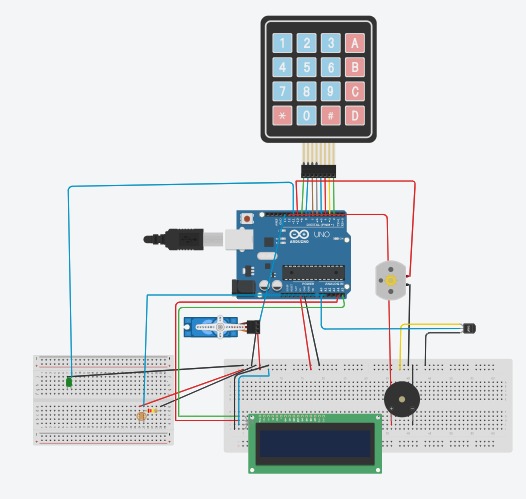
**SMART HOME SYSTEM**

**1.Introduction:**

A smart home monitoring system uses sensors to detect movement and physiological indicators to improve comfort, quality of life, and safety.

We created a password door lock in this project that triggers an alarm if the password is entered incorrectly three times. We also have a temperature sensor that activates a fan when the room becomes too hot, as well as a light sensor that activates the lights when the room becomes dark.

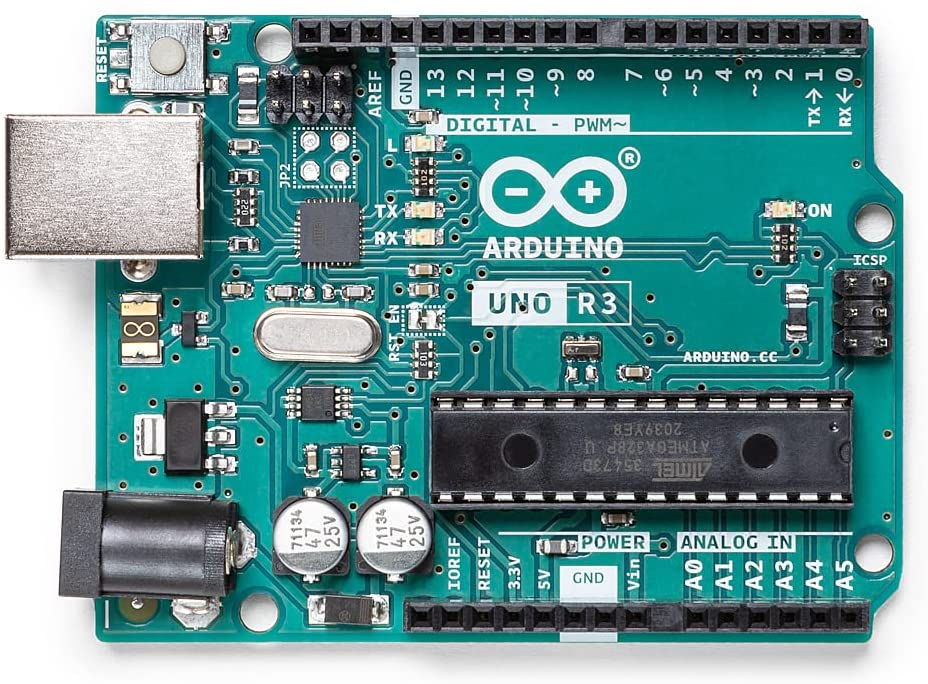
**2.Block Diagram**



**3.Components Used**

**A. Arduino Uno**

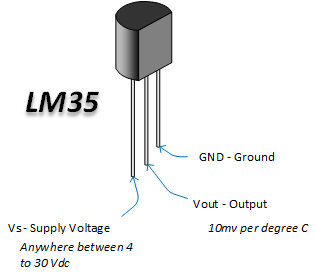
Arduino uno is a microcontroller based upon ATmega328P(datasheet). It has 14 digital I/O pins, 6 analog inputs, 16 MHz ceramic resonator (CSTCE16M0V53-R0),USB connectivity, power, ICSP headers and reset button.It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can play around with the Uno without worrying about doing something wrong; worst case scenario, you can replace the chip for a few dollars and start over again.



|  |  |
| --- | --- |
| MICROCONTROLLER | ATmega328P |
| OPERATING VOLTAGE | 5V |
| INPUT VOLTAGE (RECOMMENDED) | 7-12V |
| INPUT VOLTAGE (LIMIT) | 6-20V |
| DIGITAL I/O PINS | 14 (of which 6 provide PWM output) |
| PWM DIGITAL I/O PINS | 6 |
| ANALOG INPUT PINS | 6 |
| DC CURRENT PER I/O PIN | 20 mA |
| DC CURRENT FOR 3.3V PIN | 50 mA |
| FLASH MEMORY | 32 KB of which 0.5 KB used by bootloader |
| SRAM | 2 KB (ATmega328P) |
| EEPROM | 1 KB (ATmega328P) |
| CLOCK SPEED | 16 MHz |
| LED\_BUILTIN | 13 |
| LENGTH | 68.6 mm |
| WIDTH | 53.4 mm |
| WEIGHT | 29 g |

**B) Temperature sensor- LM 35**

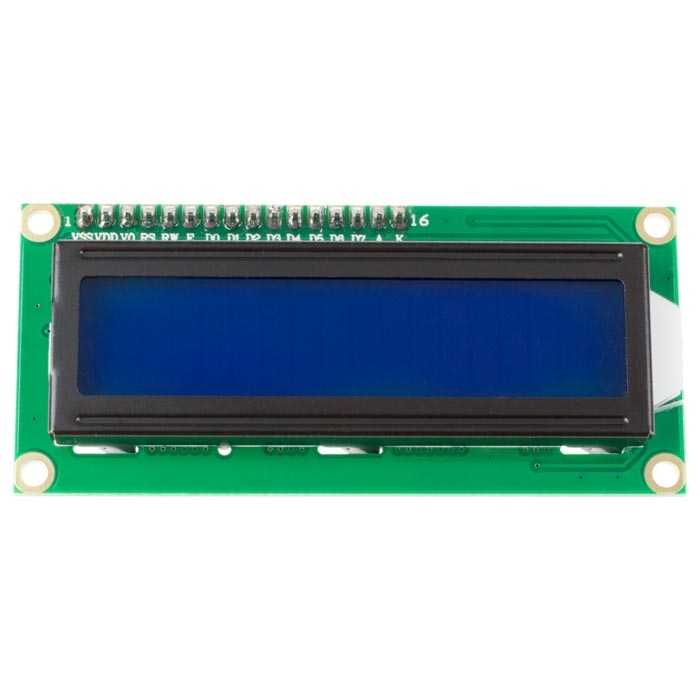
LM35 series are precision integrated circuits. A temperature device with an output voltage that scales linearly with temperature in degrees Celsius. LM35 devices have advantages over linear devices Temperature sensor calibrated in Kelvin, as user No need to subtract a large constant voltage From Exit to Comfortable Celsius zoom. The LM35 device does not require a externally calibrated or trimmed to provide typical Accuracy of ±¼°C and ±¾°C at room temperature Over the full temperature range of -55°C to 150°C. reduce Cost is guaranteed through fine-tuning and calibration wafer level. Low output impedance, linear output, and accurate self-calibration of LM35 devices Interface to readout or control circuitry Extremely light. The device operates on a single current supply or positive and negative supplies. The LM35 device draws only 60 μA from its power supply. Self-heating in still air is very low, below 0.1°C.



|  |  |
| --- | --- |
| Local sensor accuracy (Max) (+/- C) | 1 |
| Operating temperature range (C) | -40 to 110, -55 to 150, 0 to 100, 0 to 70 |
| Supply voltage (Min) (V) | 4 |
| Supply voltage (Max) (V) | 30 |
| Supply current (Max) (uA) | 114 |
| Interface type | Analog output |
| Sensor gain (mV/Deg C) | 10 |

**C) LCD 16×2 (LM016L)**

The most commonly used character-based LCDs are based on Hitachi's HD44780 controller or other HD44580-compatible controllers. We will discuss character based LCDs, their interface to different microcontrollers, different interfaces (8bit/4bit), programming, special features and tricks that can bring a whole new look to your application using these simple looking LCDs .

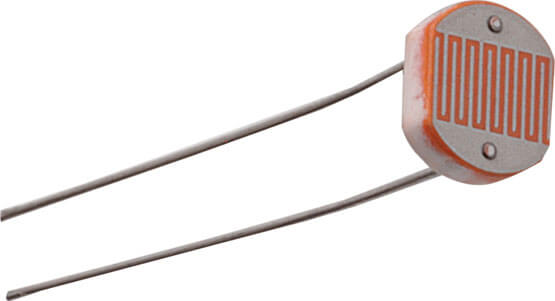


The most common LCDs on the market today are 1-line, 2-line or 4-line LCDs, which have only 1 controller and support up to 80 characters, while LCDs that support more than 80 characters use 2 HD44780 controllers. Most 1-controller LCDs have 14 pins, and 2-controller LCDs have 16 pins (both pins are extra for the LED backlight connector). The pin descriptions are shown in the table below.

|  |  |  |
| --- | --- | --- |
| Pin No. | Name | Description |
| 1 | D7 | Data bus line 7 (MSB) |
| 2 | D6 | Data bus line 6 |
| 3 | D5 | Data bus line 5 |
| 4 | D4 | Data bus line 4 |
| 5 | D3 | Data bus line 3 |
| 6 | D2 | Data bus line 2 |
| 7 | D1 | Data bus line 1 |
| 8 | D0 | Data bus line 0 |
| 9 | EN1 | Enable signal for row 0 and 1 (1stcontroller) |
| 10 | R/W | 0 = Write to LCD module  1 = Read from LCD module |
| 11 | RS | 0 = Instruction input  1 = Data input |
| 12 | VEE | Contrast adjust |
| 13 | VSS | Power supply (GND) |
| 14 | VCC | Power supply (+5V) |
| 15 | EN2 | Enable signal for row 2 and 3 (2ndcontroller) |
| 16 | NC | Not Connected |

**D) LDR**

An LDR or photoresistor is a passive component that reduces the resistance of the sensitive surface of the component to receive brightness. The resistance of a photoresistor decreases with increasing light intensity; in other words, it exhibits photoconductivity. Photoresistors can be used as resistive semiconductors in photodetector circuits as well as in light-activated and dark-activated circuits. In the dark, a photoresistor can have a resistance of several megaohms (MΩ), while in the light, a photoresistor can have a resistance of only a few hundred ohms. When the light incident on the photoresistor exceeds a certain frequency, the photons absorbed by the semiconductor give the bound electrons enough energy to jump into the conduction band. The resulting free electrons conduct electricity, thereby reducing resistance. The resistance range and sensitivity of photoresistors can vary widely from device to device. Moreover, unique photoresistors may react substantially differently to photons within certain wavelength bands.



**E) 4\*4 Matrix Keypad**

In most cases, we use a key, a button, or a switch to obtain input value in our projects. One GPIO pin is required when one key, button, or switch is connected to the microcontroller. However, if we want to interface a large number of keys, such as 9, 12, or 16, we'll need a microcontroller with a lot of GPIO pins, and we'll lose a lot of GPIO pins.

Don't be concerned! A device that can remedy this problem is the 4×4 matrix keypad. The 4×4 matrix keypad is an input device that is typically used to provide input values for projects. It features a total of 16 keys, allowing for a total of 16 input values. The most intriguing aspect is that it just required 8 GPIO pins on a microcontroller.



This is a non-encoded matrix keypad consisting of 16 keys in parallel.There are four rows and four columns pins R1-R4 control the rows and pins L1-L4 control the columns.

|  |  |  |
| --- | --- | --- |
| Pin Number | Pin Number | Description |
| 1 | R1 | Taken out from 1st ROW |
| 2 | R2 | Taken out from 2nd ROW |
| 3 | R3 | Taken out from 3rd ROW |
| 4 | R4 | Taken out from 4th ROW |
| 5 | C1 | Taken out from 1st COLUMN |
| 6 | C2 | Taken out from 2nd COLUMN |
| 7 | C3 | Taken out from 3rd COLUMN |
| 8 | C4 | Taken out from 4th COLUMN |

**How a microcontroller can read these lines for a button-pressed state**

1.All of the column and row lines are set to input by the microcontroller.

2.The microcontroller then sets the row to LOW.

3.After that, it goes over each column line one by one.

4.The button on the row has not been touched if the column connection remains HIGH.

5.The microcontroller understands which row was set to LOW and which column was detected as LOW when verified whether it goes LOW.

6.Finally, it determines which button corresponds to the detected row and column.

|  |  |
| --- | --- |
| Parameter | Value |
| Product type | 4 X 4 keypad module |
| Maximum Voltage rating | 24V DC |
| Maximum Current rating | 30 mA |
| Operating temp range | 32 to 122 °F (0 to 50 °C) |
| Numbers of Pin | 8 |
| Keypad Dimensions | 6.9 cm X 7.9 cm |
| Cable dimensions | 2.0 cm X 8.8 cm |
| Design | Ultra-thin |

**F) Buzzer**

A buzzer or beeper is a mechanical, electromechanical, or piezoelectric audio signaling device (piezo for short).The main purpose of this is to transform an audio signal to a sound signal. It is commonly used in timers, alarm devices, printers, alarms, computers, and other equipment that are powered by DC voltage. It may produce various sounds such as alert, music, bell, and siren depending on the varied designs.



The buzzer's pin arrangement is illustrated below. It has two pins, one positive and one negative. The '+' symbol or a longer terminal is used to represent the positive terminal of this and is powered through 6 volts. The negative terminal is represented by the '-'symbol or short terminal and is connected to the GND terminal.

**The buzzer's specifications include the following.**

1.Black is the colour.

2.3,300Hz is the frequency range.

3.Temperatures range from –20°C to +60°C during operation.

4.The operating voltage is 3V to 24V DC.

5.The supply current is less than 15mA and the sound pressure level is 85dBA or 10cm.

**G) Servo motors**

A servomotor is a rotary or linear actuator that provides for precise control of angular or linear position, velocity, and acceleration. It is made of an appropriate motor and a position feedback sensor. It also necessitates a complex controller, which is frequently a separate module created exclusively for servomotors. Although the term servomotor is typically used to refer to a motor appropriate for use in a closed-loop control system, it is not a specific type of motor.



**Mechanism**

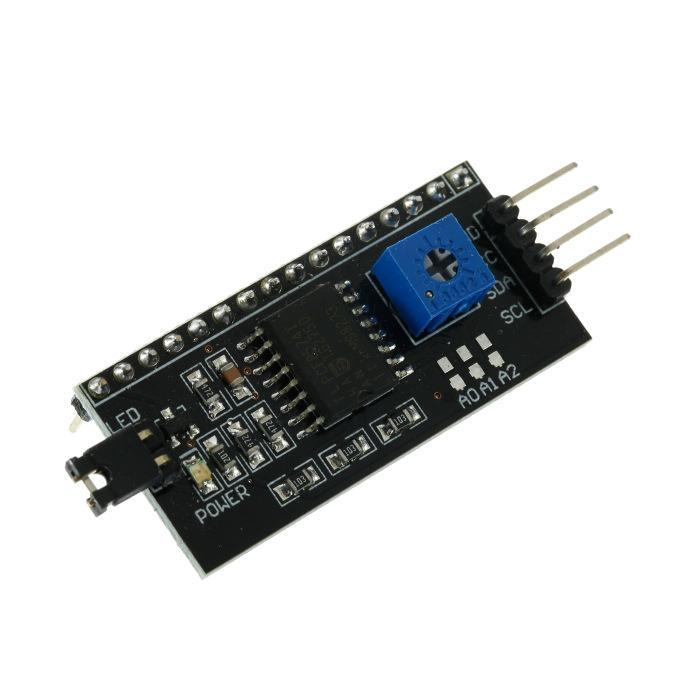
A servomotor is a closed-loop servomechanism that controls its motion and ultimate position using position feedback. A signal representing the output shaft's commanded position is fed into its control.

To provide position and speed feedback, the motor is connected to a position encoder. Only the position is measured in the most basic scenario. The measured output position is compared to the command position, which is the controller's external input. If the output position does not match the desired position, an error signal is generated, causing the motor to rotate in either direction to bring the output shaft to the correct position. The error signal decreases as the positions approach zero, and the motor stops.

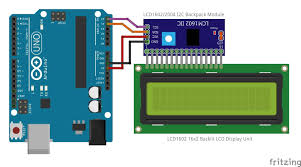
The simplest servo motors use a potentiometer for position detection and bang-bang motor control; the motor always revolves at maximum speed (or is stopped). Although this sort of servomotor is not commonly used in industrial motion control, it is the foundation for the simple and inexpensive servos used in radio-controlled models.

**H) I2C**

Philips Semiconductors invented the I2C (Inter-Integrated Circuit, eye-squared-C) serial communication bus in 1982. It is a synchronous, multi-controller/multi-target (controller/target), packet switched, single-ended serial communication bus. It's commonly used in short-distance intra-board communication to connect lower-speed peripheral ICs to processors and microcontrollers.



I2C is widely used to connect peripheral circuits to prototyping platforms like the Arduino and Raspberry Pi. Although I2C does not use a standardised connector, board designers have come up with a number of wiring schemes for I2C connectivity. Some developers have recommended utilising alternating signal and power connections of the following wiring schemes to prevent the potential damage caused by plugging 0.1-inch headers in backwards: (GND, SCL, VCC, SDA) or (GND, SCL, VCC, SDA) (VCC, SDA, GND, SCL)



The great majority of applications employ I2C in its original form—peripheral ICs directly connected to a processor on the same printed circuit board, and hence over relatively small distances of less than 1 foot, without the requirement of a connection. Another form of I2C, however, can transmit up to 20 metres via CAT5 or other cable when utilising a differential driver.

I2C signals are carried by a variety of common connectors. I2C is carried by the UEXT connector, the 10-pin iPack connector, the 6P6C Lego Mindstorms NXT connector, and a few users use the 8P8C connectors and CAT5 cable traditionally used for Ethernet physical layer to instead transport differential-encoded I2C signals or boosted single-ended I2C signals.

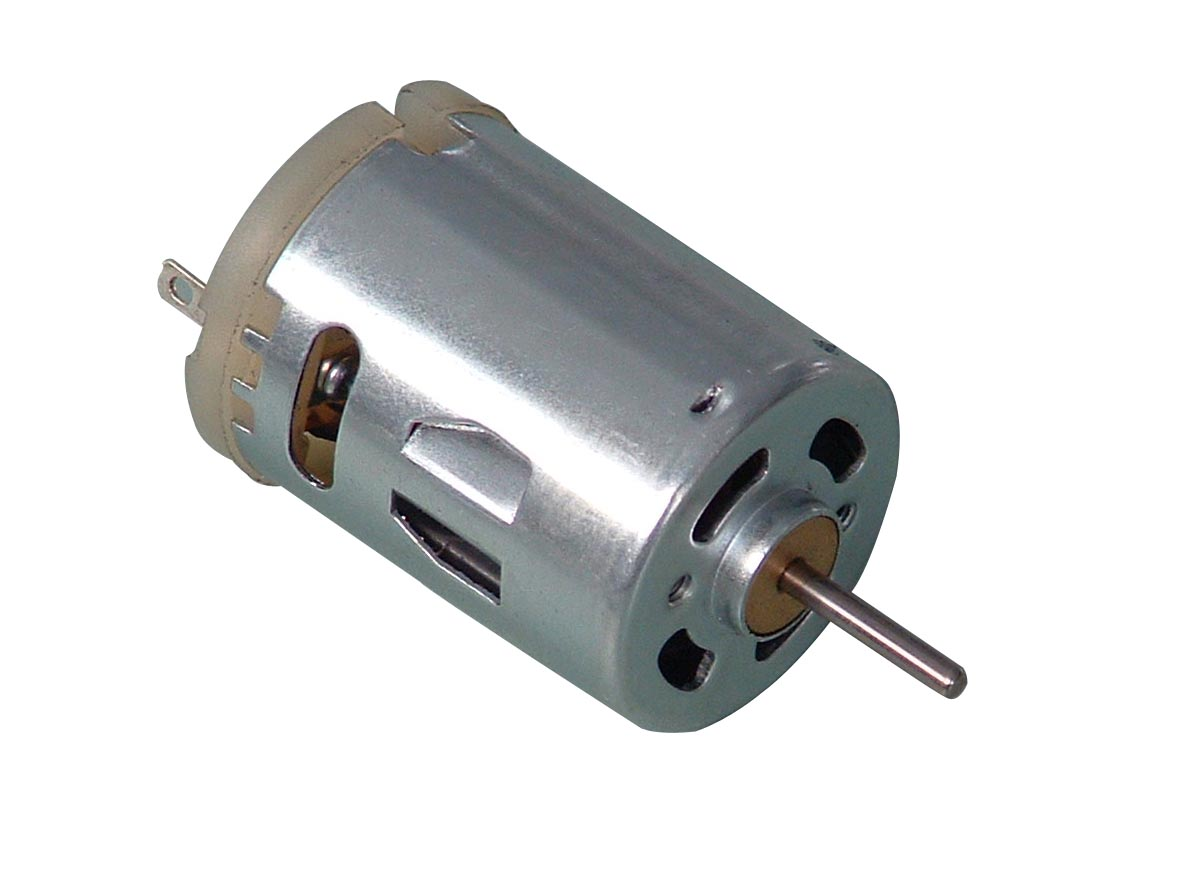
**I) LED**

A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it. Electrons recombine with electron holes in the semiconductor, producing energy in the form of photons. The energy required for electrons to pass the semiconductor's band gap determines the hue of light (equivalent to the energy of photons). Multiple semiconductors or a coating of light-emitting phosphor on the semiconductor device are used to produce white light.



**J)DC MOTOR**

A DC motor is any of a class of rotary electrical motors that converts direct current (DC) electrical energy into mechanical energy.The most common varieties rely on magnetic fields to produce forces. Almost all DC motors contain an internal mechanism, either electromechanical or electronic, that changes the direction of current in a section of the motor on a regular basis.



**4. Pseudo code :**

1. Include the required libraries
2. Initialize variables and define pin connections to Arduino.
3. Create objects for keypad, lcd and servo motor.
4. Create an array to represent keys on customkeypad.
5. void setup ()

{

1. Setup LCD with backlight and initialize

2.Create 3 individual tasks for Temperature sensing, Light sensing and Password door lock using xTaskCreate API.

}

1. Define a function to clear data on LCD.
2. To define the temperature sensor task:

{

1. Create an infinite while loop.
2. Read the analog value of the temperature from the LM35 sensor and calculate the celsius value of the temperature reading and display it on serial monitor.
3. If ( celsius value is greater than 30)

{

Turn on the fan

}

}

8. To define the Light sensor task:

{

1. Create an infinite while loop.

2. Read the analog value of the light intensity from photodiode.

3. If ( sensor value is less than or equal to 400)

{

Turn on light

}

9. To define the Password door lock task:

{

1. For ( i=0 ; i<3; ) {

1. Initialize LCD and display “ Enter Password”
2. Look for a keypress and store the keypress value into an array
3. Compare the entered password and original password.
4. If ( both are same) {

Display “ correct” and turn on servo motor}

Else

{

1. Display “incorrect” and increment ‘i’ value
2. If (i==3){ turn on buzzer}

}

2. Clear LCD display.

**5. Functional description**

This is a smart home system which has a password door lock system. When the passcode is given correctly the system automatically opens the door and if the password entered is incorrect it triggers an alarm system. It also includes a temperature sensor whose sensor values can be used by the system to maintain the temperature. If the temperature is more than 30 degree Celsius the system automatically turns on the fans. Sensor values of the photodiode are used to determine the amount of light intensity in the room, If the light intensity is less than required level the system turns on the light and viceversa.

**6. Discussion**

Initially we divided the project into 3 different tasks and tried running those in Arduino ide. We faced an issue with the temperature sensor as it was sensing temperature incorrectly. We replaced it with a new sensor and the problem was resolved.Next we wrote the three tasks in a single program but we were not able to integrate the hardware as the arduino pins were insufficient. So, we used an I2C backpack and integrated the hardware. Then, we transformed our raw arduino code into FreeRTOS code, executing separate tasks in 3 different threads. Initially there was a problem in displaying the data on LCD screen and using the vTaskDelay API function instead of delay function improved the performance of the microcontroller and the problem was resolved.

**7. Conclusion**

We designed a system that consists of a password door lock system, fan and light control using FreeRTOS running on an Arduino. The project can be expanded by including various features like movement detection to turn on/off lights and fans, a phone application to control every device in the house, sending alerts directly to smartphone etc. The system can be integrated with home security solutions to allow greater control and safety for residents.