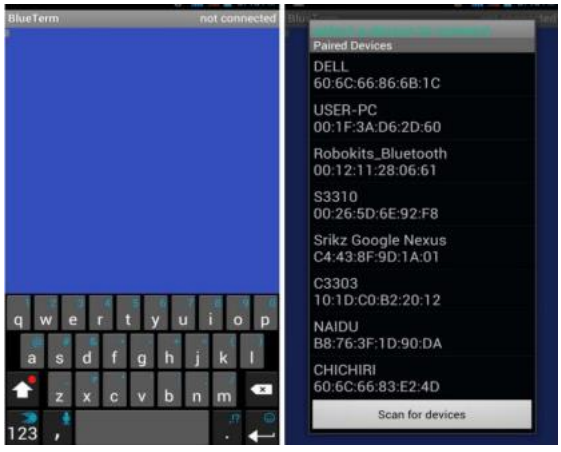
# Smart Home Project

## Project Description :

- This project is Smart Home based Bluetooth where we want to control home appliance wirelessly using Mobile App via Bluetooth.
- Two ECU's Communicate with each other the first is a control ECU which takes the input from Bluetooth and send it to the Actuator ECU via SPI to interpret which action should be taken
- The project consists of two microcontrollers(Atmega32) on is the master an the other one is salve that are connected via SPI
- The master microcontroller sends the data to the salve to use the actuators (5 on/off lamps, one dimming lamp", door, air-condition).
- The login system consists of admin and user "admin is remoted only" Admin and user can access to all applies except user cannot control the door opening.
- Admin mode can register any user or remove.
- Usernames and password are kept into memory even if the system is powered off via the eeprom.

## Prerequests :

1. Download Atmel studio or SDK(eclipse) for the code
2. Download Proteus for the simulation or use the components below
3. Download Blueterm which can be easily downloaded from Google play
   You need to pair with the Bluetooth first
   Then try to connect



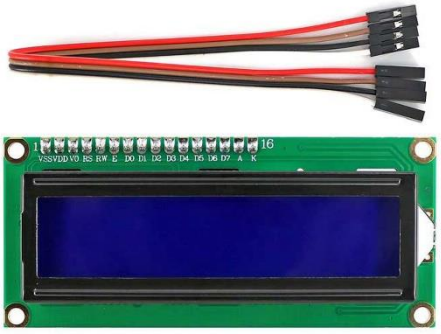
## Components:

1. Two microcontrollers (Atmega 32)
2. LCD 16x2
3. Keypad 4x4
4. Bluetooth module HC-05
5. LM35 temperature sensor
6. DC motor
7. Servo motor
8. 5 LEDs

## Specifications:

### 1. LCD& keypad:

They are used to login to system as a user only by entering username and password on keypad and display it on lcd.
They can control the system even if any user login by remoted mode except admin until allowing of admin.



Interface for keypad: 8-pin access to 4×4 matrix

The LCD is used in a 4pins data mode

‘

16x2 LCD Pinout Configuration

| Pin No: | Pin Name: | Description |
|---|---|---|
| 1 | Vss (Ground) | Ground pin connected to system ground |
| 2 | Vdd (+5 Volt) | Powers the LCD with +5V (4.7V – 5.3V) |
| 3 | VE (Contrast V) | Decides the contrast level of display. Grounded to get maximum contrast. |

| | | |
|---|---|---|
| 4 | Register Select | Connected to Microcontroller to shift between command/data register |
| 5 | Read/Write | Used to read or write data. Normally grounded to write data to LCD |
| 6 | Enable | Connected to Microcontroller Pin and toggled between 1 and 0 for data acknowledgement |
| 7 | Data Pin 0 | Data pins 0 to 7 forms a 8-bit data line. They can be connected to Microcontroller to send 8-bit data.<br>These LCD's can also operate on 4-bit mode in such case Data pin 4,5,6 and 7 will be left free. |
| 8 | Data Pin 1 | |
| 9 | Data Pin 2 | |
| 10 | Data Pin 3 | |
| 11 | Data Pin 4 | |
| 12 | Data Pin 5 | |
| 13 | Data Pin 6 | |
| 14 | Data Pin 7 | |
| 15 | LED Positive | Backlight LED pin positive terminal |
| 16 | LED Negative | Backlight LED pin negative terminal |

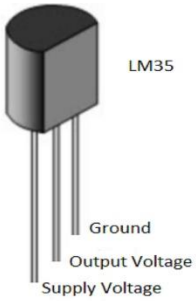
## 2. RC SERVO MOTOR:


• The operating voltage from 4.8 to 6 v (5v is the mostly used)
• Stall current = 0.7 A
• Advantages of RC servo motor :
1. Small 2. Cheap

It is used to control the door

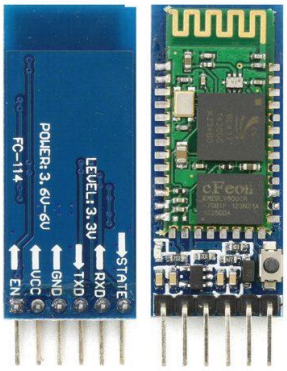
**3. LM35 temperature sensor:**

Operates from 4 V to 30 V
It is used to control the air conditioner




**4. Bluetooth module HC-05**

It is a 6 pin module
It works on serial communication (USART)
The module works on 5V or 3.3V
It is used to send the data remotely by admin or user




**5. DC motor**

DC motors normally have just two leads,
one positive and one negative
It is used as an air conditioner that
Is controlled by lm35 sensor

# Some important notes about the code :

The master controls the LCD and KEYPAD  and the slave controls all the other HAL layer

In MCAL layer in master:

1. DIO :
   ES_t DIO_enuSetPinDirection(u8 Copy_u8PortID , u8 Copy_u8PinID , u8 Copy_u8Direction)
   In this function it returns an error state to detect the error and it take the port id and pin id you want to set the direction of and sets the direction to be input or output

2. Eeprom :
   void EEPROM_vWriteByteToAddress(const u16 uiAddress, const u8 uiData);
   this function writes a byte into the address you want so it takes two parameters one is the address the second is the that in byte you want to write in eeprom

3. SPI:
   u8 SPI_ui8TransmitRecive(u8 Copy_u8Data);
   this function is used to send data or receive data between the two microcontrollers so it takes one input that is the data to send or receive

4. UART:
   ES_t UART_enuSendChar( u8 Copy_u8CharData);
   This function returns an error state to detect the error its used to send a character by uart communication protocol so it takes one parameter the char data you want to sent

In MCAL layer in slave:

1. ADC:
   int ADC_Read(char channel);
   this function reads the digital value from adc channel so it takes one parameter that is the channel number and returns the adc read

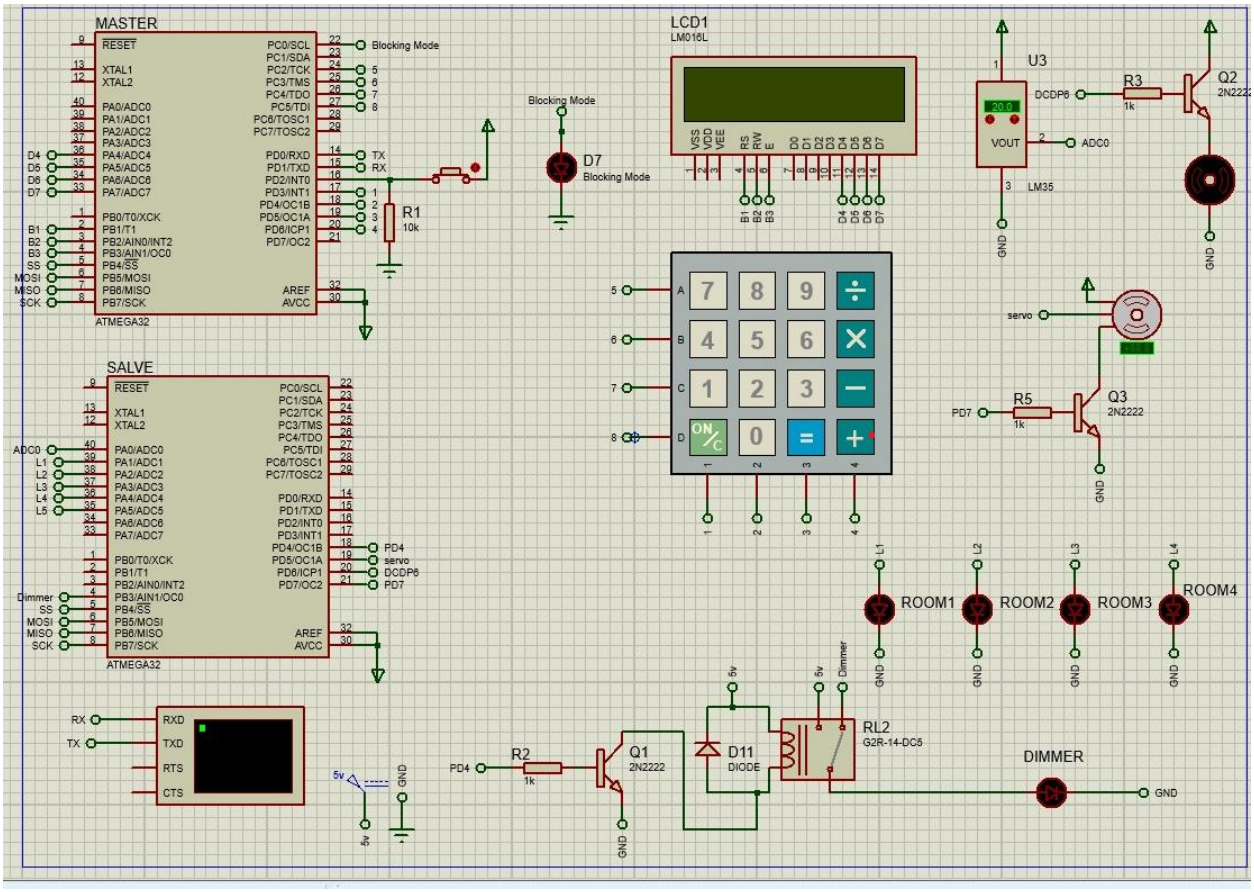
2. Timer1:
   ES_t Timer_PWM(float Copy_u16Freq ,float Copy_u8DutyCycle );
   This function generates a timer with the frequency and duty cycle you want so it returns an error state to detect the error and takes two parameters the frequency and duty cycle values

## In HAL layer:

1. KEYPAD:
   ES_t Keypad_enuGetPressedKey(u8 * Copy_pu8KeyValue);
   This function gets the pressed value on keypad and it returns an error state to detect the error

2. LCD:
   ES_t LCD_enuSendData(u8 Copy_u8Data);
   This function is used to display a character on the lcd so it takes the character to be displayed and returns an error state to detect the error

The Connection of components on proteus:




Link of Github for the project :

https://github.com/hamdy24/Smart-Home-Project