

รถไฟฟ้าสำหรับคนพิการ  
ELECTRIC WHEELCHAIR



โดย  
นายปกาศิต วรเจริญ  
นายประพันธ์ ฉลภิญโญ

เลขหมู่.....  
เลขทะเบียน 61849  
วัน,เดือน,ปี 21 ก.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมอิเล็กทรอนิกส์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถไฟฟ้าสำหรับคนพิการ  
ELECTRIC WHEELCHAIR



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมอิเล็กทรอนิกส์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# รถไฟฟ้าสำหรับคนพิการ

## ELECTRIC WHEELCHAIR

โดย นายปกาศิต วรเจริญ 44010274

นายประพันธ์ ฉลภิญโญ 44010283

อาจารย์ที่ปรึกษา ผศ. พลผดุง ผดุงกุล

### บทคัดย่อ

ปัจจุบันงานทางด้านไมโครคอนโทรลเลอร์ได้เข้ามามีบทบาทในชีวิตประจำวันอย่างมากในหลาย ๆ ด้าน โดยเฉพาะในระบบควบคุมอุปกรณ์ทางไฟฟ้าต่าง ๆ ซึ่งแต่เดิมเราจำเป็นต้องต่อวงจรด้วยอุปกรณ์หลายชนิดเพื่อควบคุมอุปกรณ์ทางอิเล็กทรอนิกส์และต้องใช้เกทจำนวนมากเพื่อให้ได้วงจรที่มีเงื่อนไขตามที่ต้องการ เราจึงได้นำเอาไมโครคอนโทรลเลอร์มาใช้ในการพัฒนางานทางด้านอิเล็กทรอนิกส์

ในรายงานฉบับนี้จะเป็นการนำเอาไมโครคอนโทรลเลอร์ MCS-51 มาเชื่อมต่อเข้ากับอุปกรณ์ต่าง ๆ เพื่อควบคุมความเร็วรอบในการหมุนของมอเตอร์ และต่อเข้ากับอุปกรณ์ joystick ใช้สำหรับเป็นตัวควบคุมการทำงานของมอเตอร์ เพื่อพัฒนาเป็นรถไฟฟ้าสำหรับคนพิการได้

### ABSTRACT

Nowadays, our daily lives are closely involved with the microcontrollers. Their multi-purpose functionalities pave their ways into various fields, especially where people want to spend less time on tedious work ,such as electrical circuit controlling systems, which we once needed several complicating elements in the circuit and a number of logic gates to produce some simple outputs. Now, microcontrollers can be found everywhere that development takes place.

In this report, the MCS-51 is used with certain equipments to control the speed of a motor via a joystick. The microcontroller will handle the calculation by following the specified program that has been recorded earlier. It'll expand to be the electrical wheel chair for crippled people

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2547

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง รถไฟฟ้าสำหรับคนพิการ

**ELECTRIC WHEELCHAIR**

ผู้จัดทำ

1. นายปกาศิต วรเจริญ 44010274
2. นายประพันธ์ ฉลภิญโญ 44010283

ลายเซ็นของอาจารย์ที่ปรึกษา

( ผศ. พลพวง ผดุงกุล )



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
<b>บทที่ 1 บทนำ</b>	1
<b>บทที่ 2 ทฤษฎีหรือหลักการ</b>	2
- โครงสร้างของไมโครคอนโทรลเลอร์ MCS-51	2
- คุณสมบัติของไมโครคอนโทรลเลอร์ ตระกูล MCS-51 อนุกรม AT89xx	2
- การจัดขาของไมโครคอนโทรลเลอร์ MCS-51	3
- โครงสร้างและการทำงานของพอร์ต	7
- การใช้งานเป็นพอร์ตอินพุต	8
- การใช้งานเป็นพอร์ตเอาต์พุต	9
- การอ่านค่าลอจิกจากพอร์ต	11
- จังหวะการทำงานของไมโครคอนโทรลเลอร์ MCS-51	11
- การจัดหน่วยความจำของ ไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลช	14
- หน่วยความจำโปรแกรม (Program memory)	14
- หน่วยความจำข้อมูล (Data memory)	17
- รีจิสเตอร์ฟังก์ชันพิเศษ	19
- รีจิสเตอร์แสดงสถานะของโปรแกรม	20
- แอ็กคิวมูเลเตอร์ (Accumulator: ACC)	20
- รีจิสเตอร์ B	20
- โปรแกรมเคาน์เตอร์ (Program Counter: PC)	21
- สแต็กพอยน์เตอร์ (Stack Pointer: SP)	21
- รีจิสเตอร์ชี้ข้อมูลหรือค้ำพอยน์เตอร์ (Data Pointer: DPTR)	22
- รีจิสเตอร์พอร์ต (Port register)	22
- รีจิสเตอร์บัฟเฟอร์ข้อมูลอนุกรม (Serial Data Buffer: SBUF)	22
- รีจิสเตอร์ไทมเมอร์ (Timer register)	22
- รีจิสเตอร์แคปเจอร์ (Capture register)	23
- รีจิสเตอร์ควบคุม (Control register)	23
<b>บทที่ 3 เทคโนโลยี IGBT</b>	25
- เทคโนโลยี IGBT	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
- โครงสร้างและสัญลักษณ์	26
- สถานะนำกระแส	27
- สถานะหยุดนำกระแส	28
- การแลตช์ใน IGBT	29
- การป้องกันการแลตช์	30
- วงจรสมมูลของIGBT	31
- ลักษณะการ Switch	33
<b>บทที่ 4 ทฤษฎี Analog To Digital Converter</b>	36
- Counting Converter	36
- Successive Approximation	37
- Dual-Slope ADC	39
- Flash Converter	42
- บทสรุป	42
<b>บทที่ 5 หลักในการออกแบบวงจร</b>	44
- แผนภาพของโครงการ	44
- 1. ภาคควบคุม	44
- 2. ภาค MOTOR DRIVER	47
<b>บทที่ 6 ซอร์ฟแวร์และการทำงาน</b>	50
- มีการทำงานดังนี้	50
- FLOWCHART การทำงาน	50
- 1. โปรแกรมย่อยอ่านข้อมูล read1	52
- 2. โปรแกรมย่อย NAV แปลงความเร็วและทิศทาง	54
- 3. โปรแกรมย่อย CONVERT แปลงค่าความเร็วและทิศทางเป็น TON และ TURN	56
- แสดง โปรแกรมสมบูรณ์ของโครงการนี้	57
<b>บทที่ 7 โครงสร้างและการประกอบ</b>	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
- โครงสร้าง	69
- แมคคานิก	70
- วัสดุที่ใช้ประกอบโครงงาน	70
- วงจร	71
- ภาพต่างๆ	77
<b>บทที่ 7 ปัญหาที่พบและการแก้ไข</b>	79
- 1. วงจร	79
- 1.1 อุปกรณ์เสีย	79
- 1.2 สายไฟมีจำนวนมาก	80
- 1.3 ค่าอุปกรณ์บางตัวที่เหมารวม	80
- 2. ซอร์ฟแวร์	80
- 3. แมคคานิก	81
- 3.1 ความแข็งแรง	81
- 3.2 น้ำหนัก	81
<b>บทที่ 9 สรุปผลการทดลอง</b>	82
- วงจร	82
1. ADC Converter	82
2. Drive และ IGBT	84
- ซอร์ฟแวร์	85
- แมคคานิก	86
<b>บทที่ 10 แนวทางการพัฒนาต่อ</b>	87
- วงจร	87
- ซอร์ฟแวร์	87
- แมคคานิก	87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

หน้า

ภาคผนวก

กิตติกรรมประกาศ

หนังสืออ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

หน้า

รูปที่ 1	รูปร่างลักษณะ โครงสร้างของรถไฟฟ้าสำหรับคนพิการส่วน โครงเหล็ก	1
รูปที่ 2.1	โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชอนุกรม AT89Cxx	4
รูปที่ 2.2	โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชอนุกรม AT89Sxx	4
รูปที่ 2.3	รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชของ Atmel	6
รูปที่ 2.4	การจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51	6
รูปที่ 2.5	วงจรภายในของพอร์ตในการใช้งานเป็นพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	10
รูปที่ 2.6	วงจรพูลอัพภายในพอร์ตของ ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	10
รูปที่ 2.7	ไซเกิลการทำงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	12
รูปที่ 2.8	ไดอะแกรมเวลาแสดงการติดต่อและเข้าถึงหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	13
รูปที่ 2.9	การจัดหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	16
รูปที่ 2.10	การเชื่อมต่อหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51	17
รูปที่ 2.11	การเชื่อมต่อหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51	18
รูปที่ 2.12	การจัดสรรพื้นที่ของหน่วยความจำข้อมูลภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	19
รูปที่ 2.13	การจัดสรรพื้นที่ของหน่วยความจำข้อมูลภายในส่วนต่างของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	19
รูปที่ 2.14	โครงสร้างของหน่วยความจำข้อมูลภายในส่วนบนของไมโครคอนโทรลเลอร์ MCS-51	21
รูปที่ 2.15	การจัดสรรพื้นที่ของรีจิสเตอร์ฟังก์ชันพิเศษ	22
รูปที่ 3.1	สัญลักษณ์และการเรียกชื่อขาของ IGBT ทั้ง 2 แบบ	26
รูปที่ 3.2	ภาพตัดขวาง โครงสร้างพื้นฐานของ IGBT	26
รูปที่ 3.3 (ก)	กราฟแสดงลักษณะคุณสมบัติระหว่างกระแสและแรงดันของ IGBT (ข) กราฟแสดงลักษณะสมบัติการถ่ายโอนของ IGBT	27
รูปที่ 3.4	ทิศทางการไหลของอิเล็กตรอนและโฮลในขณะนำกระแส	28
รูปที่ 3.5	โครงสร้างที่ปรับปรุงเพื่อป้องกันการแลตช์ใน IGBT	30
รูปที่ 3.6 (ก)	แสดงโครงสร้างที่มีทรานซิสเตอร์และมอสเฟตแฝงอยู่ภายใน (ข) วงจรสมมูลสำหรับการทำงานสภาพปกติของ IGBT (ค) วงจรสมมูลที่แสดงส่วนของทรานซิสเตอร์ที่แฝงอยู่ใน IGBT	31
รูปที่ 3.7 (ก)	ลักษณะของกระแสและแรงดันขณะนำกระแส (ข) ลักษณะของกระแสและแรงดันขณะหยุดนำกระแส	33
รูปที่ 3.8 (ก)	พื้นที่การทำงานที่ปลอดภัยในสภาวะไบแอสตรง (ข) พื้นที่การทำงานที่ปลอดภัยในสภาวะไบแอสกลับ	35
รูปที่ 4.1	Analog To Digital Converter	36
รูปที่ 4.2	Output voltage Graph from A/D converter	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

หน้า

รูปที่ 4.3 Binary Search Strategy	38
รูปที่ 4.4 Binary Search	39
รูปที่ 4.5 Dual-Slope ADC	39
รูปที่ 4.6 Dual Slope A/D Converter Output and Timing	40
รูปที่ 4.7 Dual Slope A/D Converter – Full Circuit	41
รูปที่ 4.8 Dual Slope A/D Converter – Zero Offset	41
รูปที่ 4.9 Flash Converter	42
รูปที่ 4.10 Summary of A/D Converter	43
รูปที่ 5.1 แผนภาพแสดงการทำงานของโครงงานทั้งหมด	44
รูปที่ 5.2 แสดงการเชื่อมต่อสัญญาณระหว่าง ไมโครคอนโทรลเลอร์กับ ADC0832	45
รูปที่ 5.3 แสดงการทำงานของการทำงานของมอเตอร์ที่หมุน	46
รูปที่ 5.4 แสดงการเชื่อมต่อไฟท้ายหรือ ไฟแสดงผล	46
รูปที่ 5.5 แสดงการเชื่อมต่อ MOSFET, IGBT กับ HIP4082	47
รูปที่ 5.6 แสดงฟังก์ชันการทำงานของไอซี และตำแหน่ง High Side และ Low Side ของทรานซิสเตอร์	48
รูปที่ 5.7 แสดงการทำงานของไอซีที่ทำให้ มอเตอร์หมุนทางซ้าย	49
รูปที่ 5.8 แสดงการทำงานของไอซีที่ทำให้ มอเตอร์หมุนทางขวา	49
รูปที่ 5.9 แสดงการทำงานของไอซีที่ทำให้ มอเตอร์หยุดหมุน	49
รูปที่ 6.1 แสดงสัญญาณที่ขา Gate ของ IGBT	50
รูปที่ 6.2 แสดงโพลชาร์ตการทำงานของโปรแกรมทั้งหมด	51
รูปที่ 6.3 แสดงสัญญาณสำหรับติดต่อกับไอซี ADC0832	52
รูปที่ 6.4 แสดง Time Diagram ที่โปรแกรมย่อย read1 สร้างขึ้น เพื่อแปลงสัญญาณอนาลอกเป็นดิจิตอลช่องที่ 1	52
รูปที่ 6.5 แสดง Time Diagram ที่โปรแกรมย่อย read1 สร้างขึ้นเพื่อแปลงสัญญาณอนาลอก เป็นดิจิตอลช่องที่ 2	53
รูปที่ 6.6 แสดงโพลชาร์ตการทำงานของโปรแกรมย่อย read1	53
รูปที่ 6.7 แสดงการแบ่งแกนของจอยสติ๊กเพื่อกำหนดค่าตัวแปร	54
รูปที่ 6.8 แสดงรีจิสเตอร์ PS_WF ซึ่งอนุমানขึ้นเพื่อใช้ในการบอกทิศทางให้กับรถ เพื่อให้ โปรแกรมเช็คได้	55
รูปที่ 6.9 แสดงโพลชาร์ตการทำงานของโปรแกรมย่อย NAV	56
รูปที่ 6.10 แสดงวิธีที่รถเคลื่อนที่ไปในทิศทางต่างๆ	56
รูปที่ 6.11 แสดงโพลชาร์ตการทำงานของโปรแกรมย่อย CONVERT	57
รูปที่ 7.1 แสดงหมายเลขซึ่งแทนค่า ID ของเหล็ก	69
รูปที่ 7.2 แสดงหมายเลขซึ่งแทนค่า ID ของเหล็ก	69
รูปที่ 7.3 แสดงหมายเลขซึ่งแทนค่า ID ของเหล็ก	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

	หน้า
รูปที่ 7.4 แสดงการเดินสายและติดตั้งวงจร	71
รูปที่ 7.5 ได้ Zoom เข้ามาใกล้กับแผ่นวงจรภาคควบคุมมากขึ้น	72
รูปที่ 7.6 แสดงสวิตช์ที่ใช้ปิด-เปิด ไฟท้าย , หยุดการทำงาน	72
รูปที่ 7.7 แสดงอีกมุมหนึ่งของการติดตั้งวงจร	73
รูปที่ 7.8 แสดงภาพด้านหลังของรถ ซึ่งจะเห็นการติดตั้งไฟท้ายและวงจร	73
รูปที่ 7.9 แสดงภาพด้านข้างของตัวรถ	74
รูปที่ 7.10 แสดงภาพด้านหน้าตัวรถ	74
รูปที่ 7.11 แสดงที่นั่งและการติดตั้งจอสตีก	75
รูปที่ 7.12 แสดงจอสตีกที่ใช้ในโครงการ	75
รูปที่ 7.13 แสดงจอสตีก 2 ตัวที่นำมาใช้งาน	75
รูปที่ 7.14 แสดงแกนโยกที่จะต้องถอดออกมาใช้	76
รูปที่ 7.15 แสดงบริเวณแกนโยกที่จะถูกตัดมาใช้	76
รูปที่ 7.16 ภาพด้านหน้าเมื่อเทียบกับเก้าอี้ที่นั่ง	77
รูปที่ 7.17 ภาพด้านข้างแสดงการต่อมอเตอร์ขับเคลื่อน	77
รูปที่ 7.18 ภาพด้านข้างแสดงการต่อมอเตอร์ขับเคลื่อน	78
รูปที่ 7.19 ขยายมาที่การต่อมอเตอร์ไปยังล้อ	78
รูปที่ 9.1 แสดงสัญญาณที่ขา CS ของ ADC0832	82
รูปที่ 9.2 แสดงสัญญาณที่ขา CLK ของ ADC0832	83
รูปที่ 9.3 แสดงสัญญาณที่ขา DI ของ ADC0832	83
รูปที่ 9.4 แสดงสัญญาณที่ขา DO ของ ADC0832	83
รูปที่ 9.5 แสดงสัญญาณที่ขั้วมอเตอร์ขณะที่ไม่ได้ต่อมอเตอร์ที่ Duty Cycle 70%	84
รูปที่ 9.6 แสดงสัญญาณที่ขั้วมอเตอร์ขณะที่ไม่ได้ต่อมอเตอร์ที่ Duty Cycle 90%	84
รูปที่ 9.7 แสดงสัญญาณที่ขั้วมอเตอร์ ขณะต่อมอเตอร์ที่ Duty Cycle 70%	85
รูปที่ 9.8 แสดงสัญญาณที่ขั้วมอเตอร์ ขณะต่อมอเตอร์ที่ Duty Cycle 90%	85

## สารบัญตาราง

	หน้า
ตารางที่ 2.1 รายละเอียดโดยสรุปบางส่วนขง ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชที่ Atmel ผลิตขึ้น	4
ตารางที่ 2.2 หน้าที่พิเศษของพอร์ต 1 ใน ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชของ Atmel	9
ตารางที่ 2.3 การเลือกแบงก์ของหน่วยความจำส่วนล่างเพื่อติดต่อกับรีจิสเตอร์แบงก์ R0-R7	22
ตารางที่ 5.1 แสดงสัญญาณควบคุม HIP4082	48
ตารางที่ 7.1 แสดงวัสดุที่ใช้ประกอบโครง ความยาว และจำนวนที่ใช้	71



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

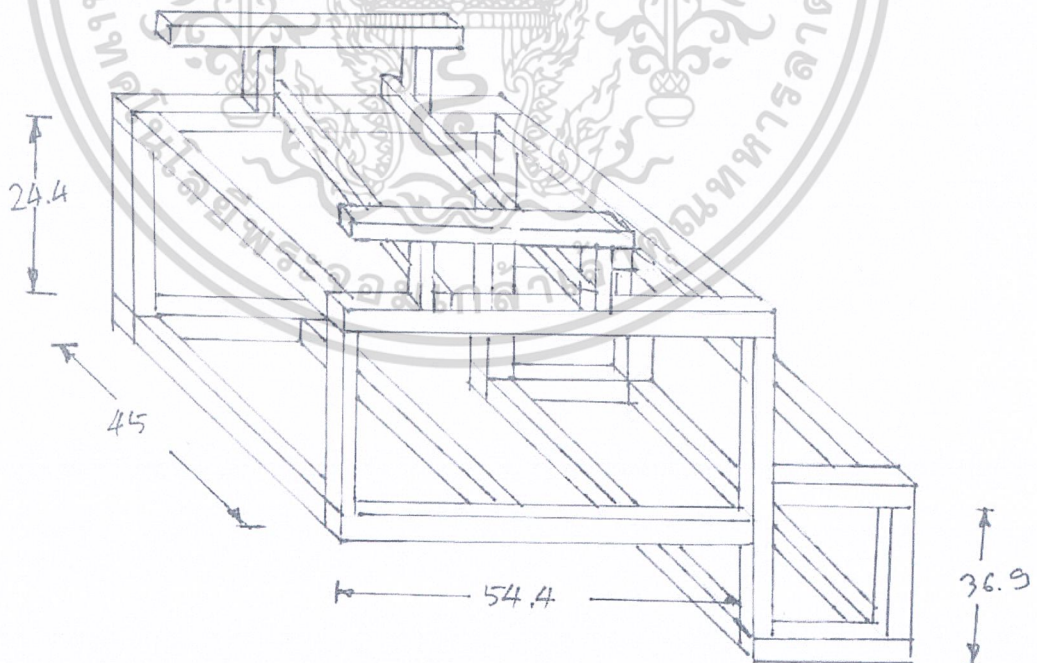
### รถไฟฟ้าสำหรับคนพิการ

#### โครงการรถไฟฟ้าสำหรับคนพิการ

มีหลักการทำงานคือ ใช้มอเตอร์กระแสตรง 12 - 24 V เป็นตัวทำให้รถเคลื่อนที่ได้ โดยติดมอเตอร์ 2 ตัวสำหรับควบคุมให้รถสามารถเลี้ยวซ้าย เลี้ยวขวา เดินหน้า ถอยหลังและใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมมอเตอร์ ซึ่งสามารถทำให้รถวิ่งช้า วิ่งเร็วได้ ทั้งนี้ คนพิการสามารถควบคุมตัวรถได้โดยใช้ joystick ตัวรถมีโครงสร้างเป็นเหล็กทั้งคัน เพื่อให้รถมีความแข็งแรงทนทาน สามารถรับน้ำหนักของคนพิการได้

#### ส่วนประกอบของรถไฟฟ้าสำหรับคนพิการ

1. วงจรส่วนควบคุมการทำงาน (MCS-51)
2. วงจร Drive motor และ HIP4082
3. วงจรส่วนของ ADC ในการรับสัญญาณควบคุมจาก joystick
4. ส่วนของตัวรถ



รูปที่ 1 รูปร่างลักษณะ โครงสร้างของรถไฟฟ้าสำหรับคนพิการส่วน โครงเหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2 ทฤษฎีหรือหลักการ

### โครงสร้างของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ที่ใช้ในวงจรเป็นไมโครคอนโทรลเลอร์ตระกูล MCS-51 สาเหตุที่เลือกไมโครคอนโทรลเลอร์ตระกูลนี้ เพราะ ไมโครคอนโทรลเลอร์ตระกูลนี้มีข้อดีหลายประการคือ

1. หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ทำให้สามารถลบและเขียนใหม่ได้นับพันครั้ง จึงสามารถใช้งานในรูปแบบของไมโครคอนโทรลเลอร์ชิปเดี่ยวไม่ต้องใช้หน่วยความจำภายนอก ส่งผลให้สามารถใช้งานงาน พอร์ตอินพุต เอาต์พุต ของไมโครคอนโทรลเลอร์อย่างเต็มประสิทธิภาพ

2. ต้นทุนและเวลาในการพัฒนาระบบไมโครคอนโทรลเลอร์ลดลงอย่างมาก เนื่องจากไม่ต้องใช้เครื่องมือพัฒนาจำพวกอิมูเลเตอร์และเครื่องโปรแกรมอีพรอม

3. บริษัทผู้ผลิตได้ทำการผลิตไมโครคอนโทรลเลอร์ตระกูลนี้ออกมาหลายเบอร์ และมีความสามารถแตกต่างกันไป ทำให้มีทางเลือกในการใช้งาน

4. ด้วยการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ ทำให้สามารถป้องกันการคัดลอกข้อมูลของหน่วยความจำโปรแกรมได้เป็นอย่างดี

5. ในบางเบอร์ของไมโครคอนโทรลเลอร์ที่ผลิตโดยAtmel สามารถทำการโปรแกรมข้อมูลในหน่วยความจำโปรแกรมได้โดยไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกมาทำการโปรแกรมใหม่หรือเรียกว่าการโปรแกรมในวงจร หรือในระบบ (In-system programming) ทำให้การพัฒนาหรือการซ่อมบำรุง ตลอดจนการปรับปรุงหรืออัปเดตข้อมูลในหน่วยความจำโปรแกรมทำได้สะดวก ภายใต้งบประมาณที่ไม่สูงมากนัก

6. ชุดคำสั่งและสถาปัตยกรรมพื้นฐานเหมือนกับไมโครคอนโทรลเลอร์ MCS-51 ของผู้ผลิตอื่น ไม่ว่าจะเป็นอินเทล ซิเมนส์ หรือ คัลลัส

### คุณสมบัติของไมโครคอนโทรลเลอร์ ตระกูล MCS-51 อนุกรม AT89xx

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีพรอม เพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานเป็น ได้ทั้งอินพุตและเอาต์พุต
- มีวงจรสื่อสารอนุกรมแบบฟูลดูพลีกซ์
- ไทเมอร์ (เคาน์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ในภายในชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89sxx
- มีวอตช์ดีค็อกไทเมอร์ในตัว สำหรับในอนุกรม AT89sxx

ในรูปที่ 2.1 เป็นโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 ในอนุกรม AT89cxx จะเห็นได้ว่าโครงสร้างของ AT89cxx จะเหมือนกับไมโครคอนโทรลเลอร์ตระกูล MCS-51 พื้นฐาน หากแต่แตกต่างกันเฉพาะหน่วยความจำโปรแกรมแบบแฟลชที่เพิ่มเติมเข้ามา หากเป็นไมโครคอนโทรลเลอร์ในอนุกรม 87xx หน่วยความจำโปรแกรมภายในจะเป็นแบบอีพรอม และบางเบอร์สามารถโปรแกรมได้เพียงครั้งเดียว

สำหรับในรูปที่ 2.2 เป็นโครงสร้างพื้นฐานของอนุกรม AT89sxx จะเห็นได้ว่า มีส่วนประกอบที่เพิ่มเติม แตกต่างจาก AT89cxx อีกหลายส่วน อาทิ วงจรเชื่อมต่ออนุกรมแบบ SPI ซึ่งในไมโครคอนโทรลเลอร์อนุกรมนี้ใช้ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมโดยไม่ต้องถอดตัวชิปออกไปจากระบบหรือเรียกว่าการโปรแกรมในวงจร ไทเมอร์/เคาน์เตอร์ขนาด 16 บิต ที่เพิ่มเติมเข้ามาอีกหนึ่งตัวเป็น ไทเมอร์ 2 และวงจรวอตช์ดีค็อกที่ใช้ในการตรวจสอบการทำงานผิดพลาดของชิป

ในตารางที่ 2.1 แสดงรายละเอียดบางส่วน of ไมโครคอนโทรลเลอร์ตระกูล MCS-51 แต่ละเบอร์ที่ Atmel ผลิตขึ้น และมีใช้งานอยู่ในปัจจุบัน

### การจัดขาของไมโครคอนโทรลเลอร์ MCS-51

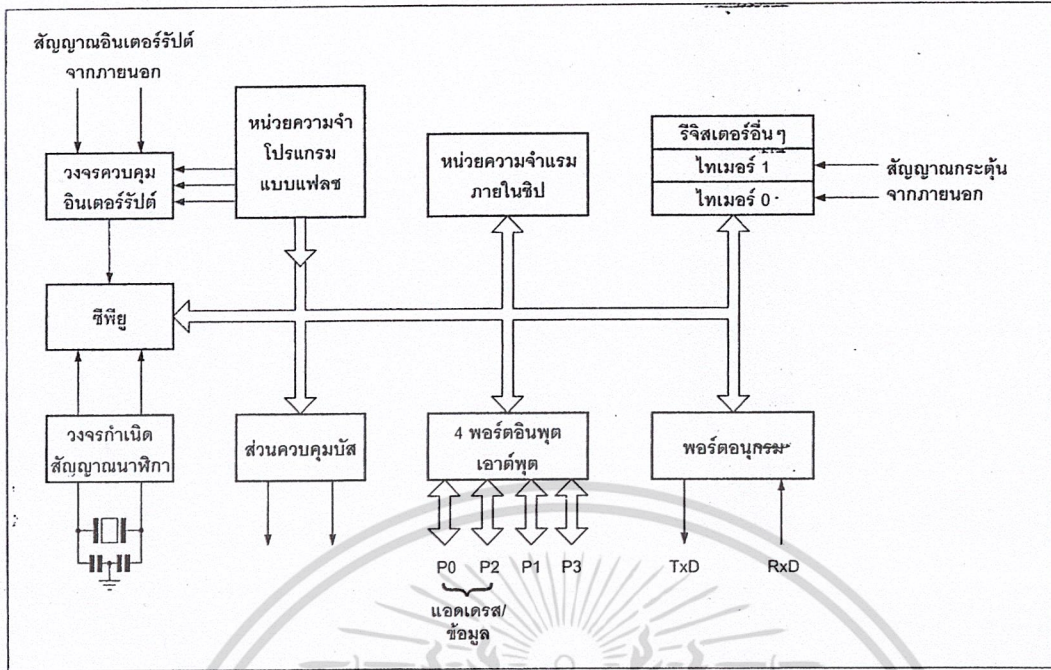
ไมโครคอนโทรลเลอร์ MSC-51 ทุกเบอร์จะมีสถาปัตยกรรมและขาใช้งานพื้นฐานเหมือนกัน ดังแสดงในรูปที่ 2.3 และ 2.4 โดยมีรายละเอียดขั้นต้น ดังนี้

ขา Vcc ใช้สำหรับต่อไฟเลี้ยง +5V

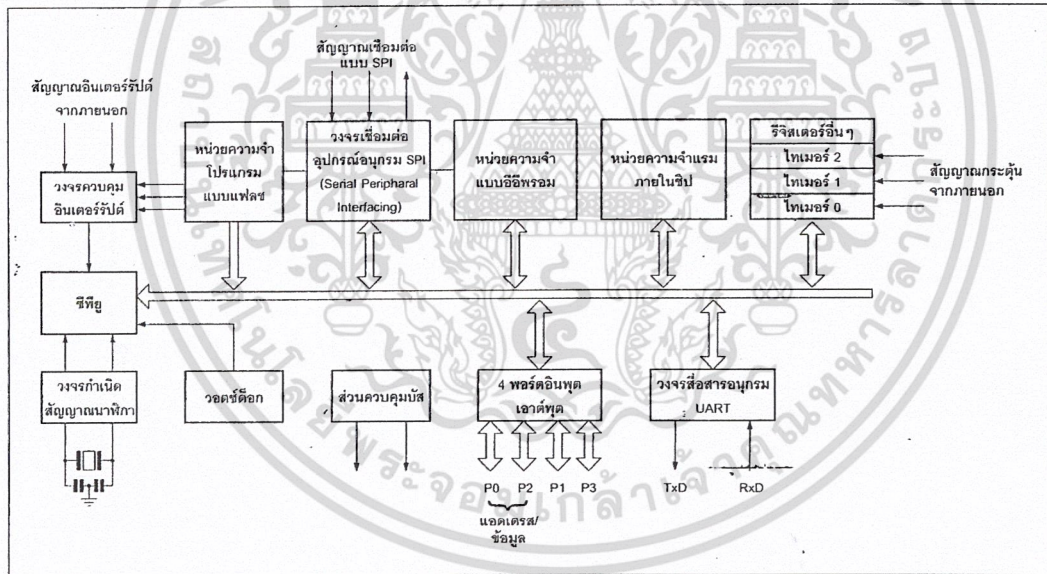
ขา GND เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

ขาพอร์ต 0 (P0-P7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุตสำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (float) จึงมีอินพุตอิมพีแดนซ์สูงสามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนี้ขาพอร์ตนี้ยังถูกใช้ในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานเป็นได้ทั้งขาติดต่อกับแอดเดรสและขาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชอนุกรม AT89Cxx



รูปที่ 2.2 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชอนุกรม AT89Sxx

เบอร์ของไมโครคอนโทรลเลอร์	หน่วยความจำโปรแกรม	หน่วยความจำข้อมูล	จำนวนไทเมอร์/เคาน์เตอร์ 16 บิต
AT89C1051	แบบแฟลช ขนาด 1 กิโลไบต์	แรม 64 ไบต์	1
AT89C2051	แบบแฟลช ขนาด 2 กิโลไบต์	แรม 128 ไบต์	2
AT89C51	แบบแฟลช ขนาด 4 กิโลไบต์	แรม 128 ไบต์	2
AT89C52	แบบแฟลช ขนาด 8 กิโลไบต์	แรม 256 ไบต์	3
AT89C55	แบบแฟลช ขนาด 20 กิโลไบต์	แรม 256 ไบต์	3
AT89S8252	แบบแฟลช ขนาด 8 กิโลไบต์	แรม 256 ไบต์	3
		อีพรอม 2 กิโลไบต์	
AT89S53	แบบแฟลช ขนาด 12 กิโลไบต์	แรม 256 ไบต์	3

ตารางที่ 2.1 รายละเอียดโดยสรุปบางส่วนของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชที่ Atmel ผลิตขึ้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุต สามารถทำได้โดยการเขียน ข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วยนอกจากนั้นในอนุกรม AT89sxx จะใช้ขา P1.0 เป็นขาอินพุตสำหรับนับค่าของไทมเมอร์ 2 และ P1.1 เป็นขาอินพุตทริกเกอร์ของไทมเมอร์ 2 ในขณะที่ขา P1.4 ถึง P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

ขาพอร์ต (P2.0-P2.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำได้โดยการเขียน ข้อมูล "1" ไปยังแต่ละบิตของพอร์ต ทดลองการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (float) จึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนั้นขาพอร์ตนี้ยังถูก ใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)

ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำได้โดยการเขียน ข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อย ลอย (float) จึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนั้นขา พอร์ต 3 ยังเป็นขาที่มีหน้าที่การใช้งานพิเศษ ดังมีรายละเอียดขั้นต้นต่อไปนี้

P3.0 ใช้เป็นขาอินพุตสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา RxD

P3.1 ใช้เป็นขาอินพุตสำหรับส่งข้อมูลจากการสื่อสารแบบอนุกรม หรือขา TxD

P3.2 ใช้เป็นขาอินพุตรับสัญญาณอินเทอร์รัปต์จากภายนอกช่อง 0 หรือขา ITO

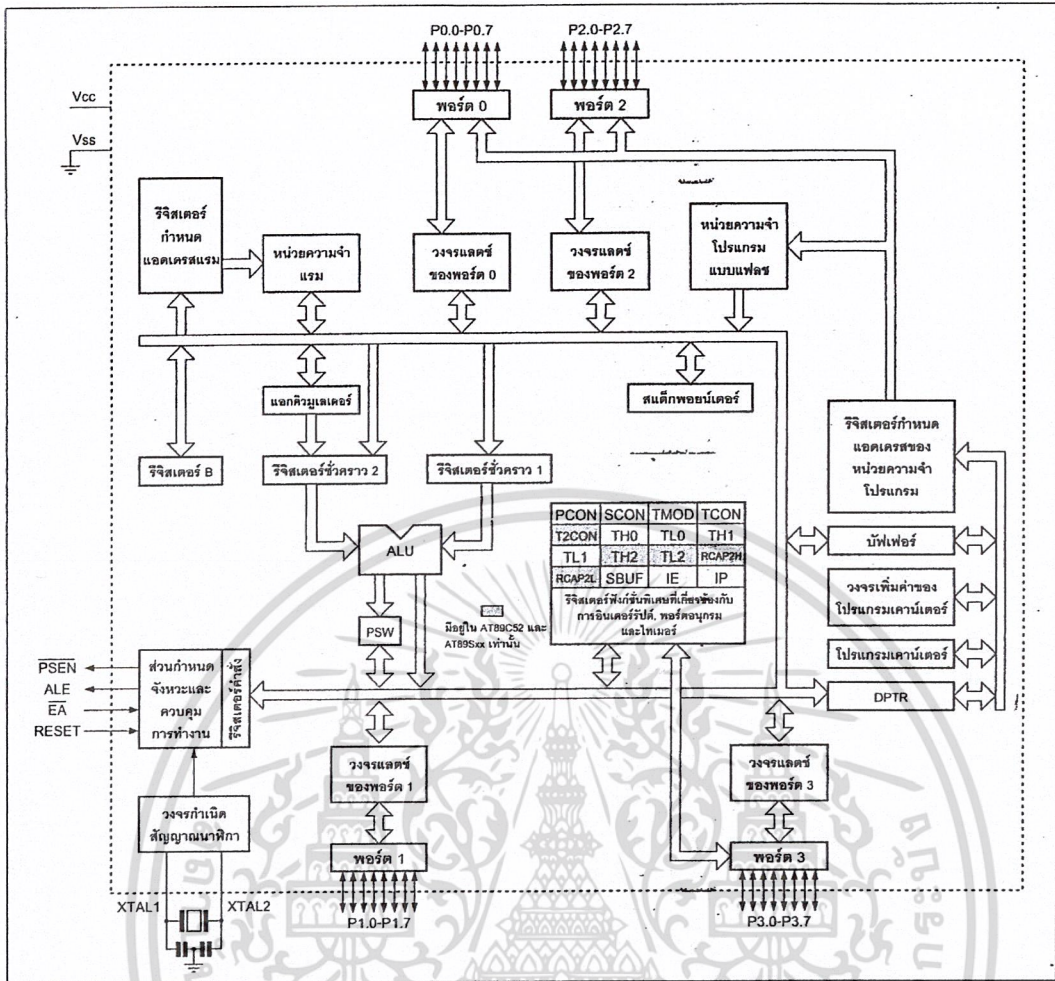
P3.3 ใช้เป็นขาอินพุตรับสัญญาณอินเทอร์รัปต์จากภายนอกช่อง 1 หรือขา IT1

P3.4 ใช้เป็นขาอินพุตสำหรับรับสัญญาณไทมเมอร์จากภายนอกช่อง 0 หรือขา T0

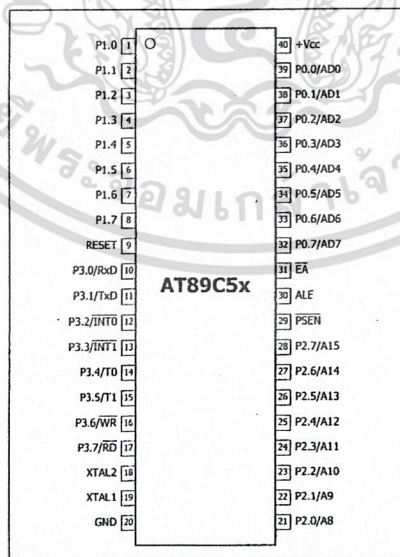
P3.5 ใช้เป็นขาอินพุตสำหรับรับสัญญาณอินเทอร์รัปต์จากภายนอกช่อง 1 หรือขา T1

P3.6 ใช้เป็นขาสัญญาณ WR ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

P3.7 ใช้เป็นขาสัญญาณ RD ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก



รูปที่ 2.3 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชของ Atmel



รูปที่ 2.4 การจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51

ขารีสต (Reset) ใช้ในการรีเซ็ตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณเพื่อรีเซ็ตสถานะที่ขานี้ต้องอยู่ในระดับรีเซ็ตอย่างน้อย 2 แมกซีนไซเคิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ

ขา  $\overline{ALE}$  / PROG (Address Latch Enable/Program pulse input) เป็นขาที่ใช้ในการควบคุมการแลตช์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนั้นขานี้ยังใช้เป็นขาสำหรับรับพัลส์ของการโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MCS-51 ในรุ่นที่มีหน่วยความจำโปรแกรมเป็นแบบอีพรอม

ขา  $\overline{PSEN}$  (Program Store Enable) ขานี้ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอกตัวไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก ตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกมาที่ขา  $\overline{PSEN}$  2 ครั้งในแต่ละแมกซีนไซเคิล แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใด ๆ ออกมา

ขา  $\overline{EA}$  / Vpp (External Access enable/Programming Voltage Input) ใช้สำหรับเลือกการติดต่อกับหน่วยความจำโปรแกรมจากภายนอกหรือภายในตัวไมโครคอนโทรลเลอร์ ถ้าหากขานี้เป็น "0" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายนอก แต่ถ้าหากขานี้เป็น "1" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ นอกจากนี้ที่ขา  $\overline{EA}$  นี้ยังใช้เป็นขาอินพุตสำหรับรับแรงดันไฟสูงสำหรับการโปรแกรมหน่วยความจำภายในไมโครคอนโทรลเลอร์ สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชต้องการแรงดันสำหรับการโปรแกรม +12V

ขา XTAL1 และ XTAL2 เป็นขาสำหรับต่อคริสตัลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

### โครงสร้างและการทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ตคือ พอร์ต 0 ถึง พอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุตสำหรับสัญญาณข้อมูลเข้า และ เอาต์พุตสำหรับส่งสัญญาณข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีวงแลตช์และวงจรขับตลอดจนบัฟเฟอร์อินพุต ดังแสดงให้เห็นในสถาปัตยกรรมรูปที่ 2.3

ที่พอร์ต 0 และพอร์ต 2 จะใช้งานเป็นพอร์ตอินพุตและเอาต์พุตสำหรับงานทั่วไป และในการติดต่อกับหน่วยความจำภายนอกสำหรับพอร์ต 3 ทั้งพอร์ตและพอร์ต 1 บางขานอกจากจะใช้เป็นขาพอร์ตอินพุตเอาต์พุตตามปกติแล้ว ยังสามารถใช้งานในหน้าที่พิเศษได้อีก ขึ้นอยู่กับว่าเป็นไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์

ใด ดังสรุปได้ในตารางที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 2.5 แสดงวงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดยในรูปที่ 2.5 (ก) เป็นวงจรของพอร์ต 0 วงจรแลตซ์ของแต่ละบิตในแต่ละพอร์ตก็คือวงจรดีฟลิปฟล็อปนั่นเอง การอ่านค่าสถานะของพอร์ตและสถานะของวงจรแลตซ์สามารถกระทำได้อย่างอิสระด้วยสัญญาณที่แยกจากกัน นั่นคือสัญญาณอ่านข้อมูลจากขาพอร์ต และสัญญาณอ่านข้อมูลจากวงจรแลตซ์ ส่วนการเขียนข้อมูลมายังพอร์ตต้องส่งสัญญาณมายังขาCLKของดีฟลิปฟล็อปในขณะที่ข้อมูลจะผ่านมายังขาบั๊ตข้อมูลภายในเข้าสู่ขา D ของดีฟลิปฟล็อป

ที่พอร์ตนี้มีวงจรมัลติเพล็กซ์สำหรับกำหนดลักษณะการทำงานของพอร์ตว่า ต้องการใช้งานเป็นขาพอร์ตอินพุตเอาต์พุตปกติหรือใช้ในการติดต่อกับหน่วยความจำภายนอก ไมโครคอนโทรลเลอร์

เนื่องจากที่ขาพอร์ต 0 ไม่มีวงจรพูลอัพภายใน หากมีการนำพอร์ต 0 ไปใช้งานเป็นพอร์ตอินพุตจะต้องต่อตัวต้านทานพูลอัพภายนอกเข้าที่ขาพอร์ต 0 ทุกขาด้วย

ในรูปที่ 2.5 (ข) เป็นวงจรของพอร์ต 1 ซึ่งมีลักษณะโดยทั่วไปคล้ายกับพอร์ต 0 หากแต่ไม่มีวงจรมัลติเพล็กซ์ เนื่องจากพอร์ตนี้จะไม่ใช้ในการติดต่อกับหน่วยความจำภายนอก แต่จะมีวงจรพูลอัพภายในที่แต่ละบิตของพอร์ตนี้แทน สำหรับรายละเอียดของวงจรพูลอัพแสดงในรูปที่ 2.6

ในรูปที่ 2.5 (ค) เป็นวงจรภายในของพอร์ต 2 จะคล้ายกับพอร์ต 0 มาก ต่างกันเพียงมีวงจรพูลอัพเพิ่มเติมเข้ามา ส่วนในรูปที่ 2.5 (ง) เป็นวงจรภายในของพอร์ต 3 จะเห็นได้ว่าคล้ายกับพอร์ต 1 มีการเพิ่มเติมวงจรบั๊ฟเฟอร์และวงจรอินพุตเอาต์พุตเมื่อทำงานในฟังก์ชันพิเศษเข้ามา เนื่องจากพอร์ต 3 สามารถนำไปใช้งานในหน้าที่พิเศษได้ทุกขา

### การใช้งานเป็นพอร์ตอินพุต

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถเป็นได้ทั้งอินพุตและเอาต์พุต ดังนั้นจึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในการกำหนดให้เป็นพอร์ตอินพุต ต้องเริ่มต้นด้วยการเขียนข้อมูล "1" มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาต์พุตของบิตนั้น ๆ ทำให้ขาสัญญาณของพอร์ตเชื่อมต่อเข้ากับวงจรพูลอัพภายในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น "1" สามารถรับสัญญาณลอจิก "0" จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะจกส่งเข้ามาแล้วเก็บไว้ในวงจรบั๊ฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านค่าเข้าไป เมื่อเป็นเช่นนั้น อุปกรณ์ภายนอกที่เชื่อมต่อกับพอร์ตอินพุตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชควรกำหนดให้ทำงานในสภาวะลอจิก "0" จะดีและสะดวกที่สุด (ซึ่งในปัจจุบันอุปกรณ์อินพุตที่เชื่อมต่อ ไมโครคอนโทรลเลอร์แทบทั้งหมดทำงานที่ลอจิก "0" แล้ว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

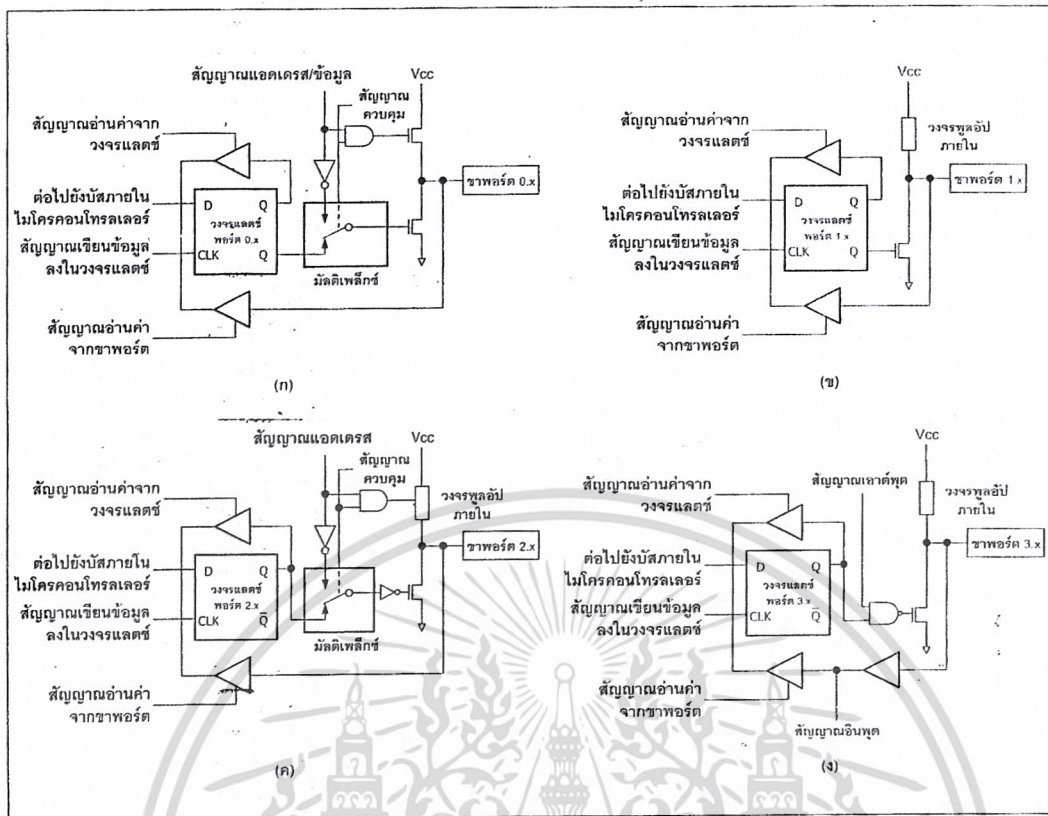
ขา	เบอร์ของ ไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P1.0	AT89C52/AT89Sxx	ขา T2 เป็นขาอินพุตระดับค่าของโทเมอร์/เคาน์เตอร์ 2 และเป็นขา
P1.1	AT89C52/AT89Sxx	และควบคุมทิศทางของสัญญาณ
P1.4	AT89Sxx	ขา SS (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไม่โครคอน- โทรลเลอร์เป็นอุปกรณ์สเลฟ ในระบบการติดต่อแบบ SPI
P1.5	AT89Sxx	ขา MOSI (Master data output, Slave data input) ใช้ในการติดต่อกับพอร์ต SPI
P1.6	AT89Sxx	ขา MISO (Master data input, Slave data output) ใช้ในการติดต่อกับพอร์ต SPI
P1.7	AT89Sxx	ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

ตารางที่ 2.2 หน้าที่พิเศษของพอร์ต 1 ใน ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชของ Atmel

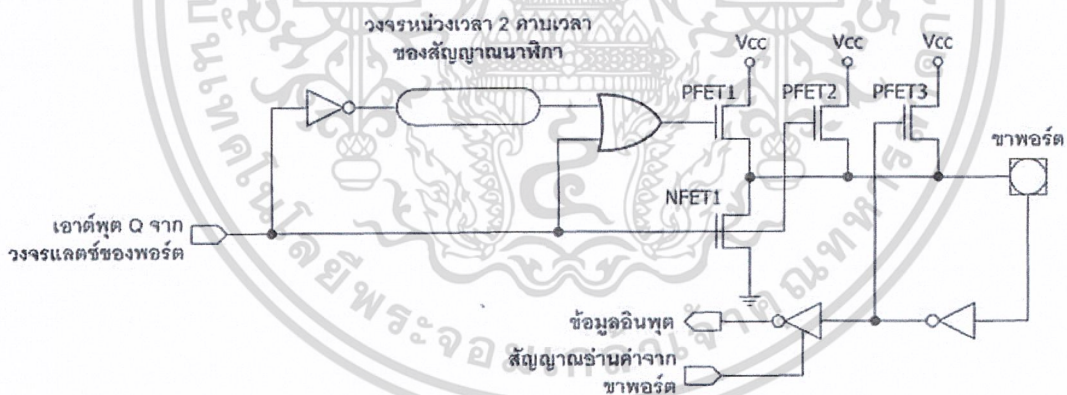
### การใช้งานเป็นพอร์ตเอาต์พุต

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาต์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไป ตรงมา กล่าวคือ เมื่อต้องการส่งข้อมูล "0" ออกไปทางเอาต์พุตก็ให้เขียนข้อมูล "0" ไปยังวงจรถ่ายแลตซ์ ซึ่งก็จะส่งต่อไปขับเฟด ทำให้เฟดทำงาน ที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก "0" ขึ้น ในทางตรงข้ามหากต้องการส่งข้อมูล "1" ออกไป ก็ให้เขียนข้อมูล "1" ไปยังวงจรถ่ายแลตซ์ วงจรขับก็จะหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรถ่ายแลตซ์เกิดเป็นลอจิก "1" ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุตจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์ แต่ถ้าเป็นเอาต์พุตจะไม่มีกรอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 วงจรภายในของพอร์ตในการใช้งานเป็นพอร์ตทุกพอร์ตใน ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช



รูปที่ 2.6 วงจรพูลอัพภายในพอร์ตของ ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เมื่อใช้งานเป็นพอร์ตเอาต์พุต แต่ละขา (หรือแต่ละบิต) ของแต่ละพอร์ตมีความสามารถในการจ่ายกระแสหรือที่ เรียกว่า กระแสซอร์ส (Source current) ได้สูงสุด 10 mA และทกขารวมกันในแต่ละพอร์ต (ทั้ง 8 บิต) สูงสุด 26 mA สำหรับพอร์ต 0 และ 15 mA สำหรับพอร์ต 1 ในกรณีที่ใช้งานทุกพอร์ตเอาต์พุตจะสามารถจ่ายกระแสได้รวมกันสูงสุด 71 mA ดังนั้นในการใช้งานเป็นพอร์ตเอาต์พุตเพื่อไม่ให้เกิดปัญหาเกี่ยวกับความสามารถในการจ่ายกระแสจึงควรต่อวงจรบัฟเฟอร์ทางเอาต์พุตเพื่อช่วยในการขับกระแสอีกทางหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การอ่านค่าลอจิกจากพอร์ต

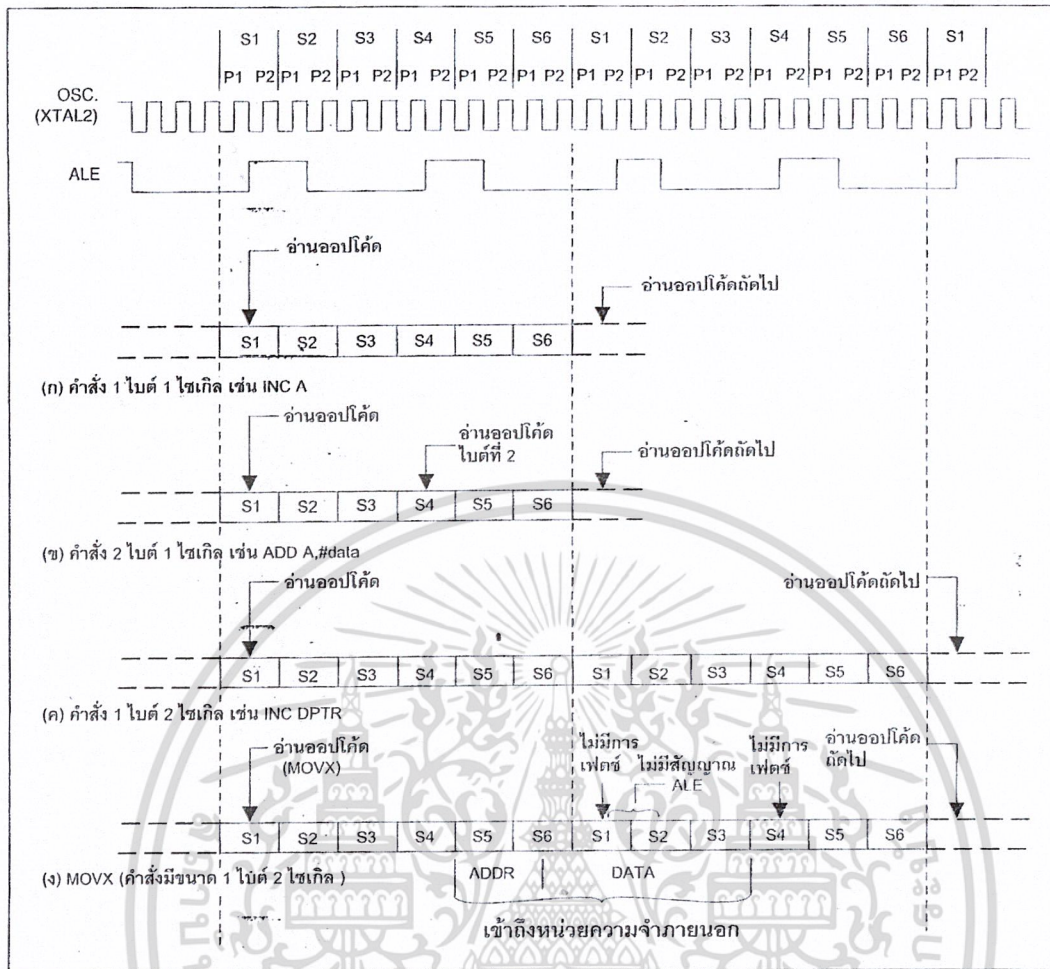
ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถอ่านค่าลอจิกจากพอร์ตได้ 2 ลักษณะคือ อ่านจากขาพอร์ตโดยตรง และอ่านจากวงจรแลตช์ของแต่ละพอร์ต

ในกรณีที่พอร์ตต่อกับขาเบสทรานซิสเตอร์ชนิด NPN และขาอิมิตเตอร์ของทรานซิสเตอร์ตัวนั้นต่อลงกราวด์ หากมีการส่งข้อมูล "1" ไปยังทรานซิสเตอร์ จะทำให้ทรานซิสเตอร์ทำงานสถานะลอจิกที่ขาพอร์ตจะเป็น 0 เนื่องจากเมื่อทรานซิสเตอร์ทำงาน จะเสมือนว่าขาพอร์ตนั้นต่อลงกราวด์ ทำให้หากอ่านค่าลอจิกที่ขาพอร์ต 1 จะได้ผลตรงข้ามกับที่ 0 ส่งออกมา แต่ถ้าหากทำงานอ่านค่าลอจิกที่วงจรแลตช์ จะได้ค่าที่ตรงกับค่าที่ต้องการส่งจริง ดังนั้น ในการอ่านค่าลอจิกจากพอร์ต จึงต้องเลือกวิธีการให้เหมาะสมกับอุปกรณ์ที่นำมาต่อดัวย

## จังหวะการทำงานของไมโครคอนโทรลเลอร์ MCS-51

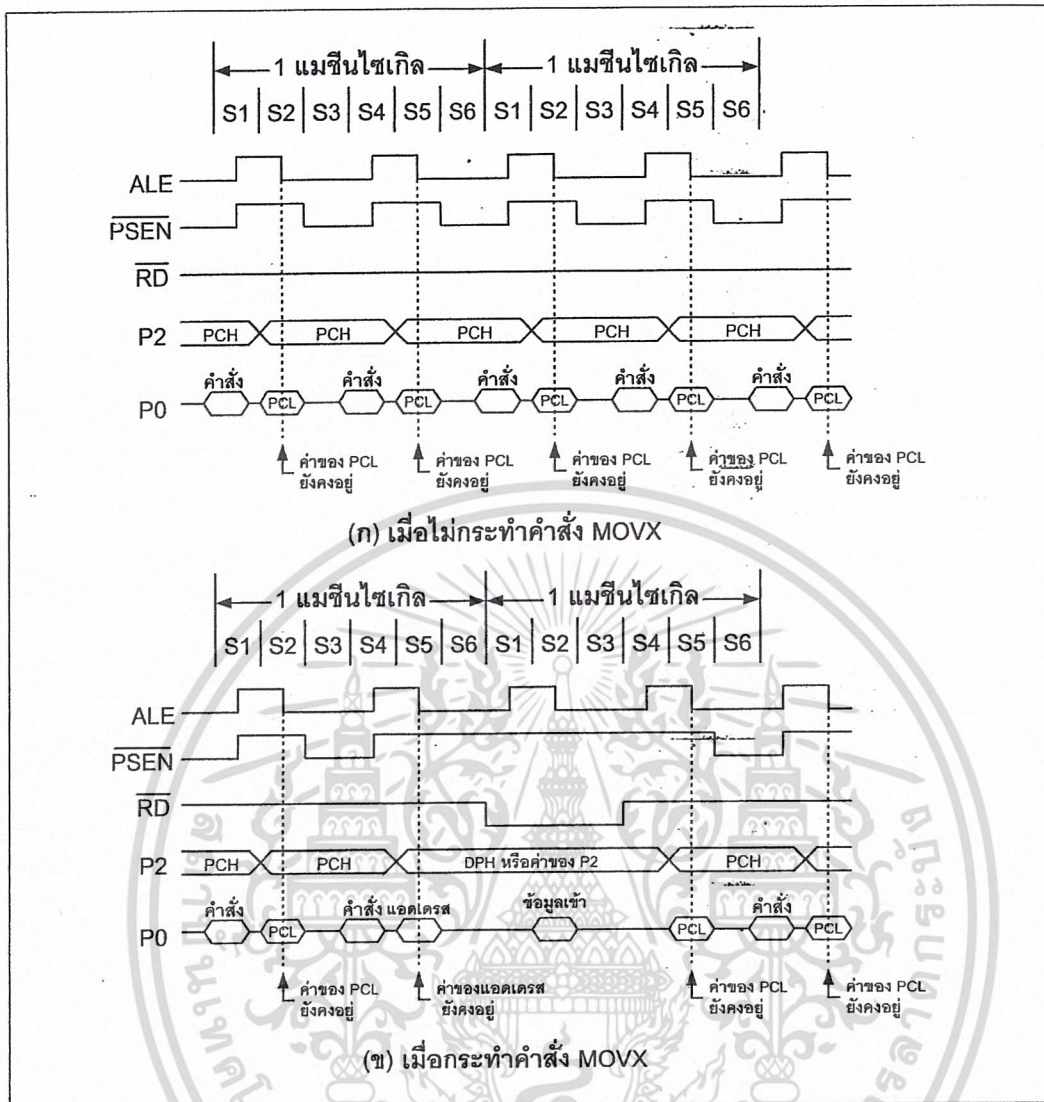
ในการใช้งานไมโครคอนโทรลเลอร์ MCS-51 จะต้องทำความเข้าใจถึงจังหวะการทำงานของซีพียู และลำดับขั้นตอนการประมวลผลคำสั่ง ในการประมวลผลคำสั่งของซีพียูจะมีขั้นตอนหลักๆ 2 ขั้นตอนคือ กระบวนการเฟตช์ (fetch) เป็นการเรียกคำสั่งออกจากหน่วยความจำโปรแกรมแล้วทำการแปลรหัสคำสั่งนั้นเป็นภาษาเครื่องเพื่อเตรียมการประมวลผล ขั้นตอนต่อมาคือกระบวนการเอกซิกิวต์ (execute) เป็นการกระทำตามคำสั่งที่กำหนด หรือตามที่เฟตช์ขึ้นมาโดยกระบวนการก่อนหน้าเมื่อทำการเอกซิกิวต์คำสั่งเรียบร้อยแล้ว ก็จะไปเริ่มกระบวนการเฟตช์คำสั่งใหม่ต่อไป

เมื่อเริ่มจ่ายไฟให้แก่ไมโครคอนโทรลเลอร์จะเกิดการรีเซ็ตในลักษณะที่เรียกว่า เพาเวอร์ออร์อนรีเซ็ต (Power On reset) ซีพียูเริ่มดำเนินการทำงานที่แอดเดรส 0000H ของหน่วยความจำโปรแกรม จังหวะการทำงานของซีพียูจะเป็นไปตามรูปแบบ โดยได้รับการกำหนดมาจากรอบการทำงานหรือแมชีนไซเคิล (machine cycle) ในรูปที่ 2.7 เป็นไคอะแกรมเวลาแสดงจังหวะการทำงานของไมโครคอนโทรลเลอร์ MCS-51 โดยใน 1 รอบการทำงานหรือแมชีนไซเคิลจะแบ่งย่อยออกเป็น 6 สเตต (state) กำหนดชื่อเป็น S1-S6 ในแต่ละสเตตมีค่าเวลาเท่ากับ 2 คาบเวลาของสัญญาณนาฬิกา ถ้าสัญญาณนาฬิกามีความถี่ 12 MHz จะมีคาบเวลาเท่ากับ 1 ms คาบเวลาทั้งสองภายในหนึ่งสเตตจะเรียกว่า เฟส 1 (Phase 1) และเฟส 2 (phase 2)



รูปที่ 2.7 ไชเกิลการทำงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในรูปที่ 2.7 (ก) และ (ข) จะเป็นการเอ็ชคิวต์คำสั่งที่ใช้เวลา 1 ไชเกิล เริ่มต้นที่สแตต 1 จะเป็นการอ่านค่าออปโค้ด อันเป็นกระบวนการแลตซ์ค่าของออปโค้ดส่งไปให้รีจิสเตอร์คำสั่ง (IR) การเฟลชครั้งที่สองจะเกิดขึ้นที่สแตต 4 ภายในเมซึนไชเกิลเดียวกัน ในกรณีที่เป็นคำสั่งไบต์เดียว การเฟลชครั้งที่ 1 2 ภายในเมซึนไชเกิลเดียวกันจะจกตัดทิ้งไป ในคำสั่งที่มีใช้เวลา 1 ไชเกิล จะสิ้นสุดการทำงานลงในสแตต 6 ของเมซึนไชเกิลเดียวกัน



รูปที่ 2.8 ไตอะแกรมเวลาแสดงการติดต่อและเข้าถึงหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในกรณีที่คำสั่งใช้เวลา 2 ไชเกิด การทำงานของคำสั่งนั้นจะสิ้นสุดลงในสแตต 6 ของแมซซึนไซเกิดที่สองดังในไตอะแกรมรูปที่ 2.7 (ค) สำหรับในการกระทำคำสั่ง MOVX ซึ่งเป็นคำสั่งขนาด 1 ไบต์ 2 ไชเกิด จะไม่มีการเฟลชเกิดขึ้นในไชเกิดที่สองของคำสั่ง MOVX นี้เนื่องจากซีพียูจะไปทำการติดต่อกับหน่วยความจำภายนอกดังแสดงในไตอะแกรมรูปที่ 2.7 (ง) จะเห็นได้ว่าเวลาในการเอ็คซีคิวต์จะไม่ได้ขึ้นอยู่กับว่าทำการติดต่อหน่วยความจำโปรแกรมภายในหรือภายนอก

ในรูปที่ 2.8 แสดงสัญญาณและไตอะแกรมเวลาของการเข้าถึงหน่วยความจำโปรแกรมภายนอกโดยในรูปที่ 2.8 (ก) เป็นไตอะแกรมเวลาในขณะที่ยังไม่มีการทำงานคำสั่ง MOVX สัญญาณที่ขา ALE และ PSEN จะเกิดการ แอกตีฟ 2 ครั้งภายในหนึ่งแมซซึนไซเกิด ในทุกครั้งที่ ALE เกิดการแอกตีฟที่พอร์ต 0 (P0) จะมีค่าของรีจิสเตอร์ PC ในไบต์ต่ำออกมา ในขณะที่พอร์ต 2 (P2) ก็จะมีค่าของ PC ในไบต์สูงเพื่อซีไปยังแอดเดรสต่อไปที่ต้อไปดำเนินการ สำหรับขา PSEN ก็จะมีเอกซารนี้เป็นเอกซารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกซารทุกครั้งที่มีการนำไปใช้

การแอกตีฟเมื่อมีการติดต่อกับหน่วยความจำโปรแกรมภายนอก ในกรณีที่กระทำคำสั่ง MOVX เพื่อเข้าถึงหน่วยความจำข้อมูลภายนอก ที่ขา PSEN จะไม่เกิดการแอกตีฟ 2 ครั้งภายใน 1 แมกซีนไซเคิล เนื่องจากบัสแอกเดรสและบัสข้อมูลจะถูกใช้ในการติดต่อกับหน่วยความจำข้อมูลภายนอกแทน แต่สำหรับสัญญาณ ALE ยังคงแอกตีฟตามจังหวะการทำงานเหมือนเดิม

จากไดอะแกรมเวลาสามารถสรุปได้ว่า ในการทำงาน 1 รอบหรือ 1 แมกซีนไซเคิล ซีพียูในไมโครคอนโทรลเลอร์ MCS-51 จะใช้เวลา 12 คาบเวลาของสัญญาณนาฬิกา นั่นคือ เวลาในการทำงาน 1 ไซเคิลมีค่าเท่ากับ 1 หรือมีความเร็วในการทำงานภายใน 1 ms ในกรณีที่ใช้ความถี่สัญญาณนาฬิกา 12 MHZ ดังนั้นถ้าต้องการทราบความเร็วของการทำงานภายในของไมโครคอนโทรลเลอร์ MCS-51 สามารถหาได้จาก ค่าความถี่สัญญาณนาฬิกาหารด้วย 12 และถ้าต้องการหาค่าเวลาของ 1 รอบการทำงานหรือ 1 แมกซีนไซเคิล สามารถทำได้โดยการหาส่วนกลับของความเร็ว ในการทำงานภายในของไมโครคอนโทรลเลอร์ MCS-51 สามารถสรุปเป็นสูตรทางคณิตศาสตร์ได้ดังนี้

$$\begin{aligned} & \text{ความเร็วในการทำงานภายในของไมโครคอนโทรลเลอร์เท่ากับ} \\ & \text{ความถี่ของสัญญาณนาฬิกา (ค่าของคริสตอลที่ต่ออยู่ที่ขา XTAL 1 และ XTAL 2) / 12} \\ & \text{เวลา 1 แมกซีนไซเคิล} = 1 / \text{ความเร็วในการทำงานภายในของไมโครคอนโทรลเลอร์} \end{aligned}$$

### การจัดหน่วยความจำของ ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีหน่วยความจำภายในหลักๆ อยู่ 2 ส่วนคือ หน่วยความจำโปรแกรมและหน่วยความจำข้อมูล ซึ่งก็มีขนาดและการจัดสรรแตกต่างกันไปในแต่ละเบอร์ ในบทนี้จะกล่าวถึงรายละเอียดของการจัดสรรหน่วยความจำภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช การเชื่อมต่อกับหน่วยความจำภายนอก และข้อมูลเบื้องต้นของรีจิสเตอร์ฟังก์ชันพิเศษที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

### หน่วยความจำโปรแกรม (Program memory)

ในรูปที่ 2.8 แสดงการจัดหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในเบอร์ต่างๆ ที่นิยมใช้งานอันประกอบด้วยเบอร์ AT89C51 และ AT89C52 จะเห็นได้ว่าทั้งสองเบอร์สามารถติดต่อกับหน่วยความจำโปรแกรมได้สูงสุด 64 กิโลไบต์ โดยสามารถเลือกใช้หน่วยความจำโปรแกรมภายในอย่างเดียวหรือรวมกับภายนอกหรือเลือกใช้หน่วยความจำภายนอกอย่างเดียวก็ได้ ดังในรูปที่ 2.8 (ก) โดยภายใน AT89C51 จะมีหน่วยความจำโปรแกรมภายใน 4 กิโลไบต์ ในขณะที่ AT89C52 จะมีขนาด 8 กิโลไบต์

ในกรณีที่ใช้หน่วยความจำภายในและภายนอกรวมกัน หากใช้ AT89C51 ก็จะสามารถติดต่อกับหน่วยความจำภายนอกได้ 60 กิโลไบต์ และถ้าใช้เบอร์ AT89C52 จะสามารถติดต่อกับหน่วยความจำโปรแกรมภายนอกได้ 56 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำโปรแกรมใช้เก็บข้อมูลของโปรแกรมควบคุมการทำงานของ ไมโครคอนโทรลเลอร์หรือที่เรียกว่า โปรแกรมมอนิเตอร์ (Monitor program) หากใช้หน่วยความจำภายนอกมักจะบรรจุอยู่ในหน่วยความจำชนิดอีพรอม (EPROM: Erasable Programmable Read Only Memory) ซึ่งสามารถทำการอ่านได้เพียงอย่างเดียว

หน่วยความจำโปรแกรมมีแอดเดรสเริ่มต้นที่ 0000H เมื่อซีพียูได้รับการรีเซ็ตให้เริ่มต้นการทำงาน จะต้องมาเริ่มต้นที่ แอดเดรส 0000H นี้เสมอ อย่างไรก็ตาม ในพื้นที่ของหน่วยความจำโปรแกรมไม่ว่าจะใช้งานจากภายในหรือภายนอกก็ตาม ต้องมีการสงวนพื้นที่บางตำแหน่งเอาไว้สำหรับการบริการอินเตอร์รัปต์ 6 ประเภท ประเภทละ 8 ไบต์ ประกอบด้วย

พื้นที่สำหรับบริการอินเตอร์รัปต์ 0 จากภายนอก กำหนดไว้ที่แอดเดรส 0003H

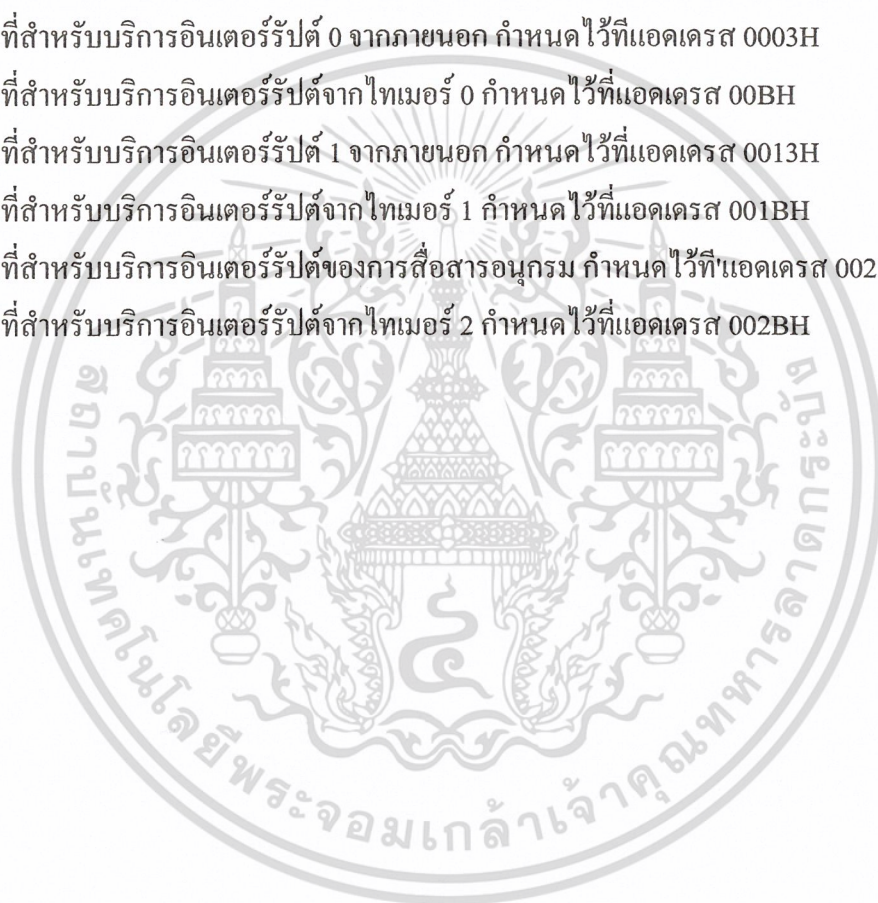
พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทมเมอร์ 0 กำหนดไว้ที่แอดเดรส 000BH

พื้นที่สำหรับบริการอินเตอร์รัปต์ 1 จากภายนอก กำหนดไว้ที่แอดเดรส 0013H

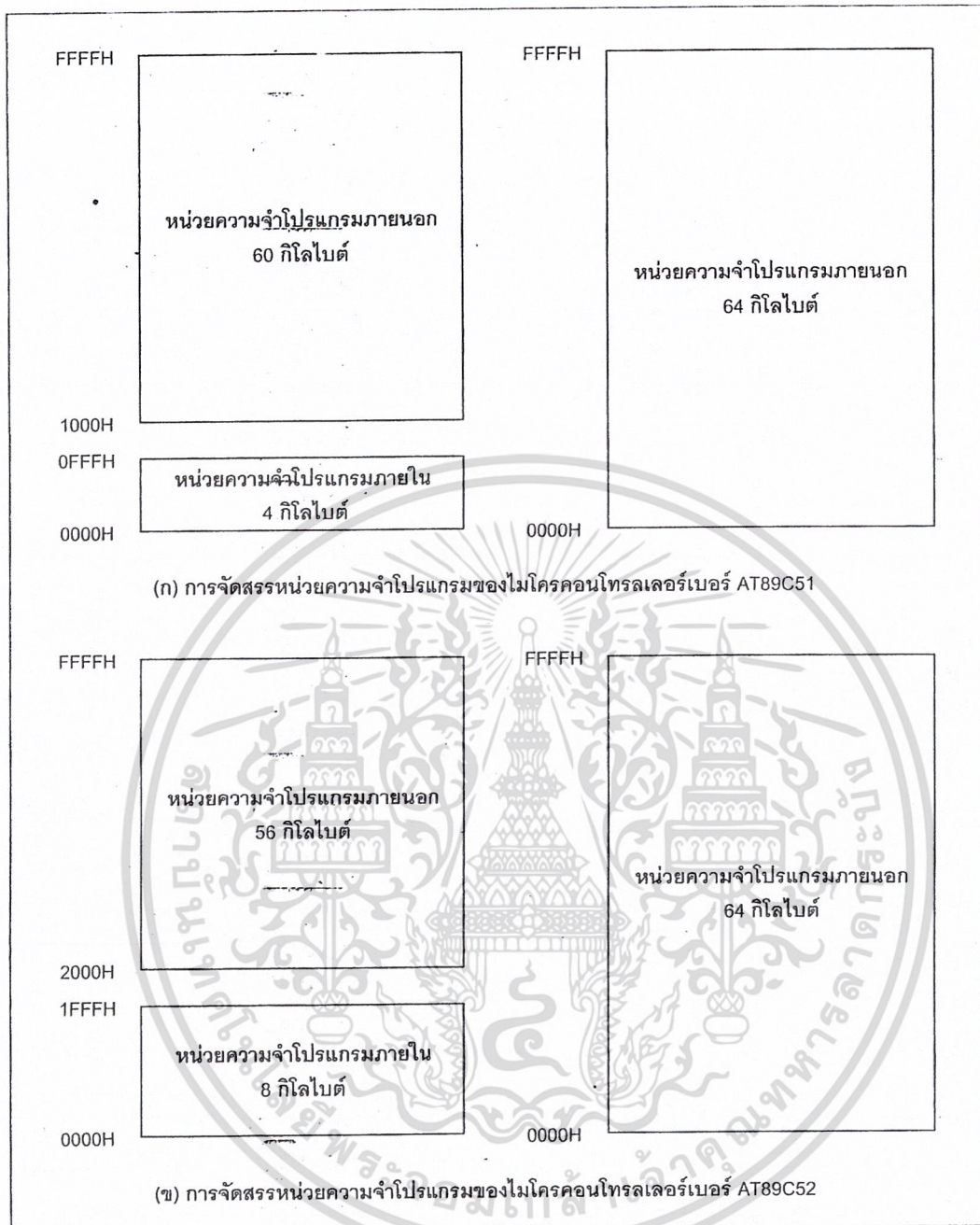
พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทมเมอร์ 1 กำหนดไว้ที่แอดเดรส 001BH

พื้นที่สำหรับบริการอินเตอร์รัปต์ของการสื่อสารอนุกรม กำหนดไว้ที่แอดเดรส 0023H

พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทมเมอร์ 2 กำหนดไว้ที่แอดเดรส 002BH



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 การจัดหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

กรณีที่ใช้ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชที่มีหน่วยความจำโปรแกรมภายใน แต่ต้องการติดต่อกับหน่วยความจำโปรแกรมภายนอกด้วย สามารถทำได้โดยต้องกำหนดแอดเดรสของหน่วยความจำโปรแกรมให้ต่อจากแอดเดรสสุดท้ายของหน่วยความจำโปรแกรมภายในของไมโครคอนโทรลเลอร์ ยกตัวอย่าง ไมโครคอนโทรลเลอร์ AT89C51 มีหน่วยความจำโปรแกรมขนาด 4 กิโลไบต์ มีแอดเดรสอยู่ระหว่าง 0000H-0FFFH เมื่อต่อหน่วยความจำโปรแกรมภายนอก ต้องกำหนดให้แอดเดรสอยู่ในช่วง 1000H-FFFFH

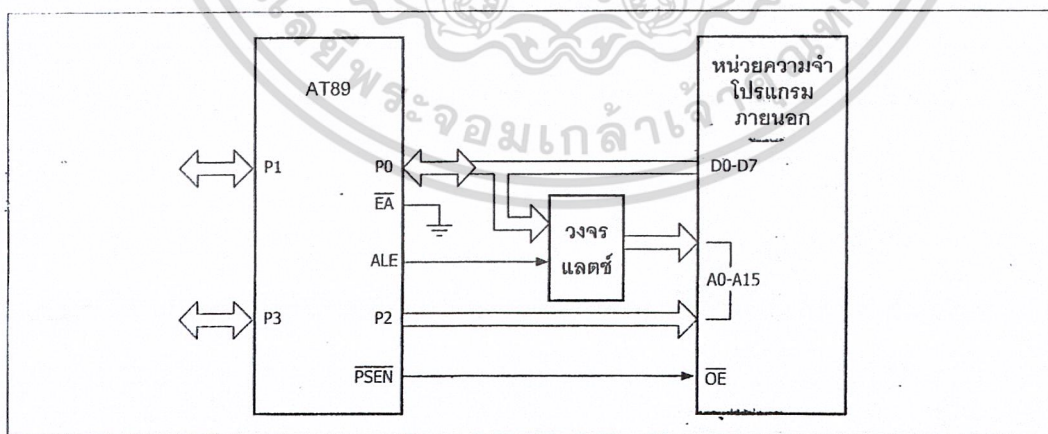
การต่อหน่วยความจำภายนอกแสดงดังในรูปที่ 2.9 ขาพอร์ต P0.0-P0.7 ใช้เป็นขาข้อมูล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D0-D7 และขาแอดเดรสไบต์ต่ำ โดยผ่านวงจรถัด ซึ่งปกติใช้ไอซีเบอร์ 74HC573 และใช้สัญญาณ ALE และ PSEN ในการเลือกใช้งานขา P0-P7 เพื่อเป็นขาข้อมูลหรือขาแอดเดรส ในขณะที่ขา P2.0-P2.7 ใช้ในการเชื่อมต่อกับขาแอดเดรสไบต์สูง A8-A15 ดังนั้นเมื่อมีการติดต่อกับหน่วยความจำ โปรแกรมภายนอกไมโครคอนโทรลเลอร์จะเหลือขาพอร์ตเพียง 16 บิต คือ ที่ขาพอร์ต P1.0-P1.7 และ P3.0-P3.7

**หน่วยความจำข้อมูล (Data memory)**

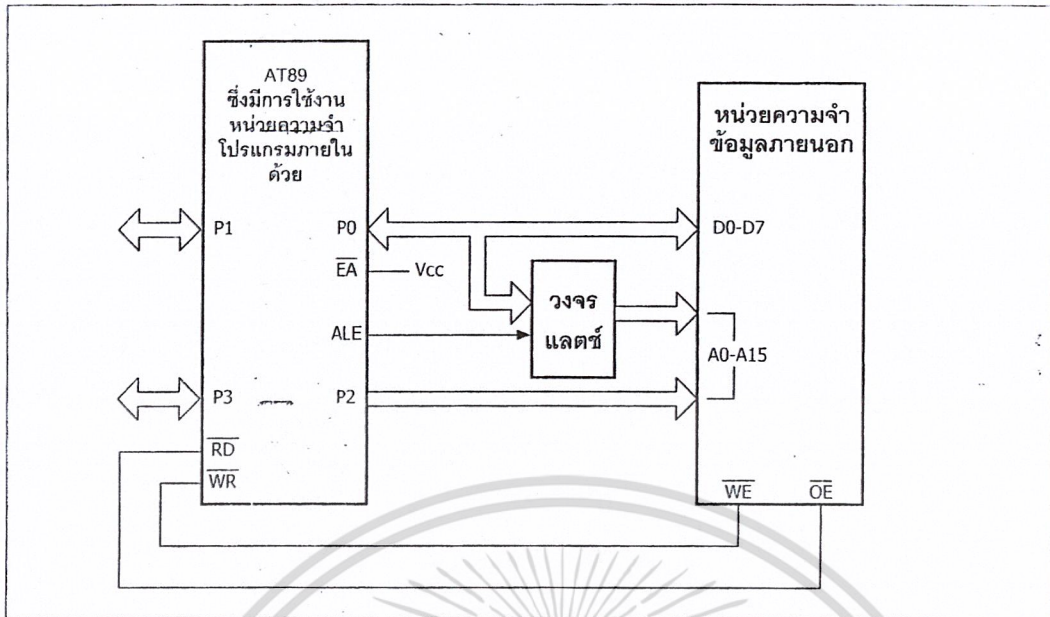
มีด้วยกัน 2 แบบคือ หน่วยความจำข้อมูลภายนอกและภายใน โดยไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 สามารถติดต่อกับหน่วยความจำข้อมูลภายนอกได้สูงสุด 64 กิโลไบต์ โดยการใช้คำสั่ง MOVX ในการติดต่อกับหน่วยความจำข้อมูลภายนอก การติดต่อกับหน่วยความจำข้อมูลภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชแสดงดังในรูปที่ 2.10 จะเห็นได้ว่า มีลักษณะคล้ายกับการติดต่อกับหน่วยความจำ โปรแกรมภายนอก แตกต่างกันที่มีสัญญาณที่ใช้สำหรับการอ่านและเขียนหน่วยความจำข้อมูลภายนอก นั่นคือ ขา RD และขา WR

สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 ทุกเบอร์จะมีหน่วยความจำข้อมูลภายในเป็นแบบแรม (RAM: Random Access Memory) โดยแต่ละเบอร์จะมีขนาดแตกต่างกันไป ในเบอร์ AT89C51 มีหน่วยความจำข้อมูลภายในขนาด 128 ไบต์ ในขณะที่เบอร์ AT89C52 มีขนาด 256 ไบต์ สำหรับการจัดสรรหน่วยความจำข้อมูลภายในแบ่งเป็น 3 ส่วนคือ หน่วยความจำข้อมูลส่วนล่าง (Lower) ส่วนบน (Upper) และรีจิสเตอร์ฟังก์ชันพิเศษ (SFR: Special Function Register) แต่ละส่วนมีขนาด 128 ไบต์ ดังแสดงการจัดสรรในรูปที่ 2.11



รูปที่ 2.10 การเชื่อมต่อหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51

61849



รูปที่ 2.11 การเชื่อมต่อหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51

จะเห็นได้ว่า หน่วยความจำข้อมูลส่วนบนและรีจิสเตอร์ฟังก์ชันพิเศษมีตำแหน่งทับซ้อนกัน แต่จะใช้การติดต่อที่แตกต่างกัน และในไมโครคอนโทรลเลอร์ MCS-51 บางเบอร์จะไม่มีหน่วยความจำข้อมูลส่วนบน

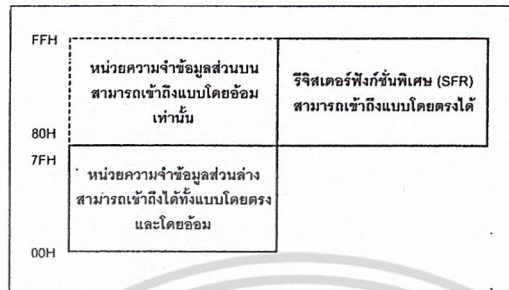
ขนาดของหน่วยความจำข้อมูลของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชโดยแท้จริงแล้วมีเพียง 256 ไบต์ แต่ด้วยการจัดการเข้าถึงที่แตกต่างกัน จึงเสมือนว่า ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีหน่วยความจำข้อมูลภายในสูงถึง 384 ไบต์ โดยในหน่วยความจำข้อมูลส่วนล่างขนาด 128 ไบต์ มีแอดเดรสอยู่ที่ 00H-7FH สามารถเข้าถึงได้โดยตรงและโดยอ้อม สำหรับหน่วยความจำข้อมูลส่วนบนมีขนาด 128 ไบต์เช่นกัน มีแอดเดรสอยู่ที่ 80H-FFH สามารถเข้าถึงแบบโดยอ้อมเท่านั้น ในขณะที่รีจิสเตอร์ SFR มีแอดเดรสอยู่ที่ 80H-FFH เช่นเดียวกับหน่วยความจำข้อมูลส่วนบน แต่สำหรับรีจิสเตอร์ SFR ใช้การเข้าถึงแบบโดยตรง ดังนั้นเพื่อความสะดวกและง่าย ตลอดจนป้องกันความสับสนในการเขียนโปรแกรมสำหรับผู้เริ่มต้นจึงควรใช้หน่วยความจำข้อมูลภายในเพียง 128 ไบต์จากหน่วยความจำข้อมูลส่วนล่างร่วมกับรีจิสเตอร์ SFR

ในรูปที่ 2.12 แสดงการจัดสรรหน่วยความจำข้อมูลส่วนล่าง หน่วยความจำ 32 ไบต์ต่ำสุดที่แอดเดรส 00H-1FH แบ่งเป็น 4 กลุ่ม เรียกว่า 4 แแบงก์ (bank) แต่ละแบงก์มีรีจิสเตอร์ 8 ตัวคือ R0-R7 การติดต่อกับหน่วยความจำในแบงก์ใดให้กำหนดที่รีจิสเตอร์ PSW (Program Status Word register)

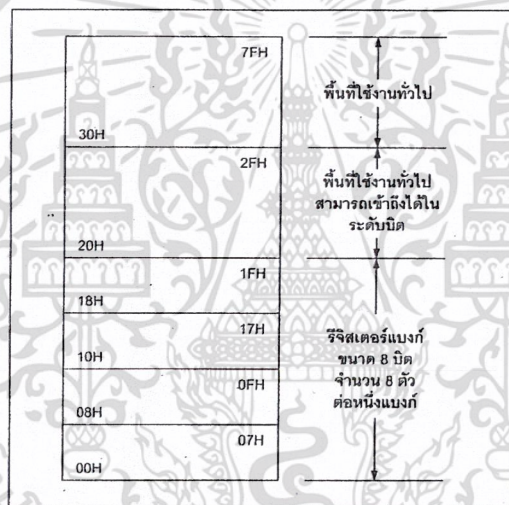
หน่วยความจำข้อมูล 16 ไบต์ถัดมาที่แอดเดรส 20H-2FH เป็นพื้นที่สำหรับใช้งานทั่วไป สามารถเข้าถึงได้ในระดับบิต (Bit addressable) และหน่วยความจำข้อมูลที่เหลือ 80 ไบต์ จะต้องแบ่งส่วนหนึ่งสำรองไว้เป็นพื้นที่ของสแต็ก (stack: ที่พักข้อมูลชั่วคราวในกรณีที่ใช้ฟังก์ชันการกระโดดไปทำงานในโปรแกรมย่อย) การเข้าถึงหน่วยความจำในส่วนนี้ต้องใช้การเข้าถึงในระดับไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 2.13 แสดงโครงสร้างของหน่วยความจำข้อมูลส่วนบน ซึ่งจะมีลักษณะที่คล้ายกับหน่วยความจำข้อมูลส่วนล่าง หากแต่ใน 80 ไบต์บนไม่จำเป็นต้องสำรองไว้สำหรับสแต็ก และต้องใช้ในการเข้าถึงในลักษณะ โดยอ้อมเท่านั้น



รูปที่ 2.12 การจัดสรรพื้นที่ของหน่วยความจำข้อมูลภายใน ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช



รูปที่ 2.13 การจัดสรรพื้นที่ของหน่วยความจำข้อมูลภายในส่วนล่างของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

### รีจิสเตอร์ฟังก์ชันพิเศษ

(Special Function Register: SFR)

เป็นรีจิสเตอร์ที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์MCS-51 มีด้วยกัน22 ตัว สำหรับเบอร์ AT89C51 และ 28 ตัวในเบอร์ AT89C52 และอนุกรม AT89Sxx ทั้งนี้เนื่องจากใน AT89C52 และAT89Sxxมีจำนวนไทเมอร์เคาน์เตอร์มากกว่าAT89C51

รีจิสเตอร์ SFR มีแอดเดรสอยู่ระหว่าง 80H-FFH ในพื้นที่ ของหน่วยความจำข้อมูลส่วนบน สามารถเข้าถึงได้โดยตรง (direct addressing) ในรูปที่ 2.14 แสดงการจัดสรรพื้นที่ที่ขอรหัสรีจิสเตอร์ SFR แต่ละตัวในหน่วยความจำข้อมูลส่วนบน สำหรับรายละเอียดเบื้องต้นของรีจิสเตอร์ SFR มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รีจิสเตอร์แสดงสถานะของโปรแกรม

(Program Status word: PSW)

เป็นรีจิสเตอร์ขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต จึงสามารถกำหนดค่าในแต่ละบิตของรีจิสเตอร์ตัวนี้ได้อย่างอิสระ มีแอดเดรสอยู่ที่ DOH ทำหน้าที่เก็บสถานะของการทำงานของโปรแกรมในขณะนั้นจะเรียกสถานะต่างๆ ของโปรแกรมว่า แฟล็ก (flag) เมื่อซีพียูกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้วเกิดการเปลี่ยนแปลงสถานะขึ้น ผลของการเปลี่ยนแปลงนั้นจะปรากฏที่บิตต่างๆ ของรีจิสเตอร์ รายละเอียดของแต่ละ บิตในรีจิสเตอร์ PSW แสดงในรูปที่ 2.15

จะเห็นได้ว่า นอกจากรีจิสเตอร์ PSW ที่ใช้ในการเก็บสถานะของโปรแกรมแล้ว ที่บิต RS0 และ S11 ยังใช้ในการเลือกแบงก์ของหน่วยความจำส่วนล่าง ซึ่งเป็นพื้นที่ของรีจิสเตอร์ R0-R7 ด้วย ดังมีรายละเอียดแสดงใน ตารางที่ 2.8 โดยปกติแล้วในการใช้งานรีจิสเตอร์ R0-R7 มักนิยมใช้แบงก์ 0 เป็นลำดับแรก หากไม่เพียงพอจึงเลือกในแบงก์อื่นๆ มาใช้ แต่ต้องระมัดระวังในการกำหนดค่าและลำดับการติดต่อให้ดี มิเช่นนั้น อาจทำให้การเขียนโปรแกรมเกิดความสับสน ดังนั้น สำหรับผู้เริ่มต้นใช้งานไมโครคอนโทรลเลอร์ MCS-51 จึงควรเลือกใช้รีจิสเตอร์ R0-R7 ในแบงก์ 0 เพียงแบงก์เดียวให้ชำนาญเสียก่อน

การกำหนดค่าของรีจิสเตอร์ PSW เพื่อเลือกใช้งานรีจิสเตอร์ R0-R7 ควรกำหนดไว้ที่ตอนต้นของโปรแกรมเสมอ เพื่อจะได้เขียนโปรแกรมติดต่อกับรีจิสเตอร์ R0-R7 ได้อย่างสะดวกและไม่เกิดความผิดพลาด

## แอกคิวมูเลเตอร์ (Accumulator: ACC)

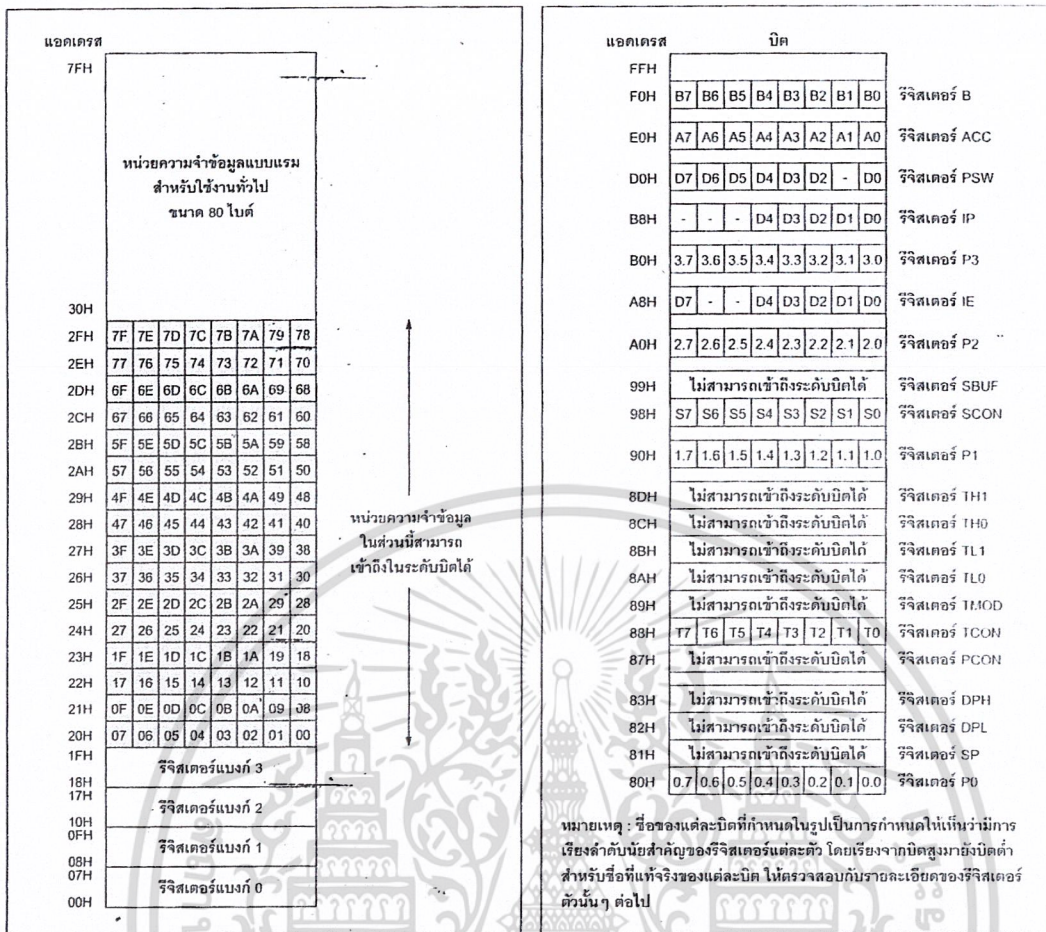
มีขนาด 8 บิต มีแอดเดรสอยู่ที่ตำแหน่ง EOH เป็นรีจิสเตอร์ที่ใช้สำหรับเก็บข้อมูลหรือผลลัพธ์ที่ได้จากการทำงานของไมโครคอนโทรลเลอร์ โดยเฉพาะอย่างยิ่งในการคำนวณทางคณิตศาสตร์และลอจิก ก่อนที่จะส่งข้อมูล หรือผลลัพธ์ที่ได้ให้แก่ซีพียูเพื่อทำการประมวลผลต่อไป อาจเรียกสั้นๆ ว่า รีจิสเตอร์ A หรือ Acc รีจิสเตอร์นี้สามารถเข้าถึงระดับบิตได้

## รีจิสเตอร์ B

มีขนาด 8 บิต มีแอดเดรสอยู่ที่ FOH มีหน้าที่พิเศษคือ หากต้องการคูณหรือหารทางคณิตศาสตร์ ต้องนำข้อมูลที่ต้องการหารหรือคำนวณมาเก็บไว้ในรีจิสเตอร์ B แล้วจึงกระทำคำสั่งการคูณหรือหารกับค่าในรีจิสเตอร์ A ต่อไป

ในกรณีที่ไม่ได้มีความต้องการคูณหรือหารข้อมูล สามารถใช้รีจิสเตอร์ B นี้ในการเก็บข้อมูลทั่วไปได้เหมือนกับรีจิสเตอร์ปกติ และสามารถเข้าถึงในระดับบิตได้เช่นเดียวกับรีจิสเตอร์ A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 โครงสร้างของหน่วยความจำข้อมูลภายในส่วนบนของไมโครคอนโทรลเลอร์ MCS-51

รูปที่ 2.15 การจัดสรรพื้นที่ของรีจิสเตอร์ฟังก์ชันพิเศษ

**โปรแกรมเคาน์เตอร์ (Program Counter: PC)**

มีขนาด 16 บิต มีหน้าที่แจ้งแอดเดรสของหน่วยความจำโปรแกรมในตำแหน่งถัดไปที่ซีพียูจะต้องไปทำงาน รีจิสเตอร์ PC เป็นรีจิสเตอร์ตัวเดียวที่ไม่ได้จัดสรรไว้ร่วมกับรีจิสเตอร์ SFR ตัวอื่นๆ การเปลี่ยนแปลงค่าของรีจิสเตอร์ PC จะขึ้นอยู่กับผลของการกระทำคำสั่งภายในหน่วยความจำโปรแกรมที่ผู้เขียนโปรแกรมกำหนด

รีจิสเตอร์ PC มีความสำคัญมาก โดยเฉพาะอย่างยิ่งในการตรวจสอบการทำงานของโปรแกรมว่าดำเนินไปตามลำดับขั้นตอนตามที่กำหนดไว้หรือไม่

**สแต็กพอยน์เตอร์ (Stack Pointer: SP)**

หรือรีจิสเตอร์ตัวชี้สแต็ก มีขนาด 8 บิต มีแอดเดรสอยู่ที่ 81H ใช้ในการเก็บค่าตำแหน่งของตัวชี้สแต็ก ซึ่งสามารถเปลี่ยนแปลงได้เมื่อซีพียูมีการกระโดดไปทำงานที่โปรแกรมย่อย หรือกระโดดโปรแกรมย่อยกลับมายังโปรแกรมหลักเมื่อมีการรีเซตเกิดขึ้น (รีเซต: การกระทำที่ส่งผลให้ซีพียูต้องเริ่มต้นการทำงานใหม่ตั้งแต่ต้น) ค่าของรีจิสเตอร์ SP จะเท่ากับ 07H ดังนั้นแอดเดรสแรกของพื้นที่ที่สำรองไว้ทำหน้าที่เป็นสแต็กจะเท่ากับ 08H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RS1	RS0	แบงก์ของรีจิสเตอร์	ช่วงแอดเดรส
0	0	แบงก์ 0	00H-07H
0	1	แบงก์ 1	08H-0FH
1	0	แบงก์ 2	10H-17H
1	1	แบงก์ 3	18H-1FH

ตารางที่ 2.3 การเลือกแบงก์ของหน่วยความจำส่วนล่างเพื่อติดต่อกับรีจิสเตอร์แบงก์ R0-R7

### รีจิสเตอร์ชี้ข้อมูลหรือค้ำพอยน์เตอร์ (Data Pointer: DPTR)

มีขนาด 16 บิต โดยแบ่งเป็นรีจิสเตอร์ชี้ข้อมูลไบต์สูง (DPH) และรีจิสเตอร์ชี้ข้อมูลไบต์ต่ำ (DPL) แต่ละตัวมีขนาด 8 บิต มีแอดเดรสอยู่ที่ 82H สำหรับ DPL และ 83H สำหรับ DPH รีจิสเตอร์ DPTR นี้ใช้ในการเก็บค่าแอดเดรสของหน่วยความจำหรืออุปกรณ์ภายนอกที่ไม่โครคอนโทรลเลอร์ ต้องการติดต่อกับ

### รีจิสเตอร์พอร์ต (Port register)

เป็นรีจิสเตอร์ขนาด 8 บิตที่ใช้เก็บข้อมูลของแต่ละพอร์ต ของไมโครคอนโทรลเลอร์ MCS-51 มี 4 ตัวคือ รีจิสเตอร์พอร์ต 0 หรือ P0 มีแอดเดรสอยู่ที่ 80H, รีจิสเตอร์พอร์ต 1 หรือ P1 มีแอดเดรสอยู่ที่ 90H, รีจิสเตอร์พอร์ต 2 หรือ P2 มีแอดเดรสอยู่ที่ A0H และรีจิสเตอร์พอร์ต 3 หรือ P3 มีแอดเดรสอยู่ที่ B0H รีจิสเตอร์ทุกตัวสามารถเข้าถึงได้ในระดับบิต เมื่อต้องการอ่านหรือเขียนข้อมูลออกไปยังพอร์ตของไมโครคอนโทรลเลอร์ จะต้องกระทำผ่านรีจิสเตอร์นี้ทุกครั้ง

### รีจิสเตอร์บัฟเฟอร์ข้อมูลอนุกรม (Serial Data Buffer: SBUF)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 99H ใช้ในการเก็บข้อมูลที่ส่งออกหรือรับเข้าของวงจรถ่ายโอนอนุกรมที่มีอยู่ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดยภายในรีจิสเตอร์ SBUF นี้จะแบ่งออกเป็น 2 ส่วนคือ รีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูล (Transmit buffer register) และ รีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูล (receive buffer register) เมื่อมีการเขียนข้อมูลมายังรีจิสเตอร์ SBUF ข้อมูลนั้นจะถูกส่งต่อไปยังบัฟเฟอร์สำหรับส่งข้อมูล เพื่อส่งออกจากไมโครคอนโทรลเลอร์ผ่านทางขา TxDหรือขา P2.8 ในกรณีที่มีการอ่านข้อมูลจากรีจิสเตอร์ SBUF ข้อมูลจะถูกส่งผ่านไปยังรีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูลเพื่อส่งไปยังไมโครคอนโทรลเลอร์ต่อไป สำหรับการรับข้อมูลอนุกรมจากภายนอกนั้นจะผ่านมาจากขา RxD หรือ P3.0 ของไมโครคอนโทรลเลอร์ MCS-51

### รีจิสเตอร์ไทมเมอร์ (Timer register)

เป็นรีจิสเตอร์ขนาด 16 บิต แบ่งเป็นไบต์สูงและไบต์ต่ำเช่นเดียวกับรีจิสเตอร์ DPTR รีจิสเตอร์ไทมเมอร์ใช้ในการเก็บค่าของตัวนับหรือเคาน์เตอร์ (counter) ภายในไมโครคอนโทรลเลอร์ เพื่อใช้ในการสร้างฐานเวลาจับเวลา หรือนับจำนวนพัลส์สัญญาณพิกภายใน บางทีเรียกเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ตัวนี้ว่า รีจิสเตอร์ไทเมอร์/เคาน์เตอร์

ในไมโครคอนโทรลเลอร์ เบอร์ AT89C51 มีรีจิสเตอร์ไทเมอร์/เคาน์เตอร์ 2 ตัว แบ่งเป็น T0 หรือ Timer0 และ T1 หรือ Timer 1 ในรีจิสเตอร์ยังแบ่งเป็นรีจิสเตอร์ไทเมอร์ไบต์ต่ำ (TL) และ รีจิสเตอร์ไทเมอร์ไบต์สูง (TH) เหมือนกัน โดยรีจิสเตอร์ TLO มีแอดเดรสอยู่ที่ 8AH รีจิสเตอร์ TH0 มีแอดเดรสอยู่ที่ 8BH ในขณะที่ TL1 และ TH1 มีแอดเดรสอยู่ที่ 8CH และ 8DH สำหรับในเบอร์ AT89C52และในอนุกรมAT89Sxx จะมีรีจิสเตอร์ไทเมอร์/เคาน์เตอร์ถึง 3 ตัว โดยมีรีจิสเตอร์ TL2 และ TH2 ซึ่งมีแอดเดรสอยู่ที่ 0CCH และ 0CDH เพิ่มเติมเข้ามา

**รีจิสเตอร์แคปเจอร์ (Capture register)**

เป็นรีจิสเตอร์ขนาด 16 บิต มีเฉพาะในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxx เท่านั้น เนื่องจากต้องใช้ร่วมกับไทเมอร์/เคาน์เตอร์ 2 (Timer 2) โดยรีจิสเตอร์แคปเจอร์นี้มีชื่อเรียกอย่างย่อว่า รีจิสเตอร์ RCAP2 ซึ่งแบ่งออกเป็นไบต์ต่ำคือ RCAP2L มีแอดเดรสอยู่ที่ 0CAH และไบต์สูงคือ RCAP2H มีแอดเดรสอยู่ที่ 0CBH

รีจิสเตอร์แคปเจอร์จะถูกใช้งานเมื่อกำหนดให้ไทเมอร์ 2 ทำงานในโหมดแคปเจอร์ ซึ่งเป็นโหมดที่กำหนดให้ไมโครคอนโทรลเลอร์ทำการตรวจจับการเปลี่ยนแปลงสถานะทางลอจิกที่ขา T2EX ทั้งนี้เพื่อใช้ประโยชน์ในการวัดคาบเวลา ความถี่ และการเปลี่ยนแปลงของสัญญาณพัลส์ที่ขา T2EX

**รีจิสเตอร์ควบคุม (Control register)**

รีจิสเตอร์ PCON เป็นรีจิสเตอร์ที่เกี่ยวข้องกับการกำหนดอัตราการรับส่งข้อมูลของวงจรงอสารอนุกรมและกำหนดการทำงานในโหมดประหยัดพลังงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

รีจิสเตอร์ SCON ใช้ในรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของวงจรงอสารอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

รีจิสเตอร์ TCON และ T2CON เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดย T2CON ใช้สำหรับไทเมอร์/เคาน์เตอร์ 2 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxxx

รีจิสเตอร์TMOD และ T2MOD เป็นรีจิสเตอร์ที่ใช้กำหนดโหมดหรือลักษณะในการทำงานของไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดย T2MOD ใช้สำหรับไทเมอร์/เคาน์เตอร์ 2 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxxx

รีจิสเตอร์ IE และ IP เป็นรีจิสเตอร์ที่เกี่ยวข้องกับการตอบสนองการอินเทอร์รัปต์

(Interrupt: การขัดจังหวะการทำงานปกติของซีพียู โดย IE เป็นรีจิสเตอร์สำหรับเอ็นเอเบิลหรือใช้ในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

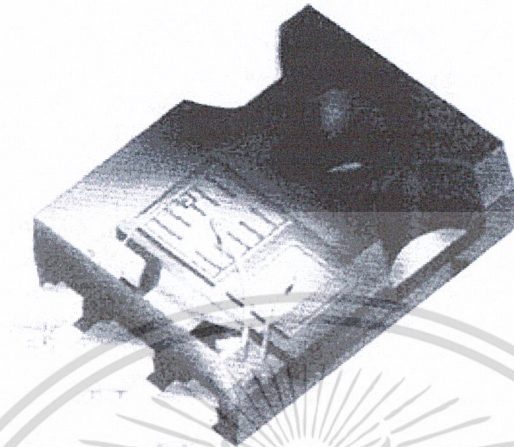
การกำหนดลักษณะของการตอบสนองการอินเทอร์เน็ต รัปต์ ในขณะที่ IP เป็นรีจิสเตอร์สำหรับกำหนดลำดับความสำคัญของการตอบสนองการอินเทอร์เน็ตว่า จะให้ซีพียูตอบสนองการเกิดอินเทอร์เน็ต รัปต์ ในลักษณะใดก่อนหรือหลัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3

## เทคโนโลยี IGBT



### เทคโนโลยีไอจีบีที IGBT

ความล้ำหน้าทางเทคโนโลยีอิเล็กทรอนิกส์คือจุดกำเนิดของ Insulate Gate Bipolar Transistor : IGBT

หากจะกล่าวถึงอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในงานในด้านเพาเวอร์กำลังหรือเพาเวอร์คอนโทรล ก็เห็นจะมีอยู่ไม่กี่ชนิด ซึ่งในแต่ละชนิดก็มีข้อจำกัดและความสามารถในการทำงาน ที่แตกต่างกันออกไปตามลักษณะของการนำไปใช้ควบคุมงาน ด้านต่างๆซึ่งอาจจะรวมไปถึงความไม่ล่าสมัยเร็วจนเกินไปของอุปกรณ์ที่ถูกเลือกมาใช้งานด้วยความผูกพันจากของเดิม

อุปกรณ์เพาเวอร์อิเล็กทรอนิกส์คอนโทรล ที่พอจะคุ้นเคยและใช้งานกันอย่างกว้างขวางในขณะนี้ ก็เห็นจะไม่พ้นเอสซีอาร์ (SCR) ไตรแอก (TRIAC) ทรานซิสเตอร์กำลังและมอสเฟต โดยเฉพาะทรานซิสเตอร์และมอสเฟต ที่จะเป็จุดพัฒนาการของอุปกรณ์ชนิดใหม่นี้ ซึ่งอุปกรณ์ทั้งสองชนิด ก็มีคุณสมบัติที่แตกต่างกันออกไป กล่าวคือ

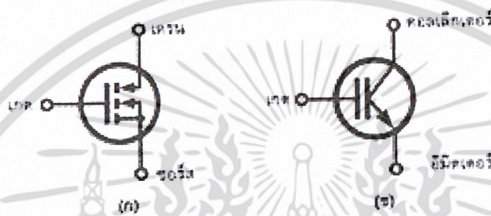
ทรานซิสเตอร์กำลังขณะอยู่ในสภานำกระแสจะมีอัตราการสูญเสียกำลังงานต่ำ มีอัตราแรงดันและขยายกระแสได้สูง แต่ความเร็วในการสวิตซ์ทำงานยังต่ำอยู่ โดยเฉพาะช่วงหยุดนำกระแส จะมีช่วงเวลาที่ยาวกว่า ซึ่งจะเป็นคุณสมบัติที่ตรงข้ามกับเพาเวอร์มอสเฟต ที่มีความเร็วในการสวิตซ์ทำงานนำกระแสและหยุดนำกระแสได้เร็วกว่ามาก แต่ก็มีอัตราการสูญเสียกำลังงานสูงมากเช่นกัน

จากเหตุผลที่กล่าวมาของเพาเวอร์ทรานซิสเตอร์และเพาเวอร์มอสเฟตจึงได้มีการพัฒนาอุปกรณ์ประเภทนี้ จนสามารถได้อุปกรณ์เพาเวอร์อิเล็กทรอนิกส์คอนโทรลชนิดใหม่ขึ้นมา โดยคุณสมบัติต่าง ๆ จะรวมเอาข้อได้เปรียบของทรานซิสเตอร์ไบโพลาร์และมอสเฟตเข้ามารวมไว้ในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ชนิดใหม่นี้ โดยมีการตั้งชื่อหรือเรียกชื่ออย่างเป็นทางการว่า ไอจีบีที (Insulate Gate Bipolar Transistor : IGBT)

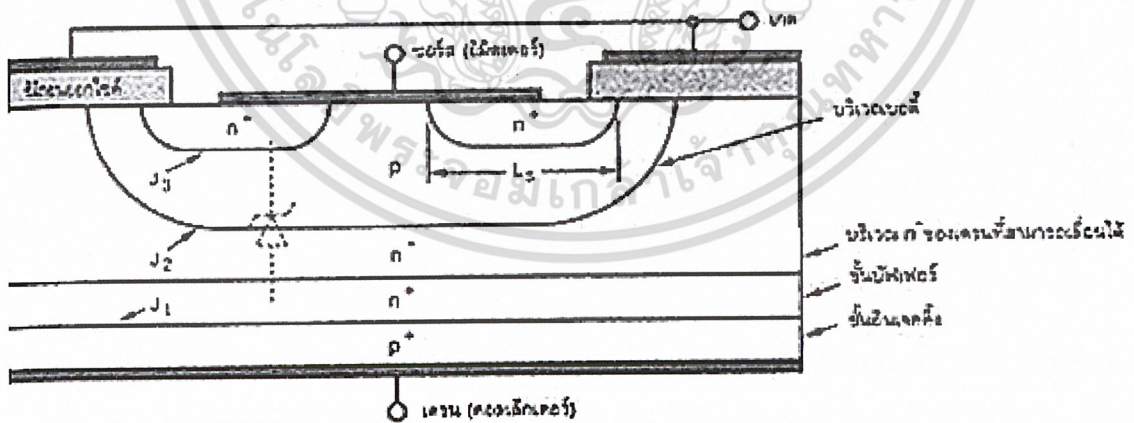
**โครงสร้างและสัญลักษณ์**

IGBT ค่อนข้างจะเป็นอุปกรณ์ชนิดใหม่อยู่ โดยเฉพาะในตลาดอิเล็กทรอนิกส์เมืองไทย ซึ่งก็พอจะมีใช้กันบ้าง และมีจำหน่ายกันหลายเบอร์ด้วยกันจากความเป็นอุปกรณ์ หน้าใหม่จึงมีสัญญาณแทนชนิดและตัว IGBT อยู่หลายรูปแบบด้วยกัน ขึ้นอยู่กับผู้ผลิตว่าจะใช้สัญลักษณ์ใดเป็นสัญลักษณ์ประจำสินค้าที่ผลิตขึ้น จากที่พบมากที่สุดก็มีใช้กันอยู่สองแบบ ดังแสดงไว้ในรูปที่ 3.1 ซึ่งเป็นสัญลักษณ์และชื่อเรียกขานต่าง ๆ ของ IGBT ชนิดเอ็นแซนเนล



รูปที่ 3.1 สัญลักษณ์และการเรียกชื่อขานของ IGBT ทั้ง 2 แบบ

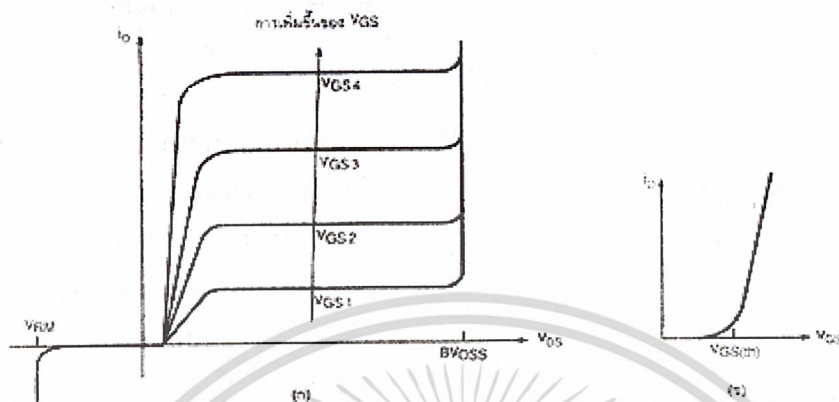
จากรูปที่ 3.1 (ก) จะเห็นว่ามีลักษณะคล้ายสัญลักษณ์ของมอสเฟตมาก เพียงแต่ว่าสัญลักษณ์ของ IGBT นั้นจะมีลูกศรเพิ่มขึ้นมาตรงขาเกรนลักษณะของลูกศรจะชี้เข้าหาตัว หรือชี้เข้าหาชั้นของซิลิคอนภายในตัว IGBT ในบทความนี้จะใช้สัญลักษณ์ในรูป (ข) จะเหมือนกับสัญลักษณ์ของทรานซิสเตอร์ แต่ตรงขาเกต (หรือเบสของทรานซิสเตอร์) จะเพิ่มขีดขึ้นมาขึ้นไม่ได้ต่อถึงกันโดยตรงกับขาที่ต่อออกมาภายนอก



รูปที่ 3.2 ภาพตัดขวาง โครงสร้างพื้นฐานของ IGBT

โครงสร้างของ IGBT ชนิดเอ็นแซนเนลแสดงเป็นภาพตัดขวางได้ดังรูปที่ 3.2 โครงสร้างโดยรวมส่วนใหญ่มีลักษณะคล้ายกับโครงสร้างของมอสเฟตมาก จะแตกต่างกันตรงที่ IGBT จะมีชั้น P+ หรือชั้นอินเจกติ้ง (injecting) ต่ออยู่ระหว่างขาเกรน ซึ่งในมอสเฟตนั้นไม่มี จากการที่ขาเกต

ถูกกั้นด้วยชั้นของซิลิคอนออกไซด์ ( $\text{SiO}_2$ ) เป็นผลทำให้ความต้านทานอินพุตที่ขาเกตมีค่าสูงมาก เหมือนกับเพาเวอร์มอสเฟต โดยทั่วไปจะมีค่าอยู่ในช่วง 109 โอห์ม



รูปที่ 3.3 (ก) กราฟแสดงลักษณะคุณสมบัติระหว่างกระแสและแรงดันของ IGBT (ข) กราฟแสดงลักษณะสมบัติการถ่ายโอนของ IGBT

จากผลดังกล่าวทำให้ลักษณะของกราฟแสดงคุณสมบัติของกระแสและแรงดันของ IGBT มีลักษณะคล้ายกราฟของทรานซิสเตอร์ แต่การควบคุมกระแสเดรน จะอาศัยการควบคุมแรงดันระหว่างขาเกตกับขาซอร์ส มากกว่า การควบคุมกระแสที่ขาเบสเหมือนกันทรานซิสเตอร์ ซึ่งกราฟแสดงคุณสมบัติของกระแสและแรงดันของ IGBT แสดงไว้ในรูปที่ 3.3 (ก) และสำหรับรูปที่ 3.3 (ข) เป็นกราฟคุณสมบัติการถ่ายโอนกระแสและแรงดัน

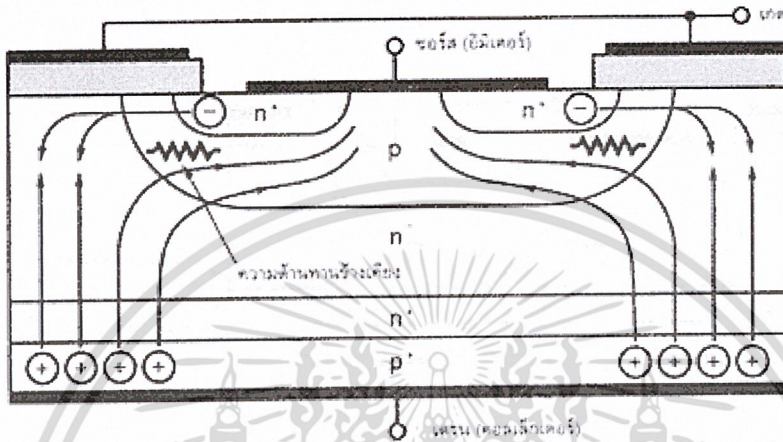
รูปกราฟแสดงให้เห็นว่าส่วนใหญ่ของเส้นกราฟจะมีลักษณะเป็นเส้นตรงแต่จะเริ่มโค้งที่กระแสเดรนมีค่าต่ำ ๆ นั่นก็คือจุดที่แรงดันระหว่างขาเกตและขาซอร์สต่ำลง ใกล้แรงดันจุดเริ่มเปลี่ยนสถานะการทำงาน (จุด threshold voltage :  $V_{gs(th)}$  โดยถ้าแรงดันระหว่างขาคอนดักเตอร์หรือคัทออฟ ในกรณีของ IGBT ชนิดพีแชนเนลนั้น คุณสมบัติจะคล้ายกับเอ็นแชนเนล แต่โครงสร้างและสัญลักษณ์จะมีลักษณะตรงกันข้ามกับเอ็นแชนเนล เช่น ชนิดของสารที่โด๊ปจากเอ็นแชนเนล ก็จะเปลี่ยนเป็นตรงกันข้าม สัญลักษณ์ลูกศรก็จะกลับเอาหัวลูกศรกลับไปในทางตรงกันข้าม

### สถานะนำกระแส

เมื่อขาเดรนได้รับแรงดันไบแอสตรงคือเป็นบวกเทียบกับซอร์ส และแรงดันระหว่างเกตกับซอร์สมีค่าเกิน  $V_{gs(th)}$  ประจุไฟฟ้าบวกที่เกิดจากแรงดันที่ขาเกตจะดึงเอาอิเล็กตรอน ให้มารวมกันอยู่ในบริเวณภายใต้เกต ทำให้ชั้นบอดี้ (body layer) ตรงส่วนใต้เกตแปรสภาพเป็น n ทำให้เกิดการต่อกันของบริเวณ n- (drift region) เข้ากับบริเวณ n+ (source region) ซึ่งลักษณะเช่นนี้เหมือนกับการทำงานของ มอสเฟต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระแสอิเล็กตรอนที่ไหลจากขาซอร์สผ่านบริเวณได้เกิดมายังบริเวณลอยเลื่อน  $n^-$  จะรวมกับโฮลที่เป็นพาหะข้างน้อยที่ถูกฉีดมาจากชั้นอินเจกต์ติ้ง  $P^+$  (ดูรูปโครงสร้างในรูปที่ 2) เพราะรอยต่อ  $J1$  ได้รับแรงดันไบแอสตรง ทำให้ JGBT อยู่ในสภาวะนำกระแส เกิดการไหลของกระแสไฟฟ้าจากเดรนไปซอร์สได้ การรวมกันของโฮลและอิเล็กตรอนภายในบริเวณ  $n^-$  เรียกว่า การมอดูเลตสภาพนำ (conductivity modulation)



รูปที่ 3.4 ทิศทางการไหลของอิเล็กตรอนและโฮลในขณะนำกระแส

ผลของการมอดูเลตนี้จะทำให้ความต้านทานของบริเวณ  $n^-$  มีค่าต่ำลงเป็นการเพิ่มความสามารรถในการขับผ่านกระแสได้สูงขึ้น ซึ่งจะมีลักษณะเหมือนกับทรานซิสเตอร์กำลัง ผลของความต้านทานที่ลดลง ทำให้แรงดันตกคร่อมที่สภาวะนำกระแสลดลง การสูญเสียกำลังงานขณะนำกระแสจึงลดลงด้วย ทิศทางการไหลของ

#### สภาวะหยุดนำกระแส

เมื่อแรงดันระหว่างเกตและซอร์สลดลงต่ำกว่าแรงดัน  $V_{gs(th)}$  จะทำให้มีแรงดันไม่เพียงพอสำหรับการแปรสภาพชั้นบอดี้  $p$  เป็น  $n^-$  ได้ ทำให้บริเวณ  $n^-$  ไม่ต่อกับบริเวณซอร์ส  $n^+$  IGBT จึงอยู่ในสภาวะหยุดนำกระแส ในสภาวะนี้รอยต่อ  $J2$  ที่ได้รับแรงดันไบแอสกลับ จะทำให้เกิดกระแสรั่วไหลเพียงเล็กน้อยเท่านั้น นอกจากนี้ยังทำให้เกิดบริเวณปลอดพาหะ (depletion region) ขึ้นที่รอยต่อ  $J2$  ด้วย

บริเวณปลอดพาหะนี้จะขยายบริเวณกว้างขึ้นจนเกินเข้ามายังบริเวณ  $n^-$  มากกว่า ที่จะขยายไปยังบริเวณชั้นบอดี้  $p$  ทั้งนี้เพราะชั้นบอดี้  $p$  มีความหนาแน่นในการโด๊ปสารมากกว่า ถ้าความหนาแน่นของสารที่โด๊ปในบริเวณลอยเลื่อน  $n^-$  มากเพียงพอ ก็จะทำให้การขยายของบริเวณปลอดพาหะ ไม่สามารถแตะกับชั้นอินเจกต์ติ้ง  $p^+$  ได้ ชั้นบัฟเฟอร์  $n^+$  (buffer layer) (ดังในรูปที่ 3.2) ก็ไม่จำเป็นต้องทำให้เกิดขึ้น หรือไม่จำเป็นต้องโด๊ปสาร

ทั้งนี้เพราะการแตะกันของบริเวณทั้งสองจะทำให้เกิดการพังทลายทางด้านไบแอสตรง สำหรับ IGBT ที่ไม่มีกริดโด้ปสารในชั้นบัฟเฟอร์  $n^+$  นี้ จะเรียกว่า IGBT แบบสมมาตร ซึ่งจะมีอัตราทนแรงดันย้อนกลับ ( $V_{rm}$  หรือ  $BV_{sds}$ ) สูงพอ ๆ กับค่าอัตราทนแรงดันไหลตรง ( $BV_{dss}$ ) เหมาะสำหรับการนำไปประยุกต์ใช้ในวงจรไฟฟ้ากระแสสลับ

การลดความหนาของบริเวณ  $n^-$  ลงแต่ยังคงความสามารถของอัตราทนแรงดันไหลตรงไว้สามารถทำได้โดยเพิ่มชั้นบัฟเฟอร์  $n^+$  เข้าไปเพื่อป้องกันการแตะกันของบริเวณปลอดพาหะ กับบริเวณอินเจ็กต์  $p^+$  ซึ่งจะเรียก IGBT ชนิดนี้ว่า IGBT แบบไม่สมมาตร และจากการลดความหนาของบริเวณลอยเลื่อน  $n^-$  ลง จะช่วยส่งผลให้เกิดข้อดีสองประการคือ

- ทำให้แรงดันตกคร่อมขณะนำกระแสต่ำลง เป็นผลให้การสูญเสียกำลังงานลดน้อยลงด้วย
- ช่วยลดช่วงเวลาหยุดนำกระแสให้สั้นลงด้วย

แต่ข้อเสียของการเพิ่มชั้นบัฟเฟอร์  $n^+$  ก็มี คือจะลดความสามารถของอัตราทนแรงดันย้อนกลับให้ลดลงเหลือเพียงไม่กี่สิบ โวลต์ ทั้งนี้เพราะเมื่อ IGBT ได้รับแรงดันไบแอสกลับ ที่ขาแตรนรอยต่อ  $J_1$  ซึ่งทั้งสองข้างมีความหนาแน่นในการโด้ปของสารมาก จะไม่สามารถทนแรงดันย้อนกลับได้สูง ดังนั้น IGBT ชนิดนี้จึงไม่เหมาะที่จะนำไปใช้ในวงจรไฟฟ้ากระแสสลับ อิเล็กตรอนและโฮลแสดงไว้ในรูปที่ 3.4

### การแลตซ์ใน IGBT

นอกจากโฮลส่วนใหญ่ที่รวมกับอิเล็กตรอนภายใต้บริเวณ  $n^-$  แล้ว ยังมีกระแสโฮลบางส่วนที่ไหลข้ามบริเวณ  $n^-$  เข้าสู่บริเวณชั้นบอดี้  $p$  โดยตรงผลของกระแสโฮลนี้ทำให้เกิด แรงดันตกคร่อมความต้านทานข้างเคียง (Internal resistance) ดังในรูปที่ 3.4 ถ้าแรงดันนี้มีค่ามากพอคือประมาณ 0.7 โวลต์ จะทำให้รอยต่อ  $J_3$  ได้รับไบแอสตรงเป็นผลให้อิเล็กตรอน จากบริเวณซอร์ส  $n^+$  ถูกฉีดเข้าไปในชั้นบอดี้  $p$

ถ้าดูจากวงจรสมมูลในรูปที่ 3.6 (ค) จะหมายถึงขาเบสและอิมิตเตอร์ของทรานซิสเตอร์เอ็นพีเอ็น ได้รับแรงดันไบแอสตรงส่งผลให้ทรินสเตอร์ ซึ่งแฝงอยู่ในโครงสร้างของ IGBT อยู่ในสภาวะแลตซ์การนำกระแส ทำให้ที่ขาเกตไม่สามารถควบคุมปริมาณของกระแสเดรนนี้จะขึ้นอยู่กับตัวต้านทานที่นำมาต่อในวงจรภายนอก ถ้าหากมีการแลตซ์เกิดขึ้นเป็นเวลานาน อาจทำให้ IGBT เสียหายได้ เพราะมีการสูญเสียกำลังงานเกินค่าพิคกที่ทนได้

ส่วนใหญ่หรือเป็นมาตรฐานคู่มือของผู้ผลิต มักจะมีการบอกค่ากระแสเดรนสูงสุด ที่สามารถไหลผ่าน IGBT ได้โดยยังไม่เกิดการแลตซ์ขึ้น ( $I_{dm}$ ) แต่เนื่องจากกระแสเดรน ถูกกำหนดหรือควบคุมโดยตรง จากแรงดันระหว่างขาเกตกับซอร์ส บางครั้งคู่มือจึงบอกค่าแรงดันระหว่างเกตและซอร์สสูงสุด ที่จะไม่ทำให้เกิดการแลตซ์แทนการบอกค่ากระแสเดรน สูงสุด ( $I_{dm}$ )

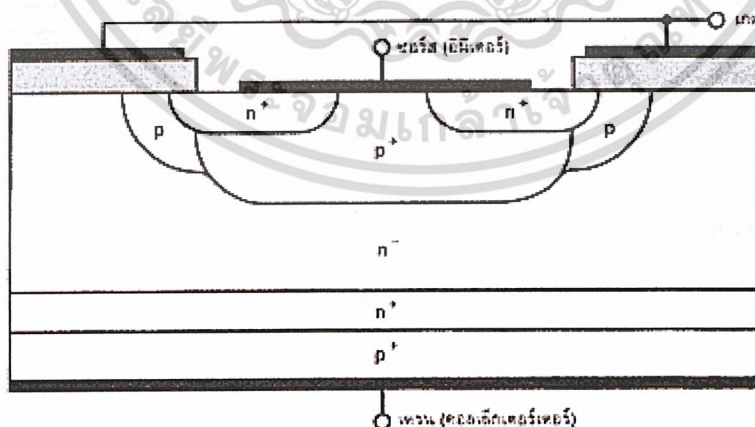
การแลตซ์ที่กล่าวถึงข้างต้นเรียกว่าการแลตซ์ใน โหมดสแตติก เพราะเกิดขึ้นเมื่อกระแสที่ไหลในสถานะนำกระแส มีค่าเกิน  $I_{dm}$  แต่ลักษณะการแลตซ์นี้อาจเกิดขึ้นได้ เรียกว่าใน โหมดไดนามิก ซึ่งจะเกิดขึ้นเมื่อมีการเปลี่ยนการทำงาน จากสถานะนำกระแสเข้าสู่สถานะหยุดนำกระแสได้ด้วย บางครั้งการแลตซ์นี้อาจเกิดขึ้นได้แม้ว่ากระแสตรง ขณะนำกระแส ยังมีค่าต่ำกว่าค่า  $I_{dm}$  ก็ตามทั้งนี้ เพราะเมื่อ IGBT เริ่มหยุดนำกระแส กระแสตรงจะตกลงอย่างรวดเร็วรอยต่อ J2 จะต้องรับแรงดันย้อนกลับที่เพิ่มขึ้นอย่างรวดเร็วเช่นกัน

ผลที่เกิดขึ้นจะทำให้บริเวณปลอดพาหะขยายบริเวณขึ้นบอดี้ P โดยเฉพาะจะขยายบริเวณ  $n^-$  มากกว่า เพราะมีความหนาแน่นของการโด๊ปต่ำกว่า การขยายบริเวณปลอดพาหะอย่างรวดเร็ว จะทำให้โฮลที่ค้างอยู่ในบริเวณ  $n^-$  ขณะนำกระแสและยังไม่ได้รวมกับอิเล็กตรอนหลุดรอดจากการจัดวางของบริเวณรอยต่อพาหะ เข้าไปสะสมอยู่ในบริเวณรอยต่อ J2 เป็นการเพิ่มกระแสที่ไหลผ่านตัวต้านทานข้างเคียงให้สูงขึ้น ทำให้ไทรสเตอร์ภายใน IGBT เกิดการแลตซ์ขึ้นได้ เมื่อเกิดการแลตซ์ขึ้นแรงดันตกคร่อมขาซอร์สและเดรนขณะนำกระแส จะมีค่าต่ำกว่าระดับปกติ นอกจากนี้การแลตซ์ยังสามารถเกิดขึ้นได้อีก ในขณะที่กระแสเดรนยังมีค่าต่ำกว่า  $I_{dm}$  อยู่ได้เช่นกัน

### การป้องกันการแลตซ์

การหลีกเลี่ยงการแลตซ์ของ IGBT สามารถทำได้ทั้งผู้ผลิตและใช้งานเอง ผู้ผลิตอาจจะออกแบบโครงสร้างในส่วนบริเวณบอดี้ p ให้มีความต้านทานข้างเคียงค่าต่ำที่สุด เพื่อจะได้เพิ่มค่าของ  $I_{dm}$  ให้มากที่สุด ซึ่งเป็นการลดโอกาสที่จะเกิดการแลตซ์ลงได้

วิธีแรกอาจทำได้โดยลดความกว้างของบริเวณซอร์ส  $n^+$  ลง นั่นคือลดค่า  $L_s$  ที่แสดงในรูปที่ 2 ลงนั่นเอง

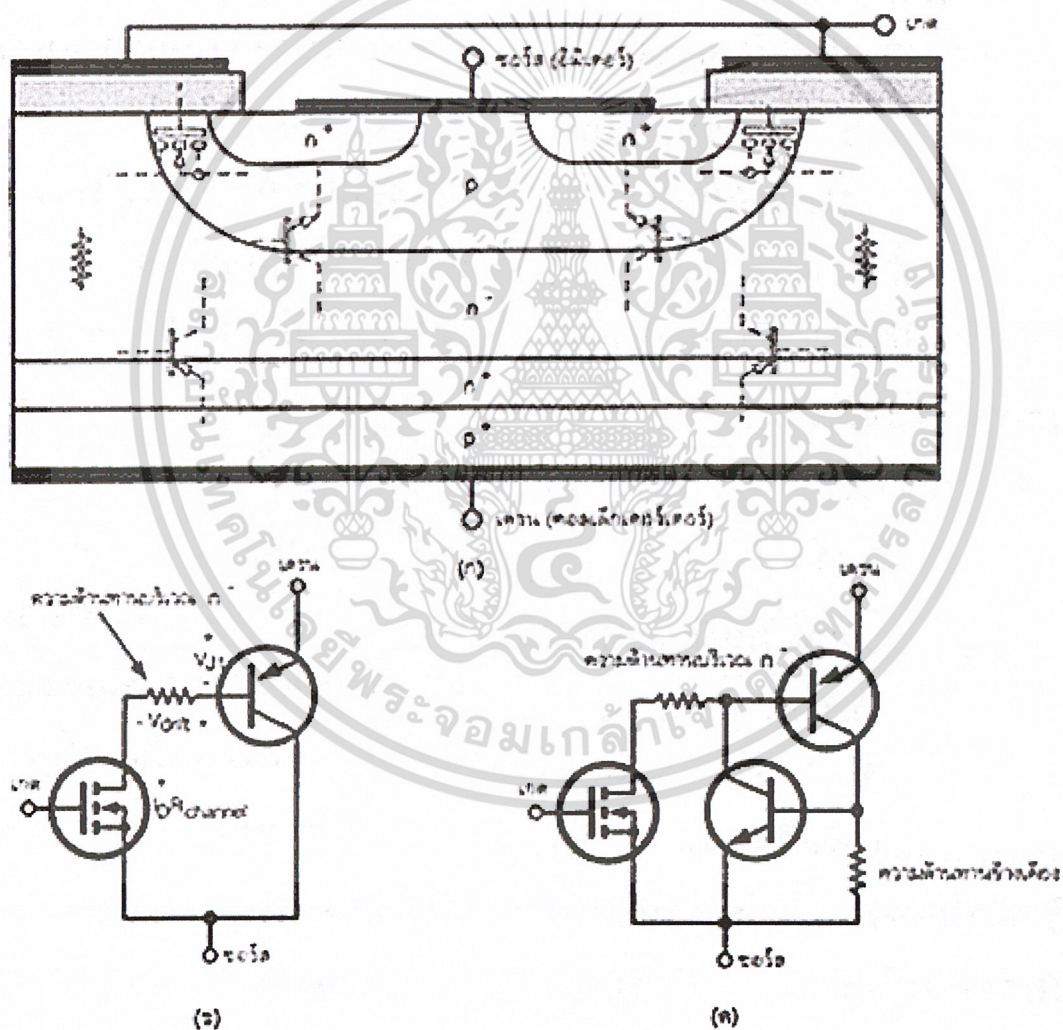


รูปที่ 3.5 โครงสร้างที่ปรับปรุงเพื่อป้องกันการแลตซ์ใน IGBT

วิธีที่สองเป็นการแบ่งระดับความหนาแน่นในการโด๊ปสารของบริเวณบอดี้ p ดังในรูปที่ 5 จะเห็นว่าบริเวณบอดี้ P ภายใต้เกตจะโด๊ปด้วยความหนาแน่นในระดับปกติ  $10^{16} \text{ cm}^{-3}$  และมีความหนาแน่นน้อยกว่าของบริเวณซอร์ส  $n^+$  แต่ส่วนอื่นที่เหลือของบริเวณบอดี้ p จะโด๊ปด้วยความหนาแน่นที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากกว่าคือ  $10^{19} \text{ cm}^{-3}$  รวมถึงความหนาแน่นจะมากกว่า ด้วยการทำให้เพิ่มความสารนในการนำกระแสให้สูงขึ้น เป็นการลดความต้านทานข้างเคียงให้น้อยลงได้ สำหรับผู้ใช้งานก็สามารถป้องกันการแลตช์ในโหมดสแตติกได้ โดยออกแบบไม่ให้กระแสที่ไหลในโหนดไหลเกินค่ากระแส  $I_{dm}$  และป้องกันการแลตช์ในโหมดไดนามิกได้ โดยหน่วงเวลาขณะหยุดนำกระแสให้ยาวนานขึ้นเพื่อให้โหนดที่ยังค้างอยู่ใน บริเวณ n- มีเวลาพอที่จะรวมกับอิเล็กตรอน เป็นการลดกระแสที่ไหลผ่านความต้านทานข้างเคียงให้น้อยลงได้ การหน่วงเวลาขณะหยุดนำกระแสให้ยาวนานขึ้น ทำได้โดยเพิ่มความต้านทานภายนอกอนุกรมเข้ากับขาเกตของ IGBT

### วงจรสมมูลของ IGBT



รูปที่ 3.6 (ก) แสดงโครงสร้างที่มีทรานซิสเตอร์และมอสเฟตแฝงอยู่ใน

(ข) วงจรสมมูลสำหรับการทำงานสภาพปกติของ IGBT

(ค) วงจรสมมูลที่แสดงส่วนของไทรสเตอร์ที่แฝงอยู่ใน IGBT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรมุมของ IGBT แสดงไว้ในรูปที่ 3.6 ซึ่งในรูปที่ 3.6 (ก) นั้นจะเห็นว่าในบริเวณบอดี้ p ชั้นบริเวณ n- และชั้นอินเจกต์ติ้ง p+ จะคล้ายกับทรานซิสเตอร์ชนิดพีเอ็นพี โดยแทนได้ด้วยขา คอลเล็กเตอร์, เบส และอิมิตเตอร์ ตามลำดับ และบริเวณภายใต้เกตก็จะแทนได้ด้วยมอสเฟตซึ่งจะมีความต้านทานบริเวณ n- เชื่อมขาเบส ของทรานซิสเตอร์พีเอ็นพีเข้ากับขาเกรนของมอสเฟต ซึ่งเมื่อเขียนวงจรมุมออกมาจะได้วงจรงรูปที่ 3.6 (ข)

จากรูปที่ 3.6 (ข) จะเห็นว่าเป็นวงจรรีจิสเตอร์ โดยมีมอสเฟตเป็นตัวขับทรานซิสเตอร์พีเอ็นพี แต่มีจุดพิเศษที่แตกต่างจากวงจรทั่วไป คือกระแสเกรนส่วนใหญ่จะไหลจากอิมิตเตอร์มายังเบส ผ่านความต้านทานบริเวณลอยเลื่อน และผ่านขาเกรนมายังคอลเล็กเตอร์และขาซอร์สสำหรับวงจรมุมในรูปที่ 3.6 (ค) จะแสดงให้เห็นว่าภายใน IGBT มีไทรสเตอร์แฝงอยู่ด้วย โดยดูได้จาก การที่ ทรานซิสเตอร์เอ็นพีเอ็นและพีเอ็นพีต่อเข้าด้วยกันในลักษณะที่มีการป้อนกลับ ทำให้เห็นได้ชัดถึง เหตุ ที่ทำให้เกิดการแลตซ์ของ IGBT

โดยถ้ากระแสส่วนน้อยที่ไหลผ่านจากอิมิตเตอร์มายังคอลเล็กเตอร์ของทรานซิสเตอร์ของ ทรานซิสเตอร์พีเอ็นพี ผ่านความต้านทานข้างเคียงแล้วทำให้เกิดแรงดันตกคร่อม ความต้านทานสูง กว่า 0.7 โวลต์ ทรานซิสเตอร์เอ็นพีเอ็นจะนำกระแส ส่งผลให้เกิดการแลตซ์ขึ้นใน IGBT สำหรับ แรงดันตกคร่อมขาเกรนและซอร์ส ของ IGBT ขณะนำกระแส ( $V_{ds(on)}$ ) สามารถเขียนเป็นสมการ ได้ดังนี้

$$(V_{ds(on)}) = V_{j1} + A_{drift} - I_{dRchannel} \dots (1)$$

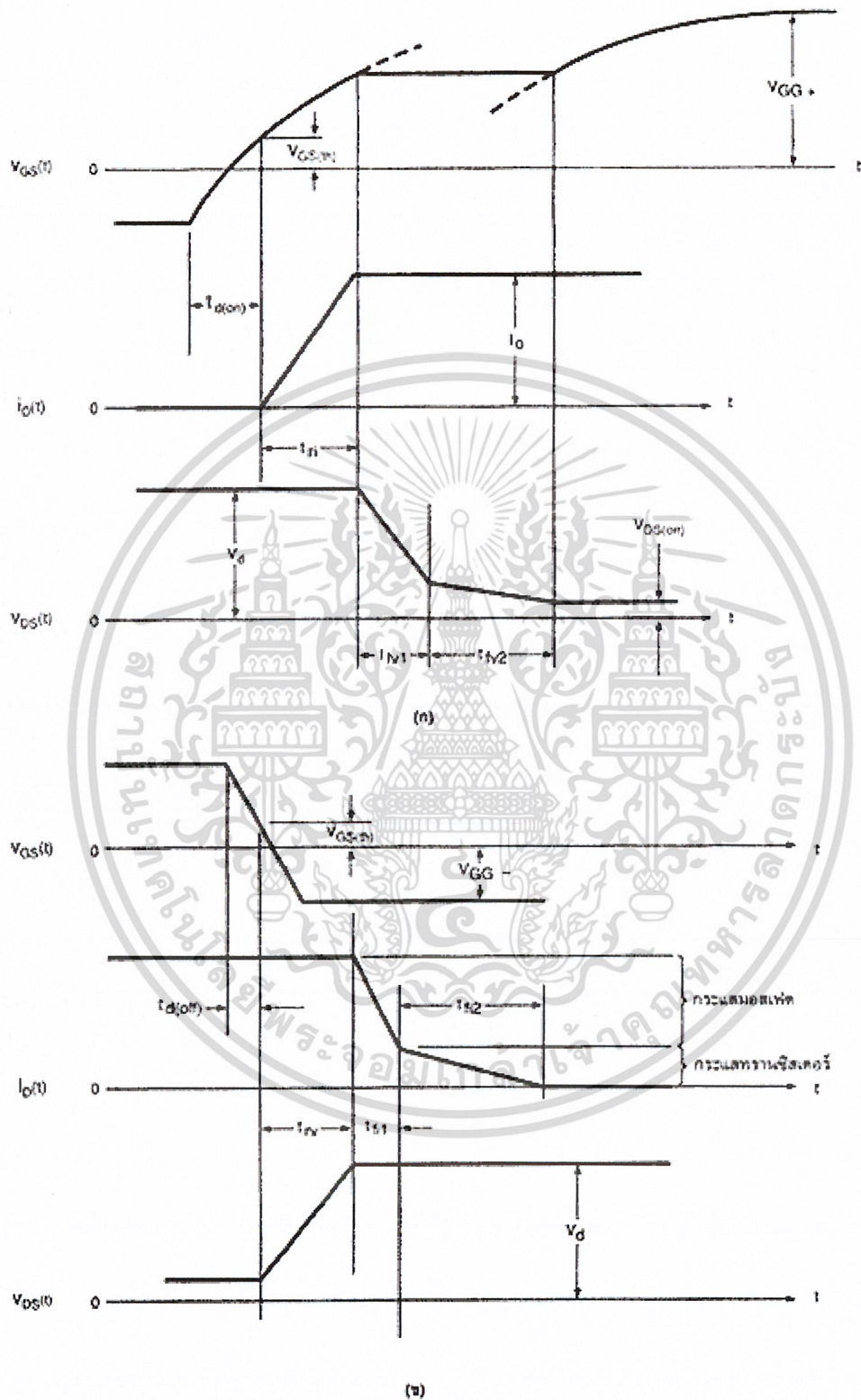
เมื่อ  $V_{j1}$  เป็นแรงดันไบแอสตรงที่ตรงคร่อมรอยต่อพีเอ็น จึงมีค่าค่อนข้างคงที่ จะมีการเปลี่ยนแปลงบ้างก็เพียงเล็กน้อยเพราะมีความสัมพันธ์โดยตรงในลักษณะเอ็กซ์โปเนนเชียล กับ กระแส ทำให้มีค่าอยู่ระหว่าง 0.7 - 1 โวลต์

$A_{drift}$  เป็นแรงดันที่ตกคร่อมความต้านทานบริเวณลอยเลื่อน ซึ่งความต้านทานนี้มีค่าค่อนข้างคงที่ แต่เมื่อเทียบกับค่าแรงดันในมอสเฟตแล้วจะมีค่าน้อยกว่าเพราะผลของการมอดูเลต สภาพนำที่ เกิดขึ้นใน IGBT

$R_{channel}$  เป็นค่าความต้านทานในย่าน 1 - 1,000 โอห์ม มีค่าค่อนข้างคงที่ ( $I_{dRchannel}$ ) เป็น แรงดันตกคร่อมมอสเฟต

ดังนั้นจึงพอจะสรุปได้ว่า ( $V_{ds(on)}$ ) จะมีค่าสูงมากขึ้นตามค่ากระแสเกรนที่สูงขึ้น โดยทั่วไป IGBT จะสามารถทำงานได้ในอุณหภูมิรอยต่อสูงสุดถึง 150 องศาเซลเซียส และผลของการเปลี่ยน อุณหภูมิห้อง ไปถึงค่าสูงสุดนี้ จะส่งผลให้เกิดการเปลี่ยนแปลงค่า ( $V_{ds(on)}$ ) เพียงเล็กน้อยเท่านั้น เพราะ IGBT มีค่า ( $V_{ds(on)}$ ) เป็นผลรวมระหว่างแรงดันตกคร่อมมอสเฟตที่มีสัมประสิทธิ์ทาง อุณหภูมิเป็นบวก (หมายถึงอุณหภูมิสูงขึ้นแรงดันตกคร่อมก็จะสูงขึ้นตาม) กับแรงดันตกคร่อมความ ต้านทาน บริเวณลอยเลื่อนที่มีสัมประสิทธิ์อุณหภูมิลบ

## ลักษณะการสวิตช์



รูปที่ 3.7 (ก) ลักษณะของกระแสและแรงดันขณะนำกระแส

(ข) ลักษณะของกระแสและแรงดันขณะหยุดนำกระแส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

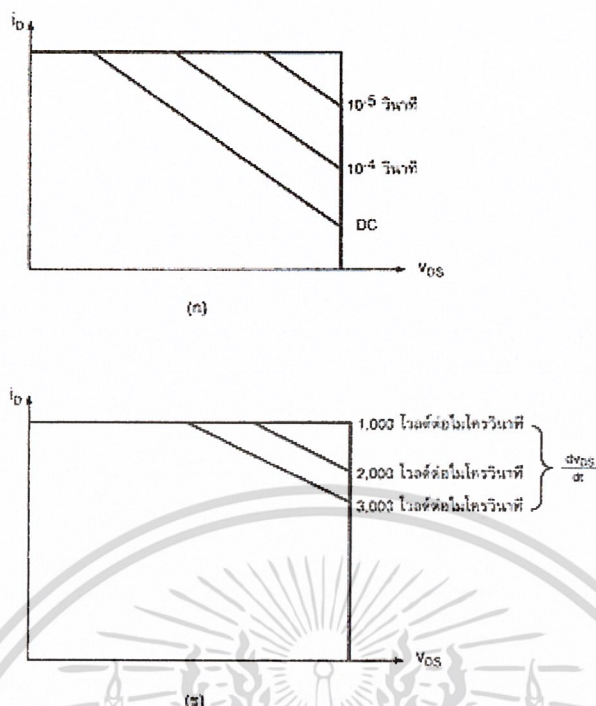
ลักษณะของสัญญาณกระแสและแรงดันในช่วงเวลาที่เกิดการนำกระแสและหยุดนำกระแส แสดงไว้ในรูปที่ 7 โดยช่วงเวลาในการนำกระแสของ IGBT แสดงไว้ในรูปที่ 3.7 (ก) ซึ่งจะมีลักษณะ คล้ายกับการนำกระแสของมอสเฟต คือจะมีเวลาก่อนการนำกระแส ( $T_d(on)$ ) นับตั้งแต่วันที่แรงดัน ระหว่างเกตกับซอร์สอยู่ในช่วง  $V_{gg-}$  จนถึง ( $V_{gs(th)}$ ) ความจริงแล้วการป้อนแรงดันนี้จะมีลักษณะ การเปลี่ยนแปลงทันทีทันใดจากค่า  $V_{gg-}$  เป็น  $V_{gg+}$  แต่กลับมีลักษณะเป็นเอ็กซ์โปเนนเชียล

เหตุที่เป็นเช่นนั้นเนื่องจากผลการชาร์จประจุของตัวเก็บประจุระหว่างเกตกับซอร์สและเกตกับ เดรนภายใน IGBT แรงดันที่ขาเดรนจะยังคงอยู่ในช่วงเวลาที่ขาขึ้น ( $T_{ri}$ ) หรือในช่วงเวลาที่กระแส เดรนยังไม่ถึงค่ากระแสทำงาน ( $I_o$ ) หลังจากนั้นกระแสเดรนก็จะคงที่ แต่แรงดันจะตกลงสู่ค่า ( $V_{ds(on)}$ ) โดยแบ่งช่วงเวลาลงเป็นสองช่วง คือช่วง  $T_{fv1}$  เป็นช่วงที่ทำงานอยู่ในย่านความต้านทานสูง ( $R_{channel}$ ) ส่วน  $T_{fv2}$  ช่วงที่ทำงานอยู่ในย่านความต้านทานต่ำ ( $R_{channel}$ )

ในรูปที่ 3.7 (ข) จะเป็นรูปแสดงลักษณะของกระแสและแรงดันในช่วงเวลาที่ IGBT หยุด นำกระแส จะเห็นว่ากระแสเดรนจะยังคงที่อยู่ตลอดช่วงเวลาที่แรงดันขาเดรนเพิ่มขึ้น และมีช่วงเวลา ลง ของกระแสเดรนที่แตกต่างชัดเจนสองช่วง โดยช่วงแรก  $T_{fv1}$  จะเป็นช่วงหยุดนำกระแสของมอส เฟตภายใน IGBT และช่วง  $T_{fv2}$  จะเป็นช่วงหยุดนำกระแสของทรานซิสเตอร์พีเอ็นพี ซึ่งจะช้ากว่า มอสเฟต ทำให้ช่วงเวลานี้นานกว่าช่วงแรก และมีการสูญเสียกำลังงานมากในช่วงนี้

### พื้นที่การทำงานที่ปลอดภัย

IGBT มีพื้นที่การทำงานที่ปลอดภัยทั้งในระหว่างนำกระแสและหยุดนำกระแส โดยมีพื้นที่การ ทำงานปลอดภัยในขณะไบแอสของ (forward bias safe operating area : FBSOA) ที่กว้างมาก เปรียบเทียบได้กับเกือบเป็นสี่เหลี่ยมสำหรับเวลาในการสวิตช์ที่สั้น ๆ แต่จะแคบลงเมื่อเวลาในการ สวิตช์ยาวนานขึ้น ซึ่งถ้าเทียบกับเพาเวอร์มอสเฟตแล้ว IGBT จะทำงานได้ในช่วงพื้นที่ที่กว้างกว่า เมื่อเวลาในการสวิตช์เท่ากัน



รูปที่ 3.8 (ก) พื้นที่การทำงานที่ปลอดภัยในสภาวะไบแอสตรง  
(ข) พื้นที่การทำงานที่ปลอดภัยในสภาวะไบแอสกลับ

ในช่วงระหว่างเริ่มนำกระแสและขณะที่นำกระแสแล้ว จุดการทำงานของ IGBT จะต้องมีความหนาแน่นกระแสและกระแสที่ขาแคโรนอยู่ในพื้นที่การทำงานที่ปลอดภัยในช่วงไบแอสตรง เสมอ ดังแสดงใน รูปที่ 3.8 (ก) หากไม่เช่นนั้นแล้วจะทำให้เกิดการเสียหายขึ้นที่ IGBT จากรูปพื้นที่การทำงานที่ปลอดภัยของ IGBT นี้ จะแสดงถึงขีดจำกัดของกระแสแคโรน อัตราหนาแน่นกระแสแคโรน และอุณหภูมิรอยต่อของ IGBT ตามลำดับ

สำหรับพื้นที่การทำงานปลอดภัยในช่วงไบแอสกลับ (reverse bias safe operating area : RBSOA) จะแตกต่างจากค่าจำกัดของค่าอัตราการเปลี่ยนแปลงแรงดัน ที่ขาแคโรนต่อเวลา ( $DV_{ds}/DT$ ) ซึ่งจะเกิดขึ้นในช่วงระหว่างหยุดนำกระแส แทนขีดจำกัดทางด้านอุณหภูมิรอยต่อ และ จะมีพื้นที่แคบลงถ้า  $DV_{ds}/DT$  มีค่าสูงมากขึ้น ส่วนเหตุผลที่ถูกจำกัดโดยค่านี้เพราะไม่ต้องการให้เกิดการแลตซ์ขึ้นที่ IGBT~

ค่า  $DV_{ds}/DT$  นี้จะมีผลโดยตรงกับช่วงเวลาหยุดนำกระแส หมายความว่า ถ้ามีอัตราการเปลี่ยนแปลงเร็วจะทำให้ช่วงเวลาหยุดนำกระแสสั้นลง แต่ก็ยังถือว่าขีดจำกัด  $DV_{ds}/DT$  ของ IGBT มีค่าสูงมาก เมื่อเทียบกับอุปกรณ์ไทรสเตอร์ตัวอื่น ๆ ดังนั้นความจำเป็น ในการใช้วงจรถับเบอร์ดเพื่อป้องกันการแลตซ์ก็ไม่มีควมจำเป็นต้องใช้ และการควบคุมค่า  $DV_{ds}/DT$  ที่เกิดขึ้นยังทำได้ง่ายขึ้นด้วยการออกแบบวงจรจับเบอร์ดที่มีค่าความต้านทานที่ต่อกับขาเกตและค่า  $V_{gg-}$  ที่เหมาะสม

## บทที่ 4

## ทฤษฎี Analog To Digital Converter

## บทนำ

สัญญาณที่ใช้ในอุปกรณ์อิเล็กทรอนิกส์ มี 2 ชนิด คือ สัญญาณอนาล็อก และสัญญาณดิจิทัล สัญญาณอนาล็อกจะใช้ในอุปกรณ์ต่างๆ ไปและใช้ในการควบคุมแบบเก่า

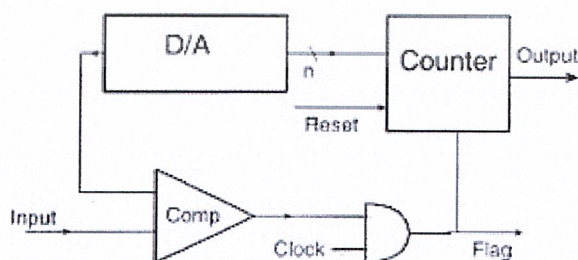
ในปัจจุบันมีไมโครโปรเซสเซอร์ และไมโครคอนโทรลเลอร์ เข้ามาช่วยในการควบคุมอุปกรณ์ต่างๆ มากมาย ซึ่งทำให้การควบคุมนั้นทำได้ง่าย และรวดเร็วยิ่งขึ้น แต่ในการควบคุมนั้น เราจำเป็นต้องใช้สัญญาณดิจิทัลในการติดต่อกับไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์ แต่ในความเป็นจริงนั้น เราใช้สัญญาณอนาล็อกในการควบคุม ดังนั้นเราจึงจำเป็นต้องมีการเปลี่ยนสัญญาณอนาล็อก เป็นสัญญาณดิจิทัล แล้วจึงนำสัญญาณนั้นเข้ามาสู่ไมโครโปรเซสเซอร์ หรือไมโครคอนโทรลเลอร์ เพื่อใช้ควบคุมระบบต่อไป

แม้ว่าสัญญาณอนาล็อกนั้นมีความแน่นอน และแม่นยำสูง แต่สัญญาณอนาล็อกนั้นก็ควบคุมได้ยาก เนื่องจากในสภาพแวดล้อม มีสัญญาณรบกวนอยู่มาก และการที่จะทำให้ การควบคุมแบบอนาล็อก มีความสามารถควบคุม เท่ากับการควบคุมแบบดิจิทัลนั้น ทำได้ยาก เนื่องจากวงจรควบคุมแบบอนาล็อกจะต้องมีความซับซ้อนสูง

อย่างไรก็ตาม สัญญาณดิจิทัลก็ไม่สามารถทดแทนความละเอียดของสัญญาณอนาล็อกได้อย่างสมบูรณ์ แต่ทำให้การควบคุมนั้นทำได้ง่าย และสะดวกยิ่งขึ้น

## Counting Converter

Counting Converter เป็นวิธีที่ง่ายที่สุดของการแปลงสัญญาณอนาล็อก เป็นสัญญาณดิจิทัล โดยใช้อัลกอริทึม การนับค่าเพิ่มขึ้นเรื่อยๆ แล้วนำผลที่ได้จากการนับไปเปรียบเทียบกับค่าที่ต้องการที่ตั้งไว้ ลักษณะการทำงานเป็นดังรูป



รูปที่ 4.1 Analog To Digital Converter

จากวงจร Counter เป็นอุปกรณ์นับค่าที่เพิ่มขึ้นทีละหนึ่ง แล้วส่งค่าที่ได้ให้ D/A มีขา Reset รับ

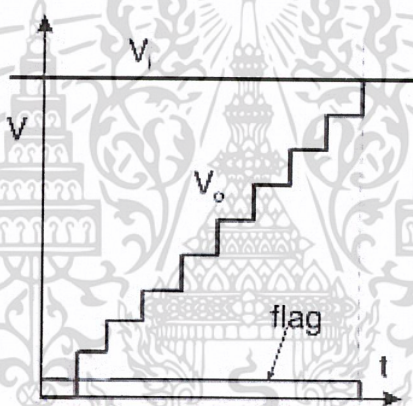
สัญญาณ Reset เมื่อต้องการให้เริ่มนับใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D/A เมื่อรับค่าที่นับเพิ่มขึ้นทีละหนึ่งจากตัวนับ ก็แปลงค่าให้เป็นสัญญาณ อนาล็อกที่มีค่าความต่างศักย์ค่าๆ หนึ่ง แล้วส่งต่อเข้าไปที่อุปกรณ์ตัวเปรียบเทียบ(Comparator)

Comparator จะเป็นอุปกรณ์ตัวเปรียบเทียบค่าความต่างศักย์ ของอินพุต และค่าจากที่ตัวนับ ถ้าหากทั้งสองสัญญาณมีค่าเท่ากันส่งค่าความต่างศักย์ 0 โวลต์ออกมา(ลอจิก 0) ถ้าไม่เท่ากันก็จะส่งความต่างศักย์ที่ไม่ใช่ 0 โวลต์ออกมา(ลอจิก 1)

ซึ่งค่าความต่างศักย์ที่ออกมา จะนำมาเข้าลอจิกเกต "และ" กับ สัญญาณนาฬิกา จะได้ค่าลอจิกออกมา ถ้าผลลัพธ์ออกมาเป็นสัญญาณนาฬิกาแสดงว่ายังไม่ได้ผลลัพธ์เท่าที่ต้องการ สัญญาณนาฬิกาจะไปทำให้ตัวนับนับเพิ่มขึ้นต่อไป และเมื่อได้ค่าผลลัพธ์ดิจิทัลที่ต้องการแล้ว ค่าที่ได้จากตัวเปรียบเทียบจะให้ค่าความต่างศักย์เป็น 0 (ลอจิก 0) ซึ่งเมื่อนำมาเข้าลอจิกเกต "และ" กับสัญญาณนาฬิกาแล้ว ก็จะให้ลอจิก 0 ซึ่งทำให้ตัวนับไม่นับเพิ่มอีก ก็จะได้ค่าดิจิทัลจากตัวนับที่ต้องการ จากคำอธิบายข้างต้นจะได้กราฟของ  $V_0$  ดังนี้



รูปที่ 4.2 Output voltage Graph from A/D converter

ข้อเสียของวิธีนี้ คือ การนับต้องเริ่มนับที่ 0 เสมอ และนับเพิ่มขึ้นเรื่อยๆ ทำให้ช้า เอาท์พุทที่ได้จะมี delay จึงไม่ค่อยนิยมใช้เท่าที่ควร จึงได้เปลี่ยนตัวนับเป็นแบบนับลงได้ด้วย ซึ่งจะอ้างอิงระดับจากระดับเก่า ทำให้ไม่จำเป็นต้องนับ 0 ใหม่ เมื่อมีการเปลี่ยนอินพุตใหม่ แต่ให้อ้างอิงกับผลลัพธ์เดิม ทำให้ได้ผลลัพธ์เร็วขึ้น

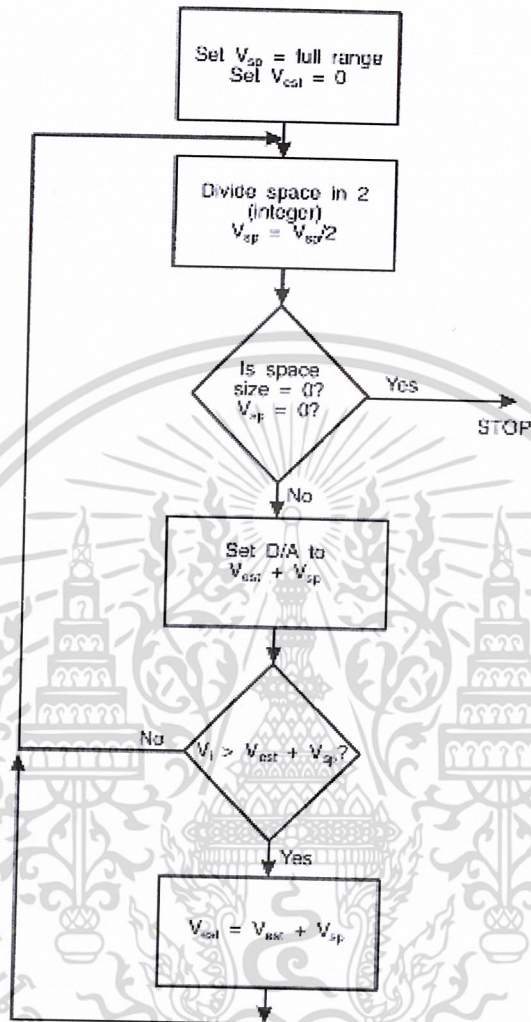
### Successive Approximation

ใช้หลักการของ "binary search" ในการหาคำตอบ โดยนำค่าผลลัพธ์มาเปรียบเทียบกับค่ากึ่งกลางของช่วง เพื่อให้ทราบว่า ค่านั้นๆ มากกว่า หรือน้อยกว่า โดยจะปรับช่วงให้แคบลงมาเรื่อยๆ แล้วเปรียบเทียบผลลัพธ์กับค่ากึ่งกลางของช่วง ไปเรื่อยๆ จนได้ผลลัพธ์ที่ต้องการ เช่น เลขที่เป็นคำตอบคือ 3 จากช่วงของคำตอบที่ 0-7 ครั้งแรกเอาค่า  $(0+7)/2 = 4$  มาเปรียบเทียบ ได้ผลว่า คำตอบที่ต้องการอยู่ในช่วงที่น้อยกว่า 4 ครั้งที่ 2 ก็เลือกค่า  $(0+4)/2 = 2$  มาเปรียบเทียบ ได้ผลว่าคำตอบที่ต้องการอยู่ในช่วงที่มากกว่า 2 แต่น้อยกว่า 4 ครั้งที่ 3 ก็เลือกค่า  $(2+4)/2 = 3$  มาเปรียบเทียบ ได้ผลว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำตอบที่ต้องการ

จากหลักการที่กล่าวมาอาจเขียน flow chart ได้ดังนี้

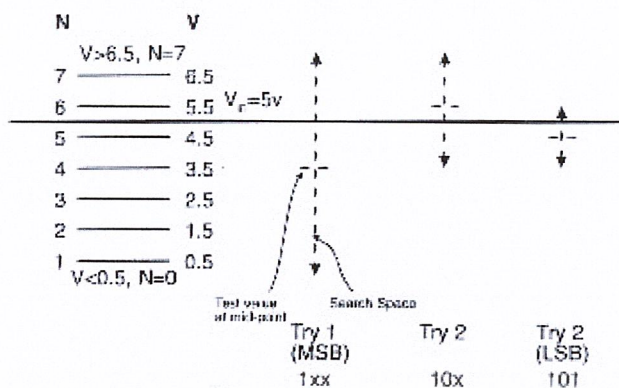


รูปที่ 4.3 Binary Search Strategy

ข้อดีของวิธีนี้ คือ เวลาที่ใช้ในการหาคำตอบ  $n$  รอบแน่นอน (สำหรับ  $n$  bit converter ซึ่งอ้างอิงได้  $2^n$  ระดับ และระดับ  $V_{in}$  ที่คงที่) ซึ่งใช้เวลาน้อยกว่าแบบ "Counting Algorithm"

แต่มีข้อเสีย คือถ้า  $V_{in}$  เปลี่ยนทันทีทันใด ขณะที่กำลังทำ binary search อยู่ นั้น คำตอบที่ได้จะผิดพลาด ตัวอย่างเช่น เปลี่ยน  $V_{in}$  จาก 5 Volt เป็น 2 Volt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 Binary Search

ช่วงของ  $V_{in}$  คือ 1-7 ใช้  $n=3$  (เพราะว่า  $2^3=8$ )

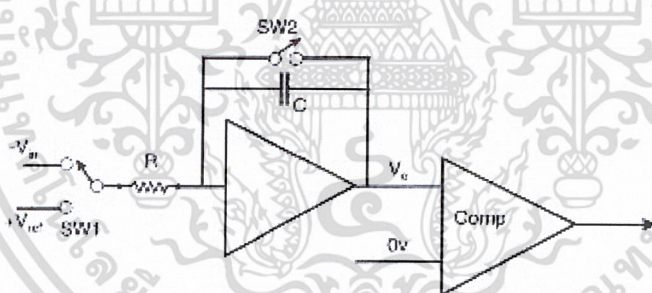
ครั้งแรก ใช้ 4 เปรียบเทียบกับ  $V_{in}$  (ซึ่งเท่ากับ 5 โวลต์) พบว่า อยู่ในช่วง lower ได้ 1xx

ครั้งที่ 2 ใช้ 2 เปรียบเทียบกับ  $V_{in}$  (ซึ่งเท่ากับ 5 โวลต์) พบว่า อยู่ในช่วง upper ได้ 10x

ครั้งที่ 3 ใช้ 3 เปรียบเทียบกับ  $V_{in}$  (ซึ่งเท่ากับ 5 โวลต์) พบว่า ผลลัพธ์ที่ได้จะผิดพลาด ได้ 100

### Dual-Slope ADC

ใช้หลักการของวงจร Integrater ทำงานร่วมกับตัว Comparater ดังรูป



รูปที่ 4.5 Dual-Slope ADC

Input Voltage มี 2 ตัว คือ ค่าความต่างศักย์นาฬิกาที่ต้องการแปลงเป็นดิจิทัล ( $-V_{in}$ ) และ ความต่างศักย์ที่คงที่ค่าหนึ่ง ( $V_{ref}$ ) และมีสวิตช์ SW1 ซึ่งทำหน้าที่เลือกค่าสัญญาณ

จากวงจรตอนเริ่มต้นสวิตช์ SW2 ทำหน้าที่คายประจุของตัวเก็บประจุ C แล้วจึงเปิด SW2 ออก

เมื่อสวิตช์ SW1 สับมาที่  $-V_{in}$  จากวงจร Integrater จะพิสูจน์สมการได้ดังนี้

$$I = C \frac{dV_0}{dt}$$

$$-V_{in} + iR - V_0 + V_0 = 0$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$-V_{in} + RC \frac{dV_0}{dt} = 0$$

$$V_{in} = RC \frac{dV_0}{dt}$$

$$\int dV_0 = \int \frac{V_{in}}{RC} dt$$

$$V_0 = \frac{V_{in}(t)}{RC}$$

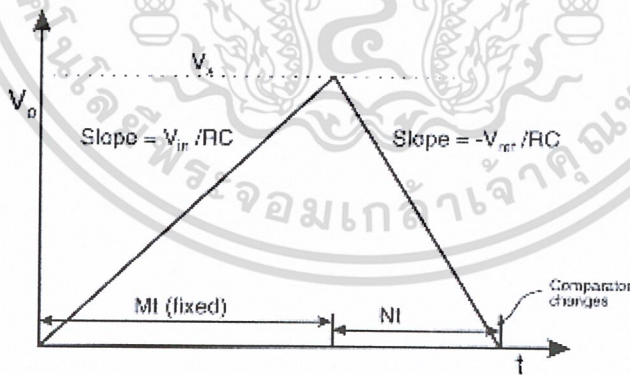
slope มีค่าเท่ากับ  $\frac{V_{in}}{RC}$

ค่า  $t$  ที่ใช้มีค่าคงที่  $t_m$   
 เมื่อ  $t$  เพิ่มจากศูนย์ถึง  $t_m$

$$V_0 = \frac{V_{ref}(t)}{RC}$$

จะได้สมการ  $\frac{V_{ref}}{RC}$   
 slope มีค่า  $\frac{V_{ref}}{RC}$

สมมติ ช่วงเวลาตั้งแต่ความต่างศักย์ที่  $t_m$  จนความต่างศักย์เป็น 0 มีค่าเท่ากับ  $t_n$   
 ได้ดังแสดงในกราฟ



รูปที่ 4.6 Dual Slope A/D Converter Output and Timing

$$V_{in} = V_{ref} \frac{t_n}{t_m}$$

จากหลักของสามเหลี่ยมคล้าย จะได้สมการ

เนื่องจาก  $V_{ref}$  และ  $t_n$  มีค่าคงที่ สัญญาณอนาล็อกขึ้นกับค่า  $t_m$  เพราะการควบคุมการเปลี่ยน

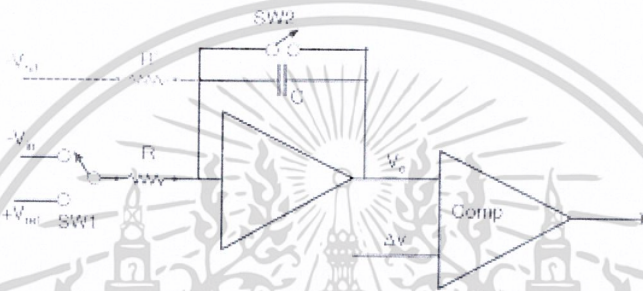
สัญญาณดิจิทัล ที่ขึ้นกับค่า  $t_m$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลงเป็นสัญญาณดิจิทัลจะทำโดยจับคู่ค่า  $t_n$  กับเอาต์พุตค่าๆ หนึ่ง ตามความเหมาะสม สำหรับ  $V_{ref}$  นั้นๆ เหมือนการเทียบค่าในตาราง

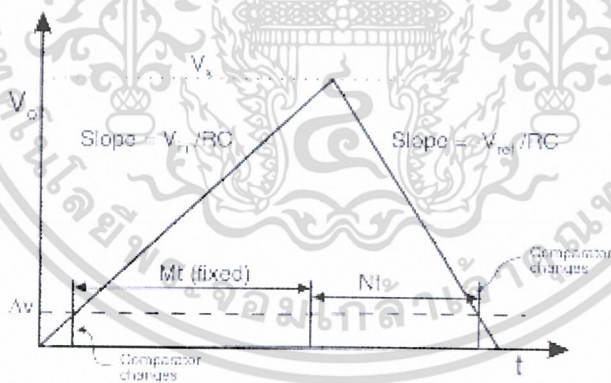
ความเร็วของการแปลงสัญญาณแบบนี้ ขึ้นอยู่กับ  $V_{in}$  และ Slope ของวงจร integrater

โดยธรรมชาติแล้ว ลักษณะของตัวเปรียบเทียบเองนั้น จะไม่เป็นอุดมคติ ก็จะมีผลต่างของความต่างศักย์อยู่  $V$  โวลต์ แม้ว่าจะต่ออินพุตทั้งสองลงกราวด์แล้วก็ตาม ซึ่งถ้า  $V_{ref}$  ที่ใช้อยู่นั้นมีค่าน้อยกว่าค่าผลต่างของความต่างศักย์ที่เกิดจากตัวเปรียบเทียบ ความชันก็จะน้อย ทำให้เวลา  $t_m$  ใช้เวลานานมาก กว่าที่จะฟื้นค่าความต่างศักย์ที่เกิดจากตัวเปรียบเทียบ เราจึงต้องนำค่าความต่างศักย์มาเพิ่มให้กับ  $V_{ref}$  เพื่อหาผลลัพธ์ ดังรูป



รูปที่ 4.7 Dual Slope A/D Converter – Full Circuit

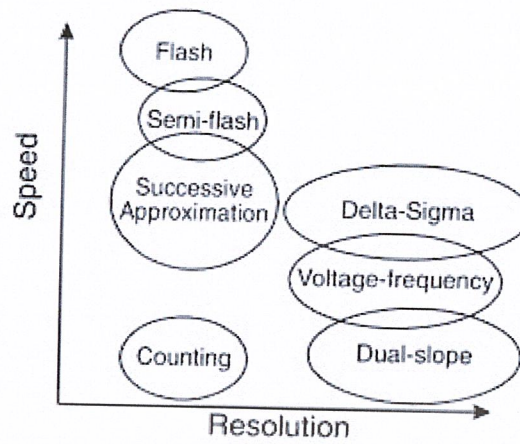
จากวงจรดังกล่าวทำให้ได้กราฟดังรูป



รูปที่ 4.8 Dual Slope A/D Converter – Zero Offset

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 4.10 Summary of A/D Converter

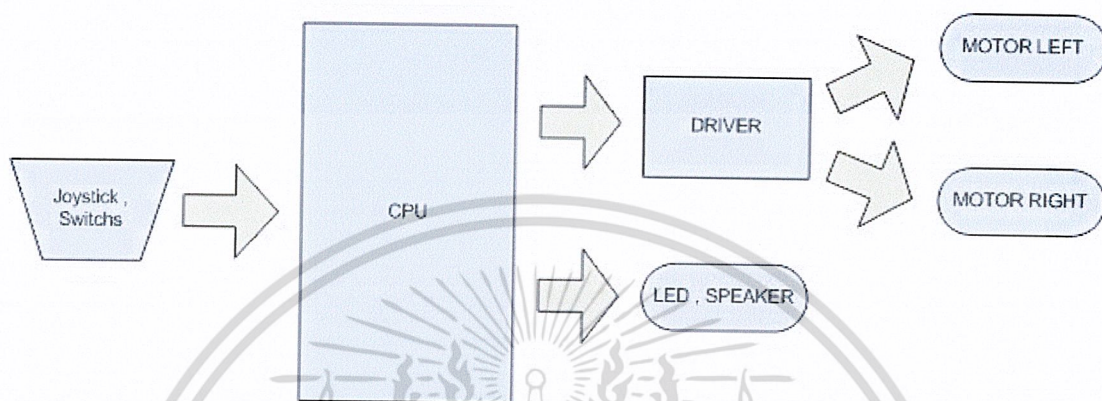


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### ออกแบบวงจร และการทำงาน

#### แผนภาพของโครงการ



รูปที่ 5.1 แผนภาพแสดงการทำงานของโครงการทั้งหมด

จากแผนภาพอธิบายได้ดังนี้ คือ ค่าที่รับจาก INPUT เป็น JOYSTICK และ SWITCH จะเข้ามาที่ไมโครคอนโทรลเลอร์เพื่อประมวลผล จากนั้นจะส่งไปยัง 2 ส่วนคือ ภาค DRIVE เพื่อขับ IGBT ให้มอเตอร์ ซ้ายและขวาหมุน และอีกส่วนที่เหลือจะต่อไปยัง LED เพื่อแสดงผล และลำโพงต่อไป

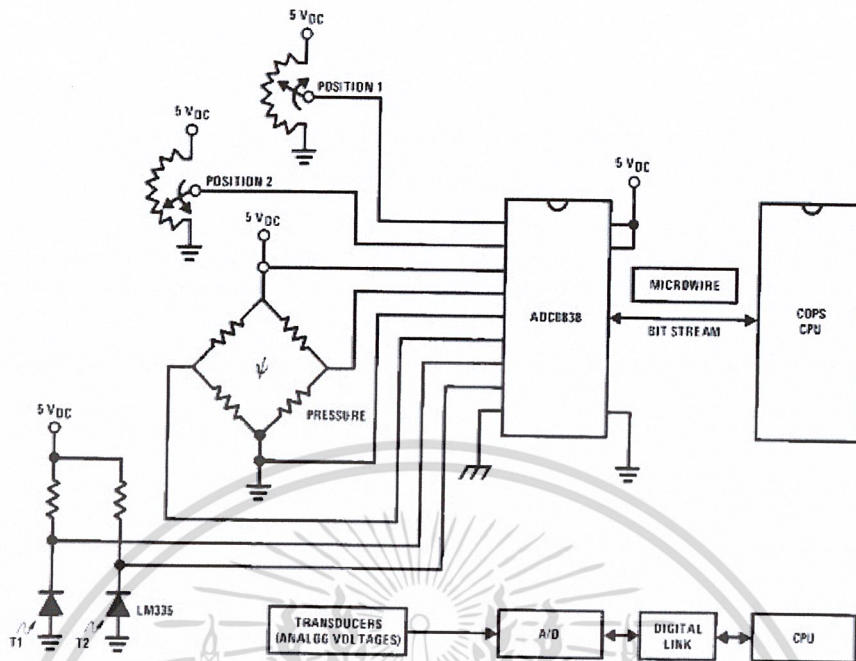
**แบ่งวงจรออกเป็นส่วนย่อยๆได้ 2 ส่วน**

1. ภาคควบคุม
2. ภาค MOTOR DRIVER

ในภาคผนวกจะแสดงวงจรสมบูรณ์ของภาคควบคุม ซึ่งมีหลังการทำงาน และการออกแบบดังนี้คือ

1. ภาคควบคุม จะใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C51 มีคริสตอล Y1 กำหนดจังหวะการทำงานที่ 11.0592MHZ และ C11, C12 ในการลดสัญญาณรบกวน ใช้โปรแกรมการทำงานที่ได้เขียนลงบนไอซี จึงต้องต่อขา EA เข้ากับ VCC ดังรูป สำหรับไอซี U2 เบอร์ DS1232 เป็น IC WATCHDOG ซึ่งหมายความว่า หากไมโครคอนโทรลเลอร์เกิดหยุดทำงาน ไม่ตอบสนองคำสั่งที่เขียนไว้ หรือ “HANG” ไอซี WATCHDOG นี้จะทำการ “รีเซ็ต” ให้ไมโครคอนโทรลเลอร์เริ่มต้นทำงานใหม่นั้นเอง โดยการเช็คสัญญาณที่ขา ALE ซึ่งจะเกิดขึ้นทุกครั้งที่มีการ FETCH คำสั่งการทำงาน หากพบว่าสัญญาณที่ขา ALE หายไปเกินช่วงเวลาหนึ่ง นั้นหมายความว่าไมโครคอนโทรลเลอร์ไม่มีการ FETCH คำสั่ง หรือกระทำตามคำสั่งแล้ว นั้นแปลว่าเกิดการ “HANG” ขึ้นนั่นเอง R30, R31 เป็นตัวต้านทานปรับค่าได้แบบ B ต่ออยู่กับแกนของจอยสติค สำหรับปรับแรงดันจาก 0V ถึง VCC ที่ 5V เพื่อส่งให้กับ INPUT ของไอซีแปลงสัญญาณอนาล็อกไปเป็นดิจิตอล U4 เบอร์ ADC0832 ดังแผนภาพนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

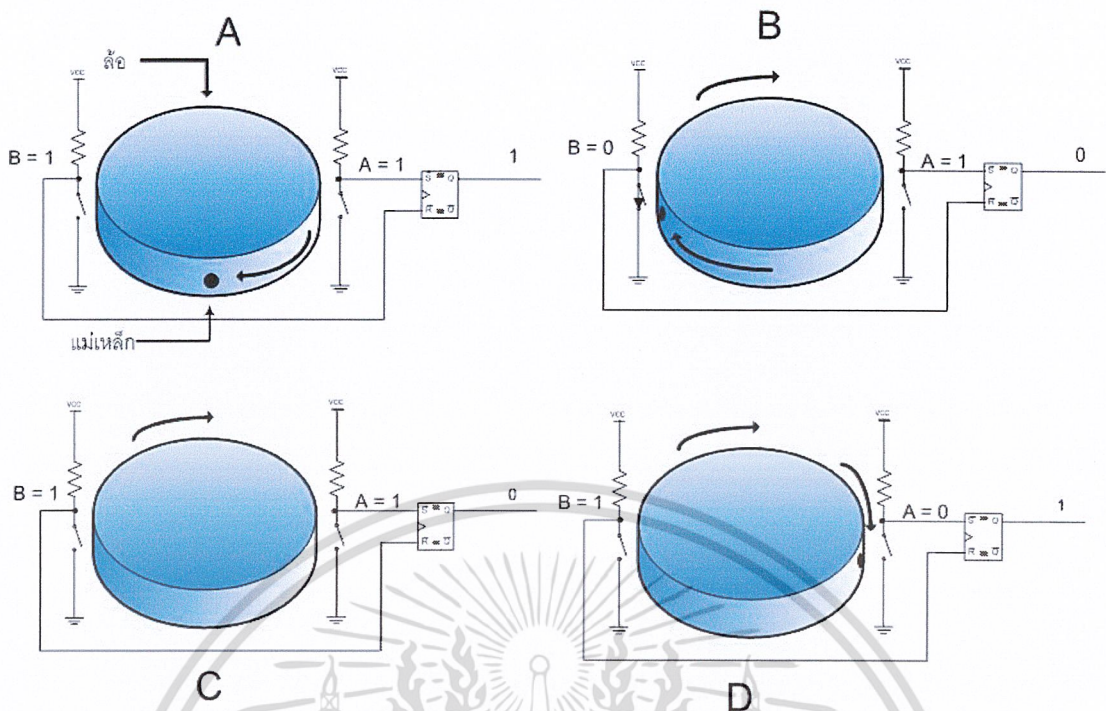


TRI-STATE® is a registered trademark of National Semiconductor Corporation.  
 COPS™ and MICROWIRE™ are trademarks of National Semiconductor Corporation.

รูปที่ 5.2 แสดงการเชื่อมต่อสัญญาณระหว่างไมโครคอนโทรลเลอร์กับ ADC0832

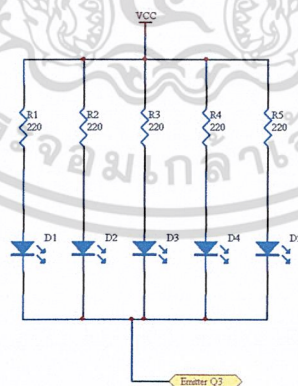
จากภาพจะเห็นว่าไอซี ADC0832 จะรับข้อมูลอนาล็อก เมื่อแปลงเป็นดิจิทัลแล้วจึงส่งให้ CPU ในรูปแบบ BIT STREAM คือแบบอนุกรม การอ่านค่านั้นจะใช้ซอฟต์แวร์ที่มีจังหวะเวลาตามที่ DATA SHEET กำหนด โดยเชื่อมต่อคั้งในรูป R23 และ D18 ต่อเพื่อแสดงผลการทำงานของ ไอซีตัวนี้ เมื่อขา CHIP SELLECT ของไอซี “ACTIVE” ก็จะทำให้ไฟติด S12 , S13 และ S14 เป็นสวิตช์สำหรับ ปิด-เปิด ไฟท้าย , ตั้ระบบขับเคลื่อนออก (override) สำหรับป้องกันมือไป โคนจอยโยกอย่างไม่ตั้งใจ ขณะที่จอดไว้ ส่วนสวิตช์ตัวที่ 3 นั้นต่อไว้ใช้งานในอนาคต โดยจะมี D21 , D22 และ D23 ต่อเพื่อในกรณีที่สวิตช์ตัวใดกด จะมี Logic “0” เกิดขึ้นที่ขา INTO ของ ไมโครคอนโทรลเลอร์เพื่อใช้ในการเขียน โปรแกรมด้านอินเตอร์รัปต์ ในส่วนของ WHEEL COUNTER นั่นคือการนับรอบการหมุนของล้อโดย R25 , R26 , S8 , S10 และ X3A สำหรับล้อ ซ้าย โดยมี R17 , R18 , D7 และ D13 คอยแสดงผล ทำงาน โดย ไอซี S-R LATCH 74LS279 ดังรูป สำหรับล้อทางขวาก็ทำงานในลักษณะเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 แสดงการทำงานของกรนับรอบลือที่หมุน

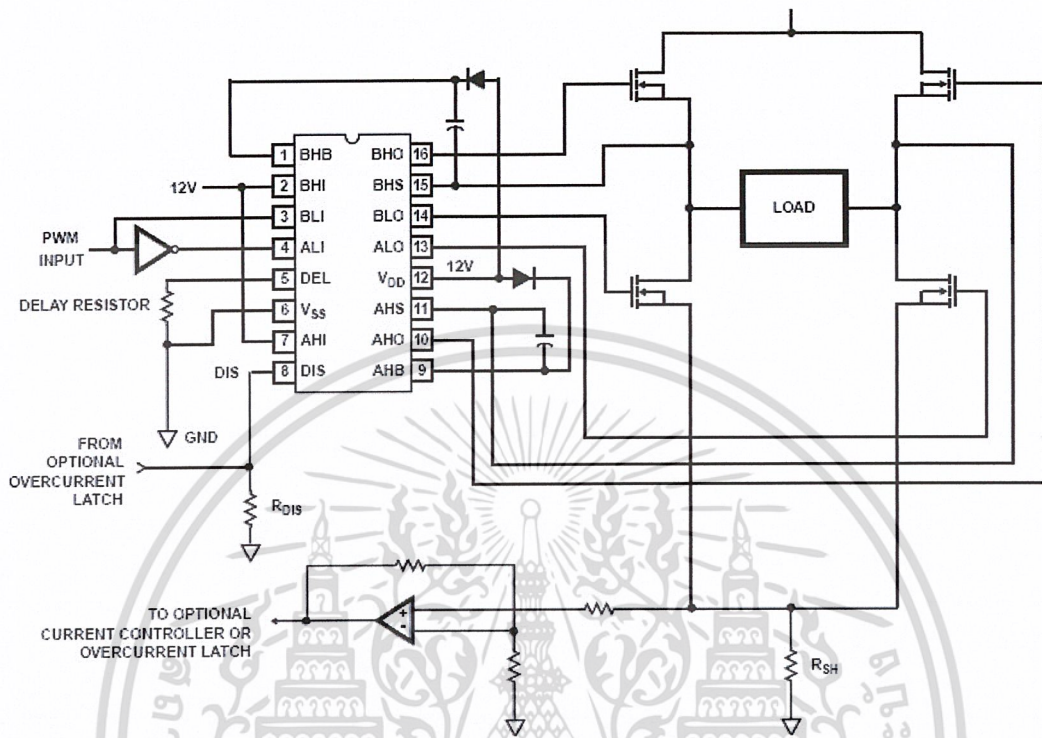
สำหรับ โครงงานจริงนั้น สวิตซ์แม่เหล็กกับแม่เหล็ก ควรต่อกับเกียร์ที่ติดกับมอเตอร์มากที่สุด เพื่อการนับรอบที่ใกล้เคียงจริงมากที่สุด ซึ่งในโครงงานนี้ไม่ได้ต่อสวิตซ์และแม่เหล็กใช้งาน แต่ทำเป็นคอนเน็คเตอร์สำหรับการต่อในอนาคตแทน ไฟท้ายและ SPEAKER คือ Q3 , R32 , Q4 , R33 ตามลำดับ สำหรับ D16 , R21 , D17 , R22 สำหรับแสดงผลการทำงาน คือติดมีมิลลิจิก “0” ในรูปวงจริงนั้น LED DISPLAY ได้ต่อบอร์ดคอนเน็คประสงค์เป็นจรวดรูปข้างล่าง



รูปที่ 5.4 แสดงการเชื่อมต่อไฟท้ายหรือ ไฟแสดงผล

ต่อ LED แบบขนาน โดยมี R 220 โอห์ม อนุกรมอยู่ในแต่ละตัวเพื่อให้แต่ละดวงสว่างเท่าๆกัน เนื่องจากแต่ละดวงมีแรงดันตกคร่อมไม่เท่ากัน สำหรับ ไอซี X4 74HC573 ทำหน้าที่เป็น Buffer ให้ไมโครคอนโทรลเลอร์ช่วยในการจ่ายกระแสแทนนั่นเอง ส่วนของ TO DRIVER นั้นจะเป็นสัญญาณที่ไปยังภาค DRIVER โดยมี TURN\_1 และ TURN\_2 เป็นสัญญาณบอกทิศทาง การเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมุนของมอเตอร์ ตัวช่วยและขาตามลำดับ อีก 4 เส้นที่เหลือต่อไปยังไอซี MOSFET DRIVER เพื่อเป็นสัญญาณ PWM ทำหน้าที่ขับ IGBT ต่อไปดังรูปข้างล่างนี้



รูปที่ 5.5 แสดงการเชื่อมต่อ MOSFET, IGBT กับ HIP4082

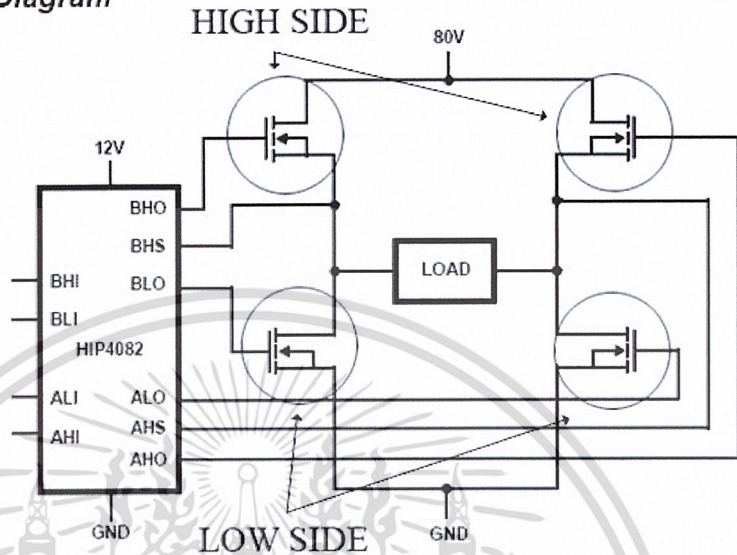
D19 , C14 , C2 , C15 , C3 และ U5 ทำหน้าที่จ่ายไฟ 5V ให้กับภาคควบคุม โดยมี U5 เป็นไอซี REGULATOR 5V เบอร์ 7805 ทำหน้าที่แปลงแรงดันจากแบตเตอรี่ 12 V ให้เหลือ 5V และทำให้เรียบขึ้น C14 , C2 , C15 , C3 ทำหน้าที่กรองสัญญาณรบกวนที่อาจเกิดขึ้น มี D19 ป้องกันการต่อกลับขั้ว และ R24 , D20 เป็นไฟแสดงการทำงานของวงจร C4 , C9 , C10 , C1 , C13 สำหรับ Bypass สัญญาณรบกวนลงกราวด์โดยต่อที่ขา VCC กับขา GND ที่ไอซี AT89C51 , DS1232 , 74LS279 , 74HC573 และ ADC0832 ตามลำดับ ในรูปได้ต่อคอนเน็คเตอร์ J3 , J4 ไว้สำหรับต่อสายไปยังอุปกรณ์ต่างๆที่ติดตั้งบนตัวรถนอกบอร์ดวงจร

2. ภาค MOTOR DRIVER ใช้ไอซี MOSFET DRIVER เบอร์ HIP4082 ของ INTERSIL โดยมีอินพุตอยู่ 4 ขาคือ ALI , BLI สำหรับควบคุม FET LOW SIDE A , B ตามลำดับ AHI , BHI สำหรับควบคุม FET HIDE SIDE A ,B ตามลำดับ รูปวงจรภาค DRIVER โดยสมบูรณ์ อยู่ในภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปข้างล่างเป็นฟังก์ชันการทำงานของ ไอซี

**Application Block Diagram**



รูปที่ 5.6 แสดงฟังก์ชันการทำงานของ ไอซี และตำแหน่ง High Side และ Low Side ของทรานซิสเตอร์

สิ่งสำคัญที่ต้องระวังมากเป็นพิเศษตามที่ DATA SHEET บอกไว้ นั่นคือ ไอซีตัวนี้มีความไวกับไฟฟ้าสถิต หรือ ไฟฟ้าที่เกิดจากการคายประจุ ซึ่ง VCC MAXIMUM ที่ ไอซีตัวนี้จะรับได้คือ 15V จึงได้ต่อ R1 , D1 , R2 , D2 ไว้เพื่อมิให้มีไฟเลี้ยง ไอซีเกินกว่า 12V ซึ่งเป็นค่าที่ ไอซีตัวนี้ยังทำงานได้ โดยซีเนอร์ไดโอดจะมีแรงดันตกคร่อม REVERSE BIAS ที่ 12V และ C1 , C3 , C2 , C8 กรองสัญญาณรบกวนที่อาจเกิดขึ้นทั้งไป D3 , D4 , C4 , C6 , D9 , D10 , C5 , C7 ต่อไว้เพื่อขยระดับแรงดันไฟให้กับ IGBT HIGH SIDE (bootstrap) R3 , R4 ที่ต่อกับขา DEL ของ ไอซี DRIVE เป็นการหน่วงเวลา เพื่อป้องกันการ Short อันเกิดจาก IGBT HIGH SIDE กับ IGBT LOW SIDE “ON” พร้อมกัน สัญญาณ OUTPUT ที่ออกจากไมโครคอนโทรลเลอร์จะมาเข้า ไอซี DRIVE โดยที่ ALI และ BLI จะต่อ NOT GATE อยู่ซึ่งเงื่อนไขการขับ IGBT นั้นมีดังตารางนี้

TRUTH TABLE

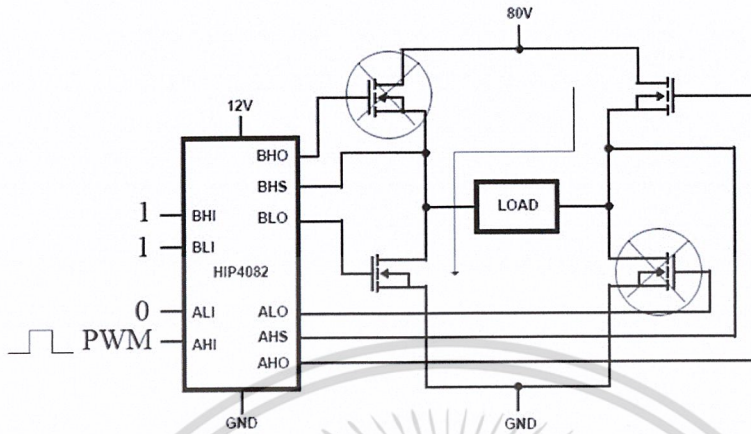
INPUT					OUTPUT	
ALI, BLI	AHI, BHI	VDDUV	VHBUV	DIS	ALO, BLO	AHO, BHO
X	X	X	X	1	0	0
X	X	1	X	X	0	0
0	X	0	1	0	0	0
1	X	0	X	0	1	0
0	1	0	0	0	0	1
0	0	0	0	0	0	0

NOTE: X signifies that input can be either a "1" or "0".

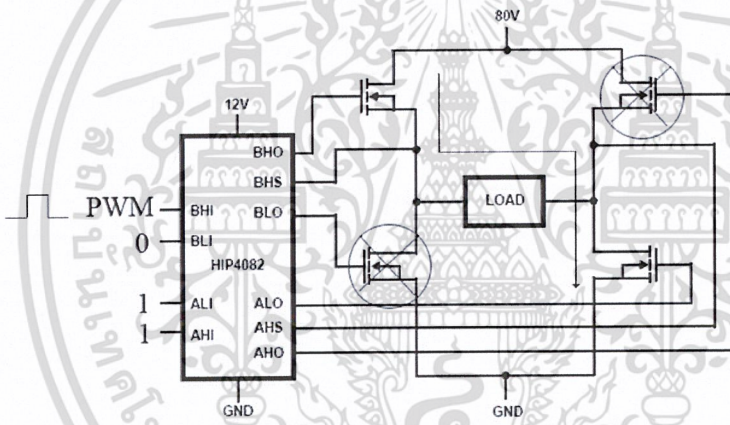
**ตารางที่ 5.1 แสดงสัญญาณควบคุม HIP4082**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

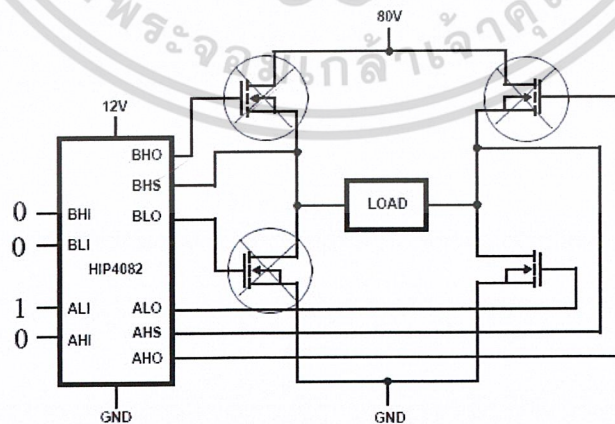
หลักการการทำงานของ การเขียน โปรแกรมเพื่อให้มอเตอร์หมุนนั้น มีดังรูปในหน้าถัดไป



รูปที่ 5.7 แสดงการทำงานของ ไอซีที่ทำให้ มอเตอร์หมุนทางซ้าย



รูปที่ 5.8 แสดงการทำงานของ ไอซีที่ทำให้ มอเตอร์หมุนทางขวา



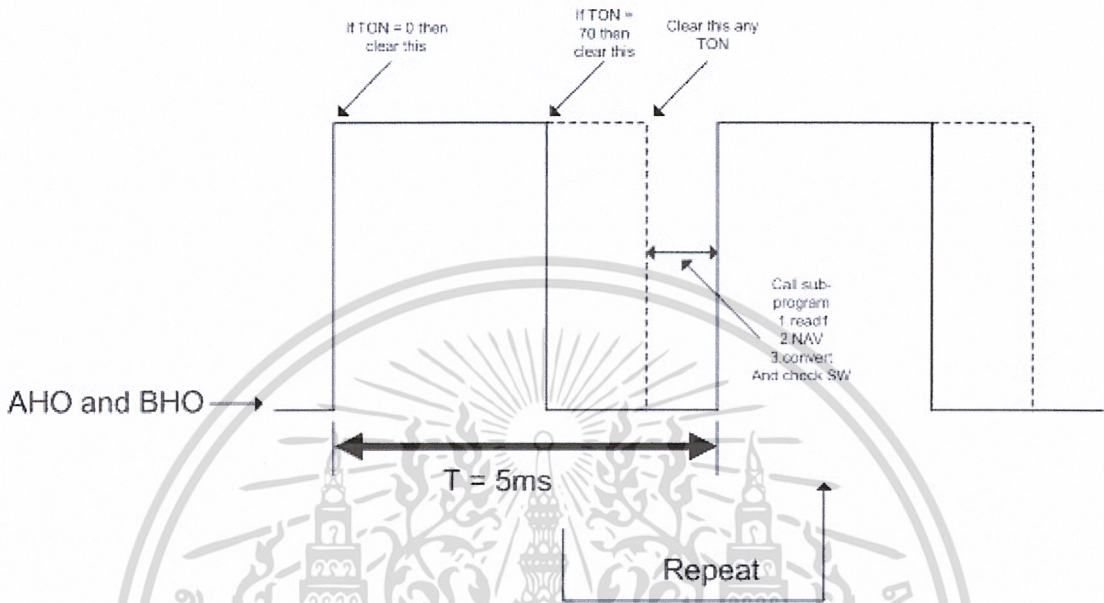
รูปที่ 5.9 แสดงการทำงานของ ไอซีที่ทำให้ มอเตอร์หยุดหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 6

### ซอร์ฟแวร์ และการทำงาน

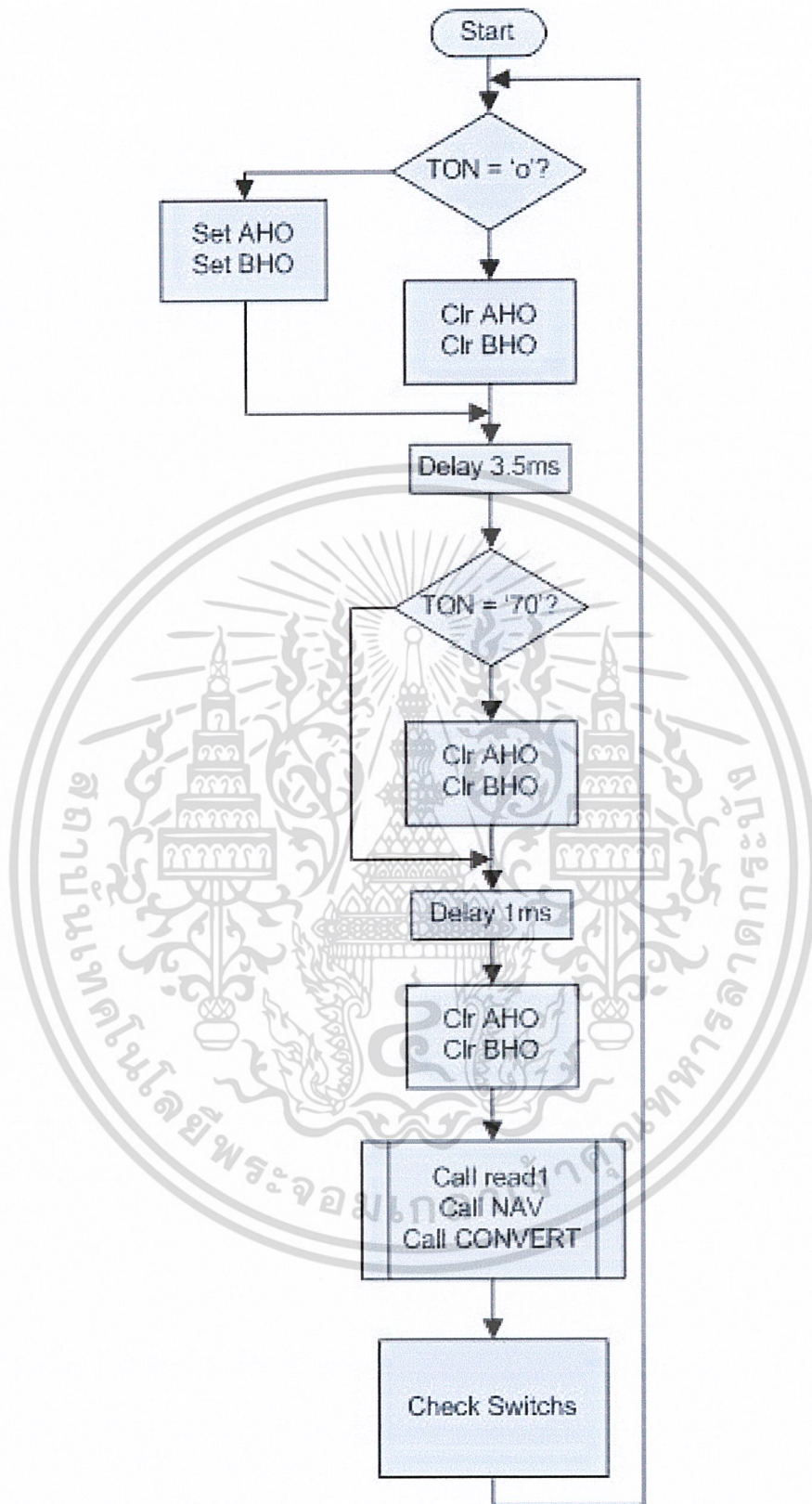
#### มีการทำงานดังนี้



รูปที่ 6.1 แสดงสัญญาณที่ขา Gate ของ IGBT

จากรูปจะเห็นว่าซอร์ฟแวร์มีการทำงานง่ายๆคือ เป็นการกำเนิดสัญญาณ duty cycle ที่ 0% 70% และ 90% ความถี่ที่ 200Hz โดยที่จุดเริ่มต้นของสัญญาณจะเช็คตัวแปร TON โดยหากเท่ากับ “0” หมายความว่าให้ผลิต duty cycle ที่ 0% ก็จะ CLEAR BIT หากไม่เท่ากับ “0” ก็จะ SET BIT จากนั้นจะหน่วงเวลาจนถึงจุดที่สอง ซึ่งจะเช็คค่า TON เท่ากับ “70” หรือไม่ หากเท่าก็จะทำการ CLEAR BIT หากไม่เท่าก็จะต่อไปยังจุดที่สาม ซึ่งจะ CLEAR BIT ในทุกกรณี แล้วเวลาหลังจากนี้จนกระทั่งถึงคลื่นลูกถัดไปจะเป็นการอ่านค่าจาก Joystick โดย โปรแกรมย่อย read1 จากนั้น แปลงค่าที่ได้เป็นความเร็ว และทิศทาง โดยโปรแกรมย่อย NAV (Navigation) จากนั้นแปลงค่าความเร็วและทิศทางเป็นค่า TON และ TURN ตามลำดับ และยังเช็คสวิตซ์ว่าถูกกดหรือไม่ในขั้นตอนนี้ด้วย หลังจากนั้นจะเข้าสู่คลื่นลูกถัดไปซึ่งมีการทำงานเช่นเดียวกับคลื่นลูกแรกเช่นนี้ไปเรื่อยๆ

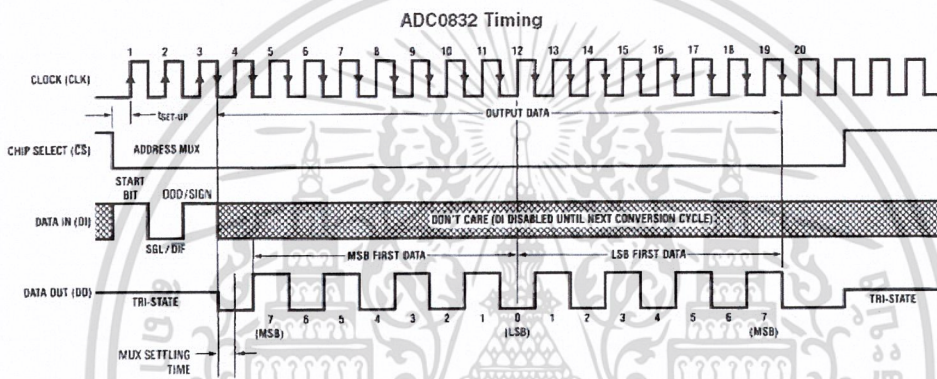
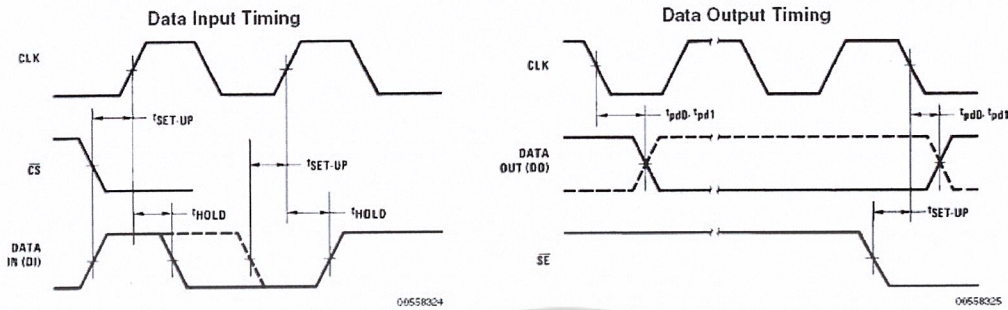
#### FLOW CHART การทำงานมีดังนี้



รูปที่ 6.2 แสดงโฟลชาร์ตการทำงานของโปรแกรมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

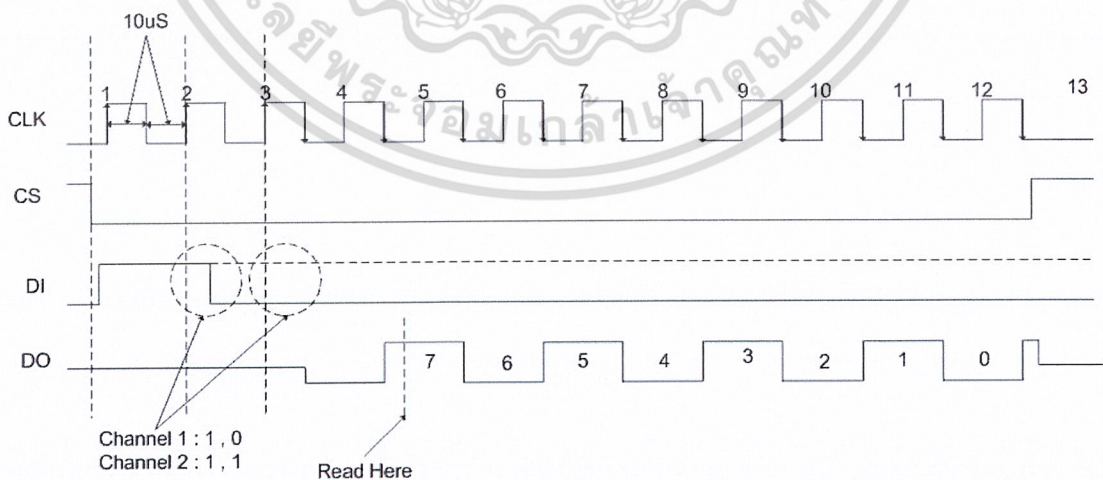
1. โปรแกรมย่อยอ่านข้อมูล read1 สำหรับติดต่อกับไอซี A to D นั้นกำหนดเวลา TimeDiagram ดังรูปข้างล่างนี้



รูปที่ 6.3 แสดงสัญญาณสำหรับติดต่อกับ ไอซี ADC0832

ภาพข้างล่างเป็น Time Diagram การอ่านสัญญาณอนาล็อกที่แปลงมาเป็นดิจิทัลแล้วของ

Channel 1 ของ ADC0832

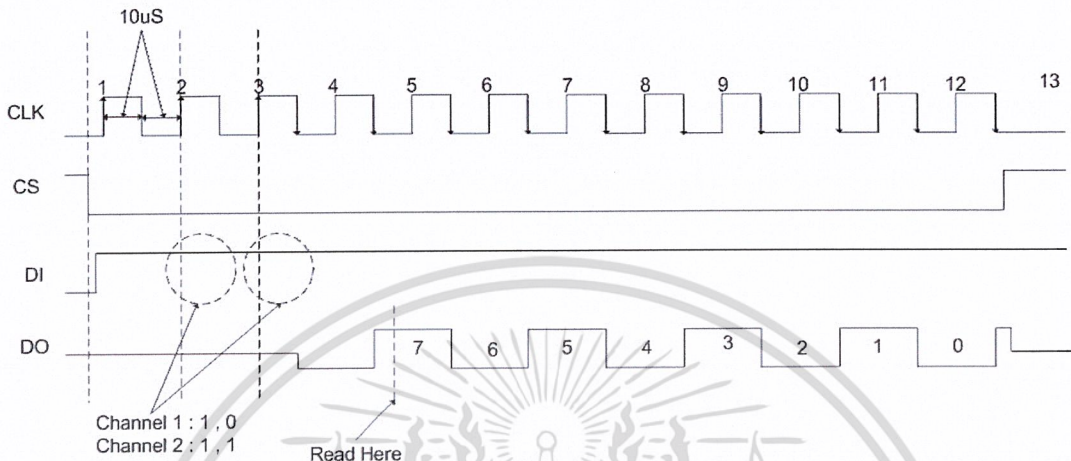


รูปที่ 6.4 แสดง Time Diagram ที่โปรแกรมย่อย read1 สร้างขึ้นเพื่อแปลงสัญญาณอนาล็อกเป็นดิจิทัลช่องที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

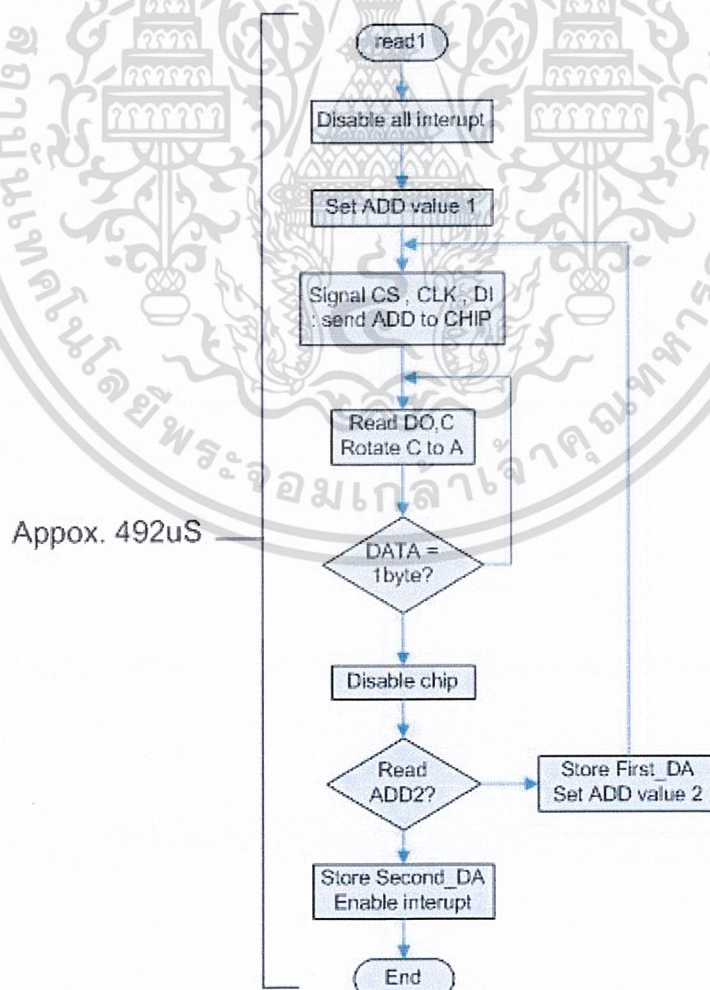
ภาพข้างล่างเป็น Time Diagram การอ่านสัญญาณอนาล็อกที่แปลงมาเป็นดิจิตอลแล้ว

ของ Channel 2 ของ ADC0832



รูปที่ 6.5 แสดง Time Diagram ที่โปรแกรมย่อย read1 สร้างขึ้นเพื่อแปลงสัญญาณอนาล็อกเป็นดิจิตอลช่องที่ 2

ภาพข้างล่างเป็น FLOWCHART ของโปรแกรม read1

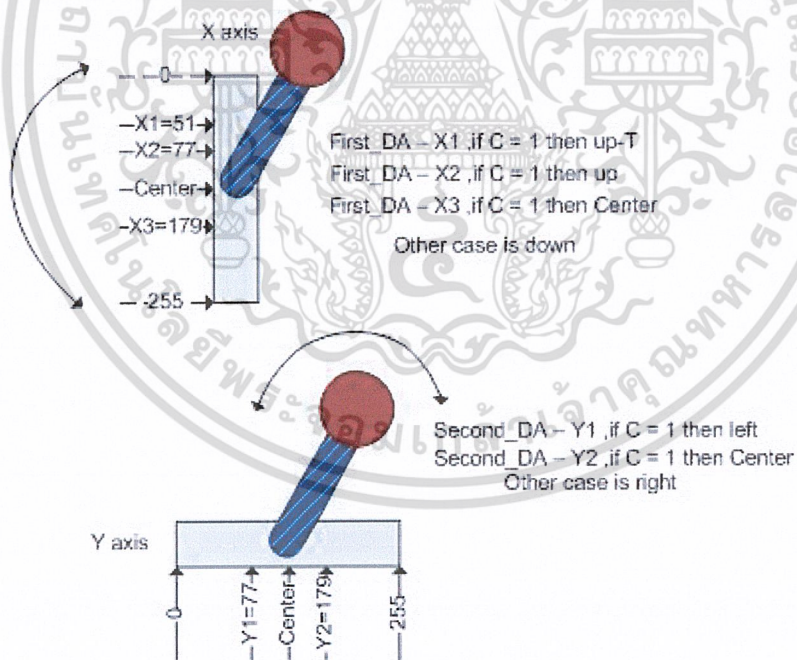


รูปที่ 6.6 แสดงโฟลชาร์ตการทำงานของโปรแกรมย่อย read1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมย่อย Sub-Program ชื่อ read1 จะส่งสัญญาณไปให้ไอซี ADC0832 ทำการแปลงสัญญาณอนาล็อกที่ CH1 และ CH2 เป็นดิจิตอลแล้วส่งข้อมูลให้กับไมโครคอนโทรลเลอร์ในรูปแบบ Bit Stream การทำงานของโปรแกรมย่อย read1 เป็นดัง FLOWCHART ข้างบน โดยเริ่มจากยกเลิกการ Interrupt ทั้งหมดที่อาจเกิดขึ้นก่อนเนื่องจากสัญญาณต่างๆที่ส่งนั้นควรมีเวลาที่แน่นอน การเกิด Interrupt จะทำให้สัญญาณที่ส่ง-รับมีคาบเวลาที่ผิดพลาดทำให้การส่งข้อมูล-รับข้อมูลผิดพลาดไป จากนั้นเก็บค่า Address ที่จะติดต่อยัง CH1 ลงไปเพื่อส่งเป็นสัญญาณออกไป เมื่อไอซีรับรู้ช่องสัญญาณที่จะแปลง A to D แล้วจะเริ่มขั้นตอนการส่งข้อมูลทั้งหมด 8 บิต จะเข้าสู่ loop เพื่อรับข้อมูลที่ละบิตผ่านรีจิสเตอร์ C แล้วหมุนค่าเข้าสู่รีจิสเตอร์ A จนครบทั้ง 8 บิตจึงจบการสื่อสารกับไอซีโดยยกบิต CS ให้เป็น "1" จากนั้นจึงตรวจสอบว่าได้อ่านค่าจาก Address ที่ติดต่อกับ CH2 หรือยัง ถ้ายังจะวนเข้าไปเพื่ออ่านข้อมูลอีกครั้งจาก CH2 ตามกระบวนการเดิม และเก็บค่าที่อ่านได้ไว้ที่ First\_DA และ Second\_DA เมื่อจบก็จะทำการ Disable ไอซีและ Enable Interrupt ให้ทำงานได้อีกครั้ง ซึ่งคำสั่งทั้งหมดได้ทำการคำนวณเวลาการทำงาน และหน่วงเวลาให้สัญญาณ Clock หรือ CLK มีความถี่ที่ 50kHz และ Duty Cycle ที่ 50%

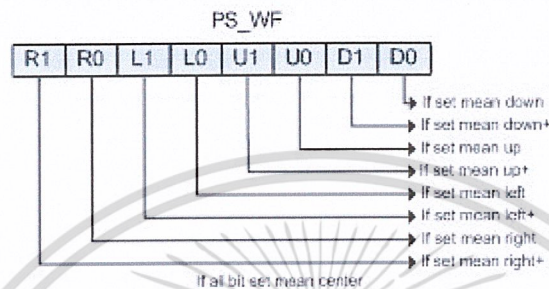
2. โปรแกรมย่อย NAV (Navigation) แปลงค่าความเร็วและทิศทาง มีการทำงานต่อจากโปรแกรมย่อย read1 ดังนี้



รูปที่ 6.7 แสดงการแบ่งแกนของจอยสติ๊กเพื่อกำหนดค่าตัวแปร

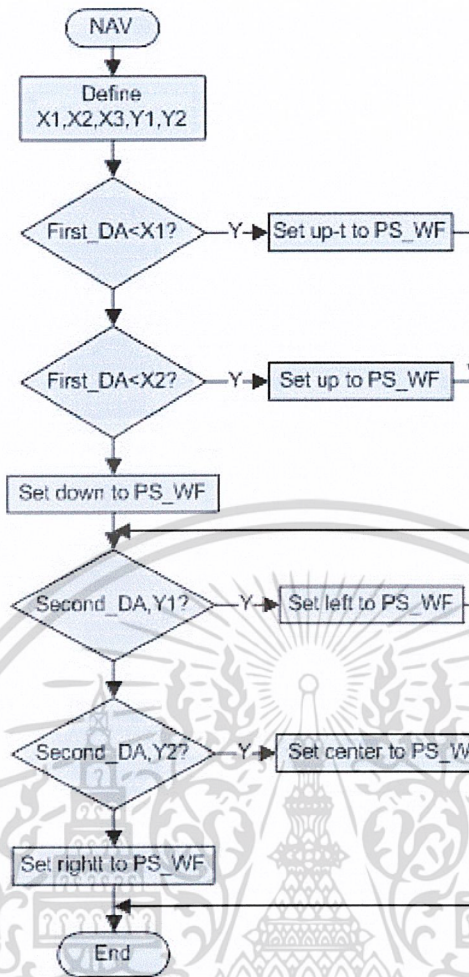
ในรูปแบบแสดงแบบจำลองของจอยสติ๊กซึ่งมี 2 แกนคือแกน X และแกน Y ต่อกับตัวต้านทานปรับค่าได้แบ่งแรงดัน 5V ส่งให้กับ Input ที่ CH1 และ CH2 ของไอซี ADC0832 ตามลำดับ ค่าที่แปลงเป็นดิจิตอลโดยไอซีตัวนี้จะสูงสุดที่ 255 (OFFH) และต่ำสุดที่ 0 ดังนั้น ค่ากึ่งกลางของจอยจะคำนวณได้ที่  $255/2 = 128$  และกำหนดตำแหน่งที่ X1, X2, X3, Y1 และ Y2 หากจอยโยกไปตกที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งใดโปรแกรมจะเช็คได้ว่าจะเคลื่อนที่ไปในทางใด บน , ล่าง , ซ้าย , ขวา หลักการทำงานของโปรแกรมคือ นำค่าที่ได้จาก โปรแกรม read1 คือค่าที่อ่านจากจอยสติ๊กทั้ง 2 แกนมาลบกับตำแหน่งที่ตั้งไว้คือ X1 , X2 , X3 , Y1 , Y2 หากลบแล้วรีจิสเตอร์ C เกิดติดลบ จะทำให้รู้ว่าขณะนี้จอยสติ๊กอยู่ในช่วงใดที่ตั้งเอาไว้ จะมาเก็บค่าการเคลื่อนที่ของรถไว้ที่รีจิสเตอร์ที่ได้ตั้งขึ้นชื่อ PS\_WF (Program Status Word Frequency) ซึ่งได้ประกาศใช้ในตอนต้นของโปรแกรม PS\_WF equ 34H มีรายละเอียดดังนี้คือ



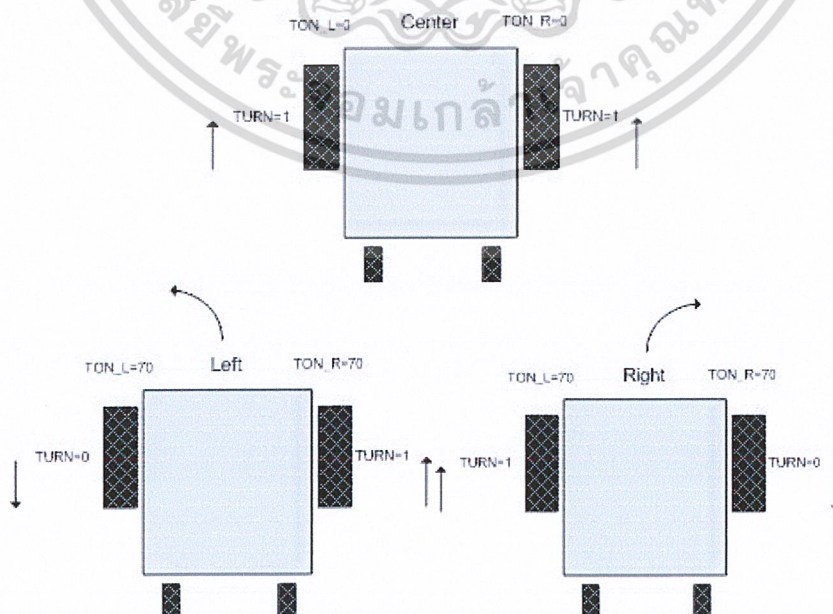
รูปที่ 6.8 แสดงรีจิสเตอร์ PS\_WF ซึ่งอนุমানขึ้นเพื่อใช้ในการบอกทิศทางให้กับรถ เพื่อให้โปรแกรมเช็คได้

คือ หากบิตใดเซต ก็จะมีหมายถึงทิศทางนั้นเช่น หากบิต U1 = '1' แปลว่า up+ เครื่องหมาย + หมายถึง speed ที่สูงกว่าที่ไม่มีเครื่องหมาย "+" นั้นหมายถึงว่ามีความเร็ว 2 speed นั้นเอง เมื่อเก็บค่าที่ PS\_WF เรียบร้อยแล้ว โปรแกรมย่อยที่ 3 ชื่อ CONVERT จะนำค่านี้ไปใช้อีกที่ FLOW CHART การทำงานของโปรแกรมย่อย NAV มีดังนี้



รูปที่ 6.9 แสดงโปรแกรมการทำงานของโปรแกรมย่อย NAV

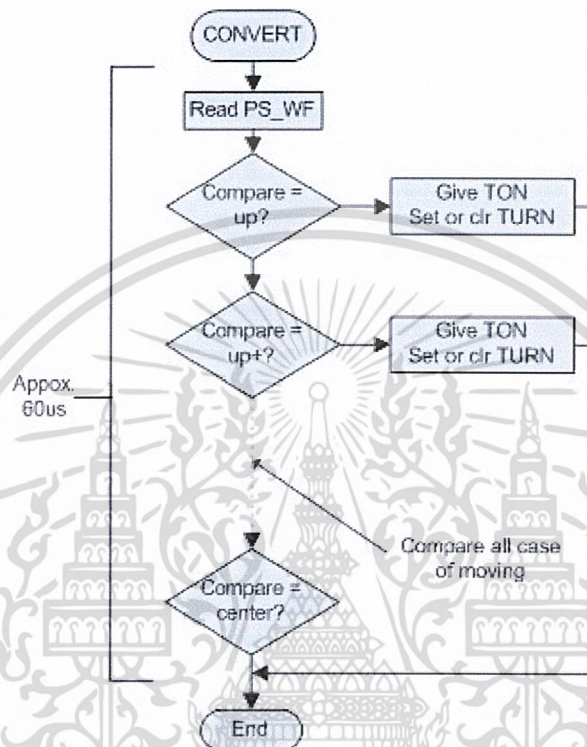
3. โปรแกรมย่อย CONVERT แปลงค่าความเร็วและทิศทางเป็นค่า TON และ TURN ตามลำดับมีหลักการทำงานดังรูปข้างล่าง



รูปที่ 6.10 แสดงวิธีที่รถเคลื่อนที่ไปในทิศทางต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป สัญญาณ TRUN เป็นสัญญาณที่กลับทิศทางการหมุนของมอเตอร์นั่นเอง และ TON จะกำหนด Duty cycle ของสัญญาณว่าจะเป็น 0% 70% หรือ 90% ส่งไปยังไอซี MOSFET DRIVER HIP4082 เพื่อควบคุมความเร็วของมอเตอร์ มี FLOW CHART การทำงานดังนี้



รูปที่ 6.11 แสดงโฟลชาร์ตการทำงานของโปรแกรมย่อย CONVERT

โดยเริ่มจากอ่านค่าหน่วยความจำ ที่ตั้งชื่อไว้ PS\_WF แล้วเปรียบเทียบค่าหากตรงกับการเคลื่อนที่แบบใด ก็ใส่ค่า TON และ TURN ที่เหมาะสมจากนั้นจึงออกจากโปรแกรมย่อย หากไม่ตรงกับค่าใดเลยหมายถึงอาจมีความผิดพลาด ในกรณีใดๆ ก็ให้ออกจากโปรแกรมย่อยไม่เปลี่ยนแปลงใดๆเลย

เมื่อโปรแกรมย่อย 3 โปรแกรมทำงานจบก็จะเช็คค่าจากสวิตช์ เพื่อ 1. เปิด ไปท้าย และ 2. หยุดการทำงานชั่วคราว จากนั้นจึงเริ่มไปทำโปรแกรมตั้งแต่ต้นใหม่อีกทีวนรอบไปเรื่อยๆ

### แสดงโปรแกรมสมบูรณ์ของโครงการนี้

CS      BIT    P0.4

CLK     BIT    P3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SW5	BIT	P0.5
SW4	BIT	P0.6
SW3	BIT	P0.7
DATA_DI	BIT	P0.2
DATA_DO	BIT	P0.3
AHO_L	BIT	P1.0
BHO_L	BIT	P1.1
AHO_R	BIT	P1.2
BHO_R	BIT	P1.3
TURN_L	BIT	P1.6
TURN_R	BIT	P1.7
ADD1	BIT	00H
ADD2	BIT	01H
SA	BIT	02H
PAUSE_1	BIT	03H
PAUSE_2	BIT	04H
PAUSE_3	BIT	05H
PAUSE_4	BIT	06H
BIT_COUNT	EQU	30H
FIRST_DA	EQU	31H
SECOND_DA	EQU	32H
LOOP	EQU	33H
PS_WF	EQU	34H
TON_L	EQU	35H
TOFF_L	EQU	36H
TON_R	EQU	37H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TOFF_R      EQU  38H
TIME        EQU  39H

X1          EQU  3AH
X2          EQU  3BH
X3          EQU  3CH
X4          EQU  3DH
Y1          EQU  3EH
Y2          EQU  3FH
Y3          EQU  40H
Y4          EQU  41H

```

```
ORG 0000H
```

```
JMP INITIAL
```

```
ORG 0030H
```

```
INITIAL:SETB TURN_L
```

```
    SETB TURN_R
```

```
    MOV TON_L,#0 ;time 11 , 1 - 11,25%
```

```
    MOV TON_R,#0
```

```
    MOV IE,#00000000B
```

```
ZERO_L:  MOV A,TON_L
```

```
    CJNE A,#0,ON_L
```

```
    CLR AHO_L
```

```
    CLR BHO_L
```

```
    JMP ZERO_R
```

```
ON_L:    SETB AHO_L
```

```
    SETB BHO_L
```

```
ZERO_R:  MOV A,TON_R
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CJNE A,#0,ON\_R  
 CLR AHO\_R  
 CLR BHO\_R  
 JMP DELAY70  
 ON\_R: SETB AHO\_R  
 SETB BHO\_R  
 DELAY70:CALL DELAY500US  
 CALL DELAY500US  
 CALL DELAY500US  
 CALL DELAY500US  
 CALL DELAY500US  
 CALL DELAY500US  
 SEVEN\_L:MOV A,TON\_L  
 CJNE A,#70,SEVEN\_R  
 CLR AHO\_L  
 CLR BHO\_L  
 SEVEN\_R:MOV A,TON\_R  
 CJNE A,#70,DELAY90  
 CLR AHO\_R  
 CLR BHO\_R  
 DELAY90:CALL DELAY500US  
 CALL DELAY500US  
 NINE\_L: CLR AHO\_L  
 CLR BHO\_L  
 NINE\_R: CLR AHO\_R  
 CLR BHO\_R  
 CALL READ1  
 CALL NAV  
 CALL CONVERT  
 JNB SW5,LIGHT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SETB P1.5
CHK_P:  JNB SW4,PAUSE
        JMP ZERO_L

```

```

DELAY500US:MOV R0,#124
LOOP500:  NOP
          NOP
          DJNZ R0,LOOP500
          RET

```

```

LIGHT:   CLR P1.5
        JMP CHK_P

```

```

PAUSE:   CALL BEEP
        CLR AHO_L
        CLR BHO_L
        CLR AHO_R
        CLR BHO_R
        JNB SW4,$
        CALL BEEP
        JMP ZERO_L

```

```

READ1:   CLR EA
        MOV BIT_COUNT,#9
        SETB ADD1
        CLR ADD2
        JMP ADD_ING

```

```

READ_2:  MOV BIT_COUNT,#9
        MOV FIRST_DA,A
        SETB ADD1
        SETB ADD2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADD_ING:CLR CS          ;BIT_COUTN,
SETB DATA_DO
SETB DATA_DI          ;START BIT
SETB CLK               ;*****
CALL DELAY_5US
MOV C,ADD1
MOV DATA_DI,C        ;ADD #1
NOP
CLR CLK                ;*****
CALL DELAY_5US
NOP
NOP
NOP
NOP
SETB CLK               ;*****
CALL DELAY_5US
MOV C,ADD2 ;ADD #2
MOV DATA_DI,C
NOP
CLR CLK                ;*****
CALL DELAY_5US
NOP
NOP
NOP
NOP
REC_DA: SETB CLK       ;*****
CALL DELAY_5US
NOP
NOP
NOP
NOP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CLR CLK          ;*****
CALL DELAY_5US
MOV C,DATA_DO
RLC A
DJNZ BIT_COUNT,REC_DA
END_DA: SETB CS
CLR DATA_DI
SETB DATA_DO
CALL DELAY_5US
CALL DELAY_5US
JNB ADD2,READ_2
MOV SECOND_DA,A
SETB EA
RET
DELAY_5US:NOP
RET
NAV:
INI:  MOV X1,#51
      MOV X2,#77
      MOV X3,#179

      MOV Y1,#77
      MOV Y2,#179
FIRST: CLR C
      MOV A,FIRST_DA
      SUBB A,X1
      JC UP_T
      MOV A,FIRST_DA
      SUBB A,X2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JC UP
MOV A,FIRST_DA
SUBB A,X3
JC CENTER
DOWN:  MOV PS_WF,#11110001B
        JMP SECOND
CENTER: MOV PS_WF,#11111111B
        JMP SECOND
UP:MOV PS_WF,#11110100B
        JMP SECOND
UP_T:  MOV PS_WF,#11111000B

SECOND: CLR C
        MOV A,SECOND_DA
        SUBB A,Y1
        JC LEFT
        MOV A,SECOND_DA
        SUBB A,Y2
        JC CENTER2
RIGHT:  MOV A,PS_WF
        ANL A,#01001111B
        MOV PS_WF,A
        RET
CENTER2:MOV A,PS_WF
        ANL A,#11111111B
        MOV PS_WF,A
        RET
LEFT:   MOV A,PS_WF
        ANL A,#00011111B
        MOV PS_WF,A
        RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CONVERT:MOV A,PS_WF
ST0:    CJNE A,#11111000B,ST1    ;UP-T
        SETB TURN_L
        SETB TURN_R
        MOV TON_L,#90
        MOV TON_R,#90
        RET
ST1:    CJNE A,#11110100B,ST2    ;UP
        SETB TURN_L
        SETB TURN_R
        MOV TON_L,#70
        MOV TON_R,#70
        RET
ST2:    CJNE A,#00011111B,ST3    ;LEFT
        CLR TURN_L
        SETB TURN_R
        MOV TON_L,#70
        MOV TON_R,#70
        RET
ST3:    CJNE A,#01001111B,ST4    ;RIGHT
        SETB TURN_L
        CLR TURN_R
        MOV TON_L,#70
        MOV TON_R,#70
        RET
ST4:    CJNE A,#00010100B,ST5    ;LEFT UP
        SETB TURN_L
        SETB TURN_R
        MOV TON_L,#0
        MOV TON_R,#70

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RET
ST5:    CJNE A,#00011000B,ST6    ;LEFT UP-T
        SETB TURN_L
        SETB TURN_R
        MOV TON_L,#0
        MOV TON_R,#90
        RET
ST6:    CJNE A,#01000100B,ST7    ;RIGHT UP
        SETB TURN_L
        SETB TURN_R
        MOV TON_L,#70
        MOV TON_R,#0
        RET
ST7:    CJNE A,#01001000B,ST8    ;RIGHT UP-T
        SETB TURN_L
        SETB TURN_R
        MOV TON_L,#90
        MOV TON_R,#0
        RET
ST8:    CJNE A,#11110001B,ST9    ;DOWN
        CLR TURN_L
        CLR TURN_R
        MOV TON_L,#70
        MOV TON_R,#70
        RET
ST9:    CJNE A,#00010001B,ST10   ;LEFT DOWN
        CLR TURN_L
        CLR TURN_R
        MOV TON_L,#0
        MOV TON_R,#70
        RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ST10: CJNE A,#01000001B,ST11 ;RIGHT DOWN**

**CLR TURN\_L**

**CLR TURN\_R**

**MOV TON\_L,#70**

**MOV TON\_R,#0**

**RET**

**ST11: CJNE A,#11111111B,ST12 ;CENTER**

**SETB TURN\_L**

**SETB TURN\_R**

**MOV TON\_L,#0**

**MOV TON\_R,#0**

**RET**

**ST12: RET ;ERROR OTHER**

**BEEP: CALL SOUND**

**CALL D1**

**CALL SOUND**

**CALL D1**

**CALL SOUND**

**CALL D1**

**RET**

**SOUND: MOV R1,#050**

**NUM: CPL P1.4**

**DELAY: MOV R0,#60**

**RO: NOP**

**NOP**

**DJNZ R0,RO**

**DJNZ R1,NUM**

**RET**

**D1: MOV R2,#50**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**L2: MOV R3,#0FFH**

**L1: NOP**

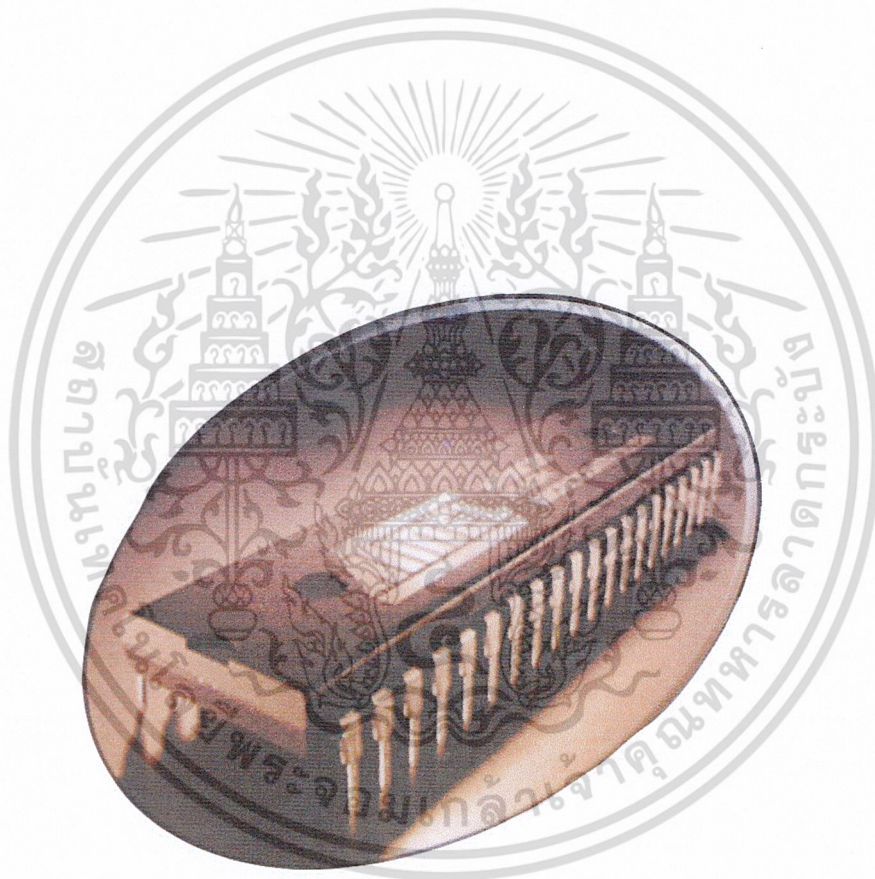
**NOP**

**DJNZ R3,L1**

**DJNZ R2,L2**

**RET**

**END**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

## โครงสร้าง และการประกอบ

โครงสร้าง วัสดุที่ใช้ประกอบขึ้นเป็นตัวโครงนั้นคือ เหล็กกลางซึ่งซื้อได้จากร้านขายเหล็ก โดยทั่วไปเมื่อประกอบเสร็จแล้วจะได้โครงสร้างดังในรูปคือ



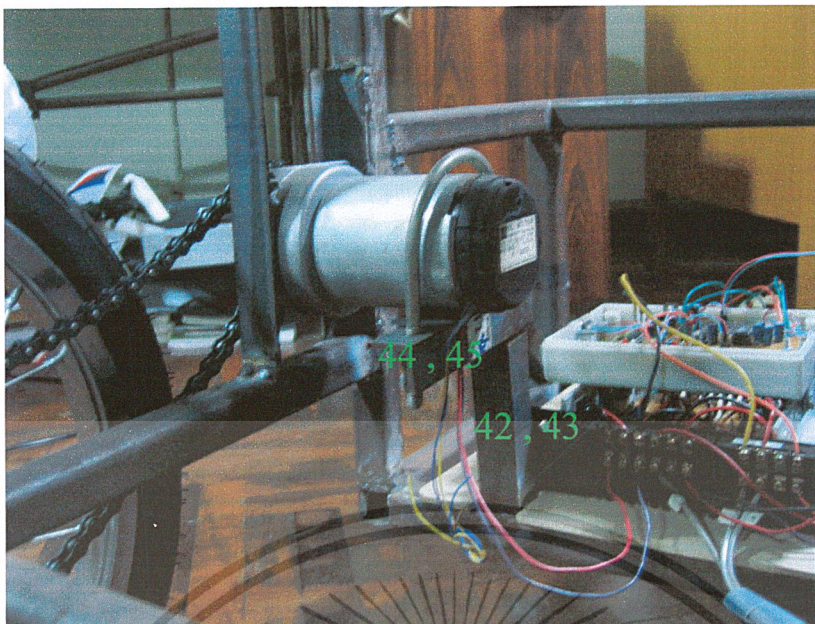
รูปที่ 7.1 แสดงหมายเลขซึ่งแทนค่า ID ของเหล็ก

ตำแหน่งที่วางเหล็กแต่ละชิ้น ได้ถูกวัดในทุกจุดแล้ว โดยการทดลองนั่งและดูความเหมาะสมซึ่งมีตัวเลขออกมาหลายขนาด มิได้ลงรายละเอียดไว้ในที่นี้ จึงขอลงเพียงตำแหน่งโดยคร่าวๆเท่านั้น



รูปที่ 7.2 แสดงหมายเลขซึ่งแทนค่า ID ของเหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.3 แสดงหมายเลขซึ่งแทนค่า ID ของเหล็ก

แมคคาณิก สังเกตในรูปจะใช้มอเตอร์ทศรอบ 2 ตัวที่ใกล้เคียงกันมากที่สุดเพื่อให้รอบที่หมุนใกล้เคียงกัน และใช้โซ่จักรยานเพื่อขับไปยังล้อ หากสังเกตที่ล้อจะใช้ล้อจักรยานเส้นผ่านศูนย์กลางประมาณ 32 ซม. ต่อเข้ากับ"จาน" ซึ่งจานหมายถึงเฟืองจักรยานล้อหน้านั่นเอง ร้านจักรยานทั่วไปจะเข้าใจคำว่าจานนี้ โดยนำมาคว้านรูแล้วใส่กับล้อ ซึ่งเป็นล้อหลัง จากนั้นนำที่ล้อเฟืองจักรยานของล้อหลังที่มีขายตามร้านจักรยานทั่วไปขันให้แน่น แล้วนำไปให้ช่างเชื่อมให้เพื่อป้องกันจานหมุนฟรี สำหรับที่มอเตอร์จะต่อกับเฟืองจักรยานล้อหลัง เป็นเฟืองที่หมุนทางเดียวเรียกว่า "สเตอร์" แต่สเตอร์นี้เดิมทีจะใส่กับล้อหลังของรถจักรยานจึงมีรูที่กว้างกว่าแกนมอเตอร์มาก จึงต้องกลึงปลอกสำหรับใส่สเตอร์เข้ากับแกนของมอเตอร์ แล้วนำไปเชื่อมให้สเตอร์ติดแน่นกับปลอกมากยิ่งขึ้น โดยการยึดปลอกนี้เข้ากับมอเตอร์ เมื่อสังเกตในรูปจะมีน็อตสี่ตัวหนึ่งขั้วล๊อคกับมอเตอร์เรียกว่า "ตัวหนอน" หากซื้อค้อนข้างยากมีแหล่งอยู่ที่ปากซอยสุขุมวิท 101/1 เดินมาทางฝั่งไปรษณีย์ไม่แน่ใจว่าสาขาอะไร เหตุที่ใช้ตัวหนอนแทนน็อตปกติธรรมดาเนื่องจากถ้าเป็นน็อตปกติไม่คงทนเท่าตัวหนอนใช้ไปจะเกิดการรูดได้ตามคำบอกเล่าของช่างเหล็ก

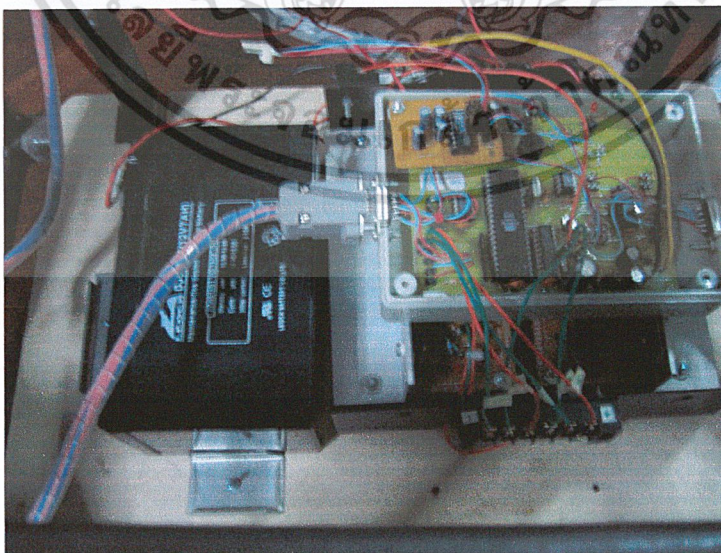
**วัสดุที่ใช้ประกอบโครงมีดังนี้**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัสดุ	ยาว	ID	จำนวน
แป็บโป่ง 1นิ้ว หนา2มม.	52ซม.	1,2	2
	45ซม.	3,4,15,16,19,20,25	7
	24.4ซม.	9,10	2
	34.4ซม.	11,12	2
	54.5ซม.	13,14	2
	28ซม.	17,18	2
	25.5ซม.	23,24	2
	31ซม.	26,27	2
	50ซม.	28,29	2
	28.5ซม.	34,35,37,38	4
	35ซม.	36,39	2
	20ซม.	21,22	2
	เหล็กฉาก 1นิ้ว หนา4มม	24.4ซม.	30,32
13.1ซม.		42,43	2
12ซม.		44,45	2
5ซม.		40,41	2
15ซม.		31,33	2

ตารางที่ 7.1 แสดงวัสดุที่ใช้ประกอบ โครง ความยาว และจำนวนที่ใช้

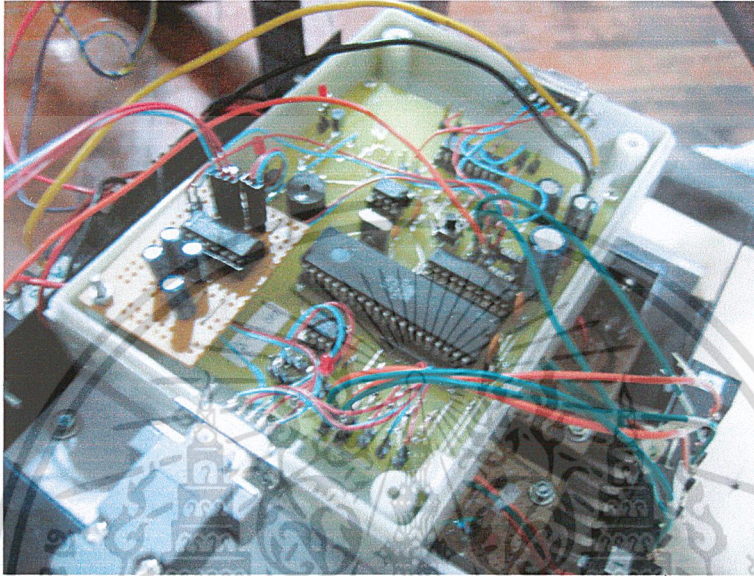
วงจร การประกอบวงจรนั้นสำหรับภาคไมโครคอนโทรลเลอร์ได้ทำการออกแบบลายวงจรแบบ 2 หน้า PTH ไว้โดยโปรแกรม Protel 99SE แต่มิได้แสดงไว้ ณ. ที่นี้ ทั้งนี้ได้แสดง Schematic ไว้ด้วยแล้ว เนื่องจากจุดต่อสายมีมาก จึงได้ใช้คอนเน็คเตอร์ในการต่อสายดังที่จะเห็นได้จากในรูป สำหรับภาค DRIVE จะเห็นว่าอยู่ข้างใต้ภาคควบคุม จากจุดนี้ทำให้การเดินสายค่อนข้างทำได้ลำบาก และการตรวจเช็คสัญญาณยิ่งทำได้ลำบาก หากในโอกาสหน้าการนำมาอยู่ในระนาบเดียวกันไม่ซ้อนทับกันจะดีกว่า สายไฟที่เดิน ไปยังส่วนต่างๆเช่นสวิทช์ ตัวต้านทานปรับค่าได้ เนื่องจากมีหลายเส้นควรใช้สายพลาสติกหุ้มให้เป็นระเบียบ หาซื้อได้จากร้านชื่อ นัฐพงษ์ เซลล์เอนด์เซอร์วิส ย่านบ้านหม้อ



รูปที่ 7.4 แสดงการเดินสายและติดตั้งวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นแบตเตอรี่ขนาด 7AH วางอยู่ข้างๆกล่องใส่วางจอร์ ที่เห็นลักษณะเหมือน เหล็กฉากสีขาวนั้นคือฮีทซิงค์สำหรับ IGBT ซึ่งจริงๆแล้ว IGBT นั้นแทนจะไม่มีความร้อนเลยแต่ได้ ติดตั้งเพื่อไว้มันเอง และสังเกตที่แผ่นปริ๊นออกประกอบสังกะสีเล็กๆที่มีตัวเก็บประจุ 4 ตัวนั้นคือ IC MAX232 สำหรับส่งข้อมูล RS-232 เพื่อส่งไปยังคอมพิวเตอร์สำหรับการดูค่าต่างๆที่เกิดขึ้น



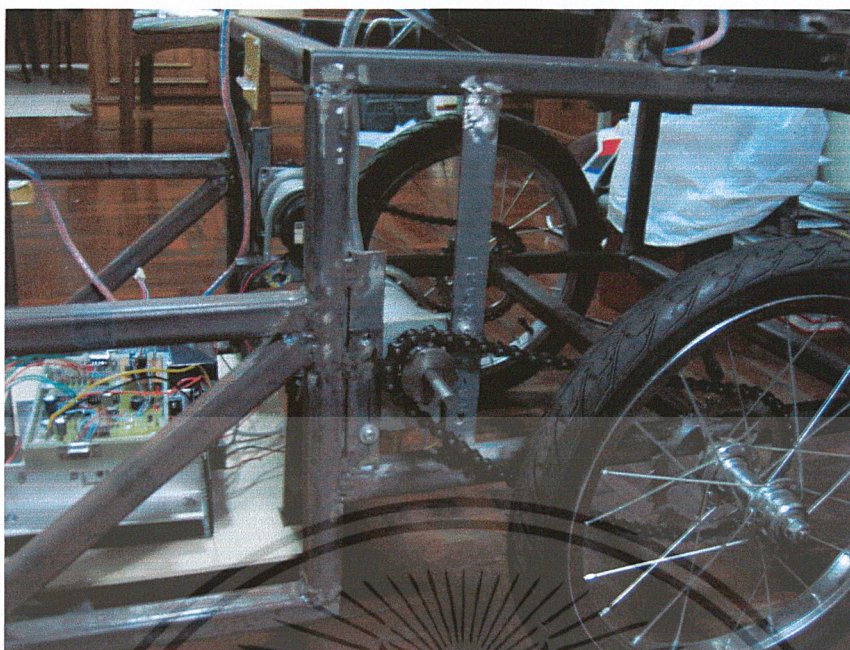
รูปที่ 7.5 ใต้ Zoom เข้ามาใกล้กับแผ่นวงจรภาคควบคุมมากขึ้น

จะเห็นว่ามียอนเน็กเตอร์ที่ตัวแผ่นวงจรหลายตัว ที่ต่อมายัง DB-9 เพื่อต่อไปยังส่วนต่างๆของรถต่อไป



รูปที่ 7.6 แสดงสวิตช์ที่ใช้ปิด-เปิด ไฟท้าย , หยุดการทำงาน

สวิตช์ 2 ตัวด้านล่างสำหรับเปิด-ปิด การทำงานบอร์ดภาคควบคุม กับ ภาค DRIVE ตามลำดับ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.7 แสดงอีกมุมมองหนึ่งของการติดตั้งวงจร

แผ่นที่รองรับวงจรสีขาวๆ นั้นคือ กระดาษอัด เหตุที่เลือกใช้กระดาษอัดเนื่องจากมีราคาถูกเมื่อเทียบกับไม้ทำการฝีมือ และเทียบความเหนียวกับความแข็งแรงแล้วพบว่า กระดาษอัดมีความเหมาะสมกว่าไม้

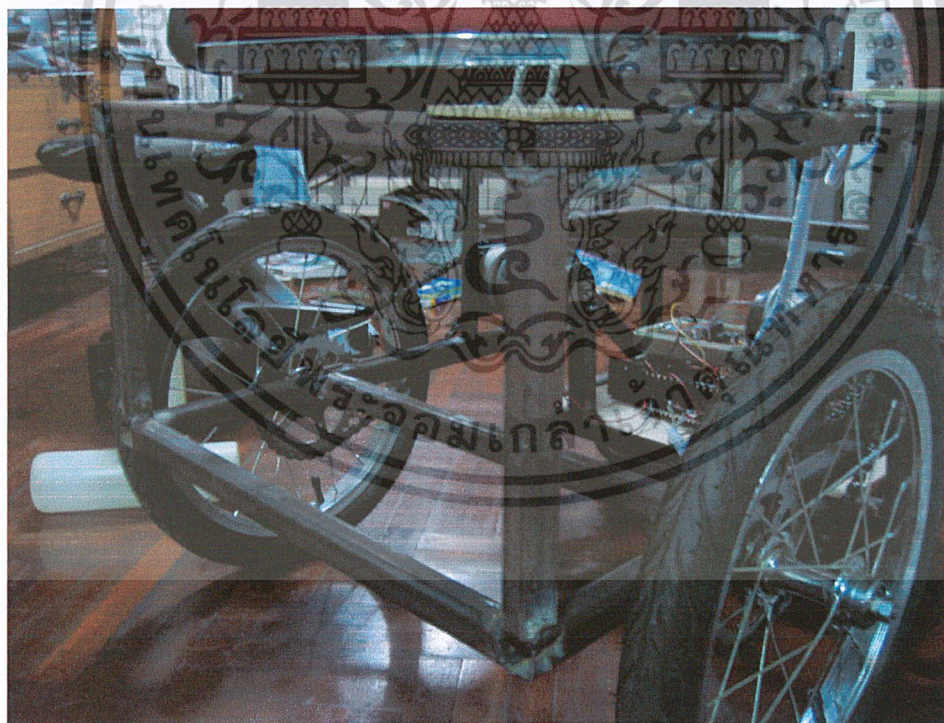


รูปที่ 7.8 แสดงภาพด้านหลังของรถ ซึ่งจะเห็นการติดตั้งไฟท้ายและวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.9 แสดงภาพด้านข้างของตัวรถ



รูปที่ 7.10 แสดงภาพด้านหน้าตัวรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.11 แสดงที่นั่งและการติดจอยสติ๊ก

สำหรับจอยสติ๊กที่นำมาใช้กับ โครจงานนี้ ชื้อจากห้างพันทิพย์พลาซ่า แล้วนำมาประกอบเข้ากับตัวรถ โดยซื้อมาทั้งหมด 2 ชุดเนื่องจากชุดแรกเมื่อประกอบและทดลองแล้วปรากฏว่าตัวด้านทานปรับค่าได้ที่แกนของจอยเกิดขาด และไม่สามารถหาซื้อลักษณะดังกล่าวได้จึงต้องซื้อชุดที่ 2 ดังรูป

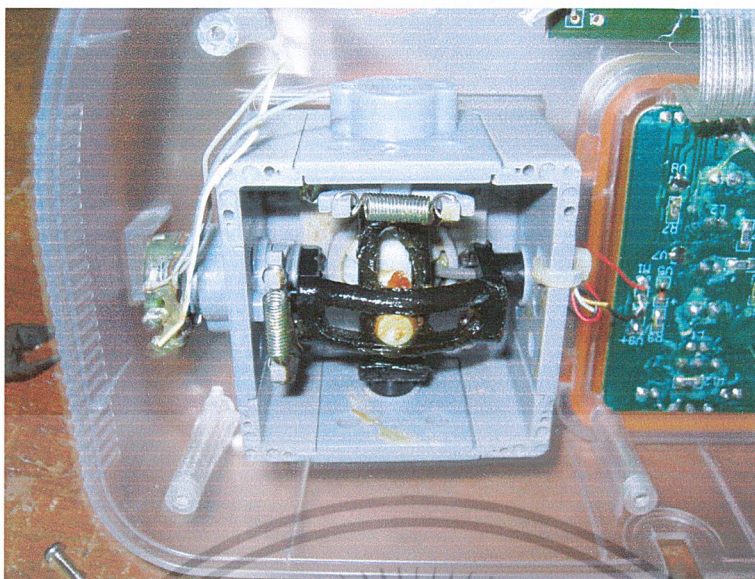


รูปที่ 7.12 แสดงจอยสติ๊กที่ใช้ในโครจงาน

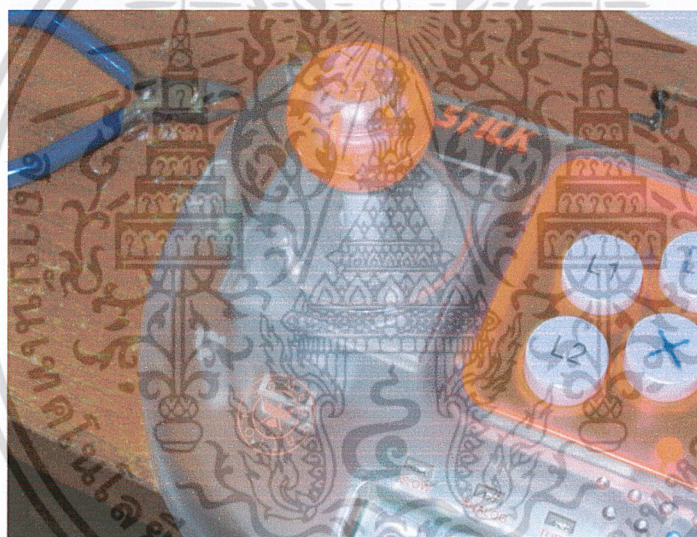


รูปที่ 7.13 แสดงจอยสติ๊ก 2 ตัวที่นำมาใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.14 แสดงแกน โยกที่จะต้องถอดออกมาใช้



รูปที่ 7.15 แสดงบริเวณแกน โยกที่จะถูกตัดมาใช้

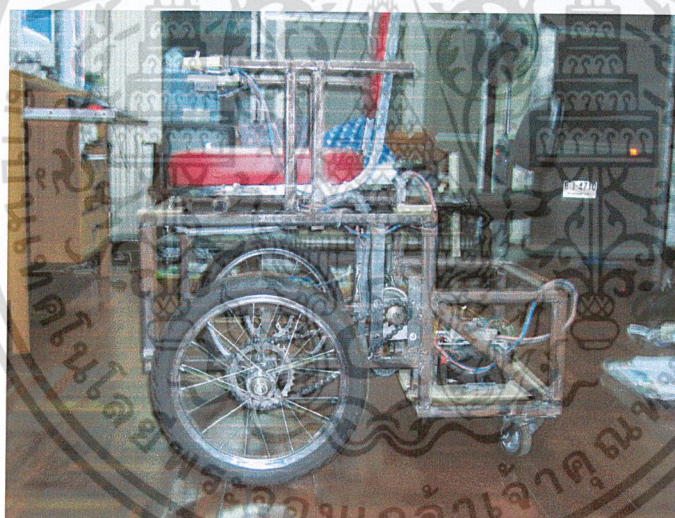
ได้ทำการตัดส่วนของจอยสติ๊กออกมาใช้ในโครงการนี้ โดยจะเห็นว่ามี 2 แกนอยู่บนจอยเดี่ยวและตัวต้านทานปรับค่าได้ที่จะนำมาใช้ได้เลยทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาพต่างๆ

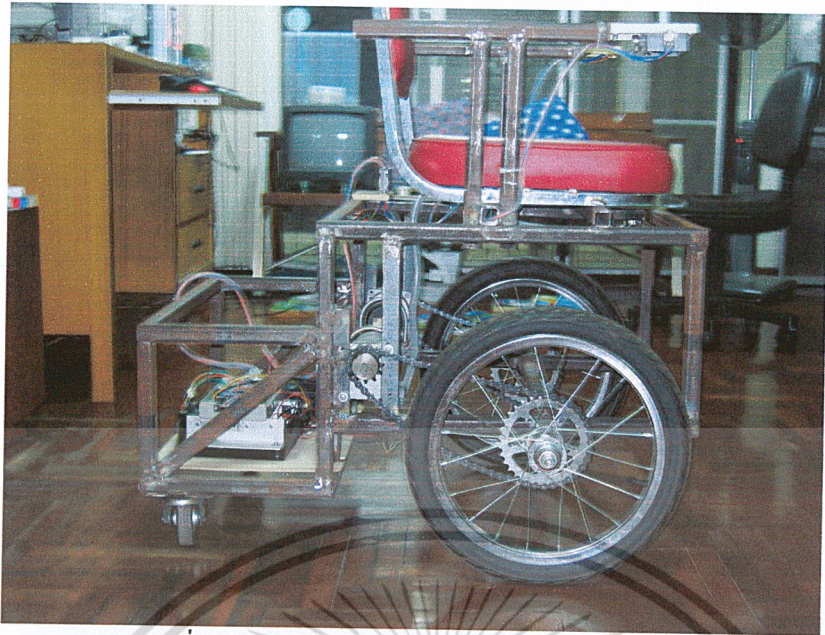


รูปที่ 7.16 ภาพด้านหน้าเมื่อเทียบกับเก้าอี้นั่ง



รูปที่ 7.17 ภาพด้านข้างแสดงการต่อมอเตอร์ขับเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.18 ภาพด้านข้างแสดงการต่อมอเตอร์ขับเคลื่อน



รูปที่ 7.19 ขยายมาที่การต่อมอเตอร์ไปยังล้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 8

### ปัญหาที่พบ และการแก้ไข

#### 1) วงจร

##### 1.1 อุปกรณ์เสีย

- ไอซี drive hip4082
- IGBT
- ไดโอด
- ไอซี NOT GATE
- อื่นๆ

**ไอซี MOSFET DRIVER เบอร์ HIP4082** เนื่องจากไอซีตัวนี้ มีความไวกับไฟฟ้าสถิตมาก ตามที่ DATASHEET บอกไว้ และ VDD maximum ที่ 15 VDC เท่านั้น ในการทดลองพบว่าไอซีตัวนี้ เสียไปเป็นจำนวนมาก ซึ่งในเบื้องต้น สันนิษฐานว่า เกิดจาก IGBT ช็อตจากกระแสกระชากของ มอเตอร์ ทำให้มีแรงดันไฟ ถึง 12 V เข้ามาที่ขาของ ไอซีทำให้เสียหายได้ เนื่องจากหลายครั้งที่ IGBT เสีย พบว่า ไอซีตัวนี้จะเสียหายไปด้วยทุกครั้ง ซึ่งการแก้ปัญหา ทำโดยเพิ่มความระมัดระวัง และใช้ IGBT ที่ทนกระแส และแรงดัน BREAK DOWN สูงขึ้นแทน พบว่า อัตราการเสียลดลง แต่ก็ยังคงพบว่ามีบางครั้งเมื่อเซ็นรต ซึ่งทำให้มอเตอร์ก็หมุนด้วยก็ดี หรือ ทดลองให้มอเตอร์หมุน ด้วยก็ดี ยังพบว่า ไอซี และ IGBT ยังคงเสียอยู่บ้าง ทำให้สันนิษฐานต่อไปว่า อาจจะมี แรงดันไฟรบกวนมาที่เกิดจากมอเตอร์เข้ามาที่ VDD ของไอซีเกินกว่า 15V ทำให้ไอซีเสียและ ทำงานผิดพลาดในการ DRIVE ให้ IGBT ทำงานพร้อมกันใน HIGH SIDE และ LOW SIDE ฝั่ง เดียวกัน ทำให้เกิด Short ดังกล่าวขึ้น ทั้งนี้ ไอซีเบอร์นี้มีฟังก์ชันการทำงานที่ป้องกัน IGBT ด้าน HIGH SIDE และ LOW SIDE "ON" พร้อมกันอยู่ แต่จากการทดลองพบว่า IGBT ที่เสียนั้นมักจะเสีย ที่ HIGH SIDE กับ LOW SIDE ฝั่งเดียวกัน ดังนั้นความผิดปกติที่จะเกิดที่ ไอซี MOSFET DRIVER นี้แล้วลามไปยัง IGBT เพื่อทดสอบสมมุติฐานนี้จึงได้ต่อ ตัวต้านทาน ซีเนอร์ไดโอด และตัวเก็บ ประจุเพื่อป้องกันมิให้ ขา VDD ของ ไอซี มีแรงดันไฟฟ้าเกิน 12V พบว่าสามารถแก้ปัญหา ไอซี และ IGBT เสียหายได้

**ไดโอด** จำนวนเล็กน้อยในลักษณะของการช็อต จากการใช้ความร้อนสูงในการบัดกรีเข้ากับ IGBT เนื่องจากเป็นจุดบัดกรีขนาดใหญ่

**ไอซี NOT GATE** ไอซีตัวนี้ ต่อเข้ากับ INPUT ของ ไอซี MOSFET DRIVER HIP4082 โดยตรงเมื่อ ไอซี DRIVE เกิดเสียหาย อาจทำให้มีแรงดัน ไฟ 12V เข้ามาทำความเสียหายได้ พบว่า GATE ที่ใช้ ต่อกับ ไอซี DRIVE ได้เสียพร้อมกับไอซี DRIVE ซึ่งการป้องกันสามารถใช้อุปกรณ์ ประเภท OPTOR ISOLATOR ในการแยก NOT GATE และ ไอซี DRIVE ออกจากกันได้ เป็น แนวทางในการปรับปรุงวงจรต่อไป

อื่นๆ เช่น แผ่นปรินต์ ซึ่งเสียหายจากกระแสไฟฟ้าที่ลัดวงจรอันเกิดจาก IGBT ช็อต ,การใช้คอนเน็คเตอร์ DB9 มาเป็นตัวเชื่อมสายไฟหลายเส้นเข้าด้วยกันเพื่อความมีระเบียบและตรวจสอบง่าย เป็นต้น

### 1.2 สายไฟมีจำนวนมาก

มีสายไฟต่อไปยังสวิทช์ จอยสติ๊ก คอนเน็คเตอร์ มอเตอร์ ฯลฯ พบว่าการใช้คอนเน็คเตอร์แบบขันน็อต แทนการบัดกรี จะเป็นระเบียบและตรวจสอบได้ง่ายกว่าการบัดกรีโดยตรง จากนั้น ใช้พลาสติกรัดสายไฟเข้าด้วยกันให้ดูมีระเบียบ

### 1.3 ค่าอุปกรณ์บางตัวที่เหมาะสม

ค่า C สำหรับ voltage level shift ที่เหมาะสม จากการทดลอง พบว่า ไอซี DRIVE ต้องการไดโอด และตัวเก็บประจุเพื่อ เป็นแรงดันให้กับขา GATE และขา EMITTER ของ IGBT ด้าน HIGH SIDE ซึ่งเป็น NON LATCHING ซึ่งหมายถึง แรงดันนี้ จะค่อยๆลดลงเรื่อยๆ ไม่ได้มีอยู่ตลอดเวลา จึงต้องใช้ตัวเก็บประจุเก็บค่าไว้ ไม่ให้ ลดค่าเร็วจนเกินไป ส่งผลให้ การ ON ของ IGBT ทำได้ไม่เต็มที่ และมีแรงดันคร่อม ขา C และ E ถึง 5V เหลือที่มอเตอร์เพียง 6V จากที่ควรจะเป็นประมาณ 11V (IGBT มี  $V_{ce SAT}$  ที่ 0.3 - 0.6 V) R สำหรับ LED LED จะสว่างเต็มที่สำหรับการจ่ายกระแสต่อเนื่องประมาณ 15 - 20 mA การให้กระแสสูงกว่านี้จะเป็นอันตรายต่อ LED

## 2) ซอร์ฟแวร์

เกิดจากสาเหตุใหญ่ 2 อย่างคือ 1)ฮาร์ดแวร์ กล่าวคือ วงจรมีการเดินสายไฟที่ค่อนข้างมากทำให้เกิดความสับสน พบว่าได้ต่อสายไฟผิด เมื่อเขียน โปรแกรม พบว่าไม่ได้ผลดังที่ตั้งใจไว้ เมื่อค้นหาสาเหตุจึงพบว่าเกิดจากฮาร์ดแวร์ซึ่งเสียเวลาไปมาก 2) โปรแกรม กล่าวคือ เนื่องจาก โปรแกรมมีความยาวอีกทั้งยังไม่สามารถดูผลของรีจิสเตอร์ที่เปลี่ยนแปลงได้ทำให้การหาข้อผิดพลาดเป็นไปด้วยความยากลำบาก

การแก้ปัญหาทั้ง 2 ข้อใหญ่ที่กล่าวมาจะใช้ 3 วิธีประกอบกันและถือหลักการที่ง่ายและซับซ้อนน้อยที่สุด 1)ใช้โปรแกรม Proteus จำลองโปรแกรมที่เขียนขึ้นบนคอมพิวเตอร์ก่อน เมื่อทำงานได้ตรงกับที่ตั้งใจไว้แล้วจึงทำ 2) การทดลองกับฮาร์ดแวร์คือทดลองจริง และร่วมกับวิธีที่ 3)คือต่อวงจรสื่อสารอนุกรม RS-232 ระหว่างไมโครคอนโทรลเลอร์ กับคอมพิวเตอร์ ซึ่งทางผู้จัดทำได้ใช้เครื่อง Palm ร่วมกับโปรแกรม Terminal ในการดูข้อมูลที่ส่งออกมาแทนเครื่องคอมพิวเตอร์ ซึ่ง ได้ผลดีและมีความคล่องตัวเป็นอย่างมาก ซึ่ง 3 วิธีนี้ทำให้ทราบสถานะของโปรแกรม และแก้ไข "BUG" ได้สะดวกรวดเร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3) เมคคานิก

#### 3.1 ความแข็งแรง

#### 3.2 นำหนัก

ปัญหาที่พบมีมาก แต่ทางผู้จัดทำได้ปรึกษากับผู้ที่มีความรู้ในด้านนี้จึงทำให้แก้ปัญหาและประกอบชิ้นส่วนได้อย่างรวดเร็ว

- แบบแปลน
- วัสดุที่ใช้
- แหล่งซื้อ
- ปรับแก้ไข โครงสร้าง
- ปรับระบบขับเคลื่อน

ผู้จัดทำมีเพื่อนที่รู้จักกันได้ดูแบบ และมีการปรับแบบ โครงสร้างหลายครั้ง ด้วยมีประเด็นสำคัญสรุปได้ดังนี้

1. โครงสร้างที่ง่ายที่สุด
2. จุดเชื่อมที่น้อยที่สุด
3. ใช้วัสดุจำนวนชิ้นน้อยที่สุด
4. มีความคงทน
5. มีความสวยงาม
6. ราคาประหยัด

ดังนั้น หลังจากที่ผู้จัดทำแก้ไขแบบเสร็จแล้ว จึงซื้อวัสดุ ตัดเหล็ก แล้วว่าจ้างให้ร้านเชื่อมเหล็กทำการเชื่อมให้ พบว่าเมื่อแบบสร้างเสร็จแล้วนั้นการทำโครงสร้างดำเนินไปได้รวดเร็วมาก พบปัญหาหลักๆคือ

1. วัสดุที่จะนำมาเป็นล้อและเพลลา เพื่ออง หาที่ราคาถูกได้ยากจึงใช้ล้อจักรยาน ซึ่งก็มีความแข็งแรงไม่มากนัก การเชื่อมต่อระมัดระวัง ความร้อนที่สูงเกินไปทำให้เฟืองเกียร์เบี้ยวได้
2. ล้อเกาะพื้นได้ไม่ครบทั้ง 4 ล้อ โดยเฉพาะล้อเล็กช่วยพยุงล้อหลัง การเชื่อมต่อพยายามทำให้ได้ฉาก และตรวจสอบการเชื่อมเพลลา และยึดล้อหลังให้ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

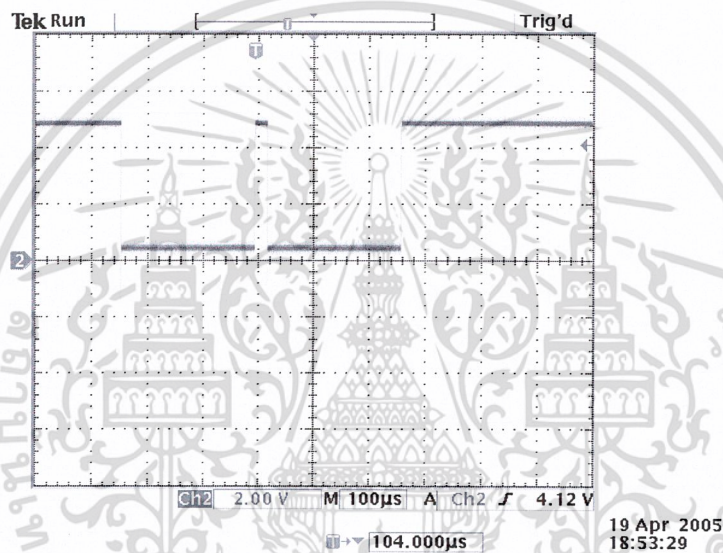
## บทที่ 9

## สรุปผลการทดลอง

## วงจร

วงจรควบคุม อันประกอบด้วยไมโครคอนโทรลเลอร์ ,Watchdog ,ADC Converter และอื่นๆ ทำงานได้กล่าวคือ

1. ADC Converter จากรูปที่ 9.1 แสดงให้เห็นถึงสัญญาณที่ขา CS หรือ Chip Select ซึ่งแอกทีฟที่ลอจิกโลว์



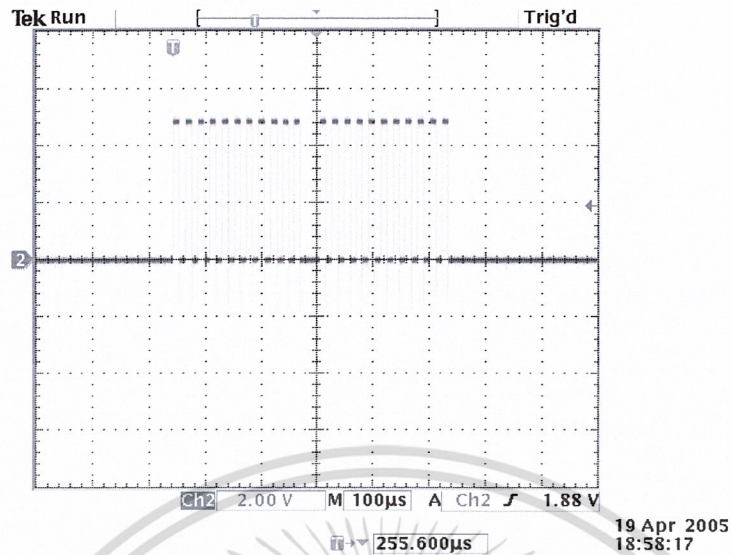
รูปที่ 9.1 แสดงสัญญาณที่ขา CS ของ ADC0832

จากรูปสัญญาณที่ปรากฏ แสดงว่าไอซีจะทำการอ่านและแปลงค่าจากสัญญาณอนาลอกไปเป็นสัญญาณดิจิทัล ในขณะที่ลอจิก โลว์ 2 ครั้งในเวลาใกล้เคียงกัน ซึ่งก็เป็นการอ่านค่าจาก Joystick แกน X และแกน Y นั้นเอง

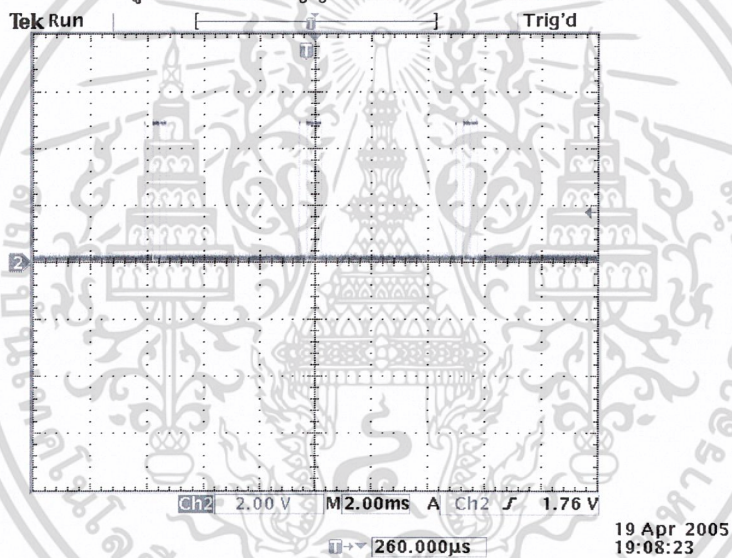
ในรูปที่ 9.2 แสดงให้เห็นถึงสัญญาณ CLK หรือ Clock ที่เกิดจากไมโครคอนโทรลเลอร์ส่งสัญญาณไปเป็นฐานเวลาให้กับไอซี ADC0832 เพื่อใช้ในการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล แสดงให้เห็นว่าโปรแกรมสามารถผลิตฐานเวลาได้ตามการคำนวณการหน่วงเวลาของสัญญาณ On และ Off ที่ทำให้ได้ความถี่ประมาณ 10kHz

ในรูปที่ 9.3 และ 9.4 แสดงให้เห็นถึงสัญญาณ DI หรือ Data in และ Do หรือ Data out ตามลำดับ จากสัญญาณที่วัดแสดงให้เห็นว่า สัญญาณ DI ซึ่งใช้ในการกำหนด Address ที่อ่านค่าจาก Channel 1 ,2 ของไอซี ทำให้ไอซีส่งค่าที่อ่านออกมาทาง DO และจากการโยกแกน Joystick ที่ให้สัญญาณที่ DO เปลี่ยน ไปแสดงให้เห็นว่าการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลทำงานได้

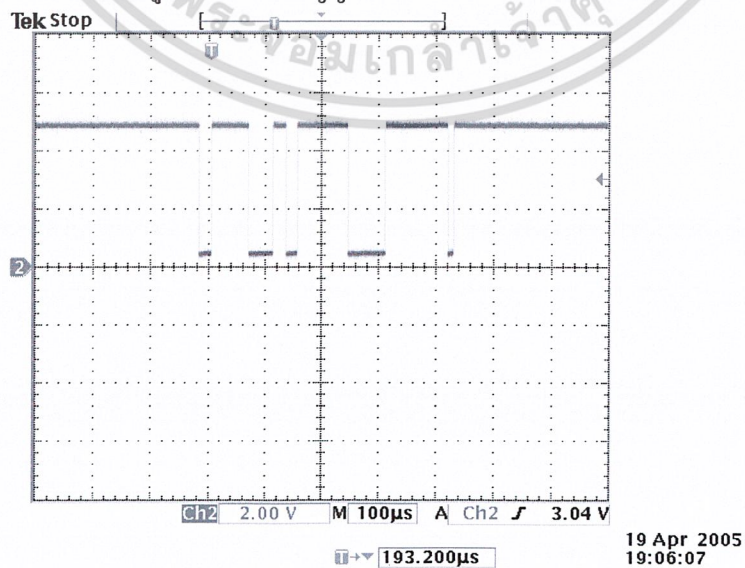
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9.2 แสดงสัญญาณที่ขา CLK ของ ADC0832



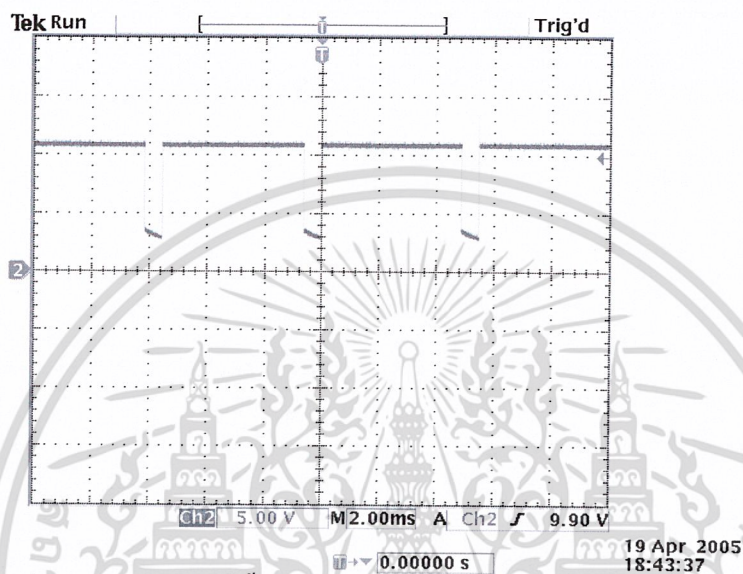
รูปที่ 9.3 แสดงสัญญาณที่ขา DI ของ ADC0832



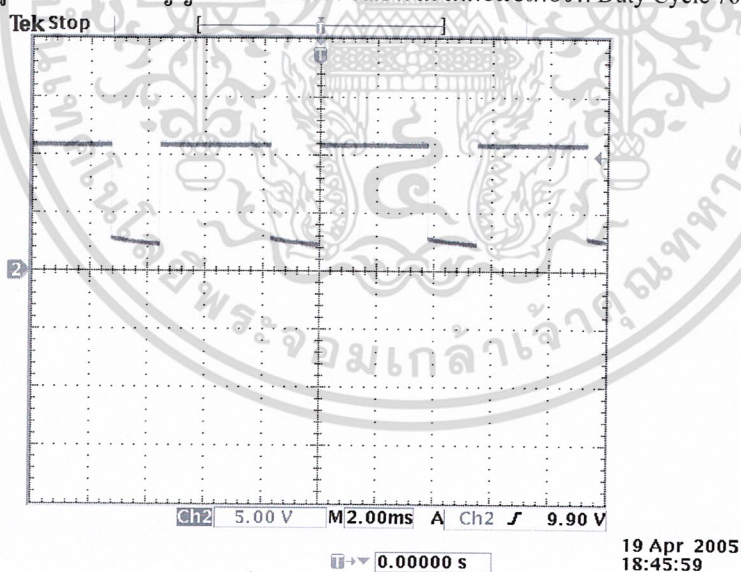
รูปที่ 9.4 แสดงสัญญาณที่ขา DO ของ ADC0832

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Drive และ IGBT จากรูปที่ 9.5 , 9.6 เป็นสัญญาณที่วัดคล่อมที่ขั้วต่อมอเตอร์ ซึ่งไม่ได้ต่อมอเตอร์ไว้ ต่อไว้แต่เพียงตัวต้านทานที่อนุกรมกับ LED เพื่อแสดงว่ามอเตอร์หมุนในทิศทางไหน เท่านั้น ในรูปแสดงสัญญาณที่ Duty Cycle ที่ 70% และ 90% ตามลำดับ จากสัญญาณจะเห็นว่าเมื่อ IGBT หยุดนำกระแส แรงดันจะลดลงอย่างรวดเร็วในช่วงแรกเห็นเป็นเส้นลงมา จากนั้นจะค่อยๆ ลดลงไป ซึ่งตรงกับทฤษฎีคือ IGBT จะมีพฤติกรรมของมอสเฟตและทรานซิสเตอร์อยู่ดังรูปที่ 3.7



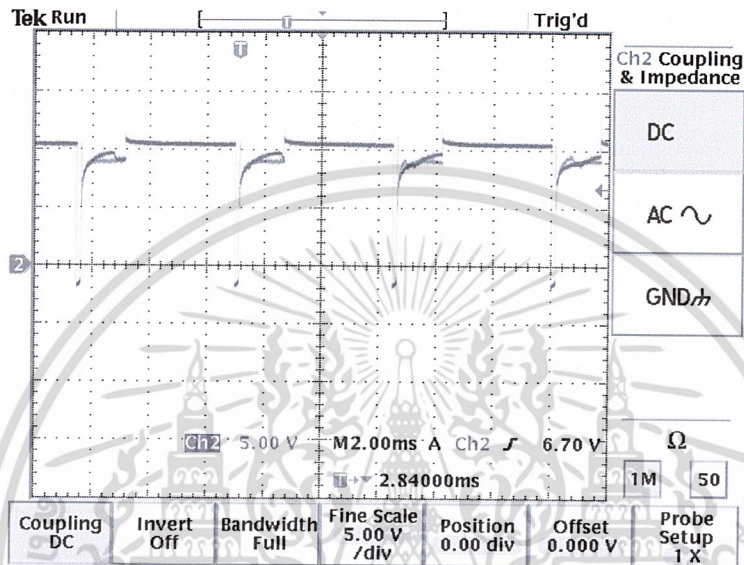
รูปที่ 9.5 แสดงสัญญาณที่ขั้วมอเตอร์ขณะที่ไม่ได้ต่อมอเตอร์ที่ Duty Cycle 70%



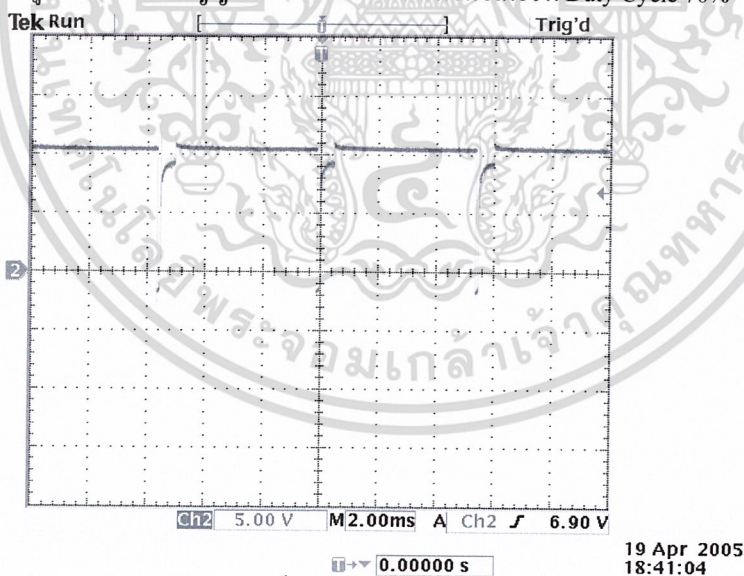
รูปที่ 9.6 แสดงสัญญาณที่ขั้วมอเตอร์ขณะที่ไม่ได้ต่อมอเตอร์ที่ Duty Cycle 90%

ในรูปที่ 9.7 , 9.8 เป็นสัญญาณที่วัดคล่อมที่ขั้วต่อมอเตอร์เช่นกันแต่ได้ต่อมอเตอร์ 12-24V ซึ่งเป็นมอเตอร์ที่ใช้ในโครงการนี้ไว้ และให้หมุนแบบ Free Load คือไม่ได้ต่อโซ่และล้อ จากสัญญาณจะเห็นว่าเมื่อ IGBT หยุดนำกระแส แรงดันจะตกลงอย่างรวดเร็วและต่ำกว่า 0V ซึ่งตรงนี้เป็นผลจากการที่  $di/dt$  ของขดลวดที่มอเตอร์ คือเมื่อกระแสตกลงอย่างรวดเร็วจะทำให้เกิดความต่างเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัปดาห์ก่อนมอเตอร์ซึ่งเป็นแรงดัน Reverse bias ให้กับ IGBT หากค่านี้มีค่าเกิดแรงดัน Breakdown ของ IGBT จะทำให้ IGBT เสียหายได้ซึ่งตรงนี้ได้ต่อ Diode เพื่อป้องกันปัญหานี้แล้ว และจะสังเกตเห็นว่าจากนั้นแล้วแรงดันจะสูงขึ้นทั้งที่ IGBT ยังคงหยุดนำกระแสอยู่ ซึ่งแตกต่างจากเมื่อไม่ได้ต่อมอเตอร์ดังรูปที่ 9.5 และ 9.6 ตรงนี้เกิดจากมอเตอร์ซึ่งเป็นโหลดชนิด Inductance ยังคงคายพลังงานออกมานั่นเอง



รูปที่ 9.7 แสดงสัญญาณที่ขั้วมอเตอร์ ขณะต่อมอเตอร์ที่ Duty Cycle 70%



รูปที่ 9.8 แสดงสัญญาณที่ขั้วมอเตอร์ ขณะต่อมอเตอร์ที่ Duty Cycle 90%

## ซอฟต์แวร์

ในตอนต้นได้ทำการเขียนโปรแกรมซึ่งมีความซับซ้อนทำให้มี “บั๊ก” เกิดขึ้นเป็นจำนวนมาก ภายหลังจากได้ปรับโปรแกรมให้ง่ายมากขึ้นการตรวจสอบสามารถทำได้รวดเร็วและเข้าใจง่าย ทำให้โปรแกรมทำงานได้เป็นอย่างดี ซึ่งความยืดหยุ่นอาจขาดไปหากต้องการปรับ duty cycle จะต้องเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการปรับปรุงโปรแกรมอีกมากทีเดียว และจากสัญญาณที่วัดได้เห็นได้ว่าการอ่านค่าจาก Joystick หรือค่าที่รับจาก ADC0832 สามารถสื่อสารกันได้จริง มีการสร้างสัญญาณ Clock ที่ได้คำนวณไว้ ออกมาที่ 10kHz ได้และรับค่าจาก DO มาแปลงเป็นค่าที่เหมาะสมกับการขับมอเตอร์ต่อไป และสัญญาณที่ส่งไปยังภาค Drive Motor ได้ Duty Cycle ที่ 70% และ 90% ตามที่ต้องการ

### แมคคาณิก

ทางด้านโครงสร้างซึ่งเป็นเหล็กนั้น เมื่อประกอบเข้าด้วยกันจะมีความแข็งแรงมาก ๆ รับน้ำหนักมากๆ ได้ดี แต่ก็มีจุดที่ด้อยอยู่คือ เพลากับล้อ ซึ่งเพลาคูที่ใช้เป็นแกนของล้อจักรยานที่เสี้ยนผ่านศูนย์กลางไม่ใหญ่นักเมื่อเทียบกับน้ำหนักทั้งหมดรวมกับน้ำหนักของคนนั่งด้วย ซึ่งเดิมที่เป็นล้อจักรยานสำหรับเด็กทำให้อาจจะรับน้ำหนักได้ไม่มากนัก อาจเกิดอาการฉีกขาดของเพลาคู หรืออาจเกิดอาการ “ล้อเบะ” ได้ซึ่งจะได้ปรับปรุงในอนาคตแต่ทั้งหมดก็ยังสามารถทำให้ผู้จัดทำซึ่งหนัก 65 กิโลกรัม นั่งและวิ่งได้เป็นปกติ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 10

### แนวทางการพัฒนาต่อ

แบ่งออกเป็น 3 กลุ่มคือ 1. วงจร 2. ซอร์ฟแวร์ 3. แมคคาณิก มีแนวทางดังนี้คือ

#### วงจร

- ปรับปรุงเรื่องเสถียรภาพ การแก้ปัญหาสัญญาณรบกวนที่อาจเกิดขึ้น
- ปรับปรุงเรื่องเสถียรภาพ ป้องกันอุปกรณ์ชำรุดจากสัญญาณรบกวน , กระแสกระชาก
- ปรับปรุงเรื่องการหยุดของรถ ให้เบรกได้ โดยการช้อตขาของมอเตอร์เข้าด้วยกันเพื่อ ล็อคโรเตอร์
- ออกแบบลายวงจรให้ขนาดของแผ่น PCB เล็กกลงและใช้อุปกรณ์ประเภท SMD หรือ อุปกรณ์ติดพื้นผิวเพื่อช่วยให้มีขนาดที่เล็กลง และกินพลังงานน้อยลง
- เพิ่มขนาดของแบตเตอรี่ให้ใหญ่ขึ้น เพื่อที่จะวิ่ง ได้นานขึ้น
- มีวงจรชาร์จแบตเตอรี่ในตัวเพื่อชาร์จพลังงาน ได้ทุกที่
- มีวงจรวัดพลังงานที่เหลือในแบตเตอรี่

#### ซอร์ฟแวร์

- เพิ่มโปรแกรมเพื่อ offset ล้อทั้งสองข้างให้หมุนเท่ากัน
- ทำให้มี Speed ที่ Smooth คือเคลื่อนที่โดยไม่กระตุก
- เขียนโปรแกรมอ่านข้อมูล A to D ให้เร็วมากกว่านี้
- มีระบบเช็คสภาพตัวเองเพื่อหาจุดบกพร่องได้

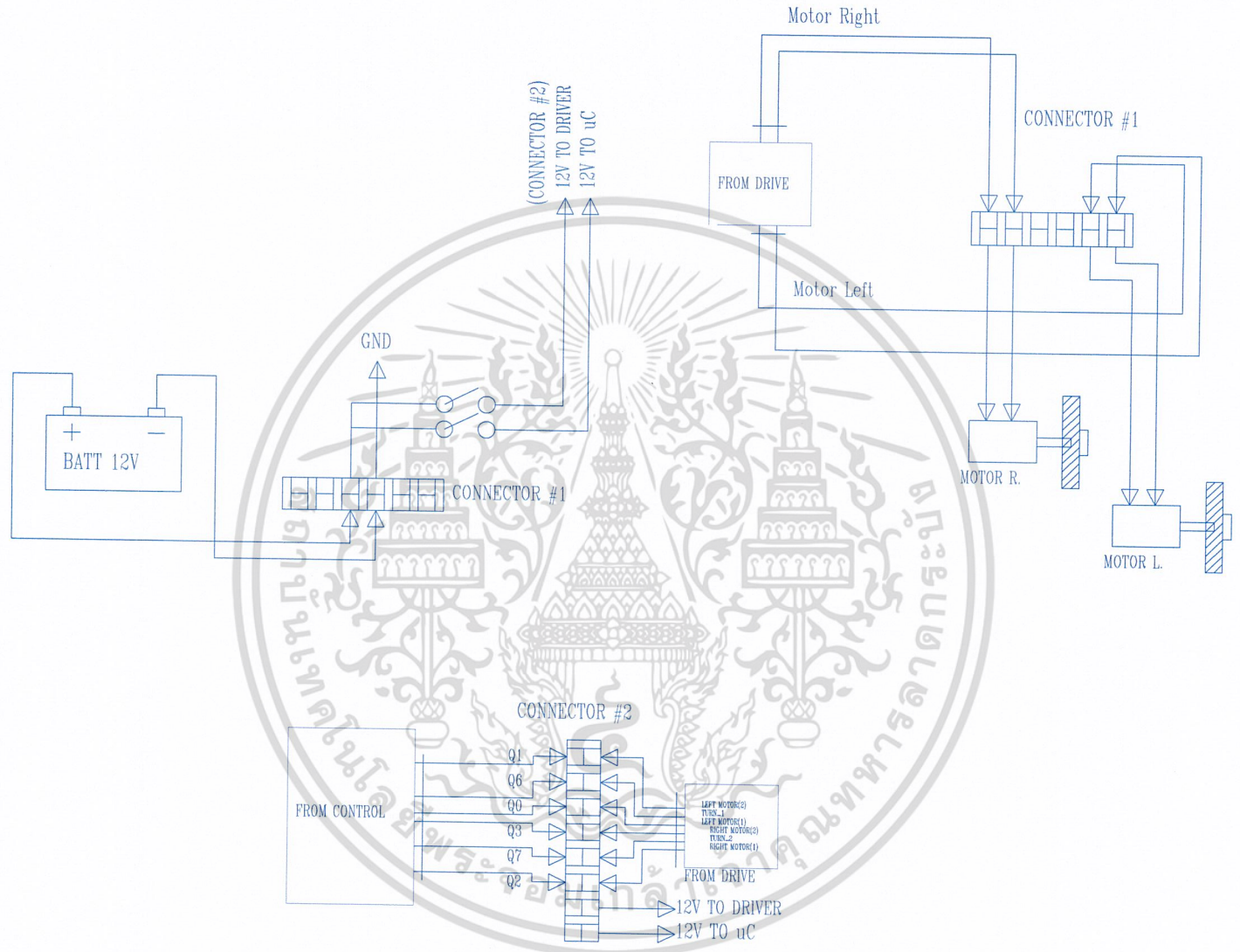
#### แมคคาณิก

- ทาสีกันสนิม และขัดส่วนที่คมออกให้หมด
- เปลี่ยนเพลและล้อให้แข็งแรงขึ้นมากกว่านี้
- เปลี่ยนเกียร์ให้แข็งแรงมากกว่านี้
- ปรับสมดุลของน้ำหนักให้การทรงตัวดีขึ้นไม่คว่ำง่าย
- ปรับให้ที่นั่งยกตัวขึ้นสูงได้ ซึ่งในต่างประเทศมีระบบไฮดรอลิกยกตัวเองขึ้นสูงได้ด้วย
- เปลี่ยนมอเตอร์ให้มีกำลังมากขึ้น ในกรณีที่ลากรถเงิน หรือเครื่องมืออื่นได้
- ต้องทำให้การดูแลรักษาง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

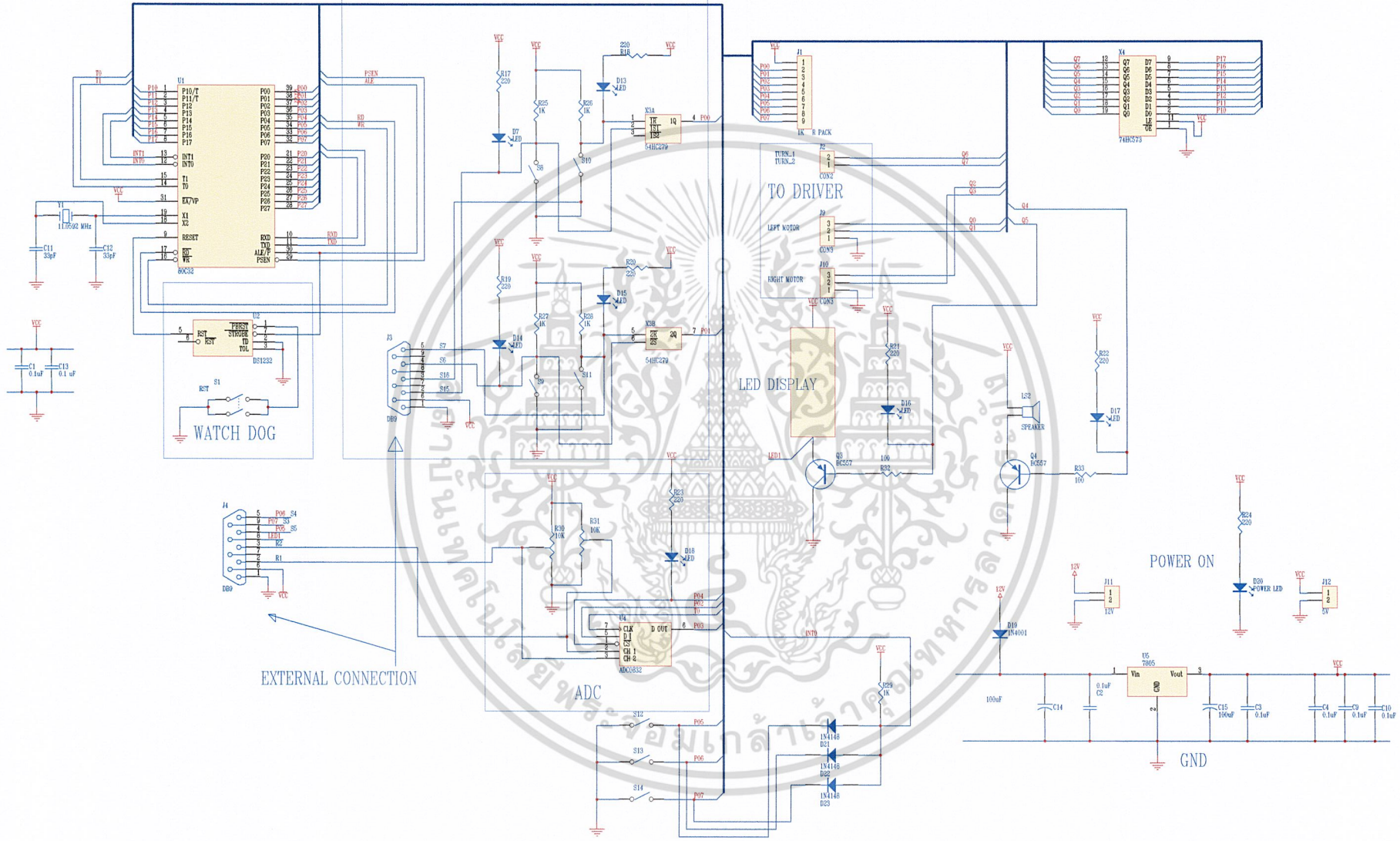


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title		
Size	Number	Revision
C		
Date:	21-Apr-2005	Sheet of
File:	C:\2004\sheet\control.DWG	Drawn By:

WHEEL COUNTER



CONNECTOR #1

Title		
Size	Number	Revision
C		
Date:	23-Apr-2005	Sheet: 01
File:	C:\doc\wheelcounter.DDB	Drawn By:

D

C

B

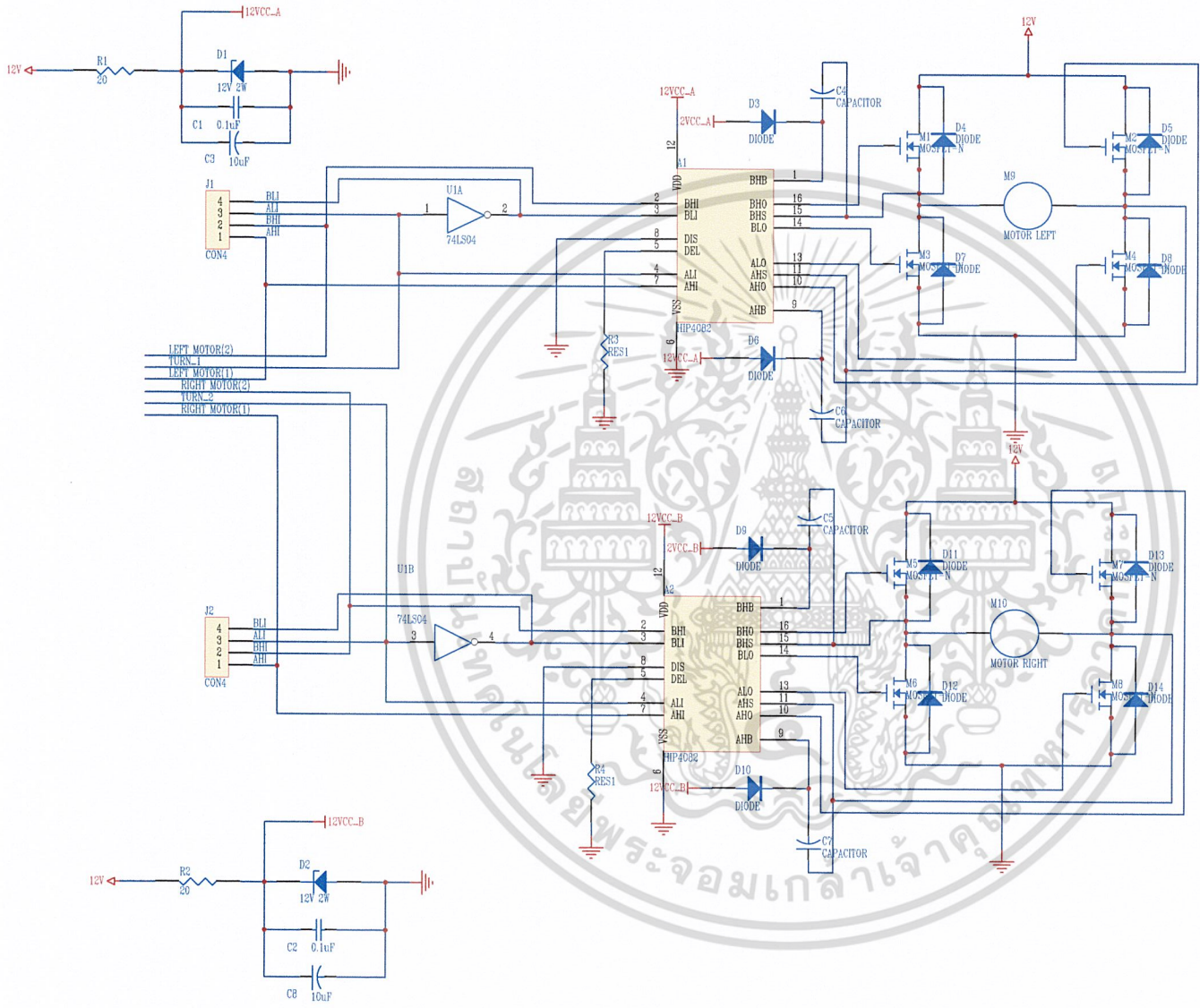
A

D

C

B

A



Title		
Size B	Number	Revision
Date: 23-Apr-2005	Sheet of	
File: C:\ppp\wheel\control.DDB	Drawn By:	

## กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลือของกลุ่มบุคคลดังต่อไปนี้ ผศ.พลผดุง ผดุงกุล อาจารย์ที่ปรึกษา คุณสุเทพ และคุณทองอยู่ ฉลภิญโญ บิดามารดา ด้านทุนทรัพย์ คุณกฤษ เรืองศรีนุกุลกิจ พิจารณาและปรับปรุงแปลน โครงสร้าง และกลุ่มบุคคลอีกหลายกลุ่มที่มีได้กล่าวไว้ในที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หนังสืออ้างอิง

1. รศ.ดร. สมเกียรติ สุภเดช “เซมิคอนดักเตอร์ดีไวซ์” แผนกตำราคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2545
2. อ. บัณฑิต จามรภูติ “คู่มือการใช้งาน Protel99” สำนักพิมพ์บัณฑิต 2544
3. วรพจน์ กรแก้ววัฒนะกุล, ชัยวัฒน์ ถิมพรจิตรวิไล “เรียนรู้และปฏิบัติการ ไมโครคอนโทรลเลอร์ MCS – 51 แบบเฟลช” บริษัท อิน โนเวตีฟ เอ็กเพอริเมนต์ จำกัด 2545
4. วารสาร SEMICONDUCTER ฉบับที่ 134 เดือน ธันวาคม พ.ศ. 2536
5. [PHY 406F -Microprocessor Interfacing Techniques by JamesR.Drummon -September 1996](#)
6. คู่มือคูหาไอซี บริษัท ซีเอ็ดยูเคชั่นจำกัด
7. ECG Semiconductor Master Replacement Guide



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้