VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

Huy Hoang e1800961

Giao Phung e1800920

DIGITAL SYSTEM DESIGN

Embedded System

VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES
Digital System Design

ABSTRACT

Author	Huy Hoang and Giao Phung
Title	Digital System Design
Year	2020
Language	English
Pages	30 + 3 Appendices
Name of Supervisor	Ahvonen, Jani

Digitization was a wide range of applications, containing information, telecommunications, control systems etc. More than that, Industry required a reproducibility of information, flexibility and functionality: easier to store, transmit and manipulate information Which one was a cheaper device and easier to design.
This experiment focused on main topic was Analog to digital converter gone through some point:

- Calculation value and explanation for some important component.
- Using LTSpice XVII software to draw and simulate circuit
- Making board for testing and comparing circuit
- Using PADs layout and PADs logic for designing PCB board
- Soldering and Testing PCB

The goal was learned how to draw and simulate a digital design from basic to high level specifications. Understanding calculated result presented to compare with the reality circuit and simulation. Finally, made a PCB board from front to end for the design.

VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES
Digital System Design

CONTENTS

ABTRACTS

LI ST OF FIGURES AND TABLES

1 THE RESPONSIBILITIES OF THE STUDENT

After this experiment each student was understand and made a design an analog digital converter PCB with comprehend of its application. In order to do that, student checked these objects was show below.

- Schematic for Analog digital converter
- Testing by circuit
- Design PCB board
- Loading code with using Arduino to convert electric signal to temperature signal

2 PROCESSING

An Analog Digital converter was done after parts following:

- Analog Signal Processing
- Communication
- Analog to digital conversion
- PCB design

2.1 Analog Signal Processing

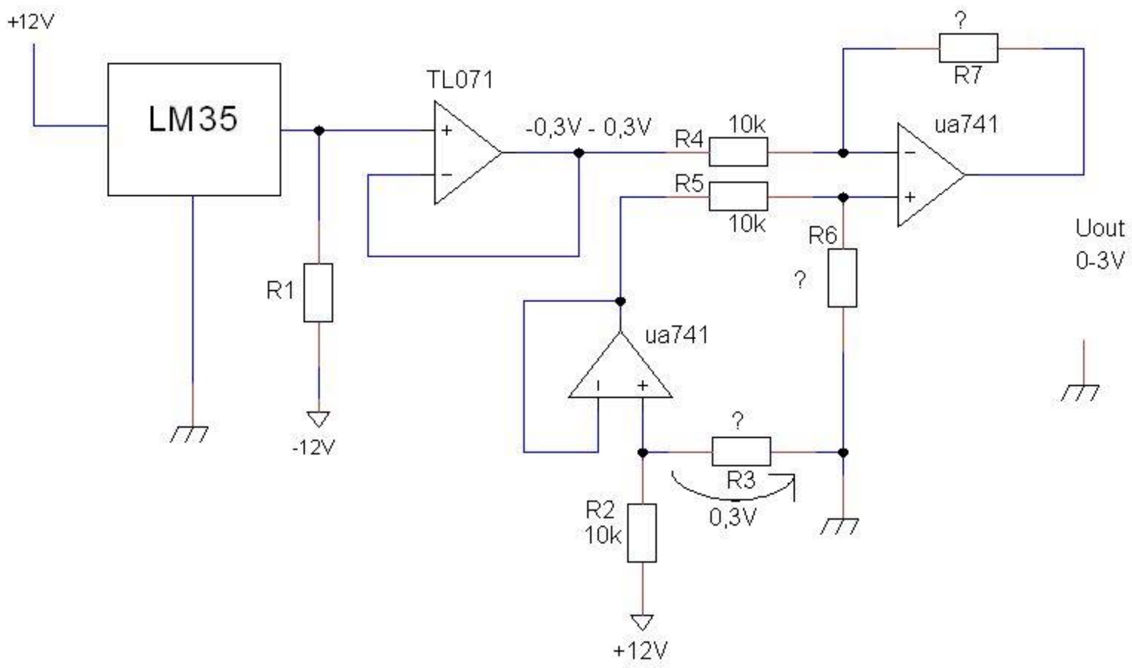


Figure 1: Analog Signal Processing

In this circuit, we designed temperature sensor circuit, buffer circuit and differential amplifier circuit. Some component values calculated. The differential amplifier used to gain input voltage (-0.3V to 0.3V) to wanted level (0V-3V) as in figure 1.

2.1.1 LM35

LM35 was used a temperature sensor, showing values in the form of output voltages instead of degree Celsius. The output voltage of LM35 is proportional to the Celsius temperature. The scale factor is .01 V/°C. LM35 had 3 pins showed in Table 1

Table 1: LM35 pins

No.	Parameter	Pin Type
1.	Vcc	Power Pin (Connected to +5V)
2	Vout	Output Pin (It should be connected with an analog pin of Microcontroller)
3	Ground	Ground Pin (Connected to 0V or GND)

However, LM35 is sensitive and difficulty to control the temperature. We first used direct power supply to TL071 to control the input voltage first then changed to LM35 and compare output result.

2.1.2 Calculation R3 and R6 and R7

$$UR2 = (U - UR3) = 13 - 0.3 = 11.7V$$

$$I = \frac{UR2}{R2} = \frac{11.7\ V}{10.000\ \Omega} = 1.17\ mA$$

$$R3 = \frac{UR3}{I} = \frac{300mV}{1.17mA} = 256\Omega$$

$$Uout = \frac{-R7}{R4}(U1a - U1b)$$

⇨ R7= $\frac{Uout}{(U1a-U1b)}R4 = \frac{3V}{-0.6V}10K\Omega = 50K\Omega$

Because the differential amplifier (ua741) voltage follower so R6=R7=50KΩ.

2.1.3 Schematic, Simulation and Analog Processing Circuit

When finished calculation component we started to draw the circuit using LTspice then simulated to compare the result.

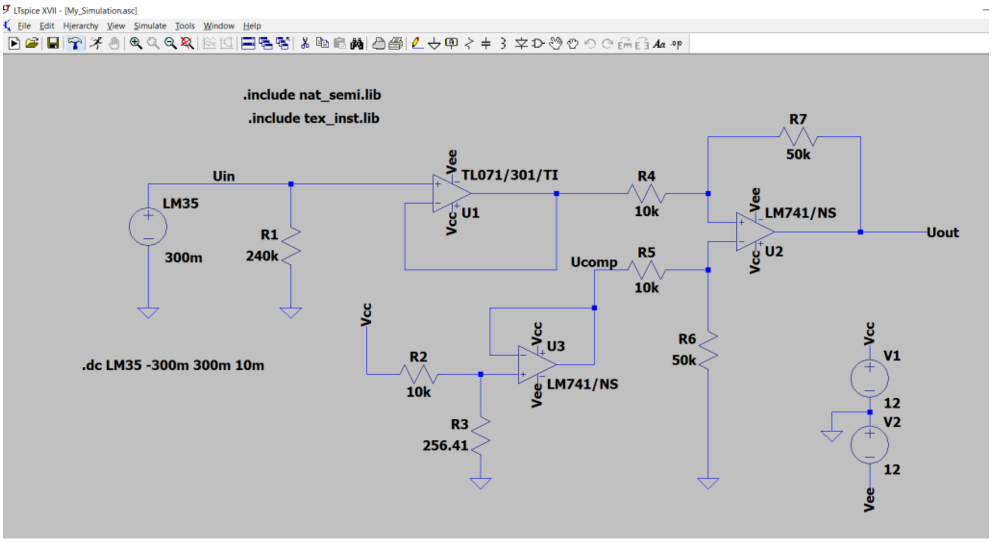


Figure 2: Analog Processing Schematic

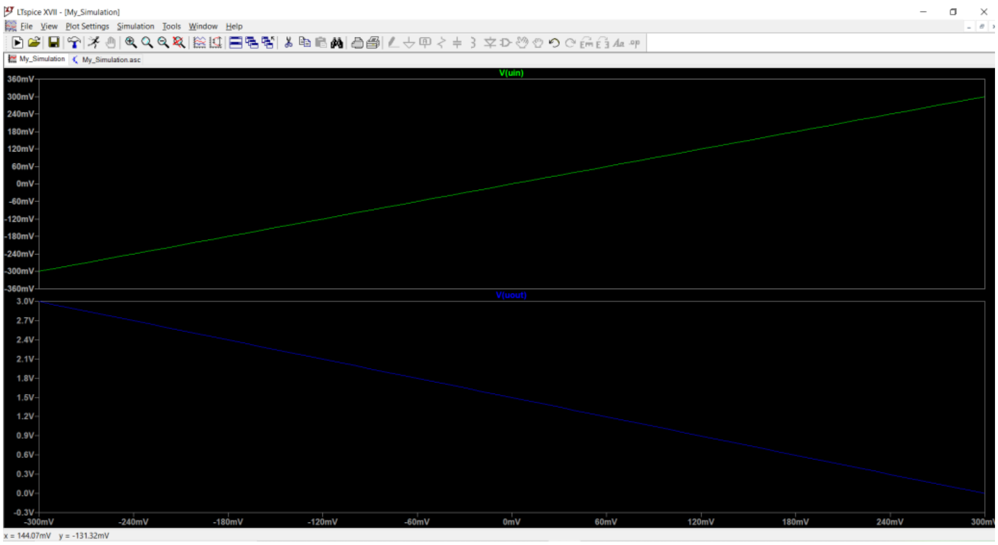


Figure 3: Analog Processing Simulate

As the result in Figure 3 showed that circuit generate output voltage from 0-3V when input voltage from -300mV to 300mV. To prove the simulate result clearly, the circuit was made

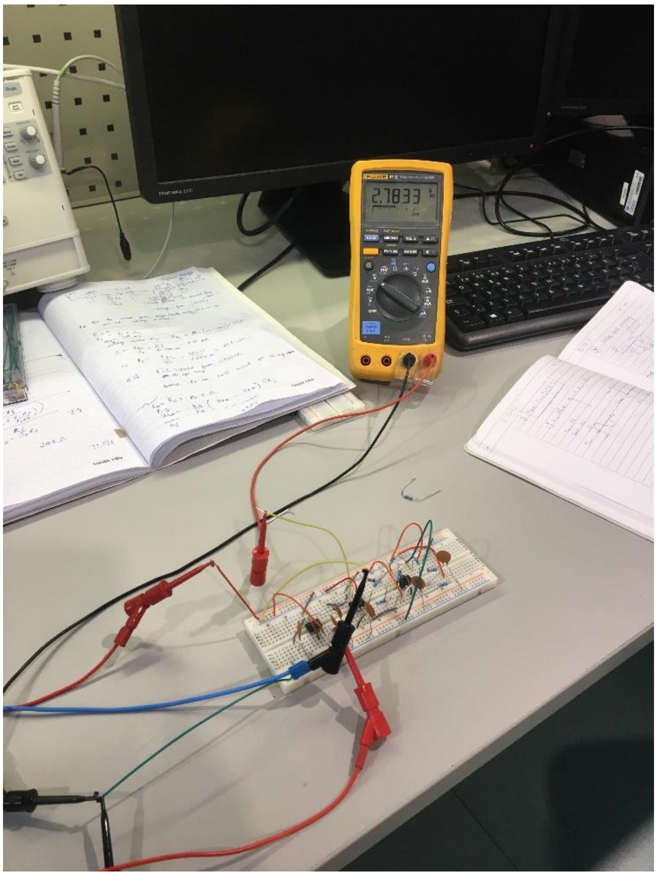


Figure 4: Analog Processing Circuit

Table 2: Analog Processing Table Result

U(LM35)(mV)	Uout	Temperature
-284,3	2,91	-30
-240	2,7	
-207	2,55	
-147	2,24	
-106,7	2,04	
-47,5	1,74	
8,8	1,46	0
42	1,29	
98,1	1,02	
147,2	0,77	
203,1	0,48	

241,1	0,29	24
295	0,02	

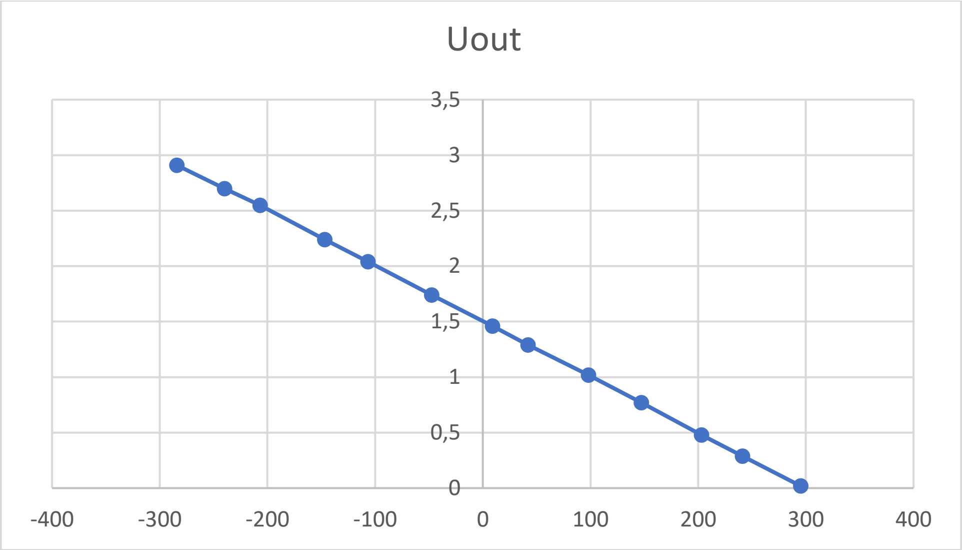


Figure 5 Analog Processing Output Voltage Chart

As the result showed in Table 2 and Figure 5, the circuit receipted signal voltage from power supply from -300mV to 300mV generated linearly to output signal from 0- 3V.

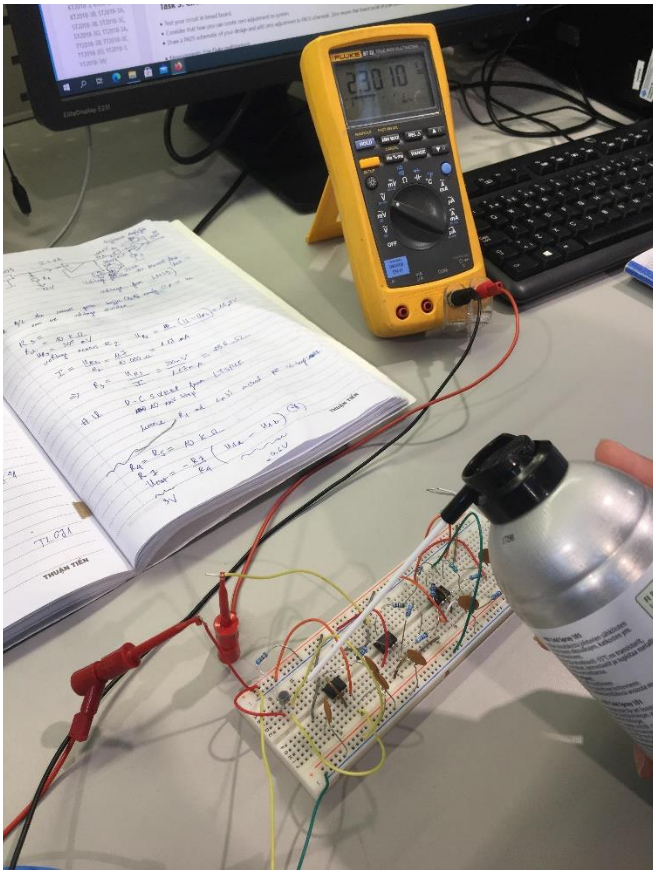


Figure 6: Analog Processing with LM35 circuit

When we used LM35 in Figure 6 The output voltage was corresponding with temperature from -30 degree to 30 degree.

2.2 Communication

2.2.1 UA741 and BC547

UA741 was an amplifier. Famous as an ideal amplifier for voltage follower applications with great stability even though without using the external components. It was nicely to short circuit protection, latch up free operation, no frequency compensation, offset voltage nulling capability, large common mode voltages.

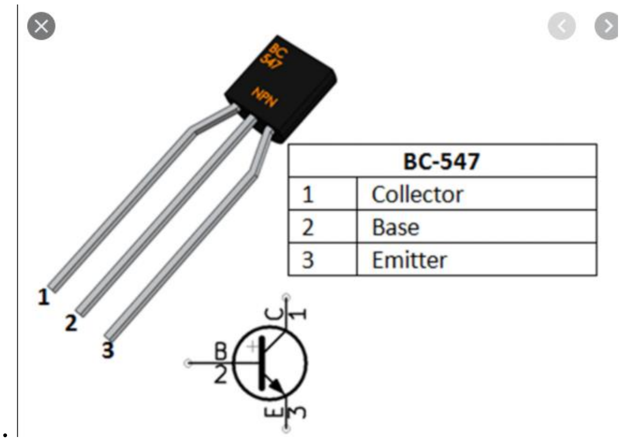
Figure 7: UA741

UA741 had eight (8) pins with different function showed in Figure 8

UA741 Pins	
Pin. No	**Pin Name**
1	Offset Null 1
2	IN-
3	IN+
4	GND
5	NC
6	V_{CC}
7	OUT
8	Offset N2

Figure 8: UA741 pin function

BC547 was an NPN Bipolar Junction Transistor. Used for the switching purpose or for amplification purposes. Smaller current at the base used to control the larger currents at collector and emitter Figure 9



BC-547	
1	Collector
2	Base
3	Emitter

Figure 9: BC547

2.2.2 Transmitter

From DC power supply created output voltage from sensor (LM35), which is connected to uA741 operational amplifier (+) input pin. So that, a 0-20 mA current generated passed through R1 when input voltage varies 0-3 V DC. When the current 20 mA flowed in circuit the resistor R2 must have 5V over it.

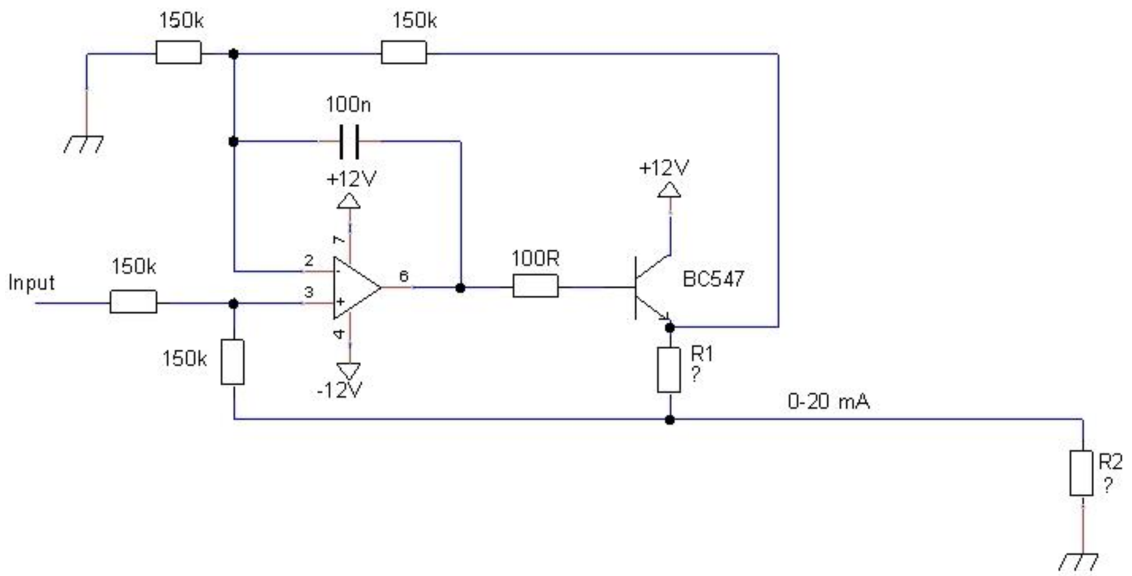


Figure 10: Transmitter

2.2.3 Calculation R1, R2

The input voltage from the previous circuit (Analog processing) was 0-3V gone through transmitter circuit at UA741 voltage follower has the same value is 3V. Therefore, UR1=Uin=3V.

$$R1 = \frac{UR1}{I} = \frac{3V}{20 * 10^{-3}A} = 150\Omega$$

$$R2 = \frac{UR2}{I} = \frac{5V}{20 * 10^{-3}A} = 250\Omega$$

2.2.4 Schematic, Simulation of Transmitter

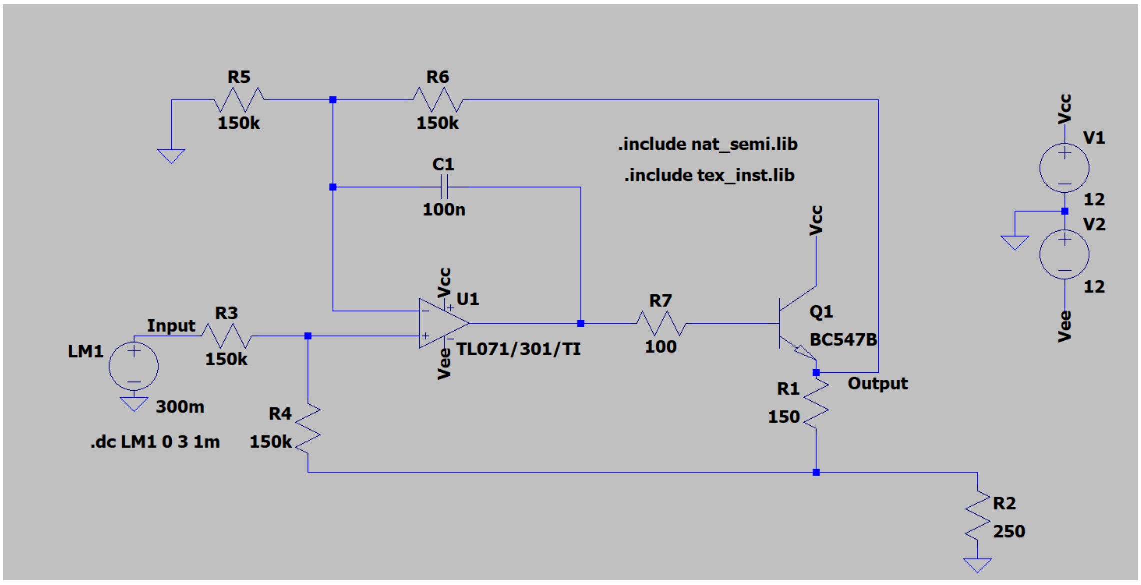


Figure 11: Schematic of Transmitter

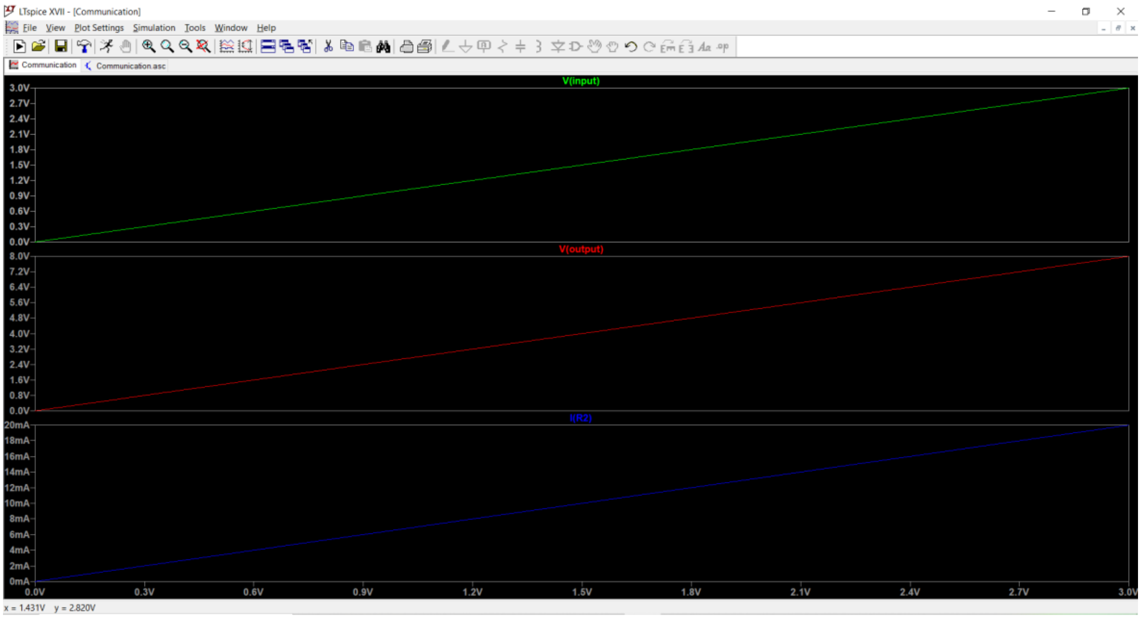


Figure 12: Simulation of Transmitter

Schematic and simulation resulted that after received 0-3V voltage from Analog processing transmitter circuit generated output voltage from 0 to over 5V, and current through R2 was 0-20mA

Table 3: Transmitter Current Output Table

Uin(V)	I out (mA)
3,41	22,46
3,074	20,16
2,853	18,81
2,62	17,14
2,379	15,68
2,138	14,1
1,925	12,78
1,556	10,36
1,355	9,05
1,033	6,88
0,908	6,11
0,613	4,08
0,368	2,46

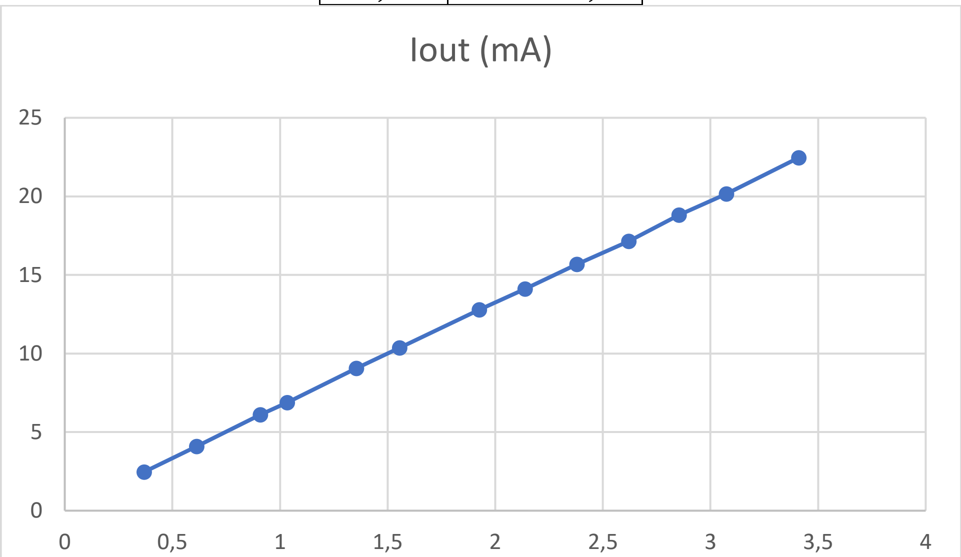


Figure 13: Transmitter Current Output Chart

2.2.5 Wire Resistance Between Transmitter and Receiver

The question was how long the wire can make between transmitter and receiver. To answer the we calculated the value for R(wire)

$$Rtotal = \frac{U}{I} = \frac{12V}{0.02A} = 600\Omega$$

$$Rwire = Rtotal - R1 - R2 = 600 - 250 - 150 = 200\Omega$$

$$R = \frac{\delta * l}{A} => l = \frac{R * A}{\delta} = \frac{200\Omega * 0.2mm^2}{0.017\frac{\Omega}{m}mm^2} = 2340m$$

The maximum wire length between transmitter and receiver was 2340m

2.2.6 Receiver

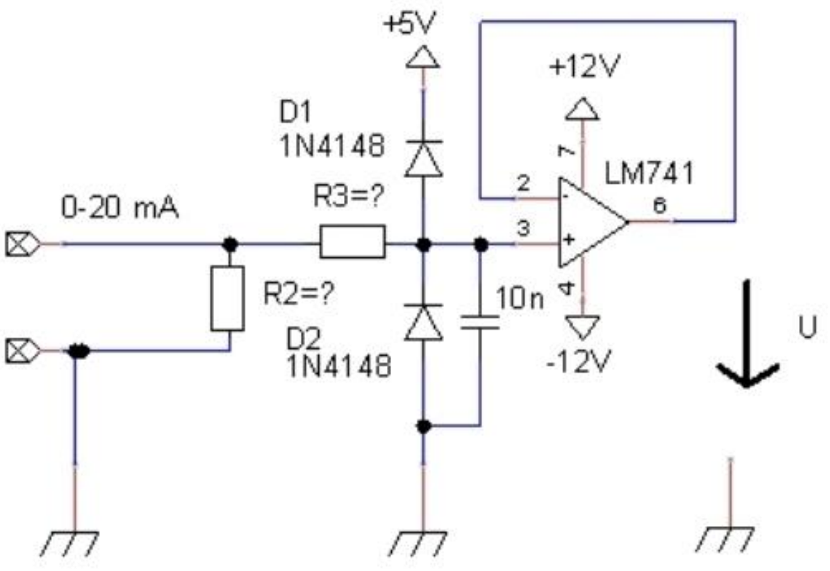


Figure 14:Receiver

Previous transmitter circuit gives 0 to 20 mA current and this current (signal) changed to voltage (signal) for ADC (analog to digital converter). This can be done by using receiver circuit [mA / V] (current to voltage transformer).

2.2.7 Calculation R2 and R3

R2 was the same value with R2 in transmitter R2=250Ω.

In case of input signal contains 8 kV ESD discharge voltage. Diode 1 and Diode 2 protected the circuit D1=D2=0.8A. It just accepted the low frequency gone to LM741 (the low pass filter cut-off frequency).

$$R3 = \frac{Uin}{I} = \frac{8kV}{0.8A} = 10K\Omega$$

2.2.8 Schematic, Simulation of Receiver

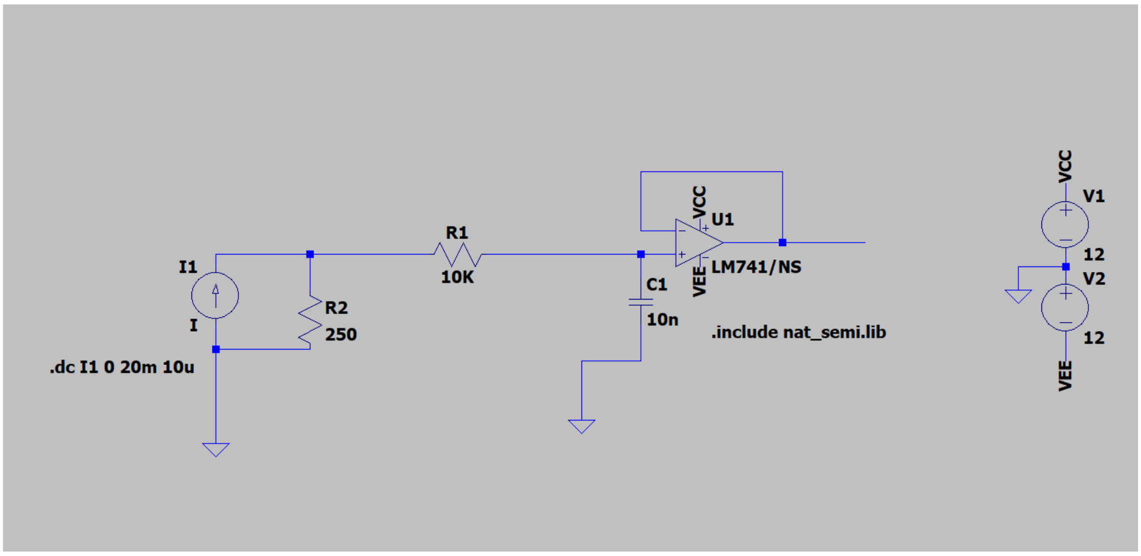


Figure 15: Schematic Receiver

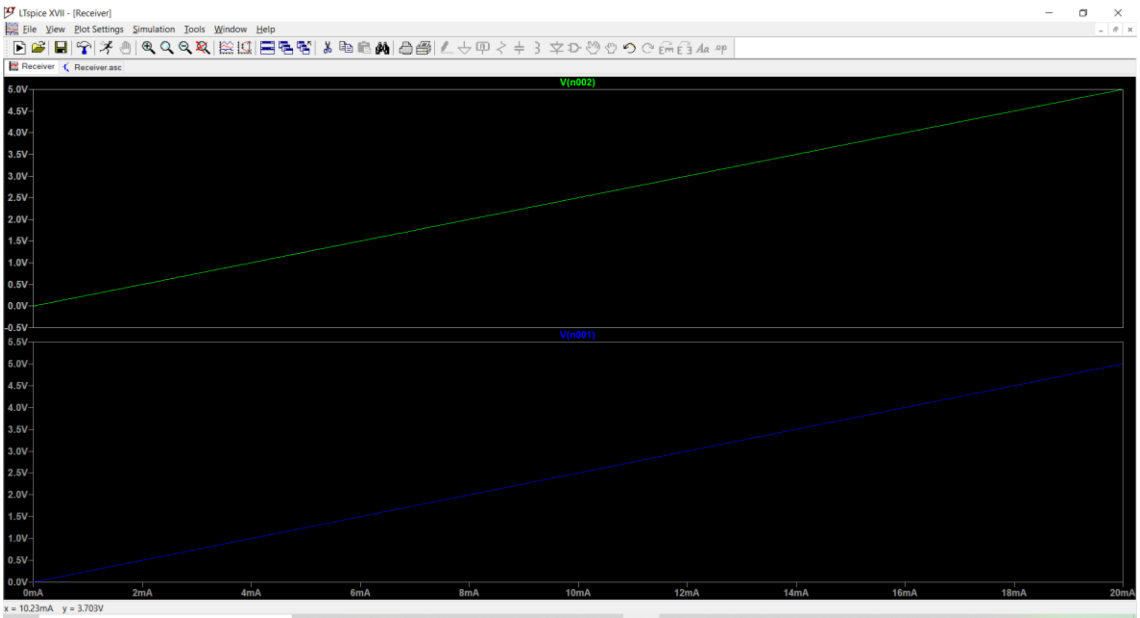


Figure 16:Simulation Receiver

In simulation, we clearly saw that when current from transmitter (0-20mA) went to receiver low pass filter generated voltage value from 0-5V.

Table 4:Output Voltage of Receiver

I(0-20mA)	Vout
0	0
5,1	1,275
10,18	2,548
15,01	3,76
20,27	5,0836

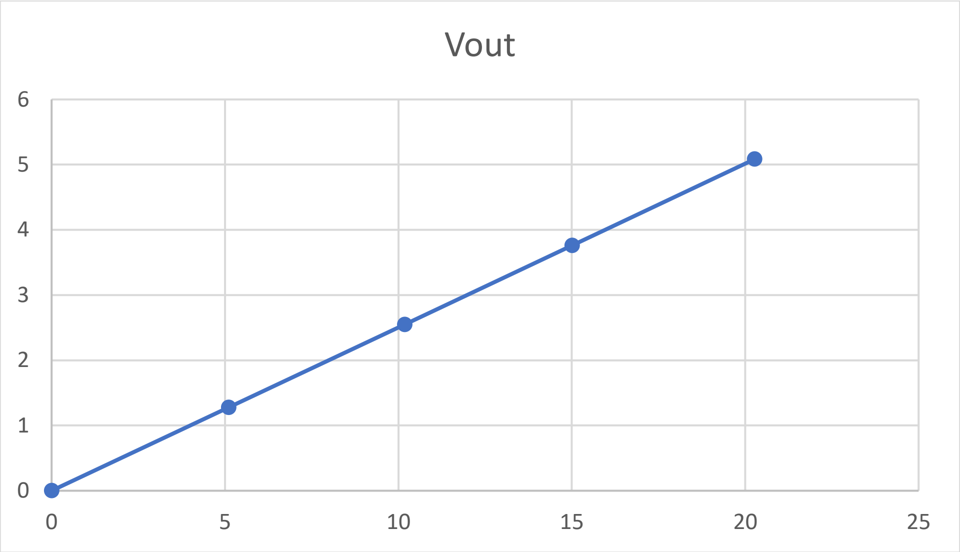


Figure 17: Receiver Output Voltage Chart

2.2.9 Communication circuit

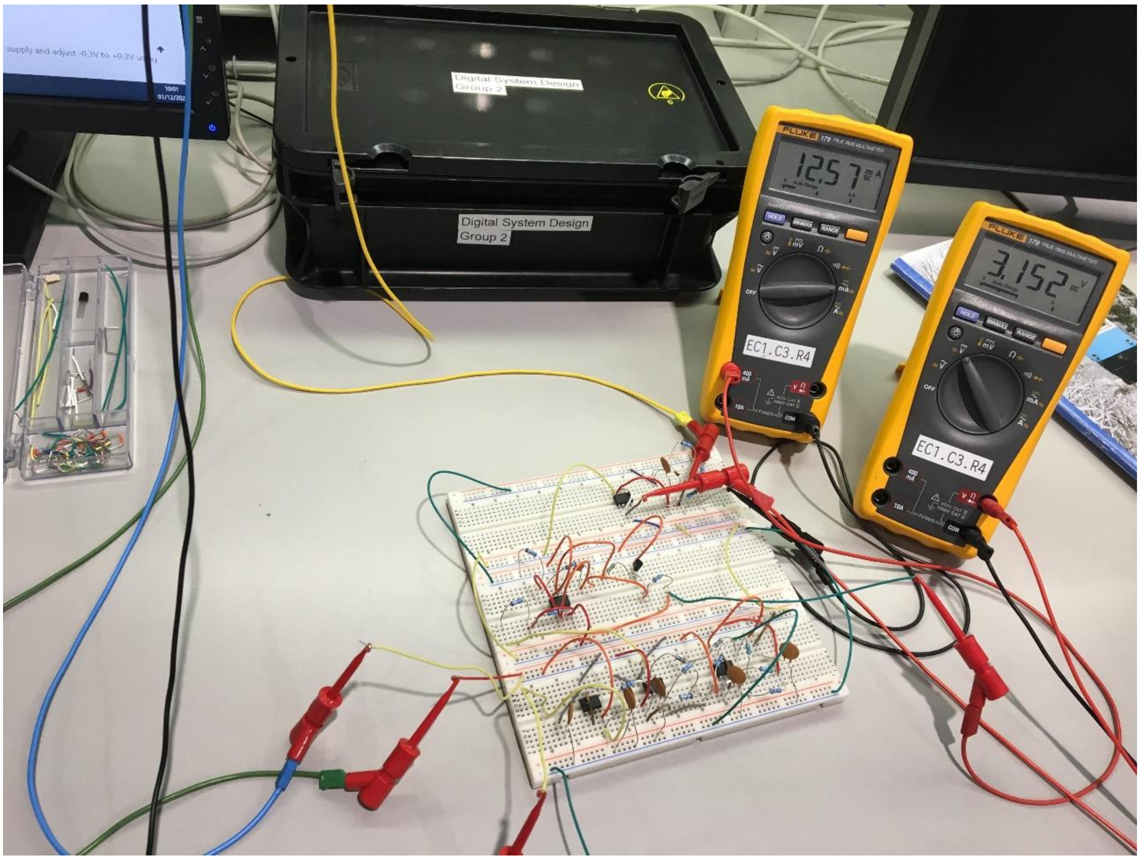


Figure 18: Communication circuit

2.3 Analog to Digital Conversion

2.3.1 MCP 3002 and ADC0832 comparison

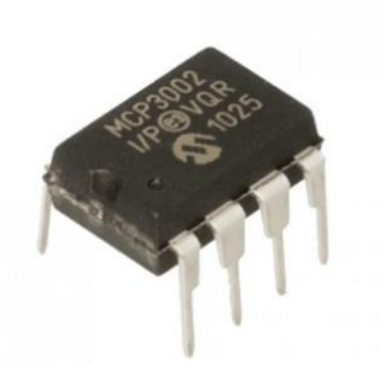


Figure 19:MCP3002

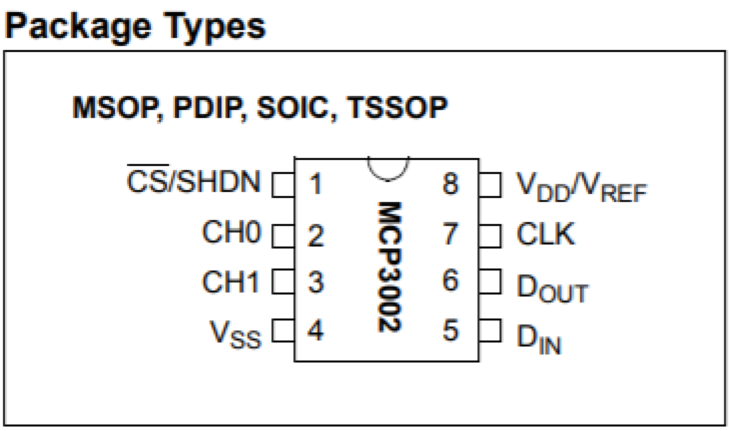


Figure 20: MCP3002 pin function

MCP3002	ADC0832
10 bits analog to digital converter 8-pin MSOP, PDIP, SOIC and TSSOP packages Single supply operation: 2.7V - 5.5V Clock Frequency 1.2MHz-3.2MHz	8 bits analog to digital converter 20 pin Single 5V supply Clock Frequency 0.1MHz-3MHz

2.3.2 Measure Temperature With Arduino

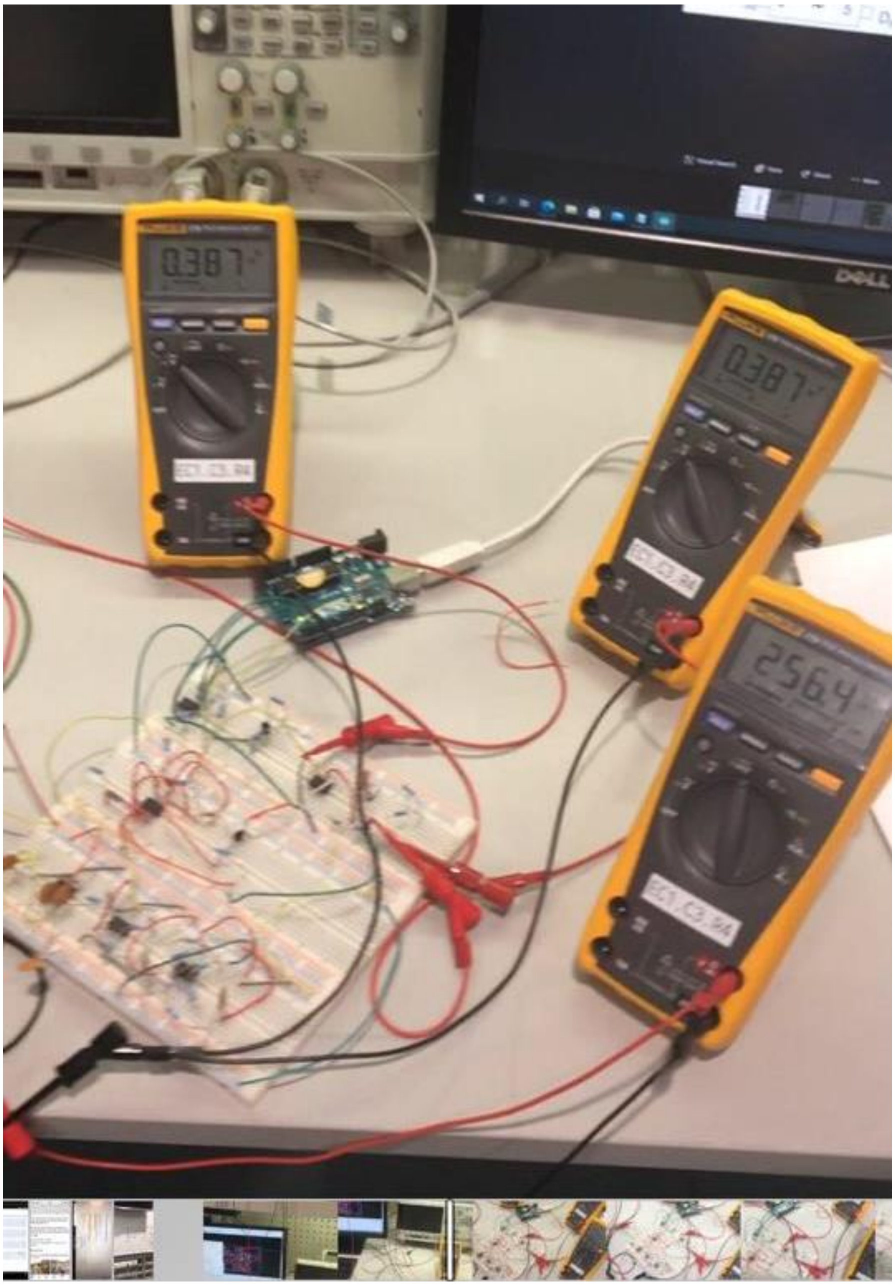


Figure 21: Analog To Digital Conversion Measuring Temperature With Arduino

Temperature information comes from receiver circuit. The receiver circuit output is 0V to 5V. We used reference voltage from reference chip LM431

When the voltage power supplier in the first circuit (Analog digital processing) from -300mV to 300mV the receiver circuit generated a 0-5V output voltage to Chanel 1 of MCP3002 then send voltage signal to Arduino.

Software testing code

```
void setup()

{

Serial.begin(9600);



pinMode(2, OUTPUT);

pinMode(3, OUTPUT);

pinMode(4, OUTPUT);

pinMode(5, INPUT);

pinMode(13, OUTPUT); //Arduino UNO pin 13 extra LED for ADC event



#define CS_HIGH digitalWrite(2, HIGH)

#define CS_LOW digitalWrite(2, LOW)



#define CLK_HIGH digitalWrite(3, HIGH)

#define CLK_LOW digitalWrite(3, LOW)
```

```
#define DIN_HIGH digitalWrite(4, HIGH)

#define DIN_LOW digitalWrite(4, LOW)



#define LED_ON digitalWrite(13, HIGH) //Arduino UNO pin 13 extra LED for ADC event

#define LED_OFF digitalWrite(13, LOW)



#define DOUT digitalRead(5)

}



void loop()

{

// CLK_HIGH;

// CS_HIGH;

// delay(50);



int ADC_result;

  float temperature;

  ADC_result=get_AD_value();

  temperature=-(60.0/1023.0)*ADC_result+30.0;

  Serial.print("ADCC value ");
```

```
  Serial.print(ADC_result);

  Serial.println();

  Serial.print("Temperature ");

  Serial.print(temperature);

  Serial.println();

  ADC_result=0;

  temperature=0;

  LED_OFF;

  delay(50);

}

int get_AD_value()

  {

 int ADC_result;

 ADC_result=0;

  int mask=512;

  char pulse=0;

  char i=0;

  char k=0;

  CLK_LOW;

  CS_LOW;
```

```
 LED_ON; //extra LED on

 delay(10);

 DIN_HIGH;

 delay(10);



 for(i=0;i<5;i++)

  {

  CLK_HIGH;

  delay(10);

  CLK_LOW;



  if(pulse==3)

   {

   DIN_LOW;

   }



  delay(10);

  pulse++;

  }
```

```
for(k=0;k<10;k++)

  {

  CLK_HIGH;







  if(DOUT==1)

    {

    ADC_result=(ADC_result|mask);

    }



  mask=mask>>1;





  delay(10);

  CLK_LOW;

  delay(10);

  }



CS_HIGH;
```

```
  delay(20);



  return ADC_result;

  }
```

The signals will send to Arduino board investigate ADC value to temperature value

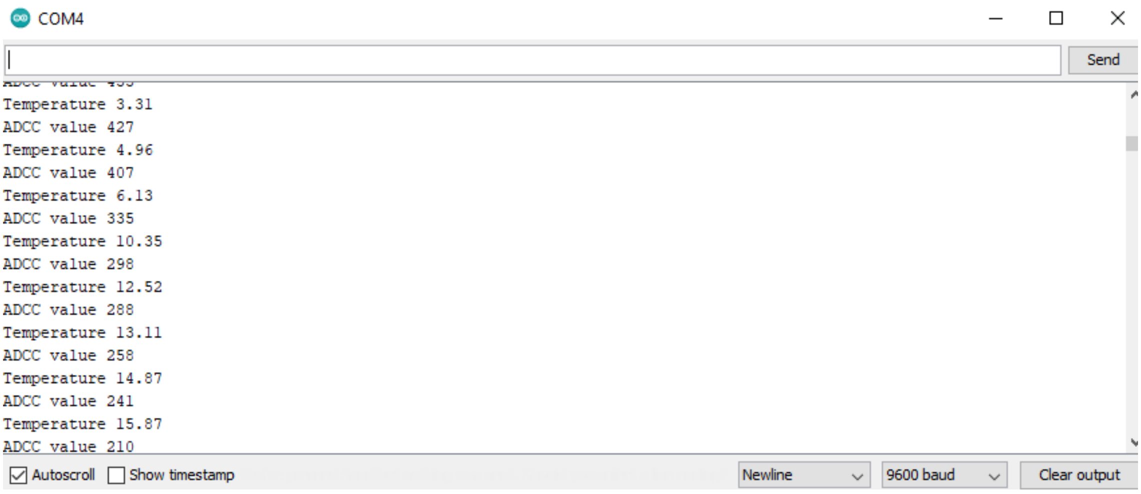


Figure 22: Arduino Temperature Value

2.4 PCB Design

After finishing with LTspice simulations, we draw a full schematic with PADS LOGIC:

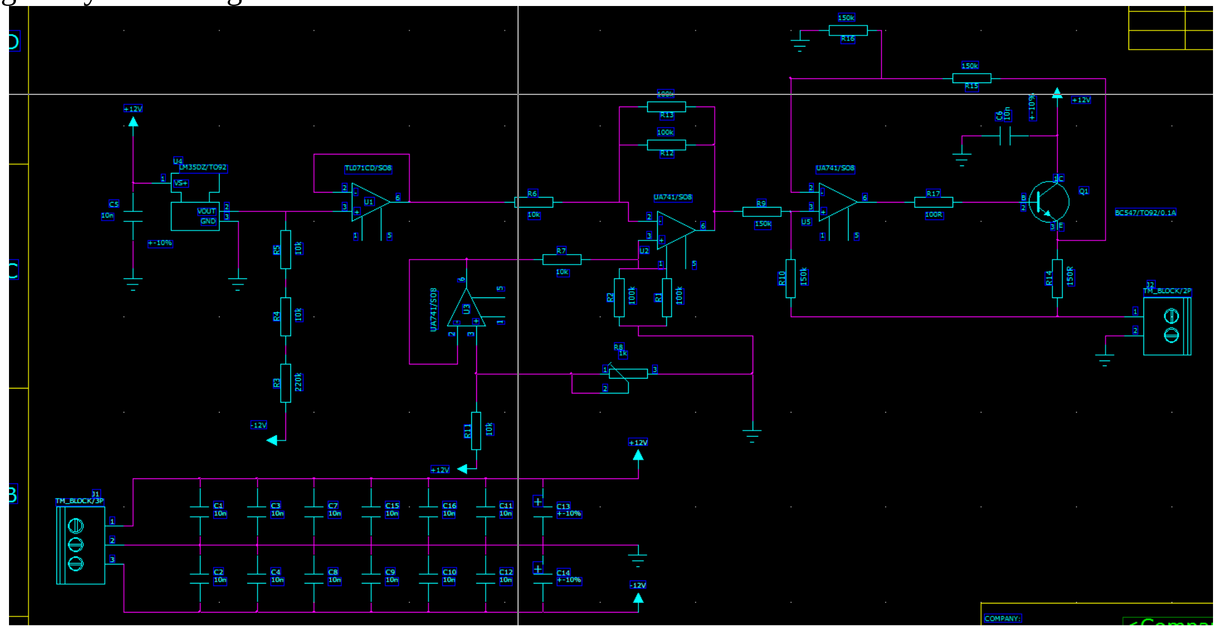


Figure 23: Schematic circuit of LM35 project

As noticed, we add 10nF capacitor to each of Vcc (+15V) and Vee (-15V) pins of all the OpAmp-s for protection in case of short circuit.

Then we send this net list to PADS LAYOUT (tools -> PADS Layout -> "choose Layout from search" -> back to Logics, choose Design -> Send Net List). If we make any changes in the schematic net, "ECO to PCB" is used to update the net in Layout.

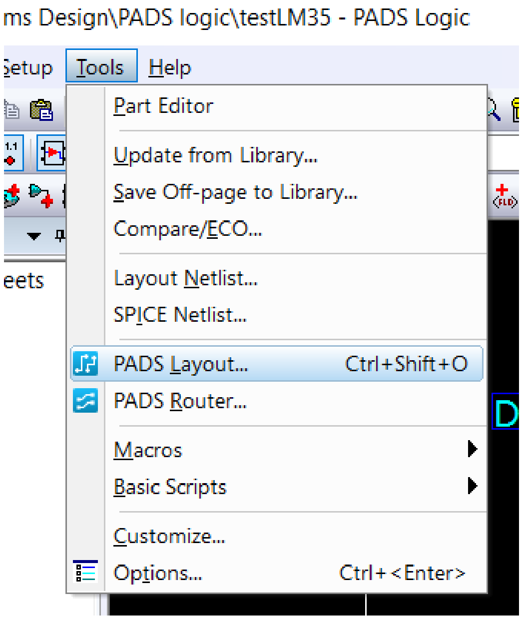


Figure 24: Choose Layout From Tools

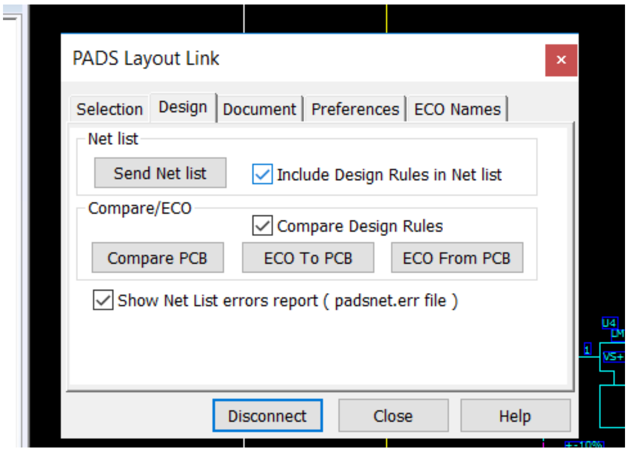


Figure 25: Send Net List And ECO

Next part is with PADS LAYOUT:

First thing can be noticed, all the components from schematic circuit are sent to this layout (also change from symbols to decals) and connected to each others. We may select a set of components in LOGICS and move them to the desired postion in LAYOUT.

Firstly, we change the design units for measuring from “mils” to “metric” in Tools -> Option -> General. Also go to Setup -> Design rules -> Defaul -> Clearance and change the trace width and all other things to 0.5 (mm); Via to 0.8 mm.

We need to draw an outline for the board (50mm – 95mm) as the white outline in the next figure. Then we start to design the PCB with adding traces and vias.

All the SMD components are on the Bottom side; all the Through-hole components are on the Top side.

Plus, for an easier designing work, we divide the circuit into 4 parts as simulation in LTSpice like this, and design each part at a time.

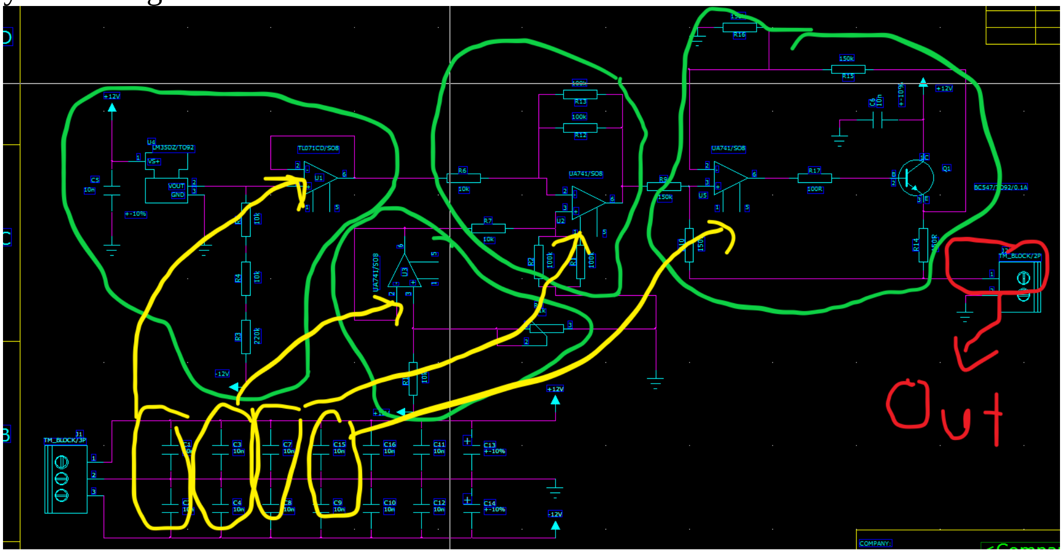


Figure 26: Divided Circuits

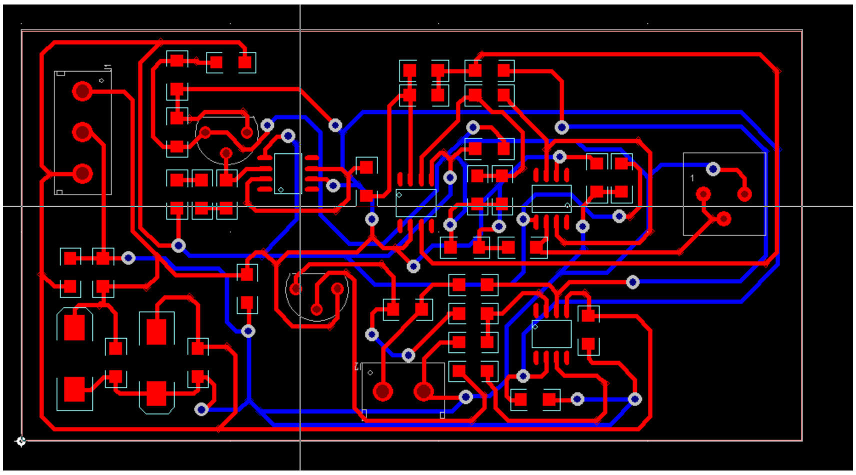


Figure 27: Our PCB after designing

After designing, go to Tools -> Verify design check for clearance and connectivity (remember that clearance is checked only for the circuit appears in the current screen).

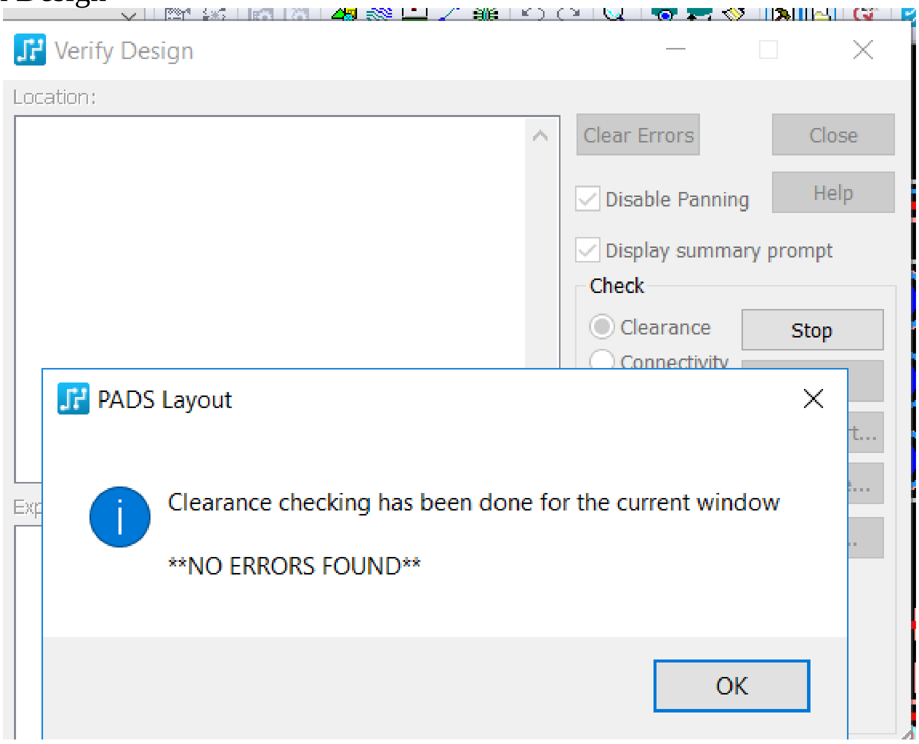


Figure 28: Verifying Design

Next part is to “pour copper” onto the PCB.

Start with Top-side; then follow the instruction: https://www.youtube.com/watch?v=L748DNqS6g4&feature=youtu.be

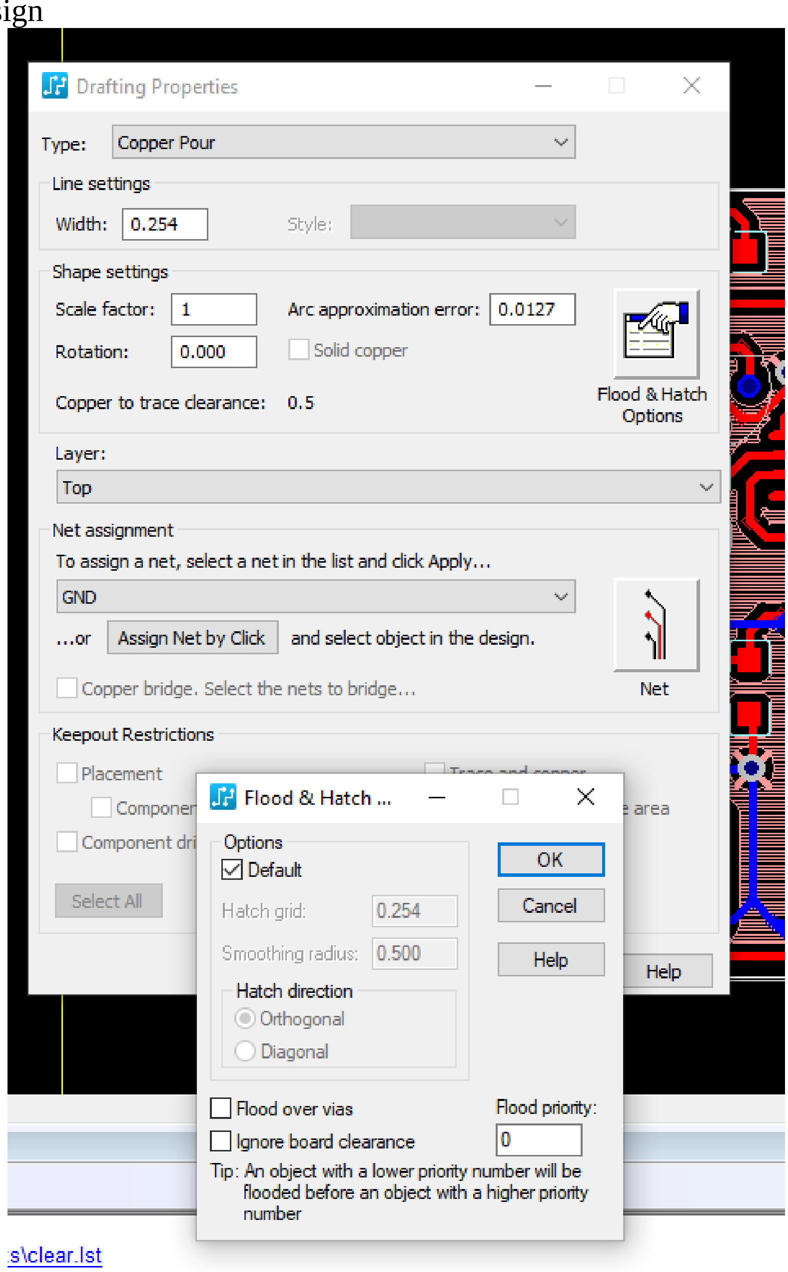


Figure 29: Pouring Copper Onto The Layout

After done with pour copper, the board looks like this (in Top-side view):

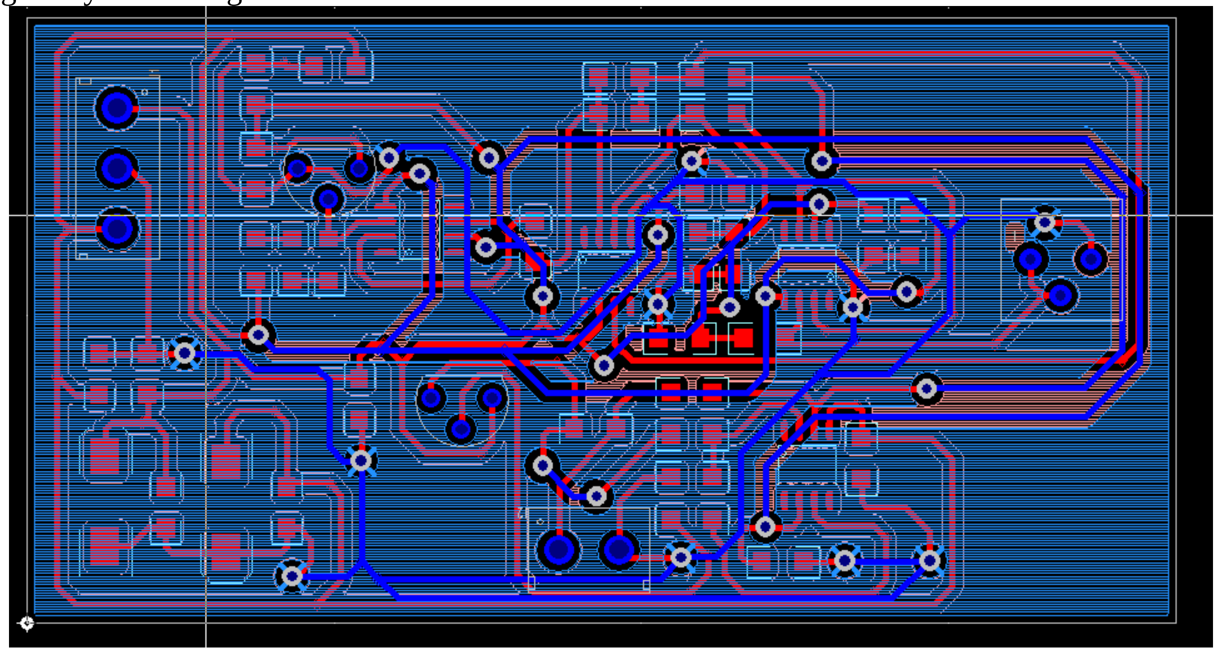


Figure 30: Top-Side After Pouring Copper

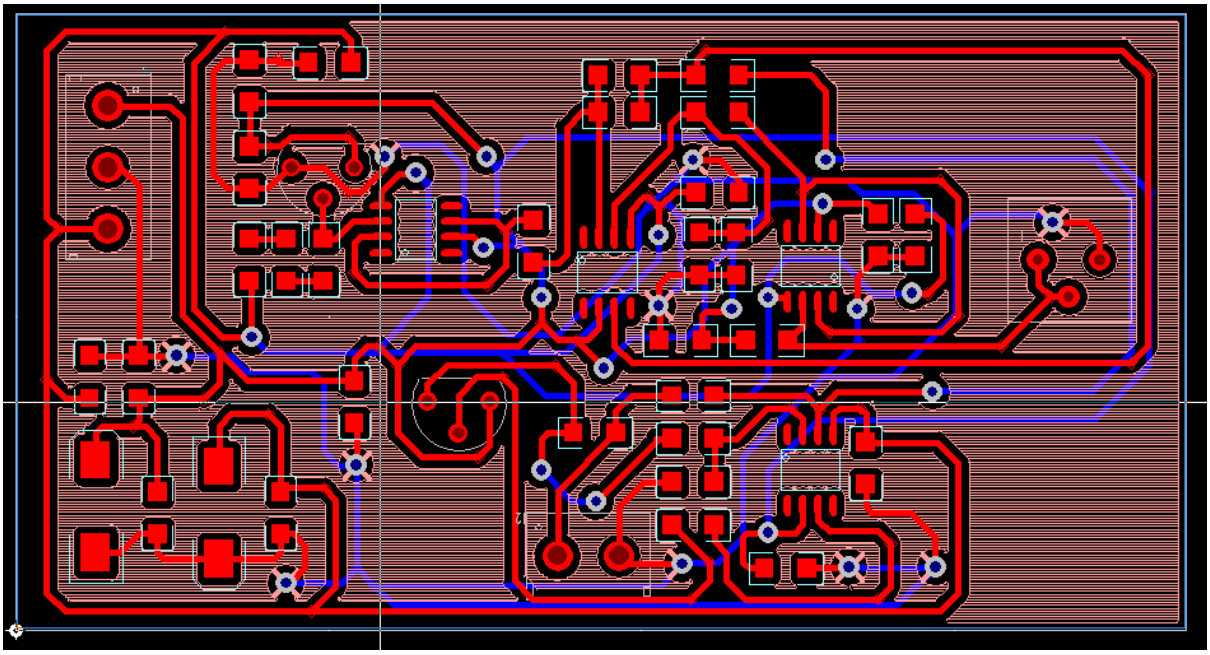


Figure 31. Bottom-side after pouring copper

Next part is to get some CAM files for both Top and Bottom layers: File -> CAM -> Add -> Layers and have these fields ticked:

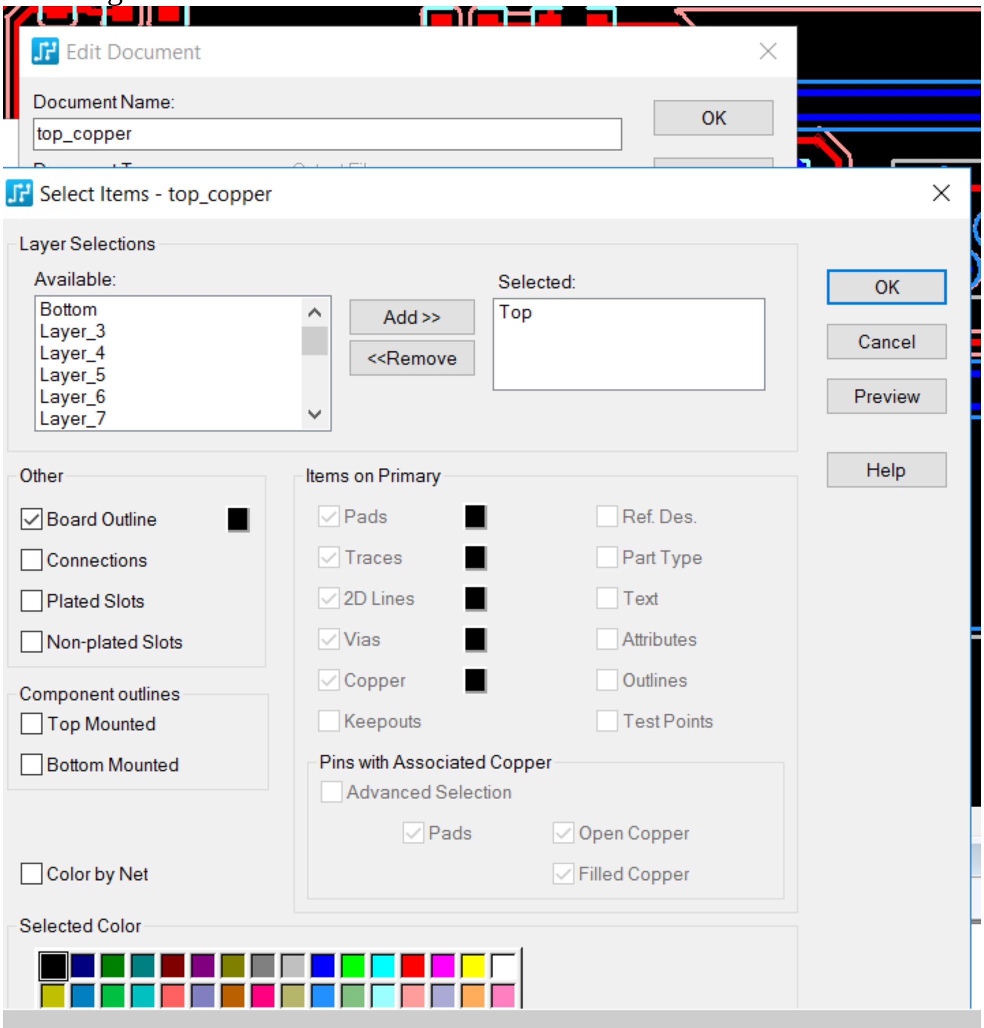


Figure 32. Top-layer copper CAM

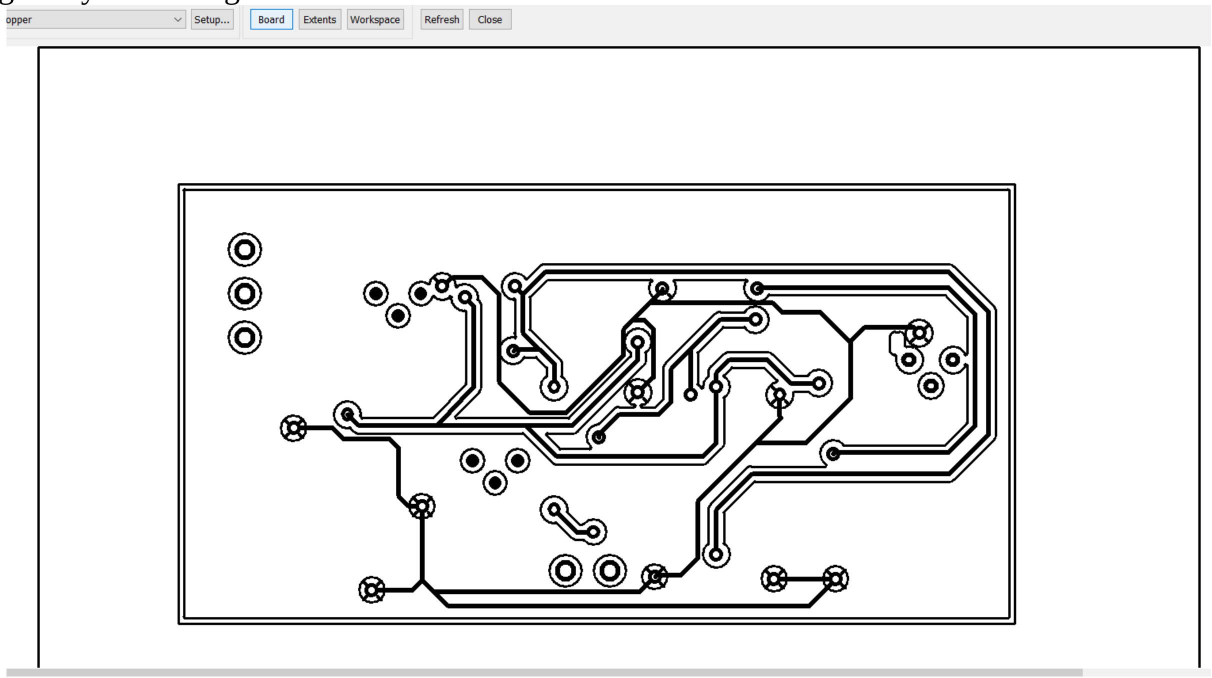


Figure 33. Preview of Top-copper

This file will be printed onto a A4 PLASTIC and we have one more file (for each layer) having only traces and pins of components for later soldering work. As seen, we have another outline for the PCB printing, for later aligning when print onto the physical board.

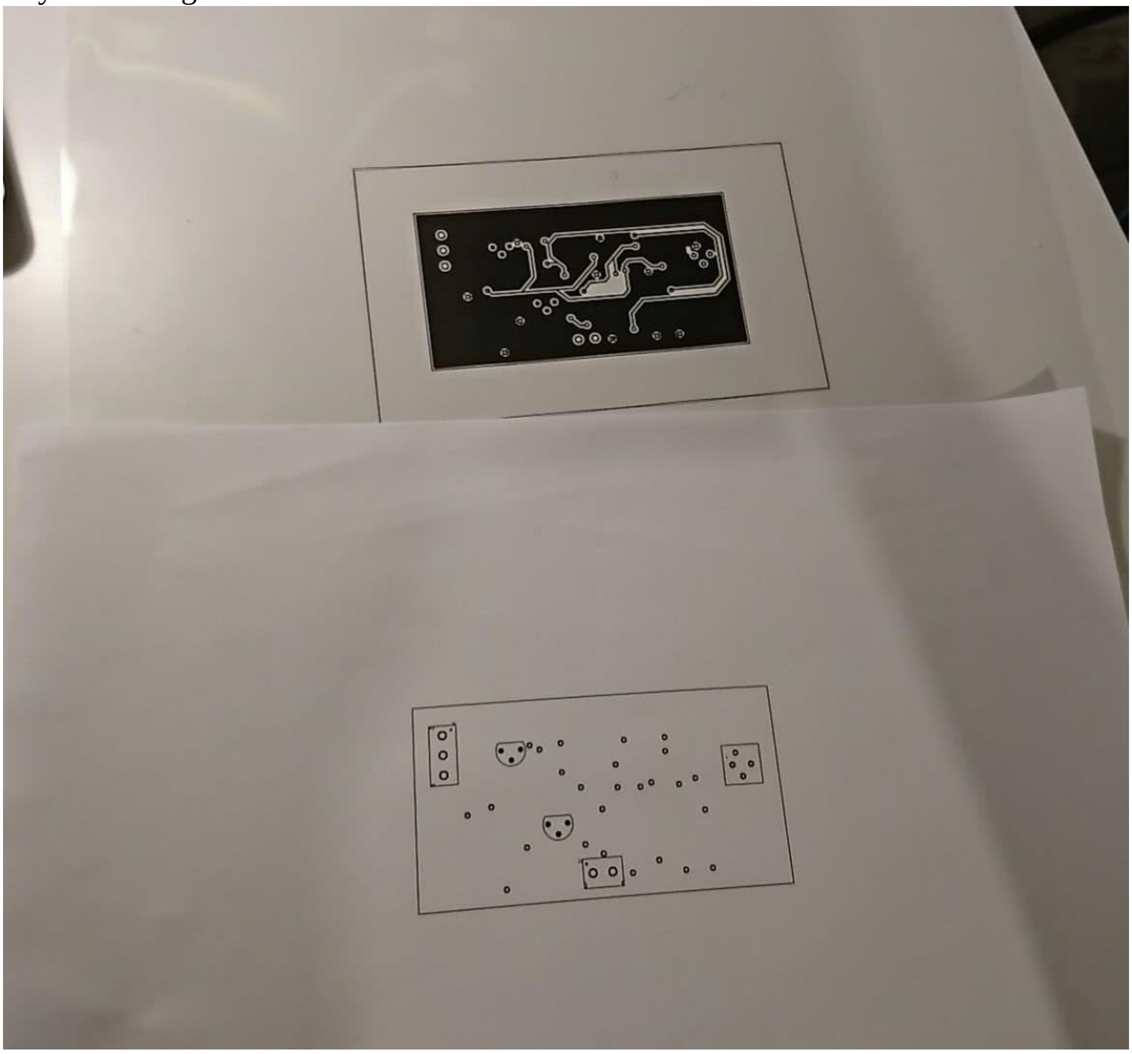


Figure 34. Printed CAM files

Next part is to print the design onto the physical board; firstly, choose the board size a little larger than 50*95mm. After some processes with high-pressured-and-concentrated-light-beam machine, 2 kinds of acids (I forgot to take the picture of the process, sorry ☹) several hours passed, we manage to print 2 well-qualified PCBs (we think so):

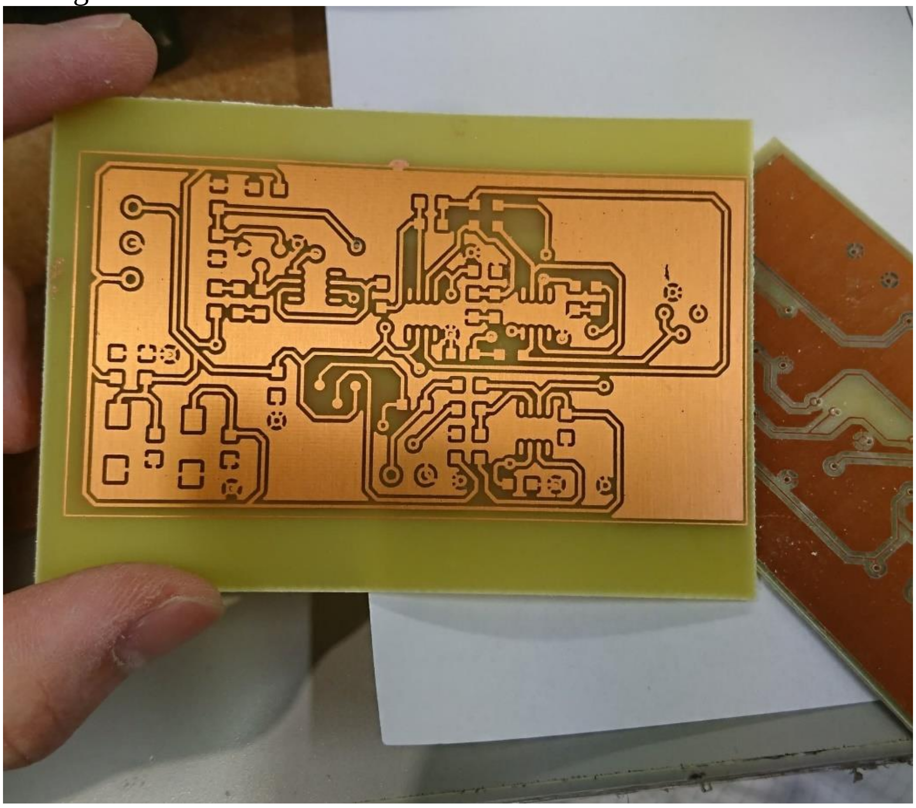


Figure 35. Printed PCB (Bottom side)

Although this PCB of mine (in the figure above) is inverted-mirrorly printed ☹, I managed to change the pins of asymmetric components (I regret printing wrongly for hours of this work) and started to solder them onto the board! (my teammate does not have a wrong PCB, just me!)

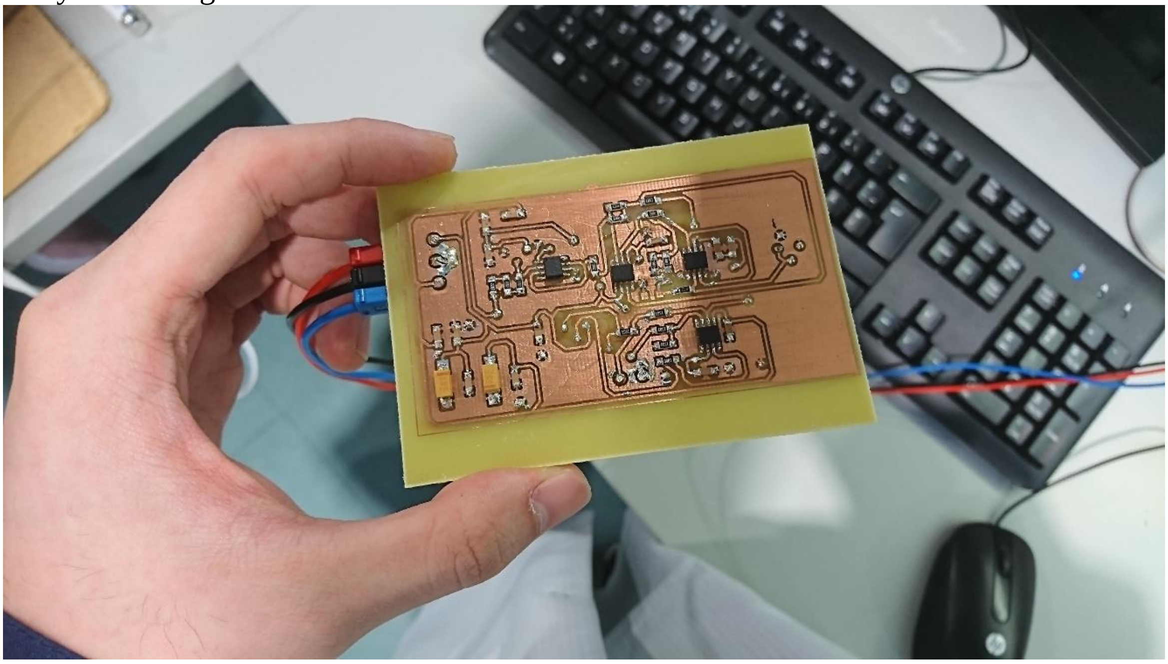


Figure 36. Completed PCB (Bottom side)

Final work includes Via and through holes drilling, cleaning the board, SMD and Through-hole components soldering and Via soldering.

The last step is to test the working system on the PCB!

3 CONCLUSION

After the course, we have revised experience in working with Simulation Softwares and Designing PCB; moreover, we also gain new knowledge about SMD components, soldering, testing system,… This Laboratory course is really informative and helpful for us!