

ECE212/212s Lecture Notes

Lecture: Chapter 8

Dr. Clay Gloster, Jr.

Word Problems

1. A finite state recognizer has one input (X) and one output (Z). The output is asserted whenever the input sequence 010 has been observed as long as 100 has never been seen.
2. A synchronous counter has a mode control input m . When $m = 0$, the counter steps through the sequence 000,001,010,011,100,101,110,111, and repeats. When $m = 1$, the counter advances through the Gray code sequence 000,100,110,010,011,111,101,001, and repeats.
3. A 4-bit serial lock is used to allow entry to a locked room. The lock has a RESET button, an ENTER button, and a two position switch to represent the key being entered (KEY-W). When the output UNLOCK is asserted, the door will open. The unlock process begins by the user pressing RESET. He/She then sets the INPUT switch followed by pressing ENTER. This is repeated 4 times. An Error light should illuminate if an incorrect code has been entered.

Finite State Machine Design Steps (Page 388)

1. Understand the Problem (try some input sequences)
2. Draw State Diagram.
3. Perform State Minimization (Formal method later)
4. Perform State Assignment
5. Draw State Transition Table (Symbolic)
6. Draw State Transition Table (Encoded)
7. Choose FF type (J-K's tend to reduce gate count , D's are easier)
8. Use excitation tables to fill in K-Maps for FF inputs
9. Perform logic simplification
10. Implement the FSM

Sequential Circuit Design

A particular sequential circuit has 1 input and 1 output. The output is equal to 1 if and only if the past 3 previous inputs were 101. Design the sequential circuit using J-K flip flops.

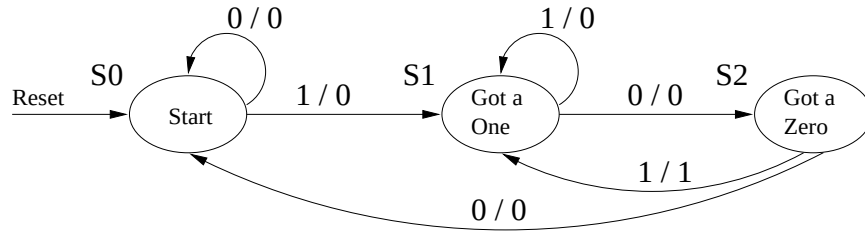


Figure 1: State Diagram

Symbol	Y_2Y_1
S_0	00
S_1	01
S_2	11

State Assignment

State Transition Table: Symbolic from state diagram

	NS A = 0	Z A = 1
S_0	$S_0,0$	$S_1,0$
S_1	$S_2,0$	$S_1,0$
S_2	$S_0,0$	$S_1,1$

State Transition Table (Encoded)

PS Y_2Y_1	NS A = 0 $Y_2^+Y_1^+Z$	Z A = 1 $Y_2^+Y_1^+Z$
00	00,0	01,0
01	11,0	01,0
11	00,0	01,1

PS	NS A = 0	Z A = 1
Y_2Y_1	$J_{Y_2}^+K_{Y_2}^+J_{Y_1}^+K_{Y_1}^+Z$	$J_{Y_2}^+K_{Y_2}^+J_{Y_1}^+K_{Y_1}^+Z$
00	0X0X,0	0X1X,0
01	1XX0,0	0XX0,0
11	X1X1,0	X1X0,1

Q	Q ⁺	R	S	J	K	T	D
0	0	X	0	0	X	0	0
0	1	0	1	1	X	1	1
1	0	1	0	X	1	1	0
1	1	0	X	X	0	0	1

Minimized equations.

$$J_{Y_2}^+ = \overline{A}Y_1 \quad K_{Y_2}^+ = Y_2 \quad Z = AY_2$$

$$J_{Y_1}^+ = A \quad K_{Y_1}^+ = Y_2$$

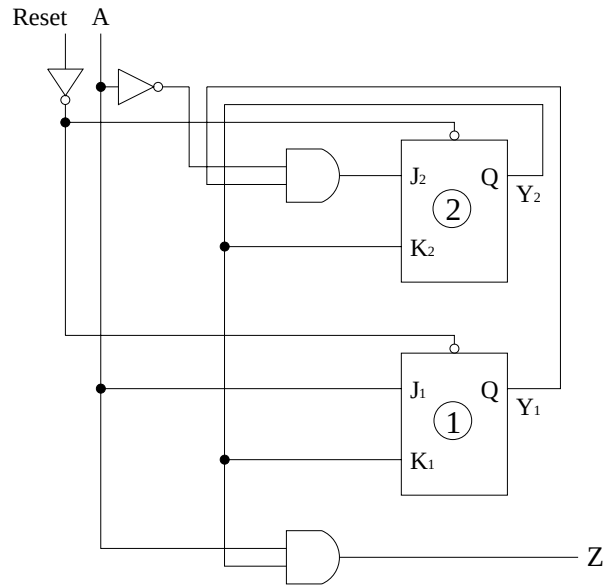
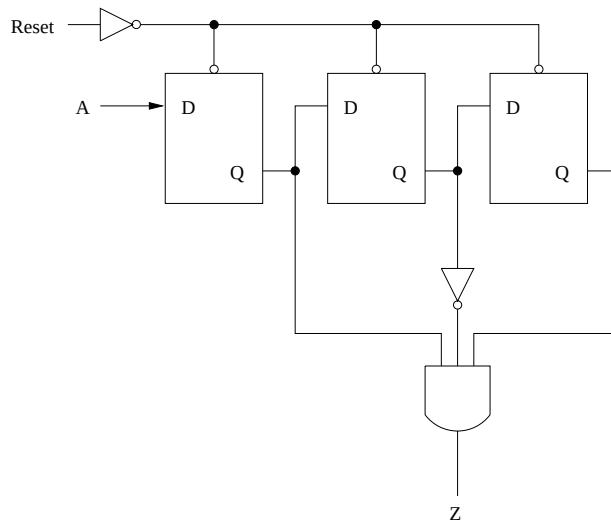
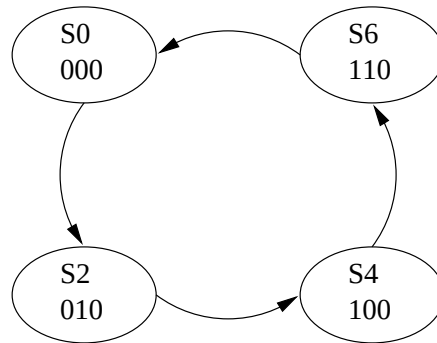
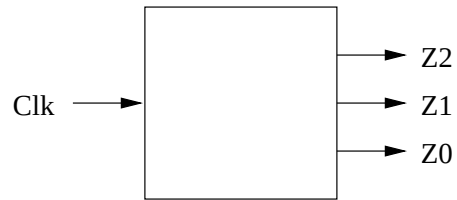


Figure 2: State Diagram

An alternate implementation of the 101 sequence detector.



Example: Design a 3-bit counter that counts 0, 2, 4, 6



Using T FF's

STT (Symbolic)	PS	NS	OUTPUT	State Asigment	$S_0 \Rightarrow 000$
	S0	S2	000		$S_2 \Rightarrow 010$
	S2	S4	010		$S_4 \Rightarrow 100$
	S4	S6	100		$S_6 \Rightarrow 110$
	S6	S0	110		

STT (Encoded)	PS			NS			OUTPUT		
	Y_2	Y_1	Y_0	Y_2^+	Y_1^+	Y_0^+	Z_1	Z_2	Z_3
	0	0	0	0	1	0	0	0	0
	0	1	0	1	0	0	0	1	0
	1	0	0	1	1	0	1	0	0
	1	1	0	0	0	0	1	1	0

TFF's	PS			NS			OUTPUT		
	Y_2	Y_1	Y_0	$T_{Y_2}^+$	$T_{Y_1}^+$	$T_{Y_0}^+$	Z_1	Z_2	Z_3
	0	0	0	0	1	0	0	0	0
	0	1	0	1	1	0	0	1	0
	1	0	0	0	1	0	1	0	0
	1	1	0	1	1	0	1	1	0

$Y_2 Y_1$					
Y_0	0	0	1	1	0
	1	X	X	X	X

$$TY_2^+ = Y_1$$

$Y_2 Y_1$					
Y_0	0	1	1	1	1
	1	X	X	X	X

$$TY_1^+ = 1$$

$Y_2 Y_1$					
Y_0	0	0	0	0	0
	1	X	X	X	X

$$TY_0^+ = 0$$

$$Z_2 = Y_2$$

$$Z_1 = Y_1$$

$$Z_0 = Y_0$$

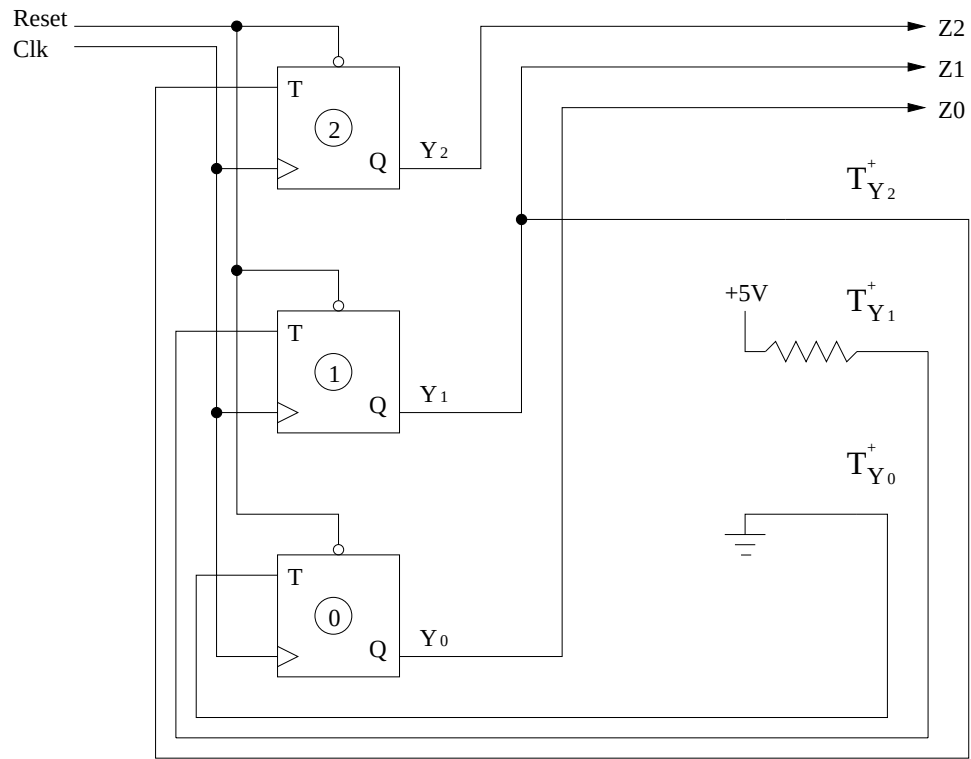
Excitation Table (Chapter 6 $\Rightarrow$ Page 308)

Q(+)	Q(t+1)	R	S	J	K	T	D
0	0	X	0	0	X	0	0
0	1	0	1	1	X	1	1
1	0	1	0	X	1	1	0
1	1	0	X	X	0	0	1

Implement the Counter using D FF's

Y_2	Y_1	Y_0	Y_2^+	Y_1^+	Y_0^+
0	0	0	0	1	0
0	1	0	1	0	0
1	0	0	1	1	0
1	1	0	0	0	0

Y_2	Y_1	Y_0	$D_{Y_2}^+$	$D_{Y_1}^+$	$D_{Y_0}^+$
0	0	0	0	1	0
0	1	0	1	0	0
1	0	0	1	1	0
1	1	0	0	0	0



$Y_2 Y_1$					
Y_0	0	0	1	0	1
1	X	X	X	X	X

$$DY_2^+ = \overline{Y_2} Y_1 + Y_2 \overline{Y_1} = Y_2 \oplus Y_1$$

$Y_2 Y_1$					
Y_0	0	1	0	0	1
1	X	X	X	X	X

$$DY_1^+ = \overline{Y_2}$$

$Y_2 Y_1$					
Y_0	0	0	0	0	0
1	X	X	X	X	X

$$DY_0^+ = 0$$

Design a 2-bit up-counter with enable.

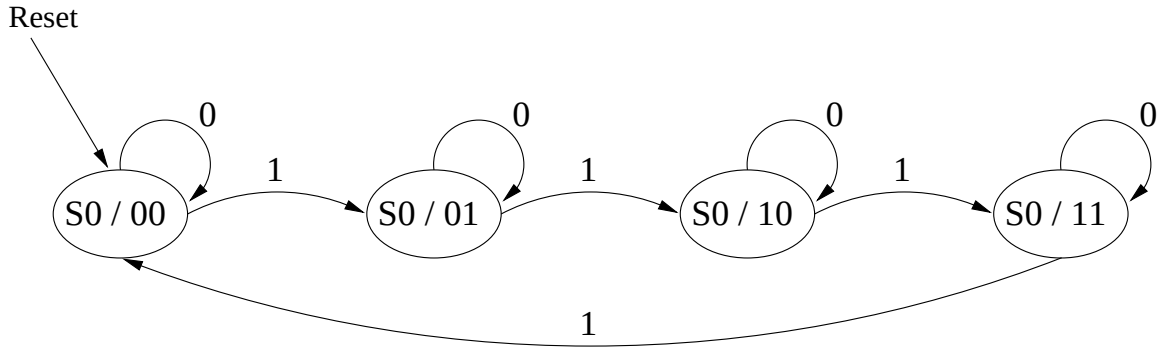


Figure 3: State Diagram (Moore Machine)

State Assignment

	B	A
S_0	0	0
S_1	0	1
S_2	1	0
S_3	1	1

Choose FF type: DFF

State Transition Table (Symbolic)

Inputs INC	PS	NS	Outputs Z_1, Z_0
0	S0	S0	00
1	S0	S1	00
0	S1	S1	01
1	S1	S2	01
0	S2	S2	10
1	S2	S3	10
0	S3	S3	11
1	S3	S0	11

STT (Encoded)

INC	BA	B^+A^+	Z_1Z_0
0	00	00	00
1	00	01	00
0	01	01	01
1	01	10	01
0	10	10	10
1	10	11	10
0	11	11	11
1	11	00	11

INC B

A	00	01	11	10
0	0	1	1	0
1	0	1	0	1

$DB = \overline{A}B + \overline{INC}B + INC\overline{A}$

INC B

A	00	01	11	10
0	0	0	1	1
1	1	1	0	0

$DA = \overline{A}INC + A\overline{INC} = A \oplus INC$

Using TFF

State Assignment		B	A
	S_0	0	0
	S_1	0	1
	S_2	1	0
	S_3	1	1

Choose FF type: TFF

State Transition Table (Symbolic)	Inputs INC	PS	NS	Outputs Z_1, Z_0
	0	S0	S0	00
	1	S0	S1	00
	0	S1	S1	01
	1	S1	S2	01
	0	S2	S2	10
	1	S2	S3	10
	0	S3	S3	11
	1	S3	S0	11

STT (Encoded)	INC	BA	B^+A^+	Z_1Z_0
	0	00	00	00
	1	00	01	00
	0	01	01	01
	1	01	10	01
	0	10	10	10
	1	10	11	10
	0	11	11	11
	1	11	00	11

TFF	INC	BA	B^+A^+	Z_1Z_0
	0	00	00	00
	1	00	01	00
	0	01	00	01
	1	01	11	01
	0	10	00	10
	1	10	01	10
	0	11	00	11
	1	11	11	11

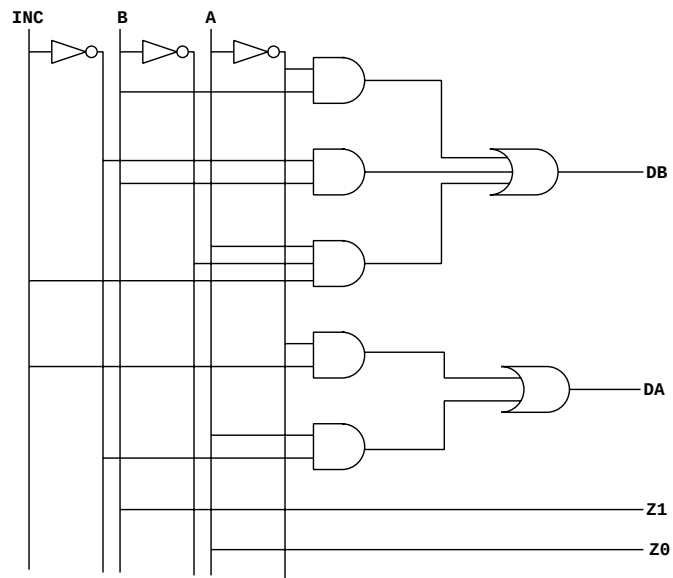
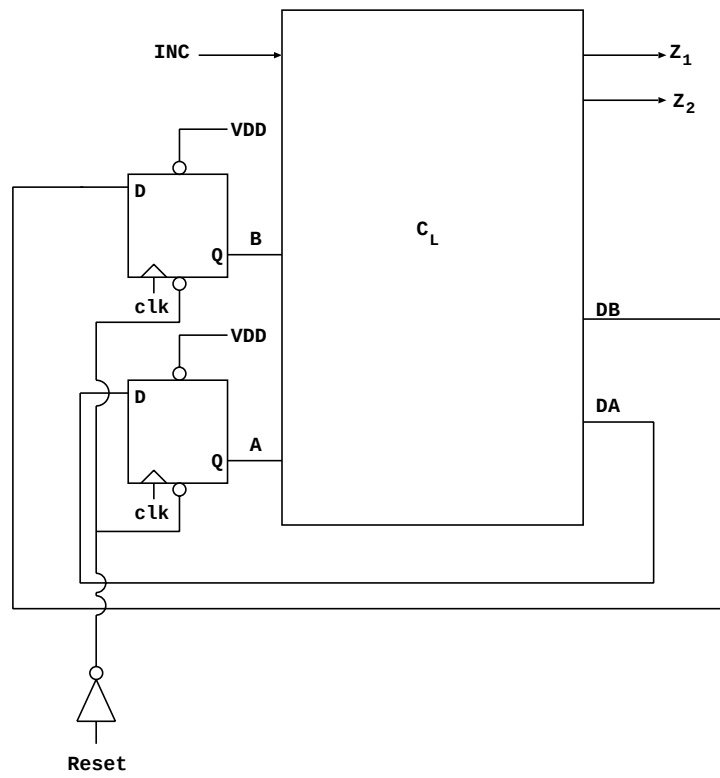


Figure 4: CL for 2-bit up counter with enable

Minimized Equations

$$B = INCY_0$$

$$A = INC$$

$$Z_1 = Y_1$$

$$Z_0 = Y_0$$

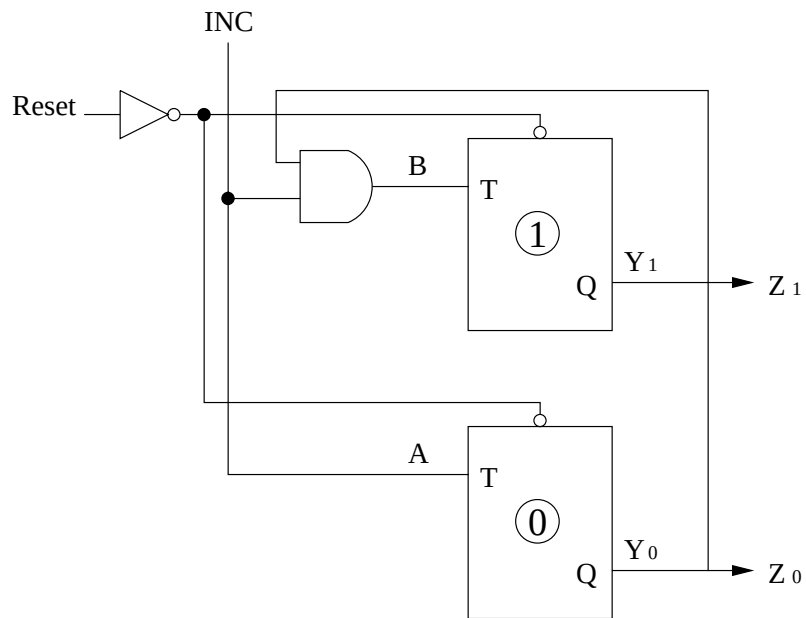


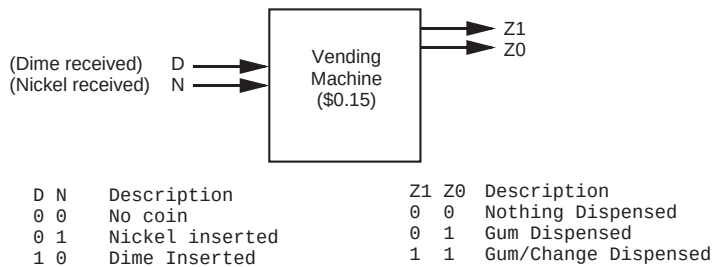
Figure 5: 2-bit up counter with enable using TFFs

Vending Machine Example

You are to design a vending machine. The vending machine accepts nickels and dimes only and only contains one product gum that costs 0.15. Since the machine takes only nickels and dimes, it has two inputs (N and D). The machine can either dispense nothing, dispense gum, or dispense gum and 0.05 change. Design the vending machine using J-K flip-flops. Use state minimization to reduce the number of states and use the state assignment heuristics presented in class to justify your state assignment.

The block diagram and state diagram of the vending machine is shown in Figure 6.

Block Diagram:



State Diagram: Arcs are labeled Inputs/Outputs => D,N / Z1,Z0

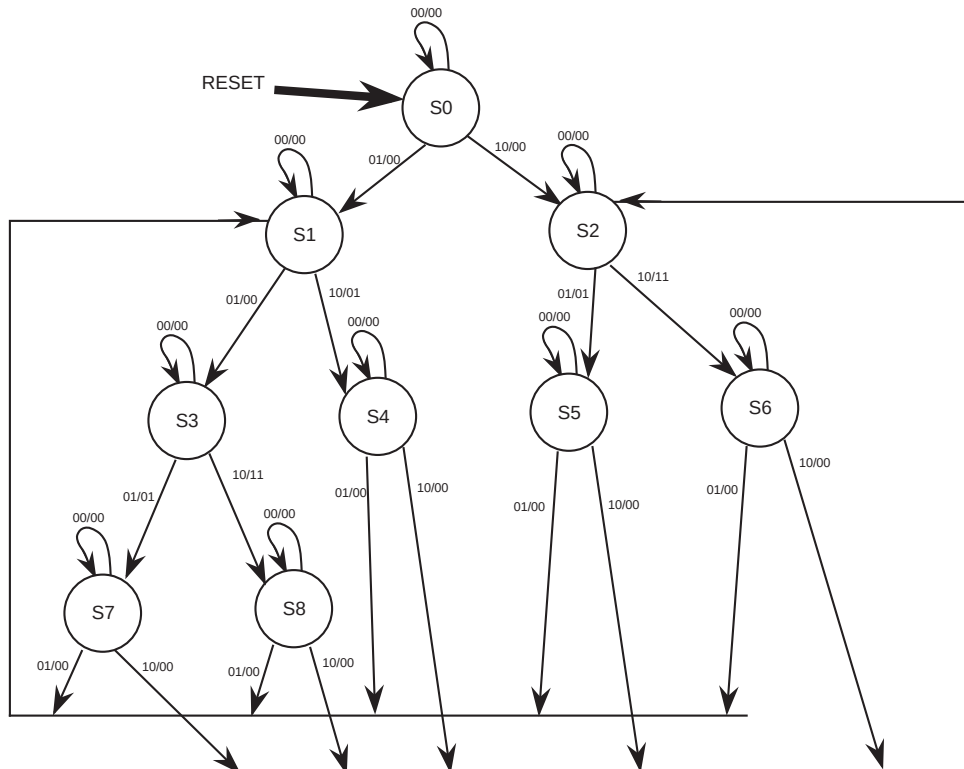


Figure 6: State Diagram for the Simple Vending Machine.

S1								
S2								
S3								
S4								
S5								
S6								
S7								
S8								
	S0	S1	S2	S3	S4	S5	S6	S7

Figure 7: Implication Table for the Simple Vending Machine.

The corresponding state table is given by Table 1.

Table 1: State Table for the Simple Vending Machine.

PS	NS, Z_1 , Z_0		
	DN=01	DN = 10	DN = 00
S0	S1,00	S2,00	S0,00
S1	S3,00	S4,01	S1,00
S2	S5,01	S6,11	S2,00
S3	S7,01	S8,11	S3,00
S4	S1,00	S2,00	S4,00
S5	S1,00	S2,00	S5,00
S6	S1,00	S2,00	S6,00
S7	S1,00	S2,00	S7,00
S8	S1,00	S2,00	S8,00

We use the implementation table method for state minimization which results in the following final implication table. The table is shown in Figure 7. We see from the table that states (S0,S4,S5,S6,S7,S8) are all equivalent and will be replaced by the state S0'. Also, states S2 and S3 are equivalent (to be replaced with S2').

The resulting state table after state minimization is shown in Table 2.

Next we must perform state assignment using the guidelines that were presented in chapter 9 of the text. To encode 3 states, S0', S1, and S2' using the minimum number of state bits or flip-flops, we requires 2 state bits.

We select S0' as the reset state, therefore, S0 is assigned the value AB=00.

The high priority heuristics says that S1 and S2' should be adjacent, since they both have the same next state (S0') under the input DN=10.

The medium priority heuristic says that s1 and s2' should be adjacent, since they are both the next state of the state S0'. It also says that S0' and S1 should be adjacent since they have the same output($Z_1Z_0=00$) under the same input condition DN=01. We will not enumerate any more potential adjacency pairs since we have enumerated all of the possible pairs.

Therefore, attempting to meet the above constraints in priority order, we selected the state assignment:

Using the previous state assignment, the encoded state transition table is as shown in Table 4.

Since the problem states that we must implement this vending machine using J-K flip-flops, then we must use the excitation tables for J-K flip-flops to obtain the excitation table for the vending machine shown in Table 5.

Using K-Maps, we get,

$$\begin{aligned}
J_A(D, N, A, B) &= \sum m(5, 8) + d(2, 3, 6, 7, 10, 11, 12, 13, 14, 15) = D\overline{B} + NB \\
K_A(D, N, A, B) &= \sum m(7, 11) + d(0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13, 14, 15) = D + N \\
J_B(D, N, A, B) &= \sum m(4, 8) + d(1, 2, 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15) = D + N \\
K_B(D, N, A, B) &= \sum m(7, 9, 11) + d(0, 2, 4, 6, 8, 10, 12, 13, 14, 15) = D + NA \\
Z_1(D, N, A, B) &= \sum m(11) + d(2, 6, 9, 12, 13, 14, 15) = DA \\
Z_0(D, N, A, B) &= \sum m(7, 9, 11) + d(2, 6, 9, 12, 13, 14, 15) = DB + NA
\end{aligned}$$

Table 2: Reduced State Table for the Simple Vending Machine .

PS	NS, Z_1, Z_0		
	DN=01	DN = 10	DN = 00
S0'	S1,00	S2',00	S0',00
S1	S2'00	S0'01	S1,00
S2'	S0',01	S0',11	S2',00

Table 3: State Assignment for the Simple Vending Machine .

State	AB
S0'	00
S1	01
S2'	11

Table 4: Encoded State Table for the Simple Vending Machine

PS	NS, Z_1, Z_0		
	DN=01	DN = 10	DN = 00
AB	A^+B^+, Z_1Z_0	A^+B^+, Z_1Z_0	A^+B^+, Z_1Z_0
00	01,00	11,00	00,00
01	11,00	00,01	01,00
11	00,01	00,11	11,00

Table 5: Excitation Table for the Simple Vending Machine using J-K flip-flops.

PS	NS, Z_1, Z_0		
	DN=01	DN = 10	DN = 00
AB	$J_A K_A J_B K_B, Z_1 Z_0$	$J_A K_A J_B K_B, Z_1 Z_0$	$J_A K_A J_B K_B, Z_1 Z_0$
00	0X1X,00	1X1X,00	0X0X,00
01	1XX0,00	0XX1,01	0XX0,00
11	X1X1,01	X1X1,11	X0X0,00