Team 6 Project: SHARP-IR Sensor

By Rishi Zaveri

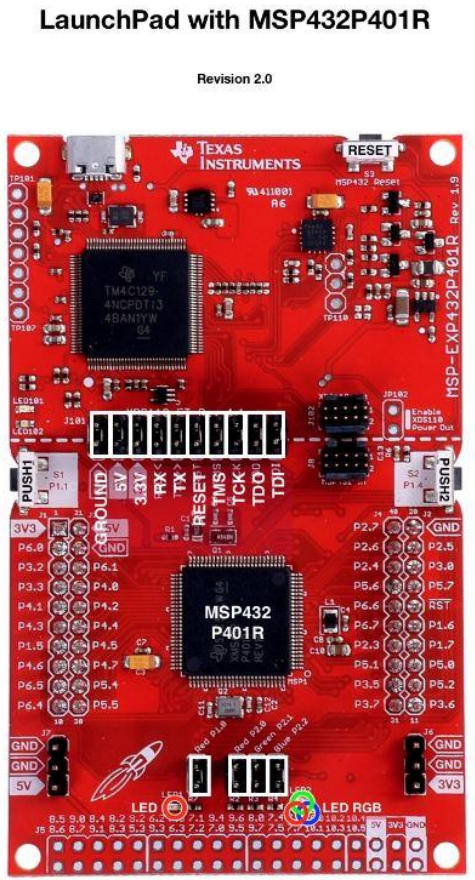
IR Sensor GP2Y0A02YK0F

This project uses a precision IR sensor to measure distance from 20cm to 150cm. It requires a supply voltage around 5 volts and a ground pin connection. It outputs an analog voltage between 0 and 2.85 volts.

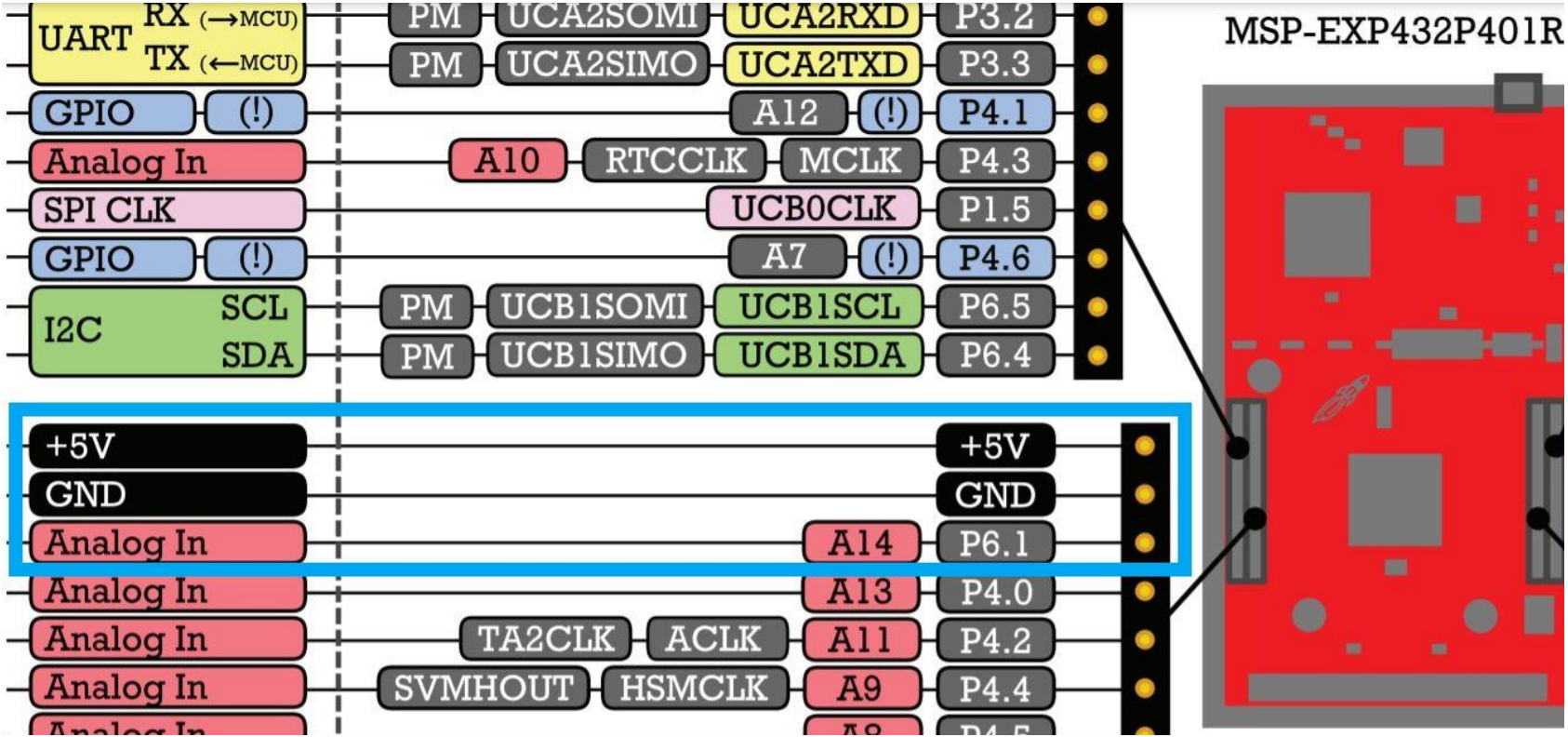
Distance Measuring Sensor Unit
Measuring distance: 20 to 150 cm
Analog output type

ADC Converter TI MSP432

An ADC converter is used to obtain the analog voltage data from the sensor. The USB cable is used for both powering the device and transferring data to the Raspberry Pi. The Energia IDE helps to code this device.

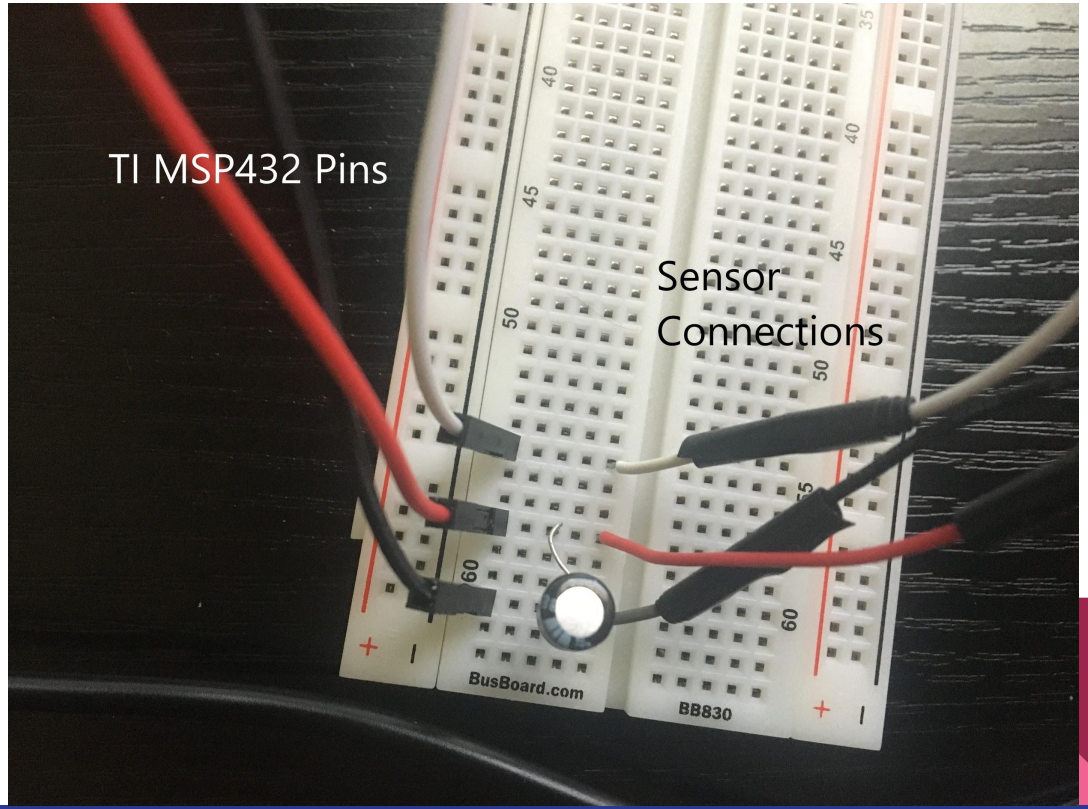


TI MSP-EXP432 Pins 21-23



Circuit

Circuit connections between sensor and MSP-EXP432 device, color coded appropriately. 10uF by-pass capacitor used to stabilize power supply line.



C++ Code for MSP432

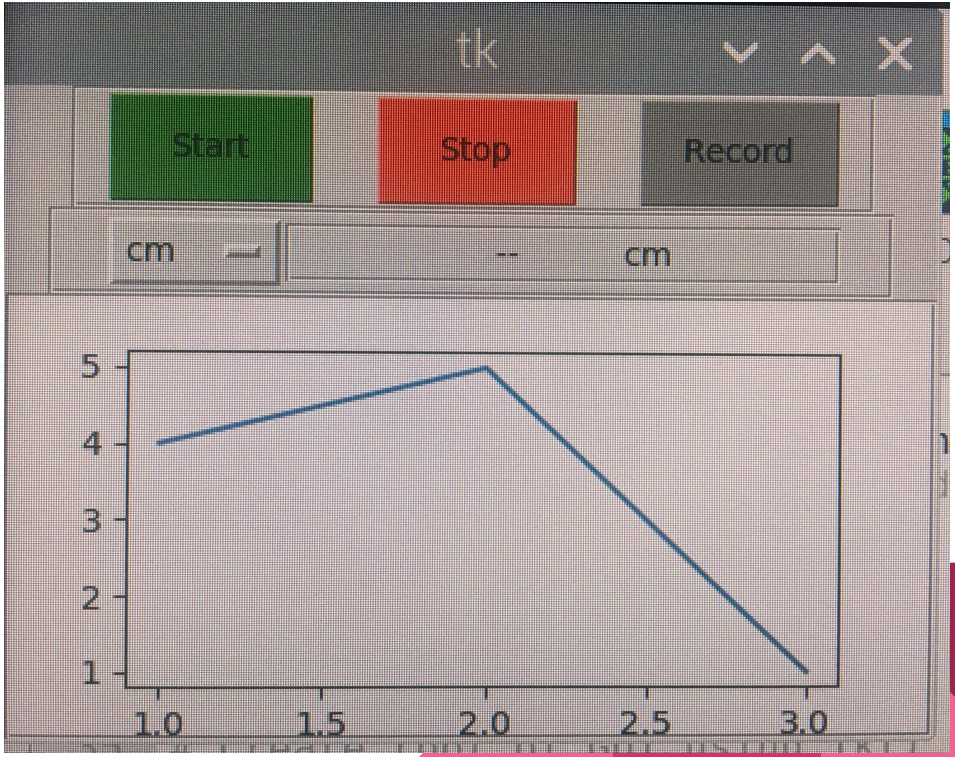
In programming this device, pin 23 is used as the analog input. An integer variable is used to capture the analog value from this pin. The Serial library prints this value every half second. The baud rate is set for 9600 bits/seconds.

EE437Team6ProjectUSBOut

```
// Code by Rishi Zaveri
// Subject: EE437
// 2 November 2020

int analogPin = 23; // Pin 23 Analog Input
int analogValue = 0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  analogValue = analogRead(analogPin);
  Serial.println(analogValue);
  delay(500);
}
```

Program GUI

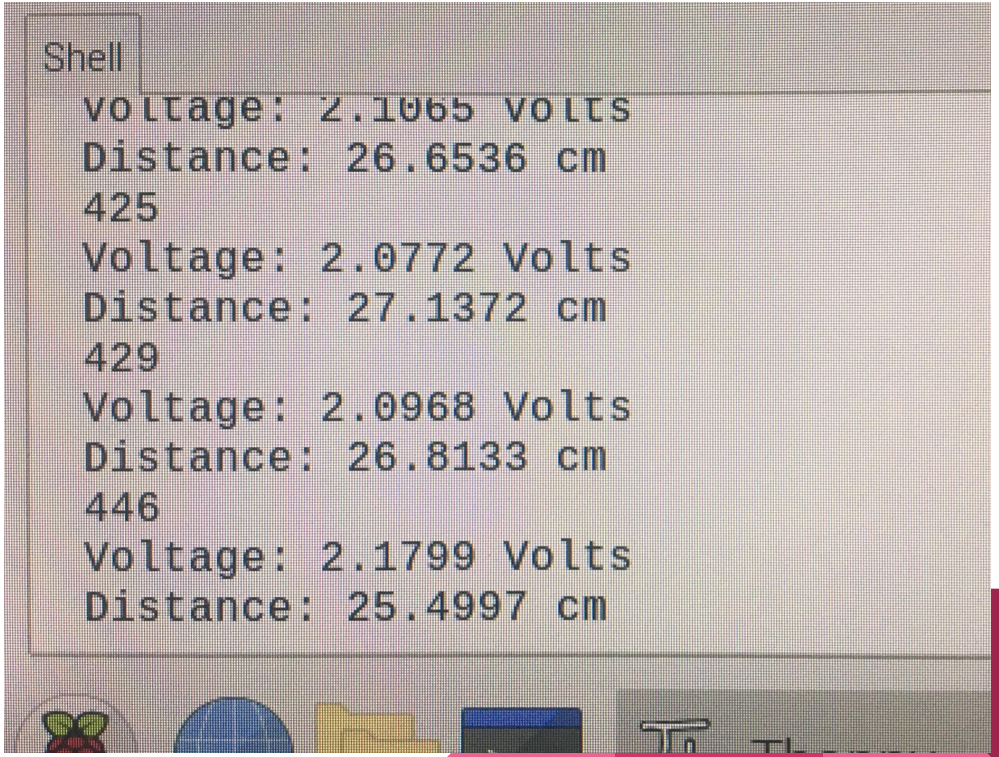
A GUI is produced during runtime using Python's Tkinter library. The first row features three buttons for user interaction. The second row displays the measurement value and contains a drop down menu for switching between cm, mm, and inch. Finally, the third row displays the graph.



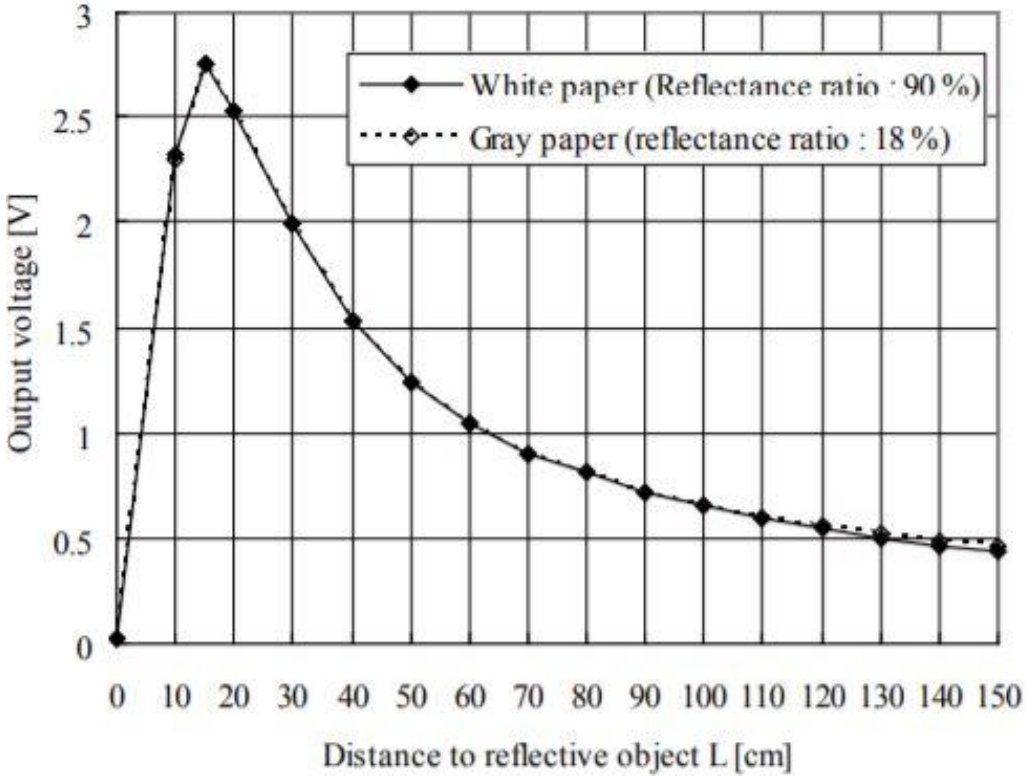
```
# Produce sensor values
sensorValue = " "
voltageValue = 0.0
ser = serial.Serial('/dev/ttyACM0')

def sensing():
    while True:
        sensorValue = ser.readline()
        sensorValue = sensorValue.decode('utf-8').strip()
        print(sensorValue)
        voltageValue = int(sensorValue) * (5.0/1023.0)
        print("Voltage: " + str(round(voltageValue, 4)) + " Volts")

        cmDistance = 10650.08 * pow(int(sensorValue),-0.935) - 10
        print("Distance: " + str(round(cmDistance, 4)) + " cm")

t1 = threading.Thread(target=sensing)
t1.daemon = True # daemon thread terminates when program exits
t1.start()
```

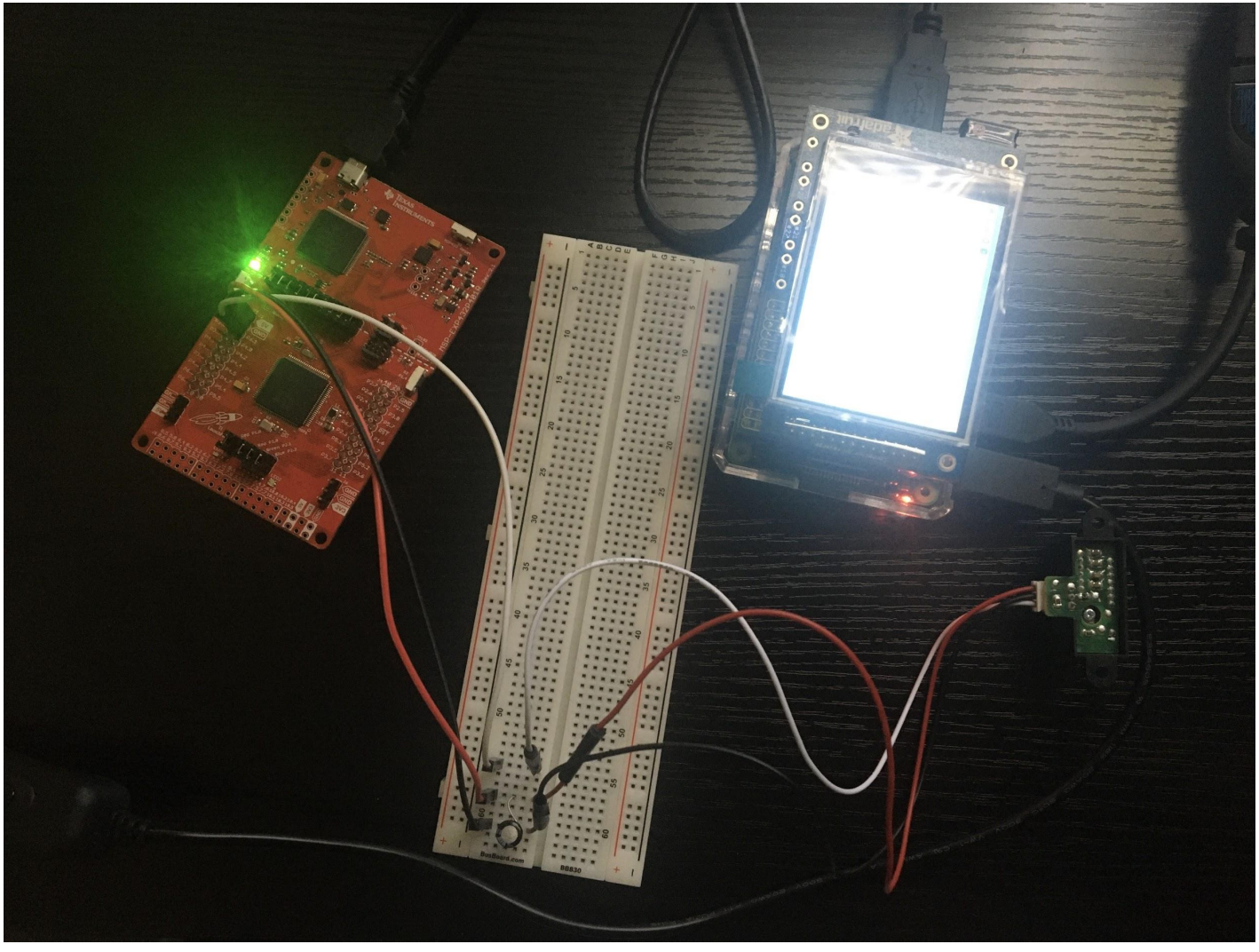
Data Collection

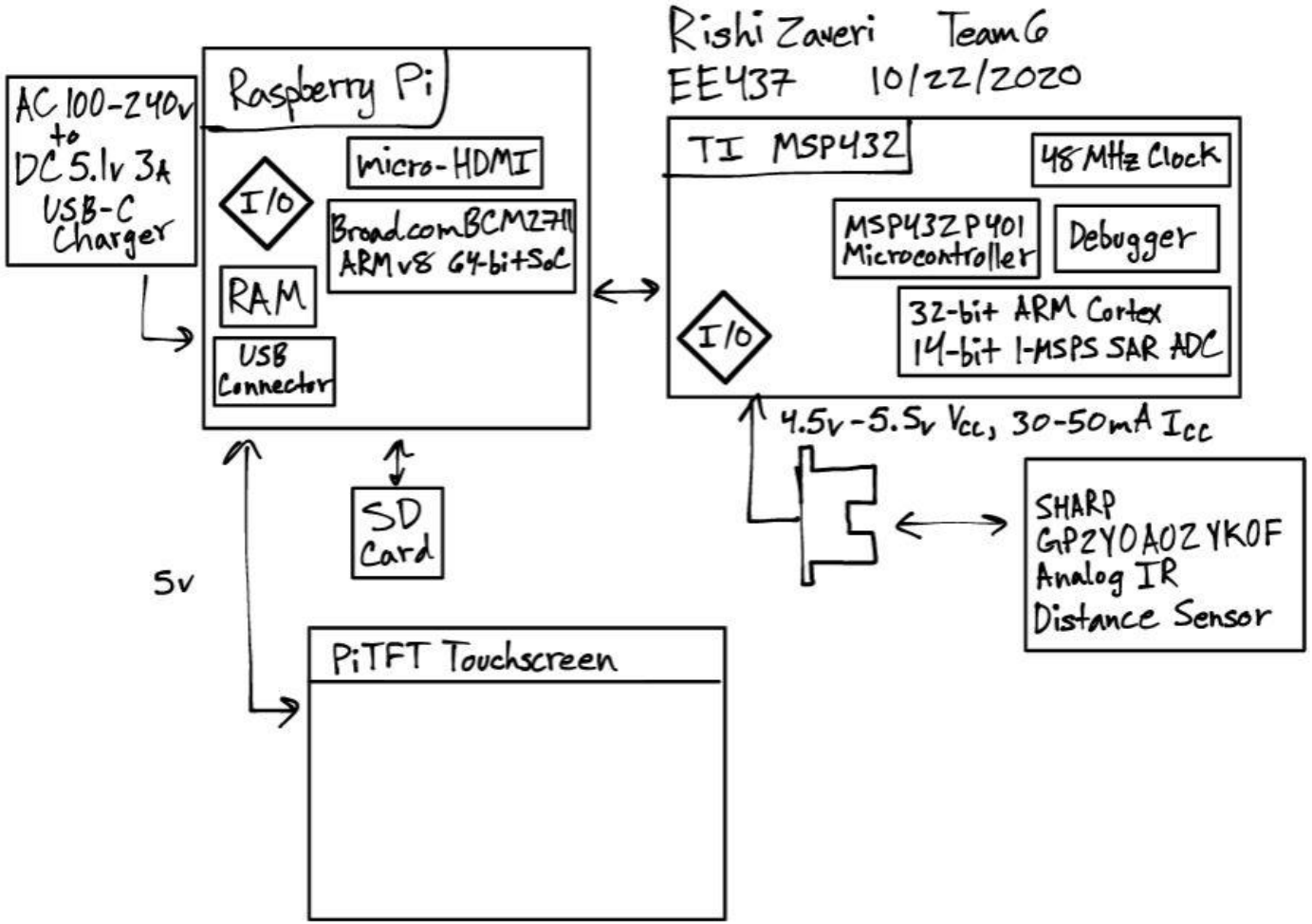
While the program is running, the Raspberry Pi collects information from the sensor in digital form. The data is then processed into the more useful voltage and distance values.



IR Sensor Distance-Voltage Relationship






Rishi Zaveri Team 6
EE437 10/22/2020
AC 100-240v to DC 5.1v 3A USB-C Charger
Raspberry Pi
Micro-HDMI
I/O
Broadcom BCM2711 ARMv8 64-bit SoC
RAM
USB Connector
SD Card
5v
PiTFT Touchscreen
TI MSP432
48 MHz Clock
MSP432P401 Microcontroller
Debugger
I/O
32-bit ARM Cortex
14-bit 1-MSPS SAR ADC
4.5v - 5.5v Vcc, 30-50mA Icc
SHARP GP2Y0A02YK0F Analog IR Distance Sensor

Rishi Zaveri, Team 6	Bill of Materials	
	Part	**Qty**
1	TI MSP-EXP432P401R with	1
2	Raspberry Pi 4 Model B	1
3	SD Card with Raspbian OS	1
4	SHARP GP2Y0A02YK0F IR Sensor	1
5	Breadboard	1
6	Raspberry Pi Power Supply 5V	1
7	Pi TFT 2.8'' Touchscreen	1
8	Jumper Wires Male/Female	3
9	10 microFarad Capacitor	1

End