EW-II Project-2

Self Adaptive Circuit

Mentor: Prof. Zia Abbas

TA: Neeraj Chanumolu

J Koundinya : 2021112022

P Chandana: 2021102006

Link to Presentation

Contents

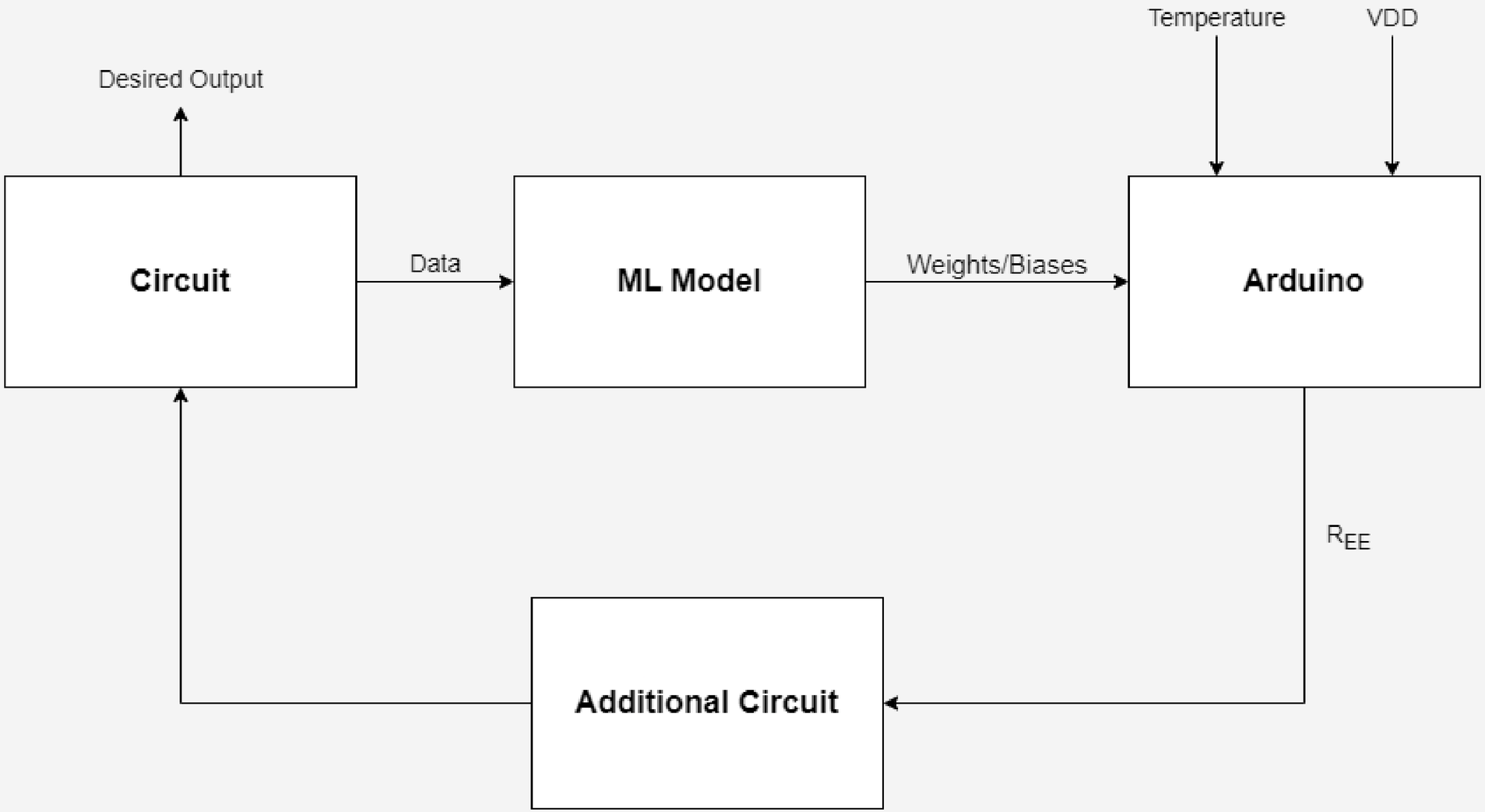
Introduction

- Self-adaptive circuits, also known as adaptive systems, are electronic circuits that can adapt to changes in their environment or to variations in their input signals. These circuits are designed to be flexible and able to adjust their behavior to optimize performance and energy efficiency.
- Self-adaptive design techniques are techniques for designing integrated circuits that can adjust the value of a circuit parameter during run time by feedback control, or allow the values of some key run-time parameters to be set after design time, by the use of resistors, capacitors, inductors, fuses, registers, memory, or some similar means.
- Overall, self-adaptive circuits have a wide range of applications in different fields, where the ability to adapt to changing conditions and optimize performance and energy efficiency is critical.

Goal

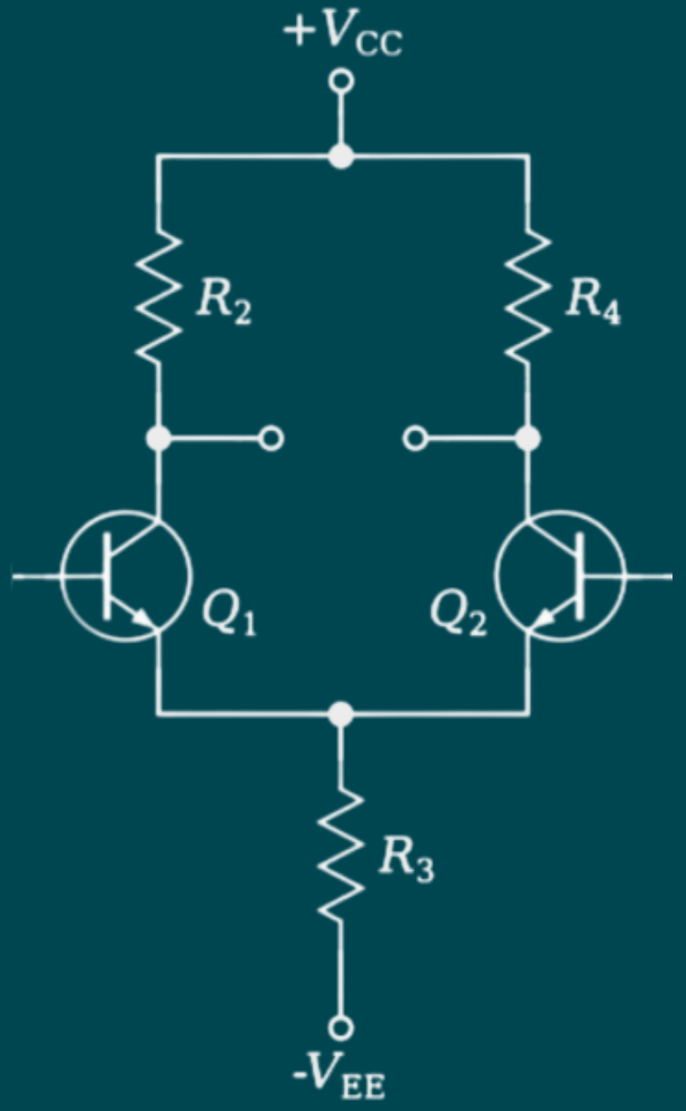
- The main aim of this project is to design a circuit and self-adapt it.
- For this, we first find out the dominating parameters of the circuit and analyze how the circuit depends on these parameters.
- We then use this dependency to find out how to control the output received.
- The output can be self-adapted by changing resistors, capacitors, inductors, fuses, registers, memory, etc. to obtain the desired output.

Block Diagram



Circuit

- The circuit which we have chosen to self-adapt is the Differential Amplifier.
- A differential amplifier is an electronic circuit that amplifies the difference between two input signals and gives the output voltage that is proportional to the difference between the two input voltages, amplified by a gain factor. It is a fundamental building block of many electronic circuits, including operational amplifiers, instrumentation amplifiers, and differential input stages of analog-to-digital converters.
- We designed our differential amplifier to have a gain of 150.



Design

- For the differential amplifier to have 150 as gain we used the following equations and obtained the value of R as follows:

$$I_{c_1} + I_{c_2} = I_{EE}$$

$$I_{c_1} = I_{c_2} = I_c$$

$$I_c = \frac{I_{EE}}{2}$$

Design

$$A_d = \frac{I_c}{V_T} \times \frac{R_c}{2}$$

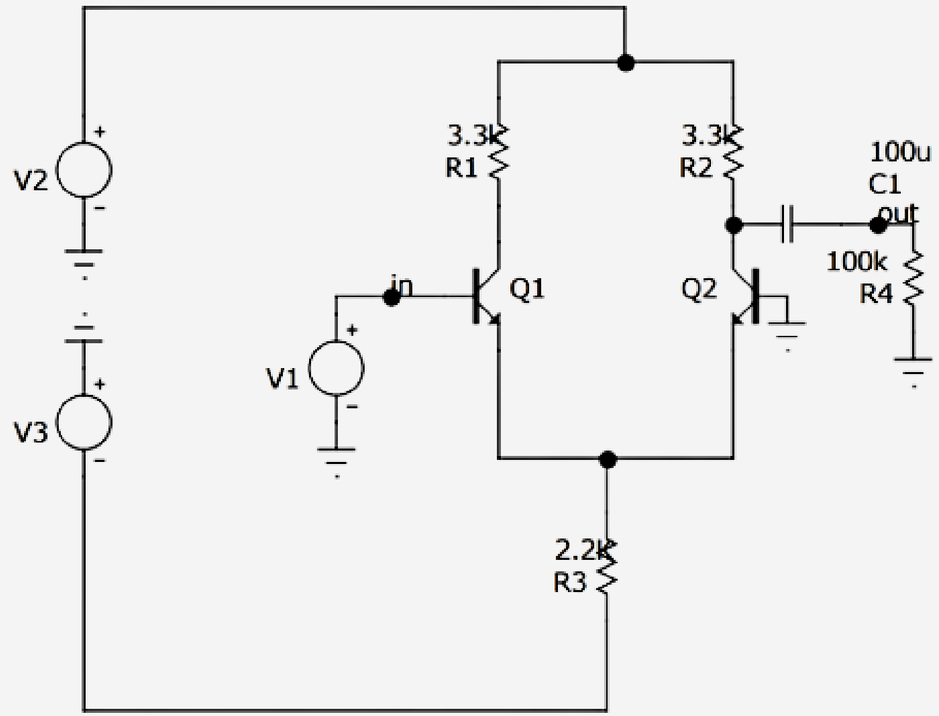
$$R_c = 3.3k\Omega$$

- By KVL, we get the following equation, and by solving it we get the value of R as follows:

$$12 - 2I_{EE}R_{EE} - 0.7 = -12$$

$$R_{EE} = 2.63k\Omega$$

Differential Amplifier



- This is the circuit schematic from Micro Cap

ML Model

We use the data from the circuit simulation to train the machine learning model, which uses linear regression to determine the dependence of the parameters and compute the weights and biases. With the help of these weights and biases, we can predict the result given an input.

Linear Regression model

Code:

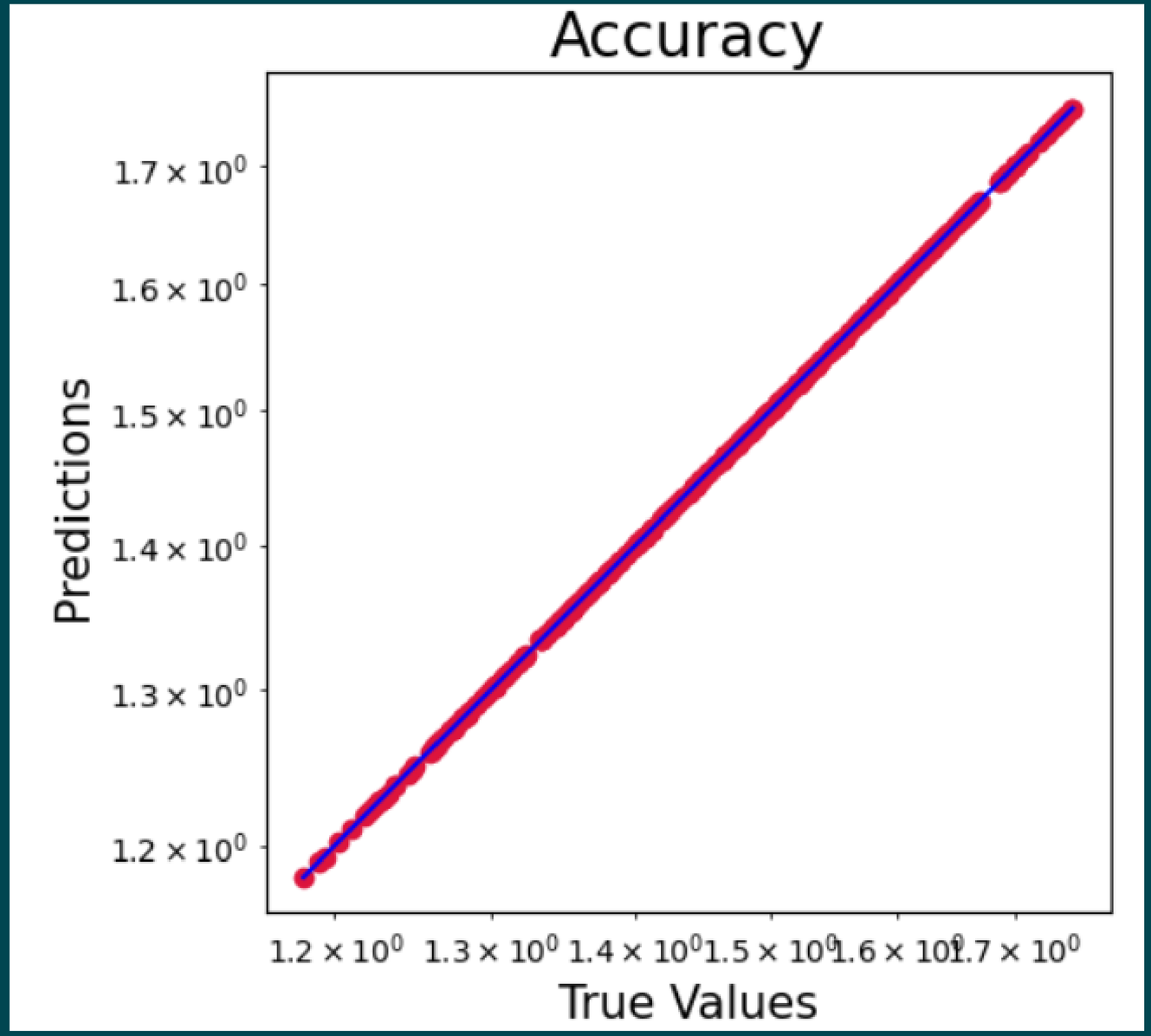
```
regr = LinearRegression()

regr.fit(x_train, y_train)
predicted_outcome = regr.predict(x_test)
# print(y_test);
print(predicted_outcome[:5])
regr.coef_
```

```
# trying polynomial regression

from sklearn.preprocessing import PolynomialFeatures
from sklearn.pipeline import make_pipeline
from sklearn.metrics import accuracy_score

poly_reg = PolynomialFeatures(degree = 2)
x_poly = poly_reg.fit_transform(x_train)
lin_reg_2=LinearRegression().fit(x_poly, y_train)
y_pred = lin_reg_2.predict(poly_reg.fit_transform(x_test))
# print(pred);

lin_reg_2.coef_
```

This linear regression model gives us the weights and biases that can be used to calculate the output voltage given the input values. We can find the output voltage as follows,

$$\text{Output} = \text{weight}[1] \times \text{Temp} + \text{weight}[2] \times \text{VDD} + \text{Bias}$$

Accuracy of the ML model



The red line represents the output voltage obtained by passing test inputs to the model.
The blue line represents our linear regression model that predicts the output voltage.

The accuracy of the model is more than 99% which gives us a very accurate output.

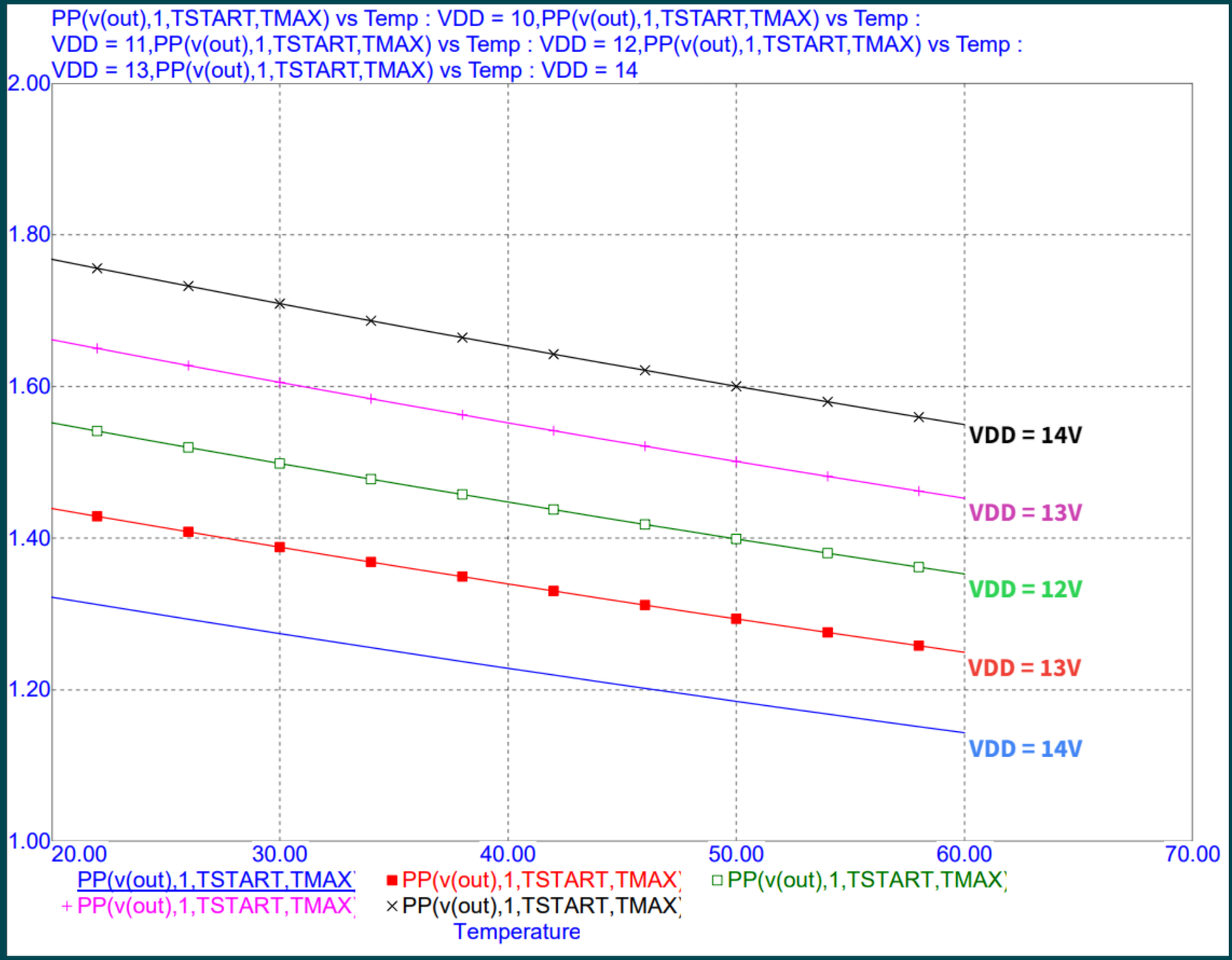
Microprocessor

We modelled an equation that takes temperature and VDD as inputs and predicts the outcome using the weights and biases from the ML model. Now, the change in REE is calculated using this predicted output. With the help of a BMP sensor and a simple voltage divider circuit, the Arduino measures the room temperature and the voltage. Given is the Arduino code for predicting REE after predicting the output.

```
// Conditions after getting the output from the ML model
if(output >= 1.13 && output < 1.18)
{
  R_EE = 1.6;
}
else if(output >= 1.18 && output < 1.25)
{
  R_EE = 1.7;
}
else if(output >= 1.25 && output < 1.30)
{
  R_EE = 1.8;
}
else if(output >= 1.30 && output < 1.36)
{
  R_EE = 1.9;
}
else if(output >= 1.36 && output < 1.41)
{
  R_EE = 2.0;
}
else if(output >= 1.41 && output < 1.48)
{
  R_EE = 2.1;
}
else if(output >= 1.48 && output < 1.52)
{
  R_EE = 2.2;
}
else if(output >= 1.52 && output < 1.58)
{
  R_EE = 2.3;
}
else if(output >= 1.58 && output < 1.65)
{
  R_EE = 2.4;
}
```

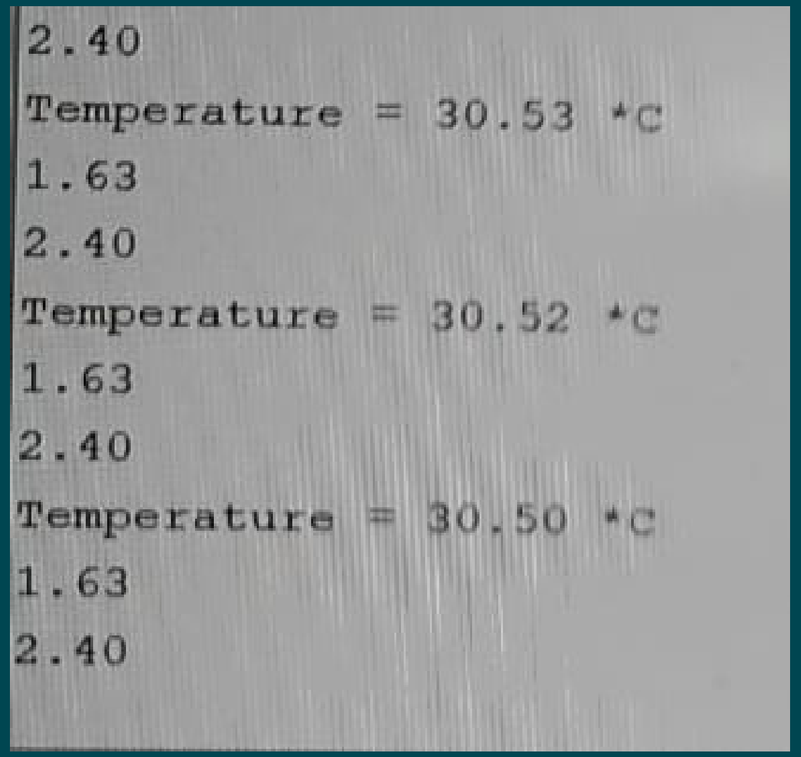
Results

This is the graph obtained for the output vs temperature at different VDD. We observed that the output varies linearly with the temperature



Results

The output from the serial monitor of Arduino is observed as follows, where for a temperature of 30.52 degrees Celsius, and a VDD of 13, the predicted output was 1.63V, and by changing the value of REE to 2.4K Ohm, we get desired output of 1.50 V.



Here is the link for the working model of our circuit,

Video Link

Applications

- **Communications systems:** Self-adaptive circuits can be used to optimize signal quality and reliability. For example, in a wireless communication system, the circuit can adjust the power level, modulation scheme, and channel allocation to compensate for variations in the signal strength and interference.

- **Power management:** Self-adaptive circuits can adjust the voltage and frequency of the processor and other components based on the workload and the battery level, which can extend the battery life and reduce heat dissipation.

- **Sensor networks:** Self-adaptive circuits can be used in sensor networks to adjust the sampling rate and threshold values based on the sensor data and the application requirements. This can reduce the amount of data transmitted and processed, which can save energy and improve the scalability of the network.

- **Autonomous vehicles:** Self-adaptive circuits can be used in autonomous vehicles to adjust the sensor data fusion and the control algorithms based on the road conditions and the vehicle dynamics. This can improve the safety and efficiency of the vehicle in different driving scenarios.

- **Medical devices:** Self-adaptive circuits can be used in medical devices to optimize therapy delivery and patient monitoring based on the patient's physiological data and the treatment plan. This can improve the effectiveness and comfort of the therapy, and reduce the risk of adverse effects.

Acknowledgements

- We would really like to thank Prof. Zia Abbas, Prof. Anshu Sarje, Prof. Arti Yardi and Prashant Ranjan sir for helping and supporting us during the course of this project. We would also like to thank the TAs Neeraj sir and Vishwanath sir for guiding us throughout the project.


Thank
You