ECL-202: MICROPROCESSORS AND INTERFACING

FLOUR PACKING MACHINE

USING MICROPROCESSOR 8085

UNDER THE KIND GUIDANCE OF DR. PARITOSH PESHWE

GROUP MEMBERS

1) RACHIT GUPTA BT21ECE064
2) ISHAN BAJPAI BT21ECE008
3) SUJAL AGARWAL BT21ECE055
4) SHRIYANS CHAVAN BT21ECE016
5) HRITIK GUPTA BT21ECE049

Problem Statement

Design a Microprocessor based flour packing machine. You have flour contained in the silo. The user keys in the required amount of flour per packet. The system should take the input and pack the specified amount of flour upon press of a START key. It is also required to monitor the temperature of the floor where packing is going on. This temperature range can be user settable and also should be displayed. Display the number of packets packed in every hour. An alarm for any malfunctioning of the system should be provided.

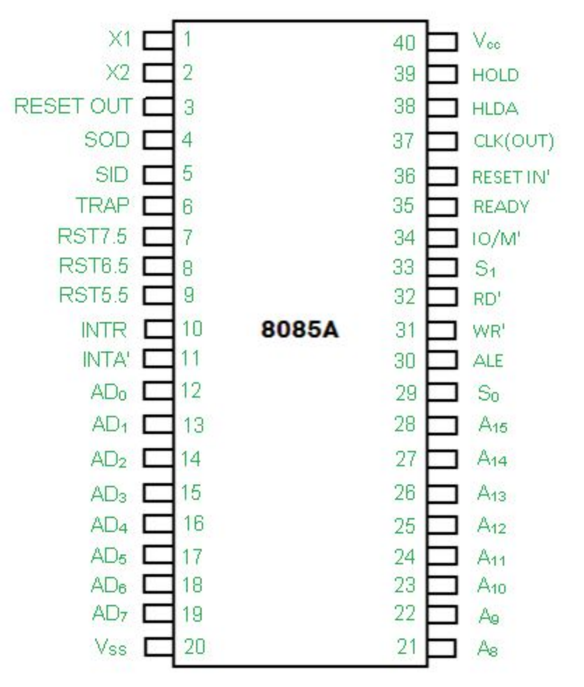
Constituent units of the machine

- Keyboard Input Unit
- Start Button
- Filling Unit
- Bag Counter
- Hour Timer
- Temperature Check
- Display
- Alarm

Components Used

- Microprocessor 8085
- IC 8254
- IC 8255
- Seven Segment Display
- BCD to Seven Segment Decoder
- IC 555
- Keypad
- IC LM135
- LED
- Speaker
- A2D Convertor

Microprocessor 8085

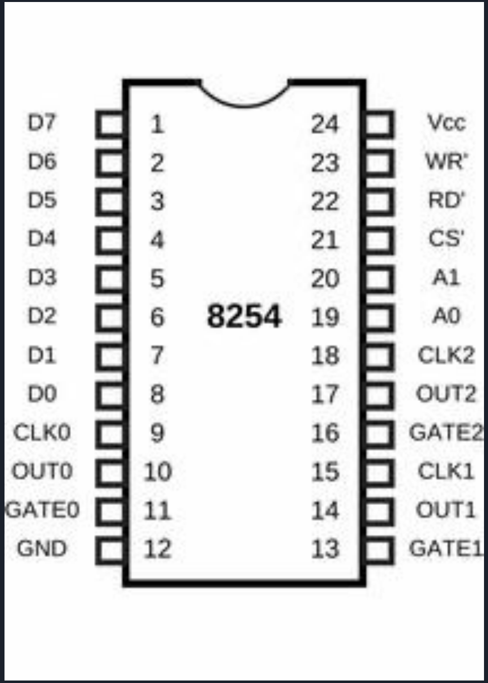


- 8085 is a 8-bit microprocessor produced by Intel and introduced in 1976

- It is a 40 pin IC consisting of multiplexed address/data bus

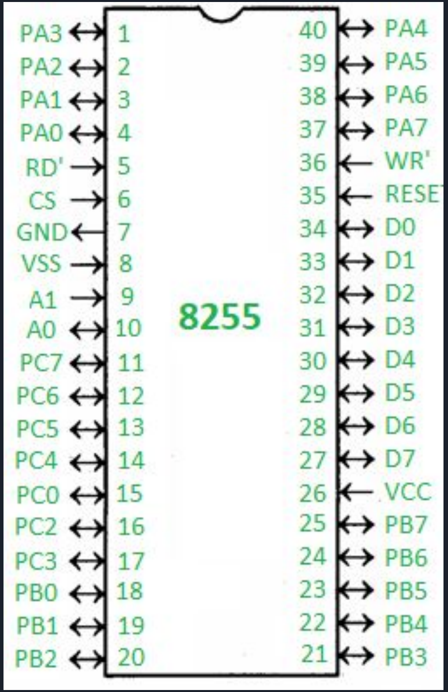
- Clock speed of 8085A is 3MHz, which goes up to 6MHz on the P8085AH-1 microprocessor

IC 8254



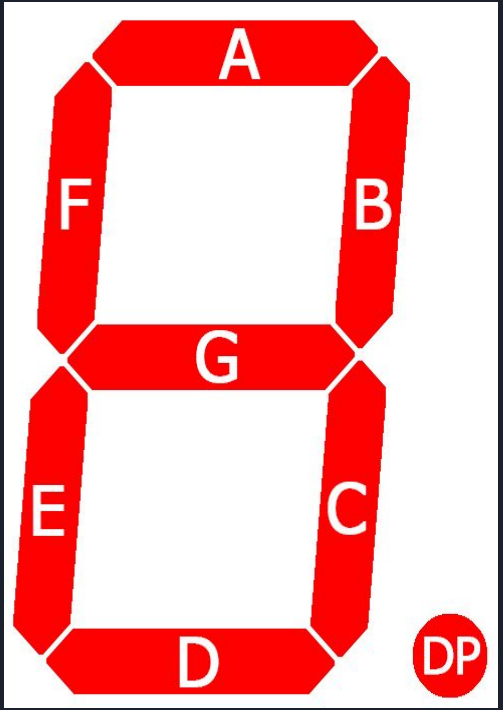
- 8254 is a timer IC developed by Intel. It was an upgrade to its predecessor 8253
- It consists of 3 independent counters, each of which can be run in 6 different modes
- 8254 differs from 8253 because of the inclusion of read back command

IC 8255

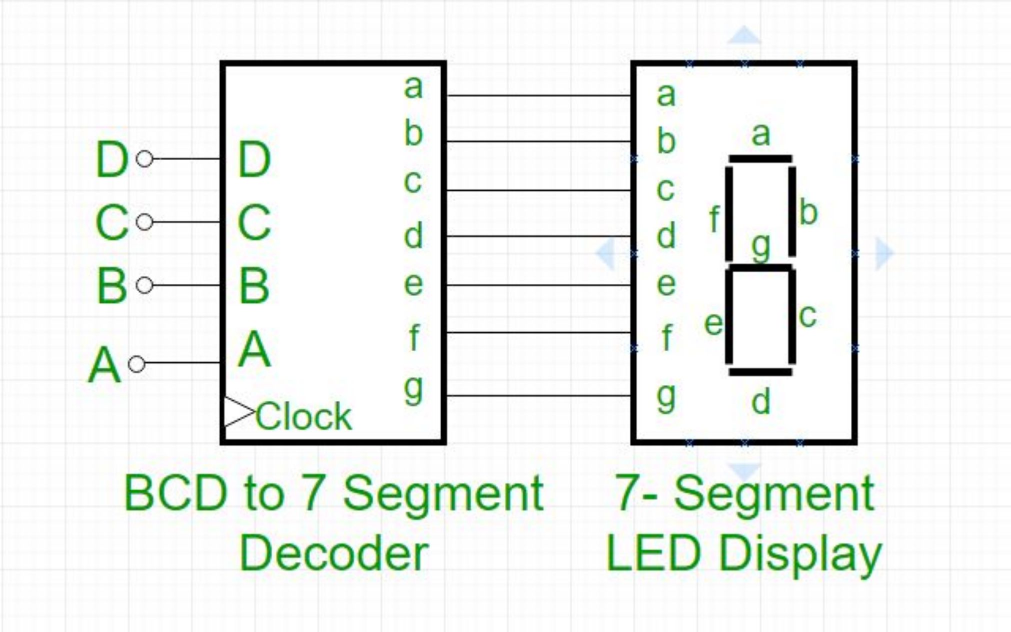


- Programmable I/O device used to interface CPU with peripheral devices
- Consists of 3 eight-bit bidirectional input/output ports namely A, B and C
- Port C is further divided into 2 four-bit ports, namely C-Upper and C-Lower.

Seven Segment Display

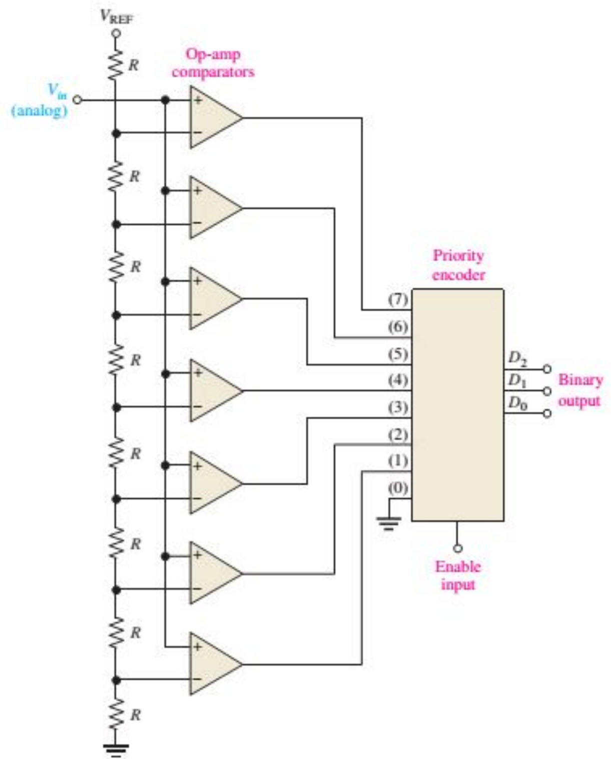


BCD to Seven Segment Decoder

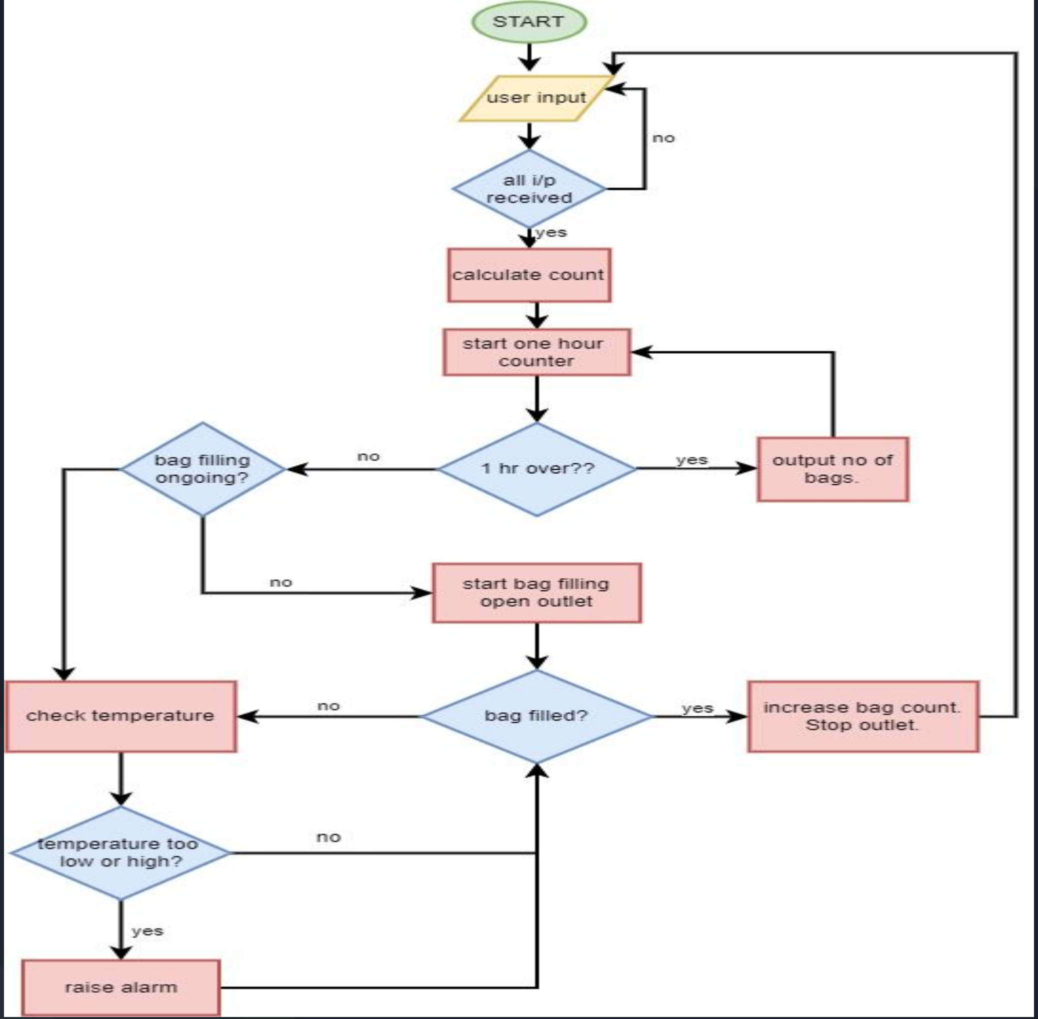


IC LM135

Analog to Digital Convertor



FLOWCHART


START
user input
no
all i/p received
yes
calculate count
start one hour counter
1 hr over??
yes
output no of bags.
no
bag filling ongoing?
no
start bag filling open outlet
bag filled?
yes
increase bag count. Stop outlet.
no
check temperature
temperature too low or high?
no
yes
raise alarm

User Inputs

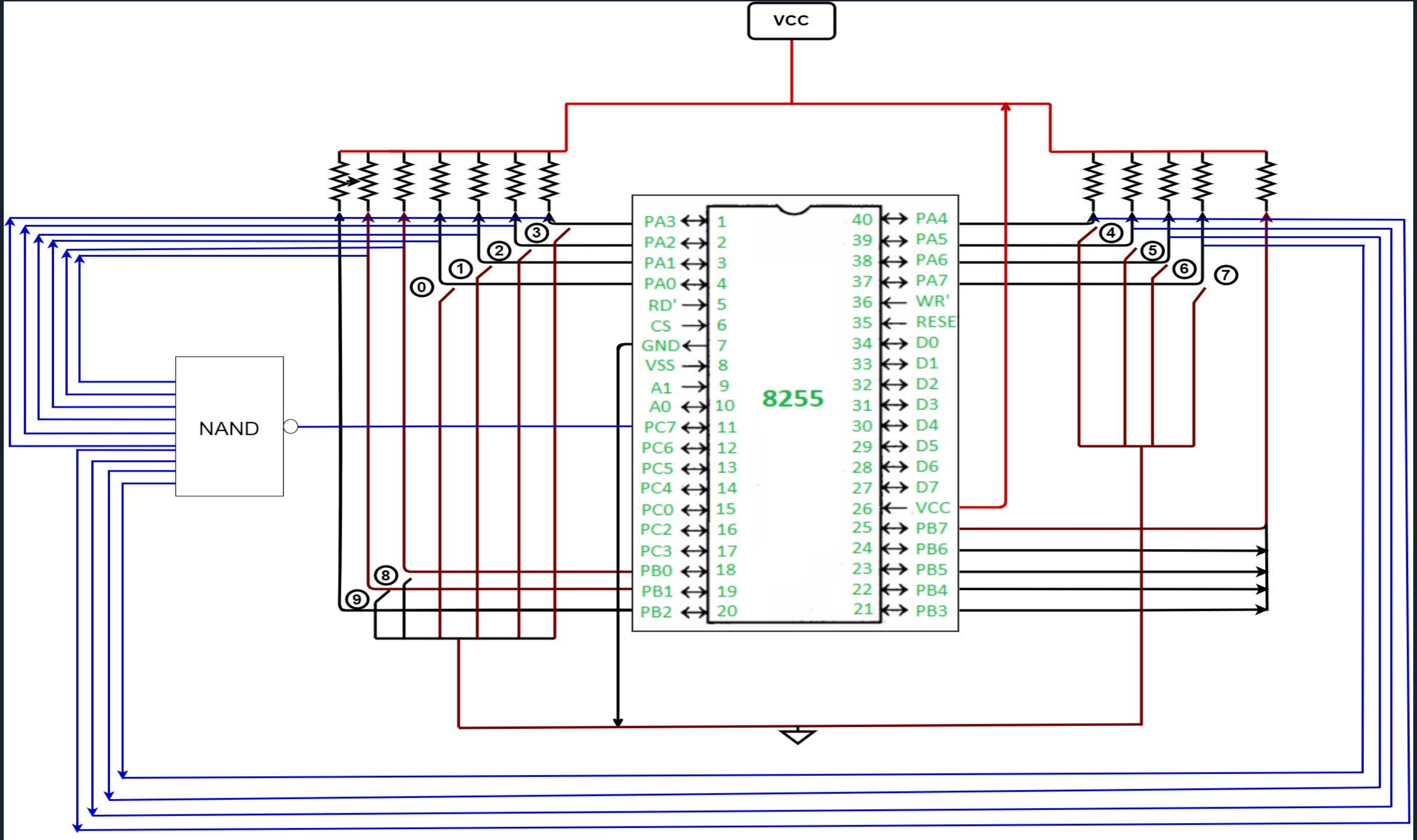
Approach

- User inputs weight of flour to be packed on a number pad

- Number pad is interfaced with the ports of an IC 8255

- Whenever a key is pressed, logic low (0) is received on the corresponding pin, while the rest are high

- Input is read by the microprocessor and is stored in the accumulator for calculations.

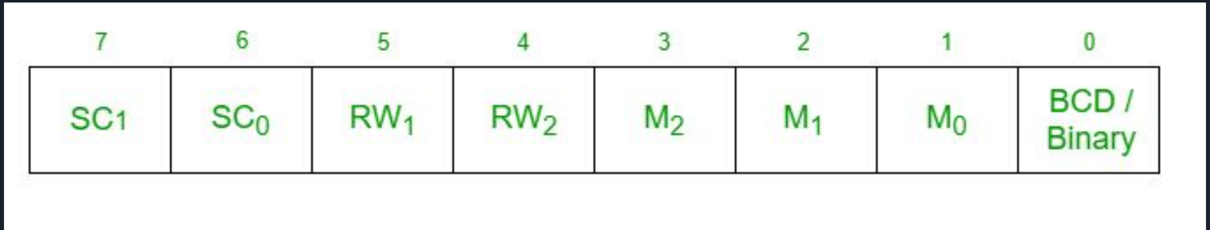


Keypad Interfacing

- Register D stores (02)H initially, number of keystrokes remaining and gets decremented upon each stroke
- Numbers 0 to 7 are connected to port A and 8,9 to port B
- Program runs in an infinite loop, exiting the loop only when a key is pressed
- HL pair initialized with a memory location to hold MSB and LSB of input
- If number pressed is 8 or 9, it is transferred to memory
- Else, accumulator is rotated until we find the bit containing ‘0’, and the corresponding number is stored in memory

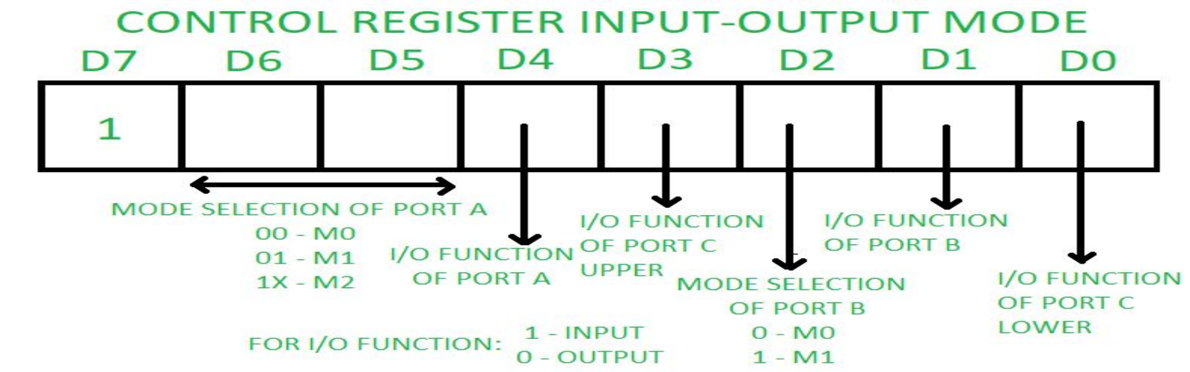
- After the first keystroke, program returns to the infinite loop again, awaiting second input
- When second stroke is received, same process takes place, and the corresponding digit is committed to memory.
- If keystrokes are exhausted, the program moves forward and engages in another loop, awaiting for temperature input.
- Temperature is taken as input in the same manner as weight of each packet, and then stored in memory.
- Program now awaits the pressing of the ‘Start’ button

CWR Format



7	6	5	4	3	2	1	0
SC_1	SC_0	RW_1	RW_2	M_2	M_1	M_0	BCD / Binary

8254



CONTROL REGISTER INPUT-OUTPUT MODE

D7	D6	D5	D4	D3	D2	D1	D0
1							

MODE SELECTION OF PORT A
00 - M0
01 - M1
1X - M2

I/O FUNCTION
OF PORT A

I/O FUNCTION
OF PORT C
UPPER

MODE SELECTION
OF PORT B
0 - M0
1 - M1

I/O FUNCTION
OF PORT B

I/O FUNCTION
OF PORT C
LOWER

FOR I/O FUNCTION: 1 - INPUT
0 - OUTPUT

8255

```
[KEY_IP]: MVI C,02H

LXI H,3001H
MVI A,9BH
OUT 43H //CWR
EI

[RE]: IN 42H
      RAL
      JNC <RE>

IN 41H
CALL <DELAY>
CPI 7FH
JZ <L1>

    L[1]: MVI B,08H
          MOV M,B
         DCX H
         DCR C
         MOV A,C
         ORI FF
         JNZ <RE>
         RET
```

```
CPI BFH
JZ <L2>
     [L2]: MVI B,09H
          MOV M,B
          DCX H
          DCR C
          MOV A,C
          ORI FF
          JNZ <RE>
          RET

IN 40H
CALL <DELAY>
     [DELAY]: LXI D,09C4H
           [L4]: DCX D
                 MOV A,E
             ORA D
             JNZ <L4>
          RET
```

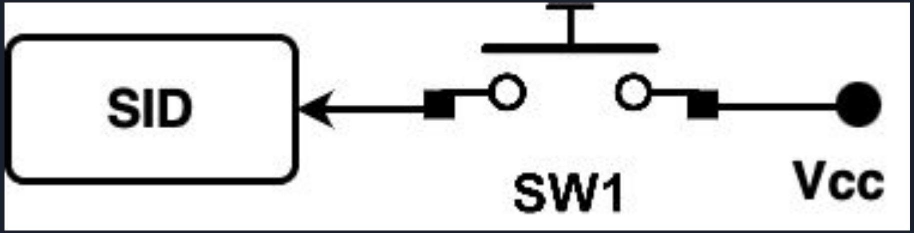
```
CMA
ORA A
MVI D,08H

[find0]: DCR D
         RAL
         JC <find0>

MOV M,D
DCX H
MOV A,C
ORI FF
JNZ <RE>
```

Start Button

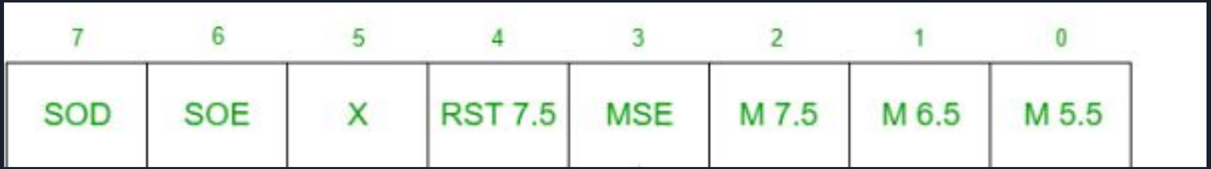
- Start button connected to serial input port of microprocessor
- After taking in the keyboard inputs, the microprocessor keep running in a loop, checking for active high on the serial input pin
- When input high is received, the processor breaks out of the loop and starts further execution.
- In a similar way, a push switch is also connected to the TRAP pin, which is used to halt the microprocessor in an emergency situation



```
[loop]: RIM
        RAL
        JZ <loop>
```

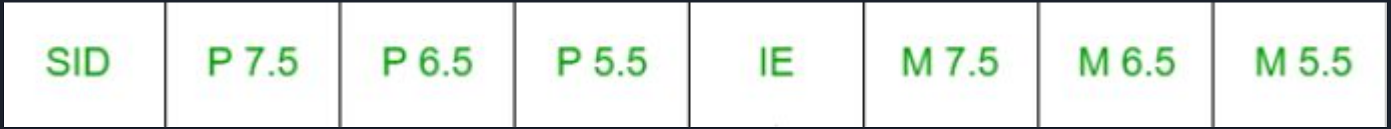
SIM AND RIM FORMAT

SIM



7	6	5	4	3	2	1	0
SOD	SOE	X	RST 7.5	MSE	M 7.5	M 6.5	M 5.5

RIM



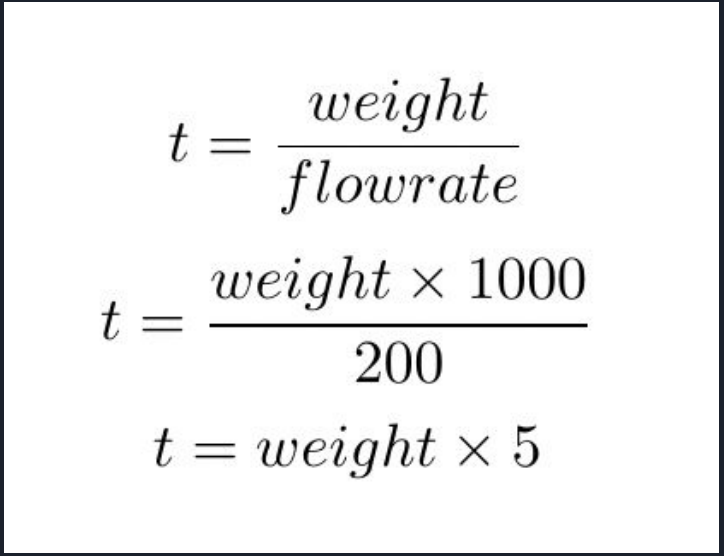
SID	P 7.5	P 6.5	P 5.5	IE	M 7.5	M 6.5	M 5.5

Initialization

Approach

- It is assumed that the hoppers have a uniform flow rate, filling the bags at a constant rate of 200 grams per second
- The timer responsible for turning ON and OFF the outlet is initialized with a suitable count
- The temperature is measured using IC LM135, the output of which is passed to an analog-to-digital convertor
- The digital data received in an 8-bit format is used to decide the upper limit for temperature, the calculations for which are also done

- MSB of weight is pulled from memory to a register
- It is then multiplied by (0A)H, and added to the LSB. The result gives back the actual weight in hexadecimal
- Time required for filling the bag is then calculated-



$$t = \frac{weight}{flowrate}$$

$$t = \frac{weight \times 1000}{200}$$

$$t = weight \times 5$$

- The clock being given to the timer is of 100ms. So now a suitable count is calculated corresponding to the time delay to be generated to fill the bag
- One second delay requires 10 clock pulses, and on each pulse, the counter decrements by 1.
- So the count becomes time multiplied by 10, or (0A) in hexadecimal.
- This count is stored in a register to be supplied later

```
INX H
MOV C,M
CALL <MULT>
     [MULT]: XRA A
          MVI B,0AH
           [L7]: ADD C
                 DCR B
                 JNZ <L7>
                 DCX H
                 ADD M
                 LXI H,3003H
                 MOV M,A
                   DCR B
                   JNZ <L7>
DCX H
ADD M
LXI H,3003H
MOV M,A

XRA A
MOV E,M
MVI B,05H
```

```
[loc]: ADD E
     JC <L8>
     [L8]: MVI D,01H
           RET

      DCR B
      JNZ <loc>

MOV C,E
MOV B,D
```

Clock Generator

- Clock is generated using an astable multivibrator
- The free running multivibrator is constructed using IC 555. It has a 50% duty cycle and does not require any triggering to change states.
- The time period for which the output is high is calculated using the following formula-

$$T_{high} = 0.693\,(R_A + R_B)\,C$$

- To get a clock of 100ms duration, the following values of R and C are chosen to be 7.215 kΩ and 100 nF respectively
- To obtain a duty cycle of 50%, a diode is also added to the circuit

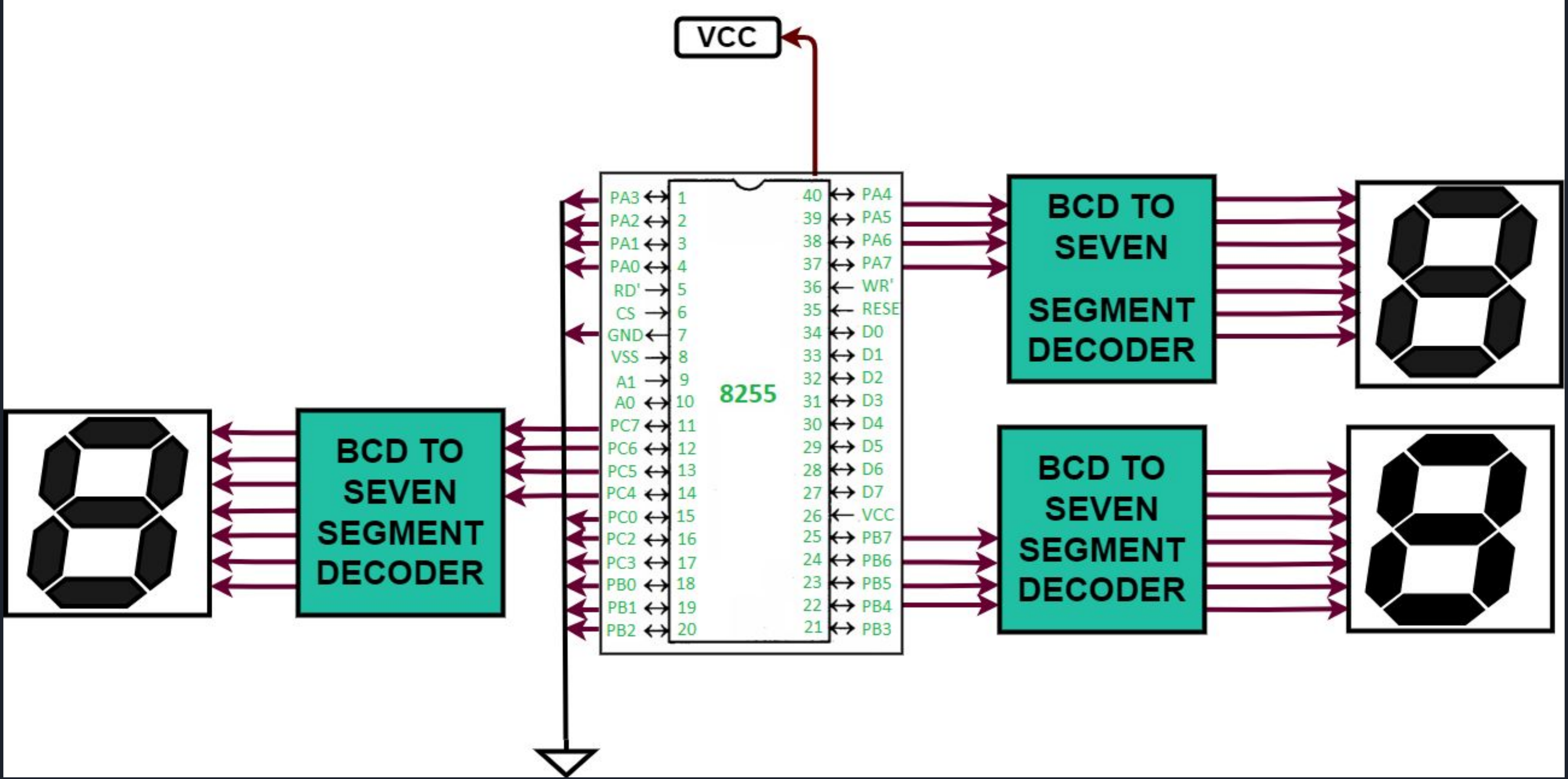
Hour Counter

Approach

- IC 8254 is used for this purpose
- Timer is initialized with a suitable counter, and is then started
- The counter keeps decrementing for a time duration of 3600 seconds, the clock for which is provided by the astable multivibrator
- GATE pin is always kept high, so that counting begins as soon as count is loaded
- At the end of execution, OUT pin goes high, which triggers an interrupt service routine

- Clock input is of 100ms, and an hour consists of 3600 seconds. This gives a count of 36000, which is (8CA0)H in hexadecimal
- GATE pin of the timer is connected to a +5V source, so that it always remains high
- OUT pin is redirected to an interrupt, RST 6.5, which tells the processor what is to be done after an hour is complete
- Count is loaded onto the timer. As soon as it is loaded, the countdown starts
- The timer is operated in mode 0, which stops after the counting is complete, and the OUT pin shifts to logic high

- Interrupt subroutine also triggers the display of number of bags packed in the last hour
- The number of bags packed is stored in DE register pair in BCD format
- Numbers in BCD format are sent to ports of another IC 8255, which are connected to seven segment displays via BCD to Seven Segment Decoders
- The number of bags packed in an hour is displayed for the entire duration until the next hour ends



Seven Segment Display Interfacing

```
[counter] : LXI D,0000H
           MVI A,B0H
            OUT 83H
            MVI A0H
            OUT 82H
            MVI 8CH
            OUT 82H
```

```
[ISR 6.5]: XRA A
           MVI A,80H
           OUT 53H
           MOV A,D
           OUT 50H

           MOV A,E
           OUT 51H
           JMP <counter>
```

Filling Mechanism

Approach

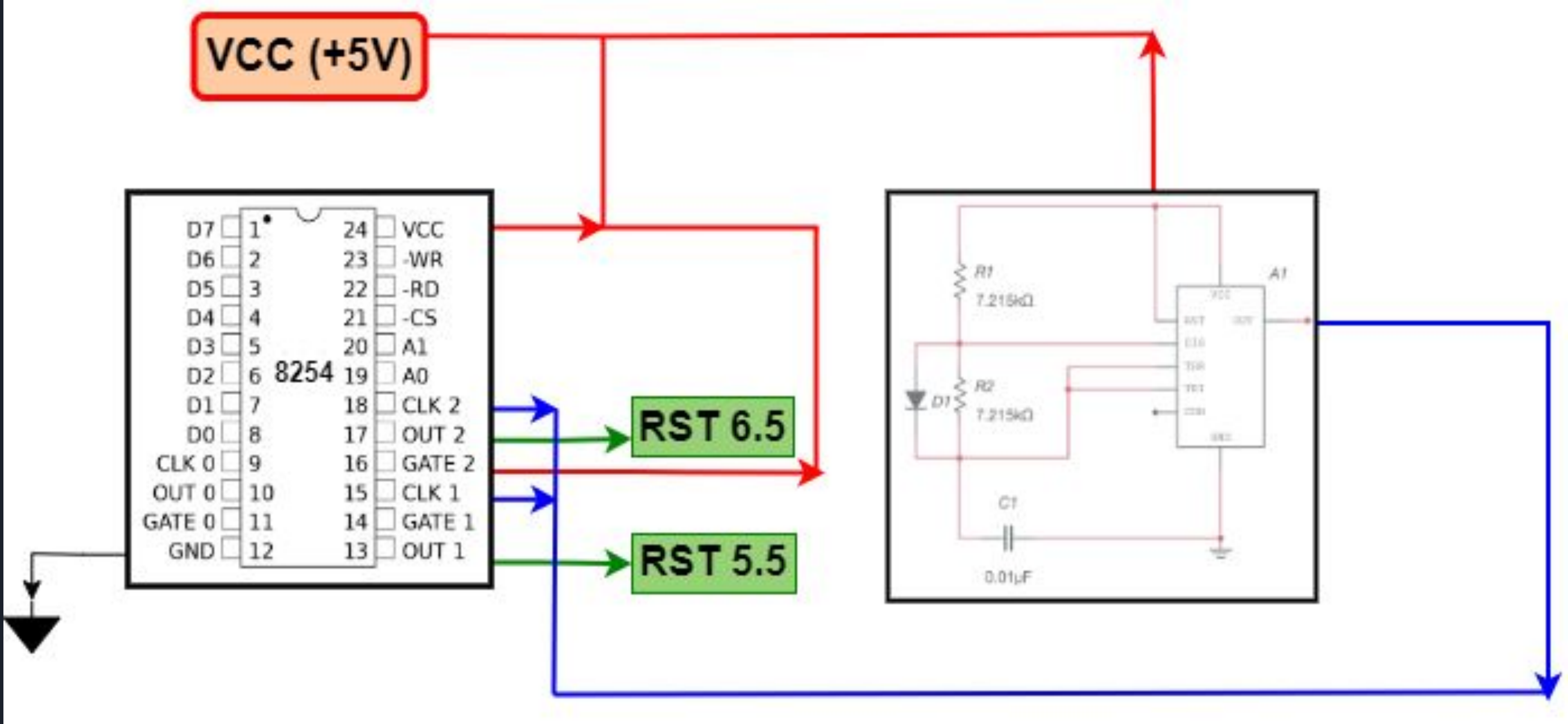
- Initialize timer with appropriate count
- Start the timer by giving GATE signal
- Start the outlet along with the timer
- When timer stops, stop the outlet and increment bag count

- Appropriate count for the timer has already been calculated and committed to memory
- The count is retrieved and the timer is loaded with the count
- Alongside the timer, the outlet pipe is let open to start the flow of flour into the packet
- When the timer ends, a logic high level is obtained at the output, which triggers another interrupt
- The interrupt subroutine turns off the outlet pipe
- Bag count in the microprocessor is incremented by 1
- If LSB of count becomes ‘A’, register contents are incremented by 6 to keep the number in BCD format

```
MVI A,92H
OUT 73H
MVI A,80H
OUT 70H

[B1] : MVI A,30H
       OUT 93H
       MOV A,C
       OUT 90H
       MOV A,B
       OUT 90H
       [Loc]: Call <TEMPCHK>
              JMP <Loc>
```

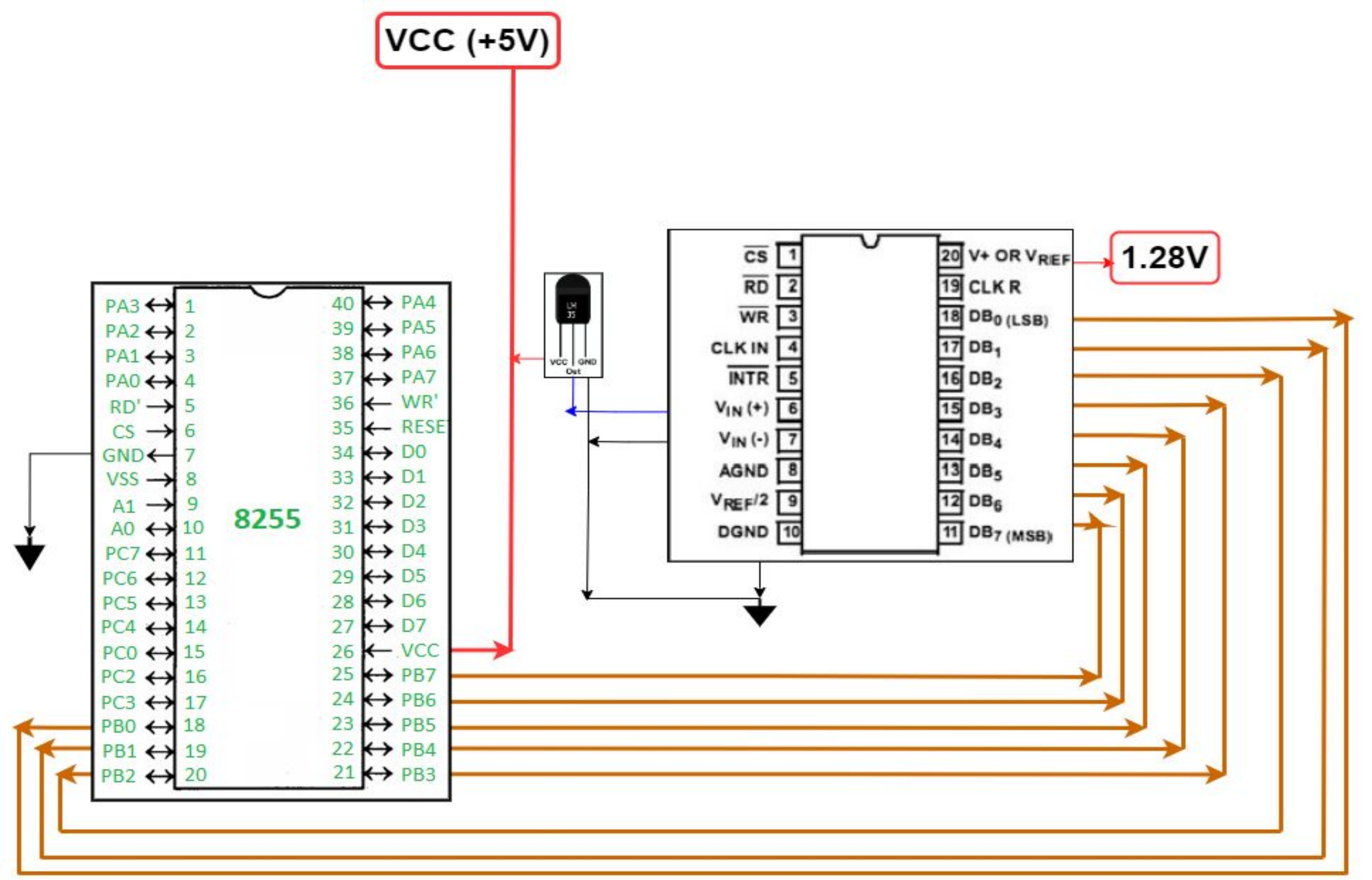
```
[ISR 5.5]: MVI A,00H
           OUT 70H
           INX D
           MOV A,E
           ANI 0FH
           CPI 0AH
           CZ <LOC>
                [LOC]: LXI H,0006H
                       DAD D
                       XCHG
           JMP <KEY_IP>
```


VCC (+5V)
D7
D6
D5
D4
D3
D2
D1
D0
CLK 0
OUT 0
GATE 0
GND
1
2
3
4
5
6
7
8
9
10
11
12
8254
24
23
22
21
20
19
18
17
16
15
14
13
VCC
-WR
-RD
-CS
A1
A0
CLK 2
OUT 2
GATE 2
CLK 1
GATE 1
OUT 1
RST 6.5
RST 5.5
R1
7.215kΩ
R2
7.215kΩ
D1
C1
0.01µF
A1
VCC
RST
OUT
DIS
THR
TRI
CON
GND

Temperature Unit

- LM 135 IC is used to convert temperature to a voltage signal
- An 8-bit analog to digital converter is used to convert incoming voltage to an 8-bit digital value
- V+ and V- for ADC are chosen in such a way that the sensor is able to measure temperatures in the range of 0 to 127 degree celsius
- At 0 degrees, the input received is (00)H while at 128 degrees it shoots to (FF)H
- For 128 discrete values of temperature, there are corresponding 256 values contained in 8 bits
- Each degree change in temperature corresponds to a change of 2 in the digital domain

- The upper and lower limits of temperature (stored in HEX) are converted to another numerical value which corresponds to the temperature input from the A2D convertor
- A subroutine is created which takes in the temperature in hex as input, and compares it with lower and upper limits of temperature
- If the ambient temperature falls below the specified level OR rises further than the safe limit, another subroutine is executed, which turns on a light and a siren
- The temperature checking subroutine is repetitively called while the bag is being filled
- One more subroutine is tasked out with displaying the limits of temperature on seven segment displays



LM135 Interfacing

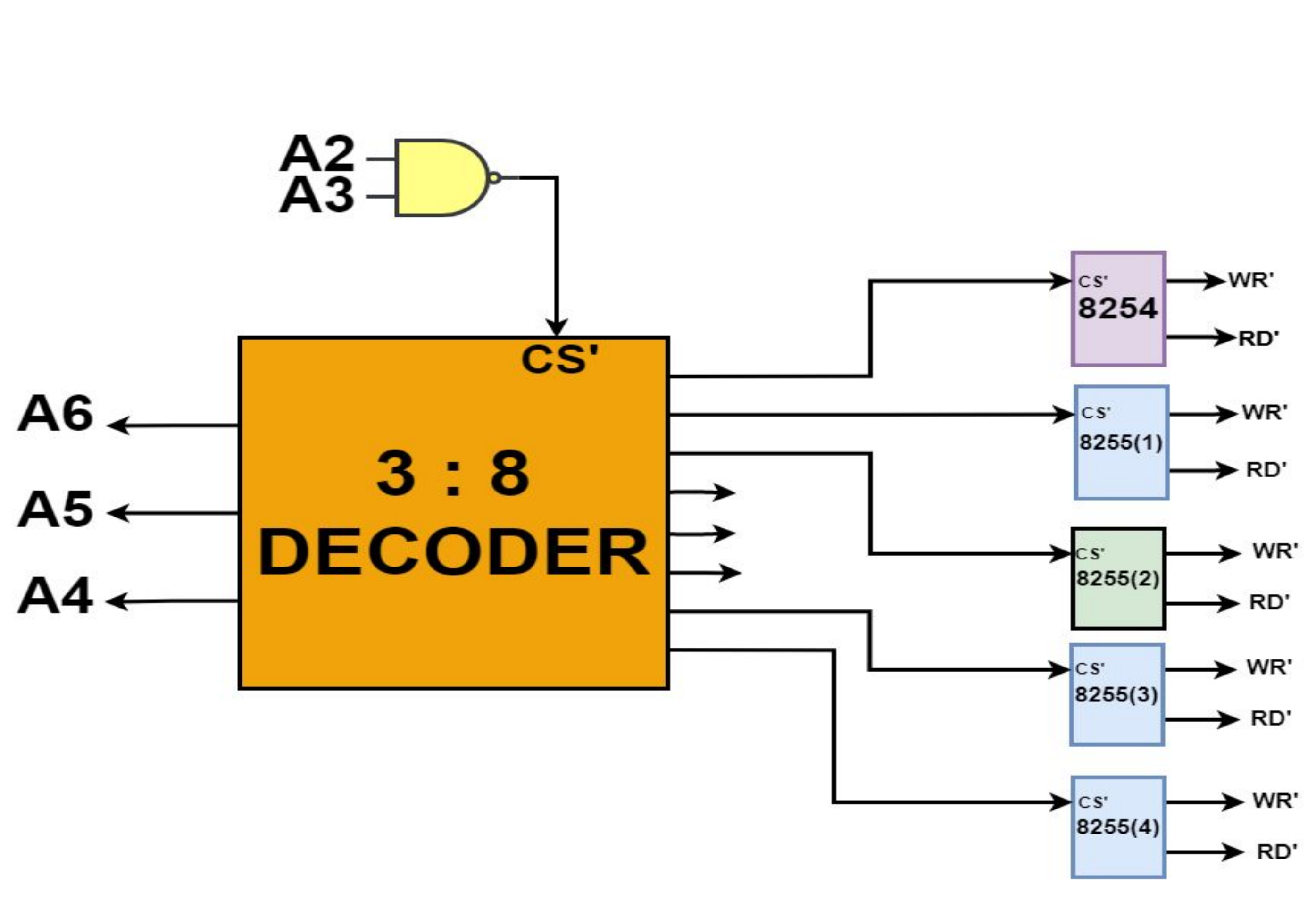
```
[TEMPDISP]: LXI H,300BH
            MVI 89H
            OUT 63H
            CALL <HEX2BCD>
            OUT 61H
            LXI H,300CH
            CALL <HEX2BCD>
            OUT 62H
            RET


[ALARM]: MVI A,80H
           OUT 73H
            MVI A,80H
            OUT 71H
            OUT 72H
            RET
```

```
[TEMPCHECK]: MVI A,89H
             OUT 63H
             IN 62H
             CALL <DELAY1>
             MOV H,A
             LDA 300BH
             CMP H
             JNC <ALARM>
             IN 62H
             CALL <DELAY1>
             MOV H,A
             LDA 300CH
             CMP H
             JC <ALARM>
             XRA A
             RET
```

```
[HEX2BCD]: MVI C,00H
     MOV B,C
     LDA 0000H
     [L1]:CPI 64H
          JC <L2>
          SUI 64H
          INR C
          JMP <L1>
     [L2]:CPI 0AH
          JC <L3>
          SUI 0AH
          INR B
          JMP <L2>
     [L3]:MOV L,A
          MOV A,B
          RLC
          RLC
          RLC
          RLC
          ADD L
          MOV L,A
          MOV H,C
          RET
```

S No	IC	Port	Hex Address	Binary Address
1	8255	Port A	40 H	0100 0000
		Port B	41 H	0100 0001
		Port C	42 H	0100 0010
		CWR	43 H	0100 0011
2	8255	Port A	50 H	0101 0000
		Port B	51 H	0101 0001
		Port C	52 H	0101 0010
		CWR	53 H	0101 0011
3	8255	Port A	60 H	0110 0000
		Port B	61 H	0110 0001
		Port C	62 H	0110 0010
		CWR	63 H	0110 0011
4	8255	Port A	70 H	0111 0000
		Port B	71 H	0111 0001
		Port C	72 H	0111 0010
		CWR	73 H	0111 0011
5	8254	Counter 0	80 H	1000 0000
		Counter 1	81 H	1000 0001
		Counter 2	82 H	1000 0010
		CWR	83 H	1000 0011


A2
A3
CS'
A6
A5
A4
3 : 8
DECODER
CS'
8254
WR'
RD'
CS'
8255(1)
WR'
RD'
CS'
8255(2)
WR'
RD'
CS'
8255(3)
WR'
RD'
CS'
8255(4)
WR'
RD'

THANK YOU