Åbo Akademi University
Programming embedded systems 2015 course
Project requirements specification

Student name: Vu Nguyen (71141)
Gao Yuan (71142)

Equipment used

- Arduino Uno (reference: http://www.arduino.cc/en/Main/arduinoBoardUno) (x1)
- Push buttons (reference: https://www.sparkfun.com/products/97) (x2)
- LEDs (x3)
- 16x2 LCD (x1)
- TSL2561 light sensor (reference: http://www.adafruit.com/datasheets/TSL256x.pdf) (x1)
- DHT11 humidity and temperature sensor (reference in English and Chinese http://www.micropik.com/PDF/dht11.pdf http://www.adafruit.com/datasheets/DHT11-chinese.pdf) (x1)

Hardware connection

DHT11 humidity and temperature sensor

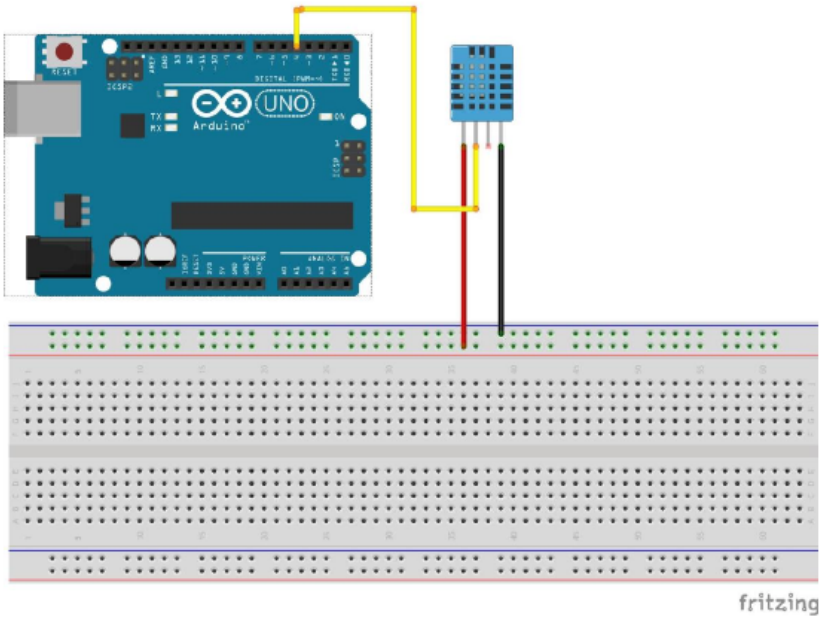

Figure 1. Connection between DHT11 sensor and Arduino Uno

In the project, the DHT11 sensor shall be connected to Arduino Uno via Arduino digital pin 4. The 5V power supply for the sensor shall be taken from the prototype board. Pin 4 of the sensor will be connected to the ground of the prototype board.

TSL2561 light sensor

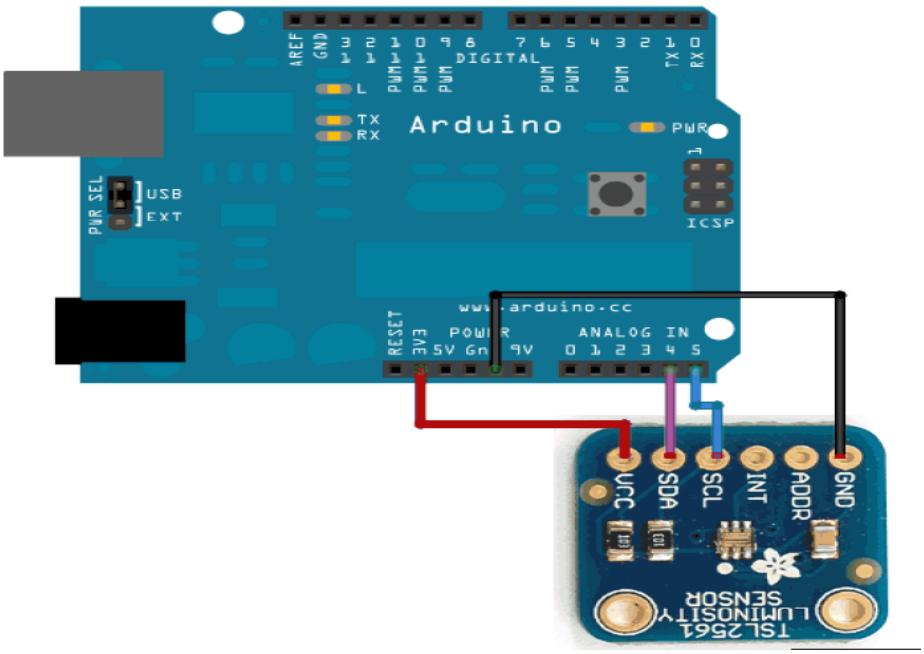

Figure 2. Connection between the TSL2561 digital light sensor and Arduino Uno

The I2C pins on the Arduino Uno platform is mapped to analogue pins 4 and 5. Analog pin 4 on Arduino Uno shall be connected to the data pin (SDA) of the sensor, and analog pin 5 on Arduino Uno shall be connected to the clock pin (SCL) of the sensor. The 5V power supply and the ground for the breakout board of the sensor are taken from the prototype board.

User interface

The user interface consists of a 16x2 LCD, 2 push buttons with Schmitt trigger circuits for debouncing the buttons, and 3 red, green, and white LEDs. The following block diagram is used to better illustrate this user interface circuit. The following figure shows the user interface of the system.

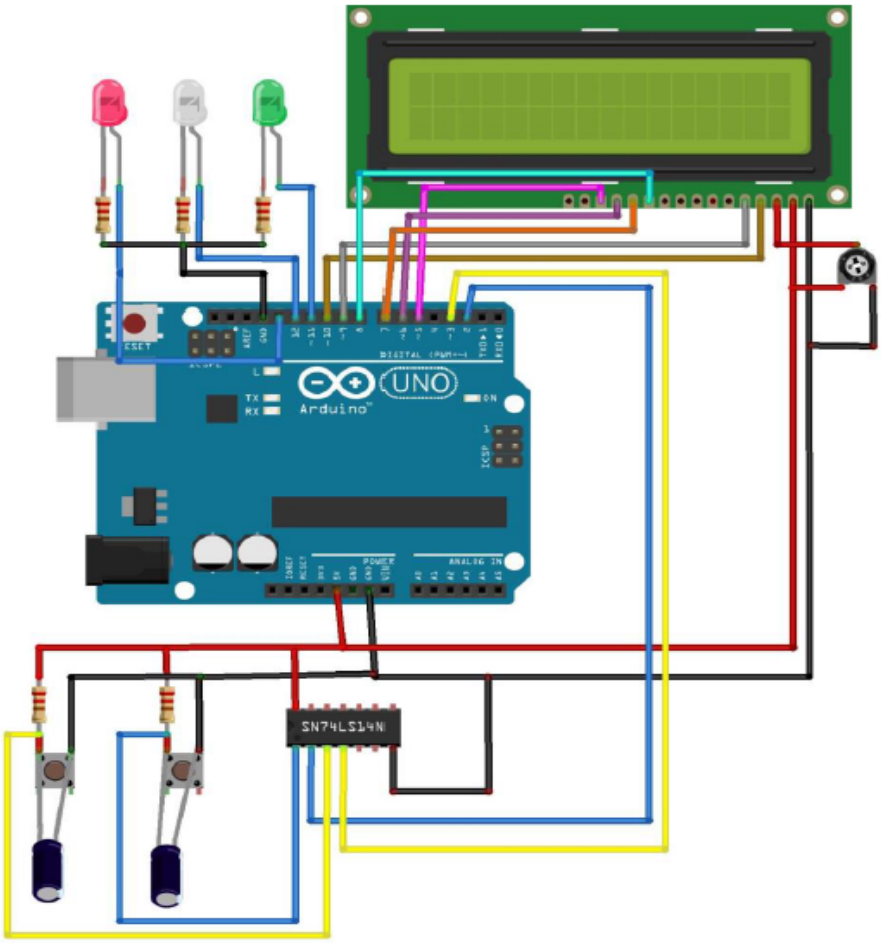

Figure 3. User interface of the system

The 16x2 LCD is connected to the Arduino Uno according to the mapping as follows:

LCD pin name	**LCD pin**	**Arduino Uno pin**	**Potentiometer pin**
Vss	1	GND	3
Vcc	2	5V	1
Vo	3		2
Rs	4	10	
R/W	5	GND	
E	6	9	
D4	11	8	
D5	12	7	
D6	13	6	
D7	14	5	

Pin 2 of the 10K potentiometer is connected to pin Vo of the LCD to adjust the contrast of the LCD. The button on the left-hand side in Figure 3 is connected to the Schmitt trigger circuit, which in turn is connected to Arduino Uno pin 2. The button on the right-hand side in Figure 3 is connected to also to the Schmitt trigger circuit, which in turn is connected to Arduino Uno pin 3. These buttons are used by a user to set the temperature and light intensity level and to turn on/off the light manually regardless of the light intensity level already set in the controlling unit. The green LED shall be connected to Arduino Uno pin 11. The white LED shall be connected to Arduino Uno pin 12, and the red LED shall be connected to Arduino pin 13 as seen in Figure 3.

Description of the operation of the system

Display of system information

The16x2 LCD shall be used to display information about the system to the user. It also displays the interface for the user to set temperature and light intensity level as well as turn the light on/off regardless of the light intensity level. Specifically, it shall have five display modes, namely mode1, mode2, mode3, mode4, and mode5. The first mode (mode1) displays to the LCD information about the current temperature, humidity and light intensity level (in lux). All of these information must be displayed on the LCD at the same time. Mode2 displays the interface about temperature selection. In this mode, the user can set a desired temperature by pressing the right-hand side button. Mode3 displays the interface about humidity selection. In the same fashion as mode2, the user can set the humidity level by using the right-hand side button. Mode4 displays the interface about light intensity level selection. Like mode2 and mode3, this mode allows the user to select the light intensity level. For example, he/she can set the light intensity level to 450 lux which is the ideal level in an office. The last mode (mode 5) allows the user to turn light on/off manually regardless of the light intensity level read from the TSL2561 light sensor. The left-hand side button seen in Figure 3 shall be used by the user to select modes (e.g, mode1 or mode 2), while the right-hand side button shall be used to set temperature, light intensity, humidity, and turn the light on/off manually. Reading the status of the buttons should be implemented using interrupt since it is considered a priority. The following gives a rough interface of each mode that a user shall see on the 16x2 LCD.

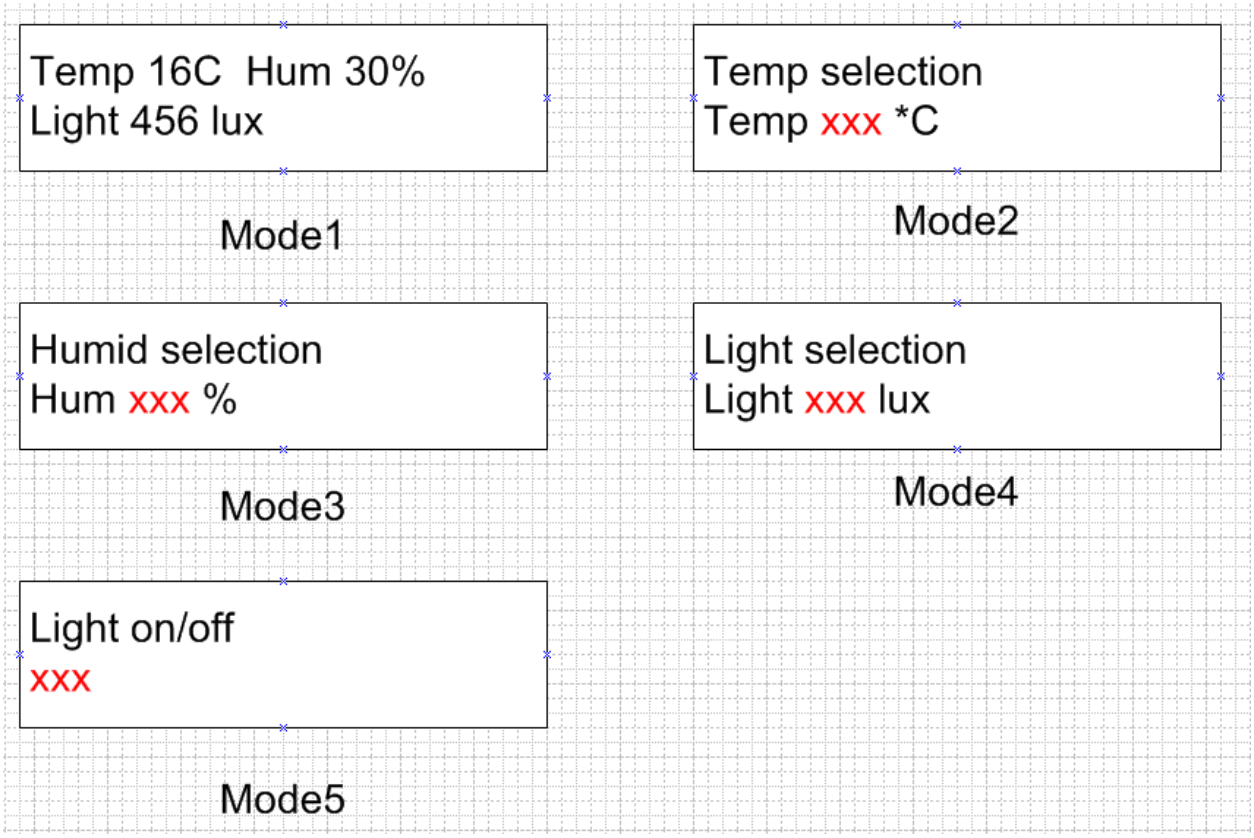

Figure 4. Selection modes on LCD

The xxx in red is the variable that will be displayed on the LCD when the user selects temperature, light intensity level, humidity or light status (e.g, on or off).

Control of temperature, light intensity, and humidity

After power-on, the system will keep controlling temperature, light intensity and humidity. It compared the values read from the sensors with the value set by the user to turn on/off the LEDs accordingly. When the temperature, light intensity and humidity levels are higher than their threshold values set in the system, the controlling unit will turn off the corresponding LEDs. When these parameters are lower than their threshold values set in the system, the controlling unit will turn on the corresponding LEDs.

Design of software

This section should not be part of the project specification. However, I put it here so that it is easy for you to have an idea how the software should be developed.

Low-level function prototypes for application development

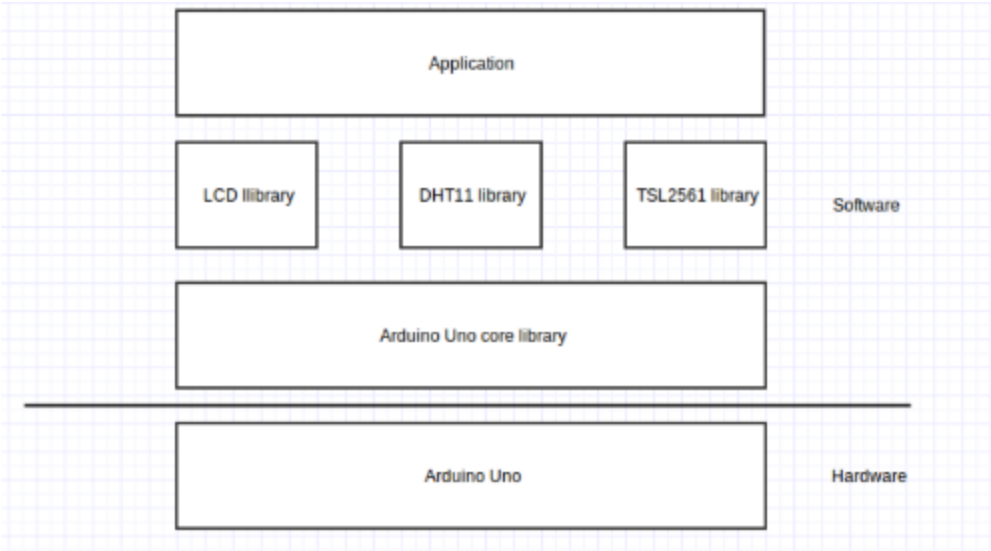

Figure 5. Abstraction layer for software development

In this project, the prefix PES_ should be used to name functions and class since it is easy to know that we the ones with PES_ prefix are coded by us. Application software shall be built on top of the libraries. The application read temperature, humidity, and light intensity by calling functions from the libraries. It also uses Arduino Uno core library when needed. Following are preliminary function prototypes offered from the low-level sensor library.

DHT11 temperature and humidity sensor library functions

Header file name: PES_dht11.h
Class name: PES_DHT
Member functions:

```
//Set up the pin of the DHT11 on Arduino Uno
void begin (uint8_t dht_pin);

//Read humidity from DHT11
uint8_t read_humidity (void);

//Read temperature from DHT11
int8_t read_temperature (void);
```

TSL2561 light sensor library functions

Header file name: PES_tsl2561.h
Class name: PES_TSL2561
Member functions:

```
//This function configures the sensor; check if the sensor is connected,
//set gain and integration time. It returns true if everything is OK, false if something wrong
happen
boolean begin(void);

//Read light intensity level in lux
uint32_t read_light_intensity_lux (void);
```