計算機システムⅡ
主記憶装置とALU，レジスタの制御

和田俊和

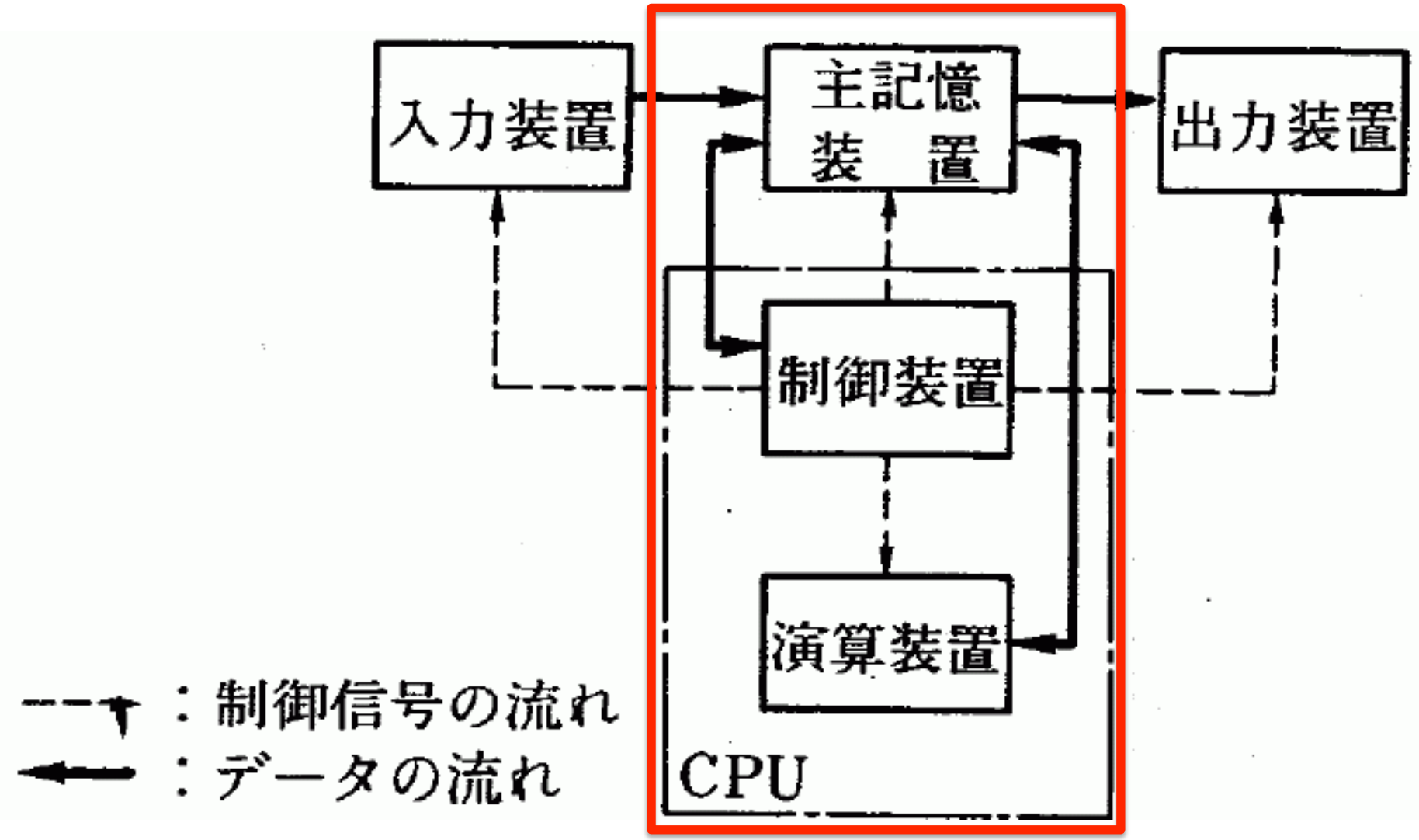
講義計画

1. コンピュータの歴史1
2. コンピュータの歴史2
3. コンピュータの歴史3
4. 論理回路と記憶，計算：レジスタとALU
5. 主記憶装置とALU，レジスタの制御（←本日）
6. 命令セットアーキテクチャ
7. 演習問題
8. パイプライン処理
9. メモリ階層：キャッシュと仮想記憶
10. 命令レベル並列処理
11. 命令実行順序の変更
12. 入出力と周辺装置：DMA,割り込み処理
13. 演習問題
14. 現代的な計算機アーキテクチャの解説
15. 総括と試験

- 教科書：坂井修一著：電子情報通信学会レクチャーシリーズC−9，コンピュータアーキテクチャ，コロナ社

- 最終回の試験によって成績評価を行う．5回以上欠席で不合格とする．

本日の講義の範囲



２．１　主記憶装置

2.1.1レジスタとALUだけでは計算はできない

1. 大量のデータはレジスタだけではまかなえない.
2. 制御信号の生成ができない.
3. 条件分岐や，繰り返し計算は，このままでは実行できない.

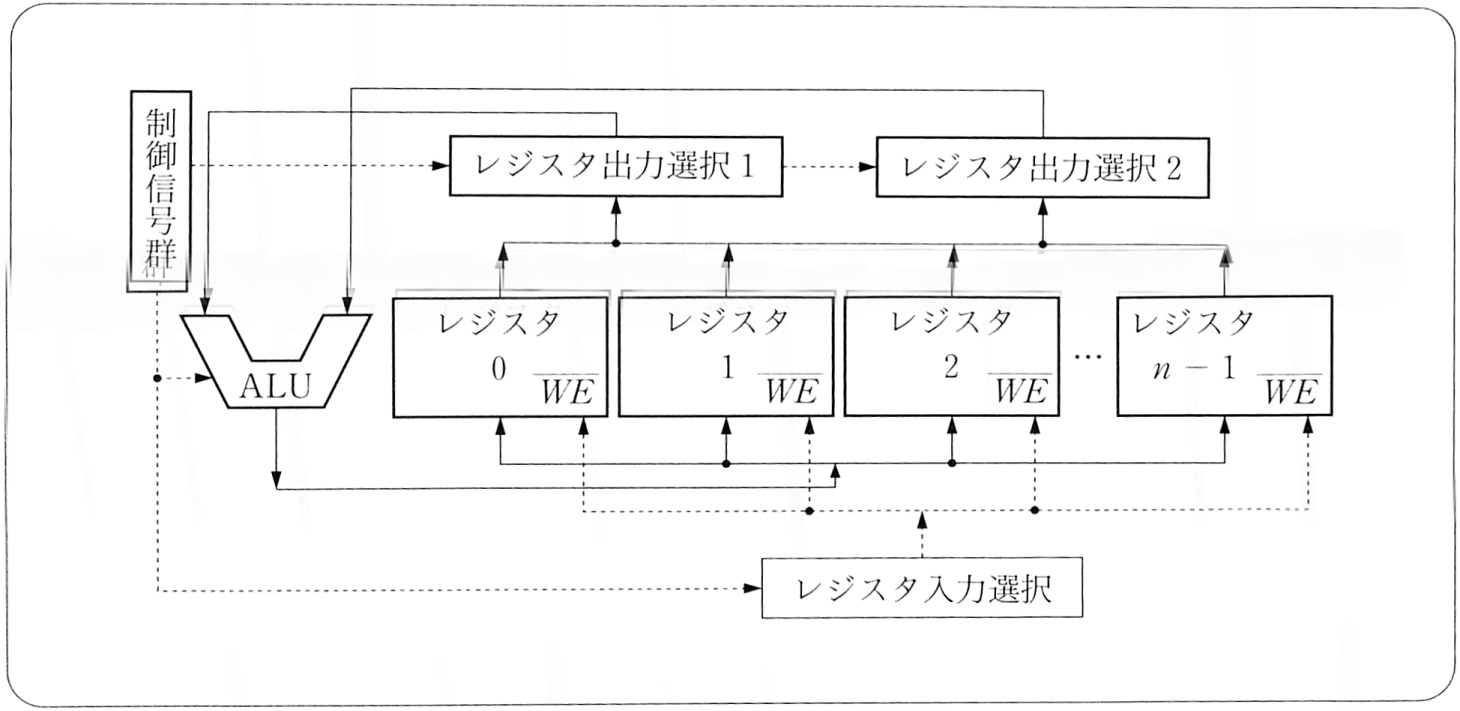


図 1.13　コンピュータの演算のサイクル

2.1.2 主記憶装置

1. 読み出し: 主記憶→レジスタ
2. 計算: レジスタ(群)→ALU→レジスタ
3. 書き込み: レジスタ→主記憶

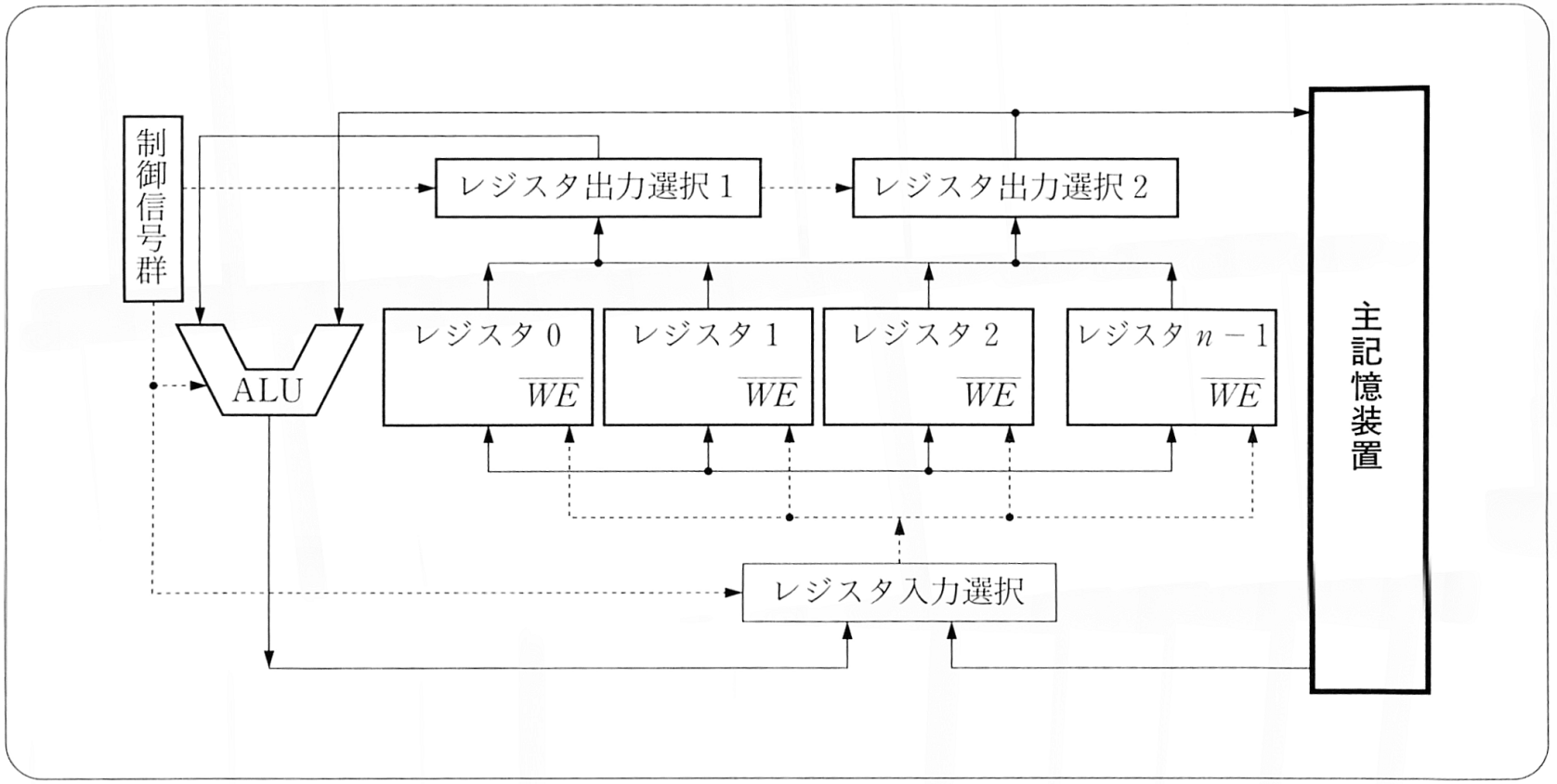


図 2.1 主記憶装置を含む演算実行機構

2.1.3 メモリの構成

- アドレス線の値（A_0～A_{n-1}）により，アドレス（0～2^n-1）のいずれかがアクティブになる．
- チップ選択信号が与えられ，
  - 読み出しを指示した場合は，データ線に選択された語の内容が出力される．
  - 書き込みの場合，データ線の内容が選択された語に書き込まれる．

実際のメモリは，複数のチップから構成されており，アドレス線をデコードすることにより，チップ選択信号が生成されている．このときに，連続するアドレスの記憶内容が同じチップに記録されないようにすることで，高速なメモリアクセスができるように工夫されている．

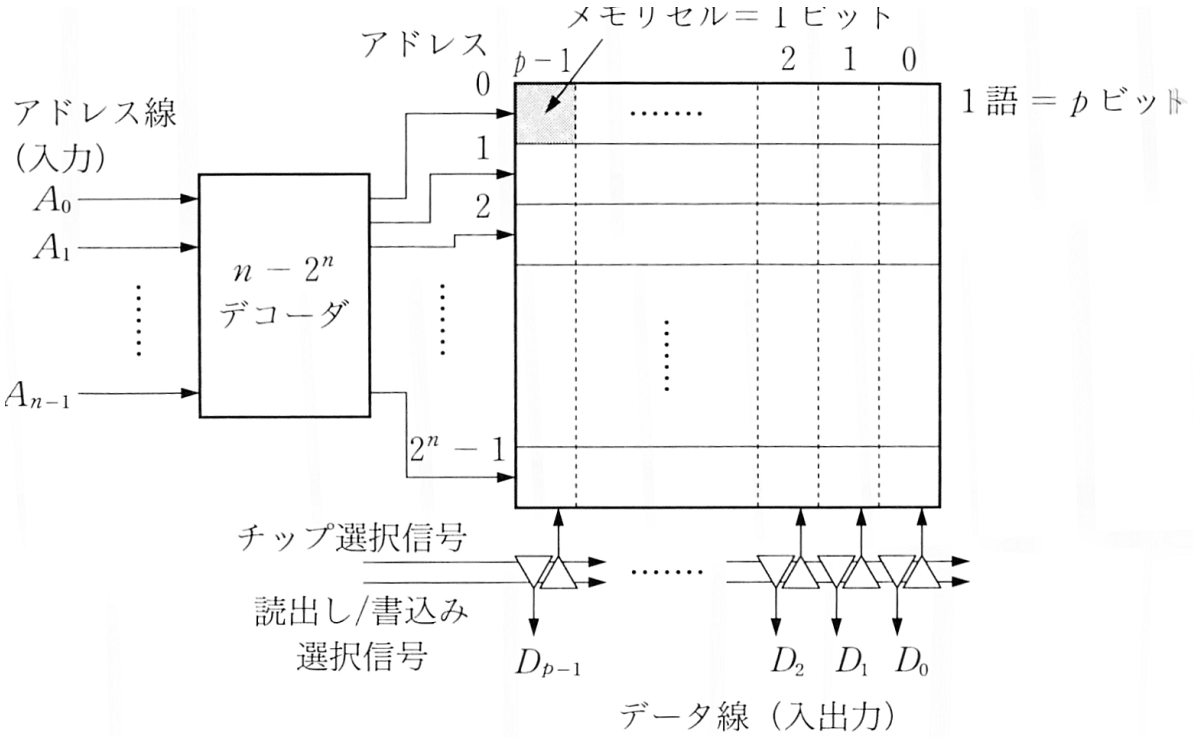


図 2.2　メモリの構成

備考:「語」「ワード」「word」

- これは記憶の単位.
- p-bitが集まって，1語（ワード，word）になる.
- 計算機アーキテクチャ上では「データ線の本数」,「メモリの記憶単位」,「演算用レジスタのbit数」「ALUの入力ビット幅」などと同義である.
- これは「バイト」「byte」という一般的な情報量の単位ではなく，アーキテクチャ上の単位である.

n→2^nデコーダ（3→8）

$$Y_0 = \overline{A_2} \wedge \overline{A_1} \wedge \overline{A_0} \quad (000)$$
$$Y_1 = \overline{A_2} \wedge \overline{A_1} \wedge A_0 \quad (001)$$
$$Y_2 = \overline{A_2} \wedge A_1 \wedge \overline{A_0} \quad (010)$$
$$Y_3 = \overline{A_2} \wedge A_1 \wedge A_0 \quad (011)$$
$$Y_4 = A_2 \wedge \overline{A_1} \wedge \overline{A_0} \quad (100)$$
$$Y_5 = A_2 \wedge \overline{A_1} \wedge A_0 \quad (101)$$
$$Y_6 = A_2 \wedge A_1 \wedge \overline{A_0} \quad (110)$$
$$Y_7 = A_2 \wedge A_1 \wedge A_0 \quad (111)$$

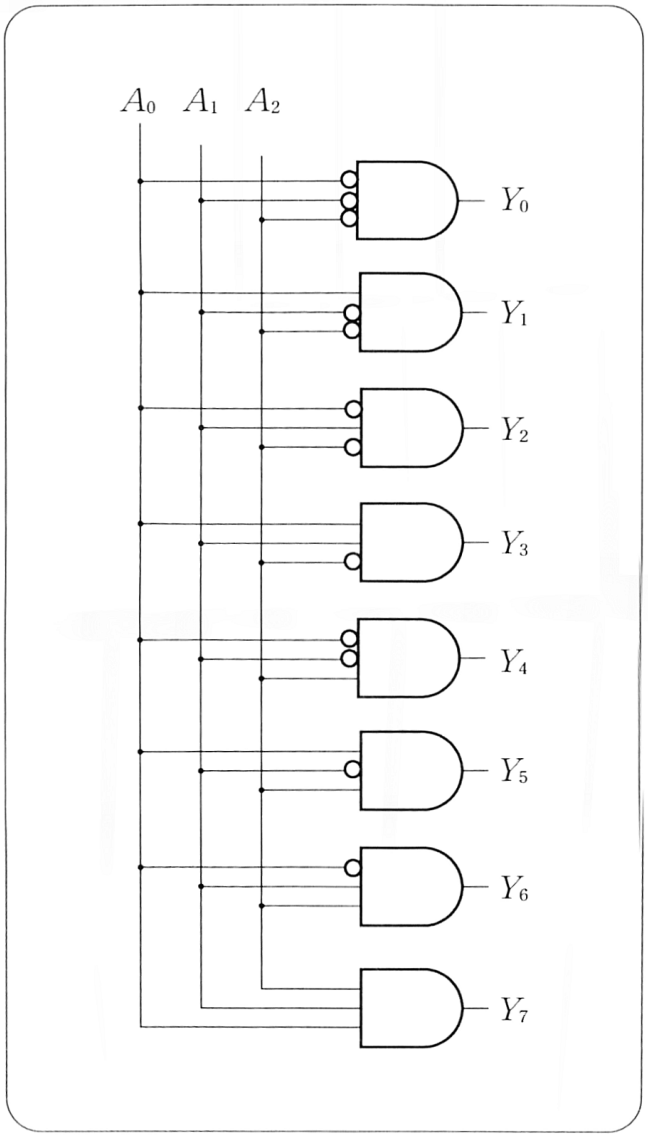


図 2.3　3 入力 8 出力デコーダ

1メガワードのメモリには，20→2^{20}のデコーダが必要．

1語＝1-byteの場合，4Giga-byte=2^{32}のメモリを扱うために必要なアドレス線の本数は，32本．

現行の32bitCPUは，データ線の本数とアドレス線の本数は同じで，メモリの最大は4Gが上限．PAEがない場合．

2.1.4 メモリの分類

- ROM
  - マスクROM
  - PROM
    - ヒューズROM
    - EPROM
- RAM
  - SRAM
  - DRAM

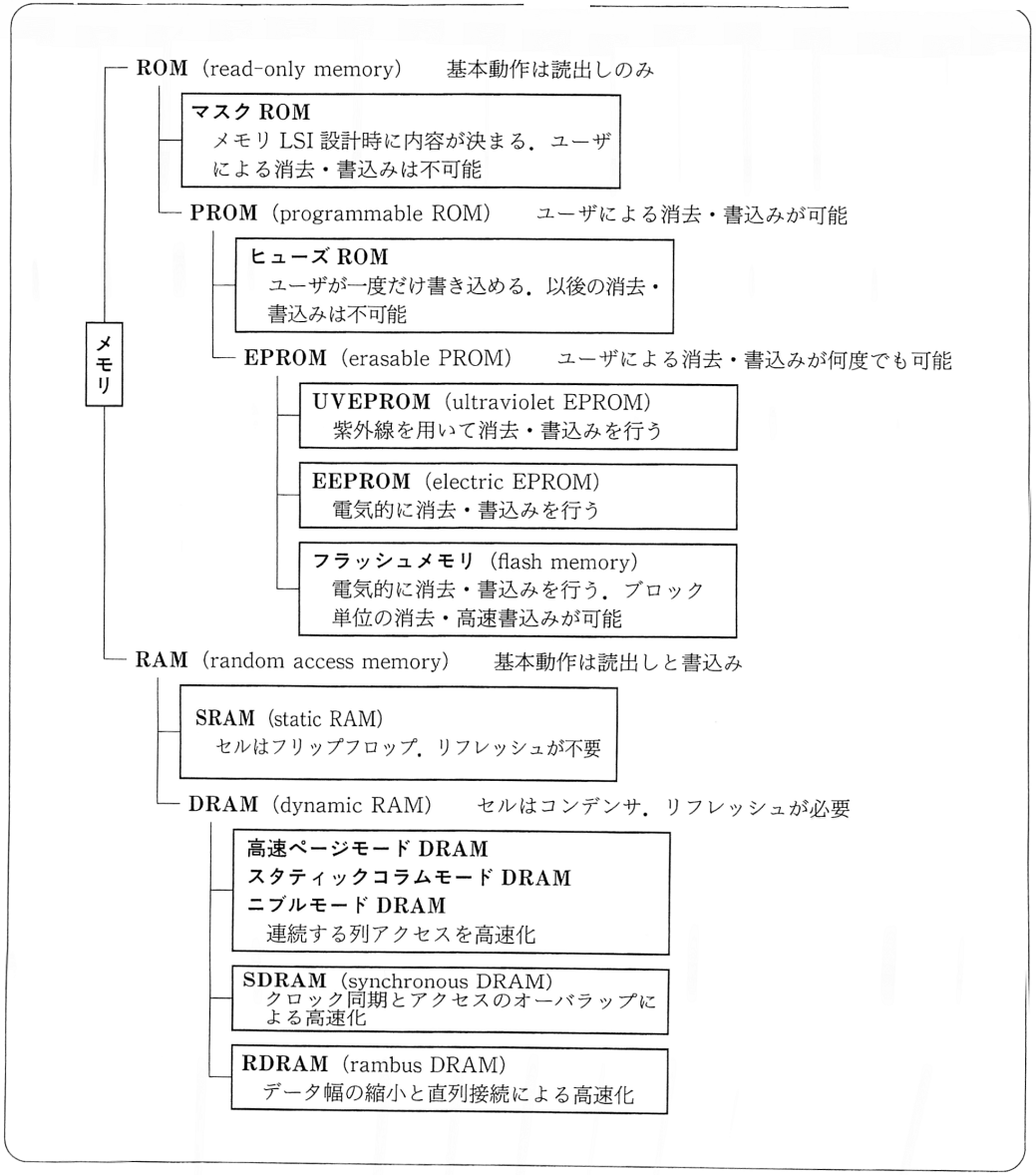


図 2.4　メモリの分類

2.1.5 レジスタファイル

- アドレス付けされたレジスタ群（通常32個程度）

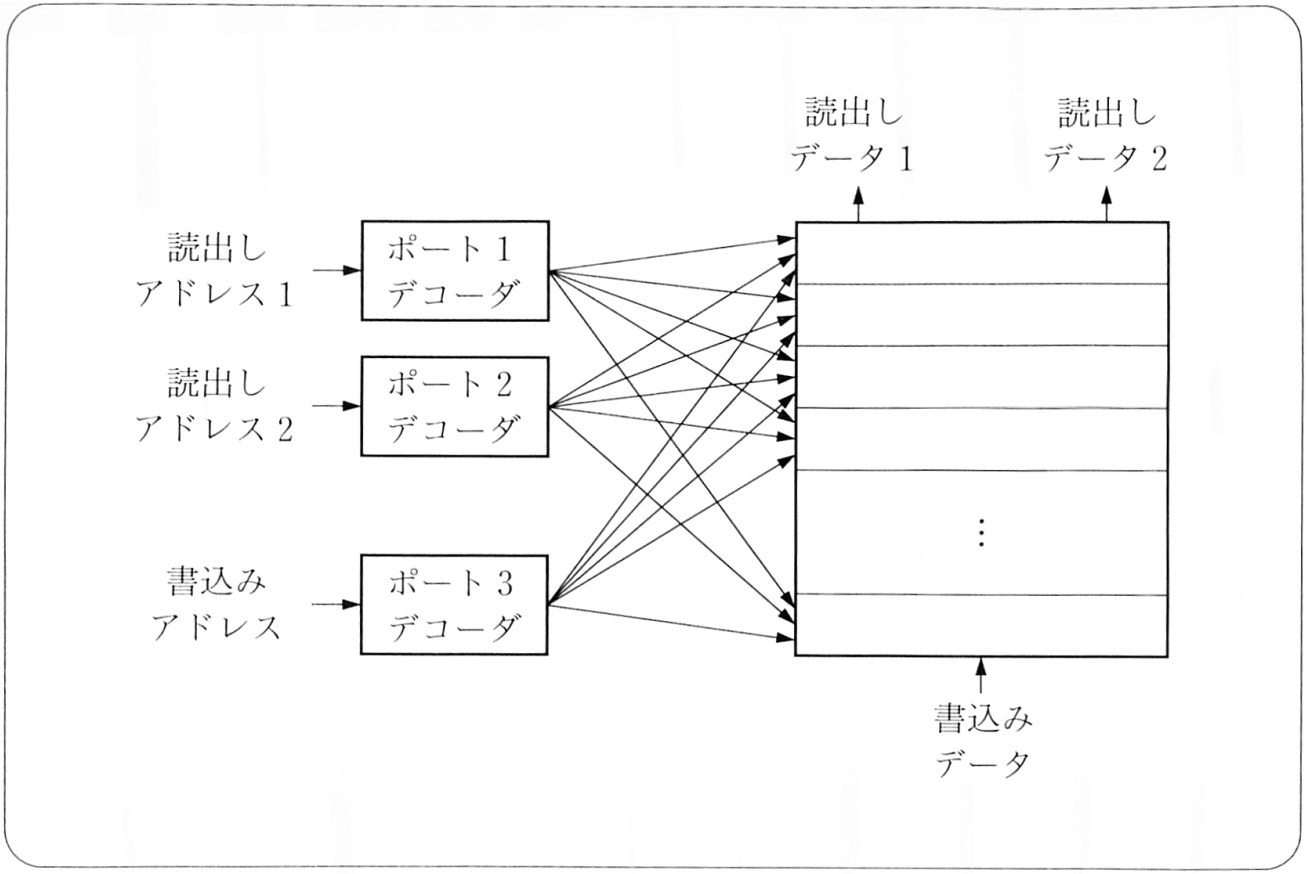


図 2.5　レジスタファイルの構成

レジスタはアドレスを指定してアクセスされる.

SRAMで構成され，書き込みと，2並列以上の読み出しとができる.（ALUに対応）

2.1.6 主記憶装置の接続

- レジスタ群→レジスタファイル
- メモリアドレスとメモリ制御が追加

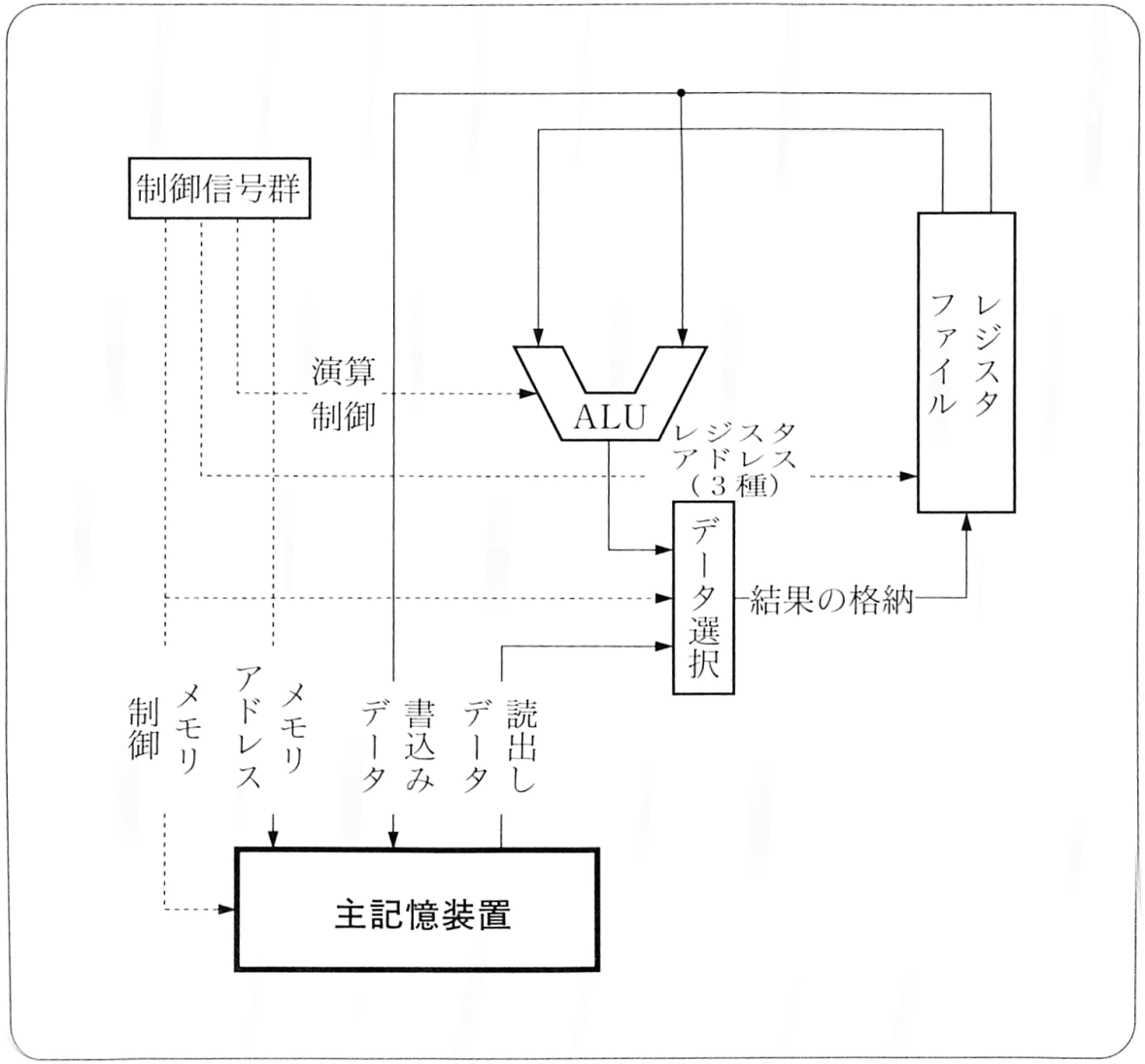


図 2.6　コンピュータの実行機構

2.2 命令とは何か

2.2.1 命令

- 制御信号を生成して，コンピュータの動作を決めるもの　＝　命令
- 命令は一つのデータとして表現され，メモリに順に配置される.

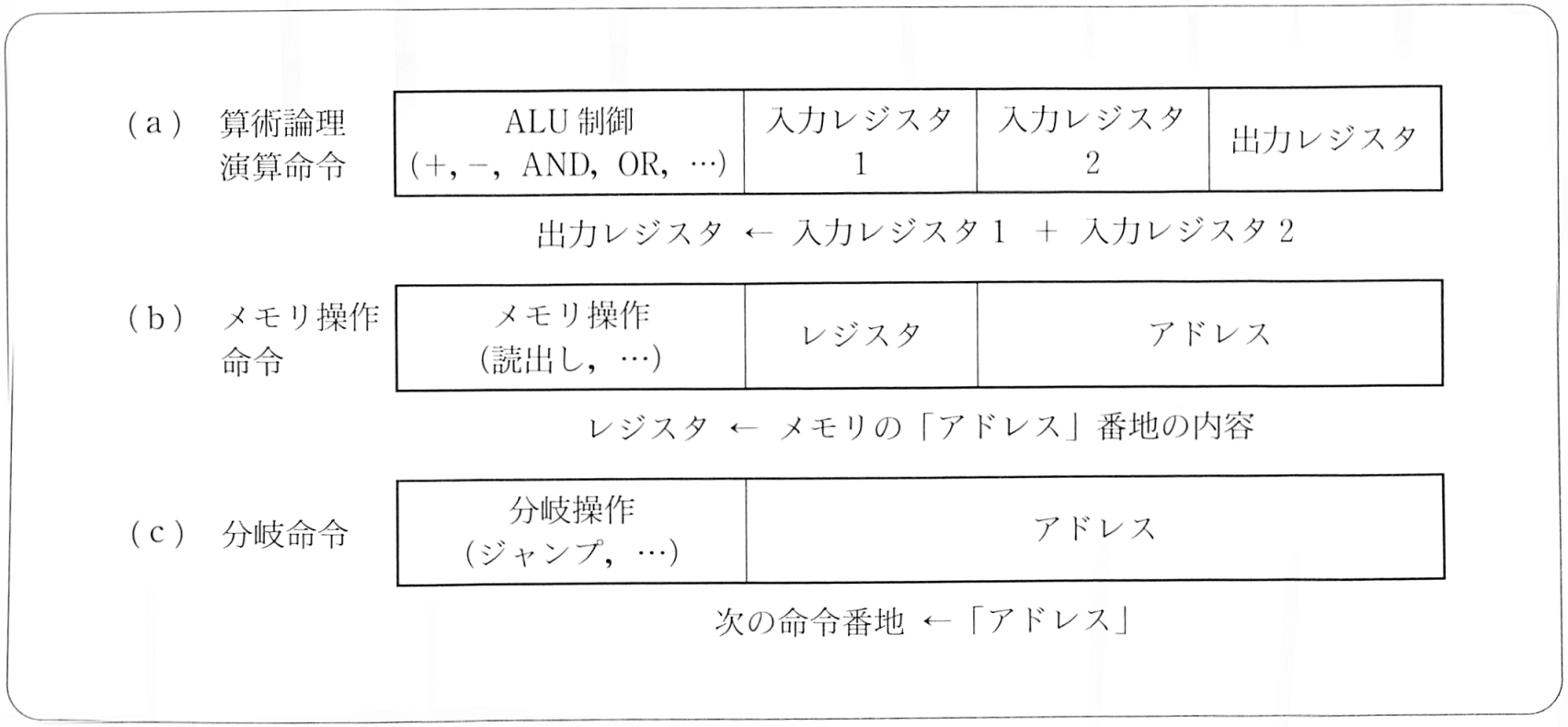


図 2.7　命令の種類と形式

2.2.2 命令実行の仕組み

1. 命令フェッチ
メモリ→命令レジスタ

2. 解釈（デコード）
命令レジスタ→命令デコーダ=>制御信号

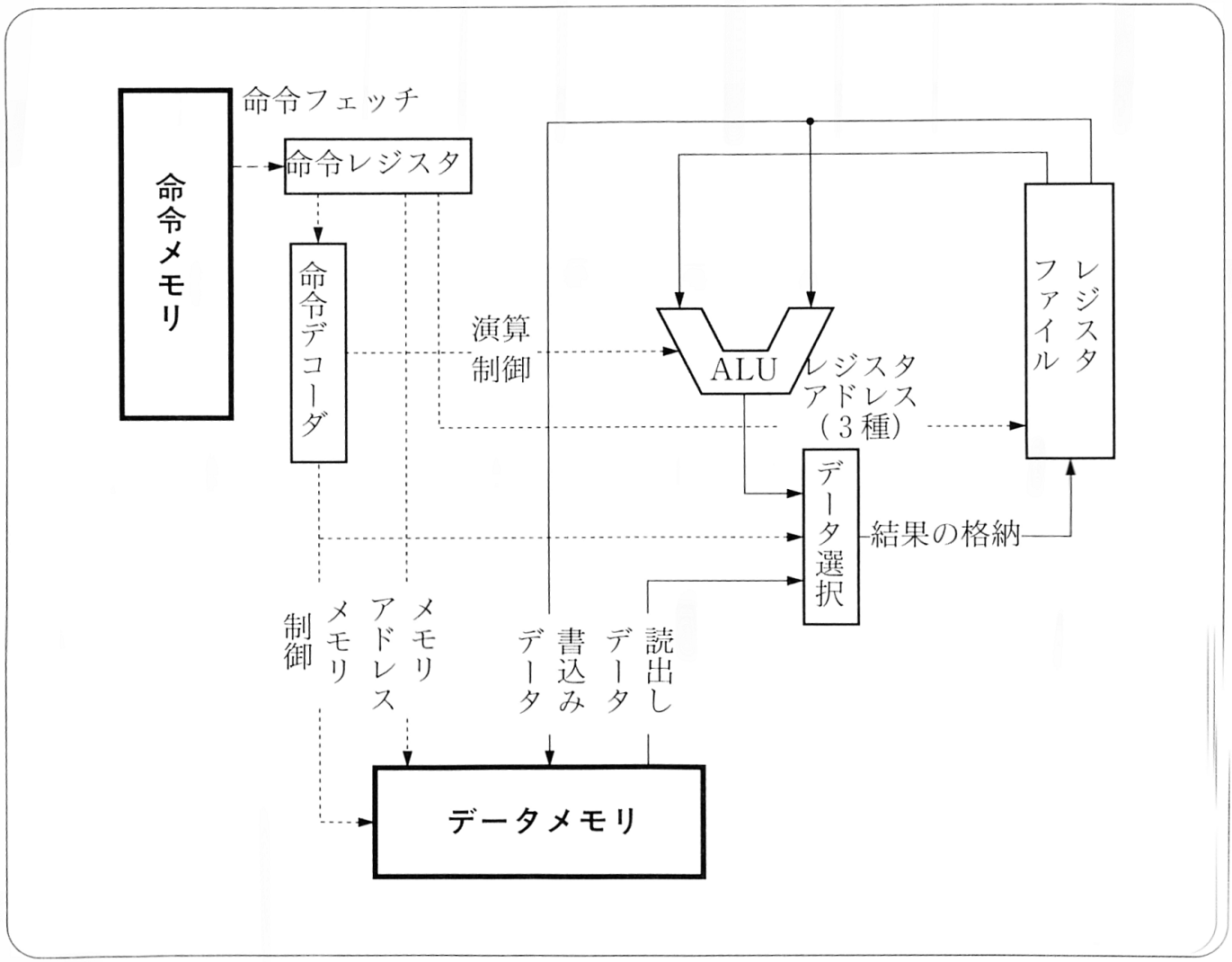


図 2.8　命令実行の基本形

2.2.3 算術論理演算命令の実行サイクル

実行の基本的サイクル

1. 命令フェッチ（図2.7(a)）
2. 命令デコード
3. 演算実行
4. 結果の格納

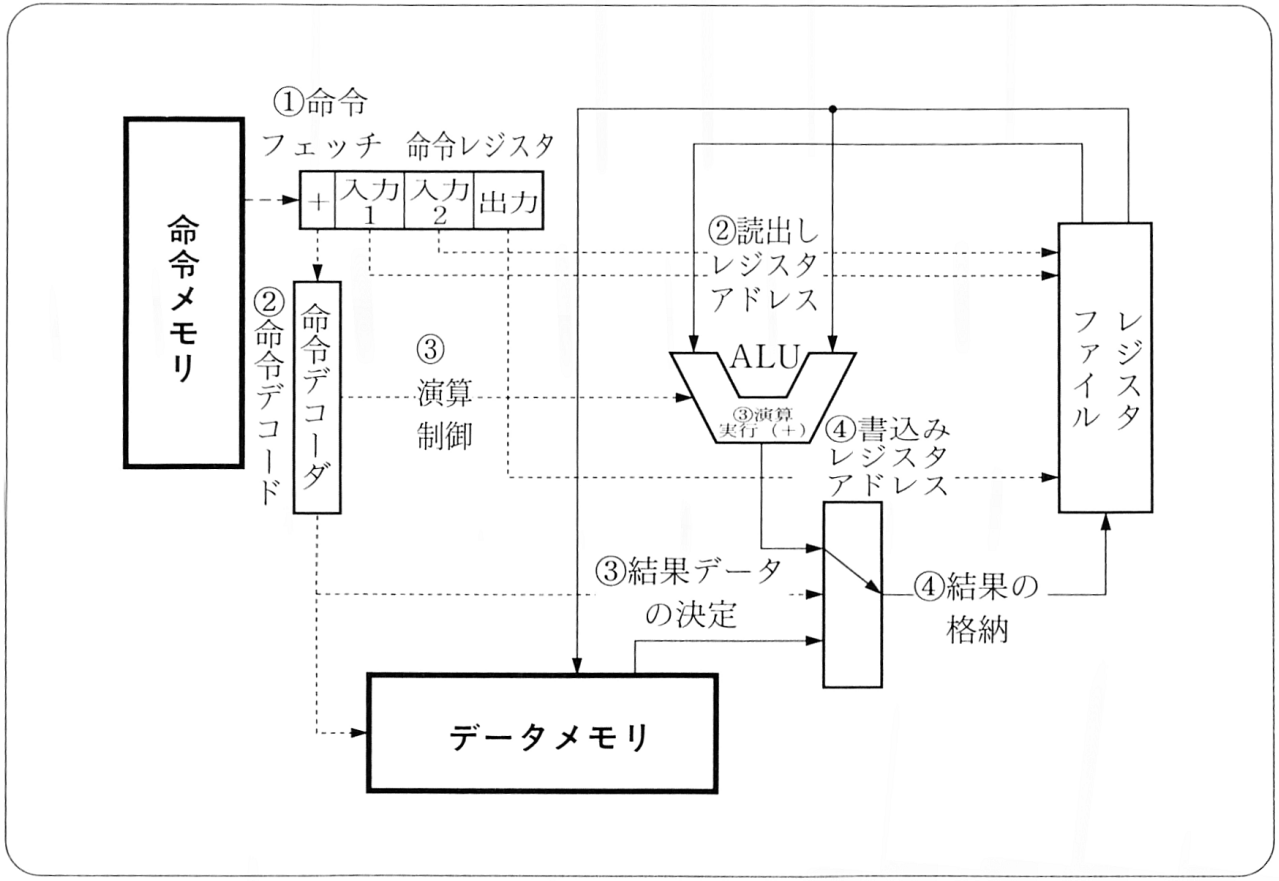


図 2.9　算術演算の実行

2.2.4 メモリ操作命令の実行サイクル

実行の基本的サイクル

1. 命令フェッチ（図2.7(b)）
2. 命令デコード
3. 演算実行
4. 結果の格納

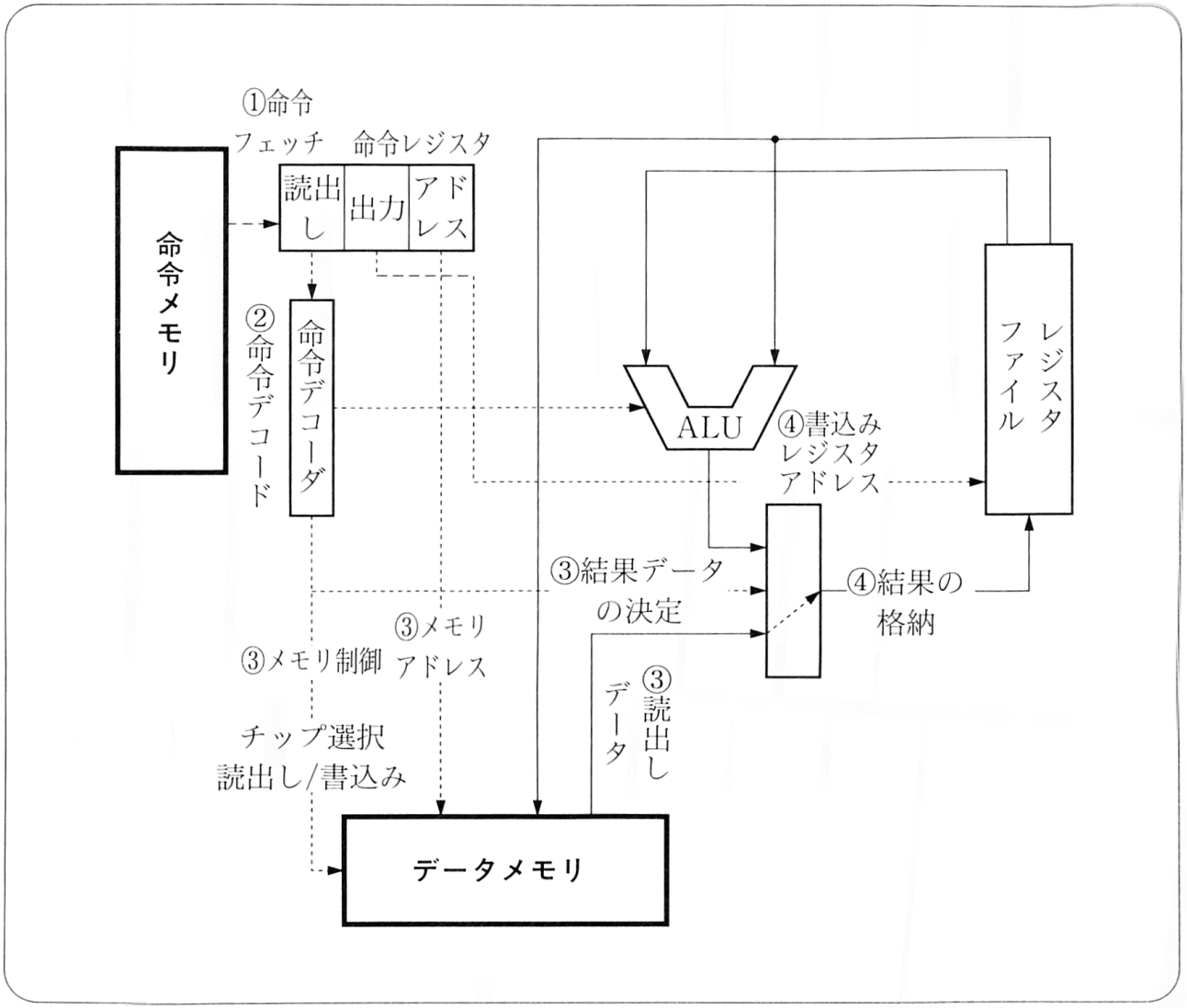


図 2.10　メモリ操作命令（読出し）の実行

2.3 シーケンサ

2.3.1 シーケンサとはなにか

- 次にどの命令を実行するのかを決定する機構
プログラム内蔵型コンピュータには必須の機構

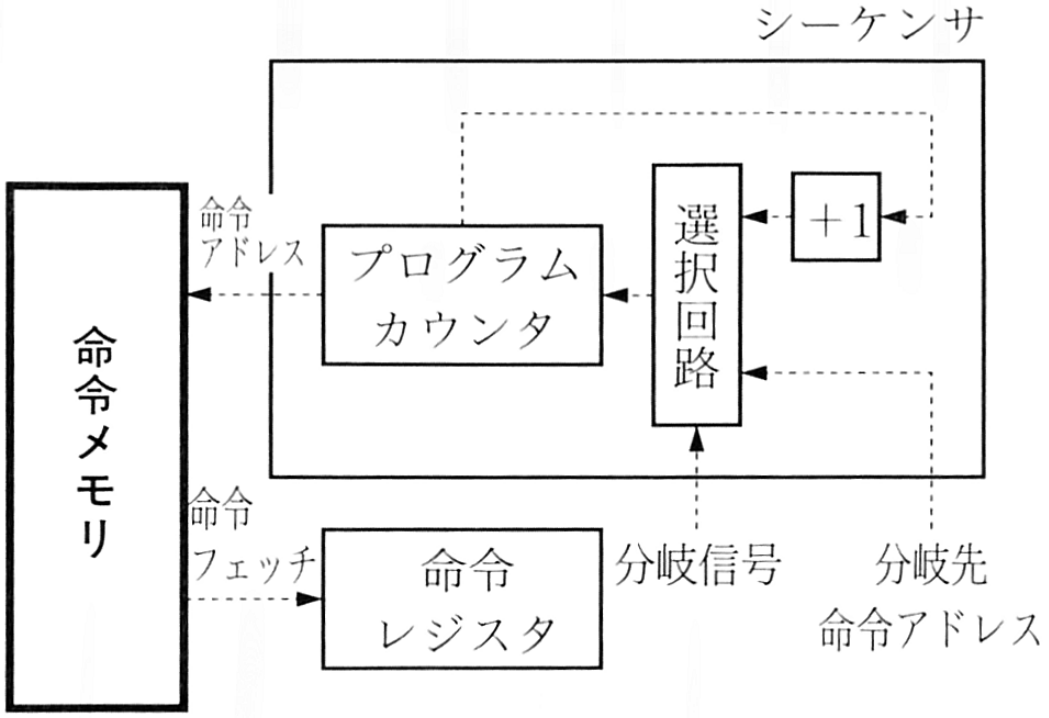


図 2.11　簡単なシーケンサ

シーケンサは，PCの値を決定する回路

- 通常の算術論理演算命令やメモリ操作命令の後は次の命令語を実行する．このため，＋1
- GOTOなどの無条件分岐命令の場合は分岐先命令アドレスを生成
- 条件分岐の場合は，直前の演算結果フラグから分岐信号を生成して選択回路に与える．

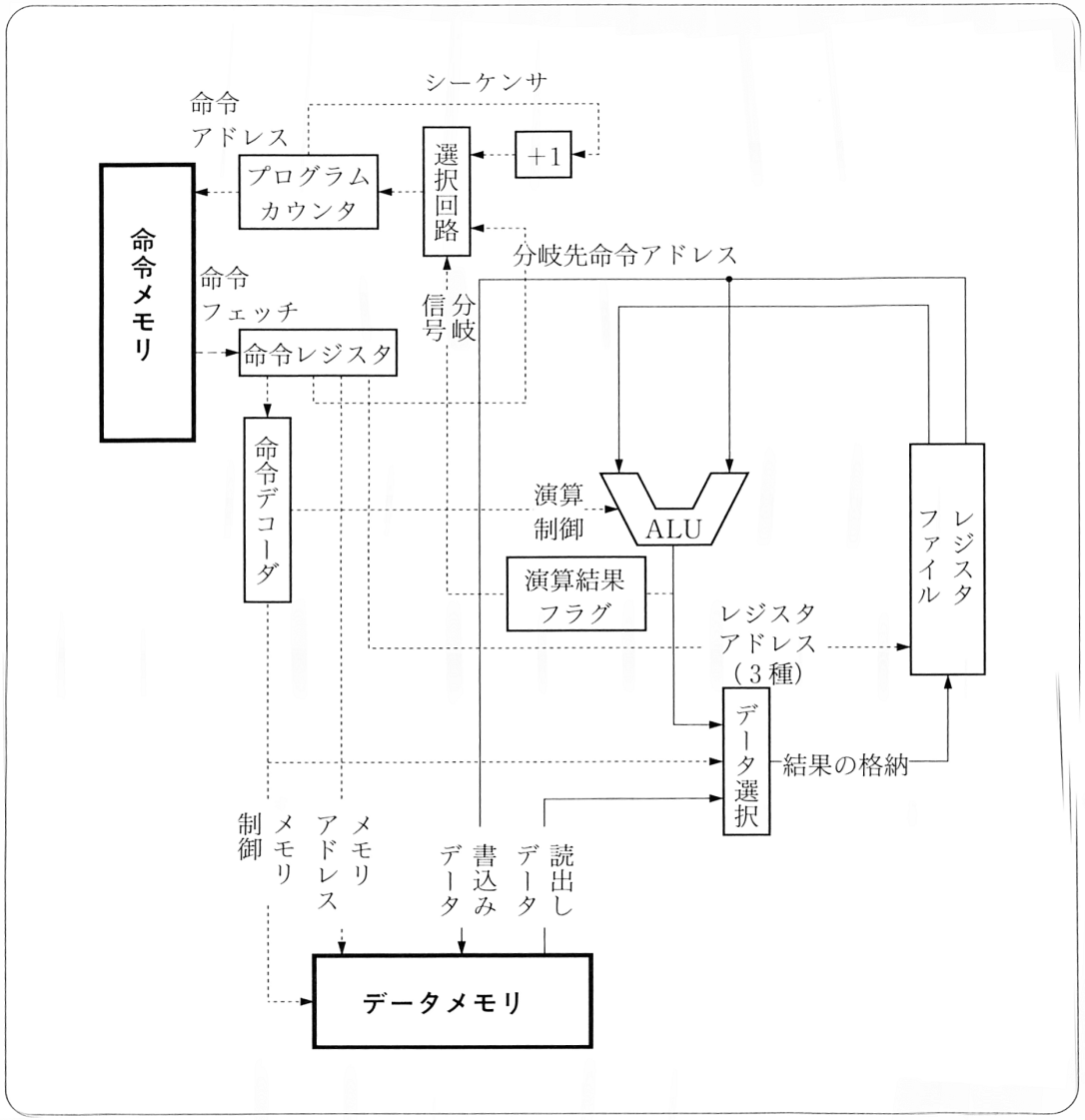


図 2.12　コンピュータ中枢部の構成

2.3.2 条件分岐命令の実行サイクル

- 命令フェッチ
  - 図2.7(c)
- 命令デコード
  - 条件分岐命令であることが判明
- 演算実行
  - 直前の命令実行結果から分岐信号を作る
- プログラムカウンタの値の設定
  - 選択回路で決定されたアドレスをPCに設定

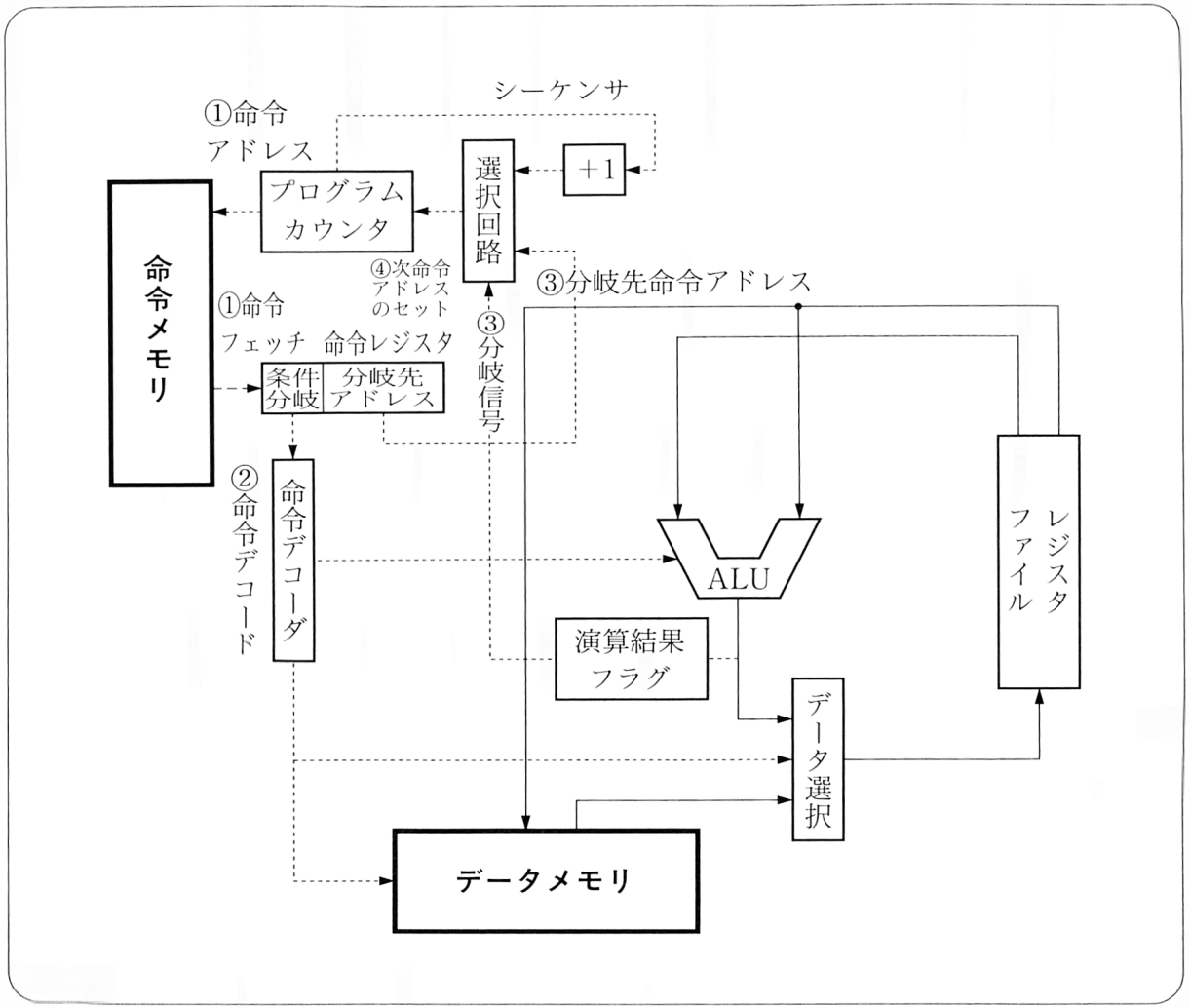


図 2.13　条件分岐命令の実行

本日の講義の範囲