

Home Security System

Idea:

"The idea for the project is home security system."

Description:

With time security breaches and thefts have become increasingly common, but this home security alarm system offers a reliable defense against unauthorized access and potential burglaries. This alarm system is designed to alert user when someone there is a movement or entry of someone in your house after the alarm is activated.

Components:

The home security system protects you against thieves. It is made with advanced technology such as "p18f4520" which is a programmable logic controller. It also includes a "motion detection sensor" and a "magnetic door sensor" for its working. So, we can list inputs as

- Motion Sensor
- Magnetic Door Sensor

When a suspicious activity is detected the p18f4520 triggers an alerting system. The alerting system consists of an "indicator light" and a "buzzer," a signal is also sent to a notifier system which gives you a notification on your phone. Outputs can be listed as

- Buzzer
- Indicator Light
- Signal To Notifier

To make all of this happen, we start by setting up the pins of the PLC (Programmable Logic Controller) designating which ones will be used for input and output. This allows us to connect the sensors and control the alarm components effectively.

Then we will read the signal value of each of the sensor which is as 0 volts when there is no motion and 5 volts when a motion is detected, same goes for the other sensor. We are going to use "OR" logic between these two Inputs and will be triggering our alerting system when either of these gives a detected signal.

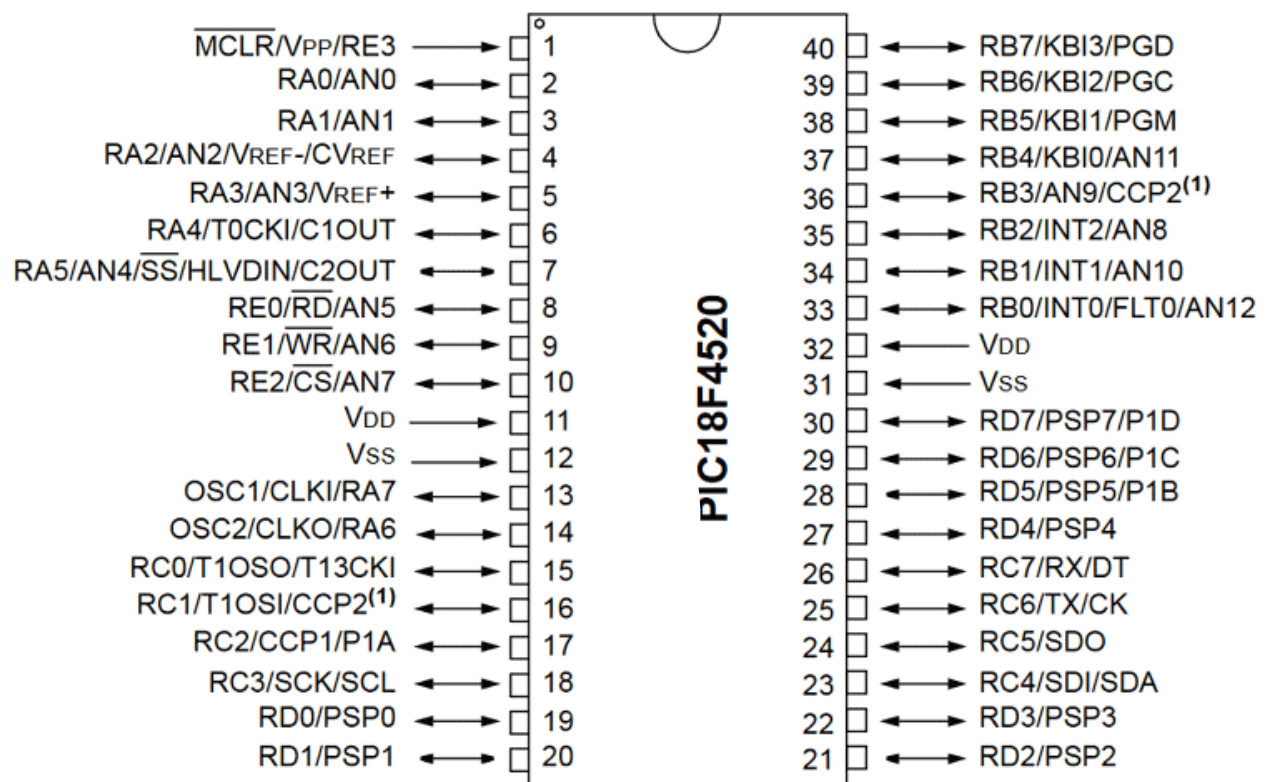
The PIR (passive infrared) sensor, which in fact is motion sensor in our case consists of two or more infrared-sensitive elements that detect the difference in infrared radiation as an object moves across the sensor's detection area. When a warm object, such as a person or animal, moves into the sensor's range, it causes a change in the infrared radiation pattern detecting motion.

A magnetic door sensor uses a reed switch and magnet to detect the opening and closing of doors or windows. When the magnet moves away, the circuit opens, triggering the system.

The two sensors work together making it impossible for the thief to enter your house and leaving no space for security related issues.

Programable Logic Controller 18F4520

The programmable logic controller in our case looks like this.



MCLR (Master Clear): This pin is responsible for external reset and is activated when pulled low. It allows you to reset the microcontroller and restore it to its default state.

RA0 (AN0): This pin serves as an input for reading analog voltages. It can also be used as a digital input or output for connecting various devices.

RA1 (AN1): Like RA0, this pin is an input for reading analog voltages or a digital input/output pin for general-purpose use.

RA2 (AN2/VREF-): This pin can be used for analog input, allowing you to measure voltages accurately. It can also serve as the negative voltage reference input for the analog-to-digital converter (ADC).

RA3 (AN3/VREF+): Like RA2, this pin is an analog input and can be used for precise voltage measurements. It can also function as the positive voltage reference input for the ADC.

RA4/T0CKI: This versatile pin can be used for multiple purposes, such as an external clock input or a digital input/output pin.

RA5/SS/HLVDIN: This multipurpose pin can be used as the Slave Select input for SPI communication or as a digital input/output pin. It can also serve as the input for the High/Low Voltage Detect module.

RE0/RD/AN5, RE1/WR/AN6, RE2/CS/AN7: These pins are multipurpose and can be configured as digital input/output pins or analog inputs. They also have additional functions related to memory interfaces.

VDD and VSS: These pins are the power supply pins. VDD is connected to the positive power source, while VSS is connected to ground.

RC0/T1OSO/T1CKI, RC1/T1OSI, RC2/CCP1: These pins have various functions, including digital input/output and association with specific modules like Timer1 oscillator and Capture/Compare/PWM.

RC3/SCK/SCL, RC4/SDI/SDA, RC5/SDO: These pins are involved in communication protocols like SPI and I2C. They can serve as digital input/output pins or as inputs for specific communication signals.

RC6/TX/CK, RC7/RX/DT: These pins are associated with USART communication and can function as digital input/output pins or transmit/receive lines.

VREF- and VREF+: These pins are used as reference voltage inputs for the ADC, ensuring accurate analog voltage measurements.

Schematic for the Home Security System:

Pin 36 (RB3/PGM): It is used for in-circuit debugging and programming. In the home security alarm system, it may not be directly utilized for output purposes.

Pin 37 (RB4): Another general-purpose output pin that can be connected to control a visual indicator, such as an LED or a set of lights. It illuminates to visually indicate the presence of a fire.

Pin 38 (RB5/INT): This pin can be set up as a general-purpose input or output. It can be connected to a Buzzer in alerting mechanism.

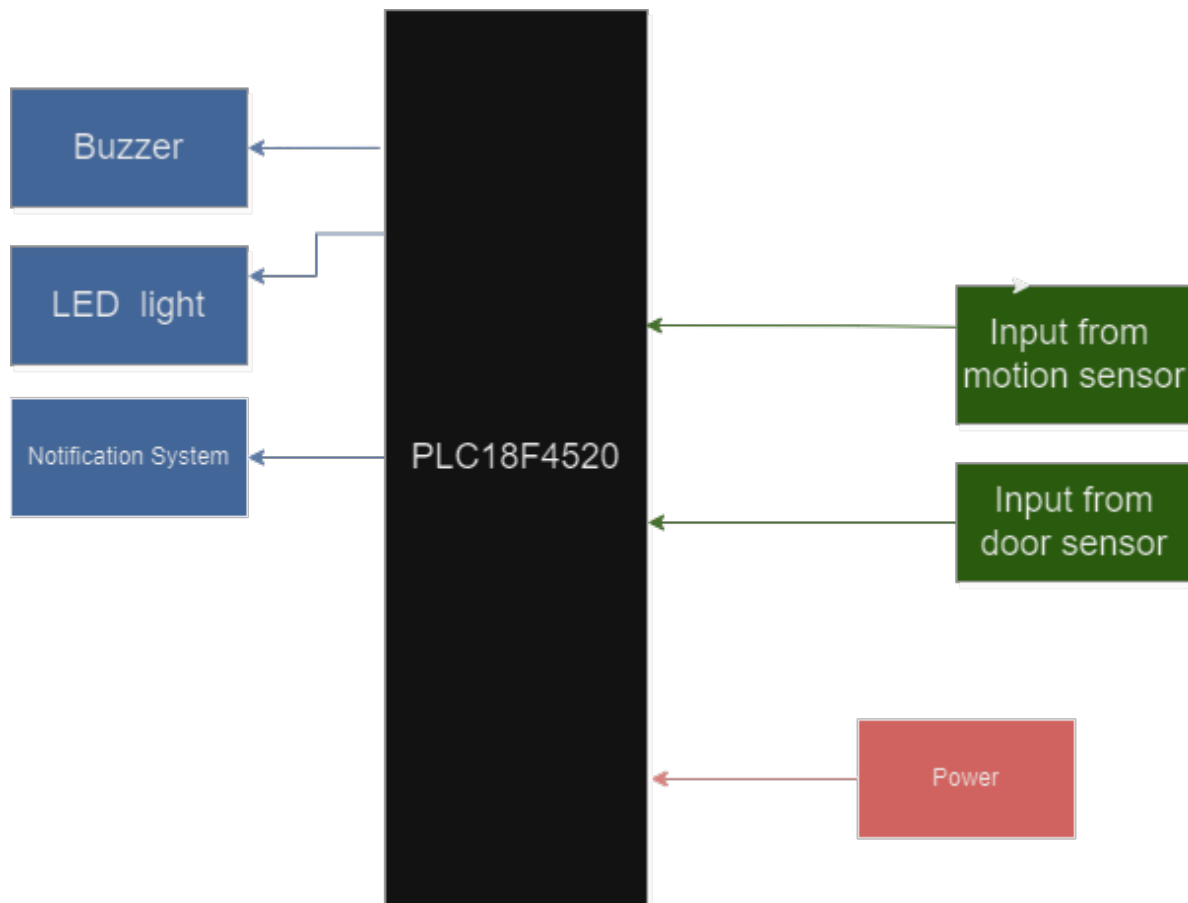
Pin 39 (RB6) is a general-purpose input or pin that can be used to send signal to notifier.

Pin 22 (RD03): This pin is a general-purpose input/output pin and can be connected to motion sensor to input motion related data.

Pin 27 (RD04): This pin can be set up as a general-purpose input or an external interrupt input. It can be connected to magnetic door sensor in the home security system,

These pins we will be using in our home security Alarm. It is to be noted that the selection of the pin is arbitrary.

Block Diagram:



The controller will manipulate the data given by the sensor in the way it is programmed.

The input pins will take data from the sensors they are inputting because the controller will read from this pin and other pin which are termed output are because the controller will write on these pins.

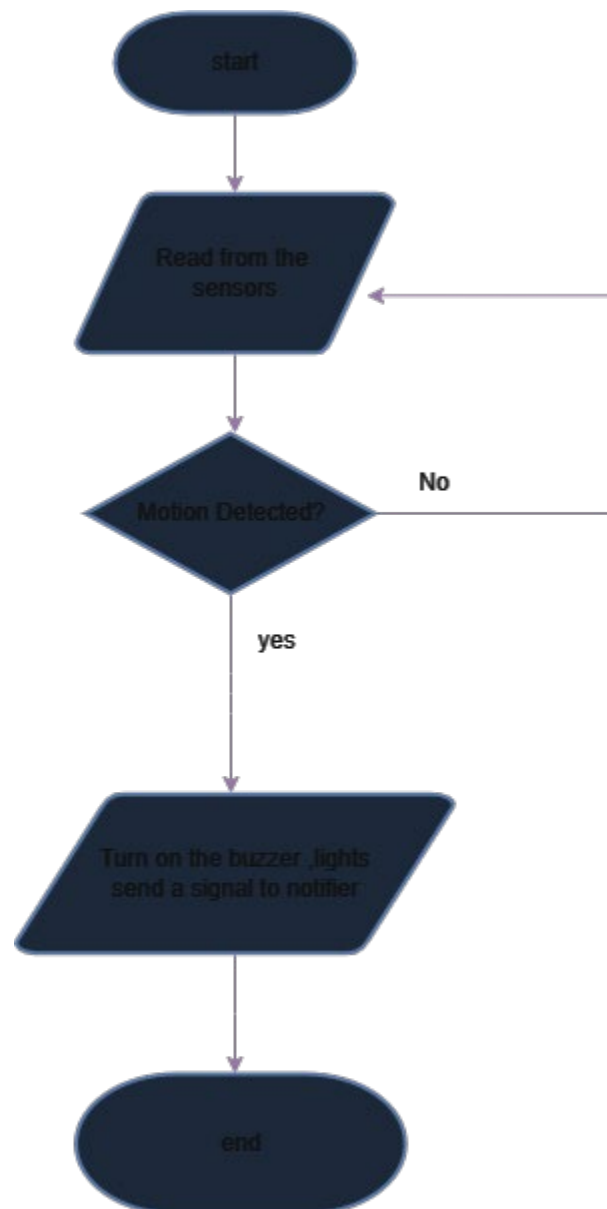
These pins we will be using in our home security Alarm. It is to be noted that the selection of the pin is arbitrary.

Now we will move to the flowchart which will explain logic of our code which is the actual soul of our program.

Flowchart:

The flowchart of the program looks like this.

Flowcharts shows the logic flow



Explanation:

Start: The home security alarm program initiates its execution.

Read Sensor Inputs: Continuously reads the status of the motion sensor and magnetic door sensor.

Check for Intrusion: An OR gate condition is implemented to check if either the motion sensor or magnetic door sensor detects any unauthorized entry or movement.

Intrusion Detected: If an intrusion is detected, the program proceeds to activate the alarm system.

Activate Buzzer and Lights: The output pins controlling the buzzer and lights are activated to indicate an unauthorized entry and send a signal to the emergency response

Not Detected: If no intrusion is detected, the program returns to the sensor reading part to maintain continuous surveillance.

End: The program execution concludes.

Description

The home security alarm program starts by initializing the microcontroller and setting up the required connections for the sensors and alarm components.

Next, the program enters a loop where it continuously reads the inputs from the motion sensor and magnetic door sensor.

An OR gate condition is implemented to check if either the motion sensor or the magnetic door sensor detects any unauthorized entry or movement.

If an intrusion is detected, the program activates the alarm system, which includes sounding the buzzer and activating the lights to alert the occupants and deter potential intruders.

The program also sends a signal to homeowner, notifying them of the security breach.

Once the alarm is triggered, the program continues to monitor the sensor inputs, ready to detect any further unauthorized activities.

The loop keeps running until the system is manually reset or disarmed, ensuring continuous surveillance and protection for the home.

Now the we will move to the real code

Code

```
// Including the library
```

```
#include <xc.h>
```

```

// Define sensor names and corresponding ports
#define DOOR_SENSOR PORTDbits.RD3 // Magnetic Door Sensor (connected to RD3)
#define MOTION_SENSOR PORTDbits.RD4 // Motion Sensor (connected to RD4)

// Define actuator names and corresponding ports
#define NOTIFIER LATBbits.LATB6 // Notifier (connected to RB6)
#define LED LATBbits.LATB2 // LED (connected to RB2)
#define BUZZER LATBbits.LATB5 // LED (connected to RB2)

void initialize();

void main() {
    initialize();

    while (1) {
        // Check the sensor values
        if (DOOR_SENSOR || MOTION_SENSOR) {
            // Either the door is open or motion is detected
            NOTIFIER = 1; // Activate Notifier
            LED = 1;
            BUZZER=1;// Turn on LED
        } else {
            // No security threat detected
            NOTIFIER = 0; // Deactivate Notifier
            LED = 0;
            BUZZER=0;// Turn off LED
        }
    }
}

void initialize() {

```

```

// Set sensor pin configurations as inputs
TRISDbits.TRISD3 = 1; // Set RD3 as input (Magnetic Door Sensor)
TRISDbits.TRISD4 = 1; // Set RD4 as input (Motion Sensor)

// Set actuator pin configurations as outputs
TRISBbits.TRISB6 = 0; // Set RB6 as output (Notifier)
TRISBbits.TRISB2 = 0;
TRISBbits.TRISB5 = 0; // Set RB5 as output (LED)

// Initialize actuators to be off
NOTIFIER = 0; // Deactivate Notifier
LED = 0;
BUZZER=0;
// Turn off other LEDs
LATB = 0x0000;
}

```

This is the code for the project

The code has following functions,

- **Initializer:** In this function we are setting up the pin which we are going to use and configuring them either they will be used as inputs or outputs
- **Main function.:** This is the function which calls all the functions we have featured in an infinite loop in it.

The home security alarm program starts by initializing the microcontroller and setting up the required connections for the sensors and alarm components.

Next, the program enters a loop where it continuously reads the inputs from the motion sensor and magnetic door sensor.

An OR gate condition is implemented to check if either the motion sensor or the magnetic door sensor detects any unauthorized entry or movement.

If an intrusion is detected, the program activates the alarm system, which includes sounding the buzzer and activating the lights to alert the occupants and deter potential intruders.

The program also sends a signal to the emergency response system or the homeowner, notifying them of the security breach.

Once the alarm is triggered, the program continues to monitor the sensor inputs, ready to detect any further unauthorized activities.

The loop keeps running until the system is manually reset or disarmed, ensuring continuous surveillance and protection for the home.

The “Xc.h” is an essential header for PLC coding.

Conclusion:

The project takes input from sensors and depending upon them decides what to do. Whenever a door is unlocked after an alarm has been turned on or motion is detected, the alarm will ring.