Vending Machine Controller

By

Marina Abdalkarim

Matt Alecci

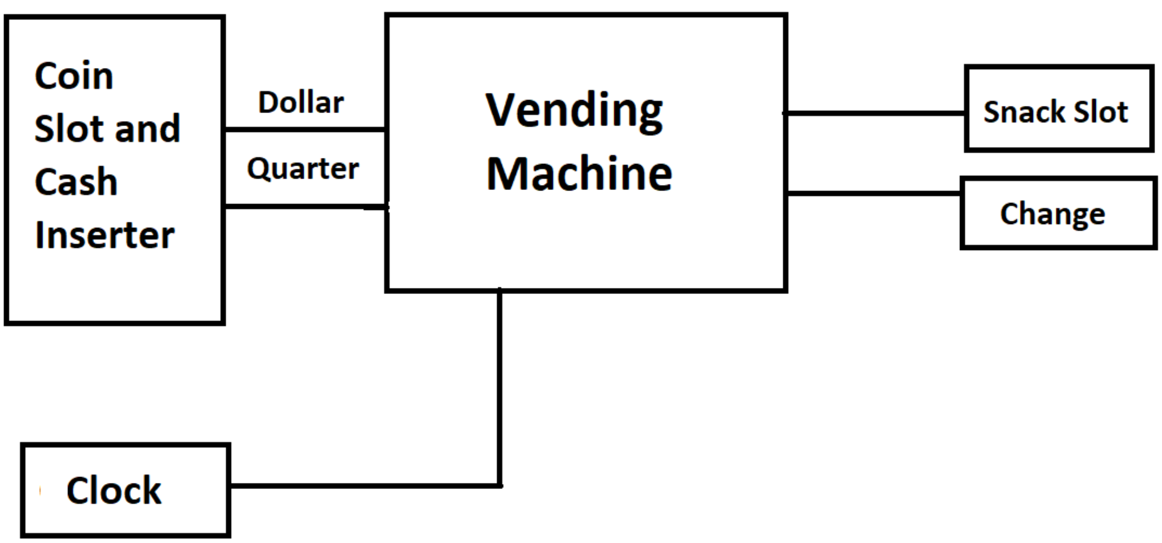
Introduction

A vending machine is a device used in many public places as a means to satisfy a relatively hungry patron. Once encountered, a customer will peer through the transparent glass into the machine, and glance at the choices that lie within. Once he finds a choice that appears the most appetizing, he looks at the price of said item and inserts the required amount of currency to retrieve it. The item then falls into the slotted area to be picked up and enjoyed by the customer.

Specification

The circuit that we have currently designed has 3 main components, the coin slot/dollar insertion, the vending machine, and the snack and change dispenser. The chips inside the machine cost 75¢, while the chocolate costs $1.00.

Block Diagram:



- Coin Slot
    - The coin slot is displayed as four choices: the first quarter (Quarter1), the second quarter (Quarter2), the third quarter (Quarter3), and the dollar (Dollar). They are implemented as binary circuits and in order to insert one you must switch it on, click the 1. In order to retrieve the chips you must select all three quarters (it will not work if you do not select all of them). Another method would be to instead just select the dollar switch (will not work if you have the quarters selected). For the chocolate you can only choose the dollar option.
- Vending Machine
    - This is where the computations are done. There are two D flips flops: A and B, as well as three inputs: C (chocolate/chips), X (quarters), and Y (dollar). This results in three outputs: W (chips), T (chocolate), and Z (change). A series of adders collects the numbers of the selected currency which once passes through the D flip flops. There is a binary switch that serves as the user's choice between two options: chips or chocolate. If you select "1" you choose chips, if you select "0", you choose chocolate. Then they all descend to the bottom where the snack and change dispenser is and the machine uses AND circuits to determine whether or not you've entered 3 quarters and a dollar and whether you've chosen chips or chocolate.
- Change/Snack

- ➢ Once you’ve selected three quarters and a choice of snack, the chips binary probe will display a “1” to inform you they’ve landed but there will be a zero in the change area since you entered the exact amount. If you however choose a dollar instead, the chips area will again display a “1” as well as the change to inform you, you’ve received a quarter in return. If you’ve chosen chocolate, a “1” will be displayed where chocolate is and “0” for change since it is the exact amount.

Comparison

While our design does perform the basic functions any vending machine should, such as displaying change, choosing between different options and allowing realistic coin options at reasonable prices, it also has a touch of added realism by not allowing you to choose more than one item at a time. As in, if one item is selected, you cannot enter coins for the other and expect their output to show up. That would render the whole choice of item aspect pointless. Therefore, since ours seems to be closer to the inner machinations of a real vending machine, we would say it is better by comparison.

Design Details

State Table

	Present State			Input		Next State		Output		
	A	B	C	X	Y	A	B	W	T	Z
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	1
2	0	0	0	1	0	0	1	0	0	0
4	0	0	1	0	0	0	0	0	0	0
5	0	0	1	0	1	0	0	1	1	0

6	0	0	1	1	0	0	1	0	0	0
8	0	1	0	0	0	0	1	0	0	0
9	0	1	0	0	1	0	1	0	0	1
10	0	1	0	1	0	1	0	0	0	0
12	0	1	1	0	0	0	1	0	0	0
13	0	1	1	0	1	0	1	0	0	0
14	0	1	1	1	0	1	0	0	0	0
16	1	0	0	0	0	0	0	0	0	0
17	1	0	0	0	1	0	0	0	0	1
18	1	0	0	1	0	1	0	0	0	0
20	1	0	1	0	0	0	0	0	0	0
21	1	0	1	0	1	0	0	0	0	0
22	1	0	1	1	0	1	0	0	1	0
24	1	1	0	0	0	0	1	0	0	0
25	1	1	0	0	1	0	1	0	0	1
26	1	1	0	1	0	1	0	0	0	0
28	1	1	1	0	0	0	1	0	0	0
29	1	1	1	0	1	0	1	0	0	0
30	1	1	1	1	0	1	0	0	1	0

K-maps

A’

bc\xy	00	01	11	10
00			X	
01			X	
11			X	1
10			X	1

A

bc\xy	00	01	11	10
00			X	1
01			X	1
11			X	1
10			X	1

Da = BX + AX

A’

bc\xy	00	01	11	10
00			X	1
01			X	1
11	1	1	X	
10	1	1	X	

A

bc\xy	00	01	11	10
00			X	
01			X	

11	1	1	X	
10	1	1	X	

Db = BX’ + A’B’X

A’

bc\xy	00	01	11	10
00			X	
01		1	X	
11			X	
10			X	

A

bc\xy	00	01	11	10
00			X	
01			X	
11			X	
10			X	

Z = A’B’CY

A’

bc\xy	00	01	11	10
00			X	
01		1	X	
11			X	
10			X	

A

bc\xy	00	01	11	10

00			X	
01			X	1
11			X	1
10			X	

W = A'B'CY

A'

bc\xy	00	01	11	10
00		1	X	
01			X	
11			X	
10		1	X	

A

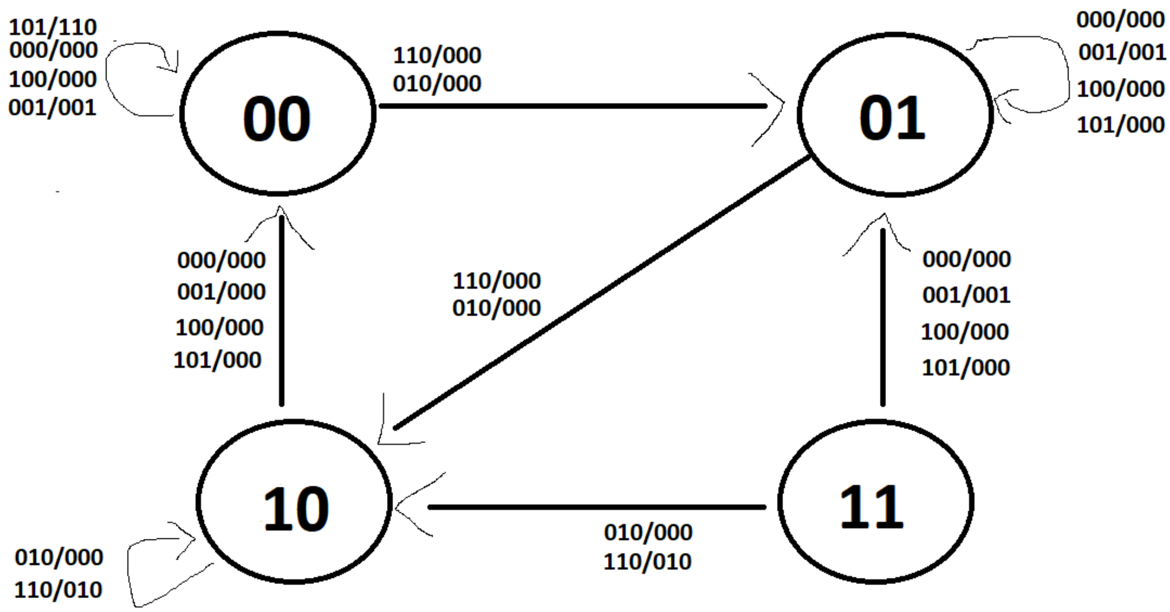
bc\xy	00	01	11	10
00		1	X	
01			X	
11			X	
10		1	X	

T = YC'

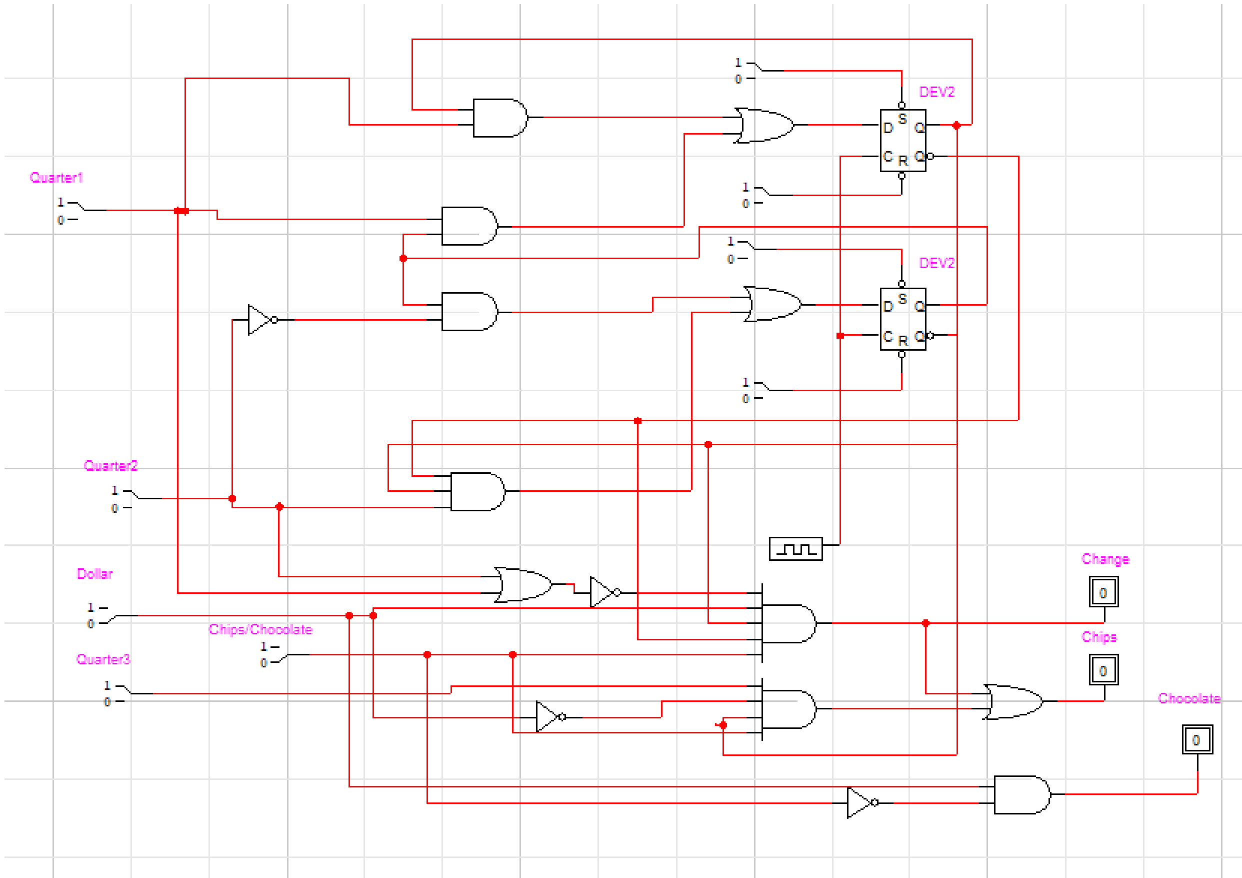
Verification

Present State AB	Use Input Equation	Next State AB
00	Da = BX + AX = 0 Db = BX' + A'B'X = 1	01
00	Da = BX + AX = 0 Db = BX' + A'B'X = 1	01
01	Da = BX + AX = 1 Db = BX' + A'B'X = 0	10
01	Da = BX + AX = 1 Db = BX' + A'B'X = 0	10
10	Da = BX + AX = 1 Db = BX' + A'B'X = 0	10
10	Da = BX + AX = 1 Db = BX' + A'B'X = 0	10
11	Da = BX + AX = 1 Db = BX' + A'B'X = 0	10
11	Da = BX + AX = 1 Db = BX' + A'B'X = 0	10

State Diagram



Circuit



DEV2
DEV2
Quarter1
Quarter2
Dollar
Chips/Chocolate
Quarter3
Change
Chips
Chocolate