Assignment 2

Haohu Shen UCID: 30063099
CPSC 359 - Computing Machinery II
Mar 6, 2019

Step 1

State the problem in words:

A 4-bit Gray Code counter counts upwards from decimal 0 to 15 and using the patterns below:

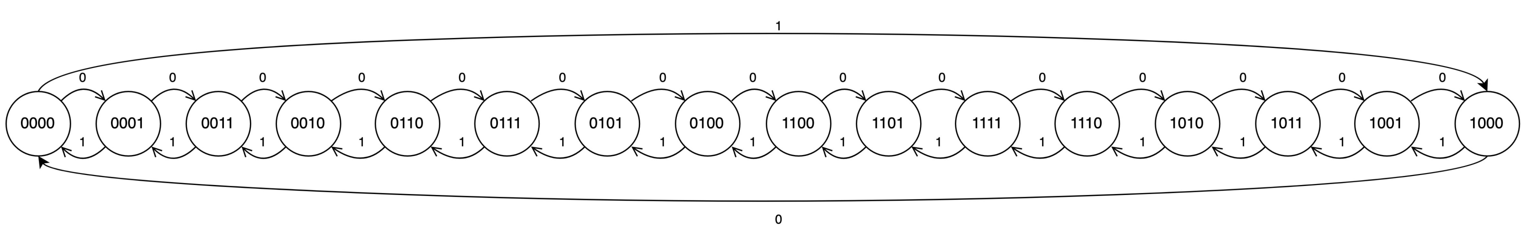
- When the counter reaches decimal 0, it is mapping to bits 0000
- When the counter reaches decimal 1, it is mapping to bits 0001
- When the counter reaches decimal 2, it is mapping to bits 0011
- When the counter reaches decimal 3, it is mapping to bits 0010
- When the counter reaches decimal 4, it is mapping to bits 0110
- When the counter reaches decimal 5, it is mapping to bits 0111
- When the counter reaches decimal 6, it is mapping to bits 0101
- When the counter reaches decimal 7, it is mapping to bits 0100
- When the counter reaches decimal 8, it is mapping to bits 1100
- When the counter reaches decimal 9, it is mapping to bits 1101
- When the counter reaches decimal 10, it is mapping to bits 1111
- When the counter reaches decimal 11, it is mapping to bits 1110
- When the counter reaches decimal 12, it is mapping to bits 1010
- When the counter reaches decimal 13, it is mapping to bits 1011
- When the counter reaches decimal 14, it is mapping to bits 1001
- When the counter reaches decimal 15, it is mapping to bits 1000
- Once 1000 (15 in decimal) is reached, the code “wraps around” to 0000 (0 in decimal)

Design a synchronous sequential logic circuit that implements a 4-bit Gray Code counter such that

- it has one input called **Direction**: if 0, the counter counts up; if 1, the counter counts down
- it has 4 outputs such that each output is corresponding to each bit in the Gray Code
- it uses JK flip-flops

Step 2

Create a state diagram for the design:



Step 3

Determine the inputs, outputs, number of flip-flops needed and their type:

Input

- **Direction** (we can call it **x**)

Output

4 LED lights, we can call them:

- LED **A**
- LED **B**
- LED **C**
- LED **D**

The number of flip-flops

- 4 JK flip-flops

Step 4

Base on the naming convention mentioned in the previous step, we derive the excitation table for the state machine as below:

$A(t)$	$B(t)$	$C(t)$	$D(t)$	x	$A(t+1)$	$B(t+1)$	$C(t+1)$	$D(t+1)$	J_A	K_A	J_B	K_B	J_C	K_C	J_D	K_D
0	0	0	0	0	0	0	0	1	0	x	0	x	0	x	1	x
0	0	0	0	1	1	0	0	0	1	x	0	x	0	x	0	x
0	0	0	1	0	0	0	1	1	0	x	0	x	1	x	x	0
0	0	0	1	1	0	0	0	0	0	x	0	x	0	x	x	1
0	0	1	1	0	0	0	1	0	0	x	0	x	x	0	x	1
0	0	1	1	1	0	0	0	1	0	x	0	x	x	1	x	0
0	0	1	0	0	0	1	1	0	0	x	1	x	x	0	0	x
0	0	1	0	1	0	0	1	1	0	x	0	x	x	0	1	x
0	1	1	0	0	0	1	1	1	0	x	x	0	x	0	1	x
0	1	1	0	1	0	0	1	0	0	x	x	1	x	0	0	x
0	1	1	1	0	0	1	0	1	0	x	x	0	x	1	x	0
0	1	1	1	1	0	1	1	0	0	x	x	0	x	0	x	1
0	1	0	1	0	0	1	0	0	0	x	x	0	0	x	x	1
0	1	0	1	1	0	1	1	1	0	x	x	0	1	x	x	0

0	1	0	0	0	1	1	0	0	1	x	x	0	0	x	0	x
0	1	0	0	1	0	1	0	1	0	x	x	0	0	x	1	x
1	1	0	0	0	1	1	0	1	x	0	x	0	0	x	1	x
1	1	0	0	1	0	1	0	0	x	1	x	0	0	x	0	x
1	1	0	1	0	1	1	1	1	x	0	x	0	1	x	x	0
1	1	0	1	1	1	1	0	0	x	0	x	0	0	x	x	1
1	1	1	1	0	1	1	1	0	x	0	x	0	x	0	x	1
1	1	1	1	1	1	1	0	1	x	0	x	0	x	1	x	0
1	1	1	0	0	1	0	1	0	x	0	x	1	x	0	0	x
1	1	1	0	1	1	1	1	1	x	0	x	0	x	0	1	x
1	0	1	0	0	1	0	1	1	x	0	0	x	x	0	1	x
1	0	1	0	1	1	1	1	0	x	0	1	x	x	0	0	x
1	0	1	1	0	1	0	0	1	x	0	0	x	x	1	x	0
1	0	1	1	1	1	0	1	0	x	0	0	x	x	0	x	1
1	0	0	1	0	1	0	0	0	x	0	0	x	0	x	x	1
1	0	0	1	1	1	0	1	1	x	0	0	x	1	x	x	0
1	0	0	0	0	0	0	0	0	x	1	0	x	0	x	0	x
1	0	0	0	1	1	0	0	1	x	0	0	x	0	x	1	x

Step 5

Derive the circuit output functions and flip-flop input functions, using the map method:

We first derive the circuit output functions using the map method:

- For K-Map of next state of A(that is, $A(t+1)$), we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	0	1	0	0
001	0	0	0	0
011	0	0	0	0
010	0	0	0	0

- When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	0	1	1	1
101	1	1	1	1
111	1	1	1	1
110	1	0	1	1

Thus we have

$$A(t+1) = AD + AC + B'C'D'x + BC'D'x'$$

such that $A = A(t)$, $B = B(t)$, $C = C(t)$, $D = D(t)$

- For K-Map of next state of B(that is, $B(t+1)$), we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	0	0	0	0
001	1	0	0	0
011	1	0	1	1
010	1	1	1	1

  - When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	0	0	0	0
101	0	1	0	0
111	0	1	1	1
110	1	1	1	1

Thus we have

$$B(t+1) = BC' + BD + A'CD'x' + ACD'x$$

such that $A = A(t)$, $B = B(t)$, $C = C(t)$, $D = D(t)$

- For K-Map of next state of C(that is, $C(t+1)$), we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	0	0	0	1
001	1	1	0	1
011	1	1	1	0
010	0	0	1	0

  - When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	0	0	1	0
101	1	1	1	0
111	1	1	0	1
110	0	0	0	1

Thus we have

$$C(t+1) = CD' + A'B'Dx' + A'BDx + AB'Dx + ABDx'$$

such that $A = A(t)$, $B = B(t)$, $C = C(t)$, $D = D(t)$

- For K-Map of next state of D(that is, $D(t+1)$), we have
  - When $A = 0$

$ABC \backslash Dx$	00	01	11	10
000	1	0	0	1
001	0	1	1	0
011	1	0	0	1
010	0	1	1	0

  - When $A = 1$

$ABC \backslash Dx$	00	01	11	10
100	0	1	1	0
101	1	0	0	1
111	0	1	1	0
110	1	0	0	1

Thus we have

$$D(t+1) = A'B'C'x' + A'B'Cx + A'BC'x + A'BCx' + AB'C'x + AB'Cx' + ABC'x' + ABCx$$

such that $A = A(t)$, $B = B(t)$, $C = C(t)$, $D = D(t)$

Now we derive all 4 JK flip-flop input functions using the map method:

- For K-Map of J_A, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	0	1	0	0
001	0	0	0	0
011	0	0	0	0
010	1	0	0	0

- When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	x	x	x	x
101	x	x	x	x
111	x	x	x	x
110	x	x	x	x

Thus, we have

$$J_A = B'C'D'x + BC'D'x'$$

- For K-Map of K_A, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	x	x	x	x
001	x	x	x	x
011	x	x	x	x
010	x	x	x	x

  - When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	1	0	0	0
101	0	0	0	0
111	0	0	0	0
110	0	1	0	0

Thus, we have

$$K_A = B'C'D'x' + BC'D'x$$

- For K-Map of J_B, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	0	0	0	0
001	1	0	0	0
011	x	x	x	x
010	x	x	x	x

  - When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	0	0	0	0
101	0	1	0	0
111	x	x	x	x
110	x	x	x	x

Thus, we have

$$J_B = A'CD'x' + ACD'x$$

- For K-Map of K_B, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	x	x	x	x
001	x	x	x	x
011	0	1	0	0
010	0	0	0	0

- When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	x	x	x	x
101	x	x	x	x
111	1	0	0	0
110	0	0	0	0

Thus, we have

$$K_B = A'CD'x + ACD'x'$$

- For K-Map of J_C, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	0	0	0	1
001	x	x	x	x
011	x	x	x	x
010	0	0	1	0

  - When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	0	0	1	0
101	x	x	x	x
111	x	x	x	x
110	0	0	0	1

Thus, we have

$$J_C = A'B'Dx' + A'BDx + AB'Dx + ABDx'$$

- For K-Map of K_C, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	x	x	x	x
001	0	0	1	0
011	0	0	0	1
010	x	x	x	x

  - When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	x	x	x	x
101	0	0	0	1
111	0	0	1	0
110	x	x	x	x

Thus, we have

$$K_C = A'B'Dx + A'BDx' + AB'Dx' + ABDx$$

- For K-Map of J_D, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	1	0	x	x
001	0	1	x	x
011	1	0	x	x
010	0	1	x	x

- When $A = 1$

$ABC\backslash Dx$	00	01	11	10
100	0	1	x	x
101	1	0	x	x
111	0	1	x	x
110	1	0	x	x

Thus, we have

$$J_D = A'B'C'x' + A'B'Cx + A'BC'x + A'BCx' + AB'C'x + AB'Cx' + ABC'x' + ABCx$$

- For K-Map of K_D, we have
  - When $A = 0$

$ABC\backslash Dx$	00	01	11	10
000	x	x	1	0
001	x	x	0	1
011	x	x	1	0
010	x	x	0	1

  - When $A = 1$

$ABC \backslash Dx$	00	01	11	10
100	x	x	0	1
101	x	x	1	0
111	x	x	0	1
110	x	x	1	0

Thus, we have

$$K_D = A'B'C'x + A'B'Cx' + A'BC'x' + A'BCx + AB'C'x' + AB'Cx + ABC'x + ABCx'$$

Step 6

NOTE The file *assign2.circ* has been uploaded to D2L dropbox. And diagrams for J_A, K_A, J_B, K_B, J_C, J_D and the diagram for the whole circuit are shown below by hand-written: