Anastasiia Kravtcova

Development of cooling HVAC-type system based on thermoelectric effect

Helsinki Metropolia University of Applied Sciences

Bachelor of Engineering

Degree Programme in Information Technology

Thesis

31st of March 2015

Author(s) Title Number of Pages	Anastasiia Kravtcova Development of cooling HVAC-type system based on thermoelectric effect 55 pages + 3 appendices
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Embedded Engineering
Instructor(s)	Raimo Miettinen, Project Manager Anssi Ikonen, Principal Lecturer

The project was carried out in a company with the purpose to develop a cooling HVAC-type system, based on thermoelectric technology into a special application. The project was completed over a period of half a year from August 2014 to January 2015, and will be developed later on. The major goals of the project were achieved, and further steps were figured out.

This project was implemented in several steps. Research of the proposed technology was the initial point. In the beginning, no clear solution was known, so the current market situation study was completed as well. On that basis the development of the final product was done. Evaluation of advantages and disadvantages, further improvements and aims finalized the project.

Thermoelectric effect was stated as the basic technology for the cooling system. The specific application was designed according to the purpose of integration the system into the company's product. Two parts were included in the project. The first one was cooling the device physical prototype design. The second part was Arduino-based control system implementation. Those steps were successfully completed in accordance with the plan.

All in all, the project was completed successfully. The cooling system based on a thermoelectric principle prototype and its control system were implemented, and integrated into the company's main product. However, obvious limitations of the system were analysed, and possible solution for improvements was offered.

Keywords	Thermoelectric effect, Peltier element, Arduino, Grove shield, control system, HVAC system

Contents

Appendices

Appendix 1. Code Version 1

Appendix 2. Code Version 2.

Appendix 3. Code Version 3

1 Introduction

Over the last century, technologies such as heating, ventilation and air conditioning (HVAC) and thermoelectric effect went through a significant development process, from window conditioners to energy-efficient, fully controlling and cost-available systems. Thermoelectric effect got its place in areas such as refrigerating and cooling of electronics. Those processes are constantly moving forward today as well.

The thermoelectric (TE) technology started its history in the 17th century, and has been developing over the ages. With time passing, many discussions were conducted, following which many disadvantages of the TE technology were found. Despite all negative observations and opinions, the TE principles are used in many areas, and in HVAC systems as well.

Nowadays, HVAC systems are most commonly associated with indoor conditioners and climate control car systems. However, this subject includes much more scope. The HVAC technology is a complicated area, which includes several physics and engineering lines such as thermodynamics, fluid mechanics and heat transfer. HVAC might be found both in traditional and unconventional applications.

The aim of my project was to study and investigate the TE technology principles and implement them into a cooling HVAC-type system used in a particular company’s application. The control system design was the second major part of the work. After that, the physical prototype of the designed system was assembled. Evaluation and testing, as a result of the work done, were the final steps in this thesis.

2 Overview of thermoelectric effect

The thermoelectric effect is a process of the production of electromotive force out of temperature difference on two conductor- or semiconductor-based electronic components, and backwards-temperature difference obtained by applying a DC power. In the second backward case, heating and cooling are controlled by changing a polarity on the voltage source.

Thermoelectric materials consist of holes and free electrons, which carry charges as well as heat. In p-type materials where free charges are positive, the cold side has a positive potential, and similarly, with n-type materials where the cold side has a negative potential. Figure 1 illustrates the heat and its potential distribution in thermoelectric material. [1]

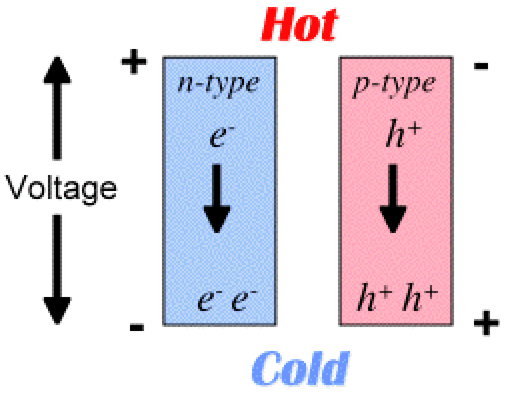


Figure 1. Diagram of charges and heat distribution. Copied from Snyder et al. [1].

The term “thermoelectric effect” contains three topics. Seebeck effect, Peltier effect and Thompson effect are the main components of the thermoelectric technology [8]. The first two of them are described in detail in sections 2.1 and 2.2.

2.1 History overview

Thomas Seebeck was the first scientist who noticed that when two metals with different temperatures are attached to each other, the electric current appears. To explain this process, it is necessary to understand the behaviour of electrons inside the metal. That will be described in more detail further in sections 2.2.1 and 2.2.2. [2]

Several years later Jean Peltier noticed the backward process. He managed to create a temperature difference by applying a current through two metals. Moreover, the continuous version of the Peltier effect was observed by Lord Kelvin. In this process, the heating or cooling is going with a temperature gradient, which occurs when different materials are used. [2]

2.2 Seebeck effect

The Seebeck effect was the first step in thermoelectric theory development [2]. In 1821, Thomas Johnathan Seebeck noticed that in a circuit made of two different metals, the compass arrow turned. He decided it to be magnetism effect. Nevertheless, later he induced that there was a current appearing, and the voltage difference was obtained [19].

In semiconductors, some electrons are floating freely and are unbound. Their amount, or density, varies in different materials. For better understanding, refer to Figure 2.

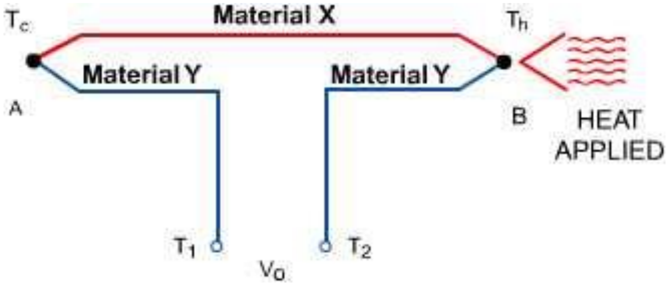


Figure 2. Operation principle of the Seebeck effect. Copied from Ferrotec Corporation website [19].

As figure 2 illustrates, when two different materials (X and Y) are placed together and heat is applied on one side of the system (point B, *Th*), free electrons start to relocate, and the metal becomes oppositely charged, while the other side of the material becomes cold (point A, *Tc*). This causes a potential difference across the junction (T1-T2 junction, *V0* potential). The temperature defines the amount of electron floated, and consequently, the strength of potential. [19]

Voltage *V0* is Seebeck EMF. It can be mathematically expressed as in Equation 1.

$$V0 = Axy * (Th - Tc)\ [V], \tag{1}$$

where *V0* is output voltage in volts, *Axy* is the Seebeck coefficient between two materials X and Y in Volts/Kelvins, *Th* and *Tc* are hot and cold temperatures in Kelvins.

Analyzing Equation 1, the voltage generated by the Seebeck effect depends on the temperature difference achieved, and the materials used. For many semiconductors, Seebeck coefficient is a constant pre-defined value. For example, n-type Bi_2Te_3 (Bismuth telluride) has the Seebeck coefficient of -230μV/K, and for PbTe (Lead telluride) this value is equal to -180μV/K [22]. Consequently, Bi_2Te_3 will allow getting a higher voltage than PbTe with the same temperature difference achieved.

2.2.1 Seebeck effect in thermometers

The Seebeck effect is a platform for thermo-elements and thermocouples. With thermocouples, it is possible to measure temperature, and this technology is used in simple thermometers.

The operation of a thermometer is clearer if it considered it gradually. A basic thermometer consists of a thermocouple which is connected to a voltmeter. The voltmeter displays a thermo-voltage which is strictly dependent on the temperature difference applied to the thermocouple. By converting voltage to a temperature value, the user gets the result of the measurement. This conversion might be done by manual calculations, or by an additional component such as a temperature sensor, which implements an analog-to-digital conversion.

2.2.2 Seebeck effect in power generation

Generating of electric power for special applications can also be implemented by the use of the Seebeck effect. It is possible due to diffusion of charge carriers after the heat transfer over the thermo-element. However, the main disadvantage of this application is that the value of the generated power is usually very low due to a small current. The factual efficiency of this process is only 5-8%.

Although the efficiency of power generators based on Seebeck effect is so low, there are real applications in industry and science. The first one is to get radioisotope energy to power up space crafts. The second one is to get and recycle energy from car and machine waste heat. Those applications, nevertheless, are not independent power sources but are used only as a supportive technology.

2.3 Peltier effect

The Peltier effect was the second step in the thermoelectric processes investigation. In 1834 during the experiments with electricity, Jean Peltier connected together bismuth and copper wires, and then attached them to a battery. After switching the power on, one wire became hot while another became cold. That is what later helped to create a refrigerator. [8]

Overall, the Peltier effect is an inverse of the Seebeck effect. They both are components of one scientific principle, which demonstrates that electricity applied to two dissimilar semiconductors can be the reason for their heat difference. However, the efficiency of this heat production is very low, as an unreasonably big amount of power should be applied to get a valuable temperature difference.

2.3.1 Basics of Peltier effect

For a better understanding and analysis of Peltier effect, several equations and relations were found. Figure 3 represents the Peltier operation principle. The basic impulse for Peltier effect is a current (*I*) flowing through different materials and resulting in temperature differences (*Th* and *Tc),* even if the surrounding temperature remains the same (Figure 3). Consequently, there is a relation between heat production and the materials and current flowing, as indicated by Equation 2. [9; 19]

$$W = (Ax - Ay) * I \qquad [\text{V*A}], \qquad (2)$$

where Ax and Ay are differential Peltier coefficients for different materials in volts, and W is the heat production rate in watts. *W* might be positive as well as negative depending on current direction.

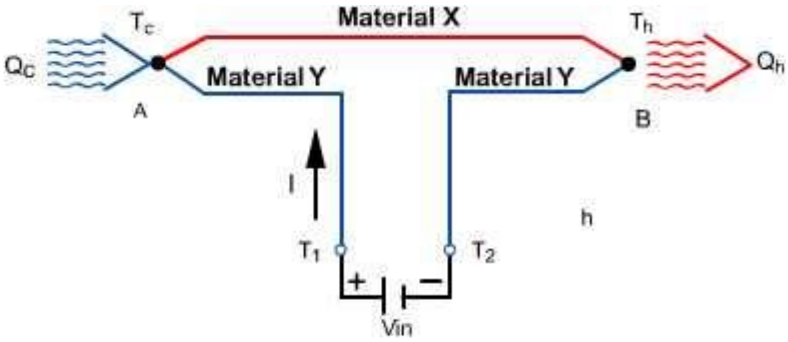


Figure 3. Operation principle of the Peltier effect. Copied from Ferrotec Corporation website [19] [online].

Peltier effect is a reversible technology. A change of the current direction results in different processes. Positive current direction means that the cooling is going on, while negative means that heating is happening. With this specification, heating or cooling can be produced depending on the requirements of the concrete application. [9]

2.3.2 Advantages and disadvantages of Peltier effect

As any technology, Peltier effect has both advantages and disadvantages. They are valuable and have to be estimated in each area where Peltier effect is going to be integrated. It allows previewing if the technology is reasonable to be used, or some alternative is relevant to be found.

Peltier effect is implemented via a Peltier element, which will be presented in section 3. However, the main drawbacks and benefits have to be mentioned at this point to understand the main problems to deal with. Firstly, Peltier element takes a large amount of current and power, while the heat production is not huge. Another disadvantage is condensation. Condensation is obvious when cooling of one side of the element is strong, while the other side becomes excessively hot. Water appearance might cause potentially dangerous situation such as a short circuit, which is most commonly destructive for electronics. [8]

All the negatives aside, Peltier effect has relevant potential as a technology. It is very reliable because Peltier elements do not have any moving parts, and then no maintenance will be needed. Also, Peltier elements are very transportable. In addition, their price is affordable, and is around 20-100 euro depending on the efficiency and power produced. [8]

3 Peltier element

Peltier element is also known as a Thermoelectric Cooler (TEC). That device implements Peltier effect principles, and allows to obtain a temperature difference when powered up, and the current is flowing. Its main advantage is that it provides effective cooling without adding complex machinery or any dynamic parts. [2]

3.1 Materials of Peltier element

The most common combination of materials used in Peltier element are semiconductors Bismuth and Telluride (Bi_2Te_3). Lead telluride (PbTe) and calcium manganese oxide (Ca–Mn–O) can be also used. Silicon Germanium (SiGe) as well as Bismuth-Antimony (Bi-Sb) may be employed in some special cases when exactly these materials are required for use. All those materials are presented in a cubic form of p- or n- types, are located under the copper conductor plates for electrical and thermal contact, and are covered by insulator to provide safe use. [2], [12]

A variety of materials are used to design Peltier element, and their performance is different. Their efficiency varies as well. On different temperatures, the materials show different efficiencies of operation. Figure 4 shows how the performance of three materials depends on the temperature. [12]

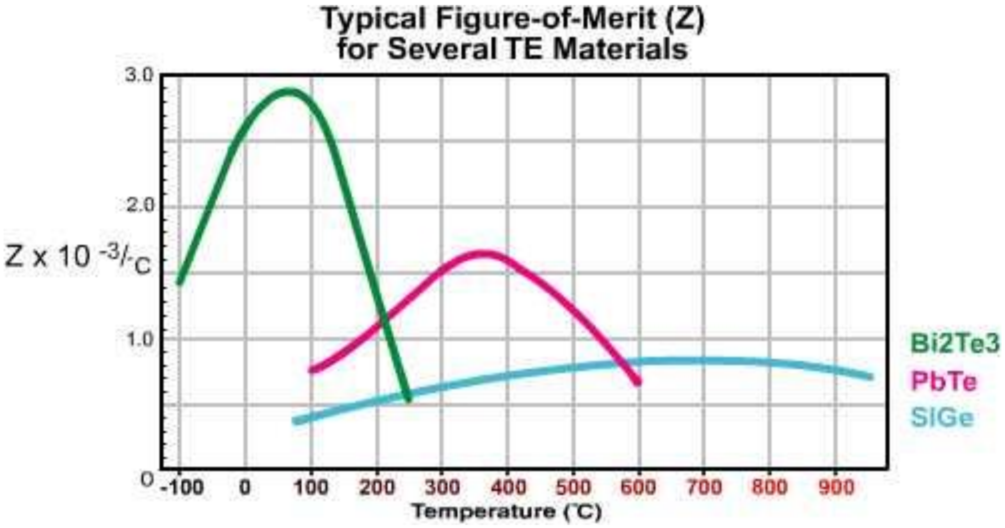


Figure 4. Performance of the thermoelectric materials at various temperatures. Copied from Ferrotec Corporation website [12].

It is clearly visible that Bi_2Te_3 produces the highest peak of temperature. This means that this material can provide the biggest temperature difference, and consequently is the best suited for use in the Peltier element structure. Currently, Bi_2Te_3 remains the most popular fabricating material, and is implemented by major manufacturers.

3.2 Structure and process of Peltier element

Usually, a thermoelectric cooling module consists of two or several parts made of semiconductor material. Those parts, or elements, are joined electrically in series, and thermally in parallel by conducting plates on each side (Figures 5 and 6). Normally, ceramic substrates are placed above these connections to insulate the system. Soldering is applied to connect semiconductor parts and ceramic cover together. This structure is necessary for the first, to provide mechanical stabilization, and for the second, electrically insulate the elements from each other and from other surfaces. [12]

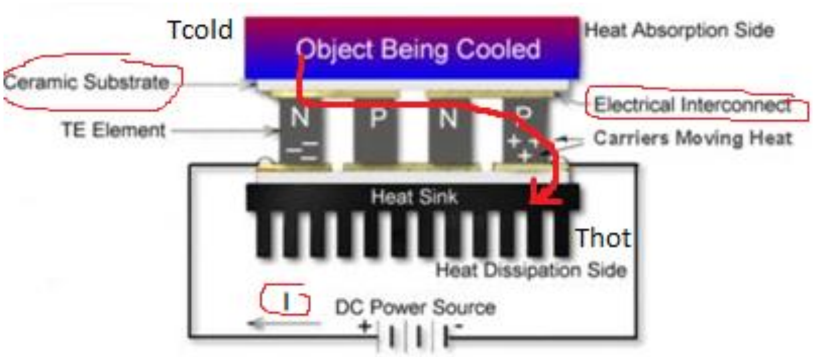


Figure 5. Schematic diagram of a typical thermoelectric cooler. Copied from Ferrotec Corporation website [12].

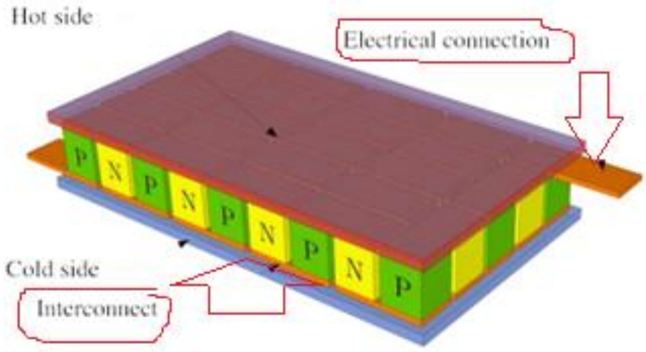


Figure 6. Schematic diagram of a typical thermoelectric cooler. Copied from Warren M. Rohsenow [17].

The size of Peltier elements varies depending on the power of the module. Commonly, the dimensions of a rectangular module are 2.5-50 mm^2 and 2.5-5mm in height. [12] The number of semiconductor parts in Peltier element depends on the element's size. However, for thermoelectric process, it should be a minimum of two: one n-type, and one p-type.

The heat transfer occurs by means of n-type and p-type particles transfer inside the interconnection material. n-type has an excess of electrons which are carrying the heat energy to "holes" of p-type material, which has a lack of electrons [12]. Thus, heat is moving in one direction (from negative n to positive p), while the current is moving in an opposite direction (from positive to negative connection).

Figure 7 shows precisely how heat transfer occurs. A small model of the Peltier element is represented.

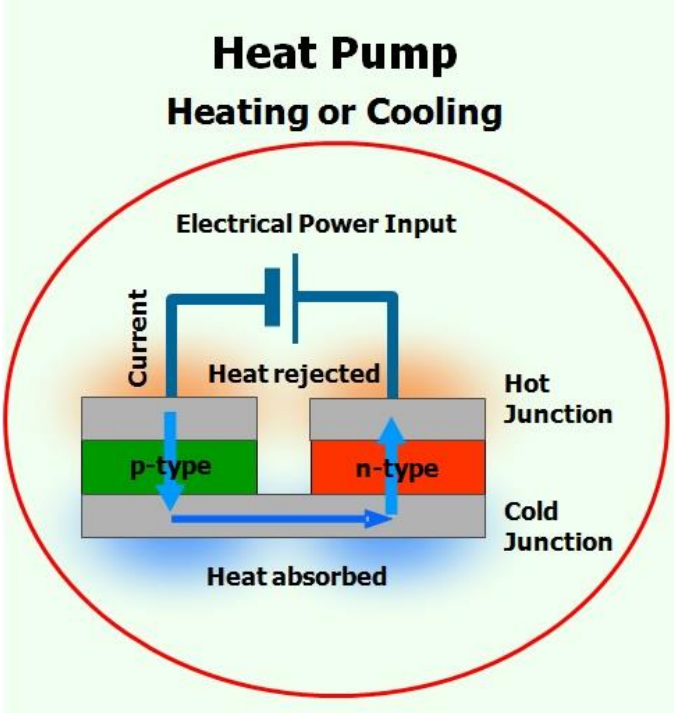


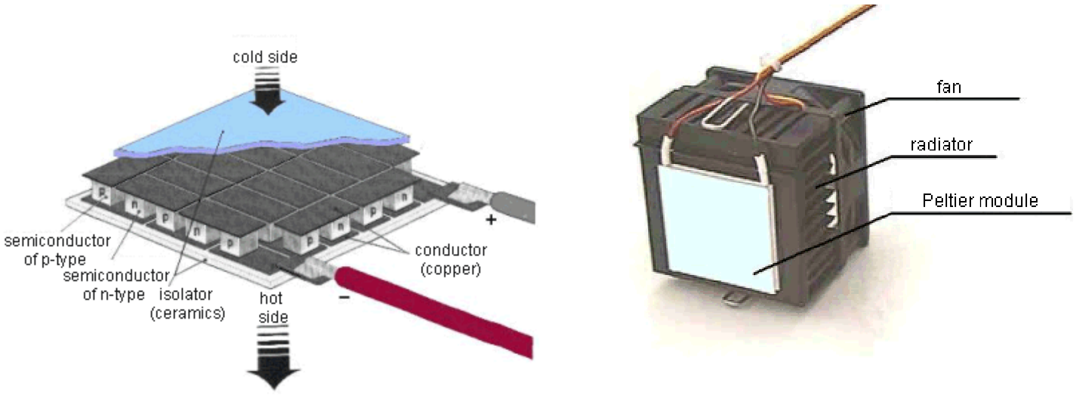
Figure 7. Peltier effect process. Copied from Nat Bowers [20].

The applied current flows from the power input via p- and n- type interconnections, extracting heat from the surroundings, and transferring it from one junction to another. Interconnections are joined electrically in series and thermally in parallel. This process defines the Peltier element operation.

3.3 Operation requirements of Peltier element

Temperature difference on the Peltier module can reach over 50ºC, which can cause device breakage. The reason for that is that the operation temperature for the most of thermoelectric devices is around 80ºC. This limit is defined due to solder applied between semiconductor parts and ceramic cover. At the temperature over 80ºC after prolonged exposure, the solder starts defusing into thermoelectric materials, which is not appropriate for a normal operation.

Figure 8 shows the distribution of heat and cool in operating the Peltier element, which was already described in section 3.2. When the current is flowing, heat is absorbed on the cold side of Peltier module and transferred to the hot side. Consequently, it is necessary to provide heat dissipation on the hot side to avoid overheating. It can be done with a heat sink. For even more effective dissipation, the fan must be attached to the heat sink. It will generate air flow, and consequently more intensive heat transfer. Figure 9 represents one possible solution of this system. [2]



Figures 8/ 9. Peltier element structure and operation/ Assembly with Peltier element. Copied from Physics 212 Web Project [2].

Peltier elements can be used in a backward principle-Seebeck effect, which was considered in section 2.2. If the temperature difference is applied to both sides of the element (cold and hot simultaneously), the voltage difference is built up between two sides, and the electric current starts to flow. The voltage can be measured by an ordinary multimeter attached to the Peltier element wires. In this case, Peltier element plays the role of a thermoelectric generator. However, the efficiency of this backward usage is very low as the generated power is not higher than a few milliWatts.

3.4 Characteristics of Peltier element

After all the constituents of the Peltier element structure and operation have been considered, it is necessary to define the main characteristics of the device. Those are useful when choosing a thermoelectric module and designing the system based on it.

The Peltier device operation is defined by several important characteristics such as the maximum thermal load (Q_{max}, the amount of thermal load at maximum current, in Watts); the maximum temperature difference (ΔT_{max}, at maximum current, in ºC/K); the maximum voltage which causes maximum temperature difference at maximum current (U_{max}, in Volts); and the maximum direct current (I_{max}, in Ampers) at which the highest ΔT is achieved.

The temperature difference depends on voltage, and consequently on power applied to the module. Voltage is not a standard value and varies from module to module depending on the manufacturer and target application. For a comparison between different Peltier elements, temperature-voltage and temperature-power curves are normally provided in datasheets. One example of voltage VS temperature curves is shown if Figure 10.

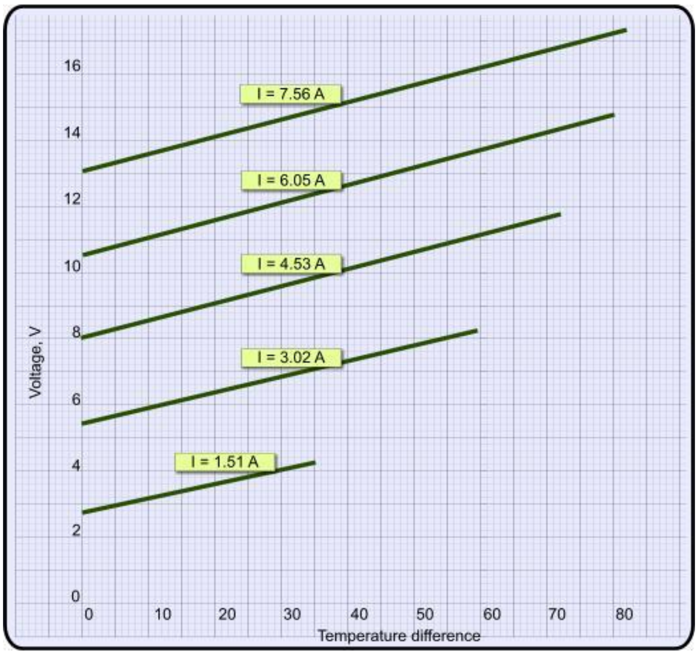


Figure 10. Example of Peltier element curve representing dependence between temperature difference (ΔT) and applied voltage (V).

As figure 10 shows, some unspecified Peltier element provides a different temperature difference from 0ºC to 83ºC at the voltage range of 2.3V-17.6V. For example, if the supplied voltage to this element is 2.3V or 13V, the maximum temperature difference is 35ºC or 82ºC respectively. Comparing such characteristics of different Peltier modules, the right selection of the device will be provided correctly.

Another important comparison value is the coefficient of performance (COP) which defines the efficiency of a Peltier device. COP is a ratio found as the division of heat absorbed on the cold side and the consumed power (Equation 3).

$$COP = Q_{Cold} / P_{Peltier} \ [W / W] \qquad (3)$$

COP defines Peltier device efficiency and allows comparing different modules. The maximum COP means that the minimum input power is supplied, and consequently the minimum heat has to be rejected by the heat sink. However, to select a heat sink properly, one more characteristic must be considered.

Thermal resistance is the rate of the amount of the heat transferred from the device to the air. In the case when heat sink is attached to the Peltier element, this amount of heat is transferred by the heat sink. Equation 4 shows how thermal resistance is calculated.

$$R = \frac{Tcase\_max - Tin\_max}{Pdissipation} \left[\frac{{}^0C}{W}\right] \qquad (4)$$

Tcase_max [ºC] is the maximum temperature, which can be achieved without damaging the Peltier element. *Tin_max [ºC]* is the highest inlet temperature achieved. *Pdissipation [W]* defines the amount of power to be dissipated from the device.

Usually, datasheets of Peltier modules' provide thermal resistance of the element, so it is not necessary to do it manually. According to that given value, the correct heat sink can be selected. Thermal resistance of the heat sink should be equal to or smaller than that of the Peltier module. Smaller thermal resistance gives better cooling.

3.5 Thermoelectric cooling superiority over traditional refrigeration

Nowadays, most refrigerating systems are based on compressors and refrigerants. However, designs based on the thermoelectric modules are becoming a reasonable alternative for those systems, in special circumstances. There are several considerations for this replacement, which explain why Peltier elements were used in this project, instead of any traditional refrigerating system. [10]

Considering the maintenance and safety of the systems based on thermoelectric modules, some advantages have to be mentioned:

- No moving parts
- Integrated chip design
- No human hazardous emissions (while possible from refrigerants)
- Low noise operation
- Compact size (for example, no need for huge compressors). [10]

All in all, thermoelectric modules are useful to replace refrigerators when small areas (local points) have to be cooled, and the minimum managing work is required.

There are also some advantages when considering operation:

- Reliability (over 100000 operation hours)
- High accuracy (+/- 0.1 degree Celsius)
- Easy switching mechanism between cooling and heating (current reverse)
- Low DC voltage requirements
- Immediate response in change of temperature by adjusting current. [10]

Considering all the mentioned points, the thermoelectric technology can be used in applications where precise and reliable cooling is needed. Thermoelectric modules are simply maintaining, which is why they are useful in very scrupulous processes. Due to these properties, Peltier modules were chosen as the main technology in this project.

4 Existing Peltier element-based solutions

Before starting to design a custom system, all available ready-manufactured solutions were considered. This industry and market research provided an idea of how experienced manufacturers overcome Peltier technology disadvantages, and implement its strengths. Moreover, existing solutions gave a vision of a design which is the most optimal and efficient.

4.1 Areas of implementation

Nowadays, Peltier modules are used in both consumer and industrial areas. They are implemented when heat removing requirements are from milliwatts to several tens watts. As mentioned earlier in section 3.3, it is not possible to use a pure Peltier element without modifications and installation of additional parts. Moreover, each application requires precise consideration of supplementary details depending on the area of use and approach.

In the consumer area, Peltier modules are widely used in portable cooling devices as well as for cooling small electronics such as processors in a PC. Portable cooling devices are represented by refrigerators utilized in camping and other outdoor conditions. Commonly, those applications do not require big heat dissipation, and consequently massive additional parts. That is why Peltier modules are successfully implemented here.

The biggest application of Peltier modules in science is cooling of devices used for laboratory experiments and examinations such as DNA and cell research. Usually those devices are tiny, which is why a Peltier element can be freely used to cool down the whole working area. Another scope is lying in telescopes, spectrometers and digital cameras in astronomy where cooling of photon detectors is needed. It provides noise reduction, and consequently allows to gain clearer results without extra dark counts.

For example, DNA research is done with special engines designed specifically for such examination. These machines provide precise constant temperature of the surface where material is placed during the experiment. For this purpose, Peltier modules are implemented. This configuration makes research easy controlled and robust. Such machines are widely used in scientific and medical certified centers, and are very reliable.

4.2 Assemblies based on Peltier elements

Not only pure Peltier modules are integrated into industrial and consumer applications. Many assemblies targeted for custom implementation are developed nowadays. Several configurations of such assemblies are openly available on the market, and allows meeting all possible requirements and configurations in a customer's own design.

As was previously mentioned in section 3.2, heat dissipation is required to achieve good operation of a Peltier module, and to avoid overheating. There are two approaches to do that with heat sink installation. The first one is to use a metal heat sink with a fan attached to create an airflow which blows away warm air. The second implements water-cooled heat sinks. Here a water radiator is attached to the heat sink. The water flow plays the same role as airflow in the previous method.

Nowadays several manufacturers in the world produce assemblies based on Peltier modules. They offer all the types of those constructions: air-to-air, air-to liquid and air-to-contact devices. These types were needed to be considered to decide which configuration would be best for this project application.

4.2.1 Contact-to-air

Contact-to-air coolers, or contact coolers, which example is presented in figure11, are the best method to cool down the object, primary with a flat surface to achieve the ideal contact. In this case, the operation is maximally efficient. Usually, if there is no possibility to provide a flat side, different adapters might be used. However, it might slightly decrease the efficiency of cooling. [13]

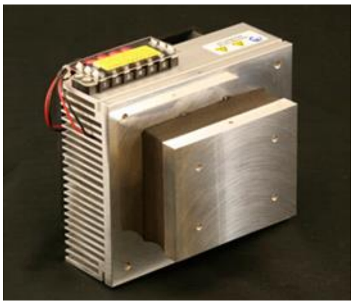


Figure 11. Contact-to-air cooler. Copied from TE technology Inc website [13].

The normal capacity range of contact-to-air coolers is 30-200 watts at 0 degrees temperature difference depending on the size of the Peltier module and some other factors. [13]

4.2.2 Liquid-to-air

Liquid-to-air coolers, or liquid coolers, which example is presented in figure12, have more specific application than contact coolers. It is useful when small remote objects need to be cooled and contact coolers cannot be installed due to unsufficient location. In this case cold liquid might be delivered to the object by one tube, and returned by the other tube. There are some crucial requirements which have to be followed for proper operation. The required amount of water flow is a minimum of 1.6L/minute. Also a pump is needed to arrange a water flow. [14]

Figure 12. Liquid-to-air cooler. Copied from TE technology Inc website [14].

Power consumption of liquid-to-air coolers is 35-200 watts at a 0 degree temperature difference, which is approximately the same as for contact coolers. [14]

4.2.3 Air-to-air

Air-to-air heat exchangers, or air coolers, are extremely useful when objects inside the fridge cabinets or racks are needed to be cooled, and there is no possibility to allow direct contact with contact-to-air modules. Air coolers always contain a cold sink which is the place where all the condensation is formed. This allows avoiding condensation formation on the cooled objects. That type of cooler device is the most useful one for cooling the electronics located inside enclosures. [15]

Figure 13. Air-to-air cooler. Copied from TE technology Inc website [15].

Power consumption of air-to-air coolers is 24-200 watts at a 0 degree temperature difference. [15]

For all the above described types of cooling assemblies, the operation voltages are 12VDC, 24VDC or 48VDC, depending on the power consumption. Fans are normally powered up by 12VDC, but this value might fluctuate due to the manufacturer’s requirements. Another useful addition to those systems is a controller. Several types of controllers are available on the market now, which allows managing the system temperature, fan rotation, the rate of moisture and other more complicated options.

However, an obvious disadvantage for these three types of assemblies is their price. Rarely does a purchaser need only one item, but several or even dozens of such modules have to be bought. That purchase is very expensive, and small developers cannot afford it. That is why custom-made solutions are often developed to reduce costs, as was done in this project.

5 HVAC systems

5.1 Areas of HVAC systems

Nowadays, HVAC systems are widely used in houses and cars to allow better conditions for a human being. Those systems provide flexible environmental customization according to personal and scope-oriented needs. The development of HVAC systems is advancing fast, offering more efficient, productive solutions to achieve not only human needs, but also environmentally friendly usage.

Although it is correct that the main places of HVAC systems implementation are everyday life human oriented objects, there are varieties of other specialized applications. For example, space crafts and stations require having an HVAC system integrated to allow proper operation of the equipment in conditions of zero gravity. Another area of application might be a submarine board. Acting under the giant water pressure, HVAC implementation is necessary to provide fresh air exchange and CO2 dissipation. One further area of HVAC systems installation is data centres which contain tons of equipment which have to be protected from excessive humidity and overheating.

This project does not cover each area of HVAC systems, but is mostly concentrated on the cooling part in air conditioning. Humidity and ventilation are out of the scope of this project. Human comfort is the main concern for development.

5.2 Control of HVAC systems

All the areas where HVAC systems are required have one common factor. Their environmental conditions have to be easily controlled and maintained. It is crucial to provide correct and time-strict operation of HVAC systems. For that reason, strong and reliable control systems have to be implemented.

Different ways of controlling were developed with modernisation of systems themselves. The most common type of control is sensor-detection. Usually, there is a need to maintain the operation of an HVAC system according to a temperature or humidity values. The most efficient way is to make measurements by sensors and to sending the data to control units. Control units then act according to a prescribed plan.

Several devices are targeted to maintaining the environmental situation. Thermostats are oriented to change the temperature if sensor data is not as required. Special humidity controllers regulate humidity consequently. However, today many solutions are present to maintain several conditions by one device. Moreover, those special devices might provide an energy-saving mode, usage of recycled energy or efficient energy distribution.

6 Control system

To create a useful and effective thermoelectric system based on a Peltier module, it is necessary to be able to control it. Basically, such a system should contain a sensor (most commonly is a temperature sensor), controller circuit such as microprocessor, transistor, comparator, relay etc. which includes a programmed code or logic, and a connection to a PC with software to collect and monitor the gathered data and manage the operation. Figure 14 presents an example of the described system.

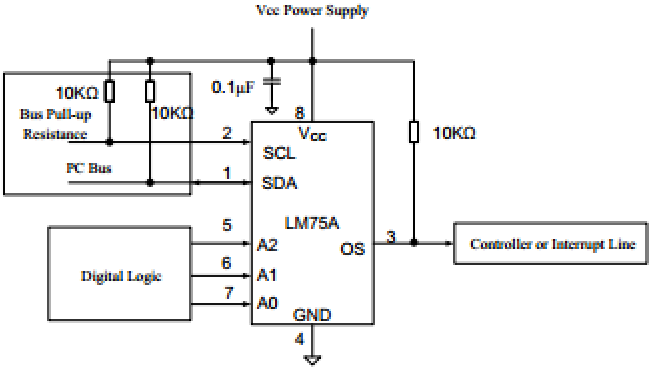


Figure 14. Typical connection of LM75 temperature sensor. Copied from Xuejun Yue [21, 246].

The temperature sensor for the control system is usually selected according to the required range of measured values and accuracy. To make the control system efficient, the measured temperature should be processed appropriately. An analog-to-digital converter allows converting the obtained signal into digital format, which can then be handled by a microcontroller.

Figure 14 shows the diagram implemented by Xuejun Yue [21, 246] who created a temperature sensing system. In that work, LM75 sensor was implemented. The measured temperature was observed and monitored, and an I2C interface was added to control access to the temperature sensor.

With researches in Peltier technology implementation, various ways of control were developed. Each of them, however, has advantages and disadvantages. That is why it is necessary to consider all factors and specifications of a specific system, before creating its control module. Moreover, many possibilities were developed which allowed simulation of the control system before making the actual installation.

6.1 Control system simulations

Although no simulations were done in this project, it was valuable to cover such experience gained in different researches. That study helped to make a list of important points which required consideration when proceeding to my own system design and debugging.

According to Yaqing Qil et al. [3] one way of simulating the operation control of the system is implementing SCM (software configuration management). This type of simulated system includes revision control, as well as baselines establishment. They are necessary to adopt temperature precision which is possible with a phase-shift. When the system is fully configured, it is possible to react to changes precisely, and act in accordance with the actions prescribed.

Song Shaojing et al. [4] offer to use NN-PID (Neutral Network Proportional-Integral-Derivative) algorithm, which is a control loop feedback mechanism. It includes three basic values (three-term control): the proportional (P), the integral (I), and derivative (D) parameters. The major mechanism of this type of controller is to calculate the error, which is the difference between the actual measured value and the desired set value. In their work, Song Shaojing et al. adopted their controller model to calculate the control signal and then adjust the current of thermoelectric module [4].

Jin Wang et al. [5] proposed the use of MatLab/Simulink environment to develop and simulate control system for thermoelectric cooling. MatLab itself proved the most popular and reliable software for the purpose of modeling and simulation. Jin Wang et al. were interested in power losses acquired during the cooling process, so their system measured this value and compared it to a fixed threshold. According to that, TEC current was calculated and power was given to drive TEC module. Constant control is achieved with PI and PWM generator use, which adjusted current and power permanently.

All in all, study of those simulation techniques provided several crucial aspects to consider. It is important to note temperature accuracy, and speed of its changes, to design an effective system. Also, different methods of current calculations should be considered to define the best one for precise operation. Moreover, power losses have to be observed because they affect the power applied to the system.

6.2 Real control technology

Despite many simulation techniques, several ways of physical implementation of control systems were developed as well. They are mainly based on a microcontroller, sensors and a PC to provide control of the electronics. This basic structure allows adding extra parts to make the system more user-oriented and easily controlled.

6.2.1 LabView

LabView proved itself to be robust and effective software for cooling process control. Nahdini K.K. et al. [6] created a system, which adjusted power supply according to the temperature. A temperature sensor then collects data processed by a comparator, to compare set voltage and sensor voltage. Transistors conduct only voltages, which are higher than the set voltage. By this way, the constant temperature is maintained by the system. Temperature measurements are also recorded by a USB acquisition device connected to LabView software, which is programmed to adjust the cooler supplied voltage according to the recorded temperature.

6.2.2 Arduino single-board microcontroller family

Besides all the described and developed ways, for my work I found another way to control the cooling system. The Arduino system is very efficient when creating custom control and measurement systems. It proposed a variety of solutions for different kind of applications. Several developers have been working on special shields and modules to meet every consumer’s requirements.

Today, Arduino provides solutions from wireless transmission and high current control, to wearable electronics for every-day usage. It supports such operating systems as Windows, MAC OS and Linux. Arduino implements its own libraries; however it is compatible with Java environments and C/C++ microcontroller libraries. Its own software, as well as, hardware are available open source.

Arduino shields provide additional opportunities and features while creating custom hardware systems. There are eight officially certified shields currently available.

Moreover, dozens of unofficial custom-made shields are also floating on the market. As with every device, it is more reliable to use officially confirmed products to avoid minor mismatches and drawbacks. Also, manufacturers of certified products can provide support, which might be unavailable when using custom products.

7 Design and implementation of cooling system

While accomplishing the thesis project, I have been employed in a company whose area of operation is an innovative flooring system. My job was integration of mechanical and electronic systems into the floor providing a more comfortable and modern working environment. However, not only electronics was the part of interest, but also cooling and air conditioning was planned to be implemented.

Before start of that project, I received very limited instructions without definite final solution and procedure of accomplishing, so extensive research and development activity was executed. Study of theoretical part was already described in chapters 1-6, and practical implementation will be covered further.

7.1 Problem statement and concept

Before starting any practical work and considering the ways of its implementation, the problem has to be clearly stated. According to the problem, a technology which fulfil the necessary results can be considered. Therefore, I first established a concrete target to be achieved.

7.1.1 Problem description

The area of my personal research was restricted by flooring system limitations as well as by the main implementation technology. I received strict instructions to use Peltier technology to implement the idea. Overall, I was fully responsible to handle the whole project's process from investigation until implementation. To manage with a problem I had to work out a list of subjects for consideration, which is obviously necessary for any project.

- What is the best way of working with Peltier technology?
- What are possible market available solutions?
- Which advantages and disadvantages do those solutions have?
- How a custom solution could be implemented and integrated into a floor system?

According to those questions, the actual plan of managing the project was done.

1. Technology study
2. Market study
3. Physical evaluation of the technology
4. Physical implementation and integration into a floor system
5. Control system design and installation.

The main challenge of integration of a cooling air conditioning system into the floor was that the space under the floor was very limited, which restricted placement of massive modules and parts. Moreover, significant power sources were expected. Therefore, the aim was to work out a solution which would be maximally energy efficient and compact in size.

7.1.2 Concept description

Despite the stated working technology, concept figures were available when announcing the project prospective (Figure 15). The final product design had to be implemented according to those images.

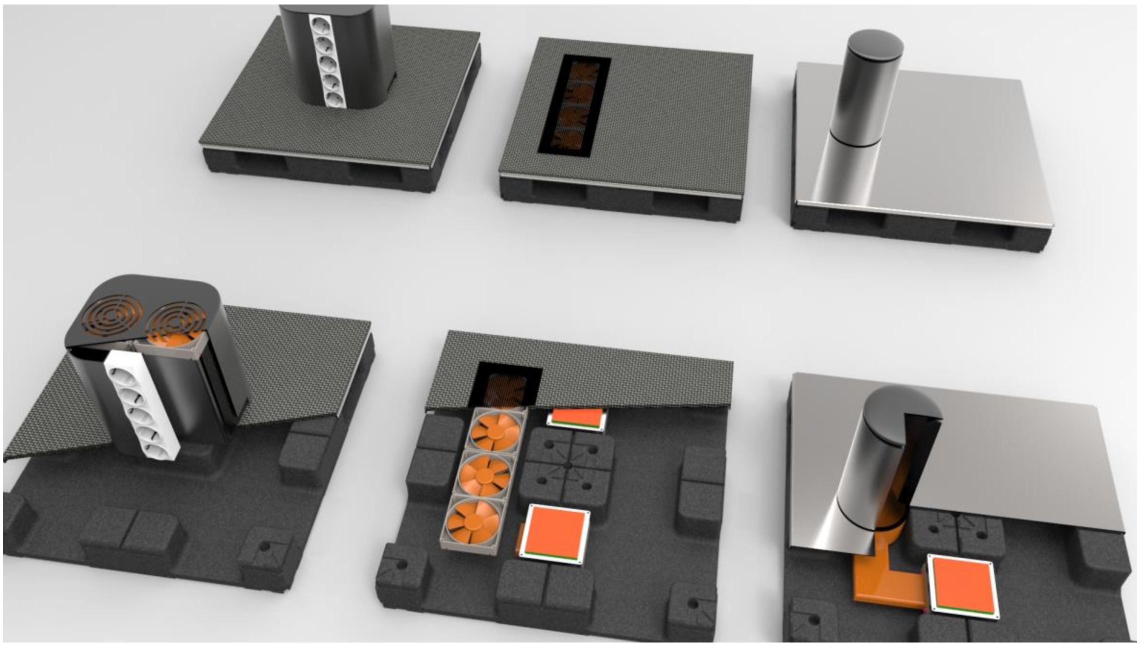


Figure 15. Concept pictures of the resulting system. Floor plates with additional modules installed.

Obviously, necessary improvements were essential because the concept pictures were done by a person who was not familiar with the Peltier technology and its characteristics. However, the idea of placing an extra part above the floor with the cooling system inside, and partly under the floor, had to be followed.

7.2 Physical evaluation of the technology

7.2.1 Manufacturer selection

First, before ordering Peltier modules, a comparison of the present manufacturers was done. Three different companies were selected. They were CUI Inc., Laird Technologies and Multicomp, which are the biggest producers of Peltier elements nowadays. Three models of Peltier elements of each manufacturer were chosen, and their datasheets were revised. To simplify the comparison the table based on the datasheets was composed (Table 1).

Table 1. Comparison between different manufacturers according to main characteristics based on datasheet information.

Manufacturer name	Module name	Iinput, A	V (ΔT=30C), V	Q (ΔT=30C), W	Price, euro
CUI Inc	CP85438	8.5	2	5	15.41
CUI Inc	CP30338	2.4	3	3.1	13.24
CUI Inc	CP20151	1.2	2.3	1.4	11.96
Laird Technologies	7950002-601	8.5	13	45	42.33
Laird Technologies	430027-501	2.4	3.8	6	25.31
Laird Technologies	430511-504	1.2	1.2	1	27.65
MULTICOMP	MCPE1-12708AC-S				58.76
MULTICOMP	MCPE-071-14-16	2.4	4	9	38.57
MULTICOMP	MCPF-031-10-25	1.2	2.1	1.7	33.82

Table 1 contains nine different Peltier elements, three from each manufacturer. The elements differ in current, voltage, obtained power and price. To analyze this data easily, the gathered information was converted into a graph (Figure 16).

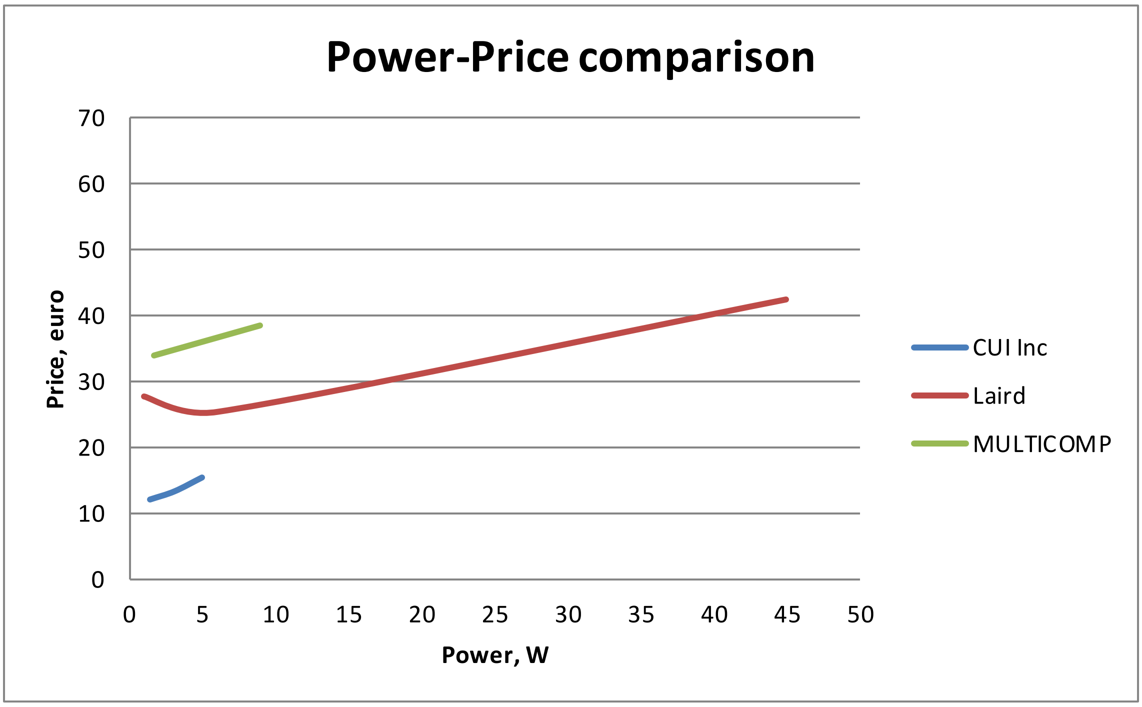


Figure 16. Comparison between different manufacturers according to power and price formed from Table 1.

As clearly seen from the graph in Figure 16, a comparison in power-price axes was considered. Laird products gave the highest power with a highest price accordingly. Overall, the dependence between power and price was proportional for all manufacturers. However, CUI Inc gaves the most reasonable rate of price and power, but in lower power range.

Although the result of the comparison showed that CUI Inc. was the best choice, I had to stay on the Laird Technology option, because it provided more power, which meant that the temperature difference was also higher. Moreover, this manufacturer provided a wider range of products in Europe.

7.2.2 Peltier elements order

After choosing the manufacturer, Peltier elements were ordered. Three modules (from the Laird Technologies manufacturer) with different characteristics were investigated. Table 2 represents the characteristics of these modules.

Table 2. Ordered Peltier modules and their characteristics.

	Model	Size	Voltage, Vmax	Current, Amax	Power, Wmax
Peltier element 1	CP2,127,06	62x62	16,4	14,3	134,4
Peltier element 2	SH10,95,06	30x30	10,6	3	19,7
Peltier element 3	UT15,12,F2,4040	40x40	14,4	14,6	126

When observing Table 2, Peltier element 2 is the smallest and provides significantly lower power than two other modules. Elements 1 and 3 are of different sizes but provide approximately the same power, and require almost the same voltage and current for operation. Those characteristics are important to note when considering the efficiency of the device defined by COP, as was mentioned in section 3.4.

As was noted earlier in theoretical part in sections 3.1-3.3, a pure Peltier element cannot work properly. It is necessary to provide heat dissipation from the hot side of the module. For this purpose, heat sinks and fans were purchased. The heat sink was selected according to the Peltier elements thermal resistance stated in the datasheets. For all modules, this value was around 1.1°C/W, which allows using a heat sink with the same thermal resistance or with the smaller one. Table 3 shows the selection of those parts.

Table 3. Additional parts list for Peltier modules evaluation.

Reference	Description/Size
Heat sink	250x101.6x32, 0.05°C/W
Heat sink	250x100x40, 1°C/W
Fan	DC Fan 92x32mm, 12VDC
Fan	DC Fan 172x147x25mm, 24VDC

Two options of both heat sinks and fans were bought to compare them and their effectiveness. Heat sinks were same size but with dramatically different thermal resistance values. Fans were of different supply voltage ranges.

7.2.3 Operation assessment

Before starting prototyping, the initial step was to realize what the limits of thermoelectric modules are, and what the conditions are which affects the operation. For this purpose, simple tests were implemented.

A coin was fixed on a heat sink located on the cold side of the Peltier module. The hot side was placed on a heat sink to dissipate the heat. Heat paste was applied between heat sinks and Peltier sides for a better thermal contact. Datasheets were carefully studied to provide the correct power. The assembly is shown in Figure 17.

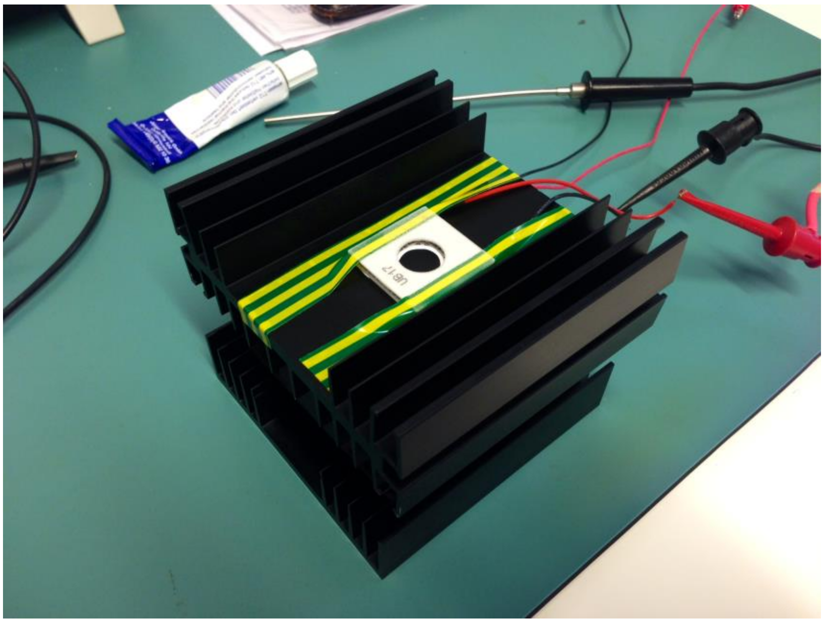


Figure 17. Initial test assembly.

After the power was ON, the temperature was measured on the cold side. The minimum value achieved was -8^{0}C. Due to such a low temperature, frost was formed on the coin. Figures 18 and 19 represents states of the coin before and during the test.

Figures 18/19. Coin state before and during the test.

After switching off the power, frost disappeared immediately, as the cold side heated up from the hot side in several seconds. The test showed that implementation of a Peltier element without cool dissipation might cause a short circuit in some cases due to water formation. Consequently, to arrange a safety operation, something should be placed on the cold side of the element to increase the cold area. The simplest option is another heat sink.

All three ordered TE modules were tested in the same configuration with a coin. However, one module (Peltier element 3 in Table 2) showed the best efficiency as it provided the biggest temperature difference between the cold and the hot sides. This practical outcome was proven by theoretical information provided in the datasheet (Figure 20).

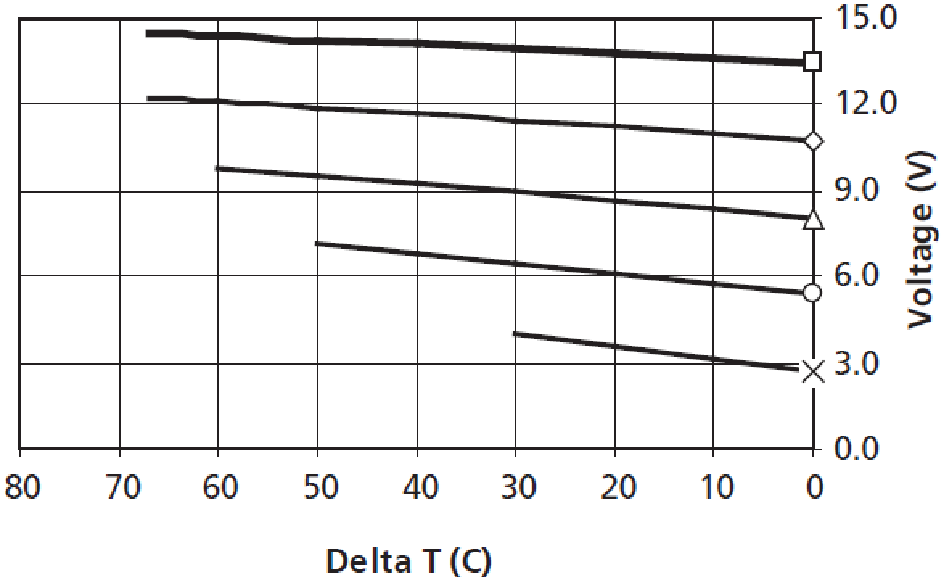


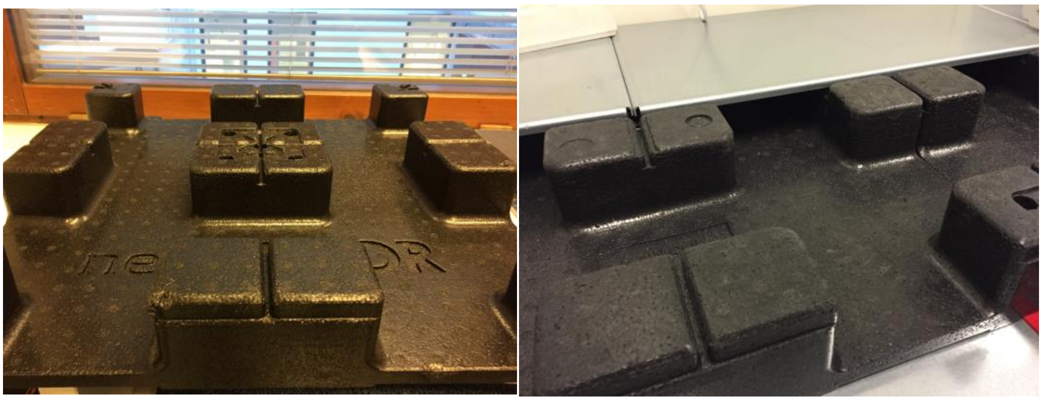
Figure 20. Temperature-voltage curve of the selected Peltier element. Copied from Laird Technologies website [16].

Figure 20 showed that in the range of 11V-14V, the maximum temperature difference of almost 70ºC might be achieved. For two other Peltier elements 1 and 2, this value was not as effective. That is why this Peltier module with model number CP2,127,06 was chosen as a working part for the future prototype.

7.3 Real configuration design

The first concept of Peltier based system implementation was to integrate in into the Nextfloor flooring system. The precise aim was cool airflow generation above the floor, resulting in comfort working conditions. Different prototype configurations of that concept were evaluated. However only one of those appeared efficient and applicable.

In practice, the prototype model had to be integrated into the floor system, which consisted of light weight blocks of a height of 6cm, and a metal plate placed on top of each block (Figures 21 and 22).



Figures 21/22. Floor base element as the location of Peltier system placement.

Several problems appeared in that configuration.

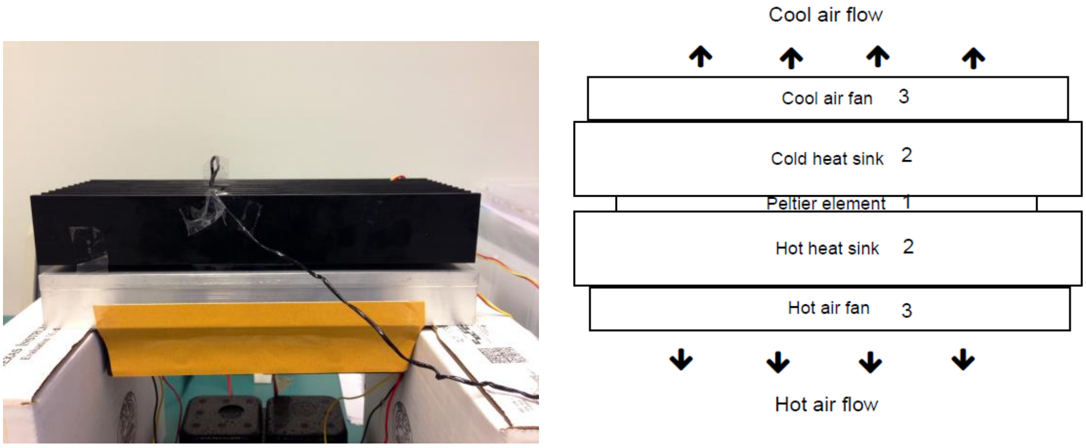
1. Air supply for fans
2. The size of the heat sinks
3. Heat dissipation from the hot side of the Peltier module and heat sink

All these problems were solved successfully and will be clarified in further chapters 7.3.1-7.3.3. Nextfloor flooring base element was used as a space for the hot side fan air supply as well as for heat dissipation. Then the hot air will be flowing under the floor, while the cold air should be above.

7.3.1 First model of the assembly

The application requires formation of cool airflow over the metal plates. It was decided that the most efficient way to do that was to attach a heat sink to the cold side of the Peltier module, and use a fan to blow the cold air from it.

The hot side has to be cooled as well to avoid overheating of the TE element. A heat sink of the same size as for the cold side, was placed to the hot side. Figure 23 presents the assembly without its integration to a floor system, Figure 24 shows the system's diagram.



Figures 23/24. Assembly before integration to a floor system. No fan yet physically attached to the cold side heat sink/System diagram

Figure 24 shows just a model of the system. It contains only one Peltier element, 2 heat sinks attached to both sides, as well as two fans to blow away cold and hot air. However figure 23 represents a real system. A grey heat sink is attached to the hot side of the Peltier module. An orange tape is holding two fans installed to the hot heat sink to provide heat dissipation. A black heat sink is placed on the cold side of the Peltier module. One more fan for the cold airflow has to be added.

To evaluate the efficiency of the airflow channel, a tube from a carton was made (Figure 25).

Figure 25. Carton tube over the cold heat sink with the fan creating airflow.

A case when the fan is attached to the cold heat sink, the carton tube helps to direct the flow without its dissipation around. This first draft was not obviously efficient, because it was not sealed, and heat from the hot side was mixing with the cold airflow. Nevertheless, it helped to evaluate that this type of system might work if properly constructed.

7.3.2 Alternative model

To prove the best efficiency of the designed system for a comparison purpose, another configuration was considered as well (Figure 26).

Figure 26. Vertical configuration of the prototype draft.

In this configuration, presented in Figure 26, cold airflow is directed vertically, not horizontally as previously. However, this model appeared to be inefficient because in that way the fan does not get enough supply air. As a result, configuration was rejected.

7.3.3 Additional points for consideration before installation

Additional details were considered. One Peltier module seemed to be insufficient to achieve the necessary coolness of a heat sink, so it was decided to use two similar Peltier elements. In addition, it was obvious that some dissipation channels had to be installed under the floor to blow away warm air more effectively and far from the fans' supply. However, that had to be considered after the installation was completed.

For a proper operation, fans require enough distance between the blade and surface to provide enough air supply for rotation. Due to the small distance between the floor basement and metal plates, hot side fans were installed under the metal plate leaving Peltier modules with a hot heat sink and other parts on top. In that configuration, fans have enough supply air under the floor. The supplied air goes through the hot heat sink blowing the hot air away under the floor as well.

Another important point is how the upper fan affects the temperature of the heat sink. Experiments showed that during some time of fan operation, the cold heat sink became warmer. Consequently, the system needed to recover for some time without the upper blowing fan. This was necessary to be taken into account later when implementing a control system.

After these evaluation tests were completed, an external cover and enclosure of the prototype were planned and designed.

7.4 Physical prototype of main system

7.4.1 System assembly

During the designing process, it was decided to install the hot side fans under the floor metal plate, while the hot side heat sink and the whole cold side had to be placed over the floor. To accomplish that, the holes in the metal plate were cut (Figure 27). The big hole was used for bottom fans. Two small holes were for hot air dissipation.

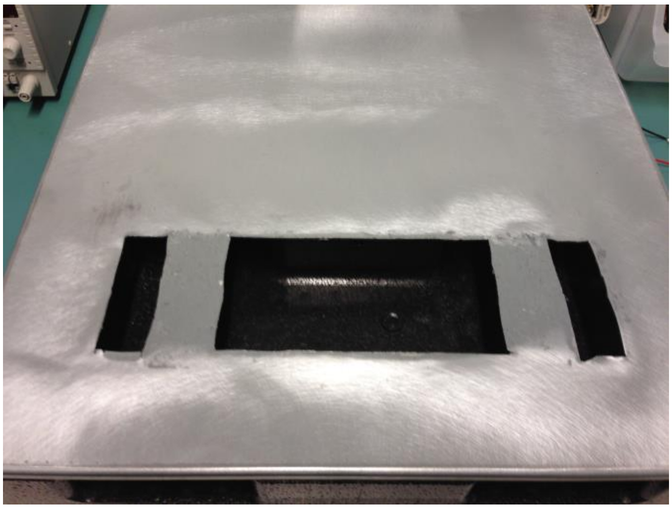
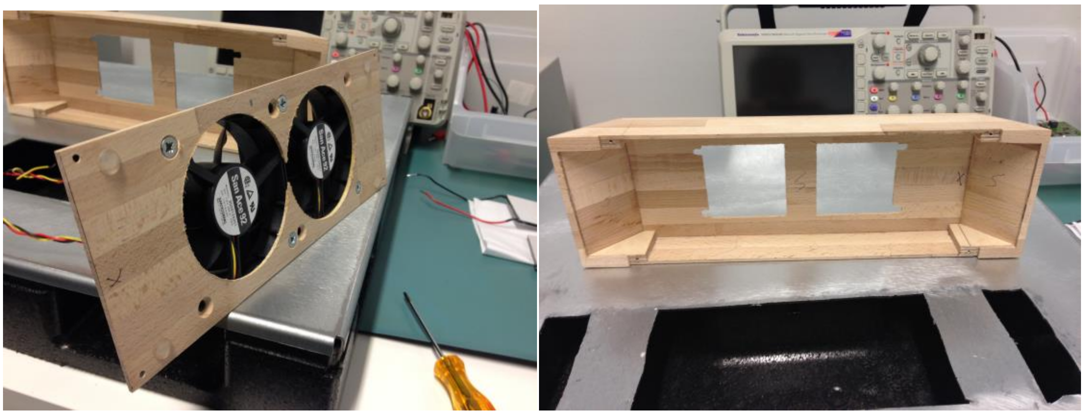


Figure 27. Holes in the metal place.

The wooden parts were constructed to fix the fans, as well as to place the bottom heat sink inside (Figures 28 and 29).



Figures 28/29. Wooden assemblies for bottom fans and bottom heat sink.

Assembly for the bottom heat sink has two holes, which are intended for the Peltier modules. They are exactly of the same sizes in order to install modules reliably without spaces. Fans are then placed through the holes under the metal plate, and are fixed by the wooden platform shown in Figure 28.

Figure 30 represents the whole assembly, which covers the hot side of the system. Peltier modules are connected with the heat paste to the heat sink to provide firm contact.

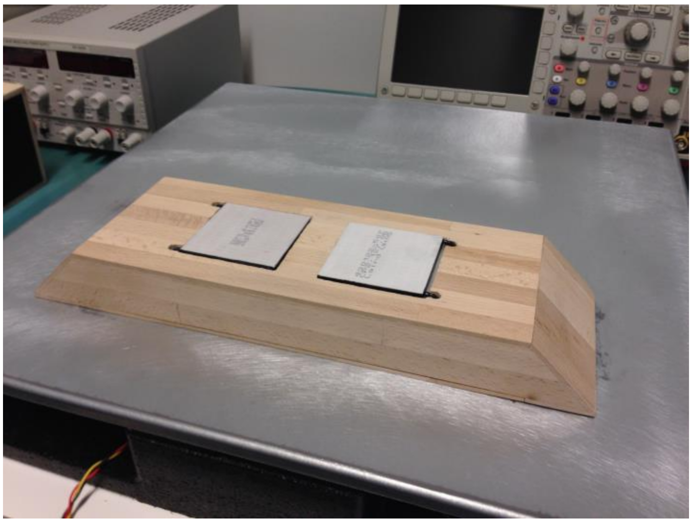
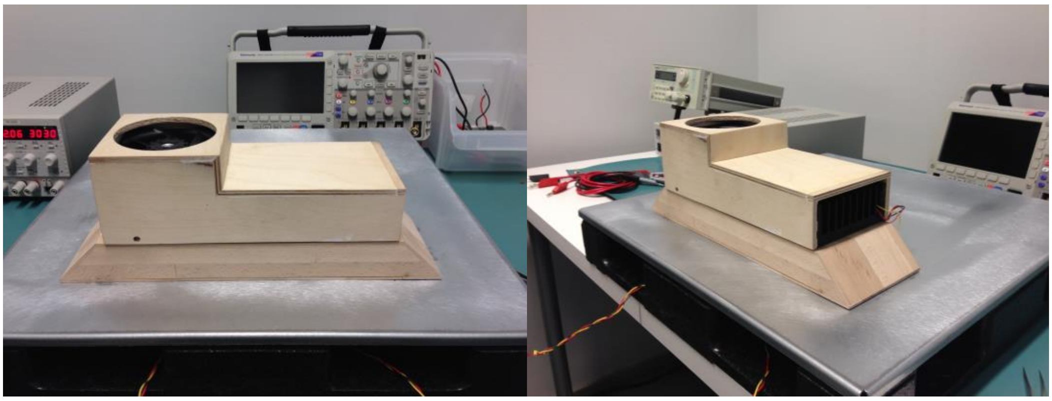


Figure 30. Assembly for the hot side of the system.

The next part of the physical prototype was the system’s cold side. This part contains another heat sink and one fan to produce cool airflow. The same wooden material was used to cover these parts. The assembled cold side system is then placed on Peltier elements' cold side with a heat paste applied between them. The whole assembly is shown in Figures 31/32.



Figures 31/32. The whole assembly with the upper cold side installed.

At that point, the prototype creation was completed and I could proceed to control system planning and implementation.

7.4.2 Additional parts and details

The assembled system was evaluated when installation was completed. Visual and tactile tests helped to understand disadvantages, which were not considered before. Some of those could be improved immediately by additional planning, while others had to be left as such in that configuration.

First of all, circulation of hot air under the floor was not effective. The bottom fans blew the hot air through the hot heat sink, and forwarded it via two small apertures into the metal plate under the floor. It became a problem because fans supplied same hot air and, consequently, could not cool down hot heat sink effectively. To resolve that, tubes were attached to hot air holes, which directed hot air far away from bottom fans supply point. Figure 33 shows the system with tubes added. In that configuration, hot air dissipates under the floor more effectively.

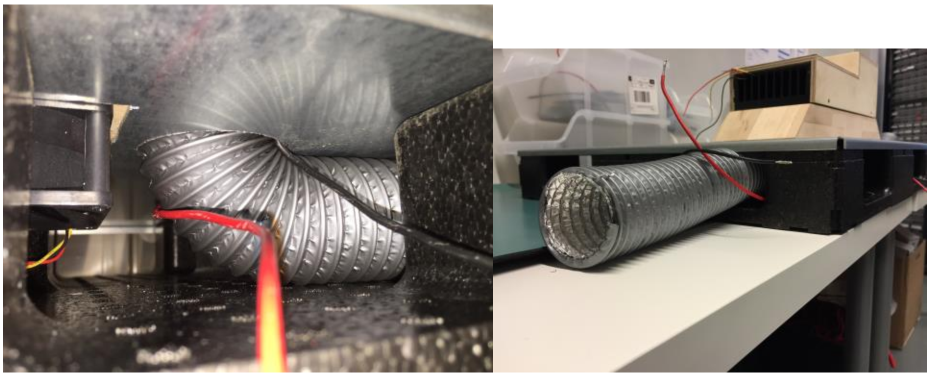


Figure 33. System with tube for heat dissipation under the floor.

One more important detail, which is not yet implemented, is power supply. At the testing stage, a stationary DC power supply was used. However, it cannot be used in real life. To make the system portable and compact, a plug-in AC-DC power supply should be implemented. In that configuration, only a wall socket would be required to power up the system.

8 Design and implementation of the control system

After the physical prototype was completed and installed into the floor, a control system had to be designed. The purpose of the control system is to control power coming to Peltier elements and fans according to temperature changes. According to these requirements, parts of the system can be selected.

8.1 Selection of Arduino shield and parts

As was mentioned in theoretical part in section 6, Arduino became a working technology for the system because it provides a wide range of various shields useful in different applications. Before deciding the most appropriate, the parts required for my project were considered. Power to Peltier modules and fans has to be switched on and off. It can be effectively implemented by relays. Temperature might be measured by temperature sensors. Such additional parts as buttons and displays are required to provide communication with a user.

According to the parts’ requirements, several Arduino shields available on the market were considered. However, the Grove model provides the highest power range for relay, which was crucial for this project as high power is used. That is why the Arduino Grove shield was chosen. Arduino Uno board was used as the main controller. Moreover, relays and temperature sensors, as well as buttons and display, compatible with a Grove shield were purchased. Precise models of these modules are described in Table 4.

Before proceeding to code writing, the system was completely installed. All in all, six digital ports, three analog ports and one I2C port were used to connect system components. Table 4 represents the connections table.

Table 4. System connections and module types.

Port	Module	Type of module
I2C	OLED display	Grove - OLED Display 1.12"
D6	Relay 6 (Peltier elements)	Grove - SPDT Relay(30A)
D7	Relay 7 (Upper fan, cool flow)	Grove - SPDT Relay(30A)
D8	Relay 8 (Bottom fans, hot flow)	Grove - SPDT Relay(30A)
D2	Button 2 (Peltier elements)	Grove - Button
D3	Button 3 (Upper fan, cool flow)	Grove - Button
D4	Button 4 (Bottom fans, hot flow)	Grove - Button
A0	Temperature sensor 1 (cool air flow)	Grove - Temperature Sensor
A1	Temperature sensor 2 (room temperature)	Grove - Temperature and Humidity Sensor Pro

Table 4 presents ports numbers and corresponding module connected to that port. It also contains types of modules.

Correct connection to the necessary ports is very important because it guarantees proper operation of the further program. Figure 34 shows the Grove shield with used ports marked on it. Figure 35 represents the real picture of the system with all parts connected.

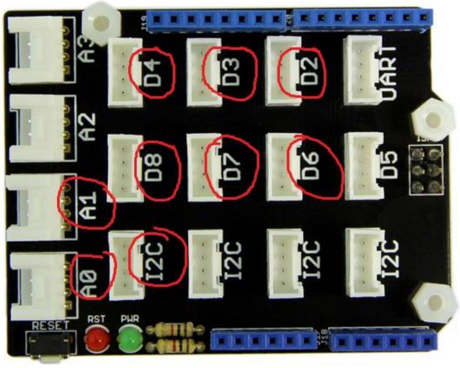


Figure 34. Grove shield. Copied from http://www.seeedstudio.com/wiki/Grove_-_Base_Shield_V1.3

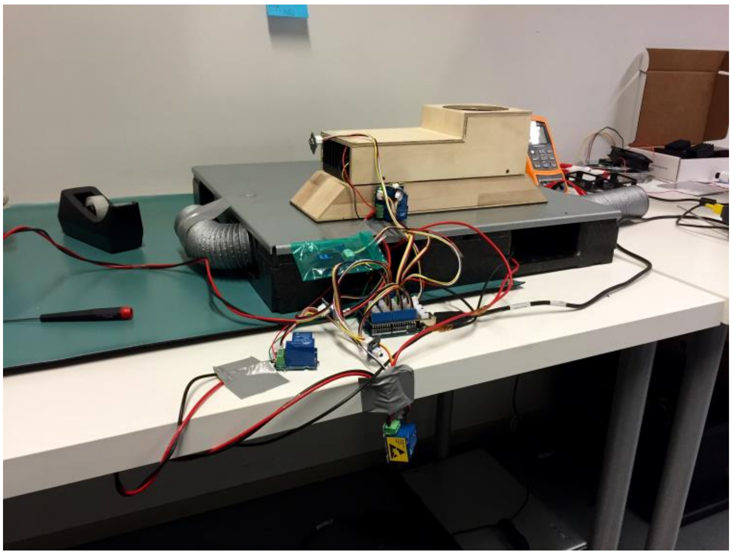


Figure 35. Arduino control system connected. No enclosure.

After the control system was connected separately, it was attached to the physical cooling system. The final installation will be described in chapter 8.2.

8.2 Control system design

Before proceeding to code writing, the control system was connected to the cooling assembly. Special high power thick wires (16AWG) were used to connect Peltier modules to power supply via relay. The same wires were used to connect fans via relays. Temperature sensors were fixed to necessary positions to measure cool airflow and room temperatures.

After the installation had been completed, the system's program block diagram was considered. Figure 36 shows a diagram of the program operation.

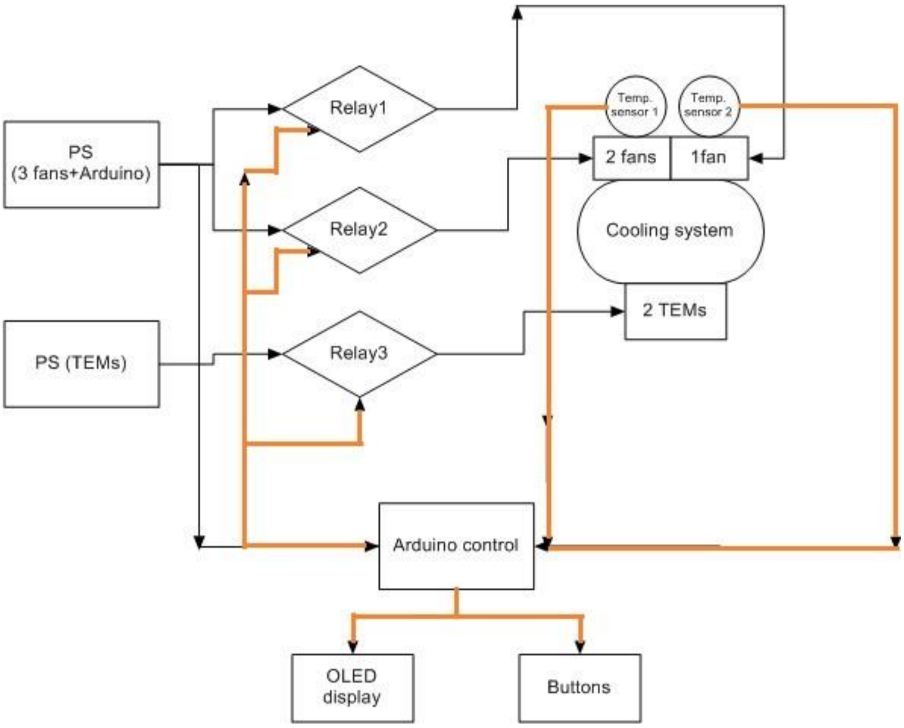


Figure 36. Control system block diagram.

As the diagram in Figure 36 indicates, one relay is connected to two Peltier elements, while two other relays control cold and hot air fans separately. Temperature sensors measure temperature at different points, and send those values to Arduino. Arduino switches relays according to temperature changes, powering on and off Peltier modules and/or fans. All those actions are controlled according to a code uploaded to Arduino from a PC.

First of all, before writing any program, strict actions have to be planned. It that case, it is necessary to define, which temperatures to measure, and what corresponding actions are expected from the code. As was decided, temperature of cold airflow above the floor as well as the room ambient temperature were measured. In that case, it is possible to compare the achieved coolness of airflow with the average temperature, and thus efficiency of the system.

Secondly, as the system configuration states, cool airflow is possible when the Peltier module operates. From the control system side, it means that the upper fan matters only if the Peltier element operates, what occurs when its relay is on. If sensed cool airflow temperature is higher than needed, this upper fan has to be stopped by its relay to allow system cool itself again. Moreover, bottom fans, designed for hot air dissipation, must be switched on always when the Peltier element is operating. According to these points, a program code was completed.

One test was done in advance before proceeding with the code. It was necessary to define how much time was needed for the Peltier module to become cold enough to produce efficiently cool airflow. That time was 6-8 minutes, and it then was noticed by the user to manage the system as timing was not planned to be implemented to the code.

8.3 Control system code implementation

The program was implemented according to the stated plan, in three versions, which differ by complexity. Each time the next version was slightly improved and more automated than the previous one. However, some parts of the algorithm were common in all versions.

The necessary basic code algorithms were defined in the beginning. Those parts compose the basic code structure. Listing 1 represents libraries declaration in the program.

```
#include "DHT.h" // Temperature sensor DHT library
#include <math.h>  // Math operations library
#include <Wire.h> // Communication with I2C/TWI devices
#include <SeeedGrayOLED.h> // OLED display library
#include <avr/pgmspace.h> // Flash memory communication
```

Listing 1. Main libraries definition.

The libraries announced in the beginning, set functions required for the operation of sensors and define output methods and communication with the user.

Another static part in all the code versions was pins assignment. Listing 2 describes all the defined pins.

```
#define DHTPIN A1 // 1st temperature sensor pin
#define DHTTYPE DHT22 // 2nd temperature sensor pin
const int buttonPin2 = 2; // number of the pushbutton pin
const int buttonPin3 = 3;
const int buttonPin4 = 4;
```

Listing 2. Pin declarations.

Buttons and sensors are attached to the digital pins of the Arduino Grove shield. The buttons are defined on 2nd, 3rd and 4th pins. The sensors' pin definitions are implemented via their libraries.

Listing 3 shows a void setup () function, which sets each of the digital channels, starts libraries and communication.

```
void setup()
{
dht.begin();
pinMode(6, OUTPUT); // Relay is attached to D6 port of Base Shield
pinMode(7, OUTPUT); // Relay is attached to D7 port of Base Shield
pinMode(8, OUTPUT); // Relay is attached to D8 port of Base Shield
// initialize the pushbutton pin as an input:
pinMode(buttonPin2, INPUT);
pinMode(buttonPin3, INPUT);
pinMode(buttonPin4, INPUT);
// OLED setup
Wire.begin();

SeeedGrayOled.init(); // initialize SEEED OLED display
SeeedGrayOled.clearDisplay(); // Clear Display.
```

```
SeeedGrayOled.setNormalDisplay(); // Set Normal
Display Mode
SeeedGrayOled.setVerticalMode(); // Set to vertical
mode for displaying tex
SeeedGrayOled.setGrayLevel(15); // Set Grayscale
level. Any number between 0
}
```

Listing 3. Setup algorithm.

In the void setup () function output and input pins are defined, the display is configured to output sensor values, and sensors' operation begins. The display gets instructions on writing the measured temperatures.

The temperature analysis segment, as well as displaying of the temperature values, have always been implemented in the same way. Listings 4 and 5 present those parts.

```
// Temperatures output
// Temp sensor
SeeedGrayOled.setTextXY(3,0);
SeeedGrayOled.putString("Temp1=");
SeeedGrayOled.putNumber(temperature);
// Temp&Hum sensor, Room temperature
SeeedGrayOled.setTextXY(4,0);
SeeedGrayOled.putString("TempRoom=");
SeeedGrayOled.putNumber(t);
```

Listing 4. Temperature output on OLED display.

```
// Temp. sensor measurements, airflow temperature measurement
a=analogRead(0);
resistance=(float)(1023-a)*10000/a; // get the resistance of the
sensor
temperature=1/(log(resistance/10000)/B+1/298.15)-273.15; //
convert to temperature
// Temp&Hum sensor measurements, Room temperature indication
float h = dht.readHumidity();
float t = dht.readTemperature();
```

Listing 5. Temperature analysis and analog to digital conversion.

According to Listings 4 and 5, temperature measurements were implemented by two different sensors with slightly different calculations. However, the output mechanism is the same for them both.

8.3.1 Versions 1 and 2

The first program version was fully manual. This means that all relays are switched by pressing buttons. The installed display helps a user to understand when the button must be pushed to change the power state. The block diagram is presented in Figure 37. If the indicated temperature of cool airflow is inappropriately high or low, the power of the Peltier element should be changed. Also fans have to be switched accordingly. The main disadvantage of that code is that the user has to follow temperature changes accurately to maintain the airflow necessarily cold.

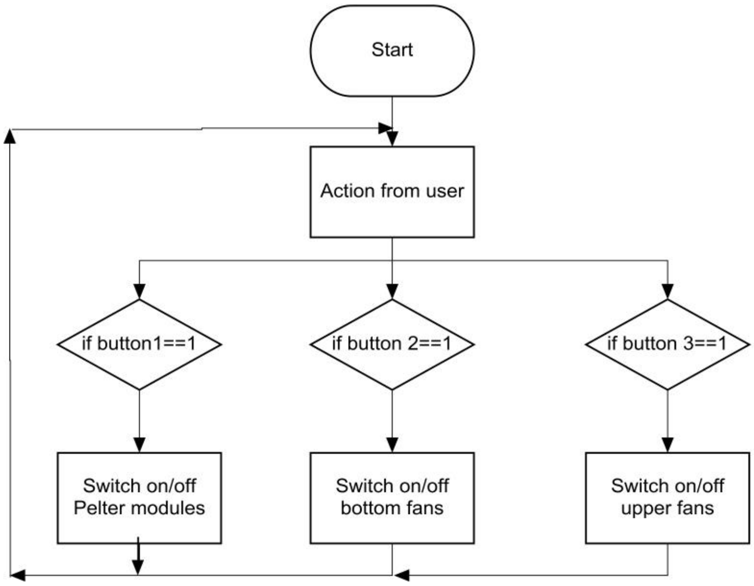


Figure 37. First code version block diagram.

The second version of the code was called “semi-automatic”, because part of the algorithm was still controlled manually. Figure 38 shows a block diagram of the code. The idea of that program was to avoid letting the cool airflow become inefficiently warm. In the case that the temperature is sensed higher than allowed, the upper fan must be switched off to maintain the cold side of the Peltier element to be cold enough again. It is done automatically. However, all other parts are still controlled manually. For example, the user must turn on the bottom fans with a button, when the Peltier modules are switched on (also with a button). Otherwise, the system will overheat and break.

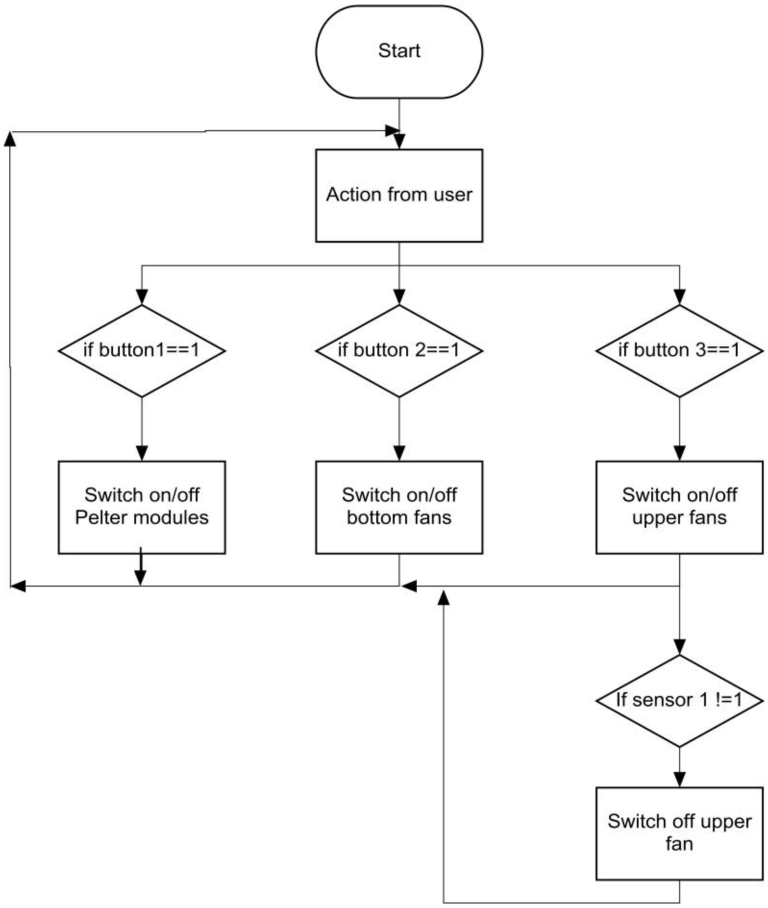


Figure 38. Second code version block diagram.

In two first program versions, only one temperature sensor's measurements (cool air flow temperature) were used. The second sensor's values (room temperature) were only displayed on a screen, but were not employed due to its inapplicability. That omission was improved in the next version. Appendix 1 and 2 contain a full code of these two versions.

8.3.2 Version 3

The last third program version is automated in the most possible way, which was appropriate on the current stage of the project, because no exact application was yet defined. The code diagram is represented in Figure 39. This code operates so that when Peltier modules are switched on manually by the user, the bottom fans are turned on automatically to exclude overheating. Moreover, the upper fan is programmed to turn off when the cool airflow becomes unnecessarily warm, which allows the system work more efficiently. After the time needed for the module to become cold again, the upper fans might be turned on again. All the temperatures are still displayed on the screen.

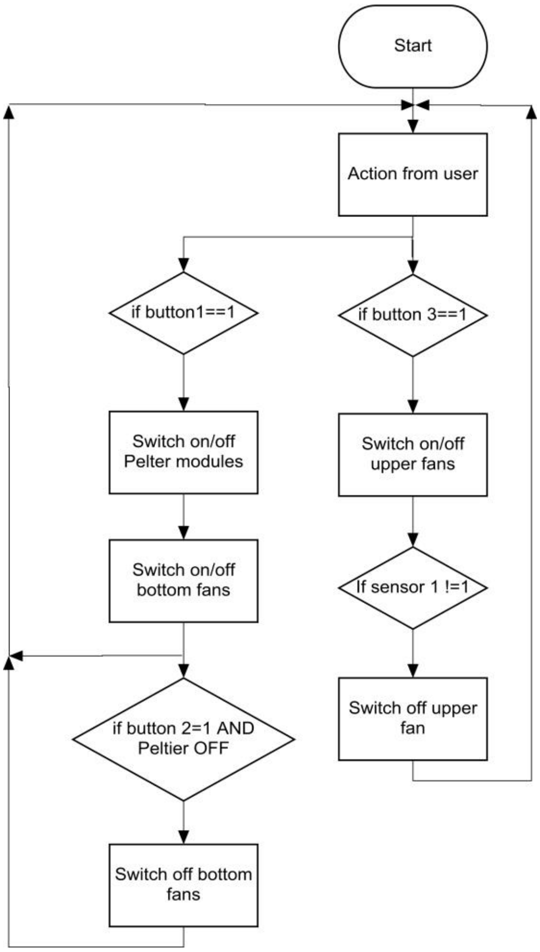


Figure 39. Third code version block diagram.

As Figure 3 shows, the bottom fans can be switched off manually as well as the other two parts. However, it is possible only when the Peltier modules are in an off-state. Otherwise, if the bottom fans are switched off when the Peltier modules operate, the system will be destroyed due to overheating.

For the last version, it was decided to compare the cool airflow temperature with the room temperature by the means of two sensors. Ineffective difference was defined as 2ºC. This means, for example, that if measured room temperature is 23ºC, then the upper fan will be turned off when the airflow temperature rises up to 21ºC. After that, the necessary maintenance time of several minutes has to be kept to recover heat sink coolness. Appendix 3 contains the full code of the third version.

8.4 Further development of the control system

At this point, the basic code of the control system is implemented. When some definite application is proposed, the program might be improved. However, several improvements might be done already at this stage.

First of all, some kind of timing mechanism should be implemented. The main disadvantage of the control system now is that the user has to notice the time to make sure that the cold heat sink is cooled enough to implement effectively cool airflow. However, there is an alternative to the timing algorithm. Another temperature sensor might be installed to measure the temperature of the cold heat sink. In that case, it is possible to switch on the upper fan only when this value is appropriate.

Secondly, a special cover should be designed to place the system inside. This enclosure should accommodate all Arduino modules. It should provide special apertures for wires which connect the relay with the system’s parts. Moreover, buttons and display have to be fixed attractively to allow smooth communication with the user. Moreover, if the whole cooling system is integrated into some big device of furniture, the control unit might be allocated inside as well, that wires will be outside, but only the buttons and a display will be interfaced.

The suggested improvements can be done immediately. They will improve the quality of the control unit designed, and will be more user-oriented. Nevertheless, at this prototyping stage, a partly manual system is still appropriate. The advantage is that it helps to test and evaluate the system effectively.

9 Results and discussion

9.1 Evaluation tests

After both the cooling system assembly and the control unit were physically implemented, evaluation tests were planned and completed. The main purpose of these tests was to define if that system was effective to be used in the required application. The concept of the project was to create cool airflow for a comfort working environment. According to that, temperature evaluation of the cool flow was done.

The most precise temperature measurements might be done with the sensor implemented in the control unit. However, writing values by hand from the OLED display was not a suitable option because it would not provide robust result due to personal factors. To collect temperature measurements effectively, Putty terminal was used. The Putty provided serial communication between Arduino and a PC. After the experiment, the temperature values were copied into Excel for analysis.

The experiment was done in a full operation cycle. After the Peltier modules and bottom fans, accordingly, were powered up, twelve minutes were counted. After that period, the upper fan was switched on generating cool airflow. The temperature sensor was measuring its coolness, constantly sending these values to a PC. At the same time, the normal room temperature was measured with the second sensor, and was constant over the experiment. According to the code, the upper fan switched off when further airflow became ineffective. Figure 40 represents a graph of the measurements collected by Putty and analysed in Excel.

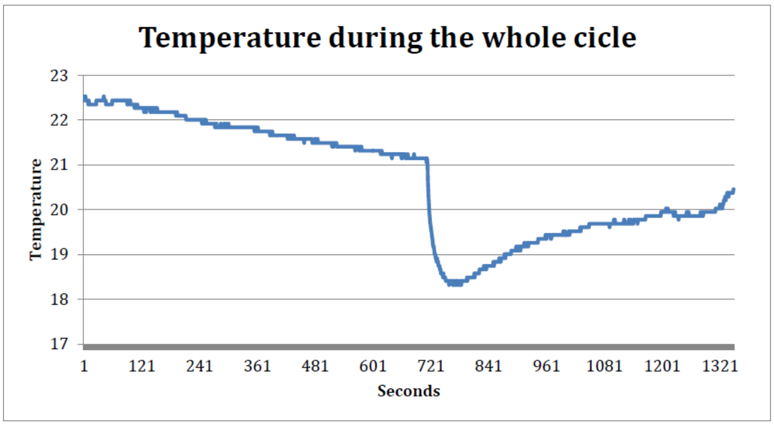


Figure 40. Temperature measurements graph.

Figure 40 shows dependence of temperature versus time in seconds. It is important to note that the upper fan was powered up at 721st second. Before that, the sensor was measuring cooling of the heat sink because they were located near each other.

In the beginning, the temperature was equal to the room temperature, which was 22.5ºC. During the first 12 minutes, the heat sink was cooling, affecting the sensor's indication. At 721st second, the upper fans switched on and the temperature of the airflow was measured as 18.2ºC. Over the next 10 minutes, the airflow temperature was gradually increasing before reaching 20.5ºC. At this point, the upper fan switched off automatically, because the detected temperature was 2ºC lower than the room one.

Besides airflow temperature evaluation, other values were noted by an ordinary thermometer presented by the thermocouple connected to the multimeter. The temperature of the cold heat sink was measured at its lowest point before turning on the upper fan. It was 8-9ºC, and approximately 5 minutes was required to achieve that value reliably. Consequently, 12 minutes, which were counted in the main test, can be reduced in two times.

Moreover, the point of interest was the temperature under the floor. The main concern about that was the possible floor warmth, which cou;d be caused by circulation of the hot air under the floor. For that reason, the temperature under the floor was measured at the latest point of the upper fan operation, and several minutes after. This helped later to define the time required for the hot heat sink to cool down before the new cycle of cold airflow. Figure 41 shows the graph with hot air temperature dynamics.

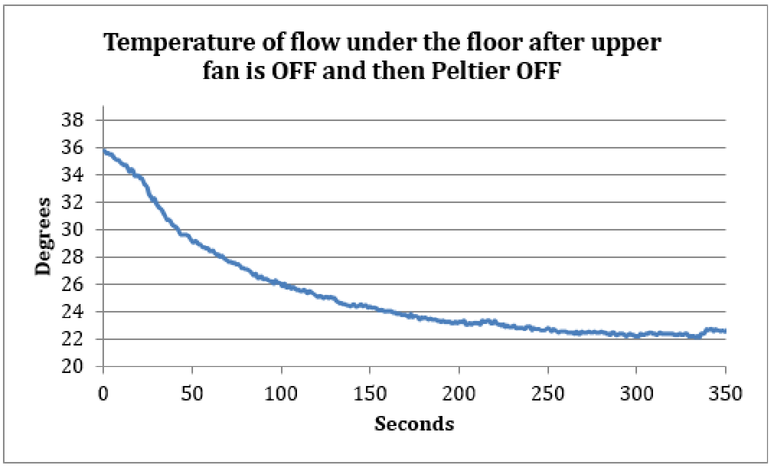


Figure 41. Hot air temperature changes under the floor after cool airflow cycle.

At the final moment of the upper fan operation, hot air under the floor was at the temperature of 36ºC. During the next period of almost 6 minutes, it was falling down to 22.5ºC, which is the room temperature. Although, 36ºC under the floor was 13ºC higher than in the room, the thick metal and floor cover secured the surrounding air in the room from warming up.

Analysing the results of several tests, it is clear that over the whole code cycle from starting Peltier modules until the full recovery after the cool airflow generated, the system can be effectively cooled up and recover from warming. Moreover, hot air circulation under the floor does not distract the cooling effect above the floor. Even though cool airflow is only around 4.5ºC lower than the room temperature, this difference is very tangible, as humans are very sensitive to very small variations. All in all, effectiveness of the created system is appropriate and can be used in further implementations.

9.2 System disadvantages

The main purpose of the project was successfully achieved. Effective cool airflow above the floor was generated and controlled. However, several obvious disadvantages of the system were revealed.

First of all, noise from fans is inappropriately high when all three devices operate at the same time. One fan has a sound level of 40 dB in full power operation. For comparison, normal sound level conversation creates 60 dB noise. When three fans work simultaneously, a silent room becomes noisy and can disturb people working inside. This drawback might be improved later by purchasing special low-noise fans, which could improve the situation.

Another serious disadvantage of the whole system is its power consumption. Table 5 shows power requirements for each part of the system separately and as a whole.

Table 5. System’s power requirements.

	Voltage, V	Current, A	Power, W
Peltier element	16	20	320
Fans	12	3.6	43.2
Arduino	12	1	12
Sum		24.6	375.2

According to Table 5, plug-in AC-DC power supply of 16V and 25A is required. Devices with a power higher than 400W, are freely available on the market. However, they are expensive. For that reason, no power unit was purchased yet, before the exact application would be defined.

9.3 Further system development and plans

Now the cooling module is fully prototyped. Although the control system code is ready as well, it requires to be properly covered and installed. However, as mentioned earlier in section 9.2 it depends on the application which will be stated later.

Another detail for improvement is power supply installation. If a unit with 16V output is acquired, 12V voltage for fans and Arduino has to be restricted. Moreover, proper power connection of the parts has to be planned. It is necessary to consider which parts can be powered via a single connector and which must be separated.

At the moment, preliminary plans concerning the application are defined. It is possible that the system will be integrated into a special floor stand, but not directly into the floor. In addition, the necessity of half-size system is under consideration. In that case, only one Peltier element should be implemented, instead of two as in the current prototype.

10 Conclusion

To summarize, the targets stated at the beginning of the project were achieved. A prototype of the cooling system and its control system, as well as their integration into the flooring system, were successfully implemented. Advantages and disadvantages were considered, and further development was planned. Future improvements will depend on the proposed application and will be completed accordingly.

The project was performed according to the stated steps without any dramatic changes in the plans. At first, the technology was learned by theoretical research and market analyses. Then the design and practical implementation of the physical cooling system were implemented. The control system was the last step in the main plan. To finalize the current project stage, testing and planning of further improvements were done.

All in all, the designed cooling system produces cool airflow effectively. Hot air dissipation under the floor is sufficient to avoid floor warming. The high power consumption requirement is significant disadvantage, but can be satisfied with a suitable power supply. Noise might be reduced with less noisy fans. A control system enclosure must be designed in the future, according to the specific application. To conclude, the cooling system can effectively satisfy the working environment requirements, and be used in the targeted way.

References

1. G. Jeffrey Snyder and Eric S. Toberer "Complex Thermoelectric Materials" Nature Materials 7 [online], 105-114 (2008).
   URL:http://www.thermoelectrics.caltech.edu/thermoelectrics/index.html
   Accessed 5 October 2014
2. Physics 212 Web Project “The Peltier Effect and Thermoelectric Cooling”, March 2007 [online].
   URL:http://ffden-2.phys.uaf.edu/212_spring2007.web.dir/sedona_price/phys_212_webproj_peltier.html
   Accessed 5 October 2014
3. Yaqing Qil, Zhihua Liz, Jianzhong Zhang. “Peltier Temperature Controlled Box for Test Circuit Board” [online]. Department of Electronics, Tianjin University. Tianjin, China, 2003.
   URL:http://ieeexplore.ieee.org.ezproxy.metropolia.fi/stamp/stamp.jsp?tp=&arnumber=5631607
   Accessed 5 October 2014
4. Song Shaojing, Qinqin. “Temperature Control of Thermoelectric cooler Based on Adaptive NN-PID” [online]. International Conference on Electrical and Control Engineering, Shanghai, China, 2010.
   URL:http://ieeexplore.ieee.org.ezproxy.metropolia.fi/stamp/stamp.jsp?tp=&arnumber=1287595
   Accessed 10 October 2014
5. A. Weingarten. R. Shpitalnik, S. Alexiou. “Observations of Magnetic and Electric Fields in a Nanosecond Plasma Opening Switch Experiment” [online]. Department of Particle Physics, Weiunann Institute of Science, Rehovot, Israel, 1995.
   URL:http://ieeexplore.ieee.org.ezproxy.metropolia.fi/stamp/stamp.jsp?tp=&arnumber=531618
   Accessed 10 October 2014
6. Nandini K.K., Muralidhara, “PELTIER BASED CABINET COOLING SYSTEM USING HEAT PIPE AND LIQUID BASED HEAT SINK” [online]. Department of Mechanical Engineering, NMAMIT, Nitte, India, 2013.
   URL:http://ieeexplore.ieee.org.ezproxy.metropolia.fi/stamp/stamp.jsp?tp=&arnumbe

r=6851583

Accessed 10 October 2014

7. Arduino official web site, “About Arduino”.
   URL:http://arduino.cc/en/Guide/HomePage
   Accessed 20 October 2014
8. Jenn Ratliff, Bronwyn Harris, “What is the Peltier Effect?” [online]. 2003-2015 Conjecture Corporation, January 2015.
   URL:http://www.wisegeek.org/what-is-the-peltier-effect.htm
   Accessed 1 November 2014
9. Klaus Froboese, “The Peltier effect” [online]. University of Konstanz, Germany, June 2000.
   URL:http://www.uni-konstanz.de/FuF/Physik/Jaeckle/papers/thermopower/node2.html
   Accessed 1 November 2014
10. Ferrotec Corporation, Thermoelectric Cooling Versus Traditional Refrigeration, USA.
    URL:https://www.ferrotec.com/technology/thermoelectric/teVsTraditional/
    Accessed 7 November 2014
11. Dr. Jens Thielmann, “Thermoelectric Cooling Technology” [online]. BINDER GmbH, Germany, February 2013.
    URL:www.**binder**-world.com/ext/getdownload.cfm?pud=74
    Accessed 7 November 2014
12. Ferrotec Corporation, Thermoelectric Technical Reference — Basic Principles of Thermoelectric Materials, USA.
    URL:https://www.ferrotec.com/technology/thermoelectric/thermalRef02/
    Accessed 7 November 2014
13. TE technology Inc., Cold Plate Coolers, USA.
    URL:https://tetech.com/product-category/cold-plate-coolers/
    Accessed 7 November 2014
14. TE technology Inc., Liquid Coolers, USA.
    URL:https://tetech.com/product-category/liquid-coolers/
    Accessed 7 November 2014
15. TE technology Inc., Air Coolers, USA.
    URL:https://tetech.com/product-category/air-coolers/
    Accessed 7 November 2014

16. Laird Technologies, Thermoelectric modules catalog, UltraTECTM Series [online]. URL:http://www.lairdtech.com/brandworld/library/THR-DS-UT15,12,F2,4040%201013.pdf
Accessed 15 November 2014

17. WARREN M. ROHSENOW, “Thermal to electrical energy conversion technologies” [online]. Mechanical Engineering department, Mitopen Courseware, Massachusetts Institute of Technology, USA, fall 2009.
URL:http://ocw.mit.edu/courses/mechanical-engineering/2-997-direct-solar-thermal-to-electrical-energy-conversion-technologies-fall-2009/audio-lectures/MIT2_997F09_lec02.pdf
Accessed 14 November 2014

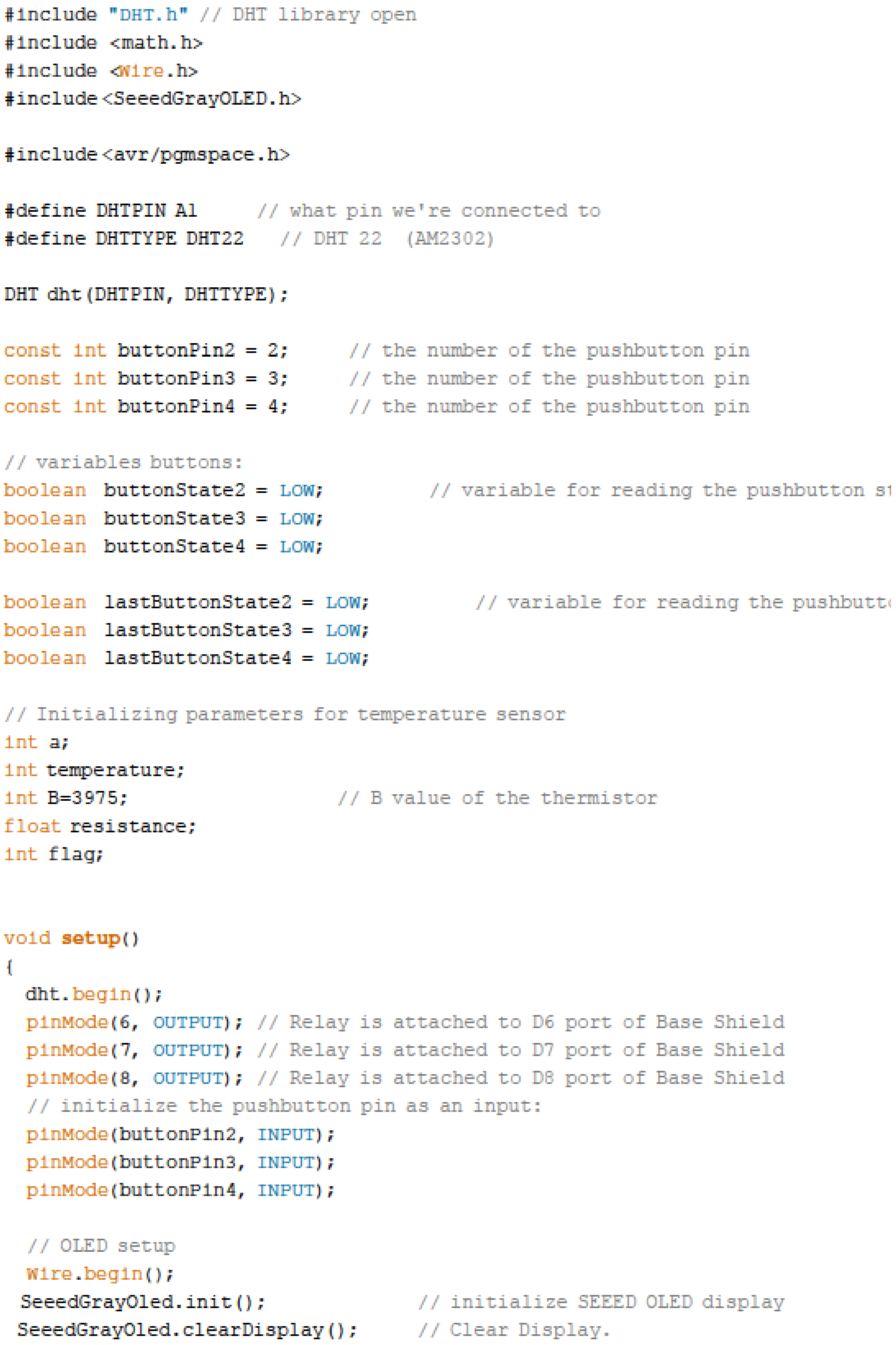
19. Ferrotec Corporation, Thermoelectric Technical Reference — Introduction to Thermoelectric Cooling, USA.
URL:https://www.ferrotec.com/technology/thermoelectric/thermalRef01/
Accessed 14 November 2014

20. Nat Bowers, “Scavenger hunt” [online]. Electronic Specifier, United Kingdom, October 2013. URL:http://alternativeenergy.electronicspecifier.com/energy-harvesting/eteg-hv56-nextreme-es-design-magazine-scavenger-hunt
Accessed 14 November 2014

21. Xuejun Yue, ”Design of Temperature Sensor on LM75” [online]. Fifth International Conference on MEMS NANO, and Smart Systems, Guangzhou, China, 2009.
URL:http://ieeexplore.ieee.org.ezproxy.metropolia.fi/stamp/stamp.jsp?tp=&arnumber=5489305

22. Clemens J. M. Lasance, ” The Seebeck Coefficient” [online]. ElectronicsCooling, November 2006.
URL: http://www.electronics-cooling.com/2006/11/the-seebeck-coefficient/

Code version 1

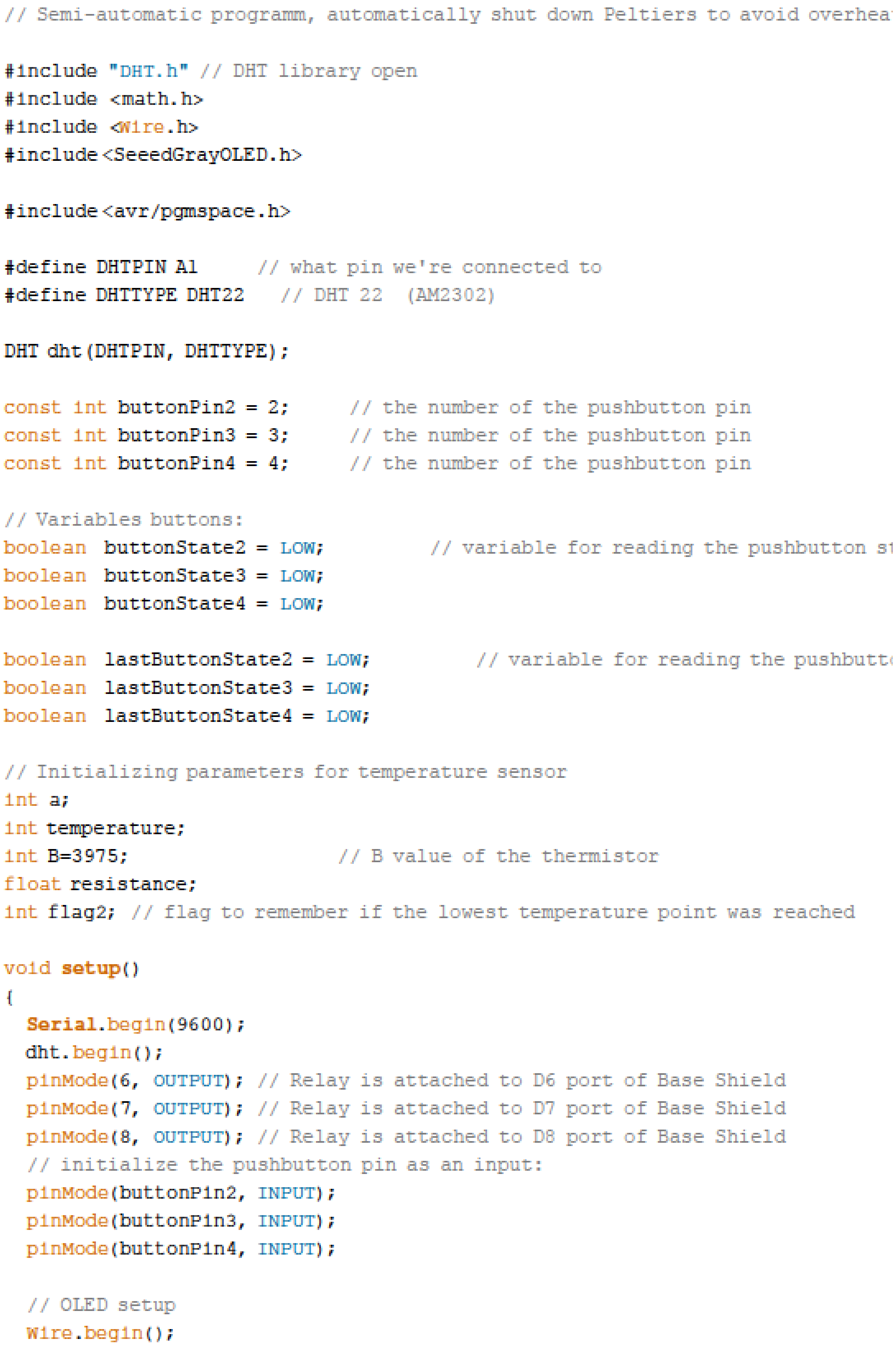


```
#include "DHT.h" // DHT library open
#include <math.h>
#include <Wire.h>
#include<SeeedGrayOLED.h>

#include<avr/pgmspace.h>

#define DHTPIN A1     // what pin we're connected to
#define DHTTYPE DHT22   // DHT 22  (AM2302)

DHT dht(DHTPIN, DHTTYPE);

const int buttonPin2 = 2;     // the number of the pushbutton pin
const int buttonPin3 = 3;     // the number of the pushbutton pin
const int buttonPin4 = 4;     // the number of the pushbutton pin

// variables buttons:
boolean buttonState2 = LOW;         // variable for reading the pushbutton s
boolean buttonState3 = LOW;
boolean buttonState4 = LOW;

boolean lastButtonState2 = LOW;         // variable for reading the pushbutt
boolean lastButtonState3 = LOW;
boolean lastButtonState4 = LOW;

// Initializing parameters for temperature sensor
int a;
int temperature;
int B=3975;                  // B value of the thermistor
float resistance;
int flag;


void setup()
{
  dht.begin();
  pinMode(6, OUTPUT); // Relay is attached to D6 port of Base Shield
  pinMode(7, OUTPUT); // Relay is attached to D7 port of Base Shield
  pinMode(8, OUTPUT); // Relay is attached to D8 port of Base Shield
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin2, INPUT);
  pinMode(buttonPin3, INPUT);
  pinMode(buttonPin4, INPUT);

  // OLED setup
  Wire.begin();
 SeeedGrayOled.init();             // initialize SEEED OLED display
 SeeedGrayOled.clearDisplay();     // Clear Display.
```

```
 SeeedGrayOled.setNormalDisplay(); // Set Normal Display Mode
 SeeedGrayOled.setVerticalMode();  // Set to vertical mode for displaying te:
  //SeeedGrayOled.setTextXY(0,0);  //set Cursor to ith line, 0th column
 SeeedGrayOled.setGrayLevel(15); //Set Grayscale level. Any number between 0

}


void loop()
{

//////////////////////////////////////RELAYS & Buttons/////////////////////

  buttonState2 = digitalRead(buttonPin2);
  buttonState3 = digitalRead(buttonPin3);
  buttonState4 = digitalRead(buttonPin4);

     if (buttonState2==HIGH && lastButtonState2==LOW)
    {
      digitalWrite(6, !digitalRead(6));
    }

    if (buttonState3==HIGH && lastButtonState3==LOW)
    {
      digitalWrite(7, !digitalRead(7));
    }

    if (buttonState4==HIGH && lastButtonState4==LOW)
    {
      digitalWrite(8, !digitalRead(8));
    }

    lastButtonState2 = buttonState2;
    lastButtonState3 = buttonState3;
    lastButtonState4 = buttonState4;

////////////////////////////////////TEMPERATURE/////////////////////////////

    a=analogRead(0);
    resistance=(float)(1023-a)*10000/a; // get the resistance of the sensor
    temperature=1/(log(resistance/10000)/B+1/298.15)-273.15;// convert to ter

    float h = dht.readHumidity();
    float t = dht.readTemperature();


////////////////////////////////////Display/////////////////////////////////

   ///Relay 6
```

```
  if (digitalRead(6)==HIGH)
   {
    SeeedGrayOled.setTextXY(0,0);
    SeeedGrayOled.putString("R6 ON ");
   }
   else
   {
    SeeedGrayOled.setTextXY(0,0);
    SeeedGrayOled.putString("R6 OFF");
   }

   ///Relay 7
   if (digitalRead(7)==HIGH)
   {
    SeeedGrayOled.setTextXY(1,0);
    SeeedGrayOled.putString("R7 ON ");
   }
   else
   {
    SeeedGrayOled.setTextXY(1,0);
    SeeedGrayOled.putString("R7 OFF");
   }

   ///Relay 8
   if (digitalRead(8)==HIGH)
   {
    SeeedGrayOled.setTextXY(2,0);
    SeeedGrayOled.putString("R8 ON ");
   }
   else
   {
    SeeedGrayOled.setTextXY(2,0);
    SeeedGrayOled.putString("R8 OFF");
   }

  // Display on OLED screen
    SeeedGrayOled.setTextXY(3,0);
    SeeedGrayOled.putString("Temp1=");
    SeeedGrayOled.putNumber(temperature);


    SeeedGrayOled.setTextXY(4,0);
    SeeedGrayOled.putString("Temp2=");
    SeeedGrayOled.putNumber(t);

}
```

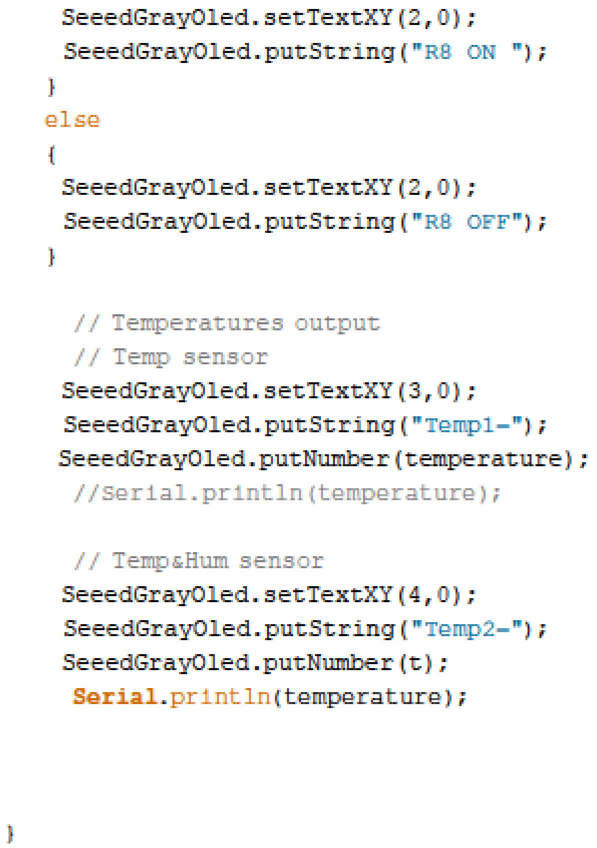
Code version 2



```
// Semi-automatic programm, automatically shut down Peltiers to avoid overhea

#include "DHT.h" // DHT library open
#include <math.h>
#include <Wire.h>
#include<SeeedGrayOLED.h>

#include<avr/pgmspace.h>

#define DHTPIN A1     // what pin we're connected to
#define DHTTYPE DHT22   // DHT 22  (AM2302)

DHT dht(DHTPIN, DHTTYPE);

const int buttonPin2 = 2;     // the number of the pushbutton pin
const int buttonPin3 = 3;     // the number of the pushbutton pin
const int buttonPin4 = 4;     // the number of the pushbutton pin

// Variables buttons:
boolean  buttonState2 = LOW;         // variable for reading the pushbutton st
boolean  buttonState3 = LOW;
boolean  buttonState4 = LOW;

boolean  lastButtonState2 = LOW;         // variable for reading the pushbutt
boolean  lastButtonState3 = LOW;
boolean  lastButtonState4 = LOW;

// Initializing parameters for temperature sensor
int a;
int temperature;
int B=3975;                  // B value of the thermistor
float resistance;
int flag2; // flag to remember if the lowest temperature point was reached

void setup()
{
  Serial.begin(9600);
  dht.begin();
  pinMode(6, OUTPUT); // Relay is attached to D6 port of Base Shield
  pinMode(7, OUTPUT); // Relay is attached to D7 port of Base Shield
  pinMode(8, OUTPUT); // Relay is attached to D8 port of Base Shield
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin2, INPUT);
  pinMode(buttonPin3, INPUT);
  pinMode(buttonPin4, INPUT);

  // OLED setup
  Wire.begin();
```

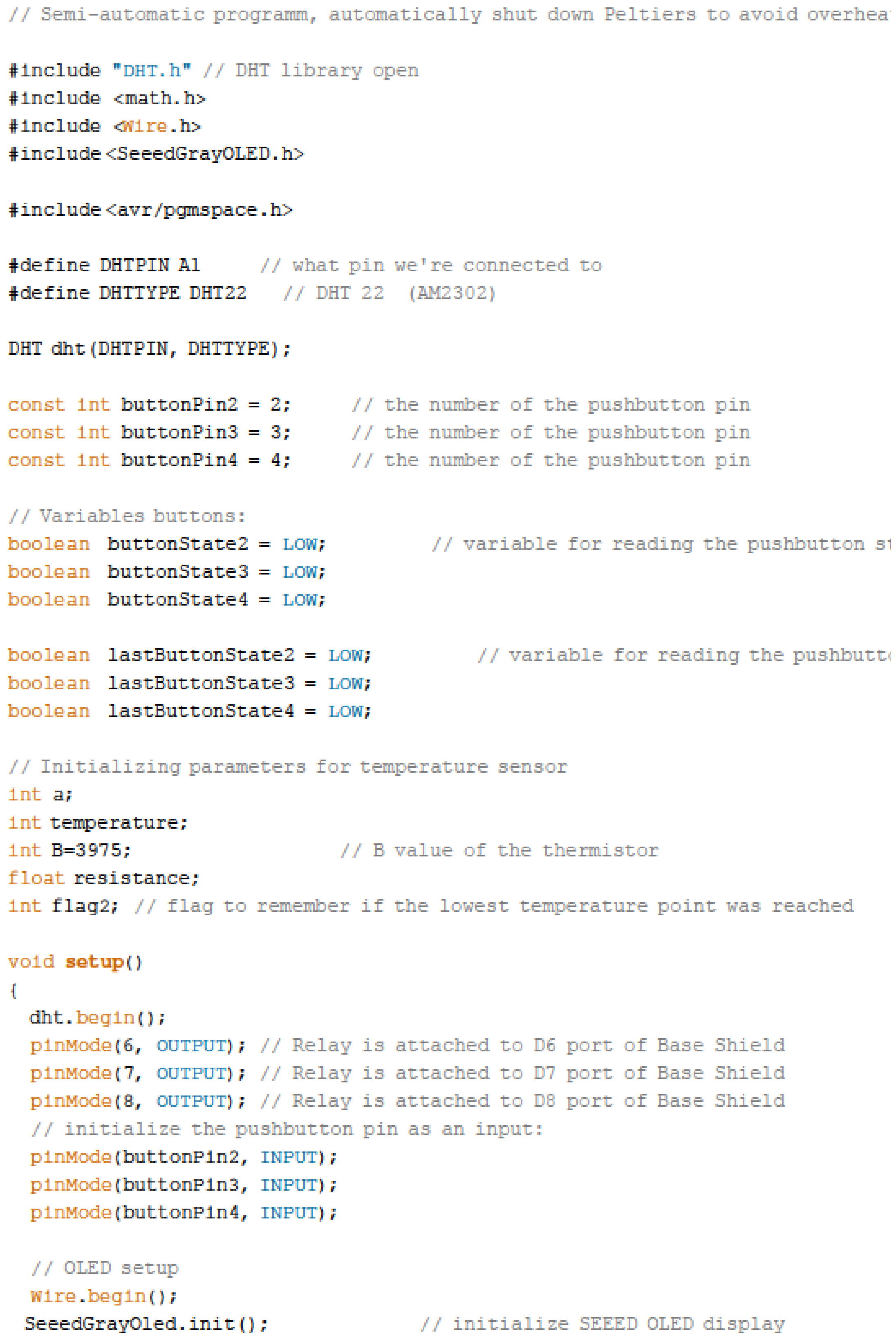
```
 SeeedGrayOled.init();              // initialize SEEED OLED display
 SeeedGrayOled.clearDisplay();      // Clear Display.
 SeeedGrayOled.setNormalDisplay(); // Set Normal Display Mode
 SeeedGrayOled.setVerticalMode();  // Set to vertical mode for displaying te
 SeeedGrayOled.setGrayLevel(15); // Set Grayscale level. Any number between
}


void loop()
{

////////////////////////////////////RELAYS & Buttons/////////////////////

  buttonState2 = digitalRead(buttonPin2);
  buttonState3 = digitalRead(buttonPin3);
  buttonState4 = digitalRead(buttonPin4);

    // If Peltiers button pressed- change Peltiers state
    if (buttonState2==HIGH && lastButtonState2==LOW)
    {
      digitalWrite(6, !digitalRead(6));
      // Always switch ON bottom fans if Peltiers are ON, they can be switche
      if (digitalRead(6)==HIGH)
      digitalWrite(8, HIGH);
    }


    if (temperature == 18)
      flag2=1; // After switching


    // Switch OFF R6 is temp limit was exceeded
    if (flag2==1 && temperature > 19)
    {
      digitalWrite(6, LOW);
      flag2=0;
    }


    // Upper fan switching on only manually (supposed to be after 8-9 minutes
    if (buttonState3==HIGH && lastButtonState3==LOW)
    {
            digitalWrite(7, !digitalRead(7));
    }

    // Manual switching off of bottom fans only if Peltiers are down
    if (buttonState4==HIGH && lastButtonState4==LOW && digitalRead(6)==LOW)
    {
      digitalWrite(8, !digitalRead(8));
```

```
    }



   lastButtonState2 = buttonState2;
   lastButtonState3 = buttonState3;
   lastButtonState4 = buttonState4;



////////////////////////////////////TEMPERATURE/////////////////////////////

    a=analogRead(0);
    resistance=(float)(1023-a)*10000/a; // get the resistance of the sensor
    temperature=1/(log(resistance/10000)/B+1/298.15)-273.15;// convert to tem

    float h = dht.readHumidity();
    float t = dht.readTemperature();


////////////////////////////////////Display/////////////////////////////////

   // Relay 6
   if (digitalRead(6)==HIGH)
    {
     SeeedGrayOled.setTextXY(0,0);
     SeeedGrayOled.putString("R6 ON ");
    }
    else
    {
     SeeedGrayOled.setTextXY(0,0);
     SeeedGrayOled.putString("R6 OFF");
    }

    // Relay 7
    if (digitalRead(7)==HIGH)
    {
     SeeedGrayOled.setTextXY(1,0);
     SeeedGrayOled.putString("R7 ON ");
    }
    else
    {
     SeeedGrayOled.setTextXY(1,0);
     SeeedGrayOled.putString("R7 OFF");
    }

    // Relay 8
    if (digitalRead(8)==HIGH)
    {
```

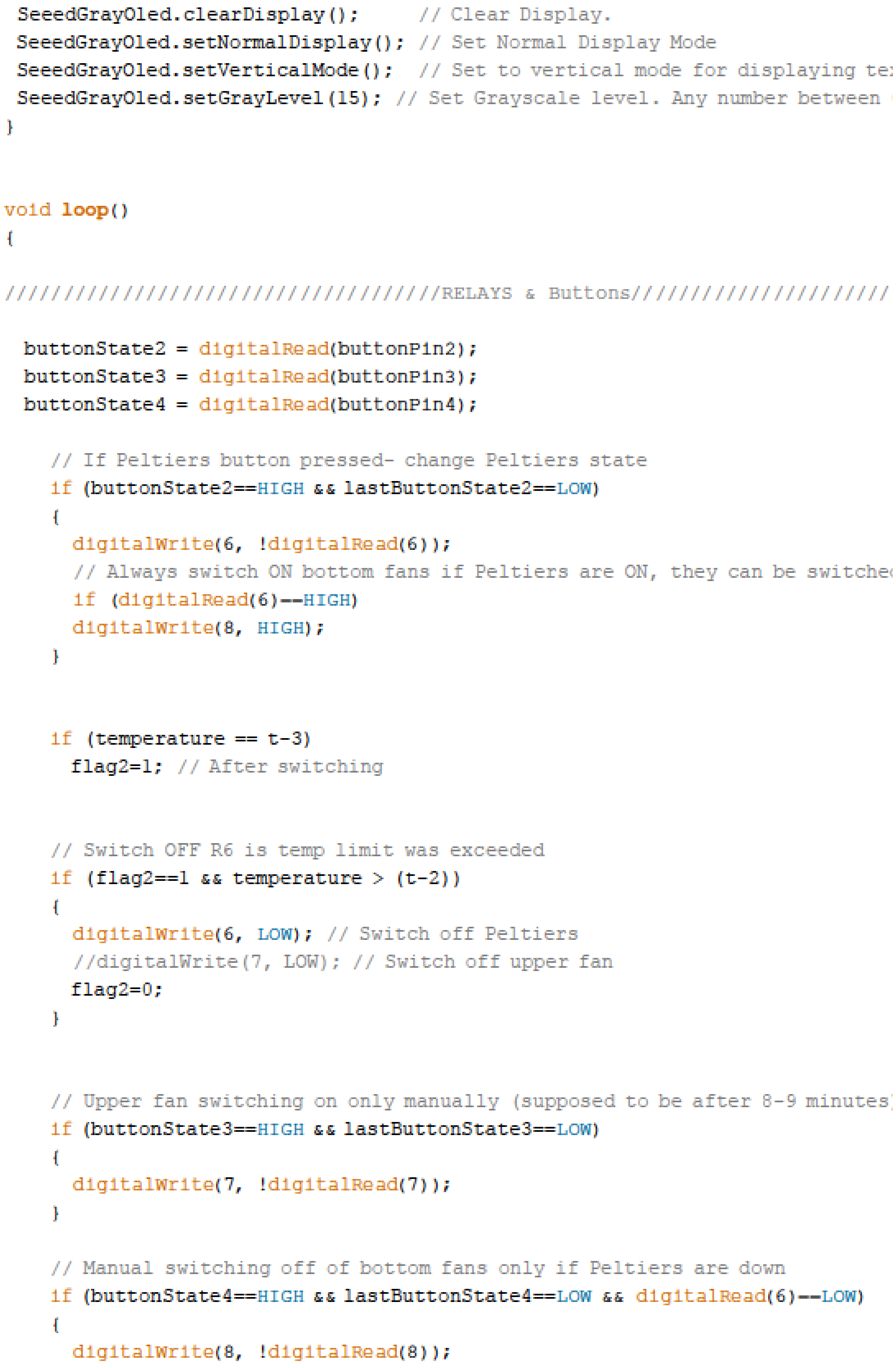


```
     SeeedGrayOled.setTextXY(2,0);
     SeeedGrayOled.putString("R8 ON ");
   }
   else
   {
     SeeedGrayOled.setTextXY(2,0);
     SeeedGrayOled.putString("R8 OFF");
   }

      // Temperatures output
      // Temp sensor
     SeeedGrayOled.setTextXY(3,0);
     SeeedGrayOled.putString("Temp1-");
    SeeedGrayOled.putNumber(temperature);
      //Serial.println(temperature);

      // Temp&Hum sensor
     SeeedGrayOled.setTextXY(4,0);
     SeeedGrayOled.putString("Temp2-");
     SeeedGrayOled.putNumber(t);
      Serial.println(temperature);



}
```

Code version 3

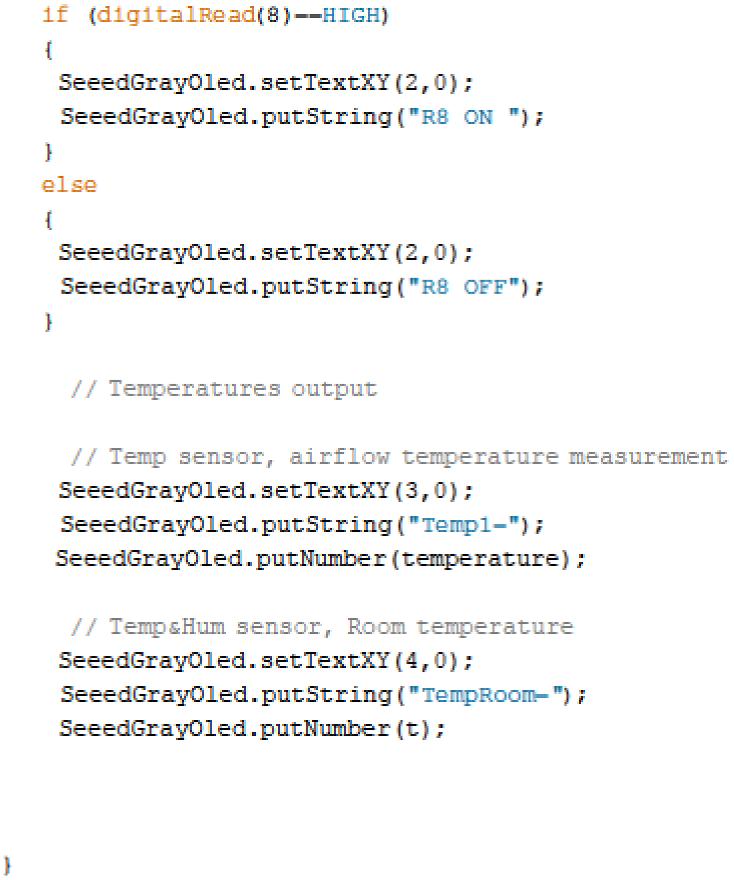


```
// Semi-automatic programm, automatically shut down Peltiers to avoid overhea

#include "DHT.h" // DHT library open
#include <math.h>
#include <Wire.h>
#include <SeeedGrayOLED.h>

#include <avr/pgmspace.h>

#define DHTPIN A1     // what pin we're connected to
#define DHTTYPE DHT22   // DHT 22  (AM2302)

DHT dht(DHTPIN, DHTTYPE);

const int buttonPin2 = 2;     // the number of the pushbutton pin
const int buttonPin3 = 3;     // the number of the pushbutton pin
const int buttonPin4 = 4;     // the number of the pushbutton pin

// Variables buttons:
boolean  buttonState2 = LOW;         // variable for reading the pushbutton st
boolean  buttonState3 = LOW;
boolean  buttonState4 = LOW;

boolean  lastButtonState2 = LOW;         // variable for reading the pushbutt
boolean  lastButtonState3 = LOW;
boolean  lastButtonState4 = LOW;

// Initializing parameters for temperature sensor
int a;
int temperature;
int B=3975;                  // B value of the thermistor
float resistance;
int flag2; // flag to remember if the lowest temperature point was reached

void setup()
{
  dht.begin();
  pinMode(6, OUTPUT); // Relay is attached to D6 port of Base Shield
  pinMode(7, OUTPUT); // Relay is attached to D7 port of Base Shield
  pinMode(8, OUTPUT); // Relay is attached to D8 port of Base Shield
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin2, INPUT);
  pinMode(buttonPin3, INPUT);
  pinMode(buttonPin4, INPUT);

  // OLED setup
  Wire.begin();
 SeeedGrayOled.init();             // initialize SEEED OLED display
```



```
  SeeedGrayOled.clearDisplay();     // Clear Display.
  SeeedGrayOled.setNormalDisplay(); // Set Normal Display Mode
  SeeedGrayOled.setVerticalMode();  // Set to vertical mode for displaying te
  SeeedGrayOled.setGrayLevel(15); // Set Grayscale level. Any number between
}


void loop()
{

//////////////////////////////////////RELAYS & Buttons//////////////////////

  buttonState2 = digitalRead(buttonPin2);
  buttonState3 = digitalRead(buttonPin3);
  buttonState4 = digitalRead(buttonPin4);

    // If Peltiers button pressed- change Peltiers state
    if (buttonState2==HIGH && lastButtonState2==LOW)
    {
      digitalWrite(6, !digitalRead(6));
      // Always switch ON bottom fans if Peltiers are ON, they can be switche
      if (digitalRead(6)==HIGH)
      digitalWrite(8, HIGH);
    }


    if (temperature == t-3)
      flag2=1; // After switching


    // Switch OFF R6 is temp limit was exceeded
    if (flag2==1 && temperature > (t-2))
    {
      digitalWrite(6, LOW); // Switch off Peltiers
      //digitalWrite(7, LOW); // Switch off upper fan
      flag2=0;
    }


    // Upper fan switching on only manually (supposed to be after 8-9 minutes
    if (buttonState3==HIGH && lastButtonState3==LOW)
    {
      digitalWrite(7, !digitalRead(7));
    }

    // Manual switching off of bottom fans only if Peltiers are down
    if (buttonState4==HIGH && lastButtonState4==LOW && digitalRead(6)==LOW)
    {
      digitalWrite(8, !digitalRead(8));
```

```
    }



   lastButtonState2 = buttonState2;
   lastButtonState3 = buttonState3;
   lastButtonState4 = buttonState4;



///////////////////////////////////TEMPERATURE/////////////////////////////

    // Temp. sensor measurements, airflow temperature measurement
    a=analogRead(0);
   resistance=(float)(1023-a)*10000/a; // get the resistance of the sensor
   temperature=1/(log(resistance/10000)/B+1/298.15)-273.15;// convert to tem

    // Temp&Hum sensor measurements, Room temperature indication
    float h = dht.readHumidity();
    float t = dht.readTemperature();


///////////////////////////////////Display/////////////////////////////////

   // Relay 6
   if (digitalRead(6)==HIGH)
    {
     SeeedGrayOled.setTextXY(0,0);
     SeeedGrayOled.putString("R6 ON ");
    }
    else
    {
     SeeedGrayOled.setTextXY(0,0);
     SeeedGrayOled.putString("R6 OFF");
    }

    // Relay 7
    if (digitalRead(7)==HIGH)
    {
     SeeedGrayOled.setTextXY(1,0);
     SeeedGrayOled.putString("R7 ON ");
    }
    else
    {
     SeeedGrayOled.setTextXY(1,0);
     SeeedGrayOled.putString("R7 OFF");
    }

    // Relay 8
```



```
   if (digitalRead(8)==HIGH)
   {
    SeeedGrayOled.setTextXY(2,0);
    SeeedGrayOled.putString("R8 ON ");
   }
   else
   {
    SeeedGrayOled.setTextXY(2,0);
    SeeedGrayOled.putString("R8 OFF");
   }

     // Temperatures output

     // Temp sensor, airflow temperature measurement
    SeeedGrayOled.setTextXY(3,0);
    SeeedGrayOled.putString("Temp1-");
    SeeedGrayOled.putNumber(temperature);

     // Temp&Hum sensor, Room temperature
    SeeedGrayOled.setTextXY(4,0);
    SeeedGrayOled.putString("TempRoom-");
    SeeedGrayOled.putNumber(t);



}
```