Final Report for Group#30

REMOTE CONTROL LED WITH TIME/TEMPERATURE/DATE DISPLAYING LCD SCREEN AND TOUCH SENSOR

By

Yuming Jin
Jintao Chen
Jingfan Zhang

December 11,2019

Abstract

This report reviews and summarizes what we have done in this semester to make a remotely controlled LED system with cool features such as twelve different brightness levels, humidity, temperature and time displayed in two large LCD screens.

Contents

1. Introduction

While most of the LEDs that we commonly used are being controlled by a single wired switch, sometimes when we are right about to fall asleep in our cozy bed, especially in the winter, we find it so tormenting to get up, reach the switches, and turn off the LEDs. A friendly designed wireless bedside system that controls all the LEDs around the house would avoid such cases.

We came up with this idea when one member in the team, Jintao, moved to a new apartment without a good illuminating system. Then we added features to add difficulty and function of the design to make it a feasible ECE445 level project.

Since some people might prefer other bedside functionalities as well, we initially proposed to create a multi-functional user-friendly remotely controlled LED system that has some designed features including:

1. Bed-side with the master light switch using capacitive touch and a LCD screen displaying the current time, room temperature and room humidity.
2. Wall-side (LED) with the slave switch and two knobs for brightness level up and brightness level down respectively, with a LCD screen displaying the same current time, room temperature and room humidity.
3. 433 MHz FM transmitter and receiver wireless communication unit to connect and transfer data between the bed-side and the wall-side.

Twelve LEDs brightness level controlled by a power MOSFET, BJT and resistor combined analog circuit; brightness level is saved each time before turning off so that it remains in the same brightness level when turned on next time.

2. Design

2.1 Design Procedure

We formed the idea at the very beginning of the semester when we were writing RFA and proposal and communicating with TAs and professors. For remote control, we used to hesitate between Bluetooth and FM module. We finally chose FM over Bluetooth as we decided that FM module was more technically viable than Bluetooth and that we were more familiar with FM module. However, we did not know what sensors to use for clock time, temperature and humidity display at that time.

To give a brief view, our block diagram looks as follows:

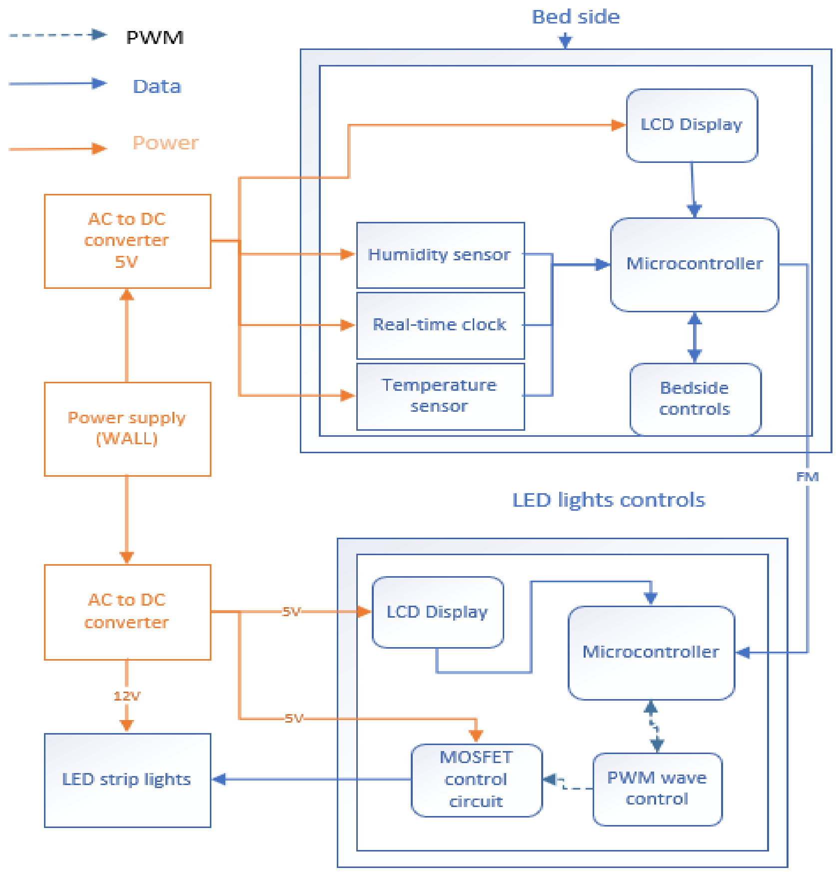


Figure 1 Block Diagram

We originally decide to have two PCBs, with one for bed-side and one for wall side.

For different sensors, we planned to use 3.3V, 5V and 12V ac to dc voltage converter to provide power. We eventually ended up using only 12V for brightness control circuit and 5V for Arduino Mega 2560. For time display, since we wanted our time to be accurate we had steadily decided that we would use DS3231 real-time clock crystal which has automatic compensation for leap-years and for months with fewer than 31 days. For brightness control, we planned to use MOSFET combined with PWM wave control using codes stored at arduino pro mini to output different power for different brightness levels, resulting in the brightness-varying performance of wall-side LED strip.

2.2 Design Details

Present the detailed design, with diagrams and component values. Show how the design equations were applied. Give equations and diagrams with specific design values and data. Place large data tables in an appendix. Circuit diagrams that are too large to be readable on a single page should be broken into pieces for presentation. The full diagram may be included in an appendix. Use photographs only as necessary and treat them, along with all other graphics except tables, as *figures*.

Our initial plan was to make our system with a physical outlook shown in the following figure.

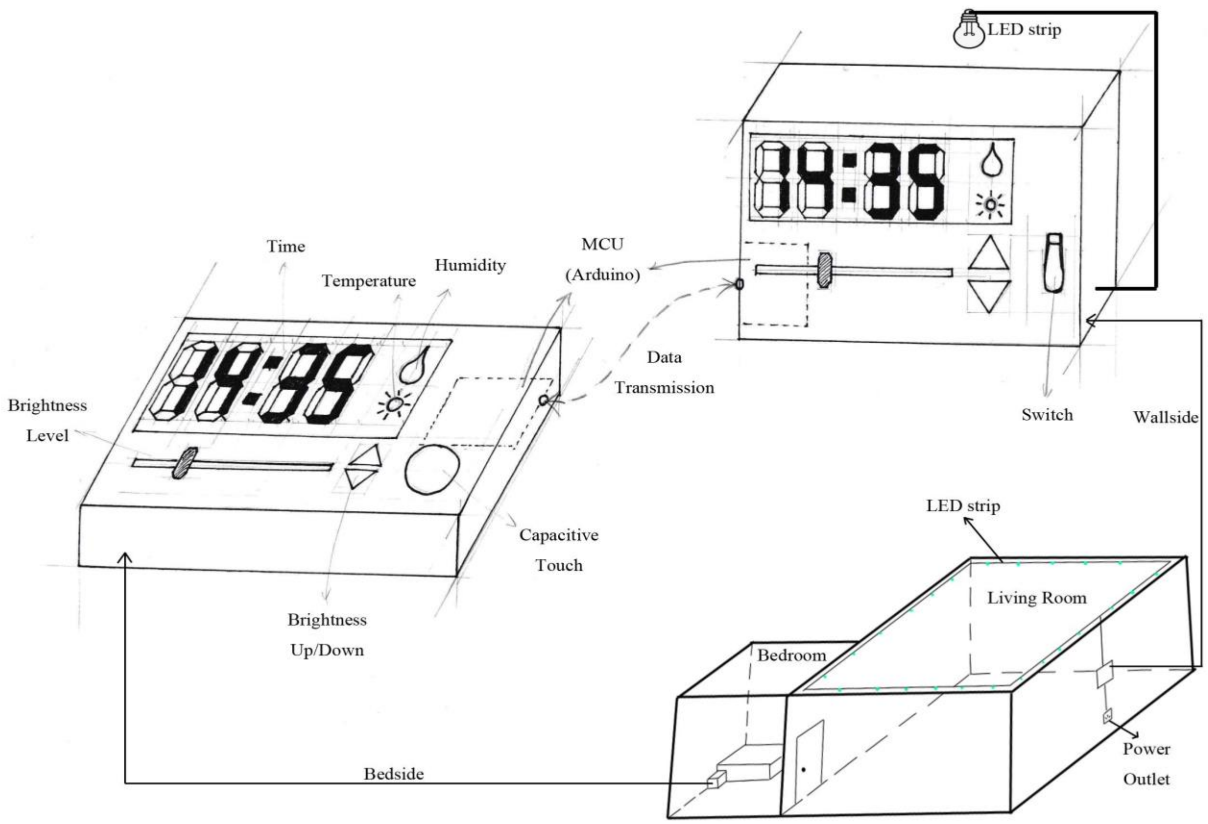


Figure 2 Visual Aid (Physical Design)

Both sides of the LCD screens show the time, humidity and temperature of the room. Brightness control switch consists of three knobs, the master knob using capacitive touch, the brightness up knob and the brightness down knob.

However, our actually design ends up being different from our initial design, as is shown in the following figure:

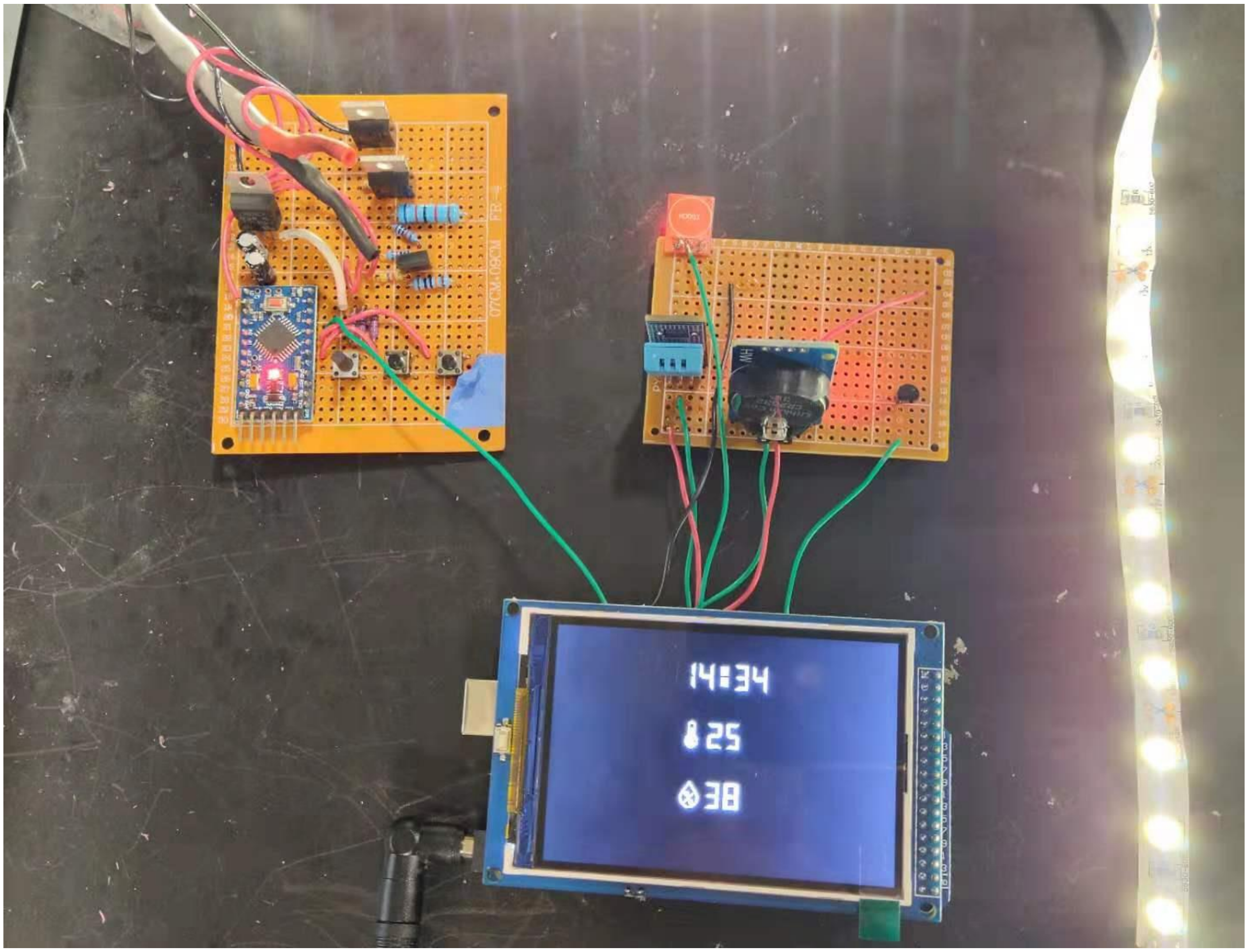


Figure 3 Demo Day Layout

As is shown in the demo, on the bottom we have a LCD screen connected with a MEGA2560 board for the bedside. It is displaying time in 24h format read from DS3231 real-time clock, temperature in degree Celsius which comes from the reading of the LM35dz temperature sensor, and humidity level in percentile which comes from the reading of DHT humidity sensor.

On the upper right PCB laminated board, we have three sensors outputting data to display humidity, temperature and clock; and a capacitive touch which acts as a master switch to control the turning on and off of LED strip.

On the upper left PCB laminated board, we have our brightness control circuit.
The brightness control unit is shown in the following circuit

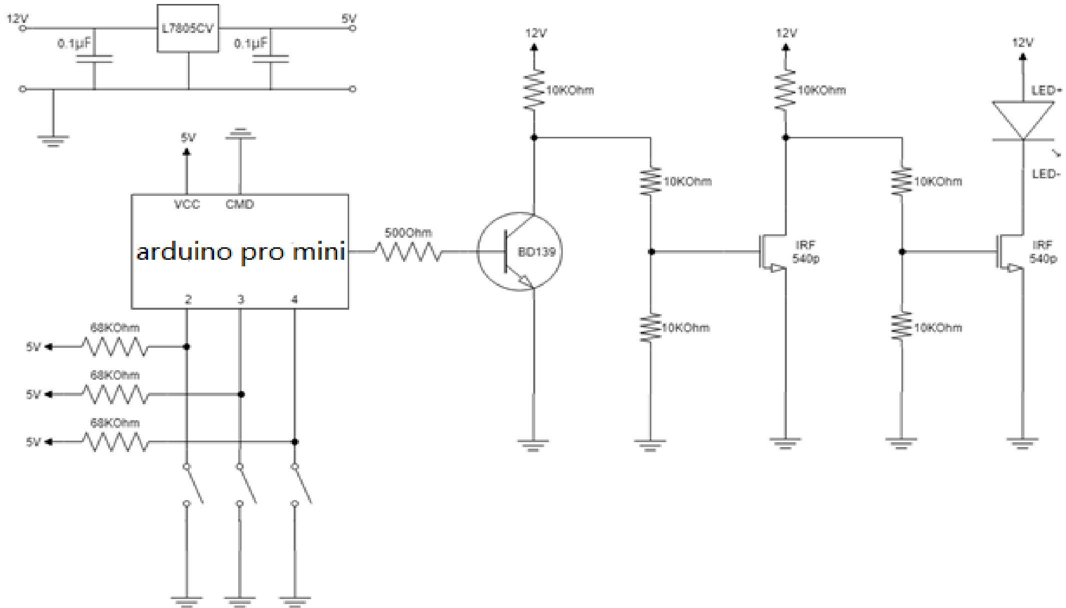


Figure 4 Brightness Control Circuit

On the top right part, L7805CV electronic parts distributor has three pins, with its input pin takes in 12V dc voltage and outputs 5V dc voltage from the output pin, and its GND pin is grounded. We add two 0.1uF capacitors on both input and output as filters.

For arduino pro mini, we are using pin2 as the main switch to turn on and off LED strip, pin3 as brightness up switch and pin4 as brightness down switch. The GND pin is grounded and Vcc takes in 5V dc voltage generated by L7805CV and its output pin11 sends data to the amplifier circuit to provide input power for LED strip.

We define our PWM control unit to have twelve brightness levels, for better user experience.

3. Verification

3.1 LM35 Temperature Sensor

A temperature sensor that shows real time temperature to be put in two LCD screens.

Requirement	Verification
1. Functional under normal indoor temperature level (20°C to 25°C) (5 points) 2. Functional under 5V supply voltage	1. See if it’s reading the correct temperature, or see if temperature value changes when we touch it for a long enough of time.

Table 1 LM35 R&V

For temperature sensor, we directly tested its functionality by touching it with a finger and check if its temperature raised gradually. Since it would raise gradually each time, we touched it and the starting temperature was always around 24 degree Celsius, close to room temperature, we concluded that it was working.

3.2 DS3231 Real-time Clock

A real-time clock that gives very accurate time information to be indicated in two LCD screens.

Requirement	Verification
1. Outputs time under normal indoor temperature level (20°C to 25°C). (10 points)	1. If the clock functions well, it should output hour and minute.

Table 2 DS3231 R&V

For DS3231, we tested its accuracy by check if it's moving to the next minute every time after 60 seconds. We also checked if it drops to 00:00 after passing 23:59. It was observed to be functioning well.

3.3 TTP223B Capacitive Touch

The capacitive touch should allow the users to turn on and turn off remote-controlled LED strips without directly touching the switch. This is the most complicated and important sensor in all these four sensors.

Requirement	Verification
1. Operates under 5V DC voltage supply to be used to control the turning on and off of LED (10 points) 2. Functional under normal indoor temperature level (20°C to 25°C).	1. Turn it on and off to check if it controls the on and off of LED strip.

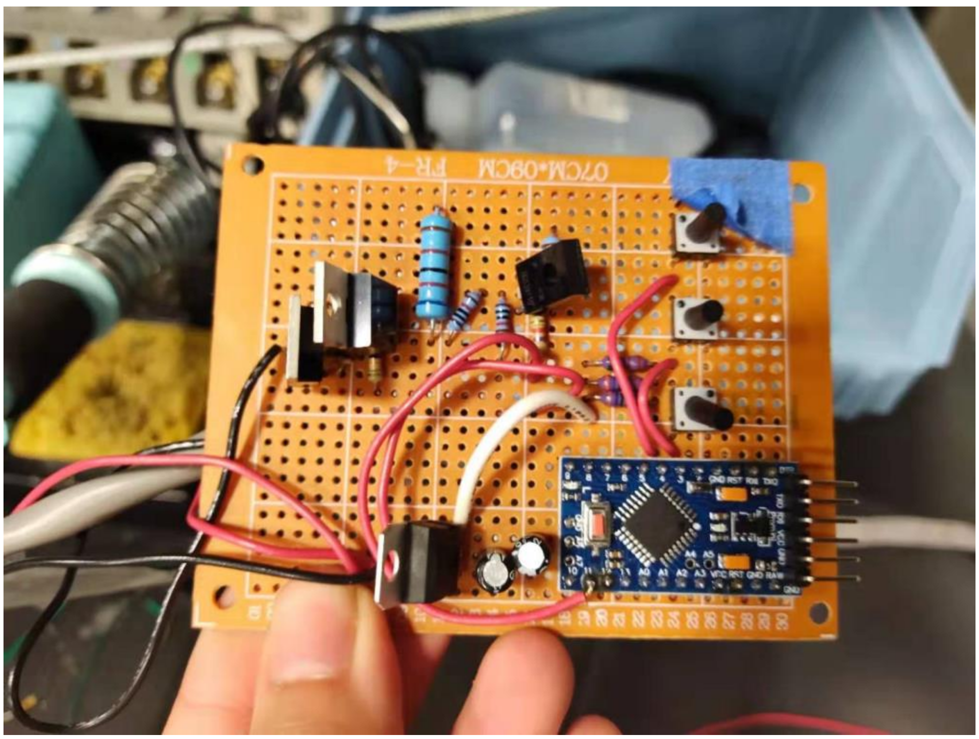
Table 3 TTP223B R&V

3.4 Brightness control circuit

Requirement	Verification
1. Works well independently to control LED brightness with 12 brightness levels (15 points) 2. Saves brightness level each time when turned off (5 points)	1. Check if the circuit controls the LED with 12 different brightness levels 2. Check if turn-on knob, brightness-down and brightness-up knobs work well.

Table 4 Brightness control R&V

For brightness control unit, we tested by observing the brightness change of LED strip when we press knobs to change brightness level, and we turned on and off our circuit to check if it had the same brightness as last time when turned on. We counted a total of 12 different brightness levels, which exactly fit into what we were expecting. The brightness control board is shown in the figure below:



3.5 LCD display

Requirement	Verification
1. LCD screen displaying the correct humidity level, time and temperature read from other sensors (10 points)	1. Provide power and see if the screen is displaying correct humidity level, time and temperature level.

Table 5 LCD R&V

The 2 LCD screens we have are powered by MEGA2560 board. We test them by connecting them with a MEGA2560 board and write the sample LCD display drawing code to the boards. After changing the libraries and adjusting the code, we finally having the screens displayed in the right format.

3.6 HDC2010 Humidity Sensor

For humidity sensor, we first carried out direct testing by putting finger to it and blowing hot air into the sensor's detecting surface and see if its humidity level would change. The original value, when demoing, was 38 percent. After blowing hot air to it the value increased to 40 percent, which indicated that it was indeed working. Prior to the demo we also tested it using the DHTtester library.

4. Costs

4.1 Labor Cost

We chose average salary of $40 an hour for an ECE graduate from University of illinois. For ECE445 we estimated in total we spend around 5 hours/week for 16 weeks this semester.
(20$/hour) x (6 hours/week * 16 weeks) * 2.5 = $4800.

4.2 Parts Cost

Parts	Price ($)
3 ATMEGA328-AU	3 x 1.83 = 5.49
2 LM35 Temperature sensor	2 x 0.44 = 0.88
2 DM3231S#-ND Integrated circuits	2 x 8.63 = 17.26
10 TTP223 Capacitive touch sensor	0.9
2 L7805CV electronic parts distributor	2 x 0.5 = 1
2 LED strips	2 x 25 = 50
4 HDC2010YPAR Humidity sensor with temperature sensor	4 x 0.99 = 3.96
1 3.3V 1A AC adapter to DC power adapter, 5.5/2.1mm	5.80
1 5.5V 2A AC to DC power adaptor, 10W for LED strip lights	6.99
1 12V 2A power adaptor AC to DC 2.1mm x 5.5mm plug	7.99
1 6-foot Power strip with 6-outlet surge protector	9.98
2 Mega2560 20x4 LCD display module with screen panel expansion board white on blue, 4 pin jump cables	2 x 12.39 = 14.78
2 Voltage regulators	2 x 0.44 = 0.88
2 Bluetooth module V4.2	5.99 x 2 = 11.98
2 DS3231 board	2 x 5 = 10
2 Arduino Mega 2560	2 x 15.00 = 30
1 Arduino Uno	30
1 Arduino pro mini	10
Total Price	217.89

5. Conclusions

General conclusion

- For the project, we basically have every part functioning and working independently. However, when we fit them together as a whole, we burned out our FM model so we were not able to demo the information exchange between the FM transmitter and receiver modules. But we did make videos showing their functionality. We also made videos of brightness control unit, LM35 temperature sensor, DS3232 real-time clock and DHT humidity sensor and these parts all ended out to be working well in the demo.
- The overall design is successful, and we made progress step by step throughout the semester. We also spent time investigating and then decided that we would want to use FM module rather than Bluetooth, which might need much more code than the FM module to function well.

We do think we did a fine job on the project.

However, if we had more time, we can further improve our project in the following ways.

- Apply a better, working FM module which will not be burned out and exchanges signal in a longer distance such as more than 10 meters so that our design would cover a larger effective range.
- More stable capacitive touch circuit to gain better control on the switch sensitivity so that the LED system will be more stable to be turned on and off, without blinking.
- Better overall design and so that we can replace our perf boards with two PCBs, one on the bed-side and the other one on the wall side.
- Try to use Atmega328P directly rather than Arduino Mega2560 and Arduino pro mini to add difficulty of the design.

6. References

[1] Siemens Digital Alarm Clock with Thermometer, Online, url:https://support.industry.siemens.com/cs/attachments/61600566/197835a_201207161405130338.pdf

[2] LM35 ±0.5°C Temperature Sensor with Analog Output and 30V Capability, Online, url:http://www.ti.com/lit/gpn/lm35

[3] DS3231 Extremely Accurate I²C-Integrated RTC/TCXO/Crystal, Online, url:https://datasheets.maximintegrated.com/en/ds/DS3231.pdf

[4] TTP223 Capacitive Touch Switch Circuit, Online, url:https://cdn.solarbotics.com/products/datasheets/ttp223.pdf

[5] ATmega328P - 8-bit AVR Microcontrollers, Online, url:https://www.microchip.com/wwwproducts/en/ATmega328p

[6] L7805CV STMicroelectronics, Online, url:https://www.digikey.com/product-detail/en/stmicroelectronics/L7805CV/497-1443-5-ND/585964

[7] IEEE Policies, Section 7 - Professional Activities (Part A - IEEE Policies) - 7.8 IEEE Code of Ethics, Online, url:https://www.ieee.org/about/corporate/governance/p7-8.html