สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# ระบบป้ายจอดรถประจำทางเพื่อผู้พิการทางสายตา

# BUS STOP SYSTEM FOR VISUAL IMPAIRED

โดย

นายทศพล อาจหาญ

นายทัศนัย เชียงบาล

นายสมชาย ชัยสุขสันติ์

เลขหมู่.....
เลขทะเบียน 72127
วัน,เดือน,ปี 11 มิ.ย. 2550

.b.....
.i.....

**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต**

**สาขาวิชาอิเล็กทรอนิกส์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2549**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ระบบป้ายจอดรถประจำทางเพื่อผู้พิการทางสายตา

# BUS STOP SYSTEM FOR VISUAL IMPAIRED


**โดย**

นายทศพล อาจหาญ รหัส 46010259

นายทัศนัย เชียงบาล รหัส 46010262

นายสมชาย ชัยสุขสันติ์ รหัส 46010804


**อาจารย์ที่ปรึกษา**

ดร.กิตติพล ชิตสกุล

**ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต**

**สาขาวิชาอิเล็กทรอนิกส์**

**คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2549**


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบป้ายจอดรถประจำทางเพื่อผู้พิการทางสายตา

ผู้จัดทำ

1.นายทศพล อาจหาญ

2.นายทัศนัย เชียงบาล

3.นายสมชาย ชัยสุขสันติ์

อาจารย์ที่ปรึกษา ดร.กิติพล ชิตสกุล

ลงชื่อ.....

(ดร.กิติพล ชิตสกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ระบบป้ายจอดรถประจำทางเพื่อผู้พิการทางสายตา

นาย ทศพล อาจหาญ รหัส 46010259
นาย ทัศนัย เชียงบาล รหัส 46010262
นาย สมชาย ชัยสุขสันต์ รหัส 46010804
ดร. กิติพล ชิตสกุล อาจารย์ที่ปรึกษา
ปีการศึกษา 2549

**บทคัดย่อ**

โครงงานนี้เป็นการออกแบบระบบป้ายจอดรถประจำทางเพื่อผู้พิการทางสายตาและสร้างอุปกรณ์ขึ้นมารองรับกับระบบที่ออกแบบไว้ ประกอบไปด้วยสองส่วนคือส่วนป้ายเรียกรถประจำทางและส่วนนาฬิกาบอกเวลา โดยในส่วนของการเรียกรถประจำทางจะมีสวิตช์เลือกหมายเลขรถประจำทางที่ต้องการจะโดยสารได้ และนาฬิกาเพื่อใช้บอกเวลา ณ ขณะนั้น ซึ่งจะมีเสียงจากไอซีอัดเสียงและเล่นเสียงบอกให้ผู้พิการทางสายตาทราบถึงหมายเลขรถประจำทางและเวลา โดยใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมส่วนต่างๆของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# BUS STOP SYSTEM FOR VISUAL IMPAIRED


| | |
|---|---|
| Mr. Totsapon Athan | ID.46010259 |
| Mr. Tassanai Cheangban | ID.46010262 |
| Mr. Somchai Chaisuksant | ID.46010804 |
| Dr. Kitiphol Chitsakul | (Advisor) |

Education Year 2006



**Abstract**

This project is design and construction devices for support the bus stop system for visual impaired. There are two parts, bus calling and clock. Bus calling have many switches for choosing a bus. The clock is used to inform the moment time. This system generates sound from voice recorder and playback chip to inform the visual impaired about chosen bus's number and the moment time. Microcontrollers are used to control all devices in the system.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

**ขอขอบคุณ**

ขอขอบคุณท่านอาจารย์ดร.กิติพล ผู้เป็นที่ปรึกษาของโครงงานนี้ สำหรับคำแนะนำต่างๆทั้งแนวคิด การออกแบบและตลอดจนการแก้ไขปัญหาที่เกิดขึ้น

ขอขอบคุณท่านอาจารย์ภาควิชาอิเล็กทรอนิกส์ที่ให้ความรู้ต่างๆเพื่อมาทำโครงงานชิ้นนี้

ขอบคุณเพื่อนภาควิชาอิเล็กทรอนิกส์ห้อง 3 ที่ให้ความช่วยเหลือกันอย่างดี ขอบคุณเป็นพิเศษสำหรับ นาย ทัศไนย์ เทวีอำนวยทรัพย์ ที่ช่วยทั้งเรื่องวิชาการและกิจกรรม

สุดท้ายขอขอบคุณ คุณพ่อคุณแม่ ที่ให้ความไววางใจและเชื่อใจในการช่วยเหลือด้านการเงินตลอดจนความห่วงใยที่อบอุ่นเสมอมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


**บทคัดย่อ** I

**Abstract** II

**กิตติกรรมประกาศ** III

**สารบัญ** IV,V

**สารบัญรูปภาพ** VI

**สารบัญตาราง** VII

**บทที่ 1 บทนำ**

1.1 วัตถุประสงค์ 1

1.2 ขอบเขตของโครงงาน 1

1.3 ขอบเขตของรายงาน 1

**บทที่ 2 ทฤษฎี**

2.1 การเชื่อมโยงอุปกรณ์ด้วยระบบบัส $I^2C$ 2

2.1.1 ความรู้เบื้องต้นเกี่ยวกับ $I^2C$ 2

2.1.2 หลักการของบัส $I^2C$ 5

2.1.3 สภาวะที่เกิดขึ้นบนบัส $I^2C$ 6

2.1.4 การทำงานบนบัส $I^2C$ 7

2.1.5 การอ้างถึงแบบ 7 บิต (7-bit addressing) 8

2.1.6 การอ้างถึงแบบ 10 บิต 9

2.2 พื้นฐานการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI 9

2.2.1 การเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI 10

2.2.2 โครงสร้างการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI 10

2.3 ทฤษฎีไมโครคอนโทรลเลอร์ MCS-51 10

2.3.1 ลักษณะสำคัญของไมโครคอนโทรลเลอร์ตระกูลMCS-51 11

2.3.2 การจัดขาและโครงสร้างของไมโครคอนโทรลเลอร์ 11

2.3.3 พอร์ทอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 18



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**บทที่ 3 การออกแบบ**

3.1 ขอบเขตของการออกแบบ 21

3.2 การออกแบบระบบป้ายเลือกสายรถประจำทาง 21

3.3 การออกแบบระบบแสดงเวลา 24

3.4 การติดต่อกับไอซีบันทึกและเล่นเสียงเบอร์ ISD4003 24

3.5 โปรแกรมการทำงานของไมโครคอนโทรลเลอร์ 26

3.6 การแสดงผล 27

3.6.1 การแสดงผลของ 7-Segment 27

3.6.2 การขยายสัญญาณ (Amplifying) ของสัญญาณเสียง 30

**บทที่ 4 การทดลองและผลการทดลอง**

4.1 การทำงานของไมโครคอนโทรเลอร์ 31

4.2 การติดต่อกับไอซีสร้างฐานเวลาจริง DS1307 ผ่านระบบบัส $I^2C$ 31

4.3 การติดต่อกับไอซีบันทึกและเล่นเสียง ISD4003 โดยใช้การเชื่อมต่อแบบ SPI 31

4.4 การแสดงผลของตัวเลขเจ็ดส่วน (7-segment) และเสียง 33

**บทที่ 5 สรุป**

5.1สรุปผลการทดลอง 35

5.1วิเคราะห์ผลการทดลอง 35

5.2 สรุปผลของโครงงาน 36

**บรรณานุกรม**

**ภาคผนวก**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูปภาพ


รูปที่ 2.1 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่างๆบนรหัสบัส $I^2C$

รูปที่ 2.2 แสดงวงจรเอาต์พุตของอุปกรณ์ในระบบบัส $I^2C$

รูปที่ 2.3 การต่อตัวต้านทานพูลอัปบนสายสัญญาณในระบบบัส $I^2C$

รูปที่ 2.4 การต่อตัวต้านทาน $R_s$ เพื่อลดสัญญาณรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส $I^2C$

รูปที่ 2.5 ไดอะแกรมเวลาแสดงสถานะต่างๆในบัส $I^2C$

รูปที่ 2.6 รูปแบบของข้อมูลกำหนดแอดเดรสที่ใช้ในการอ้างถึงแบบ 7 บิต

รูปที่ 2.7 รูปแบบของข้อมูลอนุกรมที่ใช่ติดต่อกับอุปกรณ์บัส $I^2C$ เมื่อใช่การอ้างถึงแบบ 7 บิต

รูปที่ 2.8 รูปแบบของข้อมูลอนุกรมที่ใช้ติดต่อกับอุปกรณ์บัส $I^2C$ เมื่อใช้การอ้างถึงแบบ 10 บิต

รูปที่ 2.9 โครงสร้างการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI

รูปที่ 2.10การจัดขาของไมโครคอนโทรลเลอร์เบอร์ AT89C2051

รูปที่ 2.11 การจัดขาของไมโครคอนโทรลเลอร์เบอร์ AT89S8253

รูปที่ 2.12 Block Diagram แสดงโครงสร้างของไมโครคอนโทรลเลอร์ MCS 51 เบอร์ AT89C2051

รูปที่ 2.13 Block Diagram แสดงโครงสร้างของไมโครคอนโทรลเลอร์ MCS 51 เบอร์ AT89S8253

รูปที่ 3.1 บล็อกไดอะแกรมของระบบป้ายเฉือกสายรถประจำทาง

รูปที่ 3.2 บล็อกไดอะแกรมของระบบแสดงเวลา

รูปที่ 3.3 โฟลว์ชาร์ตการติดต่อรหว่างไมโครคอนโทรลเลอร์กับไอซี DS1307

รูปที่ 3.4 ลำดับการทำงานของไมโครคอนโทรลเลอร์ในการแสดงเวลาและบอร์ด7-Segment

รูปที่ 3.5 ลำดับการทำงานของไมโครคอนโทรลเลอร์ตัวควบคุมการเลือกสายรถ

รูปที่ 4.1 แสดงสัญญาณควบคุม ISD4003

รูปที่ 4.2 แสดงรูปของกล่องควบคุม

รูปที่ 4.3 การใช้งานร่วมกันของกล่องควบคุมและ 7-Segment module

รูปที่ 4.4 นาฬิกาแสดงเวลาที่ใช้ในโครงงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

**ตารางที่ 2.1 การใช้งานพิเศษของพอร์ต 1 ของ AT89S8253**

**ตารางที่ 2.2 การใช้งานพิเศษของพอร์ต 3 ของ AT89C2051**

**ตารางที่ 2.3 การใช้งานพิเศษของพอร์ต 3 ของ AT89S8953**

**ตารางที่ 2.4 แสดงตำแหน่งแอดเดรสของ SFR และค่าเมื่อรีเซต**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

## 1.1 ความเป็นของโครงงาน

โครงงานนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของอุปกรณ์ที่จะช่วยเหลือกลุ่มผู้พิการทางสายตา เนื่องจากผู้พิการทางสายตาเป็นกลุ่มคนที่ยังต้องดำเนินชีวิตร่วมกับคนทั่วไป แต่อุปกรณ์ส่วนมากไม่ได้เอื้ออำนวยกลุ่มคนเหล่านี้มากนัก ดังนั้นจึงได้มีความคิดที่จะใช้เสียงเป็นสื่อกลางระหว่างผู้พิการทางสายตากับอุปกรณ์ที่จะช่วยอำนวยความสะดวกให้ผู้พิการทางสายตา

## 1.2 วัตถุประสงค์

1) สร้างระบบป้ายจอดรถประทางที่สามารถนำไปใช้ประโยชน์ได้จริงในการช่วยเหลือผู้พิการทางสายตา หรือเป็นต้นแบบให้สามารถนำไปพัฒนาต่อสำหรับผู้ที่สนใจ
2) เพื่อศึกษาการทำงานการเชื่อมต่อไมโครคอนโทรลเลอร์แบบบัส $I^2C$, SPI และ RS-232

## 1.3 ขอบเขตของโครงงาน

1) รับค่าและแสดงผลของค่าที่ต้องการได้ ทั้งในรูปของเสียงและส่วนแสดงผลตัวเลขเจ็ดส่วน โดยใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุม
2) สามารถส่งเสียงบอกเวลา ข้อมูลรถประจำทาง สายรถที่เข้าจอดได้

## 1.4 ขอบเขตของรายงาน

รายงานชุดนี้ประกอบด้วยเนื้อหาอีกสี่บทได้แก่

1) บทที่ 2 ทฤษฎีพื้นฐาน ประกอบไปด้วยการเชื่อมโยงอุปกรณ์ด้วยระบบบัส $I^2C$ ทฤษฎีพื้นฐานการเชื่อมโยงอุปกรณ์แบบ SPI และโครงสร้างและการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51
2) บทที่ 3 การออกแบบ ประกอบไปด้วย การออกแบบระบบการทำงานของป้ายรถประจำทางเพื่อผู้พิการทางสายตาเพื่อผู้พิการทางสายตาและการแสดงผล
3) บทที่ 4 ผลการทดลองประกอบไปด้วย การทำงานของไมโครคอนโทรลเลอร์ การติดต่อกับไอซีสร้างฐานเวลาจริง การติดต่อกับไอซีบันทึกและเล่นเสียง และการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) บทที่ 5 บทสรุป ประกอบด้วยสรุปผลการทดลอง วิเคราะห์ผลการทดลอง และสรุปผลของโครงงาน



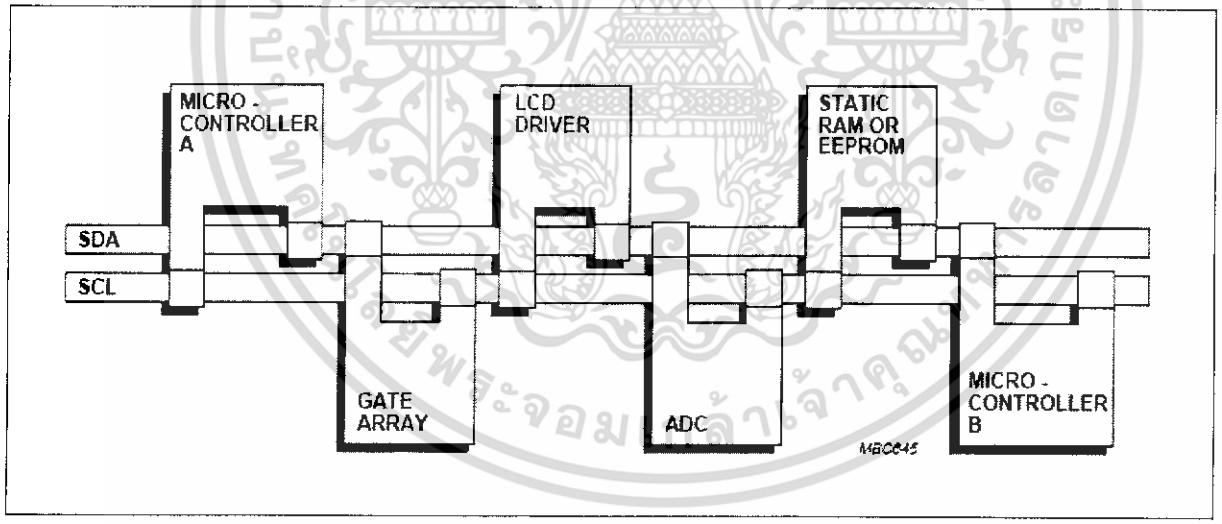
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2
# ทฤษฎี

## 2.1 การเชื่อมโยงอุปกรณ์ด้วยระบบบัส $I^2C$

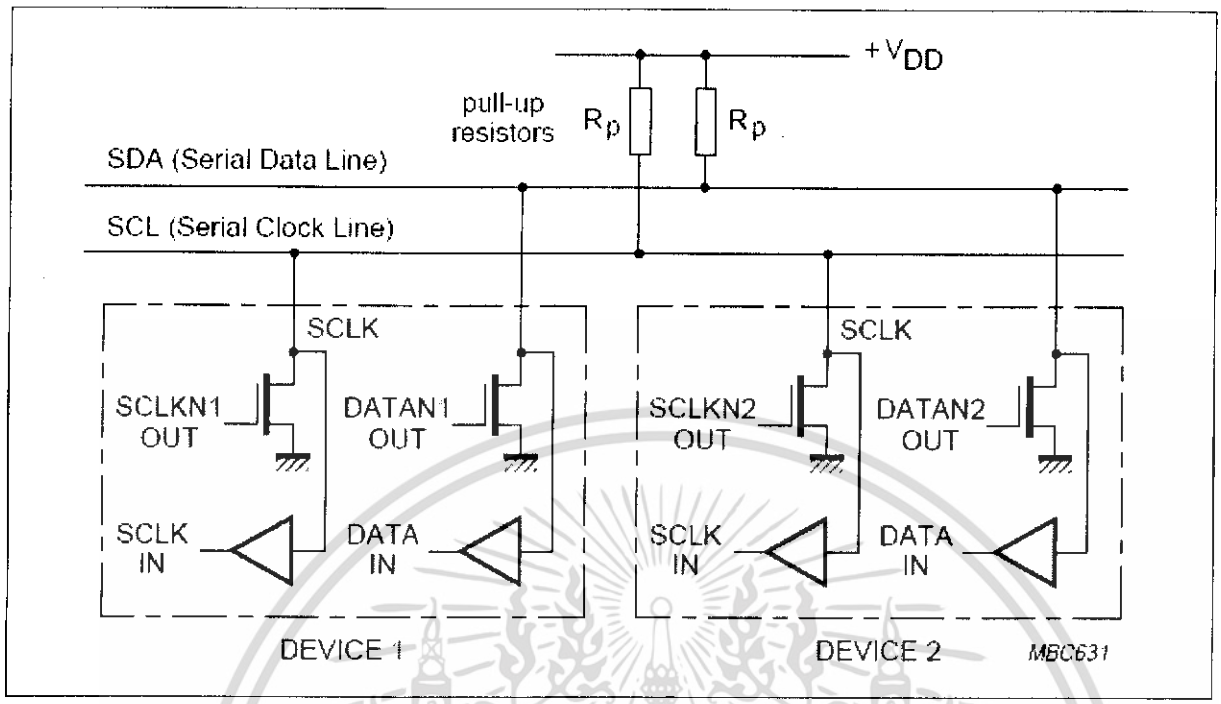
2.1.1 ความรู้เบื้องต้นเกี่ยวกับ $I^2C$

$I^2C$ ย่อมาจากInter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี โดยบัส $I^2C$ ได้รับการพัฒนาขึ้นโดยฟิลิปส์(Philips)ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ สั่งงาน และควบคุมภายใต้สายสัญญาณเพียง2 เส้นเส้นหนึ่งคือ สายข้อมูลอีกเส้นหนึ่งคือ สายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส $I^2C$ ทำได้ง่ายมาก เพียงต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว




รูปที่ 2.1 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่างๆบนรหัสบัส $I^2C$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.2 แสดงวงจรเอาต์พุตของอุปกรณ์ในระบบบัสI2C

สายข้อมูลบนบัส **I²C** มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรมหรือ SDA (Serial Data line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรมหรือSCD (Serial Clock line)ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า สาย SDAและSCL

ในรูปที่ **2.1** แสดงผังของการเชื่อมต่ออุปกรณ์ต่าง ๆ บนบัส $I^2C$ จะเห็นได้ว่า อุปกรณ์ที่ทำการเชื่อมต่อบนบัส $I^2C$ มีหลากหลาย ไม่ว่าจะเป็นไอซีขยายพอร์ตอินพุตเอาต์พุต (I/O Expander) ไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอลADCและแปลงสัญญาณดิจิตอลเป็นอนาลอก (DAC) ไอซีรีลไทม์คล็อก (RTC) ไอซีขับโมดูล LCD หน่วยความจำอีอีพรอม และไมโครคอนโทรลเลอร์

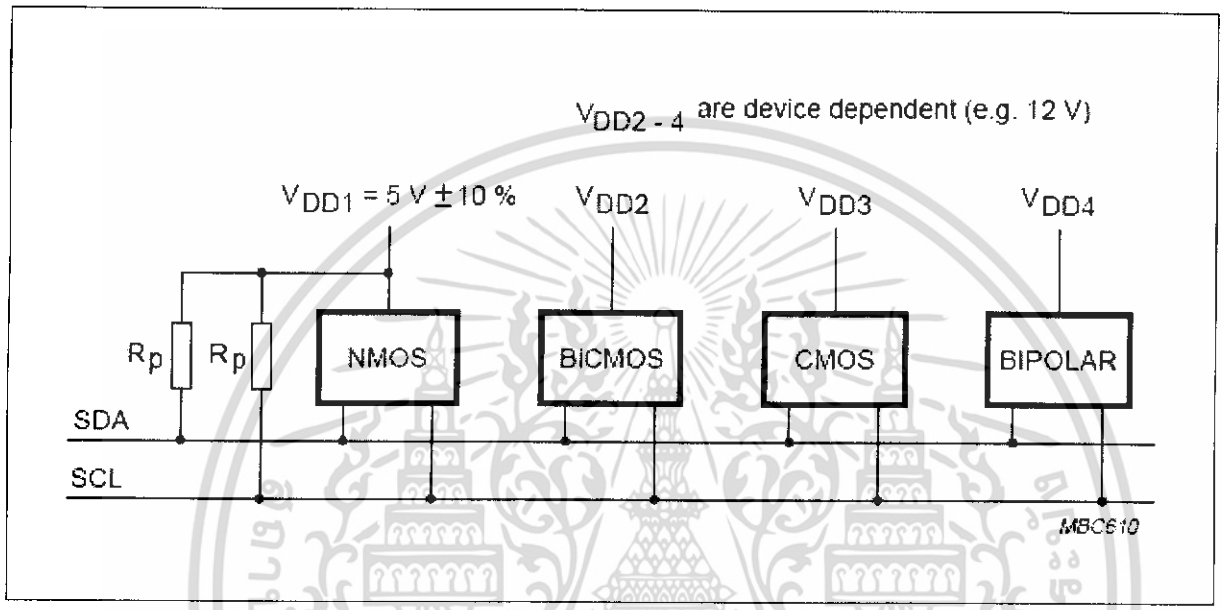
## คุณสมบัติโดยทั่วไปของบัส

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (bi-directional line) ต้องมีการต่อตัวต้านทานพูลอัปกับแรงดัน +5v ไว้ตลอดเวลาเพื่อให้สายมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งาน ทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสองวงจรเอาต์พุต ของอุปกรณ์ที่ติดต่อบนบัส $I^2C$ ต้องมีลักษณะเป็นวงจรเดรนเปิด (open-drain) หรือคอลเล็กเตอร์เปิด (open-collector)

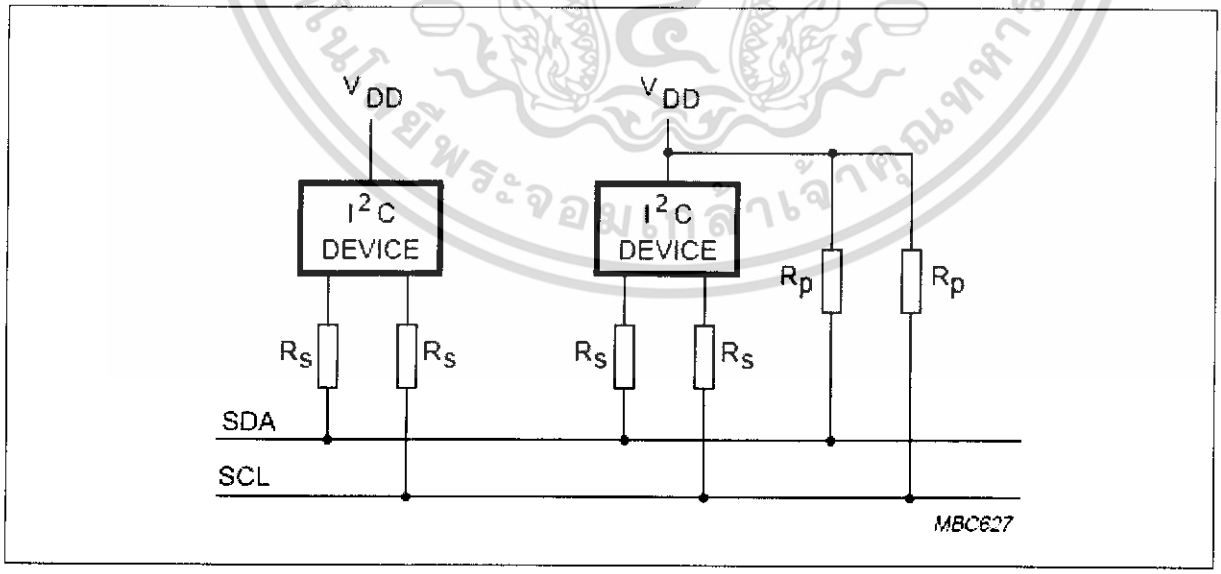
ดังรายละเอียดในรูป **2.2**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราการถ่ายทอดข้อมูลบนบัส $I^2C$ สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ (standard mode) และสูงถึง400กิโลบิตต่อวินาทีในโหมดความเร็วสูง (fast mode) อุปกรณ์ที่ต่อร่วมกันอยู่บนบัส $I^2C$ จะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF การเข้าถึงอุปกรณ์บนบัส $I^2C$ ใช้ข้อมูลสำหรับการเข้าถึง 2 ค่าคือ 7 บิต ( 7-bit addressing) หรือ (10-bit addressing)




รูปที่ 2.3 การต่อตัวต้านทานพูลอัปบนสายสัญญาณในระบบบัส $I^2C$




รูปที่ 2.4 การต่อตัวต้านทาน $R_s$ เพื่อลดสัญญาณรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส $I^2C$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเด่นอีกประการหนึ่งของบัส $I^2C$ คือ การเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลียงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้ โดยอุปกรณ์บนบัส $I^2C$ ตัวหนึ่งอาจใช่ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้ โดยอุปกรณ์บนบัส $I^2C$ ตัวหนึ่งอาจใช้ไฟเลี้ยง +5v ในขณะที่อีกตัวหนึ่งใช้ไฟเลียง +12v การต่อร่วมกันบนบัส $I^2C$ สามารถกระทำได้ในลักษณะเดียวกันกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากัน กล่าวคือ ให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่ออุปกรณ์ต้านทานพูลอัป ($R_p$) เข้ากับแรงดัน+5V ไว้ด้วยเสมอ ดังแสดงในรูปที่ P2-3

ในกรณีที่อาจมีแรงดันไฟฟ้ากระชากขนาดใหญ่ปะปนเข้ามาในบัสI2C ที่ขาSDAและของอุปกรณ์แต่และตัวต้องต่อตัวต้านทานอนุกรมเข้ากับขา SDA และ SCL เรียกว่า $R_s$ ก่อนต่อตัวเข้ากับบัส000 ดัง รูปที่ 2.4

2.1.2 หลักการของบัส $I^2C$

บัส $I^2C$ ประกอบด้วยสายสัญญาณ 2 เส้น ดังที่กล่าวมาแล้วคือ SDA และ SCL อุปกรณ์ที่ต่อพวงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส หรือเรียกว่า โปรโตคอล (protocol) เพื่อให้ผู้ใช่งานได้ทราบว่า ขณะนี้อุปกณ์ใดติดต่อกันอยู่ และอุปกรณ์ตัวใดเป็นตัวรับและตัวส่ง ต่อไปนี้จะขออธิบายลักษณะหน้าที่ และนิยามของอุปกรณ์ที่ต่ออยู่บนบัส $I^2C$ เพื่อเป็นข้อตกลงพื้นฐานก่อที่จะอธิบายการทำงานของบัส $I^2C$ ต่อไป

อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (transmitter)

อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (receiver) อุปกรณ์บนบัสI2Cสามารถเป็นได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส000ที่ทำหน้าที่เป็นตัวส่งเพียงอย่างเดียว

อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส $I^2C$ มาสเตอร์ (master)

อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส $I^2C$ เรียกว่า สเลฟ (slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส $I^2C$ คือ

(1) การถ่ายทอดข้อมูลเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

(2) ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 สภาวะที่เกิดขึ้นบนบัส $I^2C$

มีด้วยกัน 5 สภาวะ ดังนี้

(1) **บัสว่าง (Bus not busy)** สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

(2) **เริ่มต้นการถ่ายทอดข้อมูล (start data transfer)** เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูงเรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะเริ่มต้น (START)

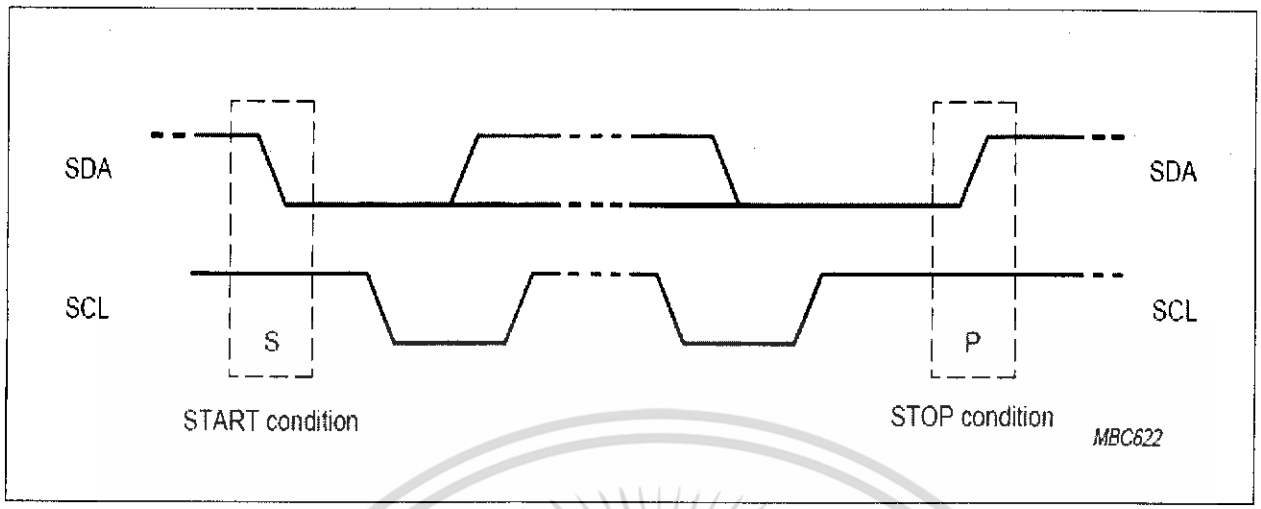
(3) **หยุดการถ่ายทอดข้อมูล (stop data transfer)** เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูงในขณะที่สาย SCL มีสถานะลอจิกสูงเรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะหยุด (stop)

(4) **ข้อมูลคำรงอยู่บนบัส (data valid)** สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอดเมื่อสาย SCL เป็นลอจิกสูงสถานะที่สาย SDA ต้องคงที่เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่า เป็น "0"หรือ"1"ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำแต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ทำให้ข้อมูลที่ทำการถ่ายทอดนั้นเกิดความผิดพลาดขึ้น

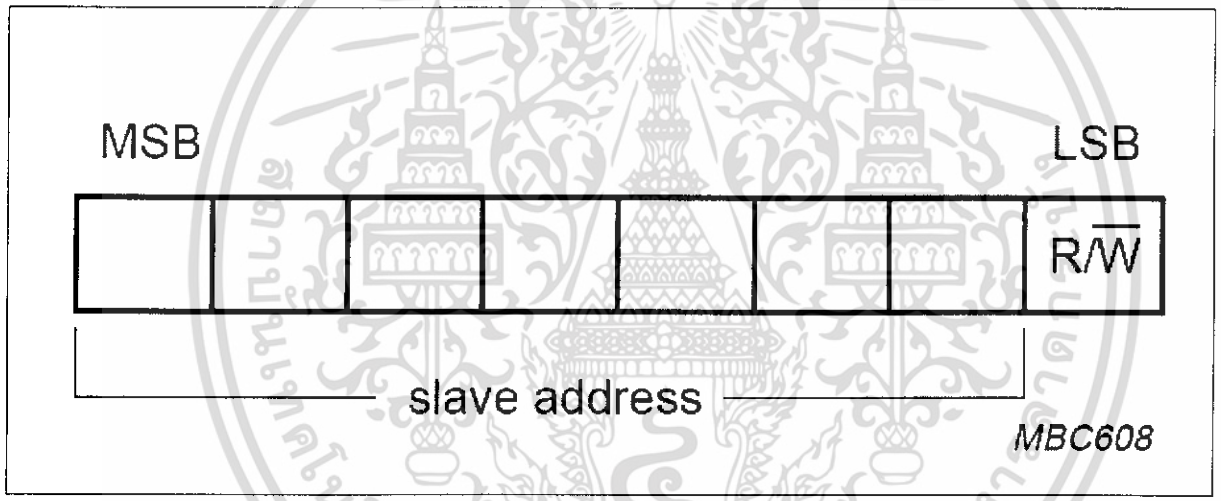
(5) **รับรู้ข้อมูล (acknowledge)** เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิต เรียกว่า บิตรับรู้ (acknowledge bit) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกาเพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่ง ทางด้านตัวรับจะส่งบิตรับรู้ ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลในแต่ละไบต์เรียบร้อยแล้ว

ในรูปที่ 2.5 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสภาวะต่างๆ บนบัส $I^2C$ ไม่ว่าจะเป็นสภาวะบัสว่าง,เริ่มต้น, ถ่ายทอดข้อมูล, รับรู้, และหยุดการถ่ายทอดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.5 ไดอะแกรมเวลาแสดงสถานะต่างๆในบัส $I^2C$




รูปที่ 2.6 รูปแบบของข้อมูลกำหนดแอดเดรสที่ใช้ในการอ้างถึงแบบ 7 บิต

2.1.4 การทำงานบนบัส $I^2C$

ก่อนที่จะเริ่มต้นการถ่ายทอดข้อมูลระหว่างอุปกรณ์ต่างๆ ที่ต่ออยู่บนบัส ต้องมีการอ้างถึงเสียก่อน โดยการอ้างถึงอุปกรณ์บนบัส $I^2C$ นั้นจะใช้การอ้างถึงแบบ 7 บิตหรือ 10 บิต ในกรณีที่มีอุปกรณ์ต่ออยู่บนบัส ไม่มาก ใช้การอ้างถึงแบบ 7 บิต ก็เพียงพอ แต่ถ้ามีอุปกรณ์ต่ออยู่บนบัส มากกว่า 127 แอดเดรส จำเป็นต้องใช้การอ้างถึงแบบ 10 บิต หลังจากที่ติดต่ออุปกรณ์แต่ละตัวได้เรียบร้อยแล้ว ก็จะเริ่มต้นการถ่ายทอดข้อมูลกันต่อไป

ดังนั้นหัวใจสำคัญในอันดับแรกของการทำงานบนบัส $I^2C$ คือการอ้างถึงอุปกรณ์แต่ละตัว ซึ่งในที่นี้จะอธิบายรายละเอียดของการอ้างถึงทั้ง 2 รูปแบบ

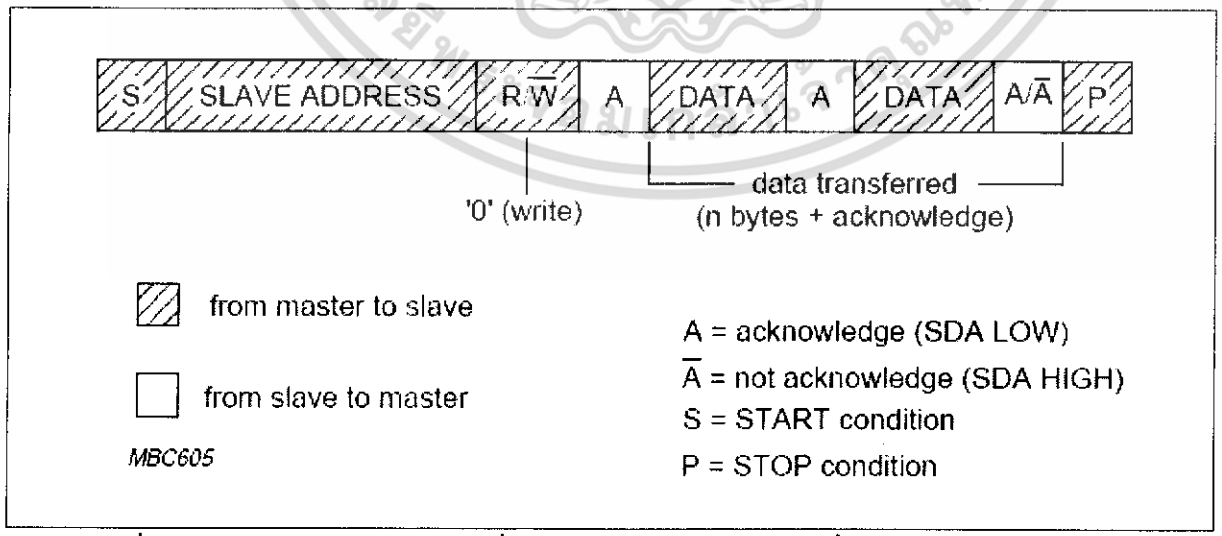
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การอ้างถึงแบบ 7 บิต (7-bit addressing)

ข้อมูลไบต์แรกที่เกิดขึ้นหลังจากสภาวะเริ่มต้นคือ **ข้อมูลที่ใช้ในการอ้างถึงอุปกรณ์ที่ต้องการติดต่อ** หรือ **ข้อมูลกำหนดแอดเดรส** โดยมีรูปแบบแสดงในรูปที่ 2.6 ใน 7 บิตบนรวมทั้งบิต MSB ด้วยจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดแอดเดรสคงที่ (fixed address bit) จำนวน 4 บิต ซึ่งข้อมูลนี้อุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิต ไม่สามารถเปลี่ยนแปลงแก้ไขได้ ถัดมาอีก 3 บิต เป็นบิตกำหนดแอดเดรสที่สามารถโปรแกรมได้ (programmable address bit) โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้แก่ขา A0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัส $I^2C$ ส่วนในบิต LSB เป็นบิตที่ใช้กำหนดการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวนั้นๆ หากบิต LSB เป็น 0 หมายถึงต้องการเขียนข้อมูล ไปยังอุปกรณ์นั้น ถ้าเป็น 1 จะเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ

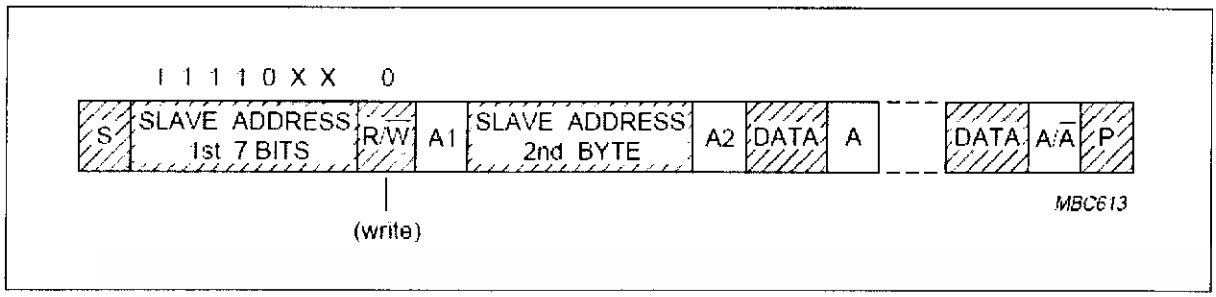
ข้อมูลในไบต์ต่อมา คือ **ข้อมูลควบคุม (control byte )** ในอุปกรณ์แต่ละตัวมีการกำหนดข้อมูลควบคุมที่แตกต่างกันไป ยกตัวอย่าง ไอซีขยายพอร์ต มีข้อมูลควบคุม ที่ใช้กำหนดว่า บิตใดเป็นอินพุต บิตใดเป็นเอาต์พุตในขณะที่ ไอซี ADC/DAC ต้องการข้อมูลควบคุมเพื่อกำหนดให้ทำงานเป็นวงจร ADC หรือ DAC เป็นต้น

ข้อมูลในไบต์ต่อมา คือ **ข้อมูลที่ทำการถ่ายทอดจริง (data)**
หลังจากที่มีการถ่ายทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟ ที่ได้รับการติดต่อ ต้องส่งสัญญาณรับรู้ตอบกลับมาด้วยทุกครั้ง เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้ในรูปที่ P18-7 แสดงรูปแบบข้อมูลอนุกรมที่เกิดขึ้นในการติดต่อบนบัส $I^2C$ ของการอ้างถึงแบบ 7 บิต




รูปที่ 2.7 รูปแบบของข้อมูลอนุกรมที่ใช่ติดต่อกับอุปกรณ์บัสI2C เมื่อใช่การอ้างถึงแบบ 7 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.8 รูปแบบของข้อมูลอนุกรมที่ใช้ติดต่อกับอุปกรณ์บัส I2C เมื่อใช้การอ้างถึงแบบ 10 บิต

2.1.6 การอ้างถึงแบบ 10 บิต

ในการอ้างถึงแบบนี้ ยังคงใช้รูปแบบข้อมูลอนุกรมที่เหมือนกับแบบ 7 บิต หากแต่จะมีข้อมูลเพิ่มเติมขึ้นมาเล็กน้อย โดยในข้อมูลไบต์แรกหลังจากเกิดสภาวะเริ่มต้น **ต้องกำหนดให้ 5 บิตบนมีข้อมูลเป็น 11110** ส่วน อีก 2 บิตถัดมาเป็นบิตแอดเดรสของอุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์แรกยังคงเป็นการกำหนดว่าต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวที่ต้องการติดต่อด้วย ข้อมูลไบต์ต่อมาเป็นข้อมูลแอดเดรสในไบต์ที่ 2 ของอุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนั้นก็จะเป็นข้อมูลจริงที่ใช่ในการติดต่อ

เช่นเดียวกับการอ้างถึงแบบ 7 บิต หลังจากถ่ายทอดข้อมูลครบทุกไบต์ ต้องมีสภาวะรับรู้เกิดขึ้นเพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปในรูปที่ P2-8 แสดงรูปแบบข้อมูลถึงการอ้างถึงแบบ 10 บิต

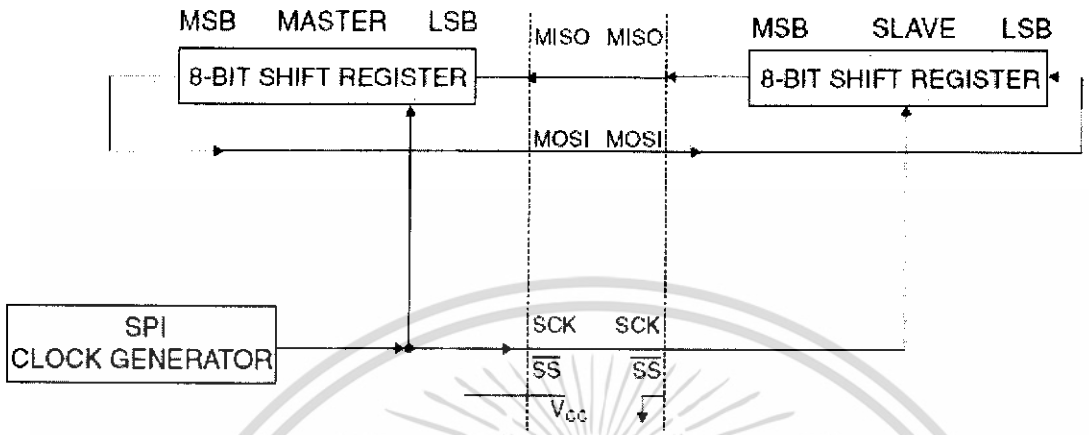
## 2.2 พื้นฐานการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI

2.2.1 การเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI

การเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI เป็นเทคโนโลยีที่คิดขึ้นโดยบริษัทโมโตโรล่า (Motorola Inc.) เพื่อใช้ในการเชื่อมโยงอุปกรณ์ต่างๆเข้าด้วยกัน เพื่อรับ-ส่งสัญญาณแบบซิงโครนัส โดยใช้สายสัญญาณ 3 หรือ 4 เส้น(ขึ้นอยู่กับความต้องการของระบบที่จะใช้) ซึ่งเมื่อเทียบกับบการรับส่งแบบ $I^2C$ ที่เป็นการรับส่งสัญญาณแบบซิงโครนัสเช่นกัน จะใช้จำนวนสายสัญญาณมากกว่าการเชื่อมโยงแบบ $I^2C$ แต่ความเร็วในการรับ-ส่งที่ได้จะมากกว่าแบบ $I^2C$ 5-50 เท่า ทั้งนี้ขึ้นอยู่กับอุปกรณ์ที่นำมาต่อร่วม ดังนั้นจึงนิยมนำมาใช้ในงานประเภทการแปลงสัญญาณระหว่างอนาล็อกและดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 โครงสร้างการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI




รูปที่ 2.9 โครงสร้างการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI

รูปที่ 2.9 เป็นโครงสร้างพื้นฐานของการเชื่อมโยงอุปกรณ์แบบอนุกรมแบบ SPI ที่ตั้งอยู่บนพื้นฐานของอุปกรณ์ประเภทรีจิสเตอร์เลื่อนข้อมูลชนิด 8 บิต( 8-BIT SHIFT REGISTER) อุปกรณ์ที่เป็นตัวควบคุม (Master) และอุปกรณ์ที่นำมาต่อร่วม (Slave) เชื่อมโยงกันด้วยสายสัญญาณสามเส้น คือ *SCK* (Serial Clock), *MISO* (Master-In Slave-Out) และ *MOSI* (Master-Out Slave-In) โดยที่ SCK จะให้สัญญาณนาฬิกา (Clock) ออกมาเฉพาะในระหว่างที่มีการรับ-ส่งข้อมูล โดยอุปกรณ์ที่เป็นตัวควบคุมจะเป็นตัวกำเนิดสัญญาณนาฬิกาออกมา และการรับ-ส่งข้อมูลจะเกิดขึ้นพร้อมกัน คือในขณะที่มีการส่งข้อมูลออกจากตัวควบคุม อุปกรณ์ที่เป็นตัวต่อร่วมก็จะส่งข้อมูลกลับมาที่ตัวควบคุมในเวลาเดียวกัน ส่วนสายสัญญาณ *SS*(Slave Select) มีไว้เพื่อกำหนดสถานะของอุปกรณ์ที่ต่อร่วมกันว่าจะให้อุปกรณ์ตัวไหนทำหน้าที่เป็นอุปกรณ์ที่นำมาต่อร่วม(Slave) การส่งข้อมูลจะเริ่มจากการส่งบิตที่มีความสำคัญมากที่สุด(Most Significant Bit : MSB) ออกไปก่อนทุกครั้ง

## 2.3 ทฤษฎีไมโครคอนโทรลเลอร์ MCS-51

2.3.1 ลักษณะสำคัญของไมโครคอนโทรลเลอร์ตระกูลMCS-51

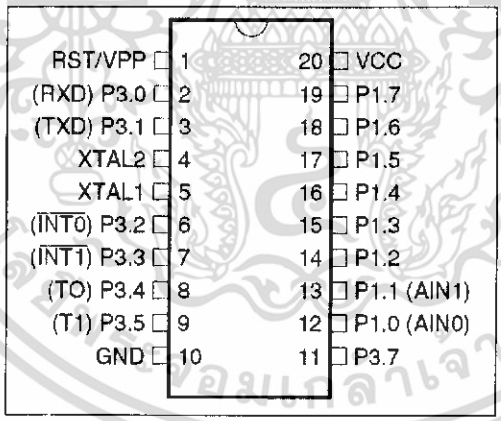
1. เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
2. ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลช สามารถลบและเขียนใหม่ได้พันครั้ง
3. ทำงานในย่านแรงดันตั้งแต่ 2.7V ถึง 6V (ขึ้นอยู่กับเบอร์ของไมโครคอนโทรลเลอร์ที่นำมาใช้งาน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. มีพอร์ทอินพุท เอาท์พุทให้เขียนโปรแกรมใช้งานได้
5. การสื่อสารผ่านพอร์ทอนุกรมเป็นแบบ UART และ SPI (มีเฉพาะบางเบอร์ของไมโครคอนโทรลเลอร์)
6. มี Timer/counter ขนาด 16 bit
7. สามารถเกิดการอินเตอร์รัปต์ได้อย่างน้อยจาก 6 แหล่งกำเนิดและอาจมากกว่าขึ้นอยู่กับเบอร์ของไมโครคอนโทรลเลอร์ที่นำมาใช้งาน

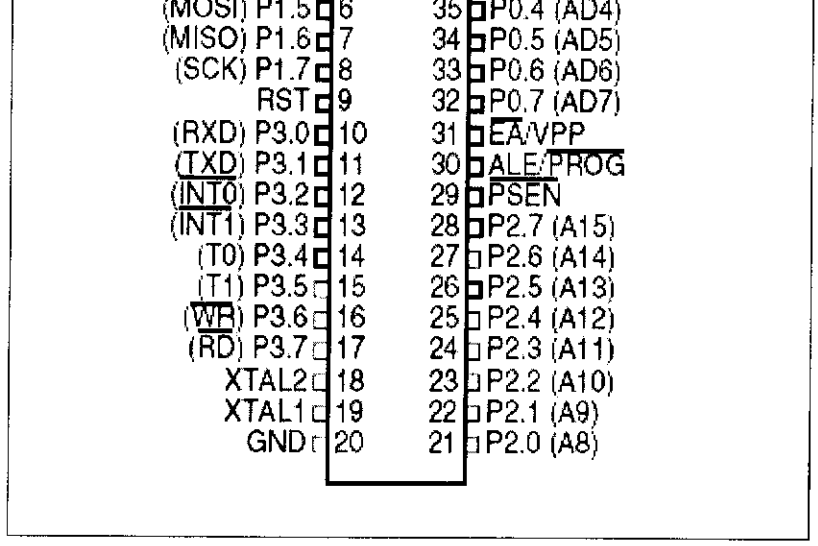
2.3.2 การจัดขาและโครงสร้างของไมโครคอนโทรลเลอร์

โครงสร้างของไมโครคอนโทรลเลอร์จะมีความคล้ายคลึงกันระหว่างไมโครคอนโทรลเลอร์แบบ 40 ขา(AT89S8253) กับไมโครคอนโทรลเลอร์แบบ 20 ขา (AT89C2051) ซึ่งไมโครคอนโทรลเลอร์แบบ 20 ขา จะตัดพอร์ท 0 กับพอร์ท 2 ออก เหลือให้ใช้งาน พอร์ท 1 กับ พอร์ท 3 เท่านั้นและนอกจากนี้ไมโครคอนโทรลเลอร์ยังได้นำพอร์ท 3 ไปใช้สำหรับเป็น Analog Comparator จึงไม่มีการนำไปใช้เป็นI/O พอร์ท โครงสร้างและการจัดขาของไมโครคอนโทรลเลอร์เบอร์ AT89C2051 แสดงได้ดัง รูปที่ 2.10




รูปที่ 2.10การจัดขาของไมโครคอนโทรลเลอร์เบอร์ AT89C2051

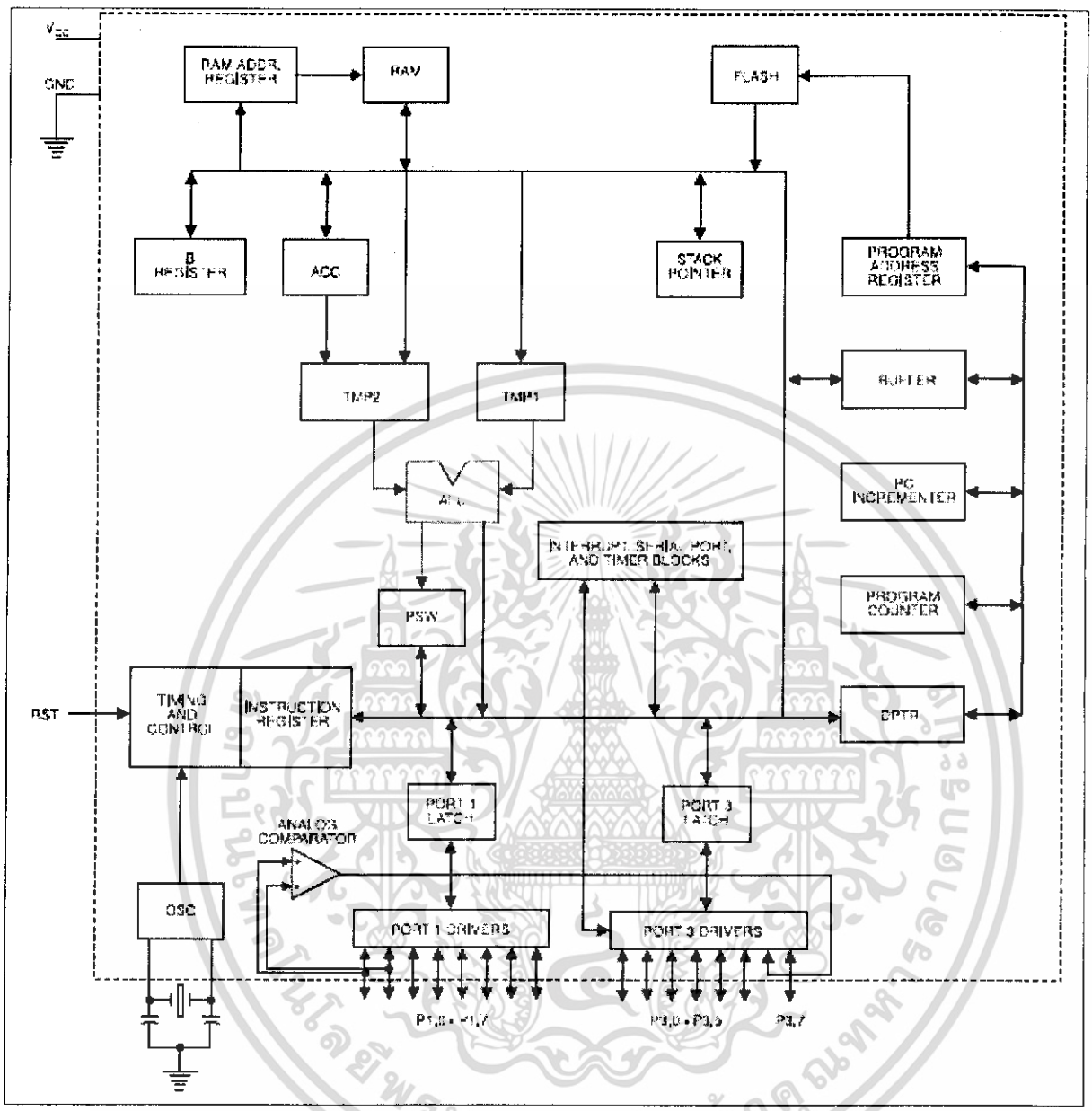
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.11 การจัดขาของไมโครคอนโทรลเลอร์เบอร์ AT89S8253

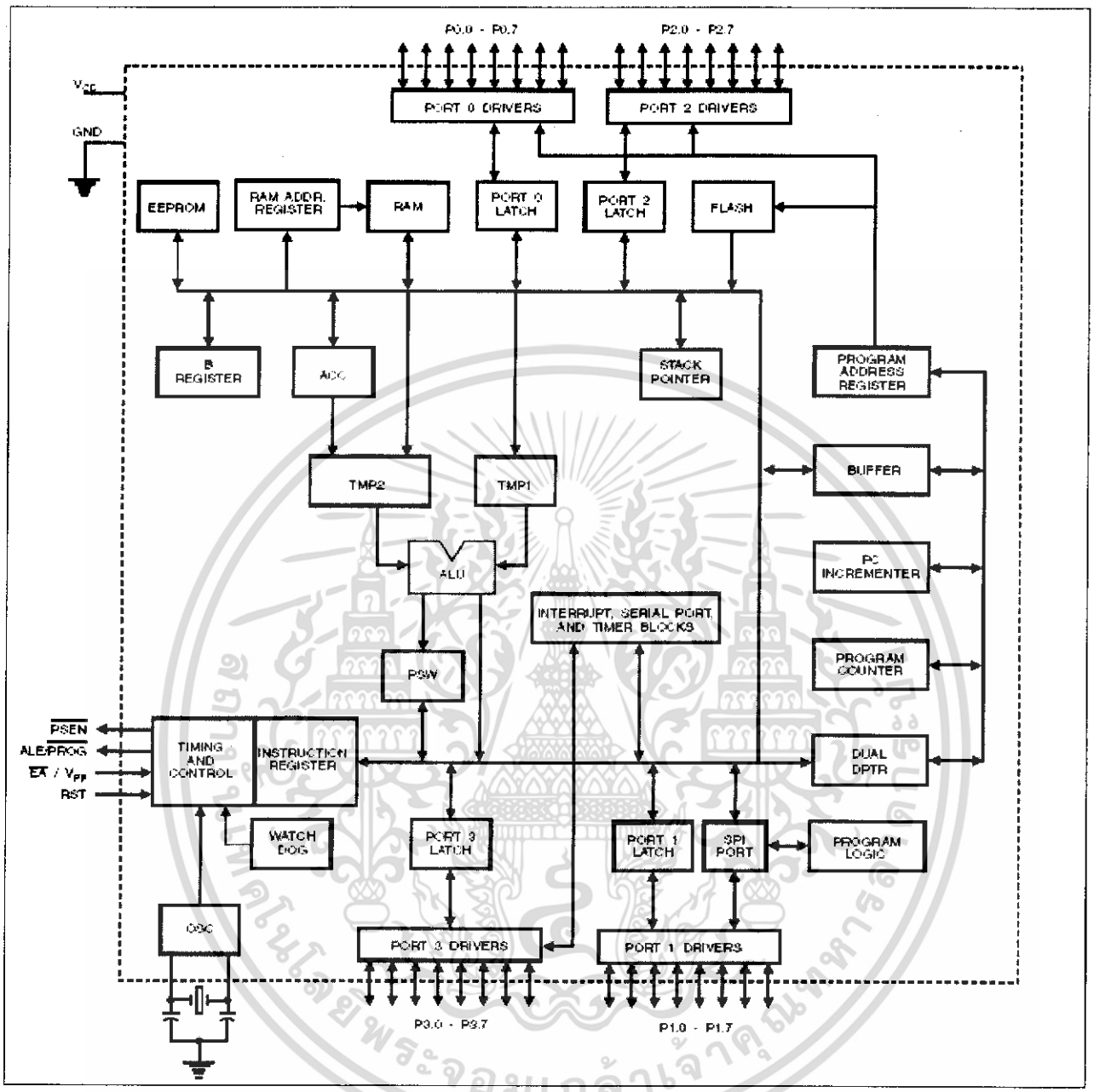


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.12 Block Diagram แสดงโครงสร้างของไมโครคอนโทรลเลอร์ MCS 51 เบอร์ AT89C2051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.13 Block Diagram แสดงโครงสร้างของไมโครคอนโทรลเลอร์ MCS 51 เบอร์ AT89S8253

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Pin Description**

**1. ขาVCC** ใช้สำหรับต่อไฟเลี้ยง

**2. ขาGND** เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

**3. ขาพอร์ต 0 (P0.0-P0.7)** มีขา 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุท สามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย

**4. ขาพอร์ต 1 (P1.0-P1.7)** มีขา 8 ขา มี pull ups ในพอร์ต แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุท สามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย

| Port Pin | Alternate Functions |
|---|---|
| P1.0 | T2 (external count input to Timer/Counter 2), clock-out |
| P1.1 | T2EX (Timer/Counter 2 capture/reload trigger and direction control) |
| P1.4 | $\overline{SS}$ (Slave port select input) |
| P1.5 | MOSI (Master data output, slave data input pin for SPI channel) |
| P1.6 | MISO (Master data input, slave data output pin for SPI channel) |
| P1.7 | SCK (Master clock output, slave clock input pin for SPI channel) |

ตารางที่ 2.1 การใช้งานพิเศษของพอร์ต 1 ของ AT89S8253

**5. ขาพอร์ต 2 (P2.0-P2.7)** มี 8 ขา มี pull ups ในพอร์ต แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย

**6. ขาพอร์ต 3 (P3.0-P3.7)** มี 8 ขา มี pull ups ในพอร์ต แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ต 3 ยังเป็นขาที่มีหน้าที่การใช้งานพิเศษ มีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

| Port Pin | Alternate Functions |
|---|---|
| P3.0 | RXD (serial input port) |
| P3.1 | TXD (serial output port) |
| P3.2 | $\overline{\text{INT0}}$ (external interrupt 0) |
| P3.3 | $\overline{\text{INT1}}$ (external interrupt1) |
| P3.4 | T0 (timer 0 external input) |
| P3.5 | T1 (timer 1 external input) |

**ตารางที่ 2.2 การใช้งานพิเศษของพอร์ต 3 ของ AT89C2051**

| Port Pin | Alternate Functions |
|---|---|
| P3.0 | RXD (serial input port) |
| P3.1 | TXD (serial output port) |
| P3.2 | $\overline{\text{INT0}}$ (external interrupt 0)[1] |
| P3.3 | $\overline{\text{INT1}}$ (external interrupt 1)[1] |
| P3.4 | T0 (timer 0 external input) |
| P3.5 | T1 (timer 1 external input) |
| P3.6 | $\overline{\text{WR}}$ (external data memory write strobe) |
| P3.7 | $\overline{\text{RD}}$ (external data memory read strobe) |

ตารางที่ 2.3 การใช้งานพิเศษของพอร์ต 3 ของ AT89S8952

**7. ขารีเซต (Reset)** ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ ทุกขา I/O จะเป็น "1"เมื่อขา RST อยู่ในสถานะ Highโดยใช้การป้อนสัญญาณเพื่อรีเซตสถานะที่ขานี้ต้องอยู่ในระดับรีเซตอย่างน้อย 2 แมชชีนไซเคิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ

**8. ขา XTAL1 และ XTAL2** เป็นขาสำหรับต่อคริสตัลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

72127
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Special Function Registers**

เป็นรีจิสเตอร์ที่ทำหน้าที่ควบคุมการทำงานของอุปกรณ์ หรือพอร์ตของไมโครคอนโทรล- เลอร์ ทั้งหมดโดยมีตำแหน่งอยู่ในบริเวณแอดเดรส 80H-FFH ดังแสดงในตาราง ที่ 2.4 การใช้งานรีจิสเตอร์หน้าที่พิเศษเหล่านี้สามารถทำได้ทั้งการระบุถึงชื่อรีจิสเตอร์หรือตำแหน่งแอดเดรสที่เป็นของรีจิสเตอร์นั้นก็ได้ นอกจากนี้ในตารางที่ 2.4 ยังได้แสดงค่าของรีจิสเตอร์หน้าที่พิเศษนี้เมื่อได้รับสัญญาณรีเซตไว้อีกด้วย รายละเอียดของรีจิสเตอร์พิเศษ มีดังต่อไปนี้

1.แอกคิวมูเลเตอร์ (Accumulator) เป็นรีจิสเตอร์ขนาด 8 บิต ทำหน้าที่เก็บข้อมูลที่ส่งให้หน่วยทำงานในซีพียูและเก็บผลลัพธ์ที่ได้จากการทำงานนั้น การใช้งานในโปรแกรมจะเรียกว่า รีจิสเตอร์ A

2. รีจิสเตอร์ B เป็นรีจิสเตอร์ที่ใช้สำหรับการทำคำสั่งการคูณหารตัวเลข ในกรณีที่ไม่ใช้การคำนวณทางด้านคณิตศาสตร์ ก็สามารถนำไปใช้งานเช่นเดียวกับรีจิสเตอร์ทั่วไปได้

3.โปรแกรมเคาน์เตอร์ (Program Counter) เป็นรีจิสเตอร์ที่ใช้สำหรับการชี้ตำแหน่งแอดเดรสของหน่วยความจำโปรแกรม ซึ่งจะต้องไปทำงานในลำดับถัดไป การใช้งานในโปรแกรมจะเรียกว่า รีจิสเตอร์ PC

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0F8H | | | | | | | | 0FFH |
| 0F0H | B<br>00000000 | | | | | | | 0F7H |
| 0E8H | | | | | | | | 0EFH |
| 0E0H | ACC<br>00000000 | | | | | | | 0E7H |
| 0D8H | | | | | | | | 0DFH |
| 0D0H | PSW<br>00000000 | | | | | | | 0D7H |
| 0C8H | | | | | | | | 0CFH |
| 0C0H | | | | | | | | 0C7H |
| 0B8H | IP<br>XXX00000 | | | | | | | 0BFH |
| 0B0H | P3<br>11111111 | | | | | | | 0B7H |
| 0A8H | IE<br>0XX00000 | | | | | | | 0AFH |
| 0A0H | | | | | | | | 0A7H |
| 98H | SCON<br>00000000 | SBUF<br>XXXXXXXX | | | | | | 9FH |
| 90H | P1<br>11111111 | | | | | | | 97H |
| 88H | TCON<br>00000000 | TMOD<br>00000000 | TL0<br>00000000 | TL1<br>00000000 | TH0<br>00000000 | TH1<br>00000000 | | 8FH |
| 80H | | SP<br>00000111 | DPL<br>00000000 | DPH<br>00000000 | | | | PCON<br>0XXX0000 | 87H |

ตารางที่ 2.4 แสดงตำแหน่งแอดเดรสของ SFR และค่าเมื่อรีเซต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. สแต็กพอยน์เตอร์ (Stack Pointer) เป็นรีจิสเตอร์ขนาด 8 บิต ทำหน้าที่เก็บตำแหน่งของตัวชี้หรือพอยน์เตอร์ของบริเวณสแต็กสำหรับเก็บข้อมูลแอกคิวมูเลเตอร์ รีจิสเตอร์ต่างๆ รวมทั้งข้อมูลจากโปรแกรม ค่าเริ่มต้นของสแต็กจะอยู่ที่ตำแหน่ง 07H การใช้งานในโปรแกรมจะเรียกว่ารีจิสเตอร์ SP

5. ตัวชี้ข้อมูลหรือดาต้าพอยน์เตอร์ (Data Pointer) เป็นรีจิสเตอร์ขนาด 16 บิต ซึ่งเรียกว่า รีจิสเตอร์ DPTR และสามารถใช้งานแยกออกเป็นรีจิสเตอร์ขนาด 8 บิต สองตัว คือ รีจิสเตอร์ DPH และ DPL เพื่อเก็บค่าแอดเดรสของหน่วยความจำที่จะต้องใช้งานภายในโปรแกรม หรืออาจเป็นแอดเดรสของอุปกรณ์ภายนอก

6. โปรแกรมสเตตัสเวิร์ด (PSW) รีจิสเตอร์นี้ทำหน้าที่บอกถึงแฟลกสภาวะการทำงานต่างๆ รวมทั้งบิตสำหรับการกำหนดเลือกแบงก์ (Bank) ของรีจิสเตอร์ที่ใช้งานด้วย

7. รีจิสเตอร์ที่เกี่ยวข้องกับพอร์ต(Port Register) รีจิสเตอร์เหล่านี้จะมีความเกี่ยวข้องกับการทำงานของพอร์ตอินพุทเอาท์พุทโดยตรงซึ่งจะเป็นรีจิสเตอร์ขนาด 8 บิต สามารถใช้งานได้ทั้งในลักษณะการอินพุทหรือการเอาท์พุทข้อมูลได้

8. รีจิสเตอร์ SBUFเป็นบัฟเฟอร์ขนาด 8 บิต สำหรับการสื่อสารข้อมูลแบบอนุกรมทั้งการรับและการส่งข้อมูล

9. รีจิสเตอร์ PCON เป็นรีจิสเตอร์ควบคุมการทำงานในสามลักษณะ ซึ่งได้แก่ การควบคุมการทำงานของโปรแกรมเซ็นเซอร์ การกำหนดอัตราทวีคูณของอัตราเร็วในการสื่อสารข้อมูลอนุกรมและแฟลกสภาวะการทำงานทั่วไป

10. รีจิสเตอร์ IP, IE, TMOD, SCON เป็นกลุ่มรีจิสเตอร์ที่ทำหน้าที่ควบคุมการทำงานของอินเตอร์รัปต์ต่างๆ

2.3.3 พอร์ทอนุกรมของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์ 1 ชุด(วงจรสื่อสารแบบฟูลดูเพล็กซ์ หมายถึง วงจรสื่อสารที่สามารถทำการรับและส่งข้อมูลในลักษณะ 2 ทิศทางได้ในเวลาเดียวกัน) โดยใช้ขาสัญญาณของพอร์ท 3 คือ ขา P3.0 เป็นขารับข้อมูลเข้าหรือ RxD และขา P3.1 เป็นขาส่งข้อมูลออกหรือ TxD โดยวงจรสื่อสารข้อมูลแบบอนุกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-51 แบบแฟลชเป็นแบบอะซิงโครนัสปกติแล้วพอร์ทอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 จะใช้ในการติดต่อสื่อสารกับพอร์ทอนุกรมของคอมพิวเตอร์ โดยใช้มาตรฐาน RS-232 แต่ในปัจจุบันสามารถติดต่อกันในมาตรฐาน RS-422 หรือ RS-485 ได้แล้วโดยใช้ไอซีพิเศษที่ทำหน้าที่ในการแปลงสัญญาณการสื่อสารดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การสื่อสารข้อมูลแบบอะซิงโครนัส**

การสื่อสารข้อมูลแบบอะซิงโครนัสคือการรับและส่งข้อมูลโดยไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยแต่จะใช้การกำหนดค่าอัตราเร็วในการรับและส่งข้อมูลให้มีค่าเท่ากัน ซึ่งเรียกอัตราเร็วนี้ว่าอัตราบอดหรือบอดเรต(Baud rate) มีหน่วยเป็นบิตต่อวินาที (bit per second: bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนด้วยกันคือ 1. บิตเริ่มต้น (start bit) มีขนาด 1บิต

2. บิตข้อมูลแบบอนุกรม มีขนาด 8 บิต

3. บิตตรวจสิบพาริตี้ (parity bit) มีขนาด 1 บิตหรือไม่มี

4. บิตปิดท้ายหรือบิตหยุด (stop bit) มีขนาด 1 บิต

อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัสหรืออัตราบอดหรือบอดเรตที่ใช้สำหรับพอร์ทอนุกรม RS-232 มีด้วยกันหลายค่า ตั้งแต่ 110 ถึง 19,200 บิตต่อวินาที โดยมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ เนื่องจากอัตราบอดคือค่าของจำนวนบิตที่สามารถส่งไปได้ใน 1 วินาที สมมติว่า ข้อมูลอนุกรมมีขนาด 8 บิต ไม่มีการตรวจสอบพาริตี้ มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูล 1 ไบต์จะมีความยาวเท่ากับ 10บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9,600 บิตต่อวินาที ก็จะสามารถรับส่งข้อมูลได้ด้วยความเร็ว 960 ไบต์ต่อวินาที

การตรวจสิบพาริตี้สามารถกำหนดให้เป็นแบบคี่ (odd), แบบคู่ (even) หรือไม่มีการตรวจสิบพาริตี้ก็ได้ พาริตี้คี่หรือพาริตี้คู่แสดงถึงจำนวนลอจิก "1" ทั้งหมดภายในข้อมูลที่ส่งไป 1 ไบต์ รวมบิตพาริตี้ว่ามีจำนวนเป็นเลขคู่หรือเลขคี่ ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิต มีค่าเท่ากับ 99H หรือ 10011001B จะเห็นว่าข้อมูลในไบต์นี้มีจำนวนลอจิก "1" จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตี้เป็นคู่ ค่าของบิตพาริตี้จะต้องมีลอจิกเป็น "0" แต่ถ้ากำหนดพาริตี้เป็นคี่ ค่าของบิตพาริตี้จะต้องเป็น "1" เพื่อให้ข้อมูล 1ไบต์ รวมทั้งบิตพาริตี้เป็นคี่

บิตพาริตี้ถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART (Universal Asynchronous Receiver Transmitter: เป็นอุปกรณ์ที่ใช้ในการรับและส่งข้อมูลอนุกรม) ซึ่งทางภาครับต้องกำหนดการตรวจสอบพาริตี้ที่ตรงกันเอาไว้ว่าจะตรวจสอบพาริตี้คี่หรือคู่ จากนั้นภาครับของ UART จะทำการตรวจสอบค่าพาริตี้ที่เกิดขึ้นว่าเป็นคู่หรือเป็นคี่ โดยการนับจำนวนลอจิก "1" ทั้งหมดรวมทั้งบิตพาริตี้ด้วย ถ้ากำหนดพาริตี้ไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้ตัวเลขเป็นคี่ ทางภาครับจะแสดงข้อผิดพลาดออกมาให้ผู้ใช้ทราบ กระบวนการดังกล่าวเป็นวิธีการตรวจสอบความผิดพลาดที่เกิดขึ้นในการรับส่งข้อมูลที่ง่ายที่สุด แต่มันสามารถตรวจสอบได้เมื่อมีบิตข้อมูลที่ทำการส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผลสำหรับการตั้งพาริตี้เป็น NONE นั้นทั้งภาครับและส่งจะไม่มีการตรวจสอบพาริตี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

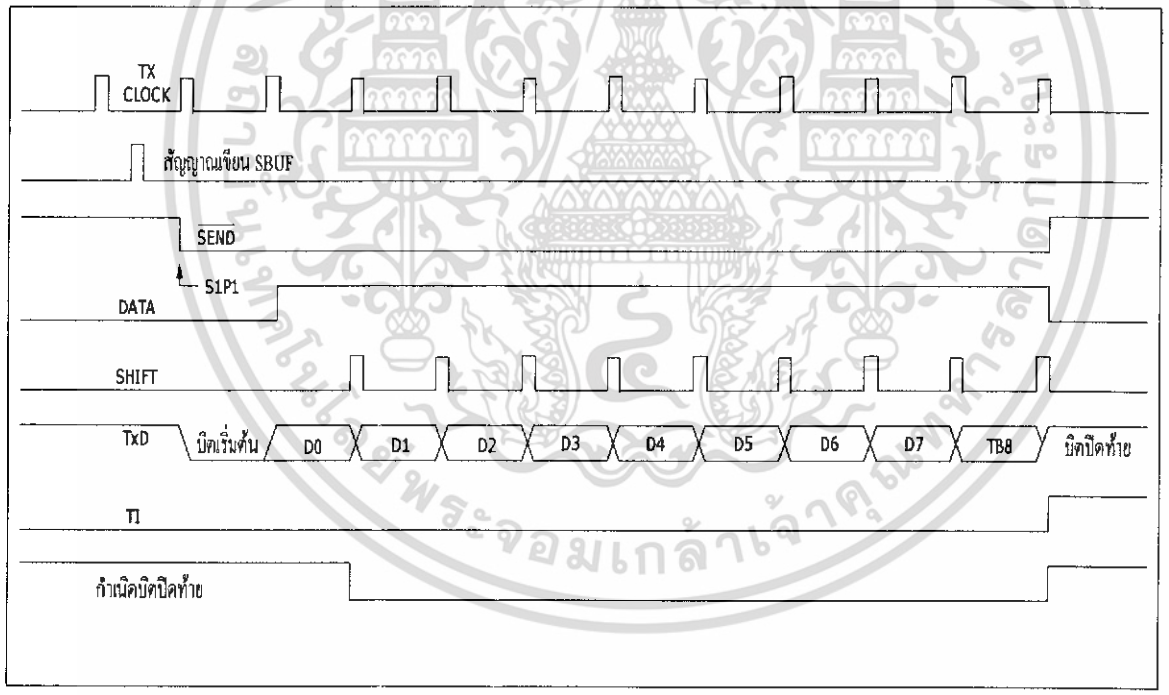
**โหมดการทำงานของพอร์ทอนุกรมใน MCS-51**

พอร์ทอนุกรมในไมโครคอนโทรลเลอร์ MCS-51 สามารถเลือกการทำงานได้ถึง 4 โหมด คือ

1. โหมด 0 เป็นการกำหนดให้พอร์ทอนุกรมทำงานในลักษณะชิฟต์รีจิสเตอร์
2. โหมด 1 เป็นการกำหนดให้เป็น UART ขนาด 8 บิต สามารถเลือกอัตราบอดได้
3. โหมด 2 เป็นการกำหนดให้เป็น UART ขนาด 9 บิต โดยมีอัตราบอดคงที่
4. โหมด 3 เป็นการกำหนดให้เป็น UART ขนาด 9 บิต สามารถเลือกอัตราบอดได้

การเลือกโหมดทำได้ด้วยการกำหนดข้อมูลให้แก่บิต SM0 และ SM1 ในรีจิสเตอร์ SCON ในที่นี้จะขอกล่าวถึงการทำงานของพอร์ทอนุกรมในโหมด 1 เท่านั้น เนื่องจากการทำงานในโหมดนี้เป็นที่นิยมและได้เลือกใช้ในโครงงานนี้

การทำงานในโหมด 1 ของวงจรพอร์ทอนุกรมมีไดอะแกรมทางเวลาแสดงดังรูปที่ **2.14**




รูปที่ 2.14 ไทมิ่งไดอะแกรมของวงจรพอร์ทอนุกรมขณะทำการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3

# การออกแบบ

## 3.1 ขอบเขตของการออกแบบ

1) ออกแบบและสร้างระบบป้ายเลือกสายรถประจำทางโดยมีไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงานของระบบ

3) สามารถกำหนดสายรถที่ให้เลือกได้ผ่านทางสวิตส์แมตทริก และสามารถตรวจสอบสายรถจำทางที่มาถึงที่ป้ายได้ พร้อมทั้แสดงเวลาปัจจุบัน

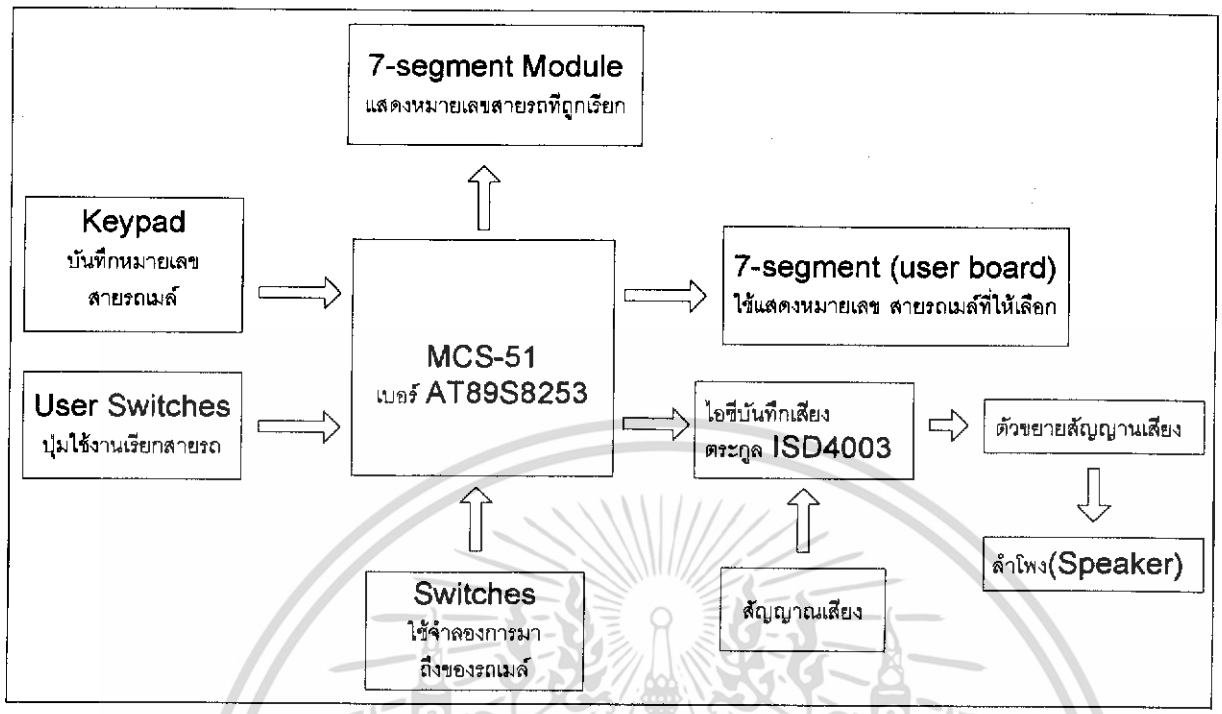
2.) แสดงผลการทำงานด้วยเสียงและจอแสดงผลทั้งในส่วนของป้ายรถประจำทางและเวลาผ่านไอซีบันทึก/เล่นเสียงตระกูล ISD4003และตัวเลขเจ็ดส่วน (7-Segment)

วงจรต่างๆที่ใช้ในโครงงานสามารถดูได้จากภาคผนวก

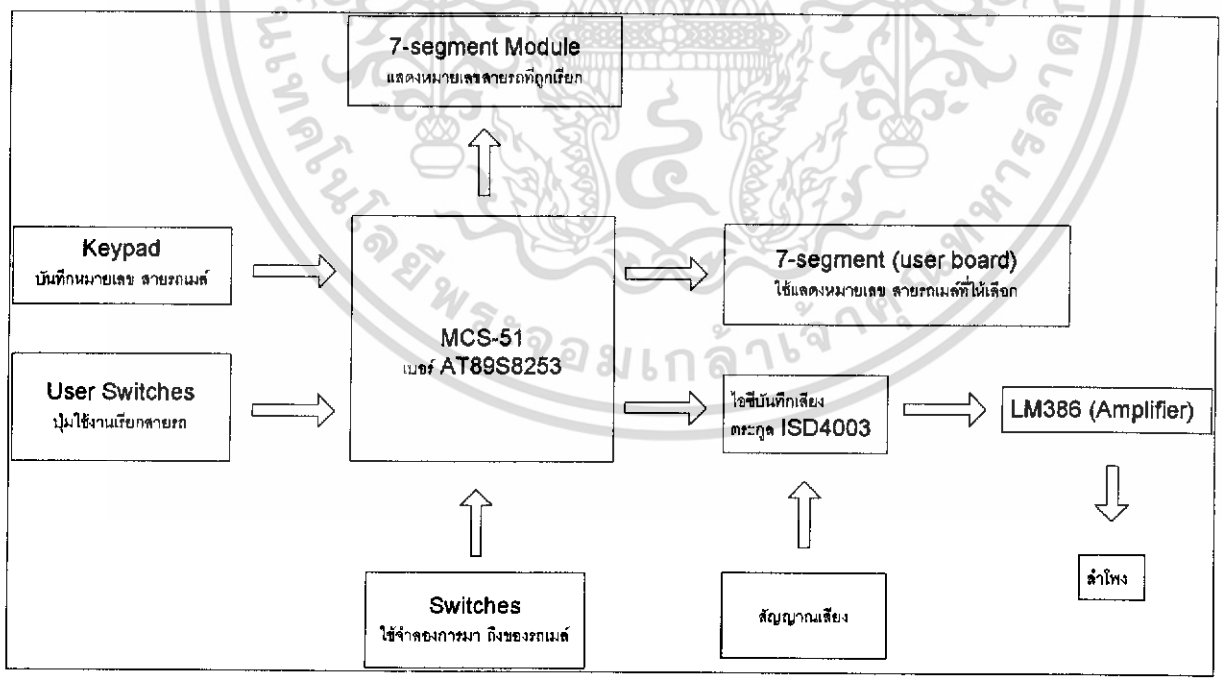
## 3.2 การออกแบบระบบป้ายเลือกสายรถประจำทาง

บล็อกไดอะแกรมของระบบป้ายเลือกสายรถประจำทางแสดงได้ดังรูปที่ 3.1 การทำงานของระบบจะเริ่มโดยผู้ออกแบบจะทำการบันทึกหมายเลขสายรถเมล์ที่ต้องการใช้งานลงในไมโครคอนโทรลเลอร์ ผ่านทาง Keypad จากนั้น ไมโครคอนโทรลเลอร์จะทำการแสดงผลหมายเลขสายรถเมล์นั้นๆด้วย 7-segment เพื่อให้ผู้ใช้ได้เลือกสายรถเมล์ได้ถูกต้อง พร้อมกันนี้จะส่งเลขสายรถเมล์ที่ได้ตั้งค่าไปยังบอร์ดแสดงผล 7-segment module ที่อยู่ด้านบนซึ่งใช้แสดงผลให้พนักงานขับรถและผู้ใช้งานป้ายรถประจำทางคนอื่นทราบผ่านทางพอร์ทอนุกรมของไมโครคอนโทรเลอร์ โดยที่บอร์ด 7-segment module นี้จะมีไมโครคอนโทรลเลอร์อยู่ในวงจรเพื่อทำการเก็บค่าสายรถเมล์ที่ตั้งแต่ยังไม่นำค่าที่ได้มาแสดงผลในทันที และเมื่อผู้ใช้กดสวิตช์เลือกสายรถเมล์ที่ต้องการแล้ว ISD 4003 ก็จะทำงานโดยการส่งสัญญาณเสียงผ่านทางลำโพงบอกหมายเลขสายรถเมล์ที่ถูกเลือกนั้นเพื่อบอกแก่ผู้พิการทางสายตาได้รับรู้ว่าสายที่เลือกเป็นสายอะไรพร้อมทั้งบอกว่าเดินทางจากไปนไปแทนการแสดงผลเป็นตัวเลข และเมื่อผู้พิการได้ฟังเสียงแล้ว หากต้องการเลือกสายนั้นสามารถทำได้โดยการกดปุ่มเลือกสายนั้นๆค้างไว้จนกว่าจะได้ยินเสียงว่าได้เลือกสายรถนั้นๆแล้ว LED ที่ประจำอยู่ที่ตำแหน่งของสายนั้นบนป้ายเลือกสายรถจะสว่างขึ้น เพื่อแสดงให้ทราบว่ารถสายนั้นได้ถูกเรียกแล้ว ในขณะเดียวกัน 7-segment module ก็จะแสดงผลสายรถเมล์ตามหมายเลขที่ผู้ใช้เลือก ทั้งนี้เพื่อป้องกันการกดเลือกสายผิด และเมื่อรถเมล์หมายเลขที่เรียกมาถึง ก็จะมีสัญญาณส่งมาที่ไมโครคอนโทรลเลอร์เพื่อให้ยกเลิกการแสดงผลของ LED และ 7-segment module ของสายรถเมล์ที่มาถึง พร้อมกับมีเสียงแจ้งให้ผู้ใช้ทราบว่ารถเมล์สายนั้นมาถึงแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3.1 บล็อกไดอะแกรมของระบบป้ายเลือกสายรถประจำทาง




รูปที่ 3.2 บล็อกไดอะแกรมของระบบแสดงเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การออกแบบระบบแสดงเวลา

ในการออกแบบจะประกอบไปด้วยส่วนสำคัญสองส่วนคือ การติดต่อระหว่างไมโครคอนโทรลเลอร์กับไอซีสร้างฐานเวลาจริงเบอร์ DS1307 และการติดต่อระหว่างไมโครคอนโทรลเลอร์กับไอซีบันทึก/เล่นเสียงตระกูล ISD4003 โดยจะใช้ 7-segment และลำโพงเป็นเอาท์พุทของระบบโดย 7-segment จะทำการแสดงค่าเวลาชั่วโมงและนาทีในขณะนั้นๆตลอดเวลา และเมื่อมีการกดปุ่มบอกเวลาก็จะมีเสียงจากลำโพงบอกเวลาในขณะนั้นให้แก่ผู้พิการทางสายตา บล็อกไดอะแกรมของระบบแสดงเวลาแสดงได้ดังรูปที่ 3.2

3.1 การติดต่อกับไอซีสร้างฐานเวลาจริง

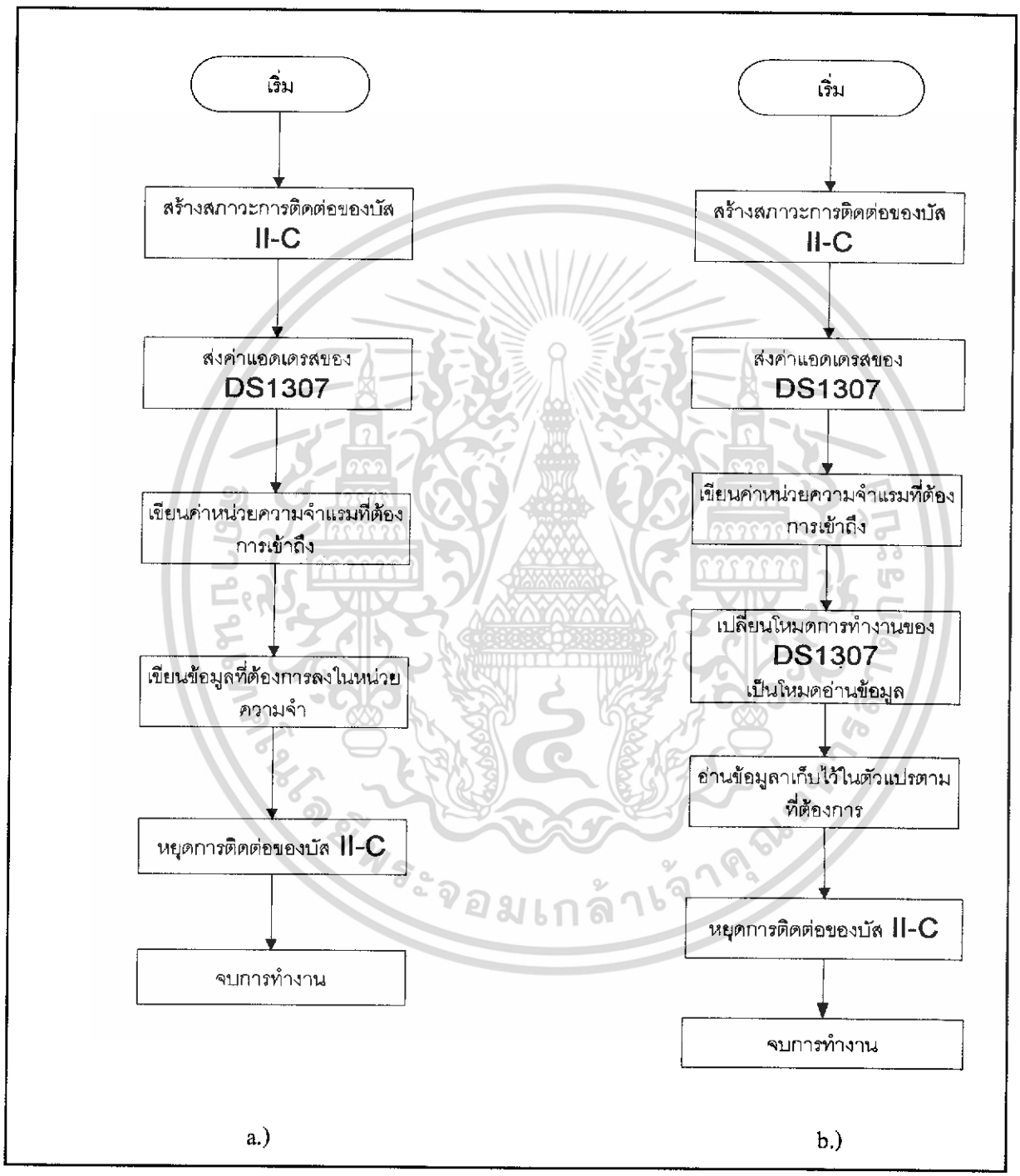
โดยปรกติแล้วไมโครคอนโทรลเลอร์ตะกูล MCS-51 สามารถสร้างฐานเวลานาฬิกาจริงได้จากตัวไทเมอร์ (Timer) ที่อยู่ภายในตัวของมันเองได้ แต่จะไม่สามารถรักษาค่าเวลาล่าสุดได้หากเกิดการรีเซ็ท ดังนั้นจึงได้นำไอซีพิเศษที่ใช้สร้างฐานเวลาจริงมาใช้ เนื่องจากเป็นไอซีที่สามารถรักษาเวลาและข้อมูลไว้ได้แม้จะไม่มีไฟเลี้ยง โดยเบอร์ของไอซีที่นำมาใช้คือ DS1307 และการที่สามารถรักษาข้อมูลไว้ได้แม้จะไม่มีไฟเลี้ยงเนื่องมาจากมีแบตตารี่ต่อร่วมอยู่ที่ขา $V_{BAT}$ (ขา 3 ใช้ต่อกับแบตเตอรี่ 3 V) ของ DS1307 แต่ต้องระวังในเรื่องของไฟเลี้ยงเพราะหากไฟเลี้ยงมีค่าต่ำกว่า 1.25x $V_{BAT}$ หรือประมาณ 1.25 V จะทำให้ไม่สามารถติดต่อกับ DS1307 ได้ ถ้าหากไฟเลี้ยงมีค่าต่ำกว่า $V_{BAT}$ ไอซี DS1307 จะเข้าสู่โหมดสำรองข้อมูลกระแสต่ำทันที แต่วงจรสร้างฐานเวลายังคงทำงานเพื่อให้ค่าของเวลาเดินไปอย่างไม่ผิดพลาด การเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับ DS1307 เป็นแบบระบบบัส $I^2C$ โดย DS1307 จะทำงานเป็นอุปกรณ์เสลฟ เสมอโดยเราต้องทำการเขียนข้อมูลของเวล เที่เ เจะตั้งไป ให้กับ DS1307 และการแสดงค่าของเวลาเราต้องทำการเขียนโปรแกรมเพื่ออ่านค่าของเวลาออกมา โฟลชาร์ตการเขียนข้อมูลและการอ่านค่าข้อมูลจาก DS1307 แสดงดังรูปที่ 3.3 a.) และ 3.3 b.) ตามลำดับ

### 3.4 การติดต่อกับไอซีบันทึกและเล่นเสียง ISD4003

ไอซีบันทึกและเล่นเสียง ISD4003 เป็นไอซีที่สามารถแปลงสัญญาณเสียงที่เข้ามาจากสัญญาณอนาล็อกเป็นข้อมูลในรูปแบบดิจิตอลได้โดยใช้วงจรแซมปลิ้งแอนด์โฮลด์ (Samplig&Hold circuit) ภายในตัวมันเองแล้วเก็บข้อมูลไว้ในหน่อยความจำถาวร (Nonvolatile memory) ภายในตัวมันเองเช่นกัน จึงช่วยลดความยุ่งยากของการออกแบบไปได้มากโดยผู้ใช้งานเพียงแค่เขียนโปรแกรมจากไมโครคอนโทรลเลอร์ไปควบคุมการทำงานของไอซี เท่านั้น โดยการทำงานของไอซีบันทึก/เล่นเสียงเบอร์ ISD4003 จะแบ่งออกเป็นทั้งหมดห้าสภาวะ ได้แก่ เริ่มการทำงาน (Power up), การบันทึกเสียง (Record) ,สภาวะการเล่นเสียงที่บันทึกไว้ (Playback),สภาวะหยุด(Stop) และสุดท้ายคือ สภาวะที่หยุดการทำงานและอยู่นิ่ง(Stop and Standby หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Stop and Power down) โดยสภาวะทั้งหมดนี้จะถูกควบคุมด้วยบิตควบคุมทั้งหมด 5 บิต (รายละเอียดของแต่ละบิตดูได้ใน datasheet)




รูปที่ 3.3 โฟลว์ชาร์ตการติดต่อรหว่างไมโครคอนโทรลเลอร์กับไอซี DS1307

a.) การเขียนข้อมูล และ b.) การอ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงงานนี้จะใช้การติดต่อระหว่างไมโครคอนโทรลเลอร์และ ISD4003 แบบ 16 บิต ผ่านพอร์ต SPI โดย ISD4003 จะเป็นสเลฟของไมโครคอนโทรลเลอร์ ข้อมูล 16 บิตที่ใช้รับส่งจะแบ่งเป็นแอดเดรสของตำแน่งหน่วยความจำที่ต้องการบันทึก 11 บิต และบิตควบคุมการทำงานของ ISD 4003 อีก 5 บิต การเริ่มต้นการติดต่อจะเริ่มหลังจากที่ ขา *SS* ของ ISD4003 มีสถานะเป็น LOW และจะสิ้นสุดการทำงานเมื่อ ขา *SS* ของ ISD4003 มีสถานะเป็น HIGH สิ่งที่ต้องระวังในการใช้งาน ISD4003 เพื่อไม่ให้เกิดข้อผิดพลาดของการบันทึกและเล่นเสียงก็คือ การเกิดสัญญาณ อินเตอร์รัป ของ ISD 4003 โดยในการบันทึก/เล่นเสียงหากเราไม่ได้กำหนดค่าแอเดรสที่ต้องการ การบันทึก/เล่นเสียงจะทำต่อจากแอดเดรสล่าสุดของหน่วยความจำ ถ้าไม่มีการส่งคำสั่งหยุดในการบันทึก ISD4003 จะส่งสัญญาณอินเตอร์รัปออกมาเมื่อการบันทึกเต็มหน่วยความจำของไอซี และในการเล่นเสียงก็จะมีสัญญาณอินเตอร์รัปส่งออกมาทุกครั้งเมื่อจบข้อควาที่บันทึกในแอดเดรสนั้นหรือเมื่อเล่นครบทุกข้อมูลในหน่วยความจำ การรับสัญญาณอินเตอร์รัปจาก ไอซีบันทึก/เล่นเสียงเบอร์ ISD4003 นั้น ไมโครคอนโทรลเลอร์จะทำการรับเข้าที่ขา INT โดยสัญญาณอินเตอร์รัปที่ออกจาก ISD4003 นั้นต้องนำไปเข้าวงจรเปลี่ยนค่าระดับแรงดันจาก 3V เป็น 5V (ในโครงงานนี้ใช้วงจร Emitter follower โดยใช้ทรานซิสเตอร์แบบ PNP ดูรูปประกอบได้จาภาคผนวก) ก่อนไม่ควรจะต่อกับขา INT ของไมโครคอนโทรลเลอร์โดยตรงเพราะอาจทำการเกิดความเสียหายต่อ ISD4003 ได้

เราสามารถบันทึกเสียงที่ต้องการลงไปได้ผ่านขาออดิโออิน (Audio in) ของไอซี โดยสัญญาณเสียงที่เราต้องการบันทึกสามารถมาได้จากทั้งสัญญาณ ไมโครโฟนหรือแหล่งกำเนิดสัญญาณเสียงที่เราต้องการ เช่นเครื่องเล่นซีดี หรือจากคอมพิวเตอร์ และในการเล่นเสียงเราจะนำสัญญาณจากขา ออดโอเอาท์ (Audio out) มาเข้ากับแอมป์พลิไฟเออร์ก่อน เนื่องจาก ความต้านทานขาออก (output impedance) ของ ISD4003 มีขนาด 5 กิโลโอห์ม ซึ่งค่าความต้านทานที่มากนี้จะทำให้ไม่สามารถขับลำโพงได้โดยตรง วงจรที่ใช้ในการติดต่อระหว่างไมโครคอนโทรลเลอร์กับ ISD4003 ในโครรงานนี้สามารถดูได้จากในภาคผนวก

### 3.5 โปรแกรมการทำงานของไมโครคอนโทรลเลอร์

ในโครงงานนี้เราเลือกใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89S8253 ซึ่งเป็นไมโครคอนโทรลเลอร์ 8 บิต ที่มีขาอินพุต/เอาท์พุตพิเศษรองรับการรับ-ส่งข้อมูลแบบ SPI ซึ่งเราจะใช้ในการติดต่อกับไอซีบันทึก/เล่นเสียงตระกูล ISD4003 ส่วนการติดต่อกับไอซีสร้างฐานเวลาจริงจะใช้ขาอินพุต/เอาท์พุตทั่วไปของไมโครคอนโทรลเลอร์ในการสร้างสภาวะการติดต่อแบบบัส $I^2C$ และเนื่องจาการติดต่อระหว่างไมโครคอนโทรลเลอร์กับ ISD4003 เป็นแบบ 16 บิต แต่ AT89S8253 ที่เลือกใช้เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ดังนั้นการรับส่งข้อมูลของไมโครคอนโทรลเลอร์จึงต้องทำการแบ่งข้อมูล 16 บิตออกเป็นเป็น 8 บิตสองชุด และต้องกำหนดให้ส่งค่า LSB ของข้อมูลก่อนเนื่องจากเป็นมาตรฐานโปรโตรคอลของ ISD 4003 สำหรับขั้นตอน การเขียนโปรแกรมเพื่อการบันทึกเสียงและเล่นเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถดูได้จาก datasheet และการบันทึกและเล่นเสียงแบบหลายข้อความสามารถดูรายละเอียดเพิ่มได้จาก Application note ของ ISD 4003 ที่แนบอยู่ในส่วนของภาคผนวก

สำหรับการรับส่งข้อมูลของระหว่างไมโครคอนโทรเลอร์ตัวควบคุมการเลือกสายรถและที่7-segment module จะใช้การรับส่งแบบ UART โดยไมโครคอนโทรลเลอร์ที่ตัว 7-Segment module จะทำการรับข้อมูลที่ขอ Rx แบบ Serial Interrupt ซึ่งวิธีการแบบนี้ทำให้ไมโครคอนโทรลเลอร์ไม่เสียต้องเสียเวลาในการรอรับข้อมูลที่จะเข้ามา ทำให้ไมโครคอนโทรลเลอร์ สามารถทำงานอย่างได้

ลำดับการทำงานของไมโครคอนโทรลเลอร์ในการออกแบบระบบแสดงเวลาและที่ 7-Segment module แสดงรูปที่ 3.4 a.) และ 3.4 b.) ตามลำดับ การทำงานของไมโครคอนโทรลเลอร์ตัวควบคุมการเลือกสายรถจะแสดงดังรูปที่ 3.5

## 3.6 การแสดงผล

ในการแสดงผลจะแบ่งออกเป็นส่วนของหน้าจอแสดงผลที่ใช้ 7-Segment และลำโพงที่ใช้แสดงเสียงพูด

### 3.6.1 การแสดงผลของตัวเลขเจ็ดส่วน(7-segment)

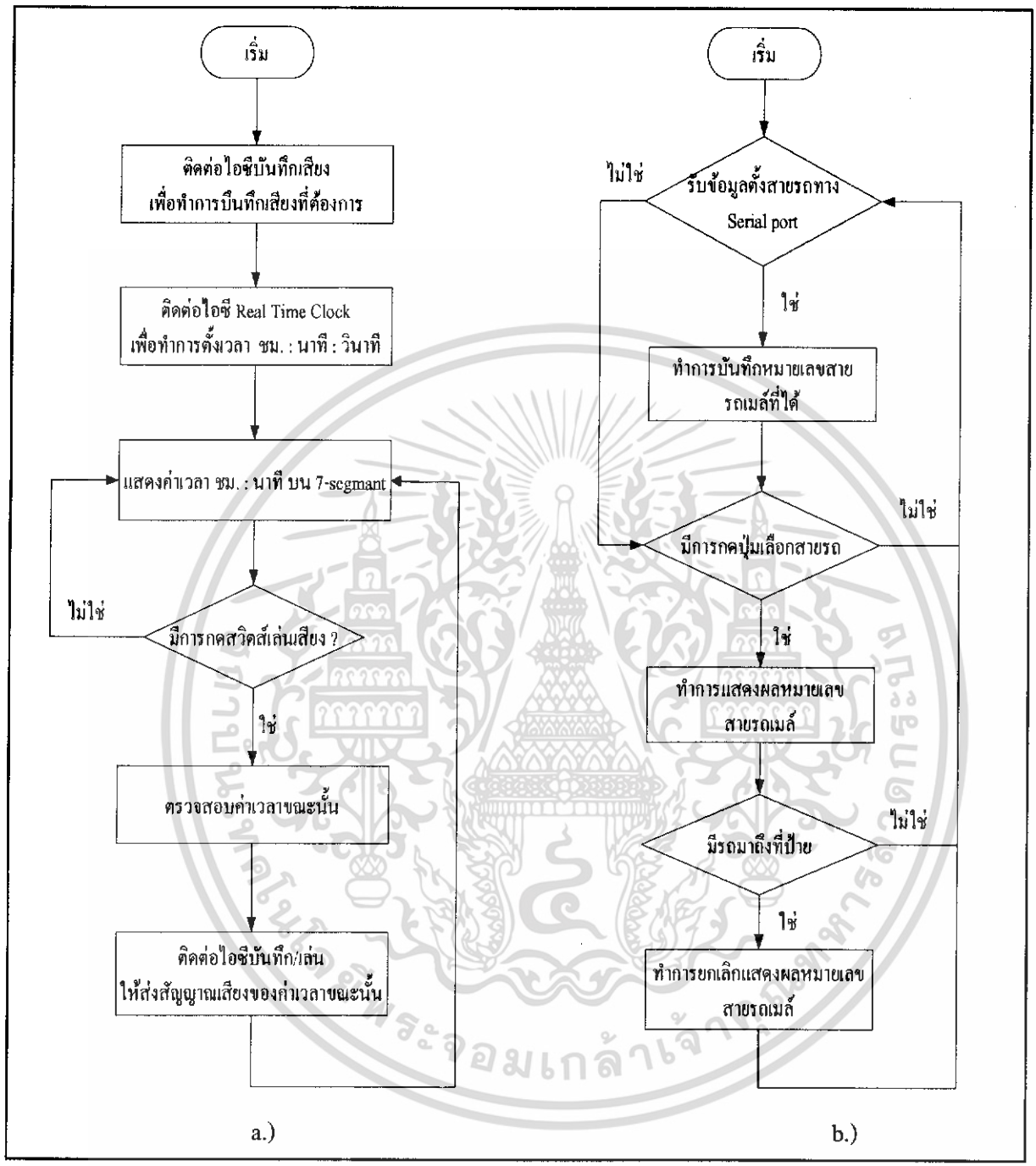
การแสดงผลของ 7-segment นั้นจใช้ไอซีขับกระแส (Driver) ที่เรียกว่า BCD (ย่อมากจาก Binary to Code Decimal)เบอร์ CD4511B ซึ่งเป็นไอซีที่ช่วยขับกระแสมี 4 อินพุทและให้จะสัญญาณเอาท์พุทออกมา 7 สัญญาณ ซึ่งจะเหมือนการทำงานของไอซีถอดสัญญาณ(Decoder) เมื่อนำไปต่อร่วมกับ 7-segment และทำการกำนดค่าอินพุทต่างๆจะทำให้ได้ตัวเลขออกมาแสดงที่ 7-segment ได้ตั้งแต่เลข 0-9 แต่การที่จะให้ 7-segment ทำงานพร้อมกันหลายๆตัวตลอดเวลาจะเกิดการสูญเสียกำลังงานโดยไม่จำเป็นจึงนำวิธีการที่เรียกว่าการมัลติเพล็กซ์ (Multiplex) มาใช้ร่วมด้วย คือ การทำให้ 7-segment สลับกันติดสว่างทีละหลักให้เร็วกว่าสายตามนุษย์จะจับได้ (ประมาณ 25 ภาพ/วินาที ในการออกแบบนี้ใช้ 30 ภาพ/นาที) จะทำให้เห็นเสมือน 7-segment สว่างพร้อมกันทุกตัว เช่นถ้าต้องการมัลติเพล็กซ์ให้ 7-segment 5 หลักเสมือนสว่างพร้อมกันทุกตัว เราสามารถคำนวณค่าเวลาที่ต้องให้แต่ละตัวติดสว่างได้ดังนี้

จำนวนครั้งที่ต้องมัลติเพล็กซ์ทั้งหมด 30x5 =150 ภาพ/วินาที

ดังนั้นการสว่างหนึ่งครั้งของแต่ละหลักต้องมีช่วงเวลาประมาณ 1/150=0.006 วินาที /ภาพ

ค่ากระแสที่ใช้ขับ 7-Segment ในการมัลติเพล็กซ์ควรจะมากกว่าการใช้งานแบบติดสว่างปรกติ เนื่องจากการไหลของกระแสจะไม่ได้เกิดขึ้นตลอด ค่าของกระแสที่ใช้จะขึ้นอยู่กับขนาดซึ่งต้องดูไม่ให้เกิน peak, average และ %pulse width

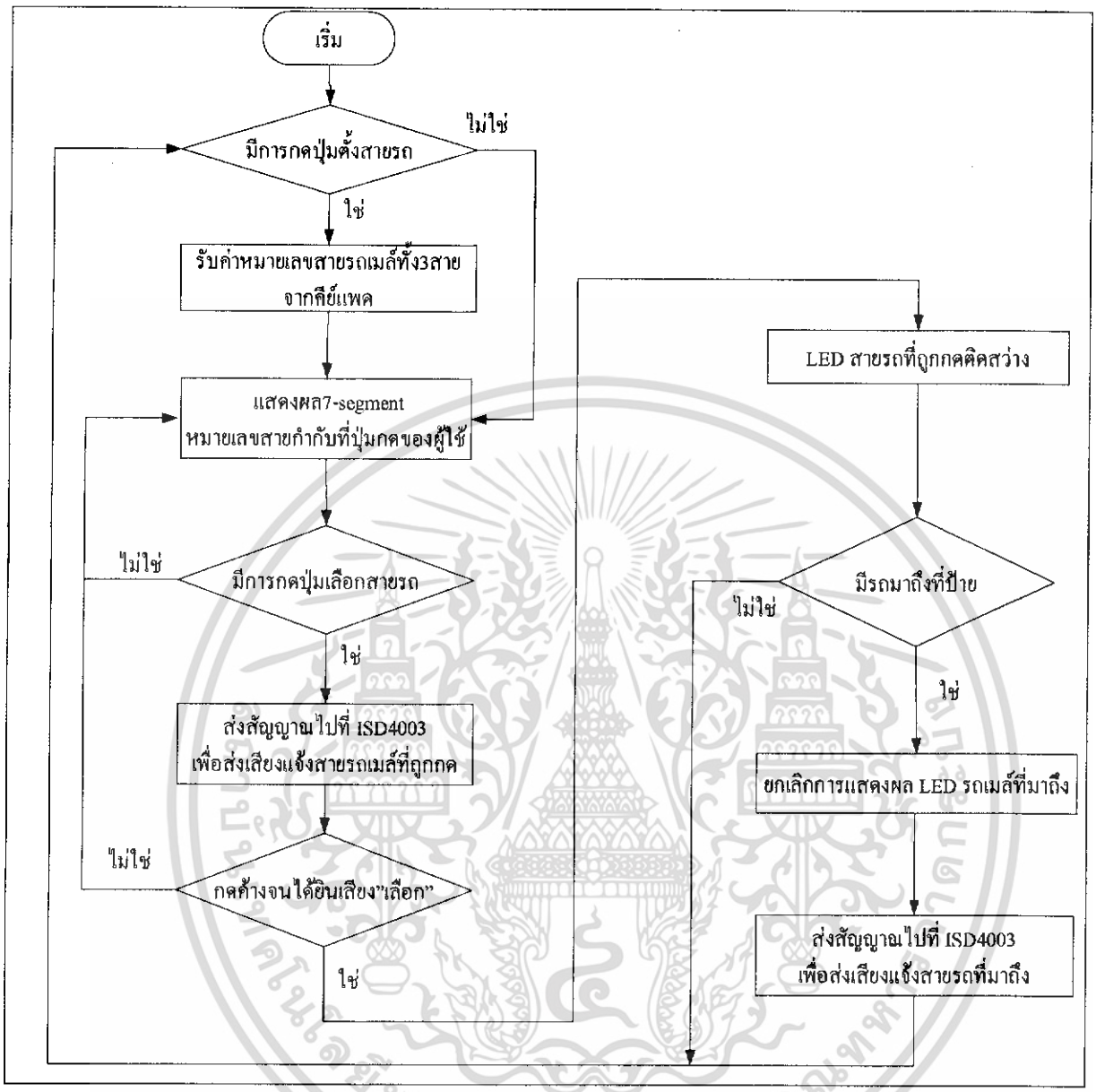
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3.4 ลำดับการทำงานของไมโครคอนโทรลเลอร์

a.) การแสดงเวลา และ b.) 7-Segment module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3.5 ลำดับการทำงานของไมโครคอนโทรลเลอร์ตัวควบคุมการเลือกสายรถ

ในกรณีที่เราใช้ 7-Segment ที่ต้องใช้แรงดันค่าสูงกว่า 5V เช่น 12V ในกรณีนี้ CD4511 จะต้องใช้ไฟเลี้ยงประมาณ 12V เช่นกัน และค่าแรงดันอินพุทขั้นต่ำที่จะทำให้ได้แรงดันที่เอาท์พุท 12V ก็จะต้องมีค่าประมาณ 10 V (ทั้งนี้ขึ้นอยู่กับค่า Noise margin ของไอซีแต่ละตัวด้วย) MCS-51 นั้นจะให้แรงดันเอาท์พุทสูงสุดที่ 5V จึงใช้เป็นสัญญาณอินพุทให้กับ CD4511 โดยตรงไม่ได้ จึงต้องใช้ ไอซี ULN2003 มาเป็นตัวเชื่อมต่อ ซึ่งข้างใน ULN 2003 นี้เป็นทรานซิสเตอร์อะเรย์ที่มีเอาท์พุทเป็นโอเพ่นคอลเลคเตอร์(open collector) จึงช่วยยกค่าระดับแรงดันที่เข้ามา 5V ให้เป็น 12V ได้ (ดูรูปได้วงจรที่ใช้ในภาคผนวก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2 การขยายสัญญาณ (Amplifying) ของสัญญาณเสียง

เนื่องจากความต้านทานขาออก (output impedance) ของ ISD4003 มีขนาด 5 กิโลโอห์มทำให้ไม่สามารถนำมาขับลำโพงได้โดยตรง การขยายสัญญาณจึงเป็นสิ่งจำเป็น ในโครงงานนนี้ใช้ไอซีเบอร์ LM386 ซึ่งเป็นไอซีขยายสัญญาณที่มีราคาไม่แพง สามารถปรับอัตราขยายสัญญาณ ($A_v$) ได้ตั้งแต่ 20-200 เท่าขึ้นอยู่กับผู้ใช้ และให้เอาท์พทสูงสุด 0.25 W ซึ่งเพียงพอต่อการขับลำโพงขนาดไม่ใหญ่มากและให้ความดังเสียงที่เพียงพอต่อการใช้งานในโครงงาน โดย LM386 ใช้หลักการของการขยายสัญญาณแบบไม่กลับเฟส (Non Inverting Amplifier) แต่สิ่งที่ระต้องระวังคือค่าแรงดันออฟเซ็ท (Offset Voltage) ที่อาจเกิดขึ้นได้หากใช้อัตราขยายสูงๆ ซึ่งอาจแก้ได้ด้วยการจัดไบอัสที่เหมาะสมให้กับวงจร และควรใส่ตัวเก็บประจุบายพาสในจุดสำคัญๆเช่น ขาสัญญาณเข้าและไฟเลี้ยงเป็นต้น เพื่อลดผลจากสัญญาณรบกวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4

# การทดลองและผลการทดลอง

## 4.1 การทำงานของไมโครคอนโทรเลอร์

จากการทดลองการทำงานของไมโครคอนโทรเลอร์เมื่อมีการทำงานหลายอย่างในเวลาเดียวกัน พบว่า

1) ไมโครคอนโทรเลอร์ยังคงสามารถทำงานได้ทันโดยไม่เกิดการค้าง(Hang)
2) เมื่อมีการใช้งานวงจรของไมโครคอนโทรเลอร์ร่วมกับวงจรอื่นๆที่มีค่าไฟเลี้ยงวงจรต่างกันไป โดยการใช้กราวด์ของวงจรร่วมกันพบว่าบางครั้งไมโครคอนโทรลเลอร์ ไม่สามารถทำงานได้

## 4.2 การติดต่อกับไอซีสร้างฐานเวลาจริง DS1307 ผ่านระบบบัส $I^2C$

จากการทดลองการใช้งานพอร์ทพอร์ตอินพุตเอาท์พุตทั่วไปของ MCS-51 มาสร้างสร้างระบบบัส $I^2C$ เพื่อใช้ในการติดต่อกับไอซีสร้างฐานเวลาจริง DS1307 และการนำไฟเลี้ยงออกจากวงจรพบว่า

1) สามารถติดต่อกับ DS1307 ได้ทั้งการตั้งค่าเวลา และการดึงค่าเวลาจาก DS1307 มาแสดงผลได้อย่างถูกต้องได้โดยไม่จำเป็นต้องใช้พอร์ท $I^2C$ พิเศษที่มีอยู่ในไมโครคอนโทรลเลอร์บางตระกูล
2) DS1307 ยังคงสามารถรักษาค่าเวลาให้ถูกต้องไว้ได้ แม้จะมีการนำไฟเลี้ยงออกจากวงจร

## 4.3 การติดต่อกับไอซีบันทึกและเล่นเสียง ISD4003 โดยใช้การเชื่อมต่อแบบ SPI

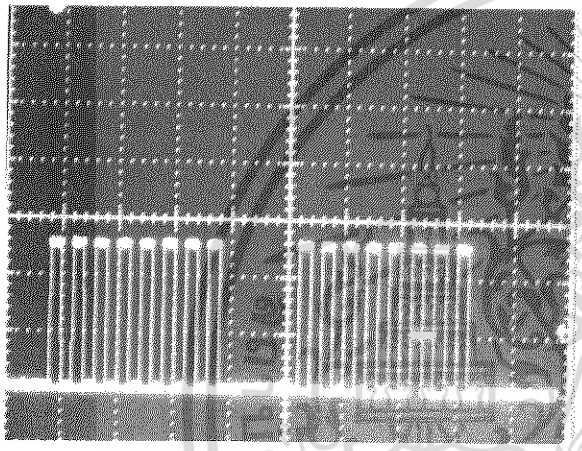
จากการทดลองโดยการเขียนโปรแกรมจากไมโครคอนเลอร์ควบคุมการทำงานของ ISD4003 ในการบันทึกเสียง (ใช้ไมค์ในการบันทึกเสียง) และเล่นเสียงที่บันทึกทั้งแบบไม่กำหนดแอดเดรสและกำหนดพบว่า

1) ในแบบที่ไม่กำหนดแอดเดรสในการบันทึกเสียงที่อัดจะอัดต่อกันไปเรื่อยๆ โดยดูได้จากการเล่นเสียง จะเล่นต่อกันไปในแต่ละข้อความไม่สามารถเลือกเล่นเป็นข้อความที่บันทึกก่อนหรือหลังได้
2) ในแบบที่กำหนดแอดเดรสในการบันทึก เราสามารถเลือกเล่นเสียงตามที่เราต้องการได้โดยการกำหนดให้เล่น ณ ตำแหน่งเดียวกันกับแอดเดรสของเสียงที่บันทึกนั้นๆ เราจึงสามารถอัดแยกเป็นข้อความและนำมาเล่นต่อกันได้

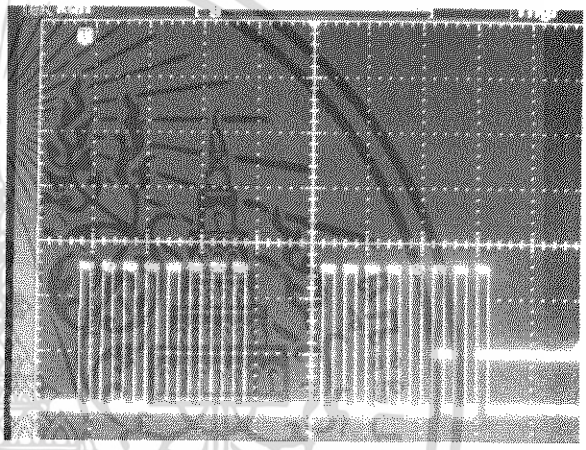
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ในการเล่นเสียงต่อกันจากหลายข้อความจะมีช่วงห่างของเสียงเล็กน้อยเนื่องจาก ISD4003 จะมีค่าหน่วงเวลาของแต่ละคำสั่งการทำงาน และความเร็วของคำพูดจะขึ้นอยู่กับการอัดจากผู้ออกแบบ

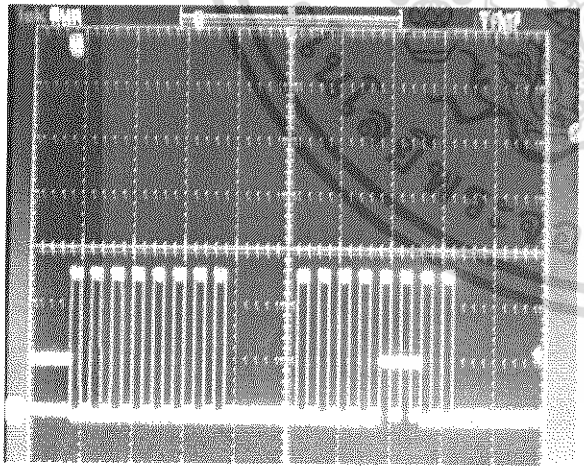
ตัวอย่างรูปของสัญญาณคำสั่งควบคุมจากไมโครคอนโทรลเลอร์ที่ได้จากการใช้ออสซิลโลสโคปจับแสดงดังรูปที่ 4.1 จะสังเกตุเห็นว่า จะมีสัญญาณพัลส์ 8บิต สองขบวนรวมเป็นทั้งหมด 16 บิตซึ่งวัดจากขาสัญญาณ SCK และสัญญาณที่อยู่ระหว่างขบวนพัลส์ที่สองคือสัญญาณควบคุมจากไมโครคอนโทรเลอร์วัดจากขา MOSI โดยมีแอดเดรสบิตเป็น 00000000 ทั้งหมด ข้อมูลที่ส่งจะเรียนจากบิต LSB ไป MSB



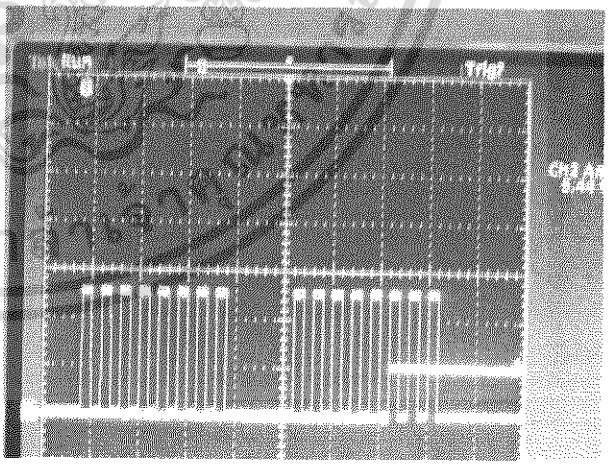
a)



b)



c)



d)

รูปที่ 4.1 แสดงสัญญาณควบคุม ISD4003 a) ส่งค่า00100000 เพื่อPower up, b) ส่งค่า 10100000 เพื่อ Record, c) ส่งค่า 00110000 เพื่อ Stop, และd) ส่งค่า 11100000 เพื่อ Play

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

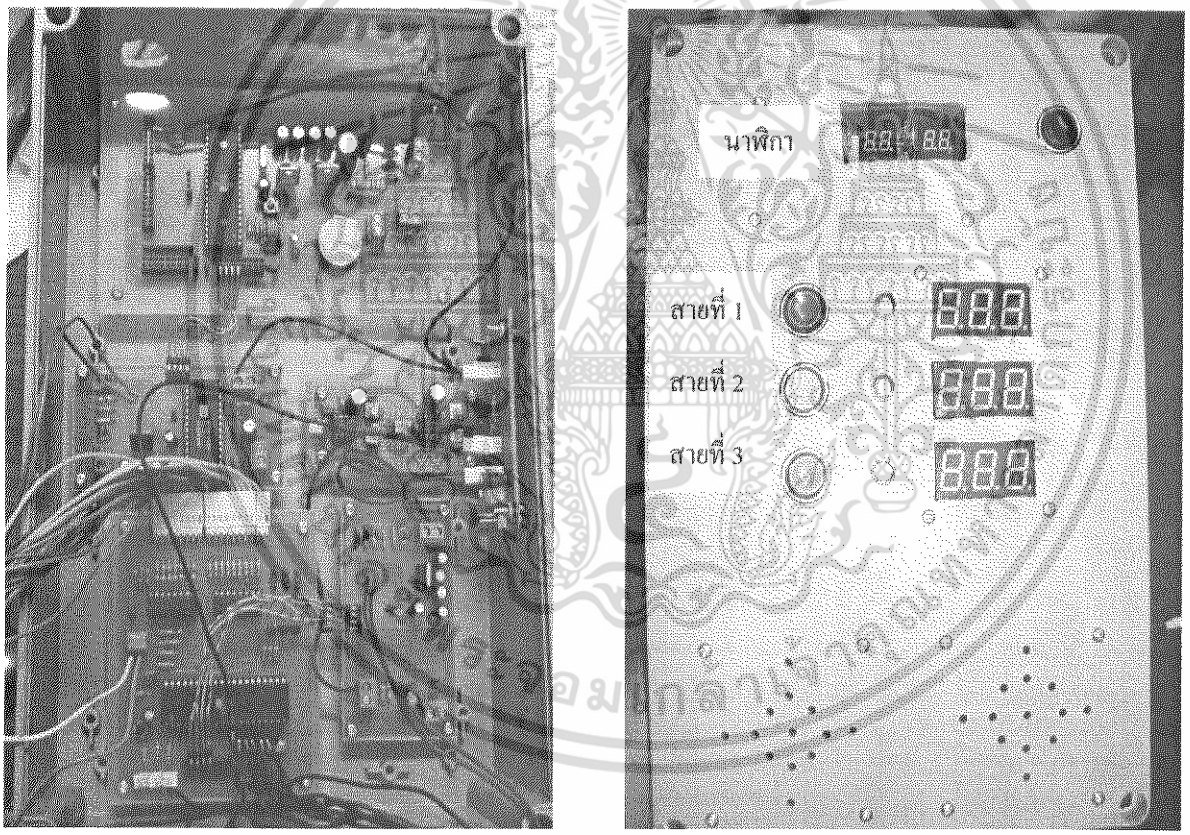
**4.4 การแสดงผลของตัวเลขเซ็ดส่วน(7-segment) และเสียง**

จากการทดลองการสั่งงาน 7-Segment แบบมัลติเพล็กซ์โดยไม่ใช้ไอซีแลตซ์ พบว่า

1) 7-Segment แสดงค่าได้ถูกต้องตามที่กำหนด
2) เมื่อค่าแรงดันสูงพอที่ทำให้ 7-Segment ติดสว่างแล้ว ค่าความเข้มแสงจะขึ้นอยู่กับกระแสที่ไหลผ่าน 7-Segment

จากการทดลองเล่นเสียงโดยผ่านการขยายสัญญาณของ LM386 ออกทางลำโพงพบว่า

1) เสียงที่ได้มีความดังในระดับที่ผู้ใช้งานสามารถได้ยินอย่างชัดเจน
2) คุณภาพของเสียงอยู่ในระดับปานกลางซึ่งสามารถฟังได้รู้เรื่องและชัดเจนแต่ยังคงมีสัญญาณรบกวน โดยเฉพาะเมื่อสายสัญญาณของลำโพงอยู่ใกล้กับสายสัญญาณของ 7-Segment



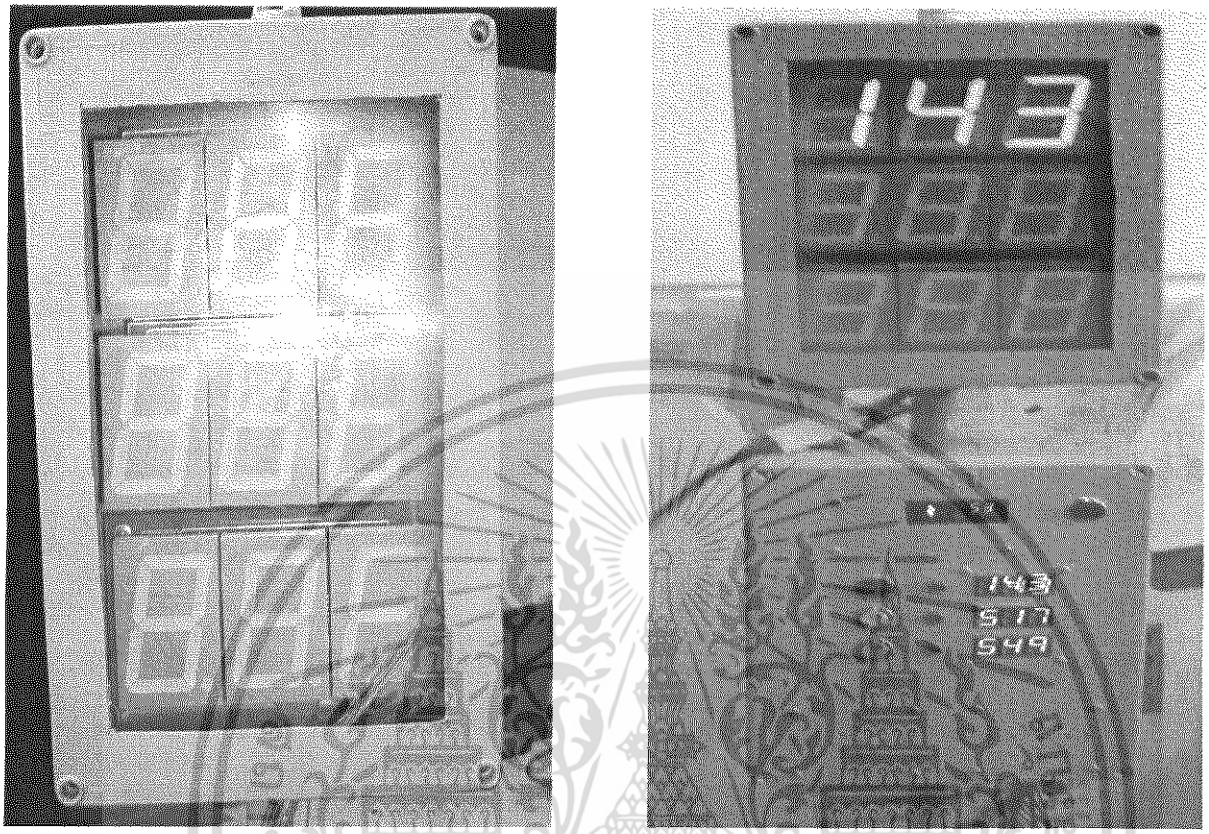

a.) b.)

รูปที่ 4.2 แสดงรูปของกล่องควบคุม

a.)วงจรภายในกล่องควบคุม

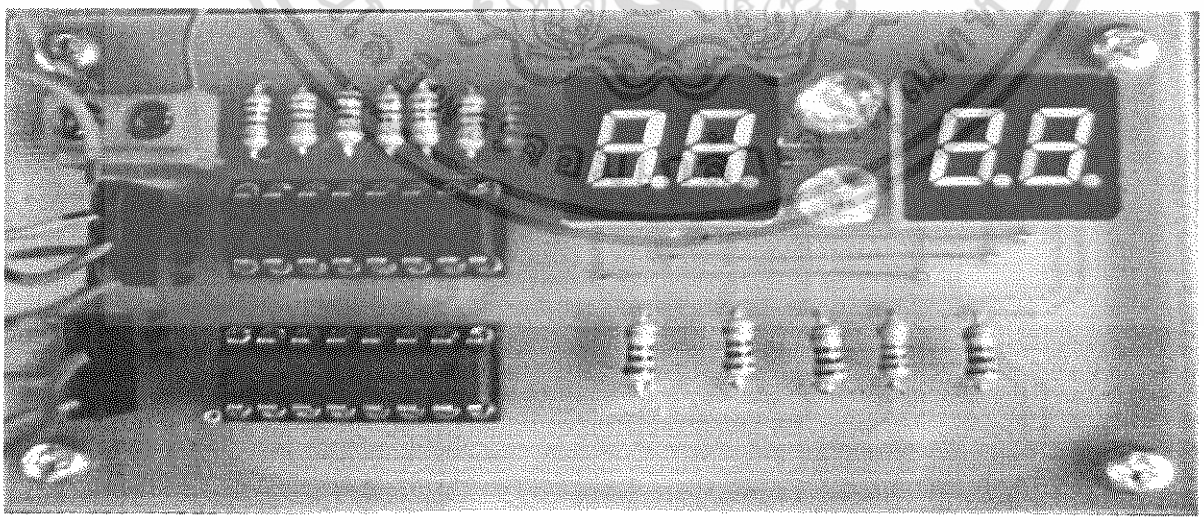
b.) ส่วนติดต่อกับผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



a.) b.)

รูปที่ 4.3 a.) 7-Segment module แสดงสายรถที่ถูกเรียก
b.) การใช้งานร่วมกันของกล่องควบคุมและ 7-Segment module



รูปที่ 4.4 นาฬิกาแสดงเวลาที่ใช้ในโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# บทสรุป

ปริญญานิพนธ์ฉบับนี้ได้กล่าวถึง ความเป็นมาของโครงงาน ป้ายรถประจำทางเพื่อผู้พิการทางสายตา การสร้างและทดสอบทางเทคนิค รวมถึงทฤษฎีที่เกี่ยวข้อง เพื่อเป็นแนวทาง ให้ผู้สนใจนำไปพัฒนาต่อให้สมบูรณ์ ยิ่งขึ้น ลักษณะโดยรวมของโครงงาน คือระบบที่อำนวยความสดวกในการเดินทางโดยรถโดยสารประจำาทาง โดยใช้เสียงในการสื่อสารระหว่างป้ายกับผู้พิการ เช่นบอกเวลา ข้อมูลรถประจำทาง หมายเลขรถที่เข้าจอดเป็นต้น

## 5.1 สรุปผลการทดลอง

จากผลการทดลองเราสรุปได้ว่าเราสามารถใช้งานไมโครคอนโทรเลอร์ในการทำงานพร้อมๆกันได้หลายอย่าง ทั้งการรับส่งข้อมูลในรูปแบบต่างๆและการใช้งานเพื่อเป็นอินพุตเอาท์พุตต่างๆได้เป็นอย่างดี โดยคุณภาพของการแสดงผลในส่วนต่างๆจะขึ้นอยู่กับลักษณะของวงจรส่วนนั้นๆเป็นหลัก ไมโครคอนโทรลเลอร์เพียงทำหน้าที่เป็นตัวควบคุมการทำงานให้ถูกต้องตามที่ออกแบบไว้

## 5.2 วิเคราะห์ผลการทดลอง

1) การที่ไมโครคอนโทรลเลอร์ไม่สามารถทำงานได้เมื่อใช้กราวด์ร่วมกับวงจรอื่นอาจเกิดจากสัญญาณรบกวนเนื่องจากใช้ค่าไฟเลี้ยงวงจรที่ต่างกัน
2) การใช้งาน ISD4003 ต้องระวังในเรื่องของจำนวนหน่วยความจำของ ISD4003 และการเกิดอินเตอร์รัปจาก ISD4003 เพราะทั้งสองอย่างนี้อาจทำให้เกิดความผิดพลาดในการบันทึกและเล่นเสียงแบบหลายข้อความได้
3) ระหว่างติดต่อกับ ISD4003 ไมโครคอนโทรเลอร์จะไม่สามารถทำงานอื่นได้เพราะต้องรอการตอบสนองจาก ISD4003 อาจแก้ได้โดยการใช้อินเตอร์รัปเข้ามาช่วย
4) 7-Segment ที่มีความสว่างน้อยและยังมีการสั่น อาจเกิดจากการให้กระแสที่ไหลผ่าน 7-Segment น้อยเกินไป เนื่องจากการทำมัลติเพล็กซ์จะทำให้ช่วงเวลาของกระแสที่ไหลผ่านมีค่าน้อย โดยกระแสจะมีลักษณะเป็นพัลส์
5) คุณภาพเสียงจะขึ้นอยู่กับหลายปัจจัยหลายตั้งแต่ไมโครโฟนไปจนถึงการจัดวงจรขยายสัญญาณ การเดินสายสัญญาณและลำโพง ดังนั้นการปรับปรุงคุณภาพเสียงให้ดีที่สุดนั้นจึงเป็นเรื่องที่ต้องใช้เทคนิคและประสบการณ์ของผู้ทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.3 สรุปผลของโครงงาน

ผลของโครงงานของระบบป้ายรถประจำทางสามารถรับคำสั่งเรียกสายรถประจำทางได้และเล่นเสียงให้ผู้ใช้งานทราบได้ การตรวจจับการมาถึงของรถประจำทางยังใช้เพียงสวิตช์กดบอกไมโครคอนโทรลเลอร์ในการจำลองการมาถึง ส่วนของนาฬิกาเพื่อบอกเวลาด้วยเสียง สามารถทำได้ตรงกับเวลาที่แสดงในขณะนั้น รวมทั้งสามารถตั้งค่าสายรถประทางและนาฬิกาได้ตามที่ผู้ออกแบบต้องการ

ในการพัฒนาต่อนั้นเราอาจจะทำการเปลี่ยนอุปกรณ์ที่ใช้ในการตรวจจับการมาถึงของสายรถประจำทางเป็นอุปกรณ์ประเภทไม่ต้องใช้การสัมผัสโดยตรง(Contact less) เช่น RFID หรือ อินฟราเรด และเราอาจเพิ่มหน่วยความจำถาวรเช่น EEPROM เพื่อใช้ในการเก็บค่าของสายรถประจำทางที่ทำการตั้งเข้าไปใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บรรณานุกรม


1. ธีรบูลย์ หล่อวิเชียรรุ่ง และ นคร ภักดีชาติ, ชัยวัฒน์ ลิ้มพรจิตรวิไล,"ปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 ด้วยโปรแกรมภาษา C", บริษัท อินโนเวทีฟ เอ็กเพอริเมนต์ จำกัด, 367 หน้า, 2546
2. ประจิน พลังสันติกุล,"การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ AVR ด้วยภาษา C กับ WinAVR(C Compiler)" บริษัท แอพซอฟต์เทค จำกัด,365 หน้า, 2549
3. เทอดศักดิ์ ลิ่วหาทอง,"สไลด์ประกอบการสอนวิชา Microprocessor and Application", ภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สจล.
4. Roger L. Tokheim, "Digital Electronic-Second Edition",Mcgraw-Hill,230p. , 1985
5. Phillips Semiconductor, "THE I $^{2}$C-BUS SPECIFICATION VERSION 2.1", document order number: 9398 393 40011, 46p. , 2000
6. A Winbond company, " The ISD4000 SPI Control Port Operations, a Simplified Guide", Application Note 5a,5p.,2003



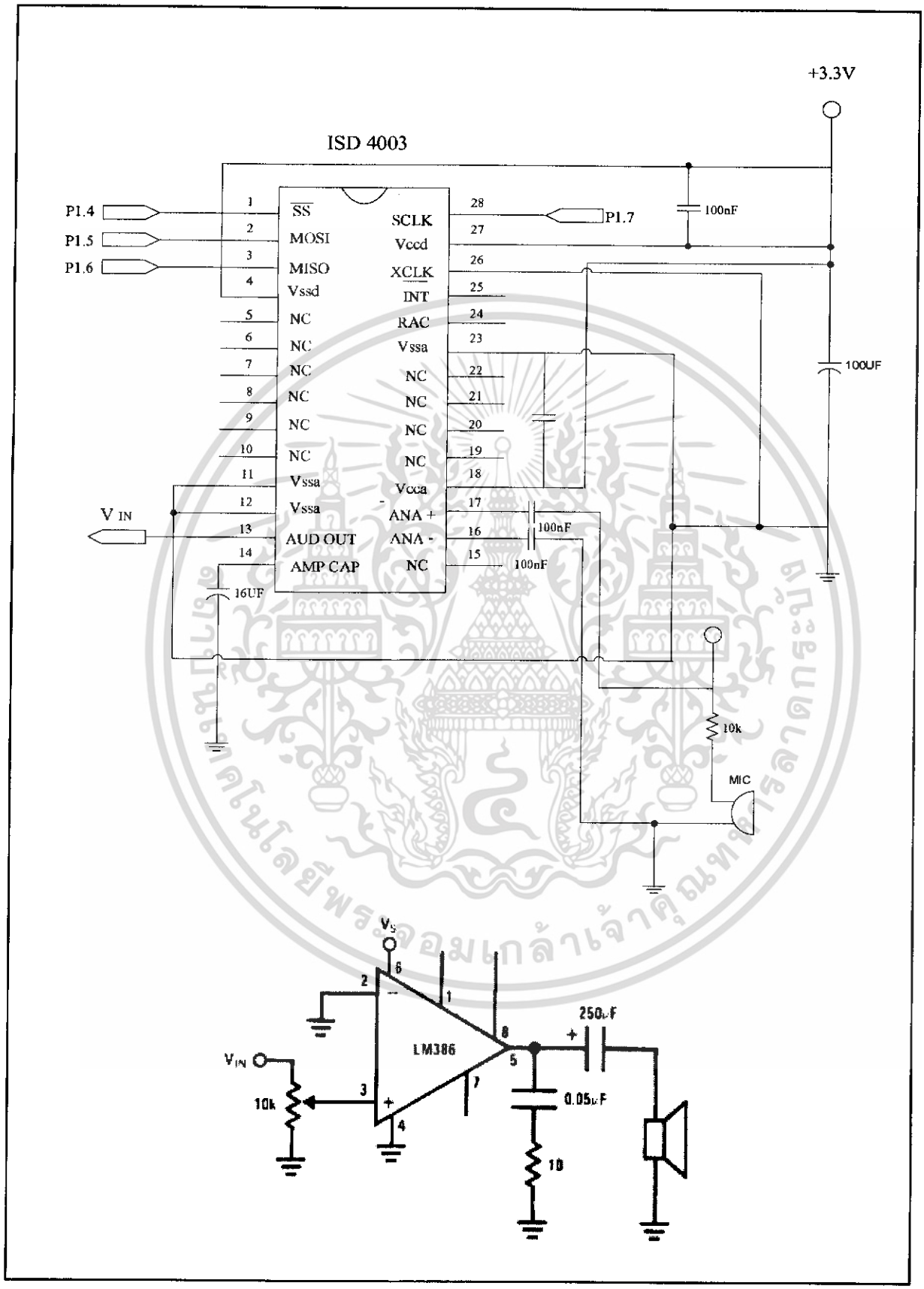
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

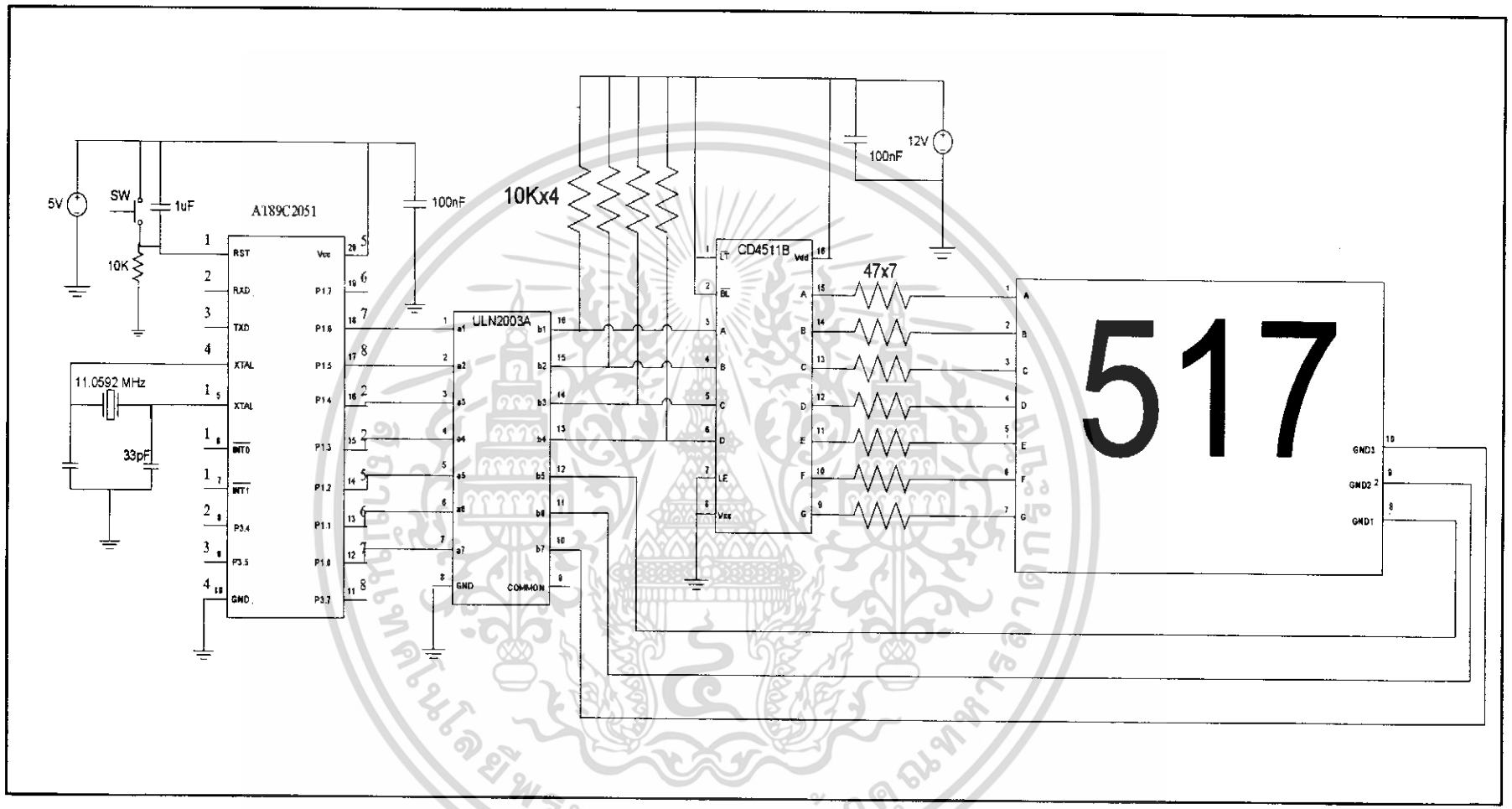
# วงจรที่ใช้ในโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

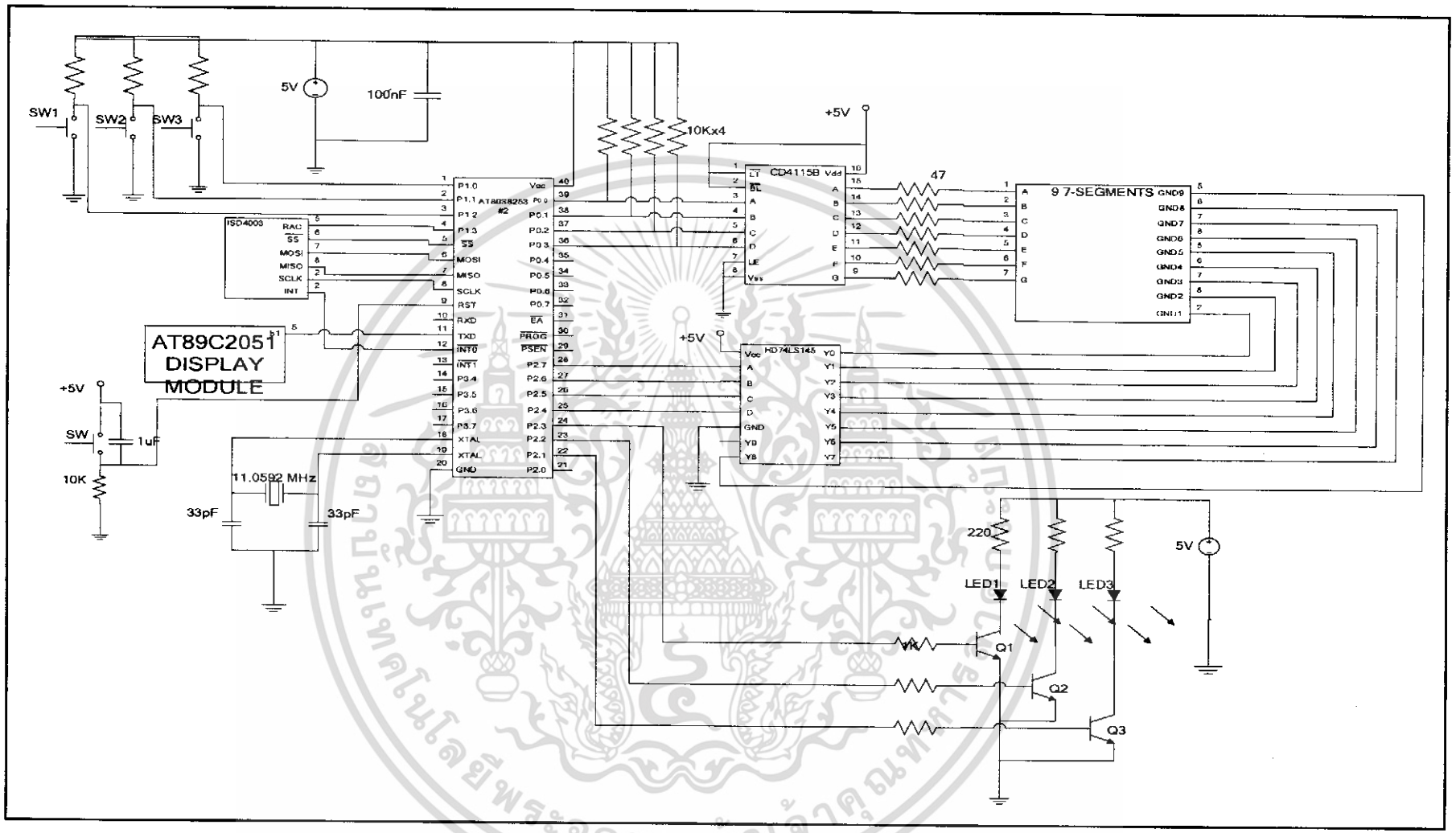



วงจรที่ใช้ในการบันทึกและเล่นเสียงของ ISD4003

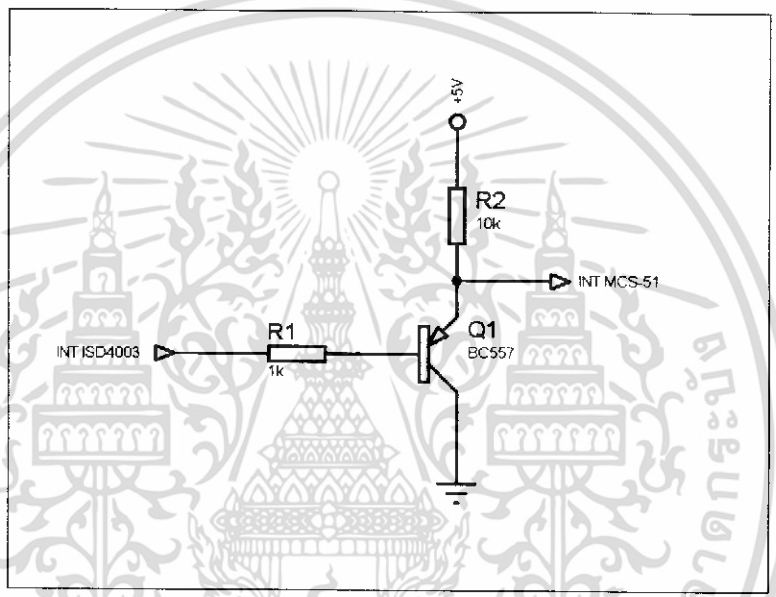
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



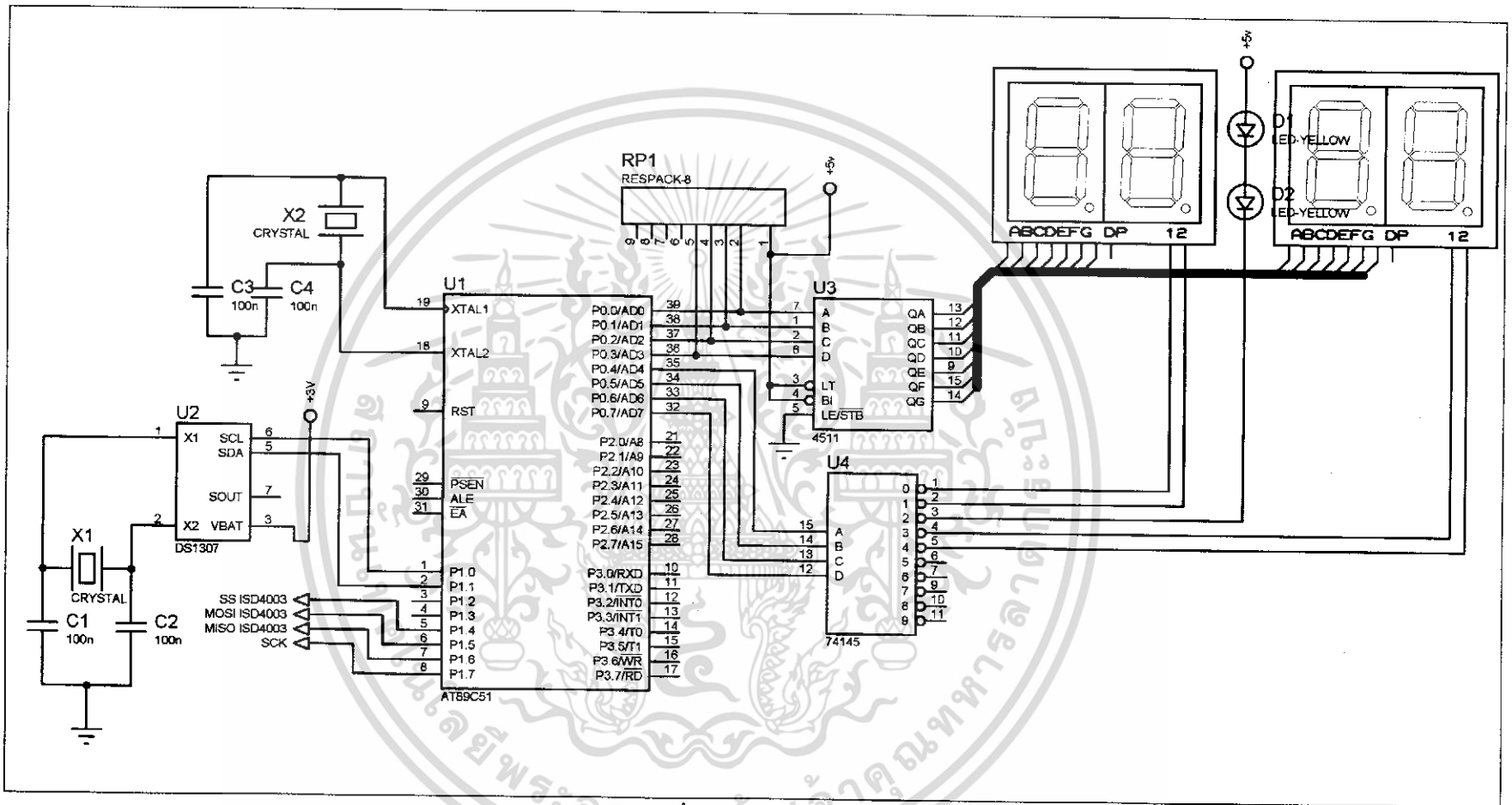

วงจรบอร์ดแสดงผล 7-segment module




วงจรควบคุมของป้ายรถประจำทาง




วงจรที่ใช้ในการเปลี่ยนระดับแรงดันจาก 3V เป็น




วงจรที่ใช้ในการแสดงเวลา

# Application Note

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A Winbond Company

ChipCorder®
TECHNOLOGY BY ISD

# Applications Note 5A

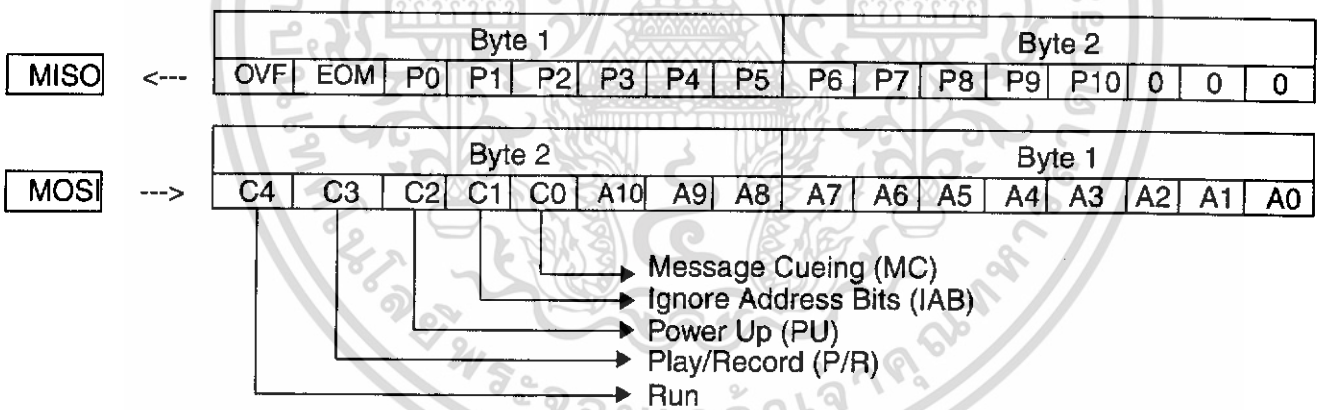
## The ISD4000 SPI Control Port Operations, a Simplified Guide

A simple but powerful command structure is built into the ISD4000 series SPI control port. It's inherent flexibility allows the software programmer to direct the operation of the ISD4000 device with the minimum number of control cycles while allowing full control of the device in it's several modes. The ISD4002 and ISD4003 devices use a 2-byte control structure and the ISD4004 uses a 3-byte control structure. See Applications Brief #40, "*The ISD4003 vs. the ISD4004.*"

For simplicity, the following discussion will refer specifically to the ISD4003. The following paragraphs give some tips on the use of this port and suggest some methods of simplifying some operations.

The SPI command registers in the ISD4003 have the following composition:

**Figure 1:**




The MOSI pin of the SPI port on the device is an input. MOSI stands for Master Out, Slave In. The MISO is an output. MISO stands for Master In, Slave Out. The ISD4003 is operating as a "slave" device to the microcontroller that is running the system. The SPI port also has a serial clock input called SCLK and a select input called SS. The SS stands for Slave Select and is an active LOW signal.

All input command cycles in the ISD4003 start with SS pin going LOW; and end with SS pin going HIGH. The bit position of the input data at the time the SS pin goes HIGH determines how the device reacts to the command. Any number of bits may be clocked into the SPI port on an input cycle. The last 5 bits input to the MOSI pin at the time the SS pin goes HIGH determine the command being input, even if many bytes were clocked into the port while the SS pin was LOW. Only the last 16 bits shifted into the SPI port are retained in the SPI port hardware.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The SCLK signal clocks the data into the ISD4003. The input data to the MOSI pin must be valid on the rising edge of the SCLK. The output data from the MISO pin changes following the falling edge of the SCLK pin. Consult the data sheet on the ISD4003 for the exact timing. The examples stated below assume the initial state of the SCLK pin is LOW when the SS pin goes LOW to start the command cycle but this is not required.

## EXAMPLE #1, POWER UP THE DEVICE

The following is an example of a simple command to the ISD4003 SPI port. In this operation, the device is to be powered up so that a Record or Playback cycle can be started later. Consult the diagram at the beginning of this Application Note that shows the SPI command registers. This drawing shows the bit positions of the 5 bits that compose the heart of the controls, e.g. C0 through C4. This diagram should be used as a reference when reading the explanations below:

1. The device is powered up by setting the PU bit. That is, bit C2 must be HIGH and bits C0, C1, C3 and C4 must all be LOW at the end of the command cycle. The 5 control bits in the SPI control register should then be set as <00100> where the bit positions are as indicated by the diagram above.
2. In this example, an entire byte will be shifted in, since some hardware SPI ports can only operate on whole bytes. It is necessary therefore to shift in HEX address 20, <0010 0000> just to set the C2 bit. Note that the state of the A9, A8 and "X" bits during this operation do not matter. This will be explained in more detail later.
3. The command cycle starts by changing SS to a LOW. The input to the MOSI pin is also LOW. The input byte, <0010 0000>, is to be clocked in, right to left. The first 5 bits clocked in, therefore, are all LOW. A single HIGH bit is then clocked in followed by two more LOW bits. This sequence is shown graphically below. The command ends with the SS pin going back HIGH and the ISD4003 will then begin a power up cycle. In $T_{PUD}$[1] time, the device will be powered up.

The table below shows the power up bit data being shifted from the command byte in the microcontroller into the MOSI pin of the ISD4003 and into byte 2 of the input side of the SPI input port register. Note that the PU bit is shifted to the right out of the microcontroller and into the ISD4003 SPI port.

1. TPUD is the power up delay time of the device. See the ISD4003 data sheet.

| Shift # | command byte in microcontroller | | SPI port in ISD4003 after "N" shifts | Shift # | command byte in microcontroller | | SPI port in ISD4003 after "N" shifts |
|---|---|---|---|---|---|---|---|
| Initial condx. | <0010 0000> | ---> | <0000 0000> | | | | |
| 1 | <0001 0000> | ---> | <0000 0000> | 5 | <0000 0001> | ---> | <0000 0000> |
| 2 | <0000 1000> | ---> | <0000 0000> | 6 | <0000 0000> | ---> | <1000 0000> |
| 3 | <0000 0100> | ---> | <0000 0000> | 7 | <0000 0000> | ---> | <0100 0000> |
| 4 | <0000 0010> | ---> | <0000 0000> | 8 | <0000 0000> | ---> | <0010 0000> |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

In this first example, we did not care about the value of the address bits A0 through A9. In fact, since the Run bit, C4, was input as a LOW only the HIGH PU bit had any effect on the ISD4003 and the state of C0, C1 and C3 also did not matter. *ISD recommends that anytime data is shifted into a SPI port command register bit position that "does not matter" it should be a "0" for compatibility with possible future features that may be added to the device.* Also, note that in some cases, it is necessary to give an ISD device more than one power up command to fully power up the device. See the appropriate data sheet.

## EXAMPLE #2, RECORD AT AN ADDRESS

This second example assumes the ISD4003 is already powered up and that recording is to begin at a specific address. In this example we will use address 92, or HEX address 5C, <00 0101 1100>. Each control bit in the input side of the command register of the device is set up as follows:

C0—The MC bit must be LOW since a message cueing cycle is not desired.

C1—The IAB bit must be LOW since the address is not being ignored.

C2—The PU bit must remain HIGH since the device is already powered up and must remain powered up after this command cycle.

C3—The P/R bit must be LOW since a Record cycle is to be started.

C4—The RUN bit must be HIGH since an active cycle is to be started.

A0 - A9—The address bits must be all be defined since the IAB bit is LOW.

1. The transfer of the two bytes of data is begun by changing the SS pin to a LOW. The bytes to be sent are: HEX address A0 5C, < 1010 0000> <0101 1100>.

2. Again, we are clocking the bits in, right to left[2] . Therefore, the first bit clocked into the SPI port will be the A0 address bit which is a "0," followed by A1 which is a "0," followed by A2 which is a "1" etc.

3. The last of the 16 bits clocked into the SPI port will be the RUN bit which, of course, is a "1."

4. The command cycle ends with the SS pin going back HIGH. At this time, the Record cycle starts at the address as specified by A0 through A9.

---

**NOTE** *After each command cycle, the data shifted into the ISD4003 remains in the chip's SPI port hardware. As subsequent operations shift more data in, bits shifted to the right out of the 16-bit register space are lost. A record or play operation started with only a 5 or 8 bit input command, and with the IAB bit not set (i.e. at "0") will cause the A0 through A9 data resident at the end of the operation to be treated as an address pointer. This will, of course, be data input to the SPI port in the previous command cycle mostly from the C0 through C4 bit positions. This is probably not valid address data.*

---

## EXAMPLE #3, PLAYBACK AT AN ADDRESS

This second example assumes the ISD4003 is already powered up and that playback is to begin at a specific address. In this example we will use address 92, or HEX address 5C, <00 0101 1100>. Each control bit in the input side of the command register of the device is set up as follows:

C0—The MC bit must be LOW since a message cueing cycle is not desired.

C1—The IAB bit must be LOW since the address is *not* being ignored.

C2—The PU bit must remain HIGH since the device is already powered up and must remain powered up after this command cycle.

---

2. Some microcontrollers with hardware SPI ports only have the ability to do a left to right shift. Consult the mP data sheet for details. In this case, the bit placement of the data shifted in would be inverted. In this example, the A0 would become 05 and the 5C would become 3A. When doing the bit placement inversion, don't forget that two of the address bits fall in the "control byte" of the data sent into the device.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C3—The P/R bit must be HIGH since a Playback cycle is to be started.

C4—The RUN bit must be HIGH since an active cycle is to be started.

A0 - A9—The address bits must be all be defined since the IAB bit is LOW.

1. The transfer of the two bytes of data is begun by changing the SS pin to a LOW. The bytes to be sent are: HEX address E0 5C, < 1110 0000> <0101 1100>.

2. The first bit clocked into the SPI port will be the A0 address bit which is a "0," followed by A1 which is a "0," followed by A2 which is a "1" etc.

3. The last of the 16 bits clocked into the SPI port will be the RUN bit which, of course, is a "1."

4. The command cycle ends with the SS pin going back HIGH. At this time, the Playback cycle starts at the address as specified by A0 through A9.

## EXAMPLE #4, STOP RECORD OR PLAYBACK AND POWER DOWN THE DEVICE

This example assumes that a Record cycle or a Playback cycle is underway and that the operation is to be terminated. The device is also to be powered down by this control cycle. The same command may be used to perform this function in both the Record and the Playback cycles. In this example, a full byte transfer is assumed.

1. A record or playback operation is terminated by clocking a command cycle into the SPI port with the RUN bit LOW. The PU bit is also low in this example so that the device powers down. The IAB bit must be HIGH at end of a record cycle to make sure the EOM gets set in the correct place. The rest of the bits in the command byte do not matter but by convention they are to be left LOW. The command byte needed, to end and power down a record cycle is therefore HEX address 10, <0001 0000> and the command byte needed to end and power down a playback cycle is HEX address 00, <0000 0000>.

2. Again, the transfer of the command byte of data is begun by changing the SS pin to a LOW followed by the byte as described above.

3. Record or playback will end in approximately 50 milliseconds (again, the data sheet will have exact information) followed by device power down.

## EXAMPLE #5, ADDRESS JUMP DURING RECORD OR PLAYBACK

One of the ISD4003's key features is to seamlessly jump from one address to any other address during the course of a Record or Playback cycle. This function enables the Message Management operations described in Application Note #7, "Message Management in the ISD4003 Series." The following discussion describes how this feature works:

The ISD4003 memory array may be thought of as an address map of 600 rows by 1600 columns. Only the rows may be addressed. When an operation begins on a row at address "n," it begins in column "0" and proceeds through the row to column 1199 if not stopped by a command cycle along the way.

The internal address sequencer in the ISD4003 operates in one of two possible modes. As a record or playback operation proceeds through the device, at the end of each row, a decision must be made as to where to go next. If the IAB bit is set ("1") when the end of the row is reached, then the address sequencer increments by 1 count and record or playback proceeds to the next row in sequence. If the IAB bit is cleared ("0") when the end of the row is reached, then the address sequencer pulls the address from the SPI command register and record or playback proceeds at the row defined by this address. This transition from the end of one row to the beginning of some other row not in sequence is seamless. No samples are lost as record or playback proceeds across the boundary without problems.

## INTERRUPT SERVICE

The interrupt service structure of the ISD4003 may be read with only 8 SPI clock cycles. The following example demonstrates how this is possible:



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## EXAMPLE #5, READ INTERRUPT STATUS AND CLEAR THE DEVICE INTERRUPT CONDITION

The ISD4003 device has only two interrupting conditions: (1) End Of Message (EOM) which signifies that a set EOM bit has been found during a Playback cycle and (2) Overflow (OVF) which indicates that the end of device memory has been encountered during either a Record or a Playback cycle. Note that an Overflow interrupt can only occur at the end of the last row in the device's memory.

Both of these conditions result from the end of device operation, i.e. playback or record has ceased. Both conditions cause the INT[3] pin of the device to be pulled LOW. This pin will remain LOW until the next SPI cycle in the ISD4003 device.

As the interrupt condition is read and clocked out of the MISO pin, data is being simultaneously being clocked into the MOSI pin. This input data will be interpreted as a control input and the device will react accordingly. It is necessary, therefore, to make sure this input data leaves the ISD4003 in a desired state. Since the interrupt always results from the end off an operation, shifting in a power-up command, as illustrated in example #1 above, is usually a safe response. In this example below, we are assuming we are shifting in HEX address 20, <0010 0000> while the interrupt data is being shifted out.

The following description shows how interrupt status may be read from the ISD4003:

1. The interrupt status read cycle begins with the SS pin of the device being changed to a LOW. At that time, the state of the OVF bit will be presented to the MISO output. This level is static. That is, it will remain low as long as the SS pin stays LOW and the SCLK pin is not clocked.

2. The first clock cycle of the input to the SCLK pin will now cause the state of the EOM bit to be presented to the MISO output.

3. Seven more clocks must be presented to the SCLK pin to insure the interrupt condition is properly cleared.

4. The interrupt status read is terminated by changing the SS pin back to a HIGH. When the SS pin goes HIGH, the INT pin output will be allowed to go back HIGH, pulled up by the external pull-up resistor.

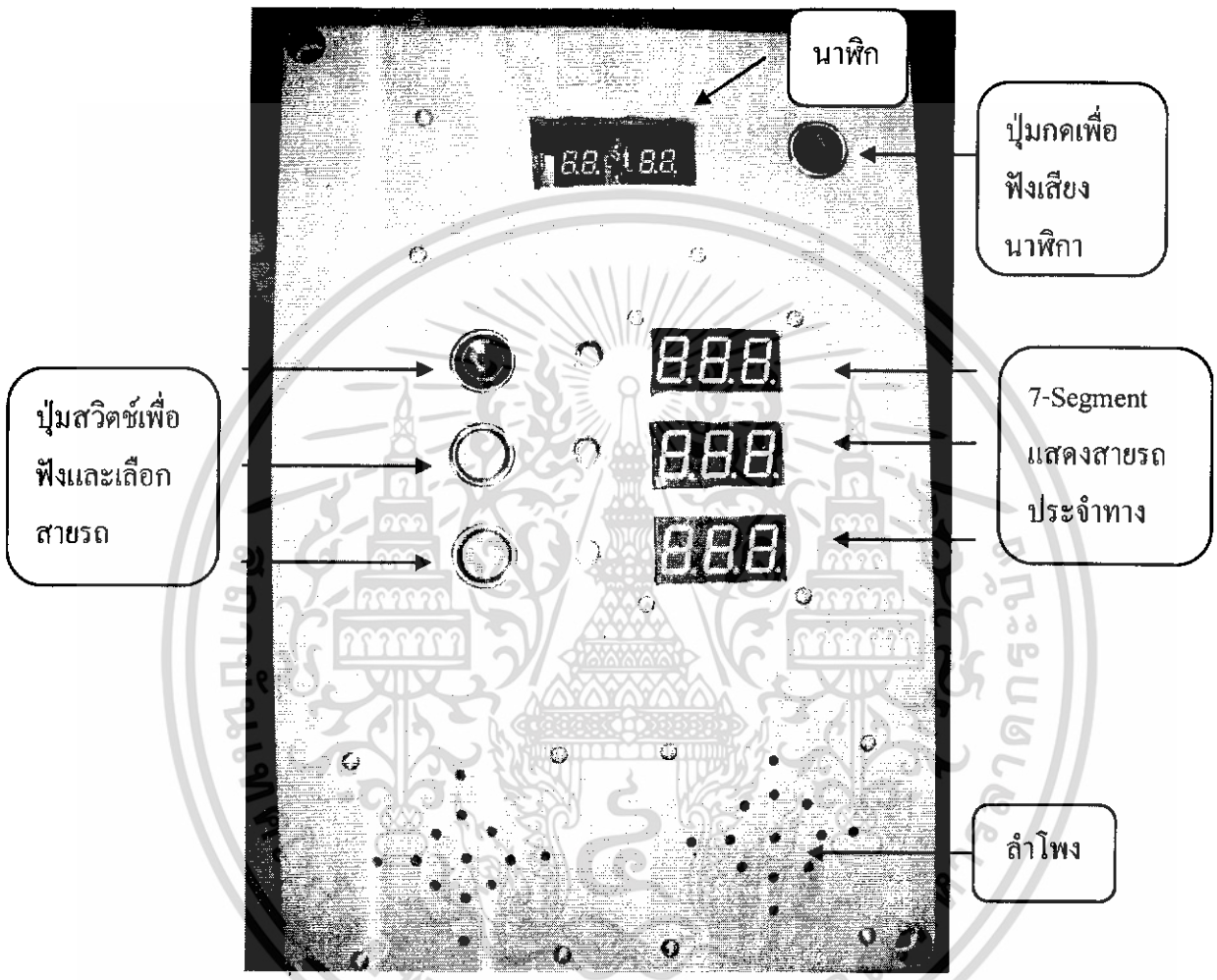
The above examples demonstrate the ease of use of the ISD4003 family device. A Record or Playback cycle may be started and controlled with a minimum number of SPI clock cycles. This translates into a voice record and playback system that requires very little overhead from the controller in the system.

3. INT is an active LOW open drain output. If this output is to be used, a pull-up resistor must be installed to pull the pin up to the proper HIGH level.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

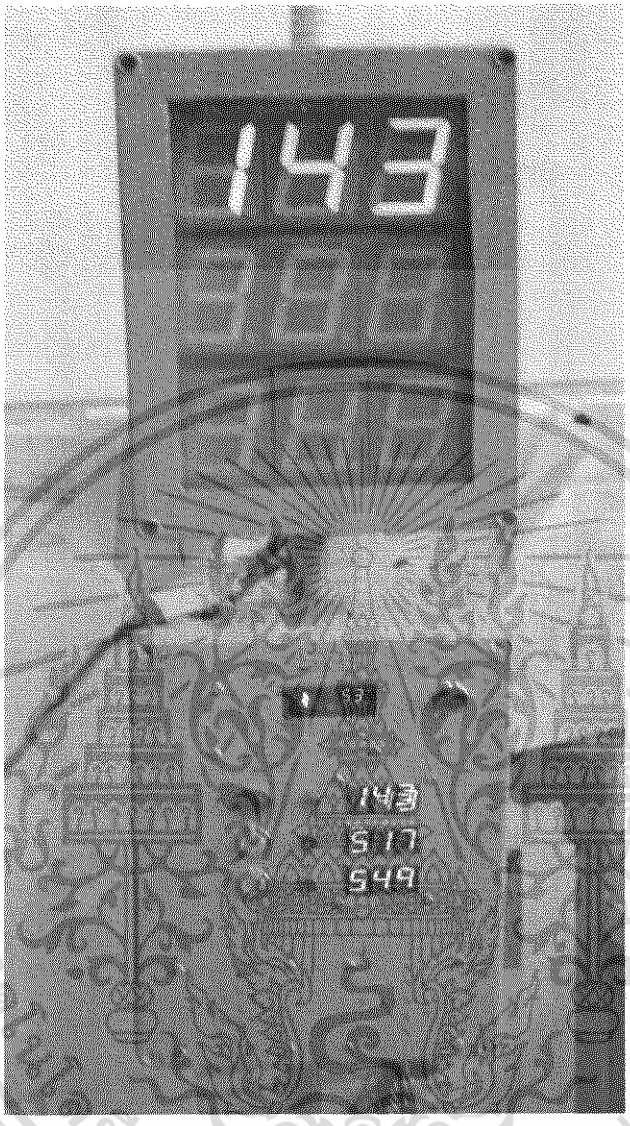
# คู่มือการใช้งานป้ายเรียกรถประจำทาง




**การใช้งานสำหรับผู้พิการทางสายตาและบุคคลทั่วไป**

1. กดสวิตช์เลือกสายรถเพื่อฟังข้อมูลของสายรถนั้นๆ (ที่ปุ่มจะมีอักษรเบลคำกับอยู่ว่าเป็นรถสายใด)
2. ฟังเสียงของสายที่เลือก โดยจะบอกข้อมูลของสายที่เลือกว่าเป็นรถหมายเลขใด เดินทางจากไหนไปไหน
3. เมื่อฟังแล้ว หากต้องการเลือกสายนั้นให้ทำการกดสวิตช์เลือกสายนั้นๆค้างไว้จนได้ยินคำว่า "เลือก" จึงปล่อยสวิตช์ออก
4. สายรถที่ถูกเลือกจะติดขึ้นที่ จอแสดงให้คนขับรถประจำทางทราบทางด้านบน ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**การใช้งานในการตั้งค่าสายรถประจำทางและการบันทึกเสียง**

ในโครงงานงานนี้เมื่อตอนเปิดเครื่องจะมีการตั้งค่าของสายรถประจำทางมาให้แล้ว 3 สาย รวมทั้งตัว 7-Segment module ด้านบนก็จะทำการตั้งมาแล้วเช่นกัน หากเราต้องการเปลี่ยนค่าดังกล่าวสามารถทำได้ตามขั้นตอนดังต่อไปนี้

1. กดสวิตช์สำหรับตั้งสายรถประจำทาง (เขียนบอกไว้ที่ตัวสวิตช์) ค่าสายรถประจำทางทั้งหมดจะถูกลบไป
2. ทำการบันทึกค่าสายรถประจำทางทีละสาย (เลข 3 หลัก)ผ่านทาง Keypad ที่อยู่ทางด้านหลังกล่อง โดยหน้าจอ 7-Segment จะโชว์สายรถประจำทางที่ตั้งทุกๆครั้งที่กดเลขครบสามหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. หากกดผิดหรือต้องการเปลี่ยนค่าสายก่อนหน้านี้ให้กดสวิตช์ "เคลียร์" แล้วทำตามข้อ 1. ใหม่เพื่อบันทึกค่า อนึ่งการกดสวิตช์ "เคลียร์" สามารถกระทำได้ตั้งแต่แรกสุดเช่นกัน โดยจะเป็นการเคลียร์ค่าสายรถสายที่ 3
4. ในการจะตั้งค่าที่ 7-Segment module จะต้องทำพร้อมกับการตั้งค่าสายรถตามขั้นตอนที่1. โดยเราจะต้องนำสายสัญญาณรับข้อมูลแบบอนุกรม (RX ของ Serial port) ของ 7-Segment module (เขียนบอกอยู่ที่สายสัญญาณ) มาเชื่อมต่อกับสายส่งสัญญาณอนุกรมของ MCS-51 ตัวควบคุมการเลือกสายรถประจำทาง โดยในที่นี้เราจะทางเป็นสวิตช์สามทางไว้ให้เลือก หากต้องการตั้งค่าของ 7-Segment module บอร์ดไหนก็ให้สับสวิตช์มาที่ตำแหน่งนั้น แล้วทำการตั้งค่าสายรถประจำทางตามปรกติ 7-Segment module จะทำการเก็บค่า ตัวเลขโดยอัตโนมัติ

หลักจากตั้งค่าสายรถประจำทางที่ต้องการแล้วจะทำการอัดเสียงของระบบดังนี้

1. กดสวิตช์ "บันทึกเสียง" แล้วพูด คำว่า "สาย" แล้วจึงกดสวิตช์ "หยุด" สังเกตุเสียงของลำโพงระหว่างที่ไม่ได้บันทึกกับระหว่างที่บันทึก จะต่างกันทั้งนี้เพราะตอนที่ไม่มีการบันทึก ISD4003 จะยังไม่เข้าสู่สภาวะเริ่มทำงาน (Power up)
2. หลังจากที่บันทึกคำว่าสายแล้ว ให้ทำการกดสวิตช์ "บันทึกเสียง" แล้วพูด ข้อมูลเกี่ยวกบสายรถเมล์สายที่หนึ่ง ทั้งเลขและทิศทางว่าจากไหนไปไหน แล้วจึงกดสวิตช์ "หยุด" ทำซ้ำเช่นเดียวกันนี้สำหรับสายรถที่สองและสาม
3. เมื่อบันทึกข้อมูลเรียบร้อยทั้งสามสายแล้วให้ทำเช่นเดียวกับขั้นตอนที่ 1. แต่เปลี่ยนเป็นบันทึกคำว่า "เลือก" แทน
4. เมื่อทำการบันทึกเสียงครบหมดแล้วหากต้องการบันทึกใหม่ ต้องทำการปิดเครื่องแล้วเปิดใหม่เพื่อเป็นการเคลียร์ค่าเวกเตอร์ที่ใช้ชี้แอดเดรสของ ISD4003 แต่เราสามารถบันทึกใหม่แค่สายแรกได้หากไม่ต้องการเปลี่ยนข้อมูลอีกสองสาย

ข้อควรระวังในการบันทึกเสียงคือ ไม่ควรจะกดบันทึกแล้วปล่อยทิ้งไว้นานจนเกินไปแล้วจึงพูด เพราะจะทำให้เกิดการทับกันของเสียงที่พูดไปก่อนหน้ากับเสียงที่พูดทีหลัง ทั้งนี้เนื่องมาจากการเก็บข้อมูลเสียงลง ISD4003 นั้นจะทำการเก็บเป็นแอดเดรส ซึ่งจะเรียงต่อไปกันไปและมีความยาวของแต่และแอดเดรสเป็นค่าๆหนึ่ง (ขึ้นอยู่กับการเขียนโปรแกรมและความจุของ ISD4003 เบอร์ที่ใช้)

การบันทึกค่าเสียงและตั้งค่านาฬิกาสามารถกระทำได้คล้ายๆกัน จะต่างกับการตั้งสายรถประจำทางตรงที่พอร์ทที่ใช้เชื่อต่อกับ Keypad (ขึ้นอยู่กับโปรแกรมสามารถแก้ไขได้) และข้อความที่บันทึก โดยจะบันทึกข้อความ 0-9,10,20 และ คำว่า "ขณะนีเวลา" คำว่า "นาฬิกา" และคำว่า "นาที" ตัวโปรแกรมจะทำการเปรียบเทียบค่าเวลาและทำการเล่นเสียงเองโดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้