MEL 5450 - Embedded System Project

Use Of Peltier Devices for Battery Charging

Malith Weerasekara 2019AE41

Master of Science in Applied Electronics

Department of Physics, University of Colombo, Colombo 3

March 2022

Abstract

There are lots of wasted heat sources like sunlight or heat generated from some other process as a side effect. These energies can be harvested as electricity using thermoelectric generators. In this project use of Peltier devices for battery charging was investigated. A data recording device was constructed for recording temperature on both surfaces of the Peltier devices, voltage, current and power generated by the Peltier devices. Temperature differences vs generated voltage, current and power were observed. Multiple Peltier devices can be used to generate more power and Peltier devices can be connected in series or parallel connection. It was identified when there is no temperature difference between both surfaces or very small difference available the both ends of a Peltier device acts like short circuited. Because of this if the Peltier devices are connected in parallel one Peltier can short circuit another. For generating maximum power, its is better to connect Peltier devices in series was major finding in this project. After these investigations a Li battery (18650) charging device using 8 Peltier devices in series was able to constructed successfully. The device can harvest power from either sunlight or fire from a stove.

01.Introduction

When a current is passing the junction between two conductors, one side of the junction becomes hot while the other side becomes cool. This is known as the Peltier effect.[1] Devices based on the Peltier effect, known as Peltier coolers, are widely used for cooling CPUs in computers. The same devices can be used in the opposite direction i.e., to generate electricity when one side is maintained at a lower temperature compared to the other side.

These Peltier devices can be used to generate electricity using freely available heat sources such as sunlight or heat generated from some other processes as a side effect. The heat flows through the Peltier devices and generated electricity. Wasted heat sources can be used to generate electricity using these Peltier devices.

TEC1-12706 is the model of the Peltier device that selected for this project because of easily available in the market[03]. A device was constructed to record current, voltage and temperature difference to a SD card. A single Peltier device was connected to the data recording device and recorded data by applying heat difference to the Peltier device. Open circuit voltage, short circuit current, voltage and current for different loads (1kohm variable resister) was recorded by increasing the temperature difference (ΔT) and analyzed those data. The objective is to construct a device to charge 3.7V Li batteries (18650) for that one Peltier device was not generating enough power hence I have to use more than one Peltier devices. To find out the most efficient way to connect Peltier devices I connected two Peltier devices in series and another two Peltier devices in parallel and recorded same data set. By plotting those data, it was clear by connecting Peltier devices in series were more efficient than connecting them in parallel.

For this project 8 Peltier devices were connected in series. Two aluminum sheets were used to stick the Peltier modules. The heating surface of the Peltier devices were attached to one aluminum sheet and it was painted mat black to absorb more heat from the sunlight. 9 old computer heat sinks were attached to the other aluminum sheet which was attached to the other surface of the Peltier devices.

02.Design and Methodology

This project was carried out to explore the use of Peltier devices for battery charging. My intensions were to get freely available heat sources like sunlight or heat generated from some other processes as a side effect to generate electricity for the battery charging. There were two questions initially when the charging devices was constructed.

Q1. How many Peltier devices are needed for the Li battery (18650) charging setup?

Q2. How to connect these Peltier devices, series vs parallel?

To address above Q1 and Q2 experimental setup was designed.

02.1 Experimental Design

Setup I

In this setup a single Peltier device was used. Open circuit voltage, short circuit current and voltage and current for different loads (1 kΩ variable resister was used for the load) were measured by applying heat to one surface of the Peltier device to keep a temperature difference. Temperature readings ware also recorded.

Figure 1 - one Peltier setup

As shown in the figure 1 an aluminum sheet was used to attach the Peltier device, a candle was used to heat the aluminum sheet, an aluminum cup filled with water was used to keep the other surface cooler. Thermal paste was used to stick the Peltier device.

Setup II

Setup I experiment was performed again using two Peltier devices (parallel connection).

Figure 2 two Peltier parallel setup

As shown in the figure 2 two Peltier devices ware used to collect data after attaching Peltier devices to the setup another aluminum sheet was sticked to the upper surface of the Peltier devices and insulation sheet was also used in the middle.

Setup III

Setup II experiment was performed again using two Peltier devices in series connection.

02.2 Measuring and Recording Device

To perform above experiments a device was constructed to measure voltage, current[07] and temperatures[05] from two surfaces and generated power was displayed and record these data to a SD card[08] as a text file.

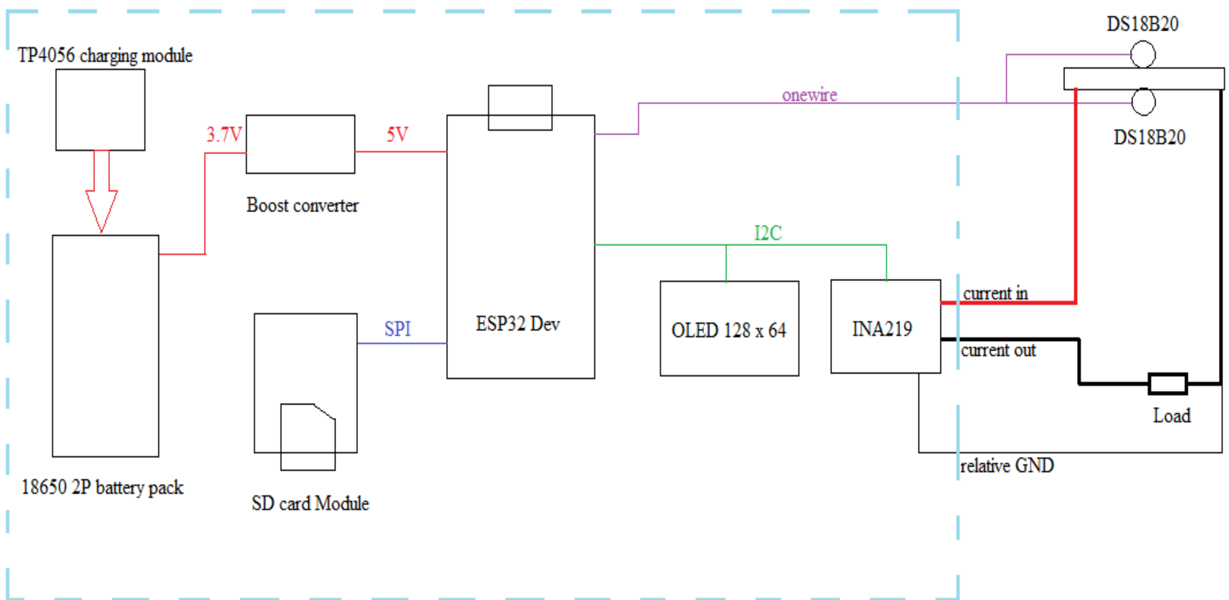


Figure 3 Data recording setup

Following components are used to build the data recording device.

ESP32 Dev module, I2C OLED display (128 x 64), INA219 current/voltage sensor, 2 x DS18B20 temperature sensor, boost converter, SD card Module, 2 x 18650 battery, TP4056 battery charging module

This device is powered using two parallel 18650 batteries. A battery charging module (TP4056) [09] was used to charge the batteries via micro-USB cable. A boost converter is used to boost the battery voltage to 5V to supply power for the esp32 Dev board and other modules. The two temperature sensors (DS18B20) are used to measure the top and the bottom surfaces of the Peltier device. INA219 module is used to measure current and the voltage across the load. Both temperature values, load voltage, current, power generated, and the record number were shown on the OLED display [06] and at the same time these data will be recorded to the SD card.

Figure 4 Measuring and recording device

02.3 Li Battery Charging device from sunlight.

After analyzing Setup, I, II, III results 8 Peltier devices were selected and connected them in series. Two aluminum sheets were used to attach Peltier devises. One aluminum sheet was painter mat black to absorb more heat from sun light and 9 computer heat sinks ware to attached to the other sheet to cool down the surfaces. A buck/boost converter is connected to the device get constant 5V supply from the setup (any device capable of charging from USB can connected to this setup). TP4056 module is used to charge the Li battery.

Figure 5 Battery charging device from sunlight

02.4 Li Battery Charging device from stove

02.3 Li battery charging device from sunlight was modified by removing aluminum sheet with heat sinks and attached aluminum tray. The heating side is facing fire from the stove. Device will generate electricity while boiling the water.

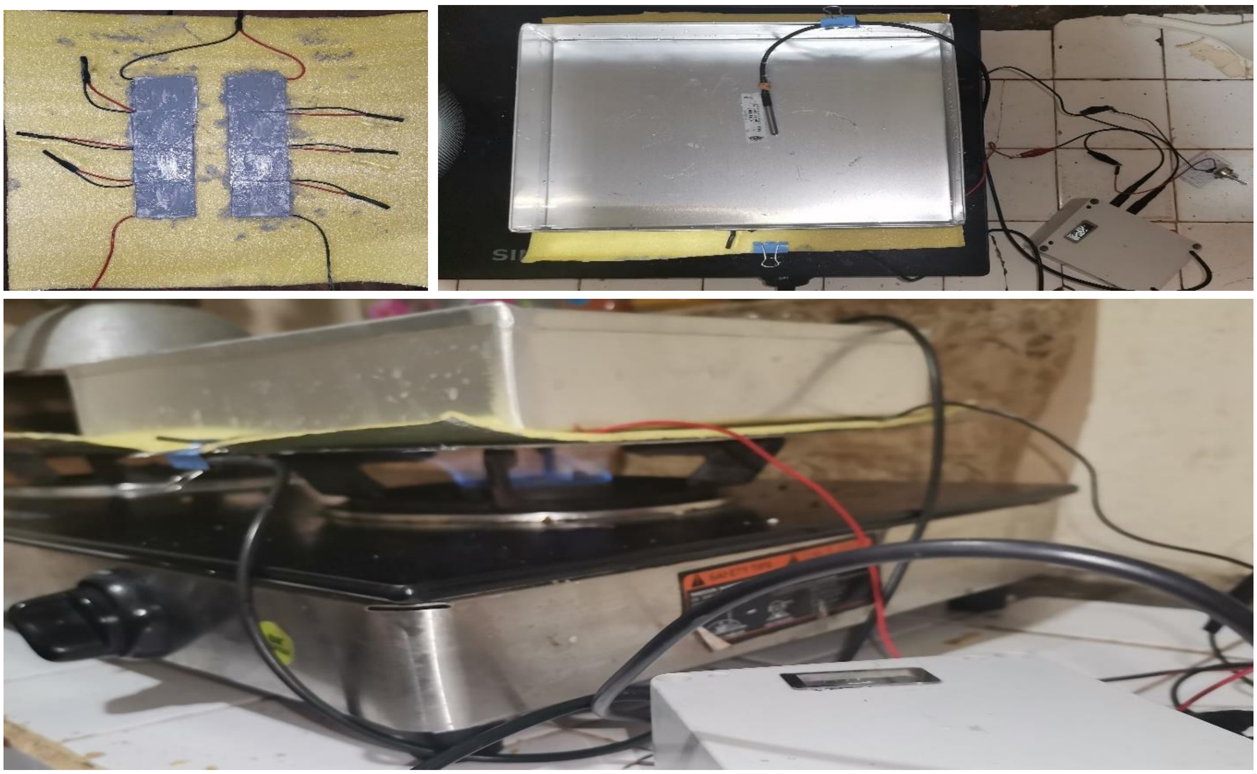


Figure 6 Battery charging device from gas stove

Results and Discussion

01. Sample data set recorded from the recording device

	A	B	C	D	E	F
1	Reading ID	temp1_C	temp2_C	voltage_V	current_mA	
2	1	30.88	30.88	0.36	-0.2	
3	2	30.94	30.81	0.36	0.2	
4	3	31.06	30.88	0.36	0	
5	4	31.12	30.88	0.37	0.1	
6	5	31.25	30.94	0.37	-0.2	
7	6	31.37	30.88	0.38	-0.2	

Figure 7 Sample data set

Data was recorded using the data recoding device that was built. Reading no, temperature values of the both side of the Peltier devices, voltage and the current were recorded to the SD card. Figure 7 shows the sample recorded data set.

02. Setup I

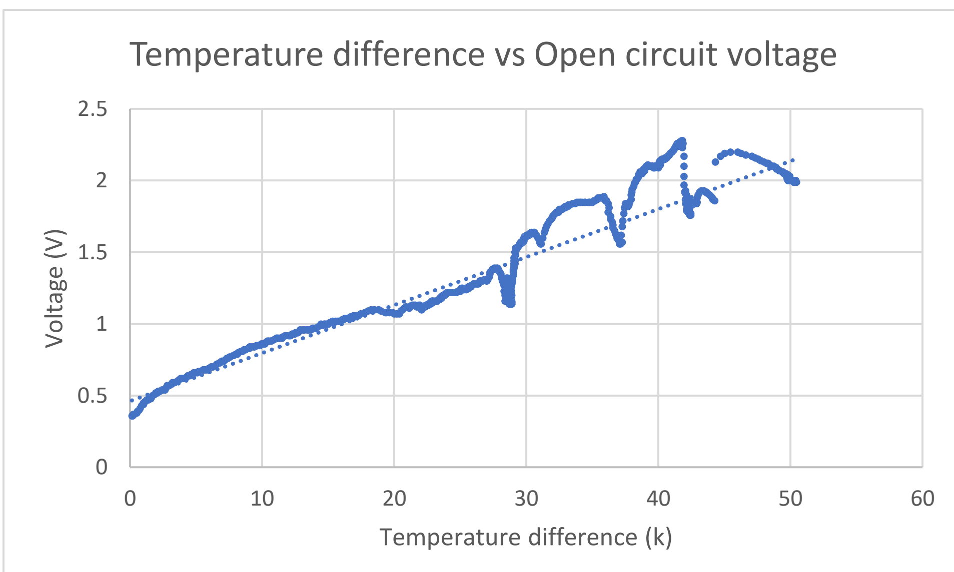


Figure 8 chart of the Temperature difference vs the open circuit voltage for a single Peltier

Figure 8 shows the how the open circuit voltage behaves against for the temperature differences(ΔT). It seems this voltage rises linearly when the ΔT increases until 27K. Higher voltages can be achieved with higher ΔT. In this project maximum ΔT was 50K and the maximum temperature was achieved on the hot side was 81.25 ^{O}C (354.4K). The maximum temperature that can be achieved without destroying the Peltier module is 138 ^{O}C [03].

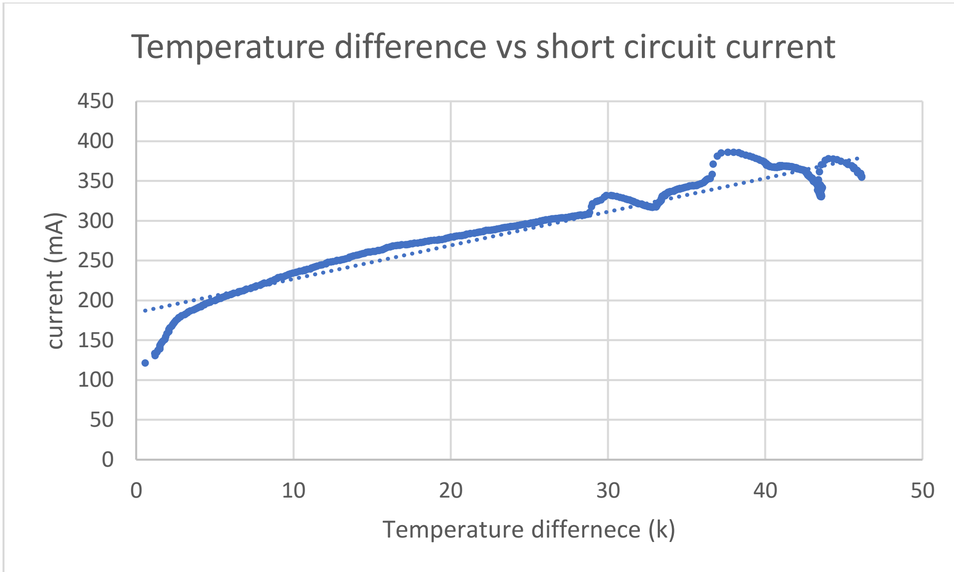


Figure 9 the chart of Temperature difference vs short circuit current for a single Peltier device.

Same as the figure 8 this chart (figure 9) it shows linear characteristics between ΔT 6K to 27K. To achieve higher currents higher ΔT needs. But maximum temperature on the hot side is 138 ^{O}C [03].

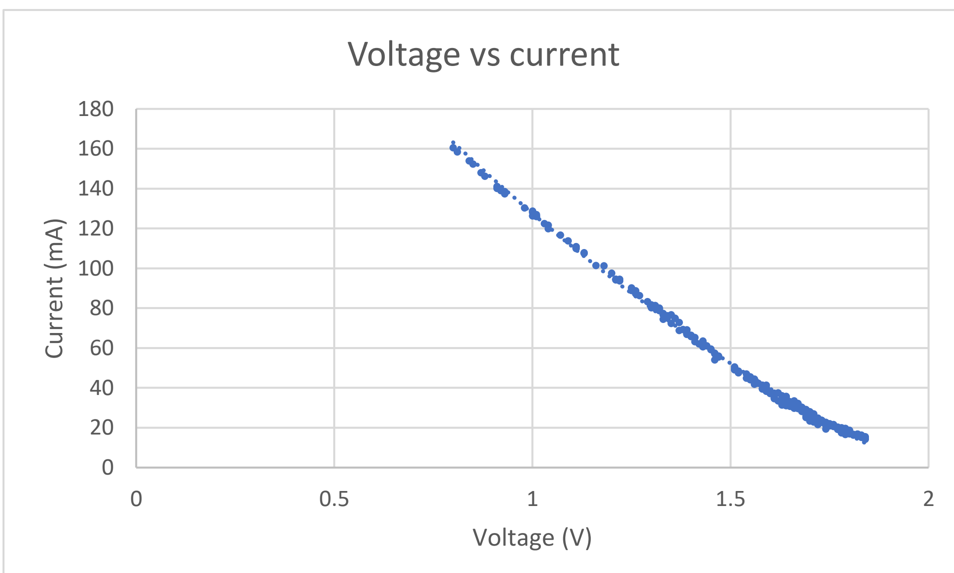


Figure 10 The chart of Voltage vs Current for a single Peltier device

By keeping ΔT at 35K and increasing the load resistance 0 to 200 Ohm voltage vs current for a single Peltier was plotted. When there is no load resistance 160 mA was obtained and when resistance increased gradually flowing current was decreases and voltage was increased from 0.8V to 1.8V. When the resistance reaches 120ohm the current and the voltage stayed at 18mA and 1.8V.

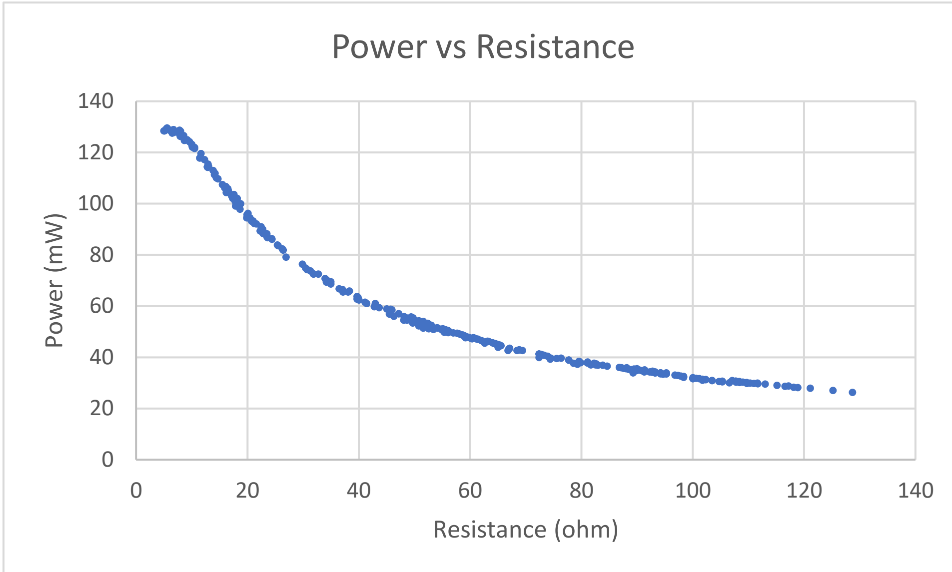


Figure 11 The chart of the Power vs Resistance

By keeping ΔT at 35K and increasing load resistance current and voltage readings obtained and Power and resistance data was calculated and plotted in the figure 11. With less load resistance maximum power can be generated.

03. Setup II and Setup III

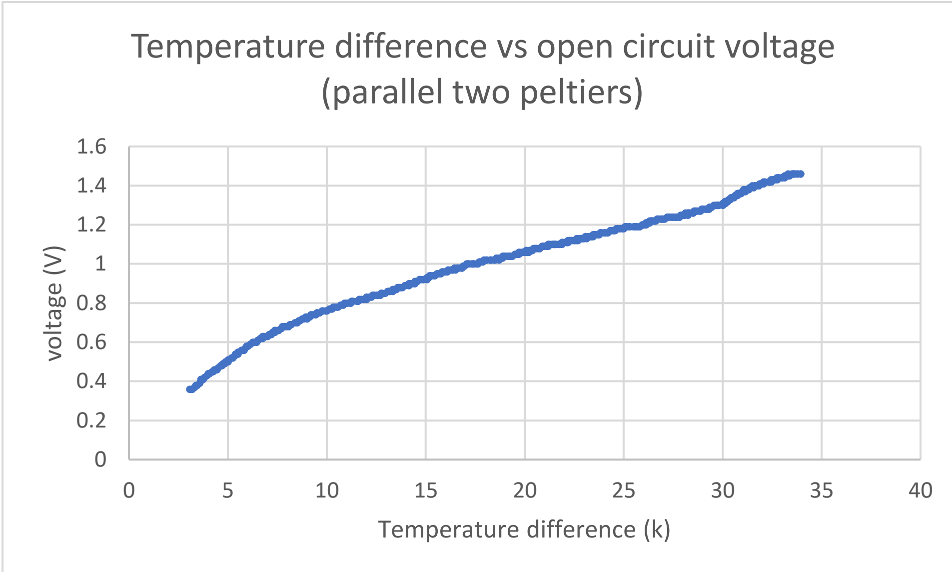


Figure 12 the Chart of Open circuit voltage two parallel Peltier devices.

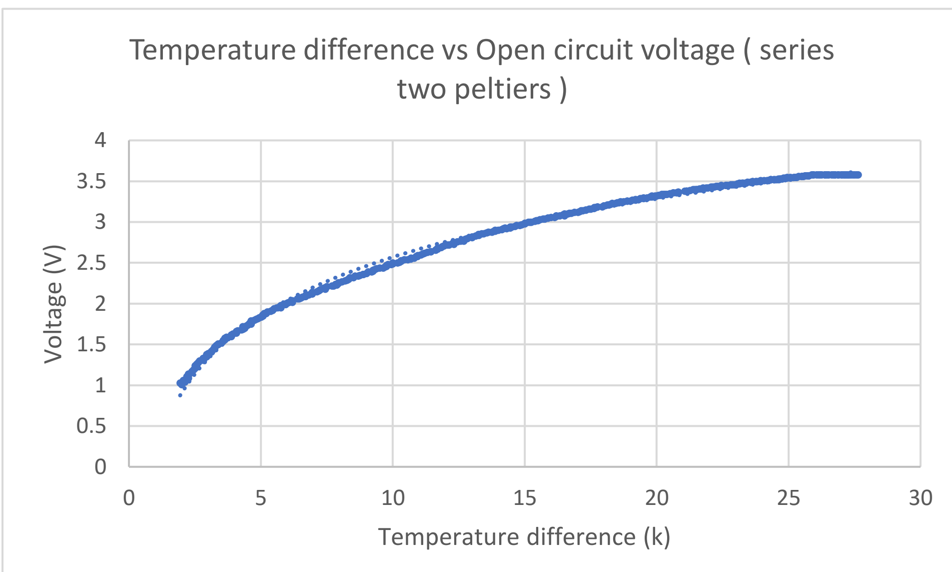


Figure 13 the Chart of Open circuit voltage two series Peltier devices.

Connecting Peltier devices in series can achieved higher voltage than connecting them in parallel. Figure 12 and 13 shows at ΔT 25 parallel connection only reaches 1.2V but series connection reaches 3.5V. this is happening because one Peltier module is short circuiting the other Peltier module.

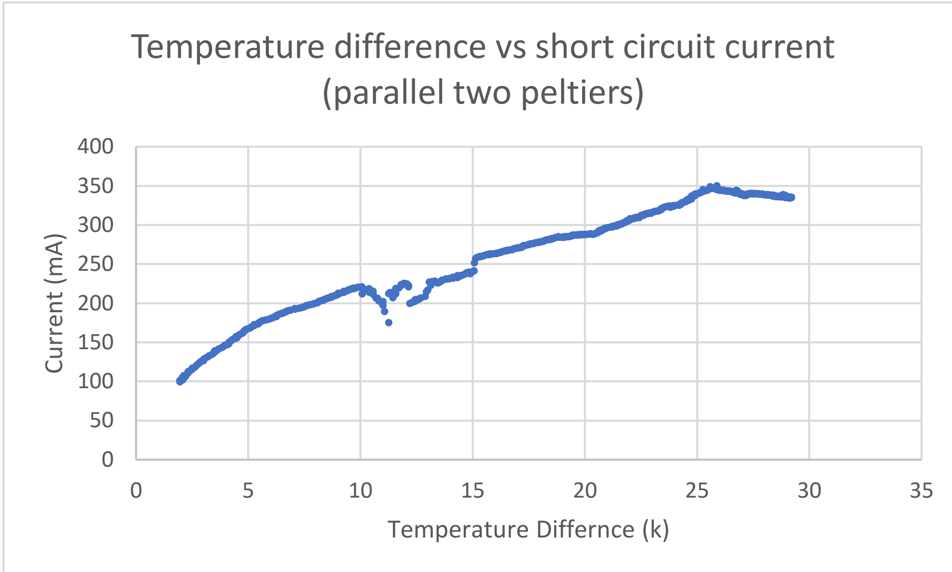


Figure 15 The chat of short circuit current vs Temperature difference for tow Peltier devices in parallel

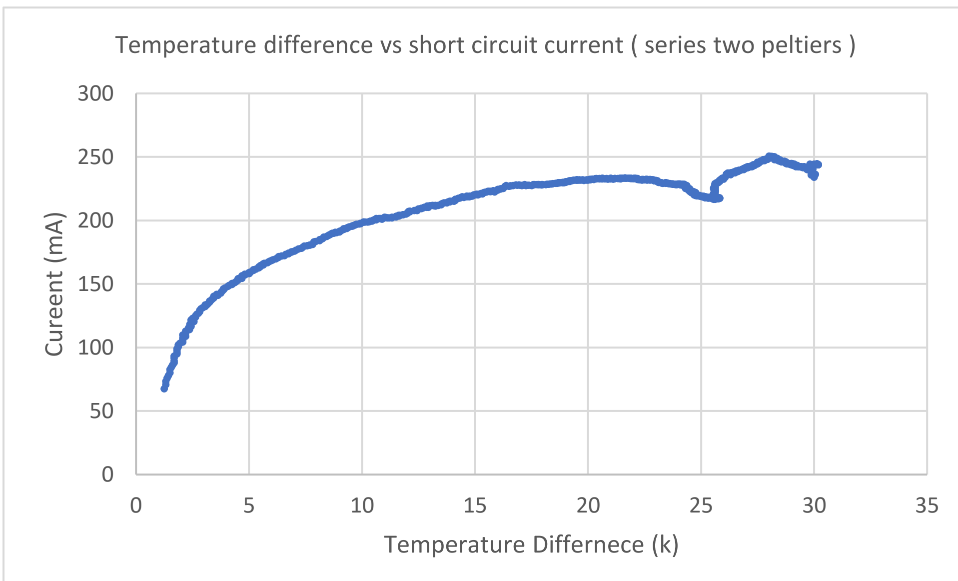


Figure 14 The chat of short circuit current vs Temperature difference for tow Peltier devices in parallel

By looking at figure 14 and 15 connecting Peltier devices parallel can achieves more current. ΔT at 25K parallel connection produces 340mA and series connection produces 220mA.

When it comes to the power generation using figure 12,13,14 and 15. Parallel connection produces 408mW and series connection produces 770W. With these data it is better to connect Peltier devices is series to generate more power.

04. Li Battery Charging Device from Sunlight (8 Peltier devices connected in series)

Maximum temperature reached hot side of the device 59 ^{O}C and other side of the device was at environment temperature (normally about 30 ^{O}C). Maximum temperature difference obtained 29K. Hot side transfers the heat to other side and generate electricity. Device need to be supplied with heat from sunlight and cooled from the other side in order to generate electricity. After few minutes it getting heated the cooled side of the device and temperature difference decreases. To keep the bottom side at constant temperature the heat sinks of the device were immersed in the water. From this way second side can be kept at constant temperature for some time (around 30 minutes). Then temperature of the water increases slowly. It’s better to change water for every 30 minutes or supplies with a constant water flow.

Maximum Volage generated 11.7V and maximum charging current was drawn initially 789 mA.

05. Li Battery Charging Device from Stove

The temperature must not be exceeded 138 ^{O}C other wise Peltier devices will be destroyed [03]. From this devices Li batteries can be charged and at the same time as a product water on the container will be heated.

Conclusion

Connecting Peltier devices in series better than connecting them in parallel.

When there is no temperature difference between surfaces Peltier device acts like short circuit.

If the Peltier devices connected in parallel one Peltier can short circuit another Peltier.

Maximum temperature achieved from sunlight 59 ^{O} C and its possible to have 25 K temperature difference using sunlight.

To fully charge an 18650 battery 4hours and 27 minutes was taken from the Li battery charging device using sunlight. (Initially battery was at 3.0V. after charging the battery voltage was 4.V. battery capacity was labeled at the battery was 1800mAh)

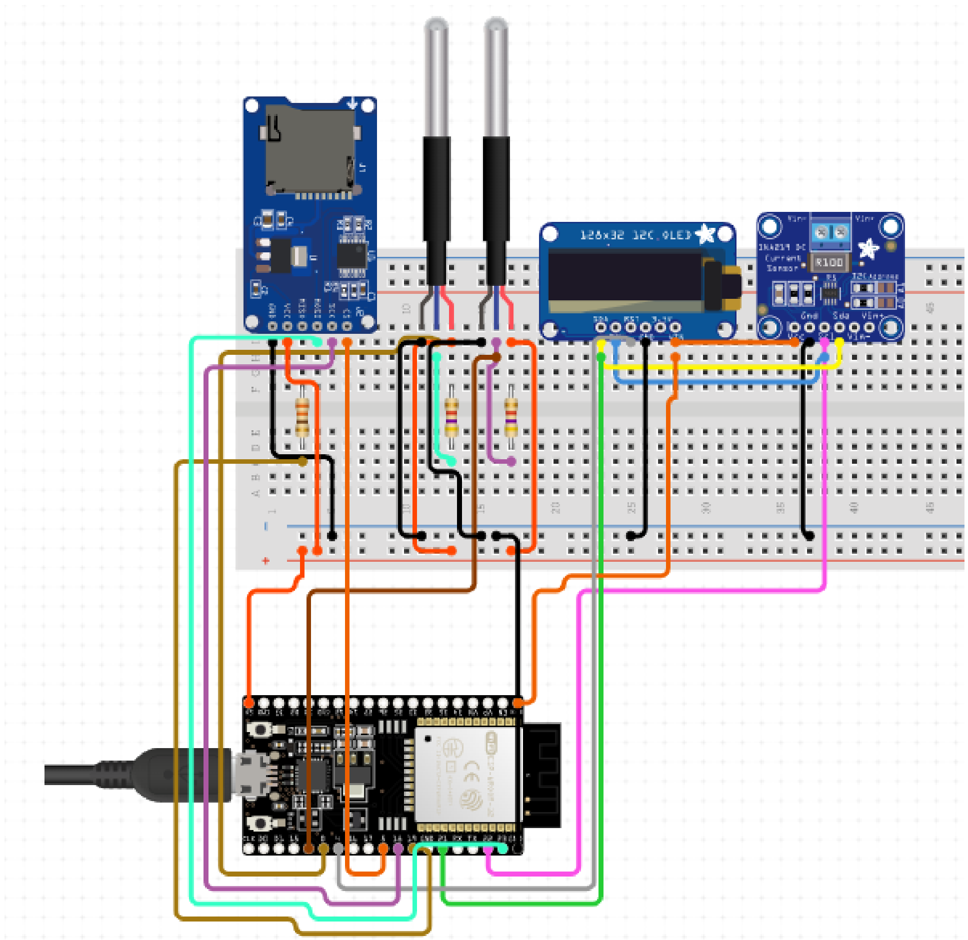
Peltier devices can be used successfully to harvest wasted heat energy into electricity.

References

01. The Peltier effect https://link.springer.com/article/10.1007/s10973-007-8336-9
02. Experimental study on the influence of Peltier effect on the output performance of thermoelectric generator and deviation of maximum power point https://www.sciencedirect.com/science/article/abs/pii/S0196890419310805
03. TEC1 12706 Peltier datasheet thttps://pdf1.alldatasheet.com/datasheet-pdf/view/227422/ETC2/TEC1-12706.html
04. Thermoelectric power generation: Peltier element versus thermoelectric generator https://ieeexplore.ieee.org/abstract/document/7793029
05. One wire DS18B20 connecting multiple temperature sensors https://randomnerdtutorials.com/esp32-ds18b20-temperature-arduino-ide/
06. OLED with esp32 https://randomnerdtutorials.com/micropython-oled-display-esp32-esp8266/
07. INA219 datasheet https://www.ti.com/product/INA219?utm_source=supplyframe&utm_medium=SEP&utm_campaign=not_alldatasheet&DCM=yes&dclid=CjkKEQjw_tWRBhCiv-HIw7XOxOMBEiQAIbdsu5_-iKNewP_pj-tdeLiYHCVlEYEXX2KO7q-UqLm25Kzw_wcB
08. SD card reader https://randomnerdtutorials.com/altimeter-datalogger-esp32-bmp388/
09. TP4056 datasheet https://html.alldatasheet.com/html-pdf/1133270/TPOWER/TP4056/1278/5/TP4056.html

Annexures

01. Circuit Diagram of the data recording device.



02. ESP32 CODE

```
#include <OneWire.h>

#include <DallasTemperature.h>

#include <Adafruit_GFX.h>

#include <Adafruit_SSD1306.h>

#include <Adafruit_INA219.h>

#include "FS.h"
```

```
#include "SD.h"

#include <SPI.h>



#define ONE_WIRE_BUS 4 // Data wire is plugged TO GPIO 4

#define SCREEN_WIDTH 128 // OLED display width, in pixels

#define SCREEN_HEIGHT 64 // OLED display height, in pixels

#define SD_CS 5 // Define CS pin for the SD card module


OneWire oneWire(ONE_WIRE_BUS);

DallasTemperature sensors(&oneWire);

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1); // Declaration for an SSD1306
display connected to I2C (SDA, SCL pins)

Adafruit_INA219 ina219;


String dataMessage;

int readingID = 0;

float temp1;

float temp2;


float shuntvoltage;

float busvoltage;

float current_mA;

float loadvoltage_V;

float power_mW;


void setup(){

  Serial.begin(115200); // start serial port

  sensors.begin();
```

```
  if (! ina219.begin()) {
    Serial.println("Failed to find INA219 chip");
    while (1) { delay(10); }}
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println(F("SSD1306 allocation failed"));
    for(;;);
  }
  delay(2000);
  display.clearDisplay();
  display.setTextColor(WHITE);

  SD_Card_Initialize();



}

void loop(){
  sensors.requestTemperatures(); // Send the command to get temperatures
  Serial.print("Temperature for device1: ");
  temp1 = sensors.getTempCByIndex(0);
  Serial.println(temp1);
  Serial.print("Temperature for device12: ");
  temp2 = sensors.getTempCByIndex(1);
  Serial.println(temp2);

  display.clearDisplay();// clear display

  // display temperature
  display.setTextSize(1);
```

```
display.setCursor(0,0);
display.print("Temp1: ");
display.setTextSize(1);
display.setCursor(0,10);
display.print(temp1);
display.print(" ");
display.print("`C");

display.setTextSize(1);
display.setCursor(64,0);
display.print("Temp2: ");
display.setTextSize(1);
display.setCursor(64,10);
display.print(temp2);
display.print(" ");
display.print("`C");



shuntvoltage = 0;
busvoltage = 0;
current_mA = 0;
loadvoltage_V = 0;
power_mW = 0;

shuntvoltage = ina219.getShuntVoltage_mV();
busvoltage = ina219.getBusVoltage_V();
current_mA = ina219.getCurrent_mA();
power_mW = ina219.getPower_mW();
loadvoltage_V = busvoltage + (shuntvoltage / 1000);
```

```
Serial.print("Load Voltage_V : ");
Serial.println(loadvoltage_V);
Serial.print("Current_mA : ");
Serial.println(current_mA);
Serial.print("Power_mW : ");
Serial.println(power_mW);

//display voltage current and power

display.setTextSize(1);
display.setCursor(0,20);
display.print("Load voltage :");
display.print(loadvoltage_V);
display.print(" V");



display.setCursor(0,30);
display.print("Current :");
display.print(current_mA);
display.print(" mA");



display.setCursor(0,40);
display.print("Power :");
display.print(power_mW);
display.print(" mW");
display.display();
```

```
  if(SD.begin(SD_CS)) {
  readingID++;
  SD_Card_Initialize();
  logSDCard();
  display.setCursor(0,50);
  display.print("Saved");
  display.setCursor(40,50);
  display.print("Rec_No :");
  display.print(readingID);
  display.display();
  }
  else{
  display.setCursor(0,50);
  display.print("Insert a SD card!");
  display.display();
  }
  delay(10);
}


void logSDCard() {  // Write the sensor readings on the SD card
  dataMessage = String(readingID) + "," + String(temp1) + "," + String(temp2) + "," +
String(loadvoltage_V) + "," + String(current_mA)+ "," + String(power_mW) + "\r\n";
  Serial.print("Save data: ");
  Serial.println(dataMessage);
  appendFile(SD, "/data.txt", dataMessage.c_str());
}
```

```
void writeFile(fs::FS &fs, const char * path, const char * message) {  // Write to the SD card (DON'T MODIFY THIS FUNCTION)
  Serial.printf("Writing file: %s\n", path);


  File file = fs.open(path, FILE_WRITE);
  if(!file) {
    Serial.println("Failed to open file for writing");
    return;
  }
  if(file.print(message)) {
    Serial.println("File written");
  } else {
    Serial.println("Write failed");
  }
  file.close();
}



void appendFile(fs::FS &fs, const char * path, const char * message) {   // Append data to the SD card (DON'T MODIFY THIS FUNCTION)
  Serial.printf("Appending to file: %s\n", path);


  File file = fs.open(path, FILE_APPEND);
  if(!file) {
    Serial.println("Failed to open file for appending");
    return;
  }
  if(file.print(message)) {
    Serial.println("Message appended");
```

```
 } else {
   Serial.println("Append failed");
 }
 file.close();
}

void SD_Card_Initialize(){
 SD.begin(SD_CS);
 if(!SD.begin(SD_CS)) {
   Serial.println("Card Mount Failed");
   return;
 }
 uint8_t cardType = SD.cardType();
 if(cardType == CARD_NONE) {
   Serial.println("No SD card attached");
   return;
 }
 Serial.println("Initializing SD card...");
 if (!SD.begin(SD_CS)) {
   Serial.println("ERROR - SD card initialization failed!");
   return;    // init failed
 }

 // If the data.txt file doesn't exist
 // Create a file on the SD card and write the data labels
 File file = SD.open("/data.txt");
 if(!file) {
   Serial.println("File doens't exist");
   Serial.println("Creating file...");
```

```
    writeFile(SD, "/data.txt", "Reading ID, temp1_C, temp2_C, voltage_V, current_mA, power_mW \r\n");
  }
  else {
    Serial.println("File already exists");
  }
  file.close();

}
```