

## 計算機アーキテクチャ レポート1

(2007 年 10 月 22 日課題)

提出様式: 回答はA4用紙で、様式は自由。下記の内容を1 ページ目の最初に書く。

**\*\* 計算機アーキテクチャレポート1 学籍番号 氏名 \*\***

提出方法: 課題を配布した講義の次週の講義時に提出。二重聴講など講義を欠席する場合は、情報事務室(G棟4階)に提出。

その他: 解答を後に配布するので、提出する前に解答をコピーして保管しておくこと。

[問1] (大阪大学大学院情報科学研究科入試問題から抜粋)

CPU(中央処理装置)は、演算処理部と実行制御部から構成される。図1は、CPUの基本構成の一例を示しており、プログラムカウンタ(PC)、命令レジスタ(IR)、メモリアドレスレジスタ(MAR)、メモリデータレジスタ(MDR)、汎用レジスタ(R1~Rn)、一時的に演算結果などを記憶するレジスタ(X、Y、Z)、算術論理演算器(ALU)が、バスを介して信号を転送している。なお、主記憶装置との信号転送は、アドレスバス、データバスを介して行われる。図中の●は制御ゲートを示しており、制御信号がオンとなった際にデータを送受できる。C は、桁上げ入力である。

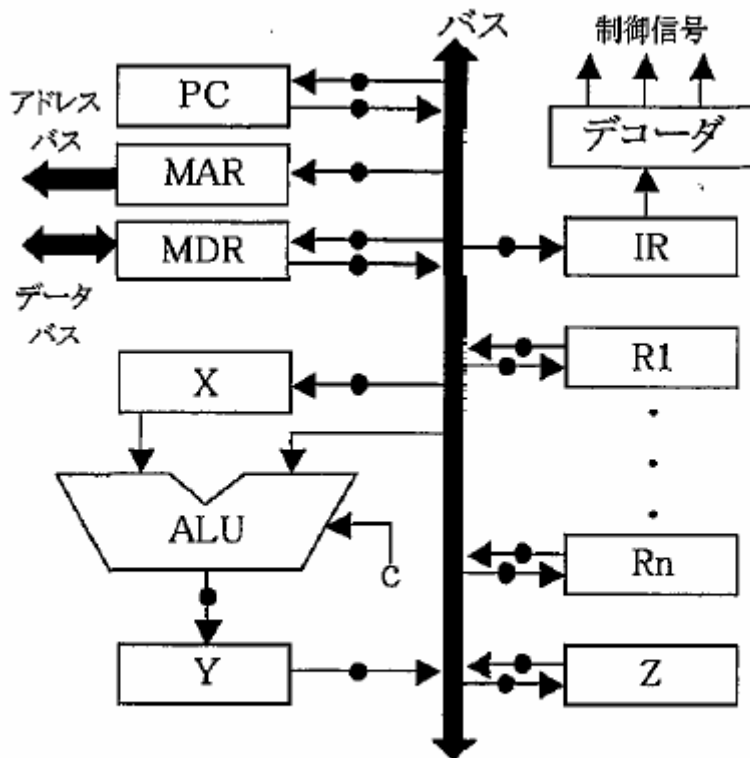


図1 CPUの基本構成

(1-1)

CPU が演算命令を実行する手順を説明する以下の文章として、文中空欄 (a) ～ (j) に当てはまる適当な語句を下記の選択肢から 1 つ選び、その番号を答えよ。同じものを複数回用いても良い。

命令フェッチ(命令読み出し)では、(a) の指示する主記憶アドレスから命令を取り出し、(b) に格納する。このことは、次のような複数のステップに分けて実行される。まず、(c) の内容を (d) に転送し、次に主記憶に対して読み出し指示を与える。ここで、主記憶からの読み出し動作が完了し、MDR へデータが格納されるまで、CPU は待機する。なお、この間に PC の内容を ALU により 1 だけインクリメントしている。MDR への格納完了を通知する信号を主記憶から受け取った後に、その内容を (e) に転送する。以下の各手順も、同様に複数のステップで実行される。

命令の解読では、IR に取り出された命令後の命令コード部を解読し、実行すべき動作を決定する。オペランドフェッチ(オペランド読み出し)では、取り出された命令語の (f) から、オペランドの格納されているレジスタ、あるいは主記憶内の位置を知り、そこからオペランドを取り出す。この際の処理は、(g) で若干異なり、(h) の場合は、命令語のアドレス部で指定された汎用レジスタの内容がオペランドになる。直接アドレス指定の場合は、アドレスそのものが命令語のアドレス部で指定される。(i) の場合は、命令語のアドレス部で指定された汎用レジスタの内容が、実効アドレスとなる。オペランドフェッチが終了すると、命令コード部で指定された演算を実行する。この後、演算結果を指定されたレジスタ、あるいは主記憶に格納する。主記憶に格納するためには、書き込むアドレスを (j) に格納した後、書き込むデータを MDR に格納し、書き込み命令を実行する。引き続き、命令フェッチに戻って、次の命令が実行される。

#### 選択肢

- ① PC ② MDR ③ アドレス部 ④レジスタ間接指定 ⑤ MAR
- ⑥ 即値指定 ⑦ インデックス指定 ⑧ IR ⑨ アドレス交換 ⑩ レジスタ
- ⑪ アドレス指定モード(アドレス指定形式) ⑫ ALU ⑬ レジスタ指定

(1-2)

命令を実行する手順の各ステップでの制御信号は、制御ゲートの信号のオン、オフや演算器などの動作を指定するビットパターンとらえることができる。このビットパターンを、各ステップでのハードウェアユニットの動作を規定する命令とみなすことで、制御回路を構築することができる。このようなビットパターンをマイクロ命令と呼び、これを用いて実行を制御する方式を、マイクロプログラム制御方式という。この際に、例えばレジスタへのデータ格納などの操作(マイクロ操作)ごとに各 1 ビットを割り当てると、各ステップの動作を記述するのに多数のビットが必要となり、しかもその大部分が 0 である。そこで、マイクロ命令の語長を以下のように短縮することを考える。図 1 に示す CPU の基本構成の汎用レジスタ数を 4 とする。例えば、各レジスタからバスへの出力を  $R_{out}$  で表すと、 $R_{out}$  としては、次のような 8 種類のマイクロ操作が含まれ、これらは同時に実行されることはない。

$$R_{out} = \{PC_{out}, MDR_{out}, R1_{out}, R2_{out}, R3_{out}, R4_{out}, Y_{out}, Z_{out}\}$$

このように同時に実行されることのないマイクロ操作をまとめて、区別できるようにすることで、マイクロ命令の語長を短くできる。図 1 に示した CPU の基本構成に対するマイクロ操作の例を、表 1 に示す。なお、表中の NONE は「操作なし」を示している。

表 1 マイクロ操作

$R_{out}$	$R_{in}$	ALU	R/W	Clear X	$Y_{in}$	c	WMS
NONE	NONE	NONE	NONE	NONE	NONE	c=0	NONE
$PC_{out}$	$PC_{in}$	Add	Read	Clear X	$Y_{in}$	c=1	WMS
$MDR_{out}$	$MAR_{in}$	Sub	Write				
$R1_{out}$	$MDR_{in}$	AND					
$R2_{out}$	$IR_{in}$	OR					
$R3_{out}$	$R1_{in}$	XOR					
$R4_{out}$	$R2_{in}$						
$Y_{out}$	$R3_{in}$						
$Z_{out}$	$R4_{in}$						
	$X_{in}$						
	$Z_{in}$						
レジスタから バスへのデータ出力	バスからレジスタへのデータ出力	ALU の演算	主記憶からの読み出し、 主記憶への書き込み	レジスタ X の内容をクリア	レジスタ Y の内容をクリア	桁上げ入力の値	主記憶からの読み出し動作完了の待機

この時、命令フェッチは、次の 3 ステップで示される。

	$R_{out}$	$R_{in}$	ALU	R/W	Clear X	$Y_{in}$	c	WMS
ステップ 1	$PC_{out}$	$MAR_{in}$	Add	Read	Clear X	$Y_{in}$	c=1	NONE
ステップ 2	$Y_{out}$	$PC_{in}$	NONE	NONE	NONE	NONE	c=0	WMS
ステップ 3	$MDR_{out}$	$IR_{in}$	NONE	NONE	NONE	NONE	c=0	NONE

同様にして、表 1 のマイクロ操作表記をもとに、以下の加算命令に関して、命令解読後の各ステップを示せ。

- (i) 加算命令 (レジスタ R1 の内容とレジスタ R2 の内容を加算し、レジスタ R2 に格納)
- (ii) 加算命令 (レジスタ R1 の内容で指定される主記憶アドレスに格納されている内容とレジスタ R2 の内容を加算し、レジスタ R2 に格納)

[問2] 2の補数系の加算演算でアンダーフローした場合(例えば4ビットの場合 $-7+(-6)$ )、オーバーフローした場合(例えば  $7+6$ )に、どのように検出できるか？

[問3]

(3-1) 2進数表現 11.11 を 10 進数で表現せよ。

10ビットの浮動小数点2進数表現  $X_{10}=(x_9, x_8, \dots, x_1, x_0)$  (ただし、 $x_9$  は符号ビット(1の時負)、 $(x_8, x_7, x_6, x_5)$ が指数部で指数のバイアスを7(7余り表現)、 $(x_4, x_3, x_2, x_1, x_0)$ が仮数部の絶対値で正規化され最上位ビットの1は省略されるものとする。(0,0,0,0,0,0,0,0,0,0)は0とする。)において下記の問いに答えよ。

例:10進数7. 5は、2進数表現で 111.1 となる  $1.111 \times 2^2$  であり、符号ビットは正であり  $x_9=0$ 、指数部はバイアスを足した  $9(=2+7)$  となり  $(x_8, x_7, x_6, x_5)=(1, 0, 0, 1)$ 、仮数部は最上位ビットの1は省略されるので  $(x_4, x_3, x_2, x_1, x_0)=(1, 1, 1, 0, 0)$  となる。

(3-2) 表現できる正の数の最大値はいくらかを示せ、又その時のビット表現を示せ。

(3-3) 表現できる負の数の最大値はいくらかを示せ、又その時のビット表現を示せ。

(3-4) 指数部が(1,0,0,1)の時の ulp(units in the last place:表現できる数の間隔)はいくらか。また絶対誤差(元の実数値と表現された数の差の最大値)はいくらか

(3-5) この表現のマシンエプシロンはいくらか？