Design assignment on

Chocolate Vending Machine

In partial fulfilment of the course

Microprocessors and Interfacing

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE

K. K BIRLA GOA CAMPUS

PROBLEM STATEMENT No. : 19

BATCH : B191

Submitted by:

1: R. Srinath (2012B5A7560G)

2: S. Bhaargav (2013A7PS006G)

3: Sudhindra Sajjalguddam (2013A7PS065G)

4: Arpit Pandey (2013A7PS075G)

Problem Statement

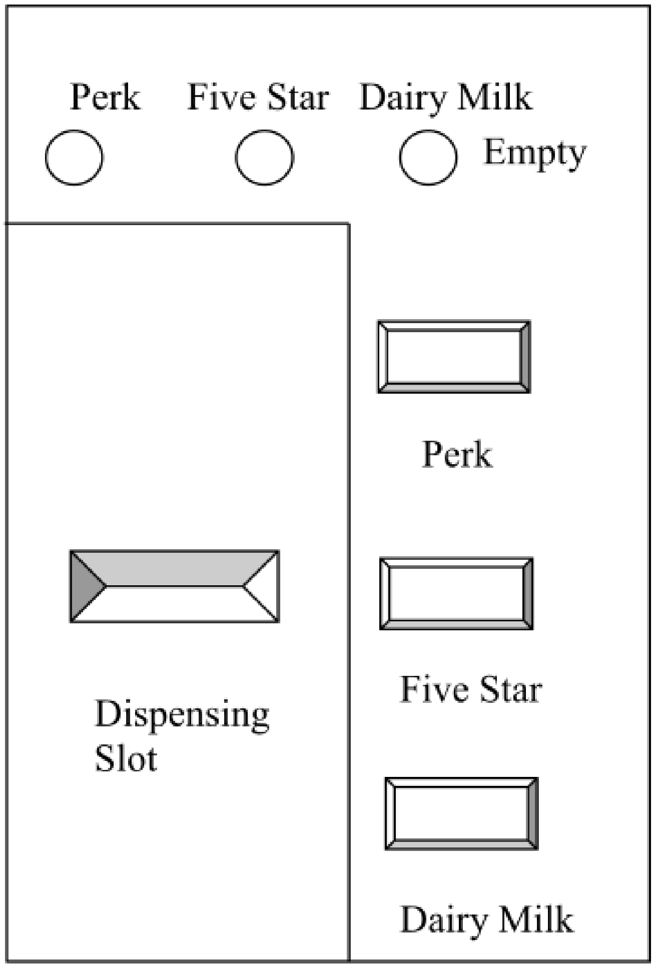
System to be designed: *Chocolate Vending Machine*

Description: This automatic machine vends three different kinds of chocolates.

- Perk: Rs. 5.00
- Five-Star: Rs. 10.00
- Dairy Milk: Rs. 20.00

The currency has to be given in terms of 5 rupee coins. A weight sensor is used to detect whether the coin is a Rs. 5 coin or not. There are three buttons available for the selection of the chocolate. After the chocolate has been selected, the user has to put the correct currency in to the coin slot. When the user has dropped the entire amount into the slot, the machine dispenses the correct chocolate.

LED's are used as indicators to show if any of the chocolates being vended are not available.



Components Required

Item	Number	Description
RAM 6116	2	Random Access Memory
ROM 2732	2	Read Only Memory

Item	Number	Description
8086	1	Microprocessor
8255	1	Programmable Peripheral Interface

Item	Number	Description
74LS373	3	Latching the address lines of 8086.
74LS245	2	Buffering the data lines of 8086.
74LS138	2	Decoder for memory interfacing.

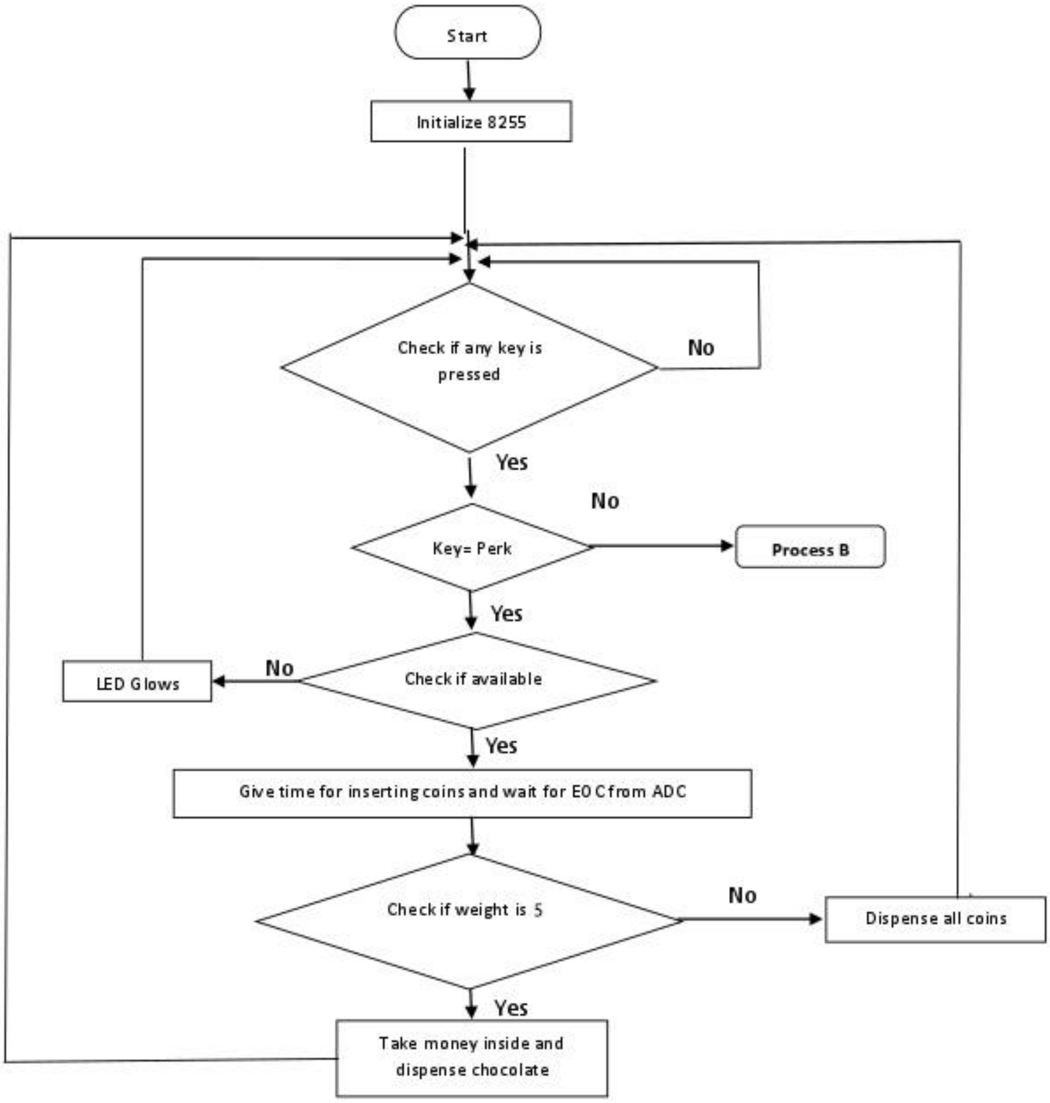
Item	Number	Description
LMD293D	2	Drivers for running motors.
ADC0808	1	For converting analog voltage shown by weight sensor to digital value.

Item	Number	Description
12V DC Motor	4	For operating dispensing slots.
Switches	3	For taking in input from user.
LED	3	To display if a particular chocolate is unavailable.
Weight Sensor (Potentiometer)	1	Verifying the weight of the coins.

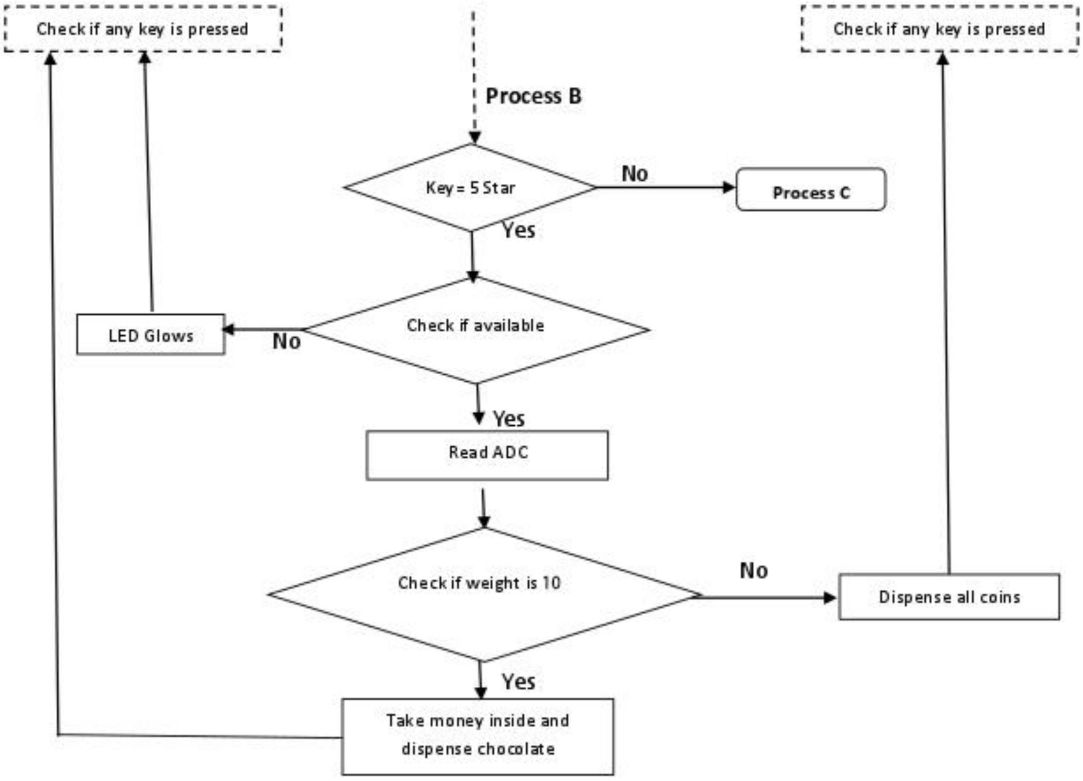
Logic Gates:

Item	Number	Description
NOT	8	Invertors
OR	6	OR Gates

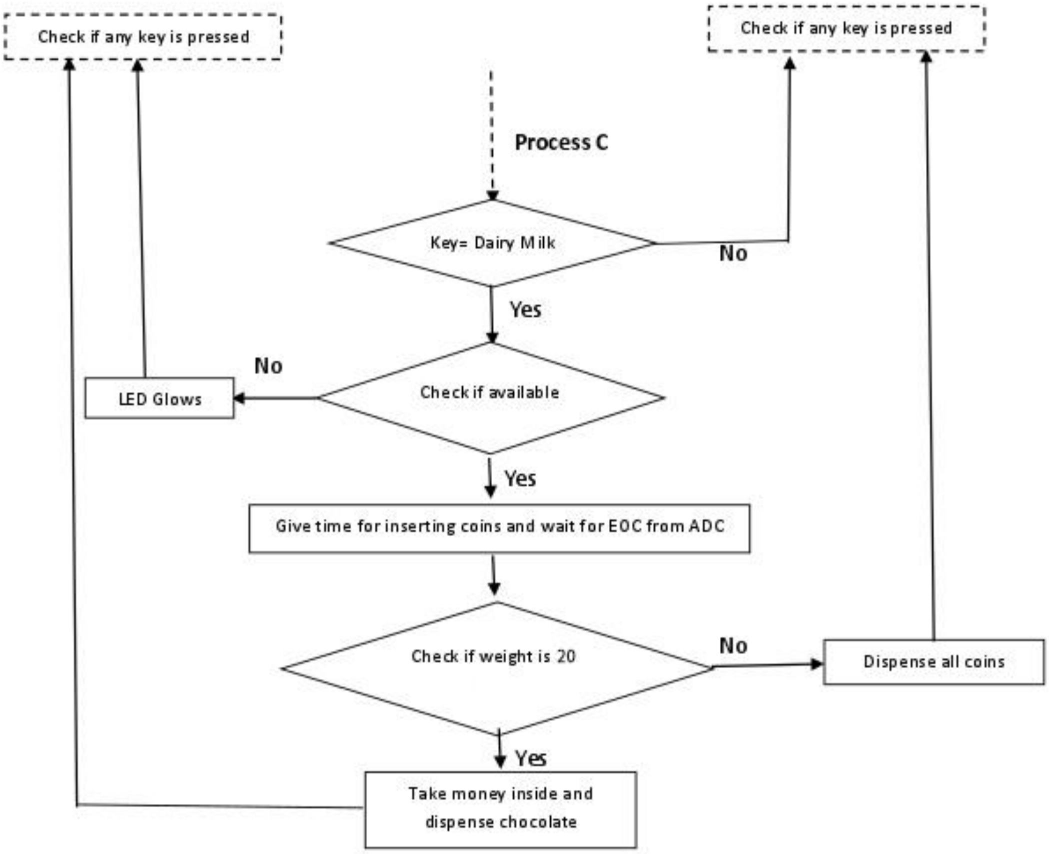
Flowchart for the Process



Process B:


Check if any key is pressed
Process B
Check if any key is pressed
Key = 5 Star
No
Process C
Yes
LED Glows
No
Check if available
Yes
Read ADC
Check if weight is 10
No
Dispense all coins
Yes
Take money inside and dispense chocolate

Process C:


Check if any key is pressed
Check if any key is pressed
Process C
Key= Dairy Milk
No
Yes
No
LED Glows
Check if available
Yes
Give time for inserting coins and wait for EOC from ADC
Check if weight is 20
No
Dispense all coins
Yes
Take money inside and dispense chocolate

ASSUMPTIONS:

- The maximum no of each chocolate is 100.
- If a user inserts a wrong coin, all the coins are returned and he/she has to start from the beginning.
- The user can press only one button at a time.
- LEDs are used to indicate the unavailability of a chocolate.
- The unavailability of a chocolate is indicated only after the user presses the button for that chocolate.
- All the Re. 5 coins weigh the same.
- One chocolate can be dispensed at a time.
- Voltage across LED is 5 volts.
- There are three gates for each of the chocolates that open into the same dispensing slot.
- The wrong coins are also dispensed through the same slot as that for chocolates.
- Only one quantity of any chocolate is ordered at any point of time.
- If extra or less amount of money is inserted the whole amount will be ejected and user has to start over again.
- Machine never exits the code unless it is powered off, because even when no chocolate is available LEDs need to be switched on forever to show user that vending machine is empty.

Working of weight sensor

The rated voltage of the Potentiometer Bridge is 0.5 V / unit % length of resistor. Here we are using a maximum voltage of 50 V as excitation voltage. So the output voltage when the jockey rests at the end point of the resistor will be 50 V. As we are using a range of 0-50g. So output will be 1 V / g

As Vref+ of ADC is connected to 50 V and Vref- is connected to ground so the resolution of ADC 0808 will be (50-0)/255 = 0.2 mV/LSB

Potentiometer produces different voltages for different lengths and using this concept to calculate voltages corresponding to different weights of chocolates, we get

Weight of coins in grams (g)	Voltage (V)	Length of resistance (% of total length)
5g	5V	10% of total length of resistor.
10g	10V	20% of total length of resistor.
20g	20V	40% of total length of resistor.

And the corresponding binary values given by the ADC are:

PERK: 01011000B

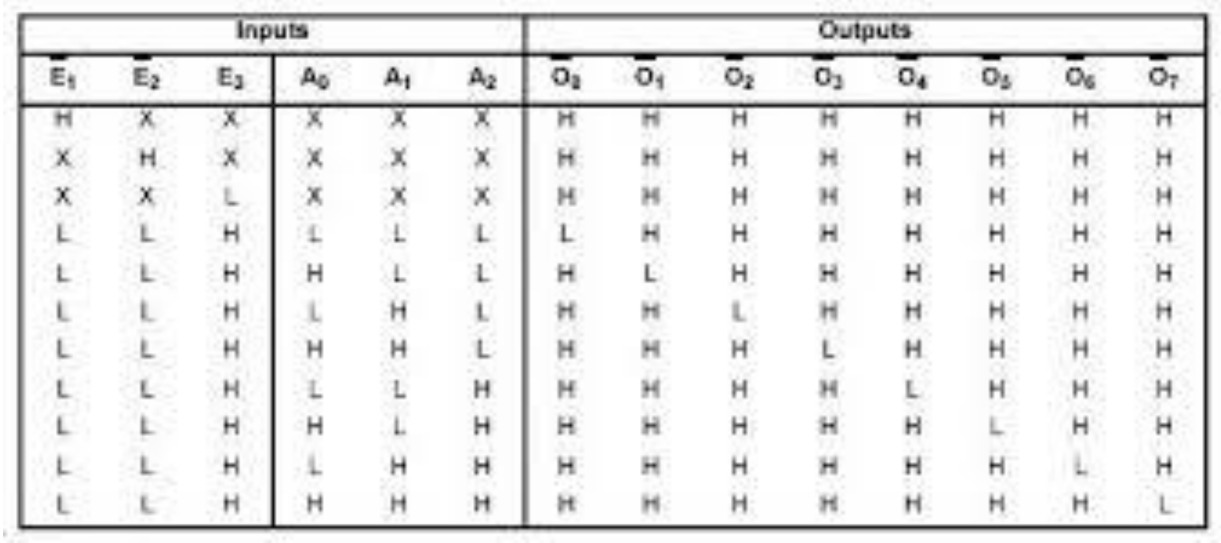
FIVESTAR: 11001100B

DAIRYMILK: 01100110B

Motor Functions:

IN1	IN2	MOTOR FUNCTION
0	0	Stop
0	1	Rotate Backward
1	0	Rotate Forward
1	1	Stop

Truth Table for Decoder:



Inputs						Outputs							
$\overline{E_1}$	$\overline{E_2}$	E_3	A_0	A_1	A_2	$\overline{O_0}$	$\overline{O_1}$	$\overline{O_2}$	$\overline{O_3}$	$\overline{O_4}$	$\overline{O_5}$	$\overline{O_6}$	$\overline{O_7}$
H	X	X	X	X	X	H	H	H	H	H	H	H	H
X	H	X	X	X	X	H	H	H	H	H	H	H	H
X	X	L	X	X	X	H	H	H	H	H	H	H	H
L	L	H	L	L	L	L	H	H	H	H	H	H	H
L	L	H	H	L	L	H	L	H	H	H	H	H	H
L	L	H	L	H	L	H	H	L	H	H	H	H	H
L	L	H	H	H	L	H	H	H	L	H	H	H	H
L	L	H	L	L	H	H	H	H	H	L	H	H	H
L	L	H	H	L	H	H	H	H	H	H	L	H	H
L	L	H	L	H	H	H	H	H	H	H	H	L	H
L	L	H	H	H	H	H	H	H	H	H	H	H	L

System Description

- There is one input slot for inserting coins for the chocolates.

- A weight sensor is used to measure the weight of the coin inserted and verify that the coin entered is actually are Rs. 5 coins.

- When the user inserts a coin, the weight sensor measures the weight and if equal to the weight of the required number coins for that chocolate, it takes the money inside and dispenses the chocolate.

- If any other coin is inserted, the coin would be returned back to the user.

- The coins that collect on the piezoelectric sensor are moved either to the coin collection box or are dispensed back to the user using a motor.

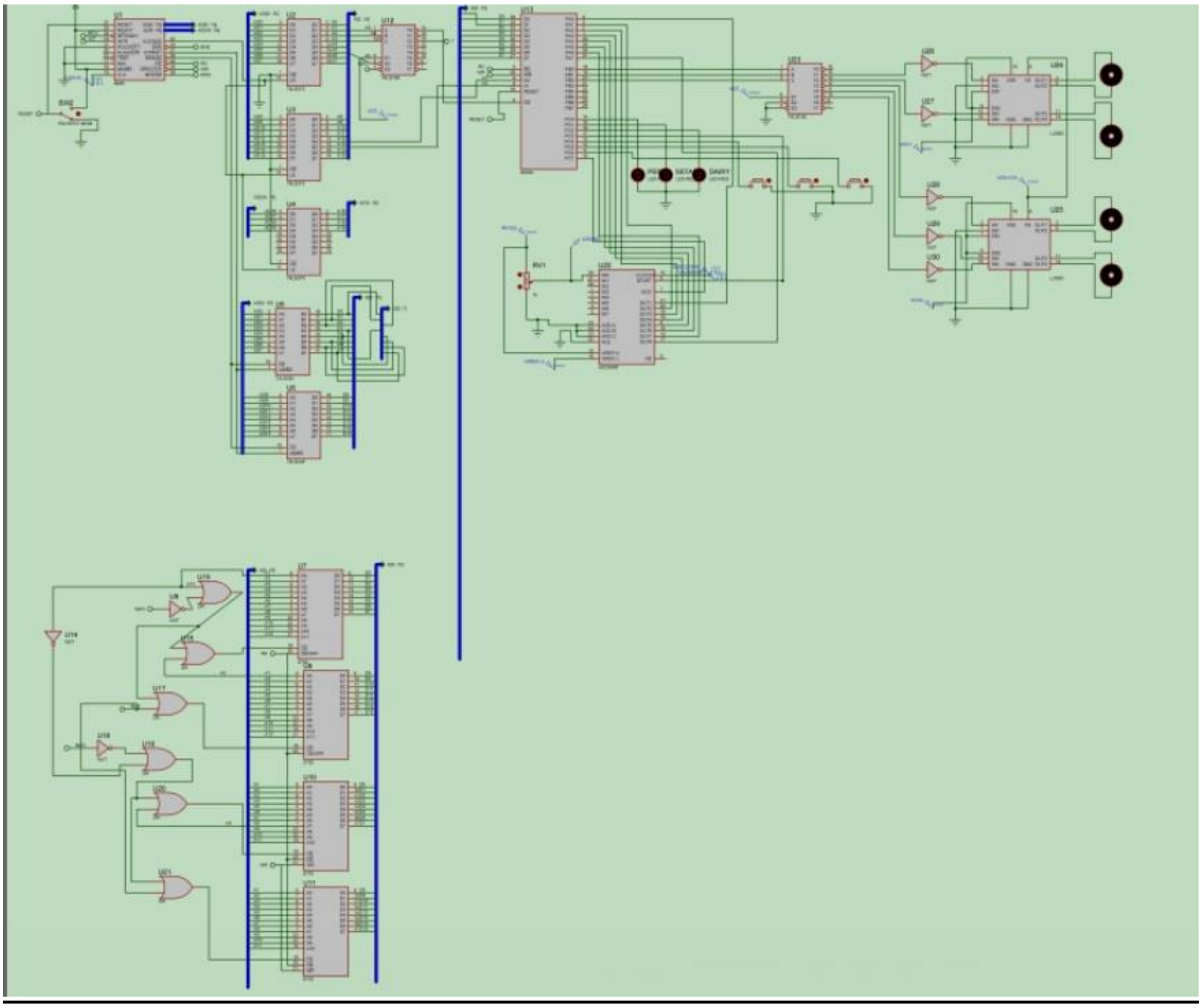
- The owner stores each of the different chocolates in the specified place in the machine.

- The counter gets decremented by 1 every time a chocolate is dispensed.

- LED's are used to indicate non-availability of a particular chocolate.

- Three different motors are used to dispense the chocolates, one for each brand.

Hardware Implementation:



ALP CODE:

```
#make_bin#
#LOAD_SEGMENT=FFFFh#
#LOAD_OFFSET=0000h#
#CS=0000h#
#IP=0000h#
#DS=0000h#
#ES=0000h#
#SS=0000h#
#SP=0FFFEh#
#AX=0000h#
#BX=0000h#
#CX=0000h#
#DX=0000h#
#SI=0000h#
#DI=0000h#
#BP=0000h#


        jmp     st1
        db      1021 dup(0)
; Main program

st1:    cli
; intialize ds, es,ss to start of RAM
         mov       ax,0200h
         mov       ds,ax
         mov       es,ax
         mov       ss,ax
         mov       sp,0FFFEH

PORTA EQU 0000H
PORTB EQU 0002H

; DATA
JMP     START

PORTA           EQU     00h
PORTB           EQU     02h
PORTC           EQU     04h
CREG1           EQU     06h

START:

; Initialize 8255A
; portA as input, portB as output , portC lower as output and portC upper as input

            MOV             AL, 10011000b
            OUT             CREG1, AL

; loading count of chocolates

            MOV             AX, 100
            MOV             [6F0h], AX
            MOV             AX, 100
            MOV             [6F2h], AX
            MOV             AX, 100
            MOV             [6F4h], AX



; setting portb
            MOV             AL, 00000000B
            OUT             PORTB, AL
            MOV             AL, 08H
            OUT             PORTC, AL


MAIN:
; check if key is pressed and figure out whether it is perk or 5star or dairy milk
            IN              AL, PORTC
            AND             AL, 70h
            CMP             AL, 01100000B
            JE              PERK
            CMP             AL, 01010000B
            JE              FIVESTAR
            CMP             AL, 00110000B
            JE              DAIRYMILK
            JMP             MAIN

; perk

PERK:

            MOV             CL, 5
    W1:
            CALL            DELAY_1S
            NOP
            LOOP            W1

            MOV             DL, 01
            LEA             SI, [6F0h]
            MOV             AL, [SI]
            CMP             AL, 0000h
            JNE             ADCWAIT                 ; if perk is present jump to ADCWAIT
            MOV             AL, 00000001B           ; PC0 is set so as to glow the led
            OUT             CREG1, AL
            CALL            DELAY_1S                ; DELAY
            MOV             AL, 98H                 ; resetting 8255A
            OUT             CREG1, AL
            JMP             MAIN

; checking for weight of perk. From ADCWAIT, we jump to P1 this if perk was the selected input.

P1:
            MOV             AL, 98H
            OUT             CREG1, AL
            MOV             AH, 01011000B
            IN              AL, PORTA
            CMP             AL, AH
            JNE             MONEYBACK
```

```
        DEC         [SI]

  ; Logic to rotate the motor so as to take the money in
        MOV         AL, 98h
        OUT         CREG1, AL
        MOV         AL, 00000100B
        OUT         PORTB, AL
        CALL        DELAY_1S
        MOV         AL, 98h
        OUT         CREG1, AL
        MOV         AL, 00000000B           ; stop the motors
        OUT         PORTB, AL

        ; Logic to rotate the motor so as to dispense the chocolate out

        MOV         AL, 00000001B
        OUT         PORTB, AL
        CALL        DELAY_1S
        MOV         AL, 00000000B           ; stopping the motors
        OUT         PORTB, AL
        JMP         CHOCOUT

; 5star

FIVESTAR:
        MOV         DL, 02
        LEA         SI, [6F2h]
        MOV         AL, [SI]
        CMP         AL, 00h
        JNE         ADCWAIT                 ; if 5star is present jump to ADCWAIT
        MOV         AL, 0000011B            ; PC1 is set
        OUT         CREG1, AL
        CALL        DELAY_1S                ; DELAY
        MOV         AL, 98h                 ; resetting
        OUT         CREG1, AL
        JMP         MAIN

; checking for weight of 5star. From ADCWAIT, we jump to P2 if 5star was the selected input.

P2:
        MOV         AL, 98H
        OUT         CREG1, AL
        MOV         AH, 11001100b
        IN          AL, PORTA
        CMP         AL, AH
        JNE         MONEYBACK
        DEC         [SI]

  ; Logic to rotate the motor so as to take the money in
        MOV         AL, 98h
        OUT         CREG1, AL
        MOV         AL, 00000100B
        OUT         PORTB, AL
        CALL        DELAY_1S
        MOV         AL, 98h
        OUT         CREG1, AL
        MOV         AL, 00000000B           ; stop the motors
        OUT         PORTB, AL

        ; Logic to rotate the motor so as to dispense the chocolate out

        MOV         AL, 00000010B
        OUT         PORTB, AL
        CALL        DELAY_1S
        MOV         AL, 00000000B       ; stopping the motors
        OUT         PORTB, AL
        JMP         CHOCOUT

; Dairy milk

DAIRYMILK:
        MOV         DL, 03
        LEA         SI, [6F4h]
        MOV         AL, [SI]
        CMP         AL, 0000h
        JNE         ADCWAIT             ; if dairy milk is present jump to ADCWAIT
        MOV         AL, 00000101B       ; PC2 is set
        OUT         CREG1, AL
        CALL        DELAY_1S            ; DELAY
        MOV         AL, 98h             ; resetting
        OUT         CREG1, AL
        JMP         MAIN

; checking for weight of dairy milk. From ADCWAIT, we jump to P3 if dairy was the selected input

P3:
        MOV         AL, 98H
        OUT         CREG1, AL
        MOV         AH, 01100110b
        IN          AL, PORTA
        CMP         AL, AH
        JNE         MONEYBACK
        DEC         [SI]

        MOV         AL, 98h
        OUT         CREG1, AL
        MOV         AL, 00000100B
        OUT         PORTB, AL
        CALL        DELAY_1S
        MOV         AL, 98h
        OUT         CREG1, AL
        MOV         AL, 00000000B       ; stop the motors
        OUT         PORTB, AL

        MOV         AL, 00000011B
        OUT         PORTB, AL
        CALL        DELAY_1S
        MOV         AL, 00000000B       ; stopping the motors
        OUT         PORTB, AL
        JMP         CHOCOUT
```

```
; adcwait

ADCWAIT:
    ; delay to give time for the user to set weight

            MOV             CL, 07
  T3:
            CALL            DELAY_1S
            LOOP            T3

    ; setting adc on

            IN              AL, PORTC
            AND             AL, 11110111B
            OUT             PORTC, AL


; loop till adc interrupt is raised .ie. eoc is 1

J1:         N               AL, PORTC
            AND             AL, 80H
            CMP             AL, 0
            JE              J1

            IN              AL, PORTA
            JMP             EXITADC

; Logic to decide whether to go perk, or 5star, or dairy milk

T1:         MOV             AH, 1
            CMP             AH, DL
            JE              P1


            MOV             AH, 2
            CMP             AH, DL
            JE              P2

            MOV             AH, 3
            CMP             AH, DL
            JE              P3

; exitadc

EXITADC:
            ; Clear adcst and ale in PC3

            MOV             Al, 98h
            OUT             CREG1, AL
            MOV             AL, 00000111b
            OUT             CREG1, AL

            JMP             T1
```

```
; moneyback

MONEYBACK:
        MOV             AL, 98h
        OUT             CREG1, AL
        MOV             AL, 00000101B
        OUT             PORTB, AL
        CALL            DELAY_1S
        MOV             AL, 98h
        OUT             CREG1, AL
        MOV             AL, 00000000B           ; stop the motors
        OUT             PORTB, AL
        JMP             MAIN

; chocout

CHOCOUT:
        MOV             AL, 98h
        OUT             CREG1, AL
        MOV             AL, 00000101B
        OUT             PORTB, AL
        CALL            DELAY_1S
        MOV             AL, 98h
        OUT             CREG1, AL
        MOV             AL, 00000000B           ; stop the motors
        OUT             PORTB, AL
        JMP             MAIN

; Procedures used

; delay_20ms
DELAY_20MS proc

        MOV             CH, 5
        X1:             NOP
                        NOP
                        DEC     CH
                        JNZ     X1
        RET
DELAY_20MS endp

; delay_1s
DELAY_1S proc

        MOV             BX, 15000
        X2:             CALL            DELAY_20MS
                        DEC             BX
                        JNZ             X2
        RET
DELAY_1S endp
```