2012-06-07

2012年 前学期 TOKYO TECH

計算機アーキテクチャ 第一 (E)

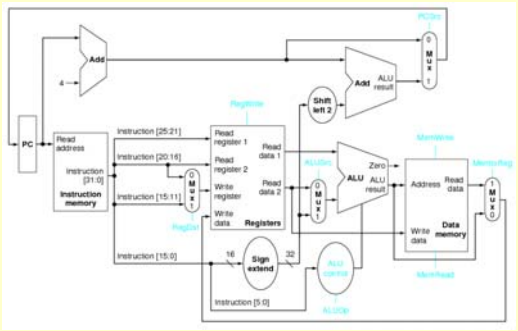
メモリ1：半導体メモリシステム

吉瀬 謙二　計算工学専攻
kise_at_cs.titech.ac.jp
W641講義室　木曜日13:20 － 14:50

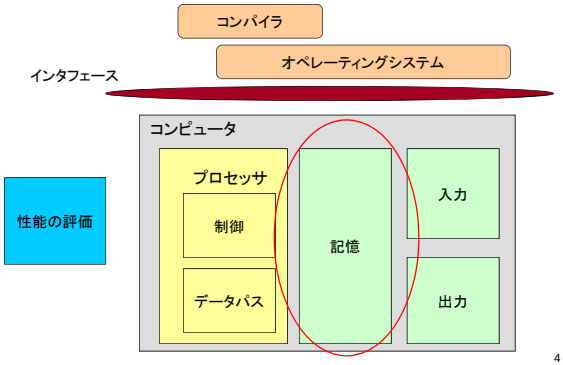
Acknowledgement

- Lecture slides for Computer Organization and Design, Third Edition, courtesy of **Professor Mary Jane Irwin**, Penn State University
- Lecture slides for Computer Organization and Design, third edition, Chapters 1-9, courtesy of **Professor Tod Amon,** Southern Utah University.

プロセッサの**データパス**（シングル・サイクル）



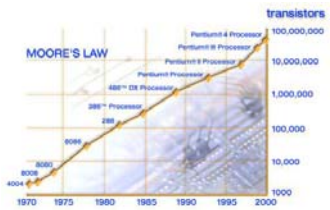
コンピュータ（ハードウェア）の古典的な要素



ムーアの法則によるトランジスタ数の増加

ムーアの法則
チップで利用できるトランジスタの数は2年間で2倍に増加する。

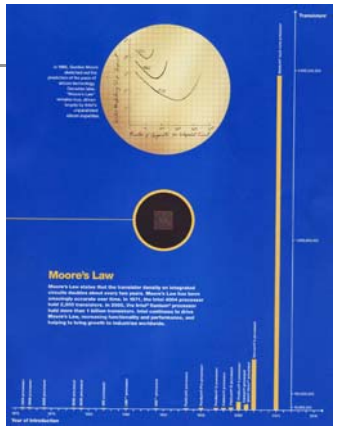
プロセッサ	出荷年	トランジスタ数
4004	1971	2,250
8008	1972	2,500
8080	1974	5,000
8086	1978	29,000
286	1982	120,000
386™ processor	1985	275,000
486™ DX processor	1989	1,180,000
Pentium® processor	1993	3,100,000
Pentium II processor	1997	7,500,000
Pentium III processor	1999	24,000,000
Pentium 4 processor	2000	42,000,000



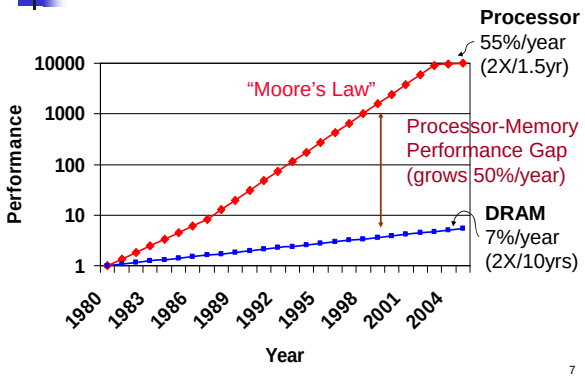
ムーアの法則に従ってトランジスタ数が増加してきた．今後も同様の増加が見込まれる．

出典：Intel社, http://www.intel.com/research/silicon/mooreslaw.htm

Moore's Law



Processor-Memory(DRAM) Performance Gap

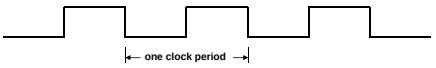


7

Machine Clock Rate

- **Clock rate** (MHz, GHz) is inverse of clock cycle time (**clock period**)

Clock period = 1 / (**clock rate**)



10 nsec clock cycle	=>	100 MHz clock rate
5 nsec clock cycle	=>	200 MHz clock rate
2 nsec clock cycle	=>	500 MHz clock rate
1 nsec clock cycle	=>	1 GHz clock rate
500 psec clock cycle	=>	2 GHz clock rate
250 psec clock cycle	=>	4 GHz clock rate
200 psec clock cycle	=>	5 GHz clock rate

8

Clock Cycles per Instruction, **CPI**

- Not all instructions take the same amount of time to execute

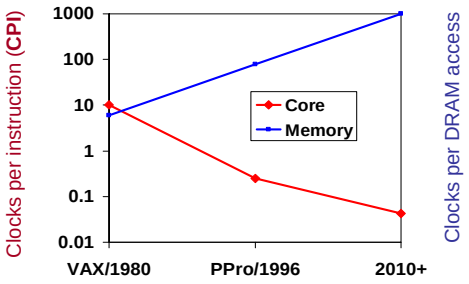
# CPU clock cycles for a program	=	# Instructions for a program	x	Average clock cycles per instruction

- **Clock cycles per instruction** (**CPI**) – the average number of clock cycles each instruction takes to execute
  - CPI = 10.0
  - CPI = 1.0
  - CPI = 0.5
  - CPI = 0.1

9

The "**Memory Wall**"

- **Arithmetic** vs **DRAM** speed gap continues to grow



10

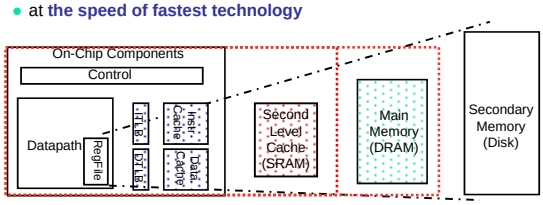
The Memory System **Goal**

- **Fact**:
  Large memories are slow and
  fast memories are small
- How do we create a memory that gives the illusion of being large, cheap and fast ?
  - With **hierarchy**（階層）
  - With **parallelism**（並列性）

11

A Typical Memory **Hierarchy**

- By taking advantage of **the principle of locality**（局所性）
  - Present **much memory** in **the cheapest technology**
  - at **the speed of fastest technology**



Speed (%cycles):	½'s	1's	10's	100's	1,000's
Size (bytes):	100's	K's	10K's	M's	G's to T's
Cost:	highest				lowest

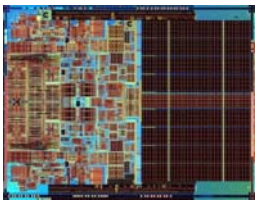
12

DRAM (dynamic random access memory)

13

Cache

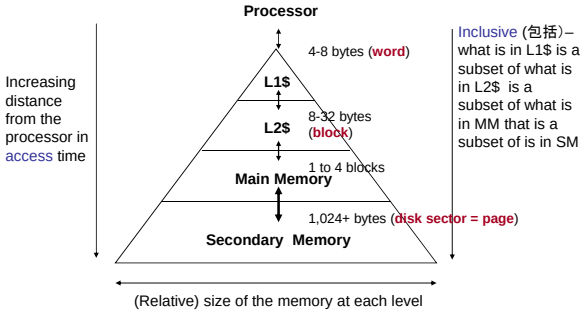
- *Cache memory* consists of a small, fast memory that acts as a buffer for the large memory.
- The nontechnical definition of *cache* is a safe place for hiding things.



Intel Core 2 Duo

14

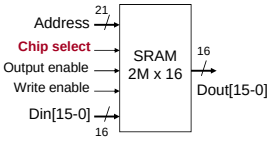
Characteristics of the Memory Hierarchy



15

Memory Hierarchy Technologies

- Caches use SRAM (static random access memory) for speed and technology compatibility
  - **Low density** (**6 transistor cells**), high power, expensive, fast
  - **Static**: content will last "forever" (until power turned off)



- Main Memory uses DRAM for size (density)
  - **High density** (**1 transistor cells**), low power, cheap, slow
  - **Dynamic**: needs to be "refreshed" regularly (~ every 8 ms)
    - 1% to 2% of the active cycles of the DRAM
  - Addresses divided into 2 halves (row and column)
    - **RAS** or Row Access Strobe triggering row decoder
    - **CAS** or Column Access Strobe triggering column selector

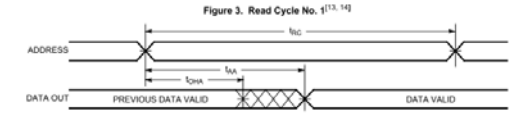
16

非同期式 SRAMメモリ

CYPRESS PERFORM

CY7C1049DV33

Switching Waveforms

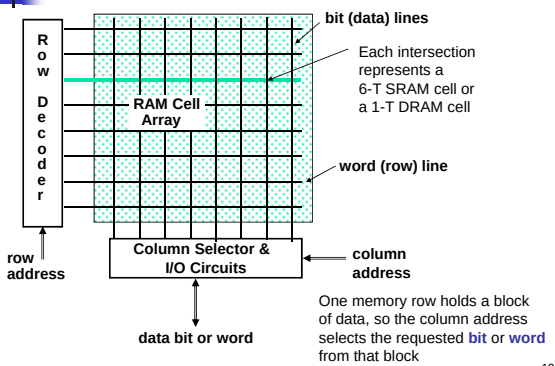


Memory Performance Metrics

- **Latency(レイテンシ, 応答時間)**:
  Time to access one word
  - **Cycle time**: time between requests
  - **Access time**: time between the request and when the data is available (or written)
  - Usually **cycle time** > **access time**
- **Bandwidth(バンド幅, スループット)**:
  How much data from the memory can be supplied to the processor per unit time
  - width of the data channel * the rate at which it can be used

18

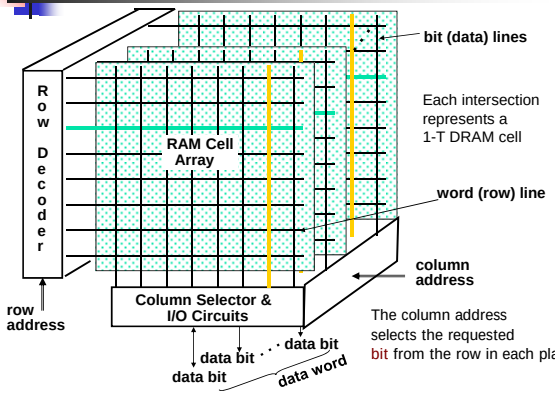
Classical RAM Organization (~Square)



One memory row holds a block of data, so the column address selects the requested **bit** or **word** from that block

19

Classical **DRAM** Organization (~Square Planes)

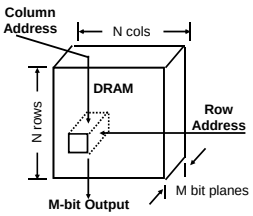


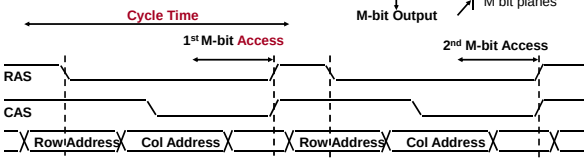
The column address selects the requested bit from the row in each plane

20

Classical **DRAM Operation**

- DRAM Organization:
  - N rows x N column x M-bit
  - Read or Write M-bit at a time
  - Each M-bit access requires a RAS / CAS cycle





21

アナウンス

- 講義スライドおよびスケジュール
  - www.arch.cs.titech.ac.jp
  - 講義日程が変更になることがあるので頻繁に確認すること.

22