int analogPinA0=A0;

int value=0;

double Vref=5.0;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

}

void loop() {

value=analogRead(analogPinA0);

double voltageA0=value\*Vref/1023;

Serial.println((String)"Voltage given to ADC A0: "+voltageA0);

Serial.println("------------------------------");

delay(3000);

}

int random\_variable;

int static\_variable = 500;

void setup() {

Serial.begin(9600);

}

void loop() {

random\_variable = random(0, 1000);

Serial.print("Variable\_1:");

Serial.print(random\_variable);

Serial.print(",");

Serial.print("Variable\_2:");

Serial.println(static\_variable);

delay(500);

}

int analogPinA0=A0;

int value=0;

double Vref=5.08;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

}

void loop() {

value=analogRead(analogPinA0);

double voltageA0=value\*Vref/1023;

Serial.println(voltageA0);

delay(50);

}

int analogPinA0=A0;

int value=0;

double Vref=3.3;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

analogReference(EXTERNAL);

}

void loop() {

value=analogRead(analogPinA0);

double voltageA0=value\*Vref/1023;

Serial.println((String)"Voltage given to ADC A0: "+voltageA0);

Serial.println("------------------------------");

delay(3000);

}

//ADC

//On ATmega based boards (UNO, Nano, Mini, Mega), it takes about 100 microseconds

//to read an analog input, so the maximum reading rate is about 10,000 times a second.

//https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/

int analogPinA0=A0;

int value=0;

double Vref=5.0;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

}

void loop() {

double t0=micros();

value=analogRead(analogPinA0);

double t1=micros();

int deltaT=t1-t0;

Serial.println((String)"Value read from ADC A0: "+value);

Serial.println();

Serial.println((String)"Conversion time (us): "+deltaT);

Serial.println();

double voltageA0=value\*Vref/1023;

Serial.println((String)"Voltage given to ADC A0: "+voltageA0);

Serial.println("-------------------------------------");

delay(3000);

}

int analogPin=A0;

double voltage=0;

double Vref=5.0;

const int LED1=2;

const int LED2=3;

const int LED3=4;

const int LED4=5;

const int LED5=6;

void setup() {

// put your setup code here, to run once:

pinMode(LED1,OUTPUT);

pinMode(LED2,OUTPUT);

pinMode(LED3,OUTPUT);

pinMode(LED4,OUTPUT);

pinMode(LED5,OUTPUT);

}

void loop() {

// put your main code here, to run repeatedly:

voltage=analogRead(analogPin)/1023.0\*Vref;

if (voltage>0.0 && voltage<=1.0){

digitalWrite(LED1,HIGH);

digitalWrite(LED2,LOW);

digitalWrite(LED3,LOW);

digitalWrite(LED4,LOW);

digitalWrite(LED5,LOW);

}

if (voltage>1.0 && voltage<=2.0){

digitalWrite(LED1,HIGH);

digitalWrite(LED2,HIGH);

digitalWrite(LED3,LOW);

digitalWrite(LED4,LOW);

digitalWrite(LED5,LOW);

}

if (voltage>2.0 && voltage<=3.0){

digitalWrite(LED1,HIGH);

digitalWrite(LED2,HIGH);

digitalWrite(LED3,HIGH);

digitalWrite(LED4,LOW);

digitalWrite(LED5,LOW);

}

if (voltage>3.0 && voltage<=4.0){

digitalWrite(LED1,HIGH);

digitalWrite(LED2,HIGH);

digitalWrite(LED3,HIGH);

digitalWrite(LED4,HIGH);

digitalWrite(LED5,LOW);

}

if (voltage>4.0 && voltage<=5.0){

digitalWrite(LED1,HIGH);

digitalWrite(LED2,HIGH);

digitalWrite(LED3,HIGH);

digitalWrite(LED4,HIGH);

digitalWrite(LED5,HIGH);

}

delay(250);

}

int value=0;

float Vref=5.0;

float sensorSensitivity=66e-3; //ACS712ELC-30 is used here.

float current=0;

float sum=0;

int n=50;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

for (int i=0;i<n;i++){

value=analogRead(A0)-511;

sum=sum+(value\*Vref/1023)/sensorSensitivity;

delay(3);

}

current=sum/n;

sum=0;

Serial.println((String)"Measured current is: "+current);

Serial.println("---");

delay(500);

}

/\*PCF8591 Module Analog to Digital test program.

Essentially, this tests the I2C communications to the chip.

The chip address is 0x90.

\*/

#include <Wire.h>

#define PCF8591 (0x90 >> 1) // Device address = 0

#define PCF8591\_DAC\_ENABLE 0x40

#define PCF8591\_ADC\_CH0 0x40

#define PCF8591\_ADC\_CH1 0x41

#define PCF8591\_ADC\_CH2 0x42

#define PCF8591\_ADC\_CH3 0x43

byte adc\_value;

float voltage=0;

byte getADC(byte config){

Wire.beginTransmission(PCF8591);

Wire.write(config);

Wire.endTransmission();

Wire.requestFrom((int) PCF8591,2);

while (Wire.available()){

adc\_value = Wire.read(); //This needs two reads to get the value.

adc\_value = Wire.read();

}

return adc\_value;

}

void setup(){

Serial.begin(9600);

Wire.begin();

Serial.println("ADC Test");

}

void loop()

{

adc\_value = getADC(PCF8591\_ADC\_CH3); //Channel 3 is the pot

voltage=adc\_value\*5.0/255.0;

Serial.println((String)"Voltage of center terminal of the potentiometer is: "+voltage+" V.");

delay(500);

}

/\*PCF8591 Module Digital To analog test program.

Essentially, this tests the I2C communications to the chip.

The chip address is 0x90.

\*/

#include <Wire.h>

#define PCF8591 (0x90 >> 1) // Device address = 0

#define PCF8591\_DAC\_ENABLE 0x40

#define PCF8591\_ADC\_CH0 0x40

#define PCF8591\_ADC\_CH1 0x41

#define PCF8591\_ADC\_CH2 0x42

#define PCF8591\_ADC\_CH3 0x43

byte dac\_value=0;

byte enteredValue=0;

void putDAC(byte dac\_value){

Wire.beginTransmission(PCF8591); //Calls the 8591 to attention.

Wire.write(PCF8591\_DAC\_ENABLE); //Send a DAC enable word.

Wire.write(dac\_value); //Send the desired DAC value (0-255)

Wire.endTransmission();

}

void setup(){

Serial.begin(9600);

Serial.println("Please enter a number in the 0-255 range:");

Wire.begin();

}

void loop(){

while(Serial.available()>0){

enteredValue=Serial.parseInt();

Serial.println((String)"Entered number is:"+enteredValue);

putDAC(enteredValue);

Serial.println();

Serial.println("Please enter a number in the 0-255 range:");

}

}