Memory

Shao-Yi Chien

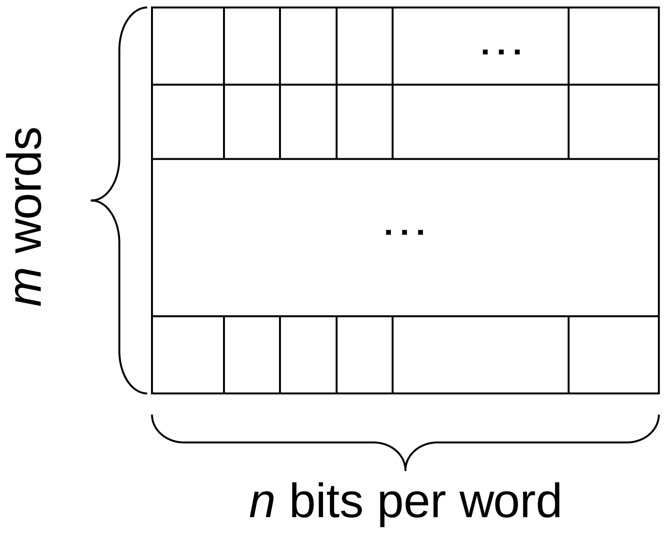
Outline

- Memory write ability and storage permanence
- Common memory types
- Advanced RAM
- Memory hierarchy and cache
- Memory management unit (MMU)
- New memory packaging in SoC

Memory: Basic Concepts

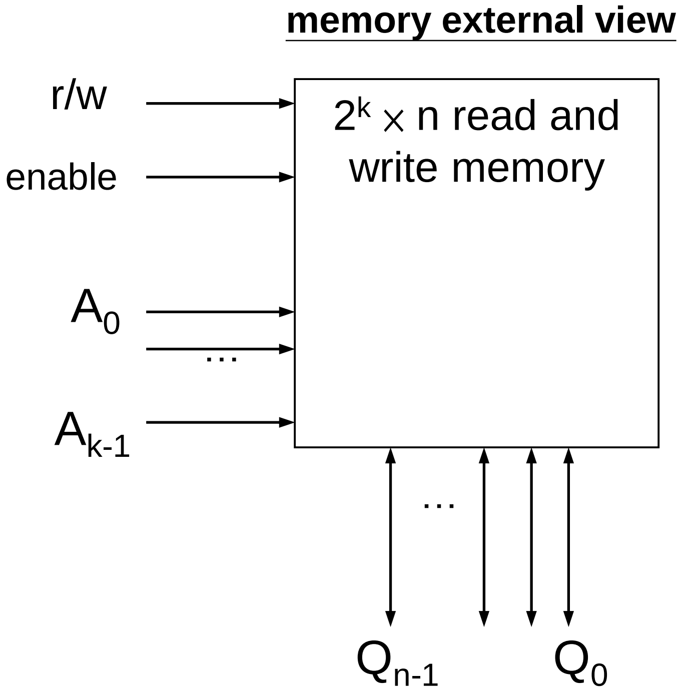
- Stores large number of bits
  - □ *m* x *n*: *m* words of *n* bits each
  - □ k = $\text{Log}_2(m)$ address input signals
  - □ or *m* = 2^k words
  - □ e.g., 4,096 x 8 memory:
    - 32,768 bits
    - 12 address input signals
    - 8 input/output data signals

m × n memory



Memory: Basic Concepts

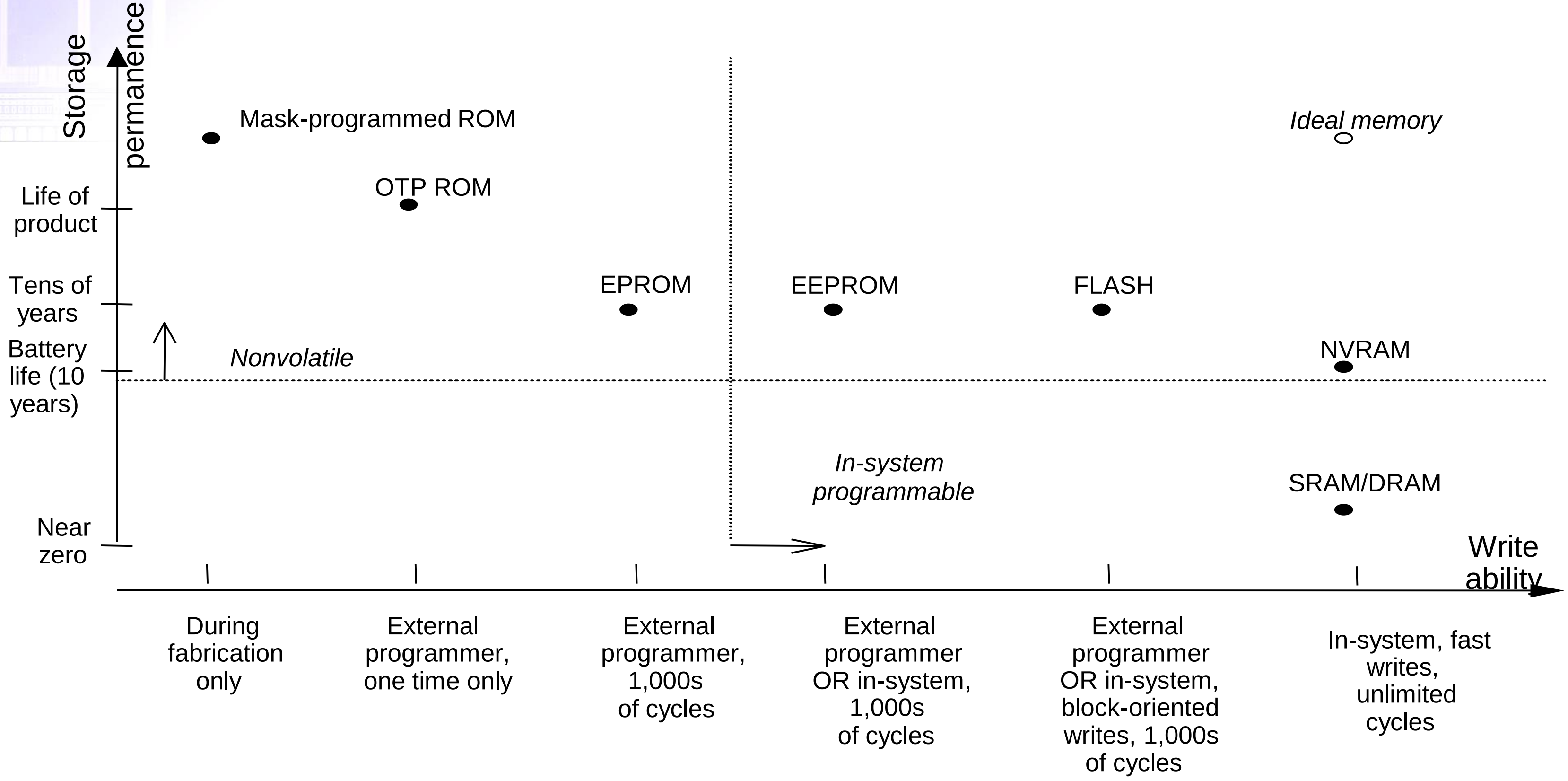
- Memory access
  - □ r/w: selects read or write
  - □ Enable: read or write only when asserted
  - □ Multiport: multiple accesses to different locations simultaneously



Write Ability/ Storage Permanence

- Traditional ROM/RAM distinctions
  - ROM
    - Read only, bits stored without power
  - RAM
    - Read and write, lose stored bits without power
- Traditional distinctions are blurred
  - Advanced ROMs can be written to
    - e.g., EEPROM
  - Advanced RAMs can hold bits without power
    - e.g., NVRAM (Nonvolatile RAM)
  - New types of memory: FeRAM, PCM, MRAM, …
- **Write ability**
  - Manner and speed a memory can be written
- **Storage permanence**
  - Ability of memory to hold stored bits after they are written

Write Ability/ Storage Permanence



Write ability and storage permanence of memories,
showing relative degrees along each axis (not to scale).

Write Ability

- Ranges of write ability
  - □ High end
    - Processor writes to memory simply and quickly
    - e.g., RAM
  - □ Middle range
    - Processor writes to memory, but slower
    - e.g., FLASH, EEPROM
  - □ Lower range
    - Special equipment, “programmer”, must be used to write to memory
    - e.g., EPROM, OTP ROM
  - □ Low end
    - Bits stored only during fabrication
    - e.g., Mask-programmed ROM
- In-system programmable memory
  - □ Can be written to by a processor in the embedded system using the memory → do not need “programmer”
  - □ Memories in high end and middle range of write ability

Storage Permanence

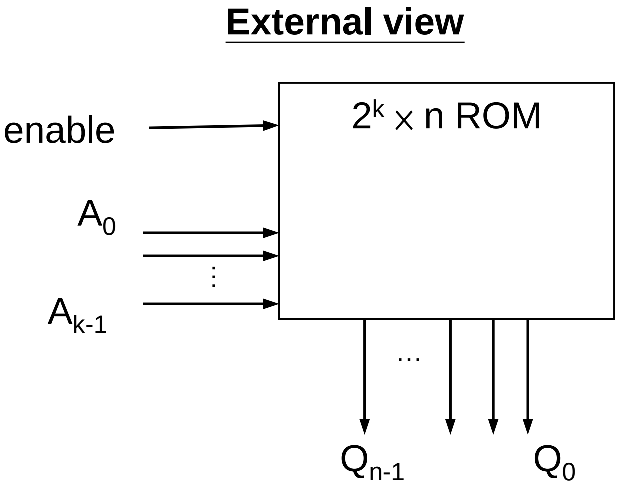
- Range of storage permanence
  - High end

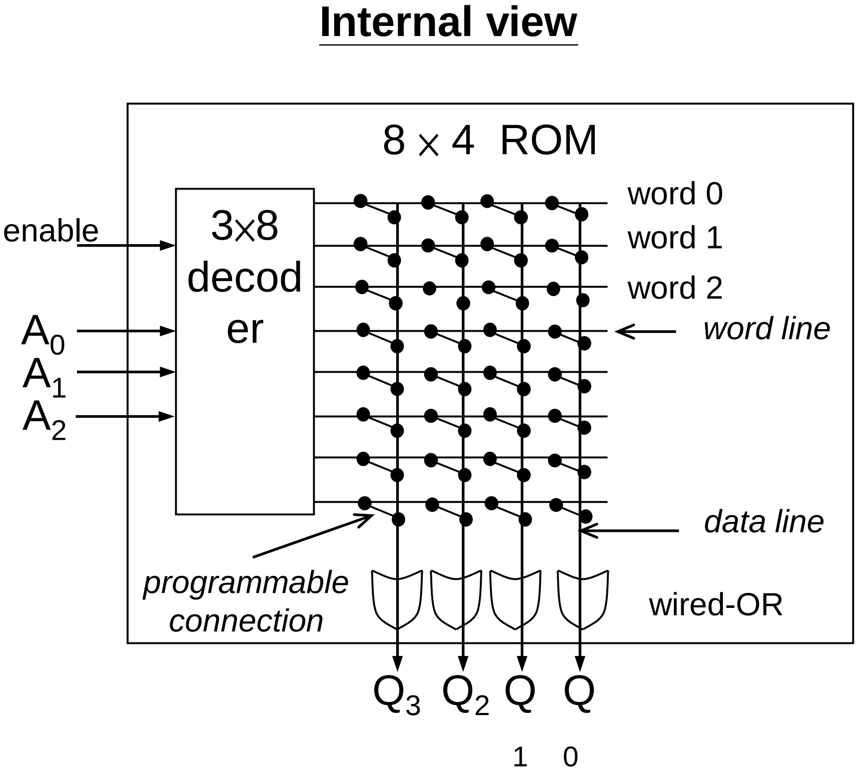
    Essentially never loses bits
    - e.g., mask-programmed ROM
  - Middle range
    - Holds bits days, months, or years after memory’s power source turned off
    - e.g., NVRAM
  - Lower range
    - Holds bits as long as power supplied to memory
    - e.g., SRAM
  - Low end
    - Begins to lose bits almost immediately after written
    - e.g., DRAM
- Nonvolatile memory
  - Holds bits after power is no longer supplied
  - High end and middle range of storage permanence

ROM: “Read-Only” Memory

- Nonvolatile memory
- Can be read from but not written to, by a processor in an embedded system
- Traditionally written to, “programmed,” before inserting to embedded system
- Uses
  - Store software program for general-purpose processor
  - Store constant data needed by system
  - Implement combinational circuits

ROM: “Read-Only” Memory





Mask-Programmed ROM

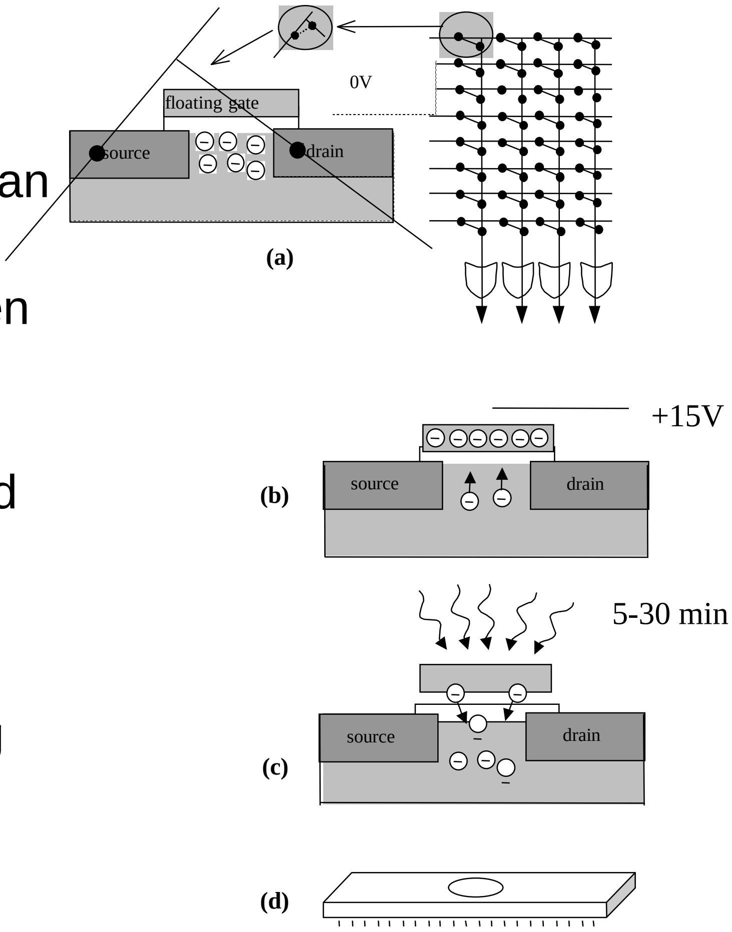
- Connections “programmed” at fabrication
    - Set of masks
- Lowest write ability
    - Only once
- Highest storage permanence
    - Bits never change unless damaged
- Typically used for final design of high-volume systems
    - Spread out NRE cost for a low unit cost

OTP ROM: One-Time Programmable ROM

- Connections “programmed” after manufacture by user
  - Use a machine called ROM programmer
  - Each programmable connection is a fuse
  - ROM programmer blows fuses where connections should not exist
- Very low write ability
  - Typically written only once and requires ROM programmer device
- Very high storage permanence
  - Bits don’t change unless reconnected to programmer and more fuses blown
- Commonly used in final products
  - Cheaper, harder to inadvertently modify

EPROM: Erasable Programmable ROM

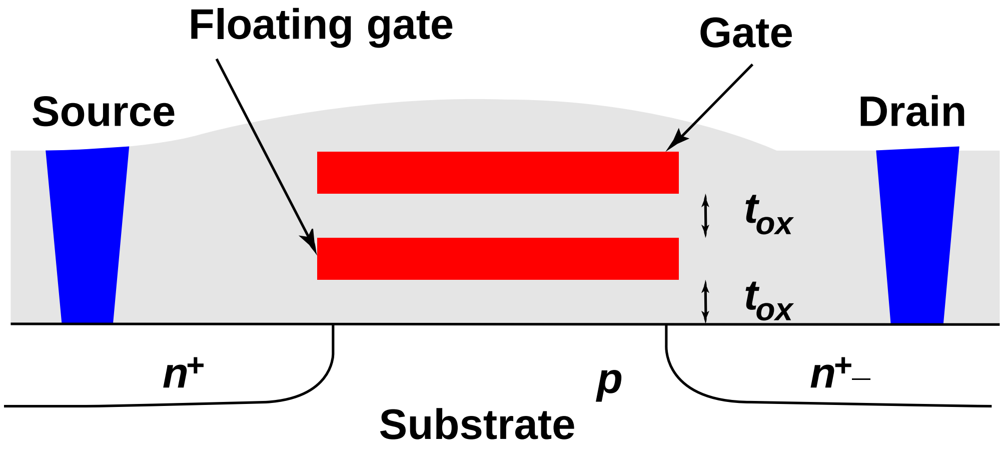
- **Programmable component is a MOS transistor**
  - Transistor has ***floating gate*** surrounded by an insulator
  - **(a)** Negative charges form a channel between source and drain storing a logic 1
  - **(b)** Large positive voltage at gate causes negative charges to move out of channel and get trapped in floating gate storing a logic 0
  - **(c)** (Erase) Shining UV rays on surface of floating-gate causes negative charges to return to channel from floating gate restoring the logic 1
  - **(d)** An EPROM package showing quartz window through which UV light can pass



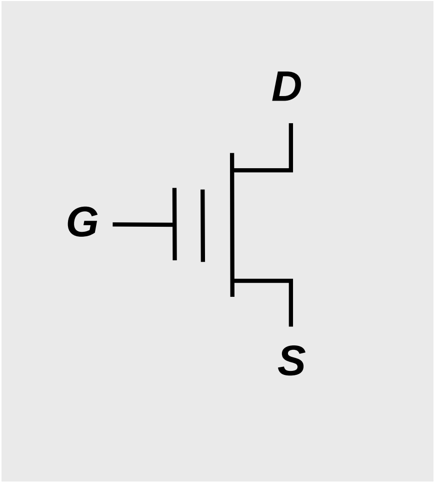
EPROM: Erasable Programmable ROM

- Better write ability
  - Can be erased and reprogrammed thousands of times
- Reduced storage permanence
  - Program lasts about 10 years but is susceptible to radiation and electric noise
- Typically used during design development

Non-Volatile Memories
The Floating-Gate Transistor (FAMOS)



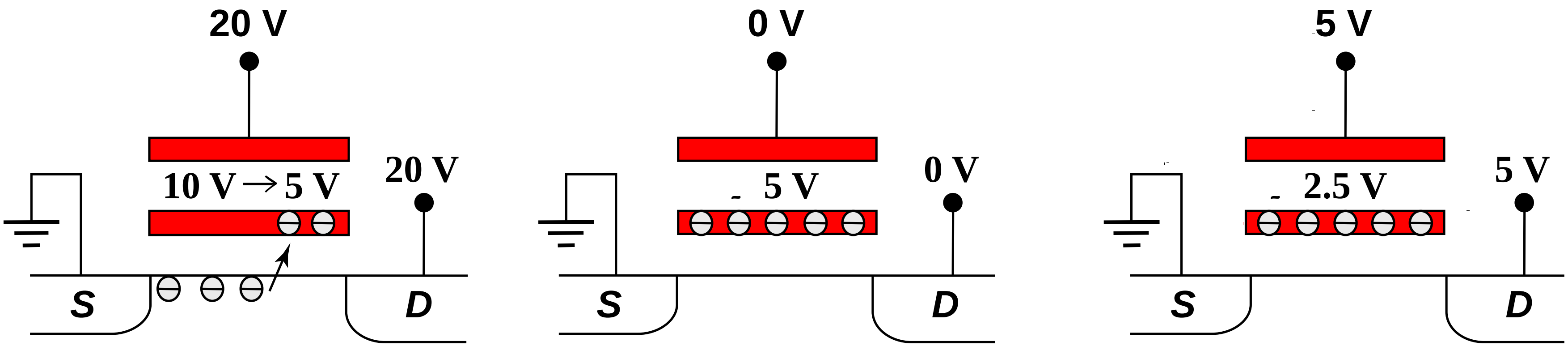
Device cross-section



Schematic symbol

Source: J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 2nd Ed., Prentice Hall, 2003.

Floating-Gate Transistor Programming



Avalanche injection

Removing programming voltage leaves charge trapped

Programming results in higher V_T.

Source: J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 2nd Ed., Prentice Hall, 2003.

EEPROM: Electrically Erasable Programmable ROM

- Programmed and erased electronically
  - Typically by using higher than normal voltage
  - Can program and erase individual words
- Better write ability
  - Can be in-system programmable with built-in circuit to provide higher than normal voltage
    - Built-in memory controller commonly used to hide details from memory user
  - Writes very slowly due to erasing and programming
    - "Busy" pin indicates to processor EEPROM still writing
  - Can be erased and programmed tens of thousands of times
- Similar storage permanence to EPROM (about 10 years)
- Far more convenient than EPROMs, but more expensive

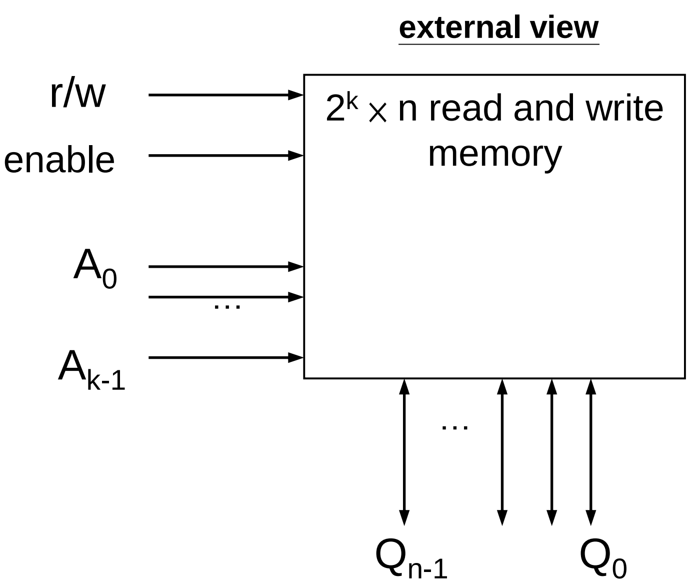
Flash Memory

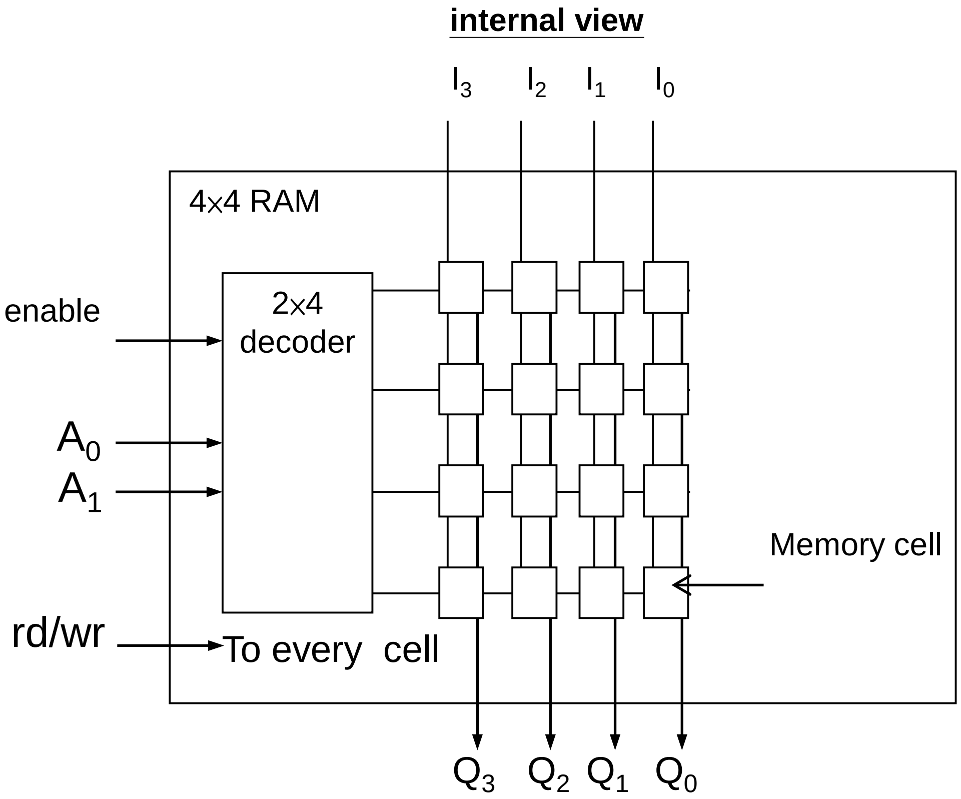
- Extension of EEPROM
  - Same floating gate principle
  - Same write ability and storage permanence
- Fast erase
  - Large blocks of memory erased at once, rather than one word at a time
  - Blocks typically several thousand bytes large
- Writes to single words may be slower
  - Entire block must be read, word updated, then entire block written back
- Used with embedded systems storing large data items in nonvolatile memory
  - e.g., Digital cameras, TV set-top boxes, cell phones

RAM: “Random-Access” Memory

- Typically volatile memory
  - Bits are not held without power supply
- Read and written easily by embedded system during execution
- Internal structure is more complex than ROM
  - A word consists of several memory cells, each storing 1 bit
  - Each input and output data line connects to each cell in its column
  - Rd/wr connected to every cell
  - When row is enabled by decoder, each cell has logic that stores input data bit when rd/wr indicates write or outputs stored bit when rd/wr indicates read

RAM: “Random-Access” Memory

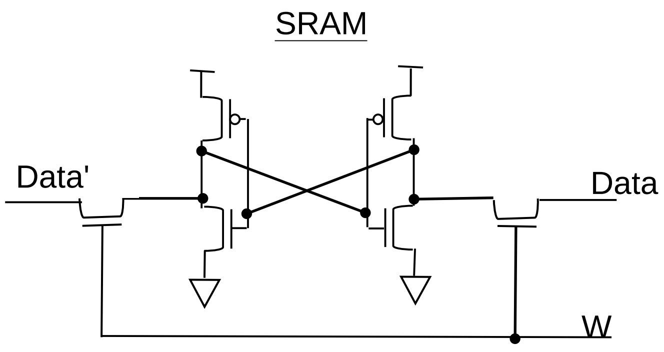


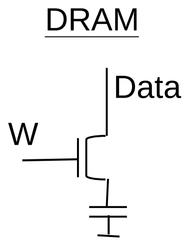


Basic Types of RAM

- SRAM: Static RAM
  - □ Memory cell uses latch to store bit
  - □ Requires 6 transistors
  - □ Holds data as long as power supplied
- DRAM: Dynamic RAM
  - □ Memory cell uses MOS transistor and capacitor to store bit
  - □ More compact than SRAM
  - □ "Refresh" is required due to capacitor leak
    - Word's cells refreshed when read
  - □ Typical refresh rate 15.625 microsec.
  - □ Slower to access than SRAM

memory cell internals





RAM Variations

- PSRAM: Pseudo-static RAM
  - DRAM with built-in memory refresh controller
  - Popular low-cost high-density alternative to SRAM
- NVRAM: Nonvolatile RAM
  - Holds data after external power removed
  - Battery-backed RAM
    - SRAM with its own permanently connected battery
    - Many NVRAMs have batteries that can last for 10 years
    - Writes as fast as reads
    - No limit on number of writes unlike nonvolatile ROM-based memory
  - SRAM with EEPROM or flash
    - Stores complete RAM contents on EEPROM or flash before power turned off

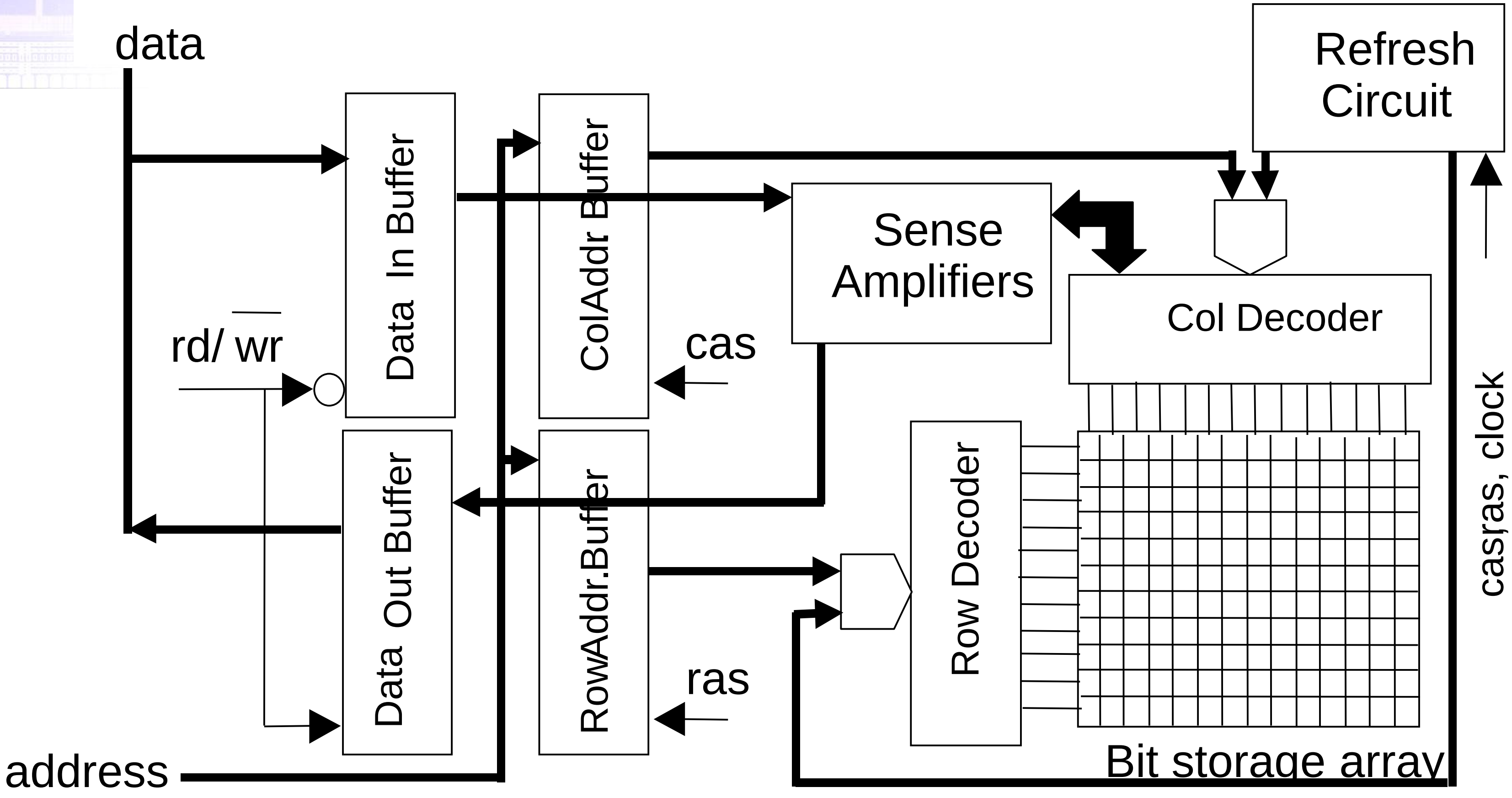
Advanced RAM

- DRAMs are commonly used as main memory in processor based embedded systems
  - High capacity, low cost
- Many variations of DRAMs proposed
  - Need to keep pace with processor speeds
  - FPM DRAM: fast page mode DRAM
  - EDO DRAM: extended data out DRAM
  - SDRAM/ESDRAM: synchronous and enhanced synchronous DRAM
  - RDRAM: rambus DRAM
  - …

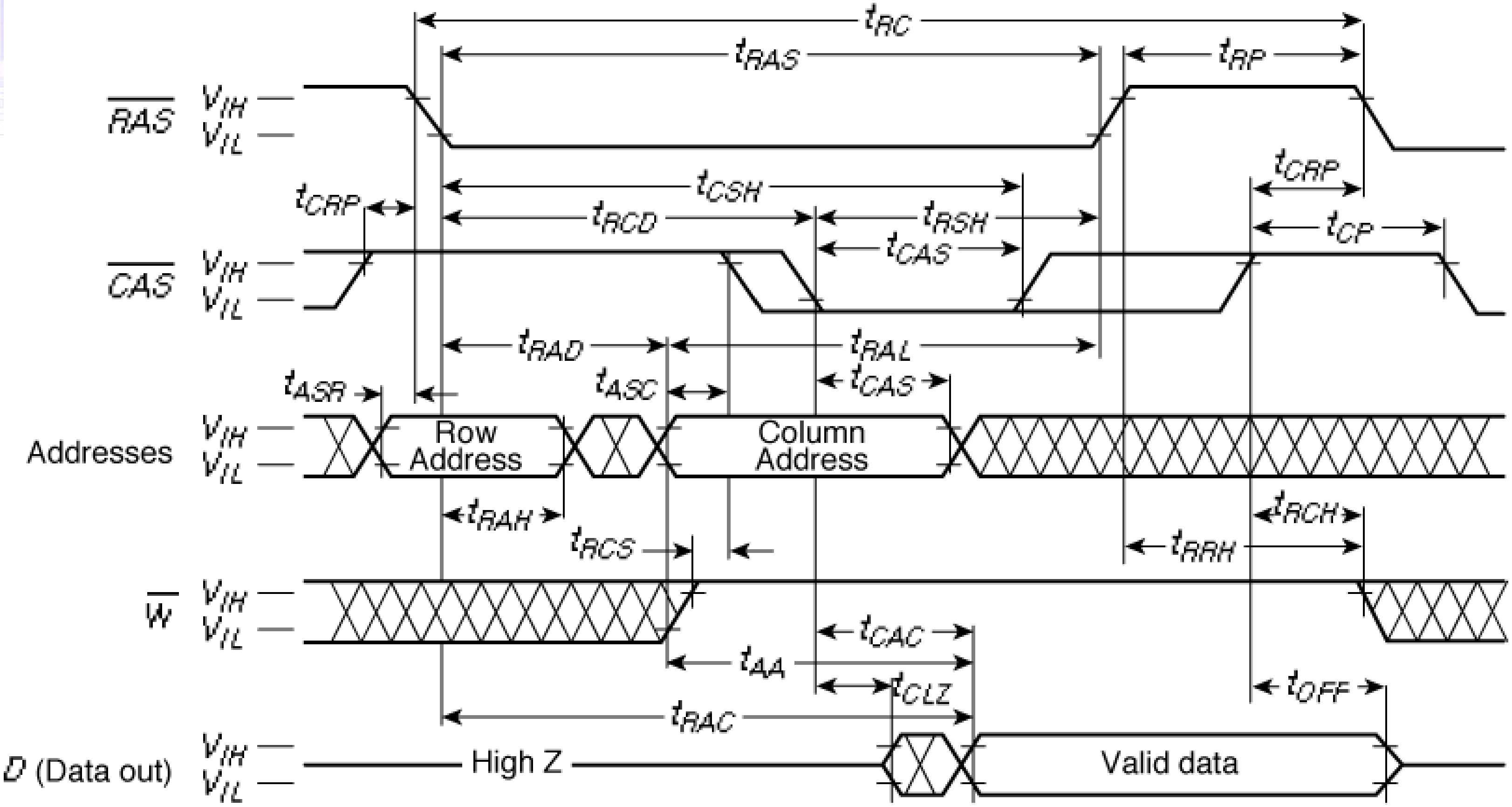
Basic DRAM

- Address bus multiplexed between row and column components
- Row and column addresses are latched in, sequentially, by strobing *ras* and *cas* signals, respectively
- Refresh circuitry can be external or internal to DRAM device
  - Strobes consecutive memory address periodically causing memory content to be refreshed
  - Refresh circuitry disabled during read or write operation

Basic DRAM


data
Refresh Circuit
Data In Buffer
ColAddr Buffer
Sense Amplifiers
Col Decoder
rd/ wr
cas
Data Out Buffer
RowAddr.Buffer
Row Decoder
casras, clock
ras
address
Bit storage array

Basic DRAM

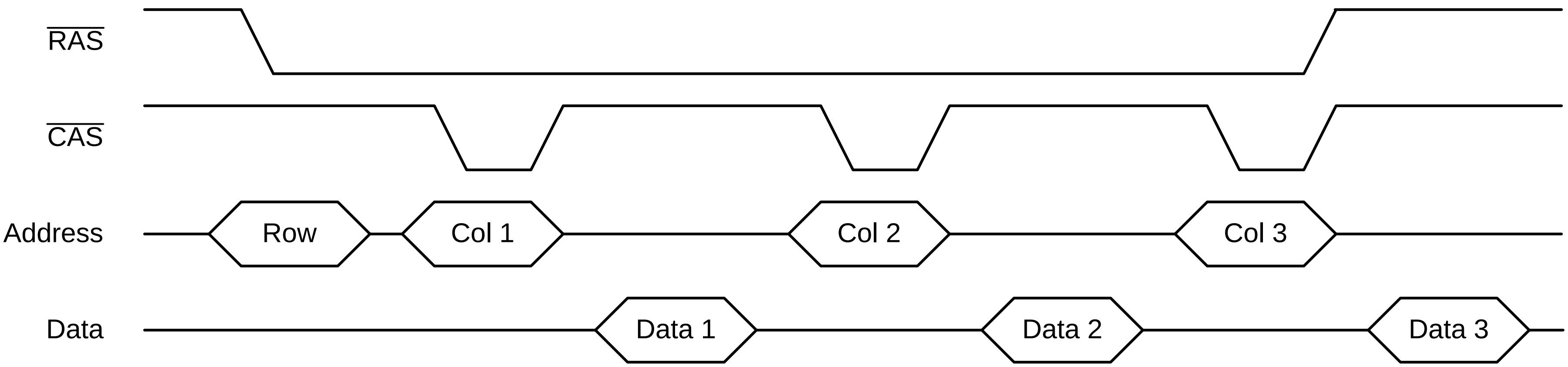


Source: J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 2nd Ed., Prentice Hall, 2003.

Fast Page Mode DRAM (FPM DRAM)

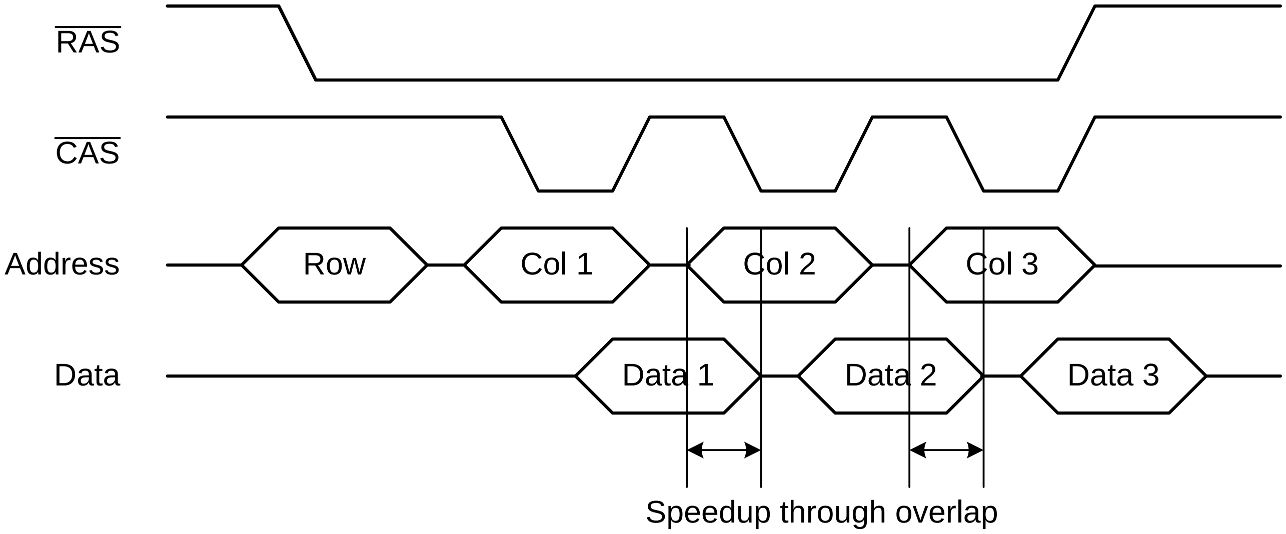
- Each row of memory bit array is viewed as a page
- Page contains multiple words
- Individual words addressed by column address
- Timing diagram on the next slide:
  - Row (page) address sent
  - 3 words read consecutively by sending column address for each
- Extra cycle eliminated on each read/write of words from the same page

Fast Page Mode DRAM (FPM DRAM)



Extended Data Out DRAM (EDO DRAM)

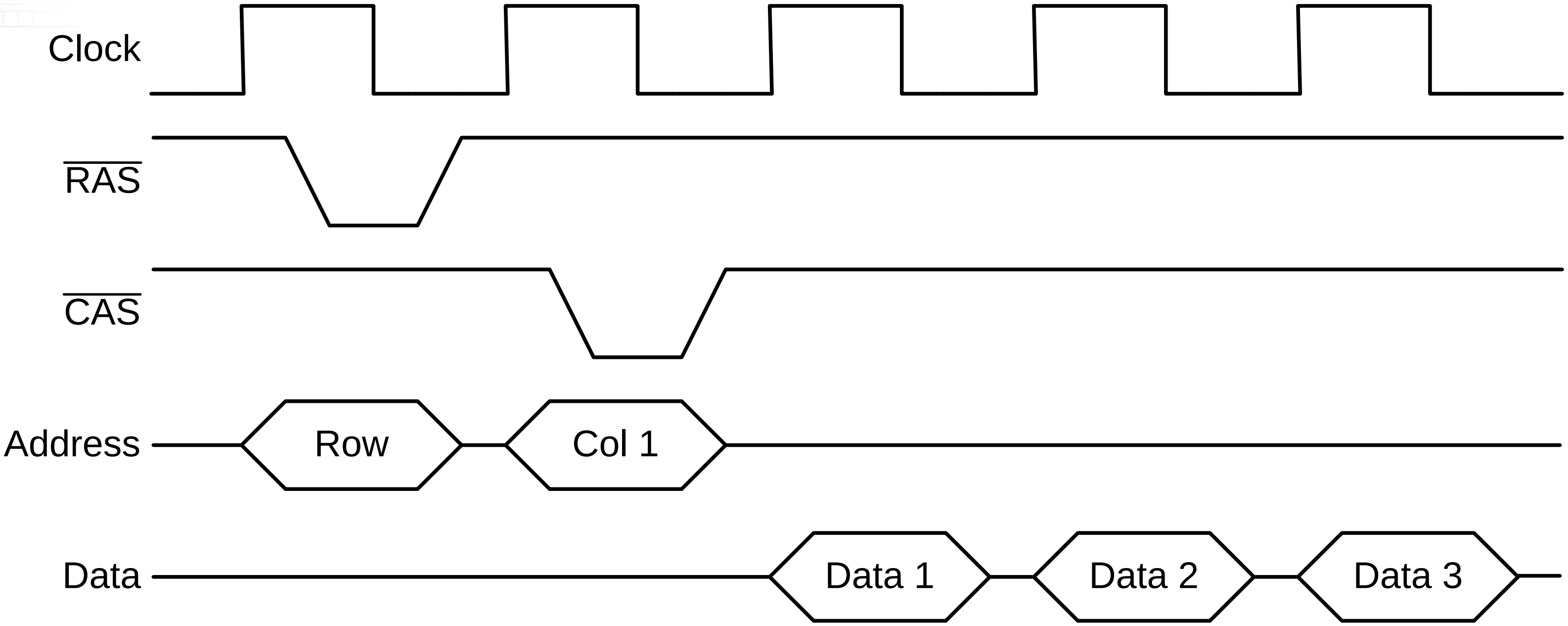
- Improvement of FPM DRAM
- Extra latch before output buffer
  - Allows strobing of *cas* before data read operation completed
- Reduces read/write latency by additional cycle



(S)ynchronous and Enhanced Synchronous (ES) DRAM

- SDRAM latches data on active edge of clock
- Eliminates time to detect *ras/cas* and *rd/wr* signals
- A counter is initialized to column address then incremented on active edge of clock to access consecutive memory locations
- ESDRAM improves SDRAM
  - Added buffers enable overlapping of column addressing
  - Faster clocking and lower read/write latency possible

SDRAM Timing



Varies Access of SDRAM

1. Pre-charge
2. Active
3. CAS

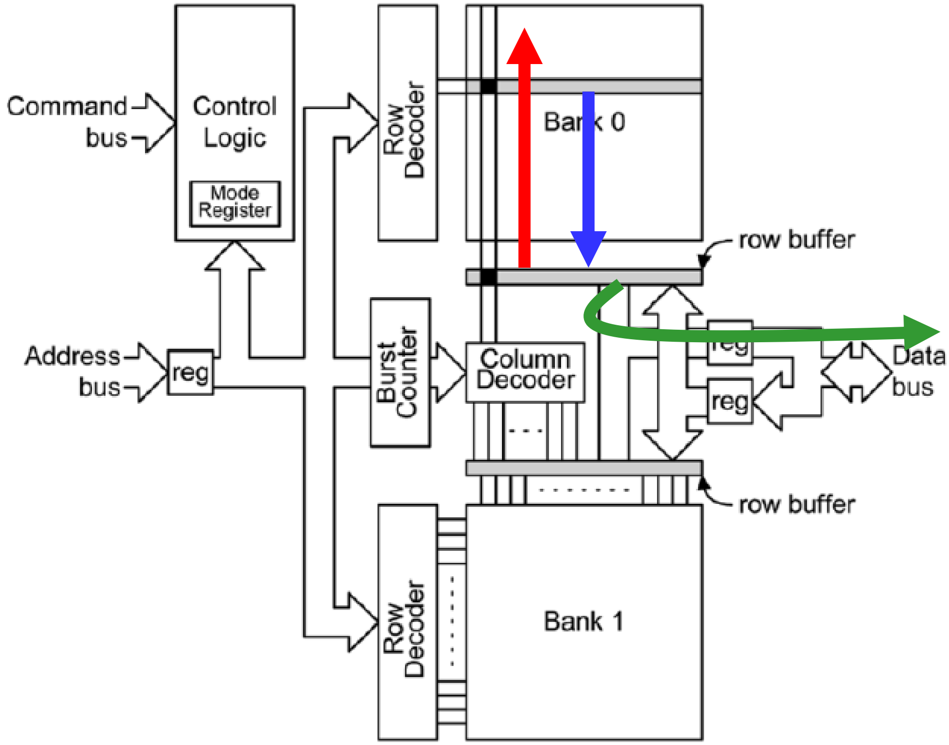
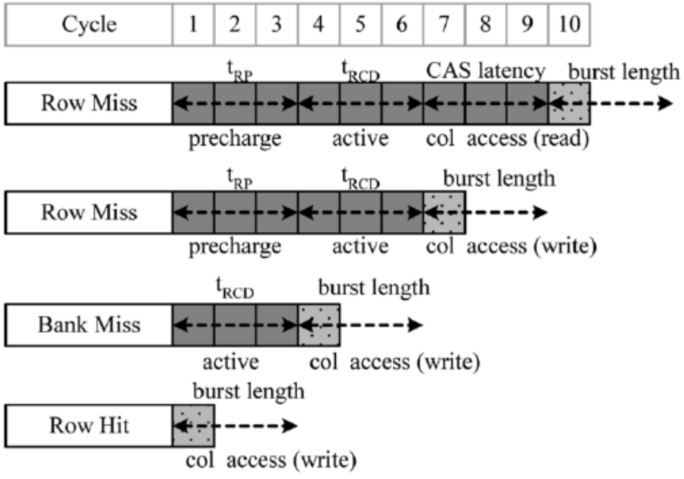


Fig. 1. Simplified architecture of a two-bank SDRAM.

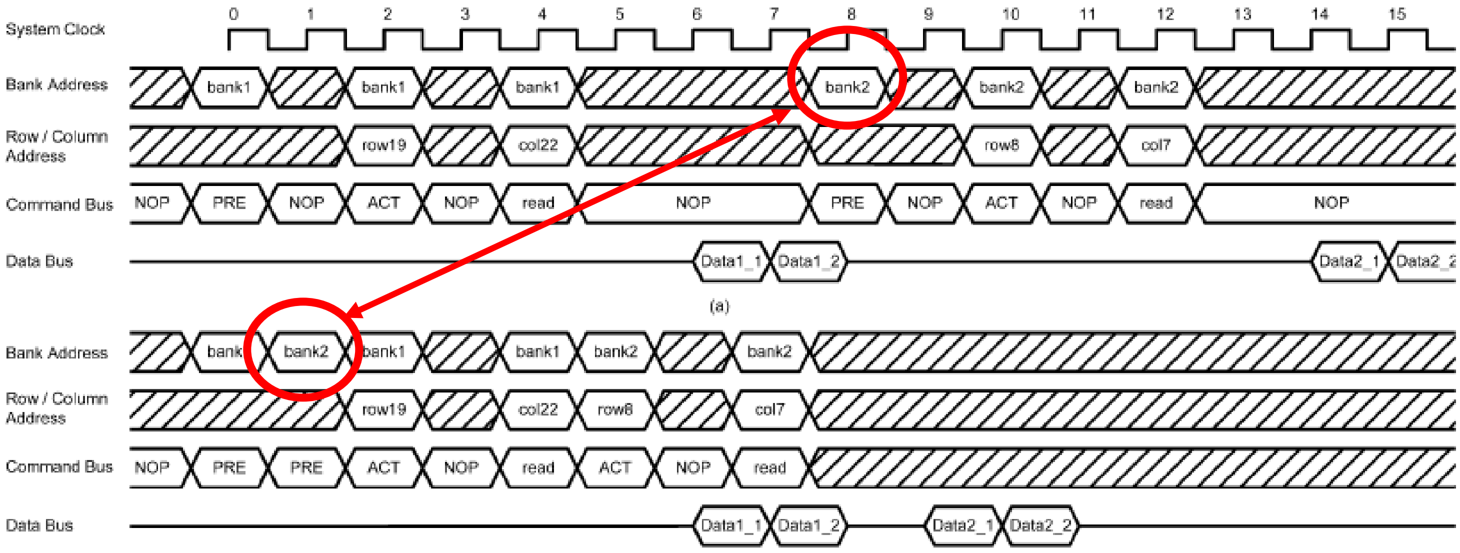


(b)

Source: K.-B. Lee, T.-C. Lin, and C.-W. Jen, "An efficient quality-aware memory controller for multimedia platform SoC," IEEE CSVT, May 2005.

Bank Interleave

- Different banks can operate concurrently


System Clock
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
Bank Address
bank1 bank1 bank1 bank2 bank2 bank2
Row / Column Address
row19 col22 row8 col7
Command Bus
NOP PRE NOP ACT NOP read NOP PRE NOP ACT NOP read NOP
Data Bus
Data1_1 Data1_2 Data2_1 Data2_2
(a)
Bank Address
bank bank2 bank1 bank1 bank2 bank2
Row / Column Address
row19 col22 row8 col7
Command Bus
NOP PRE PRE ACT NOP read ACT NOP read
Data Bus
Data1_1 Data1_2 Data2_1 Data2_2
(b)

Pre-charge bank2

Fig. 5. Two row-miss accesses (in different banks) processed by (a) MIS-I and (b) MIS-II.

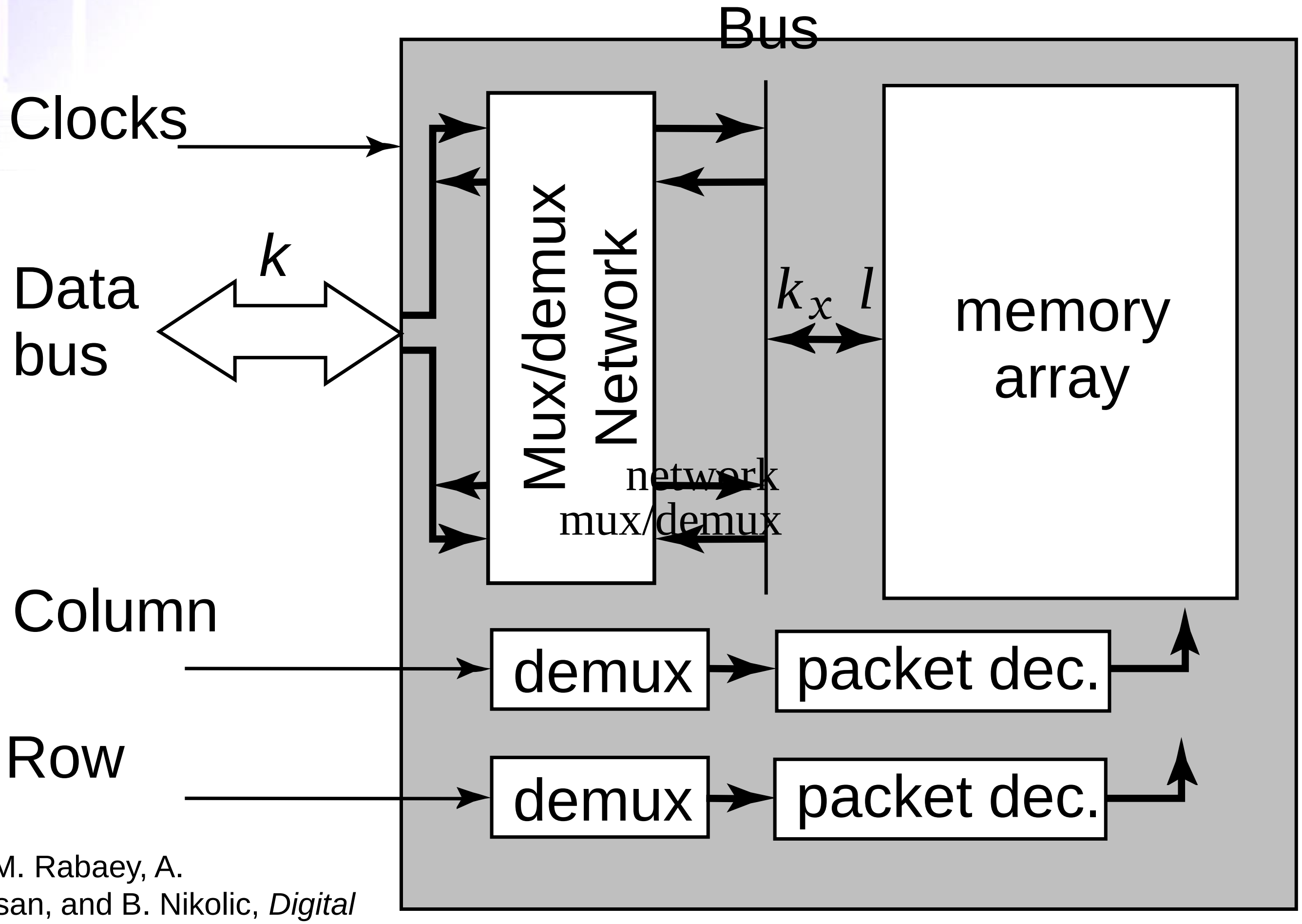
while control lines are idle

Source: K.-B. Lee, T.-C. Lin, and C.-W. Jen, "An efficient quality-aware memory controller for multimedia platform SoC," IEEE CSVT, May 2005.

Rambus DRAM (RDRAM)

- More of a bus interface architecture than DRAM architecture
- Data is latched on both rising and falling edge of clock (300MHz)
- Broken into 4 banks each with own row decoder
  - Can have 4 pages open at a time
- Multiple open page scheme
- Capable of very high throughput

RDRAM Architecture



Source: J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 2nd Ed., Prentice Hall, 2003.

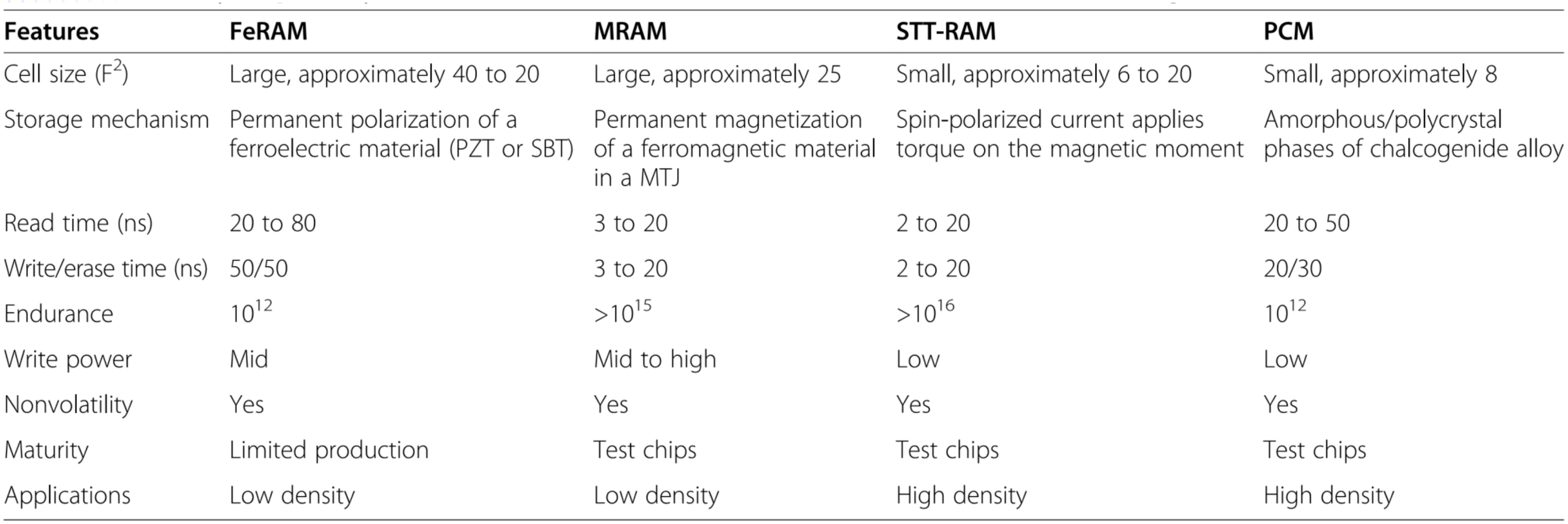
DRAM Integration Problem

- SRAM is easily integrated on the same chip
  - □ Ex. processor
- DRAM is more difficult
  - □ Different chip making process between DRAM and conventional logic
  - □ Goal of conventional logic (IC) process:
    - Minimize parasitic capacitance to reduce signal propagation delays and power consumption
  - □ Goal of DRAM process:
    - Create capacitor cells to retain stored information
  - □ Integration processes: embedded DRAM

Emerging NVM

- PCM (PRAM)
- MRAM, STT-RAM
- FeRAM
- 3D XPoint

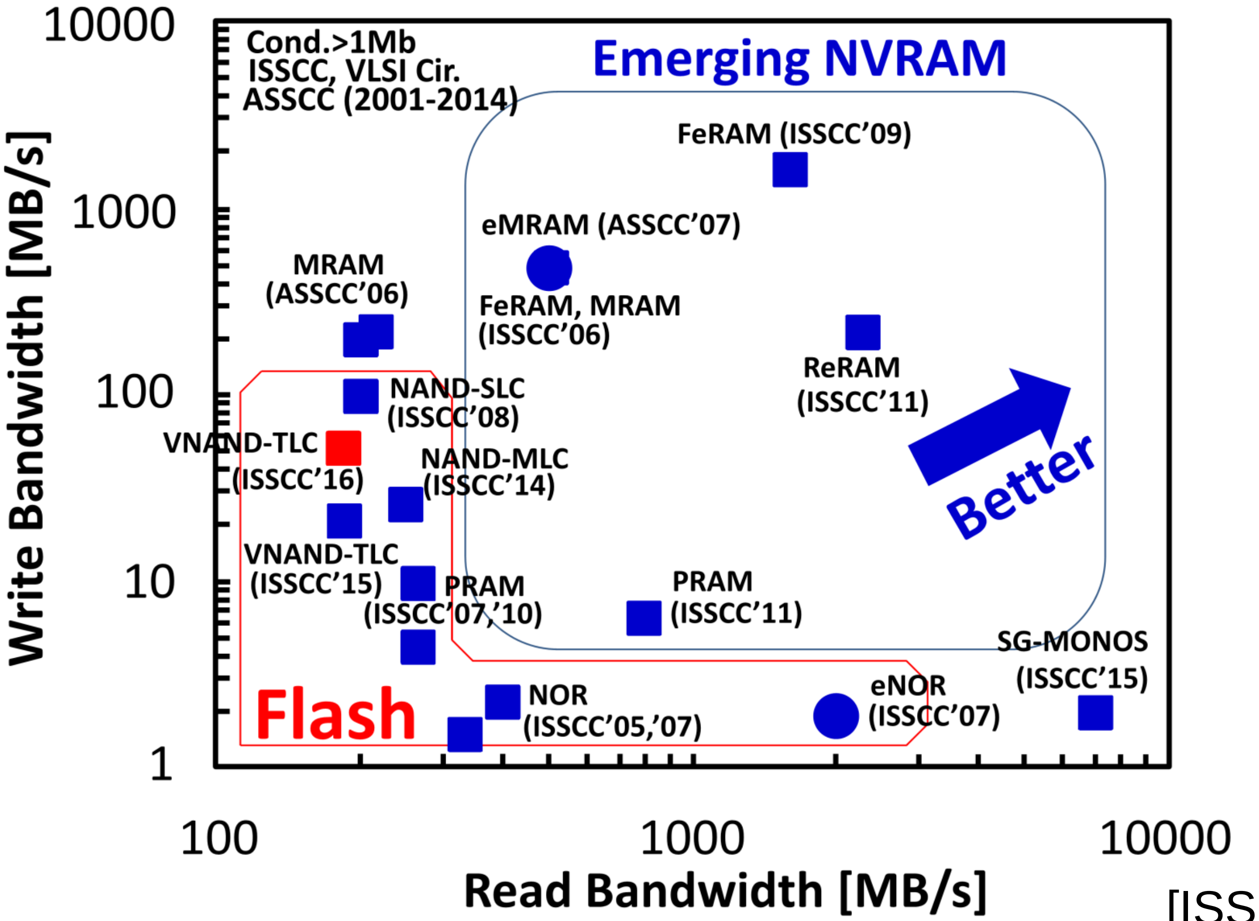
Emerging NVM



Features	FeRAM	MRAM	STT-RAM	PCM
Cell size (F^2)	Large, approximately 40 to 20	Large, approximately 25	Small, approximately 6 to 20	Small, approximately 8
Storage mechanism	Permanent polarization of a ferroelectric material (PZT or SBT)	Permanent magnetization of a ferromagnetic material in a MTJ	Spin-polarized current applies torque on the magnetic moment	Amorphous/polycrystal phases of chalcogenide alloy
Read time (ns)	20 to 80	3 to 20	2 to 20	20 to 50
Write/erase time (ns)	50/50	3 to 20	2 to 20	20/30
Endurance	10^{12}	$>10^{15}$	$>10^{16}$	10^{12}
Write power	Mid	Mid to high	Low	Low
Nonvolatility	Yes	Yes	Yes	Yes
Maturity	Limited production	Test chips	Test chips	Test chips
Applications	Low density	Low density	High density	High density

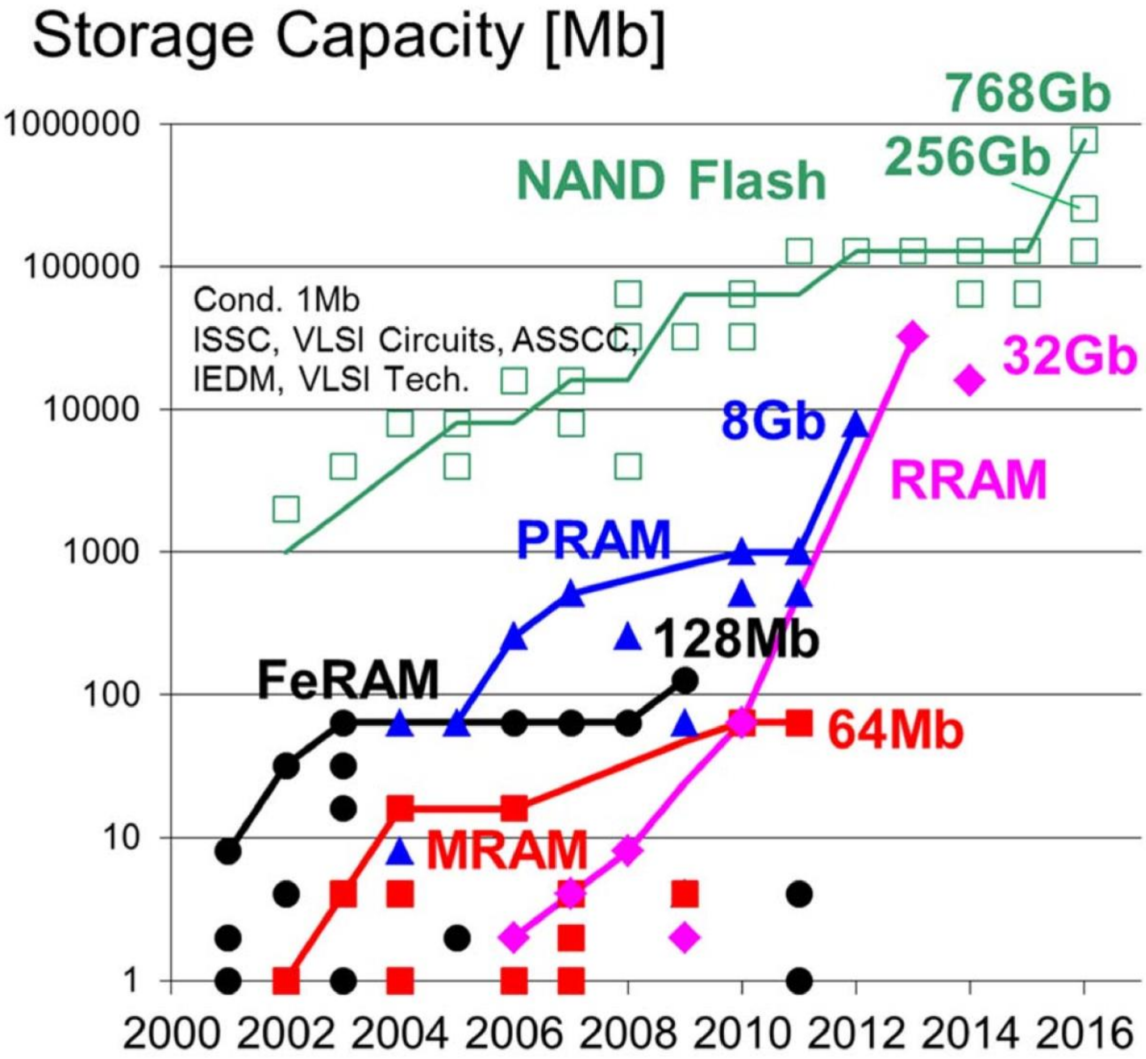
Jagan Singh Meena, Simon Min Sze, Umesh Chand and Tseung-Yuen Tseng, "Overview of emerging nonvolatile memory technologies," *Nanoscale Research Letters*, 2014, 9:526.

Emerging NVM



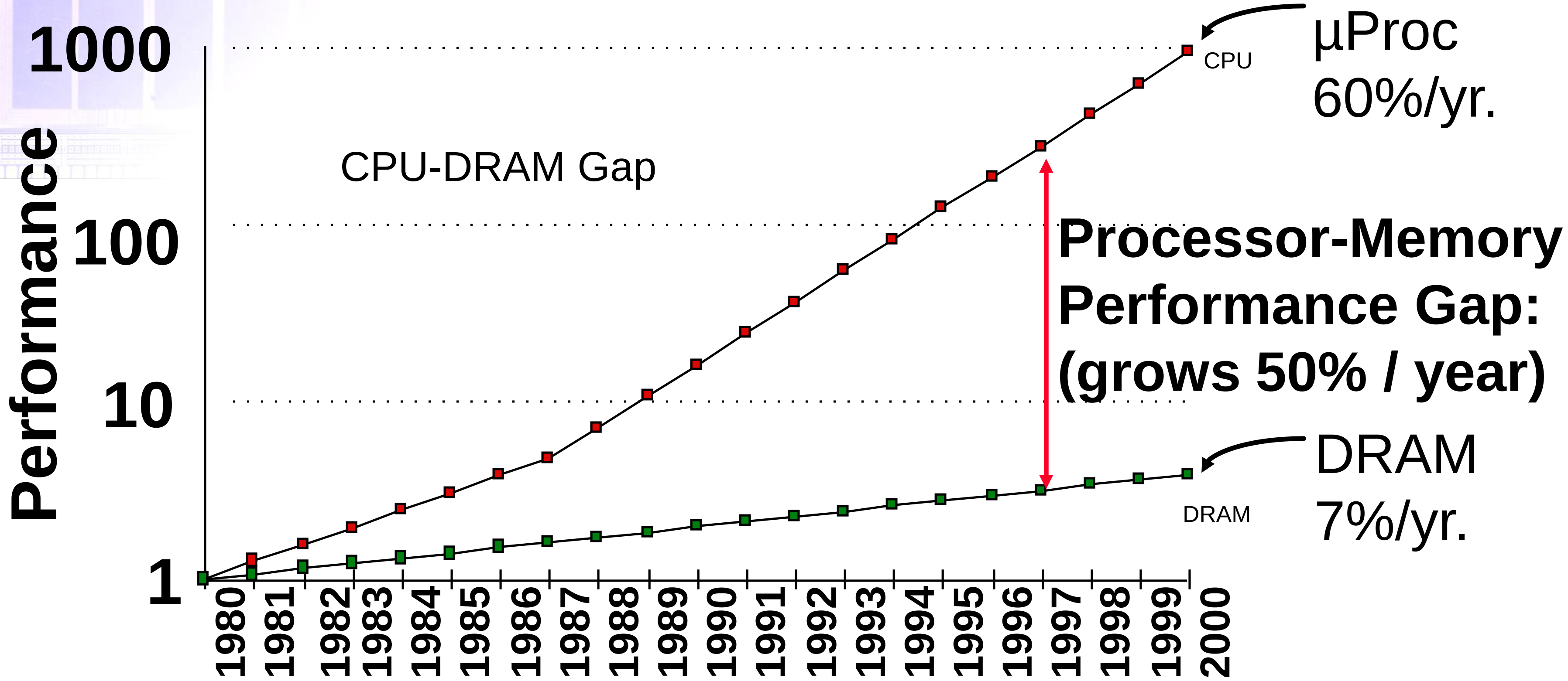
[ISSCC2016 Trends]

Emerging NVM



[ISSCC2016 Trends]

Who Cares About the Memory Hierarchy?



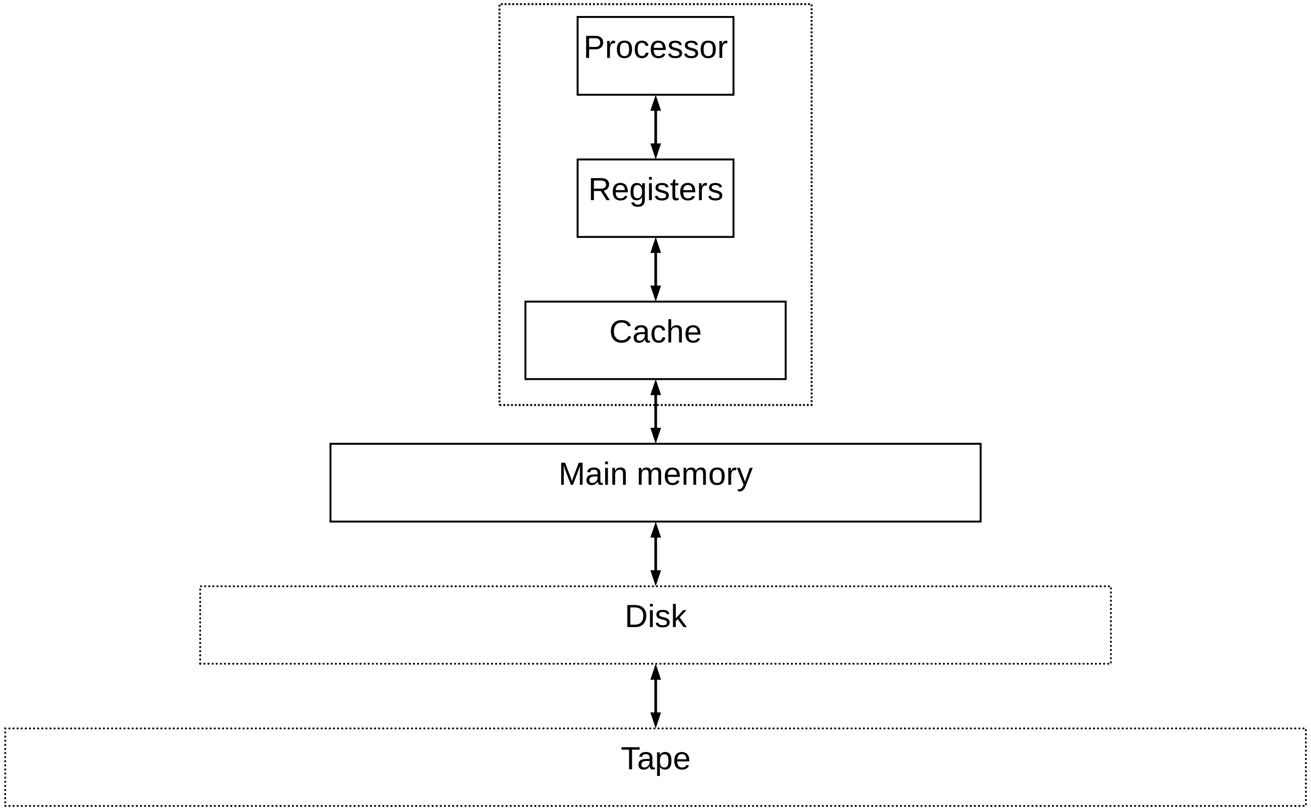
- 1980: no cache in µproc; 1995 2-level cache on chip (1989 first Intel µproc with a cache on chip)

Source: J. L. Hennessy and D. A. Patterson, Ch. 5, *Computer Architecture: a Quantitative Approach*, 3rd Ed., Morgan Kaufmann, 2003.

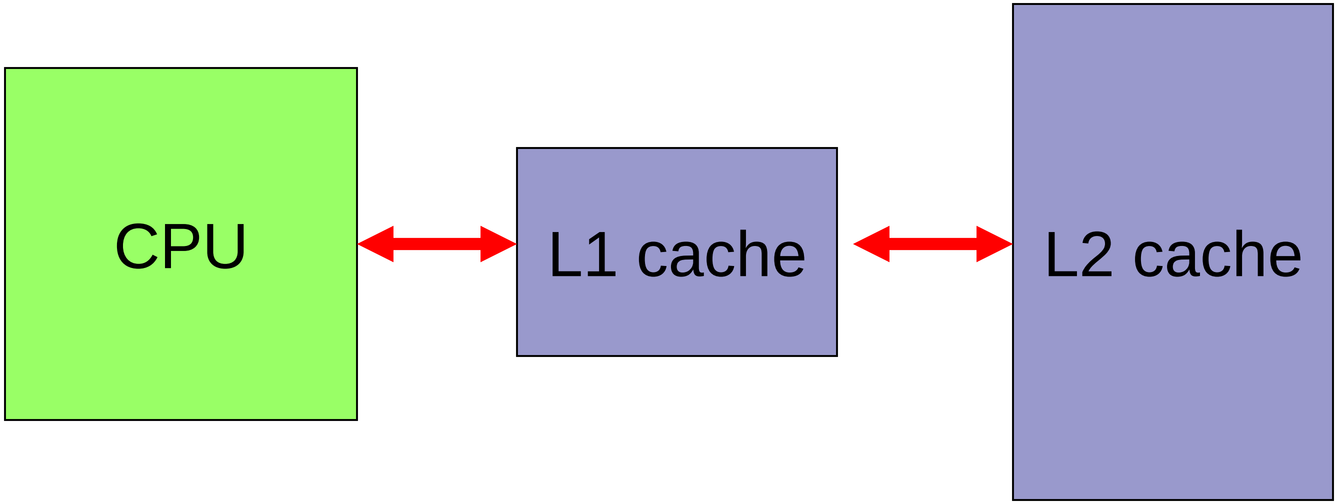
Memory Hierarchy

- Microprocessor clock rates are increasing at a faster rate than memory speeds
- Want inexpensive, fast memory
- Main memory
  - Large, inexpensive, slow memory stores entire program and data
- Cache
  - Small, expensive, fast memory stores copy of likely accessed parts of larger memory
  - Can be multiple levels of cache
  - Increase the average performance of the memory system
- Local buffer/scratchpad
  - Like cache
  - Sometimes used in dedicated hardware accelerators

Memory Hierarchy



Multiple Levels of Cache



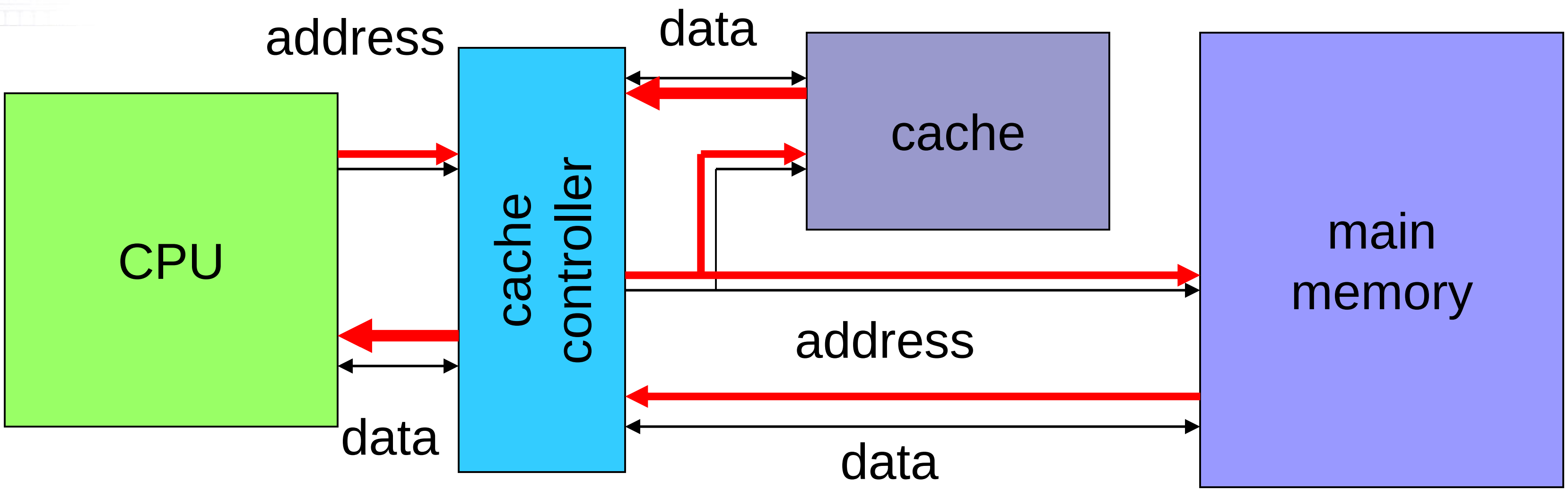
Cache

- A cache is a small, fast memory that holds copies of some of the contents of main memory
- Useful when the CPU is using only a relatively small set of memory locations (**working set**)
- Usually designed with SRAM
  - Faster but more expensive than DRAM
- Usually on the same chip as processor
  - Space limited, so much smaller than off-chip main memory
  - Faster access (1 cycle vs. several cycles for main memory)
- **Cache controller** is required
- Several cache design choices
  - **Cache mapping**, **replacement policies**, and **write techniques**

Cache Operation

- Many main memory locations are mapped onto one cache entry
- May have caches for:
  - instructions;
  - data;
  - data + instructions (**unified**)
- Memory access time is no longer deterministic
- Request for main memory access (read or write)
- First, check cache for copy
  - **Cache hit**
    - Copy is in cache, quick access
  - **Cache miss**
    - Copy not in cache, read address and possibly its neighbors into cache

Caches and CPUs



Types of Misses

- Compulsory (cold): location has never been accessed.
- Capacity: working set is too large.
- Conflict: multiple locations in working set map to same cache entry.

Memory System Performance

- h = cache hit rate.
- t_{cache} = cache access time, t_{main} = main memory access time.
- Average memory access time:
  - □ $t_{av} = ht_{cache} + (1-h)t_{main}$

Multi-Level Cache Access Time

- h_1 = cache hit rate.
- h_2 = rate for miss on L1, hit on L2.
- Average memory access time:
  - □ $t_{av} = h_1 t_{L1} + h_2 t_{L2\,+} (1 - h_2 - h_1) t_{main}$

Cache Mapping

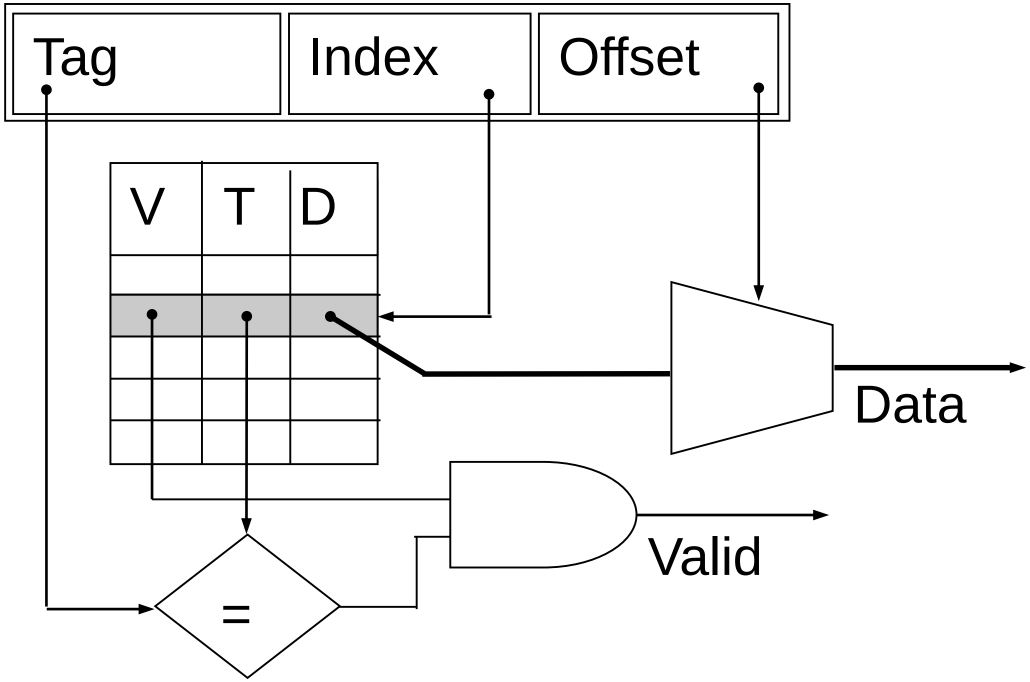
- Far fewer number of available cache addresses
- Are address' contents in cache?
- Cache mapping is used to assign main memory address to cache address and determine hit or miss
- Three basic techniques:
  - □ Direct mapping
  - □ Fully associative mapping
  - □ Set-associative mapping
- Caches are partitioned into indivisible **cache blocks** or **cache lines** of adjacent memory addresses
  - □ Usually 4 or 8 addresses per line

Direct Mapping

- Main memory address is divided into 2 fields
  - Index
    - Cache address
    - Number of bits determined by cache size
  - Tag
    - Compared with tag stored in cache at address indicated by index
    - If tags match, check valid bit
- Valid bit
  - Indicates whether data in slot has been loaded from memory
- Offset
  - Used to find particular word in cache line

Direct Mapping

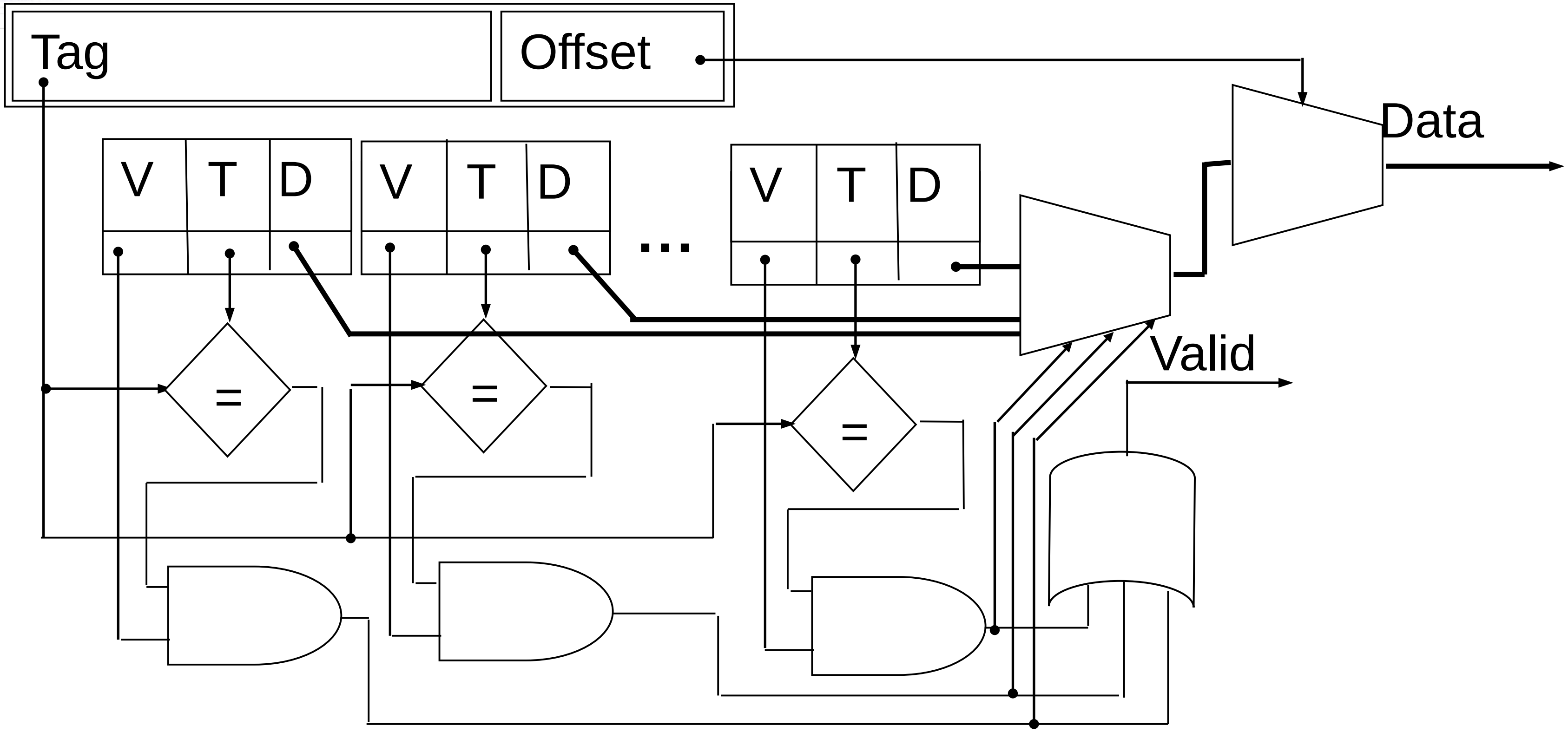
V: Valid
T: Tag
D: Data



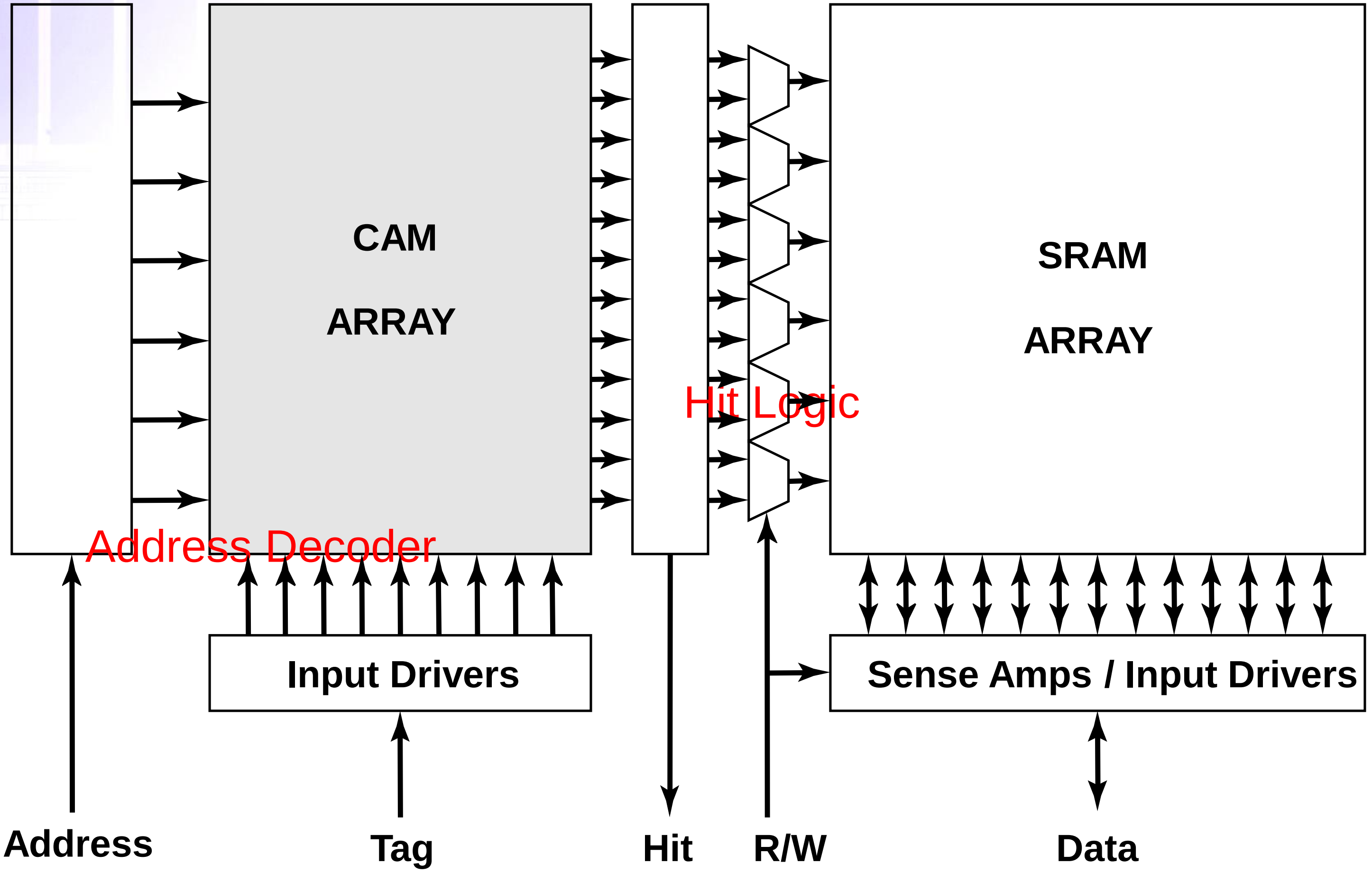
Fully Associative Mapping

- Complete main memory address stored in each cache address
- All addresses stored in cache simultaneously compared with desired address
  - Can also be implemented with CAM (content-addressable memory)
- Valid bit and offset are the same as direct mapping

Fully Associative Mapping



CAM in Cache Memory

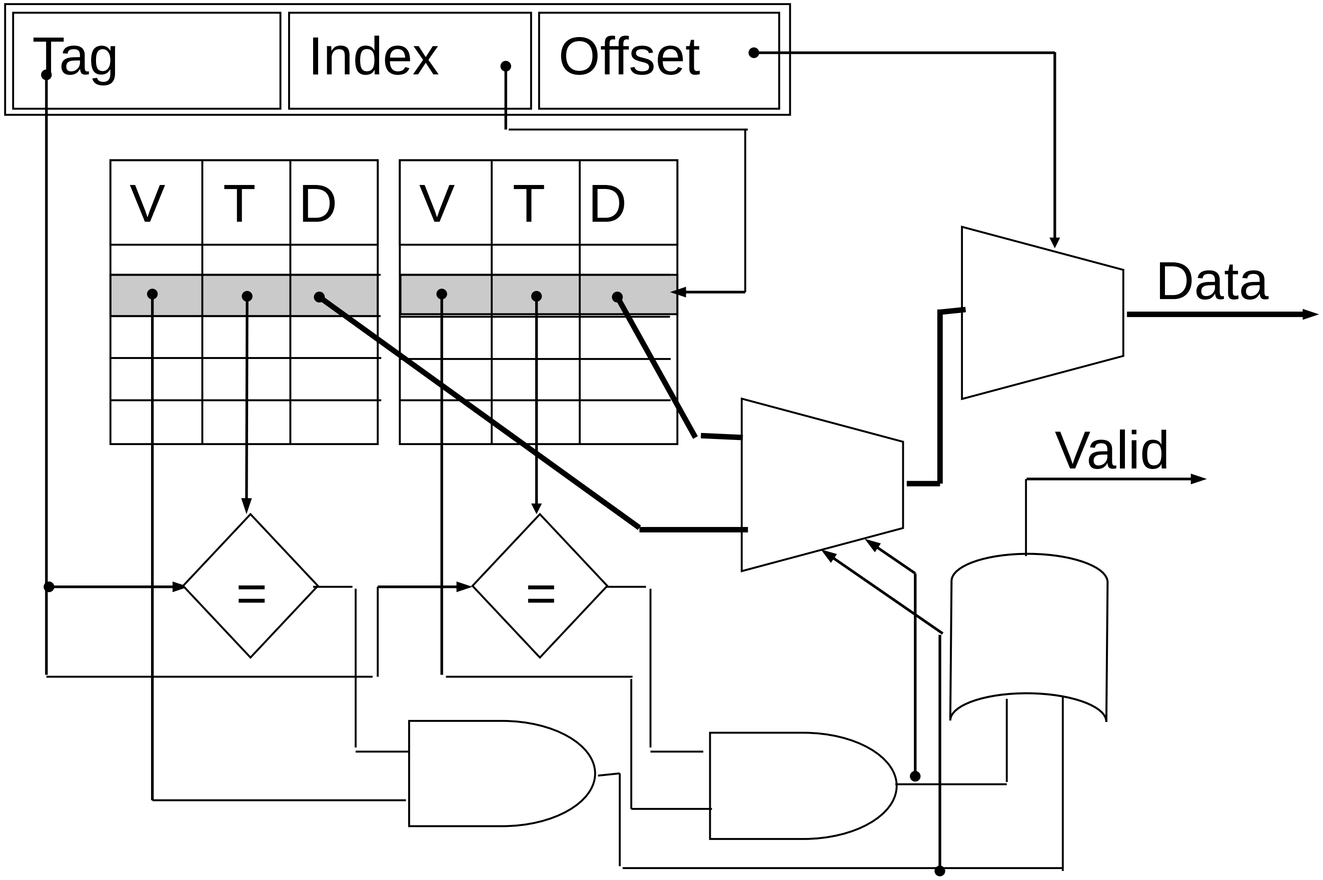


Source: J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 2nd Ed., Prentice Hall, 2003.

Set-Associative Mapping

- Compromise between direct mapping and fully associative mapping
- Index is the same as in direct mapping
- But, each cache address contains content and tags of 2 or more memory address locations
- Tags of that **set** simultaneously compared as in fully associative mapping
- Cache with set size N called N-way set-associative
  - 2-way, 4-way, 8-way are common

Set-Associative Mapping



Cache-Replacement Policy

- Technique for choosing which block to replace
  - When fully associative cache is full
  - When set-associative cache's line is full
- Direct mapped cache has no choice
- Random
  - Replace block chosen at random
- LRU: least-recently used
  - Replace block not accessed for longest time
- FIFO: first-in-first-out
  - Push block onto queue when accessed
  - Choose block to replace by popping queue

Cache Write Techniques

- When written, data cache must update main memory
- **Write-through**
  - Write to main memory whenever cache is written to
  - Easiest to implement
  - Processor must wait for slower main memory write
  - Potential for unnecessary writes
- **Write-back**
  - Main memory only written when “dirty” block replaced
  - Extra **dirty bit** for each block set when cache block written to
  - Reduces number of slow main memory writes

Example: Direct-mapped vs. Set-Associative

address	data
000	0101
001	1111
010	0000
011	0110
100	1000
101	0001
110	1010
111	0100

Direct-Mapped Cache Behavior

- After 001 access:

block	tag	data
00	-	-
01	0	1111
10	-	-
11	-	-

- After 010 access:

block	tag	data
00	-	-
01	0	1111
10	0	0000
11	-	-

Direct-Mapped Cache Behavior

- After 011 access:

block	tag	data
00	-	-
01	0	1111
10	0	0000
11	0	0110

- After 100 access:

block	tag	data
00	1	1000
01	0	1111
10	0	0000
11	0	0110

Direct-Mapped Cache Behavior

- After 101 access:

block	tag	data
00	1	1000
01	1	0001
10	0	0000
11	0	0110

- After 111 access:

block	tag	data
00	1	1000
01	1	0001
10	0	0000
11	1	0100

2-Way Set-Associative Cache Behavior

- Final state of cache (twice as big as direct-mapped):

set	blk 0 tag	blk 0 data	blk 1 tag	blk 1 data
00	1	1000	-	-
01	0	1111	1	0001
10	0	0000	-	-
11	0	0110	1	0100

2-Way Set-Associative Cache Behavior

- Final state of cache (same size as direct-mapped):

set	blk 0 tag	blk 0 data	blk 1 tag	blk 1 data
0	01	0000	10	1000
1	10	0111	11	0100

Example Caches

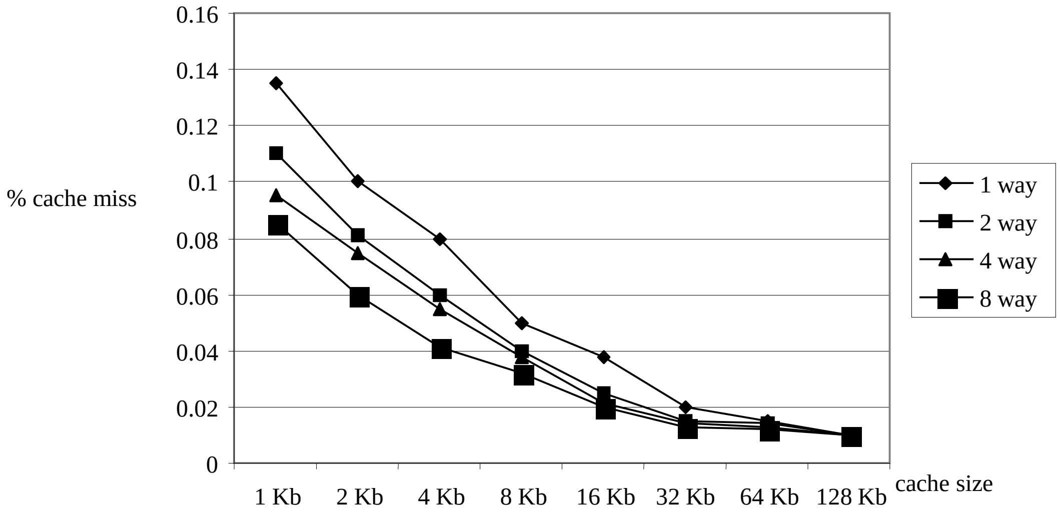
- StrongARM:
    - 16 Kbyte, 32-way, 32-byte block instruction cache.
    - 16 Kbyte, 32-way, 32-byte block data cache (write-back).
- SHARC:
    - 32-instruction, 2-way instruction cache.

Cache Impact on System Performance

- Most important parameters in terms of performance:
  - □ Total size of cache
    - Total number of data bytes cache can hold
    - Tag, valid and other house keeping bits not included in total
  - □ Degree of associativity
  - □ Data block size
- Larger caches achieve lower miss rates but higher access cost
  - □ e.g.,
    - 2 Kbyte cache: miss rate = 15%, hit cost = 2 cycles, miss cost = 20 cycles
      - □ avg. cost of memory access = (0.85 * 2) + (0.15 * 20) = 4.7 cycles
    - 4 Kbyte cache: miss rate = 6.5%, hit cost = 3 cycles, miss cost will not change
      - □ avg. cost of memory access = (0.935 * 3) + (0.065 * 20) = 4.105 cycles **(improvement)**
    - 8 Kbyte cache: miss rate = 5.565%, hit cost = 4 cycles, miss cost will not change
      - □ avg. cost of memory access = (0.94435 * 4) + (0.05565 * 20) = 4.8904 cycles **(worse)**

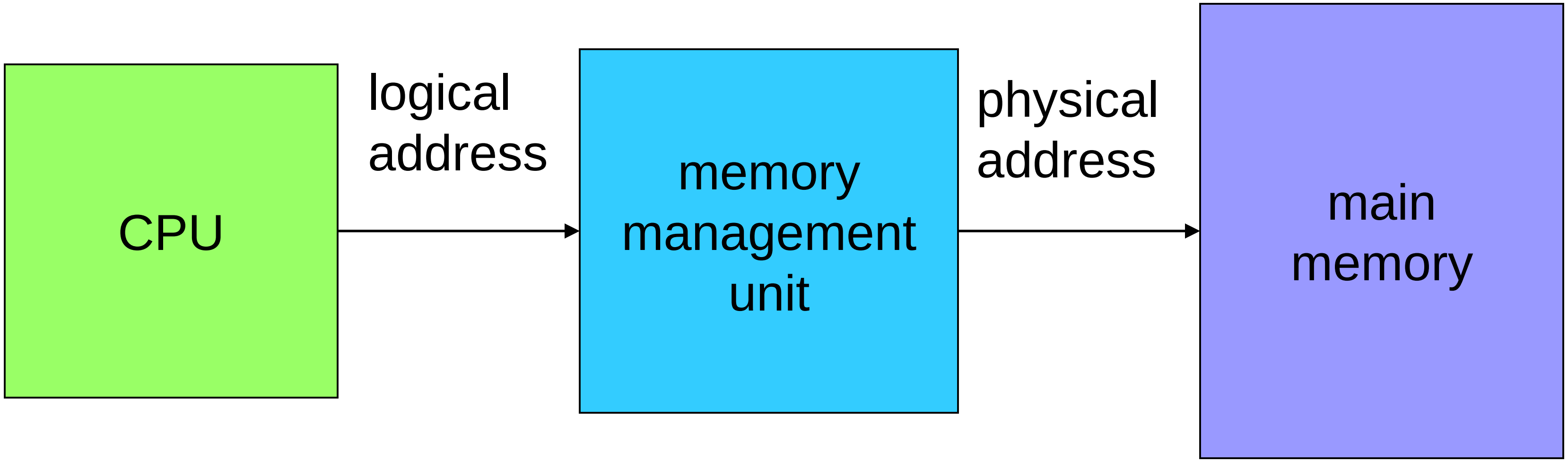
Cache Performance Trade-Offs

- Improving cache hit rate without increasing size
  - Increase line size
  - Change set-associativity



Memory Management Units

- Memory management unit (MMU) translates addresses:



(Hardware page-table walker)

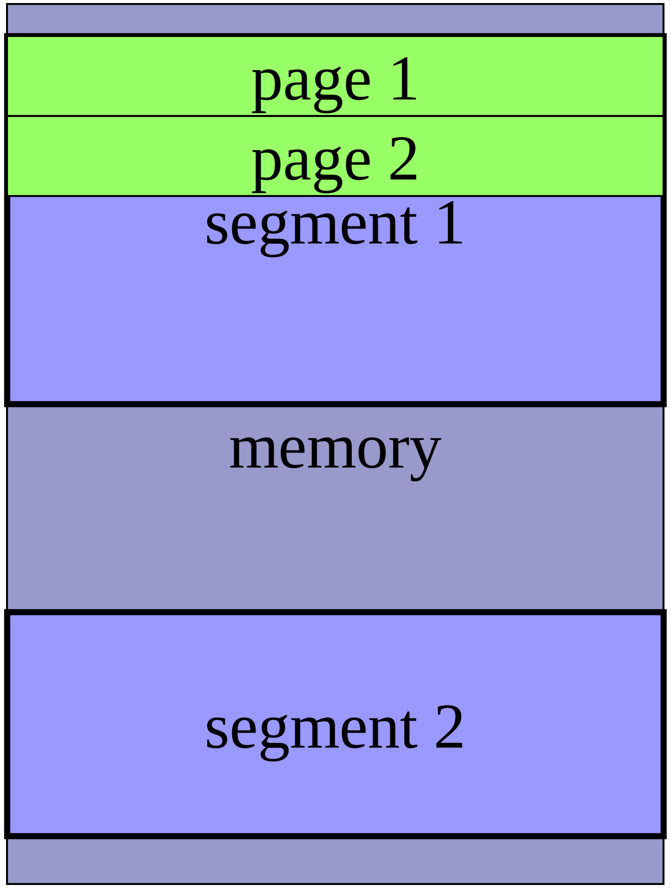
Memory Management Tasks

- Allows programs to move in physical memory during execution
- Allow software to manage multiple programs in a single physical memory, each with its own address space
- Allows virtual memory:
  - □ Memory images kept in secondary storage;
  - □ Images returned to main memory on demand during execution.
- Page fault: MMU generates an exception when requesting for location not resident in memory.

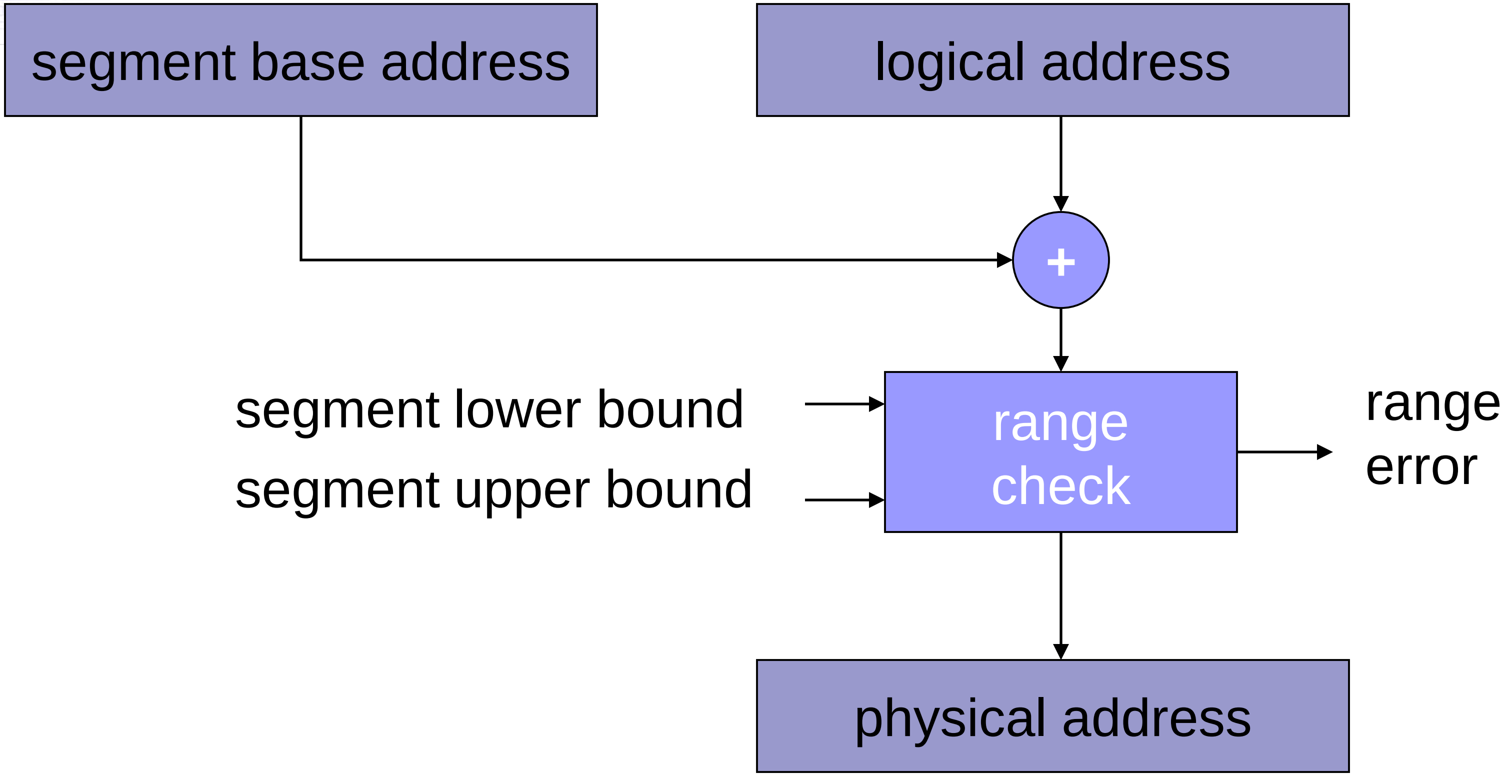
Address Translation

- Requires some sorts of register/table to allow arbitrary mappings of logical to physical addresses.
- Two basic schemes:
  - □ Segmented: arbitrary size, described by its start address and size
  - □ Paged: uniform size, which simplifies the hardware required for address translation
    - Can support **fragmentation** for a program
- Segmentation and paging can be combined (x86): divide each segment into pages and using two steps for address translation

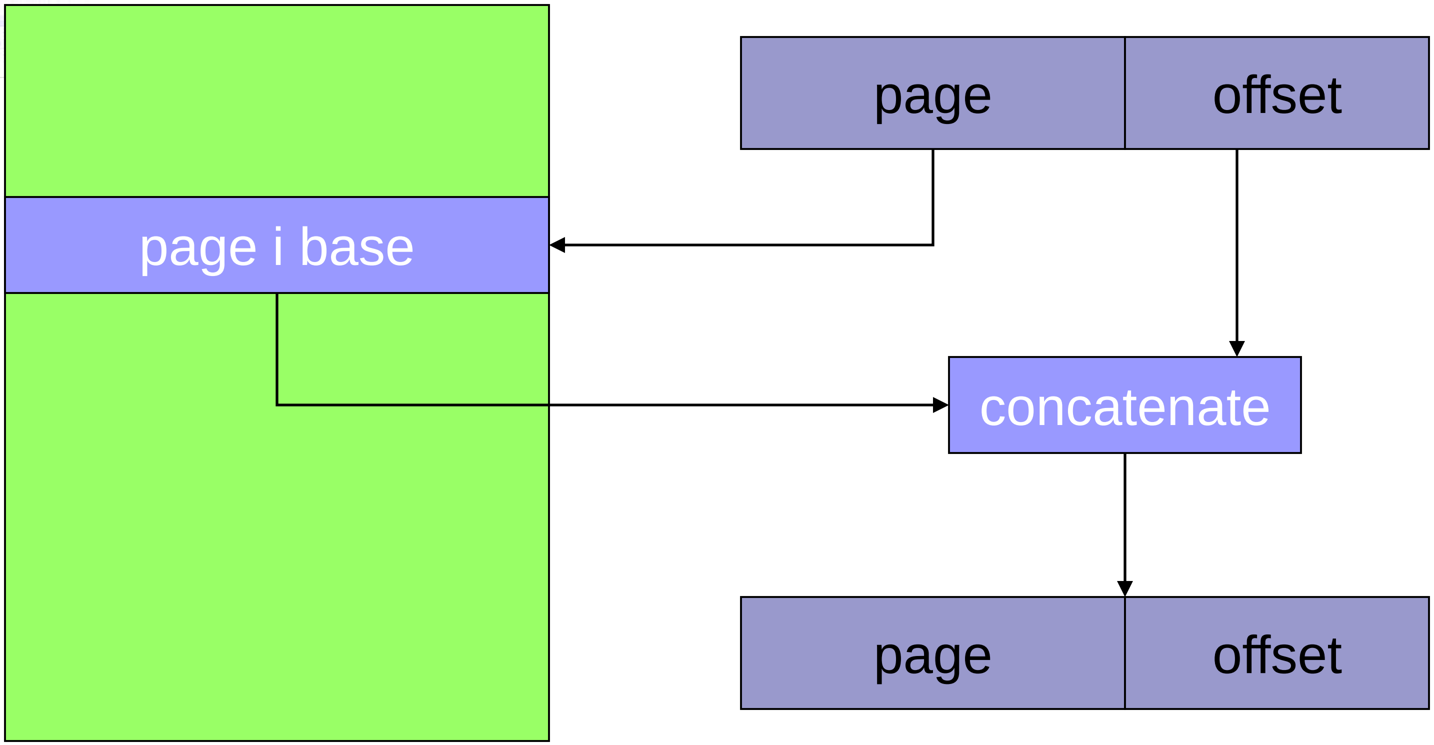
Segments and Pages



Segment Address Translation

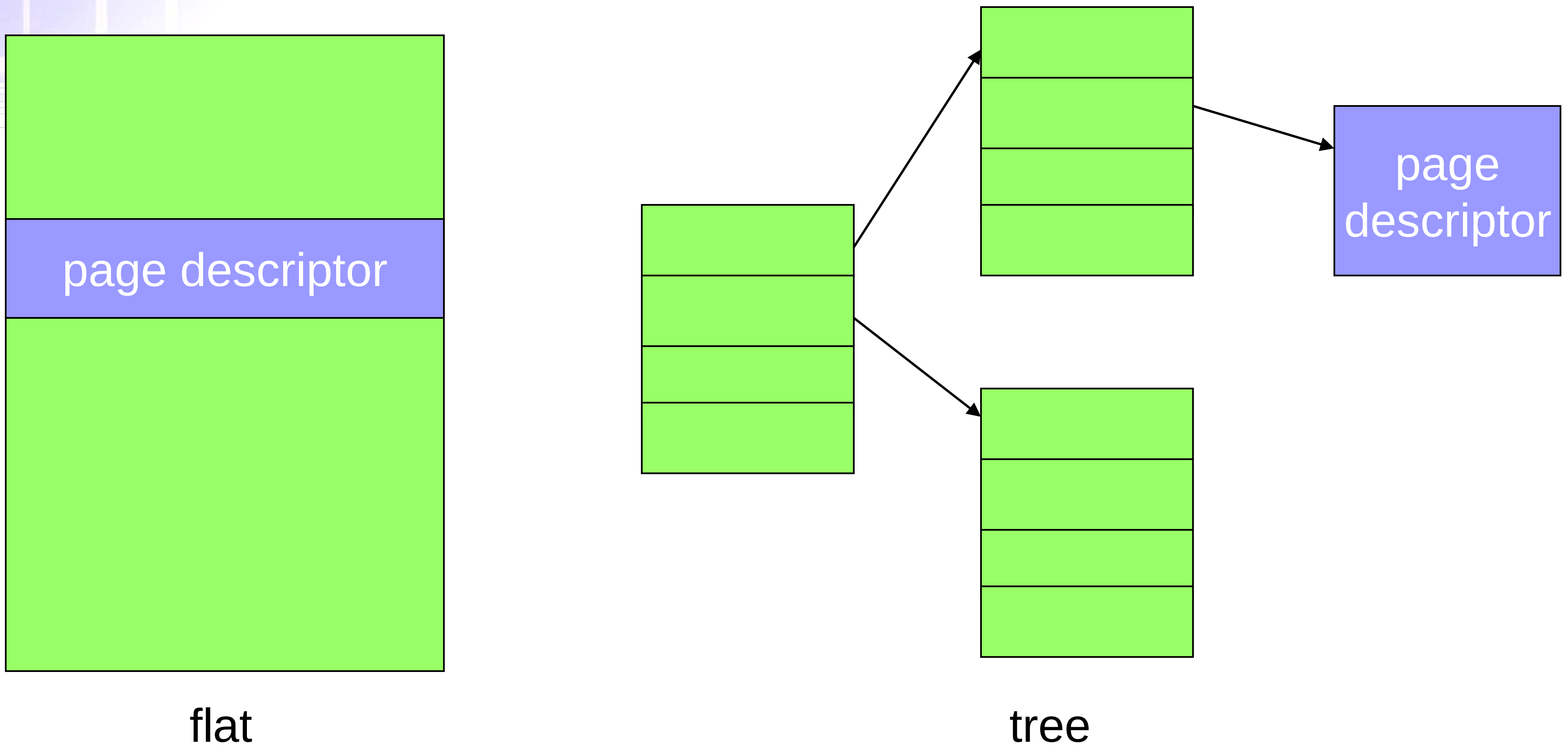


Page Address Translation



Do not need to check the boundary
Typical page size: 512 bytes to 4K bytes

Page Table Organizations



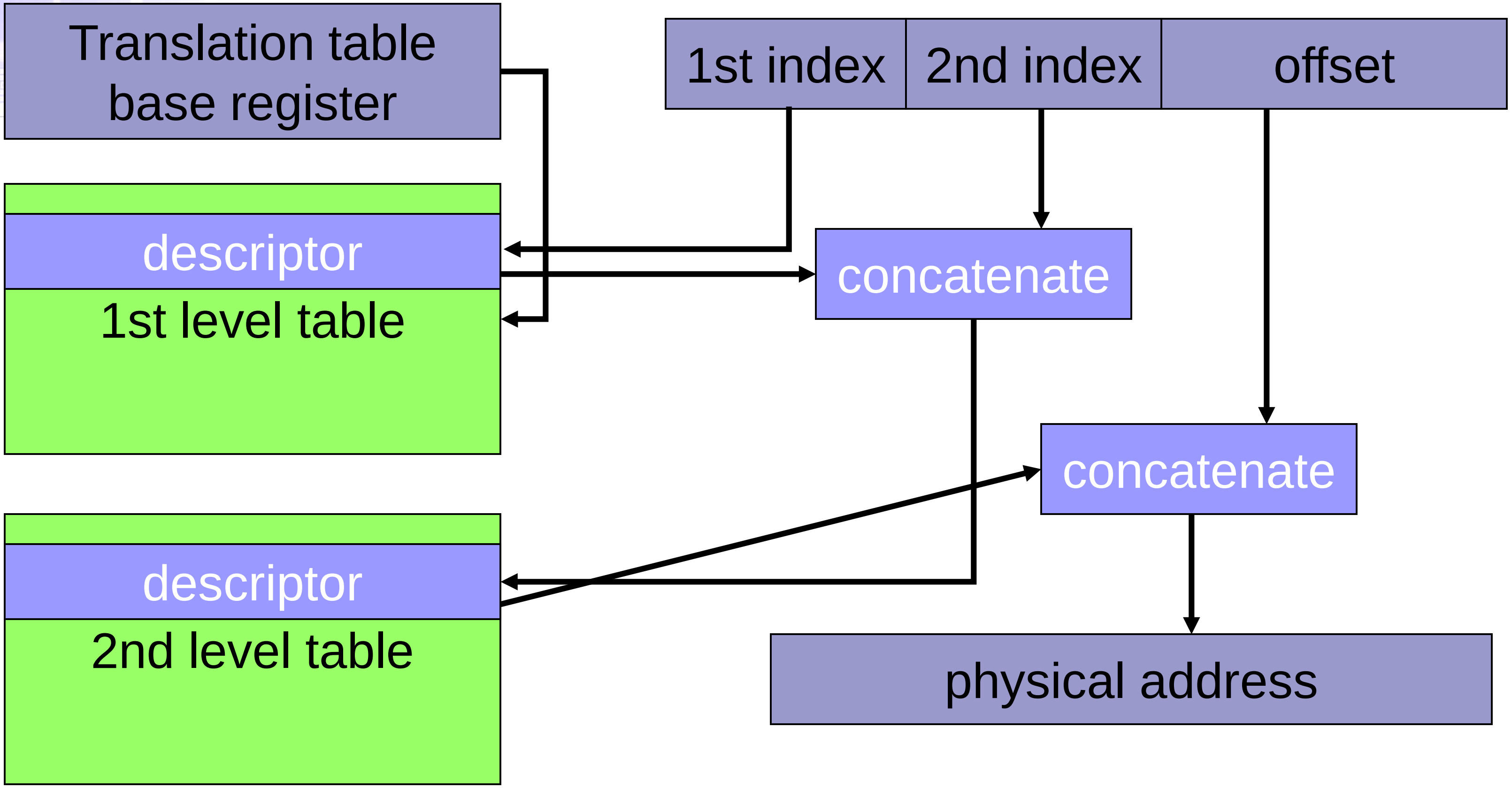
Caching Address Translations

- Large translation tables require main memory access.
- TLB: translation lookaside buffer, cache for address translation.
  - Typically small

ARM Memory Management

- Memory region types:
  - Section: 1 Mbyte block;
  - Large page: 64 kbytes;
  - Small page: 4 kbytes.
- An address is marked as section-mapped or page-mapped.
- Two-level translation scheme.

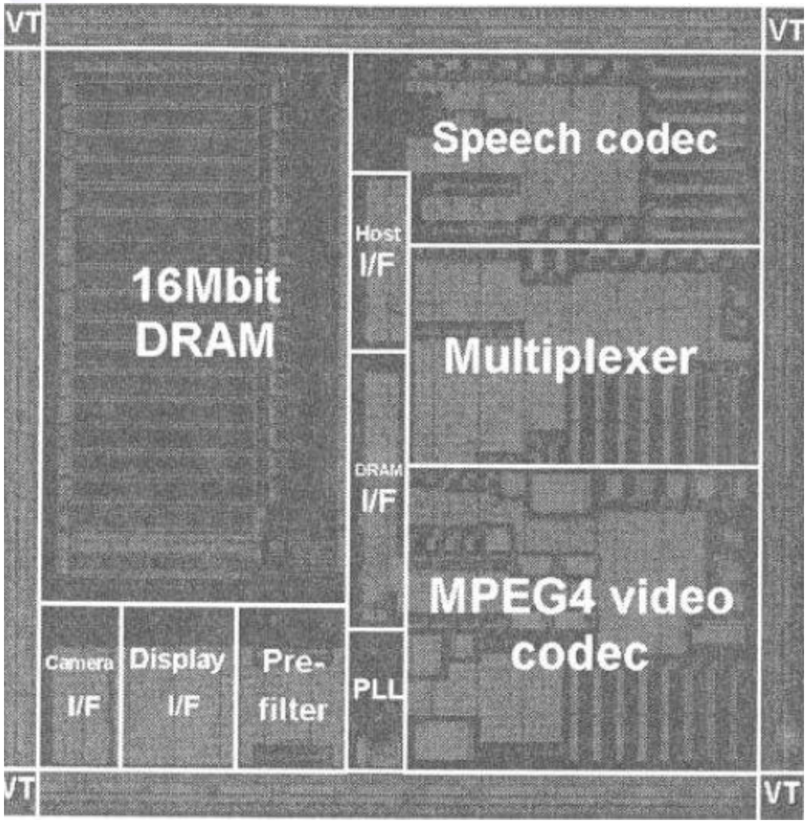
ARM Address Translation



New Memory Packaging in SoC

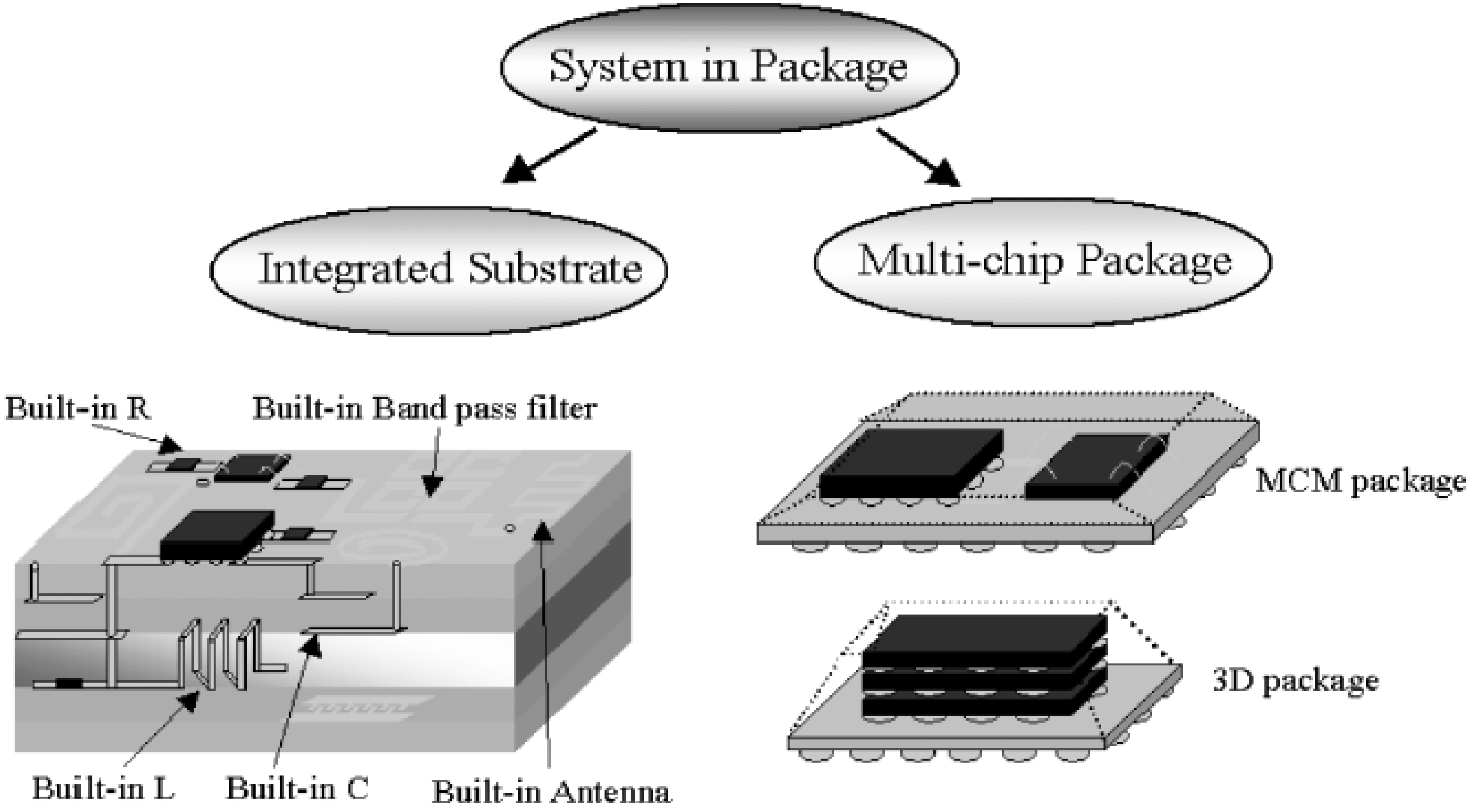
- Embedded DRAM
- Stacked memory
  - □ Known good die (KGD)
- System in Package, silicon interposer
- 3D-IC
- Wide I/O, HBM, HMC
- …

Embedded DRAM

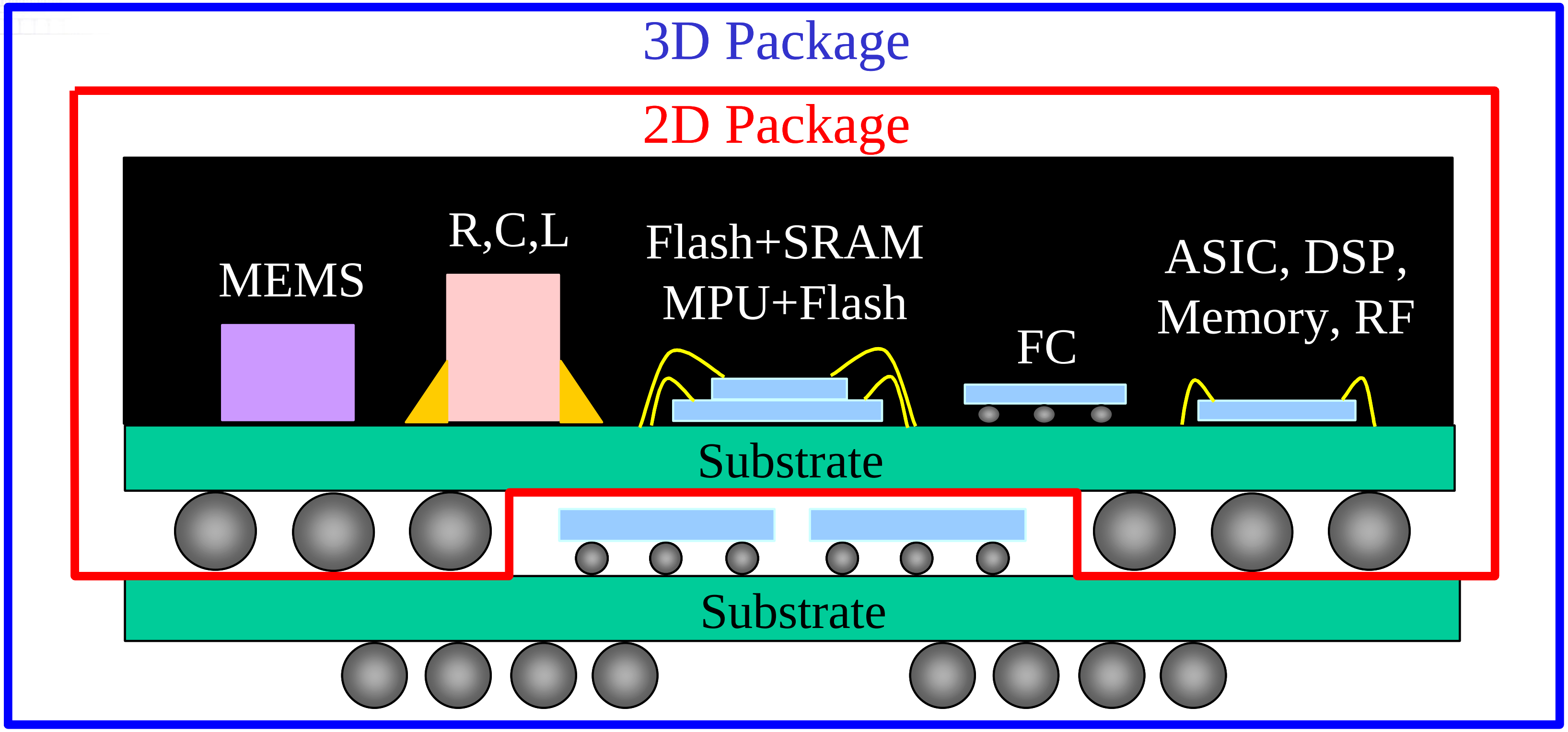


Source: T. Nishikawa et al., "A 60MHz 240mW MPEG-4 Video-Phone LSI with 16Mb Embedded DRAM," *ISSCC2000*.

Stacked Memory



Stacked Memory

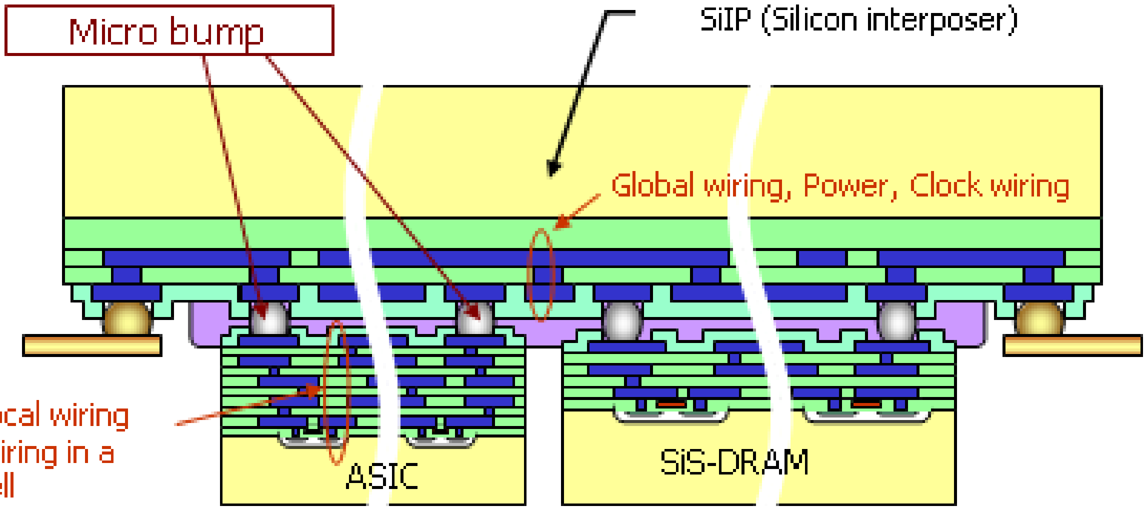


Source: 鉅景科技；工研院IEK-ITIS計畫(2003/06)

System in Silicon

SiS (System-in-Silicon) architecture

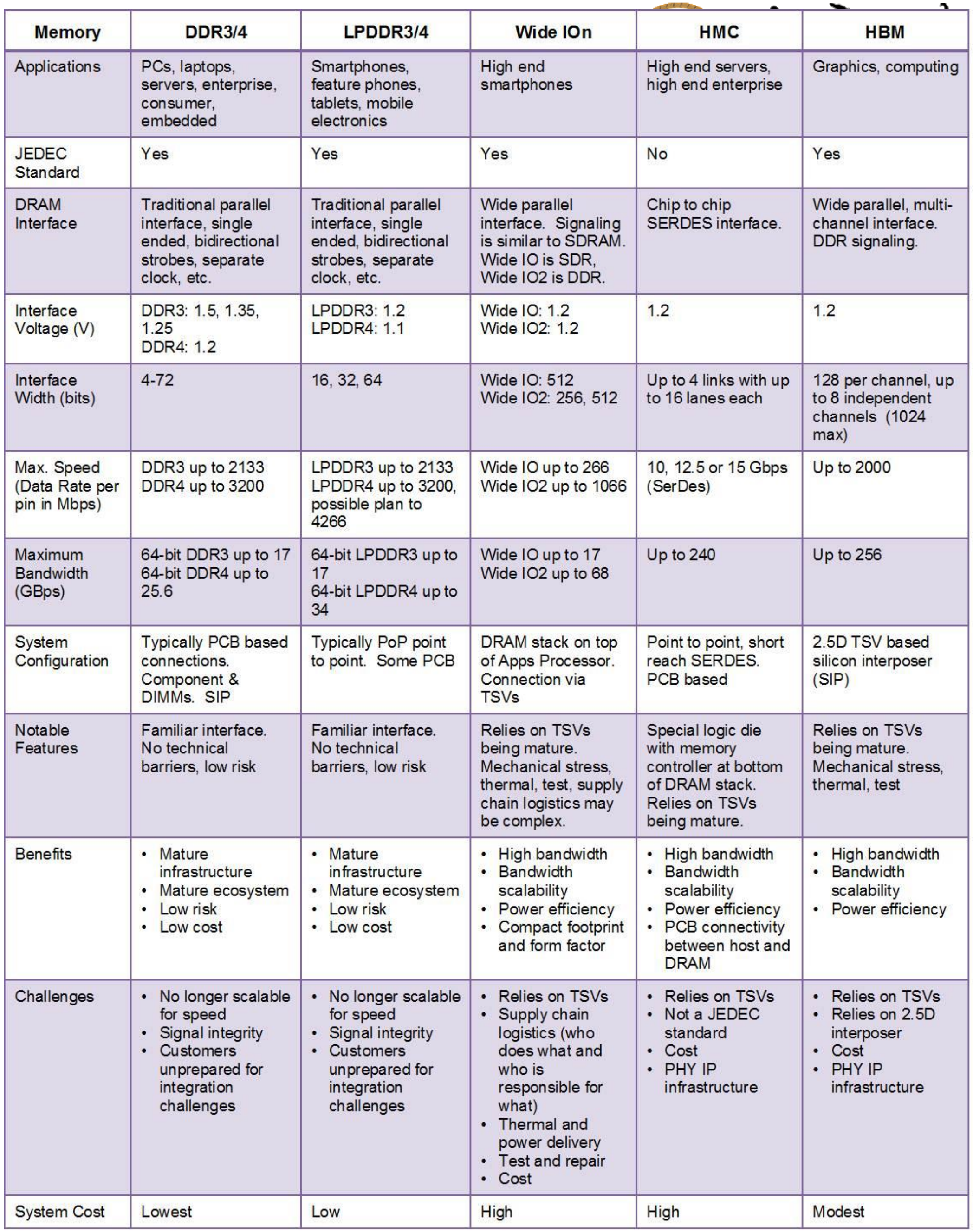
- an applied technology of silicon interposer(SiIP) and micro bump
- SoC design methodology + Multichip fabrication



Source: System Fabrication Technologies (SFT)

New Standards

- Wide I/O, HBM, HMC

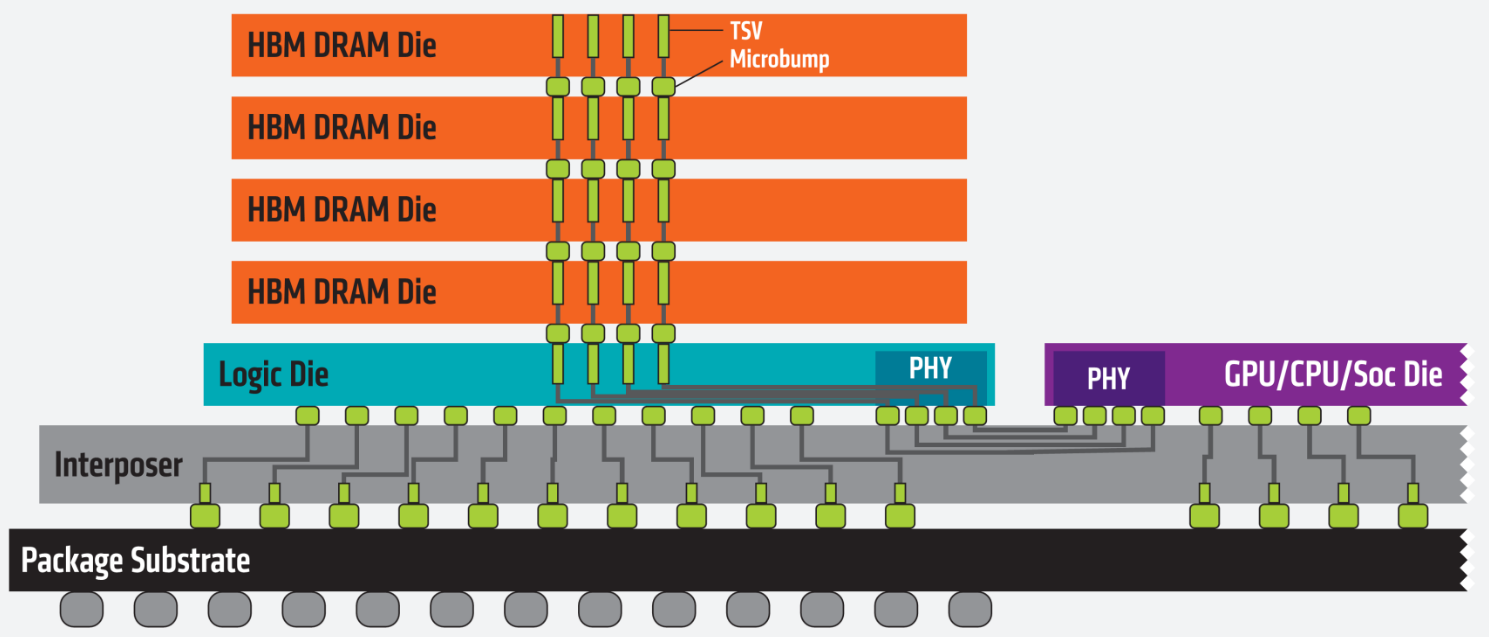


Memory	DDR3/4	LPDDR3/4	Wide IOn	HMC	HBM
Applications	PCs, laptops, servers, enterprise, consumer, embedded	Smartphones, feature phones, tablets, mobile electronics	High end smartphones	High end servers, high end enterprise	Graphics, computing
JEDEC Standard	Yes	Yes	Yes	No	Yes
DRAM Interface	Traditional parallel interface, single ended, bidirectional strobes, separate clock, etc.	Traditional parallel interface, single ended, bidirectional strobes, separate clock, etc.	Wide parallel interface. Signaling is similar to SDRAM. Wide IO is SDR, Wide IO2 is DDR.	Chip to chip SERDES interface.	Wide parallel, multi-channel interface. DDR signaling.
Interface Voltage (V)	DDR3: 1.5, 1.35, 1.25 DDR4: 1.2	LPDDR3: 1.2 LPDDR4: 1.1	Wide IO: 1.2 Wide IO2: 1.2	1.2	1.2
Interface Width (bits)	4-72	16, 32, 64	Wide IO: 512 Wide IO2: 256, 512	Up to 4 links with up to 16 lanes each	128 per channel, up to 8 independent channels (1024 max)
Max. Speed (Data Rate per pin in Mbps)	DDR3 up to 2133 DDR4 up to 3200	LPDDR3 up to 2133 LPDDR4 up to 3200, possible plan to 4266	Wide IO up to 266 Wide IO2 up to 1066	10, 12.5 or 15 Gbps (SerDes)	Up to 2000
Maximum Bandwidth (GBps)	64-bit DDR3 up to 17 64-bit DDR4 up to 25.6	64-bit LPDDR3 up to 17 64-bit LPDDR4 up to 34	Wide IO up to 17 Wide IO2 up to 68	Up to 240	Up to 256
System Configuration	Typically PCB based connections. Component & DIMMs. SIP	Typically PoP point to point. Some PCB	DRAM stack on top of Apps Processor. Connection via TSVs	Point to point, short reach SERDES. PCB based	2.5D TSV based silicon interposer (SIP)
Notable Features	Familiar interface. No technical barriers, low risk	Familiar interface. No technical barriers, low risk	Relies on TSVs being mature. Mechanical stress, thermal, test, supply chain logistics may be complex.	Special logic die with memory controller at bottom of DRAM stack. Relies on TSVs being mature.	Relies on TSVs being mature. Mechanical stress, thermal, test
Benefits	• Mature infrastructure • Mature ecosystem • Low risk • Low cost	• Mature infrastructure • Mature ecosystem • Low risk • Low cost	• High bandwidth • Bandwidth scalability • Power efficiency • Compact footprint and form factor	• High bandwidth • Bandwidth scalability • Power efficiency • PCB connectivity between host and DRAM	• High bandwidth • Bandwidth scalability • Power efficiency
Challenges	• No longer scalable for speed • Signal integrity • Customers unprepared for integration challenges	• No longer scalable for speed • Signal integrity • Customers unprepared for integration challenges	• Relies on TSVs • Supply chain logistics (who does what and who is responsible for what) • Thermal and power delivery • Test and repair • Cost	• Relies on TSVs • Not a JEDEC standard • Cost • PHY IP infrastructure	• Relies on TSVs • Relies on 2.5D interposer • Cost • PHY IP infrastructure
System Cost	Lowest	Low	High	High	Modest

http://www.extremetech.com/computing/197720-beyond-ddr4-understand-the-differences-between-wide-io-hbm-and-hybrid-memory-cube/2

New Standards

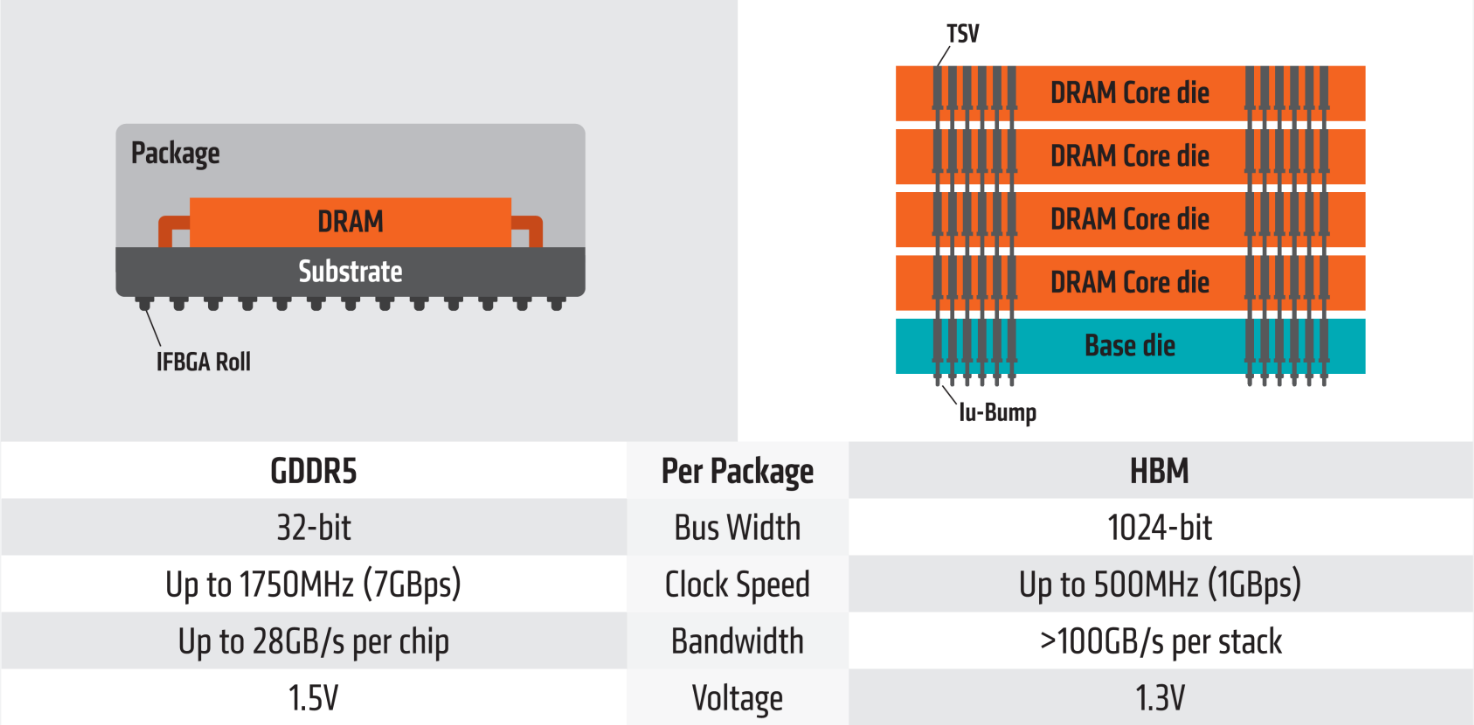
- HBM



[AMD]

New Standards

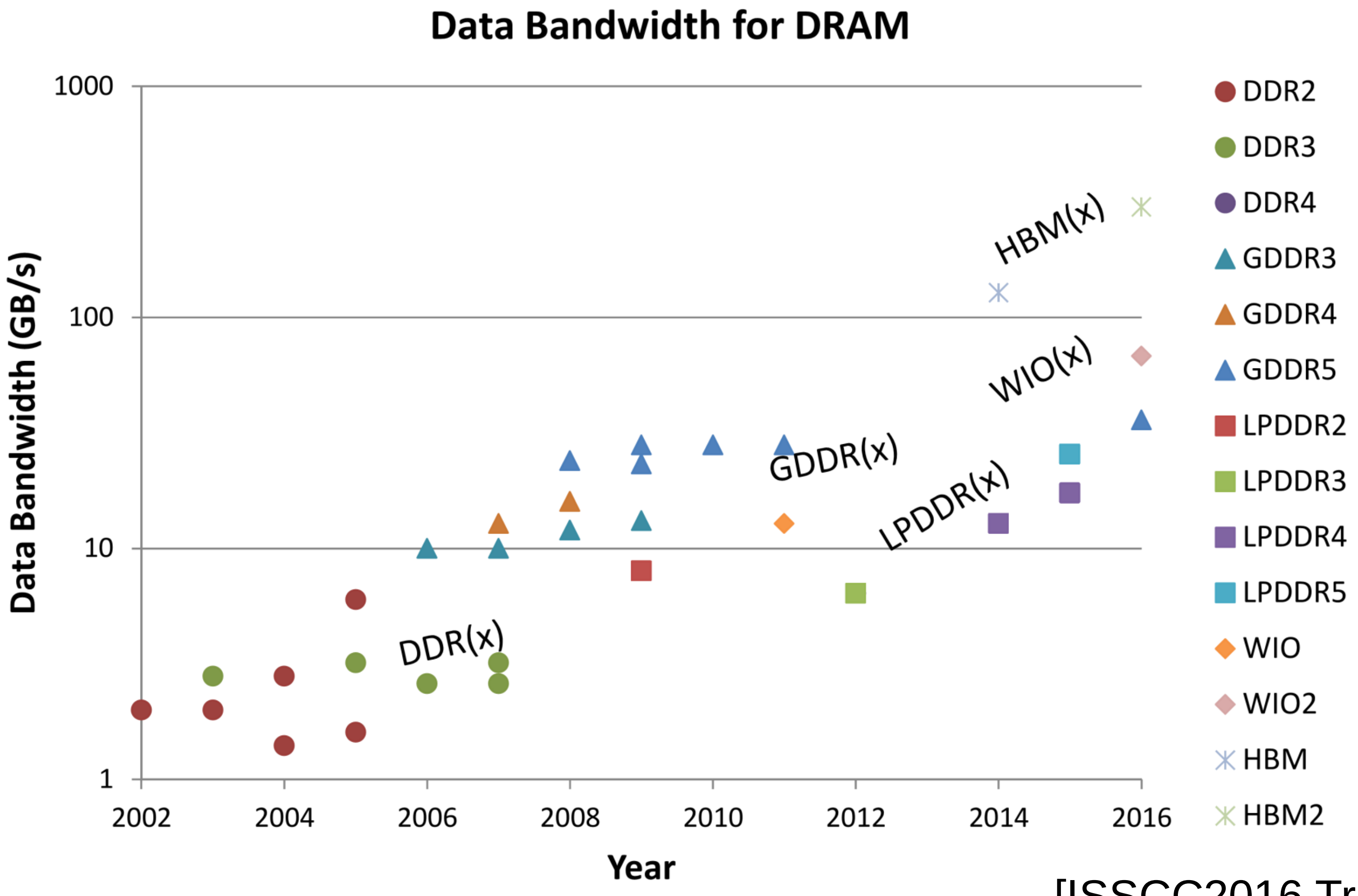
- HBM



GDDR5	Per Package	HBM
32-bit	Bus Width	1024-bit
Up to 1750MHz (7GBps)	Clock Speed	Up to 500MHz (1GBps)
Up to 28GB/s per chip	Bandwidth	>100GB/s per stack
1.5V	Voltage	1.3V

[AMD]

New Standards



[ISSCC2016 Trends]