บทที่ 3

การออกแบบและพัฒนาโปรแกรม

การออกแบบและพัฒนาโปรแกรมโดยใช้ Microsoft Visual Basic

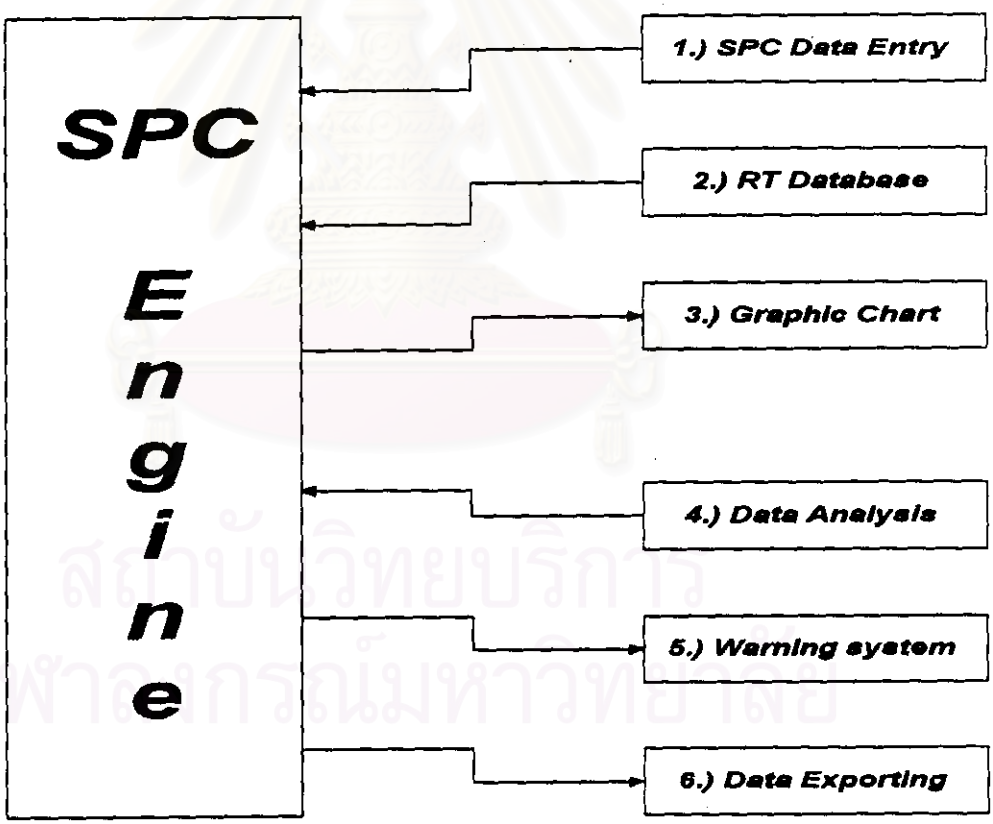
การออกแบบ และ พัฒนาโปรแกรม สำหรับระบบควบคุมกระบวนการเชิงสถิติแบบตามเวลาจริง นั้นเป็นการออกแบบซอฟท์แวร์ที่ทำงานภายใต้ Microsoft Windows 95 เพื่อตอบสนองการใช้งานคอมพิวเตอร์อย่างกว้างขวาง ซึ่งปัจจุบันอาจกล่าวได้ว่า ระบบปฏิบัติการภายใต้ Windows ได้เข้ามาแทนที่ระบบปฏิบัติการเดิมที่ทำงานภายใต้ DOS เกือบจะทั้งหมด ดังนั้น การพัฒนาโปรแกรมคอมพิวเตอร์ในปัจจุบัน จึงมักนิยมทำภายใต้ Windows สำหรับงานวิจัยนี้ ได้พิจารณาเลือก Microsoft Visual Basic version 4.0 Professional edition เป็นเครื่องมือสำคัญ ในการออกแบบ และ พัฒนาโปรแกรมสำหรับระบบควบคุมกระบวนการเชิงสถิติแบบตามเวลาจริง ทั้งนี้เนื่องจากว่า Microsoft Visual Basic เป็นภาษาคอมพิวเตอร์ที่ใช้สำหรับสร้าง Application ที่ทำงานภายใต้ Windows ที่ดีภาษาหนึ่ง และยังสนับสนุนการเขียนโปรแกรม แบบ Event-Driven ดังที่ได้กล่าวไว้แล้วในบทที่ 2 ทำให้สามารถออกแบบ User interface ได้อย่างมีประสิทธิภาพ สามารถรันโปรแกรมย่อยของแต่ละ Object ได้อย่างอิสระต่อกัน จึงช่วยให้การพัฒนาโปรแกรม มีความยืดหยุ่นเป็นอิสระตามเหตุการณ์ที่ตอบสนองของแต่ละ Object กล่าวคือ โปรแกรมจะถูกดำเนินการ ก็ต่อเมื่อ มีเหตุการณ์ที่กำหนดเกิดขึ้นบน User Interface นั้นๆ

การพัฒนาโปรแกรมโดยอาศัย Microsoft Visual Basic มีหลักการที่สำคัญ 2 ส่วน ได้แก่ การออกแบบ รูปแบบของส่วนที่ติดต่อกับผู้ใช้งาน (User Interface Design) และในส่วนของการเขียนคำสั่งการทำงานของโปรแกรม (Command Code) ตัวอย่างเช่น การสร้างปุ่มคำสั่งต่างๆ ดัง เช่น ปุ่ม OKหรือ ปุ่ม Cancel ภายใน Object ของปุ่ม OK จะมีคำสั่งให้โปรแกรมทำงานตามที่ต้องการ ส่วนปุ่ม Cancel ภายใน Object นี้ จะมีคำสั่งเพื่อยกเลิกการทำงานของโปรแกรม เป็นต้น การพัฒนาโปรแกรมโดย Microsoft Visual Basic นั้นจะขาดหลักการที่สำคัญทั้ง 2 อย่างมิได้ ดังนั้น โปรแกรมจะออกมา ในลักษณะใดนั้น ขึ้นอยู่กับการสร้างUser Interface ตามที่ ต้องการ และการเขียนโค็ดคำสั่งที่สามารถตอบสนองตามเหตุการณ์ที่สอดคล้องกันได้อย่างเหมาะสม

โครงสร้างของโปรแกรม

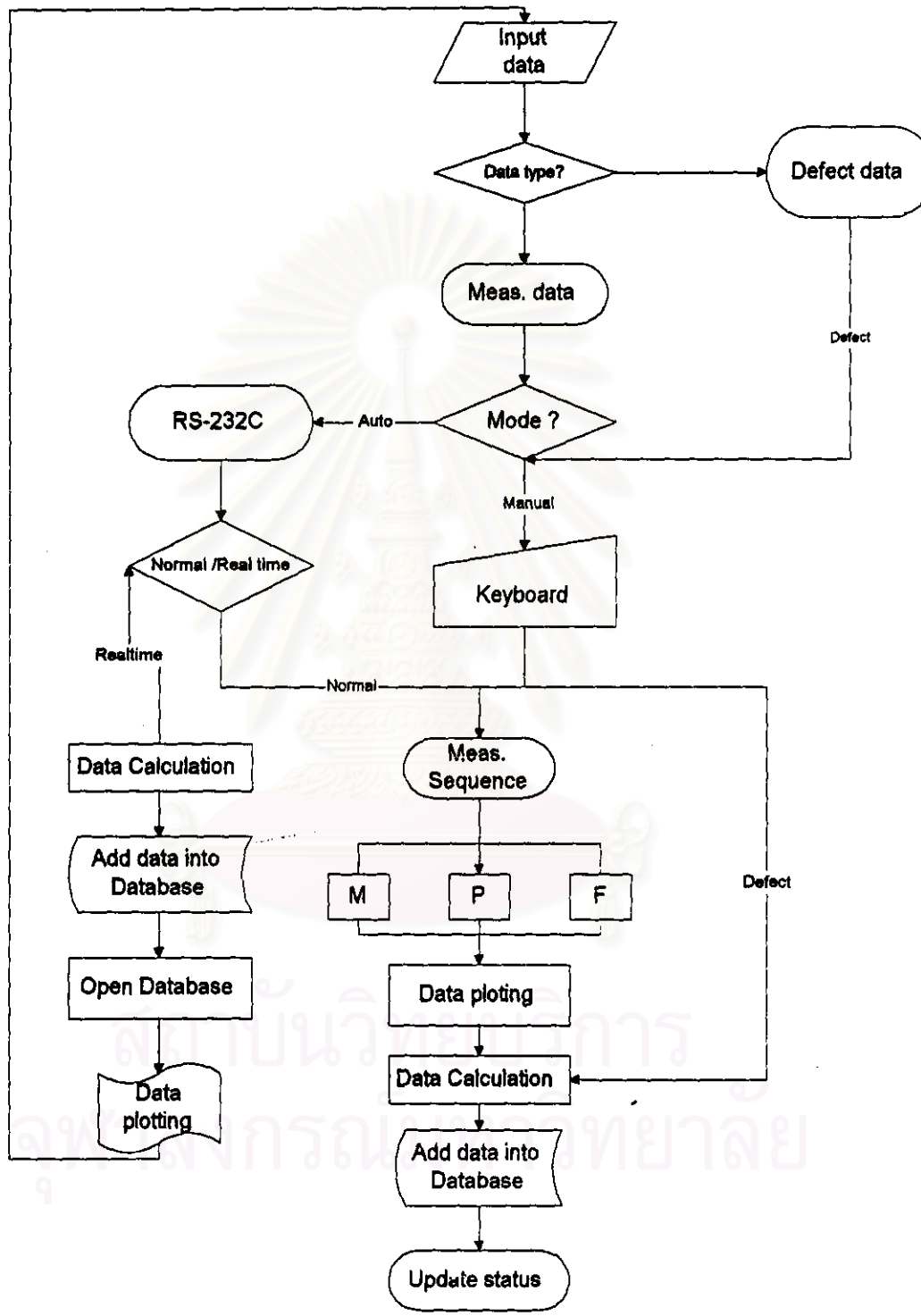
งานวิจัยนี้ได้ออกแบบและพัฒนาโปรแกรมโดยใช้ชื่อว่า SPC_Engine ซึ่ง มีองค์ประกอบหลักๆ ของโปรแกรม ดังต่อไปนี้ (ดูรูปที่ 3.1)

1. ระบบป้อนข้อมูลเข้า (SPC Data Entry)
2. ระบบฐานข้อมูลแบบเชิงเวลาจริง (Real Time Database)
3. ระบบแสดงผลในรูปแบบกราฟฟิคของแผนภูมิต่างๆ (Graphic Charts)
4. ระบบวิเคราะห์ข้อมูล (Data Analysis)
5. ระบบการเตือน (Warning)
6. ระบบการส่งข้อมูลออกสู่ภายนอก (Data Exporting)



รูปที่ 3.1 แสดงโครงสร้างหลักของโปรแกรม SPC_Engine

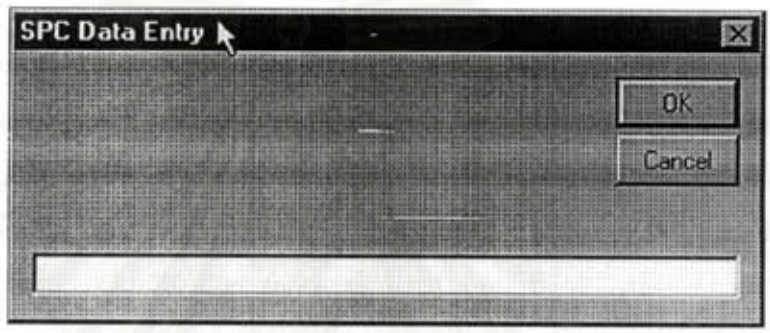
1.ระบบป้อนข้อมูลเข้า (SPC Data Entry)



รูปที่ 3.2 แสดง Flow chart ของ SPC Data Entry

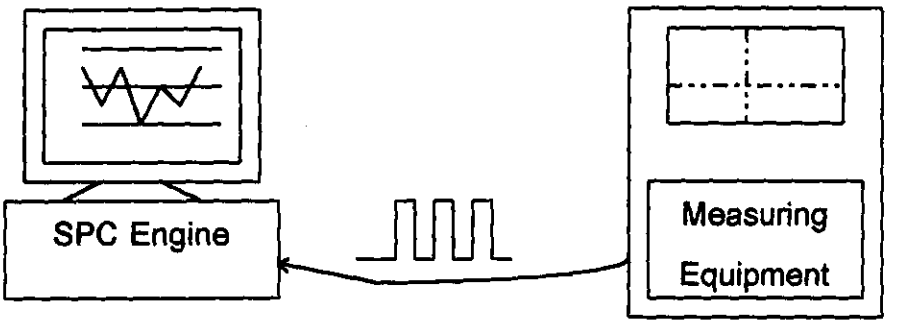
การป้อนข้อมูลเข้าสู่ระบบควบคุมกระบวนการเชิงสถิติแบบตามเวลาจริง ในที่นี้ได้มีการออกแบบ ให้ระบบสามารถรับข้อมูลเข้าจากภายนอกได้ 2 วิธี ดังนี้

1.) การรับข้อมูลเข้าโดยการป้อนผ่านคีย์บอร์ด (Manual Data Entry) วิธีการนี้ใช้ สำหรับข้อมูลชนิดต่อเนื่อง (Continuous value) และข้อมูลชนิดขาดตอน (Discrete value) การป้อนข้อมูลโดยวิธีการนี้เป็นการเรียกใช้ฟังก์ชั่น InputBox ของ Visual Basic ให้แสดงช่องรับข้อมูลบนหน้าจอ ลักษณะ InputBox จะมีลักษณะเป็นกรอบสี่เหลี่ยมผืนผ้า มีช่องรับข้อมูล , ปุ่มคำสั่ง OK และ Cancel ดังรูปที่ 3.3



รูปที่ 3.3 แสดง InputBox ของโปรแกรม SPC_Engine

2.) การรับข้อมูลเข้าโดยการส่งข้อมูลผ่านพอร์ตสื่อสารอนุกรมตามมาตรฐาน RS 232 (Auto Data Entry) แสดงดังรูปที่ 3.4 วิธีการนี้ทำได้โดยการเชื่อมต่อคอมพิวเตอร์เข้ากับเครื่องมือวัด (Measuring Equipment) ที่สามารถส่งข้อมูลผ่านทางพอร์ตอนุกรมได้เช่นกันซึ่งกล่าวได้ว่า เป็นการส่งข้อมูลผ่าน RS232 (Sending the data over RS 232 interface) ลักษณะของพอร์ตอนุกรมที่ใช้ ถ้าเป็นเครื่องคอมพิวเตอร์ส่วนบุคคล จะเป็นแบบ 9 ขา (DB 9) ในขณะที่เครื่องมือวัด พอร์ตที่ใช้ อาจเป็นได้ทั้งแบบ 9 ขา (DB 9) หรือแบบ 25 ขา (DB 25) ซึ่งหลักการออกแบบสาย RS-232 นั้นขึ้นอยู่กับประเภทของฮาร์ดแวร์ที่จะทำการเชื่อมต่อด้วย โดยรายละเอียดในส่วนนี้ได้แสดงในภาคผนวก ข



รูปที่ 3.4 แสดงการส่งข้อมูลโดยผ่าน RS 232 ของโปรแกรม SPC_Engine

การออกแบบในส่วนนี้จะใช้ Custom Control ของ Visual Basic ที่เรียกว่า MS Comm. โดย object นี้จะมี Properties และ Method สำหรับสั่งคอมพิวเตอร์ให้มีการเปิดพอร์ต (Com Port Open) และ กำหนดข้อมูลในการสื่อสารที่เรียกว่า Serial Protocal ซึ่งได้แก่ Baud Rate, Data Bit , Parity Bit , และ Stop Bit ดังเช่น Baud Rate = 9600 Bps, 8 Data Bits , No Parity , 1 Stop Bit เป็นต้น ในที่นี้จะสร้างหน้าต่าง ที่เรียกว่า RS 232 Configuration ซึ่งประกอบไปด้วย Option box สำหรับกำหนด Serial Protocal ก่อนที่การส่งข้อมูลจะเกิดขึ้น เนื่องจากว่าโปรแกรมนี้ทำหน้าที่เป็นตัวรับข้อมูลเท่านั้น ดังนั้น หลักการทำงานของโปรแกรม จึงประกอบไปด้วย (1.) การกำหนด Serial Protocal (2.) การเปิดพอร์ต (Com port Open) 3.) การนับจำนวนอักขระที่เข้ามาใน Receive Buffer (In Buffer Count) 3.) การอ่านข้อมูลที่เข้ามาใน Receive Buffer (Data Input) และ 4.) การปิดพอร์ต (Com port Close) ตามลำดับ

นอกจากนี้การออกแบบโปรแกรมในส่วนนี้ยังรวมถึงการสร้างฐานข้อมูล ที่ใช้ สำหรับเก็บข้อมูล Serial protocal ของ เครื่องมือวัดต่างๆที่ต้องการเชื่อมต่อกับระบบ เพื่อไว้สำหรับอ้างอิง ในที่นี้ทำได้โดยใช้ Microsoft Access สร้าง ตารางฐานข้อมูลที่ประกอบไปด้วย Fields ต่างๆ ได้แก่ Equipment name , Baud Rate , Data Bit , Parity Bit , และ Stop bit จากนั้นอาศัย Object ที่ใช้สำหรับจัดการฐานข้อมูล ที่เรียกว่า Data Control จากนั้นทำการเชื่อมต่อ ตารางฐานข้อมูลของ Microsoft Access นี้ เข้ากับ Microsoft Visual Basic เพื่อให้สามารถจัดการกับข้อมูลต่างๆ โดยดำเนินการผ่าน Microsoft Visual Basic

การป้อนข้อมูลเข้าสำหรับข้อมูลชนิดต่อเนื่อง หรือ ที่เรียกว่า Measurement Data นั้นได้มีการออกแบบให้มีลำดับในการป้อนข้อมูลเข้า (Measurement Sequence) ได้ 3 แบบ ดังนี้

1.) ลำดับในการป้อนข้อมูลเข้าแบบ M Sequence มีรูปแบบการป้อนข้อมูลดังนี้

Sample 1 ตัวอย่างที่ 1 ของพารามิเตอร์ที่ทำการวัดค่า

Sample 2 ตัวอย่างที่ 2 ของพารามิเตอร์ที่ทำการวัดค่า

Sample 3 ตัวอย่างที่ 3 ของพารามิเตอร์ที่ทำการวัดค่า

.

.

.

Sample n ตัวอย่างที่ n ของพารามิเตอร์ที่ทำการวัดค่า

2.) ลำดับในการป้อนข้อมูลเข้าแบบ F Sequence มีรูปแบบการป้อนข้อมูลดังนี้

Parameter 1

Sample 1 ตัวอย่างที่ 1 ของพารามิเตอร์ที่ 1

Sample 2 ตัวอย่างที่ 2 ของพารามิเตอร์ที่ 1

.

.

Sample n ตัวอย่างที่ n ของพารามิเตอร์ที่ 1

Parameter 2

Sample 1 ตัวอย่างที่ 1 ของพารามิเตอร์ที่ 2

Sample 2 ตัวอย่างที่ 2 ของพารามิเตอร์ที่ 2

.

.

Sample n ตัวอย่างที่ n ของพารามิเตอร์ที่ 2

Parameter n

Sample 1 ตัวอย่างที่ 1 ของพารามิเตอร์ที่ n

Sample 2 ตัวอย่างที่ 2 ของพารามิเตอร์ที่ n

.

.

Sample n ตัวอย่างที่ n ของพารามิเตอร์ที่ n

3.) ลำดับในการป้อนข้อมูลเข้าแบบ P Sequence มีรูปแบบการป้อนข้อมูลดังนี้

Sample 1 ตัวอย่างที่ 1 ของทุกๆ พารามิเตอร์ที่ทำการวัด

Parameter 1

Parameter 2

.

.

Parameter n

Sample 2 ตัวอย่างที่ 2 ของทุกๆ พารามิเตอร์ที่ทำการวัด

Parameter 1

Parameter 2

.

.

Parameter n

.

.

Sample n ตัวอย่างที่ n ของทุกๆ พารามิเตอร์ ที่ทำการวัด

Parameter 1

.

Parameter n

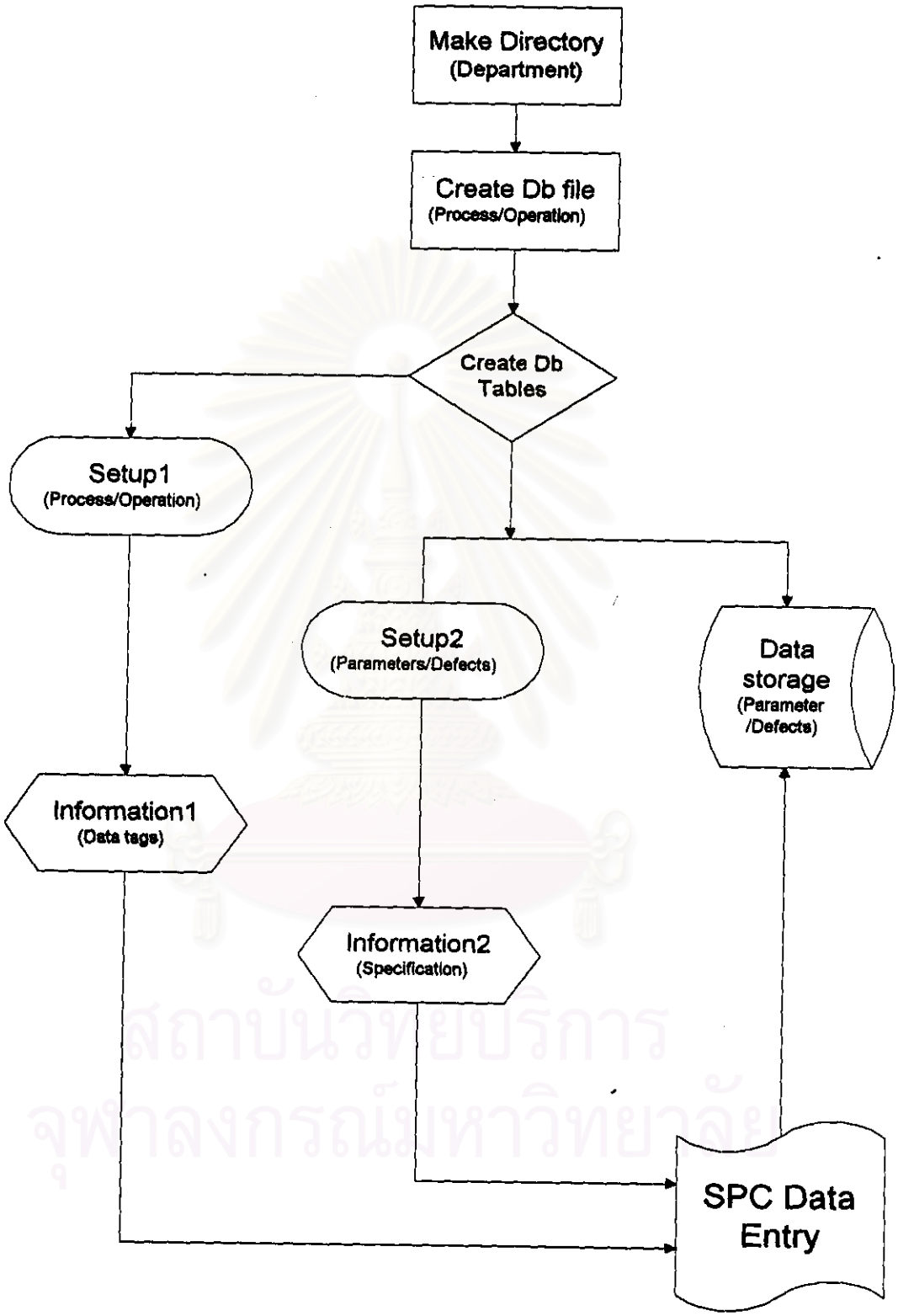
การป้อนข้อมูลเข้า สำหรับข้อมูลชนิดขาดตอน หรือ ที่เรียกว่า Defect Data นั้น ได้มีการออกแบบการรับข้อมูลเข้าในรูปแบบของ จำนวนของเสีย (Decfective Unit) และ จำนวนรอยตำหนิ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

(Defect point) เพื่อให้สามารถคำนวณ สัดส่วนของเสียที่เกิดขึ้น และสัดส่วนรอยตำหนิที่เกิดบนตัวผลิตภัณฑ์ โดยใช้แนวคิดที่ว่า ของเสีย 1 ชิ้น มีอาการเสีย 1 อาการ และมี จำนวนตำหนิตั้งแต่ 1 รอยขึ้นไป ทั้งนี้ เนื่องจากว่า ในทางปฏิบัตินั้น วิธีการตรวจสอบที่เรียกว่า Visual Inspection มักจะเน้นที่อาการเสียที่รุนแรงสูงไปหาอาการเสียรุนแรงต่ำกว่า ดังนั้น เมื่อมีการตรวจพบอาการเสียแรก ชิ้นงานนั้นก็จะถูกปฏิเสธ (Reject) ไปทันที จากนั้นจึงมีการพิจารณาชิ้นงานถัดไป เมื่อเสร็จสิ้นการตรวจชิ้นงานข้อมูลต่างๆ จะอยู่ในใบ Check Sheet ซึ่งสามารถนำไป ป้อนผ่านลงสู่ ระบบควบคุมกระบวนการเชิงสถิติแบบตามเวลาจริงต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. ระบบฐานข้อมูลแบบเชิงเวลาจริง (Real time Database)



รูปที่ 3.5 แสดง Flow chart ของ Real time Database

การออกแบบและพัฒนาโปรแกรม ในส่วนนี้ เป็นการสร้างระบบฐานข้อมูลที่สนับสนุนการทำงานแบบเชิงเวลาจริง เพื่อให้สอดคล้องกับการทำงานในพื้นที่การผลิต ซึ่งมีความจำเป็นอย่างยิ่งในการรับข้อมูลที่ถูกต้อง รวดเร็ว และทันต่อเหตุการณ์ที่กำลังดำเนินอยู่ ตลอดจนความยืดหยุ่นในการใช้งาน ดังนั้นลักษณะของฐานข้อมูลในที่นี้ จึงจำเป็นต้องมีคุณลักษณะที่สอดคล้องกับความต้องการต่างๆดังที่กล่าวมา

การออกแบบฐานข้อมูลด้วย Microsoft Visual Basic สามารถทำได้โดยการใช้ Object ของฐานข้อมูล (Database Object) ซึ่งในที่นี้จะมีการใช้ Database Object 2 อย่างร่วมกัน ได้แก่

1. JET Database Engine
2. Data Control

ระบบการจัดการฐานข้อมูล JET นั้นใช้สำหรับการออกแบบในส่วนการสร้างระบบฐานข้อมูลขึ้นมาในขณะ Run time ตลอดจนฟังก์ชั่นการในใช้งาน ที่เกี่ยวกับการจัดการฐานข้อมูล กล่าวคือ ผู้ใช้งาน (User) สามารถสร้างฐานข้อมูลขึ้นใหม่ (Create new database) ที่จะมารองรับพารามิเตอร์ตามที่ต้องการ ในขณะ Run โปรแกรมได้ทันที หรือแม้แต่การลบฐานข้อมูล ที่มีการยกเลิกการใช้งานแล้วได้เช่นเดียวกัน ส่วนนี้ จึงจัดได้ว่า มีการตอบสนองต่อการใช้งานเชิงเวลาจริง และมีความยืดหยุ่นต่อการนำไปปฏิบัติในพื้นที่การผลิต.

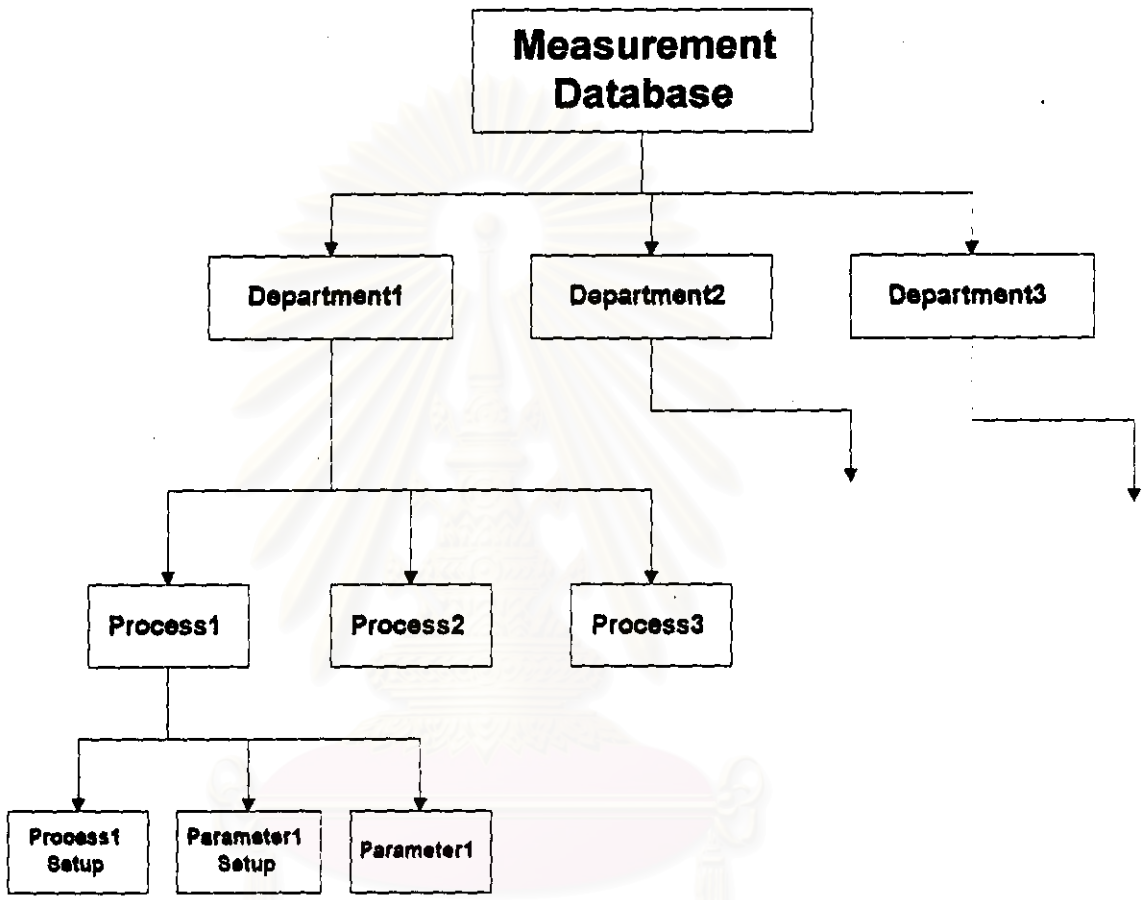
กรณีของการจัดการฐานของมูลด้วย Data Control นั้นใช้สำหรับการออกแบบ ในส่วน ของการจัดการฐานข้อมูลในระดับ Record set กล่าวคือการใช้งาน Data Control เป็นการดำเนินการกับข้อมูลของแต่ละ Record ในตารางฐานข้อมูล เช่น การดูข้อมูลในแต่ละ record โดยการเลื่อน (Move) ทีละ record หรือ การค้นหา record ที่ต้องการ (Find) การแก้ไข (Edit) , ก ารลบ (Delete) และอื่นๆ เป็นต้น

การออกแบบระบบฐานข้อมูล สามารถแบ่งออกได้ 2 ส่วนใหญ่ๆ ตามชนิดของข้อมูลที่จัดเก็บดังต่อไปนี้

1. ระบบฐานข้อมูลสำหรับ Variable Data
   หรือ ในที่นี้หมายถึง Measurement Data
2. ระบบฐานข้อมูลสำหรับ Attribute Data
   หรือ ในที่นี้หมายถึง Defect Data

1. ระบบฐานข้อมูลสำหรับ Measurement Data

โครงสร้างฐานข้อมูลของ Measurement Data สามารถแสดงได้ ดังรูปที่ 3.6 ดังต่อไปนี้



รูปที่ 3.6 แสดงโครงสร้างของ Measurement Database ของโปรแกรม SPC_Engine

จาก Block diagram ข้างต้น แสดงองค์ประกอบของ Measurement Database ซึ่งการจัดเก็บข้อมูลต่างๆจะทำภายใต้ Path ของ C: \SPCE \ แฟ้มข้อมูลที่จัดเก็บจะอยู่ในรูปของ Microsoft Access โดยรายละเอียดในการออกแบบมีดังต่อไปนี้

1.) การออกแบบ Department

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้าง Directory สำหรับจัดเก็บแฟ้มฐานข้อมูลต่างๆ (Database files) ด้วยคำสั่งที่จัดการเกี่ยวกับระบบแฟ้มข้อมูล (file) ของ Microsoft Visual Basic โดยชื่อของ Department จะตั้งให้สอดคล้องกับความเป็นจริงเพื่อให้การจัดเก็บข้อมูลมีความเป็นสัดส่วนอย่างเหมาะสม ตัวอย่างการตั้งชื่อของ Department ได้แก่

Department 1 = Machining

Department 2 = Monitor

Department 3 = SPS

Department 4 = QA

Department 5 = Finishing

Department 6 = EMI

Deparment 7 = R & D

2.) การออกแบบ Process

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างแฟ้มฐานข้อมูล (Database file) ซึ่งจะอยู่ภายใต้ Directory ของ Department ขั้นตอนการสร้าง Department และ Process ในที่นี้ ได้ออกแบบให้สร้างขึ้นในขั้นตอนเดียวกันโดยจะอยู่ในส่วนของ Process Setup แฟ้มฐานข้อมูลที่สร้างขึ้นนี้ จะมีรูปแบบเดียวกันกับ Microsoft Access ที่มาจากการใช้คำสั่ง Create Database ของ JET Engine ใน Microsoft Visual Basic การตั้งชื่อ Process นั้นสามารถให้เป็นอะไรก็ได้ แต่เพื่อความสะดวกในการใช้งานสามารถตั้งให้สอดคล้องกับกระบวนการผลิตที่กำลังสนใจอยู่ ดังเช่น

Process 1 = Wire bond

Process 2 = Probing

Process 3 = White balance

Process 4 = Auto scrubing

Process 5 = Wave Soldering

Process 6 = Lapping

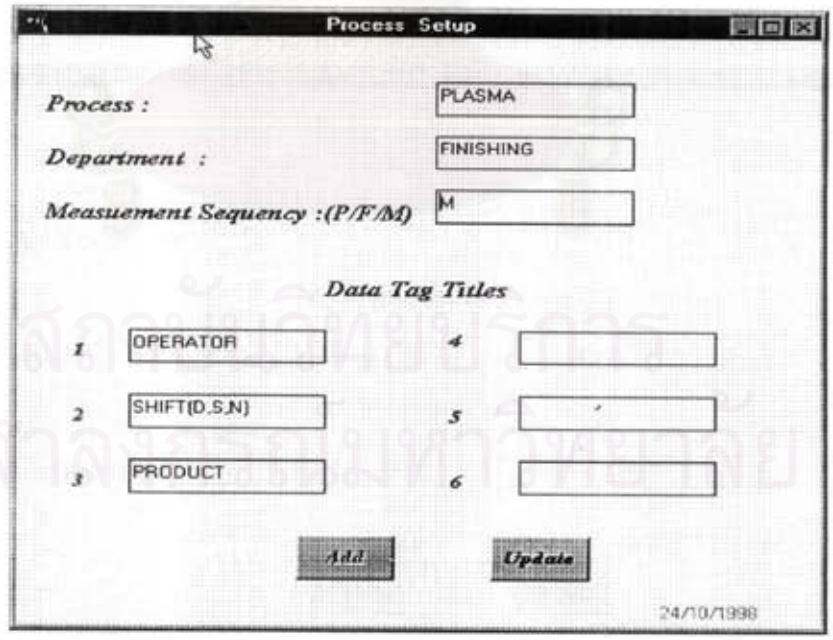
Process 7 = Ecthing

3.) การออกแบบ Process Setup

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างตารางฐานข้อมูล (Database Tables) ที่อยู่ภายใต้ แฟ้มฐานข้อมูล (Database files) ที่สร้างขึ้นของแต่ละ Department ข้อมูลของ Process Setup จะมีเพียง record เดียวเท่านั้น โดยมี Fields ต่างๆ ดังต่อไปนี้

1. Department สำหรับ เก็บชื่อของ Department
2. Process สำหรับ เก็บชื่อของ Process
3. Measurement Sequence สำหรับ เก็บลำดับการป้อนข้อมูล
4. Data tag1 สำหรับ เก็บหัวเรื่องที่ 1
5. Data tag2 สำหรับ เก็บหัวเรื่องที่ 2
6. Data tag3 สำหรับ เก็บหัวเรื่องที่ 3
7. Data tag4 สำหรับ เก็บหัวเรื่องที่ 4
8. Data tag5 สำหรับ เก็บหัวเรื่องที่ 5
9. Data tag6 สำหรับ เก็บหัวเรื่องที่ 6

การสร้างตาราง Database ส่วนนี้จะเกิดขึ้นขณะ Run time โดยมีรูปแบบ ของ User Interface รูปที่3.7



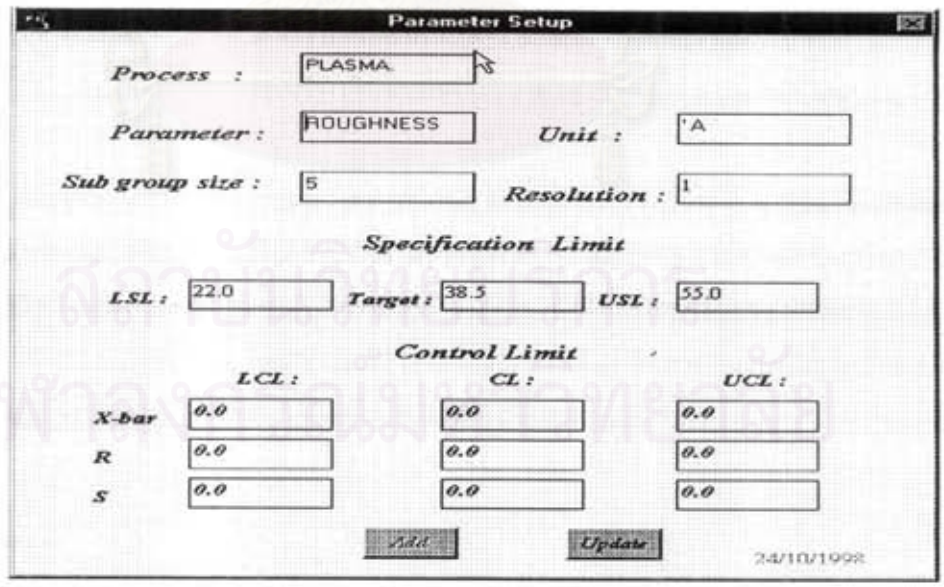
รูปที่ 3.7 แสดง User Interface ของ Process Setup ในโปรแกรม SPC_Engine

4.) การออกแบบ Parameter Setup

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างตารางฐานข้อมูล (Database Tables) สำหรับเก็บข้อมูลในส่วนของการ Set up ของพารามิเตอร์ ที่กำลังสนใจอยู่ ด้วย JET Engine ของ Microsoft Visual Basic โดยข้อมูลจะมีเพียง 1 record เท่านั้น Database Fields ในส่วนนี้ได้แก่

1. Process สำหรับ เก็บชื่อของ Process
2. Parameter สำหรับ เก็บชื่อของ Parameter
3. Unit สำหรับ เก็บ หน่วยของการวัด
4. Sub group no สำหรับ เก็บ ขนาดของการสุ่มตัวอย่าง
5. Resolution สำหรับ เก็บ ค่าความละเอียดในการวัดเป็นจำนวนจุดทศนิยม
6. LSL สำหับ เก็บ ค่าขอบเขตข้อกำหนดด้านล่าง (Lower Spec.)
7. Target สำหับ เก็บ ค่าเป้าหมายที่ต้องการ
8. USL สำหรับ เก็บค่าขอบเขตข้อกำหนดด้านบน (Upper Spec.)
9. Control limit สำหรับ เก็บค่าขอบเขตควบคุม ของแผนภูมิควบคุมชนิดต่างๆ

การสร้างตาราง Parameter Setup นี้จะเกิดในขณะที่ Run time ของโปรแกรมเช่นเดียวกัน โดยมีรูปแบบของ User Interface ดังรูปที่ 3.8



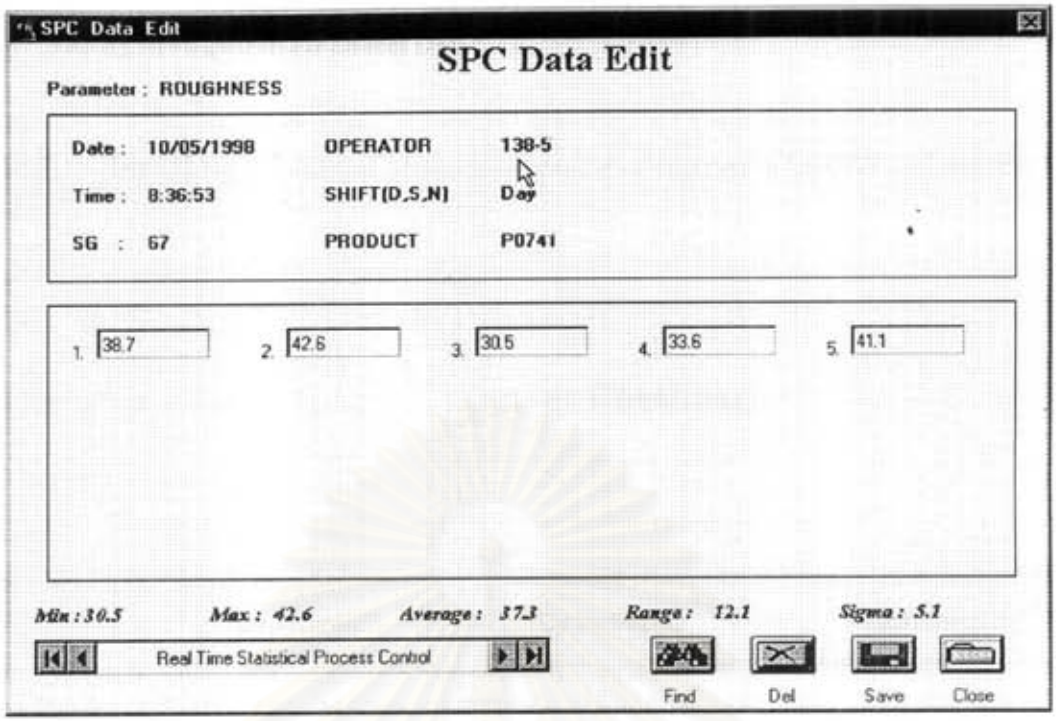
รูปที่ 3.8 แสดง User Interface ของ Parameter Setup ในโปรแกรม SPC_Engine

5.) การออกแบบ Parameter

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างตารางฐานข้อมูลหลัก สำหรับการเก็บข้อมูลดิบที่ได้จากการวัด (Measurement Data) ดังนั้นจำนวน record จึงมีตั้งแต่ 1 record เป็นต้นไป ตาราง Parameter นี้ประกอบด้วย Database Fields ต่างๆ ดังต่อไปนี้

1. Sub no	สำหรับ เก็บ ตัวเลขลำดับที่ของแต่ละ Subgroup
2. Date	สำหรับ เก็บ ตัวเลขของวันที่
3. Time	สำหรับ เก็บ ตัวเลขของเวลา
4. Data tag1	สำหรับ เก็บข้อมูลของ Data tag1
5. Data tag2	สำหรับ เก็บข้อมูลของ Data tag2
6. Data tag3	สำหรับ เก็บข้อมูลของ Data tag3
7. Data tag4	สำหรับ เก็บข้อมูลของ Data tag4
8. Data tag5	สำหรับ เก็บข้อมูลของ Data tag5
9. Data tag6	สำหรับ เก็บข้อมูลของ Data tag6
10. Samp1 ~ Samp n	สำหรับ เก็บข้อมูล ของ Sample ที่ 1 ถึง Sample ที่ n
11. Min	สำหรับ เก็บค่าต่ำสุดของ Sample ใน Subgroup
12. Max	สำหรับ เก็บค่าสูงสุดของ Sample ใน Subgroup
13. Average	สำหรับ เก็บค่าเฉลี่ยของ Sample ใน Subgroup
14. Range	สำหรับ เก็บค่าพิสัยของ Sample ใน Subgroup
15. Sigma	สำหรับ เก็บค่าเบี่ยงเบนมาตรฐาน ของ Sample ใน Subgroup

การสร้างตาราง Parameter นี้ จะทำพร้อมๆ กับขั้นตอนการสร้างตาราง Parameter Setup โดยใช้ User Interface เดียวกัน แต่ยังไม่มีข้อมูลในตารางดังกล่าว เมื่อมีการป้อนข้อมูลเข้าสู่ระบบ ข้อมูลจะถูกจัดเก็บ ลงสู่ตารางของ Parameter table ทุกๆ records ถ้าต้องการดูข้อมูลของแต่ละ record ทำได้โดยใช้ SPC Data Edit ซึ่ง เกิดจากการใช้ Data Control Object ในการจัดการกับฐานข้อมูลได้อย่างมีประสิทธิภาพ โดยมีรูปแบบของ User Interface ดังรูปที่ 3.9

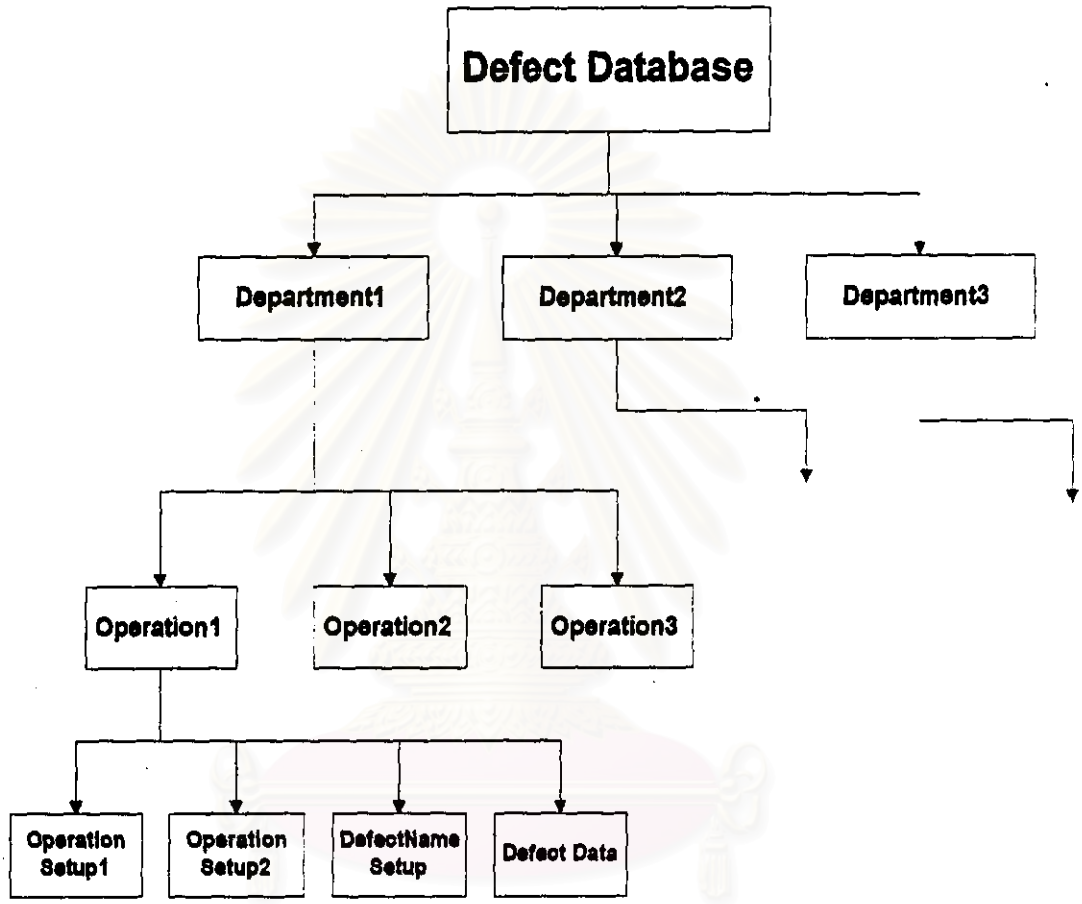


รูปที่ 3.9 แสดง User Interface ของ SPC Data Edit ในโปรแกรม SPC_Engine

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. ระบบฐานข้อมูลสำหรับ Defect Data

โครงสร้างฐานข้อมูลของ Measurement Data สามารถแสดงได้ดังรูปที่ 3.10 ดังต่อไปนี้



รูปที่ 3.10 แสดงโครงสร้างของ Defect Database ในโปรแกรม SPC_Engine

จาก Block diagram ข้างต้น แสดงองค์ประกอบของ Defect Database ซึ่งการจัดเก็บข้อมูลต่างๆจะทำภายใต้ Path ของ C: \SPDE \ แฟ้มข้อมูลที่จัดเก็บจะอยู่ในรูปของ Microsoft Access โดยรายละเอียดในการออกแบบมีดังต่อไปนี้

1.) การออกแบบ Department

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้าง Directory สำหรับจัดเก็บแฟ้มฐานข้อมูลต่างๆ (Database files) ด้วยคำสั่งที่จัดการเกี่ยวกับระบบแฟ้มข้อมูล (file) ของ Microsoft Visual Basic โดยชื่อของ Department จะตั้งให้สอดคล้องกับความเป็นจริง ทั้งนี้ ชื่อของDepartment จะตั้งให้สอดคล้องกับความเป็นจริง เพื่อให้การจัดเก็บข้อมูล มีความเป็นสัดส่วน อย่างเหมาะสม ตัวอย่างการตั้งชื่อของ Department ได้แก่

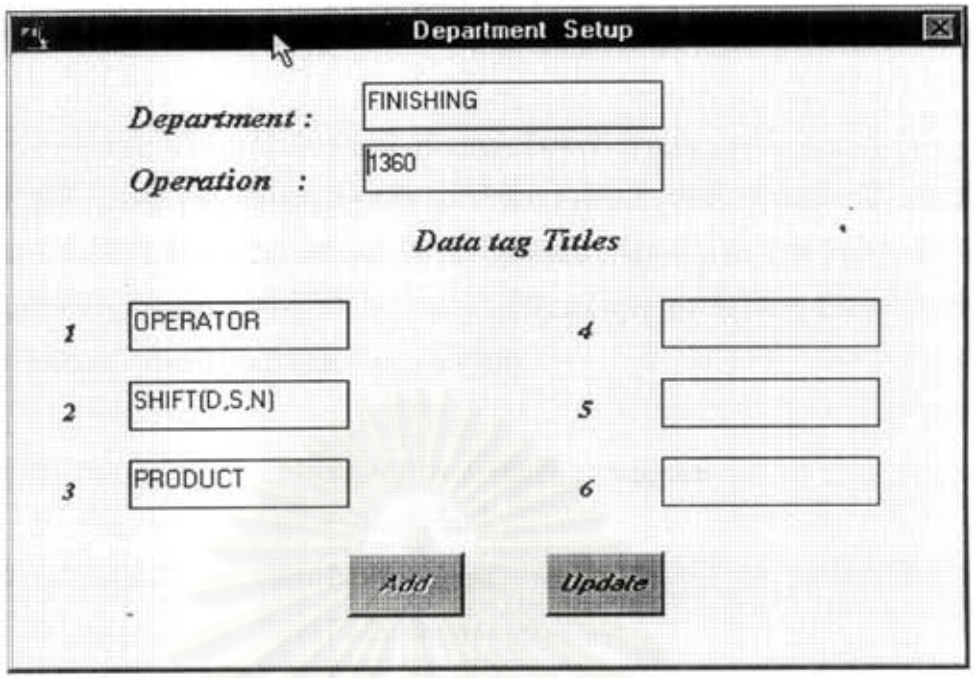
Department1 = Product Eng.
Department2 = Machining
Department3 = Lapping
Department4 = Finishing

2.) การออกแบบ Operation Setup#1

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างแฟ้มฐานข้อมูลของ Defect data ณ ขณะ Run time ของโปรแกรม โดยอาศัยวิธีการ Create Database ของ JET Engine จาก Microsoft Visual Basic ลักษณะแฟ้มฐานข้อมูลที่สร้างขึ้นนี้ จะมีรูปแบบเดียวกันกับ Microsoft Access และ มีการสร้างตารางฐานข้อมูล (Database Table) สำหรับเก็บค่า Set up ต่างๆ ของ แฟ้มฐานข้อมูล Operation โดยที่ข้อมูลที่ถูกจัดเก็บมี เพียง 1 record เท่านั้น Data Fields ต่างๆ ของตาราง DeptSetup ได้แก่

1. Department สำหรับเก็บชื่อของ Department
2. Operation สำหรับเก็บชื่อของ Operation
3. Data tag1 สำหรับ เก็บหัวเรื่องที่ 1
4. Data tag2 สำหรับ เก็บหัวเรื่องที่ 2
5. Data tag3 สำหรับ เก็บหัวเรื่องที่ 3
6. Data tag4 สำหรับ เก็บหัวเรื่องที่ 4
7. Data tag5 สำหรับ เก็บหัวเรื่องที่ 5
8. Data tag6 สำหรับ เก็บหัวเรื่องที่ 6

ขั้นตอนในการสร้าง Directory ของ Department และ แฟ้มฐานข้อมูล Operation Setup จะเกิดขึ้น ในเวลาและ ขั้นตอนเดียวกัน โดยมีรูปแบบของ User Interface ดังรูปที่ 3.11 ต่อไปนี้



รูปที่ 3.11 แสดง User Interface ของ Department Setup ในโปรแกรม SPC_Engine

3.) การออกแบบ DefectName

การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างตารางฐานข้อมูล (Database Table) สำหรับเก็บค่า Defect codes ต่างๆของ แต่ละ Operation ที่อยู่ภายใต้ฐานข้อมูล Operation โดยตารางนี้ประกอบไปด้วย Filelds ต่างๆ ดังต่อไปนี้

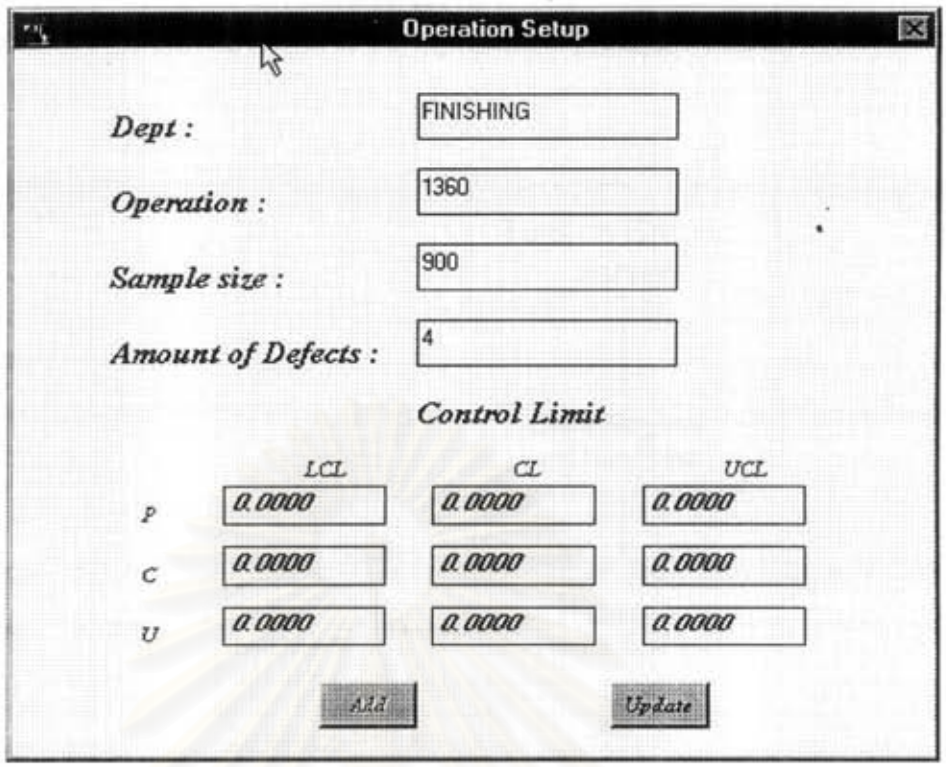
1. Defs1 สำหรับเก็บ ชื่อ Defect code ตัวที่ 1
2. Defs2 สำหรับเก็บ ชื่อ Defect code ตัวที่ 2
3. Defs3 สำหรับเก็บ ชื่อ Defect code ตัวที่ 3
4. . สำหรับเก็บ ชื่อ Defect code ตัวที่ 4
5. . สำหรับเก็บ ....
. . สำหรับเก็บ .....
n . Defs n สำหรับเก็บ ชื่อ Defect code ตัวที่ n

4.) การออกแบบ OperatonSetup#2 และ Operation

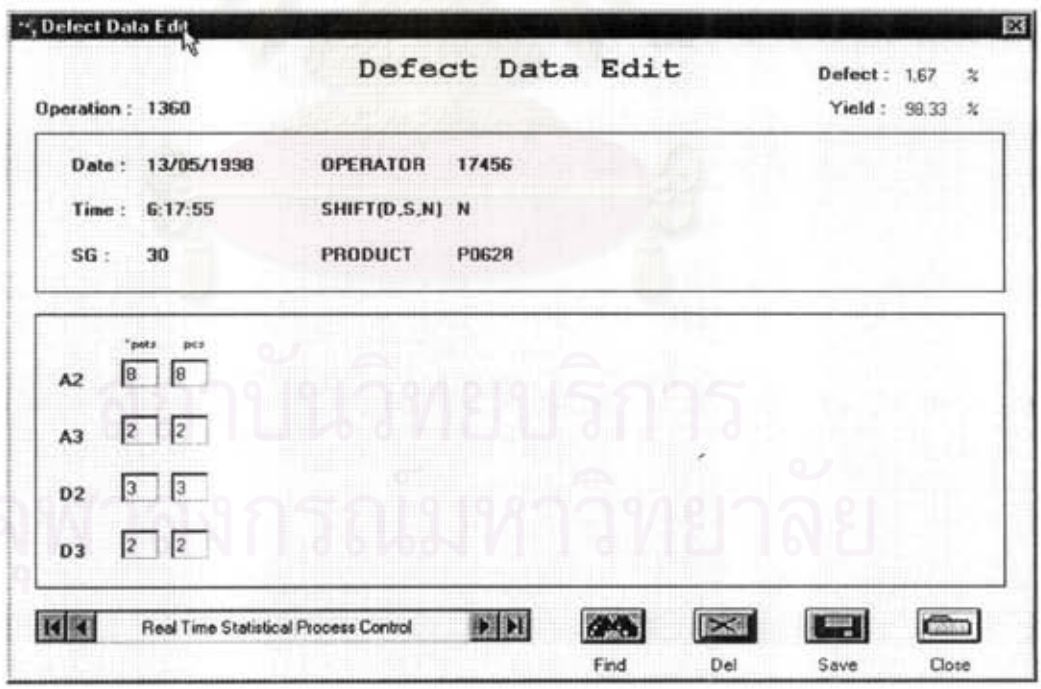
การออกแบบโปรแกรมในส่วนนี้ เป็นการสร้างตารางฐานข้อมูล (Database Tables) โดยที่ตาราง OperationSetup#2 เป็นตารางฐานข้อมูลที่มีเพียง 1 record ทำหน้าที่ เก็บค่า Set up ต่างๆ ของ Operation ซึ่งได้แก่ ชื่อของ Department , ชื่อของ Operation , Subgroup size , และ จำนวน Defect code ในขณะที่ ตาราง Operation ทำหน้าที่เก็บข้อมูลหลักที่ป้อนเข้าสู่ระบบ ซึ่งได้แก่ จำนวนของเสีย และจำนวนรอยตำหนิ โดยมี Data fields ต่างๆ ดังต่อไปนี้

1. Sub no	สำหรับ เก็บค่าลำดับเลขที่ของ Subgroup
2. Date	สำหรับ เก็บค่า วันที่
3. Time	สำหรับ เก็บ ค่า เวลา
4. Data tag1	สำหรับเก็บ ข้อมูลของ Data tag1
5. Data tag2	สำหรับเก็บ ข้อมูลของ Data tag2
6. Data tag3	สำหรับเก็บ ข้อมูลของ Data tag3
7. Data tag4	สำหรับเก็บ ข้อมูลของ Data tag4
8. Data tag5	สำหรับเก็บ ข้อมูลของ Data tag5
9. Data tag6	สำหรับเก็บ ข้อมูลของ Data tag6
10. Def1 ~ Def n	สำหรับ เก็บ จำนวนจุดตำหนิของตัวอย่างที่ 1 ~ ตัวอย่างที่ n
11. defx1 ~ defx n	สำหรับ เก็บ จำนวนชิ้นงานเสียของตัวอย่างที่ 1 ~ ตัวอย่างที่ n
12. Total Defect	สำหรับ เก็บ จำนวนรอยตำหนิทั้งหมด
13. TTL Defective	สำหรับ เก็บ จำนวนของเสียทั้งหมด
14. Yield	สำหรับเก็บค่าผลผลิตที่ได้ (Yield)

ขั้นตอนการทำงานทั้งหมดของการสร้างตารางฐานข้อมูล OperationSetup#2 และ Operation จะดำเนินการ ในขั้นตอนเดียวกัน ณ เวลา Run time ของโปรแกรม โดยมีรูปแบบของ User Interface แสดงดังรูปที่ 3.12 เมื่อมีการป้อนข้อมูลเข้าสู่ระบบ ข้อมูลจะถูกจัดเก็บลงในตารางฐานข้อมูลดังกล่าวข้างต้น ซึ่งข้อมูลดังกล่าว สามารถเรียกดูได้จาก คำสั่ง SPC Data Edit ใน mode ของ Defect data การทำงานของส่วนนี้ถูกควบคุมด้วย Data Control Object ที่ใช้ในการจัดการฐานข้อมูลของแต่ละ record ได้อย่างมีประสิทธิภาพ แสดงดังรูปที่ 3.13

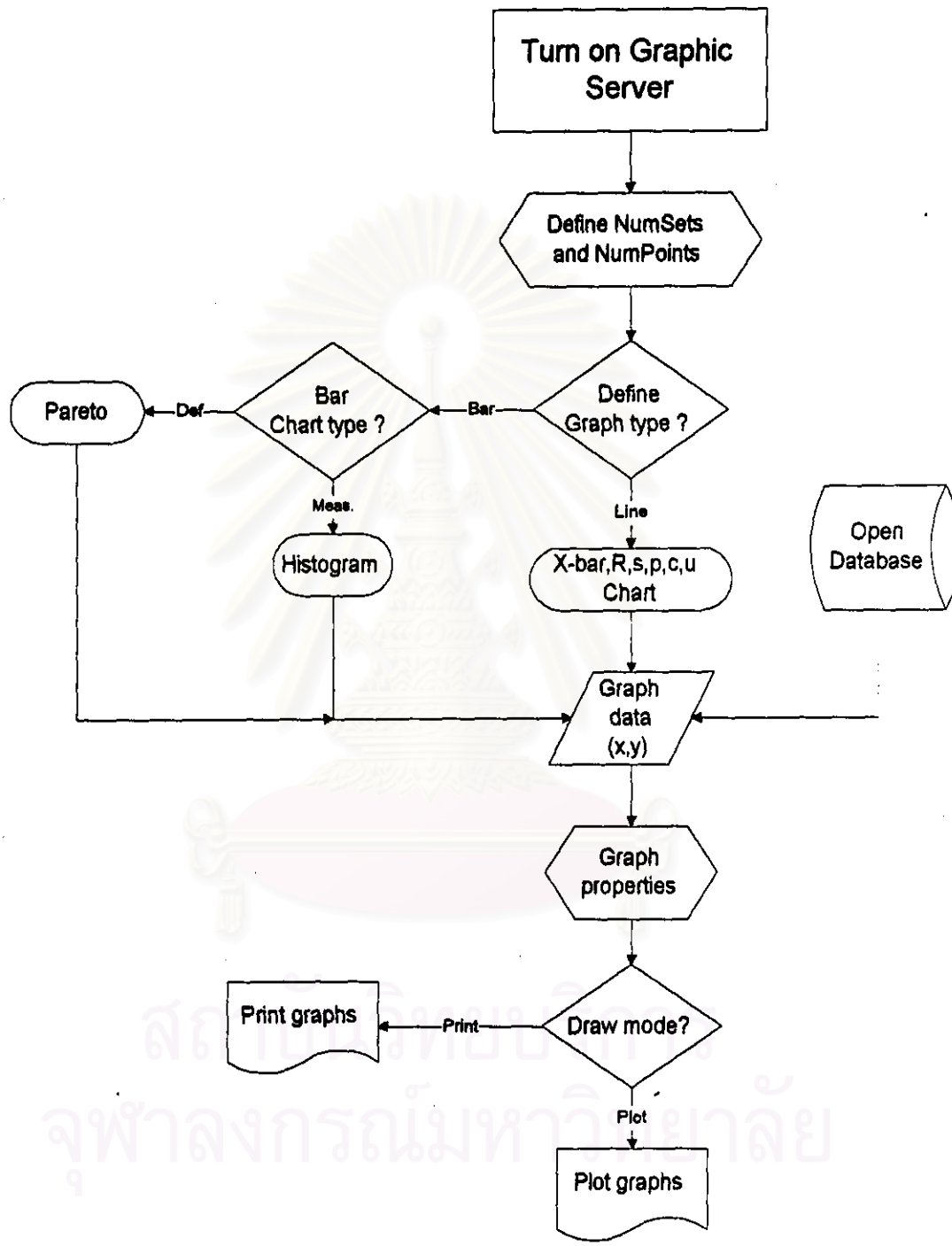


รูปที่ 3.12 แสดง User Interface ของ Operation Set up ในโปรแกรม SPC_Engine



รูปที่ 3.13 แสดง User Interface ของ Defect SPC Data Edit ในโปรแกรม SPC_Engine

3.) ระบบแสดงผลทางด้านกราฟฟิคของแผนภูมิชนิดต่างๆ (Graphic Charts)



รูปที่ 3.14 แสดง Flow chart ของ Graphic Chart

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

การออกแบบและพัฒนาโปรแกรมในส่วนของ การแสดงผลทางด้านกราฟฟิค ในรูปของแผนภูมิต่างๆ การออกแบบครั้งนี้มีการใช้โปรแกรมเสริม (Add on Program) ที่ชื่อว่า Graphic Server ร่วมกับ Microsoft Visual Basic เพื่อเพิ่มเติมศักยภาพให้กับ Visual Basic ทางด้านการออกแบบกราฟฟิคในการสร้างกราฟชนิดต่างๆ ได้อย่างมีประสิทธิภาพ นอกจากนี้ Graphic Server ยังตอบสนองต่อเหตุการณ์ต่างๆที่กระทำต่อ Object ของ Graphic Server เช่น การ Click เมาส์ที่ จุดของกราฟ (Data points) แล้วทำให้แสดงข้อมูลของจุดนั้นๆได้ (Real time reporting)

การเชื่อมต่อ Graphic Server ทำได้โดยการติดตั้ง Software ของ Graphic Server ลงสู่ Hard drive ที่มีโปรแกรม Visual Basic ติดตั้งไว้ก่อนแล้ว จากนั้นทำการ Add Custom Control ของ Graphic Server เพื่อให้สามารถเรียกใช้งานได้ เปรียบเสมือนการเรียกใช้ Custom Control Object ทั่วๆไปออกมาใช้งานเช่นเดียวกัน โดยหลักการของ Graphic Server คือ การเรียกใช้ Properties ต่างๆ ในการสร้าง Graph เช่น Graphtype, GraphData , ColorData , Datafield , DrawMode , LegendText , LabelText และอื่นๆ ที่ เกี่ยวข้อง ซึ่งสามารถทำได้ ทั้งขณะ Design time และ Run time โดยทั้งนี้ ขึ้นอยู่กับความเหมาะสมในการทำงานของโปรแกรม

การออกแบบแผนภูมิชนิดต่างๆจะแบ่งประเภทตามชนิดของข้อมูลซึ่งในที่นี้สามารถจำแนกออกเป็น 2 กลุ่มใหญ่ๆ ดังนี้

1. แผนภูมิสำหรับข้อมูลชนิดต่อเนื่อง (Continuous data or Variable data)
   1.1 แผนภูมิควบคุมค่าเฉลี่ยของกระบวนการ (X- Bar Chart)
   1.2 แผนภูมิควบคุมค่าพิสัยของกระบวนการ (R Chart)
   1.3 แผนภูมิควบคุมค่าเบี่ยงเบนมาตรฐานของกระบวนการ (S Chart)
   1.4 ฮีสโตแกรม (Histogram)

2. แผนภูมิสำหรับข้อมูลชนิดขาดตอน (Discrete data or Attribute data)
   2.1 แผนภูมิควบคุมสัดส่วนของเสีย (P Chart)
   2.2 แผนภูมิควบคุมจำนวนข้อบกพร่อง (C Chart)
   2.3 แผนภูมิควบคุมจำนวนข้อบกพร่องต่อหน่วย (U Chart)
   2.4 แผนภาพพาเรโต (Pareto diagram)

หลักการออกแบบการสร้างกราฟชนิดต่างๆ

1.) การกำหนดจำนวนจุดข้อมูล (Data points) ในที่นี้ได้กำหนดให้สามารถแสดงได้สูงสุดทีละ 50 จุด

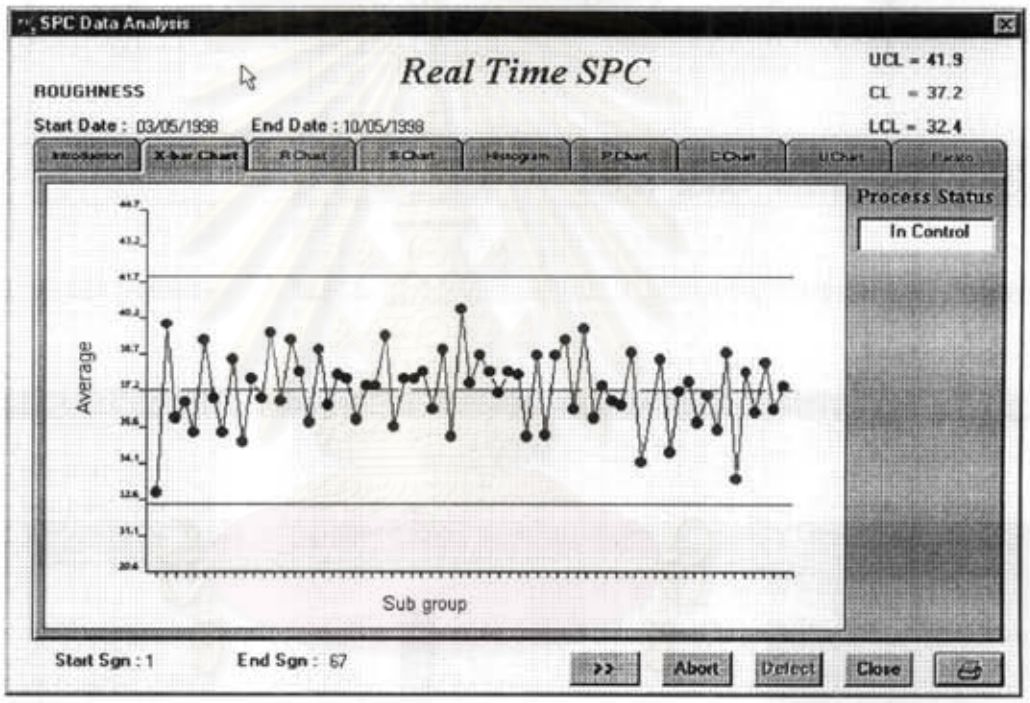
2.) การเชื่อมต่อ Graphic Server เข้ากับตารางฐานข้อมูลเพื่อนำข้อมูลของแต่ละ record มาพล็อตกราฟ

3.) กำหนดชนิดของกราฟที่จะใช้งาน เช่น Line , Bar , Pareto เป็นต้น

4.) กำหนด GraphData ของ แกน X และ แกน Y

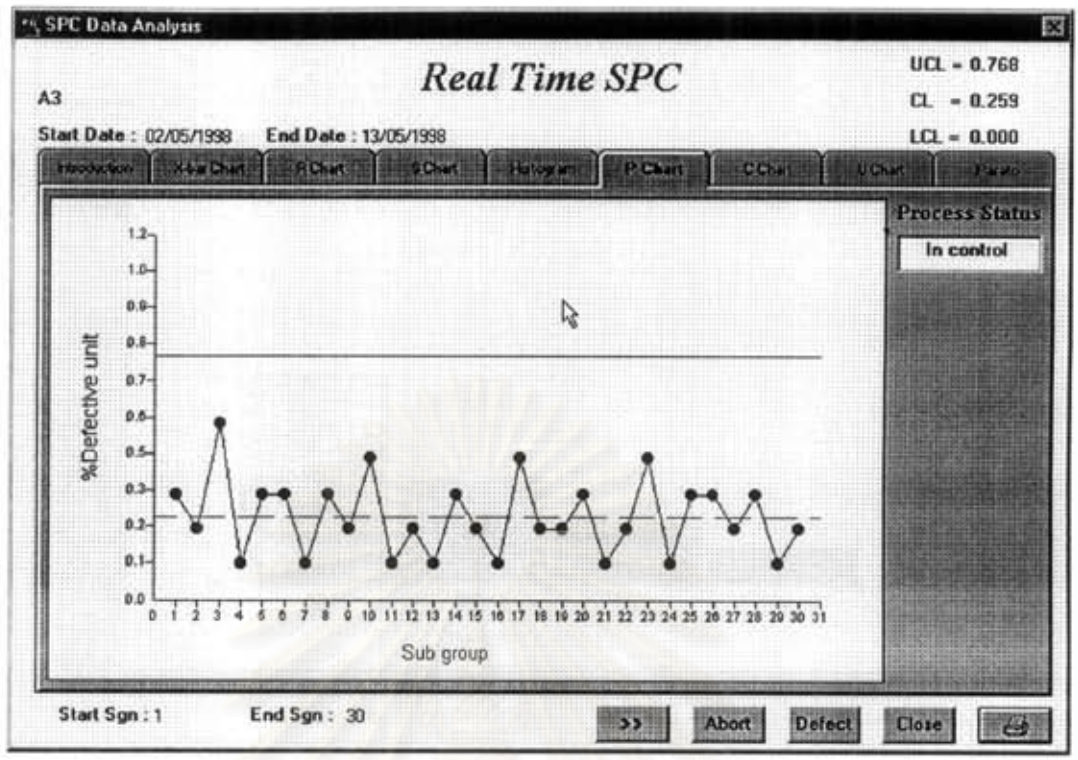
5.) กำหนด Properties อื่นๆ ที่สนับสนุนการสร้างกราฟ เช่น Color , Legend , Label ,etc.

6.) รูปแบบของ User Interface ของกราฟต่างๆแสดงดังรูปที่ 3.15 , 3.16 , 3.17 , 3.18

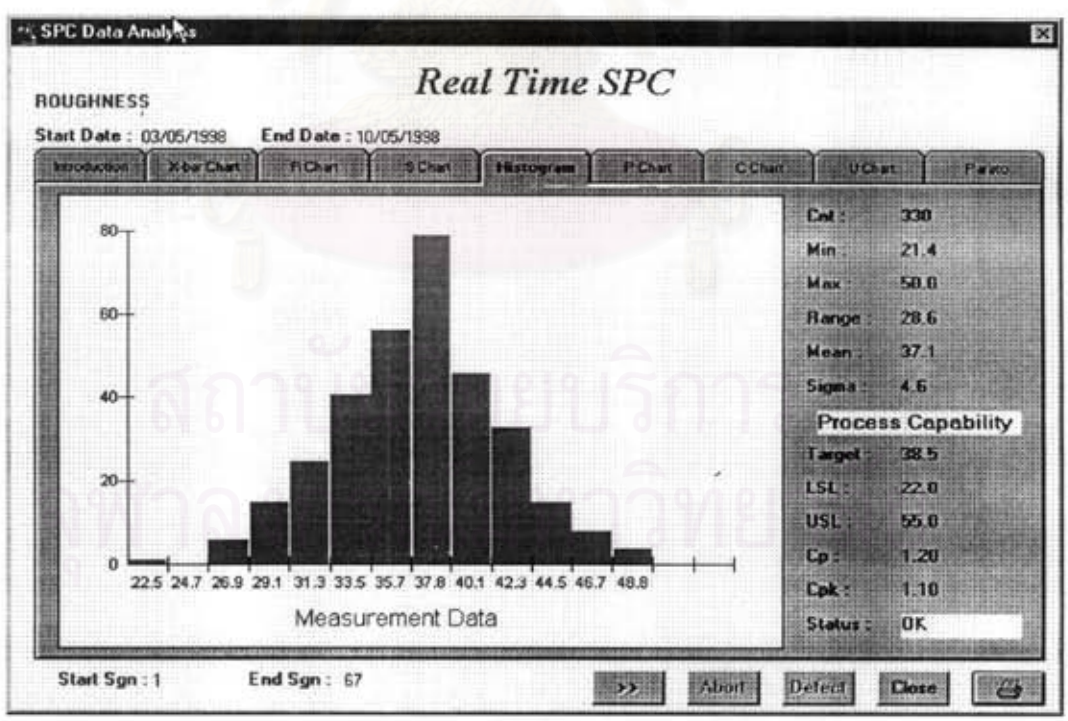


รูปที่ 3.15 แสดงลักษณะUser Interface ของ X bar , R , S Chart ในโปรแกรม SPC_Engine

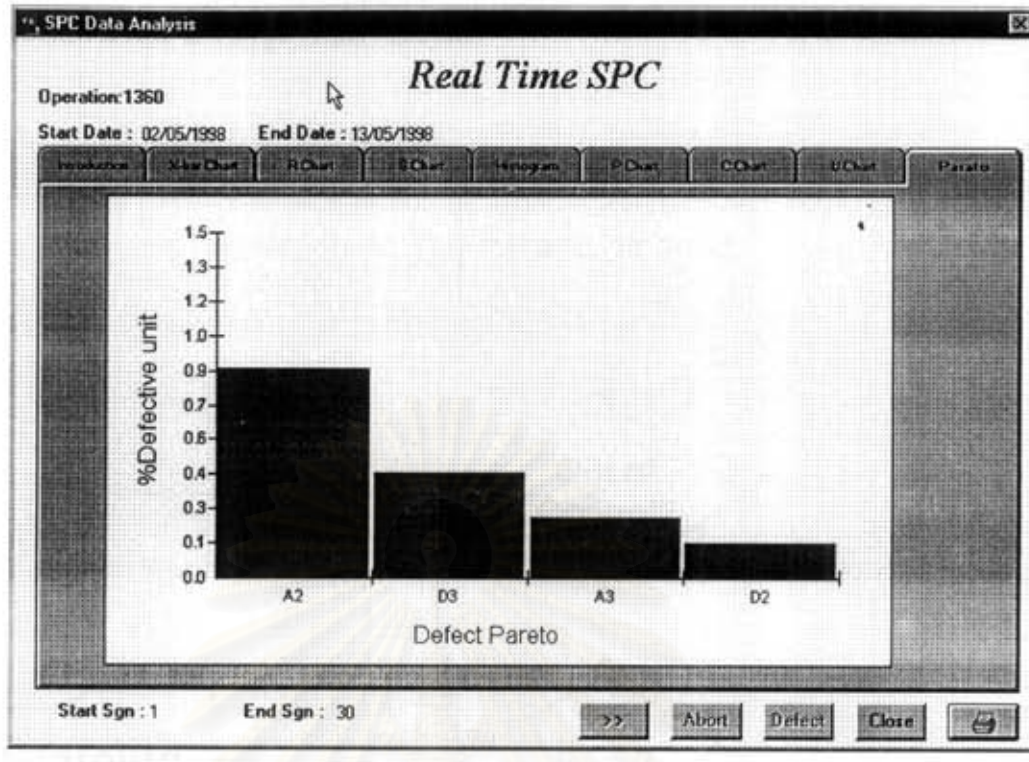
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.16 แสดงลักษณะUser Interface ของ P ,U ,C Chart ในโปรแกรม SPC_Engine



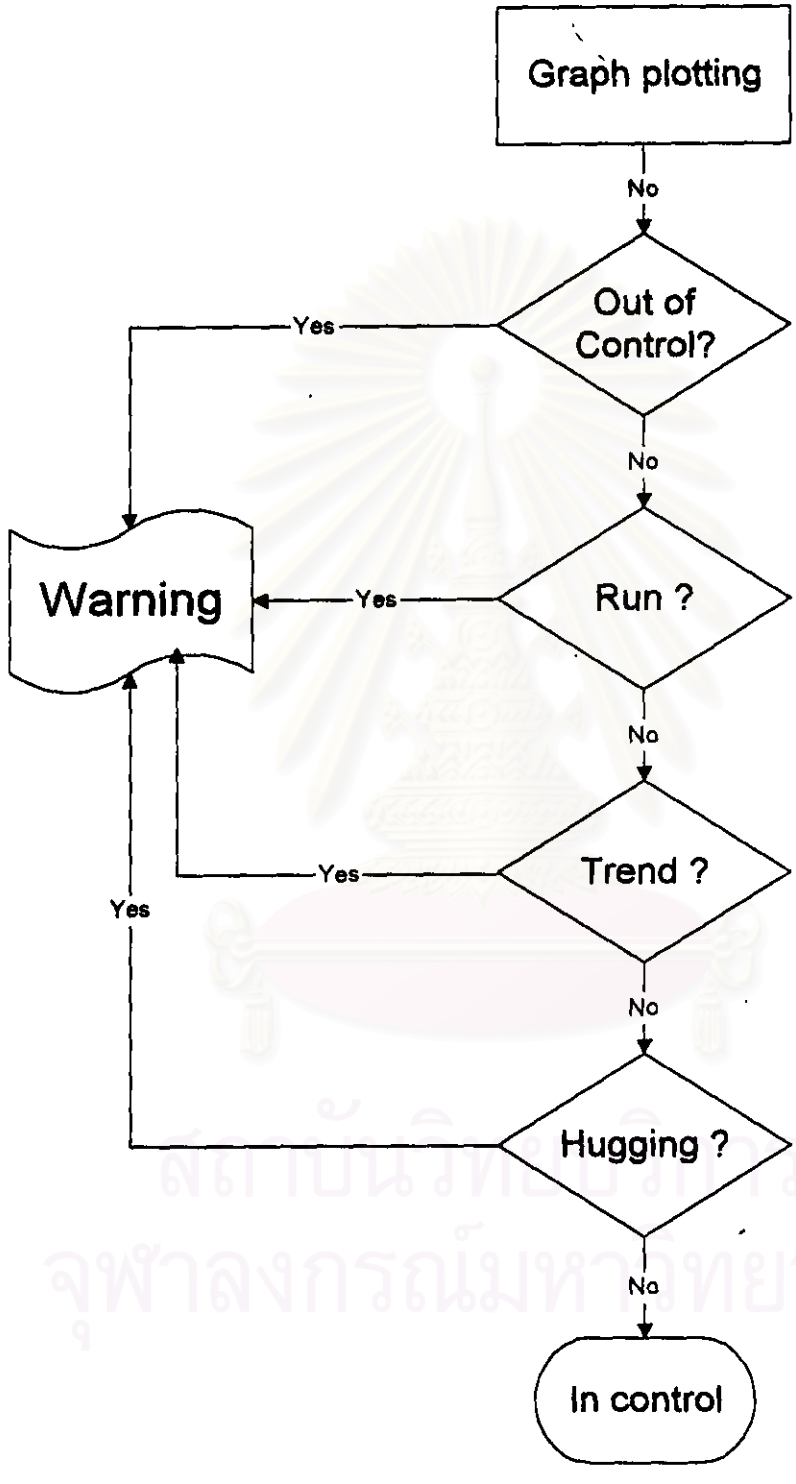
รูปที่ 3.17 แสดง User Interface ของ Histogram ในโปรแกรม SPC_Engine



รูปที่ 3.18 แสดง User Interface ของ Pareto diagram ในโปรแกรม SPC_Engine

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

4. ระบบการวิเคราะห์ข้อมูล (Data Analysis)



รูปที่ 3.19 แสดง Flow chart ของ Data analysis

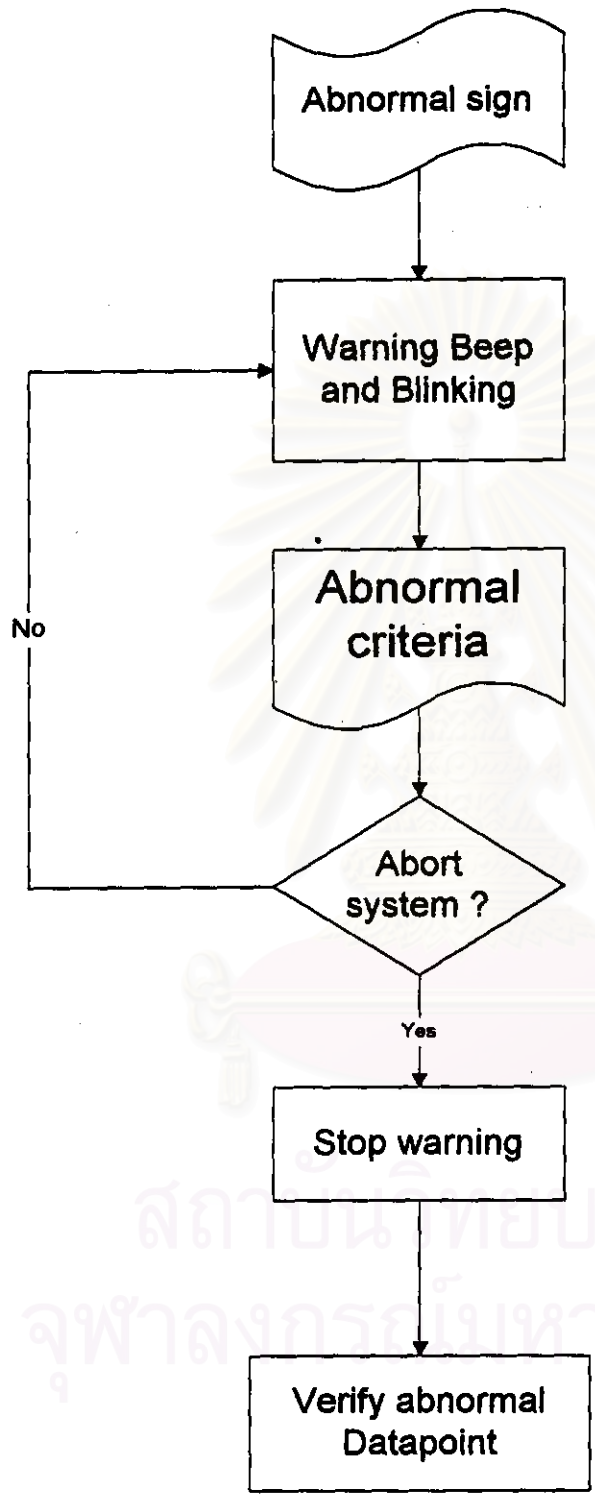
การออกแบบและพัฒนาโปรแกรมในส่วนนี้ จะทำงานอยู่กับส่วนแสดงผลทางกราฟฟิค โดยทุกๆ ครั้งที่มีการพล็อตกราฟ ระบบจะทำการวิเคราะห์แผนภูมิควบคุมตามเกณฑ์ดังนี้

1. การวิเคราะห์ว่ามีจุดข้อมูลใดๆออกนอกเส้นควบคุม UCL และ LCL หรือไม่ (Out of Control)
2. การวิเคราะห์ว่ามีจุดข้อมูล 5~7 จุด วิ่งอยู่ด้านใดด้านหนึ่งของเส้นควบคุม ทั้งสองหรือไม่ (Run)
3. การวิเคราะห์ว่ามีจุดข้อมูล 5~7 จุด มีแนวโน้มขึ้นหรือลงหรือไม่ (Trend)
4. การวิเคราะห์ว่ามีจุดข้อมูลเกาะกลุ่ม อยู่ใกล้เส้นกึ่งกลาง CL ของเส้นควบคุม (Hugging of the control line)
5. การวิเคราะห์ว่ามีจุดข้อมูลเกาะกลุ่ม อยู่ใกล้เส้นควบคุม (Approach to the limit)

สำหรับรูปแบบอื่นๆ เช่น การเกิด Cyclic pattern นั้นอยู่นอกขอบเขตการวิจัยครั้งนี้ เนื่องจากว่า ลักษณะของ Cyclic pattern นั้นไม่มีกฎเกณฑ์ที่แน่ นอนตายตัว จึงต้องอาศัยการวิเคราะห์ด้วยการพิจารณารูปแบบที่ปรากฏ (Visual Detection) นอกจากกรณีเหล่านี้แล้วจะถือได้ว่ากระบวนการอยู่ภายใต้การควบคุม

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

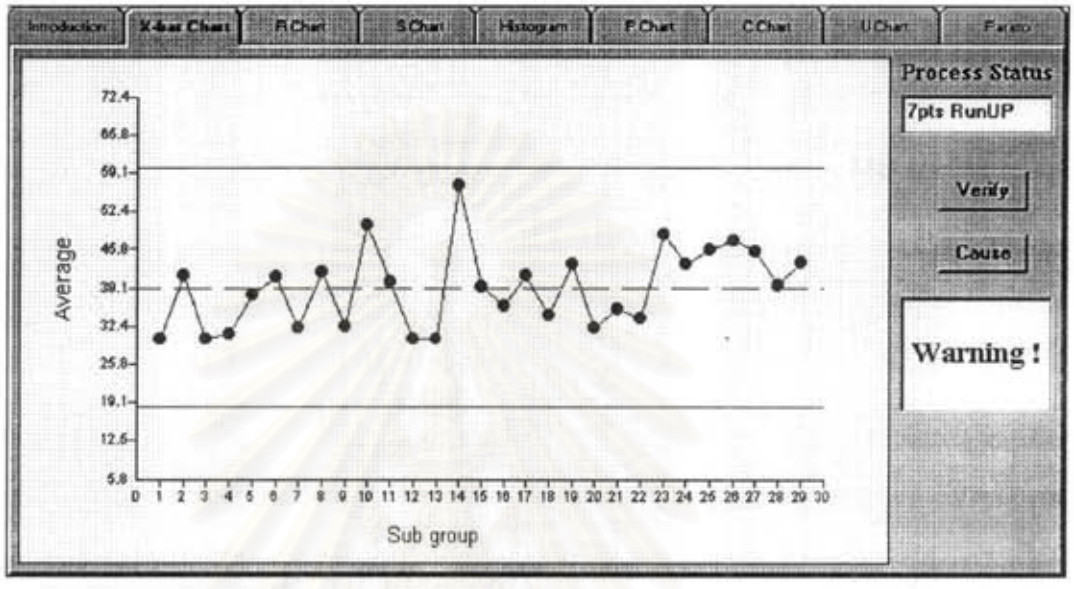
5. ระบบแสดงการเตือน (Warning)



รูปที่ 3.20 แสดง Flow chart ของระบบWarning ในโปรแกรม SPC_Engine

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

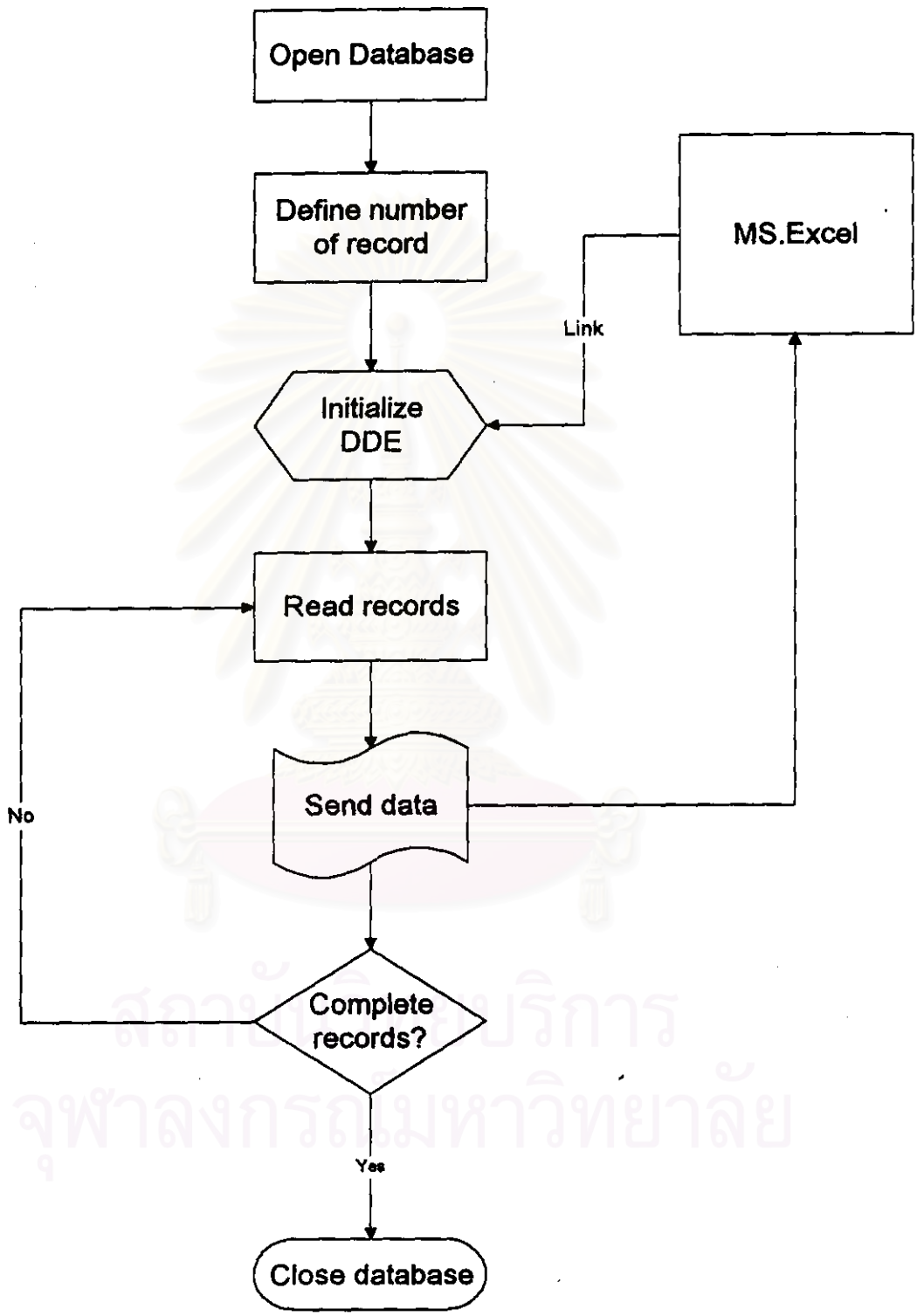
ระบบแสดงการเตือนนี้จะทำงานร่วมกับระบบการวิเคราะห์ข้อมูลโดยที่ถ้าการวิเคราะห์พบว่ากระบวนการได้ออกนอกการควบคุม ไม่ว่าเกิดจากกรณีใดๆก็ตาม ระบบจะแสดงการเตือนทันที โดยแสดงข้อความว่า "Warning" กระพริบพร้อมเสียง Beep ตลอดเวลา เพื่อเป็นการเตือนให้ผู้ปฏิบัติงานได้รับทราบแสดง ดังรูปที่ 3.21



รูปที่ 3.21แสดง User Interface ของระบบ Warning ในโปรแกรม SPC_Engine

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

6. ระบบการส่งข้อมูลออกสู่ภายนอก (Data Exporting)



รูปที่ 3.22 แสดง Flow chart ของระบบ Data Exporting ในโปรแกรม SPC_Engine

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

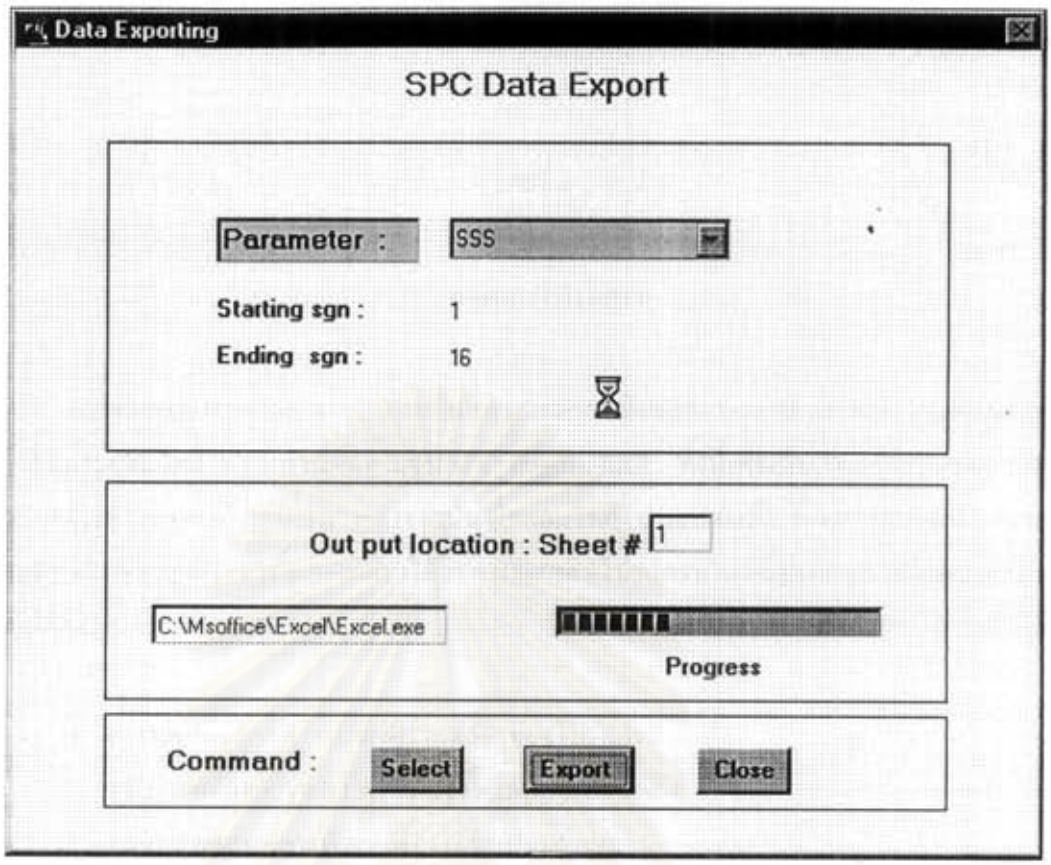
การออกแบบและพัฒนาโปรแกรม ในส่วนของการส่งข้อมูลออกสู่ภายนอกเป็นการออกแบบให้ ระบบสามารถส่งข้อมูลจากฐานข้อมูลไปสู่แอพพลิเคชั่นอื่นๆ ที่รันภายใต้Windows ในที่นี้ได้เลือกใช้โปรแกรม Microsoft Excel เป็นแอพพลิเคชั่นที่ใช้รองรับข้อมูลที่ส่งจากฐานข้อมูล ของโปรแกรม SPC_ Engine เนื่องจากว่า Microsoft Excel เป็นโปรแกรมทางด้าน Spread Sheet และมีความสามารถในการคำนวน ได้อย่างมีประสิทธิภาพ ตลอดจนเป็นที่นิยมในหมู่ผู้ใช้ คอมพิวเตอร์ทั่วๆไป

หลักการสำคัญ ในการสร้างฟังชั่นการทำงานในส่วนนี้คือ Dynamic Data Exchange หรือ ที่เรียกย่อๆ ว่า DDE เป็นความสามารถส่วนหนึ่งของระบบ Windows ที่ให้การสนับสนุนด้านการแลกเปลี่ยนข้อมูล และการพูดคุยระหว่างแอพพลิชั่นต่างๆที่รันบน Windows การแลกเปลี่ยนข้อมูลระหว่างแอพพลิเคชั่นโดยต้นทางและแอพพลิเคชั่นปลายทาง คล้ายๆกับการพูดคุยกันระหว่าง 2 ฝ่าย ดังนั้นขบวนการที่ DDE ใช้ในการแลกเปลี่ยนข้อมูลระหว่างกันนี้เรียกว่า การพูดคุยระหว่างแอพพลิเคชั่น (Conversation) ส่วนแอพพลิ เคชั่นทั้งสองที่มีการแลกเปลี่ยนข้อมูลจะถูกเรียกว่า แอพพลิเคชั่นต้นทาง (Source Application) และแอพพลิ เคชั่นปลายทาง (Destination Application) โดยแอพพลิเคชั่น ที่ทำหน้าที่กำหนดค่าเริ่มต้น สำหรับเรียกร้องการพูดคุยจะถูกเรียกว่า แอพพลิเคชั่นปลายทาง ส่วนแอพพลิเคชั่นที่ตอบสนองต่อการเรียกร้องนี้เรียกว่า แอพพลิเคชั่นต้นทาง

ขั้นตอนการทำงานของ Data Exporting มีดังนี้

1. การกำหนดฐานข้อมูลที่จะใช้สำหรับส่งข้อมูลออก
2. การกำหนดช่วงข้อมูลที่ต้องการ Export และ work sheet สำหรับรับข้อมูล
3. การ load โปรแกรม Microsoft Excel
4. การกำหนดค่าเริ่มต้นของ DDE
5. การเปิดตารางฐานข้อมูลเพื่ออ่านค่าในแต่ละ record
6. การส่งข้อมูลออกจาก SPC_Engine ไปสู่ Microsoft Excel ที่ละ record จนครบตามที่กำหนดไว้
7. การปิดฐานข้อมูล

รูปแบบของ User Interface ของระบบการส่งข้อมูลออกสู่ภายนอก Darta Exporting แสดงดังรูปที่ 3.23



รูปที่ 3.23 แสดง User Interface ของระบบ Data Exporting ในโปรแกรม SPC_Engine

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย