



เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง

Real Time TTL In-Circuit Testing

โดย

นาย ปรีชา แซ่ลิ่ม

วัน เดือน ปี.....-5.ค.ศ.2541  
เลขทะเบียน.....038547  
เลขเรียกหนังสือ.....T.4008R 24/70.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง 038547

ปริญญานิพนธ์ ปีการศึกษา 2540

ภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

สาขาวิชา วิศวกรรมการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง

ผู้จัดทำ

นาย ปรีชา แซ่ลิ่ม เลขประจำตัว 38013408



.....อาจารย์ที่ปรึกษา

(ศศ.ทรงชัย วีระทวีมาศ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง  
นักศึกษา นาย ปรีชา แซ่ลิ้ม 38013408

อาจารย์ที่ปรึกษา ผศ. ทรงชัย วีระทวีมาศ

ระดับการศึกษา วิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมการวัดคุมทางอุตสาหกรรม  
ปีการศึกษา พ.ศ. 2540

### บทคัดย่อ

ในปัจจุบันแม้ว่าการพัฒนาทางด้านเทคโนโลยีจะก้าวหน้าไปอย่างรวดเร็วก็ตาม ในระบบดิจิตอล ไอซีตระกูล TTL (Transistor Transistor Logic) ยังถูกนำมาใช้งานกันอย่างแพร่หลาย ทั้งทางด้านการศึกษาและการนำไปใช้ในระบบอุตสาหกรรม ซึ่งหลังจากการใช้งานไปแล้วในระดับหนึ่งก็อาจเกิดการเสื่อมสภาพ โดยการตรวจสอบการทำงานหรือหาข้อบกพร่องที่เกิดขึ้นนั้น ก็อาจใช้เครื่องมือต่าง ๆ เช่น Logic Probe มาช่วยในการตรวจสอบการทำงานได้ แต่อาจทำให้ล่าช้า

ปริญญานิพนธ์นี้จึงได้นำเสนอ เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง โดยอาศัยหลักการที่เรียกว่า DRC (Dynamic Reference Comparison) มาใช้ในการทำงานของเครื่อง ซึ่งส่งผลให้การค้นหาจุดบกพร่อง ที่ต้องการเป็นไปได้อย่างรวดเร็วยิ่งขึ้น

**Thesis** : Real Time TTL In-Circuit Testing  
**Students** : Mr. Preecha Saelim 38013408  
**Advisor** : Assis.Prof. Songchai Weerathaweemas

**Education Level** : Bachelor of Industrial Instrument Engineering

**Education Year** : 1997

### Abstract

Today , technology going on with high speed. In digital systems , TTL integrated circuit has been use with popularity both education and Industrial. Which is after work may be degenerate. We can use with more instrument to check it fault such as logic probes but may be too difficult.

In this Thesis represent Real Time TTL In-Circuit Testing. Which is use DRC ( Dynamic Reference Comparison) checking concept. Cosequence can find the fault point very quite.

### กิตติกรรมประกาศ

ขอขอบพระคุณ คุณพ่อคุณแม่ ผู้ที่ให้ชีวิต ขอขอบพระคุณ อาจารย์ ทรงชัย วีระทวิมาศ ผู้ที่ได้  
 ประสิทธิ์ประสาทวิชา ขอขอบพระคุณ อาจารย์ ธวัชชัย และ คุณนงลักษณ์ เกื่อนฉวี ซึ่งเป็นผู้  
 ที่มีพระคุณเป็นอย่างยิ่ง และขอขอบคุณ นายภาวิน โกมลวิภาค ที่ให้ยืมคอมพิวเตอร์, เครื่องมือ,  
 หนังสือ และอุปกรณ์ต่าง ๆ รวมถึงที่พักอาศัยและเพื่อนๆ ทุกคนที่ให้กำลังใจ จนทำให้ปริญญา  
 นิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี



นาย ปรีชา แซ่ลิ้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	IX
สารบัญ Flowchart	X
บทที่ 1 บทนำ	1
ความเป็นมา	1
วัตถุประสงค์ของโครงการ	1
ขอบเขตของโครงการ	1
เนื้อหาของปริิญาานิพนธ์	2
บทที่ 2 ทฤษฎีเบื้องต้น	3
2.1 ความหมายของศัพท์ที่ควรทราบ	3
2.2 ไอซี ทรานซิสเตอร์ TTL (Transistor Transistor Logic)	4
2.3 ข้อควรปฏิบัติในการใช้งาน IC ทรานซิสเตอร์ TTL	11
บทที่ 3 หลักการและการออกแบบวงจรเครื่องตรวจ IC TTL ในวงจร	14
3.1 หลักการ DRC (Dynamic Reference Comparison)	14
3.2 วงจรเปรียบเทียบ	15
3.3 วงจรบัฟเฟอร์	15
3.4 วงจรจัดขาอินพุตเอาต์พุต	16
3.5 วงจรเอาต์พุต	21
3.6 การจ่ายไฟเลี้ยงให้กับไอซีอ้างอิง	22
3.7 วงจรเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง	24
3.8 วงจรกำเนิดเสียงกคคีย์	27

## สารบัญ(ต่อ)

หน้า

3.9 การต่อใช้งานคีย์บอร์ด	27
บทที่ 4 การเขียนโปรแกรมควบคุมการทำงานเครื่องตรวจ IC TTL ในวงจร	28
4.1 แนวทางการเริ่มเขียนโปรแกรมควบคุมการทำงานเครื่องตรวจ ฯ	28
4.2 การทำงานของโปรแกรมควบคุมเครื่องตรวจ ฯ	32
บทที่ 5 การทดลองนำไปใช้งาน	51
5.1 การทดลองใช้งาน	51
5.1.1 การทดลองกับ IC ที่มีเอาต์พุตเป็นแบบ Totem - Pole	51
5.1.2 การทดลองกับ IC ที่มีเอาต์พุตเป็นแบบ Open Collector	52
บทที่ 6 สรุปผลโครงการและแนวทางการพัฒนาต่อ	55
6.1 สรุปผลโครงการ	55
6.2 ปัญหาที่พบและข้อมูลเสนอแนะ	55
6.3 แนวทางในการพัฒนาต่อ	55
บรรณานุกรม	57
ภาคผนวก ก.	58
การใช้งานของเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง	58
1. ลักษณะของ Key Panel	58
2. การป้อนเบอร์ไอซี	59
3. การ Replay Right & Replay Left	61
4. การ Clear	61
5. การ Delete	62
6. การใช้ Function	63
6.1 Sound On/Off	63
6.2 Keyboard Test	64
6.3 Speaker Test	65
6.4 Restart System	66

## สารบัญ(ต่อ)

	หน้า
6.5 LCD Test	67
7. การ Test เพื่อตรวจสอบว่าไอซีทดสอบดีหรือเสีย	68
ภาคผนวก ข.	70
Data Sheet ของ Dip Reed Relay	70



## สารบัญรูป

หน้า

รูปที่ 2.1	ทึทึแอลแบบ Open-Collector	
รูปที่ 2.1 (ก)	วงจรภายในของ 3 Input NAND แบบ Open-Collector	7
รูปที่ 2.1 (ข)	NAND-Gate แบบ Open-Collector 2 ตัว นำเอาค้พุดมาต่อร่วมนกัน	8
รูปที่ 2.1 (ค)	ผลเทียบเท่าของวงจร (ข) คือ NOR ของ AND	8
รูปที่ 2.2	ทึทึแอลแบบ Totem - Pole Output	9
รูปที่ 2.3	ทึทึแอลแบบ Tri-State Logic	10
รูปที่ 2.4	แสดงการทำงานของอินพุตควบคุมของเกตแบบ Tri-State	
รูปที่ 2.4 (ก)	Tri-State buffer with active-high control	11
รูปที่ 2.4 (ข)	Tri-State inverter with active-low control	11
รูปที่ 3.1	การเปรียบเทียบไอซีที่ต้อการทดสอบกับไอซีอ้างอิงโดยใช้หลักการ DRC	14
รูปที่ 3.2	วงจรเปรียบเทียบ โดยใช้ไอซีเบอร์ 7485	15
รูปที่ 3.3	วงจรบัพเฟอร์	16
รูปที่ 3.4	แสดงวิธีการต่อสวิตช์ 2 ทางตามแนวกิด	17
รูปที่ 3.5	แสดงวิธีการต่อคิพสวิตช์แทนสวิตช์ 2 ทาง	18
รูปที่ 3.6	แสดงวิธีการต่อคิพสวิตช์เพียง 3 ตัว แทนวงจรในรูปที่ 3.5	18
รูปที่ 3.7	แสดงวงจรเมื่อค้คสวิตช์ SW2 และ SW4 ตามรูปที่ 3.6 ออก	19
รูปที่ 3.8	แสดงการนำเอาไอซี 74125 มาใช้งานแทนคิพสวิตช์	20
รูปที่ 3.9	แสดงวงจรเอาค้พุดใช้ 7474 ที่ต่อรับผลการเปรียบเทียบ จาก 7485	22
รูปที่ 3.10	แสดงตำแหน่ง ขา $V_{cc}$ และ Gnd ของไอซีทึทึแอลที่ปรากฏในคู่มือ	23
รูปที่ 3.11	แสดงวงจรสำหรับจ่ายไฟเลี้ยงให้ไอซีอ้างอิงต่อหนึ่งขาไอซี	23
รูปที่ 3.12	วงจรที่สมบูรณ์ของเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง	25
รูปที่ 3.13	วงจรที่สมบูรณ์ของวงจรจ่ายไฟเลี้ยงให้กับไอซีอ้างอิง	26
รูปที่ 3.14	วงจรกำเนิดเสียงกคคิย	27
รูปที่ 4.1	ไอซีทึทึแอลเบอร์ 7408	28

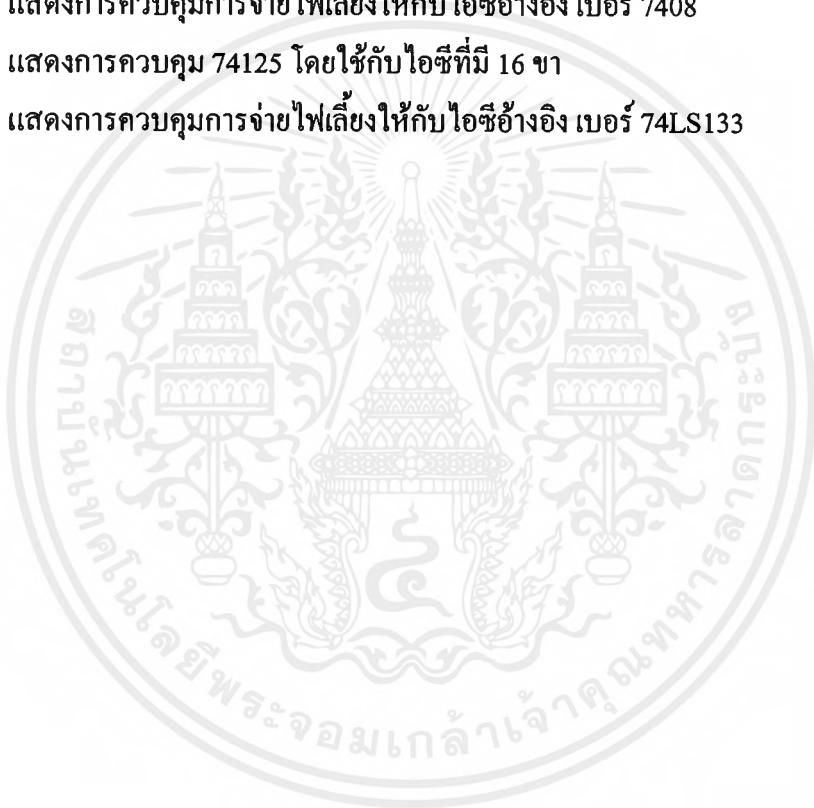
## สารบัญรูป(ต่อ)

	หน้า
รูปที่ 4.2 ไอซีทีทีแอลเบอร์ 74133	30
รูปที่ 5.1 วงจร Shift Register ขนาด 4 Bit	53
รูปที่ 5.2 วงจร Scan Display	54
ภาคผนวก รูปที่ 1 แสดงคีย์ 16 คีย์ ใช้งาน และ LCD	58



## สารบัญตาราง

	หน้า
ตารางที่ 2.1 คุณสมบัติของ TTL แบบต่าง ๆ	4
ตารางที่ 2.2 คุณสมบัติของ COMS แบบต่าง ๆ และมี $V_{cc}$ ระหว่าง 4.5 ถึง 5.5 V	6
ตารางที่ 3.1 ตารางสรุปสถานะทาง Logic ที่ใช้ควบคุมการทำงานของ 74125	20
ตารางที่ 4.1 แสดงการควบคุม 74125 โดยใช้กับไอซีที่มี 14 ขา	29
ตารางที่ 4.2 แสดงการควบคุมการจ่ายไฟเลี้ยงให้กับ ไอซีอ้างอิง เบอร์ 7408	30
ตารางที่ 4.3 แสดงการควบคุม 74125 โดยใช้กับไอซีที่มี 16 ขา	31
ตารางที่ 4.4 แสดงการควบคุมการจ่ายไฟเลี้ยงให้กับ ไอซีอ้างอิง เบอร์ 74LS133	31



## สารบัญ Flowchart

	หน้า
Flowchart ที่ 4.1 แสดงการ Initial ในส่วนต่าง ๆ และรอรับการกดคีย์บอร์ด	32
Flowchart ที่ 4.2 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “0”	35
Flowchart ที่ 4.3 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “1”	36
Flowchart ที่ 4.4 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “2”	37
Flowchart ที่ 4.5 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “3”	38
Flowchart ที่ 4.6 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “4”	39
Flowchart ที่ 4.7 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “5-9”	40
Flowchart ที่ 4.8 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “ENT/YES”	41
Flowchart ที่ 4.9 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “CLR/DEL”	42
Flowchart ที่ 4.10 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “Replay Right & Function”	43
Flowchart ที่ 4.11 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “Replay Left”	44
Flowchart ที่ 4.12 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “MON/CAN”	45
Flowchart ที่ 4.13 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “TEST”	46
Flowchart ที่ 4.14 แสดงวิธีการเปิดตารางเพื่อ ส่ง Data ไปที่ 8255 Port	49

# บทที่ 1

## บทนำ

### ความเป็นมา

ในปัจจุบันไม่ว่าจะเป็นในด้านการศึกษาหรือการนำไปประยุกต์ใช้งานจริงสำหรับทางด้านดิจิทัลแล้ว ไอซีในตระกูล TTL (Transistor Transistor Logic) ก็ยังคงได้รับการยอมรับ และนำไปใช้งานกันอย่างแพร่หลาย ซึ่งในการนำไปใช้งานต่าง ๆ ก็ย่อมต้องการเครื่องมือที่จะช่วยสนับสนุนในการตรวจสอบการทำงานของวงจร เช่น Logic Probe โดยการนำเอามาตรวัดสถานะทาง Logic ว่าในขณะนั้นมีสถานะ Logic ใดอยู่ และเป็นไปตามเงื่อนไขของการทำงานที่ถูกต้องหรือไม่ ซึ่งอาจทำให้การค้นหาจุดบกพร่องของวงจรที่ต้องการตรวจสอบนั้นล่าช้าไปบ้าง

ด้วยเหตุนี้จึงได้เกิดแนวคิดที่จะออกแบบและสร้างเครื่องมือ ที่จะช่วยในการค้นหาจุดเสียหรือข้อบกพร่องของวงจร คือ เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง (Real Time TTL In-Circuit Testing)

### วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาการออกแบบวงจรดิจิทัล
2. เพื่อศึกษาการนำเอา ไมโครคอนโทรลเลอร์ MCS-51 มาประยุกต์ใช้งาน

### ขอบเขตของโครงการ

ขอบเขตของโครงการนี้จะเป็นการสร้างแผงวงจรของ เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง โดยจะสามารถตรวจสอบไอซีในตระกูล TTL (74 Series) ได้สูงสุดจำนวน 16 ขา โดยหลักการที่ได้นำมาใช้ในการทำงานและออกแบบ เครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง คือ หลักการDRC (Dynamic Reference Comparison) โดยหลักการนี้จะใช้ลอจิกไอซีเบอร์เดียวกันกับที่จะตรวจสอบ และให้มีทำงานคู่กับไอซีในวงจรที่ต้องการหาข้อบกพร่องของการทำงานขณะที่ใช้งานจริงแล้วเปรียบเทียบเอาต์พุตที่ได้ว่าตรงกันหรือไม่ ก็จะบอกได้ว่าไอซีที่อยู่ในวงจรมันดีหรือเสีย

## เนื้อหาของปริญญาโท

- บทที่ 2 กล่าวถึงทฤษฎีเบื้องต้นของไอซีทีทีแอลโดยทั่วไปที่ควรทราบ
- บทที่ 3 กล่าวถึงหลักการและการออกแบบวงจรของ เครื่องตรวจ IC TTL ในวงจร
- บทที่ 4 กล่าวถึงการเขียนโปรแกรมควบคุมการทำงานของเครื่องตรวจ IC TTL ในวงจร
- บทที่ 5 กล่าวถึงการทดลองใช้งานของเครื่องตรวจ IC TTL ในวงจร
- บทที่ 6 กล่าวถึงการสรุปผลโครงการและแนวทางการพัฒนาต่อ



## บทที่ 2

### ทฤษฎีเบื้องต้น

#### 2.1 ความหมายของศัพท์ที่ควรทราบ

ก่อนที่จะกล่าวถึงรายละเอียดต่าง ๆ ของไอซีในตระกูล TTL จะขออธิบายความหมายของคำศัพท์บางคำที่ควรทราบดังต่อไปนี้

Power Dissipation / Gate หมายถึง กำลังไฟฟ้า (ค่าเฉลี่ย) ที่สูญเสียไปในเกต ในสภาวะที่เปลี่ยนสถานะเอาต์พุต ไปมาอย่างต่อเนื่อง

Propagation Delay / Gate หมายถึง ระยะเวลาหน่วงที่เกิดขึ้น ระหว่างการเปลี่ยนสถานะที่อินพุตของเกตกับการเปลี่ยนสถานะที่เอาต์พุต

Noise Margin หมายถึง ขนาดใหญ่ที่สุดของสัญญาณรบกวนที่ไม่มีผลต่อการทำงานของเกต

Fan In หมายถึง จำนวนอินพุตของ เกต ใด เกต หนึ่ง

Fan Out หมายถึง จำนวนอินพุตของเกตตระกูลเดียวกันที่สามารถนำมาต่อกับเอาต์พุตของเกตตัวหนึ่ง ๆ ได้ โดยที่เกตตัวนั้นยังสามารถทำงานได้ตามปกติ

Low-level input voltage (V)  $V_{ILmax}$  หมายถึง ระดับแรงดันสูงสุดทางอินพุตที่เป็นสถานะ Low

Low-level output voltage (V)  $V_{OLmax}$  หมายถึง ระดับแรงดันสูงสุดทางเอาต์พุตที่เป็นสถานะ Low

High-level input voltage (V)  $V_{IHmin}$  หมายถึง ระดับแรงดันต่ำสุดทางอินพุตที่เป็นสถานะ High

High-level output voltage (V)  $V_{OHmin}$  หมายถึง ระดับแรงดันต่ำสุดทางเอาต์พุตที่เป็นสถานะ High

Low-level input current (mA)  $I_{ILmax}$  หมายถึง กระแสสูงสุดทางอินพุตที่ต้องการสำหรับดึงให้เป็นสถานะ Low โดยมีค่าเป็นลบและกระแสจะไหลออกจากขาอินพุต

Low-level output current (mA)  $I_{OLmax}$  หมายถึง กระแสสูงสุดทางเอาต์พุตที่สามารถจ่ายได้ เมื่อเป็นสถานะ Low ขณะที่รักษาระดับแรงดันให้น้อยกว่าหรือเท่ากับ  $V_{OHmin}$  โดยมีค่าเป็นบวกและกระแสจะไหลเข้าขาเอาต์พุต

High-level input current (mA)  $I_{IHmax}$  หมายถึง กระแสสูงสุดทางอินพุตที่ต้องการสำหรับดึงให้เป็นสถานะ High โดยมีค่าเป็นบวกและกระแสจะไหลเข้าขาอินพุต

High-level output current (mA)  $I_{OHmax}$  หมายถึง กระแสสูงสุดทางเอาต์พุตที่สามารถจ่ายได้ เมื่อเป็นสถานะ High ขณะที่รักษาระดับแรงดันให้มากกว่าหรือเท่ากับ  $V_{OHmin}$  โดยมีค่าเป็นลบและกระแสจะไหลออกจากขาเอาต์พุต

## 2.2 ไอซี ทรานซิสเตอร์ TTL (Transistor Transistor Logic)

วงจรรวมทรานซิสเตอร์ทีแอล มีโครงสร้างที่ประกอบด้วยทรานซิสเตอร์ทั้งที่ตำแหน่งอินพุตและที่ตำแหน่งเอาต์พุต ทรานซิสเตอร์ที่ตำแหน่งอินพุตเป็นทรานซิสเตอร์ ที่มีอิมิตเตอร์หลายอันเป็นวงจรรวมทรานซิสเตอร์แรกๆ ที่ออกแบบเพื่อผลิตเป็นไอซีโดยตรง จึงมีโครงสร้างที่เหมาะสมกับการผลิตเป็นไอซีเป็นอย่างมาก ทำให้สามารถผลิตได้ในราคาถูก นอกจากนี้ทีแอลยังมีคุณสมบัติที่เหนือกว่าวงจรรวมทรานซิสเตอร์ก่อนๆ อีกหลายอย่าง เช่น Propagation Delay น้อย อัตราการสูญเสียพลังงาน (Power Dissipation) ต่ำ ด้วยเหตุผลเหล่านี้จึงทำให้ทีแอลกลายเป็นวงจรรวมทรานซิสเตอร์ที่ได้รับความนิยมสูงสุดในปัจจุบันนี้ วงจรรวมทรานซิสเตอร์ในระบบดิจิทัลได้รับการพัฒนาอย่างกว้างขวางทำให้เกิดการใช้งานในรูปแบบที่ต่างกันหลายรูปแบบ และจากการใช้งานที่แตกต่างกันนี้จึงทำให้เกิดการพัฒนาทีแอลแบบต่างๆ ซึ่งเน้นคุณสมบัติเฉพาะอย่างต่างกัน ดังแสดงใน ตารางที่ 2.1

ตารางที่ 2.1 คุณสมบัติของ TTL แบบต่าง ๆ

Description	Symbol	Family						
		74	74L	74H	74S	74LS	74AS	74ALS
Typical propagation delay (ns)		9	33	6	3	9	1.6	5
Power consumption per gate (mW)		10	1	22	20	2	20	1.3
Speed-power product (pJ)		90	33	132	60	18	32	6.5
LOW-level input voltage (V)	$V_{ILmax}$	0.8	0.7	0.8	0.8	0.8	0.8	0.8
LOW-level output voltage (V)	$V_{OLmax}$	0.4	0.4	0.4	0.5	0.5	0.5	0.5
HIGH-level input voltage (V)	$V_{IHmin}$	2.0	2.0	2.0	2.0	2.0	2.0	2.0
HIGH-level output voltage (V)	$V_{OHmin}$	2.4	2.4	2.4	2.7	2.7	2.7	2.7
LOW-level input current (mA)	$I_{ILmax}$	-1.6	-0.18	-2.0	-2.0	-0.4	-2.0	-0.2
LOW-level output current (mA)	$I_{OLmax}$	16	3.6	20	20	8	20	8
HIGH-level input current ( $\mu$ A)	$I_{IHmax}$	40	10	50	50	20	20	20
HIGH-level output voltage ( $\mu$ A)	$I_{OHmax}$	-400	-200	-500	-1000	-400	-2000	-400

การเลือกใช้ที่ทีแอลจึงต้องเลือกตามคุณลักษณะที่ต้องการเน้นตามการใช้งาน เป็นต้นว่าถ้าต้องการประหยัดไฟ (เช่นกรณีใช้ไฟจากแบตเตอรี่ที่มีขนาดจำกัด) ควรเลือก 74L ถ้าต้องการความเร็วสูงควรเลือก 74H หรือ 74S แต่ถ้าต้องการประนีประนอมระหว่างความเร็วกับการประหยัดไฟ ก็ควรเลือก 74LS เป็นต้น

### คุณสมบัติของ TTL ในแบบต่างๆ พอสรุปสาระสำคัญได้ดังนี้

Standard TTL(74) เป็นรูปแบบดั้งเดิม ความเร็วปานกลาง กินไฟปานกลาง มีแฟนเฮาต์ (Fan Out) เท่ากับ 10 ใช้งานได้ที่ความถี่สูงสุด 20 MHz

Low Power TTL (74L) ออกแบบให้ดึงกระแสและกำลังไฟฟ้าน้อยกว่า Standard TTL (74) อยู่ประมาณ 10 เท่า แต่มีผลทำให้ Propagation Delay สูงขึ้น และ แฟนเฮาต์ลดลง

High Speed TTL (74H) ออกแบบโดยเน้นที่ความเร็วสูง โดยยอมให้ดึงกระแสและกำลังไฟฟ้ามาก จึงได้ Propagation Delay ต่ำ และแฟนเฮาต์ได้มาก แต่มีอัตราการสูญเสียพลังงานสูง

Schottky TTL (74S) มีการวางซอตต์กีไดโอด (Schottky Diode) ระหว่างคอลเลกเตอร์กับเบสของทรานซิสเตอร์ทุก ๆ ตัวในวงจร ทำให้การเปลี่ยนสถานะเกิดขึ้นได้รวดเร็วขึ้น โดยไม่ต้องเพิ่มกระแสและกำลังไฟฟ้า

Low Power Schottky TTL (74LS) ใส่ทั้งซอตต์กีไดโอดให้แก่ทรานซิสเตอร์ และลดกระแสในวงจรด้วยจึงให้ทั้งความเร็ว และกินกำลังไฟฟ้าน้อย

Advanced Schottky TTL (74AS) เป็นรุ่นที่พัฒนาต่อมาจาก 74S ให้กินกำลังไฟฟ้าต่ำลง 7 mW ต่อเกต มีเวลาหน่วง 1.5 ns ต่อเกต

Advanced Low Power Schottky TTL (74ALS) เป็นรุ่นที่พัฒนาต่อมาจาก 74LS กินกำลังไฟฟ้าเพียง 1 mW ต่อเกต มีเวลาหน่วง 4 ns ต่อเกต

Fast TTL (74F) มีความเร็วในการทำงานสูงกว่า 74L และ 74LS มีเวลาหน่วงอยู่ระหว่าง 1.5-6.0 ns

นอกจาก TTL แบบต่าง ๆ ที่ได้กล่าวมาแล้วนั้น ยังมีการผลิตไอซีที่มี Function การทำงานเป็นแบบ TTL แต่มีคุณสมบัติอื่น ๆ แบบ CMOS อันได้แก่ 74HC , 74AC และไอซีที่เป็น CMOS และมี Function การทำงานแบบ TTL แต่สามารถใช้แทน TTL ได้โดยตรง ได้แก่ 74HCT , 74ACT

### คุณสมบัติของ CMOS ที่มี Function การทำงานแบบ TTL พอสรุปสาระสำคัญได้ดังนี้

High Speed COMS (74HC and 74HCT) ใช้งานกับความถี่สูงสุด 30 MHz

74HC ทางอินพุตรับสัญญาณลอจิกในระดับCOMS และเอาต์พุตสามารถขับได้ทั้งCOMSและTTL

74HCT ทางอินพุตรับสัญญาณลอจิกในระดับTTL และเอาต์พุตสามารถขับได้ทั้งCOMSและTTL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Fast Advance CMOS (74AC and 74ACT) มีความเร็วในการทำงานสูงกว่า 74HC,74HCT มาก กินกำลังต่ำคือ 0.1 mW ด้านทานต่อสัญญาณรบกวนได้ดี

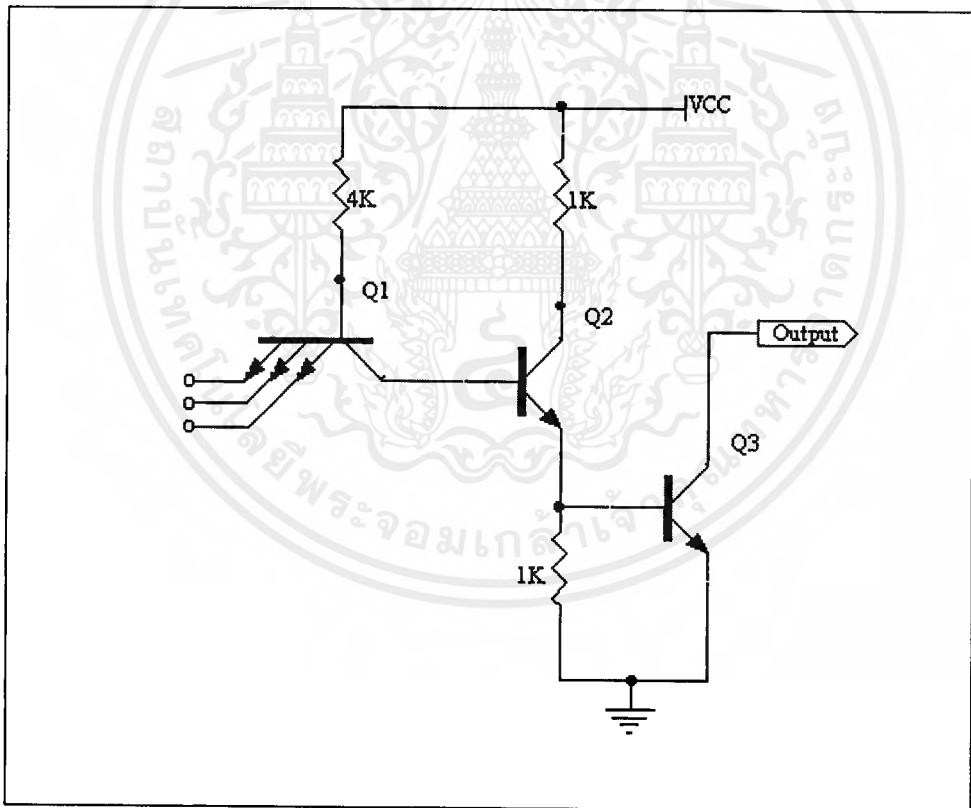
74AC และ 74ACT มีความแตกต่างกันเหมือนกับรุ่น 74HC ,74HCT คุณสมบัติอย่างอื่นของ 74HC,74HCT,74AC,74ACT สามารถดูเพิ่มเติมได้จากตารางที่ 2.2

ตารางที่ 2.2 คุณสมบัติของ CMOS แบบต่าง ๆ และมี  $V_{CC}$  ระหว่าง 4.5 ถึง 5.5 V

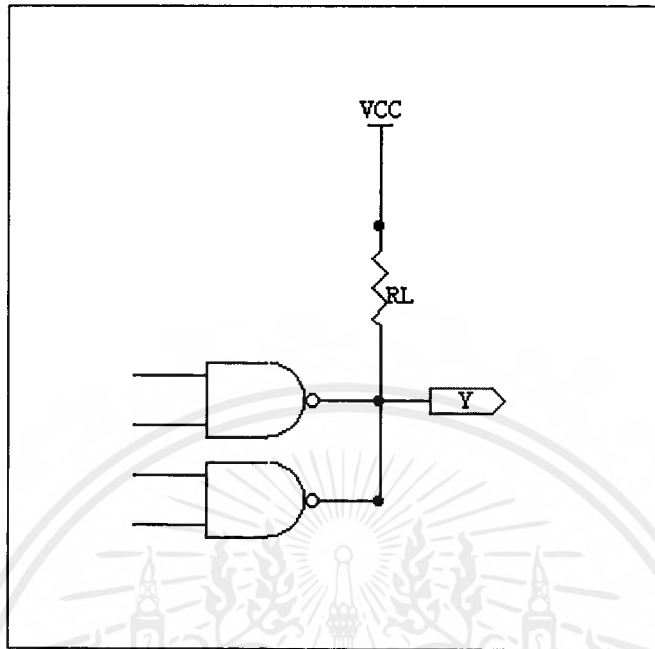
Description	Symbol	Condition	Family			
			74HC	74HCT	74AC	74ACT
Typical propagation delay (ns)			18	18	5.25	4.75
Quiescent power dissipation per gate (mW)		$V_{in} = 0$ or $V_{CC}$	0.0025	0.0025	0.005	0.005
Power dissipation capacitance (pF)	$C_{PD}$	$V_{in} = 2.4$ V	n/a	12	n/a	7.5
Dynamic power dissipation per gate (mW/MHz)			0.60	0.60	0.75	0.75
Total power dissipation per gate (mW)		$f = 100$ kHz	0.0625	0.0625	0.080	0.080
Speed-power product (pJ)		$f = 1$ MHz	0.6025	0.6025	0.755	0.755
		$f = 10$ MHz	108	108	39	36
		$f = 100$ kHz	1.1	1.1	0.4	0.4
Input leakage current ( $\mu$ A)		$f = 1$ MHz	10.8	10.8	3.9	3.6
		$f = 10$ MHz	108	108	39	36
		$V_{in} = \text{any}$	$\pm 1$	$\pm 1$	$\pm 1$	$\pm 1$
LOW-level input voltage (V)	$V_{ILmax}$		1.35	0.8	1.35	0.8
HIGH-level input voltage (V)	$V_{IHmin}$		3.85	2.0	3.85	2.0
LOW-level output current (mA)	$I_{OLmaxC}$	CMOS load	0.02	0.02	0.05	0.05
	$I_{OLmaxT}$	TTL load	4.0	4.0	24.0	24.0
LOW-level output voltage (V)	$V_{OLmaxC}$	$I_{out} \leq I_{OLmaxC}$	0.1	0.1	0.1	0.1
		$I_{out} \leq I_{OLmaxT}$	0.33	0.33	0.37	0.37
HIGH-level output current (mA)	$I_{OHmaxC}$	CMOS load	-0.02	-0.02	-0.05	-0.05
	$I_{OHmaxT}$	TTL load	-4.0	-4.0	-24.0	-24.0
HIGH-level output voltage (V)	$V_{OHminC}$	$ I_{out}  \leq  I_{OLmaxC} $	4.4	4.4	4.4	4.4
	$V_{OHminT}$	$ I_{out}  \leq  I_{OLmaxT} $	3.84	3.84	3.76	3.76

นอกจากความแตกต่างในเรื่องการกินกระแสและความเร็วแล้ว ทีทีแอลยังแบ่งออกเป็น 3 ชนิด ตามลักษณะของวงจรด้านเอาต์พุตคือ

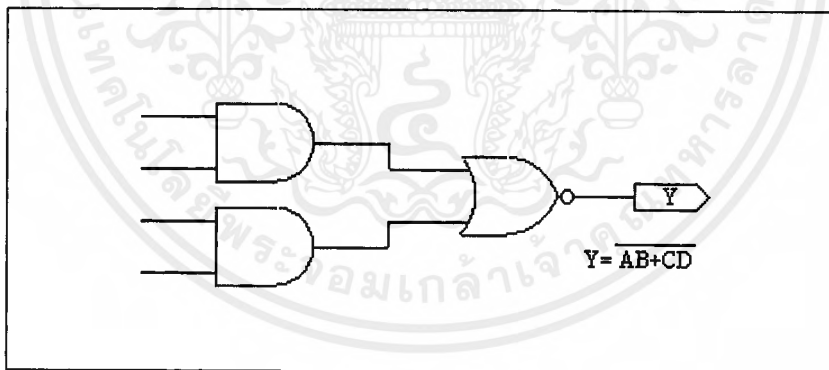
1. Open Collector ทีทีแอลชนิดนี้จะไม่มีตัวต้านทาน ที่ใช้เป็นโหลดภายในสำหรับตัวทรานซิสเตอร์ที่เอาต์พุต ผู้ใช้จะต้องนำขั้วเอาต์พุตต่อผ่านตัวต้านทานภายนอกไปยังขั้วบวกของแหล่งจ่ายไฟ เราจะเลือกใช้ทีทีแอลชนิดนี้ในกรณีที่ต้องการให้ไอซีสามารถจ่ายกระแสโหลดมาก ๆ ประโยชน์อีกอย่างหนึ่งของทีทีแอลแบบ Open Collector คือสามารถนำเอาเอาต์พุตของเกตหลายๆ ตัวมาต่อร่วมกัน การทำงานของเกตที่นำเอาต์พุตมาต่อร่วมกันนี้ เป็นการทำงานในลักษณะของเกตแบบแอนด์ (AND-Gate) และวิธีนี้เรียกว่า “Wired AND” คือเกตแบบแอนด์ที่เกิดจากการต่อสายนั่นเอง ดังรูปที่ 2.1 (ข)



รูปที่ 2.1 (ก)



รูปที่ 2.1 (ข)



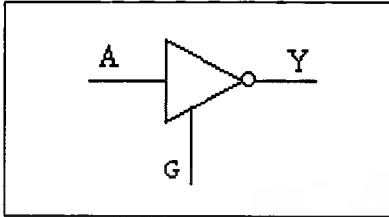
รูปที่ 2.1 (ค)

## รูปที่ 2.1 ทีทีแอลแบบ Open-Collector

- (ก) วงจรภายในของ 3 Input NAND แบบ Open-Collector
- (ข) NAND-Gate แบบ Open-Collector 2 ตัว นำเอาต์พุตมาต่อร่วมกัน
- (ค) ผลเทียบเท่าของวงจร (ข) คือ NOR ของ AND





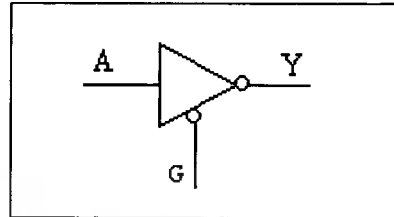


รูปที่ 2.4 (ก)

Tri-State buffer

with active-high control

$$Y = \bar{A} \text{ ถ้า } G = 1 \text{ และ } Y = \text{HI Z} \text{ ถ้า } G = 0$$



รูปที่ 2.4 (ข)

Tri-State inverter

with active-low control

$$Y = \bar{A} \text{ ถ้า } G = 0 \text{ และ } Y = \text{HI Z} \text{ ถ้า } G = 1$$

รูปที่ 2.4 แสดงการทำงานของอินพุตควบคุมของเกตแบบ Tri-State

### 2.3 ข้อควรปฏิบัติในการใช้งาน IC ตระกูล TTL

เนื่องจากไอซีทีทีแอลถูกนำมาใช้กันอย่างแพร่หลายในปัจจุบัน และยังเป็นที่ยอมรับใช้กันต่อไปอีกนานสำหรับพื้นฐานทั่วไป สิ่งที่เราควรจะรู้รายละเอียดที่จำเป็นต่อการออกแบบวงจรที่ใช้ไอซีทีทีแอลในที่นี้จะกล่าวถึงส่วนที่สำคัญ ๆ เท่านั้น ส่วนรายละเอียดนอกเหนือจากนั้นสามารถหาอ่านเพิ่มเติมได้จาก หนังสือประเภทคู่มือการออกแบบวงจร ไอซีทีทีแอล วงจรแบบต่างๆ ของไอซีตระกูลทีทีแอลนั้น สามารถนำมาต่อกันได้โดยตรง เอาต์พุตจากทีทีแอลตัวหนึ่งสามารถนำมาป้อนอินพุตของทีทีแอลอีกตัวหนึ่ง หรือหลาย ๆ ตัว จำนวนเกตที่เอาต์พุตสามารถขับหรือป้อนสัญญาณให้ได้นั้นเรียกว่า “แฟนเอาต์” (Fan Out) ทีทีแอลโดยทั่วไปสามารถขับเกตจากเอาต์พุตของมันได้ 10 ตัว จึงเรียกว่า มีค่าแฟนเอาต์เท่ากับสิบ ทีทีแอลที่ออกแบบไว้เป็นแบบพิเศษให้มีค่าแฟนเอาต์สูงกว่านี้ (ตามปกติ 30) เรียกว่า “บัฟเฟอร์” (Buffer) นอกจากนี้ยังมีไอซีทีทีแอลที่มีคุณสมบัติทางแรงดันและกระแสที่เอาต์พุต ในกรณีพิเศษสำหรับขับโหลดบางอย่างที่ไม่ใช่เกตแบบทีทีแอล เช่น วงจรขับหลอดตัวเลขแบบเรืองแสง หรือแบบแอลอีดี เป็นต้น ในกรณีหลังนี้เราไม่พูดถึงค่าแฟนเอาต์เพราะไม่มีความหมาย ถ้าเอาต์พุตของทีทีแอลถูกนำไปขับเกตมากกว่าจำนวนแฟนเอาต์ของมัน ปัญหาแรกที่จะพบ ก็คือสัญญาณรบกวนที่มีโอกาสเข้าไปรบกวนการทำงานของวงจรมาก

ขึ้น และค่าความแตกต่างแรงดันและกระแสระหว่างสถานะ Hi กับ Lo อาจน้อยเกินไปจนทำให้เกิดการทำงานผิดพลาดได้

เอาต์พุตของทีทีแอลตามปกติจะมีเพียงสองสถานะเท่านั้น คือสถานะ Lo ซึ่งมีค่าแรงดันสูงกว่าคินเล็กน้อย (ไม่เกิน 0.8 โวลต์) สถานะนี้เราเรียกว่า ค่าตรรกะ “0” สำหรับระบบสัญญาณตรรกะบวกในสถานะนี้เอาต์พุตของทีทีแอลสามารถรับกระแส (Sink Current) จากภายนอกได้สูงทีเดียว (ตามปกติประมาณ 16 มิลลิแอมป์) กับสถานะ Hi ซึ่งเรียกว่า ค่าตรรกะ “1” สำหรับระบบสัญญาณตรรกะบวก เมื่อเอาต์พุตของเกิดต่อกับวงจรทีทีแอลอื่น เอาต์พุตเพียงแต่ต้องรักษาสถานะไปอัสกลับทางของอิมิตเตอร์ที่อินพุตเหล่านั้น และจ่ายกระแสเพียงกระแสรั่ว (Leakage Currents) เท่านั้น ค่าแรงดันที่สถานะ Hi จะมีค่าเกิน 2.4 โวลต์แต่ต่ำกว่า +5 โวลต์ของแหล่งจ่ายไฟค่อนข้างมาก โดยในเกณฑ์ปกติจะมีค่าประมาณ 3.3 โวลต์

หากเราต้องการค่าเต็มของแหล่งจ่ายไฟ เราต้องใช้ R ต่อโงระหว่างเอาต์พุตกับแหล่งจ่ายไฟ R ตัวนี้เรียกว่า “Pull-up Resistor” วงจรนี้มีประโยชน์สำหรับต่อเชื่อมทีทีแอลกับวงจรตรรกะตระกูลอื่น หรืออุปกรณ์อย่างอื่นซึ่งต้องการค่าเอาต์พุตในสถานะ Hi สูงถึง 5 โวลต์ แต่ไม่จำเป็นสำหรับการต่อเชื่อมกับทีทีแอลด้วยกัน ถ้าไม่มี Pull-up Resistor สถานะ Hi จะให้เอาต์พุตประมาณ 3.3 โวลต์เท่านั้น คุณสมบัติอีกอย่างหนึ่งของทีทีแอล ซึ่งจะต้องทำความเข้าใจดี เพื่อมิให้เกิดปัญหาในการใช้งานนั่นคือ การเปลี่ยนสถานะของเอาต์พุต เมื่อเอาต์พุตอยู่ในสถานะ Lo ทรานซิสเตอร์ตัวล่างของ Totem Pole เท่านั้นที่นำกระแส ส่วนในสถานะ Hi ตัวบนเท่านั้นที่นำกระแส ในระหว่างการเปลี่ยนสถานะจาก Lo ไป Hi ทรานซิสเตอร์ ทั้งสองตัวนำกระแสอย่างหนักกระแสชั่วขณะอาจมีค่าสูงเป็นสิบเท่าของกระแสปกติวิธีนี้ช่วยให้เอาต์พุตเปลี่ยนสถานะได้เร็ว แต่มันทำให้เกิดสัญญาณกวนขอดแหลม (Spike) ของกระแสที่ไหลผ่านสายจ่ายไฟเลี้ยง ซึ่งอาจมีขนาดสูงเป็นร้อยมิลลิแอมป์ และเกิดในช่วงเวลาประมาณ 10 ns สัญญาณนี้มีโอกาสรบกวนการทำงานของไอซีตัวอื่นได้มากทีเดียว เราจึงต้องแก้ด้วยการใส่ตัวเก็บประจุ (Filter Capacitor) ครอบแหล่งจ่ายไฟไว้ข้าง ๆ ตัวไอซีนั่นเลย เพื่อลดปัญหาการรบกวนกันทางสายจ่ายไฟ โดยปกติจะใช้ค่าประมาณ 0.1  $\mu\text{f}$

เนื่องจากทีทีแอลทำงานด้วยความเร็วสูง และสร้างสัญญาณรบกวนขอดแหลมออกมาด้วย นั้นการต่อวงจรจึงควรทำด้วยความระมัดระวัง วงจรที่ต่อกันแบบยุ่งเหยิง มีสายไขว้ไปมา และมักจะประสบปัญหาทำงานผิดพลาดได้ง่าย

อินพุตของทีทีแอลก็มีสองสถานะเช่นกัน ในสถานะ Lo เราต้องให้มีกระแสไหลลงคิน 1.6 มิลลิแอมป์ และต้องหาวิธีให้แรงดันที่มีอิมิตเตอร์มีค่า 0.8 โวลต์หรือต่ำกว่า ในสถานะ Hi เราเพียงแต่รักษาค่าแรงดันอินพุตให้สูงกว่า 2.4 โวลต์ และจ่ายกระแสรั่วเพียงเล็กน้อยให้แก่อินพุต

ส่วนอินพุตของทีทีแอลทีปปล่อยลอยไว้จะดึงตัวเองอยู่ในสถานะ Hi แต่ในทางปฏิบัติเราควรตรึงอินพุตที่ต้องการให้เป็น Hi กับแหล่งจ่ายไฟ +5 โวลต์ ไม่ควรปล่อยลอยไว้เฉยๆ เพราะอาจทำให้เกิด Ringing และเสี่ยงต่อการถูกรบกวนจากสัญญาณภายนอก ซึ่งอาจแก้ไขโดยให้ต่ออินพุตที่ต้องการให้เป็น Hi ผ่านความต้านทานค่า 1K กับ สาย +5 v แต่เราอาจต่ออินพุตเข้าหาแหล่งจ่ายไฟโดยตรง โดยไม่ต้องผ่าน R เลขก็ได้ อย่างไรก็ตามในการต่อวงจรทดลอง หรือวงจรต้นแบบที่มีไอซีไม่มากนัก เราอาจปล่อยอินพุตที่เป็น Hi ลอยไว้ได้ แต่อย่าโยงสายจากขาไอซี นั้น ๆ เมื่อถึงขั้นสร้างจริงแล้วการปล่อยลอยเป็นสิ่งที่ไม่ควรทำ



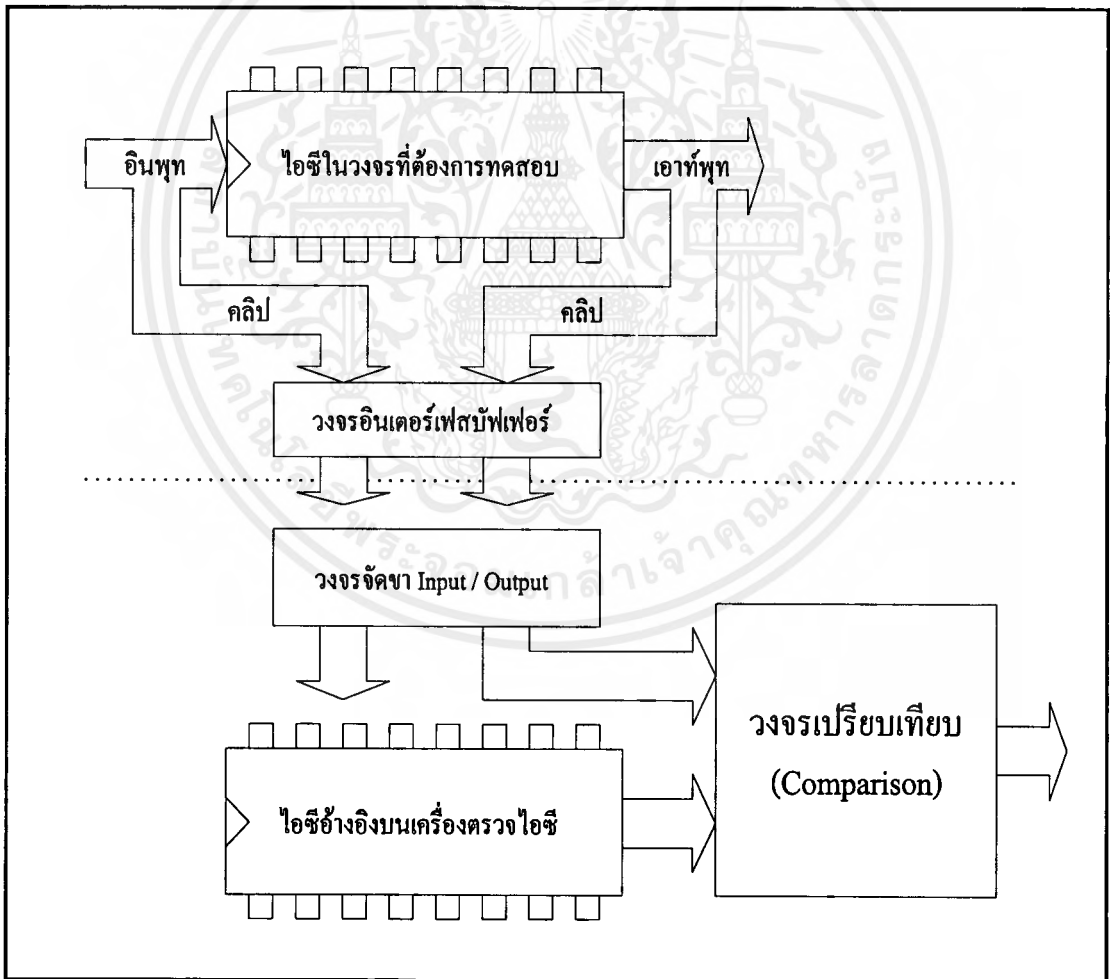
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### หลักการและการออกแบบวงจรเครื่องตรวจ IC TTL ในวงจร

#### 3.1 หลักการ DRC ( Dynamic Reference Comparison )

ในการออกแบบเครื่องตรวจ IC TTL นั้นได้นำ หลักการ DRC มาใช้ในการออกแบบตัววงจร ซึ่งหลักการนี้จะใช้ลอจิกไอซีเบอร์เดียวกับที่จะตรวจสอบทำงานคู่กับไอซีในวงจร ขณะที่ใช้งานจริง แล้วเปรียบเอาต์พุตที่ได้ว่าตรงกันหรือไม่ ก็จะบอกได้ว่าไอซีที่อยู่ในวงจรมันดีหรือเสีย จากหลักการที่กล่าวมาสามารถแสดงเป็นบล็อกไดอะแกรมการทำงานของวงจรได้ดังรูปที่ 3.1



รูปที่ 3.1 การเปรียบเทียบ ไอซีที่ต้องการทดสอบกับ ไอซีอ้างอิง โดยใช้หลักการ DRC

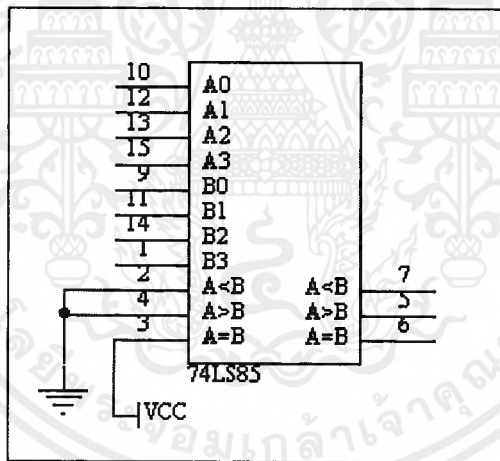
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 ได้แสดงให้เห็นถึงส่วนประกอบของหลักการ DRC และวงจรที่จะนำมาใช้ว่ามีส่วนประกอบอย่างไรบ้าง เช่น วงจรเปรียบเทียบ, วงจรบัพเฟอร์ ซึ่งสามารถอธิบายการทำงานของแต่ละส่วนของวงจรได้ดังหัวข้อต่อไปนี้

### 3.2 วงจรเปรียบเทียบ

การเลือกไอซีที่จะนำมาใช้ในวงจรนี้ต้องดูที่จำนวนอินพุตว่ามีเท่าใด ซึ่งในขอบเขตของโครงการนี้ได้กำหนดไว้ให้ไอซีที่ทดสอบสามารถตรวจสอบได้สูงสุด 16 บิต ดังนั้นเอาต์พุตสูงสุดของไอซีที่ใช้ทดสอบจึงเท่ากับ 16 เอาต์พุต จึงทำให้ต้องใช้วงจรเปรียบเทียบที่มีขาอินพุตเปรียบเทียบถึง 16 อินพุต เมื่อศึกษาจากในคู่มือไอซีที่ที่แอลแล้ว พบว่ามีไอซีที่สามารถทำหน้าที่เกี่ยวกับการเปรียบเทียบอยู่ด้วยกันหลายเบอร์ เช่นเบอร์ 7485, 7486, 74521, 74522, 74688 เป็นต้น

แต่ในที่นี้ได้เลือกใช้ ไอซีเบอร์ 7485 ซึ่ง 1 ตัวจะสามารถเปรียบเทียบได้ 4 อินพุต จึงต้องใช้ 7485 จำนวน 4 ตัว จึงสามารถเปรียบเทียบได้เท่ากับ 16 อินพุต



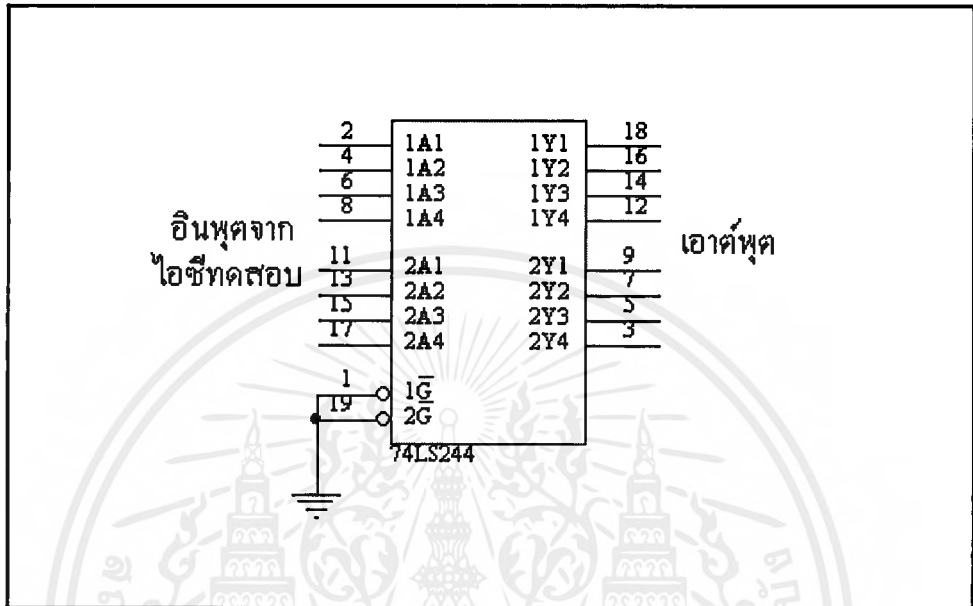
รูปที่ 3.2 วงจรเปรียบเทียบ โดยใช้ไอซีเบอร์ 7485

### 3.3 วงจรบัพเฟอร์

วงจรมีจะทำหน้าที่เป็นส่วนขับสัญญาณที่ได้มาจากขาไอซีทดสอบ ในวงจรเป้าหมายที่เราต้องการตรวจสอบการทำงานให้มีระดับลอจิกที่เหมาะสม เนื่องจากต้องใช้สายแพต่อจากไอซีที่ต้องการทดสอบมาเข้ากับเครื่องที่เราจะสร้างขึ้น ซึ่งอาจจะทำให้ขนาดของสัญญาณหรือระดับลอจิกลดลงได้ จากขอบเขตที่ได้กำหนดไว้ว่าสามารถตรวจสอบได้สูงสุด 16 บิต จากนั้นก็นำมาเป็นข้อพิจารณาว่าจะเลือกไอซีที่นำมาใช้ในส่วนนี้อย่างไร จึงได้เลือกเอาไอซีเบอร์ 74244 โดยไอซีเบอร์นี้คือ “Octal Buffer 3-State” มาใช้งานเป็นวงจรบัพเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยต้องใช้ไอซีเบอร์นี้ จำนวน 2 ตัว ในที่นี้จะแสดงขาส่วนอินพุตและเอาต์พุตให้ดูเพียง 1 ตัว ดังรูปที่ 3.3



รูปที่ 3.3 วงจรบัฟเฟอร์

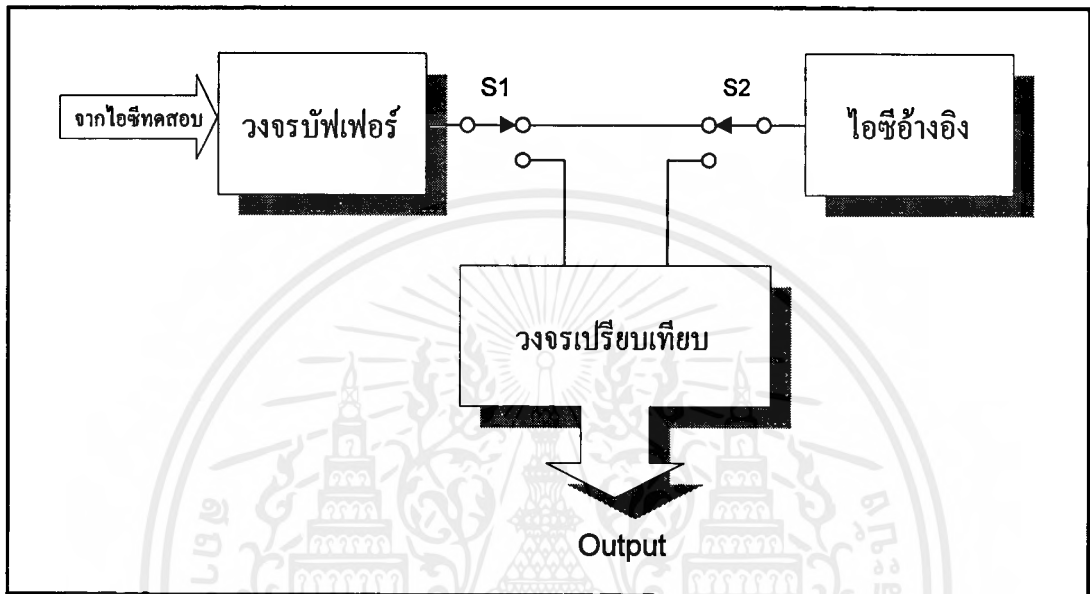
### 3.4 วงจรจัดขาอินพุตเอาต์พุต

เนื่องจากไอซีในแคบอร์มีการจัดขาอินพุตและขาเอาต์พุตที่ไม่เหมือนด้วยเหตุนี้ จึงจำเป็นต้องมีส่วนที่จะมาจัดการเกี่ยวกับการจัดขาอินพุตและขาเอาต์พุต ให้เหมาะสมกับไอซีเบอร์นั้น ๆ เพื่อจะได้นำไปใช้เป็นอินพุตที่ป้อนให้กับวงจรเปรียบเทียบได้อย่างถูกต้อง

ในเบื้องต้นของการออกแบบเราสามารถพิจารณาจากการใช้สวิตช์ธรรมดา ก่อน โดยจะใช้สวิตช์ธรรมดาเป็นตัวเลือกขาของไอซีทดสอบหรือไอซีที่ต้องการตรวจสอบว่าเป็นขาอินพุต หรือขาเอาต์พุต ให้ลองต่อแบบสองทางก่อน โดยพิจารณาสัญญาณที่ออกมาจากทั้งไอซีทดสอบและไอซีอ้างอิงเพียงขาเดียวก่อน สายสัญญาณที่จะต้องต่อมีดังต่อไปนี้

1. จากไอซีทดสอบไปที่ไอซีอ้างอิง
2. จากไอซีทดสอบไปที่วงจรเปรียบเทียบ
3. จากไอซีอ้างอิงไปที่วงจรเปรียบเทียบ

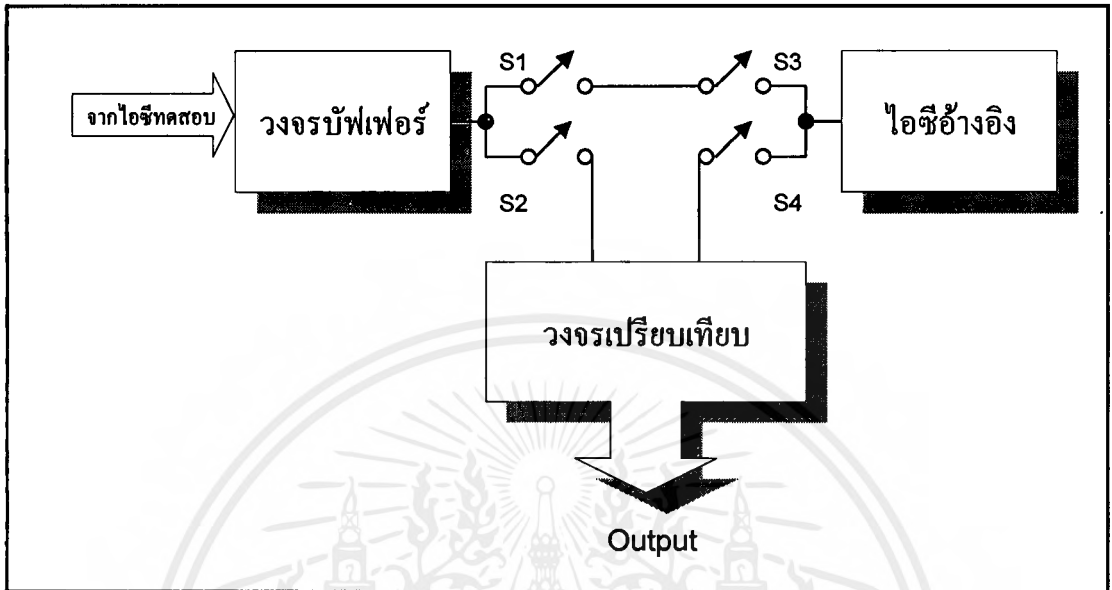
จะเห็นได้ว่าจากไอซีทดสอบจะมี 2 ทางเลือก และจากไอซีอ้างอิงก็มี 2 ทางเลือกเช่นกัน ดังนั้นจึงต้องใช้สวิทช์ 2 ตัวเลือกให้ไอซีทดสอบและไอซีอ้างอิงแสดงคังรูปที่ 3.4



รูปที่ 3.4 แสดงวิธีการต่อสวิทช์ 2 ทาง ตามแนวความคิด

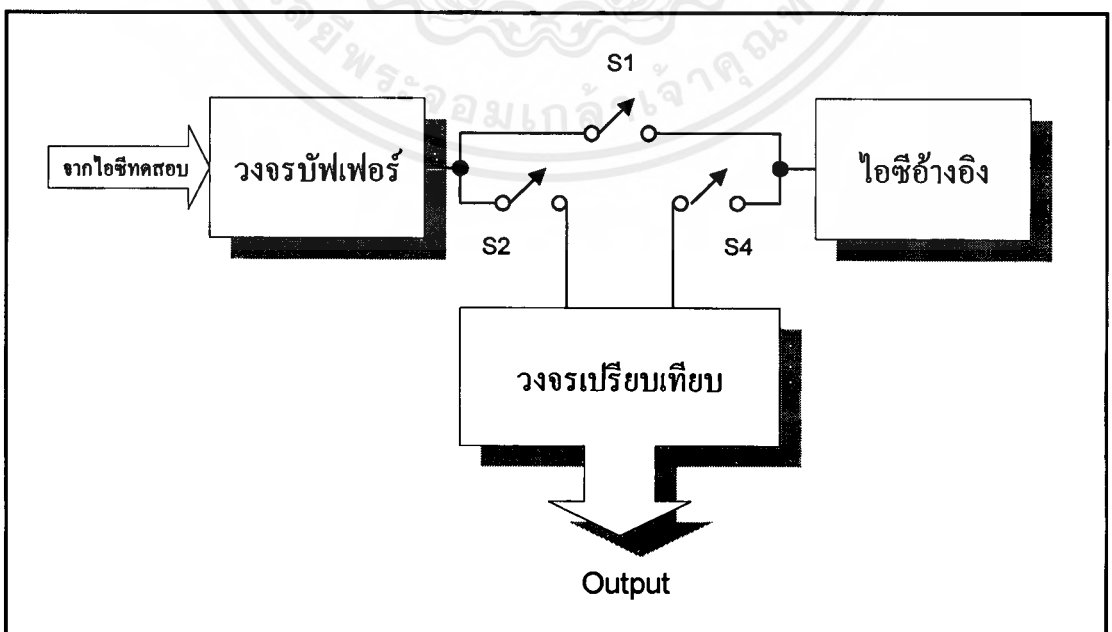
หากขาไอซีที่ทดสอบเป็นอินพุตจะต้องสับสวิทช์ทั้ง 2 ขึ้นด้านบน เพื่อให้สัญญาณจากไอซีทดสอบต่อไปเป็นอินพุตของไอซีอ้างอิง และถ้าหากขาไอซีที่ทดสอบเป็นเอาต์พุต จะต้องสับสวิทช์ทั้ง 2 ลง เพื่อนำเอาต์พุตของทั้งไอซีทดสอบกับไอซีอ้างอิงมาเปรียบเทียบกันได้

จากรูปที่ 3.4 เมื่อเปลี่ยนจากสวิทช์ธรรมดาเป็นคิพสวิทช์ซึ่งเป็นสวิทช์ทางเดียว โดยแบ่งสวิทช์ธรรมดาออกเป็น 2 ตัว และใช้คิพสวิทช์ต่อเข้าไปแทนจะได้คังรูปที่ 3.5



รูปที่ 3.5 แสดงวิธีการต่อคิฟสวิทช์แทนสวิทช์ 2 ทาง

จากรูปที่ 3.5 จะเห็นว่าคิฟสวิทช์ S1 และ S3 ต่ออนุกรมกันอยู่ ดังนั้นจึงสามารถตัดสวิทช์ออกได้ 1 ตัว การทำงานก็ยังคงเหมือนเดิมและเป็นการประหยัดสวิทช์ไป 1 ตัว ด้วย เขียนวงจรใหม่ได้ดังรูปที่ 3.6



รูปที่ 3.6 แสดงวิธีการต่อคิฟสวิทช์ เพียง 3 ตัว แทนวงจรในรูปที่ 3.5

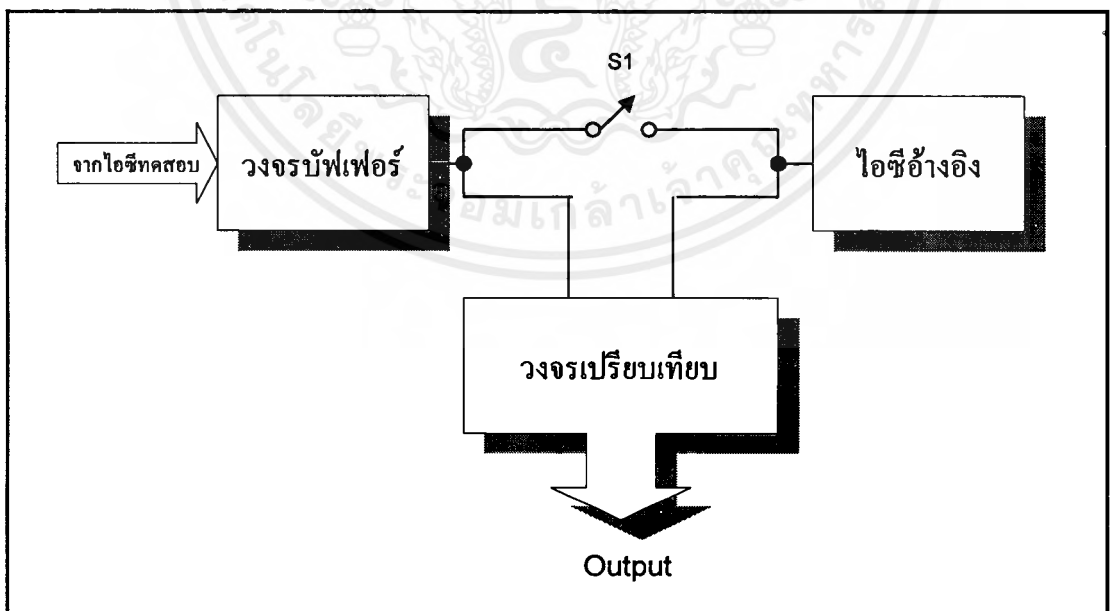


จากรูปที่ 3.6 ดิฟสวิทช์ทั้ง 3 ตัวนี้จะใช้กำหนดขาของไอซีทดสอบแต่ละขา ว่าเป็นอินพุต หรือเอาต์พุต โดยหากขาไอซีทดสอบเป็นอินพุต S1 จะต้องต่อสัญญาณ ส่วน S2 และ S4 จะต้องตัด สัญญาณออกและถ้าหากให้ขาไอซีทดสอบเป็นขาเอาต์พุต S1 จะต้องตัดสัญญาณออก ส่วน S2 และ S4 จะต้องต่อสัญญาณซึ่งการทำงานจะเหมือนกับรูปที่ 3.4

แต่ก็เกิดปัญหาว่าในขณะที่ตัด S2 และ S4 ขาอินพุตเปรียบเทียบของวงจรเปรียบเทียบ ทั้ง จะถูกต่อลอยไว้ ทำให้เอาต์พุตผิดพลาดได้หากมีสัญญาณรบกวนเข้ามา และเมื่อพิจารณาแล้ว ก็เห็นว่า หากต่อสวิทช์ทั้ง 2 ไว้ตลอด วงจรจะทำงานได้หรือไม่ ปรากฏว่าวงจรทำงานได้คือ

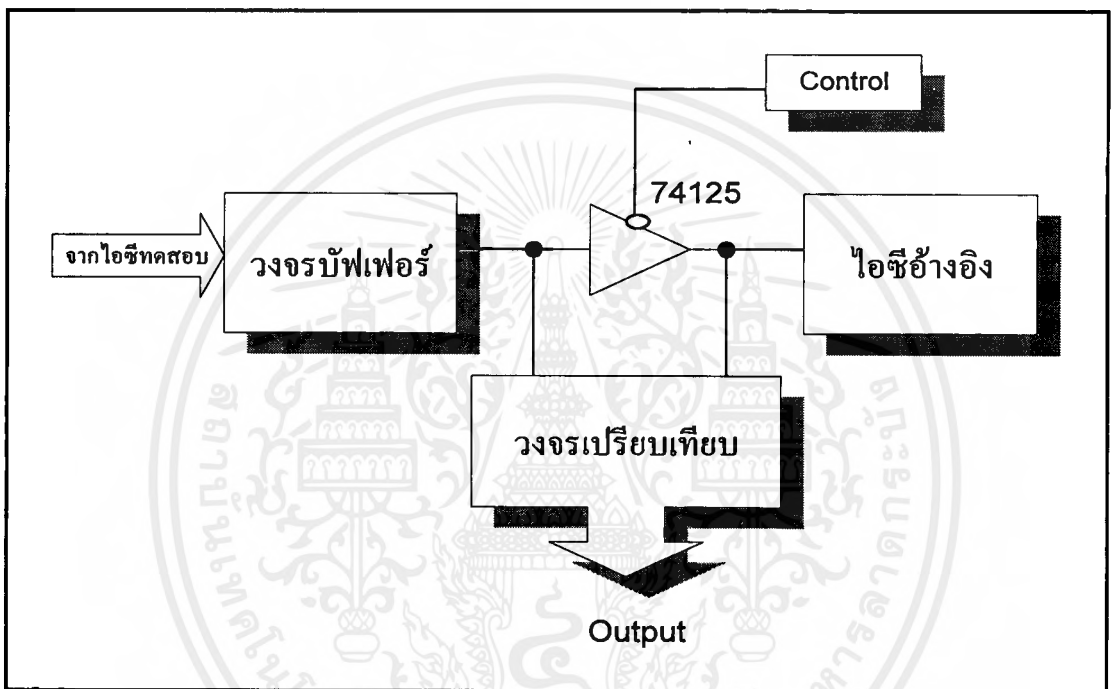
- เมื่อให้ขาไอซีทดสอบเป็นขาอินพุต S1 จะต่อสัญญาณ และขาเปรียบเทียบทั้ง 2 ก็ยังคง ต่ออยู่ วงจรเปรียบเทียบจะทำการเปรียบเทียบสัญญาณอินพุตของไอซีทดสอบกับ ไอซีอ้างอิง
- เมื่อให้ขาของ ไอซีทดสอบเป็นเอาต์พุต S1 จะตัดสัญญาณ วงจรเปรียบเทียบก็จะทำการ เปรียบเทียบสัญญาณเอาต์พุตของ ไอซีทดสอบกับ ไอซีอ้างอิง

ซึ่งวงจรก็ จะทำงานปกติ ดังนั้นจึงไม่มีความจำเป็นต้องใช้ S2 และ S4 และเมื่อเขียนวงจร ใหม่จะได้ดังรูปที่ 3.7



รูปที่ 3.7 แสดงวงจรเมื่อตัดสวิทช์ SW2 และ SW4 ตามรูปที่ 3.6 ออก

จากรูปที่ 3.7 นั้น เป็นการใช้คิฟสวิทช์เข้ามาเป็นตัวเลือกว่าเอาไอซีทดสอบเป็นขาอินพุตหรือขาเอาต์พุต ซึ่งทำให้การใช้งานค่อนข้างลำบาก และไม่สะดวกเนื่องจกัจำนวนคิฟสวิทช์ที่นำมาใช้เป็นจำนวนมาก จึงได้นำเอาไอซี 3 State Buffer มาใช้แทนคิฟสวิทช์ทั้งหมด คือ ไอซีเบอร์ 74125 แล้วควบคุมสถานะในการทำงานของไอซีเบอร์ดังกล่าวด้วยไมโครคอนโทรลเลอร์ ทำให้การใช้งานสะดวกยิ่งขึ้นดังรูปที่ 3.8 ซึ่งเป็นการแสดงการนำไอซีเบอร์ 74125 มาแทนคิฟสวิทช์



รูปที่ 3.8 แสดงการนำเอาไอซี 74125 มาใช้งานแทนคิฟสวิทช์

จากรูปที่ 3.8 สามารถสรุปการControl 74125 ได้ดังนี้

เงื่อนไขของขาไอซี	สถานะ Logic ที่ขั้วControl ของ 74125
กรณีขาอินพุต	Active Lo
กรณีขาเอาต์พุต	Active Hi
กรณีขา NC	Active Lo
กรณีขา ไฟเลี้ยง Vcc , Gnd	Active Hi

ตารางที่ 3.1 ตารางสรุปสถานะทาง Logic ที่ใช้ควบคุมการทำงานของ 74125

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

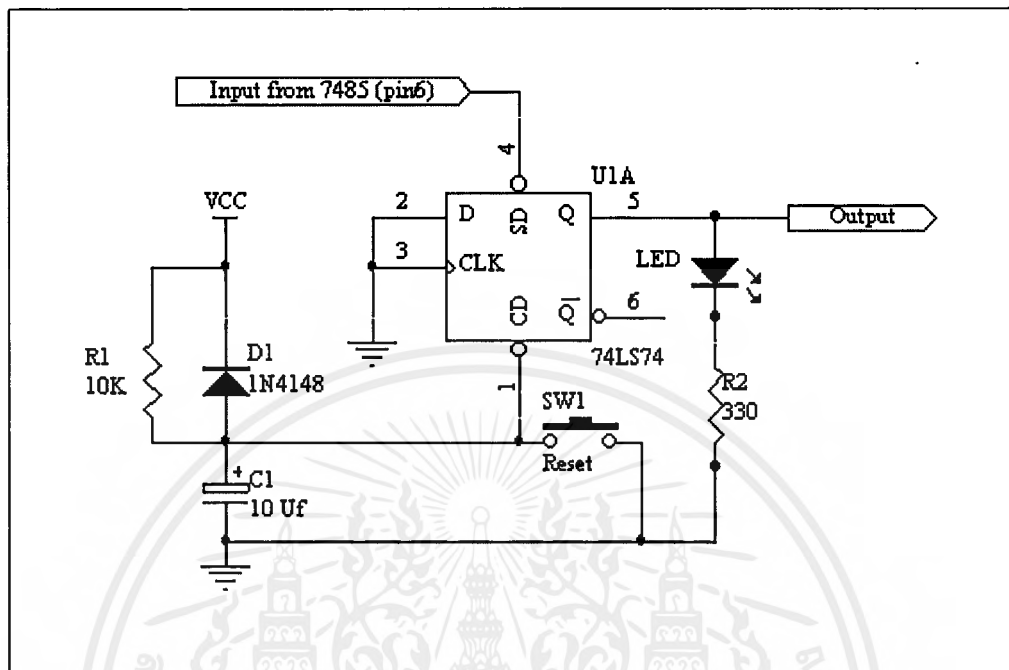
การจัดการเกี่ยวกับการควบคุมขา Control ของไอซีเบอร์ 74125 นั้น จะถูกควบคุมโดย ไมโครคอนโทรลเลอร์ โดยไมโครคอนโทรลเลอร์จะมีข้อมูลเกี่ยวกับขาอินพุตเอาต์พุตของแต่ละเบอร์ที่เก็บไว้ในหน่วยความจำ เมื่อเราป้อนไอซีเบอร์นั้น ๆ ไมโครคอนโทรลเลอร์ ก็จะทำการ Set ขา Control ของ 74125 ให้เป็นไปอย่างเหมาะสม

ในส่วนของไฟเลี้ยงที่จะใช้จ่ายให้กับไอซีอ้างอิงนั้นสิ่งสำคัญก็คือต้องให้จ่ายพร้อม ๆ กันกับไอซีทดสอบ เพื่อที่จะให้ทั้ง ไอซีอ้างอิงและไอซีทดสอบนั้น มีจุดเริ่มต้นการทำงานที่พร้อมกัน ๆ หรือเริ่มทำงานที่จุดเดียวกัน จากการรับไฟเลี้ยงพร้อม ๆ กัน และได้รับสัญญาณอินพุตต่าง ๆ เช่น สัญญาณ Reset พร้อม ๆ กันนั้นเป็นสิ่งจำเป็นมาก สำหรับการตรวจสอบไอซี จำพวก Counter หรือ Flip-Flop ถ้าหากทั้ง ไอซีอ้างอิงและไอซีทดสอบมีจุดเริ่มต้นการทำงานที่ไม่พร้อมกันแล้ว ก็จะทำให้ผลการตรวจสอบนั้นผิดพลาดได้

### 3.5 วงจรเอาต์พุต

จากเอาต์พุตของวงจรเปรียบเทียบ หากอินพุตเปรียบเทียบทั้ง 2 ไม่เท่ากัน ที่ขาเอาต์พุต (ขา6)  $A=B$  จะเป็นลอจิก “0” แต่ถ้าหากอินพุตทั้ง 2 เท่ากัน ที่ขาเอาต์พุต (ขา6)  $A=B$  จะเป็นลอจิก “1” ดังนั้นจะต้องใช้วงจรที่ตรวจจับสัญญาณที่เป็นลอจิก “0” ไม่ว่าจะสัญญาณนั้นจะมีขนาดพัลส์ที่เล็กเพียงใด และหากตรวจพบก็จะคงสถานะนั้นไว้จนกว่าจะมีการรีเซ็ตใหม่ สำหรับวงจรนี้จะใช้ ดี-ฟลิปฟล็อป เบอร์ 7474 โดยไอซีเบอร์นี้จะมีทั้งขาเคลียร์และปริเซต ขาเหล่านี้จะมีความสามารถคงสถานะได้คือ หากให้ลอจิก “0” เข้าไปที่ขาเคลียร์ ที่เอาต์พุตขา Q จะเป็นลอจิก “0” และเมื่อให้ขาเคลียร์กลับมาเป็นลอจิก “1” ขา Q ก็ยังคงสถานะเดิม ขาปริเซตก็ทำงานในลักษณะเดียวกันเพียงแต่ที่เอาต์พุตจะมีลอจิกกลับกัน ดังนั้นเราจะต่อขาเอาต์พุตของวงจรเปรียบเทียบเข้ากับขาปริเซต และต่อขาเคลียร์เข้ากับสวิตช์ใช้สำหรับรีเซ็ตระบบส่วนขาที่เหลือ คือขาอินพุต ดี (D) และขาอินพุตสัญญาณนาฬิกา (ck) กราวด์ ดังรูปที่ 3.9

เอาต์พุต LED จะติดก็ต่อเมื่อไอซีที่ใช้ทดสอบมีการทำงานไม่ตรงกับไอซีอ้างอิง และมันจะติดจนกว่าจะมีการรีเซ็ต

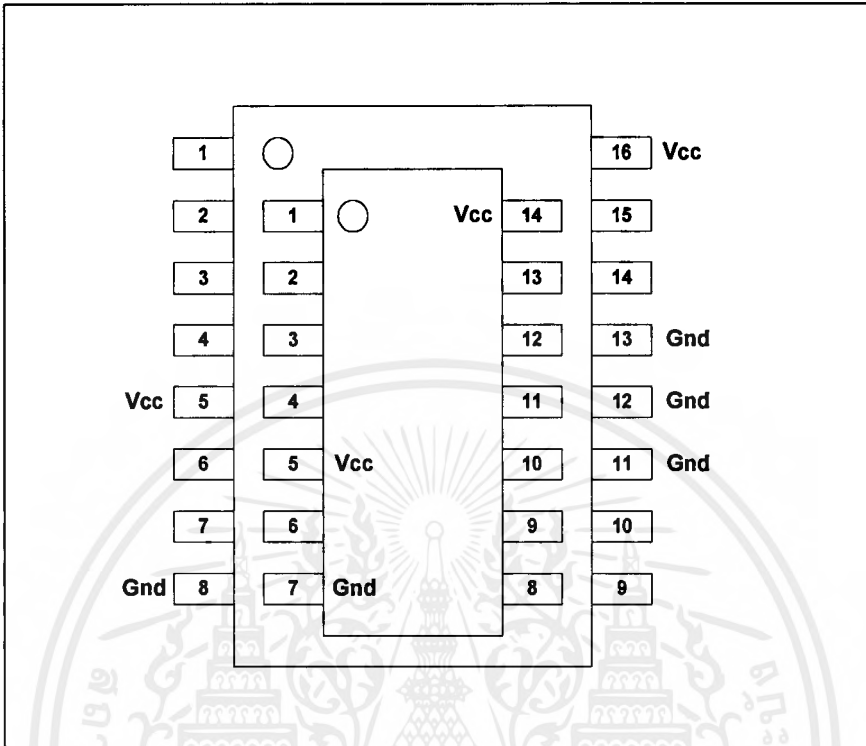


รูปที่ 3.9 แสดงวงจรเอาต์พุตใช้ 7474 ที่ต่อรับผลการเปรียบเทียบ จาก 7485

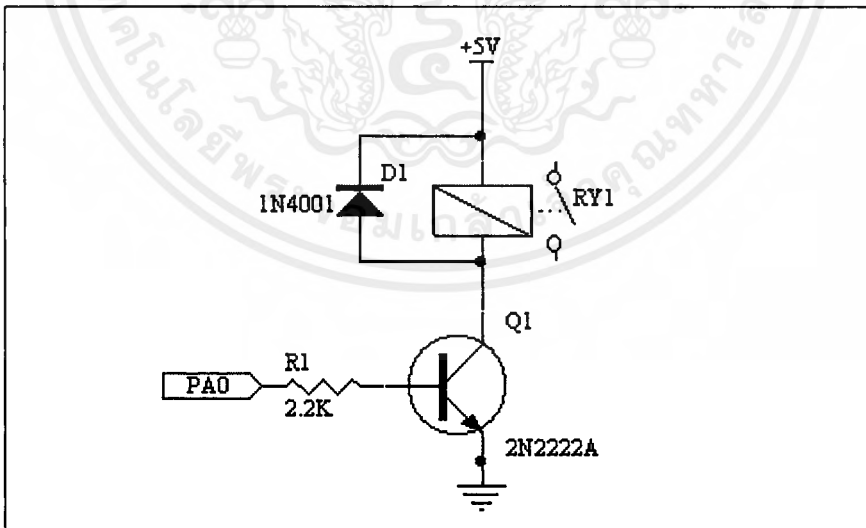
จากรูปที่ 3.9 จะเห็นว่าที่ขา 5 ของ 7474 เป็นขาเอาต์พุต โดยจะนำเอาต์พุตที่ ขา 5 ของ 7474 ไปต่อเข้าบอร์ด ไมโครคอนโทรลเลอร์ เพื่อรับสถานะไปแสดงผลการตรวจของเครื่องตรวจไอซีต่อไป

### 3.6 การจ่ายไฟเลี้ยงให้กับไอซีอ้างอิง

ในส่วนของ การจ่ายไฟเลี้ยงให้กับ ไอซีอ้างอิงนั้น เมื่อเราพิจารณาจากคู่มือ ไอซีทีทีแอลแล้ว จะพบว่าขาไฟเลี้ยงของไอซีในแต่ละเบอร์ไม่ได้มีตำแหน่งขาที่ตรงกันทุกเบอร์ ซึ่งแสดงดังรูปที่ 3.10 และจากรูปที่ 3.10 จึงต้องมีวงจรสำหรับจ่ายไฟเลี้ยงให้กับไอซีอ้างอิงให้เหมาะสมตามเบอร์ ไอซีที่นำมาใช้งานดังแสดงในรูปที่ 3.11 ซึ่งเป็นวงจรที่จ่ายไฟเลี้ยงต่อหนึ่งขา ไอซีเท่านั้น เมื่อใช้งานจริงต้องมีวงจрдังรูปที่ 3.11 ทั้งหมด 8 ชุด เพื่อที่จะจ่ายไฟเลี้ยงได้ครบตามรูปที่ 3.10 จากรูปที่ 3.11 นั้น การควบคุมจะใช้ Prot A ของ 8255 มาควบคุมการทำงานและไฟเลี้ยงที่จ่ายให้กับไอซีอ้างอิง จะให้ไฟเลี้ยงชุดเดียวกันกับที่จ่ายให้กับไอซีที่จะใช้ตรวจสอบ เพื่อที่จะกำหนดให้มีจุดเริ่มต้นการทำงานที่พร้อมกัน



รูปที่ 3.10 แสดงตำแหน่ง ขา V<sub>cc</sub> และขา Gnd ของไอซีทีทีแอลที่ปรากฏในกลุ่มนี้

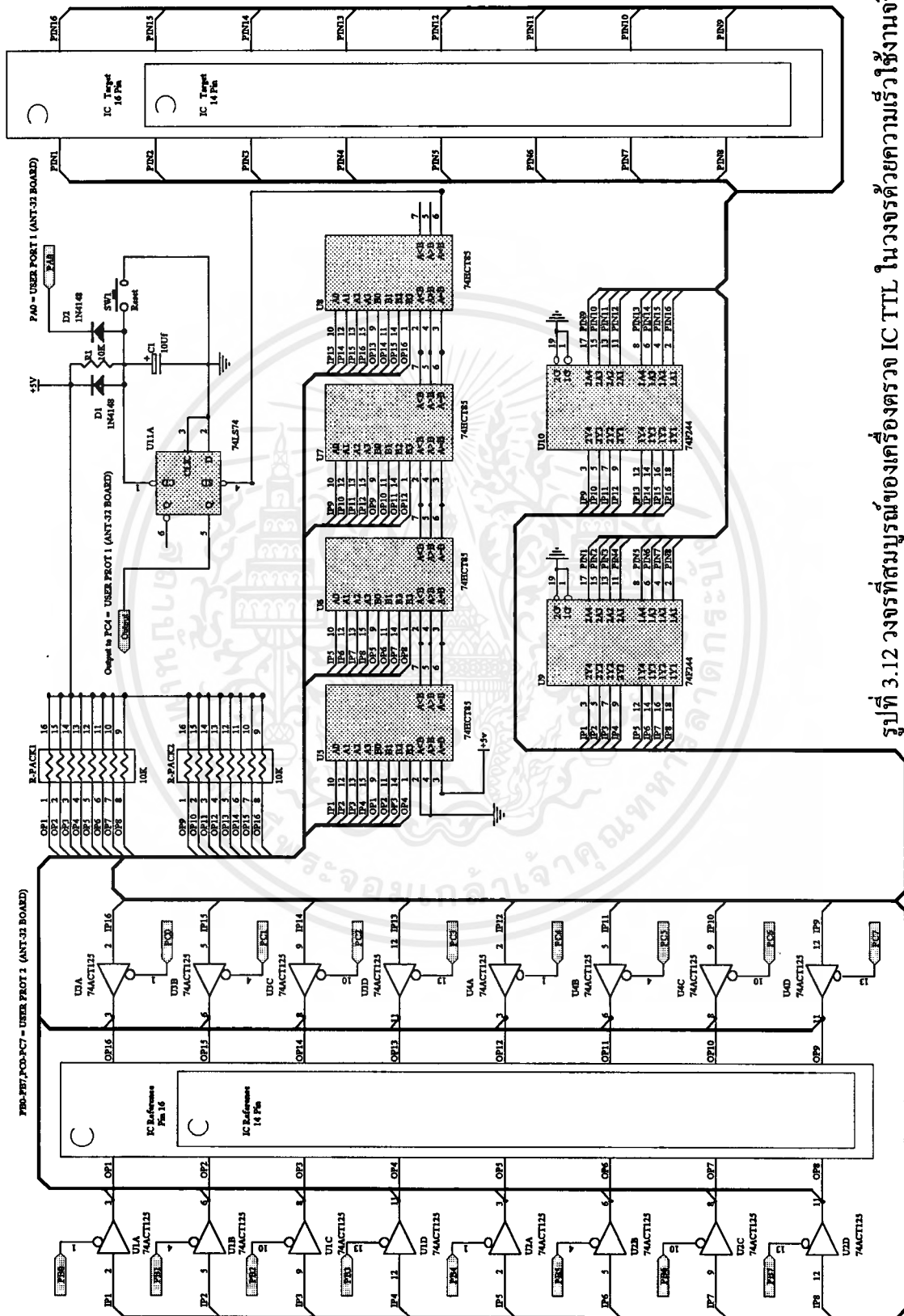


รูปที่ 3.11 แสดงวงจรสำหรับจ่ายไฟเลี้ยงให้ไอซีอ้างอิงต่อหนึ่งขาไอซี

### 3.7 วงจรเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง

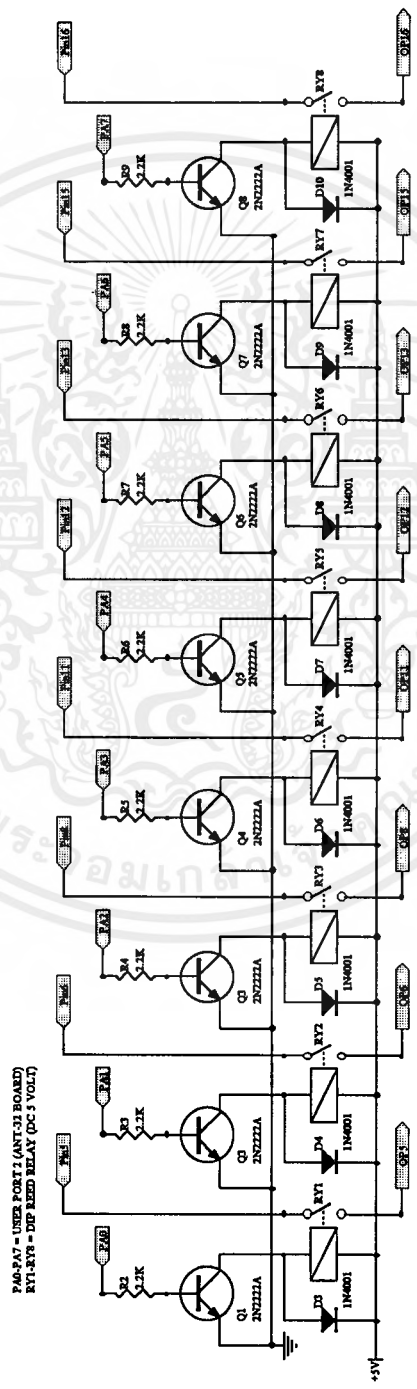
ในหัวข้อนี้คือการนำเอาวงจรในแต่ละส่วนที่ได้อธิบายในแต่ละหัวข้อที่ผ่านมา เพื่อรวมเข้าด้วยกันเป็นวงจรที่สมบูรณ์และสามารถใช้งานจริงได้ ดังรูปที่ 3.12 และรูปที่ 3.13

จากรูปที่ 3.12 จะเห็นว่าที่ตัวไอซี 7474 นั้น สามารถรีเซ็ตได้จาก 3 ทาง คือ 1. RC รีเซ็ต 2. สวิตช์ กดติด-ปล่อยดับ รีเซ็ต และ 3. รีเซ็ตโดยรับสัญญาณมาจาก 8255 User Port 1 ในบอร์ด ANT-32 ซึ่งในส่วนที่ 3 นี้ จะรีเซ็ตโดย การเขียนโปรแกรมมาควบคุมการรีเซ็ต ส่วนการควบคุมการทำงานของ ตัวไอซีเบอร์ 74125 นั้น จะใช้ 8255 User Port 2 คือ Port B (PB0-PB7) และ Port C (PC0-PC7) มาควบคุมการทำงาน ซึ่งในการควบคุมนั้น ตัว 8255 User Port 2 จะทำการเซตสถานะทางลอจิกให้เป็นไปอย่างเหมาะสมตามไอซีที่นำมาใช้เป็นไอซีอ้างอิง เนื่องจากโปรแกรมมี Library ของไอซีแต่ละเบอร์อยู่ (สูงสุดไม่เกิน 16 ขา) เมื่อเราป้อนเบอร์ไอซีที่ต้องการ โปรแกรม ก็จะสั่งงานให้ 8255 ทำการเซตสถานะตามไอซีเบอร์นั้นได้อย่างถูกต้อง ว่าขาใดเป็นขาอินพุตหรือขาใดเป็นขาเอาต์พุต และส่วนของ รูปที่ 3.13 นั้น เป็นส่วนของวงจรที่ทำหน้าที่จ่ายไฟเลี้ยงให้กับไอซีอ้างอิง โดยใช้ Dip Reed Relay มีตัวโครงสร้างคล้ายกับ ไอซี ขนาด 14 ขา ซึ่งในการทำงานในแต่ละครั้ง จะมีรีเลย์เพียง 2 ตัว เท่านั้นที่ทำงาน เพราะจ่าย +5V กับ Gnd เพียง 2 ขา ในไอซีในแต่ละเบอร์ การควบคุมการทำงานของวงจรในรูปที่ 3.13 ก็จะใช้ Port A User Port 2 (PA0-PA7) มาควบคุมการทำงานซึ่งก็เช่นเดียวกัน ก็จะมี Library ของตำแหน่งของขาไฟเลี้ยงไอซีในแต่ละเบอร์ไว้อยู่ในโปรแกรม เมื่อป้อนเบอร์ไอซี โปรแกรมก็จะสั่งงานให้ 8255 ทำการเซตการทำงานของวงจรในรูปที่ 3.13 ให้ ทราานซิสเตอร์นำกระแสให้รีเลย์ทำงานได้อย่างเหมาะสม



รูปที่ 3.12 วงจรที่สมบูรณ์ของเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

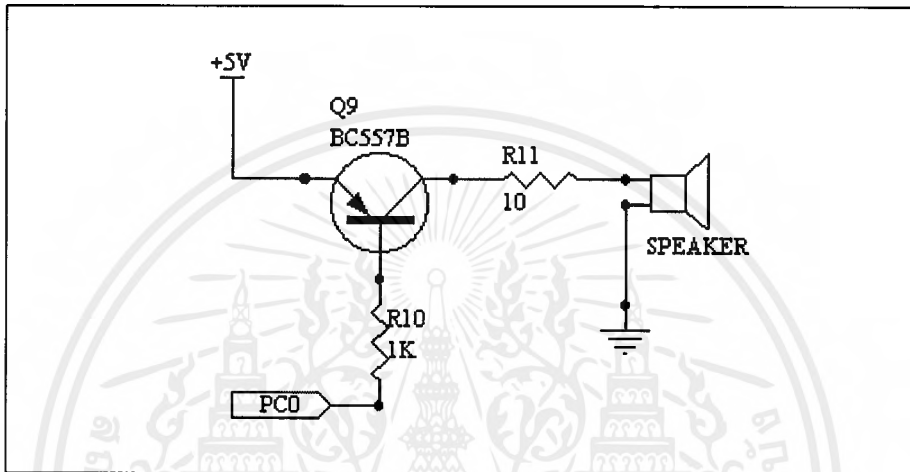


รูปที่ 3.13 วงจรที่สมรรถนะของวงจรจ่ายไฟเลี้ยงให้กับไอซีอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.8 วงจรกำเนิดเสียงกดคีย์

วงจรกำเนิดเสียงกดคีย์จะใช้ลำโพงขนาดเล็กและใช้ทรานซิสเตอร์มาขับลำโพง ให้กำเนิดเสียงที่ความถี่ต่าง ๆ ดังแสดงในรูปที่ 3.14 และการควบคุมการทำงานงานของทรานซิสเตอร์ นั้นจะใช้ 8255 Port C (PC0) User Port 1 จากบอร์ด ANT-32 มาควบคุมการทำงาน โดยเขียน โปรแกรมให้กำเนิดเสียงตามการกดคีย์หรือเงื่อนไขการทำงานอื่นๆ ของโปรแกรม



รูปที่ 3.14 วงจรกำเนิดเสียงกดคีย์

### 3.9 การต่อใช้งานคีย์บอร์ด

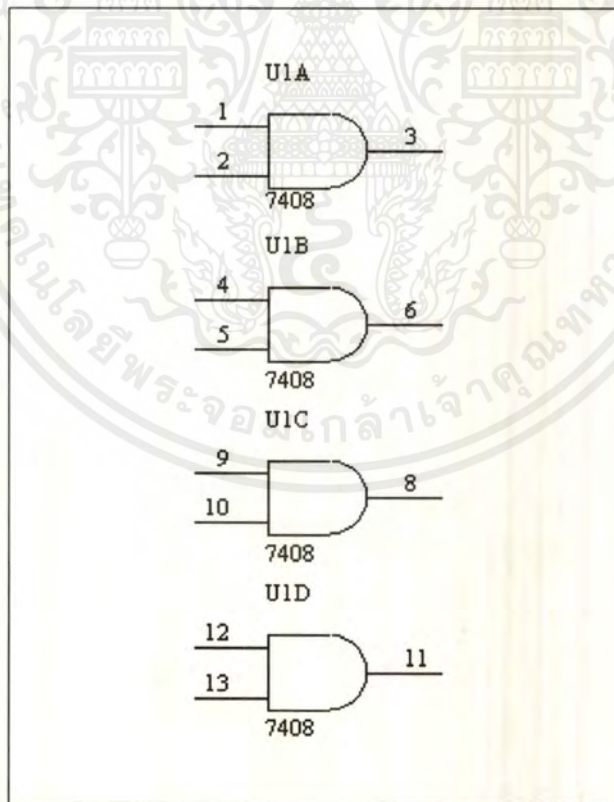
จะเป็นวงจรคีย์บอร์ดขนาด 4x4 (16 คีย์) แล้วต่อเข้ากับ Port 1 ของ MCS-51 เพื่อรับ Data ที่กดว่าคีย์ใดแล้วนำ Data ที่ได้ไปใช้งานต่อไป

## บทที่ 4

### การเขียนโปรแกรมควบคุมการทำงานของเครื่องตรวจ IC TTL ในวงจร

#### 4.1 แนวทางการเริ่มเขียนโปรแกรมควบคุมการทำงานของเครื่องตรวจ IC TTL ในวงจร

ในการใช้ไมโครคอนโทรลเลอร์มาควบคุมการทำงานของวงจรเครื่องตรวจไอซี นั้นจะใช้บอร์ด ANT-32 มาควบคุมการทำงาน และจากการทำงานของวงจรเครื่องตรวจไอซี ในบทที่ 3 นั้นจะใช้ 8255 Port โดยกำหนดให้เป็น Output Port ที่ทำหน้าที่ควบคุมการทำงานของวงจรอยู่ 3 Port คือ Port A , Port B , Port C (User Port 2) โดยกำหนดให้ Port A ควบคุมการจ่ายไฟเลี้ยงให้กับไอซีอ้างอิงและ Port B กับ Port C จะใช้ควบคุมการทำงานของ 74125 ว่าขาของไอซีอ้างอิงนั้นขาใดเป็นขาอินพุตหรือขาเอาต์พุต ถ้าเราเริ่มพิจารณาจากไอซีที่มีจำนวนขา 14 ขาก่อน โดยนำเอาไอซีเบอร์ 7408 ซึ่งเป็น AND Gate 2 Input มาเป็นตัวอย่างในการพิจารณา ดังรูปที่ 4.1



รูปที่ 4.1 ไอซีทีทีแอลเบอร์ 7408

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.1 จะเห็นได้ว่ามีขาที่เป็นขาอินพุตคือขา 1, 2, 4, 5, 9, 10, 12, 13 ส่วนขาที่เป็นขาเอาต์พุตคือขา 3, 6, 8, 11 และขาที่เป็นไฟเลี้ยงคือ ขา14 (+5V) และ ขา7 (Gnd) เมื่อนำเขียนเป็นตารางแสดงการควบคุมได้ดังตารางที่ 4.1

PORT	PB 0	P B1	P B2	P B3	P B4	PB 5	PB 6	PB 7	PC 7	PC 6	PC 5	PC 4	PC 3	PC 2	PC 1	PC 0
14 Pin	X	1	2	3	4	5	6	7	8	9	10	11	12	13	14	X
16 Pin	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
7408	0	0	0	1	0	0	1	1	1	0	0	1	0	0	1	0

ตารางที่ 4.1 แสดงการควบคุม 74125 โดยใช้กับไอซีที่มี 14 ขา

จากรูปที่ 4.1 เงื่อนไขในการควบคุมก็คือ ขาที่เป็นขาอินพุตจะเป็นลอจิก “0” ส่วนขาที่เป็นขาเอาต์พุตจะเป็นลอจิก “1” และขาที่เป็นไฟเลี้ยง คือ +5 V กับ Gnd จะเป็นลอจิก “1” ส่วนขาที่ถูกปล่อยลอย เนื่องจากเป็นไอซี 14 ขา คือ PB0 กับ PC0 ก็จะถูกเซตให้เป็นลอจิก “0” เช่นเดียวกันกับกรณีที่เป็นขาอินพุต ลอจิกในแต่ละบิตของ Port B และ Port C ก็จะแตกต่างกันไปตามไอซีในแต่ละเบอร์ ซึ่งพอจะเห็นแนวทางแล้วว่า เมื่อเรารวบรวมเอาข้อมูลของไอซีในแต่ละเบอร์มาเขียนเป็นตาราง แบบตารางที่ 4.1 ก็จะได้ Table หรือ Library ของไอซีในแต่ละเบอร์ จากนั้นก็สามารถเขียนโปรแกรมเพื่อเปิดตาราง หลังจากที่ทำการป้อนเบอร์ไอซีแล้ว ก็ให้ Output Port 8255 (PB,PC) มีลอจิกในแต่ละบิตตามตารางที่ได้รวบรวมมาแล้ว

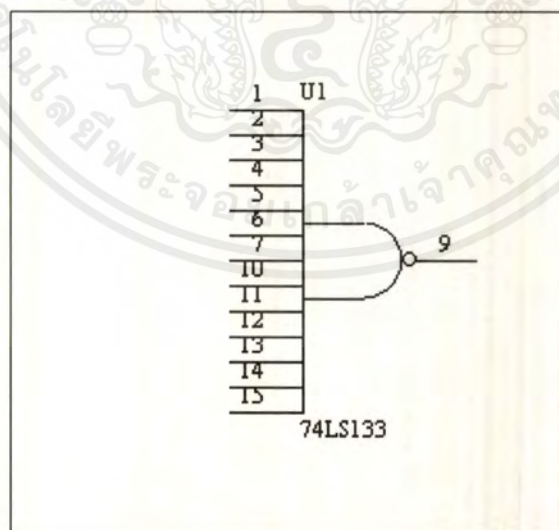
ในการทำงานเดียวกัน Port A ซึ่งเป็น Port ที่ใช้ควบคุมการจ่ายไฟเลี้ยง ก็สามารถนำมาเขียนเป็นตารางได้ดังตารางที่ 4.2 แต่การควบคุมนั้นเป็นการควบคุมการนำกระแสของตัวทรานซิสเตอร์ ซึ่งเป็นชนิด NPN เพราะฉะนั้นถ้าขาใดเป็นขาไฟเลี้ยงก็จะเซตให้ Port A ในบิตนั้นเป็นลอจิก “1” เพื่อให้ทรานซิสเตอร์สามารถนำกระแสไปเลี้ยงรีเลย์ให้ทำงานได้ เพื่อจ่ายไฟเลี้ยงให้กับไอซีอ้างอิงได้

PORT	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
16 Pin/14Pin	5/4	6/5	8/7	11/10	12/11	13/12	15/14	16
Function	Vcc	Vcc	Gnd	Gnd	Gnd	Gnd	Vcc	Vcc
Logic	0	0	1	0	0	0	1	0

ตารางที่ 4.2 แสดงการควบคุมการจ่ายไฟเลี้ยงให้กับไอซีอ้างอิง เบอร์ 7408

จากตารางที่ 4.2 จะเห็นว่ามีเพียง PA 2 กับ PA 6 เท่านั้นที่มีลอจิกเป็น “1” เนื่องจากที่บิตทั้งสองนี้จะใช้ควบคุมการจ่ายไฟเลี้ยงให้กับไอซี 7408 ที่นำมาเป็นตัวอย่างในการพิจารณาโดย ที่ขา 14 ซึ่งเป็นขา +5V และ ขา 7 ซึ่งเป็นขา Gnd

พิจารณากรณีที่เป็นไอซีแบบ 16 ขา โดยนำเอา ไอซีเบอร์ 74133 มาพิจารณา ซึ่งจากกรณีที่เป็นแบบ 14 ขา PB0 กับ PC0 ก็จะถูกเซตให้เป็นลอจิก “0” ตลอด แต่ในกรณีของ 16 ขา นี้จะมีการใช้งานไม่ได้ถูกปล่อยลอยเหมือนกรณีแบบ 14 ขา จึงมีโอกาสที่จะเป็นได้ทั้ง ลอจิก “0” และ ลอจิก “1” ดังแสดงในตารางที่ 4.3 และตารางที่ 4.4



รูปที่ 4.2 ไอซีทีทีแอล เบอร์ 74LS133

PORT	PB 0	PB 1	PB 2	PB 3	PB 4	PB 5	PB 6	PB 7	PC 7	PC 6	PC 5	PC 4	PC 3	PC 2	PC 1	PC 0
14 Pin	X	1	2	3	4	5	6	7	8	9	10	11	12	13	14	X
16 Pin	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
74LS133	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1

ตารางที่ 4.3 แสดงการควบคุม 74125 โดยใช้กับไอซีที่มี 16 ขา

PORT	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
16 Pin/14Pin	5/4	6/5	8/7	11/10	12/11	13/12	15/14	16
Function	Vcc	Vcc	Gnd	Gnd	Gnd	Gnd	Vcc	Vcc
Logic	0	0	1	0	0	0	0	1

ตารางที่ 4.4 แสดงการควบคุมการจ่ายไฟเลี้ยงให้กับไอซีอ้างอิง เบอร์ 74LS133

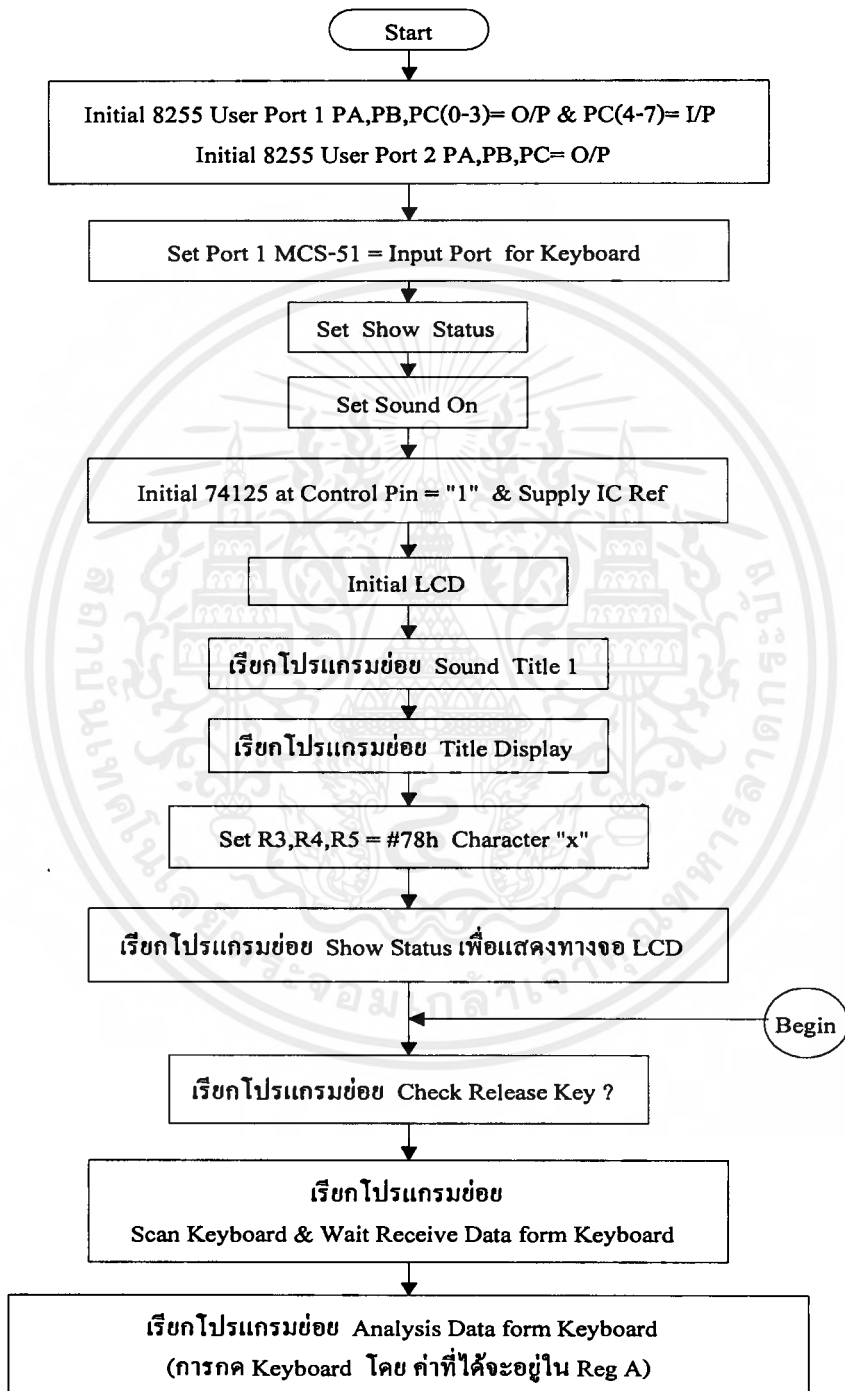
จากตัวอย่างของไอซีนามาพิจารณาทั้งกรณีแบบ 14 ขา และ กรณีแบบ 16 ขา สรุปได้ว่าการเขียนโปรแกรมที่จะใช้ควบคุมการทำงานจะประกอบไปด้วยรายละเอียดดังต่อไปนี้

- การรับค่าจากการกดคีย์บอร์ด
- แสดงผลการทำงานทางจอ LCD
- การกำเนิดเสียงการกดคีย์
- การรีเซต 7474
- การรับสถานะลอจิกจาก Output ของ 7474 ไปประมวลผลว่า ไอซีทดสอบดีหรือเสีย
- การเปิดตารางเพื่อ ควบคุม 74125 และการจ่ายไฟเลี้ยงให้ไอซีอ้างอิง
- Function พิเศษ อันได้แก่ Test LCD , Test Keyboard , Test Speaker , Sound On/Off , Restart System
- และอื่น ๆ

ซึ่งจะอธิบายดังในหัวข้อต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 การทำงานของโปรแกรมควบคุมเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง



Flowchart ที่ 4.1 แสดงการ Initial ในส่วนต่าง ๆ และรอรับการกดคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Flowchart ที่ 4.1 เมื่อเริ่ม Run โปรแกรมแล้ว จะต้องทำการ Initial ในส่วนต่าง ๆ ก่อน ดังนี้

1. ทำการ Initial 8255 ให้เป็น Input หรือ Output Port ตามหน้าที่การใช้งาน โดยกำหนดให้ User Port 2 อันได้แก่ Port A , Port B , Port C ทั้ง 3 Port ให้เป็น Output Port ซึ่ง Port A จะใช้ควบคุมการจ่ายไฟเลี้ยงให้กับไอซีอ้างอิงและ Port B กับ Port C จะใช้ควบคุมว่าขาไอซีอ้างอิงขาใดเป็นขาอินพุตหรือขาเอาต์พุต

2. เซตให้ Port 1 ของตัว CPU MCS-51 ให้เป็น Input Port เพื่อให้สามารถรับ Data จากการกดคีย์บอร์ดได้

3. เซต การ Show Status Display

4. เซตให้ Sound On เพื่อให้สามารถกำเนิดเสียง Sound Title ได้ทางลำโพง

5. ทำการ Initial ไอซี 74125 โดยให้ที่ขา Control ทุก ๆ ขา มีลอจิกเท่ากับ “1” ซึ่งทำให้เสมือนหนึ่งว่าถูกปลดออกจากวงจรอยู่และ Initial ให้ทรานซิสเตอร์ทุกตัวไม่นำกระแสไปจ่ายรีเลย์ให้ทำงานได้

6. ทำการ Initial LCD ให้แสดงผลได้ 16 ตัวอักษร 1 บรรทัด

7. เรียกโปรแกรมย่อยกำเนิดเสียง Sound Title

8. เรียกโปรแกรมย่อย Show Title Display โดยแสดงข้อความ “KMITL,Eng,Ins#17” วจิงจากขวามือมาหยุดรอสักครู่เพื่อให้อ่านแล้ววิ่งหายไปทางซ้ายมือ จากนั้นก็มีข้อความชื่อ Project คือ “ Real Time TTL In-Circuit Testing ” มาแสดงทางจอแล้ววิ่งหายไปทางซ้ายมือเช่นเดียวกัน และสุดท้ายคือ แสดง Version คือ Version 1.0 บนจอ LCD จึงจบการทำงานของโปรแกรมย่อย Show Title Display

9. ทำการ Set R3 , R4 , R5 ซึ่งเป็น Register ตัวแปรที่ใช้เก็บค่าเบอร์ไอซีที่ทำการป้อนในแต่ละหลัก โดย R3 ใช้เก็บตัวเลขในหลักร้อย , R4 ใช้เก็บตัวเลขในหลักสิบ , R5 ใช้เก็บตัวเลขในหลักหน่วย สำหรับในกรณีเบอร์ไอซีที่ทำการป้อนเป็นเบอร์ที่มี 3 หลัก ตัวอย่างเช่นเบอร์ไอซีที่ทำการป้อนคือเบอร์ 74125 ก็จะได้ค่าใน Register ต่างๆ ดังนี้ R3 = 01h , R4 = 02h , R5 = 05h หรือเบอร์ไอซีที่ทำการป้อนคือเบอร์ 7400 ก็จะได้ค่าใน Register ต่างๆ ดังนี้ R3 = 01h , R4 = 02h , R5 = 78h จะสังเกตได้ว่า R5 = 78h ซึ่งหมายถึง อักษร “x” คือส่วนที่เป็นตัวใช้บอกได้ว่าเบอร์ที่ป้อนเป็นเบอร์ไอซีที่มีเลขเพียง 2 หลัก ในส่วนของโปรแกรมย่อยนี้จะมีหน้าที่ Set ให้ Register R3 , R4 , R5 มีค่าเป็นเท่ากับ 78h ซึ่งหมายถึง อักษร “x” เพื่อแสดงให้เห็นว่าเมื่อเริ่มต้นจะยังไม่มีการป้อนไอซีเบอร์ใดและที่ LCD ใช้ค่าใน Register R3 , R4 , R5 มีค่าเป็นเท่ากับ 78h ซึ่งหมายถึง อักษร “x” มาแสดง คือ “74xxx”

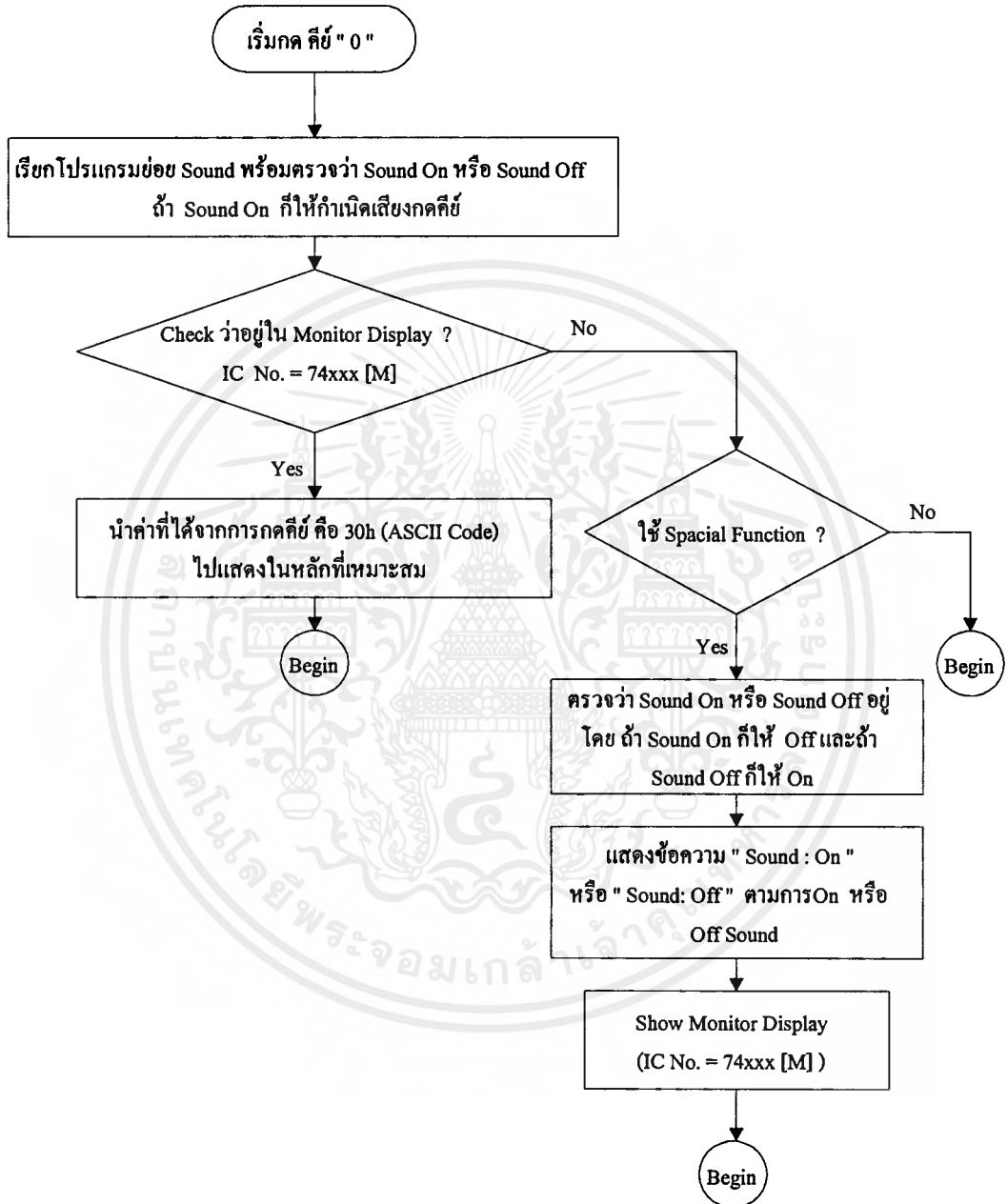
10. เรียกโปรแกรมย่อย Show Status เพื่อแสดง Status บนจอ LCD โดยจะมีเงื่อนไขการ แสดงคือจอ LCD จะแสดงข้อความ “Status: Wait” เมื่อยังไม่มีการป้อนเบอร์ไอซีและ LCD จะ แสดงข้อความ “Status: Ready” เมื่อมีการป้อนเบอร์ไอซี

11. เรียกโปรแกรมย่อยเพื่อตรวจสอบว่าปล่อยการกดคีย์แล้วหรือยัง
12. เรียกโปรแกรมย่อยเพื่อรอรับการกดคีย์บอร์ด
13. เรียกโปรแกรมย่อยเมื่อมีการกดคีย์แล้วและทำตามหน้าที่ของคีย์ที่ทำการกด

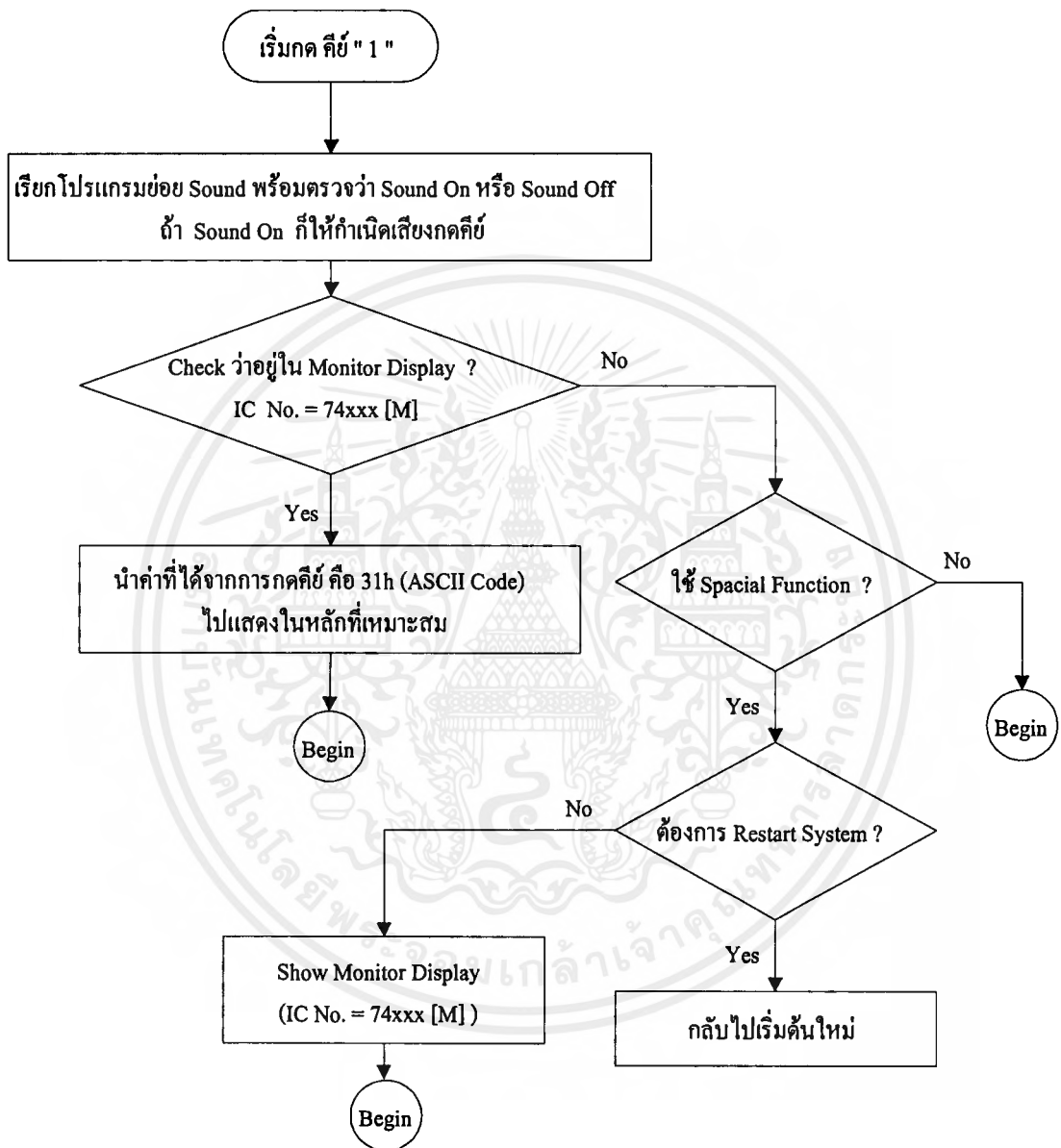
ในส่วนที่จะอธิบายต่อไปคือ Flowchart ของคีย์ต่าง ๆ ว่ามีการทำงานของโปรแกรมอย่างไร ซึ่งคีย์บอร์ดที่ใช้งานมีทั้งหมดอยู่ 16 คีย์โดยมีค่าประจำคีย์เป็นเลขฐาน 16 นั่นคือ 00h ถึง 0Fh ตัวอย่างเช่น เมื่อกดคีย์ 0 ก็จะได้ Data เท่ากับ 00h หรือกดคีย์ 9 ก็จะได้ Data เท่ากับ 09h เป็นต้น คีย์ทั้ง 16 คีย์จะประกอบไปด้วย

- |                  |                                       |
|------------------|---------------------------------------|
| 1. “0” Key (00h) | 9. “8” Key (08h)                      |
| 2. “1” Key (01h) | 10. “9” Key (09h)                     |
| 3. “2” Key (02h) | 11. ENT/YES Key (0Ah)                 |
| 4. “3” Key (03h) | 12. CLR/DEL Key (0Bh)                 |
| 5. “4” Key (04h) | 13. Replay Left Key (0Ch)             |
| 6. “5” Key (05h) | 14. MON/CAN Key (0Dh)                 |
| 7. “6” Key (06h) | 15. Replay Right / FUNCTION Key (0Eh) |
| 8. “7” Key (07h) | 16. Test Key (0Fh)                    |

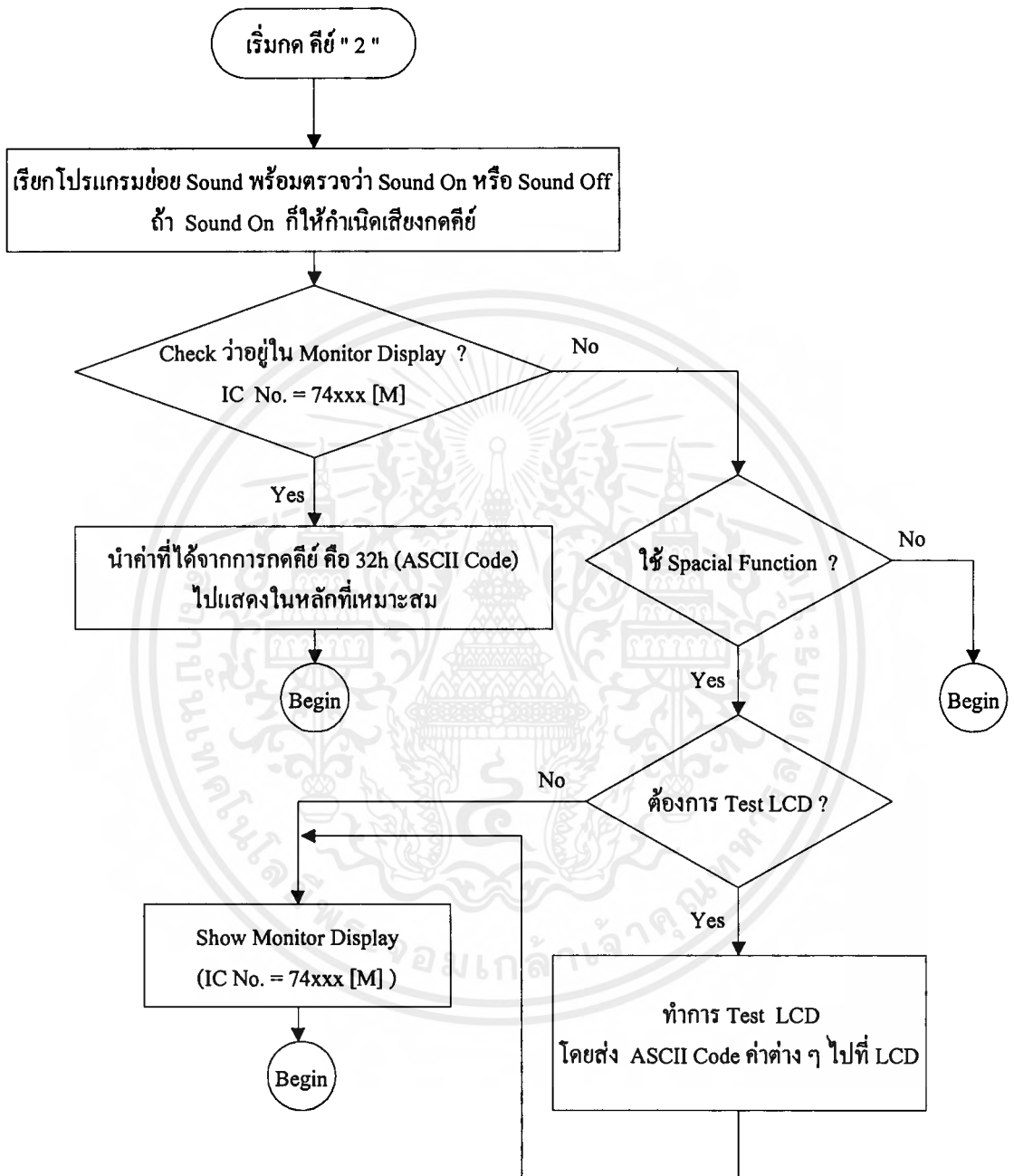
จะเห็นว่าค่าในการกดคีย์คือคีย์ “0” ถึงคีย์ “9” มีค่าคือ 00h-09h ซึ่งเวลานำไปแสดงผลทางจอ LCD นั้นต้องทำให้เป็น ASCII Code ก่อน จึงจะแสดงได้อย่างถูกต้อง เช่นจะแสดง “0” ก็จะได้เป็น 30h หรือจะแสดง “9” ก็จะได้เป็น 39h



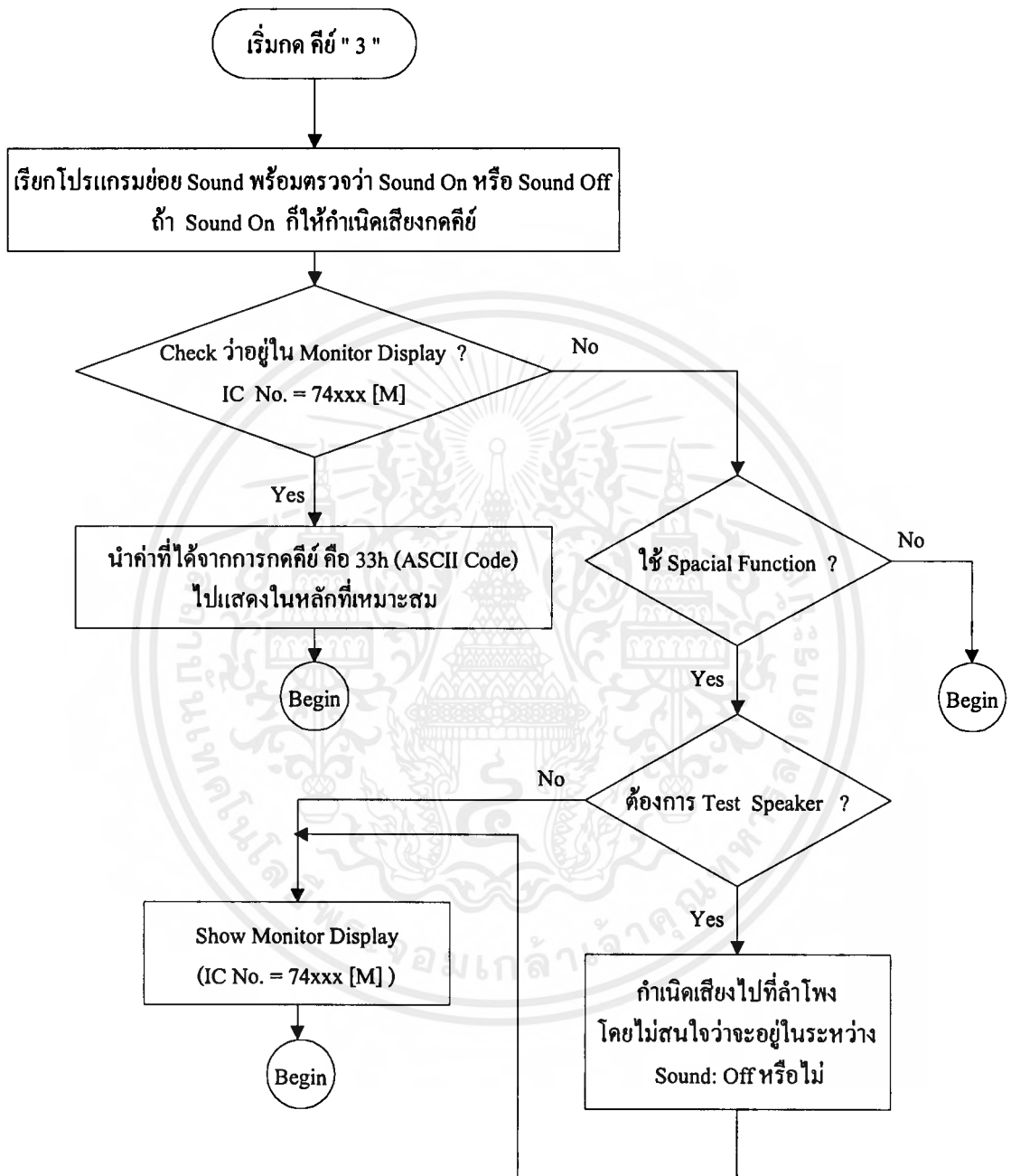
Flowchart ที่ 4.2 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "0"



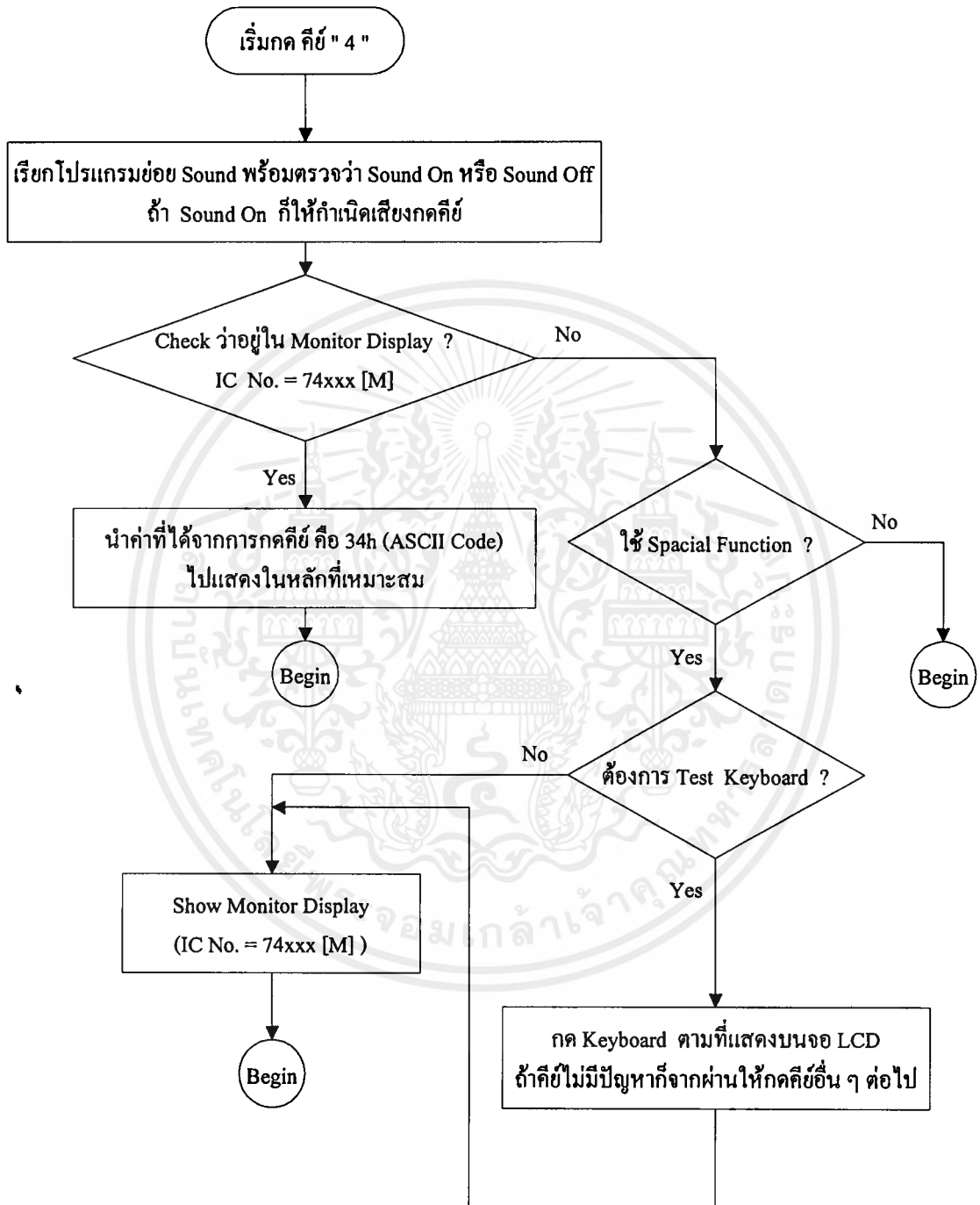
Flowchart ที่ 4.3 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "1"



Flowchart ที่ 4.4 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "2"

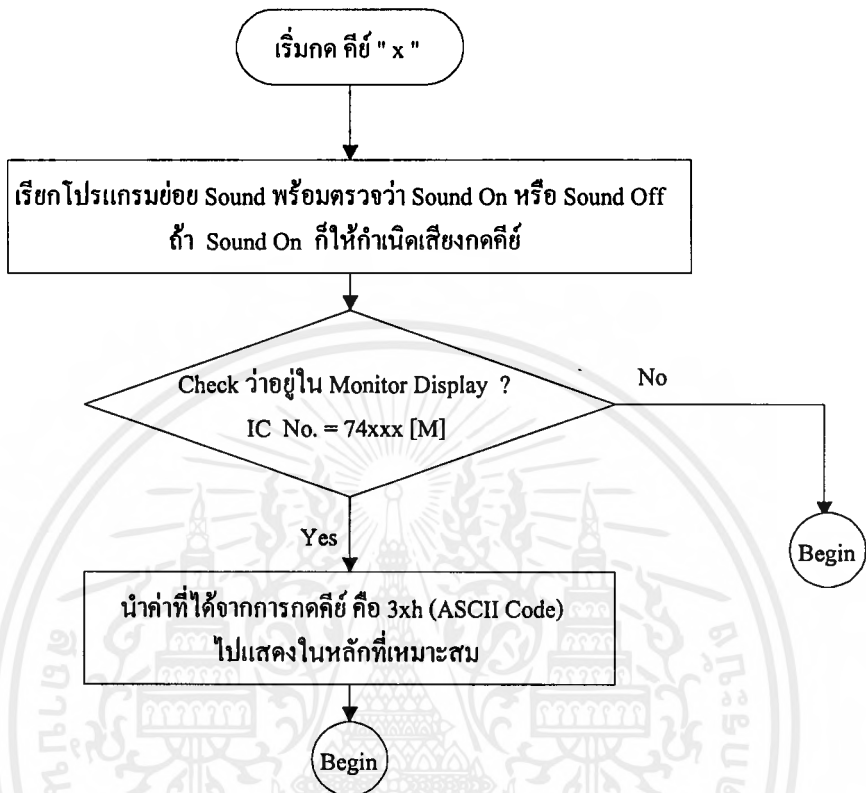


Flowchart ที่ 4.5 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “3”



Flowchart ที่ 4.6 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "4"

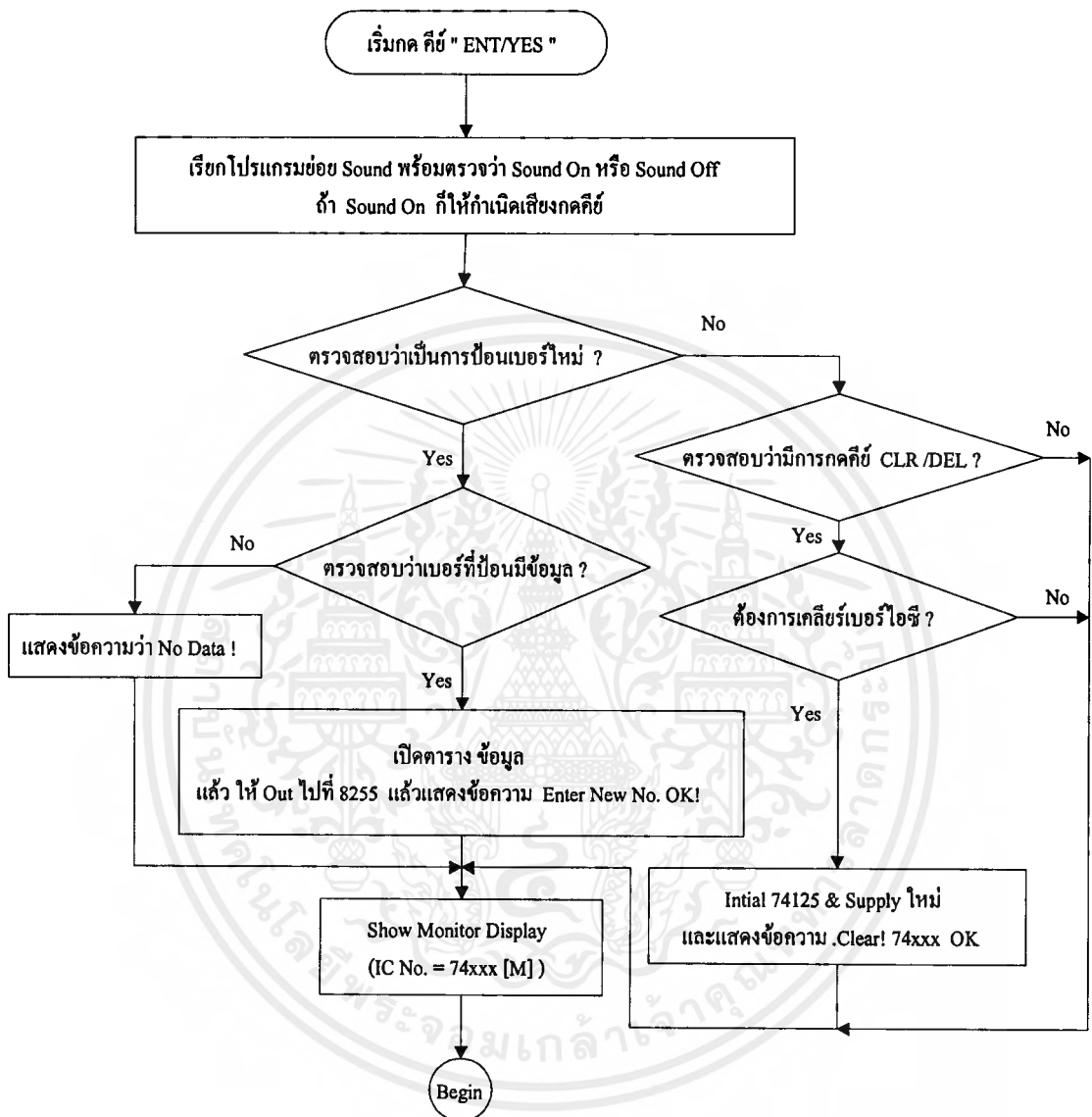
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



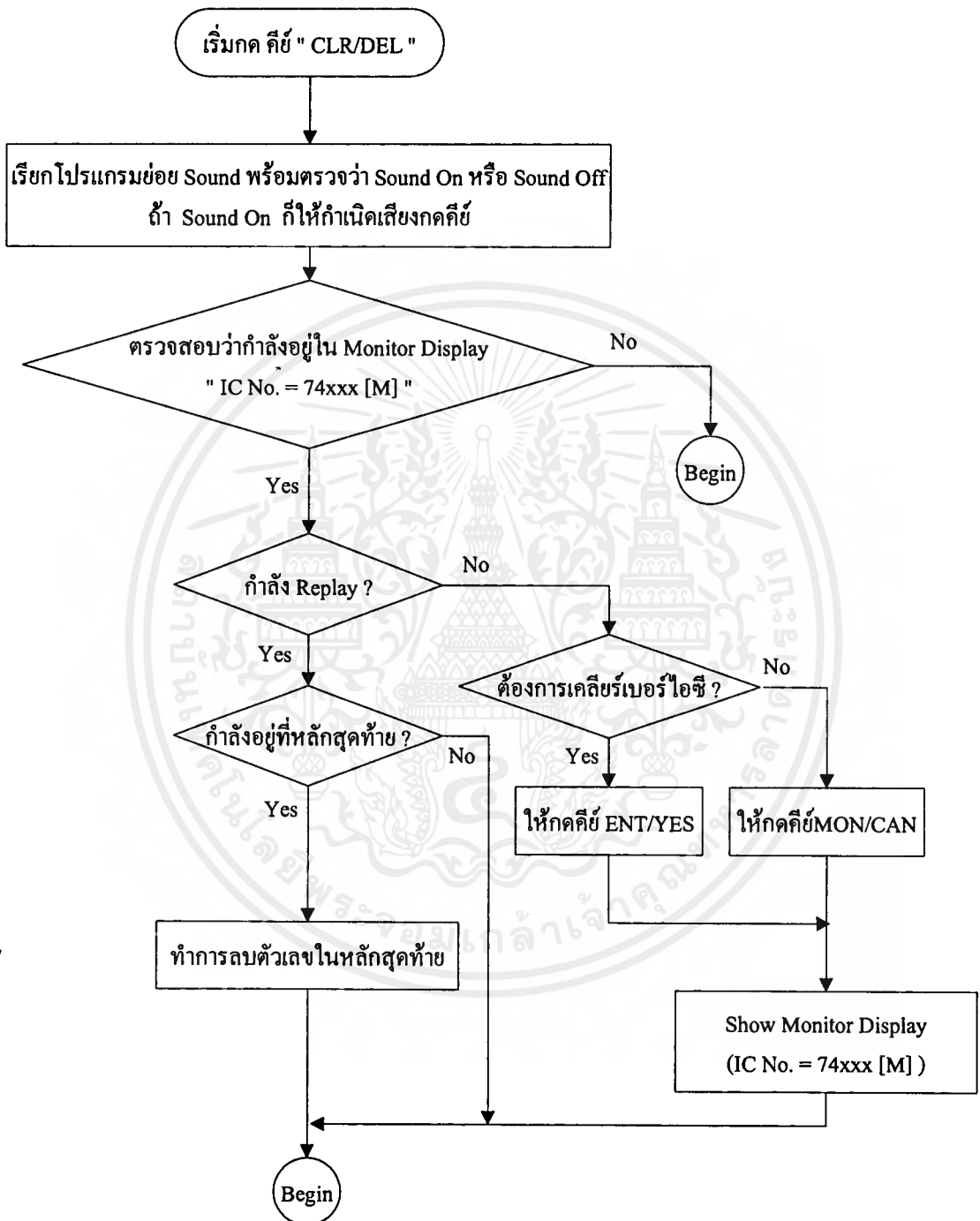
Flowchart ที่ 4.7 แสดงการทำงานของ โปรแกรมเมื่อกดคีย์ “5-9”

จาก Flowchart ที่ 4.7 จะเป็นการแสดงการทำงานของโปรแกรม เมื่อกดคีย์ 5 ถึง คีย์ 9 โดยใน Flowchart จะเห็นได้ว่ามีตัวแปร x อยู่ซึ่งจะเปลี่ยนไปตามคีย์ที่กด (คีย์ 5 ถึง คีย์ 9) ถ้ากดคีย์ 9 ก็ให้นำเลข 9 เข้าไปแทนในตัวแปร x

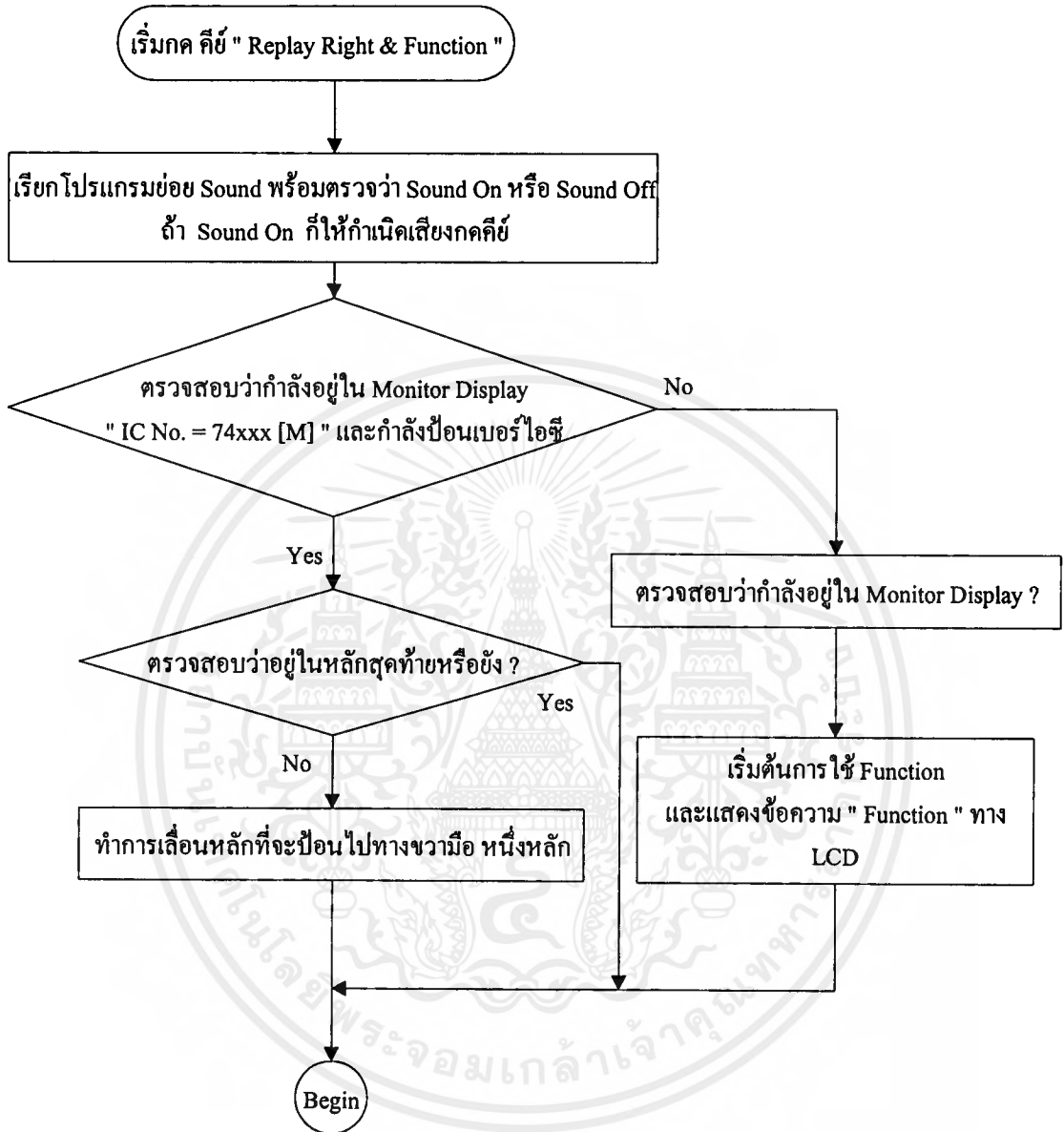
ส่วนคีย์อื่นๆ ก็สามารถอธิบายการทำงานของโปรแกรมเป็น Flowchart ที่ 4.8 ถึง Flowchart ที่ 4.13 ดังต่อไปนี้



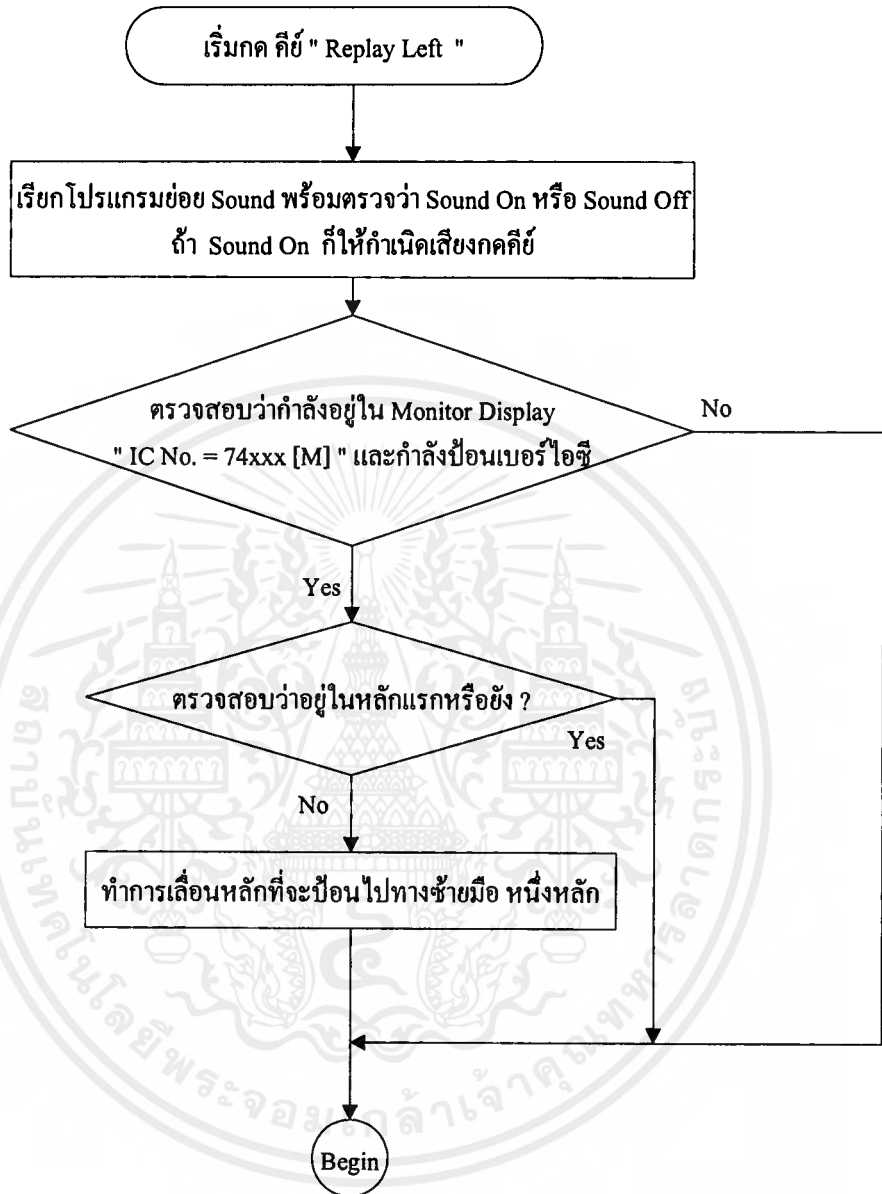
Flowchart ที่ 4.8 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “ENT/YES”



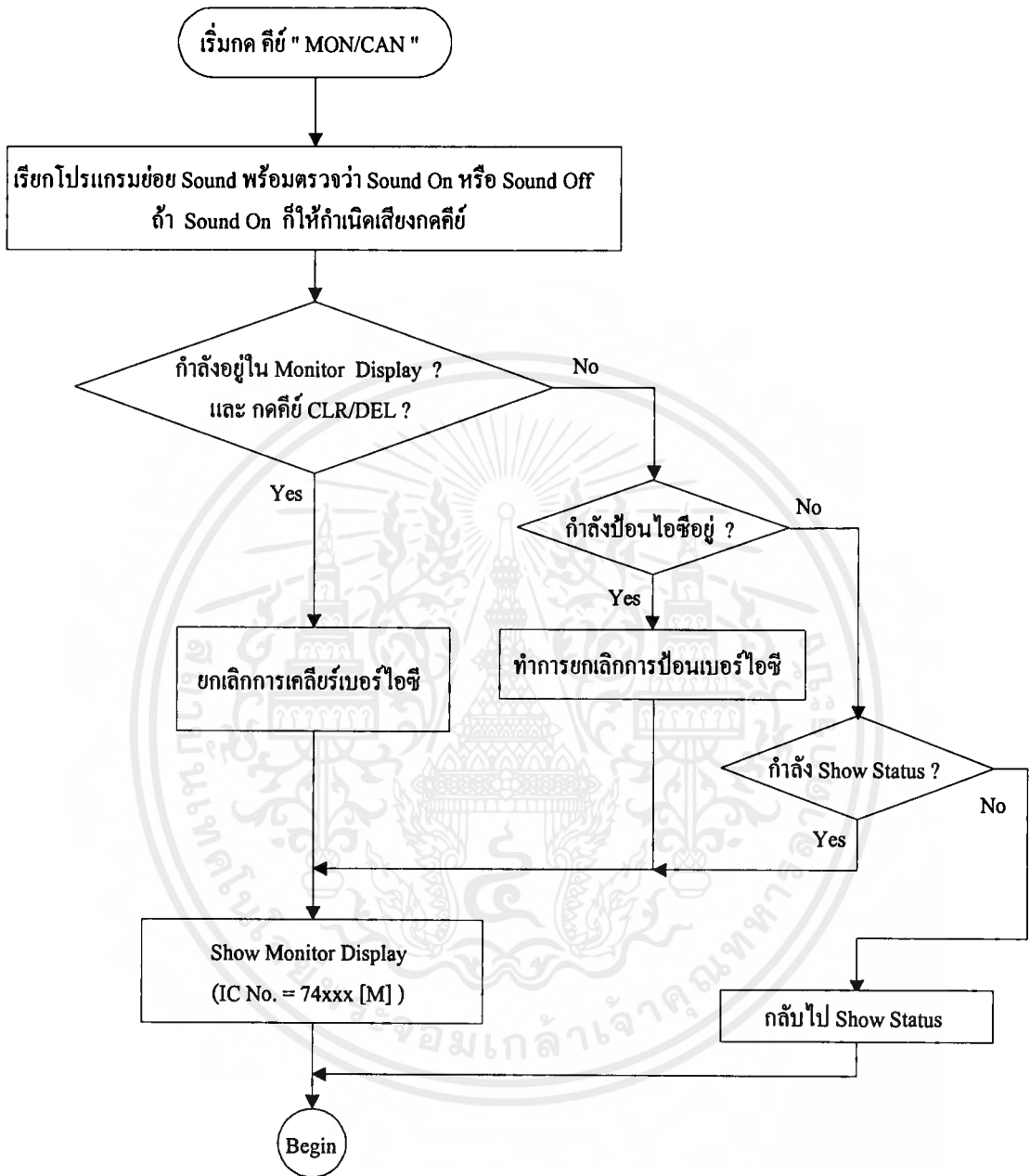
Flowchart ที่ 4.9 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "CLR/DEL"



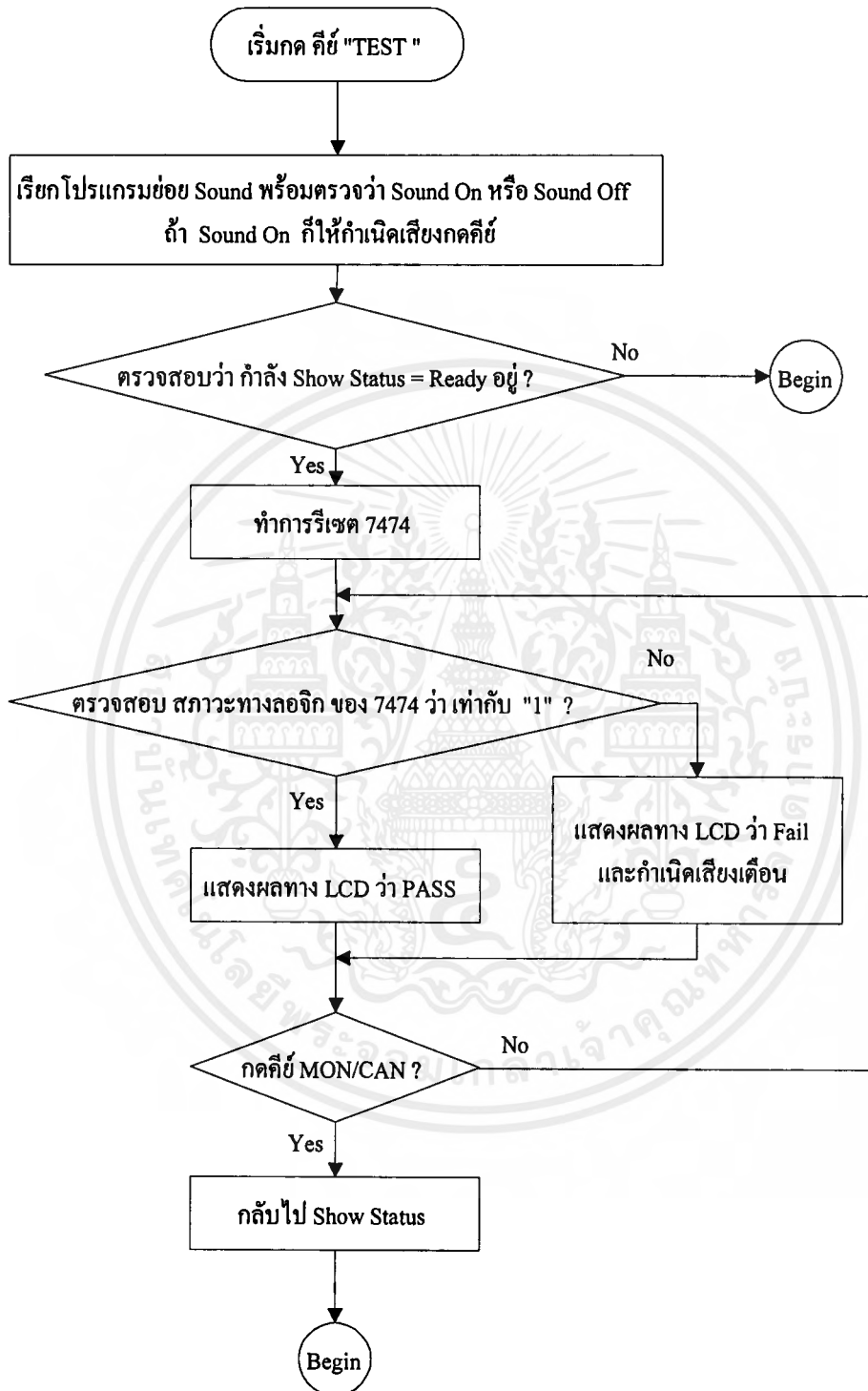
Flowchart ที่ 4.10 แสดงการทำงานของ โปรแกรมเมื่อกดคีย์ “Replay Right & Function ”



Flowchart ที่ 4.11 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ “Replay Left”



Flowchart ที่ 4.12 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "MON/CAN"



Flowchart ที่ 4.13 แสดงการทำงานของโปรแกรมเมื่อกดคีย์ "TEST"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อไปคือการอธิบายถึงวิธีเขียนโปรแกรมการเปิดตาราง เพื่อควบคุมการทำงานของ 74125 ที่ทำหน้าที่จัดขาว่าขาใดเป็นขาอินพุตหรือขาเอาต์พุตและการจัดการเกี่ยวกับการจ่ายไฟเลี้ยงให้ กับตัว IC อ้างอิง จากหัวข้อที่ 4.1 ไม่ว่าจะเป็กรณที่เป็น IC แบบ 14 ขา หรือ 16 ขา จะเห็นว่า Port ของ 8255 ที่ใช้งานจะประกอบไปด้วย 3 Port คือ Port A ใช้ควบคุมการจ่ายไฟเลี้ยงให้กับ IC อ้างอิง Port B และ Port C ใช้ควบคุมว่าขาใดเป็นขาอินพุตหรือขาเอาต์พุต จะเห็นใน IC ในแต่ละเบอร์นั้น จะต้องใช้ Data เป็นจำนวนทั้งหมด 3 Byte เพราะฉะนั้น โปรแกรมที่เขียนก็ต้องควบคุมการเปิด ตาราง จำนวน 3 Byte

ในการป้อนเบอร์ IC จะเป็นตัวเลขจำนวน 3 หลัก ซึ่งสามารถจัดเป็นกลุ่มใหญ่ ๆ ได้ 10 กลุ่ม คือ

กลุ่มใหญ่กลุ่มที่ 1	จะเป็น 7400x-7499x
กลุ่มใหญ่กลุ่มที่ 2	จะเป็น 74100-74199
กลุ่มใหญ่กลุ่มที่ 3	จะเป็น 74200-74299
กลุ่มใหญ่กลุ่มที่ 4	จะเป็น 74300-74399
กลุ่มใหญ่กลุ่มที่ 5	จะเป็น 74400-74499
กลุ่มใหญ่กลุ่มที่ 6	จะเป็น 74500-74599
กลุ่มใหญ่กลุ่มที่ 7	จะเป็น 74600-74699
กลุ่มใหญ่กลุ่มที่ 8	จะเป็น 74700-74799
กลุ่มใหญ่กลุ่มที่ 9	จะเป็น 74800-74899
กลุ่มใหญ่กลุ่มที่ 10	จะเป็น 74900-74999

ซึ่งในการแบ่งกลุ่มของเบอร์ IC เป็น 10 กลุ่มใหญ่ ก็เพื่อประโยชน์ในการเปิดตาราง เนื่องจากเบอร์ IC ที่ใช้งานกันมีอยู่เป็นจำนวนมาก การเขียนโปรแกรมควบคุมก็จะเขียนเพื่อ วิเคราะห์ ว่าตัวเลขที่เราป้อนเข้าไปนั้นในตัวเลขหลักร้อยหรือหลักแรกเป็นตัวเลขอะไร จากนั้นก็ ให้กระโดด ไปทำงานต่อไปตามกลุ่มใหญ่ที่ได้จัดไว้ ซึ่งในแต่ละกลุ่มใหญ่ ๆ ก็จะจัดแบ่งออกเป็นกลุ่มย่อยคือ

#### กลุ่ม 00-99

การที่แบ่งเป็นกลุ่มย่อยคือ 00-99 ของแต่ละกลุ่มใหญ่ก็เพราะว่าหลังจากวิเคราะห์ตัวเลขใน หลักแรกหรือหลักร้อยแล้ว ก็จะตัดหลักนี้ไป ก็จะเหลืออยู่สองหลักหลักคือ หลักสิบกับหลักหน่วย จากนั้นก็นำตัวเลขสองหลักดังกล่าวมาเป็นตัวเลขที่ใช้ในการตารางได้ แต่ในกลุ่มใหญ่กลุ่มที่ 1 จะเห็นว่าไม่มีตัวเลขแค่ สองหลักเอง จึงไม่ต้องวิเคราะห์ สามารถนำตัวเลขดังกล่าวมาใช้เปิดตารางได้ เลย คือหลักสิบกับหลักหน่วย

ในการเปิดตาราง ก็จะใช้คำสั่ง `MOVC A,@A+DPTR` คำสั่งนี้มีจุดประสงค์เพื่อใช้ในการเปิดค่าข้อมูลที่เก็บไว้ในหน่วยความจำสำหรับเก็บโปรแกรม ซึ่งข้อมูลไม่สูญหายแม้ว่าจะไม่มีพลังงาน

การทำงานของคำสั่งนี้จะใช้ค่าของรีจิสเตอร์ใช้เฉพาะ DPTR มารวมกับค่าในรีจิสเตอร์ A เพื่อไปชี้ตำแหน่งของหน่วยความจำสำหรับเก็บโปรแกรมซึ่งเก็บข้อมูลไว้ ดังนั้นค่าในรีจิสเตอร์ใช้งานเฉพาะ DPTR จะต้องมีค่าเท่ากับตำแหน่งต้นของหน่วยความจำส่วนที่เก็บข้อมูลที่ต้องการ ส่วนค่าของรีจิสเตอร์ A จะเป็นตัวเลือกข้อมูลที่อยู่ในหน่วยความจำ

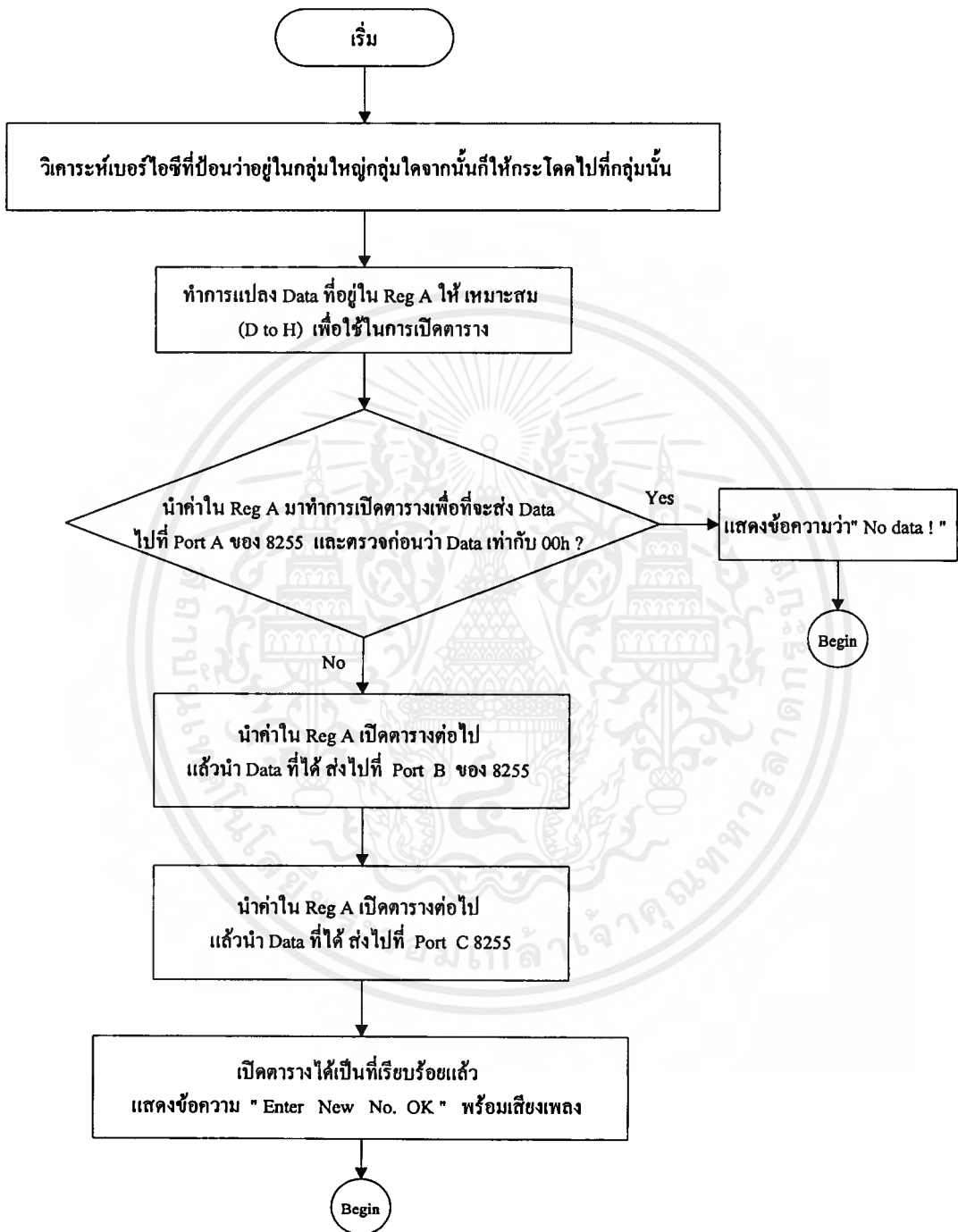
จากการทำงานของคำสั่ง `MOVC A,@A+DPTR` ที่ได้อธิบายไปแล้วนั้น ก็จะได้ว่า ค่าที่จะอยู่ในรีจิสเตอร์ A คือ ตัวเลขในกลุ่มย่อย ของแต่ละกลุ่มใหญ่ คือ 00-99 นั้นเอง ส่วนค่าที่จะอยู่ในรีจิสเตอร์ใช้งานเฉพาะ DPTR ก็คือค่าตำแหน่งต้นของหน่วยความจำที่เก็บข้อมูลของกลุ่มใหญ่ จากนั้นก็ทำการเปิดตาราง ได้อย่างถูกต้องถึงแม้ว่าเบอร์ IC จะมีเป็นจำนวนมากก็ตาม

การนำ Data มาจัดเรียงเป็นตารางและใช้ในการเปิดตารางนั้น จะมีทั้งส่วนที่เป็นไอซีที่มีจำนวนขา 14 ขา, 16 ขา และที่มากกว่า 16 ขา สำหรับไอซีที่มีจำนวนขามากกว่า 16 ขา นั้น จะถือว่าเป็นส่วนที่ไม่มีข้อมูล กล่าวคือไม่สามารถตรวจสอบหรือใช้งานได้ก็จะให้ Data ในส่วนนี้มีข้อมูลเป็น 00h เพื่อให้รู้ว่าเป็นส่วนของไอซีที่มีจำนวนขาที่มากกว่า 16 ขา และไม่สามารถเปิดตารางได้

สิ่งสุดท้ายและเป็นสิ่งที่สำคัญที่จะกล่าวสำหรับวิธีการเปิดตารางก็คือ ค่าของรีจิสเตอร์ A ที่ได้มาจากกลุ่มย่อย ๆ ของกลุ่มใหญ่ 10 กลุ่ม ในความคิดของเราเองจะเห็นว่าเป็นเลขฐาน 10 ในตอนที่เราได้ทำการป้อนเบอร์ IC นั้น แต่ในความเป็นจริงแล้ว MCS-51 จะมองเห็นว่าตัวเลขที่ป้อนเป็นเลขฐาน 16 ตัวอย่างเช่น IC TTL เบอร์ 74LS145 ก็จะอยู่ในกลุ่มใหญ่กลุ่มที่ 2 และมีค่าในรีจิสเตอร์ A เท่ากับ 45h ซึ่งเท่ากับ 69d การที่จะนำค่าที่อยู่ในคือ 45h ในรีจิสเตอร์ A ไปใช้ในการเปิดตารางนั้นก็จะทำให้การเปิดตารางได้ข้อมูลที่ผิด เนื่องจากในการจัดเรียงตารางจะจัดเรียงกันไปเรื่อย ๆ เป็น Address ซึ่งใน 1 Address ก็จะเก็บ Data ได้ 1 Byte และในแต่ละ Address ก็จะจัดเรียงกันไปตามเบอร์ไอซี นั่นคือ 1 Address ก็เท่ากับไอซี 1 เบอร์ การนับ Address ที่จัดเรียงกันไปเราก็จะนับเป็นฐาน 10 เพราะฉะนั้นจึงต้องทำการแปลงค่าในรีจิสเตอร์ A ก่อน เมื่อทำการแปลงแล้วจะได้ว่า 45h เท่ากับ 2Dh จึงค่อยนำ Data ในรีจิสเตอร์ A ซึ่งเท่ากับ 2Dh ไปใช้ในการเปิดตารางได้ด้วยเหตุผลนี้จึงต้องทำการเขียนโปรแกรมเพื่อที่จะแปลงจาก 45h ให้เป็น 2Dh ก่อน จึงสามารถจะนำค่าในรีจิสเตอร์ A ไปใช้เปิดตารางได้อย่างถูกต้องตรงตามเบอร์ IC ที่ได้จัดเรียงตารางเอาไว้ในแต่ละกลุ่ม วิธีการเปิดตารางจำนวน 1 Byte ดังที่กล่าวไปแล้วนั้น ก็จะต้องทำการเปิดตารางในแต่ละ Byte ทีละ Byte แล้วนำค่าที่ได้ ส่งออกไปยัง 8255 Port ทีละ Byte จนครบทั้ง 3 Byte ซึ่งในการเปิดตารางในแต่ละ Byte ก็จะใช้หลักการเดียวกันทั้ง 3 Byte

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวิธีการเปิดตารางที่ได้อธิบายไปแล้วนั้นสามารถเขียนเป็น Flowchart ได้ดังนี้



Flowchart ที่ 4.14 แสดงวิธีการเปิดตารางเพื่อ ส่ง Data ไปที่ 8255 Port

สำหรับ Source Code ที่สมบูรณ์ของเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง Version 1.0 นี้จะถูกรรจไว้ในแผ่น Disk ที่อยู่ท้ายเล่มของปฏิญญาฉบับนี้ ซึ่งสามารถนำมาศึกษาถึงการทำงานของโปรแกรมโดยละเอียดได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### การทดลองใช้งานเครื่องตรวจ IC TTL ในวงจร

#### 5.1 การทดลองใช้งาน

ในบทนี้จะกล่าวถึงการทดลองนำเครื่องตรวจ IC TTL ในวงจรไปใช้งาน โดยนำไปทดลองกับวงจรที่มี IC ตระกูล TTL ใช้งานอยู่ในวงจร ทั้งที่มีลักษณะเอาต์พุตแบบ Totem - Pole และแบบ Open Collector ซึ่งในการนำไปใช้ต้องปฏิบัติตามขั้นตอนต่อไปนี้

1. เชียบ IC เบอร์เดียวกันกับ IC ที่ต้องการตรวจในวงจรเป้าหมาย ลงใน Socket IC อ้างอิง
2. ป้อนเบอร์ IC ที่ต้องการตรวจให้ถูกต้อง
3. นำคัลิป Test ไปหนีบคร่อมบนตัว IC ที่ต้องการตรวจในบอร์ดเป้าหมาย
4. Power On ให้กับบอร์ดเป้าหมายและให้มีการทำงานเกิดขึ้น
5. สังเกตผลการตรวจสอบได้จาก LCD และ ลำโพง

จากขั้นตอนที่กล่าวมาแล้วนั้น ในขั้นตอนข้อที่ 2 และขั้นตอนในข้อที่ 6 สามารถศึกษาเพิ่มเติมได้จากภาคผนวก

#### 5.1.1 การทดลองกับ IC ที่มีเอาต์พุตเป็นแบบ Totem - Pole

ในการทดลองสำหรับ IC ที่มีเอาต์พุตเป็นแบบ Totem-Pole นั้น ในที่นี้จะทดลองใช้กับวงจรประเภทที่เป็น Flip-Flop คือ IC เบอร์ 74HCT74 ซึ่งเป็น “Dual D-Type Flip-Flop” โดยต่อกันเป็นวงจร Shift Register ขนาด 4 Bit ดังรูปที่ 5.1 จากรูปมีการทำงานของวงจรคือ เมื่อป้อน Data ไป Data ที่ป้อนเข้าไปจะมีการเลื่อน Data ตามจังหวะของสัญญาณ Clock ที่ป้อนเข้าไปด้วย หรือกล่าวได้ว่าความเร็วในการเลื่อน Data ขึ้นอยู่กับความถี่ที่ป้อนให้กับวงจร จะเห็นได้ว่าวงจรนี้จะมีจังหวะของการทำงานตามสัญญาณ Clock ด้วยว่าเป็นขอบขาขึ้นหรือขอบขาลง ในการตรวจสอบนั้นดังที่ได้เคยกล่าวไปแล้วว่าต้องให้ทั้ง IC อ้างอิงและ IC ที่ต้องการทดสอบมีจุดเริ่มต้นการทำงานที่พร้อมกันด้วย มิฉะนั้นผลการตรวจสอบก็ผิดพลาดได้

ทดลองป้อนสัญญาณ Clock ที่ความถี่ต่ำ ๆ ก่อน คือ 1 Hz และเพิ่มเป็น 2 Hz, 20Hz, 200Hz 2KHz, 20KHz, 200KHz, 2MHz พร้อมกับป้อน Data ทั้งที่เป็น ลอจิก “1” และ ลอจิก “0”

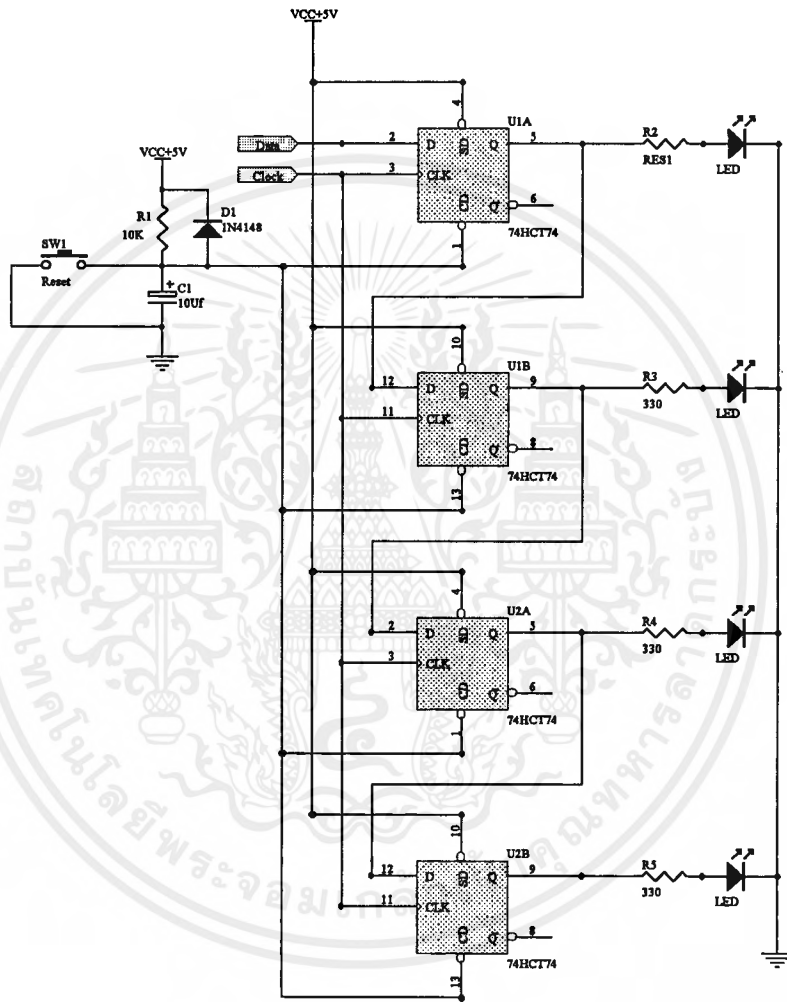
เครื่องตรวจ IC TTL ในวงจร ก็สามารถตรวจสอบการทำงานได้ที่ ณ ระดับความถี่ต่าง ๆ ดังที่ได้กล่าว จะเห็นได้ว่าเครื่องตรวจ IC TTL ในวงจร สามารถตรวจ IC ที่เป็น Flip-Flop ได้ และ

ที่เป็น CMOS คือพวก 74HCTxxx หรือ 74ACTxxx ได้ เนื่องจาก IC ทั้ง สองแบบ มีการรับอินพุตที่มีระดับลอจิกเป็นแบบ TTL และเอาต์พุตสามารถขับได้ทั้ง TTL และ CMOS

### 5.1.2 การทดลองกับ IC ที่มีเอาต์พุตเป็นแบบ Open Collector

ในการทดลองกับ IC ที่มีเอาต์พุตเป็นแบบ Open Collector นั้นจะใช้วงจรดังรูปที่ 5.2 จากรูป เป็นวงจร Scan Display โดยมี IC เบอร์ 74LS145 ซึ่งเป็น IC ในตระกูล TTL ที่มีเอาต์พุตเป็นแบบ Open Collector ทำหน้าที่เป็นตัว Scan Display และ Display ที่ใช้จะเป็น 7 Segment จำนวน 4 หลัก วงจรนี้จะควบคุมการทำงานด้วย MCS-51 ในที่นี้จะใช้บอร์ด CP-SB31 แล้วเขียนโปรแกรมให้ Scan Display ที่ละหลักด้วยความเร็วที่สามารถมองเห็นการ Scan Display ได้จากนั้นก็ให้เพิ่มความเร็วในการ Scan Display เองและเพิ่มมากขึ้นเรื่อย ๆ จนกระทั่งมองเห็นว่าการ Scan Display ดิจพร้อมกันทั้ง 4 หลัก ในขณะที่ดิจพร้อมกันทั้ง 4 หลักนี้จะเห็นเป็นตัวเลขว่า “0123”

จากการทดลองใช้งานกับวงจรถัดกล่าว ณ ความเร็วในการ Scan Display ที่ความเร็วระดับต่าง ๆ เครื่องตรวจ IC TTL ในวงจรก็สามารถตรวจสอบการทำงานได้



รูปที่ 5.1 วงจร Shift Register ขนาด 4 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## บทที่ 6

### สรุปผลโครงการและแนวทางการพัฒนาต่อ

#### 6.1 สรุปผลโครงการ

จากการทำโครงการเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง ซึ่งมีขอบเขตการใช้งาน คือ สามารถใช้ตรวจ IC TTL ที่มีจำนวนขาได้สูงสุดจำนวน 16 ขา หลังจากได้ทดลองนำเอาเครื่องไปทดลองใช้กับวงจรที่มี IC TTL อยู่ในวงจรแล้ว จากผลการทดลองคือ สามารถตรวจสอบได้ทั้ง IC ที่มีเอาต์พุตเป็นแบบ Totem - Pole หรือที่มีเอาต์พุตเป็นแบบ Open Collector และ IC จำพวก Flip-Flop, Counter ได้ โดยตรวจได้ทั้งแบบ 14 ขา และ แบบ 16 ขา และตรวจ IC CMOS ที่มี Function การทำงานแบบ IC TTL ได้ เช่น 74HCTxxx ได้ เนื่องจาก IC CMOS สามารถใช้แทน IC TTL ได้ จึงไม่มีปัญหาในเรื่องของระดับลอจิก

#### 6.2 ปัญหาที่พบและข้อมูลเสนอแนะ

ในการสร้างตัวเครื่องตรวจ IC TTL ในวงจรมันั้น ปัญหาที่พบก็คือเรื่องของความเร็วในการทำงานของเครื่องตรวจ IC TTL ในวงจร ซึ่งบางครั้งในการใช้งานอาจผิดพลาดได้จึงควรพยายามลด Propagation Delay ให้ได้มากที่สุด โดยใช้ IC ที่มีการทำงานด้วยความสูงเข้ามาช่วย เช่น เบอร์ 74HCTxxx , 74ACTxxx , 74Fxxx ซึ่งก็จะช่วยให้การทำงานของวงจรเครื่องตรวจ IC ในวงจร มีการทำงานที่เร็วขึ้นได้ อีกทั้งเทคนิคการออกแบบวงจรด้วย

#### 6.3 แนวทางในการพัฒนาต่อ

เนื่องจากขอบเขตของโครงการนี้ได้กำหนดไว้ให้ใช้งานกับ IC ในตระกูล TTL ได้สูงสุดจำนวน 16 ขา แต่ IC ในตระกูล TTL มีจำนวนขาออกจาก 14 ขา , 16 ขา แล้ว ยังมีจำนวนขาแบบที่มากกว่า 16 ขา เช่น 20 ขา , 24 ขา อีกด้วย ดังนั้นเราจึงสามารถนำหลักการ DRC มาใช้ เพื่อให้สามารถตรวจสอบได้กับไอซีที่มีจำนวนขา มากกว่า 16 ขาได้ และในการใช้งานนั้นควรที่จะให้เครื่องตรวจ IC TTL ในวงจร สามารถที่จะตรวจ ตัว IC ที่นำเป็น IC อ้างอิงได้ ก่อนที่จะนำไปใช้ตรวจกับ IC ที่อยู่ในวงจรเพราะว่า ถ้า IC ที่นำมาเป็น IC อ้างอิงเสียอยู่ก็จะทำให้ ผลการตรวจผิดพลาดได้ ซึ่งการทำให้เครื่องตรวจ IC TTL ในวงจร สามารถตรวจตัว IC TTL ที่นำเป็น IC อ้างอิงได้ก็คือ การศึกษาถึงเงื่อนไขการทำงานทางลอจิกของ IC TTL ในแต่ละเบอร์ว่า มีการทำงานอย่างไร จากนั้นก็ทำการเขียนโปรแกรมควบคุมการป้อนอินพุตตามเงื่อนไขการทำงาน แล้วก็ตรวจสอบ

ว่าเอาต์พุตที่ได้หลังจากป้อนอินพุตแล้วว่าถูกต้องเป็นไปตามเงื่อนไขหรือไม่ก็จะบอกได้ว่า IC TTL ที่นำมาเป็น IC อ้างอิงนั้นดีหรือเสีย และสิ่งที่ควรแก้ไขและพัฒนาต่อก็คือการทำให้สามารถตรวจ IC TTL ประเภทที่มีการทำงานที่สามารถสลับให้เป็นได้ทั้งขาอินพุตและขาเอาต์พุตในขาเดียวกันได้หรือระบบบัสที่มีการรับส่งข้อมูลแบบสองทิศทาง (Bidirectional Bus) เช่น 74242 , 74243



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

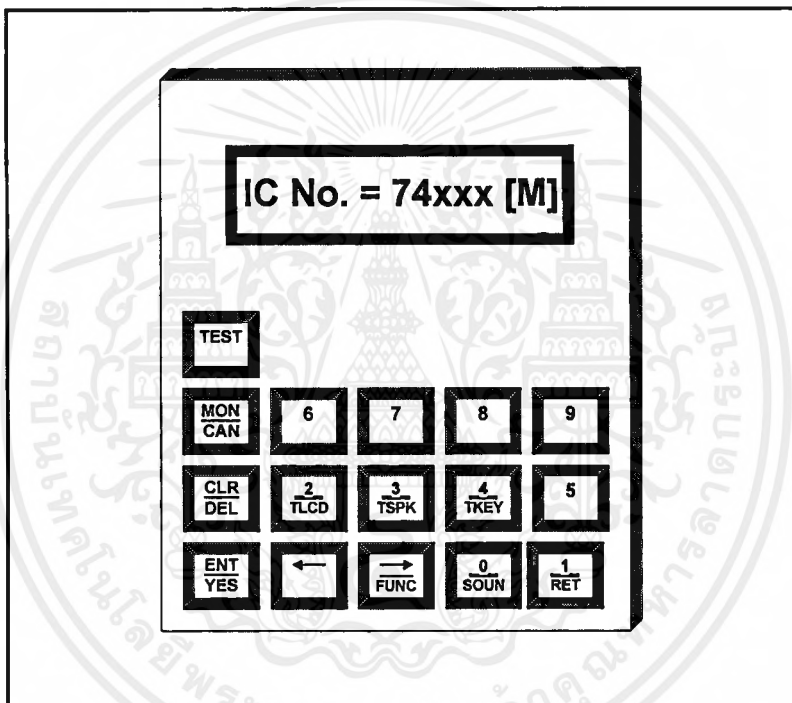
1. ทรงชัย วีระทวีมาศ , “บทความเรื่อง เทคนิคการออกแบบวงจรดิจิทัล ตอน: วงจรตรวจสอบไอซีในวงจรด้วยความเร็วสูง” , วารสาร เซมิคอนดักเตอร์ ฉบับที่ 110 ก.ย.-ต.ค. 34 , บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน)
2. รศ.ดร. ถวัลย์วงศ์ ไกรโรจนานันท์ , “อิเล็กทรอนิกส์ระบบดิจิทัล” , กองบริการสื่อสารสนเทศ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ
3. ชัยวัฒน์ ลิ้มพรจิตรวิไล , “คู่มืออิเล็กทรอนิกส์” , บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน)
4. ประเมษฐ์ ประณยานันท์ และ ปิยพงศ์ เผ่าวิช , “คู่มือและการประยุกต์ใช้งาน ไมโครคอนโทรลเลอร์ MCS-51 ” , บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน)
5. คู่มือการใช้งานบอร์ด ANT-32 Version 3.0 EMBEDDED CONTROL BOARD , บริษัท ศิลา รีเสิร์ช จำกัด
6. คู่มือการใช้งาน LCD , บริษัท ศิลา รีเสิร์ช จำกัด
7. JOHN F. WAKERLY , “DIGITAL DESIGN PRINCIPLES AND PRACTICES” , Prentice-Hall International Editions
8. James E. Buchanan and Bert D. Buchanan , “Signal and Power Integrity in Digital System : TTL , CMOS , and BiCMOS” , McGraw-Hill
9. PARAG K. LALA , Digital Circuit Testing and Testability , Academic Press

## ภาคผนวก ก.

## การใช้งานของเครื่องตรวจ IC TTL ในวงจรด้วยความเร็วใช้งานจริง

## 1. ลักษณะของ Key Panel

ลักษณะของเครื่องโดยทั่วไปสามารถแสดงได้ดังรูปที่ 1



รูปที่1 แสดงคีย์ 16 คีย์ใช้งาน และ LCD

เมื่อ Power On เครื่องตรวจ IC TTL ในวงจร ครั้งแรกจะมีเสียงเพลง Title ที่ลำโพงเล็ก และมีข้อความว่า “KMITL,Eng,Ins#17” และ “Real Time TTL In-Circuit Testing” วิ่งออกมาจากทางขวามือไปทางซ้ายมือ และแสดงข้อความว่า “Version 1.0”

จากนั้นก็เข้าสู่การแสดงผลสถานะของเครื่อง คือ เมื่อไม่มีการป้อนเบอร์ไอซีใด ๆ เลข จะมีข้อความที่ LCD ว่า

**Status: Wait**

ในทางตรงกันข้ามถ้ามีการป้อนเบอร์ไอซีแล้ว ก็จะแสดงข้อความ จอ LCD ว่า

**Status: Ready**

## 2. การป้อนเบอร์ไอซี

เมื่อจอ LCD อยู่ในการแสดงสถานะ ดังที่กล่าวมาคือแสดง

**Status: Wait**

หรือ

**Status: Ready**

อยู่ให้

1. กด



จอ LCD จะแสดง

**IC No. = 74xxx [M]**

ข้อความนี้แสดงว่าขณะนี้ไม่มีการป้อนเบอร์ไอซีเบอร์ใดๆอยู่ สังเกตจาก 74 แล้ว ตามด้วย xxx ส่วนตัว M ที่อยู่ใน [ ] หมายถึงกำลังอยู่ใน Monitor Display ซึ่งเป็นการแสดงข้อความว่า ณ ขณะนี้ได้ป้อนไอซีเบอร์อะไร ไปบ้างแล้วหรือยังไม่ได้มีการป้อนอยู่

2. กดตัวเลขตามเบอร์ไอซีที่ต้องการ เช่น 74125 ก็ กด



แล้วกด  จอจะแสดง

**Enter New No. OK**

ถ้ามี Data ของไอซีเบอร์ที่ป้อนไป และจอจะแสดง

**IC No. = 74125 [M]**

ส่วนกรณีที่ไม่มี Data จอ LCD จะแสดง


**No Data !**

หลังจากกด 

สังเกตว่าขณะที่ป้อนเบอร์ตัวอักษรในกรอบ [ ] โดยปกติจะเป็น M (Monitor) ซึ่งแสดงให้รู้ว่ากำลังอยู่ใน Monitor อยู่ และจะเปลี่ยนไปเป็น E (Edit) ซึ่งแสดงให้รู้ว่ากำลังป้อนเบอร์ไอซีอยู่และ

จะกลับมาเป็น ตัว M เมื่อกดคีย์  หรือ 

**IC No. = 741xx [E]**




3. ขณะป้อนเบอร์หากต้องการยกเลิกการป้อนเบอร์ ให้กด  จอ LCD จะแสดงข้อความ

## Cancel Enter No.

แล้วจอแสดงข้อความก็จะกลับไปแสดงเบอร์เดิมก่อนหน้าที่ยกป้อนไว้หรือถ้าไม่ได้ป้อนไว้ก่อนหน้าก็จะแสดงข้อความ

## IC No. = 74xxx [M]

### 3. การ Replay Right & Replay Left

ในขณะที่ป้อนเบอร์ไอซีเราสามารถย้อนไปแก้ไขในหลักที่กดป้อนตัวเลขได้ เช่น เราต้องการป้อนเบอร์ 74126 แต่ไปกด 74135 และยังไม่กด  ให้กด  หรือ  เพื่อย้อนกลับไปแก้ไขตัวเลขในหลักที่เราคิดได้ ทั้งทางมือและทางขวามือ ในตัวเลขทั้ง 3 หลัก สังเกต Cursor ที่กระพริบย้อนไปมาในหลักตัวเลขตามที่เรากดย้อนไป

เมื่อ Cursor กำลังกระพริบที่หลักใด และเมื่อเรากดตัวเลขตัวใหม่ที่ต้องการแก้ไข ก็จะเกิดการเปลี่ยนตัวเลขในหลักนั้นๆ เป็นตัวเลขตัวใหม่ที่เรากดทันที เมื่อแก้ไขเป็นที่เรียบร้อยและถูกต้องแล้ว

ก็กด 

### 4. การ Clear

1. เมื่อเราต้องการ Clear เบอร์ไอซีที่เราเคยป้อนไปแล้วก่อนหน้านี้ เช่น 74125

ก็สามารถทำได้โดย กด  ในขณะที่กำลังอยู่ที่ Monitor [M]

จอ LCD แสดงข้อความ

## Are You Sure ?

ถ้าแน่ใจว่าต้องการ Clear ก็ให้กด  จอ LCD จะแสดงข้อความ

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับบริการเท่านั้น เมื่อผู้เช่าได้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Clear! 74125 OK**

แล้วกลับไปแสดงข้อความ

**IC No. = 74xxx [M]**

ซึ่งหมายถึงว่า ณ ขณะนี้ไม่มีการป้อนเบอร์ไอซีเบอร์ใด ๆ อยู่

แต่ถ้าต้องการยกเลิกการ Clear ก็ให้กด  
จอ LCD จะแสดงข้อความ

**MON  
CAN**


**Cancel Clear**

## 5. การ Delete


1. ถ้าเราต้องการป้อนเบอร์ 7420 แต่เราคิดผิดไปเป็น 74200 เราสามารถแก้ไขในกรณีนี้ได้ โดยจะเห็นได้ว่าสองหลักแรกที่ได้ป้อนไปจะถูกตัดออกอยู่แล้ว มีเพียงแต่เลขศูนย์ที่เกินมา 1 ตัวในหลักสุดท้ายเท่านั้นที่ต้องแก้ไข

เราสามารถทำการ Delete เลขศูนย์ที่กดเกินออกได้ ถ้ายังไม่ได้กด

**ENT  
YES**

วิธี Delete คือ กด  ย้อนกลับไปหลักสุดท้าย คือเลขศูนย์ที่กดเกินมาหรือจะเป็นตัวเลขอื่นก็ตาม

สังเกต Cursor ที่กระพริบไปในหลักดังกล่าว แล้วกด  เลขศูนย์ในหลักสุดท้ายที่ได้กดเกินออกไปก็จะหายไป จะเห็นว่าเป็น 7420 ตามที่ต้องการแล้ว

จากนั้นก็ให้กด  การ Delete นี้ จะสามารถทำการ Delete ได้เฉพาะในหลักสุดท้ายเพียงหลักเดียวเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6. การใช้ Function

นอกจากการใช้งานปกติแล้วยังมี Function ใช้อื่นๆ อีกดังนี้

1. Sound On/Off
2. Keyboards Test
3. Speaker Test
4. Restart System
5. LCD Test

ทุก ๆ Function ใช้งานต้อง อยู่ใน Monitor Display [M] จึงจะใช้งานได้

### 6.1 Sound On/Off


กดคีย์ FUNC จอ LCD จะแสดงข้อความ

**Function**

กด  เพื่อออกจาก Function และจะกลับไป Monitor Display อีกครั้ง

หรือ กด  ก็จะ Off Sound และ  
จอ LCD จะแสดงข้อความ

**Sound: Off**

เมื่อต้องการให้ On Sound ก็ให้กดคีย์ FUNC แล้วกด  อีกครั้งก็จะ On Sound  
จอ LCD จะแสดงข้อความ

**Sound: On**

## 6.2 Keyboards Test


กดคีย์ FUNC จอ LCD จะแสดงข้อความ

**Function**

กด  เพื่อออกจาก Function และจะกลับไป Monitor Display อีกครั้ง

หรือ กด  จอ LCD จะแสดงข้อความ

**Keyboards Test**

ถ้าต้องการ Test ตัว Keyboards ก็ให้กด   
จอจะแสดงคีย์ที่ต้องการให้กด เช่น จอ LCD แสดงข้อความว่า

**Key : Test**

ก็ให้กด  จอ LCD แสดงข้อความต่อว่า

**Key : MON/CAN**

ซึ่งเป็นคีย์ต่อไปที่ต้องการให้ Test ก็ให้กด  หลังจากกดคีย์แล้วถ้าคีย์ยังใช้งานได้อยู่ก็จะผ่านไปให้กดคีย์อื่น ๆ ต่อไป จนครบทั้ง 16 คีย์ เมื่อ Test ตัว Keyboard เสร็จแล้ว

จอ LCD จะแสดงข้อความ

**Finish**

แล้วกลับไป Monitor Display อีกครั้ง

ถ้าต้องการยกเลิก การ Test ตัว Keyboards ให้กด  หลังจอ LCD จะแสดงข้อความ

**Keyboards Test**

ก็จะเป็นการยกเลิกการ Test ตัว Keyboards แล้วจอ LCD ก็จะกลับไป Monitor Display อีกครั้ง

### 6.3 Speaker Test

กดคีย์ FUNC จอ LCD จะแสดงข้อความ

**Function**

กด  เพื่อออกจาก Function และจะกลับไป Monitor Display อีกครั้ง

หรือ กด  จอ LCD จะแสดงข้อความ

**Speaker Test**

กด **ENT YES** อีกครั้ง เครื่องจะทำการ Test ลำโพงโดยไม่ว่าจะ เป็นการ Sound Off อยู่หรือไม่ และจอ LCD จะแสดงข้อความ

## Speaker Testing

และขณะเดียวกันก็ทำการ Test ลำโพงโดยกำเนิดเสียงต่างๆ ออกมาทางลำโพง เมื่อ Test ลำโพง เสร็จแล้ว

จอ LCD จะแสดงข้อความ

**Finish**

แล้วกลับไป Monitor Display อีกครั้ง

### 6.4 Restart System

กดคีย์ FUNC จอ LCD จะแสดงข้อความ

**Function**

กด **MON CAN** เพื่อออกจาก Function และจะกลับไป Monitor Display อีกครั้ง

หรือ กด **1 RET** จอ LCD จะแสดงข้อความ

**Restart System**

กด  เครื่องจะทำการ Restart

ถ้าขกเลิกRestart ให้กด  ขณะที่ จอ LCD จะแสดงข้อความ

## Restart System

ก็จะกลับไป Monitor Display อีกครั้ง

### 6.5 LCD Test

กดคีย์ FUNC จอ LCD จะแสดงข้อความ

## Function

กด  เพื่อออกจาก Function และจะกลับไป Monitor Display อีกครั้ง

หรือ กด  จอ LCD จะแสดงข้อความ

## LCD(16x1) Test

แล้วกด  จอ LCD จะแสดงข้อความ

## Begin LCD Test

จากนั้นเครื่องก็จะเริ่มทำการ Test จอ LCD โดยส่ง ASCII Code ไปให้กับ จอ LCD แสดงในทุกๆ Character ทั้ง 16 ตัวอักษรว่าในแต่ละหลักตัวอักษรสามารถแสดง Character ดังที่ได้ส่งข้อมูลไปได้หรือไม่และเมื่อ Test เสร็จ

จอ LCD จะแสดงข้อความ

**Finish**


แล้วกลับไป Monitor Display อีกครั้ง

ถ้าต้องการยกเลิก Test จอ LCD ให้กด  ขณะที่ จอ LCD จะแสดงข้อความ

**LCD(16x1) Test**

ก็จะกลับไป Monitor Display อีกครั้ง

7. การ Test เพื่อตรวจสอบว่า ไอซีทดสอบดีหรือเสีย

เมื่อทำการป้อนเบอร์ไอซีเป็นที่เรียบร้อยแล้ว ให้กด 

จอ LCD จะแสดงข้อความ

**Status: Ready**


ซึ่งหมายถึงว่าขณะนี้ตัวเครื่องพร้อมที่จะตรวจสอบไอซีตามเบอร์ที่ได้ป้อนไปแล้ว โดยต้อง

1. เสียบไอซีอ้างอิงเบอร์เดียวกันกับไอซีที่จะตรวจเป็นที่ถูกต้องและเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. ป้อนเบอร์ไอซีที่ต้องตรวจเป็นที่ถูกต้องและเรียบร้อยแล้ว

จากนั้นก็ให้นำเอาตัวคลิบ Test ไปคลิบบนตัวไอซีที่ต้องตรวจสอบ ในบอร์ดเป้าหมายแล้ว  
จึงเริ่ม Power On ให้กับบอร์ดเป้าหมายเพื่อให้บอร์ดทำงาน

แล้วก็กด  เครื่องก็จะแสดงผลการตรวจสอบว่าไอซีที่ต้องการตรวจสอบนั้นดีหรือ  
เสีย โดยถ้าไอซีที่ตรวจสอบนั้นดี จอ LCD จะแสดงข้อความ

**Result: Pass**

และถ้าหากว่าไอซีที่ตรวจสอบนั้นเสีย จอ LCD จะแสดงข้อความ

**Result: Fail**

และมีเสียงเตือนด้วย

เครื่องจะคอยตรวจสอบการทำงานของไอซีที่ต้องการตรวจสอบอยู่ตลอดเวลา เมื่อมีการทำ  
งานที่ผิดพลาดก็จะแสดงผลรายงานการตรวจสอบทันที

เมื่อต้องการออกจากการตรวจสอบนี้ให้ กด  จอ LCD จะแสดงข้อความ

**Status: Ready**

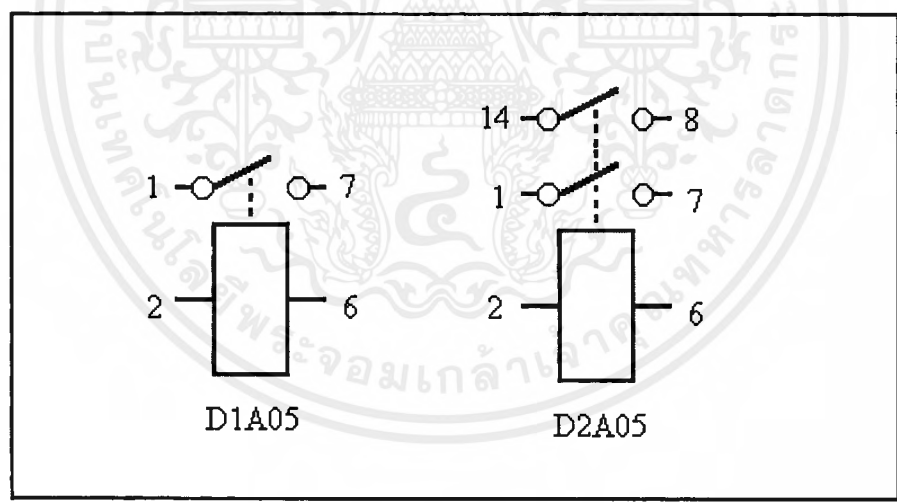
และถ้าหากจะกลับไป Monitor Display ก็ให้กด  อีกครั้ง



ภาคผนวก ข.

Data Sheet ของ Dip Reed Relay

Contact Form	Number	Nominal Voltage (VDC)	Coil Resistance $\pm 10\%$	Must Operate (VDC)	Must Release (VDC)	Rated Current (mA)	Continuous Voltage (ma)
1A SPST-NO	D1A05 (D)	5	500	3.75	1.0	10	10
2A DPST-NO	D2A05 (D)	5	140	3.75	1.0	35.7	10



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้