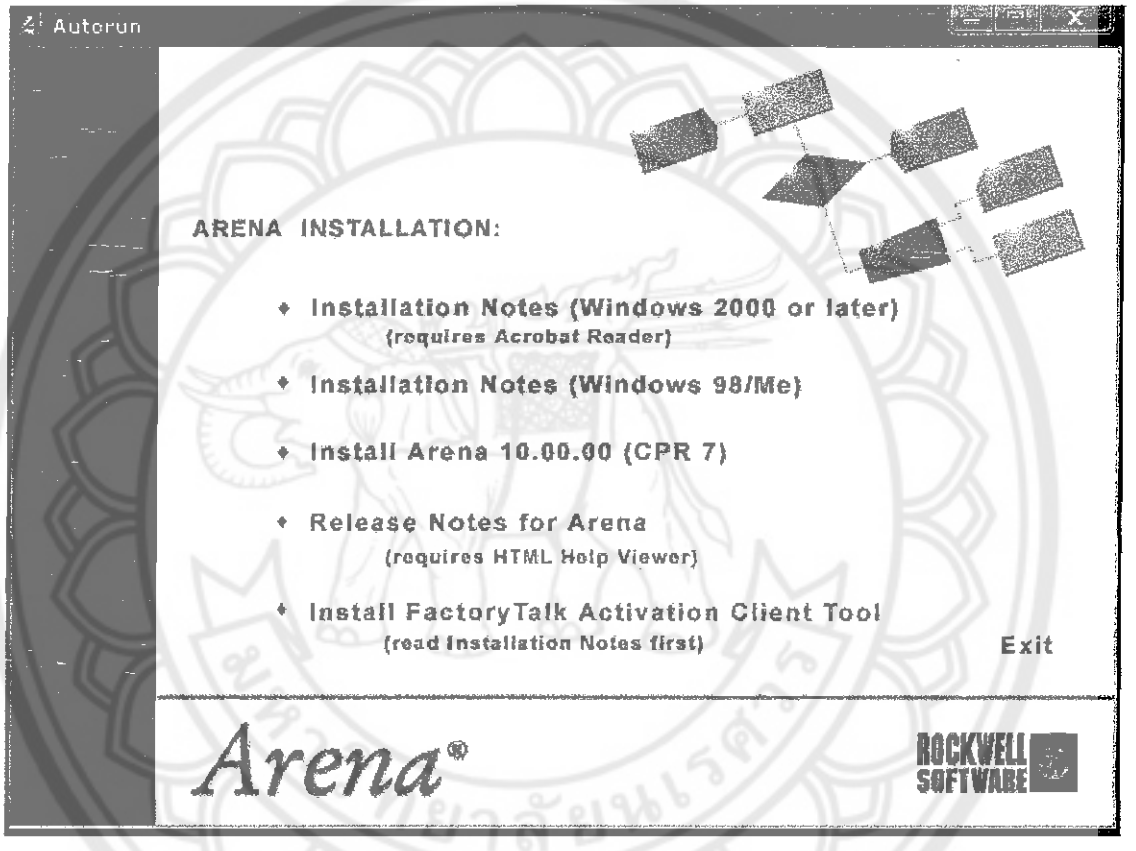
ภาคผนวก ก.
วิธีการติดตั้งโปรแกรม Arena

ภาคผนวก ก.
วิธีการติดตั้งโปรแกรม Arena

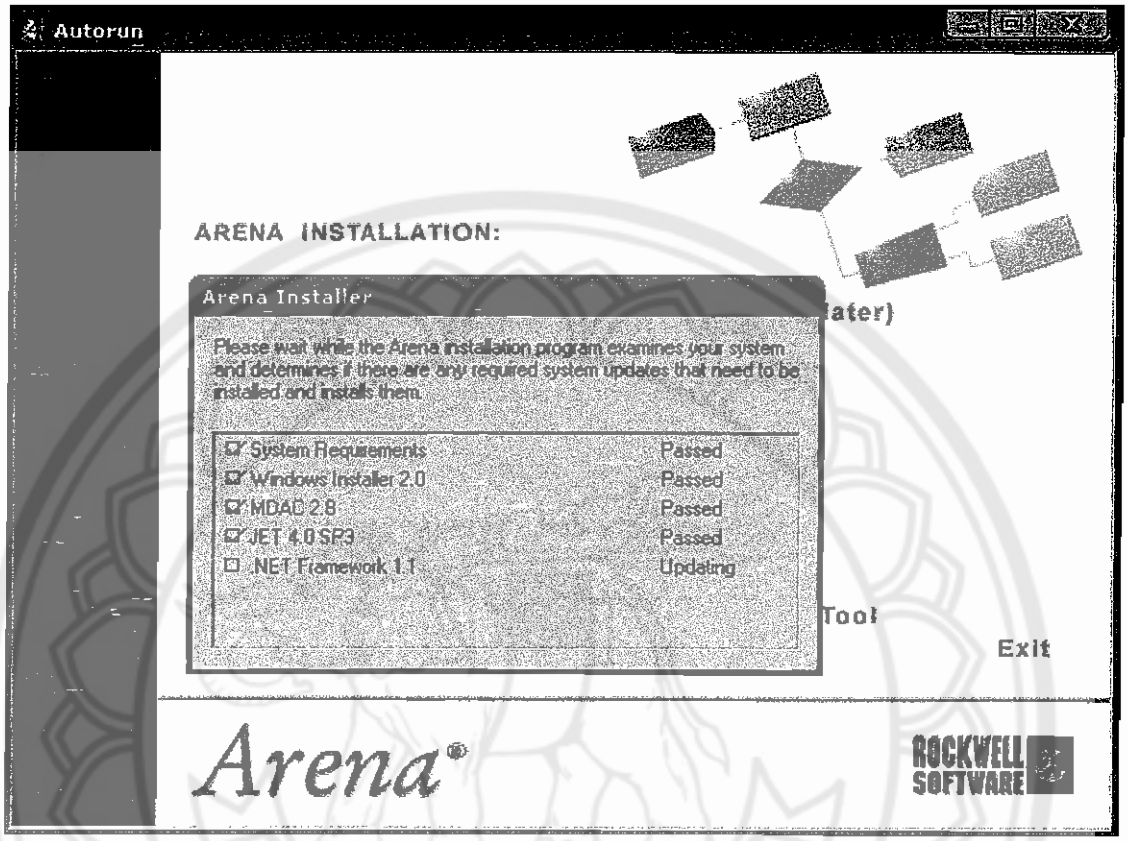
1 การตั้งโปรแกรม Arena V.10

1.1 ใส่แผ่นโปรแกรม Arena V.10 จะปรากฏหน้าจอของการติดตั้งโปรแกรมดังรูป



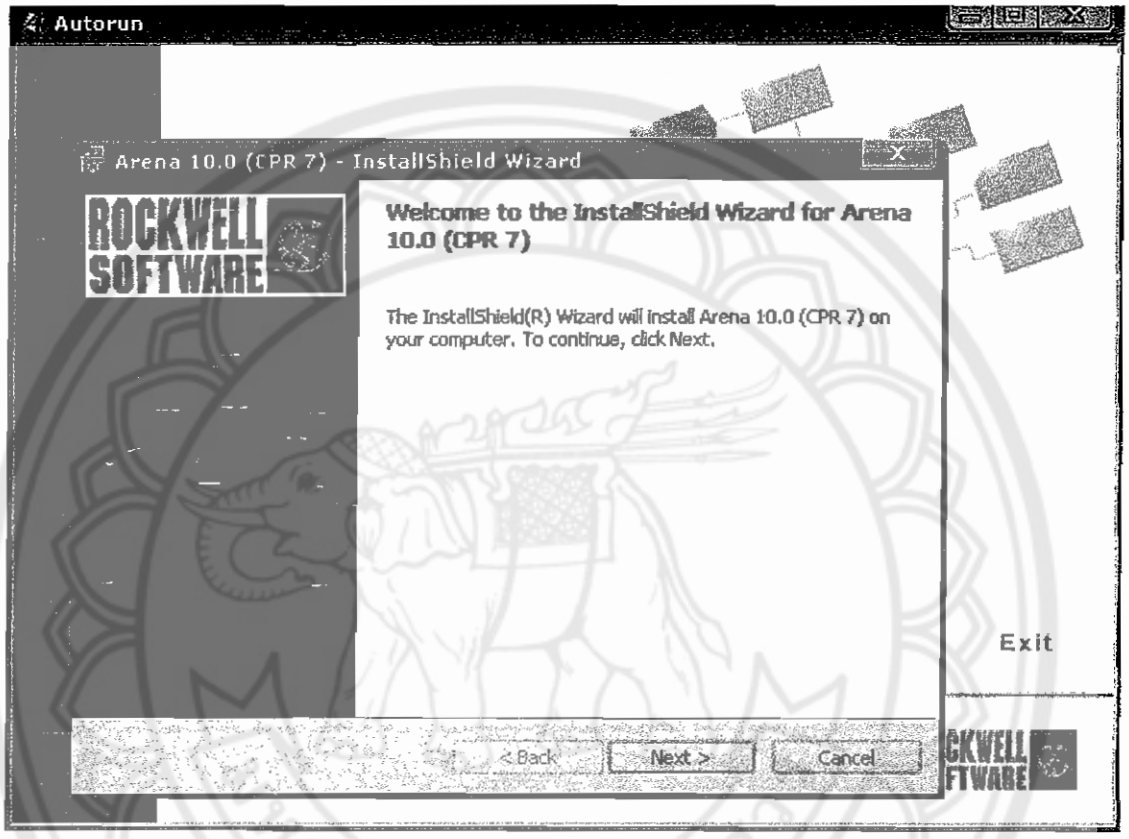
รูปที่ ก.1 แสดงหน้าจอการติดตั้งโปรแกรม Arena V.10

1.2 คลิกที่เมนู Install Arena จะปรากฏหน้าจอต้อนรับการติดตั้งโปรแกรม Arena V.10



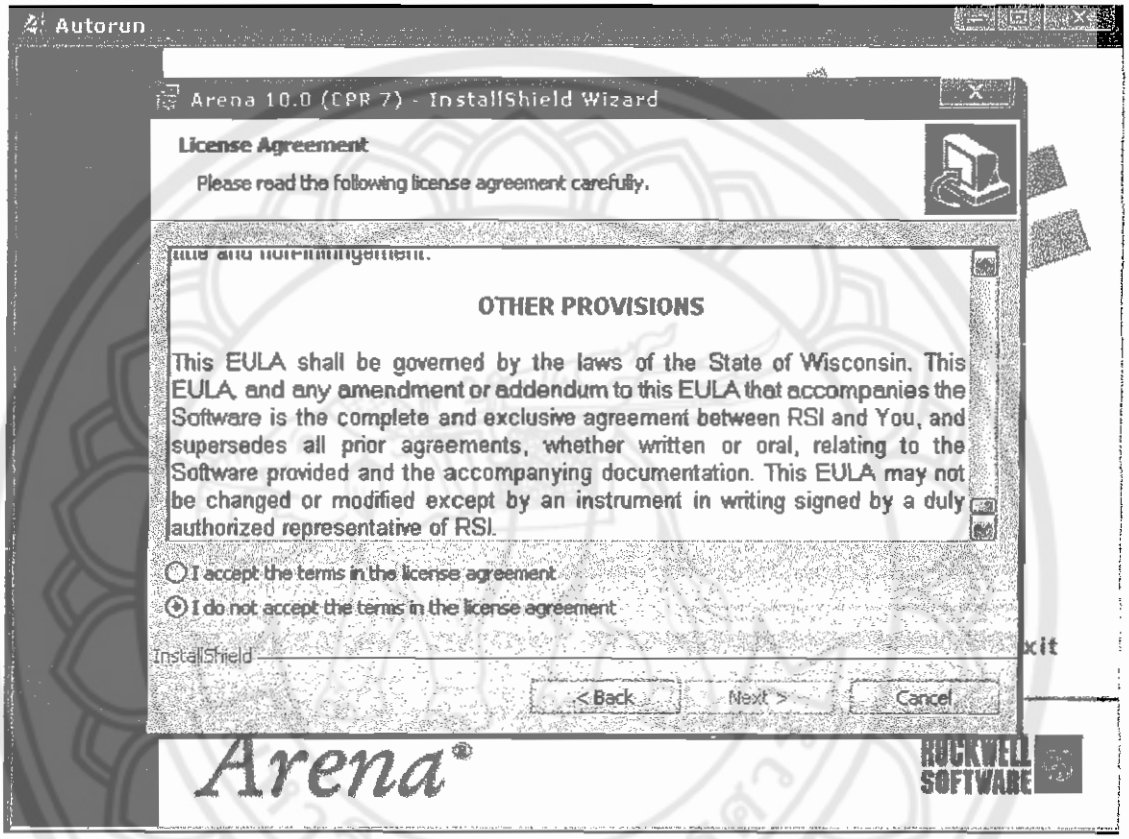
รูปที่ ก.2 แสดงหน้าจอต้อนรับการติดตั้งโปรแกรม Arena V.10

1.3 คลิกที่ปุ่ม Next> จะปรากฏหน้าจอของการยอมรับในเงื่อนไขต่างๆของโปรแกรม Arena V.10



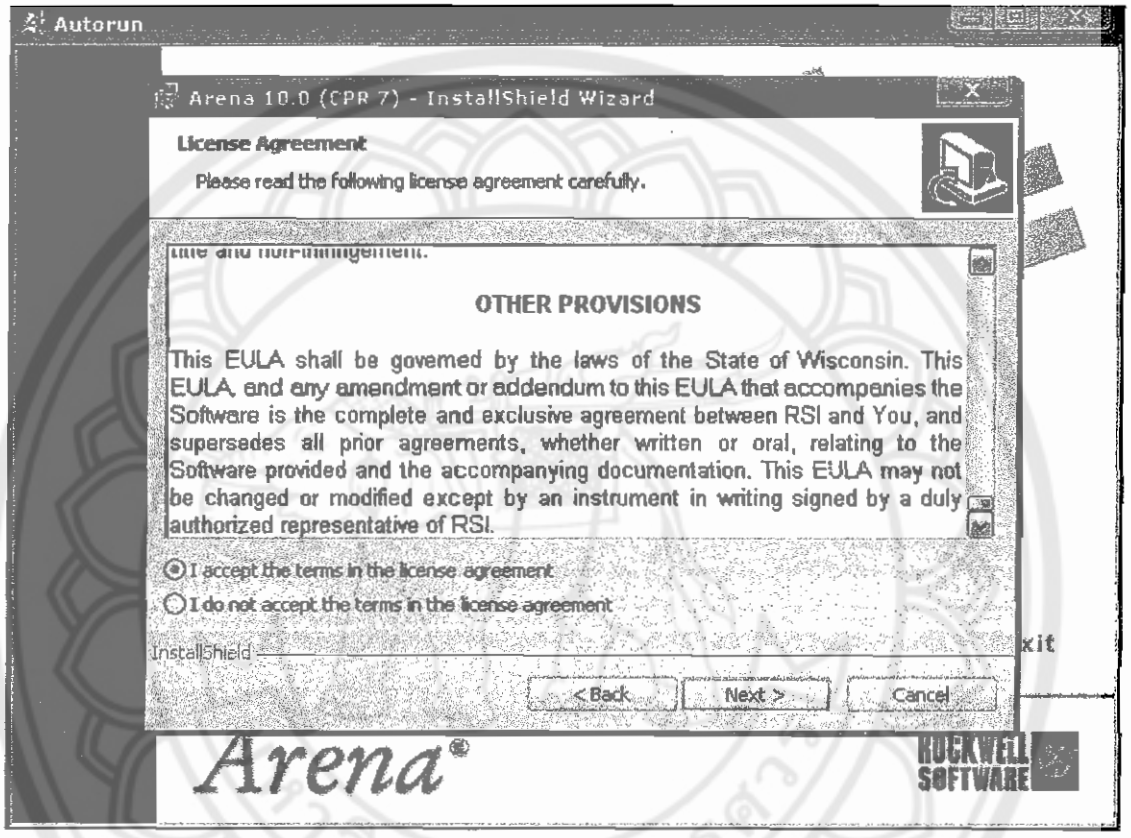
รูปที่ ก.3 แสดงหน้าจอของการยอมรับในเงื่อนไขต่างๆของโปรแกรม Arena V.10

1.4 คลิกที่ปุ่ม I accept the terms in the license agreement เพื่อยอมรับเงื่อนไขในการติดตั้งโปรแกรม Arena V.10 และกดปุ่ม next จากนั้นจะปรากฏหน้าจอให้ใส่ข้อมูลของผู้ใช้งาน โดยที่ในช่อง Serial Number ให้ใส่ Student ดังรูป



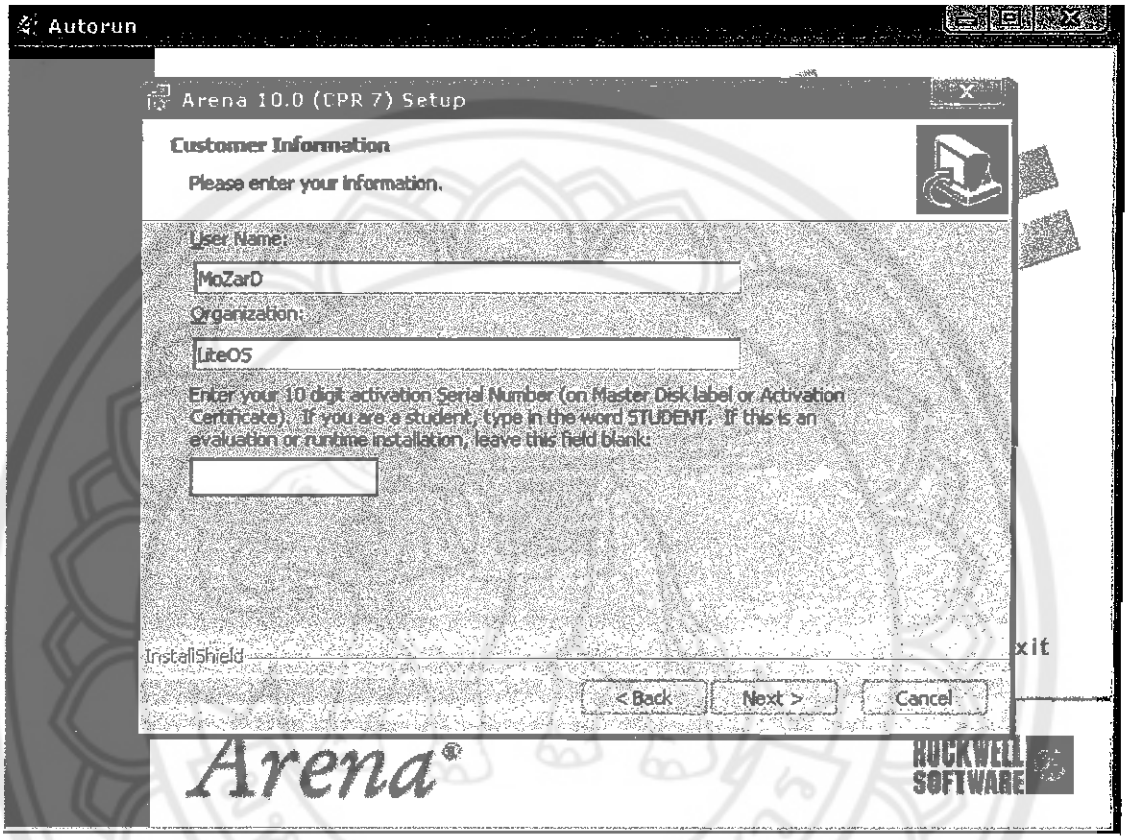
รูปที่ ก.4 แสดงหน้าจอของการใส่ข้อมูลต่างๆ

1.5 เมื่อทำการกรอกข้อมูลต่างๆเสร็จให้ทำการคลิกที่ปุ่ม Next เพื่อเข้าสู่หน้าจอของการกำหนดตำแหน่งของโปรแกรมที่จะติดตั้งลงในเครื่องคอมพิวเตอร์ถ้าไม่เปลี่ยนให้คลิกที่ปุ่ม Next จะปรากฏหน้าจอดังรูป



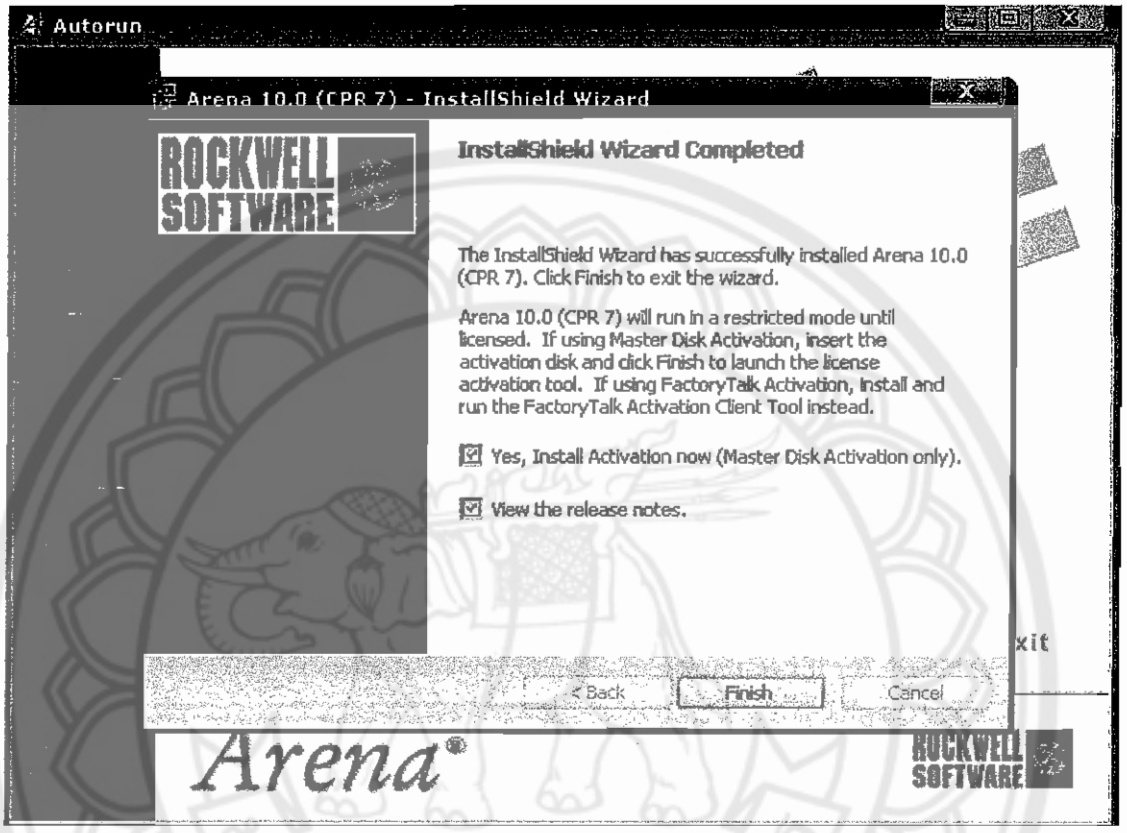
รูปที่ ก.5 แสดงตำแหน่งของโปรแกรมที่จะติดตั้งลงในคอมพิวเตอร์

1.6 จากนั้นทำการคลิกที่ปุ่ม Next 2 ครั้งจะปรากฏหน้าจอของการติดตั้งโปรแกรมจนเสร็จสมบูรณ์ (ถ้าติดตั้งแล้วเกิดข้อความ Error ให้ทำการติดตั้งใหม่อีกครั้ง)



รูปที่ ก.6 แสดงหน้าจอของการติดตั้งโปรแกรม Arena V.10

1.7 จากนั้นรอจนติดตั้งโปรแกรมจนเสร็จสมบูรณ์จากนั้นทำการกดปุ่ม Finish



รูปที่ ก.7 แสดงหน้าจอเสร็จสมบูรณ์ของการติดตั้งโปรแกรม Arena V.10

ภาคผนวก ข

เกี่ยวกับโปรแกรม Arena V.10 Simulation V.10

ภาคผนวก ข
เกี่ยวกับโปรแกรม Arena V.10 Simulation V.10

โปรแกรม Arena Simulation ตัวนี้เป็น Version 10 ซึ่งผลิตโดยบริษัท Rockwell Software (Rockwell software Inc) ซึ่งถูกต้องตามลิขสิทธิ์(Copyright C2000) สำหรับการศึกษา โดยใช้พื้นฐานในด้าน Industrial Engineering(Basic IE)ซึ่งสามารถหาข้อมูลเพิ่มเติมได้จาก www.arenasimulation.com/support www.software.rockwell..com

Arena 10 เป็นโปรแกรมที่ใช้แก้ปัญหาการจำลองการตัดสินใจ เป็นโปรแกรมที่สร้างและให้ทดลองใช้งานบนแบบจำลองระบบที่เราสร้างขึ้นบนคอมพิวเตอร์ แทนการใช้งานจริงซึ่งจะทำให้เรารู้ถึงอนาคต และสามารถนำไปปรับปรุงแก้ไขระบบให้ดียิ่งขึ้น โดยมีขั้นตอนการปฏิบัติ 5 ขั้นตอน ดังนี้

สร้างแบบตัวอย่าง โดยการจำลองระบบงานอย่างคร่าว

1 การเก็บข้อมูลของระบบนั้น เช่น ความต้องการทรัพยากร รายละเอียดกระบวนการ

2 เลียนแบบตัวอย่าง เป็นการเริ่มใช้งานแบบจำลองภายใต้เงื่อนไขที่กำหนด เพื่อพิสูจน์ว่าแบบจำลองที่เราสร้างขึ้นนั้นถูกต้อง เหมาะสมและใช้งานได้กับระบบตามความเป็นจริง

3 วิเคราะห์ผลลัพธ์แบบจำลอง ศึกษารายงานของแบบจำลอง

4 ปรับปรุงแก้ไขหาทางเลือกที่ดีที่สุด หรือเปลี่ยนแปลงแบบจำลองเพื่อให้ได้สิ่งที่เราต้องการที่ดีที่สุด

1 ส่วนที่สำคัญในการสร้าง Simulation Model มีดังนี้

1.1 Entities

เป็นสิ่งแรกที่ต้องทำในการสร้าง Model ขึ้นมา Entities เป็นตังยิ่งบอกถึงการมีอยู่, การเคลื่อนที่ผลกระทบต่อการวัดประสิทธิภาพของ Out put ที่ออกมาเราสามารถกำหนดให้ Entities เป็นไปในรูปแบบที่ต้องการให้เหมือนกันในระบบจริงที่เราต้องการศึกษา ซึ่งใน 1 Model สามารถมี Entities ได้หลายตัวและ Entities ทุกตัวต้องล้วนมีความสัมพันธ์กัน เช่น เราศึกษาระบบที่มีคน, เวลา, และการรอเรากำหนดใน Entities คือ คน1 คนเป็นคนที่เข้ามาในระบบ เมื่อคนในระบบเพิ่มขึ้นเรื่อยๆทำให้เวลาที่ใช้ในกระบวนการเพิ่มมากขึ้นเรื่อยๆทำให้เวลาที่ใช้ในระบบเพิ่มขึ้นเรื่อยๆ

1.2 Attributes

เป็นการอ้างเหตุผลให้ Entities เป็นการกำหนดชื่อ, จำนวนให้แก่Entities ตามความเหมาะสมเพื่อการเรียกใช้ Entities ที่กำหนดได้ถูกต้อง เราสามารถกำหนดคุณลักษณะให้กับ Entities ให้แตกต่างกันใน Arena V.10 นี้กำหนด Attributes ไว้โดยอัตโนมัติแล้วแต่เราสามารถเปลี่ยนแปลงแก้ไข กำหนดให้เป็นในแบบที่เราต้องการได้

1.3 Variables

เป็นตัวแปรต่างๆในการกำหนดค่าตัวแปรลงใน Model ยิ่งมีEntities มากเท่าไหร่ยิ่งมีตัวแปรในระบบมากเท่านั้น แต่ตัวแปรต่างๆต้องมีความแตกต่างกันและไม่ซ้ำแบบกัน (Unique)

1.4 Resources

เป็นการกำหนดทรัพยากรต่างๆให้แก่ระบบที่เราต้องการศึกษามาว่าจะเป็นคน, เครื่องจักร, เอกสาร เป็นต้น ในกระบวนการเราสามารถกำหนดให้ Entities สามารถใช้งานร่วมกับทรัพยากรที่เรากำหนดขึ้นมาในระบบได้

1.5 Queues

แถวคอยจะเกิดขึ้นเมื่อ Entities ต่างๆใช้ทรัพยากรอยู่ทำให้ Entities ต่อๆมาไม่สามารถเคลื่อนที่ต่อไปได้จึงต้องมีพื้นที่สำหรับการรอคอย

1.6 Statistical Accumulators

เป็นการคำนวณสถิติสะสม ไม่ว่าจะเป็นระบบจริงหรือระบบจำลองระบบ จะต้องมีการเกิดการสะสมเกิดขึ้น ในระหว่างที่มีแถวค่อย ในกระบวนการไม่ว่าจะเป็น เวลาในกระบวนการสะสม, เวลาคอยสะสม, จำนวนคอยสะสม เป็นต้น

1.7 Events

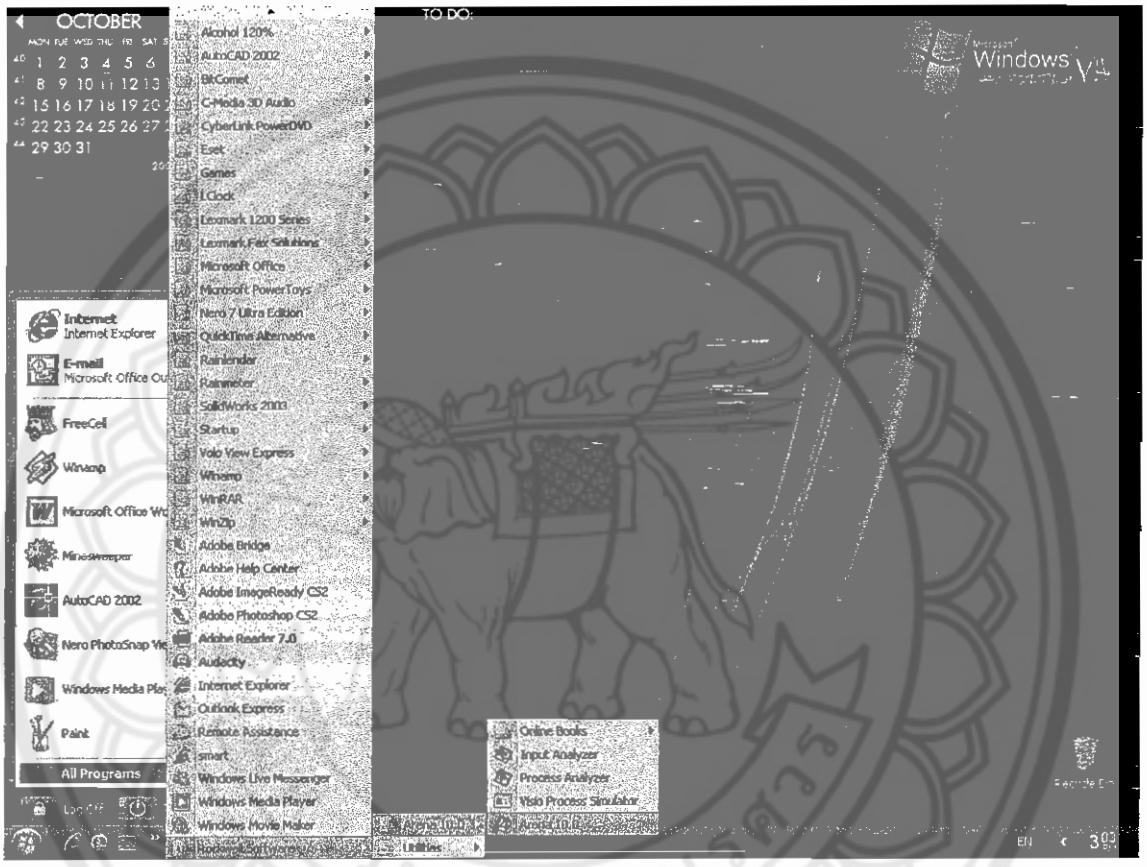
เหตุการณ์ต่างๆสามารถกำหนดได้เมื่อเรามีการประมวลผล จะเห็นได้ว่าเรากำหนดเหตุการณ์เป็นไปตามที่ต้องการหรือไม่

1.8 Simulation Clock

เป็นเวลาจำลองที่ใช้ในการประมวลผลแบบจำลองที่เราสร้างขึ้น เป็นเวลาเสมือนกับเวลาจากระบบจริงแต่เราสามารถประมวลแบบจำลองได้ในเวลาไม่นาน คือ เร็วกว่าระบบจริง

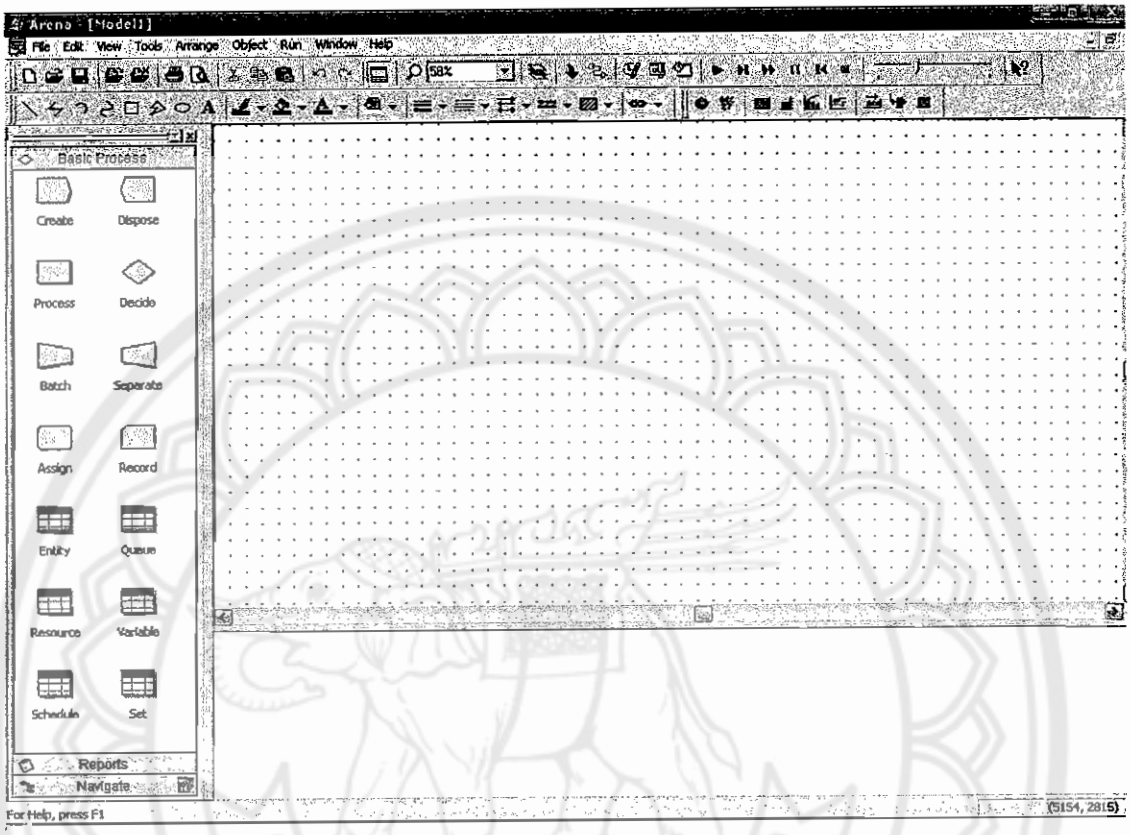
2 การใช้งานโปรแกรม Arena V.10 เบื้องต้น

2.1 ทำการเปิดโปรแกรม Arena V.10 โดยเลือกที่ All Programs>Rockwell Software> Arena > Arena V.10 จะปรากฏหน้าจอของโปรแกรมดังรูป



รูปที่ ข.1 แสดงการเรียกใช้งานโปรแกรม Arena V.10

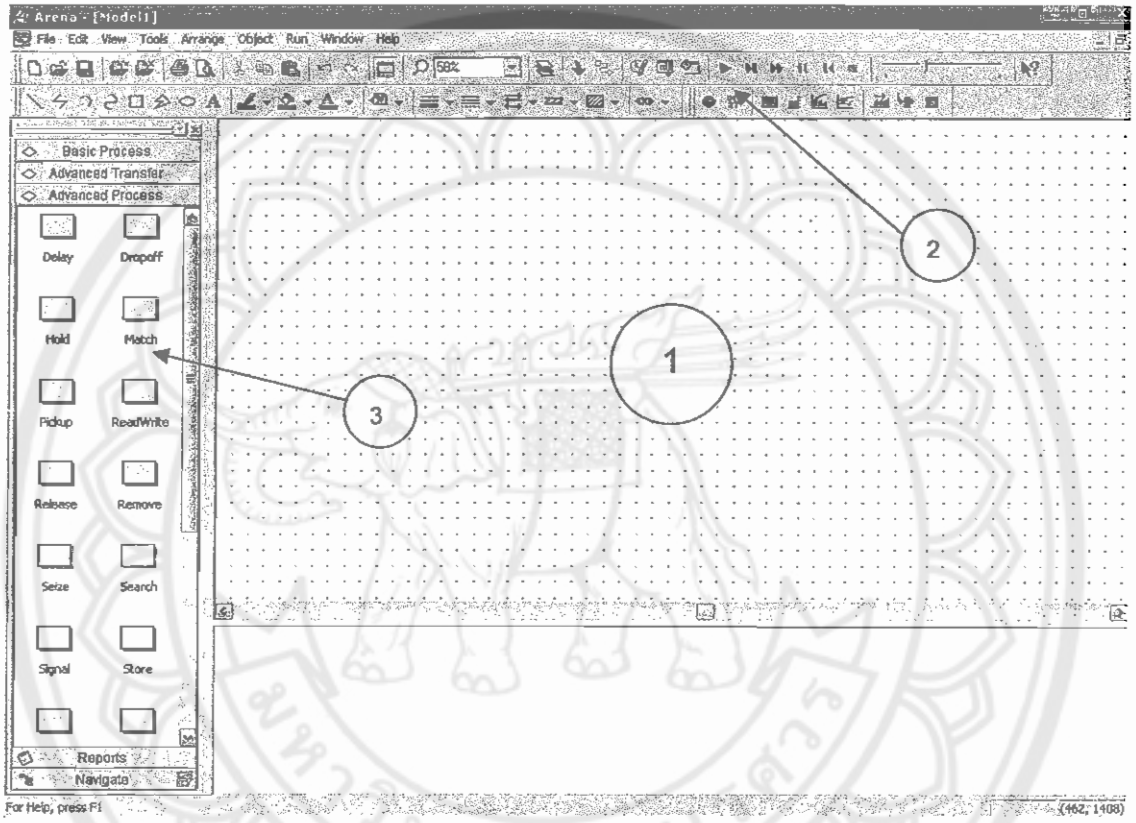
2.2 ทำการเปิดโปรแกรม Arena V.10 จะเกิดหน้าต่างของ Arena สามารถใช้งานได้เลย



รูปที่ ข.2 แสดงหน้าตาของโปรแกรม Arena V.10

2.3 **ส่วนประกอบต่างๆของโปรแกรม** Arena V.10

เมื่อเปิดโปรแกรม Arena V.10 **ครั้งแรก โปรแกรมจะปรากฏหน้าจอซึ่งมีส่วนประกอบหลักต่อไปนี้**



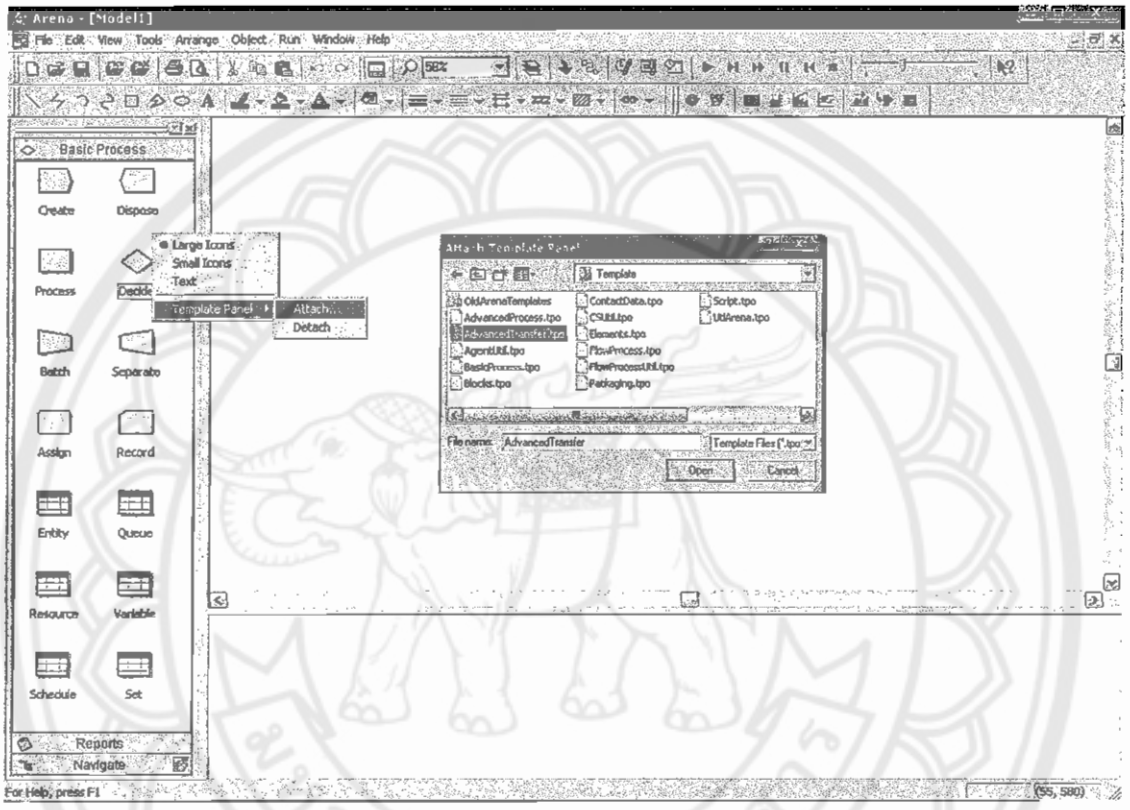
รูปที่ ข.3 ส่วนประกอบต่างๆของโปรแกรม Arena V.10

หมายเลข 1 หมายถึง พื้นที่ในการทำงานของโปรแกรม Arena

หมายเลข 2 หมายถึง ทูลบาร์ หรือ เครื่องมือช่วยในการทำงาน

หมายเลข 3 หมายถึง โมดูล หรือ โมเดลที่นำมาสร้างแบบจำลองบนโปรแกรม Arena

2.4 การเลือกโมดูลใน Arena V.10 มาใช้ในการสร้างแบบจำลอง คลิกขวาด้านพื้นที่ของโมดูล เลื่อนเมาส์ ไปที่ Template Panel > Attach แล้วเลือกโมดูลที่ต้องการ โมดูลจะปรากฏด้านซ้ายมือบนหน้าต่าง Arena ดังรูป



รูปที่ ข.4 การเลือกใช้โมดูล

2.5 ทูลบาร์ (Toolbar)

พิจารณาจากหน้าจอ จะเห็นปุ่มต่างๆที่วางเรียงเป็นแผงควบคุม ช่วยสามารถเรียกใช้งานคำสั่งได้อย่างรวดเร็ว เพียงแคคลิกเมาส์เท่านั้น รายละเอียดของปุ่มต่างๆมีดังนี้

รูปที่ ข.5 ทูลบาร์

2.6 Toolbar โปรแกรม Arena V.10

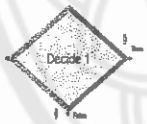
ตารางที่ ข.1 หน้าที่ของโปรแกรม Arena V.10

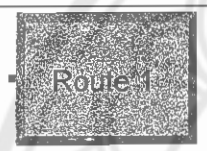
ลำดับ	คำสั่ง	หน้าที่
1	New	สร้างโมเดลไฟล์ใหม่ขึ้นมา
2	Open	เปิดโมเดลไฟล์ที่มีบันทึกอยู่แล้ว
3	Save	บันทึกโมเดลไฟล์
4	Template Attach	แสดงรายการชนิดของแฟ้มที่จะแสดง
5	Template Detach	ปิดรายการของแฟ้ม
6	Print	พิมพ์เอกสาร
7	Print Preview	แสดงตัวอย่าง
8	Cut	ตัดข้อมูล
9	Copy	คัดลอกข้อมูล
10	Past	วางข้อมูล
11	Undo	ย้อนกลับ
12	Redo	ไปข้างหน้า
13	Toggle Split Screen	จัดหน้าจอให้มีแต่แบบโมเดลไม่แสดงรายละเอียด
14	View Region	ขยาย
15	Percent	ค่าของการขยายหน้าจอ
16	Layers	แสดงชั้นคำสั่ง
17	Connect	การเชื่อมต่อระหว่างโมดุล
18	Sub model	ปุ่มเรียก Sub model
19	Go	สั่งให้โปรแกรมทำการประมวลผล
20	Step	สั่งให้โปรแกรมทำการประมวลรวมเร็วจนเสร็จ
21	Fast-Forward	สั่งให้โปรแกรมทำการประมวลอย่างรวดเร็วจนเสร็จ
22	Pause	การหยุดการทำงานชั่วคราว
23	Start Over	ให้โปรแกรมพร้อมทำการประมวลผลต่อไป
24	Stop	การหยุดการทำงาน
25	Help	คำสั่งช่วยเหลือ

26	Storage	กำหนดพื้นที่การจัดเก็บ
27	Seize	ขนาดพื้นที่
28	Parking	พื้นที่หยุดของภาพเคลื่อนไหว
29	Transporter	ภาพที่ต้องการให้เคลื่อนที่การขนส่ง
30	Station	สถานีงานที่กำหนด
31	Intersection	ทางแยก จุดตัด
32	Route	กำหนดเส้นทางการเคลื่อนที่ให้ภาพ
33	Segment	ส่วนของเซ็กเมนต์
34	Distance	ระยะทางของการเคลื่อนที่ Transporter
35	Network	เส้นทางการเคลื่อนที่ของ Intersect
36	Promote path	เส้นทางการส่งเสริมการเดินทาง
37	Clock	ใช้สร้างนาฬิกา
38	Date	วันที่
39	Variable	ตัวแปร
40	Level	แสดงการวัดระดับ
41	Histogram	กราฟแสดงผลแบบ Histogram
42	Plot	ให้โปรแกรมทำการ Plot กราฟ
43	Queue	กำหนดเส้นทางการรอคอย
44	Resource	ทรัพยากรที่มีอยู่ เช่น เครื่องจักร, พนักงาน เป็นต้น
45	Global	รูปภาพทั่วๆไป
46	Line	เส้นตรง
47	Poly line	เส้นตรงที่กำหนดจุดได้
48	Are	เส้นวงกลม
49	Bezier Curve	เส้นโค้ง
50	Box	กล่องสี่เหลี่ยม
51	Polygon	เส้นหลายเหลี่ยม
52	Cycle	การสร้างวงกลม
53	Text	การสร้างตัวอักษร

2.7 ตารางสัญญาลักษณ์คำสั่งแบบตัวอย่าง Model

ตารางที่ ข.2 ตารางสัญญาลักษณ์คำสั่งแบบตัวอย่าง Model

สัญญาลักษณ์ Icon	ชื่อคำสั่ง	หน้าที่
	Create	เป็นModelเริ่มต้นการสร้างแบบจำลอง สร้าง Entitiesให้เข้ามาในระบบเป็นทางเข้าให้กับEntities
	Dispose	เป็น Model สิ้นสุดกระบวนการเป็นทางออกให้กับ Entities ออกจากระบบ
	Process	เป็น Model ที่กำหนดทรัพยากรให้แก่ระบบ เพื่อให้ Entities เข้ามาใช้ ทรัพยากรในระบบ
	Decide	เป็น Model ที่กำหนดการตัดสินใจ ในกรณีที่มีทางเลือกแก่ Entities
	Record	เป็น Model ที่ทำการบันทึกผลต่างๆ เพื่อแสดงในรายงาน (Report)
	Enter	กำหนด Entities เข้าสู่สถานีงานต่างๆ ในเวลาต่างๆ ตามที่ต้องกำหนด

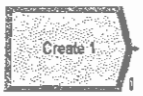
	Leave	กำหนด Entities ออกจากสถานีงานในเวลาต่างๆ
	Station	เป็นสถานที่รับ Entities เข้ามาเพื่อเข้าสู่กระบวนการในระบบ
	Route	กำหนดเวลาในการเดินทางจากสถานีหนึ่งไปอีกสถานีหนึ่ง และเส้นทางการเดินทางของ Entities
	Pick Station	เป็นการเลือกสถานีให้ Entities โดยเจาะให้เข้าสู่สถานีโดยตรง และต้องรอแถวคอยในสถานีนั้นๆก่อน
	Entity	เป็นตัวยิ่งบอกถึงการมีอยู่ การเคลื่อนที่ , ผลกระทบต่อการวัดประสิทธิภาพของ Output ที่ออกมา
	Queue	แถวคอยจะเกิดขึ้นเมื่อ Entities ต่างๆ ใช้ทรัพยากรอยู่ ทำให้ Entities ต่อๆ มาไม่สามรถเคลื่อนที่ต่อไปได้จึงต้องมีพื้นที่สำหรับการรอคอย
	Resource	เป็นการกำหนดทรัพยากรต่างๆ ให้แก่ระบบ เช่น คน , เครื่องจักร และเอกสาร เป็นต้น

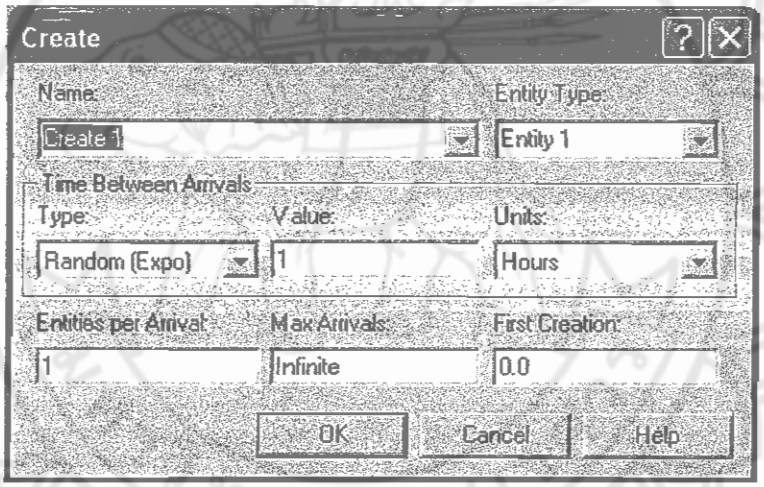
ภาคผนวก ค
การเขียนโมเดล

ภาคผนวก ค
การเขียนโมเดล

1 ตัวอย่างการเขียนโมเดล

1.1 โมดูลมีการเข้ามาของ Entities เข้ามาใช้ทรัพยากร (Resource) ในระบบและออกแบบในเวลาที่กำหนด ถือเป็น 1 รองจากการทำงานเริ่มต้นจากเลือก โมดูล Create แล้วป้อนค่าต่างๆที่ต้องการลงไป ดังรูป

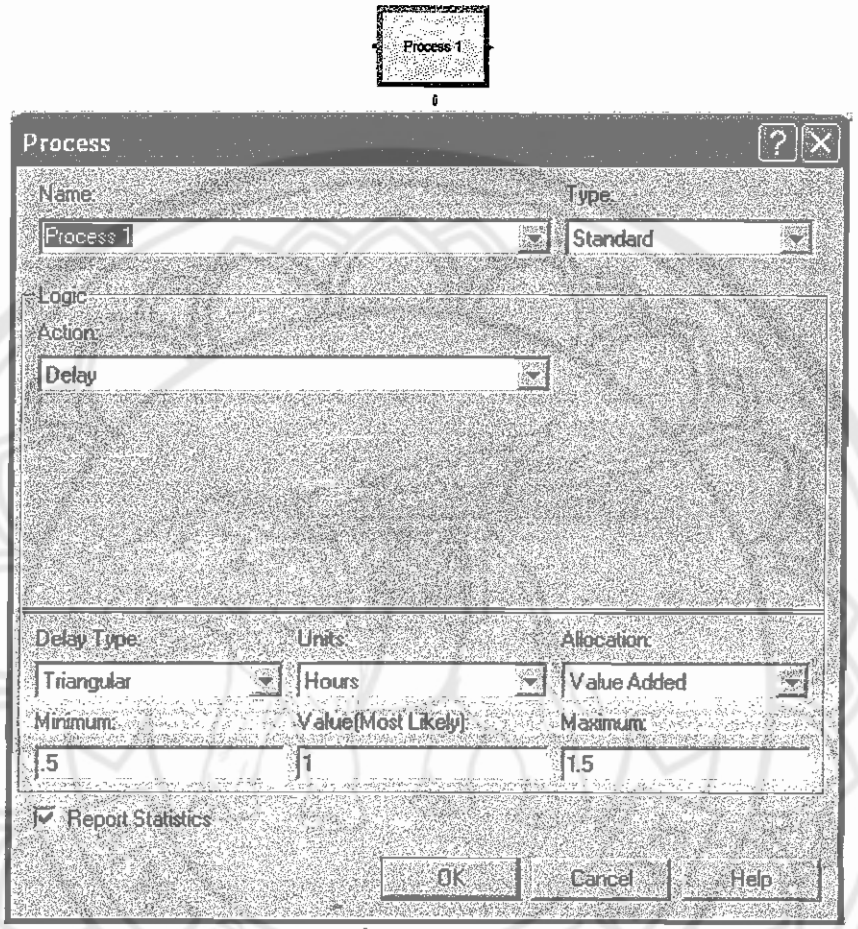




รูปที่ ค.1 Create

Name	Part Arrives to System
Entity	Part
Time Between Arrivals area	
Type	Random (Expo)
Value	5
Units	minutes

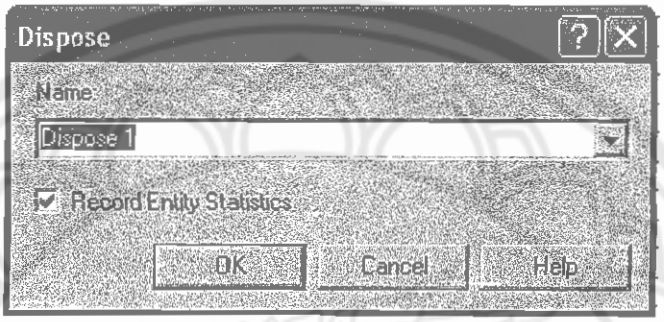
1.2 เลือกโมดูลพร้อม ทั้งป้อนค่าลงไป กำหนดทรัพยากร (Resource) ให้แก่ระบบ ดังรูป



รูปที่ ค.2 Process

Action	Seize Delay Release
Resources (secondary dialog value Add button)	
Type	Resources
Resources Name	Drill Press
Quantity	
Delay Type	Triangular
Units	Minutes
Minimum	1
Value	3
Maximum	6

1.3 เลือกโมดูล Dispose เพื่อกำหนดค่าลงไปใน Entities ได้ออกจากระบบ ดังรูปที่



รูปที่ ค.3 Dispose

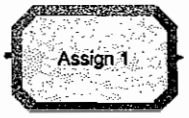
Name	Part Arrives to System

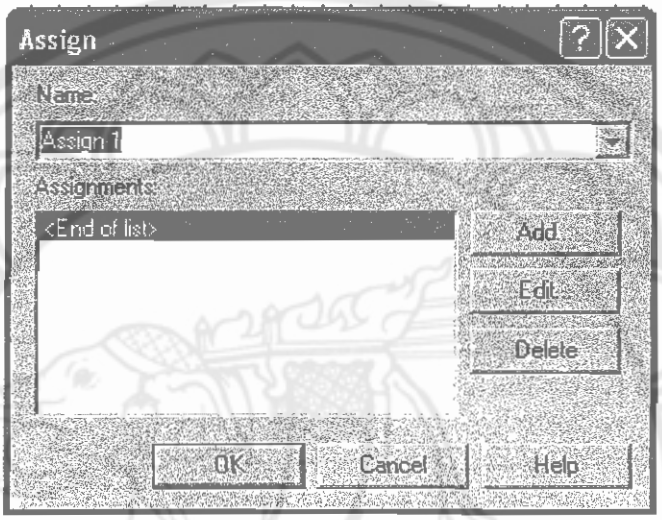
1.4 ทำการเชื่อม Module แต่ละกระบวนการเข้าด้วยกันโดยใช้ Connect ที่ Object ดังรูป่

รูปที่ ค.4 Model ตัวอย่าง

1.5 ทำการประมวลผล (Run) หรือเลือกไอคอน GO เพื่อทำการประมวลผล เพื่อทำการประมวลผล ผลที่จะแสดงใน Report เพื่อทำการประมวลผลเพื่อที่จะได้นำ ผลไปวิเคราะห์ต่อไป

1.6 Module Assign: ใช้สำหรับกำหนดคุณลักษณะของ Entity ที่สร้างขึ้น





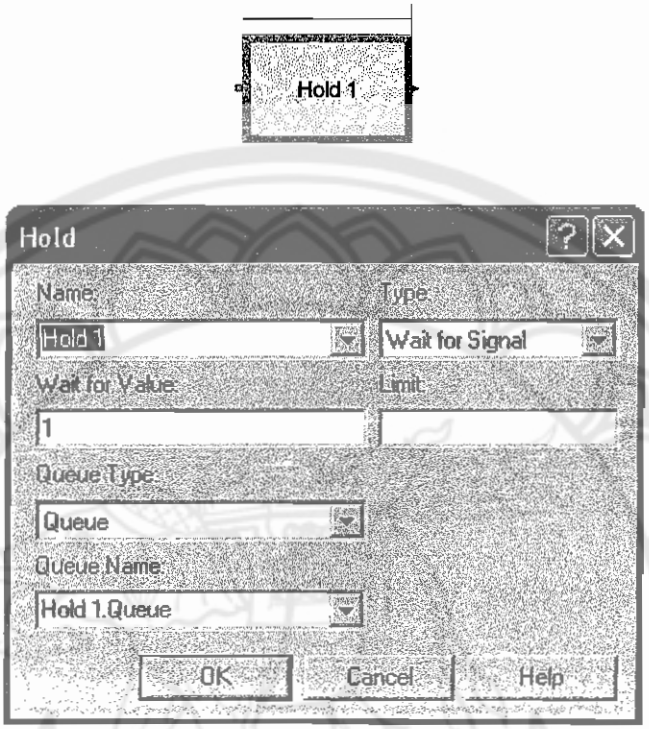
รูปที่ ค.5 แสดงการใช้งาน Module Assign

Name : ชื่อของ Module (การตั้งชื่อต้องไม่ซ้ำกัน)

Type : กำหนดรูปแบบของการตัดสินใจ

Conditions : กำหนดเงื่อนไขการตัดสินใจ

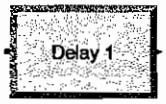
1.7 Module Hold : ใช้สำหรับกำหนดให้หยุดรอแบบมีเงื่อนไข

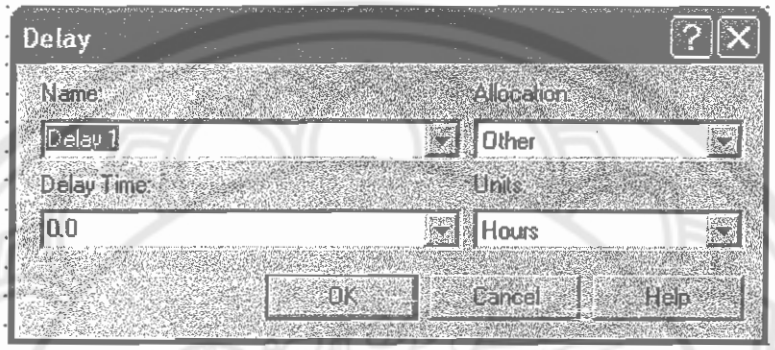


รูปที่ ค.6 แสดงการใช้งาน Module Hold

Name	: ชื่อของ Module (การตั้งชื่อต้องไม่ซ้ำกัน)
Type	: กำหนดรูปแบบของเงื่อนไขการรอ
Conditions	: กำหนดเงื่อนไขการรอ
Queue Type	: รูปแบบของการรอคอย
Queue Name	: ชื่อของ Queue (การตั้งชื่อต้องไม่ซ้ำกัน)

1.8 Module Delay: ใช้สำหรับหน่วงเวลา





รูปที่ ค.7 แสดงการใช้งาน Module Delay

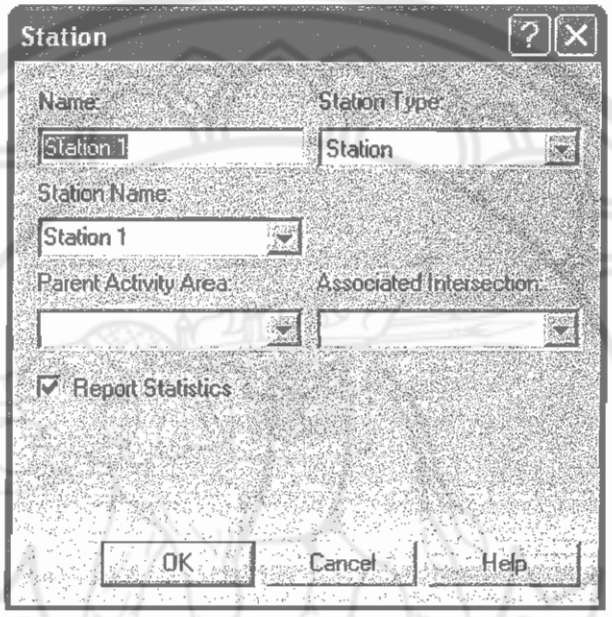
Name : ชื่อของ Module (การตั้งชื่อต้องไม่ซ้ำกัน)

Allocation : ข้อกำหนด

Delay Time : ระยะเวลาของการหน่วงเวลาการทำงาน

Unit : หน่วยของเวลา

1.9 Module Station: ใช้สำหรับกำหนดสถานีต้นงาน



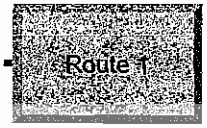
รูปที่ ค.8 แสดงการใช้งาน Module Station

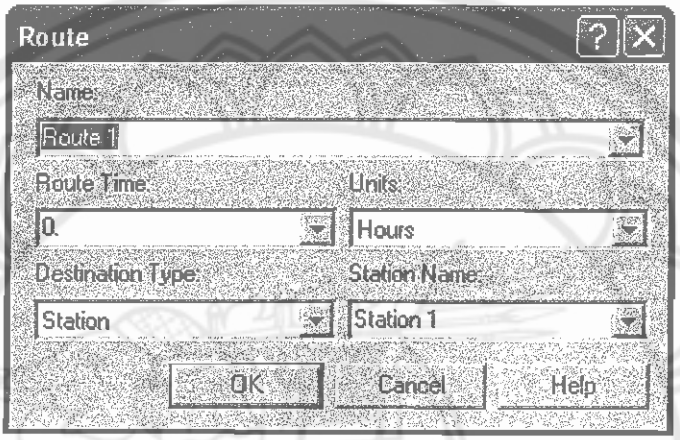
Name : ชื่อของ Module (การตั้งชื่อต้องไม่ซ้ำกัน)

Station Type : กำหนดรูปแบบของสถานีต้นงาน

Station Name : กำหนดชื่อของสถานีต้นทาง

1.10 Module Root: ใช้สำหรับการกำหนดการเคลื่อนที่





รูปที่ ค.9 แสดงการใช้งานของ Module Root

Name : ชื่อของ Module (การตั้งชื่อต้องไม่ซ้ำกัน)

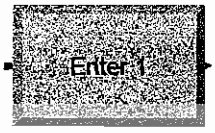
Root Time : ระยะเวลาการเคลื่อนที่

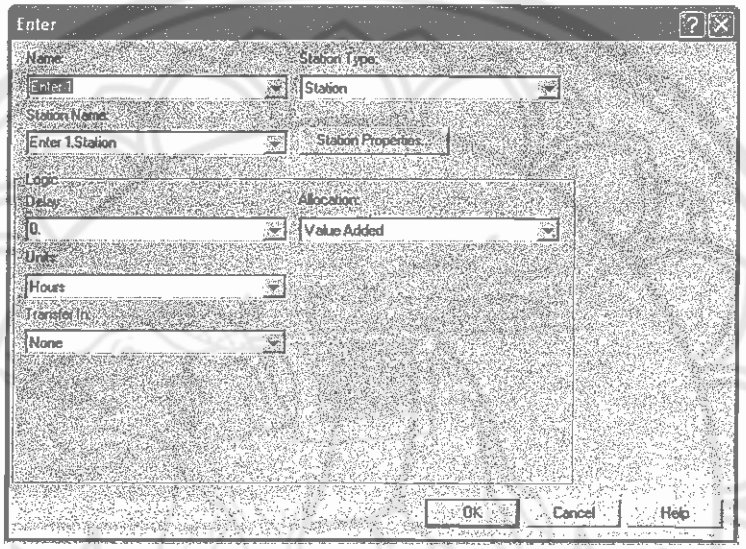
Unit : หน่วยของเวลา

Destination Type : กำหนดรูปแบบของจุดหมายปลายทาง

Station Name : กำหนดชื่อของสถานีปลายทาง

1.11 Module Enter: ใช้สำหรับกำหนดสถานีปลายทาง





รูปที่ ค.10 แสดงการใช้งาน Module Enter

Name : ชื่อของ Module (การตั้งชื่อต้องไม่ซ้ำกัน)

Station Type : กำหนดรูปแบบของสถานีงานปลายทาง

Station Name : กำหนดชื่อของสถานีปลายทาง

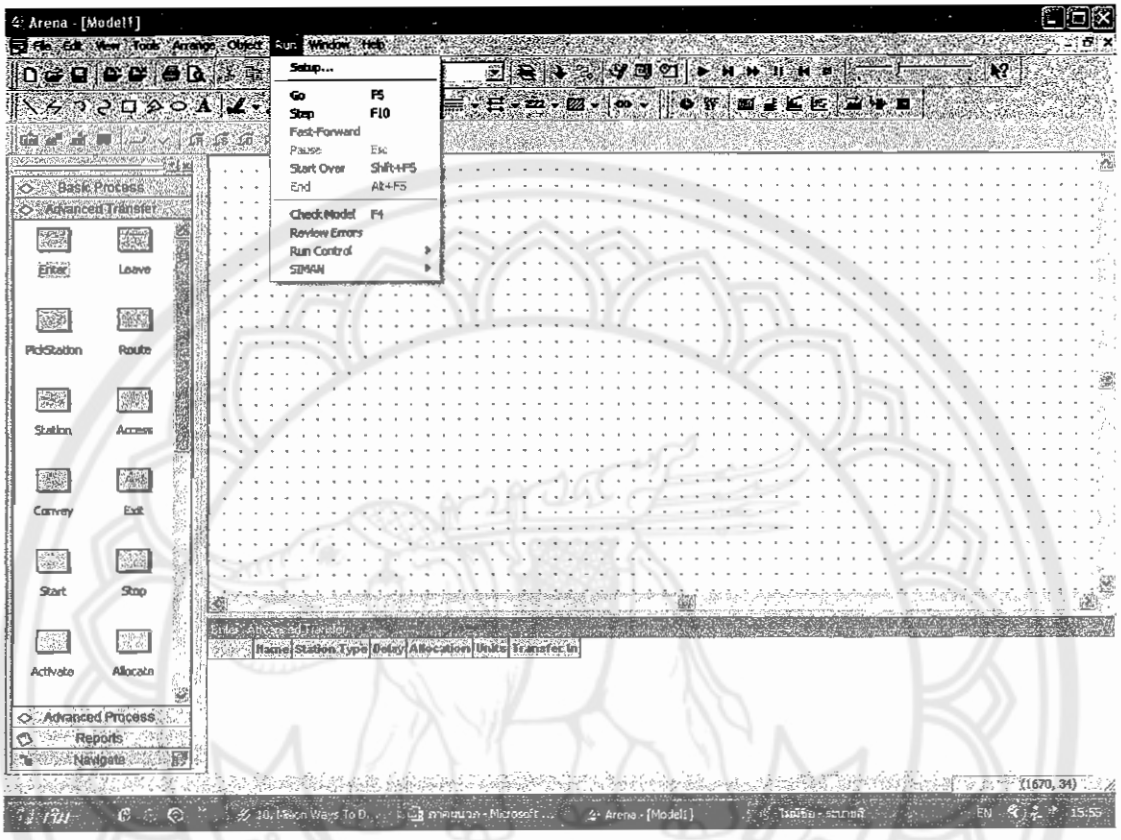
Delay : ระยะเวลาของการหน่วงเวลา

Allocation : ข้อกำหนด

Unit : หน่วยของเวลา

2 การประมวลผลโปรแกรม Arena (Run)

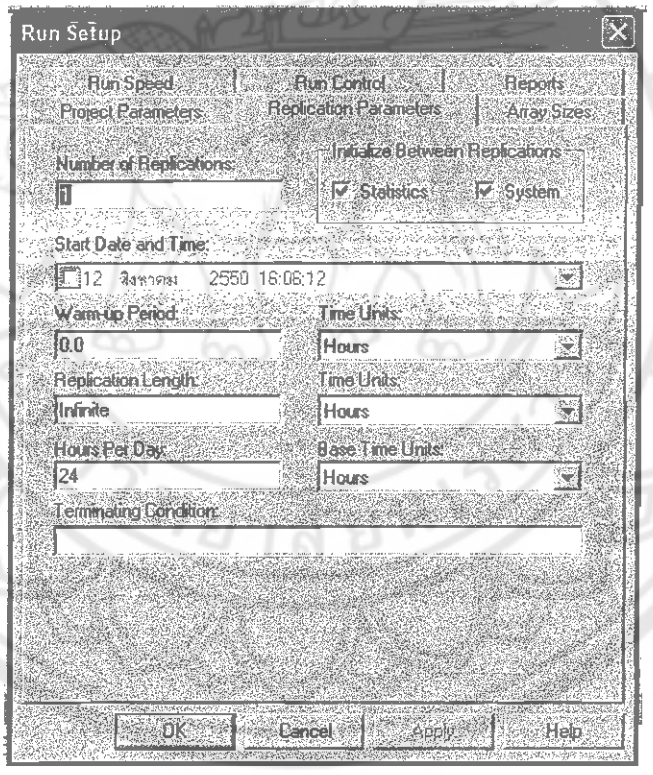
2.1 เลือกเมนู Run =>Run Setup จะปรากฏหน้าจอดังรูปที่ 3.28



รูปที่ ค.11 แสดงเลือกตั้งค่า (Run Setup)

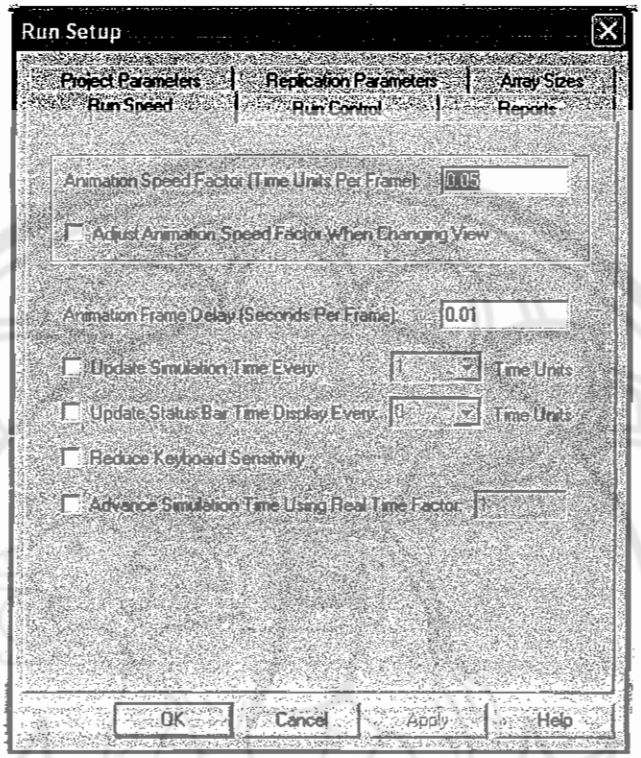
2.2 ทำการกำหนดค่าต่างๆ ก่อนทำการประมวล (Run) ซึ่งความหมายของตัวแปรที่ต้องกำหนด
มีดังนี้

Number of Replication	คือ การกำหนดรอบของ Rum
Warm-up Period	คือ ให้ทำการ Warm ก่อนทำการคำนวณค่า
Replication Length	คือ ระยะเวลาที่ทำการ Run (ได้จากการ Run หาช่วง Steady State)
Hours per Day	คือ กำหนดระยะเวลาทำงานในต่อวัน
Base Time Unit	คือ หน่วยเวลาพื้นฐานที่ต้องการ



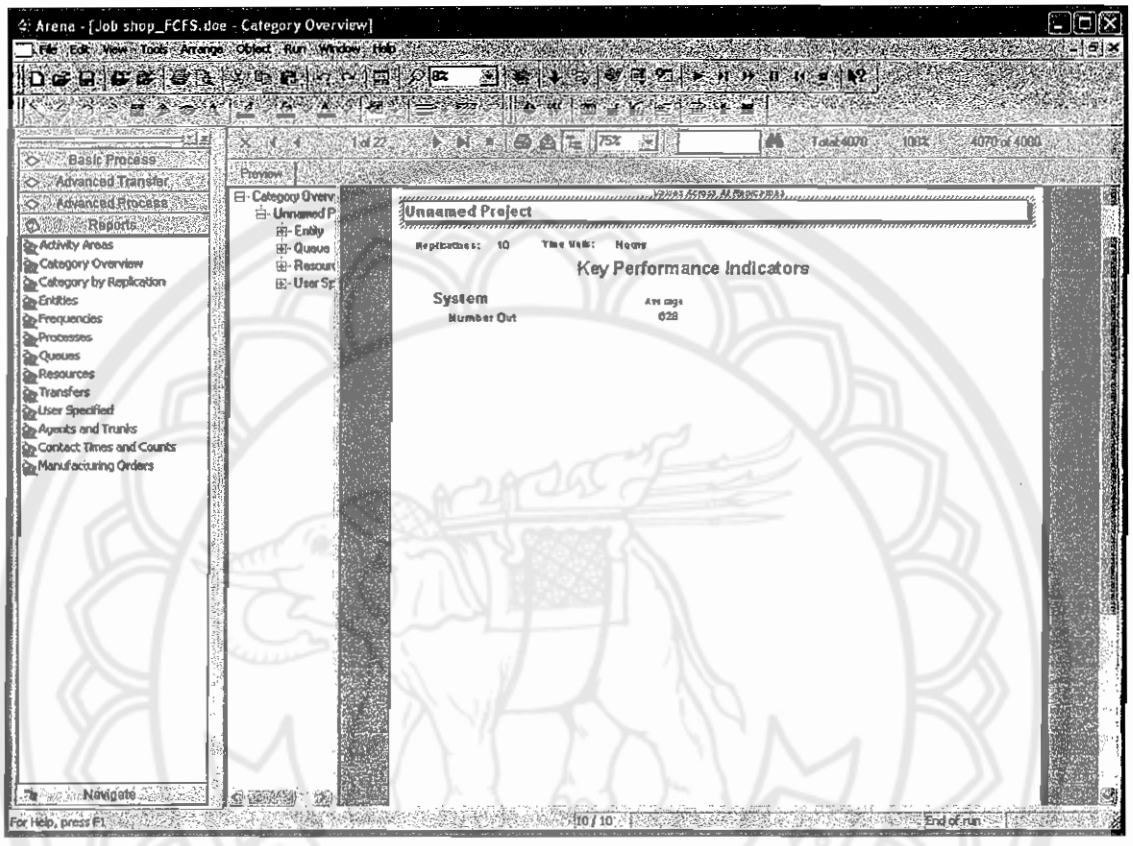
รูปที่ ค.12 แสดงการตั้งค่าในเมนูคำสั่ง Run Setup=>Replication Parameters

2.3 เลือกที่ Speed เพื่อที่จะเลือกความเร็วในการ Run



รูปที่ ค.13 การตั้ง Speed ในการ Run

2.4 ผลที่ได้จากการประมวลผลจะแสดงออกมาในรูปแบบของรายงาน (Report)



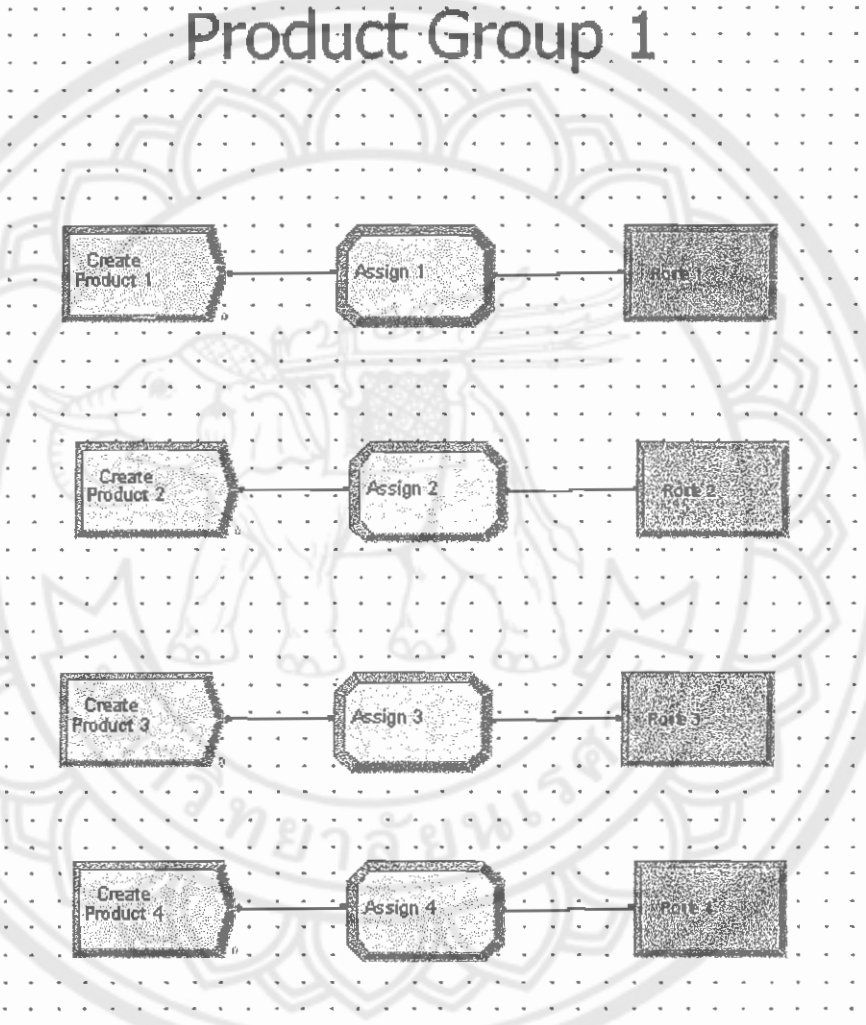
รูปที่ ค.14 แสดงรายงาน (Report) เมื่อทำการ Run เสร็จสิ้น

ภาคผนวก ง
การจำลองสถานการณ์บนโปรแกรม Arena

การจำลองสถานการณ์บนโปรแกรม Arena

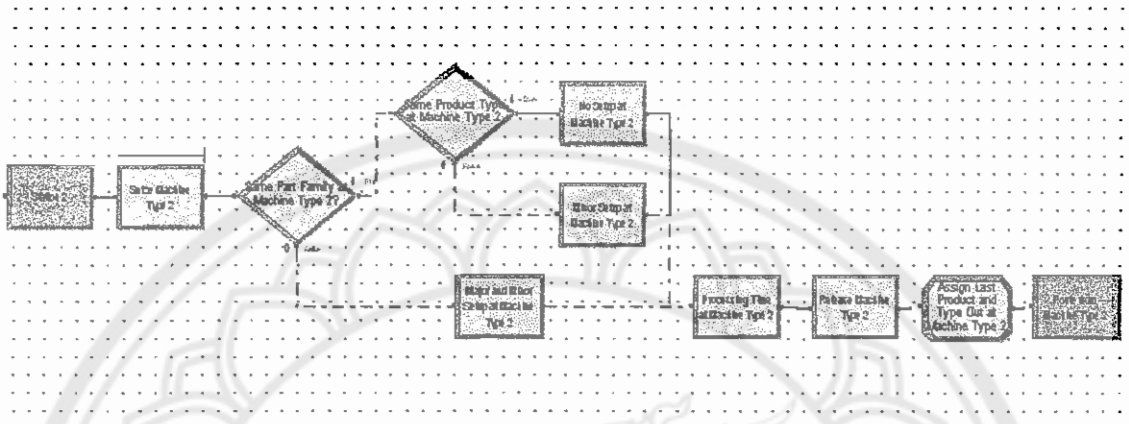
1. การจำลองสถานการณ์บนโปรแกรม Arena ของระบบการผลิตแบบ Job Shop

1.1 สร้างแบบจำลองของกลุ่มผลิตภัณฑ์ ของระบบการผลิตแบบ Job Shop



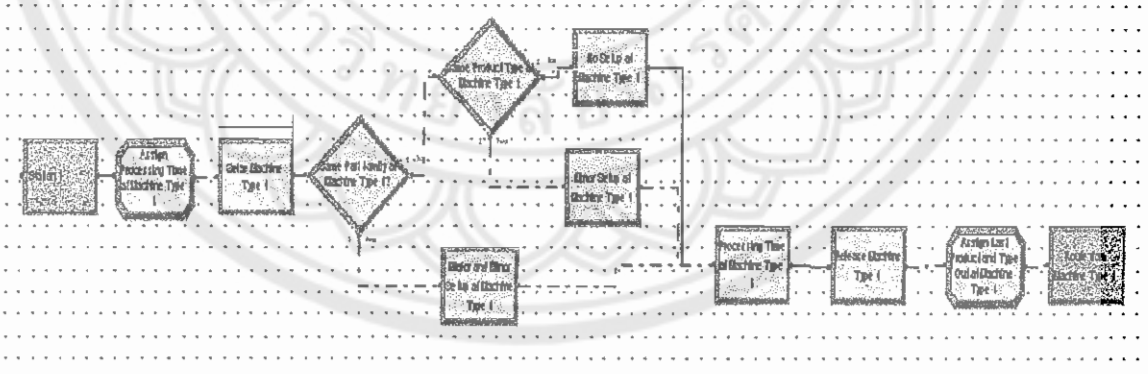
รูปที่ ง.1 แสดงการสร้างแบบจำลองของกลุ่มผลิตภัณฑ์ในระบบการผลิตแบบ Job Shop

1.2 **สร้างแบบจำลองของกลุ่มเครื่องจักรระบบการผลิตแบบ Job Shop โดยใช้หลักการกฎการจัดลำดับงานแบบ JS_FCFS**



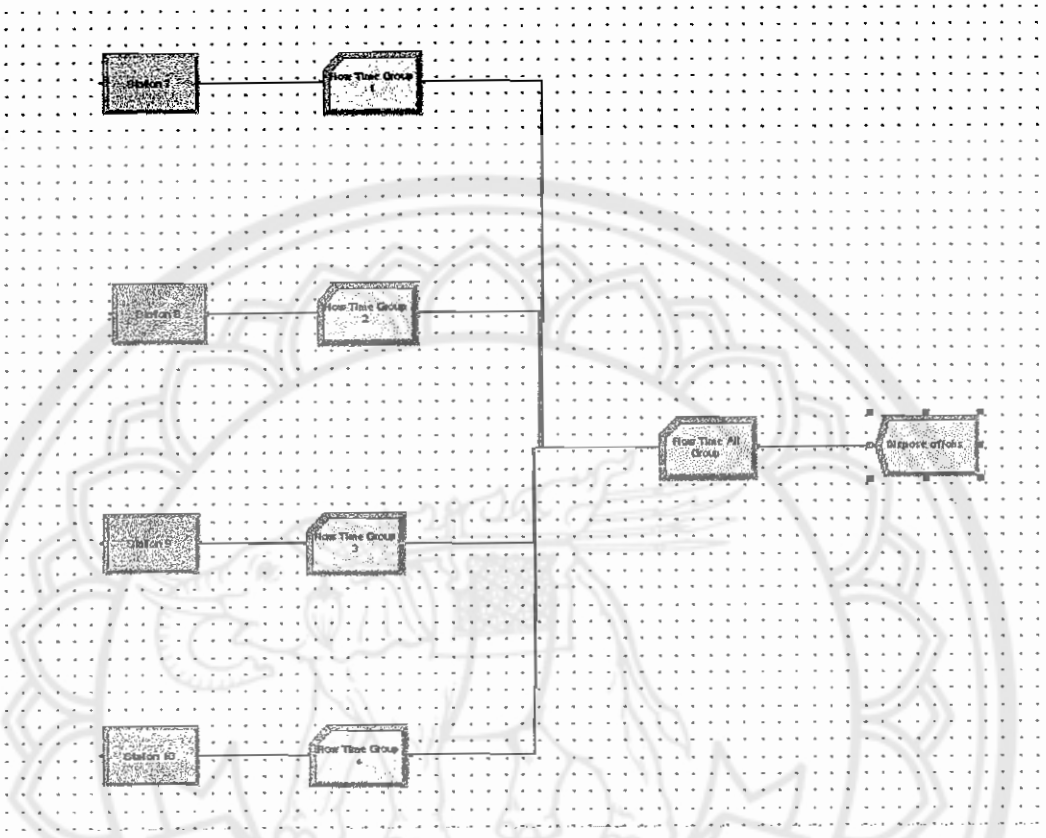
รูปที่ ง.2 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรในระบบการผลิตแบบ Job Shop โดยใช้หลักการกฎการจัดลำดับงานแบบ JS_FCFS

1.3 สร้างแบบจำลองของกลุ่มเครื่องจักรระบบการผลิตแบบ Job Shop โดยใช้หลักการกฎการจัดลำดับงานแบบ JS_SPT



รูปที่ ง.3 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรในระบบการผลิตแบบ Job Shop โดยใช้หลักการกฎการจัดลำดับงานแบบ JS_SPT

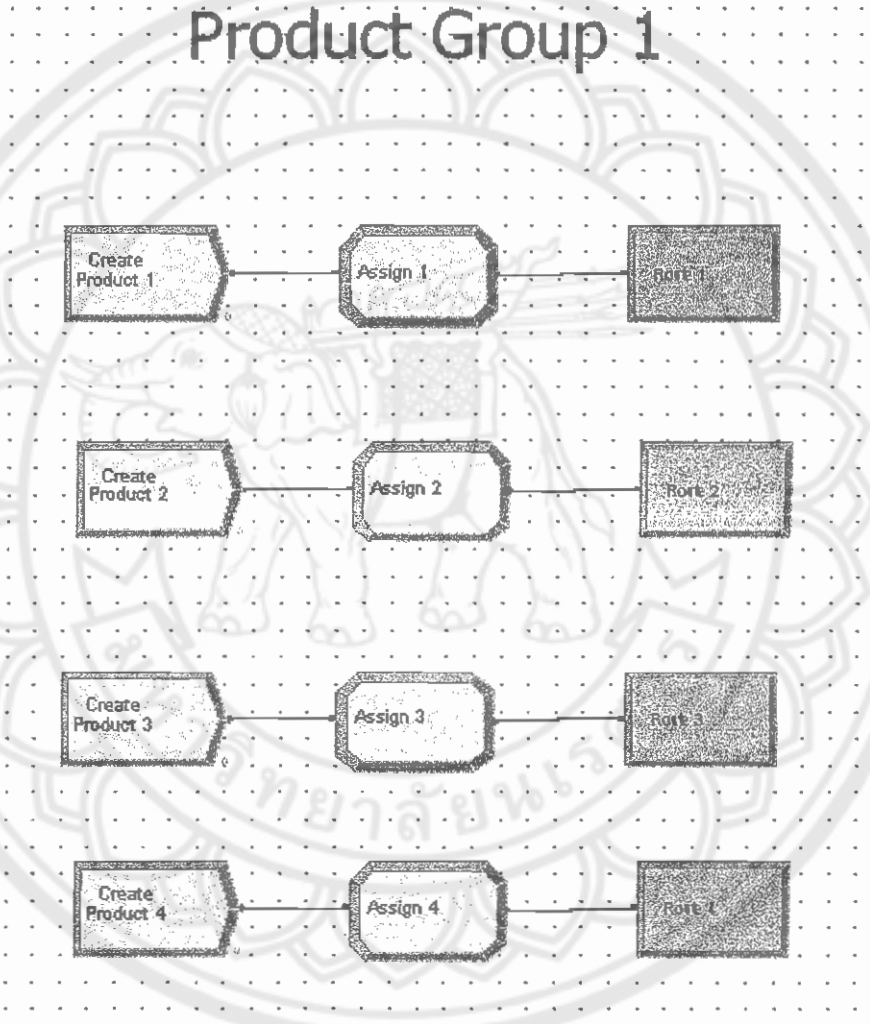
1.4 สร้างแบบจำลองเก็บข้อมูลของผลิตภัณฑ์ของระบบการผลิตแบบ Job Shop



รูปที่ ง.4 แสดงการสร้างแบบจำลองเก็บข้อมูลของผลิตภัณฑ์ในระบบการผลิตแบบ Job Shop

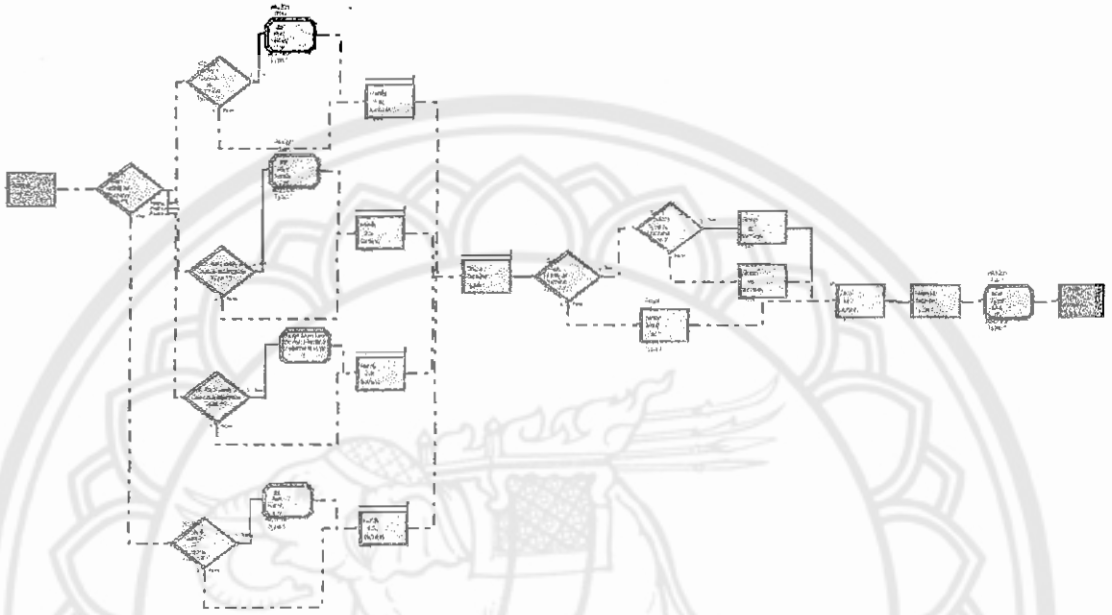
2. การจำลองสถานการณ์บนโปรแกรม Arena ของระบบการผลิตแบบ Functional Layout (FLF)

2.1 สร้างแบบจำลองของกลุ่มผลิตภัณฑ์ ของระบบการผลิตแบบ Functional Layout (FLF)



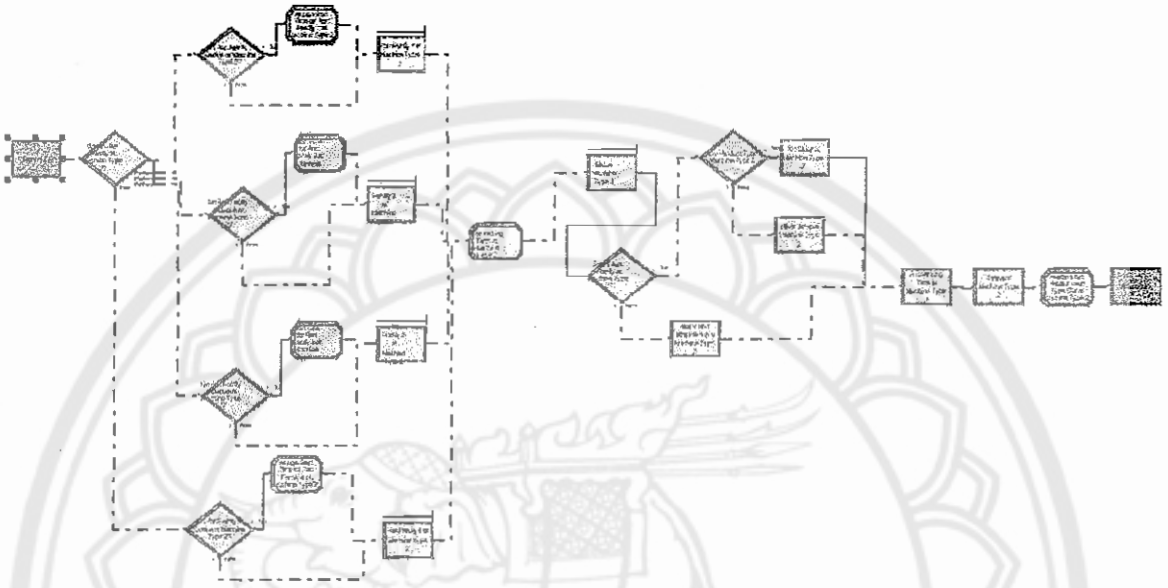
รูปที่ ง.5 แสดงการสร้างแบบจำลองของกลุ่มผลิตภัณฑ์ในระบบการผลิตแบบ Functional Layout (FLF)

2.2 สร้างแบบจำลองของกลุ่มเครื่องจักรของระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_FCFS_FCFS



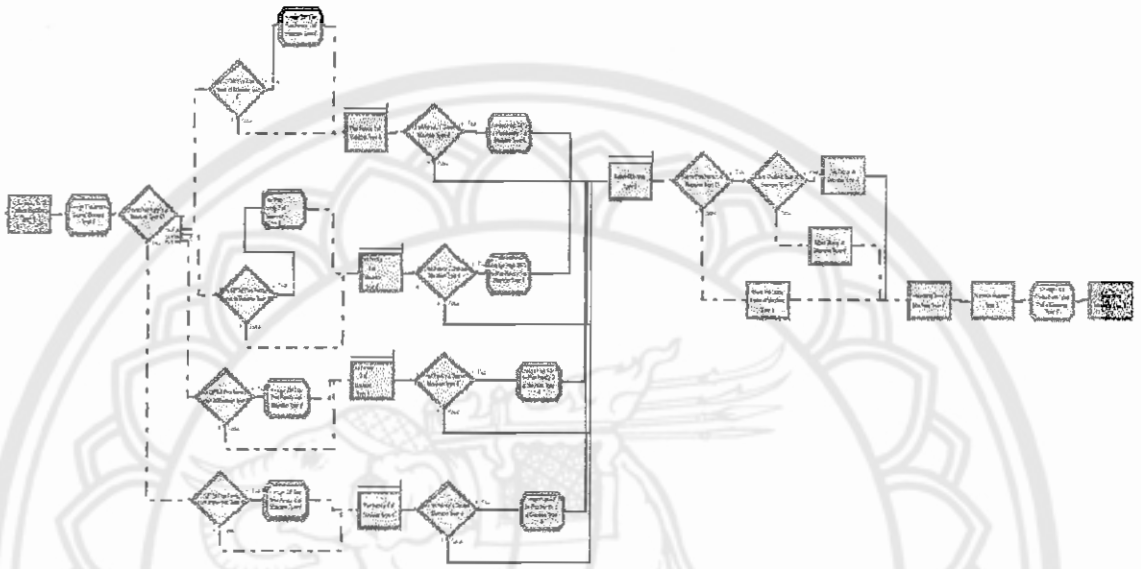
รูปที่ ง.6 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรในระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_FCFS_FCFS

2.3 **สร้างแบบจำลองของกลุ่มเครื่องจักรของระบบการผลิตแบบ** Functional Layout (FLF) **โดยใช้หลักการกฎการจัดลำดับงานแบบ** FLF_FCFS_SPT



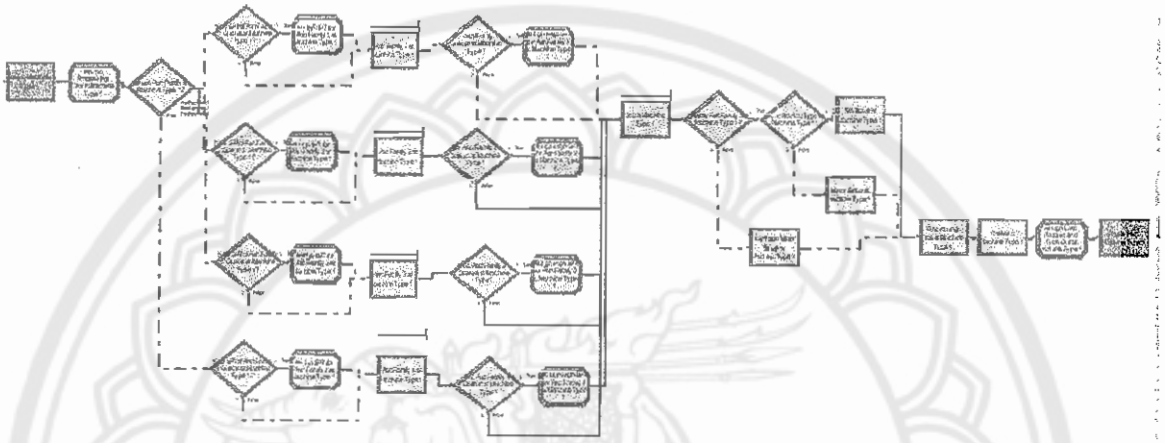
รูปที่ ง.7 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรในระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_FCFS_SPT

2.4 สร้างแบบจำลองของกลุ่มเครื่องจักรของระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_SPT_FCFS



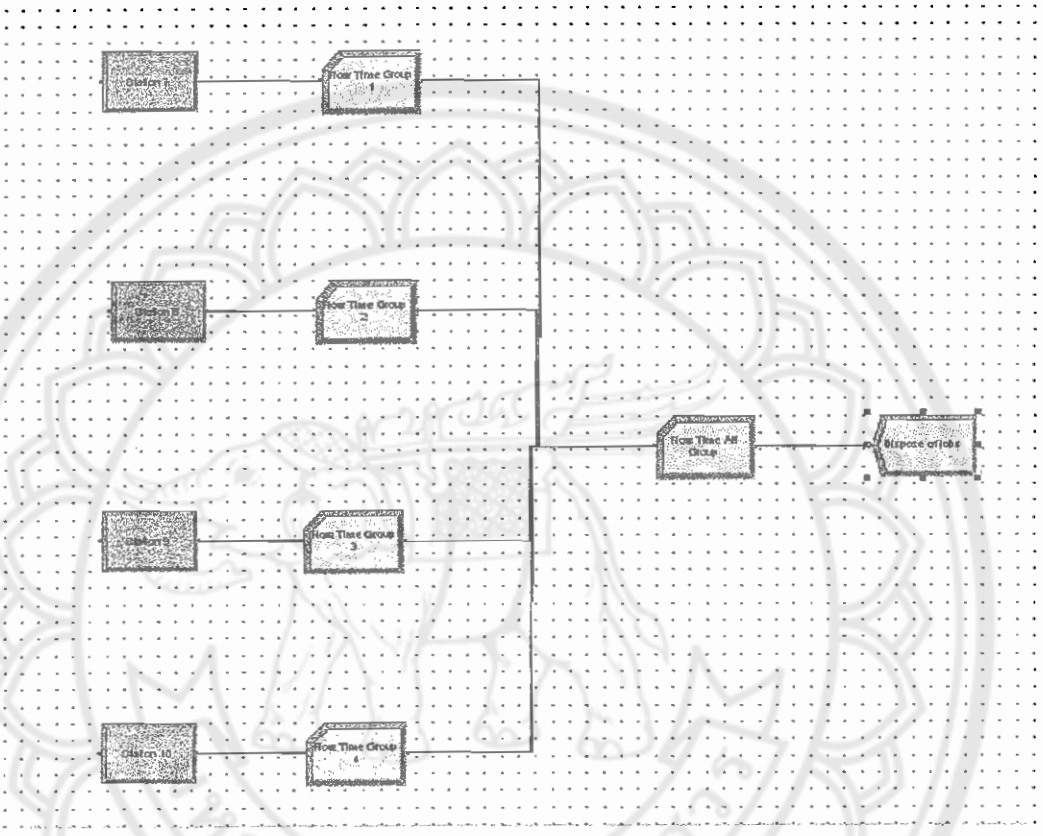
รูปที่ ง.8 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรในระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_SPT_FCFS

2.5 สร้างแบบจำลองของกลุ่มเครื่องจักรของระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_SPT_SPT



รูปที่ ง.9 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรในระบบการผลิตแบบ Functional Layout (FLF) โดยใช้หลักการกฎการจัดลำดับงานแบบ FLF_SPT_SPT

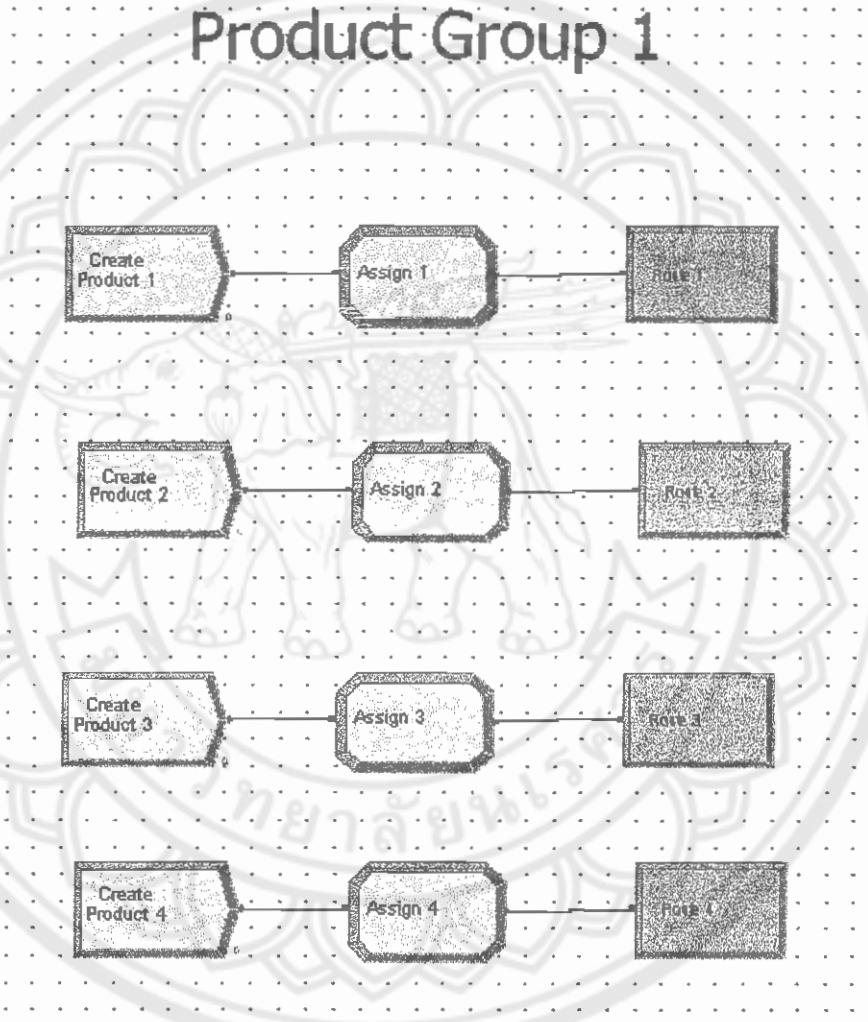
2.6 สร้างแบบจำลองเก็บข้อมูลของผลิตภัณฑ์ของระบบการผลิตแบบ Functional Layout (FLF)



รูปที่ ง.10 แสดงการสร้างแบบจำลองเก็บข้อมูลของผลิตภัณฑ์ในระบบการผลิตแบบ Functional Layout (FLF)

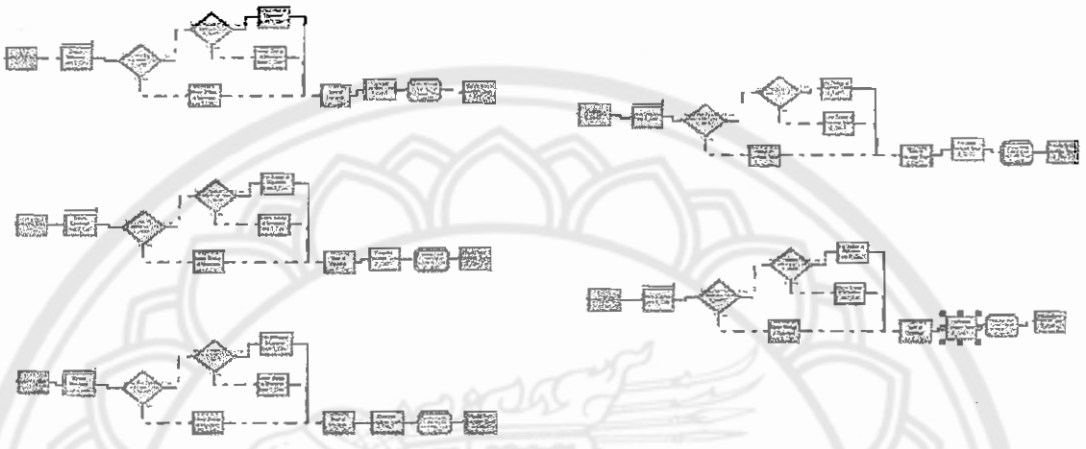
3. การจำลองสถานการณ์บนโปรแกรม Arena ระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบต่างๆ

3.1 สร้างแบบจำลองของกลุ่มผลิตภัณฑ์ของระบบการผลิตแบบ Cellular Manufacturing สามารถสร้างแบบจำลองได้ทั้ง 3 แบบ คือ CM1, CM2, CM3



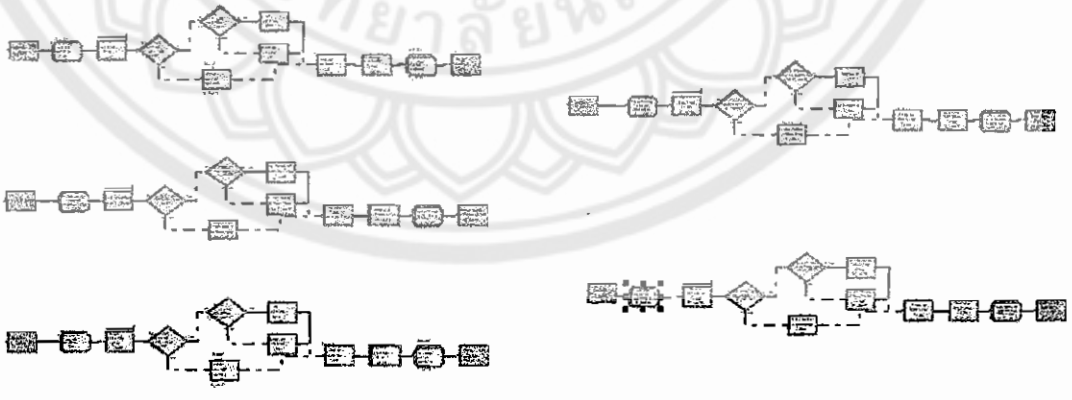
รูปที่ ง.11 แสดงการสร้างแบบจำลองของกลุ่มผลิตภัณฑ์ในระบบการผลิตแบบ Cellular Manufacturing

3.2 สร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 1 เซลล์ โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_JS_FCFS



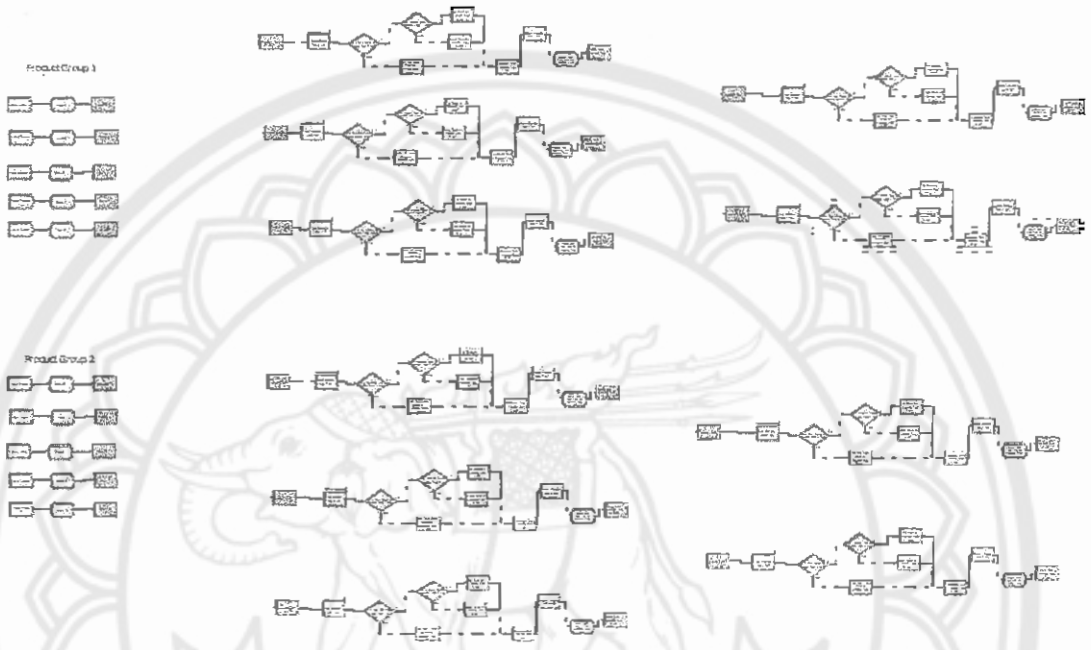
รูปที่ ง.12 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 1 เซลล์ในระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_JS_FCFS

3.3 สร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 1 เซลล์ โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_JS_SPT



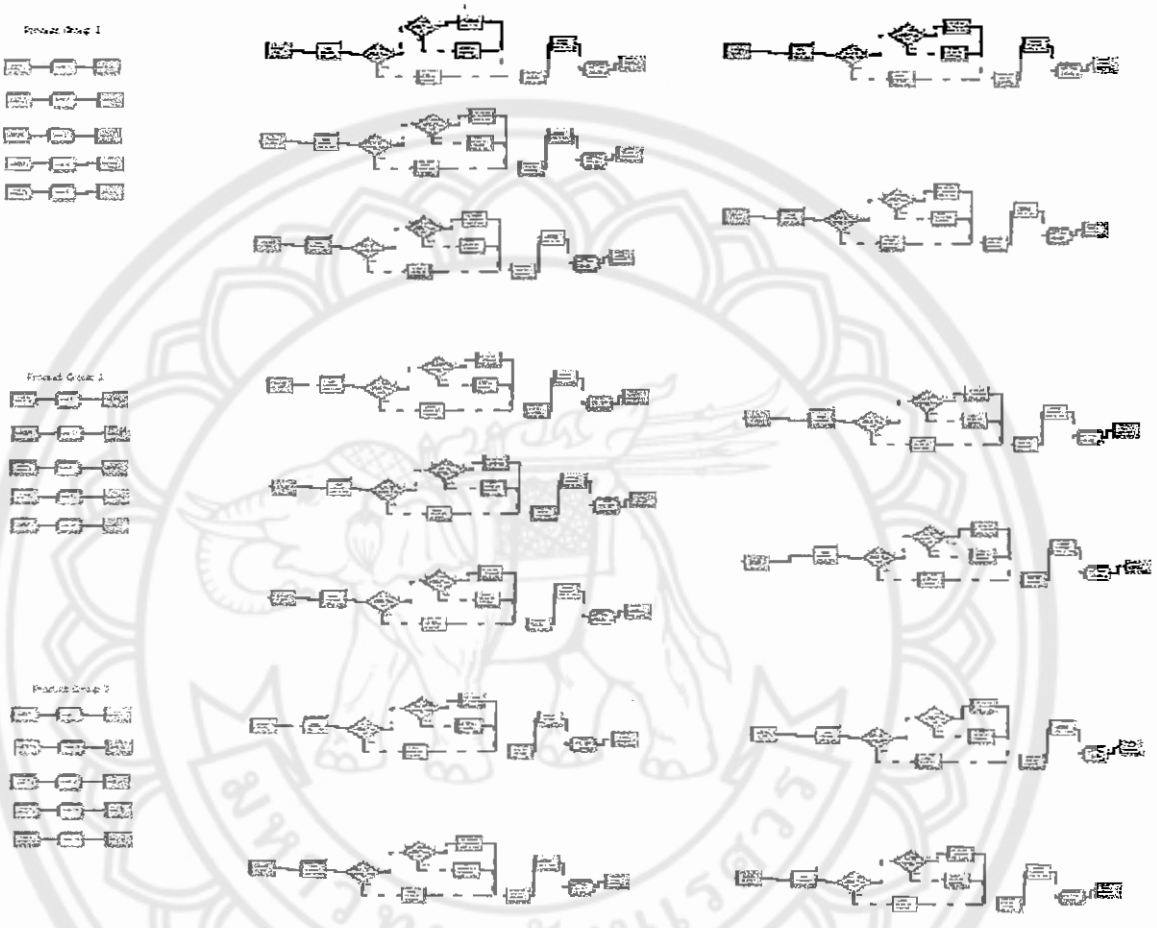
รูปที่ ง.13 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 1 เซลล์ในระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_JS_SPT

3.4 สร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 2 เซลล์ โดยใช้หลักการกฎการจัดลำดับงานแบบ CM2_JS_FCFS



รูปที่ ง.14 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 2 เซลล์ในระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM2_JS_FCFS

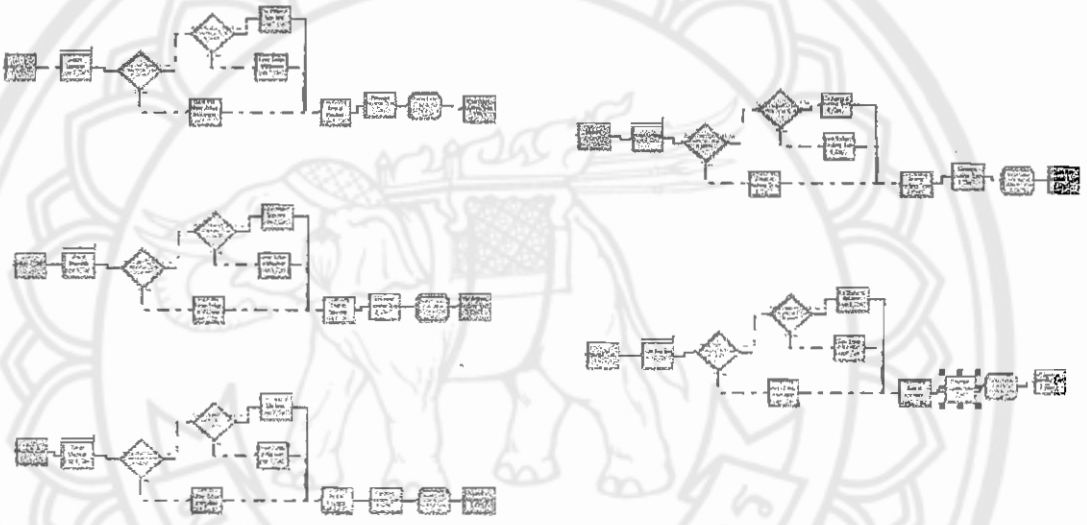
3.5 สร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 3 เซลล์ โดยใช้หลักการกฎการจัดลำดับงานแบบ CM3_JS_FCFS



รูปที่ ง.15 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรจำนวน 3 เซลล์ในระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM3_JS_FCFS

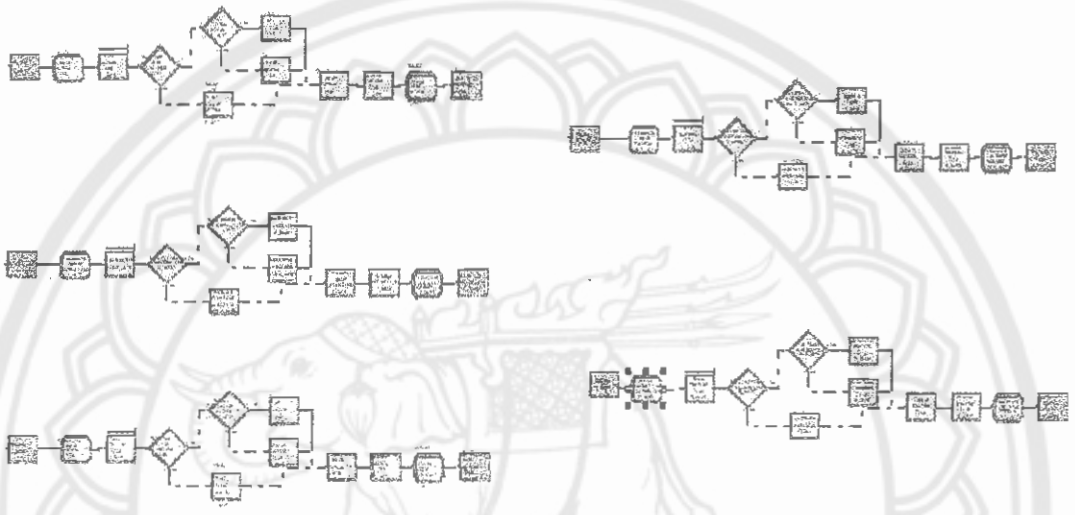
3.6 สร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดในเซลล์ของระบบการผลิตแบบ Cellular Manufacturing (CM1, CM2, CM3)

3.6.1 สร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดในเซลล์ของระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_JS_FCFS , CM2_ JS_FCFS_JS_FCFS และ CM3_ JS_FCFS_JS_FCFS



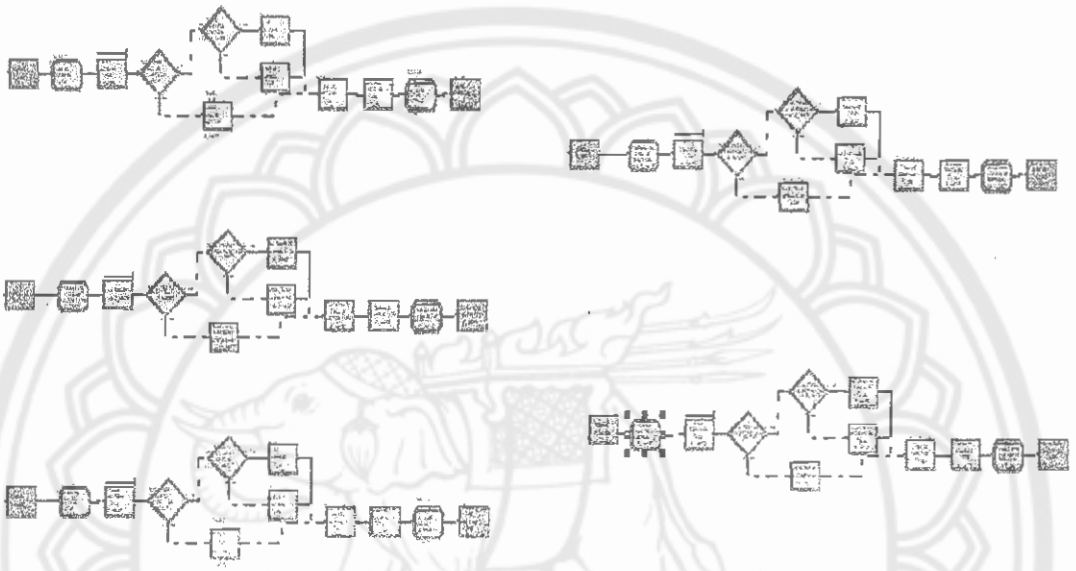
รูปที่ ง.16 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดเครื่องจักรในเซลล์ของระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_JS_FCFS , CM2_ JS_FCFS_JS_FCFS และ CM3_ JS_FCFS_JS_FCFS

3.6.2 สร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดในเซลล์ของระบบการผลิตแบบCellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_JS_FCFS, CM2_ JS_FCFS_JS_FCFS และ CM3_ JS_FCFS_JS_FCFS



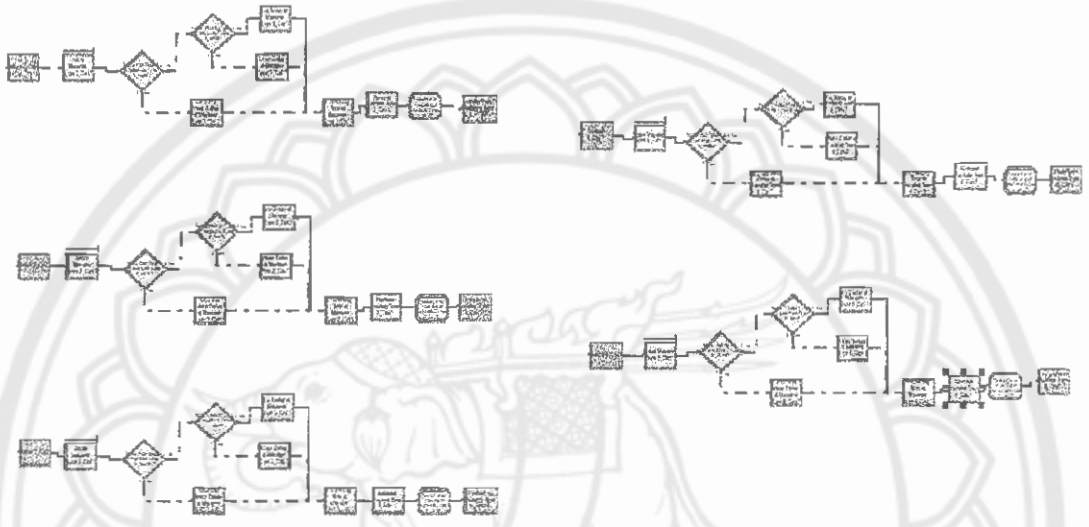
รูปที่ ง.17 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดเครื่องจักรในเซลล์ของระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_JS_FCFS, CM2_ JS_FCFS_JS_FCFS และ CM3_ JS_FCFS_JS_FCFS

3.6.3 **สร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดในเซลล์ของระบบการผลิตแบบCellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_ JS_SPT, CM2_ JS_FCFS_ JS_SPT และ CM3_ JS_FCFS_ JS_SPT**



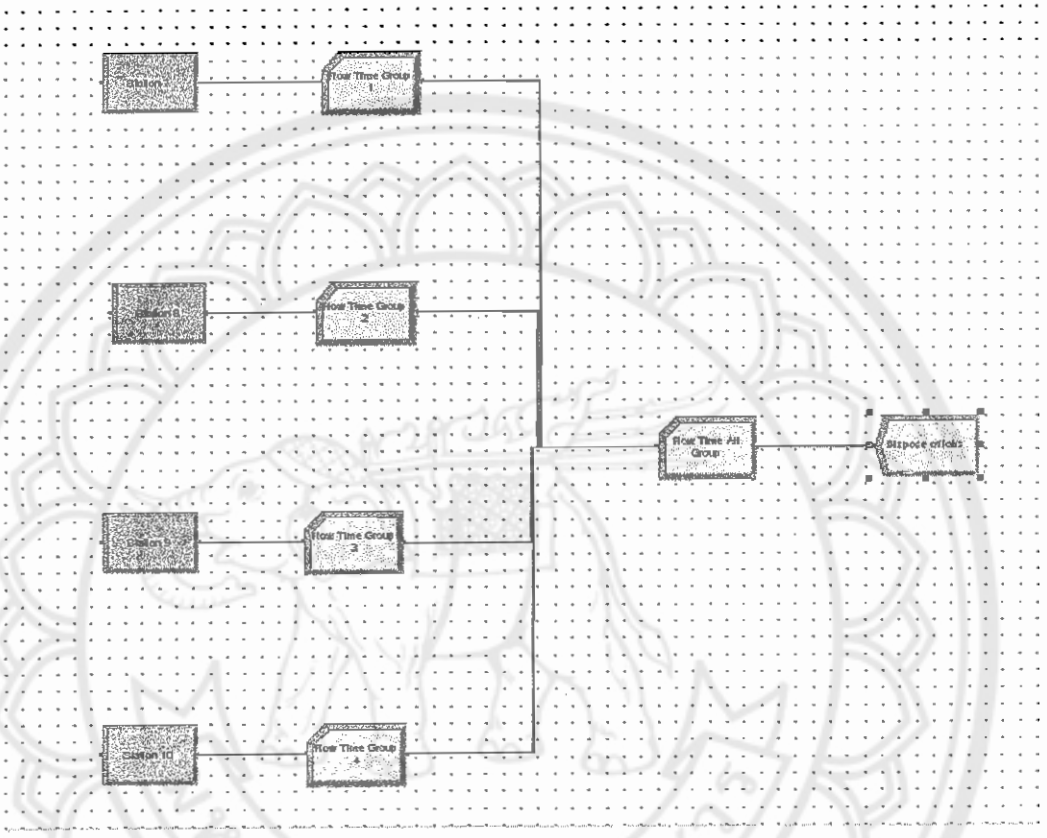
รูปที่ ง.18 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดเครื่องจักรในเซลล์ของระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_ JS_SPT, CM2_ JS_FCFS_ JS_SPT และ CM3_ JS_FCFS_ JS_SPT

3.6.4 สร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดในเซลล์ของระบบการผลิตแบบCellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_ JS_SPT, CM2_ JS_FCFS_ JS_SPT และ CM3_ JS_FCFS_ JS_SPT



รูปที่ ง.19 แสดงการสร้างแบบจำลองของกลุ่มเครื่องจักรที่เหลือจากการจัดเครื่องจักรในเซลล์ของระบบการผลิตแบบ Cellular Manufacturing โดยใช้หลักการกฎการจัดลำดับงานแบบ CM1_ JS_FCFS_ JS_SPT, CM2_ JS_FCFS_ JS_SPT และ CM3_ JS_FCFS_ JS_SPT

3.7 **สร้างแบบจำลองเก็บข้อมูลของผลิตภัณฑ์ของระบบการผลิตแบบ** Cellular Manufacturing **สามารถสร้างแบบจำลองได้ทั้ง** 3 **แบบ คือ** CM1, CM2, CM3



รูปที่ ง.20 แสดงการสร้างแบบจำลองเก็บข้อมูลของผลิตภัณฑ์ในของระบบการผลิตแบบ Cellular Manufacturing