป็นต้องรู้ถึงข้อมูลที่ function นั้นต้องการอีกทั้งรู้ถึง address ที่เริ่มต้นของ function นั้นด้วย

เมื่อ Foxpro ไปค้นหา function เราจำเป็นต้องให้ข้อมูลเหล่านี้

- ชื่อของ function ที่ปรากฏในโปรแกรม .PLB
- ชื่อของ function ที่เรียกใช้
- จำนวนและชนิดของ Parameter ที่ผ่านไปยัง Function

จากข้างต้นเราจำเป็นต้องสร้าง Module พิเศษในไฟล์ .PLB

ตัวอย่าง ของ module

```
#include < "..\pro_ext.h">
extern far myfunc();
FoxInfo FoxPara[] = {
  { "MY_FUNC",      /* Program Call */
    myfunc,        /* C function */
    2,             /* number of Paras */
    ".C,?" }
};
FoxTable _FoxTable =
{
  (FoxTable FAR *)0,
  sizeof (FoxPara)/ sizeof (FoxInfo),
  FoxPara
};
```

-เริ่มต้นด้วยการบอกถึง header file ในภาษาซี ซึ่งจำเป็นสำหรับ Watcom C compiler เพื่อให้เข้าใจถึงข้อมูลในภาษาซี

-การประกาศถึง function ที่เรียกใช้ในที่นี้คือ myfunc() ซึ่งมันจะถูกอ้างถึงตอน Link โปรแกรม หากเราไม่ได้ประกาศไว้ Linker จะละเลยฟังก์ชันพวกนี้ มีผลให้ เราไม่สามารถเรียกใช้ function นี้ใน Foxpro ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-มีข้อมูล 2 ตาราง ในการที่ Foxpro จะเข้าถึง function

FoxInfo Table

ตารางนี้จะบอกรายละเอียดที่จำเป็นเกี่ยวกับ Function ซึ่งประกอบด้วยข้อมูล 4 ตัว

ข้อมูลที่ 1 บอกถึงชื่อที่เรียกใช้ใน file .PRG หรือใน Command Window

ข้อมูลที่ 2 เป็นชื่อของ C function ซึ่ง Foxpro จะเรียกใช้ เมื่อมันพบชื่อของ MY_FUNC()

ข้อมูลที่ 3 บอกถึงจำนวนพารามิเตอร์ที่มากที่สุดในการผ่านจาก Foxpro

ข้อมูลที่ 4 เป็นชนิดของพารามิเตอร์ที่จะผ่านไป

ชนิดของข้อมูลมีดังนี้

C Character

I Number (Integer)

N Number (any other type)

D Date

L Logical

? Free format , any data type

ถ้าเราไม่แน่ใจในชนิดของข้อมูลนั้นเราอาจใช้ "?" แทนชนิดของข้อมูลที่สามารถใช้ได้ ซึ่ง Syntax นั้นจะถูกเช็คโดย โปรแกรมภาษาซีเอง

จากตัวอย่างที่ปรากฏ ชนิดของ Parameter ตัวแรก คือ Character ส่วนตัวที่สองจะเป็นชนิดใดก็ได้ parameter เหล่านี้ ถูกใช้โดย Foxpro เพื่อสร้าง Parameter Block ที่จะผ่านไปให้ Library ซึ่งถือเป็นการเช็ค Syntax ระดับหนึ่ง

Foxtable Table

เป็น table ที่ใช้ภายใน Foxpro เมื่อเราใช้คำสั่ง Set library to Lib_name อาจเป็นไปได้ที่จะมี Library มากกว่าหนึ่ง activate ดังนั้น Foxpro จะมองหาไฟล์ .PLB แล้วจึงมองหาตารางที่ตรงกับข้างบน ชื่อและฟังก์ชันจะถูก load เข้ามาใน Internal symbol table เราสามารถดูชื่อของฟังก์ชันใน Library โดยการคีย์ที่ Command Window

Set library to MY_lib

Disp status

ท้ายสุดของ status listing จะเป็นชื่อของฟังก์ชันใน function library

Buliding Block (function)

การสร้าง function เราจำเป็นต้องทราบ 2 สิ่งคือ

- การอ่านพารามิเตอร์ จาก Foxpro ไปยังฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การ Return ค่ากลับมายัง Foxpro

Reading Parameter

พารามิเตอร์ที่ Foxpro ใช้จะเป็นรูปแบบของ Parameter Block (ParamBlk) ซึ่งถูกสร้างภายใน Foxpro และมี Pointer ที่ไปยังฟังก์ชันโดย Foxpro ส่วนฟังก์ชันในภาษาซี จะถือว่าเป็นหนึ่ง parameter และมี Pointer ที่ไปยัง Parameter Block ดังตัวอย่าง

```
My_func (ParamBlk *parm)
```

```
{
```

```
}
```

ส่วนการ declare ใน header file pro_ext.h จะเป็นดังนี้

```
typedef union {
```

```
    Value val; /* An 'R' in l_type means the locator */
```

```
    Locator loc; /* pcount parameter */
```

```
}
```

```
typedef struct {
```

```
    short int pCount /* Number of Parameter PASSES */
```

```
    Parameter p[1]; /* pCount Parameter */
```

```
}
```

ลักษณะของพารามิเตอร์จะเป็นการรวมกันของ Structure ซึ่งเก็บข้อมูลต่อไปนี้

- จำนวนของพารามิเตอร์ที่ผ่านมา

- ชนิดของข้อมูล

- ค่าของพารามิเตอร์

- บอกว่าเป็นการผ่านโดย Reference หรือโดย value ถ้าผ่านโดย reference เราสามารถ return ค่ากลับมาจาก

พารามิเตอร์ได้

แต่ละพารามิเตอร์ที่ส่งผ่านจะแบ่งเป็น

- Value structure ซึ่งบอกถึงชนิดของข้อมูลและค่าของพารามิเตอร์

- Locator structure บอกถึงสถานะของพารามิเตอร์

The Value Structure

ev_type บอกชนิดของข้อมูล

ev_width ความกว้างของตัวเลขที่แสดง

ev_lenght ความยาวของสตริง หรือ logical

ev_long ค่าของ integer

ev_real ค่าของ number และ date

ev_handle Memory handle สำหรับข้อมูลแบบสตริง

Memory handle เป็นวิธีในการจัดการด้าน memory ของ Foxpro สำหรับสตริงที่ส่งผ่านจะถูกเข้าถึงโดยทาง memory ซึ่งค่าของพารามิเตอร์ที่ถูกพบโดย API function HandToPtr() ซึ่งจะ convert ค่าของ memory เป็น Pointer ที่ไปยัง character ที่จะอ่าน

The Locator Structure

l_type R, ถ้าผ่านแบบ reference

l_where จำนวนของ database หรือ -1 ถ้าเป็นตัวแปรแบบ memory

l_NTI ตำแหน่งของพารามิเตอร์ใน Symbol table ภายใน Foxpro

l_offset ถ้าเป็น field บอกจำนวน field ใน database

l_subs ถ้าเป็นตัวแปรแบบ memory หรือบอกจำนวนของ subscripts

0: memory variable

1: 1 -dimensional array

2: 2 -demensional array

l_sub1: 1st dimension setting

l_sub2: 2nd dimension setting

ส่วนการ declare ใน header file pro_ext.h จะเป็นดังนี้

```
#define MANDLE unsigned short
```

```
typedef structure {
```

```
    char        ev_type ;
```

```
    short       ev_width;
```

```
    unsigned short ev_lenght;
```

```
    long        ev_long;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
double      ev_real;
MHANDLE    ev_handle;
```

```
} Value;
```

การใช้ `l_subs` บอกถึงชนิดของตัวแปร ถ้าเป็น array มันจะถูกเช็คให้มามีค่ามากกว่า 0 ในการที่จะอ่านค่าของ element แรก ให้เช็คค่า `l_Sub1` เป็น 1 แล้วจึงอ่านค่าใน Value structure ถ้าต้องการอ่านค่า element 1,2 ให้เช็ค `l_subs` เป็น 1

Return Value

```
_RetChar()    Return a character string
_RetInt()     Return a integer value
_RetFloat()   Return a double precision number
_RetDateStr() Return a character as a date in the format 'mm/dd/yyyy'
_RetLogical() Return a logical value: 0 for false, for true
_RetVal()     Return a Value structure
```

ในการเขียนโปรแกรมติดต่อกับ PLC นั้นใช้ภาษาซีในการเขียนโปรแกรมโดยติดต่อกันทาง serial port แล้ว return ค่าที่อ่านได้มาให้กับ foxpro ซึ่งในการ return

ค่า นั้นจะใช้ function ของ Watcom C ที่สามารถ return ค่ามาให้ Foxpro ได้

การ Compile และ การ Link Foxpro กับภาษาซี

การ Compile ภาษาซีนั้นจะใช้ Software ของ Watcom C โดยตัวแปลภาษาที่เรียกว่า wcc เมื่อคอมไพล์ไฟล์ ภาษาซีแล้วจะได้ .obj ซึ่งจะนำไฟล์ .obj มา Link โดยใช้ wlink ซึ่งเป็นโปรแกรมในการทำไฟล์ .obj ที่ได้จากการคอมไพล์ภาษาซีให้เป็น Library ที่ Foxpro สามารถเรียกใช้งานได้โดย library ที่ Foxpro มองเห็นจะเก็บในรูปแบบของไฟล์ .plib ในแต่ละ Library อาจมีได้หลายฟังก์ชัน และการเรียกใช้

ฟังก์ชันในภาษาซี Foxpro จะเรียกใช้ได้ด้วยคอมมานด์ set library to [path]

(path คือ directory ที่ฟังก์ชันในภาษาซีปรากฏอยู่)

ส่วนการทำให้โปรแกรมที่เขียนด้วย Foxpro เป็น execute file นั้นจะต้องมีการสร้าง Project file ก่อน (.PJX) จาก นั้นจึงสร้าง Application

file (.APP) แล้วนำมารวมเป็น execute file (.exe)

การ Compile ด้วย Watcom C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WATCOM C option

Command line syntax ของ Watcom C แสดงดังต่อไปนี้

WCC [options] [d:] [path] filename[.ext] [option]

WCC -เป็น version ของ WATCOM C ซึ่ง run บน dos หรือ os/2

d: -เป็น option ของไดรฟ์ที่อ้างถึงเช่น A:,B: ถ้าไม่บ่งบอกจะเป็น default ไดรฟ์

path -เป็น option ของ directory ถ้าไม่บ่งบอกจะเป็น directory ปัจจุบัน

filename -เป็นชื่อของ file ที่ถูกคอมไพล์

ext -เป็นชื่อ extension ของไฟล์ที่ถูกคอมไพล์ ถ้าไม่ใช่ .c จะเป็น default

option -เป็น option ต่าง ๆ ของ watcom c ซึ่งต้องนำหน้าด้วย "/"

Option ต่าง ๆ ในการคอมไพล์ภาษาซีด้วย Watcom C ซึ่ง option เหล่านี้จะใส่หลังคำสั่ง wcc

Compiler options Description

/dl	include line number debugging information
/e<number>	set error limit number (default /e20)
/fi=<file_name>	force file_name to be include
/fo=<file_name>	set preprocessor output filename
/fpc	generate calls to floating-point library
/fpi	generate inline 80x87 instructions with emulation (default)
/fpi87	generate inline 80x87 instructions (equivalent to "7" option below)
/m{s,m,c,l,h}memory	model (ms=small, mm=medium, mc=compact, ml=large, mh=huge) (default/ms)
/s	remove stack overflow checks
/0	8088 and 8086 instructions
/1	80188 and 80186 instructions
/2	80286 instructions

รายละเอียดต่าง ๆ เกี่ยวกับ option สามารถดูได้จากหนังสือ reference

(Watcom c optimize compiler and tools user'guide)

PLC (Programmable Logic Control)

ระบบซีควেনซ์ (Sequence Control) แต่เดิมจะประกอบด้วยอุปกรณ์ไฟฟ้าเชิงกล ซึ่งได้แก่ Relay Timer Counter และอื่น ๆ ลักษณะการทำงานของระบบควบคุมนี้ จะมีอยู่ด้วยกันสองสถานะคือ เปิดกับปิด ระบบนี้มีขนาดใหญ่และมี ปัญหาในการทำงานหลายอย่าง

เนื่องจากเทคโนโลยีทางด้านไมโครโปรเซสเซอร์ (Microprocessor) ในปัจจุบันได้เจริญก้าวหน้าไปมาก โดยเฉพาะในด้านของการควบคุมแบบต่อเนื่อง ทำให้มีการค้นคว้าทดลองเครื่องควบคุมชนิดโปรแกรมได้ (Programmable Logic Control) หรือเรียกสั้นๆว่า PLC

ข้อดีของ PLC

1. สิ้นเปลืองเนื้อที่น้อยเพราะมีขนาดเล็ก
2. สามารถใช้ควบคุมเครื่องจักรหรือระบบกระบวนการทำงานใด ๆ ก็ได้
3. การเปลี่ยนลำดับหรือขั้นตอนการทำงานทำได้ง่าย
4. การติดตั้งทำได้สะดวกและง่าย
5. ขยายระบบให้ใหญ่ขึ้นทำได้ง่าย
6. ราคาถูกกว่าระบบ Relay
7. ความน่าเชื่อถือ (Reliability) ดีเพราะเป็นอุปกรณ์สารกึ่งตัวนำไม่มีการเดินสายมาก
8. มีระบบตรวจสอบความผิดพลาดด้วยตัวเอง
9. บำรุงรักษาง่าย
10. เวลาในการทำงานเร็วกว่าระบบ Relay
11. ต่อเข้ากับระบบคอมพิวเตอร์เพื่อให้ทำงานได้หลายแบบ

อุปกรณ์สมบัติต่างมีดังนี้

1. I/O UNIT

INPUT, OUTPUT MODULE

OPTICAL TRANSMITTING I/O UNIT

2. SPECIAL I/O UNITS

ANALOG INPUT/OUTPUT UNITS

PID UNIT

TEMPERATURE SENSOR UNIT

HIGH-SPEED COUNTER UNIT

POSITION CONTRIL UNIT

ASCII UNIT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LADDER PROGRAM I/O UNIT

FILE MEMORY UNIT

MCR UNIT (MAGNETIC CARD READER)

3. LINK SYSTEM

REMOTE O/I SYSTEM

I/O LINKS

PC. LINK SYSTEMS

HOST LINK SYSTEMS

SYSMAC NET LINK SYSTEMS

4. PERIPHERAL TOOLS

PROGRAMMING CONSOLE

EPROM WRITER

PRINTER INTERFACE UNIT

SYSMAC C-SERIES SUPPORT TOOLS

FIT : Factory Intelligent Terminal

LSS : Ladder Support Software

GPC : Graphics Programming Console

โครงสร้างโดยทั่วไปของ PLC

ส่วนประกอบที่สำคัญของ PLC แบ่งออกเป็น 4 ส่วน

1. หน่วยประมวลผล (CPU Unit)
2. หน่วยความจำ (Memory Unit)
3. หน่วยอินพุต-เอาต์พุต (Input-Output Unit)
3. อุปกรณ์ต่อร่วม (Peripheral Device)

1. หน่วยประมวลผล (CPU Unit)

ทำหน้าที่ควบคุมการทำงานของระบบทั้งหมด โดยรับ Input เข้ามาทำการประมวลผลแล้วส่งผลที่ได้ออกไปจากนั้น ก็วนกลับไปรับข้อมูล Input เข้ามาอีก แล้วทำซ้ำๆ ในลักษณะนี้อีกไปเรื่อยๆ การทำงานของ CPU จะอยู่ภายใต้การควบคุม ของโปรแกรมที่ผู้ใช้ป้อนเข้าไป โดยที่การทำงานในแต่ละรอบนี้เรียกว่า การสแกน (SCANNING) สำหรับเวลาของการสแกน ขึ้นอยู่กับขนาดของหน่วยความจำและความเร็วของหน่วยประมวลผล ช่วงเวลาของการสแกนจะทำให้ทราบถึงความสามารถ ของการตอบสนองต่อการเปลี่ยนแปลงของ Input-Output ว่ามีความเร็วเพียงใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หน่วยความจำ (Memory Unit)

เป็นองค์ประกอบสำคัญของระบบเพราะใช้เป็นที่เก็บโปรแกรมและข้อมูล ขนาดของหน่วยความจำจะเป็นสิ่งที่กำหนดความสามารถของระบบ หน่วยความจำของ PLC แบ่งเป็น

1. RAM (Random Access Memory)
2. EPROM (Erasable Programmable Read Only Memory)
3. หน่วยอินพุต-เอาต์พุต

หน่วยอินพุต

หน่วยอินพุตทำหน้าที่สัญญาณจากอุปกรณ์ภายนอก แล้วแปลงสัญญาณชนิดขาเข้าให้เป็นสัญญาณที่เหมาะสมเพื่อส่งเข้าไปให้หน่วยประมวลผลกลาง

ปกติหน้าที่ของ Input มีดังนี้

- เปลี่ยนแปลงระดับของสัญญาณเข้าให้เป็นระดับสัญญาณที่เหมาะสม
- การส่งสัญญาณระหว่างหน่วยอินพุตกับหน่วยประมวลผล

หน่วยเอาต์พุต

ทำหน้าที่รับค่าสถานะที่ได้จากการประมวลผล เพื่อนำค่าสถานะเหล่านั้นไปควบคุมอุปกรณ์ภายนอก

4. อุปกรณ์ต่อร่วม (Peripheral Device) อุปกรณ์ที่ใช้ในการต่อร่วมกับ PLC มีหลายชนิดเพื่ออำนวยความสะดวกในการพัฒนาโปรแกรม

รายละเอียดเกี่ยวกับการติดต่อกับ PLC มีดังนี้

การติดต่อกับ PLC เพื่อนำข้อมูลจากเครื่องจักรเข้ามาเก็บใน database และคำนวณในสูตร

ดังนั้นข้อมูลที่ได้รับมาจาก PLC มีดังนี้

- ค่าของ machine number (หมายเลขเครื่องจักร) เนื่องจากโปรแกรมนี้อาศัยกับเครื่องจักรหลายตัว

ดังนั้นต้องมีการ อ่านค่าหมายเลขเครื่องจักร เพื่อเก็บข้อมูลลงใน database ให้ถูกต้อง

- ข้อมูลเกี่ยวกับสถานะของเครื่องจักร ได้แก่ speed , output, temperature

- การ reset plc ที่ช่องของ output เพราะเนื่องจากแต่ละครั้งที่เครื่องหยุดเดิน ต้องเริ่มอ่านข้อมูลเกี่ยวกับ output ใหม่ แต่ข้อมูลเดิมของ output ยังคงค้างอยู่ดังนั้นจึงต้องมีการ reset ค่านั้นให้เป็น 0

การติดต่อกับ PLC จะมีการส่ง Command ไปยัง PLC จากนั้นจะคำนวณ FCS (Frame check sequence) แล้วจึงส่ง Command ที่คำนวณค่า FCS แล้วไปให้ PLC ดังจะเห็นจาก ฟังก์ชัน CalcFCS() ในโปรแกรมภาษาซี

การอ่านค่า machine number นั้นค่าของ machine number จะถูกเช็คที่ช่อง 1999 และคำสั่งที่ส่งไปคือ

@00RD19990001 หมายถึงให้อ่านข้อมูลที่ช่อง 1999 หนึ่งช่อง

การอ่านค่าของสถานะเครื่องจักร จะกระทำเช่นเดียวกับการอ่านค่า machine number คือส่ง command และมีค่าจำนวน FCS คำสั่งที่ส่งไปคือ

@00RD00010010 หมายถึงให้อ่านข้อมูลที่ช่อง 1 จนถึงช่อง 10 เพราะข้อมูลที่ต้องการคือ speed, output และ temperature อีก 8 ช่อง

การทำงานในช่วง realtime

ลักษณะของโปรแกรมรูปแบบ realtime นั้นจุดประสงค์เพื่อต้องการทราบข้อมูล ณ. ระยะเวลาหนึ่ง ซึ่งข้อมูลที่ต้องการทราบคือ output และ speed ของเครื่องจักร สิ่งที่สำคัญที่สุดของระบบนี้คือเวลาในการทำงานของโปรแกรม เนื่องจากค่าที่อ่านได้มาจากสถานะของเครื่องจักร จะถูกนำมาประมวลผลเพื่อให้ได้ค่า efficiency ต่างๆ ซึ่งการคำนวณของ สูตรนั้นจะมีเวลาที่เกี่ยวข้องเสมอ เพราะค่าที่ได้จากการคำนวณจะต้องเป็น realtime ด้วย เวลาที่ใช้ในการคำนวณสูตรจะมี 2 ค่า คือ ช่วงเวลาที่เริ่มตั้งแต่เปิดเครื่องจนกระทั่งจบโปรแกรม เรียกว่า plan time คือเวลาที่ใช้ในการดำเนินการของโปรแกรมและ down time คือช่วงเวลาที่เครื่องหยุดเดิน การทำงานของโปรแกรมในระบบ realtime นี้จะใช้ algorithm แบบ Polling คือเป็นการทำงานแบบวน loop การทำงานของโปรแกรมนี้อาจจะใช้สำหรับสถานะของเครื่องจักร ซึ่งผู้ที่ดูข้อมูลจะอยู่ใน office ส่วนข้อมูลที่อ่านจากเครื่องจักรจะอ่านในสายการผลิต (Line) ดังนั้น โปรแกรมจึงแบ่งเป็น 2 ส่วนคือ

- ส่วนที่อยู่ในสายการผลิต (Line)
- ส่วนที่อยู่ใน office

เริ่มต้นของโปรแกรมจะเช็ก่อนว่าเครื่อง computer ที่ใช้ขณะนั้น (Work station เนื่องจากใช้ระบบ LAN) อยู่ใน office หรืออยู่ใน Line ถ้าเครื่องอยู่ใน Line function ที่จะ call ไป จะทำหน้าที่ทั้งรับข้อมูลจาก PLC ทำการคำนวณ และ พร้อมกับเก็บข้อมูลลงบนฐานข้อมูล database ที่ได้เตรียมไว้และเขียนข้อมูลลงบน temp file เพื่อให้ Computer ที่อยู่ใน office สามารถทราบถึงสถานะขณะนั้นได้

ส่วน Computer ที่อยู่ใน office จะทราบสถานะของเครื่องจักรในสายการผลิตได้ จากการอ่านข้อมูล temp file ที่เก็บ อยู่บน file server พร้อมทั้งอ่านข้อมูลจาก database file เพื่อให้ทราบสถานะ ในช่วงเวลา ที่ผ่านมาก็คืออะไร ขึ้นบ้าง

ในส่วนการเกิด downtime นั้นจะมีตัวแปรตัวหนึ่งคอยเก็บค่าช่วงเวลาที่เกิด downtime ซึ่งเมื่อเหตุการณ์นั้นสิ้นสุดลง operator (ผู้ป้อนข้อมูล) จะป้อนสาเหตุของ downtime เพื่อจะได้รู้ว่า downtime ที่เกิดขึ้นนั้นเป็น downtime ประเภทอะไร เมื่อทราบประเภทของ downtime แล้ว ค่าของตัวแปรนั้นจะถูกเก็บไว้เป็นค่าของช่วงเวลาที่เกิด downtime ประเภทนั้นๆ

การนำเสนอข้อมูล

Dialy Report In The Part

เป็นการนำเสนอข้อมูลในอดีต เกี่ยวกับประสิทธิภาพของเครื่องจักร พนักงาน จำนวนผลิตภัณฑ์ ความเร็วเฉลี่ยของ เครื่องจักร ในช่วงเวลาที่ผู้ใช้กำหนด ข้อมูลที่สามารถดูได้มีดังนี้

- Machine Efficiency
- Maintenance Downtime
- Production Downtime
- Machine Availibity
- Labour Efficiency
- จำนวน และรายชื่อของพนักงาน
- ช่วงเวลาที่เครื่องจักรทำงาน และสาเหตุที่เครื่องจักรหยุดทำงาน

Monthly Report

เป็นการนำเสนอข้อมูล ในช่วงเวลาที่กำหนด การนำเสนอข้อมูลในเมนูนี้ เป็นการเปรียบเทียบผลิตภัณฑ์ชนิดเดียวกันในแต่ละเครื่องจักร ข้อมูลที่แสดงมีดังนี้

- Speed
- Output
- Machine Efficiency
- Labour Efficiency
- Machine Availibity
- Production Downtime
- Number of Crews

Dialy Total Report

เป็นการนำเสนอข้อมูลต่าง ๆ ในขณะปัจจุบันเปรียบเทียบกันทุกเครื่องจักร การนำเสนอข้อมูลแบบนี้ สามารถเปรียบเทียบการทำงานของเครื่องจักรในขณะนี้ได้เลย

ข้อมูลที่น่าเสนอข้อมูลมีดังนี้

- Speed
- Output
- Machine Efficiency

- Production downtime
- Machine Availability
- labour efficiency

Quality Control

เป็นการตรวจสอบคุณภาพของผลิตภัณฑ์

ว่าได้มาตรฐานตามที่กำหนดหรือไม่ถ้าตรวจสอบแล้วต่ำกว่ามาตรฐาน ก็อาจจะต้องมีการยกเลิกที่ผลิตในรายการนั้น

วิธีการตรวจสอบ จะใช้วิธีสุ่มผลิตภัณฑ์ที่ผลิตได้ในขณะนั้นนำมาชั่งหาน้ำหนักของผลิตภัณฑ์ (Pulp Weight) น้ำหนักรวม (Total Weight) วัดตำแหน่งของกาว (Position Glue) วัดความกว้างของผลิตภัณฑ์ (Pulp Length) วัดความกว้าง ความหนา (Thickness) และวัดความยาวของแถบกาว (Tab Length) ในการสุ่ม ในแต่ละรายการนั้น จะสุ่มหลายๆ ครั้ง เพื่อให้ได้ผลิตภัณฑ์ที่ได้มาตรฐานอย่างสม่ำเสมอ

ข้อมูลที่ได้นั้นจะเก็บไว้ในฐานข้อมูล และจะนำมาคำนวณตามหลักสถิติ และรายงานผลต่อไป

ค่าทางสถิติที่ใช้ในการรายงานมีดังนี้

- Average เป็นการหาค่าเฉลี่ยของข้อมูล หาได้จาก ผลรวมของข้อมูล/จำนวนข้อมูล
- Standard Division
- Maximum
- Minimum
- Over
- Under
- Upper Limit
- Lower Limit
- Target
- Variance
- Test
- SPC (CP)
- SPC (CCP)

การรายงานผลมีการรายงานผลในรูปของตารางและกราฟ

- แบบตาราง แบ่งเป็น

1. Summary Report เป็นการนำเสนอรายงานที่ เปรียบเทียบค่าทาง สถิติและค่ามาตรฐาน ที่กล่าวข้างต้น ของแต่ละ เครื่องจักร ซึ่งเป็นค่าของวันที่ๆ ผู้ใช้ต้องการและชนิดของผลิตภัณฑ์ที่ต้องการจะเปรียบเทียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำเสนอข้อมูลแบบนี้ จะเห็นภาพโดยรวมของเครื่องจักรทุกเครื่อง และสามารถเห็นได้ทันทีว่าผลิตภัณฑ์ได้ มาตรฐานเพียงใด สมควรจะยกเลิกหรือไม่ และสามารถเปรียบเทียบการทำงานของแต่ละเครื่องจักรได้

2. Parameter in the part เป็นการนำเสนอรายงานโดยไม่เจาะจงผลิตภัณฑ์ แต่จะนำเสนอข้อมูลในช่วงวันที่ที่ผู้ใช้ ต้องการ เพื่อคุณลักษณะย้อนหลังของแต่ละเครื่องจักรเปรียบเทียบกัน

- การนำเสนอแบบกราฟ แบ่งเป็น

1. กราฟแท่ง เป็นการแสดงเปรียบเทียบค่าทางสถิติ เปรียบเทียบกันในแต่ละเครื่องจักร เพื่อแสดงให้เห็นความแตกต่างได้ง่ายขึ้น

2. กราฟเส้น เป็นกราฟที่แสดงให้เห็นว่าข้อมูลมีค่าใกล้เคียงหรือห่างจากค่ามาตรฐานมากน้อยเพียงใด ในแต่ละเครื่องจักร



DATABASE

TIDPROD.DBF

P_ID	P_NAME
------	--------

TPRODUCT.BDF

P_ID	DATE	SHIFT_NO	MC_EFF	MC_BREAK	MC_AVAIL
LABOUR_EFF	P_NO				

TIDCREW.DBF

CREW_ID	SHIFT_NO	MACHINE_NO	CREW_NAME
---------	----------	------------	-----------

TCREWS.DBF

P_ID	DATE	SHIFT_NO	CREW_NAME
------	------	----------	-----------

TIDDOWN.DBF

D_ID	D_NAME
------	--------

TLIST.DBF

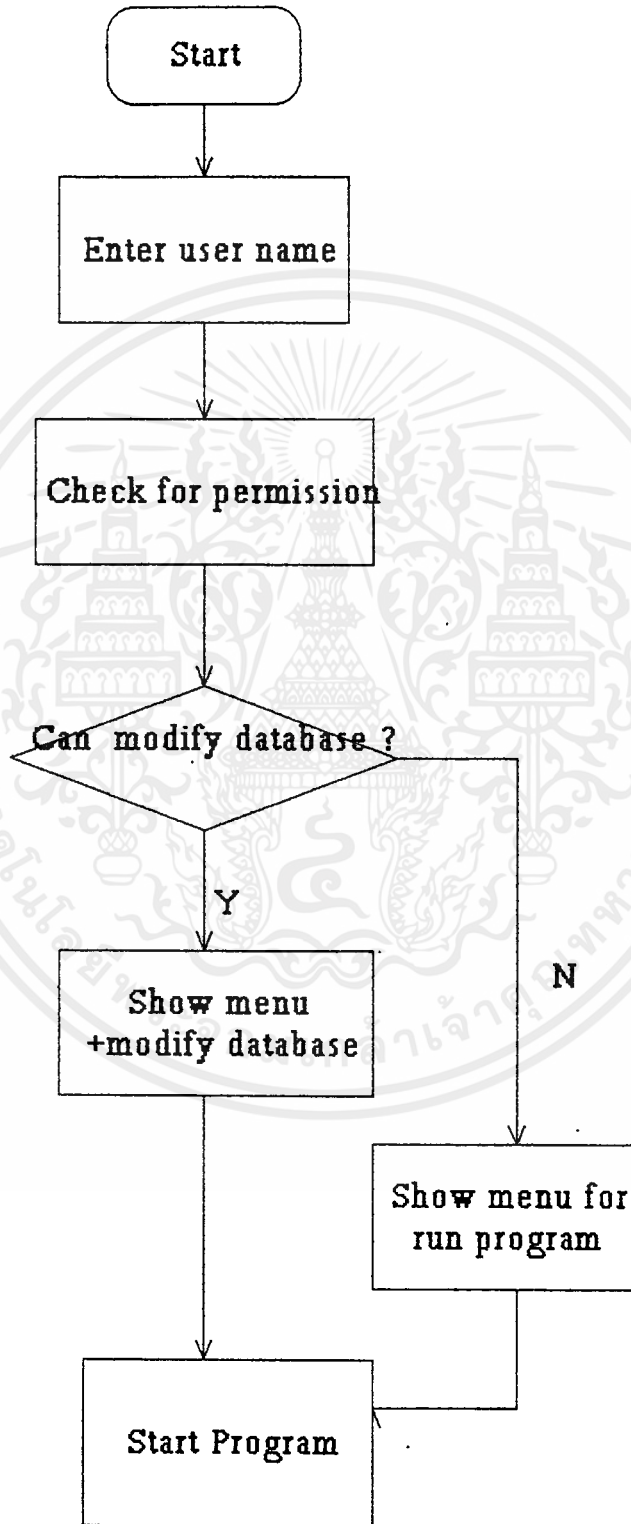
P_NO	TIME1	TIME2	ACTIVITY	SPEED	OUTPUT
------	-------	-------	----------	-------	--------

P_WEIGHT.DBF ,T_WEIGHT.DBF ,T_LENGTH .DBF ,P_GULE.DBF ,THICK.DBF,P_LENGTH.DBF

P_NO	DATE	SHIFT_NO	TIME	P_ID	VALUE
------	------	----------	------	------	-------

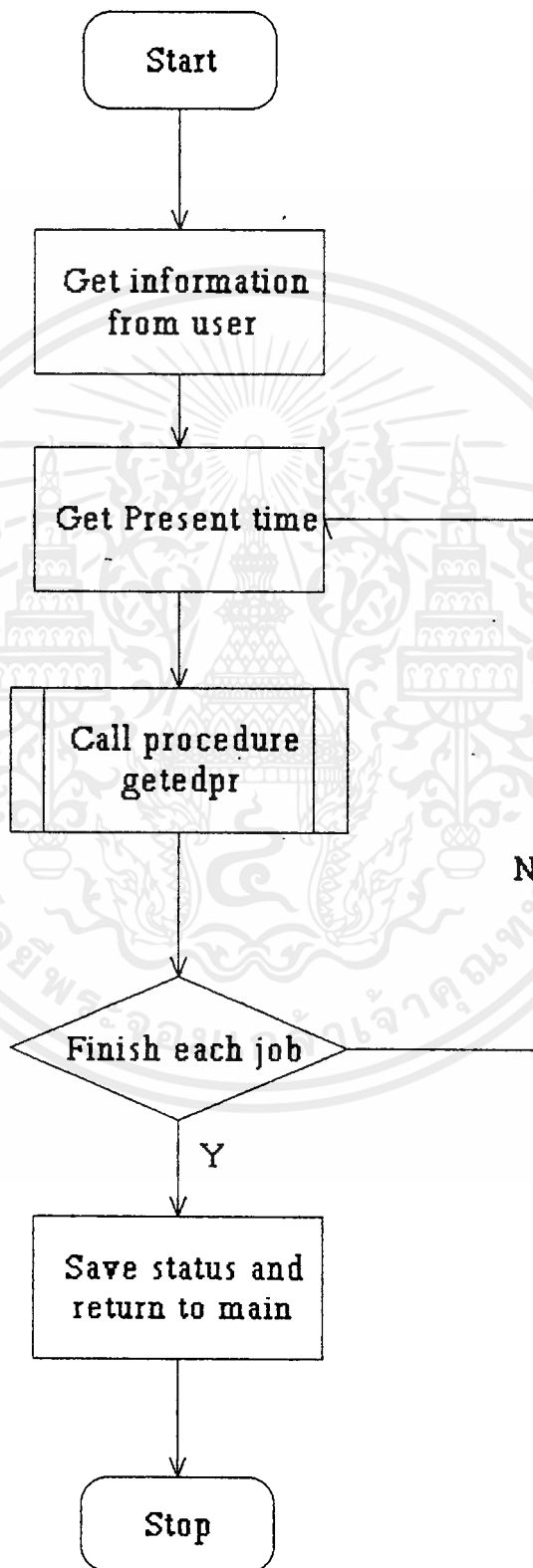
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Main Program



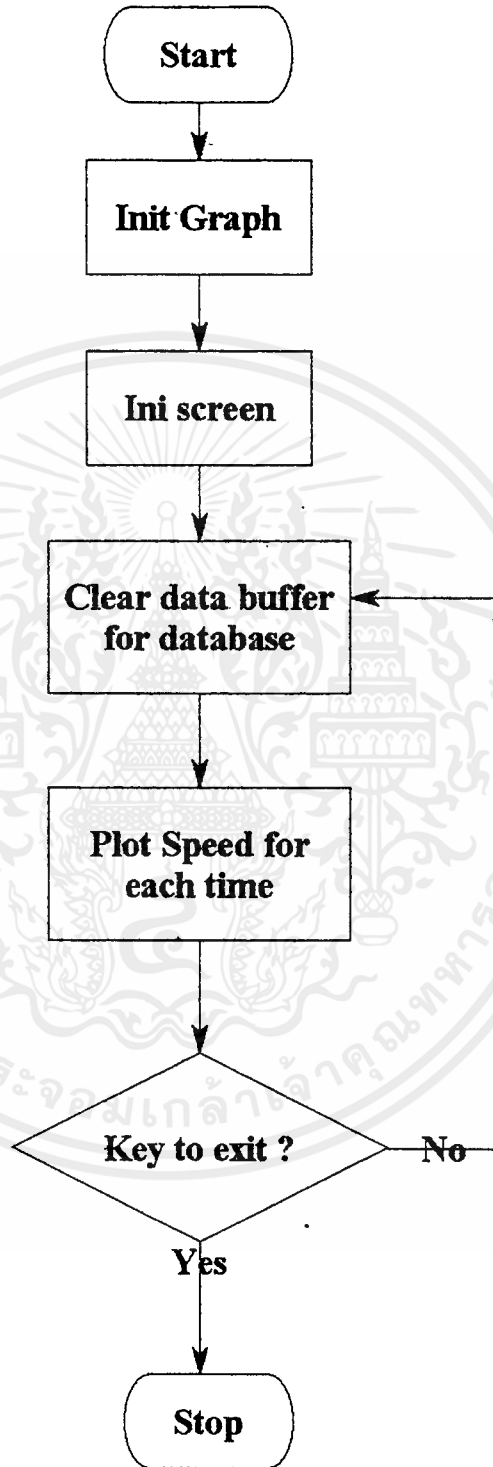
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Procedure each machine



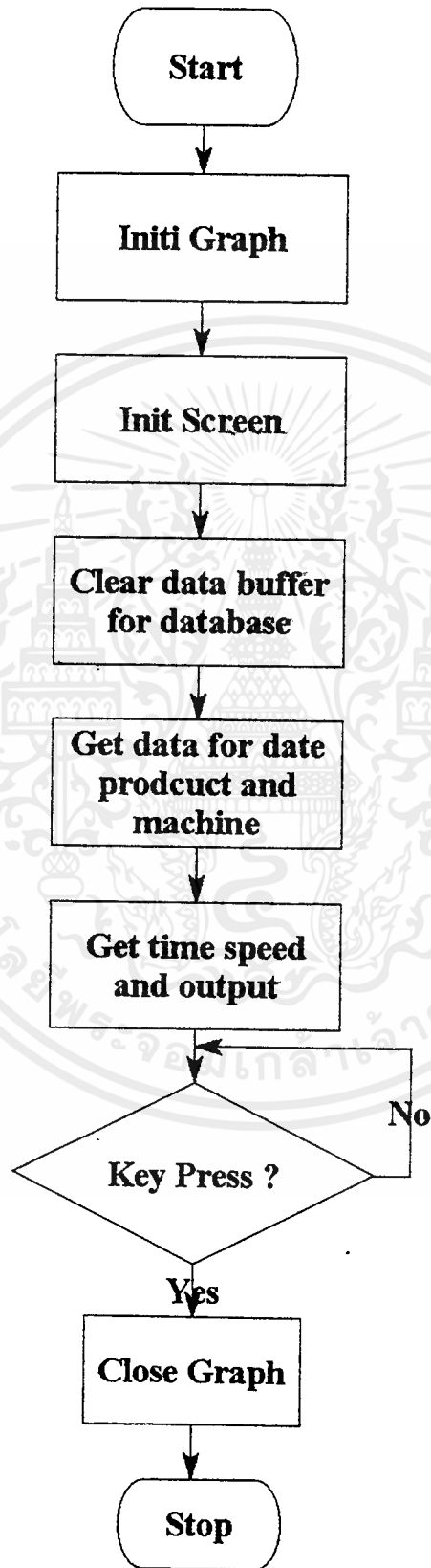
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Algorithm for real time graph



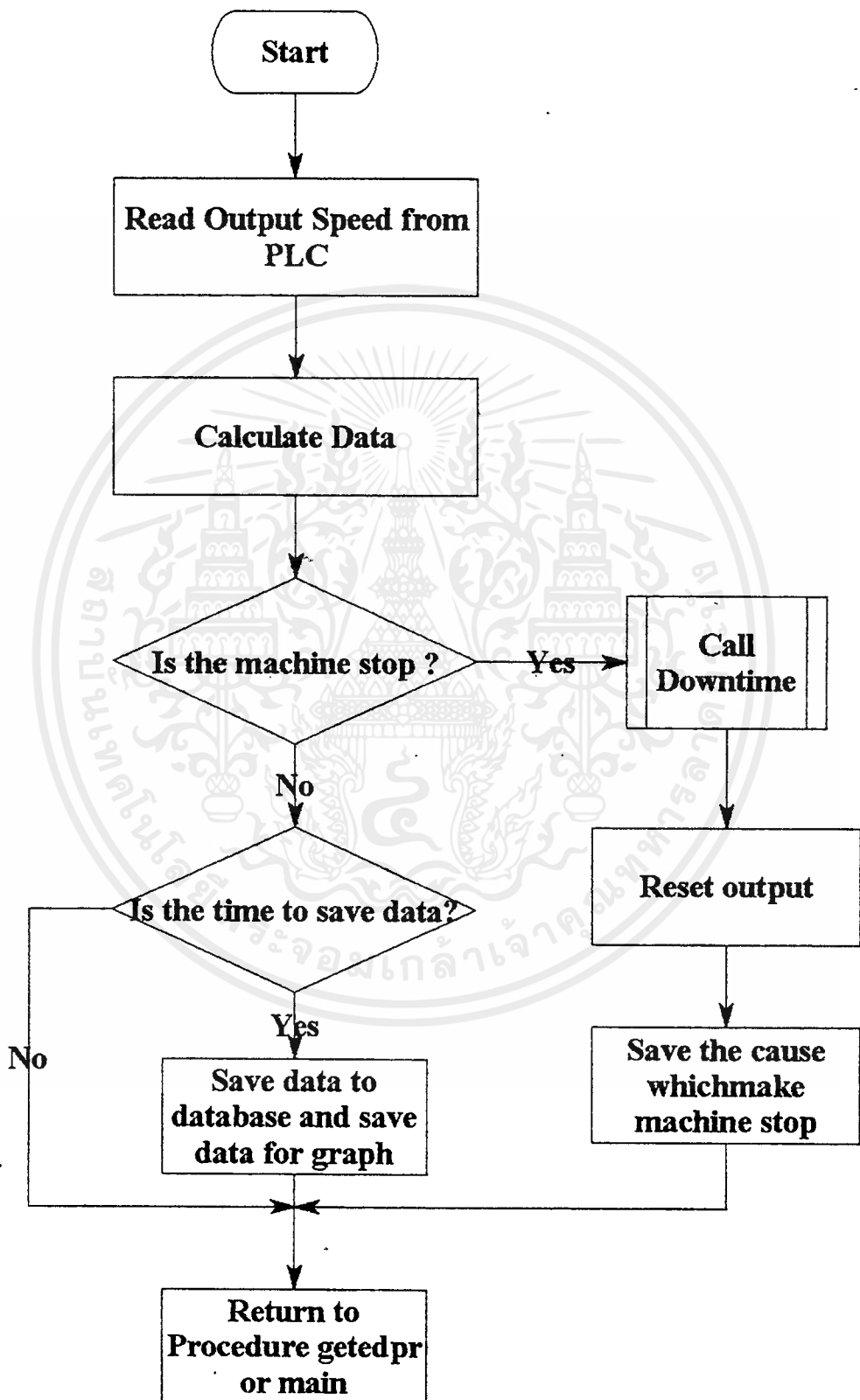
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Graph In the part

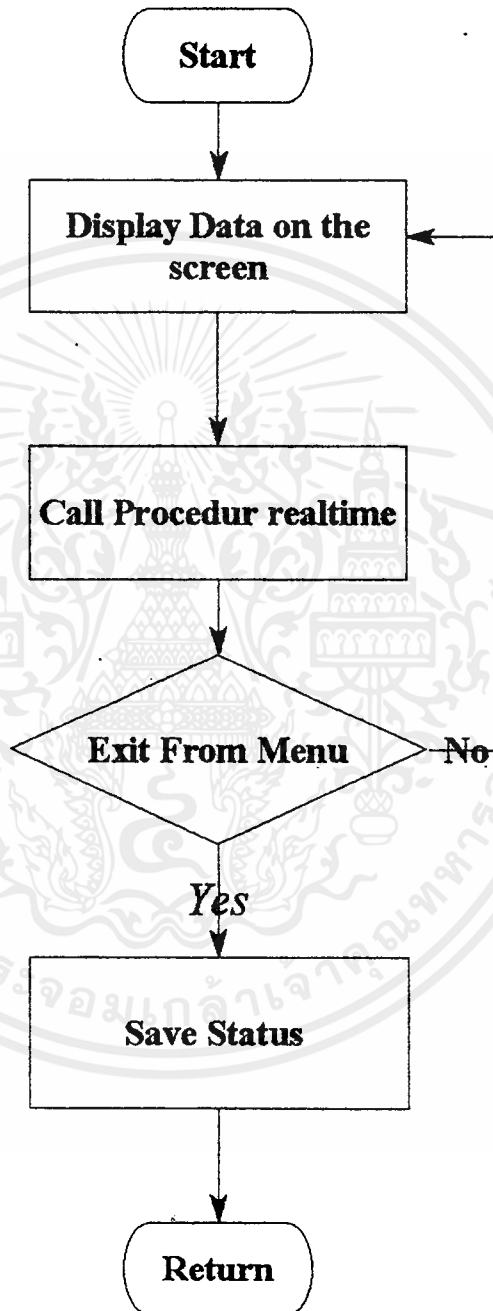


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Procedure Real Time

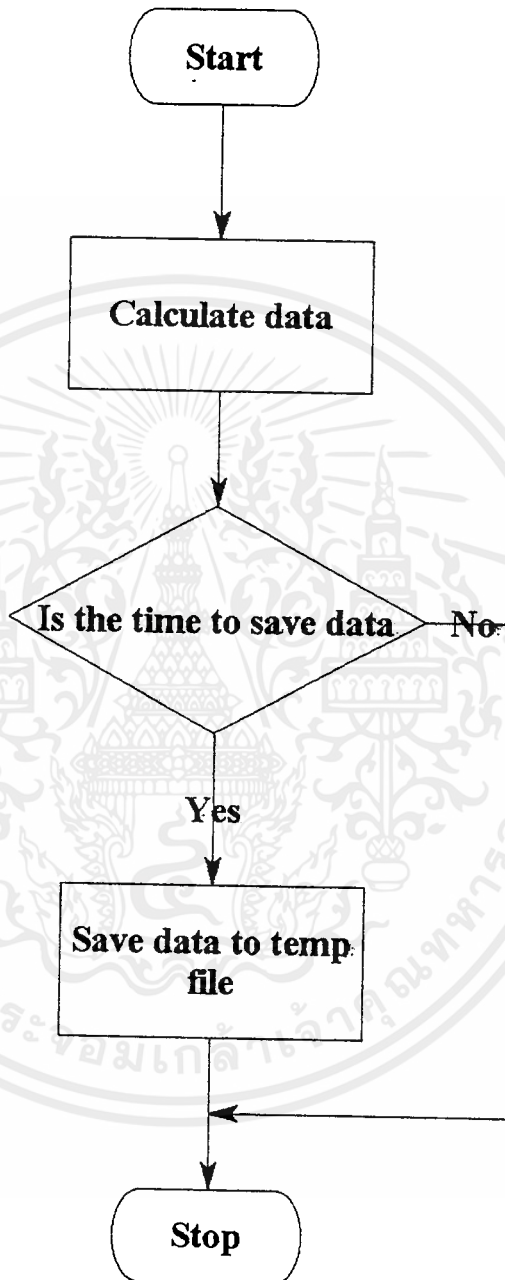


Procedure Getedpr



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Procedure Real - down



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Foxpro 2.0 Library Construction Kit WATCOM C Optimizing Compiler and Tools Fox Software ,Inc 1991

Foxpro 2.0 Library Construction Kit WATCOM C Library Reference Fox Software ,Inc 1991

Foxpro 2.0 Command & Function ,Inc 1991

สุพรรณ กุลพานิชย์ : PROGRAMMABLE CONTROLLER เทคนิคและการใช้งานเบื้องต้น บริษัทอมร่อน
ตรีศกดิ์ จำกัด พิมพ์ครั้งที่ 1 พ.ศ. 2533



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้