Group No 15

Ritik
(20IE10051)

Samridhi Sinha
(20IE10052)

Arunesh Manna
(20IE10053)

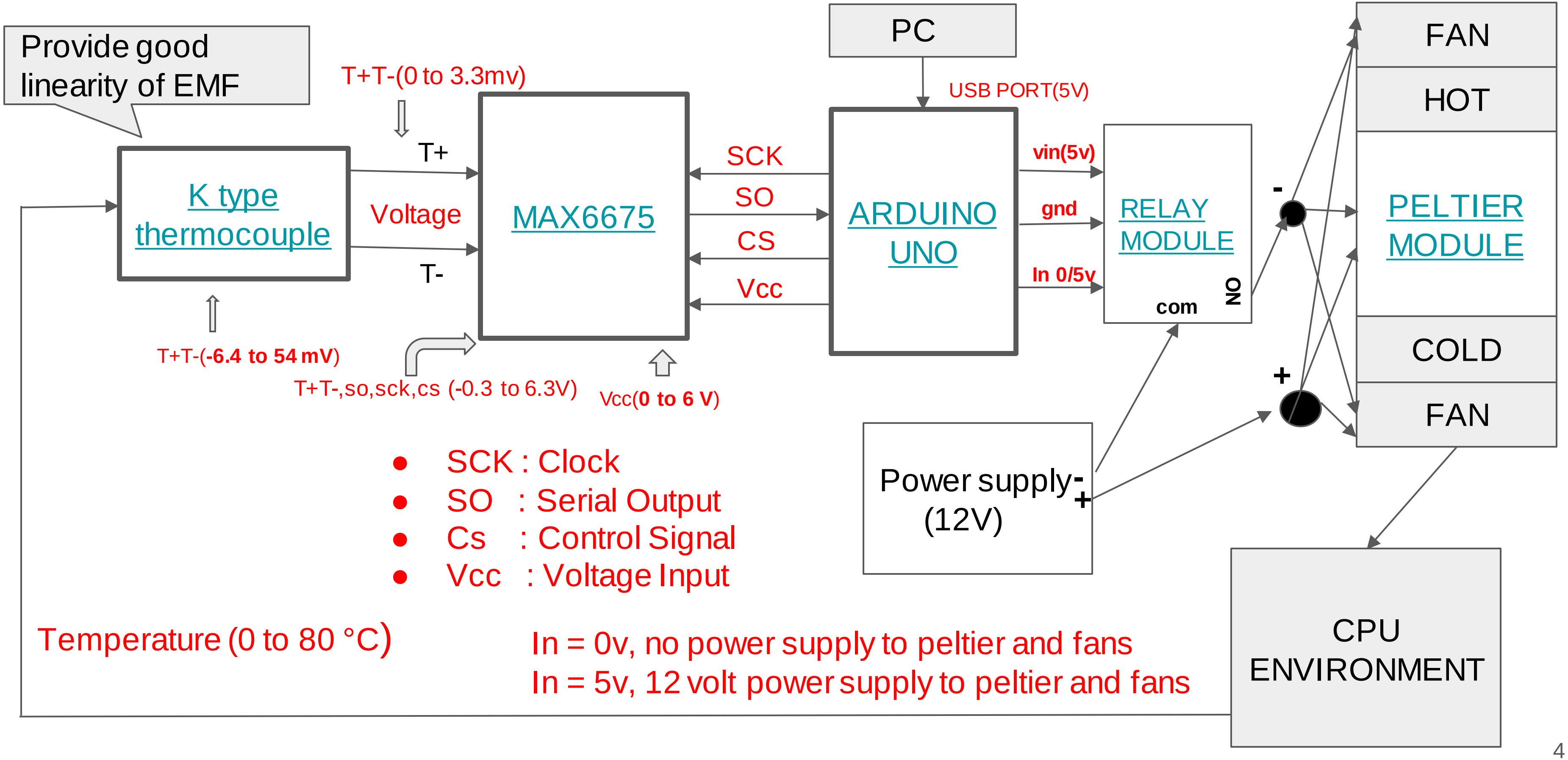
Problem Statement

To design an IC cooling system using the Peltier effect to embed it on the CPU

Bill of Material

S.No	Components	Amount (in Rs)
1.	K-type Thermocouple	100
2.	MAX6675 Module	300
3.	Arduino-UNO	600
4.	Peltier Module Kit	1100
5.	Relay Module	75
6.	Other Components (Wires,Breadboard,Veroboard)	200
7.	Power Supply	__
	Total	2375

Block Diagram



Provide good linearity of EMF

K type thermocouple

T+T-(-6.4 to 54 mV)

T+T-(0 to 3.3mv)

T+

Voltage

T-

MAX6675

T+T-,so,sck,cs (-0.3 to 6.3V)

Vcc(0 to 6 V)

SCK

SO

CS

Vcc

PC

USB PORT(5V)

ARDUINO UNO

vin(5v)

gnd

In 0/5v

RELAY MODULE

com

NO

-

+

FAN

HOT

PELTIER MODULE

COLD

FAN

Power supply (12V)

-

+

CPU ENVIRONMENT

- SCK : Clock
- SO : Serial Output
- Cs : Control Signal
- Vcc : Voltage Input

Temperature (0 to 80 °C)

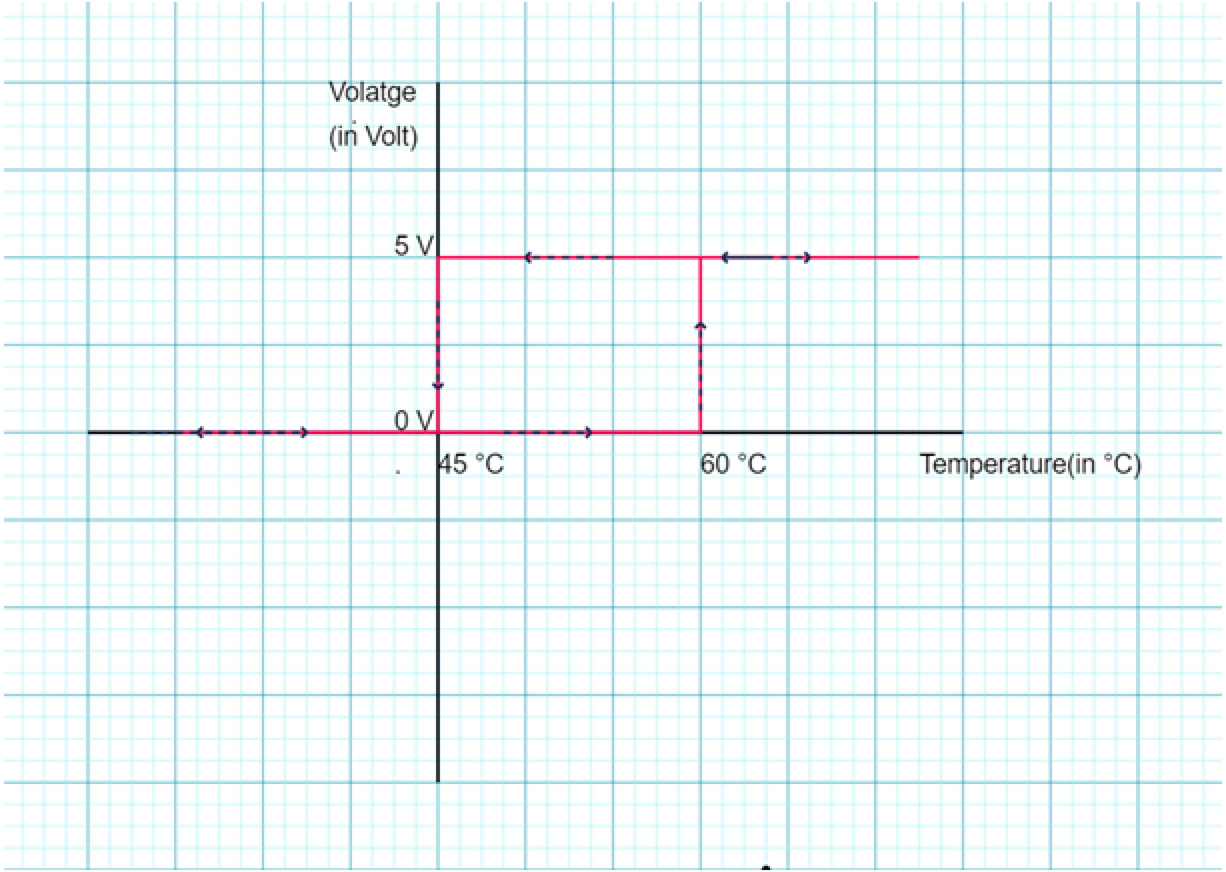
In = 0v, no power supply to peltier and fans

In = 5v, 12 volt power supply to peltier and fans

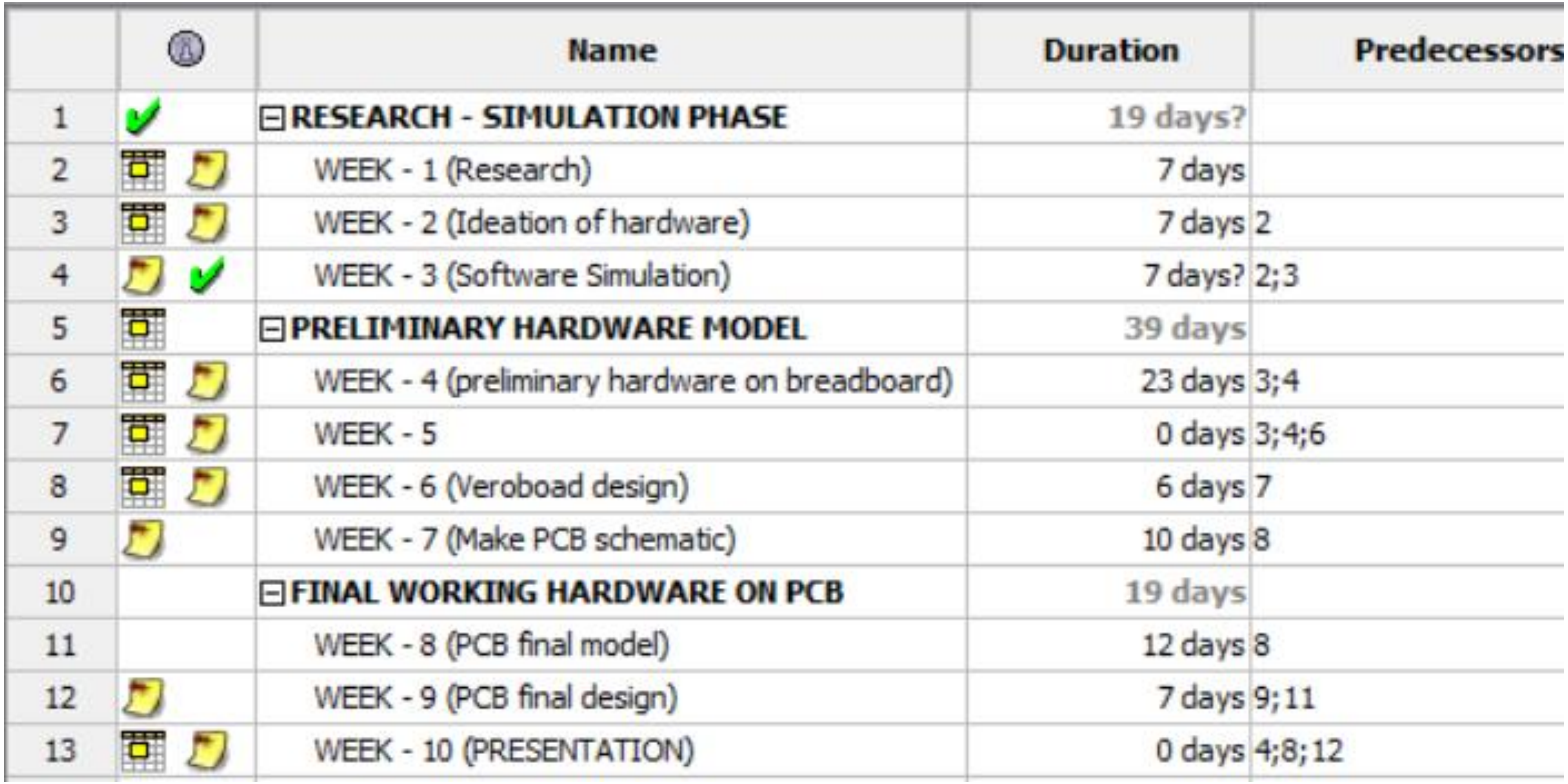
MAX6675

- ENOB(Effective Number Of Bits) = (SNR-1.76)/(6.02)
- Resolution of MAX6675 is 0.25°C
- Maximum error is ± 2°C
- Attach a capacitor between gnd and Vcc pin of MAX6675, to reduce noise
- Clock frequency is 4.3 Mhz
- Clock jitter increases noise effect at high frequency but in MAX6675 the frequency used is 4.3Mhz, ideally clock jitter effect is less in the range of few Khz to 10 Mhz.

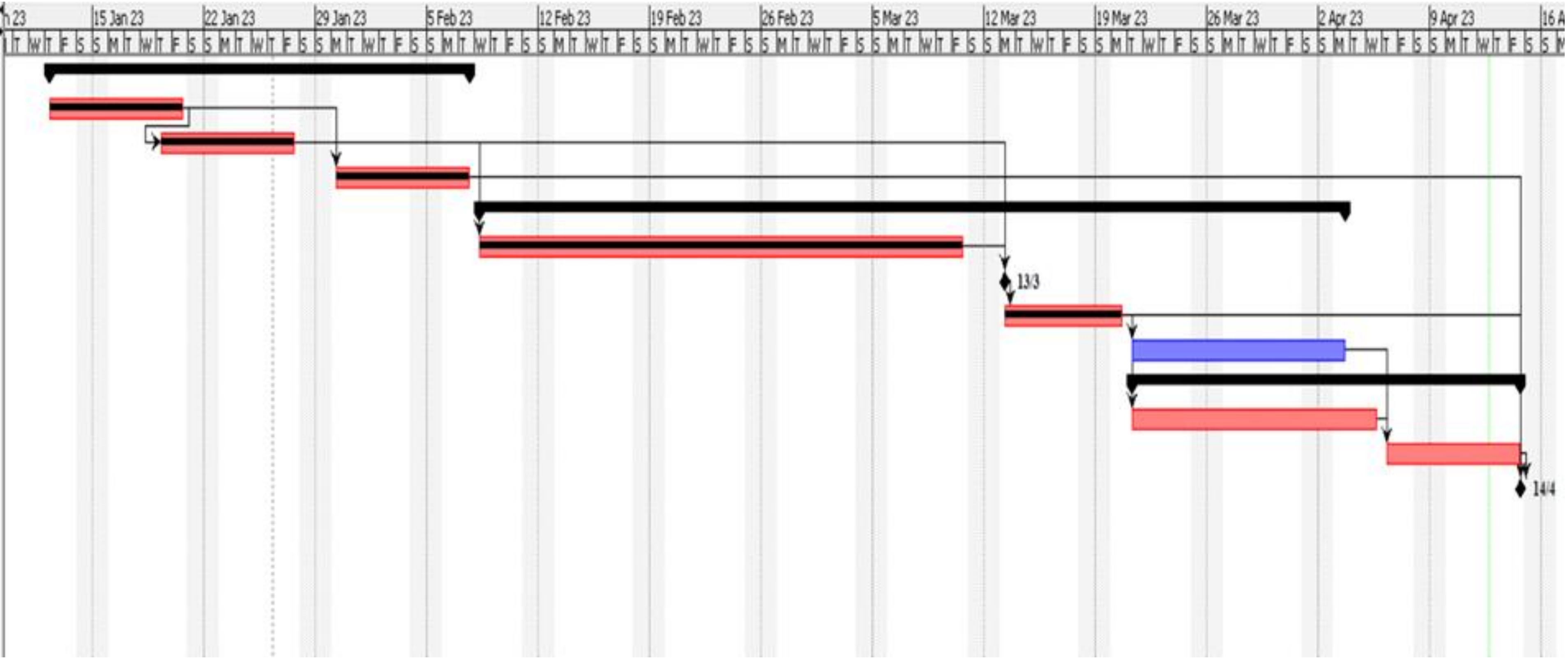
Peltier Input Voltage Vs Temperature Sense by K-Type Thermocouple



Weekly Milestone



	ⓘ	Name	Duration	Predecessors
1	✓	⊟ **RESEARCH - SIMULATION PHASE**	19 days?	
2	📅 📝	WEEK - 1 (Research)	7 days	
3	📅 📝	WEEK - 2 (Ideation of hardware)	7 days	2
4	📝 ✓	WEEK - 3 (Software Simulation)	7 days?	2;3
5	📅	⊟ **PRELIMINARY HARDWARE MODEL**	39 days	
6	📅 📝	WEEK - 4 (preliminary hardware on breadboard)	23 days	3;4
7	📅 📝	WEEK - 5	0 days	3;4;6
8	📅 📝	WEEK - 6 (Veroboad design)	6 days	7
9	📝	WEEK - 7 (Make PCB schematic)	10 days	8
10		⊟ **FINAL WORKING HARDWARE ON PCB**	19 days	
11		WEEK - 8 (PCB final model)	12 days	8
12	📝	WEEK - 9 (PCB final design)	7 days	9;11
13	📅 📝	WEEK - 10 (PRESENTATION)	0 days	4;8;12



Click here to get a projectlibre file

Risks Involved and Mitigation Plans

- **User Discomfort-**

  Risk : Noise, vibration, heat transfer to user

  Mitigation : Use noise-dampening materials, optimize the cooling system for low noise, use insulation to minimize heat transfer

- **Overheating of Peltier Module-**

  Risk : Can cause permanent damage

  Mitigation : Use thermal switch and heat sink

- **Arduino Compatibility-**

  Risk : Incompatible components, coding errors

  Mitigation : Test all components, use updated libraries and programming techniques

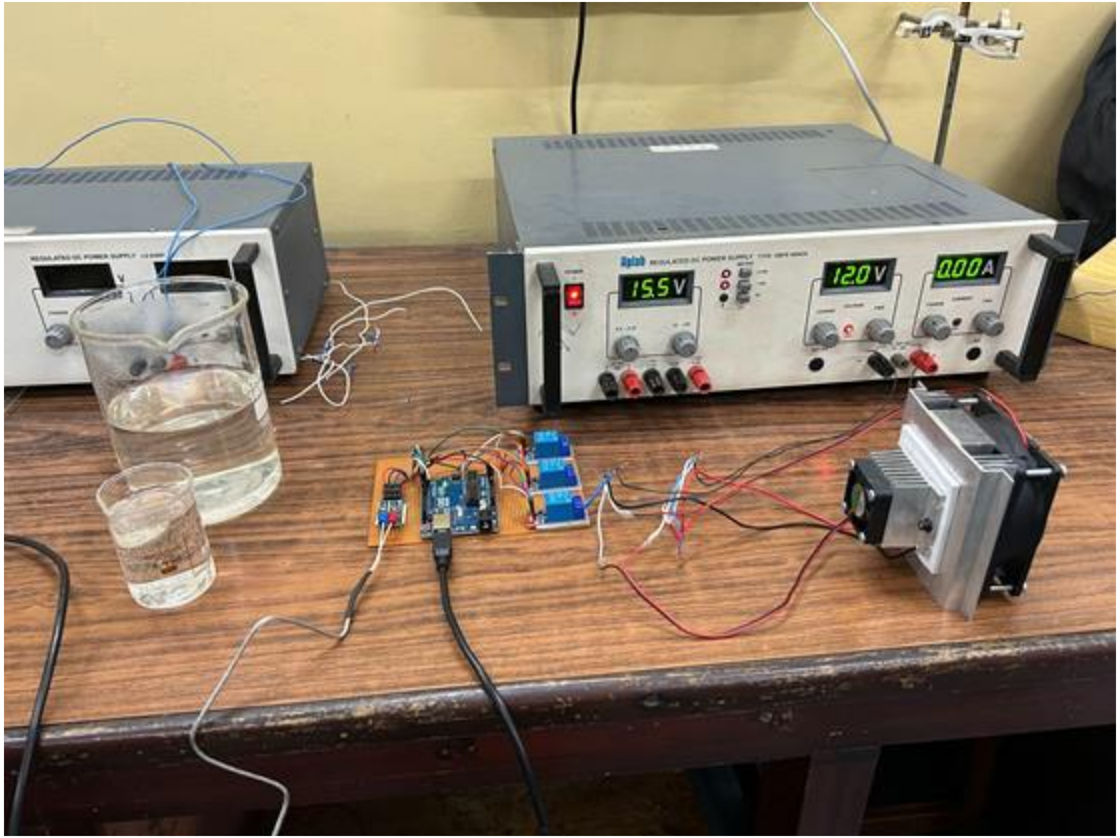
Learning Aspects

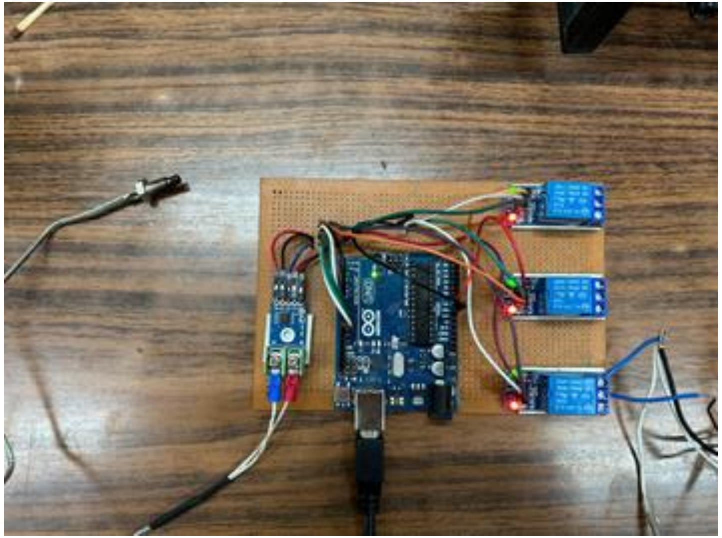
1) Integration of devices with different current and voltage driving capabilities (Peltier module with Arduino)

1) Analysing the startup behaviour, current/voltage and temperature specifications of any device before executing the circuit on hardware

1) Minimizing the cost of overall hardware without changing its output performance

Final Set-up





Shortcomings and its improvement

1) **Duty cycle of Peltier module**

   The percentage of time that it is on versus off, can affect its efficiency and lifespan, as hysteresis range is not wide

   One potential improvement is to increase hysteresis range using an advanced peltier module

1) **Limited temperature sensing accuracy**

   Needs to ensure that thermocouple is calibrated regularly and correctly

   Use of high quality temperature sensor such as PRT or Thermistor instead of K type thermocouple as accuracy , for PRT and thermistor within 0.1°C to 1°C where for K type thermocouple around ±2°C to ±5°C

Shortcomings and its improvement

3) **Not compatible with all types of CPUs (automotive, household, data-centre)**

Bring in flexibility in setting the temperature range which the peltier module controls so that the module can be used in different environments on different types of devices as automotive CPUs work under high range of temperature which might be too high for normal CPUs

Take user feedback for dynamic temperature control range

4) **Switching loss**

Need to reduce the number of switchings in case the environment is such that causes large number of heating up and cooling down cycles

Instead of switching we control the fan speed and peltier cooling such that an equilibrium state is reached and maintained from there on

Challenges

- To drive peltier with a voltage of 12V from 5/0V arduino digital signal
- Initially used motor driver to integrate arduino and peltier but the motor couldn't take up the high load
- Faced difficulty in debugging particular components with improper functioning
- Difficulty in understanding the current driving capability of motor driver with load
- Weren't able to perform the software analysis due to unavailability of peltier module in proteus software
- To ensure the hysteresis of module and fans as per our required temperature range

Thank You!