# Weather Monitoring Station

ACHYUDH RAM - 2013B3A7439G

DHRUV JAIN – 2013B3A7826G

SHANTANU JAIN – 2013B3A7547G

SNEHITH ALAPATI – 2013B3A7709G

BATCH NO. 52

BITS Pilani K.K. Birla Goa Campus
Microprocessor Programming and Interfacing (CS F241)

# ACKNOWLEDGEMENT

We would like to mention our gratitude towards Dr. K.R Anupama for her support towards completion of the design project. We would like to cite her research in the field of wireless sensor networks and Microprocessor assembly development as a key study that helped us in understanding and implementing the project of designing a weather monitoring system. We are grateful to her for the education endowed to us during the Microprocessor Programming and Interfacing course which provided us the platform necessary to work on this project.

We would also like to mention our gratitude towards Mr. Kadam Bhushan, who was our immediate mentor for this project and helped us in understanding the intricacies of the design that was required from us.

# Design Problem

<u>System Description:</u> This system monitors weather parameters such as: Air Temperature, Air-Humidity, barometric Pressure, and Displays the average over regular intervals of an hour on a seven-segment display. The Display is continuous. Update of the display is done once in an hour. Weather parameters are sensed at regular intervals of 5 minutes.

The display is of the format: "Temperature – Value °C" and so on.

Other than the regular display, the user can request the display of the weather parameters to be updated at any point of time by pressing a push button key. The accuracy of the parameters monitored has to be up to two decimal points.

# Assumptions

- The Temperature sensor WE700 is capable of a resolution up to 0.1°C but the resolution used in this design is 1°C as an 8-bit ADC converter 0808 is being used. WE700 is capable of detecting temperatures ranging from -50°C to +50°C but this design uses the sensor over the range -20°C to +50°C, so as to accommodate only the temperatures within the operating temperature ranges of the WE100 Pressure Sensor and the WE600 Humidity Sensor.
- The simulation has a PIT 8253 interrupt output with time period that is faster than 5 minutes so that the changes can be seen quickly during debugging and demo. The actual design uses a counter value that produces regular 5-minute interval interrupts. Moreover, the actual design uses an 8254 with a 5 MHz clock input but due to compatibility issues in the simulation software an 8253 with a 2 MHz clock input is used.
- Due to compatibility issues with PIC 8259 in the simulation software (Proteus), the interrupts are simulated using software interrupts rather than actual hardware interrupts.
- The ROMs starting at address FE000h are not used in the simulation as the microprocessor can be made to start directly at 00000h in the Proteus software
- The display on the LCD displays an average of the previous 12 values read, i.e., the previous hour.
- Each time the user presses the external button, the clocks are not reset, implying that the next reading continues to take place as per the original 5-minute scheme which is set. On the button press, a new value is taken, added to data stored in memory and then, the past 12 values are taken for averaging, scaling and displaying on the LCD monitor.
- The button press does not clash with the 5-minute interrupt in normal usage since the probability for the same is very small in real-time usage of the weather monitoring station. However, in case of clash during operation (highly unlikely) and non-servicing of button interrupt, a second press will ensure the servicing of the interrupts, without affecting the 5-minute interrupt servicing.
- We are assuming that the humidity will not be 100% on average over the last 12 readings. In case the humidity is 100%, the seven segment displays display all zeroes and a LED glows next to the humidity reading.
- In case the temperature is negative, the seven segment displays will show the values with a LED glowing next to them to indicate a negative sign.

## Components Used

- Random Access Memory UT-6116
- Read Only Memory UT-2732
- 8-bit latches 74LS373
- 8-bit bidirectional buffers 74LS245
- MAX472 High Side Current Sense Amplifier
- Microprocessor 8086
- Clock Generator 8284
- Programmable Interrupt Controller 8259
- Programmable Peripheral Interface 8255
- 8x1 Decoders 74LS138
- Programmable Interval Timer 8254
- Analog to Digital Converter ADC0808
- External Push Button
- Resistors of 10k ohm
- Quad 2-input OR Gates 7432
- BCD to Seven Segment Decoders 7447
- Seven Segment LED displays
- Temperature Sensor WE700
- Humidity Sensor WE600
- Pressure Sensor WE100
- 2N5086 Transistor Amplifier

## Specifications

The analog input for the system is received from the sensors which provide output in variable current between 4-20mA. This output is converted into analog voltage ranging from 0-5V using the MAX472 amplifier. This analog voltage output is received by an 8 bit parallel ADC (0808), which in turn, generates an 8 bit value between 0 and 255.

There is a timer IC 8254 that generates interrupts every 5 minutes. An 8259 Programmable Interrupt controller device is connected so as to receive three interrupts from various sources, namely the 8254 timer, an external button and an EOC interrupt from the ADC. The Interrupt Vector Table (IVT) is stored in the ROM at memory addresses of 200h onwards (corresponding to a vector number 80h*4=00200h).

Every five minutes, an interrupt is generated and an ISR is invoked, inside which the ADC value is read and this digital data of the temperature, pressure and humidity is stored in the RAM. A count of the number of 5-minute interrupts received is stored in memory. A value of 12 in this memory location signifies the passage of 1 hour, upon which the average of the last 12 stored values is scaled and displayed on the respective Seven Segment displays via the 7447 decoder. The interrupt count stored in the memory is then reset. It is always ensured that the number of readings stored in memory does not exceed 12, and older readings are overwritten in memory.

There is also an external button which on pressing, generates an interrupt, the ISR of which takes a reading from the ADC and averages the last 12 readings (including the current reading i.e. the past hour). This scaled value is then displayed on the respective Seven Segment displays via the 7447 decoder.

## Sensor Specifications

### Temperature Sensor WE700

| | |
|---|---|
| Range of detection | -20°C to +50°C |
| Output Range | 4mA to 20 mA |
| Accuracy | ±1°C |
| Warm up time | 5 seconds minimum |
| Operating Temperature | -50°C to +100°C |
| Operating Voltage | 10V to 36V DC |
| Current Drawn | Same as Sensor Output (4-20mA) |

Working Equation: $T = \frac{7V-10000}{500}$ T- Temperature in °C V-Voltage received by ADC 0808 in mV

Sensor Resolution: 71.4 mV/°C

### Pressure Sensor WE100

| | |
|---|---|
| Range of detection | 800mB to 1100mB (mB- milliBars) |
| Output Range | 4mA to 20 mA |
| Accuracy | ±2 mB |
| Warm up time | 3 seconds minimum |
| Operating Temperature | -40°C to +55°C |
| Operating Voltage | 10V to 36V DC |
| Current Drawn | Same as Sensor Output (4-20mA) |

Working Equation: $P = \frac{3V+40000}{50}$ P-Pressure in milli Bar V-Voltage received by ADC 0808 in mV

Sensor Resolution: 33.3 mV/2mB

### Humidity Sensor WE600

| | |
|---|---|
| Range of detection | 0% to 100% RH (Relative Humidity) |
| Output Range | 4mA to 20 mA |
| Accuracy | ±1% |
| Warm up time | 3 seconds minimum |
| Operating Temperature | -40°C to +55°C |
| Operating Voltage | 10V to 36V DC |
| Current Drawn | 3mA + Sensor Output (7-23mA) |

Working Equation: $H = (V)/50$ H- Relative Humidity (in %) V-Voltage received by ADC 0808 in mV

Sensor Resolution: 50 mV/%RH

## ADC Scaling Functions

The 8 bit ADC 0808 used in the design generates an output between 0 and 255d (FFh) for the voltage input received from the MAX472 connected to the sensors. To scale it to the values for Temperature, Pressure and Humidity, the scaling functions employed are:

| SENSOR | SCALING FUNCTION |
|---|---|
| WE700 Temperature Sensor | $ADC * \frac{46h}{FFh} - 14h$ |
| WE100 Pressure Sensor | $ADC * \frac{12Ch}{FFh} + 320h$ |
| WE600 Humidity Sensor | $ADC * 64h/FFh$ |

## I/O Interfacing

The design uses Fixed Addressing mode to interface the I/O devices to the addressing lines.

| IC CONNECTED | STARTING ADDRESS |
|---|---|
| **PROGRAMMABLE INTERRUPT CONTROLLER 8259** | **10h** |
| **PROGRAMMABLE PERIPHERAL INTERFACE 8255 – Display (2)** | **20h** |
| **PROGRAMMABLE PERIPHERAL INTERFACE 8255 – Sensor Input (1)** | **30h** |
| **PROGRAMMABLE INTERVAL TIMER 8254** | **40h** |

# Memory Interfacing

The design uses 6116 RAM chips and 2732 ROM chips to interface a total of 8k + 8k of ROM and 4k of RAM. ROM1 addressing starts at 00000h. The complete memory addressing is:

ROM1 - 00000h - 01FFFh

RAM1 - 02000h - 02FFFh

ROM2 - FE000h – FFFFFh

Both even and odd banks of memory have been incorporated in this design. The decoding logic is obtained from:

ROM1:

| 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

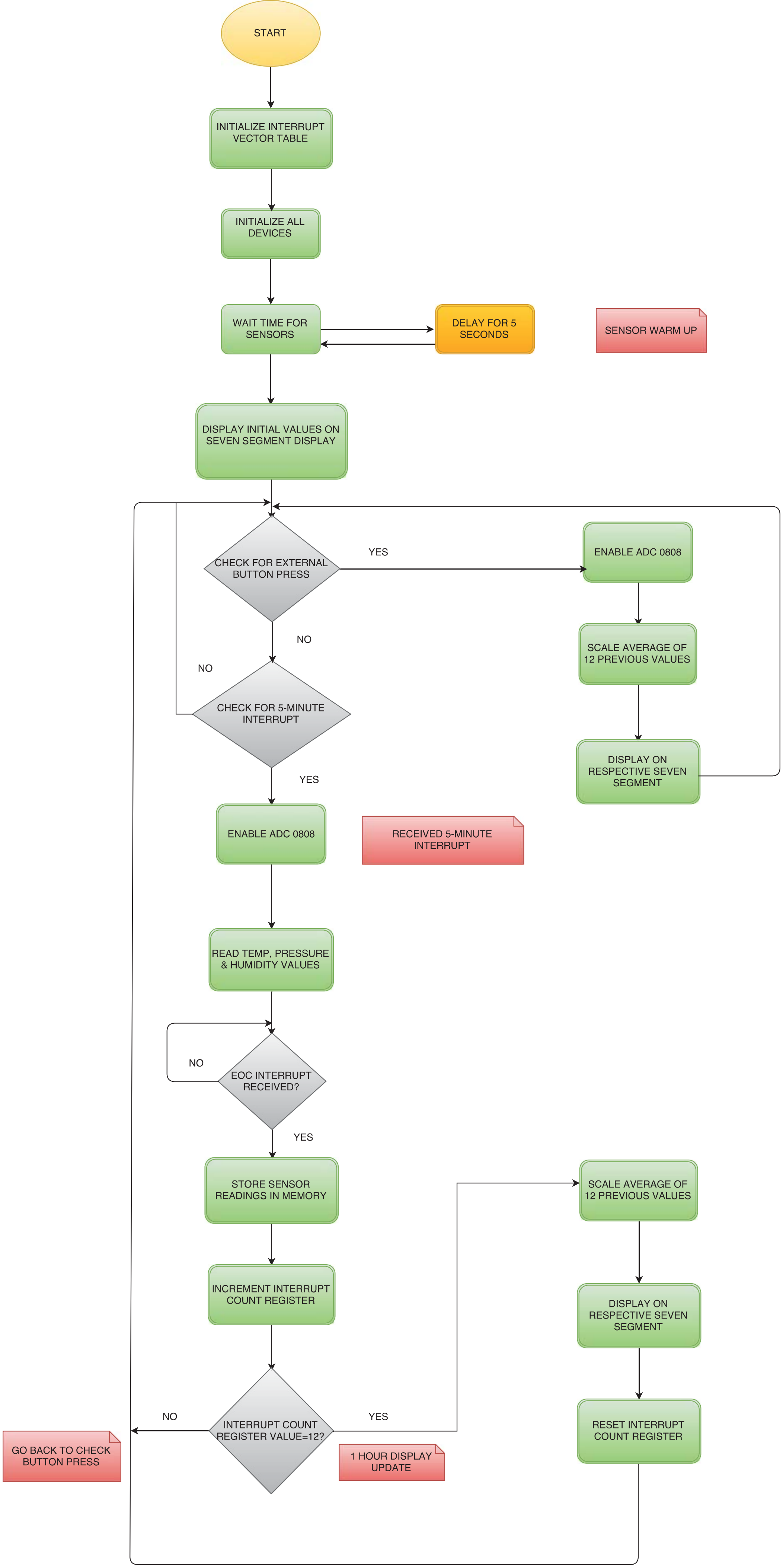
RAM1:

| 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

ROM2:

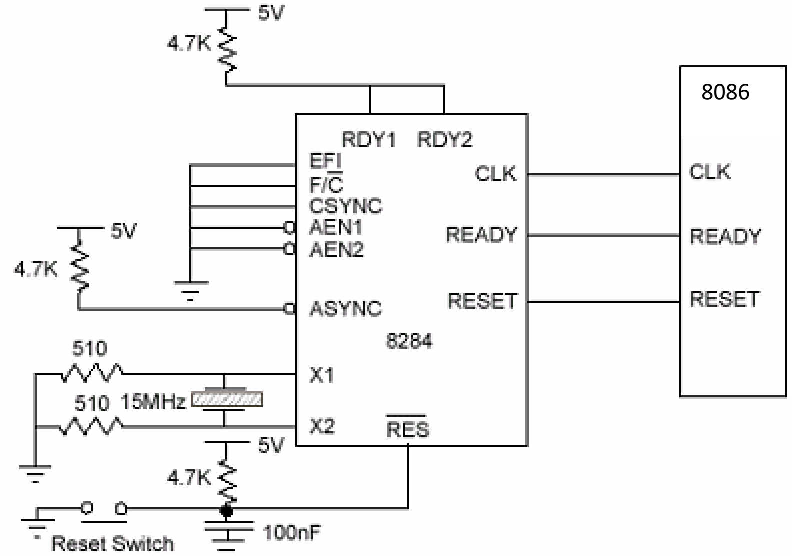
| 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Hence, to decode memory, we use bits A15, A14 and A13 of the address lines.

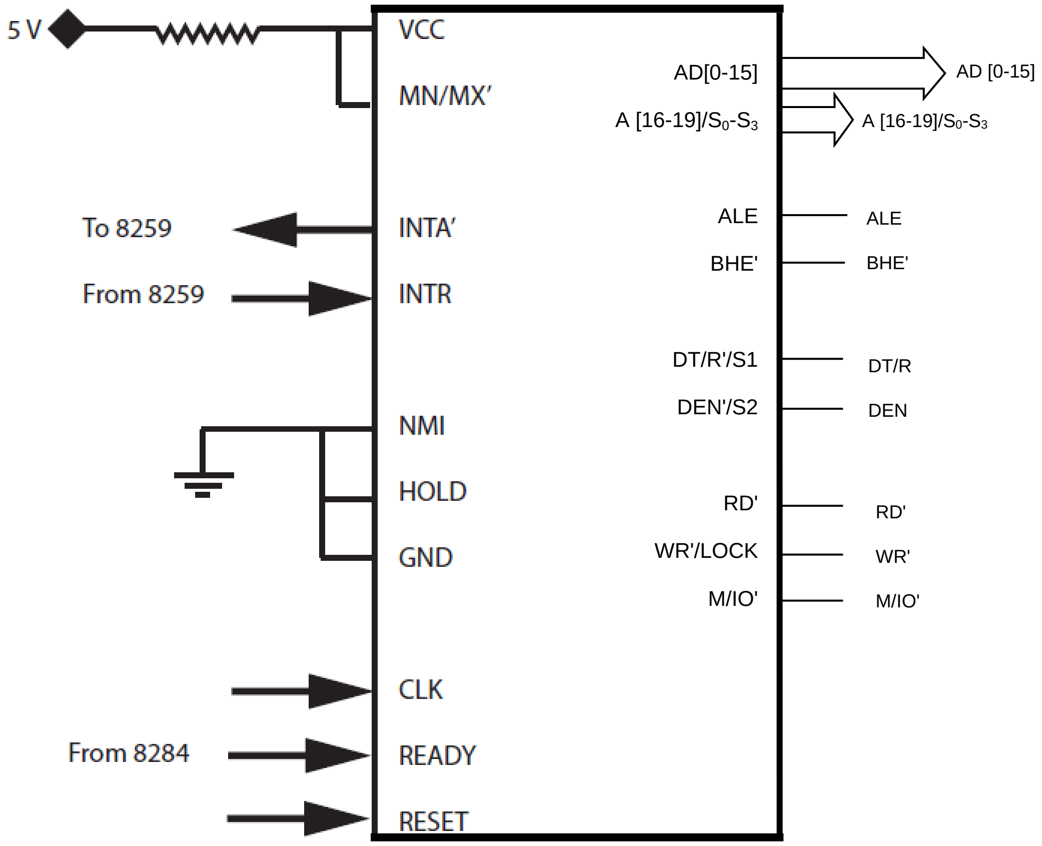

START
INITIALIZE INTERRUPT VECTOR TABLE
INITIALIZE ALL DEVICES
WAIT TIME FOR SENSORS
DELAY FOR 5 SECONDS
SENSOR WARM UP
DISPLAY INITIAL VALUES ON SEVEN SEGMENT DISPLAY
CHECK FOR EXTERNAL BUTTON PRESS
YES
ENABLE ADC 0808
SCALE AVERAGE OF 12 PREVIOUS VALUES
DISPLAY ON RESPECTIVE SEVEN SEGMENT
NO
NO
CHECK FOR 5-MINUTE INTERRUPT
YES
ENABLE ADC 0808
RECEIVED 5-MINUTE INTERRUPT
READ TEMP, PRESSURE & HUMIDITY VALUES
NO
EOC INTERRUPT RECEIVED?
YES
STORE SENSOR READINGS IN MEMORY
INCREMENT INTERRUPT COUNT REGISTER
SCALE AVERAGE OF 12 PREVIOUS VALUES
DISPLAY ON RESPECTIVE SEVEN SEGMENT
NO
INTERRUPT COUNT REGISTER VALUE=12?
YES
GO BACK TO CHECK BUTTON PRESS
1 HOUR DISPLAY UPDATE
RESET INTERRUPT COUNT REGISTER

# Hardware Circuit Diagrams

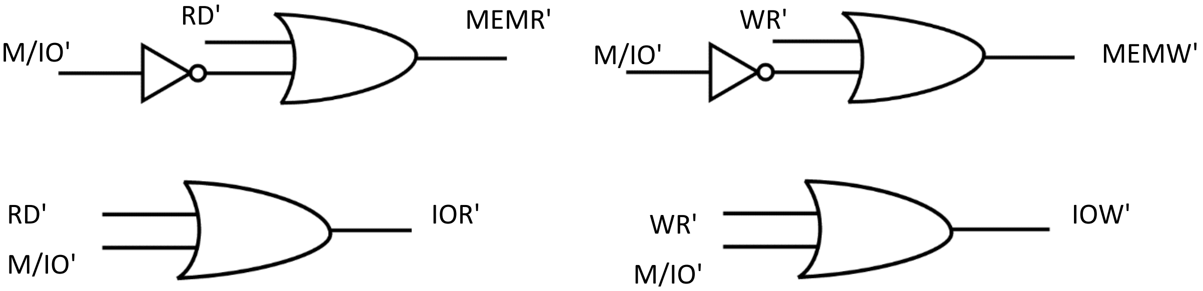
## Clock Generator



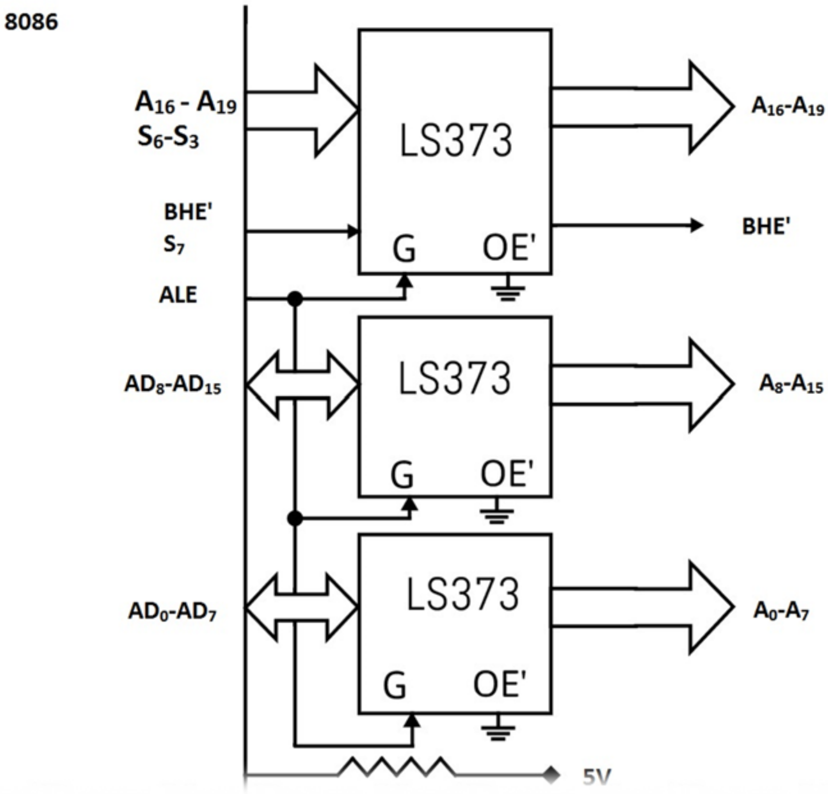
## 8086 – Pin out



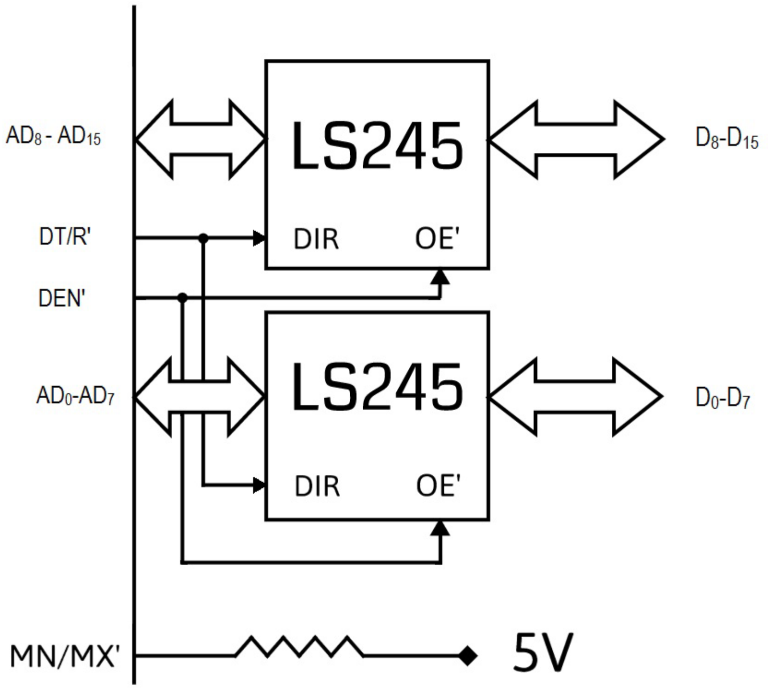
## System Bus – Control Signal



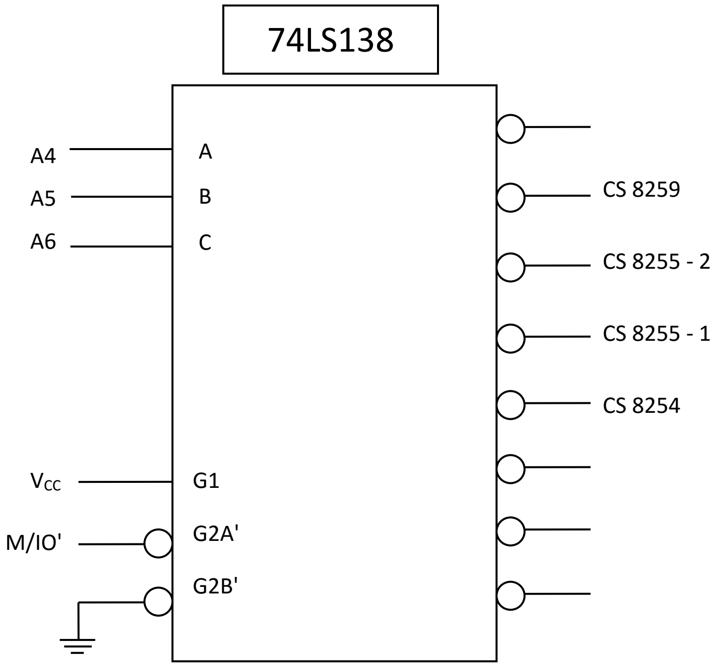
## System Bus - Address



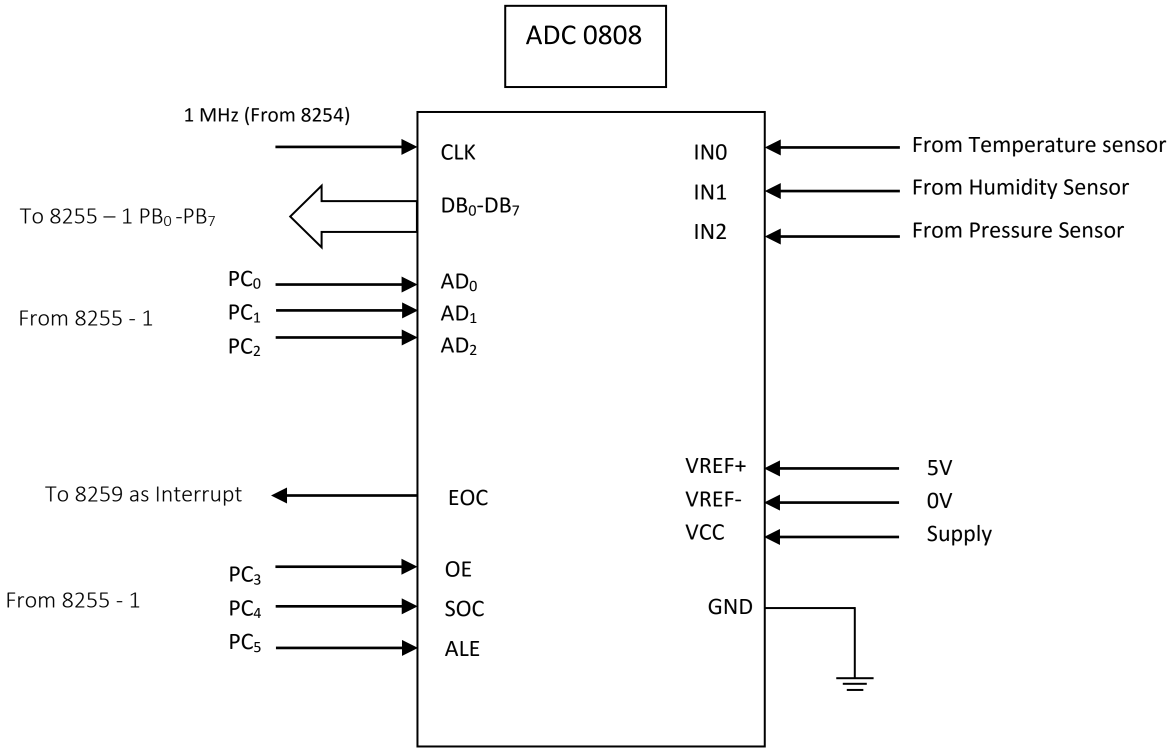
## System Bus - Data



## I/O Decoder

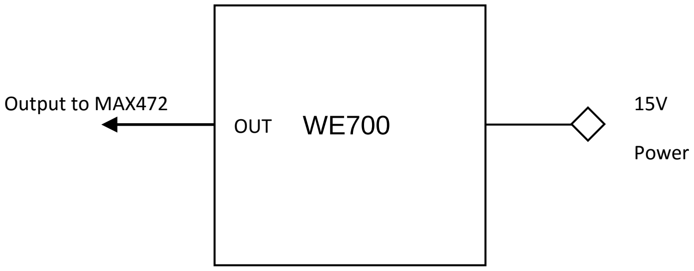


## ADC

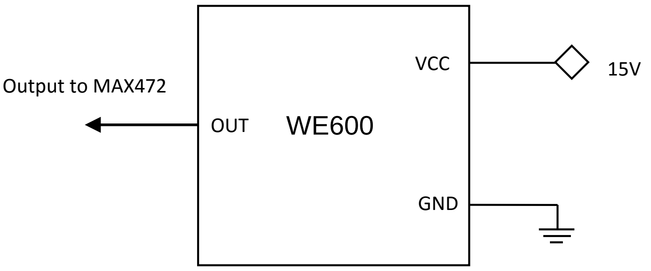


## Temperature, Pressure and Humidity Sensors

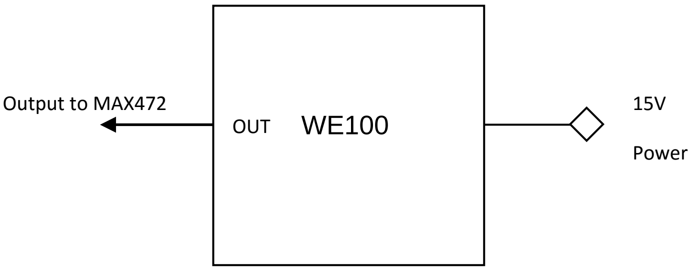
### Temperature Sensor WE700



### Humidity Sensor WE600

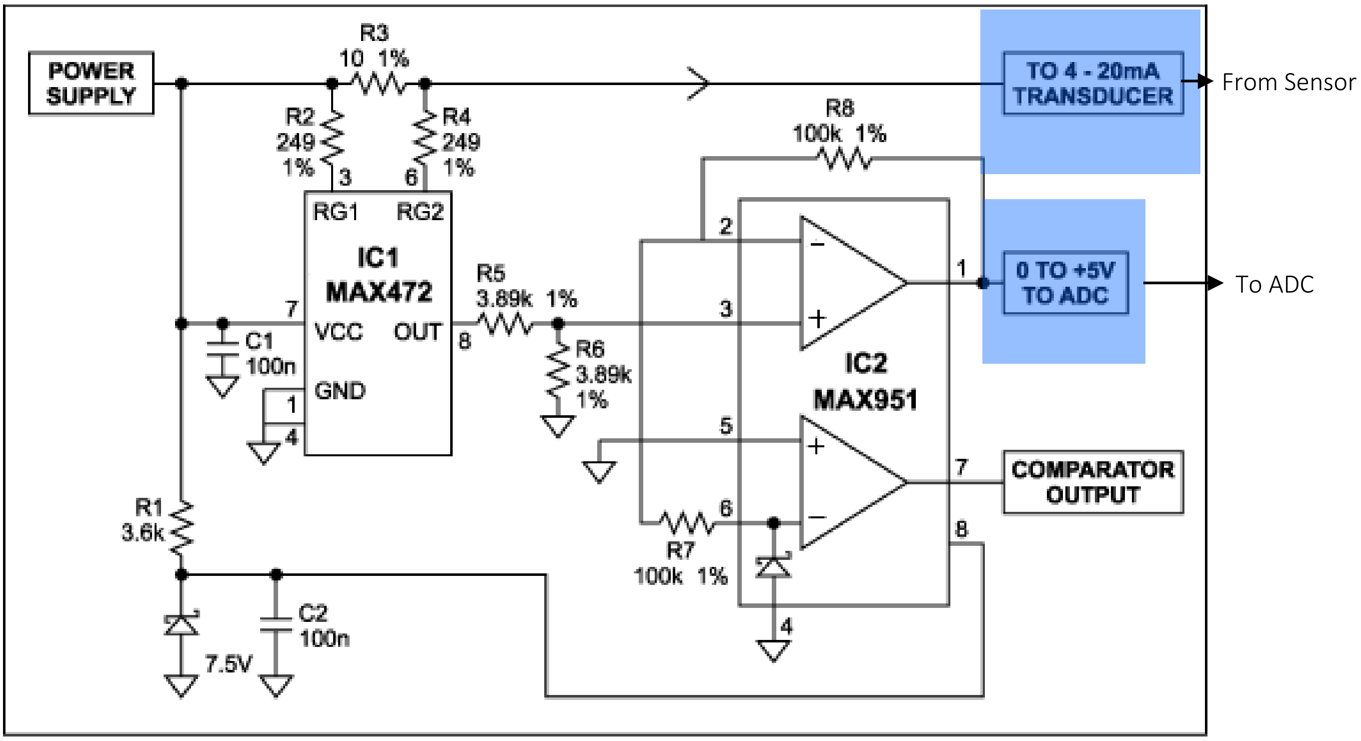


## Pressure Sensor WE100

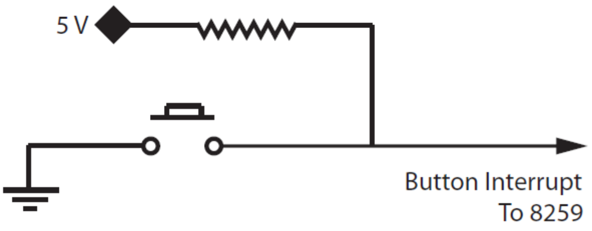


## MAX 472

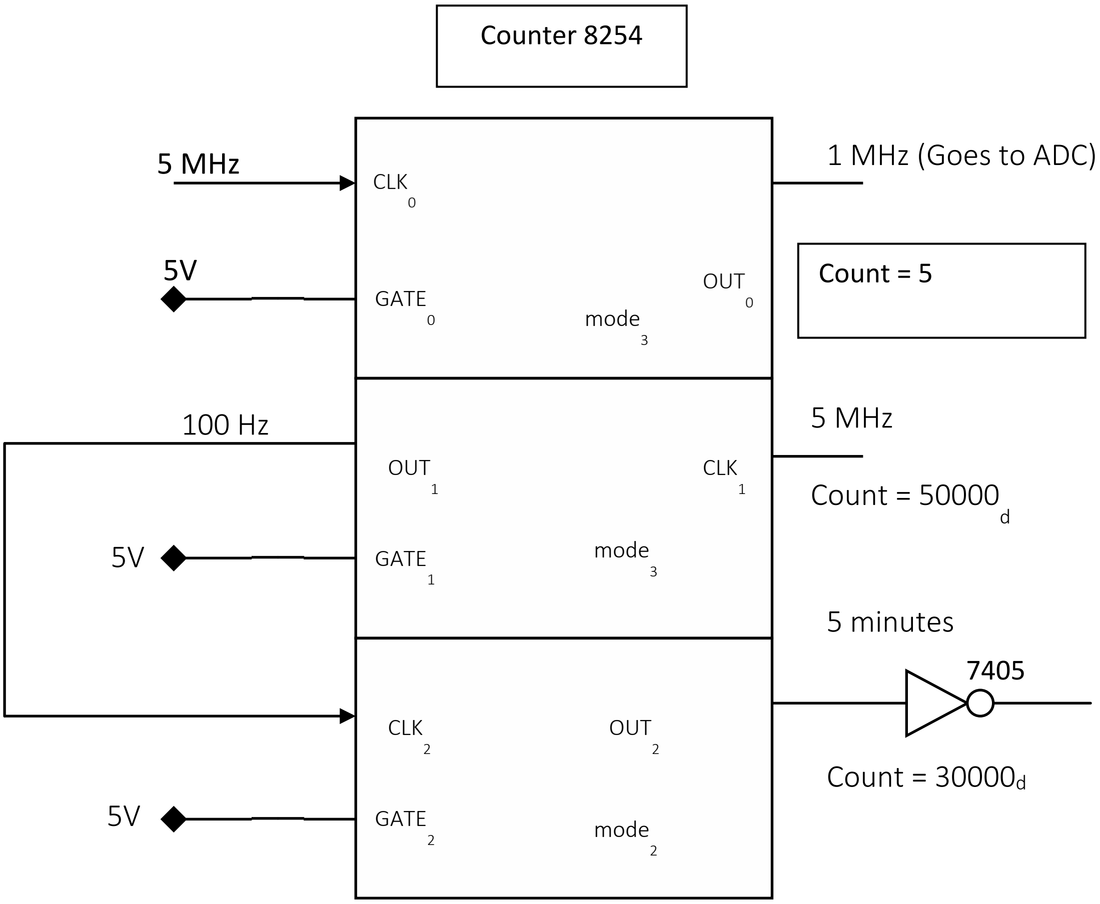
3 MAX 472s, one each connected to temperature, humidity and pressure sensor. Incoming current converted to Voltage between the range 0-5V and output to ADC.

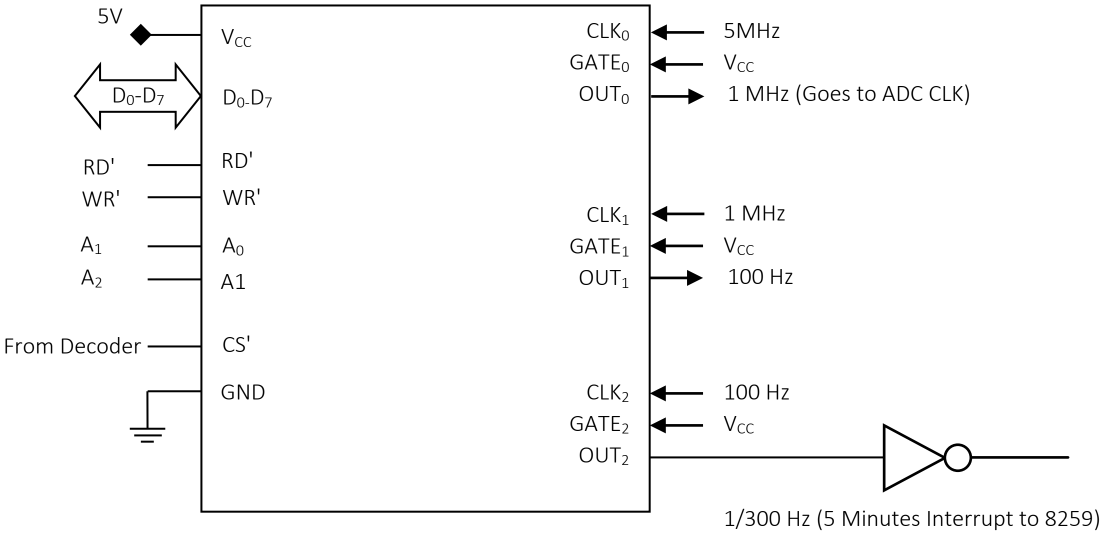


## Button

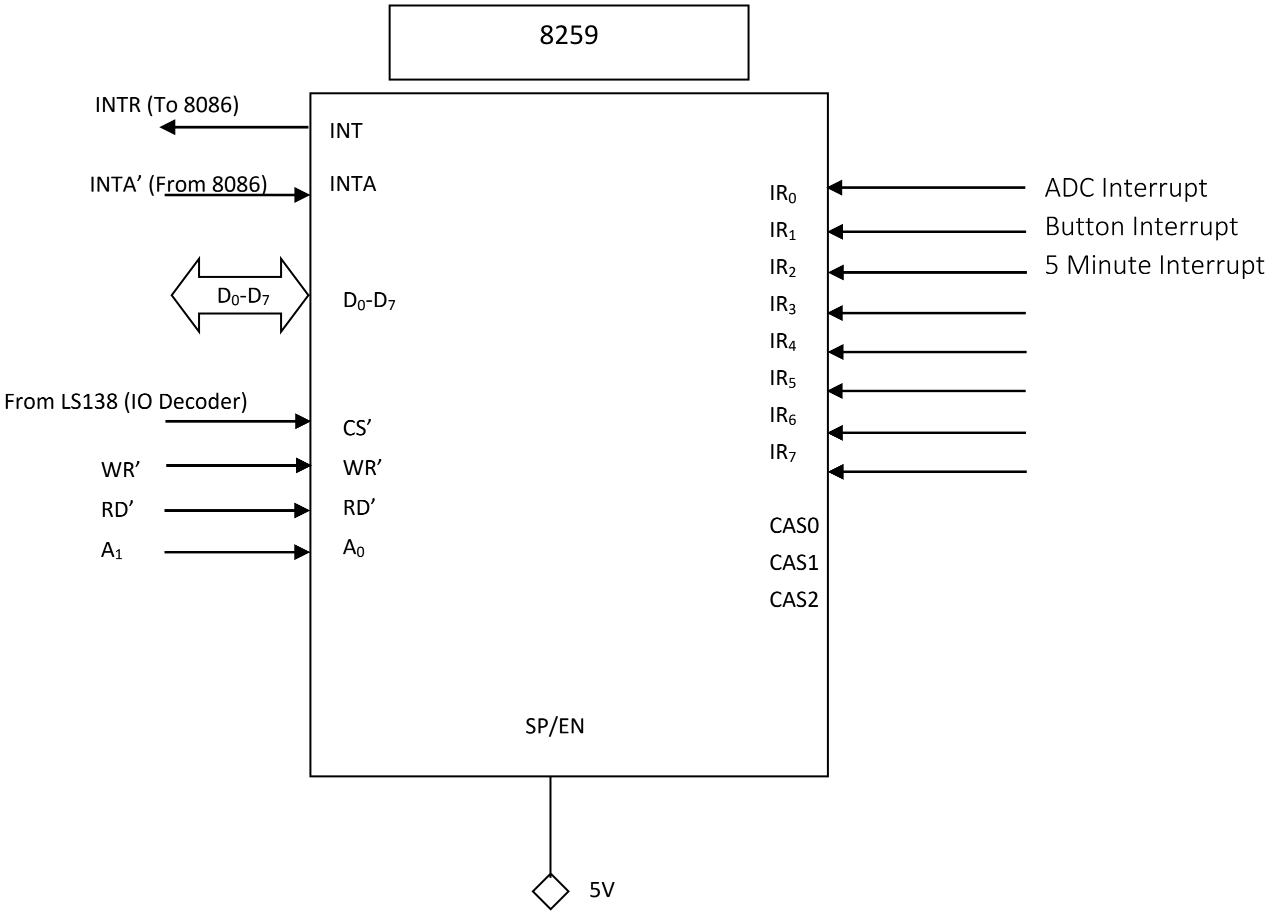


## Counter

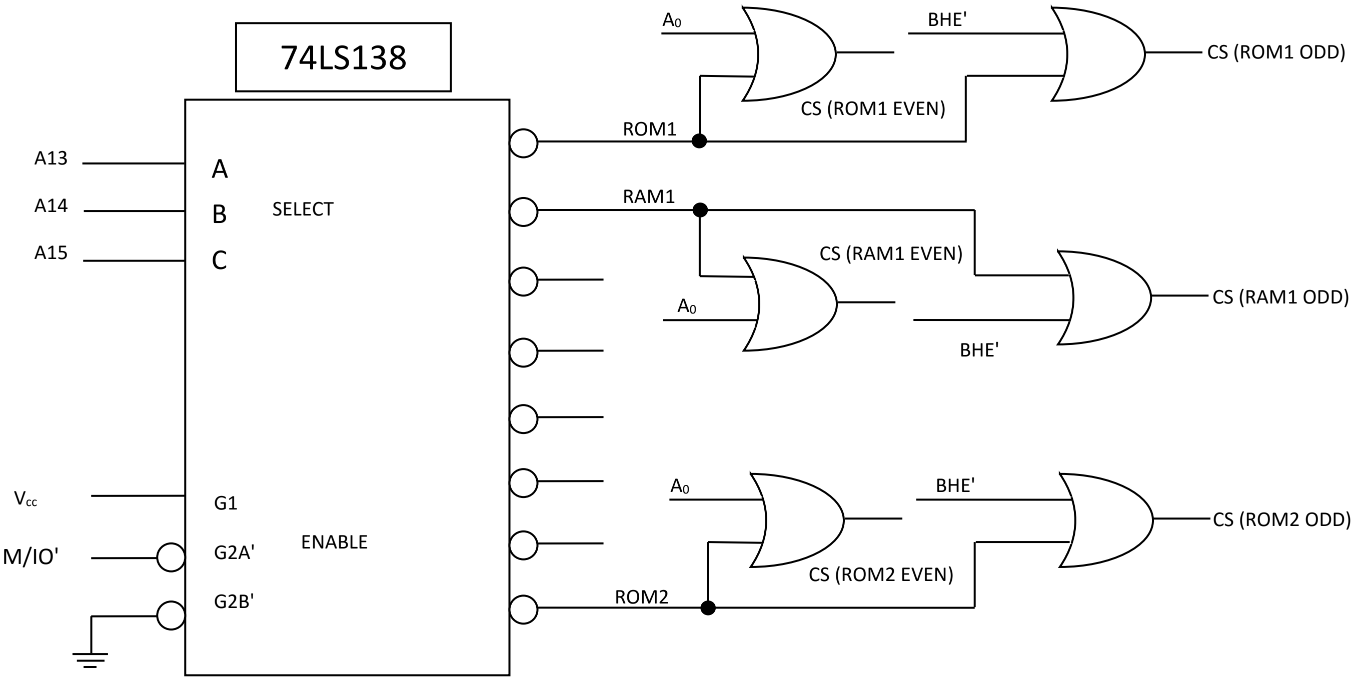




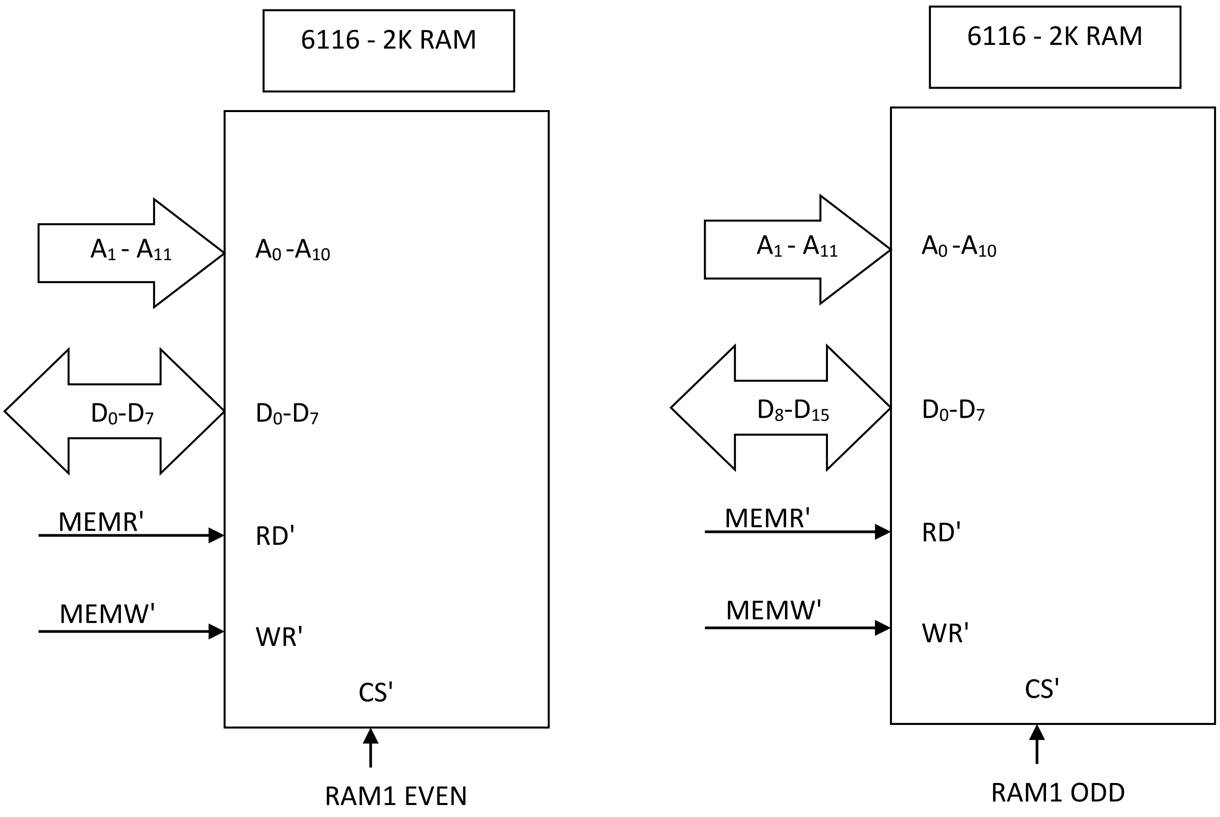
## 8259 – Priority Interrupt Controller

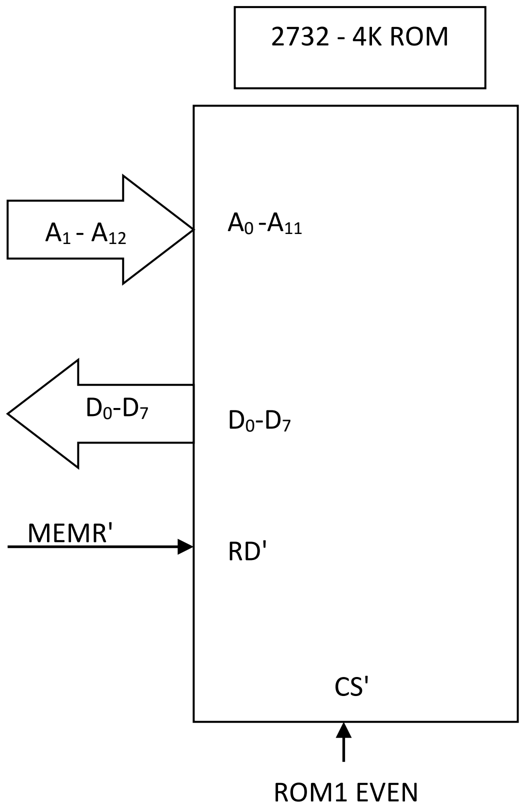


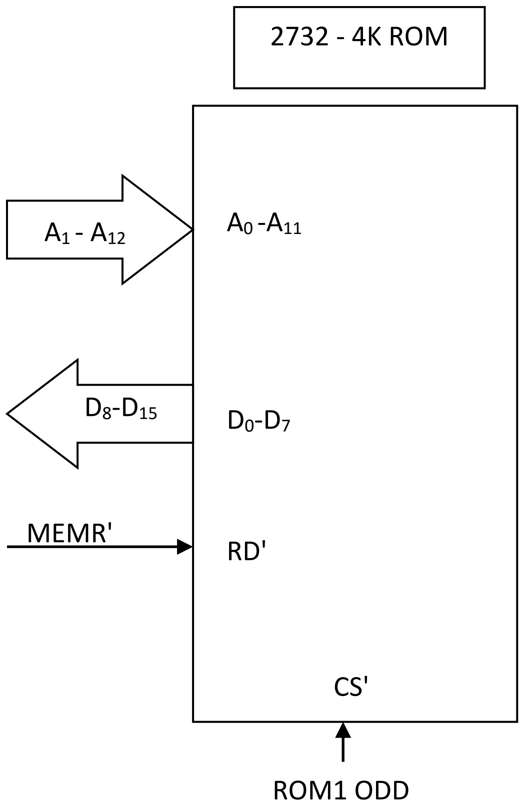
## Memory Interfacing

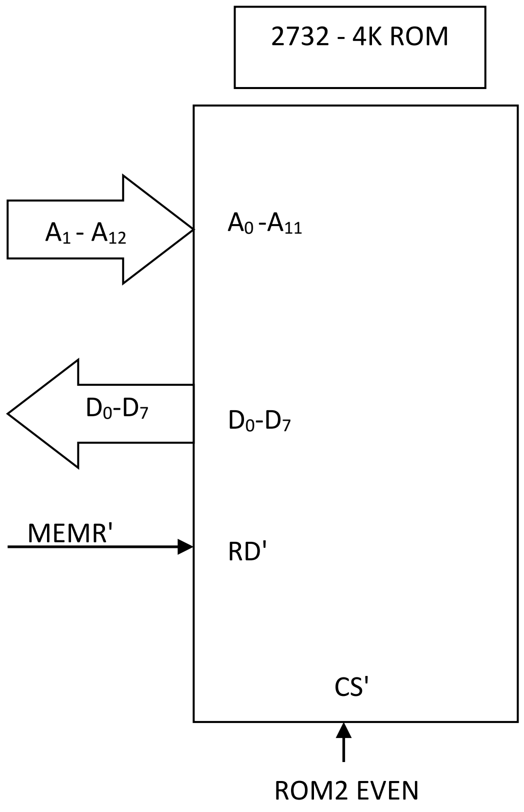


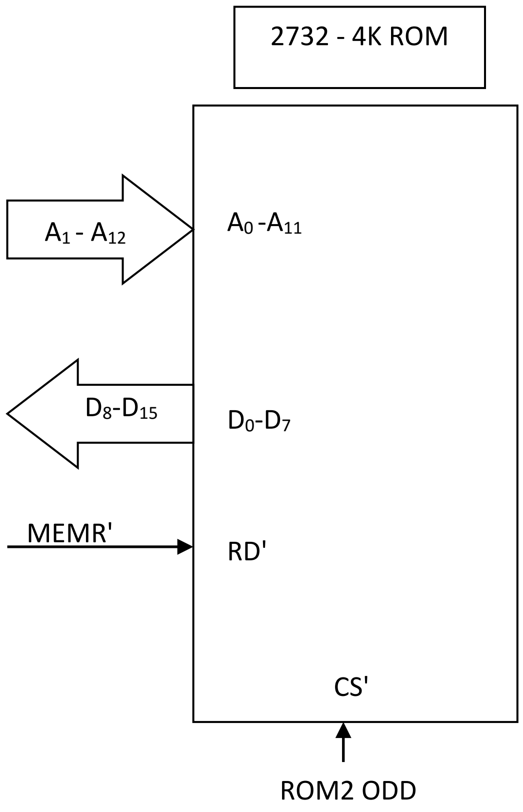
## Memory Chips



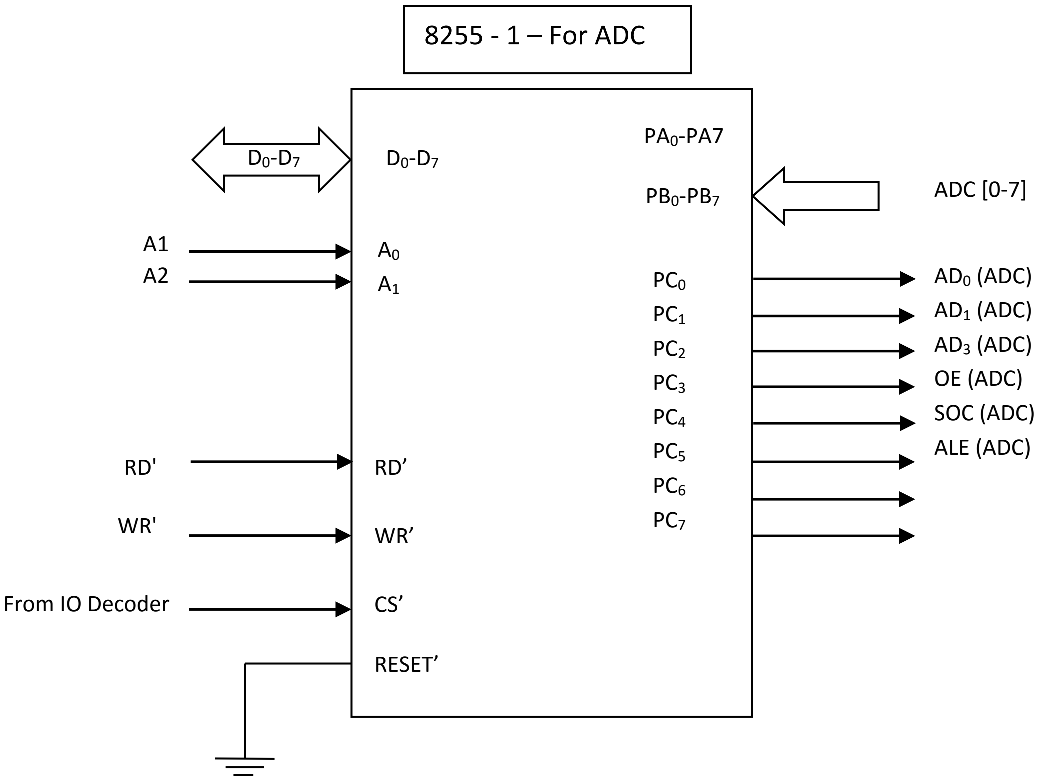

2732 - 4K ROM
$A_1$ - $A_{12}$
$A_0$ -$A_{11}$
$D_0$-$D_7$
$D_0$-$D_7$
MEMR'
RD'
CS'
ROM1 EVEN

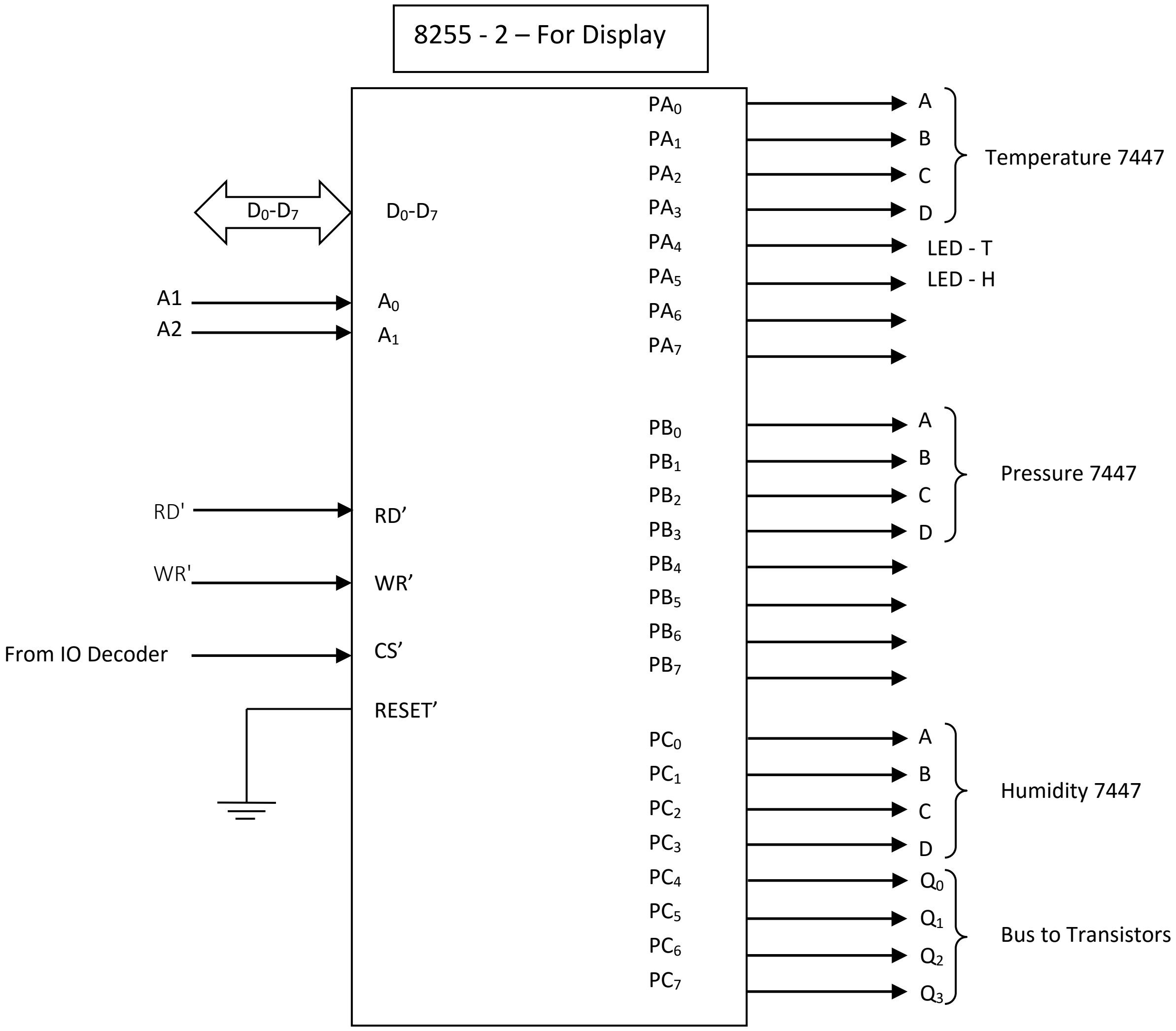

2732 - 4K ROM
$A_1$ - $A_{12}$
$A_0$ -$A_{11}$
$D_8$-$D_{15}$
$D_0$-$D_7$
MEMR'
RD'
CS'
ROM1 ODD


2732 - 4K ROM
$A_1$ - $A_{12}$
$A_0$ -$A_{11}$
$D_0$-$D_7$
$D_0$-$D_7$
MEMR'
RD'
CS'
ROM2 EVEN


2732 - 4K ROM
$A_1$ - $A_{12}$
$A_0$ -$A_{11}$
$D_8$-$D_{15}$
$D_0$-$D_7$
MEMR'
RD'
CS'
ROM2 ODD

# Interfacing I/O to Processor ADC and Display – 8255-1 and 8255-2

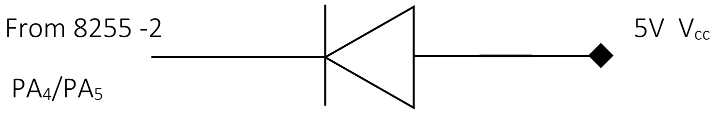




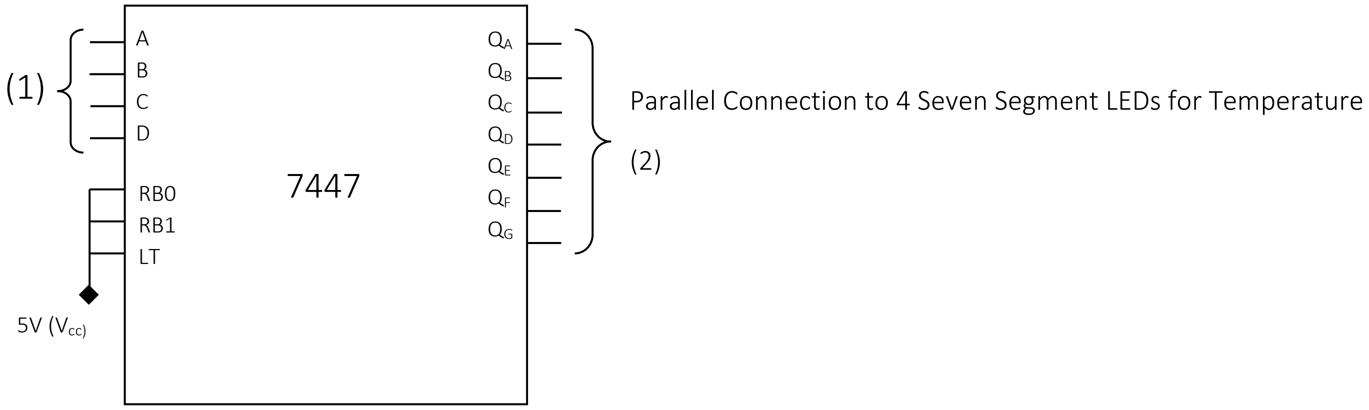
## LED

LED-H Humidity LED - To indicate humidity $R_h$ = 100%

LED-T Temperature LED - To indicate negative temperature



## 7447 BCD to Seven-Segment Decoder



3 7447s, 1 each for Temperature, Pressure and Humidity Display

(1) = $PA_0$-$PA_3$ - For Temperature Display

= $PB_0$-$PB_3$ - For Pressure Display

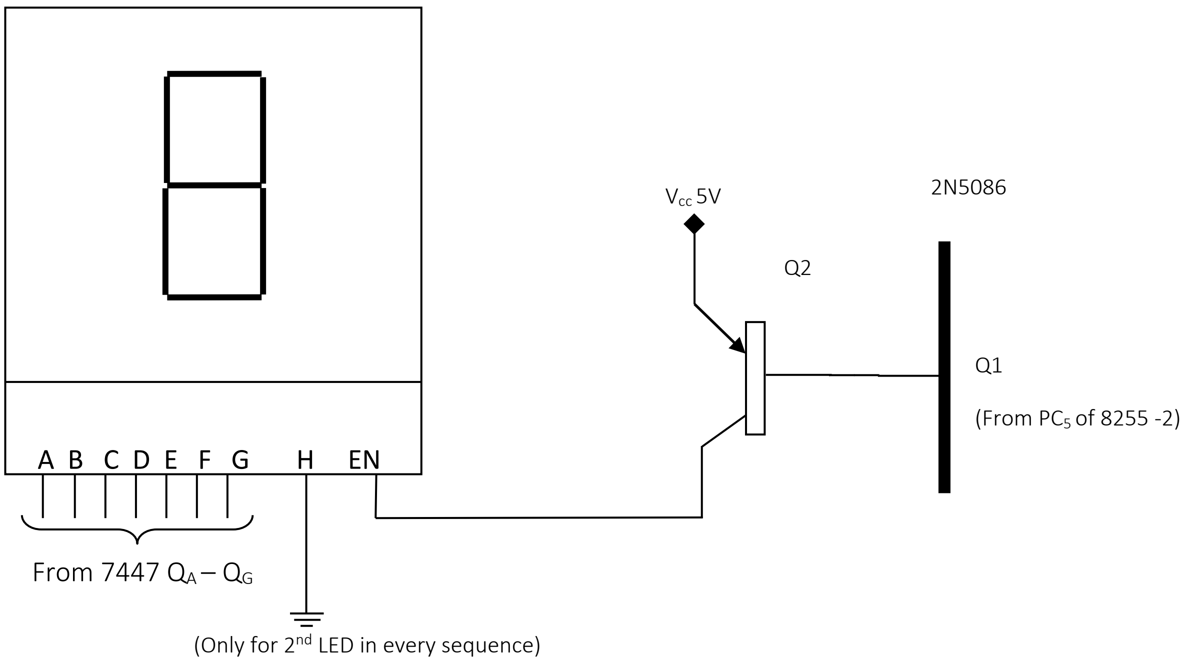
= $PC_0$-$PC_3$ - For Humidity Display

(2) Each 7447 is in parallel connection to 4 Seven Segment LED's. Connected to port A-G of seven segment LED.

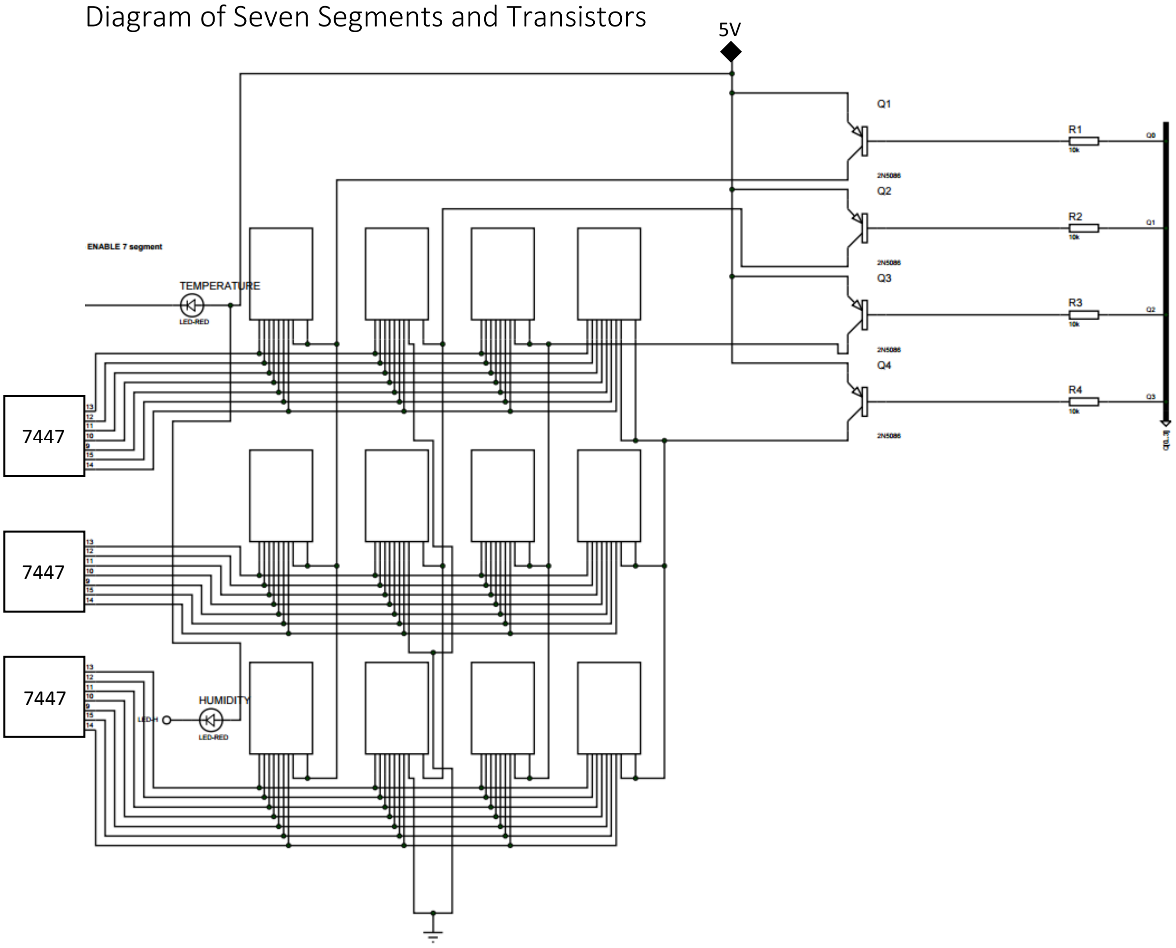
Total 12 Seven Segment LEDs. 4 for each parameter.

# Seven Segment LED and 2N5086 Transistor Amplifier

## 12 Seven Segment LEDs



## Diagram of Seven Segments and Transistors



# CITATIONS

Microprocessor Programming and Interfacing, BITSX/ CS F241 – DESIGN EXAMPLE OF WEATHER MONITORING SYSTEM (By Dr. K.R Anupama)

Microprocessor Programming and Interfacing, BITSX/ CS F241 – S8: Interfacing Input/ Output Devices (By Dr. K.R Anupama)

Microprocessor Programming and Interfacing, BITSX/ CS F241 – Proteus Demo Examples

Sensors WE700, WE100 and WE600 as mentioned in Design Example of Weather Monitoring System by Dr. K.R Anupama