

1. 以下の問に答えよ。(3×10=30 点)

(a) 次の 10 進数を 4 bit の 2 進数で表せ。ただし, 負数は 2 の補数で表すことにする。

$$(1) 3 = (0011)_2$$

$$(2) -3 = (1101)_2$$

$$(3) -6 = (1010)_2$$

(b) 実数  $x$  を

$$x = (-1)^s \cdot 2^E \cdot (1.f_1 f_2 f_3 \dots)_2,$$

$$s = 0, 1, \quad E = \text{整数}, \quad f_i = 0, 1 (i = 1, 2, \dots)$$

という形の浮動小数点表示で表わしたとき, 以下の数について, それぞれ  $s, E, f_1, f_2, \dots$  を求めよ。

$$(1) x = -1.25$$

$$(2) x = 7.0$$

$$(1) s = 1, E = 0, f_1 = 0, f_2 = 1, f_3 = f_4 = \dots = 0$$

$$(2) s = 0, E = 2, f_1 = f_2 = 1, f_3 = f_4 = \dots = 0$$

(c) 次の単語を英語で表せ。

(1) キャッシュメモリ: cache memory (2) レジスタ: register

(3) デコーダ: decoder (4) フェッチ: fetch (5) ライトスルー: write through

解説およびコメント

$-x$  の表示は  $2^4 - x$  の表示になる。(a), (b) は大抵の人ができていた。(c) でライトスルーを “right through” と書いた人があまりにも多かった。“light through” も少しいた。

2. 以下の問に答えよ。(2×10=20 点)

(a) 論理式  $AB + A\bar{B} + \bar{A}B$  を簡単化せよ

$$AB + A\bar{B} + \bar{A}B = A(B + \bar{B}) + \bar{A}B = A + \bar{A}B = A + B$$

(b) 論理式  $A \cdot B + C \cdot (A \oplus B) = A \cdot B + C \cdot (A + B)$  証明せよ。

下の解説参照

解説およびコメント

(a) は解説不要であろう。(b) の証明は式の変形でも可能だが, 以下のように考えれば簡単である:

$A + B$  と  $A \oplus B$  が同じ値を持つときは, 当然, 左辺 = 右辺 になるので, これら 2 つが同じ値にならない場合を考えればよい。それはどういう場合

かと言えば  $A = B = 1$  となる場合である。しかし、このときは  $C$  の値にかかわらず両辺は 1 になるので、上の等式はやはり常に成り立つことになる。

3. 次の問に答えよ。(6 × 5 = 30 点)

(a) 命令パイプラインがスムーズに流れないことがある。そのような例を 2 つあげて説明せよ。

(1) あるステージを実行中の命令 A と別なステージを実行中の命令 B が同じハードウェア資源を利用するとき (構造ハザード), (2) 命令 A の結果を直後の命令 B が用いるとき (データハザード)

(b) キャッシュミスが起こるのはどのような場合か。3 つの例を示せ。

(1) 最初にアクセスするときに起こるミス (初期参照ミス), (2) 同じインデックスを持つ異なるアドレスに参照したとき (競合性ミス), (3) キャッシュに置いておきたいデータの容量がキャッシュの容量を越えたとき (容量性ミス)

(c) プログラム中、ロード / ストア命令の占める割合が 20%、キャッシュのミス率が 5%、ミスペナルティが 10 のとき実行時間相対値を求めよ。

$$1 + 0.2 \times 0.05 \times 10 = 1.1$$

(d) キャッシュメモリはデータ用と命令用に分けている。その理由を述べよ。

パイプライン動作時に命令のフェッチとデータのロード / ストアが競合するのを避けるために分けている。

(e) 仮想記憶は何のために考え出された機構か。

主記憶容量があたかも命令語で定義されるアドレス空間だけあるように見せかける機能である。物理的には、主記憶に入りきれなかったデータは、二次記憶 (ハードディスク) に追い出される。こうすることによって、いくつものプロセスを並行して走らせることができる。

(f) 数式  $(a + b * c) * d$  を逆ポーランド記法で表すとどのようになるか。

$abc * +d*$  (日本語と同じ語順)

4. 表 1 (別紙) に示す仕様のアセンブリ言語のプログラムに関して、以下の設問に答えよ。ただし、ここで

- 命令語長は 32 ビット = 4 バイト
- r0 には常に 0 が設定されている

とする。(2 × 10 = 20 点)

#### 4.1 次のプログラム

アドレス：	命令
0100：	blt r2, r3, 8
0104：	addi r1, r3, 0
0108：	beq r0, r0, 4
010C：	addi r1, r2, 0
0110：	(プログラム終了)

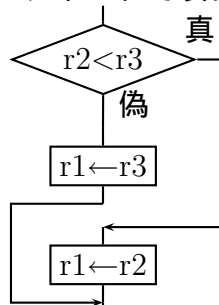
をアドレス 0100 から実行する場合について、以下の問いに答えよ。

- (a)  $r2=5, r3=4$  の場合、プログラム終了後に  $r1$  はいくつになるか。
- (b)  $r2=2, r3=6$  の場合、プログラム終了後に  $r1$  はいくつになるか。

答え アドレス 0100 の命令は条件付きジャンプであり、条件式  $r2 < r3$  が真であればアドレス  $0100+4+8=010C$  へジャンプする。

また、アドレス 0108 の命令は条件付きジャンプだが条件式  $r0=r0$  は常に真であるため、これはアドレス  $0108+4+4=0110$  への無条件ジャンプと同等である。

よってこのプログラムをフローチャートで表わすと



となる。このフローチャートからわかるとおり、このプログラムは  $r2$  と  $r3$  のうち小さい方を  $r1$  にセットする。

よって、プログラム終了時には

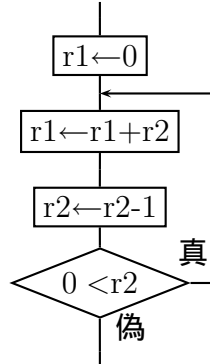
- (a)  $r2=5, r3=4$  の場合は、 $r1=4$  となる。
- (b)  $r2=2, r3=6$  の場合は、 $r1=2$  となる。

## 4.2 次のプログラム

アドレス： 命令  
0100： addi r1, r0, 0  
0104： add r1, r1, r2  
0108： subi r2, r2, 1  
010C： blt r0, r2, -12  
0110： (プログラム終了)

をアドレス 0100 から実行する場合について， $r_2$  の初期値が自然数  $n$  の場合，プログラム終了後に  $r_1$  はいくつになるか。

答え アドレス 010C の条件付きジャンプ命令では， $0 < r_2$  ならばアドレス 010C+4-12=0104 へジャンプするので，プログラムをフローチャートで表わすと



となる。 $r_2$  は初期値が自然数  $n$  なので，上のフローチャートは  $r_2$  が  $n, n - 1, \dots, 2, 1$  と変化する繰り返し処理であることがわかる。

また  $r_1$  については，初期値を 0 として繰り返し処理中に  $r_2$  の値を足し合せていくので，プログラム終了時には

$$r_1 = n + (n - 1) + \dots + 2 + 1 = \sum_{k=1}^n k = \frac{n(n + 1)}{2}$$

となる。