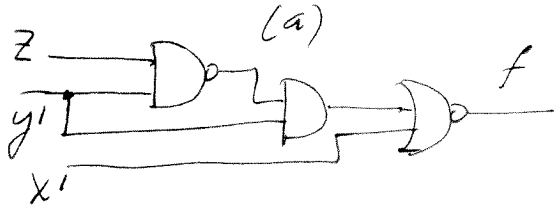
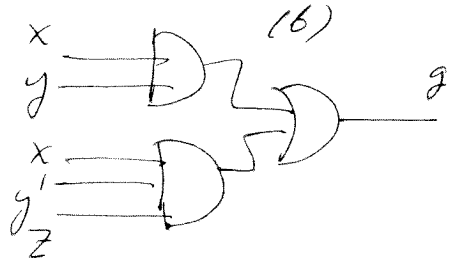
SOLUTIONS

FINAL EXAM ENEL 353 FALL 2006

(1) For the given circuits,

(a) Use algebraic transformations to prove or disprove that the two circuits implement the same function:





$f = \overline{\overline{zy'} \cdot y'} \vee x' =$

$= (\bar{z} \vee y) \cdot \bar{y} \cdot \bar{x} =$

$= (\bar{z}\bar{y} \vee y\cdot\bar{y}) \cdot x =$

$= (\overline{\bar{z} \vee y}) \cdot x = (z \vee y)x =$

$= xz \vee xy$ //

$g = xy \vee xy'z =$

$= xy \vee x\bar{y}z = x(y \vee \bar{y}z) =$

$= x(y(\bar{z} \vee z) \vee \bar{y}z) = x(y\bar{z} \vee yz \vee \bar{y}z) =$

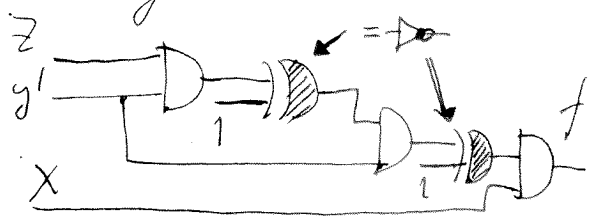
$= x(y(\bar{z} \vee z) \vee (y \vee \bar{y})z) =$

$= x(y \vee z) = xy \vee xz$ //

The circuits implement the same function.

(b) Re-design circuit (a) using 2-input AND and EXOR

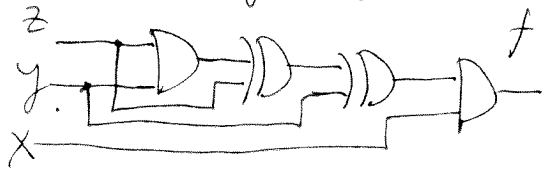
Way 1: on the circuit



Way 2: algebraically:

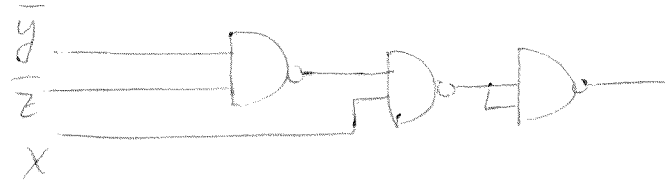
$xz \vee xy = x(z \vee y) =$

$= x(z \oplus y \oplus zy)$



(c) Redesign using **NAND** gates:

$$xy + xz = x(y + z) = x\,\overline{\bar{y}\cdot\bar{z}} = \overline{x\cdot\overline{\bar{y}\cdot\bar{z}}}$$

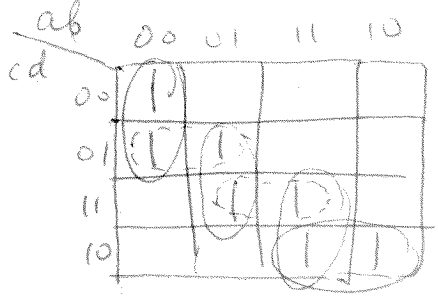


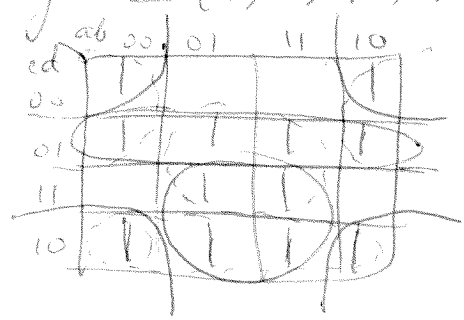
2. Consider $f = \Sigma(0,4,5,10,11,13,15)$

$g = \Pi(1,2,12,14)$.

(a) Draw K-maps and find minimal SOP & POS

$g = \Sigma(0,3,4,5,6,7,8,5,10,11,13,15)$





$f_{SOP} = \bar{a}\bar{b}\bar{c} \vee \bar{a}bd \vee ab\bar{c} \vee a\bar{c}\bar{d}$

or

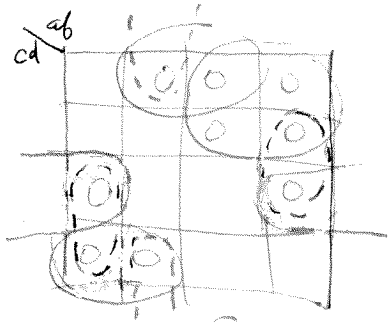
$f_{SOP} = \bar{a}\bar{b}\bar{c} \vee \bar{a}\bar{c}d \vee b\bar{c}d \vee a\bar{c}\bar{d}$

$g_{SOP} = \bar{c}d \vee b\bar{d} \vee b\bar{c}$

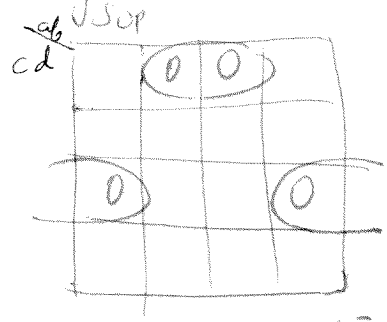
or

$g = \bar{b}\bar{c} \vee bd \vee c\bar{d}$

f_{POS}



g_{POS}



$f_{POS} = (\bar{a} \vee c)(\bar{b} \vee c \vee d)(b \vee \bar{c} \vee \bar{d})(a \vee \bar{c} \vee \bar{d})$

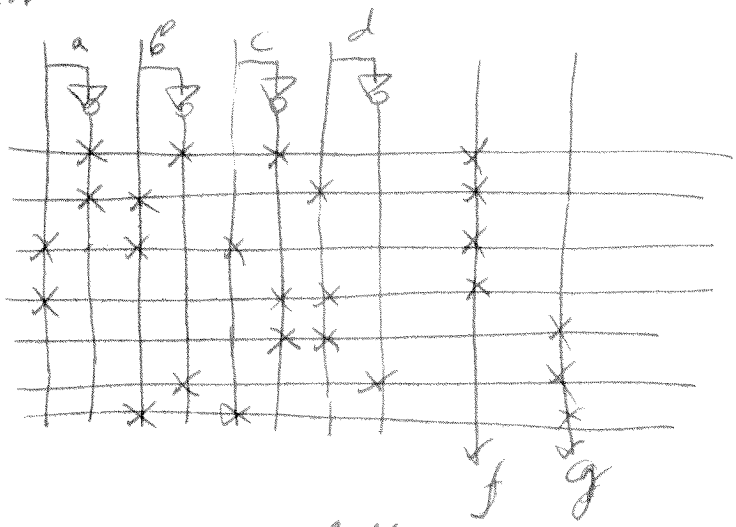
or $f_{POS} = (\bar{a} \vee c)(a \vee \bar{b} \vee d)(\bar{a} \vee b \vee \bar{d})(a \vee b \vee \bar{c})$

$g_{POS} = (\bar{b} \vee c \vee d)(b \vee \bar{c} \vee \bar{d})$

(b) Implement both functions using a PLA:

$f_{SOP} = \bar{a}\bar{b}\bar{c} \vee \bar{a}bd \vee abc \vee ac\bar{d}$

$g_{SOP} = \bar{c}d \vee \bar{b}\bar{d} \vee bc$

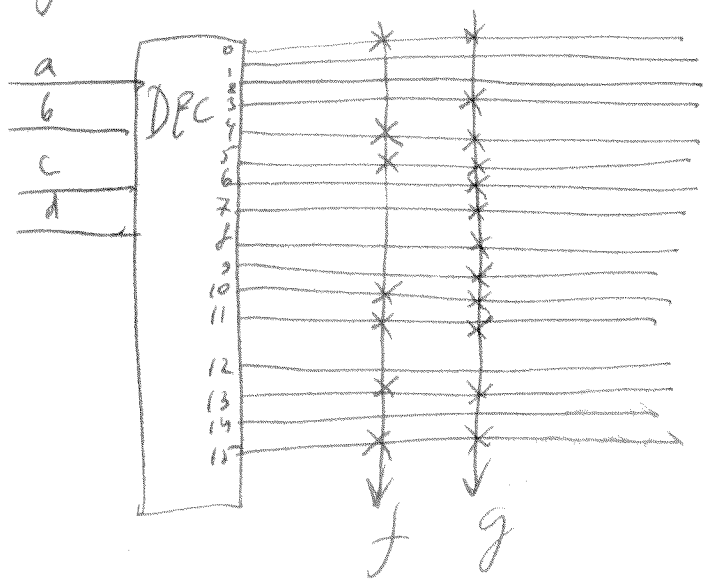


(c) Implement both functions using ROM.

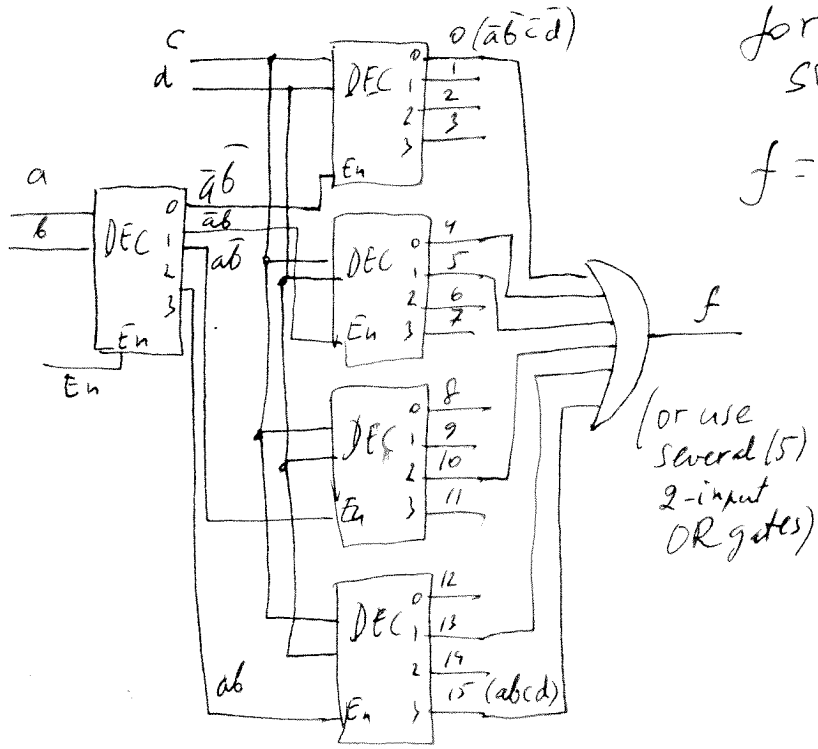
We use the standard (canonical), not minimized forms

$f = \sum (0, 4, 5, 10, 11, 13, 15)$

$g = \sum (0, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15)$



(d) Design the circuit for f using 2-to-4 decoders and any additional combinational gates:



for a DEC, we need a standard (canonical) form:

$f = \Sigma(0, 4, 5, 10, 13, 15)$

and use a decoder-tree

(e) Design the circuit for g using 2-to-1 multiplexer (no other gates).

Let us use SOP of the g and Shannon expension.

$g = \bar{c}d \vee \bar{b}\bar{d} \vee bc$

$= \bar{b} \cdot g_{b=0} \vee b \cdot g_{b=1} =$

$= \bar{b}(\bar{c}d \vee \bar{d}) \vee b(\bar{c}d \vee c)$

$= \bar{b}(\bar{c} \vee \bar{d}) \vee b(d \vee c)$

$g_{b=0} = \bar{c} \vee \bar{d} = \bar{c} \cdot g_{\substack{b=0 \\ c=0}} \vee c \cdot g_{\substack{b=0 \\ c=1}} =$

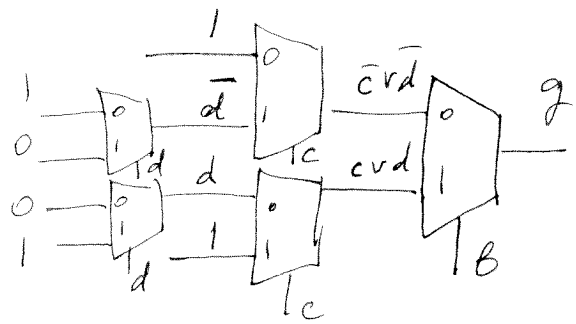
$= \bar{c}(1) \vee c(\bar{d})$

$g_{b=1} = d \vee c = \bar{c} g_{\substack{b=1 \\ c=0}} \vee c \cdot g_{\substack{b=1 \\ c=1}}$

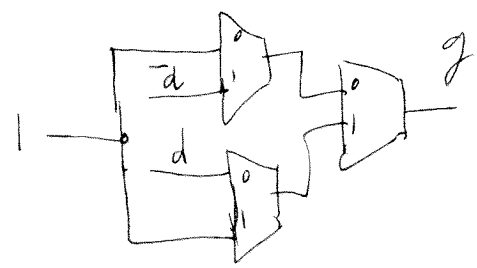
$= \bar{c}(d) \vee c(1)$

$g_{\substack{b=0 \\ c=1}} = \bar{d} = \bar{d}(1) \vee d(0)$

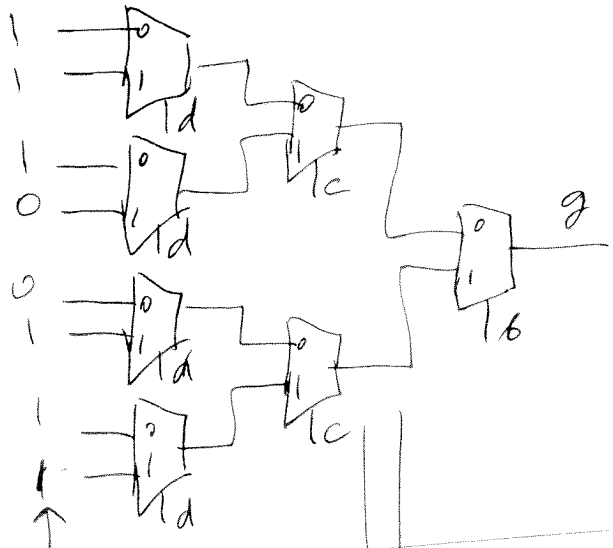
$g_{\substack{b=1 \\ c=0}} = d = \bar{d}(0) \vee d(1)$



or

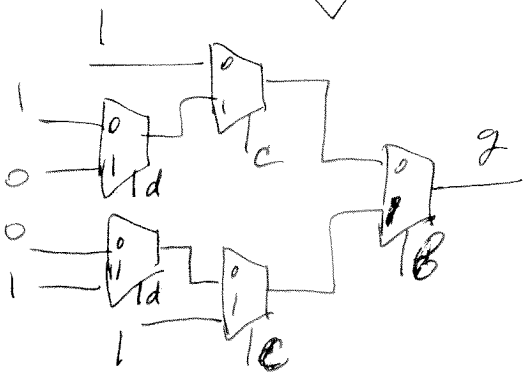


Alternatively, you can derive the complete binary MUX-tree (using the truth table for g) and then reduce it:

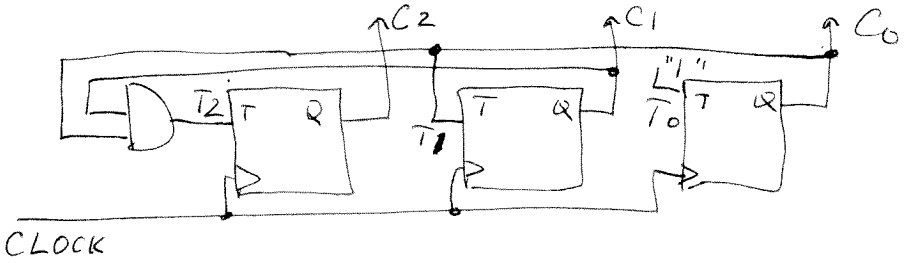


Since g depends only on b, c, d, we get

b	c	d	g
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



(3). Consider the circuit:



(a) Derive the state table and state diagram.

Here $Q_2 = C_2$, $Q_1 = C_1$, $Q_0 = C_0$, and no comb. inputs/outputs

C_2	C_1	C_0	C_2^+	C_1^+	C_0^+
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

use it for state

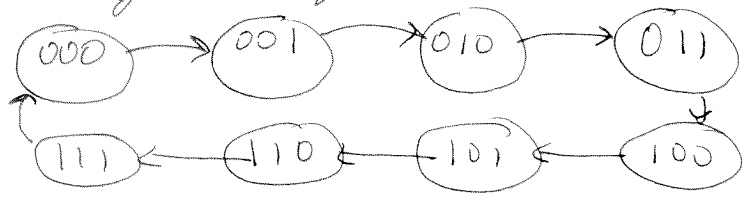
Since we have T FF, let us use the TFF characteristic equation to find C_2^+, C_1^+, C_0^+:

$$C_2^+ = C_2 \oplus T_2$$
$$C_1^+ = C_1 \oplus T_1$$
$$C_0^+ = C_0 \oplus T_0$$

Also, from the circuit,
$T_0 = 1$, $T_1 = C_0$,
$T_2 = \overline{C_0} \cdot \overline{C_1}$ $\Rightarrow$

$$\Rightarrow C_2^+ = C_2 \oplus \overline{C_0} \cdot \overline{C_1},$$
$$C_1^+ = C_1 \oplus C_0$$
$$C_0^+ = C_0 \oplus 1 = \overline{C_0}$$

Since there are 3 T FFs, we have $2^3 = 8$ states encoded using values of $C_2 C_1 C_0$: 000, 001, .., 111
The state diagram is given below:



There are no comb. inputs/outputs, and the only outputs are C_2, C_1, C_0 (same as the codes for the states), so this is the Moore-type FSM.

(b) What does this circuit implements?
The circuit is simply binary up-counter.

(c) Redesign the circuit using JK FFs.

Let us use the table from (a); and add JK columns using the JK FF characteristic table:

Q	Q^+	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

CURR STATE			NEXT STATE								
C_2	C_1	C_0	C_2^+	C_1^+	C_0^+	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	0	0	0	X	1	X	1	X	1

J_2:

C_2C_1 \ C_0	00	01	11	10
0	0	0	X	X
1	0	1	X	X

J_1:

C_2C_1 \ C_0	00	01	11	10
0	0	X	X	0
1	1	X	X	1

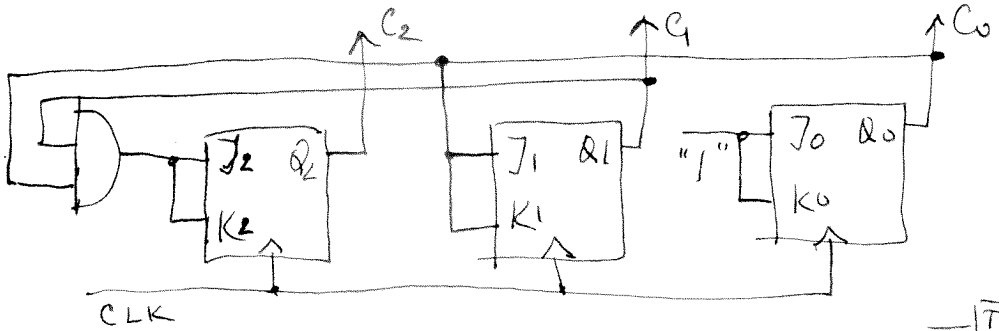
K_2:

C_2C_1 \ C_0	00	01	11	10
0	X	X	0	0
1	X	X	1	0

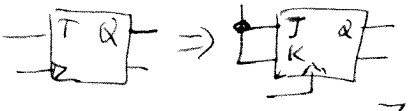
K_1:

C_2C_1 \ C_0	00	01	11	10
0	X	0	0	X
1	X	1	1	X

$J_2 = K_2 = C_1 C_0$ $\quad$ $J_2 = K_2 = C_0$ $\quad$ $J_0 = K_0 = 1$

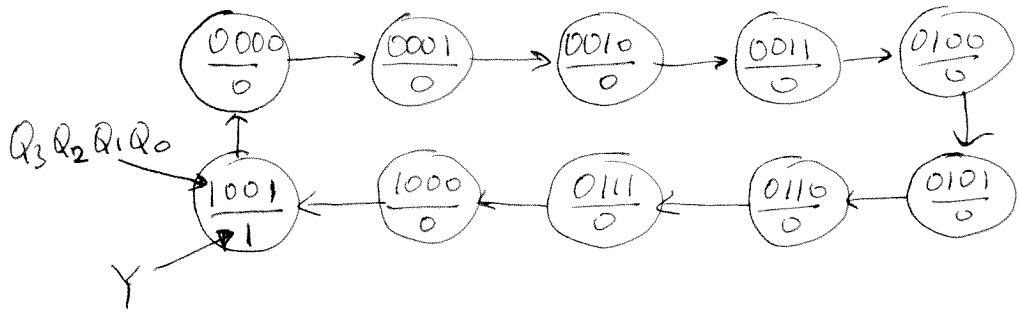


NOTE: You could use the initial circuit and substitute T FF with JK FF:



(4) Design a 4-bit BCD counter using T FFs. An additional output Y indicates that the count reaches "1000".

(a) Moore-type diagram:



(b) STATE TABLE:

CURR. STATE $Q_3\ Q_2\ Q_1\ Q_0$	NEXT STATE $Q_3^+\ Q_2^+\ Q_1^+\ Q_0^+$	$T_3\ T_2\ T_1\ T_0$	Output Y
0 0 0 0	0 0 0 1	0 0 0 1	0
0 0 0 1	0 0 1 0	0 0 1 1	0
0 0 1 0	0 0 1 1	0 0 0 1	0
0 0 1 1	0 1 0 0	0 1 1 1	0
0 1 0 0	0 1 0 1	0 0 0 1	0
0 1 0 1	0 1 1 0	0 0 1 1	0
0 1 1 0	0 1 1 1	0 0 0 1	0
0 1 1 1	1 0 0 0	1 1 1 1	0
1 0 0 0	1 0 0 1	0 0 0 1	0
1 0 0 1	0 0 0 0	1 0 0 1	1
1 0 1 X	X X X X	X X X X	X
1 1 X X	X X X X	X X X X	X

We used these equations to create the table

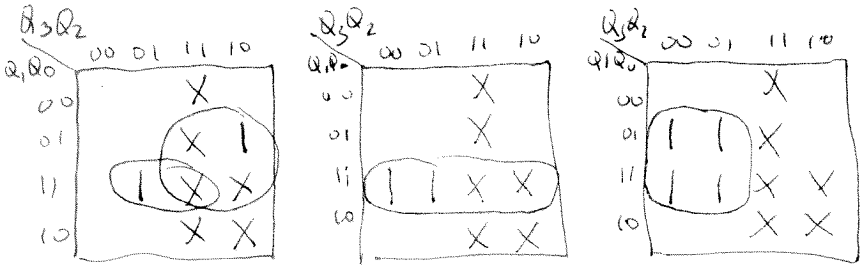
$T_3 = Q_3 \oplus Q_3^+$

$T_2 = Q_2 \oplus Q_2^+$

$T_1 = Q_1 \oplus Q_1^+$

$T_0 = Q_0 \oplus Q_0^+$

To minimize T_3, T_2, T_1, T_0, use K-maps:

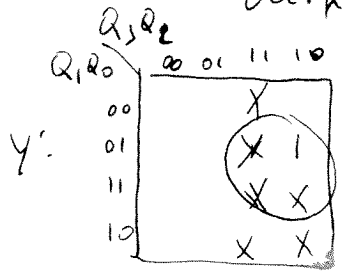


(c) Excitation equations

$T_3 = Q_3Q_0 \vee Q_2Q_1Q_0$ $\quad T_2 = Q_1Q_0$ $\quad T_1 = \bar{Q}_3Q_0$

$T_0 = 1$

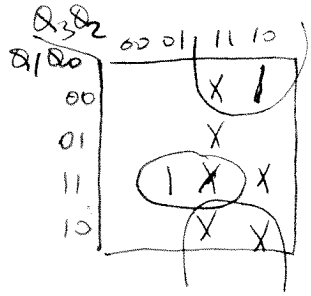
Output equation



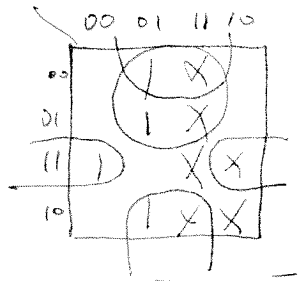
$Y = Q_3 Q_0$

(d) Re-design the system using D FFs:

For D FFs: $Q_3^+ = D_3$, $Q_2^+ = D_2$, $Q_1^+ = D_1$, $Q_0^+ = D_0$



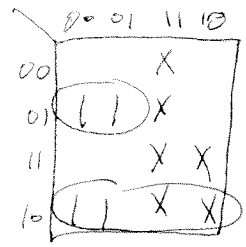
$D_3 = Q_3\bar{Q}_0 \vee Q_2 Q_1 Q_0$



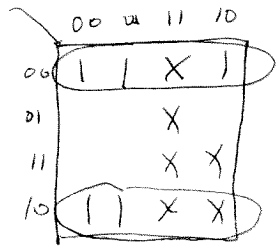
$D_2 = Q_2\bar{Q}_1 \vee Q_2\bar{Q}_0 \vee \bar{Q}_2 Q_1 Q_0$

$= Q_2 \overline{Q_1 Q_0} \vee \bar{Q}_2 Q_1 Q_0$

$= Q_2 \oplus Q_1 Q_0$



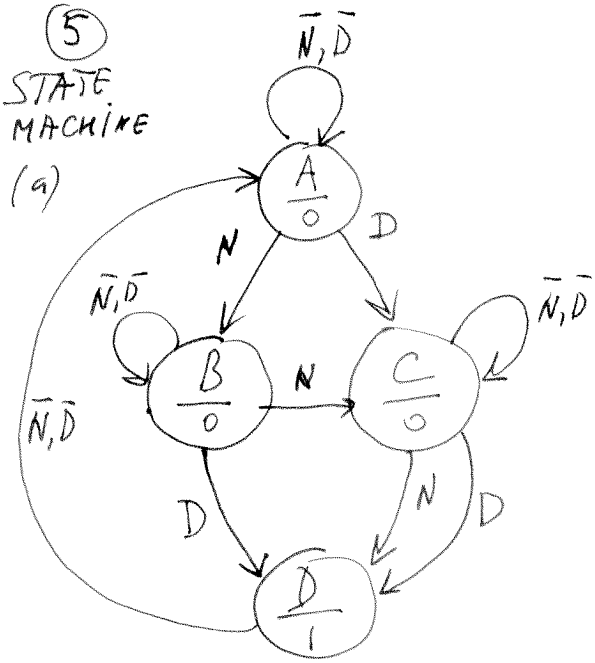
$D_1 = \bar{Q}_3\bar{Q}_1 Q_0 \vee Q_1\bar{Q}_0$



$D_0 = \bar{Q}_0$

(5) STATE MACHINE

(a)



STATES:

A ≡ no money deposited, OPEN=0

B ≡ ¢5, OPEN=0

C ≡ ¢10, OPEN=0

D ≡ ¢15, OPEN=1

N - nickel, D - dime and:

N̄, D̄ means "no coins inserted", or ND=00;

D means ND=01

N means ND=10

and ND=11 never happens.

(b) Since we have four states, we need 2 bits to represent states, or two FFs outputs:

$Q_1 Q_0$

A ⇒ $Q_1Q_0 = 00$

B ⇒ $Q_1Q_0 = 01$

C ⇒ $Q_1Q_0 = 10$

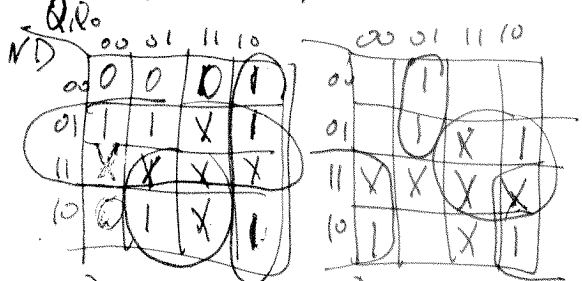
D ⇒ $Q_1Q_0 = 11$

STATE TABLE:

CURR. ST. Q_1Q_0	INPUT ND	NEXT ST. Q_1^+ Q_0^+	OUTPUT, OPEN
0 0	0 0	0 0	0
0 0	0 1	0 1	0
0 0	1 0	1 0	0
0 0	1 1	X X	X
0 1	0 0	0 1	0
0 1	0 1	1 0	0
0 1	1 0	X X	X
0 1	1 1	X X	X
1 0	0 0	1 0	0
1 0	0 1	1 1	0
1 0	1 0	X X	X
1 0	1 1	X X	X
1 1	0 0	0 0	1
1 1	0 1	X X	X
1 1	1 0	X X	X
1 1	1 1	X X	X

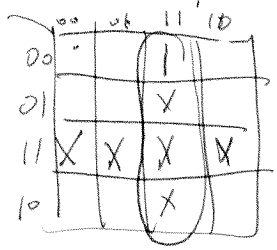
(c) Design a circuit on D FFs:

$D_1 = Q_1^+$, $D_0 = Q_0^+$



$D_1 = Q_1\bar{Q}_0 \vee D \vee Q_0 N$

$D_0 = \bar{Q}_1 Q_0 \bar{N} \vee Q_1 D \vee \bar{Q}_0 N$



OPEN = Q_1Q_0