

เรียนรู้และเข้าใจ

สถาปัตยกรรมไมโครคอนโทรลเลอร์

# PIC16F877

## Microchip PICmicro® MCUs

# PIC

คู่มือการทดลอง PIC  
**PIC16F877 และ PIC18F458**  
ด้วยภาษา Assembly และ ภาษา Basic  
CP-PIC V3/V3 EXP, ET-JWBOX300, ET-HARDWARE KIT V1



**ETT** บริษัท อีทีที จำกัด  
1112/96-98 ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ 10110 <http://www.etteam.com>  
1112/96-98 Sukhumvit Rd., Phrakonong Klongtoey BANGKOK 10110 <http://www.ett.co.th>  
TEL: 02-712 1120 FAX: 02-391 7216 e-mail: [sale@etteam.com](mailto:sale@etteam.com)



**ETT**  
[www.ett.co.th](http://www.ett.co.th)

บริษัท อีทีที จำกัด

**ETT CO., LTD.**

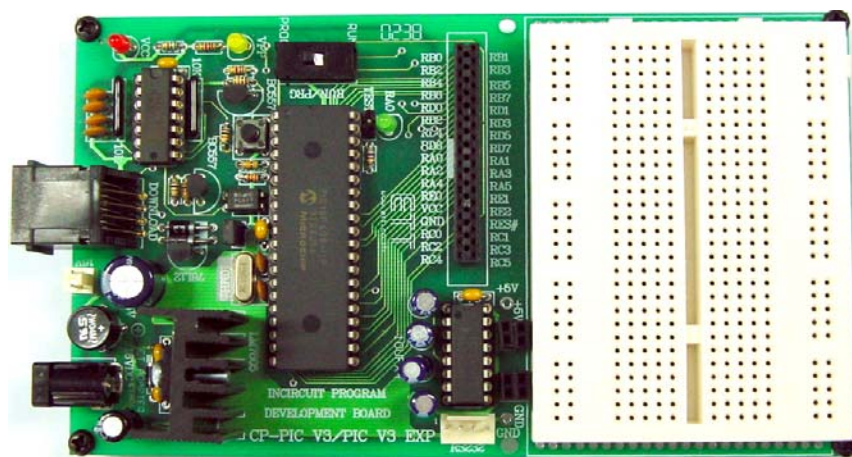
1112/96-98 ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ 10110 <http://www.etteam.com>

1112/96-98 Sukhumvit Rd., Phrakonong Klongtoey Bangkok 10110 <http://www.ett.co.th>

Tel : 02-7121120 Fax : 02-3917216

email : [sale@etteam.com](mailto:sale@etteam.com)

## PICmicro® Devices



## Microchip PICmicro® MCUs

ชื่อหนังสือ “เรียนรู้และเข้าใจสถาปัตยกรรมไมโครคอนโทรลเลอร์ PIC16F877”

ISBN 974-91058-6-9

ผู้เขียน นายวัชรินทร์ เคารพ

พิมพ์ครั้งที่ 1 จำนวน 1000 เล่ม

7 มีนาคม 2546

จำนวน 240 หน้า

ราคา 220 บาท

สงวนลิขสิทธิ์ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537  
ห้ามลอกเลียนไม่ว่าส่วนหนึ่งส่วนใดของหนังสือเล่มนี้  
ไม่ว่าในรูปแบบใดนอกจากจะได้รับอนุญาตเป็นลาย  
ลักษณ์อักษรจากผู้จัดพิมพ์

จัดพิมพ์โดย

บริษัท อีทีที จำกัด

1112/96-98 ถนนสุขุมวิท แขวงพระโขนง

เขตคลองเตย กรุงเทพฯ 10110

โทร. (02) 712-1120 - 1 FAX (02) 391-7216.

**ETT**  
www.ett.co.th

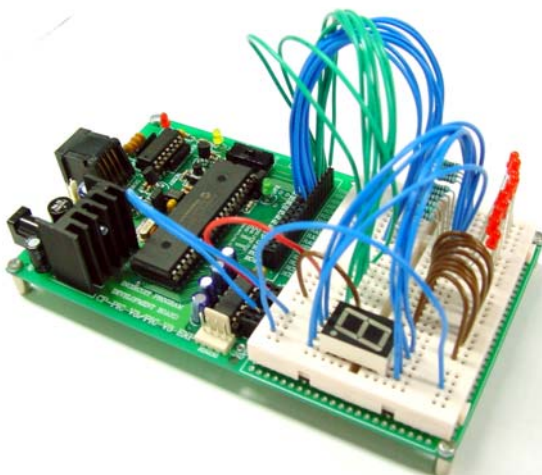
## คำนำ

หนังสือเล่มนี้ จัดทำขึ้นเพื่อประกอบการเรียนรู้ และ ใช้งานไมโครคอนโทรลเลอร์ในตระกูล PIC เบอร์ 16F877 ซึ่งเป็นไมโครคอนโทรลเลอร์ ที่ได้รับความนิยมอย่างแพร่หลายในปัจจุบัน แต่ในเรื่องของแหล่งข้อมูล หรือ หนังสือตำราต่างๆ ที่จะนำมาใช้ในการอ้างอิง และ ศึกษาเกี่ยวกับ CPU เบอร์นี้ยังมีอยู่น้อย หรือ ส่วนใหญ่ก็จะเป็นภาษาอังกฤษ ทำให้เป็นอุปสรรคต่อการศึกษารเรียนรู้ ใน CPU ดังกล่าว ด้วยเหตุนี้จึงได้มีการจัดทำหนังสือเล่มนี้ขึ้นมา เพื่อให้เกิดความสะดวกในการศึกษาใช้งาน CPU เบอร์นี้มากยิ่งขึ้น

เหตุที่เลือก CPU เป็นเบอร์ PIC 16F877 ก็เพราะเป็น CPU ที่มีคุณสมบัติ และ ฟังก์ชันการทำงานต่างๆ ที่ค่อนข้างครบถ้วนและ มีโครงสร้างการทำงานที่ไม่ยุ่งยากมากนัก สามารถศึกษาไว้เป็นพื้นฐานสำหรับการใช้งานไมโครคอนโทรลเลอร์ในเบอร์อื่นๆ ที่สูงขึ้นไปได้เป็นอย่างดี ซึ่งไมโครคอนโทรลเลอร์ในตระกูล PIC นี้จะมีโครงสร้างการทำงานที่ใกล้เคียงกัน แม้ว่า จะแบ่งออกเป็นแบบ 12 Bit Core ,14Bit Core หรือ 16 Bit Core ก็ตาม ซึ่งในเบอร์ 16F877 นี้จะเป็นแบบ 14 Bit Core คือ มีโครงสร้างของคำสั่งเป็นแบบ 14 บิต และ ข้อดีของ CPU ในตระกูล PIC นี้ที่เห็นได้ชัดก็คือเรื่องของ การจ่าย และ การรับกระแสของขาสัญญาณ I/O พอร์ตต่างๆ ที่สามารถจ่ายและรับกระแสสูงสุดได้ถึง 25 mA ทำให้สามารถนำไปขับโหลดต่างๆ ได้เป็นอย่างดีโดยไม่จำเป็นต้องต่อบัฟเฟอร์ หรือ วงจรรขยายกระแสอื่นๆแต่อย่างใด ประกอบกับมีฟังก์ชันในการทำงานต่างๆ มากมาย เช่น Power-on Reset, Power-up Timer, Oscillator Start-up Timer, Watchdog Timer, SLEEP Mode, Brown-out Reset, Code Protection, PWM, Timer/Counter, A to D และ USART เป็นต้น ซึ่งส่วนต่างๆ เหล่านี้ท่านสามารถเรียนรู้ได้ จากเนื้อหาต่างๆ ภายในหนังสือเล่มนี้

ภายในหนังสือเล่มนี้จะประกอบไปด้วยเนื้อหาต่างๆ มากมายโดยจะแบ่งออกเป็นบทๆ ตามประเภทของเนื้อหา และ ฟังก์ชันการทำงานต่างๆ ของ CPU โดยในท้ายบทของเนื้อหาจะมีตัวอย่างการทดลองต่างๆ ตามฟังก์ชันของเนื้อหาในบทนั้นๆ อีกด้วย ทั้งนี้ก็เพื่อให้เกิดความเข้าใจและเห็นภาพการนำไปใช้งานต่างๆ ที่เป็นรูปธรรมมากขึ้น โดยตัวอย่างที่ให้ไว้กับหนังสือเล่มนี้ จะอ้างอิงกับบอร์ดไมโครคอนโทรลเลอร์ PIC คือ CP-PIC V3 PLUS EXP ซึ่งเป็นบอร์ดไมโครคอนโทรลเลอร์ PIC เบอร์ 16F877-20 โดยความถี่ในการใช้งานของบอร์ดนี้คือ 10 MHz และ เป็นบอร์ดที่ใช้การโปรแกรมแบบแรงดันไฟสูง (High Voltage ICSP Program) ทำให้สามารถใช้งานขาสัญญาณ I/O ได้ครบทั้ง 33 ขาสัญญาณ ซึ่งท่านสามารถนำมาศึกษาทดลองร่วมกับชุด ET-HARDWARE KID V1 (ประกอบไปด้วยอุปกรณ์อิเล็กทรอนิกส์พื้นฐานต่างๆ) และ ชุด ET-JW BOX300 (ประกอบด้วยสายต่อวงจร JUMPER ขนาดต่างๆ อย่างดี) ของทาง บริษัท อีทีที ซึ่งสิ่งต่างๆ เหล่านี้ได้จัดเตรียมไว้เพื่อความสะดวกในการศึกษา เรียนรู้ทดลอง เกี่ยวกับไมโครคอนโทรลเลอร์นี้โดยเฉพาะ

ซึ่งทางเรา บริษัท อีทีที หวังเป็นอย่างยิ่งว่าหนังสือเล่มนี้จะช่วยให้ท่านสามารถเกิดการเรียนรู้ และ เข้าใจ จนถึงขั้นนำไปประยุกต์ใช้งานต่างๆ ที่ท่านต้องการได้เป็นอย่างดี



วัชรินทร์ เคาพ  
บริษัท อีทีที จำกัด

สารบัญ

<b>บทที่ 1 โครงสร้างและสถาปัตยกรรม</b>	<b>1</b>
คุณสมบัติของ PIC 16F877	1
<b>บทที่ 2 หน่วยความจำ</b>	<b>7</b>
2.1 หน่วยความจำโปรแกรม	7
2.2 หน่วยความจำข้อมูล (Data Memory)	8
2.3 รีจิสเตอร์ PCL และ PCLATH	17
2.4 การเข้าถึงข้อมูลแบบทางอ้อม (Indirect Addressing)	19
2.5 หน่วยความจำข้อมูล EEPROM	20
<b>บทที่ 3 การใช้งาน I/O พอร์ต</b>	<b>21</b>
3.1 การใช้งานพอร์ต A (PORTA)	21
3.2 การใช้งานพอร์ต B (PORTB)	24
3.3 การใช้งานพอร์ต C (PORTC)	26
3.4 การใช้งานพอร์ต D (PORTD)	28
3.5 การใช้งานพอร์ต E (PORTE)	29
ตัวอย่างการทดลองที่ 3.1 การใช้งานพอร์ต A เป็นเอาต์พุตดิจิทัล	32
ตัวอย่างการทดลองที่ 3.2 การใช้งานพอร์ต B เป็นอินพุต และ พอร์ต D เป็นเอาต์พุต	35
<b>บทที่ 4 ฟังก์ชันพิเศษต่างๆของ CPU</b>	<b>37</b>
4.1 การกำหนดโหมดของสัญญาณนาฬิกา	39
4.2 การรีเซตซีพียู (CPU)	41
4.3 โหมดประหยัดพลังงาน (SLEEP Mode)	47
<b>บทที่ 5 ชุดคำสั่งของ PIC 16F877</b>	<b>50</b>
5.1 ไซเคิลการทำงานของ CPU	50
5.2 ลักษณะการทำงานแบบไพป์ไลน์ (Pipe Line)	51
5.3 คำสั่งในภาษาแอสเซมบลี	52
<b>บทที่ 6 ภาษาแอสเซมบลี</b>	<b>85</b>
6.1 ส่วนประกอบต่างๆของภาษาแอสเซมบลี	85
6.2 การกำหนดค่าคงที่ในโปรแกรม	87
6.3 คำสั่งเทียม (Directive)	88
6.4 แนะนำการใช้งาน MPLAB	90
<b>บทที่ 7 EEPROM และ Flash Memory</b>	<b>100</b>
7.1 หน่วยความจำ EEPROM	101
7.2 หน่วยความจำแบบ FLASH	104
ตัวอย่างการทดลองที่ 7.1 การทดสอบการ อ่านและ เขียนข้อมูลในหน่วยความจำ EEPROM ภายใน	109
ตัวอย่างการทดลองที่ 7.2 การทดลองเขียนและ อ่านค่าในหน่วยความจำแบบแฟลช (Flash Memory)	113
<b>บทที่ 8 Timer Counter</b>	<b>117</b>
8.1 ไทเมอร์ 0 ( Timer0 Module)	117
8.2 ไทเมอร์ 1 ( Timer 1 Module )	123
8.2 ไทเมอร์ 2 (Timer 2 Module)	130

ตัวอย่างการทดลองที่ 8.1 โปรแกรมกำเนิดความถี่ 100 Hz โดยไทเมอร์ 0 (ใช้การอินเทอร์รัพท์)	133
ตัวอย่างการทดลองที่ 8.2 การทำงานของไทเมอร์ 0 ในโหมดการนับ	138
ตัวอย่างการทดลองที่ 8.3 โปรแกรมกำเนิดความถี่ 100 Hz ด้วยไทเมอร์ 1	141
ตัวอย่างการทดลองที่ 8.4 การทำงานของไทเมอร์ 1 ในโหมดการนับ (Counter Mode)	146
ตัวอย่างการทดลองที่ 8.5 โปรแกรมกำเนิดความถี่ 100 Hz ด้วยไทเมอร์ 2	150
<b>บทที่ 9 Capture/Compare/PWM</b>	<b>155</b>
9.1 โมดูล CCP1 (Capture/Compare/PWM 1)	155
9.2 โหมดตรวจจับสัญญาณ (Capture Mode)	157
9.3 โหมดการเปรียบเทียบข้อมูล (Compare Mode)	159
9.4 โหมดการสร้างสัญญาณ PWM (Pulse Width Modulation)	161
ตัวอย่างการทดลองที่ 9.1 การทำงานของโมดูล CCP1 ในโหมดตรวจจับสัญญาณ	166
ตัวอย่างการทดลองที่ 9.2 การทดสอบการทำงานของโหมดการเปรียบเทียบ	170
ตัวอย่างการทดลองที่ 9.3 การทดสอบการทำงาน ในโหมดการสร้างสัญญาณพัลส์วidthมอด	174
<b>บทที่ 10 โมดูล Analog to Digital Converter</b>	<b>179</b>
10.1 กระบวนการแปลงสัญญาณของโมดูล A/D	180
10.2 การทำงานของ A/D ในโหมดประหยัดพลังงาน (Sleep Mode)	183
10.3 ผลกระทบต่อโมดูล A/D จากการเกิดรีเซ็ต	183
ตัวอย่างการทดลองที่ 10.1 การทำงานของโมดูล A/D (Analog to Digital Converter)	184
<b>บทที่ 11 การสื่อสารอนุกรม USART</b>	<b>187</b>
11.1 รีจิสเตอร์ควบคุมการรับส่งข้อมูล	187
11.2 อัตราความเร็วในการรับส่งข้อมูล (Baud Rate Generator : BRG)	189
11.3 การสื่อสารในโหมด อะซิงโครนัส (USART Asynchronous Mode)	191
11.4 การกำหนดการทำงานแบบ 9 บิต โดยใช้ในการตรวจจับแอดเดรส	195
11.5 การสื่อสารแบบซิงโครนัส (Synchronous Mode )	198
ตัวอย่างการทดลองที่ 11.1 การทำงานของโหมดการสื่อสารแบบอะซิงโครนัส	205
11.6 แนะนำการใช้งานโปรแกรม Hyper terminal	211
<b>บทที่ 12 การอินเทอร์รัพท์</b>	<b>214</b>
12.1 รีจิสเตอร์ควบคุมการอินเทอร์รัพท์	215
12.2 การเขียนโปรแกรมรองรับการอินเทอร์รัพท์ (Interrupt Service Routine)	220
12.3 การอินเทอร์รัพท์ภายนอก RB0/INT	222
12.5 การอินเทอร์รัพท์จากการเปลี่ยนแปลงที่ขาสัญญาณ PORTB<7:4>	223
12.6 การใช้งานอินเทอร์รัพท์หลายๆ ส่วนพร้อมกัน	224
ตัวอย่างการทดลองที่ 12.1 การทดสอบการทำงานของอินเทอร์รัพท์ภายนอก INT	225
ตัวอย่างการทดลองที่ 12.2 การอินเทอร์รัพท์จากการเปลี่ยนแปลงที่ขาสัญญาณ RB7:RB4	230



## บทที่ 1

## โครงสร้างและสถาปัตยกรรม

ไมโครคอนโทรลเลอร์ได้มีการคิดค้นพัฒนาอย่างต่อเนื่อง จนในปัจจุบัน มีไมโครคอนโทรลเลอร์ตระกูลใหม่ๆ เกิดขึ้นมามากมาย ล้วนแล้วแต่มีศักยภาพในการทำงานสูงด้วยกันทั้งสิ้น แต่ในที่นี้เราจะมาศึกษาการใช้งานไมโครคอนโทรลเลอร์ตระกูล PIC ของทางบริษัท Microchip ซึ่งเป็นไมโครคอนโทรลเลอร์ที่ได้รับความนิยมอย่างแพร่หลายในบ้านเรา เนื่องจากเป็นไมโครคอนโทรลเลอร์ที่มีความสามารถ และ เพียบพร้อมไปด้วยทรัพยากร หรือ ฟังก์ชันการใช้งานต่างๆ มากมาย เช่น โมดูล Analog to Digital , Timer/Counter , USART , SPI , I<sup>2</sup>C , PWM และ อื่นๆ ซึ่งส่วนต่างๆ เหล่านี้จะถูกสร้างรวมอยู่ภายใน CPU เพียงตัวเดียว ทำให้ CPU เพียงตัวเดียวนี้ สามารถทำงานได้หลายๆ อย่าง และสามารถลดในส่วนของฮาร์ดแวร์บางอย่างลง ส่วนในเรื่องของความเร็ว CPU ตระกูลนี้จะใช้เวลาในการกระทำคำสั่งต่างๆ เพียง 1 หรือ 2 ไชเคิล ต่อคำสั่งเท่านั้น โดยการทำงานนี้จะเป็นลักษณะ ไลน์ (Pipe Line) ทำให้มีความเร็วในการทำงานมากกว่า CPU ทั่วไป (ที่ความถี่เดียวกัน) ในเบอร์ที่เราจะได้ศึกษาในหนังสือเล่มนี้เป็นเบอร์ 16F877 ซึ่งเป็นเบอร์ที่ได้รับความนิยมอีกเบอร์หนึ่งโดย ในหนังสือเล่มนี้จะแบ่งหัวข้อการศึกษาออกเป็น บทๆ ตามโหมด หรือ ฟังก์ชันการทำงานของไมโครคอนโทรลเลอร์

## คุณสมบัติของ PIC 16F877

คุณสมบัติต่างๆ ของไมโครคอนโทรลเลอร์ PIC 16F877 สามารถสรุปอย่างคร่าวๆ ได้ดังนี้

- 35 Instruction คำสั่ง
- ในการปฏิบัติงานคำสั่งต่างๆ จะใช้ Cycle เดียว และ 2 Cycle ในคำสั่งที่เป็นการกระโดด
- ความถี่สูงสุดที่ทำงานได้คือ 20 MHz (16F877-20/P)
- การทำงานจะเป็นลักษณะ Pipeline ทำให้มีการทำงานที่เร็วขึ้น
- หน่วยความจำโปรแกรม FLASH Program Memory มีขนาด 8k (14-Bit Words)
- หน่วยความจำข้อมูล (RAM) 368 Bytes
- หน่วยความจำข้อมูล (EEPROM) 256 Bytes
- สามารถตอบสนองการอินเตอร์รัพต์ได้ถึง 14 แหล่ง
- STACK 8 ระดับ
- เพาเวอร์อนรีเซต(POR),เพาเวอร์อัปไทมเมอร์(PWRT) และ Oscillator Start-Up Timer
- Watchdog Timer
- สามารถเลือกการป้องกันข้อมูลได้ (Code Protection)
- โหมดประหยัดพลังงาน (Sleep Mode)
- เลือกโหมดของ สัญญาณนาฬิกาได้หลายโหมด
- สามารถโปรแกรมโดยใช้แรงดัน +5V ได้
- ฟังก์ชันการโปรแกรมแบบ ICSP (In-Circuit Serial Programing)
- ทำงานที่ไฟเลี้ยง 2.0V ถึง 5.5V
- กระแสทั้งซิงค์และซอร์สของพอร์ตคือ 25mA
- Timer/Counter จำนวน 3 ตัว คือ Timer0,Timer1 และ Timer2
- โมดูล Capture/Compare/PWM จำนวน 2 ชุด
- Analog to Digital Converter ความละเอียด 10 บิต 8 แชนเนล ภายในตัว

- มีโมดูลการสื่อสาร USART
- มีโมดูลตรวจจับระดับแรงดันไฟเลี้ยง Brown – out reset (BOR)
- มีพอร์ต I/O 5 พอร์ตประกอบด้วย A,B,C,D และ E แต่ละพอร์ตจะมีจำนวนบิตไม่เท่ากันซึ่งรวมแล้วจะมี I/O จำนวน 33 บิต

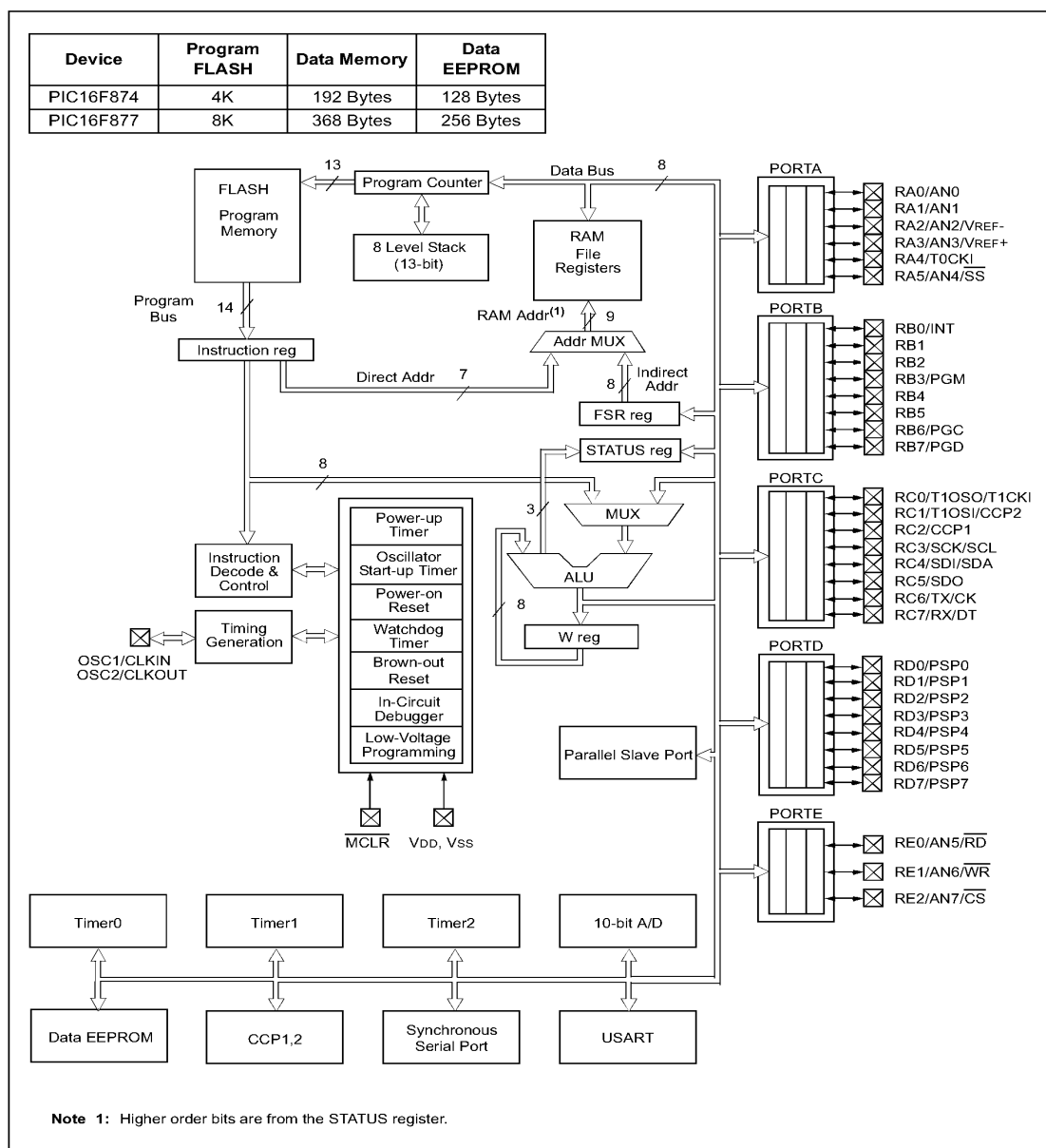
PORTA = RA5 – RA0 จำนวน 6 บิต

PORTB = RB7 – RB0 จำนวน 8 บิต

PORTC = RC7 – RC0 จำนวน 8 บิต

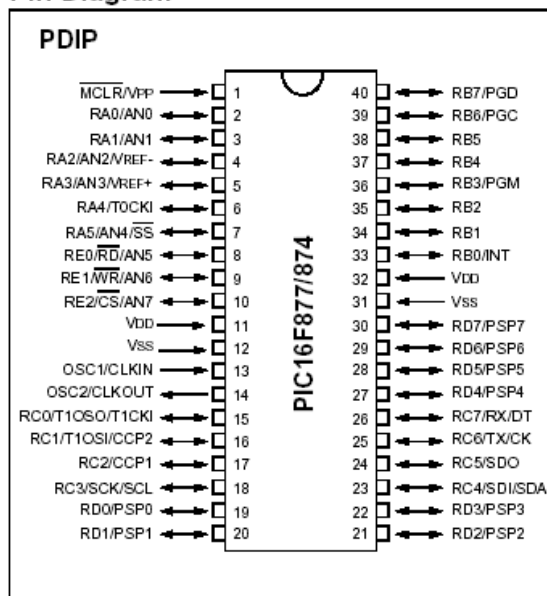
PORTD = RD7 – RD0 จำนวน 8 บิต

PORTE = RE2 – RE0 จำนวน 3 บิต



รูปที่ 1.1 แสดงสถาปัตยกรรมภายในของไมโครคอนโทรลเลอร์ PIC 16F877

Pin Diagram



รูปที่ 1.2 แสดงตัวถังของ CPU PIC 16F877 และ การจัดวางตำแหน่งขาสัญญาณต่างๆ

ขาสัญญาณของ PIC เบอร์ 16F877 นี้จะมีทั้งหมด 40 ขาจะประกอบไปด้วยขาที่ทำหน้าที่ต่างๆโดยจะมีขาสัญญาณ I/O Ports ทั้งหมดจำนวน 33 ขา โดยสามารถนำไปใช้ป็นอินพุต/เอาต์พุตได้ทั้งหมดทุกขา ยกเว้นขา RA4 ซึ่งโครงสร้างภายในเป็นแบบ Open Drain ดังนั้นหากต้องการนำไปใช้ป็นขาสัญญาณเอาต์พุต จะต้องต่อตัวต้านทานพูลอัพ (Pull-up) ไว้ด้วย ส่วนขาที่เหลือ สามารถใช้งานได้ตามปกติ นอกจากขาสัญญาณ I/O แล้ว ยังประกอบไปด้วยขาสัญญาณอื่นๆ อีกคือ ขาไฟเลี้ยง , กราวด์ , ขารีเซ็ต และ ขาออสซิลเลเตอร์ ซึ่งสามารถสรุปหน้าที่การทำงานของขาสัญญาณในแต่ละขาได้ดังตารางต่อไปนี้



ขาสัญญาณ	DIP PIN	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	I	ST/CMOS	ขาสัญญาณอินพุตสัญญาณนาฬิกาของ CPU
OSC2/CLKOUT	14	O	-	เป็นขาเอาต์พุตของสัญญาณนาฬิกา (1/4 ของ CLKIN) , ใช้ต่อร่วมกับขาสัญญาณ OSC1 เพื่อกำหนดสัญญาณนาฬิกา ในกรณีที่ใช้ คริสตัลแบบเรโซเนเตอร์ หรือ วงจร RC ภายนอก
MCLR/VPP	1	I/P	ST	ขาสัญญาณรีเซ็ตแควตซ์ที่ฟ “0” , ขารับแรงดันสำหรับการโปรแกรม
RA0/AN0	2	I/O	TTL	ขาสัญญาณอินพุต/เอาต์พุต PORTA - ขาสัญญาณ อินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณ อนาลอก AN0
RA1/AN1	3	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณ อนาลอก AN1
RA2/AN2/VREF-	4	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณ อนาลอก AN2 และ ขาสัญญาณแรงดันอ้างอิงลบ (-)
RA3/AN3/VREF+	5	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณ อนาลอก AN3 และ ขาสัญญาณแรงดันอ้างอิงบวก (+)
RA4/T0CKI	6	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณ นาฬิกาของ ไทม์เมอร์ 0
RA5/SS/AN4	7	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณ อนาลอก AN4 และ ขาสัญญาณ Slave Select ในโหมดการ สื่อสาร แบบ Synchronous Serial Port
RB0/INT	33	I/O	TTL/ST	ขาสัญญาณอินพุต/เอาต์พุต PORTB - ขาสัญญาณอินพุต/เอาต์พุต , ขาอินพุตสัญญาณอินเตอร์รัพท์ภายนอก
RB1	34	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต
RB2	35	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต
RB3/PGM	36	I/O	TTL	- ขาสัญญาณอินพุตเอาต์พุต , ขาสัญญาณการโปรแกรม แบบแรงดันต่ำ
RB4	37	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินเตอร์รัพท์จาก การเปลี่ยนแปลงสถานะของขาสัญญาณ

ขาสัญญาณ	DIP PIN	I/O/P Type	Buffer Type	Description
RB5	38	I/O	TTL	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินเทอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ
RB6/PGC	39	I/O	TTL/ST <sup>(2)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินเทอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ และ เป็นขาสัญญาณนาฬิกาในโหมดการโปรแกรม
RB7/PGD	40	I/O	TTL/ST <sup>(2)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินเทอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ และ เป็นขาสัญญาณข้อมูลในโหมดการโปรแกรม
RC0/T1OSO/T1CKI	33	I/O	ST	ขาสัญญาณอินพุต/เอาต์พุต PORTC - ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณออสซิลเลเตอร์เอาต์พุต และ ขาอินพุตสัญญาณนาฬิกาของไทมเมอร์ 1
RC1/T1OSI/CCP2	16	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตออสซิลเลเตอร์ไทมเมอร์ 1 และ เป็นขาสัญญาณ Capture 2 input /Compare 2 output /PWM2 output
RC2/CCP1	17	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต และ ขาสัญญาณ Capture 1 input /Compare 1 output /PWM1 output
RC3/SCK/SCL	18	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณอินพุตสัญญาณนาฬิกาในการสื่อสาร Synchronous , ขาสัญญาณนาฬิกาในโหมด I <sup>2</sup> C และ SPI
RC4/SDI/SDA	23	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต , ขาอินพุตสัญญาณข้อมูลในโหมด SPI , ขาอินพุต/เอาต์พุตสัญญาณข้อมูลในโหมด I <sup>2</sup> C
RC5/SDO	24	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต , ขาเอาต์พุตสัญญาณข้อมูลในโหมด SPI
RC6/TX/CK	25	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต , ขาเอาต์พุตสัญญาณด้านส่งของการสื่อสาร USART , ขาสัญญาณนาฬิกาในโหมดการสื่อสาร Synchronous
RC7/RX/DT	26	I/O	ST	- ขาสัญญาณอินพุต/เอาต์พุต , ขาอินพุตสัญญาณด้านรับของการสื่อสาร USART , ขาสัญญาณข้อมูลในโหมดการสื่อสาร Synchronous

ขาสัญญาณ	DIP PIN	I/O/P Type	Buffer Type	Description
RD0/PSP0	19	I/O	ST/TTL <sup>(3)</sup>	ขาสัญญาณอินพุต/เอาต์พุต PORTD สามารถใช้เป็น Slave Port กรณีติดต่อกับระบบบัสของไมโครโปรเซสเซอร์
RD1/PSP1	20	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RD2/PSP2	21	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RD3/PSP3	22	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RD4/PSP4	27	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RD5/PSP5	28	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RD6/PSP6	29	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RD7/PSP7	30	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต
RE0/RD/AN5	8	I/O	ST/TTL <sup>(3)</sup>	ขาสัญญาณอินพุตเอาต์พุต PORTE
RE1/WR/AN6	9	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณควบคุมการอ่านในโหมด Parallel Slave Port และเป็นขาสัญญาณอินพุตสัญญาณอนาล็อก AN5
RE2/CS/AN7	10	I/O	ST/TTL <sup>(3)</sup>	- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณควบคุมการเขียนในการติดต่อแบบ Parallel Slave Port และ เป็นขาสัญญาณอินพุตสัญญาณอนาล็อก AN6
				- ขาสัญญาณอินพุต/เอาต์พุต , ขาสัญญาณควบคุม Control Select ในการติดต่อแบบ Parallel Slave Port และ เป็นขาสัญญาณอินพุตสัญญาณอนาล็อก AN7
Vss	12,31	P	-	ขาสัญญาณกราวด์
VDD	11,32	P	-	ขาแรงดันไฟเลี้ยง

I = อินพุต

O = เอาต์พุต

I/O = อินพุต/เอาต์พุต

P = Power

- = ไม่ใช้

TTL = ระดับสัญญาณ TTL

ST = วงจรสมิตทริกเกอร์ อินพุต

#### Note

1: Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อมีการใช้งานอินเตอร์รัพท์ภายนอก

2: Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อทำงานในโหมดการโปรแกรม

3: Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อใช้งานทั่วไป และ จะเป็นแบบ TTL เมื่ออยู่ในโหมด Parallel Slave Port ในการติดต่อกับระบบบัสของไมโครโปรเซสเซอร์

4: Buffer จะเป็นแบบ ชมิททริกเกอร์เมื่อใช้งานในโหมด RC oscillator และจะเป็นแบบ CMOS เมื่อใช้งานอื่นๆ