สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์

EXPERT SYSTEM FOR DIAGNOSIS ON MAINBOARD MICROCOMPUTER

ร้อยเอก ศิริโรจน์ ฉันทใด

Capt. SIRIROTE JANTADAI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน.....24792
วัน, เดือน, ปี.....18 ส.ค. 2539

พ.ศ. 2538

ISBN 974-621-393-8

ลิขสิทธิ์ของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EXPERT SYSTEM FOR DIAGNOSIS ON MAINBOARD MICROCOMPUTER

SIRIROTE JANTADAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE

MASTER OF SCIENCE PROGRAM IN COMPUTER SCIENCE

AND INFORMATION TECHNOLOGY

KING MONGKUT 'S INSTITUTE OF TECHNOLOGY LADKRABANG

1995

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	ระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์
นักศึกษา	ร้อยเอก ศิริโรจน์ ฉันทใด
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร. ชม กิ้มปาน
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต
	สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
	สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง
พ.ศ.	2538

บทคัดย่อ

เป็นการนำเอาระบบผู้เชี่ยวชาญมาใช้กับ การค้นหาสาเหตุข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ โดยดำเนินการในส่วนของ การออกแบบฐานความรู้ (knowledge base) และใช้การแสดงความรู้ (knowledge representation) แบบกฎ โดยได้ทำการพัฒนาการแสดงความรู้บน เครื่องมือช่วยพัฒนาระบบผู้เชี่ยวชาญ ชื่อ VP - Expert และใช้วิธีการค้นหาแบบเดินหน้า ตามลักษณะของความรู้และการวิเคราะห์ปัญหา ระบบนี้สามารถค้นหาสาเหตุข้อขัดข้องแบบสแตติกได้ทั้งแบบจุดเดียวและหลายจุด โดยได้แบ่งสาเหตุข้อขัดข้องต่างๆ ที่สามารถเกิดขึ้นได้บนตัวอุปกรณ์ออกเป็น 4 กลุ่มได้แก่ กลุ่มสัญญาณออก, กลุ่มไฟเลี้ยงวงจร, กลุ่มสัญญาณควบคุม และกลุ่มสัญญาณเข้าตามลำดับ จากสาเหตุข้อขัดข้องทั้ง 4 กลุ่มนี้นำมาจัดทำเป็นโครงสร้างแฟ้มความรู้ และทำการพัฒนาบนโปรแกรมจัดการฐานข้อมูล เช่น dBase หรือ Foxpro ในการพัฒนาได้เพิ่มเติมรูปร่างของอักขระใหม่บางตัว เพื่อให้สามารถแสดงรูปร่างของแผนผังเวลาได้ และได้เพิ่มเติมโปรแกรมย่อยต่างๆ เพื่อสนับสนุนการอนุมานและการเริ่มต้นการทำงาน สำหรับความรู้ที่จะบรรจุในแฟ้มความรู้ได้มาจาก การทดลอง. คู่มือ IBM PC Advanced Troubleshooting & Repair และจาก TECHPRO Software for Troubleshooting the IBM XT ในการดำเนินการค้นหาข้อขัดข้องจะใช้แบบไกด์โพรบ โดยเริ่มต้นจากผลของข้อขัดข้องที่เกิดขึ้นแล้ว ให้ผู้ใช้ทำการจับสัญญาณตามจุดต่างๆ ตามคำแนะนำที่ปรากฏบนจอภาพ จากนั้นเปรียบเทียบสัญญาณที่ตรวจจับได้กับข้อมูลอ้างอิงที่อยู่ในแฟ้มความรู้ โดยติดต่อกับระบบผ่านทางจอภาพและแป้นพิมพ์ หลังจากนั้นระบบก็จะให้คำแนะนำ เพื่อทำการค้นหารุดหน้าต่อไปจนกว่าจะพบคำตอบ

จากวิธีการดังกล่าว นำมากำหนดเป็นแนวทางในการออกแบบการค้นหาที่เหมาะสมโดยใช้ฮิวริสติกและออกแบบกฎต่างๆ เพื่อใช้ในการเลือกเส้นทางการค้นหาข้อขัดข้องจำนวน 24 กฎ สำหรับในส่วนของการเรียนรู้ของระบบ เกิดจากการนำไปใช้งานแล้วนำผลการใช้งานมาปรับปรุงฐานความรู้ให้สามารถค้นหาสาเหตุข้อขัดข้องได้ดีกว่าเดิม ในการทดลอง ได้ทำการทดลองบนแผงวงจรหลักไมโครคอมพิวเตอร์ PC/XT โดยมีจำนวนสาเหตุข้อขัดข้องที่ทำการทดลองทั้งสิ้น 1155สาเหตุ ผลการทดลองสามารถค้นหาข้อขัดข้องได้ 1015 สาเหตุ หรือ 87.87 เปอร์เซ็นต์ของข้อขัดข้อง

จากแผนงานดังกล่าวนอกจากจะสามารถใช้ได้กับ แผงวงจรหลักไมโครคอมพิวเตอร์แล้ว ยังสามารถนำมาใช้ได้กับแผงวงจรดิจิตอลอื่นๆได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Expert System for diagnosis on Mainboard Microcomputer
Student	Capt. Sirirote Jantadai
Thesis Advisor	Assoc Prof.Dr. Chom Kimpan
Level of Study	Master of Science Program in Computer science and Information Technology
Department	Mathematics and Computer Science King Mongkut's Institute of Technology Ladkrabang
Year	1995

ABSTRACT

This content is about presenting the expert system to search for the malfunction on the main board microcomputer. The task is to design knowledge base, and use rule base as knowledge representation. Rule base is developed by using the expert system development tools , VP-Expert , as the main development tools. The strategy is forward reasoning which gets from analyzing the scope of problem and knowledge characteristic. This system can detect both, single stuck fault and multi stuck fault. For each device the malfunction can be divided into 4 groups : output group, power supply group, control signal group and input group which are managed to be the knowledge files structure. It also makes knowledge files on the database management program, such as dBase, Foxpro. Furthermore, it adds some new font so that the timing diagram can be showed on the screen; and adds other sub - programs which help initialization and inference engine.

The knowledge kept in knowledge base gets from the experiment, IBM PC Advanced Trouble shooting & Repair book and TECHPRO Software for Troubleshooting the IBM XT manuals. Fault isolation techniques is base upon guide probe. In order to find out the malfunctions, the user can use the screen and the keyboard to interface with the system. And then begining with the effect of the malfunction, check the related signal and compare with the data in the knowledge base. After that follow the instructions from the screen until the answer can be found.

So it can make guidlines for design heuristic search and rules (total 24 rules) that help us choose the best way in searching. The system can learn in the sense that we can apply the acquired knowledge and use it to improve the knowledge base better than before. The experiment try on mainboard micro computer PC/XT. It can find out 1015 faults from 1155 faults simulation or 87.87 percent fault coverage.

The mentioned plan can not only be used with the mainboard microcomputer but the other digital cards also.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ รองศาสตราจารย์ ดร. ชม กิ้มปาน ที่ได้ให้ความอนุเคราะห์ ทั้งทางด้านความรู้ คำปรึกษา และแนะนำการจัดทำวิทยานิพนธ์ฉบับนี้จนสำเร็จลุล่วงด้วยดี

ขอกราบขอบพระคุณ เหล่าคณาจารย์และเจ้าหน้าที่ สำนักงานวิจัยและบริการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง ในความช่วยเหลือทุกๆด้าน

ขอกราบขอบพระคุณ ผู้บังคับบัญชา และ นายทหาร กองซ่อมเครื่องสื่อสารอิเล็กทรอนิกส์ กรมการทหารสื่อสาร ทุกนายที่ให้การสนับสนุน อุปกรณ์และเครื่องมือ

ขออบคุณ ทุกคนในครอบครัว ที่ให้การช่วยเหลือสนับสนุน ข้าพเจ้ามาโดยตลอด

และสุดท้ายนี้ขอขอบพระคุณ

รองศาสตราจารย์ ดร. ชม กิ้มปาน

รองศาสตราจารย์ ดร. ครรชิต ไมตรี

ผู้ช่วยศาสตราจารย์ ดร. บุญวัฒน์ อัตชู

ดร. เอื้อน ปิ่นเงิน

ที่ให้ความกรุณาเป็นกรรมการสอบวิทยานิพนธ์ของข้าพเจ้าในครั้งนี้

ร.อ. ศิริโรจน์ ฉันทใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VIII

สารบัญภาพ IX

บทที่ 1. บทนำ 1

1.1. ความเป็นมาและความสำคัญของระบบผู้เชี่ยวชาญ 1

1.2. วัตถุประสงค์ของการวิจัย 3

1.3. แนวความคิดที่นำมาใช้ 3

1.4. ขอบเขตการวิจัย 4

1.5 วิธีที่ใช้ในการดำเนินการวิจัย 4

บทที่ 2. ทฤษฎี 5

2.1. องค์ประกอบระบบผู้เชี่ยวชาญ 5

2.1.1 ฐานความรู้ 6

2.1.2 หน่วยความจำที่ใช้งาน 6

2.1.3 กลไกการอนุมาน 6

2.1.4 การเชื่อมโยงกับผู้ใช้ 7

2.1.5 กลไกการพัฒนา 7

2.2. การแสดงความรู้ในรูปกฎ 7

2.2.1 ทฤษฎี Post machine 8

2.2.2 โปรดักชั่นซิสเต็ม 8

2.2.3 การควบคุมกลยุทธ 10

2.2.4 แฟกเตอร์ความแน่นอน 11

2.3. รูปแบบของโครงสร้างข้อมูลที่ใช้สำหรับการค้นหา 12

2.4. การค้นหาข้อขัดข้อง 12

2.4.1 ชนิดของการทดสอบวงจรโลจิก 12

2.4.2 การแบ่งระดับของการค้นหาข้อขัดข้อง 17

2.4.3 การค้นหาสาเหตุข้อขัดข้องทั่วไป 17

2.4.4 การค้นหาแบบฮิวริสติก 19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.5. การเรียนรู้โดยเครื่องจักร 21

บทที่ 3. ปัญหาและข้อมูล 22

3.1. ลักษณะทั่วไปของปัญหา 22

3.1.1 โมเดลข้อขัดข้องเชิงโลจิก....22

3.1.2 โมเดลข้อมูลเชิงโลจิก 23

3.2. ลักษณะของปัญหาในงานวิจัย 24

3.2.1 คุณลักษณะของข้อขัดข้อง 24

3.2.2 สถานะเริ่มต้น 24

3.2.3 แนวทางการแก้ไขปัญหา 25

3.2.4 สถานะเป้าหมาย 25

3.3. แฟ้มข้อมูลอ้างอิง 26

3.3.1 โมเดลข้อมูลของ SCIRTSS 26

3.3.2 โมเดลของตัวอุปกรณ์ในงานวิจัย 27

3.3.3 โครงสร้างแฟ้มข้อมูล 27

3.3.4 ข้อมูลฟังก์ชั่น 30

3.3.5 แผนผังเวลา 33

3.3.6 ตัวอย่างแฟ้มข้อมูล 37

บทที่ 4. การสร้างกฎและการอนุมาน 41

4.1. เครื่องมือที่ช่วยในการสร้าง 41

4.1.1 เปลือกระบบผู้เชี่ยวชาญ 41

4.1.2 เครื่องวิเคราะห์โลจิก 44

4.2. วิธีแสดงและใช้ความรู้....44

4.3 ฮิวริสติก 45

4.3.1 ฮิวริสติกในการค้นหาของ SCIRTSS 45

4.3.2 ฮิวริสติกฟังก์ชั่นในงานวิจัย 46

4.3.3 แนวทางการแก้ไขปัญหาแบบไกด์โพรบ 47

4.3.4 การนำค่าฮิวริสติกมาใช้โดยปริยาย 49

4.3.5 ตัวอย่างความจำเป็นที่ต้องใช้ผู้เชี่ยวชาญ 51

4.4. การสร้างกฎ 54

4.4.1 กฎที่ทำหน้าที่เกี่ยวกับการกรองหรือการคัดเลือก 55

4.4.2 กฎที่ทำหน้าที่เกี่ยวกับการแสดงข้อมูลอ้างอิง หรือ การแจ้งข่าวสาร 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

4.4.3 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 1 57
4.4.4 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 2 57
4.4.5 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยเชิงกายภาพ 58
4.4.6 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นสุดท้าย 59
4.4.7 กฎที่ทำหน้าที่เกี่ยวกับการควบคุมลูป 60
4.4.8 สรุปหน้าที่แต่ละกฎ 61
4.5. ทิศทางการอนุมาน 63
4.6. การควบคุมกลยุทธ 72
4.6.1 การเลือกกฎขึ้นมาแสดง 72
4.6.2 การควบคุมวงรอบการอนุมาน 73
4.6.3 ความรู้เมตา 77
4.7 การเรียนรู้ 78
4.7.1 แนวทางการพัฒนาการเรียนรู้ 79
บทที่ 5. การทำงานของโปรแกรมและตัวอย่างการทดสอบ 81
5.1. ขั้นตอนการทำงานและการค้นหาของโปรแกรม 81
5.2. ตัวอย่างการทดสอบ 86
การทดสอบเบื้องต้นก่อนการค้นหาข้อขัดข้องบนแผงวงจรหลัก 86
การค้นหาข้อขัดข้องบนแผงวงจรหลัก 86
การทดสอบที่ 1 90
การทดสอบที่ 2 92
การทดสอบที่ 3 95
การเรียนรู้ จากผลการทดสอบที่ 1, 2 และ 3 101
ตัวอย่าง สาเหตุข้อขัดข้อง 101
การทดสอบที่ 4 103
การทดสอบที่ 5 110
การสร้างแฟ้มข้อมูลโดยไม่ต้องใช้โปรแกรมการจัดเรียงขาอุปกรณ์ 114
การทดสอบที่ 6 115
การทดสอบที่ 7 118
บทที่ 6. สรุปผลและข้อเสนอแนะ 120
6.1. ลักษณะทั่วไป 120
6.2. ข้อจำกัด 122

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

6.3 ข้อเสนอแนะ 122
6.4 ลักษณะผู้เชี่ยวชาญที่ใช้ในการค้นหาข้อขัดข้อง 123
เอกสารอ้างอิง 124
ภาคผนวกที่ 1. คู่มือการใช้งาน 125
ภาคผนวกที่ 2. แฟ้มความรู้ CHIP.KBS 133
ภาคผนวกที่ 3. แฟ้มความรู้ START.DBF และ ชื่อสัญญาณในวงจร 143
ภาคผนวกที่ 4. แฟ้มความรู้ RUNCPU.DBF 147
ภาคผนวกที่ 5. แฟ้มความรู้ ADDRBUS1.DBF 149
ภาคผนวกที่ 6. แฟ้มความรู้ READBIOS.DBF 152
ภาคผนวกที่ 7. แฟ้มความรู้ DMATIMER.DBF 155
ภาคผนวกที่ 8. แฟ้มความรู้ ADDRBUS2.DBF 158
ภาคผนวกที่ 9. โปรแกรม COMPO.PRG 160
ภาคผนวกที่ 10. โปรแกรม STATEXT.C, STATE.FON file 163
ภาคผนวกที่ 11. โปรแกรม TRANSF.C 166
ภาคผนวกที่ 12. โปรแกรม SINK.C 168
ภาคผนวกที่ 13. โปรแกรม TSRKBD.C 170
ภาคผนวกที่ 14. โปรแกรม ROMTEXT.ASM 174
ภาคผนวกที่ 15. รายละเอียด วงจรหลัก PC/XT 175
ประวัติผู้เขียน 183

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 ชนิดของการทดสอบวงจรโลจิก 13
ตารางที่ 4.1 เปรียบเทียบการทดสอบข้อขัดข้องแบบไกด์โพรบของ ATE กับที่ใช้ในงานวิจัย 48
ตารางที่ 4.2 การสร้างกฎเกี่ยวกับการวินิจฉัยขั้นที่ 1 57
ตารางที่ 4.3 การสร้างกฎเกี่ยวกับการวินิจฉัยขั้นที่ 2 58
ตารางที่ 4.4 การสร้างกฎเกี่ยวกับการวินิจฉัยวินิจฉัยเชิงกายภาพ 59
ตารางที่ 4.5 การสร้างกฎเกี่ยวกับการวินิจฉัยขั้นสุดท้าย 59
ตารางที่ 4.6 ประกอบตัวอย่างที่ 4.3 การอนุมาน 65
ตารางที่ 5.1 ลำดับการทดสอบแผงวงจรหลัก.....89
ตารางผลการค้นหาข้อขัดข้องจากสัญญาณ RESET 91
ตารางผลการค้นหาข้อขัดข้องจากสัญญาณ CLK88 93
ตารางผลการค้นหาข้อขัดข้องจากสัญญาณ READY 97
ตารางผลการค้นหาข้อขัดข้องจากสัญญาณ ADDRESS 104

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพประกอบ

หน้า

รูปที่ 2.1 องค์ประกอบพื้นฐานระบบผู้เชี่ยวชาญ 5
รูปที่ 2.2 วงรอบการเรียกคำสั่งมาปฏิบัติ ของ โปรดักชั่นซิสเต็ม 9
รูปที่ 2.3 แสดงเส้นมาตราแฟคเตอร์ความแน่นอน 11
รูปที่ 2.4 การเปลี่ยนโครงสร้างข้อมูลจากแบบกราฟเป็นแบบต้นไม้ 12
รูปที่ 2.5 ตัวอย่างการทดสอบแบบฟังก์ชั่น 14
รูปที่ 2.6 ตัวอย่างการทดสอบแบบสแตติก พาราเมตริก 15
รูปที่ 2.7 ตัวอย่างการทดสอบแบบไดนามิกฟังก์ชั่น 16
รูปที่ 2.8 ตัวอย่างการทดสอบแบบไดนามิก พาราเมตริก 16
รูปที่ 2.9 การค้นหาแบบทางลึกก่อน 20
รูปที่ 2.10 การค้นหาแบบทางกว้างก่อน 20
รูปที่ 3.1 การแทนข่ายสัญญาณด้วยไดกราฟ 23
รูปที่ 3.2 โมเดลข้อมูลของ SCIRTSS 26
รูปที่ 3.3 โมเดลของตัวอุปกรณ์ในงานวิจัย 27
รูปที่ 3.4 โครงสร้างแฟ้มข้อมูล 28
รูปที่ 3.5 ผังงานของโปรแกรมจัดเรียงข้อมูล 32
รูปที่ 3.6 ข้อมูลแผนผังเวลาประกอบตัวอย่างที่ 3.1 34
รูปที่ 3.7 การแสดงรูปร่างอักขระของแผนผังสถานะ 36
รูปที่ 3.8 วงจร - แฟ้มความรู้ ประกอบตัวอย่างที่ 3.2 38
รูปที่ 3.9 วงจร - ตาราง ประกอบตัวอย่างที่ 3.3 39
รูปที่ 4.1 บล็อกไดอะแกรมของเปลือกระบบผู้เชี่ยวชาญ VP-Expert 41
รูปที่ 4.2 ทิศทางการอนุมาณของ VP-Expert 43
รูปที่ 4.3 ขบวนการค้นหาของ SCIRTSS 45
รูปที่ 4.4 การติดตามข้อขัดข้องของไกด์โพรบ 47
รูปที่ 4.5 วงจร - ตาราง ประกอบคำอธิบายการนำค่าฮิวริสติกมาใช้โดยปริยาย 49
รูปที่ 4.6 เส้นทางการค้นหาข้อขัดข้องประกอบตัวอย่างที่ 4.1 52
รูปที่ 4.7 วงจร - ตาราง - ไดกราฟประกอบตัวอย่างที่ 4.2 53
รูปที่ 4.8 โมเดล E-R ระหว่างตัวอุปกรณ์ 55
รูปที่ 4.9 การโยงกฎในงานวิจัย 62
รูปที่ 4.10 วงจร RESET ประกอบตัวอย่างการอนุมาน 64
รูปที่ 4.11 แผนผังการโยงกฎ เมื่อจับสัญญาณตอบสนอง บนขา ควบคุม หรือ อินพุต 71
รูปที่ 4.12 ผังงานการอนุมาณค้นหาสาเหตุข้อขัดข้อง 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพประกอบ(ต่อ)

หน้า

รูปที่ 4.13 โครงสร้างต้นไม้และแฟ้มความรู้เพื่อปรับปรุงแฟ้มความรู้ 79
รูปที่ 5.1 ส่วนประกอบของระบบ 81
รูปที่ 5.2 โครงสร้างฐานความรู้ START.DBF 82
รูปที่ 5.3 การเลือกใช้แฟ้มความรู้ 83
รูปที่ 5.4 การทดสอบเบื้องต้น ก่อนการค้นหาข้อขัดข้องบนแผงวงจรหลัก 88
รูปที่ 5.5 ผังงานลำดับการทดสอบแผงวงจรหลัก 89
รูปที่ 5.6 เส้นทางการค้นหาข้อขัดข้องจากสัญญาณ RESET 90
รูปที่ 5.7 เส้นทางการค้นหาข้อขัดข้องจากสัญญาณ CLK88 92
รูปที่ 5.8 การแก้ไขเส้นทางการค้นหาข้อขัดข้องบนสัญญาณ READY 95
รูปที่ 5.9 เส้นทางการค้นหาข้อขัดข้องบนสัญญาณ READY 96
รูปที่ 5.10 การรวมโครงสร้างต้นไม้หลายต้นเพื่อใช้ค้นหาข้อขัดข้องเมื่อสัญญาณADDRESSไม่ปกติ103
รูปที่ 5.11 เส้นทางการค้นหาข้อขัดข้องของการทดสอบที่ 4 และ 7 104
รูปที่ 5.12 การปรับปรุงแฟ้มความรู้ READBIOS.DBF 110
รูปที่ 5.13 เส้นทางการค้นหาข้อขัดข้องบนสัญญาณการอ่าน ROMBIOS 111
รูปที่ 5.14 เส้นทางการค้นหาข้อขัดข้องบน U82 115
รูปที่ 5.15 เส้นทางการค้นหาข้อขัดข้องบนสัญญาณ HRQDMA, DRQ0, OUT1 116
รูปที่ 5.16 การรวมโครงสร้างต้นไม้หลายต้นเพื่อใช้ค้นหาข้อขัดข้องเมื่อสัญญาณ AENBRDไม่ปกติ 118
รูปที่ 6.1 ไดอะแกรมสถานะ 121

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของระบบผู้เชี่ยวชาญ

ปัจจุบันคอมพิวเตอร์ได้เข้ามามีบทบาทอย่างมากมายในเกือบทุกวงการ ทั้งภาครัฐและภาคเอกชนมีการประยุกต์นำเอาคอมพิวเตอร์ไปใช้กันอย่างกว้างขวาง เช่น ช่วยเก็บและเตรียมเอกสาร ช่วยงานออกแบบ ช่วยควบคุมโรงงาน ช่วยในการเรียนการสอน เป็นต้น และอีกงานหนึ่งที่กำลังเป็นที่นิยมมาก คือ การนำคอมพิวเตอร์ไปใช้ในลักษณะผู้เชี่ยวชาญ ซึ่งคอมพิวเตอร์จะทำหน้าที่เก็บรวบรวมความรู้ ประสบการณ์ ความเชี่ยวชาญของผู้ชำนาญการในวงการต่างๆ เช่น วงการแพทย์, วิศวกรรม, การศึกษา, การทหาร, การขนส่ง เป็นต้น แล้วนำความรู้ ประสบการณ์ ความชำนาญเหล่านั้นมาคอยช่วยเหลือ ให้คำแนะนำในการตัดสินใจแก่ผู้ใช้ เหมือนกับได้รับคำปรึกษาจากตัวผู้เชี่ยวชาญในวงการนั้นๆ ซึ่งเป็นสาขาใหม่ในวงการคอมพิวเตอร์ ได้มีผู้ให้ความหมายคำว่า " ระบบผู้เชี่ยวชาญ " ไว้หลายแบบ แต่ก็สามารถนำมาสรุปได้ใจความว่าระบบผู้เชี่ยวชาญ คือระบบคอมพิวเตอร์ซึ่ง ประกอบขึ้นด้วยทั้ง ฮาร์ดแวร์ และ ซอฟต์แวร์ ทำงานเลียนแบบผู้เชี่ยวชาญ ในการแก้ไขปัญหาที่สลับซับซ้อน ในสาขาใดสาขาหนึ่ง (Field)โดยเฉพาะ เช่น ด้านการแพทย์, วิศวกรรม, สถาปัตยกรรม, การศึกษา, ... เป็นต้น และในวิทยานิพนธ์นี้เป็นการนำเสนอระบบผู้เชี่ยวชาญในการตรวจซ่อมแผงวงจรหลักของคอมพิวเตอร์ ซึ่งอยู่ในสายงานด้าน วิศวกรรมศาสตร์

ระบบผู้เชี่ยวชาญมีกำเนิดมาจากปัญญาประดิษฐ์ โดยเริ่มมาจากความเชื่อที่ว่า ด้วยกฎของการหาเหตุผลผนวกเข้ากับความสามารถของคอมพิวเตอร์ก็จะสามารถทำให้เกิดการทำงานที่เป็นไปอย่างเดียวกับที่ ผู้เชี่ยวชาญทำได้ โดยในกลางปี ค.ศ. 1960 Newell และ Simon ได้ค้นพบจากประสบการณ์ทางการวิจัยของเขาเองว่า กลยุทธ์ ของกระบวนการ การหาเหตุผลที่ต้องการให้มีลักษณะทั่วไปนั้น ยังมีข้อจำกัดอย่างมากในการนำมาใช้เพื่อแก้ปัญหาที่มีความซับซ้อนมากๆ จากการค้นพบครั้งนี้นี่เอง ทำให้งานวิจัยทางด้านปัญญาประดิษฐ์หันมาสนใจปัญหาที่มีขอบเขตจำกัดไม่ใหญ่โตครอบจักรวาล และได้มีการพัฒนาภาษา LISP เพื่อใช้ในการเขียนโปรแกรมเกี่ยวกับปัญญาประดิษฐ์ ระบบผู้เชี่ยวชาญเจริญก้าวหน้ามาโดยลำดับ ซึ่งต่อมาในปี ค.ศ. 1970 ก็ได้กำเนิดภาษาใหม่อีกภาษาหนึ่งเพื่อใช้ในการเขียนโปรแกรมเกี่ยวกับปัญญาประดิษฐ์เช่นเดียวกัน ชื่อ PROLOG ซึ่งมีรากฐานอยู่บนวิชาตรรกวิทยา และเป็นภาษาที่กำลังได้รับความนิยมมากอีกภาษาหนึ่งในปัจจุบัน หลังจากนั้นเมื่อเข้าครึ่งหลังของปี ค.ศ. 1970 จึงได้เกิดระบบผู้เชี่ยวชาญขึ้นอย่างมากมาย เช่น MYCIN ซึ่งใช้ในการวินิจฉัยและแนะนำเกี่ยวกับการตรวจโรคติดเชื้อในเลือด, DENTRAL ซึ่งใช้ในการวิเคราะห์โครงสร้างของสารประกอบทางเคมีชีวภาพ, SAINT ซึ่งใช้ในการคำนวณทางคณิตศาสตร์ เป็นต้น นอกจากนี้การพัฒนาระบบผู้เชี่ยวชาญยังได้ก้าวหน้าขึ้นมามาก โดยในปัจจุบันระบบผู้เชี่ยวชาญได้ถูกสร้างให้มีเฉพาะโครงสร้างภายนอกที่เรียกว่า เปลือกระบบผู้เชี่ยวชาญ ระบบนี้สามารถที่จะนำฐานความรู้ไปใส่ทีหลังได้ ตัวอย่างระบบผู้เชี่ยวชาญแบบนี้ เช่น Exsys,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GURU, PC plus, VP-Experet เป็นต้น ซึ่งในงานวิจัยนี้ได้ใช้เปลือกระบบผู้เชี่ยวชาญ VP-Expert ช่วยในการพัฒนา

ปัจจุบันประเทศผู้นำทางเทคโนโลยีทั่วโลกได้ให้ความสนใจกับวิทยาการแขนงนี้เป็นอย่างมาก ประเทศสหรัฐอเมริกา ประเทศในแถบยุโรปและหลายประเทศในเอเชีย เช่น ประเทศญี่ปุ่น ได้เห็นความสำคัญและประโยชน์ของการประยุกต์ใช้งานระบบผู้เชี่ยวชาญ เปรียบเสมือนเป็นกลยุทธ์ของการแข่งขันในเชิงธุรกิจ เพื่อที่จะช่วยปรับปรุงการบริการ การช่วยเพิ่มผลผลิต และการเพิ่มประสิทธิภาพของการตัดสินใจ ตัวอย่างการ ประยุกต์ ใช้งานจริง เช่น GATES[1] เป็นระบบผู้เชี่ยวชาญในการกำหนดประตูจอดเครื่องบินและติดตามตำแหน่งเครื่องบิน ที่ใช้งานอยู่ที่สนามบินนานาชาติ เจ เอฟ เค ในเมืองนิวยอร์ค สหรัฐอเมริกา, HESS[2] เป็นระบบผู้เชี่ยวชาญที่พัฒนาขึ้นโดยมหาวิทยาลัยยุสตัน สหรัฐอเมริกา ใช้สำหรับการกำหนดตารางการผลิต (Product Scheduling)ในโรงกลั่นน้ำมันหลักในรัฐเท็กซัส, FXAA[3]ระบบผู้ช่วยตรวจสอบบัญชีซื้อขายระหว่างประเทศของธนาคารเคมิคอล สหรัฐอเมริกา, ExperTax ระบบวางแผนการเสียภาษี เป็นต้น นอกจากนี้ประเทศญี่ปุ่นยังได้ประกาศไว้เป็นจุดมุ่งหมายหนึ่งใน " ระบบคอมพิวเตอร์รุ่นที่ 5 " [4] (Fifth Generation Computer System - FGCS) อันแสดงให้เห็นถึงความสำคัญของระบบผู้เชี่ยวชาญ

จากตัวอย่างระบบผู้เชี่ยวชาญที่กล่าวมาจะเห็นว่าปัญหาต่างๆที่เหมาะสมที่จะนำเอาระบบผู้เชี่ยวชาญไปช่วยแก้ไขนั้น แบ่งออกเป็น 2 ประเภทใหญ่ๆคือ

1. ปัญหาเชิงวิเคราะห์ เป็นปัญหาประเภทที่คอมพิวเตอร์จะต้องมีความรู้ความสามารถในการคิดหาเหตุผล ในการวิเคราะห์และพิจารณาข้อมูลต่างๆที่เข้ามาสู่ระบบแล้วหาข้อสรุปที่ถูกต้อง หรือเหมาะสม ปัญหากลุ่มนี้ อาจจะแบ่งแยกเป็นหลายลักษณะ เช่น ปัญหาการวินิจฉัย (Diagnosis) ปัญหาการตีความหมายจากข้อมูล (Intertretation) และปัญหาการพยากรณ์ เป็นต้น

2. ปัญหาเชิงสังเคราะห์ ปัญหาในประเภทนี้มักจะเกี่ยวข้องกับการใช้ความรู้ ความเชี่ยวชาญเพื่อการสร้างสรรค์ การออกแบบ หรือการสังเคราะห์ สร้างผลลัพธ์ โดยไม่ให้ขัดกับข้อจำกัดที่กำหนดไว้เป็น อินพุต ของระบบ ปัญหาประเภทนี้อาจจะแยกย่อยเป็นกลุ่มต่างๆ เช่น ปัญหาการออกแบบ ปัญหาการจัดวางผังระบบ ปัญหาการวางแผน เป็นต้น

แต่ก่อนที่จะนำเอาระบบผู้เชี่ยวชาญไปใช้แก้ปัญหาใดๆนั้น จะต้องจัดองค์ประกอบพื้นฐานให้กับระบบผู้เชี่ยวชาญก่อน ซึ่งประกอบด้วย 3 ส่วนที่สำคัญ ดังนี้

1. ฐานความรู้ ซึ่งเป็นที่รวบรวมความรู้ที่จะใช้ในการแก้ไขปัญหานั้นๆ สำหรับในงานวิทยานิพนธ์นี้ได้มาจาก การทดลอง, คู่มือ IBM PC Advanced Troubleshooting & Repair และ TECHPRO Software for Troubleshooting the IBM XT

2. กลไกการอนุมาน เป็นซอฟต์แวร์ที่ประกอบไปด้วยขบวนการอัตโนมัติต่างๆ ที่มีความสามารถในการนำความรู้จากฐานความรู้มาใช้ มาคิดหาเหตุผลเพื่อการแก้ไขปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่วนเชื่อมโยงกับผู้ใช้ ทำการเชื่อมโยงโปรแกรมระบบผู้เชี่ยวชาญกับโลกภายนอก หรือผู้ใช้ ส่วนเชื่อมโยงนี้จะรับข้อมูลปัญหาหรือเงื่อนไขต่างๆแล้วส่งต่อให้กลไกการอนุมาน คิดหาเหตุผลและดึงความรู้ที่เกี่ยวข้องมาจากฐานความรู้เพื่อทำการแก้ไขปัญหาที่เข้ามาสู่ระบบ แล้วส่งผลลัพธ์ที่อยู่ในรูปของคำแนะนำ คำอธิบาย หรือคำตอบ ตามลักษณะของปัญหาต่างๆ ออกมา ซึ่งในงานวิทยานิพนธ์นี้ ผู้ใช้ทำการติดต่อกับระบบโดยผ่านทางจอภาพและแป้นพิมพ์

จากองค์ประกอบพื้นฐานทั้ง 3 ข้อดังกล่าวมาแล้ว จะเห็นว่าหัวใจสำคัญของระบบผู้เชี่ยวชาญคือฐานความรู้ที่จะต้องบรรจุอยู่ภายในเครื่องคอมพิวเตอร์ ปัญหาจึงอยู่ที่ว่าจะบรรจุความรู้ ความเข้าใจของผู้เชี่ยวชาญที่เป็นมนุษย์ในรูปแบบอย่างไร ที่จะทำให้คอมพิวเตอร์สามารถนำความรู้นั้นมาประมวลผลได้ ในงานวิจัยนี้ได้เสนอแนวทางการออกแบบฐานความรู้ และ แนวทางการออกแบบการแสดงความรู้ในรูปแบบกฎที่จะใช้ในการพิจารณาแก้ไขปัญหา " การค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ " ซึ่งเป็นปัญหาเชิงวิเคราะห์ในกลุ่มของการวินิจฉัย อันเป็นอีกแนวทางหนึ่งในการประยุกต์เอาระบบผู้เชี่ยวชาญมาใช้งาน

1.2 วัตถุประสงค์ของการวิจัย

(1) เพื่อต้องการให้มีระบบผู้เชี่ยวชาญ ที่มีความรู้ สามารถวิเคราะห์ปัญหาสาเหตุข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ได้อย่างรวดเร็ว ถูกต้องแม่นยำ สามารถให้คำชี้แนะ และชักนำไปพบสาเหตุข้อขัดข้องบนแผงวงจรได้

(2) ต้องการให้เครื่องคอมพิวเตอร์เป็นที่ปรึกษาแก่ช่างซ่อม หรือผู้ที่ต้องการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ ได้

(3) เพื่อเป็นแนวทางในการพัฒนาระบบผู้เชี่ยวชาญการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ หรือแผงวงจรอื่นๆ ที่สมบูรณ์ต่อไป

1.3 แนวความคิดที่นำมาใช้

เป็นการนำระบบผู้เชี่ยวชาญมาใช้เกี่ยวกับปัญหาการวินิจฉัย สาเหตุข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ โดยใช้การแทนความรู้หรือการแสดงความรู้แบบกฎ และใช้แนวความคิดในการค้นหาคำตอบแบบเหมาะสมก่อนโดยใช้ฮิวริสติก [5] (Heuristic Search) ซึ่งเป็นการค้นหาไปในทิศทางที่ได้ประโยชน์หรือเป็นไปได้มากที่สุด (ในกรณี มีเส้นทางที่ต้องเลือกมากกว่าหนึ่งเส้นทาง) ซึ่งทิศทางนี้อาจจะไม่เป็นจริงทุกครั้งไป หรือไม่สามารถอธิบายด้วยเหตุผลทั้งหมด แต่ได้มาจากประสบการณ์หรือความชำนาญของผู้เชี่ยวชาญ ซึ่งเป็นผู้ถ่ายทอด หรือบรรจุความรู้เข้าไปในฐานความรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตการวิจัย

ในการดำเนินการวิจัยในครั้งนี้ จะดำเนินการในส่วนของการออกแบบ ฐานความรู้ (Knowledge base หรือ Information base), การแสดงความรู้ (Rule base) และ โปรแกรมสนับสนุนการทำงาน ส่วนการนำข้อมูลเข้า (สัญญาณอ้างอิงจากจุดต่างๆ) ให้ผู้ใช้เป็นผู้ดำเนินการพิมพ์เข้าด้วยมือ และการเปรียบเทียบสัญญาณตอบสนอง กับสัญญาณอ้างอิงให้ผู้ใช้เป็นผู้ตอบคำถาม ถูก/ผิด (Yes/No) จากกลไกการอนุมาน

1.5 วิธีที่ใช้ในการดำเนินการวิจัย

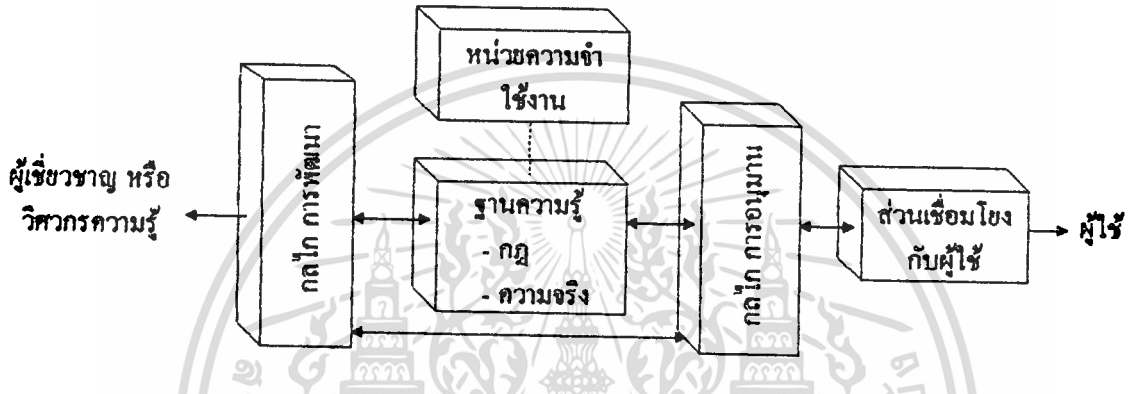
ในการดำเนินการวิจัย ระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์นี้ ดำเนินการพัฒนาบนเปลือกระบบผู้เชี่ยวชาญ VP-Expert ซึ่งเป็นเปลือกระบบผู้เชี่ยวชาญขนาดเล็ก และดำเนินการดังนี้

1. ดำเนินการออกแบบ/ปรับปรุง กฎการแสดงความรู้ต่างๆบน VP-Expert
2. ดำเนินการออกแบบ/ปรับปรุง โครงสร้างแฟ้มความรู้ บนโปรแกรมฐานข้อมูล FoxPro
3. วิเคราะห์เส้นทางการค้นหาข้อขัดข้องปัญหา โดยใช้ฮิวริสติก
4. ดำเนินการเก็บข้อมูลโดยใช้ เครื่องมือวิเคราะห์โลจิก (Logic Analyzer) จับสัญญาณต่างๆบนแผงวงจรหลักไมโครคอมพิวเตอร์ PC/XT (mainboard PC/XT)
5. ขยาย, ทดสอบ และปรับปรุงระบบด้วยการทดลองใช้งานจริงแล้ว กลับไปดำเนินการจากข้อ 1. มาใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 องค์ประกอบระบบผู้เชี่ยวชาญ



รูปที่ 2.1 องค์ประกอบพื้นฐานระบบผู้เชี่ยวชาญ

ในการที่จะทำให้เครื่องคอมพิวเตอร์มีขีดความสามารถเสมือนหนึ่งมีสติปัญญา สามารถที่จะวิเคราะห์หาเหตุผลได้ เราเรียกสติปัญญาเทียมๆเหล่านี้ว่าปัญญาประดิษฐ์ (Artificial Intelligence) ซึ่งปัจจุบันได้มีการประยุกต์นำเอาปัญญาประดิษฐ์นี้ไปใช้กันอย่างกว้างขวาง ทำให้เกิดการแตกสาขาเป็นศาสตร์หลายแขนงวิชาที่สำคัญได้แก่ การประมวลผลภาษาธรรมชาติ (Natural language processing) และ ระบบผู้เชี่ยวชาญ (Expert system)

ระบบผู้เชี่ยวชาญ (Expert system) คือระบบคอมพิวเตอร์ซึ่งประกอบขึ้นด้วยทั้ง ฮาร์ดแวร์และซอฟต์แวร์ ทำงานเลียนแบบผู้เชี่ยวชาญในการแก้ไขปัญหาที่สลับซับซ้อนในสาขาใดสาขาหนึ่ง (field) โดยเฉพาะ องค์ประกอบพื้นฐานของระบบผู้เชี่ยวชาญประกอบด้วย

(1) ฐานความรู้ (knowledge base)

(2) กลไกการอนุมาน (inference engine)

(3) การเชื่อมโยง (interface)

(4) กลไก การพัฒนา (development engine)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.1 แสดงองค์ประกอบพื้นฐานระบบผู้เชี่ยวชาญ ซึ่งแต่ละส่วนมีหน้าที่พอสรุปได้ดังนี้

2.1.1 ฐานความรู้ (Knowledge base)

คือส่วนที่ใช้เก็บรวบรวม ความรู้ที่เป็นจริง, กฎเกณฑ์ต่างๆ หรือประสบการณ์ความชำนาญต่างๆที่จะต้องนำเอามาสร้างเป็นระบบผู้เชี่ยวชาญ ประกอบด้วย

(1) ส่วนที่เป็นความจริงแน่นอนของสาขาวิชาที่ปัญหาเกี่ยวข้องอยู่ เช่น กฎ, ความจริงต่างๆ

(2) ส่วนที่เป็นจริงเฉพาะที่เกี่ยวข้องกับปัญหาที่กำลังแก้ไขอยู่ เช่น ข้อมูลต่างๆจากแฟ้มข้อมูล (หรือแฟ้มความรู้) ภายในฐานความรู้

ส่วนสำคัญของฐานความรู้คือ การแสดงความรู้ (Knowledge representation) ซึ่งเป็นส่วนที่ขบวนการในการค้นหาคำตอบ จะต้องนำไปใช้อนุมานหาคำตอบ ส่วนนี้ประกอบด้วย กฎ ความจริง และข้ออ้างต่างๆ ที่อธิบายเกี่ยวกับเรื่องที่ต้องการแก้ปัญหา ตัวอย่างของการแสดงความรู้ที่ใช้ในระบบผู้เชี่ยวชาญได้แก่ ข่ายความหมาย (semantic networks), กรอบ (frame and script), ตรรกวิทยา (predicate logic), กฎ(production rules) ซึ่งแต่ละแบบก็มีข้อดีข้อเสียแตกต่างกันไป โดยปกติแล้วไมโครคอมพิวเตอร์ [6] มักจะใช้การแสดงความรู้ด้วยกฎ ซึ่งรวมทั้งในงานวิจัยนี้ด้วย

2.1.2 หน่วยความจำที่ใช้งาน (Working memory)

คือพื้นที่หน่วยความจำที่ใช้ในการดึงข้อมูล, ความจริง และ กฎเกณฑ์ต่างๆ จากฐานความรู้นำมาประสมประสานกับข่าวสารข้อมูลที่ได้จากผู้ใช้ เพื่อเปรียบเทียบและดำเนินกรรมวิธีอนุมานหาคำตอบ (solution) ออกมา ซึ่งในวิทยานิพนธ์นี้ใช้หน่วยความจำที่จัดการโดย DOS 6.2 ขนาด 1 MB

2.1.3 กลไกการอนุมาน (Inference Engine)

คือกรรมวิธีและส่วนประกอบกรรมวิธี ในการที่จะนำเอาข้อเท็จจริง ความรู้ ความชำนาญต่างๆมาแก้ไขปัญหาที่เข้ามาสู่ระบบ กลไกการอนุมานจะทำการพิจารณา เหตุผลเป็นลำดับ เป็นขั้นตอน และเลือกทิศทางการให้เหตุผลที่จะนำผู้ใช้ไปสู่คำตอบตามกลยุทธของผู้เชี่ยวชาญแต่ละสาขาที่ได้วางขั้นตอนเอาไว้ เพื่อแก้ไขปัญหาที่มีลักษณะเช่นนั้น วิธีการหาเหตุผลของกลไกการอนุมานมีอยู่ 2 แบบคือ

(1) แบบเดินหน้าจากเหตุไปสู่ผล

(2) แบบถอยหลังจากผลเป้าหมายไปหาเหตุ

ในงานวิจัยนี้เลือกใช้แบบ เดินหน้าตามลักษณะของความรู้ และ การวิเคราะห์ปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 การเชื่อมโยงกับผู้ใช้(User interface)

คือการที่ผู้ใช้ จะติดต่อกับระบบผู้เชี่ยวชาญ เช่น อาจจะต้องป้อนปัญหาให้กับระบบ อาจจะต้องคอยตอบคำถามจากระบบ หรืออาจจะเป็นผู้ที่ถูกระบบผู้เชี่ยวชาญกำลังชี้แนะหนทางปฏิบัติ เป็นต้น ซึ่งในงานวิจัยนี้จะให้ผู้ใช้ทำการติดต่อกับระบบโดยผ่านทางจอภาพและแป้นพิมพ์

2.1.5 กลไกการพัฒนา (Development Engine)

กลไกการพัฒนาหรือเรียกอีกอย่างหนึ่งได้ว่าส่วนแก้ไขปรับปรุง เพื่อให้วิศวกรความรู้ หรือผู้เชี่ยวชาญ ใช้ในการสร้าง, เปลี่ยนแปลงแก้ไข, เพิ่มหรือลบข่าวสารข้อมูลให้กับฐานความรู้ โดยปกติแล้วกลไกการพัฒนาแบ่งออกเป็น 2 ประเภทคือ

(1) ภาษาคอมพิวเตอร์ ในระบบผู้เชี่ยวชาญต้องอาศัยขบวนการวินิจฉัยและถ่ายทอดความรู้ที่เรียกว่า กลไกการอนุมาน ดังนั้นจึงแตกต่างจากระบบฐานข้อมูลปกติที่เราเรียกข้อมูลมาใช้ทั้งนี้เพราะภาษาคอมพิวเตอร์แบบปกติ เป็นแบบกระบวนความ (procedure) จึงเหมาะที่จะใช้ในการเรียกข้อมูลหรือคำเนินการทำงานตามอัลกอริทึม แต่ไม่เหมาะที่จะเอามาเขียนระบบผู้เชี่ยวชาญเพราะยังขาดกลไกการอนุมาน ดังนั้นภาษาที่อาศัยขบวนการ ที่เรียกว่า อำนาจจำแนก[8] (resolution) หรือการถ่ายทอดความรู้ในลักษณะกลไกการอนุมาน จึงต้องเป็นภาษาในลักษณะพิเศษ ปัจจุบันมีการพัฒนาภาษาในลักษณะนี้ ซึ่งได้แก่ LISP และ PROLOG ขึ้นมา ภาษาคอมพิวเตอร์ที่พัฒนามาเฉพาะนี้มีกลไกการอนุมานโดยเฉพาะ จึงทำให้การวินิจฉัยและถ่ายทอดความรู้จากฐานความรู้ เป็นไปได้ง่ายและเหมาะสมกว่าการใช้ภาษาคอมพิวเตอร์แบบเก่า อย่างไรก็ตามภาษาคอมพิวเตอร์ปกติ เช่น FORTRAN, PASCAL, COBOL, C ก็สามารถใช้พัฒนาระบบผู้เชี่ยวชาญได้เช่นเดียวกัน แต่สิ้นเปลืองเวลามาก เพราะต้องคำนึงถึงกลไกการอนุมานด้วย

(2) เปลือกระบบผู้เชี่ยวชาญ เนื่องจากวิวัฒนาการของระบบผู้เชี่ยวชาญได้เจริญขึ้นมาก ทำให้ส่วนที่เป็นฐานความรู้ และส่วนที่ทำหน้าที่ในการแก้ปัญหาไม่จำเป็นที่จะต้องถูกสร้างขึ้นมาพร้อมกัน ด้วยเหตุผลดังกล่าวนี้จึงมีผู้สร้างระบบผู้เชี่ยวชาญที่มีเฉพาะ ส่วนที่ทำหน้าที่ในการแก้ปัญหา ระบบนี้สามารถนำไปใส่ความรู้ทีหลังได้ ระบบผู้เชี่ยวชาญแบบนี้ถูกเรียกว่าเปลือกระบบผู้เชี่ยวชาญ (Expert System Shell) และเพื่อเป็นการหลีกเลี่ยงการเริ่มต้นด้วยการพัฒนาระบบซอฟต์แวร์ซึ่งใช้เวลานานมาก ในงานวิจัยนี้จึงเลือกใช้ เปลือกระบบผู้เชี่ยวชาญ VP-Expert

2.2 การแสดงความรู้ในรูปกฎ

การแสดงความรู้ด้วยกฎมีชื่อเรียกอีกอย่างหนึ่งว่าการแสดงความรู้ในรูปโปรดักชั่นซิสเต็ม (Production system) ซึ่งถือได้ว่าเป็นโมเดลคอมพิวเตอร์ชนิดหนึ่งที่มีรากฐานทางทฤษฎีอยู่บน Post machine ที่เสนอโดย E.L. Post ในปี ค.ศ.1943

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ทฤษฎี Post machine[7]

เครื่องคอมพิวเตอร์ในปัจจุบันมีพื้นฐานทางทฤษฎีอยู่บน Turing machine ใน Turing machine จะบันทึกขั้นตอนการควบคุมไว้ทั้งหมด การปฏิบัติ หรือการประมวลผลของเครื่อง จะเป็นไปตามขั้นตอน การควบคุมนั้น แต่ใน Post machine ขั้นตอนการปฏิบัติการ หรือการประมวลผล จะถูกบันทึกอยู่ในรูปของเซ็ตของกฎ กฎอันไหนจะถูกใช้ก่อนหลังนั้นไม่ได้ขึ้นอยู่กับลำดับการบันทึกกฎ แต่ขึ้นอยู่กับว่าเงื่อนไขของกฎนั้นครบถ้วนสมบูรณ์หรือไม่ ถ้าครบก็จะมีการปฏิบัติตามกฎนั้น มีคนพิสูจน์แล้วว่าทั้ง Turing และ Post machine มีความสามารถในการประมวลผลเท่าเทียมกัน ดังนั้นความสามารถในการประมวลผลของโปรดักชั่นซิสเต็ม จึงเท่ากับภาษาโปรแกรมมิ่งอื่นๆ เช่น FORTRAN หรือ PASCAL กล่าวคือโปรแกรมทุกโปรแกรมที่เขียนด้วยภาษา FORTRAN หรือ PASCAL หรือภาษา โปรแกรมมิ่งอื่นๆ จะแทนได้ด้วยการใช้ โปรดักชั่นซิสเต็ม

กฎใน โปรดักชั่นซิสเต็ม จะอยู่ในรูป

IF .....

THEN .....

ส่วนของ IF เรียกว่า ส่วนเงื่อนไข และส่วนของ THEN เรียกว่าส่วนข้อสรุปหรือส่วนการปฏิบัติ

2.2.2 โปรดักชั่นซิสเต็ม (Production System)

โปรดักชั่นซิสเต็ม เปรียบเสมือนกับเป็นภาษาคอมพิวเตอร์อีกภาษาหนึ่ง แนวความคิดนี้ได้รับความสนใจจากนักคอมพิวเตอร์ไม่น้อย และมีการพัฒนา โปรดักชั่นซิสเต็ม ขึ้นเป็นภาษาคอมพิวเตอร์จริงๆ เช่น OPS5, ACT*[5] นอกจากนี้การแสดงความรู้ด้วยโปรดักชั่นซิสเต็มยังได้รับการนำไปใช้ในการสร้างเปลือกระบบ ผู้เชี่ยวชาญอีกด้วย เช่น SOAR[5], VP-Expert เป็นต้น

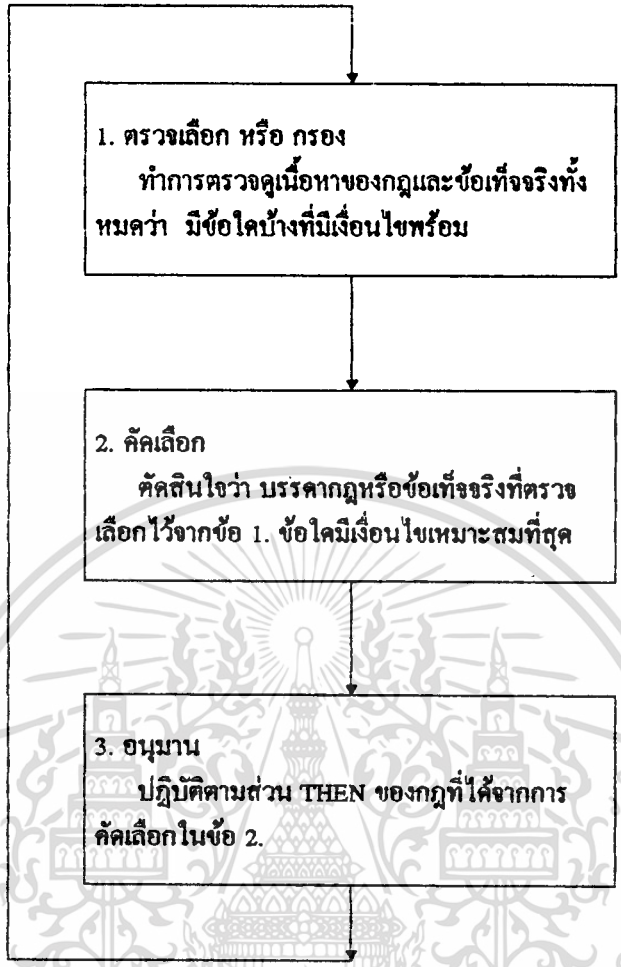
กลไกของโปรดักชั่นซิสเต็ม ประกอบด้วย

1. ชุดคำสั่งซึ่งแต่ละคำสั่งจะอยู่ในรูปแบบ *LS* --> RS[9] คือมีเงื่อนไขหรือสถานการณ์อยู่ทางซ้ายและปฏิบัติการอยู่ทางขวา เช่นเดียวกับคำสั่ง IF....THEN....

2. ฐานข้อมูล สำหรับบันทึกสถานะของการปฏิบัติการ หรืออีกนัยหนึ่ง คือ ส่วน"ความคิด" ในโมเดลข้างต้น และจะตรงกับส่วน "หน่วยความจำที่ใช้งาน" ในรูปที่ 2.1

3. กลไกการเรียกคำสั่งมาปฏิบัติเป็นลูปที่เลือกคำสั่งที่สามารถปฏิบัติได้มาปฏิบัติ วนอยู่อย่างนี้จนจบโปรแกรมดังแสดงในรูปที่ 2.2 ในส่วนนี้จะตรงกับการควบคุมการอนุมาน หรือ การควบคุมกลยุทธ ซึ่งจะได้กล่าวต่อไปในหัวข้อที่ 2.2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 วงรอบการเรียกคำสั่งมาปฏิบัติ ของ โปรดักชั่นซิสเต็ม

การเลือกกฎ ของ โปรดักชั่นซิสเต็ม

จากรูปที่ 2.2 จะเห็นว่าหลังจากทำการตรวจดูเนื้อหาของกฎทั้งหลายแล้ว อาจจะได้กฎที่ผ่านการกรองมาหลายกฎ ดังนั้นจึงจำเป็นจะต้องคัดเลือกเอากฎใดกฎหนึ่งขึ้นมา วิธีการคัดเลือกกฎอาจจะทำได้หลายวิธีดังนี้

1. เลือกตามลำดับของกฎ วิธีการนี้จะนำกฎมาทำการกรอง กฎแรกที่ผ่านการกรองมาได้ จะได้รับการปฏิบัติทันที
2. เลือกตามลำดับความละเอียดของส่วน IF ของกฎ วิธีการนี้กฎที่ส่วน IF มีการบันทึกเงื่อนไขไว้ละเอียดหรือเฉพาะเจาะจงมากที่สุดจะได้รับการคัดเลือกก่อน
3. เลือกตามความใหม่ของกฎ วิธีการนี้ กฎที่ได้รับการปฏิบัติ ล่าสุดจะได้รับการเลือกก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เลือกตามความใหม่ของข้อมูลใน หน่วยความจำใช้งาน วิธีการนี้กฎที่สอดคล้องกับข้อมูลล่าสุดในหน่วยความจำใช้งานจะได้รับการคัดเลือกก่อน

5. แบบขนาน วิธีการนี้ กฎทุกกฎที่ผ่านการกรอง จะได้รับการปฏิบัติพร้อมกัน

สำหรับในงานวิจัยนี้ใช้เปลือกระบบผู้เชี่ยวชาญ VP-Expert เป็นเครื่องช่วยพัฒนา ดังนั้นวิธีการเลือกกฎจึงเป็นไปตามกระบวนการของ VP-Expert คือ เลือกตามลำดับของกฎและเลือกตามความละเอียดของส่วน IF

2.2.3 การควบคุมกลยุทธ

การปฏิบัติงานทั้งหมดของโปรดักชั่นซิสเต็ม ในโปรแกรมระบบผู้เชี่ยวชาญจะอยู่ภายใต้การควบคุมของลูปใหญ่ ซึ่งทำหน้าที่เลือกคำสั่งที่เหมาะสมมาปฏิบัติในแต่ละรอบ เรียกว่า การควบคุมกลยุทธ (control strategy) หรือการควบคุมการอนุมาน เนื่องจากแต่ละคำสั่งจะทำงานได้ก็ต่อเมื่อเงื่อนไขทางซ้ายมือของคำสั่งเป็นจริง เราจึงเรียกการวนลูปนี้ว่า [10]Recognize - act cycle

หัวใจของลูปนี้ คือ การเลือกคำสั่งมาปฏิบัติ ซึ่งจะมีความเกี่ยวพันกัน 2 เรื่อง คือ เรื่องตัวคำสั่งกับเรื่องความรู้ในฐานข้อมูล เนื่องจากคำสั่งส่วนใหญ่จะปฏิบัติได้ ก็ต่อเมื่อ ได้มีการพิสูจน์ความจริงของเงื่อนไขทางซ้ายมือแล้ว โดยอาศัยความรู้หรือหลักฐานในฐานข้อมูล

สำหรับทิศทางการหาเหตุผลของกลไกการอนุมานจะมีอยู่ 2 แบบคือ

(1) แบบเดินหน้าจากเหตุไปสู่ผล (Forward Chaining) หรือบางครั้งเรียกว่า การหาเหตุผลโดยใช้ข้อมูลเป็นตัวขับ [7](Data driven Reasoning) การหาเหตุผลแบบนี้ระบบผู้เชี่ยวชาญอาจจะรับข้อมูลบางอย่างจากผู้ใช้ แล้วนำข้อมูลเหล่านี้ไปเปรียบเทียบกับเงื่อนไขของกฎต่างๆ ในฐานความรู้ แล้วใช้กฎข้อนั้นๆมาประกอบกันเป็นลำดับในการหาเหตุผล ระบบอาจจะมีการตอบโต้กับผู้ใช้เพื่อหาข้อมูลเพิ่มเติมในการประกอบการตัดสินใจเพื่อมุ่งไปสู่คำตอบ การหาเหตุผลแบบนี้เป็นการที่ระบบพยายามที่จะใช้กฎเกณฑ์ทีละข้อส่งผลลัพธ์ไปกระตุ้นให้ผลลัพธ์ข้ออื่นๆเป็นจริงตามไปเป็นลูกโซ่ จนกระทั่งถึงผลลัพธ์ที่เป็นคำตอบ หรือเป้าหมาย ในกรณีที่เงื่อนไขของกฎหลายข้อเป็นจริงพร้อมกัน ระบบอาจจะต้องมีการทดลองหาเหตุผลไปในหลายทิศทางก็ได้

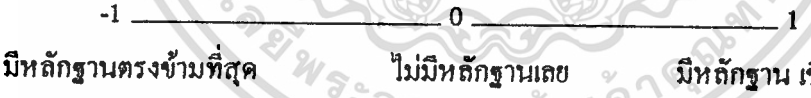
(2) แบบถอยหลังจากผลเป้าหมายไปหาเหตุ (Backward Chaining) หรือบางครั้งเรียกว่า การหาเหตุผลโดยการใช้ผลลัพธ์เป้าหมายเป็นตัวขับ [7](Goal driven Reasoning) การหาเหตุผลแบบนี้จะมีลักษณะสวนทิศทางกับการหาเหตุผลแบบเดินหน้า คือ ระบบจะต้องรับอินพุตที่เป็นเป้าหมายหรือข้อสรุปจากผู้ใช้ แล้วระบบจะพยายามหาข้อยืนยันหรือพยายามพิสูจน์ให้ได้ว่าเป้าหมายหรือข้อสรุปที่กำหนดให้นั้นเป็นจริง (หรือไม่จริง) โดยใช้กฎเกณฑ์เงื่อนไขและข้อเท็จจริงต่างๆที่มีอยู่ในฐานความรู้ว่ากฎข้อไหนที่ตรงกับเป้าหมายหรือข้อสรุป ถ้ากฎข้อนี้ยังมีหลักฐานไม่เพียงพอที่จะสรุปคำตอบได้ กฎข้อนี้ก็จะถูกนำไปใช้เพื่อสืบหลักฐานจากกฎข้ออื่นๆต่อกันไปเป็นเส้นทาง จนกระทั่งถึงผลลัพธ์ที่เป็นคำตอบ หรือเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 แฟคเตอร์ความแน่นอน

การแสดงความรู้ด้วยกฎเป็นหนึ่งในปัญญาประดิษฐ์ที่ ยอมรับสถานการณ์การอนุมานในลักษณะที่ สามารถสรุปข้อเท็จจริงด้วยปริมาณหลักฐานเท่าที่มี เมื่อมีหลักฐานเพิ่มขึ้น หรือเปลี่ยนไปเป็นอื่นก็สามารถปรับข้อสรุปนั้นได้ อย่างไรก็ดีการใช้เหตุผลแบบนี้ไม่มีวิธีสำหรับการพิสูจน์ความจริงของข้อความของตัวเอง คือไม่มีสัจจพจน์ (axiom) จึงไม่มีการพิสูจน์ความมั่นคง (consistency)หรือความสมบูรณ์(completeness) ของระบบ ทำให้เราไม่สามารถระบุความถูกต้องของการทำงานของการใช้เหตุผลแบบนี้ได้อย่างสมบูรณ์ เราจึงเชื่อผลการทำงานของโปรแกรมระบบผู้เชี่ยวชาญได้โดยการนำไปใช้จริงเท่านั้น ถ้าใช้ได้ผลระบบก็อาจจะถูกต้อง ถ้าไม่ได้ผลระบบก็มีข้อผิดพลาดบางอย่าง ในลักษณะเดียวกับการจ้างผู้เชี่ยวชาญมนุษย์ ซึ่งอาจจะทำงานถูกต้องก็ได้ หรืออาจจะผิดพลาดก็ได้ เราไม่มีทางรู้จนกว่าผลงานจะปรากฏ

กลไกหลักของการแก้ปัญหาได้แก่ การตั้งสมมติฐาน การพิสูจน์สมมติฐาน และการเลือกระหว่างคำตอบหลายๆคำตอบ การตั้งสมมติฐานคือการตั้งเป้าหมายของการอนุมานขึ้นมาหลายๆเป้าหมาย หลังจากนั้นโปรแกรมก็พยายามจะสืบสาวโดยใช้ความรู้ที่เป็นกฎเกณฑ์ ประกอบกับหลักฐานทำการอนุมานเพื่อให้ได้ข้อสรุปออกมา สมมติฐานแต่ละข้อจะมีค่าความเชื่อ (measure of belief ย่อว่า MB) และค่าความไม่เชื่อ (measure of disbelief ย่อว่า MD) อยู่คู่หนึ่งซึ่งในการอนุมานใช้ความรู้แต่ละข้อ จะมีการคำนวณเพื่อปรับค่าคู่นี้ จนในที่สุดเมื่อเสร็จสิ้นการอนุมาน จะได้ค่าออกมาค่าหนึ่งเรียกว่าค่าแฟคเตอร์ความแน่นอน (certainty factor ย่อว่า CF) ของสมมติฐานข้อนั้นซึ่งหาได้จาก CF = MB - MD ซึ่งผลที่ได้นี้จะนำมาเปรียบเทียบกัน ระหว่างสมมติฐานที่ต่างกัน ทำให้โปรแกรมสามารถเลือกข้อสมมติฐานที่มีค่า CF สูงสุด เป็นคำตอบของปัญหาได้ ค่า CF นี้ เป็นค่าที่ขึ้นกับหลักฐาน คือมีความหมายดังแสดงเป็นเส้นมาตรา ดังรูปที่ 2.3



รูปที่ 2.3 แสดงเส้นมาตราแฟคเตอร์ความแน่นอน

ตัวอย่างเช่น CF = .7 มีความหมายว่า " ถ้าเงื่อนไขต่อไปนี้....เป็นจริง มีหลักฐานให้เชื่อได้ 70 % ว่าสมมติฐานข้อนี้....ถูกต้อง "

สำหรับค่า CF ของ VP-Expert ซึ่งใช้ในงานวิจัย มีค่าอยู่ระหว่าง 0 - 100 หรือ ในรูปของเปอร์เซ็นต์ความเชื่อได้นั่นเอง

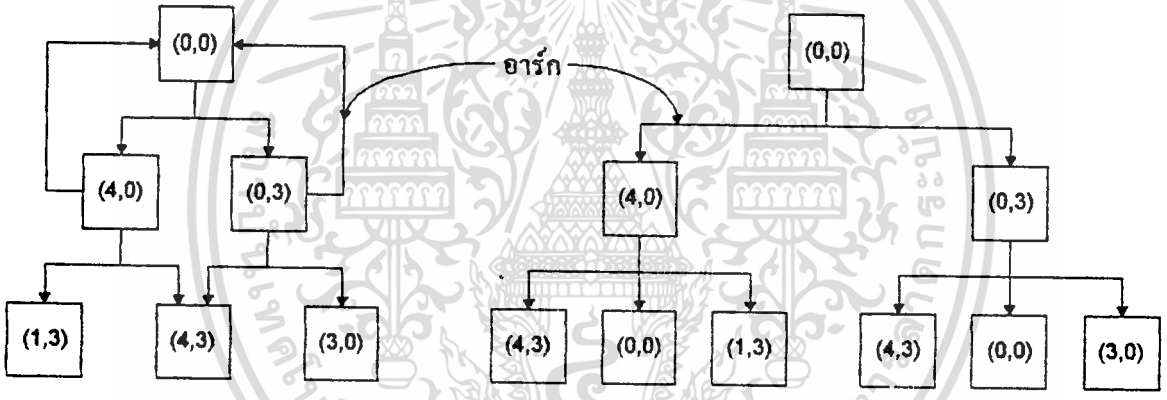
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 รูปแบบของโครงสร้างข้อมูลที่ใช้สำหรับการค้นหา

ในการกำหนดโครงสร้างข้อมูลสำหรับการค้นหานั้น โดยส่วนใหญ่แล้วจะถูกกำหนดให้อยู่ในรูปของโครงสร้างต้นไม้ หรือโครงสร้างไดกราฟ [11](Digraph) ซึ่งโครงสร้างทั้งสองนี้สามารถเปลี่ยนแปลงไปมาได้ องค์ประกอบของโครงสร้างแบบนี้ประกอบด้วย โหนด และอาร์ก โดยที่โหนดใช้แทนความหมายที่จะแสดงในฐานความรู้ และอาร์กใช้แทนความสัมพันธ์ของโหนดต่างๆ

สำหรับการเปลี่ยนกระบวนการค้นหาในโครงสร้างกราฟเป็นกระบวนการค้นหาในโครงสร้างต้นไม้ สามารถทำได้ด้วยวิธีการ ดังต่อไปนี้

1. ในแต่ละโหนด ตรวจสอบว่ามีอาร์กของโหนดอื่นชี้เข้ามาหาหรือไม่ (indeg > 0)
2. ถ้าไม่มีก็สร้างโหนดใหม่ในระดับถัดไปได้
3. ถ้ามีให้สร้างโหนดที่ถูกชี้โดยอาร์กขึ้นมาเพิ่มเติมเข้าไปใหม่



รูปที่ 2.4 ก โครงสร้างข้อมูลแบบกราฟ

รูปที่ 2.4 ข โครงสร้างข้อมูลแบบต้นไม้

รูปที่ 2.4 การเปลี่ยนโครงสร้างข้อมูลจากแบบกราฟเป็นแบบต้นไม้

2.4 การค้นหาข้อขัดข้อง

2.4.1 ชนิดของการทดสอบวงจรโลจิก[12]

การทดสอบวงจรดิจิตอล แบ่งการทดสอบออกเป็นการทดสอบ " ฟังก์ชั่น ", "พาราเมตริก" หรือ "ไดนามิก" โดยที่การทดสอบแบบฟังก์ชั่นเป็นการตรวจสอบว่า อุปกรณ์ทำงานตรงตามตารางความจริงหรือไม่ การทดสอบแบบพาราเมตริกเป็นการวัดค่าอะนาลอกพารามิเตอร์ต่างๆ เช่น ค่าแรงดัน, กระแส ว่ามีค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลาดเคลื่อนไปหรือไม่ การทดสอบแบบไดนามิกเป็นการตรวจสอบความถูกต้องของแผนผังเวลาการทำงานของอุปกรณ์ การทดสอบทั้ง 4 แบบนี้สามารถแสดงให้เห็นข้อแตกต่างได้ดังตารางที่ 2.1

	สแตติก - ฟังก์ชั่น	ไดนามิก - ฟังก์ชั่น
ฟังก์ชั่น	ตรวจสอบ : ฟังก์ชั่น หรือ ตารางความจริง ตอบคำถาม : วงจรทำงานปกติหรือไม่ ระดับการใช้ : แผงวงจร	ตรวจสอบ : clock rate ตอบคำถาม : สัญญาณเร็วขนาดนี้วงจรทำงานหรือไม่ ? ระดับการใช้ : ไอซี หน่วยความจำ, ไมโครโปรเซสเซอร์, MOS LSI อื่นๆ...
	สแตติก - พารามเตริก	ไดนามิก - พารามเตริก
พารามเตริก	ตรวจสอบ : พารามเตริก หรือ DC ตอบคำถาม : ระดับแรงดัน / กระแส มีค่าเท่าไร ระดับการใช้ : ตัวอุปกรณ์ ไอซี	ตรวจสอบ : ไดนามิก หรือ AC ตอบคำถาม : ช่วงเวลา ขอบขาขึ้น/ลง การหน่วงเวลา มีค่าเท่าไร ระดับการใช้ : ตัวอุปกรณ์ ไอซี

ตารางที่ 2.1 ชนิดของการทดสอบวงจรโลจิก

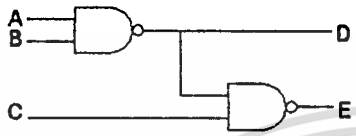
การทดสอบแบบสแตติกเป็นการทดสอบ ณ ที่การเปลี่ยนแปลงสัญญาณมีอัตราความเร็วต่ำๆ เมื่อเทียบกับความเร็วในการทำงานของอุปกรณ์ ส่วนการทดสอบแบบไดนามิกเป็นการทดสอบ ณ ที่การเปลี่ยนแปลงสัญญาณมีอัตราความเร็วสูงๆเข้าใกล้ความเร็วในการทำงานของอุปกรณ์ ตัวอย่าง เช่น การทดสอบที่ความเร็ว 2 MHz จะถือว่าเป็นการทดสอบแบบสแตติกของวงจร ECL และจะถือว่าเป็นการทดสอบแบบไดนามิก ของวงจร MOS ในทำนองเดียวกันนี้เครื่องมือทดสอบ ที่สามารถตรวจจับช่วงเวลาการเปลี่ยนแปลงของสัญญาณได้ขนาด 500 นาโนวินาที จะถือว่าเป็นเครื่องมือทดสอบแบบไดนามิกของวงจร MOS แต่จะเป็นเครื่องมือทดสอบแบบสแตติกของวงจร TTL การทดสอบแบบฟังก์ชั่นเป็นการทดสอบความถูกต้องในหน้าที่การทำงานของอุปกรณ์ ส่วนการทดสอบแบบพารามเตริกเป็นการวัดคุณสมบัติเฉพาะของพารามิเตอร์ต่างๆของอุปกรณ์

ในการทดลองโดยใช้เครื่องวิเคราะห์โลจิกเป็นหลักในการจับสัญญาณ และใช้ VP-Expert เป็นเครื่องพัฒนาระบบผู้เชี่ยวชาญ ปรากฏว่าเนื่องจากขีดความสามารถในการแสดงผลของ VP-Expert ดังนั้นจึงทำการทดสอบได้ 3 แบบ คือ แบบสแตติก-ฟังก์ชั่น, แบบสแตติก-พารามเตริก และ แบบไดนามิกฟังก์ชั่น และในงานวิจัยนี้จะใช้เพียง 2 แบบ คือ แบบสแตติก-ฟังก์ชั่น และ แบบสแตติก-พารามเตริก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) การทดสอบแบบสแตติก - ฟังก์ชั่น

การทดสอบแบบสแตติกฟังก์ชั่นหรือการทดสอบตารางความจริง เป็นการตรวจจับสัญญาณ อินพุต หรือเอาพุต ตามจุดต่างๆ เพื่อตรวจดูว่า ตรงกับตารางความจริงหรือไม่ ตัวอย่างเช่น มี NAND gate อยู่ 2 ตัว ดังในรูปที่ 2.5 สามารถที่จะทำการทดสอบด้วยการป้อนสัญญาณอินพุต high (1) และ low (0) ที่จุด A,B และ C



A	B	C	D	E
0	1	0	1	1
1	0	1	1	0
1	1	1	0	1

รูปที่ 2.5 ตัวอย่างการทดสอบแบบฟังก์ชั่น

แล้วสังเกตสัญญาณ เอาพุต ที่จุด D และ E การทดสอบแบบสแตติกฟังก์ชั่นเป็นการตอบคำถามที่ว่า "วงจรนี้ทำงานปกติหรือไม่ ? " ซึ่งนิยมใช้กับการทดสอบในระดับแผงวงจร ทั้งนี้เพราะ

1. ตัวอุปกรณ์ทั้งหลาย มักได้รับการสุ่มทดสอบมาแล้วจากโรงงานผู้ผลิต ดังนั้นเมื่อเกิดข้อขัดข้องจึงมักเป็นข้อขัดข้องชนิดที่ตรวจสอบหรือสังเกตได้โดยการทดสอบแบบสแตติกฟังก์ชั่น เช่น สายวงจรขาด / ลัดวงจร ใช้อุปกรณ์ผิดเบอร์ เป็นต้น

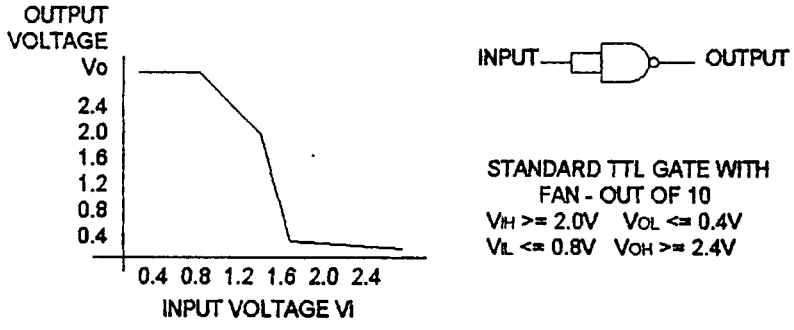
2. ตัวอุปกรณ์ทั้งหลายที่นำมาประกอบเป็นวงจร ส่วนใหญ่ไม่ได้ถูกนำมาใช้งาน ณ จุดขีดจำกัดสูงสุด / ต่ำสุด ดังนั้นถึงแม้ว่าภายในวงจรอุปกรณ์เหล่านั้นจะได้รับค่าพารามิเตอร์คลาดเคลื่อนไปบ้าง แต่ผลที่เกิดขึ้นก็จะไม่กระทบกระเทือนต่อฟังก์ชั่นการทำงานของวงจร นอกเสียจากว่าอุปกรณ์เหล่านั้นจะได้รับค่าพารามิเตอร์ที่คลาดเคลื่อนหรือผิดพลาดอย่างมหันต์จนทำงานผิดพลาดไป หรือ ไม่ทำงาน ซึ่งค่าพารามิเตอร์ที่ผิดพลาดเหล่านั้นสามารถที่จะตรวจสอบได้โดยการทดสอบแบบสแตติก ฟังก์ชั่น

3. ข้อขัดข้องส่วนใหญ่บนแผงวงจรมักจะเกิดจาก ความผิดพลาดของเจ้าหน้าที่ในการปฏิบัติงาน เช่น การบัดกรีที่ทำให้เกิดการลัดวงจรหรือเกิดการหลวมหลุด, การใช้ตัวอุปกรณ์ผิดเบอร์, การใส่ขาอุปกรณ์ผิดตำแหน่ง เหล่านี้เป็นต้น ซึ่งสามารถตรวจพบได้ในระดับสแตติกฟังก์ชั่น

(2) การทดสอบแบบสแตติก - พาราเมตริก

การทดสอบแบบสแตติก - พาราเมตริกเป็นการวัดค่า d.c. พารามิเตอร์ต่างๆ เช่น แรงดัน thresholds, กระแส source / sink, การสูญเสียพลังงาน เป็นต้น ตัวอย่าง เช่น ในรูปที่ 2.6 เป็นการแสดงคุณลักษณะของ TTL gate การทดสอบแบบสแตติกพาราเมตริกเป็นการตอบคำถามที่ว่า "ระดับ แรงดัน/กระแส มีค่าเท่าใด ?" หรือคำถามที่ว่า "ค่าระดับของแรงดัน/กระแส ขนาดนี้ถูกต้องตามคุณลักษณะเฉพาะของตัวอุปกรณ์หรือไม่ ?" การทดสอบแบบสแตติก พาราเมตริกนิยมใช้กับการทดสอบระดับตัวอุปกรณ์มากกว่าในระดับแผงวงจร ทั้งนี้เนื่องจากเป็นการสิ้นเปลืองมาก ถ้าจะทำการทดสอบ ไอซี หรือ ตัวอุปกรณ์ทุกตัวทุกขา บนแผงวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ตัวอย่างการทดสอบแบบสแตติก พาราเมตริก

(3) การทดสอบแบบไดนามิก - ฟังก์ชั่น

การทดสอบแบบไดนามิกฟังก์ชั่นเป็นการทดสอบความถูกต้องในขณะทำงานของอุปกรณ์ มักเรียกว่า การทดสอบแบบ "clock- rate" ทั้งนี้เนื่องจากเป็นการทดสอบที่ การเปลี่ยนแปลงของสัญญาณมีอัตราความเร็วเท่ากับระดับความเร็วการทำงานของอุปกรณ์นั่นเอง ในการทดสอบอาจจะใช้ระดับความเร็วสัญญาณที่สูงกว่า หรือ ต่ำกว่า ระดับความเร็วในการทำงานปกติของตัวอุปกรณ์ก็ได้ เพื่อเป็นการตอบคำถามที่ว่า "วงจรจะทำงานหรือไม่ที่ความเร็วขนาดนี้ ?" โดยเฉพาะกับวงจรซิงโครนัสโลจิกที่มีสัญญาณนาฬิกา เป็นตัวให้จังหวะ นิยมใช้กับ การทดสอบหน่วยความจำแบบสารกึ่งตัวนำ, ไมโครโปรเซสเซอร์, ชิปรีจิสเตอร์และวงจร MOS - LSI อื่นๆ การทดสอบแบบไดนามิกฟังก์ชั่น สามารถที่จะตรวจสอบข้อขัดข้องของแผงวงจรได้หลายแบบ ที่การทดสอบแบบสแตติกฟังก์ชั่นทดสอบไม่ได้ ตัวอย่าง เช่น ข้อขัดข้องที่เกิดจากตัวอุปกรณ์ในวงจร one- short, ข้อขัดข้องที่เกิดจาก pull-up resistors ขาด เป็นต้น ดังตัวอย่างในรูปที่ 2.7 เป็นแสดงข้อผิดพลาดที่เกิดจาก pull-up resistors ของวงจรโลจิกแบบ open - collector ขาด

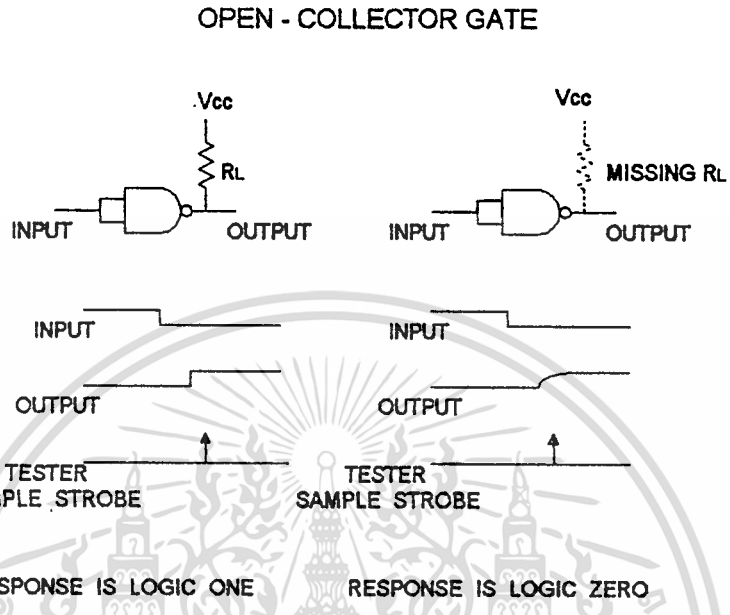
ถึงแม้ว่าในงานวิจัยนี้จะยังไม่ได้ทำการทดสอบแบบนี้ก็ตาม แต่ที่ได้กล่าวนำไว้ ก็เพื่อเป็นแนวทางในการปรับปรุงสำหรับการวิจัยครั้งต่อไป

(4) การทดสอบแบบไดนามิก - พาราเมตริก

การทดสอบแบบไดนามิก - พาราเมตริกปืนการวัดค่าคุณลักษณะต่างๆของแผนผังเวลา เช่นช่วงขอบขาขึ้น ช่วงขอบขาลง, การหน่วงเวลา, ช่วงความยาวของ pulse และ พารามิเตอร์การสวิชชิ่งของแผนผังเวลาต่างๆ เป็นต้น ดังตัวอย่างในรูปที่ 2.8 การทดสอบแบบนี้เป็นการตอบคำถามที่ว่า " ช่วงเวลาขอบขาขึ้น / ลง ใช้เวลานานเท่าไร ?" หรือคำถามที่ว่า " พารามิเตอร์การสวิชชิ่งของแผนผังเวลาถูกต้องตามคุณลักษณะเฉพาะของอุปกรณ์หรือไม่ ? " การทดสอบแบบนี้แทบไม่มีใช้กับการทดสอบระดับแผงวงจรเลย เนื่องจากว่าในระดับแผง

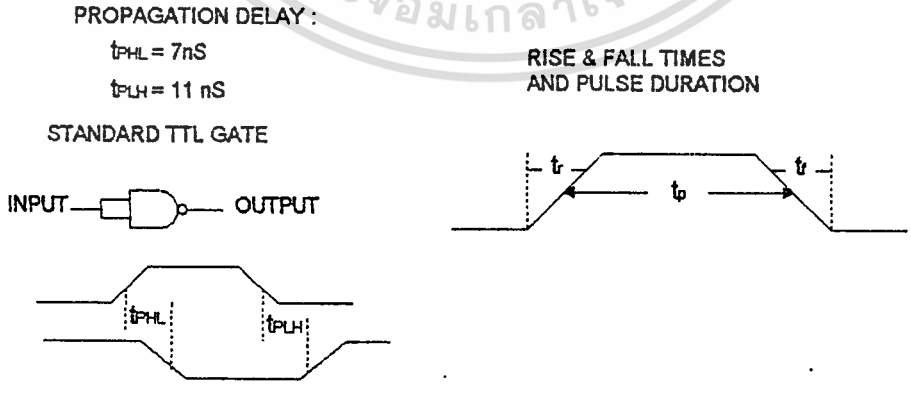
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจร ข้อผิดพลาดเกือบทั้งหมดสามารถตรวจจับได้โดยการทดสอบแบบสแตติกฟังก์ชั่น ดังนั้นการทดสอบแบบนี้จึงนิยมใช้กับการทดสอบในระดับตัวไอซี หรือ ตัวอุปกรณ์



รูปที่ 2.7 ตัวอย่างการทดสอบแบบไดนามิกฟังก์ชั่น

อนึ่งถึงแม้ว่าระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องที่ประกอบด้วยเครื่องวิเคราะห์โลจิกและเครื่องมือพัฒนา VP-Expert จะยังไม่สามารถทำการทดสอบแบบไดนามิก - พารามเมตริกได้ เนื่องจากขีดความสามารถในการแสดงผลของ VP-Expert ก็ตาม แต่ที่ได้กล่าวนำไว้ ก็เพื่อเป็นแนวทางสำหรับการวิจัยครั้งต่อๆไป



รูปที่ 2.8 ตัวอย่างการทดสอบแบบไดนามิก พารามเมตริก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2. การแบ่งระดับของการค้นหาข้อขัดข้อง

ในการค้นหาอาการชำรุดบนแผงวงจรเชิงดิจิตอล ก็มีลักษณะเช่นเดียวกับการค้นหาอาการชำรุดของวงจรทั่วๆไป กล่าวคือจะต้องมีการตั้งสมมติฐานก่อนว่าลักษณะอาการเช่นนี้ส่วนที่ชำรุด ควรจะอยู่ ณ ที่บริเวณใดของระบบ และอาจจะต้องมีการทดสอบสมมติฐานด้วย โดยการทดสอบ(test) จุดที่สำคัญ 2 - 3 จุดเพื่อยืนยันสมมติฐาน จากนั้นจึงทำการแบ่งอาณาบริเวณที่ต้องสงสัยให้แยกย่อยลงไปอีก แล้วทำการทดสอบ (test) เพื่อยืนยันสมมติฐานใหม่ กระทำซ้ำเช่นนี้เรื่อยๆไป จาก ระบบ - แผงวงจรใหญ่ - แผงวงจรย่อย - ชิป, ตัวอุปกรณ์ ในการทดสอบแต่ละระดับ ยิ่งระดับต่ำลงการทดสอบก็ยิ่งละเอียดมากขึ้น กระทำหลายจุดมากยิ่งขึ้น บางครั้งบริเวณที่ต้องสงสัยมีสัญญาณที่ผิดปกติมากมายหลายจุด ก็ต้องจัดลำดับความสำคัญของสัญญาณเหล่านั้นว่าสัญญาณใดเป็นสาเหตุที่สำคัญสูงสุด แล้วพิจารณาจากสัญญาณนั้นก่อน ซึ่งแต่ละแผงวงจร ก็แตกต่างกันไปตามลักษณะและความซับซ้อนของวงจร ตัวอย่างเช่น บนแผงวงจรหลักไมโครคอมพิวเตอร์ PC/XT ที่ใช้ทดลองมี ไมโครโปรเซสเซอร์ 8088 ลำดับความสำคัญของสัญญานที่จะทำให้ 8088 ทำงานคือ VCC, CLK88, RESET, READY... เป็นต้น

2.4.3 การค้นหาสาเหตุข้อขัดข้องทั่วไป

การค้นหาอาการชำรุดบนแผงวงจรดิจิตอล ก็มีหลักการทดสอบ (test) เช่นเดียวกันกับหลักการทดสอบอุปกรณ์โดยทั่วไป ซึ่งประกอบไปด้วย 3 ขั้นตอนคือ

(1) การป้อนแบบสัญญาณทดสอบ

(2) การจับสัญญาณตอบสนอง

(3) ทำการเปรียบเทียบสัญญาณตอบสนองกับสัญญาณที่คาดหวัง

แต่ละขั้นตอนอธิบายได้ดังนี้

(1) การป้อนแบบสัญญาณทดสอบ

จุดมุ่งหมายของการค้นหาข้อขัดข้อง ก็เพื่อต้องการที่จะตรวจสอบว่าวงจรนั้นมีอาการชำรุดอย่างไรและต้องการที่จะแยกวงจรว่า วงจรส่วนใดชำรุด วงจรส่วนใดปกติ ในการดำเนินการ ก่อนอื่นจะต้องทำให้วงจรอยู่ในสถานะที่กำหนด หรือทราบค่าเสียก่อน กรรมวิธีเช่นนี้เรียกว่า " การแรกเริ่ม[12](initialization) " ซึ่งมีเทคนิคอยู่ 3 วิธีดังนี้

ก. ป้อนสัญญาณโลจิกเข้าที่สายสัญญาณ รีเซต โดยตรงเพื่อให้วงจรเข้าสู่สถานะแรกเริ่ม

ข. ป้อนสัญญาณ ซีเคว้น อินพุต เฉพาะอย่างเพื่อบังคับให้วงจรอยู่ในสถานะที่ทราบค่า

ค. เฝ้าสังเกตดูสัญญาณ เอาต์พุต ว่าการเปลี่ยนแปลงสถานะมาถึง จุดที่รอคอยหรือยัง ทั้งนี้อาจจะต้องมีการป้อนสัญญาณอินพุตบางอย่างประกอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจาก " แรกเริ่ม " แล้วจึงทำการป้อนสัญญาณเพื่อทดสอบต่อไป การป้อนสัญญาณอินพุตเพื่อการทดสอบนี้มีวิธีการใหญ่ๆ 3 วิธีดังนี้

ก. ป้อนด้วยมือ เป็นการกำหนดสัญญาณอินพุตต่างๆด้วยมือ

ข. ป้อนแบบ pseudorandom เป็นการกำหนดสัญญาณอินพุตด้วยการสุ่มทางสถิติ

ค. ป้อนโดยการใช้เครื่องคอมพิวเตอร์ช่วยในการกำหนดสัญญาณอินพุตให้

สำหรับในงานวิจัยนี้ได้ทำการทดลองบนแผงวงจรหลักไมโครคอมพิวเตอร์ PC/XT และอาศัยสัญญาณ ทดสอบจากการทำงานของแผงวงจร ดังนั้นจึงต้องทำการป้อนสัญญาณโลจิกเข้าที่สายสัญญาณรีเซต โดยตรงแล้ว เฝ้าสังเกตดูสัญญาณเอาต์พุต ว่าการเปลี่ยนแปลงสถานะมาถึง จุดที่รอคอยหรือยัง

(2) การจับสัญญาณตอบสนอง

การจับสัญญาณตอบสนอง สามารถดำเนินการ ณ จุดใดๆก็ได้ แต่เมื่อจับแล้วต้องไม่กระทบกระเทือนการทำงานของวงจรทำให้การทำงานผิดพลาดหรือไม่ทำงาน ดังนั้นจุดที่จะจับสัญญาณตอบสนองควรมีจำนวนน้อยที่สุด แต่สามารถครอบคลุมอาการข้อขัดข้องของวงจรได้มากที่สุด โดยปกติแล้วแผงวงจรต่างๆมักจะมีจุดทดสอบ (test point) ไว้ให้ทำการตรวจสอบสัญญาณหรือข้อขัดข้องต่างๆ ดังนั้นในการพิจารณาจุดที่จะทำการจับสัญญาณตอบสนองจึงควรพิจารณาจุดทดสอบเหล่านี้ก่อนเป็นลำดับแรก หลังจากนั้นจึงพิจารณาส่วนอื่นๆของวงจร ซึ่งสามารถพิจารณาได้ดังนี้

วงจรโลจิกโดยทั่วไปประกอบด้วยส่วนที่เป็นหน่วยความจำของวงจร ซึ่งได้แก่การดงค้าง(latch)สถานะของฟลิบฟลอบต่างๆ และส่วนที่เป็นคอมบิเนชั่นเนตเวอร์ก ได้แก่ gate ต่างๆ ส่วนที่เป็นหน่วยความจำของวงจรจะถูกใช้เป็นที่เก็บข่าวสารข้อมูลของวงจรโดย

ก. ในระหว่างการทำงาน (running) จะใช้เป็นตัวกลางเก็บสถานะวงรอบการทำงานต่างๆของวงจร ซึ่งการปฏิบัติงานที่ต่างกัน จะมีสถานะต่างกัน

ข. ในระหว่างไม่ได้ทำงาน (not running) จะใช้เป็นตัวเก็บสถานะครั้งหลังสุดก่อนจบการทำงาน หรือสถานะแรกเริ่ม (initial) ก่อนเริ่มต้น (start) ทำงาน

ดังนั้นส่วนที่เป็นหน่วยความจำของวงจรจึงมีความเหมาะสมที่จะใช้เป็นจุดเก็บสัญญาณตอบสนอง แต่สำหรับในงานวิจัยนี้ได้แบ่งการจับสัญญาณตอบสนองออกเป็น 2 ขั้นตอนคือ ขั้นตอนแรกจะเลือกจับบางจุดตามหลักการดังกล่าวข้างต้นโดยมีจุดประสงค์เพื่อแบ่งระดับการค้นหาข้อขัดข้อง ขั้นตอนที่สองจะจับสัญญาณตอบสนองโดยละเอียดทุกจุดบนเส้นทางการค้นหาข้อขัดข้อง

(3) การเปรียบเทียบสัญญานตอบสนอง

เวลาและค่าใช้จ่ายส่วนใหญ่ของทั้งหมดในการซ่อมบำรุง คือ การตรวจสอบว่าวงจรนั้นมีอาการชำรุดอย่างไร และจะแยกวงจรว่า ส่วนใดชำรุด ส่วนใดปกติได้อย่างไร หรือ การค้นหาสาเหตุข้อขัดข้องนั่นเอง ในการแยกวงจรว่า ส่วนใดชำรุด ส่วนใดปกตินั้นมีเทคนิคมากมายหลายวิธี แต่ส่วนใหญ่แล้วมีพื้นฐานมาจาก การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบสัญญาณตอบสนองที่จับได้กับสัญญาณอ้างอิง ซึ่งทำให้การแยกวงจรส่วนชำรุดออกจากส่วนปกติมี 2 วิธีใหญ่ๆ คือ

ก. fault dictionary สร้างขึ้นโดยการรวบรวมอาการเสียที่เป็นไปได้ทั้งหมด แล้วบันทึกไว้ในฐานความรู้ การอนุมานใช้ความน่าจะเป็นของสาเหตุ

ข. operater probing สร้างขึ้นโดยอาศัยหลักการ การติดตามสัญญาณที่ไม่ถูกต้อง แล้วสืบเสาะค้นหาย้อนกลับไปสู่สาเหตุที่แท้จริงของมัน

และการเปรียบเทียบกับค่าที่คาดหวัง ก็สามารถทำได้ 2 วิธีใหญ่ๆด้วยกันคือ

ก. เปรียบเทียบกับค่าที่เก็บไว้ใน ฐานความรู้ ซึ่งค่าตอบสนองแบบนี้อาจจะได้มาจากการคำนวณแล้วบันทึกคำตอบที่ได้ไว้เลย หรือ วัดจากแผงวงจรที่ดีแล้วบันทึกเป็นคำตอบเข้าไว้ คำตอบเหล่านี้อาจจะอยู่ในรูปของ แผนผังเวลา หรือ ลายเซ็น (signature) ก็ได้ ดังนั้นภายในฐานความรู้ จึงประกอบไปด้วยแฟ้มข้อมูลต่างๆที่เก็บข้อมูล แผนผังเวลา หรือ ลายเซ็น ของโหนดต่างๆ ณ.ห้วงเวลาต่างๆ

ข. เปรียบเทียบกับค่าที่ได้จากการประเมินผล (process) ในฐานความรู้ ก่อนที่จะได้ค่าสัญญาณตอบสนองแบบนี้ออกมาช่างซ่อมหรือเจ้าหน้าที่เทคนิค จะต้องบรรยาย(describe)รายละเอียดการจัดต่อวงจรขึ้นมาก่อน เพื่อนำไปคำนวณค่าเอาต์พุตของฟังก์ชั่นออกมา แล้วนำไปเปรียบเทียบกับสัญญาณตอบสนองที่จับได้ ดังนั้นภายในฐานความรู้จึงประกอบไปด้วยแฟ้มข้อมูลต่างๆที่เก็บเบอร์และรายละเอียดต่างๆของ ไอซี และอุปกรณ์ต่างๆ, ตารางโลจิกต่างๆ, สมการเปลี่ยนสถานะต่วงๆเป็นต้น

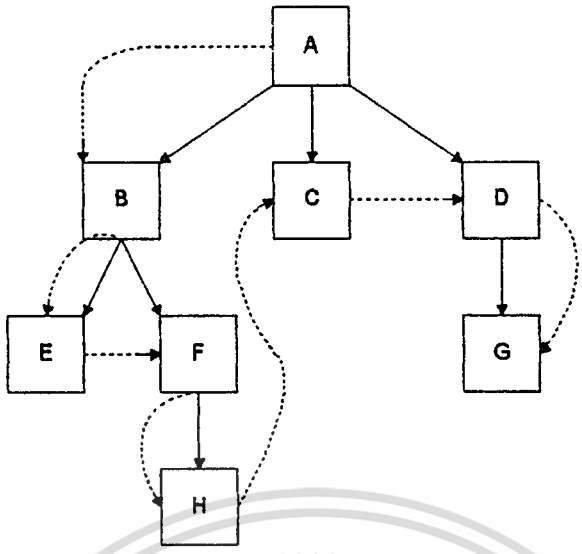
สำหรับงานวิจัยนี้ใช้การแยกส่วนชำรุดออกจากส่วนปกติโดยวิธี operator probing โดยการเปรียบเทียบค่าที่โพรบได้ กับค่าที่เก็บอยู่ในฐานความรู้

2.4.4 การค้นหาแบบฮิวริสติก

หลังจากที่ได้ทำการวิเคราะห์ปัญหาในแง่มุมต่างๆแล้ว สิ่งที่จะได้คือคุณลักษณะและโครงสร้างของปัญหา ซึ่งใช้เป็นการกำหนดทิศทางการค้นหาข้อมูลความรู้ของระบบผู้เชี่ยวชาญ เทคนิคที่จะช่วยให้การค้นหา ไปถึงคำตอบได้อย่างรวดเร็วและเหมาะสมเรียกว่า เทคนิคฮิวริสติก ซึ่งเป็นเทคนิคที่ช่วยทำให้การค้นหาดำเนินการอย่างมีภูมิ โดยจะทำการตรวจสอบข้อมูลเฉพาะที่เป็นไปได้เท่านั้น วิธีการดังกล่าวอาศัยฮิวริสติกฟังก์ชั่น กำกับทิศทางกระบวนการค้นหา ฮิวริสติกฟังชั่นจะทำหน้าที่ในการวัดขนาดความเป็นไปได้ในการแก้ปัญหา แล้วบอกว่าควรจะเลือกทิศทางใดจึงจะได้ประโยชน์มากที่สุด ดังนั้นในการแก้ไขปัญหาหนึ่งๆจะดีหรือไม่นั้นจึงขึ้นอยู่กับฮิวริสติกฟังก์ชั่นด้วย วิธีการต่างๆของการค้นหาแบบฮิวริสติกมีหลายแบบ ดังตัวอย่างเช่น

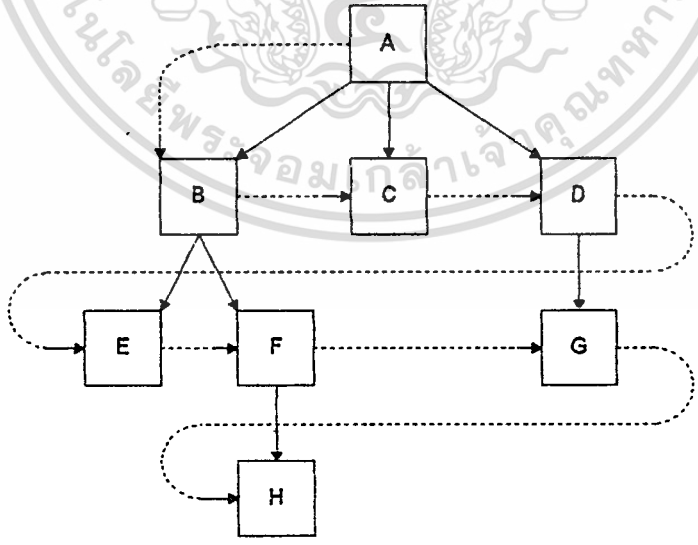
(1) การค้นหาแบบทางลึกก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 การค้นหาแบบทางลึกก่อน

เป็นการค้นหาที่อาศัยกฎเกณฑ์ที่ว่า จากโหนดในโครงสร้างต้นไม้ เมื่อทำการค้นหาให้เลือกอาร์กที่อยู่ทางซ้ายสุดก่อนแล้วเลื่อนไปยังโหนดที่อยู่ใต้ไปหนึ่งระดับ ทำเช่นนี้จนกว่าจะพบโหนดที่เป็นคำตอบ หรือลงไปจนถึงระดับล่างสุด แล้วจึงย้อนกลับขึ้นมา เพื่อทดลองอาร์กที่อยู่ขวามือถัดไป ทำเช่นนี้จนกว่าจะพบคำตอบ



รูปที่ 2.10 การค้นหาแบบทางกว้างก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) การค้นหาแบบทางกว้างก่อน

เป็นการค้นหาที่อาศัยกฎเกณฑ์ที่ว่า จากโหนดในโครงสร้างต้นไม้ เมื่อทำการค้นหาให้เลือกอาร์กทุกอาร์ก ที่เป็นไปได้ แล้วทำการตรวจดูว่ามีโหนดที่เป็นคำตอบหรือไม่ ถ้าเป็นก็หยุดการค้นหา แต่ถ้าไม่เป็นก็เลือกอาร์กทุกอาร์กที่อยู่ลึกลงไปอีกระดับหนึ่ง แล้วทำการตรวจสอบอีก ทำเช่นนี้จนกว่าจะพบคำตอบ

(3) การค้นหาแบบที่เหมาะสมก่อน

เป็นการค้นหาที่อาศัยคุณค่าของโหนดช่วยกำกับทิศทางการค้นหา โดยจะเอาคุณค่าของแต่ละโหนดไปผสมกับการค้นหาแบบทางกว้างก่อน ความมาก/น้อยของคุณค่าของแต่ละโหนดเป็นตัวกำหนดลำดับในการตรวจสอบในแต่ละระดับความลึก แทนที่จะเป็นลำดับจากซ้ายไปขวา หรือจากโหนดแรกไปโหนดหลังตามลำดับอย่างเดียวกับการค้นหาในทางกว้างก่อน ซึ่งในงานวิจัยนี้ได้ใช้วิธีในการค้นหาแบบที่เหมาะสมก่อน

2.5 การเรียนรู้โดยเครื่องจักร

การเรียนรู้เป็นวิธีการหรือกระบวนการเพื่อเพิ่มเติม หรือเพื่อปรับปรุงแก้ไขความรู้เดิม หลักการของการเรียนรู้อาศัยกระบวนการที่สำคัญ 2 อย่างคือ กระบวนการในการดึงความรู้เพื่อใช้ในการแก้ปัญหา และกระบวนการในการแก้ไขความรู้เพื่อการทำงานที่ดีขึ้น คอมพิวเตอร์ที่ใช้เทคนิคการเรียนรู้ในการแก้ปัญหาจะมีลักษณะดังนี้ คือ

1. สามารถเก็บสถานการณ์หรือตัวอย่างที่จะนำมาเปรียบเทียบเพื่อช่วยในการตัดสินใจ
2. สามารถประเมินหรือคำนวณค่า หรือหาข้อสรุปของผลที่เกิดจากการตัดสินใจเหล่านั้นได้
3. สามารถนำค่าหรือข้อมูลในข้อ 2 มาใช้ให้เป็นประโยชน์ หรือปรับแต่งให้เข้ากับสถานการณ์ใหม่ที่คล้ายคลึงกับสถานการณ์เดิม

สำหรับโมเดลการเรียนรู้ที่สำคัญได้แก่

1. การเรียนรู้จากลักษณะทั่วไปของกลุ่มตัวอย่าง (Inductive learning)
2. การเรียนรู้จากการเปรียบเทียบความเหมือน (Analogy learning)
3. การเรียนรู้จากผลลัพธ์ของปัญหาที่เคยเกิดขึ้นมาก่อน (Case-base Reasoning)
4. การเรียนรู้จากประสบการณ์ (Experience-base learning)

สำหรับในงานวิจัยนี้ใช้โมเดลการเรียนรู้จากประสบการณ์ และได้วางแนวทางเพื่อปรับปรุงการเรียนรู้ให้เป็นไปโดยอัตโนมัติ ตามลักษณะทั้ง 3 ข้อที่กล่าวมา เพื่อการวิจัยในครั้งต่อไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ปัญหาและข้อมูล

3.1 ลักษณะทั่วไปของปัญหา

แผงวงจรหลักไมโครคอมพิวเตอร์เป็นแผงวงจรดิจิตอลชนิดหนึ่ง มีสาเหตุข้อขัดข้องแบ่งออกเป็น 2 ลักษณะใหญ่ๆคือ

1. สาเหตุที่เกิดจาก ข้อมูลหรือโปรแกรม
2. สาเหตุที่เกิดจาก อุปกรณ์บนแผงวงจร

สาเหตุข้อขัดข้องที่เกิดจากตัวอุปกรณ์เป็นปัญหาทางกายภาพที่เกิดขึ้นตามสภาพการใช้งานของแผงวงจร เช่นการชำรุด หรือเสื่อมสภาพของอุปกรณ์ การเชื่อมต่อของสายสัญญาณเปลี่ยนแปลง เช่น ขาด หรือลัดวงจร สภาพแวดล้อมที่ไม่เหมาะสม เช่น ความชื้น อุณหภูมิ การสั่นสะเทือน ตลอดจนสัญญาณรบกวนต่างๆ เป็นต้น ปัญหาข้อขัดข้องเหล่านี้ แบ่งตามห้วงเวลาที่เกิดขึ้นได้เป็น

ก. ปัญหาข้อขัดข้องแบบถาวร ปัญหาชนิดนี้เมื่อเกิดขึ้นแล้วจะคงสภาพของปัญหาตลอดไป จนกว่าจะได้รับการแก้ไข เช่น ตัวอุปกรณ์ชำรุด สายสัญญาณขาด เป็นต้น

ข. ปัญหาข้อขัดข้องแบบภายในห้วงเวลาหนึ่ง เช่น การทำงานที่ผิดพลาดของอุปกรณ์เมื่อได้รับการสะสมอุณหภูมิที่สูงขึ้นๆ

ค. ปัญหาข้อขัดข้องแบบชั่ววูบ (transient) เช่นสัญญาณรบกวนช่วงสั้นๆจากสิ่งแวดล้อม

เมื่อเกิดปัญหาทางกายภาพขึ้นบนแผงวงจร ปัญหานั้นก็อาจจะส่งผลกระทบไปยังฟังก์ชั่นการทำงานของวงจร หรืออาจจะไม่มีผลกระทบต่อฟังก์ชั่นการทำงานใดๆของแผงวงจรเลย ขึ้นอยู่กับว่าทิศทางและระยะที่ผลของปัญหาจะส่งผ่าน (propagate) ไปถึง อย่างไรก็ตามส่วนใหญ่ของปัญหาข้อขัดข้องทางกายภาพเหล่านี้สามารถที่จะแปลงไปเป็นปัญหาข้อขัดข้องเชิงโลจิกได้ ซึ่งจะได้กล่าวในหัวข้อต่อไป

3.1.1 โมเดลข้อขัดข้องเชิงโลจิก

โมเดลข้อขัดข้องเชิงโลจิกมีความเหมาะสมที่จะเป็นตัวแทนการแสดงข้อขัดข้องทางกายภาพของระบบ เนื่องจาก

1. ในการวิเคราะห์ปัญหาข้อขัดข้อง โมเดลข้อขัดข้องเชิงโลจิกสามารถที่จะลดความซับซ้อนลงได้มาก
2. โมเดลข้อขัดข้องเชิงโลจิกบางชนิดเป็นอิสระต่อเทคโนโลยี ซึ่งหมายความว่าข้อขัดข้องแบบเดียวกันสามารถที่จะดัดแปลงไปใช้ได้กับเทคโนโลยีหลายรูปแบบ ดังนั้นแม้ว่าเทคโนโลยีจะเปลี่ยนแปลงไป แต่กรรมวิธีการทดสอบและการวินิจฉัยจึงยังคงใช้ได้เสมอ

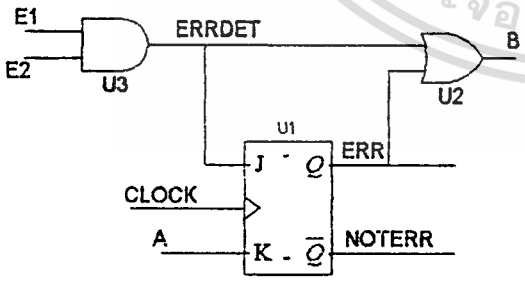
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การทดสอบสำหรับข้อขัดข้องเชิงโลจิกยังอาจจะใช้ได้กับข้อขัดข้องทางกายภาพที่ผลของข้อขัดข้องที่มีต่อวงจรไม่สามารถเข้าใจได้อย่างสมบูรณ์ หรือมีความซับซ้อนมากเกินไป

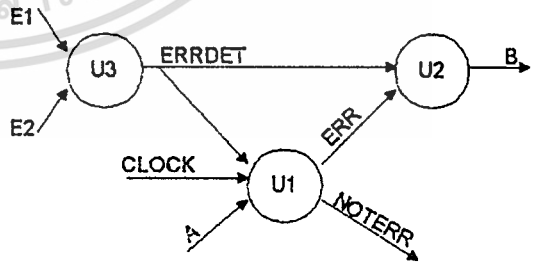
ข้อขัดข้องภายในห้วงเวลาและข้อขัดข้องแบบชั่ววูบต้องการข้อมูลทางสถิติ เพื่อสนับสนุนความน่าจะเป็นของเหตุการณ์ ข้อมูลเหล่านี้นำมาเพื่อกำหนดการทดสอบว่าจะต้องทำการทดสอบซ้ำกี่ครั้ง จึงจะทำให้ค่าความน่าจะเป็นของการพบข้อขัดข้อง(ซึ่งเกิดขึ้นไม่แน่นอน) มีค่าสูงสุด ซึ่งในงานวิจัยนี้จะยังไม่สนใจข้อขัดข้องทั้ง 2 ชนิดนี้ ถึงแม้ว่าในความเป็นจริงข้อขัดข้องที่เกิดขึ้นบนแผงวงจรอาจจะเกิดขึ้นได้คราวละมากกว่า 1 จุด แต่ในงานวิจัยนี้จะถือว่าข้อขัดข้องที่เกิดขึ้นมีเพียงจุดเดียว ทั้งนี้เนื่องจากในทุกๆกรณี ข้อขัดข้องแบบหลายจุดสามารถ ที่จะตรวจพบได้จากการค้นหาข้อขัดข้องแบบจุดเดียว แต่กระทำหลายครั้งจนพบข้อขัดข้องครบทุกจุด

3.1.2 โมเดลข้อมูลเชิงโลจิก

ระบบดิจิตอลสามารถที่จะจำลองได้ด้วยโมเดลเชิงโลจิกหลายรูปแบบ แต่ในงานวิจัยนี้จะใช้โมเดลเชิงโลจิกแบบโครงสร้าง ซึ่งเป็นโมเดลที่มองแผงวงจรเป็นเสมือนกล่องๆหนึ่ง ภายในกล่องประกอบด้วยกล่องย่อยเล็กๆมากมายเชื่อมต่อเข้าด้วยกัน กล่องย่อยเล็กๆเหล่านี้ก็คือตัวอุปกรณ์ต่างๆนั่นเอง และฟังก์ชั่นของอุปกรณ์ตัวหนึ่งๆ ก็จะถูกแสดงด้วยชนิดของอุปกรณ์ เช่น CPU, RAM, AND, OR, DECODER เป็นต้น ชนิดหรือฟังก์ชั่นต่างๆเหล่านี้ สามารถที่จะแสดงด้วยรูปร่าง สัญญลักษณ์ได้เช่นกัน ดังตัวอย่างในรูปที่ 3.1 ก ส่วนการเชื่อมต่อระหว่างกล่องก็คือสายสัญญาณต่างๆที่เข้า/ออกจากตัวอุปกรณ์ สายเหล่านี้ทำหน้าที่ในการส่งผ่านสัญญาณไปยังอุปกรณ์ต่างๆเข้าด้วยกัน ดังนั้นสายสัญญาณจึงเรียกอีกอย่างหนึ่งได้ว่า "ข่าย$^{[14]}$(net)" ถ้าข่ายใดทำการเชื่อมโยงสัญญาณจากแหล่งกำเนิดหนึ่งแห่ง ไปยังจุดหมายปลายทางหลายแห่ง เราเรียกข่ายนี้ว่าเป็นข่ายที่มี "แฟนเอา" เช่น ข่าย ERRDET ในรูปที่ 3.1 ก ข่ายเหล่านี้สามารถที่จะแทนได้ด้วย โมเดลของกราฟที่เรียกว่า directed graph หรือ diagraph ดังรูปที่ 3.1 ข



รูปที่ 3.1 ก



รูปที่ 3.1 ข

รูปที่ 3.1 การแทนข่ายสัญญาณด้วยไดกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงสถานะของสัญญาณบนข่ายใดข่ายหนึ่ง ในเวลาใดเวลาหนึ่ง เราเรียกว่า เหตุการณ์ ซึ่งเหตุการณ์หนึ่งๆก็อาจจะส่งผลกระทบต่ออุปกรณ์ตัวเดียว หรือหลายตัวก็ได้ ขึ้นอยู่กับว่าข่ายนั้น มีการเชื่อมโยงอยู่กับอุปกรณ์ต่างๆมากหรือน้อย

ปัญหาสาเหตุข้อขัดข้องที่เกิดจากอุปกรณ์แผงวงจรดิจิตอล สามารถที่เกิดขึ้นในบริเวณใดๆของตัวอุปกรณ์ก็ได้ แต่โดยทั่วไปแล้วโมเดลข้อขัดข้องแบบโครงสร้างจะถือว่าข้อขัดข้องเหล่านั้นจะต้องปรากฏอยู่ที่ขาของตัวอุปกรณ์ หรือสายส่งผ่านสัญญาณเท่านั้น รูปแบบของข้อขัดข้องที่พบเห็นบ่อยๆคือ การลัดวงจร และวงจรเปิด ตัวอย่างเช่น การลัดวงจรระหว่างสายสัญญาณกับไฟเลี้ยง หรือสายสัญญาณกับ gnd จะทำให้สายสัญญาณเส้นนั้นมีระดับแรงดันของสัญญาณค้างหรือติดอยู่ที่ระดับแรงดันทางไฟฟ้าระดับใดระดับหนึ่ง เช่น +5V หรือ 0V เป็นต้น ดังนั้นจึงเรียกข้อขัดข้องแบบวงจรเปิด หรือ ลัดวงจรลง gnd ว่าปัญหาข้อขัดข้องแบบ " ติดอยู่กับ " ระดับ... ส่วนการลัดวงจรระหว่างสายสัญญาณกับสายสัญญาณ ส่วนใหญ่มักจะก่อให้เกิดปัญหา ฟังก์ชั่นทางโลจิกใหม่ขึ้นมา เราเรียกข้อขัดข้องแบบนี้ว่า ปัญหาขอขัดข้องแบบ "บริดจ์ดิง [14](bridging)"

3.2 ลักษณะของปัญหาในงานวิจัย

ลักษณะของปัญหาเป็นปัญหาประเภทที่ต้องการแต่เฉพาะตัวคำตอบเท่านั้น ไม่ต้องการคำอธิบายกรรมวิธี หรือ เส้นทางในการค้นหาคำตอบ

3.2.1 คุณลักษณะของข้อขัดข้อง

จากลักษณะทั่วไปของปัญหา และ โมเดลเชิงโลจิกของข้อขัดข้องและข้อมูล สามารถนำมากำหนดเป็นคุณลักษณะของข้อขัดข้องในงานวิจัยได้ดังนี้

1.) เป็นข้อขัดข้องเชิงโลจิกที่เกิดจากอุปกรณ์บนแผงวงจรหลักไมโครคอมพิวเตอร์

2.) เป็นปัญหาข้อขัดข้องแบบถาวรที่มีผลกระทบต่อฟังก์ชั่นการทำงานบนแผงวงจรหลักไมโครคอมพิวเตอร์ และถ้าไม่ได้กำหนดเป็นอย่างอื่น ให้ถือว่าเป็นปัญหาข้อขัดข้องแบบ " ติดอยู่กับ " ระดับสถานะโลจิก " 1 " หรือ " 0 "

3.2.2 สถานะเริ่มต้น

กำหนดว่า ข่ายหรืออุปกรณ์ตัวใดตัวหนึ่งบนแผงวงจรหลักไมโครคอมพิวเตอร์ชำรุดทำให้แผงวงจรหลักไมโครคอมพิวเตอร์ ไม่ทำงานหรือทำงานไม่ปกติ โดยเขียนเป็นโมเดลทางคณิตศาสตร์ได้ดังนี้

ให้ Y เป็นเซตของแผนผังเวลาหรือแผนผังสถานะฟังก์ชั่นเอาพุตบนข่ายและอุปกรณ์ต่างๆของแผงวงจรที่ปกติ

ให้ X เป็นเซตของแผนผังเวลาหรือแผนผังสถานะฟังก์ชั่นเอาพุตบนข่ายและอุปกรณ์ต่างๆของแผงวงจรที่มีข้อขัดข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$X = \{ \forall x \mid x =$ แผนผังเวลา(สถานะ)ฟังก์ชั่นเอาต์พุตที่มีข้อขัดข้อง $\}$

$Y = \{ \forall y \mid y =$ แผนผังเวลา(สถานะ)ฟังก์ชั่นเอาต์พุตที่ไม่มีข้อขัดข้อง $\}$

$z =$ แผนผังเวลา(สถานะ)ที่จับได้ขณะนั้น

ดังนั้นสถานะเริ่มต้นคือ $z = x$

$$z \oplus y = 1$$

3.2.3 แนวทางแก้ไขปัญหา

การดำเนินกรรมวิธีเพื่อให้ได้มาซึ่งคำตอบโดยสรุปดังนี้ (รายละเอียดจะได้กล่าวถึงในบทที่ 4)

1.) จับสัญญาณตามจุดต่างๆแล้วเปรียบเทียบกับสัญญาณอ้างอิงจากแฟ้มข้อมูล

2.) ใช้ทิศทางการหาเหตุผลของการอนุมานแบบเดินหน้า (หัวข้อที่ 2.2.3)

3.) แยกส่วนชำรุดออกจากส่วนปกติโดยใช้วิธี operater probing (หัวข้อที่ 2.4.3)

4.) ใช้เทคนิคฮิวริสติกการค้นหาแบบที่เหมาะสมก่อน (หัวข้อที่ 2.4.4)

กำหนดให้ $n =$ หมายเลขระเบียนแฟ้มข้อมูล(record number)

$H =$ อันดับ(sequence)ของค่าฮิวริสติกฟังก์ชั่นของโหนดใดโหนดหนึ่ง

$H_n =$ ค่าฮิวริสติกฟังก์ชั่นของข้อมูลในระเบียนที่ n

$w =$ หมายเลขกลุ่มของขาอุปกรณ์ ซึ่งมีค่าตั้งแต่ 0, 1, 2....9

$p =$ ค่าที่ใช้ในการตัดสินใจเลือกเส้นทาง

$$H = \{n, H(n) \mid n = 1, 2, 3..... \}$$

$$H_n = w \div n$$

$$p = \min(H_1, H_2, H_3,.....H_n)$$

การค้นหาจะดำเนินต่อไป ตราบเท่าที่ เงื่อนไข

$$z \oplus y = 1$$

ยังเป็นจริง

3.2.4 สถานะเป้าหมาย

การดำเนินการค้นหาคำตอบจะยุติเมื่อ

1.) แผนผังเวลา(สถานะ)ฟังก์ชั่นเอาต์พุตที่จับได้ เหมือนกับ ข้อมูลอ้างอิง

2.) สิ้นสุดเส้นทางการค้นหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นสถานะเป้าหมายคือ

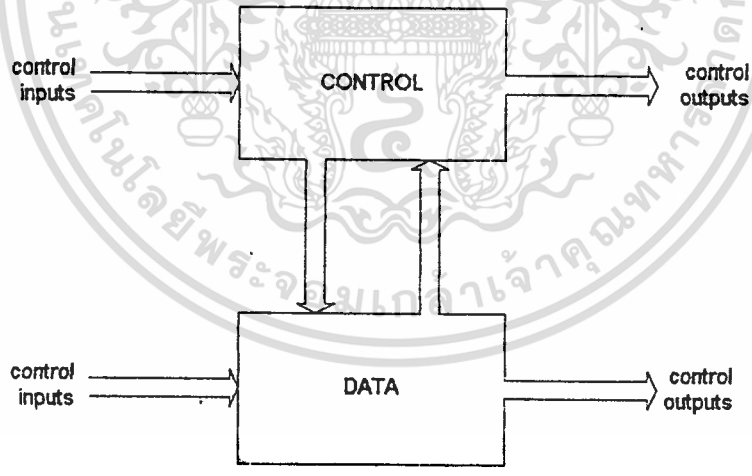
$$z \oplus y = 0$$

หรือ end of data

3.3 แฟ้มข้อมูลอ้างอิง

3.3.1 โมเดลข้อมูลของ SCIRTSS

SCIRTSS[14] (Sequential Circuit Test Search System) เป็นระบบกำเนิดสัญญาณทดสอบสำหรับค้นหาข้อขัดข้องบนวงจรซีเควนเชี่ยน ซึ่งเสนอโดย Hill และ Huey ในปี ค.ศ. 1977 ระบบนี้ถือว่าการทดสอบวงจรซีเควนเชี่ยนคือการทดสอบวงจรคอมบิเนชั่นครั้งละ 1 กรอบเวลา (time frame) หลายๆครั้งติดกัน โดยแยกวงจรส่วนอินพุตกับวงจรควบคุมออกจากกัน ดังแสดงในรูปที่ 3.2 ทำให้การดำเนินการต่อส่วนอินพุตกับส่วนควบคุมแตกต่างกัน กลไกที่ SCIRTSS ใช้ในการค้นหา คือ กำหนดว่าอินพุตซีเควนจะส่งผ่านข้อขัดข้องเข้าไปเก็บไว้ในตัวแปรสถานะก่อนที่จะส่งออกไปยังเอาพุต ดังนั้นจึงปันสัญญาณควบคุมทุกรูปแบบเข้าไปทดสอบก่อน และปล่อยให้ผู้ใช้กำหนดสัญญาณอินพุตเอาตามที่ต้องการ หรือกำเนิดสัญญาณแบบสุ่มๆขึ้นมา ด้วยกลยุทธเช่นนี้ SCIRTSS จะถือว่าสัญญาณควบคุมมีอิทธิพลต่อสถานะถัดไปของวงจร มากกว่าสัญญาณอินพุต แต่มีจำนวนสัญญาณน้อยกว่าสัญญาณอินพุต



รูปที่ 3.2 โมเดลข้อมูลของ SCIRTSS

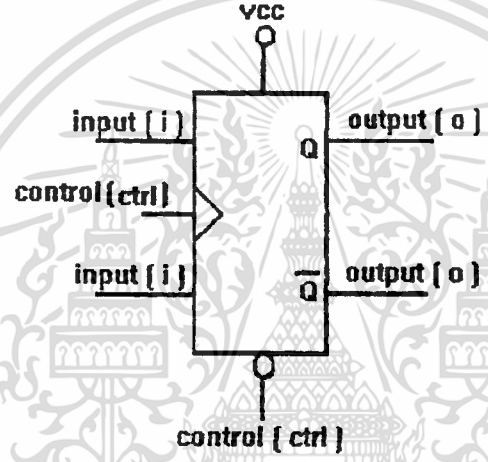
ระบบ SCIRTSS เป็นระบบกำเนิดสัญญาณทดสอบแบบอัตโนมัติที่ได้รับความนิยมมากระบบหนึ่ง เนื่องจากใช้เทคนิคการค้นหาแบบฮิวริสติกเข้ามาช่วย ซึ่งเทคนิคนี้จะได้กล่าวต่อไปในบทที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 โมเดลของตัวอุปกรณ์ในงานวิจัย

ในงานวิจัยนี้ได้ยึดถือแบบอย่างกลไกการค้นหาของ SCIRTSS ในการกำหนดโมเดลของตัวอุปกรณ์ โดยกำหนดว่าอุปกรณ์ใดๆก็ตามจะต้องมีสัญญาณอย่างน้อย 1 ชนิด ดังต่อไปนี้

1. ไฟเลี้ยงวงจรและ gnd
2. สัญญาณเอาท์พุต
3. สัญญาณควบคุม
4. สัญญาณอินพุต



รูปที่ 3.3 โมเดลของตัวอุปกรณ์ในงานวิจัย

โดยที่อุปกรณ์แต่ละตัวอาจจะมีเพียงสัญญาณเดียวหรือหลายสัญญาณก็ได้ เช่น มีสัญญาณอินพุต 2 เส้น, มีสัญญาณเอาท์พุต 2 เส้น, มีสัญญาณควบคุม 1 เส้น เป็นต้น

3.3.3 โครงสร้างแฟ้มข้อมูล

ในหัวข้อที่ 2.1.1 ได้กล่าวถึงฐานความรู้ ซึ่งประกอบด้วย

1. ส่วนที่เป็นความจริงแน่นอน เช่น กฎต่างๆ
2. ส่วนที่เป็นความจริงเฉพาะปัญหาที่กำลังเกี่ยวข้องอยู่

แฟ้มข้อมูลอ้างอิงคือส่วนที่เป็นความจริงเฉพาะปัญหาที่กำลังเกี่ยวข้องอยู่ เนื่องจากแฟ้มข้อมูลอ้างอิงเป็นที่เก็บรวบรวม แผนผังเวลา, แผนผังสถานะ, คำแนะนำ หรือข้อมูลที่จำเป็นต้องใช้ในการเปรียบเทียบกับสัญญาณตอบสนองที่จับได้จากแผงวงจรที่ขัดข้อง แฟ้มข้อมูลอ้างอิงแฟ้มหนึ่งจะบรรจุข้อมูลเฉพาะเรื่องใดเรื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนึ่ง หรือกลุ่มใดกลุ่มหนึ่งที่เกี่ยวข้องกัน เช่น แฟ้มข้อมูลอ้างอิงสำหรับทดสอบสัญญาณ DMA ก็จะประกอบไปด้วยแผนผังเวลา หรือแผนผังสถานะของสัญญาณต่างๆบนวงจร DMA และวงจร TIMER เป็นต้น

เนื่องจากแฟ้มข้อมูลอ้างอิงเป็นส่วนประกอบหนึ่งของฐานความรู้ ดังนั้นแฟ้มข้อมูลอ้างอิงนอกจากจะแสดงข้อมูลสำหรับอ้างอิงแล้ว ยังจะต้องมีขีดความสามารถในการถ่ายทอดความรู้ได้ด้วย ซึ่งเมื่อพิจารณาจากโมเดลข้อมูลเชิงโลจิกในหัวข้อที่ 3.1.2, แนวทางในการแก้ไขปัญหาในหัวข้อที่ 3.2.3 และโมเดลของตัวอุปกรณ์ในหัวข้อที่ 3.3.2 แล้ว จะเห็นว่าแฟ้มข้อมูลอ้างอิงควรจะประกอบด้วยสคมภ์สำหรับเก็บสิ่งต่างๆ ดังต่อไปนี้

1. ชื่อของตัวอุปกรณ์ต่างๆ
2. หมายเลขขาของอุปกรณ์
3. หน้าที่ของขาของอุปกรณ์แต่ละขา
4. ชื่อของข่ายต่างๆ
5. แผนผังเวลา หรือแผนผังสถานะสำหรับเปรียบเทียบ
6. ค่าฮิวริสติกฟังก์ชั่น

อนึ่งแฟ้มข้อมูลอ้างอิงต่อไปจะเรียกว่า แฟ้มข้อมูล หรือ แฟ้มความรู้

แฟ้มข้อมูลถูกสร้างขึ้นจากโปรแกรมฐานข้อมูล Foxpro, dBase หรืออื่นๆในตระกูลเดียวกัน โดยมีสคมภ์สำหรับเก็บข้อมูลต่างๆดังที่ได้กล่าวมาแล้ว ซึ่งสามารถกำหนดเป็นโครงสร้างได้ดังรูปที่ 3.4

compo	pin	func	group	t_diagm	signame

รูปที่ 3.4 โครงสร้างแฟ้มข้อมูล

สคมภ์ต่างๆของโครงสร้างแฟ้มข้อมูลในรูปที่ 3.4 มีความหมายโดยย่อดังนี้

1.) compo เป็นสคมภ์ที่ใช้เก็บชื่ออุปกรณ์ประกอบต่างๆที่อยู่บนแผงวงจร เช่น U14, R21, C16 เป็นต้น ชนิดของข้อมูลเป็นตัวอักขระ ขนาดความกว้างสคมภ์ 6 ตัวอักขระ

2.) pin เป็นสคมภ์ที่ใช้เก็บหมายเลขขาของอุปกรณ์เช่น U14pin7, U1pin8 เป็นต้น ชนิดของข้อมูลเป็นตัวอักขระ ขนาดความกว้างสคมภ์ 5 ตัวอักขระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.) func เป็นสคมภ์ที่ใช้เก็บหน้าที่ของขาแต่ละขาของอุปกรณ์นั้นๆเช่น U14pin7เป็น GND, U1pin8 เป็น clock เป็นต้น ซึ่งรายละเอียดการจัดเรียงลำดับ หน้าที่ของขาอุปกรณ์จะได้กล่าวต่อไปในหัวข้อที่ 3.3.4 ชนิดของข้อมูลเป็นตัวอักขระ ขนาดความกว้างสคมภ์ 4 ตัวอักขระ

4.) group เป็นสคมภ์ที่ใช้เก็บค่าฮิวริสติกฟังก์ชั่น ขนาดความกว้างสคมภ์ 1 หลักเลขฐานสิบซึ่งรายละเอียด จะได้กล่าวต่อไปในบทที่ 4

5.) t_diagm เป็นสคมภ์ที่ใช้เก็บแผนผังเวลาหรือแผนผังสถานะของแต่ละขาของอุปกรณ์ในห้วงเวลาที่สนใจ ตลอดจนข่าวสารข้อมูลอื่นๆที่จำเป็น ชนิดของข้อมูลเป็นตัวอักขระ ขนาดความกว้างสคมภ์ 4 ตัวอักขระ

สำหรับแผนผังเวลานั้นเป็นข้อมูลที่มีลักษณะเป็นรูปภาพ ดังนั้นจึงไม่สามารถนำเข้าโดยใช้แป้นพิมพ์ธรรมดา จึงต้องทำการแก้ไขโดยดัดแปลง รูปร่างของตัวอักขระบางตัวที่ไม่ค่อยได้ใช้หรือ ไม่ค่อยมีความจำเป็นมากนัก เช่น เลข ๐, ๑, ๓, ๕ เป็นต้น ให้เป็นรูปร่างของ แผนผังเวลาแทน เช่น

๐ = สถานะ low ..... = ‾

๓ = การเปลี่ยนสถานะจาก high เป็น Low = ⌊

๑ = สถานะ high ..... = ‾

๕ = การเปลี่ยนสถานะจาก low เป็น High = ⌈

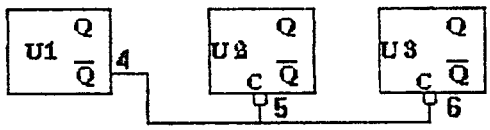
เป็นต้น ดังตัวอย่าง เช่น

๐.๕.๑.๑.๓.๐.๐.๕.๓.๕.๓.๐.๕.๑.๓

รายละเอียดจะได้กล่าวต่อไปในหัวข้อที่ 3.3.5

6.) signame เป็น สคมภ์ที่ใช้เก็บชื่อของสัญญาณ หรือ ข่ายต่างๆที่เข้ามา หรือ ออกไป จากขาของอุปกรณ์แต่ละขา เช่น READY, RD, WR, ACK, BUSY เป็นต้น

อนึ่งสำหรับขาที่ไม่ได้กำหนดชื่อสัญญาณเอาไว้ ก็จะใช้วิธีตั้งชื่อเอาเองโดยไม่ให้ซ้ำกัน หรือจะใช้วิธีเรียกชื่อจากขาอุปกรณ์ต้นทาง -ปลายทางก็ได้ เช่น



จะเขียนได้เป็น U1.4U2.5U3.6 (U1pin4_ to_ U2pin5_to_U3pin6) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4 ข้อมูลฟังก์ชั่น

(1) การกำหนดฟังก์ชั่นขาของอุปกรณ์

ตามปกติอุปกรณ์ตัวหนึ่งๆบริษัทผู้ผลิตมักจะกำหนดฟังก์ชั่นมาให้เรียบร้อยแล้ว เช่น บอกว่าขาใดเป็นอินพุตขาใดเป็นเอาท์พุต แต่ก็มีอุปกรณ์บางประเภท เช่น ตัวต้านทาน ตัวเก็บประจุ ไดโอดหรืออุปกรณ์ประเภท 2 ขาอื่นๆไม่ได้มีการกำหนดมาให้ หรือในบางครั้งจะพบว่าขาหนึ่งๆของอุปกรณ์ตัวใดตัวหนึ่งก็ทำหน้าที่หลายฟังก์ชั่น เช่น ขาคาต้าของ RAM เป็นได้ทั้งอินพุตและเอาท์พุต ดังนั้นในงานวิจัยนี้จึงได้กำหนดฟังก์ชั่นขาของอุปกณ์ดังนี้

1.) กำหนดตามบริษัทผู้ผลิต

2.) อุปกรณ์ 2 ขา เช่น ตัวต้านทาน ตัวเก็บประจุ ให้ยึดถือทิศทางการอนุมานเป็นหลัก คือ การอนุมานจะมีทิศทางย้อนจากเอาท์พุตไปหาอินพุตเสมอ

3.) ถึงแม้ว่าจะมีบางขาของอุปกรณ์ทำหน้าที่หลายฟังก์ชั่น แต่ก็ไม่ได้หมายความว่าขานั้นจะทำหน้าที่ทุกฟังก์ชั่นพร้อมกัน ดังนั้นจึงให้ยึดถือฟังก์ชั่นที่กำลังพิจารณาสาเหตุข้อขัดข้องขณะนั้นเป็นหลักเพียงฟังก์ชั่นเดียว

(2) การจัดเรียงลำดับข้อมูลตามฟังก์ชั่น

ความมุ่งหมายในการจัดเรียงข้อมูลในสคมภ์ func ก็เพื่อช่วยให้กลไกการอนุมานสามารถวิเคราะห์ค้นหาสาเหตุข้อขัดข้องของวงจรได้ซึ่งมีการจัดลำดับความสำคัญตามหน้าที่ ของแต่ละขาของอุปกรณ์ที่เป็นต้นเหตุ ให้แผงวงจรไม่ทำงานตามปกติ ตามลำดับก่อน/หลังดังนี้

1.) เอาต์พุต เช่นขา Q

2.) ไฟเลี้ยง และ gnd

3.) สัญญาณควบคุม เช่นขา clear, preset, clock, enable เป็นต้น

4.) อินพุต เช่นขา J, K, D เป็นต้น

เหตุผลที่จัดลำดับความสำคัญแบบนี้สืบเนื่องมาจากว่า ถ้ามีอุปกรณ์ตัวใดๆ ไม่ทำงาน หรือทำงานไม่ปกติ ประการแรกที่สามารถทราบได้ ก็คือการตรวจดูเอาต์พุตว่าผิดปกติหรือไม่ ถ้าปรากฏว่าเอาต์พุตไม่มี หรือมีแต่ไม่ปกติ ประการต่อไป (นอกจาก การตรวจสอบทางกายภาพแล้ว)คือตรวจดูว่ามีไฟเลี้ยงวงจรตามปกติหรือไม่ ถ้า ปรากฏว่ามีไฟเลี้ยงวงจรตามปกติ ประการต่อไปคือตรวจดูว่ามีสัญญาณควบคุม เข้ามาปกติเรียบร้อยหรือไม่ ถ้าปรากฏว่าปกติดีก็จะตรวจสอบสัญญาณอินพุตที่เข้ามาต่อไป สัญญาณ อินพุต, ควบคุม, ไฟเลี้ยงและ gnd เหล่านี้ ถ้าไม่ปกติหรือไม่มี ก็จะทำการตรวจสอบไปยังที่มาของสัญญาณเหล่านั้นต่อไป แต่ถ้าตรวจสอบแล้วปรากฏว่าสัญญาณทั้ง 4 ประเภทที่กล่าวมาแล้วข้างต้นเรียบร้อยปกติหมด เราก็สรุปได้ว่า อุปกรณ์ตัวนี้ชำรุด

ส่วนตัวอุปกรณ์ในสคมภ์ compo นั้น จะจัดเรียงก่อน หลัง อย่างไรก็ได้ ซึ่งในอุปกรณ์บางตัวอาจจะทำหน้าที่หลายๆอย่าง หรือมีหลายๆภาค อยู่ในตัวเดียวกัน ตัวอย่างเช่น TTL เบอร์ 7400 มี NAND gate 4 ตัวเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อิสระต่อกัน ดังนั้นการใส่ในช่อง(สดมภ์) compo ก็อาจจะแยกให้เห็นความแตกต่างของแต่ละตัว หรือแต่ละภาค เช่น ถ้า U5 คือ TTL 7400 ตัวที่กล่าวมาก็ควรจะแยกเป็น U5/1, U5/2, U5/3, U5/4 เป็นต้น ส่วนการเรียงนั้น จะเรียง U5/4, U5/1, U5/3, U5/2 สลับอย่างไรก็ได้

การวิเคราะห์เส้นทางการค้นหาข้อขัดข้อง ในบางครั้งจำเป็นจะต้องแบ่งตัวอุปกรณ์ออกเป็นภาคๆ เพื่อความสะดวกในการอนุมาน จึงกำหนดหลักการ การแบ่งภาคตัวอุปกรณ์ดังนี้

1.) แบ่งตามเส้นทางการค้นหาข้อขัดข้อง

2.) แบ่งตามคู่มือ

3.) แบ่งตามหน้าที่การทำงาน

(3) โปรแกรมช่วยจัดเรียงลำดับข้อมูลตามฟังก์ชั่น

ในวงจรที่มีความซับซ้อนมากขึ้น ถ้าผู้ใช้จะต้องมาคอยค้นหาและจัดเรียงข้อมูลด้วยตนเองก็จะทำให้เกิดความล่าช้า และผิดพลาดได้ง่ายเพื่อความสะดวกต่อผู้ใช้ในการสร้างและปรับปรุง (update) ฐานความรู้ ดังนั้นจึงควรมีโปรแกรมประยุกต์ (application) เพื่อทำหน้าที่ในการจัดกลุ่มและจัดเรียงลำดับขาอุปกรณ์ ดังที่ได้กล่าวมาแล้ว โดยเสนอแนวความคิดตาม ผังงาน(flowchart) ในรูปที่ 3.5 และมีรายละเอียด source code ชื่อ COMPO.PRG อยู่ในภาคผนวกที่ 9

ตามรูปที่ 3.5 อธิบายการทำงานของโปรแกรมได้ดังนี้

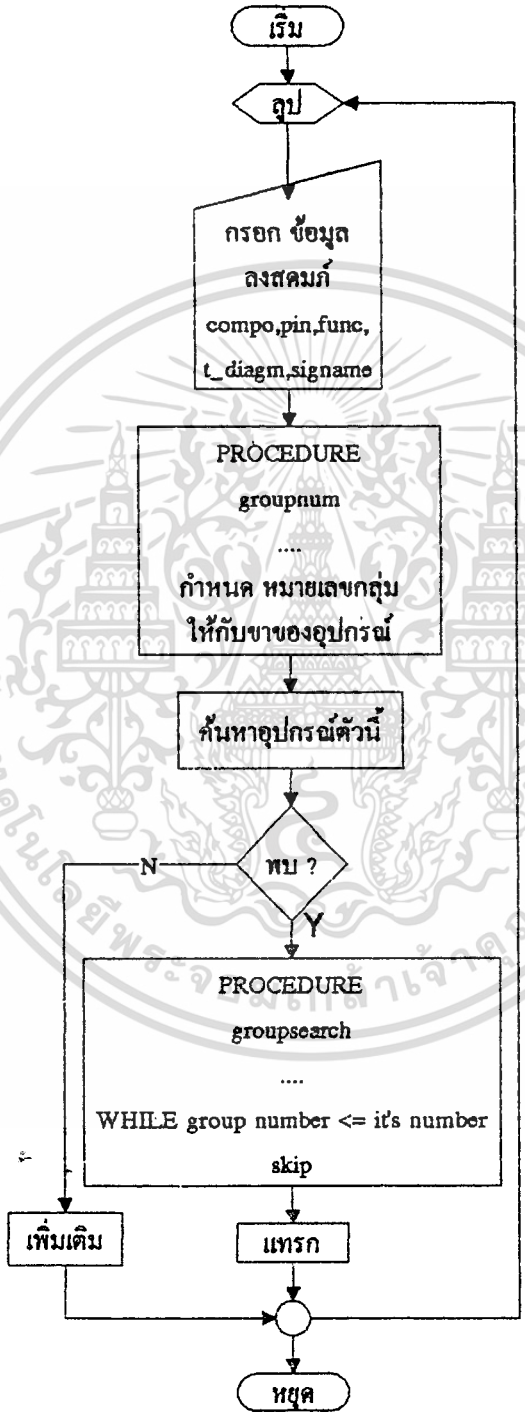
ในแต่ละวงรอบของการทำงาน (loop) ตัวโปรแกรมจะปรากฏรายการ ชื่อ, ขา, หน้าที่,ผังเวลา และชื่อของสัญญาณเพื่อให้ผู้ใช้พิมพ์เข้าไป หลังจากนั้นโปรแกรมย่อย groupnum จะดูที่หน้าที่ (function) ของขาของอุปกรณ์ขณะนี้ว่าทำหน้าที่อะไรเพื่อจะได้กำหนดหมายเลขกลุ่ม (group)ให้ ตามลำดับความเร่งด่วนซึ่งกำหนดไว้ ดังนี้

'o' หมายถึง เอาต์พุต ลำดับ 0
'vcc' หมายถึง ไฟเลี้ยง ลำดับ 2
'gnd' หมายถึง ดิน ลำดับ 2
'c' หมายถึง clear, reset ลำดับ 5
's' หมายถึง set, preset ลำดับ 6
'i' หมายถึง อินพุต ลำดับ 9

สำหรับหน้าที่ของขาของอุปกรณ์นอกจากนี้แล้ว จะถือว่าเป็นสัญญาณควบคุมทั้งสิ้น ทั้งนี้เพราะสัญญาณควบคุมมีได้หลายชื่อหลายหน้าที่ เช่น R/W, enable, clock, select, busy, ...เป็นต้น. ซึ่งผู้ใช้จะพิมพ์อะไรเข้าไปก็ได้ ตัวโปรแกรมย่อย groupnum จะใส่หน้าที่ให้เป็น 'ctrl' หมด และจัดลำดับให้อยู่ในกลุ่มที่ 8 ซึ่งอยู่ก่อนสัญญาณอินพุต (กลุ่ม 9) และอยู่หลังสัญญาณ clear, set ตามลำดับ ที่จริงแล้วสัญญาณ clear และ set ก็เป็นสัญญาณควบคุมด้วย แต่เหตุที่จัดลำดับความเร่งด่วนไว้ก่อนสัญญาณควบคุมอื่นๆ ก็เพราะว่าถ้าสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

clear หรือ set เกิดขัดข้อง เช่น ทำงานตลอดเวลา ก็จะไปบังคับให้สัญญาณควบคุม อื่นๆไม่มีผลต่อการทำงาน ดังนั้นจึงต้องจัดสัญญาณ clear, set ไว้ก่อนสัญญาณควบคุมอื่นๆดังกล่าว



รูปที่ 3.5 ผังงานของโปรแกรมจัดเรียงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่โปรแกรมย่อย groupnum ได้ทำหน้าที่กำหนดหมายเลขกลุ่มให้กับขาของอุปกรณ์แล้ว ตัวโปรแกรมหลักก็จะดำเนินการค้นหา ว่าภายในแฟ้มความรู้นี้เคยมีอุปกรณ์ที่มีชื่อเช่นนี้หรือไม่ ถ้ายังไม่มีจะถือว่าเป็นอุปกรณ์ตัวใหม่ ก็จะนำรายการที่ผู้ใช้พิมพ์เข้ามาทั้งระเบียน (record)ไปเพิ่มเติมข้างท้าย(append) แฟ้มความรู้ แล้ววนกลับไปเริ่มต้นวงรอบ (loop) ใหม่ แต่ถ้าโปรแกรมหลักดำเนินการค้นหาแล้วพบว่าภายในแฟ้มความรู้แฟ้มนี้ มีอุปกรณ์ที่มีชื่อเช่นนี้อยู่ก่อนแล้วก็จะมอบให้โปรแกรมย่อย groupsearch ดำเนินการค้นหาเพื่อที่จะสอดแทรกระเบียน (record) ที่ผู้ใช้พิมพ์เข้ามานี้ให้ถูกต้องตามลำดับความเร่งด่วนดังที่กล่าวมาแล้ว หลังจากนั้นก็จะวนกลับไปเริ่มต้นวงรอบ (loop) ใหม่

นอกจากนี้ยังมีโปรแกรมสำเร็จรูปอีกมากมายที่ช่วยอำนวยความสะดวก ในการแก้ไขปรับปรุงการจัดเรียงลำดับข้อมูลดังกล่าว ตัวอย่าง เช่น เอ็กเซลบนระบบวินโดว์

โดยปกติในการค้นหาอาการชำรุดบนแผงวงจร เรามักจะแบ่งออกเป็นภาคๆ แล้วแต่ละภาคก็ยังแบ่งออกเป็นส่วนย่อยๆลงไปอีก ดังนั้นแฟ้มข้อมูลหนึ่งๆจึงมีขนาดไม่ใหญ่มากนักสามารถที่จะใช้การค้นหาแบบเชิงเส้นได้ (linear search)

3.3.5 แผนผังเวลา

(1) รายละเอียดทั่วไปของ แผนผังเวลา

แผนผังเวลาเป็นแผนภูมิที่แสดงให้ทราบถึงการเปลี่ยนแปลงสถานะ ของสัญญาณต่อฐานเวลา ปกติแล้วสถานะของสัญญาณตามจุดต่างๆบนแผงวงจรมักจะมีการเปลี่ยนแปลง (Transition) อยู่ตลอดเวลา บางจุดก็เปลี่ยนแปลง กลับไป/กลับมา รวดเร็วมาก บางจุดก็นานๆจึงจะมีการเปลี่ยนแปลงสถานะครั้งหนึ่ง

ในการจับสัญญาณเพื่อต้องการเก็บไว้เป็นตัวอ้างอิง หรือเพื่อต้องการทดสอบก็ตาม ในงานวิจัยนี้จะใช้วิธีกำหนดเหตุการณ์ใดเหตุการณ์หนึ่งซึ่งจะใช้เป็นจุดสังเกตและติดตามค้นหาข้อขัดข้องขึ้นมาก่อน หลังจากนั้นจะเปิดให้เครื่องเริ่มต้นและทำงานไปตามปกติ แล้วจึงรอคอยเพื่อที่จะจับสัญญาณ ณ จุดที่ได้กำหนดไว้ดังกล่าวแล้ว สำหรับห้วงเวลา หรือจังหวะใดที่ควรจะจับสัญญาณ และจะจับนานเท่าใดนั้น ขึ้นอยู่กับลักษณะของวงจร, ประสบการณ์ของผู้ใช้และขีดความสามารถของเครื่องมือที่ใช้ ตัวอย่างเช่น บางจุดต้องการที่จะสังเกต แผนผังเวลา ตลอดระยะเวลา 100 มิลิวินาที บางจุดต้องการสังเกต แผนผังเวลา หลังจากเกิดเหตุการณ์ใดเหตุการณ์หนึ่งเพียงแค่ 50 นาโนวินาที หรือ 100 นาโนวินาที เป็นต้น สำหรับแผนผังเวลาเหล่านี้ ถ้าจับไว้เพื่อใช้เป็นสัญญาณอ้างอิง ก็จะต้องทำการนำสัญญาณนั้นเข้าไปเก็บในแฟ้มความรู้โดยการพิมพ์ (key) เข้าไป ซึ่งจะต้องมีการสร้างรูปร่างของอักขระขึ้นมาใหม่ เพื่อให้สามารถพิมพ์รูปร่างของ แผนผังเวลาเหล่านั้นเข้าไปได้ ซึ่งรายละเอียดจะได้กล่าวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

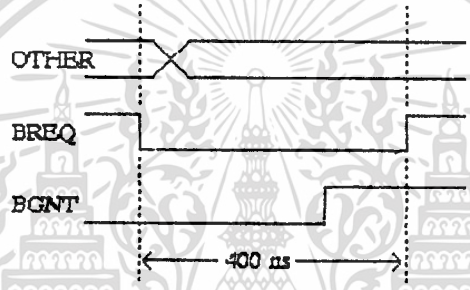
<u>ตัวอย่างที่ 3.1</u> ตัวอย่างข้อมูลแผนผังเวลา

สมมติว่าเราต้องการที่จะจับเวลาของสัญญาณตอบสนองระหว่างสองสัญญาณคือ สัญญาณ Bus Requst (BREQ) และสัญญาณ Bus Grant(BGNT) ดังรูปที่ 3.6 ก

สัญญาณที่ควรจะใช้เป็นตัวกระตุ้น (Trigger) หรือให้จังหวะให้เครื่องวิเคราะห์โลจิก เริ่มจับสัญญาณ คือ Bus Request(BREQ)

ห้วงเวลาในการจับสัญญาณควรจะยาวเท่ากับ หรือมากกว่าขนาดความกว้างของสัญญาณ Bus Request (BREQ pulse width)

เหตุการณ์ต่างๆที่กล่าวมาแล้ว เขียนเป็น แผนผังเวลา ได้ดังนี้



รูปที่ 3.6 ก

ในแฟ้มความรู้ ในช่อง (สคมภ์) สำหรับเก็บแผนผังเวลา ก็จะพิมพ์เข้าไปดังนี้

compo	pin	func	group	t_diagm	signame
.....	...	...	....	.....	.....
					BREQ
					BGNT
.....	...	...	....	.....	.....

รูปที่ 3.6 ข

รูปที่ 3.6 ข้อมูลแผนผังเวลาประกอบตัวอย่างที่ 3.1

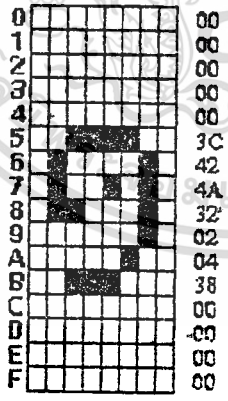
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) การสร้าง รูปร่างอักขระ ของ แผนผังเวลา

ดังที่ได้กล่าวมาแล้วว่า แผนผังเวลา เป็นแผนภูมิแสดงการเปลี่ยนแปลงของสัญญาณต่อฐานเวลา ในระบบดิจิตอลซึ่งมีเพียงแค่ 2 สถานะ คือ ลอจิก"1" และ "0" หรือ High และ Low เท่านั้น แผนผังเวลา ของระบบดิจิตอล จึงเป็นแผนภูมิแสดงระดับสัญญาณ 2 สถานะ ที่แน่นอนในห้วงเวลานั้นๆ

ดังนั้นในการสร้างรูปร่างอักขระให้สามารถพิมพ์แผนผังเวลาได้ ก็เพียงสร้างรูปร่างอักขระของสถานะ High, สถานะ Low, การเปลี่ยนแปลงสถานะจาก high ไป low และ การเปลี่ยนแปลงสถานะจาก low ไป high ก็เป็นการเพียงพอแล้ว ซึ่งมีรายละเอียดดังนี้

ใช้วิธี บิตแมพ(Bit map) ซึ่งเป็นวิธีการหนึ่งที่ใช้ในการสร้างภาพบนจอภาพ ไม่ว่าจะเป็นรูปภาพหรือตัวหนังสือ (Text) โดยที่อักขระต่างๆ (Charactor) ที่แสดงอยู่บนจอภาพของคอมพิวเตอร์ ไม่ว่าจะเป็นภาษาไทยก็ดีหรือภาษาอังกฤษก็ดี อักขระแต่ละตัวจะประกอบไปด้วยจุดต่างๆเรียงกันอยู่ในลักษณะของตาราง 2 มิติ โดยกำหนดในตาราง 2 มิติให้เป็นขาวหรือดำ เพื่อต่อกันเป็นรูปแบบของอักขระ เรียกว่า ฟอนต์ (FONT) ซึ่งขนาดของตารางของมิตินี้ อาจจะเป็น 8 x 8 , 8 x 16, 8 x 20..... ขึ้นอยู่กับความสามารถของการ์ดแสดงผล และจอภาพ จุดๆหนึ่งในตารางเหล่านี้สามารถที่จะแทนได้ด้วยข้อมูลอย่างน้อยที่สุด 1 bit ดังนั้น ตัวอักขระ 1 ตัว จึงประกอบไปด้วยข้อมูลอย่างน้อย 8 ไบต์ ตัวอย่างเช่นข้อมูลของเลข ๑ ที่เก็บอยู่ในตารางขนาด 8 x 16ในลักษณะเลขฐานสิบหก คือ 00,00,00,00,00,3C,42,4A,32,02,04,38,00,00,00,00 ซึ่งนำมาเรียงต่อกัน ก็จะได้ดังรูปข้างล่างนี้



ในการนำมาใช้กับ แฟ้มข้อมูลเพื่อให้สามารถที่จะบันทึกแผนผังเวลาตามจุดทดสอบได้ เราจึงจะต้องมีการดัดแปลงฟอนต์บางตัวที่ไม่ค่อยได้ใช้หรือไม่ค่อยมีความจำเป็นมากนัก เช่น เลข 0, ๑, ๒.....๙ ให้เป็นฟอนต์ของ แผนผังสถานะโดยกำหนดดังนี้

เลข 0 ให้เป็น ฟอนต์ ของ Low State ตามที่นิยมใช้ "0" เป็น Low State

เลข ๑ ให้เป็น ฟอนต์ ของ High State ตามที่นิยมใช้ "1" เป็น High State

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลข ๓ ให้เป็น ฟอนต์ ของ การเปลี่ยนแปลงสถานะจาก high ไป low ตามลักษณะการเขียน

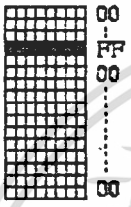
เลขสาม คือ ๓

เลข ๕ ให้เป็น ฟอนต์ ของ การเปลี่ยนแปลงสถานะจาก low ไป high ตามลักษณะการเขียน

เลขเก้า คือ ๕

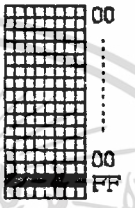
ฟอนต์ทั้งหมดนี้จะเก็บอยู่ในตารางขนาด 8 x 16 หรือ รูปร่างอักขระหนึ่งตัวใช้หน่วยความจำ 16 ไบต์ ดังแสดงอยู่ในรูปข้างล่างนี้

บิตแมพ ของ สถานะHigh



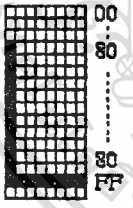
รูปที่ 3.7 ก

บิตแมพ ของ สถานะLow



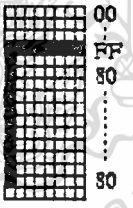
รูปที่ 3.7 ข

บิตแมพของการเปลี่ยนแปลงสถานะจากhighไปlow



รูปที่ 3.7 ค

บิตแมพของการเปลี่ยนแปลงสถานะจากlowไปhigh



รูปที่ 3.7 ง

รูปที่ 3.7 การแสดงรูปร่างอักขระของแผนผังสถานะ

ในการแทนที่ฟอนต์ทั้งหลายลงในหน่วยความจำ กำเนิดตัวอักษร (Charactor Generator) เขาจะใช้หมายเลขประจำตัวแทนฟอนต์เหล่านั้น เช่น เลข ๑ = 241 (ทั้งนี้ตามประกาศ รหัสมาตรฐานอุตสาหกรรม มอก .620 - 2529) เรียกว่ารหัสแอสกี (ASCII code) จากการที่เราได้เปลี่ยนแปลงรูปร่างของอักขระให้เป็นรูปร่างของแผนผังสถานะแทนดังที่ได้กล่าวมาแล้วข้างต้น ทำให้ สถานะเชิงโลจิก ต่างๆ มีรหัสเอสกี ดังนี้

สถานะ Low = 240

สถานะ High = 241

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงสถานะจาก high ไปเป็น low = 243

การเปลี่ยนแปลงสถานะจาก lowไป high = 249

และเพื่อให้ ฟอนต์ ต่างๆที่ได้ออกแบบมานี้สามารถแสดงออกบนหน้าจอภาพได้ ดังนั้นจึงต้องเขียนโปรแกรม และ แฟ้มข้อมูลเพิ่มเติม ดังนี้

แฟ้ม STATE. FON เป็นแฟ้มที่ใช้เก็บฟอนต์ของ แผนผังสถานะตามที่ได้ออกแบบมาแล้ว ภายในแฟ้มจะประกอบไปด้วย ตารางบิตแมพของตัวเลข 0 - ๙ ซึ่งมีรหัสแอสกี ตั้งแต่ 240 - 249 โดยที่รูปร่างของตัวเลข 0 ,๑, ๓, ๘ และ ๕ ได้ถูกเปลี่ยนไปเป็นรูปร่างของ แผนผังสถานะแทน สำหรับเลข ๕ ที่เพิ่มเติมขึ้นมานี้ใช้กับสถานะ Low เช่นเดียวกับเลข 0 เนื่องจากการใช้เลข 0 เป็น สถานะ Low เวลาใช้งานจริงอาจจะไม่ค่อยสะดวก

โปรแกรม STATEXT. EXE เป็นโปรแกรมที่เขียนด้วยภาษาซี คอมไพล์ด้วยบอร์แลนด์ซี 3.1 ทำหน้าที่ในการโหลดตารางบิตแมพเข้าไปเก็บไว้ใน RAMโดยการเรียกใช้ bios interrupt 10h function 11h[15] (charactor generation) subfunction 00h(load user text font) เมื่อเรียกใช้โปรแกรม STATEXT.EXE แล้วก็สามารถใช้ฟอนต์ที่สร้างขึ้นได้ตลอดไป จนกว่าจะปิดเครื่อง หรือเรียกโปรแกรมสำหรับยกเลิก ซึ่งจะได้กล่าวต่อไป

โปรแกรม ROMTEXT.EXE เป็นโปรแกรมที่เขียนด้วยภาษา แอสเซมบลี และ คอมไพล์ด้วยเทอร์โบแอสเซมบลี ทำหน้าที่ในการยกเลิก หรือคืนตารางบิตแมพเดิมให้กับจอภาพเพื่อทำการโหลดตารางบิตแมพ จาก ROM เข้ามาแทน โดยการเรียกใช้ bios interrupt 10h function 11h [15](charactor generator) subfunction 04h (load ROM 8x16 text font (VGA))

รายละเอียดของโปรแกรม STATEXT.C , ROMTEXT.ASM และแฟ้ม STATE.FON อยู่ในภาคผนวกที่ 10 และ 14 ตามลำดับ

3.3.6 ตัวอย่างแฟ้มข้อมูล

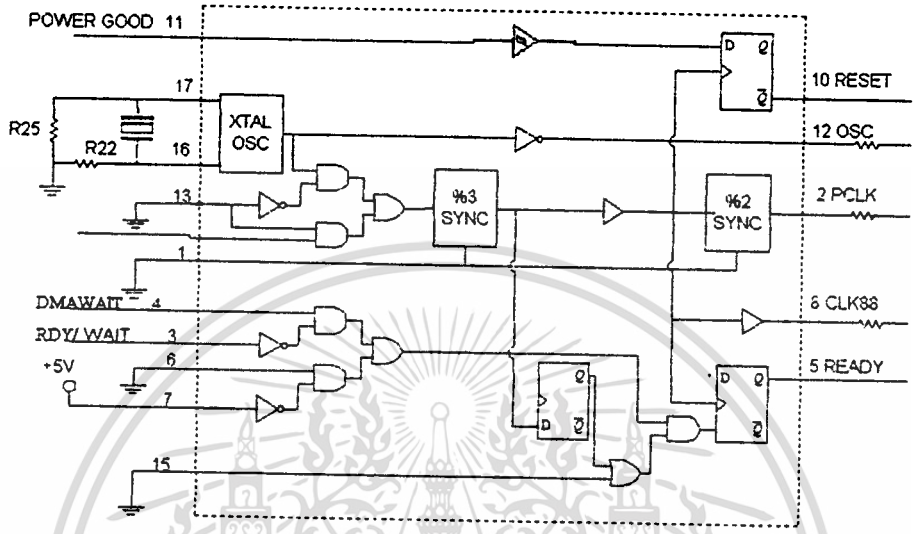
<u>ตัวอย่างที่ 3.2</u>

ดังได้กล่าวมาแล้วว่าในแฟ้มความรู้หนึ่งๆอาจจะใช้เก็บข้อมูลของวงจรใดวงจรหนึ่ง หรือ ข้อมูลของหลายๆวงจรที่มีความเกี่ยวพันต่อเนื่องกันก็ได้และในทางกลับกัน วงจรที่มีความสลับซับซ้อนมากๆก็อาจจะแยกเก็บอยู่ในหลายๆแฟ้มความรู้ก็ได้

รูปที่ 3.8 ก แสดงส่วนประกอบภายในตัวอุปกรณ์ I.C เบอร์ 8284 (U1)ซึ่งจะเห็นว่าอุปกรณ์ตัวนี้มีหน้าที่ 3 อย่างคือ ผลิตสัญญาณ RESET, ผลิตสัญญาณนาฬิกา และผลิตสัญญาณ READY ดังนั้นในการเก็บข้อมูลลงแฟ้มความรู้จึงต้องแยกอุปกรณ์ตัวนี้ออกเป็น U1/1, U1/2 และ U1/3 รูปที่ 3.8 ข แสดงตัวอย่างการเก็บข้อมูลความรู้ของอุปกรณ์ตัวนี้ไว้ใน แฟ้มความรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 3.8 ค แสดงตัวอย่างการเก็บข้อมูลของสัญญาณ READY ซึ่งมีความซับซ้อนมากขึ้นจึงต้องแยกเก็บเป็นสองแฟ้มความรู้ย่อย และแสดงการโยงจากแฟ้มความรู้หลักไปยังแฟ้มความรู้ย่อย



รูปที่ 3.8 ก. วงจรประกอบคำอธิบายตัวอย่างที่ 3.2

compo	pin	func	group	t_diagm	signame
U1/2	8	O	0		CLK88
U1/2	2	O	0		PCLK
U1/2	12	O	0		OSC
U1/2	13	CTRL	8	ระหว่าง 0 - 0.3 V	GND
U1/2	1	CTRL	8	ระหว่าง 0 - 0.3 V	GND
U1/2	16	I	9		XTAL-
U1/2	17	I	9		XTAL+
R22	0	O	0		XTAL-
R22	1	I	9	ระหว่าง 0 - 0.3 V	GND
R25	0	O	0		XTAL+
R25	1	I	9	ระหว่าง 0 - 0.3 V	GND

รูปที่ 3.8 ข. แฟ้มความรู้ประกอบตัวอย่างที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

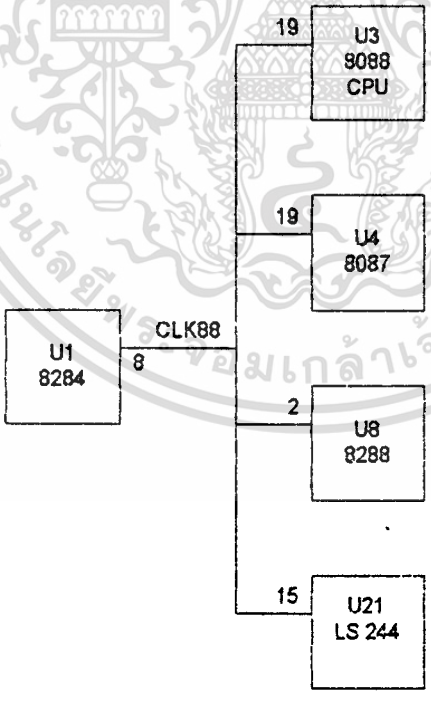
compo	pin	func	group	t_diagm	signame
....	..	....	....	.....	.....
U1/3	3	CTRL	8	เปิดแฟ้ม DMA WAIT	NEXT
U1/3	4	CTRL	8	เปิดแฟ้ม RDY/WAIT	NEXT
....	..	....	....	.....	.....

รูปที่ 3.8 ศ. ตัวอย่างความรู้ที่โยงไปแฟ้มความรู้อื่น

ตัวอย่างที่ 3.3

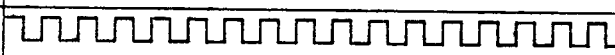
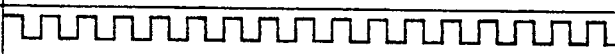
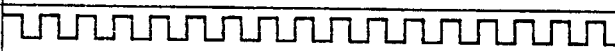
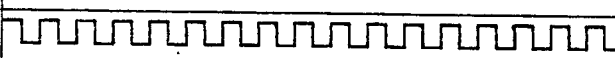
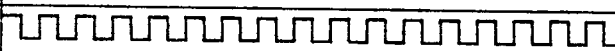
รูปที่ 3.9 ก. แสดงวงจรสัญญาณนาฬิกาที่ออกขา U1ขา8 จากตัวอย่างที่ 3.2 จ่ายไปให้กับ U3, U4, U8 และ U21

รูปที่ 3.9 ข. แสดงแฟ้มความรู้ของวงจรในรูป 3.9 ก. ซึ่งมีลักษณะเหมือนกับแฟ้มความรู้ที่ใช้งานจริง



รูปที่ 3.9 ก. วงจรประกอบตัวอย่างที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
u1	8	o	0		clk88
u3	19	ctrl	8		clk88
u4	19	ctrl	8		clk88
u8	2	ctrl	8		clk88
u21	15	i	9		clk88

รูปที่ 3.9 ข. ตารางประกอบตัวอย่างที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

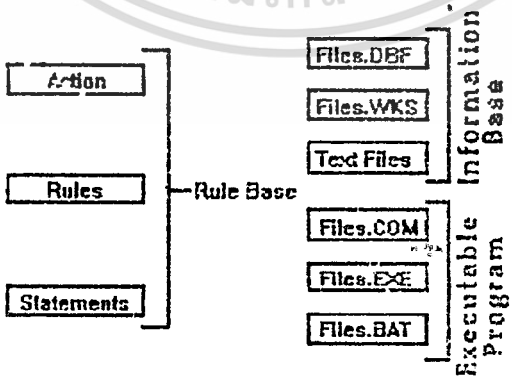
ฮิวริสติก การสร้างกฎและการอนุมาน

4.1 เครื่องมือที่ช่วยในการสร้าง

4.1.1 เปลือกระบบผู้เชี่ยวชาญ VP-Expert

ในการวิจัยจะใช้ เปลือกระบบผู้เชี่ยวชาญ ชื่อ VP-Expert ของบริษัท PAPERBACK SOFTWARE INTERNATIONAL, 2830 Ninth Street, Berkely, California 94710[16] ซึ่งมีคุณลักษณะดังนี้

1. เป็นเครื่องมือในการพัฒนาระบบผู้เชี่ยวชาญแบบ อิงกฎ (Rule base)
2. การเขียนกฎ ใช้ภาษาอังกฤษธรรมดา
3. ใช้งานบนไมโครคอมพิวเตอร์ PC/... ภายใต้ระบบปฏิบัติการ MS-DOS version 2.xx หรือสูงกว่า
4. ใช้หน่วยความจำ RAM 256 KB หรือมากกว่า
5. สามารถทำการแลกเปลี่ยนข้อมูลกับแฟ้มข้อมูลจากภายนอกได้ เช่น dBASE, LOTUS หรือแฟ้มข้อมูลชนิดตัวอักษรอื่นๆ
6. กลไกการอนุมานสามารถใช้ได้ทั้งแบบถอยหลังจากผลไปหาเหตุ และ แบบเดินหน้าจากเหตุไปสู่ผล
7. สามารถเลือกช่องวินโดว์เพื่อเฝ้าดูการทำงานของกลไกการอนุมานได้ ถ้าต้องการ
8. สามารถกำหนดค่าแฟคเตอร์ความแน่นอนให้กับข้อมูลได้ ถ้าต้องการ
9. มีเท็กอิดิเตอร์ ภายใน
10. นอกจากนี้ยังสามารถเรียกใช้โปรแกรมอื่นๆที่มีนามสกุล .COM, .EXE และ .BATมาร่วมทำงานได้



รูปที่ 4.1 บล็อกไดอะแกรมของเปลือกระบบผู้เชี่ยวชาญ VP-Expert

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) กลไกการอนุมาน

ประกอบด้วยส่วนพื้นฐาน 3 ส่วน คือ

ก. แอคชั่นบล็อก

ข. กฎ

ค. ถ้อยแถลง (statement)

ก. แอคชั่นบล็อกประกอบด้วยคีย์เวิร์ด "ACTIONS" ตามด้วยอนุประโยคที่เป็นลำดับในการแก้ไขปัญหา และปิดท้ายด้วยเครื่องหมาย ";" ความหมายของแอคชั่นบล็อกคือเป็นการบอกกลไกการอนุมานว่าต้องการค้นหาอะไร และมีลำดับการค้นหาอย่างไร แอคชั่นบล็อกเปรียบเสมือนโมดูลหลักของโปรแกรม

อนุประโยคที่เป็นลำดับในการค้นหา มีลักษณะคล้ายๆกับลำดับของคำสั่งต่างๆ ภายในโมดูลของโปรแกรม อนุประโยคที่สำคัญ ได้แก่อนุประโยค FIND ซึ่งเป็นคำสั่งที่สั่งให้หา "ตัวแปรเป้าหมาย"

แอคชั่นบล็อกมีรูปแบบการใช้งานดังนี้

```
ACTIONS
  < clause 1 >
  [ clause 2 ]
  [ clause 3 ]
  [ clause n ]
    ;
```

ข. กฎของ VP - Expert คือการแสดงความรู้ด้วยโปรดักชั่นซิสเต็ม ดังที่ได้กล่าวมาแล้วในหัวข้อที่ 2.2.2

ค. ถ้อยแถลง มีลักษณะคล้ายๆกับคำสั่งที่สั่งให้มอบหมายค่า หรือมอบหมายคุณลักษณะ อย่างใดอย่างหนึ่งให้กับตัวแปร ถ้อยแถลงที่สำคัญได้แก่ ASK ซึ่งสั่งให้ผู้ใช้ต้องทำการมอบหมายค่าใดค่าหนึ่งให้กับตัวแปร เช่น

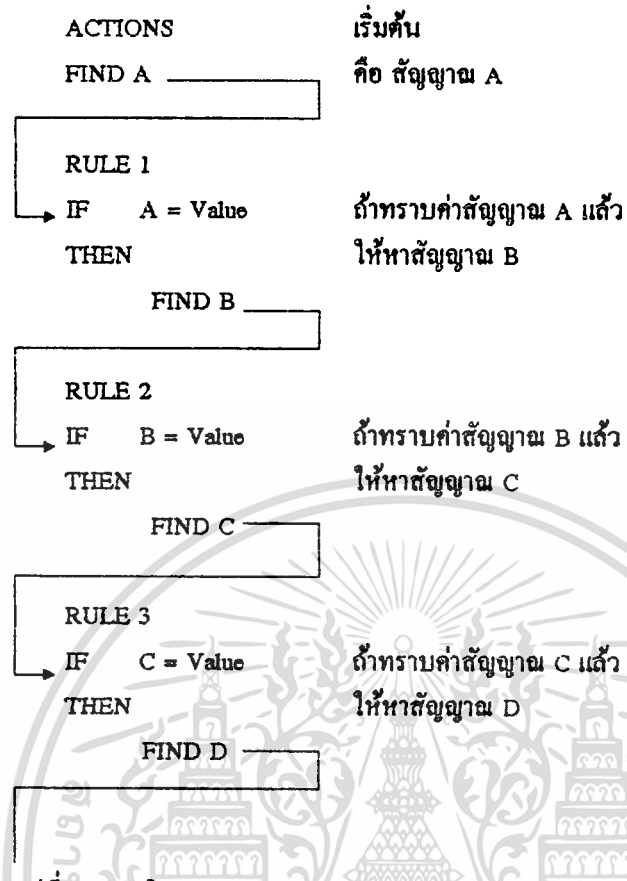
```
ASK state : " What is the logic state ? " ;
```

เมื่อผู้ใช้ตอบคำถาม เช่น ตอบว่า "Hight" สถานะทางโลจิกที่ผู้ใช้ตอบก็จะถูกนำไปใส่ให้กับตัวแปร state เป็นต้น

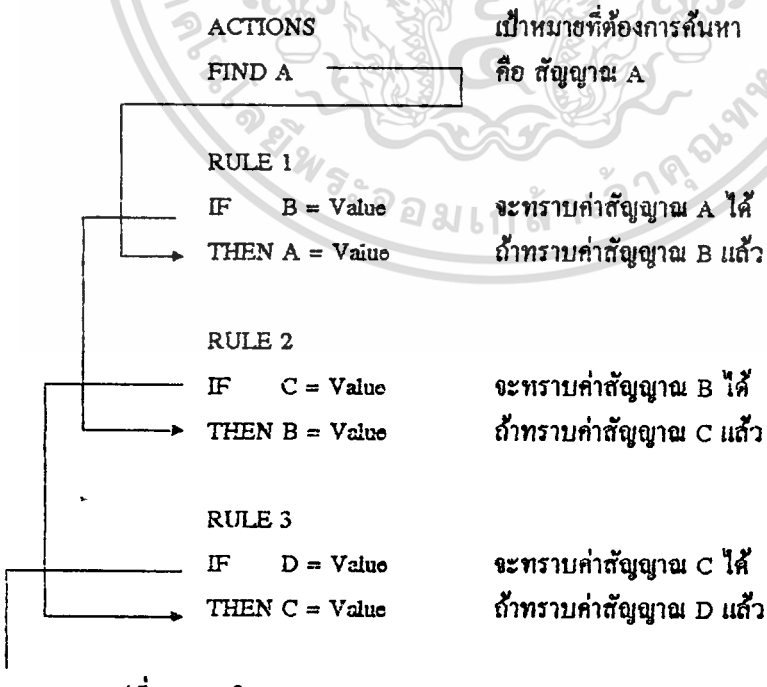
(2) ทิศทางการอนุมาน

ทิศทางการอนุมานมี 2 ทิศทาง คือ แบบเดินหน้าจากเหตุไปสู่ผล และแบบถอยหลังจากผลไปหาเหตุ ดังแสดงในรูปที่ 4.1 ก และ 4.1 ข ซึ่งในงานวิจัยนี้ได้ใช้แบบเดินหน้าจากเหตุไปสู่ผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ก ทิศทางการอนุมานของ VP-Expert แบบเดินหน้า



รูปที่ 4.2 ข ทิศทางการอนุมานของ VP-Expert แบบถอยหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 เครื่องวิเคราะห์โลจิก

เป็นเครื่องมือที่มีความเหมาะสมในการค้นหาข้อขัดข้องบนไมโครคอมพิวเตอร์มากที่สุด [17] สามารถที่จะจับและเก็บสัญญาณได้พร้อมๆกันมากกว่า 40 ช่อง และสามารถที่จะเรียกสัญญาณที่จับได้และเก็บไว้ออกมาแสดงได้หลายรูปแบบ เช่น แผนผังเวลา, แผนผังสถานะ เป็นต้น ประโยชน์ที่สำคัญมากที่สุดของเครื่องวิเคราะห์โลจิก คือสามารถที่จะรอคอยให้ข้อมูลหรือเหตุการณ์ใดเหตุการณ์หนึ่งปรากฏ แล้วจึงเริ่มจับสัญญาณได้ ทั้งนี้เพราะตัวเครื่องวิเคราะห์โลจิกเองก็คือไมโครคอมพิวเตอร์ระบบหนึ่ง[17] ซึ่งผู้ใช้สามารถที่จะโปรแกรมให้ทำงานในลักษณะต่างๆได้จากปุ่มปรับต่างๆบนแผงหน้าเครื่อง

ข้อเสียของเครื่องวิเคราะห์โลจิก คือการเซ็ตอัพ[17](setup)ยากมาก เทคนิครายละเอียดปลีกย่อย การใช้งานและขีดความสามารถของแต่ละเครื่องจะแตกต่างกันไปตามรุ่นและยี่ห้อ ในงานวิจัยนี้ใช้เครื่องวิเคราะห์โลจิกยี่ห้อ Tektronix รุ่น 1241 แต่อย่างไรก็ตามเทคนิคการจับสัญญาณจะใช้วิธีบรรยายธรรมดา เพื่อที่จะได้สามารถนำไปใช้ได้กับเครื่องวิเคราะห์โลจิกแบบใดก็ได้

ในการจับสัญญาณตอบสนองจะใช้วิธีเปิดเครื่องให้แผงวงจรที่จะทำการค้นหาข้อขัดข้องทำงานไปก่อน แล้วจึงรอคอยเพื่อที่จะจับสัญญาณ ณ เหตุการณ์ที่ได้กำหนดไว้ล่วงหน้า ซึ่งได้กล่าวมาแล้วในหัวข้อที่ 3.3.5.1 ทั้งนี้เนื่องจากงานวิจัยนี้ไม่ได้ใช้เครื่องกำเนิดสัญญาณทดสอบ ดังนั้นในบางครั้งเหตุการณ์ที่รอคอยจึงอาจจะมีมากกว่า 1 ระดับ เช่น หลังจากเปิดเครื่องแล้วรอคอยให้เหตุการณ์ A เกิดขึ้นก่อน เมื่อเหตุการณ์ A เกิดขึ้นแล้วให้หน่วงเวลา หรือรอคอยให้เหตุการณ์ B เกิดขึ้นเป็นลำดับต่อมา เป็นต้น

4.2 วิธีแสดงและใช้ความรู้

เนื่องจากระบบผู้เชี่ยวชาญฯ ที่ทำการวิจัยนี้ได้ทำการพัฒนาบนเปลือกระบบผู้เชี่ยวชาญ VP-Expert ดังนั้นการแสดงความรู้จึงแสดงอยู่ในรูปของกฎ ดังที่ได้กล่าวมาแล้วในหัวข้อที่ 4.1 ซึ่งส่วนประกอบของกฎใน VP -Expert มีดังนี้

1. ชื่อของกฎ ประกอบด้วยตัวอักษรหรือตัวเลขไม่เกิน 20 ตัว

2. ส่วน IF ประกอบด้วยเงื่อนไขต่างๆที่นำมาใช้หาเหตุผลไม่เกิน 10 เงื่อนไข สามารถใช้โอเปอร์เรเตอร์ ความสัมพันธ์ได้ ดังนี้

= เท่ากับ	> มากกว่า
< น้อยกว่า	>= มากกว่า หรือ เท่ากับ
<= น้อยกว่า หรือ เท่ากับ	<> ไม่เท่ากับ

และสามารถใช้โอเปอร์เรเตอร์โลจิก AND และ OR ได้

3. ส่วน THEN คือส่วนปฏิบัติการหรือส่วนสรุป ประกอบด้วยสมการที่มีตัวแปรตัวเดียว และค่าเดียวอย่างน้อย 1 สมการ สมการนี้จะมอบหมายค่าให้กับตัวแปรในส่วน IF ถ้ากฎเป็นจริง ค่าในสมการนี้สามารถกำหนดแฟคเตอร์ความแน่นอนได้ โดยที่ค่าแฟคเตอร์นี้มีค่าอยู่ระหว่าง 0 ถึง 100 เช่นเดียวกับเปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เครื่องหมาย " ; " เพื่อแสดงการสิ้นสุดของกฎแต่ละกฎ

4.3 ฮิวริสติก

4.3.1 ฮิวริสติกในการค้นหาของ SCIRTSS[14]

ในโมเดลของ SCIRTSS ข้อขัดข้องจะส่งผ่านจากอินพุตซีเควนท์ เข้าไปเก็บไว้ในตัวแปรสถานะก่อนที่จะส่งออกไปยังเอาท์พุต ดังนั้นในการหาทางเดินของข้อขัดข้องจึงต้องมีการคำนวณหาสถานะถัดไปก่อน เพื่อใช้เป็น อินพุตเวคเตอร์ ทุกครั้งที่มีการคำนวณหาสถานะถัดไป SCIRTSS จะตัดสินใจว่าสถานะใหม่ที่ได้นี้ คือเป้าหมายใช่หรือไม่ ถ้าไม่ใช่ก็จะเลือกสถานะที่ยังไม่ได้ทำการคำนวณขึ้นมาใหม่ แล้วทำการคำนวณและตัดสินใจ กระทำซ้ำเช่นนี้เรื่อยไป เทคนิคในการเลือกสถานะที่ยังไม่ได้คำนวณสถานะถัดไปอาศัยฮิวริสติกฟังก์ชั่นเข้าช่วยในการหาทางเดินที่มีโอกาสประสบผลสำเร็จมากที่สุด ซึ่งนิยามไว้ว่า

$$H_n = G_n + wF_n$$

โดยที่ H_n คือ ค่าฮิวริสติกของสถานะที่ n

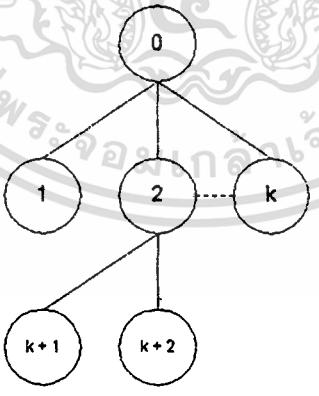
G_n คือ ความยาวของซีเควนท์จากรูตมาถึง n ดังรูปที่ 4.2

$G_1 = G_2 = G_k = 1$ และ $G_{k+1} = G_{k+2} = 2$

F_n คือ ฮิวริสติกฟังก์ชั่นหลัก

w คือ น้ำหนักในการช่วยนำทางของ F_n

สถานะที่ยังไม่ได้รับการคำนวณที่มีค่า H_n น้อยที่สุดจะได้รับเลือก



รูปที่ 4.3 ขบวนการค้นหาของ SCIRTSS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 ฮิวริสติกฟังก์ชั่นในงานวิจัย

ในหัวข้อที่ 3.3.4 ได้กล่าวถึงเหตุผลในการจัดเรียงขาอุปกรณ์ตามลำดับความสำคัญไว้ว่า " ถ้ามีอุปกรณ์ตัวใดๆ ไม่ทำงาน หรือทำงานไม่ปกติ ประการแรกที่สามารถทราบได้ก็คือ การตรวจดูเอาต์พุตว่าผิดปกติหรือไม่ ถ้าปรากฏว่าเอาต์พุต ไม่มี หรือมีแต่ไม่ปกติ ประการต่อไป (นอกจาก การตรวจสอบทางกายภาพแล้ว)คือตรวจดูว่ามีไฟเลี้ยงวงจรตามปกติหรือไม่ ถ้าปรากฏว่ามีไฟเลี้ยงวงจรตามปกติ ประการต่อไปคือตรวจดูว่ามีสัญญาณควบคุม เข้ามาปกติเรียบร้อยหรือไม่ ถ้าปรากฏว่าปกติดีก็จะตรวจสอบสัญญาณอินพุตที่เข้ามาต่อไป สัญญาณอินพุต, สัญญาณควบคุม, ไฟเลี้ยง และ gnd เหล่านี้ ถ้าไม่ปกติหรือไม่มี ก็จะทำการตรวจสอบไปยังที่มาของสัญญาณเหล่านั้นต่อไป แต่ถ้าตรวจสอบแล้วปรากฏว่าสัญญาณทั้ง 4 ประเภทที่กล่าวมาแล้วข้างต้นเรียบร้อยปกติหมด เราก็สรุปได้ว่า อุปกรณ์ตัวนี้ชำรุด " ซึ่งในหัวข้อที่ 3.3.4 ได้ทำการแบ่งกลุ่มฟังก์ชั่นขาของอุปกรณ์เหล่านี้ ตามลำดับความสำคัญ ดังนี้

'o' หมายถึง เอาต์พุต ลำดับ 0
'vcc' หมายถึง ไฟเลี้ยง ลำดับ 2
'gnd' หมายถึง ดิน ลำดับ 2
'c' หมายถึง clear, reset ลำดับ 5
's' หมายถึง set, preset ลำดับ 6
'i' หมายถึง อินพุต ลำดับ 9

จากลำดับต่างๆที่กล่าวมาจะเห็นว่าลำดับที่ 1, 3,4,7 ไม่ได้ถูกนำมาใช้งาน ทั้งนี้เพื่อเว้นไว้ให้การพัฒนาในโอกาสต่อไปมีโอกาสที่จะสอดแทรกฟังก์ชั่นอื่นๆที่เหมาะสมเข้าไปได้

เมื่อพิจารณาเปรียบเทียบกับฮิวริสติกฟังก์ชั่นของ SCIRTSS แล้ว ค่าเหล่านี้ก็คือค่าน้ำหนักในการช่วยนำทางนั่นเอง และถ้ากำหนดให้ $F_n = 1$ แล้วจะได้ว่า

$$H_n = G_n + w$$

โดยที่ G_n หมายถึง หมายเลขระเบียนของข้อมูล (Record number)

ดังนั้น ฮิวริสติกฟังก์ชั่น ของระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโคร คอมพิวเตอร์ในงานวิจัยนี้ คือ

หมายเลขระเบียน + ลำดับความสำคัญฟังก์ชั่นขาอุปกรณ์

ถ้าให้ H_n = ค่าฮิวริสติกฟังก์ชั่นของข้อมูลระเบียนใดระเบียนหนึ่ง

n = หมายเลขระเบียน

w = หมายเลขกลุ่มขาของอุปกรณ์ของระเบียนนั้น

จะได้ว่า

$$H_n = w + n$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการตัดสินใจเลือกเส้นทาง จะเลือกค่า H ที่น้อยที่สุด

ถ้าให้ H = เซ็ตของค่าฮิวริสติกฟังก์ชั่นของโหนดใดโหนดหนึ่ง

p = ค่าการตัดสินใจเลือกเส้นทาง

จะได้ว่า

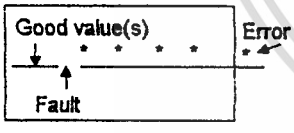
$$H = \{n, H(n) \mid n = 1, 2, 3..... \}$$

$$H_n = w + n$$

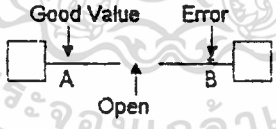
$$p = \min(H_1, H_2, H_3,.....H_n)$$

4.3.3 แนวทางการแก้ไขปัญหาแบบไกด์โพรบ

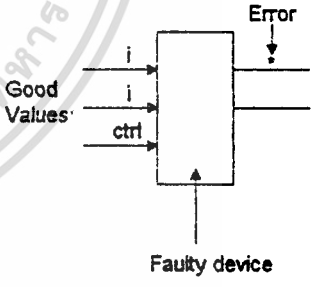
แนวทางในการแก้ไขปัญหาที่ได้กล่าวมาแล้วในหัวข้อที่ 3.2.3 เป็นวิธีการค้นหาข้อขัดข้องแบบหนึ่งที่มีวิธีการเช่นเดียวกับ การทดสอบข้อขัดข้องแบบไกด์โพรบของเครื่องมือทดสอบแบบอัตโนมัติ [14] (ATE = Automatic Test Equitment) ซึ่งมีหลักการทดสอบ ด้วยการติดตามข้อขัดข้อง จากการวัดที่ขาเอาท์พุตของอุปกรณ์ย้อนกลับเข้าไปหาจุดกำเนิดข้อขัดข้องที่แท้จริง (ข้อขัดข้องเชิงกายภาพ) สาเหตุข้อขัดข้องที่มักจะพบได้จากการทดสอบแบบนี้ คือ สาเหตุข้อขัดข้องจากวงจรเปิด และสาเหตุข้อขัดข้องจากตัวอุปกรณ์ชำรุด ดังรูปที่ 4.4 ข แสดงสาเหตุข้อขัดข้องที่เกิดจากวงจรเปิด ซึ่งตรวจพบได้จากการจับสัญญาณที่จุด B แล้วปรากฏว่าไม่เหมือนกับสัญญาณที่จุด A รูปที่ 4.4 ค แสดงสาเหตุข้อขัดข้องที่เกิดจากตัวอุปกรณ์ชำรุด ซึ่งตรวจพบได้จากการป้อนสัญญาณปกติให้กับอินพุตและขาควบคุม แล้วจับสัญญาณที่เอาท์พุต ปรากฏว่าเป็นสัญญาณที่ผิดพลาด



รูปที่ 4.4 ก
การติดตามข้อขัดข้องแบบย้อนกลับ



รูปที่ 4.4 ข
ข้อขัดข้องแบบวงจรเปิด



รูปที่ 4.4 ค
ข้อขัดข้องแบบอุปกรณ์ชำรุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบข้อขัดข้องแบบไกด์โพรบของเครื่องมือทดสอบอัตโนมัติ สามารถค้นพบสาเหตุข้อขัดข้องที่เกิดจากตัวอุปกรณ์ชำรุดได้ทุกกรณี ด้วยการป้อนสัญญาณทดสอบที่เหมาะสม และการเก็บค่าคาดหวังไว้เป็นสัญญาณอ้างอิงทุกจุดที่โพรบได้

ส่วนการค้นหาข้อขัดข้องแบบไกด์โพรบในงานวิจัย แม้ว่าจะไม่สามารถค้นพบสาเหตุข้อขัดข้องบนตัวตัวอุปกรณ์ที่ชำรุดได้ทุกกรณี เนื่องจากใช้สัญญาณทดสอบจากตัวเครื่องเอง ก็ตาม แต่ก็สามารถค้นหาสาเหตุข้อขัดข้องที่เกิดจากการใช้งานจริงได้ทุกกรณีด้วยการเก็บค่าคาดหวัง ณ เวลาที่เหมาะสม และการแบ่งเหตุการณ์การทำงานของเครื่องที่ถูกต้อง ตารางที่ 4.1 แสดงข้อเปรียบเทียบระหว่างไกด์โพรบของ ATE กับไกด์โพรบที่ใช้ในงานวิจัย

ไกด์โพรบของ ATE	ไกดุโพรบของงานวิจัย
1.เริ่มต้นการค้นหาเส้นทางของข้อขัดข้องด้วยการจับสัญญาณที่ผิดพลาดจากภาคเอาท์พุต	1.เริ่มต้นการค้นหาเส้นทางของข้อขัดข้องด้วยการจับสัญญาณที่ผิดพลาดจากภาคเอาท์พุต
2. ทิศทางการติดตามค้นหาข้อขัดข้องมีทิศทางย้อนกลับจากภาคเอาท์พุตที่ปรากฏข้อขัดข้อง ย้อนกลับไปสู่ต้นกำเนิดข้อขัดข้อง	2. ทิศทางการติดตามค้นหาข้อขัดข้องมีทิศทางย้อนกลับจากภาคเอาท์พุตที่ปรากฏข้อขัดข้อง ย้อนกลับไปสู่ต้นกำเนิดข้อขัดข้อง
3. กำเนิดสัญญาณทดสอบด้วยเครื่องกำเนิดสัญญาณ	3. กำเนิดสัญญาณทดสอบด้วยการเปิดให้เครื่องทำงานแล้วอาศัยสัญญาณที่เกิดขึ้นบนตัวเครื่อง
4. ต้องการรับการสนับสนุนข้อมูลอ้างอิงบนทุกจุดที่ทำการจับสัญญาณทดสอบ	4. ต้องการรับการสนับสนุนข้อมูลอ้างอิงบนทุกจุดที่ทำการจับสัญญาณทดสอบ
5. ทำการทดสอบวงจรซีเควนด้วยเวคเตอร์สัญญาณทดสอบ และกรอบของเวลา	5. ทำการค้นหาข้อขัดข้องของวงจรซีเควนด้วยสัญญาณบนเครื่องที่ถูกทดสอบ และความยาวของแผนผังเวลาหรือแผนผังสถานะ

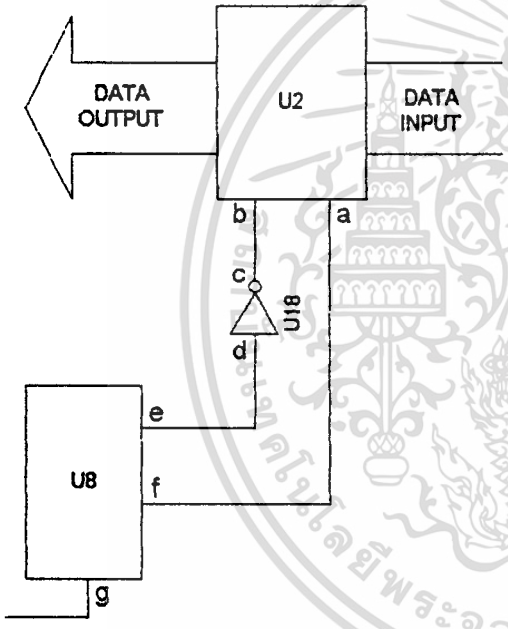
ตารางที่ 4.1 เปรียบเทียบการทดสอบข้อขัดข้องแบบไกด์โพรบของ ATE กับที่ใช้ในงานวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4 การนำค่าฮิวริสติกมาใช้โดยวิธีปริยาย

ปัจจัยหนึ่งที่ทำให้ระบบผู้เชี่ยวชาญล้มเหลวคือความล่าช้าอันเนื่องมาจากการค้นหากฎที่สอดคล้องกับเงื่อนไข และความล่าช้าอันเนื่องมาจากการค้นหาข้อมูลความรู้ ในงานวิจัยนี้ได้พยายามแก้ไขปัญหาดังกล่าวด้วยการออกแบบกฎเท่าที่จำเป็น และลดหรือตัดขั้นตอนบางอย่างที่ไม่จำเป็นออก เช่น ในระหว่างทำการตัดสินใจเลือกเส้นทางการค้นหาจะตัดกรรมวิธีการคำนวณค่าฮิวริสติกออก แล้วใช้วิธีการนำค่าฮิวริสติกมาใช้โดยปริยายแทน ซึ่งมีวิธีการดังตัวอย่างต่อไปนี้

สมมติว่ารูปที่ 4.5 ก เป็นส่วนหนึ่งของวงจรที่กำลังค้นหาข้อขัดข้อง และมีข้อมูลในแฟ้มความรู้ ดังรูปที่ 4.5 ข และการค้นหาข้อขัดข้องได้ดำเนินการมาถึง U2 ซึ่งไม่มีเอาท์พุตออกไป



rec	compo	pin	func	group	
1	U2	a	ctrl	8	
2	U2	b	ctrl	8	
3	U2	inp	i	9	
4	U2	inp	i	9	
...	.....	.....	.....	.....	.....
10	U2	inp	i	9	
11	U18	c	o	0	
12	U18	d	i	9	
13	U8	e	o	0	
14	U8	f	o	0	
15	U8	g	ctrl	8	

รูปที่ 4.5 ก

รูปที่ 4.5 ข

รูปที่ 4.5 ก. แสดงวงจรประกอบคำอธิบายการนำค่าฮิวริสติกมาใช้โดยปริยาย

ข. แสดงตารางที่เป็นส่วนหนึ่งของแฟ้มข้อมูล โดยที่กำหนดให้ขาของอุปกรณ์เป็น a, b, c,...g และ inp แทนหมายเลขขา 1, 2, 3,...เพื่อความสะดวกในการอธิบาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมติให้ข้อมูลในรูปที่ 4.5 ข มีหมายเลขระเบียน(record number)เป็น 1,2,3,...ตามลำดับ จากฮิวริสติกฟังก์ชั่น และค่าการตัดสินใจ

$$H = \{ n, H(n) \mid n = 1,2,3,...\}$$

$$H_n = w + n$$

$$p = \min (H1,H2,...)$$

พิจารณาที่โหนด U2 ซึ่งไม่มีสัญญาณเอาท์พุตออกไป และ ถ้าตรวจสอบแล้วปรากฏว่าที่ขาเอาท์พุตของ U2 ทุกขาไม่พบข้อขัดข้อง จึงทำการเลือกเส้นทางการค้นหาข้อขัดข้องดังต่อไปนี้

$$H_a = 8 + 1 = 9$$

$$H_b = 8 + 2 = 10$$

$$H_{inp} = 9 + (3,4,...,10)$$ ..... ทุกขาที่เป็นinput

$$p = \min (9,10,12,13,...,19) = 9$$

ดังนั้นจุดที่ถูกเลือกคือจุด a เพื่อใช้เป็นเส้นทางในการค้นหาข้อขัดข้องต่อไป

ในทำนองเดียวกันถ้าในแฟ้มข้อมูล ระเบียนของข้อมูลที่จุด b อยู่ก่อนระเบียนข้อมูลของจุด a ส่วนหนึ่งของแฟ้มข้อมูลก็จะเป็นดังนี้

1	U2	b	ctrl	8
2	U2	a	ctrl	8

การเลือกเส้นทางการค้นหาข้อขัดข้องก็จะเปลี่ยนเป็น

$$H_a = 8 + 2 = 10$$

$$H_b = 8 + 1 = 9$$

$$H_{inp} = 9 + (3,4,...10)$$

$$p = (10,9,12,13,...19) = 9$$

ดังนั้น จุด b จึงถูกเลือกเพื่อใช้เป็นเส้นทางในการค้นหาข้อขัดข้องต่อไป

จากตัวอย่างจะเห็นว่าในทางปฏิบัติแล้ว ไม่ได้มีการคำนวณค่า H_n และ p แต่อย่างใด ทั้งนี้เพราะข้อมูล w ได้ถูกจัดเรียงลำดับจากน้อยไปหามากมาแล้ว ดังกล่าวไว้ในหัวข้อที่ 3.3.4 และ 3.3.4 และข้อมูล n ก็คือหมายเลขระเบียนของข้อมูลซึ่งใช้กรรมวิธีการค้นหาแบบเชิงเส้น ซึ่งมีลำดับหมายเลขระเบียนอยู่แล้ว

ดังนั้นค่าโดยปริยายของฮิวริสติกฟังก์ชั่นก็คือลำดับข้อมูลในแฟ้มข้อมูลนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.5 ตัวอย่างความจำเป็นที่ต้องใช้ผู้เชี่ยวชาญ

ตัวอย่างที่ 4.1

จากตัวอย่างในหัวข้อที่แล้ว(4.3.4) จะเห็นว่าลำดับก่อน - หลังของข้อมูล มีอิทธิพลต่อทิศทางการค้นหาข้อขัดข้องเป็นอย่างมาก ตัวอย่างเช่น

ก). ถ้าให้ข้อมูลที่จุด a อยู่ก่อนจุด b

เส้นทางการค้นหาก็จะเรียงจากจุด a บน U2 มายังจุด f บน U8

พิจารณาที่โหนด U8

Hf = 0 + 14 = 14

He = 0 + 13 = 13

Hg = 8 + 15 = 23

p = min (14,13,23) = 13

ดังนั้นจุดที่ถูกเลือกต่อไป คือจุด e

และมีกฎอยู่ 2 ชุดซึ่ง (จะกล่าวต่อไปในหัวข้อที่ 4.4) สนับสนุนฮิวริสติกโดย

กฎชุดแรกกล่าวว่า " ถ้าข้อขัดข้องปรากฏขึ้นที่ขาเอาท์พุตขาใดขาหนึ่งแล้ว ให้ทำการตรวจสอบทุกขาที่เป็นเอาท์พุตของอุปกรณ์ตัวนั้น และผลการตรวจสอบต้องปรากฏว่าทุกขามีข้อขัดข้องปรากฏขึ้นทั้งสิ้น จึงจะทำการตรวจสอบขาถัดไปได้ ..."

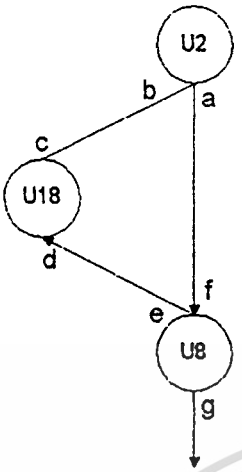
กฎชุดหลังกล่าวว่า " ถ้าข้อขัดข้องปรากฏขึ้นที่ขาเอาท์พุตขาใดขาหนึ่งแล้ว ให้ทำการตรวจสอบอุปกรณ์ทุกตัวที่ต่อร่วม (แฟนเอาท์) อยู่กับข่ายสัญญาณนั้น..."

จากเหตุผลทั้งหมดดังกล่าวทำให้เส้นทางการค้นหาข้อขัดข้องดำเนินต่อจากจุด a และจุด f คือ จุด e, d และ g ซึ่งเมื่อนำมาเขียนเป็นลำดับ จะได้ดังนี้

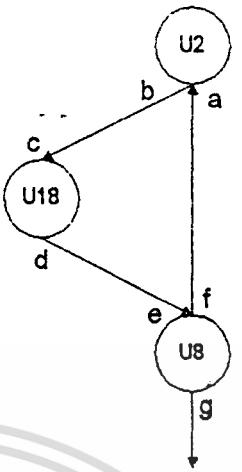
เส้นทาง = { a,f,e,d,g }

และนำมาเขียนเป็นไดกราฟได้ ดังรูปที่ 4.6 ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 ก



รูปที่ 4.6 ข

รูปที่ 4.6 แสดงเส้นทางการค้นหาข้อขัดข้องประกอบตัวอย่างที่ 4.1

ข). ถ้าให้ข้อมูลที่จุด b อยู่ก่อนจุด a

ก็จะได้เส้นทางการค้นหาข้อขัดข้อง ดังนี้

เส้นทาง = { b,c,d,e,f,a,g }

และนำมาเขียนเป็นไดกราฟได้ ดังรูปที่ 4.6 ข

เมื่อพิจารณาวงจรในรูปที่ 4.4 ก แล้วจะเห็นว่าเส้นทางการค้นหาข้อขัดข้องในรูปที่ 4.5 ข มีความถูกต้องเหมาะสมกว่ารูปที่ 4.5 ก ทั้งนี้เป็นผลมาจากการเรียงลำดับข้อมูล ซึ่งในทางปฏิบัติแล้วผู้ที่จะทำการกำหนดลำดับข้อมูลที่มีน้ำหนักเท่ากันเช่นนี้ จึงควรเป็นผู้เชี่ยวชาญ

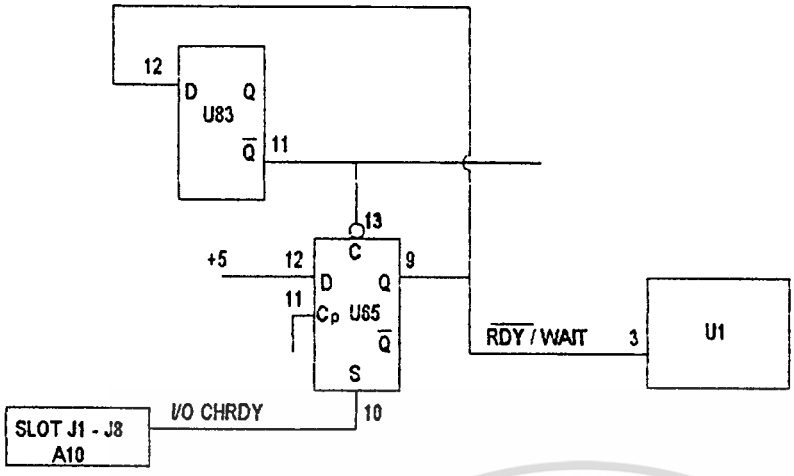
<u>ตัวอย่างที่ 4.2</u>

วงจรในรูปที่ 4.7 ก เป็นส่วนหนึ่งของวงจร wait state ของ cpu 8088 และมีการจัดเก็บข้อมูล ดังรูปที่ 4.7 ข

เพื่อความสะดวกจึงกำหนดรูปย่อของขาอุปกรณ์ ดังนี้

Umn หมายถึง ขาที่ n ของอุปกรณ์ตัวที่ m เช่น U1 ขา 3 เขียนได้เป็น U1.3 เป็นต้น

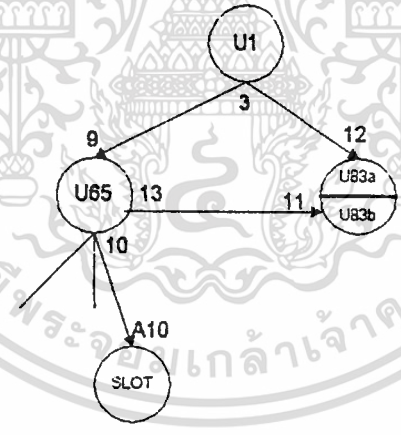
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



compo	pin	func	group
U1	3	o	0
U65	9	o	0
U65	13	c	5
U65	10	s	6
U65	11	ctrl	8
U65	12	i	9
U83	11	o	0
U83	12	i	9
SLOT	A10	o	0

รูปที่ 4.7 ก วงจรประกอบตัวอย่างที่ 4.2

รูปที่ 4.7 ข ตารางประกอบตัวอย่างที่ 4.2



รูปที่ 4.7 ค ไดกราฟประกอบตัวอย่างที่ 4.2 ซึ่งวงกลมหมายถึงตัวอุปกรณ์และตัวเลขที่อยู่รอบๆหมายถึงขาของตัวอุปกรณ์

เมื่อพิจารณาจากข้อมูลแล้วจะเห็นได้ว่าเส้นทาง { U1.3, U65.9, U65.13, U83.11, U83.12 } มีลักษณะเป็นวงรอบ ซึ่งถ้าดำเนินการค้นหาข้อขัดข้องแล้วจะเกิดการวนลูปไม่สิ้นสุด เพื่อเป็นการแก้ไขปัญหานี้จึงแบ่งภาค U83 ออกเป็น 2 ส่วน ตามเส้นทางการค้นหาข้อขัดข้อง (ดังที่ได้กล่าวมาแล้วในหัวข้อที่ 3.3.4) ได้แก่ U83a และ U83b ซึ่งเมื่อนำเส้นทางการค้นหาข้อขัดข้องนี้มาเขียนไดกราฟ จะได้ดังรูปที่ 4.6 ค และสามารถดำเนินการค้นหาข้อขัดข้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่เมื่อพิจารณาจากวงจรในรูปที่ 4.6 ก็แล้วจะเห็นว่า เมื่อเปรียบเทียบความน่าจะเป็นในการเกิดข้อขัดข้องบนโหนด U65 ความน่าจะเป็นของขา 10 (set) มีมากกว่าความน่าจะเป็นของขา 12 (clear) ดังนั้นจึงควรจัดเรียงข้อมูลในแฟ้มความรู้ใหม่ โดยนำข้อมูลในระเบียน U65.10 ให้อยู่ก่อน U65.13 ซึ่งในกรณีนี้จึงเป็นกรณีพิเศษที่แตกต่างไปจากข้อกำหนดที่กำหนดไว้แต่เดิม

จะเห็นว่าข้อมูลในแฟ้มความรู้นั้น แตกต่างจากข้อมูลในแฟ้มข้อมูลธรรมดามาก ในแฟ้มความรู้จะต้องอาศัยเทคนิคต่างๆเข้าช่วย เช่น การแบ่งภาคตัวอุปกรณ์ การจัดลำดับข้อมูล ซึ่งลักษณะเช่นนี้จึงจำเป็นต้องอาศัยผู้เชี่ยวชาญเข้ามาช่วยดำเนินการ

4.4 การสร้างกฎ

เป็นการแสดงความรู้ของ VP Expert ซึ่งมีลักษณะที่สำคัญ คือ ในแต่ละกฎจะมีลักษณะเป็นโมดูลๆ หนึ่ง หรือโปรแกรมย่อยโปรแกรมหนึ่ง แสดงอยู่ในรูปของ IF... THEN... แต่ละกฎก็มีหน้าที่ของตัวเอง กฎสามารถที่จะเพิ่มเติมหรือลบออกได้ แต่ถ้าเพิ่มเติมกฎแล้ว มีความขัดแย้งกันก็จะล้มเลิกกฎเก่า หันมาใช้กฎใหม่แทน ถึงแม้ว่ากฎแต่ละกฎจะมีความเป็นอิสระของตัวเอง แต่ทุกกฎก็ถูกใช้เป็นตัวแสดงความรู้บนฐานความรู้อันเดียวกัน

ในการค้นหาข้อขัดข้อง กลไกการอนุมาน จะต้องทำการค้นหาไปตามกฎต่างๆเหล่านี้ ว่าสอดคล้องกับเงื่อนไขหรือไม่ ซึ่งถ้ากฎมีจำนวนมากเกินไปหรือมีเงื่อนไขละเอียดเกินไป ก็จะทำให้การค้นหาเสียเวลามาก ดังนั้นในการออกแบบของเราจึงใช้กฎเท่าที่จำเป็น จำนวน 24 กฎ และจัดความสัมพันธ์ระหว่างกฎหรือโมดูลลดหลั่นลงมาดังรูปที่ 4.8

การสร้างกฎ อาศัยแนวทางในการดำเนินการค้นหาข้อขัดข้องดังที่ได้กล่าวมาแล้วในหัวข้อที่ 3.2.3 และ 4.3.3 ซึ่งแบ่งตามหน้าที่ได้ดังนี้

1. กฎที่ทำหน้าที่เกี่ยวกับการกรองหรือการคัดเลือก
2. กฎที่ทำหน้าที่เกี่ยวกับการแสดงข้อมูลอ้างอิง หรือ การแจ้งข่าวสาร
3. กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 1
4. กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 2
5. กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยเชิงกายภาพ
6. กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นสุดท้าย
7. กฎที่ทำหน้าที่เกี่ยวกับการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

4.4.1 กฎที่ทำหน้าที่เกี่ยวกับการกรองหรือการคัดเลือก

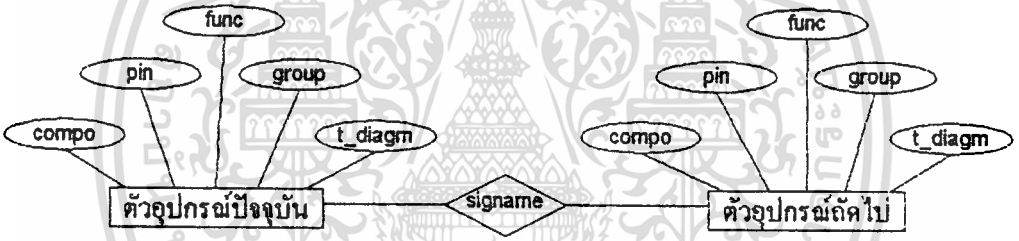
ในการดำเนินการค้นหาข้อขัดข้อง เริ่มต้นจากการจับสัญญาณที่ขาของอุปกรณ์ซึ่งอาการข้อขัดข้องปรากฏ แล้วติดตามข้อขัดข้องแบบย้อนกลับไปสู่ต้นกำเนิดหรือสาเหตุข้อขัดข้องที่แท้จริง ดังนั้นกฎที่จำเป็นประการแรก คือกฎที่ทำให้กลไกการอนุมาน ทำการอนุมานเสมือนหนึ่งเดินทางย้อนกลับไปสู่ต้นกำเนิดข้อขัดข้องได้ กฎเกี่ยวกับการเดินทางย้อนกลับมีอยู่ 2 แบบ คือ

(1) จากอุปกรณ์ตัวหนึ่งย้อนกลับไปยังอีกตัวหนึ่ง

(2) การย้อนกลับจากการท่องไปยังโหนดต่างๆที่ต่อร่วมกันอยู่

(1) จากอุปกรณ์ตัวหนึ่งย้อนกลับไปยังอีกตัวหนึ่ง

จากการพิจารณาวิเคราะห์ โครงสร้างแฟ้มข้อมูลในหัวข้อที่ 3.3.3 จะพบว่าความสัมพันธ์ระหว่างตัวอุปกรณ์ กับ ตัวอุปกรณ์บนข่ายสัญญาณใดๆ สามารถที่จะแสดงความสัมพันธ์ได้ด้วยการประยุกต์ใช้โมเดล E-R [18](Entity Relationship) ดังรูปที่ 4.7 จากความสัมพันธ์จะเห็นว่าหนทางที่จะเดินทางย้อนกลับได้ จะต้องอาศัยชื่อของข่ายสัญญาณเป็นทางเดิน



รูปที่ 4.8 โมเดล E-R ระหว่างตัวอุปกรณ์

ดังนั้นการสร้างกฎที่จะทำให้กลไกการอนุมาน ทำการอนุมานเสมือนหนึ่งเดินทางย้อนกลับจากอุปกรณ์ตัวที่ปรากฏอาการข้อขัดข้องตัวแรกไปสู่อุปกรณ์ตัวที่ปรากฏอาการข้อขัดข้องตัวถัด ก็คือการสร้างกฎที่ทำหน้าที่เกี่ยวกับการกรองหรือการคัดเลือกระเบียนที่มีชื่อตัวอุปกรณ์ และชื่อข่ายสัญญาณเป็นคีย์นั่นเอง ซึ่งเขียนเป็นกฎได้ ดังนี้

```
RULE 10a
        IF      search = (signal[x]) AND
                pin_func = (funct[x])
        THEN  cond = same1;
```

กฎที่มีลักษณะเช่นนี้ในแฟ้มความรู้ CHIP.KBSคือกฎ 10a

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) การย้อนกลับจากการท่องไปยังโหนดต่างๆที่ต่อร่วมกันอยู่

สมมุติว่ามีวงจรดังรูปที่ 3.9 ก และถ้าอาการข้อขัดข้องปรากฏขึ้นที่ขาเอาต์พุต (U1ขา8) ของตัวอุปกรณ์ที่มี แฟนเอา มากกว่าหนึ่ง ข้อขัดข้องที่ปรากฏขึ้นนั้นอาจจะไม่ได้เกิดมาจากอุปกรณ์ตัวนั้น(U1ขา8) ก็ได้ อาจจะเกิดจากอุปกรณ์ตัวอื่นๆที่ต่อร่วมอยู่เกิดการขัดข้อง แล้วส่วนหนึ่งของผลของข้อขัดข้องนั้นก็ส่งผ่านสายสัญญาณมาปรากฏที่ เอาต์พุตของอุปกรณ์ตัวที่กำลังสนใจอยู่(U1ขา8) ดังนั้นเพื่อเป็นการตรวจสอบสมมุติฐานจึงควรมีกฎที่ทำหน้าที่ท่องไปยังโหนดต่างๆที่ต่อร่วมอยู่ แล้วจึงย้อนกลับมายังขาเอาต์พุต (U1ขา8) ดังนี้

เมื่อพิจารณาการท่องไปยังโหนดต่างๆที่ต่อร่วมกันอยู่แล้วย้อนกลับมายังขาเอาต์พุตนี้ จะเห็นว่ามีลักษณะ คล้ายกับการท่องไปในโครงสร้างข้อมูลแบบไบนารีทรีอย่าง Postorder[11] ซึ่งการย้อนกลับมายังขาเอาต์พุตหลังจากที่ท่องไปยังโหนดร่วมต่างๆเหล่านี้ ก็คือการสร้างกฎที่ทำหน้าที่เกี่ยวกับการกรองหรือการคัดเลือกระเบียนที่มีชื่อตัวอุปกรณ์และฟังชั่นก์ขาอุปกรณ์ เป็นคีย์นั่นเอง ซึ่งเขียนเป็นกฎได้ดังนี้

```
RULE 10b
        IF      search = (xnode[x]) AND
                pin_func = (funct[x])
        THEN cond = same2;
```

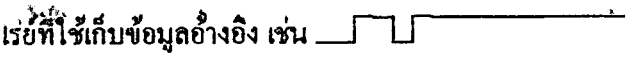
กฎที่มีลักษณะเช่นนี้ในแฟ้มความรู้ CHIP.KBSคือกฎ 10b

4.4.2 กฎที่ทำหน้าที่เกี่ยวกับการแสดงข้อมูลอ้างอิง หรือ การแจ้งข่าวสาร

กฎเหล่านี้คือส่วนที่ทำหน้าที่ติดต่อเชื่อมโยงกับผู้ใช้ ซึ่งมีจุดมุ่งหมายเพื่อต้องการที่จะแจ้งหรือแสดงข้อมูลข่าวสารให้ผู้ใช้ทราบ โดยมากแล้วกฎเหล่านี้จะมีตัวแปรหลังส่วน THEN ไม่มีความหมายอะไร แต่ที่ต้องมีไว้ก็เพื่อให้ถูกหลักไวยากรณ์ของ VP-Expert เท่านั้น ความสำคัญของกฎเหล่านี้อยู่ที่เงื่อนไขที่อยู่หลัง IF .... และ อนุประโยคหลัง THEN ... เช่น DISPLAY ..., FIND ... เป็นต้น ดังตัวอย่างต่อไปนี้

```
RULE 20b
        IF      cond = same1 OR
                cond = same2
                ..... .....
        THEN disp = meet
                DISPLAY "{xnode[x]} pin{xpin[x]} = {tdiagm[x]}" <---ข้อมูลอ้างอิง
                หรือ
                DISPLAY "For next diagnosis please...{tdiagm[x]}" <--- ข่างสาร
                .....
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง {tdiagm[x]} คือตัวแปรแบบอะเรย์ที่ใช้เก็บข้อมูลอ้างอิง เช่น  หรือ ข่าวสาร เช่น select ADDRBUS1 knownledge file. เป็นต้น

กฎที่มีลักษณะเช่นนี้ในแฟ้มความรู้ CHIP.KBSคือกฎ 20a, 20b, 50, dummy1

4.4.3 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 1

ในการค้นหาข้อขัดข้อง สมมติว่าได้ดำเนินการติดตามสาเหตุข้อขัดข้องมาถึงอุปกรณ์ตัวหนึ่งๆ ซึ่งมีโมเดลดังหัวข้อที่ 3.3.2 จะเห็นว่าสามารถวิเคราะห์ปัญหาที่จะเกิดขึ้นบนขาใดๆของตัวอุปกรณ์และสร้างเป็นกฎต่างๆ ได้ดังนี้

ฟังชั่นของขาอุปกรณ์	ผลการเปรียบเทียบ	ชื่อกฎ	ข้อสรุป
output	Yes	31a	จากรูปที่ 4.3 ข แสดงว่าเกิดวงจรเปิด
	No	31b	ยังสรุปไม่ได้ ต้องทดสอบขาอื่นๆ และ โหนดอื่นๆที่ต่อร่วมอยู่ ด้วย
ctrl และ input	Yes	34a	จากรูปที่ 4.3 ค แสดงว่าขานี้ปกติ ให้ทำการทดสอบขาอื่นๆต่อไป
	No	34b	ยังสรุปไม่ได้ ต้องทดสอบขาอื่นๆ, โหนดอื่นๆที่ต่อร่วมอยู่ และ ตรวจสอบว่า ขานี้ต่อ อยู่กับ ไฟเลี้ยง หรือ gnd หรือไม่
Vcc และ gnd	Yes	32a	จากรูปที่ 3.3 แสดงขานี้ปกติ ให้ทำการทดสอบขาอื่นๆต่อไป
	No	32b	แสดงว่าเกิด วงจรเปิดหรือลัดวงจร

หมายเหตุ ผลการเปรียบเทียบหมายถึง การเปรียบเทียบระหว่าง สัญญาณตอบสนองที่จับได้ กับ สัญญาณอ้างอิงในแฟ้มข้อมูล

ตารางที่ 4.2 การสร้างกฎเกี่ยวกับการวินิจฉัยขั้นที่ 1

4.4.4 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 2

จากหัวข้อที่ 4.4.2 ตารางการสร้างกฎ จะเห็นว่ากฎ 31b และ 34b เกิดจากผลการเปรียบเทียบระหว่างสัญญาณตอบสนองที่จับได้กับสัญญาณอ้างอิง แล้วปรากฏว่าได้ไม่เท่ากัน และยังไม่สามารถหาข้อสรุปได้ เนื่องจากยังไม่ทราบว่ามีโหนดอื่นๆต่อร่วมอยู่หรือไม่ และโหนดเหล่านั้นปกติหรือไม่ ซึ่งคุณสมบัติที่ถูกถ่ายทอดมาเหล่านี้ นำมาสร้างเป็นกฎใหม่ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โยงมาจากกฎ	คุณสมบัติที่ได้รับการถ่ายทอดมา	ผลการเปรียบเทียบใหม่	ชื่อกฎ	ข้อสรุป
31b	เป็นโหนดที่ต่อร่วมกับขาเอาพุต	Yes	60b	60b. จากรูปที่ 3.9 ก สัญญาณวัดได้ปกติ
	ของโหนดที่มีสัญญาณไม่ปกติ	No	60a	ณ จุดนี้แสดงว่าเกิดวงจรเปิด
34b	เป็นโหนดที่ต่อร่วมกับขาอินพุต หรือขาควบคุมของโหนดที่มีสัญญาณไม่ปกติ และ ถ้าขาของโหนดนี้ต่ออยู่กับ			60a. จากรูปที่ 3.9 ก แสดงว่าขาสัญญาณยังคงผิดปกติอยู่ ต้องตรวจสอบทางกาย - ภาพ แล้วทดสอบขาอื่นๆ และโหนดอื่นๆที่ต่อร่วมอยู่ต่อไป
	1. สัญญาณใดๆที่ไม่ใช่ไฟเลี้ยง และ gnd	Yes	60b	
		No	60a	
	2. ไฟเลี้ยง	Yes	60b	
	หรือ gnd	No	41	แสดงว่าเกิดวงจรเปิดหรือลัดวงจร

หมายเหตุ ผลการเปรียบเทียบหมายถึง การเปรียบเทียบระหว่าง สัญญาณตอบสนองที่จับได้ กับ สัญญาณอ้างอิงในแฟ้มข้อมูล

ตารางที่ 4.3 การสร้างกฎเกี่ยวกับการวินิจฉัยขั้นที่ 2

จากตารางจะเห็นว่ากฎใหม่ที่ถูกสร้างขึ้นได้รับการถ่ายทอดคุณสมบัติมาจากกฎเดิม ซึ่งความจริงสามารถสร้างกฎใหม่ได้ถึง 6 กฎ แต่เนื่องจากเมื่อสร้างขึ้นมาทั้ง 6 กฎแล้ว จะเกิดการกระทำที่ซ้ำซ้อนกัน ดังนั้นจึงลดทอนมาเหลือเพียง 3 กฎ คือกฎ 60a, 60b และ 41

4.4.5 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยเชิงกายภาพ

ในการค้นหาข้อขัดข้อง สิ่งที่มีความสำคัญประการหนึ่งคือ ความสามารถในการค้นหาข้อขัดข้องได้ครอบคลุม ซึ่งบางครั้งการวินิจฉัยเชิงโลจิกอย่างเดียวอาจจะไม่เป็นการพอเพียง ในการสร้างกฎต่อไปนี้ขึ้นมาเพื่อ

จุดมุ่งหมายประการแรกก็เพื่อต้องการให้หยุดเพื่อตรวจสอบสภาพทางกายภาพ ของจุดที่กำลังจับสัญญาณอยู่ว่า มีปัญหาหรือไม่ เช่น จับผิดขุด จับไม่แน่น หรือ ขาอุปกรณ์หลวมหลุด หรือลัดวงจร เป็นต้น โดยใช้การตรวจสอบ ด้วยการ สังเกต หรือ เครื่องมืออื่นๆ เช่น โอห์มมิเตอร์ เครื่องตามรอยกระแส เป็นต้น

จุดมุ่งหมายประการต่อมาคือ ข้อขัดข้องบางชนิดยากต่อการวินิจฉัยเชิงโลจิกได้ เช่น กรณีที่เป็นโหนดที่มีแฟนเอามากกว่า 1 เมื่อเกิดการลัดวงจรที่โหนดใดโหนดหนึ่ง ในเชิงโลจิกแล้วยากที่จะระบุได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นโหนดใด ดังนั้นจึงต้องอาศัยการตรวจสอบสภาพทางกายภาพ เช่น การใช้โอห์มมิเตอร์, การใช้เครื่องตามรอยกระแสเข้าช่วย เป็นต้น เพื่อเป็นการช่วยเสริมการค้นหาข้อขัดข้องเชิงโลจิก ที่ยังไม่ครอบคลุม หรือยังไม่มีอำนาจการจำแนกพอเพียง จึงสร้างกฎเหล่านี้ขึ้นมาเพื่ออาศัยการตรวจสอบเชิงกายภาพมาช่วยเสริมความถูกต้องแม่นยำในการวินิจฉัย

โยงมาจากกฎ	คุณสมบัติที่ได้รับการถ่ายทอดมา	ผลการตรวจทานใหม่	นำไปเขียนเป็นกฎ	ข้อสรุป
31b	เป็นขาเอาพุต	Yes		Yes = ตรวจสอบแล้วไม่ก่อให้เกิดปัญหาใดๆ ทำการวินิจฉัยต่อไปได้
	ที่ผลการเปรียบเทียบ = No	No	43a	
34b	เป็นขาอินพุตหรือขาควบคุม	Yes		
	ที่ผลการเปรียบเทียบ = No	No	43b	No = ตรวจสอบแล้วปรากฏว่ามีปัญหา หยุดทำการวินิจฉัย
60a	เป็นขาของโหนดที่ต่อร่วม	Yes		
	ที่ผลการเปรียบเทียบ = No	No	70	

ตารางที่ 4.4 การสร้างกฎเกี่ยวกับการวินิจฉัยเชิงกายภาพ

อย่างไรก็ตาม แม้ว่าข้อขัดข้องและข้อผิดพลาดจะถูกค้นพบได้โดยกฎเกี่ยวกับการวินิจฉัยเชิงกายภาพเหล่านี้แล้วก็ตาม แต่ก็ไม่แน่ว่าจะเป็นสาเหตุข้อขัดข้องที่แท้จริงหรือไม่ เช่น ข้อผิดพลาดอันเนื่องมาจากการจับสัญญาณผิดจุด เป็นต้น ดังนั้นในงานวิจัยนี้จึงกำหนดค่าแฟคเตอร์ความแน่นอนไว้ที่ 50(ตามข้อกำหนดของ VP-Expert ดังที่ได้กล่าวมาแล้วในหัวข้อที่ 2.2.4) ซึ่งหมายถึงอาจจะมีสาเหตุอื่นที่เป็นสาเหตุข้อขัดข้องที่แท้จริงก็ได้ จึงได้ให้ผู้ใช้ทำการทดสอบซ้ำอีกครั้งหนึ่ง

4.4.6 กฎที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นสุดท้าย

จากหัวข้อที่ 4.4.2 ตารางการสร้างกฎ จะเห็นว่ากฎ 31b, 34a และ 32a เกิดจากผลการเปรียบเทียบระหว่างสัญญาณตอบสนองที่จับได้กับสัญญาณอ้างอิง แล้วปรากฏว่าได้ไม่เท่ากัน และยังไม่สามารถหาข้อสรุปได้ เนื่องจากยังไม่ทราบว่ามีขาอื่นๆอีกหรือไม่ ที่ขาเหล่านั้นปกติหรือไม่ปกติ ซึ่งคุณสมบัติที่ถูกถ่ายทอดมาเหล่านี้ นำมาสร้างเป็นกฎใหม่ได้ดังตารางที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โยงมาจากกฎ	คุณสมบัติที่ได้รับการถ่ายทอดมา	ชื่อกฎ	ข้อสรุป
31b, 34a, 32a	จับสัญญาณทุกขาได้ปกติหมด แต่ไม่มีสัญญาณเอาพุตออกไป หรือมีสัญญาณเอาพุตที่ไม่ปกติ	42b	จากรูปที่ 4.3 ค แสดงว่า อุปกรณ์ตัวนี้ชำรุด

ตารางที่ 4.5 การสร้างกฎเกี่ยวกับการวินิจฉัยขั้นสุดท้าย

4.4.7 กฎที่ทำหน้าที่เกี่ยวกับการควบคุมลูป

เป็นกฎที่ทำหน้าที่ในการควบคุมการวนลูปต่างๆ ประกอบด้วยกฎต่อไปนี้

กฎ 30. ควบคุมกรณีที่ เส้นทางการค้นหาข้อขัดข้องเกิดเป็นลูป

กฎ 40. ควบคุมการวนลูปเพื่อตรวจสอบโหนดต่างๆที่ต่อร่วมอยู่กับขาใดขาหนึ่ง

กฎ 42a. ควบคุมการเคลื่อนที่ไปจับสัญญาณตามขาต่างๆ ภายในอุปกรณ์ตัวเดียวกัน

กฎ ff. ตรวจสอบการสิ้นสุดข้อมูล

กฎ ffa. ตรวจสอบการสิ้นสุดข้อมูล ระหว่างการวนลูปเพื่อตรวจสอบโหนดต่างๆที่ต่อร่วมอยู่กับขาใดขาหนึ่ง

4.4.8 สรุปหน้าที่ของแต่ละกฎ

สรุปหน้าที่ของแต่ละกฎเรียงตามลำดับชื่อมีดังนี้

10a หน้าที่ : กรองหรือดักรับ ระเบียน ที่มีชื่อของสัญญาณและหน้าที่ของขาอุปกรณ์ตรงตามที่กำหนดไว้เข้ามาทำการอนุมาน

10b หน้าที่ : กรองหรือดักรับ ระเบียน ที่มีชื่อของตัวอุปกรณ์และหน้าที่ของขาอุปกรณ์ตรงตามที่กำหนดไว้เข้ามาทำการอนุมาน

20a. หน้าที่ : แสดงข้อมูลข่าวสารในกรณีที่ การวินิจฉัยยังไม่บรรลุเป้าหมาย แต่ต้องการให้กระทำอย่างหนึ่งอย่างใดก่อน

20b. หน้าที่ : แสดงชื่อตัวอุปกรณ์ ขา และรูปร่างสัญญาณของ ระเบียนที่ถูกกรองหรือดักรับเข้ามา

30 หน้าที่ : แสดงข้อความถ้าเป็น ข้อผิดพลาดประเภท ป้อนกลับ(feedback)

31a หน้าที่ : ทำการอนุมาน เมื่อจุดที่กำลังตรวจสอบเป็น ขาเอาท์พุต และวัดสัญญาณที่ จุดนี้ได้ตามปกติ

31b หน้าที่ : ทำการอนุมาน เมื่อจุดที่กำลังตรวจสอบเป็น ขาเอาท์พุตและวัดสัญญาณที่จุดนี้ได้ผิดปกติ

32a หน้าที่ : ทำการอนุมาน เมื่อจุดที่กำลังตรวจสอบเป็น ไฟเลี้ยง หรือ gnd และวัดสัญญาณที่จุดนี้ได้ตามปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

32b หน้าที่ : ทำการอนุมาน เมื่อจุดที่กำลังตรวจสอบเป็น ไฟเลี้ยง หรือ gnd และวัดสัญญาณที่จุดนี้ได้ผิดปกติ

34a หน้าที่ : ทำการอนุมาน เมื่อจุดที่กำลังตรวจสอบเป็น ขาควบคุม หรือ ขาอินพุต และวัดสัญญาณที่จุดนี้ได้ตามปกติ

34b หน้าที่ : ทำการอนุมาน เมื่อจุดที่กำลังตรวจสอบเป็น ขาควบคุม หรือ ขาอินพุต และวัดสัญญาณได้ผิดปกติ

40 หน้าที่ : ควบคุมการวนรอบลูป หาอุปกรณ์ตัวอื่นๆที่ต่อร่วมอยู่ จุดทดสอบจุดนี้

41 หน้าที่ : คอยตรวจสอบว่า ถ้าจุดที่กำลังตรวจสอบอยู่นี้เป็น ขาควบคุม หรือ ขาอินพุต แล้วจุดนี้ต่ออยู่กับ ไฟเลี้ยง หรือ gnd หรือไม่ ถ้าใช่จะยุติการอนุมาน

42a หน้าที่ : ควบคุมการเคลื่อนที่ในการตรวจสอบจุดถัดไปบนอุปกรณ์ตัวหนึ่ง

42b หน้าที่ : ยุติการเคลื่อนที่ในการ ตรวจสอบจุดถัดไปเนื่องจากพบว่าอุปกรณ์ตัวนี้ชำรุด

43a หน้าที่ : ยุติการอนุมาน เมื่อพบว่า จุดที่กำลังทดสอบอยู่นี้เป็นสัญญาณ เอาท์พุต และไม่ปกติ เช่น เกิดหลวม/หลุด หรือ ลัดวงจร

43b หน้าที่ : ยุติการอนุมาน เมื่อพบว่า จุดที่กำลังทดสอบอยู่นี้เป็นสัญญาณ ขาควบคุม หรือ ขาอินพุต และไม่ปกติ เช่น เกิดหลวม/หลุด หรือ ลัดวงจร และยกเลิก การวนรอบหาอุปกรณ์ตัวอื่นๆที่ต่อร่วมอยู่กับจุดนี้

50 หน้าที่ : คอยตรวจสอบว่า จุดที่กำลังตรวจสอบอยู่นี้มีอุปกรณ์ตัวอื่นๆต่อร่วมอยู่หรือไม่

60a หน้าที่ : คอยตรวจสอบว่าอุปกรณ์ตัวอื่นๆที่ต่อร่วมอยู่กับจุดที่กำลังทดสอบอยู่นี้มีตัวใดบ้างที่ไม่ปกติ เช่น เกิดหลวม/หลุด หรือ ลัดวงจร

60b หน้าที่ : คอยตรวจสอบว่าอุปกรณ์ตัวอื่นๆที่ต่อร่วมอยู่กับจุดที่กำลังทดสอบอยู่นี้มีตัวใดบ้างที่วัดสัญญาณได้ตามปกติ

70 หน้าที่ : ยุติการอนุมาน เมื่อพบว่า มีอุปกรณ์ตัวอื่นๆตัวใดตัวหนึ่งซึ่งต่อร่วมอยู่กับจุดนี้ ไม่ปกติเช่น เกิดการลัดวงจร

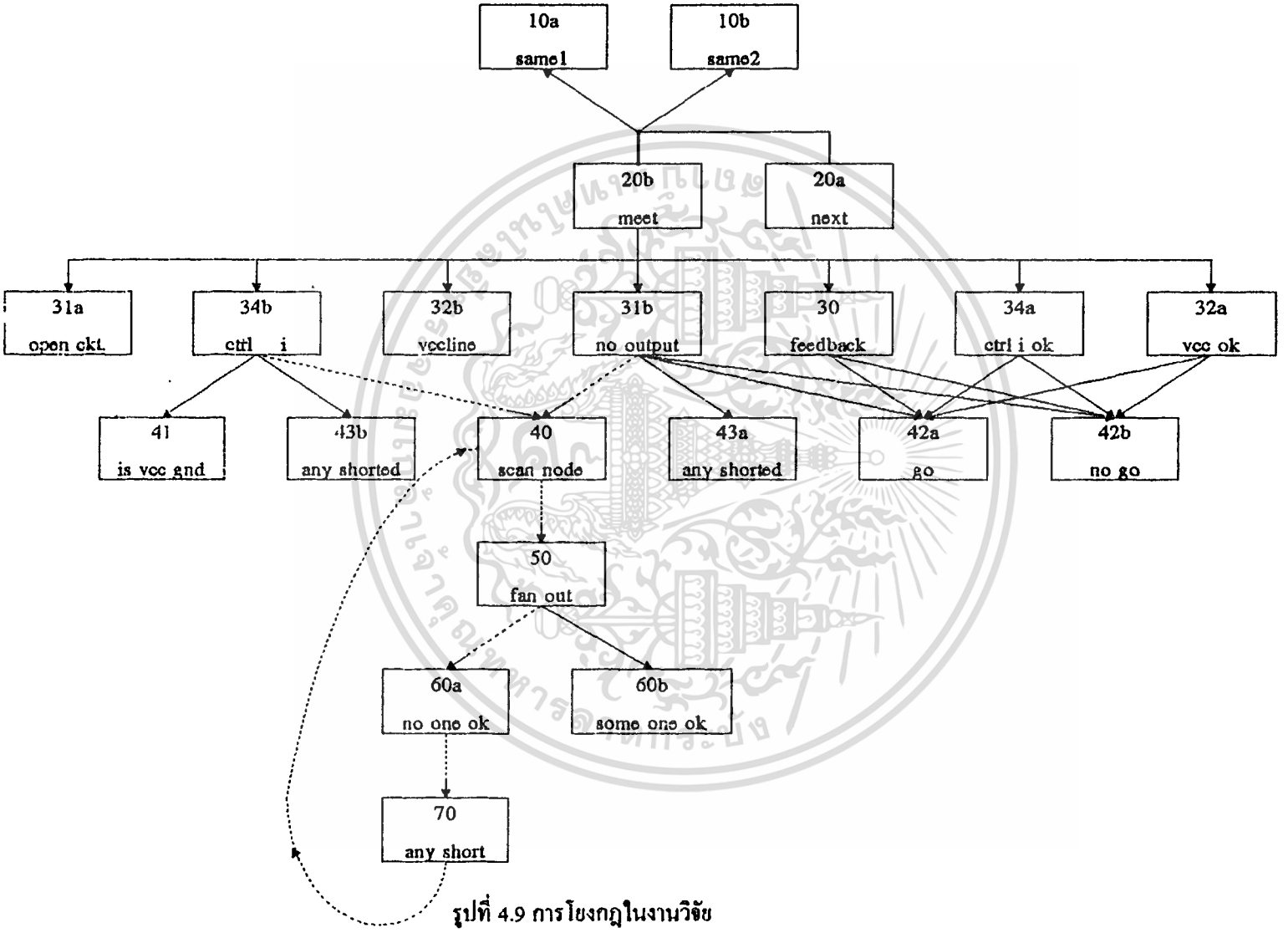
dummy1. หน้าที่ : แสดงข้อความให้ตรวจสอบทางกายภาพ

ff. หน้าที่ :. ตรวจสอบการสิ้นสุดข้อมูล

ffa. หน้าที่ : ตรวจสอบการสิ้นสุดข้อมูล ระหว่างการวนลูปเพื่อตรวจสอบโหนดต่างๆที่ต่อร่วมอยู่กับขาใดขาหนึ่ง

สำหรับรายละเอียดของกฎเหล่านี้จะอยู่ในแฟ้มการแสดงความรู้(rules base) ชื่อ CHIP.KBS ซึ่งอยู่ในภาคผนวกที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 การโยงกฎในงานวิจัย

4.5 ทิศทางการอนุมาน

ดังที่ได้กล่าวมาแล้วในบทนำว่าปัญหา " การค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ " เป็นปัญหาเชิงวิเคราะห์ ปัญหาเริ่มต้นจากอาการข้อขัดข้องที่ปรากฏ และทำการแก้ไขปัญหาด้วยการสืบสวนจากอาการข้อขัดข้องย้อนกลับไปหาสาเหตุ ลักษณะที่สำคัญประการหนึ่งของวงจรหลักไมโครคอมพิวเตอร์คือ อาการข้อขัดข้องที่ปรากฏขึ้นอย่างใดอย่างหนึ่ง สามารถมีสาเหตุได้มากมายหลายสาเหตุ หรือกล่าวอีกอย่างหนึ่งว่าเป็นปัญหาที่มีเป้าหมายเป็นไปได้หลายเป้าหมาย ดังนั้นทิศทางการค้นหาแบบเดินหน้าจึงมีความเหมาะสมมากกว่าแบบถอยหลัง โดยในขั้นแรกจะทำการวิเคราะห์ปัญหาแยกออกเป็นระดับต่างๆ (ดังกล่าวมาแล้วในหัวข้อที่ 2.4.2) เมื่อสามารถระบุบริเวณที่จะทำการค้นหาได้แล้ว จึงทำการวินิจฉัยโดยใช้ทิศทางการหาเหตุผลแบบเดินหน้า (Forward chaining โดยใช้ Topdown refinement) ในรูปที่ 4.9 แสดงให้เห็นลำดับขั้น(hierarchi) การโยง(chaining)กฎต่างๆเมื่อทำการดำเนินการอนุมานตามลักษณะ ของกฎที่ได้สร้างขึ้นตามหัวข้อที่ 4.4

จากรูปที่ 4.9 ได้แบ่งกฎเป็นระดับต่างๆ ตามลักษณะการใช้งานหรือการโยง ทั้งนี้เพื่อสะดวกในการอธิบายและความเข้าใจ โดยกฎที่อยู่ในระดับใดก็จะขึ้นต้นด้วยเลขนั้น เช่น ระดับ 3 ก็จะขึ้นต้นด้วย 30, 31a, 31b, 32a,.... เป็นต้น สำหรับกฎที่โยงด้วยเส้นประ หมายถึงการโยงเข้าสู่การลูปเพื่อทำการค้นหาโหนดที่ต่อร่วมอยู่กับจุดนั้นๆ และป้ายชื่อที่อยู่ภายในบล็อกใต้ตัวเลขชื่อกฎ ก็คือส่วนข้อสรุปที่อยู่หลัง THEN นั่นเองโดยในแต่ละบล็อกหมายถึงกฎหนึ่งกฎ และการทำงานในแต่ละวงรอบ (loop)มีดังนี้

1. กฎในระดับที่ 2 (rules 20a,20b) จะถูกเรียกขึ้นมาก่อนเป็นลำดับแรก เพื่อต้องการให้แสดงข้อมูลที่ใช้เป็นคำสั่ง หรือข้อมูลที่ใช้เป็นตัวอ้างอิง แต่การที่ rule 20 จะแสดงข้อมูลได้นั้นจะต้องโยงย้อนหลัง(backward chain)ไปยังกฎในระดับที่ 1 ก่อน (rules 10a, 10b) เพื่อให้ค้นหาว่าข้อมูลเหล่านั้นมีอยู่ในแฟ้มความรู้หรือไม่

2. ถ้ากฎในระดับที่ 1 พบว่าในระเบียนปัจจุบัน มีข้อมูลที่ไม่สอดคล้องตามเงื่อนไข ก็จะส่งสัญญาณมาบอกแก่ rule 20 ว่า ข้อมูลไม่สอดคล้อง ทำให้ rule 20 ไม่สามารถแสดงข้อมูลที่ใช้เป็นตัวอ้างอิงออกมาได้ การค้นหาก็จะดำเนินต่อไปในระเบียนถัดไป จนกว่าจะพบข้อมูลที่สอดคล้องตามเงื่อนไข หรือหมดแฟ้มความรู้

ถ้ากฎในระดับที่ 1 พบว่าในระเบียนปัจจุบัน มีข้อมูลสอดคล้องตามเงื่อนไขแล้วก็จะส่งคำตอบมาให้ rule ในระดับที่ 2 แสดงผลเป็นข้อมูลอ้างอิงให้ทำการเปรียบเทียบต่อไป

3. การอนุมานในขั้นต่อไปหลังจากที่ข้อมูลอ้างอิงถูกค้นพบแล้ว จะเป็นการให้ผู้ใช้ทำการจับสัญญาณตอบสนองจากแผงวงจรที่ทำการทดสอบ แล้วทำการตอบคำถามว่าเหมือนกับข้อมูลอ้างอิงหรือไม่ (Yes/No)

4. กฎในระดับที่ 3 เป็นต้นไป (rules 30 - 70) จะนำคำตอบของผู้ใช้มาอนุมานค้นหาสาเหตุข้อขัดข้องแบบวิธีที่เหมาะสมก่อน ตามที่ได้กล่าวมาแล้วในหัวข้อที่ 2.4.4 ซึ่งถ้าคำตอบของผู้ใช้และข้อมูลในปัจจุบัน ยังไม่พอเพียงที่จะสรุปหาคำตอบได้ การทำงานก็จะวนลูปไปเรียกใช้กฎในระดับที่ 2 เพื่อค้นหาข้อมูลและโยงไปตามกฎต่างๆเพื่อค้นหาคำตอบต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 4.3 ตัวอย่างการอนุมาน

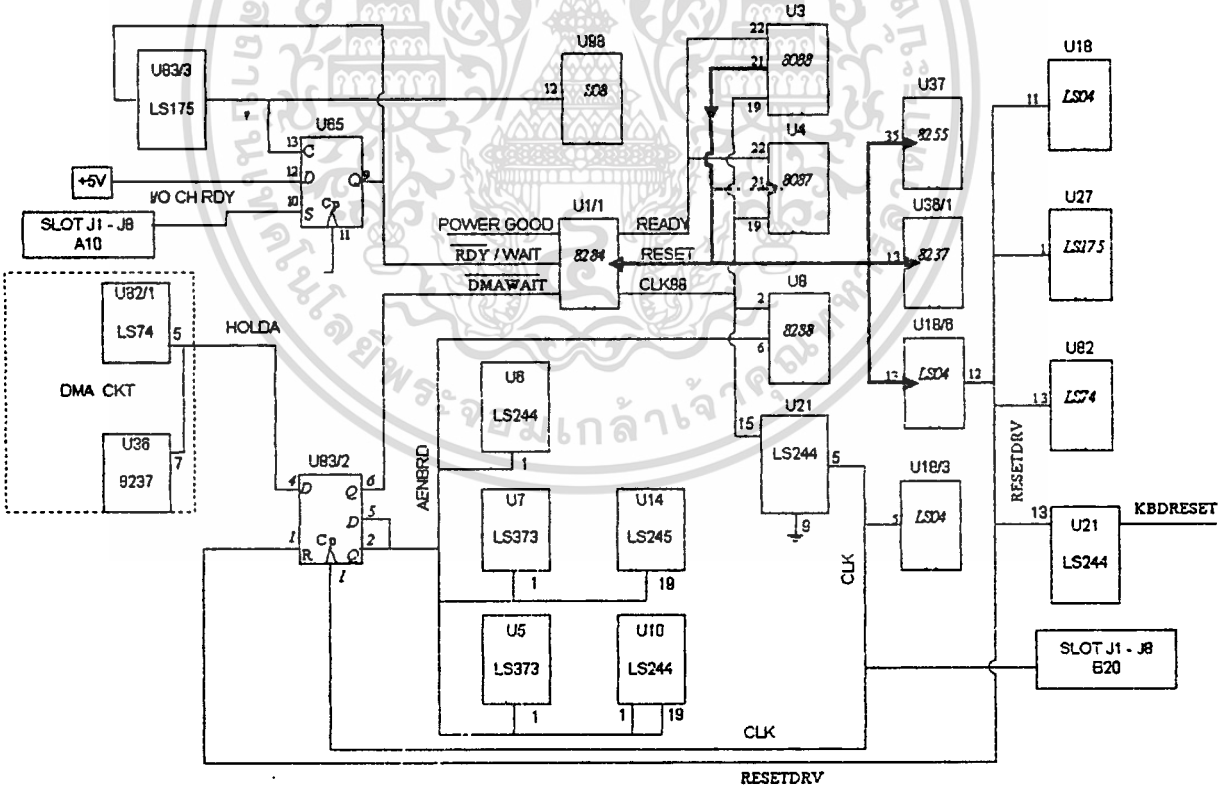
เพื่อแสดงตัวอย่างการทำงานของกลไกการอนุมาน ในที่นี้จะใช้วงจร RESET ในรูปที่ 4.10 จากการทดสอบที่ 1 มาเป็นตัวอย่าง โดยมีตารางข้างล่างนี้เป็นส่วนหนึ่งของแฟ้มข้อมูลที่ใช้ในการค้นหาข้อขัดข้อง

สมมติว่า U3(8088) ไม่ทำงาน และได้ทำการจับสัญญาณบนขาต่างๆแล้วปรากฏว่าสัญญาณ reset ไม่ปกติ จากวงจร การค้นหาข้อขัดข้องจะเริ่มต้นจาก U3 ขา 21

ในขั้นเตรียมการก่อนที่จะเข้าสู่วงรอบการอนุมาน จะมีดัชนี x และ y เป็นตัวแสดงค่าหมายเลขระเบียนปัจจุบันและหมายเลขระเบียนถัดไปของแฟ้มข้อมูลที่กำลังถูกดำเนินการอยู่ โดยกำหนดให้

$$y = (x + 1)$$

และมีตัวแปร 2 ตัวทำหน้าที่เป็นเครื่องนำทางไปยังจุดต่างๆ โดยกำหนดไว้ว่าชื่อของสัญญาณชื่อแรก และฟังก์ชั่นแรกในแฟ้มข้อมูล จะถูกป้อนให้กับตัวแปรทั้งสอง คือ



รูปที่ 4.10 วงจร RESET ประกอบตัวอย่างการอนุมาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REC	NODE	PIN	FUNC	G	SIGNAME
1	u3	21	ctrl	5	reset
2	u3	19	ctrl	8	clk88
3	u3	22	ctrl	8	ready
4	u3	...	ctrl	9	NEXT
5	u4	19	ctrl	8	clk88
6	u4	21	ctrl	5	reset
7	u4	22	ctrl	8	ready
8	u8	2	ctrl	8	clk88
9	u21	15	i	9	clk88
10	u21	13	i	9	_resetdrv
11	u1/2	8	o	0	clk88
12	u1/2	2	o	0	pclk
13	u1/2	12	o	0	osc
14	u1/2	13	ctrl	8	gnd
15	u1/2	1	ctrl	8	gnd

REC	NODE	PIN	FUNC	G	SIGNAME
16	u1/2	16	i	9	oscxtal
17	u1/2	17	i	9	oscxtal
18	R1	0	o	0	oscxtal
19	R1	1	i	9	gnd
20	R2	0	o	0	oscxtal
21	R2	1	i	9	gnd
22	xtal		o	0	oscxtal
23	u18/6	13	i	9	reset
24	u36/1	13	ctrl	5	reset
25	u37	35	ctrl	5	reset
26	u1/1	10	o	0	reset
27	u1/1	11	i	9	vcc
28	u1/3	5	o	0	ready
29	u1/3	7	ctrl	8	vcc
30	u1/3	3	i	9	_rdy/wait

ตารางที่ 4.6 ประกอบตัวอย่างที่ 4.3 การอนุมาน

search = reset

pin_func = ctrl

เมื่อเข้าสู่วงรอบการอนุมานรอบที่ 1

กฎ 20 b จะโยงไปยังกฎ 10 a เพื่อทำการค้นหาข้อมูลที่สอดคล้องกับตัวแปร search และ pin_func ซึ่งผลการค้นหาจะพบได้จากระเบียนที่ 1 ซึ่งมีข้อมูลดังนี้

REC	NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
1	u3	21	ctrl	5	⌐ OR	reset

จากนั้นจะทำการแสดงข้อมูลในช่อง T_DIAGM จากระเบียนที่ 1 นี้ เพื่อให้ผู้ใช้ทำการเปรียบเทียบกับสัญญาณตอบสนองที่จับได้ ผลการเปรียบเทียบจะถูกโยงไปยังกฎต่างๆที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และสมมติว่าสัญญาณตอบสนองที่จับได้รูปร่างไม่เหมือนกับข้อมูลในช่อง T_DIAGM ดังนั้นกฎที่จะถูกโยงไปหา คือ กฎ 34 b

กฎ 34 b จะทำการตรวจสอบดังนี้

ตรวจสอบสัญญาณที่กำลังวินิจฉัยอยู่ขณะนี้ว่าต่อกับไฟเลี้ยง หรือ gnd หรือไม่ โดยโยงไปยังกฎ 41 จากตารางจะเห็นว่าข้อมูลในช่อง SIGNAME <> vcc หรือ gnd

ตรวจสอบสภาพทางกายภาพของ ขา 21 U3 โดยโยงไปยังกฎ 43b และสมมติว่าปรากฏว่าปกติ

เมื่อการตรวจสอบดำเนินมาถึงขั้นนี้แสดงว่าข้อขัดข้องไม่ได้มีสาเหตุมาจาก ขา 21 U3 ดังนั้นการอนุมานในรอบต่อไป ก็คือทำการค้นหาว่าสัญญาณ reset นี้ ออกมาจากเอาท์พุตของอุปกรณ์ตัวใด โดยป้อนค่า "o" ให้กับตัวแปร pin_func ดังนั้นตัวแปรทั้งสองในขณะนี้ จึงมีค่าเป็น

search = reset

pin_func = o

แต่ก่อนที่จะทำการอนุมานในรอบต่อไป จำเป็นที่จะต้องตรวจสอบโหนดต่างๆที่ต่อร่วมอยู่กับ U3 ขา 21 นี้เสียก่อน โดยโยงไปที่กฎ 40

กฎ 40 ทำการควบคุมลูป

โดยเริ่มต้นกำหนดให้ค่าตัวแปรควบคุมเท่ากับ 1 แล้วในทุกๆวงรอบจะเพิ่มค่าตัวแปรทีละ 1 ซึ่งเขียนเป็นอนุประโยคหลัง THEN ได้ดังนี้

x = 1

.....

x = x + 1

จากนั้นจะโยงไปที่กฎ 50 ทำการคัดเลือกข้อมูลที่มีเงื่อนไขสอดคล้องกับ

search = reset และ

pin_func <> o

ซึ่งหมายถึงโหนดทุกโหนดที่มีขาต่ออยู่กับสัญญาณ reset แต่มีฟังก์ชั่นขานั้นไม่ใช่เอาท์พุต การคัดเลือกดำเนินไปจากระเบียนที่ 1, 2, ... จนถึงระเบียนที่ 6 ซึ่งเป็นระเบียนแรกที่สอดคล้องตามเงื่อนไขและมีข้อมูลดังนี้

REC	NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
6	u4	21	ctrl	5	┐..... OR	reset

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะทำการแสดงข้อมูลในช่อง T_DIAGM จากระเบียนที่ 6 นี้ เพื่อให้ผู้ใช้ทำการเปรียบเทียบกับสัญญาณตอบสนองที่จับได้ โดยผลของการเปรียบเทียบจะถูกโยงไปยังกฎ 60b หรือ 60a ขึ้นอยู่กับว่าสัญญาณที่จับได้เหมือนกับข้อมูลในช่อง T_DIAGM จากระเบียนที่ 6 นี้ หรือไม่ สมมติว่าให้ผลการเปรียบเทียบไม่เหมือนกัน ดังนั้นกฎที่ถูกโยงไปหาคือ กฎ 60a

กฎ 60a จะแสดงข้อความให้ตรวจสอบที่ขา 21 ของ U4 สมมติให้ผลการตรวจสอบ ปรากฏว่าปกติ

เมื่อผลของการตรวจสอบของกฎ 60a เป็นปกติ ก็จะทำให้การตรวจสอบโหนดร่วมยังคงดำเนินการต่อไป ตามวงรอบการควบคุมของกฎ 40

การคัดเลือกดำเนินต่อมาจากระเบียนที่ 7, 8, ...จนกระทั่งถึงระเบียนที่ 24 ซึ่งเป็นระเบียนที่สอดคล้องตามเงื่อนไขและมีข้อมูลดังนี้

REC	NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
24	u36/1	13	ctrl	5	OR	reset

จากนั้นการดำเนินการก็จะกระทำซ้ำเช่นเดียวกับการดำเนินการต่อข้อมูลในระเบียนที่ 6 ดังที่ได้กล่าวมาแล้ว จนถึงระเบียนที่ 25

REC	NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
25	u37	35	ctrl	5	OR	reset

หลังจากนั้นก็จะไม่มีข้อมูลใดในแฟ้มข้อมูลสอดคล้องกับเงื่อนไขอีก จนถึงระเบียนสุดท้าย กิจกรรมการตรวจสอบโหนดร่วมของกฎ 34b ก็สิ้นสุดลง และกลับไปยังวงรอบการอนุมานรอบที่ 2 ต่อไป

ก่อนที่จะเข้าสู่วงรอบการอนุมานรอบที่ 2 ตัวแปร search และ pin_func มีค่าดังนี้

search = reset

pin_func = o

เมื่อเข้าสู่วงรอบการอนุมานรอบที่ 2

กฎ 20b จะโยงไปยังกฎ 10a เพื่อทำการค้นหาข้อมูลที่สอดคล้องกับตัวแปร search และ pin_func ซึ่งผลการค้นหาจะพบได้จากระเบียนที่ 26 ซึ่งมีข้อมูลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REC	NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
26	u1/1	10	o	0	OR	reset

จากนั้นจะทำการแสดงข้อมูลในช่อง T_DIAGM จากระเบียนที่ 26 เพื่อให้ผู้ใช้ทำการเปรียบเทียบ กับสัญญาณตอบสนองที่จับได้ ผลของการเปรียบเทียบจะถูกโยงไปยังกฎต่างๆที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 1 และสมมติว่าสัญญาณตอบสนองที่จับได้รูปร่างไม่เหมือนกับข้อมูลในช่อง T_DIAGM ดังนั้นกฎที่จะถูกโยงไปหา คือ กฎ 31b

กฎ 31b จะทำการตรวจสอบดังนี้

ตรวจสอบสภาพทางกายภาพของ ขา 10 U1/1 โดยโยงไปยังกฎ 43a และสมมติว่าปรากฏว่าปกติ แสดงว่าข้อขัดข้องไม่ได้มีสาเหตุมาจาก ขา 10 U1/1 ดังนั้นการอนุมานในรอบต่อไป ก็คือทำการตรวจสอบว่าสัญญาณควบคุมของ U1/1 ปกติหรือไม่

แต่ก่อนที่จะทำการอนุมานในรอบต่อไป จำเป็นที่จะต้องตรวจสอบโหนดต่างๆที่ต่อร่วมอยู่กับ U1/1 ขา 10 นี้เสียก่อนโดยป้อนค่า "don't care" ให้กับตัวแปร pin_func ทั้งนี้เพื่อให้ทำการตรวจสอบทุกโหนดไม่ยกเว้น เอาท์พุต เหมือนการอนุมานรอบแรก ดังนั้นตัวแปรทั้งสองในขณะนี้ จึงมีค่าเป็น

search = reset

pin_func = not_care

หลังจากนั้นจึงโยงไปที่กฎ 40

กฎ 40 ทำการควบคุมลูป

โดยเริ่มต้นกำหนดให้ค่าตัวแปรควบคุมเท่ากับ 1 แล้วในทุกๆวงรอบจะเพิ่มค่าตัวแปรทีละ 1 เช่นเดียวกับที่กล่าวมาแล้วในการอนุมานรอบแรก

จากนั้นจะโยงไปที่กฎ 50 ทำการคัดเลือกข้อมูลที่มีเงื่อนไขสอดคล้องกับ

search = reset และ

pin_func <> not_care

ซึ่งหมายถึงโหนดทุกโหนดที่มีขาต่ออยู่กับสัญญาณ reset โดยไม่จำกัดฟังก์ชั่นขา การคัดเลือกดำเนินไปจากระเบียนที่ 1, 2, ... เรื่อยไป แต่จะไม่มีระเบียนใดที่สอดคล้องตามเงื่อนไขนี้เลย เนื่องจากว่าทุกครั้งที่มีการตรวจสอบ ณ ที่ระเบียนใดก็ตาม จะมีการทำเครื่องหมายไว้ทุกระเบียน ซึ่งแสดงอยู่ในรูปอนุประโยคได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

....

signal[x] = blank

.....

ดังนั้น ทุกๆโหนดที่มีขาต่อร่วมอยู่กับ ขา 10 ของ U1/1 ก็คือ โหนดทุกโหนดที่มีขาต่อร่วมอยู่กับ ขา 1 ของ U3 ซึ่งได้ทำการตรวจสอบไปแล้วนั่นเอง แต่เหตุที่ต้องให้มีการตรวจสอบโหนดที่ต่อร่วมกับขาเอาต์พุตอีกก็เพราะว่า ต้องการให้ใช้ได้กับทุกๆกรณี เช่น กรณีที่เริ่มต้นการอนุมานจากขาเอาต์พุต ไม่ใช่ขาควบคุมเหมือนกรณีนี้ เป็นต้น

เมื่อไม่มีระเบียนใดสอดคล้องตามเงื่อนไขเลย กิจกรรมการตรวจสอบโหนดร่วมของกฎ 31b ก็สิ้นสุดลง และเตรียมที่จะเข้าสู่วงรอบการอนุมานรอบที่ 3 ต่อไป โดยการโยงไปยังกฎ 42a หรือ 42b ขึ้นอยู่กับว่าระเบียนข้อมูลขณะนี้เป็นระเบียนสุดท้ายของโหนดที่กำลังตรวจสอบอยู่หรือไม่ จากตารางจะเห็นว่ายังไม่ใช่ระเบียนสุดท้าย ดังนั้นจึงโยงไปที่กฎ 42a

กฎ 42a ทำหน้าที่ควบคุมการเคลื่อนที่ไปยังขาต่างๆภายในตัวอุปกรณ์เดียวกัน

โดยจะทำการกำหนดค่าตัวแปร search และ pin_func ดังนี้

search = (xnode[x])

pin_func = (funct[y])

ซึ่ง y = (x + 1) ดังที่ได้กล่าวมาแล้ว และจากตารางจะได้ว่า

pin_func = i

ก่อนที่จะเข้าสู่วงรอบการอนุมานรอบที่ 3 ตัวแปร search และ pin_func มีค่าดังนี้

search = reset

pin_func = i

เมื่อเข้าสู่วงรอบการอนุมานรอบที่ 3

กฎ 20b จะโยงไปยังกฎ 10a เพื่อทำการค้นหาข้อมูลที่สอดคล้องกับตัวแปร search และ pin_func ซึ่งผลการค้นหาจะพบได้จากระเบียนที่ 26 ซึ่งมีข้อมูลดังนี้

REC	NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
27	u1/1	11	i	9	$\overline{\text{XXXXXXXXXXXX}}$	vcc

หมายเหตุ $\overline{\text{XXXXXXXXXXXX}}$ หมายถึงสถานะโลจิกที่ไม่สนใจ (don't care)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะทำการแสดงข้อมูลในช่อง T_DIAGM จากระเบียนที่ 27 เพื่อให้ผู้ใช้ทำการเปรียบเทียบ กับสัญญาณตอบสนองที่จับได้ ผลของการเปรียบเทียบจะถูกโยงไปยังกฎต่างๆที่ทำหน้าที่เกี่ยวกับการวินิจฉัยขั้นที่ 1 และสมมติว่าสัญญาณตอบสนองที่จับได้รูปร่างไม่เหมือนกับข้อมูลในช่อง T_DIAGM ดังนั้นกฎที่จะถูกโยงไปหา คือ กฎ 34b

กฎ 34 b จะทำการตรวจสอบดังนี้

ตรวจสอบสัญญาณที่กำลังวินิจฉัยอยู่ขณะนี้ว่าต่อกับไฟเลี้ยง หรือ gnd หรือไม่ โดยโยงไปยังกฎ 41 จากตารางจะเห็นว่าข้อมูลในช่อง SIGNAME = vcc สอดคล้องตามเงื่อนไข ดังนั้นจึงได้ข้อสรุปจากกฎ 41 ว่า

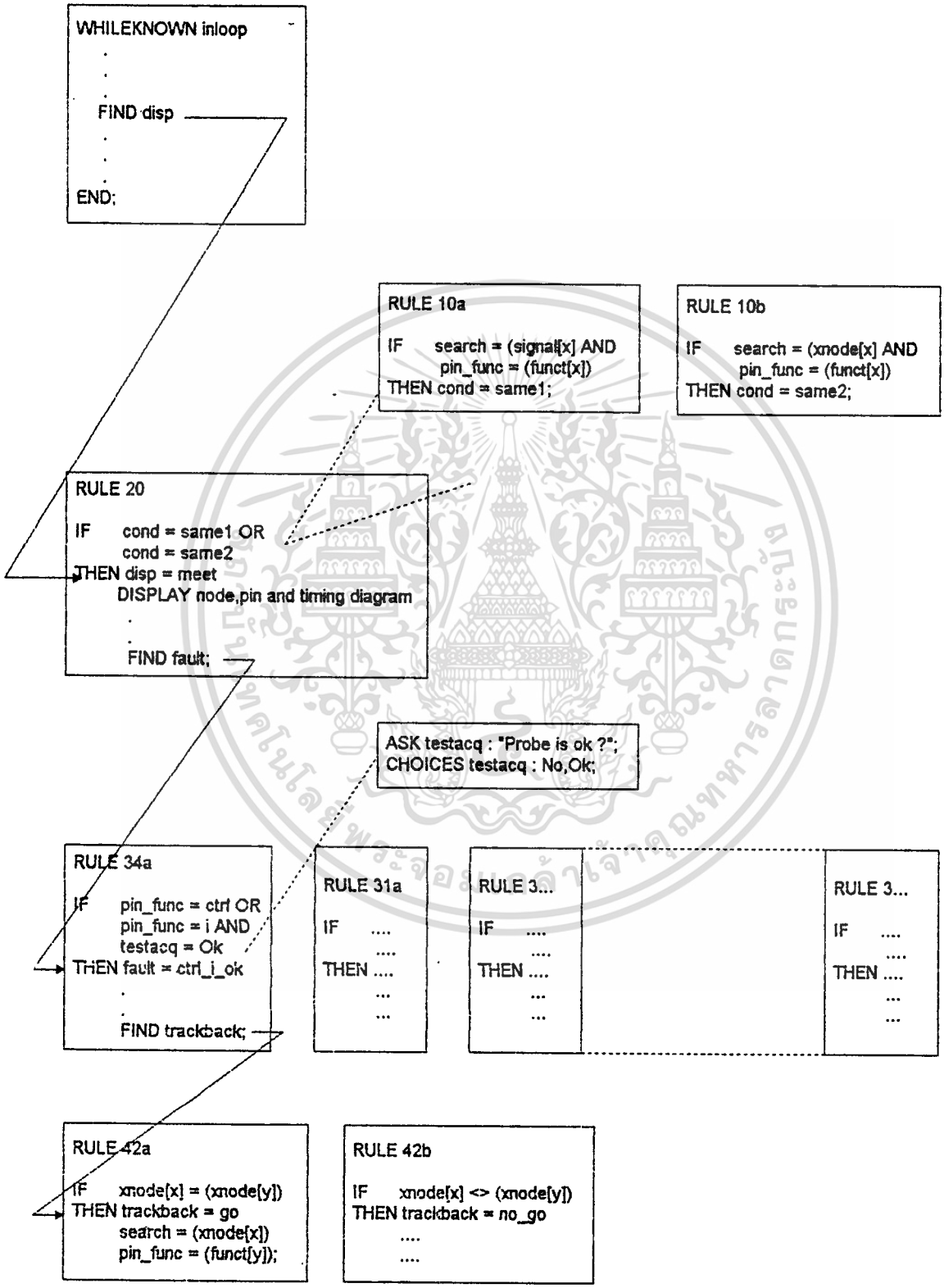
DISPLAY "{xnode[x]} pin{xpin[x]} or circuit. May be open/loose."

ซึ่งหมายความว่า U1/1 ขา 11 หรือ สายวงจร อาจจะขาดหรือ หลวมหลุด

การอนุมานจึงสิ้นสุดลง

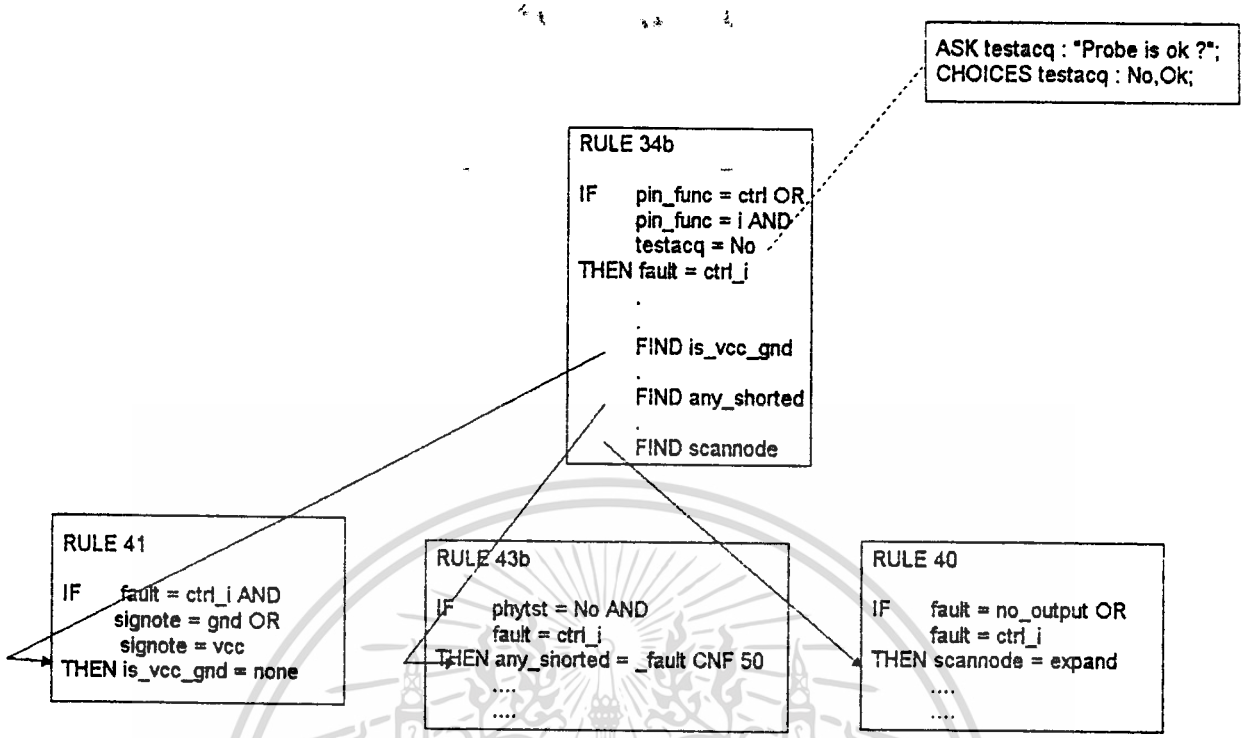
จากตัวอย่างที่ 4.3 นี้ แสดงให้เห็นเกี่ยวกับ จุดเริ่มต้น, การติดตาม และการวินิจฉัยปัญหาข้อขัดข้อง โดยเฉพาะการวนลูปเพื่อค้นหาข้อขัดข้องที่อาจจะเกิดจากโหนดอื่นๆที่ต่อร่วมอยู่กับจุดที่กำลังพิจารณาอยู่ ในรูปที่ 4.11 แสดงแผนผังการทำงานของกลไกการอนุมาน ในการโยงไปยังกฎต่างๆที่เกี่ยวกับการวินิจฉัยสัญญาณตอบสนองที่จับได้ บนขาควบคุม หรือ ขาอินพุตของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ก แผนผังการโยงกฎ เมื่อขับสัญญาณตอบสนอง บนขา ควบคุม หรือ อินพุต ได้ปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ข แผนผังการโยงกฎ เมื่อขับสัญญาณตอบสนอง บนขา ควบคุม หรือ อินพุต ได้ผิดปกติ

4.6 การควบคุมกลยุทธ

4.6.1 การเลือกกฎขึ้นมาแสดง

สำหรับการเลือกกฎขึ้นมาแสดง และใช้งานมีลำดับการเลือก ดังนี้

ก. เลือกตามลำดับความละเอียดของส่วน IF ถ้ากฎใดมีเงื่อนไขของส่วน IF ละเอียดถูกต้องมากที่สุด จะได้รับเลือกก่อน

ข. เลือกตามลำดับก่อน - หลัง ถ้ากฎที่มีเงื่อนไขของส่วน IF ละเอียดถูกต้องเท่ากัน กฎที่พบก่อนเป็นลำดับแรกจะได้รับการเลือกก่อน

ตัวอย่างที่ 4.4 แสดงการเลือกกฎขึ้นมาแสดง

ในแฟ้มความรู้ chip.kbs มีกฎที่ทำหน้าที่เกี่ยวกับการแสดงข้อมูลข่าวสาร และ กฎที่ทำหน้าที่เกี่ยวกับการแสดงข้อมูลสัญญาณอ้างอิง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RULE 20a
IF   cond = same1 OR
     cond = same2 AND
     next_dbf = (signal[x])
THEN disp = nextfile
.............. ..............

RULE 20b
IF   cond = same1 OR
     cond = same2
THEN disp = meet
.............. ..............
```

จากกฎ 20a และ 20b จะเห็นว่าถ้าข้อมูลมีเงื่อนไขสอดคล้อง cond = same1 หรือ same2 แล้ว ก็ควรที่จะผ่านกฎ 20a มาแสดงที่ กฎ 20b ได้ทุกกฎ แต่เนื่องจากกฎ 20a มาก่อนกฎ 20b และกฎ 20a ก็มีความละเอียดของส่วน IF มากกว่ากฎ 20b ด้วย ดังนั้นข้อมูลที่มีเงื่อนไขสอดคล้อง cond = same1 หรือ same2 และ next_dbf จึงถูกแสดงที่กฎ 20a แทน

4.6.2 การควบคุมวงรอบการอนุมาน

ขั้นตอนที่ 1

การควบคุมวงรอบการอนุมานขั้นตอนที่ 1 อาศัยลักษณะของข้อมูลที่ได้มาจากกฎข้อ 10a หรือ 10b และ จากคำตอบ (Yes/No) ของผู้ใช้ในกฎข้อ 20b จะทำให้ทราบว่าขณะนี้จุดที่กำลังทดสอบต่ออยู่กับสัญญาณแบบใด ดังต่อไปนี้ เอาท์พุต, ไฟเลี้ยง, gnd, ควบคุม หรือ อินพุต ดังแสดงในรูปที่ 4.12 ก จากนั้นการทำงานก็จะโยงไปยังกฎต่างๆ ตามลักษณะของสัญญาณดังกล่าว ดังนี้

1. ไฟเลี้ยงและ gnd (รูปที่ 4.12 ข)

1.1 การทำงานจะโยงไปยังกฎข้อ 32b ถ้าผู้ใช้ตอบว่าผลตรวจวัดผิดจากข้อมูลอ้างอิง ซึ่งหมายความว่าเกิดการหลวมหลุดหรือลัดวงจรและจะยุติการอนุมาน

1.2 การทำงานจะโยงไปยังกฎข้อ 32a ถ้าผู้ใช้ตอบว่าผลตรวจวัดได้ตามข้อมูลอ้างอิงและการทำงานก็จะโยงไปยังกฎข้อ 42a หรือ 42b ซึ่งจะได้อธิบายต่อไปในขั้นตอนที่ 2

2. สัญญาณ output (รูปที่ 4.12 ค)

2.1 การทำงานก็จะโยงไปยังกฎ 31a ถ้าผู้ใช้ตอบว่าสัญญาณที่วัดได้เหมือนกับสัญญาณอ้างอิง ซึ่งหมายความว่าเกิดวงจรเปิดระหว่างจุดที่กำลังตรวจวัดอยู่ขณะนี้กับจุดที่ตรวจวัดก่อนหน้านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ถ้าผู้ใช้ตอบว่าสัญญาณที่วัดได้ ผิดพลาดไปจากข้อมูลที่อ้างอิง การทำงานจะโยงไปยังกฎข้อ 31b และจะบอกให้ผู้ใช้ทำการตรวจสอบดูว่ามีการหลวมหลุด หรือลัดวงจร ณ ที่จุดนี้หรือไม่ จากนั้นการทำงานจะโยงไปยังกฎข้อ 43a

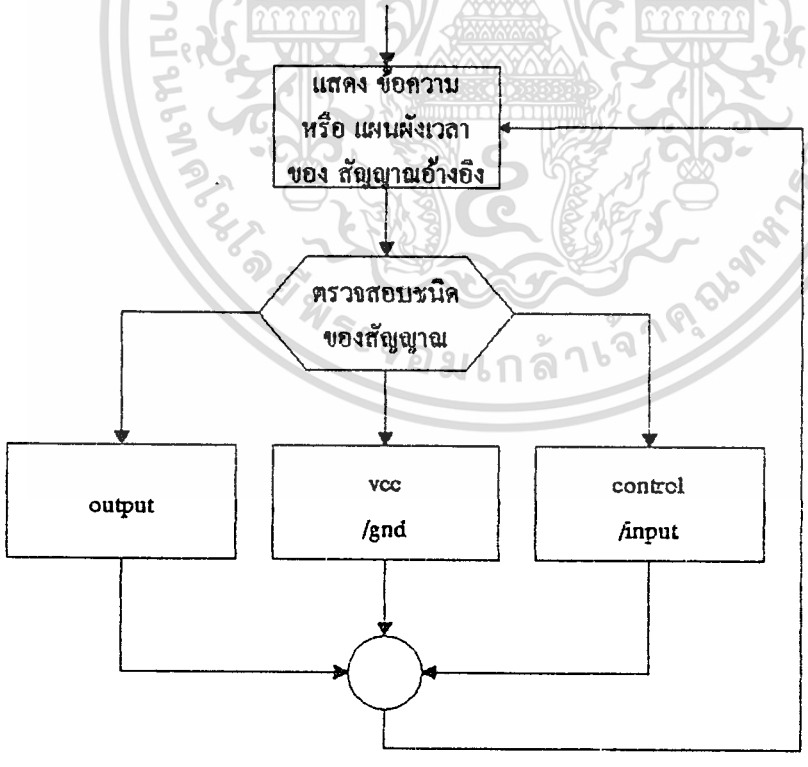
2.2.1 กฎข้อ 43a จะยุติการอนุมาน ถ้าผลการตรวจสอบของผู้ใช้ปรากฏว่าเกิดการ หลวมหลุด หรือลัดวงจร

แต่ถ้าผลการตรวจสอบของผู้ใช้ปรากฏว่าไม่มีการหลวมหลุด หรือลัดวงจรแล้วจากกฎข้อ 31bจะโยงต่อไปยังกฎข้อ 40

2.2.2 กฎข้อ 40 จะทำหน้าที่ควบคุมวงรอบการตรวจสอบว่ามีอุปกรณ์ตัวอื่นๆต่อร่วมกับจุดนี้หรือไม่ โดยแต่ละวงรอบจะโยงไปยังกฎข้อ 50

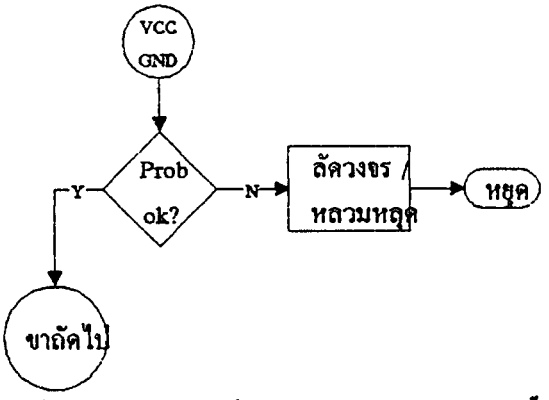
กฎข้อ 50 จะแสดงข้อมูลอ้างอิงของอุปกรณ์ ถ้าพบว่าอุปกรณ์ตัวนั้นต่อร่วมอยู่กับจุดที่กำลังตรวจสอบพร้อมทั้งให้ผู้ใช้ทำการเปรียบเทียบกับสัญญาณที่จับได้ จากนั้นจะโยงไปยังกฎข้อ 60a หรือ 60b ขึ้นอยู่กับคำตอบของการเปรียบเทียบ

กฎข้อ 60b จะแสดงข้อความว่าเกิดวงจรเปิด ถ้าผลการเปรียบเทียบจากกฎข้อ 50 สัญญาณที่จับได้กับสัญญาณอ้างอิงเหมือนกัน

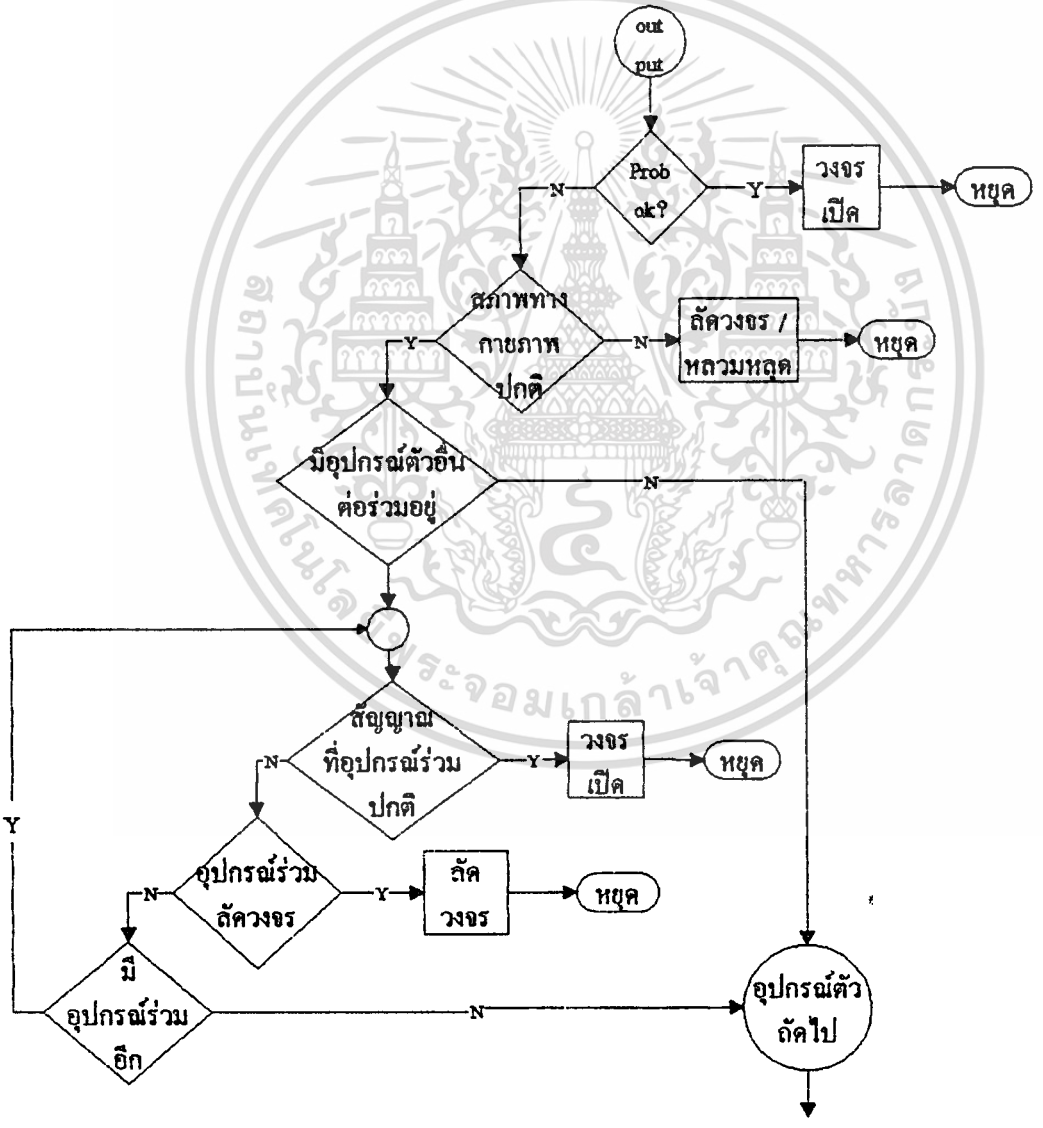


รูปที่ 4.12 ก. ผังงาน ภาพรวม การอนุมานค้นหาสาเหตุข้อขัดข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

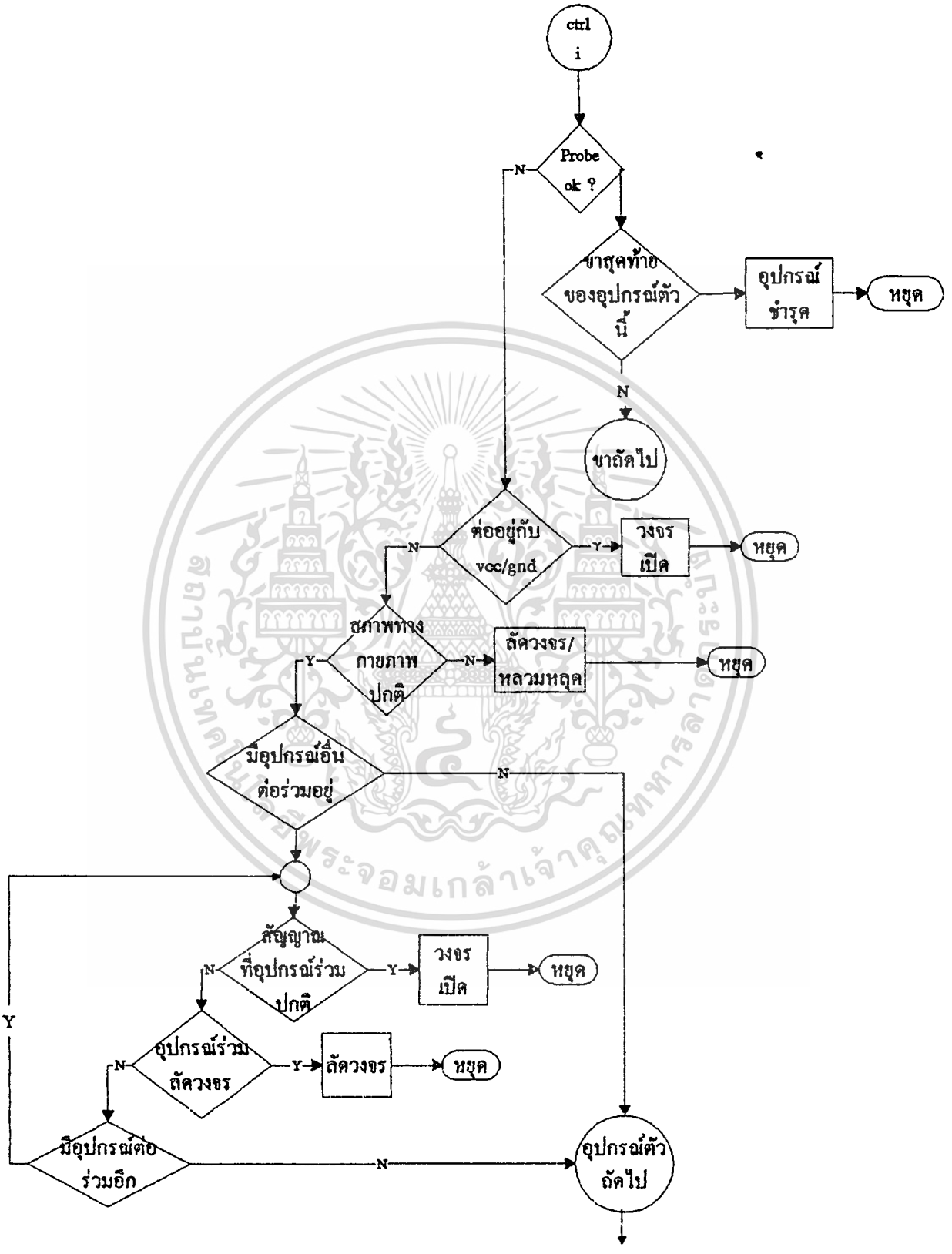


รูปที่ 4.12 ข ผังงาน เมื่อจุดทดสอบต่ออยู่กับ ไฟเลี้ยง หรือ gnd



รูปที่ 4.12 ค ผังงาน เมื่อจุดทดสอบต่ออยู่กับสัญญาณ เอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 ง ผังงานเมื่อจุดทดสอบต่ออยู่กับสัญญาณ ควบคุม หรือสัญญาณ อินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กฎข้อ 60a จะบอกให้ผู้ใช้ทำการตรวจสอบดูว่า มีการลัดวงจรหรือไม่ ถ้าผลการเปรียบเทียบจากกฎ 50 สัญญาณที่จับได้ยังคงไม่เหมือนกับสัญญาณอ้างอิง จากนั้นการทำงานจะโยงไปยังกฎข้อ 70

กฎข้อ 70 จะยกเลิกวงรอบและยุติการอนุมาน ถ้าผลการตรวจสอบของผู้ใช้ปรากฏว่าเกิดการลัดวงจร

หลังจากนั้นจะวนรอบไป กฎข้อ 40 ใหม่ จนกว่าวงรอบการตรวจสอบหาอุปกรณ์อื่นๆที่ต่อร่วมกับจุดนี้จะสิ้นสุดลง แล้วการทำงานก็จะโยง จากกฎข้อ 31b ไปยังกฎข้อ 42a หรือ 42b ซึ่งจะได้อธิบายต่อไปในขั้นตอนที่ 2

3. สัญญาณ ควบคุม และ อินพุต (รูปที่ 4.12 ง)

3.1 การทำงานจะโยงไปยังกฎข้อ 34a ถ้าผู้ใช้ตอบว่าสัญญาณที่วัดได้เหมือนกับสัญญาณอ้างอิง ซึ่งหมายความว่ายังไม่พบสาเหตุ และจะต้องทำการค้นหาต่อไป แล้วการทำงานก็จะโยง ไปยังกฎข้อ 42a หรือ 42b ซึ่งจะได้อธิบายต่อไปในขั้นตอนที่ 2

3.1 การทำงานจะโยงไปยังกฎข้อ 34b ถ้าผู้ใช้ตอบว่าสัญญาณที่วัดได้ผิดพลาดไปจากสัญญาณอ้างอิง และจะต่อไปยังกฎข้อ 41

กฎข้อ 41 จะแสดงข้อความว่าเกิดการหลวมหลุด ถ้าจุดที่กำลังตรวจวัดอยู่ขณะนี้ต่ออยู่กับไฟเลี้ยง หรือ gnd

จากนั้น จะเป็นการแสดงข้อความให้ผู้ใช้ทำการตรวจสอบการหลวมหลุด หรือลัดวงจร ซึ่งถ้าไม่มีการหลวมหลุดหรือลัดวงจร การทำงานลำดับต่อไปจะเหมือนกับการค้นหาจากสัญญาณเอาท์พุต ดังที่ได้กล่าวมาแล้ว

ขั้นตอนที่ 2

การอนุมานในขั้นนี้เป็นการอ่านข้อมูลไปตามระเบียนต่างๆของแฟ้มความรู้ เมื่อในขั้นตอนที่ 1 ยังไม่สามารถสรุปหาข้อขัดข้องได้ โดยการเรียกใช้กฎ 42a หรือจะยุติการอนุมานและสรุปว่าอุปกรณ์ตัวนี้ชำรุด ถ้าการดำเนินงานในขั้นที่ 1 และ 2 ครบทุกขาของอุปกรณ์ส่วนนี้แล้วปรากฏว่าเหมือนกับข้อมูลอ้างอิงทุกประการ แต่ไม่มีสัญญาณเอาท์พุตออกไป โดยเรียกใช้กฎข้อ 42b

4.6.3 ความรู้แบบเมตา (meta knowledge)

จากตัวอย่างในรูปที่ 3.8 ก และ 3.8 ข ข้อมูลในสคมภ์ signame นอกจากจะใช้เก็บชื่อของสัญญาณแล้ว ยังใช้ในการบรรยายการต่อวงจรได้โดยปริยาย ทำให้ทราบถึงลักษณะการจัดต่อถึงกันของอุปกรณ์ต่างๆในวงจร โดยมีข้อสังเกตว่าอุปกรณ์ใดๆ ถ้ามีขาต่อกับสัญญาณชื่อเดียวกัน แสดงว่าขาของอุปกรณ์เหล่านั้นต่อถึงกันหมด ซึ่งเป็นหลักการให้ กลไกการอนุมานใช้ในการติดตามค้นหาข้อขัดข้องในวงจรต่อไป ดังตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นถ้า U1/2 ขา 16 ไม่มีสัญญาณ XTAL- เข้ามา กลไกการอนุมานก็จะทราบได้ว่าสัญญาณ XTAL- ที่ว่านี้ ออกมาจากที่ใด และจ่ายไปให้กับอุปกรณ์ตัวใดบ้าง โดยการสืบเสาะค้นหาไปตามระเบียนต่างๆในสคมภ์ signame ที่มีชื่อสัญญาณว่า XTAL- ซึ่งก็จะพบ R22 ต่อร่วมอยู่ คุณสมบัติการแสดงความรู้โดยปริยายนี้เรียกว่า " ความรู้แบบเมตา "

4.7 การเรียนรู้

ในการสร้างแฟ้มความรู้เพื่อการค้นหาข้อขัดข้องวงจรใดวงจรหนึ่งในครั้งแรก แฟ้มความรู้ก็อาจจะบรรจุ ข้อมูลได้ไม่ครบถ้วนสมบูรณ์ ไม่สามารถค้นหาจุดขัดข้องได้ครอบคลุมทุกประการ หรือบางครั้งก็บรรจุข้อมูลที่ซ้ำซ้อน ทำให้การตรวจวัดค้นหาข้อขัดข้องต้องวนเวียนทำซ้ำอยู่ที่จุดเดิมหลายครั้งหลายหน โดยไม่จำเป็น ดังนั้นในครั้งต่อๆมา จึงต้องมีการปรับปรุง แฟ้มความรู้ให้ครอบคลุม รวดเร็ว และลดความซ้ำซ้อนอยู่เสมอ ยิ่งมีการใช้งานหลายครั้งมากขึ้น แฟ้มความรู้ก็ยิ่งมีความสมบูรณ์มากขึ้น ตัวอย่าง เช่น แฟ้มความรู้ในรูปที่ 3.8 ก ถ้าไม่มีสัญญาณ CLK88 ออกจาก U1/2 ขา 8 กลไกการอนุมานก็จะทำการค้นหาว่า U1/2 นั้นมีสัญญาณออก (output) ที่ขาใดอีกบ้าง ซึ่งก็จะพบได้ที่ขา 2 (PCLK) และเมื่อตรวจสอบขา 2 ถ้ายังไม่มีสัญญาณออกมาอีก กลไกการอนุมานก็จะค้นหาต่อไปจนกว่าจะหมดขาที่ทำหน้าที่ส่งสัญญาณออก ซึ่งก็คือขา 12 (OSC) แต่เมื่อดูในรูปที่ 3.8 ข แล้วจะเห็นว่าไม่มีความจำเป็นที่จะต้องตรวจสอบสัญญาณที่ขา 2 เนื่องจากอุปกรณ์เหล่านั้นถูกบรรจุอยู่ภายในตัว IC 8284 ไม่สามารถถอดเปลี่ยนทดแทนได้ ดังนั้นจึงควรข้ามไปตรวจสอบที่ขา 12 ได้เลย และลบข้อมูลในระเบียนของ U1/2 ขา 2 ออก เป็นการปรับปรุงแฟ้มข้อมูล สิ่งเหล่านี้เราเรียกว่าเป็นการเรียนรู้ ทั้งนี้เพราะเกิดจากการสะสมถ่ายทอดประสบการณ์ความรู้ของผู้ใช้เข้าไปแก้ไขปรับปรุงแฟ้มความรู้ให้ดียิ่งขึ้นเรื่อยๆ นั่นเอง

จากตัวอย่างที่กล่าวมาจะเห็นว่าการเรียนรู้ของ VP-Expert อาศัย กระบวนการในการแก้ไขความรู้ที่มีอยู่เดิม เพื่อการทำงานครั้งต่อไปที่ดีขึ้น ซึ่งเป็นโมเดลการเรียนรู้จากประสบการณ์

ตัวอย่างการเรียนรู้ ที่เห็นได้ชัดเจนอีกตัวอย่างหนึ่งคือ ตัวอย่างที่ 4.1 ในหัวข้อที่ 4.3.5 จะเห็นว่าลำดับก่อน - หลังของข้อมูลมีอิทธิพลต่อทิศทางการค้นหาข้อขัดข้องเป็นอย่างมาก ดังจะเห็นได้จากรูปที่ 4.6 ก และ 4.6 ข ซึ่งนำมาสรุปได้ว่า " ถ้าข้อมูลอยู่ในกลุ่มความสำคัญเดียวกัน ข้อมูลที่อยู่ก่อนจะได้รับการค้นหาก่อน "

จากข้อสรุปดังกล่าวเมื่อนำมาเรียบเรียงใหม่เป็นบทเรียน การเรียนรู้จากประสบการณ์จะกล่าวได้ว่า "สำหรับอุปกรณ์ที่มีขาอินพุตหรือขาควบคุมหลายขา ขาที่สำคัญที่สุดจะต้องอยู่ก่อนขาที่มีความสำคัญรองลงมา" และในทางกลับกัน "สำหรับอุปกรณ์ที่มีขาเอาท์พุตหลายขา ขาที่สำคัญที่สุดจะต้องอยู่ล่างสุดถัดจากขาที่มีความสำคัญรองลงมา" ที่เป็นเช่นนี้ก็เพราะว่าจากการเรียงลำดับของข้อมูลเมื่อขาเอาท์พุตที่สำคัญที่สุดเกิดข้อขัดข้องการตัดสินใจค้นหาข้อขัดข้องก็สามารถกระทำได้ทันทีโดยไม่ต้องตรวจสอบขาเอาท์พุตอื่นๆอีก(เนื่องจากเป็นระเบียนที่อยู่ล่างสุด) ในทำนองเดียวกันถ้าขาเอาท์พุตที่สำคัญรองลงไปเกิดข้อขัดข้อง ก่อนที่จะตัดสินใจค้นหาข้อขัดข้องจะต้องตรวจดูขาเอาท์พุตอื่นๆที่อยู่ในระเบียนที่ต่ำกว่าว่าปกติหรือไม่ ซึ่งการเกิดการทำงานเช่นนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้เพราะมีระเบียน ขาเอาต์พุตอื่นๆที่สำคัญกว่าอยู่ในระเบียนถัดไปนั่นเอง

4.7.1 แนวทางการพัฒนาการเรียนรู้

จากหัวข้อที่ 4.7 จะเห็นได้ว่าการเรียนรู้ของ VP - Expert ได้มาจากการเรียนรู้จากประสบการณ์แล้วนำไปปรับปรุงการทำงานให้ดีขึ้นกว่าเดิม ซึ่งส่วนใหญ่ของการดึงความรู้ เกิดจากการสังเกตและทดลองของผู้เชี่ยวชาญ ซึ่งการกระทำดังกล่าวยังไม่เป็นอัตโนมัติ ต้องอาศัยมนุษย์เป็นผู้ทำการปรับปรุงแฟ้มความรู้ ในงานวิจัยนี้ได้วางแนวทางการพัฒนาการเรียนรู้ไว้ตามลักษณะการเรียนรู้โดยเครื่องจักรในหัวข้อที่ 2.5 ดังตัวอย่างต่อไปนี้

ตัวอย่างที่ 4.4

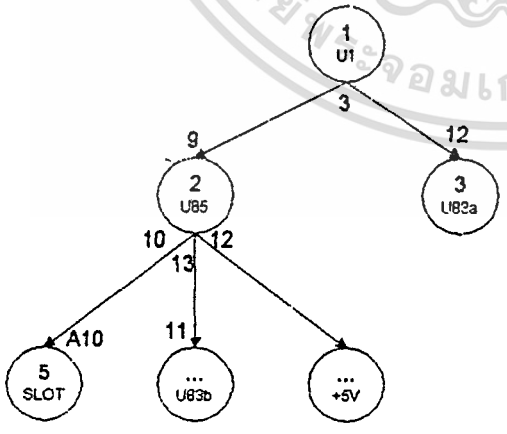
จากวงจรและแฟ้มข้อมูลในตัวอย่างที่ 4.2 ซึ่งเมื่อปรับปรุงแล้วนำมาเขียนเป็นโครงสร้างต้นไม้ ก็จะได้ดังรูปที่ 4.13 ก ในการพัฒนาการเรียนรู้ ดำเนินการดังนี้

1. ทำการบันทึกเส้นทางการค้นหาข้อขัดข้องไว้ในอะเรย์ 1 มิติ t[x] ตามลำดับก่อนหลัง

2. คัดลอกข้อมูลดังต่อไปนี้คือ ชื่ออุปกรณ์, ขา และลำดับของจุดต่างๆที่ทำการค้นหาข้อขัดข้อง t[x] แล้วเก็บเข้าแฟ้มความรู้ซึ่งมีโครงสร้าง ดังรูปที่ 4.13 ข

3. นำแฟ้มความรู้จากข้อ 2 มาใช้งานโดย

3.1 นำแต่ละระเบียนของแฟ้มความรู้จากข้อ 2 มาเขียนเป็นโครงสร้างต้นไม้ โดยมีโหนดต่างๆ เรียงตามลำดับข้อมูลในอะเรย์ t[x] และพิจารณาร่วมกับแฟ้มความรู้ของวงจรนั้นๆ ซึ่งจะได้เป็นต้นไม้เส้นทางการค้นหาข้อขัดข้อง เพื่อนำไปวิเคราะห์ว่าเหมาะสมหรือไม่ ต้องปรับปรุงแฟ้มข้อมูลอย่างไร เป็นต้น



รูปที่ 4.13 ก โครงสร้างต้นไม้เพื่อปรับปรุงการเรียนรู้ ซึ่งตัวเลขภายในวงกลม หมายถึงลำดับ t[x]

NODE	PIN	t[x]
U3	3	1
U65	9	2
U83a	12	3
U65	10	4
SLOT	A10	5

รูปที่ 4.13 ข แฟ้มความรู้เพื่อปรับปรุงการเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

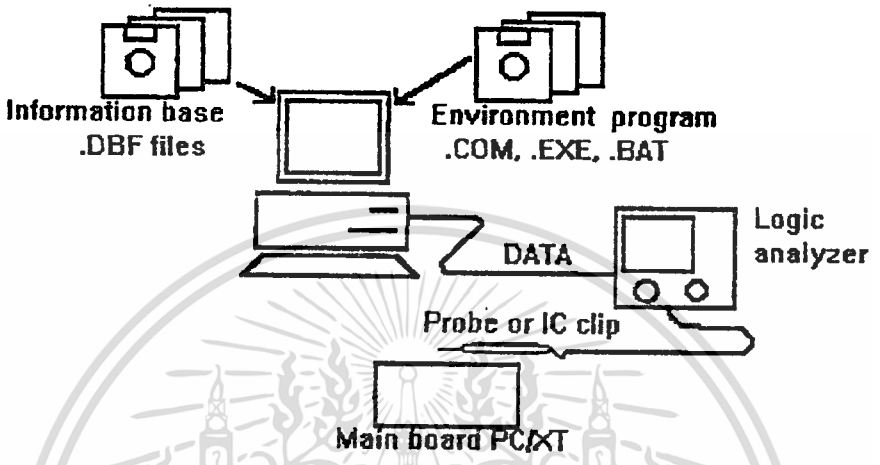
3.2 เขียนโปรแกรมสนับสนุนการทำงาน โดยให้โปรแกรมดังกล่าวมีความสามารถในการปรับหรือเคลื่อนย้ายข้อมูลในระเบียนต่างๆของแฟ้มความรู้ของวงจรนั้นได้ ซึ่งการปรับหรือเคลื่อนย้ายข้อมูลเหล่านี้เป็นไปตามสถิติ เส้นทางการค้นหาข้อขัดข้องจากข้อ 3.1

<u>หมายเหตุ</u> แนวทางการพัฒนาในข้อที่ 1 ได้ดำเนินการไปแล้วซึ่งสามารถศึกษารายละเอียดได้จากแฟ้มความรู้ CHIP.KBS. ในภาคผนวกที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทำงานของโปรแกรมและตัวอย่างการทดลอง



รูปที่ 5.1 ส่วนประกอบของระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลัก

ส่วนประกอบของระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้อง บนแผงวงจรหลักไมโครคอมพิวเตอร์ ประกอบด้วย 2 ส่วนหลักคือ ส่วนที่หนึ่งไมโครคอมพิวเตอร์และโปรแกรมผู้เชี่ยวชาญ VP-Expert ทำหน้าที่ในการวินิจฉัยข้อขัดข้อง และส่วนที่สองเครื่องวิเคราะห์โลจิก ทำหน้าที่ในการจับและแสดงข้อมูล ดังรูปที่ 5.1 สำหรับการเชื่อมโยงระหว่างทั้ง 2 ส่วนนั้น อาศัยผู้ใช้เป็นตัวกลางดังที่ได้กล่าวมาแล้วในบทที่ 1

รายละเอียดการสร้างแฟ้มความรู้และการเรียกโปรแกรมขึ้นมาใช้งาน ดูได้จากภาคผนวกที่ 1 (คู่มือการใช้งาน) ต่อไปนี้จะกล่าวถึงขั้นตอนการทำงานของโปรแกรมเมื่อถูกเรียกขึ้นมาแล้ว และตัวอย่างการทดสอบขั้นที่ 1 ถึง 7 สำหรับการแฟ้มความรู้ของการทดสอบขั้นต่างๆที่ได้ทำการทดลองมาแล้ว ดูได้จากภาคผนวก

5.1 ขั้นตอนการทำงานและการค้นหาของโปรแกรมที่ทดลอง

การทำงานและการค้นหาของโปรแกรมที่ทดลองประกอบด้วยส่วนหลักๆ 2 ส่วนคือ

1. ส่วนเตรียมการ
2. ส่วนการอนุมาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.ส่วนเตรียมการ

ดังได้กล่าวมาแล้วในหัวข้อที่ 2.4.2 เรื่อง การแบ่งระดับของการค้นหาข้อขัดข้อง ว่าได้แบ่งระดับของการค้นหา จากระดับใหญ่ไปสู่ระดับย่อย เช่นจาก ระบบ - แผงวงจรใหญ่ - แผงวงจรย่อย - ตัวอุปกรณ์ เป็นต้น

ดังนั้นในส่วนเตรียมการนี้ หน้าที่ประการแรกคือ นำข้อมูลจากแฟ้มความรู้ระดับต่างๆเข้ามาไว้ในพื้นที่หน่วยความจำใช้งาน โดยจะทำการโหลดแฟ้มความรู้ของการทดสอบระดับใหญ่ เช่น ระบบ หรือแผงวงจรหลักเข้ามาก่อน ซึ่งภายในแฟ้มความรู้ระดับ แผงวงจรหลักเหล่านี้จะบรรจุรายชื่อแฟ้มความรู้ของ แผงวงจรย่อย หรือการ์ดต่างๆ ในระดับย่อยลงไป และแฟ้มความรู้ของการ์ดเหล่านี้ ก็อาจจะบรรจุรายชื่อแฟ้มความรู้ของ อุปกรณ์หรือการทดสอบในระดับย่อยเล็กลงไปได้อีก แต่ในการวิจัยครั้งนี้เป็นการทดลองบนแผงวงจรหลัก ไมโครคอมพิวเตอร์ PC/XT ดังนั้นจึงแบ่งเป็นเพียง 2 ระดับเท่านั้น คือ ระดับวงจรใหญ่และระดับวงจรย่อย โดยมีการทำงานดังนี้

1. กำหนดโครงสร้างแฟ้มข้อมูลของ ระดับวงจรใหญ่ (ใช้ชื่อ START.DBF) ดังรูปที่ 5.2 ซึ่งก็คือแฟ้มความรู้ของแผงวงจรหลักไมโครคอมพิวเตอร์ PC/XT นั่นเอง ภายในแฟ้มนี้ข้อมูลระเบียนหนึ่งๆจะเป็นข้อมูลรายชื่อของ วงจรย่อยในบริเวณใดบริเวณหนึ่งของแผงวงจรหลัก

Card	Descript

รูปที่ 5.2 โครงสร้างฐานความรู้ START.DBF

ก. Card เป็น สคมภ์ ที่ใช้เก็บชื่อของวงจรย่อยต่างๆ ซึ่งชื่อของวงจรย่อยเหล่านี้เปรียบเสมือน pointer ชี้ไปยังแฟ้มข้อมูลจริงๆอีกทีหนึ่ง

ข. Descript เป็น สคมภ์ ที่ใช้เก็บคำบรรยาย หรือ รายละเอียดเกี่ยวกับวงจรย่อยนั้นๆ

2. ใช้ VP-Expert อ่านรายชื่อแฟ้มความรู้ของวงจรย่อยต่างๆจากฐานความรู้ แล้วแสดงรายชื่อวงจรย่อยเหล่านี้ เพื่อให้ผู้ใช้เลือกที่จะทดสอบ

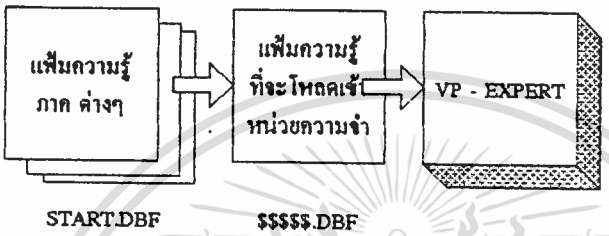
3. นำชื่อวงจรย่อย ที่ผู้ใช้เลือกจากข้อ 2 มา ทำการอนุมาน

เนื่องจากการแสดงความรู้แบบอิงกฎของ VP-Expert สามารถเรียกใช้แฟ้มข้อมูลจากภายนอกได้เพียงคราวละ 1 แฟ้มความรู้เท่านั้น ดังนั้นจึงกำหนดเปลือกแฟ้มความรู้ที่มีชื่อว่า $$$$$.dbf ขึ้นมาก่อน (ซึ่งเป็นแฟ้มความรู้เดียวที่กลไกการอนุมานนี้รู้จักและเรียกใช้ได้) แฟ้มความรู้นี้มีแต่เพียงชื่อเท่านั้น ไม่มีข้อมูลจริงๆ และเมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะใช้แฟ้มข้อมูลใด ก็เพียงแต่เปลี่ยนชื่อแฟ้มข้อมูลนั้นเป็น $$$$$.dbf แล้วทำการค้นหาข้อขัดข้องไปบนแฟ้มนี้ ดังนั้นแฟ้มความรู้ของวงจรย่อยต่างๆ ก่อนที่จะทำการอนุมาน ค้นหาสาเหตุข้อขัดข้องจะต้องดำเนินการดังนี้

- เปลี่ยนเป็นชื่อ $$$$$.dbf ก่อน
- ทำการโหลดข้อมูลจาก $$$$$.dbf ลงพื้นที่หน่วยความจำที่ใช้งาน ดังรูปที่ 5.3
- เปลี่ยนชื่อจาก $$$$$.dbf กลับไปเป็นชื่อเดิม
- ดำเนินการอนุมานค้นหาสาเหตุข้อขัดข้องกับข้อมูลที่อยู่ในหน่วยความจำต่อไป



รูปที่ 5.3 การเลือกใช้แฟ้มความรู้

เพื่อให้การเตรียมการเป็นไปตามวัตถุประสงค์จึงต้องเขียนโปรแกรมเพิ่มเติมดังนี้

ก. TRANSF.EXE เป็นโปรแกรมที่เขียนด้วยภาษาซีและแอสเซมบลี้ ทำหน้าที่เปลี่ยนชื่อแฟ้มข้อมูลของวงจรย่อยต่างๆ ให้เป็นชื่อ $$$$$.dbf

ข. SINK.EXE เป็นโปรแกรมที่เขียนด้วยภาษาซีและแอสเซมบลี้ ทำหน้าที่เปลี่ยนชื่อแฟ้ม $$$$$ .dbf ให้เป็นชื่อเดิม

รายละเอียดของโปรแกรม TRANSF.C และ SINK.C อยู่ในภาคผนวกที่ 11 และ 12 ตามลำดับ.

ขั้นตอนของการเตรียมการดังกล่าวเขียนเป็นกิจกรรมของแฟ้มความรู้ VP-Expert ได้ดังนี้

```
RUNTIME;
ACTIONS
MENU example,ALL,start,card     ! แสดงรายชื่อวงจรย่อย ให้ผู้ใช้เลือก
FIND example
WHILEKNOWN card
   GET ALL,start,card
END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
dbase = (example)              ! ชื่อวงจรย่อยที่ผู้ใช้เลือก ถูกเก็บอยู่ในตัวแปรชื่อ dbase
SHIP @@@@@,dbase               ! เก็บค่าที่อยู่ในตัวแปร dbase ถูกเก็บไว้ในแฟ้มชื่อ @@@@@
CALL transf                    ! เปลี่ยนชื่อแฟ้มที่มีชื่อตรงกับข้อมูลในแฟ้ม @@@@@ -
                               ! เป็น $$$$$.dbf
!---------- Load database file ----------
x = 1
DISPLAY "Loading knowledge base."
WHILEKNOWN node
   GET ALL,$$$$$.ALL           ! โหลดข้อมูลในแฟ้ม $$$$$.dbf ลงมาไว้ในหน่วยความจำใช้งาน
   xnode[x] = (node)
   xpin[x] = (pin)
   funct[x] = (func)
   tdiagm[x] = (t_diagm)
   signal[x] = (signame)
   t[x] = 0
   x = (x+1)
END
bottom = (RECORD_NUM)
CLOSE $$$$$
CALL sink                      ! เปลี่ยนชื่อแฟ้ม $$$$$.dbf เป็นชื่อเดิม
CLS
```

หน้าที่ประการที่สองของส่วนเตรียมการคือทำการกำหนดค่าเริ่มต้นต่างๆให้กับแฟ้มการแสดงความรู้ (rules base) โดย

1. กำหนดว่า การค้นหาจะต้องเริ่มต้นจากระเบียนแรกเสมอ

2. ภายใต้การวนลูปเพื่อการอนุมานค้นหาสาเหตุข้อขัดข้อง กำหนดว่ากฎที่ถูกเรียกมาใช้งานมีเพียง กฎที่ทำหน้าที่แสดงข้อมูลอ้างอิง (rules 20x) และ กฎที่คอยตรวจสอบการสิ้นสุดของข้อมูลในแฟ้ม (rule ff) ซึ่งแสดงถึงว่า มีข้อมูลความรู้บรรจุอยู่ในแฟ้มพอเพียงหรือไม่ เท่านั้น

สำหรับรายละเอียดวงรอบลูป การอนุมาน จะเป็นเรื่องของการโยงไปยังกฎต่างๆดังรูปที่ 4.8 ตามลักษณะของวงจร และคำตอบ ต่างๆของผู้ใช้ ซึ่งเขียนเป็นกิจกรรมของแฟ้มความรู้ (rules base)ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
!---------- Initialize ----------
x = 1                    ! กำหนดค่าตัวชี้ แรกเริ่มให้ชี้ไปที่ระเบียนที่ 1
tx = 0                   ! กำหนดค่าตัวนับ แรกเริ่มลำดับของจุดที่ทำการทดสอบ = 0
inloop = go              ! กำหนดค่า แฟลกในการลูป
search = (signal[1])     ! ตัวแปร ตัวแรก = ชื่อสัญญาณจากระเบียนที่ 1
pin_func = (funct[1])    ! ตัวแปร ตัวที่สอง = ฟังก์ชั่นขาอุปกรณ์จากระเบียนที่ 1

!-------------------------------------------------
WHILEKNOWN inloop
   y = (x+1)             ! กำหนดให้ ตัวชี้ตัวที่สอง ชี้นำหน้าตัวที่หนึ่งอยู่เสมอ
   RESET cond
   RESET testacq
   RESET disp
   RESET scannode        ! ป้องกันข้อผิดพลาดอันเนื่องมาจากการตรวจสอบโหนดร่วม
   FIND disp             ! โยงไปยังกฎที่ทำหน้าที่แสดงข้อมูลอ้างอิงหรือข้อมูลข่าวสาร
   x = (x+1)             ! เพิ่มค่าตัวชี้
   RESET full
   FIND full             ! โยงไปยังกฎที่คอยตรวจสอบการสิ้นสุดข้อมูลในแฟ้ม
END;
```

2.ส่วนอนุมาน

วงรอบการทำงานของการอนุมานค้นหาสาเหตุข้อขัดข้อง เริ่มต้นจากคำถามที่ว่า " ทำไมจุดที่ทำการทดสอบจุดนี้ จึงมีสัญญาณผิดปกติไป หรือทำไมจึงไม่มีสัญญาณ " และตั้งคำถามต่อไปว่าจุดที่ทำการทดสอบหรือสังเกตอยู่นี้ได้รับสัญญาณเจากที่ใด และจุดที่ว่านั้นมีสัญญาณออกมาปกติหรือไม่ แล้วจึงทำการทดสอบสมมติฐานโดยการโพรบ และทำการอนุมานค้นหาคำตอบต่อไปดังกล่าวมาแล้ว

สำหรับภาพรวมของการอนุมานค้นหาสาเหตุข้อขัดข้อง เหตุผลในการโยงไปยังกฎต่างๆ ดังได้กล่าวมาแล้วในหัวข้อที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ตัวอย่างการทดสอบ

การทดสอบเบื้องต้นก่อนการค้นหาข้อขัดข้องบนแผงวงจรหลัก

ในการทำงานจะให้ผู้ใช้ทำการทดสอบ ขั้นแผงวงจรย่อยหรือการ์ดก่อน เพื่อแยกภาคที่เสียว่า อยู่บนการ์ด หรืออยู่บนแผงวงจรหลัก ตามผังงานในรูปที่ 5.4 ซึ่งเมื่อปรากฏเป็นที่แน่นอนว่า สาเหตุข้อขัดข้องเกิดจากแผงวงจรหลัก จึงจะดำเนินการค้นหาสาเหตุข้อขัดข้องในระดับอุปกรณ์บนแผงวงจรหลักต่อไป ภายใต้เงื่อนไขการกำหนดคุณลักษณะของข้อขัดข้องในหัวข้อที่ 3.2.1 ซึ่งได้กำหนดไว้ว่า เป็นข้อขัดข้องเชิงโลจิกที่เกิดจากอุปกรณ์บนแผงวงจรหลัก และเป็นข้อขัดข้องแบบถาวรที่มีผลกระทบต่อฟังชั่นการทำงานของแผงวงจรหลักเท่านั้น ตัวอย่าง เช่น สมมุติว่ามีลายวงจรที่ต่ออยู่กับช่องเสียบการ์ด (SLOT) ขาด ทำให้เกิดวงจรเปิด แม้ว่าข้อขัดข้องนี้จะมีผลทำให้การ์ดอื่นๆที่เสียบอยู่ทำงานไม่ปกติก็ตาม แต่ถ้าไม่ได้ส่งผลกระทบต่อการทำงานของแผงวงจรหลักเลย กล่าวคือแผงวงจรหลักยังคงทำงานได้ตามปกติ ก็ไม่ถือว่าเป็นข้อขัดข้องในงานวิจัยนี้ หรือในบางกรณีย์ เช่น ถ้าขาของอุปกรณ์ตัวหนึ่ง ซึ่งเป็นขาที่ทำหน้าที่ในการควบคุมการทำงานของตัวอุปกรณ์ และปกติขานี้ต่ออยู่กับไฟเลี้ยง ถ้าสมมุติว่าขาของตัวอุปกรณ์ขานี้หัก หรือ หลุดออกจากลายวงจร และผลของข้อขัดข้องไม่ได้ทำให้การทำงานของแผงวงจรหลักหยุดชะงัก หรือทำงานผิดพลาดแต่อย่างใด กรณีย์นี้ก็ไม่ถือว่าเป็นข้อขัดข้องในงานวิจัยนี้เช่นเดียวกัน แต่ถ้าผลของข้อขัดข้องนี้ทำให้การทำงานของแผงวงจรหลักหยุดชะงักลง เช่น ไม่สามารถทำการทดสอบตัวเองได้(Power On Self Test)จึงจะถือว่าเป็นข้อขัดข้องในงานวิจัยนี้เป็นต้น (สำหรับข้อกำหนดอื่นๆจะยึดถือตามลักษณะของปัญหาในงานวิจัย ในหัวข้อที่ 3.2 ดังที่ได้กล่าวมาแล้วในบทที่ 3)

เมื่อมีข้อขัดข้องที่เป็นไปตามคุณลักษณะที่กำหนดเกิดขึ้น การค้นหาก็จะทำการติดตามสัญญาณที่ผิดปกติ ซึ่งเกิดจากผลของข้อขัดข้องดังกล่าวไปตามอุปกรณ์ต่างๆที่เกี่ยวข้องในวงจร ย้อนหลังจากอุปกรณ์ตัวหนึ่งไปยังอุปกรณ์อีกตัวหนึ่งเรื่อยไปจนกว่าจะพบสาเหตุข้อขัดข้อง หรือสิ้นสุดเส้นทางการค้นหา ดังที่ได้กล่าวมาแล้วในหัวข้อที่ 3.2.4 ซึ่งจะแสดงตัวอย่างให้ทราบต่อไปในหน้าที่ 101

สำหรับอาการข้อขัดข้องที่เกิดจากแผงวงจรหลัก, สาเหตุต่างๆ, ลำดับการค้นหาและเส้นทางการค้นหาจะได้แสดงให้ทราบต่อไปตามลำดับ

การค้นหาข้อขัดข้องบนแผงวงจรหลัก

เริ่มต้นด้วยการทดสอบว่า CPU 8088 ทำงานหรือไม่ โดยการตรวจสอบที่สัญญาณ RESET, CLK88 และ READY ถ้าสัญญาณเหล่านี้ปกติ ก็แสดงว่า CPU พร้อมที่จะทำงาน การทดสอบขั้นต่อไปจึงเป็นการทดสอบว่า CPU สามารถควบคุม บัสแอดเดรสและบัสข้อมูล ได้หรือไม่ โดยทำการตรวจสอบญญาณAENBRD ถ้าสัญญาณนี้ปกติก็แสดงว่า CPU สามารถควบคุมบัสทั้งสองได้ ซึ่งก็หมายความว่า CPU จะสามารถอ่านโปรแกรม BIOS ได้ และการทำงานก็จะเป็นไปตามโปรแกรม BIOS แต่ถ้าสัญญาณ AENBRD ไม่ปกติ ข้อขัดข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็สามารถที่จะเกิดขึ้นได้ ตั้งแต่ขั้นตอนการอ่าน EPROM BIOS, วงจร TIMER และ DMA ตามลำดับ ดังผังงานในรูปที่ 5.5 ซึ่งบล็อกสี่เหลี่ยมหมายถึงชื่อแฟ้มข้อมูล ซึ่งใช้ทำการตรวจสอบวงจรที่เกี่ยวข้องดังกล่าวมาแล้ว และจากผังงานดังกล่าวนำมาเขียนเป็นตารางลำดับการค้นหาข้อขัดข้องแผงวงจรหลักได้ดังตารางที่ 5.1

อนึ่งในระบบผู้เชี่ยวชาญที่ทำการวิจัยนี้ ตัวโปรแกรมการอนุมานจะแสดงคำแนะนำ ในการเรียกใช้แฟ้มข้อมูลที่สอดคล้องกับการวินิจฉัยให้เองโดยผู้ใช้เพียงแต่เริ่มต้นจากแฟ้มความรู้ RUNCPU.DBF เท่านั้น

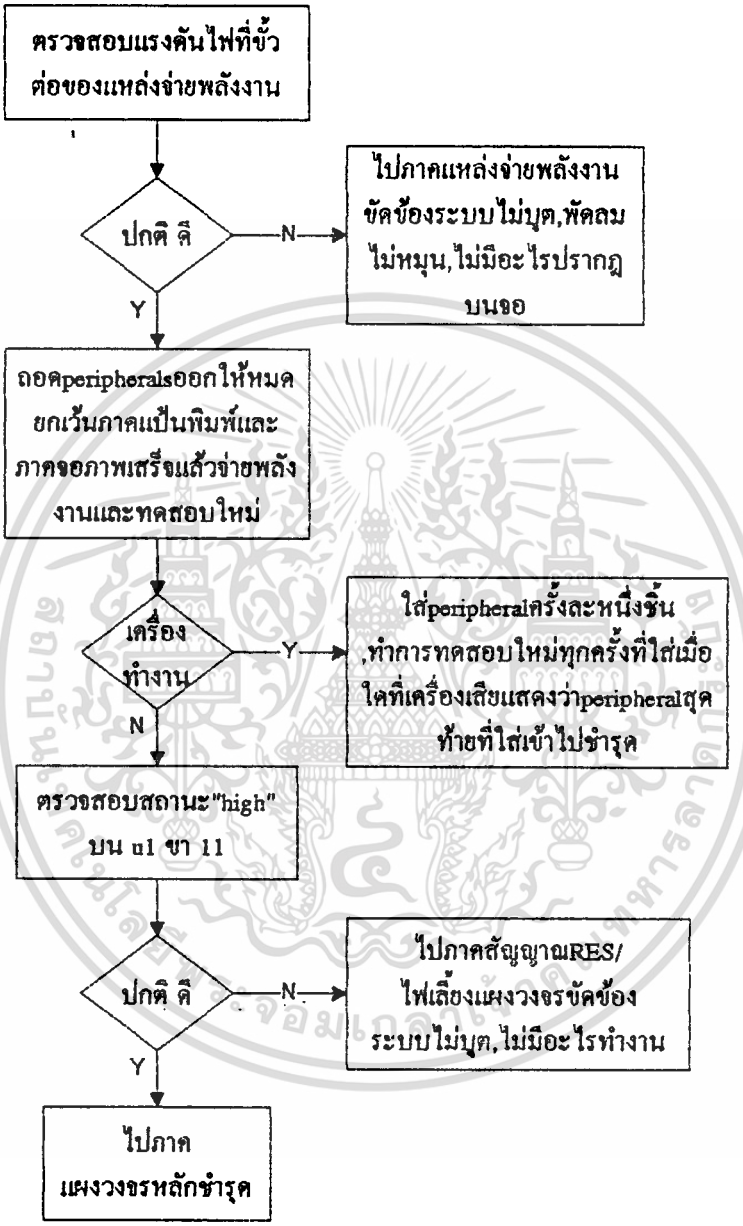
สำหรับสรุปผลการค้นหาข้อขัดข้องของการทดสอบแต่ละขั้น จะสรุปให้ทราบดังนี้

1. สาเหตุที่ทำการค้นหาทั้งสิ้น หมายถึง จำนวนข้อขัดข้องที่มีลักษณะตามหัวข้อที่ 3.2 ที่สามารถเกิดขึ้นได้ทั้งหมด
2. สามารถตรวจพบได้ขณะนี้ หมายถึง จำนวนข้อขัดข้องที่การทดสอบแต่ละขั้นสามารถตรวจพบได้
3. สามารถตรวจพบได้ข้างหน้า หมายถึง ข้อขัดข้องบางสาเหตุที่การทดสอบในขณะนั้นไม่สามารถตรวจพบได้ ต้องทำการทดสอบต่อไปจึงจะตรวจพบ
4. ไม่สามารถตรวจพบได้ หมายถึง ข้อขัดข้องซึ่งข้อมูลจากแฟ้มความรู้ทั้ง 7 แฟ้มนี้ ยังไม่สามารถตรวจพบ หรือ ยังไม่สามารถค้นหาได้ ซึ่งข้อขัดข้องเหล่านี้อาจจะสามารถตรวจพบหรือค้นหาได้ถ้าได้ดำเนินการสร้างแฟ้มความรู้ต่อไป

การทดสอบที่	ทำการค้นหาข้อขัดข้องจาก	จุดมุ่งหมาย	ชื่อแฟ้มความรู้
1, 2, 3	สัญญาณ RESET, CLK88, READY	เป็นการทดสอบว่า CPU สามารถทำงานได้หรือไม่	RUNCPU
4	บนสัญญาณ address A11-A0	เป็นการทดสอบ address bus A11 - A0 ขาด/ลัด วงจรหรือไม่	ADDRBUS1
5	การอ่าน EPROM BIOS	เป็นการทดสอบว่า CPU สามารถอ่านโปรแกรมจาก EPROM BIOS ได้หรือไม่	READBIOS
6	สัญญาณ HRQDMA, AENBRD	เป็นการทดสอบว่า DMA, TIMER ได้รับโปรแกรมจาก CPU หรือไม่	ADDRBUS2
7	สัญญาณ HRQDMA, OUT1	เป็นการทดสอบว่า DMA, TIMER ทำงานตามโปรแกรมหรือไม่	DMATIMER

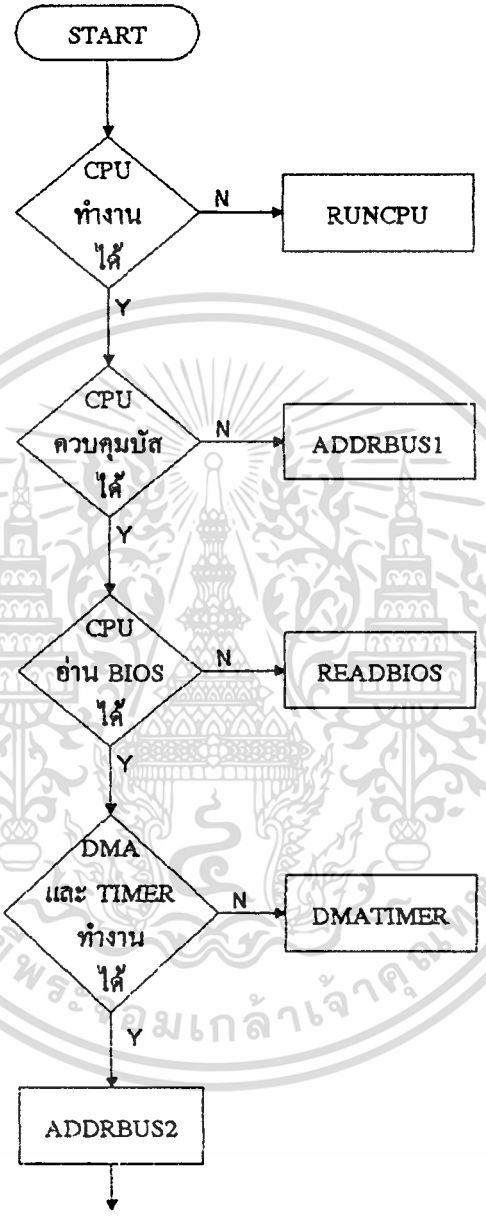
ตารางที่ 5.1 ลำดับการทดสอบแผงวงจรหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 การทดสอบเบื้องต้น ก่อนการค้นหาข้อขัดข้องบนแผงวงจรหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.5 ผังงานลำดับการทดสอบแผงวงจรหลัก

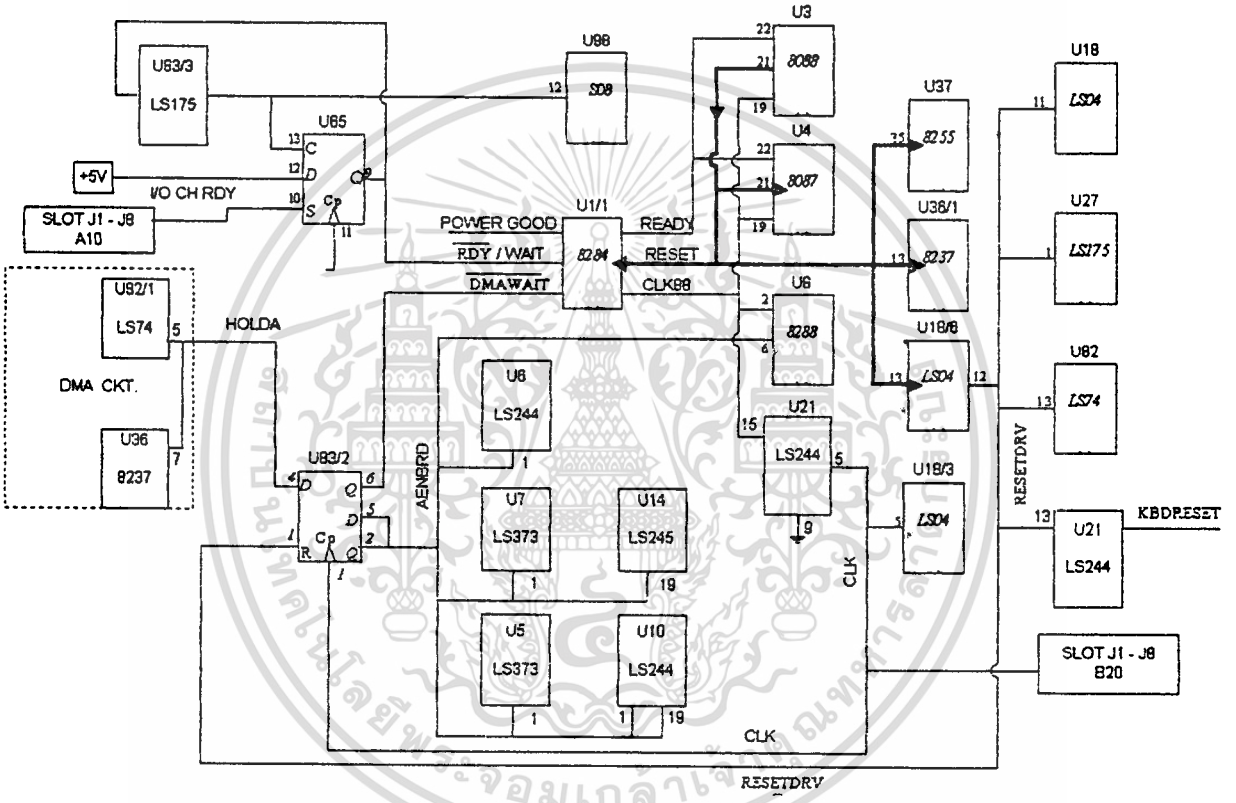
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบที่ 1

การค้นหาข้อขัดข้องเมื่อสัญญาณ RESET ของ U3(8088)ไม่ปกติ

การวิเคราะห์เส้นทางการค้นหา

รูปที่ 5.6 แสดงเส้นทางการค้นหาข้อขัดข้อง ซึ่งเมื่อพิจารณาแล้วจะเห็นว่าอยู่ในรูปโครงสร้างต้นไม้ แล้ว จึงสามารถนำสัญญาณอ้างอิงไปเก็บไว้ในแฟ้มความรู้ได้ดังแสดงอยู่ในแฟ้มความรู้ชื่อ RUNCPU.DBF ในภาคผนวกที่ 4



รูปที่ 5.6. เส้นทางการค้นหาข้อขัดข้องจากสัญญาณ RESET ของ U3(8088)

การค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้จับสัญญาณแบบ Asynchronous, ความถี่ของสัญญาณประมาณ 5 MHz (200 nsec)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⌐‾_) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณตามจุดต่างๆ

ถ้า Logic Analyzer สามารถจับสัญญาณได้ ให้ข้ามไปดำเนินการในข้อ 3.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า Logic Analyzer ไม่สามารถจับสัญญาณได้ ให้ใช้สัญญาณที่ U1 ขา 11(POWER GOOD หรือ RES)ช่วงเปลี่ยนสถานะจาก 'L' เป็น 'H' (⎍⎍) เป็นตัวกระ ตุ้น (trigger) ให้ทำการจับสัญญาณตามจุดต่างๆ แล้ว

3. ทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ 5.6.

ผลการค้นหาข้อขัดข้อง ดังแสดงอยู่ในตารางผลการค้นหาข้อขัดข้อง

ตาราง ผลการค้นหาข้อขัดข้องจากสัญญาณ RESET

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
U3	21	short	43b	
		open	43b	
U1/1	10	short	43a	
		open	43a	
	11	short	41	
		open	41	
U4	21	short	70	open เครื่องสามารถทำงานได้
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
U37	35	short	70	open สัญญาณ RESET ไม่ผิดปกติ
		open	ตรวจไม่พบ	
U36	13	short	70	open เครื่องสามารถทำงานขั้นต้นได้
		open	ตรวจไม่พบ	แต่แสดงรหัสผิดพลาด "101"
U18/6	13	short	70	
		open	ตรวจพบได้จากการทดลองที่ 3	

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	13	สาเหตุ
สามารถตรวจพบได้ขณะนี้	10	สาเหตุ
สามารถตรวจพบได้ข้างหน้า	1	สาเหตุ
ไม่สามารถตรวจพบได้	2	สาเหตุ

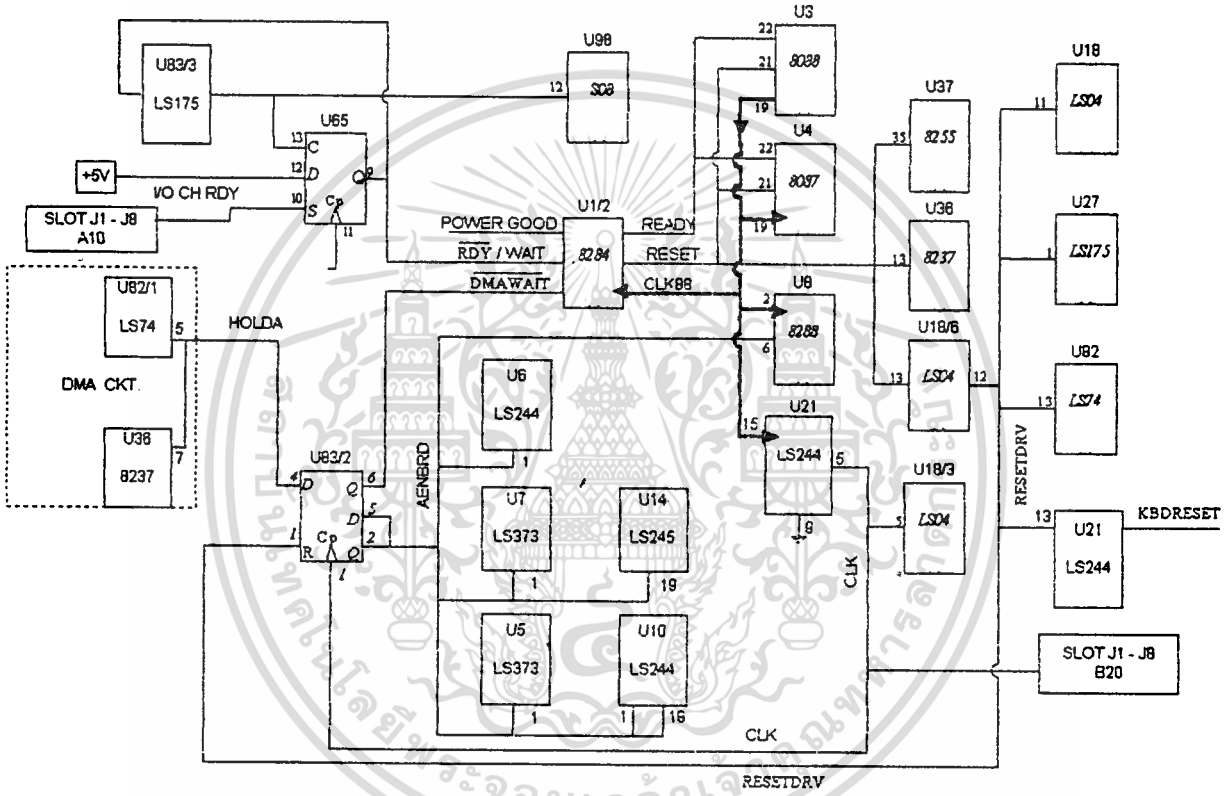
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 2

การค้นหาข้อขัดข้องเมื่อสัญญาณ CLK88 ของ U3(8088)ไม่ปกติ

การวิเคราะห์เส้นทางการค้นหา

รูปที่ 5.7 แสดงเส้นทางการค้นหาข้อขัดข้อง ซึ่งเมื่อพิจารณาแล้วจะเห็นว่าอยู่ในรูปโครงสร้างต้นไม้ แล้ว จึงสามารถนำสัญญาณอ้างอิงไปเก็บไว้ในแฟ้มความรู้ได้ ดังแสดงอยู่ในแฟ้มความรู้ชื่อ RUNCPU.DBF ในภาคผนวกที่ 4



รูปที่ 5.7 เส้นทางการค้นหาข้อขัดข้องจากสัญญาณ CLK88 ของ U3(8088)

การค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้ Synchronous กับ สัญญาณ CLK88 (U1 ขา 8)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⌐‾‾↓__) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณตามจุดต่างๆ

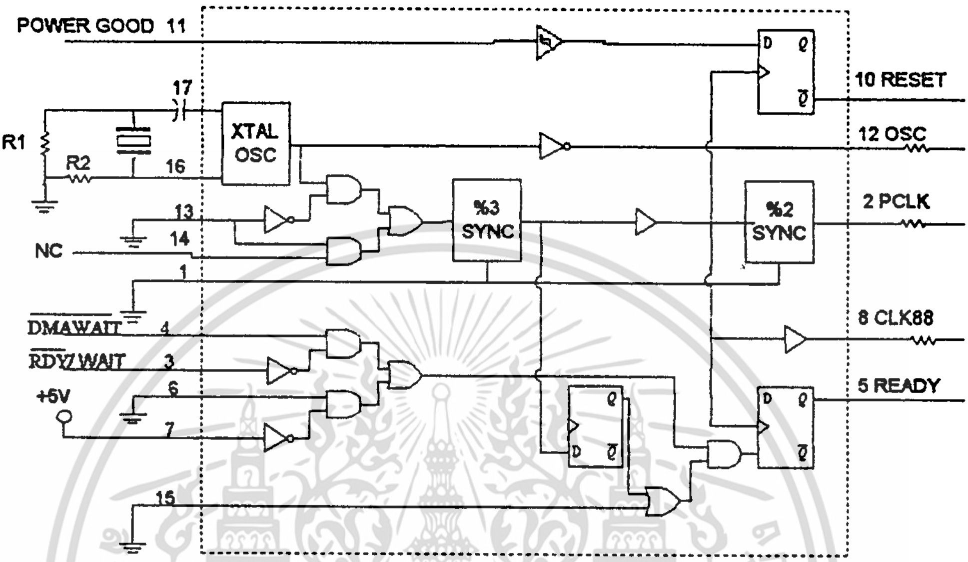
ถ้า Logic Analyzer สามารถจับสัญญาณได้ ให้ข้ามไปดำเนินการในข้อ 3.

ถ้า Logic Analyzer ไม่สามารถจับสัญญาณได้ให้ ตั้งค่า(set) Logic Analyzer ให้จับสัญญาณแบบ Asynchronous, ความถี่ของสัญญาณประมาณ 5 MHz (200 nsec) แล้วใช้สัญญาณที่ U1 ขา 10 (RESET)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' ( ) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณตามจุดต่างๆ แล้ว

3. ทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ 5.7



รูป แสดงส่วนประกอบภายใน U1 (8284)

ประกอบเส้นทางการค้นหาข้อขัดข้องบนสัญญาณ RESET, CLK88, และ READY

ผลการค้นหาข้อขัดข้อง ดังแสดงอยู่ในตารางผลการค้นหาข้อขัดข้อง

ตาราง ผลการค้นหาข้อขัดข้องจากสัญญาณ CLK88

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
U3	19	short	43bหรือ60b	
		open	43bหรือ60b	
U4	19	short	70	open เครื่องสามารถทำงานได้
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
U21	15	short	70	open สัญญาณ CLK88 ไม่ผิดปกติ
		open	ตรวจพบได้จากการทดสอบที่ 3	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
U8	2	short	-70-	open สัญญาณ CLK88 ไม่ผิดปกติ
		open	ตรวจพบได้จากการทดสอบที่ 5	
U1/2	8	open	43a	
		short	70	
	1	open	41	ต่ออยู่กับ GND
		short	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	เมื่อลัดวงจรจึงไม่มีผลต่อการทำงาน
	2	short		
		open	ตรวจพบได้จากการทดสอบที่ 6	
	12	short		open เครื่องสามารถทำงานได้
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
	16	short	43b	
		open	43b	
	17	short	43b	
		open	43b	
R22		short	43b	
		open	43b	
R23		short	43b	
		open	43b	

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	21	สาเหตุ
สามารถตรวจพบได้ขณะนี้	18	สาเหตุ
สามารถตรวจพบได้ข้างหน้า	3	สาเหตุ
ไม่สามารถตรวจพบได้	-	สาเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

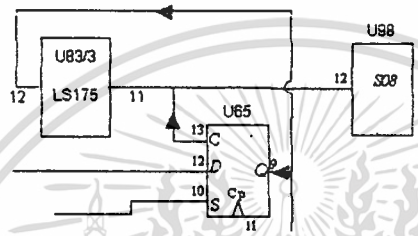
การทดสอบที่ 3

การค้นหาข้อขัดข้องเมื่อสัญญาณ READY ไม่ปกติ

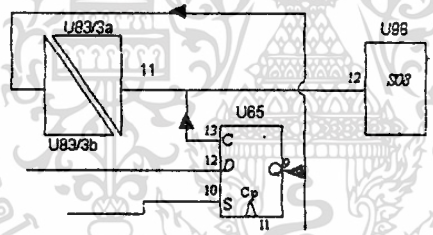
การวิเคราะห์เส้นทางการค้นหา

ในการทดสอบก่อนอื่นจะทำการเฝ้าสังเกตสัญญาณ READY ก่อน เมื่อสัญญาณนี้ผิดปกติจึงจะทำการติดตามค้นหาสาเหตุไปตามเส้นทางการค้นหาหรือสัญญาณอื่นๆที่เกี่ยวข้องต่อไป

รูปที่ 5.9 แสดงเส้นทางการค้นหาข้อขัดข้อง ซึ่งเมื่อพิจารณาแล้วจะเห็นว่า ระหว่าง U65, U83/3 และ U98 เส้นทางการค้นหาไม่ได้อยู่ในรูปของโครงสร้างต้นไม้ดังแสดงในรูปที่ 5.8 ก



รูปที่ 5.8 ก.



รูปที่ 5.8 ข.

รูปที่ 5.8 แสดงการแก้ไขเส้นทางการค้นหาข้อขัดข้องบนสัญญาณ READY ซึ่งไม่ได้อยู่ในรูปโครงสร้างต้นไม้ ให้อยู่ในรูปโครงสร้างต้นไม้ (ก). เส้นทางการค้นหาเดิม (ข). เส้นทางการค้นหาที่แก้ไขให้อยู่ในรูปโครงสร้างต้นไม้แล้ว

เพื่อให้การค้นหาข้อขัดข้องสามารถดำเนินไปได้ตลอดเส้นทาง จึงจำเป็นต้องดัดแปลงเส้นทางการค้นหาให้อยู่ในรูปของโครงสร้างต้นไม้เสียก่อนซึ่งสามารถทำได้หลายวิธี ในที่นี้จะใช้วิธีการแบ่ง U83/3 ออกเป็น 2 ส่วนโดยการตั้งชื่อให้แตกต่างกันเล็กน้อย ซึ่งจะทำให้กลไกการอนุมานมองเห็นวงจรเป็นดังรูปที่ 5.8 ข. เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

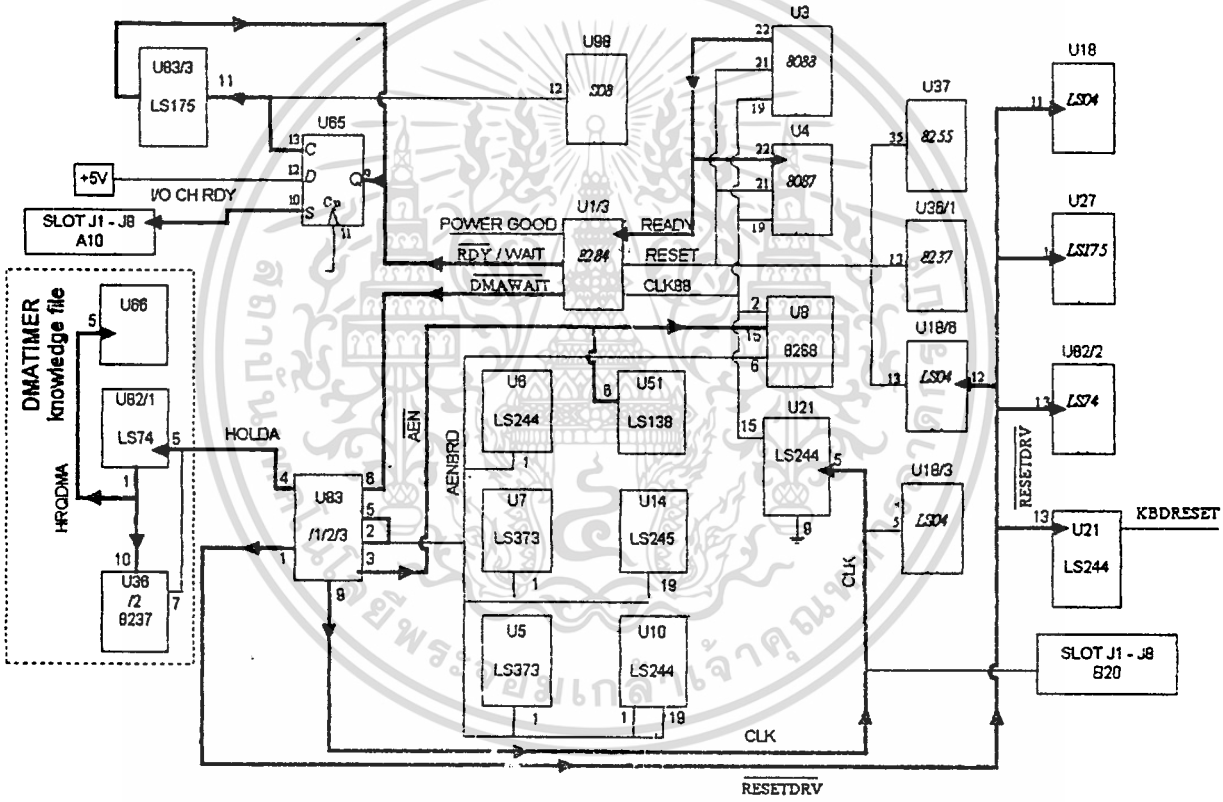
อยู่ในรูปโครงสร้างต้นไม้แล้ว จึงสามารถนำสัญญาณอ้างอิงไปเก็บไว้ในแฟ้มความรู้ได้ดังแสดงอยู่ในแฟ้มความรู้ชื่อ RUNCPU.DBF ในภาคผนวกที่ 4

การค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้ Synchronous กับ สัญญาณ CLK88 (U1 ขา 8)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⌐‾_) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณตามจุดต่างๆ

3. ทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ 5.9



รูปที่ 5.9 เส้นทางการค้นหาข้อขัดข้องบนสัญญาณ READY

ผลการค้นหาข้อขัดข้อง ดังแสดงอยู่ในตารางผลการค้นหาข้อขัดข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง ผลการค้นหาข้อขัดข้องจากสัญญาณ READY

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
U3	22	short	43b	
		open	60b	
U4	22	short	70	open เครื่องสามารถทำงานได้
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
U1/3	5	short	43a	
		open	43a	
	7	short	การทดสอบเบื้องต้น	ต่ออยู่กับ VCC เพื่อให้เป็นสถานะ "1"
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	เมื่อ open จึงไม่มีผลต่อการทำงาน
	3	short	43b	
		open	43b	
	4	short	43b	
		open	43b	
U65	9	short	43a	
		open	43a	
	13	short	43a	
		open	31a หรือ 43b	
	12	short	การทดสอบเบื้องต้น	ต่ออยู่กับ VCC เพื่อให้เป็นสถานะ "1"
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	เมื่อ open จึงไม่มีผลต่อการทำงาน
	10	short	43b	
		open	31a, 43b หรือ 60b	
SLOT J8-J1	A 10	short	43a	open เครื่องสามารถทำงานได้
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
U83/3	11	short	ตรวจไม่พบ	ปกติระดับสัญญาณที่ขานี้ = "0" และเมื่อ
		open	43a	short สัญญาณ READY ไม่ผิดปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
	f2	short	ตรวจไม่พบ	ปกติระดับสัญญาณที่ขานี้ = "0" และเมื่อ
		open	43a	short สัญญาณ READY ไม่ผิดปกติ
U98	12	short	ตรวจพบได้จากการทดสอบที่ 6	open เครื่องสามารถทำงานขั้นต้นได้
		open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
U83/2	6	short	ตรวจไม่พบ	short สัญญาณ READY ไม่ผิดปกติ
		open	43a	
	1	short	43b	ปกติระดับสัญญาณที่ขานี้ = "1"
		open	ตรวจไม่พบ	open เครื่องสามารถทำงานขั้นต้นได้
	9	short	ตรวจพบได้จากการทดสอบที่ 6	short สัญญาณ READY ไม่ผิดปกติ
		open	43b หรือ 60b	
	5	short	ตรวจพบได้จากการทดสอบที่ 7	short สัญญาณ READY ไม่ผิดปกติ
		open	43b	
U83/1	2	short	ตรวจพบได้จากการทดสอบที่ 7	short สัญญาณ READY ไม่ผิดปกติ
		open	31a, 43b หรือ 60b	
	4	short	ตรวจพบได้จากการทดสอบที่ 6	short สัญญาณ READY ไม่ผิดปกติ
		open	43b หรือ 31a	
U8	6	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจพบได้จากการทดสอบที่ 4	
U6	1	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจพบได้จากการทดสอบที่ 4	
U7	1	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจพบได้จากการทดสอบที่ 4	
U14	19	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจพบได้จากการทดสอบที่ 4	
U5	1	short	ตรวจพบได้จากการทดสอบที่ 7	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
		open	ตรวจพบได้จากการทดสอบที่ 4	
U10	1,	short	ตรวจพบได้จากการทดสอบที่ 7	
	19	open	ตรวจพบได้จากการทดสอบที่ 4	
U18/6	12	short	43a	open เครื่องสามารถทำงานขั้นต้นได้
		open	ตรวจไม่พบ	แต่แสดงรหัสผิดพลาด " 0400 7F 201"
U82/2	13	short	70	
		open	43b หรือ 60b	
U18/5	11	short	70	open สัญญาณ READY ไม่ผิดปกติ
		open	ตรวจพบได้จากการทดสอบที่ 6	
U27	1	short	70	
		open	70	
SLOT	B	short	43a	open เครื่องสามารถทำงานได้
J8-J1	20	open	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	
U18/3	5	short	ตรวจพบได้จากการทดสอบที่ 6	
		open	ตรวจพบได้จากการทดสอบที่ 6	
U21	15	short	70	open สัญญาณ READY ไม่ผิดปกติ
		open	ตรวจพบได้จากการทดสอบที่ 6	
	9	short	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	ต่ออยู่กับ GND เพื่อให้เป็นสถานะ "0"
		open	32b	เมื่อ short จึงไม่มีผลต่อการทำงาน
	5	short	ตรวจไม่พบ	short สัญญาณ READY ไม่ผิดปกติ
		open	43b หรือ 60b	
U82/1	5	short	ตรวจพบได้จากการทดสอบที่ 6	short สัญญาณ READY ไม่ผิดปกติ
		open	43a	
	1	short	43b	
		open	31a หรือ 43b	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
U36/2	10	short	43a	
		open	43a	
U36/2	7	short	ตรวจพบได้จากการทดสอบที่ 6	short สัญญาณ READY ไม่ผิดปกติ
		open	ตรวจพบได้จากการทดสอบที่ 6	open สัญญาณ READY ไม่ผิดปกติ
U66	5	short	70	
		open	70	

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	73	สาเหตุ
สามารถตรวจพบได้ขณะนี้	43	สาเหตุ
สามารถตรวจพบได้ข้างหน้า	24	สาเหตุ
ไม่สามารถตรวจพบได้	6	สาเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเรียนรู้ จากผลการทดสอบที่ 1, 2 และ 3

จากเส้นทางการค้นหาข้อขัดข้องของการทดสอบที่ 1 และ 2 จะเห็นว่าข้อมูลอยู่ในโครงสร้างต้นไม้เรียบร้อยแล้ว สามารถนำไปใส่ในแฟ้มความรู้ได้โดยไม่ต้องแก้ไข แต่มีสาเหตุของข้อขัดข้องบางสาเหตุไม่สามารถตรวจพบได้ เนื่องจากสัญญาณที่เฝ้าตรวจไม่มีการผิดปกติ จึงต้องอาศัยการค้นหาจากทดสอบที่ 3 ซึ่งมีเส้นทางการค้นหาบางเส้นทางวนมาบรรจบกับเส้นทางการทดสอบที่ 1 และ 2 เนื่องจากข้อมูลของเส้นทางการค้นหาข้อขัดข้องของการทดสอบที่ 1 และ 2 มีขนาดน้อยเกินไปที่จะบรรจุอยู่ในแฟ้มหนึ่งๆ ดังนั้นจึงรวมเอาข้อมูลของการทดสอบทั้ง 3 ให้อยู่ในแฟ้มเดียวกันคือ RUNCPU.DBF แต่ผลที่ตามมาก็คือทำให้โครงสร้างข้อมูลเส้นทางการค้นหาเดิมไม่ได้อยู่ในโครงสร้างต้นไม้ต่อไปอีก จึงต้องมีการปรับปรุงแก้ไขเพิ่มเติมให้อยู่ในโครงสร้างต้นไม้ และสามารถใช้ได้กับเส้นทางการค้นหาที่วนมาบรรจบกันได้ ดังนี้

จากรูปที่ 5.6, 5.7 และ 5.9 แสดงเส้นทางการค้นหาข้อขัดข้องของการทดสอบทั้ง 3 จะเห็นว่ามีเส้นทางที่วนมาบรรจบกัน 2 เส้นทางคือ ที่ U21 และ U18/6 ซึ่งในการปรับปรุงแฟ้มข้อมูลของการทดสอบ ได้ทำการเพิ่มเติมข้อมูลใหม่ที่มีลักษณะคล้ายกับข้อมูลเดิม คือ U21. , U18/6. และ U10. เข้าไปในแฟ้มความรู้ เพื่อให้กลไกการอนุมานมองเห็นเป็นโหนดใหม่ของโครงสร้างต้นไม้ แต่ผู้ใช้ยังคงเห็นเป็นอุปกรณ์ตัวเดิมอยู่ จากนั้นก็เพิ่มเติมชื่อสัญญาณใหม่ ที่แสดงถึงการเรียกใช้ เส้นทางการค้นหาที่วนกลับมาใช้เส้นทางเดิมซ้ำ คือ reset_reset และ clk88_clk88 ผลของการดัดแปลงเพิ่มเติมนี้ทำให้สามารถค้นหาข้อขัดข้องได้ครบวงรอบการค้นหา ทั้งนี้เนื่องมาจากการเรียนรู้ของการทดสอบครั้งก่อนๆนั่นเอง และข้อมูลที่ดัดแปลงแล้วนี้อยู่ในแฟ้มความรู้ชื่อ RUNCPU.DBF ในภาคผนวกที่ 4

ตัวอย่าง สาเหตุข้อขัดข้อง

จากการทดสอบที่ผ่านมาแสดงถึง การวิเคราะห์เส้นทางและการค้นหาข้อขัดข้อง ซึ่งจะเห็นว่าอาการข้อขัดข้องหนึ่งๆสามารถมีสาเหตุได้มากมายหลายสาเหตุ และต่อไปนี้เป็นจะแสดงตัวอย่างสาเหตุข้อขัดข้อง

สาเหตุข้อขัดข้องเนื่องจากการลัดวงจร

จากรูปที่ 5.9 หน้าที่ 96 สมมุติว่ามีเศษโลหะตกลงไปในช่องเสียบการ์ด ทำให้เกิดการลัดวงจรระหว่างสัญญาณ I/O CHRDY กับ GND ซึ่งจะทำให้ U65 ถูกเซต ตลอดเวลา สัญญาณ $\overline{\text{RDY}}$/WAIT จึงค้างหรือติดอยู่ที่สภาวะ "1" เป็นผลให้ U1 (8284) ขับสัญญาณ READY ออกมาเป็น "0" ตลอดเวลา ผลของข้อขัดข้องที่เกิดขึ้นทำให้ CPU(8088) ค้างหรือหยุดชะงักการทำงานตลอดไปจนกว่าสัญญาณ READY จะกลับคืนสู่สภาวะปกติ อาการข้อขัดข้องที่ปรากฏคือ เครื่องได้รับพลังงานจากแหล่งจ่ายปกติ แต่ไม่ทำงาน, จอภาพมืดและเครื่องไม่มีการตอบสนองใดๆต่อผู้ใช้

สาเหตุข้อขัดข้องเนื่องจากวงจรเปิด

ในหน้าที่ 93 จากรูปแสดงส่วนประกอบภายในของ U1(8284) สมมุติว่าขา 13 หรือขา 1 เกิดวงจรเปิด เช่น เสียบเข้ากับซ็อกเกตไม่แน่น, ซ็อกเกตสกปรก หรือขาของซ็อกเกตหลวมหลุดออกจาก ลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจร ซึ่งจะทำให้ U1 ทำงานไม่ปกติ คือไม่มีสัญญาณ PCLK ออกไป และสัญญาณ READY ผิดปกติ ซึ่งผลของสัญญาณ READY ผิดปกติดังได้กล่าวมาแล้ว สำหรับสัญญาณ PCLK ที่ขา 2 (ดังแสดงในรูปที่5.15 ข หน้าที่ 116) เมื่อไม่มีสัญญาณนี้จะทำให้ วงจร TIMER U49(8253) ไม่ทำงานเพราะไม่มีสัญญาณนาฬิกาเข้ามา ซึ่งจะทำให้ไม่มีสัญญาณ OUT1 และ DRQ0 ไปกระตุ้นให้วงจร DMA U36(8237) ผลิตสัญญาณไปรีเฟรชหน่วยความจำ และยังทำให้สัญญาณ HRQDMA ผิดปกติไปด้วย ซึ่งจะส่งผลที่ผิดปกตินี้ต่อไปยัง U83 และ U1 ตามลำดับ ผลลัพธ์ที่สำคัญเนื่องจากข้อขัดข้องนี้ก็คือ CPU ค้างหรือยุดชะงักการทำงาน อาการข้อขัดข้องที่ปรากฏคือ เครื่องได้รับพลังงานจากแหล่งจ่ายปกติ แต่ไม่ทำงาน, จอภาพมืดและเครื่องไม่มีการตอบสนองใดๆ ต่อผู้ใช้

สาเหตุข้อขัดข้องเนื่องจากตัวอุปกรณ์เสีย

ในหน้าที่ 111 จากรูปที่ 5.13 ก ที่ตัวอุปกรณ์ U12(74LS20) ซึ่งเป็นตัวอุปกรณ์ NAND gate แบบ 4 อินพุต 2 ตัวอยู่ใน chip เดียวกันดังที่ปรากฏในรูปเป็น U12/1 และ U12/2 ตามลำดับ สมมุติว่า U12/2 เสีย ทำให้ถอดรหัสแอดเดรส XA19-XA16 ไม่ได้ ซึ่งก็จะทำให้ไม่มีสัญญาณไปควบคุม U83(74LS138) ทำให้ PROM หรือ EPROM ทุกตัวรวมทั้ง BIOS ด้วย ไม่ถูกเลือกให้ทำงาน และ CPU ก็ไม่สามารถอ่านโปรแกรมใน BIOS ได้ อาการข้อขัดข้องที่ปรากฏก็คือ เครื่องได้รับพลังงานจากแหล่งจ่ายปกติ แต่ไม่ทำงาน, จอภาพมืดและเครื่องไม่มีการตอบสนองใดๆต่อผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

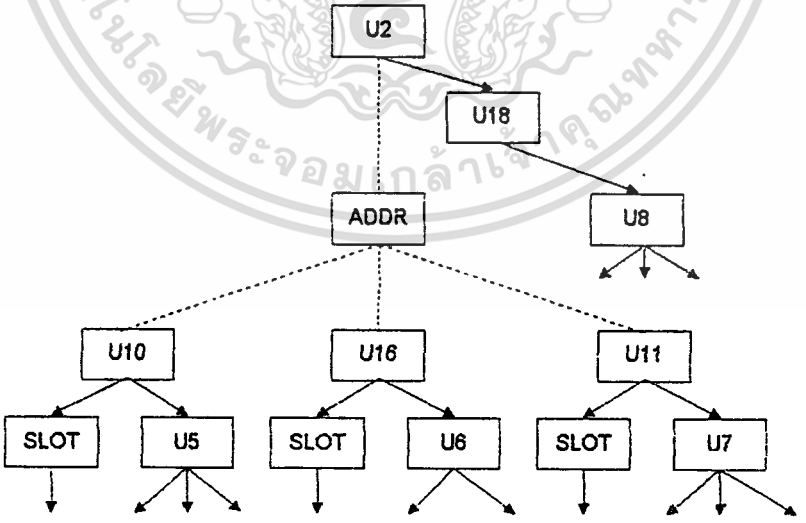
การทดสอบที่ 4

การค้นหาข้อขัดข้องเมื่อสัญญาณ ADDRESS A11-A0ไม่ปกติ

การวิเคราะห์เส้นทางการค้นหา

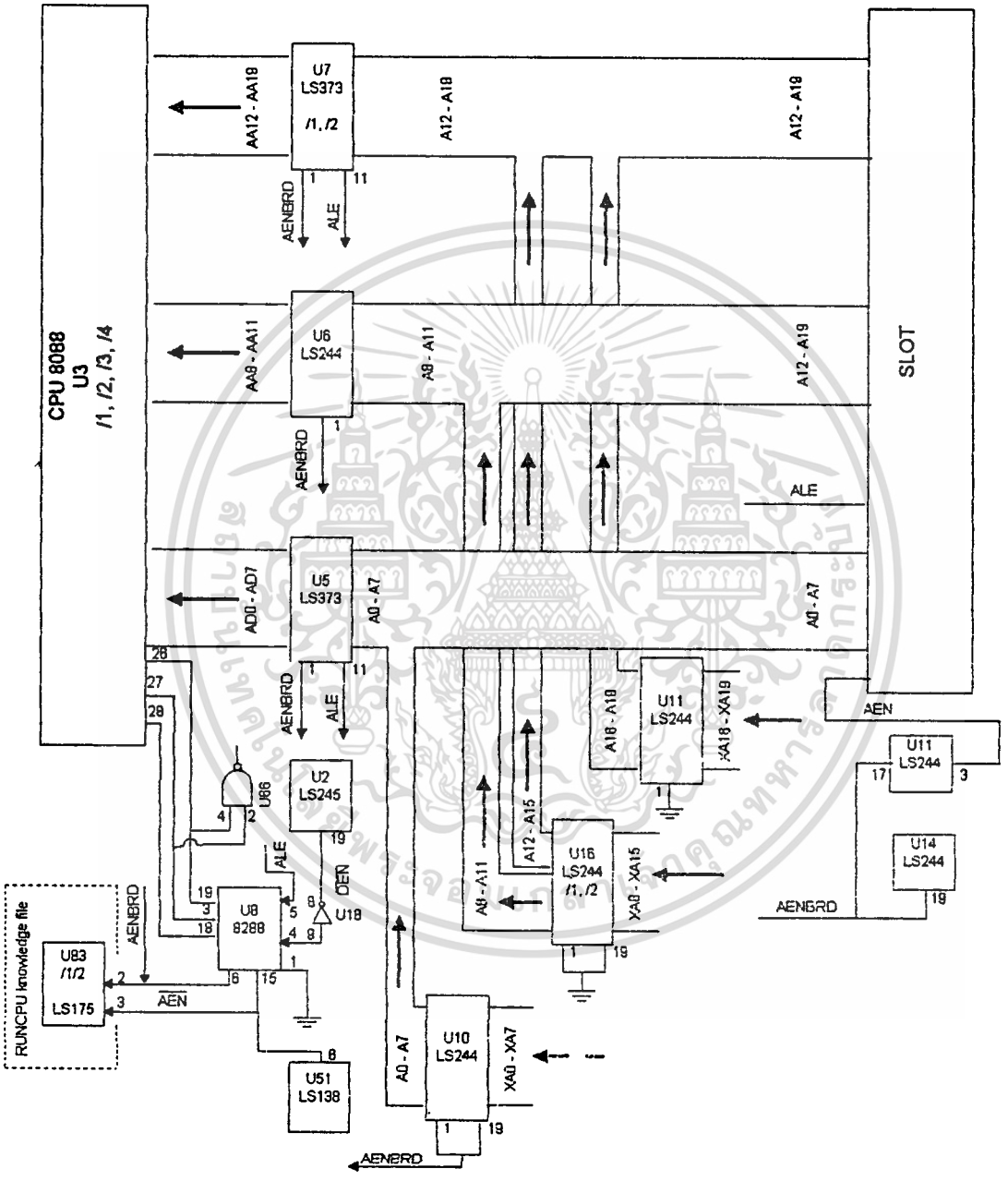
เมื่อการทดสอบดำเนินมาถึงการทดสอบที่ 4 นี้แล้ว ตามลำดับการทำงานของแผงวงจรหลัก สัญญาณ address A11-A0 จะต้องมีสถานะเปลี่ยนแปลงขึ้น - ลงตลอดเวลา ดังนั้นในการจับสัญญาณเพื่อทดสอบจะต้องจับให้ถูกต้องตามห้วงเวลาการทำงานของอุปกรณ์ ในการทดลองใช้การจับสัญญาณในระหว่างห้วงเวลาของสัญญาณ $\overline{DEN}$ และ ALE มีสถานะเป็น "0" แต่เนื่องจากสัญญาณดังกล่าวได้มาจาก U8(8288) ดังนั้นจึงต้องทำการทดสอบอุปกรณ์ตัวนี้ก่อน ถ้าสัญญาณที่ออกจากอุปกรณ์ตัวนี้ปกติจึงจะทำการทดสอบสัญญาณ address ต่อไป ดังรูปที่ 5.10

รูปที่ 5.11 แสดงเส้นทางการค้นหาข้อขัดข้อง ซึ่งเมื่อพิจารณาแล้วจะเห็นว่า ถึงแม้ว่าเส้นทางจะอยู่ในรูปโครงสร้างต้นไม้แล้วก็ตาม แต่ก็ประกอบไปด้วยต้นไม้หลายต้น ดังนั้นจึงต้องทำการรวมต้นไม้เหล่านั้นเข้าด้วยกันดังแสดงในรูปที่ 5.10 โดยการสร้างโหนดเทียมขึ้นมาเป็นโหนดราก ชื่อว่า ADDR อุปกรณ์ที่โหนดรากนี้ไม่มีตัวตนจริงๆ แต่ตั้งชื่อขึ้นมาเพื่อช่วยให้กลไกการอนุมานสามารถเดินทางไปตามเส้นทางการค้นหาข้อขัดข้องบนโครงสร้างต้นไม้ทั้งสามได้ (ซึ่งแสดงเส้นทางด้วยเส้นประ) หลังจากนั้นจึงสามารถนำสัญญาณอ้างอิงไปเก็บไว้ในแฟ้มความรู้ได้ดังแสดงอยู่ในแฟ้มความรู้ชื่อ ADDRBUS1.DBF ในภาคผนวกที่ 5



รูปที่ 5.10 แสดงการรวมโครงสร้างต้นไม้หลายต้นเพื่อใช้ในการค้นหาข้อขัดข้องเมื่อสัญญาณ ADDRESS ของระบบไม่ปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.11 เส้นทางการค้นหาข้อขัดข้องของการทดสอบที่ 4 และ ที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้ Synchronous กับ สัญญาณ CLK88 (U1 ขา 8)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' แล้วหน่วงเวลาไป 121 มิลิ - วินาที (⌐¯¯_ delay 121 mS) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณ AENBRD

ถ้าปรากฏว่าจับได้เป็น ...⌐¯¯_⌐¯¯¯¯¯¯¯¯_⌐¯¯_...ให้ข้ามไปการทดสอบที่ 6

แต่ถ้าจับได้แล้วไม่เป็นรูปดังกล่าวให้ดำเนินการต่อไปในข้อ 3

3. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⌐¯¯_) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณ $\overline{DEN}$

ถ้าปรากฏว่าจับได้เป็น ¯¯_⌐¯¯_⌐¯¯_⌐¯¯ ให้ใช้สัญญาณนี้ร่วมกับสัญญาณ ALE ในห้วงเวลาที่มีสถานะเป็น "0" ทั้งคู่ ทำการจับสัญญาณ address ตามจุดทดสอบต่างๆ และดำเนินการต่อไปในข้อ 4

แต่ถ้าจับได้แล้วไม่เป็นรูปดังกล่าวให้ดำเนินการต่อไปในข้อ 4 ได้เลย

4. ทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ 5.11

ผลการค้นหาข้อขัดข้อง ดังแสดงอยู่ในตารางผลการค้นหาข้อขัดข้อง

ตารางผลการค้นหาข้อขัดข้องจากสัญญาณ ADDRESS ของระบบไม่ปกติ

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
u2	19	short	43b	
		open	31a หรือ 43b	
u18	8	short	43a	
		open	43a	
	9	short	43b	
		open	43b	
u8	4	short	43a	
		open	31a หรือ 43a	
	1	short	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	ต่ออยู่กับ GND เพื่อให้เป็นสถานะ "0"
		open		เมื่อ short ไม่มีผลต่อการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์	ขา	สาเหตุ	ตรวจพบโดยกฎ/หรือ	หมายเหตุ
u8	15	short	43b	
		open	ตรวจพบได้จากการทดสอบที่ 7	
	6	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	31a หรือ 43b	
	18	short	43b	ปกติระดับสัญญาณที่ขานี้ = "1"
		open	ตรวจไม่พบ	และเมื่อ open สัญญาณ $\overline{\text{DEN}}$ ไม่ผิดปกติ
	3	short	43b	
		open	31a หรือ 43b	
	19	short	43b	
		open	31a หรือ 43b	
	5	short	43a	
		open	43a	
u51	6	short	43b	
		open	ตรวจพบได้จากการทดสอบที่ 7	
u83	2	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจพบได้จากการทดสอบที่ 3	
	3	short	43a	
		open	ตรวจพบได้จากการทดสอบที่ 7	
u66	4	short	43b	
		open	31a หรือ 43b	
	2	short	43b	
		open	31a หรือ 43b	
u3/1	26	short	43a	
		open	43a	
	27	short	43a	
		open	43a	
	28	short	43a	ปกติระดับสัญญาณที่ขานี้ = "1"
		open	ตรวจไม่พบ	และเมื่อ open สัญญาณ $\overline{\text{DEN}}$ ไม่ผิดปกติ
u10		short	43a	ขา 3, 5, 7, 9, 18, 12, 14, 16
		open	43a	XA0 - XA7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u10		short	43b	ขา 17,15, 13, 11, 2, 8, 6, 4
		open	31a, 43b หรือ 60b	A0 - A7
	1,	short	ตรวจพบได้จากการทดสอบที่ 7	
	19	open	43b หรือ 60b	
u5		short	43a	ขา 2, 19, 16, 5, 6, 15, 12, 9
		open	43a	A7 - A0
		short	43b	ขา 3, 18, 17, 4, 7, 14, 13, 8
		open	31a หรือ 43b	AD7 - Ad0
	1	short	ตรวจพบได้จากการทดสอบที่ 4	
		open	43b หรือ 60b	
	11	short	43b หรือ 60b	
		open	31a, 43b หรือ 60b	
u3/2		short	43a	ขา 16, 15, 14, 13, 12, 11, 9
		open	43a	AD0 - AD7
u16		short	43a	ขา 9, 12, 14, 7, 3, 5, 18, 16
		open	43a	XA8 - XA15
		short	43b	ขา 11, 8, 6, 13, 17, 15, 2, 4
		open	31a, 43b หรือ 60b	A8 - A15
	1,	short	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	ต่ออยู่กับ GND เพื่อให้เป็นสถานะ "0"
	19	open	41	เมื่อ short ไม่มีผลต่อการทำงาน
u6		short	43a	ขา 12, 14, 16, 18
		open	43a	A11 - A8
		short	43b	ขา 8, 6, 4, 2
		open	31a หรือ 43b	AA 11 - AA8
	1	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	43b หรือ 60b	
u3/3		short	43a	ขา 5, 6, 7, 8
		open	43a	AA11 - AA8
u11		short	43a	ขา 18, 16, 14, 12 คือ XA16 - XA19
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		short	43b	ขา 2, 4, 6, 8 คือ A16 - A19
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
	1	short	ไม่ใช่ข้อขัดข้องในงานวิจัยนี้	ต่ออยู่กับ GND
		open	41	
u7		short	43a	ขา 19, 2, 5, 16, 6, 15, 9 คือ A19 - A13
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
		short	43b	ขา 18, 3, 4, 17, 7, 14, 8 คือAA19-AA13
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
	12,	short	43a	คือ A12, AA12
	13	open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
	1	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	43b หรือ 60b	
	11	short	43b หรือ 60b	
		open	31a, 43b หรือ 60b	
u3/4		short	43a	ขา 35 - 39, 2, 3 คือ AA19 - AA13
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
	4	short	43a	AA12
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
SLOT	A	short	43b	A 31, 30, 29, 28, 27, 26,25, 24, 23, 22,
		open	43b หรือ 60b	21, 20 = A00 - A11
	A	short	43b	A12
	19	open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
	A	short	43b	ขา A 18 - 12 คือ A13 - A19
		open	ตรวจไม่พบ	ขณะนี้มีสถานะ = "1"
	B	short	43b หรือ 60b	ALE
	28	open	31a, 43b หรือ 60b	
u14	19	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจพบได้จากการทดสอบที่ 5	
u11	17	short	ตรวจพบได้จากการทดสอบที่ 7	
		open	ตรวจไม่พบ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	294	สาเหตุ
สามารถตรวจพบได้ขณะนี้	240	สาเหตุ
สามารถตรวจพบได้ข้างหน้า	11	สาเหตุ
ไม่สามารถตรวจพบได้	43	สาเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบที่ 5

การค้นหาข้อขัดข้องเมื่อไม่สามารถอ่านข้อมูลบน ROMBIOS ได้

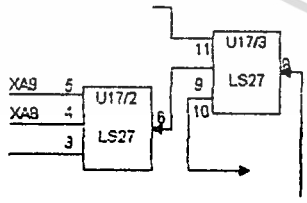
การวิเคราะห์เส้นทางการค้นหา

ในการทดลองก่อนอื่นจะทำการเฝ้าสังเกตสัญญาณ $\overline{S0}$ ก่อน เนื่องจากสัญญาณนี้เป็นตัวแสดงสถานะการทำงานของไมโครโปรเซสเซอร์ 8088 เกี่ยวกับการอ่าน/เขียนข้อมูลที่สำคัญสัญญาณหนึ่งซึ่งสังเกตุได้ง่าย เมื่อสัญญาณนี้ผิดปกติจึงจะทำการติดตามค้นหาสาเหตุไปตามเส้นทางการค้นหาหรือสัญญาณอื่นๆเกี่ยวข้องต่อไป สำหรับสัญญาณ data ซึ่งเป็นสัญญาณที่เปลี่ยนแปลงขึ้นๆ ลงๆ อยู่เสมอเช่นเดียวกับสัญญาณ address ของการทดสอบที่ 4 ในการค้นหาข้อขัดข้องจึงต้องดูสัญญาณ $\overline{S0}$ ประกอบด้วย ซึ่งหมายความว่าสัญญาณที่ปกติจะต้องประกอบด้วยทั้งสัญญาณ $\overline{S0}$ ปกติ และสัญญาณ data ที่ขึ้นๆลงๆ ไม่ค้างอยู่ที่สถานะใดสถานะหนึ่ง ดังนั้นในการเก็บเป็นข้อมูลอ้างอิงจะเก็บไว้เป็นข้อความและแผนผังเวลาดังนี้

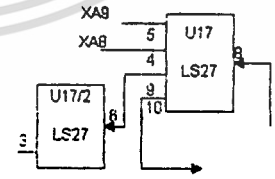
data = not stable AND $\overline{S0}$ =

ดังที่ปรากฏอยู่ในแฟ้มความรู้ READBIOS.DBF ในภาคผนวกที่ 6

รูปที่ 5.13 ก และ 5.13 ข แสดงอุปกรณ์ต่างๆที่มีส่วนร่วมในการที่จะทำให้ ไมโครโปรเซสเซอร์ 8088 สามารถอ่านข้อมูลจาก ROMBIOS ได้หรือไม่ ซึ่งเมื่อพิจารณาแสดงเส้นทางการค้นหาข้อขัดข้อง แล้วจะเห็นว่าระหว่าง U17 และ U17/2 แม้ว่าเส้นทางการค้นหาจะอยู่ในรูปของโครงสร้างต้นไม้แล้วก็ตาม แต่จากการ ทดลองปรากฏว่า ถ้าเหตุการณ์ชำรุดปรากฏอยู่บนสัญญาณ XA9 หรือ XA8 สัญญาณที่ต่ออยู่ระหว่าง U17 ขา 6 และขา 9 จะไม่เกิดการผิดปกติแต่อย่างใด ซึ่งจะทำให้การติดตามค้นหาข้อขัดข้องต้องหยุดชะงักลงไม่สามารถติดตามต่อไปได้ ดังนั้นจึงต้องมีการปรับปรุงแฟ้มความรู้เสียใหม่ จากรูปที่ 5.12 ก ให้เป็นดังรูปที่ 5.12 ข



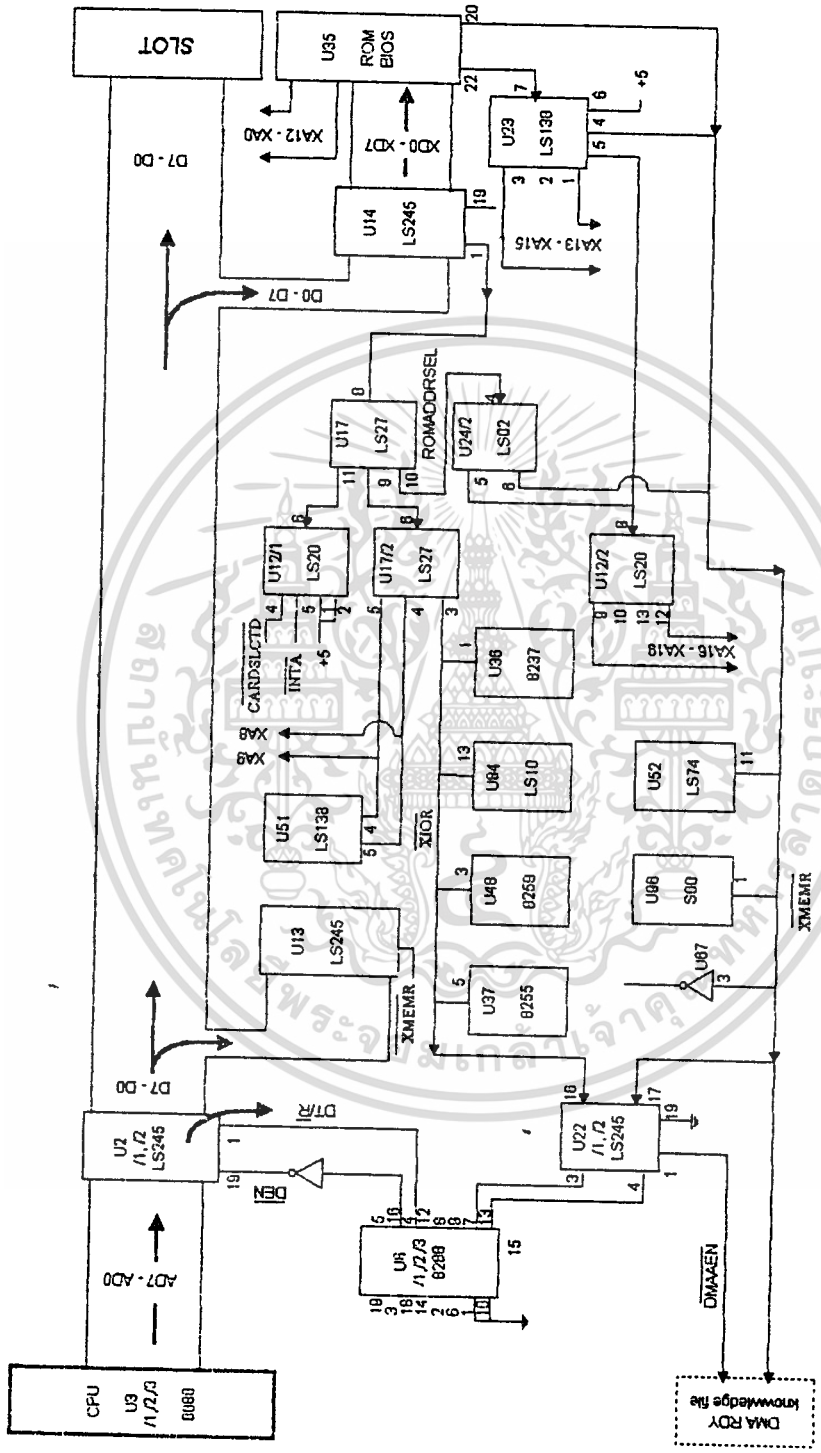
รูปที่ 5.12 ก



รูปที่ 5.12 ข

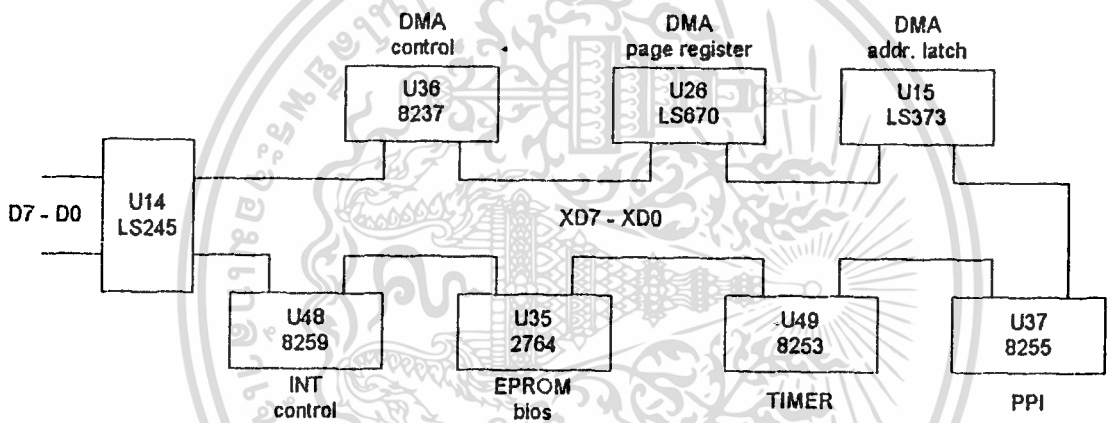
รูปที่ 5.12 แสดงการปรับปรุงแฟ้มความรู้ READBIOS.DBF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า



รูปที่ 5.13 ก เส้นทางการค้นหาข้อผิดพลาดของบนสัญญาณการอ่าน ROMBIOS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.13 ข เส้นทางการค้นหาข้อขัดข้องบนสัญญาณ XD7 - XD0

การค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้ Synchronous กับ สัญญาณ CLK88 (U1 ขา 8)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⌐‾_) แล้วรอคอย สัญญาณที่ U3 ขา 26 ($\overline{S0}$) ช่วงสถานะเป็น 'L' เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณตามชุดต่างๆ

3. ทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ ผ 5.2 ก. และ ผ 5.2 ข.

ผลการค้นหาข้อขัดข้อง สรุปได้ดังนี้

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	315	สาเหตุ
สามารถตรวจพบได้ขณะนี้	281	สาเหตุ
ไม่สามารถตรวจพบได้	34	สาเหตุ

สาเหตุที่ไม่สามารถตรวจพบได้

1. สัญญาณที่เฝ้าสังเกต $\overline{S0}$ ไม่ผิดปกติ แม้ว่าจะมีข้อขัดข้องเกิดขึ้นก็ตาม
2. สัญญาณอ้างอิง ที่เก็บอยู่ในสถานะ "1" ไม่สามารถเปรียบให้เห็นข้อแตกต่าง กับ ข้อขัดข้องที่เกิดจากวงจรเปิดได้
3. สัญญาณอ้างอิง ที่เก็บอยู่ในสถานะ "0" ไม่สามารถเปรียบให้เห็นข้อแตกต่าง กับ ข้อขัดข้องที่เกิดจากการลัดวงจรได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างแฟ้มข้อมูลโดยไม่ต้องใช้โปรแกรมการจัดเรียงขาอุปกรณ์

จากประสบการณ์ การเรียนรู้จากการทดสอบที่ผ่านมาจะพบว่าสาระที่สำคัญของแฟ้มข้อมูลคือ

1. การกำนดหมายเลขกลุ่ม หรือค่าฮิวริสติกฟังก์ชั่นให้กับขาต่างๆของอุปกรณ์ โดยมีจุดประสงค์เพื่อใช้ในการเลือกเส้นทางที่เหมาะสมในการค้นหาข้อขัดข้อง

2. การแบ่งภาคตัวอุปกรณ์ต่างๆตามเส้นทางการค้นหาข้อขัดข้องและหน้าที่การทำงาน โดยมีจุดประสงค์เพื่อให้ได้แฟ้มข้อมูลที่มีลักษณะเป็นโครงสร้างต้นไม้

3. การควบคุมกลยุทธโดยใช้กฎที่ 42a และ 42b ในการควบคุมการเคลื่อนที่ในการโพรบไปตามขาต่างๆ ของอุปกรณ์ตัวหนึ่งๆกฎทั้งสองนี้จะคอยตรวจสอบว่า ขาต่างๆที่กำลังโพรบอยู่ในขณะใดขณะหนึ่งเป็นขาของอุปกรณ์ตัวเดียวกันหรือไม่ โดยจะทำการเปรียบเทียบชื่อโหนดในระเบียนที่อยู่ติดกัน

จากสาระทั้ง 3 ข้อที่กล่าวมาจะเห็นว่า ในการสร้างแฟ้มข้อมูลของวงจรใดวงจรหนึ่ง อุปกรณ์ที่มีหลายฟังก์ชั่นไม่จำเป็นต้องแบ่งภาคการทำงานโดยการกำหนดชื่อให้แตกต่างกันเสมอไป ซึ่งสามารถที่จะแบ่งภาคได้ด้วยการจัดให้ข้อมูลของแต่ละภาคจะต้องไม่อยู่ในระเบียนที่เรียงชิดติดกัน (ซึ่งจะทำให้การควบคุมกลยุทธมองเห็นเป็นอุปกรณ์ตัวเดียวกัน) นั่นคือในทางปฏิบัติ ผู้เชี่ยวชาญสามารถที่จะวิเคราะห์เส้นทางการค้นหาข้อขัดข้องแล้วใส่ข้อมูลตามไปได้ทันที โดยไม่ต้องกังวลเกี่ยวกับการแบ่งภาคตัวอุปกรณ์ซึ่งจะทำให้ได้แฟ้มข้อมูลที่มีข้อมูลของโหนดต่างๆ ที่ผู้ใช้สามารถเข้าใจได้ง่ายไม่ต้องมีเครื่องหมาย "/" หรืออื่นๆเพื่อแสดงการแบ่งภาค

ข้อเสียของการสร้างแฟ้มความรู้ด้วยวิธีนี้คือ ไม่สามารถใช้โปรแกรมการจัดเรียงลำดับขาอุปกรณ์ COMPO.PRG ได้ ซึ่งถ้าผู้สร้างแฟ้มข้อมูลผลอหรือหลงลืม ก็จะทำให้ได้แฟ้มข้อมูลที่มีเส้นทางการค้นหาที่สับสนได้

แฟ้มข้อมูล DMATIMER.DBF ในภาคผนวกที่ 7 เป็นตัวอย่างการสร้างแฟ้มข้อมูลแบบนี้ ซึ่งจะสังเกตได้ที่ข้อมูลของ U83

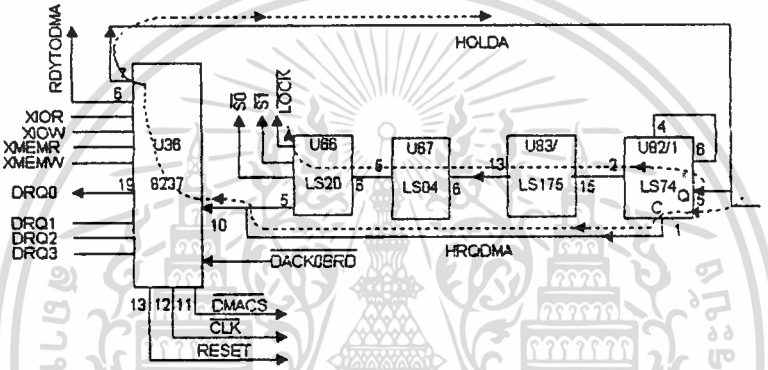
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 6

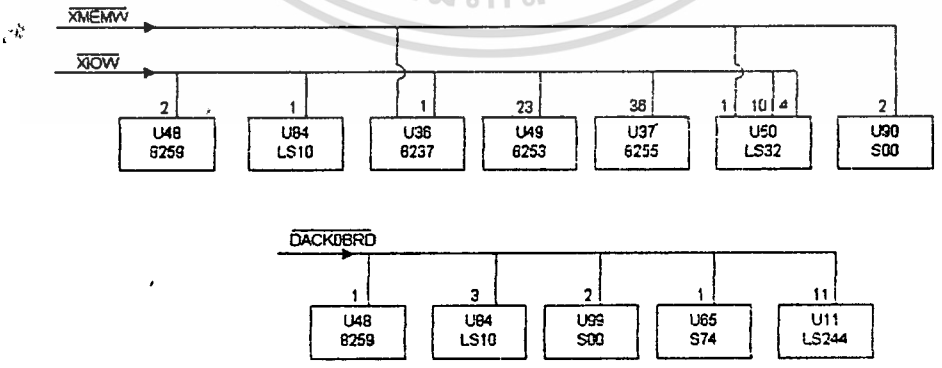
การค้นหาข้อขัดข้องบนวงจร DMA และ TIMER

การวิเคราะห์เส้นทางการค้นหา

ในรูปที่ 5.15แสดงเส้นทางการค้นหาข้อขัดข้องซึ่งเมื่อพิจารณาแล้วจะเห็นว่าระหว่างU82/1กับU36 เส้นทางการค้นหามีลักษณะเป็นวงรอบ และในขณะเดียวกัน U82/1 ก็อาจจะถูกใช้เป็นเส้นทางการค้นหาถึง 2 เส้นทางดังแสดงในรูปที่ 5.14 การแก้ไขปัญหาจึงต้องใช้ทั้งการกำหนดชื่อให้แตกต่างกันไป และการแยกระเบียนข้อมูลของการทำงานที่อยู่คนละเส้นทางการค้นหาไม่ให้อยู่เรียงชิดติดกัน

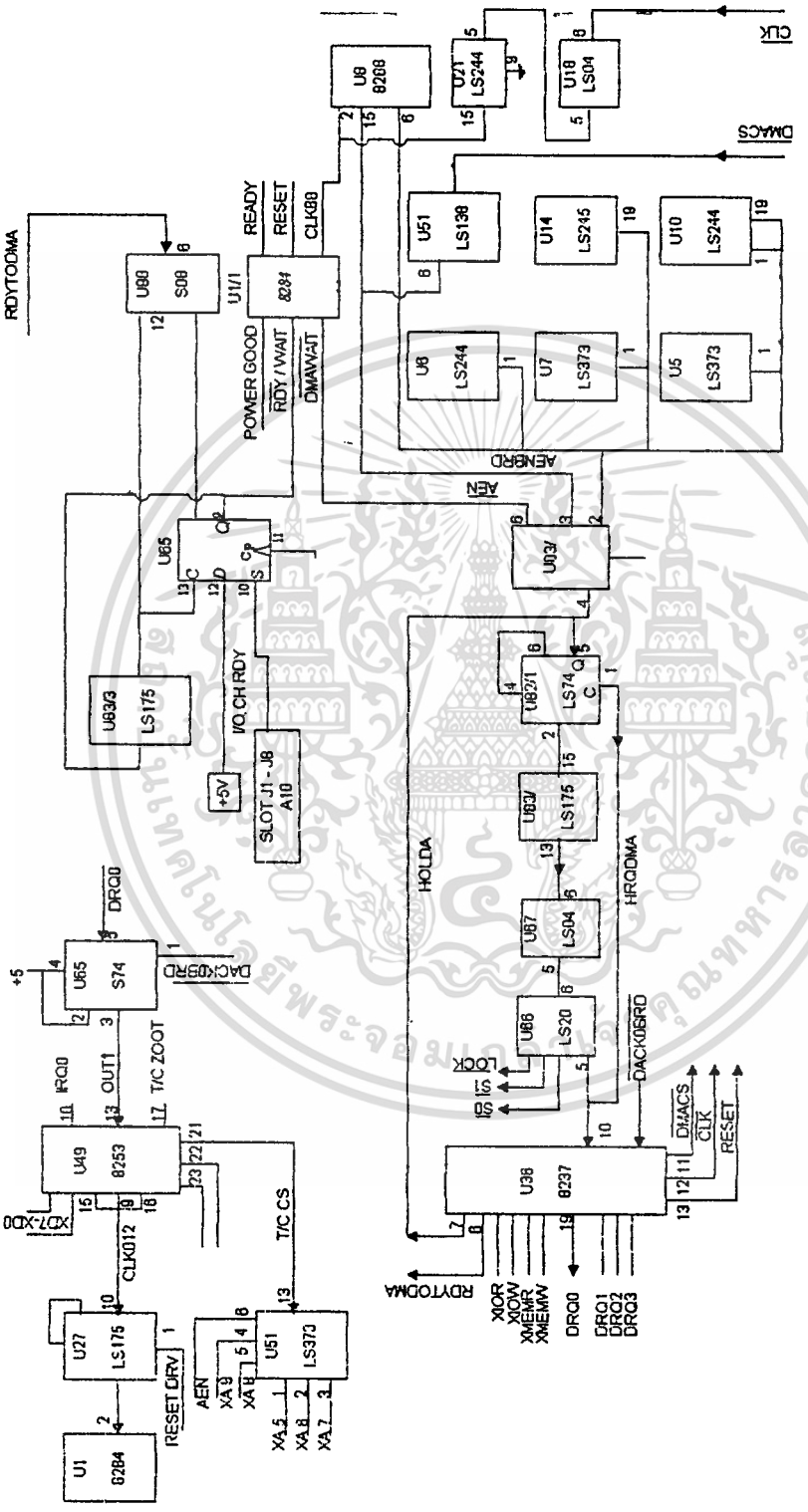


รูปที่ 5.14 เส้นทางการค้นหาข้อขัดข้องบน U82



รูปที่ 5.15 ก เส้นทางการค้นหาข้อขัดข้องบนสัญญาณ $\overline{\text{XMEMW}}$, $\overline{\text{XIOW}}$ และ $\overline{\text{DACK0BRD}}$ ที่ต่อมาจากรูปที่ 5.15 ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.15 ข เส้นทางการค้นหาข้อขัดข้องบนสัญญาณ HRQDMA, DRQ0, และ OUT1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้ Synchronous กับ สัญญาณ PCLK (U1 ขา 2)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⎍‾‾‾‾_) แล้ว หน่วงเวลาประมาณ 120.969 ไมโครวินาที (หรือ PCLK ประมาณ 288 ลูก) ทำการตรวจสอบสัญญาณ AEN

ถ้าปกติ = _⎍_____⎍_____⎍_ ให้ข้ามไปการทดสอบที่ 7

3. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' (⎍‾‾‾‾_) แล้ว หน่วงเวลาประมาณ 7.350 ไมโครวินาที (หรือ CLK ประมาณ 17 ลูก) ทำการตรวจสอบสัญญาณ T/C CS

4. นำผลจากข้อ 2 และ 3 ทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ 5.15 ก. และ 5.15 ข.

ผลการค้นหาข้อขัดข้อง สรุปได้ดังนี้

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	164	สาเหตุ
สามารถตรวจพบได้ขณะนี้	143	สาเหตุ
ไม่สามารถตรวจพบได้	21	สาเหตุ

สาเหตุที่ไม่สามารถตรวจพบได้

1. สัญญาณที่เฝ้าสังเกตุ AEN ไม่ผิดปกติ แม้ว่าจะมีข้อขัดข้องเกิดขึ้นก็ตาม
2. สัญญาณอ้างอิง ที่เก็บอยู่ในสถานะ "1" ไม่สามารถเปรียบให้เห็นข้อแตกต่าง กับ ข้อขัดข้องที่เกิดจากวงจรเปิดได้
3. สัญญาณอ้างอิง ที่เก็บอยู่ในสถานะ "0" ไม่สามารถเปรียบให้เห็นข้อแตกต่าง กับ ข้อขัดข้องที่เกิดจากการลัดวงจรได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

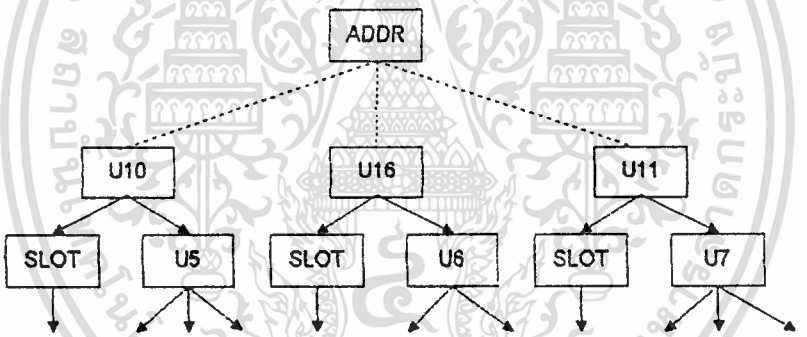
การทดสอบที่ 7

การค้นหาการลัดวงจรบนสัญญาณ AENBRD

การวิเคราะห์เส้นทางการค้นหา

เป็นการค้นหาการลัดวงจรบนสัญญาณ AENBRD ก่อนที่จะไปทำการทดสอบสัญญาณอื่นต่อไป สัญญาณ AENBRD เป็นสัญญาณควบคุม แอดเดรสบัฟเฟอร์ที่ได้มาจากวงจร DMA สำหรับรูปเส้นทางการค้นหาข้อขัดข้องใช้รูปที่ 5.11 จากการทดสอบที่ 4

รูปที่ 5.11 แสดงเส้นทางการค้นหาข้อขัดข้อง ซึ่งการพิจารณาก็เช่นเดียวกับการทดสอบที่ 4คือประกอบไปด้วยต้นไม้หลายต้น ดังนั้นจึงต้องทำการรวมต้นไม้เหล่านั้นเข้าด้วยกันดังแสดงในรูปที่ 5.16 โดยการสร้างโหนดเทียมขึ้นมาเป็นโหนดราก ชื่อว่า ADDR เพื่อช่วยให้กลไกการอนุมานสามารถเดินทางไปตามเส้นทางการค้นหาข้อขัดข้องบนโครงสร้างต้นไม้ได้ เช่นเดิม (ซึ่งแสดงเส้นทางด้วยเส้นประ) หลังจากนั้นจึงสามารถนำสัญญาณอ้างอิงไปเก็บไว้ในแฟ้มความรู้ได้ดังแสดงอยู่ในแฟ้มความรู้ชื่อ ADDRBUS2.DBF ในภาคผนวกที่ 8



รูปที่ 5.16 แสดงการรวมโครงสร้างต้นไม้หลายต้นเพื่อใช้ในการค้นหาข้อขัดข้อง เมื่อสัญญาณ AENBRD ของระบบไม่ปกติ

การค้นหาข้อขัดข้อง

1. ตั้งค่า(set) Logic Analyzer ให้ Synchronous กับ สัญญาณ PCLK (U1 ขา 2)

2. ใช้สัญญาณที่ U1 ขา 10(RESET)ช่วงเปลี่ยนสถานะจาก 'H' เป็น 'L' แล้วหน่วงเวลาไป 121 มิลิ - วินาที (⌐¯¯¯¯¯¯¯¯¯¯¬ delay 121 mS) เป็นตัวกระตุ้น(trigger) ให้ทำการจับสัญญาณ AENBRD และสัญญาณ $\overline{\text{DEN}}$

3.ใช้สัญญาณ $\overline{\text{DEN}}$ ร่วมกับสัญญาณ ALE ในห้วงเวลาที่มีสถานะเป็น "0" ทั้งคู่ ทำการจับสัญญาณ address ตามจุดทดสอบต่างๆ แล้วทำการค้นหาข้อขัดข้องตามเส้นทาง ในรูปที่ 5.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการค้นหาข้อขัดข้อง สรุปได้ดังนี้

รวมสาเหตุที่ทำการค้นหาทั้งสิ้น	275	สาเหตุ
สามารถตรวจพบได้ขณะนี้	241	สาเหตุ
ไม่สามารถตรวจพบได้	34	สาเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

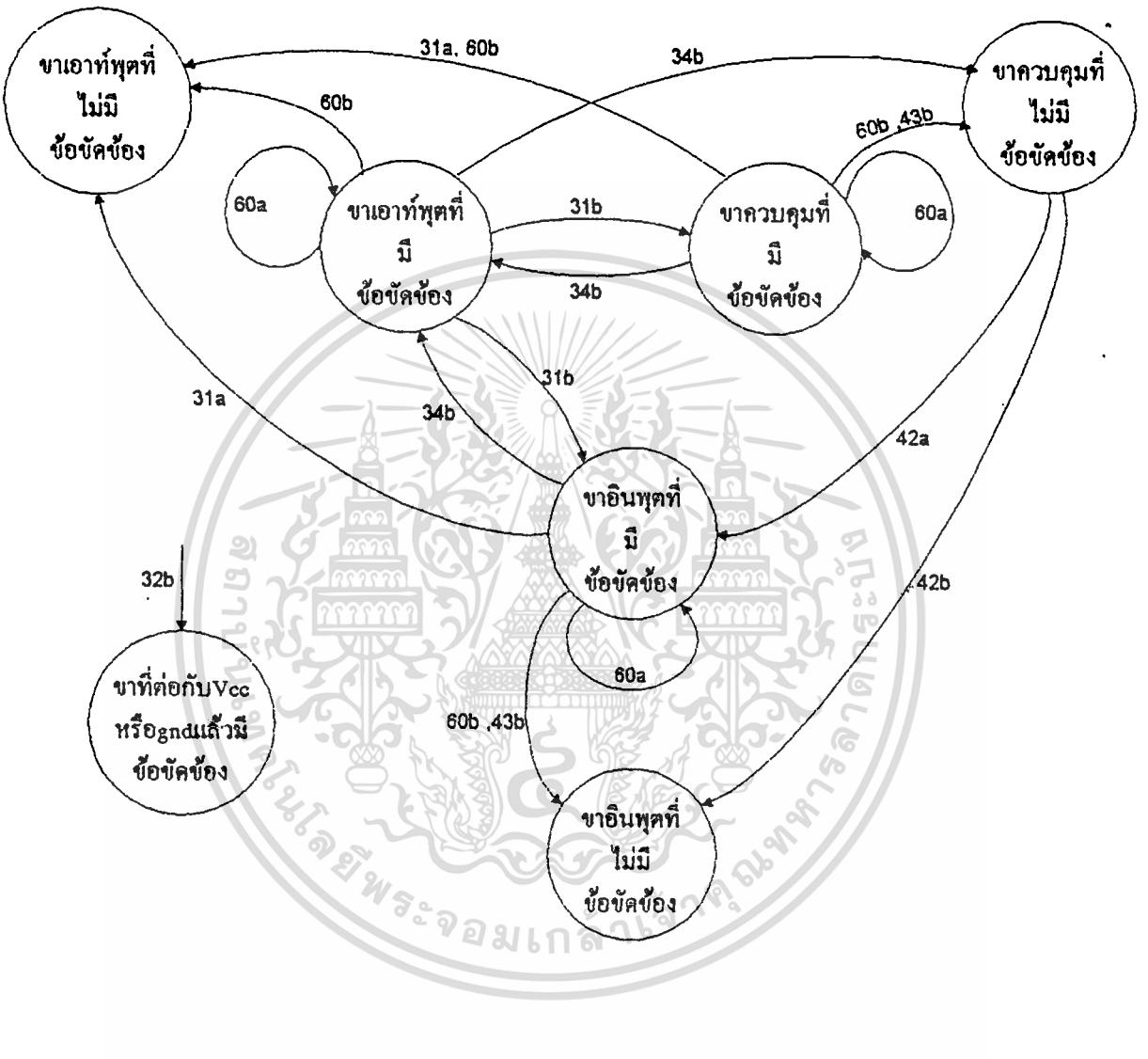
บทที่ 6
สรุปและข้อเสนอแนะ

6.1 ลักษณะทั่วไป

จุดประสงค์หลักของการค้นหาข้อขัดข้องบนแผงวงจร คือต้องการค้นหาว่าข้อขัดข้องนั้นเกิดขึ้นได้อย่างไรและอยู่ที่ไหน ค่าใช้จ่ายในการค้นหาข้อขัดข้องประกอบด้วยปัจจัยหลัก 2 ประการ ปัจจัยประการแรกคือเครื่องมือซึ่งประกอบไปด้วย Hardware และ Software ปัจจัยประการที่สอง คือ กิจกรรมซึ่งประกอบไปด้วยตัวบุคลากรและเวลา ระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ เป็นการนำเอาความรู้ความชำนาญของช่างเทคนิคอาวุโส หรือวิศวกรผู้เชี่ยวชาญมาใส่ไว้ในฐานความรู้ เพื่อช่วยให้ช่างเทคนิคปัจจุบันสามารถค้นหาข้อขัดข้องเป็นไปได้อย่างรวดเร็ว และยังเป็นการนำเอาเครื่องมือ เครื่องตรวจวัดที่มีอยู่แล้วมาประยุกต์เข้ากับเทคโนโลยีสารสนเทศ เพื่อใช้ทรัพยากรที่มีอยู่ให้เกิดประโยชน์มากยิ่งขึ้น ระบบนี้นอกจากจะใช้ในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์แล้วยังสามารถใช้ในการค้นหาข้อขัดข้องบนแผงวงจรดิจิตอลทั่วไปได้อีกด้วย โดยสามารถค้นหาได้ทั้งข้อขัดข้องแบบจุดเดียว (Single fault) และข้อขัดข้องแบบหลายจุด (Multi fault) สำหรับชนิดของการทดสอบ ใช้การทดสอบแบบไกด์โพรบทำการทดสอบโดยอาศัย สัญญาณจากแผงวงจรในขณะที่แผงวงจรกำลังทำงานอยู่ ทั้งนี้เพื่อขจัดปัญหาเรื่องฐานเวลาระหว่างเครื่องตรวจวัด และแผงวงจรที่ทำการทดสอบ โดยจะตั้งเครื่องตรวจวัดให้รอคอยเหตุการณ์ที่สนใจ เหตุการณ์ใดเหตุการณ์หนึ่งบนแผงวงจร เมื่อเหตุการณ์นั้นมาถึงก็จะไปกระตุ้น (Trig) ให้เครื่องมือตรวจวัดทำการบันทึกข้อมูลต่างๆที่เกิดขึ้น แล้วนำไปเปรียบเทียบกับค่าที่คาดหวังในฐานความรู้ เพื่อทำการวิเคราะห์ค้นหาสาเหตุ และบริเวณข้อขัดข้องต่อไป

สำหรับกฎเกณฑ์ในการวิเคราะห์ ได้มาจากลักษณะของตัวอุปกรณ์ที่ประกอบด้วย 4 ส่วนหลัก คือ ส่วนไฟเลี้ยง, ส่วนอินพุต, ส่วนควบคุม และส่วนเอาท์พุต และจะถือว่าสถานะส่วนต่างๆเหล่านี้มีอยู่ 2 สถานะ คือ สถานะที่ไม่มีข้อขัดข้อง หรือ สถานะที่มีข้อขัดข้อง อย่างใดอย่างหนึ่ง ถ้ากำหนดให้การโพรบเป็นอินพุต, ผลของการเปรียบเทียบเป็นเอาต์พุต และ กฎต่างๆเป็นฟังก์ชั่นการเปลี่ยนสถานะแล้ว จะสรุปเป็น ไดอะแกรมสถานะของ เครื่องจักรที่มีสถานะจำกัด [19](finite state machine)ได้ดังรูปที่ 6.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.1 ไดอะแกรมสถานะ

สำหรับการทดสอบต่างๆที่ได้ดำเนินการมา ได้ทำการทดสอบบนแผงวงจรหลักไมโครคอมพิวเตอร์ PC/XT ตามลำดับเวลาของเหตุการณ์ต่างๆที่เกิดขึ้น เริ่มตั้งแต่ การทดสอบเบื้องต้น, การทดสอบ CPU, จนกระทั่งถึงการทดสอบวงจร DMA และ TIMER ซึ่งยังมีการทดสอบอีกหลายวงจรที่การวิจัยครั้งต่อไปสามารถที่จะดำเนินการทดสอบต่อไปได้ เช่น วงจรขัดจังหวะ, วงจรเกี่ยวกับหน่วยความจำ เป็นต้น การทดสอบเหล่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ศิวุรจะดำเนินการต่อมาจนกระทั่งถึงเหตุการณ์ที่แผงวงจรหลักสามารถติดต่อกับจอภาพได้และสามารถแสดงข้อมูลข่าวสารออกบนจอภาพได้ ซึ่งหลังจากนั้นการ ค้นหาข้อขัดข้อง ควรจะเป็นหน้าที่ของโปรแกรมทดสอบตัวเองของแผงวงจร

6.2 ข้อจำกัด

ก. เนื่องจากระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรไมโครคอมพิวเตอร์นี้ ใช้ความรู้ที่ได้จากฐานความรู้ซึ่งทำการบันทึกด้วยมือ ดังนั้นประสิทธิภาพในการค้นหาข้อขัดข้องให้ได้ครอบคลุม (Fault coverage) จึงขึ้นอยู่กับ ความชาญฉลาดในการเก็บบันทึกข้อมูลของช่างเทคนิคอาวุโส หรือวิศวกรผู้เชี่ยวชาญเป็นสำคัญ

ข. ในระหว่างการดำเนินการค้นหาข้อขัดข้อง ระบบจะทำการติดต่อกับผู้ใช้ผ่านทางจอภาพและแป้นพิมพ์ ถ้าผู้ใช้หลงลืมหรือตอบคำถามผิดก็จะทำให้การวิเคราะห์ผิดพลาดไปด้วย

ค. เนื่องจากการวิเคราะห์นี้ได้มาจากการรอคอยที่จับเหตุการณ์ใดเหตุการณ์หนึ่งแล้วเปรียบเทียบ กับความรู้ในฐานความรู้ ดังนั้นจึงไม่สามารถควบคุมพฤติกรรมทั้งหมดของแผงวงจรได้

ง. ประสิทธิภาพ ในการทดสอบวงจรโลจิก สามารถทำการทดสอบได้เพียงแบบสแตติกเท่านั้น เนื่องจากการแสดงข้อมูลอ้างอิงของระบบ เป็นแบบแสดงข้อความหรือกลุ่มตัวอักขระเท่านั้น ไม่สะดวกต่อการแสดงข้อมูลเชิงรูปภาพ (graphical) ซึ่งสามารถใช้อธิบายข้อมูลการทดสอบได้ทั้งแบบ ไดนามิค และ สแตติก ได้ชัดเจนกว่า

จ. สาเหตุข้อขัดข้องที่ยังไม่สามารถตรวจพบได้

1. สัญญาณที่เฝ้าสังเกตไม่ผิดปกติ แม้ว่าจะมีข้อขัดข้องเกิดขึ้นก็ตาม
2. สัญญาณอ้างอิง ที่เก็บอยู่ในสถานะ "1" ไม่สามารถเปรียบให้เห็นข้อแตกต่าง กับ ข้อขัดข้องที่เกิดจาก วงจรเปิดได้
3. สัญญาณอ้างอิง ที่เก็บอยู่ในสถานะ "0" ไม่สามารถเปรียบให้เห็นข้อแตกต่าง กับ ข้อขัดข้องที่เกิดจาก การลัดวงจรได้

6.3 ข้อเสนอแนะ

เนื่องจากการค้นหาข้อขัดข้องระบบนี้ อาศัยการรอคอยเหตุการณ์ที่น่าสนใจที่เกิดขึ้นบนแผงวงจร ไปกระตุ้นเครื่องมือตรวจวัดเท่านั้น ไม่สามารถควบคุมหรือกำหนดพฤติกรรมของแผงวงจรได้ ทำให้การค้นหาข้อขัดข้องต้องดำเนินการจากผลของข้อขัดข้องที่เกิดขึ้นไปสู่สาเหตุของข้อขัดข้อง ซึ่งถ้าสามารถควบคุมหรือกำหนดพฤติกรรมของแผงวงจรได้ เช่น เพิ่มเติมเครื่องมือผลิตสัญญาณทดสอบ (Test pattern generator) การค้นหาข้อขัดข้องก็สามารถดำเนินการได้ทั้งจากผลไปสู่เหตุและจากเหตุไปสู่ผล หรือทั้ง 2 แบบผสมกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการเชื่อมโยง (Interface) ระหว่างผู้ใช้กับระบบซึ่งใช้จอภาพและแป้นพิมพ์ตลอดจนการป้อนข้อมูลด้วยมือ ก็เป็นสาเหตุหนึ่งที่ทำให้การวิเคราะห์ผิดพลาดได้ ถ้าทำให้ระบบเชื่อมโยงเป็นอัตโนมัติทั้งหมด เช่น การป้อนข้อมูลของเครื่องตรวจวัดโดยทำการจับสัญญาณจากแผงวงจรอ้างอิง แล้วป้อนผ่านทางระบบเชื่อมโยงมาตรฐาน เช่น IEEE488, RS232 หรืออื่นๆ เข้าสู่ฐานความรู้โดยตรง ก็จะทำให้ได้ความรู้ที่ถูกต้องแม่นยำยิ่งขึ้น หรือในระหว่างดำเนินการค้นหาข้อขัดข้อง ถ้าให้เครื่องมือจับสัญญาณจากแผงวงจรเป้าหมายทุกจุด แล้วป้อนผ่านทางระบบเชื่อมโยงมาตรฐานดังกล่าว เข้าไปทำการเปรียบเทียบได้เลยทีเดียวก็จะลดเวลาในการค้นหาข้อขัดข้องลงได้มาก และอีกประการหนึ่งที่จะช่วยลดเวลาในการค้นหาได้อีก คือ การนำค่าการตั้งเครื่องตรวจวัด (Setup) ผ่านทางระบบเชื่อมโยงมาตรฐานเข้าไปสู่เครื่องมือตรวจวัดโดยตรง ซึ่งจะทำให้ทุกอย่างเป็นอัตโนมัติ รวดเร็ว ถูกต้องแม่นยำ บรรลุตามจุดประสงค์

จากการดำเนินงานที่ผ่านมา เป็นการออกแบบแฟ้มความรู้และออกแบบกฎที่ใช้ในการอนุมานเท่าที่จำเป็นเท่านั้น ในการดำเนินงานต่อไปสามารถที่จะพัฒนาระบบให้มีประสิทธิภาพการใช้งานที่กว้างขวางมากขึ้นได้มากมาย หลายแง่มุม เช่น การใช้ระบบจัดการฐานข้อมูลเข้ามาช่วยพัฒนาฐานความรู้, การแสดงข้อมูลอ้างอิงในรูปแบบอื่นที่ดีกว่าการใช้แผ่นผังเวลา, การเก็บข้อมูลแบบลายเซ็น เป็นต้น และควรเพิ่มการทดลองให้มีขอบเขตกว้างขวางมากยิ่งขึ้น เช่น ทดลองใช้กับแผงวงจรหลักไมโครคอมพิวเตอร์ที่ใช้ชุดตัวอุปกรณ์เฉพาะ (chip set)ของบริษัทต่างๆ เป็นต้น เพื่อเป็นการขยายฐานความรู้ให้เป็นระบบผู้เชี่ยวชาญที่มีความรู้กว้างขวางครอบคลุมมากขึ้น

6.4 ลักษณะของระบบผู้เชี่ยวชาญที่ใช้ในการค้นหาข้อขัดข้อง

ในการแก้ไขปัญหา การค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ ระบบผู้เชี่ยวชาญที่ใช้ควรมีลักษณะขั้นต้นดังนี้

1. สามารถประมวลผลเชิงสัญลักษณ์ได้
2. การอนุมานเป็นแบบเดินหน้า
3. แสดงความรู้โดยใช้กราฟหรือข่ายความหมาย และสามารถแทนลำดับของเหตุการณ์ได้
4. สามารถเชื่อมโยงกับภาษาคอมพิวเตอร์อื่นๆ เช่น ภาษาซี, แอสเซมบลี ได้
5. สามารถเชื่อมโยงกับระบบเชื่อมโยงมาตรฐาน และ เครื่องมือตรวจวัดได้เป็นอย่างดี

จากความต้องการดังกล่าวจะเห็นว่า ถึงแม้ว่า VP-Expert จะสามารถแก้ไขปัญหาการค้นหาข้อขัดข้องได้ แต่ก็ยังไม่เหมาะสมนัก เนื่องจากยังขาดคุณสมบัติบางข้ออยู่

เอกสารอ้างอิง

1. Brazile, R.P. and K.M. Swigger, "GATES An Airline Gate Assignment and Tracking Expert System", IEEE Expert, Summer 1988, pp. 33-39

2. ดร. สมนึก คีรีโต, " ระบบอิงความรู้", ไมโครคอมพิวเตอร์, ฉบับที่ 98 (กันยายน 2536) : 227-238

3. AI WEEK, "Chemical Bank Develops Foreign Trading ES." AI Week, September 15, 1988, pp. 8-9

4. พันธุ์ปิติ เปี่ยมสง่า, " โครงการคอมพิวเตอร์รุ่นที่ 5 กับระบบความรอบรู้", ไมโครคอมพิวเตอร์, ฉบับที่ 70 (พฤษภาคม 2534) : 233-236

5. RICH, KNIGHT, Artificial intelligence , 2 nd edition, McGraw-Hill,Inc. 1991

6. WOLFGRAM, DEAR, GALBRAITH, Expert System for The Technical Professional, John wiley & son, pp. 49

7. ดร. วิลาศ วูวงศ์, บุญเจริญ ศิริเนาวกุล, ระบบผู้เชี่ยวชาญ , ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ. (กันยายน 2535)

8. ดร. ก่อเกียรติ เก่งสกุล, บุญเจริญ ศิริเนาวกุล, ทฤษฎีและการประยุกต์ใช้งาน ปัญญาประดิษฐ์และระบบผู้เชี่ยวชาญ , บ.ซีเอ็ดยูเคชั่น จำกัด. 2521

9. LAURIE'RE, Problem solving and Artificial intelligence , Prentice Hall, 1990

10. สุรวุฒิ สุชินโรจน์, " สถาปัตยกรรม ระบบผู้เชี่ยวชาญ ", ไมโครคอมพิวเตอร์, ฉบับที่ 68 (มีนาคม 2534) : 205-212

11. LIPSCHUTZ, Data Structures , schaum's outline series, McGraw-Hill, 1986.

12. GenRad,Inc., GR2235/LTS & GR2225/PSP Basic Programing Manual , GenRad, 1979

13. TSUI, LSI / VLSI testability design , McGraw-Hill, 1987, pp.144

14. ABRAMOVICI, BREUER, FRIEDMAN, Digital Systems TESTING and Testable DESIGN , W.H.Freeman and Company, 1990

15. ทอม โฮแกน, รวบรวมข้อมูล PC สำหรับโปรแกรมเมอร์ , บริษัท ซีเอ็ดยูเคชั่น จำกัด, พ.ศ. 2536. หน้า 4-25

16. Anne, Dan, VP-Expert Rule-Based Expert System Development Tool , Paperback Software International

17. UFFENBECK, Microcomputers and microprocessors the 8088, 8085, and Z80 programming, interfacing, and troubleshooting , Prentice-Hall International Inc, 1985

18. KORTH, SILBERSCHATZ, Database system concepts , McGraw-Hill, 1986. pp. 29

19. ดร. วิทยา วัชระวิทยากุล, ดร. สมชาย ประสิทธ์จูตระกูล, คณิตศาสตร์ดิสครีต ประยุกต์ , บ. ซีเอ็ดยูเคชั่น จำกัด, 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 1
คู่มือการใช้งาน

สารบัญ

เนื้อหา	หน้า
การสร้างและปรับปรุงแฟ้มความรู้	126
การเรียกโปรแกรมขึ้นมาใช้งาน	128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 1
คู่มือการใช้งาน

การสร้างและปรับปรุงแฟ้มความรู้

การสร้างและปรับปรุงแฟ้มความรู้มีขั้นตอนดังนี้

1. โหลด อักขระที่ใช้สำหรับการแสดงแผนผังเวลา เข้าไปในหน่วยความจำโดยเรียกใช้โปรแกรม statext.exe ดังแสดงในรูปที่ ผ 1.1

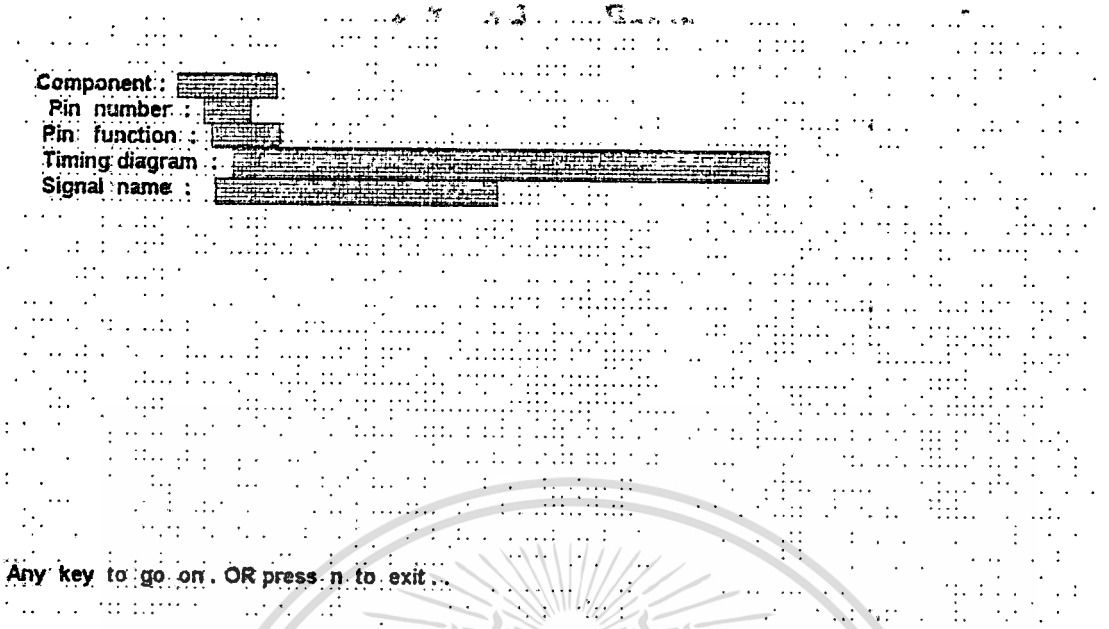
2. ทำให้แป้นพิมพ์สามารถรับข้อมูลได้ทั้ง แผนผังเวลา และอักขระธรรมดา โดยเรียกใช้ โปรแกรม tsrkbd.exe ซึ่งเป็นโปรแกรมประเภทฝังตัวอยู่ในหน่วยความจำ ดังรูปที่ ผ 1.1

```
A:\>statexte <enter>
Ready
Test by hold Alt and enter any numerical keypad from 241 - 249
A:\>tsrkbd <enter>
```

รูปที่ ผ 1.1 การโหลด อักขระที่ใช้ในการอ่าน/เขียน แผนผังเวลา

3. เรียกโปรแกรมจัดการฐานข้อมูล foxpro เพื่อเปิดแฟ้มข้อมูลของการ์ดที่ต้องการสร้างขึ้นใหม่ หรือต้องการปรับปรุงแก้ไข ซึ่งถ้าเป็นการแก้ไขเล็กๆน้อยๆ เช่น แก้ไขชื่อสัญญาณ, แก้ไขรูปร่างแผนผังเวลา, ลบหรือเพิ่มระเบียนใด(record)ระเบียนหนึ่ง ก็สามารถใช้คำสั่ง browse ได้ แต่ถ้าเป็นการเพิ่มเติมระเบียนจำนวนมากๆ หรือสร้างแฟ้มใหม่ ควรเรียกใช้โปรแกรมช่วยในการจัดเรียงข้อมูล COMPO.PRG ซึ่งจะปรากฏหน้าจอให้ใส่ข้อมูลต่างๆ ดังรูปที่ ผ 1.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ผ 1.2 แสดงหน้าจอที่ใช้ในการรับข้อมูลของโปแกรม COMPO.PRG

3.1 แป้นพิมพ์ [ ] ทำหน้าที่ในการ สลับ ไปมาระหว่างอักขระภาษาไทย ๑,๓,๕,๖,๐ กับอักขระที่ใช้สำหรับเขียนแผนผังเวลา

3.2 ช่อง pin function ใช้อักษรย่อแทนความหมายดังต่อไปนี้

o = output

i = input

ctrl = control

vcc = ไฟเลี้ยง

gnd = ดิน

c = clear, reset

s = set, preset

อื่นๆ = หมายถึงขาควบคุมทั้งสิ้น

4. สำหรับภาคการค้นหาข้อขัดข้องทุกภาคหรือทุกวงจรย่อยจะต้องใส่ชื่อแฟ้มความรู้ไว้ในฐานความรู้ ชื่อ START.DBF ซึ่งมีโครงสร้างและรายการ ดังรูปที่ 5.1 ซึ่งจะได้กล่าวมาแล้วในหัวข้อที่ 5.1

5. เมื่อเสร็จจากการใส่ข้อมูล, save แฟ้มข้อมูล และ ออกจากโปรแกรม foxpro แล้วถ้าไม่ต้องการใช้ (เขียน) แผนผังเวลาอีก ก็สามารถนำโปรแกรม tsrkbd.exe ซึ่งฝังตัวอยู่ในหน่วยความจำ ออกไปได้ โดยการกดแป้น [*] ที่ numerical keypad (ไม่ใช่แป้น [8])ซึ่งจะปรากฏข้อความตามรูปที่ ผ 1.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
A : \ >TSR already remove
A : \ >
```

รูปที่ ผ 1.3 แสดงความไม่ต้องการใช้แป้นพิมพ์เขียนแผนผังเวลาอีก

การเรียกโปรแกรมขึ้นมาใช้งาน

- ขั้นแรกเราจะเรียกโปรแกรมสำหรับโหลด อักขระแผนผังเวลาขึ้นมาทำงานก่อน ดังรูปที่ ผ 1.4

```
A: \ >statexte <enter>
Ready
Test by hold Alt and enter any numerical keypad from 241 - 249
```

รูปที่ ผ 1.4 การเรียกโปรแกรมสำหรับโหลด อักขระแผนผังเวลา

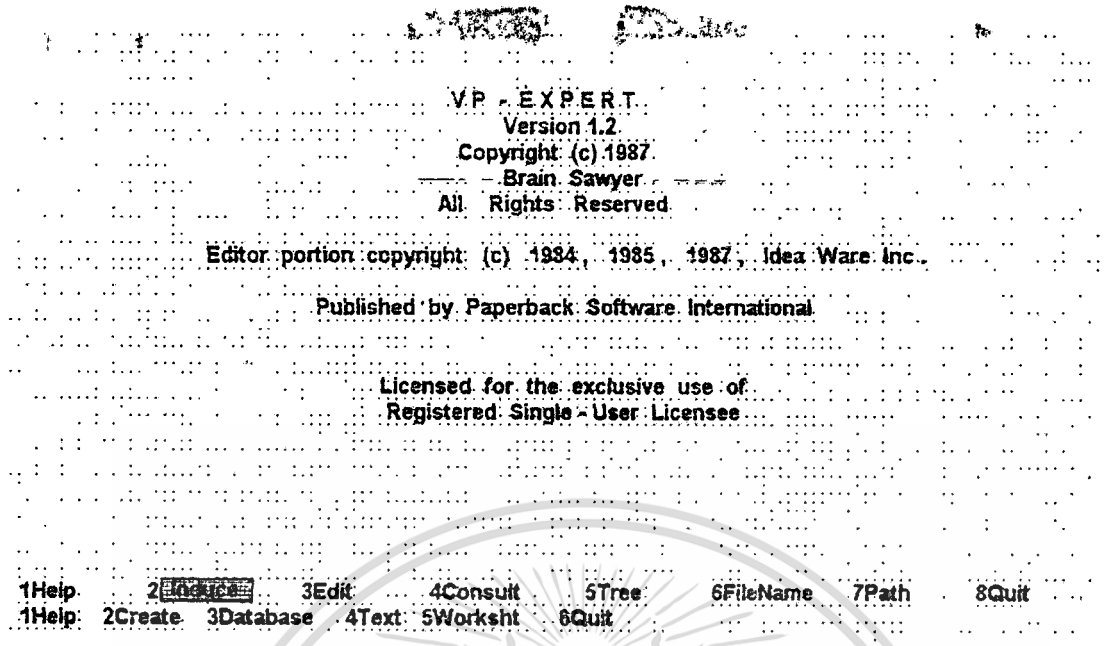
- ต่อจากนั้นจึงเรียก Expert System shell ที่ชื่อ VPX ดังรูปที่ ผ 1.5

```
A: \ >vpx <enter>
```

รูปที่ ผ 1.5 การเรียก Expert System shell

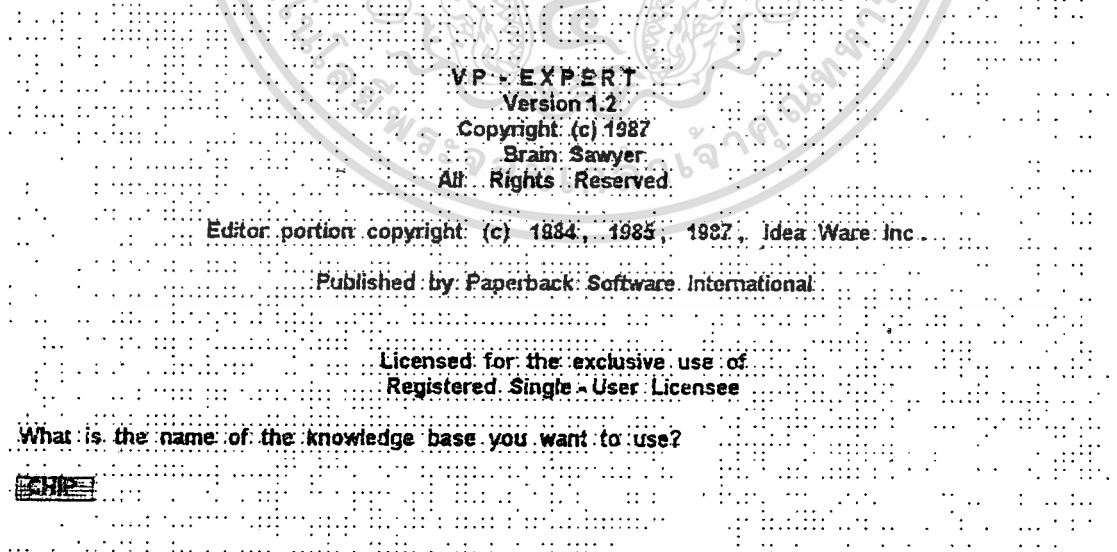
- หลังจาก VPX (Expert System shell) ถูกโหลดเข้ามาแล้วหน้าจอจะปรากฏดังรูปที่ ผ 1.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ผ 1.6 หน้าจอ VPX เมื่อถูกเรียกขึ้นมาทำงาน

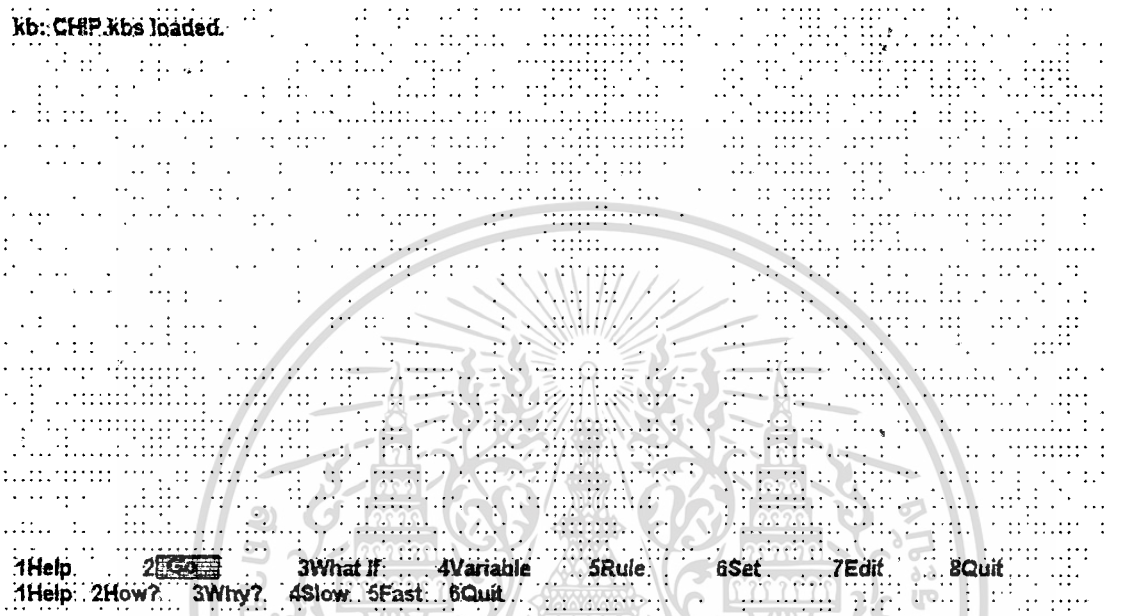
- ต่อจากนั้นให้เลือกหัวข้อการอนุมาน โดยใช้แป้นพิมพ์ ⬆ ⬇ ➡ ⬅ เลื่อนไปเลือก<consult>กด <enter> จะปรากฏหน้าจอ ดังรูปที่ ผ 1.7 ซึ่งจะมีชื่อของ แฟ้มความรู้แบบอิงกฎ ให้เป็นตัวเลือก ซึ่งในการทดลองใช้เพียงแฟ้มเดียว คือ "CHIP" ดังนั้นจึงกด <enter> เข้าไปทำงานได้เลย



รูปที่ ผ 1.7 ชื่อแฟ้มความรู้แบบอิงกฎ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หลังจากที่ แฟ้มความรู้แบบอิงกฎ "CHIP" โหลดเข้ามาแล้วจะปรากฏรายการตัวเลือกการทำงานขึ้นมาให้เลือก ซึ่งในที่นี้เลือก <go> กด <enter> เพื่อ ทำงานต่อไป ดังรูปที่ ผ 1.8



รูปที่ ผ 1.8 ตัวเลือกการทำงานเพื่อเข้าสู่การวินิจฉัย

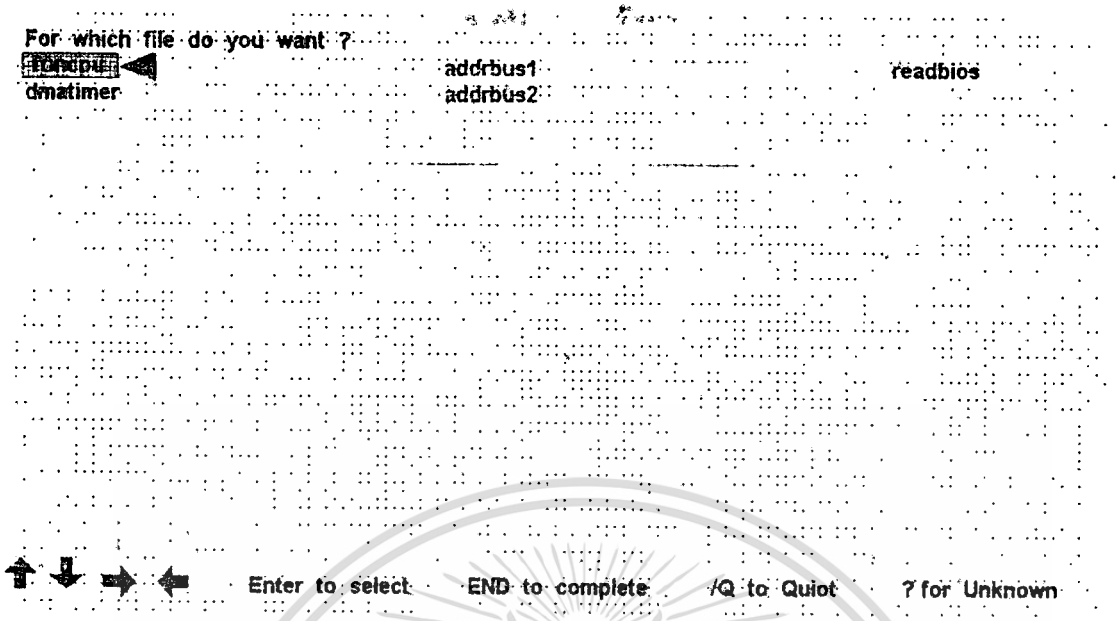
- ต่อจากนั้นจะปรากฏแฟ้มความรู้ต่างๆให้เป็นตัวเลือกดังรูปที่ ผ 1.9 ซึ่งแต่ละแฟ้มจะเป็นตัวแทนของวงจรย่อยในวงจรใหญ่ หรือแผงวงจรย่อยบนแผงวงจรใหญ่ สำหรับในการเลือก

ใช้แป้นพิมพ์ ลูกศร เช่น ↑ ↓ → ← เป็นการเคลื่อนที่

กด <enter> เมื่อเลือกได้แล้ว ซึ่งจะปรากฏเครื่องหมาย ◄ ที่ตัวเลือก

กด <End> เป็นการตกลงใจ หรือ กด <Delete> เมื่อต้องการเปลี่ยนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

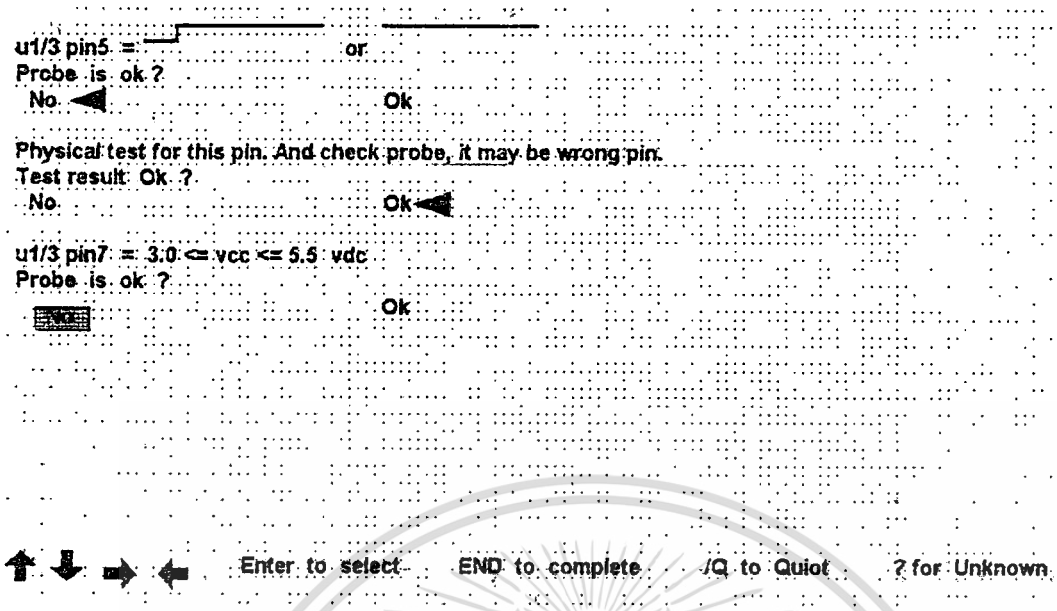


รูปที่ ผ 1.9 แฟ้มความรู้ต่างๆของวงจรย่อย

- หลังจากนั้นจะเป็นการติดต่อระหว่างกลไกการอนุมานกับผู้ใช้ ดังรูปที่ ผ 1.10โดย inference engine จะแสดงข้อความต่างๆเช่น บอกชื่อ , ขาของตัวอุปกรณ์, รูปร่างของสัญญาณเปรียบเทียบ หรือบอกให้กระทำอย่างใดอย่างหนึ่ง เช่น ตรวจ, วัด ว่าขาของอุปกรณ์หลวมหลุดหรือไม่ เป็นต้น และ inference engine จะถามว่า "Probe is ok ? " ซึ่งหมายถึงว่าผลของการตรวจวัดให้ผลเป็นอย่างไร ? ได้ผลตรงกับข้อมูลในแฟ้มความรู้หรือไม่ ? หรือมีการหลวมหลุด, ลัดวงจรหรือไม่ ?

แล้วผู้ใช้จะเป็นผู้เลือกคำตอบโดยการเลือก และการตัดสินใจ คงใช้วิธีเช่นเดิม ดังที่ได้กล่าวมาแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ผ 1.10 การติดต่อระหว่างกลไกการอนุมานกับผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 2

แฟ้มความรู้ CHIP.KBS

```
RUNTIME;
ACTIONS
MENU example,ALL,start,card
FIND example
WHILEKNOWN card
   GET ALL,start,card
END
dbase = (example)
SHIP @@@@@,dbase
CALL transf
!---------- Load database file ----------
 x = 1
 DISPLAY "Loading knowledge base."
 WHILEKNOWN node
  GET ALL,$$$$$,ALL
  xnode[x] = (node)
  xpin[x] = (pin)
  funct[x] = (func)
  tdiagm[x] = (t_diagm)
  signal[x] = (signame)
  t[x] = 0
  x = (x+1)
 END
 bottom = (RECORD_NUM)
 CLOSE $$$$$
 CALL sink
 CLS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
!---------- Initialize ----------
  x = 1
  tx = 0
  inloop = go
  next_dbf = NEXT
  search = (signal[1])
  pin_func = (funct[1])
!---------- Inference loop ----------
  WHILEKNOWN inloop
    y = (x+1)
    RESET cond
    RESET testacq
    RESET disp
    RESET scannode        ! Prevent error during scan common node
    FIND disp
    x = (x+1)
    RESET full
    FIND full
  END;
!========== RULES ==========

RULE 10a
IF   search = (signal[x]) AND
     pin_func = (funct[x])
THEN cond = same1;

RULE 10b
IF   search = (xnode[x]) AND
     pin_func = (funct[x])
THEN cond = same2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RULE 20a
IF   cond = same1 OR
     cond = same2 AND          ! (same1 OR same2) AND next_dbf
     next_dbf = (signal[x])
THEN disp = nextfile
     DISPLAY "For next diagnosis please...{tdiagm[x]}"
     RESET inloop;

RULE 20b
IF   cond = same1 OR
     cond = same2
THEN disp = meet
     DISPLAY "{xnode[x]} pin{xpin[x]} = {tdiagm[x]}"
     RESET fault
     RESET feedback
     FIND feedback
     FIND fault;

RULE 30
IF   signal[x] = ^^^
THEN feedback = problem
     RESET trackback
     DISPLAY "It is a loop problem. This version can't solve it."
     DISPLAY "You should open or short {xnode[x]}pin{xpin[x]} to the logic state as you want."
     DISPLAY "And then create or use other data base."
     DISPLAY "Any key to go.~"
     fault = feedback_problem
     FIND trackback;

RULE 31a
IF   pin_func = o AND
     testacq = Ok
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    signal[x] = ^^^^          ! mark this record's been visited.
    tx = ((tx)+1)
    t[x] = (tx)
    RESET is_vcc_gnd
    RESET messg1
    RESET any_shorted
    RESET scannode
    FIND is_vcc_gnd
    FIND messg1
    RESET phytst
    FIND any_shorted
    search = (signote)        ! scan every common node except out put pin
    pin_func = o              ! set pin_func = o for exception
    y = (x)                   ! store old x
    FIND scannode
    x = (y)                   ! retrieve old x
    z = (x);                  ! set pointer before goto output pin of next device

RULE 40
IF   fault = no_output OR
     fault = ctrl_i
THEN scannode = expand
     x = 1
     scan = go
     WHILEKNOWN scan
       RESET fanout
       FIND fanout
       x = (x+1)
       RESET full_scan
       FIND full_scan
     END
     x = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RULE 41
IF   fault = ctrl_i AND    ! Inherited.. Probe = No. from rule 34b
     signote = gnd OR
     signote = vcc
THEN is_vcc_gnd = none
     messg1 = cancel
     any_shorted = cancel
     scannode = cancel
     DISPLAY "{xnode[x]} pin{xpin[x]} or circuit.May be open/loose."
     RESET inloop;

RULE 42a
IF   xnode[x] = (xnode[y])
THEN trackback = go
     search = (xnode[x])
     pin_func = (funct[y]);

RULE 42b
IF   xnode[x] <> (xnode[y])
THEN trackback = no_go
     DISPLAY "{xnode[x]} is out of order."
     RESET inloop;

RULE 50
IF   scannode = expand AND      ! For ctrl_i
     search = (signal[x]) AND        ! scan every common node except output pin
     pin_func <> (funct[x])
THEN fanout = jointly           ! For o (output)
     signal[x] = ^^^^           ! scan every common node, no exception.
     tx = ((tx)+1)
     t[x] = (tx)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    RESET com_node
    DISPLAY "{xnode[x]} pin{xpin[x]} = {tdiagm[x]}"
    RESET testacq
    FIND com_node;

RULE 60a
IF   fanout = jointly AND
    testacq = No
THEN com_node = no_one
    RESET any_short
    RESET phytst
    DISPLAY "Physical test for this pin.And check probe, it may be wrong pin."
    FIND any_short;

RULE 60b
IF   fanout =jointly AND
    testacq = Ok
THEN com_node = some_one
    DISPLAY "Open circuit {xnode[x]} pin{xpin[x]} - {xnode[y]} pin{xpin[y]}."
    RESET scan
    RESET inloop;

RULE 43a
IF   phytst = No AND
    fault = no_output
THEN any_shorted = _fault CNF 50
    DISPLAY " Find out { #any_shorted}."
    DISPLAY "Test {xnode[x]} pin{xpin[x]} for shorted or loose.And reprobe again."
    scannode = cancel
    trackback = cancel
    RESET inloop;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RULE 43b
IF   phytst = No AND
     fault = ctrl_i
THEN any_shorted = _fault CNF 50
     DISPLAY " Find out { #any_shorted}."
     DISPLAY "Test {xnode[x]} pin{xpin[x]} for shorted or loose.And reprobe again."
     scannode = cancel
     RESET inioop;

RULE 70
IF   phytst = No AND
     com_node = no_one
THEN any_short = _fault CNF 50
    DISPLAY " Find out { #any_short}."
    DISPLAY "Test {xnode[x]} pin{xpin[x]} for shorted or loosed.And reprobe again."
     RESET scan
     RESET inloop;

RULE ff
IF   x > (bottom)
THEN full = over
     DISPLAY "Bottom of data base.  Any key to go...~"
     RESET inloop;

RULE ffa
IF   x >= (bottom)
THEN full_scan = over
     RESET scan;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
!.....Dummy RULE use for display message.....

RULE dummy1
IF inloop = go
THEN messg1 = message
    DISPLAY "Physical test for this pin.And check probe, it may be wrong pin.";

!========================= ASK ========================
ASK testacq : "Probe is ok ?";
CHOICES testacq : No,Ok;

ASK phytst : "Test result Ok ?";
CHOICES phytst : No,Ok;

ASK example: "For which file do you want ?";

!-------------------------------------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 3

แฟ้ม START.DBF

CARD	DESCRIPT
runcpu	Test signal RESET, CLK88 and READY
addrbus1	Test address bus, signal AENBRD and ALE open.
readbios	Test CPU can read BIOS normaly or not.
dmatimer	Test DMA for begining.
addrbus2	Test address bus, signal AENBRD and ALE open - short.

สรุป ชื่อ-หน้าที่ สัญญาณบนแผงวงจรหลัก PC/XT

ชื่อสัญญาณ	หน้าที่
A0-A19	Address Line 0-19
AD0-AD7	Buffered Address/Data 0-7
ADSTB	Address Strobe
ADDR SEL	Address select
AENBRD	Address Enable Board
AEN	Address Enable
BALE	Buffered Address Latch Enable
BA8-BA19	Buffered Address Line 8-19
CAS0-CAS3	Column Address Strobe 0-3
CLK	clock
CLK88	clock88
CS2-CS7	Chip Select Line 2-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D0-D7	Data Line 0-7
DACK	DMA Acknowledge
DACK & TC	DMA Acknowledge & Teminal Count
DCLK	Data Clock
$\overline{DEN}$	Data Enable
DMA AEN	DMA Address Enable
$\overline{DMACS}$	DMA Chip Select
$\overline{DMAWAIT}$	DMA Wait
DRQ	DMA Request
DT/$\overline{R}$	Data Transmit/Receiver
EN I/O CK	Enable I/O Check
EN I/O clk	Enable I/O clock
ENB RAM PCK	Enable RAM Peripheral check
HOLDA	Hold Acknowledge
INT	Interrupt
INTA	Interrupt Acknowledge
INIT	Initialize
I/O CHCK	I/O channel check
I/O CH CLK	I/O channel clock
I/O CH RDY	I/O channel Ready
QS0, QS1	Queue status 0, 1
RA0-RA3	Row Address Line 0-3
RAM ADDR SEL	RAM Address Select
$\overline{RDY}$/WAIT	Ready/Wait
Refresh	Refresh gate
$\overline{RESETDRV}$	Reset Drive
ROM ADDR	ROM Address Select
RQ/GT	Request/Grant
$\overline{S0}$-$\overline{S2}$	Status Line 0-2
Serial	Serial Data
SLCT	Select

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SP/EN	Slave Program/Enable Buffer
SPKR DATA	Speaker Data
SD0-SD15	System Data 0-15
SA0-SA19	System Address 0-19
SMEMR	System Memory Read
SMEMW	System Memory Write
Strobe	Strobe
TC	Terminal Count
TIM 2 Gate	Timer 2 Gate Speaker
IOR	I/) Read
IOW	I/O Write
IRQ	Interrupt Request
IRQ EN	Interrupt Request Enable
$\overline{\text{LOCK}}$	Lock
LA17-LA23	Unlatched Address 17-23
MA0-MA7	Memory Address 0-7
MD0-MD7	Memory Data 0-7
MDP	Memory Data Parity
$\overline{\text{MEMW}}$	Memory Write
$\overline{\text{MEMR}}$	Memory Read
MASTER	Master
MRQ DMA	Memory Request DMA
NMI	Non-Maskable Interrupt
NP INSTL	Numerical Processor Installation Switch
NP NPI	Numerical Processor Numerical Processor Interrupt
OSC	Oscillator
PB6 PB7	Port B6&7 of 8255
PCLK	Peripheral clock
PWR GOOD	Power Good
WE	Write Enable
WRT DMA PG REG	Write DMA Page Register

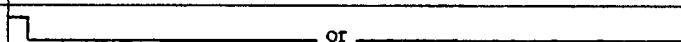
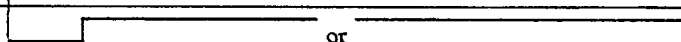
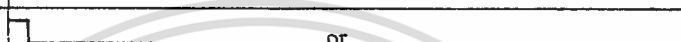
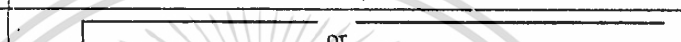
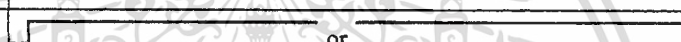
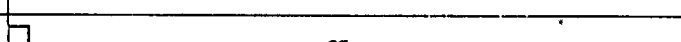
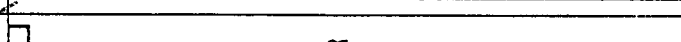
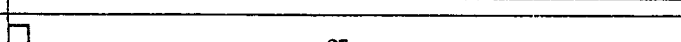
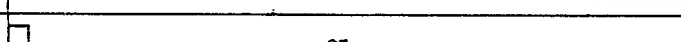
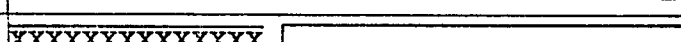
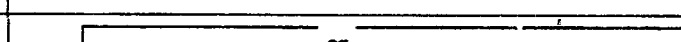
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WRT NMI REG	Write Non-Maskable Intr Register
XA0-XA12	Buffered Address 0-12
XD0-XD7	Buffered Data 0-7
$\overline{\text{XIOR}}$	Buffered I/O READ
$\overline{\text{XIOW}}$	Buffered I/O Write
$\overline{\text{XMEMR}}$	Buffered Memory Read
$\overline{\text{XMEMW}}$	Buffered Memory Write

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่4

แฟ้มความรู้ RUNCPU.DBF

NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
u3	21	ctrl	5		reset
u3	19	ctrl	8		clk88
u3	22	ctrl	8		ready
u3	...	ctrl	9	select ADDRBUS1 knownledge file.	NEXT
u4	19	ctrl	8		clk88
u4	21	ctrl	5		reset
u4	22	ctrl	-8		ready
u8	2	ctrl	8		clk88
u21	15	i	9		clk88
u21	13	i	9		_resetdrv
u1/2	8	o	0		clk88
u1/2	2	o	0		pclk
u1/2	12	o	0	clock 14.3 MHz	osc
u1/2	13	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u1/2	1	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u1/2	16	i	9	clock 14.3 MHz	oscxtal
u1/2	17	i	9	clock 14.3 MHz	oscxtal
R1	0	o	0	clock 14.3 MHz	oscxtal
R1	1	i	9	0 <= gnd <= 0.3 vdc	gnd
R2	0	o	0	clock 14.3 MHz	oscxtal
R2	1	i	9	0 <= gnd <= 0.3 vdc	gnd
xtal		o	0	clock 14.3 MHz	oscxtal
u18/6	13	i	9		reset
u36/1	13	ctrl	5		reset
u37	35	ctrl	5		reset
u1/1	10	o	0		reset
u1/1	11	i	9		vcc
u1/3	5	o	0		ready
u1/3	7	ctrl	8	3.0 <= vcc <= 5.5 vdc	vcc

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u1/3	4	i	9	h state	_dmawait
u65	9	o	0	or	_rdy/wait
u65	10	ctrl	6	h state	i/o_ch_rdy
u65	13	ctrl	5	1 state	u83 11u65.13u98
slot	a10	o	9	h state	i/o_ch_rdy
u98/4	12	i	9	1 state	u83 11u65 13u98
u83/3a	11	o	0	1 state	u83.11u65.13u98
u83/3a	...	i	9	change U65(74S74) & U83(74LS175)	NEXT
u83/3b	12	i	9	or	_rdy/wait
u83/2	6	o	0	h state	_dmawait
u83/2	1	ctrl	5	or	_resetdrv
u83/2	9	ctrl	8		clk
u83/2	5	i	9	1 state	aenbrd
u83/1	2	o	0	1 state	aenbrd
u83/1	3	o	0	h state	_aen
u83/1	4	i	9	1 state	holda
u51	6	ctrl		h state	_aen
u8	15	ctrl	8	h state	_aen
u82/2	13	ctrl	5	or	_resetdrv
u18/5	11	i	9	or	_resetdrv
u27	1	ctrl	5	or	_resetdrv
u18/6.	12	o	0	or	_resetdrv
u18/6.	13	i	9	or	reset_reset
u1.	10	o	0	check open circuit this signal.	reset_reset
u21.	5	o	0		clk
u21.	9	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u21.	15	i	9		clk88_clk88
u1.	8	i	9	check open circuit this signal.	clk_clk
u82/1	5	o	0	1 state	holda
u82/1	1	ctrl	5	1 state	hrqdma
u36/2	10	o	0	1 state	hrqdma
u36/2	...	i	9	select DMA knownledge file.	NEXT
u66	5	i	9	1 state	hrqdma

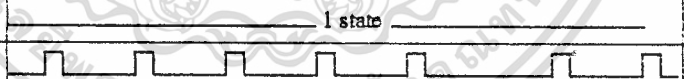
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 5

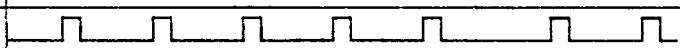
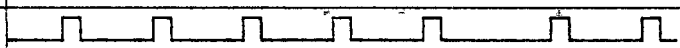
แฟ้มความรู้ ADDRBUS1.DBF

NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
u8.	6	ctrl	8		dummy1
u8.	...	i	9	select ADDRBUS2 knownledge file.	NEXT
u2	.	o	0	dummy node please select probe = No.	dummy1
u2	19	ctrl	8		_den
u2	.	i	9	dummy node please select probe = No.	dummy2
ADDR	.	o	0	dummy node please select probe = No.	dummy2
ADDR	.	i	9	A0-A7 check all of them.	xa0_xa7
ADDR	.	i	9	A8-A11 check all of them.	xa8_xa11
ADDR	.	i	9	A12-A15 check all of them.	xa12_xa15
ADDR	.	i	9	A16-A19 check all of them.	xa16_xa19
ADDR	...	i	9	select ADDRBUS2.DBF knownledge file.	NEXT
u18	8	o	0		_den
u18	9	i	0		den
u8/1	4	o	0		den
u8/1	15	ctrl	8	h state	_aen
u8/1	6	ctrl	8	1 state	aenbrd
u8/1	1	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u8/1	19	i	9		_s0
u8/1	3	i	9		_s1
u8/1	18	i	9	h state	_s2
u51	6	ctrl	8	h state	_aen
u66	4	i	9		_s0
u66	2	i	9		_s1
u3/1	26	o	0		_s0
u3/1	27	o	0		_s1
u3/1	28	o	0	h state	_s2
u3/1	...	i	9	change U3(8088) and test again.	NEXT
u83/1	3	o	0	h state	_aen

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u83/1	.	i	9	return to select RUNCPU and test again.	NEXT
u10	_oput	o	0	XA0-XA7 can change state from 1-0; 0-1.	xa0_xa7
u10	1, 19	ctrl	8	1 state	aenbrd
u10	_iput	i	9	A0-A7 can change state from 1-0; 0-1 .	a0_a7
u16/1	_oput	o	0	XA8-XA11 can change state from 1-0; 0-1.	xa8_xa11
u16/1	1, 19	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u16/1	_iput	i	9	A8-A11 can change state from 1-0; 0-1 .	a8_a11
u16/2	_oput	o	0	XA12-XA15 can change state from 1-0; 0-1	xa12_xa15
u16/2	_iput	i	9	A12-A15 can change state from 1-0; 0-1.	a12_a15
u11	_oput	o	0	XA16_XA19 can change state from 1-0; 0-1	xa16_xa19
u11	1	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u11	_iput	i	9	A16-A19 can change state from 1-0; 0-1.	a16_a19
u5	_oput	o	0	A0-A7 can change state from 1-0; 0-1.	a0_a7
u5	1	ctrl	8	1 state	aenbrd
u5	11	ctrl	8		ale
u5	_iput	i	9	AD0-AD7 can change state from 1-0; 0-1.	ad0_ad7
slot A	30-24	i	9	A0-A7 can change state from 1-0; 0-1.	a0_a7
u6	_oput	o	0	A8-A11 can change state from 1-0; 0-1.	a8_a11
u6	1	ctrl	8	1 state	aenbrd
u6	_iput	i	9	AA8-AA11 can change state from 1-0; 0-1.	aa8_aa11
slot A	23-20	i	9	A8-A11 can change state from 1-0; 0-1.	a8_a11
u7/1	_oput	o	0	A12-A15 can change state form 1-0; 0-1.	a12_a15
u7/1	1	ctrl	8	1 state	aenbrd
u7/1	11	ctrl	8		ale
u7/1	_iput	i	9	AA12-AA15 can change state from 1-0; 0-1	aa12_aa15
slot A	19-16	i	9	A12-A15 can change state form 1-0; 0-1.	a12_a15
u7/2	_oput	o	0	A16-A19 can change state form 1-0; 0-1.	a16_a19
u7/2	_iput	i	9	AA16-AA19 can change state from 1-0; 0-1	aa16_aa19
slot A	15-12	i	9	A16-A19 can change state form 1-0; 0-1.	a16_a19
u3/1	16-9	o	0	AD0-AD7 can change state from 1-0: 0-1.	ad0_ad7
u3/1	...	i	9	change U3(8088) and test again.	NEXT
u3/2	8-5	o	0	AA8-AA11 can change state from 1-0; 0-1	aa8_aa11
u3/2	...	i	9	change U3(8088) and test again.	NEXT
u3/3	2-4,.	o	0	AA12-AA15 can change state from 1-0; 0-1	aa12_aa15
u3/3	...	i	9	change U3(8088) and test again.	NEXT

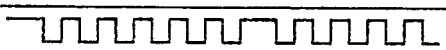
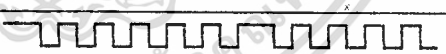
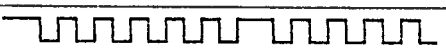
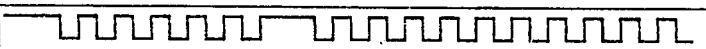
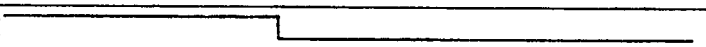
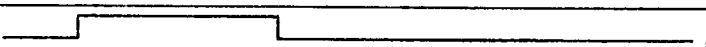
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u3/4	38-35	o	0	AA16-AA19 can change state from 1-0; 0-1	aa16_aa19
u3/4	...	i	9	change U3(8088) and test again.	NEXT
slot B	28	i	9		ale
u8/2	5	o	0		ale
u8/2	5	i	9	change U8(8288) and test again.	NEXT
u83/2	2	o	0	1 state	aenbrd
u83/2	...	i	9	return to select RUNCPU and test again.	NEXT

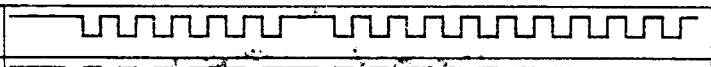
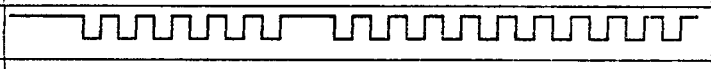
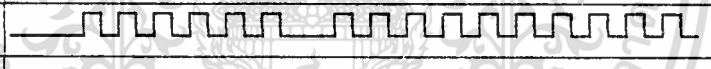
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 6

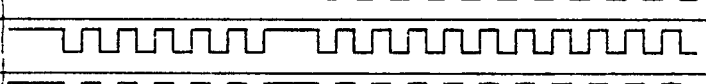
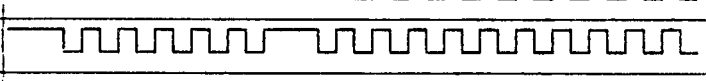
แฟ้มความรู้ READBIOS.DBF

NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
u3	9-16	i	9	not stable AND u3.26 = 	ad7-ad0
u3	...	i	9	select ADDBUS2 knownledge file.	NEXT
u2	_oput	o	0	not stable AND u3.26 =	ad7-ad0
u2	1	ctrl	8		dt/_r
u2	_iput	i	9	not stable AND u3.26 =	d7-d0
u8	4	o	0		dt/_r
u8	18	ctrl	8	h state	_s2
u8	3	ctrl	8		_s1
u8	19	ctrl	8		_s0
u3/1	28	o	0	h state	_s2
u3/1	27	o	0		_s1
u3/1	26	o	0		_s0
u3/1	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
u13	1	ctrl	8		_xmemr
u13	_iput	i	9	not stable AND u3.26 =	d7-d0
slots	A2-A9	i	9	not stable AND u3.26 = 	d7-d0
u14	_oput	o	0	not stable AND u3.26 =	d7-d0
u14	1	ctrl	8		u17.8u14.1
u14	_iput	i	9	not stable AND u3.26 =	xd7-xd0
u35	_data	o	0	not stable AND u3.26 = 	xd7-xd0
u35	20	ctrl	8		_xmemr
u35	22	ctrl	8		_cs7
u35	10-3	i	9	XA3-XA7 can change state from 1-0; 0-1.	xa0-xa7
u35	25	i	9		xa8
u35	24	i	9		xa9
u35	23,21	i	9	XA11-XA10 can change state from 1-0; 0-1	xa10-xa11
u35	2	i	9	1 state	xa12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u23	7	o	0		_cs7
u23	4	ctrl	8		_xmemr
u23	5	ctrl	8	1 state	u12.8u23.5u24.5
u23	6	ctrl	8	3.0 <= vcc <= 5.5 vdc	vcc
u23	3,2,1	i	9	h state	xa13-xa15
u16	_oput	o	0	h state	xa13-xa15
u16	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
u17	8	o	0		u17.8u14.1
u17	9	i	9	1 state	u17.9u17.6
u17	10	i	9		romaddrsel
u17	4	i	9		xa8
u17	5	i	9		xa9
u51	5	ctrl	8		xa8
u51	4	ctrl	8		xa9
uxx	...	o	0	return to select ADDRBUS1 knowledge file	xa8
uxx	...	o	0	return to select ADDRBUS1 knowledge file	xa9
u17/2	6	o	0	1 state	u17.9u17.6
u17/2	3	i	9	1 state	_xior
u24/2	8	o	0		romaddrsel
u24/2	5	i	9	1 state	u12.8u23.5u24.5
u24/2	6	i	9		_xmemr
u12/2	4	o	0	1 state	u12.8u23.5u24.5
u12/2	10	i	9	h state	xa18
u12/2	13	i	9	h state	xa17
u12/2	12	i	9	h state	xa16
u12/2	9	i	9	h state	xa19
u11a	12	o	0	h state	xa19
u11a	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
u11b	14	o	0	h state	xa18
u11b	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
u11c	16	o	0	h state	xa17
u11c	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
u11d	18	o	0	h state	xa16
u11d	...	i	9	return to select ADDRBUS1 knowledge file	NEXT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u52	11	ctrl	8		_xmemr
u96	1	i	9		_xmemr
u67	3	i	9		_xmemr
u22/1	17	o	0		_xmemr
u22/1	19	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u22/1	3	i	9		_memr
u8/2	7	o	0		_memr
u8/2	...	i	9	change u8(8288) and test again.	NEXT
u36	1	i	9	h state	_xior
u84	13	i	9	h state	_xior
u48	3	i	9	h state	_xior
u37	5	i	9	h state	_xior
u22/2	16	i	9	h state	_xior
u22/2	4	i	9	h state	_ior
u8/3	13	o	0	h state	_ior
u8/3	...	i	9	change u8(8288) and test again.	NEXT

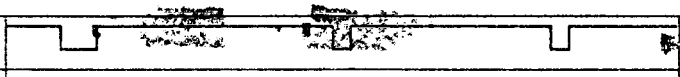
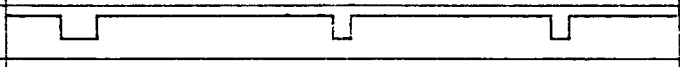
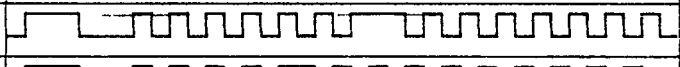
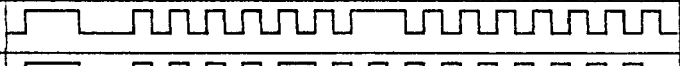
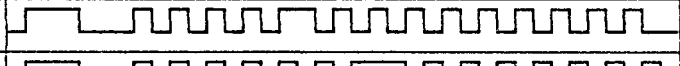
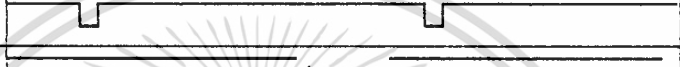
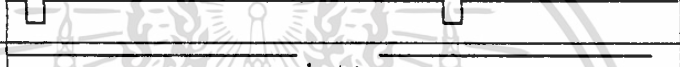
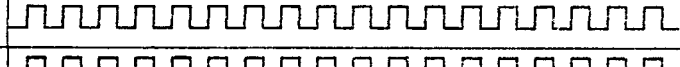
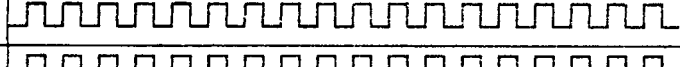
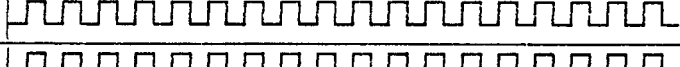
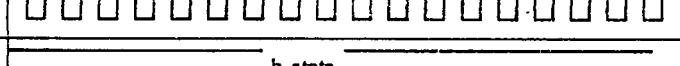
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 7

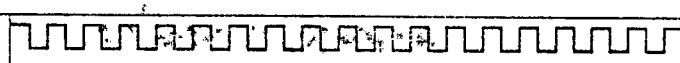
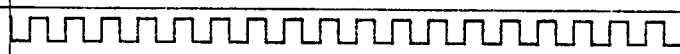
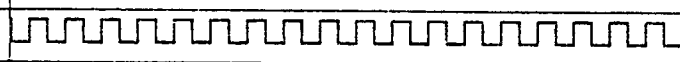
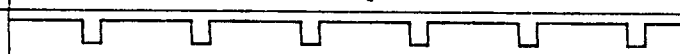
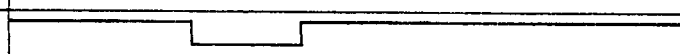
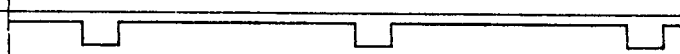
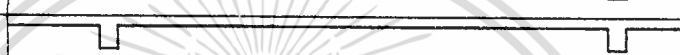
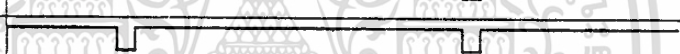
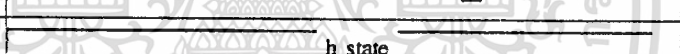
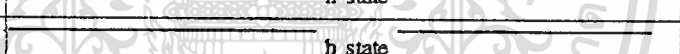
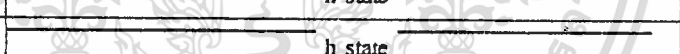
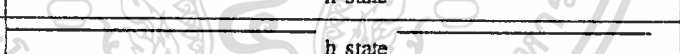
แฟ้มความรู้ DMATIMER.DBF

NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
u83	3	o	0		_aen
u83	2	o	0		aenbrd
u83	4	i	9		u82.5u83.4
u82	5	o	0		u82.5u83.4
u82	6	o	0		u82.6u82.4
u82	1	ctrl	5		hrqdma
u82.	4	o	6		u82.6u82.4
u36	10	o	0		hrqdma
u36	13	ctrl	5	h state	reset
u36	12	ctrl	8		_clk
u36	6	ctrl	8		rdytodma
u36	11	ctrl	8		_dmacs
u36	1	ctrl	8	h state	_xior
u36	2	ctrl	8		_xiow
u36	3	ctrl	8		_xmemr
u36	4	ctrl	8		_xmemw
u36	7	i	9		holda
u36	19	i	9	h state	drq0
u1	10	o	0	h state	reset
u1	...	i	9	return to select RUNCPU knoeledge file.	NEXT
u18	6	o	0		_clk
u18	...	i	9	return to select RUNCPU knowledge file.	NEXT
u82	5	o	0		holda
u82	2	i	9		u83.15u82.2
u83	15	o	0		u83.15u82.2
u83	13	i	9		u67.6u83.13
u67	6	o	0		u67.6u83.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u67	5	i	9		u66.6u67.5
u66	6	o	0		u66.6u67.5
u66	1	i	9	h state	_lock
u66	4	i	9		_s0
u66	2	i	9		_s1
u8	19	i	9		_s0
u8	3	i	9		_s1
u3	26	o	0		_s0
u3	27	o	0		_s1
u3	29	o	0	h state	_lock
u98	11	o	0		rdytodma
u98	13	i	9	h state	u65.8u98.13
u98	12	i	9		u83.11u98.12
u65	5	o	0	h state	drq0
u65	1	ctrl	5	h state	_dack0brd
u65	4	ctrl	6	3.0 <= VCC <= 5.5 VDC	vcc
u65	3	ctrl	8		out1
u65	2	i	9	3.0 <= vcc <= 5.5 vdc	vcc
u36	25	o	0	h state	_dack0brd
u36	...	i	9	change U36(8237) and test again.	NEXT
u49	13	o	0		out1
U49	21	ctrl	8		_tccs
u49	11	ctrl	8	3.0 <= vcc <= 5.5 vdc	vcc
u49	14	ctrl	8	3.0 <= vcc <= 5.5 vdc	vcc
u49	16	ctrl	8	1 state	timzgatespk
u49	19	i	9	XA0 can change state from 0-1; 1-0.	xa0
u49	20	i	9	XA1 can change state from 0-1; 1-0.	xa1
u49	_8-1	i	9	XD0-XD7 can change state 0-1; 1-0.	xd0-xd7
u49	9	i	9		clk012
u49	15,16	i	9		clk012
u49		i	9		clk012
u27	10	o	0		clk012
u27	11	o	0		u27.11u27.12
u27	1	ctrl	5	h state	_resetdrv

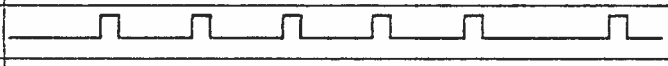
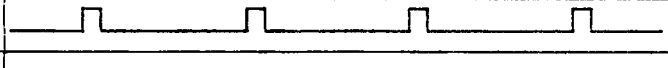
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u27	12	ctrl	8		u27.11u27.12
u27	9	i	9		pclk
u1	2	o	0		pclk
u1	...	i	9	return to select RUNCPU knowledge file.	NEXT
u51	15	o	0		_dmacs
u51	13	o	0		_t/ccs
u51	6	ctrl	8		_den
u51	4	ctrl	8	XA9 can change state from 0-1; 1-0	xa9
u51	5	ctrl	8	XA8 can change state from 0-1; 1-0	xa8
u51	1,2,3	i	9	XA5-XA7 can change state form 0-1; 1-0.	xa5-xa7
u48	2	ctrl	8		_xiow
u84	1	i	9		_xiow
u84	3	ctrl	8		_xiow
u37	36	ctrl	8		_xmemw
u50	4, 10	ctrl	8		_xiow
u50	1	ctrl	8		_xmemw
u96	2	ctrl	8		_xmemw
u67	1	i	9	 h state	_dack0brd
u84	3	i	9	 h state	_dack0brd
u99	2	i	9	 h state	_dack0brd
u11	15	i	9	 h state	_dack0brd
.....	...	o	0	check this signal again.	xa0
.....	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
.....	...	o	0	check this signal again.	xa1
.....	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
.....	...	o	0	check this signal again.	xa8
.....	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
.....	...	o	0	check this signal again.	xa9
.....	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
.....	...	o	0	check this signal again.	xa5-xa7
.....	...	i	9	return to select ADDRBUS1 knowledge file	NEXT
.....	...	o	0	check these signals again.	xd0-xd7
.....	...	i	9	return to select READBIOS knowledge file	NEXT

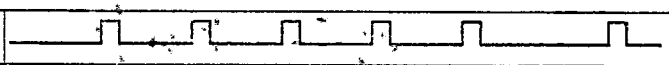
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 8

แฟ้มความรู้ ADDRBUS2.DBF

NODE	PIN	FUNC	G	T_DIAGM	SIGNAME
ADDR	.	i	9	A0-A7 check all of them.	xa0_xa7
ADDR	.	i	9	A8-A11 check all of them.	xa8_xa11
ADDR	.	i	9	A12-A15 check all of them.	xa12_xa15
ADDR	.	i	9	A16-A19 check all of them.	xa16_xa19
ADDR	...	i	9	select next knownledge file.	NEXT
u10	_oput	o	0	XA0-XA7 can change state from 1-0; 0-1.	xa0_xa7
u10	1, 19	ctrl	8		aenbrd
u10	_iput	i	9	A0-A7 can change state from 1-0; 0-1 .	a0_a7
u16/1	_oput	o	0	XA8-XA11 can change state from 1-0; 0-1.	xa8_xa11
u16/1	1, 19	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u16/1	_iput	i	9	A8-A11 can change state from 1-0; 0-1 .	a8_a11
u16/2	_oput	o	0	XA12-XA15 can change state from 1-0; 0-1	xa12_xa15
u16/2	_iput	i	9	A12-A15 can change state from 1-0; 0-1.	a12_a15
u11	_oput	o	0	XA16_XA19 can change state from 1-0; 0-1	xa16_xa19
u11	1	ctrl	8	0 <= gnd <= 0.3 vdc	gnd
u11	_iput	i	9	A16-A19 can change state from 1-0; 0-1.	a16_a19
u5	_oput	o	0	A0-A7 can change state from 1-0; 0-1.	a0_a7
u5	1	ctrl	8		aenbrd
u5	11	ctrl	8		ale
u5	_iput	i	9	AD0-AD7 can change state from 1-0; 0-1.	ad0_ad7
slot A	30-24	i	9	A0-A7 can change state from 1-0; 0-1.	a0_a7
u6	_oput	o	0	A8-A11 can change state from 1-0; 0-1.	a8_a11
u6	1	ctrl	8		aenbrd
u6	_iput	i	9	AA8-AA11 can change state from 1-0; 0-1.	aa8_aa11
slot A	23-20	i	9	A8-A11 can change state from 1-0; 0-1.	a8_a11
u7/1	_oput	o	0	A12-A15 can change state form 1-0; 0-1.	a12_a15
u7/1	1	ctrl	8		aenbrd

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

u7/1	11	ctrl	8		ale
u7/1	_iput	i	9	AA12-AA15 can change state from 1-0; 0-1	aa12_aa15
slot A	19-16	i	9	A12-A15 can change state form 1-0; 0-1.	a12_a15
u7/2	_oput	o	0	A16-A19 can change state form 1-0; 0-1.	a16_a19
u7/2	_iput	i	9	AA16-AA19 can change state from 1-0; 0-1	aa16_aa19
slot A	15-12	i	9	A16-A19 can change state form 1-0; 0-1.	a16_a19
u3/1	16-9	o	0	AD0-AD7 can change state from 1-0; 0-1.	ad0_ad7
u3/1	...	i	9	change U3(8088) and test again.	NEXT
u3/2	8-5	o	0	AA8-AA11 can change state from 1-0; 0-1	aa8_aa11
u3/2	...	i	9	change U3(8088) and test again.	NEXT
u3/3	2-4,.	o	0	AA12-AA15 can change state from 1-0; 0-1	aa12_aa15
u3/3	...	i	9	change U3(8088) and test again.	NEXT
u3/4	38-35	o	0	AA16-AA19 can change state from 1-0; 0-1	aa16_aa19
u3/4	...	i	9	change U3(8088) and test again.	NEXT
slot B	28	i	9		ale
u8/2	5	o	0		ale
u8/2	5	i	9	change U8(8288) and test again.	NEXT
u83/2	2	o	0		aenbrd
u83/2	...	i	9	return to select RUNCPU and test again.	NEXT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 9

โปรแกรม COMPO.PRG

```
clear
set talk off
store 'y' to walk
pingroup = 9
DO WHILE walk <> 'n'
   store space(6) to component
   store space(5) to pinnumber
   store space(4) to function
   store space(40) to timingdi
   store space(16) to signalname
   @ 2,4 say 'Component : ' get component
   @ 3,4 say 'Pin number : ' get pinnumber
   @ 4,4 say 'Pin function : ' get function
   @ 5,4 say 'Timing diagram : ' get timingdi
   @ 6,4 say 'Signal name : ' get signalname
   read
   DO groupnum with function
   locate for node = component
   IF found()
      DO groupsearch with pingroup
   ENDIF
   insert blank before
   replace node with component
   replace pin with pinnumber
   replace func with function
   replace group with pingroup
   replace t_diagm with timingdi
   replace signame with signalname
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  wait 'Press any key to go on. OR press n to exit ..' to walk
ENDDO

PROCEDURE groupnum
  PARAMETER group_str
    DO CASE
      CASE group_str = 'o'
          pingroup = 0
      CASE group_str = 'vcc'
          pingroup = 2
      CASE group_str = 'gnd'
          pingroup = 2
      CASE group_str = 'c'
          pingroup = 5
          store 'ctrl' to group_str
      CASE group_str = 's'
          pingroup = 6
          store 'ctrl' to group_str
      CASE group_str = 'i'
          pingroup = 9
      OTHERWISE
          pingroup = 8
          store 'ctrl' to group_str
    ENDCASE
RETURN

PROCEDURE groupsearch
  PARAMETER itsnum
    DO WHILE group <= itsnum .AND. node = component
      skip
    ENDDO
RETURN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PROCEDURE after
  PARAMETER pinfunct
    DO WHILE func = pinfunct .AND. node = component
      skip
    ENDDO
RETURN
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 10

โปรแกรม STATEXT.C

```
/*
  Program to load logic state font from STATE.FON to create timing diagram

  By   : LT. Sirirote      # 34128031
  date : January 1, 1994
*/

#include <stdio.h>
#include <dos.h>
#include "state.fon"

void  write_char_gen( int ch, int n, char far *map );

/*=================================================================
  main()
=================================================================*/
void  main()
{ int c;
  int tdiagm[] = {241,243,249,241,243,245};  /* display timing on CRT */

  /* loads state character generator 1 - 9 graphic characters */
  write_char_gen( 241, 9, (char far *)font );

  for(c=0;c<6;c++) putchar(tdiagm[c]);
  printf( "\nReady\n" );
  printf( "Test by hold Alt and enter any numerical keypad from 241 - 249\n" );
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*=====================================================================
  write_char_gen()
    changes character generator for 'n' characters starting
    from the character 'ch' to 'map'
=====================================================================*/
void  write_char_gen( int ch, int n, char far *map )
{
  _CX = n;                      /* 'n' characters */
  _DX = (unsigned char)ch;      /* start with character 'ch'  */
  _BX = 0x1000;                 /* 16 bytes per character */
                                /* using map number 0 */
  _ES = FP_SEG( map );          /* segment of new char gen */
  _BP = FP_OFF( map );          /* offset of new char gen */
  _AX = 0x1100;                 /* func. 11h - load char gen */
                                /* sub func. 0 - user defined */
  geninterrupt( 0x10 );         /* BIOS call  (int. 10h) */
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 10 (ต่อ)

แฟ้ม STATE.FON

```
/*
  Include-font-file using fonts from NORMAL.FON of CU Writer
  by   : LT. J.Sirirote      # 34128031
  date : January 1, 1994
*/

typedef unsigned char  font_t[16];
const font_t  font[10]  /* 241..249 */
  = {
     { 0x00, 0x00, 0x00, 0xff, 0x00, 0x00, 0x00, 0x00,      /* 241 */
       0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 },
     { 0x00, 0x00, 0x00, 0x00, 0x80, 0x94, 0xAA, 0xAA,    /* 242 */
       0xA2, 0xB2, 0x82, 0xFC, 0x00, 0x00, 0x00, 0x00 },
     { 0x00, 0x00, 0x00, 0x80, 0x80, 0x80, 0x80, 0x80,     /* 243 */
       0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0xff, 0x00 },
     { 0x00, 0x00, 0x00, 0x00, 0x02, 0x3C, 0x40, 0x80,    /* 244 */
       0x98, 0x90, 0x88, 0x7C, 0x00, 0x00, 0x00, 0x00 },
     { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,    /* 245 */
       0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0x00 },
     { 0x00, 0x00, 0x00, 0x80, 0x40, 0x3C, 0x02, 0x01,    /* 246 */
       0x21, 0x51, 0x62, 0x3C, 0x00, 0x00, 0x00, 0x00 },
     { 0x00, 0x00, 0x00, 0x00, 0x02, 0x52, 0xAA, 0xAA,    /* 247 */
       0x8A, 0xCA, 0xAC, 0x48, 0x00, 0x00, 0x00, 0x00 },
     { 0x00, 0x00, 0x00, 0x00, 0x02, 0x04, 0x78, 0x80,    /* 248 */
       0x86, 0x96, 0x92, 0x6C, 0x00, 0x00, 0x00, 0x00 },
     { 0x00, 0x00, 0x00, 0xff, 0x80, 0x80, 0x80, 0x80,    /* 249 */
       0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00 },
   };
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 11

โปรแกรม TRANSF.C

```
/*
  Program to read VPX parameter file, search for the same file name(.dbf)
          and then rename it to $$$$$.dbf

  By  : LT. Sirirote      # 34128031
  date : June 1, 1994
*/

#include <stdio.h>

char vpx_send[16];
char vpx_recieve[] = "$$$$$.dbf";

int main(void)
{ FILE *fp;
  int ch;
  int i;

  i = 0;
  fp = fopen("@@@@@@","rt");              /* read VPX parameter */
  while((ch = fgetc(fp)) != EOF)
     { vpx_send[i] = ch;
       i++;
     }
  fclose(fp);
  vpx_send[i-1] = '.';                    /* concate to be .dbf file name */
  vpx_send[i]  =  'd';
  vpx_send[i+1] = 'b';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  vpx_send[i+2] = 'f';
  vpx_send[i+3] = '\0';
  fp = fopen(vpx_send,"r");                    /* check if exist .dbf file */
  if(fp == NULL)
     { printf("Cannot open %s\n",vpx_send);
       exit(1);
     }
  fclose(fp);
  asm                                          /* rename current .dbf file to $$$$$.dbf */
     { mov dx, OFFSET vpx_send                  /* current file */
       mov di, OFFSET vpx_recieve               /* new file */
       mov ah, 0x56
       int 0x21
     }
  return (0);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 12

โปรแกรม SINK.C

```
/*
  Program to rename $$$$$.dbf to its old name
  By   : LT. Sirirote      # 34128031
  date : June 1, 1994
*/

#include <stdio.h>

char vpx_send[16];
char vpx_recieve[] = "$$$$$.dbf";
char vpx_param[] = "@@@@@@";

int main(void)
{ FILE *fp;
  int ch;
  int i;

  i = 0;
  fp = fopen(vpx_param,"rt");              /* read VPX parameter */
  while((ch = fgetc(fp)) != EOF)
     { vpx_send[i] = ch;
       i++;
     }
  fclose(fp);
  vpx_send[i-1] = '.';                     /* concate to be .dbf file name */
  vpx_send[i]  =  'd';
  vpx_send[i+1] = 'b';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  vpx_send[i+2] = 'f';
  vpx_send[i+3] = '\0';
  asm                                   /* rename $$$$$.dbf file to its old .dbf file */
    { mov dx, OFFSET vpx_recieve         /* current file */
      mov di, OFFSET vpx_send           /* new file */
      mov ah, 0x56
      int 0x21

      mov ah, 0x41                      /* delete @@@@@ file */
      mov dx, OFFSET vpx_param
      int 0x21
    }
  return (0);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 18

โปรแกรม TSRKBD.DOC

```
/*
  Program to load logic state font from STATE.FON to create timing diagram

  By   : LT. Sirirote      # 34128031
  date : June 1, 1994
*/
#include <conio.h>
#include <dos.h>

extern unsigned _heaplen = 512;
extern unsigned _stklen = 512;
static void interrupt (*old_key)();          /*  int 09H interrupt pointer  */
static void interrupt tsr_key(void);         /*  Function for new int 09H  */
static void interrupt (*old_int64)();        /*  int 64H interrupt pointer  */
static void interrupt new_int64();           /*  Function for new int 64H */
static void active_tsr();                    /*  Function active TSR  */

static unsigned myss;                        /*  old stack segment keeper */
static unsigned stack;                       /*  old stack pointer keeper */
static char key;                             /*  activity  choice  */

main()
{ struct SREGS s;
  unsigned sizeprogram;

  key = 0;
  segread(&s);
  myss = s.ss;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  stack = _SP;

  /*..... Look. It's been loaded. Has'n it ? ..... */
  old_int64 = getvect(0x64);              /* empty interrupt  */
  if(! old_int64)                         /* use for store signature  */
     setvect(0x64, new_int64);
  else
     { cprintf("TSR already loaded\n\r");
       exit(0);
     }

  old_key = getvect(9);                   /*  get old ISR address  */
  setvect(9,tsr_key);                     /* Set interrupt vecters to new handle */

  sizeprogram = myss+((stack+50)/16)-_psp;
  keep(0, sizeprogram);
}

void interrupt new_int64()
{
  return;
}

void interrupt tsr_key(void)
{ union REGS r;
  char far *t = (char far*) 1050;/*  head pointer 0000:041A  */
  char far *ta = (char far*) 1050;/*  head pointer 0000:041A  */

  (*old_key)();                           /*  Call old int 09H  */
  if(*t != *(t+2))                        /*  if not empty  */
     { t += *t-30+5;                      /*  point to the scan code position  */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ta += *ta-30+4;              /*  point to the ascii code position  */
switch(*t)
  {
    case 0x09:               /* number 5 thai keybord scan code */
      if( key )
        *ta = 245;           /* change to state ascii code */
      break;
    case 0x03:               /* number 1 thai keybord scan code */
      if( key )
        *ta = 241;           /* change to state ascii code */
      break;
    case 0x05:               /* number 3 thai keybord scan code */
      if( key )
        *ta = 243;           /* change to state ascii code */
      break;
    case 0x0d:               /* number 9 thai keybord scan code */
      if( key )
        *ta = 249;           /* change to state ascii code */
      break;
    case 0x29:               /* toggle normal - state key */
      key = !key;
      r.h.ah = 0x10;
      int86(0x16, &r, &r);
      break;
    case 0x37:               /* Remove STR stat font program */
      if( *ta = 0x2a)        /* by '*' key  */
        { disable();
            _SP = stack;
            _SS = myss;
          enable();
          setvect(9, old_key);
          setvect(0x64, old_int64);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                freemem(_psp);
                cprintf("TSR already remove\n\r");
                exit(0);
              }
        }
    }
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกที่ 14

โปรแกรมย่อย ROMTEXT.ASM

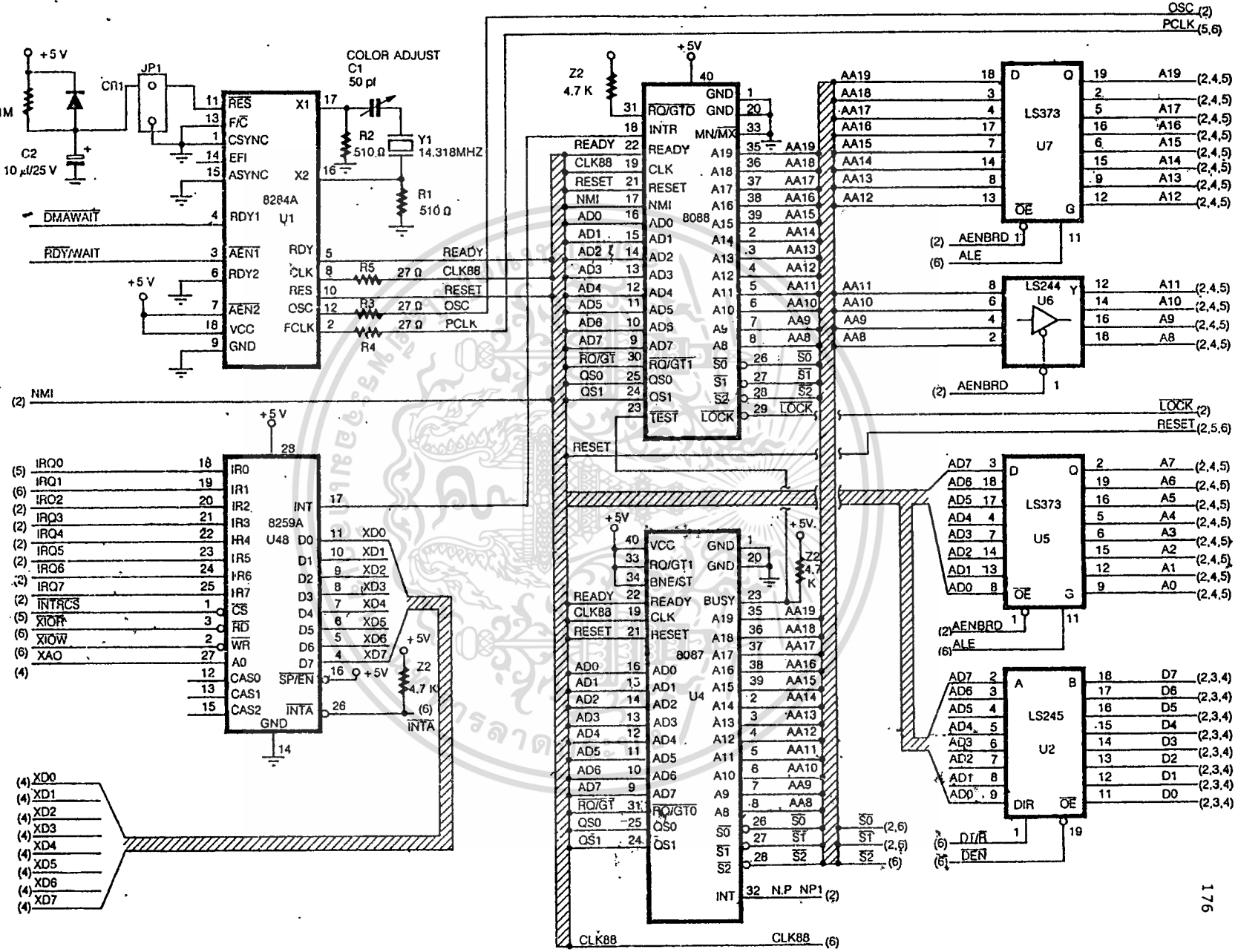
```
/*
  Program to reload font from ROM character generator
  By   : LT. Sirirote      # 34128031
  date : ....., ..., 1994
*/


        .MODEL    small

        .CODE
        PUBLIC    _LoadRomFon

_LoadRomFon PROC
        mov       bl,0
        mov       ax,0002h
        int       10h
        .EXIT
_LoadRomFon ENDP

END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

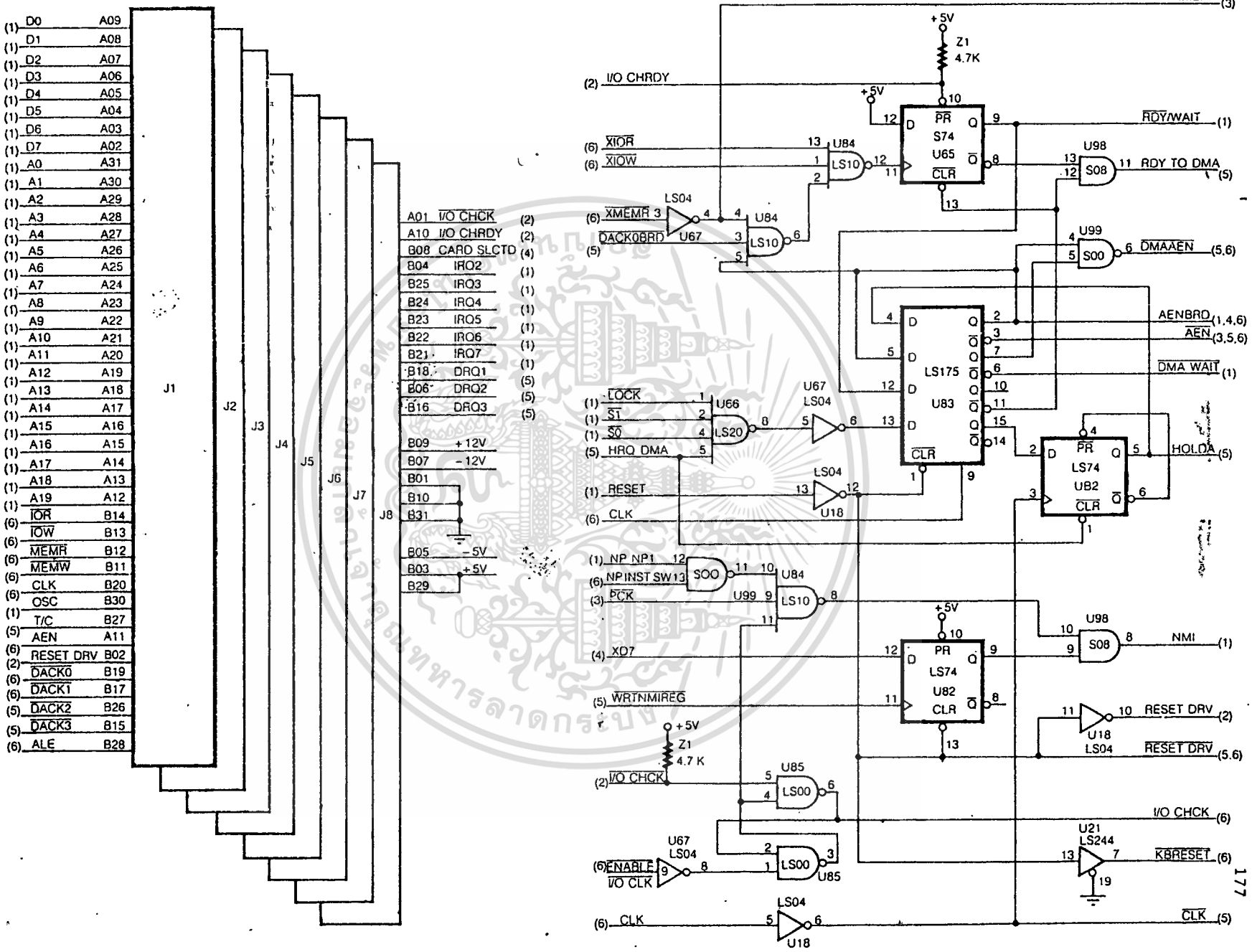
ภาคผนวกที่ 15

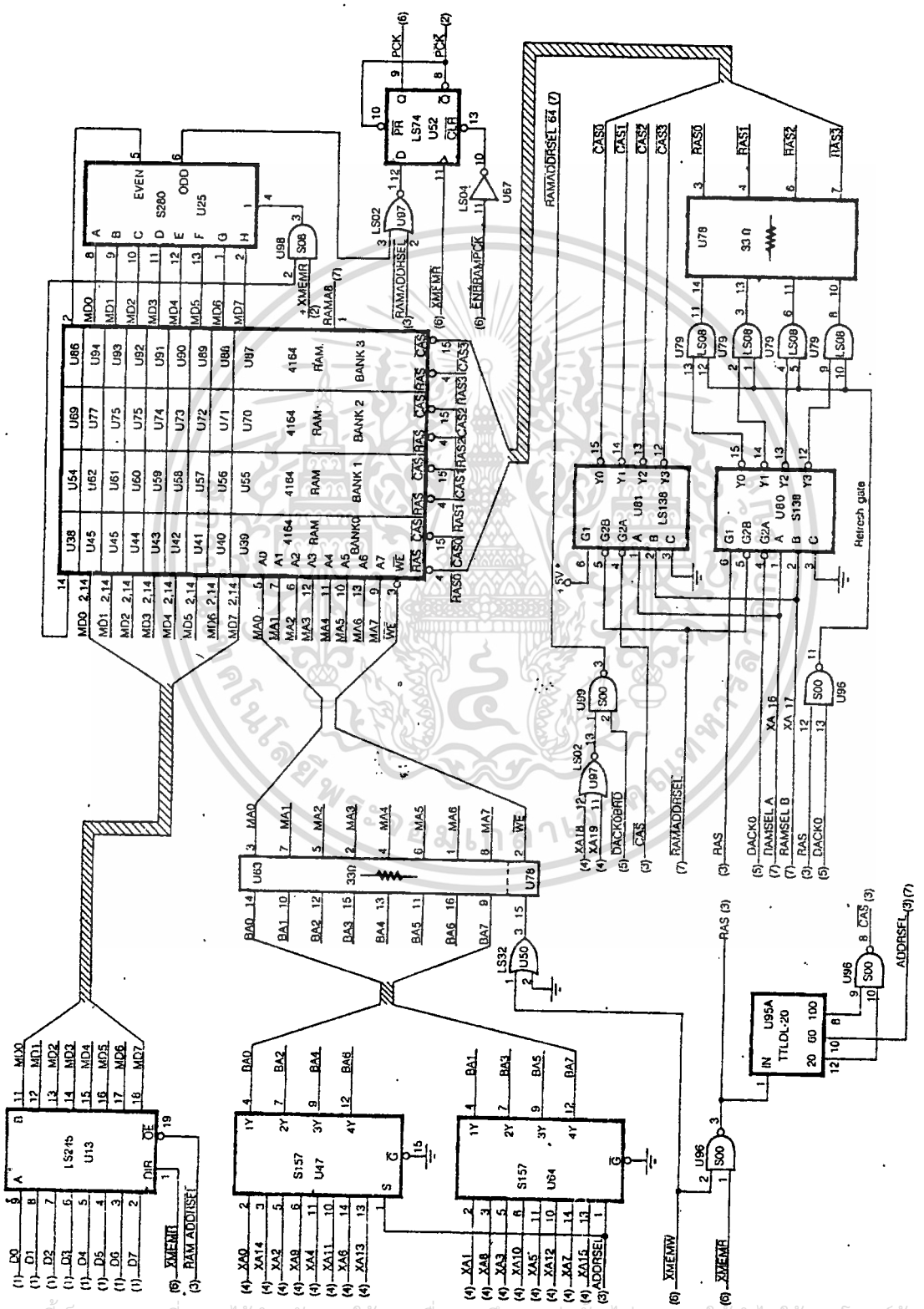
รายละเอียด วงจรหลัก PC/XT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



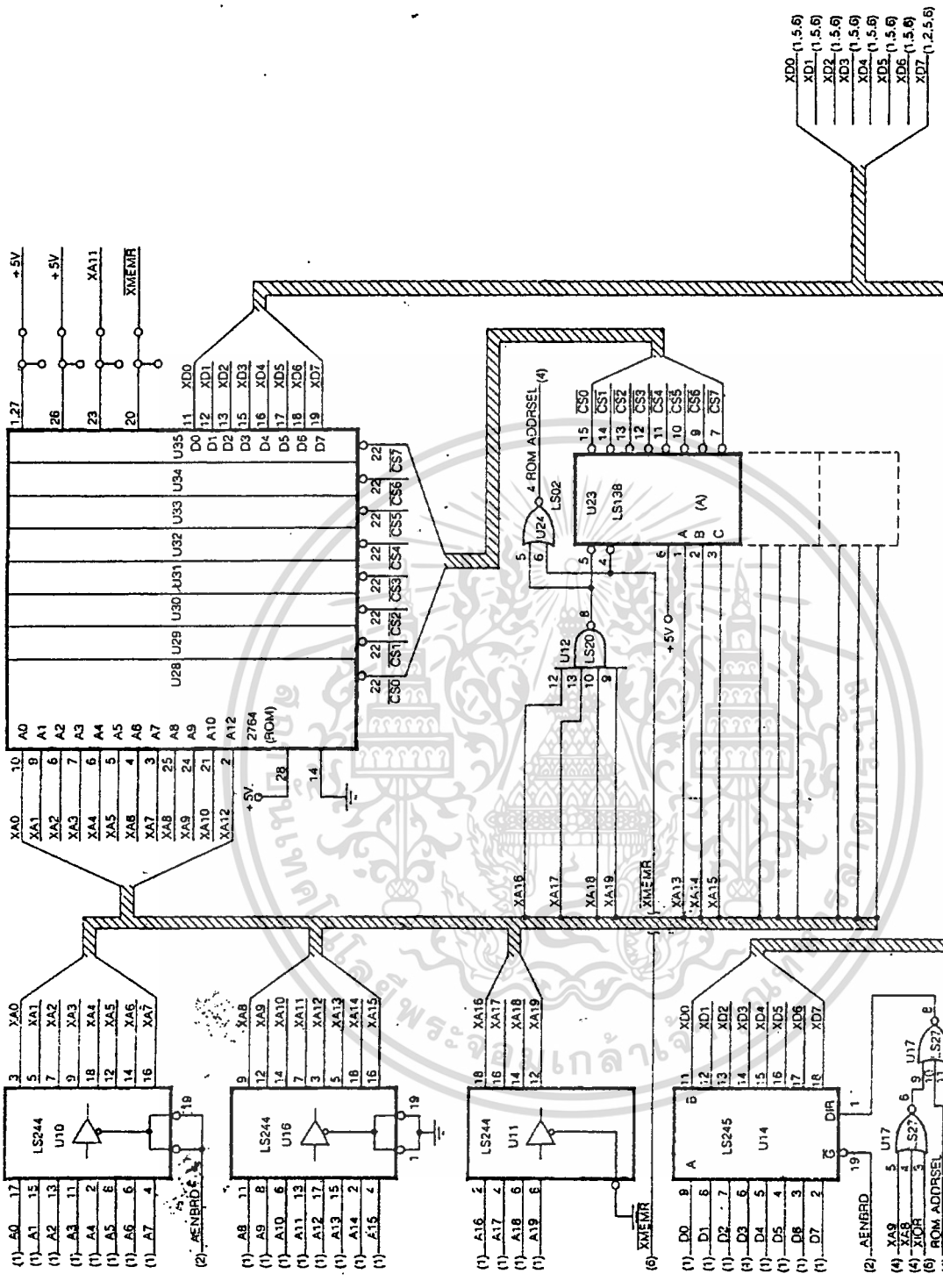
รูปวงจร PC/XT แผ่นที่ 1


J1
J2
J3
J4
J5
J6
J7
J8
(1) D0 A09
(1) D1 A08
(1) D2 A07
(1) D3 A06
(1) D4 A05
(1) D5 A04
(1) D6 A03
(1) D7 A02
(1) A0 A31
(1) A1 A30
(1) A2 A29
(1) A3 A28
(1) A4 A27
(1) A5 A26
(1) A6 A25
(1) A7 A24
(1) A8 A23
(1) A9 A22
(1) A10 A21
(1) A11 A20
(1) A12 A19
(1) A13 A18
(1) A14 A17
(1) A15 A16
(1) A16 A15
(1) A17 A14
(1) A18 A13
(1) A19 A12
(1) IOR B14
(6) IOW B13
(6) MEMR B12
(6) MEMW B11
(6) CLK B20
(6) OSC B30
(1) T/C B27
(5) AEN A11
(6) RESET DRV B02
(2) DACK0 B19
(6) DACK1 B17
(6) DACK2 B26
(5) DACK3 B15
(6) ALE B28
A01 I/O CHCK (2)
A10 I/O CHRDY (2)
B08 CARD SLCTD (4)
B04 IRQ2 (1)
B25 IRQ3 (1)
B24 IRQ4 (1)
B23 IRQ5 (1)
B22 IRQ6 (1)
B21 IRQ7 (1)
B18 DRQ1 (5)
B06 DRQ2 (5)
B16 DRQ3 (5)
B09 +12V
B07 -12V
B01
B10
B31
B05 -5V
B03 +5V
B29
(2) I/O CHRDY
+5V
Z1
4.7K
(6) XIOR
(6) XIOW
U84
LS10
RDY/WAIT (1)
U98
S08
RDY TO DMA (5)
(6) XMEMR
DACK0BRD
U67
LS04
U99
S00
DMAAEN (5,6)
LS175
U83
AENBRD (1,4,6)
AEN (3,5,6)
DMA WAIT (1)
(1) LOCK
(1) S1
(1) S0
(5) HRQ DMA
U66
LS20
U67
LS04
LS74
U82
HOLDA (5)
(1) RESET
(6) CLK
U18
(1) NP NP1
(6) NP INST SW13
(3) PCK
SOO
U84
U99
LS10
(4) XD7
NMI (1)
(5) WRTNMIREG
RESET DRV (2)
RESET DRV (5,6)
U85
LS00
I/O CHCK (6)
U21
LS244
KBRESET (6)
(6) ENABLE
I/O CLK
CLK (5)
177



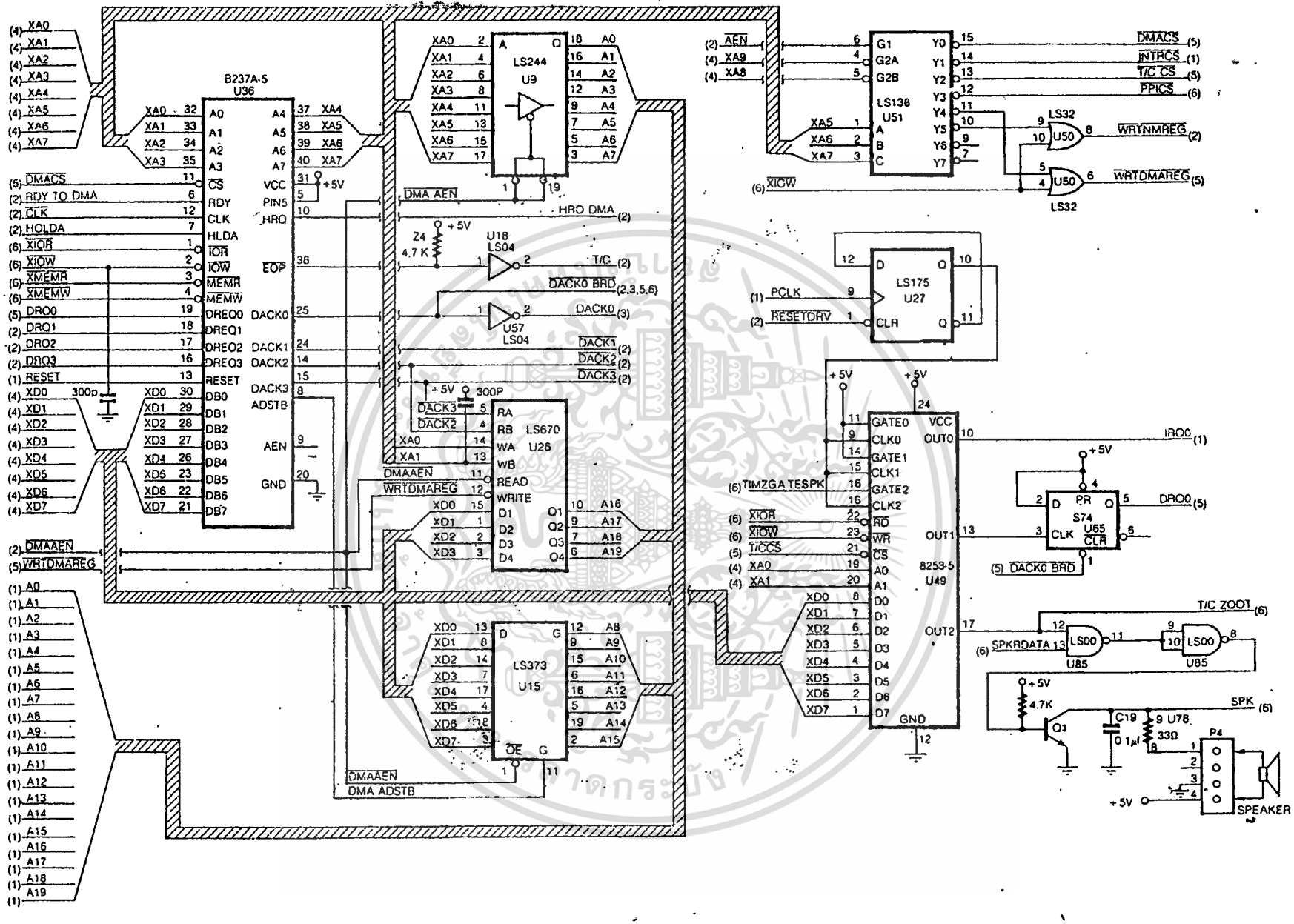
รูปที่ 3.3 วงจร PC/XT แผ่นที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

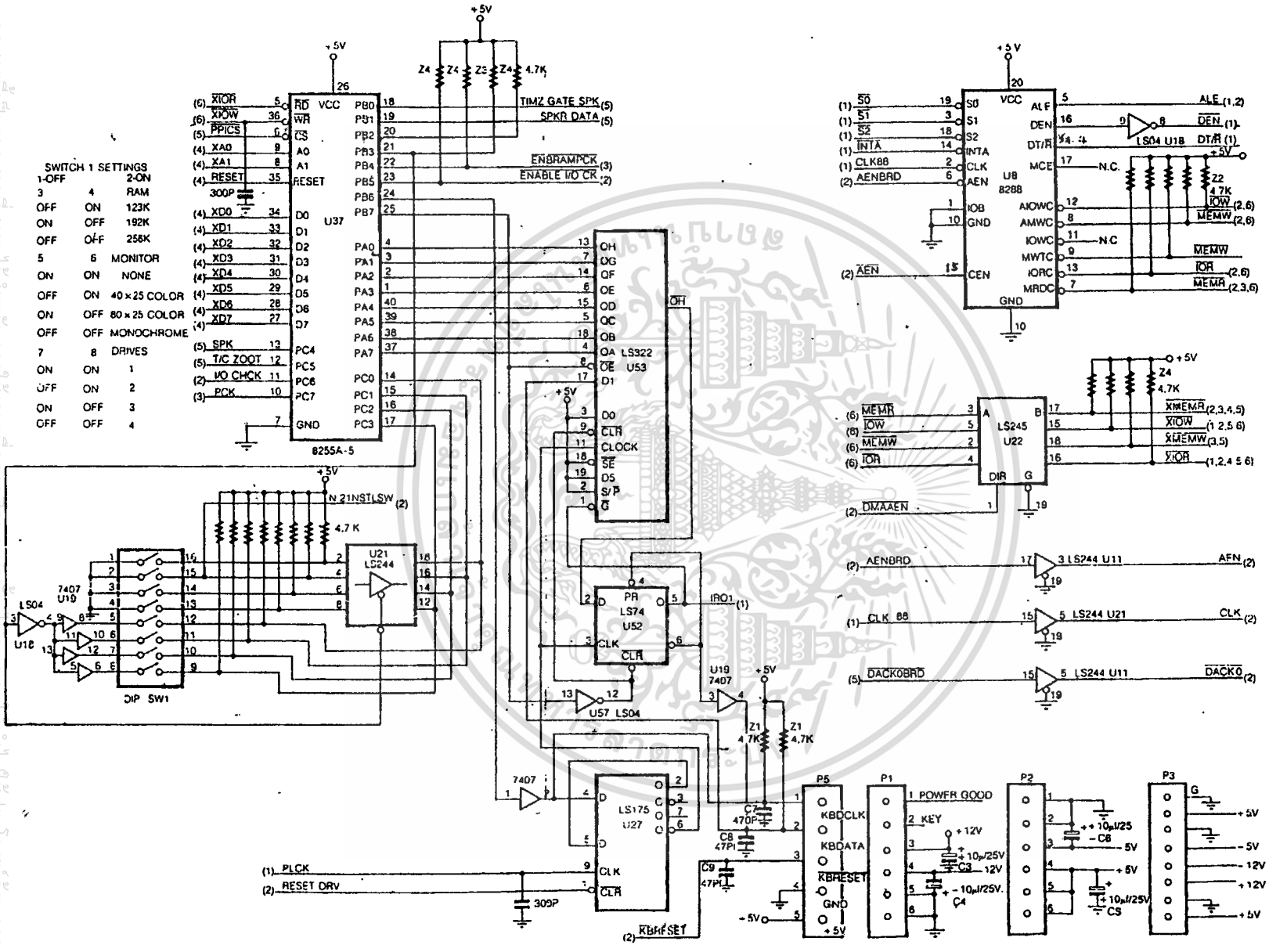


รูปวงจร PC/XT แผ่นที่ 4

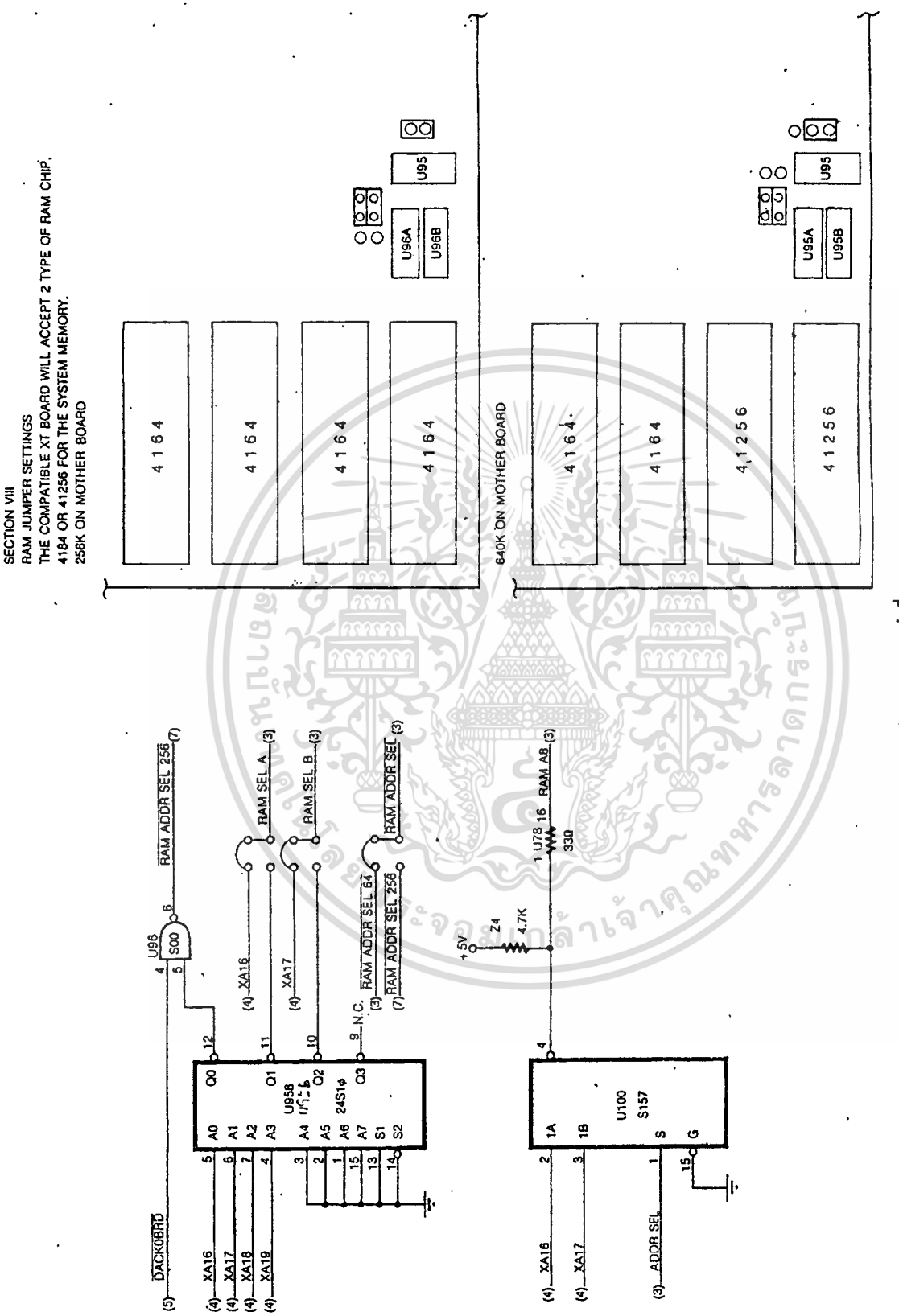
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจร PC/XT แผ่นที่ 6



รูปวงจร PC/XT แผ่นที่ 6



รูปวงจร PC/XT แผ่นที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	ร้อยเอก ศิริโรจน์ ฉันทใด
วันเดือนปีเกิด	วันที่ 15 มกราคม พ.ศ. 2496
สถานที่เกิด	จังหวัดสุโขทัย
วุฒิการศึกษาระดับปริญญาตรี	การศึกษาศาสตร์บัณฑิต วิชาเอกฟิสิกส์
สถานที่สำเร็จการศึกษา	มหาวิทยาลัยศรีนครินทรวิโรฒ พิษณุโลก
ปีที่สำเร็จการศึกษา	ปีการศึกษา 2523
ผลงานทางวิชาการที่ได้รับการการตีพิมพ์	เรื่องระบบผู้เชี่ยวชาญในการค้นหาข้อขัดข้องบนแผงวงจรหลักไมโครคอมพิวเตอร์ วารสารพระจอมเกล้าลาดกระบัง ฉบับที่ 2 ปีที่ 3
รางวัลหรือทุนที่เคยได้รับ	-
ประสพการณ์การทำงาน	พ.ศ. 2528 ดูงาน การซ่อมบำรุง เรดาร์ RASIT ประเทศฝรั่งเศส พ.ศ. 2531 ดูงาน เครื่องตรวจซ่อมแผงวงจรดิจิตอล GenRad ประเทศสหรัฐอเมริกา พ.ศ. 2537 ดูงาน การซ่อมบำรุงเครื่องหาทิศวิทยุTWR ประเทศสหรัฐอเมริกา
อาชีพปัจจุบัน	รับราชการตำแหน่ง หัวหน้าหมวดสื่อสาร กองซ่อมเครื่องสื่อสารอิเล็กทรอนิก กรมการทหารสื่อสาร กองทัพบก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้