



Memory



Agenda

- Random-Access Memory (Read-Write Memory)
- Static and Dynamic Memory
- Memory System
- Direct Memory Access
- Paging and Other Memory Extension Techniques



Random-Access Memory (1)

- ในการทำงานของ Microprocessor พบว่าต้องมี การอ่านและเขียนข้อมูลลงหน่วยความจำ
- ข้อมูลที่อยู่ใน หน่วยความจำ จะเป็นคำสั่ง, ข้อมูลที่ใช้ในการประมวลผล หรือ ข้อมูลหลังจากการประมวลผล
- บางระบบ อาจใช้เป็น Sequence Access Memory



Random-Access Memory (2)

- ในการอ่านข้อมูลใน Sequence Access Memory ถ้าต้องการอ่านข้อมูลในตำแหน่งที่ N จะต้องทำการอ่านข้อมูลตั้งแต่ตำแหน่งแรก จนถึงตำแหน่ง N
- ตัวอย่างเช่น Magnetic Tape
- ส่วนใหญ่ใช้เก็บข้อมูลที่มีปริมาณมาก ๆ และไม่ต้องการความเร็วในการเข้าถึงข้อมูล



Random-Access Memory (3)

- สิ่งที่สำคัญในเรื่องการทำงานของ ระบบ หน่วยความจำ คือ
 - Memory Access Time
 - Memory Cycle Time
- Memory Access Time
 - Read Access Time คือ ระยะเวลาที่ใช้ในการนำข้อมูลสู่ Data Bus จากหน่วยความจำ หลังจากที่ยกตำแหน่งที่ต้องการ



Random-Access Memory (4)

- Write Access Time คือระยะเวลาที่ใช้ในการ เขียน ข้อมูล ลงในหน่วยความจำเมื่อยกตำแหน่งที่ต้องการ
- เวลาส่วนใหญ่จะเป็น nanosecond เช่น 50 ns (0.05 microsecond)
- Memory Cycle Time คือระยะเวลาที่สั้นที่สุด ระหว่างการทำงาน 2 คำสั่งของ Microprocessor ที่ มีการติดต่อกับ หน่วยความจำ



Random-Access Memory (5)

- ประเภทของหน่วยความจำ
 - Volatile Memory
 - Nonvolatile Memory
- Microprocessor ไม่สามารถทำงานได้ ถ้าขาดคำสั่ง ในตอนที่ทำการจ่ายกระแสไฟฟ้าให้ Microprocessor เริ่มทำงาน ต้องมีโปรแกรมในส่วน ของ Bootstrap หรือ Boot Program เก็บไว้ใน หน่วยความจำที่เป็น Nonvolatile Memory



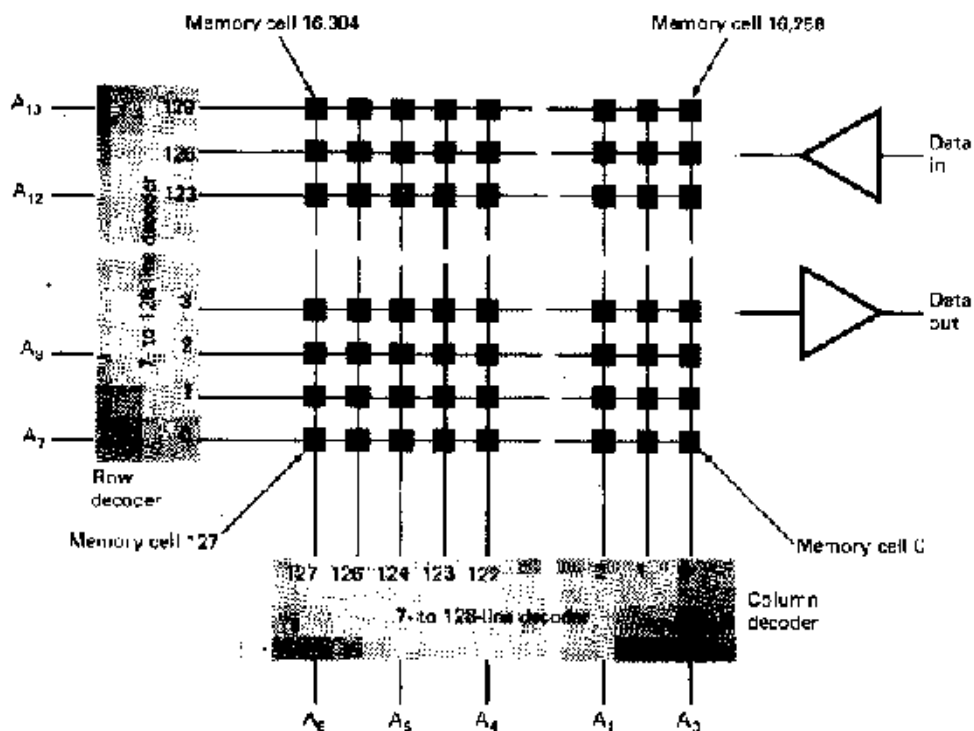
Random-Access Memory (6)

- ตัวอย่างของ Nonvolatile Memory คือ ROM, Magnetic Tape, Disk
- ตัวอย่างของ Volatile Memory คือ Semiconductor Read - Write Memory

Static and Dynamic Memory (1)

- Static Memory เป็นหน่วยความจำที่ง่ายต่อการใช้งาน ใช้สำหรับระบบหน่วยความจำขนาดเล็ก
- Dynamic Memory เป็นหน่วยความจำที่มีราคาถูก แต่ยากต่อการใช้งานจะต้องมี Hardware พิเศษ เพื่อช่วยในการ Refresh หน่วยความจำ

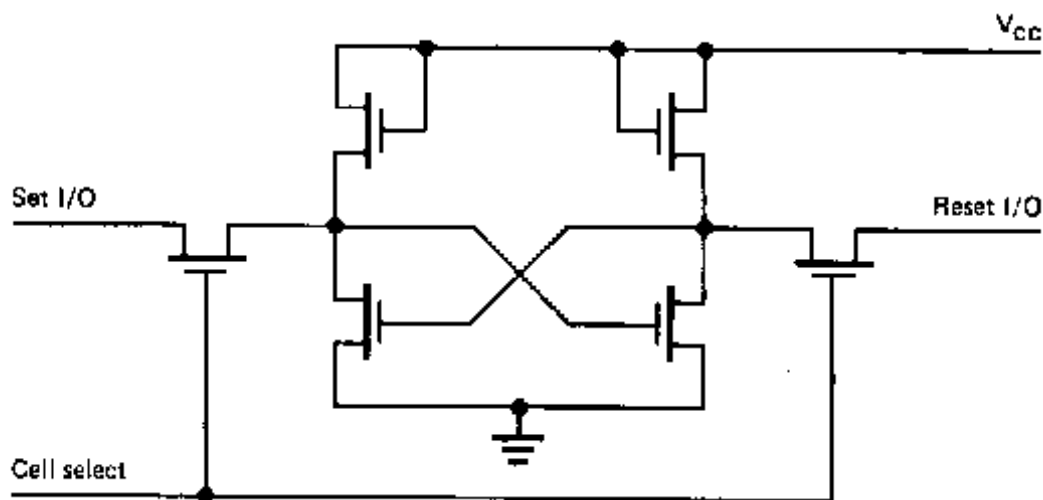
Static and Dynamic Memory (2)



Static and Dynamic Memory (3)

- แต่ละ Cell จะมีการทำงานเหมือนกับ Flip Flop 1 ตัว
- การอ้างตำแหน่งจะอยู่ในพื้นฐานของ Row and Column เพื่อใช้ในการอ้างตำแหน่งของ Cell
- จากตัวอย่าง
 - มีทั้งหมด 16384 Cell
 - ใช้ Address 14 เส้นในการ Decode
 - 7 Bits Row และ 7 Bits Column ($128 * 128$)

Static and Dynamic Memory (4)

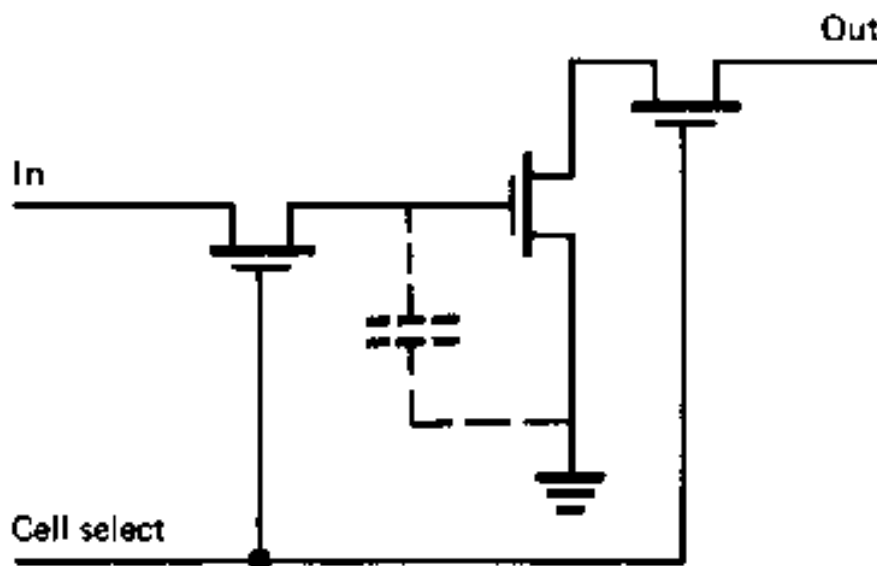


- Static Memory

Static and Dynamic Memory (6)

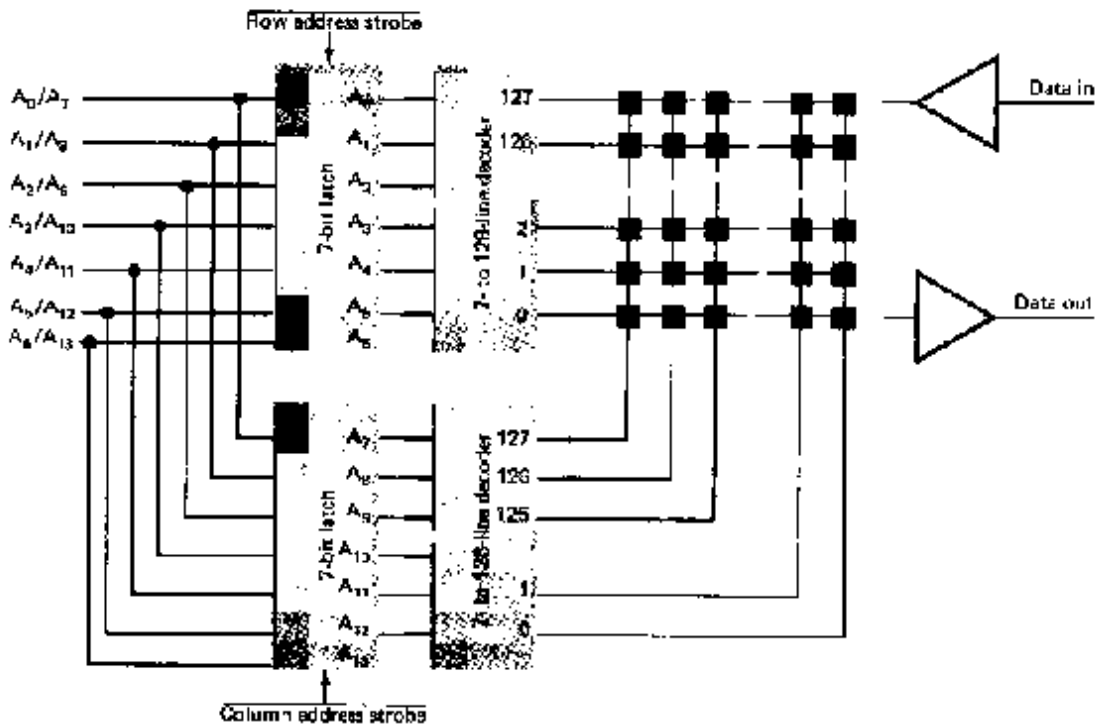
- Dynamic Memory Cell โครงสร้างจะไม่เหมือน Flip Flop แต่จะใช้การทำงานของ Capacitor ในการเก็บข้อมูล
- หลังจากที่ Capacitor เก็บสภาวะ Logic 0 หรือ 1 เรียบร้อยแล้ว จะเก็บได้เพียงไม่กี่ millisecond หลังจากนั้นจะต้องมีการเขียนซ้ำใหม่
- ขบวนการนี้เรียกว่าการ Refreshing ข้อมูล

Static and Dynamic Memory (5)

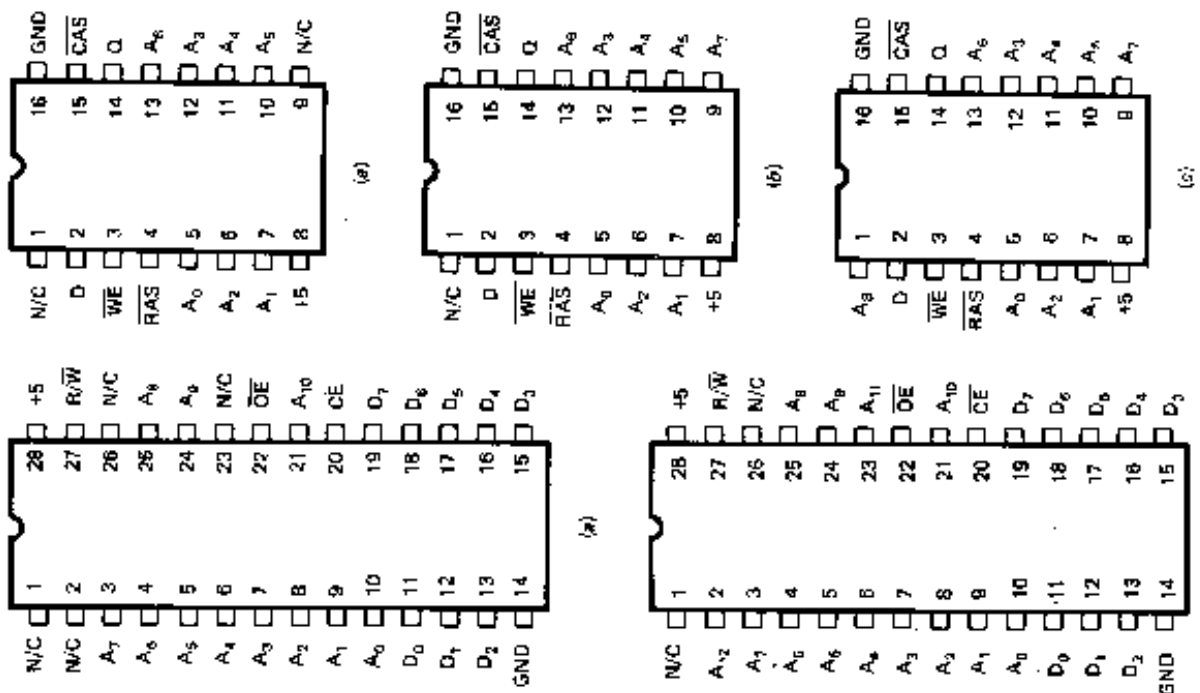


- Dynamic Memory

Static and Dynamic Memory (7)



Static and Dynamic Memory (8)





Static and Dynamic Memory (9)

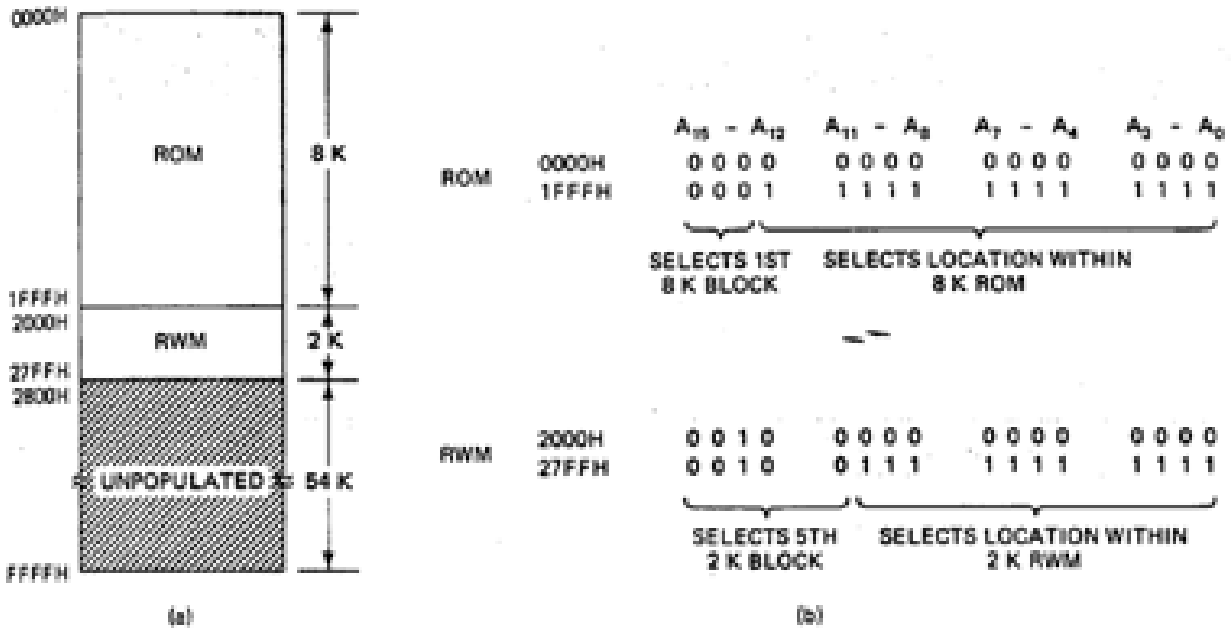
- หน่วยความจำชนิดอื่น ๆ
 - PROM
 - EPORM
 - EEPROM



Memory System (1)

- การออกแบบระบบหน่วยความจำเพื่อเชื่อมต่อกับ CPU
 - เขียน Memory Map
 - เขียน Address Decode
 - เลือกอุปกรณ์หน่วยความจำ และวิธีในการติดต่อกับหน่วยความจำ
 - เขียนวิธีการที่เชื่อมต่ออุปกรณ์หน่วยความจำ กับ CPU
 - เขียนวงจรรวม

Memory System (2)



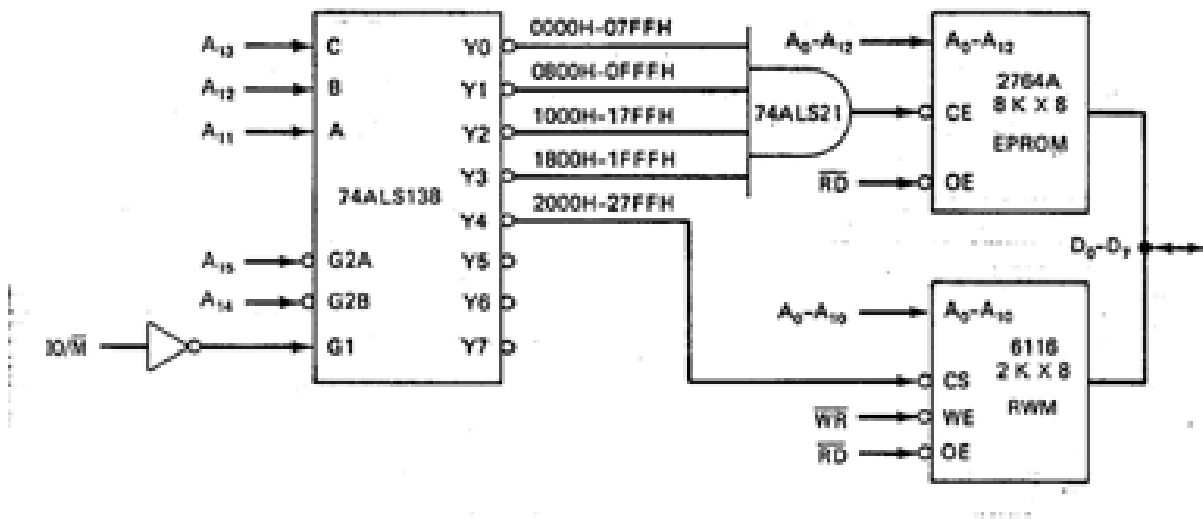
Memory System (3)

128k	64k	32k	16k	จำนวนบิต	16k	32k	64k	128k			
27126	2764	TC5564	2732		6116	2716	2732	TC5564	2764	27128	
Vpp	NC			1						Vcc	
A12				2			R/W	PGM			
	A7			3	24	26	Vcc, Vdd	CE2	NC	A13	
	A5			4	23	25	A8				
	A5			5	22	24	A9				
	A4			6	21	23	R/W	Vpp	A11		
	A3			7	20	22	CE1	CS	OE/Vpp	OE	
	A2			8	19	21	A10				
	A1			9	18	20	CE2	PD/PGM	CE	CE1	CE
	A0			10	17	19	D7				
	D0			11	16	18	D6				
	D1			12	15	17	D5				
	D2			13	14	16	D4				
	GND			14	13	15	D3				

* สำหรับบิต 6116 จะมีขา 20 เป็น OE และขา 18 เป็น CE

ภาพมองจากด้านบน

Memory System (4)



Direct Memory Access (DMA) (1)

- ในกรณีที่ CPU ต้องการอ่านข้อมูลจากอุปกรณ์ภายนอกมาเก็บไว้ในหน่วยความจำ จะทำการอ่านข้อมูลมาเก็บไว้ใน Register ก่อนแล้วจึงนำไปเก็บไว้ในหน่วยความจำ
- ในกรณีที่ข้อมูลที่ต้องการอ่านจาก I/O มีจำนวนมาก ๆ จะเสียเวลาในการอ่านเข้ามาภายใน CPU เท่ากับจำนวนข้อมูลที่ต้องการจะอ่าน
- การทำ DMA จะลดเวลาในการทำขั้นตอนนี้



Direct Memory Access (DMA) (2)

- DMA จะทำหน้าที่ในการจัดการโดยตรงระหว่าง หน่วย ความจำ กับอุปกรณ์ ที่ต้องการติดต่อ
- ตัวที่ทำหน้าที่นี้คือ DMA Controller
- เมื่อต้องการทำ DMA ; CPU จะส่งข้อมูลที่เป็น จำนวน Address ที่เริ่มนำข้อมูลไปเก็บไว้ใน หน่วยความจำ, จำนวนของข้อมูลที่จะทำการ DMA ,อุปกรณ์ที่จะติดต่อโดยตรงกับหน่วยความจำ



Direct Memory Access (DMA) (3)

- CPU ปลดตัวเองออกจากระบบคือให้ Address Bus และ Data Bus ของ CPU เป็น High Impedance
- DMA Controller จะเข้ามาทำการควบคุมการทำงาน ของ อุปกรณ์ต่าง ๆ แทน CPU
- เมื่อทำ DMA เสร็จเรียบร้อยแล้ว ก็จะส่งสัญญาณไปบอก CPU แล้วทำการคืนการควบคุมการทำงาน ให้กับ CPU

Paging and Other Memory Extension Techniques (1)

- ในกรณีที่มีขนาดของหน่วยความจำที่ CPU สามารถอ้างถึงได้ในขนาดหนึ่ง ถ้าต้องการอ้างหน่วยความจำที่สูงกว่าความสามารถของ CPU มีอยู่หลายวิธี
- Paging Techniques
 - แบ่งหน่วยความจำออกเป็น 2 ส่วน คือ Directly Address และ Bank Switched Address
 - มี Hardware ทำหน้าที่เป็น Memory Management Unit (MMU)

Paging and Other Memory Extension Techniques (2)

