แผนการสอนประจำสัปดาห์ที่ 5

หัวข้อเรื่อง สถาปัตยกรรมคอมพิวเตอร์เบื้องต้น
(Introduction to computer architecture)

รายละเอียด

ระบบคอมพิวเตอร์ประกอบด้วยหน่วยประมวลผลกลาง (CPU) หน่วยความจำ (Memory) อุปกรณ์รับข้อมูล (Input Units) และอุปกรณ์แสดงผล (Output Unit) เรียนรู้การเชื่อมโยงของอุปกรณ์ต่าง ๆ และโครงสร้างภายในขององค์ประกอบเหล่านั้น

จำนวนชั่วโมงที่สอน 3 ชั่วโมง/สัปดาห์

กิจกรรมการเรียนการสอน

1. บรรยาย
2. สืบเสาะหาความรู้
3. ค้นคว้าเพิ่มเติม
4. ตอบคำถาม

สื่อการสอน

1. สื่ออิเล็กทรอนิกส์
2. เพาเวอร์พอยต์ พรีเซนเตชัน
3. บทเรียนออนไลน์
4. เอกสารอ้างอิงประกอบการค้นคว้า

แผนการประเมินผลการเรียนรู้

1. ผลการเรียนรู้

1.1 สังเกตจากงานที่กำหนดให้ไปทำมาส่ง

1.2 สังเกตจากการตอบคำถาม

1.3 สังเกตจากการนำความรู้ไปใช้

2. วิธีการประเมินผลการเรียนรู้

2.1 ตรวจผลงานภาคปฏิบัติ

2.2 ตรวจรายงาน

2.3 ตรวจแบบฝึกหัด

3. สัดส่วนของการประเมิน

3.1 ใบงานที่นักศึกษาทำมาส่ง

3.2 คะแนนเก็บในชั้นเรียน

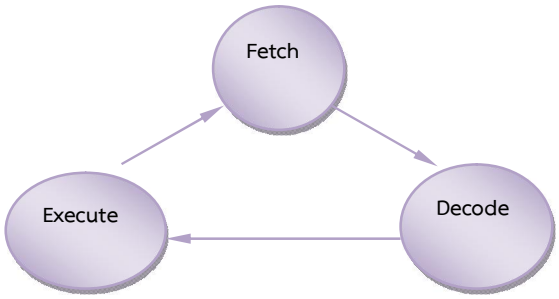
3.3 การเข้าชั้นเรียน

เนื้อหาที่สอน

ในสัปดาห์ที่ 5 การจัดการเรียนการสอน จะเกี่ยวข้องกับหน่วยประมวลผลกลาง หน่วยความจำ การเชื่อมต่อระหว่างอุปกรณ์ต่าง ๆ และของระบบไมโครโปรเซสเซอร์ตระกูล 80x86 ซึ่งระบบคอมพิวเตอร์ประกอบด้วยหน่วยประมวลผลกลาง (CPU) หน่วยความจำ (Memory) อุปกรณ์รับข้อมูล (Input Units) และอุปกรณ์แสดงผล (Output Unit) โดยเริ่มพิจารณาการเชื่อมโยงของอุปกรณ์ต่าง ๆ และโครงสร้างภายในขององค์ประกอบเหล่านั้น

5.1 หน่วยประมวลผลกลาง

หน่วยประมวลผลกลางมีหน้าที่ประมวลผลข้อมูลต่าง ๆ ในระบบคอมพิวเตอร์ โดยหน่วยประมวลผลกลางจะทำงานตามโปรแกรมที่ระบุโดยผู้ใช้ ขั้นตอนการทำงานของหน่วยประมวลผลกลางมีลักษณะเป็นวงรอบ โดยขั้นแรกหน่วยประมวลผลกลางจะอ่านคำสั่งจากหน่วยความจำ (fetch) จากนั้นหน่วยประมวลผลกลางจะตีความคำสั่งนั้น (decode) และในขั้นตอนสุดท้ายหน่วยประมวลผลกลางก็จะประมวลผลตามคำสั่งที่อ่านเข้ามา (execute) เมื่อทำงานเสร็จหน่วยประมวลผลก็จะเริ่มอ่านคำสั่งเข้ามาอีกครั้ง ขั้นตอนดังกล่าวมีลักษณะ แสดงได้ดังภาพที่ 5.1



ภาพที่ 5.1 แสดงขั้นตอนการทำงานของหน่วยประมวลผล

หน่วยประมวลผลกลางจะทำงานตามชุดคำสั่ง (instructions) ที่อ่านมาจากหน่วยความจำหลักเท่านั้น และเรียกสถาปัตยกรรมของระบบคอมพิวเตอร์ที่มีการเก็บโปรแกรมและข้อมูลไว้ในหน่วยความจำหลัก โดยที่หน่วยประมวลผลมีทำหน้าที่ที่เรียกว่า Stored Program Architecture หรือคอมพิวเตอร์แบบวอนนอยแมน (von Neumann Computer) โดยตั้งชื่อเพื่อเป็นเกียรติให้กับ John von Neumann [1]

ชุดคำสั่งของคอมพิวเตอร์ โดยทั่วไปจะประกอบด้วยส่วนย่อย ๆ 2 ส่วนคือ ส่วน Opcode ซึ่งเป็นส่วนที่ระบุประเภทของการประมวลผล และส่วน Operand ซึ่งเป็นส่วนที่ระบุข้อมูลสำหรับการประมวลผลตามที่ระบุใน opcode โดยปกติแล้วมักนิยมใช้ไมโครโปรเซสเซอร์ทำหน้าที่เป็นหน่วยประมวลผลกลางในระบบคอมพิวเตอร์ ดังนั้นเมื่อมีการอ้างถึงไมโครโปรเซสเซอร์จะอ้างถึงหน้าที่ในหน่วยประมวลผลกลาง โดยคำสองคำนี้อาจใช้แทนกันได้

5.2 หน่วยความจำ

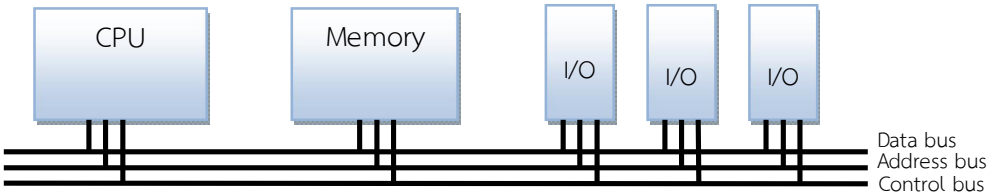
การเก็บข้อมูลในหน่วยความจำมีหน่วยที่เล็กที่สุดในการเก็บข้อมูลคือบิต แต่ในการเรียกข้อมูลจากหน่วยความจำนั้นจะกระทำในรูปของข้อมูลที่มีขนาดใหญ่กว่า คือจะมีขนาด 8 บิต หรือ 1 ไบท์ ภายในหน่วยความจำของระบบคอมพิวเตอร์หนึ่ง ๆ จะประกอบด้วยหน่วยย่อย ๆ ขนาด 8 บิตเหล่านี้อยู่มากมาย แต่ละหน่วยย่อย ๆ จะมีหมายเลขเฉพาะตัว เพื่อให้หน่วยประมวลผลสามารถใช้อ้างถึงเมื่อจะอ่านหรือเขียนข้อมูลลงไปในหน่วยย่อยหน่วยนั้นได้ จะต้องระบุหมายเลขซึ่งเรียกว่า **แอดเดรส (Address)** ของข้อมูลชิ้นนั้นให้ได้ด้วย

---

[1] ผู้บุกเบิกเทคโนโลยีคอมพิวเตอร์เชื่อว่าชื่อของคอมพิวเตอร์แบบวอนนอยแมน ให้เกียรติกับ von Neumann ผู้เขียนแนวคิดเกี่ยวกับคอมพิวเตอร์รูปแบบนี้มากไป โดยให้ความสำคัญกับ J.Presper Eckert และ John Mauchly วิศวกรผู้สร้างเครื่องคอมพิวเตอร์ลักษณะนี้ขึ้นมาน้อยเกินไป

5.3 การเชื่อมต่อระหว่างอุปกรณ์ต่าง ๆ

บัส (Bus) : ช่องทางสื่อสาร อุปกรณ์ต่าง ๆ จะเชื่อมต่อกับโดยผ่านทางกลุ่มของสายสัญญาณ ที่เราเรียกว่า “บัส” อุปกรณ์ต่าง ๆ จะส่งและรับสัญญาณผ่านทางกลุ่มสายสัญญาณชุดเดียวกัน แสดงดังภาพที่ 5.2



ภาพที่ 5.2 แสดงการเชื่อมต่อของอุปกรณ์ต่าง ๆ ผ่านระบบบัส

ระบบบัสสามารถแบ่งการทำงานออกได้ 3 กลุ่ม คือ

1. บัสข้อมูล (Data bus) ใช้สำหรับส่งรับข้อมูล
2. บัสตำแหน่งหรือแอดเดรสบัส (Address bus) ใช้สำหรับส่งรับตำแหน่งที่ใช้สำหรับอ้างอิงข้อมูลจากหน่วยความจำหรือจากอุปกรณ์ I/O
3. บัสควบคุม (Control bus) ใช้สำหรับส่งสัญญาณควบคุม

แอดเดรสบัส (Address bus) ในระบบคอมพิวเตอร์ที่หน่วยประมวลผลเชื่อมต่อกับอุปกรณ์อื่น ๆ ผ่านทางบัส ข้อมูลต่าง ๆ ที่ส่ง/รับกันระหว่างอุปกรณ์ต่าง ๆ นั้นจะส่งผ่านทางบัสข้อมูล ดังนั้นการที่หน่วยประมวลผลจะติดต่อกับหน่วยความจำหรืออุปกรณ์รับและแสดงผลข้อมูลที่ต้องการได้นั้น ทั้งหน่วยความจำ และ อุปกรณ์รับหรือแสดงผลข้อมูลทุกอุปกรณ์ จะต้องมีหมายเลขเฉพาะ หมายเลขนี้สำหรับหน่วยความจำก็คือแอดเดรส ส่วนอุปกรณ์อินพุตและอุปกรณ์เอาท์พุตก็มีหมายเลขเฉพาะสำหรับอุปกรณ์หนึ่ง ๆ เช่นเดียวกัน โดยเรียกว่า หมายเลข I/O (I/O address) เมื่อหน่วยประมวลผลต้องการติดต่อกับหน่วยความจำที่ตำแหน่งใด หรือติดต่อกับอุปกรณ์ใดก็จะส่งแอดเดรสของหน่วยความจำนั้น หรือของอุปกรณ์นั้นมา ในการเลือกว่าหมายเลขแอดเดรสที่ส่งมาเป็นของหน่วยความจำหรือของอุปกรณ์อินพุตเอาท์พุตโดยหน่วยประมวลผลจะส่งสัญญาณระบุมาทางสัญญาณในบัสควบคุม

5.4 สถาปัตยกรรมของระบบไมโครโปรเซสเซอร์ตระกูล 80x86

5.4.1 ความเป็นมา ไมโครโปรเซสเซอร์ตระกูล 80x86 เป็นไมโครโปรเซสเซอร์ที่พัฒนาขึ้นโดยบริษัท Intel โดยมีการพัฒนามาตั้งแต่รุ่น 4040 ซึ่งเป็นไมโครโปรเซสเซอร์ขนาด 4 บิต จนกระทั่งในปัจจุบันได้พัฒนาเป็นไมโครโปรเซสเซอร์รุ่น Pentium รายละเอียดคร่าว ๆ ของไมโครโปรเซสเซอร์รุ่นต่าง ๆ เป็นดังต่อไปนี้

1. 8086 เป็นไมโครโปรเซสเซอร์ขนาด 16 บิต รุ่นแรกที่ Intel ผลิตขึ้น สามารถอ้างหน่วยความจำได้ 1 MByte มีริจิสเตอร์ภายในขนาด 16 บิต

2. 8088 เนื่องจากในขณะนั้นอุปกรณ์ต่าง ๆ โดยมากเป็นอุปกรณ์ขนาด 8 บิต บริษัท Intel จึงได้ผลิตไมโครโปรเซสเซอร์ 8088 ซึ่งมีสถาปัตยกรรมภายในเหมือน 8086 แต่มีการติดต่อกับระบบภายนอกเป็นแบบ 8 บิต

3. 80186 เป็นหน่วยประมวลผลซึ่งเพิ่มการจัดการเกี่ยวกับอุปกรณ์รอบข้างเข้าไปเพื่อให้เป็นหน่วยประมวลผลสำหรับงานควบคุมต่าง ๆ

4. 80286 เป็นหน่วยประมวลผลขนาด 16 บิตที่มีความเร็วในการทำงานสูงขึ้น ขยายขอบเขตการอ้างหน่วยความจำเป็น 24 MByte เพิ่มความสามารถในการจัดการหน่วยความจำ

5. 80386 เป็นหน่วยประมวลผลขนาด 32 บิต อ้างหน่วยความจำได้ถึง 4 กิกะไบต์ มีความสามารถในการจัดการหน่วยความจำที่ซับซ้อน และสามารถใช้หน่วยความจำแบบเสมือนได้ มาตรฐานของชุดคำสั่งและกรรมวิธีในการจัดการหน่วยความจำของหน่วยประมวลผลรุ่นนี้ยังคงใช้เป็นมาตรฐานอยู่จนถึงปัจจุบันนี้ แม้แต่ระบบปฏิบัติการ Windows 95 ยังสามารถนำมาทำงานได้บนหน่วยประมวลผลรุ่นนี้ แต่จะทำงานได้ช้ามากเท่านั้นเอง

6. 80386SX เป็นหน่วยประมวลผลซึ่งมีสถาปัตยกรรมภายในเหมือน 80386 แต่มีระบบบัสภายนอกเป็น 16 บิต หน่วยประมวลผลรุ่นนี้ได้รับการออกแบบขึ้นคล้ายกับ 8088

7. 80486 ได้รับการพัฒนาต่อจากรุ่น 80386 โดยทางบริษัท Intel ได้มีการเพิ่มหน่วยประมวลผลเลขทศนิยมเข้าไป

8. 80486SX เป็นหน่วยประมวลผลรุ่น 80486 ซึ่งตัดความสามารถในส่วนของการประมวลผลเลขทศนิยมออก

9. Pentium, Pentium Pro, Pentium II เป็นหน่วยประมวลผลรุ่นล่าสุดของบริษัท Intel มีการเพิ่มความสามารถในการประมวลผลให้สูงขึ้น โดยใช้เทคโนโลยีต่าง ๆ เช่นการประมวลผลแบบไปป์ไลน์ การประมวลผลแบบซูเปอร์สเกลาร์ เป็นต้น นอกจากไมโครโปรเซสเซอร์ตระกูล 80x86 แล้ว ยังมีหน่วยประมวลผลกลางที่ผลิตและพัฒนาโดยบริษัทต่าง ๆ อีกหลายตระกูลเช่น

ตระกูล 68000 และ PowerPC ของบริษัทโมโตโรล่าตระกูล Alpha ของบริษัท DEC และตระกูล Sparc ของบริษัท Sun Microsystem เป็นต้น

5.4.2 ลักษณะทั่วไปของไมโครโปรเซสเซอร์ 8086

ระบบบัส ไมโครโปรเซสเซอร์ 8086 มีแอดเดรสบัสขนาด 20 บิต ทำให้สามารอ้างแอรดเดรสได้ 1 เมกะไบต์และมีบัสข้อมูลขนาด 16 บิต ซึ่งทำให้การอ่านและเขียนข้อมูลครั้งละ 2 ไบต์ หน่วยประมวลผลทางคณิตศาสตร์และตรรกศาสตร์ภายใน 8086 สามารถประมวลผลได้กับข้อมูลขนาด 16 บิต รีจิสเตอร์ภายในไมโครโปรเซสเซอร์ 8086 มีขนาด 16 บิต

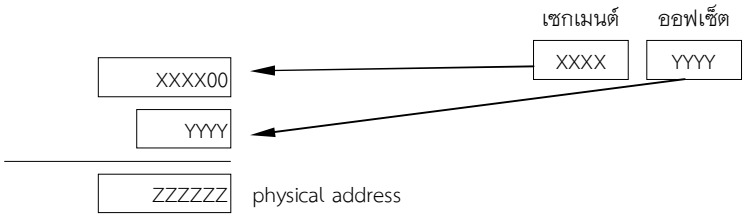
การจัดการหน่วยความจำ ภายในหน่วยประมวลผล 8086 มีรีจิสเตอร์ขนาด 16 บิต แต่มีแอดเดรสบัสขนาด 20 บิต ด้วยสาเหตุดังกล่าวหน่วยประมวลผลจะไม่สามารถเก็บตำแหน่งข้อมูลภายในหน่วยความจำได้ภายในรีจิสเตอร์เพียงตัวเดียว ดังนั้นการจัดเก็บตำแหน่งของข้อมูลภายในหน่วยความจำ 8086 จึงต้องเก็บด้วยรีจิสเตอร์ 2 ตัว ซึ่งมีวิธีการจัดเก็บ แบบ **เซกเมนต์:ออฟเซ็ต (segment:offset)** แอดเดรสที่แท้จริง (physical address) ขนาด 20 บิต จะถูกจัดเก็บด้วยรีจิสเตอร์ขนาด 16 บิต 2 ตัว ค่าที่เก็บในรีจิสเตอร์ตัวแรกเรียกว่าเซกเมนต์ (segment) ส่วนค่าที่เก็บในรีจิสเตอร์อีกตัวเรียกว่าออฟเซ็ต (offset)

การอ้างถึงตำแหน่งภายในหน่วยความจำแบบเซกเมนต์ : ออฟเซ็ตนั้น อาจเปรียบได้เสมือนกับการที่แบ่งหน่วยความจำเป็นส่วนย่อย ๆ โดยส่วนย่อยนี้เรียกว่า เซกเมนต์ การที่มีการอ้างถึงตำแหน่งใด ๆ มักจะอ้างถึงเซกเมนต์ที่ตำแหน่งข้อมูลนั้นอยู่ และระบุระยะห่างของหน่วยความจำที่คิดเทียบกับจุดเริ่มต้นของเซกเมนต์ที่ระบุไป

การแปลงแอดเดรสที่เก็บอยู่ในรูปของ เซกเมนต์ : ออฟเซ็ต เป็นแอดเดรสขนาด 20 บิต มีขั้นตอนดังนี้

1. เลื่อนบิตของค่าเซกเมนต์ไปทางซ้าย 4 บิต ดังนั้นจากข้อมูล 16 บิต เราจะได้ข้อมูล 20 บิต ที่มี 4 บิตทางขวาเป็น 0 ในทุกหลัก

2. นำค่าออฟเซ็ตมาบวกเข้ากับค่าเซกเมนต์ที่เลื่อนบิตแล้ว จะได้แอดเดรสขนาด 20 บิต ที่จะนำไปอ้างตำแหน่งที่แท้จริงของข้อมูล ขั้นตอนทั้งสอง แสดงได้ดังภาพที่ 5.3



ภาพที่ 5.3 แสดงการแปลงแอดเดรสจากแบบเซกเมนต์ : ออฟเซ็ต เป็น แอดเดรสขนาด 20 บิต

ตัวอย่าง เช่น 123Ah : 22B6h แปลงเป็นแอดเดรสขนาด 20 บิตได้ดังนี้

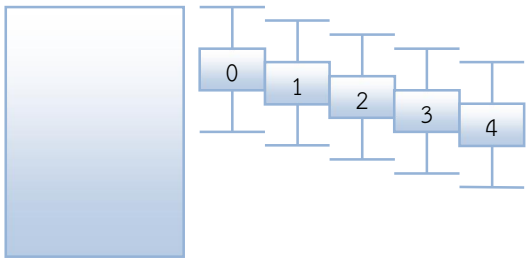
เลื่อนบิตของ 123Ah ไป 4 บิต ได้ 123A0h

นำ 22B6h มาบวก 22B6h

จะได้แอดเดรสขนาด 20 บิตคือ 14656h

ในทางกลับกันที่ตำแหน่ง แอดเดรส 14656h ก็จะสามารถอ้างแบบ เซกเมนต์ : ออฟเซ็ต ได้เป็น 123Ah : 22B6h เช่นเดียวกันแต่ตำแหน่งแอดเดรส 14656h สามารถอ้างแบบ เซกเมนต์ : ออฟเซ็ต ค่าอื่นก็ได้เช่น 1465h : 0006h หรือ 1460h : 0056h และยังอ้างโดยใช้คู่เซกเมนต์ : ออฟเซ็ต คู่อื่น ๆ ได้อีกหลายคู่

ผลจากการใช้การอ้างแอดเดรสแบบ เซกเมนต์ : ออฟเซ็ต ทำให้ลักษณะของหน่วยความจำที่ 8086 มองเห็นจะมีลักษณะเป็นส่วน ๆ ที่อ้างอิงตามค่าของเซกเมนต์ โดยแต่ละส่วนนี้จะมีขนาดส่วนละ 64 กิโลไบท์ ส่วนของหน่วยความจำขนาด 64 กิโลไบต์นี้เรียกว่า เซกเมนต์ การจัดเรียงตัวของเซกเมนต์ต่าง ๆ ในหน่วยความจำจะจัดเรียงเป็นส่วน ๆ ที่มีการเหลื่อมกัน แสดงได้ดังภาพที่ 5.4



ภาพที่ 5.4 แสดงลักษณะการเหลื่อมกันของเซกเมนต์

เนื่องจากการอ้างถึงข้อมูลใด ๆ ในหน่วยความจำของไมโครโปรเซสเซอร์ 8086 จะต้องอ้างตำแหน่งเป็นคู่เซกเมนต์ : ออฟเซต์ การอ้างถึงข้อมูลในตำแหน่งต่าง ๆ อาจทำได้ยุ่งยากเพราะต้องมีการระบุทั้งเซกเมนต์และออฟเซ็ต ในไมโครโปรเซสเซอร์ 8086 จึงได้ออกแบบรีจิสเตอร์พิเศษขึ้น 4 ตัวเพื่อใช้เก็บค่าของเซกเมนต์ต่าง ๆ ที่กำลังใช้งานอยู่ในขณะนั้น กลุ่มของรีจิสเตอร์นั้นเรียกว่า **เซกเมนต์รีจิสเตอร์** ซึ่งได้แก่ CS (Code segment) DS (Data segment) ES (Extra segment) และ SS (Stack segment) เซกเมนต์รีจิสเตอร์ทั้ง 4 ตัวนี้จะใช้ประกอบกับค่าออฟเซ็ตต่าง ๆ เพื่อระบุตำแหน่งของโปรแกรม (code) ข้อมูล (data) และแสต็ก (stack) ส่วนรีจิสเตอร์ ES มีหน้าที่เก็บเซกเมนต์ของข้อมูลที่ใช้ในการสั่งงานคำสั่งพิเศษบางประเภท เช่น คำสั่งเกี่ยวกับข้อความ ดังนั้นจะเห็นได้ว่า ถึงแม้ไมโครโปรเซสเซอร์ 8086 จะสามารถมองหน่วยความจำได้รวมถึง 1 เมกะไบต์ แต่โปรแกรมที่ทำงานอยู่จะมองเห็น

หน่วยความจำได้พร้อม ๆ กันแค่เพียง 4 เซกเมนต์เท่านั้น นั่นคือ เซกเมนต์ของโปรแกรม เซกเมนต์ของข้อมูล 2 เซกเมนต์ และ เซกเมนต์ของแสต็ก

แสต็ก (Stack) ภายในหน่วยความจำของระบบไมโครโปรเซสเซอร์ 8086 จะมีหน่วยความจำส่วนหนึ่งที่ถูกกันเนื้อที่ไว้สำหรับเป็น แสต็กโดยเซกเมนต์ของแสต็กจะถูกชี้โดยรีจิสเตอร์ SS ลักษณะเฉพาะของหน่วยความจำแบบแสต็กคือการที่ระบบจะเก็บข้อมูลและอ่านข้อมูลออกไปแบบ เข้าก่อน ออกทีหลัง (First In Last Out : FILO) โดยอาจมองลักษณะเหมือนการวางซ้อนข้อมูลเหมือนซ้อนจาน ข้อมูลที่ถูกนำมาเก็บก่อนจะอยู่ทางด้านล่าง ข้อมูลถัดไปจะวางซ้อนอยู่ด้านบน ข้อมูลที่อยู่ทางด้านล่างจะไม่สามารถอ่านออกไปได้ถ้ามีข้อมูลอื่นที่เก็บทีหลังและยังไม่ได้อ่านออกไป ระบบจะใช้แสต็กในการเรียกโปรแกรมย่อยเป็นส่วยใหญ่

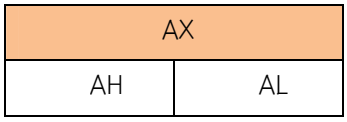
5.4.3 รายละเอียดของส่วนประกอบภายในไมโครโปรเซสเซอร์

หน่วยประมวลผลทางคณิตศาสตร์และตรรกศาสตร์ (ALU) ไมโครโปรเซสเซอร์ 8086 มี ALU ที่สามารถประมวลผลได้ครั้งละ 16 บิต ด้วยความสามารถในการประมวลผลที่ละ 16 บิตนี้ ทำให้เราเรียกไมโครโปรเซสเซอร์ 8086 ว่าเป็นไมโครโปรเซสเซอร์ขนาด 16 บิต

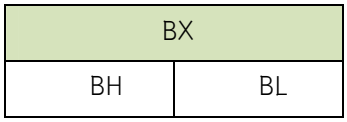
หน่วยความจำชั่วคราว (รีจิสเตอร์ – Register) รีจิสเตอร์ใน 8086 มีทั้งขนาด 16 บิต และ 8 บิต โดยจะแบ่งเป็นกลุ่ม ๆ ได้ดังนี้

1.รีจิสเตอร์สำหรับใช้งานทั่วไป (General–Purpose Registers) รีจิสเตอร์ในกลุ่มนี้ ผู้เขียนโปรแกรมสามารถนำไปใช้งานได้ตามความต้องการ โดยในกลุ่มนี้จะมีรีจิสเตอร์ที่ใช้ขนาด 16 บิต อยู่ 4 ตัว โดยรีจิสเตอร์ขนาด 16 ทั้ง 4 ตัวจะแบ่งได้เป็นรีจิสเตอร์ขนาด 8 บิต อีก 8 ตัว รีจิสเตอร์ขนาด 16 บิตและคู่รีจิสเตอร์ขนาด 8 บิต มีดังต่อไปนี้

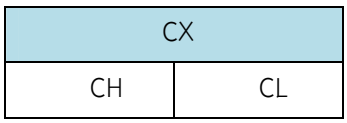
1.1. รีจิสเตอร์ AX (Accumulator Register)



1.2. รีจิสเตอร์ BX (Base Register)



1.3. รีจิสเตอร์ CX (Counter Register)



1.4. รีจิสเตอร์ DX (Data Register)

DX	
DH	DL

สำหรับรีจิสเตอร์ทั้ง 4 ตัวนี้ นอกจากจะมีไว้สำหรับใช้งานทั่วไปได้แล้ว ยังมีหน้าที่เฉพาะอื่น ๆ อีก ซึ่งจะกล่าวถึงในการสอนประจำสัปดาห์ที่ 6 ต่อไป

2. รีจิสเตอร์สำหรับอ้างอิง (Index Register) ไมโครโปรเซสเซอร์ 8086 มีรีจิสเตอร์สำหรับอ้างอิง 2 ตัว คือ SI (Source Index) และ DI (Destination Index) รีจิสเตอร์ในกลุ่มนี้ใช้สำหรับการอ้างตำแหน่งแบบอ้างอิงและใช้ในคำสั่งที่เกี่ยวกับข้อความ แต่ผู้ใช้สามารถนำไปใช้งานทั่วไปได้ด้วยเช่นกัน

รีจิสเตอร์สำหรับการชี้ (Pointer Register) รีจิสเตอร์กลุ่มนี้คือ SP และ BP รีจิสเตอร์ SP ใช้ประกอบกับรีจิสเตอร์ SS มีหน้าที่ชี้ตำแน่งปัจจุบันของแสต็ก โดยรีจิสเตอร์ BP ส่วนใหญ่จะใช้เพื่อชี้ตำแหน่งของแสต็กส่วนของการส่งพารามิเตอร์นิยมใช้ในส่วนโปรแกรมย่อย

3. เซกเมนต์รีจิสเตอร์ (segment register) เซกเมนต์รีจิสเตอร์ทั้ง 4 ตัวคือ CS DS ES และ SS ใช้ประกอบกับค่าของออฟเซ็ตเพื่อชี้ตำแหน่งของโปรแกรมข้อมูลปรกติ ข้อมูลพิเศษและแสต็ก ตามลำดับ

4. แฟล็ก (flag) ไมโครโปรเซสเซอร์ 8086 จะเก็บลักษณะของผลลัพธ์ของการคำนวณทางคณิตศาสตร์ไว้ใน แฟล็ก

5. รีจิสเตอร์อื่น ๆ ของระบบ นอกจากรีจิสเตอร์ต่าง ๆ ที่ผู้ใช้สามารถกำหนดและใช้งานได้แล้ว ยังมีรีจิสเตอร์อีกกลุ่มหนึ่งซึ่งผู้เขียนโปรแกรมไม่สามารถเรียกใช้ได้ รีจิสเตอร์ในกลุ่มนี้ เช่น IP ซึ่งเป็นรีจิสเตอร์ที่ใช้ประกอบกับ CS เพื่อชี้ตำแหน่งของคำสั่งที่จะทำงานต่อไปหรือ IR ซึ่งรีจิสเตอร์ที่เก็บคำสั่งปัจจุบันที่ไมโครโปรเซสเซอร์อ่าน (fetch) ขึ้นมาจากหน่วยความจำ

5.4.4 โหมดการอ้างแอดเดรส ไมโครโปรเซสเซอร์ 8086 สามารถอ้างถึงข้อมูลได้หลายแบบ โดยวิธีการต่าง ๆ ที่ 8086 อ้างถึงข้อมูลนั้น เรารวมเรียกว่า**โหมดการอ้างแอดเดรส (Addressing mode)** ซึ่งรูปแบบที่ 8086 อ้างถึงข้อมูลนั้นแบ่งเป็นกลุ่ม ๆ ได้ 3 กลุ่มใหญ่ ๆ คือ กลุ่มที่อ้างถึงข้อมูลในรีจิสเตอร์ กลุ่มที่อ้างถึงข้อมูลที่ระบุในคำสั่ง และกลุ่มที่อ้างถึงข้อมูลในหน่วยความจำ เป็นต้น

5.4.5 การอินเตอร์รัพท์ (Interrupt) การอินเตอร์รัพท์หรือการขัดจังหวะ คือการสั่งให้หน่วยประมวลผลหยุดการทำงานชั่วคราว แล้วกระโดดไปทำงานบางอย่างเพื่อตอบสนองการ

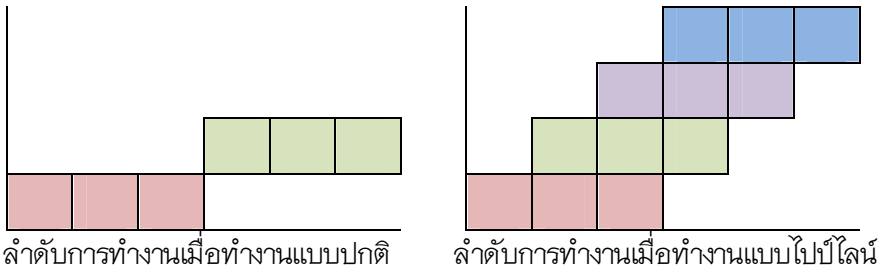
ขัดจังหวะนั้น ตัวอย่างของการขัดจังหวะ เช่น อุปกรณ์บางชิ้นได้รับข้อมูล หรือ ข้อมูลได้รับเขียนเก็บลงในฮาร์ดดิสก์เรียบร้อยแล้ว เป็นต้น เมื่อหน่วยประมวลผลตอบสนองการขัดจะหวะเรียบร้อยแล้ว ก็จะคืนสู่สถานะเดิมและกลับไปประมวลผลงานเก่าที่ประมวลผลค้างไว้ เสมือนไม่มีอะไรเกิดขึ้น การขัดจังหวะนี้มี 2 ประเภทคือ ซอฟต์แวร์อินเตอร์รัพท์ และฮาร์ดแวร์อินเตอร์รัพท์ นิยมใช้ซอฟต์แวร์อินเตอร์รัพท์ในการเรียกใช้การบริการต่าง ๆ ของระบบ ส่วนฮาร์ดแวร์อินเตอร์รัพท์จะนิยมใช้ในการแจ้งการเปลี่ยนสถานะของอุปกรณ์อินพุตเอาท์พุตต่าง ๆ

5.4.6 สถาปัตยกรรมของระบบคอมพิวเตอร์สมัยใหม่ เทคโนโลยีของสถาปัตยกรรมคอมพิวเตอร์สมัยใหม่ได้พัฒนาไปอย่างรวดเร็วมาก เครื่องคอมพิวเตอร์ที่ใช้ในปัจจุบันมีประสิทธิภาพมากกว่าเครื่องคอมพิวเตอร์เมื่อ 2–3 ปีก่อนหลายเท่า ทั้งนี้เนื่องจากมีการวิจัยและสร้างหน่วยประมวลผลกลางและระบบคอมพิวเตอร์ที่มีประสิทธิภาพสูงขึ้นมาก

เทคโนโลยีของหน่วยประมวลผลกลาง

1. หน่วยประมวลผลแบบ RISC ชุดคำสั่งของหน่วยประมวลผลยุคเก่ามีลักษณะเป็นแบบ CISC : Complex Instruction Set Computer นั่นคือชุดคำสั่งจะหนึ่ง ๆ จะมีความซับซ้อนมาก การที่ชุดคำสั่งซับซ้อนทำให้การออกแบบส่วนควบคุมภายในหน่วยประมวลผลทำได้ยาก ในปัจจุบันหน่วยประมวลผลต่าง ๆ ได้เปลี่ยนแนวทางในการพัฒนาไปเป็นแบบ RISC : Reduced Instruction Set Computer โดยเน้นชุดคำสั่งที่มีความซับซ้อนน้อยลง แต่มีความเร็วในการทำงานสูงขึ้น การทำให้ชุดคำสั่งมีรูปแบบที่ง่ายขึ้นทำให้การออกแบบส่วนควบคุมทำได้ง่ายขึ้นและยังสามารถใช้วิธีการแบบไปป์ไลน์ (Pipeline) และซูปเปอร์สเกลาร์ (Superscalar) ในการเพิ่มประสิทธิภาพของหน่วยประมวลผลได้ง่ายขึ้นด้วย

2. ไปป์ไลน์ (Pipeline) หน่วยประมวลผลรุ่นใหม่จะมีการประมวลผลแบบไปป์ไลน์ ซึ่งจะมีการ fetch decode และ execute คำสั่งที่เหลื่อมเวลากัน แสดงได้ดังภาพที่ 5.5 การประมวลผลเหลื่อมกันนี้ทำให้ประสิทธิภาพของการประมวลผลสูงขึ้นมาก

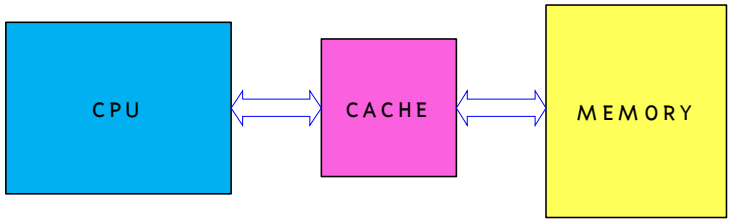


ภาพที่ 5.5 แสดงการทำงานแบบไปป์ไลน์เทียบกับการทำงานแบบปกติ

3. ซูเปอร์สเกลาร์ (Superscalar) ในหน่วยประมวลผลที่มีประสิทธิภาพสูงบางรุ่นจะประมวลผลชุดคำสั่งหลายชุดคำสั่งได้พร้อมกัน การที่หน่วยประมวลผลประมวลผลคำสั่งได้หลายชุดพร้อมกันนี้เรียกว่า ซูเปอร์สเกลาร์

ระบบบัสสมัยใหม่ ระบบคอมพิวเตอร์สมัยก่อน หน่วยประมวลผลมีความเร็วในการประมวลผลไม่มากนักทำให้การโอนย้ายข้อมูลระหว่างหน่วยประมวลผลกับหน่วยความจำกระทำได้โดยไม่ก่อให้เกิดการเสียเวลา แต่การพัฒนาของหน่วยประมวลผลเป็นไปอย่างรวดเร็วกว่าการพัฒนาของหน่วยความจำมากทำให้ปัจจุบันอัตราการประมวลผลของหน่วยประมวลผลสูงกว่าอัตราการโอนย้ายข้อมูลระหว่าหน่วยประมวลผลกับหน่วยความจำมาก ซึ่งสถานการณ์เช่นนี้ก่อให้เกิดปัญหาในรูปแบบคอขวด (Bottleneck problem) ขึ้นนั่นคือจุดเชื่อมต่อระหว่างหน่วยความจำกับหน่วยประมวลผลทำให้ประสิทธิภาพของระบบลดลง ซึ่งเรียกคอขวดระหว่างหน่วยประมวลผลกับหน่วยความจำว่า คอขวดของวอนนอยแมน (von Neumann Bottleneck)

วิธีการแก้ปัญหานี้คือ การใช้หน่วยความจำที่มีความเร็วสูงและขนาดเล็กมาเป็นบัฟเฟอร์ (ที่พักข้อมูลชั่วคราว) ระหว่างหน่วยความจำและหน่วยประมวลผล หน่วยความจำที่มีความเร็วสูงนี้เรียกว่า **หน่วยความจำแคช (cache memory)** ในหน่วยประมวลผลปัจจุบันหลายรุ่น ได้มีการบรรจุหน่วยความจำแคชลงไปภายในไมโครโปรเซสเซอร์ด้วย ลักษณะของบัสที่มีการใช้หน่วยความจำแคช แสดงได้ดังภาพที่ 5.6



ภาพที่ 5.6 แสดงลักษณะของบัสที่มีการใช้หน่วยความจำแคช

สรุป

สถาปัตยกรรมไมโคร (Micro Architecture) เป็นลักษณะของ เครื่องคอมพิวเตอร์ส่วนใหญ่ที่มีการใช้งานมาตั้งแต่ยุคเริ่มต้น จนถึงยุคปัจจุบัน ได้รับการออกแบบโครงสร้างและการทำงานโดยจอนวอนนิวแมน(John Von Neumann) ซึ่งเป็นผู้นำในการออกแบบเครื่องคอมพิวเตอร์ โดยเครื่องคอมพิวเตอร์ ที่เขาได้ออกแบบ มีส่วนประกอบที่สำคัญ 3 อย่างด้วยกัน คือ หน่วยประมวลผลกลาง (Central Processing Unit) หน่วยความจำ (Main Memory) และหน่วยเชื่อมต่ออุปกรณ์ภายนอก (Input/Output) สำหรับไมโครโปรเซสเซอร์ ในตระกูล x86 ออกแบบโดยใช้หลักการเดียวกันกับที่ วอนนิวแมนได้กำหนดไว้ กล่าวคือ การประมวลผลทั้งหมด ที่หน่วยประมวลผลกลาง สำหรับข้อมูล ไม่ว่าจะเป็นข้อมูลที่ใช้สำหรับประมวลผล หรือข้อมูลที่เป็นคำสั่งก็ตาม จะถูกเก็บไว้ในหน่วยความจำจนกว่าจะมีการเรียกใช้งานจากหน่วยประมวลผลกลาง สำหรับส่วนต่อเชื่อมกับอุปกรณ์ภายนอกนั้น การทำงานของหน่วยประมวลผลกลางก็จะคล้ายกับการทำงานกับหน่วยความจำ ทั้งนี้เพราะทั้งหน่วยความจำและอุปกรณ์ภายนอก เมื่อมีการทำงานกับหน่วยประมวลผลกลางก็มีลักษณะของการเรียกใช้ข้อมูลและนำ ข้อมูลไปเก็บ ต่างกันเพียงสถานที่ และ วิธีการเข้าถึงเท่านั้น บัสของระบบ หน่วยงานต่าง ๆ ภายในเครื่องคอมพิวเตอร์นั้น ต่อเชื่อเข้าด้วยกันด้วยระบบบัส สำหรับโปรเซสเซอร์ในตระกูล 80x86 มีด้วยกัน 3 กลุ่มอันได้แก่ ดาต้าบัส (Data Bus) แอดเดรสบัส (Address Bus) และคอมโทรลบัส (Control Bus) เป็นต้น

คำถามทบทวน

1. ขั้นตอนการทำงานของหน่วยประมวลผลมีทั้งหมดกี่ขั้นตอนอะไรบ้าง
2. เราสามารถแบ่งกลุ่มของบัสออกเป็นกี่กลุ่มอะไรบ้าง จงยกตัวอย่างพร้อมวาดรูปประกอบคำอธิบาย
3. จงแสดงวิธีการแปลงแอดเดรส 234Ah : 44B6h เป็นแอดเดรสขนาด 20 บิต
4. โปรเซสเซอร์ในตระกูล 80x86 ประกอบไปด้วยกลุ่มข้อมูลอะไรบ้าง จงยกตัวอย่างพร้อมวาดรูปประกอบคำอธิบาย
5. รีจิสเตอร์สำหรับใช้งานทั่วไป (General-Purpose Registers) มีอะไรบ้าง ผู้เขียนโปรแกรมสามารถนำรีจิสเตอร์ในกลุ่มไปใช้งานได้อย่างไร จงอธิบายพร้อมยกตัวอย่างประกอบ
6. การทำงานของ SI (Source Index) และ DI (Destination Index) มีหน้าที่ต่างกันอย่างไร
7. รีจิสเตอร์สำหรับการชี้ (Pointer Register) มีอะไรบ้าง พร้อมอธิบายการทำงาน
8. รีจิสเตอร์ใน 8086 มีทั้งขนาด 16 บิต และ 8 บิต แบ่งเป็นกลุ่ม ๆ ได้อย่างไรบ้าง
9. จงอธิบายความแตกต่างระหว่างหน่วยประมวลผลแบบ RISC กับแบบ CISC
10. วิธีการทำงานแบบไปป์ไลน์ (Pipeline) และซูปเปอร์สเกลาร์ (Superscalar) แตกต่างกันอย่างไร

เอกสารอ้างอิง

ราชบัณฑิตยสถาน. (2544). *ศัพท์บัญญัติ ราชบัณฑิตยสถาน.* ค้นเมื่อ 28 ธันวาคม 2556,จาก: http://rirs3.royin.go.th/coinages/

ไมโครโปรเซสเซอร์. (2556). *วิกิพีเดีย สารานุกรมเสรี.* ค้นเมื่อ 28 ธันวาคม 2556, จาก: http: //th.wikipedia.org/wiki/

ชูชัย ธนสารตั้งเจริญ, กำธร พานิชปฐมพงษ์. *ภาษาแอสแซมบลี 80286/80386(PC).* กรุงเทพฯ :สำนักพิมพ์ซีเอ็ดยูเคชั่น บมจ, 2536.

ธีรวัฒน์ ประกอบผล. *ระบบคอมพิวเตอร์และภาษาแอสแซมบลี.* กรุงเทพฯ :สำนักพิมพ์ส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2537.