```
/*
Single scan, fixed values
V_Plate: 3-27 V step 0.1 every 200 ms.
V_G1: 0 -3 V step 0.2V
*/

#include <Wire.h>
#include <TimerThree.h>

long const vref = 5100;
long const aref = 4096;
long const Pstart = 1000;
long const Pend  = 25000;// max value for the current measurement circuit
long const Pstep = 100;
long const G1start = 0;
long const G1end = 2400;
long const G1step = 400;
String inputString  ="";        // a string to hold incoming data
boolean stringComplete  =false;  // whether the string is complete
char buf[10];
long value;
long millivoltPlate = Pstart;
long millivoltG1  = G1start;// start value
int analogPin = 0;
int analogPinA = 1;
long    out     =     0;// variable to store the value read
unsigned long time;
unsigned long prevtime = 0;
long millivoltPlateRead;
long milliampPlateRead;
long uAPlate;
// G1 DAC
const byte PWMDAC1pin  = 3;// PWM DAC in MEGA
const byte period  = 32;//  for  8  bit  DAC
volatile boolean G1value  =false;
boolean done  =false;
long G1out;

void setup()
{
  Wire.begin() ;
  Serial.begin(9600);
  Serial.println("i2c DAC V:1.0");
  // reserve 200 bytes for the inputString:
 inputString.reserve(200);
  pinMode(PWMDAC1pin, OUTPUT);
```

```
   Timer3.initialize(period);
  pinMode(13, OUTPUT);
  digitalWrite(13, LOW);    // set the LED off
  analogReference(EXTERNAL);
  value = (millivoltPlate*1023)/(vref*20);
   Set_Dac(value);// value not adjusted
  G1out= 1023 * millivoltG1 / vref;
  Timer3.pwm(PWMDAC1pin, G1out);
}

#define MCP4716_DEVICE 96

void
loop()
{
  if (Serial.available() > 0) {
    char incomingByte =Serial.read();
    // add it to the inputString:
   inputString += incomingByte;
//    digitalWrite(13, HIGH);    // set the LED on during input
    // if the incoming character is a newline, set a flag
    // so the main loop can do something about it:
    if (incomingByte =='\n') {
      stringComplete =true;
     inputString.toCharArray(buf,sizeof(inputString));
      buf[sizeof(inputString)+1] ='\0';
//    Serial.print(buf);
      if (G1value) {
       millivoltG1 = atoi(buf);
        Serial.println("G1: ");
        Serial.print(millivoltG1);
      }else{
       millivoltPlate = atoi(buf) * 1000;
        Serial.println("Plate: ");
        Serial.print(millivoltPlate);
      }
      Serial.println("mV");
       inputString ="";
    }
    }
  if (stringComplete){
    if (G1value) {
       G1value =false;
      G1out= 255 * millivoltG1 / vref;// 1023/4 (amplifier factor)
      Timer3.pwm(PWMDAC1pin, G1out);
    }else{
       G1value =true   ;// next time G1 value
      value = (millivoltPlate*1023)/(vref*20);// 10 bits DAC mV/20
     Set_Dac(value);
      digitalWrite(13, HIGH);    // set the LED on
      delay(10); // wait a bit
     millivoltPlateRead = ReadValue(analogPin);
     value = ( value * millivoltPlate ) / millivoltPlateRead;
     Set_Dac(value);
      digitalWrite(13, LOW);    // set the LED off
    }
     stringComplete =false;
    }
  time =millis();
  if ( time - prevtime > 200 && !done){
      prevtime  =  time;
    Serial.print("-");
    Serial.print(millivoltG1/1000);
    Serial.print(".");
    Serial.print((millivoltG1-(millivoltG1/1000)*1000)/100);
    Serial.print(",");
   millivoltPlateRead = ReadValue(analogPin);
   milliampPlateRead = ReadValue(analogPinA);
 // step Plate voltage
   millivoltPlate += Pstep;
    if (millivoltPlate > Pend) {
     millivoltPlate = Pstart;
      millivoltG1 += G1step;//step G1 voltge
      if (millivoltG1 > G1end) done =true;
      G1out= 255 * millivoltG1 / vref;// 1023/4 (amplifier factor)
      Timer3.pwm(PWMDAC1pin, G1out);
      delay(100); // wait a bit
    }
    uAPlate = (milliampPlateRead * 5) / 22 ;//milliV * 5 / 20 * 1.1 (Resistor f
    Serial.print(millivoltPlateRead);
    Serial.print(",");
    Serial.println(uAPlate);
    value = (millivoltPlate*1023)/(vref*20);// 10 bits DAC mV/20
   Set_Dac(value);
    digitalWrite(13, HIGH);    // set the LED on
    delay(10); // wait a bit
   millivoltPlateRead = ReadValue(analogPin);
   value = ( value * millivoltPlate ) / millivoltPlateRead;
   Set_Dac(value);
    delay(100); // wait a bit
    digitalWrite(13, LOW);    // set the LED off
  }
```

```
}

long ReadValue(int pin){
  long result;
  long integ;
  long rmin = 1000000;
  long rmax = 0;
  integ = 0;
  for (int i=0; i<102; i++){// some filtering by averaging and elim. min & max
     out =analogRead(pin);
    if (out < rmin) rmin = out;
    if (out > rmax) rmax = out;
    integ += out;
  }
    result = ((integ-rmin-rmax)*aref)/(1023*5);// Resistor factor 20;filter 100
    return result;
}

void  Set_Dac(long pvalue) {
  Wire.beginTransmission(MCP4716_DEVICE);
  Wire.write(64);                     // cmd to update the DAC
  Wire.write(pvalue     >>     2);// the 8 most significant bits...
  Wire.write((pvalue & 3) << 6);// the 2 least significant bits...
  Wire.endTransmission();
}
```