

専門科目（午後）

15 大修

情報工学・通信工学

時間 13:30 ~ 16:00

注意事項

1. 次の6題の中から3題を選択して解答せよ。4題以上解答した場合はすべて無効とする。
  2. 解答は1題ごとに別々の解答用紙に記入せよ。
  3. 各解答用紙に問題番号及び受験番号を記入せよ。
  4. 電子式卓上計算機等の使用は認めない。
-

1. デジタル信号処理回路は、図 1.1 に示す加算器、乗算器、および遅延器を用いて構成される。以下では  $n$  は整数とし、信号  $x(n)$  は  $x(n) = 0 (n < 0)$  を満たす因果的な信号とする。

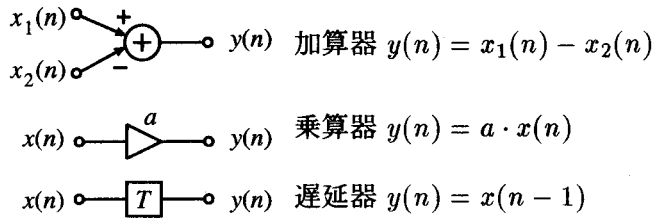


図 1.1: デジタル信号処理回路の構成要素

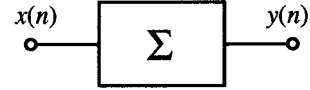


図 1.2: デジタル積分器

- 1) 図 1.2 はデジタル積分器と呼ばれる回路で、その機能は、任意の入力信号  $x(n)$  に対し

$$y(n) = \sum_{k=0}^n x(k) \quad (1.1)$$

を出力するものである。デジタル積分器は、図 1.1 の加算器 1 つ、遅延器 1 つを用いて構成できる。実際に、加算器 1 つ、遅延器 1 つを用いて図 1.2 の回路を完成せよ。

- 2) 1) で求めた図 1.2 の回路に対応する  $x(n)$  と  $y(n)$  の関係を差分方程式として表せ。また、得られた差分方程式に  $z$  変換を適用し、 $x(n)$ 、 $y(n)$  の  $z$  変換それぞれ  $X(z)$ 、 $Y(z)$  の関係を求めよ。ただし、信号  $w(n)$  の  $z$  変換  $W(z)$  を

$$W(z) = \sum_{n=0}^{\infty} w(n)z^{-n} \quad (1.2)$$

とする。

- 3) 図 1.2 のデジタル積分器を用いた図 1.3 の回路を考える。この回路の伝達関数  $H(z) = \frac{Y(z)}{X(z)}$  を求めよ。

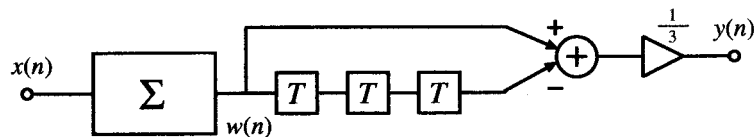


図 1.3: デジタル積分器を用いた回路

- 4) 図 1.3 と同一の伝達関数を持つ非巡回型の回路（帰還ループをもたない回路）を導き、図 1.1 の回路記号を用いて表せ。
- 5) 図 1.3 は入力信号  $x(n)$  に対してどのような動作をする回路か、簡潔に説明せよ。
- 6) 図 1.3 の回路の振幅応答を角周波数  $\omega$  [rad/sample] の関数として求めよ。
- 7) 図 1.3 および 4) で得られた回路の伝達関数は同一であるが、回路実現の観点からは両回路には大きな違いが生じる。両回路をそれぞれ 16 ビット固定小数点表示のデジタル信号処理プロセッサを用いて実現したとしよう。このとき、特に  $x(n)$  の平均値

$$m = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} x(k) \quad (1.3)$$

に着目し、両者に生じる違いについて論ぜよ。

2. 時刻  $t = iT$  ( $i$  は整数,  $T$  はシンボル長) において送信信号  $a(i)$  が送信される。ここで  $a(i)$  は確率変数で 1 か  $-1$  の 2 値しか取りえないものとし, 受信信号  $y(i)$  は

$$y(i) = ha(i) + n(i)$$

と表せるものとする。なお,  $h$  は既知の正の定数で,  $n(i)$  は平均値 0, 分散  $\sigma^2$  の正規分布に従う確率変数とする。即ち,  $n(i)$  の確率密度関数  $p[n(i)]$  は

$$p[n(i)] = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{\{n(i)\}^2}{2\sigma^2}\right]$$

となる。

受信信号  $y(i)$  の情報を基に  $a(i)$  を判定する問題を考える。なお,  $a(i)$  の判定は以下の判定規則に従うものとし,  $a(i)$  の判定値は  $\hat{a}(i)$  で表すことにする。

- 判定規則

以下の条件

$$p[y(i) - h] > p[y(i) + h] \quad (2.1)$$

を満足するとき,  $\hat{a}(i) = 1$  とする。この条件を満足しないとき, 即ち  $p[y(i) - h] \leq p[y(i) + h]$  のとき,  $\hat{a}(i) = -1$  とする。

以下の問いに答えよ。

1) 式 (2.1) は  $y(i) > 0$  と等価であることを証明せよ。

2)  $a(i) = 1$  のときに  $\hat{a}(i) = -1$  となる条件付き確率を  $P_{e1}$ ,  $a(i) = -1$  のときに  $\hat{a}(i) = 1$  となる条件付き確率を  $P_{e2}$  とする。 $P_{e1}$  と  $P_{e2}$  を以下で定める誤差補関数  $\operatorname{erfc}(x)$  を用いて表せ。

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} \exp(-t^2) dt$$

3)  $a(i) = 1$  となる確率と  $a(i) = -1$  となる確率がともに  $1/2$  となるとき, 判定誤りを起こす確率, 即ち  $\hat{a}(i) \neq a(i)$  となる確率  $P_e$  を  $\operatorname{erfc}(x)$  を用いて表せ。また,  $a(i) = 1$  となる確率が  $p_a$  ( $0 \leq p_a \leq 1$ ) で,  $a(i) = -1$  となる確率が  $1 - p_a$  となる場合,  $\hat{a}(i) \neq a(i)$  となる確率  $P'_e$  を  $\operatorname{erfc}(x)$  を用いて表せ。

4)  $a(i) = 1$  となる確率と  $a(i) = -1$  となる確率がともに  $1/2$  で, かつ  $a(i)$  と  $a(k)$  は  $i \neq k$  のとき互いに独立な確率変数とする。同様に,  $n(i)$  と  $n(k)$  も  $i \neq k$  のとき互いに独立な確率変数とする。 $N$  ( $N \geq 1$ ) 個の送信信号  $\{a(i) | 1 \leq i \leq N\}$  を  $N$  個の受信信号  $\{y(i) | 1 \leq i \leq N\}$  から判定するとき,  $N$  個の内,  $l$  ( $0 \leq l \leq N$ ) 個以上判定が誤る確率を  $P_e(l)$  とする。この  $P_e(l)$  を上記の  $P_e$  を用いて表せ。

3. 図 3.1 のような 1 ビットの ALU を考える。この ALU は、2 つの 1 ビット入力 A, B に対し、AND 演算、OR 演算、加算、もしくは減算(A-B)の結果を X に出力する。また、キャリー入力  $C_i$  とキャリー出力  $C_o$  および演算モードを示すための入力  $M_1$  と  $M_2$  を持ち、 $\{M_1, M_2\}$  が  $\{1, 1\}$  の時 AND 演算、 $\{1, 0\}$  の時 OR 演算、 $\{0, 0\}$  の時加算、 $\{0, 1\}$  の時減算を行う。また、負数は 2 の補数を用い表現し、 $C_i$  と  $C_o$  は加減算とも正論理（つまり、減算の時も最下位ビットの  $C_i$  は 0 が入力される）で、AND、OR 演算の時には Don't Care とする。

1) 上記 ALU を PLA (Programmable Logic Array) で実現する。ただし、PLA は、 $n_i$  本の入力線、 $n_m$  本の内部線、 $n_o$  本の出力線を持ち、入力の肯定および否定線と内部線の間を AND アレイ、内部線と出力線の間を OR アレイとする。例えば、図 3.2 の  $n_i=2$ 、 $n_m=3$ 、 $n_o=2$  の PLA は、 $R=PQ$  および  $S=\overline{P}Q+\overline{Q}$  を実現している。ALU には、 $n_i=5$  (A,B, $C_i$ , $M_1$ , $M_2$ )、 $n_m=12$ 、 $n_o=2$  (X, $C_o$ ) の PLA を使う。

a) まず、入力と出力の間の論理式を表現するため、行側(横向き)のグレイコードの並びを  $C_iBA$ 、列側(縦向き)のグレイコードの並びを  $M_1M_2$  として、出力 X と  $C_o$  に対するカルノー図を描け。

b) 上記カルノー図から上記 1 ビット ALU の各出力を表す簡便化した加法標準形の論理式を求めよ。

c) 入力を左から A, B,  $C_i$ ,  $M_1$ ,  $M_2$ 、出力を左から X,  $C_o$  とした PLA の内部の結合状態を図示せよ。

2) 上記 1 ビット ALU を 4 個(ALU<sub>0</sub>~ALU<sub>3</sub>)と、4 ビット長のレジスタ Ra(図 3.3)、シリアル入出力とパラレル出力を持つ 4 ビットシフトレジスタ Rb(図 3.4)、シリアル入出力、パラレル出力と入力  $I_d$  が 1 のときシフト前に値をパラレルにロードする機能を持つ 4 ビットシフトレジスタ Rc(図 3.5)、1 ビットシフトレジスタ Rd、および制御回路 CNT を使い、4 ビット正負数の被乗数と乗数間の乗算結果(8 ビット)を計算する回路 M を構成する。ただし、シフトレジスタはクロックに同期して値を 1 ビット左にシフトし、初期状態では Ra に被乗数、Rb に乗数を格納し、Rc の初期状態は 0、また計算結果の下位 4 ビットは Rb に、上位ビットは Rc に入る。乗算のアルゴリズムは、ブース(Booth)のアルゴリズムを用いる。ブースのアルゴリズムでは、乗数の中で連続する 1 があったときに、連続する 1 の中で最下位の桁を減算に置き換え、最上位の次の桁で加算を行う。つまり、その時点の乗数の連続する下位 2 ビットを観測し、01 であれば被乗数を中間積の上位に加えた新中間積と乗数をシフトし、10 であれば被乗数を中間積の上位から引いた新中間積と乗数をシフトする。00 と 11 の場合には中間積と乗数を単にシフトする。Rd は最初のステップでの連続する 2 ビットを比較するための仮のビットで初期状態では 0 である。

a) 以上を基に回路図 M を作成せよ。

b) 上記の制御回路 CNT を PLA で構成する場合の入出力線と内部の結合状態を図示せよ。

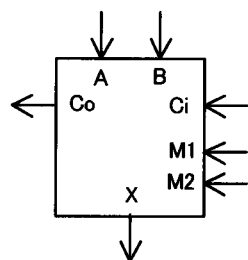


図3.1 ALU

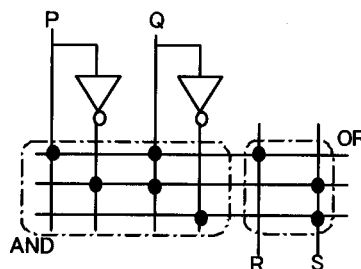


図3.2 PLA

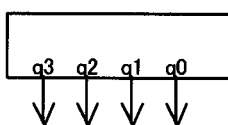


図3.3 Ra

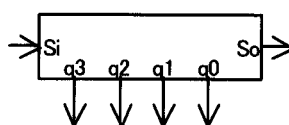


図3.4 Rb

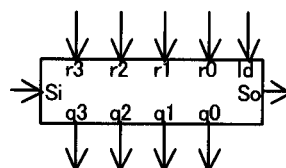


図3.5 Rc

4. オペレーティングシステムにおけるページ置換アルゴリズムに関する以下の問に答えよ。

1) FIFO アルゴリズムとは何か, 説明せよ。

2) LRU アルゴリズムとは何か, 説明せよ。

3) ページ参照列

1 3 2 5 1 3 4 1 3 2 5 4

について考える。また, 初期状態では主メモリ上の全てのページフレームは空であるとする。

a) 主メモリ上のページフレーム数が, 2, 3, 4, 5の場合のそれぞれについて, FIFO アルゴリズムではページフォールトは何回発生するかを答えよ。

b) 主メモリ上のページフレーム数が, 2, 3, 4, 5の場合のそれぞれについて, LRUアルゴリズムではページフォールトは何回発生するかを答えよ。

4) スタックアルゴリズムとは何か説明せよ。

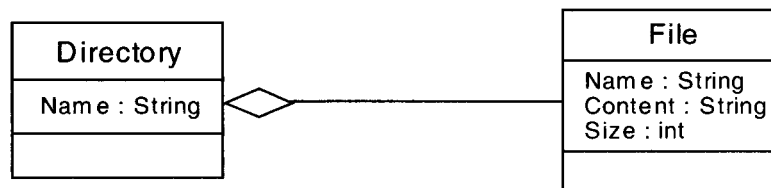
5) 以下のページ置換アルゴリズムはスタックアルゴリズムか。理由もつけて答えよ。

a) FIFOアルゴリズム

b) LRUアルゴリズム

5. オブジェクト指向設計, オブジェクト指向プログラミングについて以下の間に答えよ。

- 1) 「オブジェクト指向」の考え方の中で, 特徴的なことを3つあげ, 説明せよ。
- 2) UNIXのようなファイルシステムの階層構造を, オブジェクト指向設計法を用いて, 以下のようにクラス図で表現した。図中で **Name** は, **Directory** や **File** の名前を, **Content** は **File** の中身, **Size** は **File** の大きさを表している属性である。



このとき, **Directory** オブジェクトに属している **File** オブジェクトすべての大きさの合計値を計算するメソッド `getTotalSize()` の Java プログラムを以下のように書いた。空欄(a), (b), (c), (d)を埋めてプログラムを完成せよ。なお, ここでは簡単のため, 1つの **Directory** オブジェクトに直下に属している **File** オブジェクトの数は最大100としている。

```

public class (a) {
    ...
    (b) has[] = new (b)[100];
    int getTotalSize() {
        int totalsize = (c) ;
        for (int i=0; i<has.length(); i++) {
            (d)
        }
        return totalsize;
    }
}
  
```

- 3) 前問 2) のクラス図は, ファイルシステムの階層構造を表現する上で大きな問題点を含んでいる。それを指摘し, 修正したクラス図を書け。
- 4) 前問 2) で出てきたプログラムを参考にし, **Directory** または **File** の大きさを返すメソッド `calcTotalSize()` を, 修正したクラス図に従って書け。ただし, ここでは, **Directory** の大きさとは, その **Directory** に属するすべての **File** と **Directory** の大きさの和として再帰的に定義する。属する **File** や **Directory** を何も持たない **Directory** の大きさは0と定義する。

6. グラフ中の2節点の経路を求める以下のような Prolog プログラムを作成した。述語  $p(N1, N2, D)$  は節点  $N1$  と  $N2$  が長さ  $D$  の辺で接続されていることを表わす。述語  $go(S, G, P)$  は節点  $S$  から  $G$  までの経路を  $P$  に単一化する。  $findall(Var, Goal, Bag)$  は  $Goal$  を満足する解をバックトラックによってすべて求め、それぞれの解で得られる  $Var$  の具体化された値を解の得られた順序にリストの末尾に追加し、このリストを  $Bag$  に単一化する述語である。ただし、 $Goal$  を満足する解がない場合、 $findall$  は  $Bag$  に空リストを単一化し、成功する。  $member(Elem, List)$  は  $Elem$  がリスト  $List$  中のいずれかの要素に単一化できれば成功する述語である。述語  $not$  は、ゴール “?- call(P).” の呼び出しが失敗するときにゴール “?- not(P).” の呼び出しが成功するような述語である。このプログラムについて以下の間に答えよ。

```

p(a,b,5).
p(a,d,1).
p(b,c,3).
p(b,e,2).
p(c,e,1).
p(d,e,5).
pp(X,Y,D,T):-p(X,Y,D);p(Y,X,D),not(member(Y,T)).
go(S,G,P):-go(S,G,[],P).
go(G,G,_,[G]).
go(S,G,H,[S|P]):-findall(D-I,pp(S,I,D,H),Q),Q=Q1,go_loop(Q1,G,[S|H],P).
go_loop([_S|_],G,H,P):-go(S,G,H,P).
go_loop([_|T],G,H,P):-go_loop(T,G,H,P).

```

- 1) このプログラムで定義されているグラフを辺の長さも含めて図示せよ。
- 2) ゴール “?- go(a,e,P)” を呼び出し、バックトラックによってすべての解を求める。変数  $P$  に束縛される値を順序も含めてすべて答えよ。
- 3) 述語  $go$  で用いている探索戦略を何というか。
- 4) 節点間の距離を考慮して  $go/4$  の2番目の節の述語呼出し “ $Q=Q1$ ” を “ $sort(Q,Q1)$ ” に変更した。ただし、 $sort/2$  は要素として構造体  $N-X$  を持つリスト  $Q$  を  $N$  の昇順に並べかえたリストを  $Q1$  に単一化する述語である。たとえば、 “?- sort([3-c,1-a,2-b],Q1).” のゴール呼び出しの結果、 $Q1$  には [1-a,2-b,3-c] が束縛される。
  - a) 変更した  $go$  はどのような戦略で探索をおこなうか説明せよ。
  - b) 変更した  $go$  を使ってバックトラックによって “?- go(a,e,P)” の解をすべて求める。変数  $P$  に束縛される値を順序も含めてすべて答えよ。
- 5) 4) の探索戦略は必ず最短経路を最初の解として出力するかどうか答えよ。もし、必ずしも最短経路を最初の解として出力しないなら、そのような始点と終点の例をひとつあげ、それに対するすべての解が出力される順序とそれぞれの経路長を答えよ。