



เอกสารประกอบการสอน

วิชาไมโครคอนโทรลเลอร์ รหัสวิชา 3104 - 2011

หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) พุทธศักราช 2545

(ปรับปรุง 2546)

ประเภทวิชาช่างอุตสาหกรรม สาขาวิชาไฟฟ้ากำลัง

จัดทำโดย

นายธีรวัฒน์ ติบอ้าย

แผนกวิชาช่างไฟฟ้ากำลัง

สำนักงานคณะกรรมการการอาชีวศึกษา

วิทยาลัยเทคนิคน่าน

กระทรวงศึกษาธิการ

เอกสารประกอบการสอน (ใบความรู้)

วิชาไมโครคอนโทรลเลอร์ รหัสวิชา 3104 - 2011

หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) พุทธศักราช 2545

(ปรับปรุง 2546)

ประเภทวิชาช่างอุตสาหกรรม สาขาไฟฟ้ากำลัง

จัดทำโดย

นายธีรวัฒน์ ตีบ้าย

แผนกวิชาไฟฟ้ากำลัง

สำนักงานคณะกรรมการการอาชีวศึกษา

วิทยาลัยเทคนิคน่าน

กระทรวงศึกษาธิการ

หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง พุทธศักราช 2546

ประเภทวิชาอุตสาหกรรม

สาขาวิชาไฟฟ้ากำลัง

จุดประสงค์

ผู้ที่สำเร็จการศึกษาหลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง พุทธศักราช 2546 ประเภทวิชา อุตสาหกรรม สาขาวิชาไฟฟ้ากำลัง สามารถปฏิบัติงานระดับช่างเทคนิค ผู้ควบคุมงาน ผู้ช่วยวิศวกร หรือประกอบอาชีพส่วนตัวมีความรู้ ความสามารถ เจตคติ และประสบการณ์ด้านต่างๆ ดังต่อไปนี้

1. เพื่อให้มีความรู้และทักษะพื้นฐานเกี่ยวกับภาษา สังคม มนุษยศาสตร์ คณิตศาสตร์ วิทยาศาสตร์ นำไปใช้ในการค้นคว้า พัฒนาตนเองและวิชาชีพช่างไฟฟ้าให้เกิดความเจริญก้าวหน้า
2. เพื่อให้มีความรู้และทักษะในหลักการและกระบวนการทำงานพื้นฐานของช่างเทคนิคที่ เกี่ยวกับการบริหารจัดการและการวางแผนในงานอุตสาหกรรม และสามารถติดตาม ความเจริญก้าวหน้าทางเทคโนโลยีนำมาพัฒนางานอาชีพช่างไฟฟ้าให้มีประสิทธิภาพ และประสิทธิผล
3. เพื่อให้มีความคิดวิเคราะห์ แก้ปัญหา สร้างสรรค์และนำเทคโนโลยีมาใช้ในการพัฒนา งานไฟฟ้ากำลัง
4. เพื่อให้มีบุคลิกภาพที่ดี มีความรับผิดชอบต่อตนเอง ครอบครัวและสังคม มีคุณธรรม จริยธรรมและกิริยาที่ดีในงานอาชีพ
5. เพื่อให้สามารถประกอบอาชีพในสถานประกอบการอุตสาหกรรมหรือสร้างสรรค์หรือ ประกอบอาชีพอิสระในสาขาวิชาไฟฟ้ากำลัง

หลักสูตรรายวิชา ไมโครคอนโทรลเลอร์ รหัสวิชา 3104-2011

จำนวน 3 หน่วยกิต จำนวนชั่วโมง/สัปดาห์ 4 ชั่วโมง จำนวนชั่วโมงรวม 72 ชั่วโมง

จุดประสงค์รายวิชา

1. เพื่อให้มีความรู้ ความเข้าใจ โครงสร้าง ระบบ และการทำงานของไมโครคอนโทรลเลอร์
2. ให้มีความสามารถใช้เครื่องมือเขียนโปรแกรมควบคุมการติดต่อระบบ และตรวจสอบความผิดพลาดของโปรแกรม
3. เพื่อประยุกต์ใช้งานไมโครคอนโทรลเลอร์ในการควบคุมระบบไฟฟ้าได้อย่างเหมาะสม
4. เพื่อให้มีกิจนิสัยในการทำงานด้วยความประณีต ปลอดภัยอย่างมีคุณธรรม จริยธรรม

มาตรฐานรายวิชา

1. เข้าใจโครงสร้าง วงจรประกอบ และการทำงานของไมโครคอนโทรลเลอร์
2. ใช้เครื่องมือโปรแกรมประกอบในการเขียน และทดสอบโปรแกรม
3. เขียนโปรแกรมควบคุมการทำงานส่วนต่างๆ ในระบบไมโครคอนโทรลเลอร์
4. ประยุกต์ใช้งานไมโครคอนโทรลเลอร์ในการควบคุมระบบไฟฟ้าแบบต่าง ๆ

คำอธิบายรายวิชา

ศึกษาโครงสร้างของไมโครคอนโทรลเลอร์ วงจรประกอบการทำงานของไมโครคอนโทรลเลอร์ รีจิสเตอร์ ระบบบัสและการทำงาน ชุดคำสั่งของไมโครคอนโทรลเลอร์ ใช้เครื่องมือเขียนโปรแกรมและโปรแกรมช่วยในการทดสอบ โดยใช้ชุดคำสั่งของไมโครคอนโทรลเลอร์ ควบคุมการติดต่อซีพียูกับหน่วยความจำ เอาท์พุต อินพุต การอินเตอร์รัพต์ การติดต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกแบบเบื้องต้นและผ่านระบบสื่อสาร การประยุกต์ใช้งานในระบบควบคุมทางไฟฟ้าแบบต่าง ๆ

โครงการสอน

หน่วยที่	ชื่อเรื่อง	จำนวนชั่วโมง
1	โครงสร้างไมโครคอนโทรลเลอร์	8
	1.1 ความหมายของไมโครคอนโทรลเลอร์	
	1.2 โครงสร้างไมโครคอนโทรลเลอร์	
	1.3 ระบบบัสและการทำงาน	
	1.4 หน่วยความจำและรีจิสเตอร์	
	1.5 วงจรประกอบการทำงานของไมโครคอนโทรลเลอร์	
	ใบงานที่ 1 การทดสอบวงจรการทำงานของไมโครคอนโทรลเลอร์	
2	ชุดคำสั่งของไมโครคอนโทรลเลอร์	8
	2.1 ชุดคำสั่งโปรแกรมภาษาเบสิก	
	2.2 ชุดคำสั่งโปรแกรมภาษาแอสเซมบลี	
	2.2 ชุดคำสั่งโปรแกรมภาษาซี	
	ใบปฏิบัติงานที่ 1 การเขียนรายละเอียดชุดคำสั่งซีซีเอสซี คอมไพเลอร์	
3	เครื่องมือเขียนชุดคำสั่งและโปรแกรมทดสอบการทำงาน	12
	3.1 โปรแกรมเขียนชุดคำสั่งภาษาซี	
	3.2 โปรแกรมทดสอบการทำงาน	
	3.3 โปรแกรมบันทึกภาษาเครื่อง	
	ใบงานที่ 2 การใช้โปรแกรมเขียนชุดคำสั่งภาษาซี	
	ใบงานที่ 3 การใช้โปรแกรมทดสอบการทำงาน	
	ใบงานที่ 4 การใช้โปรแกรมบันทึกภาษาเครื่อง	
4	การเขียนชุดคำสั่งควบคุมพอร์ตอินพุตเอาต์พุต	8
	4.1 ชุดคำสั่งควบคุมพอร์ตอินพุตเอาต์พุต	
	4.2 โครงสร้างขาพอร์ตของไมโครคอนโทรลเลอร์	
	ใบงานที่ 5 การเขียนชุดคำสั่งควบคุมพอร์ตเอาต์พุตแบบดิจิทัล	
	ใบงานที่ 6 การใช้ขาพอร์ต A เป็นพอร์ตอินพุตดิจิทัล	
5	การอินเตอร์รัพต์	4
	5.1 รีจิสเตอร์ที่เกี่ยวข้องกับอินเตอร์รัพต์	
	5.2 การเขียนชุดคำสั่งควบคุมการอินเตอร์รัพต์	
	5.3 ข้อความสั่งตัวประมวลผลก่อนการอินเตอร์รัพต์	

โครงการสอน (ต่อ)

หน่วยที่	ชื่อเรื่อง	จำนวนชั่วโมง
	5.4 ฟังก์ชันการอินเวอร์รต์จากคลังข้อมูล	
	ใบงานที่ 7 การอินเวอร์รต์จากไทเมอร์ 0 แบบวิธีการนับ	
6	การต่อประสานไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก	12
	6.1 การต่อประสานกับหน่วยแสดงผลแอลอีดีตัวเลข 7 ส่วน	
	6.2 การต่อประสานกับหน่วยแสดงผลแอลซีดีโมดูล	
	6.3 การต่อประสานกับเมทริกซ์สวิตช์	
	6.4 การต่อประสานกับโมดูลแปลงสัญญาณแอนะล็อกเป็นดิจิทัล	
	ใบงานที่ 8 การต่อประสานเมทริกซ์สวิตช์แสดงผลแอลอีดีตัวเลข 7 ส่วน	
	ใบงานที่ 9 การต่อประสานแผงเมทริกซ์สวิตช์แสดงผลแอลซีดี	
	ใบงานที่ 10 การเขียนชุดคำสั่งรับสัญญาณแอนะล็อกแปลงเป็นดิจิทัล	
7	ระบบและรูปแบบการสื่อสาร	8
	7.1 ระบบการสื่อสาร	
	7.2 การสื่อสารของพอร์ตอนุกรม	
	7.3 การสื่อสารกับอุปกรณ์แบบ I2C บัส	
	7.4 การต่อประสานอุปกรณ์แบบ 1 สาย	
	ใบงานที่ 11 การสื่อสารระบบบัสแบบ 1 สาย	
8	การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ควบคุมทางไฟฟ้า	12
	8.1 การออกแบบระบบต่อประสานไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก	
	8.2 การพัฒนาระบบควบคุมอุปกรณ์ไฟฟ้าด้วยไมโครคอนโทรลเลอร์	
	8.3 การประยุกต์ใช้ไมโครคอนโทรลเลอร์ควบคุมอุปกรณ์ไฟฟ้า	
	ใบงานที่ 12 การเขียนชุดคำสั่งควบคุมการรดน้ำต้นไม้อัตโนมัติ	
	ใบงานที่ 13 การเขียนชุดคำสั่งการควบคุมอุณหภูมิอัตโนมัติ	
	ใบงานที่ 14 การเขียนชุดคำสั่งควบคุมการสตาร์ทมอเตอร์ 3 เฟสแบบสตาร์ทตรง	
รวม		72

	ใบความรู้ที่ 1	หน่วยที่ 1
	ชื่อวิชา ไมโครคอนโทรลเลอร์ รหัสวิชา 3104-2011	สอนครั้งที่ 1-2
	ชื่อหน่วย โครงสร้างของไมโครคอนโทรลเลอร์	จำนวนชั่วโมงรวม 8 ชั่วโมง
	ชื่อเรื่อง โครงสร้างของไมโครคอนโทรลเลอร์	จำนวนชั่วโมงสอน 4 ชั่วโมง

สาระสำคัญ

ไมโครคอนโทรลเลอร์ คืออุปกรณ์อิเล็กทรอนิกส์ที่ใช้กับระบบควบคุมทางอิเล็กทรอนิกส์ในงานอุตสาหกรรมหรือควบคุมการทำงานของเครื่องใช้ไฟฟ้า สามารถเปลี่ยนแปลงลำดับการทำงานได้ด้วยการเปลี่ยนแปลงชุดคำสั่งที่บรรจุอยู่ในหน่วยความจำ ไมโครคอนโทรลเลอร์ใช้ควบคุมการทำงานของเครื่องใช้ไฟฟ้าหรือระบบควบคุมทางอิเล็กทรอนิกส์ ให้มีประสิทธิภาพในการทำงานสูงขึ้น มีโครงสร้างหลัก 5 หน่วย คือ หน่วยประมวลผลกลาง หน่วยเก็บข้อมูล หน่วยสำหรับต่อประสานกับอุปกรณ์ภายนอก วงจรกำเนิดสัญญาณนาฬิกา หน่วยบริการ การอินเตอร์รัพต์

สาระการเรียนรู้

1. ความหมายของไมโครคอนโทรลเลอร์
2. โครงสร้างของไมโครคอนโทรลเลอร์
3. ระบบบัสและการทำงาน
4. หน่วยความจำและรีจิสเตอร์
5. วงจรประกอบการทำงานของไมโครคอนโทรลเลอร์

จุดประสงค์การเรียนรู้

จุดประสงค์ทั่วไป

เพื่อให้มีความรู้และความเข้าใจเกี่ยวกับโครงสร้างของไมโครคอนโทรลเลอร์

จุดประสงค์เชิงพฤติกรรม

1. บอกความหมายของไมโครคอนโทรลเลอร์ได้ถูกต้อง
2. บอกโครงสร้างของไมโครคอนโทรลเลอร์ได้ถูกต้อง
3. อธิบายการทำงานของระบบบัสได้ถูกต้อง
4. อธิบายการทำงานของหน่วยความจำและรีจิสเตอร์ได้ถูกต้อง
5. อธิบายวงจรประกอบการทำงานของไมโครคอนโทรลเลอร์ได้ถูกต้อง

ในยุคปัจจุบันไมโครคอนโทรลเลอร์ได้เข้ามามีบทบาทต่อวงการอุตสาหกรรมเป็นอย่างมาก เนื่องจากใช้เป็นอุปกรณ์หลักควบคุมการทำงานของผลิตภัณฑ์ สามารถเขียนชุดคำสั่งเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ จึงนำมาใช้ควบคุมระบบอัตโนมัติของอุปกรณ์ไฟฟ้าและระบบอิเล็กทรอนิกส์อุตสาหกรรม เช่น ระบบอัตโนมัติของเครื่องจักร ระบบสมองกลฝังตัว เป็นต้น

1.1 ความหมายของไมโครคอนโทรลเลอร์

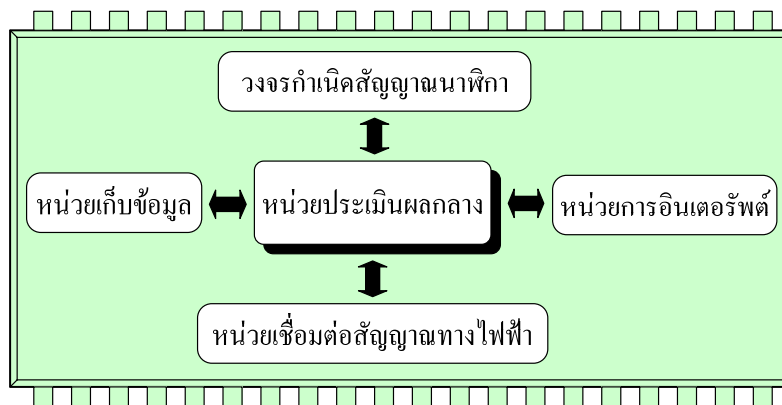
ไมโครคอนโทรลเลอร์ เป็นชื่อของอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่ง ที่รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และ วงจรรับสัญญาณอินพุต วงจรขับสัญญาณเอาต์พุต หน่วยความจำ วงจรกำเนิดสัญญาณไว้ด้วยกัน สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี ทำให้ลดจำนวนอุปกรณ์และขนาดของระบบ ในขณะที่มีขีดความสามารถสูงขึ้นภายใต้งบประมาณที่เหมาะสม ไมโครคอนโทรลเลอร์มาจากคำ 2 คำรวมกันคือ “ไมโคร” (Micro) ซึ่งหมายถึง ไมโครโปรเซสเซอร์ (Microprocessor) เป็นอุปกรณ์ประมวลผลขนาดเล็ก ภายในประกอบด้วยหน่วยประมวลผลกลางหรือซีพียู (Central Processing Unit : CPU) หน่วยคำนวณทางคณิตศาสตร์และ (Arithmetic Logic Unit : ALU) วงจรเชื่อมต่อหน่วยความจำและวงจรสัญญาณนาฬิกา อีกคำหนึ่งคือ “คอนโทรลเลอร์” (Controller) หมายถึงอุปกรณ์ควบคุม ดังนั้นไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยสามารถเขียนชุดคำสั่งเพื่อกำหนดรูปแบบการควบคุมได้อิสระ

ไมโครคอนโทรลเลอร์ คืออุปกรณ์อิเล็กทรอนิกส์ที่ทำหน้าที่เสมือนคอมพิวเตอร์ขนาดเล็ก ใช้ควบคุมการทำงานของเครื่องใช้ไฟฟ้าหรือระบบควบคุมทางอิเล็กทรอนิกส์ ให้มีประสิทธิภาพในการทำงานสูงขึ้น สามารถเปลี่ยนแปลงลำดับขั้นตอนการทำงานได้ด้วยการแก้ไขชุดคำสั่งภายในหน่วยความจำ ไมโครคอนโทรลเลอร์มีโครงสร้างหลัก 5 หน่วยส่วนทำงานสัมพันธ์กัน คือ

- 1) หน่วยประมวลผลกลาง (Processing Unit)
- 2) หน่วยสำหรับเก็บข้อมูล (Memory Unit)
- 3) หน่วยสำหรับเชื่อมต่อสัญญาณทางไฟฟ้า (Interface Unit)
- 4) วงจรกำเนิดสัญญาณนาฬิกา (Oscillator Circuit)
- 5) หน่วยการอินเทอร์รัพต์ (Interrupt) สัญญาณ

สรุปได้ว่า ไมโครคอนโทรลเลอร์คือ อุปกรณ์อิเล็กทรอนิกส์ที่รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และ วงจรรับสัญญาณอินพุต วงจรขับสัญญาณเอาต์พุต หน่วยความจำ วงจรกำเนิดสัญญาณไว้ด้วยกัน ซึ่งนำไปใช้ควบคุมการทำงานของเครื่องใช้ไฟฟ้าหรือระบบควบคุมทางอิเล็กทรอนิกส์ สามารถเขียนชุดคำสั่งเพื่อกำหนดรูปแบบการควบคุมได้อิสระ ไมโครคอนโทรลเลอร์ประกอบด้วยโครงสร้างหลัก 5 ส่วน¹ ดังรูปที่ 1.1

¹ สมาร์ทเลิร์นนิ่ง, PIC Microcontroller Learning-By-Doing ด้วยภาษา C, (กรุงเทพฯ : สมาร์ทเลิร์นนิ่ง, 2550), 1.



รูปที่ 1.1 โครงสร้างหลักของไมโครคอนโทรลเลอร์
ที่มา : (สมาร์ทเลิร์นนิ่ง, 2550, 3)

1.1.1 หน่วยประมวลผลกลาง

หน่วยประมวลผลกลาง คือส่วนที่ทำหน้าที่คำนวณทางคณิตศาสตร์หรือการตัดสินใจแบบมีเงื่อนไข ซึ่งมีการทำงานที่ซับซ้อน ลำดับในการทำงานของส่วนประมวลผลขึ้นอยู่กับ การจัดลำดับคำสั่งซึ่งบรรจุอยู่ในของส่วนพื้นที่เก็บข้อมูล

1.1.2 หน่วยเก็บข้อมูล

หน่วยเก็บข้อมูล ทำหน้าที่เก็บข้อมูลแบ่งได้ 2 แบบคือ เก็บข้อมูลแบบชั่วคราว (Random Access Memory : RAM) และแบบถาวรหรืออีอีพรอม (Erasable Program ble Read Only Memory : EPROM) การเก็บข้อมูลแบบชั่วคราวข้อมูลสามารถเปลี่ยนแปลงได้ตลอดเวลาและถูกใช้เก็บค่าตัวแปรในการคำนวณ (Variable) ข้อมูลนี้จะสูญหายเมื่อไม่มีไฟเลี้ยง การเก็บข้อมูลแบบถาวรเป็นการเก็บรหัสต้นฉบับการทำงาน (Code Program) ข้อมูลประเภทนี้สามารถเปลี่ยนแปลงได้ด้วยวิธีการพิเศษ ข้อมูลจะไม่สูญหายถึงแม้ว่าไม่มีไฟไปเลี้ยงตัวไมโครคอนโทรลเลอร์

1.1.3 หน่วยเชื่อมต่อสัญญาณทางไฟฟ้า

หน่วยเชื่อมต่อสัญญาณทางไฟฟ้า ทำหน้าที่ติดต่อสัญญาณระหว่างอุปกรณ์ภายนอกกับไมโครคอนโทรลเลอร์มี 2 แบบคือ อินพุตเอาต์พุตแบบดิจิทัล (Digital I/O) โดยรับและส่งข้อมูลด้วยสัญญาณทางดิจิทัล (Digital Signal) และอินพุตเอาต์พุตแบบแอนะล็อก (Analog I/O) โดยรับและส่งข้อมูลด้วยสัญญาณทางแอนะล็อก (Analog Signal)

1.1.4 วงจรกำเนิดสัญญาณนาฬิกา

วงจรกำเนิดสัญญาณนาฬิกา ทำหน้าที่กำเนิดสัญญาณนาฬิกาด้วยวงจรออสซิลเลเตอร์ (Oscillator Circuit) ประกอบด้วยอุปกรณ์หลักคือ คริสตัล (Crystal) ทำหน้าที่กำหนดช่วงเวลาการประมวลผล (Execute Time) ของหน่วยประมวลผลกลางหรือซีพียู โดยจะมีผลต่อความเร็วในการประมวลผลของไมโครคอนโทรลเลอร์ นอกจากนี้สัญญาณนาฬิกาจะทำหน้าที่เป็นตัวกำหนดความเร็วในการรับส่งข้อมูลดิจิทัลแบบอนุกรม (Digital Series Communication Signal) และกำหนดความถี่ในส่วนของตัวตั้งเวลา (Timer) ภายในตัวไมโครคอนโทรลเลอร์ด้วย

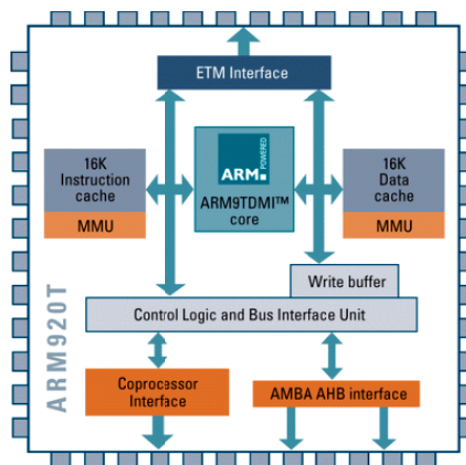
1.1.5 หน่วยบริการอินเตอร์พรีตสัญญาณ

หน่วยบริการอินเตอร์พรีตสัญญาณ ทำหน้าที่จัดลำดับความสำคัญในการทำงาน ในกรณีที่ไมโครคอนโทรลเลอร์ทำงานหลายงานพร้อมกัน (Multitasking) ซึ่งเป็นการอำนวยความสะดวกในการเขียนชุดคำสั่งเพื่อรองรับการทำงานลักษณะนี้

1.1.6 ประเภทของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์แบ่งตามลักษณะการประมวลผลได้ 2 ประเภทคือ ประเภท RISC (Reduced Instruction Set Computer) และประเภท CISC (Complex Instruction Set Computer) ซึ่งแต่ละประเภทจะมีการทำงานและโครงสร้างภายในที่แตกต่างกันดังนี้

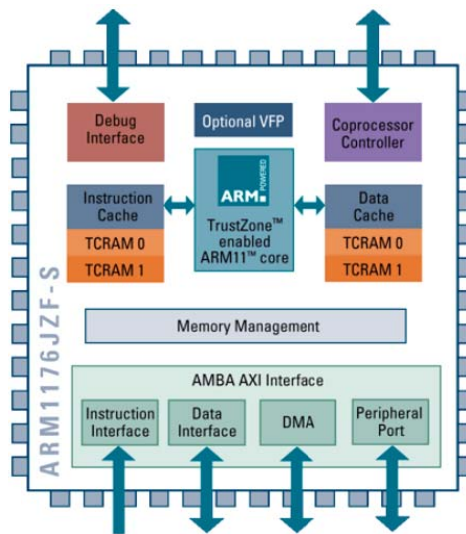
1.1.6.1 ไมโครคอนโทรลเลอร์ประเภท RISC โครงสร้างการทำงานมีชุดคำสั่งน้อย แต่ละคำสั่งทำงานแบบง่าย ๆ มีความเร็วในการทำงานแต่ละคำสั่งสูงมาก แต่มีข้อด้อยคือชุดคำสั่งมีเพียงคำสั่งที่ใช้ทำงานง่าย ๆ เช่น คำสั่งในการคำนวณจะมีแค่เพียงคำสั่ง บวกและลบ ตัวอย่างไมโครคอนโทรลเลอร์แบบ RISC คือ ไมโครคอนโทรลเลอร์อนุกรม PIC16F87X ของบริษัทไมโครชิพ (Microchip) เป็นต้น โครงสร้างการทำงาน แสดงดังรูปที่ 1.2



รูปที่ 1.2 โครงสร้างหลักของไมโครคอนโทรลเลอร์แบบ RISC

ที่มา : (http://www.prism.gatech.edu/~kadams6/iges/risc_i_scr.gif)

1.1.6.2 ไมโครคอนโทรลเลอร์ประเภท CISC โครงสร้างการทำงานมีชุดคำสั่งมากกว่าแบบ RISC แต่ละคำสั่งสามารถทำงานที่ซับซ้อนได้โดยใช้เพียงแค่ 1 คำสั่ง เช่น การเขียนชุดคำสั่งคำสั่งคูณค่า 10 กับค่าตัวแปร ถ้าเราใช้คำสั่งของไมโครคอนโทรลเลอร์แบบ RISC เราจะต้องใช้การวนคำสั่งบวกถึง 10 รอบแต่ ไมโครคอนโทรลเลอร์ประเภท CISC จะทำเสร็จภายในคำสั่งเดียว ตัวอย่างไมโครคอนโทรลเลอร์แบบ CISC คือไมโครคอนโทรลเลอร์ตระกูล 64HC ของบริษัทโมโตโรล่า เป็นต้น แสดงดังรูปที่ 1.3



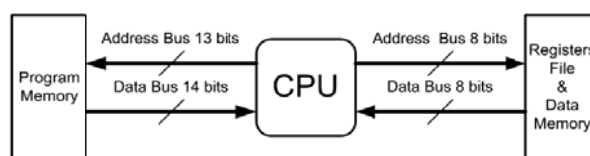
รูปที่ 1.3 โครงสร้างหลักของไมโครคอนโทรลเลอร์แบบ CISC

ที่มา : (<http://dev.emcelettronica.com/3-steps-to-your-first-development-board>.)

เอกสารประกอบการสอนนี้ใช้ไมโครคอนโทรลเลอร์ PIC16F877A ของบริษัทไมโครชิพ เนื่องจากมีความเร็วในการประมวลผลสูง คอมไพเลอร์และชุดคำสั่งมีความเหมาะสมกับพื้นฐาน ของนักศึกษาระดับประกาศนียบัตรวิชาชีพชั้นสูง สาขาวิชาไฟฟ้ากำลัง

1.2 โครงสร้างของไมโครคอนโทรลเลอร์

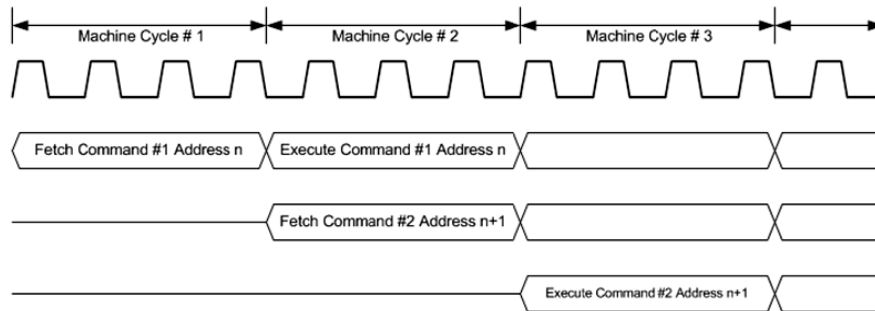
ไมโครคอนโทรลเลอร์ PIC16F877A ได้รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์ และ วงจรรับสัญญาณอินพุตและขับสัญญาณเอาต์พุต วงจรกำเนิดสัญญาณและหน่วยความจำไว้ด้วยกัน ประมวลผลคำสั่งแบบไปป์ไลน์ (Pipe Line) มีสถาปัตยกรรมแบบฮาร์วาร์ด (Harvard Architecture) โดยมีการแยกหน่วยความจำโปรแกรม (Program Memory) กับหน่วยความจำข้อมูล (Data Memory) ออกจากกัน แสดงดังรูปที่ 1.4



รูปที่ 1.4 สถาปัตยกรรมแบบฮาร์วาร์ด

ที่มา : (ฉัฐพล วงศ์สุนทรชัยและชัยวัฒน์ ลิ้มพรจิตรวิไล, 2549, 7)

การประมวลผลคำสั่งแบบไปป์ไลน์ คือ ณะกระทำการคำสั่ง (Execute) ในปัจจุบัน สามารถไปนำ (Fetch) ชุดคำสั่งถัดไปมาแปลเป็นภาษาเครื่องเตรียมไว้ จึงทำให้ใช้เวลาในการประมวลผลชุดคำสั่ง 1 ชุดคำสั่ง ภายในสัญญาณนาฬิกา 1 ลูก แสดงดังรูปที่ 1.5



รูปที่ 1.5 แผนภาพกระบวนการอ่านข้อความตั้งของไมโครคอนโทรลเลอร์ PIC16F877A
ที่มา : (วิบูลย์ ชื่นแขก, 2549, 15)

1.2.1 ชนิดของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ แบ่งตามชนิดของหน่วยความจำโปรแกรม ได้ 3 ชนิดคือ

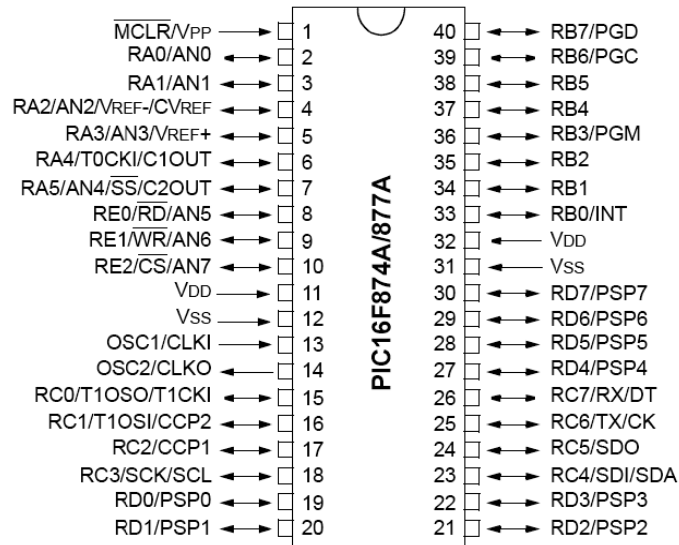
1.2.1.1 หน่วยความจำชนิดบันทึกภาษาเครื่องได้ครั้งเดียว (One Time Programable : OTP) เป็นไมโครคอนโทรลเลอร์ ที่สามารถบันทึกภาษาเครื่อง (Machine Language) ได้ครั้งเดียว ไม่สามารถปรับปรุงแก้ไขภาษาเครื่องใหม่ได้ ใช้ตัวอักษร C เป็นรหัสแสดงที่ตัวไมโครคอนโทรลเลอร์

1.2.1.2 หน่วยความจำชนิดอีพรีอัม (Erasable Programable Read Only Memory : EPROM) ไมโครคอนโทรลเลอร์ชนิดนี้ สามารถลบและบันทึกภาษาเครื่องใหม่ได้ โดยฉายแสงอัลตราไวโอเลต (Ultraviolet) ทางช่องใส ที่อยู่ด้านบนของไมโครคอนโทรลเลอร์ ประมาณ 5-10 นาที แต่มีการจำกัดจำนวนครั้งในการเปลี่ยนแปลงเช่นกัน ใช้ตัวอักษร JW เป็นรหัสแสดงที่ตัวไมโครคอนโทรลเลอร์

1.2.1.3 หน่วยความจำชนิดอีอีพรีอัม (Electrically Erasable Programable ROM : EEPROM) ส่วนของหน่วยความจำโปรแกรม สามารถลบและบันทึกภาษาเครื่องใหม่ได้สูงสุด 100,000 ครั้ง เก็บข้อมูลได้ถึง 40 ปี ใช้ตัวอักษร F เป็นรหัสแสดงที่ตัวไมโครคอนโทรลเลอร์ เช่น PIC16F877A

ไมโครคอนโทรลเลอร์ที่นำมาใช้ศึกษาในเอกสารประกอบการสอนนี้ เป็นตระกูล PIC16F877A มีลักษณะตัวถังแบบ PDIP มีขาจำนวน 40 ขา หน้าที่และตำแหน่งขาแสดงดังรูปที่ 1.6

40-Pin PDIP



รูปที่ 1.6 หน้าที่และตำแหน่งขาของ PIC16F877 ตัวถังแบบ PDIP

ที่มา : (Microchip, 2548, 3)

1.2.2 คุณสมบัติหลักของไมโครคอนโทรลเลอร์ PIC16F877A

1.2.2.1 ชิพเป็นแบบ RISC (Reduced Instruction Set Computer) เป็นชิพที่ใช้เวลาในการประมวลผลคำสั่งเพียง 1 หรือ 2 Machine Cycle ต่อคำสั่งเท่านั้น การประมวลผลคำสั่งเป็นลักษณะไปป์ไลน์ (Pipe Line) คือขณะประมวลผลคำสั่งแรกจะทำการโหลดคำสั่งถัดไปมาเตรียมรอไว้ทำให้การทำงานที่รวดเร็วมากขึ้น และมีคำสั่งใช้งานจำนวน 35 คำสั่ง

1.2.2.2 สามารถกระทำคำสั่งโดยใช้สัญญาณเพียงหนึ่งรอบ (Cycle) ยกเว้นคำสั่งกระโดด

1.2.2.3 ความถี่ของสัญญาณนาฬิกา เพื่อกำหนดจังหวะการทำงานในการประมวลผล มีค่าตั้งแต่ไฟตรงถึงค่าสูงสุดที่ 20 เมกกะเฮิร์ตซ์ (MHz)

1.2.2.4 มีหน่วยความจำโปรแกรม

(Program Memory) เป็นที่เก็บโปรแกรมที่เกิดจากการเขียนโปรแกรมโดยที่สามารถลบและเขียนซ้ำลงไปใหม่ได้ มีขนาด 8 กิโลเวิร์ด หน่วยความจำข้อมูลแรมหรือรีจิสเตอร์ (RAM or Register) เป็นที่เก็บข้อมูลของตัวแปรไว้ หน่วยความจำนี้ต้องมีการจ่ายแรงดันไฟฟ้าให้วงจรตลอดเวลา มีขนาด 368 ไบต์ และหน่วยความจำข้อมูลอีพროม (EEPROM) เพื่อใช้เก็บข้อมูลในตัวแปรต่างๆ ที่สำคัญ เมื่อหยุดจ่ายไฟหน่วยความจำแบบนี้ยังสามารถเก็บข้อมูลไว้ได้ มีขนาด 256 ไบต์

1.2.2.5 สามารถตอบสนองแหล่งกำเนิดอินเตอร์รัพต์สูงสุดได้ 15 แหล่ง ซึ่งเป็นฟังก์ชันเสริมที่คอยรับสัญญาณการจัดจังหวะการทำงานทั้งภายในและภายนอก เมื่อไมโครคอนโทรลเลอร์ประมวลผลคำสั่งต่างๆ ตามโปรแกรมที่กำหนดขึ้น

1.2.2.6 สามารถเลือกกระดัดการป้องกันข้อมูล (Code Protection) ได้ทั้งในหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล

1.2.2.7 มีโหมดประหยัดพลังงาน (Sleep Mode) เป็นฟังก์ชันการทำงานของชิพที่มีการทำงานในระดับกระแสไฟฟ้าที่ต่ำมากทำให้ประหยัดพลังงานของแหล่งจ่ายไฟได้

1.2.2.8 สามารถเลือกแหล่งสัญญาณพิก้า (Oscillator) ได้หลายโหมด เพื่อกำหนดจังหวะการทำงานในการประมวลผล เช่น สัญญาณพิก้าจากคริสตัลพลังงานต่ำ (LP), สัญญาณพิก้าจากคริสตัลแบบเรโซเนเตอร์ (XT), สัญญาณพิก้าจากคริสตัลแบบความเร็วสูง (HS) และ สัญญาณพิก้าจากวงจรอาร์ซี (RC)

1.2.2.9 มีวงจรเพาเวอร์ออนรีเซต (Power-On Reset : POR) เกิดขึ้นหลังจากที่มีการจ่ายแรงดันไฟฟ้าให้กับไมโครคอนโทรลเลอร์ ซึ่งจะควบคุมให้ไมโครคอนโทรลเลอร์อยู่ในสภาวะรีเซตจนกว่าแรงดันไฟฟ้าจะเพิ่มขึ้นถึงจุดที่สูงเพียงพอให้ไมโครคอนโทรลเลอร์พร้อมทำงานได้

1.2.2.10 มีวงจรเพาเวอร์อัปไทมเมอร์ (Power-up Timer : PWRT) เป็นตัวกำหนดให้ไมโครคอนโทรลเลอร์อยู่ในสภาวะรีเซตจนกว่าแรงดันไฟฟ้าจะเพิ่มถึงจุดที่ทำงานได้ และออสซิลเลเตอร์สตาร์ทอัปไทมเมอร์ (Oscillator Start-up Timer : OST) ทำหน้าที่หน่วงเวลาการทำงานของชิพ เพื่อให้ส่วนกำเนิดสัญญาณพิก้าได้เริ่มต้นทำงานและสร้างสัญญาณพิก้าที่มีเสถียรภาพเพียงพอ แล้วออสซิลเลเตอร์สตาร์ทอัปไทมเมอร์จะทำงานก็ต่อเมื่อเลือกโหมดสัญญาณพิก้าเป็นแอลพี (LP), เอ็กซ์ที (XT) หรือ เอชเอส (HS) เท่านั้น

1.2.2.11 มีวงจรวอตช์ด็อกไทมเมอร์ (Watchdog Timer) เป็นวงจรสำหรับการรีเซตเมื่อไมโครคอนโทรลเลอร์อยู่ในสภาวะไม่ทำงานหรือทำงานผิดพลาดบางอย่าง วอตช์ด็อกไทมเมอร์จะสร้างสัญญาณรีเซตภายในส่งมายังชิพ และทำให้เกิดการรีเซตไมโครคอนโทรลเลอร์ขึ้น ซึ่งมีความเที่ยงตรงในการทำงานสูง

1.2.2.12 สามารถทำงานที่ไฟเลี้ยงวงจรตั้งแต่ +2 โวลต์ ถึง +5.5 โวลต์

1.2.2.13 กระแสซิงก์และกระแสซอร์ส (Current Sink and Current Source) แต่ละพอร์ตสามารถจ่ายกระแสได้ไม่เกิน 25 มิลลิแอมป์

1.2.2.14 สามารถอ่านและเขียนหน่วยความจำโปรแกรมได้

1.2.2.15 สามารถโปรแกรมข้อมูลลงหน่วยความจำ โดยใช้แรงดันไฟฟ้าต่ำกว่า +5 โวลต์ (Low Voltage Programming) ได้

1.2.2.16 การใช้พลังงานไฟฟ้าในกรณีไม่ขับภาระทางไฟฟ้า (Load)

1) กระแสไฟฟ้าน้อยกว่า 2 มิลลิแอมป์ ที่แรงดันไฟเลี้ยง +5 โวลต์และสัญญาณพิก้า 4 เมกกะเฮิร์ตซ์

2) กระแสไฟฟ้า 20 ไมโครแอมป์ ที่แรงดันไฟเลี้ยง +3 โวลต์และสัญญาณพิก้า 32 กิโลเฮิร์ตซ์

3) กระแสไฟฟ้าน้อยกว่า 1 ไมโครแอมป์ ในโหมดประหยัดพลังงาน (Stand by)

1.2.2.17 มีพอร์ตสำหรับเชื่อมต่อสัญญาณทางไฟฟ้าแบบอินพุตเอาต์พุต จำนวน 5 พอร์ต ประกอบด้วย

1) พอร์ตเอ (PORTA) จำนวน 6 ขา ประกอบด้วย RA0 - RA5

2) พอร์ตบี (PORTB) จำนวน 8 ขา ประกอบด้วย RB0 - RB7

3) พอร์ตซี (PORTC) จำนวน 8 ขา ประกอบด้วย RC0 - RC7

4) พอร์ตดี (PORTD) จำนวน 8 ขา ประกอบด้วย RD0 - RD7

5) พอร์ตอี (PORTE) จำนวน 3 ขา ประกอบด้วย RE0, RE1 และ RE2

1.2.3 คุณสมบัติพิเศษของไมโครคอนโทรลเลอร์ PIC16F877A

1.2.3.1 มีขาพอร์ตทำหน้าที่ไทมเมอร์/เคาน์เตอร์ (Timer/Counter) ทำหน้าที่เป็นตัวตั้งเวลาและคำนวณจำนวน 3 ชุด ประกอบด้วย ไทมเมอร์ 0 (Timer0) ขนาด 8 บิต มีพรีสเกลเลอร์ (Prescaler) ขนาด 8 บิตในตัว สำหรับไทมเมอร์ 1 (Timer1) มีขนาด 16 บิตมีพรีสเกลเลอร์ในตัว และไทมเมอร์ 2 (Timer2) ขนาด 8 บิตมีโพสสเกลเลอร์ (Postscaler) และรีจิสเตอร์คาบเวลาขนาด 8 บิต 1 ตัว

1.2.3.2 มีโมดูลวงจรแปลงสัญญาณแอนะล็อกเป็นสัญญาณดิจิทัล (Analog/Digital Converter Module) ใช้แปลงค่าสัญญาณแอนะล็อกเป็นสัญญาณดิจิทัล เช่น แรงดันไฟฟ้าจากอุปกรณ์ตรวจจับภายนอกที่มีการเปลี่ยนแปลงสัญญาณเป็น 0 โวลต์ ถึง 5 โวลต์ที่ต่อกับขาอินพุตเพื่อนำค่าแรงดันไฟฟ้าเข้าไปประมวลผลภายใน ซึ่งผลการประมวลผลจะมีขนาด 8 บิตที่สามารถแยกความแตกต่างของสัญญาณทางอินพุตได้ 256 ระดับ และแบบ 10 บิตที่สามารถแยกความแตกต่างของสัญญาณทางอินพุตได้สูงถึง 1,024 ระดับโดยสามารถต่อที่ขาของพอร์ต A และต้องมีการกำหนดค่ารีจิสเตอร์ ADCON ด้วย ประกอบด้วย AN0 (ขาที่ 2) AN1 (ขาที่ 3) AN2/VREF- (ขาที่ 4) สามารถกำหนดด้วยโปรแกรมเป็นขาอ้างอิงโพลได้ AN3/VREF+ (ขาที่ 5) สามารถกำหนดด้วยโปรแกรมเป็นขาอ้างอิงไฟบวกได้ RA4 (ขาที่ 6) และ RA5 (ขาที่ 7)

1.2.3.3 มีโมดูลวงจรตรวจจับสัญญาณ/วงจรเปรียบเทียบสัญญาณ/วงจรกำเนิดสัญญาณทางความกว้างพัลส์ (Compare/Capture/PWM : CCP Module) จำนวน 2 ชุด โดยโมดูลวงจรตรวจจับสัญญาณ (Capture) มีขนาด 16 บิต ความละเอียดสูงสุด 12.5 นาโนวินาที โมดูลวงจรเปรียบเทียบสัญญาณ (Compare) มีขนาด 16 บิต ความละเอียดสูงสุด 200 นาโนวินาที และโมดูลวงจรกำเนิดสัญญาณทางความกว้างพัลส์ (Pulse Width Modulation) มีความละเอียดสูงสุด 10 บิต

1.2.3.4 มีโมดูลวงจรการสื่อสารอนุกรมแบบยูนิเวอแซลซิงโครนัส/อะซิงโครนัส (Universal Synchronous Asynchronous Receiver/Transmitter : USART Module) เป็นโมดูลสำหรับการสื่อสารที่อยู่ภายในตัวไมโครคอนโทรลเลอร์ ซึ่งจะมีโหมดการทำงานอยู่ 2 โหมด คือ โหมดสื่อสารอนุกรมแบบซิงโครนัส ประกอบด้วย โมดูลเชื่อมต่ออุปกรณ์แบบอนุกรม (Serial Peripheral Inter facing : SPI) โมดูลการเชื่อมต่อระบบบัสแบบไอส์แควซี (Inter-IC Communication : I2C) และโหมดสื่อสารอนุกรมแบบอะซิงโครนัส โดยสามารถนำไปใช้ติดต่อกับอุปกรณ์ต่าง ๆ เช่น ติดต่อกับไมโครคอนโทรลเลอร์ด้วยกันหรือกับอุปกรณ์อื่น ๆ ที่มีรูปแบบการสื่อสารเหมือนกัน

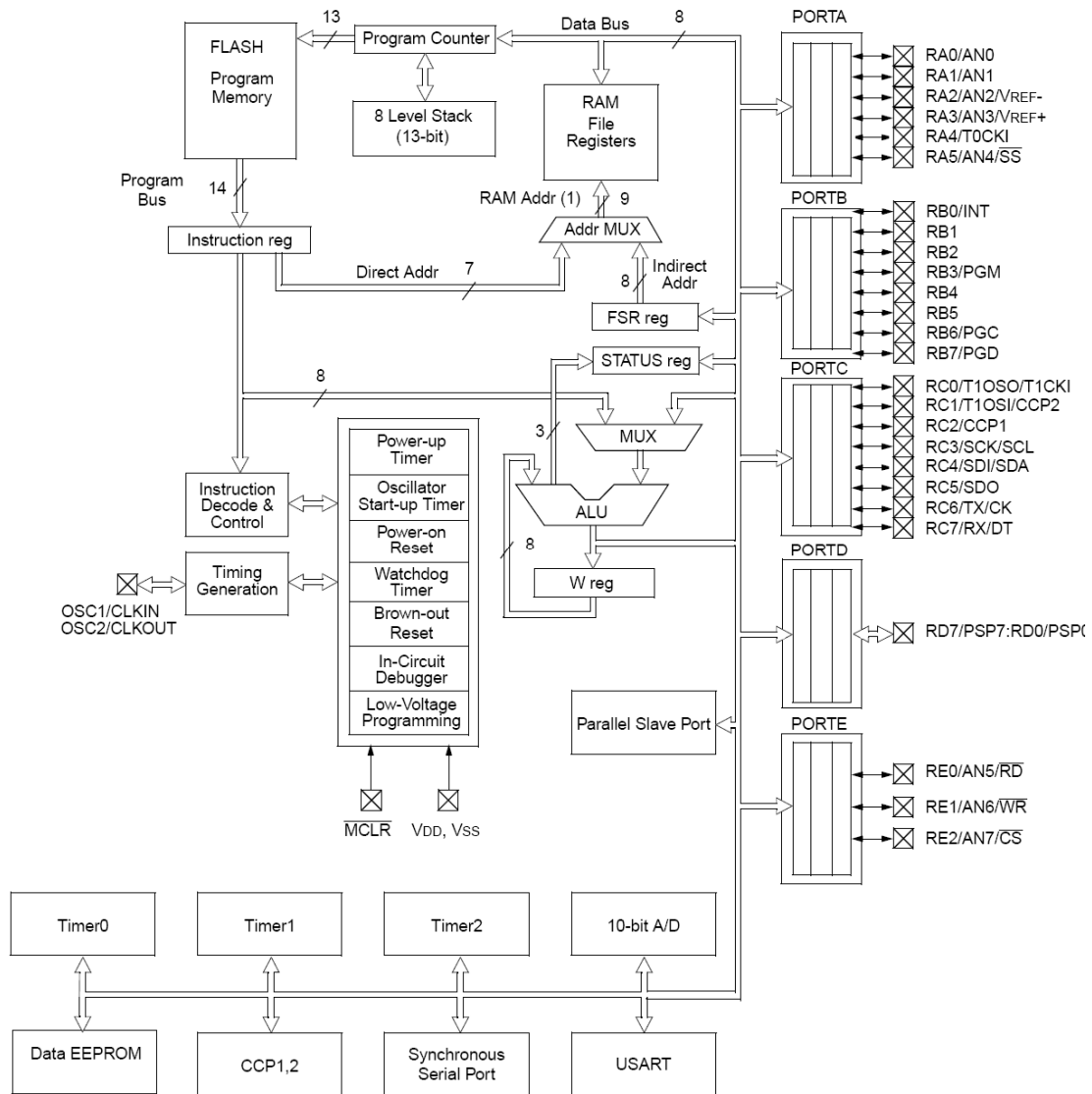
1.2.3.5 มีโมดูลวงจรตรวจจับระดับแรงดันไฟเลี้ยงเรียกว่า บราวเอาต์ดีเทกชัน (Brown Out Detection : BOR Module) เพื่อการรีเซ็ตชิพที่เรียกว่าบราวเอาต์รีเซ็ต (Brown-Out Reset : BOR)²

1.2.4 สถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC16F877A

ไมโครคอนโทรลเลอร์ PIC16F877A จัดอยู่ในประเภท RISC และมีสถาปัตยกรรมแบบฮาร์วาร์ด คือ มีการแยกหน่วยความจำโปรแกรมออกจากหน่วยความจำข้อมูล ดังรูปที่ 1.7 หน่วยความจำข้อมูลเป็นแบบกึ่งถาวรหรือเรียกว่าอีอีพรอม วิธีการแก้ไขข้อมูลในหน่วยความจำโปรแกรมทำได้ด้วยกระบวนการคิบัคเกอร์ (IN-Circuit

² สมบูรณ์ เนียมกล้า, เรียนรู้และประยุกต์ใช้งาน PIC Microcontroller, (กรุงเทพฯ : เอ็ดดิสันเพรส, 2549), 86.

Debugger : IDC) มีกระบวนการประมวลผลคำสั่งแบบไปป์ไลน์จึงทำให้มีความเร็วในการประมวลผลคำสั่งสูง



รูปที่ 1.7 สถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC16F877A
ที่มา : (Microchip, 2548, 7)

ตารางที่ 1.1 ตำแหน่งขาของไมโครคอนโทรลเลอร์ PIC16F877A และรายละเอียดการทำงาน

ชื่อขา	ตำแหน่งขา	ชนิดขา	ชนิดวงจรบัฟเฟอร์	รายละเอียดการทำงาน
OSC1/CLKIN	13	อินพุต	ขมิตต์ทริกเกอร์/ ซิมอส	- ขาต่อคริสตอล/รับสัญญาณนาฬิกาจากภายนอก
OSC2/CLKOUT	14	เอาต์พุต	-	- ขาต่อคริสตอลในโหมดควาซี เป็นขาเอาต์พุต สัญญาณนาฬิกาความถี่ 1/4 ของสัญญาณที่ขา OSC1
MCLR/VPP	1	อินพุต	ขมิตต์ทริกเกอร์	- ขารับสัญญาณรีเซ็ตหลัก (Master Clear) - ขารับแรงดันสำหรับการ โปรแกรม
RA0/AN0	2	อินพุต/ เอาต์พุต	ทีทีแอล/แอนะล็อก	- ขาพอร์ต RA0 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัลช่อง 0
RA1/AN1	3	อินพุต/ เอาต์พุต	ทีทีแอล/แอนะล็อก	- ขาพอร์ต RA1 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัลช่อง 1
RA2/AN2/ VREF-	4	อินพุต/ เอาต์พุต	ทีทีแอล/แอนะล็อก	- ขาพอร์ต RA2 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัลช่อง 2 - อินพุตแรงดันอ้างอิงลบของวงจรแปลงสัญญาณ แอนะล็อกเป็นดิจิทัล
RA3/AN3/ VREF+	5	อินพุต/ เอาต์พุต	ทีทีแอล/แอนะล็อก	- ขาพอร์ต RA3 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัลช่อง 2 - อินพุตแรงดันอ้างอิงบวกของวงจรแปลงสัญญาณ แอนะล็อกเป็นดิจิทัล
RA4/T0CKI	6	อินพุต/ เอาต์พุต	ขมิตต์ทริกเกอร์	- ขาพอร์ต RA4 กรณีใช้พอร์ตเอาต์พุตมีโครงสร้างแบบ เดรนเปิด - อินพุตสัญญาณนาฬิกาของไทเมอร์ 0
RA5/AN4/SS	7	อินพุต/ เอาต์พุต	ทีทีแอล/แอนะล็อก	- ขาพอร์ต RA5 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัลช่อง 4 - ขาสัญญาณ Slave Select ใช้ในการสื่อสารข้อมูล อนุกรมแบบซิงโครนัส
RB0/INT	33	อินพุต/ เอาต์พุต	ทีทีแอล/ ขมิตต์ทริกเกอร์	- ขาพอร์ต RB0 - อินพุตรับสัญญาณอินเตอร์รัพต์จากภายนอก
RB1	34	อินพุต/ เอาต์พุต	ทีทีแอล	- ขาพอร์ต RB1
RB2	35	อินพุต/ เอาต์พุต	ทีทีแอล	- ขาพอร์ต RB2
RB3/LVP	36	อินพุต/ เอาต์พุต	ทีทีแอล	- ขาพอร์ต RB3 - อินพุตรับแรงดัน โปรแกรมแรงดันต่ำ ถ้าได้เปิดทาง ไว้ (Enable)
RB4	37	อินพุต/ เอาต์พุต	ทีทีแอล	- ขาพอร์ต RB4 และสามารถเกิดอินเตอร์รัพต์เนื่องจากการ เปลี่ยนแปลงขึ้นที่ขา นี้ ถ้าได้เปิดทางไว้

ตารางที่ 1.1 (ต่อ)

ชื่อขา	ตำแหน่งขา	ชนิดขา	ชนิดวงจรบัฟเฟอร์	รายละเอียดการทำงาน
RB5	38	อินพุต/ เอาต์พุต	ทีทีแอล	- ขาพอร์ต RB5 และสามารถเกิดอินเตอร์รัพต์เนื่องจากการเปลี่ยนแปลงขึ้นที่ขานี้ ถ้าได้เปิดทางไว้
RB6/PGC	39	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RB6 และสามารถเกิดอินเตอร์รัพต์เนื่องจากการเปลี่ยนแปลงขึ้นที่ขานี้ ถ้าได้เปิดทางไว้
RB7/PGD	40	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RB7 - สามารถเกิดอินเตอร์รัพต์เนื่องจากการเปลี่ยนแปลงขึ้นที่ขานี้ ถ้าได้เปิดทางไว้
RC0/ T1OSO/T1CKI	15	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC0 - เอาต์พุตวงจรออสซิลเลเตอร์ของไทเมอร์ 1 - อินพุตสัญญาณนาฬิกาของไทเมอร์ 1
RC1/ T1OS1/CCP2	16	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC1 - อินพุตวงจรออสซิลเลเตอร์ของไทเมอร์ 1 - อินพุตวงจรแคปเจอร์/เอาต์พุตวงจรเปรียบเทียบ/ เอาต์พุต PWM สำหรับโมดูล CCP2
RC2/ CCP1	17	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC2 - อินพุตวงจรแคปเจอร์/เอาต์พุตวงจรเปรียบเทียบ/ เอาต์พุต PWM สำหรับโมดูล CCP1
RC3/SCK/SCL	18	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC3 - ขาสัญญาณนาฬิกาของวงจร SPI และระบบบัส I ² C
RC4/SDI/SDA	23	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC4 และขาข้อมูลอินพุตวงจร SPI - ขาข้อมูลอนุกรมของระบบบัส I ² C
RC5/SDO	24	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC5 - ขาข้อมูลเอาต์พุตของวงจร SPI
RC6/TxD	25	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC6 - ขาเอาต์พุตวงจร USART สำหรับเชื่อมต่อพอร์ตอนุกรม
RC7/RxD	26	อินพุต/ เอาต์พุต	ชนิดต์ทริกเกอร์	- ขาพอร์ต RC7 - ขาอินพุตวงจร USART สำหรับเชื่อมต่อพอร์ตอนุกรม
RD0/PSP0	19	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RD0 - ขาขยายพอร์ตแบบขนานบิต 0
RD1/PSP1	20	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RD1 - ขาขยายพอร์ตแบบขนานบิต 1
RD2/PSP2	21	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RD2 - ขาขยายพอร์ตแบบขนานบิต 2
RD3/PSP3	22	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RD3 - ขาขยายพอร์ตแบบขนานบิต 3
RD4/PSP4	27	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดต์ทริกเกอร์	- ขาพอร์ต RD4 - ขาขยายพอร์ตแบบขนานบิต 4

ตารางที่ 1.1 (ต่อ)

ชื่อขา	ตำแหน่งขา	ชนิดขา	ชนิดวงจรบัฟเฟอร์	รายละเอียดการทำงาน
RE0/AN5/RD	8	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดคัททริกเกอร์	- ขาพอร์ต RE0 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัล ช่อง 5 - ขาสัญญาณ RD สำหรับส่วนขยายพอร์ตแบบขนาน
RD6/PSP6	29	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดคัททริกเกอร์	- ขาพอร์ต RD6 - ขาขยายพอร์ตแบบขนานบิต 6
RD7/PSP7	30	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดคัททริกเกอร์	- ขาพอร์ต RD7 - ขาขยายพอร์ตแบบขนานบิต 7
RE1/AN6/WR	9	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดคัททริกเกอร์	- ขาพอร์ต RE1 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัล ช่อง 6 - ขาสัญญาณ WR สำหรับส่วนขยายพอร์ตแบบขนาน
RE2/AN7/CS	10	อินพุต/ เอาต์พุต	ทีทีแอล/ ชนิดคัททริกเกอร์	- ขาพอร์ต RE2 - อินพุตวงจรแปลงสัญญาณแอนะล็อกเป็นดิจิทัล ช่อง 7 - ขาสัญญาณ CS สำหรับส่วนขยายพอร์ตแบบขนาน
ขาต่อไฟเลี้ยงระบบ				
VDD	11, 32	อินพุต	-	- ขาต่อไฟเลี้ยงใช้ได้ตั้งแต่ +2 ถึง +5.5 โวลต์
VSS	12, 31	อินพุต	-	- ขาต่อกราวด์

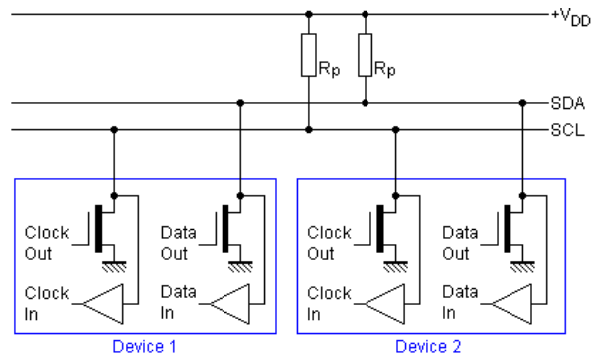
ที่มา : (Microchip, 2548, 8-11)

1.3 ระบบบัสและการทำงาน

เพื่อให้ไมโครคอนโทรลเลอร์ PIC16F877A สามารถรองรับการเชื่อมต่อกับอุปกรณ์อินพุตและอุปกรณ์เอาต์พุตได้เป็นจำนวนมากและมีขีดความสามารถสูง จึงมีการพัฒนาระบบการเชื่อมต่อให้จำนวนสายที่ใช้ลดจำนวนลง ระบบบัสที่ได้รับความนิยมมี 2 ชนิด คือระบบบัสแบบไอสแควซี (Inter-IC Communication : I2C) พัฒนาโดยบริษัทฟิลิปส์ (Philips) และระบบบัสวันไวร์ (I-wire) พัฒนาโดยบริษัทดัลลาสเซมิคอนดักเตอร์ (Dallas Semiconductor)

1.3.1 ระบบบัสแบบไอสแควซี

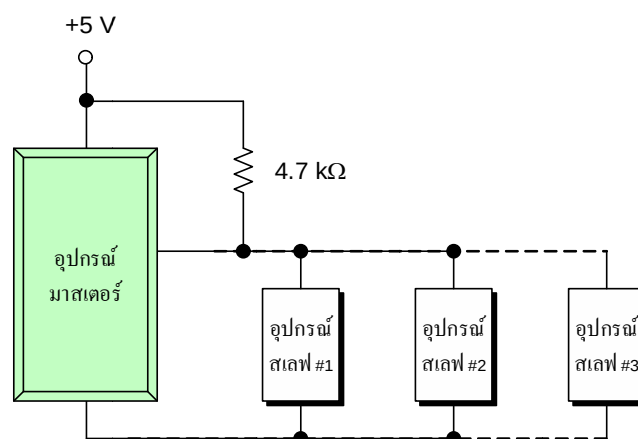
ระบบบัสไอสแควซี หมายถึงการติดต่อระหว่างไอซีกับโมดูลต่าง ๆ ด้วยสายสัญญาณ 2 เส้น ซึ่งประกอบด้วยสายข้อมูลอนุกรมกับสายสัญญาณนาฬิกาอนุกรมกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์ต่าง ๆ บนบัสไอสแควซี ทำได้โดยต่อสายข้อมูลกับสายสัญญาณนาฬิกาขนานกัน วิธีการกำหนดตำแหน่งแอดเดรสสำหรับติดต่อกับอุปกรณ์แต่ละตัว ทำได้โดยใช้รหัสข้อมูลร่วมกันกับการกำหนดสถานะที่ขาแอดเดรส อัตราการถ่ายทอดข้อมูลบนบัสไอสแควซี 100 กิโลบิตต่อวินาทีในโหมดปกติและ 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง ข้อมูลสำหรับการเข้าถึงอุปกรณ์มี 2 ค่า คือ 7 บิต และ 10 บิต รูปแบบการเชื่อมต่อแสดงดังรูปที่ 1.8



รูปที่ 1.8 รูปแบบการเชื่อมต่อมาตรฐานของบัสไอสแควซี
ที่มา : (ประจักษ์ พลังสันติกุล, 2551, 253)

1.3.2 ระบบบัสแบบวันไวร์

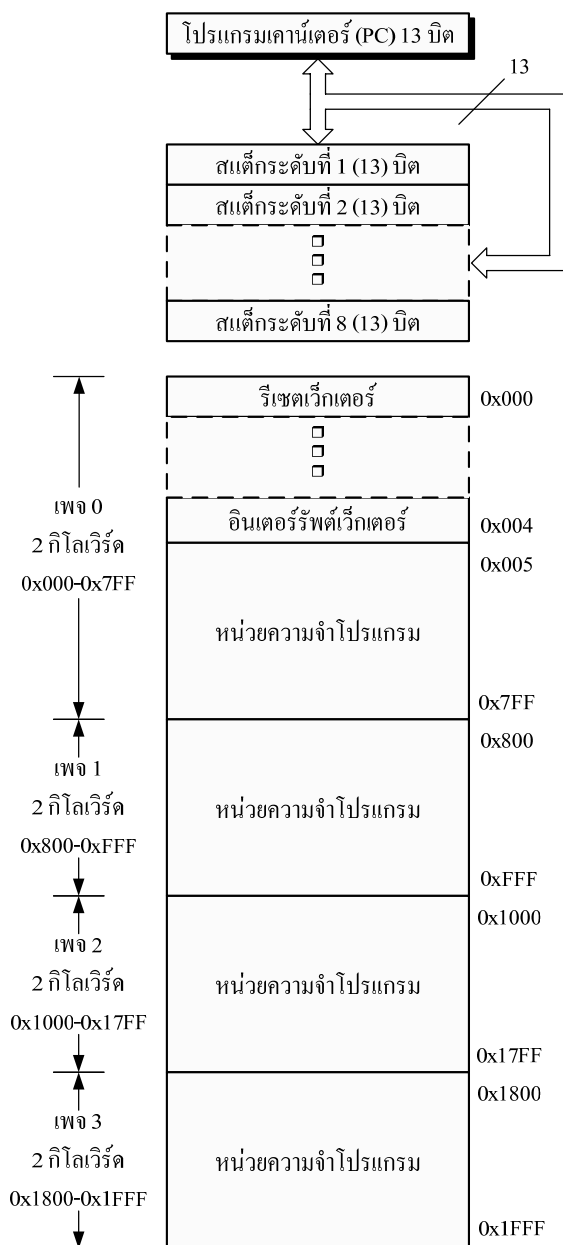
ระบบบัสแบบวันไวร์ สายสัญญาณระบบนี้เป็นแบบสองทิศทาง แต่ข้อมูลส่งได้ทิศทางเดียวภายในช่วงเวลาหนึ่ง ๆ โดยทั่วไปไมโครคอนโทรลเลอร์เป็นอุปกรณ์มาสเตอร์ ส่วนสเลฟได้แก่ ไอซีวัดอุณหภูมิ หน่วยความจำแรม เป็นต้น ระหว่างทำงานข้อมูลถูกส่งลงบนสายสัญญาณ ทั้งมาสเตอร์และสเลฟเป็นได้ทั้งตัวรับและตัวส่งขึ้นอยู่กับเงื่อนไขในขณะนั้น สายสัญญาณจะต้องกำหนดสถานะไว้ที่ลอจิกสูงโดยต่อตัวต้านทานค่า 4.7 กิโลโอห์ม พูลอัพ (Pull up) กับไฟเลี้ยง +5 โวลต์ ดังนั้นอุปกรณ์บนระบบบัสนี้จึงต้องออกแบบให้เอาต์พุตเป็นคอลเล็กเตอร์เปิด (Open Collector) รูปแบบการเชื่อมต่อแสดงดังรูปที่ 1.9 รายละเอียดต่าง ๆ เกี่ยวกับระบบบัสได้กล่าวไว้ในหน่วยที่ 7 เรื่องระบบและรูปแบบการสื่อสาร



รูปที่ 1.9 รูปแบบการเชื่อมต่อของระบบบัสวันไวร์เบื้องต้น
ที่มา : (ประจักษ์ พลังสันติกุล, 2551, 266)

1.4 หน่วยความจำและรีจิสเตอร์

ไมโครคอนโทรลเลอร์ PIC16F877A มีหน่วยความจำ 3 ประเภท คือ หน่วยความจำโปรแกรมแบบแฟลช มีความจุ 8 กิโลเวิร์ด หน่วยความจำข้อมูลแบบแรมมีความจุ 368 ไบต์ ใช้ในการเก็บข้อมูลควบคุมการทำงานของไมโครคอนโทรลเลอร์และเป็นพื้นที่สำหรับเก็บข้อมูลในการประมวลผลและหน่วยความจำข้อมูลแบบอีอีพรีอมนจำนวน 256 ไบต์



รูปที่ 1.10 การจัดสรรหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ PIC16F877A
ที่มา : (Microchip, 2548, 15)

1.4.1 หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรม หมายถึงหน่วยความจำที่ใช้สำหรับเก็บข้อมูลคำสั่งควบคุมการทำงานของไมโครคอนโทรลเลอร์ เป็นหน่วยความจำแบบแฟลช สามารถลบและเขียนใหม่ได้ 100,000 ครั้ง หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ PIC16F877A มีขนาด 8 กิโลเวิร์ด แสดงดังรูปที่ 1.10 ซึ่งมีการจัดสรรพื้นที่ออกเป็น 4 เฟจ ในแต่ละเฟจมีขนาด 2 กิโลเวิร์ด ดังนี้

เฟจ 0 มีแอดเดรสระหว่าง 0x0000 – 0x07FF (สงวน 0x0000 และ 0x0004)

เฟจ 1 มีแอดเดรสระหว่าง 0x0800 – 0x0FFF

เฟจ 2 มีแอดเดรสระหว่าง 0x1000 – 0x17FF

เฟจ 3 มีแอดเดรสระหว่าง 0x1800 – 0x1FFF

นอกจากนั้นไมโครคอนโทรลเลอร์ PIC16F877A ยังมีพื้นที่หน่วยความจำพิเศษ 13 บิต สำหรับเก็บค่าของโปรแกรมเคาท์เตอร์ชั่วคราว หรือพีซี (Program Counter : PC) เป็นตัวชี้ตำแหน่งลำดับการประมวลผล เรียกว่าสแต็ก (Stack) ซึ่งเป็นหน่วยความจำที่ใช้พักค่าตำแหน่งลำดับการประมวลผล กรณีต้องกระโดดไปทำงานโปรแกรมย่อย สแต็กสามารถเก็บค่าพีซีได้ 8 ระดับ และออนชิปโปรแกรมเมมโมรี (On-Chip Program Memory) เป็นพื้นที่สำหรับการประมวลผลคำสั่ง ประกอบด้วยรีเซตเวกเตอร์ (Reset Vector) ซึ่งจัดสรรไว้ที่ 0x000h และอินเทอร์รัพต์เวกเตอร์ (Interrupt Vector) ที่ตำแหน่ง 0x004h ที่เหลือทั้งหมดเป็นตำแหน่งที่เก็บโปรแกรมเริ่มที่ 0x0005h

1.4.2 การจัดสรรหน่วยความจำแรม (เข้าถึงโดยสุ่ม)

หน่วยความจำแรม (Random Access Memory) หมายถึง หน่วยความจำที่ใช้เก็บชุดคำสั่งของรีจิสเตอร์ต่างๆซึ่งประกอบด้วย

1.4.2.1 รีจิสเตอร์ใช้งานทั่วไป (General Purpose Registers : GPR) เป็นหน่วยความจำใช้งานทั่วไป สามารถเปลี่ยนแปลงข้อมูลได้ตลอดเวลา และข้อมูลต่างๆ จะหายไปเมื่อไม่มีไฟเลี้ยงจ่ายให้กับชิพ หน่วยความจำนี้มีขนาดจำนวน 386 ไบต์ ข้อมูลจะเป็นแบบ 8 บิตเหมาะสำหรับใช้เก็บข้อมูล หรือใช้เป็นตัวแปลต่างๆ ในการเขียนโปรแกรมโดยจะกระจายอยู่ในแบงค์ต่างๆทั้ง 4 แบงค์

1.4.2.2 รีจิสเตอร์ที่เป็นฟังก์ชันการทำงานพิเศษ (Special Function Registers : SFR) เป็นหน่วยความจำที่ใช้ในการควบคุมการทำงานในฟังก์ชันต่างๆของชิพ โดยจะจัดวางลักษณะเดียวกันกับรีจิสเตอร์ใช้งานทั่วไป คือจะอยู่ภายในแต่ละแบงค์ทั้ง 4 แบงค์ ซึ่งใช้ควบคุมการทำงานของระบบฮาร์ดแวร์ต่างๆ ภายในตัวชิพ ไม่สามารถนำไปใช้งานอย่างอื่นได้ รีจิสเตอร์แต่ละตัวจะมีที่อยู่คงที่ (Fixed Address) และกำหนดชื่อใช้งานไว้แล้ว แบบใช้งานจริงและแบบสงวนสำรองไว้ใช้งานเฉพาะ (Reserved) แสดงดังรูปที่ 1.11 จะเห็นได้ว่าหน่วยความจำข้อมูลมีการแบ่งพื้นที่หน่วยความจำออกเป็น 4 ช่วง เรียกว่าแบงค์ แต่ละแบงค์มีขนาดสูงสุด 128 ไบต์ แต่มีการใช้งานจริงในแต่ละแบงค์ต่างกัน

1.4.2.1 แบนก์ 0 มีแอดเดรสระหว่าง 0x00 – 0x07F

แอดเดรส 0x00 – 0x1F เป็นพื้นที่รีจิสเตอร์ไฟล์

แอดเดรส 0x70 – 0x7F เป็นพื้นที่ของหน่วยความจำข้อมูล สำหรับใช้งานทั่วไปจำนวน 96

ไบต์

1.4.2.2 แบนก์ 1 มีแอดเดรสระหว่าง 0x08 – 0xFF

แอดเดรส 0x80 – 0x9F เป็นพื้นที่รีจิสเตอร์ไฟล์ มีบางแอดเดรสไม่ใช่

แอดเดรส 0xA0 – 0xEF เป็นพื้นที่ของหน่วยความจำข้อมูล สำหรับใช้งานทั่วไป จำนวน 80

ไบต์

แอดเดรส 0xF0 – 0xFF บรรจุข้อมูลเหมือนกับแอดเดรส 0x70 – 0x7F ในแบนก์ 0 เพื่อให้

สามารถใช้ข้อมูลจากแอดเดรส 0x70 – 0x7F ได้ง่ายขึ้นโดยไม่ต้องเปลี่ยนแบนก์

1.4.2.3 แบนก์ 2 มีแอดเดรสระหว่าง 0x100 – 0x17F

แอดเดรส 0x100 – 0x10F เป็นพื้นที่รีจิสเตอร์ไฟล์ มีบางแอดเดรสไม่ใช่

แอดเดรส 0x110 – 0x11F เป็นพื้นที่ของหน่วยความจำข้อมูล สำหรับใช้งานทั่วไป จำนวน 16

ไบต์

แอดเดรส 0x120 – 0x16F เป็นพื้นที่ของหน่วยความจำข้อมูล สำหรับใช้งานทั่วไป จำนวน 80

ไบต์

แอดเดรส 0x170 – 0x17F บรรจุข้อมูลเหมือนกับแอดเดรส 0x70 – 0x7F ในแบนก์ 0 เพื่อให้

สามารถใช้ข้อมูลจากแอดเดรส 0x170 – 0x17F ได้ง่ายขึ้นโดยไม่ต้องเปลี่ยนแบนก์

1.4.2.4 แบนก์ 3 มีแอดเดรส 0x1800 – 0x1FFF

แอดเดรส 0x180 – 0x18F เป็นพื้นที่รีจิสเตอร์ไฟล์ มีบางแอดเดรสไม่ใช่

แอดเดรส 0x190 – 0x19F เป็นพื้นที่ของหน่วยความจำข้อมูล สำหรับใช้งานทั่วไป จำนวน 16

ไบต์

แอดเดรส 0x1A0 – 0x1EF เป็นพื้นที่ของหน่วยความจำข้อมูล สำหรับใช้งานทั่วไป จำนวน 80

ไบต์

แอดเดรส 0x1F0 – 0x1FF บรรจุข้อมูลเหมือนกับแอดเดรส 0x70 – 0x7F ในแบนก์ 0 เพื่อให้

สามารถใช้ข้อมูลจากแอดเดรส 0x1A0 – 0x1EF ได้ง่ายขึ้นโดยไม่ต้องเปลี่ยนแบนก์

1.4.3 รีจิสเตอร์หลักของไมโครคอนโทรลเลอร์ PIC16F877A

รีจิสเตอร์หลัก หมายถึง รีจิสเตอร์ที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ ประกอบด้วย รีจิสเตอร์ STATUS, OPTION_REG, PCON, W และ PC มีรายละเอียดการทำงานดังนี้

1.4.3.1 รีจิสเตอร์สเตตัส (Register Status) เป็นรีจิสเตอร์ใช้เก็บข้อมูลแสดงสถานะการทำงานของไมโครคอนโทรลเลอร์ เช่น แพลกแสดงผลการกระทำทางคณิตศาสตร์และ การทำงานของวอตช์ด็อกไทมเมอร์ การทำงานโหมดสลีปและใช้ในการเลือกแบนก์ของหน่วยความจำข้อมูลแรม มีแอดเดรสอยู่ที่ตำแหน่ง 0x03 ในแบนก์ 0, 0x83 ในแบนก์ 1, 0x103 ในแบนก์ 2 และ 0x183 ในแบนก์ 3 แสดงดังรูปที่ 1.12

	บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
	R/W - 0	R/W - 0	R/W - 0	R/W - 1	R/W - 1	R/W - x	R/W - x	R/W - x

รูปที่ 1.12 รีจิสเตอร์สเตตัส

ที่มา : (Microchip, 2548, 22)

บิต IRP (Indirect Register Bank Select Bit) ใช้เลือกแบงก์หน่วยความจำข้อมูลแรม และ รีจิสเตอร์ไฟล์ เมื่อใช้การอ้างถึงแบบโดยอ้อม (Indirect Addressing Mode)

“0” - เลือกแบงก์ 0 และ 1 (แอดเดรส 0x00-0xFF)

“1” - เลือกแบงก์ 2 และ 3 (แอดเดรส 0x100-0x1FF)

บิต RP1-RP0 (Register Bank Select Bit) ใช้เลือกแบงก์หน่วยความจำข้อมูลแรม และ รีจิสเตอร์ไฟล์ เมื่อใช้การอ้างถึงแบบโดยตรง

“00” - แบงก์ 0 (แอดเดรส 0x00-0x7F)

“01” - แบงก์ 1 (แอดเดรส 0x80-0xFF)

“10” - แบงก์ 2 (แอดเดรส 0x100-0x17F)

“11” - แบงก์ 3 (แอดเดรส 0x180-0x1FF)

บิต TO (Time-Out Bit) เป็นบิตแสดงการเกิดไทม์เอาต์เมื่อวอตช์ด็อกไทมเมอร์ (WDT) ทำงานครบเวลาที่กำหนด แอกติฟที่ “0” บิตนี้อ่านได้เพียงอย่างเดียว การเซตบิตนี้เกิดขึ้นเมื่อเริ่มต้นจ่ายไฟให้ไมโครคอนโทรลเลอร์หรือเมื่อมีการกระทำคำสั่ง CLRWDT หรือ SLEEP

บิต PD (Power-Down Bit) เป็นบิตแสดงการทำงานในโหมดประหยัดพลังงาน เมื่อกระทำคำสั่ง SLEEP บิตนี้จะเปลี่ยนเป็น “0” บิตนี้สามารถอ่านได้อย่างเดียว การเซตบิตจะเกิดขึ้นเมื่อเมื่อมีการกระทำคำสั่ง CLRWDT หรือเริ่มต้นจ่ายไฟให้แก่ไมโครคอนโทรลเลอร์

บิต Z (Zero Bits): บิตศูนย์ใช้แสดงผลการกระทำคำสั่งทางคณิตศาสตร์

“0” - หลังจากการกระทำคำสั่งทางคณิตศาสตร์หรือแล้วผลลัพธ์ไม่เป็นศูนย์

“1” - หลังจากการกระทำคำสั่งทางคณิตศาสตร์หรือแล้วผลลัพธ์เป็นศูนย์

บิต DC (Digital Carry/Borrow Bit) บิตทดหรือยืมระหว่างหลัก เป็นบิตแสดงผลทางคณิตศาสตร์ในกรณีที่กระทำคำสั่ง ADDWF และ ADDLW ที่บิต DC จะเกิดผลลัพธ์ดังนี้

“0” - หากไม่เกิดการทดข้ามจากบิต 3 ไปยังบิต 4

“1” - หากเกิดการทดข้ามจากบิต 3 ไปยังบิต 4

ในกรณีที่กระทำคำสั่ง SUBWF และ SUBLW ที่บิต DC จะเกิดผลดังนี้

“0” - หากเกิดการยืมค่าจากบิต 4 มายังบิต 3

“1” - หากไม่เกิดการยืมค่าจากบิต 4 มายังบิต 3

บิต C (Carry/Borrow Bit) บิตทดหรือยืม ใช้แสดงผลการทดและการยืมค่าทางคณิตศาสตร์ในกรณีที่กระทำคำสั่ง ADDWF และ ADDLW ที่บิต C จะเกิดผลดังนี้

“0” - หากไม่เกิดการทดจากบิต MSB หรือบิต 7

“1” - หากเกิดการทดจากบิต MSB หรือบิต 7

ในกรณีที่กระทำคำสั่ง SUBWF และ SUBLW ที่บิต DC จะเกิดผลดังนี้

“0” – หากเกิดการยืมค่าของบิต MSB

“1” - หากไม่เกิดการยืมค่าของบิต MSB

1.4.3.2 รีจิสเตอร์ OPTION_REG สามารถอ่านและเขียนได้ทุกบิต บรรจุข้อมูลที่ไว้ควบคุมการพูลอัปพอร์ต B ใช้เลือกขอบสัญญาณอินเทอร์รัพต์ขา RB0/INT ควบคุมการทำงานของไทเมอร์ 0 มีแอดเดรสอยู่ที่ 0x81 แบนก์ 1 และ 0x181 ในแบงก์ 3 แสดงดังรูปที่ 1.13

	บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
OPTION_REG	RBP $\overline{U}$	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
	R/W - 1	R/W - 1	R/W - 1	R/W - 1	R/W - 1	R/W - 1	R/W - 1	R/W - 1

รูปที่ 1.13 รีจิสเตอร์ OPTION_REG

ที่มา : (Microchip, 2548, 23)

บิต 7-RBUP (Port B Pull-Enable Bit) สำหรับเลือกการพูลอัปพอร์ต B

“0” - เอ็นเอเบิลการพูลอัป

“1” - ดิส์เอเบิลการพูลอัป

บิต 6 - INTEDG (Interrupt Edge Select Bit) ใช้เลือกขอบสัญญาณอินเทอร์รัพต์จากภายนอกที่

ขา RB0/INT

“0” – เลือกขอบขาของสัญญาณ

“1” - เลือกขอบขาขึ้นของสัญญาณ

บิต 5 - T0CS เป็นบิตใช้เลือกแหล่งกำเนิดสัญญาณนาฬิกาของไทเมอร์ 0

“0” - รับจากสัญญาณนาฬิกาที่ใช้กำหนดให้غيلการทำงาน

“1” - เลือกขอบขาขึ้นของสัญญาณ

บิต 4 - T0SE (TMR0 Source Edge Select Bit) เป็นบิตใช้เลือกการเปลี่ยนแปลงของระดับสัญญาณเพื่อให้ไทเมอร์ 0 เพิ่มค่าขึ้น

“0” - เลือกการเปลี่ยนแปลงระดับสัญญาณจากต่ำไปสูง

“1” - เลือกการเปลี่ยนแปลงระดับสัญญาณจากสูงไปต่ำ

บิต 3 - PSA (Prescaler Assignment Bit) เป็นบิตสำหรับใช้เลือกการทำงานของ
พรีสเกลเลอร์ (Prescaler)

“0” - เลือกให้สเกลเลอร์ทำงานร่วมกับไทเมอร์ 0

“1” - เลือกให้สเกลเลอร์ทำงานร่วมกับวอตช์ด็อกไทเมอร์หรือเป็นโพสต์สเกลเลอร์
(Postscaler)

บิต 2 - 0: PS2 - PS0 (Prescaler Rate Select Bit) เป็นบิตที่ใช้เลือกอัตราส่วน

ปริสเกลเลอร์ ในการกำหนดอัตราส่วน Prescaler เมื่อทำงานร่วมกับไทมเมอร์ 0 และวอตช์ด็อกไทมเมอร์สามารถกำหนดรายละเอียดได้ดังตารางที่ 1.2

ตารางที่ 1.2 รายละเอียดการเลือกใช้อัตราส่วนของปริสเกลเลอร์

PS-2	PS-1	PS-0	อัตราส่วนของปริสเกลเลอร์	
			เมื่อทำงานร่วมกับ TMR 0	เมื่อทำงานร่วมกับ WDT
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:33	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

ที่มา : (Microchip, 2548, 23)

1.4.3.3 รีจิสเตอร์พีคอน (PCON) เป็นรีจิสเตอร์ที่ใช้แจ้งสถานะเมื่อเกิดการเกิดการรีเซต มีแอดเดรสอยู่ที่ 0x8E มีบิตการใช้งานเพียง 2 บิต บิต 2-7 ไม่ใช้งาน ค่าที่อ่านได้เป็น “0” แสดง

ดังรูปที่ 1.14

	บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
PCON	-	-	-	-	-	-	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
							R/W - 0	R/W - 1

รูปที่ 1.14 รีจิสเตอร์พีคอน (PCON)

ที่มา : (Microchip, 2548, 29)

POR (Power-On Reset Status Bit): บิตแสดงสถานะการเกิดเพาเวอร์ออนรีเซต

“0” - มีเหตุการณ์เพาเวอร์ออนรีเซตเกิดขึ้น

“1” - ไม่มีเหตุการณ์เพาเวอร์ออนรีเซตเกิดขึ้น

BOR (Brown-out Reset Status Bit): บิตแสดงสถานะการเกิดบราวเอาต์รีเซต

“0” - มีเหตุการณ์บราวเอาต์รีเซตเกิดขึ้น

“1” - ไม่มีเหตุการณ์บราวเอาต์รีเซตเกิดขึ้น

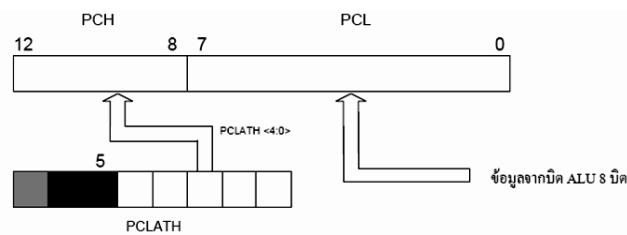
ทั้ง POR และ BOR หลังจากที่มีการบราวเอาต์รีเซตเกิดขึ้น จะต้องเซตบิตนี้ให้เป็น “1” ทันที

ด้วยกระบวนการทางซอฟต์แวร์

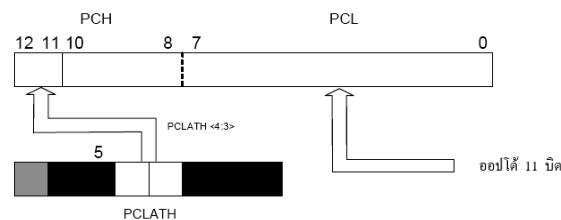
1.4.3.4 รีจิสเตอร์ W เป็นรีจิสเตอร์ที่มีบทบาทมากที่สุดของไมโครคอนโทรลเลอร์ เพราะกระบวนการประมวลผลทางคณิตศาสตร์และ การเพิ่มหรือลดค่าจะต้องกระทำผ่านรีจิสเตอร์ W การเรียกใช้รีจิสเตอร์สามารถกระทำได้ทันทีและตลอดเวลาโดยผ่านคำสั่งที่เกี่ยวข้องกับรีจิสเตอร์ W

1.4.3.5 รีจิสเตอร์โปรแกรมเคาท์เตอร์ (PC) เป็นรีจิสเตอร์พิเศษที่ทำหน้าที่ระบุแอดเดรสของหน่วยความจำโปรแกรมที่ไมโครคอนโทรลเลอร์จะต้องไปอ่านข้อมูล ไมโครคอนโทรลเลอร์ PIC16F877A มีโปรแกรมเคาท์เตอร์ขนาด 13 บิต โปรแกรมเคาท์เตอร์นี้แบ่งเป็น 2 ส่วน ส่วนแรกมีขนาด 8 บิตเป็นข้อมูลในไบต์ต่ำ (บิต 0-7) เรียกว่ารีจิสเตอร์พีซีแอล (PCL) มีแอดเดรสอยู่ที่ 0x02 อีก 5 บิตเป็นข้อมูลไบต์สูง (บิต 8-12) เรียกว่ารีจิสเตอร์พีซีเอช (PCH) รีจิสเตอร์พีซีเอชไม่สามารถเข้าถึงได้โดยตรง การปรับปรุงข้อมูลในพีซีเอชต้องกระทำผ่านรีจิสเตอร์ PCLATH ซึ่งมีแอดเดรสอยู่ที่ 0x0A รูปที่ 1.12 แสดงการกำหนดค่าลงในโปรแกรมเคาท์เตอร์ซึ่งอาจเกิดขึ้น 4 ลักษณะคือ

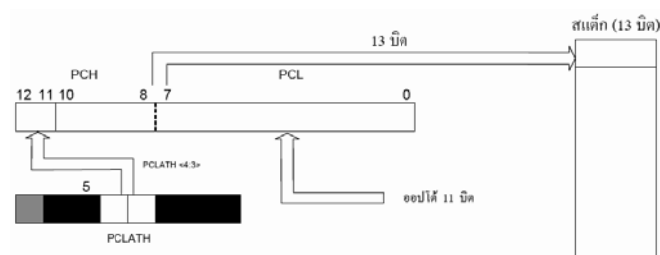
ลักษณะที่หนึ่ง เป็นการกำหนดค่าพีซี หลังจากที่ไม่โครคอนโทรลเลอร์กระทำคำสั่งทั่วไป โดยข้อมูลแอดเดรส 8 บิตใน PCL จะได้จากหน่วยประมวลผลทางคณิตศาสตร์ (ALU) ส่วนอีก 5 บิตจะผ่านมาทางรีจิสเตอร์ PCLATH แสดงดังรูปที่ 1.15 (ก)



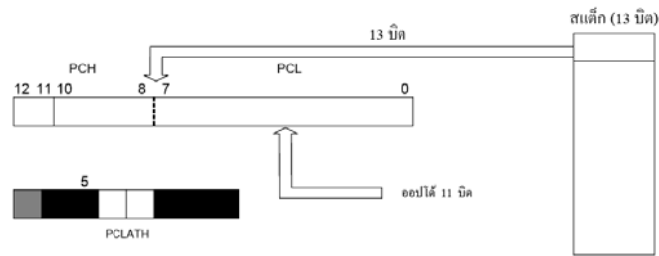
(ก)



(ข)



(ค)



(ง)

- รูปที่ 1.15 (ก) การกำหนดค่าพีซีหลังจากที่ไมโครคอนโทรลเลอร์กระทำข้อความสั่งทั่วไป
 (ข) การกำหนดค่าพีซีเมื่อกระทำคำสั่ง GOTO
 (ค) การกำหนดค่าพีซีเมื่อกระทำข้อความสั่ง CALL
 (ง) การกำหนดค่าพีซีเมื่อกระทำข้อความสั่ง RETURN, RETFIE หรือ RETLW

ที่มา : (Microchip, 2548, 30)

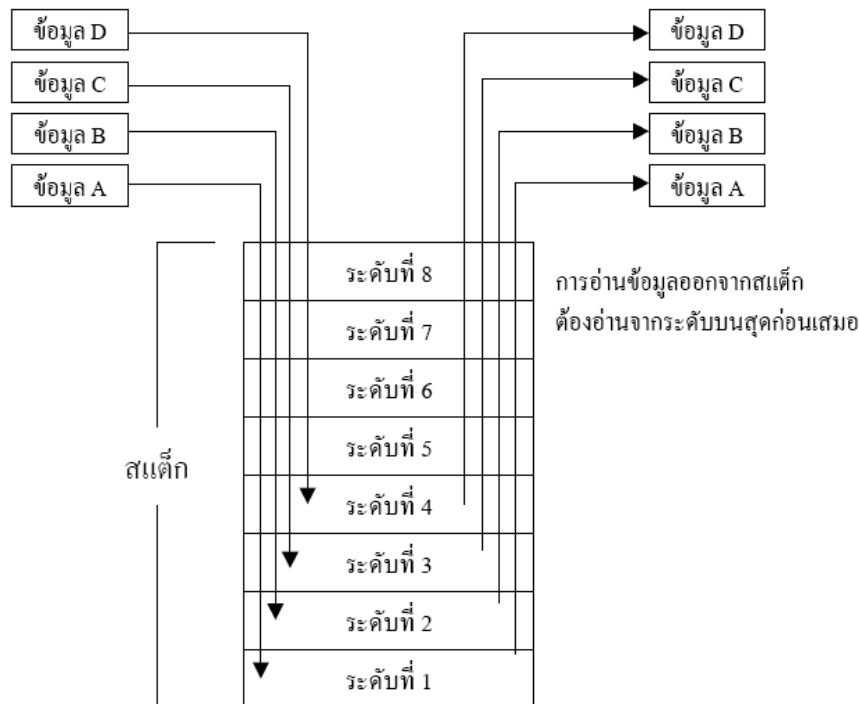
ลักษณะที่สอง เป็นการกำหนดค่าพีซี เมื่อไมโครคอนโทรลเลอร์กระทำคำสั่ง GOTO ชุดคำสั่ง 11 บิตแรก (บิต 0-10) เป็นค่าแอดเดรสปลายทางเก็บไว้ใน PCL 8 บิต รวมกับ 3 บิตล่างของ PCH ได้ข้อมูลมาจากบิต 3 และ 4 ของรีจิสเตอร์ PCLATH ดังในรูปที่ 1.15 (ข)

ลักษณะที่สาม เป็นการกำหนดค่าพีซี เมื่อไมโครคอนโทรลเลอร์กระทำคำสั่งเรียก (CALL) ค่าของพีซีเดิมทั้ง 13 บิตนำค่าไปเก็บไว้ในสแต็กเรียกกระบวนการนี้ว่าพุช (Push) จากนั้นค่าพีซีจะเปลี่ยนแปลงตามค่าของแอดเดรสที่ต้องกระโดดไปทำงาน โดยค่าพีซีใน 2 บิตบน ได้มาจากบิต 3 และ 4 ของ PCLATH ดังในรูปที่ 1.15 (ค)

ลักษณะที่สี่ หลังจาก that ไมโครคอนโทรลเลอร์กระทำคำสั่ง RETURN, RETFIE หรือ RETLW จะทำการโหลดค่าพีซีที่เก็บไว้ในสแต็กกลับมา เรียกกระบวนการนี้ว่า POP การกำหนดค่าพีซีในลักษณะนี้ไม่ต้องใช้ PCLATH ในการกำหนดค่าแต่อย่างใด ดังรูปที่ 1.15 (ง)

1.4.4 สแต็ก (Stack)

ไมโครคอนโทรลเลอร์ PIC16F877A มีหน่วยความจำสำรองสำหรับเก็บค่าโปรแกรมเคาท์เตอร์ชั่วคราวขนาด 13 บิต เรียกว่า สแต็ก ทำหน้าที่เก็บข้อมูลของรีจิสเตอร์ที่ยังต้องการใช้งานและเมื่อข้อมูลนั้นถูกนำมาเก็บไว้ในสแต็กแล้ว สามารถเปลี่ยนข้อมูลในรีจิสเตอร์ตัวนั้นได้ทันที หลังจากทำงานเรียบร้อยแล้วจึงกลับมาอ่านข้อมูลเดิมกลับจากสแต็ก กระบวนการทำงานแสดงรูปที่ 1.16



รูปที่ 1.16 แสดงกลไกการทำงานของสแต็กอย่างง่าย

ที่มา : (ณัฐพล วงศ์สุนทรชัยและชัยวัฒน์ ลิ้มพรจิตรวิไล, 2549, 43)

การเก็บข้อมูลของสแต็กมีลักษณะเป็นระดับหรือเป็นชั้นข้อมูลที่เก็บเข้ามาก่อนจะต้องอ่านออกทีหลังหรือเป็นแบบ FILO (First In Last Out) จำนวนระดับจำกัดไว้ที่ 8 ระดับ โดยพื้นที่ของสแต็กนั้นจะจัดสรรแยกไว้ต่างหากไม่เกี่ยวข้องกับพื้นที่ของหน่วยความจำแต่อย่างใดตัวชี้ตำแหน่งของสแต็กหรือสแต็กพอยเตอร์ ผู้ใช้งานไม่สามารถทำการอ่านหรือเขียนข้อมูลได้โดยค่าของสแต็กพอยเตอร์จะเปลี่ยนแปลงอัตโนมัติ จากการกระทำคำสั่ง CALL, RETURN, RETLW และ RETFIE การเก็บค่าของสแต็กจะต่อเนื่องกันเป็นวงกลม สามารถเก็บข้อมูลได้ 8 ค่า เมื่อมีการเก็บข้อมูลครั้งที่ 9 เข้ามาข้อมูลนั้นจะไปทับในสแต็กของข้อมูลครั้งแรกวนเช่นนี้ไปตลอด ดังนั้นในไมโครคอนโทรลเลอร์ PIC16F877A จึงไม่มีการแจ้งเหตุการณ์สแต็กเกิน (Stack Overflow) นอกจากนั้นการเก็บหรืออ่านค่าในสแต็กของไมโครคอนโทรลเลอร์ PIC16F877A จะเกิดขึ้นเมื่อมีการกระทำคำสั่ง CALL, RETURN, RETLW และ RETFIE หรือเกิดการอินเตอร์รัพต์เท่านั้นไม่มีคำสั่ง PUSH หรือ POP เพื่อติดต่อกับสแต็กเหมือนกับไมโครคอนโทรลเลอร์เบอร์อื่น ๆ

1.4.5 การจัดสรรหน่วยความจำประเภทอีอีพรีม

หน่วยความจำอีอีพรีม เป็นหน่วยความจำที่ใช้เก็บค่าคงที่หรือตารางข้อมูลขนาดใหญ่ ที่ไม่ต้องการการเปลี่ยนแปลงข้อมูลบ่อย ๆ ข้อมูลจะคงอยู่ไม่สูญหายถึงแม้ว่าไม่มีไฟเลี้ยงระบบ วิธีการเขียนข้อมูลไปที่หน่วยความจำนั้น จะต้องกำหนดค่าคอนฟิกูเรชันเวิร์ด (Configuration Word) เพื่อให้สามารถเขียนข้อมูลไปที่หน่วยความจำโปรแกรมได้ โดยการเซตบิต PWRTEW โดยหน่วยความจำประเภทอีอีพรีม ประกอบด้วยฟังก์ชันดังนี้

1.4.5.1 ฟังก์ชันการอ่านและเขียนหน่วยความจำข้อมูลประเภทอีอีพรีม ประกอบด้วย

ฟังก์ชัน write_eeprom () และ Read_eeprom ()

1.4.5.2 ฟังก์ชันการอ่านและเขียนหน่วยความจำโปรแกรมของอีอีพรีอม ประกอบด้วย

ฟังก์ชัน `write_eeprom ()` และฟังก์ชัน `Read_eeprom ()`

1.5 วงจรประกอบการทำงานของไมโครคอนโทรลเลอร์

วงจรประกอบการทำงานของไมโครคอนโทรลเลอร์ หมายถึง การออกแบบระบบการเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ประกอบอื่น ๆ เช่น แหล่งจ่ายไฟ สัญญาณนาฬิกา ระบบการรีเซต เป็นต้น ซึ่งมีความสำคัญต่อการเริ่มต้นการทำงานและประสิทธิภาพในการทำงานของไมโครคอนโทรลเลอร์ การออกแบบที่ดีมีองค์ประกอบหลายอย่าง เช่น ระบบต้องมีขนาดเล็กและนำไปใช้งานได้จริงมีประสิทธิภาพในการทำงานและตอบสนองความต้องการของผู้ใช้ ดังนั้นก่อนการออกแบบผู้ออกแบบจะต้องมองระบบโดยรวมว่าจะให้ความสำคัญกับจุดไหนก่อน แล้วจึงออกแบบตามความต้องการนั้น ๆ ข้อมูลเบื้องต้นที่ผู้ออกแบบวงจรประกอบการทำงานของไมโครคอนโทรลเลอร์จะต้องศึกษาประกอบด้วย

1.5.1 แหล่งจ่ายไฟ (Power Supply)

แหล่งจ่ายไฟ (V_{DD}) สำหรับไมโครคอนโทรลเลอร์ PIC16F877A สามารถทำงานได้ที่ไฟเลี้ยงวงจรตั้งแต่ +2 ถึง +5.5 โวลต์ โดยไฟบวกต่อเข้ากับขา 11, 32 และไฟลบต่อเข้ากับขา 12, 31 ใช้คาปาซิเตอร์ชนิดเซรามิกค่า 0.1 ไมโครฟารัด ต่อกับกราวด์เพื่อลดสัญญาณรบกวน และต้องออกแบบลายวงจรให้อยู่ใกล้ขาแหล่งจ่ายไฟให้มากที่สุด

1.5.2 การออกแบบและเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกา

สัญญาณนาฬิกาเป็นตัวกำหนดจังหวะการทำงานของระบบไมโครคอนโทรลเลอร์ PIC16F877A โดยสามารถเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกา (Oscillator) ได้ 2 แบบคือ

1.5.2.1 แบบภายใน ซึ่งเป็นประเภทวงจรอาร์ซี (RC Oscillator) ความถี่ 4 เมกกะเฮิร์ตซ์ อุณหภูมิ 25°C ออสซิลเลเตอร์ประเภทนี้ความถี่จะเปลี่ยนแปลงตามอุณหภูมิภายนอก

1.5.2.2 แบบภายนอก มีแหล่งกำเนิดสัญญาณนาฬิกา 3 ชนิด ดังรูปที่ 1.14 และโหมดสัญญาณนาฬิกา 4 โหมด โดยกำหนดที่บิต FOSC0 และ FOSC1 ของรีจิสเตอร์การจัดการกับไฟล์เวิร์ด (Configuration word) การใช้งานจริงจะต้องเลือกโหมดใดโหมดหนึ่ง สัญญาณนาฬิกาประกอบด้วย

1) สัญญาณนาฬิกาจากคริสตัลพลังงานต่ำ หรือ แอลพีออสซิลเลเตอร์ (Low Power Crystal : LP) ความถี่ค่า 32 ถึง 200 กิโลเฮิร์ตซ์

2) สัญญาณนาฬิกาจากคริสตัลแบบเรโซเนเตอร์ หรือ เอ็กทีออสซิลเลเตอร์ (Crystal/Resonator : XT) ความถี่ 200 กิโลเฮิร์ตซ์ ถึง 4 เมกกะเฮิร์ตซ์

3) สัญญาณนาฬิกาจากวงจรอาร์ซี หรือ อาร์ซีออสซิลเลเตอร์ (Resistor/Capacitor : RC) ความถี่สูงสุด 4 เมกกะเฮิร์ตซ์ ถึง สัญญาณนาฬิกาจากคริสตัลแบบความเร็วสูงหรือไฮสปีดออสซิลเลเตอร์ (High-Speed : HS) ความถี่ตั้งแต่ 4 เมกกะเฮิร์ตซ์ ถึง 20 เมกกะเฮิร์ตซ์

4) สัญญาณนาฬิกาจากเซรามิกออสซิลเลเตอร์ (Ceramic Resonator) สัญญาณนาฬิกาชนิดนี้ราคาถูกเมื่อเปรียบเทียบกับออสซิลเลเตอร์ประเภทอื่น ความถี่ผิดพลาด (Error) $\pm 5\%$ ไม่ต้องใช้คาปาซิเตอร์ต่อภายนอก ขากลางต่อเข้ากับกราวด์ของระบบ อีกสองขาต่อกับ OSC1 (ขา 13) และ OSC2 (ขา 14) ดังรูปที่ 1.17 (ก)

5) สัญญาณนาฬิกาจากควอตซ์คริสตัลออสซิลเลเตอร์ (Quartz Crystal Oscillator) สัญญาณนาฬิกาชนิดนี้ไม่มีวงจรขยายสัญญาณภายใน มีคริสตัลผลิตความถี่กำลังต่ำ ต้องใช้คาปาซิเตอร์ต่อลงกราวด์ ขาทั้งสองต่อเข้ากับ OSC1 (ขา 13) และ OSC2 (ขา 14) ความถี่มีความเที่ยงตรง แสดงดังรูปที่ 1.17 (ข)

6) สัญญาณนาฬิกาจากทีทีแอลคริสตัลออสซิลเลเตอร์ (TTL Crystal Oscillator) คือสัญญาณนาฬิกาที่มีคริสตัลและวงจรขยายสัญญาณอยู่ภายในตัว ทำให้ความถี่ที่ผลิตมีเสถียรภาพมาก มีราคาแพง การต่อใช้งานต่อตามข้อมูลของบริษัทผู้ผลิต แสดงดังรูปที่ 1.17 (ค)



(ก) เซรามิกออสซิลเลเตอร์



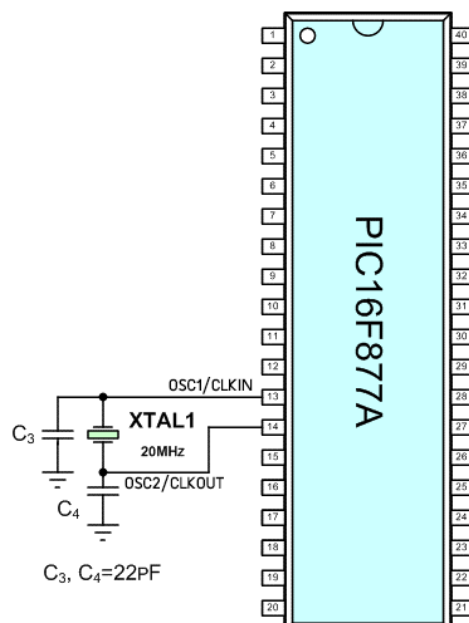
(ข) ควอตซ์คริสตัลออสซิลเลเตอร์



(ค) ทีทีแอลคริสตัลออสซิลเลเตอร์

รูปที่ 1.17 ออสซิลเลเตอร์ชนิดต่างๆ

7) ค่าคาปาซิเตอร์ C_1 และ C_2 เลือกได้จากตารางที่ 1.3



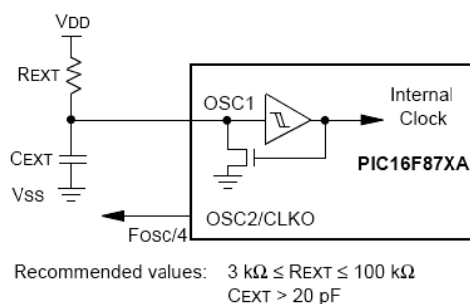
รูปที่ 1.18 การต่อแหล่งกำเนิดสัญญาณนาฬิกาเข้ากับไมโครคอนโทรลเลอร์ PIC16F877A

ตารางที่ 1.3 ค่าคาปาซิเตอร์ C_1 และ C_2 ที่ความถี่ต่าง ๆ

Oscillator Type	Crystal Freq.	Cap. Range C_1	Cap. Range C_2
LP	32 kHz	33 pF	33 pF
	200 kHz	15 pF	15 pF
XT	200 kHz	47 - 68 pF	47 - 68 pF
	1 kHz	15 pF	15 pF
	4 kHz	15 pF	15 pF
HS	4 kHz	15 pF	15 pF
	8 kHz	15 - 33 pF	15 - 33 pF
	20 kHz	15 - 33 pF	15 - 33 pF

ที่มา : (Microchip, 2548, 145)

8) ในกรณีที่ใช้แหล่งกำเนิดสัญญาณพิกากวจอร์ซี (RC) ต้องวงจรดังรูปที่ 1.19



รูปที่ 1.19 การต่อวงจรสัญญาณพิกากวจอร์ซีออสซิลเลเตอร์

ที่มา : (Microchip, 2548, 146)

1.5.3 การออกแบบวงจรไมโครคอนโทรลเลอร์ใช้งานจริงในทางปฏิบัติ

สิ่งที่ต้องพิจารณาแต่ละวงจรการต่อใช้งานจริงแสดง ดังรูปที่ 1.20

1.5.3.1 วงจรคริสตอลออสซิลเลเตอร์ ต้องออกแบบให้คริสตอล คาปาซิเตอร์ C_1 และ C_2 อยู่รวมกัน และใกล้กับขา OSC1 และ OSC2 มากที่สุด ตัวถังคริสตอลที่เป็นโลหะต้องต่อกับกราวด์

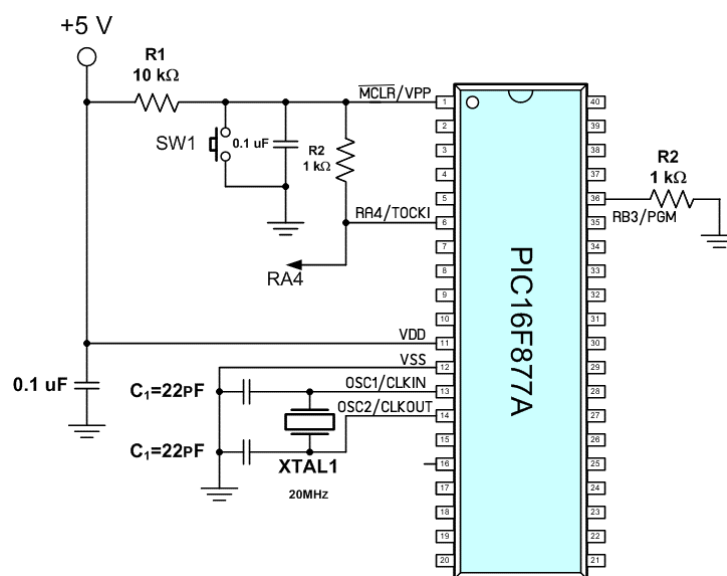
1.5.3.2 วงจรรีเซตที่ใช้ $\overline{MCLR}$ ใช้ตัวต้านทานค่า 4.7 กิโลโอห์ม เซรามิกคาปาซิเตอร์ค่า 0.1 ไมโครฟารัด ที่แรงดัน 16 โวลต์ ทำหน้าที่เป็นตัวหน่วงเวลาด้วยค่าอาร์ซีไทม์คอนสแตนต์ (RC Time Constant) เพื่อเตรียมความพร้อมให้กับอุปกรณ์ต่อพ่วงในการเริ่มต้นทำงาน

1.5.3.3 ขา RA4/T0CKI/C1OUT เป็นขาอินพุตเอาต์พุต (I/O) แบบโอเพนเดรน (Open Drain) ต้องต่อตัวต้านทานค่า 1 กิโลโอห์มอนุกรมกับโหลดและแหล่งจ่ายไฟ เพื่อป้องกันความเสียหาย

1.5.3.4 ขา RB6 และ RB7 เป็นขาที่ทำหน้าที่ 2 อย่าง คือเป็นขาอินพุตเอาต์พุตและใช้เป็นขาโปรแกรม ซีพียู ถ้าเป็นบอร์ดที่มีวงจร โปรแกรมซีพียูในตัว จะต้องมีส่วนสวิตช์สับเปลี่ยนหน้าที่

1.5.3.5 ขา RB3 ทำหน้าที่ 2 อย่างคือเป็นขาอินพุตเอาต์พุตและเป็นขากำหนดการโปรแกรมซีพียูด้วยวิธีการโปรแกรมข้อมูลด้วยแรงดันต่ำ (Low Voltage Programming) ถ้าไม่ใช้งานต้องต่อตัวต้านทานพูลอัพ เพื่อไม่ให้เกิดสถานะดังกล่าวขึ้นระหว่างการใช้งานและการโปรแกรมซีพียู

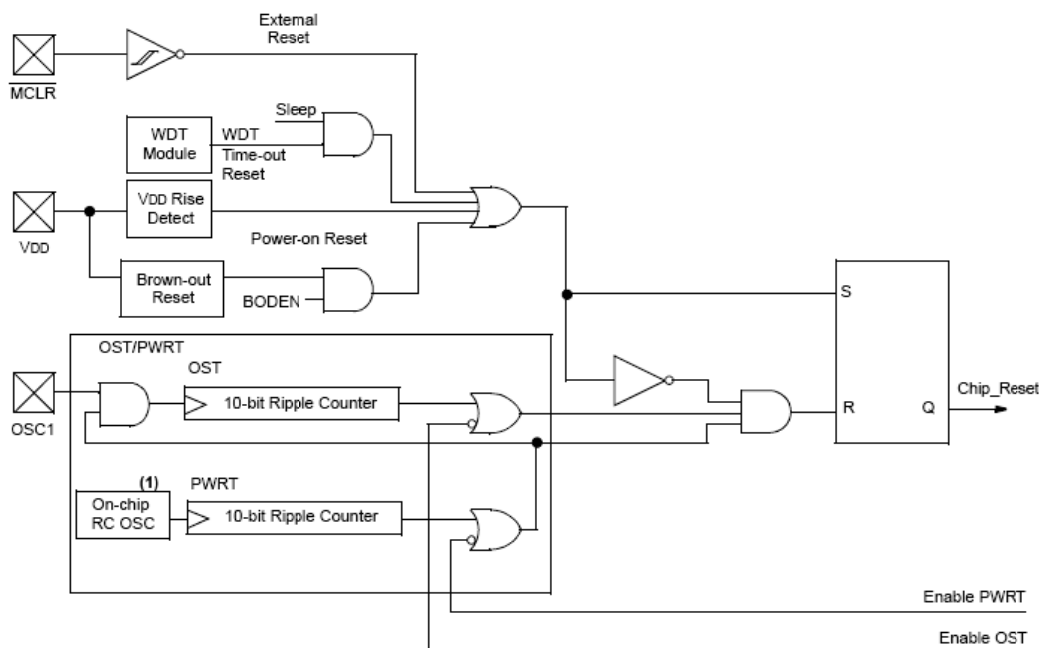
1.5.3.6 การออกแบบการต่อประสานกับอุปกรณ์ภายนอก แต่ละขาสามารถจ่ายกระแสได้ไม่เกิน 25 แอมป์ การออกแบบต้องไม่เกิน 15 มิลลิแอมป์ หรือตามความต้องการของโหลด กระแสเอาต์พุตรวมของพอร์ต A, B และ E สูงสุด 200 มิลลิแอมป์ ของพอร์ต C และ D สูงสุด 200 มิลลิแอมป์



รูปที่ 1.20 การออกแบบวงจรเตรียมความพร้อมการทำงานให้กับไมโครคอนโทรลเลอร์ PIC16F877A

1.5.4 การออกแบบกระบวนรีเซต

รีเซต เป็นกระบวนการกำหนดให้ไมโครคอนโทรลเลอร์เริ่มต้นทำงานใหม่ เพื่อแก้ไขความผิดปกติหรือทำงานผิดพลาด โดยค้างอยู่ในสถานะใดสถานะหนึ่งหรือหยุดทำงาน เมื่อเกิดรีเซตไมโครคอนโทรลเลอร์ก็จะกลับมาเริ่มต้นทำงานใหม่ การรีเซตของไมโครคอนโทรลเลอร์ PIC16F877A มี 6 ประเภทดังนี้



รูปที่ 1.21 กลไกทางของวงจรรีเซ็ตไมโครคอนโทรลเลอร์ PIC16F877A

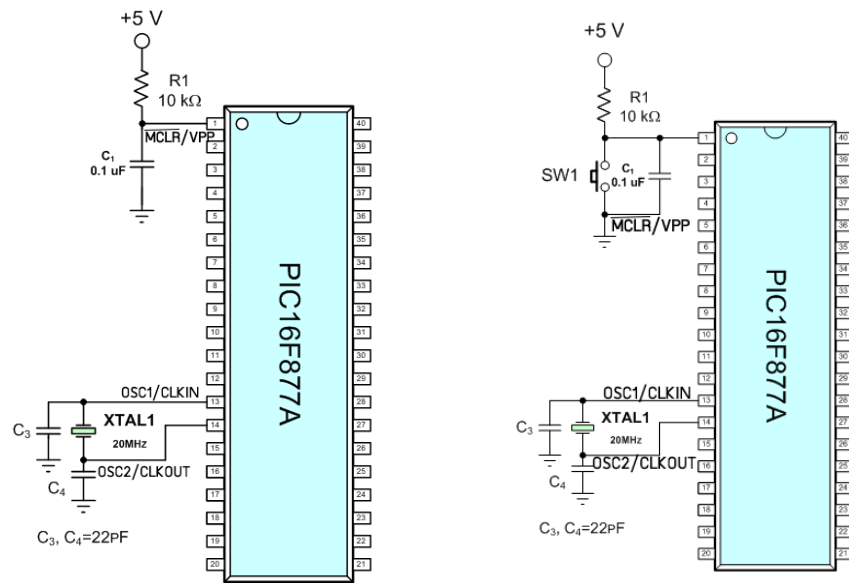
ที่มา : (Microchip, 2548, 147)

1.5.4.1 เพาเวอร์ออนรีเซ็ต (Power-On Reset : POR) เป็นการรีเซ็ตที่มีนัยสำคัญสูงสุด เกิดขึ้นหลังจากมีการจ่ายไฟเลี้ยงให้กับไมโครคอนโทรลเลอร์ วงจรเพาเวอร์ออนรีเซ็ตจะควบคุมให้ไมโครคอนโทรลเลอร์อยู่ในสภาวะรีเซ็ตจนกว่าไฟเลี้ยงจะเพิ่มขึ้นถึงจุดที่สูงเพียงพอ 2 ถึง 5.5 โวลต์หลังจากที่เกิดเพาเวอร์ออนรีเซ็ตจะมีไทมเมอร์ออนรีเซ็ตอีก 2 ตัวทำงาน เพื่อเตรียมความพร้อมให้แก่ตัวไมโครคอนโทรลเลอร์ ได้แก่เพาเวอร์อัปไทมเมอร์ (Power up Timer : PWRT) และออสซิลเลเตอร์สตาร์ทอัปไทมเมอร์ (Oscillator Start Up Timer) มีรายละเอียดดังนี้

1) เพาเวอร์อัปไทมเมอร์ (PWRT) ไทมเมอร์นี้มีคาบเวลา 72 มิลลิวินาที ทำงานหลังจากเกิดเพาเวอร์ออนรีเซ็ตหรือเบรอันเอาต์รีเซ็ต เป็นตัวกำหนดให้ไมโครคอนโทรลเลอร์อยู่ในสภาวะรีเซ็ตจนกว่าไฟเลี้ยงเพิ่มขึ้นถึงจุดที่ทำงานได้

2) ออสซิลเลเตอร์สตาร์ทอัปไทมเมอร์ เป็นไทมเมอร์ที่มีคาบเวลา 1,024 ไซเคิลของคาบเวลาสัญญาณนาฬิกาหลัก ทำงานต่อจากเพาเวอร์อัปไทมเมอร์ด้วยการหน่วงเวลา เพื่อให้แหล่งกำเนิดสัญญาณนาฬิกามีเสถียรภาพ ออสซิลเลเตอร์สตาร์ทอัปไทมเมอร์จะทำงานเมื่อเลือกโหมดสัญญาณนาฬิกาเป็นสัญญาณนาฬิกาจากคริสตัลพลังงานต่ำ (LP), สัญญาณนาฬิกาจากคริสตัลแบบเรโซเนเตอร์ (XT) หรือ สัญญาณนาฬิกาจากคริสตัลแบบความเร็วสูง (HS) ทำงานเฉพาะเมื่อเกิดเพาเวอร์ออนรีเซ็ตหรือเมื่อไมโครคอนโทรลเลอร์ถูกกระตุ้น (Wake-Up) ออกจากโหมดประหยัดพลังงาน

1.5.4.2 การรีเซ็ตที่ขา $\overline{\text{MCLR}}$ ระหว่างการทำงานปกติ (Master Clear Reset) เกิดขึ้นเมื่อขา $\overline{\text{MCLR}}$ ได้รับ “0” สามารถออกแบบวงจรการรีเซ็ตได้ 2 แบบ ดังรูปที่ 1.22 คือ รีเซ็ตแบบอัสติมมิตี ดังรูปที่ 1.22 (ก) และแบบกดสวิตช์รีเซ็ต ดังรูปที่ 1.22 (ข)



(ก) วงจรการรีเซ็ตแบบอัตโนมัติ

(ข) วงจรการรีเซ็ตแบบสวิตช์กดรีเซ็ต

รูปที่ 1.22 การออกแบบวงจรรีเซ็ต

1.5.4.3 การรีเซ็ตที่ขา $\overline{\text{MCLR}}$ เมื่อทำงานอยู่ในโหมดสลีป การรีเซ็ตในโหมดนี้ทำให้ไมโครคอนโทรลเลอร์กลับมาเริ่มต้นทำงานที่แอดเดรส 0x00 ข้อมูลในหน่วยความจำแรมไม่ได้รับผลกระทบ ถ้าไมโครคอนโทรลเลอร์ทำงานในโหมดสลีป เมื่อขา $\overline{\text{MCLR}}$ ได้รับ “0” จะเกิดการรีเซ็ตและยังทำให้เกิดการเวกอัพออกจากโหมดสลีปด้วย

1.5.4.4 การรีเซ็ตเนื่องจากวอตช์ด็อกไทเมอร์ เกิดขึ้นเมื่อเปิดการใช้งาน(Enable)ให้วอตช์ด็อกไทเมอร์ทำงานและซีพียูไม่สามารถเคลียร์ค่าการนับของไทเมอร์ได้ภายในคาบเวลาของวอตช์ด็อกไทเมอร์ เนื่องจากไมโครคอนโทรลเลอร์ทำงานผิดพลาด วนอยู่กับงานบางอย่างจนวอตช์ด็อกไทเมอร์เกิดไทม์เอาต์ วอตช์ด็อกไทเมอร์จะสร้างสัญญาณรีเซ็ตภายในส่งมายังซีพียู ทำให้เกิดการรีเซ็ตไมโครคอนโทรลเลอร์ขึ้น ถ้าในขณะนั้นไมโครคอนโทรลเลอร์อยู่ในโหมดสลีป การที่วอตช์ด็อกไทม์เมอร์เกิดไทม์เอาต์ จะเป็นการเวกอัพหรือกระตุ้นให้ไมโครคอนโทรลเลอร์ออกจากโหมดสลีป กลับมาทำงานในโหมดปกติ

1.5.4.5 วอตช์ด็อกไทเมอร์เวกอัพ (Watchdog Timer Wake-Up) เมื่อไมโครคอนโทรลเลอร์ทำงานในโหมดสลีปและวอตช์ด็อกไทเมอร์เกิดไทม์เอาต์ จะเป็นการเวกอัพหรือกระตุ้นให้ไมโครคอนโทรลเลอร์ออกจากโหมดสลีป กลับมาทำงานในโหมดปกติ

1.5.4.6 เบรอันเอาต์ดีเท็ค (Brown-Out Detect : BOD) เป็นการรีเซ็ตเนื่องจากไฟเลี้ยงลดต่ำกว่าที่กำหนด วงจรเบรอันเอาต์ดีเท็คซึ่งทำงานที่แรงดัน 3.7 โวลต์ ถึง 4.35 โวลต์ ในการทำงานจริง เมื่อแรงดันลดต่ำลงถึง 4 โวลต์ เป็นเวลามากกว่า 100 ไมโครวินาที วงจรเบรอันเอาต์ดีเท็ค จะทำงานเพื่อสร้างสัญญาณรีเซ็ต ภายในส่งต่อให้หน่วยประมวลผลเริ่มต้นทำงานใหม่ การเลือกใช้วงจรเบรอันเอาต์ดีเท็ค ได้โดยเซตบิต BOD ถ้าไม่ต้องการใช้วงจรให้ดิสเอเบิล โดยให้เคลียร์บิต BODEN ในส่วนการจัดการกับไฟส์เวิร์ด (Configuration word)

1.5.5 ส่วนควบคุมอุปกรณ์อินพุตและเอาต์พุต

ไมโครคอนโทรลเลอร์ PIC16F877A มีพอร์ตสำหรับใช้งาน 5 พอร์ต จำนวน 33 บิต ผู้ใช้สามารถกำหนดหน้าที่และการทำงานให้กับขาพอร์ตได้ด้วยกระบวนการทางซอฟต์แวร์ การออกแบบการต่อประสานกับอุปกรณ์ภายนอก แต่ละขาสัญญาณสามารถจ่ายกระแสได้ไม่เกิน 25 มิลลิแอมป์ การออกแบบควรให้จ่ายได้ไม่เกิน 15 มิลลิแอมป์หรือตามความต้องการของโหลด กระแสเอาต์พุตรวมของพอร์ต A รวมกับพอร์ต B และ E ได้สูงสุดไม่เกิน 200 มิลลิแอมป์ สำหรับการจ่ายกระแสของพอร์ต C รวมกับพอร์ต D สูงสุดได้ไม่เกิน 200 มิลลิแอมป์

สรุป

ไมโครคอนโทรลเลอร์ คือ อุปกรณ์อิเล็กทรอนิกส์ที่รวมเอาหน่วยประมวลผลกลาง หน่วยคำนวณทางคณิตศาสตร์และ หน่วยความจำและอุปกรณ์สำหรับติดต่อกับอุปกรณ์ภายนอกไว้ด้วยกัน ใช้ในระบบควบคุมทางอิเล็กทรอนิกส์และการทำงานของเครื่องใช้ไฟฟ้าได้อย่างอิสระ ด้วยการเขียนชุดคำสั่งควบคุม ซิฟิยูแบ่งตามลักษณะการประมวลผลคำสั่งได้ 2 ชนิดคือ แบบ RISC และ แบบ CISC สถาปัตยกรรมของไมโครคอนโทรลเลอร์ PIC16F877A เป็นแบบฮาร์วาร์ด มีการแยกหน่วยความจำโปรแกรมออกจากหน่วยความจำข้อมูล การประมวลผลคำสั่งเป็นแบบไปป์ไลน์ ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมด้วยบัสแอดแดรส 13 บิต ติดต่อกับบัสข้อมูลคำสั่ง 14 บิต ติดต่อกับรีจิสเตอร์และหน่วยความจำข้อมูล 8 บิต ระบบบัสสำหรับติดต่อกับอุปกรณ์ภายนอกมี 2 ระบบคือ ไอส์แควซีและระบบวันไวร์

การออกแบบระบบการต่อประสานระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ประกอบอื่นๆ มีความสำคัญต่อการเริ่มต้น เสถียรภาพและประสิทธิภาพการทำงานซึ่งประกอบด้วย แหล่งจ่ายไฟ ไมโครคอนโทรลเลอร์สามารถทำงานที่ไฟเลี้ยงวงจรตั้งแต่ 2 ถึง 5.5 โวลต์ แหล่งกำเนิดสัญญาณนาฬิกา เลือกใช้ได้ 2 แบบคือ แบบภายในซึ่งเป็นประเภทอาร์ชีออสซิลเลเตอร์ความถี่ไม่เกิน 4 เมกกะเฮิร์ตซ์ แบบภายนอก มีแหล่งกำเนิดสัญญาณนาฬิกาให้เลือก 3 ชนิด ประกอบด้วย เซรามิกออสซิลเลเตอร์, ควอตซ์คริสตอลออสซิลเลเตอร์ และทีทีแอลคริสตอลออสซิลเลเตอร์ แหล่งกำเนิดสัญญาณนาฬิกาดังกล่าวสามารถผลิตความถี่ได้ตั้งแต่ 4 เมกกะเฮิร์ตซ์ ถึง 20 เมกกะเฮิร์ตซ์

การรีเซต เป็นกระบวนการกำหนดให้ไมโครคอนโทรลเลอร์เริ่มต้นทำงานใหม่ เพื่อแก้ไขปัญหาความผิดปกติหรือทำงานผิดพลาดและค้างอยู่ในสถานะใดสถานะหนึ่ง เมื่อเกิดการรีเซตไมโครคอนโทรลเลอร์จะกลับมาเริ่มต้นทำงานใหม่ที่ตำแหน่ง 0x00

เอกสารอ้างอิง

ณัฐพล วงศ์สุนทรชัย และ ชัยวัฒน์ ลิ้มพรจิตรวิไล. เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ PIC16F877. กรุงเทพฯ : อินโนเวติฟ เอ็ดดูเคชัน. 2549

ทีมงานสมาร์ตเลิร์นนิ่ง. PIC Microcontroller Learning-By-Doing ด้วยภาษา C. กรุงเทพฯ : ห้างหุ้นส่วนสามัญสมาร์ตเลิร์นนิ่ง. 2550

ประจัน พลังสันติกุล. การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ dsPIC30F ด้วยคอมไพเลอร์ MPLAB C. กรุงเทพฯ : แอพซอฟต์เทค. 2551

วิบูลย์ ชื่นแขก. ไมโครโปรเซสเซอร์. กรุงเทพฯ : มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ. 2549

สมบูรณ์ เนียมกล้า. เรียนรู้และประยุกต์ใช้งาน PIC Microcontroller. กรุงเทพฯ : เอ็ดดิสันเพรสโปรดักส์. 2549

Microchip. PIC16F87xA Data Sheet. Microchip.

Corp.http://www.prism.gatech.edu/~kadams6/ges/risc_i_scr.gif

<http://dev.emcelettronica.com/3-steps-to-your-first-development-board>

	แบบฝึกหัดที่ 1		หน่วยที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์ รหัสวิชา 3104-2011	สอนครั้งที่ 1-2
	ชื่อหน่วย	โครงสร้างของไมโครคอนโทรลเลอร์	จำนวนชั่วโมงรวม 8 ชั่วโมง
	ชื่อเรื่อง	โครงสร้างของไมโครคอนโทรลเลอร์	จำนวนชั่วโมงสอน 4 ชั่วโมง

คำชี้แจง 1. จงตอบคำถามให้ได้ใจความสมบูรณ์ ใช้เวลาสำหรับการทำแบบฝึกหัด 20 นาที

2. แบบฝึกหัดมีจำนวน 5 ข้อ คะแนนเต็ม 10 คะแนน

1. จงบอกความหมายของไมโครคอนโทรลเลอร์ (1 คะแนน)

2. จงบอกส่วนประกอบและหน้าที่โครงสร้างหลักของไมโครคอนโทรลเลอร์ (3 คะแนน)

3. จงอธิบายโครงสร้างของสถาปัตยกรรมแบบฮาร์ดแวร์ (2 คะแนน)

4. จงอธิบายการทำงานของระบบบัสแบบวนไวก์ (1 คะแนน)

[illegible]

5. จงบอกรายละเอียดการออกแบบวงจรการทำงานของไมโครคอนโทรลเลอร์ PIC16F877A

(3 คะแนน)

[illegible]