ระบบคอมพิวเตอร์และสถาปัตยกรรม
(Computer System and Architecture)

บทที่ 11
อินพุต/เอาต์พุต
(Input/Output)

อินพุต/เอาต์พุต (Input/Output)

- การใช้ประโยชน์ของระบบคอมพิวเตอร์ขึ้นอยู่กับสิ่งอำนวยความสะดวกทางอินพุต/เอาต์พุต หรือไอโอ (I/O : Input/Output) ซึ่งเป็นการติดต่อระหว่างผู้ใช้กับคอมพิวเตอร์
- การเพิ่มความเร็วของพีซีโดยเพิ่มความเร็วของโปรเซสเซอร์ แต่ถ้าไม่ลดข้อจำกัดความเร็วในการทำงานกับไอโอจะทำให้ประสิทธิภาพของคอมพิวเตอร์ไม่ดีเท่าที่ควร

อินพุต/เอาต์พุต (Input/Output)

- กำหนดโครงสร้างอินพุตและเอาต์พุตในคอมพิวเตอร์โดยการให้มีตะกร้าอินพุต และตะกร้าเอาต์พุต
- แต่ละคำสั่งของอินพุตจะถ่ายโอนข้อมูลตะกร้าอินพุตไปยังเครื่องคิดเลข
- ในแต่ละคำสั่งของเอาต์พุตจะทำการถ่ายโอนข้อมูลออกจากเครื่องคิดเลขไปไว้ในตะกร้าเอาต์พุต
- ในคอมพิวเตอร์จริง ตะกร้าอินพุตและตะกร้าเอาต์พุตจะถูกแทนด้วยบัสอินเทอร์เฟซ

อินพุต/เอาต์พุต (Input/Output)

- ปริมาณการถ่ายโอนข้อมูลของไอโอที่ทำงานร่วมกันหลายตัวมีผลต่อการทำงาน โดยเฉพาะไอโอที่มีความเร็วในการถ่ายโอน ข้อมูลต่างกันเป็นอย่างมาก ทำให้ต้องมีการซิงโครไนซ์เพื่อป้องกันข้อมูลสูญหาย
- การทำงานของไอโอจะใช้เวลาซีพียูมาก เนื่องจากเวลาส่วนมากเกิดจากการรอจนกว่างานจะเสร็จสมบูรณ์
- มีวิธีการมากมายถูกนำมาใช้เพื่อแก้ปัญหาการซิงโครไนซ์และควบคุมการทำงานของไอโอในระหว่างที่ไอโอดีไวซ์ที่หลากหลายกำลังทำงานในปริมาณข้อมูลและความเร็วที่ต่างกัน

คุณสมบัติพื้นฐานของอินพุต/เอาต์พุต

- คีย์บอร์ดเป็นดีไวซ์สำหรับอินพุต อินพุตพื้นฐานที่เป็นตัวอักษร ที่มีอัตราการถ่ายโอนข้อมูลเป็นไปอย่างช้ามากเพื่อเปรียบเทียบกับการโปรเซสของซีพียู เนื่องจากความเร็วในการคีย์ของผู้ใช้งาน
- ปกติคอมพิวเตอร์ทำงานเพียงงานเดียว ซีพียูจะใช้เวลาส่วนมากในการรออินพุตจากคีย์บอร์ด ทำให้คอมพิวเตอร์ทำงานไม่เต็มประสิทธิภาพ
- การอินพุตด้วยคีย์บอร์ดมี 2 ประเภท
  - อินพุตที่แอปพลิเคชันคาดหมายไว้ (expected input) ในการตอบสนองโปรแกรมที่ร้องขอข้อมูล
  - อินพุตที่ไม่ได้คาดหมาย (unexpected input) เช่น การกดคีย์ Control-Alt-Delete บนพีซีจะเป็นการสตาร์ตคอมพิวเตอร์ใหม่ เป็นต้น

คุณสมบัติพื้นฐานของอินพุต/เอาต์พุต

- เมาส์เป็นอินพุตดีไวซ์ที่สามารถสร้างอินพุตได้ทั้งที่คาดหมายและไม่ได้คาดหมาย เมื่อเคลื่อนเมาส์แล้วคลิกจะเปลี่ยนวิธีเอ็กซิคิวต์โปรแกรม
- จอภาพและเครื่องพิมพ์ทำงานในอัตราการถ่ายโอนข้อมูลที่กว้าง (ตั้งแต่ความเร็วต่ำไปจนถึงความเร็วสูง)
- สำหรับเอาต์พุตที่เป็นกราฟิก อาจจะอยู่ในรูปแบบออปเจ็กต์หรือบิตแมพ ก็ได้ แต่ส่วนมากจะแสดงผลในรูปแบบของบิตแมพ ซึ่งเป็นข้อมูลขนาดใหญ่ นอกจากนี้ทั้งภาพวีดิโอภาพกราฟิกความละเอียด กราฟิกเหล่านี้ต้องการการถ่ายโอนข้อมูลที่มีความเร็วสูง เพื่อให้การใช้งานมีประสิทธิภาพ

คุณสมบัติพื้นฐานของอินพุต/เอาต์พุต

- การที่ดิสก์เป็นดีไวซ์ที่ใช้ในการเก็บข้อมูลทำให้ดิสก์เป็นได้ทั้งอินพุตและเอาต์พุตแต่ในคนละเวลากัน การถ่ายโอนข้อมูลของดิสก์มีความเร็วสูง
- ดีวีรอมพยายามที่จะแสดงวิดีโอเต็มจอภาพในอัตราของภาพยนตร์ อย่างต่อเนื่องจะต้องมีความคงที่ของข้อมูลที่มีอัตราอินพุตประมาณ 10 เมกะไบต์ต่อวินาที
- เนื่องจากเครือข่ายมีความเร็วสูง ทำให้มีการใช้เครือข่ายแทนดิสก์ในการเก็บโปรแกรมและข้อมูลในคอมพิวเตอร์ที่อยู่ห่างไกล
- อัตราการถ่ายโอนข้อมูลของไอโอดีไวซ์ต่าง ๆ แสดงดังตาราง

คุณสมบัติพื้นฐานของอินพุต/เอาต์พุต

ดีไวซ์	ไอโอ	อัตราการถ่ายโอนข้อมูล (กิโลไบต์ต่อวินาที)
คีย์บอร์ด	อินพุต	0.01
เมาส์	อินพุต	0.02
อินพุตเสียง	อินพุต	0.02
สแกนเนอร์	อินพุต	200
เอาต์พุตเสียง	เอาต์พุต	0.5
เครื่องพิมพ์อิงค์เจ็ต	เอาต์พุต	1.5
เครื่องพิมพ์เลเซอร์	เอาต์พุต	100-1000
การแสดงผล	เอาต์พุต	30000
เครือข่ายเฉพาะบริเวณ	อินพุต หรือเอาต์พุต	200-20000
อ็อพติคอลดิสก์	สื่อจัดเก็บข้อมูล	500-15000
เทปแม่เหล็ก	สื่อจัดเก็บข้อมูล	1000-15000
ดิสก์แม่เหล็ก	สื่อจัดเก็บข้อมูล	2000-60000
แหล่งข้อมูลวิดีโอที่ไม่ได้บีบอัดข้อมูล	อินพุต	1000
แหล่งข้อมูลเสียงที่ไม่ได้บีบอัดข้อมูล	อินพุต	100

ความต้องการทั่วไปสำหรับระบบคอมพิวเตอร์ที่รองรับไอโอ

- ต้องมีการกำหนดแอ็ดเดรสเฉพาะให้แต่ละดีไวซ์ที่ต่อพ่วง
- ต้องมีวิธีการที่ดีไวซ์ต่อพ่วงสามารถเริ่มสื่อสารกับซีพียู
- การควบคุมไอโอด้วยโปรแกรมมีความเหมาะสมเฉพาะสำหรับ ดีไวซ์ที่มีความเร็วต่ำ และมีการถ่ายโอนข้อมูลเป็นเวิร์ด แต่สำหรับดีไวซ์ความเร็วสูงที่มีการถ่ายโอนข้อมูลเป็นบล็อกจะมีการถ่ายโอนข้อมูลที่มีประสิทธิภาพสูงกว่าในการถ่ายโอนข้อมูลระหว่างไอโอกับหน่วยความจำ
- ต้องสามารถดูแลดีไวซ์ที่มีการควบคุมที่ต่างกันให้สามารถทำงานได้

การใช้อินเทอร์เฟซโมดูลในการเชื่อมต่อไอโอดีไวซ์โดยตรงกับซีพียู

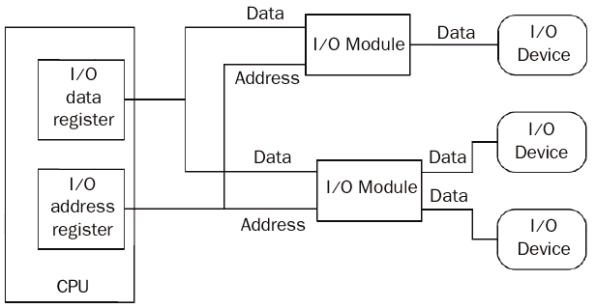
- รูปแบบที่ดีไวซ์ต่าง ๆ ต้องการข้อมูลที่แตกต่างกันทำให้ระบบใช้อินเทอร์เฟซทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ต่างกัน
- ความเร็วที่ต่างกันของดีไวซ์กับซีพียูทำให้การซิงโครไนซ์ยุ่งยากโดยเฉพาะถ้ามีหลาย ๆ ดีไวซ์พยายามกระทำต่อไอโอพร้อม ๆ กัน ทำให้อาจจะต้องใช้บัฟเฟอร์
- การเชื่อมต่อภายในที่สนับสนุนการให้บริการมัลติมีเดียจะต้องมีความสามารถในการส่งข้อมูลอย่างต่อเนื่อง รวมทั้งเครือข่ายและดีไวซ์ที่ติดต่อสื่อสารกันด้วยความเร็วสูง
- ดีไวซ์จำพวกดิสก์ไดรฟ์มีการควบคุมด้วยกลไกทางอิเล็กทรอนิกส์ที่ต้องใช้เวลาซีพียูจำนวนมากในการควบคุม

การใช้อินเทอร์เฟซโมดูลในการเชื่อมต่อไอโอดีไวซ์โดยตรงกับซีพียู

- ความแตกต่างสำหรับแต่ละไอโอดีไวซ์รวมกับความจำเป็นในการกำหนดแอ็ดเดรส, การซิงโครไนซ์, สถานะ และความสามารถในการควบคุม ภายนอก ทำให้มีความจำเป็นที่จะต้องรองรับแต่ละดีไวซ์ด้วยอินเทอร์เฟซพิเศษโดยเฉพาะ
- โดยปกติไอโอดีไวซ์จะเชื่อมต่อกับซีพียูผ่านทางไอโอโมดูล (I/O module)
- ไอโอโมดูลต้องมีความสามารถในการถ่ายโอนข้อมูลเป็นบล็อกที่ใช้บัฟเฟอร์อย่างเหมาะสมและเป็นมาตรฐานในการอินเทอร์เฟซกับซีพียู และมีความสามารถในการควบคุมดีไวซ์พิเศษที่ถูกออกแบบขึ้นมาด้วย
- ไอโอโมดูลที่ควบคุมดีไวซ์ประเภทเดียวจะเรียกว่า “ดีไวซ์คอนโทรลเลอร์”
- ถ้ามีดีไวซ์มากขึ้น ต้องมีการกำหนดแอ็ดเดรสเพื่อแยกออกจากโมดูลอื่น

การใช้อินเทอร์เฟซโมดูลในการเชื่อมต่อไอโอดีไวซ์โดยตรงกับซีพียู

การจัดเรียงไอโอโมดูลอย่างง่าย



การจัดเรียงไอโอโมดูลที่มีความสลับซับซ้อน

ปัจจัยการออกแบบไอโอ

การออกแบบระบบไอโอให้เหมาะสมจะทำให้ซีพียูทำงานได้อย่างมีประสิทธิภาพ ทั้งการซิงโครไนซ์ดีไวซ์หลายดีไวซ์ การปรับเวลาดีไวซ์ที่มีอัตราการถ่าย โอนข้อมูลต่างกัน และการปรับเวลาให้ซีพียูทำงานโดยไม่เสียเวลารอในขณะที่ไอโอกำลังทำงาน เหล่านี้เป็นตัววัดประสิทธิภาพของคอมพิวเตอร์ สำหรับ ปัจจัยที่ควรพิจารณาในการออกแบบระบบไอโอมี 3 ประการคือ

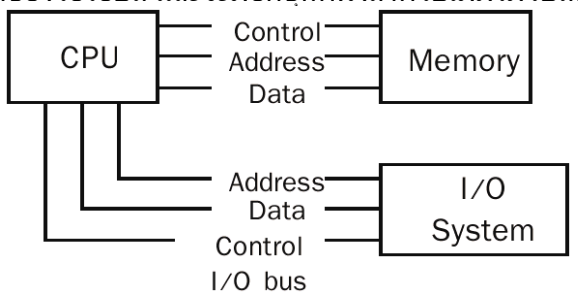
- **ตำแหน่งข้อมูล** : การเลือกดีไวซ์, แอ็ดเดรสของข้อมูลในดีไวซ์
- **การถ่ายโอนข้อมูล** : ปริมาณ, อัตราการถ่ายโอนทั้งในขณะเข้าหรือออก จากดีไวซ์
- **การซิงโครไนซ์** : ให้เอาต์พุตเฉพาะเมื่อดีไวซ์พร้อม, รับอินพุตเมื่อมีข้อมูล

มาตรฐานของไอโออินเทอร์เฟซ

- ปัจจัยในการออกแบบระบบไอโอทั้ง 3 ประการจำเป็นต้องมีการถ่ายโอนข้อมูลไอโอออกมา
- การถ่ายโอนข้อมูลขึ้นกับประเภทของไอโอดีไวซ์ และต้องเป็นมาตรฐาน
- การเชื่อมต่อซีพียูและไอโอดีไวซ์เป็นการกำหนดโครงสร้างข้อมูลที่ดีไวซ์อินเทอร์เฟซตามรายละเอียด เช่น การรวมบิตเป็นเวิร์ด, การตรวจสอบและแก้ไขข้อผิดพลาดใน low-level และการรับเวิร์ดในรีจิสเตอร์ที่อยู่ในแอ็ดเดรสในที่ว่างในไอโอ
- การเชื่อมต่อระหว่างซีพียูและและไอโอดีไวซ์เป็นการกำหนดให้กับดีไวซ์ในการเขียนข้อมูล มีการกำหนดคุณสมบัติให้อ่าน หรือเขียนข้อมูล

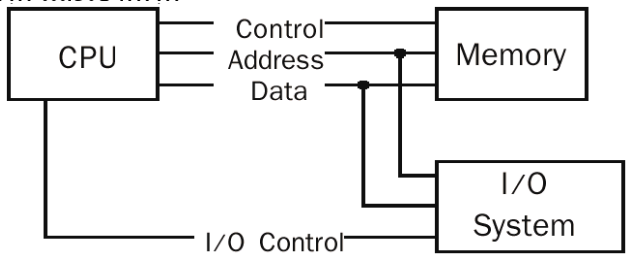
โครงสร้างของไอโอบัส

- **Isolated I/O** จะมีชุดของแอ็ดเดรสบัส, ดาต้าบัส และคอนโทรลบัส เป็นชุดของไอโอบัส และจะมีอีกชุดหนึ่งสำหรับเมมโมรีบัสที่แยกจากกัน



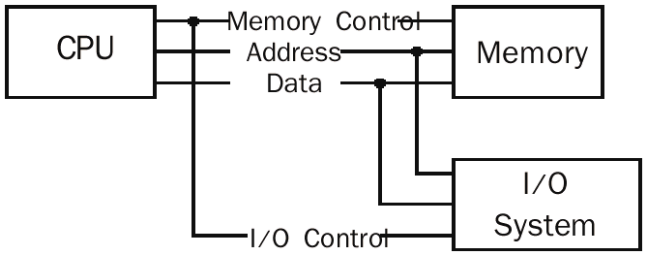
โครงสร้างของไอโอบัส

- **Shared I/O** จะใช้แอ็ดเดรสบัสและดาต้าบัสร่วมกันเป็นไอโอบัสและเมมโมรีบัสแต่มีสัญญาณควบคุมคนละเส้นสำหรับการอ่าน, เขียน, อินพุต และเอาต์พุต



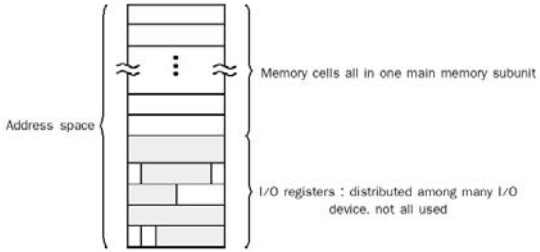
โครงสร้างของไอโอบัส

- **Memory-map I/O** จะมีบัสหนึ่งที่ใช้ทั้งในไอโอและหน่วยความจำ ในรูปเป็นการใช้คอนโทรลบัสร่วมกัน ช่วงของเมมโมรีแอ็ดเดรสจะถูกกันไว้สำหรับไอโอรีจิสเตอร์ด้วย



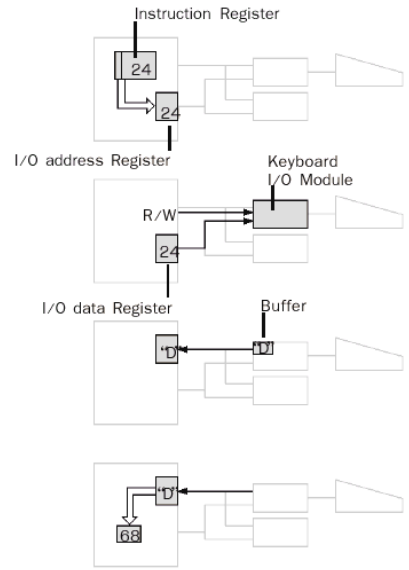
โครงสร้างของไอโอบัส

- พื้นที่แอ็ดเดรสทั้งหมดของระบบคอมพิวเตอร์ที่เป็นแบบ Memory-map I/O นั้น หน่วยความจำประกอบด้วยเมมโมรีโมดูล 2-3 โมดูล จะมีไอโอ รีจิสเตอร์จะกระจายไปตามการอินเทอร์เฟซของไอโอดีไวซ์ มีเพียง แอ็ดเดรสบางส่วนของไอโอรีจิสเตอร์เท่านั้นที่ใช้ในระบบ



เทคนิคการทำงานของไอโอ

- **การควบคุมอินพุต/เอาต์พุตด้วยโปรแกรม** (Programmed I/O) ซึ่งเป็นเทคนิคที่มีการแลกเปลี่ยนข้อมูลระหว่างโปรเซสเซอร์และไอโอโมดูล โปรเซสเซอร์จะเอ็กซิคิวต์โปรแกรมที่ควบคุมโอเปอเรชันของไอโอ รวมทั้งการตรวจสอบสถานะของดีไวซ์, การส่งคำสั่งเพื่ออ่านหรือเขียน และการถ่ายโอนข้อมูล
- การเอ็กซิคิวต์คำสั่งไอโอนั้น โปรเซสเซอร์จะต้องกำหนดแอ็ดเดรส, ไอโอโมดูล, ดีไวซ์และคำสั่งของไอโอซึ่งมี 4 ประเภทซึ่งกำหนดโดยโปรเซสเซอร์คือ **ควบคุม** (Control), **ตรวจสอบ** (Test), **อ่าน** (Read) และ**เขียน** (Write)



1. ซีพียูเอ็กซิคิวต์คำสั่ง Input ที่แอ็ดเดรส 24 ทำให้มีการก็อปปี้แอ็ดเดรสที่ 24 ไปเก็บไว้ใน I/O address register

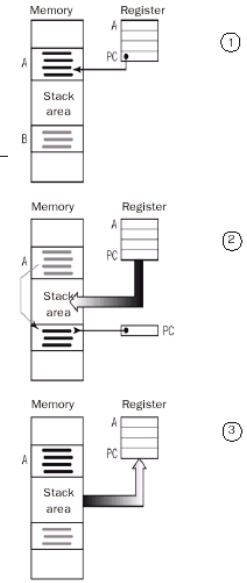
2. แอ็ดเดรส 24 เป็นไอโอโมดูลของคีย์บอร์ด สายสัญญาณควบคุมการอ่าน/เขียนชี้ให้เห็นว่าคำสั่งนั้นเป็น Input

3. บัฟเฟอร์ในไอโอโมดูลจะเก็บค่าการกดคีย์ ในกรณีนี้รหัสแอสกี้ 68 ซึ่งเป็นตัวอักษร D ข้อมูลจะถูกถ่ายโอนไปยัง I/O data register

4. ข้อมูลใน I/O data register จะถูกก็อปปี้ไปยังแอ็กคิวมูเลเตอร์ หรือรีจิสเตอร์ เป็นการสิ้นสุดโอเปอเรชัน

เทคนิคการทำงานของไอโอ

- **การควบคุมอินพุต/เอาต์พุตผ่านกลไกการอินเทอร์รัพต์** (Interrupt-driven I/O) ซึ่งเทคนิคนี้โปรเซสเซอร์จะออกคำสั่งไปยังไอโอ แล้วกลับไปเอ็กซิคิวต์คำสั่งอื่นต่อไป จนเมื่อไอโอทำงานเสร็จเรียบร้อยแล้วจะส่งสัญญาณ Interrupt แจ้งให้ทราบ



① 1. ก่อนการอินเทอร์รัพต์ โปรแกรม A ซึ่งกำลังถูกเอ็กซิคิวต์อยู่ โปรแกรมเคาเตอร์ (PC) ชี้อยู่ที่คำสั่งปัจจุบัน

② 2. เมื่อสัญญาณอินเทอร์รัพต์ถูกส่งเข้ามาที่ซีพียู จะมีการเอ็กซิคิวต์คำสั่งปัจจุบันให้เสร็จเรียบร้อยก่อน หลังจากนั้น ข้อมูลทั้งหมดที่เกี่ยวข้องกับโปรแกรมที่หยุดเอ็กซิคิวต์, ตำแหน่งของคำสั่งสุดท้าย และข้อมูลในรีจิสเตอร์ทุกตัวจะถูกเก็บในพื้นที่ที่เป็นสแต็ก หรือพื้นที่พิเศษที่เรียกว่า PCB (Process Control Block)

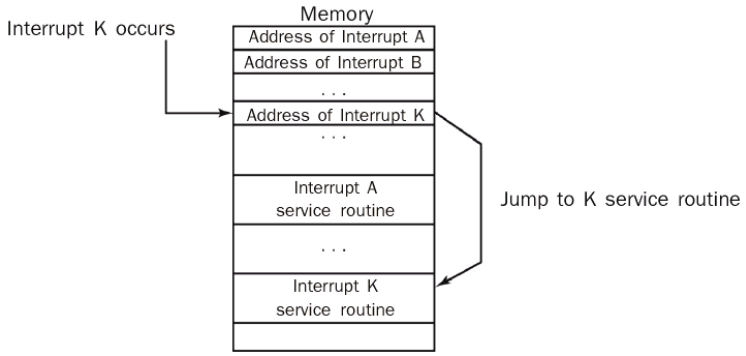
③ 3. เมื่อรูทีนของอินเทอร์รัพต์ถูกเอ็กซิคิวต์เรียบร้อยแล้ว ค่าในรีจิสเตอร์จะถูกคืนค่า รวมทั้งโปรแกรมเคาเตอร์ด้วย ทำให้โปรเซสเซอร์กลับมาเอ็กซิคิวต์คำสั่งต่อจากคำสั่งเดิมก่อนที่จะมีสัญญาณอินเทอร์รัพต์เข้ามา

การจัดลำดับความสำคัญของ Interrupt

- ถ้ามีInterrupt เกิดขึ้นมากมายในเวลาเดียวกัน ทำให้มีคำถามที่น่าสนใจ 2 คำถาม
  - คำถามแรกก็คือถ้ามี Interrupt เกิดขึ้นพร้อมกันหลาย Interrupt เพื่อคอยการตอบสนองจากซีพียู แล้วซีพียูจะมีวิธีการจัดลำดับการตอบสนอง Interrupt เหล่านั้นอย่างไร
  - คำถามที่สองก็คือซีพียูทราบได้อย่างไรว่าดีไวซ์ใดส่ง Interrupt เข้ามา

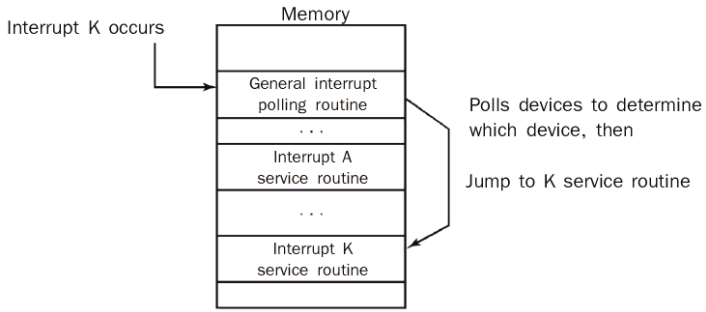
วิธีหาดีไวซ์ที่ส่ง Interrupt

- **Vectored interrupt** ซึ่งในสัญญาณ **interrupt** นั้นจะมี address ของ **interrupt** device รวมมาด้วย



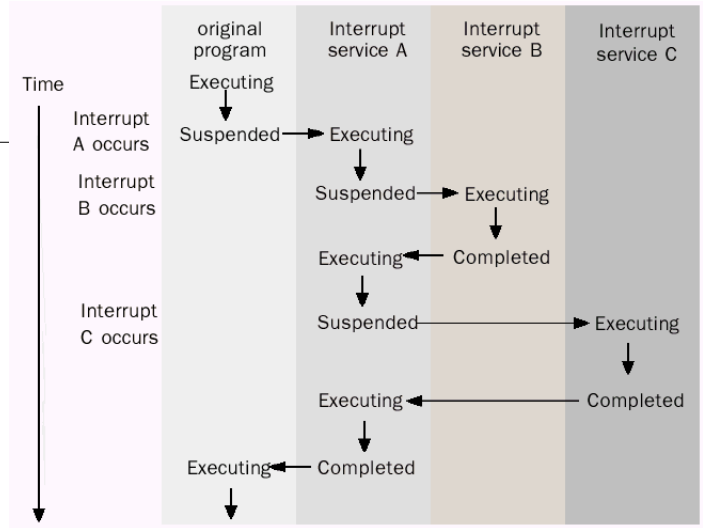
วิธีหา device ที่ส่ง interrupt

- **Polled interrupt** โดยใช้พูลลิ่งแต่ละดีไวซ์ ใช้ “daisy chaining” ซึ่งจะวาง **interrupt** ไปบนสาย **interrupt** เพียงเส้นเดียวไปยังซีพียูซึ่งกำหนดเป็นลำดับความสำคัญสูงสุดที่ซีพียูทราบได้ก่อน



การจัดการเมื่อมีหลาย interrupt

- จัดการได้โดยกำหนดลำดับความสำคัญให้แต่ละ **interrupt**
- จัดการกับ **interrupt** ที่มีลำดับความสำคัญสูงสุดก่อน หลังจากนั้นจะจัดการกับ **interrupt** ที่มีลำดับสูงกว่าตัวอื่น เรื่อยไปจนถึง **interrupt** ที่มีลำดับต่ำสุด
- ทำให้มีลำดับชั้นในการจัดการ **interrupt** ซึ่งอาจจะมี **interrupt** ซ้อนกันอยู่ในโปรแกรมที่กำลังทำงานอยู่
- การจัดการยังใช้หลักการที่ว่าจะต้องจัดการ **interrupt** ที่มีลำดับสูงกว่าก่อนจนเสร็จสมบูรณ์แล้วจึงจัดการ **interrupt** ที่มีลำดับต่ำกว่าจนกว่าจะหมดทุก **interrupt** แล้วจึงกลับมาทำงานต่อ



การทำงานเมื่อเกิด **interrupt** หลายตัว

เทคนิคการทำงานของไอโอ

- **การเข้าถึงหน่วยความจำโดยตรง** หรือ DMA (Direct Memory Access) ในเทคนิคนี้จะมีการแลกเปลี่ยนข้อมูลกันโดยตรงระหว่างไอโอโมดูลและหน่วยความจำหลัก โดยโปรเซสเซอร์ไม่มีส่วนเกี่ยวข้องเลย จะต้องมีเงื่อนไขหลัก 3 ประการคือ
  - ต้องมีวิธีการเชื่อมต่อของไอโอและหน่วยความจำ
  - ไอโอโมดูลที่สัมพันธ์กับดีไวซ์เฉพาะแบบจะต้องสามารถอ่านและเขียนข้อมูลในหน่วยความจำได้
  - ต้องหลีกเลี่ยงปัญหาระหว่างซีพียูและไอโอโมดูล

การเข้าถึงหน่วยความจำโดยตรง

- เหมาะสำหรับการถ่ายโอนข้อมูลบนดิสก์ความเร็วสูง
- ซีพียูไม่มีส่วนเกี่ยวข้องในขณะถ่ายโอน ทำให้สามารถทำงานอื่นได้
- มีประโยชน์เป็นอย่างมากใน Multi User systems
- การถ่ายโอนจะอยู่ในทิศทางใดทิศทางหนึ่ง
- ใช้โมดูลเพิ่มเติมที่เรียกว่า ดีเอ็มเอโมดูล (DMA Module) ในบัสหลัก
- เลียนแบบการทำงานของโปรเซสเซอร์ สามารถถ่ายโอนข้อมูลในหน่วยความจำผ่านบัสหลักได้
- ไม่ขัดขวางกาทำงานของโปรเซสเซอร์ ทำให้ทำงานได้อย่างต่อเนื่อง การทำงานแบบนี้เรียกว่า “การขโมยวงรอบเวลา” (cycle stealing)

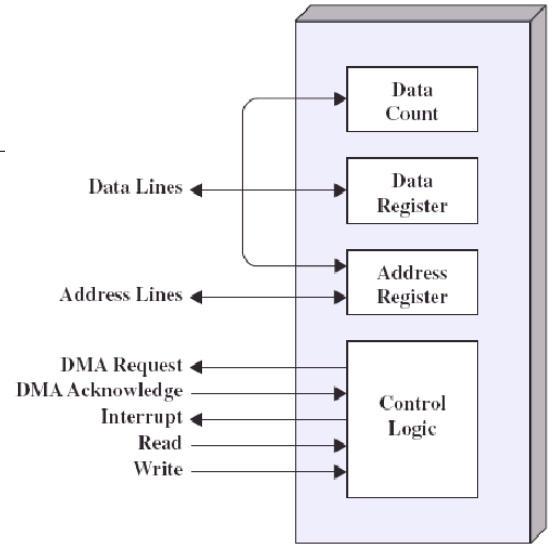
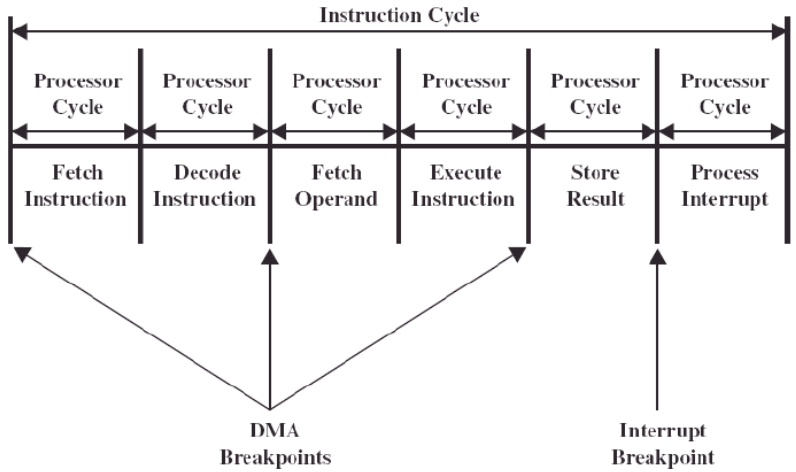


Diagram ของการเข้าถึงหน่วยความจำโดยตรง



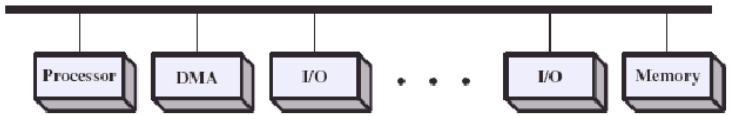
การเข้าถึงหน่วยความจำโดยตรงและจุดแยกสัญญาณ **interrupt** ในวงรอบคำสั่ง

ข้อมูลที่ต้องมีเพื่อควบคุมการเข้าถึงหน่วยความจำโดยตรง

- ตำแหน่งของข้อมูลของไอโอดีไวซ์
- ตำแหน่งเริ่มต้นของบล็อกข้อมูลในหน่วยความจำ
- ขนาดของบล็อกที่ถูกถ่ายโอน
- ทิศทางของการถ่ายโอน, อ่าน (ไอโอไปหาหน่วยความจำ) หรือเขียน (หน่วยความจำไปหาไอโอ)

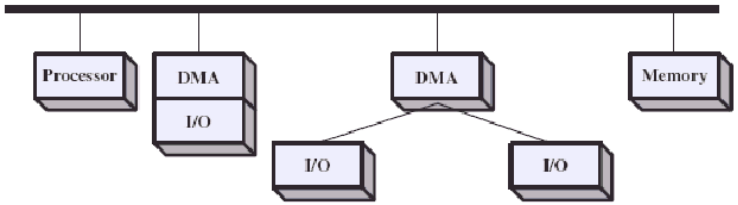
รูปแบบการเข้าถึงหน่วยความจำโดยตรง

- **ทุกโมดูลใช้บัสร่วมกัน** โดยดีเอ็มเอโมดูลจะทำหน้าที่แทนโปรเซสเซอร์ , ใช้การควบคุมอินพุต/เอาต์พุตผ่านโปรแกรมเพื่อแลกเปลี่ยนข้อมูลระหว่างหน่วยความจำและไอโอโมดูลผ่านทางดีเอ็มเอโมดูล



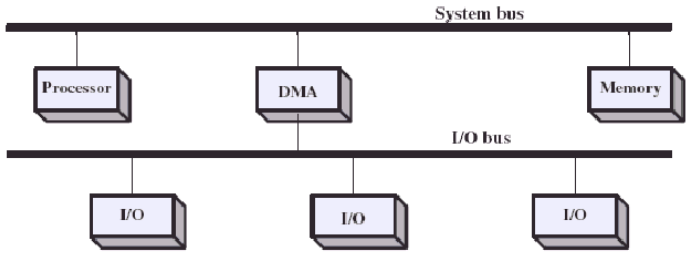
รูปแบบการเข้าถึงหน่วยความจำโดยตรง

- **การรวมดีเอ็มเอฟังก์ชันเข้ากับไอโอฟังก์ชัน** ทำให้ลดจำนวนบัส โดยจะมีเส้นทางที่แยกจากบัสเชื่อมต่อระหว่างดีเอ็มเอโมดูลและไอโอโมดูล ในทางลอจิกการทำงานของดีเอ็มเอโมดูลเป็นส่วนหนึ่งของไอโอโมดูล หรืออาจจะแยกเป็นโมดูลต่างหากที่ควบคุมการทำงานของไอโอโมดูลอื่น ๆ ก็ได้



รูปแบบการเข้าถึงหน่วยความจำโดยตรง

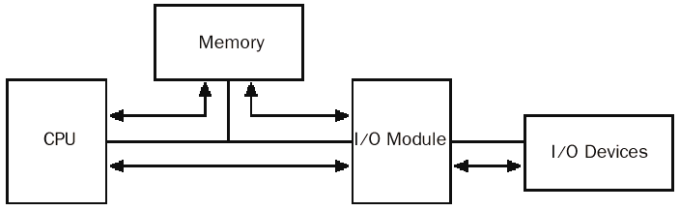
- **การเชื่อมต่อไอโอโมดูลกับดีเอ็มเอโมดูลโดยใช้ไอโอบัส** การเชื่อมต่อแบบนี้จะช่วยลดจำนวนไอโออินเทอร์เฟซในดีเอ็มเอโมดูลลดเหลือเพียงอินเทอร์เฟซเดียว และยังช่วยให้สามารถขยายความสามารถออกไปได้ด้วย



สถาปัตยกรรมซีพียู-หน่วยความจำ-อินพุต/เอาต์พุต

- องค์ประกอบพื้นฐานการอินเทอร์เฟซระหว่างซีพียูและไอโอดีไวซ์
  - ซีพียู
  - ไอโอดีไวซ์
  - หน่วยความจำ, ยกเว้นอินพุต หรือเอาต์พุตชิ้นเดี่ยวที่สามารถถูกถ่ายโอนโดยตรงจากรีจิสเตอร์
  - ไอโอโมดูลที่เป็นเสมือนอินเทอร์เฟซระหว่างซีพียูหรือหน่วยความจำและไอโอดีไวซ์
  - บัสที่เชื่อมต่อองค์ประกอบเหล่านั้นเข้าด้วยกัน

เส้นทางพื้นฐานการเชื่อมต่อ CPU- Memory- I/O



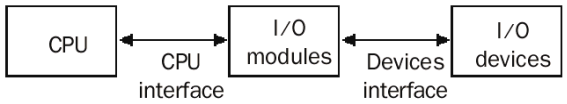
- เส้นทางที่เชื่อมต่อระหว่างซีพียูและไอโอโมดูลเพื่อให้ซีพียูใช้คำสั่งควบคุมอินพุต/เอาต์พุตด้วยโปรแกรม และส่งสัญญาณ **interrupt** ไปยังซีพียู
- ส่วนการเชื่อมต่อระหว่างไอโอโมดูลและหน่วยความจำสำหรับการเข้าถึงหน่วยความจำโดยตรง

สถาปัตยกรรมซีพียู-หน่วยความจำ-อินพุต/เอาต์พุต

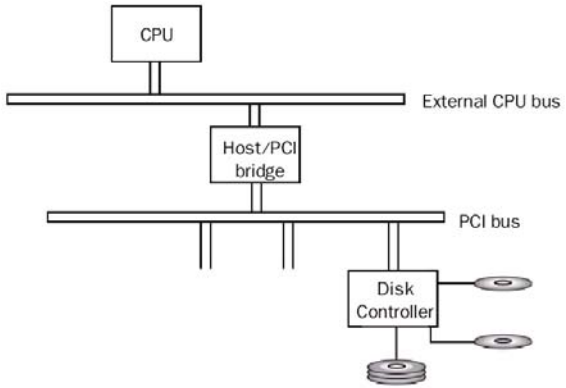
- เส้นทางที่เชื่อมต่อโดยตรงระหว่างองค์ประกอบนี้ ในความจริงแล้วโครงสร้างการเชื่อมต่อภายในพีซีอาจจะเป็นการเชื่อมต่อโดยตรง หรืออาจจะเป็นสวิตช์ที่ทำให้เกิดการเชื่อมต่อในเวลาที่ต้องการ ก็ได้
- ความแตกต่างในการเชื่อมต่อนี้เกิดสถาปัตยกรรมระบบที่ต่างกันตามความต้องการของผู้ออกแบบ, เป้าหมายที่ต่างกัน และปรัชญาการออกแบบที่ต่างกัน นั่นเอง
- สถาปัตยกรรมไอโอพื้นฐาน 2 แบบที่นำมาใช้คือ “สถาปัตยกรรมบัส” (bus architecture) และ “สถาปัตยกรรมช่องทางข้อมูล” (channel architecture)

อินพุต/เอาต์พุตโมดูล (I/O Modules)

- ไอโอโมดูลเป็นอินเทอร์เฟซระหว่างซีพียูกับดีไวซ์
- ไอโอโมดูลมีฟังก์ชันที่สำคัญ 2 ส่วน
  - **การอินเทอร์เฟซกับซีพียู** : ไอโอโมดูลจะรับคำสั่งไอโอจากซีพียู, ถ่ายโอนข้อมูลระหว่างโมดูลกับซีพียูหรือหน่วยความจำ และส่งอินเทอร์รัพต์และข้อมูลสถานะไปยังซีพียู
  - **การอินเทอร์เฟซกับดีไวซ์** : ไอโอโมดูลจะมีหน้าที่ควบคุมการทำงานของดีไวซ์ เช่น การย้ายหัวอ่านในดิสก์ และการกรอเทปเป็นต้น



อินเทอร์เฟซของไอโอโมดูล



การอินเทอร์เฟซของไอโอโมดูลของดิสก์คอนโทรลเลอร์กับดิสก์

อินพุต/เอาต์พุตโมดูล (I/O Modules)

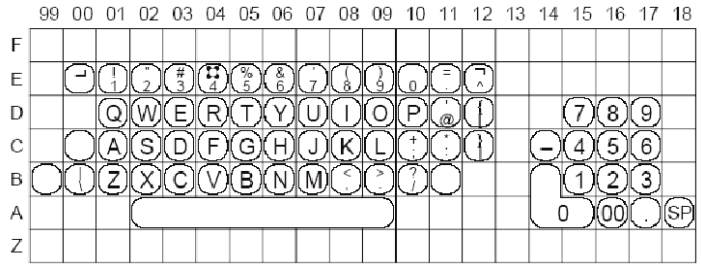
- ฟังก์ชันของไอโอโมดูล
  - รับแมสเสจที่เป็นแอ็ดเดรสเข้ามาแล้วรับคำสั่งจากซีพียูเพื่อทำงานกับดิสก์ที่กำหนด
  - เตรียมบัพเฟอร์ไว้เพื่อพักข้อมูล จนกว่าจะถ่ายโอนมายังดิสก์ได้
  - ตรียมรีจิสเตอร์ที่จำเป็น และสัญญาณควบคุมเพื่อการถ่ายโอนกับหน่วยความจำโดยตรง
  - ควบคุมดิสก์ไดรฟ์ โดยการย้ายหัวอ่านไปยังตำแหน่งที่ต้องการเขียน
  - ก็อปปี้ข้อมูลจากบัพเฟอร์ไปยังดิสก์
  - มีความสามารถในการอินเทอร์รัพต์ เพื่อใช้ในการแจ้งข่าวให้ซีพียูทราบเมื่อการถ่ายโอนข้อมูลเสร็จเรียบร้อย

อินพุต/เอาต์พุตโมดูล (I/O Modules)

- ไอโอโมดูลต้องปลดภาระงานของไอโอออกจากซีพียูเพื่อแยกโมดูลที่ออกแบบสำหรับการถ่ายโอนข้อมูลและควบคุมดีไวซ์ของไอโอออกมา
- ไอโอโมดูลที่เป็นหน่วยควบคุมที่ใช้ในการควบคุมดีไวซ์เรียกว่า “ดีไวซ์คอนโทรลเลอร์” (Device controller) บ่อยครั้งที่เรียกตามชื่อดีไวซ์
- การมีดีไวซ์คอนโทรลเลอร์ทำให้ซีพียูใช้คำสั่งเพื่อควบคุมโอเปอเรชันที่ซับซ้อนมากขึ้น ดีไวซ์หลายตัวที่อยู่ในกลุ่มประเภทเดียวกันสามารถใช้คอนโทรลเลอร์เพียงตัวเดียว ทำให้ลดเรื่องยุ่งยากในการเขียนโปรแกรมให้ซีพียูทำงานกับ ไอโอดีไวซ์ทั้งหมด
- โอโมดูลยังสนับสนุนการเข้าถึงหน่วยความจำโดยตรง ทำให้ซีพียูมีเวลาว่างจากการทำงานอื่น

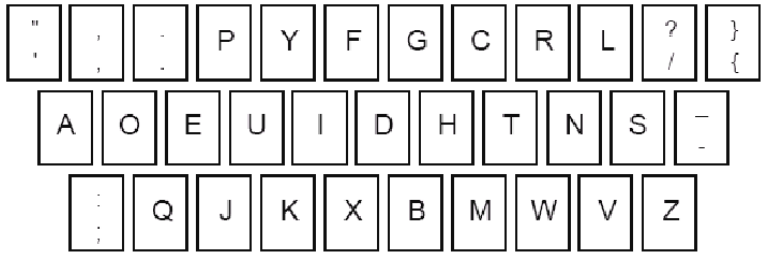
อินพุตดีไวซ์ (Input Devices)

- **คีย์บอร์ด (Keyboard)** : อุปกรณ์อินพุตพื้นฐานที่ต้องมีในคอมพิวเตอร์ทุกเครื่องจะรับข้อมูลจากการกดคีย์ แล้วทำการเปลี่ยนเป็นรหัสเพื่อส่งต่อไปให้กับคอมพิวเตอร์



Layout ของแป้นพิมพ์ตามมาตรฐาน ECMA-23

อินพุตดีไวซ์ (Input Devices)



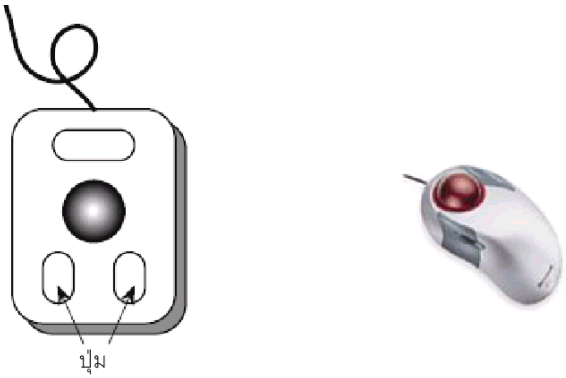
แป้นพิมพ์ Dvorak

อินพุตดีไวซ์ (Input Devices)

- **เมาส์และแทร็กบอลล์ (Mouse & Trackball)** : อุปกรณ์อินพุตเลือกรายการหรือคำสั่งด้วยภาพ หรือไอคอน (icon)

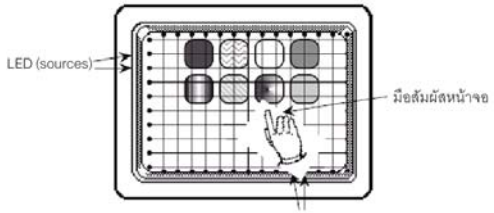
อินพุตดีไวซ์ (Input Devices)

แทร็กบอลล์ (Trackball)



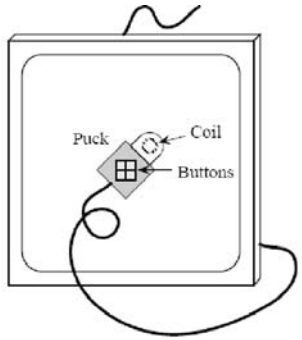
อินพุตดีไวซ์ (Input Devices)

- **ปากกาและจอสัมผัส (LightPens & Touchscreens)** : อุปกรณ์อินพุตที่มีแสงอิเล็กตรอนจะกระตุ้นสารเรืองแสง (phosphor) ที่เคลือบอยู่ด้านหลังผิวจอภาพ สารเรืองแสงนี้จะสว่างและดับกลับไปสภาวะปกติ



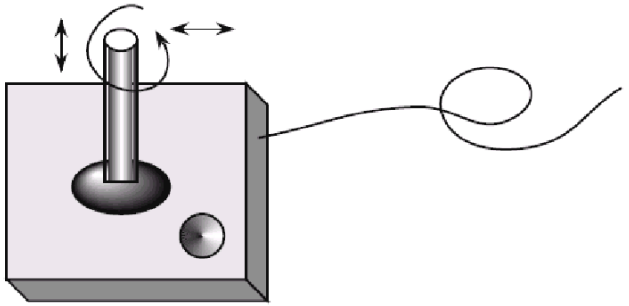
อินพุตดีไวซ์ (Input Devices)

- **Bit Pad หรือ Digitizing Tablet** : เป็นอินพุตดีไวซ์ที่ประกอบด้วยพื้นผิวเรียบ และ stylus หรือ puck แผ่นเรียบจะมีสายที่เป็นตาข่าย 2 มิติที่สามารถตรวจจับสัญญาณที่สร้างจาก puck ที่เคลื่อนที่บนแผ่นเรียบนั้น แผ่นเรียบจะส่งตำแหน่ง X-Y และสถานะของปุ่มบน puck



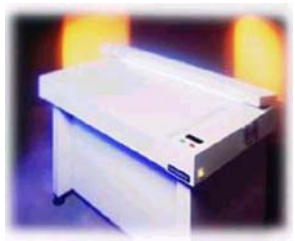
อินพุตดีไวซ์ (Input Devices)

- **จอยสติ๊ก (Joystick)** : บอกตำแหน่งแนวตั้งและแนวนอนด้วยระยะก้านที่ยื่นออกมาจากฐานของจอยสติ๊ก ส่วนมากนิยมนำมาใช้งานกับวิดีโอเกมส์ และเพื่อกำหนดตำแหน่งในระบบกราฟิก



อินพุตดีไวซ์ (Input Devices)

- **สแกนเนอร์ (Scanner)** : อุปกรณ์ซึ่งจับภาพและเปลี่ยนแปลงภาพจากรูปแบบของแอนาลอกเป็นดิจิตอล



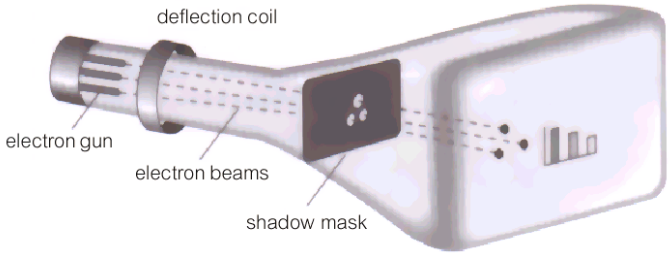
Scanner แบบเลื่อนกระดาษ, แบบแท่นนอน และแบบมือถือ

เอาต์พุตดีไวซ์ (Output Devices)

จอภาพ CRT (Cathode Ray Tube) : เอาต์พุตดีไวซ์ที่แสดงผลลัพธ์จากการประมวลของโปรเซสเซอร์ให้ผู้ใช้เห็น มีปืนอิเล็คตรอน (Electron gun) 3 กระบอกอยู่ภายในท่อจะยิงลำแสงอิเล็คตรอนจากด้านหลังเข้ามา ปืนแต่ละกระบอกจะยิงแต่ละสีของสีหลัก แดง (Red), เขียว (Green) และน้ำเงิน (Blue) ที่เรียกว่า RGB ลำแสงที่ยิงมานั้นจะกระทบสารเคลือบ (Phosphor) ที่เคลือบอยู่ที่จอภาพ ทำให้เกิดการเรืองแสงแต่ละจุดที่เรียกว่าพิกเซล (pixel) ทำให้แต่ละจุดเกิดเป็นสีแดง, เขียว หรือน้ำเงินบนจอภาพ

เอาต์พุตดีไวซ์ (Output Devices)

Diagram ของ CRT

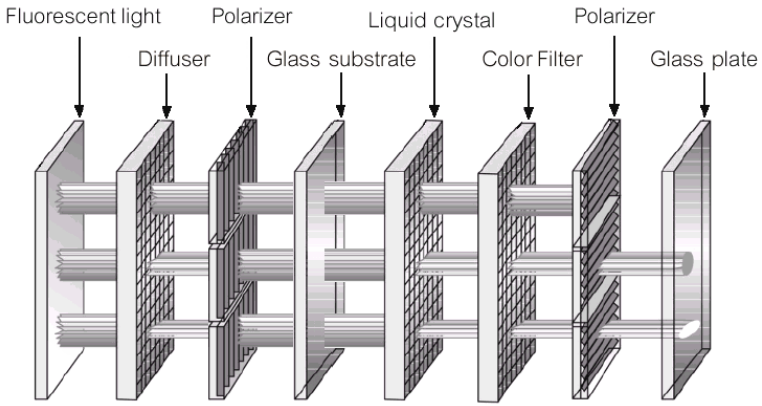


เอาต์พุตดีไวซ์ (Output Devices)

จอ LCD (Liquid Crystal Display) : จอภาพที่สร้างด้วยผลึกเหลว ซึ่งเป็นสสารที่แทบจะเรียกได้ว่าโปร่งใส มีคุณสมบัติก้ำกึ่งระหว่างของแข็ง และของเหลว ที่อุณหภูมิห้องผลึกเหลวจะอยู่ในสถานะของเหลว แต่เมื่อมีแสงผ่านมา จะเกิดการจัดเรียงโมเลกุลใหม่ที่มีคุณสมบัติเป็นของแข็งแทน เมื่อแสงผ่านไปเรียบร้อยแล้วจะกลับมามีคุณสมบัติเป็นของเหลวเหมือนเดิม

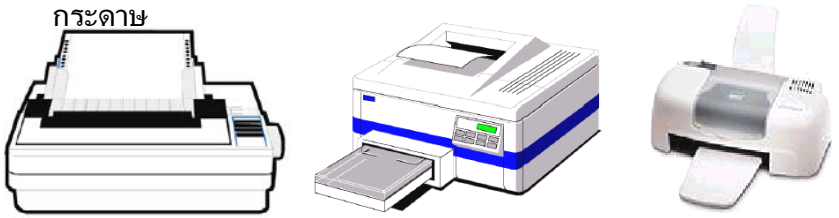
เอาต์พุตดีไวซ์ (Output Devices)

โครงสร้างของจอภาพ LCD



อินพุตดีไวซ์ (Input Devices)

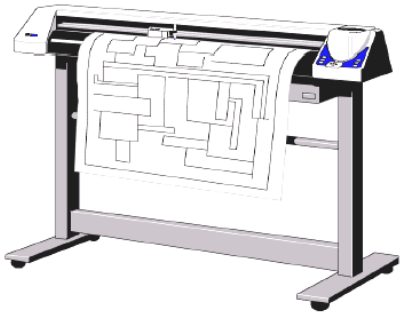
เครื่องพิมพ์ (Printer) : อุปกรณ์ที่เชื่อมต่อเข้ากับคอมพิวเตอร์เพื่อทำหน้าที่ในการแปลผลลัพธ์ที่ได้จากการประมวลผลของเครื่องคอมพิวเตอร์ให้อยู่ในรูปของอักขระหรือรูปภาพที่จะไปปรากฏอยู่บนกระดาษ



เครื่องพิมพ์ Dot Matrix (ดอตแมทริกซ์), Layser (เลเซอร์) และ Ink Jet (อิงค์เจ็ต)

เอาต์พุตดีไวซ์ (Output Devices)

พล็อตเตอร์ (Plotter) : เครื่องพิมพ์ชนิดที่ใช้ปากกาในการเขียนข้อมูลต่างๆ ลงบนกระดาษขนาดใหญ่ (A0 หรือ A1) เหมาะสำหรับงานเกี่ยวกับการเขียนแบบทางวิศวกรรม และงานตกแต่งภายในของสถาปนิก



Other Output Device

- **การระบุตำแหน่งด้วยระบบ GPS** (Global Positioning System)
- **การจดจำลายนิ้วมือ** (Fingerprint Recognition)
- **การจดจำเสียง** (Voice Recognition)