AUTOMATIC VENDING MACHINE

PROJECT REPORT

Submitted for CAL in B.Tech Digital Logic and Design (CSE1003)

By

AYUSH THADA (16BCE1333)
TANAY AGARWAL (16BCE1324)
SACHET SUNIL ZODE (16BCE1002)
RAHUL ANOOP KELKAR (16BCE1380)
RAJAT MAHESHWARI (16BCE1001)

Slot: F1

NITHYADARISINI P.S.

(SCHOOL OF COMPUTER SCIENCE AND ENGINEERING)

VIT UNIVERSITY
(Estd. u/s 3 of UGC Act 1956)
VELLORE ■ CHENNAI
www.vit.ac.in

April, 2017

CERTIFICATE

This is to certify that the Project work entitled "**Automatic Vending Machine**" that is being submitted by " Ayush Thada, Tanay Agarwal, Sachet Sunil Zode, Rahul AnoopKelkar and Rajat Maheshwarifor CAL in B. Tech/MIS Digital Logic and Design is a record of bonafide work done under my supervision. The contents of this Project work, in full or in parts, are not submitted for any other CAL course.

Place: Chennai
Date: April 5, 2017

Group Members:

Ayush Thada (16bce1333)

Tanay agarwal (16bce1324)

Sachet Sunil Zode (16bce1002)

Rahul Anoop Kelkar (16bce1380)

Rajat Maheshwari (16bce1001)

Submitted To:

Nithyadarisini P.S

ACKNOWLEDGEMENT

I would firstly like to thank my course instructor,**Nithyadarisini P.S** who gave me his extremely helpful inputs about my project on "AUTOMATIC VENDING MACHINE". Without her help we could not have proceeded with this project. This project was collaborated work of all the group members of our project. Hard work and dedication of our team is responsible for completion of this project which is really a challenging task for all of us.

ABSTRACT

A finite-state machine or finite-state automaton, finite automaton, or simply a state machine, is a mathematical model of computationIt is an abstract machine that can be in exactly one of a finite number of *states* at any given time. The FSM can change from one state to another in response to some external inputs; the change from one state to another is called a *transition*. A FSM is defined by a list of its states, its initial state, and the conditions for each transition.The behavior of state machines can be observed in many devices in modern society that perform a predetermined sequence of actions depending on a sequence of events with which they are presented.

In our project we intend to design one finite a state machine i.e. vending machines, which dispense products when the proper combination of coins is deposited. In this project we will design logic of vending machine and implement it through a digital circuit.

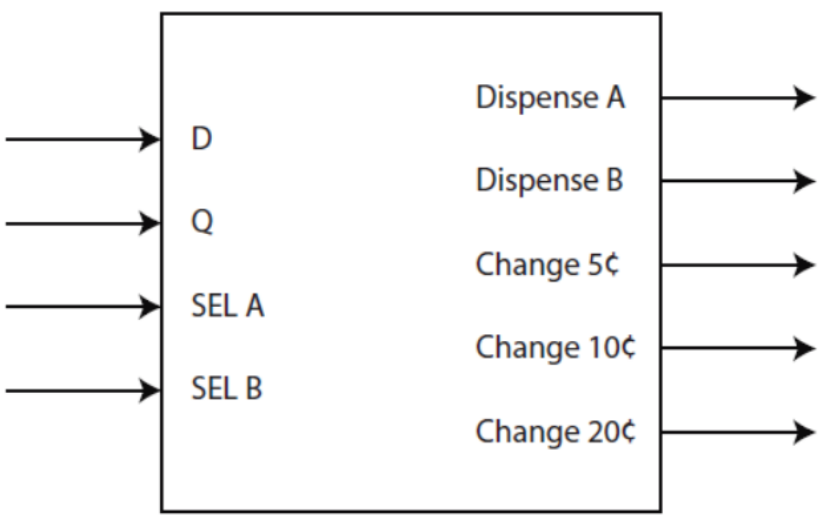
OBJECTIVE

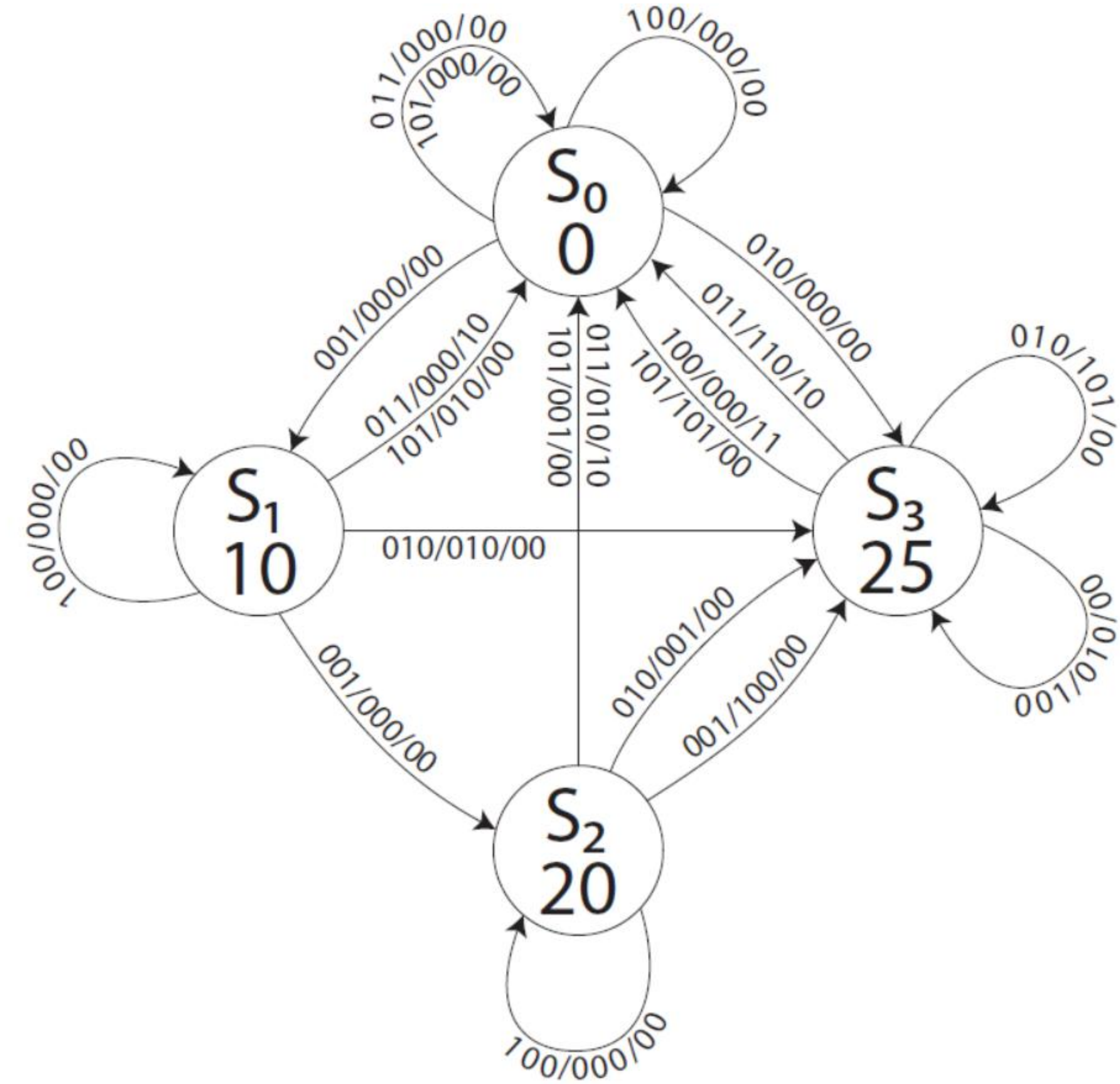
- To design a digital logic for vending machine, hence implement the circuits to design circuit of an automatic time machine.
- To demonstrate the working of an automatic vending machine.

DESIGN PROCESS

A circuit was designed that would implement the function of a vending machine controller. The machine was designed to accept 10 and 25 coins. In addition, the machine dispensed items priced at 10 (item A) and 25 (item B), when the respective amount of money was inputted. The machine accepted coins up to the price of the most expensive item (25). In other words, if the user inputted 30 the machine returned 5 in change, retained 25, and waited for the user to perform an action (select A, select).If the user deposited a total of 25 and then deposited another coin, the coins were immediately returned as change.

Figure 1: Inputs and outputs of vending machine controller





The vending machine controller was designed as a Mealy machine. In other words, the outputs of the machine depended on the inputs. The vending machine controller was first implemented in a

Figure 2: Finite state machine for the vending machine controller

(Format: input/change/dispense)

State Assignment			
S0	0¢	0	0
S1	10¢	0	1
S2	20¢	1	1
S3	25¢	1	0

Item Assignment		
A	1	0
B	1	1

Change Assignment			
0¢	0	0	0
5¢	1	0	0
10¢	0	1	0
15¢	1	1	0
20¢	0	0	1
25¢	1	0	1

Input Encoding			
Dime	0	0	1
Quarter	0	1	0
Select A	0	1	1
Select B	1	0	0
Request Change	1	0	1

With the use of the state diagram, the state assignments, change assignments, item assignments, and input encoding, the following tables were constructed:

State Transition Table															
State Q1 Q0		Inputs X_1 X_2 X_3													
		001		010		011		100		101		110		111	
0	0	0	1	1	0	0	0	0	0	0	0	d	d	d	d
0	1	1	1	1	0	0	0	0	1	0	0	d	d	d	d
1	1	1	0	1	0	0	0	1	1	0	0	d	d	d	d
1	0	1	0	1	0	0	0	0	0	0	0	d	d	d	d

The state transition table demonstrates the transition of states based on the user input. For example, if the machine is at state 01 (contains 10) and the user decides to input another 10 (input 001), the machine will now move to state 11 (containing20). The first column of each block represents Q_1^*, while the second column represents Q_0^*.

Change Table																						
State		Inputs X_1 X_2 X_3																				
Q1 Q0		001			010			011			100			101			110			111		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	d	d	d	d	d	d
0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	d	d	d	d	d	d
1	1	1	0	0	0	0	1	0	1	0	0	0	0	0	0	1	d	d	d	d	d	d
1	0	0	1	0	1	0	1	1	1	0	0	0	0	1	0	1	d	d	d	d	d	d

The change table demonstrates the amount of change to be dispensed based on the input. For example, if the machine contains 25 (state 10), and the user selects item A to be dispensed (input 011), the machine will return 15 in change (output 110).The first column of each block Represents C_1, the second column represents C_2, and the third column represents C_3.

Dispense Table															
State		Inputs X_1 X_2 X_3													
Q1 Q0		001		010		011		100		101		110		111	
0	0	0	0	0	0	0	0	0	0	0	0	d	d	d	d
0	1	0	0	0	0	1	0	0	0	0	0	d	d	d	d
1	1	0	0	0	0	1	0	0	0	0	0	d	d	d	d
1	0	0	0	0	0	1	0	0	1	0	0	d	d	d	d

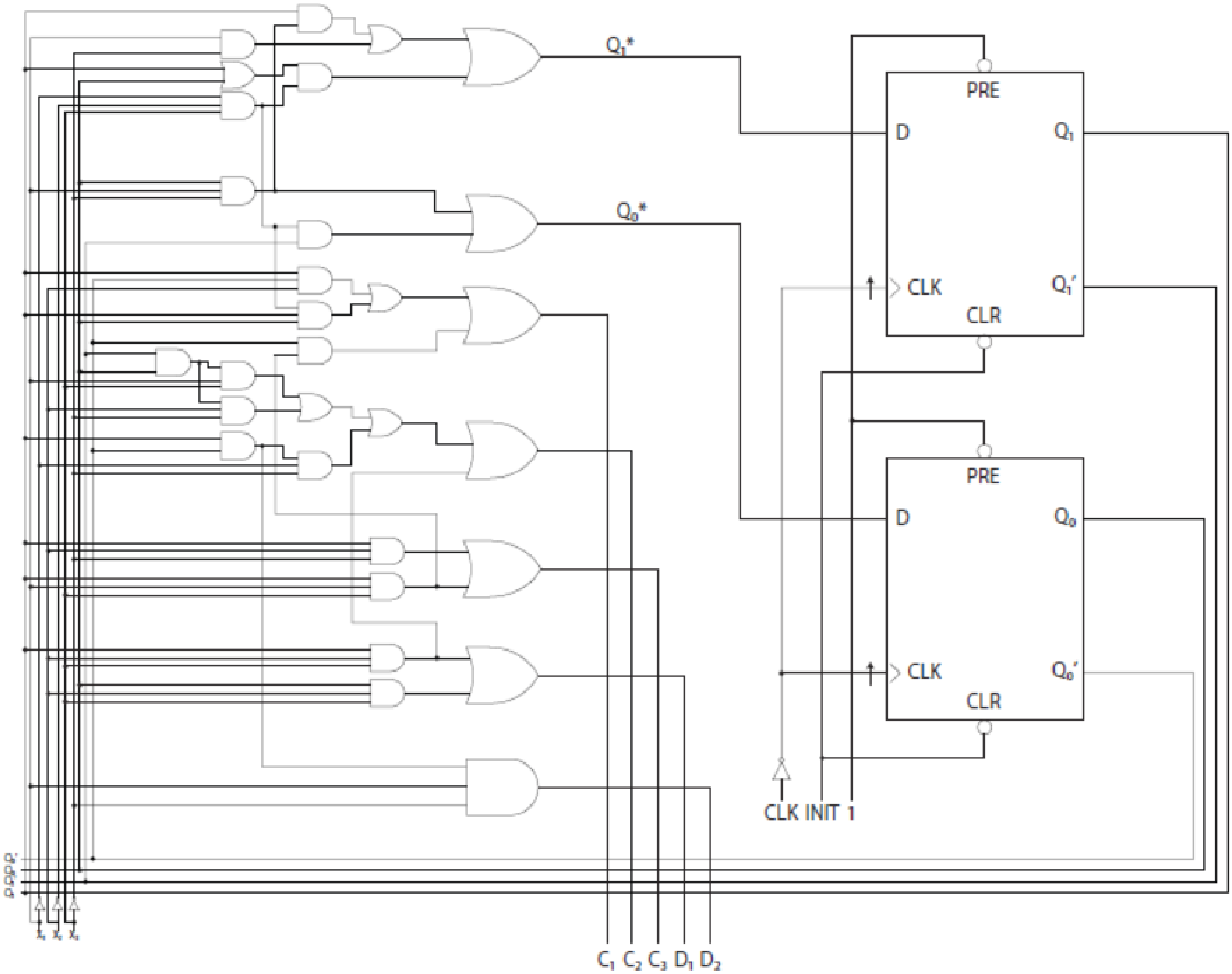
The dispense table demonstrates the item being dispense based on the input. For example, if the machine contains 10(state 01), and the user selects item A to be dispensed (input 011), the machine will dispense item A (output 10). The first column of each block represents D_1, while the second column represents D_2.

After the tables were carefully analyzed,a softwareprogramcalledEspressowasusedtominimizethelogicanddeterminethefollowingequations :

$$Q_1^* = Q_1Q_0x_1x_3^{'} + x_2x_3^{'} + Q_1x_1^{'}x_2^{'}x_3 + Q_0x_1^{'}x_2^{'}x_3$$
$$Q_0^* = Q_0x_1x_3^{'} + Q_1^{'}x_1^{'}x_2^{'}x_3$$
$$C_1 = Q_1Q_0^{'}x_2 + Q_1Q_0x_1^{'}x_2^{'} + Q_1Q_0x_1x_3$$
$$C_2 = Q_1Q_0x_1x_3 + Q_1^{'}Q_0x_2x_3^{'} + Q_1Q_0^{'}x_1^{'}x_3 + Q_1x_2x_3$$
$$C_3 = Q_1x_2x_3^{'} + Q_1x_1x_3$$
$$D_1 = Q_1x_2x_3 + Q_0x_2x_3$$
$$D_2 = Q_1Q_0^{'}x_1x_3^{'}$$

IMPLEMENTATION

The equations listed above were used to determine the hardware needed to successfully create a fully functioning vending machine dispenser. Logic gates such as AND ,OR, Inverters, and D- Flip Flops were used to create the vending machine controller on a breadboard. Due to the complex nature of the assignment, the hardware implementation proved to be far more difficult than previously thought .Often, many wires were inserted in to the wrong terminals on the breadboard. In addition, the clock (a push button) was implemented in correctly at first, and thus the machine was not able to change states. The circuit was assembled and reassembled several times in order to remove all bugs. Figure 3 depicts the circuit diagram of the vending machine controller.



U1	7474		D-Flip Flops
U2	7411	3-Input	AND
U3	7408	2-Input	AND
U4	7432	2-Input	OR
U5	7404		INV
U6	7414		SCHMIDT INV

Figure3: Circuit diagram and list of types of chips used

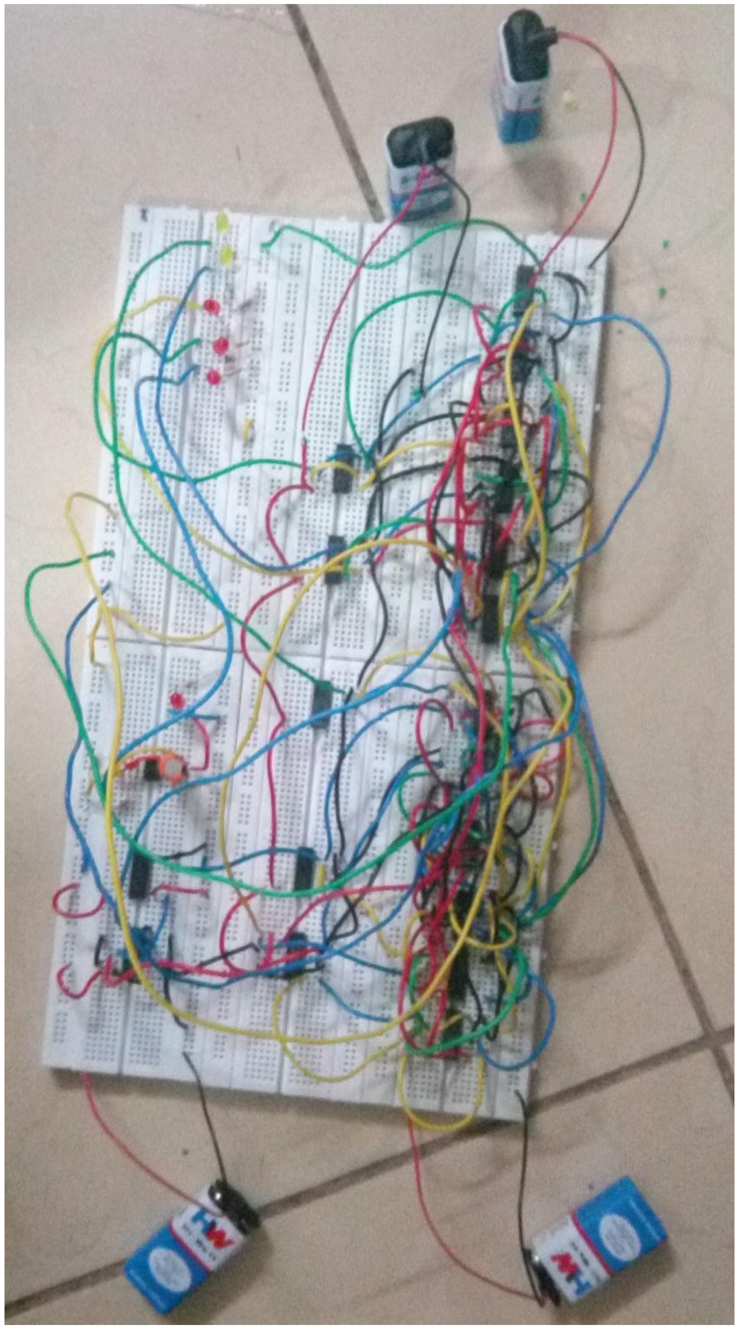


Figure: Implemented Circuit

RESULT

The vending machine controller performed exact to its design. The controller changed states, gave change, and dispensed items appropriately. Each input/output case was carefully tested and deemed to be in working condition.

CONCLUSION

In the end this was a very educational assignment .The project provided a small insight to how to designing a real-world machine is like. This project was very fascinating because it allowed students to design and implement the entire circuit from scratch. It was an appropriate end-of-course project because it required knowledge in all the topics taught in class (such as logic gates, logic minimization, combinational and sequential logic, and state diagrams etc.) .

BIBILOGRAPHY

https://electronics.stackexchange.com/questions/64838/simple-vending-machine-circuit

https://www.slideshare.net/sosososos39794/vending-machine-24580255

https://courses.cs.washington.edu/courses/cse370/09wi/LectureSlides/23-Encoding.pdf

http://www.electronics-tutorials.ws/waveforms/555_timer.html

https://www.youtube.com/watch?v=Z4Zz7n-Lj0g&t=59s