UNIVERSITY OF CALIFORNIA, RIVERSIDE

Department of Computer Science and Engineering
Department of Electrical Engineering
CS/EE120A – Logic Design
Homework 2
Given May 7, Due May 14, 2001

16

1. Use the tabulation method to simplify the function $f(w,x,y,z) = \Sigma(0,2,5,7,13,15)$. List all the PI's, EPI's, cover lists, and solutions. (4)

Answer:

Group ID	Subcube Minterms	Subcube Value				Subcube Covered
		w	*x*	*y*	*z*	
G_0	m_0	0	0	0	0	✓
G_1	m_2	0	0	1	0	✓
G_2	m_5	0	1	0	1	✓
G_3	m_7	0	1	1	1	✓
	m_{13}	1	1	0	1	✓
G_4	m_{15}	1	1	1	1	✓

Group ID	Subcube Minterms	Subcube Value				Subcube Covered
		w	*x*	*y*	*z*	
G_0	$m_{0,2}$	0	0	-	0	
G_2	$m_{5,7}$	0	1	-	1	✓
	$m_{5,13}$	-	1	0	1	
G_3	$m_{7,15}$	-	1	1	1	
	$m_{13,15}$	1	1	-	1	✓

2 points for correct tables

Group ID	Subcube Minterms	Subcube Value				Subcube Covered
		w	*x*	*y*	*z*	
G_2	$m_{5,7,13,15}$	-	1	-	1	

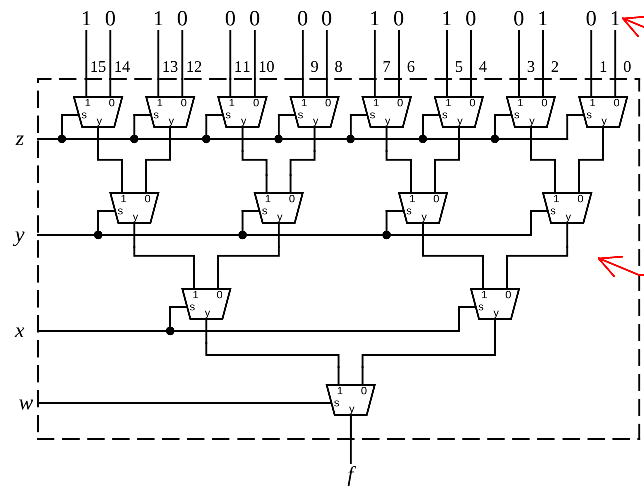
Prime Implicant Name	Prime Implicant Expression	Implicant Minterms	Function Minterms					
			0	2	5	7	13	15
P_1	$w'x'z'$	(0,2)	⊗	⊗				
P_2	$xy'z$	(5,13)			×		×	
P_3	xyz	(7,15)				×		×
P_4	xz	(5,7,13,15)			×	×	×	×
		EPI covered minterms:	0	2				
		Not covered minterms:			5	7	13	15

PI list: *w'x'z'*, *xy'z*, *xyz*, *xz*
EPI list: *w'x'z'*
cover list: *w'x'z'*, *xz*
f = *w'x'z'* + *xz*

2 points for correct PI, EPI and *f*.

2. Use only 2-to-1 multiplexers to implement the function $f(w,x,y,z) = \Sigma(0,2,5,7,13,15)$. Do not simplify it. (4)

Answer



2 points for correct connections to 0 and 1

2 points for correct circuit and labels

3. Use VHDL to design an 8-bit wide tri-state buffer. (4)

Answer

The VHDL code for an 8-bit wide tri-state buffer is shown below.

```
LIBRARY ieee;
USE IEEE.std_logic_1164.all;

ENTITY TriState_Buffer IS
   PORT(E: IN std_logic;
         D: IN std_logic_vector(7 downto 0);
         Y: OUT std_logic_vector(7 downto 0));
END TriState_Buffer;

ARCHITECTURE Behavioral OF TriState_Buffer IS
BEGIN
   IF E = '1' THEN
      Y <= D;
   ELSE
      Y <= (OTHERS => 'Z');          -- to get 8 Z values
   END IF;
END Behavioral;
```

4. Use basic gates to design a 3-to-8 decoder. (4)

Answer

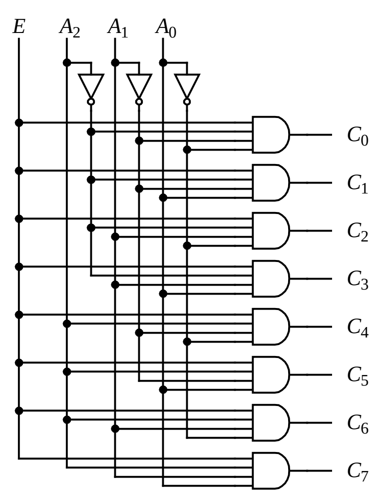
The truth table for the 3-to-8 decoder is as follows:

E	A_2	A_1	A_0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
0	×	×	×	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	1	0	0
1	0	1	1	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	1	0	0	0	0
1	1	0	1	0	0	1	0	0	0	0	0
1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0

The outputs C_i's are dependent on E and the three address lines A_i's. We have eight equations for the eight C_i's.

$C_0 = EA_2'A_1'A_0'$
$C_1 = EA_2'A_1'A_0$
$C_2 = EA_2'A_1A_0'$
$C_3 = EA_2'A_1A_0$
$C_4 = EA_2A_1'A_0'$
$C_5 = EA_2A_1'A_0$
$C_6 = EA_2A_1A_0'$
$C_7 = EA_2A_1A_0$

The circuit is



4 points for correct circuit