VFD Motor Control

Mackenzie Miller

Andrew Nguyen

Aidan Rader

Ryan Regan

**Subsystem Report**

REVISION – 0

5 December 2024

Subsystem Report

for

VFD Motor Control

Prepared by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Author Date

Approved by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Leader Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

John Lusher, P.E. Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

T/A Date

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rev.** | **Date** | **Originator** | **Approvals** | **Description** |
| **0** | 12/5/2024 | VFD Motor Control Team |  | Draft Release |

Table of Contents

[Table of Contents III](#_Toc184333838)

[List of Tables IV](#_Toc184333839)

[List of Figures V](#_Toc184333840)

[1. Introduction 1](#_Toc184333841)

[2. Optoelectronics Subsystem Report 2](#_Toc184333842)

[2.1. Subsystem Introduction 2](#_Toc184333843)

[2.2. Subsystem Details 2](#_Toc184333844)

[2.3. Subsystem Validation 3](#_Toc184333845)

[2.4. Subsystem Conclusion 4](#_Toc184333846)

[3. Microcontroller Subsystem Report 5](#_Toc184333847)

[3.1. Subsystem Introduction 5](#_Toc184333848)

[3.2. Subsystem Details 5](#_Toc184333849)

[3.3. Subsystem Validation 7](#_Toc184333850)

[3.4. Subsystem Conclusion 7](#_Toc184333851)

[4. Power Electronics Subsystem Report 8](#_Toc184333852)

[4.1. Subsystem Introduction 8](#_Toc184333853)

[4.2. Subsystem Validation 8](#_Toc184333854)

[4.2.1. Rectifier and DC Link Validation 8](#_Toc184333855)

[4.2.2. Power Control Validation 12](#_Toc184333856)

[4.2.3. 3.3 V to Iso-5 V Power Converter Validation 15](#_Toc184333857)

[4.2.4. 15 V to Iso-15 V Power Converter Validation 18](#_Toc184333858)

[4.3. Subsystem Conclusion 21](#_Toc184333859)

[5. Firmware Subsystem Report 22](#_Toc184333860)

[5.1. Subsystem Introduction 22](#_Toc184333861)

[5.2. Subsystem Details 22](#_Toc184333862)

[5.3. Subsystem Validation 22](#_Toc184333863)

[5.4. Subsystem Conclusion 24](#_Toc184333864)

List of Tables

[*Table 1: Pin interface for DSPIC33CK256MP508 Microcontroller* 7](#_Toc184333837)

List of Figures

[*Figure 1: Power to Digital Schematic* 3](#_Toc184335656)

[*Figure 2: Digital to Power Schematic* 3](#_Toc184335657)

[*Figure 3: Block diagram of the microcontroller subsystem* 6](#_Toc184335658)

[*Figure 4: Rectifier Schematic* 8](#_Toc184335659)

[*Figure 5: DC Link Schematic* 9](#_Toc184335660)

[*Figure 6: Rectifier and DC Link Circuit* 10](#_Toc184335661)

[*Figure 7: Input Phase A and Phase B* 10](#_Toc184335662)

[*Figure 8: Input Phase C* 11](#_Toc184335663)

[*Figure 9: Input Phase A, Phase B, Phase C; Output DC* 12](#_Toc184335664)

[*Figure 10: Power Control Circuit* 13](#_Toc184335665)

[*Figure 11: Power Control Output* 13](#_Toc184335666)

[*Figure 12: Single Phase PWM Signals* 14](#_Toc184335667)

[*Figure 13: Input Voltage Power Control* 15](#_Toc184335668)

[*Figure 14: 3.3V/ISO\_5V Converter and 15V/ISO\_5V Converter Schematics* 16](#_Toc184335669)

[*Figure 15: 3.3V/ISO\_5V Converter Circuit* 16](#_Toc184335670)

[*Figure 16: Input 3.3 V* 17](#_Toc184335671)

[*Figure 17: Input 3.3 V, Output Iso 5 V* 18](#_Toc184335672)

[*Figure 18: 15V/ISO\_15V Converter Circuit* 19](#_Toc184335673)

[*Figure 19: Input 15V* 19](#_Toc184335674)

[*Figure 20: Input 15 V, Output Iso 15 V* 20](#_Toc184335675)

[*Figure 21: Oscilloscope Output of PWM Signals Produced by Development Board* 23](#_Toc184335676)

[*Figure 22: Debug Print Statements (Potentiometer at 0)* 24](#_Toc184335677)

# Introduction

The VFD motor controller is designed to regulate the speed and torque of an AC motor by adjusting the frequency and voltage it receives. This project utilizes three-phase power as the input, which will be converted to DC and transmitted a microcontroller to power the motor. The microcontroller will also send and receive signals, via optoelectronics, to the DC link to deliver the user-specified frequency set through the user interface. The VFD is intended to enhance efficiency and conserve energy.

# Optoelectronics Subsystem Report

## Subsystem Introduction

The optoelectronics subsystem of the VFD serves as the go between for the microcontroller and the motor and power electronics. The microcontroller and high voltage side cannot come into contact with one another because the microcontroller operates at 3.3V, and the power electronics and motor operate at a maximum of 208 VAC. The optoelectronics help transport voltages from the power control to the microcontroller and PWMs from the microcontroller to the power control.

## Subsystem Details

The power to digital side includes three circuits that convert whatever voltages are received to around 15V so the microcontroller can process them. The first circuit takes the output of the DC rectifier, the second circuit is for the current and voltage monitoring, and the third circuit is for temperature control. These three measurements are essentially transported across an isolation barrier using light via emitters and receivers made into an integrated circuit. Below, in Figure 1, one of the circuits can be seen.

There are also three circuits that take pulse width modules (PWMs) in and transport them across another isolation barrier. The first is the PWM high signal for each of the three phases, the second is the low signal for each phase, and the third takes interrupt signals from the microprocessor and transports them to the power control. This segment is necessary in order to maintain the integrity of separating the power and digital sides. The schematic of this circuit can be seen in below in Figure 2.

There will be a tachometer that is attached directly to the motor, and that is sent to the microcontroller which is to be displayed on the user interface. The tachometer is used to measure the revolutions per minute (RPMs) of the motor.

These circuits are executed on a printed circuit board (PCB). There are a total of seven parts to this PCB design: the three power to digital circuits, the three digital to power circuits, and the tachometer. This yields a very compact PCB with a lot of hardware components.

A diagram of a circuit

Description automatically generated

*Figure 1: Power to Digital Schematic*

A diagram of a circuit

Description automatically generated

*Figure 2: Digital to Power Schematic*

## Subsystem Validation

In preliminary testing, it appeared that pin 8 of one of the 8 pin connectors had been damaged either due to solder or another reason. This led the PCB to behave very poorly when powered up since the damaged pin was PGND, the ground of the power side. This connector was removed from the board and switched with an identical connector on the other side that does not use pin 8, and the problem was solved. Additionally, one of the digital isolation packages seems to also be malfunctioning, in that it does not work. The other two packages and circuits work correctly by transporting the voltage across the barrier, but the first one seems to drop all voltage. It is most likely soldered incorrectly or was damaged in the soldering process. For testing purposes, when 5V is supplied to the power circuits, there is around 0.5V on the other side of the isolation barrier. While this could not be tested at full power because of limited equipment at the moment, there is voltage on both sides of the isolation barrier which means that the circuit works. For the digital to power side, again, there is voltage on both sides of the isolation, but the voltage across the isolation IC is zero, meaning that the voltage is being transported via optoelectronics. This indicates that the PCB is designed correctly, and barring a few components that need to be replaced, it works appropriately.

## Subsystem Conclusion

Overall, the circuits work as expected. There will be another PCB designed and assembled in the near future in order to improve the design and replace any broken parts. Another step in the near future is to implement the tachometer onto the motor and plug it into the PCB once the motor becomes available to use.

# Microcontroller Subsystem Report

## Subsystem Introduction

This subsystem refers to the hardware of the microcontroller portion of the VFD motor controller. In order for the firmware to control the motor, the microcontroller must be implemented properly. This subsystem is powered by two of the three phase AC power supply which is sent to the MCU for functionality.

## Subsystem Details

The DSPIC33CK256MP508 microcontroller was selected for this project. It has an adequate amount of analog and digital pins to support the VFD motor and the required PWM signals necessary. This MCU has 80 pins however only 32 are needed for this application. The MCU is supplied by the 15 VDC from the AC power supply fed by the main power. The voltage will then be stepped down to a usable 3.3V using a 3.3V fixed buck converter. The buck converter takes the 15V from the AC/DC converter to a usable voltage for the microcontroller. This 3.3V is needed for the microcontroller along with the Pickit4 debugger and the UART serial interfaces.

The MCU subsystem receives feedback through low-voltage analog signals representing voltage and current. The MCU then sends out PWM signals to the H bridge and power control system which are used to control the inverter stage of the VFD which helps to adjust the output voltage and frequency supplied to the motor.

The hardware necessary for the application is placed onto a PCB and the firmware will be coded onto the MCU to control the PWM signals. A potentiometer is included to vary the frequency of these PWM signals. A push button will also be used to turn the motor on and off which is indicated by a green LED. A 5 pin connector is used to connect to the Pickit4 debugger which is used to code onto the MCU. A 3 pin connector is used for UART serial communication using a cable to connect to a PC. Finally, a 20 pin connector is used to send the PWM signals back to the H bridge and to connect to the GPIO pins. This PCB will later be connected to the power and optoelectronics subsystems.

A diagram of a computer system

Description automatically generated

*Figure 3: Block diagram of the microcontroller subsystem*

|  |  |
| --- | --- |
| **Pin** | **Function** |
| 1 | PWM1H |
| 2 | GPIO |
| 3 | PWM1L |
| 4 | GPIO |
| 9 | MCLR |
| 11,32,50,70 | VSS |
| 12,31,51,71 | VDD |
| 17,19,22,39,42,44 | GPIO |
| 24 | GPIO (LED 2) |
| 25 | AVDD |
| 26 | AVss |
| 34 | Clock input |
| 37 | GPIO (LED 1) |
| 60 | PGD1 |
| 61 | PGC1 |
| 68 | UART TX |
| 69 | UART RX |
| 75 | PWM3H |
| 76 | PWM3L |
| 78 | PWM2H |
| 80 | PWM2L |

*Table* 1*: Pin interface for DSPIC33CK256MP508 Microcontroller*

## Subsystem Validation

The microcontroller subsystem was validated to meet all the requirements. The process for validation went as follows: the buck converter was tested by applying 15V using a DC supply and running it through to the MCU to ensure that the proper 3.3V was powering the PCB board. A code was also to be tested to ensure that the MCU was responding correctly to the GPIO pins using MPLAB X IDE. The results of the validation is as follows: the 3.3V was not properly sent to the MCU, a probable reason for this is an error that was found in the schematic which ultimately led to improper routing. The buck converter did step down the voltage but not to the 3.3V as intended.  This also prohibited the use of the Pickit4 debugger as it needs the 3.3V to operate. However, the curiosity board – which also uses the DSPIC33CK256MP508 MCU – was used to test the code and it was able to function properly and respond to the correct GPIO pins.

## Subsystem Conclusion

The buck converter portion of the subsystem was not functioning properly, leading to problems interfacing with the microcontroller. The correct pins are used on the microcontroller and were able to be confirmed using the curiosity board which uses the same microcontroller. Going forward, before the next semester, the schematic and PCB routing will need to be updated. The buck converter portion will need to be fixed, and the rest will be checked as well to ensure the same mistake does not occur again. A new PCB board will need to be printed and rebuilt to continue and have a fully functioning subsystem for integration.

# Power Electronics Subsystem Report

## Subsystem Introduction

The power subsystem of the VFD motor control is responsible for supplying power to both an AC induction motor and the entire system. It is designed to operate with 208 VAC three-phase power and has been tested to ensure safe and reliable performance. The following sections present an analysis of the subsystem's operation and validation.

## Subsystem Validation

### Rectifier and DC Link Validation

A rectifier converts AC to DC, while a DC link filters the output. In Figure 4, the rectifier schematic is shown. First, a power input connector receives the desired three-phase power. Next, a set of capacitors in parallel with varistors is connected across each phase. The capacitors filter the voltage to reduce noise, and the varistors provide protection against voltage surges. Finally, a full-wave rectifier converts the AC to DC.

*Figure 4: Rectifier Schematic*

A diagram of electrical wiring

Description automatically generated

In Figure 5, the dc link schematic is shown. First, a fuse provides protection against overcurrent. Next, a varistor in parallel with a set of capacitors is connected to ground. The varistor provides protection against voltage surges, and the capacitors filter the voltage to reduce noise.

A diagram of a circuit

Description automatically generated

*Figure 5: DC Link Schematic*

The rectifier and DC link were validated by supplying an input AC voltage and measuring the output DC voltage, as shown in Figure 6. While the integrated system is designed to operate at 208 VAC, the validation testing was conducted at 10 VAC. The three-phase AC was generated using three waveform generators, with phase A and phase B values in Figure 7 and phase C values in Figure 8. An amplitude of 3.536 Vrms was used to produce the desired voltage. Additionally, each phase was offset by 120˚ to simulate a three-phase system, with phase C further adjusted due to the waveform generators not being synchronized.

A circuit board with wires

Description automatically generated

*Figure 6: Rectifier and DC Link Circuit*

A screen shot of a computer

Description automatically generated

*Figure 7: Input Phase A and Phase B*

A screen shot of a computer

Description automatically generated

*Figure 8: Input Phase C*

Upon measuring the input AC waveforms shown in Figure 9, the green (phase A), blue (phase B), and purple (phase C) sinusoidal waves demonstrate a phase shift of 120 degrees and an amplitude of 10 VAC. Furthermore, the measured DC output is 9.6 VDC with 600 mV of noise. This differs slightly from the expected output of 12 VDC .This discrepancy can be attributed to the forward voltage drop across the rectifier diodes and losses within the rectifier and DC link circuitry. Despite these factors, the measured value remains within an acceptable range for the system's operation.

A screen shot of a computer

Description automatically generated

*Figure 9: Input Phase A, Phase B, Phase C; Output DC*

### Power Control Validation

The power control takes an input of DC voltage from the DC Link and, based on the incoming PWM signals from the microcontroller, inverts the DC voltage into the appropriate AC voltage to power the motor.

Unfortunately, the only control was short-circuited without enough time for a new unit to be delivered, and thus this portion of the validation is not complete. The validation would consist of applying an input dc voltage and PWM signals and measuring an output waveform. In Figure 10, the circuit is shown, and in Figure 11, the output waveforms are shown to be approximately zero, aligning with the chip not working. In Figure 12, the high and low PWM signals for a single phase were generated for full validation. Furthermore, Figure 13, is the measured input power supply voltages show the power flow of current through the circuitry surrounding the power control and that the only short circuit is the power control itself.

A green circuit board with many wires

Description automatically generated

*Figure 10: Power Control Circuit*

*A screen shot of a computer

Description automatically generated*

*Figure 11: Power Control Output*

A screen shot of a graph

Description automatically generated

*Figure 12: Single Phase PWM Signals*

A screen shot of a graph

Description automatically generated

*Figure 13: Input Voltage Power Control*

### 3.3 V to Iso-5 V Power Converter Validation

The 3.3 V to Iso-5 V power converter converts 3.3 VDC to isolated 5 VDC. In Figure 14, the 3.3V/ISO\_5V converter schematic is shown. First, an input connector receives the input 3.3 V. Next, a set of capacitors in parallel is connected to ground to filter the voltage and reduce noise. Then, a power converter steps up and isolates the 3.3 V to 5 V. Finally, another set of capacitors in parallel is connected to ground to filter the voltage and reduce noise.

A diagram of a computer system

Description automatically generated

*Figure 14: 3.3V/ISO\_5V Converter and 15V/ISO\_5V Converter Schematics*

The 3.3 V to Iso-5V power converter was validated by supplying an input voltage and measuring the output voltage, as shown in Figure 15. The validation testing was conducted at the operating voltage of 3.3 V and was generated using a DC power supply with the values in Figure 16. The amperage of 0.04 A was used per the average operating current in the datasheet.

A close up of a circuit board

Description automatically generated

*Figure 15: 3.3V/ISO\_5V Converter Circuit*

A screenshot of a computer

Description automatically generated

*Figure 16: Input 3.3 V*

Upon measuring the input 3.3V shown in Figure 17, the green horizontal line shows an amplitude of 3.2 V with 400 mV of noise. Furthermore, the measured output is 200 mV with 400 mV of noise. This differs greatly from the expected output of 5 V. This discrepancy can be attributed to a missing trace between the converter pin 3 and ground.

A screen shot of a computer

Description automatically generated

*Figure 17: Input 3.3 V, Output Iso 5 V*

### 15 V to Iso-15 V Power Converter Validation

The 15 V to Iso-15 V power converter converts 15 VDC to isolated 15 VDC. In Figure 15, the 15V/ISO\_5V converter schematic is shown. First, an input connector receives the input 15 V. Next, a set of capacitors in parallel is connected to ground to filter the voltage and reduce noise. Then, a power converter steps up and isolates the 15 V to 15 V. Finally, another set of capacitors in parallel is connected to ground to filter the voltage and reduce noise.

The 15 V to Iso-5 V power converter was validated by supplying an input voltage and measuring the output voltage, as shown in Figure 18. The validation testing was conducted at the operating voltage of 15 V and was generated using a DC power supply with the values in Figure 19. The amperage of 0.11 A was used per the average operating current in the datasheet.

A circuit board with wires and wires

Description automatically generated

*Figure 18: 15V/ISO\_15V Converter Circuit*

*Figure 19: Input 15V*

A screenshot of a computer

Description automatically generated

Upon measuring the input 15 V shown in Figure 20, the green horizontal line shows an amplitude of 15.1 V with 600 mV of noise. Furthermore, the measured output is 15.3 V with 800 mV of noise. This aligns greatly with the expected output of 15 V. To further validate the power converter, its performance was measured across its full range. In Table 2, the current was kept constant while the voltage was incrementally increased. In Table 3, the voltage was kept constant while the current was incrementally increased. The power converter behaved as expected for its full range of voltage and current values.

A screen shot of a graph

Description automatically generated

*Figure 20: Input 15 V, Output Iso 15 V*

|  |  |  |
| --- | --- | --- |
| Input V (V)​ | Output V (V)​ | Output Noise (mV)​ |
| 9.0​ | 15.3​ | 800​ |
| 12.1​ | 15.3​ | 800​ |
| 15.1​ | 15.3​ | 800​ |
| 18.1​ | 15.3​ | 800​ |
| 24.5​ | 15.4​ | 1000​ |

*Table 2: 15V/ISO\_15V Constant Current*

|  |  |  |
| --- | --- | --- |
| Input I (mA)​ | Output V (V)​ | Output Noise (mV)​ |
| 1​ | 6​ | 400​ |
| 40​ | 15.2​ | 600​ |
| 80​ | 15.2​ | 400​ |
| 110​ | 15​ | 400​ |
| 150​ | 15​ | 400​ |
| 190​ | 15​ | 400​ |
| 220​ | 15​ | 400​ |

*Table 3: 15V/ISO\_15V Constant Voltage*

## Subsystem Conclusion

In conclusion the rectifier, DC link, and 15 V to iso 15V converter are all working properly. On the other hand, the power control and 3.3V to iso 5V converter are not working properly. The power control will require a new chip before being tested and integrated into the full system as well as purchase of backup parts to avoid the time delay of needing to order emergency new parts. The 3.3V to iso 5V converter will require a new circuit board to add in the missing trace.

# Firmware Subsystem Report

## Subsystem Introduction

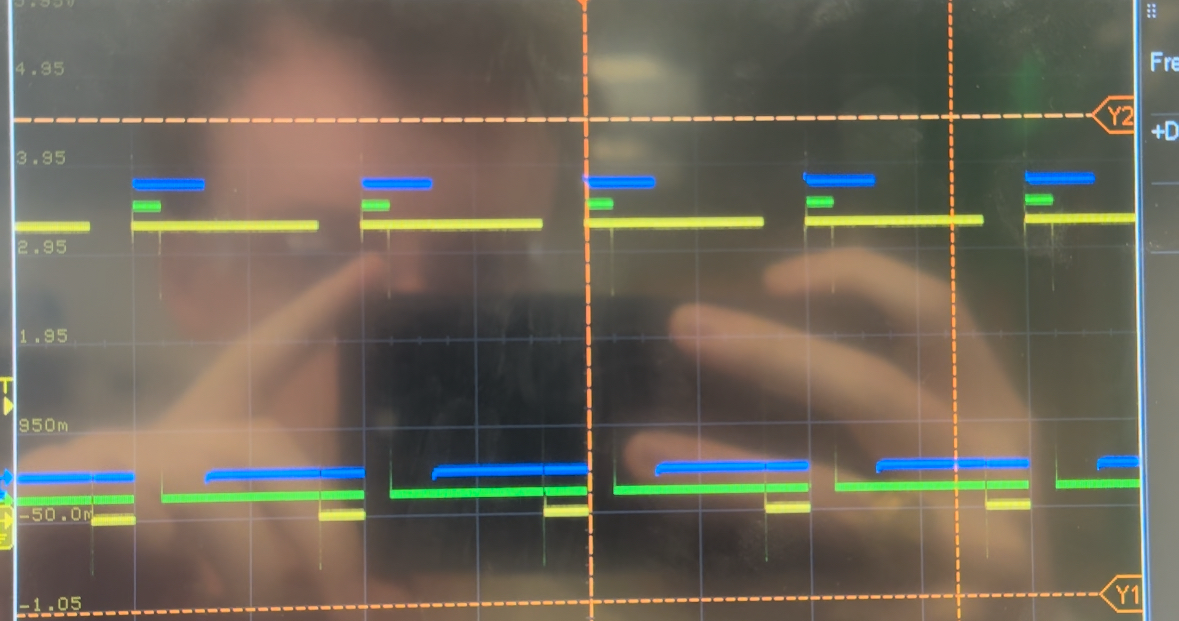
This subsystem entails all programming required to make the microcontroller work. Without the microcontroller and its firmware, the remainder of the VFD’s subsystems will not function by themselves or together to control the motor. The firmware will also be responsible for allowing the user to control different aspects of the VFD.

## Subsystem Details

This semester, the firmware was all written, tested, and demonstrated in MPLAB’s X IDE in the C language using a DSPIC33CK256MP508 microcontroller on a dsPIC33CK Curiosity Development Board (part number DM330030). This semester’s objective was to program the development board’s potentiometer to allow the user to control frequency of a three-phase PWM signal. This objective was met, and the resulting PWM signal can be shown with three channels of an oscilloscope.

The program works by continuously reading the potentiometer’s value and using that to calculate the desired frequency (10-60Hz) that should be outputted to the motor via the PWM signal. To generate the three-phase PWM signal and the different frequencies as the program runs, it makes a sine table that can be iterated through and converted into duty cycle percentages. The offsets of the three outputs are 120 degrees apart from each other to make them three-phase, and the speed that the program iterates through the sine table depends on the calculated target frequency.

## Subsystem Validation

To validate my subsystem, I tested it on the dsPIC33CK Curiosity Board. I was able to get it to properly display three different phases of a PWM signal iterating through a sine waveform of duty cycles that were all correctly offset by 120 degrees. I was also able to program the complete functionality of the potentiometer so that it can increase and decrease the frequency of the PWM signal. Though the duty cycles of the PWM waves continuously change, the oscilloscope’s output of these three phases in a single frame of the loop is shown in the figure below. This validation shows that the firmware is ready for subsystem integration next semester.

*Figure 21: Oscilloscope Output of PWM Signals Produced by Development Board*

When the program is running on my development board, it also prints several variables in each iteration of the while loop to a terminal window that can be opened in MPLAB. The variables output in each iteration are shown in the figure below.

A screenshot of a computer

Description automatically generated

*Figure 22: Debug Print Statements (Potentiometer at 0)*

## Subsystem Conclusion

The objective next semester is to integrate the development board’s firmware onto the microcontroller subsystem. This should only involve changing pin names and calibrating for the other two subsystems, but there are likely to be more challenges, so I will need to continue adding comments and print statements to ensure a clean debugging process.

Another addition I will need to make to my code is to have a low signal for each of the three phases of my PWM signal. Upon communicating with the other members of my group, they have requested that I generate these extra three signals.

One potential problem that may arise during integration is the need to calibrate the PWM signal’s frequency needed by the motor. I know I will need to calibrate this signal and do some troubleshooting because the program currently prints that its target frequencies are between 10-60Hz; upon observation of the output PWM waveforms on the oscilloscope, however, the signals are a lot slower than that.

Overall I have completed this semester’s objective for my subsystem and am ready to integrate with the other subsystems in the Spring in ECEN 404.