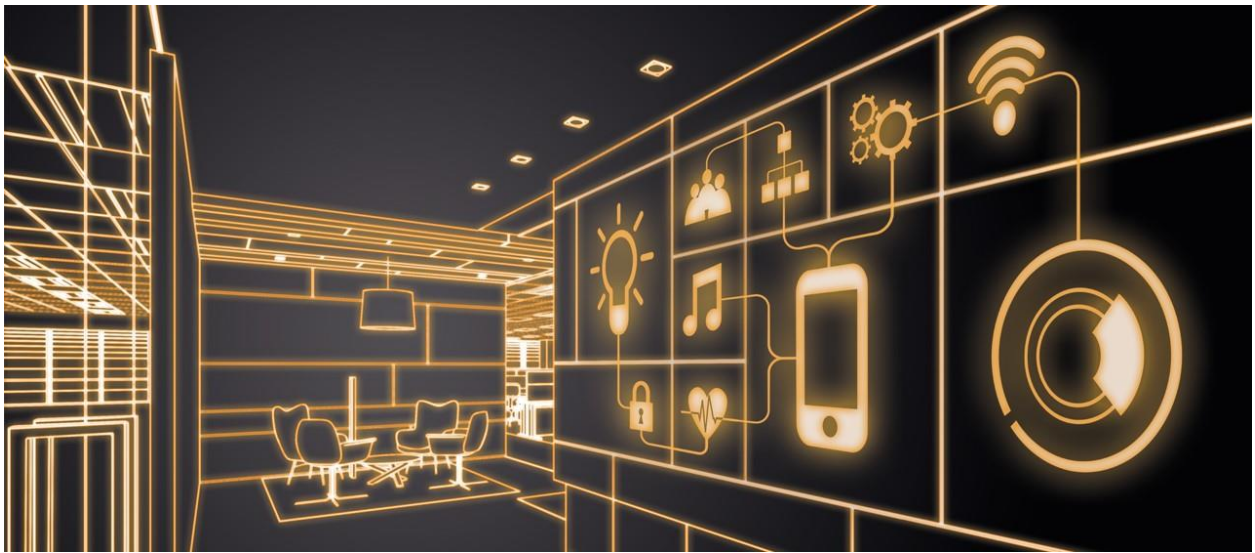# Smart Home Project

Name: Mohamed Abdou Mahmoud

Group: Alex20

Instructor: ENG. Hossam Eltokhy

Github Link:
https://github.com/MohamedAbdouMabrouk/Amit_FinalProject
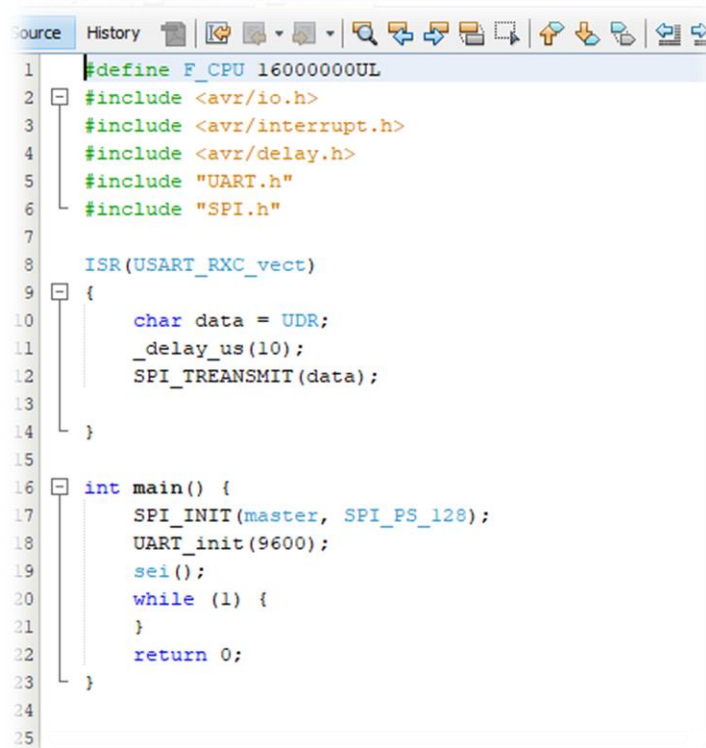
The project consists of two parts:

_____

1- Master:

This part of the project receive data from (Bluetooth OR Terminal) using UART communication protocol and send the data using SPI communication protocol to the slave (part 2).

- Syntax:

When the program start, the program will initiate the UART and SPI communication protocols using UART and SPI drivers and enable the interrupt and wait for any data coming from the UART interrupt, then send the data to the slave using the SPI.

```
1   #define F_CPU 16000000UL
2   #include <avr/io.h>
3   #include <avr/interrupt.h>
4   #include <avr/delay.h>
5   #include "UART.h"
6   #include "SPI.h"
7
8   ISR(USART_RXC_vect)
9   {
10      char data = UDR;
11      _delay_us(10);
12      SPI_TREANSMIT(data);
13
14  }
15
16  int main() {
17      SPI_INIT(master, SPI_PS_128);
18      UART_init(9600);
19      sei();
20      while (1) {
21      }
22      return 0;
23  }
24
25
```
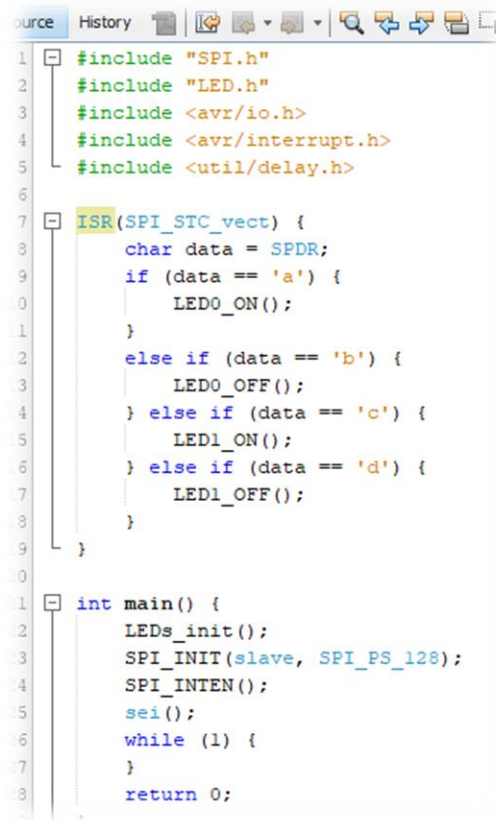
2- Slave:

After receiving data from the master (part 1) using SPI communication protocol and turn on/off the LEDs.

- Syntax:

When the program start, the program will initiate the SPI communication protocol and the LEDs using SPI and Led drivers and enable the interrupt and wait for any data coming from the SPI interrupt, then do one of the following tasks:
  - Turn on LED 1 if the data was 'a'.
  - Turn off LED 1 if the data was 'b'.
  - Turn on LED 2 if the data was 'c'.
  - Turn off LED 2 if the data was 'd'.

```c
#include "SPI.h"
#include "LED.h"
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>

ISR(SPI_STC_vect) {
    char data = SPDR;
    if (data == 'a') {
        LED0_ON();
    }
    else if (data == 'b') {
        LED0_OFF();
    } else if (data == 'c') {
        LED1_ON();
    } else if (data == 'd') {
        LED1_OFF();
    }
}

int main() {
    LEDs_init();
    SPI_INIT(slave, SPI_PS_128);
    SPI_INTEN();
    sei();
    while (1) {
    }
    return 0;
}
```