

2. コンピュータの構成要素

http://cobayasi.com/koza/ap/2_compyouso.pdf

2.1 プロセッサ

2.2 メモリアーキテクチャ

2.3 入出力装置と入出力デバイス

2.1 プロセッサ

2.1.1 プロセッサの構造と動作原理

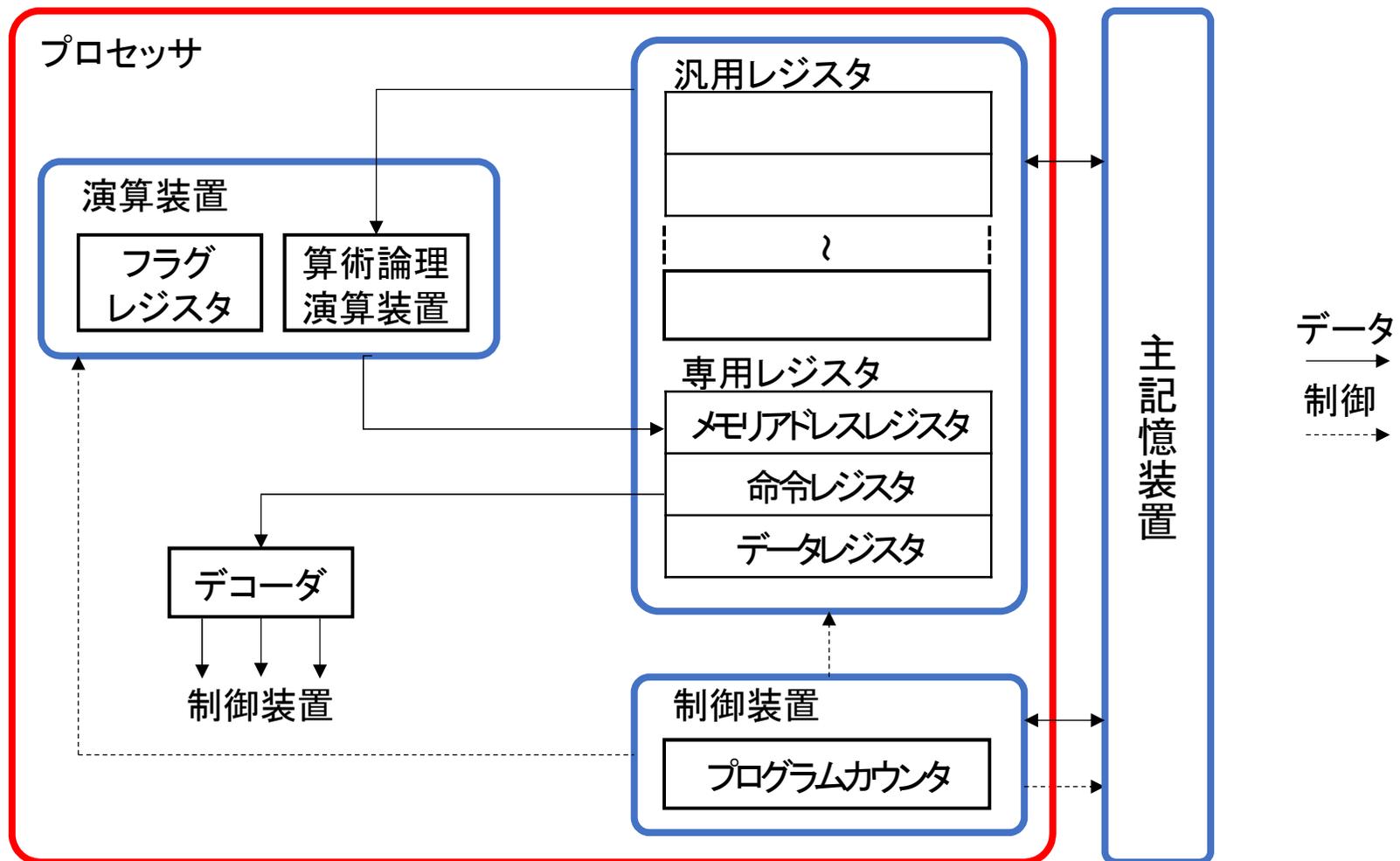
2.1.2 プロセッサの高速化

2.1.3 マルチプロセッサ

2.1.1 プロセッサの構造と動作原理

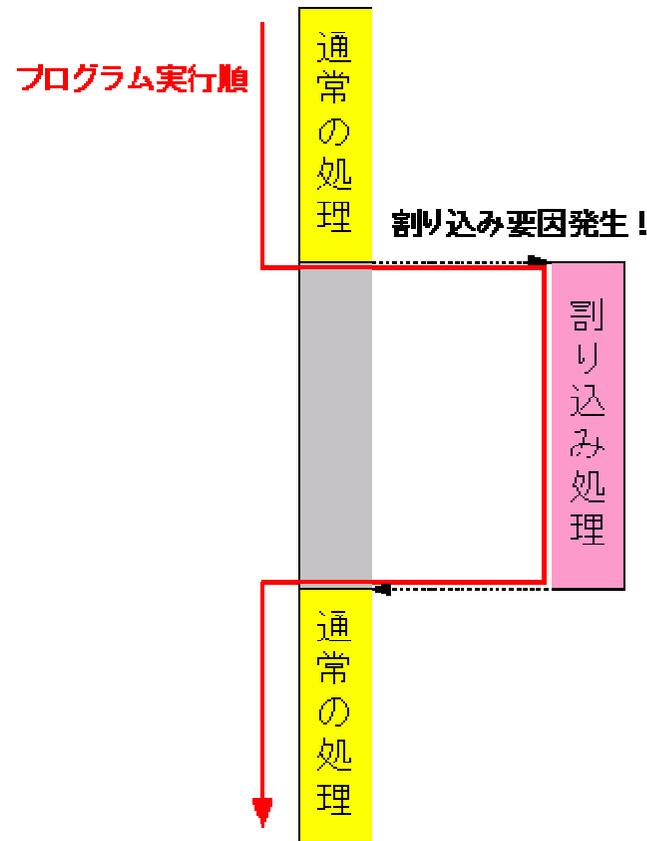
- プロセッサの構成

プロセッサとは、CPU(Central Processing System)のことで
演算装置と**制御装置**の総称



● 割込み

現在実行中の処理(プログラム)を中断して、割込みの原因に対応した処理(割込み処理)を実行し、割込み処理が完了したら、元の処理を中断したところから実行する



- 割込みの分類

外部割込み(external interrupt)またはハードウェア割込み

実行中のプログラムとは**無関係の原因**で起こる割込み

内部割込み(internal interrupt)またはソフトウェア割込み

実行中のプログラムが**関係する原因**で起こる割込み

<外部割込み>

- ✓ 機械チェック割込み

処理の誤動作、電源の異常、主記憶装置の異常などの

ハードウェア障害が原因で起こる割込み処理

- ✓ 入出力割込み

入出力処理の動作完了や状態の変化(紙切れ、電源断など)で起こる割込み処理

- ✓ 外部信号割込み

操作卓からの指示や外部機器(測定器など)からの信号が原因で起こる割込み処理

✓ タイマ割込み

所定の時間を過ぎたときに起こる割込み処理

〈内部割込み〉

✓ プログラム割込み

ゼロによる除算、桁あふれ(オーバーフロー)、書込みができない主記憶装置の場所に強制的に書き込もうとしたとき(記憶保護例外)などで、起こる割込み処理

✓ スーパーバイザコール(SVC : Super Visor Call)

入出力処理要求などのカーネル呼び出し命令が実行したときに、起こる割込み処理

✓ ページフォルト

仮想記憶で存在しないページへアクセスしようとしたとき(ページフォルト)に、起こる割込み処理

- 割り込み動作(処理)

- ① 現在実行中のプログラムのPSW(Program Status Word)をスタックに保存する
- ② 割り込み処理に対応したルーチン(プログラム)のPSWをCPUに設定する
- ③ 割り込み処理に対応したルーチンを実行する
- ④ ①でスタックに保存したPSWをCPUに設定する
- ⑤ PSWを使って中断したプログラムを再開する

※PSW:プログラム状態語

CPUで命令の実行される順番を制御したり、特定のプログラムに関連するコンピュータシステムの状態を示し保持しておくための制御ワード。実行中のCPUの状況やフラグレジスタの内容、命令アドレスレジスタ、割り込みの状態などを1語にまとめて記憶している。

2.1.2 プロセッサの高速化方式

プロセッサの内部構造によって、高速化の効率が異なる

- CISC(Complex Instruction Set Computer)

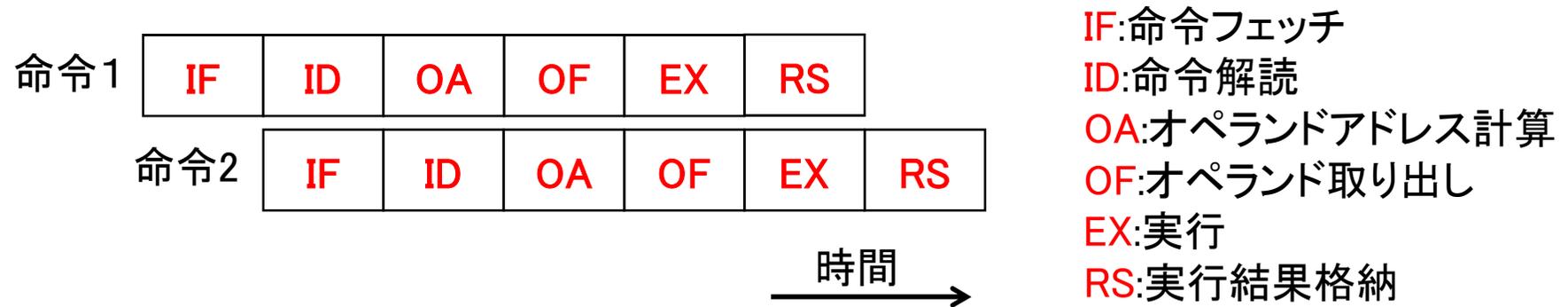
複雑な命令を、プロセッサ内に収められているマイクロプログラムを使って実行できる。様々な命令を実行することができるが、実行時間が命令ごとに異なる。パイプライン処理しても、処理速度はあまり上がらない。

- RISC(Reduce Instruction Set Computer)

内部構造を単純化することで、比較的単純な命令を組み合わせ、すべての命令をハードウェアで実行することができる。ほとんどの命令でほぼ実行時間が同一になり、パイプライン処理すると、処理速度が飛躍的に向上する。

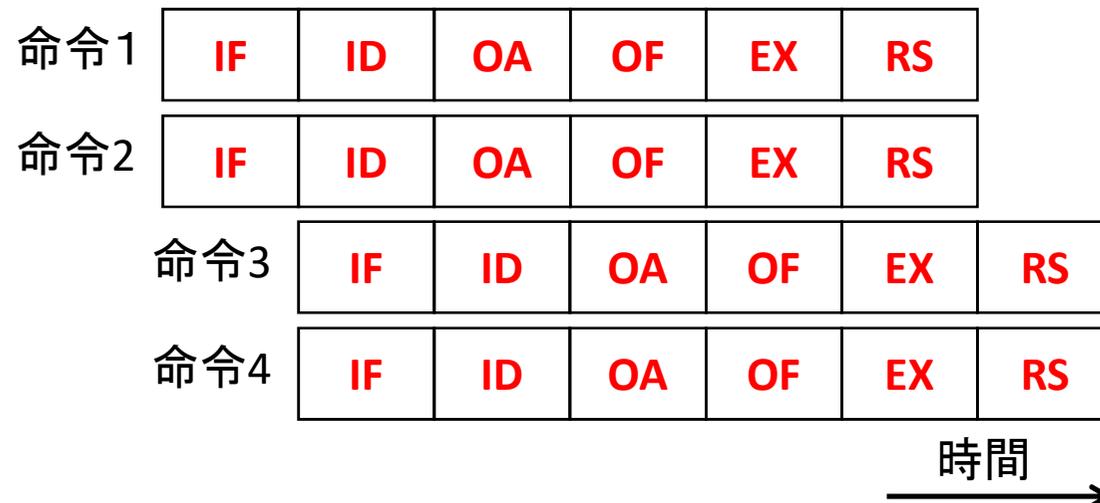
● パイプライン制御

複数の命令を、ずらしながら同時に実行する



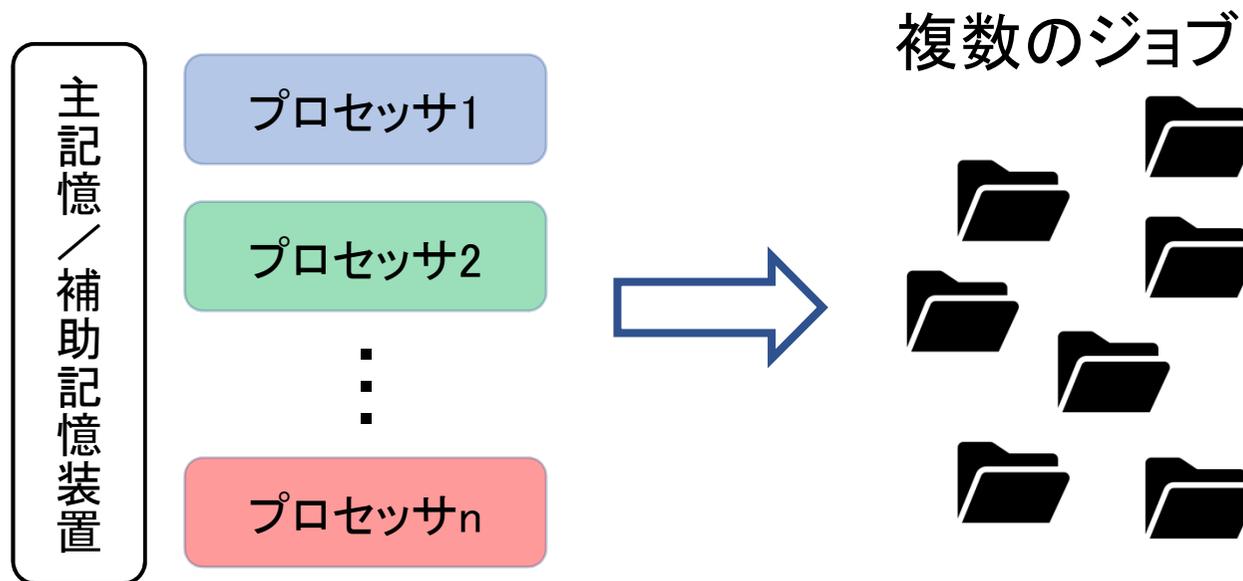
● スーパースカラ方式

複数のパイプライン処理で、同時に複数の命令を実行する



2.1.3 マルチプロセッサ

複数のプロセッサが主記憶や補助記憶などを共有し、ジョブを分担して並列処理するシステム構成を、**マルチプロセッサシステム**と呼ぶ



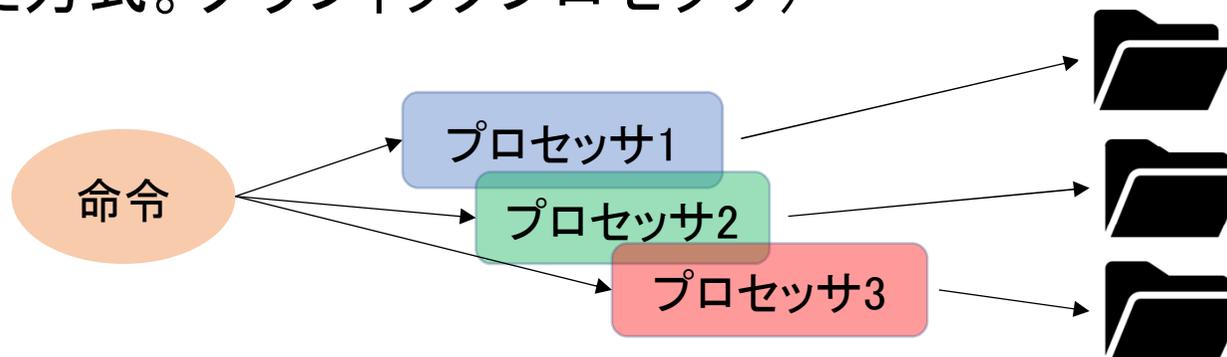
● M.Flynn(Flynn) の分類

命令の流れとデータの流れに着目して、コンピュータのアーキテクチャを分類した

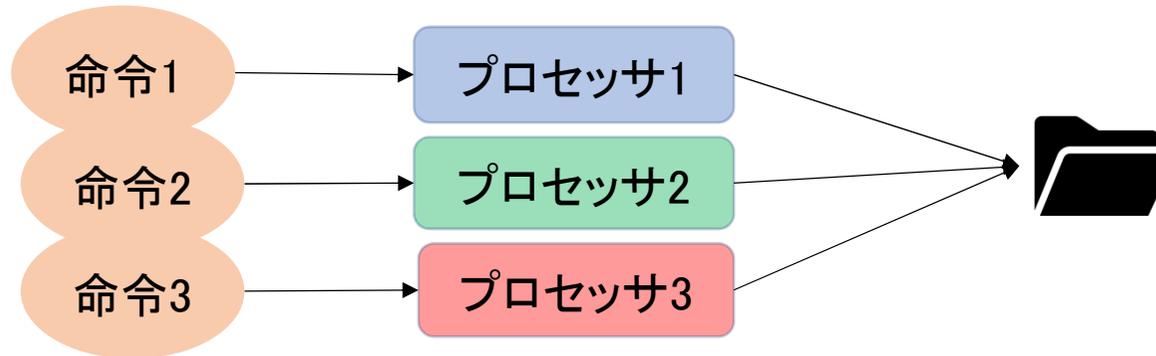
- SISD(Single Instruction Single Data stream)
1命令で1データを処理する方式(一般的なコンピュータの方式)



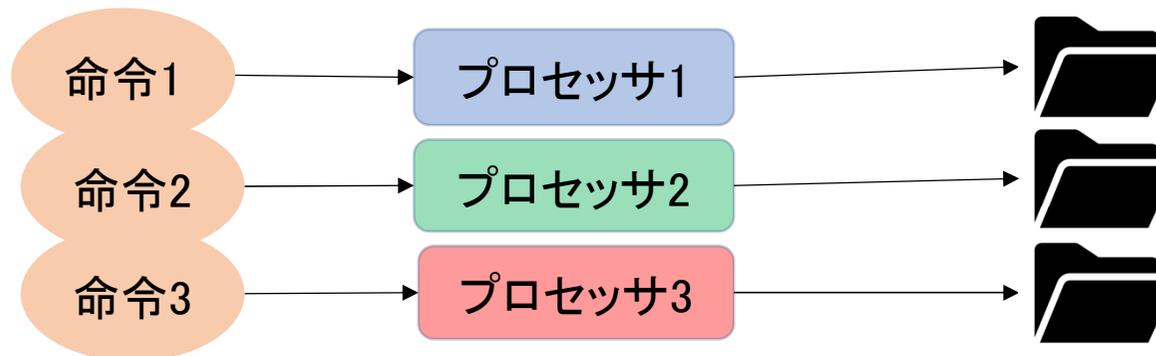
- SIMD(Single Instruction Multiple Data stream)
1命令で複数のデータを処理する方式(マルチメディア系の処理に適した方式。グラフィックプロセッサ)



- MISD(Multiple Instruction Single Data stream)
複数の命令で1データを処理する方式(実際のシステムは無)



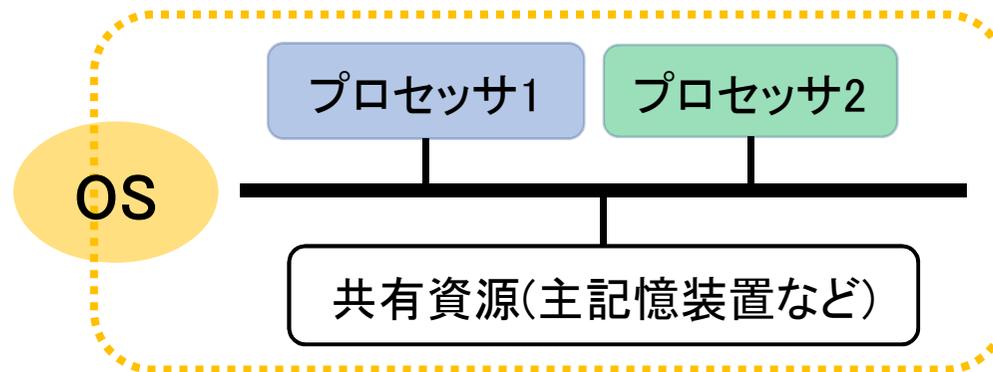
- MIMD(Multiple Instruction Multiple Data stream)
複数の命令で複数のデータを処理する方式(マルチプロセッサシステム)



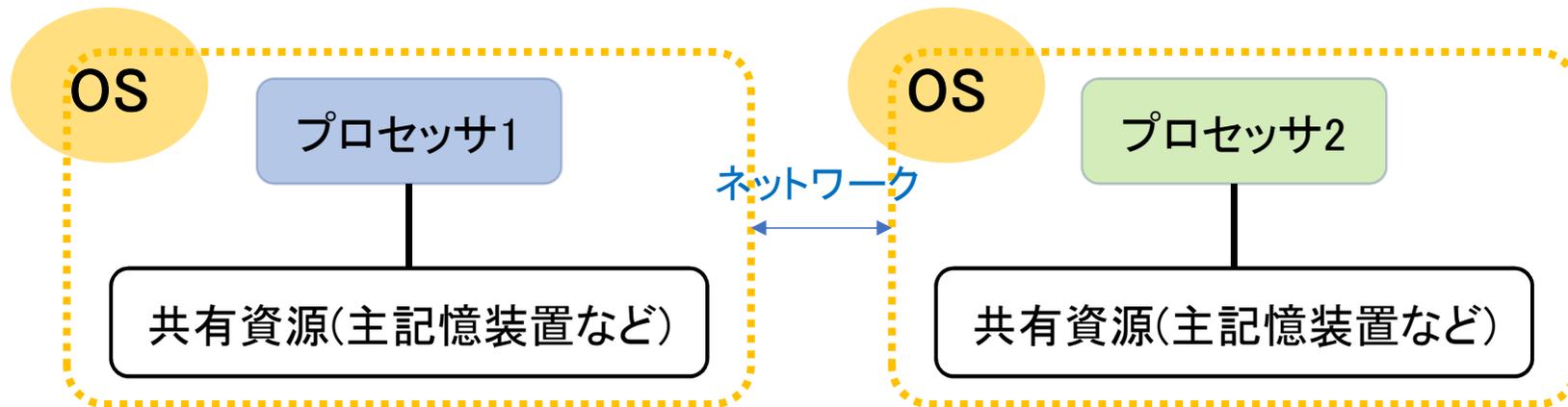
- マルチプロセッサと共有資源

複数のプロセッサが協調動作するためには、主記憶装置などの共有資源との関係が重要な問題となる。この共有資源との関係性を分類した、マルチプロセッサの構成方法。

- 蜜結合マルチプロセッサ(TCMP:Tightly Coupled MultiProcessor)
複数のプロセッサが共有資源を共有して、単一のOSが制御する



- 疎結合マルチプロセッサ(LCMP:Loosely Coupled MultiProcessor)
複数のプロセッサがそれぞれの共有資源(主記憶装置やOSなど)を持ち、各プロセッサの各OSが共有資源を制御する。各システム間をネットワークで接続して、協調動作する



- マルチプロセッサと隘路(ボトルネック)^{あいろ}

プロセッサの数とシステムの処理性能の関係は、アムダールの法則によって、以下の式で示される

$$E = \frac{1}{1 - r + \left(\frac{r}{n}\right)}$$

E : 速度向上率(単一プロセッサ比)

n : プロセッサ数

r : 高速化が期待できる割合($0 \leq r \leq 1$)

- 対称型マルチプロセッシングと
非対称型マルチプロセッシング

〈対称型マルチプロセッシング(Symmetrical Multi-Processing:SMP)〉

複数のプロセッサを同等に扱って動作させる方法。メモリ空間などの資源を共有に使用して、処理能力や耐障害性などの向上を図る。

〈非対称型マルチプロセッシング(Asymmetrical Multi-Processing:AMP)〉

各プロセッサに役割を決めて(OSカーネル実行、アプリケーション実行など)動作させる方法

■ 過去問題1

スーパスカラの説明はどれか？

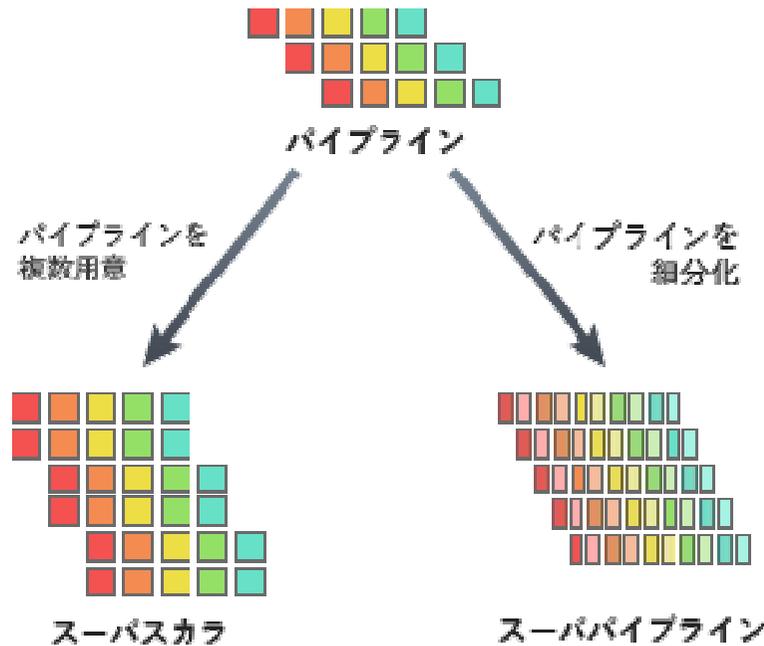
ア: 処理すべきベクトルの長さがベクトルレジスタより長い場合、ベクトルレジスタ長の組に分割して処理を繰り返す方式である

イ: パイプラインを更に細分化することによって高速化を図る方式である

ウ: 複数のパイプラインを用いて、同時に複数の命令を実行可能にすることによって高速化を図る方式である

エ: 命令語を長く取り、一つの命令で複数の機能ユニットを同時に制御することによって高速化を図る方式である

スーパースカラは、CPU内部に複数のパイプラインを用意して、パイプラインの各ステージを並列に実行することで、処理を高速化する手法である



ア: ベクトル型プロセッサ

イ: スーパーパイプライン

エ: VLIW (Very Long Instruction Word)

2.2 メモリアーキテクチャ

低速で安価な大容量の記憶装置と、高速で高価な小容量の記憶装置を、組み合わせる技術

2.2.1 情報素子

2.2.2 キャッシュメモリ

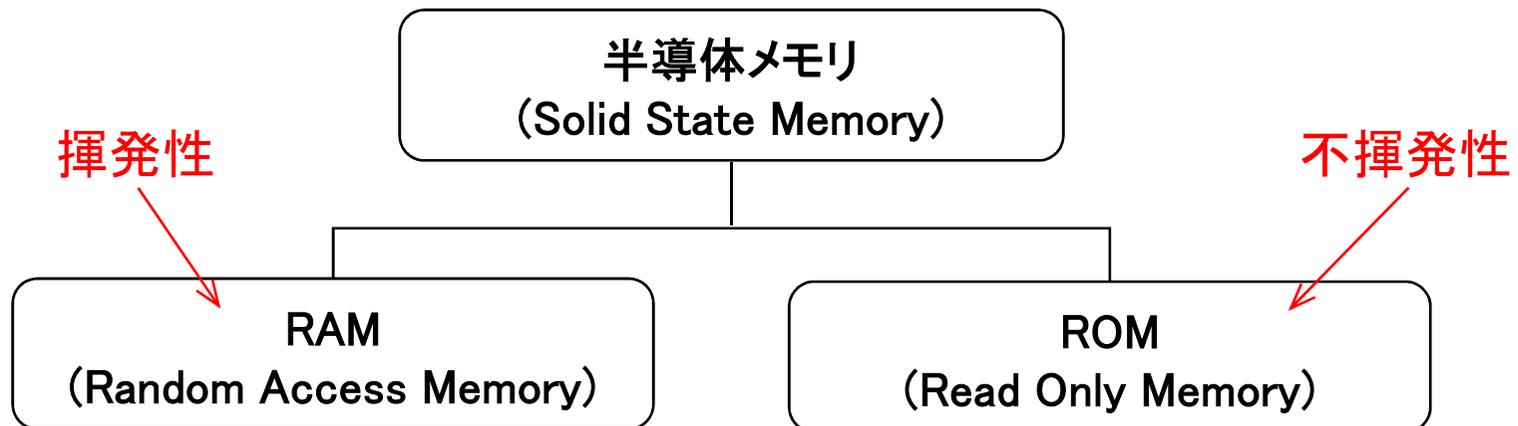
2.2.3 チェック方式

2.2.1 情報素子

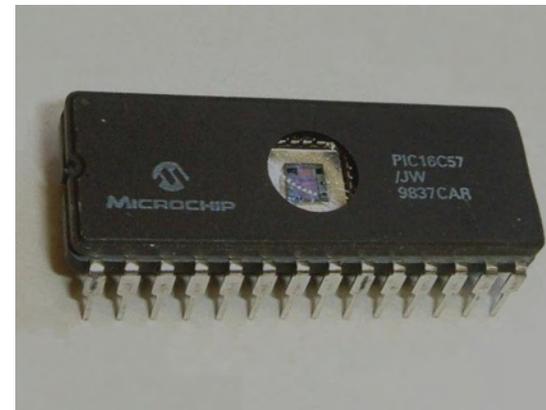
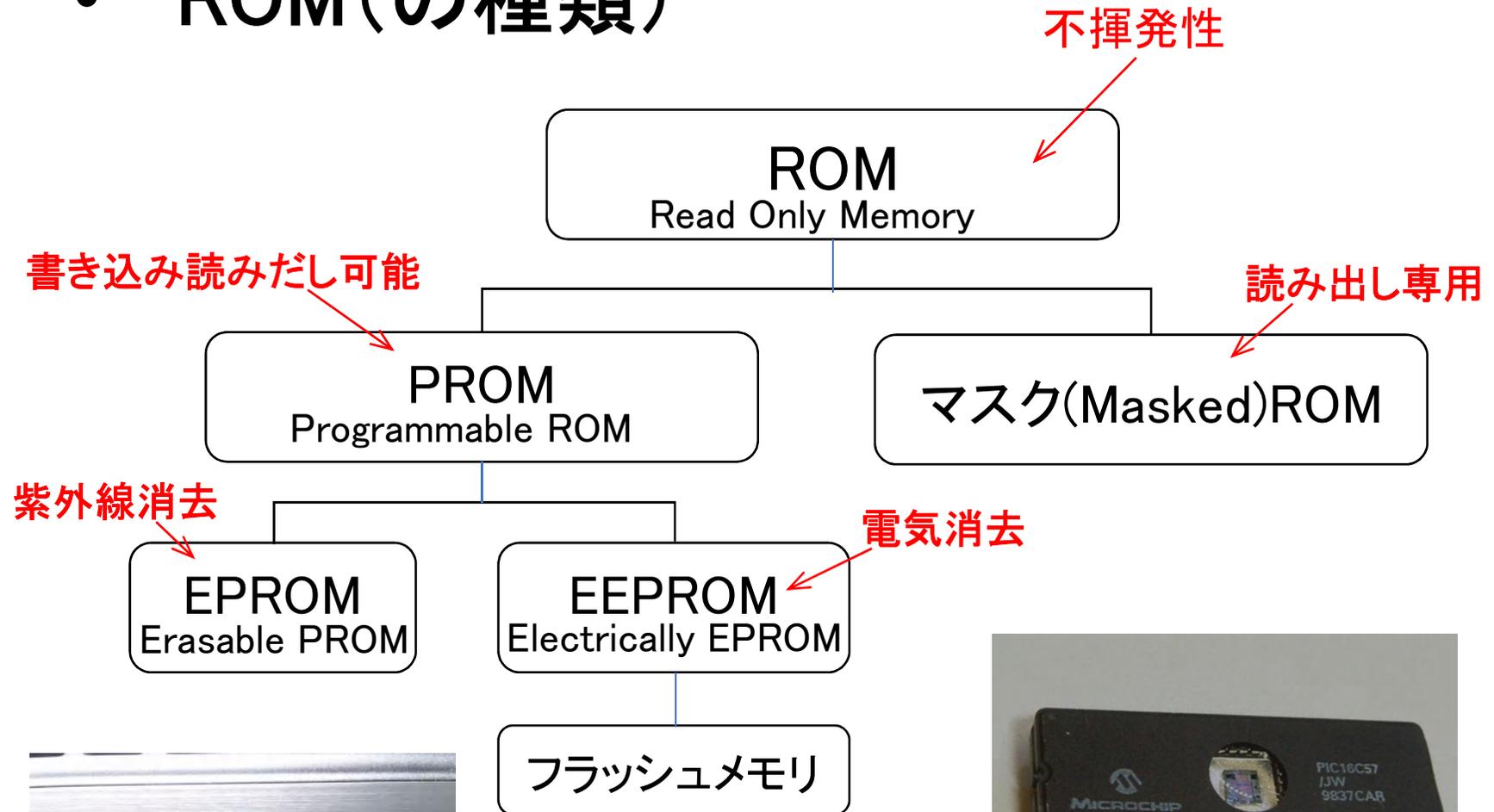
- 半導体メモリ

半導体(semiconductor)※と呼ぶ物質(シリコンなど)で出来ている記憶装置

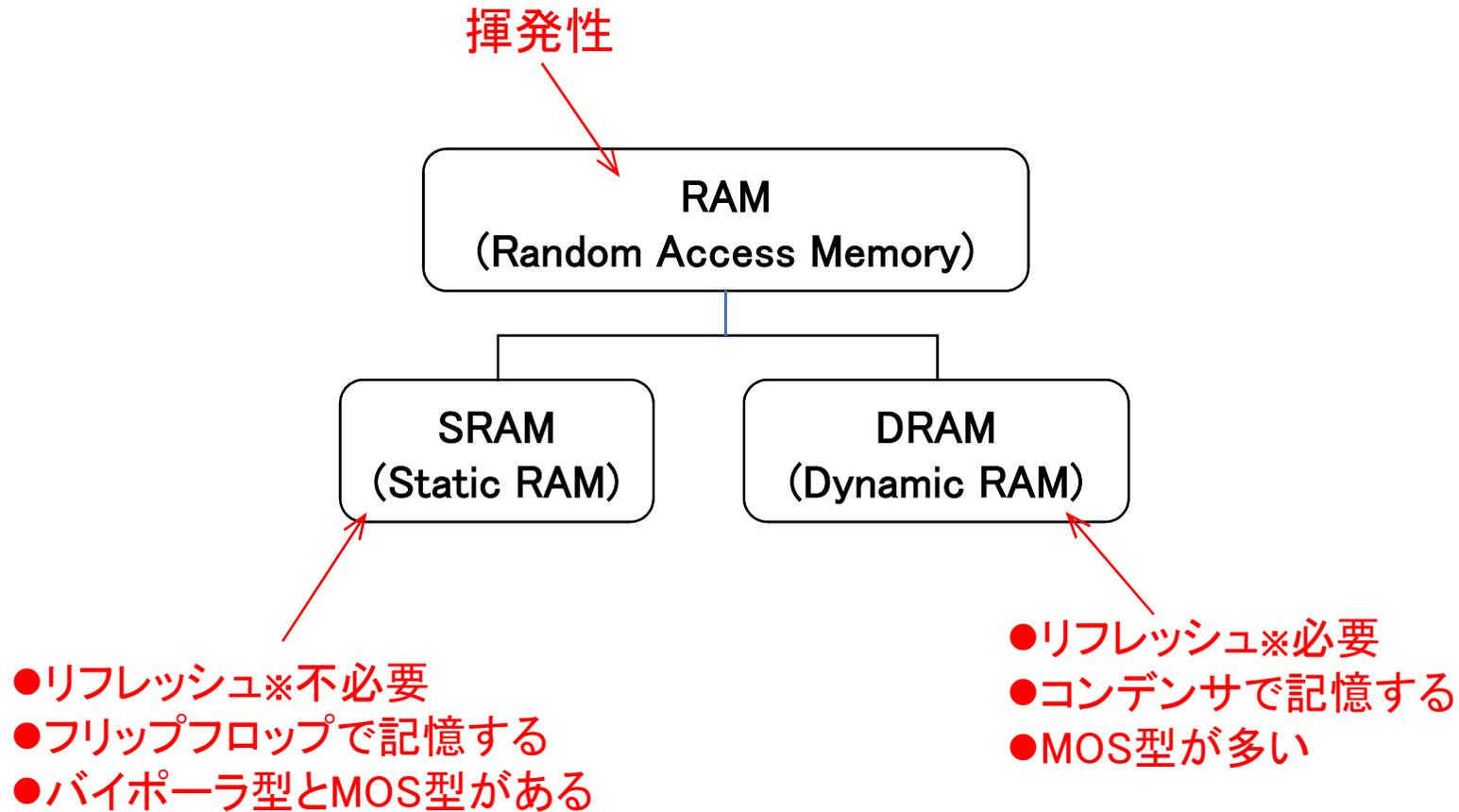
※電気を通しやすい金属などの導体(良導体)と電気を通しにくい絶縁体の中間的な抵抗率をもつ物質



- ROM(の種類)



• RAM(の種類)



※リフレッシュ
書き込んだ内容を消滅しないうちに、
もう一度書き込み直す

バイポーラ(BIPOLAR)型と

MOS(Metal-Oxide Semiconductor)型

- バイポーラ(双極)型
電子(負電荷)および正孔(正電荷)の2種類のキャリア(電荷:電気の素)が作用する半導体
- MOS(金属酸化物半導体)型
金属と半導体との間に酸化物絶縁体を挟んだ構造の半導体

RAMの型と特徴

RAMの型	動作速度	雑音の影響	集積度	ビット単位	用途
MOS	低速	小	高	安価	主記憶
バイポーラ	高速	大	低	高価	キャッシュ

磁性体(磁気)メモリ

磁気を利用した記憶媒体で、記憶容量が大きいので、主に補助記憶装置として使う



磁気ディスク(ハードディスク)



磁気テープ(カセットテープ)

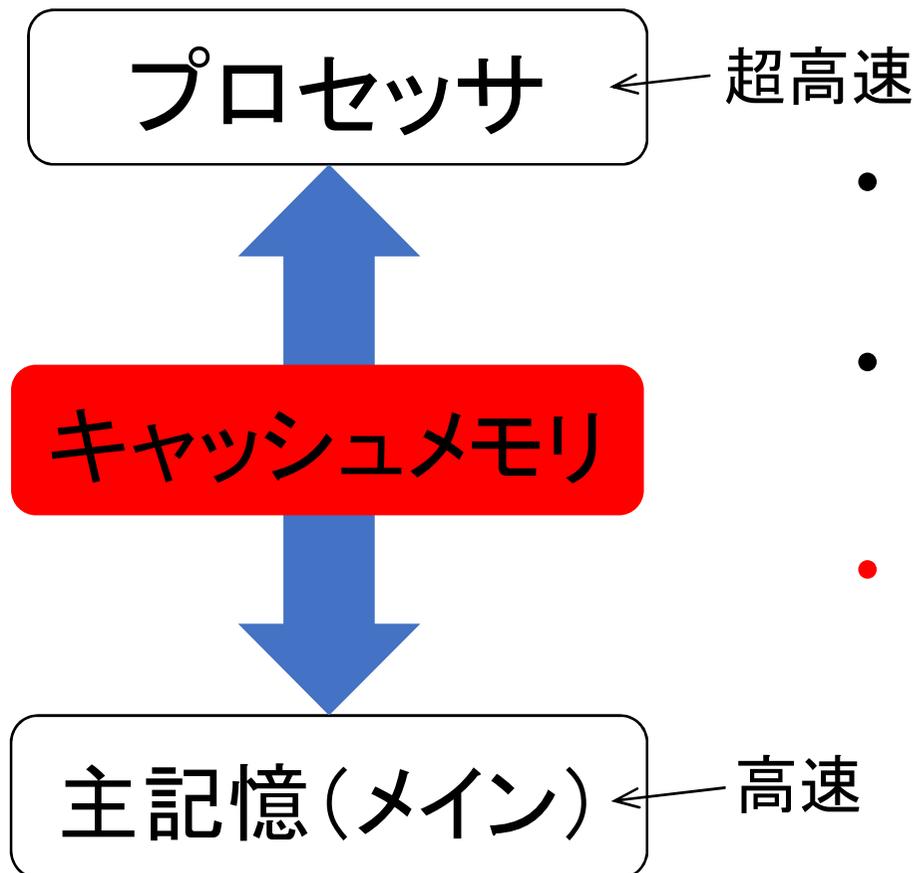


磁気テープ(ビデオテープ)



磁気テープ(DAT)

2.2.2 キャッシュメモリ

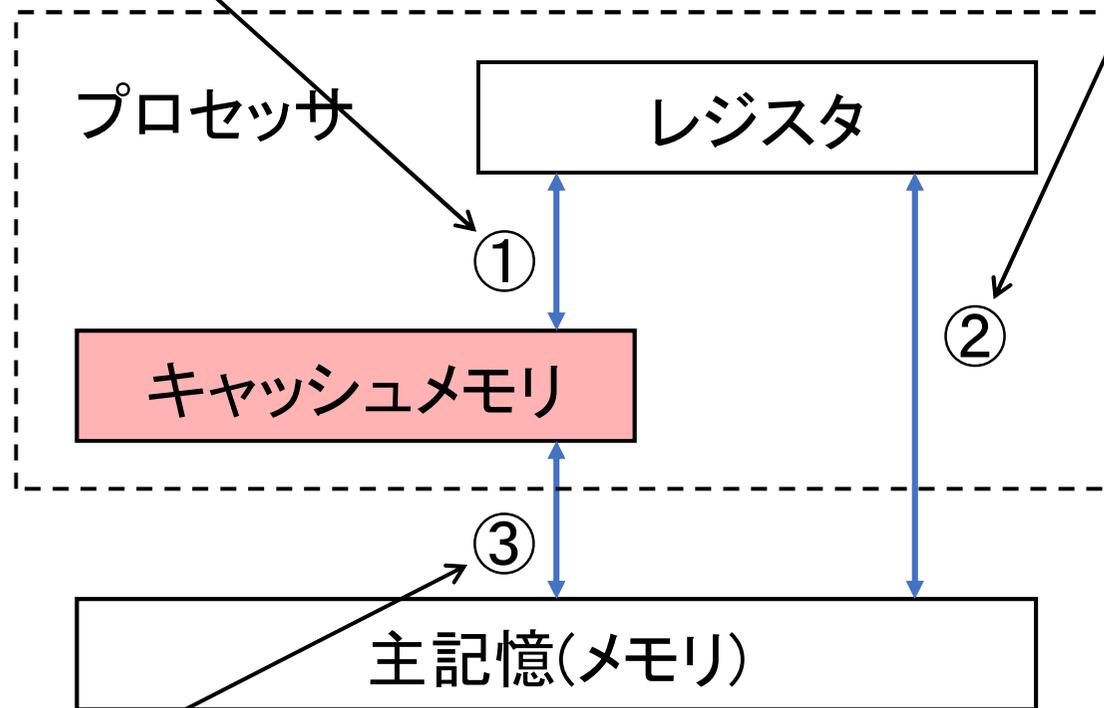


- キャッシュメモリ:
プロセッサ-主記憶
- ディスクキャッシュ:
主記憶-磁気ディスク
- **キャッシング:**
CPUと主記憶装置
のアクセス速度の
差を埋める

● キャッシュメモリの役割

データがキャッシュメモリにある場合

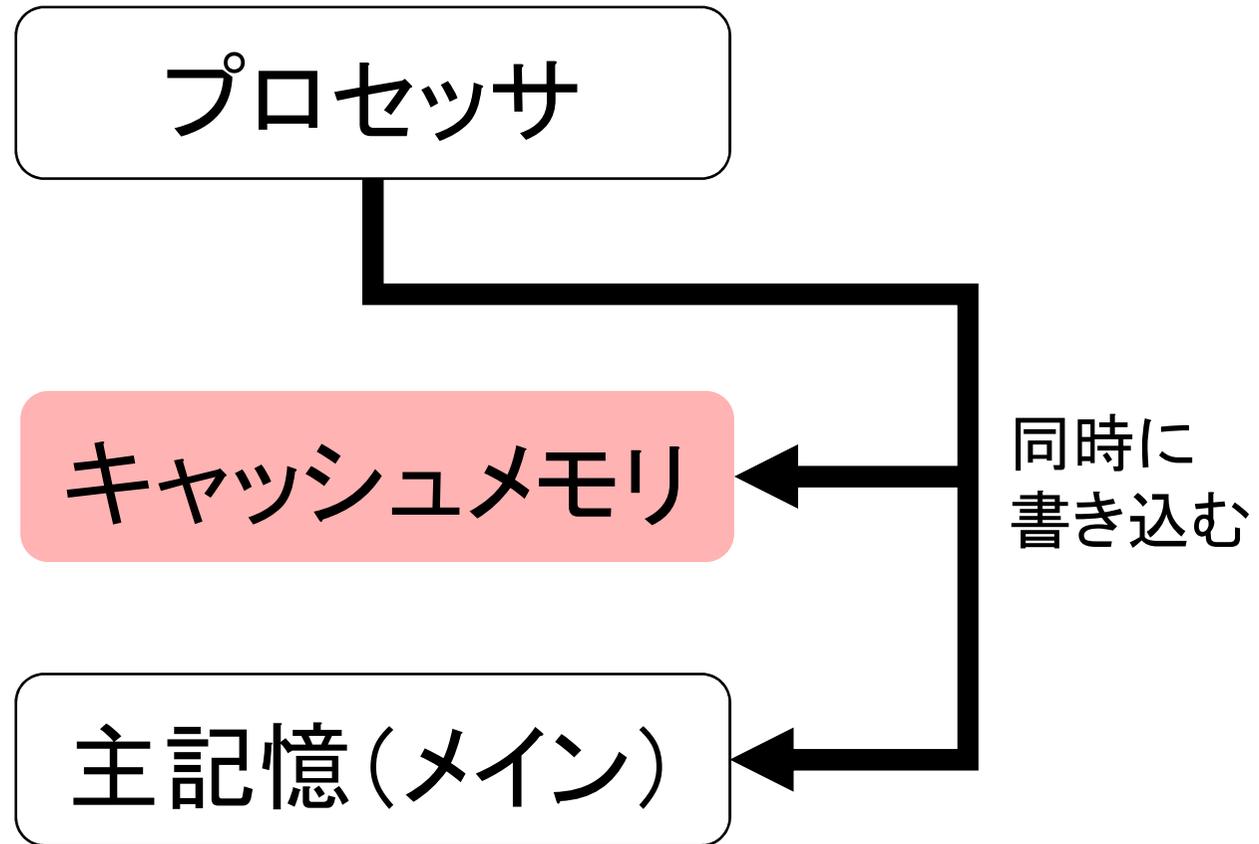
データがキャッシュメモリにない場合



よく使うデータやプログラムを予め
キャッシュメモリに保存する

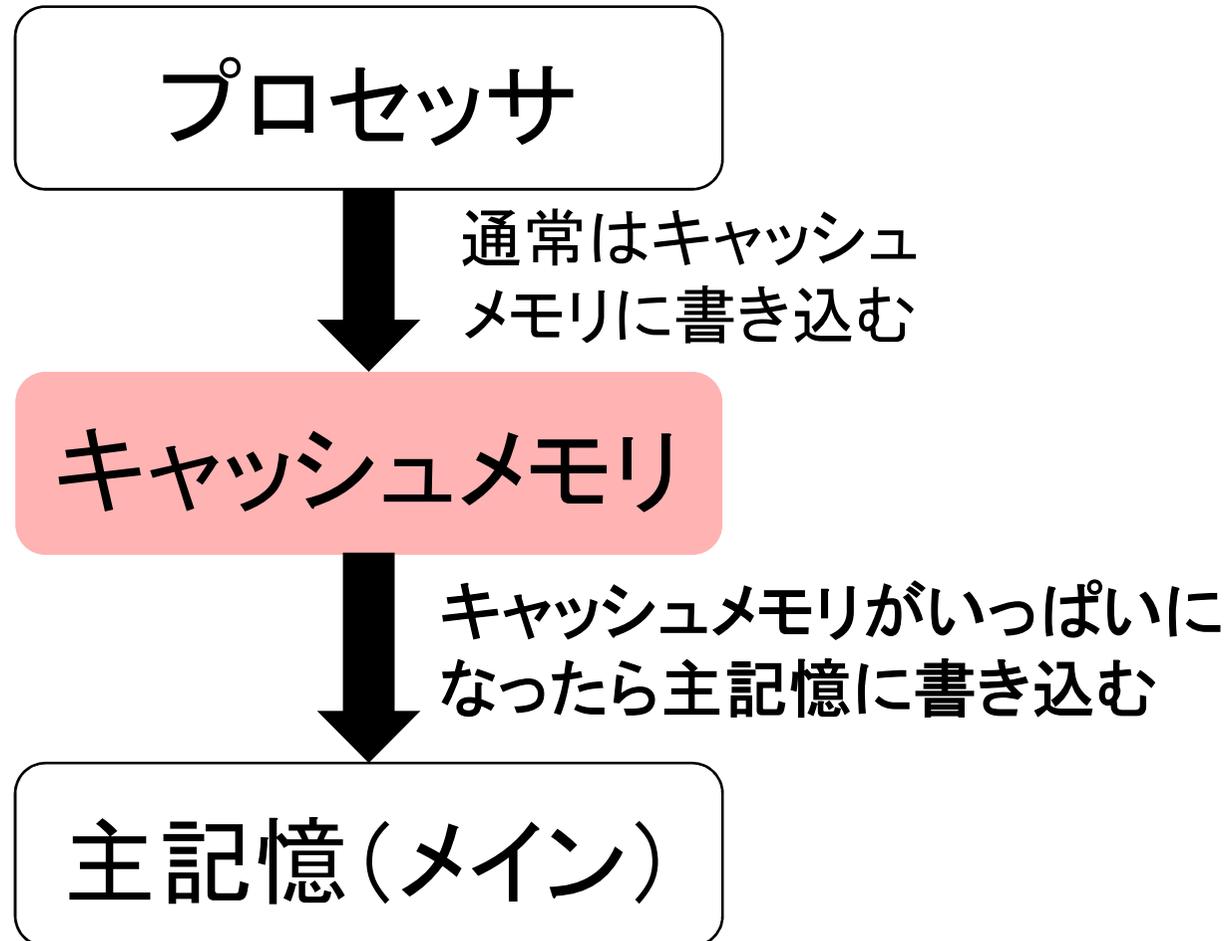
ライトスルー方式

- 読み出しが速い



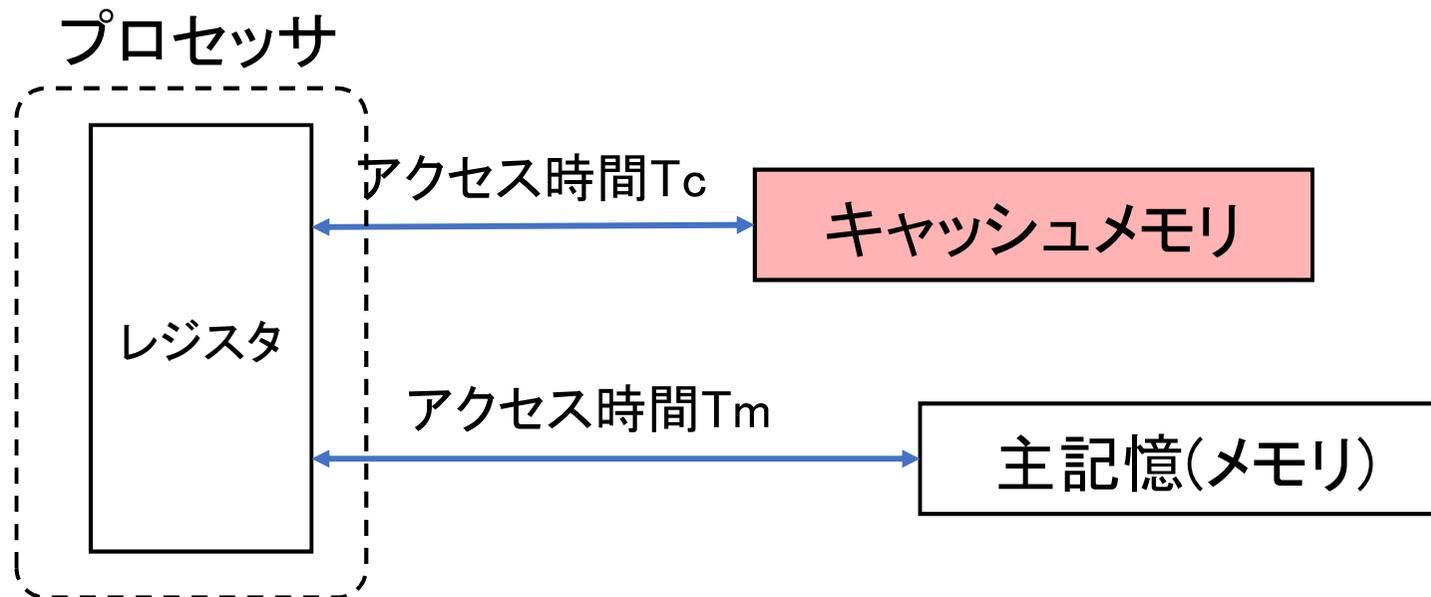
ライトバック方式

- 書き込みが速い



●平均命令実行時間

プロセッサが、目的のデータにアクセスできるまでの時間T



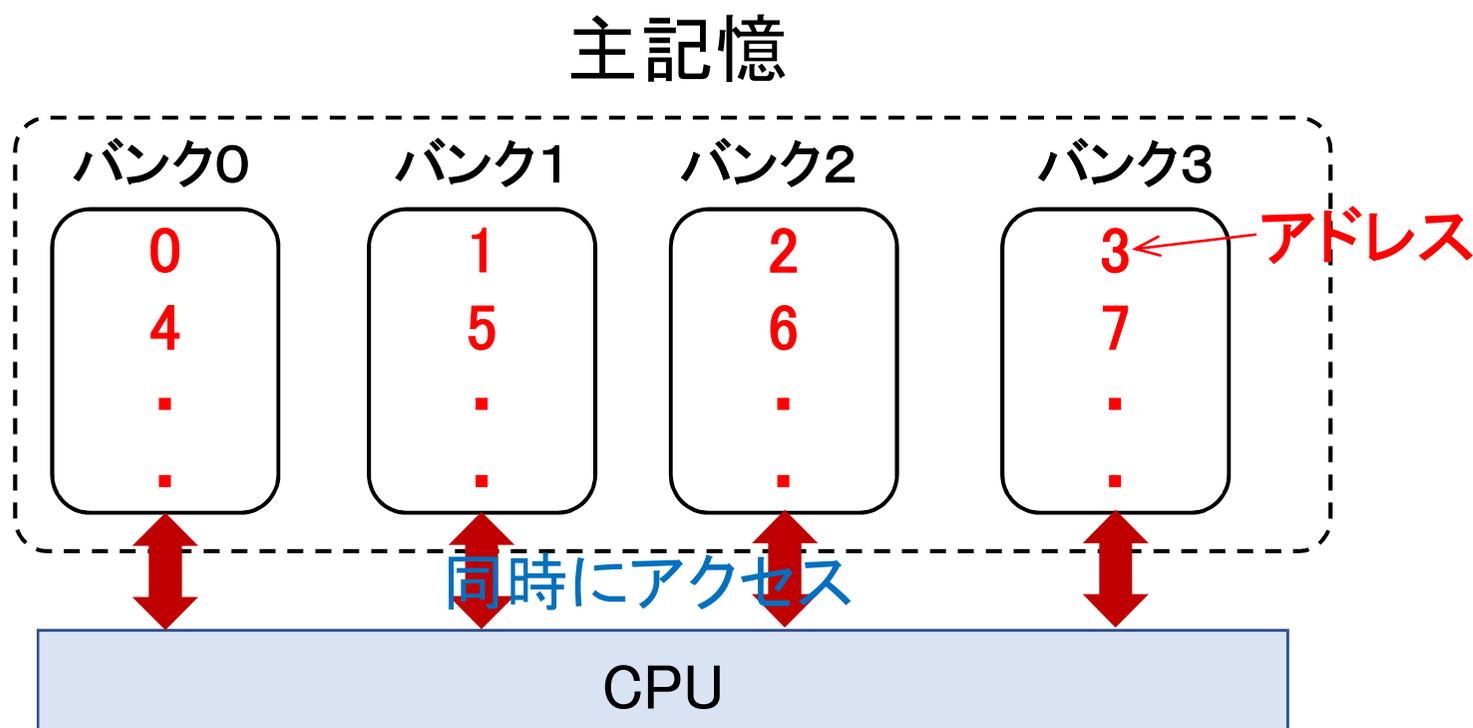
$$\begin{aligned} T &= T_c \times h + T_m \times (1 - h) \\ &= T_c \times (1 - n) + T_m \times n \end{aligned}$$

h:目的のデータがキャッシュメモリにある割合(ヒット率)

n:目的のデータがキャッシュメモリにない割合

●メモリアンタリーブ

主記憶を複数の区画(バンク)に分割し、連続したアドレスを割り当てて、複数のバンクを同時に並行的に読み出すことで、アクセス速度を向上させる



2.2.3 チェック方式

コンピュータが扱うデジタル信号は、伝送途中で雑音などの様々な要因によって、誤る(0→1,1→0)ことがある。

誤ったデジタル信号を、検出して訂正することを、**誤り制御**と呼ぶ。

●パリティチェック方式

データのビット列に**パリティビット**と呼ぶ検査用のビットを付加してデータの誤りを検出する

1ビットの誤りは検出できるが、訂正はできない

● 奇数(偶数)パリティ

ビット列に 1 の数が奇数(偶数)個になるように、0 または 1 を付加する

データ	0	1	1	0	0	0	0	
偶数パリティを使った場合	0	1	1	0	0	0	0	0
奇数パリティを使った場合	0	1	1	0	0	0	0	1

付加したパリティビット

● 垂直(水平)パリティ

- 垂直パリティ: 1文字ごとに奇数(または偶数)パリティビットを付加する
- 水平パリティ: 各文字の同じ位置にあるビットに対して奇数(または偶数)パリティビットを付加する

S	T	U	D	Y		
1	0	1	0	1	1	

水平パリティビット (偶数パリティ)

垂直パリティビット (偶数パリティ)

S	1
	1
	0
	0
	1
	0
	1
	0

● 水平垂直パリティ

垂直パリティと水平パリティを組み合わせる

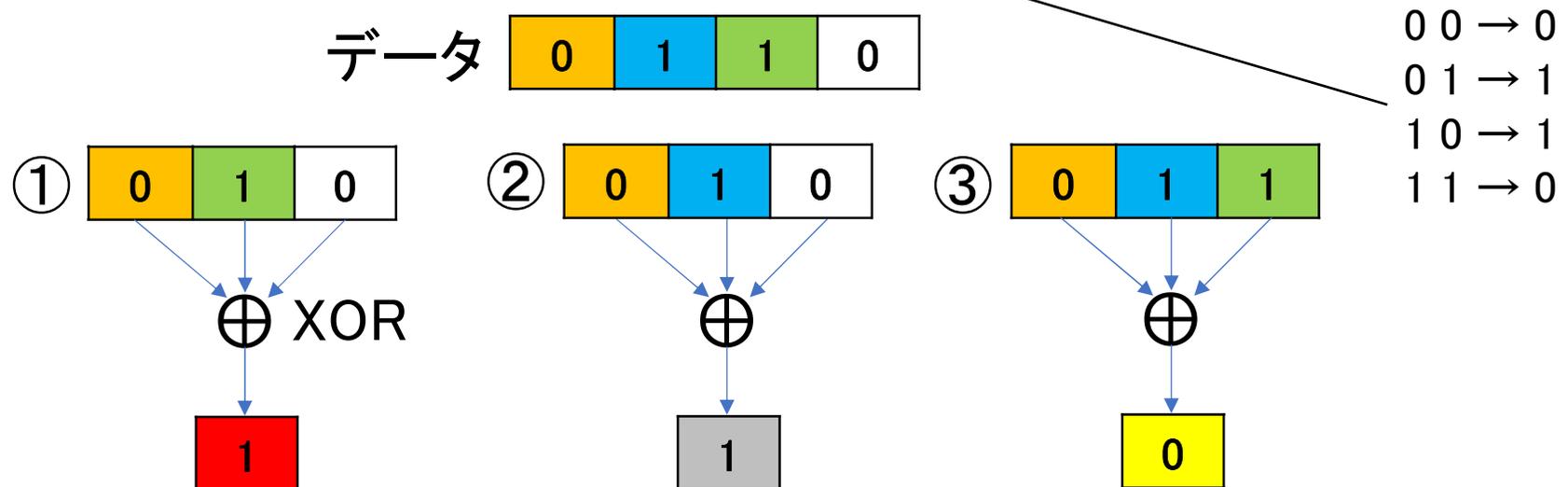
水平パリティビット(偶数パリティ)

	S	T	U	D	Y	
	1	0	1	0	1	1
	1	0	0	0	0	1
	0	1	1	1	0	1
	0	0	0	0	1	1
	1	1	1	0	1	0
	0	0	0	0	0	0
	1	1	1	1	1	1
垂直パリティビット(偶数パリティ) →	0	1	0	0	0	0

↓

● ハミング符号

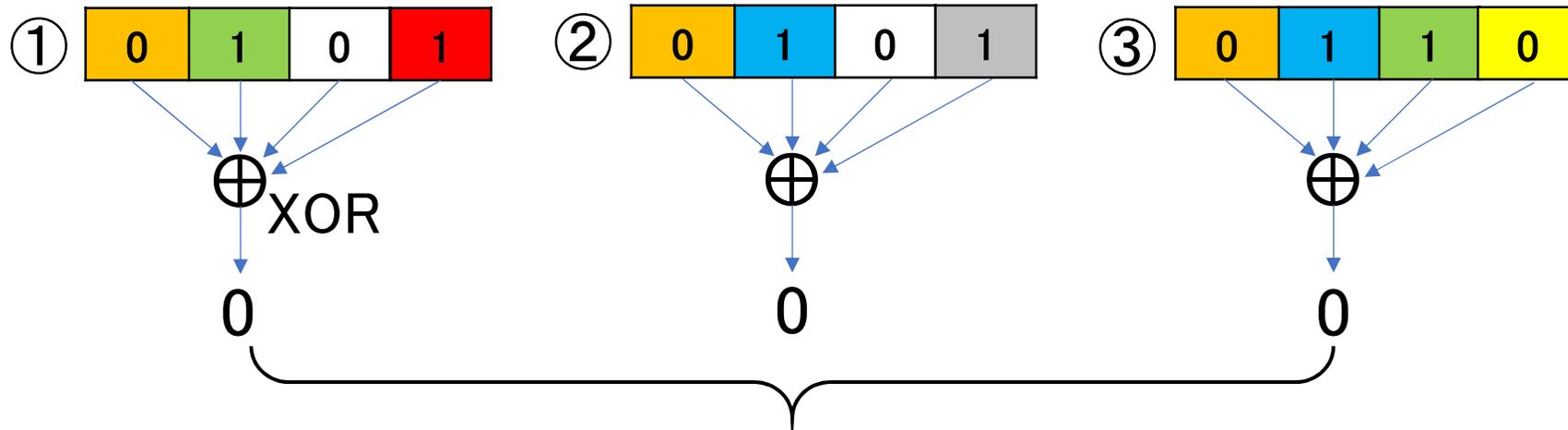
- データに誤りの有無と位置を表す **冗長符号** を付加する
 - 主メモリの **誤り制御** に用いられることが多い
 - **2ビットの誤り検出** ができ、**1ビットの訂正** ができる
- a. データから異なる組み合わせのパターン①～③を作り、このパターンの 排他的論理和(XOR) を、冗長符号とする



b. 作成した冗長符号とデータをくっつけて伝送データとする



c. 受け取り側では、受け取ったデータから組み合わせ①～③を作り、これと対になる冗長ビットを組みあわせて排他的論理和を求める



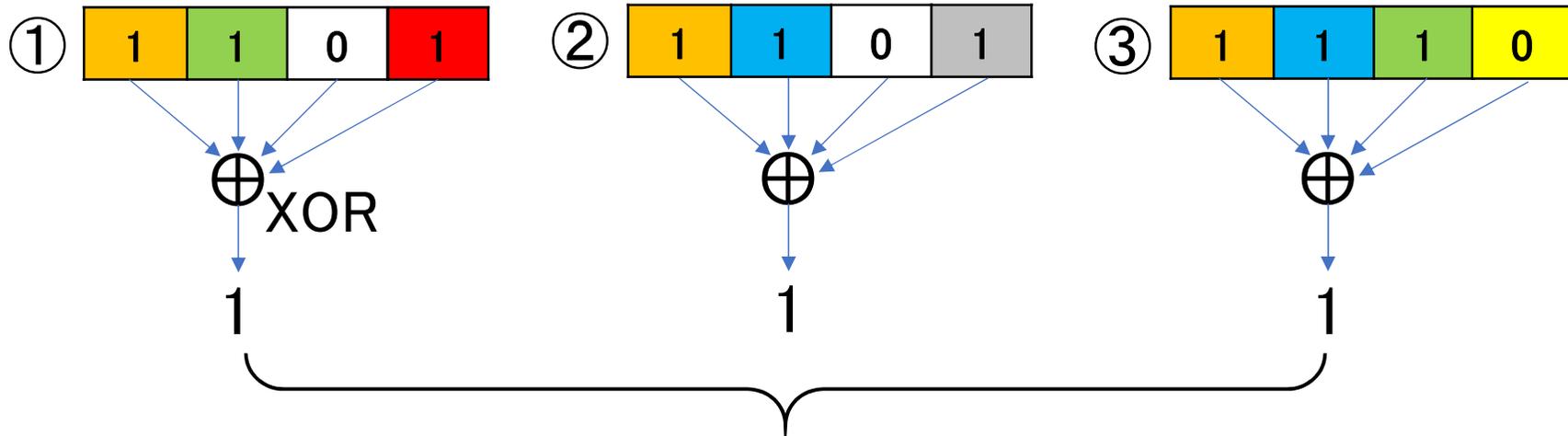
全ての値が0(ゼロ)ならば、データには誤りなし

d. 受け取り側で、もしデータに誤りがあれば、以下のようなになる

受信データ

1	1	1	0	1	1	0
---	---	---	---	---	---	---

誤り
0→1



全ての値が1であるので、組み合わせパターン①～③で共通に使われているビット

1

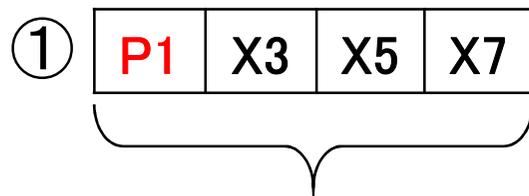
 に誤りあり

テキストでは

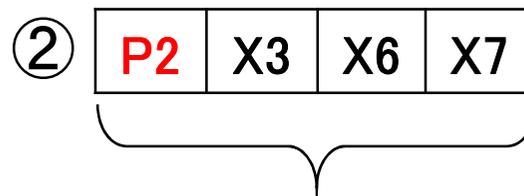
- a. データビット列 $X_3X_5X_6X_7$ とパリティビット列 $P_1P_2P_4$ から異なる組み合わせのパターン①～③を作り、このパターンで偶数パリティとなるように、 $P_1P_2P_4$ を求める

データビット列

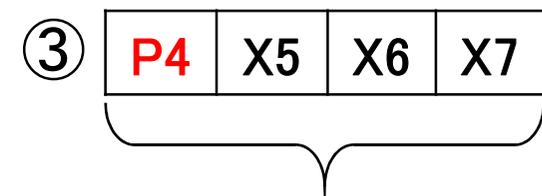
X3	X5	X6	X7
----	----	----	----



偶数パリティとなるようにP1を求める



偶数パリティとなるようにP2を求める



偶数パリティとなるようにP4を求める

- b. 求めたパリティビット列 $P_1P_2P_4$ とデータビット列 $X_3X_5X_6X_7$ をくっつけて伝送するデータビット列を作る

伝送するデータビット列

P1	P2	X3	P4	X5	X6	X7
----	----	----	----	----	----	----

- c. 受け取り側では、受け取ったデータビット列から組み合わせ
①～③を作り、これと対になるビットを組みあわせて「排他的
論理和」または「全ビットの加算値を2で割った余り」を求める

$$\textcircled{1} (P1 \oplus X3 \oplus X5 \oplus X7) \text{ または } \text{mod}((P1+X3+X5+X7),2)$$

$$\textcircled{2} (P2 \oplus X3 \oplus X6 \oplus X7) \text{ または } \text{mod}((P2+X3+X6+X7),2)$$

$$\textcircled{3} (P4 \oplus X5 \oplus X6 \oplus X7) \text{ または } \text{mod}((P4+X5+X6+X7),2)$$

これらの組み合わせ①～③の値がすべて0(ゼロ)ならば、
データビット列には誤りなし

- d. これらの組み合わせ①～③の値がすべて1ならば、共通に
使われているビットX7に誤りあり

● CRC(巡回冗長検査) : **C**yclic **R**edundancy **C**heck

- データのビット列を特定の式(生成多項式)で割り、その余りをチェックコードとして、データに付加する
- 伝送路(通信回線)の誤り検出に適している
- ビットの訂正はできないが、連続したビット誤りやランダム誤りを検出できる

$$\boxed{1\ 0\ \dots\dots\dots\ 1} \div \boxed{\text{生成多項式}} = \text{商} + \text{余り}$$

送信 $\boxed{1\ 0\ \dots\dots\dots\ 1\ \text{余り}}$

受信 $\boxed{1\ 0\ \dots\dots\dots\ 1\ \text{余り}} \div \boxed{\text{生成多項式}} = \text{商} + \text{余り}$

0ならばエラーなし
0以外ならばエラーあり

■ 過去問題2

キャッシュの書込み方式には、ライトスルー方式とライトバック方式がある。ライトバック方式を使用する目的として、適切なものはどれか？

ア: キャッシュと主記憶の一貫性(コヒーレンシ)を保ちながら、書込みを行う

イ: キャッシュミスが発生したときに、キャッシュの内容の主記憶への書き戻しを不要にする

ウ: 個々のプロセッサがそれぞれのキャッシュをもつマルチプロセッサシステムにおいて、キャッシュ管理をライトスルー方式よりも簡単な回路構成で実現する

エ: プロセッサから主記憶への書込み頻度を減らす

ライトスルー方式とライトバック方式には、次のような違いがある

- ライトスルー方式(write through)

CPUから書き込む命令が出たときにキャッシュメモリと同時に主記憶にも書き込む方式。データの整合性は得られるが処理速度は低下する。

→常に主記憶とキャッシュの内容が一致するため一貫性の確保は容易だが、主記憶への書き込みが頻繁に行われるので遅い

- ライトバック方式(write back)

CPUから書き込む命令が出たときにキャッシュメモリだけに書き込む方式。主記憶への書き込みはキャッシュメモリからデータが追い出されるときに行われる。

→主記憶とキャッシュの内容が一致しないため一貫性を保つための制御が複雑になるが、主記憶への書き込み回数が少ないため速い

ア:これは、ライトスルー方式を使用する目的。ライトバック方式では、キャッシュと主記憶の内容が常に一致しているわけではない。

イ:これは、ライトスルー方式を使用する目的。キャッシュミスが発生すると、キャッシュ内の必要のないデータを主記憶に移動し、要求されたデータを主記憶からキャッシュにコピーする。ライトバック方式では該当データがキャッシュから追い出されたタイミングで主記憶に書き出すので、一貫性を保つために書き戻し処理が必要となる。

ウ:これは、ライトスルー方式を使用する目的です。ライトスルー方式と比較した場合、ライトバック方式の同期制御は複雑になる。

2.3 入出力装置と入出力デバイス

コンピュータと周辺機器を接続するために作られた規格やデータやプログラムを入力・出力するための機器

2.3.1 RAID

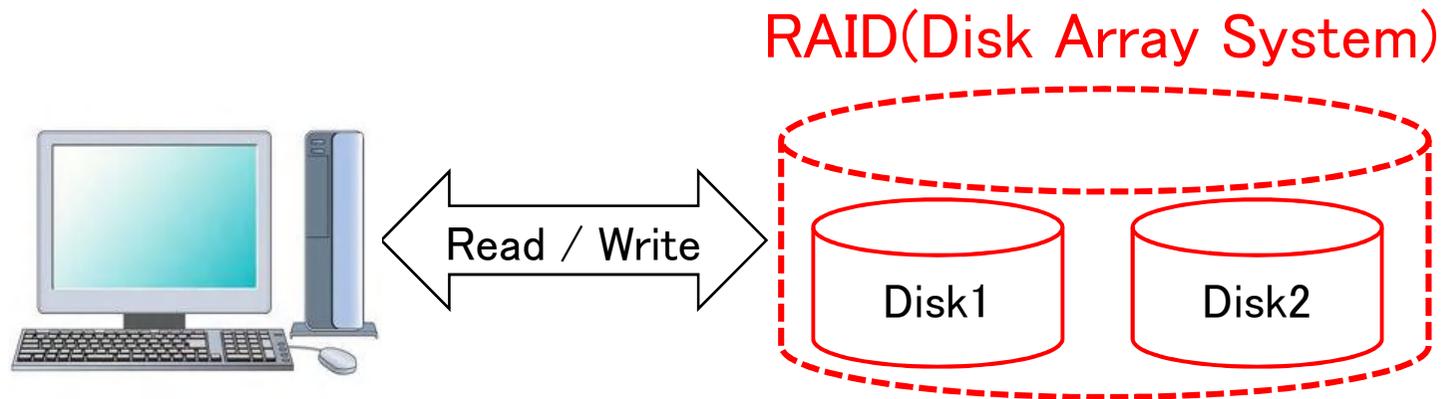
2.3.2 入出力インタフェース

2.3.3 補助記憶装置

2.3.1 RAID

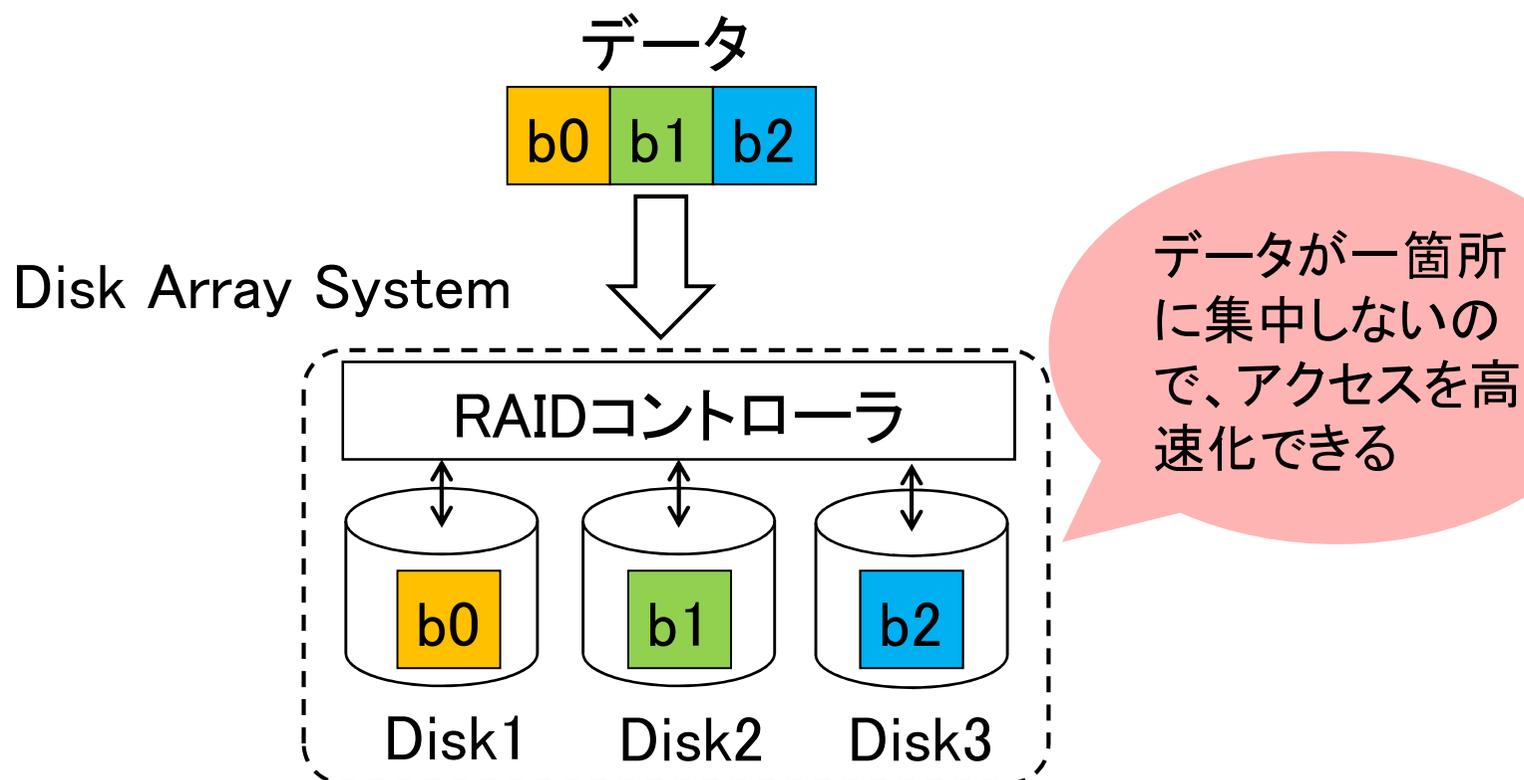
(Redundant Arrays of Inexpensive Disks)

複数のDiskを組み合わせることで、Diskの速度や信頼性を向上させる



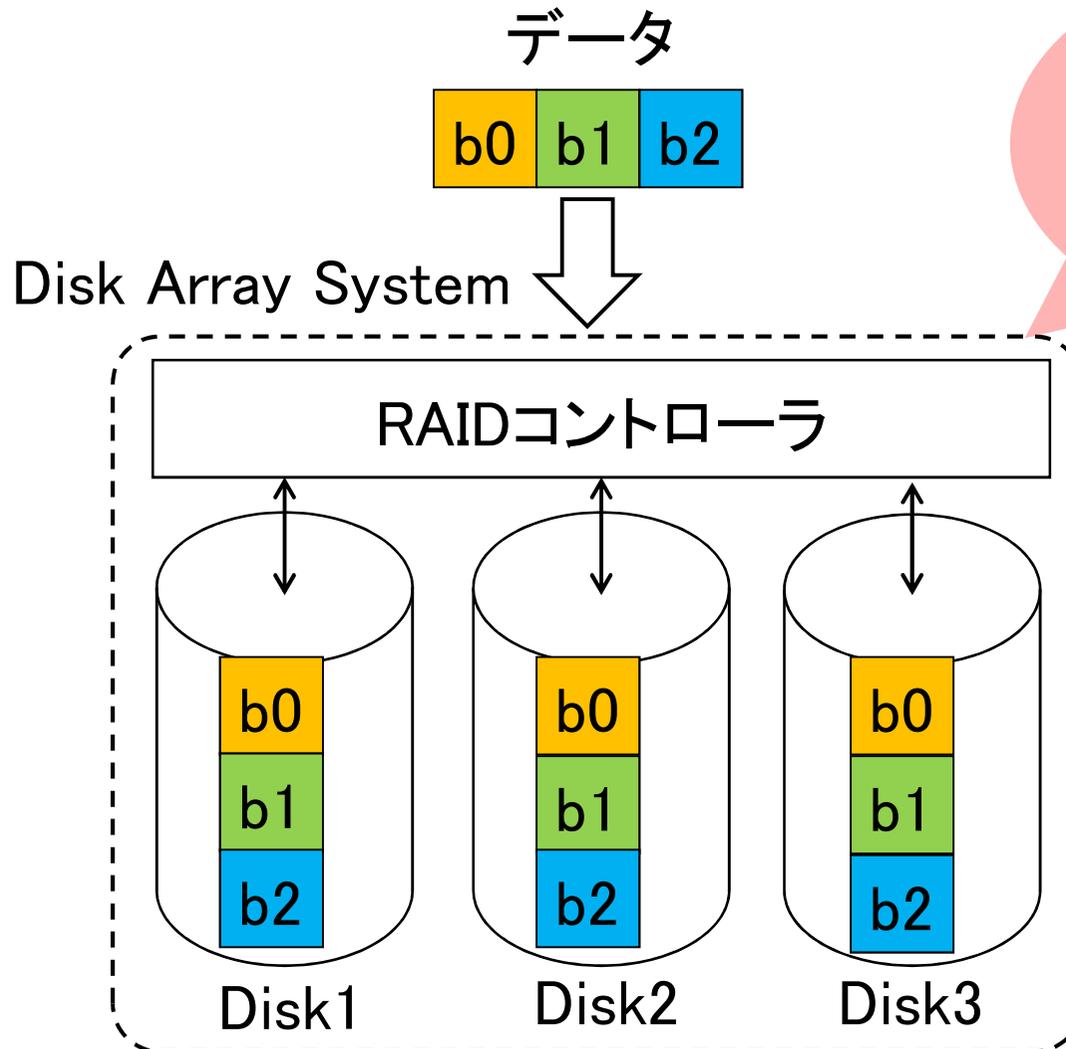
● RAID0（ストライピング）

複数のディスクに、データをブロック単位に分散してデータを書き込む



● RAID1 (ミラーリング)

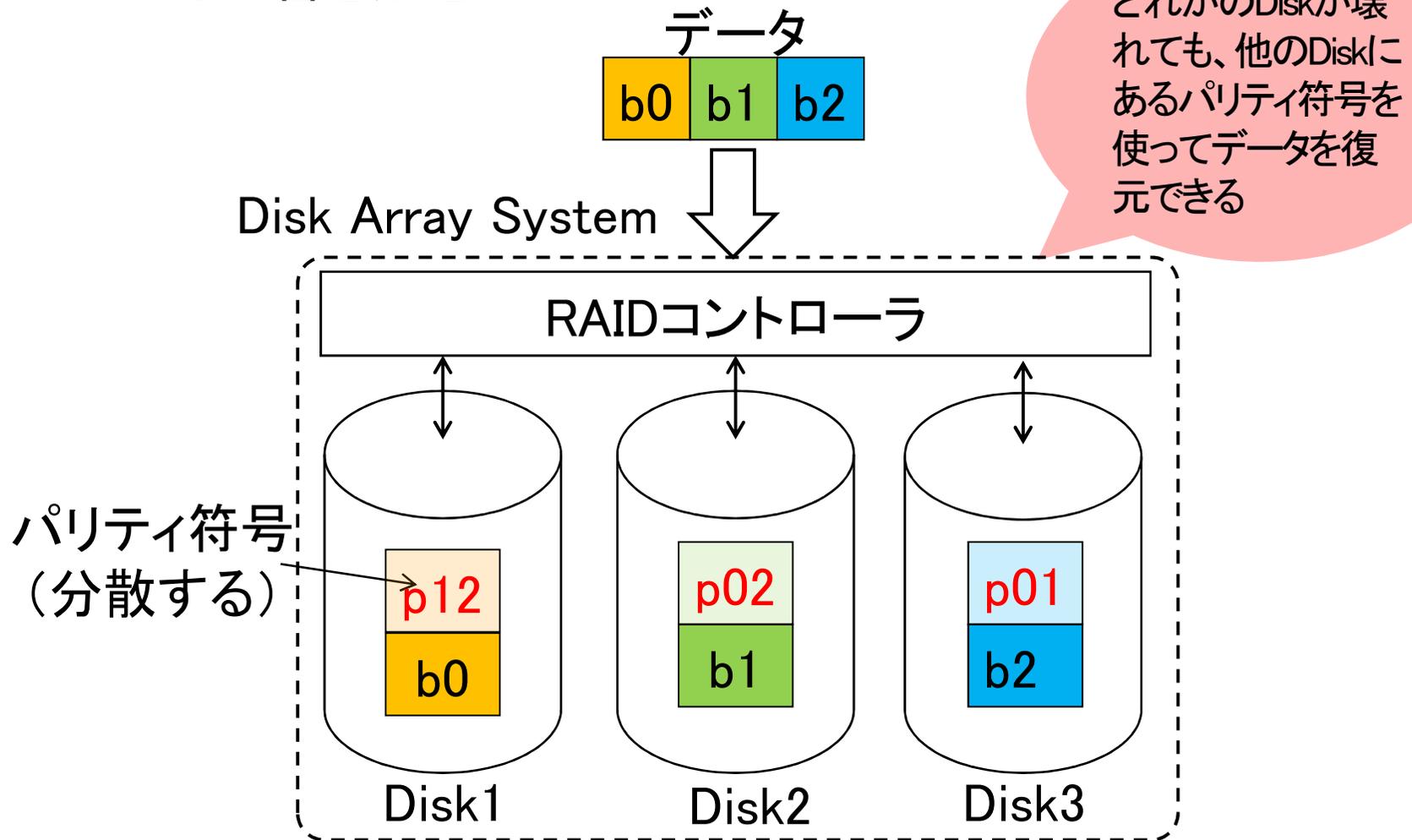
複数のディスクに、同時に同じデータを特定サイズで書き込む



同じデータが複数個所に記録されるので、データの信頼性が向上する

● RAID5 (パリティ付きストライピング)

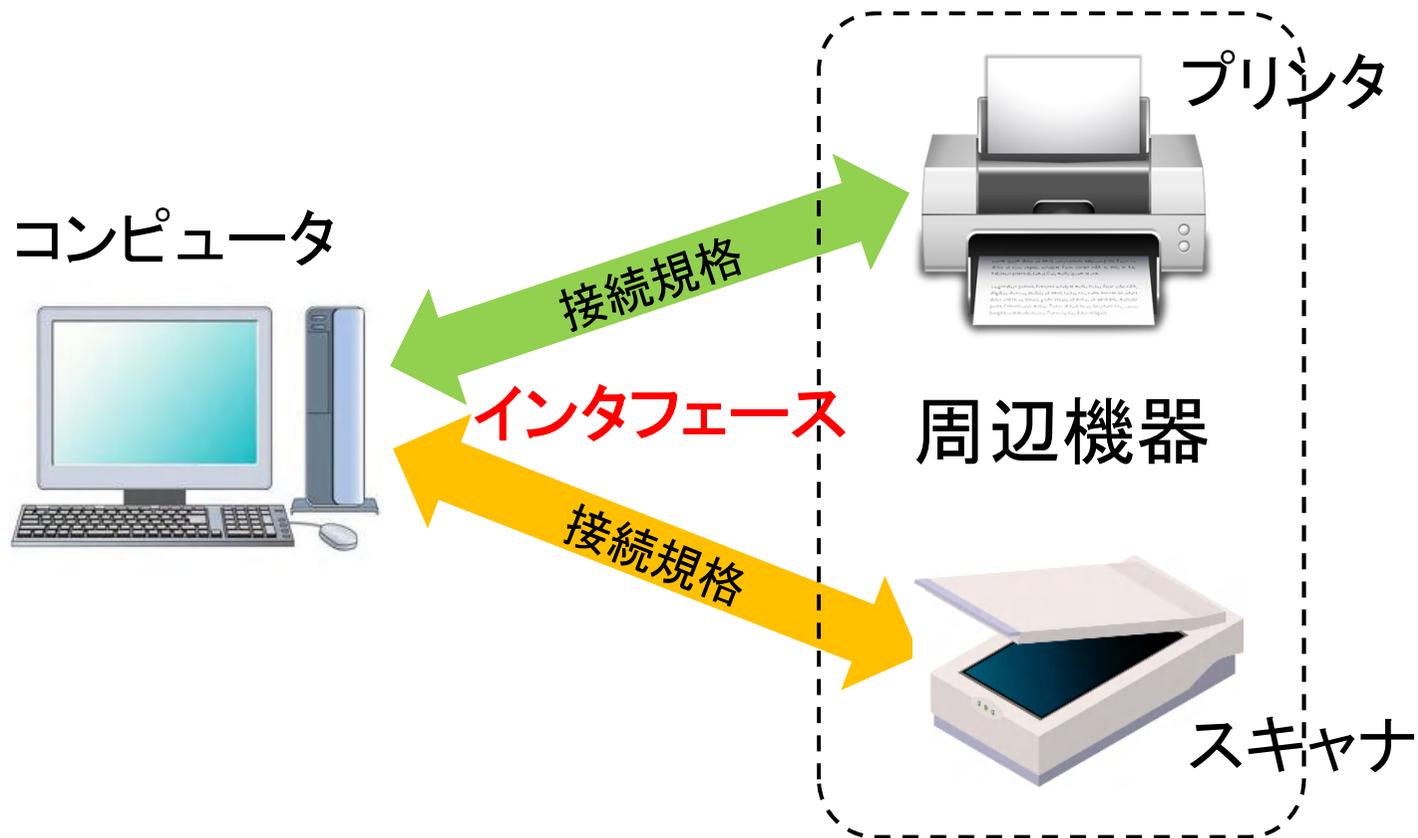
複数のディスクに分散してデータをブロック単位に書き込むと同時に、エラー検出用のパリティ符号を付加して書き込む



2.3.2 入出カインタフェース

コンピュータと周辺機器を接続する規格

- シリアル伝送(データを1ビットずつ送る)
- 平行伝送(データを数ビットまとめて送る)



インタフェースの種類

シリアル伝送

type-A



type-B



type-C



● USB(Universal Serial Bus)

一番よく使われているインタフェースで、USBハブを使って最大127台接続可能。ホットプラグ機能(電源を入れたまま抜き差し可能)やプラグアンドプレイ機能(周辺機器を接続すると自動で設定が開始する)がある。

<USB1.1>ロースピードモード:1.5Mb/s,フルスピードモード:12Mb/s

<USB2.0>ハイスピードモード:480Mb/s

<USB3.0>スーパースピードモード:5Gb/s

<USB3.1>スーパースピードプラスモード:10Gb/s

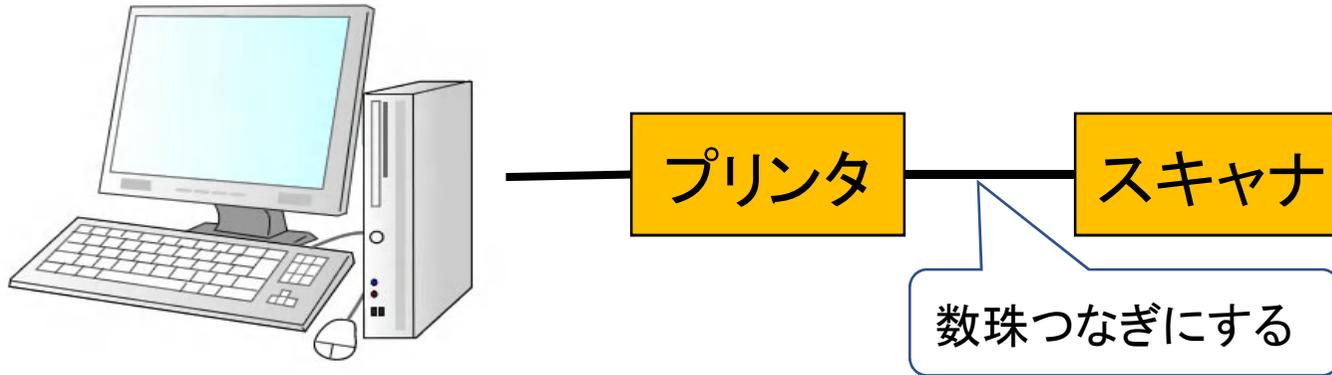
<USB3.2>スーパースピードプラスモード:20Gb/s※2017年施行

● IEEE1394(別名:FireWire,i-Link)

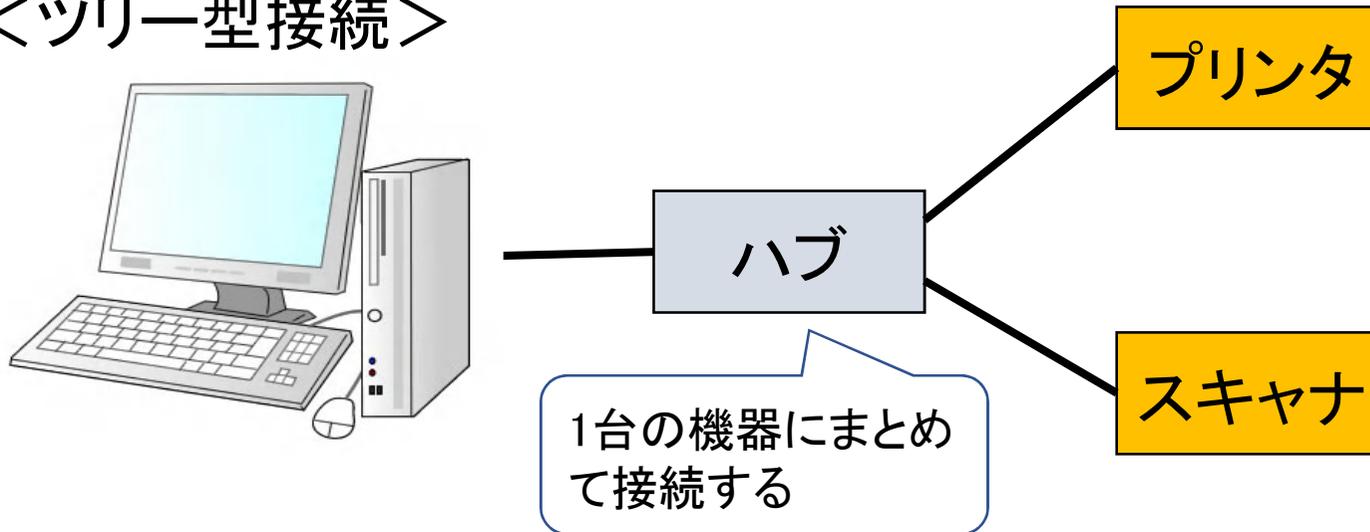
USBよりも転送速度(100Mb/s,200Mb/s,400Mb/s)が速いので、デジタルカメラやビデオなどで利用されている。ツリーで63台、デイジーチェーンで17台が接続可能。ホットプラグ機能(電源を入れたまま抜き差し可能)やプラグアンドプレイ機能(周辺機器を接続すると自動で設定が開始する)がある。

■ 複数機器の接続方法

< デイジーチェーン接続 >



< ツリー型接続 >



● Bluetooth



電波(2.4GHz)を使って、数10mの距離で対応機器(ヘッドフォンやマウスなど)を接続する。スマートフォンやゲーム機器でも利用されている。

● SATA(Serial ATA)

光ディスクなどの周辺機器をPC本体内部で接続するための高速インターフェース。ホットプラグ機能がある。

<SATA1.0>1.5Gb/s <SATA2.0>3Gb/s <SATA3.0>6Gb/s
<eSATA>3Gb/s,6Gb/s

● ZigBee

センサーネットワークを主目的とする近距離無線通信規格で、転送可能距離が短く(数10m)転送速度も非常に低速(数10kb/s～数100kb/s)である代わりに、安価で消費電力が少ないという特徴を持つ。従って、電池駆動可能な超小型機器への実装に向いている。

平行伝送

● SCSI(Small Computer System Interface)

ハードディスクやプリンタなどをPC本体と外付接続するための規格。現在は、USBに置き換わっている。デジチェーンで17台が接続可能。シリアル伝送の規格もある。

2.3.3 補助記憶装置

主記憶装置の容量の補足、データの持ち運び、データのバックアップなどで使われる

種類	特徴	用途
磁気ディスク	薄い円盤状の(金属製)ディスクに磁気を使ってデータを書き込む	ハードディスク 
光ディスク	薄い円盤状の(ポリカーボネート製)ディスクにレーザ光でデータを書き込む	CD,DVD,BD 
フラッシュメモリ	電気で書込みや消去を行う半導体メモリ。コンパクトでアクセス速度が速い	SDカード,USBメモリ,SSD,コンパクトフラッシュ 

● 光ディスク

- 再生専用型、追記型、書換え可能型がある
- 大容量で、携帯性に向いている
- ソフトウェアの配布やバックアップなどに用いられる

種類	入出インタフェース	
再生専用型	読込のみ	CD-ROM,DVD-ROM
追記型	追加のみ(上書不可)	CD-R,DVD-R,DVD+R,BD-R
書換え可能型	一括消去して上書き	CD-RW,DVD-RAM, DVD+RW,BD-RE

- CD-ROM(**C**ompact **D**isk-**R**OM),CD-R(**C**D **R**ecordable),
CD-RW(**C**D **R**ewritable)
 - 1枚のディスクで、700MB程度の記憶容量
 - CD-Rは、国際規格ISO9660のフォーマットで使用
- DVD(**D**igital **V**ersatile **D**isk),BD(**B**lue-ray **D**isk)
 - DVDは、動画像データや音声データなどの大容量のデータ(約5GB~10GB)を記録できる
 - BDは、DVDよりも大容量のデータ(25GB~100GB)を記録できる

■過去問題3

80Gバイトの磁気ディスク8台を使用して、RAID0の機能とRAID1の機能の両方の機能を同時に満たす構成にした場合、実効データ容量は何Gバイトか？

ア: 320 イ: 480 ウ: 560 エ: 640

RAID0(ストライピング)

複数のディスクに分散してデータを書込むことで、アクセス性能を向上させる

RAID1(ミラーリング)

同じデータを2台のディスクに書き込むことで、信頼性を向上させる。実質記憶容量は50%となるので記憶効率は悪い。

RAID0はディスクにデータを分散して書き込むだけなので、磁気ディスクの容量をそのまま使用できますが、RAID1では同じデータを2か所に格納しておくため、実質記憶容量は半分になる。

したがって実効データ容量は、

$$80\text{Gバイト} \times 8\text{台} \div 2 = 320\text{Gバイト}$$