

บทที่ 1

องค์ประกอบของระบบคอมพิวเตอร์

ระบบคอมพิวเตอร์ที่เราใช้กันอยู่โดยทั่วไปในปัจจุบันประกอบไปด้วยโครงสร้างใหญ่ๆ จำนวน 4 ส่วนดังแสดงในรูปของสไลด์หน้าที่ 5 ได้แก่ ซีพียู หน่วยความจำ ระบบอินพุต และระบบเอาต์พุต โดยซีพียูทำหน้าที่ในการประมวลผลข้อมูล ซึ่งจะรับข้อมูลที่จะประมวลผลเข้ามาจากระบบอินพุต เช่น คีย์บอร์ด เมาส์ สแกนเนอร์ เป็นต้น ในการประมวลผลจะต้องใช้หน่วยความจำในการเก็บข้อมูลชั่วคราวก่อน หรือเราอาจจะมองว่าหน่วยความจำเป็นเหมือนกระดานหกที่ต้องใช้ในการคำนวณก็ได้ จากนั้นจึงเอาผลลัพธ์ที่ได้ออกมาแสดงที่เอาต์พุตเช่น หน้าจอ หรือเครื่องพิมพ์เป็นลำดับถัดไป ในระบบไมโครคอมพิวเตอร์เรามักจะใช้ไมโครโพรเซสเซอร์มาทำหน้าที่เป็นซีพียู (CPU : Central processing unit)

ในการทำงานของเครื่องคอมพิวเตอร์ที่เราใช้งานกันอยู่นี้จะอิงบนหลักการ Stored program concept ของนาย Von Neumann ซึ่งแสดงหลักการของ Von Neumann ไว้ในสไลด์หน้าที่ 6 กล่าวคือ ประการแรก คือเก็บโปรแกรมและข้อมูลไว้ในหน่วยความจำ และอ้างถึงโปรแกรมและข้อมูลผ่านการอ้างอิงตำแหน่งของหน่วยความจำ รูปในสไลด์หน้าที่ 7 แสดงตัวอย่างให้เราเห็นว่าคำว่า “โปรแกรม” กับคำว่า “ข้อมูล” ต่างกันอย่างไร โปรแกรมคือลำดับของคำสั่งที่เขียนขึ้นมาเพื่อให้ไมโครโพรเซสเซอร์ทำงานตามที่ต้องการ ส่วนข้อมูลคืออินพุตที่ป้อนให้โปรแกรมทำการประมวลผลเพื่อสร้างเป็นเอาต์พุตออกมา หากจะยกตัวอย่างให้เห็นได้ชัดๆ ก็คือ การใช้โปรแกรมไมโครซอฟต์เวิร์ดพิมพ์งาน ตัวไมโครซอฟต์เวิร์ดก็คือตัวโปรแกรม ส่วนตัวข้อมูลก็คือข้อความที่เราใส่ในตัวโปรแกรมหรือรูปภาพที่เราแปะลงบนเอกสารในโปรแกรมนั่นเอง ในการพิมพ์งานโดยใช้ไมโครซอฟต์เวิร์ดตัวซีพียูจะรับข้อมูลจากอุปกรณ์อินพุตเช่น คีย์บอร์ดและนำไปเก็บไว้ในหน่วยความจำก่อนที่จะแสดงว่าข้อมูลที่เรารับเข้าไบนั้นเป็นข้อมูลอะไรให้เราเห็นทางเอาต์พุตซึ่งก็คือจอภาพนั่นเอง

จากที่กล่าวมาแล้วว่าโปรแกรมประกอบด้วยคำสั่งต่างๆ ของไมโครโพรเซสเซอร์ ตัวคำสั่งหรือ Instruction เป็นค่าของเลขฐานสองที่ใช้ในการสั่งให้ไมโครโพรเซสเซอร์ทำงานตามที่ต้องการ ในการทำงานของโปรแกรมต่างๆ จะประกอบไปด้วยคำสั่งหลายๆ คำสั่งประกอบเข้าด้วยกันขึ้นมาเป็นโปรแกรม ซึ่งยกตัวอย่างให้เห็นในสไลด์หน้าที่ 9 ซึ่งจะเป็นโปรแกรมในการบวกเลขจำนวน 3 ตัวเข้าด้วยกันคือ $D = A + B + C$ ดังนั้นก่อนที่ซีพียูจะทำการประมวลผลได้เราต้องเก็บค่าของ A, B, C, D เอาไว้ในหน่วยความจำเสียก่อน แล้วจึงนำข้อมูลในหน่วยความจำนี้มาทำการประมวลผลโดยตัวซีพียู

สไลด์หน้าที่ 10-18 แสดงให้เห็นถึงแนวคิดในการทำงานของโปรแกรมบวกเลข 3 ตัวนี้เข้าด้วยกัน โดยเริ่มจากเก็บค่าของ A, B, C, D เอาไว้ในหน่วยความจำตำแหน่ง 10000-10003 ตามลำดับ หน้าที่ของซีพียูคืออ่านข้อมูลในหน่วยความจำตำแหน่ง 10000-10003 นี้มาทำการประมวลผล โดยเริ่ม

จากทำการบวกค่าใน A กับ B ด้วยกันเสียก่อน เมื่อทำเสร็จจึงนำผลลัพธ์ที่ได้มาบวกกับ C เป็นลำดับต่อไปแล้วจึงเขียนผลลัพธ์ที่ได้ลงสู่ D แนวคิดของโปรแกรมเริ่มจากขั้นตอนที่หนึ่ง ซึ่งเป็นการอ่านค่าหน่วยความจำตำแหน่ง 10000 เข้ามาเก็บในซีพียูเสียก่อน ปัญหาคือจะเก็บเอาไว้ในไหนของซีพียู คำตอบคือภายในซีพียูทุกๆ ตัวจะมีแหล่งเก็บข้อมูลชั่วคราวอยู่ภายในหลายๆ ตัว ซึ่งเราเรียกว่ารีจิสเตอร์ (Register) ดังนั้นเราจึงอ่านค่าหน่วยความจำตำแหน่งที่ 10000 เข้ามาเก็บไว้ในรีจิสเตอร์ที่ชื่อว่า รีจิสเตอร์ R0 จากนั้นจึงอ่านค่าจากหน่วยความจำตำแหน่งถัดมาคือตำแหน่งที่ 10001 เข้ามาเก็บในรีจิสเตอร์อีกตัวหนึ่ง คือรีจิสเตอร์ R1 หลังจากนั้นทำการบวกค่าในรีจิสเตอร์ R0 กับรีจิสเตอร์ R1 เข้าด้วยกันแล้วเก็บผลลัพธ์จากการบวกไว้ในรีจิสเตอร์ R2 ในกรณีนี้จะเห็นว่าค่าในรีจิสเตอร์ R0 และรีจิสเตอร์ R1 จะยังคงอยู่เหมือนเดิมไม่เปลี่ยนแปลง แต่ความต้องการของโปรแกรมเราไม่ได้ใช้ค่าในรีจิสเตอร์ R0, R1 อีกต่อไปแล้ว ดังนั้นเราจึงสามารถที่จะใช้รีจิสเตอร์ R0, R1 ในการเก็บค่าอื่นๆ ได้อีก ลำดับต่อมาจึงอ่านค่าจากหน่วยความจำตำแหน่งถัดมาเข้ามาในซีพียูอีกคือหน่วยความจำตำแหน่ง 10002 เข้ามาเก็บในรีจิสเตอร์ R0 จากนั้นทำการบวกค่าในรีจิสเตอร์ R0 กับรีจิสเตอร์ R2 เข้าด้วยกันแล้วเก็บผลลัพธ์การบวกไว้ในรีจิสเตอร์ R1 ซึ่งจะเห็นว่าค่าในรีจิสเตอร์ R1 เป็นผลลัพธ์สุดท้ายจากการคำนวณที่ต้องการแล้ว ดังนั้นเราจึงเขียนค่าในรีจิสเตอร์ R1 ลงสู่หน่วยความจำตำแหน่ง 10003

สไลด์หน้าที่ 19 แสดงให้เห็นถึงโปรแกรมที่ใช้ในการบวกเลขทั้งสามเข้าด้วยกัน ซึ่งจะเห็นว่าโปรแกรมดังกล่าวประกอบไปด้วยคำสั่งจำนวน 6 คำสั่งด้วยกัน ซึ่งจะเห็นว่าแต่ละคำสั่งเราเขียนแทนด้วยคำพูดภาษาไทยนั้นยาวมาก ทางคอลัมน์ทางด้านซ้ายของสไลด์หน้า 19 แสดงให้เห็นถึงการใช้สัญลักษณ์ภาษาอังกฤษแทนคำสั่งจำนวน 6 คำสั่ง โดยคำสั่ง Load จะใช้ในการอ่านค่าจากหน่วยความจำเข้ามาเก็บในซีพียู ส่วนคำสั่ง ADD ใช้ในการสั่งให้ซีพียูบวกค่า และคำสั่ง Store ใช้ในการเก็บค่าในซีพียูลงไปยังหน่วยความจำตำแหน่ง ที่ต้องการ

สไลด์หน้าที่ 20 แสดงให้เห็นถึงการนำหลัก Stored program concept เข้ามาใช้กับโปรแกรมบวกเลขที่เราเขียนขึ้น ซึ่งจะเห็นว่าเราต้องเก็บโปรแกรมและข้อมูลไว้ในหน่วยความจำ โดยในตัวอย่างจะทำการเก็บโปรแกรมไว้ในหน่วยความจำตำแหน่งที่ 15001-15006 และเก็บข้อมูลไว้ในหน่วยความจำตำแหน่ง 10000-10003 จากตัวอย่างข้อมูลที่เรทำการประมวลผลเป็นข้อมูลตัวเลข แต่ข้อมูลที่เรประมวลผลโดยคอมพิวเตอร์มีมากมายหลายรูปแบบไม่ว่าจะเป็นตัวเลข ตัวอักษร รูปภาพ และเนื่องจากหน่วยความจำเก็บข้อมูลได้เฉพาะในรูปแบบตัวเลขฐานสองดังนั้นก่อนที่จะนำข้อมูลเข้ามาประมวลผลเราจะต้องมีการแทนค่าของข้อมูลด้วยค่าตัวเลขเสียก่อนซึ่งจะอธิบายรายละเอียดให้ทราบในบทที่ 2 ต่อไป

โดยหลักการแล้วเราสามารถรับข้อมูลจากอินพุตแล้วนำมาประมวลผลโดยซีพียูทันทีได้โดยไม่ต้องนำไปเก็บในหน่วยความจำก่อน แต่อย่างไรก็ตามเนื่องจากจำนวนรีจิสเตอร์ภายในซีพียูมีจำนวนจำกัดมาก ดังนั้นในทางปฏิบัติจึงมักนิยมให้ซีพียูรับข้อมูลจากอินพุตเข้ามาเก็บในหน่วยความจำก่อน

แล้วจึงค่อยนำข้อมูลในหน่วยความจำไปประมวลผลต่อไป แต่ในบางครั้งเราสามารถสั่งให้อุปกรณ์ อินพุตส่งข้อมูลให้กับหน่วยความจำได้โดยตรงโดยไม่ต้องผ่านซีพียูก็สามารถทำได้ดังแสดงในสไลด์ หน้าที่ 24 ซึ่งเราเรียกการทำการดังกล่าวนี้ว่าการทำ DMA (Direct memory access) ซึ่งจะอธิบาย รายละเอียดในบทที่ 5 ซึ่งจะกล่าวรายละเอียดของระบบอินพุตเอาต์พุต

เราอาจกล่าวได้ว่าซีพียูคือหัวใจหลักของเครื่องคอมพิวเตอร์ สถาปัตยกรรมของซีพียูที่ใช้ใน เครื่องไมโครคอมพิวเตอร์ปัจจุบันจะเป็นสถาปัตยกรรม 80x86 ของบริษัทอินเทล แต่จริงๆ แล้ว สถาปัตยกรรมของไมโครโพรเซสเซอร์มีอยู่หลายตระกูลดังแสดงให้เห็นในสไลด์หน้าที่ 25 สถาปัตยกรรมอื่นๆ จะมีใช้ในเครื่องเวิร์กสเตชันและเครื่องระดับมินิคอมพิวเตอร์ขึ้นไป เนื่องจากมีราคาสูงมาก สาเหตุที่ทำให้สถาปัตยกรรมของอินเทลได้รับความนิยมอย่างหนึ่งก็คือราคาที่ถูกลงของมันนั่นเอง สไลด์ หน้าที่ 26 แสดงให้เห็นถึงภาพรวมของวิวัฒนาการของซีพียูตระกูล 80x86 ของอินเทล

เรามักจะใช้คำว่าซีพียู ชิพ และไมโครโพรเซสเซอร์สลับกันอยู่บ่อยๆ เนื่องจาก 3 คำนี้มีความ เกี่ยวข้องกันอยู่ คำว่าซีพียูความจริงแล้วคือหน่วยประมวลผลกลางของระบบซึ่งซีพียูนี้นี้ปกติจะสร้างมา จากไมโครโพรเซสเซอร์ ดังนั้นเราจึงใช้สองคำนี้แทนกันได้หลายๆ กรณี สไลด์หน้าที่ 27-29 แสดง ให้เห็นว่าในระบบหนึ่งระบบอาจมีไมโครโพรเซสเซอร์มากกว่า 1 ตัวแต่จะมีอยู่เพียงตัวเดียวที่ทำหน้าที่ เป็นหน่วยประมวลผลกลางของระบบหรือซีพียูนั่นเอง ส่วนคำว่าชิปนั้นมีที่มาคือภายในไมโคร โพรเซสเซอร์ทุกตัวจะประกอบไปด้วยซิลิคอนชิปขนาดเล็กซึ่งรวมเอาทรานซิสเตอร์จำนวนหลายล้าน ตัวเข้าไว้ด้วยกัน ตัวชิปจริงๆ แล้วจะมีขนาดเล็กมากเมื่อเทียบกับตัวถังของไมโครโพรเซสเซอร์ดังแสดง ให้เห็นในสไลด์หน้าที่ 29 สาเหตุที่เราสร้างตัวถังให้มีขนาดใหญ่เนื่องมาจากการระบายความร้อน ของตัวชิปออกไปสู่ภายนอกให้เร็วที่สุด

สไลด์หน้าที่ 30-39 แสดงให้เห็นถึงวิวัฒนาการในแต่ละขั้นของซีพียูตระกูล 80x86 ของอินเทล โดยเริ่มจากเบอร์ 4004 ซึ่งมีขนาดแค่ 4 บิต และมีจำนวนทรานซิสเตอร์ภายในเพียงแค่ 2300 ตัวจน พัฒนามาเป็นเบอร์ 8086 ซึ่งเป็นซีพียูขนาด 16 บิตและสามารถต่อหน่วยความจำได้มากขึ้นเป็น 20 เมกะไบต์ และมีความเร็วของสัญญาณนาฬิกาในการทำงานที่สูงขึ้น ซีพียูในรุ่นถัดมาจะปรับปรุงขนาด ของซีพียูเป็น 32 บิต โดย 80386 เป็นซีพียู 32 บิตตัวแรกของอินเทล ซีพียู 80486 เป็นซีพียูตัวแรกที่ นำเอาหลักการที่เรียกว่า Pipelining technique มาใช้เพิ่มความเร็วในการทำงานซึ่งเราจะได้เรียน รายละเอียดหลักการของระบบไปปลายนี้ในบทที่ 7 นอกเหนือจากนี้ในซีพียู 80486 ยังเพิ่มหลักการใช หน่วยความจำแคช (Cache) เข้ามาใส่ไว้ภายในซีพียูอีกด้วย โดยเราจะเรียนรายละเอียดการทำงานของ หน่วยความจำแคชในบทที่ 8

ซีพียูรุ่นถัดมาจาก 80486 คือซีพียู Pentium ซึ่งได้นำสถาปัตยกรรมแบบ Superscalar มาใช้ เพื่อให้ทำงานได้ด้วยความเร็วมากขึ้น โดยเราจะเรียนหลักการของ Superscalar ในรายละเอียดในบทที่ 7 หลังจากอินเทลออกซีพียู Pentium แล้ว ก็มีซีพียู Pentium2-Pentium4 ออกตามกันมาตามลำดับ โดย

ซีพียู Pentium2 เป็นซีพียูตัวแรกที่น่าเอาหลักการประมวลผลที่เรียกว่า out-of-order execution มาใช้ในการปรับปรุงความเร็วในการทำงานซึ่งเราจะเรียนรายละเอียดการทำงานในบทที่ 9

คำว่าขนาดของซีพียูที่เราวัดกันเป็นจำนวนบิต จะหมายถึงความสามารถในการประมวลผลตัวเลขได้ในเวลาใดเวลาหนึ่ง เช่น ซีพียู ขนาด 16 บิตจะสามารถทำการคำนวณเลขขนาด 16 บิตได้ภายในการทำงานแค่ครั้งเดียว แต่ถ้าเราต้องการนำซีพียูขนาด 16 บิตไปคำนวณเลขขนาด 32 บิตก็สามารถทำได้โดยการคำนวณหลายๆ ครั้ง จะเห็นว่าหากเราเพิ่มขนาดของไมโครโปรเซสเซอร์จะสามารถลดเวลาที่ต้องใช้ในการทำงานของไมโครโปรเซสเซอร์ลงได้ หรือจะพูดอีกแง่หนึ่งก็คือจะทำให้ความเร็วในการทำงานเพิ่มขึ้นนั่นเอง
