

Lecture (09)

PIC16F84A

LCD interface

Dr. Ahmed M. ElShafee

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

-
- PIC16F84A LCD interface
 - Assignment 01, 4 Zones fire controller board
 - Assignment 02, automatic water tank controller

LCD

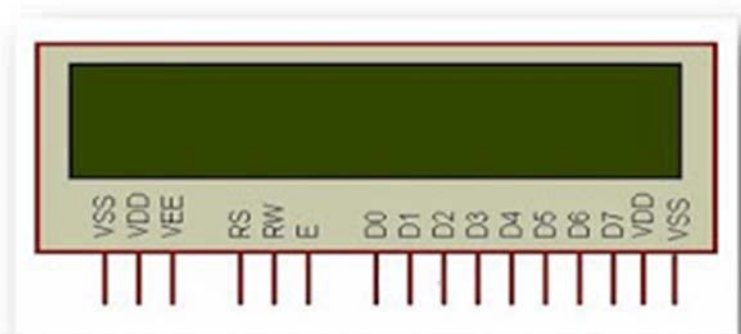
- LCD (Liquid Crystal Display) is used in all the electronics projects to display the status of the process.
- A 16x2 alphanumeric LCD is most widely used module of LCD nowadays.
- 16x2 LCD has 2 horizontal line which comprising a space of 16 displaying character. It has two type of register inbuilt that is
 - Command Register
 - Data Register.



۳

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

-
- Command register is used to insert a special command into the LCD.
 - While Data register is used to insert a data into the LCD.
 - Command is a special set of data which is used to give the internal command to LCD like Clear screen, move to line 1 character 1, setting up the cursor etc.

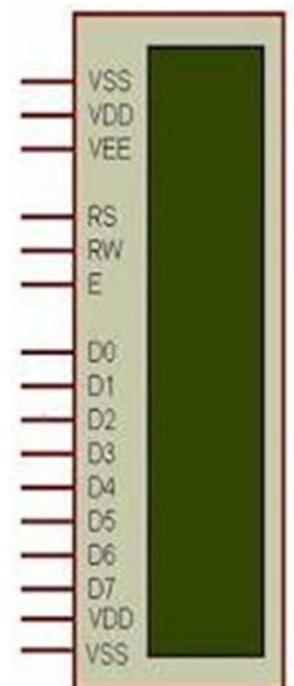


۴

Dr. Ahmed ElShafee,

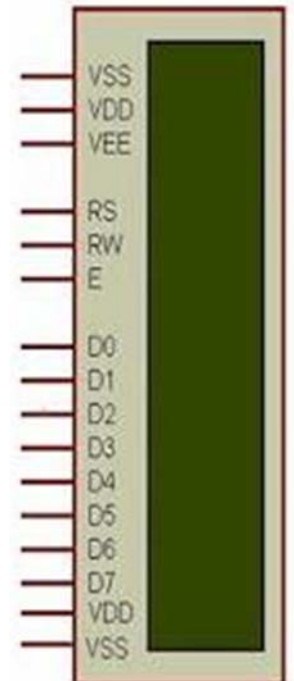
Pin No.	Pin Description
Pin 1 (GND)	This is a ground pin to apply a ground to LCD.
Pin 2 (VCC)	This is the supply voltage pin to apply voltage to LCD.
Pin 3 (VEE)	This is the pin for adjusting a contrast of the LCD display by attaching a variable resistor in between VCC and GND.
Pin 4 (RS)	RS stands for Register Select. This pin is used to select command/data register. If RS=0 then command register is selected. If RS=1 then data register is selected.
Pin 5 (R/W)	R/W stands for Read/Write. This pin is used to select the operation Read/Write. If R/W=0 then Write operation is performed. If R/W=1 then Read operation is performed.
Pin 6 (EN)	En stand for Enable signal. A positive going pulse on this pin will perform a read/write function to the LCD.
Pin 7-14 (DB0-DB7)	This 8 pin is used as a Data pin of LCD.
Pin 15 (LED+)	This pin is used with pin 16(LED-) to setting up the illumination of back light of LCD. This pin is connected with VCC.
Pin 16 (LEC-)	This pin is used with pin 15(LED+) to setting up the illumination of back light of LCD. This pin is connected with GND.

Hex Code	Command to LCD instruction Register
01	Clear display screen
02	Return home
04	Decrement cursor (shift cursor to left)
06	Increment cursor (shift cursor to right)
05	Shift display right
07	Shift display left
08	Display off, cursor off
0A	Display off, cursor on
0C	Display on, cursor off
0E	Display on, cursor blinking
0F	Display on, cursor blinking
10	Shift cursor position to left
14	Shift cursor position to right
18	Shift the entire display to the left
1C	Shift the entire display to the right
80	Force cursor to beginning to 1st line
C0	Force cursor to beginning to 2nd line
38	2 lines and 5x7 matrix



Initializing LCD

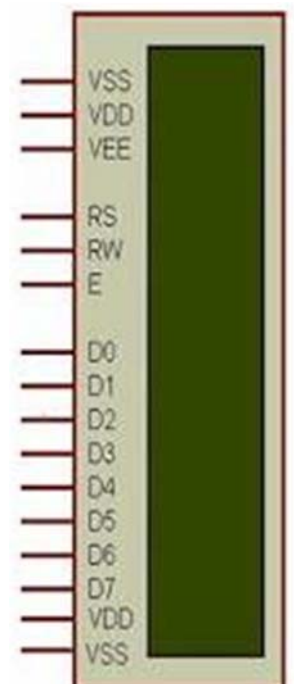
- en = 0
- RS = 0 // command register
- Data line = 0x03 // attention command
- EN =1, EN =0 + delay // execute command
- EN =1, EN =0 + delay // execute command
- EN =1, EN =0 + delay // execute command
- Data line = 0x04 // 4 bits mode
- EN =1, EN =0 + delay // execute command



v

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

- ***** starting from here commands
will be sent in 4 bits mode *****
- Write command: 0x28 // 2 lines 7x5 per character
Write command: 0x0f // Display on, cursor blinking
- Write command: 0x06// Increment cursor (shift cursor to right)
- Write command: = 0x80// Force cursor to beginning to 1st line

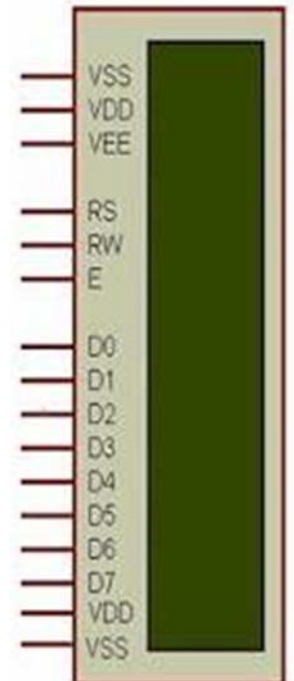


^

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

Send commands in 4 bits mode

- En =0 //disable LCD
- D0-D3 = higher nibble of command
- RS = 0 // command register
- RW = 0 // write mode
- Delay + Enable =1, enable =0 // execute command
- D0-D3 = lower nibble of command
- RS = 0 // command register
- RW = 0 // write mode
- Delay + Enable =1, enable =0 // execute command

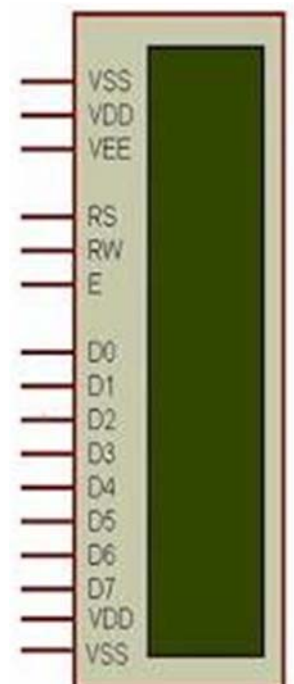


9

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

Send data in 4 bits mode

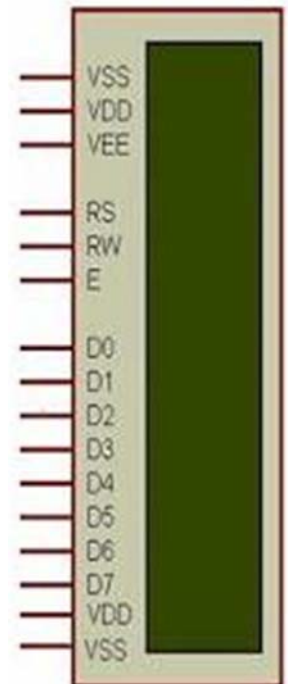
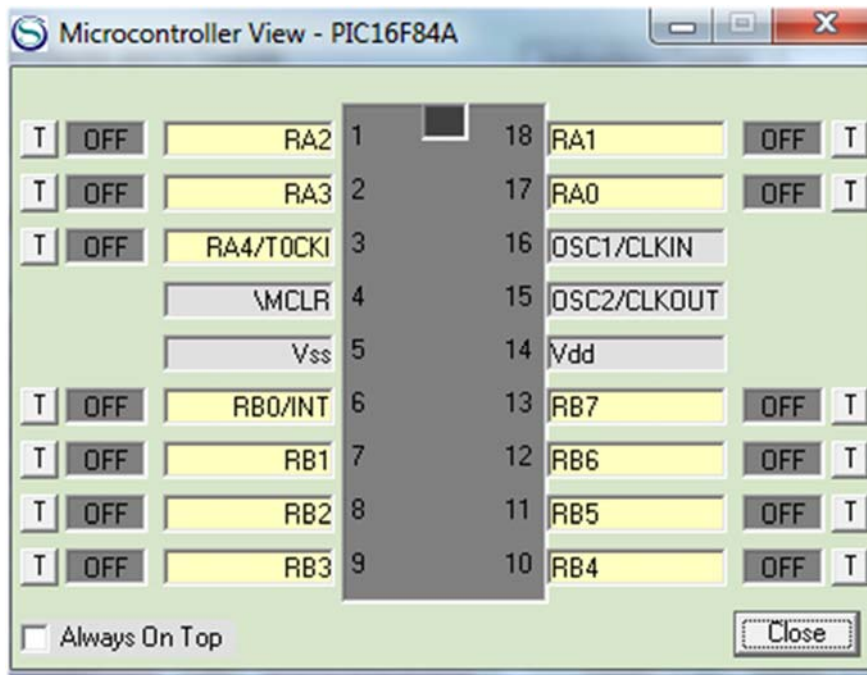
- En =0 //disable LCD
- D0-D3 = higher nibble of data
- RS = 1 // data register
- RW = 0 // write mode
- Delay + Enable =1, enable =0 // write data
- D0-D3 = lower nibble of data
- RS = 1 // data register
- RW = 0 // write mode
- Delay + Enable =1, enable =0 // write data



10

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

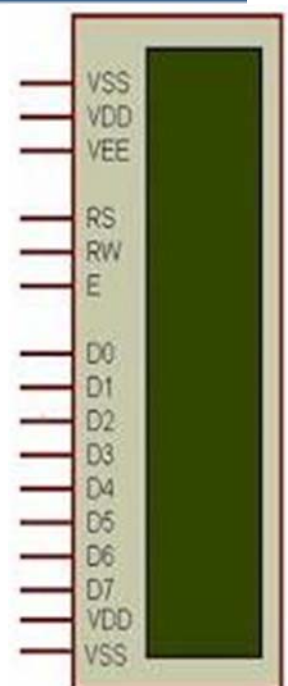
Pin configuration



11

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

LCD	Other	PIC16F84A
1 (Vss)	Gnd	
2 (Vdd)	5V	
3 (Vee)	Var resistor	
4 (RS)		RB4
5 (RW)		RB5
6 (EN)		RB6
7 (D0)	Gnd	
8 (D1)	Gnd	
9 (D2)	Gnd	
10 (D3)	Gnd	
11 (D4)		RB0
12 (D5)		RB1
13 (D6)		RB2
14 (D7)		RB4
15 (Vdd)	R100 \leftrightarrow 5V	
16 (Vss)	Gnd	



11

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

Lcd.h

```
#include <xc.h>
#include <pic16f84a.h>
#define LCD_RS RB4
#define LCD_RW RB5
#define LCD_EN RB6

void display_string(char str[]);
void send_data(char dataout);
void display_char(char data_value);
void send_cmd(char cmdout);
void write_cmd(char cmd_value);
void INIT_LCD(void);
void INIT_HW(void);
```

Lcd.c

```
#include <xc.h>
#include <pic16f84a.h>
#include <string.h>
#include "lcd.h"
void send_data(char dataout)
{
    PORTB = dataout;
    LCD_RS = 1 ;
    LCD_RW = 0 ;
    LCD_EN = 1;
    LCD_EN=0;
}
void display_string(char str[])
{
    for(int n=0;n<strlen(str);n++)
    {
        display_char(str[n]);
    }
}

void display_char(char data_value)
{
    char data_value1;
    data_value1 = ((data_value >> 4) & 0x0F);
    //mask lower nibble
    send_data(data_value1);
    _delay(50) ;
    data_value1 = (data_value & 0x0F); //
    mask higher nibble
    send_data(data_value1);
    _delay(50) ;
}
```

```

void send_cmd(char cmdout)
{
PORTB=cmdout;
LCD_RS = 0 ;
LCD_RW = 0 ;
LCD_EN = 1;
LCD_EN=0;
}
void write_cmd(char cmd_value)
{
char cmd_value1;
cmd_value1 = ((cmd_value >> 4) & 0x0F) ;
//mask lower nibble
send_cmd(cmd_value1); // send to LCD

_delay(50) ;
cmd_value1 = (cmd_value & 0x0F) ; //
mask higher nibble
send_cmd(cmd_value1); // send to LCD
_delay(50) ;
}

```

```

void INIT_LCD(void){
unsigned char i = 0 ;
LCD_RS = 0; // write control bytes
_delay(15000) ; // DelayMs(15); // power on delay
PORTB = 0x3; // attention!
LCD_EN = 1;
LCD_EN=0;
_delay(5000) ; // DelayMs(5);
LCD_EN = 1;
LCD_EN=0;
_delay(100) ; // DelayUs(100);
LCD_EN = 1;
LCD_EN=0;
_delay(5000) ; // DelayMs(5);
PORTB = 0x2; // set 4 bit mode
LCD_EN = 1;
LCD_EN=0;
_delay(40) ; // DelayUs(40);
write_cmd(0x28); //to initialize LCD in 2 lines, 5X7 dots and 4bit mode.
_delay(5000) ;
write_cmd(0x0F);
_delay(5000) ;
write_cmd(0x06);
write_cmd(0x80);
_delay(5000) ;
}

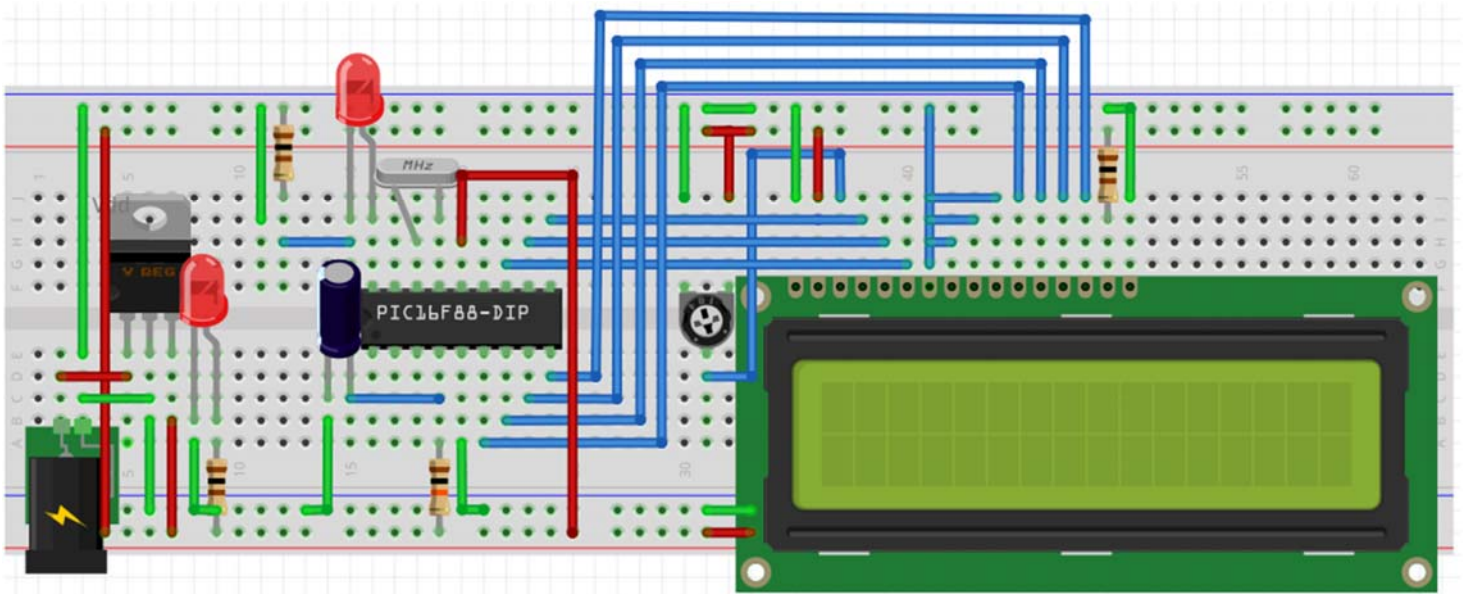
```

```

void INIT_HW(void)
{
TRISB = 0x00 ;
PORTB = 0xff ;
}

```


HelloWorldLCD



17

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

Newfile.c

```
#include <xc.h>
#include <pic16f84a.h>
#include "lcd.h"
#include "config.h"
void delay_ms(int x);
void main (){
    INIT_HW();
    TRISA = 0x00 ;
    RA2=1;
    INIT_LCD();
    write_cmd(0x80) ;
    display_string("Welcome to");
    write_cmd(0xC0) ;
    display_string("LCD world");
    while(1)
    {
        RA2=~RA2;
        delay_ms(500);
    }
}
```

```
void delay_ms(int x)
{
    for(int n=0;n<x;n++)
    {
        _delay(1000);
    }
}
```

CONFIG.h

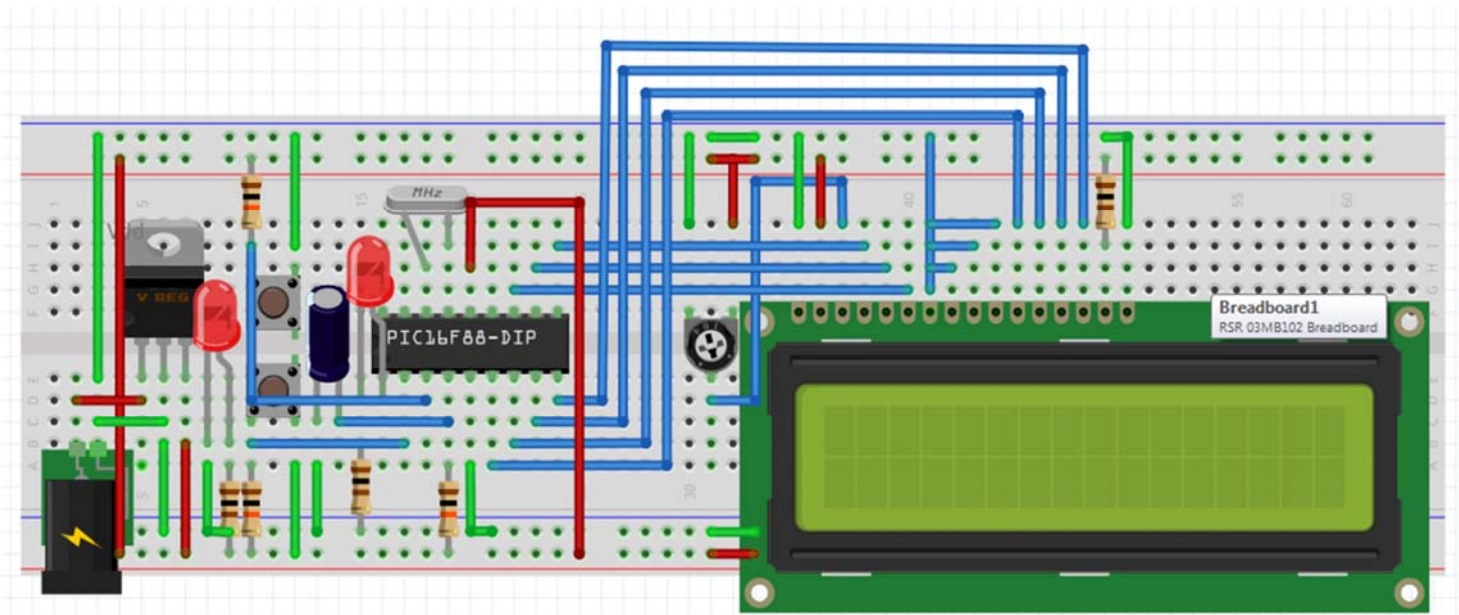
```
#pragma config FOSC = XT      //  
Oscillator Selection bits (XT oscillator)  
#pragma config WDTE = OFF    //  
Watchdog Timer (WDT disabled)  
#pragma config PWRTE = OFF   //  
Power-up Timer Enable bit (Power-up  
Timer is disabled)  
#pragma config CP = OFF      // Code  
Protection bit (Code protection disabled)
```

Assignment1: 2PressesControlledCountersWithLC

D

- You are required to write a program that display two different counters on LCD.
- Each counter is displayed on one line of LCD.
- Each counter is controlled using a press, that counter increment s after pressing its press for convenient period than updates counter on LCD screen.
- Counter overflow after exceeding 1000
- Attached on website
 - video for simulator LCD setup
 - Video for microcontroller operation
 - Empty project contains LCD library

2 Presses Controlled Counters With LCD



٢١

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

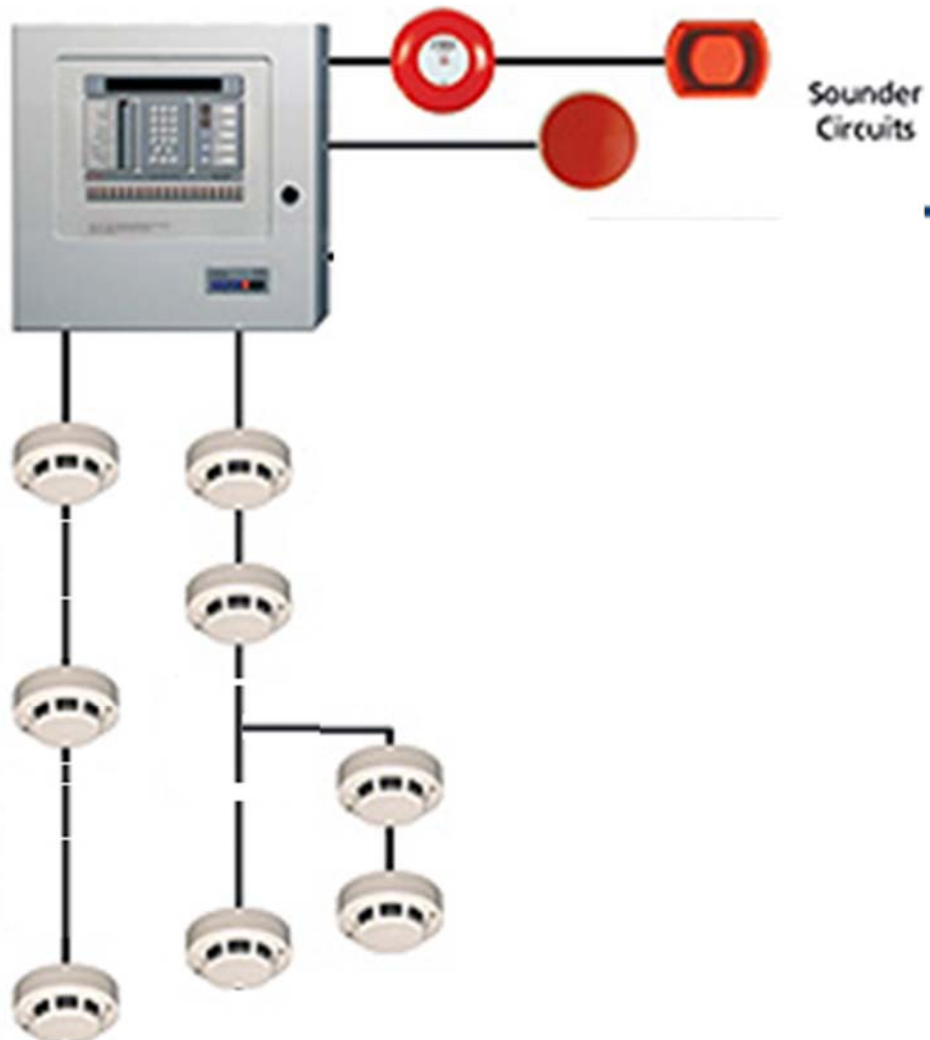
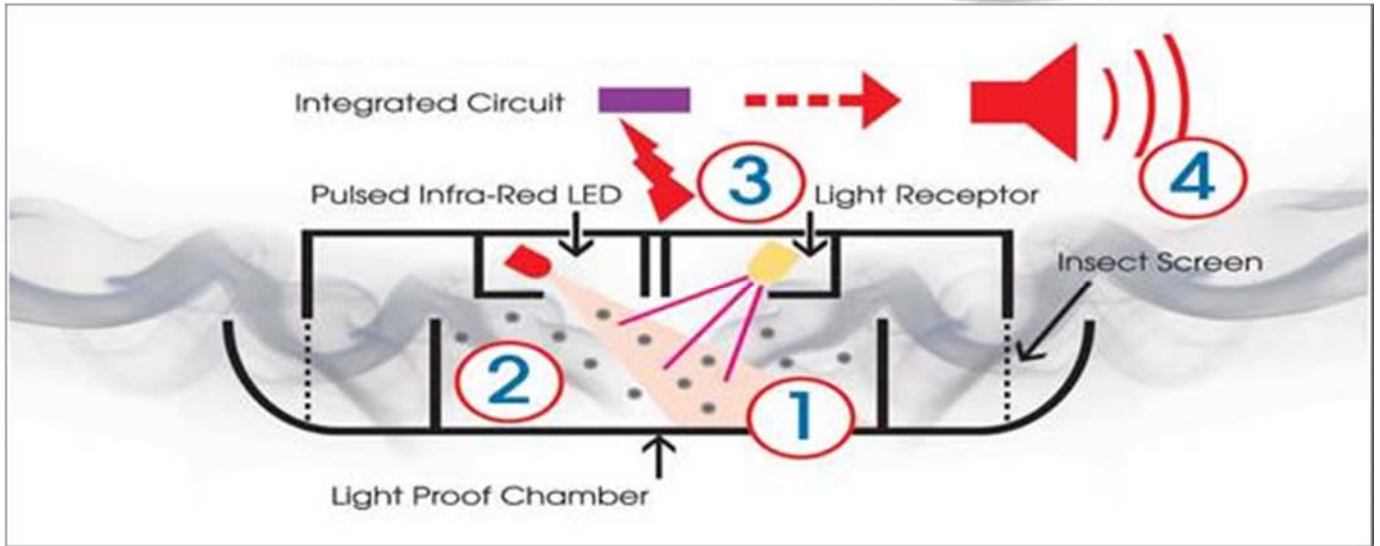
Assignment 02, 4 Zones fire controller board

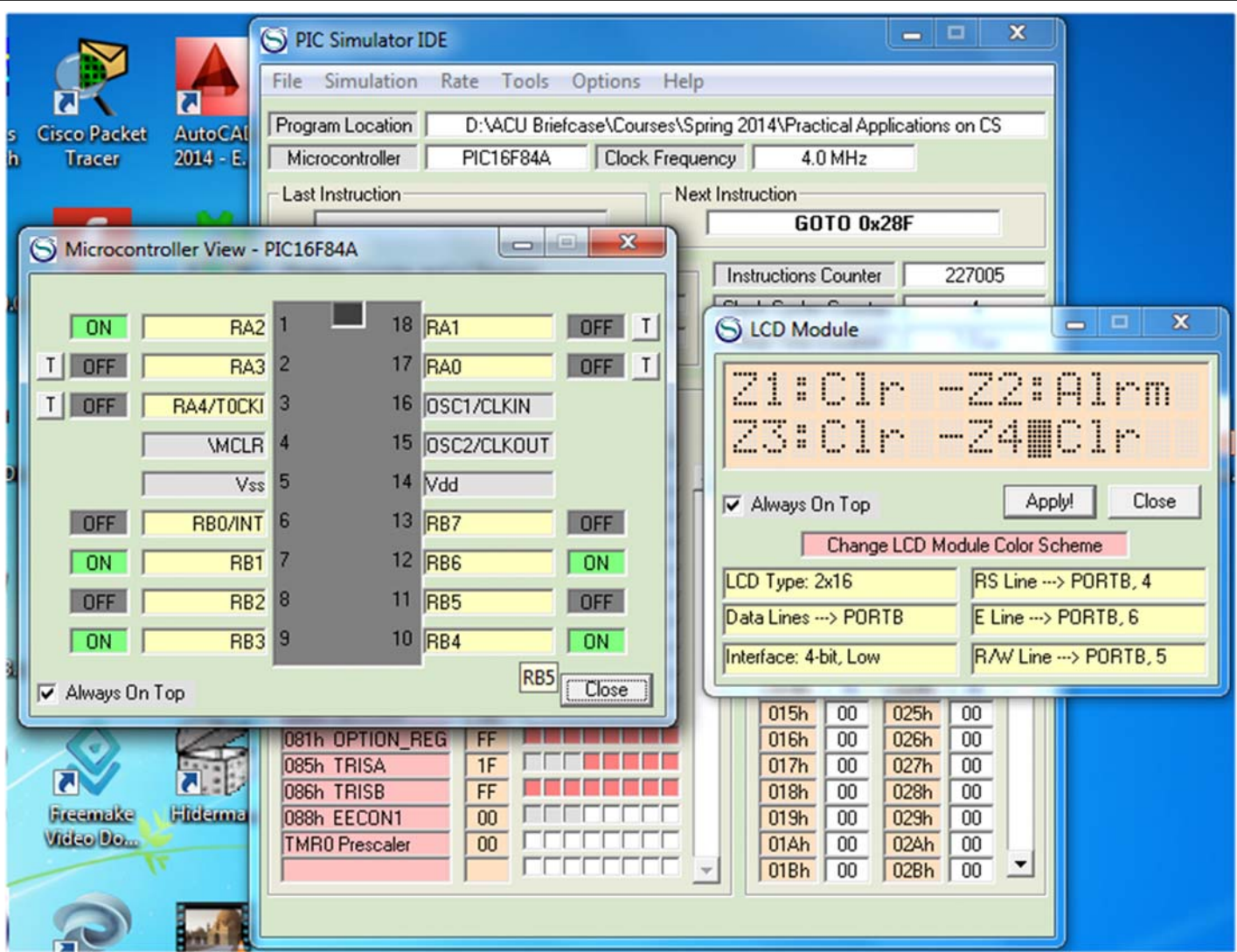
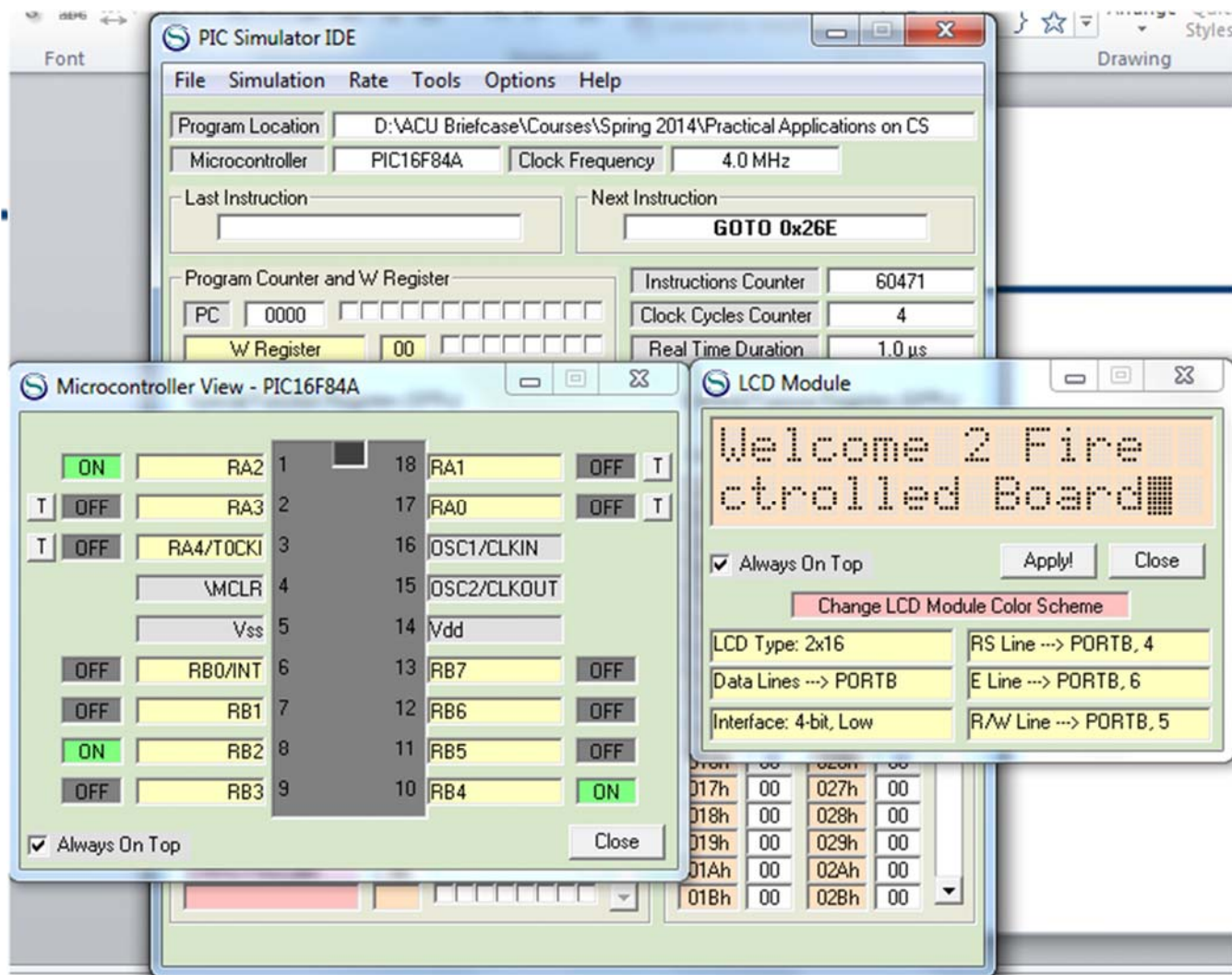
- Build a fire controller board that uses free contact fire sensors
- On start up controller display it's version and manufacturer for a while then start it's operation
- Controller always check status of all zones and display zones status on LCD
- If one or more zones are activated t triggers a buzzer
- Controller doesn't release buzzer till all zones are clear
- Attached video for microcontroller operation

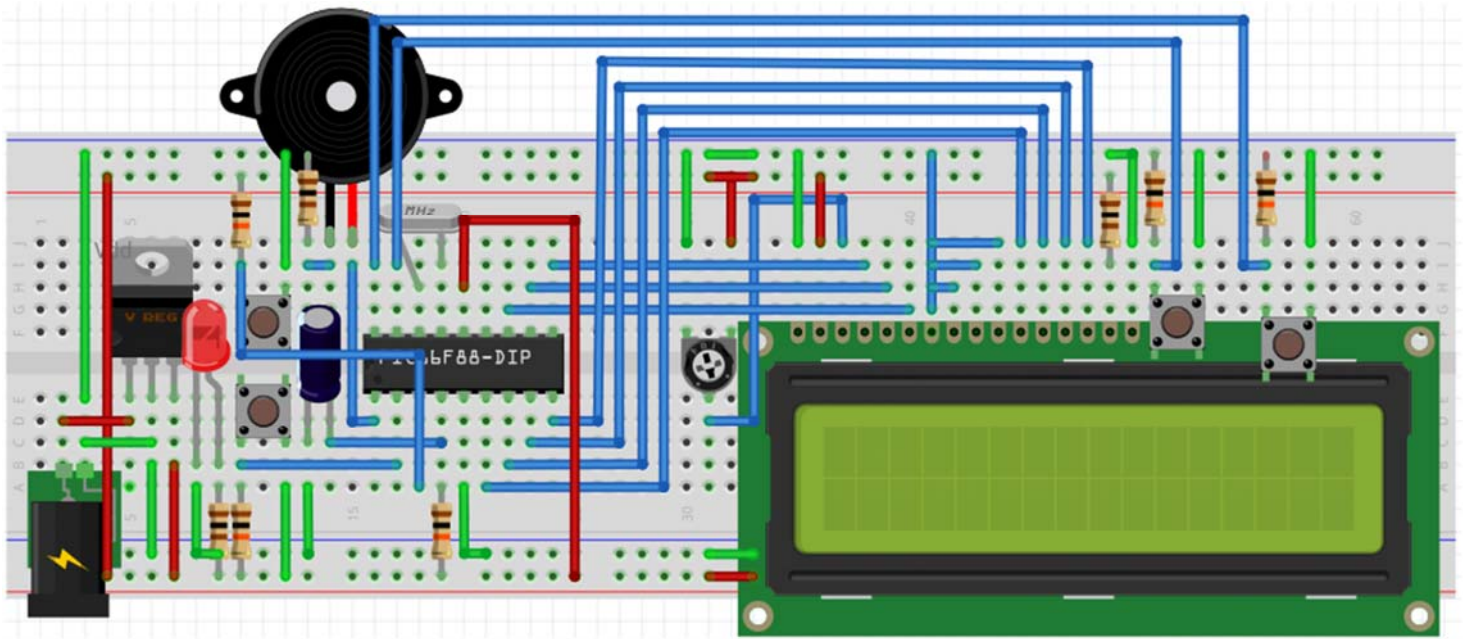
٢٢

Dr. Ahmed ElShafee, ACU Spring 2014, Information Security

- Fire sensor types







Thanks,
See you next week (ISA),...