ADC Resolution vs Sensor Less Motor Control Performance

Cristian Ornelas

Tyler Hawkins

Tamara Hussam A Basfar

John Adam King

**Functional System Requirements**

REVISION – 2

5 December 2024

Functional System Requirements

for

ADC Resolution vs Sensor Less Motor Control Performance

Prepared by:

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

Author Date

Approved by:

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

Project Leader Date

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

John Lusher, P.E. Date

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

T/A Date

**Change Record**

| **Rev.** | **Date** | **Originator** | **Approvals** | **Description** |
| --- | --- | --- | --- | --- |
| **1** | [9/26/24] | [ADC Resolution vs Sensor Less Motor Control Performance] |  | Draft Release |
| **2** | [12/04/2024] | [ADC Resolution vs Sensor Less Motor Control Performance] |  | Second Revision |

**Table of Contents**

[**Table of Contents III**](#_heading=h.1fob9te)

[**List of Tables IV**](#_heading=h.2et92p0)

[**List of Figures V**](#_heading=h.2et92p0)

[**1.**](#_heading=h.tyjcwt) **Introduction 1**

[1.1.](#_heading=h.3dy6vkm) Purpose and Scope 1

[1.2.](#_heading=h.4d34og8) Responsibility and Change Authority 2

[**2.**](#_heading=h.2s8eyo1) **Applicable and Reference Documents 2**

[2.1.](#_heading=h.17dp8vu) Applicable Documents 2

[2.2.](#_heading=h.3rdcrjn) Reference Documents 2

[2.3.](#_heading=h.26in1rg) Order of Precedence 2

[**3.**](#_heading=h.35nkun2) **Requirements 3**

[3.1.](#_heading=h.1ksv4uv) System Definition 3

[3.2.](#_heading=h.2jxsxqh) Characteristics 4

[3.2.1.](#_heading=h.z337ya) Functional / Performance Requirements 4

[3.2.2.](#_heading=h.3j2qqm3) Physical Characteristics 5

[3.2.3.](#_heading=h.1y810tw) Electrical Characteristics 6

[3.2.4.](#_heading=h.3whwml4) Failure Propagation 7

[**4.**](#_heading=h.2bn6wsx) **Support Requirements 7**

[**Appendix A Acronyms and Abbreviations**](#_heading=h.qsh70q) **8**

[**Appendix B Definition of Terms**](#_heading=h.1pxezwc) **9**

**List of Tables**

Table [1. Applicable documents………………………………………….……………………. 1](#_heading=h.1t3h5sf)

Table [2. *Reference…………………*………………………………………………………..…..](#_heading=h.44sinio)4

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

**List of Figures**

[Figure 1. Conceptual Sensorless Motor Controller…………………...……………………. 1](#_heading=h.1t3h5sf)

[Figure 2. Block Diagram of System……………………………………………………..…..](#_heading=h.44sinio)3

Figure 3. Simplified Block Diagram of System ………………………………………...…...4

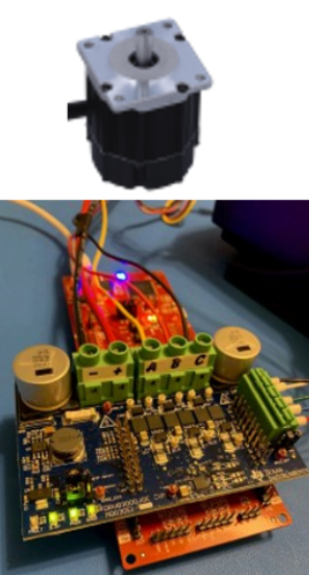
Figure 4. F28P65x Board Connections………………………………………………………6

# 

# 1. Introduction

## 1.1 Purpose and Scope

The project focuses on developing software for the TI c2000 chip to implement sensorless field-oriented control (FOC) for motor systems, as well as porting and testing an older sensorless motor code provided by TI. Instead of traditional sensors, a microcontroller will estimate the rotor’s position. By implementing the sensorless motor control, there will be cost reductions, improved efficiency and increased reliability. To enhance and optimize this estimation, the ADC resolution will be increased. Our team will test our sensorless motor control at different speeds to evaluate the ADC resolution and its accuracy. The accuracy of the motor controller will be validated by comparing its performance to those from an incremental encoder.



*Figure 1. Conceptual Sensorless Motor Controller*

The ADC sensorless motor control performance will be designed to fit the proper parameters for professionals to carry out the finished product. The milestones for the scope are listed below:

* Implement new Sensorless Motor Control subsystems in isolated environments.
* Implement Sensorless Motor Control subsystems onto F28p65x and DRV8300 board
* Port older motor control solution to F28p65x and DRV8300 board with SysConfig support
* Modify SysConfig setting and software for different ADC resolutions (12/16 bit)
* Modify the software to enable oversampling and new ADC feature in F28p65x
* Configure software to run an algorithm, in 32-bit and 64-bit floating point
* Run tests on sensorless control performance at low speed for each configuration

## 1.2 Responsibility and Change Authority

The team leader, Tyler Hawkins, is responsible for ensuring that all system specifications are met. Any changes to the deliverables or project specifications must be approved by both Tyler Hawkins and the sponsor’s official representative Kevin Allen.

# 2. Applicable and Reference Documents

## 2.1 Applicable Documents

The following documents, of the exact issue and revision shown, form a part of this specification to the extent specified herein:

| **Document Number** | **Revision/Release Date** | **Document Title** |
| --- | --- | --- |
| SBVS144C | 2012 | Integrated MCU Power Solution for C2000 Microcontrollers |
| IEEE-STD-1241 | 2001 | IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters |
| MCU117 | 2023 | LAUNCHXL-F28P65X layout |
| v.0.3.0 | 2023 | Servo Drive with CAN - Lab Projects User’s guide |

## Table 1: Applicable documents

## 2.2 Reference Documents

The following documents are reference documents utilized in the development of this specification. These documents do not form a part of this specification and are not controlled by their reference herein.

| **Document Number** | **Revision/Release Date** | **Document Title** |
| --- | --- | --- |
| v1.3 | 2021 | C2000 software guide |
| SLVUBV6 | 2020 | DRV8300Dxxx-EVM User’s guide |

## Table 2: Reference documents

## 2.3 Order of Precedence

In the event of a conflict between the text of this specification and an applicable document cited herein, the text of this specification takes precedence without any exceptions.

All specifications, standards, exhibits, drawings or other documents that are invoked as “applicable” in this specification are incorporated as cited. All documents that are referred to within an applicable report are considered to be for guidance and information only, except ICDs that have their relevant documents considered to be incorporated as cited.

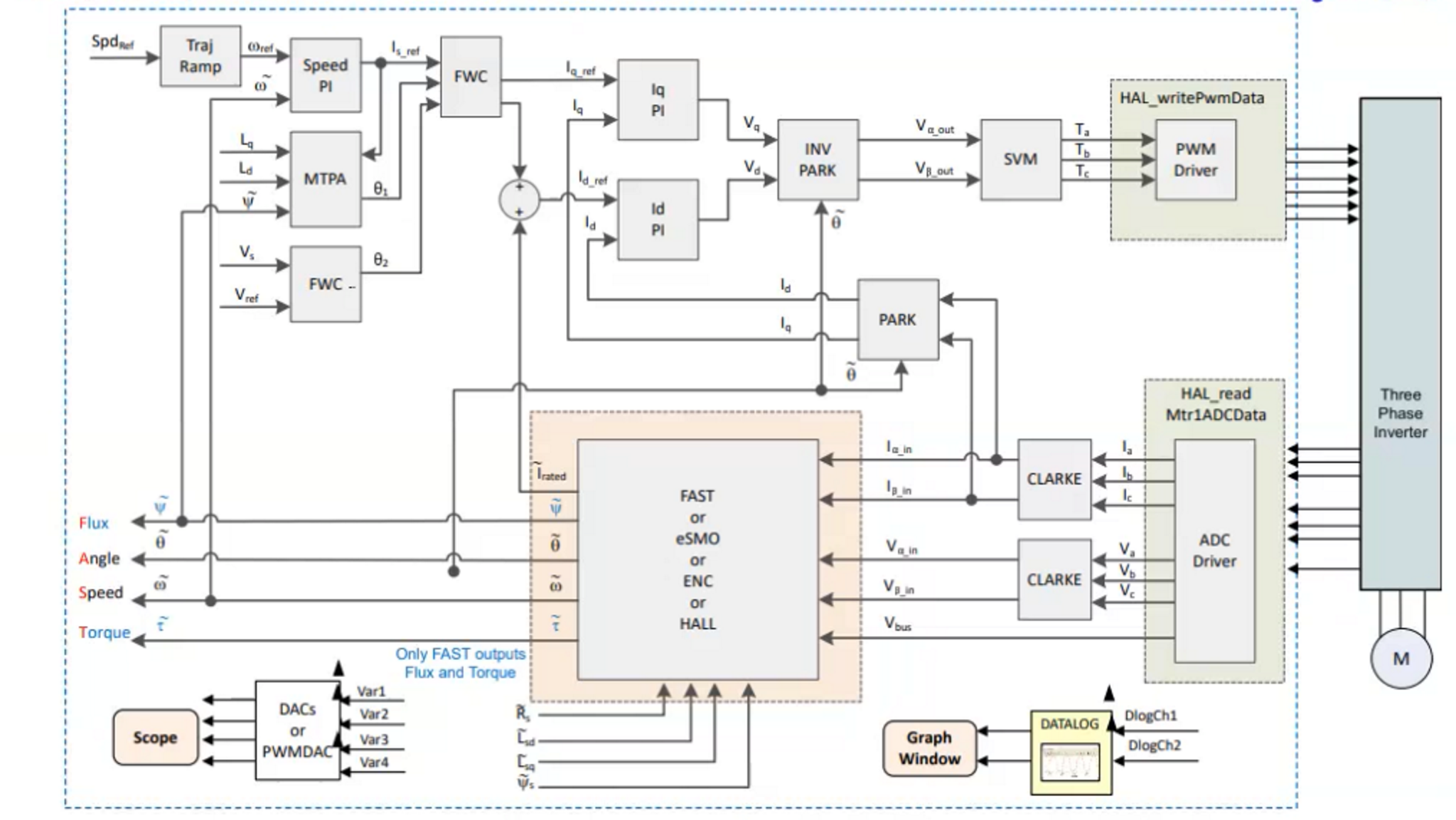
# 3. Requirements

This section defines the minimum requirements that the development item(s) must meet. The requirements and constraints that apply to performance, design, interoperability, reliability, etc., of the system, are covered.

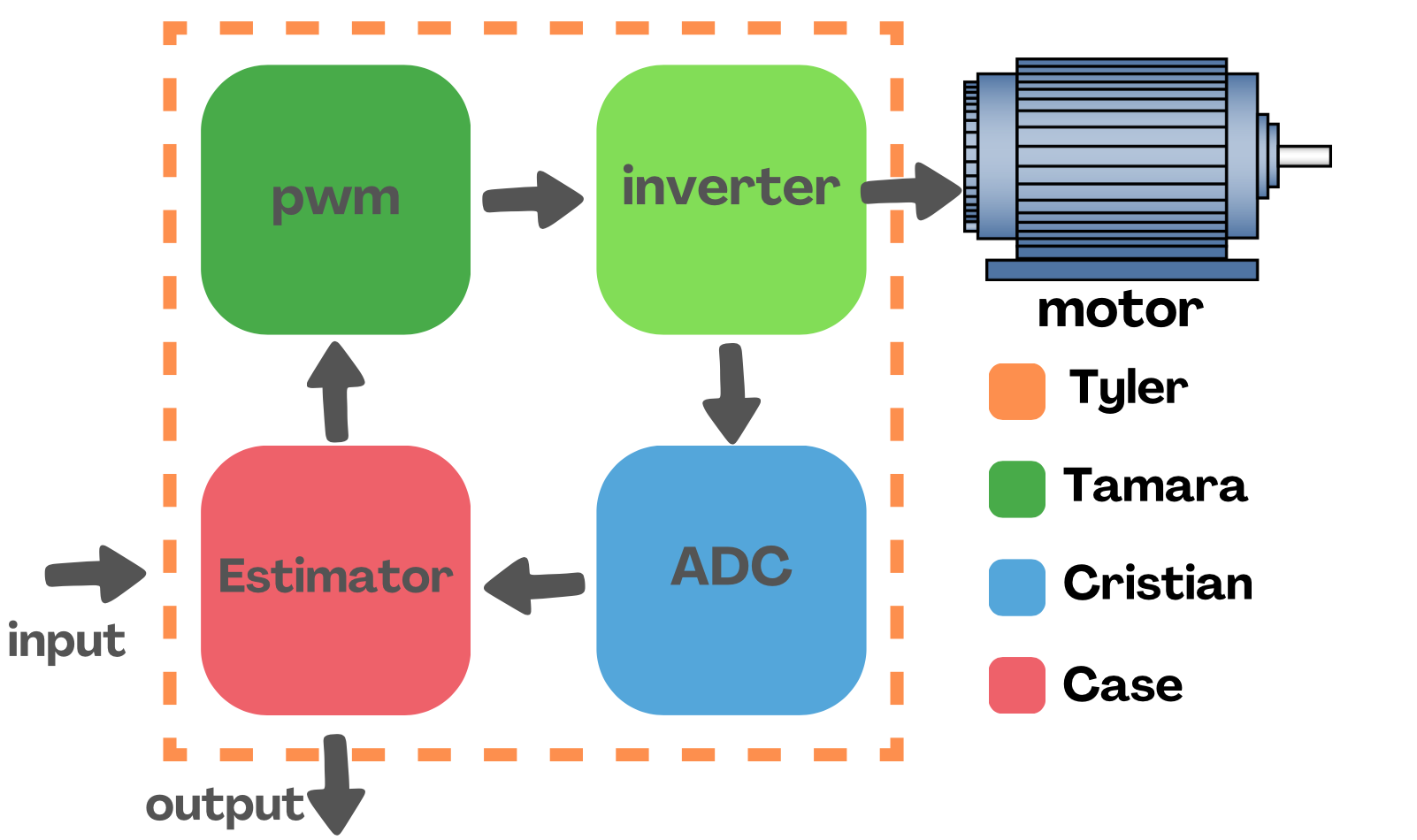
## 3.1 System Definition

The project is split up into four main subsystems: Motor Control Porting, PWM driver/three-phase inverter, ADC driver, and estimator. The porter is responsible for taking the provided motor control code and updating it to function on the new launchpad. The PWM driver/three-phase inverter is responsible for converting a DC signal into 3 phases of AC waveforms. The ADC driver focuses on ensuring that the analog signal is properly conditioned to meet the input requirements of the ADC for accurate conversion. This includes signal conditioning, impedance matching, noise filtering, and signal buffering. Finally, the estimator is responsible for taking the back EMF of the motor and using it to determine the rotor orientation. The rest of the blocks in the diagram below are responsible for general control requirements and operation dependencies.

This system/ project will include testing for ADC oversampling and different ADC resolutions (12/16 bit). This will bring better SNR to currency and voltage sensing, ultimately bringing more accurate estimation results.



*Figure 2. Block Diagram of System*



*Figure 3. Simplified Block Diagram of System*

## 

## 3.2 Characteristics

### 3.2.1 Functional / Performance Requirements

#### 3.2.1.1 Basic Motor Control

The sensorless control solution developed should effectively control the Teknic M-2310P-LN-04K Motor. The control solution should effectively determine the rotor's position, speed, torque, and rotation direction.

*Rationale: This is the core system performance requirement. The sensorless control solution’s performance is intended to be comparable to a motor sensor control solution.*

#### 3.2.1.2 SysConfig Support

Sensorless control of the motor should be configurable using the SysConfig IDE.

*Rationale: This is the core system performance requirement. Using SysConfig allows for easy change of system options such as ADC Resolutions and 32/64 floating point.*

#### 3.2.1.3 12-Bit & 16-Bit ADC Resolutions

The sensorless control solution should support ADC Resolutions of 12 bit & 16 bit

*Rationale: This is the core system performance requirement.*

#### 3.2.1.4 32 Bit & 64 Bit Floating Point

The sensorless control solution should support 32 Bit & 64 Bit Floating Point

*Rationale: This is the core system performance requirement.*

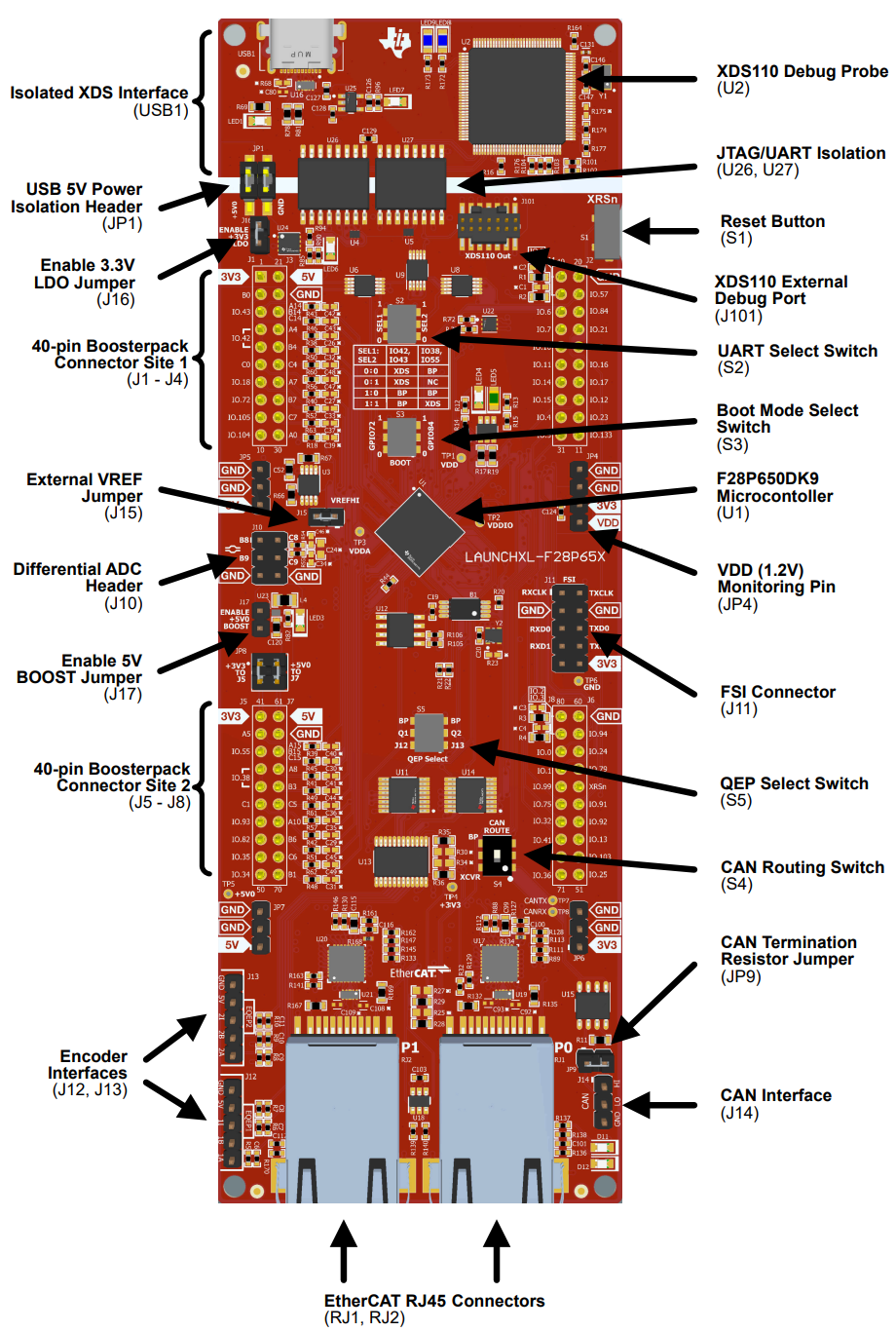
### 3.2.2 Physical Characteristics

The project is centered around a software application, however, this software is being designed to run with specific components; a microcontroller, motor controller, and motor. For development, the microcontroller is an F28p65x launchpad from Texas Instruments. While this board has dual processing cores, only Core-1 is being used. The motor controller is a DRV8300DIPW-EVM evaluation module from Texas Instruments. Lastly, the motor itself is a Low Voltage Servo Motor from Texas Instruments.

#### 3.2.2.1 Board Connections

Our F28p65x Launchpad Board should be connected to the DRV8300 motor driver board and the Teknic M-2310P-LN-04K Motor.

*Rationale: In order to control the motor listed above, the boards should be connected using the board's contact pins, in addition to jumper wires.*

**

*Figure 4. F28P65x Board Connections*

### 3.2.3 Electrical Characteristics

#### 3.2.3.1 Inputs

1. USB-C connection to the host device for the purpose of receiving control commands.
2. The Motor Driver Booster Pack is connected using the 40-pin connection at site 2 shown in the image above.
3. The motor is connected to the driver board’s corresponding phase a,b & c headers, as well as positive power & ground.
4. When applicable for testing connections to the motor’s sensor will be connected to the respective phase connectors on the driver board

*Rationale: By design, should limit the chance of damage or malfunction by user/technician error.*

##### 3.2.3.2 Input Voltage Level

The input voltage of the F28P65x Board shall be 5 VDC through the USB connection

The DRV8300Dxxx-EVM is designed for an input external supply from 6 VDC to 100 VDC and up to 25-A continuous drive current (software limited).

*Rationale: TI F28P65x & DRV83000 Board specifications*

##### 3.2.3.3 External Commands

The motor control solution shall document all external commands in the appropriate ICD.

*Rationale: The ICD will capture all interface details from the low-level electrical to the high-level packet format.*

#### 3.2.3.6 Outputs

##### 3.2.3.7 Data Output

The motor control solution shall include the Sysconfig data interface that is compatible with the system.

*Rationale: Using Sysconfig, data output motoring can be viewed through the host device.*

##### 3.2.3.8 Diagnostic Output

The motor control solution shall include the Sysconfig control and data logging interface.

*Rationale: Provides the ability to control things for debugging.*

#### 3.2.3.9 Wiring

The motor control solution shall follow the guidelines outlined in DRV8300xxx-EVM User’s Guide paragraph 2.1 Hardware Connections Overview.

*Rationale: Conform to Driver Board standard.*

### 3.2.4 Failure Propagation

**3.2.4.1 Diagnostic Errors**

The motor control solution will monitor the voltage & current levels allowing visibility of any errors of the hardware and its operation.

*Rationale: This will help preserve the integrity of the data being collected*

# 4. Support Requirements

**4.1.1. Computer USB Port**

A computer is required to compile & run motor control solutions using the CCS environment.

This requires a minimum of 4GBs of RAM, 2.5GBs disk space, & 2.0GHz single-core processor.

# Appendix A: Acronyms and Abbreviations

EMF Electro-motive Force

BIT Built-In Test

CCA Circuit Card Assembly

EMC Electromagnetic Compatibility

EMI Electromagnetic Interference

EO/IR Electro-optical Infrared

FOR Field of Regard

FOV Field of View

GPS Global Positioning System

GUI Graphical User Interface

Hz Hertz

ICD Interface Control Document

kHz Kilohertz (1,000 Hz)

LCD Liquid Crystal Display

LED Light-emitting Diode

mA Milliamp

MHz Megahertz (1,000,000 Hz)

MTBF Mean Time Between Failure

MTTR Mean Time To Repair

mW Milliwatt

PCB Printed Circuit Board

RMS Root Mean Square

TBD To Be Determined

TTL Transistor-Transistor Logic

USB Universal Serial Bus

VME VERSA-Module Europe

PWM pulse width modulation

ADC Analog to Digital converter

INV inverter

c2000 real-time microcontroller

sysConfig development tool for c2000

CCS Code Compiler Studio

CMPSS Comparator Subsystem

# Appendix B: Definition of Terms

Sysconfig: a configuration tool designed to simplify hardware and software configuration challenges to accelerate software development. SysConfig provides an intuitive graphical user interface for configuring pins, peripherals, radios, software stacks, RTOS, clock tree and other components. SysConfig will automatically detect, expose and resolve conflicts to speed software development.