Solutions to Tutorial Sheet 6 - *Counters*

1.

$A_J = A_0A_1A_2$ $A_K = A_0$ $B_J = B_K = A_0A_1$

$C_J = A_0\overline{A_3}$ $C_K = A_0$ $D_J = D_K = 1$

J K	Q
0 0	Q
0 1	0
1 0	1
1 1	$\overline{Q}$

Minterm	Count	$A_3A_2A_1A_0$	Flipflop Inputs $A_J\ A_K$	$B_J\ B_K$	$C_J\ C_K$	$D_J\ D_K$
0	0	**0 0 0 0**	0 0	0 0	0 0	1 1
1	1	**0 0 0 1**	0 1	0 0	1 1	1 1
2	2	**0 0 1 0**	0 0	0 0	0 0	1 1
3	3	**0 0 1 1**	0 1	1 1	1 1	1 1
4	4	**0 1 0 0**	0 0	0 0	0 0	1 1
5	5	**0 1 0 1**	0 1	0 0	1 1	1 1
6	6	**0 1 1 0**	0 0	0 0	0 0	1 1
7	7	**0 1 1 1**	1 1	1 1	1 1	1 1
8	8	**1 0 0 0**	0 0	0 0	0 0	1 1
9	9	**1 0 0 1**	0 1	0 0	0 1	1 1
0	0	**0 0 0 0**				

Since the minterms follow the count exactly form 0 through to 9, the BCD counter must be using weights **8 4 2 1**.

2.

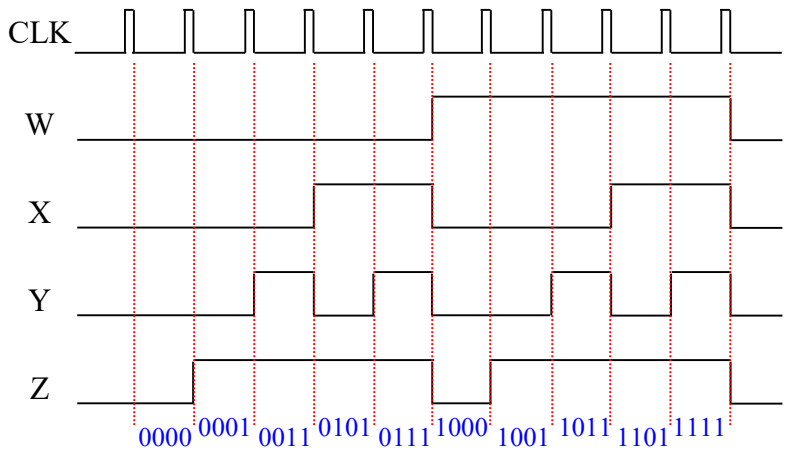
$J_A = K_A = 1$ $J_B = K_B = \overline{\overline{A}\overline{C}\overline{\overline{D}}} = A + C\overline{D}$ $J_D = C$

$J_C = K_C = \overline{\overline{A}BC\overline{\overline{D}}} = AB + C\overline{D}$ $K_D = \overline{\overline{A} + \overline{B} + \overline{C}} = ABC$

Minterm 8 4 2 1	Count	A B C D 1 2 4 2	Flipflop Inputs $J_A\ K_A$	$J_B\ K_B$	$J_C\ K_C$	$J_D\ K_D$
0	0	**0 0 0 0**	1 1	0 0	0 0	0 0
8	1	**1 0 0 0**	1 1	1 1	0 0	0 0
4	2	**0 1 0 0**	1 1	0 0	0 0	0 0
12	3	**1 1 0 0**	1 1	1 1	1 1	0 0
2	4	**0 0 1 0**	1 1	1 1	1 1	1 0
13	5	**1 1 0 1**	1 1	1 1	1 1	0 0
3	6	**0 0 1 1**	1 1	0 0	0 0	1 0
11	7	**1 0 1 1**	1 1	1 1	0 0	1 0
7	8	**0 1 1 1**	1 1	0 0	0 0	1 0
15	9	**1 1 1 1**	1 1	1 1	1 1	1 1
0	0	**0 0 0 0**				

This counter uses a **1 2 4 2 weighted code (this is the code that relates the sequence to a conventional count sequence of 0, 1, 2, 3, 4, 5, … etc.)**

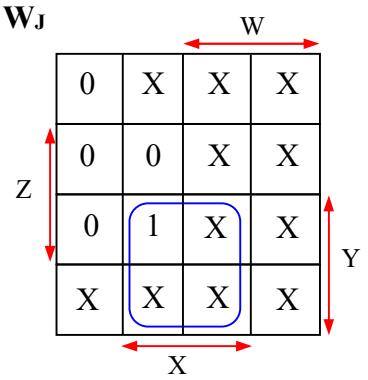
3.

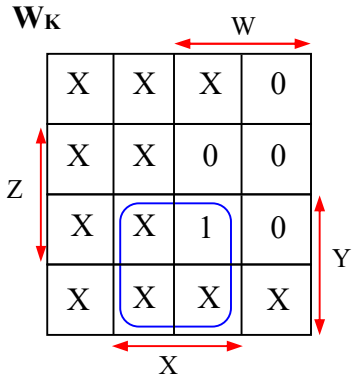


Minterm	WXYZ	Flipflop Inputs $W_J\ W_K$	$X_J\ X_K$	$Y_J\ Y_K$	$Z_J\ Z_K$
0	0 0 0 0	0 X	0 X	0 X	1 X
1	0 0 0 1	0 X	0 X	1 X	X 0
3	0 0 1 1	0 X	1 X	X 1	X 0
5	0 1 0 1	0 X	X 0	1 X	X 0
7	0 1 1 1	1 X	X 1	X 1	X 1
8	1 0 0 0	X 0	0 X	0 X	1 X
9	1 0 0 1	X 0	0 X	1 X	X 0
11	1 0 1 1	X 0	1 X	X 1	X 0
13	1 1 0 1	X 0	X 0	1 X	X 0
15	1 1 1 1	X 1	X 1	X 1	X 1
0	0 0 0 0				

Operation	J	K
Stay at 0	0	X
Stay at 1	X	0
Go to 0	X	1
Go to 1	1	X

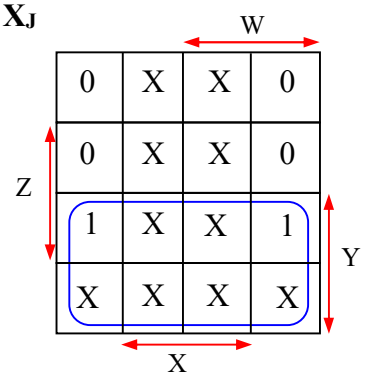
Carry out minimisation using KMaps. The unused states 2, 4, 6,10, 12 and 14 can be treated as don't care terms.



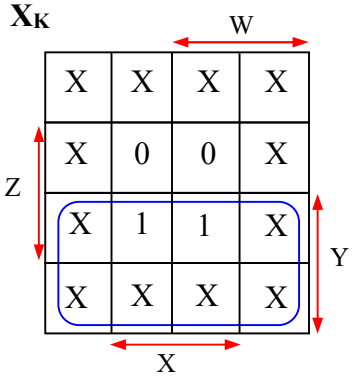


$$W_J = W_K = XY$$

X_J

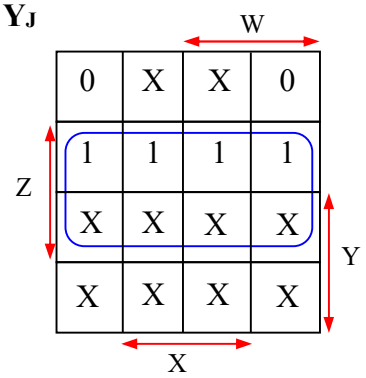


X_K



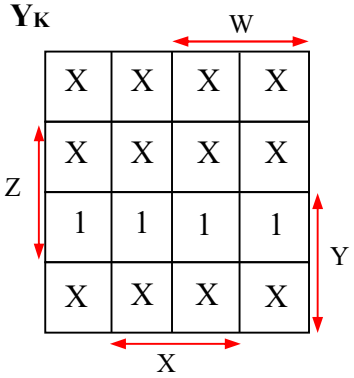
$X_J = X_K = Y$

Y_J



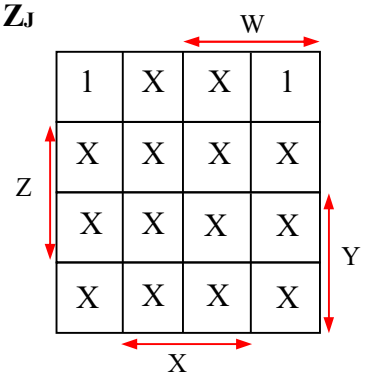
$Y_J = Z$

Y_K



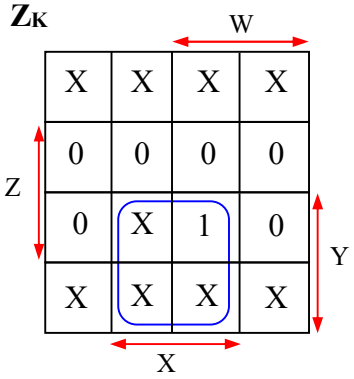
$Y_K = 1$ or Y or Z

Z_J



$Z_J = 1$ or $\overline{Z}$ or $\overline{X}$ or $\overline{Y}$

Z_K



$Z_K = XY$

Carry out a self check:

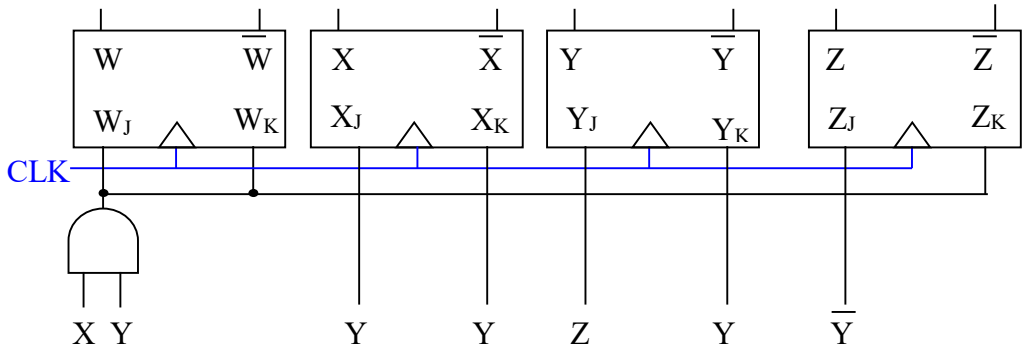
$W_J = W_K = XY$ $X_J = X_K = Y$ $Y_J = Z$ $Y_K = Y$ $Z_J = \overline{Y}$ $Z_K = XY$

Count	Minterm	WXYZ	Flipflop Inputs $W_J\ W_K$	$X_J\ X_K$	$Y_J\ Y_K$	$Z_J\ Z_K$
0	0	0 0 0 0	0 0	0 0	0 0	1 0
1	1	0 0 0 1	0 0	0 0	1 0	1 0
2	3	0 0 1 1	0 0	1 1	1 1	0 0
3	5	0 1 0 1	0 0	0 0	1 0	1 0
4	7	0 1 1 1	1 1	1 1	1 1	0 1
5	8	1 0 0 0	0 0	0 0	0 0	1 0
6	9	1 0 0 1	0 0	0 0	1 0	1 0
7	11	1 0 1 1	0 0	1 1	1 1	0 0
8	13	1 1 0 1	0 0	0 0	1 0	1 0
9	15	1 1 1 1	1 1	1 1	1 1	0 1
0	0	0 0 0 0				
	2	0 0 1 0	0 0	1 1	0 1	0 0
	4	0 1 0 0	0 0	0 0	0 0	1 0
	5	0 1 0 1	OK			
	6	0 1 1 0	1 1	1 1	0 1	0 1
	8	1 0 0 0	OK			
	10	1 0 1 0	0 0	1 1	0 1	0 0
	12	1 1 0 0	0 0	0 0	0 0	1 0
	13	1 1 0 1	OK			
	14	1 1 1 0	1 1	1 1	0 1	0 1
	0	0 0 0 0	OK			

J K	Q
0 0	Q
0 1	0
1 0	$\overline{1}$
1 1	$\overline{Q}$

No lockout, as all the unused states return to the main sequence. The counter employs a ***weighted code of 5 2 1 1***.

Final Counter Design:



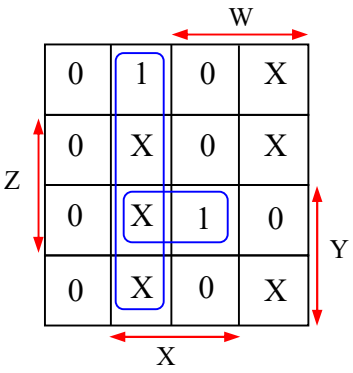
4. (a)

Minterm	WXYZ	Flipflop Inputs W_T	X_T	Y_T	Z_T
0	0 0 0 0	0	0	0	1
1	0 0 0 1	0	0	1	1
2	0 0 1 0	0	0	0	1
3	0 0 1 1	0	1	1	1
4	0 1 0 0	1	1	1	1
11	1 0 1 1	0	1	1	1
12	1 1 0 0	0	0	0	1
13	1 1 0 1	0	0	1	1
14	1 1 1 0	0	0	0	1
15	1 1 1 1	1	1	1	1
0	0 0 0 0				

Operation	T
Stay at 0	0
Stay at 1	0
Go to 0	1
Go to 1	1

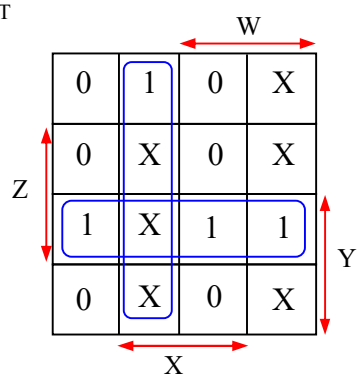
Minterms 5, 6, 7, 8, 9 and 10 can be treated as don't care terms.

W_T



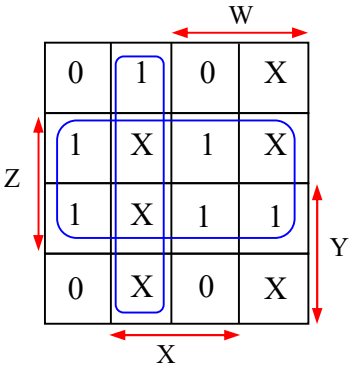
$$W_T = \overline{W}X + XYZ$$
$$= \overline{(\overline{W}X)}\,\overline{(XYZ)}$$

X_T



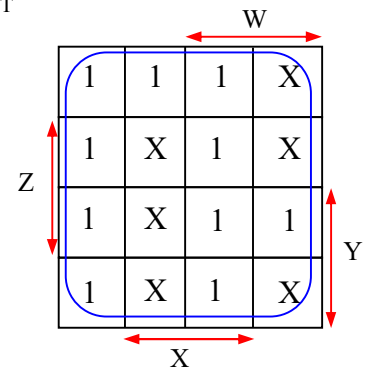
$$X_T = \overline{W}X + YZ$$
$$= \overline{(\overline{W}X)}\,\overline{(YZ)}$$

Y_T

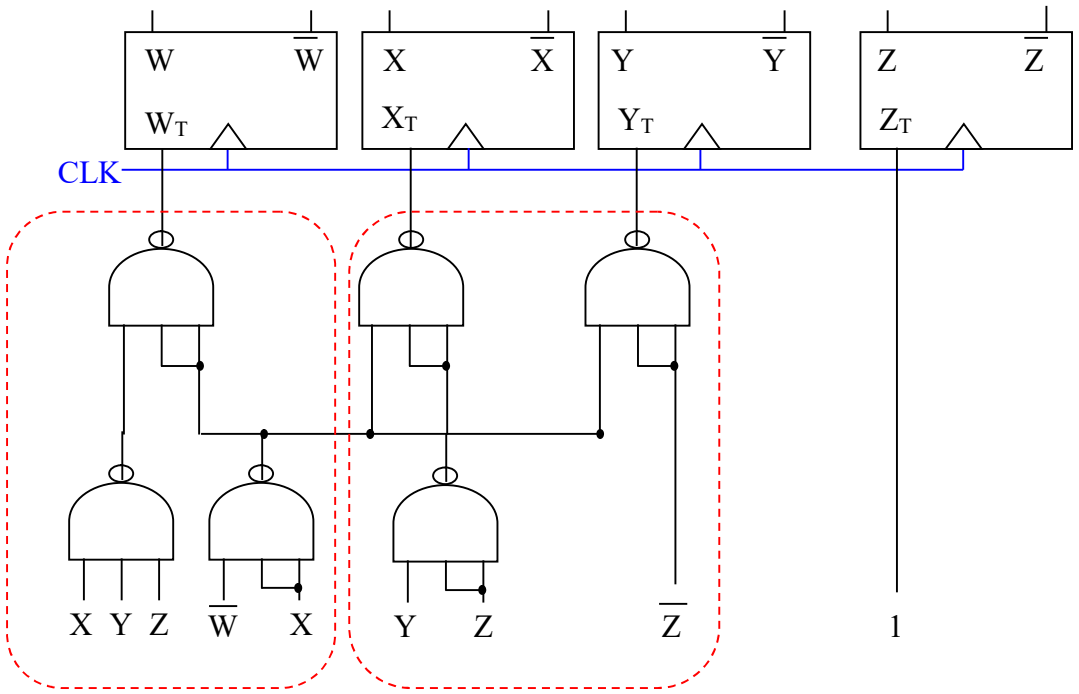


$$Y_T = \overline{W}X + Z$$
$$= \overline{(\overline{W}X)}.\overline{Z}$$

Z_T



$$Z_T = 1$$



Each dashed box represents one 7410 IC

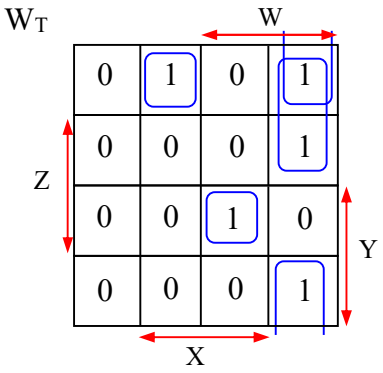
Carry out a self check:

Minterm	WXYZ	Flipflop Inputs W_T	X_T	Y_T	Z_T	
0	0 0 0 0	0	0	0	1	
1	0 0 0 1	0	0	1	1	
2	0 0 1 0	0	0	0	1	
3	0 0 1 1	0	1	1	1	
4	0 1 0 0	1	1	1	1	
11	1 0 1 1	0	1	1	1	
12	1 1 0 0	0	0	0	1	
13	1 1 0 1	0	0	1	1	
14	1 1 1 0	0	0	0	1	
15	1 1 1 1	1	1	1	1	
0	0 0 0 0					
5	0 1 0 1	1	1	1	1	
10	1 0 1 0	0	0	0	1	
11	1 0 1 1	OK				
6	0 1 1 0	1	1	1	1	**No Lockout in current design**
9	1 0 0 1	0	0	1	1	
10	1 0 1 0	OK				
7	0 1 1 1	1	1	1	1	
8	1 0 0 0	0	0	0	1	
9	1 0 0 1	OK				

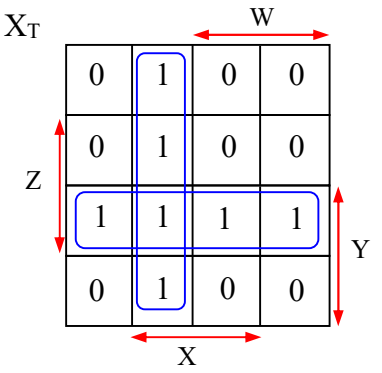
4. (b)

Minterm	WXYZ	Flipflop Inputs W_T	X_T	Y_T	Z_T
5	0 1 0 1	0	1	0	1
0	0 0 0 0				
6	0 1 1 0	0	1	1	0
0	0 0 0 0				
7	0 1 1 1	0	1	1	1
0	0 0 0 0				
8	1 0 0 0	1	0	0	0
0	0 0 0 0				
9	1 0 0 1	1	0	0	1
0	0 0 0 0				
10	1 0 1 0	1	0	1	0
0	0 0 0 0				

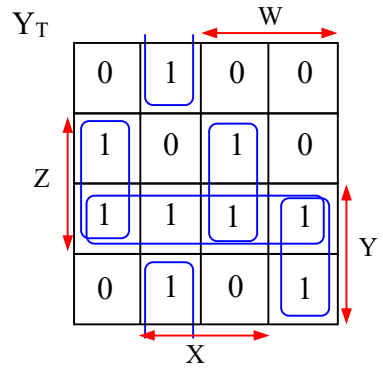
Operation	T
Stay at 0	0
Stay at 1	0
Go to 0	1
Go to 1	1



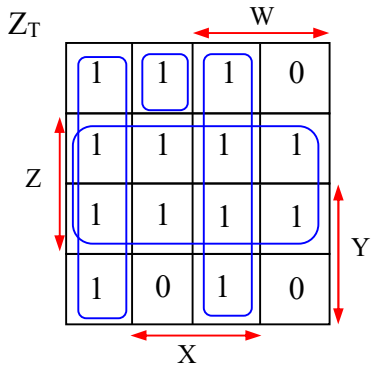
$$W_T = \overline{W}\,\overline{X}YZ + W\overline{X}\,\overline{Y}Z + W\overline{X}Y\overline{Z}$$
$$= \overline{(\overline{W}\,\overline{X}\,\overline{Y}Z)}\,\overline{(W\overline{X}\,\overline{Y}Z)}\,\overline{(W\overline{X}Y\overline{Z})}$$



$$X_T = \overline{W}X + YZ$$
$$= \overline{(\overline{W}X)}\,\overline{(YZ)}$$



$$Y_T = YZ + \overline{W}\,\overline{X}Z + \overline{W}X\overline{Z} + WXZ + WY\overline{Z}$$
$$= \overline{(YZ)}\,\overline{(\overline{W}\,\overline{X}Z)}\,\overline{(\overline{W}X\overline{Z})}\,\overline{(WXZ)}\,\overline{(W\overline{Y}\,\overline{Z})}$$



$$Z_T = Z + \overline{W}\,\overline{X} + WX + W\overline{X}Y\overline{Z}$$
$$= \overline{Z}.\overline{(\overline{W}\,\overline{X})}.\overline{(WX)}.\overline{(W\overline{X}Y\overline{Z})}$$

Hence, the modified design becomes:

$$W_T = \overline{(\overline{W}\overline{X}\overline{Y}\overline{Z})(WXYZ)(W\overline{X}\overline{Y})(W\overline{X}\overline{Z})}$$

$$X_T = \overline{(\overline{W}\overline{X})(YZ)}$$

$$Y_T = \overline{(YZ)(\overline{W}\overline{X}\overline{Z})(\overline{W}XZ)(WXZ)(W\overline{Y}\overline{Z})}$$

$$Z_T = \overline{Z}.\overline{(\overline{W}\overline{X})}.(WX).\overline{(\overline{W}XYZ)}$$

This design requires a total of 16 NAND gates, 1 five-input NAND gate, 4 four-input NAND gates, 6 three-input NAND gates and 5 two-input NAND gates.

To implement this design using only three-input NAND gates, a total of 26 gates would be required (i.e. 9 *7410 ICs*). *Note, each four-input and five-input NAND gate can be replaced with 3 three-input NAND gates!!*

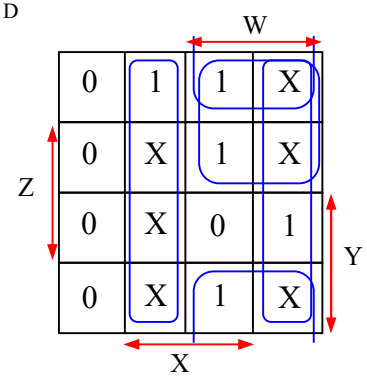
5.

Minterm	WXYZ	Flipflop Inputs			
		W_D	X_D	Y_D	Z_D
0	0 0 0 0	0	0	0	1
1	0 0 0 1	0	0	1	0
2	0 0 1 0	0	0	1	1
3	0 0 1 1	0	1	0	0
4	0 1 0 0	1	0	1	1
11	1 0 1 1	1	1	0	0
12	1 1 0 0	1	1	0	1
13	1 1 0 1	1	1	1	0
14	1 1 1 0	1	1	1	1
15	1 1 1 1	0	0	0	0
0	0 0 0 0				

Operation	D
Stay at 0	0
Stay at 1	1
Go to 0	0
Go to 1	1

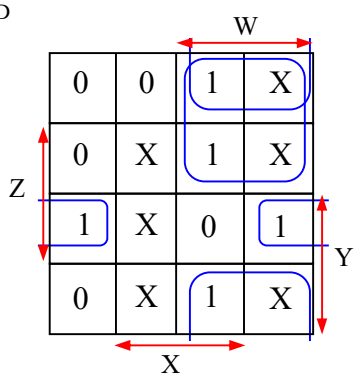
Minterms 5, 6, 7, 8, 9 and 10 can be treated as don't care terms.

W_D



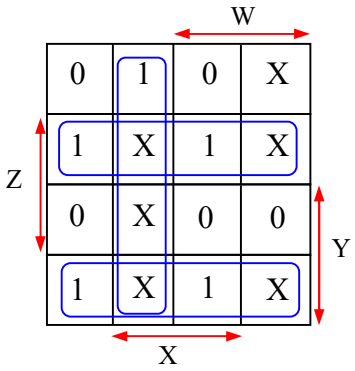
$$W_D = \overline{W}X + W\overline{Y} + W\overline{Z} + W\overline{X}$$
$$= \overline{(\overline{W}X)}\,\overline{(W\overline{Y})}\,\overline{(W\overline{Z})}\,\overline{(W\overline{X})}$$

X_D



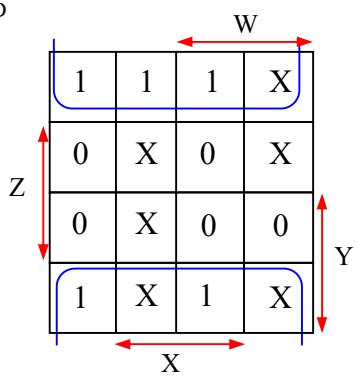
$$X_D = W\overline{Y} + W\overline{Z} + \overline{X}YZ$$
$$= \overline{(W\overline{Y})}\,\overline{(W\overline{Z})}\,\overline{(\overline{X}YZ)}$$

Y_D



$$Y_D = \overline{W}X + Y\overline{Z} + \overline{Y}Z$$
$$= \overline{(\overline{W}X)}\,\overline{(Y\overline{Z})}\,\overline{(\overline{Y}Z)}$$

Z_D



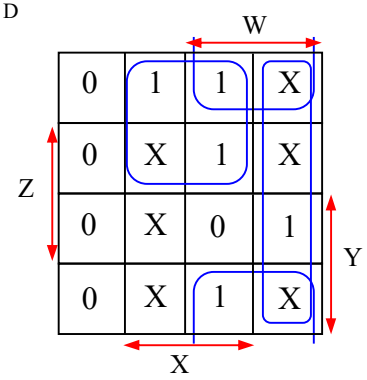
$$Z_D = \overline{Z}$$

This design requires 10 NAND gates, 1 four-input gate, 3 three-input gates and 6 two-input gates. Using three-input gates only, 12 NAND gates are necessary to implement the design. This is equivalent to using **four *7410 ICs***.

Alternative minimisation of W_D as follows:

$$W_D = X\overline{Y} + W\overline{Z} + W\overline{X}$$
$$= \overline{(X\overline{Y})}\,\overline{(W\overline{Z})}\,\overline{(W\overline{X})}$$

W_D



Yields a design that requires a total 11 three-input NAND gates. However, **four *7410 ICs* are still required**.