

Example 01: automatic water tank controller

1. Search for suitable sensors that can detect water level inside a 1000L water tank. At least we need to detect two levels one for full tank alarm ,and another almost empty alarm
2. Build suitable interface to connect sensor to a 16F84A microcontroller.
3. Build suitable interface to drive a 12V water pump or 12 volt water valve to fill tank with water.
4. Your project should equipped with LCD that indicate the current operating status (full-standby-almost empty-filling)
5. You project should be equipped with indicator (led) indicating the operating status (as possible)

```
#include <xc.h>
#include <pic16f84a.h>
#include "lcd.h"
//#include "config.h"
#define Empty RA0 // Active high input
#define Full RA1 // Active high input
#define Valve RA2 // active high output
#define Standby RA3 //active high output
#define Filling RA4 //active high output
void delay_ms(int x);
void main ()
{
    int status=0; //0 : empty, 1: filling, 2:standby
    int Valve_value=0;

    INIT_HW() ;
    // initialize PORTA (RB0,1,3,4 input zones 1-4) RB2 output buzzer
    TRISA = 0x03 ;
    // call INIT_LCD()
    INIT_LCD() ;
    // goto 1st line
    write_cmd(0x80) ;
    // display welcome string 1
    display_string("Welcome auto");
    // goto line 2
    write_cmd(0xC0) ;
    // display welcome string 2
    display_string("water tank ctrl");
    // delay 50 us
    _delay(50);
    // clear lcd
    write_cmd(0x01);
    //endless loop
    while(1)
    {
        if((Empty==0) && (Full==0))
            status=0;
        else if((Empty==1) && (Full==0) && (status==0))
            status=1;
        else if((Empty==1) && (Full==1) && (status==1))
```

```

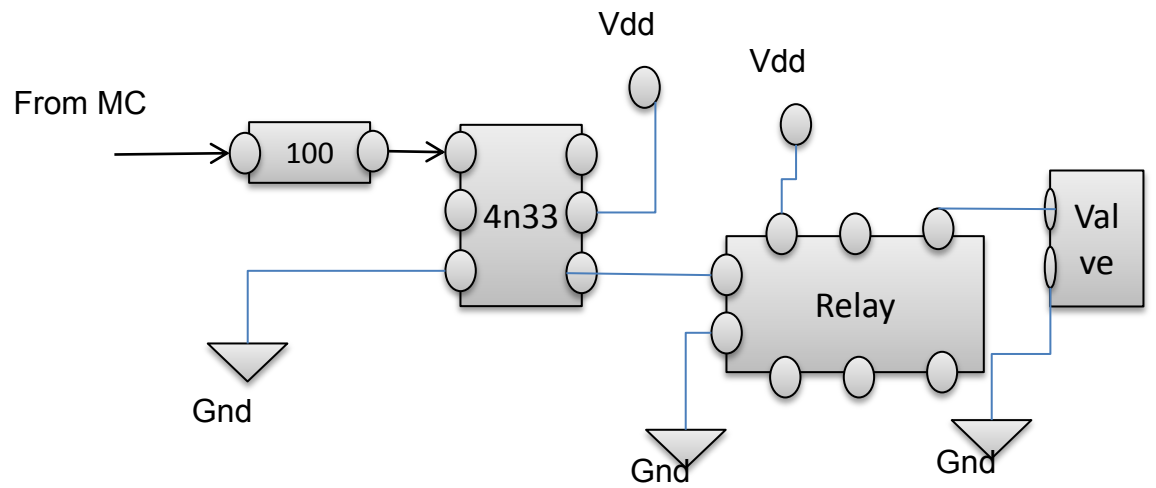
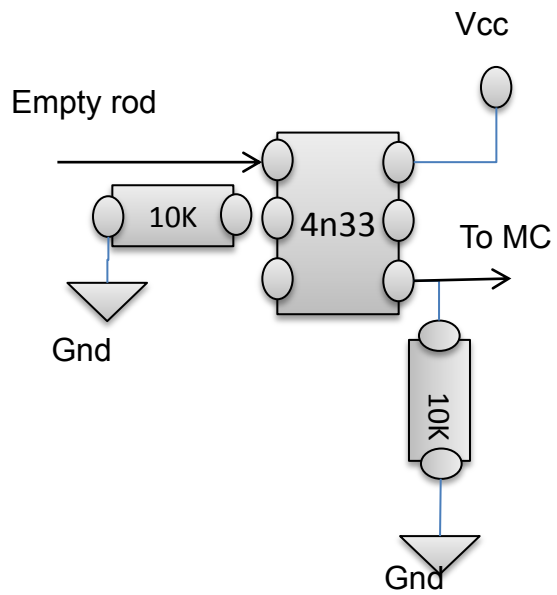
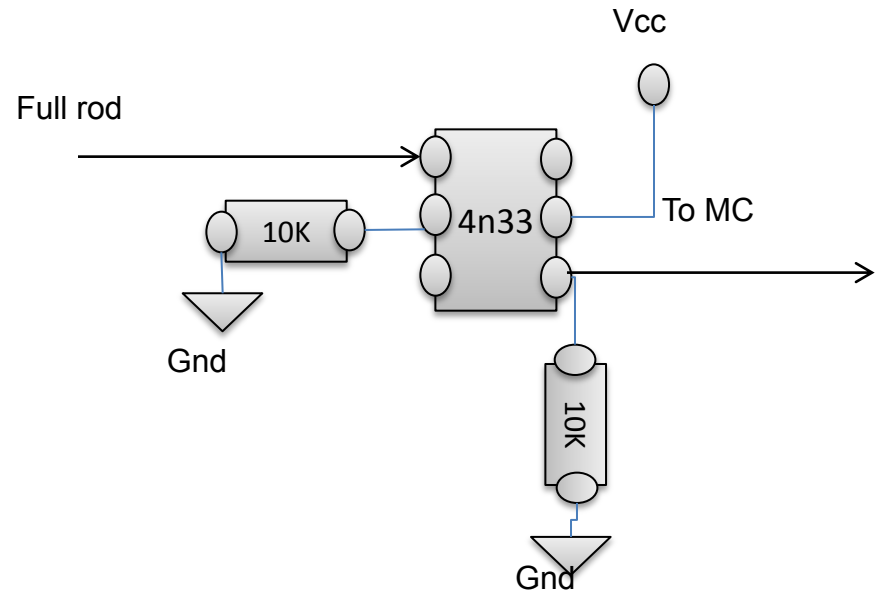
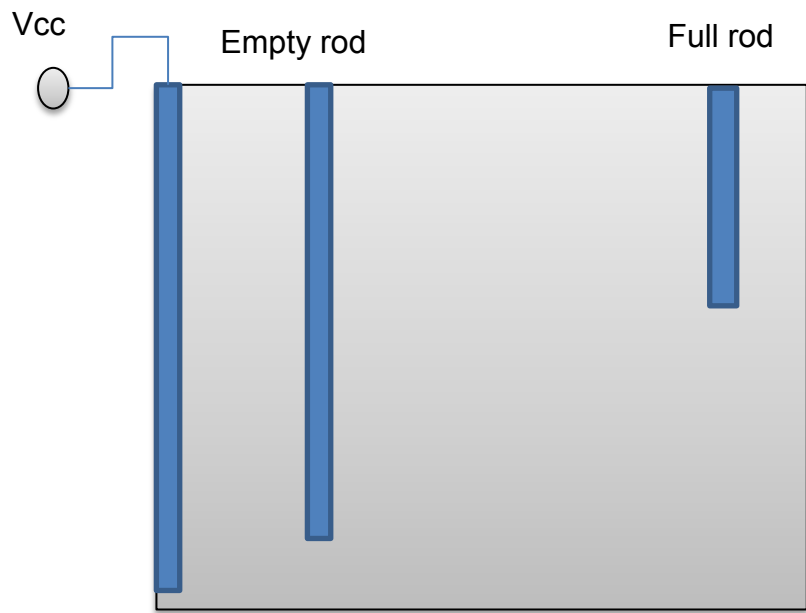
        status=2;
else if((Empty==1) && (Full==0) && (status==2))
    status=2;
if(status==0)
{
    Valve_value=1;
    Valve=Valve_value;
}
else if(status==1)
{
    Valve_value=1;
    Valve=Valve_value;
}
else if(status==2)
{
    Valve_value=0;
    Valve=Valve_value;
}

// goto 1st line
write_cmd(0x80) ;
// display line 1
display_string("Empty=");
if(Empty==0)
    display_string("Y");
else
    display_string("N");
display_string(" Full=");
if(Full==1)
    display_string("Y");
else
    display_string("N");
// goto line 2
write_cmd(0xC0) ;
// display line 2
display_string("Stats=");
if(status==0)
    display_string("E");
else if(status==1)
{
    display_string("F");
    Standby=0;
    Filling=1;
}
else
{
    display_string("S");
    Standby=1;
    Filling=0;
}
display_string(" Valve=");
if(Valve_value==0)
    display_string("N");

```

```
        else
            display_string("Y");

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



Example 02 : 2ACs controller

- Telecom room has two air conditions, that operate in series four hours each.
- When room temperature reach 27C both ACs turned on.
- Temp is measured using thermostat giving free-contact feedback to controller, when temp become above the pre-selected temp free contacts become close otherwise it's open.
- When temperature becomes below 27C, ACs start working in series again.
- Controlled gets feedback from fire control panel, it turns both ACs off when if fire alarm received, till alarm is cleared, ACs started normal operation (fire alarm makes free contact close, otherwise, it's open).
- Build suitable interface, and write code.
- Controller has LCD that displays the current operating status, and alarms.

```
#include <xc.h>
#include <pic16f84a.h>
#include "lcd.h"
//#include "config.h"
#define Temp RA0 // Active hight input
#define Fire RA1// Active hight input
#define AC1 RA2 // active hight output
#define AC2 RA3 //active hight output
#define Temp_Alarm RA4 //active hight output
void delay_ms(int x);
void main ()
{
    int timer=0;
    int status=0; //0 : empty, 1: filling, 2:standby
    int Valve_value=0;
    int AC1_status=0;
    int AC2_status=0;
    INIT_HW() ;
    // initialize PORTA (RB0,1,3,4 input zones 1-4) RB2 output buzzer
    TRISA = 0x03 ;
    // call INIT_LCD()
    INIT_LCD() ;
    // goto 1st line
    write_cmd(0x80) ;
    // display welcome string 1
    display_string("Welcome to");
    // goto line 2
    write_cmd(0xC0) ;
    // display welcome string 2
    display_string("2ACC Ctrl");
    // delay 200 us
    _delay(200);
    // cleat lcd
    write_cmd(0x01);
    //endless loop
    while(1)
    {
        if(Fire==1)
```

```

{
    timer=0;
    Temp_Alarm=0;
    AC1_status=0;
    AC2_status=0;
    AC1=AC1_status;
    AC2=AC2_status;
    // cleat lcd
    write_cmd(0x01);
    // goto 1st line
    write_cmd(0x80) ;
    // display welcome string 1
    display_string("Fire alarms");
    // goto line 2
    write_cmd(0xC0) ;
    // display welcome string 2
    display_string("ACs are off");
}
else if(Temp==1)
{
    timer=0;
    AC1_status=1;
    AC2_status=1;
    AC1=AC1_status;
    AC2=AC2_status;
    Temp_Alarm=1;
    // cleat lcd
    write_cmd(0x01);
    write_cmd(0x80) ;
    // display welcome string 1
    display_string("HighTemp alarms");
    // goto line 2
    write_cmd(0xC0) ;
    // display welcome string 2
    display_string("ACs are on");
}
else
{
    if((AC1_status==0) && (AC2_status==0))
    {
        AC1_status=1;
        AC2_status=0;
    }
    else if((AC1_status==1) && (AC2_status==1))
    {
        AC1_status=1;
        AC2_status=0;
    }
    else
    {
        if(timer==5)
        {

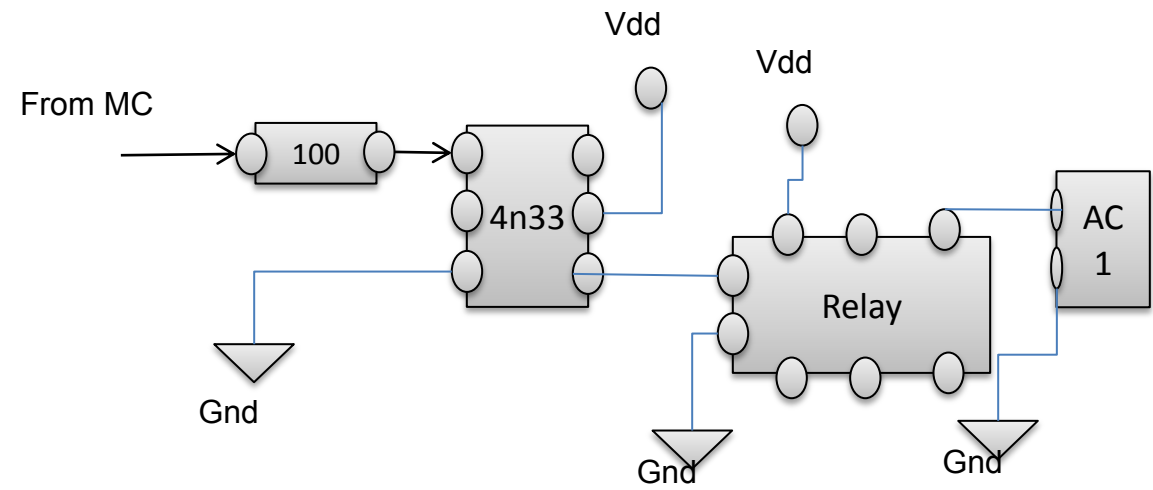
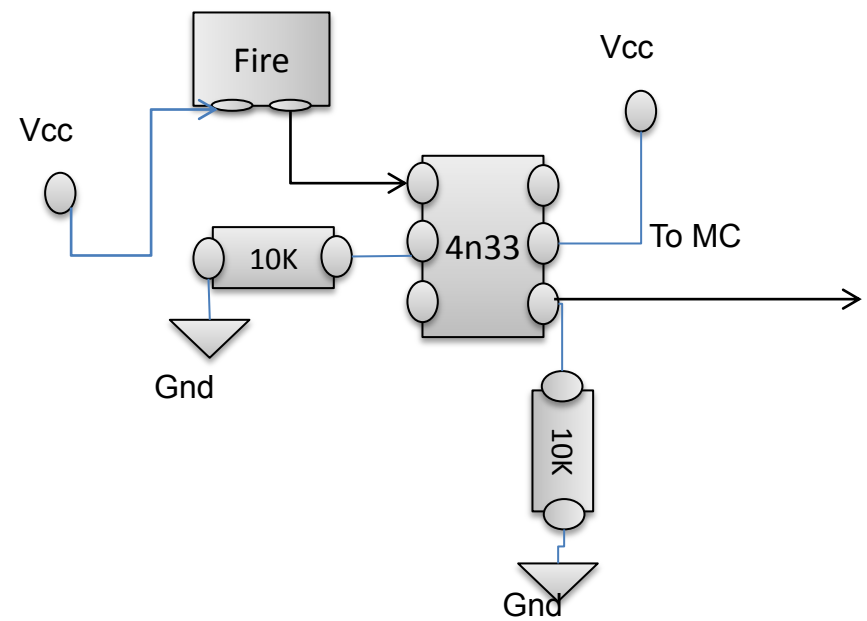
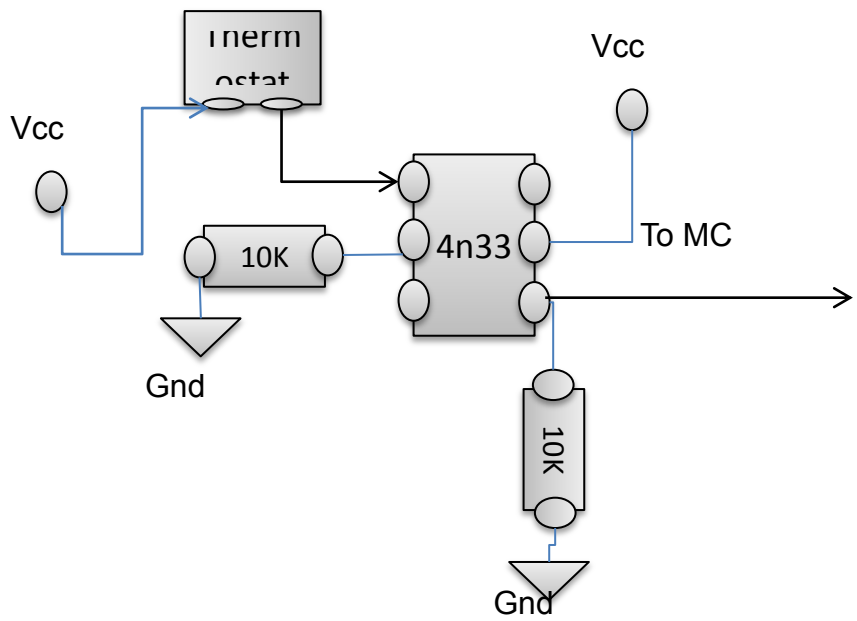
```

```

        timer=0;
        if((AC1_status==0) && (AC2_status==1))
        {
            AC1_status=1;
            AC2_status=0;
        }
        else if((AC1_status==1) && (AC2_status==0))
        {
            AC1_status=0;
            AC2_status=1;
        }
    }
}
AC1=AC1_status;
AC2=AC2_status;
Temp_Alarm=0;
timer=timer+1;
char ones=(timer%10)+'0';
char tens=(timer/10)+'0';
write_cmd(0x80) ;
// display welcome string 1
display_string("norm mod tmr=");
display_char(tens);
display_char(ones);
// goto line 2
write_cmd(0xC0) ;
// display welcome string 2
display_string("AC1=");
if(AC1_status==0)
    display_string("OFF");
else
    display_string("ON ");

display_string(" AC2=");
if(AC2_status==0)
    display_string("OFF");
else
    display_string("ON ");
}
}
}
void delay_ms(int x)
{
    for(int n=0;n<x;n++)
    {
        _delay(1000);
    }
}
}

```



Example 03 : 2PressesControlledCountersWithLCD

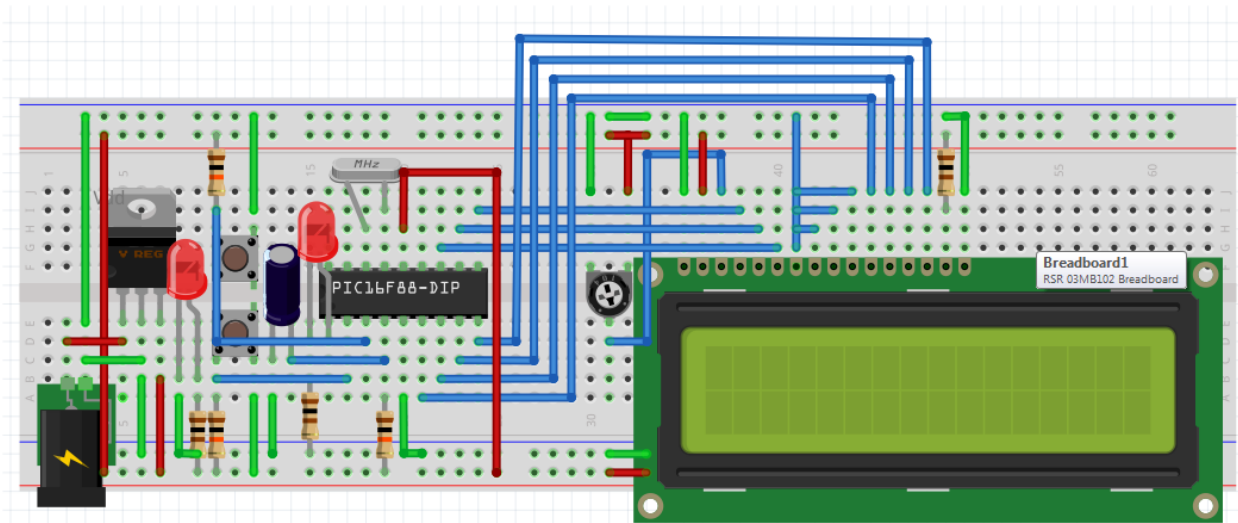
- You are required to write a program that display two different counters on LCD.
- Each counter is displayed on one line of LCD.
- Each counter is controlled using a press, that counter increments after pressing its press for convenient period than updates counter on LCD screen.
- Counter overflow after exceeding 1000

```
#include <xc.h>
#include <pic16f84a.h>
#include "lcd.h"
//#include "config.h"
void delay_ms(int x);
void main ()
{
    //initialize counter 1
    int counter1=0;
    //initialize counter 2
    int counter2=0;
    INIT_HW() ;
    // config port A RA3 & RA4 (2 presses) binput, RA2 output (LED)
    TRISA = 0x18 ;
    RA2=1;
    INIT_LCD() ;
    // goto line 1
    write_cmd(0x80) ;
    // write welcome string on 1st line
    display_string("Welcome 2 press");
    // goto line 2
    write_cmd(0xC0) ;
    // write welcome string on 2nd line
    display_string("ctrolled counter");
    //delay_ms(1000);
    // delay 50 us
    _delay(50);
    // clear lcd
    write_cmd(0x01);
    while(1)
    {
        if(RA3==1)
            counter1=(counter1+1)%1000;
        if(RA4==1)
            counter2=(counter2+1)%1000;
        // goto 1st line
        write_cmd(0x80) ;
        // display counts (3 digits) of 1st counter
        display_string("Counter1 = ");
        char d3=(counter1/100)+'0';
        char d2=((counter1-(counter1/100))/10)+'0';
        char d1=(counter1%10)+'0';
```

```

display_char(d3);
display_char(d2);
display_char(d1);
// goto 2nd line
write_cmd(0xC0) ;
// display countes (3 digits) of 2nd counter
display_string("Counter2 = ");
d3=(counter2/100)+'0';
d2=((counter2-(counter2/100))/10)+'0';
d1=(counter2%10)+'0';
display_char(d3);
display_char(d2);
display_char(d1);
// toogle status of RA2 // led blinking
RA2=~RA2;
//delay_ms(500);
// delay 50us
_delay(50);
}
}
void delay_ms(int x)
{
for(int n=0;n<x;n++)
{
_delay(1000);
}
}
}

```



Example 04 : 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

```
#include <xc.h>
#include <pic16f84a.h>
#include "lcd.h"
//#include "config.h"
void delay_ms(int x);
void main ()
{
    INIT_HW() ;
    // initialize PORTA (RB0,1,3,4 input zones 1-4) RB2 output buzzer
    TRISA = 0x1b ;
    // buzzer on
    RA2=1;
    // call INIT_LCD()
    INIT_LCD() ;
    // goto 1st line
    write_cmd(0x80) ;
    // buzzer off
    RA2=0;
    // display welcome string 1
    display_string("Welcome 2 Fire");
    // goto line 2
    write_cmd(0xC0) ;
    // display welcome string 2
    display_string("ctrolled Board");
    // delay 50 us
    _delay(50);
    // cleat lcd
    write_cmd(0x01);
    //endless loop
    while(1)
    {
        // check if any of zones are on
        if((RA0==1) || (RA1==1) || (RA3==1) || (RA4==1))
        {
            // trigger buzzer
            RA2=1;
        }
        else
        {
            // buzzer off
```

```

        RA2=0;
    }
    // goto 1st line line
    write_cmd(0x80) ;
    // write "Z1:" string
    display_string("Z1:");
    // type clr if RA0 is 0
    if(RA0==0) display_string("Clr ");
    // type Alrm if RA0 is 1
    else display_string("Alrm");
    // write "Z1:" string
    display_string("-Z2:");
    // type clr if RA1 is 0
    if(RA1==0) display_string("Clr ");
    // type Alrm if RA1 is 1
    else display_string("Alrm");
    write_cmd(0xC0) ;
    // write "Z1:" string
    display_string("Z3:");
    // type clr if RA3 is 0
    if(RA3==0) display_string("Clr ");
    // type Alrm if RA3 is 1
    else display_string("Alrm");
    // write "Z1:" string
    display_string("-Z4:");
    // type clr if RA4 is 0
    if(RA4==0) display_string("Clr ");
    // type Alrm if RA4 is 1
    else display_string("Alrm");

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

```

