A Novel Approach for Steer by Wire Technology for Road Vehicles

Behnam Eskandariun

Submitted to the
Institute of Graduate Studies and Research
In partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
September 2016
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

_____
Prof. Dr. Mustafa Tümer
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

_____
Prof. Dr. Hasan Demirel
Chair, Department of Electrical and Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

_____
Asst. Prof. Dr. Davut Solyali
Supervisor

Examining Committee
_____

1. Prof. Dr. Hasan Demirel   _____

2. Assoc. Prof. Dr. Hasan Hacışevki   _____

3. Asst. Prof. Dr. Davut Solyal   _____

ABSTRACT

Recent progress in the steer-by-wire technology has taken a significant improvement in vehicle safety and driving performance. By completely disconnecting the steering wheel from the tires provides excellent flexible features for vehicle dynamic control, it also introduces some practical difficulties for steering system control. The purpose is increase the system flexibility regarding vehicle dynamic control and safety and decrease design and production costs for the manufacturer. In this project, a novel method is being proposed for steer-by-wire technology to be used in road vehicles. This method discards the use of any mechanical linkage between the steering wheel and the road wheel's steering mechanism. This is achieved by using angular absolute position sensor to detect the position of the steering wheel. A Brushless DC Motor employed to emulate the steering feel force feedback from the tire on the steering wheel. Pneumatic actuators control the road wheels to control the steering angle.

Therefore, the main objectives are as follows:
- Steer by Wire (SbW): Removing any mechanical linkage between the steering wheel and the road wheels.
- Steering wheel position determination and translate to the road wheel controller.
- Creating the desired force feedback onto steering wheel

The level of achievement to the objectives was found completely satisfactory. The position measurement and conversation was precise according to experimental results.

The torque controller has made reasonable force feedback on the steering wheel, and it provided same force output due to the constant input in experiment repetitions.

ÖZ

Kablo kumandalı direksiyon sistemi araç güvenliği ve sürüş performansı bakımından son zamanlarda belirgin bir gelişim göstermiştir. Konvansiyonel direksiyon milini devre dışı bırakarak, önerilen kablo kumandalı direksiyon sistemi kullanarak aracın daha esnek bir şekilde dinamik kontrolunun sağlanması amaçlanmıştır. Kablo kumandalı sistemlerde amaçlanan sistem esnekliğini ve güvenliği artırmak, tasarım ve üretim maliyetlerini düşürmektir. Bu projede araçlarda kullanılmak üzere yeni bir kablo kumandalı direksiyon sistemi öngörülmektedir. Önerilen sistemde direksiyonun tekerleklere olan mekanik bağlantıları tamamen kaldırılmıştır. Önerilen kablo kumanda sisteminde mutlak pozisyon sensörü kullanılarak dümenin pozisyonu tesbit edilmektedir. Tekerleklerde oluşan kuvveti sürücüye geri dönünüşüm olarak aktarabilmek için de fırçasız doğru akım motoru dümene bağlanarak kullanılmıştır. Tekerleklerin açısı ise pnömatik silindir kontrolü yapılarak sağlanmıştır.

Dolayısı ile projenin amaçları şöyledir;

- Kablo kumandalı direksiyon sistemini, konvensiyonel mekanik bağlantıları
kaldırarak oluşturmak

- Dümen açısını tespit edilmesi ve tekerleklere aktarılması

- Tekerleklerde oluşan kuvvetin sürücüye geri dönüşüm olarak aktarılması

Projenin amaçları başarılı bir şekilde uygulanmıştır. Dümenin pozisyon ölçümü ve çevrimi elde edilen deney sonuçlarına göre kesinliği yüksek olmuştur. Tork kontrolü yine elde edilen deney sonuçlarına göre doğruluğu ve kesinliği yeterli bulunmuştur.

Anahtar Kelimeler: Kablo kumandalı direksiyon sistemi, pnömatik, InstaSpin, Tork Kontrolü

DEDICATION

To My Family,

ACKNOWLEDGMENT

I would like to record my gratitude to Asst. Prof. Dr. Davut Solyali for his supervision, advice, and guidance from the very early stage of this thesis as well as giving me extraordinary experiences throughout the work. Above all and the most needed, he provided me constant encouragement and support in various ways. His ideas, experiences, and passions have actually inspired and enrich my growth as a student. I am indebted to him more than he knows.

I would like to acknowledge the members of my graduate committee for their advice and guidance, most especially Prof. Dr. Hasan Demirel for all his advice and encouragements.

Special thanks for Mrs. Samira Asgari for her support and motivation on finishing this project and who has been a good friend to me. I am grateful in every possible way.

My thanks go to my friends who helped and encouraged me during the period of my studies and this thesis such as Mrs. GhazaalSheikhi and Mr. Payam Zar Bakhsh for his crucial contribution to this study.

Finally, I want to thank my parents for their patient support throughout all my years in school and who would have been just as happy had I decided to do something else.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

SP   Setpoint

PV   Process Variable

PM   Permanent Magnet

BLDC   BrushLess Direct Current

FOC   Field Oriented Control

EMF   Electro-Motive Force

EV   Electric Vehicle

SbW   Steer by Wire

ECU   Electronic Control Unit

HIL   Hardware in the Loop

K_c Proportional Co-efficient

K_I Integral Co-efficient

K_D Derivative Co-efficient

τ_I Integral Time usually in Minute

τ_D Derivative Time in trend

PID   Proportion Integral Derivative

PD   Proportion Derivative

PI   Proportion Integral

Chapter 1

INTRODUCTION

Drive by wire technology is being increasingly used by many automotive manufacturers for vehicle products. The Drive-by-wire (DbW), replaces the conventional mechanical linkages and mechanical actuators like steering column, pumps, intermediate shafts, pumps, belts, coolers, vacuum servos and master cylinders with electronic control systems and electromechanical actuators in the vehicle. In DbW technology, the goal is to achieve a flexible control system. The DbW gives this ability to electronic control systems and electronic decision makers to take over the vehicle control to give a comfortable driving experience to the driver and also increasing the safety in case of emergency or loose of control by the human driver.

One of the most challenging topics in the Drive-by-wire's subsystems is a steer by wire control system. The Steer by wire system eliminates the mechanical linkage between the steering wheel and road wheels. In a conventional car, there are some mechanical linkages and hydraulic or electric steering assist which are replaced with the combination of electronic control systems and electromechanical actuators to act as vehicle steering system. Figure 1, illustrates a conventional direct steering system based on rack and pinion mechanical linkages.
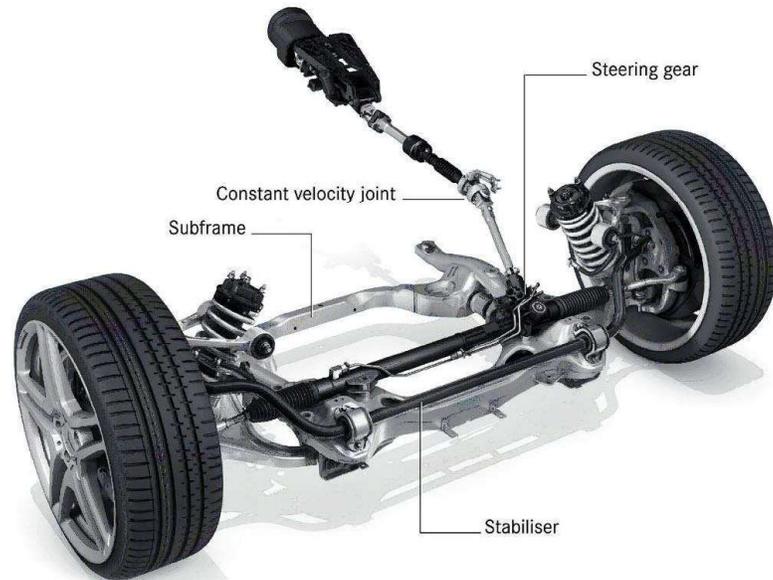
Figure 1: Direct Rack and Pinion steering system [18]

In the Steer-by-wire system (SbW), an electronic control system controls the vehicle's steering angle by getting the position of the steering wheel. An electromechanical actuator helps the Steering Electronic Control Unit (SECU) to adjust the tires angle in the proper position. The conventional mechanical linkage between the road wheels and the steering wheel gives a road condition feeling to the driver through the steering wheel. However, in the SbW system all the mechanical connections are removed, so to give real driving experiences to the human driver, the electronic control system should be able to create conventional vehicles steering feel on the steering wheel by using electromechanical actuators.
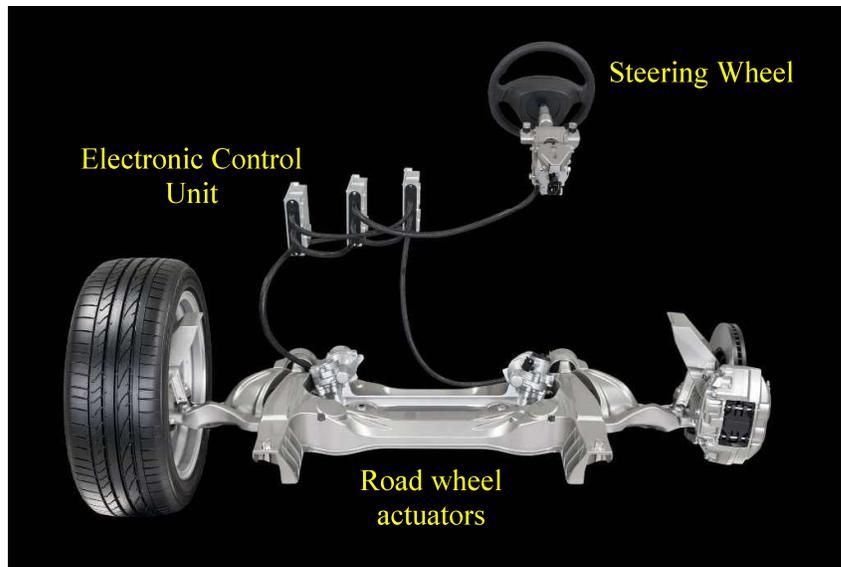
Figure 2: Steer by wire system schematic [19]

Figure 2 illustrates the SbW general schematic. The steering wheel actuator emulates the steering feel on the steering wheel and also measures the Hand wheel position and send to the electronic control unit. The electronic control unit is in charge to get the measured lateral forces on the tire and use it in the steering wheel actuator control. Moreover, the electronic control unit gets the Hand wheel position and controls the road wheel steering angle accordingly.

Therefore, the main objectives are as follows:

- Steer by Wire (SbW): Removing any mechanical linkage between the steering wheel and the road wheels.

- Steering wheel position determination and translate to the road wheel controller.

- Creating the desired force feedback onto steering wheel

In the second chapter the literature review and the other studies are discussed and in continue the third chapter is about implementation methodology and all used

hardware and actuators are introduced. Finally, the fourth and chapter represents the results for the experiments. In the last chapter the conclusion and future works are represented.

Chapter 2

LITERATURE REVIEW

This chapter is intended to provide information about single design methodologies for the Steer-by-Wire (SbW) systems. Steer by Wire uses a servo motor to control the angle of the front road tires by getting the commands from steering wheel from the driver, and also fed back road forces and conditions on the front tires to the driver through an actuator which is connected to the steering wheel.

2.1 The single design method for the SbW

Asai and *et al.* [1] proposed an updated single servo drive method of SbW implementation. SbW system structure which is used in their research and many other types of research in this field, illustrated in Figure 3. Steering wheel mounted on geared Brush DC motor with motor angle sensor installed on it. The connection between the steering wheel and DC Motor has been established with torque sensor which measures the applied torque on the steering wheel by the driver. Brushless DC motor connected to the steering rack in order to make desired force on tires to change tires angle. A current sensor measures Brushless DC Motor current for calculating the motor force. A motor angle sensor is used to coordinate rack position which is used in position feedback. Signals from the current sensor and the angle sensor are collected by the motor driver and send to the ECU. Likewise, the torque and the current sensor signals from steering wheel collects by steering wheel's motor driver and send to ECU as the same way. [1]

steering wheel
brush DC motor
motor angle sensor
torque sensor
gear
motor current sensor
motor driver
controller
steering wheel actuator
road wheel actuator
motor driver
motor current sensor
motor angle sensor
brushless DC motor
rack
road wheel
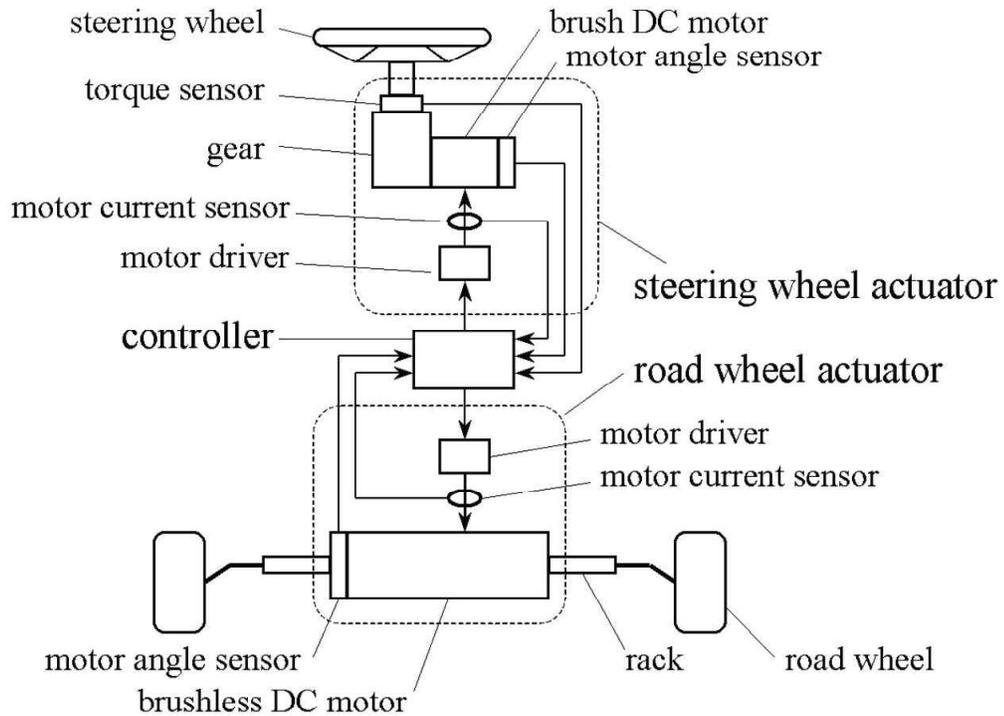
Figure 3: System structure of the steer-by-wire system[1]

The proposed method of Asai *et el.* [1] to control SbW system is shown in Figure 4. The Control block can be divided into three main part. Namely, steering wheel actuator unit, Road wheel actuator subsystem and Road Condition force estimator part. The main reason to use steering wheel actuator is to apply force on the steering wheel that reflects road conditions on it. In Asai's implemented the system, steering wheel is connected to a brushless DC (BLDC) motor shaft and another side of the motor shaft has an angle sensor to obtain steering wheel position. A current Prportional Integral controller, controls the BLDC motor torque, with Torque Command signal as a reference signal and the motor current from a current sensor as a control loop feedback signal. The Torque Command reference is summation of the Torque reference signal and torque compensation signal. The torque reference signal is obtained from reaction torque map which maps external torques on the front tires

6

to steering feel force feedback. The Compensation Unit gets the angle, current and reaction torque as feedbacks from the steering wheel and generate feedback compensation signal which is used to stabilize the system and generate better feeling on the steering wheel. The angle sensor on the steering actuator is also used to get steering wheel angle as a set point for road tires position controller.

The road wheel actuator as it is proposed in Asai *et el.* controlled by a Proportional Derivative position controller and a PI current controller respectively. The position controller uses steering wheel angle sensor as a reference and use rack position sensor as position feedback to follow steering commands properly. The current controller is used to limit torque in desired value.

A disturbance observer is employed to observe both current and position of the road actuator to estimate desired force which should be applied on the steering wheel. Its output fed back to steering torque controller after passing through gear ratio converter and reaction torque mapping block.

In conclusion the proposed method in Asai and *et al.* [1], is using two motors, one for controlling road tires and the other to control the steering wheel. Using disturbance observer in this research has an adequate response to steering feel desired forces on the steering wheel, according to their experimental results in their research.
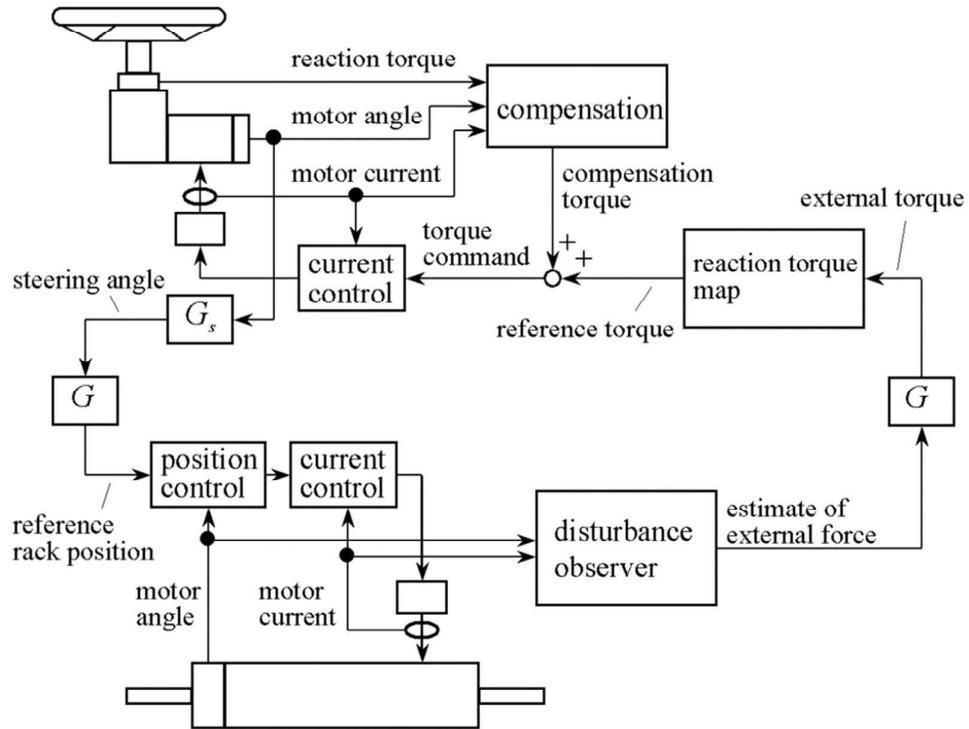
Figure 4: Control block diagram of the steer-by-wire system [1]

2.2 Force feedback and Steering feel

One of the challenging issue in the procedure of replacing the conventional mechanical steering wheel with the SBW; is finding a method to emulate the force feedback from the road tires to the steering wheel.

There are different force feedback methodologies in the literature. There are a direct method, Mathematical method, and Hybrid fusion method.

2.2.1 Direct methods

In Nguyen's [2] proposed method, the steering feeling is recreated by measuring the electrical current in the motor that drives the front wheel using a simple current sensor. The innovative aspect of the work is using a simple sensor to measure the feedback force and improve the real driving feeling accordingly. For mapping related

torque, angle effect and velocity effect terms are considered in two equations for estimating the force feedback. The proposed scheme has been simulated using LABVIEW and compared with some conventional methods of steering feeling reproduction. Simulation results have proved the efficiency of the method compared to the approaches based on force measurement or disturbance estimation.

From the mechanical perspective, the feedback force is generated by friction, damping and the moment of inertia in the front wheel. Wheel rubber properties, the speed, and other unknown factors influence the force feedback as a wheel. In fact, the driving feeling is produced because of this total force. This feeling has two basic components namely the velocity effect component and the angel effect component. This concept is the main idea of the torque map approach in which the angle of the navigating wheel and the velocity of the vehicle are measured. These measurements are utilized to implement the real sense of driving [2].

Another proposed approach for producing steering feeling is the method proposed by Pastorino *et al.* [3] based on the amplifier-motor-gearbox model. In this study, the force feedback that needs to apply on the driver is evaluated by an assembly constitute of linear quadrant servo amplifier, a coreless permanent magnet DC motor, and double stage terrestrial gearbox. The model is created based on physically significant parameters that are precisely estimated by an accurate identification process. As all the parameters of the proposed model have a direct physical interpretation, model identification procedure has been implemented in a straightforward way. Model assessment is performed by means of three real world experiments. According to the valid data collected in three real driving conditions,

the model simulated in MATLAB Simulink accurately fits the dynamics of the typical driving situations.

2.2.2 Model based mathematical methods

 An innovative method has been proposed by Mehdizadeh [4], in which a reference linear model of the vehicle is used as a virtual vehicle to reproduce the appropriate steering feeling. In SbW vehicles that driver is assisted by a lane keeping system or a steering assistant system, generating the realistic sense of driving is crucial. In fact, in these systems the steering feeling should be reproduces artificially as a force feedback at the steering wheel. However, this is not possible to apply this torque on the front wheel as it disturbs the function of automatic lane keeper. Applying this force on the wheel can simply mislead the controller since it is inferred as an unnatural behavior; the phenomena that may put the system in an unstable state. In [4], the results of simulations have confirmed that the proposed artificially generated force feedback does not intercede with the lane-keeping controller.

Another approach to tackling the problem of steering feeling is based on real-time fuzzy Proportional Integral Derivative (PID) controller that gives the driver a realistic feeling at different vehicle velocities [5]. The proposed system is designed and investigated in five stages. In the first and second step, the steering wheel and the aligning torque are mathematically modeled. Then, the scheme of the road feeling control is devised. The next step is the design of the fuzzy PID controller. Finally, the proposed method is evaluated by simulated experiments.

Jie [5] also performed some simulations to evaluate the relation between the input angle and the torque in an Steer By Wire (SbW) vehicle with a conventionally driven

one. The simulations are conducted in low speed and high speed for better comparison. Experimental results have shown that while in high speed the conventional vehicle is more efficient, at low speed, the Steer by Wire (SBW) vehicle is more accurate. In fact, in when the speed is low, input angle is smaller in SBW system compared to conventional vehicle. Also, according to the simulations, it was observed that the angle of entry and the torque required for steering were significantly smaller in SBW system than that in the conventional vehicle. Furthermore, investigations of the angle ratio have revealed that angle ration can be changed when the velocity is in the range of 20km/h to 100km/h. Simulation results have shown that the force torque could not be altered at high speeds larger than 100km/h as well.

2.2.3 Hybrid methods

Utilization of brushless motors for creating steering feeling has been proposed in a study by Bertacchini [6]. The force feedback system is a plain design based on a brushless motor as the force actuator. Evaluation of the suggested platform in Figure 5 is performed using 'hardware-in-the-loop' or briefly Hardware In the Loop (HIL) experimental settings. Firstly, the dynamics of SBW vehicle are modeled to be employed in virtual implementation. The model is also applicable to the validation procedure of the proposed force feedback reproduction method. It has stated in the paper that experimental results proved the efficiency of the control algorithm for SBW system. As a matter of fact, the electronic control unit (ECU) realization of the brushless technique has reproduced a natural steering feeling to the driver compatible to common driving feeling. Proposed system in Bertacchini *et al.* research illustrated in Figure 5. They used a well-known bicycle model [7] for virtual hardware of their experimental HIL model. This model is a simplified model of vehicle steering

11

dynamic model. Their experiments are divided into four type of test. A) Motor current vs. Torque. B) Corrugated road environment C) Torque vs. Steering wheel position D) Torque vs. Car Velocity. E) Trajectory. F) Self Alignment Moment.

In Bertacchini study, Results shows that using Field Oriented Control (FOC) method to control BLDC motor has an acceptable reproduction of conventional steering feeling.
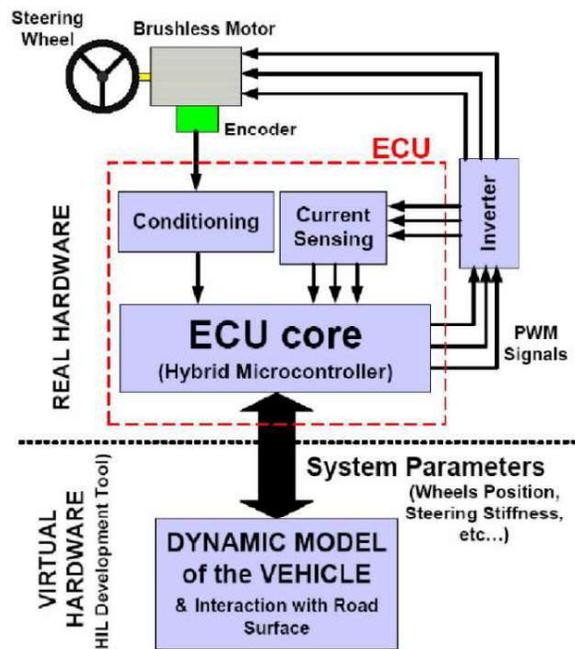
Figure 5: Bertacchini et al. Proposed HIL model [6]

An approach using heading and lateral error correction is proposed by Switches [8]. His algorithm has a lane-keeping controller in the steering feeling reproduction system. The stability degree of the SBW system is measured in different situations where alternative sources are exploited to create the force feedback. In fact, the stability of the vehicle is crucial in all the strategies implementing a steering feeling procedure. While an SBW vehicle can be directed to an unstable state by a force

feedback system (specifically when there is no driver input), a stable system does not guaranty a high-quality steering feeling. Investigation results of the study, have revealed that by keeping the steering feeling system in an acceptable condition, changes in the force feedback or the controller responsible for lane keeping, have a profound effect on the vehicle performance.

Balachandran e*t al.* [9]in the starting proposed a simple model of steering feel to capture the effects of tire torques, power assist and modifications of natural inertia and steer by wire steering system dumping [9]. These parameters do not just depend on steering system model because the vehicle dynamic also affects them in reality. To tune this parameter to feeling desired torque on the steering wheel, using a simple vehicle dynamic is proposed in this research. In the conventional steering system three major parameters influences the steering feel which are represented as follows: a) torques due to inertia b) tire moments c) torque which is affected by power assist.
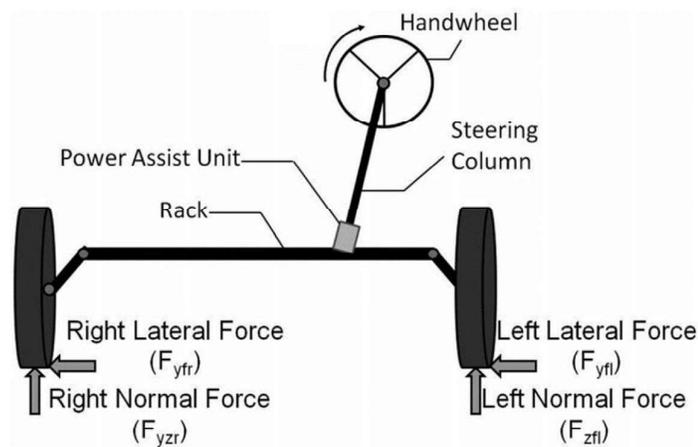
Figure 6: Forces on the road tires [9]

The simplified model is presented in Balachandran research [9], incorporates torques and tire moments according to steering system characteristics as well as simplifying

13

power assist model to be more general. Based on proposed model in this paper steering wheel motor torque is the sum of damping torque, and inertia torque and assisted tire moment for combined effect of the power assist, jacking torque and the alignment moment.

Moreover, according to Balachandran's [9] study, to recreate real steering feel on steering wheel they used objective performance measure to evaluate on-center handling and are obtained by performance of weave test according to appropriate standards. Steering feel of the vehicle, during straight line nominal driving is called on center handling. The performance measures in this paper obtains data from following parameters: 1) On-center feel; 2) return ability; 3) steering sensitivity; 4) linearity; 5) effective torque stiffness. Finally, these measures are obtained using three cross plots: (a) driver steering torque input vs. hand wheel torque. (b) Hand wheel torque vs. lateral acceleration. (c) Lateral acceleration vs. steering torque inputs.

2.3 Steering wheel actuation and control

2.3.1 PID Control

PID (proportional-integral-derivative) controller is most commonly used control method in industrial control applications [10]. It is very simple, and it has very clear control principal. Furthermore, PID controller is very easy to implement regarding hardware and software implementation. A typical basic closed loop controller and its Inputs and outputs are illustrated in Figure 7.
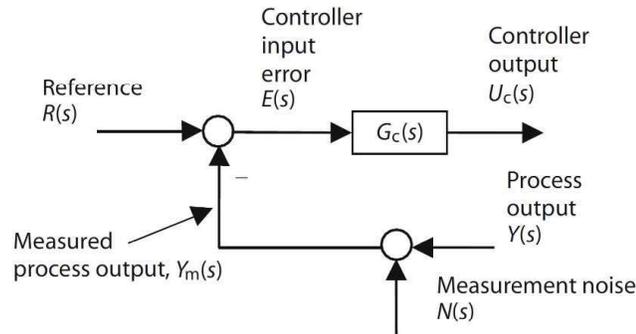
Figure 7: Inputs and outputs diagram for controller

Proportional control is used when the controller output signal is proportionally related to controller input error signal $e(t) = r(t) - y_m(t)$. Representation for proportional control in time and Laplace domain are shown in Equ 1 [10].

$$
\begin{aligned}
Time\ domain &\qquad u_c(t) = \ k_p\ e(t) \\
\\
Laplace\ Domain &\qquad U_c(s) = \ K_p\ E(s)
\end{aligned}
\qquad (1)
$$

Where the K_p, represents proportional gain in given equations. Figure 8 illustrates proportion control block diagrams.

Figure 8: Proportional control term block diagram

I-term is standing for Integral term in the PID controller, and it is used to compensate steady offset from constant reference input signal value. Also, Integral control eliminates proportional terms offset shortcoming, without the use of large control gain. The representation for time and Laplace domain of Integral controller are shown in Equ 2: [10]

15

$$Time\ domain \qquad u_c = k_I \int e(\tau)d\tau$$

(2)

$$Laplace\ domai \qquad U_c(s) = \left[\frac{K_I}{S}\right] E(S)$$

Where the K_I, represents Integral controller gain in given equations. Figure 9 illustrates block diagrams for time and Laplace domain. [10]

Figure 9: Integral control term [10]

Derivative controller brings prediction action into the controller by using the rate of change in the error signal as an input. This action uses the rate of change of an error signal and introduced by D term in a controller. Representation for time and Laplace domain for derivative control shown in Equ 2: [10]

$$Time\ Domain \qquad u_c(t) = k_D \frac{de}{dt}$$

(3)

$$Laplace\ Domain \quad U_c(S) = [K_D S]\ E(S)$$

Where the K_D, represents derivative controller gain in given equations. Figure 10 illustrates the block diagram representation for derivative control [10].
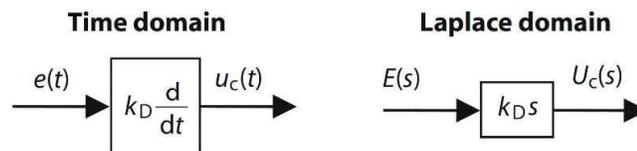
Figure 10: Derivative control block [10]

16

In a real implementation, the use of derivative control needs more care than the use of proportional and integral control. For instance, due to noise amplification in feedback measurement, a modified derivative controller should be used instead of a pure derivative controller. However, derivative action has many useful features in real world implementation [10].

Various combination of introduced three term PID controllers are usually used in process controls to meet specific performance requirements. The combinations constructed with P, I and D controllers which are connected in series or parallel. The basic parallel PID controller formula is given as follow [17]:

$$U_c(s) = \left[K_p + K_I \frac{1}{S} + K_D S \right] E(S) \tag{4}$$

Given formula is the basic formula for the parallel PID controller without any modifications that are necessary to make it work in real process control. For instance, the derivative form is used in pure form, and it is not usually implemented in this form due to noise amplification feature. Some modification should be applied on given parallel PID to make it implementable. The parallel PID controller is also known as *decoupled* PID form because it has three parallel paths. Figure 11 shows parallel architecture for PID controller. [10]

17
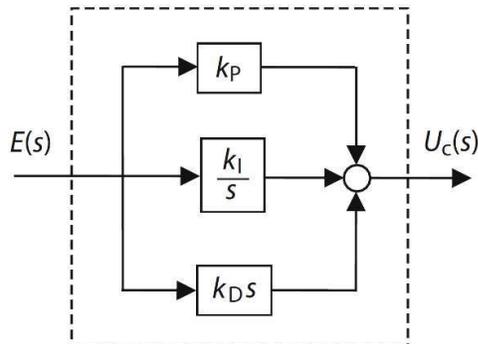
Figure 11: Parallel PID Block diagram [17]

Early industrial PID controllers which are used in pneumatic or hydraulic actuators, had a series transfer function representation for PID controllers. However, modern PID controllers are implemented in digital and parallel, but this series formula is retained by some analog manufacturers to keep compatibility.
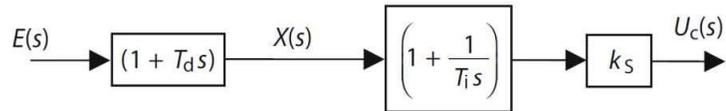
Figure 12: Series PD Block diagram [17]

The very first technology of PID controller was analog. Primary PID controllers had simple dials marked by PID constants to tune the process control manually. There was a simple relation between the response to the plant and adjustment of PID parameters. In PID controller design procedure two-part should be considered in tuning: which terms should be used in PID structure and how to choose a numerical value for PID constants.

To choose the proper structure of a PID controller, understanding the effect of each term is necessary. The heuristic behavior of PID controller is captured in Table 1. The provided information is useful to understand each term of PID controller in transient and steady state.

Table 1: PID Controller Tuning effects [17]

Tracking reference		Disturbance rejection	
Transient	Steady state	Transient	Steady state

18

P	Response speeds up by increasing in $K_p > 0$	Steady state offset remaining but increasing $K_p > 0$ reduces.	Response speeds up by increasing in $K_p > 0$	Steady state offset remaining but increasing $K_p > 0$ reduces.
I	By adding $K_I > 0$ wide range of response type achievable	Adding $K_I > 0$ eliminates reference response offset	By adding $K_I > 0$ wide range of response type achievable	Adding $K_I > 0$ eliminates offset in steady state
D	To tune response dumping, $K_D > 0$ give wide range of response	No effect on steady state	To tune response dumping, $K_D > 0$ give wide range of response	No effect on steady state

2.3.2 Field oriented control

To achieve good performance, more complex control method should be used to control PM motors. With the power of mathematical processing which is offered by new embedded computers, we can use advanced control strategies, which can decouple the magnetization function and torque generation in Permanent Magnet motors. This decoupling of magnetization and torque is called rotor flux oriented control, or Field Oriented Control (FOC) [11].

According to the electromagnetic laws, produced torque in a machine is vector cross product of two magnetic fields [11]:

$$T_{em} = \vec{B}_{stator} + \vec{B}_{rotor} \qquad (5)$$

This equation shows that when stator and rotor magnetic field are orthogonal, torque is maximum. To ensure a better dynamic and reduce torque ripple we should keep

rotor and stator magnetic field in 90° respectively. However, to keep this condition, we should know the exact position of the rotor.

The mission of the Field Oriented Control (FOC) is to vector control of stator current. This method based on a projection of three phase speed and time-dependent into d and q time invariant 2D coordinates. The torque (represented by q co-ordinate) and the flux (represented by d co-ordinates) are the necessary inputs of field oriented control machine [11].
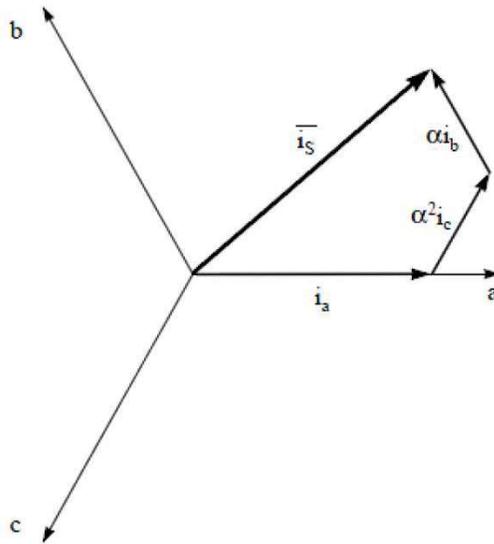
Figure 13: (a, b, c) co-ordinates of stator current [11]

Figure 13 illustrates stator current complex vector. This space vector should be transformed to two-time invariant coordinates. This action can be done in two steps:

1- Convert $(a, b, c) \Rightarrow (\alpha, \beta)$ to make it two co-ordinate time variant

2- Convert $(\alpha, \beta) \Rightarrow (d, q)$ to make it two co-ordinate time invariant

20
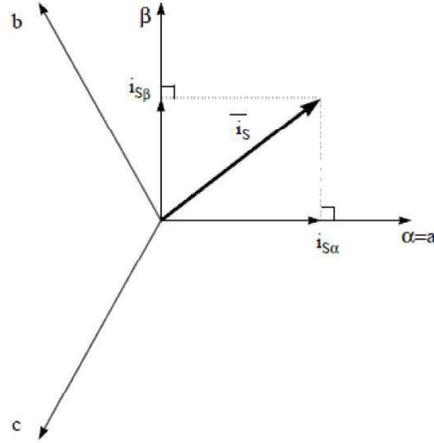
Figure 14: stationary reference frame of stator current [11]

Figure 14 shows the stationary projection (the Clarke transformation) of (a, b, c) currents to two co-ordinate time-variant system.

By using Park transformation, two phase orthogonal system (α, β) modified to d, q reference frame. Figure 15 shows the projection of time variant space vector to time invariant d, q.

Where, θ is the position of rotor flux. Following equation determines the flux and torque components [11]:

$$\begin{cases} i_{sd} = \ \ \ i_{sa} \ cos\,\theta + \ i_{s\beta} \ sin\,\theta \\ i_{sq} = -i_{sa} \ sin\,\theta + i_{s\beta} \ cos\,\theta \end{cases} \qquad (6)$$

By using the right position of the flux, the d, q components become constant. Now both phase current converted to DC quantity. With constant $i_{sd}\ (flux)$ and $i_{sq}\ (Torque)$, torque control would be much easier [11].
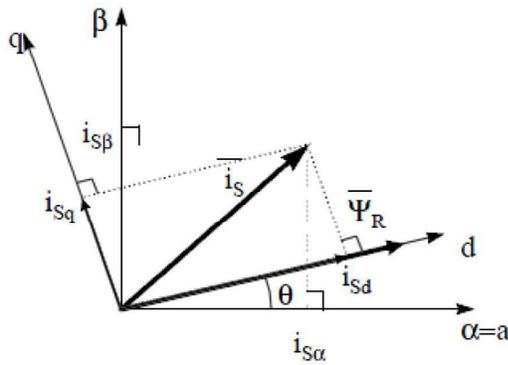
Figure 15: d, q reference frame of stator current [11]

Figure 16 illustrates diagram for basic concept of FOC control. Two phase current measured and converted by Clark and park transformation method to d, q rotating reference frame. By comparing i_{sd} and i_{sq} with reference inputs control signal can be generated. With the advantage of this method controlling synchronous or PM machines is possible with just changing reference flux and determining rotor flux position.
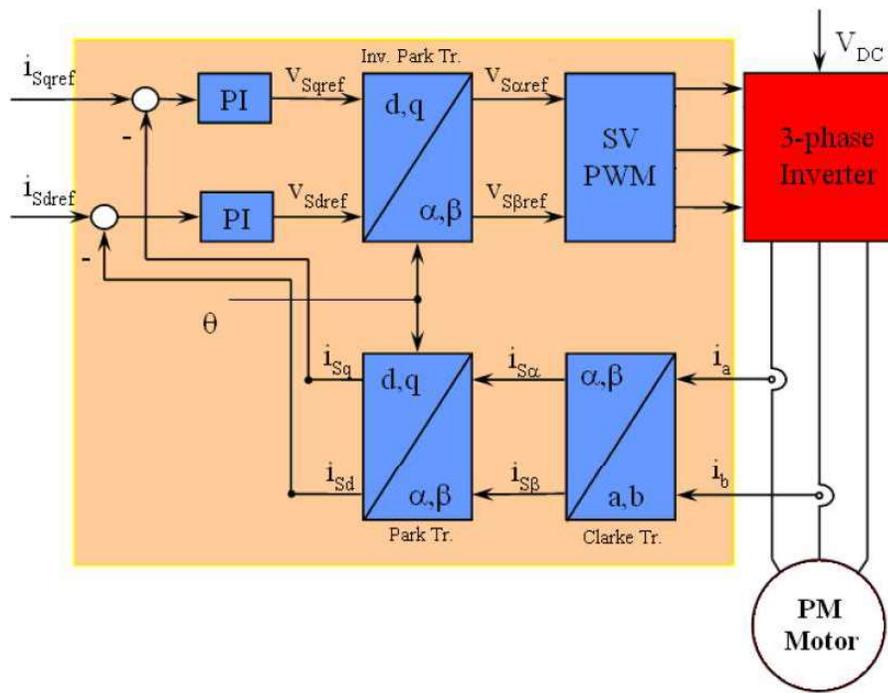
Figure 16: Basic concept of FOC [11]

Chapter 3

METHODOLOGY

The aim of this study is to recreate steering feel and forces from the road wheels on the steering wheel in the steer by wire system (SbW). In this research, a new method of actuation for road tires is proposed. A pair of pneumatic actuators is used to control the tires angle, instead of using servo motors which are used in many studies before. Reproducing steering feel is achieved by getting force feedbacks from the pneumatic actuators and emulate measured force on the steering wheel by using BLDC motor. Three main parts of implementation methodology are as follows: Mechanical, Electrical, and Software.

Figure 17 illustrates an overview of proposed method in this study. Front road wheels are connected to chassis through control arms and a pneumatic cylinder, so Rod-ties replaced by the pneumatic cylinders to control the steering angle of tires. A pneumatic position controller controls the pneumatic cylinder's shafts position to adjust tires angle. An air pressure sensor is used to measures the air pressure inside the pneumatic cylinders to measure forces on the road tires. Pneumatic cylinder equipped with a linear position sensor, to coordinate cylinder shaft position by using the pneumatic controller. Also, the measured position feedback to the steering wheel actuator to set it at the proper angle. The Hand wheel is connected to BLDC motor through a gear mechanism. The gear coupling is used to increase the BLDC torque on hand wheel and increase shaft turning range to achieve high precision

23

controllability. The absolute position sensor is employed to measure steering wheel exact position which is used in the pneumatic controller position command. Main controller estimates desired force should be applied on the steering wheel to reproduce steering feel and send it to the BLDC driver. The BLDC controller unit controls the BLDC motor with desired torque according to the amount of force on the tire.
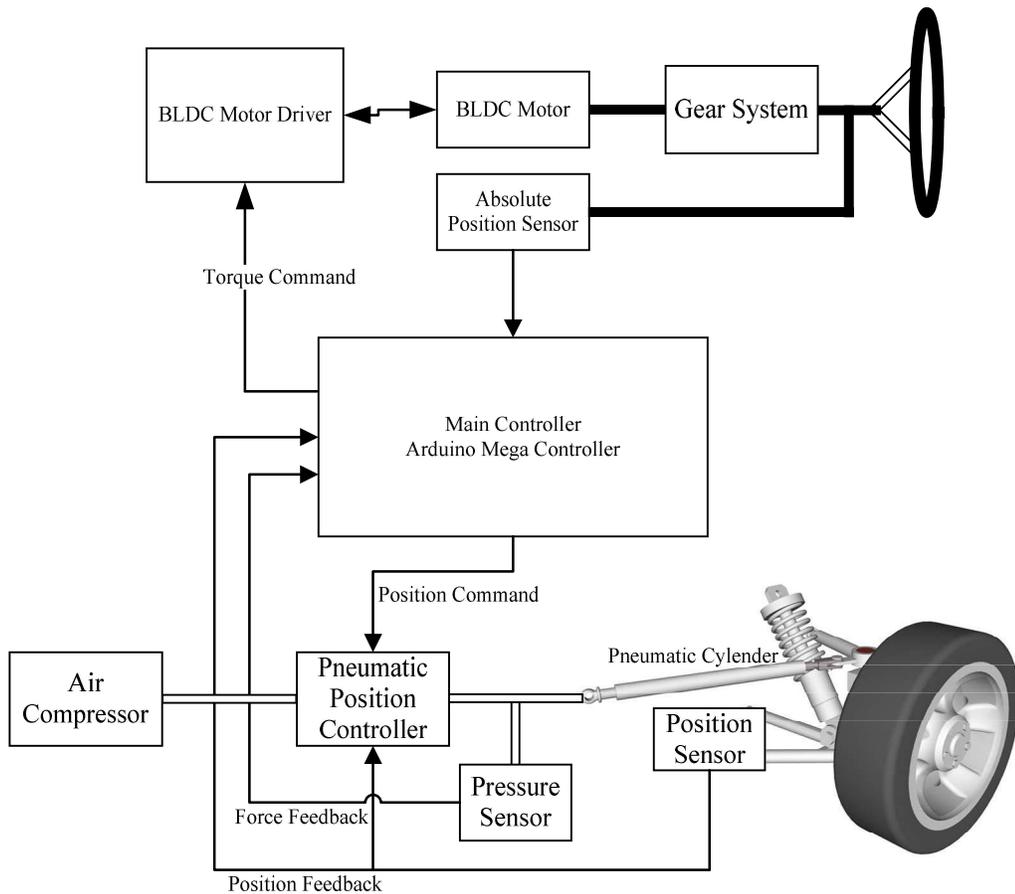
Figure 17: Pneumatic SbW System used in current study

3.2 Mechanical

The mechanical section introduces the mechanical actuation design and the reason behind the selected actuators. Figure 18 illustrates the mechanical design overview for implemented electric vehicle's steering system.
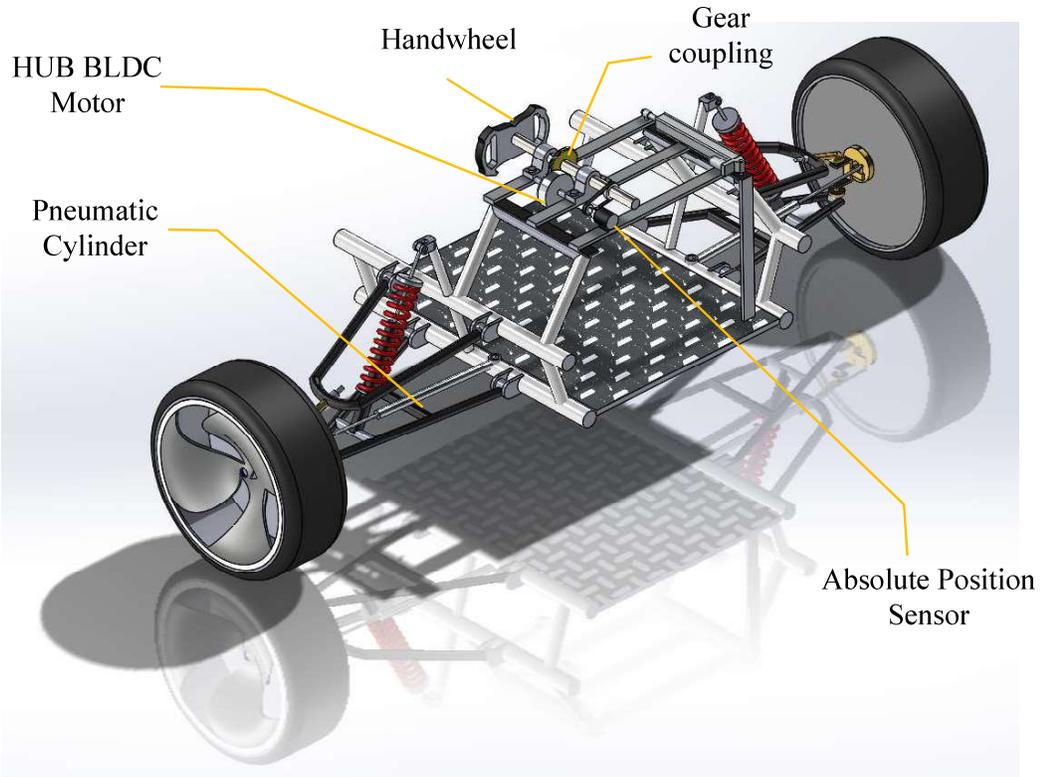
Figure 18: Electric Vehicle steering system mechanical design overview

A single seated Electric Vehicle (EV) is selected to implement the proposed pneumatic steer by wire system which the front tires and steering system are shown in Figure 18. The main parts to play steering wheel role are described in the illustration. The main parts of this implementation are Handwheel, Absolute Position Encoder, BLDC Motor and Pneumatic actuation unit (PAU).

3.2.1 Steering Wheel Actuation

After the methodology design, the very first step is the sizing calculations for actuators and also selecting the proper sensors to achieve good control performance. According to the Rosth study [12], the maximum desired force desired to turn the steering wheel is represented in Figure 19. According to the Rosth research, the maximum applied force on the steering wheel is when the vehicle is in the zero speed or the Parking state. The provided force graph in Figure 19 is for typical road vehicles. The maximum desired torque that should be applied to the steering wheel to reproduce the conventional cars steering feel be presented approximately 4 (Nm).
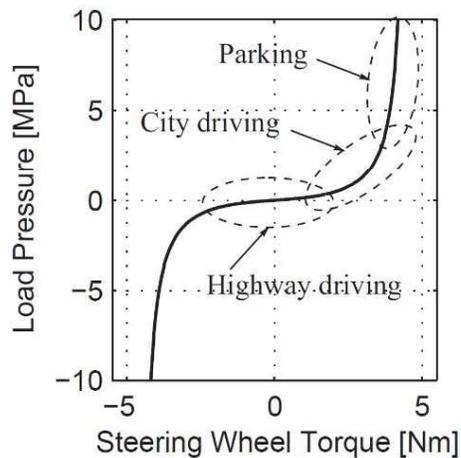
Figure 19: Steering wheel desired force in power assisted steering system[13]

A gear mechanism transfers the BLDC motor's force to the steering wheel. The gear ratio is from the Motor to the steering wheel is 1:5.2 (18:95) respectively. The gear mechanism increases the motor torque and decreases the speed to improve the motor's controllability range at the zero speed and also increase the power. According to Forkenbrock and *et el.* [13] research in the US National Highway Traffic Safety Administration Organization, the maximum steering wheel speed rate in the highest maneuverability depends on the car type and also driver behaviors. The

test has been taken with a different car and four different drivers. The test environment condition satisfies the **ISO 3888-2** standard. The standard defines the test condition for; Test track for a severe lane-change maneuver to avoid the Obstacles. So the highest desired steering wheel maneuverability speed is tested in Forkenbrock's test report.

Table 2: Totally the steering wheel rates (degree/second)

Vehicle	Driver BO	Driver DE	Driver GF	Driver LJ
2004 GMC Savana 3500	708	710	680	642
2004 Volvo XC90 4x4	721	702	683	680
2003 Toyota 4Runner 4x4	839	821	750	792
2002 Chevrolet Corvette	924	940	963	910
2003 Toyota Camry	922	918	908	912

The test is taken with four different vehicles and four different drivers. Table 2 shows the results of tests on the test track with **ISO 3888-2** standard and with disabled Electronic Stability Control system. The maximum steering wheel according to Table 2 is 963 degree per second. The 963 degree/second is equal to the 160 [rpm] for the steering wheel. Hence the steering wheel coupled to the motor by using the gear mechanism and the conversation ratio for the gear coupling is 1:5.2, so the maximum desired speed for the BLDC is 834[rpm]. The given formula [Equ 5] calculates the power for BLDC motor [17].

$$Motor\ Power\ (KW) = \frac{(Desiered\ Torque\ [N.m] \times Speed\ [rpm])}{9.5488} \qquad (5)$$

Where the maximum desired torque according to the Figure 19 is 4 Nm. The maximum desired speed of the BLDC motor according to Table 2 be calculated

834[rpm]. So the BLDC motor power should be more than 0.35 [KW] or 350W to satisfies the maximum maneuverability steering wheel speed. A 350W 3 phase BLDC motor is selected to drive the steering wheel. Figure 20 shows the BLDC motor, and the coupling to the steering wheel through the gear coupling mechanism.
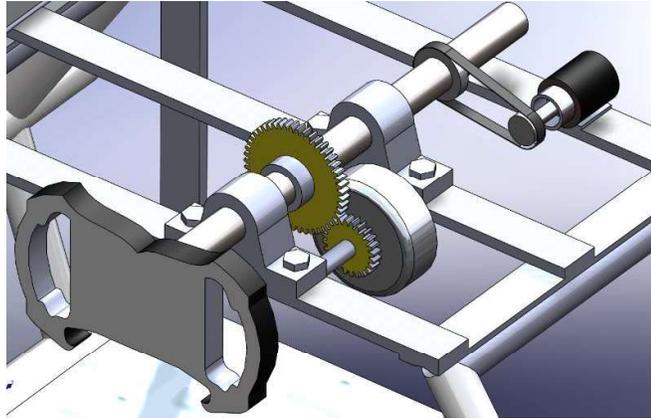
Figure 20: Connected BLDC motor through the Gear Mechanism

The belt coupling mechanism connects the absolute position encoder to measure the accurate steering wheel position. Figure 20 illustrates The steering wheel position sensor and its connections to the steering wheel rod. Theory of operation of position sensor will be described in detail in the electrical section.

3.2.2 Road wheel actuation

In this research as it introduced at the beginning of this chapter, Pneumatic cylinders control the front tires steering angle. The pneumatic should be able to change the tires angle in the full load weight. There is not any fixed force for tires available because the forces on the front wheels are strongly dependent on car dynamics, weight and mass center. Especially, in the three wheel selected car for this research, forces on the tires are different from four-wheel vehicle. Figure 21 shows the

schematic for the pneumatic steering actuator and the connections between main items.
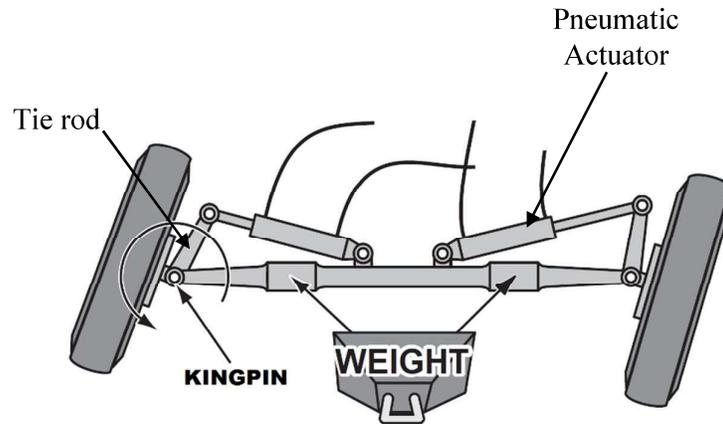
Figure 21: Front-wheel pneumatic steering system's schematic

The very first step in cylinder sizing calculation is to find the force on the KINGPIN. The Tie rod converts the pneumatic cylinder's linear force to the amount of necessary torque to change the tires angle which is connected to KINGPIN point. To calculate the required torque to change the tire angle a simplified formula is given as follows [12]:

$$T = Wu \sqrt{\frac{B^2}{8} + E^2} \qquad (7)$$

Where **T** is kingpin torque in inch/lbs. **W** is vehicles weight on the front tires; **u** is friction coefficient. It can be assumed 0.7 [12] for most applications which are using normal car tires. **E, B,** are tire size and distance from Kingpin which illustrated in Figure 22.
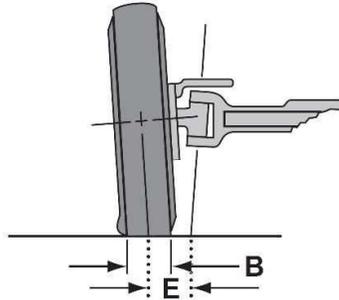
Figure 22: Road wheel steering system front view

In the selected electric vehicle in this study, W is 84 Kg, B is 104 mm, E is 82.2 mm. So the required torque to change the tire steering angle on the Kingpin calculated 52 Nm. To convert a rotational force to the linear force the given formula is used as follow [12]:

$$F = \frac{Torque}{r * sin\theta} \tag{8}$$

Where **F** is required linear force should be generated by the pneumatic cylinder in Nm, **r** is the connection arm length between the pneumatic rod and the Kingpin which is illustrated in Figure 23. Moreover, the θ is the angle between linkage arm and pneumatic actuator's rod.
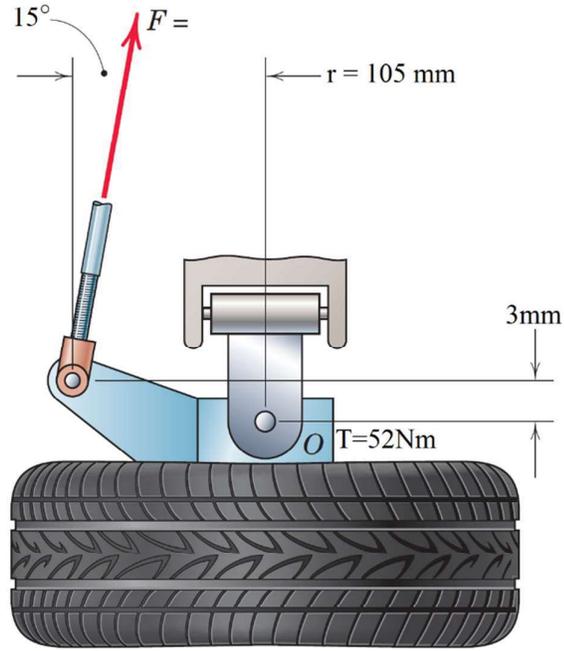
Figure 23: Pneumatic cylinder connection [12]

The required force from pneumatic actuator according to the given parameters in Figure 23 can be calculated by using Equ 8 as follows:

$$F = \frac{52Nm}{0.105 \; Sin \; (75)} = \; 508 \; [N] \qquad (9)$$

The last step in pneumatic cylinder calculation is to find the bore size of the cylinder which can be calculated with given formula [12]:

$$d = \sqrt{\frac{4F}{\rho\pi}} \qquad (10)$$

Where ρ is pneumatic air pressure in Pascal and F is desired Force from pneumatic cylinder and d is the pneumatic bore size in millimeter. The result for pneumatic cylinder bore size is 28.4 with safety factor 2. So the nearest standard value for the pneumatic cylinder is chosen by 25mm.

3.3 Electrical

Data acquisition and control system to implement SbW system in this study has three main blocks. Namely, steering wheel control block, road wheel actuator and sensors and central control unit to control all actuators.

3.3.1 Steering Wheel Position Sensors

In the steering wheel control part, there are two main actions should be considered, they are divided to control action and measurement actions. The control action is used to apply force on the hand wheel by using BLDC motor, and the measurement action is to measure hand wheel current position. An absolute position sensor measures the steering wheel shaft position which is used in road wheel position controller. BLDC driver controls motor torque by using measured motor current. Figure 24 illustrates an overview for hand wheel control.
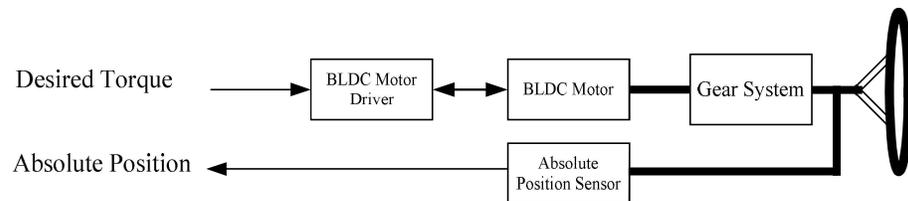
Figure 24: Steering wheel control diagram

The absolute position sensors output is necessary to control the tires angle and also it is used to achieve best position control performance in the motor driver. To take angle measurement an industrial absolute rotary encoder from the Autonics is selected. The selected sensor in this research has the *"EP50S8-1024-3-R-N-24"* Part number, and it is shown in Figure 25.

Figure 25: EP50S Series absolute rotary encoder [14]

The selected sensor has the resolution of 1024 steps per turn, and its data can be accessible through digital output with the gray coding standard. The ***reflected binary code***, also known as ***gray code***, is a digital numerical system where two sequential numbers are different with just one bit. The gray code is commonly used in error correction in the digital data transmission systems and industrial applications. The EP50S sensor is used reflected binary code to send position data out. So it has ten digital output line to be able to send out 1024 steps. Moreover, also it needs a power supply to power up digital input circuits. The absolute rotary encoders are using special kind of an optical disk to generate binary, BCD or gray outputs and also some absolute encoders use industrial serial data communication protocols to send out position data. The gray code generator optical encoder disk is shown in Figure 26.
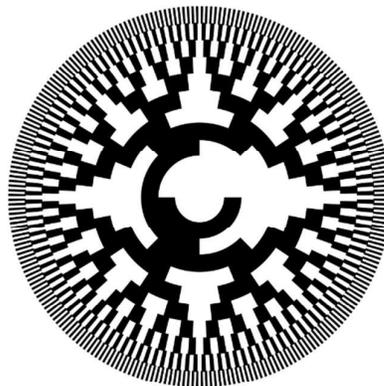
Figure 26: Optical Encoder disk for gray code [14]

In the current research, the road wheel position controller uses the position value from the absolute encoder as a position feedback to control the angle of tires. However, the position sensor's output should be translated to controller compatible inputs. As it will describe in this chapter, the pneumatic controller needs 0 to 10V for linear actuator control. Hence the output from the position sensor is not compatible with controller's acceptable input so it should be translated from digital gray code to analog zero to 5 volt for the pneumatic position controller to the 0-10V range, The position sensor's data conversation block diagram is illustrated in Figure 27.
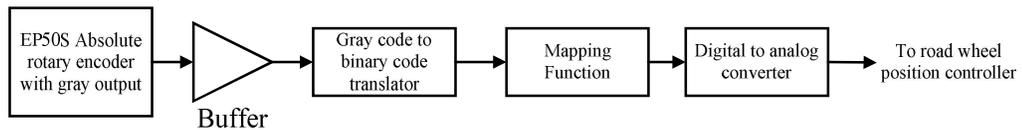
Figure 27: Position sensor data converter diagram

An Arduino board is used to implement illustrated block diagram in Figure 27. The 74HC14 Integrated Circuit (IC) is used as a buffer to convert encoder output signal to TTL level for the microcontroller. The gray code to binary converter and mapping function blocks are implemented in microcontroller code which will be described in detail in software part. Digital to analog blocks are implemented by using Pulse Width Modulation output from Arduino board and passes through an analog filter to converted to DC value. The DC analog value is amplified and buffered by LM358 Operational Amplifier (OP-AMP) IC to the proper signal level. The PWM to DC signal filter and analog buffer is shown in Figure 28.

34
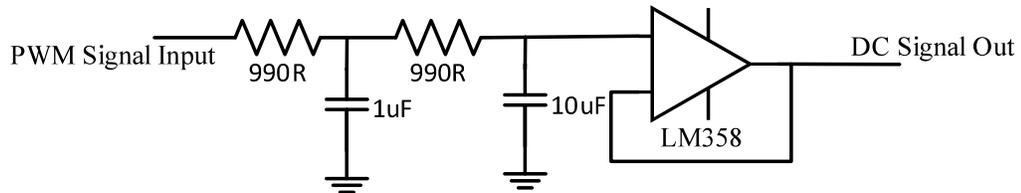
Figure 28: PWM to DC analog value converter

The cut-off frequency for illustrated analog filter in Figure 28 is 36.6Hz as it is shown in frequency response analysis in Figure 29. So this filter is ideal to filter the PWM signal and convert to suitable DC signal which can be used as feedback and reference.
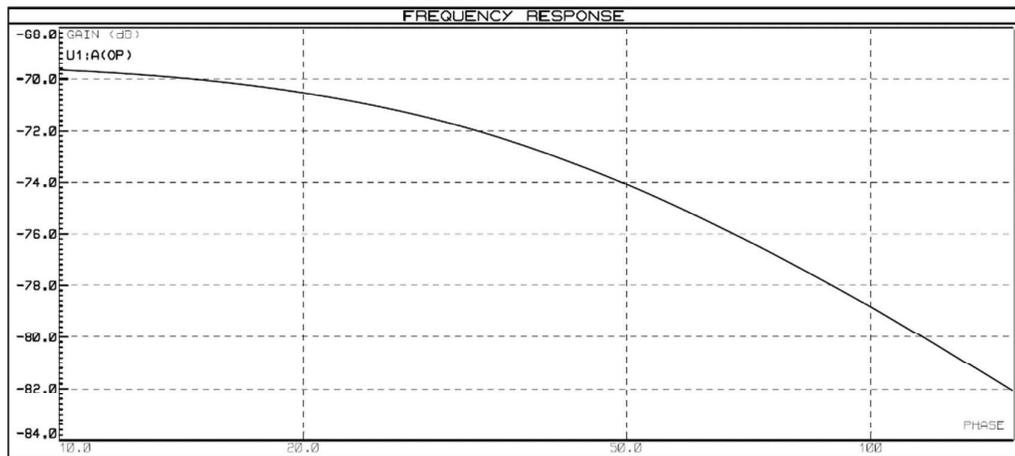
Figure 29: Frequency response analysis for Analog filter

3.3.2 Steering Wheel's BLDC Motor Torque

To emulate the steering feel forces on the steering wheel, we need an actuator to play the steering column roll. In this study, chosen actuator is a Brushless DC (BLDC) motor which is connected to BLDC driver to control and drive the motor. The BLDC driver should be capable of controlling the amount of force on the Hand wheel. As it was discussed earlier in this chapter, a bicycle HUB motor is selected as the steering

wheel actuator. This HUB motor can deliver up to 350W power to the shaft with 36V nominal working voltage.

According to the HUB motor specifications, the BLDC driver should be able to supply up to 10A and work with 36Volt. The FOC method is selected to drive BLDC motor and make sure that the motor is working with maximum adjusted torque. The FOC algorithm and torque control working with motor current, so accurate current measurement necessary in motor controller side.
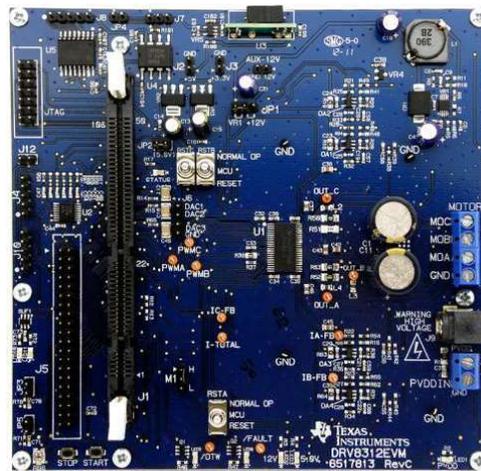
Figure 30: DRV8312-69M-KIT[11]

In this research to control BLDC motor and firmware development, the DRV8312-69M-KIT development kit chosen and it illustrated in Figure 30. The development kit equipped with TMS320F28069M Digital signal controller (DSC) from Texas Instruments (TI). The 28069M is one of TI's motor control solutions which is optimized for motor control applications to achieve better performance regarding measurement and control calculations. Power drive unit in the DRV8312-69M-KIT development board based on DRV8312 3-phase BLDC motor drive solution from TI.
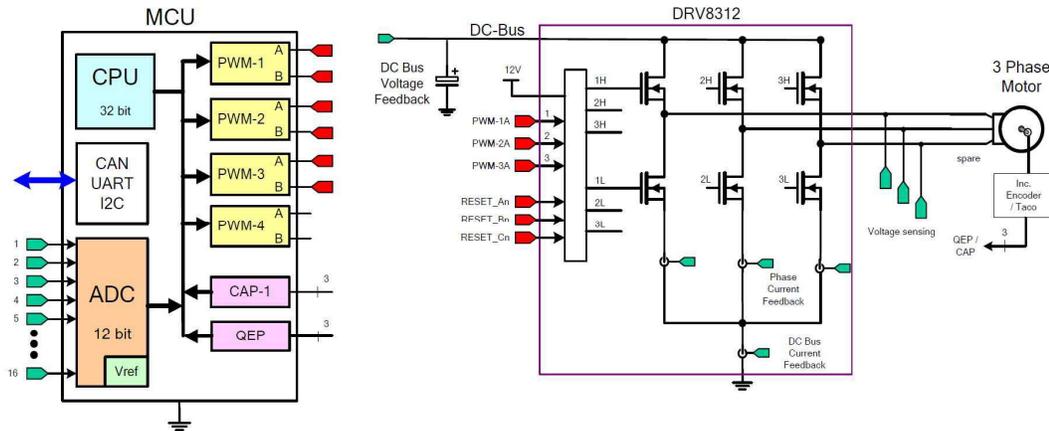
Figure 31: BLDC Motor development kit block diagram [11]

Figure 31 illustrates the general hardware overview for the BLDC driver. The three-phase BLDC motor needs three half bridge driver to source and sinks current from motor stators. The DRV8312 has a three built in half bridge controller. Current shunt resistors are necessary on each leg's ground side to convert the motor current to a voltage that measurable by the MCU's Analog to Digital converter (ADC). FOC and torque control algorithms are using measured currents from motor phases. The DRV8312 has three PWM signal input to control high side switches independently, and also, it has three reset inputs to control low side switches as well. All measured currents by using external current shunts are feedback to the microcontroller to be used in FOC and Torque control process.

Main Micro Controller Unit (MCU) provides six complementary PWM outputs to control half-bridge drivers. The PWM units inside the microcontroller generate high-frequency PWM signals with desired dead time. Two inputs are considered to get pulses from incremental rotary encoders to measure speed and shaft movements. Moreover, 16 Analog inputs are available to measure the analog signal in the range of zero to three volts.

Eight channel of sixteen analog inputs are allocated to measure motor voltages and currents, and the remaining channels are wired out for adding other functions by users. In this research, an analog input is used to get data from Handwheel position sensor. According to the BLDC motor's theory of operation is described in the second chapter, the six-step of a BLDC motor control procedure is shown in Figure 32.
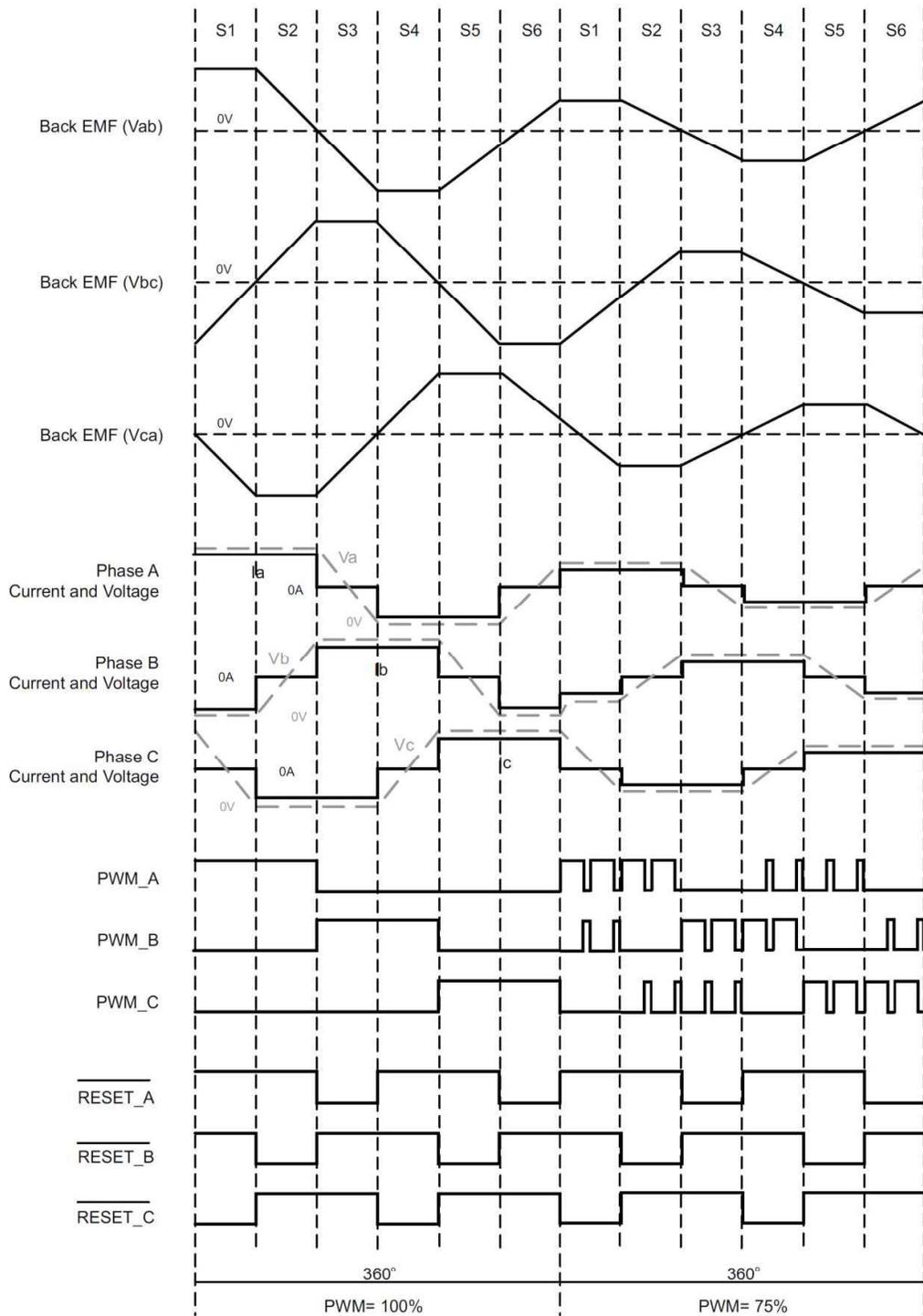
Figure 32: Trapezoidal scheme control in six steps [11]

In sensorless designs, the controllers are using back electromotive force (Back EMF)

signal to estimates the rotor position to select the correct stator coil to switch it on or

off. Figure 32 shows, two different power level is applied to the stator coils. All currents and the back EMF's for each phase illustrated respectively. Figure 32 demonstrates that the PWM signals control the high side switches, and the low side switches are just working in on or off states. Three resistor voltage divider measure the voltages on each line of motor inputs to extract the Back EMF signals for each phase. Figure 33 illustrates the voltage divider in the current design. A 47nF capacitor filters the input signals to reduce switching noises and improve the ADC measurement by increasing sample and hold unit performance.
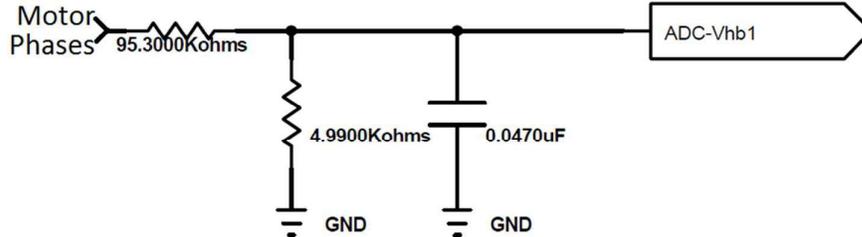
Figure 33: Back EMF Voltage divider [11]

Power MOSFETs inside the DRV8312 amplifies the current and voltage to drive BLDC motor. In another word, the half bridge configured power MOSFETs, translate the MCU 3.3V output voltage to nominal motor voltage and also amplifies the deliverable current. Figure 34 shows the connection between the microcontroller unit and the power MOSFETs inside the DRV8312.
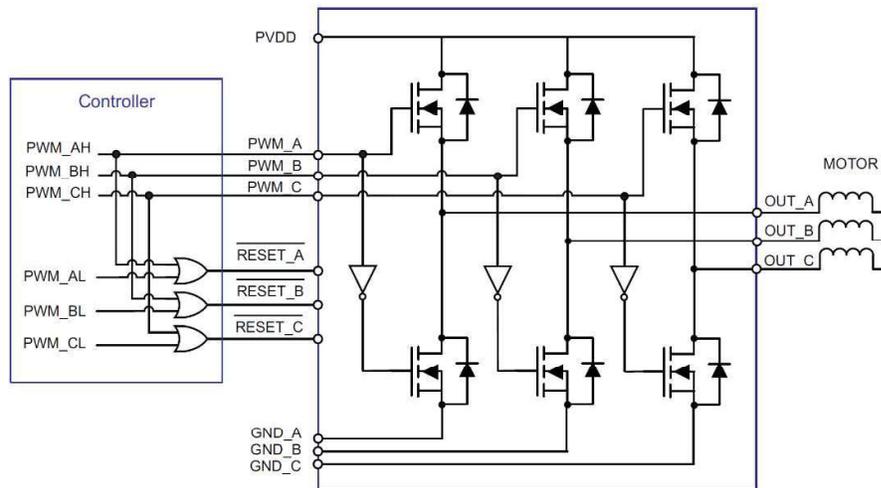
Figure 34: 3-phase Motor driver schematic [11]

The FOC algorithm needs at least two phase current for the Park and the Clark calculations. In the current design, each half bridge's current passes through the shunt resistors which connects each half bridge to the ground. Three ground lines which are illustrated by GND_A, GND_B, and GND_C, wired out from the IC to tie to the ground line through the shunt resistors.

The shunt resistors usually have low resistance and high power to measuring the passing currents to or from the consumers. Shunt resistor's values are tiny; their typical value is between one to some hundred milliohms. The availability, low cost, good frequency response, and wide bandwidth are the advantages of using shunt resistors in designs. In another hand, ground shifting in case of using shunt on the ground lines, and high voltage common mode in case of using a positive voltage, non-isolated, power losses, and low voltage measurement noises are the disadvantage of using shunt resistors in current measurement. A small value of resistors helps to improve the drawbacks especially ground shifting, but the measurable voltage is decreased in return. Also, measurement noises increased due to amplification in the

41

measured shunt voltage. So there is a trade-off between power losses and the voltage drop across the shunt resistor. Measurement improves by designing good filter and using low noise amplifiers.
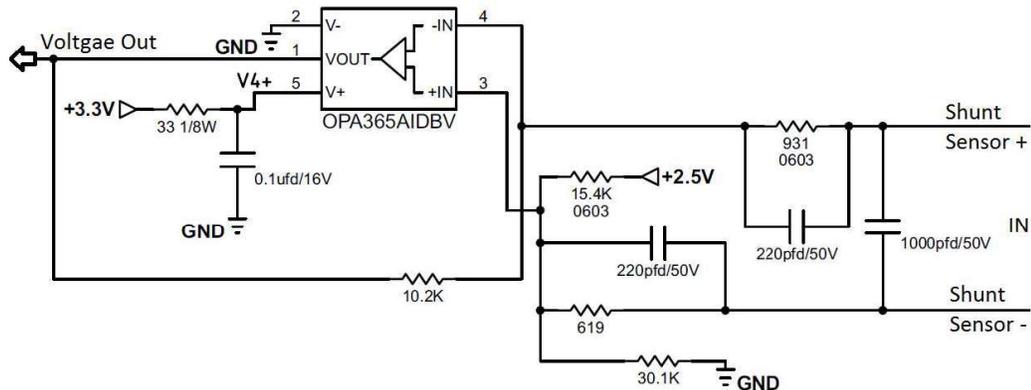
Figure 35: Shunt voltage measurement amplifier [11]

The differential amplifier to measures the shunt voltages is illustrated in Figure 35. The amplifier unit shifts the input signal and also amplifies the input signal to increase the measurement resolution. The shift value is to measure the current in both sink and source way. When the load uses the current, the voltage value of the shunt resistor is positive with respect to ground and in vice versa, if the load passes the current through itself, the current shunt's voltage become a negative value. The microcontrollers can only measure positive voltage. In the current design, input signal sum ups with 2.5V to shift whole measured signal range into the microcontroller acceptable range. Finally, the output from amplifiers goes into ADC to convert to a digital value and used in FOC algorithm and torque control feedback.

All digital and analog parts need a power supply to power up. Hence, the difference voltage levels are necessary for each part, different power supply unit with different output voltages considered in this design. The very first voltage converter decreases

42

the 15-50V input voltage to 12V. This DC to DC level converter is configured in the buck switching topology with the use of TPS54160 switching controller and embedded power switch. The 12V voltage employed in DRV8312's gate drivers for integrated MOSFETs. Figure 36 illustrates the DCDC voltage converter with TPS54160 controller from TI.
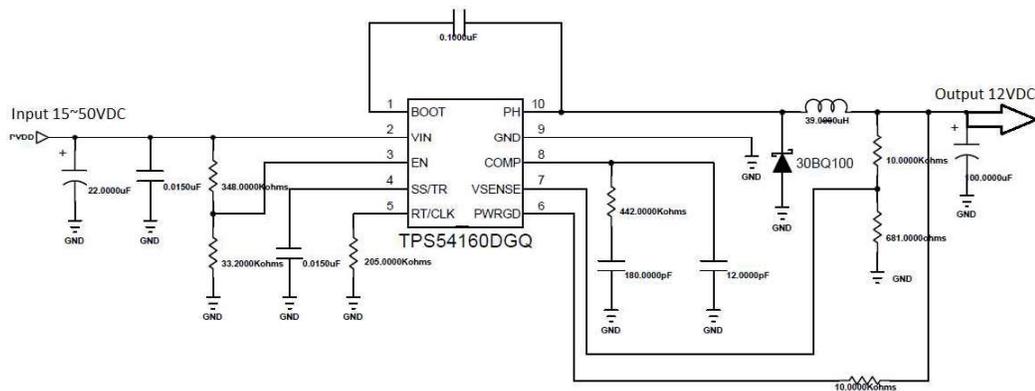
Figure 36: DC to DC converter [11]

Two linear voltage regulators considered to converts the DCDC 12V output voltage to 5V and 3.3V. The outputs from Linear Regulators supplies the analog amplifiers and control units.
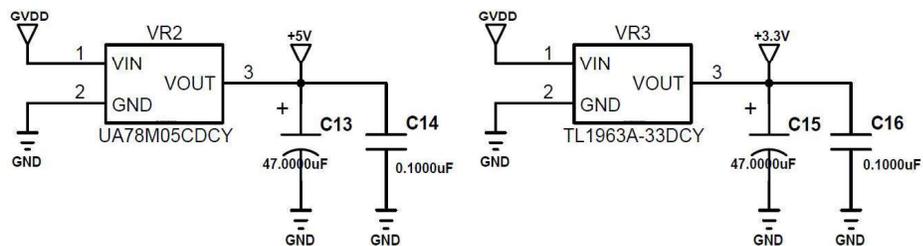
Figure 37: Linear regulators to make 3.3V and 5V [11]

In the industrial applications, the safety and noise immunity are very crucial. In many designs in industrial and automotive fields, the inputs and outputs are isolated by

using opt isolators, galvanic isolations or transformers. The isolation helps to protect from incoming and outgoing high voltage surges and also to prevent the system from electromagnetic noises. In the BLDC motor driver, the serial communication interface is isolated by proper isolator ICs. The isolator ICs needs a power supply on both sides of isolation, so an isolated voltage should be produced on board, to power up the isolated side of the isolator IC.
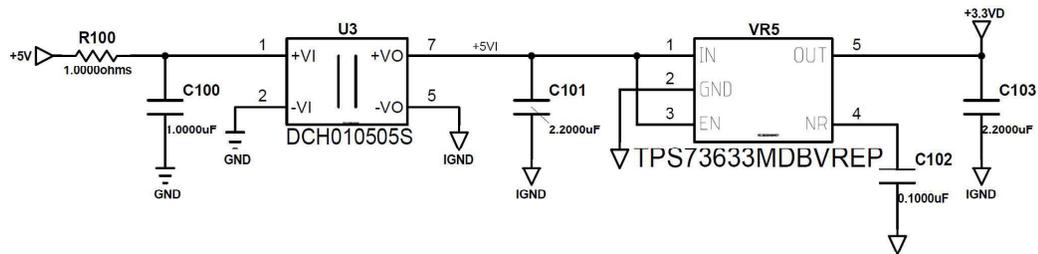
Figure 38: Isolated voltage converter [11]

The DCH010505S is an isolated DC to DC converter to converts the input 5V voltage to separate 5V Voltage at the output. Moreover, a linear regulator converts the 5V to 3.3V to supplies the serial interface ICs. Figure 38, shows the isolated DC to DC converter's schematic.

In the current design, A Controlled Area Network BUS (CANBUS) used to send and receive the commands to the BLDC motor driver. The CAN Bus is one of the most noise free and reliable communication methods in industrial and automotive environments. The physical communication layer based on differential signal transmits, like RS-485 and other differential serial protocols. Moreover, the software layer has CRC error check to control the incoming packets. CAN Bus communication will be described in detail in software part of this chapter. The

isolated digital to differential data converter for CAN bus is sends out or receives the commands by using ISO1050 IC.
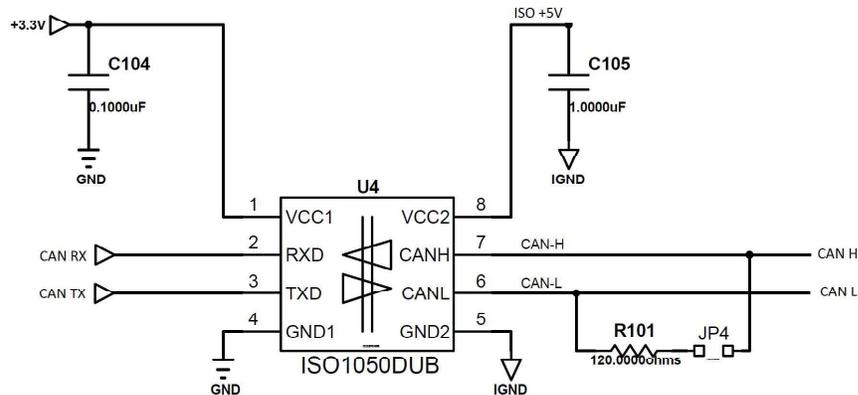
Figure 39: Isolated data communication interface [11]

The CAN bus communication needs a 120-ohm resistor at the both end of the BUS to terminate signals and prevent the bus from reflecting back the signal to BUS.

3.3.3 Road wheels linear position sensor

As introduced at the very beginning of this chapter, a new steering wheel actuation method proposed in this research. In the proposed method in this study, the pneumatic linear actuator takes the control of tire angle adjustment instead of using electrical servo motors. A position controller controls the linear pneumatic cylinder's shaft position by using mounted position sensor on the cylinder as position feedback. The reference signal to adjusts the rod position set by a main control unit that calculates the amount of shaft's linear movement according to the steering wheel position. The selected position controller is an S2 pneumatic controller from the Enfield company. The S2 controller included a closed loop position controller by using linear position sensor to determine the actuator's shaft position and control it with PID controller. The user interface software for the S2 controller gives access to the PID controller parameter and coefficients to adjust them according to the

application's environment. The pneumatic position controllers are using pneumatic servo valves to control the amount of pressure on the cylinder's both end to set the shaft in a proper position.

There are many kinds of linear position sensors available. Namely, Resistive potentiometers, Linear variable differential transformers (LVDT), Magnetostrictive, Optical encoders and Magnetic encoders. The resistive models have a fair accuracy and price, and LVDT models have an excellent precision, but their prices are quite high. The most widely used linear position sensors in the industries are the resistive and LVDT models. In the current study, LVDT model selected to achieve accurate position control and high precision data analysis and results. The LVDT sensor's theory of operation based on Transformers action. There are primary and secondary coils inside the sensor. The position measured by moving the core between primary and secondary coils where this moving changes the magnetic couple on the secondary coil. Figure 40 Illustrates the electrical schematic for an LVDT position sensor.
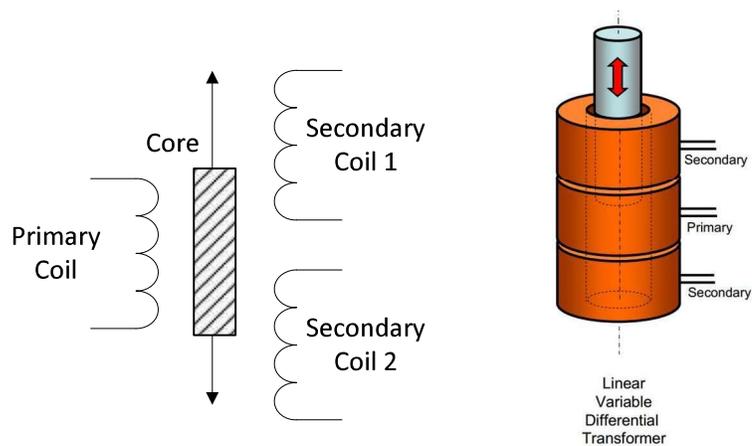
Figure 40: Electrical schematic for LVDT [14]

Figure 41, demonstrates the LVDT position sensor's windings, core and electronic signal conditioning module.

Moving core

Windings

Electronic Signal conditioning
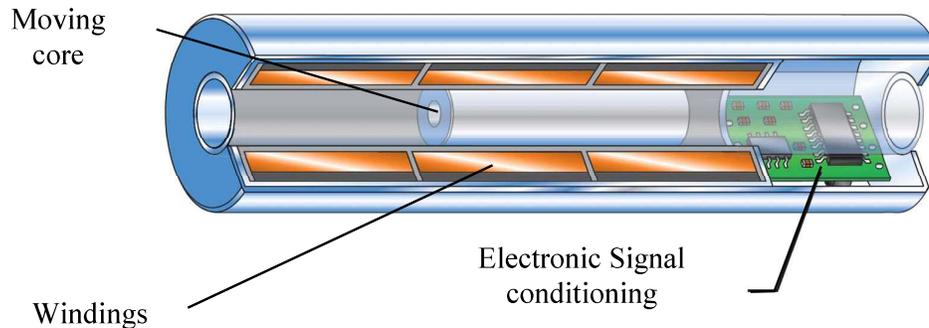
Figure 41: An LVDT Linear Position Sensor [14]

The position sensor needs 12 to 24Volt power supply to power up the electric units, and position feedback output in the selected version is 0 to 10V voltage output which is compatible with industrial controllers like Enfield S2.

3.3.4 Road wheels Pressure sensor

The general method to measures the forces on the tire and steering rack is the road wheel servo actuator current measurement but hence in this study instead of using servo motors the pneumatic cylinder selected to controls the tires angle, so the current measurement method is not available this method. The proposed method using cylinder pressure measurement to determines the forces on the road wheel. The main control unit measures the amount of pressure inside the cylinder and estimates the proper force should be applied to the steering wheel. Figure 42 illustrates the selected sensor in this research. The PSE510-T01 works with 12-24V power supply and represents the measured pressure value in the range of zero to five volts.

Figure 42: SMC Pressure sensor[14]

3.3.5 Road wheels actuator controller

The selected pneumatic controller from Enfield is one of the must well engineered high accurate position controllers in the market. The S2 controller uses the servo valves to control the amount of pressure inside the cylinder. The Linear position sensor provides the shaft's current position feedback and according to the provided feedback the controller adjusts the pressure inside the cylinder to get the shaft in the proper position. The S2 has a robust controller to control the position and also keep the position fixed due to external forces which applied to the shaft. An adaptive robust PID controller controls the rod position. A user interface software comes with a controller to adjust all PID parameters and also to set the environment control limitation such as position limits. The USB port connects the S2 controller to user interface software, USB port located on the driver. The driver works with 12 to 24 volts and accepts the commands in the zero to ten volts range. The position sponsor's analog data comes to controller directly. Figure 43 shows the S2 controller from Enfield.

Figure 43: Enfield S2 Controller [14]

The S2 controller has an input for main high-pressure air and two output which should be connected to pneumatic cylinder both sides, and there is two exhaust output on the controller. Each tire has a set of position controller, a position sensor, and a pneumatic cylinder.

3.4 Software

A TMS320F28069M Digital Signal Controller (DSC) from the Texas Instruments controls the steering wheel BLDC motor and reads the steering wheel angle position from the Absolute Rotary Encoder. The DSC collects feedback signals from the angle position sensor, the current sensor, the motor voltage sensors, the input voltage and the road wheel position sensors and uses all gathered signals as input for control algorithms. Three PI controllers are controlling the steering wheel state. Each controller and the purpose of using that will be described in Control strategy part.

3.4.1 Torque Control Strategy

The force on the tires and the road wheel current position, are two main parameters that should be feedback to the driver to emulate the real steering feel on the steering wheel. The steering wheel control scheme is shown in Figure 44.
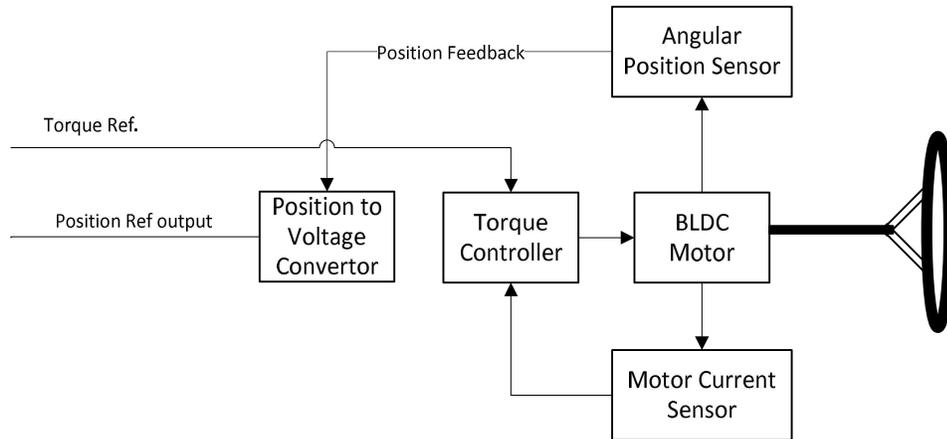
Figure 44: Steering wheel control scheme

A series Proportional-Integral (PI) controller, which is implemented in the DSC firmware to control the forces is shown in Figure 45.
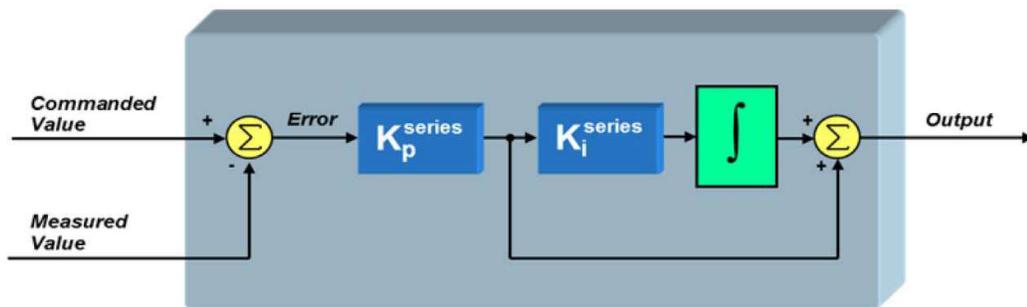
Figure 45: Series PI Torque Controller [11]

According to the Figure 45, the K_i and K_p for the given series controller can be calculated from parallel PID form as follow:

$$K_p^{series} = K_p$$

$$K_i^{series} = \frac{K_i}{K_p}$$

Where K_i and K_p are the parameters from parallel PID controller. In the series PI controller form, the gain for ALL frequencies is set by K_p^{series} , and the K_i^{series} defines the inflection point (zero) in the series controller in rad/sec.

Texas Instruments Co. provides an InstaSPIN motor control library for the TMS320F28069M microcontrollers which has lots of control functions for FOC, FAST and PID control for BLDC motors. The main block diagram for the InstaSPIN library is shown in Figure 46.
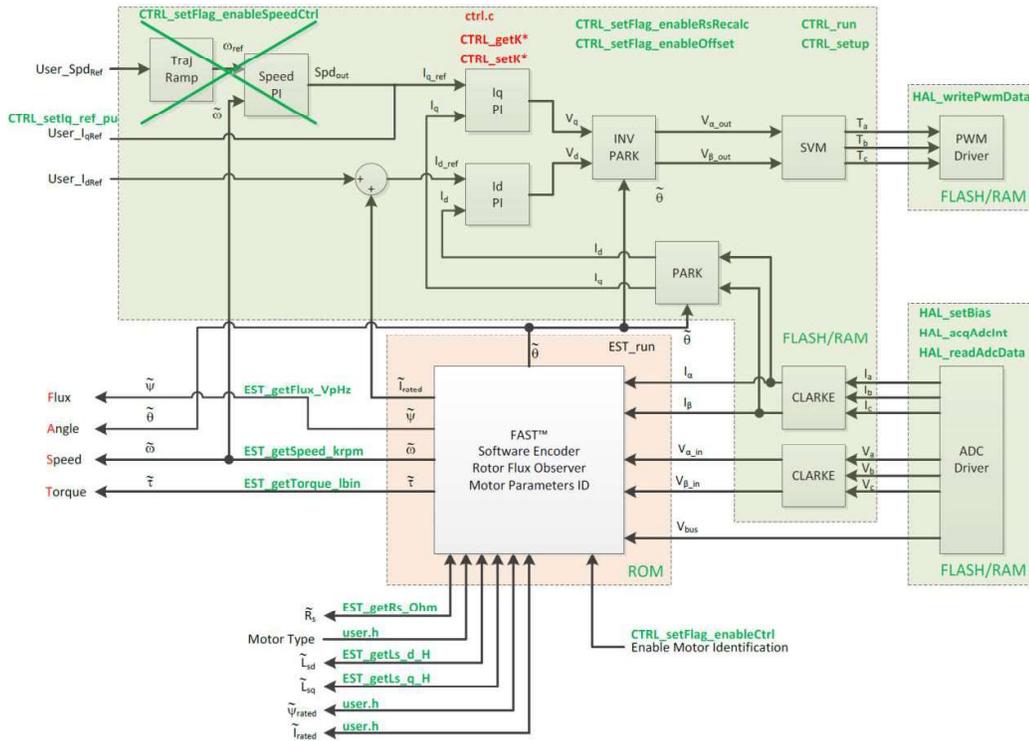
Figure 46: InstaSPIN controller block diagram

There are two current controllers to controls the *id* and *iq* which are introduced in the FOC chapter of this thesis. The Park, Clarke and the SVM conversation modules for the FOC implemented inside the microcontroller hardware to increase the speed and efficiency. The FAST (Flux, Angle, Speed, and Torque) estimation block, estimates

four fundamental parameters to control the motor speed and torque as well as providing the force and angle to use in position controlling.

The Fast algorithm estimates the motor's four fundamental state by using the motor's phase currents and phase voltages and also the main bus voltage as well as motor parameters like the inductance, resistance and motor type, and poles which should be set by the user.

In the current research, the FAST algorithm and FOC controller of the TMS320F28069M are employed to controls the amount of torque on the steering wheel. The desired torque amount on the steering wheel set by a pressure sensor on the pneumatic cylinder and the input from pressure sensor used as torque set point in the controller. The iq and id define the torque limit on the BLDC motor's shaft, so the torque set point defines the current limit value for the current controllers.

3.4.2 Position measurement and conversation

As it is illustrated in Figure 44, the absolute position sensor measures the Handwheel angular position. An Arduino board reads the absolute sensor and translates the digital sensor output to the road wheel controller voltage level. The angle sensor provides the 10-bit digital gray code for the Arduino board which should be first converted to the binary value to be meaningful for the digital to the analog unit.

Start

Read the digital inputs

Gray to Binary convertor

Normalize and Mapping to the DAC Range

Set value on the PWM
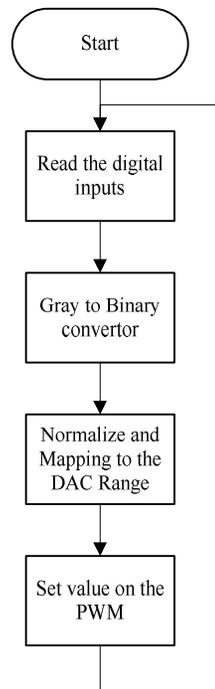
Figure 47: Absolute position sensor to analog value procedure

As it discussed earlier in this chapter in the electrical section, the Pulse Width Modulated (PWM) signal passes through a low-pass filter to generate desired analog voltage for position controller.

Chapter 4

RESULTS

4.1 Experiment setup

The main objective in this study is to implement the new idea for the SbW system. The pneumatic actuator is employed to controls the road wheel angle instead of using servo motors. The pressure sensor on the pneumatic cylinder measures the lateral force on the tire and convert them to the analog voltage which is meaningful for the steering wheel BLDC torque controller. Moreover, the steering wheel position is measured by absolute position encoder and converted to the desired voltage for the pneumatic controller.

- Steer by Wire (SbW): Removing any mechanical linkage between the steering wheel and the road wheels.

- Steering wheel position determination and translate to the road wheel controller compatible input.

- Creating the desired force feedback onto steering wheel

BLDC
Motor

Pneumatic
actuator

Absolute
Position
Sensor
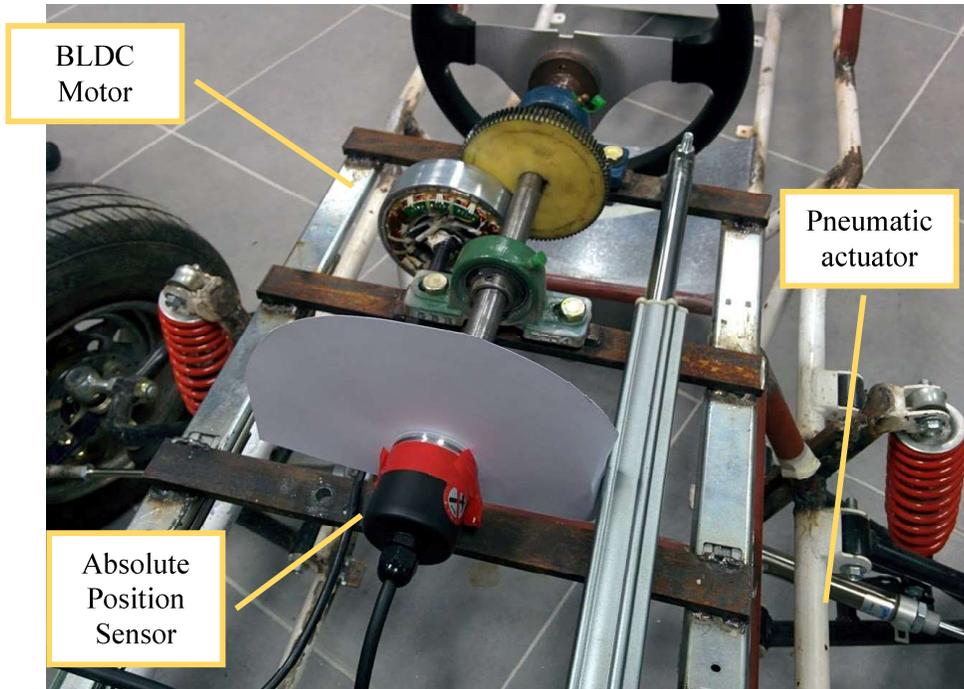
Figure 48: Steering wheel test bed

Figure 48 shows the steering wheel sensor and actuator implementation and connections for this study. A steel rod connects the steering wheel directly to the position sensor. The gear coupling connects the BLDC motor's shaft to the steel rod. The pneumatic actuator connects the tire kingpin to the vehicle chassis.
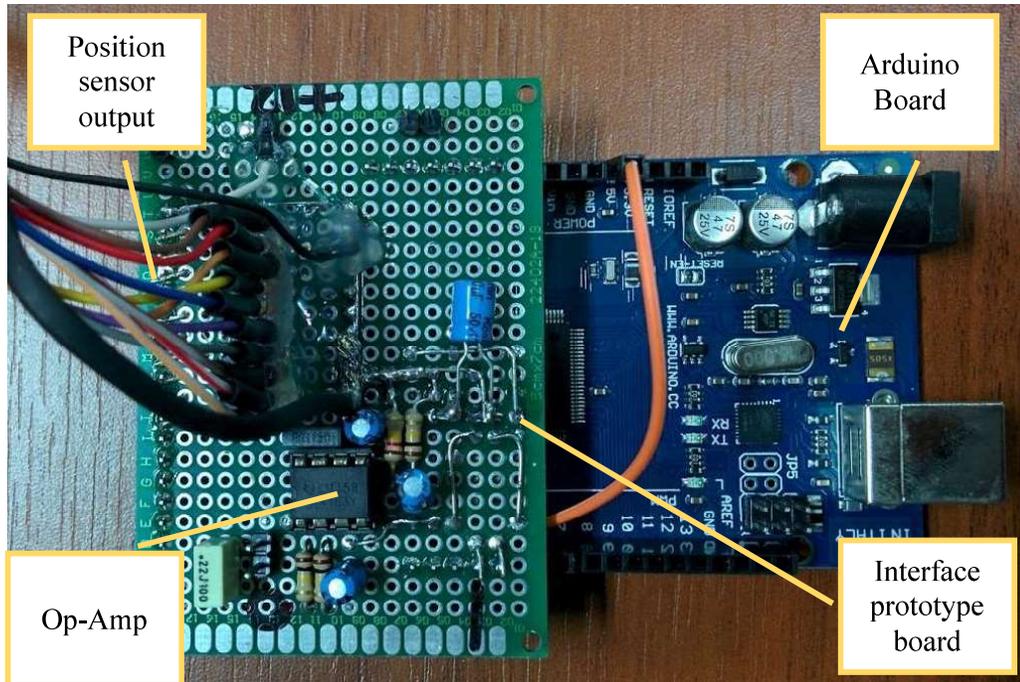
Figure 49: Position sensor data to analog convertor

Figure 49 demonstrates the position sensor data convertor to the analog value for the pneumatic position controller. The signal high level for the position sensor is 24V, but the Arduino board can accept the signals on the digital input just high as 5V. So a resistor divider network reduces the voltage level in the accepted range by the Arduino.

The operational amplifier buffers the output voltage from low pass filter to prevent the signal from changes due to current sink from external sources.
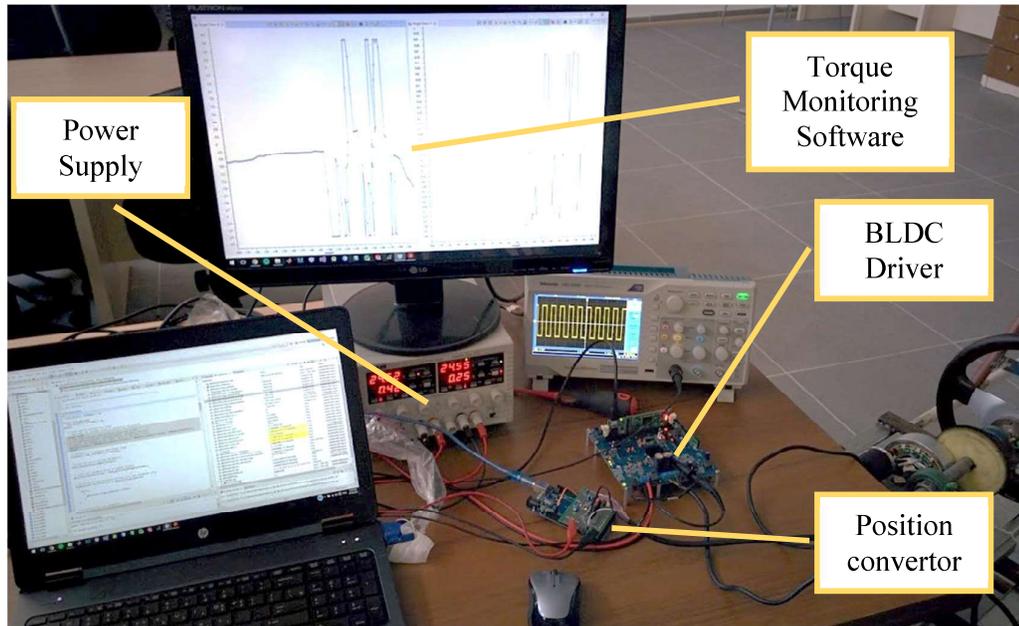
Figure 50: BLDC Motor driver and experiment desk

Figure 50 demonstrates the experiment desk and the boards connection to the mechanical test bed.

4.2 Experiment methodology

4.2.1 Position measurement
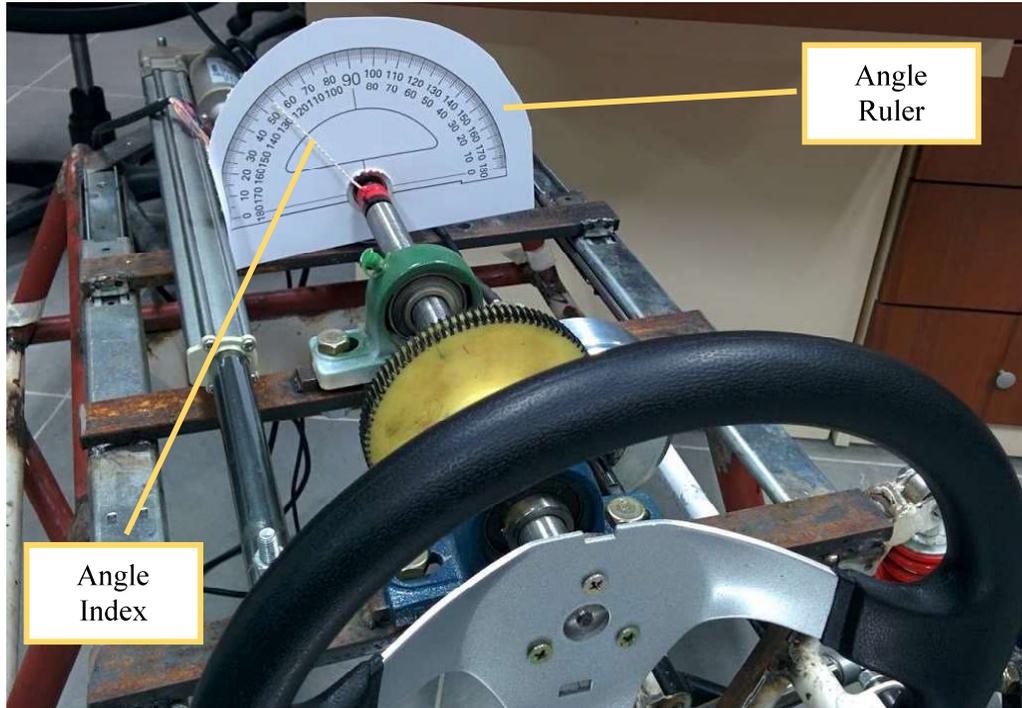
Figure 51: Position validation setup

Figure 51 shows the test setup for position measurement validating and accuracy and error determination. An angle ruler measures the steering wheel absolute position to verifying the generated voltage from the Arduino board. The output voltage observed and recorded every 5 degrees as well as analog output voltage for the steering wheel full range. The whole range for the steering wheel starts from 40 degrees up to 140 degrees at the right end. The position and voltage recording repeated ten times for full range to achieve accurate results.

4.2.2 Torque Measurement

The torque measurement experiment done by turning the steering wheel for the entire steering range with the five different force applied on the steering wheel. The steering wheel moved at a constant speed of **5.2 rad/second** to make sure that required torque for the range remains constant during the Handwheel moving. Each experiment repeated for ten times for each force to rise the most reliable results.
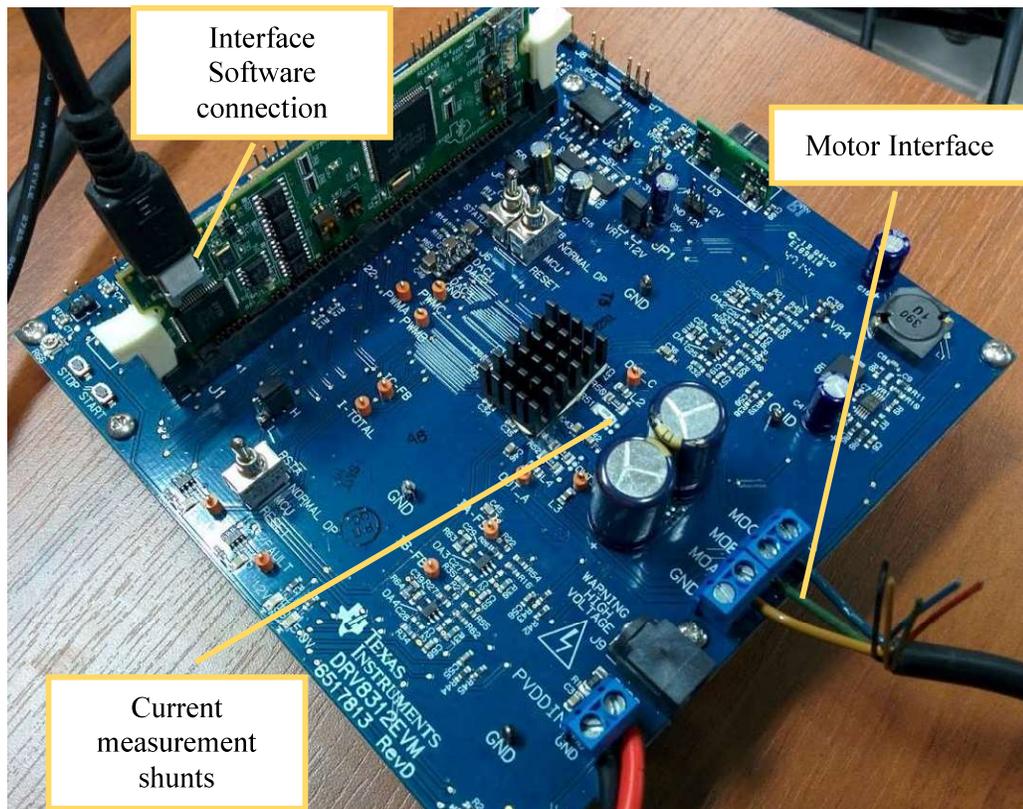
Figure 52: BLDC Driver setup

Figure 52 shows the BLDC Motor Driver configuration and the current measurement sensors to determine the motor output current to calculates the BLDC output torque. The Code Composer Studio (CCS) shows all parameter for the controller board and all parameters can be recorded on the graph or as a table for the further processing.

Figure 53: Motor spinning speed measurement setup

Figure 53 demonstrates the test setup to measure the BLDC motor speed by using precision laser tachometer to make sure that the measured torque is at the constant speed.

4.3 Result analysis

The Hand wheel's position and applying desired force on the Handwheel are two primary objectives in this research and implementation. As it discussed at the beginning of this chapter the position measurement had been done by moving in 5 degrees steps and measuring to output voltage to the position controller. Table 3 illustrates the measurements of the steering wheel position versus output voltages. The steering wheel initial position assumed at the middle. The full range for steering set by 50 degrees to each side. In Table 3 the first column shows the steering wheel position in 5 degrees steps. E1 to E10 stands for the first experiment round to 10th trial round respectively. Each round is started turning the steering wheel position from -50 degrees from the left up to the +50 degrees and record the voltage each 5 degrees. The given voltages in the table are in millivolt unit.

Table 3: Steering wheel position (Degrees) against Voltage conversion result [mV]

Degree	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
-50	872	872	872	873	873	874	873	873	873	873
-45	985	1010	985	1010	982	987	986	985	984	984
-40	1100	1080	1080	1110	1110	1090	1080	1080	1080	1080
-35	1200	1200	1200	1210	1200	1200	1200	1200	1200	1200
-30	1360	1330	1330	1330	1330	1330	1330	1330	1330	1330
-25	1440	1440	1440	1440	1440	1440	1440	1440	1440	1440
-20	1550	1550	1550	1550	1550	1550	1550	1550	1550	1550
-15	1690	1690	1690	1690	1690	1690	1690	1690	1690	1690
-10	1780	1780	1780	1780	1780	1780	1780	1780	1780	1780
-5	1900	1910	1910	1910	1910	1900	1900	1900	1910	1910
0	2020	2020	2030	2030	2030	2030	2030	2030	2030	2030
5	2130	2130	2140	2130	2130	2130	2130	2140	2130	2130
10	2250	2260	2260	2260	2260	2260	2260	2260	2260	2260
15	2380	2380	2380	2380	2380	2380	2380	2380	2380	2380
20	2480	2480	2480	2480	2480	2480	2480	2480	2480	2480
25	2590	2590	2590	2590	2590	2590	2590	2590	2590	2590
30	2690	2680	2680	2680	2680	2700	2680	2700	2680	2680
35	2820	2820	2820	2820	2820	2820	2820	2820	2820	2820
40	2940	2940	2940	2940	2940	2940	2940	2940	2940	2940
45	3060	3060	3060	3060	3060	3060	3060	3060	3060	3060
50	3170	3170	3170	3170	3170	3170	3170	3170	3170	3170

Figure 54 shows the samples graph for all experimentation repetitions from Table 3. Each series stand for an entire range experiments. According to Figure 54, the output voltage increases linearly by changes in Handwheel position.
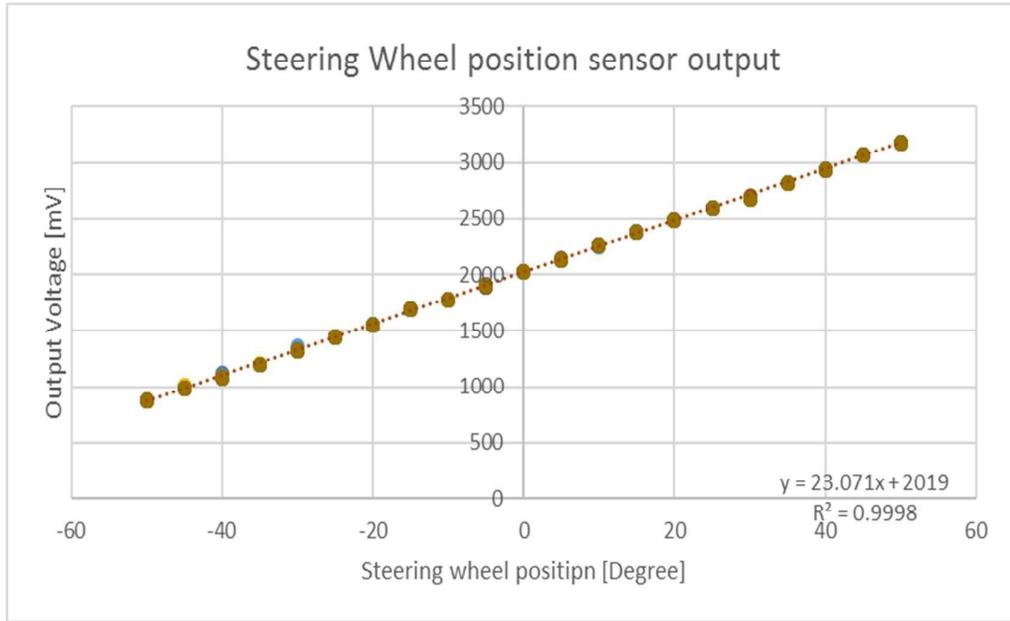
Figure 54: All experiments repetition comparison chart

Applied force on the steering wheel test, had been done by limiting the maximum motor current in the different level and measuring the desired torque to turning the steering wheel for the full range and constant speed.

Table 4: Output Steering Wheel Torque [Nm] against various current limits[mA]

Current Limit [mA]	E	E2	E3	E4	E5	E6	E7	E8	E9	E10
2000.00	2.08	1.98	2.00	1.98	1.99	1.95	1.96	1.96	1.96	1.99
2500.00	2.41	2.44	2.40	2.41	2.41	2.41	2.41	2.41	2.40	2.41
3000.00	2.90	2.88	2.89	2.89	2.79	2.69	2.87	2.87	2.89	2.87
3500.00	3.37	3.39	3.39	3.38	3.39	3.40	3.36	3.37	3.38	3.39
4000.00	3.62	3.65	3.59	3.59	3.66	3.62	3.66	3.63	3.69	3.64

Table 4 represents the output torque on the steering wheel for five different limiting currents. Experiments repeated ten times to analyses the driver operation. Figure 55 and Figure 56 are showing the visual spread chart and 3D diagram for the provided information in Table 4.
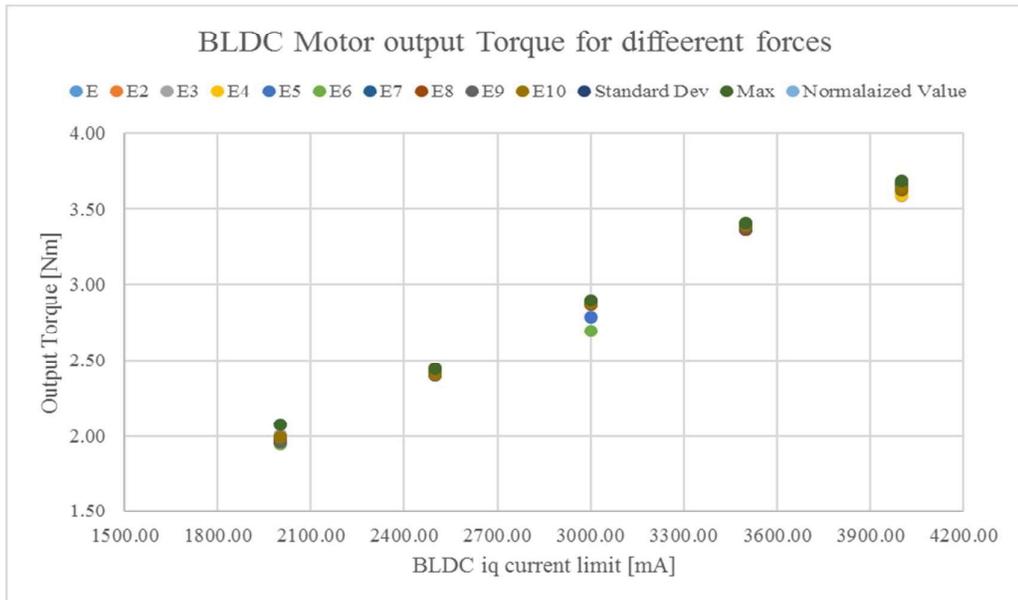
Figure 55: Steering wheel Torque vs. Different current level spread chart
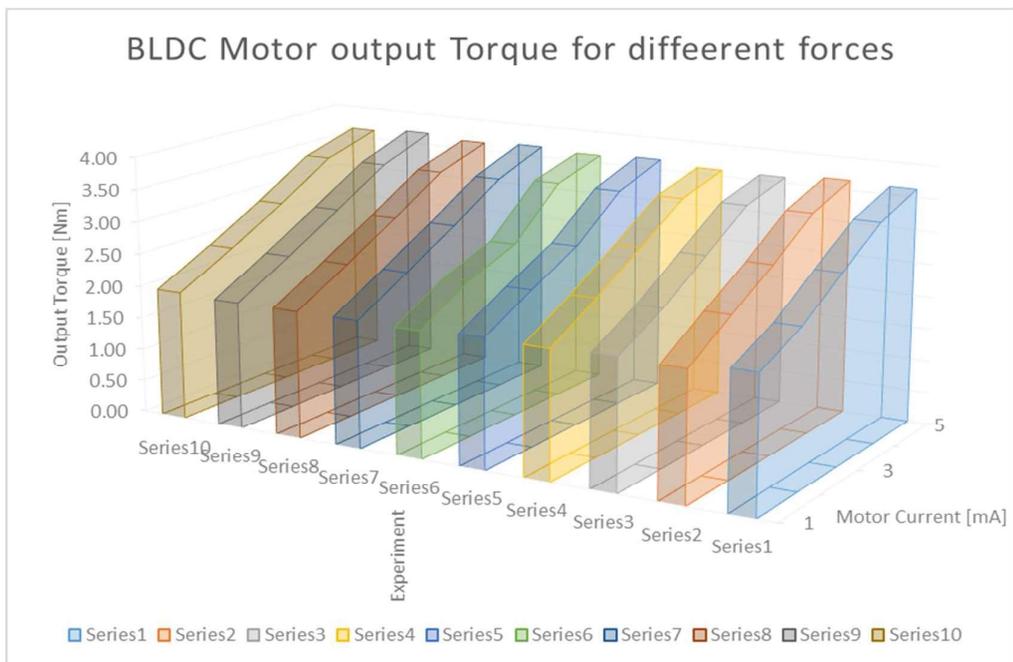
Figure 56: Steering wheel Torque(Nm) against Different current level (mA) diagram

4.3.1 Precision and Repeatability

The accuracy of a device is a degree of freedom from random output error for a particular input value. In another word, for a high precision device or a converter if a

large number of reading taken from the output for one particular input all should be very close together. The precision is not about the how output is accurate because the accuracy is how the reading value is close to the correct value but the precision is about the how the outputs are close together for and input in repeated experiments [14].

The publication of the ISO5725-1 standard describes the accuracy and precision. According to the ISO5725-1 specification, the Accuracy term is defined as the closeness of value to the real value and the general term Precision is the closeness of outputs in the sets of the results [15].

Statistically, the precision is defined as the standard deviation in statics calculation for a series of repetitive measurements [16]. The Standard Deviation **SD**, can be calculated with given formula as follow:

$$SD = \sqrt{\frac{\sum_{i=0}^{N}(X_i - \bar{X})^2}{N-1}} \quad where \quad \bar{X} = \frac{\sum_{i=0}^{N} X_i}{N} \qquad (11)$$

The given formula for standard deviation calculates precision for the each position sample groups from Table 3. To calculate the total system accuracy, the Coefficient of variation is the normalized parameter which is a point to the variability of the data points. The normalized value derives from dividing the standard deviation by the **mean** value in each test and multiply it with100. The average of all normalized standard deviation (Coefficient variation) gives the whole system weighted precision.

Table 5: Standard Deviation for each degree tests

Position Degree	Standard Deviation	Normalized Value
-50	0.6	0.000687443

64

-45	10.17644339	0.010281313
-40	12.20655562	0.011208958
-35	3	0.002497918
-30	9	0.006751688
-25	0	0
-20	0	0
-15	0	0
-10	0	0
-5	4.898979486	0.002570294
0	4	0.001972387
5	4	0.001876173
10	3	0.001328021
15	0	0
20	0	0
25	0	0
30	8.062257748	0.003002703
35	0	0
40	0	0
45	0	0
50	0	0

The weighted average for the entire system precision is equal to the sum of all normalized standard deviation divided by the number of experiments.

$$Precision = \frac{\sum Normalized\ SD}{Number\ of\ tests} = \frac{0.042176}{21} = 0.0020 \qquad (12)$$

A coefficient variable of %0.2 for precision shows that 68% of the outputs will be in the range of +/- 0.2% of the mean value and for the most of the time, the 95% of the outputs falls between +/- 0.4% of the average value.

Table 6: Standard deviation and normalized value for each experiment

Current Limit [mA]	Standard Dev	Max	Normalized Value
2000.00	0.03	2.08	0.02
2500.00	0.21	2.44	0.09
3000.00	0.23	2.90	0.08
3500.00	0.27	3.40	0.08
4000.00	0.13	3.69	0.03

Table 6 shows the standard deviation for the BLDC motor torque outputs for different experiments. The precision for the motor applied torque on the steering wheel can be calculated as follows:

$$Precision = \frac{\sum Normalized\ S}{Number\ of\ tests} = \frac{0.295757}{5} = 0.0591 \qquad (13)$$

A coefficient variable of %5.9 for precision shows that 68% of the torque outputs will be in the range of +/- 5.9% of the mean value and for the most of the time, the 95% of the outputs falls between +/- 11.8% of the average value.

4.3.2 Linearity

When the output from the system is linearly proportional to the input value, it is typically describing the system is linear. The standard procedure is to draw the best-fit line on the results graph. Figure 57 shows the results of the first experiment for entire steering range. According to the chart, the results are pretty linear, but some points are not on the appropriate line. Its also preferable to apply the least square line fitting formula. The maximum deviation of any of the output from the fitting line is defined as a non-linearity.

66
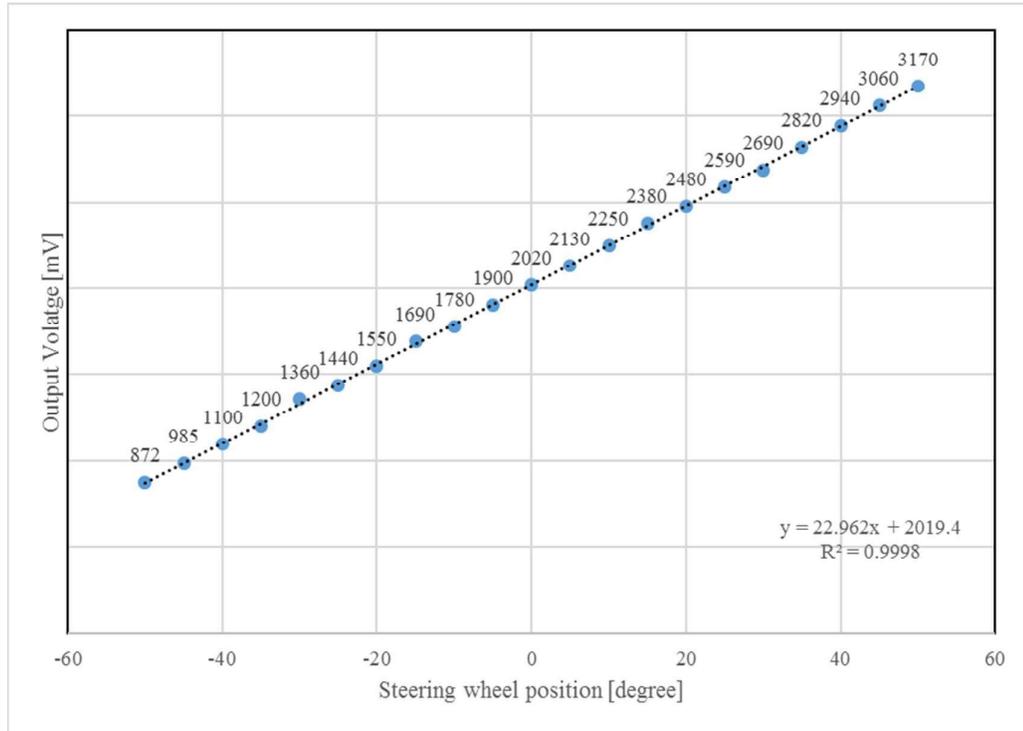
Figure 57: Position/Voltage output linearity graph

To find the best-fit line y = bx + a, over the given points by using the least square technique, the *a* and *b* constant should be optimized to minimize the error between the fitting line and the data points. Given below formulas calculates the value for a and b constants [14].

$$b = \frac{\sum(x_i y_i) - n x_m y_m}{\sum x_i{}^2 - n x_m^2} \qquad and \qquad a = y_m - b x_m \qquad (14)$$

In the first step, the mean values of x_m and y_m are necessary. For the second step the $(x_i y_i)$ and x_i^2 for all x and y should calculated.

For the given data chart in Figure 57, the $x_m = 0 \; and \; y_m = 2019.381$ and $\sum(x_i y_i) = 442025$ and the $\sum x_i{}^2 = 19250$. With these value a and b calculates as follow:

$$b = \frac{442025 - (21 \times 0 \times 2019.381)}{19250 - (21 \times 0^2)} = 22.96 \quad and \quad a$$

$$= 2019.381 - (b \times 0)$$

<div align="right">(15)</div>

By using the calculated values for a and b the best fit line can be written as follows:

$$y = 22.96\,x + 2019.381 \tag{16}$$
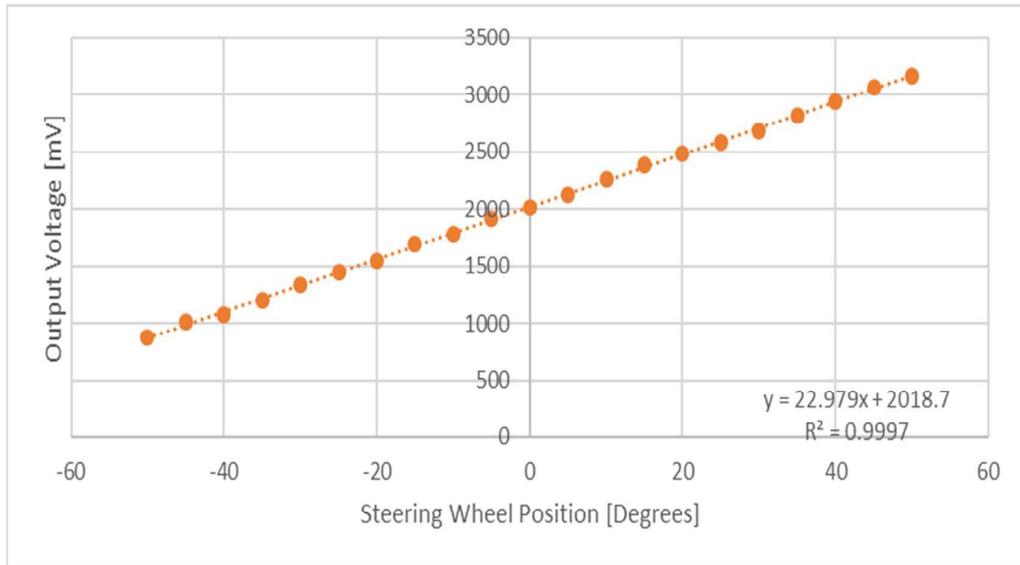
Figure 58: Best-fitting line for 2nd Experiment
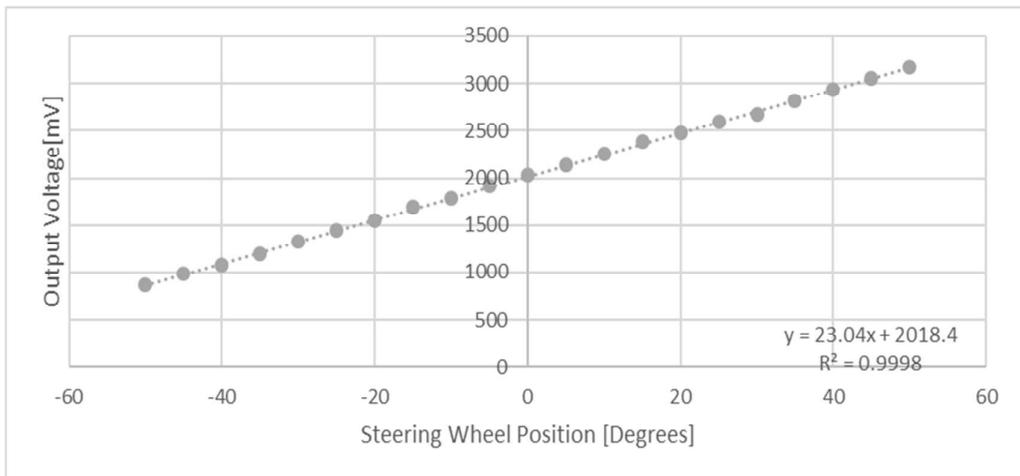
Figure 59: Best-fitting line for 3rd Experiment
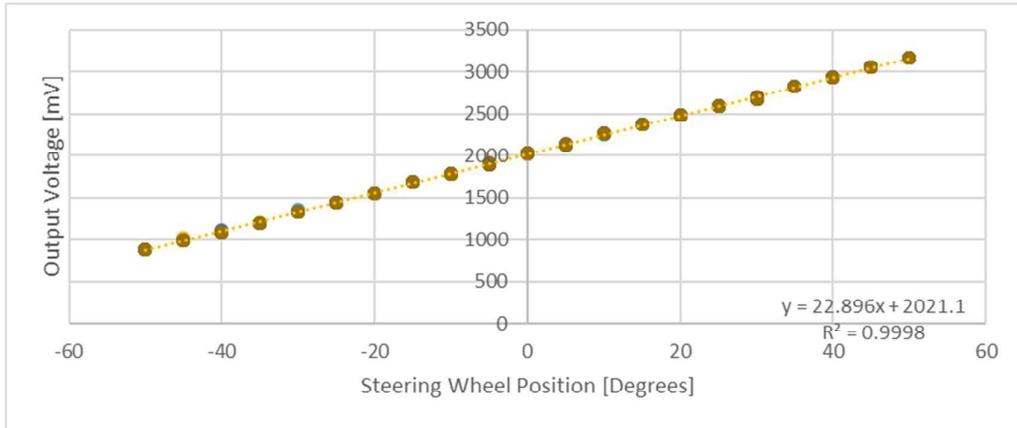
Figure 60: Best-fitting line for 4th Experiment
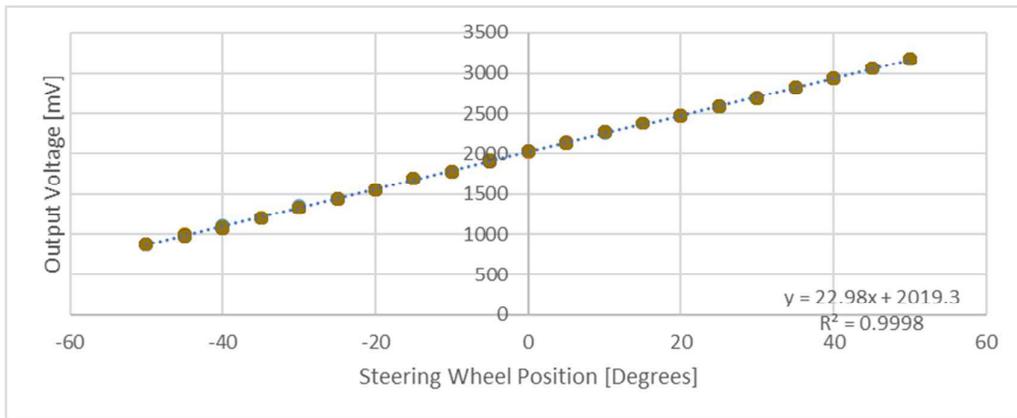
Figure 61: Best-Fitting Line for 5th Experiment
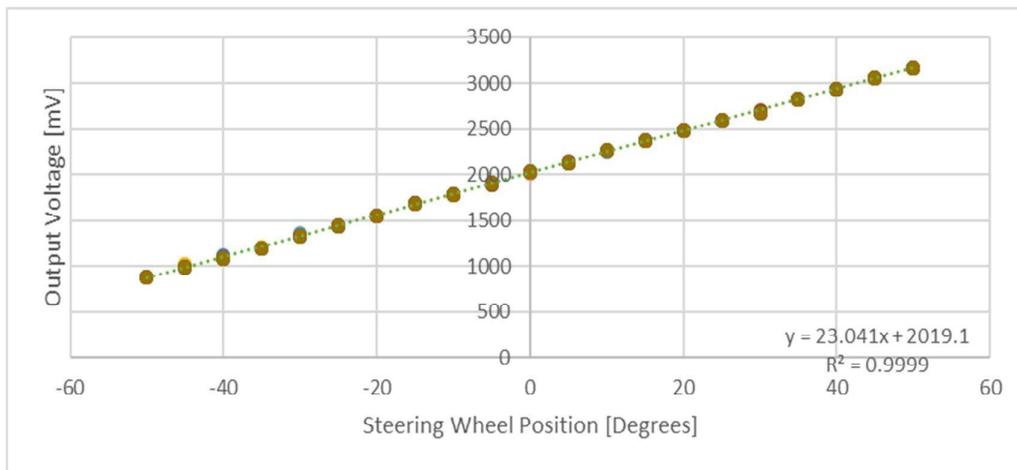
Figure 62: Best-fitting line for 6th Experiment
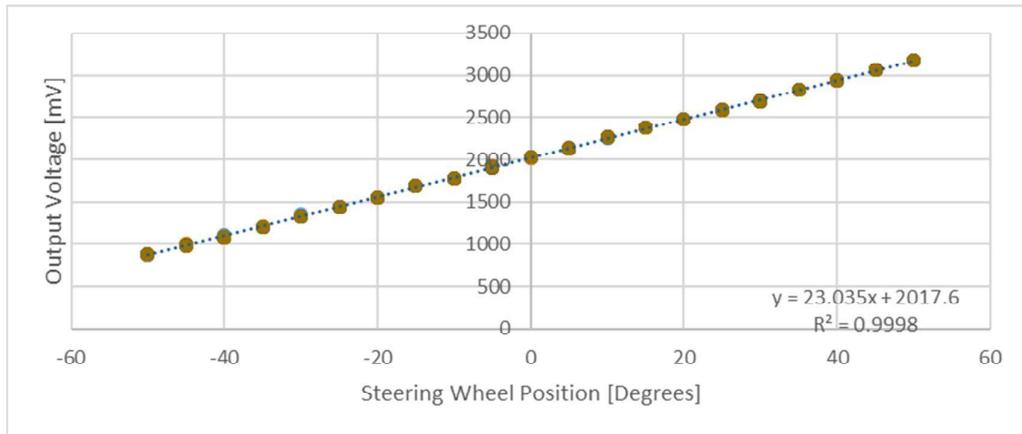
Figure 63: Best-Fitting Line for 7th Experiment
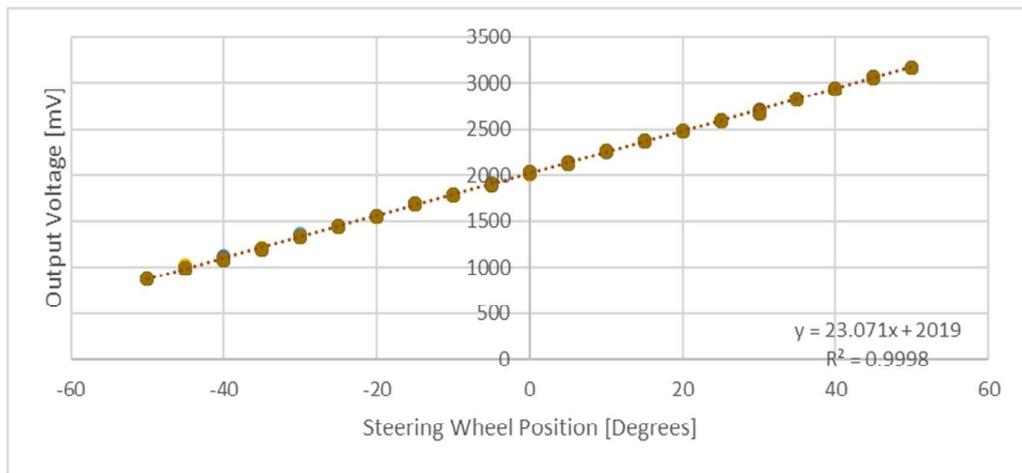
Figure 64: Best-fitting line for 8th Experiment
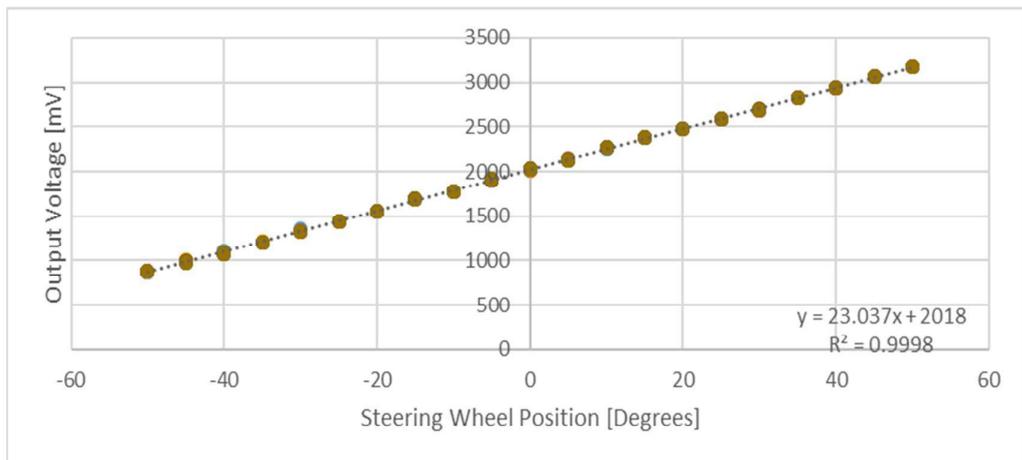
Figure 65: Best-Fitting Line for 9th Experiment
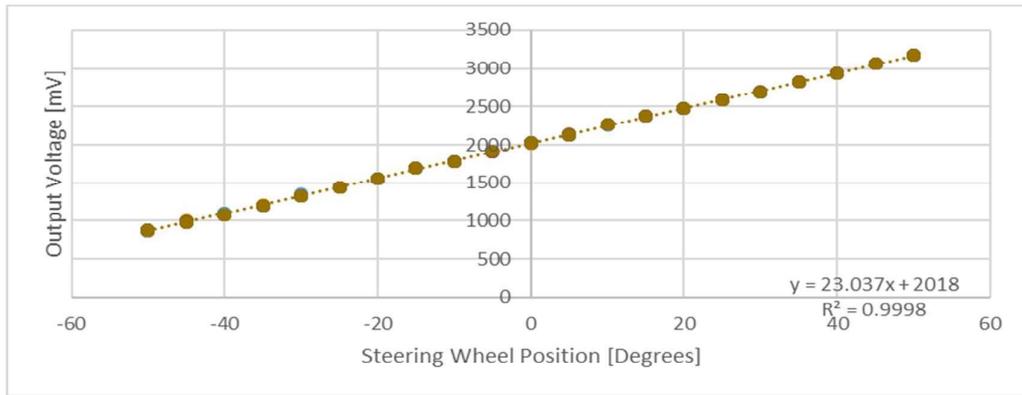
Figure 66: Best-Fitting Line for 10th Experiment

According to Figure 58 to Figure 66, and the calculated line equation for each line and maximum deviation error provided in the figures, the linearity for all experiments is better than 99.98% for all tests in a position to voltage converter unit.

For torque measurement tests the line fitting equation applied same as the position line calculation.
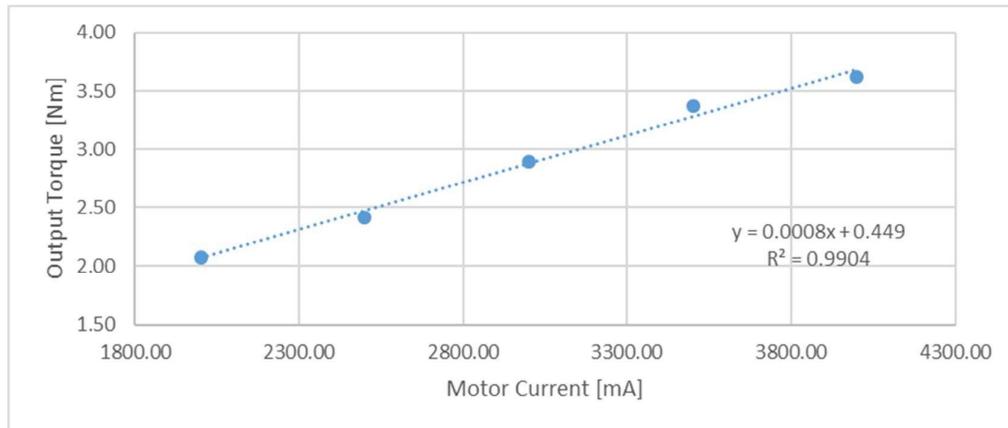
Figure 67: Best-fitting line for 1st torque Experiment
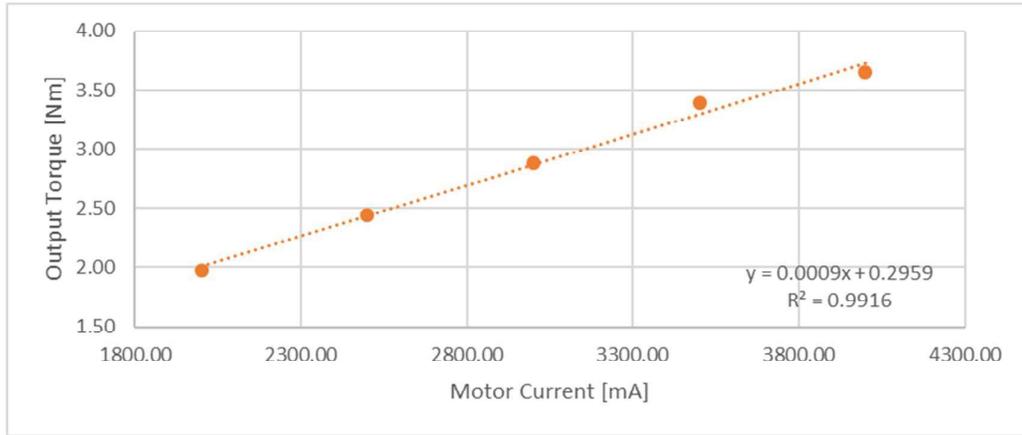
Figure 68: Best-fitting line for 2nd torque Experiment
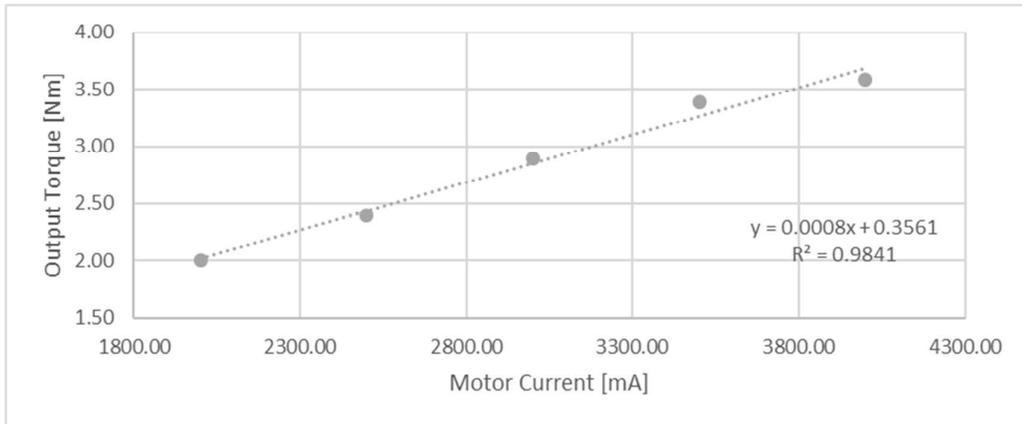
Figure 69: Best-fitting line for 3rd torque Experiment
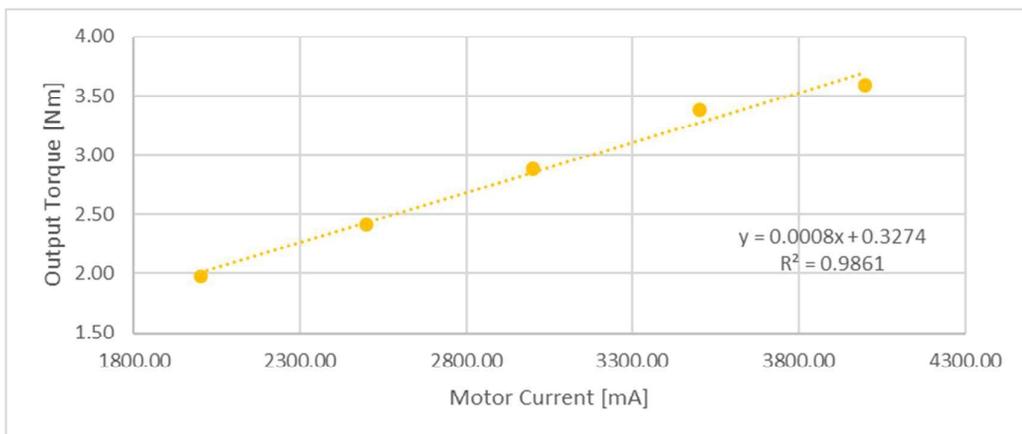
Figure 70: Best-fitting line for 4th torque Experiment
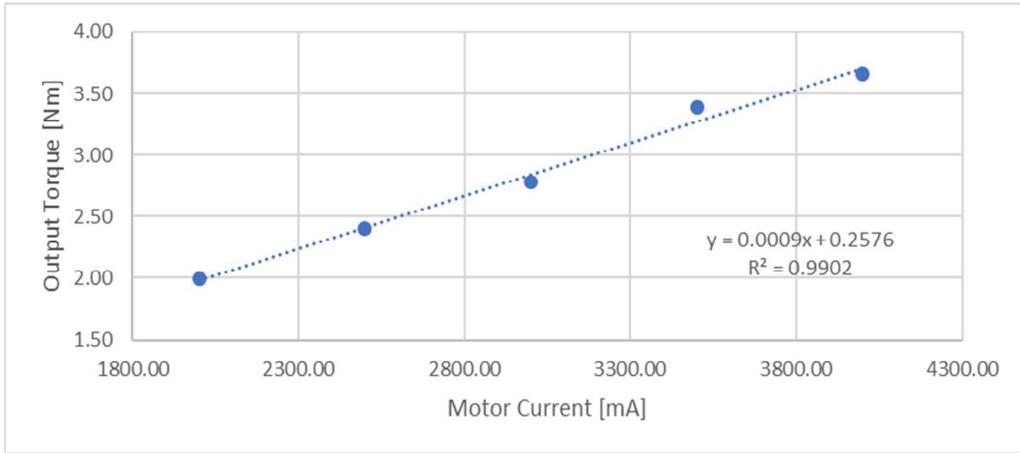
Figure 71: Best-fitting line for 5th torque Experiment
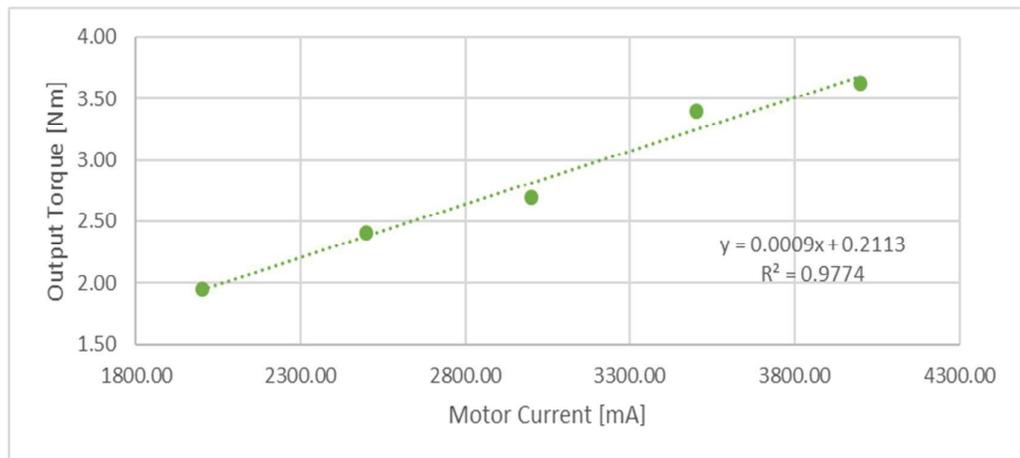
Figure 72: Best-fitting line for 6th torque Experiment
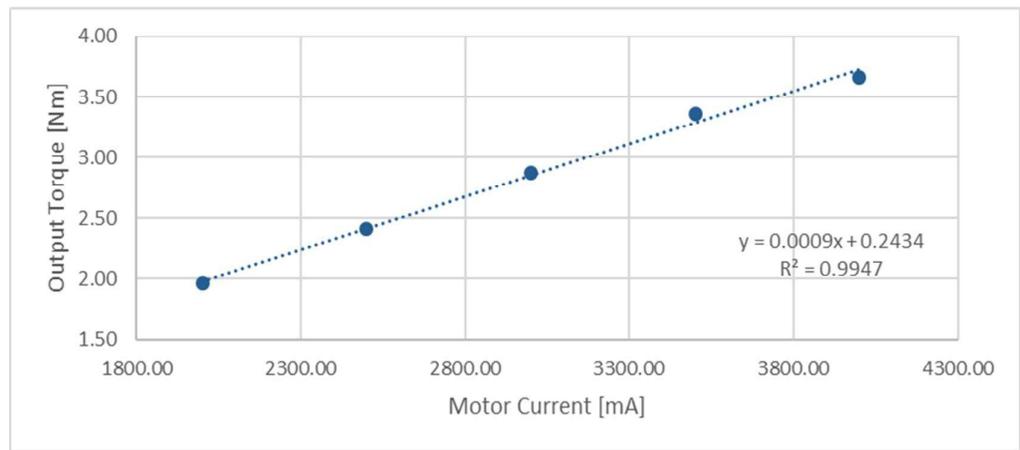
Figure 73: Best-fitting line for 7th torque Experiment
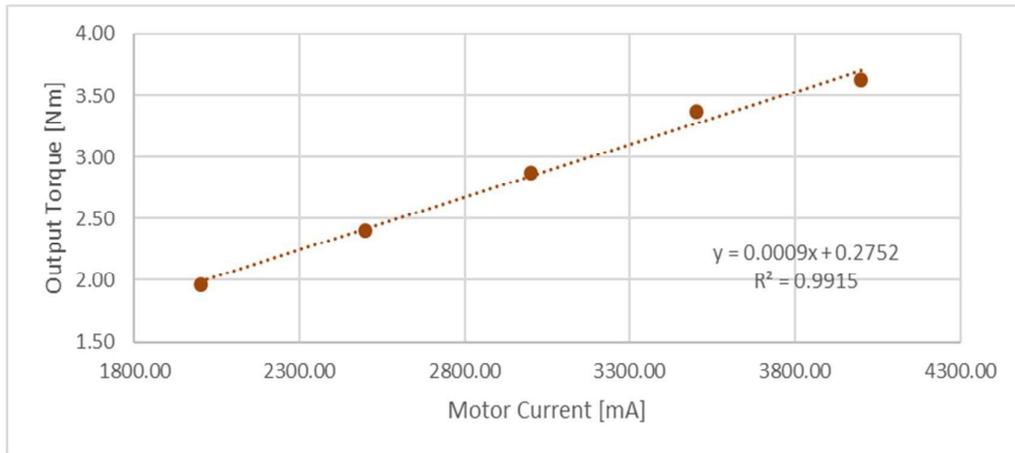
Figure 74: Best-fitting line for 8th torque Experiment
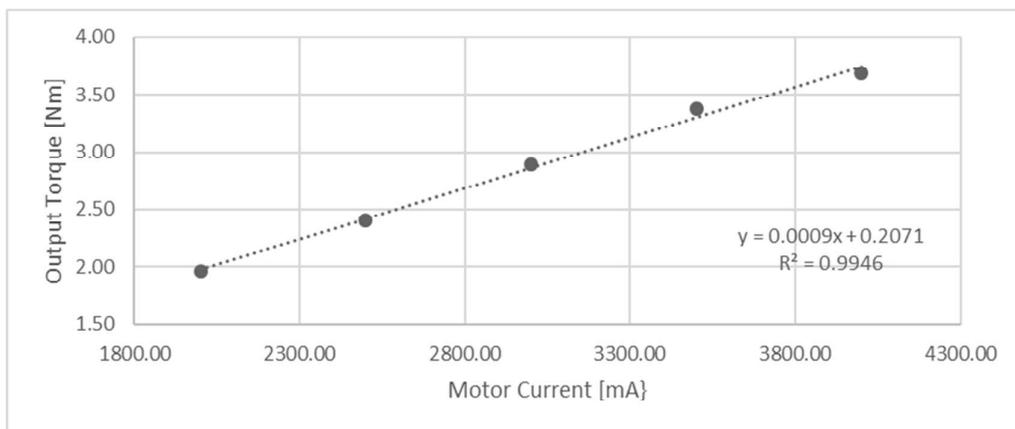
Figure 75: Best-fitting line for 9th torque Experiment
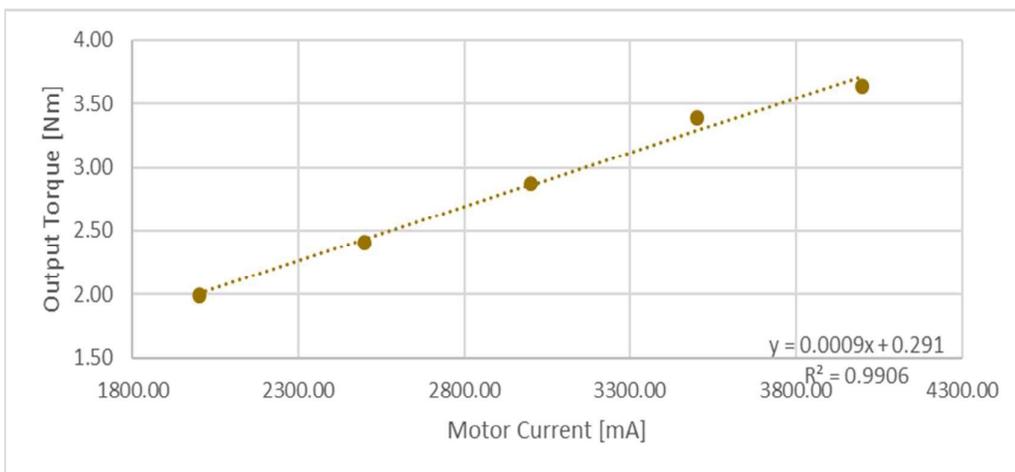
Figure 76: Best-fitting line for 10th torque Experiment

According to Figure 67 to Figure 76 the maximum non-linearity is in the 6th experiment with 2.26% deviation from the best fit line. The average non-linearity in 10 experiment is less than 1.1% and the torque output's linearity is better than 98.91% in average for all experiments. According to the provided graphs for linearity analysis, both position converter and torque controller have a good linearity response.

4.3.3 Range

The range is the maximum and minimum value of the designed system output. For the position converter, the angular input range is between -50 degrees to +50 degree from the steering wheel initial position (middle position). The voltage output minimum is 873 mV, and the maximum output voltage is 3170 mV for the specified range.

For the torque controller, the minimum input reference limiting current is 1000 mA, and the maximum limiting current is 5000mA. Moreover, the output limiting torque is between 1Nm up to 5Nm.

4.3.4 Error

There is some source of errors that disturbs the measurements and also cause the non-linearity and deviation in the results. Measuring instruments are playing the important role in data validation in this part. A digital oscilloscope is used to measuring the output voltage signal from the position converter module. The resolution for the oscilloscope is 1mV when the input is less than two volts and the resolution changes to 10mV when the input is greater than 2 volts. An angle ruler is used to observe the steering angle, the user reading angle and also the resolution of the angle ruler are the other source of the disturbance in the measurements. For the

torque controller, the current measurement circuit error can cause the deviation from the correct value.

Chapter 5

CONCLUSION

Drive by wire technology is being increasingly used by many automotive manufacturers for vehicle products. The Drive-by-wire (DbW), replaces the conventional mechanical linkages and mechanical actuators with electronic control systems and electromechanical actuators in the vehicle.

This is achieved by using angular absolute position sensor to detect the position of the steering wheel. A Brushless DC Motor employed to emulate the steering feel force feedback from the tire on the steering wheel. Pneumatic actuators control the road wheels to control the steering angle. Therefore, the main objectives are as follows:

-   Steer by Wire (SbW): Removing any mechanical linkage between the steering wheel and the road wheels.

-   Steering wheel position determination and translate to the road wheel controller compatible input.

-   Creating the desired force feedback onto steering wheel

This study has shown that force feedback on the steering wheel by using the brushless DC motor has very precise and reasonable response to the steering feel emulation. Results are shown that, the precision of the torque controller in the full range movement of steering wheel spinning from has a coefficient variable of %5.9

as well as less 98.91% linearity or in another word less than 1.1% linearity error. The coefficient variable for precision shows that 68% of the torque outputs will be in the range of +/- 5.9% of the mean value and for the most of the time, the 95% of the outputs falls between +/- 11.8% of the average value.

The steering wheel's position sensor voltage converter has a less than 0.02% non-linearity in average for all experiments. The coefficient variable for the implemented position voltage converter is 0.2%. A coefficient variable of 0.2% for precision shows that 68% of the outputs will be in the range of +/- 0.2% of the mean value and for the most of the time, the 95% of the outputs falls between +/- 0.4% of the average value.

In conclusion, the achieved results for all objectives have good precision and repeatability response as well as linearity for the entire range of working conditions. The provided experiment results in proof that the steering wheel force emulation and the position feedback from the steering wheel for road wheel tires are good enough to be replaced with the mechanical steering wheel.

The Motor driver uses the sensorless BLDC drive method for spinning the motor by getting the information from the FAST algorithm for torque control. The FAST algorithm estimates the motor Flux, Angle, Speed and Torque by using the pre-defined motor parameters and measuring the phase voltages and currents and the total input voltage. The provided information by FAST algorithm is more precise and smooth in the higher speed of the motor because of the higher frequency changes in the motor flux. An incremental rotary encoder improves the system zero speed

performance and precision. To FAST algorithm can be replaced by an incremental

encoder provided a position to make torque control more linear and more smooth.

REFERENCES

[1] Asai, Shoji; Kuroyanagi, Hiroshi; Takeuchi, Shinji; Takahashi, Toshihiro; Ogawa, Shoji;,(2004) Development of a Steer-by-Wire System with Force Feedback Using a Disturbance Observer, *SAE International Journal,* DOI:10.4271/2004-01-1100.

[2] Ba-Hai Nguyen & Jee-Hwan Ryu, (2009), Direct current measurement based steer-by-wire systems for realistic driving feeling, *IEEE International Symposium on Industrial Electronics*, Seoul, pp. 1023-1028. DOI: 10.1109/ISIE.2009.5221999

[3] R. Pastorino, M. A. Naya, J. A. Pérez & J. Cuadrado,(2011) Geared PM coreless motor modelling for driver's force feedback in steer-by-wire systems, *Mechatronics,* vol. 21, no. 6, p. 1043–1054.

[4] E. Mehdizadeh, M. Kabganian & R. Kazemi,(2011), A new force feedback for Steer-by-Wire vehicles via virtual vehicle concept, *50th IEEE Conference on Decision & Control and European Control Conference*, Orlando, FL, pp. 2281-2286. DOI: 10.1109/CDC.2011.6160941

[5] Tian Jie, Wu Jin, Chen Ning & Luo Shi, (2011), Research on road feeling control of Steer-by-wire System, *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference* on, Hohhot, pp.

4349-4353.doi: 10.1109/MACE.2011.5987967

[6] A. Bertacchini, L. Tamagnini & P. Pavan,(2006), Force Feedback in Steer-by-Wire Systems: Architecture and Experimental Results, *IEEE International Symposium on Industrial Electronics, Montreal, Que.*, 2006, pp. 3050-3055. doi: 10.1109/ISIE.2006.296102

[7] P. Yih & J. C. Gerdes, (2005), Modification of vehicle handling characteristics via steer-by-wire, *IEEE Transactions on Control Systems Technology,* vol. 13, no. 6, pp. 965 - 976.

[8] J. P. Switkes, E. J. Rossetter, I. A. Coe & J. C. Gerdes,(2005), Handwheel Force Feedback for Lanekeeping Assistance: Combined Dynamics and Stability, *J. Dyn. Sys., Meas., Control* , vol. 128, no. 3, pp. 532-542.

[9] A. Balachandran & J. C. Gerdes,(2015) Designing Steering Feel for Steer-by-Wire Vehicles Using Objective Measures, *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, vol. 20, pp. 373-383.

[10] M. A. Johnson & M. H.Moradi, (2012), PID Control: New Identification and Design Methods, London: *Springer.*

[11] B. Akin & M. Bhardwaj, (2010), Sensorless Field Oriented Control of Multiple Permanent Magnet Motors, *Texas Instruments* ,Dallas, TX.

[12] M. Rosth, (2007), Hydraulic Power Steering System Design in Road Vehicles, Linkoping University, Linkoping.

[13] G. Forkenbrock & D. Elsasser, (2005), An Assessment of Human Driver Steering Capability, *National Highway Traffic Safety Administration*, East Liberty, OH,.

[14] A. S. Morris, (2001), Measurement and Instrumentation Principles, *Butterworth-Heinemann.*

[15] Accuracy (trueness and precision) of measurement methods and results -- Part 1: General principles and definitions, *ISO 5725-1,* 1994.

[16] S. K. G. Geoffrey Vining,(2006), Statistical Methods for Engineers, *Brooks/Cole,.*

[17] C.-C. Yu, (2006), Autotuning of PID Controllers 2nd edition, London: *Springer-Verlag.*

[18] Power steering system (2006), *http://www.automechanicinfairoaks.com/power-steering/*

[19] Davies Alex, (2010), Infiniti's New Steering System Is a Big Step Forward—Unless You Love Cars, *https://www.wired.com/2014/06/infiniti-q50-steer-by-*

82

wire/

APPENDICES

Appendix A: Arduino code for analog voltage generation

```
#define ANALOG_OUT   3
#define REFERENCE_POS 517 //Try to set the Reference
position at the middle
#define ACTIVE_RANGE   300
//Active range angle [45degree assumed for each side]   *
Pulse per each angle [2.844444444 for current sensor]

#define SENSOR_MAX_VALUE   1023
int get_EncoderPosition ();
unsigned int grayToBinary(unsigned int num);
int normalizPositionInRange (int posValue);
/*
 * Init
 */
void setup()
{
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode (52,INPUT);
  pinMode (50,INPUT);
  pinMode (48,INPUT);
  pinMode (46,INPUT);
  pinMode (44,INPUT);
  pinMode (42,INPUT);
  pinMode (40,INPUT);
  pinMode (38,INPUT);
  pinMode (36,INPUT);
  pinMode (34,INPUT);
  pinMode (ANALOG_OUT,OUTPUT);
}
/*
 * Main Loop
 */
void loop()
{
  unsigned int positionValue = 0;
  unsigned int normalPosition = 0;
  positionValue = grayToBinary(get_EncoderPosition());
  normalPosition =
normalizPositionInRange(positionValue);
  //Serial.println(positionValue);
  normalPosition =
map(normalPosition,0,(2*ACTIVE_RANGE),0,255);
  analogWrite (ANALOG_OUT,normalPosition);
  //delay(200);
}
/*
```

```c
 * This function converts a reflected binary
 * Gray code number to a binary number.
 * Each Gray code bit is exclusive-ored with all
 * more significant bits.
 */
unsigned int grayToBinary(unsigned int num)
{
    unsigned int mask;
    for (mask = num >> 1; mask != 0; mask = mask >> 1)
    {
        num = num ^ mask;
    }
    return num;
}


/*
 * Get Information from Absolute position sensor
 * return Value is in Normal Gray code
 */
int get_EncoderPosition ()
{
  int posVAlue = 0;
  if (digitalRead(52))
    posVAlue |= (1 << 0);
  if (digitalRead(50))
    posVAlue |= (1 << 1);
  if (digitalRead(48))
    posVAlue |= (1 << 2);
  if (digitalRead(46))
    posVAlue |= (1 << 3);
  if (digitalRead(44))
    posVAlue |= (1 << 4);
  if (digitalRead(42))
    posVAlue |= (1 << 5);
  if (digitalRead(40))
    posVAlue |= (1 << 6);
  if (digitalRead(38))
    posVAlue |= (1 << 7);
  if (digitalRead(36))
    posVAlue |= (1 << 8);
  if (digitalRead(34))
    posVAlue |= (1 << 9);
return posVAlue;
}

/*
 * Position Normalizing Function
 * Input is current Pos Value
 * Output is between 0 and 2*ACTIVE_RANGE
 */
```

```
int normalizPositionInRange (int posValue)
{
 if (posValue < (REFERENCE_POS - ACTIVE_RANGE))
    return 0;
 else if (posValue > (REFERENCE_POS + ACTIVE_RANGE))
    return (ACTIVE_RANGE * 2);
 else
    return (posValue - (REFERENCE_POS - ACTIVE_RANGE));
}
```

Appendix B: TMS320F28069M Torque controller

```c
// system includes
#include <math.h>
#include "main.h"

#ifdef FLASH
#pragma CODE_SECTION(mainISR,"ramfuncs");
#endif

// Include header files used in the main function
//
// *****************************************************************************
// the defines
#define LED_BLINK_FREQ_Hz   10
//
// *****************************************************************************
// the globals

uint_least16_t gCounter_updateGlobals = 0;

bool Flag_Latch_softwareUpdate = true;

CTRL_Handle ctrlHandle;

HAL_Handle halHandle;

USER_Params gUserParams;

HAL_PwmData_t gPwmData = {_IQ(0.0), _IQ(0.0), _IQ(0.0)};

HAL_AdcData_t gAdcData;

_iq gMaxCurrentSlope = _IQ(0.0);

#ifdef FAST_ROM_V1p6
CTRL_Obj *controller_obj;
#else
CTRL_Obj ctrl;                          //v1p7 format
#endif

ST_Obj st_obj;
ST_Handle stHandle;

uint16_t gLEDcnt = 0;

volatile MOTOR_Vars_t gMotorVars = MOTOR_Vars_INIT;

volatile _iq torqueLimit = _IQ(0.2);

#ifdef FLASH
// Used for running BackGround in flash, and ISR in RAM
extern uint16_t *RamfuncsLoadStart, *RamfuncsLoadEnd, *RamfuncsRunStart;
#endif
```

```c
#ifdef DRV8301_SPI
// Watch window interface to the 8301 SPI
DRV_SPI_8301_Vars_t gDrvSpi8301Vars;
#endif
#ifdef DRV8305_SPI
// Watch window interface to the 8305 SPI
DRV_SPI_8305_Vars_t gDrvSpi8305Vars;
#endif

_iq gFlux_pu_to_Wb_sf;

_iq gFlux_pu_to_VpHz_sf;

_iq gTorque_Ls_Id_Iq_pu_to_Nm_sf;

_iq gTorque_Flux_Iq_pu_to_Nm_sf;

//
// *****************************************************************************
// the functions

void main(void)
{
  uint_least8_t estNumber = 0;

#ifdef FAST_ROM_V1p6
  uint_least8_t ctrlNumber = 0;
#endif

  // Only used if running from FLASH
  // Note that the variable FLASH is defined by the project
  #ifdef FLASH
  // Copy time critical code and Flash setup code to RAM
  // The RamfuncsLoadStart, RamfuncsLoadEnd, and RamfuncsRunStart
  // symbols are created by the linker. Refer to the linker files.
  memCopy((uint16_t *)&RamfuncsLoadStart,(uint16_t
*)&RamfuncsLoadEnd,(uint16_t *)&RamfuncsRunStart);
  #endif

  // initialize the hardware abstraction layer
  halHandle = HAL_init(&hal,sizeof(hal));


  // check for errors in user parameters
  USER_checkForErrors(&gUserParams);


  // store user parameter error in global variable
  gMotorVars.UserErrorCode = USER_getErrorCode(&gUserParams);


  // do not allow code execution if there is a user parameter error
  if(gMotorVars.UserErrorCode != USER_ErrorCode_NoError)
    {
      for(;;)
        {
          gMotorVars.Flag_enableSys = false;
        }
    }
```

89

```
  // initialize the user parameters
  USER_setParams(&gUserParams);


  // set the hardware abstraction layer parameters
  HAL_setParams(halHandle,&gUserParams);


  // initialize the controller
#ifdef FAST_ROM_V1p6
  ctrlHandle = CTRL_initCtrl(ctrlNumber, estNumber);              //v1p6
format (06xF and 06xM devices)
  controller_obj = (CTRL_Obj *)ctrlHandle;
#else
  ctrlHandle = CTRL_initCtrl(estNumber,&ctrl,sizeof(ctrl));       //v1p7
format default
#endif


  {
    CTRL_Version version;

    // get the version number
    CTRL_getVersion(ctrlHandle,&version);

    gMotorVars.CtrlVersion = version;
  }


  // set the default controller parameters
  CTRL_setParams(ctrlHandle,&gUserParams);


  // setup faults
  HAL_setupFaults(halHandle);


  // initialize the interrupt vector table
  HAL_initIntVectorTable(halHandle);


  // enable the ADC interrupts
  HAL_enableAdcInts(halHandle);


  // enable global interrupts
  HAL_enableGlobalInts(halHandle);


  // enable debug interrupts
  HAL_enableDebugInt(halHandle);


  // disable the PWM
  HAL_disablePwm(halHandle);
```

```
    // initialize the SpinTAC Components
    stHandle = ST_init(&st_obj, sizeof(st_obj));


    // setup the SpinTAC Components
    ST_setupVelCtl(stHandle);


#ifdef DRV8301_SPI
    // turn on the DRV8301 if present
    HAL_enableDrv(halHandle);
    // initialize the DRV8301 interface
    HAL_setupDrvSpi(halHandle,&gDrvSpi8301Vars);
#endif

#ifdef DRV8305_SPI
    // turn on the DRV8305 if present
    HAL_enableDrv(halHandle);
    // initialize the DRV8305 interface
    HAL_setupDrvSpi(halHandle,&gDrvSpi8305Vars);
#endif

    // enable DC bus compensation
    CTRL_setFlag_enableDcBusComp(ctrlHandle, true);


    // compute scaling factors for flux and torque calculations
    gFlux_pu_to_Wb_sf = USER_computeFlux_pu_to_Wb_sf();
    gFlux_pu_to_VpHz_sf = USER_computeFlux_pu_to_VpHz_sf();
    gTorque_Ls_Id_Iq_pu_to_Nm_sf =
USER_computeTorque_Ls_Id_Iq_pu_to_Nm_sf();
    gTorque_Flux_Iq_pu_to_Nm_sf = USER_computeTorque_Flux_Iq_pu_to_Nm_sf();


    for(;;)
    {
      // Waiting for enable system flag to be set
      while(!(gMotorVars.Flag_enableSys));

      // Dis-able the Library internal PI.  Iq has no reference now
      CTRL_setFlag_enableSpeedCtrl(ctrlHandle, false);

      // loop while the enable system flag is true
      while(gMotorVars.Flag_enableSys)
        {
          CTRL_Obj *obj = (CTRL_Obj *)ctrlHandle;
          ST_Obj *stObj = (ST_Obj *)stHandle;

          // increment counters
          gCounter_updateGlobals++;

          // enable/disable the use of motor parameters being loaded from
user.h

CTRL_setFlag_enableUserMotorParams(ctrlHandle,gMotorVars.Flag_enableUserPa
rams);

          // enable/disable Rs recalibration during motor startup
```

```
        EST_setFlag_enableRsRecalc(obj-
>estHandle,gMotorVars.Flag_enableRsRecalc);

        // enable/disable automatic calculation of bias values

CTRL_setFlag_enableOffset(ctrlHandle,gMotorVars.Flag_enableOffsetcalc);


        if(CTRL_isError(ctrlHandle))
          {
            // set the enable controller flag to false
            CTRL_setFlag_enableCtrl(ctrlHandle,false);

            // set the enable system flag to false
            gMotorVars.Flag_enableSys = false;

            // disable the PWM
            HAL_disablePwm(halHandle);
          }
        else
          {
            // update the controller state
            bool flag_ctrlStateChanged = CTRL_updateState(ctrlHandle);

            // enable or disable the control
            CTRL_setFlag_enableCtrl(ctrlHandle,
gMotorVars.Flag_Run_Identify);

            if(flag_ctrlStateChanged)
              {
                CTRL_State_e ctrlState = CTRL_getState(ctrlHandle);

                if(ctrlState == CTRL_State_OffLine)
                  {
                    // enable the PWM
                    HAL_enablePwm(halHandle);
                  }
                else if(ctrlState == CTRL_State_OnLine)
                  {
                    if(gMotorVars.Flag_enableOffsetcalc == true)
                    {
                      // update the ADC bias values
                      HAL_updateAdcBias(halHandle);
                    }
                    else
                    {
                      // set the current bias

HAL_setBias(halHandle,HAL_SensorType_Current,0,_IQ(I_A_offset));

HAL_setBias(halHandle,HAL_SensorType_Current,1,_IQ(I_B_offset));

HAL_setBias(halHandle,HAL_SensorType_Current,2,_IQ(I_C_offset));

                      // set the voltage bias

HAL_setBias(halHandle,HAL_SensorType_Voltage,0,_IQ(V_A_offset));

HAL_setBias(halHandle,HAL_SensorType_Voltage,1,_IQ(V_B_offset));
```

```
HAL_setBias(halHandle,HAL_SensorType_Voltage,2,_IQ(V_C_offset));
                        }

                        // Return the bias value for currents
                        gMotorVars.I_bias.value[0] =
HAL_getBias(halHandle,HAL_SensorType_Current,0);
                        gMotorVars.I_bias.value[1] =
HAL_getBias(halHandle,HAL_SensorType_Current,1);
                        gMotorVars.I_bias.value[2] =
HAL_getBias(halHandle,HAL_SensorType_Current,2);

                        // Return the bias value for voltages
                        gMotorVars.V_bias.value[0] =
HAL_getBias(halHandle,HAL_SensorType_Voltage,0);
                        gMotorVars.V_bias.value[1] =
HAL_getBias(halHandle,HAL_SensorType_Voltage,1);
                        gMotorVars.V_bias.value[2] =
HAL_getBias(halHandle,HAL_SensorType_Voltage,2);

                        // enable the PWM
                        HAL_enablePwm(halHandle);
                      }
                  else if(ctrlState == CTRL_State_Idle)
                    {
                        // disable the PWM
                        HAL_disablePwm(halHandle);
                        gMotorVars.Flag_Run_Identify = false;
                    }

                    if((CTRL_getFlag_enableUserMotorParams(ctrlHandle) ==
true) &&
                        (ctrlState > CTRL_State_Idle) &&
                        (gMotorVars.CtrlVersion.minor == 6))
                        {
                            // call this function to fix 1p6
                            USER_softwareUpdate1p6(ctrlHandle);
                        }

                }
            }

        if(EST_isMotorIdentified(obj->estHandle))
          {
              // set the current ramp
              EST_setMaxCurrentSlope_pu(obj->estHandle,gMaxCurrentSlope);
              gMotorVars.Flag_MotorIdentified = true;

              // set the speed reference
              CTRL_setSpd_ref_krpm(ctrlHandle,gMotorVars.SpeedRef_krpm);

              // set the speed acceleration

CTRL_setMaxAccel_pu(ctrlHandle,_IQmpy(MAX_ACCEL_KRPMPS_SF,gMotorVars.MaxAccel_krpmps));

              // enable the SpinTAC Speed Controller
              STVELCTL_setEnable(stObj->velCtlHandle, true);
```
93

```c
            if(EST_getState(obj->estHandle) != EST_State_OnLine)
            {
             // if the estimator is not running, place SpinTAC into reset
             STVELCTL_setEnable(stObj->velCtlHandle, false);
            }

            if(Flag_Latch_softwareUpdate)
            {
              Flag_Latch_softwareUpdate = false;

              USER_calcPIgains(ctrlHandle);

              // initialize the watch window kp and ki current values with
pre-calculated values
              gMotorVars.Kp_Idq = CTRL_getKp(ctrlHandle,CTRL_Type_PID_Id);
              gMotorVars.Ki_Idq = CTRL_getKi(ctrlHandle,CTRL_Type_PID_Id);

                     // initialize the watch window Bw value with the
default value
              gMotorVars.SpinTAC.VelCtlBw_radps =
STVELCTL_getBandwidth_radps(stObj->velCtlHandle);

              // initialize the watch window with maximum and minimum Iq
reference
              gMotorVars.SpinTAC.VelCtlOutputMax_A =
_IQmpy(STVELCTL_getOutputMaximum(stObj->velCtlHandle),
_IQ(USER_IQ_FULL_SCALE_CURRENT_A));
              gMotorVars.SpinTAC.VelCtlOutputMin_A =
_IQmpy(STVELCTL_getOutputMinimum(stObj->velCtlHandle),
_IQ(USER_IQ_FULL_SCALE_CURRENT_A));
            }

        }
      else
        {
          Flag_Latch_softwareUpdate = true;

          // the estimator sets the maximum current slope during
identification
          gMaxCurrentSlope = EST_getMaxCurrentSlope_pu(obj->estHandle);
        }


      // when appropriate, update the global variables
      if(gCounter_updateGlobals >=
NUM_MAIN_TICKS_FOR_GLOBAL_VARIABLE_UPDATE)
        {
          // reset the counter
          gCounter_updateGlobals = 0;

          updateGlobalVariables_motor(ctrlHandle, stHandle);
        }


      // update Kp and Ki gains
      updateKpKiGains(ctrlHandle);

      // set the SpinTAC (ST) bandwidth scale
```

94

```c
        STVELCTL_setBandwidth_radps(stObj->velCtlHandle,
gMotorVars.SpinTAC.VelCtlBw_radps);

        // set the maximum and minimum values for Iq reference
        STVELCTL_setOutputMaximums(stObj->velCtlHandle,
_IQmpy(gMotorVars.SpinTAC.VelCtlOutputMax_A,
_IQ(1.0/USER_IQ_FULL_SCALE_CURRENT_A)),
_IQmpy(gMotorVars.SpinTAC.VelCtlOutputMin_A,
_IQ(1.0/USER_IQ_FULL_SCALE_CURRENT_A)));

        // enable/disable the forced angle
        //EST_setFlag_enableForceAngle(obj-
>estHandle,gMotorVars.Flag_enableForceAngle);
        EST_setFlag_enableForceAngle(obj-
>estHandle,0);//////////////////////////////////////////////////////
/////////

        // enable or disable power warp

CTRL_setFlag_enablePowerWarp(ctrlHandle,gMotorVars.Flag_enablePowerWarp);

#ifdef DRV8301_SPI
        HAL_writeDrvData(halHandle,&gDrvSpi8301Vars);

        HAL_readDrvData(halHandle,&gDrvSpi8301Vars);
#endif
#ifdef DRV8305_SPI
        HAL_writeDrvData(halHandle,&gDrvSpi8305Vars);

        HAL_readDrvData(halHandle,&gDrvSpi8305Vars);
#endif
      } // end of while(gFlag_enableSys) loop


    // disable the PWM
    HAL_disablePwm(halHandle);

    // set the default controller parameters (Reset the control to re-
identify the motor)
    CTRL_setParams(ctrlHandle,&gUserParams);
    gMotorVars.Flag_Run_Identify = false;

    // setup the SpinTAC Components
    ST_setupVelCtl(stHandle);

  } // end of for(;;) loop

} // end of main() function


interrupt void mainISR(void)
{

  static uint16_t stCnt = 0;

  // toggle status LED
  if(gLEDcnt++ > (uint_least32_t)(USER_ISR_FREQ_Hz / LED_BLINK_FREQ_Hz))
  {
    HAL_toggleLed(halHandle,(GPIO_Number_e)HAL_Gpio_LED2);
```

```c
      gLEDcnt = 0;
    }


    // acknowledge the ADC interrupt
    HAL_acqAdcInt(halHandle,ADC_IntNumber_1);


    // convert the ADC data
    HAL_readAdcData(halHandle,&gAdcData);


    // Run the SpinTAC Components
    if(stCnt++ >= ISR_TICKS_PER_SPINTAC_TICK) {
            ST_runVelCtl(stHandle, ctrlHandle);
            stCnt = 1;
    }


    // run the controller
    CTRL_run(ctrlHandle,halHandle,&gAdcData,&gPwmData);


    // write the PWM compare values
    HAL_writePwmData(halHandle,&gPwmData);


    // setup the controller
    CTRL_setup(ctrlHandle);


    return;
} // end of mainISR() function


void updateGlobalVariables_motor(CTRL_Handle handle, ST_Handle sthandle)
{
    CTRL_Obj *obj = (CTRL_Obj *)handle;
    ST_Obj *stObj = (ST_Obj *)sthandle;


    // get the speed estimate
    gMotorVars.Speed_krpm = EST_getSpeed_krpm(obj->estHandle);

    // get the real time speed reference coming out of the speed trajectory
generator
    gMotorVars.SpeedTraj_krpm =
_IQmpy(CTRL_getSpd_int_ref_pu(handle),EST_get_pu_to_krpm_sf(obj-
>estHandle));

    // get the torque estimate
    gMotorVars.Torque_Nm = USER_computeTorque_Nm(handle,
gTorque_Flux_Iq_pu_to_Nm_sf, gTorque_Ls_Id_Iq_pu_to_Nm_sf);

    // get the magnetizing current
    gMotorVars.MagnCurr_A = EST_getIdRated(obj->estHandle);

    // get the rotor resistance
    gMotorVars.Rr_Ohm = EST_getRr_Ohm(obj->estHandle);
```
96

```c
  // get the stator resistance
  gMotorVars.Rs_Ohm = EST_getRs_Ohm(obj->estHandle);

  // get the stator inductance in the direct coordinate direction
  gMotorVars.Lsd_H = EST_getLs_d_H(obj->estHandle);

  // get the stator inductance in the quadrature coordinate direction
  gMotorVars.Lsq_H = EST_getLs_q_H(obj->estHandle);

  // get the flux in V/Hz in floating point
  gMotorVars.Flux_VpHz = EST_getFlux_VpHz(obj->estHandle);

  // get the flux in Wb in fixed point
  gMotorVars.Flux_Wb = USER_computeFlux(handle, gFlux_pu_to_Wb_sf);

  // get the controller state
  gMotorVars.CtrlState = CTRL_getState(handle);

  // get the estimator state
  gMotorVars.EstState = EST_getState(obj->estHandle);

  // Get the DC buss voltage
  gMotorVars.VdcBus_kV =
_IQmpy(gAdcData.dcBus,_IQ(USER_IQ_FULL_SCALE_VOLTAGE_V/1000.0));

  // get the Iq reference from the speed controller
  gMotorVars.IqRef_A = _IQmpy(STVELCTL_getTorqueReference(stObj-
>velCtlHandle), _IQ(USER_IQ_FULL_SCALE_CURRENT_A));

  // gets the Velocity Controller status
  gMotorVars.SpinTAC.VelCtlStatus = STVELCTL_getStatus(stObj-
>velCtlHandle);

  // get the inertia setting
  gMotorVars.SpinTAC.InertiaEstimate_Aperkrpm =
_IQmpy(STVELCTL_getInertia(stObj->velCtlHandle), _IQ(ST_SPEED_PU_PER_KRPM
* USER_IQ_FULL_SCALE_CURRENT_A));

  // get the friction setting
  gMotorVars.SpinTAC.FrictionEstimate_Aperkrpm =
_IQmpy(STVELCTL_getFriction(stObj->velCtlHandle), _IQ(ST_SPEED_PU_PER_KRPM
* USER_IQ_FULL_SCALE_CURRENT_A));

  // get the Velocity Controller error
  gMotorVars.SpinTAC.VelCtlErrorID = STVELCTL_getErrorID(stObj-
>velCtlHandle);

  return;
} // end of updateGlobalVariables_motor() function


void updateKpKiGains(CTRL_Handle handle)
{
  if((gMotorVars.CtrlState == CTRL_State_OnLine) &&
(gMotorVars.Flag_MotorIdentified == true) && (Flag_Latch_softwareUpdate ==
false))
    {
      // set the kp and ki speed values from the watch window
```

```c
        CTRL_setKp(handle,CTRL_Type_PID_spd,gMotorVars.Kp_spd);
        CTRL_setKi(handle,CTRL_Type_PID_spd,gMotorVars.Ki_spd);

        // set the kp and ki current values for Id and Iq from the watch
window
        CTRL_setKp(handle,CTRL_Type_PID_Id,gMotorVars.Kp_Idq);
        CTRL_setKi(handle,CTRL_Type_PID_Id,gMotorVars.Ki_Idq);
        CTRL_setKp(handle,CTRL_Type_PID_Iq,gMotorVars.Kp_Idq);
        CTRL_setKi(handle,CTRL_Type_PID_Iq,gMotorVars.Ki_Idq);
         }

  return;
} // end of updateKpKiGains() function


void ST_runVelCtl(ST_Handle handle, CTRL_Handle ctrlHandle)
{

    _iq speedFeedback, iqReference;
    ST_Obj *stObj = (ST_Obj *)handle;
    CTRL_Obj *ctrlObj = (CTRL_Obj *)ctrlHandle;

    //_iq outMin = _IQ(-0.2);
    //_iq outMax = _IQ(0.2);
    //PID_Handle pidHandle;

    // Get the mechanical speed in pu
    speedFeedback = EST_getFm_pu(ctrlObj->estHandle);

        // Run the SpinTAC Controller
        // Note that the library internal ramp generator is used to set the
speed reference
    STVELCTL_setVelocityReference(stObj->velCtlHandle,
TRAJ_getIntValue(ctrlObj->trajHandle_spd));
        STVELCTL_setAccelerationReference(stObj->velCtlHandle, _IQ(0.0)); //
Internal ramp generator does not provide Acceleration Reference
        STVELCTL_setVelocityFeedback(stObj->velCtlHandle, speedFeedback);
        STVELCTL_run(stObj->velCtlHandle);

        // select SpinTAC Velocity Controller
        iqReference = STVELCTL_getTorqueReference(stObj->velCtlHandle);


        //PID_setMinMax(pidHandle,_IQ24(-0.3),_IQ24(-0.3));

        //iqReference = torqueLimit;

        //ctrlHandle->pid_spd.outMax = _IQ24(0.3);

        //PID_setMinMax(ctrlObj->pidHandle_spd,outMin,outMax);
        // Set the Iq reference that came out of SpinTAC Velocity Control
        CTRL_setIq_ref_pu(ctrlHandle, iqReference);
}
```

Appendix C: Datasheets and User Manuals

DRV8312 EVALUATION BOARD
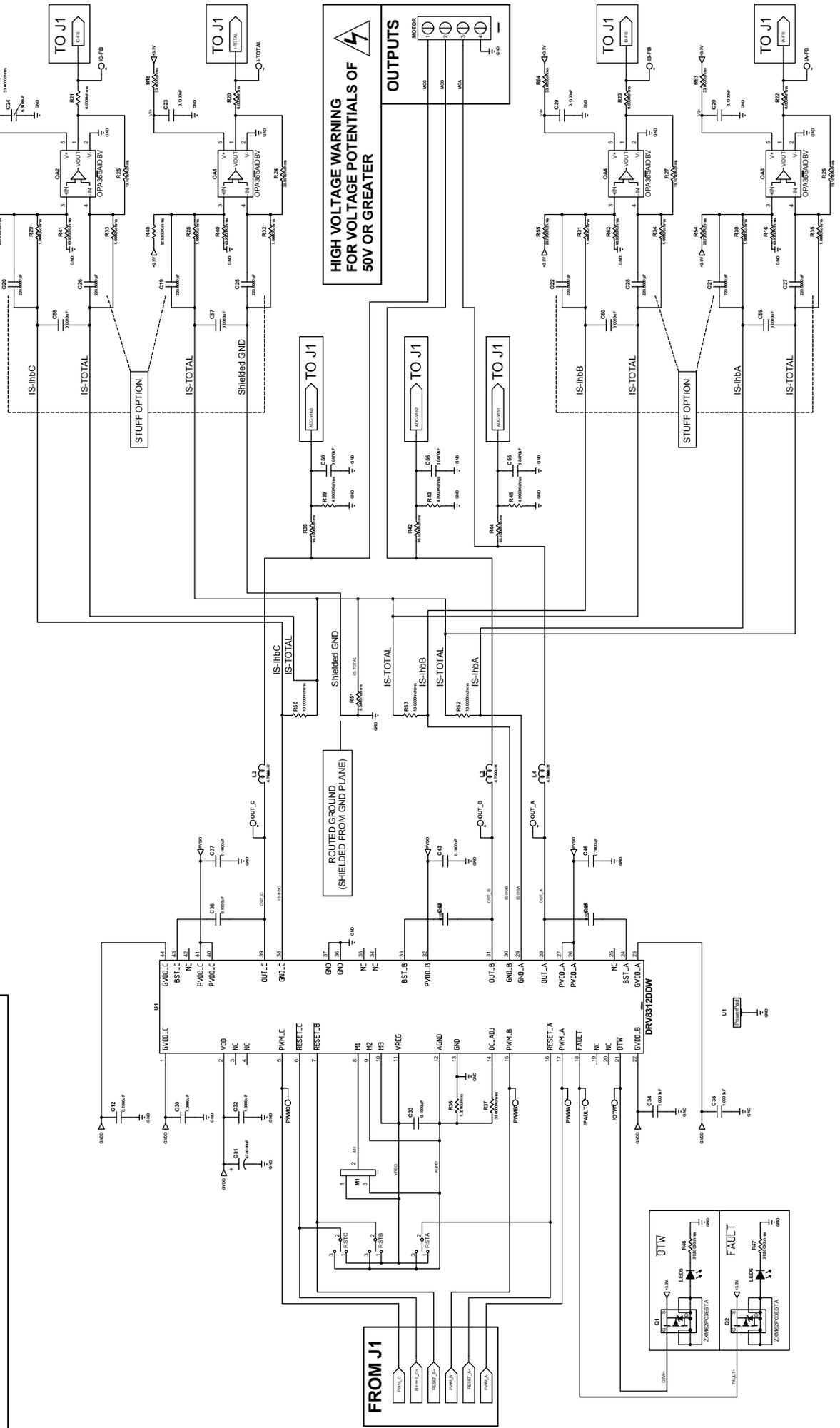
HIGH VOLTAGE WARNING
FOR VOLTAGE POTENTIALS OF
50V OR GREATER

OUTPUTS

ROUTED GROUND
(SHIELDED FROM GND PLANE)

FROM J1

OTW

FAULT

TI

DRV8312 EVALUATION BOARD

TO ISOLATION CIRCUIT

- SPI-SOMI
- SPI-STE
- CAN-TX
- CAN-RX
- SPI-CLK
- SPI-SIMO

TO U1

- PWM_B
- RESET_B
- RESET_A
- RESET_C
- PWM_C
- PWM_A

FROM U1

- ADC-Vhb1
- ADC-Vhb2
- ADC-Vhb3
- FAULT
- DTW
- I-TOTAL
- IC-FB
- IB-FB
- IA-FB

ENCODER

CAP

STUFF OPTION

TACH/SPEED INPUT

DRV8312 EVALUATION BOARD

GROUND TESTPOINTS

GND

BOM ONLY STANDOFFS

GND

EMI SNUBBER

PVDD
C2 0.1000uF
GND
PVDD R1 3.300ohms
C3 0.0100uF
GND

LED1 PVDD
R2 4.9900Kohms
GND

LED2 GVDD
R3 4.9900Kohms
GND

LED3 +5VD
R4 2.4900Kohms
GND

MAIN POWER IN

⚠ **HIGH VOLTAGE WARNING**

PVDDIN
1
2
GND

C11 470.0000uF
GND
C1 470.0000uF
PVDD
GND

PVDD
NOM = 50V
MAX = 52.5V

J9
SHUNT 3
SIGNAL 1
2
GND
SHIELD

EXTERNAL 12V INPUT

AUX-12V
1
2
GND

JP1

1-2: GVDD = VR1
2-3: GVDD = AUX 12V

GVDD
2
3 JP1 1

C10 100.0000uF
GND

R10 10.0000Kohms
R11 681.0000Kohms
GND

L1 39.0000uH

CR1 30BQ100
GND

R9 10.00Kohms

C9 12.0000pF
GND

R8 442.0000Kohms
C8 180.0000pF
GND

C7 0.1000uF

VR1
10 PH
9 GND
8 COMP
7 VSENSE
6 PWRGD
1 BOOT
2 VIN
3 EN
4 SS/TR
5 RT/CLK

TPS54160DGQ

VR1 PowerPad
GND

R7 205.0000Kohms
GND

C6 0.0150uF
GND
R5 348.0000Kohms
R6 33.2000Kohms
GND

C5 0.0150uF
GND

PVDD
C4 22.0000uF
GND

VR2
1 VIN VOUT 3
2 GND
UA78M05CDCY
GVDD
GND

+5V
C13 47.0000uF
C14 0.1000uF
GND

VR3
1 VIN VOUT 3
2 GND
TL1963A-33DCY
GVDD
GND

+3.3V
C15 47.0000uF
C16 0.1000uF
GND

VR4
1 VIN VOUT 2
3 GND
REF3025AIDBZ
+3.3V
C38 0.4700uF
+2.5V
GND

USER POWER ACCESS

J2
+5V 1
2
GND

J3
+3.3V 1
2
GND

DRV8312 EVALUATION BOARD

ISOLATION CIRCUIT schematic including VR5 (TPS73633MDBVREP), U3 (DCH010505S), U4 (ISO1050DUB), U5 (ISO7241ADW), connectors J7 and J8, and connections TO J1 (CAN-RX, CAN-TX, SPI-SOMI, SPI-SIMO, SPI-CLK, SPI-STE).

SCH REV	C
PCB REV	D
SHEET	4 OF 6
DRAWN BY	LDN

DATE MARCH 23, 2011
FILENAME DRV8312EVM_RevD

PAGE INFO: ISOLATION CIRCUIT
DESIGN LEAD RYAN KEHR

EDGE # 6517813

TI

DRV8312 EVALUATION BOARD

REVISION HISTORY

REVISION	DESCRIPTION	DATE	APPROVAL
A	INITIAL RELEASE	MAY 4, 2010	ML
B	1. DELETED OUTA, OUTB, OUTC. 2. ADDED OUTPUT CONNECTOR, TITLED MOTOR. 3. CONNECTED L2-2, L3-2 AND L4-2 TO MOTOR-1,2 AND 3, RESPECTIVELY. 4. DELETED PVDD, GND, +12V AND GND(GND2). 5. ADDED POWER CONNECTOR, VIN. 6. CONNECTED VIN-4 TO C1-1. 7. CONNECTED VIN-2,3 TO GND. 8. CONNECTED VIN-1 TO +12V. 9. ADDED TESTPOINTS TO IA-FB, IB-FB, IC-FB AND I-TOTAL. 10. ADDED 2 PIN HEADERS J2 AND J3 FOR USER POWER ACCESS. 11. CONNECTED J2-1,2 TO +5V AND GND. 12. CONNECTED J3-1,2 TO +3.3V AND GND. 13. COMBINED ENCODER AND CAP INTO ONE 2x5 HEADER, TITLED J4. 14. ADDED 3 PIN HEADER, JP3, LABELED TACH/SPEED INPUT. 15. CONNECTED JP3-1,2,3 TO +3.3V, J1-7 AND GND, RESPECTIVELY. 16. RENAMED S1 TO RSTA. 17. SWAPPED U1-29 AND U1-30. 18. ADDED 40 PIN HEADER, J5 AND CONNECTED PINS TO ACTIVE PINS OF J1. 19. ADDED SWITCHES, S1 AND S2 TO U1-76 AND U1-78. 20. COPIED DAC1 CIRCUIT, CALLED DAC3, AND CONNECTED TO J1-28. 21. CHANGED R24-R27 TO 33.0K OHMS. 22. CHANGED R28-R35, R56-R61 TO 1.00K OHMS. 23. CHANGED R48, R49, R54 AND R55 TO 49.9K OHMS. 24. ADDED ISOLATION CIRCUITS. 25. ADDED 12V CONNECTOR, J9. 26. CHANGED C4 TO 22UFD/100V. 27. CHANGED C5 AND C6 TO 100V.	MARCH 23, 2011	RK

TEXAS INSTRUMENTS DISCLAIMER

The information and materials ("Materials") provided here are provided by Texas Instruments Incorporated ("TI") as a service to its customers and/or suppliers, and may be used for informational purposes only, and only subject to the following terms. By downloading or viewing these Materials, you are signifying your assent to these terms.

1.) These preliminary evaluation schematics are intended for use for PRELIMINARY ENGINEERING DEVELOPMENT AND EVALUATION PURPOSES ONLY and are not considered by Texas Instruments to be fit as a basis for establishing production products or systems. This information may be incomplete in several respects, including but not limited to information relating to required design, marketing, and/or manufacturing-related protective considerations and product safety measures typically found in the end-product incorporating the goods.

2.) Accordingly, neither TI nor its suppliers warrant the accuracy or completeness of the information, text, graphics, links or other items contained within the Materials. TI may make changes to the Materials, or to the products described therein, at any time without notice. TI makes no commitment to update the Materials.

3.) TI assumes no liability for applications assistance, customer product design, software performance, or services that may be described or referenced in the Materials. The user assumes all responsibility and liability for proper and safe design and handling of goods. Accordingly, the user indemnifies TI from all claims arising from its use of the Materials.

4.) TI currently deals with various customers for products, and therefore our arrangement with the user will not be exclusive. TI makes no representations regarding the commercial availability of non-TI components that may be referenced in the Materials.

5.) No license is granted under any patent right or other intellectual property right of TI covering or relating to any combination, machine, or process in which such TI products or services might be or are used. Except as expressly provided herein, TI and its suppliers do not grant any express or implied right to you under any patents, copyrights, trademarks, or trade secret information.

6.) Performance tests and ratings, to the extent referenced in the Materials, are measured using specific computer systems and/or components and reflect the approximate performance of TI products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing.

7.) Resale of TI's products or services with statements different from or beyond the parameters stated by TI for that product or service in official TI data books or data sheets voids all express and any implied warranties for the associated TI product or service, and is an unfair and deceptive business practice, and TI is not responsible for any such use.

8.) The Materials are copyrighted and any unauthorized use may violate copyright, trademark, and other laws. You may only download one copy for your internal use only, unless you are specifically licensed to do otherwise by TI in writing. This is a license, not a transfer of title, and is subject to the following restrictions: You may not:

(a) modify the Materials (including any associated warranties, conditions, limitations or notices) or use them for any commercial purpose, or any public display, performance, sale or rental;

(b) decompile, reverse engineer, or disassemble software Materials except and only to the extent permitted by applicable law;

(c) remove any copyright or other proprietary notices from the Materials;

(d) transfer the Materials to another person. You agree to prevent any unauthorized copying of the Materials. TI may terminate this license at any time if you are in breach of the terms of this Agreement. Upon termination, you will immediately destroy the Materials.

9.) THE MATERIALS ARE PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT OF INTELLECTUAL PROPERTY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT SHALL TI OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, LOSS OF INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE MATERIALS, EVEN IF TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TI

Diameter ø50mm Shaft type Absolute Rotary Encoder

◧ Features

● Compact size of external diameter ø50mm
● Various output code: BCD, Binary, Gray code
● Various and high resolution(720, 1024 divisions)
● Protection structure IP64(Dust-proof, Oil-proof)

◧ Applications

● Precision machine tool, Fabric machinery, Robot, Parking system

⚠ Please read "Caution for your safety" in operation manual before using.　　CE

◧ Ordering information

EP50S	8	– 1024 –	1	R	– P	– 24
Series	Shaft diameter	Pulse/1Revolution	Output code	Revolution direction	Control output	Power supply
Diameter ø50mm shaft type	ø8mm	Refer to resolution	1 : BCD Code 2 : Binary Code 3 : Gray Code	F : Output value increase at CW direction R : Output value increase at CCW direction	P : PNP open collector output N : NPN open collector output	5 : 5VDC ±5% 24 : 12-24VDC ±5%

◧ Specifications

Item			Diameter ø50mm shaft type of absolute rotary encoder						
Resolution			6, 8, 10, 12, 16, 20, 24, 32, 40, 45, 48, 64, 90, 128, 180, 256, 360, 512, 720, 1024						
Electrical specification	Output code		BCD Code	Binary Code	Gray Code		BCD Code	Binary Code	Gray Code
	Output phase / Output angle※1	1024-division	TS: 0.3515°±15' (13bit)	TS: 0.3515°±15' (10bit)	TS: 0.703°±15' (10bit)	20-division	TP1:12°±60'(1bit) TP2:2°±60'(1bit) TS: 18°±60'(5bit) EP: 18°±60'(1bit)	TP1: 12°±60'(1bit) TP2: 2°±60'(1bit) TS: 18°±60'(5bit) EP: 18°±60'(1bit)	TP1: 12°±60'(1bit) TP2: 2°±60'(1bit) TS: 36°±60'(5bit) EP:18°±60'(1bit)
		720-division	TS: 0.5°±25' (11bit)	TS: 0.5°±25' (10bit)	TS: 1°±25' (10bit)				
		512-division	TS: 0.703°±15' (11bit)	TS: 0.703°±15' (9bit)	TS: 1.406°±15' (9bit)				
		360-division	TS: 1°±25'(10bit)	TS: 1°±25'(9bit)	TS: 2°±25'(9bit)				
		256-division	TS: 1.406°±15' (10bit)	TS: 1.406°±15' (8bit)	TS: 2.8125°±15' (8bit)	16-division	TP1: 15°±60'(1bit) TP2:2°±60'(1bit) TS: 22.5°±60' (5bit) EP: 22.5°±60' (1bit)	TP1:15°±60'(1bit) TP2:2°±60'(1bit) TS: 22.5°±60' (4bit) EP: 22.5°±60' (1bit)	TP1:15°±60'(1bit) TP2: 2°±60'(1bit) TS:45°±60'(4bit) EP: 22.5°±60' (1bit)
		180-division	TS: 2°±25'(9bit)	TS: 2°±25'(8bit)	TS: 4°±25'(8bit)				
		128-division	TS: 2.8125°±15' (9bit)	TS: 2.8125°±15' (7bit)	TS:5S625°±15' (7bit)				
		90-division	TS: 4°±25'(8bit)	TS: 4°±25'(7bit)	TS: 8°±25'(7bit)				
		64-division	TS: 5.625°±15'(7bit)	TS: 5.625°±15'(6bit)	TS: 11.25°±15'(6bit)	12-division	TP1:15°±60'(1bit) TP2:3°±60'(1bit) TS:30°±60'(5bit) EP:30°±60'(1bit)	TP1: 15°±60'(1bit) TP2: 3°±60'(1bit) TS: 30°±60'(4bit) EP: 30°±60'(1bit)	TP1: 15°±60'(1bit) TP2: 3°±60'(1bit) TS: 60°±60'(4bit) EP: 30°±60'(1bit)
		48-division	TS: 7.5°±25'(7bit)	TS: 7.5°±25'(6bit)	TS: 15°±25'(6bit)				
		45-division	TS: 8°±25'(7bit)	TS: 8°±25'(6bit)	TS: 16°±25'(6bit)				
		40-division	TP1: 5°±60'(1bit) TP2: 2°±60'(1bit) TS: 9°±60'(6bit) EP: 9°±60'(1bit)	TP1: 5°±60'(1bit) TP2: 2°±60'(1bit) TS: 9°±60'(6bit) EP: 9°±60'(1bit)	TP1: 5°±60'(1bit) TP2: 2°±60'(1bit) TS: 18°±60'(6bit) EP: 9°±60'(1bit)	10-division	TP1:30°±60'(1bit) TP2: 12°±60'(1bit) TS: 36°±60'(4bit) EP: 36°±60'(1bit)	TP1: 30°±60'(1bit) TP2: 12°±60'(1bit) TS: 36°±60'(4bit) EP: 36°±60'(1bit)	TP1: 30°±60'(1bit) TP2: 12°±60'(1bit) TS: 72°±60'(4bit) EP: 36°±60'(1bit)
		32-division	TP1:7°±60'(1bit) TP2: 2°±60'(1bit) TS: 11.25°±60'(6bit) EP: 11.25°±60'(1bit)	TP1:7°±60'(1bit) TP2: 2°±60'(1bit) TS: 11.25°±60'(5bit) EP: 11.25°±60'(1bit)	TP1: 7°±60'(1bit) TP2: 2°±60'(1bit) TS: 22.5°±60'(5bit) EP: 11.25°±60'(1bit)	10-division	TP1:39°±60'(1bit) TP2: 15°±60'(1bit) TS: 45°±60'(3bit) EP: 45°±60'(1bit)	TP1: 39°±60'(1bit) TP2: 15°±60'(1bit) TS: 45°±60'(3bit) EP: 45°±60'(1bit)	TP1: 39°±60'(1bit) TP2: 15°±60'(1bit) TS: 90°±60'(3bit) EP: 45°±60'(1bit)
		24-division	TP1: 8°±60'(1bit) TP2: 3°±60'(1bit) TS: 15°±60'(6bit) EP: 15°±60'(1bit)	TP1: 8°±60'(1bit) TP2: 3°±60'(1bit) TS: 15°±60'(5bit) EP: 15°±60'(1bit)	TP1: 8°±60'(1bit) TP2: 3°±60'(1bit) TS: 30°±60'(5bit) EP: 15°±60'(1bit)	6-division	TP1: 53°±60'(1bit) TP2: 15°±60'(1bit) TS: 60°±60'(3bit) EP: 60°±60'(1bit)	TP1: 53°±60'(1bit) TP2: 15°±60'(1bit) TS: 60°±60'(3bit) EP: 60°±60'(1bit)	TP1: 53°±60'(1bit) TP2: 15°±60'(1bit) TS: 120°±60'(3bit) EP: 60°±60'(1bit)
	Control output	PNP open collector output	Output voltage : Min. (Power supply-1.5)VDC, Load current : Max. 32mA						
		NPN open collector output	Load current : Max. 32mA, Residual voltage : Max. 1VDC						
	Response time(Rise/Fall)		Ton=800nsec, Toff=Max. 800nsec(Cable length : 2m, I sink = 32mA)						
	Max. Response frequency		35kHz						
	Power supply		• 5VDC ±5%(Ripple P-P : Max. 5%) • 12-24VDC ±5%(Ripple P-P : Max. 5%)						
	Current consumption		Max. 100mA(disconnection of the load)						
	Insulation resistance		Min. 100MΩ(at 500VDC megger between all terminals and case)						
	Dielectric strength		750VAC 50/60Hz for 1 minute(Between all terminals and case)						
	Connection		Cable type(Cable gland)						

※1: TS=Signal Pulse, Tp=Timing Pulse, EP=Even Parity

ø50mm Shaft Absolute type

■ Specifications

Item		Diameter ø50mm shaft type of absolute rotary encoder
Mechanical specification	Starting torque	Max. 40gf·cm(0.004N·m)
	Moment of inertia	Max. 40g·cm²(4×10⁻⁵kg·m²)
	Shaft loading	Radial : 10kgf, Thrust : 2.5kgf
	Max. allowable revolution※²	3000rpm
Vibration		1.5mm amplitude or 300m/s² at frequency of 10 to 55Hz(for 1 min.) in each of X, Y, Z directions for 2 hours
Shock		Approx. Max. 50G
Environ-ment	Ambient temperature	-10 to 70℃, storage : -25 to 85℃
	Ambient humidity	35 to 85%RH, storage : 35 to 90%RH
Protection		IP64(IEC standard)
Cable		ø7, 15-wire, Length : 2m, Shield cable(AWG 28, Core diameter: 0.08mm, Number of cores: 40, Insulator out diameter: ø0.8)
Accessory		Fixing bracket, Coupling
Approval		C€
Unit weight		Approx. 380g

※2: Make sure that. Max response revolution should be lower than or equal to max. allowable revolution when selecting the resolution.

$$[\text{Max. response revolution(rpm)} = \frac{\text{Max. response frequency}}{\text{Resolution}} \times 60 \text{ sec}]$$

※Environment resistance is rated at no freezing or condensation.

■ Output waveform

● 24 division (BCD code output)

Model	EP50S8-24-1 □ - □ - □

※TP1=8°±60', TP2=3°±60', TS=15°±60', P=15°±60'
※Above waveform is based on the positive logic. (The output waveform of negative logic is opposed to above waveform.)

● 24 division (Binary code output)

Model	EP50S8-24-2 □ - □ - □

※TP1=8°±60', TP2=3°±60', TS=15°±60', P=15°±60'
※Above waveform is based on the positive logic. (The output waveform of negative logic is opposed to above waveform.)
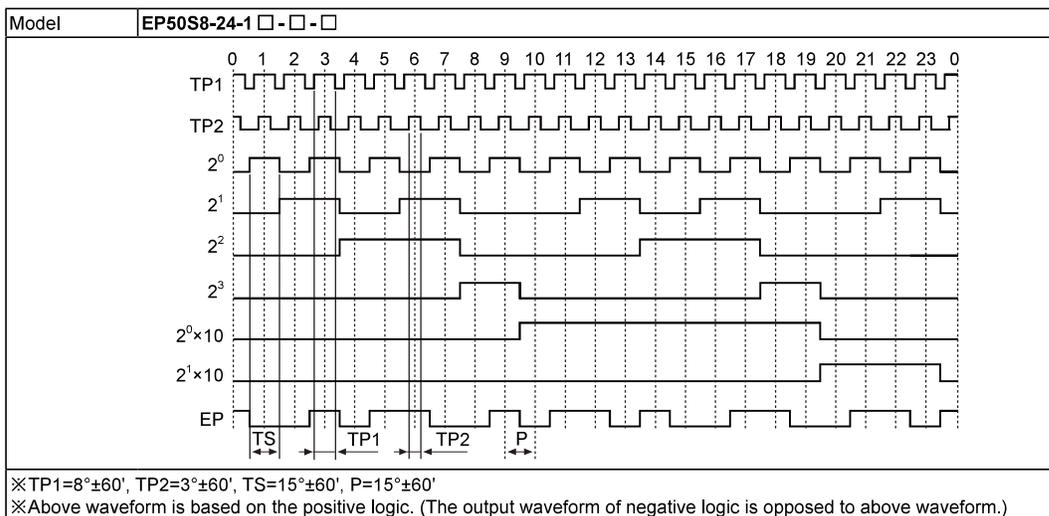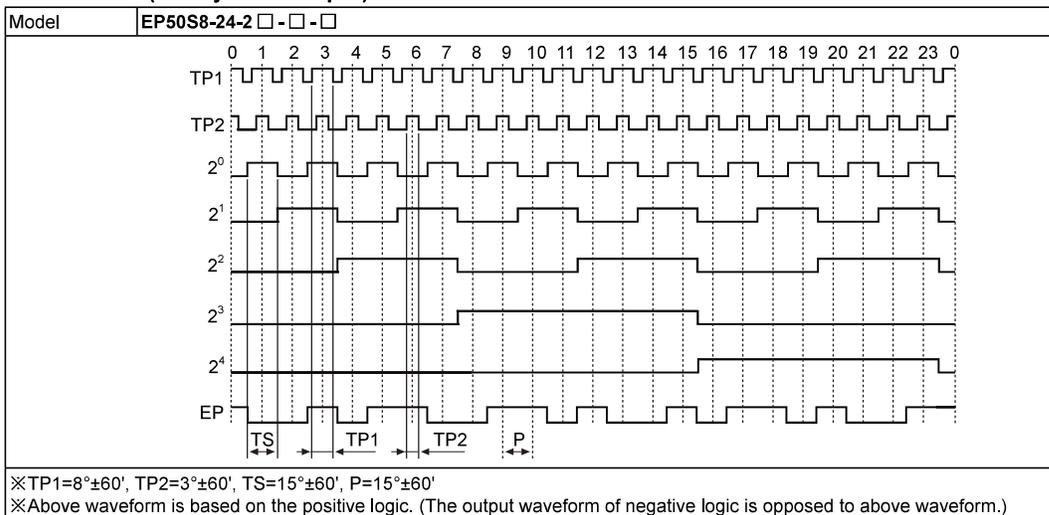
(A)
Photo electric sensor

(B)
Fiber optic sensor

(C)
Door/Area sensor

(D)
Proximity sensor

(E)
Pressure sensor

(F)
Rotary encoder

(G)
Connector/ Socket

(H)
Temp. controller

(I)
SSR/ Power controller

(J)
Counter

(K)
Timer

(L)
Panel meter

(M)
Tacho/ Speed/ Pulse meter

(N)
Display unit

(O)
Sensor controller

(P)
Switching mode power supply

(Q)
Stepper motor& Driver&Controller

(R)
Graphic/ Logic panel

(S)
Field network device

(T)
Software

(U)
Other

EP50S Series

■ Output waveform

● 360 division (BCD code output)

Model	EP50S8-360-1□-P-□

※TS=1°±25'
※Above waveform is based on the positive logic. (The output waveform of negative logic is opposite to the above waveform.)

● 360 division (Binary code output)

Model	EP50S8-360-2□-P-□

※TS=1°±25'
※Above waveform is based on the positive logic. (The output waveform of negative logic is opposite to the above waveform.)

● 1024 division (BCD code output)

Model	EP50S8-1024-1□-□-□

※TS=0.3515°±15'
※Above waveform is based on the positive logic. (The output waveform of negative logic is opposite to the above waveform.)
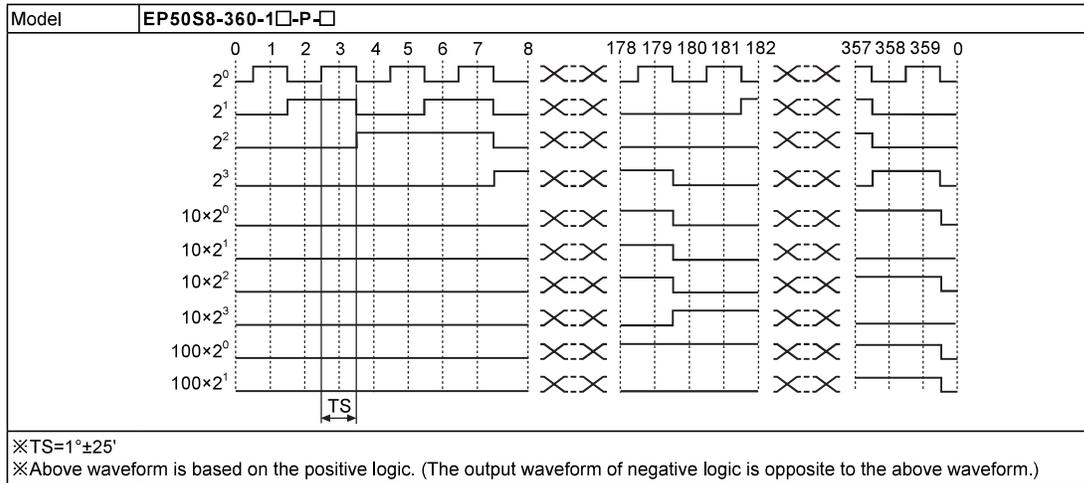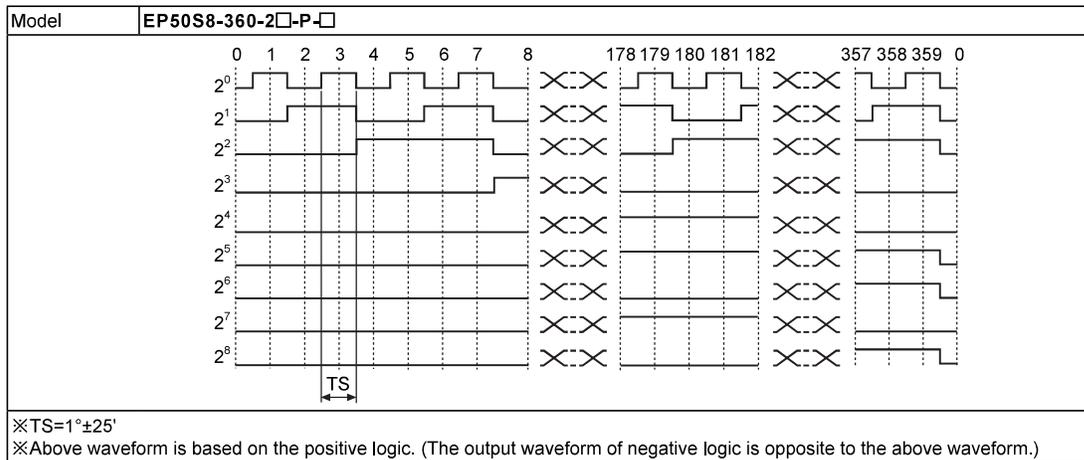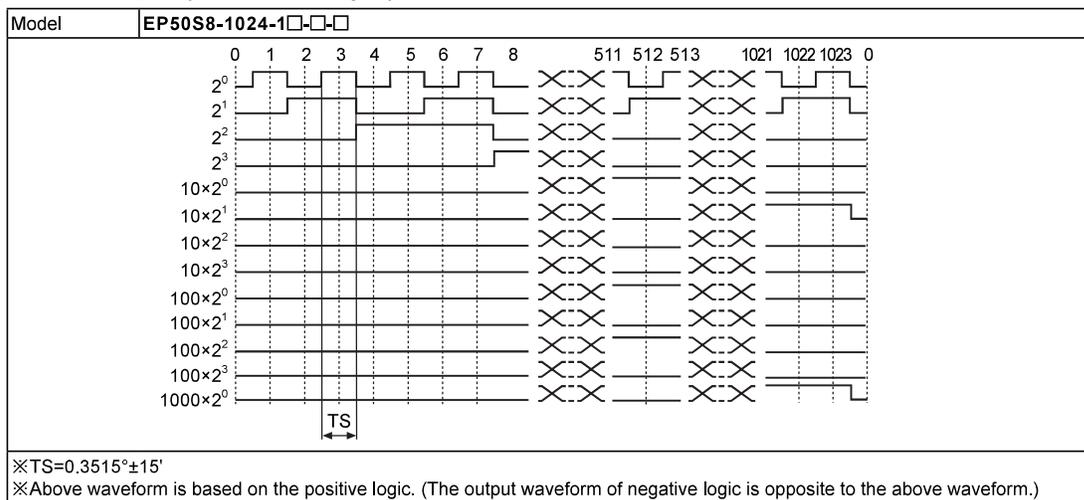
ø50mm Shaft Absolute type

▣ Output waveform
● 1024 division (Binary code output)

Model	EP50S8-1024-2□-P-□

※TS=0.3515°±15'
※Above waveform is based on the positive logic. (The output waveform of negative logic is opposite to the above waveform.)

▣ Control output diagram

PNP open collector output		NPN open collector output	
Rotary encoder circuit	Load connection	Rotary encoder circuit	Load connection
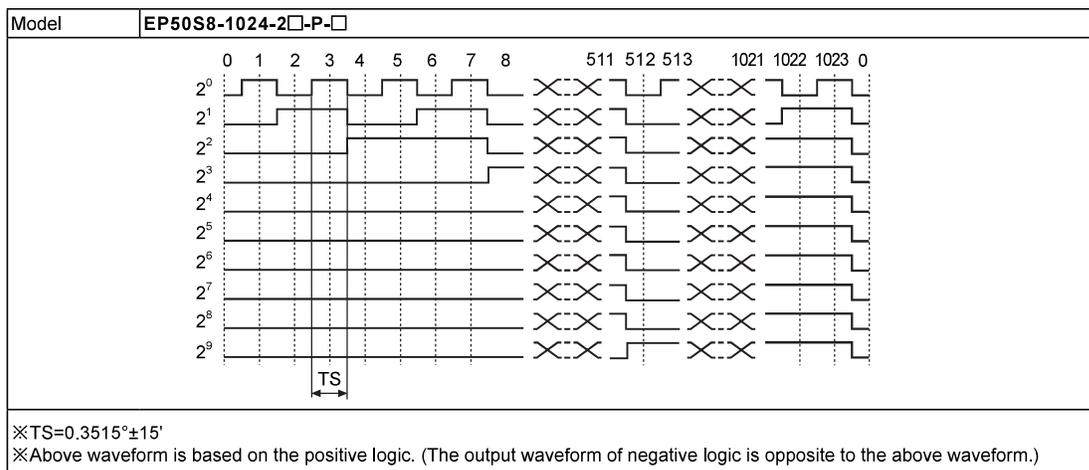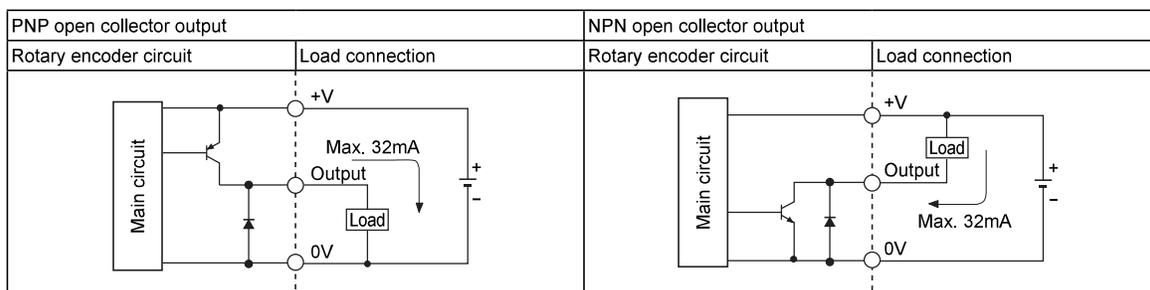
※Output circuits of all phases are the same.

▣ Connections
● BCD Code

	Resolution / Color	6-division	8-division	10-division	12-division	16-division	20-division	24-division	32-division	40-division	45-division	48-division	64-division	90-division	128-division	80-division	256-division	360-division	512-division	720-division	1024-division
Power	White	+V																			
	Boack	0V																			
Outpur wire	Brown	2^0																			
	Red	2^1																			
	Orange	2^2																			
	Yellow	N·C	2^3																		
	Blue	N·C		$2^0 \times 10$																	
	Purple	N·C			$2^{21} \times 10$																
	Gray	N·C				$2^2 \times 10$															
	White/Brown	TP1					N·C		$2^3 \times 10$												
	White/Red	TP2					N·C			$2^0 \times 10$											
	White/Orange	EP					N·C				$2^1 \times 100$										
	White/Yellow	N·C										$2^2 \times 100$									
	White/Blue	N·C																$2^3 \times 100$			
	White/Purple	N·C																$2^0 \times 1000$			
	Shield wire	F.G.																			

※Unused wires must be insulated.
※Encoder case and shield wire must be grounded(F.G.).
※N·C: Not Connected.
※Output cable must not be short-circuited, because Driver IC is used in output circuit.

■ Connections

● Binary Code/Gray Code

Color \ Resolution		6-divi-sion	8-divi-sion	10-divi-sion	12-divi-sion	16-divi-sion	20-divi-sion	24-divi-sion	32-divi-sion	40-divi-sion	45-divi-sion	48-divi-sion	64-divi-sion	90-divi-sion	128-divi-sion	80-divi-sion	256-divi-sion	360-divi-sion	512-divi-sion	720-divi-sion	1024-divi-sion
Power	White	+V																			
	Boack	0V																			
Outpur wire	Brown	2^0																			
	Red	2^1																			
	Orange	2^2																			
	Yellow	N·C	2^3																		
	Blue	N·C			2^4																
	Purple	N·C					2^5														
	Gray	N·C										2^6									
	White/Brown	TP1									N·C					2^7					
	White/Red	TP2									N·C							2^8			
	White/Orange	EP									N·C									2^9	
	Shield wire	F.G.																			

※Unused wires must be insulated.
※Encoder case and shield wire must be grounded(F.G.).
※N·C: Not Connected.
※Output cable must not be short-circuited, because Driver IC is used in output circuit.

■ Dimensions

(unit: mm)
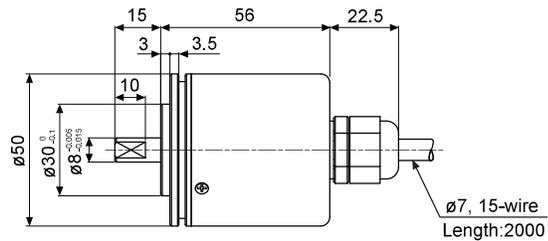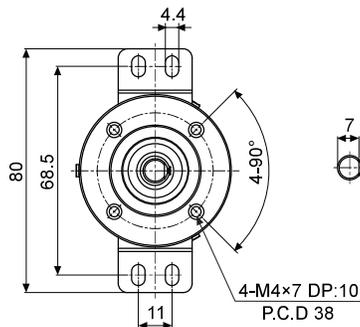
● Bracket
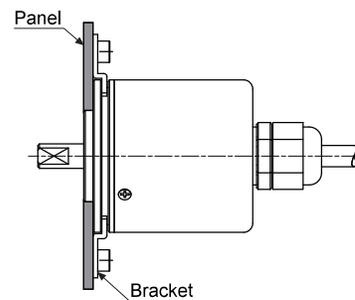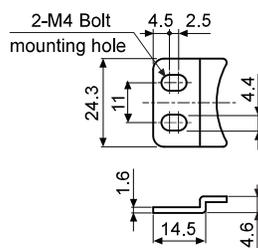
● Coupling(EP50S)
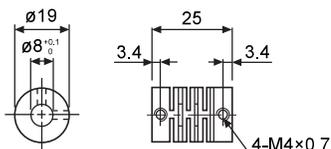
• Parallel misalignment : Max. 0.25mm
• Angular misalignment : Max. 5°
• End-play : Max. 0.2mm

※For parallel misalignment, angular misalignment, end-play terms, refer to the F-78 page.
※For flexible coupling(ERB Series) information, refer to the F-71 page.

1 Features

- Available in 8-Bump DSBGA Chip-Sized Package, (See AN-1112, SNVA009)
- Internally Frequency Compensated for Unity Gain
- Large DC Voltage Gain: 100 dB
- Wide Bandwidth (Unity Gain):   1 MHz (Temperature Compensated)
- Wide Power Supply Range:
  - Single Supply: 3V to 32V
  - Or Dual Supplies: ±1.5V to ±16V
- Very Low Supply Current Drain (500 µA)—Essentially Independent of Supply Voltage
- Low Input Offset Voltage: 2 mV
- Input Common-Mode Voltage Range Includes Ground
- Differential Input Voltage Range Equal to the Power Supply Voltage
- Large Output Voltage Swing
- Unique Characteristics:
  - In the Linear Mode the Input Common-Mode Voltage Range Includes Ground and the Output Voltage Can Also Swing to Ground, even though Operated from Only a Single Power Supply Voltage.
  - The Unity Gain Cross Frequency is Temperature Compensated.
  - The Input Bias Current is also Temperature Compensated.
- Advantages:
  - Two Internally Compensated Op Amps
  - Eliminates Need for Dual Supplies
  - Allows Direct Sensing Near GND and V_{OUT} Also Goes to GND
  - Compatible with All Forms of Logic
  - Power Drain Suitable for Battery Operation

2 Applications

- Active Filters
- General Signal Conditioning and Amplification
- 4- to 20-mA Current Loop Transmitters

3 Description

The LM158 series consists of two independent, high gain, internally frequency compensated operational amplifiers which were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

Application areas include transducer amplifiers, dc gain blocks and all the conventional op-amp circuits which now can be more easily implemented in single power supply systems. For example, the LM158 series can be directly operated off of the standard 3.3-V power supply voltage which is used in digital systems and will easily provide the required interface electronics without requiring the additional ±15V power supplies.

The LM358 and LM2904 are available in a chip sized package (8-Bump DSBGA) using TI's DSBGA package technology.

Device Information[1]

PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM158-N	TO-CAN (8)	9.08 mm x 9.09 mm
	CDIP (8)	10.16 mm x 6.502 mm
LM258-N	TO-CAN (8)	9.08 mm x 9.09 mm
LM2904-N	DSBGA (8)	1.31 mm x 1.31 mm
	SOIC (8)	4.90 mm x 3.91 mm
	PDIP (8)	9.81 mm x 6.35 mm
LM358-N	TO-CAN (8)	9.08 mm x 9.09 mm
	DSBGA (8)	1.31 mm x 1.31 mm
	SOIC (8)	4.90 mm x 3.91 mm
	PDIP (8)	9.81 mm x 6.35 mm

(1) For all available packages, see the orderable addendum at the end of the datasheet.

Voltage Controlled Oscillator (VCO)

An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

Table of Contents

4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision H (March 2013) to Revision I   **Page**

- Added *Pin Configuration and Functions* section, *ESD Ratings* table, *Feature Description* section, *Device Functional Modes*, *Application and Implementation* section, *Power Supply Recommendations* section, *Layout* section, *Device and Documentation Support* section, and *Mechanical, Packaging, and Orderable Information* section 1

Changes from Revision G (March 2013) to Revision H   **Page**

- Changed layout of National Data Sheet to TI format 25

Product Folder Links: *LM158-N LM258-N LM2904-N LM358-N*

5 Pin Configuration and Functions

D, P, and NAB Package
8-Pin SOIC, PDIP, and CDIP
Top View
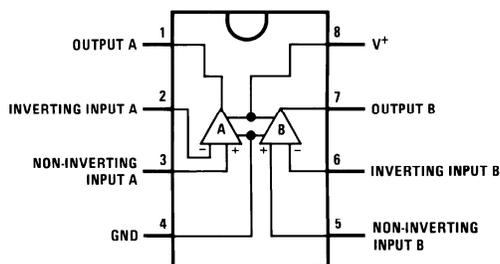
LMC Package
8-Pin TO-99
Top View
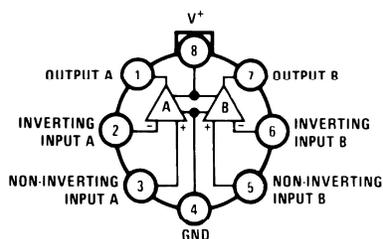
YPB Package
8-Pin DSBGA
Top View
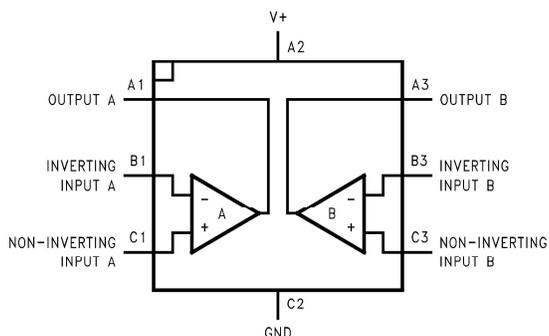
Pin Functions

PIN			TYPE	DESCRIPTION
D/P/LMC NO.	DSBGA NO.	NAME		
1	A1	OUTA	O	Output , Channel A
2	B1	-INA	I	Inverting Input, Channel A
3	C1	+INA	I	Non-Inverting Input, Channel A
4	C2	GND / V-	P	Ground for Single supply configurations. negative supply for dual supply configurations
5	C3	+INB	I	Output, Channel B
6	B3	-INB	I	Inverting Input, Channel B
7	A3	OUTB	O	Non-Inverting Input, Channel B
8	A2	V+	P	Positive Supply

6 Specifications

6.1 Absolute Maximum Ratings

See [1][2][3].

		LM158, LM258, LM358, LM158A, LM258A, LM358A		LM2904		UNIT
		MIN	MAX	MIN	MAX	
Supply Voltage, V$^+$			32		26	V
Differential Input Voltage			32		26	V
Input Voltage		−0.3	32	−0.3	26	V
Power Dissipation[4]	PDIP (P)		830		830	mW
	TO-99 (LMC)		550			mW
	SOIC (D)		530		530	mW
	DSBGA (YPB)		435			mW
Output Short-Circuit to GND (One Amplifier)[5]	V$^+$ ≤ 15 V and T$_A$ = 25°C		Continuous		Continuous	
Input Current (V$_{IN}$ < −0.3V)[6]			50		50	mA
Temperature		−55	125			°C
	PDIP Package (P): Soldering (10 seconds)		260		260	°C
	SOIC Package (D)	Vapor Phase (60 seconds)	215		215	°C
		Infrared (15 seconds)	220		220	°C
Lead Temperature	PDIP (P): (Soldering, 10 seconds)		260		260	°C
	TO-99 (LMC): (Soldering, 10 seconds)		300		300	°C
Storage temperature, T$_{stg}$		−65	150	−65	150	°C

(1) *Absolute Maximum Ratings* indicate limits beyond which damage to the device may occur. *Recommended Operating Conditions* indicate conditions for which the device is intended to be functional, but specific performance is not ensured. For ensured specifications and the test conditions, see the Electrical Characteristics.
(2) Refer to RETS158AX for LM158A military specifications and to RETS158X for LM158 military specifications.
(3) If Military/Aerospace specified devices are required, please contact the TI Sales Office/Distributors for availability and specifications.
(4) For operating at high temperatures, the LM358/LM358A, LM2904 must be derated based on a 125°C maximum junction temperature and a thermal resistance of 120°C/W for PDIP, 182°C/W for TO-99, 189°C/W for SOIC package, and 230°C/W for DSBGA, which applies for the device soldered in a printed circuit board, operating in a still air ambient. The LM258/LM258A and LM158/LM158A can be derated based on a +150°C maximum junction temperature. The dissipation is the total of both amplifiers—use external resistors, where possible, to allow the amplifier to saturate or to reduce the power which is dissipated in the integrated circuit.
(5) Short circuits from the output to V$^+$ can cause excessive heating and eventual destruction. When considering short circuits to ground, the maximum output current is approximately 40 mA independent of the magnitude of V$^+$. At values of supply voltage in excess of +15 V, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.
(6) This input current will only exist when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistors becoming forward biased and thereby acting as input diode clamps. In addition to this diode action, there is also lateral NPN parasitic transistor action on the IC chip. This transistor action can cause the output voltages of the op amps to go to the V$^+$ voltage level (or to ground for a large overdrive) for the time duration that an input is driven negative. This is not destructive and normal output states will re-establish when the input voltage, which was negative, again returns to a value greater than −0.3 V (at 25°C).

6.2 ESD Ratings

			VALUE	UNIT
V$_{(ESD)}$	Electrostatic discharge	Human-body model (HBM), per ANSI/ESDA/JEDEC JS-001[1]	±250	V

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.

6.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

	MIN	MAX	UNIT
Supply Voltage (V+ - V-):LM158, LM258, LM358	3 (±1.5)	32 (±16)	V
Supply Voltage (V+ - V-):LM2904	3 (±1.5)	26 (±13)	V
Operating Temperature: LM158	-55	125	°C
Operating Temperature: LM258	-25	85	°C
Operating Temperature: LM358	0	70	°C
Operating Temperature: LM2904	-40	85	°C

6.4 Thermal Information

THERMAL METRIC[1]	LM158-N, LM258-N, LM358-N	LM158-N	LM2904-N, LM358-N			UNIT
	LMC	NAB	YPB	D	P	
	8 PINS					
$R_{\theta JA}$ Junction-to-ambient thermal resistance	155	132	230	189	120	°C/W

(1) For more information about traditional and new thermal metrics, see the *IC Package Thermal Metrics* application report, SPRA953.

6.5 Electrical Characteristics: LM158A, LM358A, LM158, LM258

V^+ = +5.0 V, See[1], unless otherwise stated

PARAMETER	TEST CONDITIONS	LM158A			LM358A			LM158, LM258			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
Input Offset Voltage	See[2], T_A = 25°C		1	2		2	3		2	5	mV
Input Bias Current	$I_{IN(+)}$ or $I_{IN(-)}$, T_A = 25°C,		20	50		45	100		45	150	nA
	V_{CM} = 0 V, [3]										
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$, V_{CM} = 0V, T_A = 25°C		2	10		5	30		3	30	nA
Input Common-Mode	V^+ = 30 V, [4]										
Voltage Range	(LM2904, V^+ = 26V), T_A = 25°C	0		$V^+-1.5$	0		$V^+-1.5$	0		$V^+-1.5$	V
Supply Current	Over Full Temperature Range										
	R_L = ∞ on All Op Amps										
	V^+ = 30V (LM2904 V^+ = 26V)		1	2		1	2		1	2	mA
	V^+ = 5V		0.5	1.2		0.5	1.2		0.5	1.2	mA
Large Signal Voltage Gain	V^+ = 15 V, T_A = 25°C, R_L ≥ 2 kΩ, (For V_O = 1 V to 11 V)	50	100		25	100		50	100		V/mV
Common-Mode	T_A = 25°C,	70	85		65	85		70	85		dB
Rejection Ratio	V_{CM} = 0 V to $V^+-1.5$ V										
Power Supply	V^+ = 5 V to 30 V	65	100		65	100		65	100		dB
Rejection Ratio	(LM2904, V^+ = 5 V to 26 V), T_A = 25°C										

(1) These specifications are limited to −55°C ≤ T_A ≤ +125°C for the LM158/LM158A. With the LM258/LM258A, all temperature specifications are limited to −25°C ≤ T_A ≤ 85°C, the LM358/LM358A temperature specifications are limited to 0°C ≤ T_A ≤ 70°C, and the LM2904 specifications are limited to −40°C ≤ T_A ≤ 85°C.

(2) V_O ≈ 1.4 V, R_S = 0 Ω with V^+ from 5 V to 30 V; and over the full input common-mode range (0 V to V^+ −1.5 V) at 25°C. For LM2904, V^+ from 5 V to 26 V.

(3) The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.

(4) The input common-mode voltage of either input signal voltage should not be allowed to go negative by more than 0.3 V (at 25°C). The upper end of the common-mode voltage range is V^+ −1.5 V (at 25°C), but either or both inputs can go to 32 V without damage (26 V for LM2904), independent of the magnitude of V^+.

Electrical Characteristics: LM158A, LM358A, LM158, LM258 (continued)

V^+ = +5.0 V, See[1], unless otherwise stated

PARAMETER		TEST CONDITIONS	LM158A			LM358A			LM158, LM258			UNIT
			MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
Power Supply		V^+ = 5 V to 30 V	65	100		65	100		65	100		dB
Rejection Ratio		(LM2904, V^+ = 5 V to 26 V), T_A = 25°C										
Amplifier-to-Amplifier Coupling		f = 1 kHz to 20 kHz, T_A = 25°C (Input Referred), See[5]		−120			−120			−120		dB
Output Current	Source	V_{IN}^+ = 1 V, V_{IN}^- = 0 V, V^+ = 15 V, V_O = 2 V, T_A = 25°C	20	40		20	40		20	40		mA
	Sink	V_{IN}^- = 1 V, V_{IN}^+ = 0 V V^+ = 15 V, T_A = 25°C, V_O = 2 V	10	20		10	20		10	20		mA
		V_{IN}^- = 1 V, V_{IN}^+ = 0 V T_A = 25°C, V_O = 200 mV, V^+ = 15 V	12	50		12	50		12	50		µA
Short Circuit to Ground		T_A = 25°C, See[6], V^+ = 15 V		40	60		40	60		40	60	mA
Input Offset Voltage		See[2]			4			5			7	mV
Input Offset Voltage Drift		R_S = 0Ω		7	15		7	20		7		µV/°C
Input Offset Current		$I_{IN(+)} - I_{IN(-)}$			30			75			100	nA
Input Offset Current Drift		R_S = 0Ω		10	200		10	300		10		pA/°C
Input Bias Current		$I_{IN(+)}$ or $I_{IN(-)}$		40	100		40	200		40	300	nA
Input Common-Mode Voltage Range		V^+ = 30 V, See[4] (LM2904, V^+ = 26 V)	0		V^+−2	0		V^+−2	0		V^+−2	V
Large Signal Voltage Gain		V^+ = +15 V (V_O = 1 V to 11 V) $R_L ≥ 2$ kΩ	25			15			25			V/mV
Output	V_{OH}	V^+ = +30 V, R_L = 2 kΩ	26			26			26			V
Voltage		(LM2904, V^+ = 26 V), R_L = 10 kΩ	27	28		27	28		27	28		V
Swing	V_{OL}	V^+ = 5V, R_L = 10 kΩ		5	20		5	20		5	20	mV
Output Current	Source	V_{IN}^+ = +1 V, V_{IN}^- = 0 V, V^+ = 15 V, V_O = 2 V	10	20		10	20		10	20		mA
	Sink	V_{IN}^- = +1 V, V_{IN}^+ = 0 V, V^+ = 15 V, V_O = 2 V	10	15		5	8		5	8		mA

(5) Due to proximity of external components, insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitance increases at higher frequencies.

(6) Short circuits from the output to V^+ can cause excessive heating and eventual destruction. When considering short circuits to ground, the maximum output current is approximately 40 mA independent of the magnitude of V^+. At values of supply voltage in excess of +15 V, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.

Product Folder Links: *LM158-N LM258-N LM2904-N LM358-N*

6.6 Electrical Characteristics: LM358, LM2904

V^+ = +5.0 V, See[1], unless otherwise stated

PARAMETER		TEST CONDITIONS	LM358			LM2904			UNIT
			MIN	TYP	MAX	MIN	TYP	MAX	
Input Offset Voltage		See[2] , T_A = 25°C		2	7		2	7	mV
Input Bias Current		$I_{IN(+)}$ or $I_{IN(-)}$, T_A = 25°C, V_{CM} = 0 V, See[3]		45	250		45	250	nA
Input Offset Current		$I_{IN(+)}$ − $I_{IN(-)}$, V_{CM} = 0 V, T_A = 25°C		5	50		5	50	nA
Input Common-Mode Voltage Range		V^+ = 30 V, See[4] (LM2904, V^+ = 26 V), T_A = 25°C	0		V^+−1.5	0		V^+−1.5	V
Supply Current		Over Full Temperature Range							
		R_L = ∞ on All Op Amps							
		V^+ = 30 V (LM2904 V^+ = 26 V)		1	2		1	2	mA
		V^+ = 5 V		0.5	1.2		0.5	1.2	mA
Large Signal Voltage		V^+ = 15V, T_A = 25°C,							
Gain		R_L ≥ 2 kΩ, (For V_O = 1 V to 11 V)	25	100		25	100		V/mV
Common-Mode Rejection Ratio		T_A = 25°C,	65	85		50	70		dB
		V_{CM} = 0 V to V^+−1.5 V							
Power Supply Rejection Ratio		V^+ = 5 V to 30 V	65	100		50	100		dB
		(LM2904, V^+ = 5 V to 26 V), T_A = 25°C							
Amplifier-to-Amplifier Coupling		f = 1 kHz to 20 kHz, T_A = 25°C (Input Referred), See[5]		−120			−120		dB
Output Current	Source	V_{IN}^+ = 1 V,	20	40		20	40		mA
		V_{IN}^- = 0 V,							
		V^+ = 15 V,							
		V_O = 2 V, T_A = 25°C							
	Sink	V_{IN}^- = 1 V, V_{IN}^+ = 0 V	10	20		10	20		mA
		V^+ = 15V, T_A = 25°C,							
		V_O = 2 V							
		V_{IN}^- = 1 V,	12	50		12	50		µA
		V_{IN}^+ = 0 V							
		T_A = 25°C, V_O = 200 mV,							
		V^+ = 15 V							
Short Circuit to Ground		T_A = 25°C, See[6], V^+ = 15 V		40	60		40	60	mA
Input Offset Voltage		See[2]			9			10	mV
Input Offset Voltage Drift		R_S = 0 Ω		7			7		µV/°C
Input Offset Current		$I_{IN(+)}$ − $I_{IN(-)}$			150		45	200	nA
Input Offset Current Drift		R_S = 0 Ω		10			10		pA/°C
Input Bias Current		$I_{IN(+)}$ or $I_{IN(-)}$		40	500		40	500	nA

(1) These specifications are limited to −55°C ≤ T_A ≤ +125°C for the LM158/LM158A. With the LM258/LM258A, all temperature specifications are limited to −25°C ≤ T_A ≤ 85°C, the LM358/LM358A temperature specifications are limited to 0°C ≤ T_A ≤ 70°C, and the LM2904 specifications are limited to −40°C ≤ T_A ≤ 85°C.
(2) V_O ≃ 1.4 V, R_S = 0 Ω with V^+ from 5 V to 30 V; and over the full input common-mode range (0 V to V^+ −1.5 V) at 25°C. For LM2904, V^+ from 5 V to 26 V.
(3) The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.
(4) The input common-mode voltage of either input signal voltage should not be allowed to go negative by more than 0.3 V (at 25°C). The upper end of the common-mode voltage range is V^+ −1.5 V (at 25°C), but either or both inputs can go to 32 V without damage (26 V for LM2904), independent of the magnitude of V^+.
(5) Due to proximity of external components, insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitance increases at higher frequencies.
(6) Short circuits from the output to V^+ can cause excessive heating and eventual destruction. When considering short circuits to ground, the maximum output current is approximately 40 mA independent of the magnitude of V^+. At values of supply voltage in excess of +15 V, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.

Electrical Characteristics: LM358, LM2904 (continued)

V^+ = +5.0 V, See[1], unless otherwise stated

PARAMETER		TEST CONDITIONS		LM358			LM2904			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	
Input Common-Mode Voltage Range		V^+ = 30 V, See[4] (LM2904, V^+ = 26 V)		0		V^+−2	0		V^+−2	V
Large Signal Voltage Gain		V^+ = +15 V		15			15			V/mV
		(V_O = 1 V to 11 V)								
		$R_L \geq$ 2 kΩ								
Output	V_{OH}	V^+ = 30 V	R_L = 2 kΩ	26			22			V
Voltage		(LM2904, V^+ = 26 V)	R_L = 10 kΩ	27	28		23	24		V
Swing	V_{OL}	V^+ = 5 V, R_L = 10 kΩ			5	20		5	100	mV
Output Current	Source	V_{IN}^+ = 1 V, V_{IN}^- = 0 V,		10	20		10	20		mA
		V^+ = 15 V, V_O = 2 V								
	Sink	V_{IN}^- = 1 V, V_{IN}^+ = 0 V,		5	8		5	8		mA
		V^+ = 15 V, V_O = 2 V								

Product Folder Links: LM158-N LM258-N LM2904-N LM358-N

S2

Servo Pneumatic Proportional Control System

Simple by design, the S2 is an all-in-one pneumatic positioning system capable of tackling high speed and high force applications while linearly positioning a cylinder anywhere along its stroke.

Features

· Easy Tuning
· Simple Setup and Wiring
· Wide Range of Input Pressures
· Minimized Air Consumption
· All-in-One Package

Applications

· Line and Hopper Feed Rate
· Edge Guide Positioning
· Case Erect and Pack
· Pick and Place
· Lane Divide and Divert
· Gripper Position Control
· Quality Control Sorting
· Component Placing
· Actuate and Pilot Process Valves
· Material Processing
· Level and Elevator Control
· Sluice Gate Control
· Rotary Indexing and Positioning

Compatibility

Add position control to almost any application. Typical compatibility:

· Double Acting Non-Repairable Cylinders
· ISO & Tie Rod Cylinders
· Grippers & End Effectors
· Vane or Rack & Pinion Rotary Cylinders

Mechanical Specifications

Pressure:
0...10 bar (0...150 psig)

Ports:
¼" NPTF

Connector:
5-pin M8 x 1 (male)

Mounting:
2 x M5 (10-32) Thru Holes

Temperature Range:
0° – 40°C (32°– 104°F)

Filtration:
5 µm Particulate
0.3 µm Coalescing

Media:
Unlubricated, Dry, Neutral Gas

Height:
126 mm (5.00 in)

Width:
64 mm   (2.50 in)

Length:
68 mm   (2.75 in)

Weight:
0.91 kg   (2.00 lbs)

Material Specifications

Body:
Aluminum 6061

Caps:
PA66 30% Glass Filled Nylon

Other:
Nitrile, 440C SS, Nickel Plated Steel

Electrical Specifications

Power Requirement:
12 ± 2 VDC
24 ± 4 VDC @ 20W

Command Input Impedance:
0...10VDC: 100kΩ
4...20mA: 210Ω

Feedback Input Impedance:
0...10VDC: 100kΩ
4...20mA: 210Ω

Command Input:
Configurable 0...10 VDC; 4...20mA

Feedback Input:
Configurable 0...10 VDC; 4...20mA

Electronic Adjustments:
USB-Connectable User Interface

Status Indications:
2 Power and Status LEDs

Excitation:
+10V (15mA max)

Performance Specifications

Positional Accuracy:
± 0.1...1% of Full Scale (typical)

Flow:
1300 SLPM 5.5 → 0 bar
(46 SCFM 80 → 0 psi)

Leak Rate:
5.7 SLPM 10 → 0 bar
(12 SCFH 150 → 0 psi)

Order Code:
S2-025-U-04

ENFIELD TECHNOLOGIES

S2

Servo Pneumatic Proportional Control System

ENFIELD®
TECHNOLOGIES

The S2 Servo Pneumatic Proportional Control System is ideal for applications requiring high speed and high forces simultaneously. Speeds up to 2.0 m/sec are achievable while slowing the cylinder to a smooth controlled stop, preventing premature cylinder wear and damage. Handling 50+ kilogram masses, it is capable of 0.5 m/sec across a selection of cylinders. The ability to handle heavy loads while quickly and accurately following a changing input signal makes the S2 the ideal choice for machine designers.

Maximum Cylinder Speed vs Moving Mass: Bore Size Comparison

32 mm / 1.25" — 74 in/s; 0 kg / 0 lb.; 45 / 100; 2.5 / 5; 50 / 25; 10 / 5; 20 / 10; 24; 62; 34; 55; 47

50 mm / 2" — 37 in/s; 0 kg / 0 lb.; 45 / 100; 2.5 / 5; 55 / 25; 10 / 5; 25 / 10; 23; 31; 35; 34

125 mm / 5" — 7.5 in/s; 0 kg / 0 lb.; 180 / 400; 45 / 100; 300 / 135; 150 / 65; 200 / 90; 4.5; 6.8; 5.3; 6.1; 5.7

254 mm / 10" — 1.75 in/s; 0 kg / 0 lb.; 450 / 1000; 45 / 100; 600 / 270; 200 / 90; 400 / 180; 1.75; 1.75; 1.75; 1.75; 1.75
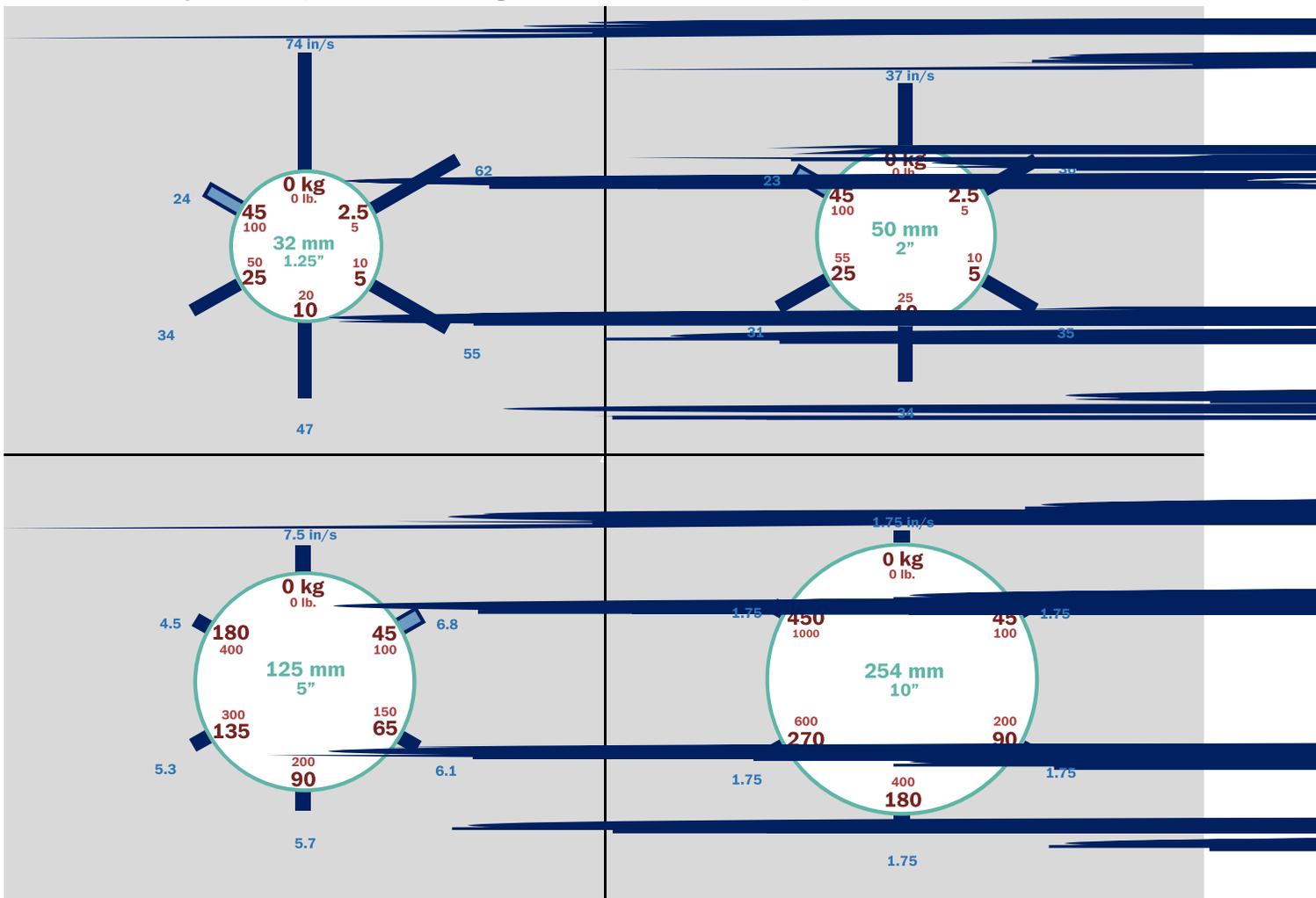
Note: Based on test results from typical setup: 250mm stroke guided horizontal cylinder with 5.5 bar inlet pressure.
Changes in moving mass, cylinder orientation or other system parameters will impact maximum speed.
Figures are to be used for general guidance only.

TMS320F28069F, TMS320F28068F, TMS320F28062F InstaSPIN™-FOC Software

Technical Reference Manual

![Texas Instruments logo]

Contents

List of Figures

List of Tables

TMS320F28069F, TMS320F28068F, TMS320F28062F InstaSPIN™-FOC Software

1 TMS320F2806xF InstaSPIN™-FOC Enabled MCUs

TMS320F2806xF are the first family of devices (69F, 68F, and 62F — 80- or 100-pin packages) from Texas Instruments that include the FAST™ (Figure 1) estimator and additional motor control functions needed for cascaded speed and torque loops for efficient three-phase field-oriented motor control (FOC).

Together — with F2806xF peripheral drivers in user code — they enable a sensorless (also known as self-sensing) InstaSPIN-FOC solution which can identify, tune the torque controller and efficiently control your motor in minutes, without the use of any mechanical rotor sensors. This entire package is called InstaSPIN-FOC, which is made available in ROM. The user also has the option of executing all FOC functions in user memory (FLASH or RAM), which makes calls to the proprietary FAST estimator firmware in ROM. InstaSPIN-FOC was designed for flexibility to accommodate a range of system software architectures and customization. The range of this flexibility is shown in Figure 2 and Figure 3.

This document is a supplement to all standard TMS320F2806x documentation, including the standard device data sheet [*TMS320F2806x Piccolo Microcontrollers* (literature number SPRS698)], technical reference manual, and user's guides. An additional document included with the InstaSPIN-FOC documentation package is the *TMS320F2806xF, TMS320F2802xF InstaSPIN-FOC/TMS320F2806xM InstaSPIN-MOTION User's Guide* (literature number SPRUHJ1), which covers the scope and functionality of:

- F2806xF devices
- F2806xF ROM contents
- FAST flux estimator
- InstaSPIN-FOC system solutions.
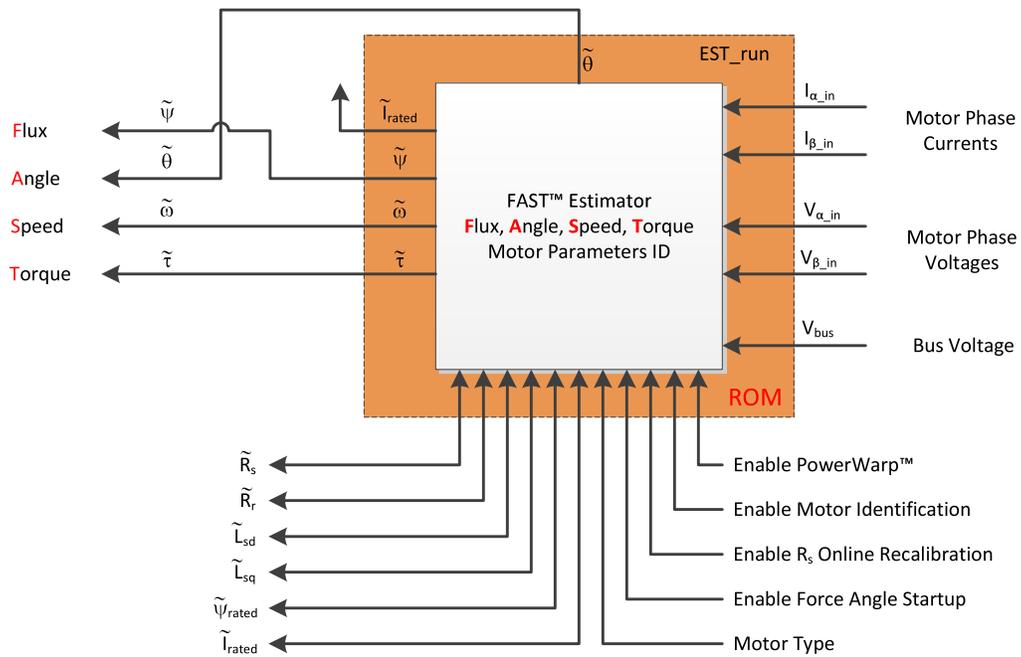
Figure 1. FAST - Estimating Flux, Angle, Speed, Torque - Automatic Motor Identification

2 FAST Estimator Features

- Unified observer structure which exploits the similarities between all motors that use magnetic flux for energy transduction
  - Both synchronous (BLDC, SPM, IPM), and asynchronous (ACIM) control are possible
  - Salient compensation for Interior Permanent Magnet motors: observer tracks rotor flux and angle correctly when Ls-d and Ls-q are provided
- Unique, high quality motor feedback signals for use in control systems
  - High-quality Flux signal for stable flux monitoring and field weakening
  - Superior rotor flux Angle estimation accuracy over wider speed range compared to traditional observer techniques independent of all rotor parameters for ACIM
  - Real-time low-noise motor shaft Speed signal
  - Accurate high bandwidth Torque signal for load monitoring and imbalance detection
- Angle estimator converges within first cycle of the applied waveform, regardless of speed
- Stable operation in all power quadrants, including generator quadrants
- Accurate angle estimation at steady state speeds below 1 Hz (typ) with full torque
- Angle integrity maintained even during slow speed reversals through zero speed
- Angle integrity maintained during stall conditions, enabling smooth stall recovery
- Motor Identification measures required electrical motor parameters of unloaded motor in under 2 minutes (typ)
- "On-the-fly" stator resistance recalibration (online Rs) tracks stator resistance changes in real time, resulting in robust operation over temperature. This feature can also be used as a temperature sensor of the motor's windings (basepoint calibration required)
- Superior transient response of rotor flux angle tracking compared to traditional observers
- PowerWarp™ adaptively reduces current consumption to minimize the combined (rotor and stator) copper losses to the lowest, without compromising ACIM output power levels

3 InstaSPIN™-FOC Solution Features

- Includes the Flux Angle Speed Torque (FAST) estimator, used to measure rotor flux (both magnitude and angle) in a sensorless field-oriented control (FOC) system
- Automatic torque (current) loop tuning, with option for user adjustments
- Automatic speed loop tuning provides stable operation for most applications. (Better transient response can be obtained by optimizing parameters for a particular application)
- Automatic or manual field weakening and field boosting
- Bus Voltage compensation
- Automatic offset calibration insures quality samples of feedback signals

4 InstaSPIN-FOC Block Diagrams
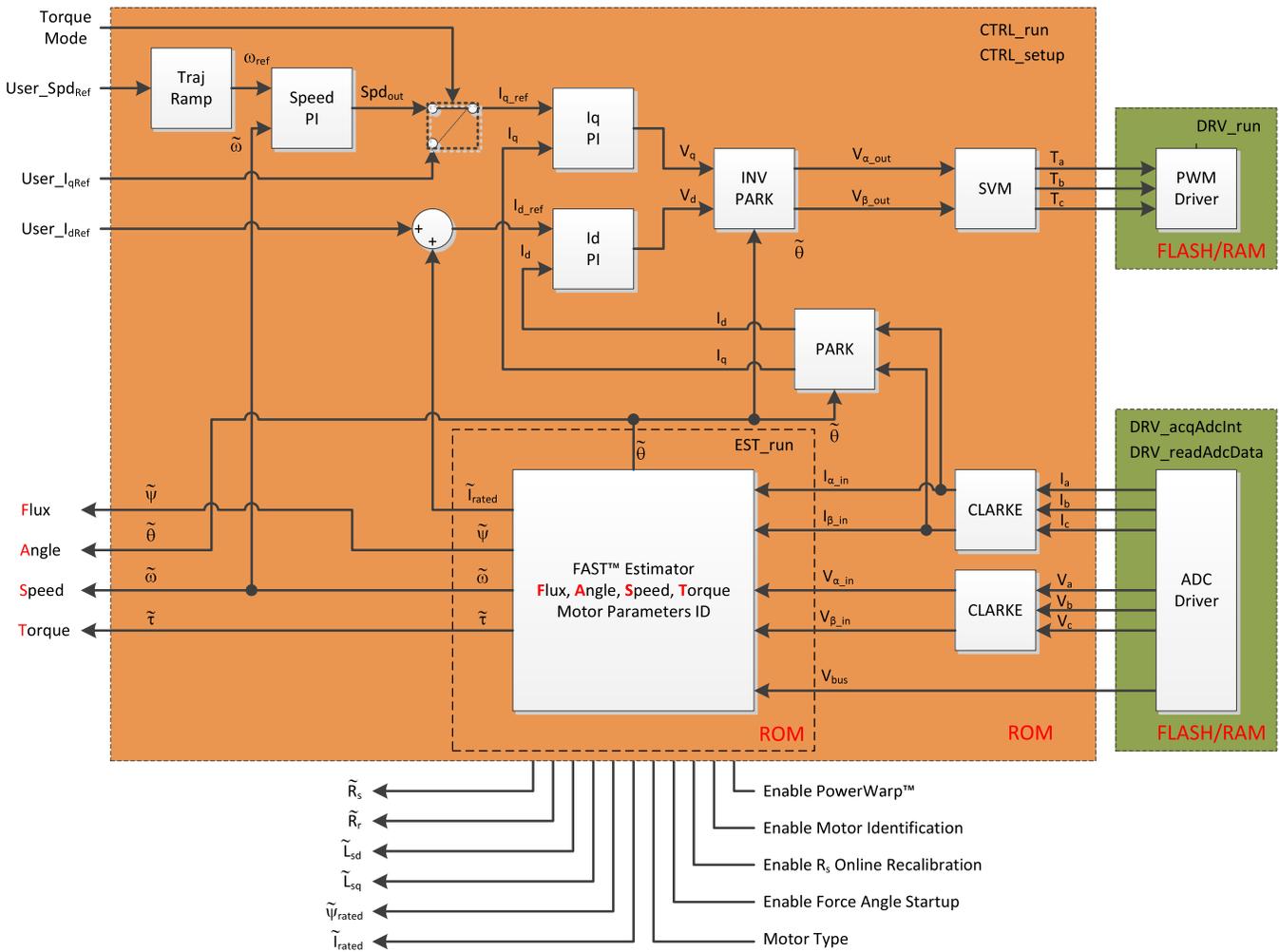
Figure 2. Block Diagram of Entire InstaSPIN-FOC Package in ROM

SPRUHI9A−February 2013−Revised January 2014   *TMS320F28069F, TMS320F28068F, TMS320F28062F InstaSPIN™-FOC*   7
Submit Documentation Feedback   *Software*

Copyright © 2013–2014, Texas Instruments Incorporated
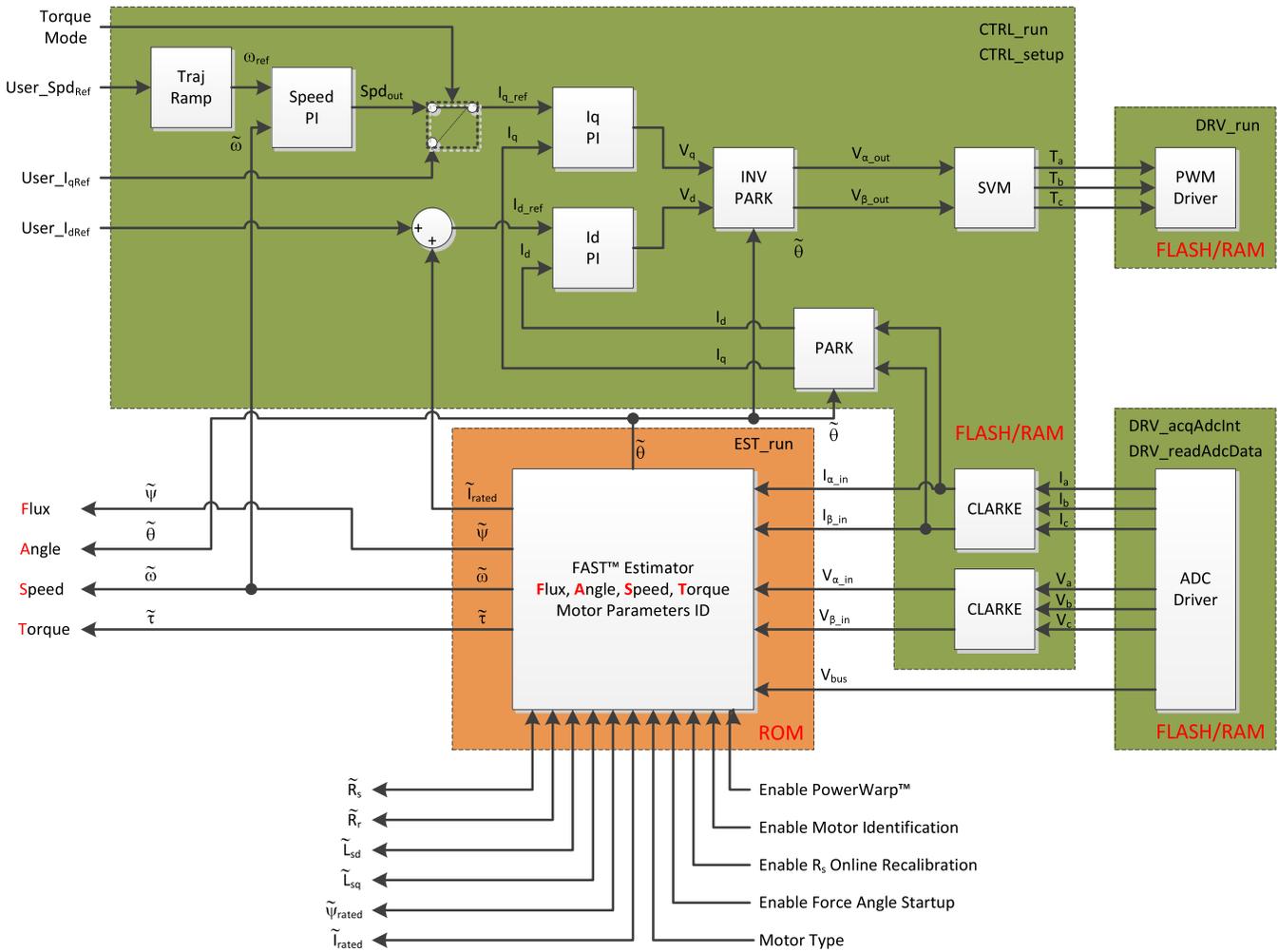
Figure 3. Block Diagram of InstaSPIN-FOC in User Memory, with Exception of FAST in ROM

5   Comparing FAST Estimator to Typical Solutions

Table 1 shows a comparison of the FAST estimator and InstaSPIN-FOC solution to typical software sensors and FOC solutions.

Table 1. FAST Estimator Compared to Typical Solutions

Topic	Typical Software Sensors and FOC Solutions	Fast Estimator and InstaSPIN-FOC Solution
Electrical Motor Parameters	Motor-model based observers heavily dependent on motor parameters.	Relies on fewer motor parameters. Off-line parameter identification of motor – no data sheet required. On-line parameter monitoring and re-estimation of stator resistance.
Estimator Tuning	Complex observer tuning, done multiple times for speed/loads, for each motor.	No estimator tuning required. Once motor parameters are identified, it works the same way every time, across speed/torque dynamics.
Estimator Accuracy	Angle-tracking performance is typically only good at over 5-10Hz with challenges at higher speeds and compensation for field weakening. Dynamic performance influenced by hand tuning of observer; Motor stalls typically crash observer.	FAST provides reliable angle tracking which converges within one electrical cycle of the applied waveform, and can track at less than 1 Hz frequency (dependent on quality and resolution of analog sensing). Angle tracking exhibits excellent transient response (even with sudden load transients which can stall the motor, thus enabling a controlled restart with full torque).
Start-up	Difficult or impossible to start from zero speed. Observer feedback at zero speed is not stable, resulting in poor rotor angle accuracy and speed feedback.	InstaSPIN-FOC includes: • Zero Speed start with forced-angle • 100% torque at start-up • FAST rotor flux angle tracking converges within one electrical cycle. FAST is completely stable through zero speed, providing accurate speed and angle estimation.
Current Loop	Tuning FOC current control is challenging – especially for novices.	Automatically sets the initial tuning of current controllers based on the parameters identified. User may update gains or use own controllers, if desired. The algorithm to fully tune the observer and torque controller takes less than 2 minutes.
Feedback Signals	System offsets and drifts are not managed.	FAST includes automatic hardware/software calibration and offset compensation. FAST requires 2-phase currents (3 for 100% and over-modulation), 3-phase voltages to support full dynamic performance, DCbus voltage for ripple compensation in current controllers. FAST includes an on-line stator resistance tracking algorithm.
Motor Types	Multiple techniques for multiple motors: standard back-EMF, Sliding Mode, Saliency tracking, induction flux estimators, or "mixed mode" observers.	FAST works with all 3-phase motor types, synchronous and asynchronous, regardless of load dynamics. Supports salient IPM motors with different Ls-d and Ls-q. Includes PowerWarp™ for induction motors = energy savings.
Field-Weakening	Field-weakening region challenging for observers - as the Back-EMF signals grow too large, tracking and stability effected.	FAST estimator allows easy field weakening or field boosting applications due to the stability of the flux estimation in a wide range, including field weakening region.
Motor Temperature	Angle tracking degrades with stator temperature changes.	Angle estimation accuracy is improved from online stator resistance recalibration.
Speed Estimation	Poor speed estimation causes efficiency losses in the FOC system and less stable dynamic operation.	High quality low noise Speed estimator, includes slip calculation for induction motors.
Torque Estimation	Torque and vibration sensors typically required.	High bandwidth motor Torque estimator.

6   FAST Provides Sensorless FOC Performance

6.1   FAST Estimator Replaces Mechanical Sensor

Field-oriented control (FOC) of an electric motor results in superior torque control, lower torque ripple, and in many cases, improved efficiency compared to traditional AC control techniques. For best dynamic response, rotor flux referenced control algorithms are preferred to stator flux referenced techniques. To function correctly, these systems need to know the spacial angle of the rotor flux with respect to a fixed point on the stator frame (typically the magnetic axis of the phase A stator coil). This has traditionally been accomplished by a mechanical sensor (for example, encoder or resolver) mounted to the shaft of the motor. These sensors provide excellent angle feedback, but inflict a heavy toll on the system design. There are six major system impacts resulting from sensored angle feedback, as discussed below and illustrated in Figure 4:

1. The sensor itself is very expensive (often over $2500 for a good resolver and several dollars for high volume integrated encoders).
2. The installation of the sensor requires skilled assembly, which increases labor costs.
3. The sensor often requires separate power supplies, which increases system costs and reduces reliability.
4. The sensor is the most delicate component of the system, which impacts system reliability, especially in harsh real-world applications.
5. The sensor feedback signals are brought back to the controller board via connectors, which also increases system costs and can significantly reduce reliability, depending on the type of connector.
6. The cabling required to bring the sensor signals back to the controller creates multiple challenges for the system designer:

    • Additional costs for the cable, especially if there is a substantial distance between the motor and controller.
    • Susceptibility to sources of noise, which requires adding expense to the cable with special shielding or twisted pairs.
    • The sensor and associated cabling must be earth grounded for safety reasons. This often adds additional cost to isolate these signals, especially if the processor which processes the sensor signals is not earth grounded.

In some applications where the motor is enclosed (for example, compressors), a sensored solution is impractical due to the cost of getting the feedback wires through the casing. For these reasons, designers of FOC systems are highly motivated to eliminate the sensor altogether, and obtain the rotor flux angle information by processing signals which are already available on the controller circuit board. For synchronous machines, most techniques involve executing software models of the motor being controlled to estimate the back-EMF waveforms (rotor flux), and then processing these sensed waveforms to extract an estimation of the rotor shaft angle, and a derivation of its speed. For asynchronous machines the process is a bit more complicated, as this software model (observer) must also account for the slip which exists between the rotor and rotor flux.

However, in both cases, performance suffers at lower speeds due to the amplitude of the back-EMF waveforms being directly proportional to the speed of the motor (assuming no flux weakening). As the back-EMF amplitude sinks into the noise floor, or if the ADC resolution cannot faithfully reproduce the small back-EMF signal, the angle estimation falls apart, and the motor drive performance suffers.

To solve the low-speed challenge, techniques have been created that rely on high frequency injection to measure the magnetic irregularities as a function of angle (that is, magnetic saliency) to allow accurate angle reconstruction down to zero speed. However, this introduces another set of control problems. First, the saliency signal is non-existent for asynchronous motors and very small for most synchronous machines (especially those with surface mount rotor magnets). For the motors that do exhibit a strong saliency signal (for example, IPM motors), the signal often shifts with respect to the rotor angle as a function of loading, which must be compensated. Finally, this angle measurement technique only works at lower speeds where the fundamental motor frequency does not interfere with the interrogation frequency. The control system has to create a mixed-control strategy, using high-frequency injection tracking at low speed, then move into Back-EMF based observers at nominal and high speeds.

With any technique, the process of producing a stable software sensor is also extremely challenging, as this motor model (observer) is essentially its own control system that needs to be tuned per motor across the range of use. This tuning must be done with a stable forward control loop. Needed is a stable torque (and usually speed) loop to tune the observer, but how do you pre-tune your forward control without a functioning observer? One option is to use a mechanical sensor for feedback to create stable current and speed loops, and then tune your software sensor in parallel to the mechanical sensor. However, the use of a mechanical sensor is often not practical. This problem has delayed market use of software sensors for sensorless FOC control.
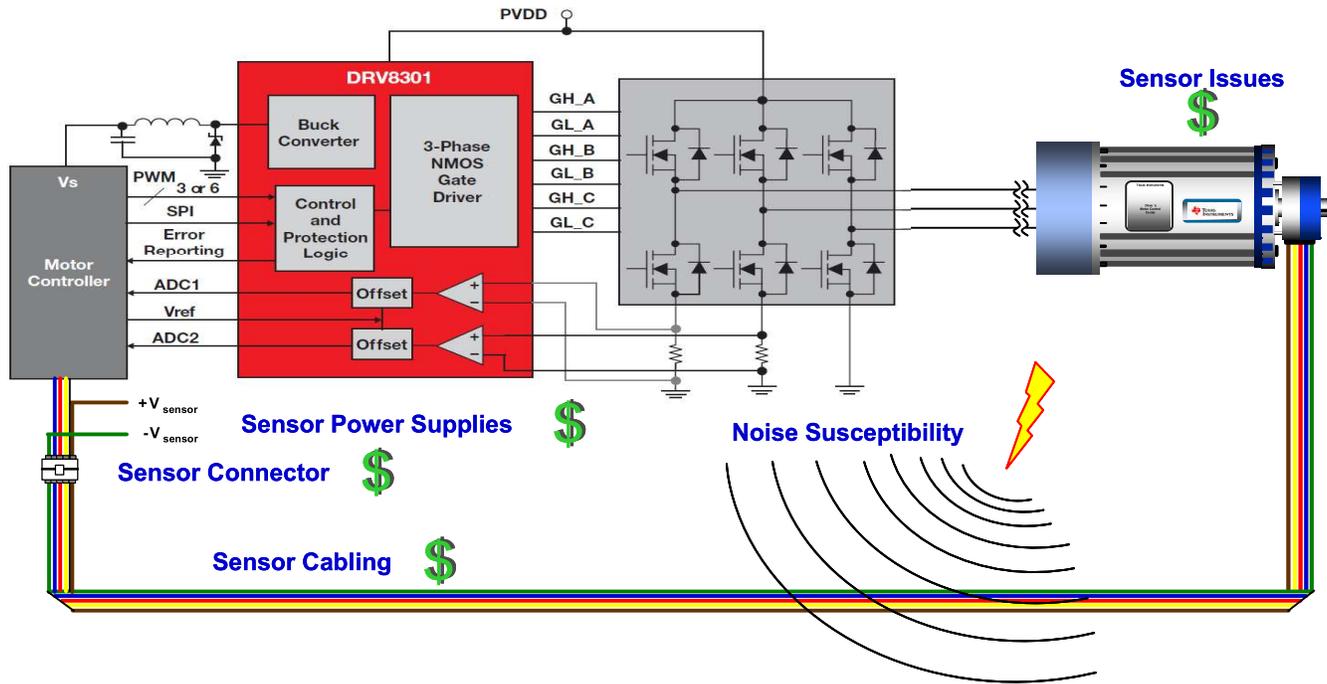

Figure 4. Sensored FOC System

In summary, these existing solutions all suffer from various maladies including:

- Poor low-speed performance (back-EMF and SMO)
- Poor high-speed performance (saliency observers)
- Poor dynamic response
- Calculation intensive (multi-modal observers)
- Parameter sensitivity
- Requirement for observer tuning.

The most recent innovation in the evolution of sensorless control is InstaSPIN-FOC. Available as a C-callable library embedded in on-chip ROM on several TI processors, InstaSPIN-FOC was created to solve all of these challenges, and more. It reduces system cost and development time, while improving performance of three-phase variable speed motor systems. This is achieved primarily through the replacement of mechanical sensors with the proprietary FAST estimator. FAST is an estimator that:

- Works efficiently with all three phase motors, taking into account the differences between synchronous/asynchronous, salient/non-salient, and permanent/non-permanent/induced magnets.
- Dramatically improves performance and stability across the entire operating frequency and load range for a variety of applications.
- Removes the manual tuning challenge of traditional FOC systems:
  – Qbservers and estimators, completely removes required tuning.
  – Current loop regulators, dramatically reduces required tuning.

- Eliminates or reduces motor parameter variation effects.
- Automatically designs a stable and functional control system for most motors in under two minutes.

6.2 Rotor Angle Accuracy Critical for Performance

Why has the need for a precise estimation of the rotor flux angle driven many to use mechanical sensors?

For efficient control of three-phase motors, the objective is to create a rotating flux vector on the stator aligned to an ideal orientation with respect to the rotor in such a way that the rotor field follows the stator field while creating necessary torque and using the minimum amount of current.

- **Stator**: stationary portion of the motor connected to the microprocessor-controlled inverter.
- **Ideal Orientation**: 90 degrees for non-salient synchronous; slightly more for salient machines, and slightly less in asynchronous machines since part of the current vector is also used to produce rotor flux.
- **Rotor**: rotating portion of the motor, produces torque on the shaft to do work.

To achieve this, you need to extract the following information from the motor:

- Current being consumed by each phase.
- Precise relative angle of the rotor flux magnetic field (usually within ± 3 electrical degrees), so you can orient your stator field correctly.
- For speed loops, you also need to know rotor speed.

6.3 Phase Currents Key to Estimator Accuracy

Resistor shunt current measurement is a very reasonable technique for measuring phase current in a motor control inverter. There are three widely used examples, the 1-, 2-, and 3-shunt resistor measurements. While at first the 1- and 2-shunt techniques seem to reduce cost, they require much faster and more expensive amplifier circuits. These 1- and 2-shunt current measurements also limit the capability of the current feedback which will limit the ability of the drive to use the full voltage that is provided to the inverter. The 3-shunt technique is superior and not much different in cost due to the advantage of using cheap slow current amplifier circuits. For best performance and cost with the FAST and InstaSPIN-FOC, the 3-shunt technique is recommended.

For more details, see the *TMS320F2806xF, TMS320F2802xF InstaSPIN-FOC/TMS320F2806xM InstaSPIN-MOTION User's Guide* (literature number SPRUHJ1).
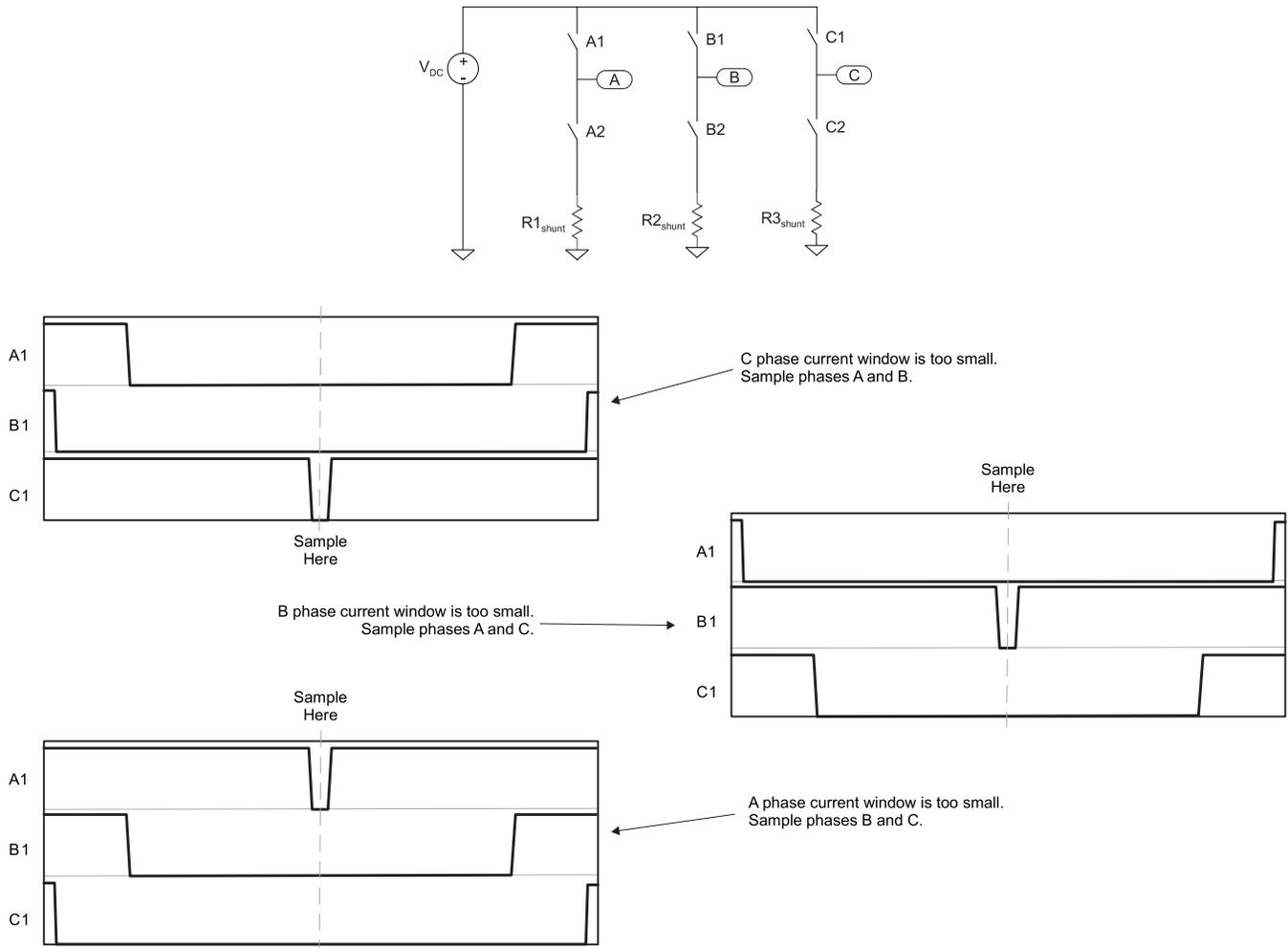
Figure 5. Inverter Using the 3-Shunt Current Sampling Technique

7   Evaluating FAST and InstaSPIN-FOC Performance

FAST and InstaSPIN-FOC performance data is being collected and will be provided in a future revision of this document.

8   Microcontroller Resources

The F2806xF microcontroller resources required by the InstaSPIN libraries are discussed in detail in the *TMS320F2806xF, TMS320F2802xF InstaSPIN-FOC/TMS320F2806xM InstaSPIN-MOTION User's Guide* (literature number SPRUHJ1).

Specifically for the library implementation and where the code is loaded and executed from, the following resources categories are discussed in this document:

*   CPU Utilization
*   Memory Allocation
*   Stack Utilization
*   Digital and Analog Pins Utilization

InstaSPIN-FOC provides flexibility throughout its design, including its software execution clock tree. Figure 6 illustrates the options available to the designer to manage the real-time scheduling of each of the major software functions. Balancing motor performance with CPU loading is not difficult, shortening system integration time.
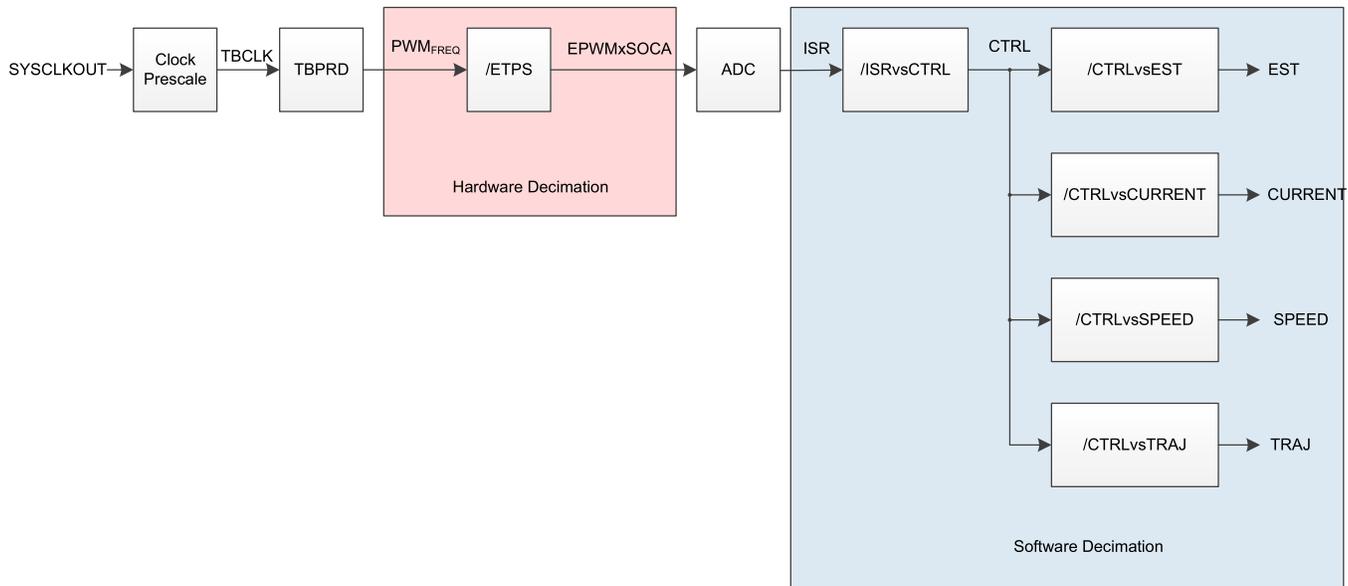
Figure 6. Software Execution Clock Tree Provides Flexibility with Real-Time Scheduling

Executing from single-cycle memory, total execution time for the full implementation of InstaSPIN-FOC will depend on the software execution clock tree.Table 2 shows the CPU cycles used when a full implementation of InstaSPIN is done, as well as users' code is loaded to FLASH. Note the impact of the software execution tree to total execution time. Table 3 shows the CPU loading and available MIPs for other system functions. The execution time does not change significantly from FULL to MIN implementations since the FAST block requires the largest number of CPU cycles and is in ROM for all implementations.

Table 2. CPU Cycles for FULL Implementation Executing from ROM and FLASH

	CPU Cycles			Executed From		
Function Name	**Min**	**Average**	**Max**	**ROM**	**RAM**	**FLASH**
DRV_acqAdcInt	25	25	25	×	×	✓
DRV_readAdcData	108	108	108	×	×	✓

Table 2. CPU Cycles for FULL Implementation Executing from ROM and FLASH (continued)

Function Name	CPU Cycles			Executed From		
	Min	Average	Max	ROM	RAM	FLASH
Ctrl_run				✓	×	×
R$_s$ Online Disabled, ISR vs CTRL = 1, CTRL vs EST = 1	2345	2355	2425			
CTRL vs EST = 2	1154	1760	2425			
CTRL vs EST = 3	1154	1562	2425			
ISR vs CTRL = 2, CTRL vs EST = 1	58	1207	2425			
CTRL vs EST = 2	58	909	2425			
CTRL vs EST = 3	58	810	2425			
ISR vs CTRL = 3, CTRL vs EST = 1	58	824	2425			
CTRL vs EST = 2	58	626	2425			
CTRL vs EST = 3	58	560	2425			
R$_s$ Online Enabled, ISR vs CTRL = 1, CTRL vs EST = 1	2807	2821	2894			
CTRL vs EST = 2	1154	1993	2894			
CTRL vs EST = 3	1154	1717	2894			
ISR vs CTRL = 2, CTRL vs EST = 1	58	1439	2894			
CTRL vs EST = 2	58	1025	2894			
CTRL vs EST = 3	58	887	2894			
ISR vs CTRL = 3, CTRL vs EST = 1	58	979	2894			
CTRL vs EST = 2	58	702	2894			
CTRL vs EST = 3	58	610	2894			
DRV_writePwmData	64	64	64	×	×	✓
CTRL_setup	37	51	178	✓	×	×

Table 3. CPU loading for FULL Implementation Executing from ROM and FLASH

2806xF CPU = 90 MHz Available MIPs = 90 MIPs PWM = 20 kHz	CPU Utilization [%]	MIPs Used [MIPS]	MIPS Available [MIPS]
R$_s$ Online Disabled, ISR vs CTRL = 1, CTRL vs EST = 1	57.71	51.94	38.06
CTRL vs EST = 2	44.49	40.04	49.96
CTRL vs EST = 3	40.09	36.08	53.92
ISR vs CTRL = 2, CTRL vs EST = 1	32.2	28.98	61.02
CTRL vs EST = 2	25.58	23.02	66.98
CTRL vs EST = 3	23.38	21.04	68.96
ISR vs CTRL = 3, CTRL vs EST = 1	23.69	21.32	68.68
CTRL vs EST = 2	19.29	17.36	72.64
CTRL vs EST = 3	17.82	16.04	73.96
R$_s$ Online Enabled, ISR vs CTRL = 1, CTRL vs EST = 1	68.07	61.26	28.74
CTRL vs EST = 2	49.67	44.7	45.3
CTRL vs EST = 3	43.53	39.18	50.82
ISR vs CTRL = 2, CTRL vs EST = 1	37.36	33.62	56.38
CTRL vs EST = 2	28.16	25.34	64.66
CTRL vs EST = 3	25.09	22.58	67.42
ISR vs CTRL = 3, CTRL vs EST = 1	27.13	24.42	65.58
CTRL vs EST = 2	20.98	18.88	71.12
CTRL vs EST = 3	18.93	17.04	72.96

Table 4. CPU loading for FULL Implementation Executing from ROM and FLASH

2806xF CPU = 90 MHz Available MIPs = 90 MIPs PWM = 20 kHz	CPU Utilization [%]	MIPs Used [MIPS]	MIPS Available [MIPS]
R$_s$ Online Disabled, ISR vs CTRL = 1, CTRL vs EST = 1	60.02	54.02	35.98
CTRL vs EST = 2	46.8	42.12	47.88
CTRL vs EST = 3	42.38	38.14	51.86
ISR vs CTRL = 2, CTRL vs EST = 1	33.49	30.14	59.86
CTRL vs EST = 2	26.87	24.18	65.82
CTRL vs EST = 3	24.67	22.2	67.8
ISR vs CTRL = 3, CTRL vs EST = 1	24.64	22.18	67.82
CTRL vs EST = 2	20.22	18.2	71.8
CTRL vs EST = 3	18.76	16.88	73.12
R$_s$ Online Enabled, ISR vs CTRL = 1, CTRL vs EST = 1	70.42	63.38	26.62
CTRL vs EST = 2	52	46.8	43.2
CTRL vs EST = 3	45.87	41.28	48.72
ISR vs CTRL = 2, CTRL vs EST = 1	38.69	34.82	55.18
CTRL vs EST = 2	29.47	26.52	63.48
CTRL vs EST = 3	26.4	23.76	66.24
ISR vs CTRL = 3, CTRL vs EST = 1	28.09	25.28	64.72
CTRL vs EST = 2	21.96	19.76	70.24
CTRL vs EST = 3	19.91	17.92	72.08

8.1 Memory Allocation and Utilization

Figure 7, Figure 8, and Table 5 show the memory map of the 28069, the location in ROM where the InstaSPIN-FOC library is located, and the required allocation of L8 RAM for the library to use. For a general memory map of these devices, see the device-specific data sheet.

Address	Data Space	Prog Space
0x00 0000	*M0 Vector RAM (Enabled if VMAP = 0)*	
0x00 0040	M0 SARAM (1K x 16, 0-Wait)	
0x00 0400	M1 SARAM (1K x 16, 0-Wait)	
0x00 0800	Peripheral Frame 0	Reserved
0x00 0D00	**PIE Vector - RAM (256 x 16) (Enabled if VMAP = 1, ENPIE = 1)**	
0x00 0E00	Peripheral Frame 0	
0x00 1400	CLA Registers	
0x00 1480	CLA-to-CPU Message RAM	
0x00 1500	CPU-to-CLA Message RAM	
0x00 1580	Reserved	
0x00 2000	Reserved	
0x00 4000	USB Control Registers(A)	Reserved
0x00 5000	Peripheral Frame 3 (4K x 16, Protected) DMA-Accessible	
0x00 6000	Peripheral Frame 1 (4K x 16, Protected)	
0x00 7000	Peripheral Frame 2 (4K x 16, Protected)	
0x00 8000	L0 DPSARAM (2K x 16) (0-Wait, Secure Zone + ECSL, CLA Data RAM2)	
0x00 8800	L1 DPSARAM (1K x 16) (0-Wait, Secure Zone + ECSL, CLA Data RAM 0)	
0x00 8C00	L2 DPSARAM (1K x 16) (0-Wait, Secure Zone + ECSL, CLA Data RAM 1)	
0x00 9000	L3 DPSARAM (4K x 16) (0-Wait, Secure Zone + ECSL, CLA Program RAM)	
0x00 A000	L4 SARAM (8K x 16) (0-Wait, Secure Zone + ECSL)	
0x00 C000	L5 DPSARAM (8K x 16) (0-Wait, DMA RAM 0)	
0x00 E000	L6 DPSARAM (8K x 16) (0-Wait, DMA RAM 1)	
0x01 0000	L7 DPSARAM (8K x 16) (0-Wait, DMA RAM 2)	
0x01 2000	L8 DPSARAM (8K x 16) (0-Wait, DMA RAM 3)	
0x01 4000	Reserved	
0x3D 7800	User OTP (1K x 16, Secure Zone + ECSL)	
0x3D 7BFA	Reserved	
0x3D 7C80	Calibration Data	
0x3D 7CC0	Get_mode function	
0x3D 7CD0	Reserved	
0x3D 7E80	PARTID	
	Calibration Data	
0x3D 7EB0	Reserved	
0x3D 8000	FLASH (128K x 16, 8 Sectors, Secure Zone + ECSL)	
0x3F 7FF8	128-Bit Password	
0x3F 8000	Boot ROM (32K x 16, 0-Wait)	
0x3F FFC0	*Vector (32 Vectors, Enabled if VMAP = 1)*	

Figure 7. 28069 Memory Map

Table 5. 2806xF Allocated Memory for InstaSPIN-FOC Library

Features	2806xF
Maximum Number of Motors that can be controlled	2
FAST Version	1.6
ROM Library [size, hex, words]	4000
ROM Library Start [address, hex]	3F 8000
Library Required RAM [size, hex, words]	800
Library Start RAM [address, hex]	01 3800

Figure 8 highlights the pieces of ROM EXE-only memory used by the libraries. EXE-only is execute only memory where read access is not possible.
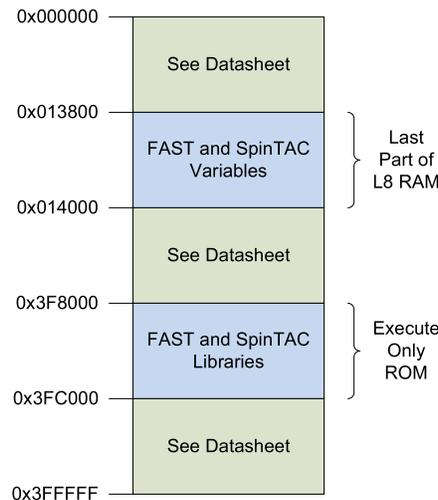

Figure 8. 2806xF Allocated Memory for InstaSPIN-FOC Library

Table 6 summarizes the memory used for the (4) most common configurations as shown in Figure 2 and Figure 3 (Full and Min implementations), with user memory optionally in FLASH or RAM. Note the code size increase as fewer functions in ROM are used.

Table 6. User Memory and Stack Sizes

Code Configurations		Memory Sizes (16bit Words)			Maximum Stack Used (16bit Words)
ROM Code	User Code	RAM	Flash	Total	
Full Implementation	RAM	0x1870	0x0000	**0x1870**	0x0120
Full Implementation	FLASH	0x001E	0x186C	**0x188A**	0x0120
Min Implementation	RAM	0x1F31	0x0000	**0x1F31**	0x0120
Min Implementation	FLASH	0x001E	0x1F2D	**0x1F4B**	0x0120

8.2 Pin Utilization

Flexibility in the design of InstaSPIN-FOC allows for multiple motors to be supported. Table 7 lists the minimum and maximum pins used per motor. Note that a F2806xF microcontroller provides (14) ePWM outputs with the 100-pin package, and (12) with the 80-pin.

Table 7. Pin Utilization Per Motor

Pin Type	Pin Name	Pins Usage Per Motor	
		Min	Max
Digital	PWM1A	3 (Requires External Fault and External Complementary Mode with Dead Time)	7
	PWM1B (Optional)		
	PWM2A		
	PWM2B (Optional)		
	PWM3A		
	PWM3B (Optional)		
	Trip Zone (Optional)		
Analog	IA	5 (Only two currents and no VBUS ripple compensation)	7
	IB		
	IC (Optional)		
	VA		
	VB		
	VC		
	VBUS (Optional)		

Appendix A Definition of Terms and Acronyms

ACIM — Alternating current induction motor.

CCStudio — Code Composer Studio.

FAST — Unified observer structure which exploits the similarities between all motors that use magnetic flux for energy transduction, automatically identifying required motor parameters and providing the following motor feedback signals:

- High-quality Flux signal for stable flux monitoring and field weakening.
- Superior rotor flux Angle estimation accuracy over wider speed range compared to traditional observer techniques independent of all rotor parameters for ACIM.
- Real-time low-noise motor shaft Speed signal.
- Accurate high bandwidth Torque signal for load monitoring and imbalance detection.

FOC — Field-oriented control.

Forced-Angle — Used for 100% torque at start-up until the FAST rotor flux angle tracker converges within first electrical cycle.

InstaSPIN-FOC — Complete sensorless FOC solution provided by TI on-chip in ROM on select devices (FAST observer, FOC, speed and current loops), efficiently controlling your motor without the use of any mechanical rotor sensors.

IPM — Interior permanent magnet motor.

Motor Parameters ID or Motor Identification — A feature added to InstaSPIN-FOC, providing a tool to the user so that there is no barrier between running a motor to its highest performance even though the motor parameters are unknown.

PI — Proportional-integral regulator.

PMSM — Permanent magnet synchronous motor.

PowerWarp™ — Mode of operation used for AC induction motors (ACIM) that allows minimum current consumption.

Rs-Offline Recalibration — InstaSPIN-FOC feature that is used to recalibrate the stator resistance, Rs, when the motor is not running.

Rs-Online Recalibration — InstaSPIN-FOC feature that is used to recalibrate the stator resistance, Rs, while the motor is running in closed loop.

SVM — Space-vector modulation.

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		