コンピュータ

【電子計算機の理論設計序説】

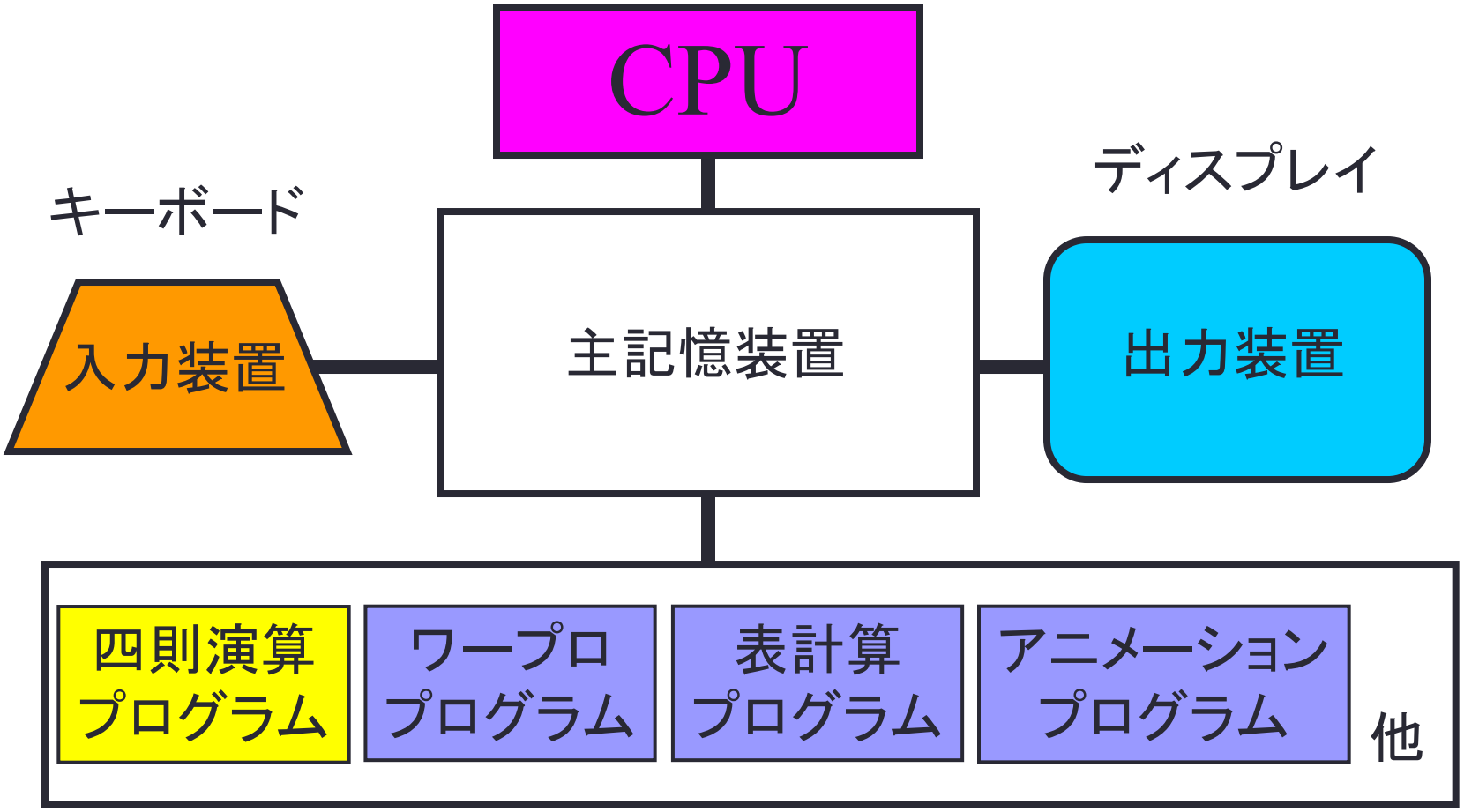
1945年にフォン・ノイマンが発表したコンピュータの動作方式
今日のコンピュータの基礎

・**電子式，２進数，ディジタル**

・**プログラム内蔵方式**
　あらかじめプログラムとデータを主記憶装置に格納し、
　制御装置がプログラム中の命令を一つずつ取り出しては
　処理していく方式
　プログラム記憶（Stored Program）方式とも呼ばれる

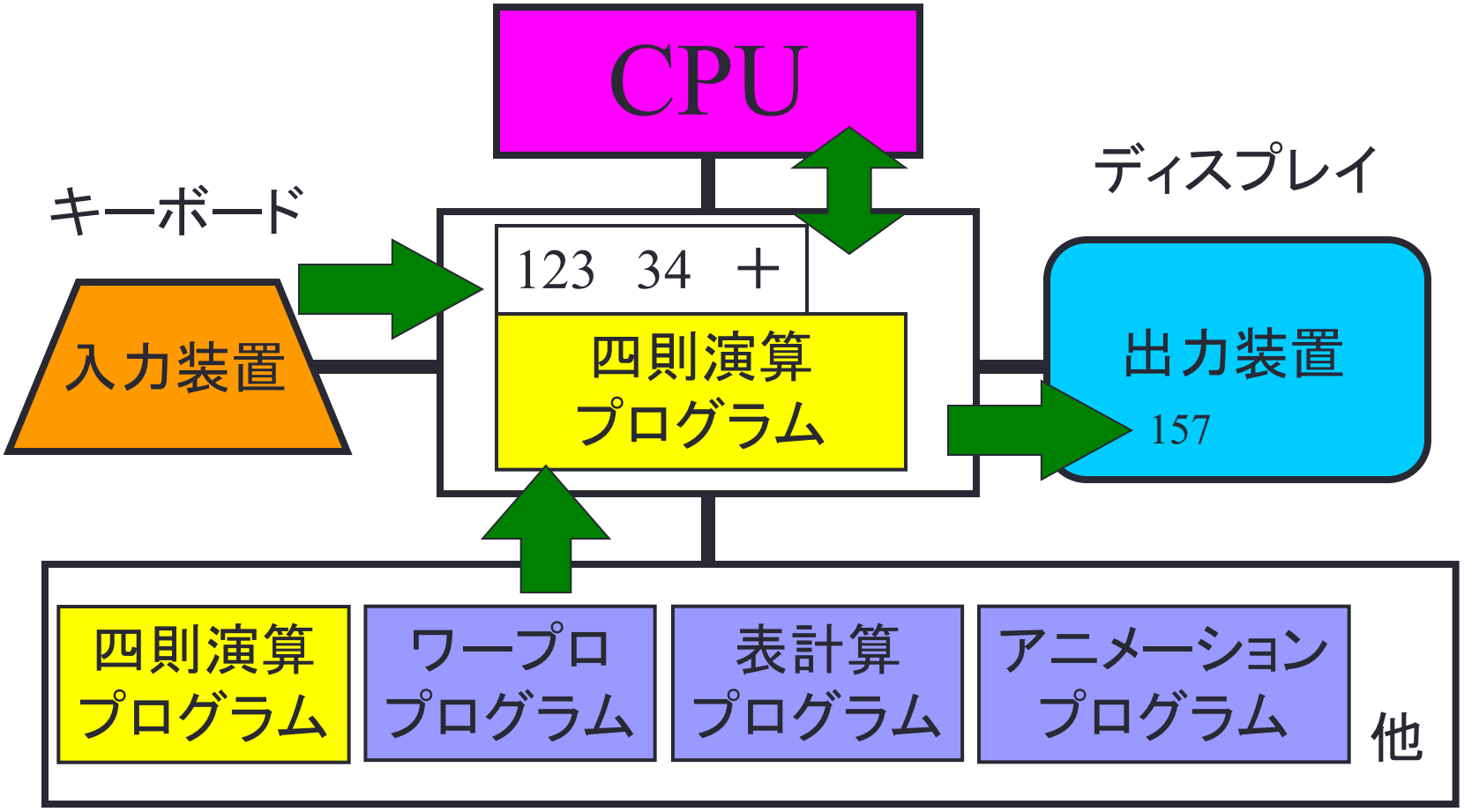
・**逐次処理方式**
　記憶されたプログラムを順番に取り出して一つ一つ実行する
　逐次制御（Serial Control）方式とも呼ばれる

プログラム内蔵方式



補助記憶装置：　ハードディスク

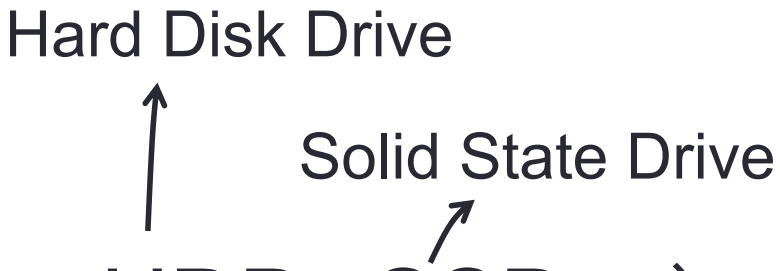
逐次処理方式



補助記憶装置： ハードディスク

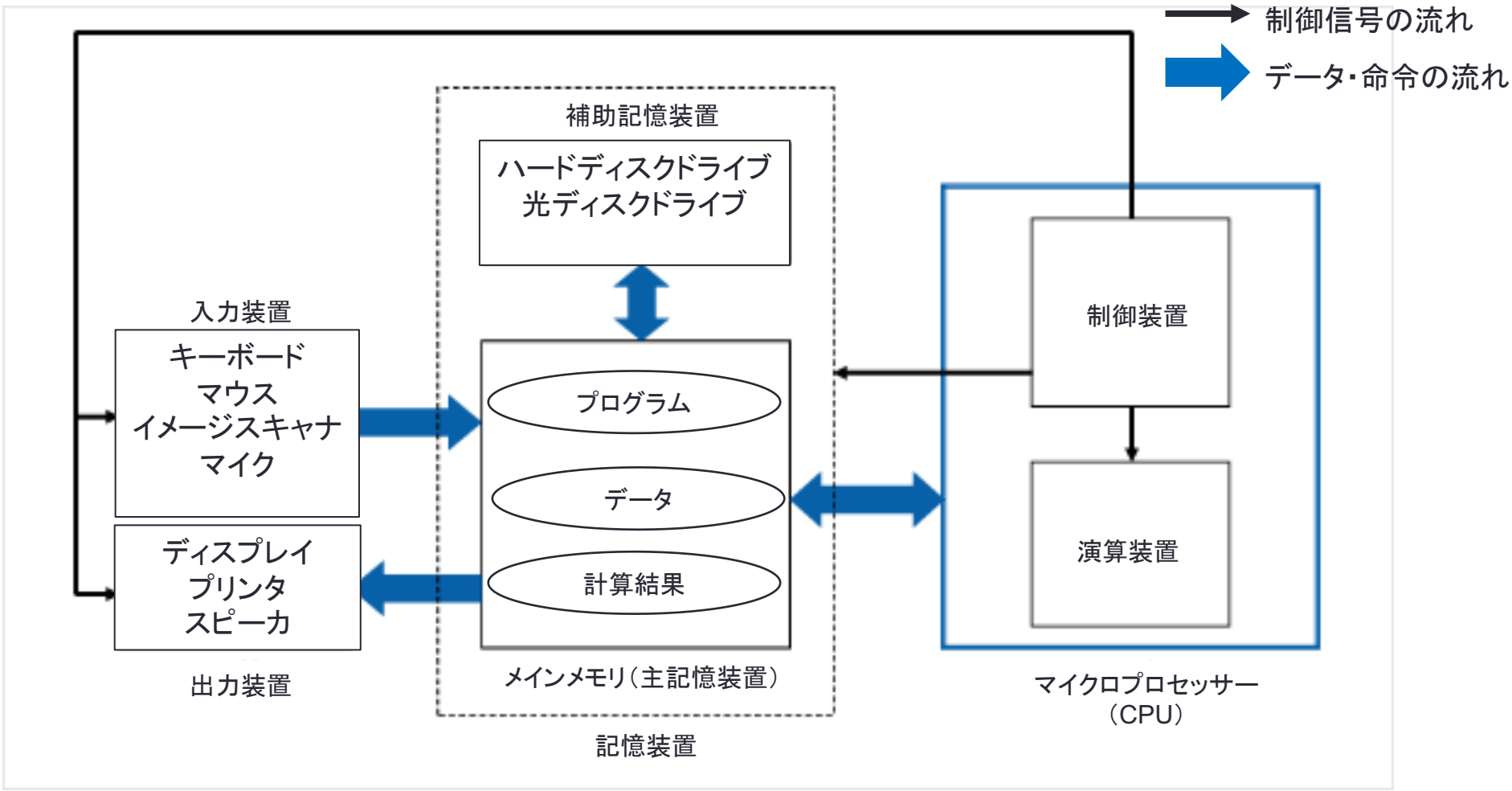
コンピュータの５大機能

- ＿＿＿装置　（CPU）
- ＿＿＿装置　（CPU）
- ＿＿＿装置　（Memory，HDD，SSD…）
- ＿＿＿装置　（Keyboard，mouse…）
- ＿＿＿装置　（Monitor，Printer…）



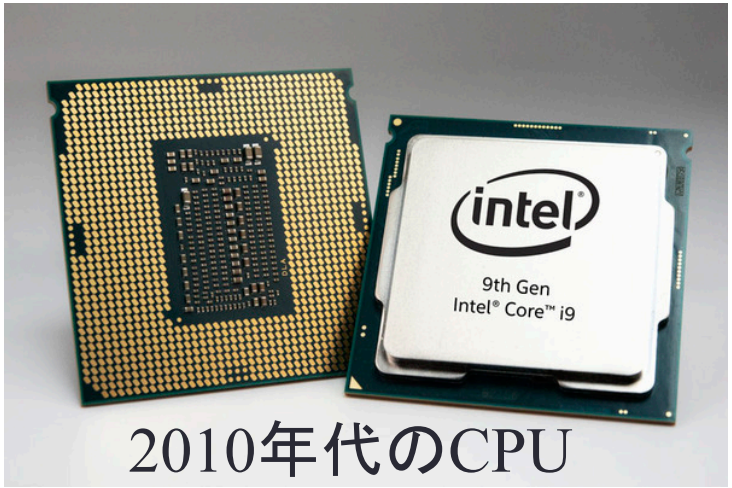
パソコンなどでは、制御装置と演算装置を一体化した
中央（演算）処理装置（CPU：Central Processing Unit）
が使われている.
マイクロプロセッサー（MPU：Micro Processing Unit）
とも呼ばれる.

コンピュータの構成



各装置は＿＿＿＿（データが流れる通路）と呼ばれる信号線で結ばれている
コンピュータは命令（プログラム）やデータを記憶装置に記録しており、制御装置や演算装置がそれを読み込んで処理を行う

中央演算処理装置：CPUの外観



2010年代のCPU


1980年代初期のCPU

【キーワード】
- ○○bit CPU
- クロック周波数
  ○○MHz，○○GHz

1990年代のCPU

- ✓ 2024年時の最速CPU
  Intel Core i9-14900K（第14世代）
  最大周波数　6 GHz（60億/秒）
  キャッシュ 36 MB

CPUの性能

現在は、64ビットCPUが主流：2^{64} ≒ 1800京

CPUのビット数： *一度の命令（信号）で行える演算処理のビット数！*

32ビットCPUの場合
計算回数：1回
一度に送れるデータは
42億種類（2^{32}）

```
  01000001011010011001111111010011
+ 00101111001011011110010101100100
--------------------------------
  01110000100101111000010100110111
```

8ビットCPUの場合
計算回数：4回
一度に送れるデータは
256種類

```
  01000001  01101001  10011111  11010011
+ 00101111  00101101  11100101  01100100
-------------------------------------------
  01110000  10010111  10000101  00110111
```

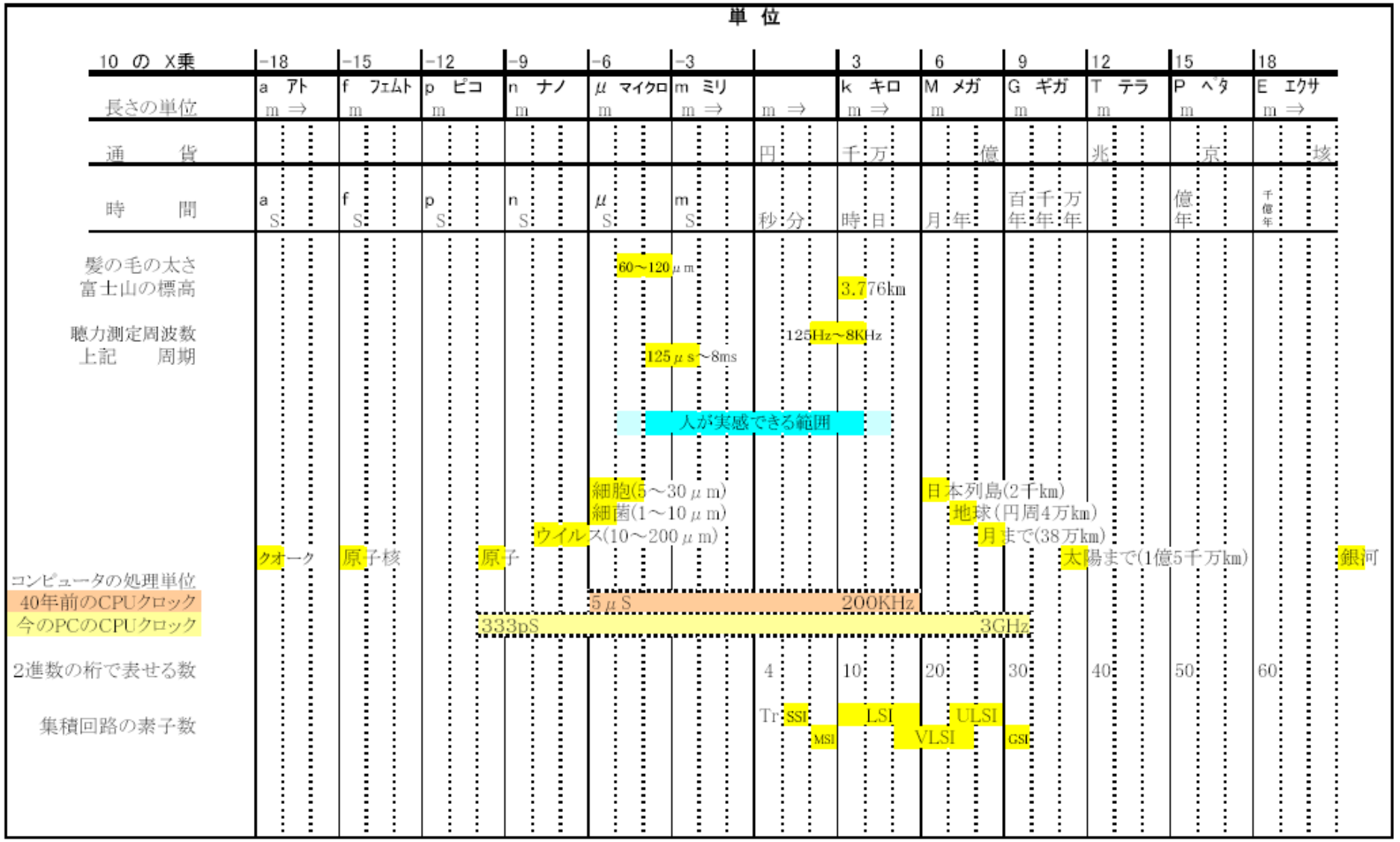
10進数表示：559,632,177

CPUのクロック周波数： *1秒間に発生できる信号の数！*

1GHz：1秒間の動作は10億回

数	指数	記号	読み	日本語
1,000,000,000,000,000	$10^{15} \rightarrow 2^{50}$	P	ペタ	1000兆
1,000,000,000,000	$10^{12} \rightarrow 2^{40}$	T	テラ	1兆
1,000,000,000	$10^{9} \rightarrow 2^{30}$	G	ギガ	10億
1,000,000	$10^{6} \rightarrow 2^{20}$	M	メガ	100万
1,000	$10^{3} \rightarrow 2^{10}$	k	キロ	千
10	10	da	デカ	十
1	10^{0}	-	-	一
0,1	10^{-1}	d	デシ	十分の一
0.01	10^{-2}	c	センチ	百分の一
0,001	10^{-3}	m	ミリ	千分の一
0,000001	10^{-6}	μ	マイクロ	百万分の一
0,000000001	10^{-9}	n	ナノ	10億分の一
0,000000000001	10^{-12}	p	ピコ	1兆分の一

いろんな単位


単 位
10 の X乗
-18
-15
-12
-9
-6
-3
3
6
9
12
15
18
長さの単位
a アト m ⇒
f フェムト m
p ピコ m
n ナノ m
μ マイクロ m
m ミリ m ⇒
m ⇒
k キロ m ⇒
M メガ m
G ギガ m
T テラ m
P ペタ m
E エクサ m ⇒
通 貨
円
千 万
億
兆
京
垓
時 間
a S
f S
p S
n S
μ S
m S
秒 分
時 日
月 年
百年 千年 万年
億年
千億年
髪の毛の太さ
富士山の標高
聴力測定周波数
上記 周期
60～120μm
3.776km
125Hz～8KHz
125μs～8ms
人が実感できる範囲
細胞(5～30μm)
細菌(1～10μm)
ウイルス(10～200μm)
日本列島(2千km)
地球(円周4万km)
月まで(38万km)
太陽まで(1億5千万km)
クオーク
原子核
原子
銀河
コンピュータの処理単位
40年前のCPUクロック
今のPCのCPUクロック
5μS
200KHz
333pS
3GHz
2進数の桁で表せる数
4
10
20
30
40
50
60
集積回路の素子数
Tr
SSI
MSI
LSI
VLSI
ULSI
GSI

計算能力の進歩

1970年代後期のCPU
インテル製　→　8 bit, 200 kHz

2003年のCPU
Pentium4　→　32 bit, 3.0 GHz

性能の向上率（単純計算）
4×15000＝60000倍の性能UP

コンピュータ全体ではさらに
すさまじく性能が向上している.
（数百万～数千万倍）
- バスの周波数
- 搭載メモリーの増大　等

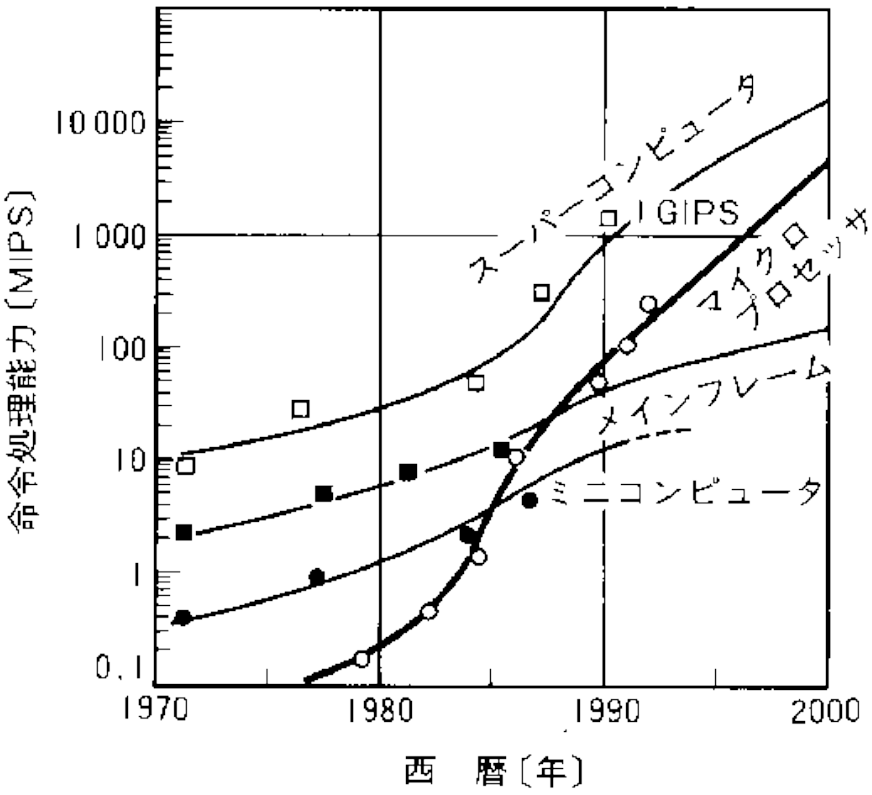


図 1　　各種コンピュータの命令処理能力の向上

2014年CPU
Intel® Core™ i7-4790 Processor
→　64 bit, 4.0 GHz, 4 core

Intel Core i9-14900K

コアは、1 個のコンピューティング・コンポーネント (ダイまたはチップ) に含まれる独立したCPUの数を示すハードウェア用語です。

プロセッサーが動作可能な最大のシングルコア周波数のことです。

CPU キャッシュは、プロセッサーに配置された高速メモリー領域です。インテル® スマートキャッシュとは、ラスト・レベル・キャッシュへのアクセスをすべてのコアで動的に共有できるアーキテクチャーのことです。

Intelウェブサイトより抜粋

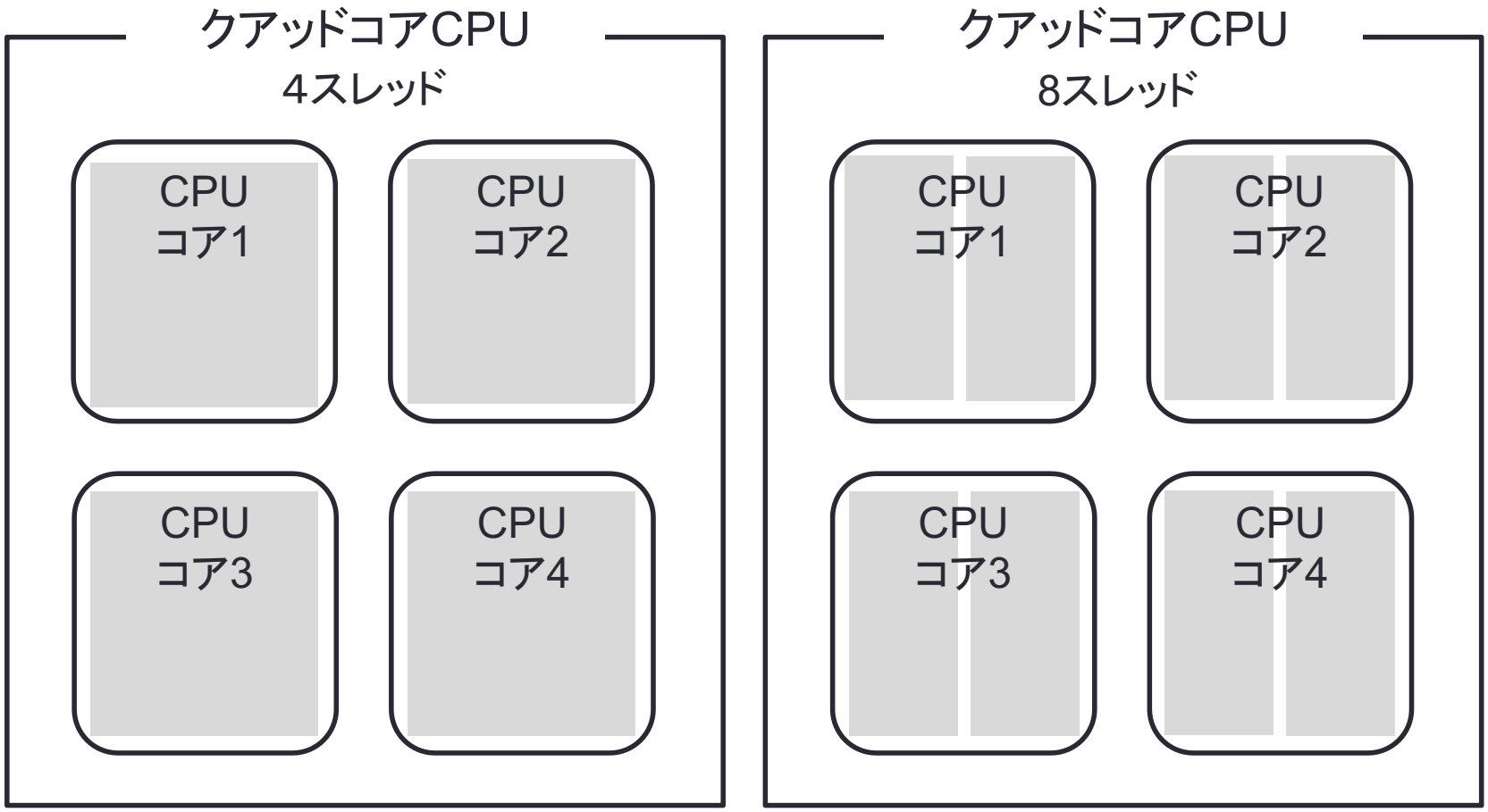
基本仕様

製品コレクション	Intel® Core™ i9 Processors (14th gen)
開発コード名	Products formerly Raptor Lake
システムの種類	Desktop
プロセッサー・ナンバー	i9-14900K
リソグラフィー	Intel 7
希望カスタマー価格	$589.00-$599.00

CPU の仕様

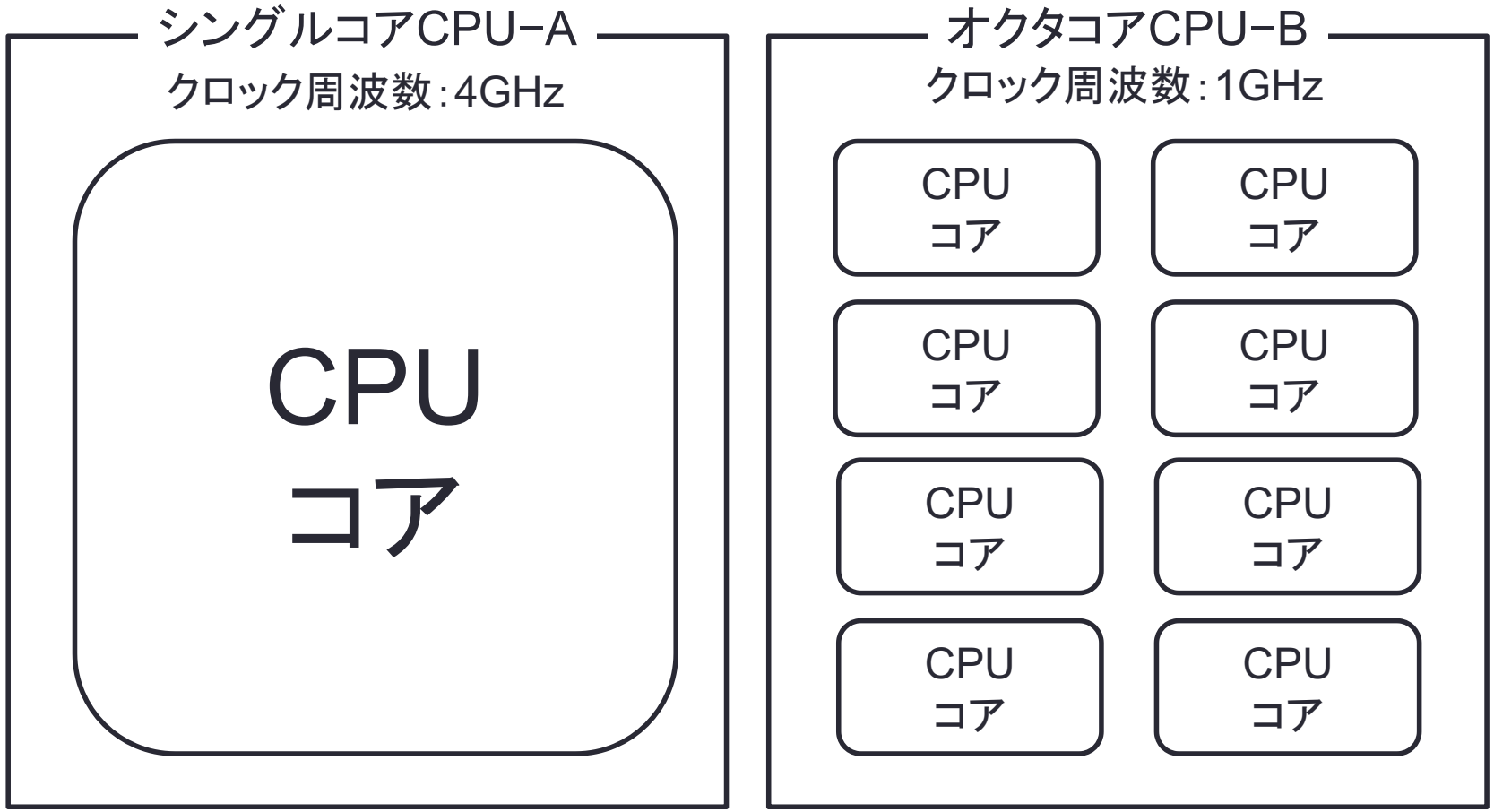
コアの数	24
Performance-coresの数	8
Efficient-coresの数	16
スレッド総数	32
ターボ・ブースト利用時の最大周波数	6 GHz
キャッシュ	36 MB Intel® Smart Cache
合計 L2 キャッシュ	32 MB

CPUのコアとスレッド



スレッド：仮想コア（物理的には存在せずプログラムでのみ認識）
並列に処理できる数（CPUに余裕があるとき）
＿＿＿＿＿＿＿＿，論理コアとも言う

CPUの処理速度



2GHzで動く1コアのCPUで10秒かかる処理　（仮定：コア数＝スレッド数）

- 並列化できない処理の場合　→　CPU−A：＿＿、CPU−B：＿＿
- 並列化できる処理の場合　→　CPU−A：＿＿、CPU−B：＿＿

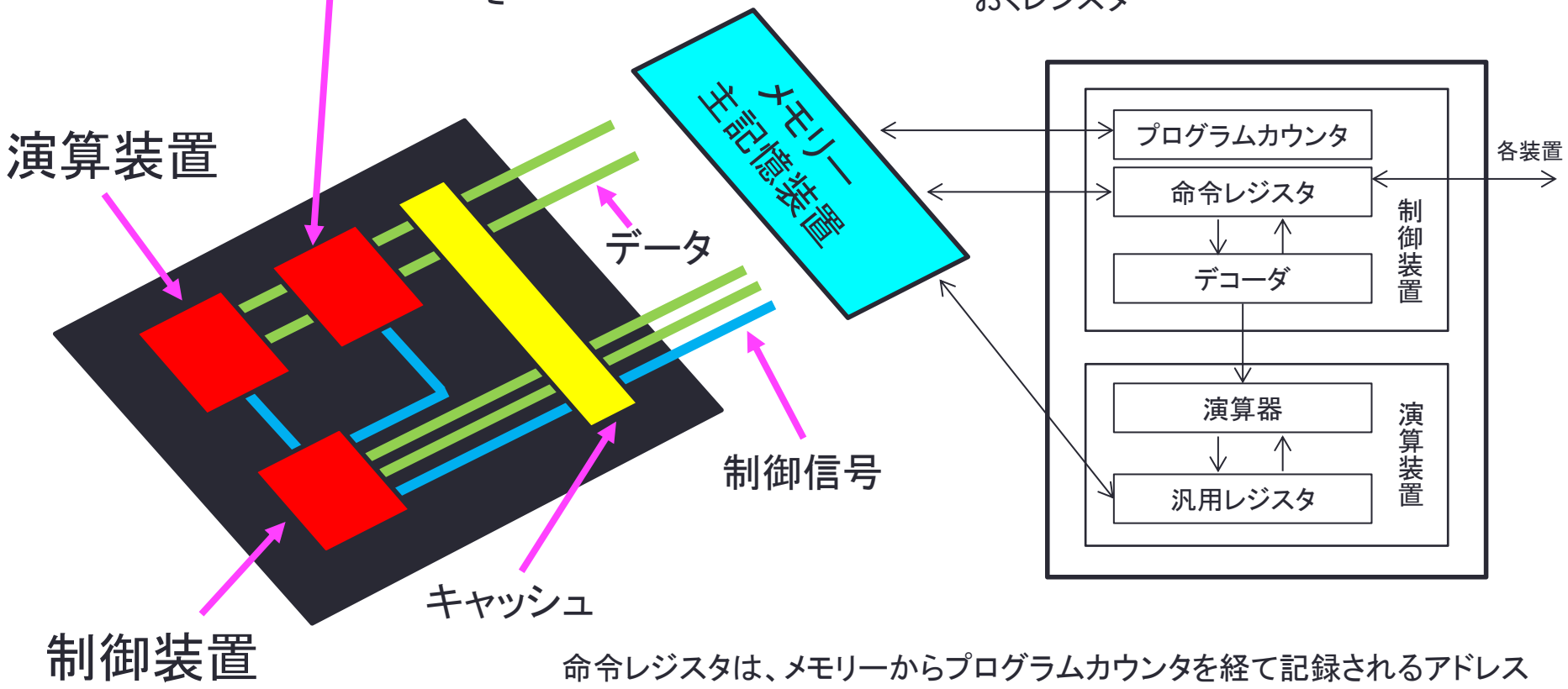
CPUの構成

- プログラムカウンタ
- 命令レジスタ
- 汎用レジスタ

次に実行する命令を取り出すためのアドレスを記憶しておくレジスタ

取り出した命令を一時的に記憶しておくレジスタ

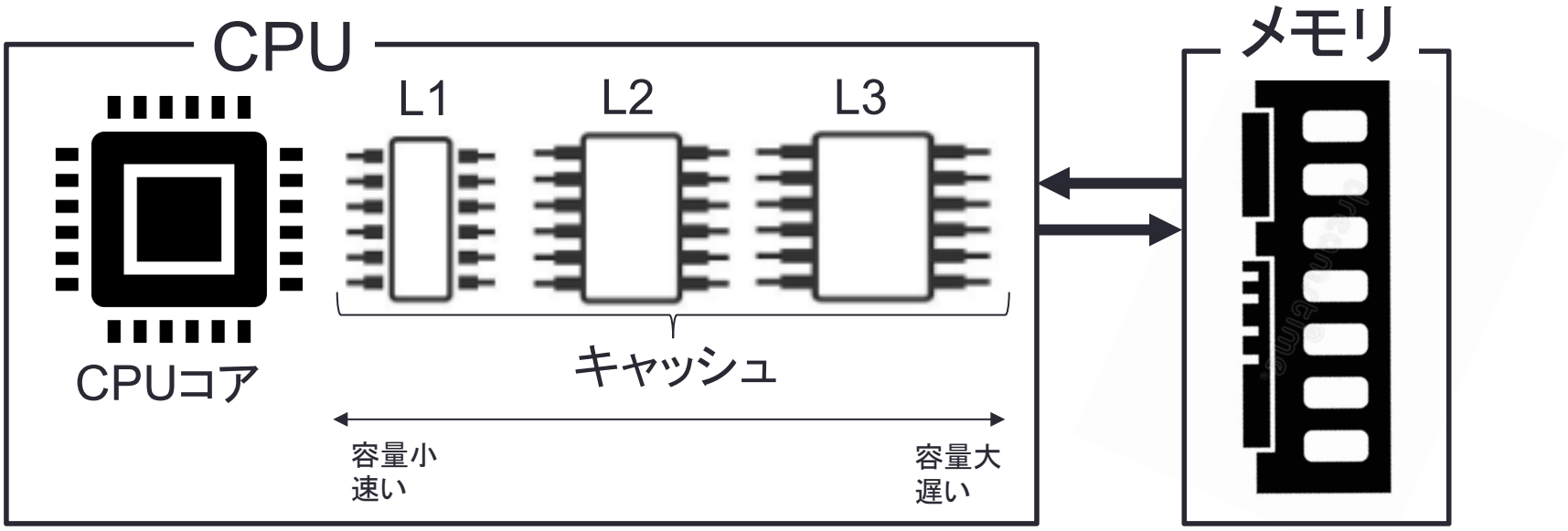
演算結果やアドレスなどを格納しておくレジスタ



命令レジスタは、メモリーからプログラムカウンタを経て記録されるアドレス（命令が置かれている場所）から命令のバイト列を取得する　→ フェッチ

取得した命令のバイト列をデコーダでCPUに意味のある機械語に翻訳（デコード）して演算装置で計算・各装置で制御する

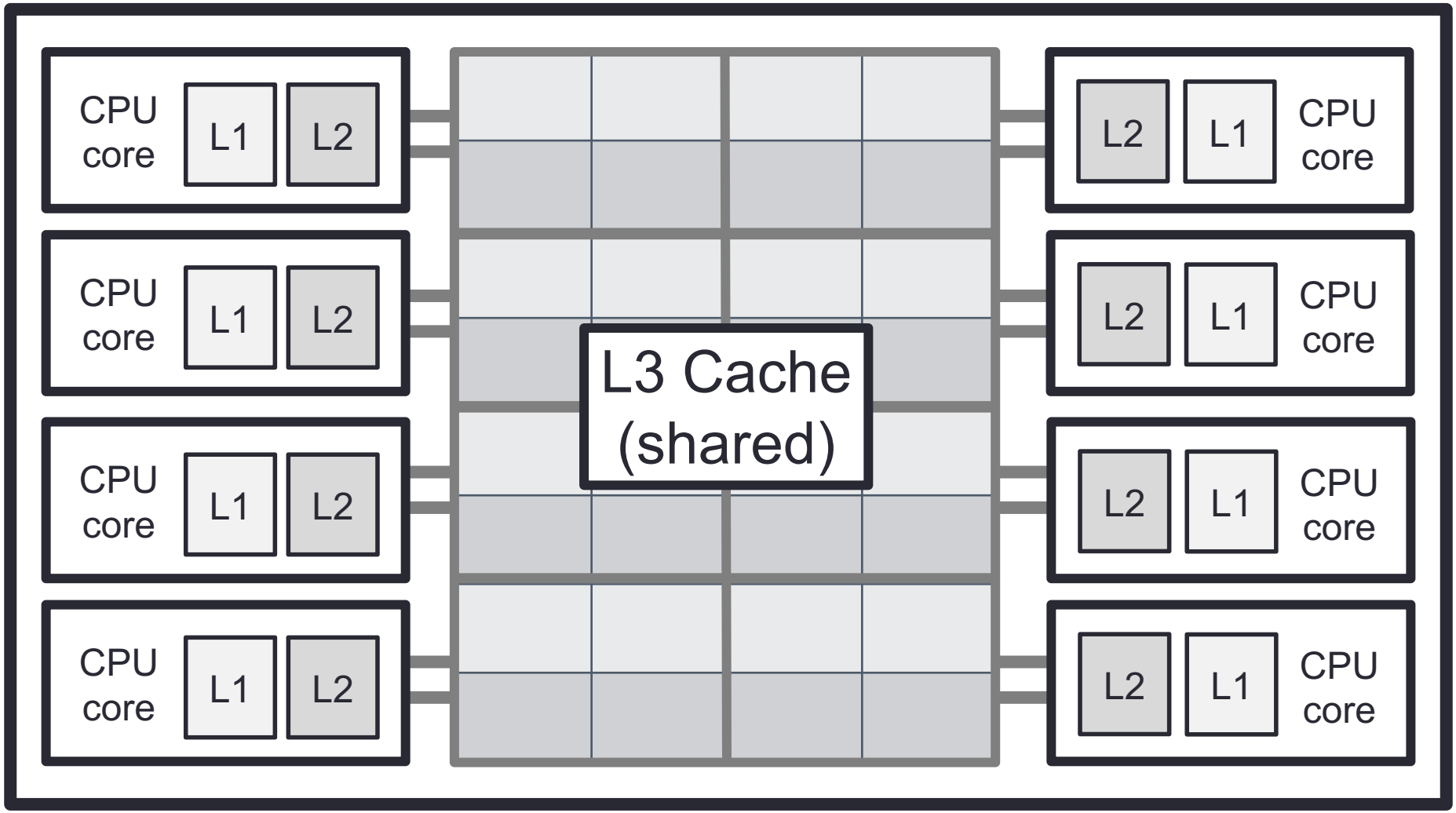
キャッシュ

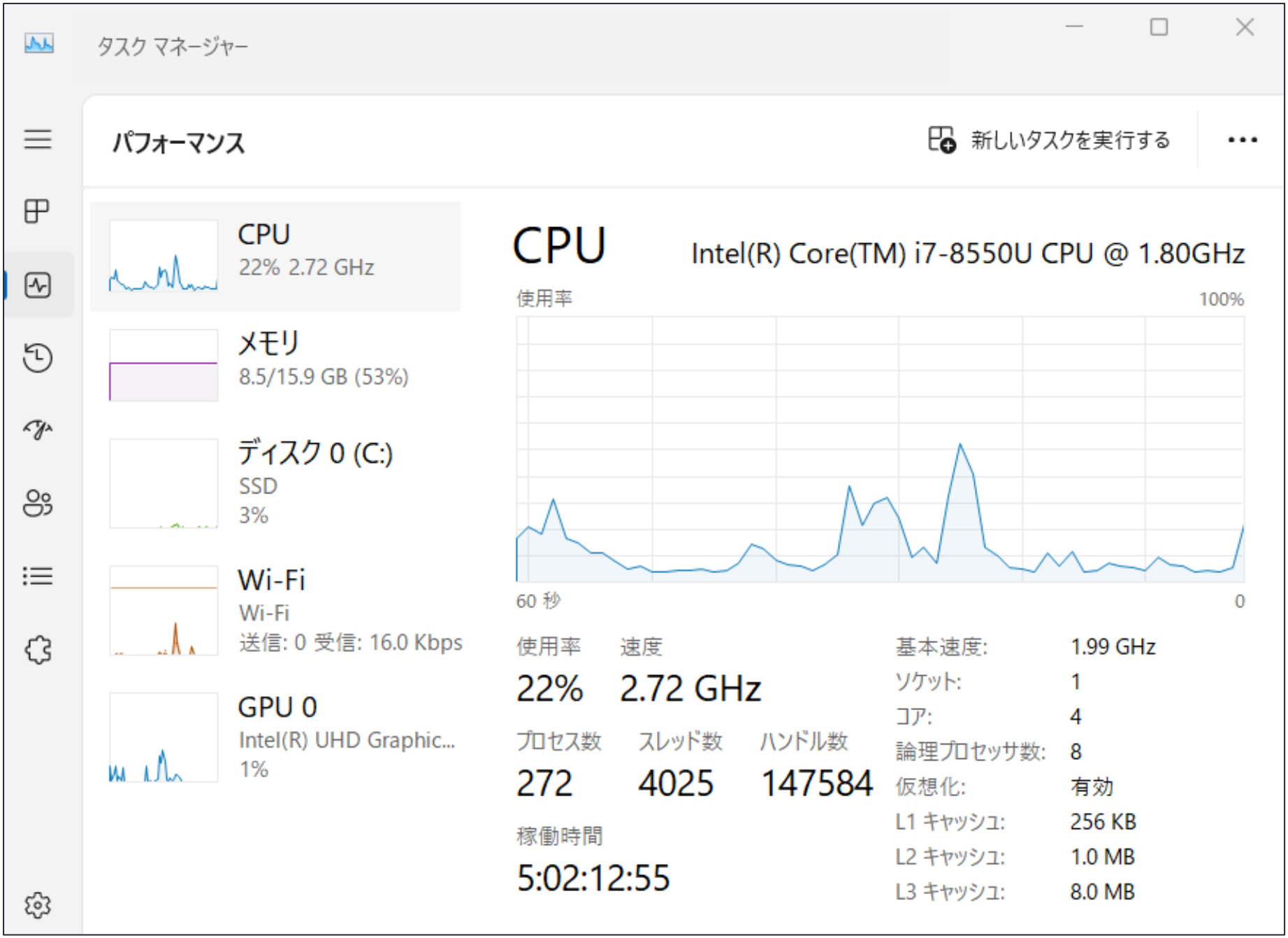


- CPU内部に配置される高速なメモリ
- CPUが頻繁にアクセスするデータや命令を一時的に保存
- データの読み書きにかかる時間が短縮され全体的な処理速度が向上
- CPUコアからの距離に応じた階層構造（L1，L2，L3）

	速度	容量
L1		
L2		
L3		

キャッシュ:CPU内の構造

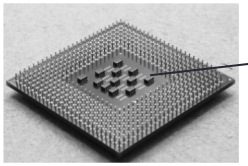



タスク マネージャー
パフォーマンス
新しいタスクを実行する
CPU
22% 2.72 GHz
メモリ
8.5/15.9 GB (53%)
ディスク 0 (C:)
SSD
3%
Wi-Fi
Wi-Fi
送信: 0 受信: 16.0 Kbps
GPU 0
Intel(R) UHD Graphic...
1%
CPU
Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz
使用率
100%
60 秒
0
使用率
速度
22%
2.72 GHz
プロセス数
スレッド数
ハンドル数
272
4025
147584
稼働時間
5:02:12:55
基本速度: 1.99 GHz
ソケット: 1
コア: 4
論理プロセッサ数: 8
仮想化: 有効
L1 キャッシュ: 256 KB
L2 キャッシュ: 1.0 MB
L3 キャッシュ: 8.0 MB

記憶装置

＿＿＿＿記憶装置

- ✓ 電子回路で構成され、情報の保持に電力が必要
- ✓ 電子の状態によって情報を保持
- ✓ 読み書きが速い



CPUの
キャッシュ，
レジスタ

SRAM
（Static RAM）

DRAM（Dynamic RAM）

＿＿＿＿記憶装置

- ✓ 電力がなくても情報を保持し続けることができる
- ✓ 情報の保持に物理的状態を利用している
- ✓ 読み書きには物理状態の読み取りや変化が必要
- ✓ 読み書きが遅い

HDD
（~16TB）

光ディスク
DVD（8.5GB）
CD（700MB）
Blue-ray disk
（128GB）

Read
Only
Memory

浮遊ゲートMOSFETと呼ばれる半導体素子を利用してデータ記録を行う不揮発性メモリの総称

フラッシュメモリ

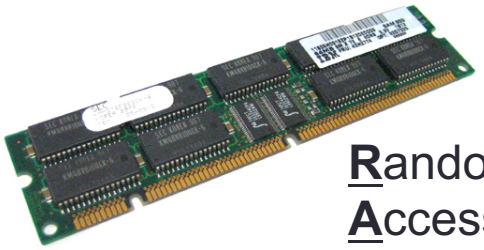
Universal
Serial
Bus （~1TB）
Memory

S（~~secure~~）
D（~~digital~~）
Card （~2TB）
Memory

SSD
（~16TB）

フラッシュメモリを
集積して構成

記憶装置



演算結果

演算・制御 装置

プログラムやデータ

- ＿＿記憶装置：半導体メモリ（揮発性）

Random
Access
Memory

- ○読み出し
- ○書き込み

演算結果

プログラムやデータ

- ＿＿記憶装置：HDD, 光ディスク, 半導体メモリ（不揮発性）

Digital Versatile Disk: DVD

Hard Disk Drive: HDD

Solid State Drive：SSD

主記憶装置

- ＿＿＿＿（Random Access Memory）　読み書き両用
  電源の供給を絶つと記憶内容が失われる揮発性メモリ

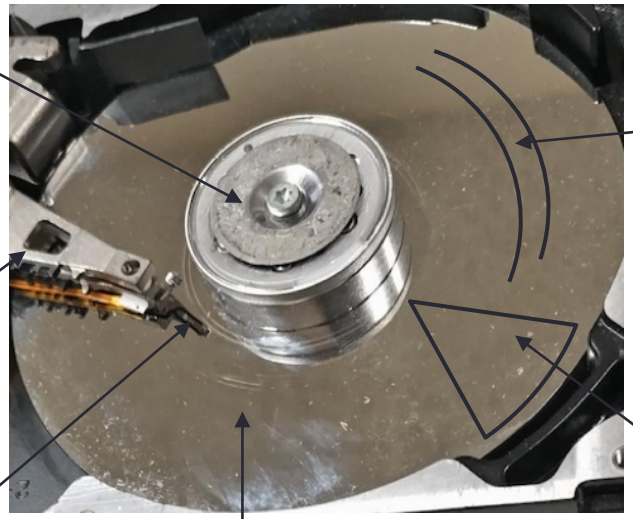
- **SRAM**（Static RAM）　→　高速，電源供給中は記憶内容を保持
  （CPU内のレジスタやキャッシュ）
- **DRAM**（Dynamic RAM）　→　集積度が高い，現在の主流
  一定時間経つと記憶内容が消える
  - **SDRAM**（Synchronous DRAM）　システムバスに同期して動作することを明示した名称

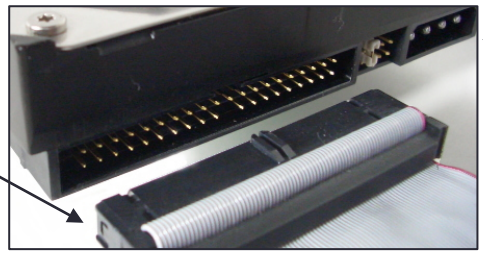
- SIMM（Single Inline Memory Module）→ 32ビット幅のメモリアクセス
- **DIMM**（Dual Inline Memory Module）　→ 64ビット幅のメモリアクセス
  現在の主流
  - DDR4-SDRAM-DIMM

補助記憶装置

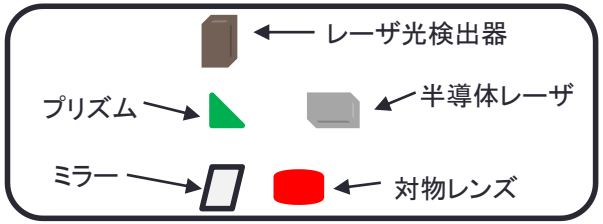
ハードディスク（HDD）

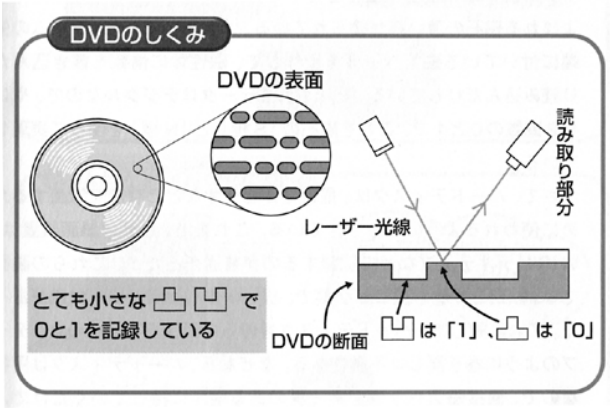
多数の小さな磁石を並べて、その磁力の向きで0と1の並びを表現している → 半永久的に記録可能



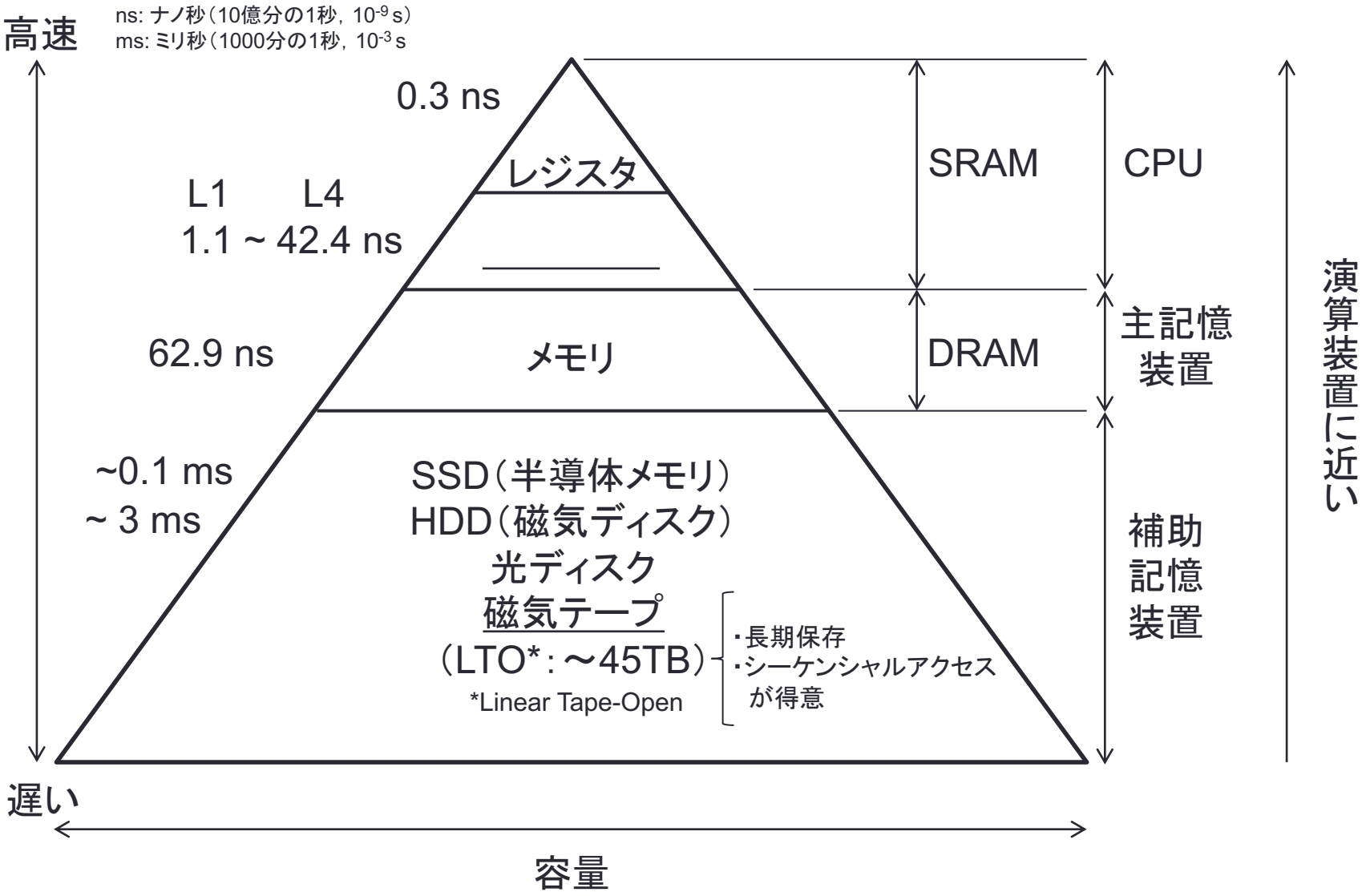


DVD ディジタル多用途ディスク（Digital Versatile Disk）

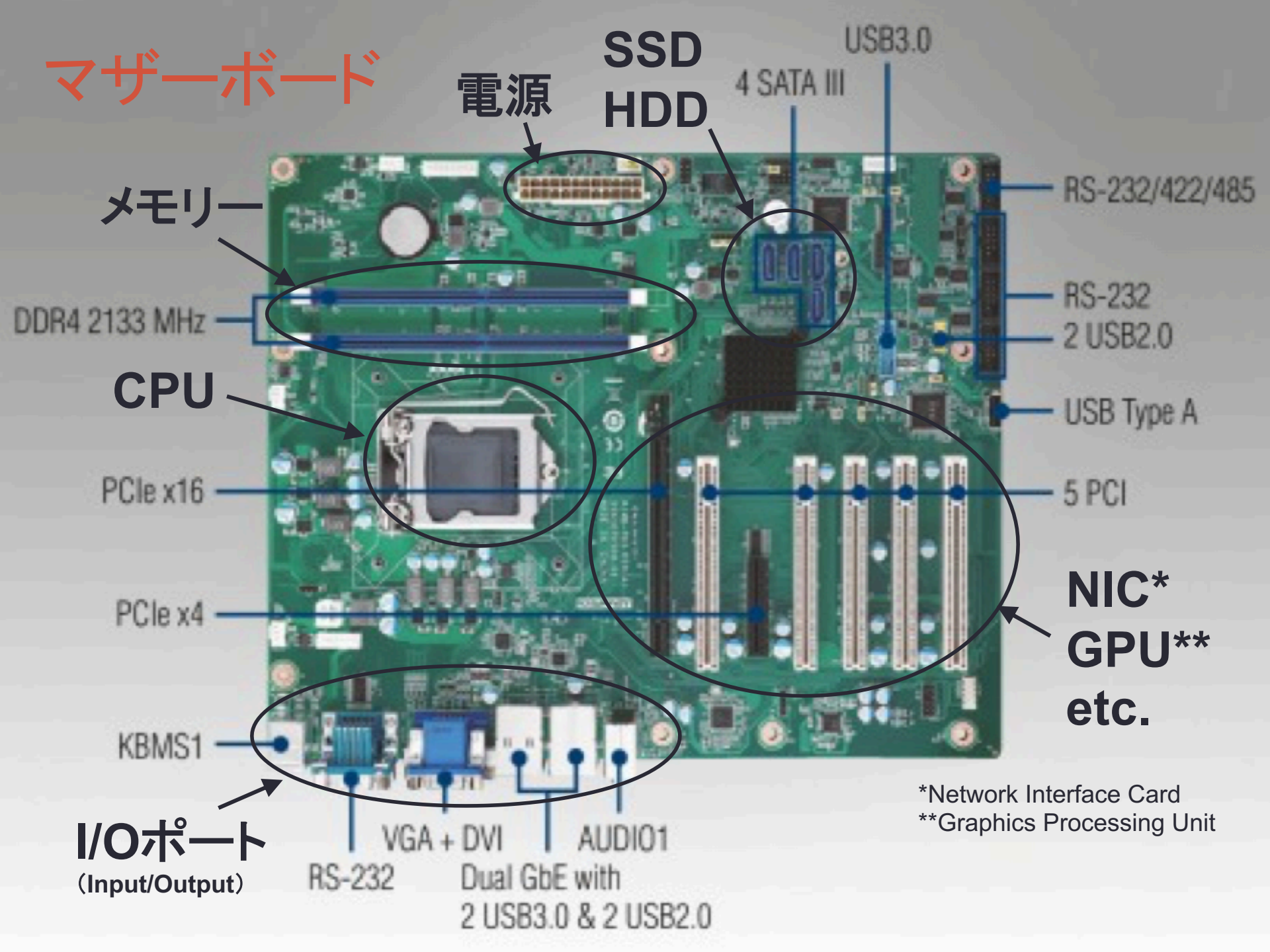




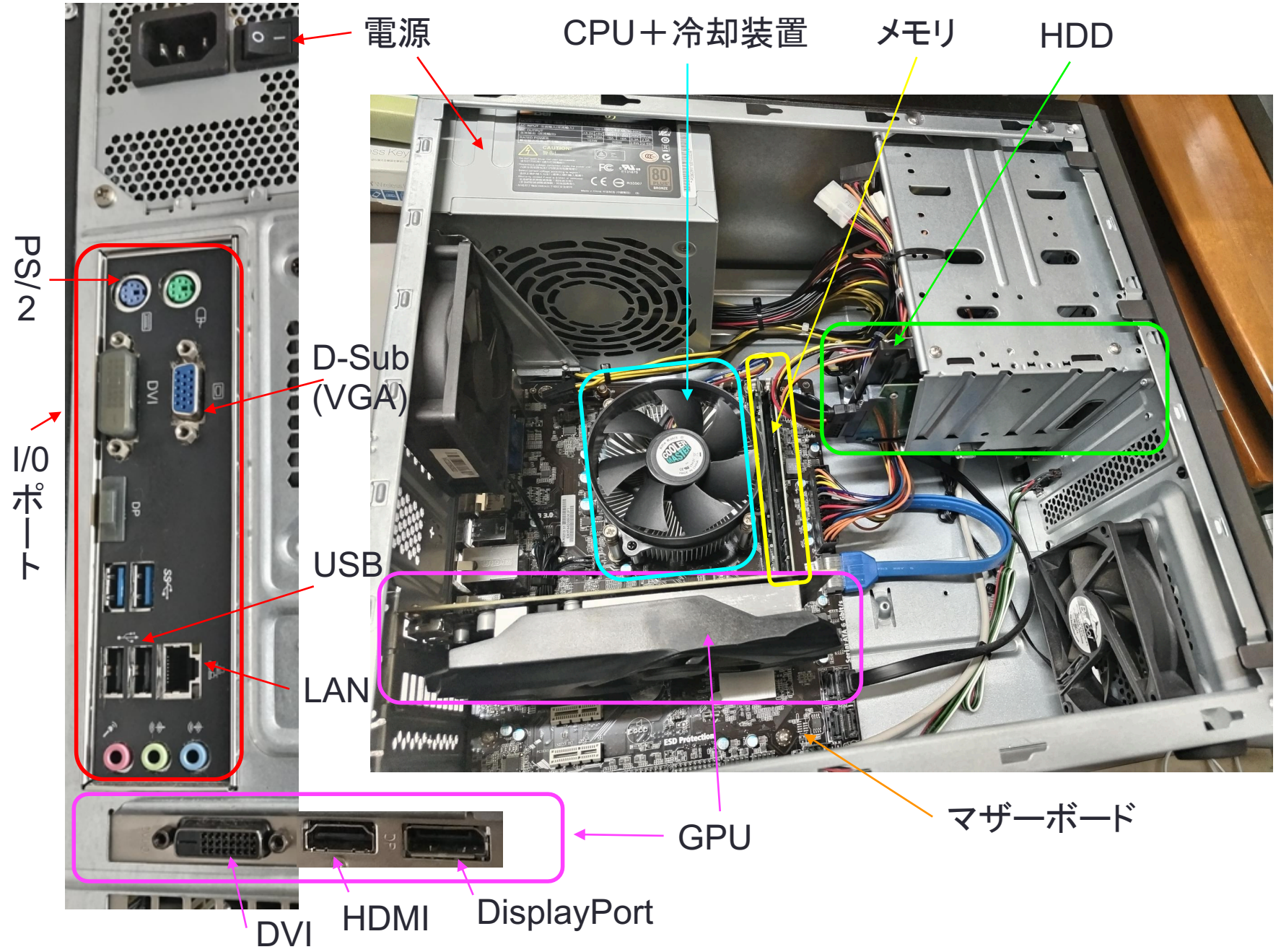
記憶装置の速さと容量


高速
ns: ナノ秒（10億分の1秒，10^{-9} s）
ms: ミリ秒（1000分の1秒，10^{-3} s）
0.3 ns
レジスタ
L1 L4
1.1 ~ 42.4 ns
62.9 ns
メモリ
~0.1 ms
~ 3 ms
SSD（半導体メモリ）
HDD（磁気ディスク）
光ディスク
磁気テープ
（LTO*：～45TB）
*Linear Tape-Open
・長期保存
・シーケンシャルアクセス
が得意
SRAM
CPU
DRAM
主記憶
装置
補助
記憶
装置
演算装置に近い
遅い
容量

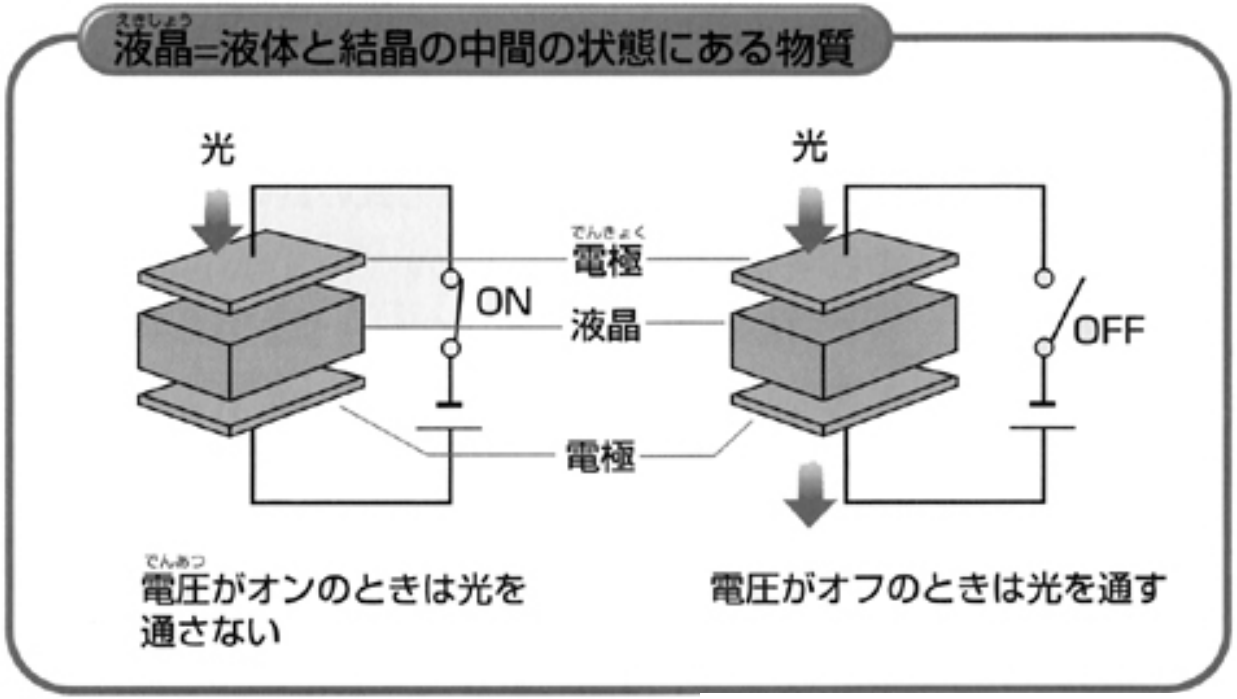
マザーボード

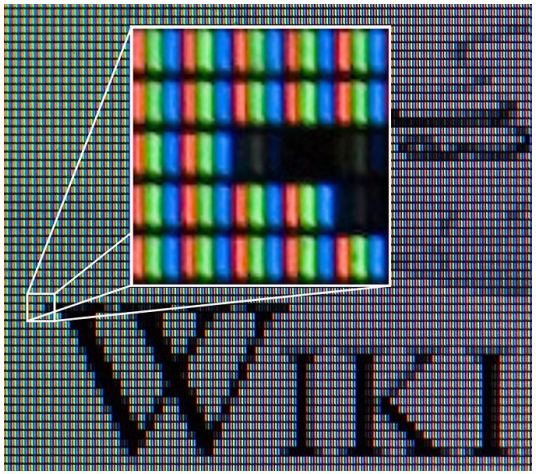


*Network Interface Card
**Graphics Processing Unit

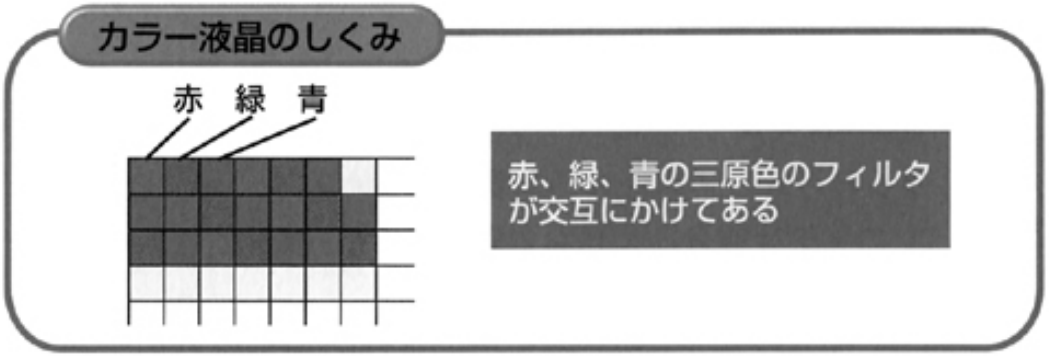

電源
CPU＋冷却装置
メモリ
HDD
PS/2
D-Sub
(VGA)
I/O
ポート
USB
LAN
GPU
マザーボード
DVI
HDMI
DisplayPort

液晶ディスプレイ　LCD:＿＿＿＿＿＿＿＿

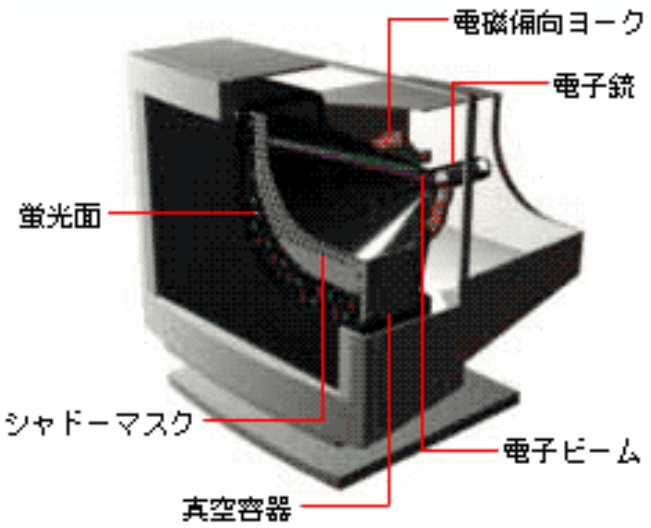


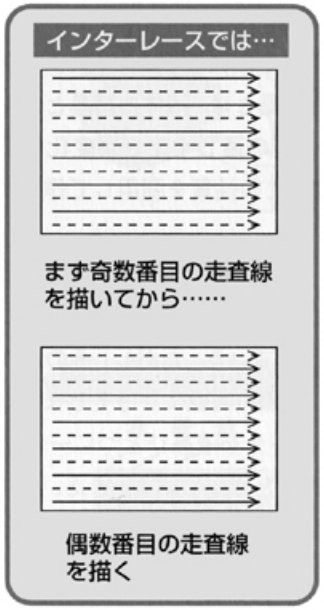


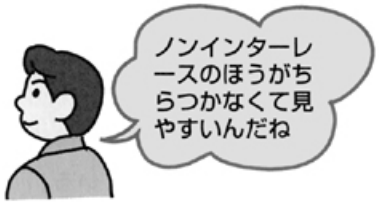
- スペースをとらない
- 消費電力が低い
- 目が疲れにくい



ディスプレイ　Cathode Ray Tube （陰極線管 ）







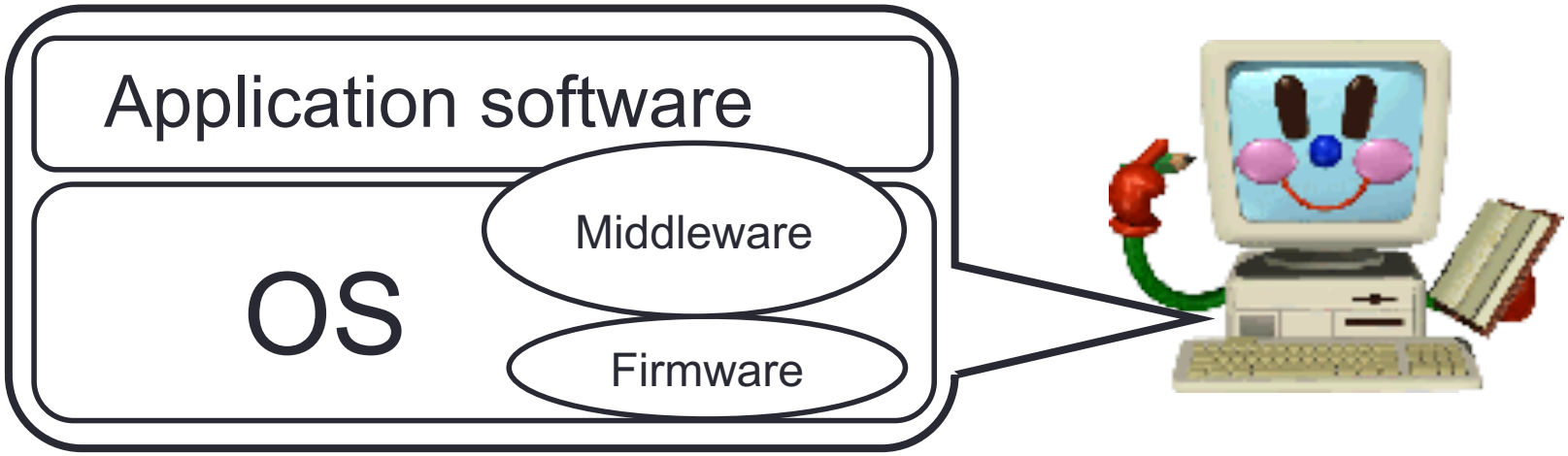
走査線数

TV　525本

CRT　1000本以上

ソフトウェア　Software

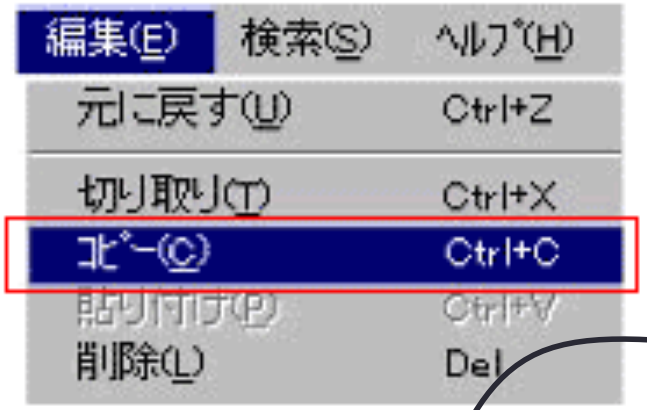
	コンピュータ	人体なら
ハードウェア	パソコン本体，キーボード マウス，プリンター等	脳，内臓，骨
ソフトウェア	OS，アプリケーション， ミドルウェア，ファームウェア	血液の循環 神経の伝達



OS（オーエス）

O　　S

コンピューターのハードウェア、ソフトウェアを有効に利用するためにコンピューター全体の動きをまとめる基本的で重要な役割を持つソフトウェア



現在はWindows 11

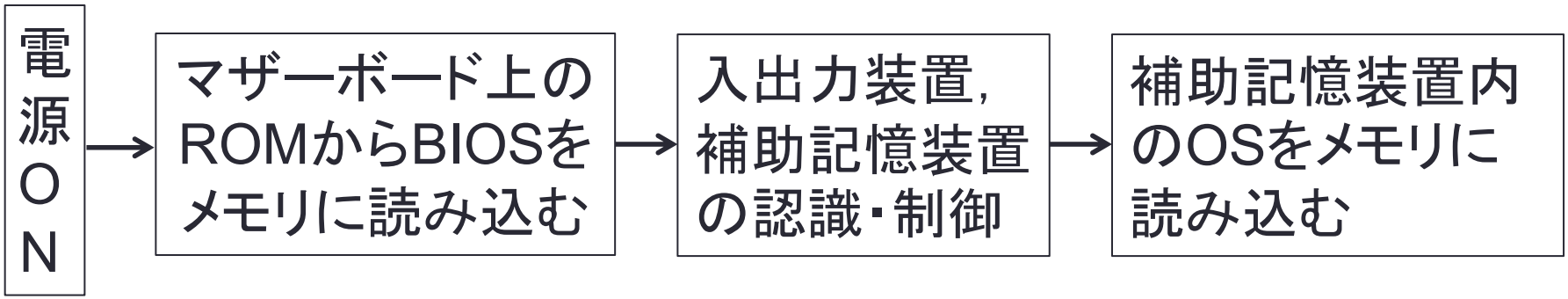
最近は地名が名前になっていることが多い
Sonoma, Ventura, など

OSの種類	動作ハードウェア
UNIX（派生）	大型コンピュータ
Windows	PC/AT互換機
Mac OS	Apple社コンピュータ
Linux	PC/AT互換機
iOS, Android（Linuxから派生）	モバイル機器

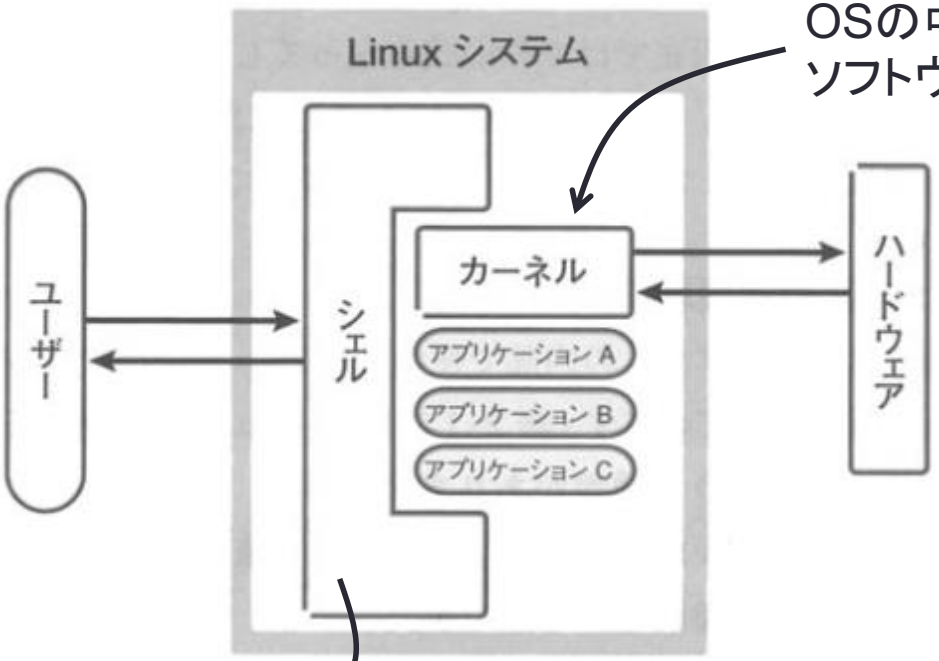
ファームウェア　Firmware

コンピュータのメモリは電源を入れた最初の状態では空であり，コンピュータを動かすには通常ハードディスクに保存されているオペレーティングシステムをメモリに読み込んでくる必要がある．この処理にはローダ（loader, boot loader）と呼ばれる読み込みのためのプログラムが必要である．ところが，ローダもプログラムであるから動くにはメモリに読み込む必要がある．

そこで，電源が消えても中身の消えないROMに，ローダを探してメモリに読み込むためのプログラム（IPL: Initial Program Loader）を記憶しておく方式がとられている．また，コンピュータの起動時に，通常，ディスプレイ，キーボード，マウス，ディスク装置が使える必要があるため，必要最小限の入出力制御プログラムもROMに記憶させてある．このようなROMに記録したソフトウェアのことを，ハードウェアとソフトウェアの中間に位置することからファームウェアと呼ぶ．IBM PC/AT互換機では，BIOS（Basic Input Output System），MacintoshではOpen Firmwareがこれに相当する．



オーダー(命令)の流れ



OSの中で特に中心的な機能・役割を果たす
ソフトウェア(ハードウェアの直接管理操作など)

カーネル
シェル
基本的なアプリケーション

OS

ユーザの指示を
解釈してカーネル
に伝えるプログラム

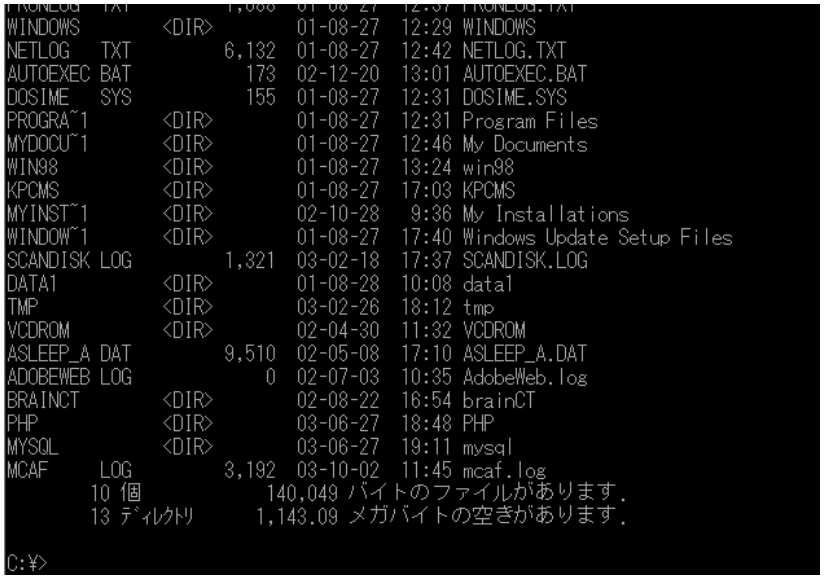
コマンド(Command)
・キーボード入力
・マウス入力

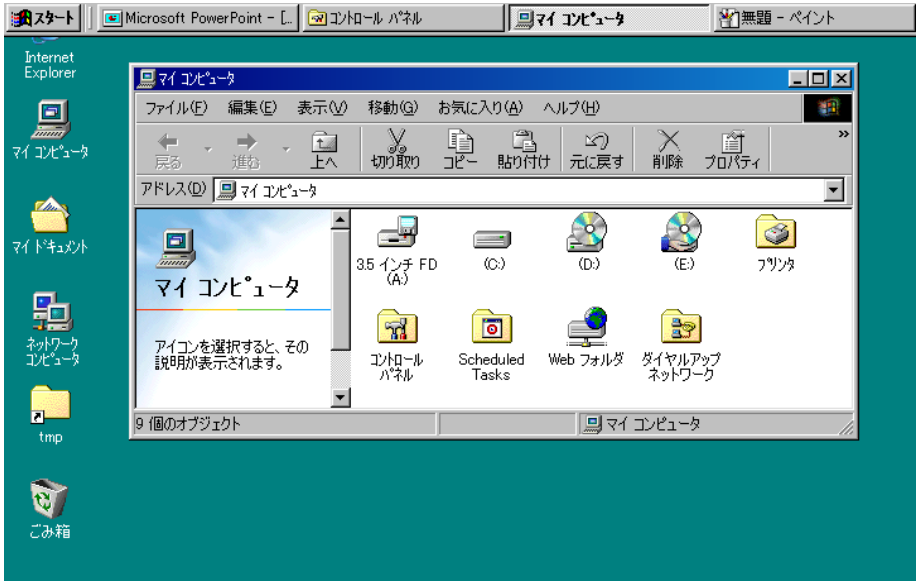
レストランなら	コンピュータ(ソフトウェア)
レストラン	OS
ウェイトレス	シェル
厨房	カーネル・アプリケーション
CUI or GUI	メニュー(注文方法)

シェフ

シェフの指示で
動く他の料理人

Character User Interface　Graphic User Interface



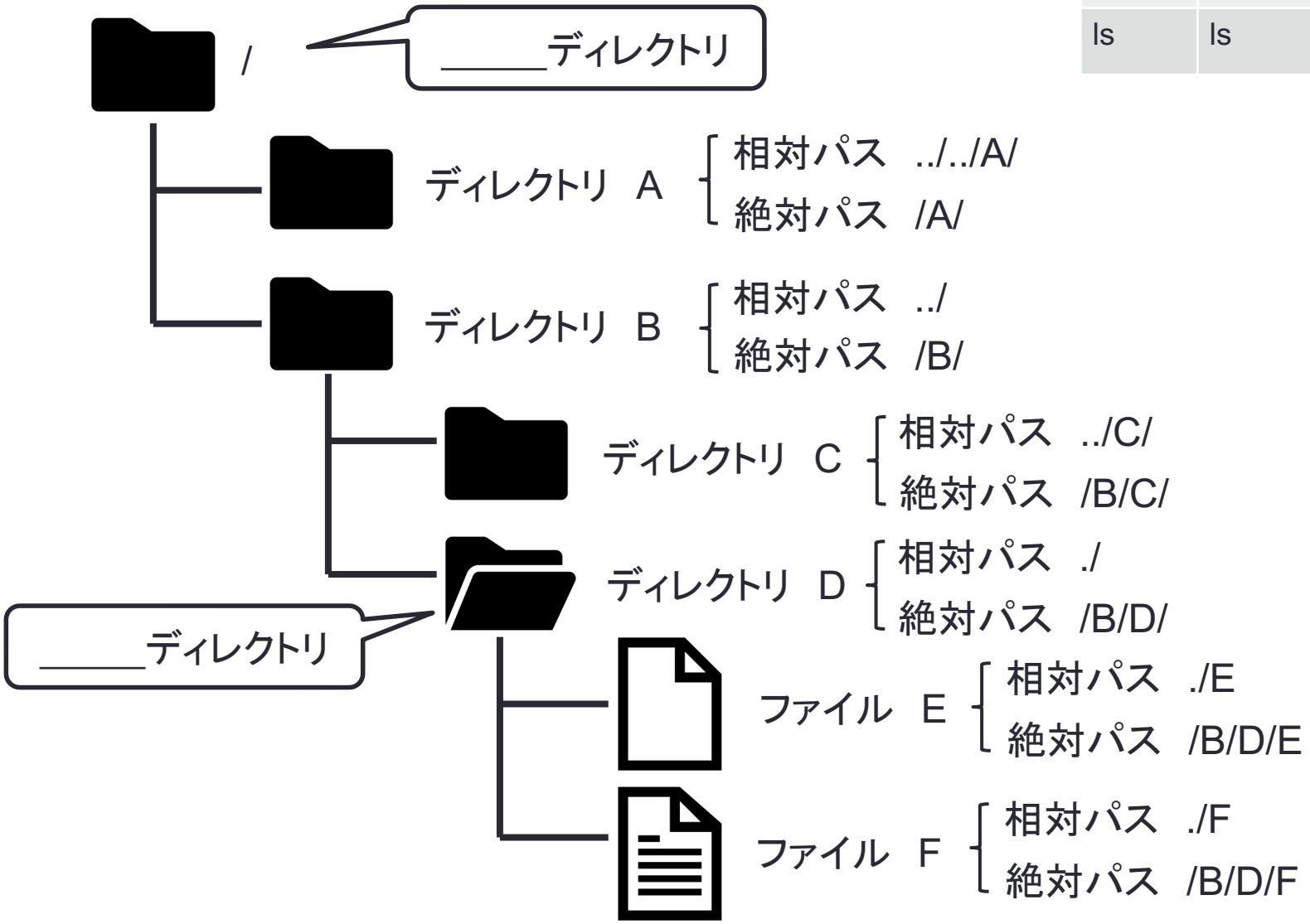


コンピュータへの命令を
文字で伝える
注文を細かく指定しなければ
ならない　→ライス、生姜焼き
例：　dir c:¥

コンピュータへの命令を
マウス，アイコンなど伝える
注文を細かく指定しなくてもよい
→生姜焼き定食
例：　cドライブをマウスクリック

階層構造ファイルシステム

コマンド	使い方	動作
pwd	pwd	カレントディレクトリの絶対パスを表示
cd	cd パス	指定したパスのディレクトリに移動
ls	ls	カレントディレクト内のファイルリストを表示




```
$ cd ../../
$ pwd
/
$ ls
A
B
$ cd B/D
$ pwd
/B/D
$ ls
E
F
$ cd /B/C
$ cd ../D
$ pwd
/B/D
```

OSの主な目的

- ハードウェアの抽象化

異なるハードウエア構成のコンピュータを同じような手順で操作できるようにする.

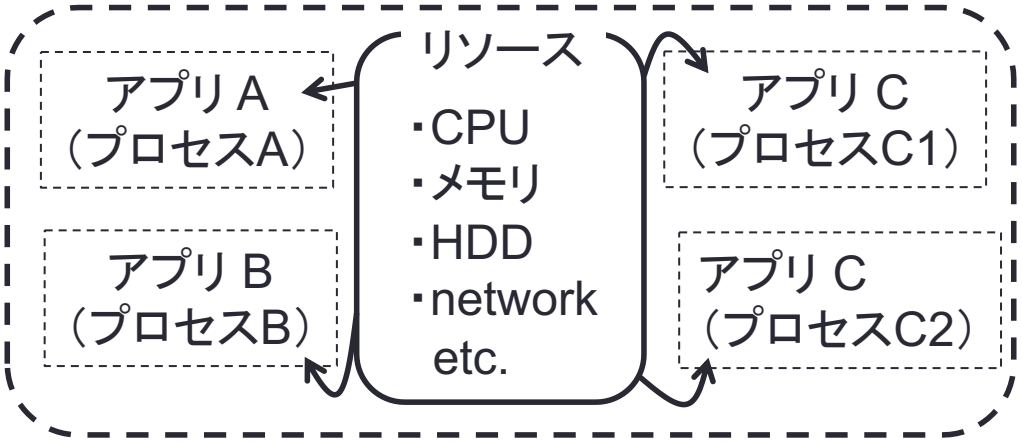
ユーザやアプリケーション開発者は，ハードウェアを個別に管理・制御しなくても扱える.

Kernel：ハードウェアを直接管理操作するプログラム
Device driver：ハードウェアを制御するプログラム

- リソースの管理

資源（CPU，メモリ，などプログラムから利用可能なもの）

各プログラムやユーザに必要なだけリソースを割り当てたり，適切に制御を行う



- コンピュータ利用効率の向上

メモリ管理，プロセス管理，周辺装置の制御，ネットワークのサポート，ファイルシステムの管理，ユーザの管理，ユーザインターフェースの提供，言語環境のサポート，電源管理

- **メモリ管理** ⟶ リソースの管理

  メモリはプロセスごとに排他的に割り当てられ，お互いに干渉しないように制御される．OSには，ハードディスクなどの補助記憶装置を使って，主記憶装置の実際の容量よりもはるかに大容量のメモリをシミュレートする機能もあり，これを仮想記憶と呼んでいる．

- **プロセス管理** ⟶ リソースの管理

  複数のプロセスを短い時間間隔で交互に処理させて，同時に処理しているようにみせる．プロセスまたはスレッドの切り替えのことをディスパッチという．プロセス管理プログラムは，多数のプロセスまたはスレッドに対し優先順位をつけながらCPUを使う時間を割り当て，ディスパッチを行う．プロセス管理のことをタスク管理ということもある．

- **周辺装置の制御** ⟶ ハードウェアの抽象化

- **ネットワークのサポート**

  ネットワーク機能を制御し，また，ネットワーク機能をアプリケーションプログラムが利用できるようにAPI（Application Program Interface）を提供するのもOSの役目である．

  System Call　とも言う

  OSのカーネルの機能をアプリケーションプログラムから利用できるようにOSが提供している関数．例：モニタに描画をする関数，キーボードなどからの入力を制御する関数

- ファイルシステムの管理

  ファイルシステムは，ファイルを正しく格納し，必要な時に必要な物を使用できるようにする仕組み．ディスクファイルシステムにはFAT（File Allocation Table），NTFS（NT File System），HFS（Hierarchical File System）等がある．

- ユーザの管理

  コンピュータのリソースに対する権限が異なるさまざまなユーザの管理や，複数ユーザの同時利用をサポートするのもOSの役割である．

- ユーザインタフェースの管理

  CUIまたはGUIにより，人間とコンピュータのコミュニケーションの手段を提供するのもOSの役割である．

- 言語環境のサポート

  OSはWindowsやMac OSなど英語圏由来のものが多い．これらのOSで英語以外の言語が使えるようにするのもOSの重要な機能である．

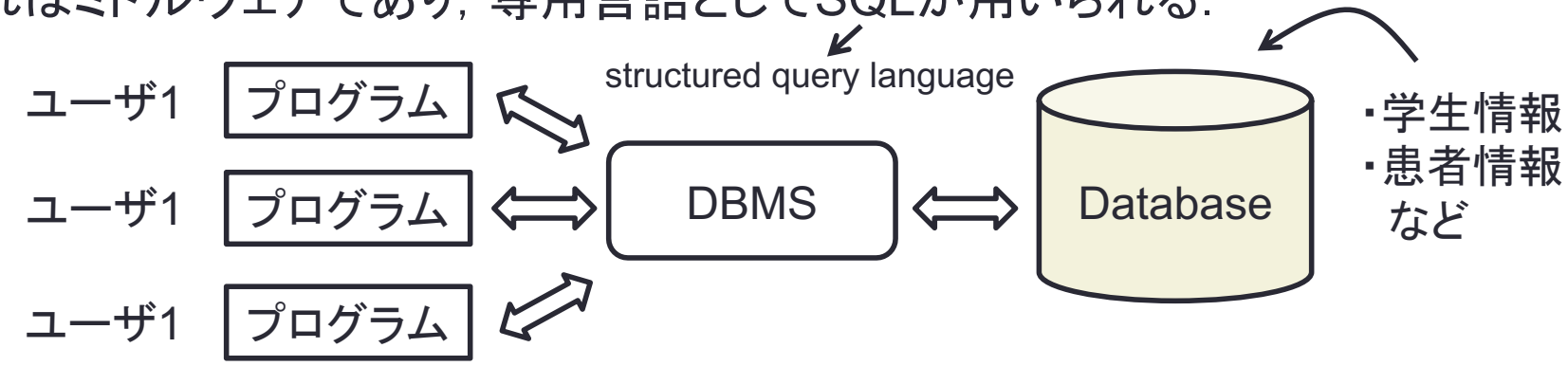
- 電源管理

  コンピュータの使用状況を監視し，一定時間使用されないときは，ディスプレイやHDDを止めたり，メモリ内容をHDDに待機させてシステム全体を一時呈するなど，電源を節約する機能で特にノートPCで重要な機能のひとつである．

ミドルウェア　middleware

オペレーティングシステムに組み入れるほどには基本的ではないが，ある分野においては複数の応用目的に共通に必要となるような機能がある．このようなレベルの共通機能を実装したソフトウェアをミドルウェア（middleware）と呼ぶ．

アプリケーションソフトウェアの多くは大量の情報を蓄積・管理し，検索等を行う．そのような共通の情報処理の要求に応じて，データベースの作成，維持，運用を効率的に行うために強力なデータ管理機能を提供するのがデータベース管理システム（DBMS：Database Management System）である．これはミドルウェアであり，専用言語としてSQLが用いられる．



アプリケーションソフトウェア

コンピュータを目的別に動かすためのソフトウェア

- ✓ 文書処理ソフトウェア（word processor, word processing software）
- ✓ 表計算ソフトウェア（spreadsheet program）
- ✓ プレゼンテーションソフトウェア（presentation software）
- ✓ 画像処理プログラム（image processing program）
- ✓ メーラー（mail software, mailer）
- ✓ ウェブブラウザー（Web browser）

など

医療情報処理システムでは，次のようなソフトウェアも利用される

- ・OMR（optical mark reader）→ マークシートを読み取り，割り当てられている文字や数値として認識させるソフトウェア
- ・OCR（optical character reader）→ 文字を含むスキャナ画像から，文字情報を自動で認識するソフトウェア
- ・バーコードリーダー（bar-code reader）
- ・QRコードリーダー（2次元バーコード）

バーコード


QRコード

プログラミング言語

低水準言語（低級言語 low-level language）

・機械語（machine language）　→　2進数で書かれた数字の列
・アセンブリ言語（assembly language）　→基本的な命令を短いアルファベットで表す

↳ アセンブラ（assembler）によって機械語に変換して実行する

高水準言語（高級言語 high-level language）

より人間の言葉に近い形式でプログラムを書くために開発されたプログラミング言語．
プログラムを実行する方法は大きく二通りある．

ひとつは，対応する機械語のプログラムに一括で変換して実行する方式，またはその変換を行うプログラムをコンパイラ（compiler）という．出来上がった実行可能なプログラムはobject programまたはobject codeという．これに対し，元の高級言語で書かれたプログラムを，ソースプログラム（source program）やソースコードという．

もうひとつの方法として，ソースプログラムを全部機械語に変換するのではなく，1ステップずつ取り出して解釈し，該当する処理を行っていく方法があり，これを行うプログラムをインタープリタ（interpreter）という．コンパイラ方式よりも実行速度は遅い．

代表的な高水準言語

- FORTRAN
- COBOL
- Pascal
- BASIC
- C
- C++
- C#
- JAVA
- Python

【例】
```c
#include <stdio.h>

int main(int argc, char **argv)
{
    int     i, j;
    for ( i = 0; i <= 9 ; i++) {
        for ( j = 0; j <= 9 ; j++) {
            printf ("%d    ", i*j);
        }
        printf ("¥n");
    }
    return 0;
}
```

C言語による
九九演算プログラム

```java
/*
  2バイト画像の離散フーリエ変換（DFT）
  コンパイル：javac -encoding UTF8 ImageDFT.java
  実行：java ImageDFT (input) (xsize) (ysize) (swap)
       input：入力画像のファイル名
       xsize：入力画像の横方向の画素数
       ysize：入力画像の縦方向の画素数
       swap：バイトスワップ（1:実行，1以外:実行しない）
  出力：power.raw（パワースペクトル画像，xsize*ysizeの2バイト画像）
*/
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.lang.Math;

public class ImageDFT {
    public static void main(String[] argv) {
        try {
            /* 変数の宣言 */
            int i,a,b,c;
            /* 変数の宣言とコマンドラインからの引数取得 */
            String input = argv[0];
            int xsize = Integer.parseInt(argv[1]);
            int ysize = Integer.parseInt(argv[2]);
            int swap =  Integer.parseInt(argv[3]);
            /* 入出力用のメモリ領域（配列）の宣言と生成 */
            short in[] = new short[xsize*ysize];
```

JAVA言語による
画像処理プログラム

74

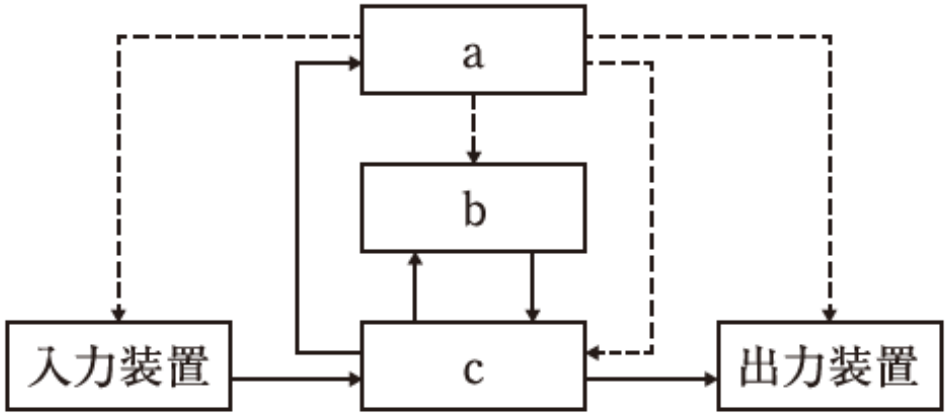
午　前

◎ 指示があるまで開かないこと。

（令和４年２月17日　９時30分～12時05分）

47　コンピュータの基本構成を図に示す。

a～cに入る装置の組合せで正しいのはどれか。



	a	b	c
1.	演算装置	記憶装置	制御装置
2.	記憶装置	演算装置	制御装置
3.	記憶装置	制御装置	演算装置
4.	制御装置	演算装置	記憶装置
5.	制御装置	記憶装置	演算装置

73

午　後

◎ 指示があるまで開かないこと。

(令和3年2月18日　13時25分～16時00分)

47　コンピュータソフトウェアにおけるオペレーティングシステム〈OS〉の機能はどれか。

1. リスク分析
2. 知的財産権の管理
3. プロジェクト管理
4. データベースシステムの管理
5. ユーザインターフェースの提供

70

午　前

◎ 指示があるまで開かないこと。

(平成 30 年 2 月 22 日　9 時 30 分 ～ 12 時 05 分)

46　コンピュータの機能と対応装置の組合せで正しいのはどれか。

1. 演　算 ―――― CRT
2. 記　憶 ―――― SSD
3. 出　力 ―――― CPU
4. 通　信 ―――― UPS
5. 入　力 ―――― RAM

69

午　前

◎ 指示があるまで開かないこと。

(平成 29 年 2 月 23 日　9 時 30 分 ～ 12 時 05 分)

46　コンピュータの 5 大機能でないのはどれか。

1. 演　算
2. 記　憶
3. 制　御
4. 通　信
5. 入　力

2013年（平成24年2月） 国家試験問題

機能と対応装置の組み合わせで正しいのはどれか。

1. 入 力 -------- モデム
2. 演 算 -------- ROM
3. 記 憶 -------- CPU
4. 通 信 -------- HUB
5. 出 力 -------- タッチパネル

2012年（平成23年2月） 国家試験問題

コンピュータの基本構成で正しいのはどれか。
2つ選べ。

1. 終端装置
2. 出力装置
3. 認証装置
4. 記憶装置
5. 経路制御装置

ソフトウェアでないのはどれか。

1. アプリケーション
2. ファームウェア
3. メインフレーム
4. オペレーティングシステム
5. グラフィックユーザインタフェース