大阪大学大学院情報科学研究科
平成 31 年度　博士前期課程　入試問題
A 情報工学　解答・解説

楠本研究室：Kim Taeyoung，富田裕也，中川将，華山魁生

2018 年 9 月 2 日

1　アルゴリズムとプログラミング

解答

(1) (A)：`a[j] > a[j+1]`

(2) 配列 `win` および配列 `grade`：配列 `win` の要素の値によって昇順に並べられる．また，配列 `grade` は配列 `win` と対応するように要素が移動する．

平均時間計算量：$O(n^2)$

その理由：`functionA` には 2 つの `for` ループが存在する．これらのループは 2 重ループとなっており，6 行目から始まる `for` ループは `n-1` 回，7 行目から始まる `for` ループも `n-1` 回の処理を行う．`for` ループ内ではオーダー表記にして $O(1)$ の処理を行うため，`functionA` 全体では $O(n^2)$ となる．

(3) 配列 `win` 内に変数 `lot` と同じ値を持つ要素があるかを，探索範囲を `n/2` → `n/4` →... と半減させながら探索範囲が 1 になるまで探索を行っている．

`functionB` の戻り値は，配列 `win` 内に変数 `lot` と同じ値を持つものがあった場合はそのインデックス，なかった場合は`-1` である．

(4)

(4-1) (iii)

(4-2) $O(n \log n)$

(5)

(ア)：`(grade, win, 0, n-1)`

(イ)：`k=functionB(win, lot, 1))!=-1`

解説

(1)

問題に記載されている表を見ると，`i=1` の時と `i=3` の時で，配列 `b` は不規則な動きをしているのに対し，配列 `a` では常に値の小さいものが前の方に移動されていることがわかる．これより，8 行目から始まる `if` 文では，条件式を `a[j] > a[j+1]` とする．なお，くじ番号はいずれもたかだか一つしか存在しないため，等号は必要ない．

(2)

(1) での考察の通り，`functionA` では配列 `a` の値を昇順に並べる処理を行っていると推測できる．また，`win.txt` のデータでは当選番号とその等級が同じ行に並べられており，それらは 1:1 に対応している．28 行目から始まる `main` 関数での処理にて，`win.txt` の 1 列目，すなわち当選番号は配列 `win` に格納され，2 列目，すなわちその等級は配列 `grade` に格納される．つまり，当選番号 `win[i]` に対応する等級は，`grade[i]` である，ということになる．`functionA` では配列 `win` の要素の値を判断し，必要であれば前後を入れ替えることを繰り返し，昇順に並べる．その時，同時に対応する配列 `grade` も並べられている．その処理回数は，$(n-1) \times (n-1) = n^2 - 2n + 1$ 回であり，オーダー表記にすると $O(n^2)$ となる．なお，このソート方法をバブルソートという．

(3)

functionB には，functionA で処理されたデータ，すなわち配列 win の値によって昇順にソートされたデータが渡される．functionB では，このことを利用して配列 win 内に変数 lot と同じ値を持つ要素があるかを探索している．具体的には，探索の範囲を n/2 → n/4…と半減させながら，現在の探索範囲の中央の値を基準とし，探索したい値 (=lot の値) が基準よりも小さいか大きいかを判断する．小さい場合は基準よりも前に，大きい場合は基準よりも後に探索したい値があるはずなので，前半分または後半分を次の探索範囲とし，探索範囲が 1 になるまで探索を続ける．探索の終了時に配列 win 内に探索したい値が見つかった場合は，そのインデックスを，見つからなかった場合は-1 を functionB が返す．なお，この探索方法を二分探索という．

(4)

問題に記載されているプログラムは，バブルソートとは異なる別のソート方法である．このプログラムは以下のようなアルゴリズムでソートを行う．

(1) 適当な数 (pivot という) を選択する
(2) pivot より小さい数を前方，大きい数を後方に移動させ，分割する
(3) 分割された各々のデータを，それぞれソートする

このプログラムでは，2. の分割方法として，以下のようなアルゴリズムを用いている．

i. pivot となる数 x として a[t] を選ぶ
ii. 探索範囲の左端から順に値を調べ，pivot 以上のものを見つけたらその位置を i とする
iii. 探索範囲の右端から順に値を調べ，pivot 以下のものを見つけたらその位置を j とする
iv. i が j より左側ならば，その二つの位置にある要素を入れ替え，ii. に戻る（ただし，次の探索は i の一つ右，j の一つ左から行う)

分割が終わった後は各々のデータをそれぞれソートするために，functionA(a, b, t, i-1) と functionA(a, b, j+1, w) とし，左半分と右半分に対して再帰的に functionA を呼び出すことでソートを行う．このアルゴリズムの計算量を考えると，pivot を選ぶ際には a[t] にアクセスすればいいため $O(1)$，分割を 1 回行うたびにデータが二分割されていくため，分割の深さは $O(\log n)$，ソートを行うためには配列の要素に線形アクセスするため $O(n)$ となり，これよりこのアルゴリズムの平均時間計算量は $O(n \log n)$ となる．なお，このソート方法をクイックソートという．

(5)

(イ) における判定を平均時間計算量 $O(1)$ で実現するには，functionB で配列へのアクセスを定数時間で行えるようにする必要がある．つまり，1 等に当選しているということを，定数時間でのアクセスで知る必要がある．そのためには，配列 win および配列 grade に対して，win の要素ではなく grade の要素の昇順としてソートを行うことで，win[0] に 1 等の当選番号が現れる．つまり，functionA での処理を win ではなく grade の要素を基準として行う．そのために，functionA の引数を (grade, win, 0, n-1) とする (ア)．これにより，(イ) を (k=functionB(win, lot, 1))!=-1 とすることで，平均時間計算量 $O(1)$ による判定が可能となる．

■■■ 所感 ■■■

2　計算機システムとシステムプログラム

■■■ 解答 ■■■

(1)

(1-1)
- 符号絶対値表現：1000　1111
- 1 の補数表現：1111　0000

(1-2)
- 最大値：0111　1111(127)
- 最小値：1000　0000(-128)

(1-3)

(1-3-1)

$$43 = (00101011)_2$$
$$-5 = (11111011)_2$$

加算すると $(100010110)_2$ となるが最上位の桁上げは無視してよいので $(00010110)_2 = 38$

(1-3-2) 減算を加算のみで表現することができる

(1-4)

(1-4-1) $c = 1$ であるかどうか

(1-4-2) $a = b = 0$ かつ $s = 1$ あるいは $a = b = 1$ かつ $s = 0$ であるかどうか

(2)

(2-1)
- デッドロック状態：資源を持っているプロセスが他の資源を待ってブロックしたままになる状態
- 飢餓状態：あるプロセスが持っている資源が，解放されるたびに他のプロセスに取られてしまうため，特定のプロセスにずっと割り当てられない状態

(2-2) アウエイ または アウイエ

(2-3)

(2-3-1)
- プロセス 1：
```
P(a);
top = top - 1;
stack(top) = push_item;
V(a);
```
- プロセス 2：
```
P(a);
pop_item = stack(top);
top = top + 1;
V(a);
```
- セマフォ a の初期値：1

(2-3-2)
- プロセス 3：
```
while(true){
  P(b);
  m = m + 1;
  V(c);
}
```
- プロセス 4：
```
while(true){
  P(c);
  print(m);
  V(b);
}
```
- セマフォ b の初期値：0
- セマフォ c の初期値：0

■■■ 解説 ■■■

(1)

(1-1)

- 符号絶対値表現：最上位ビットが 1 の時は負の数，0 のときは正の数
- 1 の補数表現：ビット反転

(1-2)

n ビットの 2 の補数表現の範囲は，$-2^{(n-1)}$ $2^{(n-1)-1}$

(1-3)

(1-3-1)

省略

(1-3-2)

これによってハードウェア設計の際に減算回路を用意する必要がなくなり，構造を簡単にすることができる.

(1-4)
(1-4-1)
　省略

(1-4-2)
　(1-3-1) から補数表現の場合は，加算器の最上位ビットからの桁上げを確認するだけではオーバーフローが発生したかどうか判定することはできない.

(2)
(2-1)
　省略

(2-2)
　省略

(2-3)
　省略
(2-3-1)
　片方のプロセスが動作しているとき，a の値が 0 になるため，そのプロセスの動作が終了する（V(a) が実行される）まで，もう片方のプロセスは動作することができなくなる.

(2-3-2)
　m は事前に 1 に初期化されているので，プロセス $4 \to 3 \to 4 \to \ldots$ の順番で動作させなければならない.

■■■ 所感 ■■■

- 前半：整数の表現と算術演算に関する問題．例年に比べると回路が絡んだりせず，素直で解きやすい問題が多かった
- 後半：排他制御に関する問題．過去問でもこの分野を扱った問題は多くなかった (気がする) ので，問題を見たときに驚いた人もいると思う．難易度自体はそこまで高くなく，授業内容を理解したうえで問題文をよく読めば大丈夫.

3　離散構造

■■■ 解答 ■■■

(1)
　(1-1)
　　(1-1-1) 解説参照
　　(1-1-2) 解説参照
　(1-2)
　　(1-2-1) $\neg E = \exists x \forall y \forall z((\neg p(y) \lor q(y)) \land p(x) \land \neg q(z))$
　　　(ただし変数名 x, y, z は，互いに重複しない範囲内で他の名前に置き換え可)
　　(1-2-2) $\neg E' = \forall y \forall z((\neg p(y) \lor q(y)) \land p(a) \land \neg q(z))$
　　　(ただし変数名 y, z, 定数名 a は，互いに重複しない範囲内で他の名前に置き換え可)
　　(1-2-3) 解説参照
(2)
　(2-1)
　　(2-1-1)
　　- R_1：対称律
　　- R_2：反射律，反対称律
　　- R_3：反射律．反対称律，推移律
　　(2-1-2) R_3 のみ
　(2-2)
　　(2-2-1) 解説参照
　　(2-2-2)

$$w = s \cup t$$
$$z = s \cup t$$

　　(2-2-3) 解説参照

■■■ 解説 ■■■

(1)
(1-1)
(1-1-1)
　$\forall x P(x)$ を仮定する.

1．$\forall x P(x)$ (仮定)

2． $\forall x P(x) \to P(x)$ (公理 A4 に $t = x$ を代入)
3． $P(x)$ (1, 2 と推論規則 B1)

$\underline{\forall x P(x)}$ のもと $\underline{P(x)}$ が成り立つので，$\underline{\forall x P(x) \vdash P(x)}$.

(1-1-2)
$\forall x(p(x) \to q(x)) \land \exists x\, p(x)$ を仮定する．

1． $\forall x(p(x) \to q(x)) \land \exists x\, p(x)$ (仮定)
2． $\forall x(p(x) \to q(x))$ (1 と定理 T1)
3． $\exists x\, p(x)$ (1 と定理 T2)
4． $p(x) \to q(x)$ (2 と定理 T4)
5． $(p(x) \to q(x)) \to (\exists x\, p(x) \to \exists x\, q(x))$ (定理 T3)
6． $\exists x\, p(x) \to \exists x\, q(x)$ (4, 5 と推論規則 B1)
7． $\exists x\, q(x)$ (3, 6 と推論規則 B1)

したがって，$\underline{\forall x(p(x) \to q(x)) \land \exists x\, p(x) \vdash \exists x\, q(x)}$.
$\forall x(p(x) \to q(x)) \land \exists x\, p(x)$ は自由変数を含まない閉論理式なので，演繹定理 D1 より

$$\vdash (\forall x(p(x) \to q(x)) \land \exists x\, p(x)) \to (\exists x\, q(x))$$

が成り立つ．
(1-2)
(1-2-1)
$\neg E$ の和積標準形は，

$$\neg E = \forall x(\neg p(x) \lor q(x)) \land \exists x\, p(x) \land \forall x\, \neg q(x)$$

$\neg E$ を冠頭標準形に変形する．まず，存在記号 ($\exists$) を論理式の先頭に出す．

$$\neg E = \exists x(\forall x(\neg p(x) \lor q(x)) \land p(x) \land \forall x\, \neg q(x))$$

次に，全称記号 ($\forall$) を論理式の先頭に出す．変数名が同じくて論理式の変形がうまくできない場合には，変数名をスコープごとに異なる名前で付け替える．

$$\neg E = \exists x(\forall y(\neg p(y) \lor q(y)) \land p(x) \land \forall z\, \neg q(z))$$
$$\neg E = \exists x \forall y \forall z((\neg p(y) \lor q(y)) \land p(x) \land \neg q(z))$$

(1-2-2)
論理式 $\neg E = \exists x \forall y \forall z((\neg p(y) \lor q(y)) \land p(x) \land \neg q(z))$ をスコーレム標準形に変形する．存在記号の変数 x を適当な定数 a に置き換えると，

$$\neg E' = \forall y \forall z((\neg p(y) \lor q(y)) \land p(a) \land \neg q(z))$$

(1-2-3)
空節を導出するために，適切な基礎節を選択する．

$$C_1 = \neg p(y) \lor q(y)$$
$$C_2 = p(a)$$
$$C_3 = \neg q(z)$$

エルブラン領域から適切な定数を選んで，変数 (y と z) に代入する．この場合，すべての変数に a を代入すればよい．

$$C_1' = \neg p(a) \lor q(a)$$
$$C_3' = \neg q(a)$$

導出原理で空節を導出する．($\oplus$ を導出の記号とする)

$$C_1' \oplus C_2 \Rightarrow q(a)$$
$$q(a) \oplus C_3' \Rightarrow \text{空節}$$

よって，$\neg E'$ は充足不能である．
(2)
(2-1)
(2-1-1)
二項関係 $R = \{(s,t) \mid s,t \in V\}$ を想定する．

- 反射律：$\forall x \in V, xRx$
- 対称律：$\forall x,y \in V, xRy \to yRx$
- 反対称律：$\forall x,y \in V, xRy \land yRx \to x = y$
- 推移律：$\forall x,y,z \in V, xRy \land yRz \to xRz$

二項関係 R_1，R_2 および R_3 において，上記の条件を満たしているか確認すればよい．

(2-1-2)
順序関係 (partial ordered set = 半順序関係) は，反射律，反対称律，および推移律をすべて満たす二項関係であり，$\underline{R_3 \text{ のみ}}$ が順序関係の条件を満たしている．

(2-2)
(2-2-1)

二項関係 $\preceq_{\mathcal{P}(X)}$ が反射律，反対称律，および推移律をすべて満たしているか確認すればよい．

- 反射律：
$\mathcal{P}(X)$ の各元 x に対し，$x \subseteq x$
よって，$\forall x \in \mathcal{P}(X), x \preceq_{\mathcal{P}(X)} x$
- 反対称律:
$\mathcal{P}(X)$ の各元 x, y に対し，$(x \subseteq y) \wedge (y \subseteq x)$ ならば $x = y$
よって, $\forall x, y \in \mathcal{P}(X), (x \preceq_{\mathcal{P}(X)} y) \wedge (y \preceq_{\mathcal{P}(X)} x) \to x = y$
- 推移律: $\mathcal{P}(X)$ の各元 x, y, z に対し，$(x \subseteq y) \wedge (y \subseteq z)$ ならば $(x \subseteq z)$
よって，$\forall x, y, z \in \mathcal{P}(X), (x \preceq_{\mathcal{P}(X)} y) \wedge (y \preceq_{\mathcal{P}(X)} z) \to (x \preceq_{\mathcal{P}(X)} z)$

したがって，二項関係 $\preceq_{\mathcal{P}(X)}$ は順序関係であり，$(\mathcal{P}(X), \preceq_{\mathcal{P}(X)})$ は順序集合である．

(2-2-2)

$\preceq_{\mathcal{P}(X)}$ と $upper(A)$ の定義より，$w = upper(A)$ ならば $\forall x \in A, x \subseteq w$ である．したがって，(問題文より)

$$s \subseteq w$$
$$t \subseteq w$$

この条件を満たす要素数最小の w は $\underline{w = s \cup t}$ である．

同様に $lower(A)$ の定義より，$z = lower(A)$ ならば $\forall x \in A, z \subseteq x$ である．したがって，

$$z \subseteq s$$
$$z \subseteq t$$

この条件を満たす要素数最大の z は $\underline{z = s \cap t}$ である．

(2-2-3)

任意の $u \in \mathcal{P}(X)$ に対し，$u \cup c(u)$ は $\mathcal{P}(X)$ の最大元，つまり X である ($\mathcal{P}(X)$ は X のべき集合であるから)．また，$u \cap c(u)$ は $\mathcal{P}(X)$ の最小元，つまり ϕ (空集合) である．

よって，$c(u)$ は u の補集合 (u^C) である．

ド・モルガンの法則より，$(s \cup t)^C = s^C \cap t^C$ が成り立つので，$\underline{c(s \cup t) = c(s) \cap c(t)}$ も自明に成り立つ．

■■■ 所感 ■■■参考資料

- 情報論理学 後半部 講義資料 (CLE)
- ウィキペディア / 二項関係, 順序集合
- LaTeX Mathematical Symbols (https://reu.dimacs.rutgers.edu/Symbols.pdf)
- LaTeX/Mathematics - Wikibooks (https://en.wikibooks.org/wiki/LaTeX/Mathematics)

4　計算理論

解答

(1)

(1-1) 図 2 参照

(1-2) 図 2 参照

(1-3) 解説参照

(1-4)

$$(\epsilon, Z)/Z$$
$$(\epsilon, 0)/0$$
$$(\epsilon, 1)/1$$

(2)

(2-1) 解説参照

(2-2)
- (イ)：解説参照
- (ウ)：$aAbA$
- (エ)：$bAaA$

解説

(1)

(1-1)

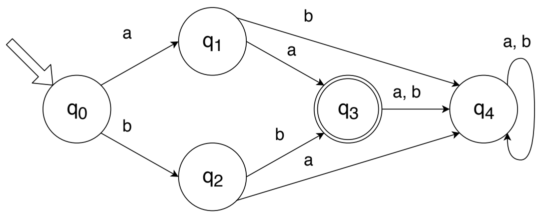

図 1　(1-1) 決定性有限オートマトン

(1-2)

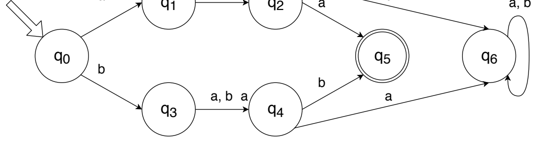

図 2　(1-2) 決定性有限オートマトン

(1-3)

言語 $\{p \in \sum_{ab}^* \mid |p| \geq, p$ は回文 $\}$ を L とする．

L を正則言語とする．このとき，正整数 n に対して，$v = a^n b a^n$ は L に含まれる．反復補題の条件を満たすように v を xyz に分解すると，$|xy| \leq n$ より，b は z に含まれる．したがって，$xyyz$ は，b の左右にある a の個数が異なるため L に属さない．

よって，L は正則言語ではない．

(1-4)

省略

(2)

(2-1)

$k-1$ 回文法規則を適用した後の文形式が $v_1 A v_2$(v_1，v_2 は文字列) のとき，k 回目で適用規則 $A \to \epsilon$ を適用すれば，k 回の導出で文字列を導出できる．ここで，v_1，v_2 は $k-1$ 回以下の適用で導出できる文字列であるから，

$$|v_1|_a = |v_1|_b$$
$$|v_2|_a = |v_2|_b$$

である．よって，$|w|_a = |w|_b$

(2-2)

導出 $A \Rightarrow \epsilon = w$ より，$w \in L_1$ が成立する．

所感

5　ネットワーク

■■■ 解答 ■■■

■■■ 解説 ■■■

■■■ 所感 ■■■作成者募集中

6　電子回路と論理設計

■■■ 解答 ■■■

(1)

(1-1) 4

(1-2) $2C_1R_{MOS}\ln 2 + C_2R_{MOS}\ln 2$

(2)

(2-1) $(0, 1, 0, 0)$

(2-2)

$$s_1 = p_2 \lor p_3$$
$$s_0 = p_1 \lor p_3$$

(3)

(3-1) 図 8 参照

(3-2) (A) S_0 (B) S_1 (C) S_3
(D) S_1 (E) S_2 (F) S_0
(G) S_2 (H) S_3 (I) S_1
(J) S_3 (K) S_0 (L) S_2

(3-3)

$$\begin{aligned} D_1 = &\neg Q_1 \land \neg Q_0 \land x_1 \\ &\lor \neg Q_1 \land Q_0 \land x_0 \\ &\lor Q_1 \land Q_0 \land \neg x_0 \\ &\lor Q_1 \land \neg Q_0 \land \neg x_1 \\ D_0 = &Q_0 \land \neg x_1 \land \neg x_0 \\ &\lor \neg Q_0 \land x_1 \\ &\lor \neg Q_0 \land x_0 \end{aligned}$$

■■■ 解説 ■■■

(1)

(1-1)

表 1　(1-1) 真理値表

x	y	z
0	0	0
0	V_{DD}	0
V_{DD}	0	0
V_{DD}	V_{DD}	V_{DD}

表 1 より，$z = x \land y$ である．

(1-2)

pMOS のオン条件より，x が V_{DD} に変化してから z が $0.5V_{DD}$ になるまでの時間 T は，

- c(図 3 参照) が V_{DD} から $0.5V_{DD}$ に変化するまでの時間 T_1
- c が $0.5V_{DD}$ になってから z が 0 から $0.5V_{DD}$ に変化するまでの時間 T_2

の和である．

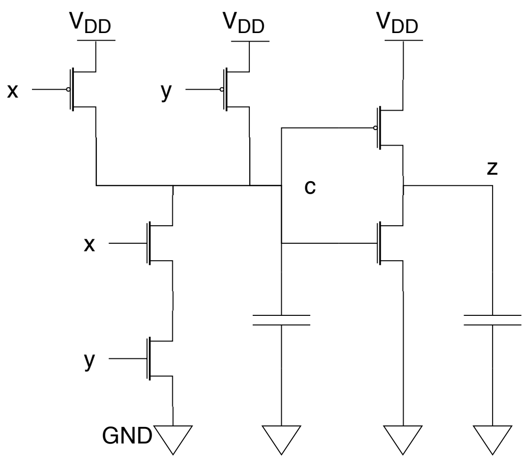

図 3　(1-2) 回路図

x が V_{DD} に変化した瞬間を $t_1 = 0$ とする．x が V_{DD} に変化したときの等価回路は，図 4 である．図 4 より，

$$V_c(t_1) = 2C_1R_{MOS}\frac{dV_c(t_1)}{dt_1} \tag{1}$$

定常状態のとき，c の電圧は 0 であるから，式 (1) の解は，

$$V_c(t_1) = A\exp\left(-\frac{t_1}{2C_1R_{MOS}}\right) \text{A は定数}$$

$V_c(+0) = V_c(-0) = V_{DD}$ であるから，$A = V_{DD}$ である．したがって，T_1 は，

$$\begin{aligned}\frac{1}{2}V_{DD} &= V_{DD}\exp\left(-\frac{T_1}{2C_1R_{MOS}}\right)\\ \exp\left(\ln\frac{1}{2}\right) &= \exp\left(-\frac{T_1}{2C_1R_{MOS}}\right)\\ \frac{T_1}{2C_1R_{MOS}} &= \ln 2\\ T_1 &= 2C_1R_{MOS}\ln 2 \end{aligned} \tag{2}$$

c が $0.5V_{DD}$ に変化した瞬間を $t_2 = 0$ とする．c が $0.5V_{DD}$ に変化したときの等価回路は，図 5 である．図 5 より，

$$V_z(t_2) = C_2R_{MOS}\frac{dV_z(t_2)}{dt_2} \tag{3}$$

定常状態のとき，z の電圧は V_{DD} であるから，式 (3) の解は，

$$V_z(t_2) = B\exp\left(-\frac{t_2}{C_2R_{MOS}}\right) + V_{DD}\text{B は定数}$$

$V_z(+0) = V_z(-0) = 0$ であるから，$B = -V_{DD}$ である．したがって，T_2 は，

$$\begin{aligned}\frac{1}{2}V_{DD} &= -V_{DD}\exp\left(-\frac{T_2}{C_2R_{MOS}}\right) + V_{DD}\\ \exp\left(\ln\frac{1}{2}\right) &= \exp\left(-\frac{T_2}{C_2R_{MOS}}\right)\\ \frac{T_2}{C_2R_{MOS}} &= \ln 2\\ T_2 &= C_2R_{MOS}\ln 2 \end{aligned} \tag{4}$$

式 (2)，(4) より，

$$T = 2C_1R_{MOS}\ln 2 + C_2R_{MOS}\ln 2$$

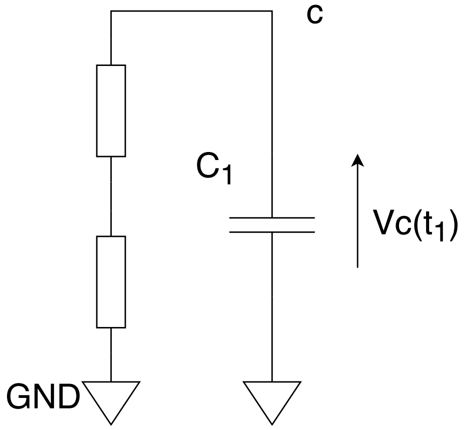

図 4　(1-2) x が V_{DD} に変化したときの等価回路

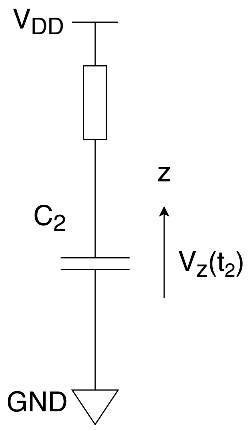

図 5　(1-2) c が $0.5V_{DD}$ に変化したときの等価回路

(2)

(2-1)

入力が $(0, 1, 1, 0)$ であるから，出力は X を 2 ビット左にしたものである．よって，$Y = (0, 1, 0, 0)$

(2-2)

s_1，s_0 のカルノー図を図 6，図 7 に示す．

カルノー図より，

$$s_1 = p_2 \vee p_3$$
$$s_0 = p_1 \vee p_3$$

s_1 (p_3p_2 \ p_1p_0)	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

図 6　(2-2) s_1 のカルノー図

s_0 (p_3p_2 \ p_1p_0)	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	1	1	1	1
10	1	1	1	1

図 7　(2-2) s_0 のカルノー図

(3)

(3-1)

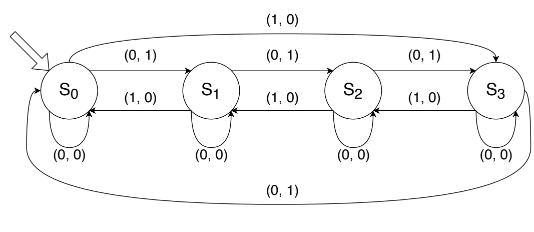

図 8　(3-1) 状態遷移図

(3-2)

状態遷移図から，

(A) S_0 (B) S_1 (C) S_3

(D) S_1 (E) S_2 (F) S_0

(G) S_2 (H) S_3 (I) S_1

D_1 (Q_1Q_0 \ x_1x_0)	00	01	11	10
00	0	0	X	1
01	0	1	X	0
11	1	0	X	1
10	1	1	X	0

図 9 (3-3) D_1 のカルノー図

D_0 (Q_1Q_0 \ x_1x_0)	00	01	11	10
00	0	1	X	1
01	1	0	X	0
11	1	0	X	0
10	0	1	X	1

図 10 (3-3) D_0 のカルノー図

(J) S_3 (K) S_0 (L) S_2

(3-3)

D_1，D_0 のカルノー図を図 9，図 10 に示す．
カルノー図より，

$$\begin{aligned} D_1 &= \neg Q_1 \wedge \neg Q_0 \wedge x_1 \\ \vee &= \neg Q_1 \wedge Q_0 \wedge x_0 \\ \vee &= Q_1 \wedge Q_0 \wedge \neg x_0 \\ \vee &= Q_1 \wedge \neg Q_0 \wedge \neg x_1 \\ D_0 &= Q_0 \wedge \neg x_1 \wedge \neg x_0 \\ \vee &= \neg Q_0 \wedge x_1 \\ \vee &= \neg Q_0 \wedge x_0 \end{aligned}$$

所感

7 信号処理

解答

解説

(1)

所感

作成者募集中