ESM Lab Grading Report

<u>Course Number:</u> 2
Module: 4: Lab 2 on Motor Voltage and Current Measurement

Lab Report Date 16th October 2018

Student Name(s) Rushi James Macwan and Poorn Mehta

We have both worked together with full efforts from both of us on this project in the best of our capacities and to the best of our extent to meet the lab assignment objectives.

Each Section of this Lab Report Counts for 20 points. Points will be allocated as follows:
20 points: section fully meets requirements of this rubric
15 points: section mostly meets requirements of this rubric
10 points: section meets roughly half the requirements of this rubric
5 points: section does not meet requirements, but shows a weak attempt
0 points: section blank

(A) Functional demonstration of your circuit to our TA. In this exercise, you schedule an appointment with your TA to show that your hardware functions as designed. For the Closed Loop C Motor Control lab, this will involve the following steps:
1. Show that all hardware is in place, and that your PSoC software can read the motor current, motor voltage and FET gate drive.
2. Alter the speed of the motor by allowing the motor to free spin or hold the motor shaft with your fingers. You should display the speed of the motor on the LCD in units of RPM.

If you are an on-campus student, then show your circuit to one of our TA's during office hours.

If you are a distance learning student, make an online appointment with your TA to demonstrate your work via Zoom meeting or other Web-based meeting tool. You can use the camera on your laptop PC or suitable plug-in webcam (Logitech etc.) to demonstrate a working circuit.

(B) Place photos here of your hardware setup, including PSoC board, connections to Oscilloscope or nScope, wiring, LCD Display, components. Label all components.

Image with all components labelled and the motor is running at full speed and no load.

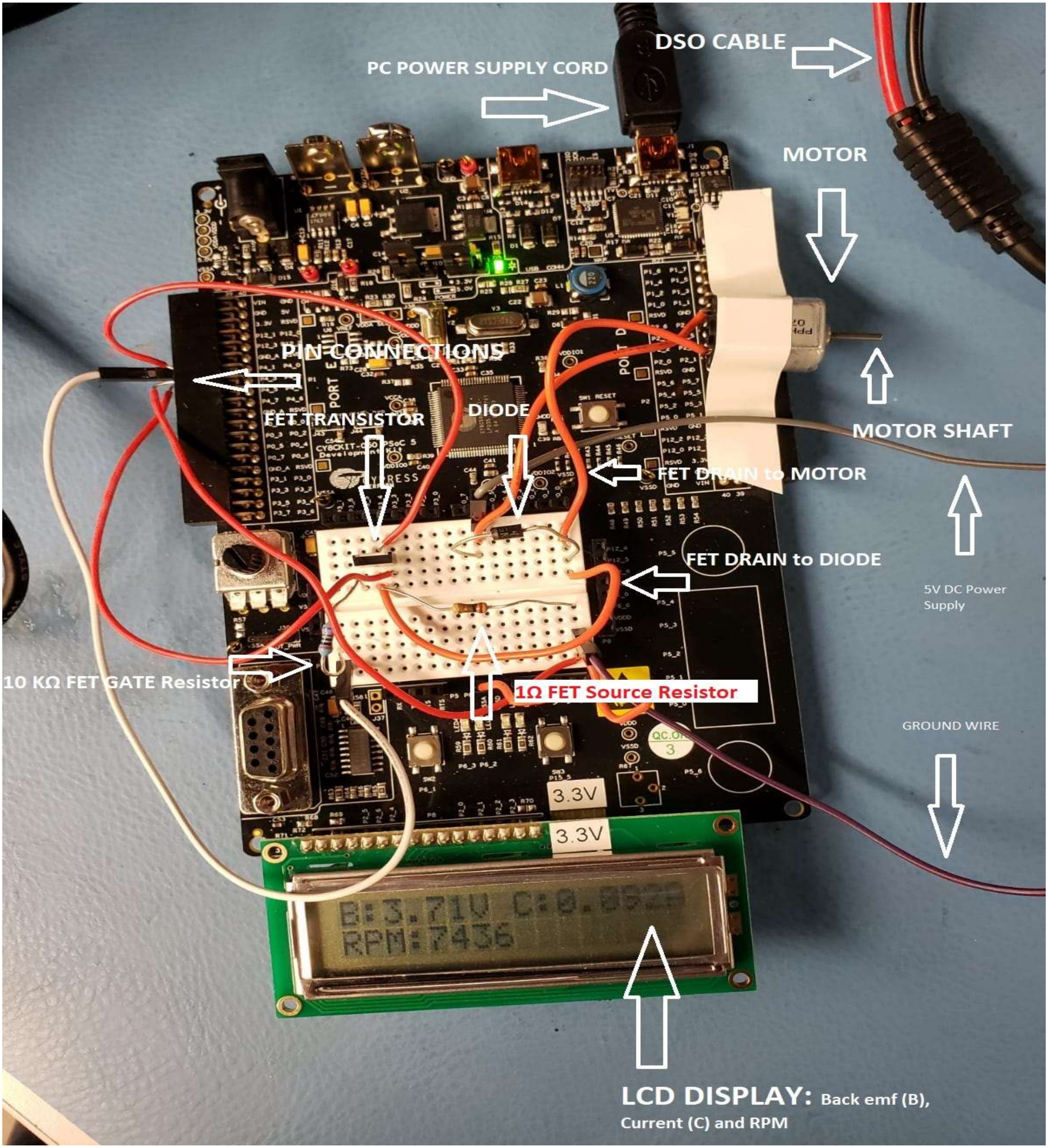


Image of the board when the motor shaft is given some load by trying to slow the RPM by touching it with hand.

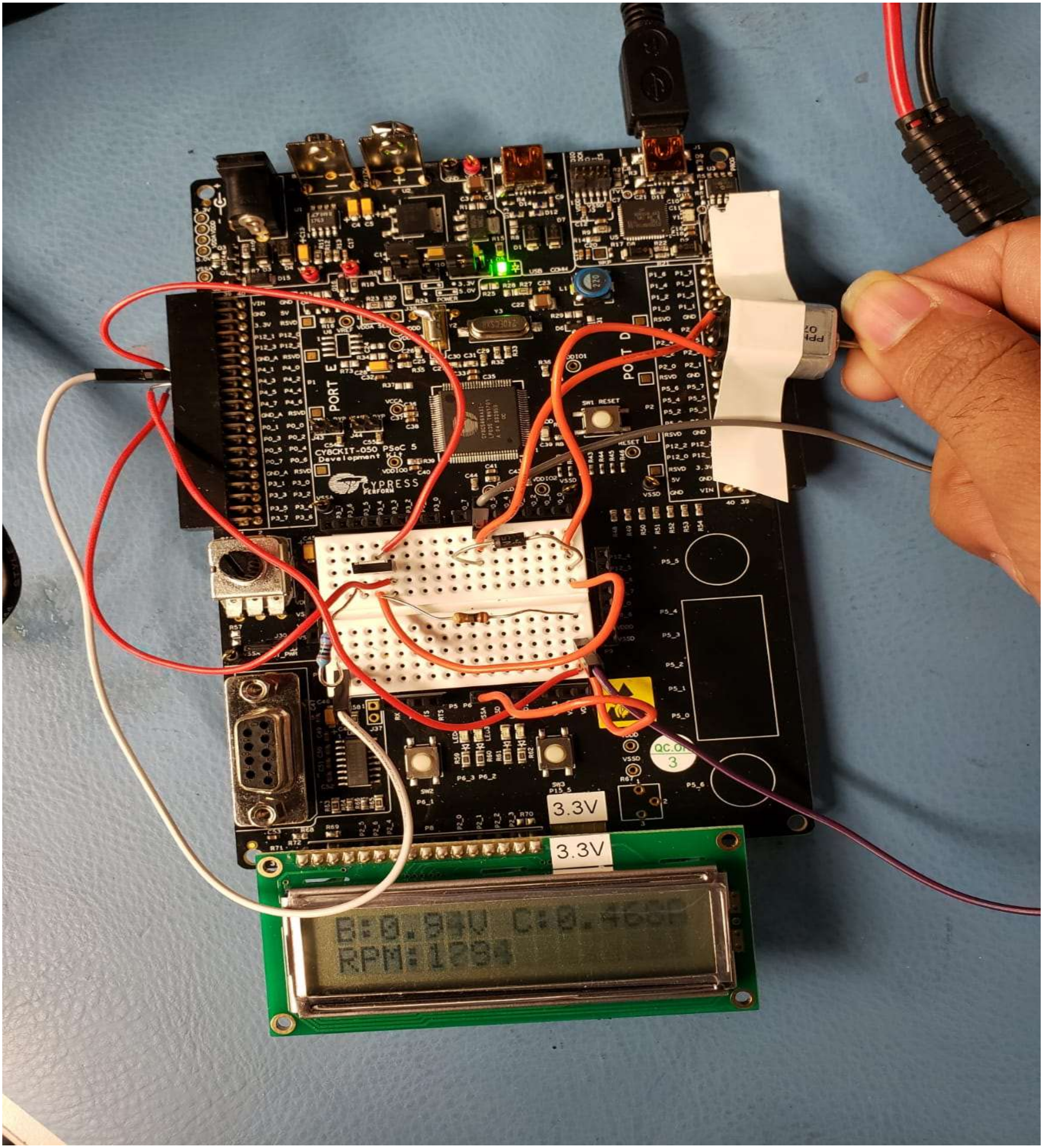
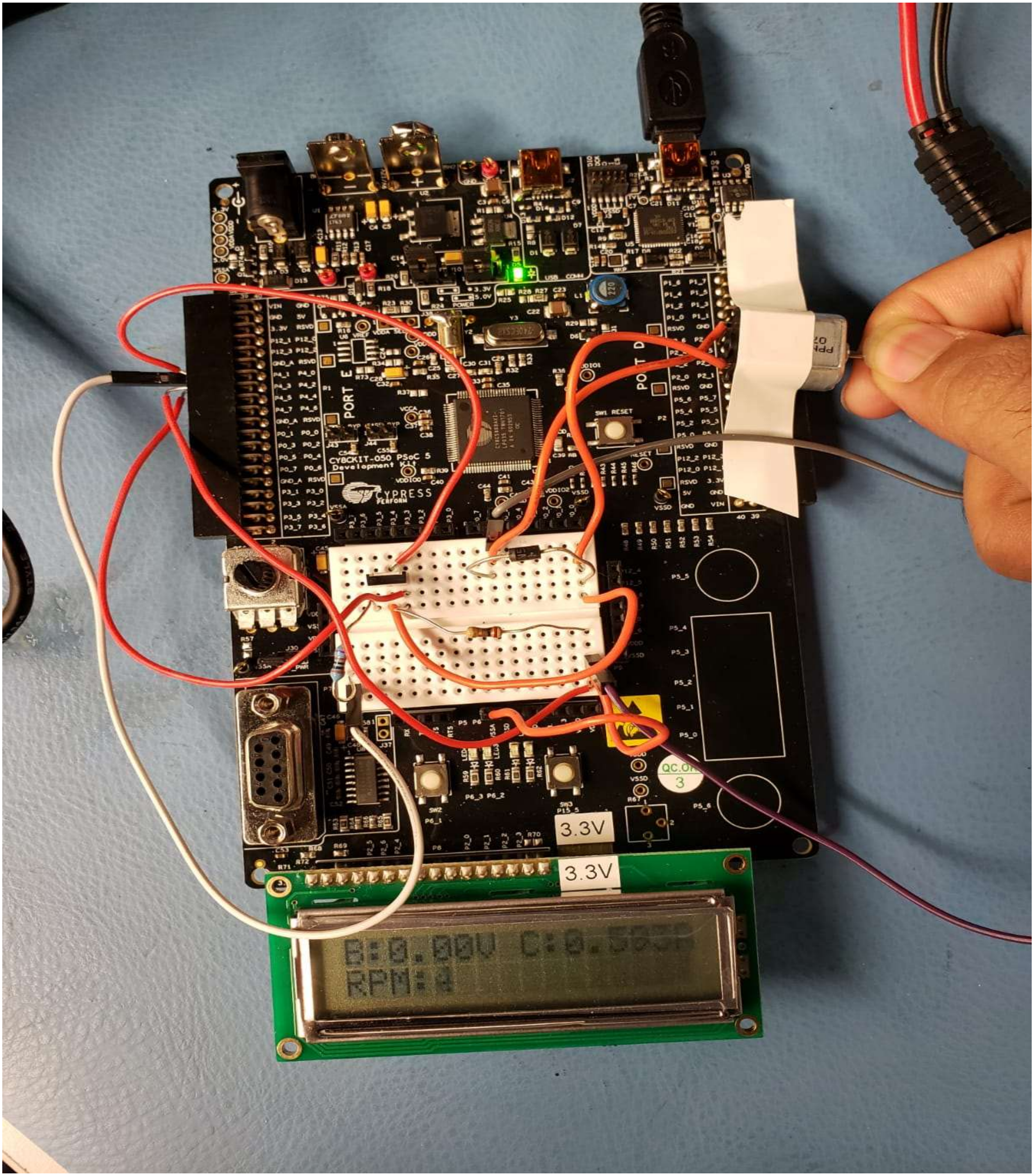
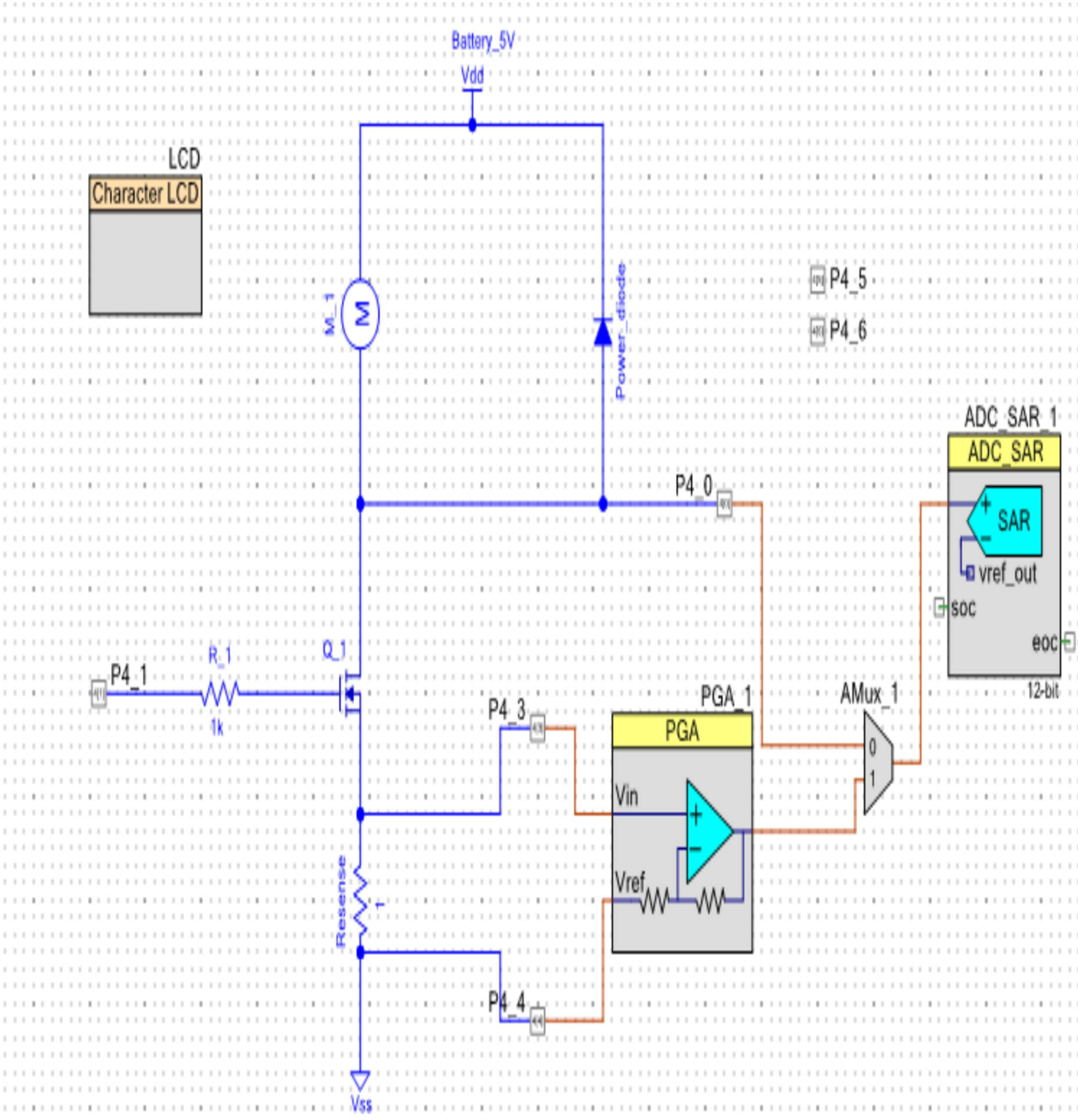


Image of the board when the motor shaft is given almost full-load with the hand.



(C) Place complete PSoC schematic here. This schematic must include internal components from the PSoC board (amplifier, MUX, etc.), as well as external components (thermistor, resistor, display).



(D) Complete PSoC software. This software must include calls to all internal functions, appropriate comments, and functional code that you included. We will not grade you on the exact syntax and structure, as there are numerous ways to structure the code and still provide the motor V & I measurement function. Instead, we will grade you on the completeness of the code relative to using the appropriate PSoC functions to gather the necessary data.

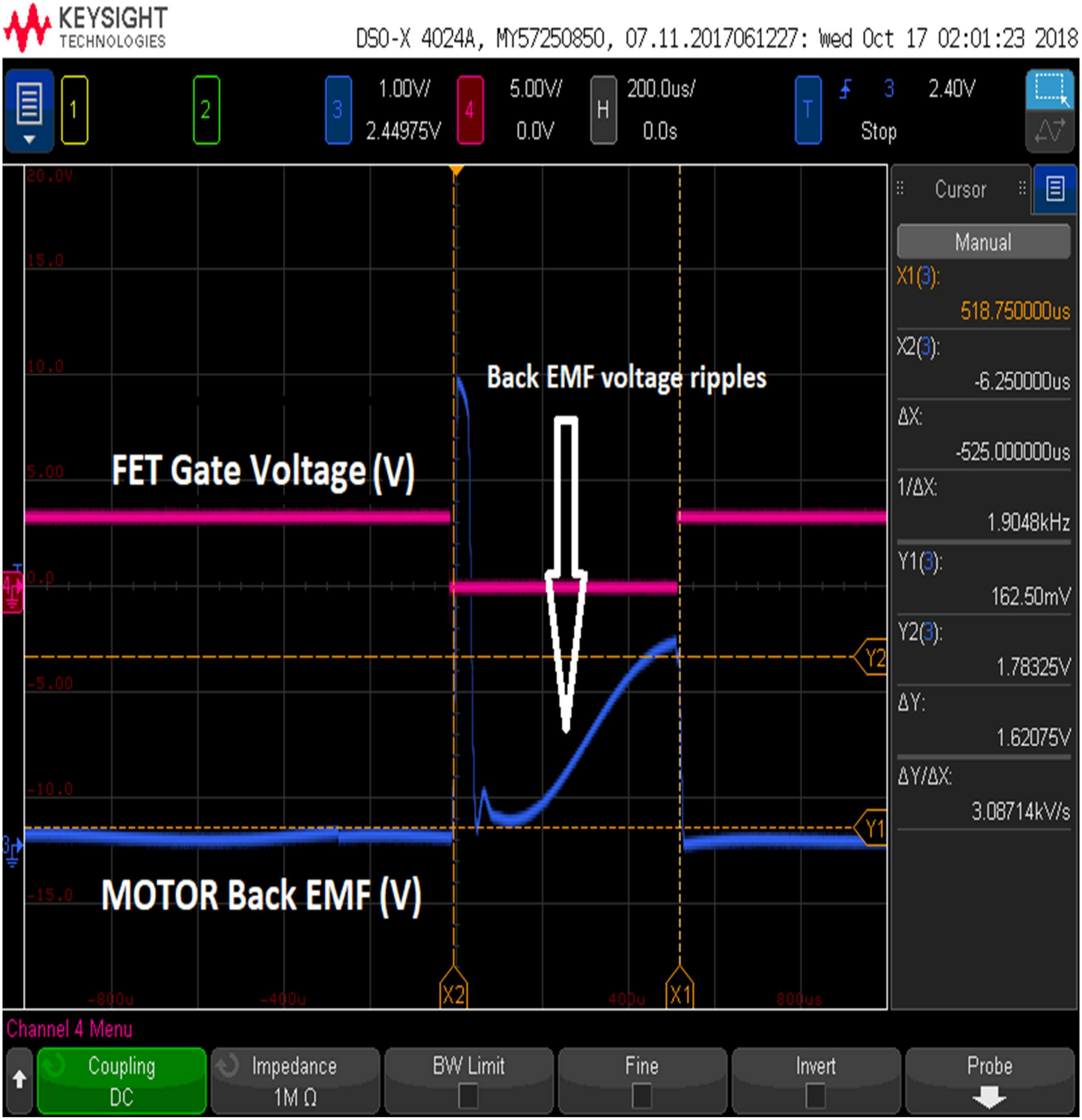
```
/* ========================================
 *
 * Copyright YOUR COMPANY, THE YEAR
 * All Rights Reserved
 * UNPUBLISHED, LICENSED SOFTWARE.
 *
 * CONFIDENTIAL AND PROPRIETARY INFORMATION
 * WHICH IS THE PROPERTY OF your company.
 *
 * ========================================
*/
#include "project.h"

#define Duty_Cycle  80

int main(void)
{
    CyGlobalIntEnable; /* Enable global interrupts. */

    LCD_Start();
    LCD_Position(0, 0);
    LCD_PrintString("Motor");

    ADC_SAR_1_Start();
//    ADC_SAR_2_Start();
    PGA_1_Start();
    AMux_1_Start();

    uint16_t Vdrop, Vcounts, Rdrop, Rcounts, Current, loop_counter, Constant, Vmotor, RPM;
    uint16_t Varray[30], Carray[30];
    uint32_t Vsum, Csum;
    loop_counter = 0;
    P4_1_Write(1);
    CyDelay(100);
    P4_1_Write(0);
    CyDelay(1);
```

```
    AMux_1_Select(0);
    ADC_SAR_1_StartConvert();
    ADC_SAR_1_IsEndConversion(ADC_SAR_1_WAIT_FOR_RESULT);
    Vcounts  = ADC_SAR_1_GetResult16();
    Vdrop = ADC_SAR_1_CountsTo_mVolts(Vcounts);
    Vmotor = 5000 - Vdrop;
    Constant = 14000 / Vmotor;
    while(1)
    {
      P4_1_Write(1);
      CyDelay((Duty_Cycle*10)/100);
      AMux_1_Select(1);
      ADC_SAR_1_StartConvert();
      ADC_SAR_1_IsEndConversion(ADC_SAR_1_WAIT_FOR_RESULT);
      Rcounts  = ADC_SAR_1_GetResult16();
      Rdrop = (ADC_SAR_1_CountsTo_mVolts(Rcounts));
      Current = Rdrop / 1;
      Carray[loop_counter] = Current;

      P4_5_Write(1);
      P4_1_Write(0);
      CyDelayUs(500);
      AMux_1_Select(0);
      ADC_SAR_1_StartConvert();
      ADC_SAR_1_IsEndConversion(ADC_SAR_1_WAIT_FOR_RESULT);
      Vcounts  = ADC_SAR_1_GetResult16();
      Vdrop = ADC_SAR_1_CountsTo_mVolts(Vcounts);
      Vmotor = 5000 - Vdrop;
      Varray[loop_counter] = Vmotor;
      P4_5_Write(0);
//      Constant = 14000 / Vmotor;
//      RPM = Constant*Vmotor;
//      CyDelay(((100-Duty_Cycle)*10)/100);

      loop_counter += 1;
      if(loop_counter > 29)
      {
        P4_6_Write(1);
        Vsum = 0;
        Csum = 0;
        for(uint8_t count = 0; count < 30; count ++)
        {
          Vsum += Varray[count];
          Csum += Carray[count];
        }
        Vsum /= 30;
```

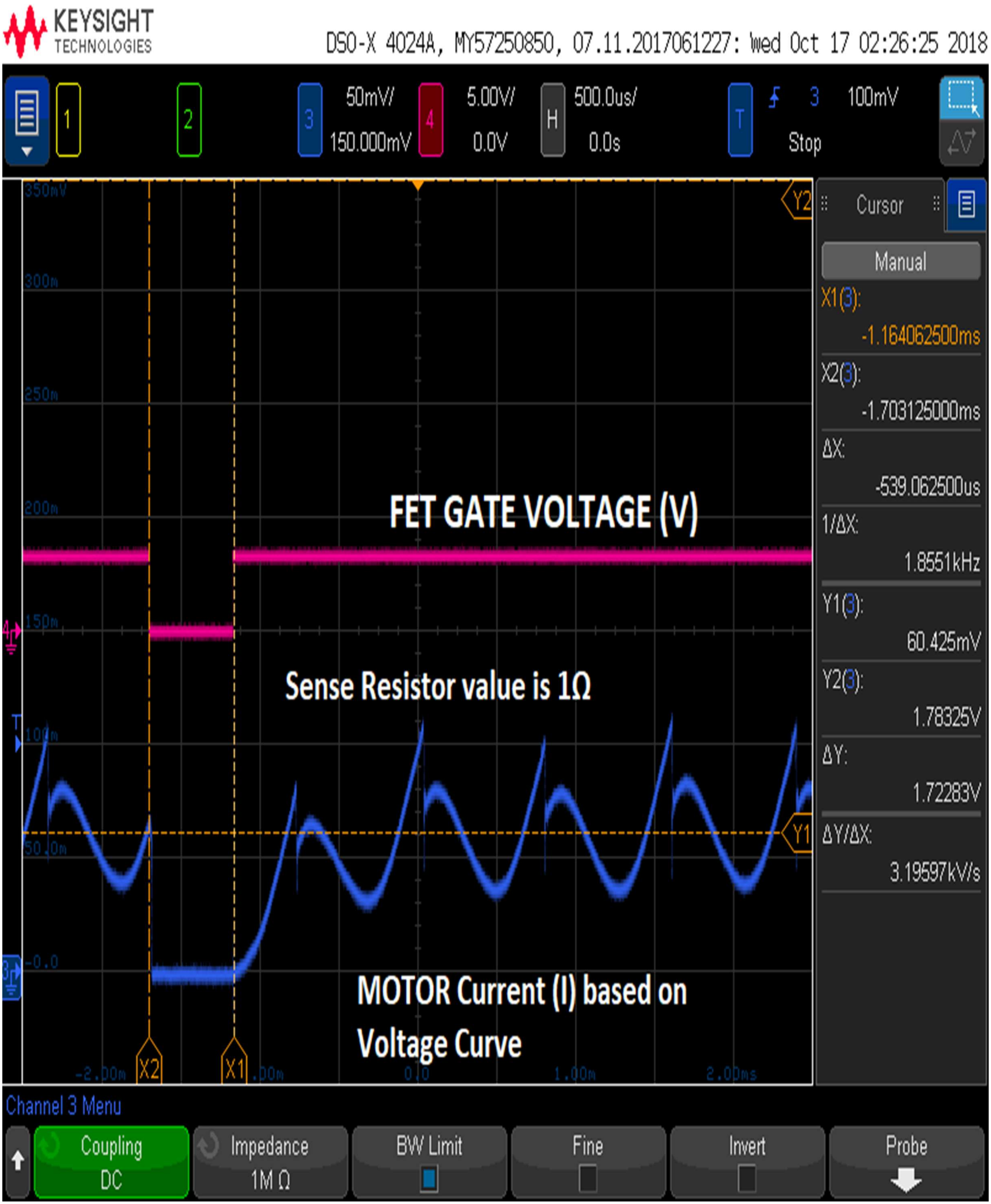
```
Csum /= 30;
Vmotor = (uint16_t)Vsum;
Current = (uint16_t)Csum;
RPM = Constant*Vmotor;
LCD_ClearDisplay();
loop_counter = 0;
LCD_Position(0, 0);
LCD_PrintString("B:");
LCD_Position(0, 2);
if(Vmotor < 1000) LCD_PrintString("0");
else LCD_PrintU32Number(Vmotor/1000);

LCD_Position(0, 3);
LCD_PrintString(".");
if((Vmotor%1000) < 10) LCD_PrintString("00");
else if((Vmotor%1000) < 100)
{
  LCD_PrintString("0");
  LCD_Position(0,5);
  LCD_PrintU32Number((Vmotor%1000)/10);
}
else
{
   LCD_Position(0,4);
   LCD_PrintU32Number((Vmotor%1000)/10);
}
LCD_Position(0,6);
LCD_PrintString("V");


LCD_Position(0, 8);
LCD_PrintString("C:");

LCD_Position(0, 10);
if(Current < 1000) LCD_PrintString("0");
else LCD_PrintU32Number(Current/1000);

LCD_Position(0, 11);
LCD_PrintString(".");
if((Current%1000) < 10)
{
   LCD_PrintString("00");
   LCD_Position(0, 14);
   LCD_PrintU32Number(Current%1000);
}
else if((Current%1000) < 100)
```

```
        {
          LCD_PrintString("0");
          LCD_Position(0,13);
          LCD_PrintU32Number(Current%1000);
        }
        else
        {
           LCD_Position(0,12);
           LCD_PrintU32Number(Current%1000);
        }
        LCD_Position(0,15);
        LCD_PrintString("A");

        LCD_Position(1,0);
        LCD_PrintString("RPM:");
        LCD_Position(1,4);
        LCD_PrintU32Number(RPM);
        P4_6_Write(0);
      }
    }
}

/* [] END OF FILE */
```

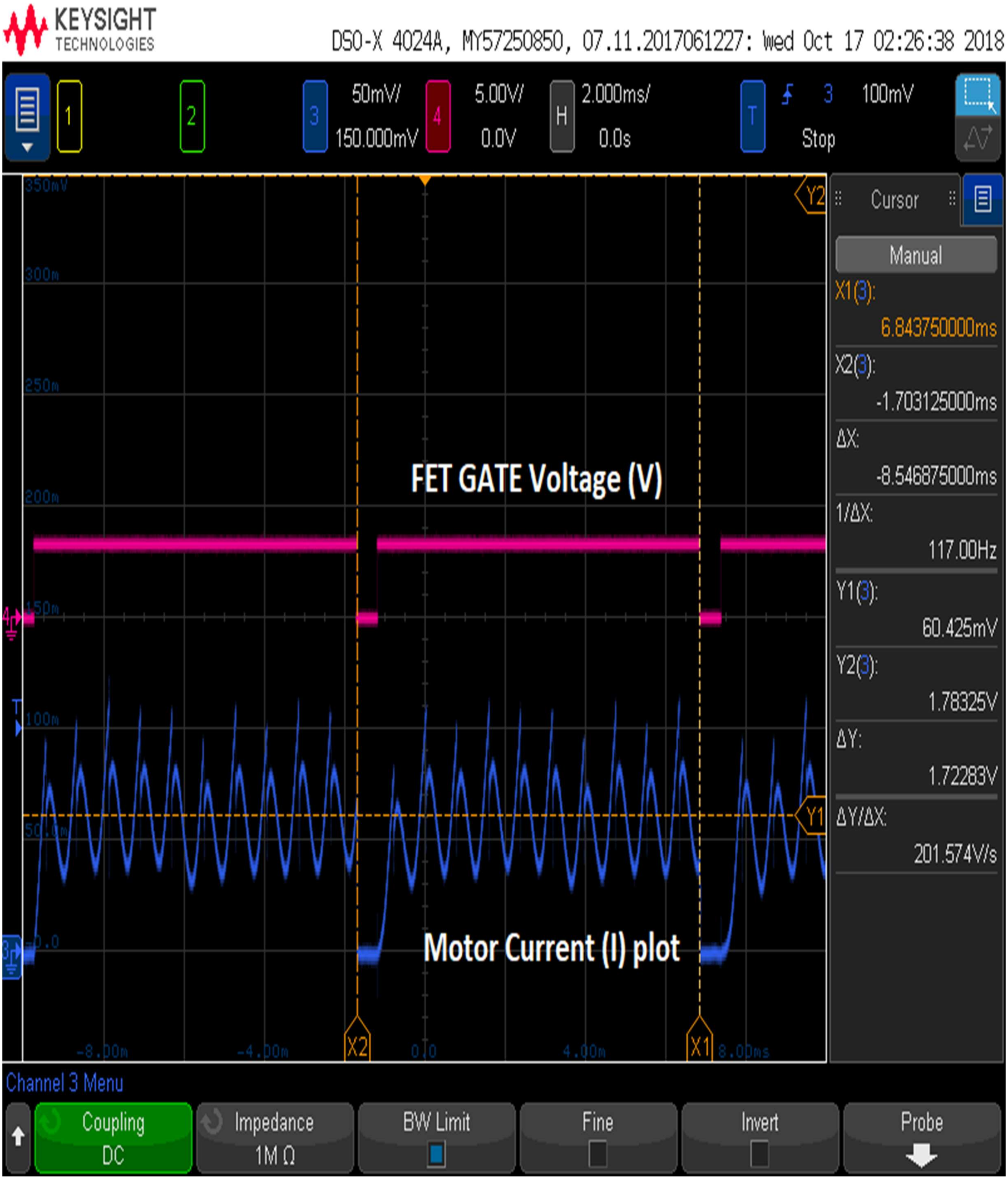
E) Place screenshots from your oscilloscope or nScope showing critical loop times or signal outputs. For this lab, you should show plots of the motor current and voltage on one channel, and the FET gate on the other, similar to what is shown in the sample below.

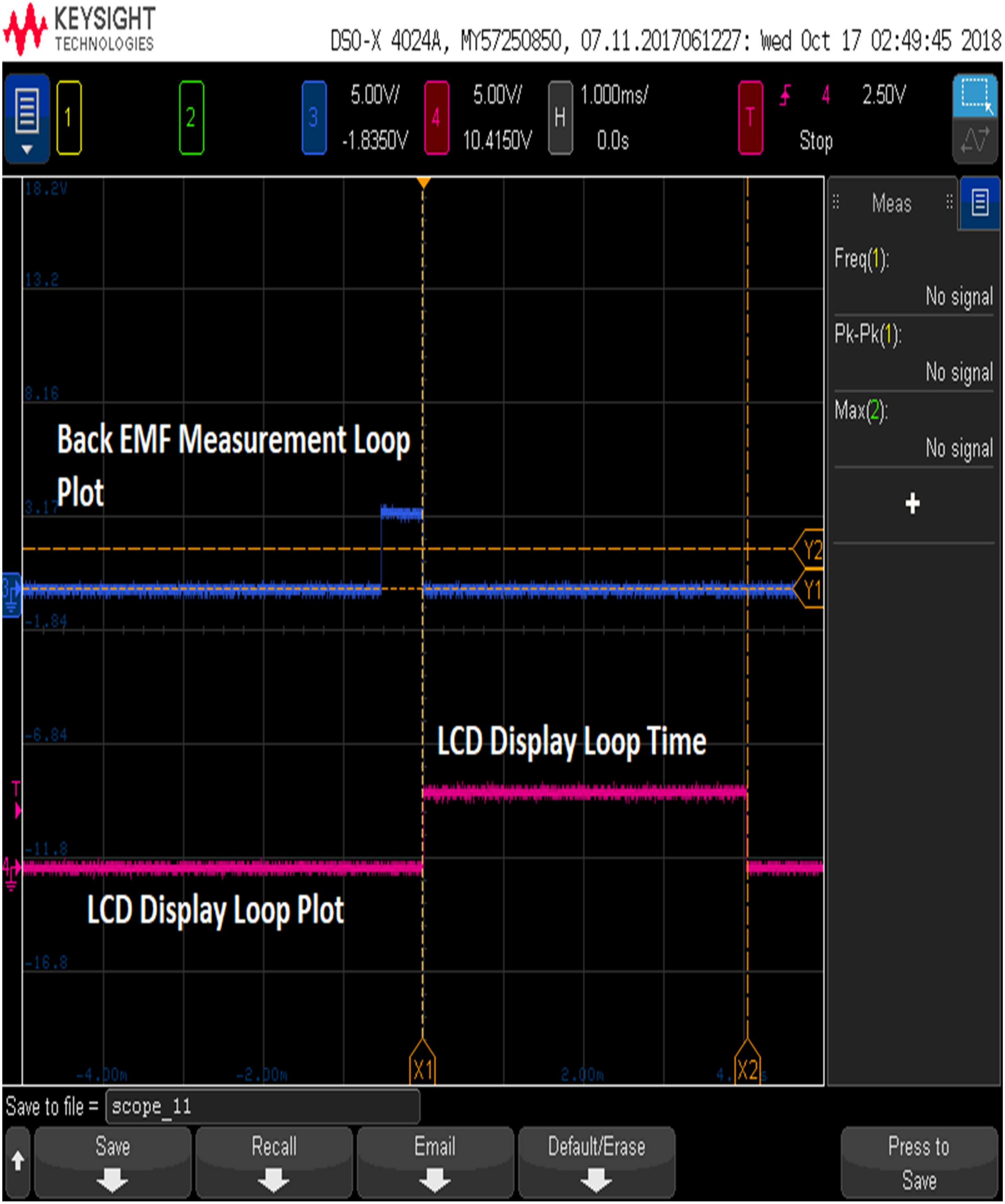


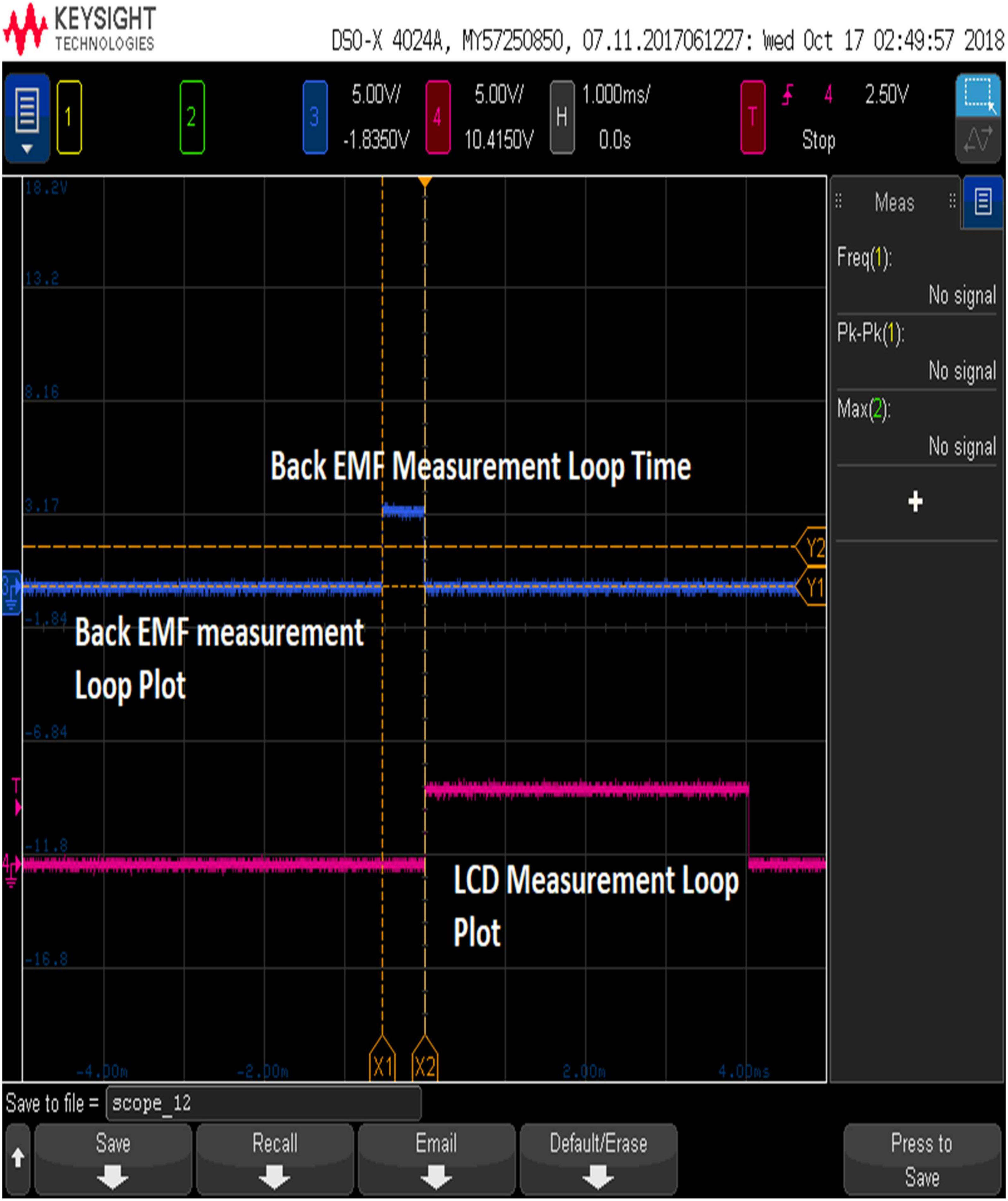
KEYSIGHT TECHNOLOGIES
DSO-X 4024A, MY57250850, 07.11.2017061227: Wed Oct 17 02:01:23 2018
1.00V/
2.44975V
5.00V/
0.0V
200.0us/
0.0s
2.40V
Stop
Cursor
Manual
X1(3):
518.750000us
X2(3):
-6.250000us
ΔX:
-525.000000us
1/ΔX:
1.9048kHz
Y1(3):
162.50mV
Y2(3):
1.78325V
ΔY:
1.62075V
ΔY/ΔX:
3.08714kV/s
Back EMF voltage ripples
FET Gate Voltage (V)
MOTOR Back EMF (V)
Channel 4 Menu
Coupling
DC
Impedance
1M Ω
BW Limit
Fine
Invert
Probe

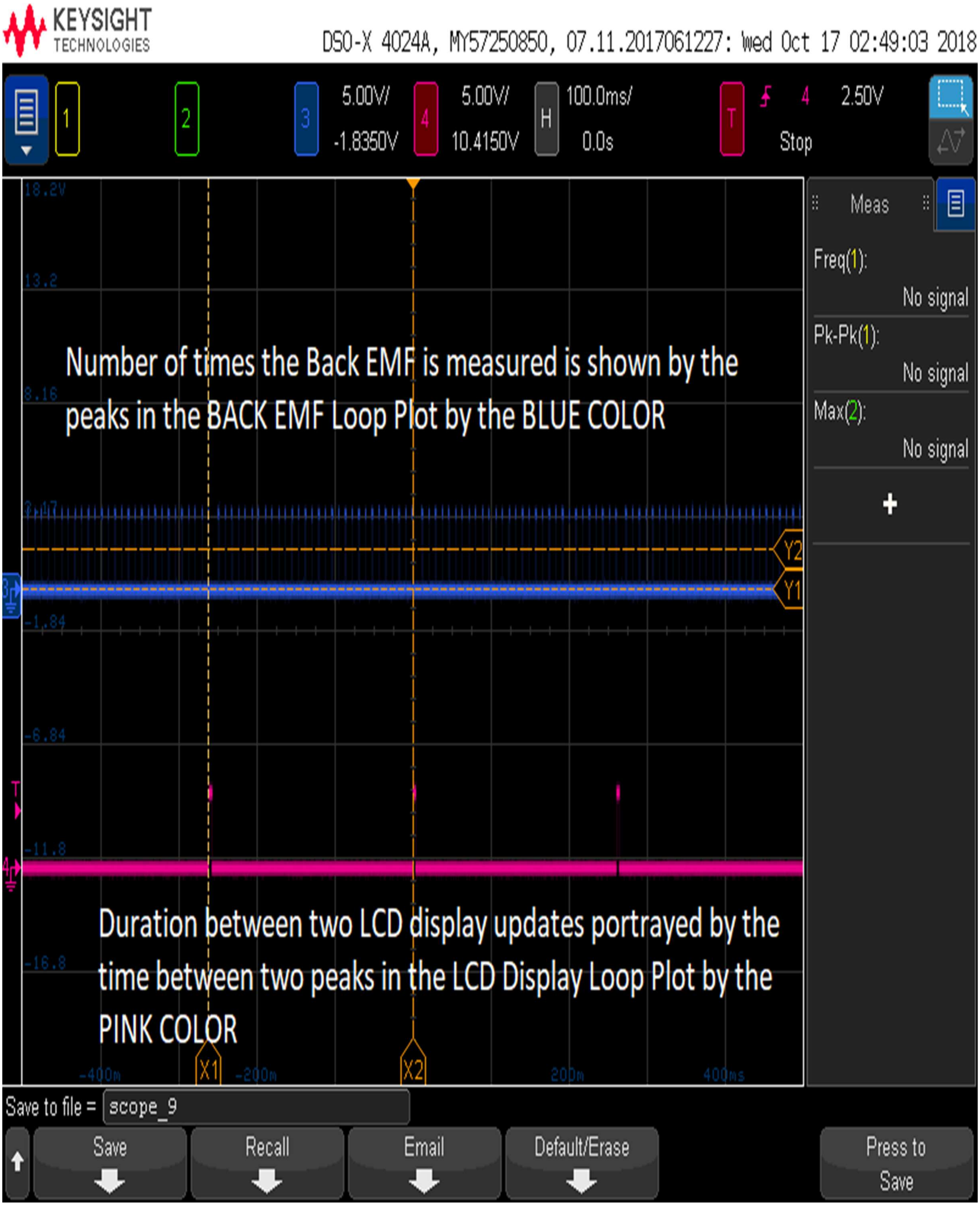


KEYSIGHT TECHNOLOGIES
DSO-X 4024A, MY57250850, 07.11.2017061227: Wed Oct 17 02:26:25 2018
50mV/
150.000mV
5.00V/
0.0V
500.0us/
0.0s
100mV
Stop
Cursor
Manual
X1(3):
-1.164062500ms
X2(3):
-1.703125000ms
ΔX:
-539.062500us
1/ΔX:
1.8551kHz
Y1(3):
60.425mV
Y2(3):
1.78325V
ΔY:
1.72283V
ΔY/ΔX:
3.19597kV/s
FET GATE VOLTAGE (V)
Sense Resistor value is 1Ω
MOTOR Current (I) based on
Voltage Curve
Channel 3 Menu
Coupling
DC
Impedance
1M Ω
BW Limit
Fine
Invert
Probe


KEYSIGHT TECHNOLOGIES
DSO-X 4024A, MY57250850, 07.11.2017061227: Wed Oct 17 02:26:38 2018
50mV/
150.000mV
5.00V/
0.0V
2.000ms/
0.0s
100mV
Stop
Cursor
Manual
X1(3):
6.843750000ms
X2(3):
-1.703125000ms
ΔX:
-8.546875000ms
1/ΔX:
117.00Hz
Y1(3):
60.425mV
Y2(3):
1.78325V
ΔY:
1.72283V
ΔY/ΔX:
201.574V/s
FET GATE Voltage (V)
Motor Current (I) plot
Channel 3 Menu
Coupling
DC
Impedance
1M Ω
BW Limit
Fine
Invert
Probe


KEYSIGHT TECHNOLOGIES
DSO-X 4024A, MY57250850, 07.11.2017061227: Wed Oct 17 02:49:45 2018
5.00V/
-1.8350V
5.00V/
10.4150V
1.000ms/
0.0s
2.50V
Stop
Meas
Freq(1):
No signal
Pk-Pk(1):
No signal
Max(2):
No signal
Back EMF Measurement Loop Plot
LCD Display Loop Time
LCD Display Loop Plot
Save to file = scope_11
Save
Recall
Email
Default/Erase
Press to Save


KEYSIGHT TECHNOLOGIES
DSO-X 4024A, MY57250850, 07.11.2017061227: Wed Oct 17 02:49:57 2018
5.00V/
-1.8350V
5.00V/
10.4150V
1.000ms/
0.0s
2.50V
Stop
Meas
Freq(1):
No signal
Pk-Pk(1):
No signal
Max(2):
No signal
Back EMF Measurement Loop Time
Back EMF measurement
Loop Plot
LCD Measurement Loop
Plot
Save to file = scope_12
Save
Recall
Email
Default/Erase
Press to Save


KEYSIGHT TECHNOLOGIES
DSO-X 4024A, MY57250850, 07.11.2017061227: Wed Oct 17 02:49:03 2018
5.00V/
-1.8350V
5.00V/
10.4150V
100.0ms/
0.0s
2.50V
Stop
Meas
Freq(1):
No signal
Pk-Pk(1):
No signal
Max(2):
No signal
Number of times the Back EMF is measured is shown by the peaks in the BACK EMF Loop Plot by the BLUE COLOR
Duration between two LCD display updates portrayed by the time between two peaks in the LCD Display Loop Plot by the PINK COLOR
Save to file = scope_9
Save
Recall
Email
Default/Erase
Press to Save