THESIS APPROVAL

GRADUATE SCHOOL, KASETSART UNIVERSITY

Master of Engineering (Information and Communication Technology for Embedded Systems)

DEGREE

Information and Communication Technology for Embedded Systems — **FIELD**

Electrical Engineering — **DEPARTMENT**

TITLE: Position Feedforward and Friction Feedforward Compensation on CNC Milling Machines

NAME: Mr. Wijin Suteeratanapirom

THIS THESIS HAS BEEN ACCEPTED BY

.... **THESIS ADVISOR**

(Assistant Professor Peerayot Sanposh, D.Sc.)

.... **THESIS CO-ADVISOR**

(Mr. Nattapon Chayophitak, Ph.D.)

.... **THESIS CO-ADVISOR**

(Professor Hiroaki Kunieda, Ph.D.)

.... **DEPARTMENT HEAD**

(Assistant Professor Srijidtra Charoenlarpnopparut, Ph.D.)

APPROVED BY THE GRADUATE SCHOOL ON ....

.... **DEAN**

(Associate Professor Gunjana Theeragool, D.Agr.)

THESIS

POSITION FEEDFORWARD AND FRICTION FEEDFORWARD COMPENSATION ON CNC MILLING MACHINES

WIJIN SUTEERATANAPIROM

A Thesis Submitted in Partial Fulfillment of
the Requirements for the Degree of
Master of Engineering (Electrical Engineering)
Graduate School, Kasetsart University
2010

Wijin Suteeratanapirom 2010: Position Feedforward and Friction Feedforward Compensation on CNC Milling Machines. Master of Engineering (Information and Communication Technology for Embedded Systems), Major Field: Information and Communication Technology for Embedded Systems, Department of Electrical Engineering.
Thesis Advisor: Assistant Professor Peerayot Sanposh, D.Sc. 78 pages.

In CNC milling machine position control, accuracy is importance because most of work pieces are used for prototype work such as mold using with plastic injection machine. CNC milling machines have many mechanical components such as motors, bearings, ball screws, linear guideways, and etc. It's sure that the friction effect is a main problem that affect to position control.

In this research, the friction effect problem is solved using both friction feedforward compensation and position feedforward controller. To use friction feedforward compensation technique, friction model and its all parameters must be known. The selected friction model is Armstrong's model. Simulation using MATLAB with Simulink is illustrated. For implementation, NI-Data Acquisition Card: PCI-6251, and MATLAB with Simulink is used. Moreover, a prototype control board using a high-performance Freescale DSP, MC56F8345, is used to control the motor position by 2-phase pulse counting. INEX experimental board, JX2148, which use ARM microcontroller, LPC2148, is used to measure position of motor. The program on DSP and ARM microcontroller is developed by using C language.

____________________ ____________________ ___ / ___ / ___

Student's signature Thesis Advisor's signature

ACKNOWLEDGEMENTS

I would sincerely like to gratefully thank, and I am deeply indebted to Asst. Prof. Dr. Peerayot Sanposh who is my thesis advisor for partially financial support, advice, encouragement, valuable suggestion for everything.

I would like to thank Dr.Nattapon Chayopitak who is my thesis co-advisor in NECTEC for advice, valuable suggestion and review my English written language in both of public papers and thesis.

I would like to thank Prof. Hiroaki Kunieda who is also my thesis co-advisor in Japan for his valuable comments and suggestion.

I would like to thank my friends in IMARC laboratory: Mr.Thitiwat Therdbankerd, Miss.Chutima Laosapan, Mr.Rapeepong Rattanawaorahirankul, Mr.Surathep Nilnond and other ICTES friends. Especially, Mr.Noppadon Katepananon who advise technical for programming and Mr.Chaiwat Suwanasaroj who persuade me to drink beer every Thursday.

This research was financially supported by Thailand Advanced Institute of Science and Technology – Tokyo Institute of Technology (TAIST-Tokyo Tech), National Science and Technology Development Agency (NSTDA), Tokyo Institute of Technology (Tokyo Tech), and Kasetsart University (KU).

Finally, I am especially appreciated my parents, my sister and brother for their continuous encouragements and support.

Wijin Suteeratanapirom

June 2010

TABLE OF CONTENTS

Page

LIST OF TABLES

Table **Page**

Appendix Table

LIST OF FIGURES

LIST OF FIGURES (Continued)

Figure **Page**

LIST OF FIGURES (Continued)

Figure **Page**

Appendix Figure

LIST OF ABBREVIATIONS

ADC	=	Analog to Digital Convertor
CNC	=	Computer Numerical Control
DAC	=	Digital to Analog Convertor
DSP	=	Digital Signal Processor
FFC	=	Friction Feedforward Compensation
m	=	Meter
N	=	Newton
ω	=	Omega
PFC	=	Position Feedforward Controller
rad	=	Radian
RPM	=	Revolution Per Minute
s	=	Second

POSITION FEEDFORWARD AND FRICTION FEEDFORWARD COMPENSATION ON CNC MILLING MACHINES

INTRODUCTION

CNC milling machine has the friction problem which affects performance and precision. Friction is in each mechanical component of CNC milling machine such as motor, linear guide, and bearing. This friction effect decrease precision of X-Y table on CNC milling machine. Due to this problem, this research is focused on studying friction effect and identifying friction model in the system. Friction models are discussed by (Erkorkmaz and Yusuf, 2001), and (Olsson *et al.,* 1998). One of interesting friction models is Armstrong's model, which is illustrated by (Olsson *et al.,* 1998) and identified by (Rafael and Jeśus, 1999). Motor and driver block diagram used in this research is described by T. Yamazaki (2003) and T. Mcintosh (2004). Block diagrams of both friction model and motor with driver model are considered for designing a friction compensator. Many researchers use different techniques to compensate friction such as (Lee and Ahn, 2008), (Lee and Kim, 1995),and (Lee and Tomizuka, 1996).

This research proposes a controller designing method using the Position Feedforward Controller (PFC) with P-Controller and Friction Feedforward Compensation with PI-Controller. The simulation yields satisfactory result. For the experiment in this research, MSMA082A1C Panasonic motor with MSDA083A1A motor drive which is set to the torque control mode is connected to a PC via a PCI-6251 NI data acquisition card. The experimental result is similar to the simulation one. The result of using PFC and FFC in torque control mode is better than using PFC and FFC in velocity control mode.

OBJECTIVES

1. To improve tracking performance of CNC milling machine using position feedforward controller with friction feedforward compensation.
2. To identify friction model using Armstrong's model.
3. To build the circuit board that can be used to control the CNC milling machine.

LITERATURE REVIEW

To design a CNC milling machine, position control is one of importance thing to be considered. Accuracy of position control depends on many parameters such as friction of mechanical part, controller type, and precision of measurement tools. There are many friction types found by many researchers. The friction models can be divided into two main categories that are dynamic models and static models.

Olsson *et al*., (1998) presented description of many friction models that include both dynamic and static friction models. The example of friction compensation is shown in this research by simulation and experimentation.

Erkorkmaz and Yusuf (2001) presented the designing method of high speed CNC system that the identification method of friction characteristic of machine tool drives is described. Moreover, this research is focused on the friction model for lubricated metallic surfaces in contact.

Friction model identification is a problem for designing the CNC milling machine. To identify the friction, high technology tools have to be used. However it has an easy way to identify it.

Rafael and Jeśus (1999) presented friction identification method using torque ramp input and determine speed of motors. A friction model of this research is a static friction, Armstrong’s model, which includes all friction parameters.

To eliminate friction effect which is a problem of position control, friction feedforward compensation technique is considered. This technique has to know the friction model of a plant to be controlled.

Tsai *et al.*, (2004) presented the friction compensation technique for feed drive system using Kanopp friction model for obtaining a transfer function to compensate friction in the system.

MATERIALS AND METHODS

Materials

1. Personal Computer (PC), Intel Core2 duo processor
2. MATLAB Simulation Software
3. CNC Milling Machine
4. Panasonic Motor, MSMA082A1C
5. Panasonic Motor Driver, MSDA083A1A
6. DSP Circuit Board using MC56F8345 digital signal processor
7. ARM7 Circuit Board, JX-2148, INEX's Product
8. Switching Adaptor 100-240VAC to +/-12VDC 2.0A and +5VDC 3.0A
9. Switching Adaptor 100-240VAC to +12VDC 3.0A
10. Switching Adaptor 240VAC to +9VDC 1.2A
11. Oscilloscope
12. RS-232 Cable
13. USB Cable A to B Type

Methods

This thesis has both simulation and implementation which use PCI-6215 and DSP circuit board to control a servo motor. The simulation use motor driver setting both velocity and torque control mode to compare performance of controller which can be divided into six simulations. The motor drive is set into two modes: velocity control mode and torque control mode.

Velocity Control Mode

1. P-Controller
2. Friction Feedforward Compensation (FFC) with P-Controller
3. Position Feedforward Controller (PFC) and Friction Feedforward Compensation (FFC) with PI-Controller

Torque Control Mode

1. PI-Controller for velocity loop and P-Controller for position loop
2. Friction Feedforward Compensation (FFC) with PI-Controller for velocity loop and P-Controller for position loop
3. Friction Feedforward Compensation (FFC) with PI-Controller for velocity loop and Position Feedforward Controller (PFC) with P-Controller for position loop

After simulation in both velocity and torque control mode, The simulation with best result is selected to implement on PCI-6251, Data Acquisition Card. After that, It is implemented on DSP circuit board which is programmed by using C-language.

Friction Model and Friction Identification

Friction force is a resistance force that it's direction is opposite the normal force be applied to an moved object. Friction force have both advantage and disadvantage, example of advantage are the break on a car, grooves on a screw top bottle, and etc. In producing CNC milling machines, Friction force is a disadvantage such as friction at bearings, ball-screws, linear guideways and etc. The friction model has many type which each model different. For this research, It use Armstrong's model that will be described in next section.

Friction Model

Friction model in this research use Armstrong's model. The Friction model can be divided into 4 components: static friction, coulomb friction, viscous friction, and Stribeck effect. The friction model is shown in Figure 1.

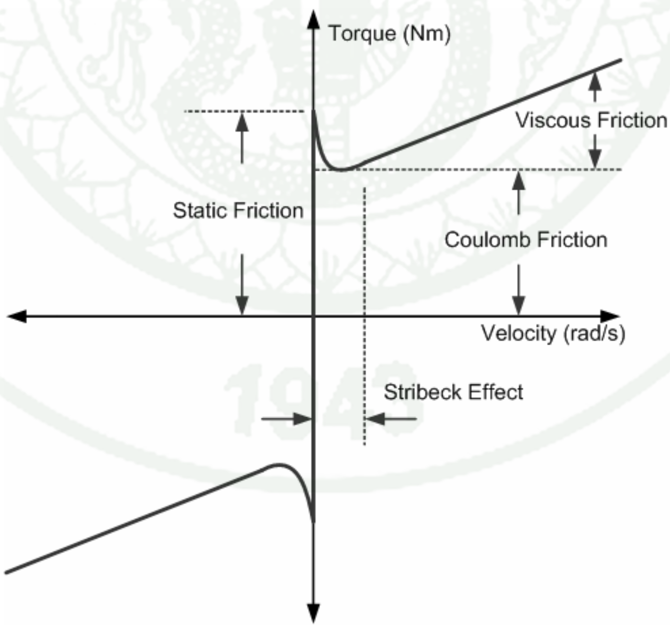


Figure 1 Friction model.

1. Static Friction

The static friction occurs when the object is stationary. If the external force is greater than this friction, the object will be moved. This friction can be obtained as

$$T = \begin{cases} T_e & \text{if } \omega{=}0 \text{ and } T_e < T_s \\ T_s sgn(T_e) & \text{if } \omega{=}0 \text{ and } T_e \geq T_s \end{cases} \quad (1)$$

Where T_e = external force

T_s = maximum magnitude of static friction,

2. Coulomb Friction

The coulomb friction or dry friction is the opposite force to object motion that has a constant magnitude. Coulomb friction, T_c, is a function of force which is represented as

$$T_c = \mu T_N \quad (2)$$

where T_c = coulomb friction

μ = coulomb friction coefficient

T_N = normal force.

3. Viscous Friction

Viscous friction is a linear force function which proportional to velocity. This friction is function of speed which is represented as

$$T = T_v \omega \quad (3)$$

where T_v = the viscous friction coefficient

ω = the velocity.

4. Stribeck Effect

The Stribeck effect is occurred when the object begins moving as shown in

Figure 1. Stribeck effect is a non-linear function that can be identified by experimentation. The lowest speed at bottom of curve is called Stribeck velocity.

The friction model used in this research is approximated by using Armstrong's model that is shown in friction model identification section. The Armstrong's model is given by

$$T(\omega) = T_v\omega + T_c sgn(\omega) + [T_s - T_c] sgn(\omega) e^{-(\frac{\omega}{v_s})^2} \tag{4}$$

where T_v = viscous friction coefficient

T_c = coulomb friction coefficient

T_s = maximum magnitude of static friction

v_s = stribeck velocity coefficient

ω = angular velocity.

Friction Identification

The friction model in this research is the Armstrong's model, which is a static model. This model has all important parameters, i.e., static friction, coulomb friction, viscous friction, and the Stribeck effect. The model parameters can be identified by generating a torque ramp input to the motor and then measuring the speed. Considering the friction model is used with a motor model, the dynamic equations of a motor are described by

$$\begin{bmatrix} \frac{d\theta}{dt} \\ \frac{d\omega}{dt} \end{bmatrix} = \begin{bmatrix} \omega \\ \frac{1}{J}\left[mt - T_v\omega - T_c sgn(\omega) - [T_s - T_c] sgn(\omega) e^{-(\frac{\omega}{v_s})^2}\right] \end{bmatrix} \tag{5}$$

where m is the slope of the torque ramp input, and J is the motor reflexive inertia. The relationship between the velocity and the time in (5) is shown in Figure 2. The friction parameters can be obtained by (Rafael and Jeśus, 1999).

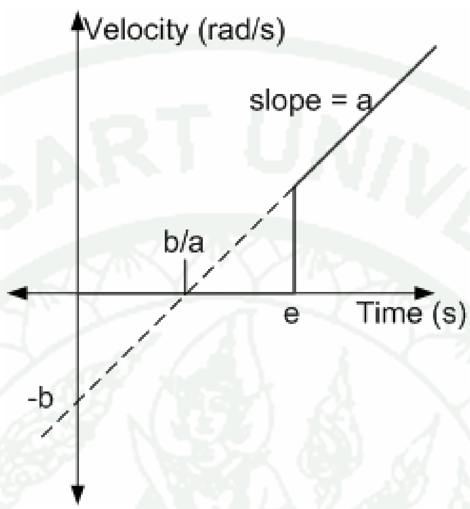


Figure 2 Armstrong's model graph.

$$T_v = \frac{m}{a} \tag{6}$$

$$T_c = \frac{b}{a}m \tag{7}$$

where a = slope of velocity

b = the point that output velocity line cross to the velocity axis

Tracking Performance Improvement

Fricion Feedforward Compensation

The goal of designing friction feedforward compensator is to reduce tracking error or disturbance of the X-Y table position control. Figure 3 shows the diagram of using a feedforward compensator to the reduce tracking error, where $G_{mv}(s)$ is the transfer function of the mechanical component of the motor, $C_v(s)$ is the transfer function of velocity controller which is PI-controller.

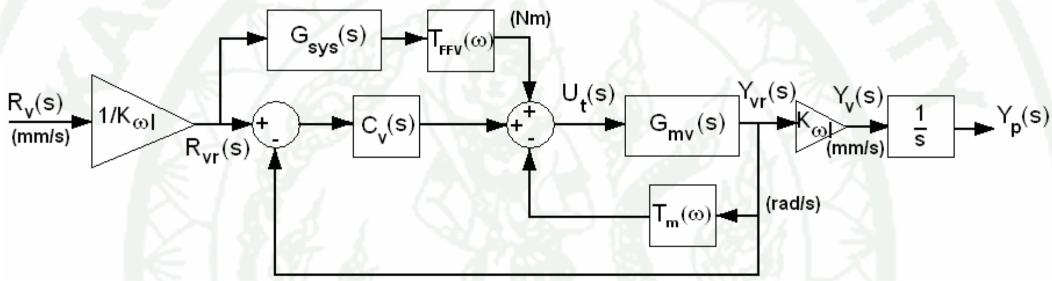


Figure 3 Friction feedforward compensation.

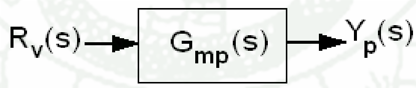


Figure 4 Reduced block diagram of feedforward friction compensation.

The overall transfer function of the represented system in Figure 3, is

$$G_{mp}(s) = \frac{Y_p(s)}{R_v(s)} \tag{8}$$

and the transfer function of mechanical part of motor is

$$G_{mv} = \frac{1}{Js(T_{tf}s+1)} \tag{9}$$

where, T_{tf} = time constant of torque filter in motor driver

J = moment of inertia of the motor's rotor.

To create the velocity perfect tracking, the transfer function must be equal to one, i.e.,

$$T_{vr}(s) = \frac{Y_{vr}(s)}{R_{vr}(s)} = 1 \tag{10}$$

Therefore, the friction feedforward transfer function, $T_{FFV}(\omega)$ must be the same as the plant friction, $T_m(\omega)$. Note that

$$G_{sys}(s) = \frac{C_v(s)G_{mv}(s)}{1+C_v(s)G_{mv}(s)} \tag{11}$$

is the system closed-loop transfer function when the feedforward path is not applied, and the plant friction is not included.

Position Feedforward Controller

Feedforward controller is usually used to improve tracking performance if the plant model is well understood. Figure 5 shows the block diagram of using position feedforward controller.

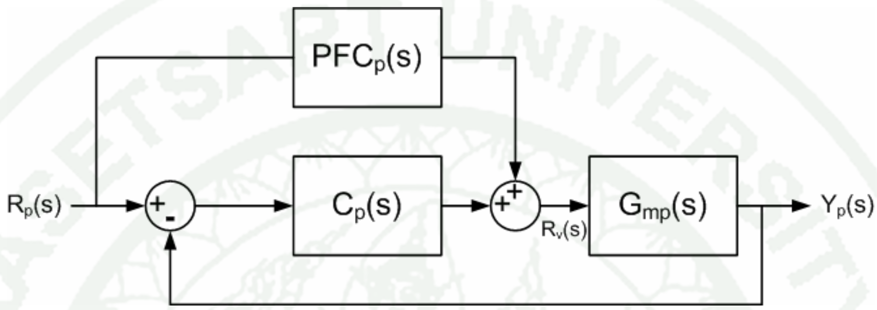


Figure 5 Position Feedforward Controller.

To achieve perfect position tracking, the position controller, $C_p(s)$, which is P-controller, and position feedforward controller, $PFC_p(s)$, is considered. The transfer function of Figure 5 is described in (12). The output of the system is position in *mm* unit. The transfer function of the position feedforward controller can be obtained as

$$T_p(s) = \frac{Y_p(s)}{R_p(s)} = \frac{PFC_p(s)G_{mp}(s) + C_p(s)G_{mp}(s)}{1 + C_p(s)G_{mp}(s)} \quad (12)$$

To create perfect tracking control, $T_p(s) = 1$, then

$$PFC_p(s) = \frac{1}{G_{mp}(s)} \quad (13)$$

Identification of Plant Friction

To identify the friction model on X-Y CNC table, coulomb friction and viscous friction model are considered for the Panasonic motor and its driver in Figure 6. The experiment of identification of the plant friction is conducted by using the motor driver in the torque control mode and applying a torque ramp with the slope of *m*, equal to 0.5, for 1 second to the motor drive. After that, the speed and the time data are captured. The data is plotted, where the x-axis is time *(s)* and the y-axis is velocity *(μm/s)* as shown in Figure 7.

The friction model parameters can be obtained by using fitting curve by the least square technique. From Figure 7, the captured data from 0.5 to 1 second is considered to identify the friction parameters, and the Curve Fitting Tool in MATLAB is used to fit the curve.

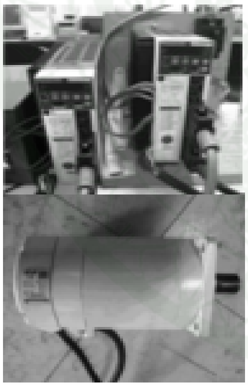


Figure 6 Motor, Driver, and X-Y table.

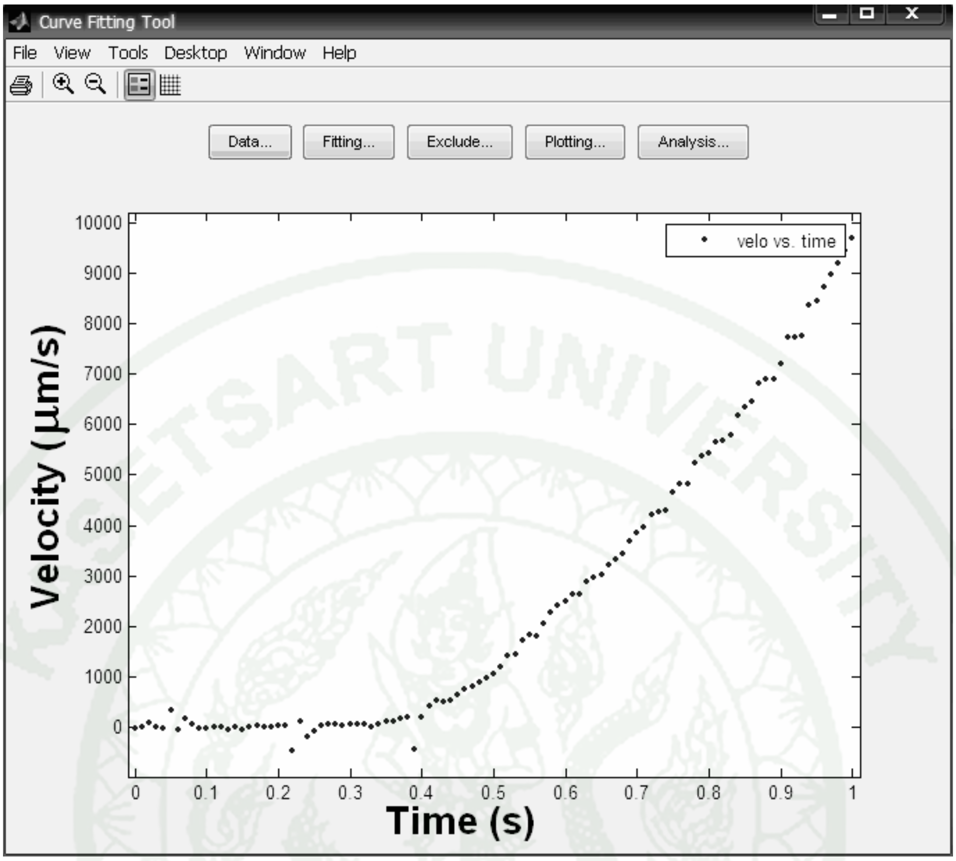


Figure 7 Velocity and time relationship for torque ramp slope 0.5.

The parameters as shown in Figure 2 are *a* = 16,670, *b* = 7,634 and *e* = 0.5. Then,

$$T_v = \frac{m}{a} = \frac{0.5}{16,670} = 2.994 \times 10^{-5}(Nm \cdot s/deg) = 0.0017(Nm \cdot s/rad)$$

$$T_c = \frac{b}{a}m = \frac{7,634}{16,670}(0.5) = 0.229,$$

where

T_v = viscous friction parameter

T_c = coulomb friction parameter.

From (4), the static friction parameter, T_s, cannot be obtained using this technique. However, this parameter can be achieved by trial applying constant torques to the motor from zero and increase it until the motor begin to rotate. The just-started torque is the static friction parameter which the parameter from the experiment is 0.3145 *Nm*. The real friction can be obtained as

$$T(\omega) = 0.0017\omega + 0.229sgn(\omega) + 0.0855sgn(\omega)0.5^{(\omega/v_s)^2} \tag{14}$$

Simulation

The simulation of motor control used in the CNC milling machine shown in this section is processed on simulation software, MATLAB and Simulink. The motor parameters used in this simulation are Permanent Magnet Synchronous Motor (PMSM), MSMA082A1C and the motor driver is MSDA083A1A. Both of devices are developed by Panasonic. They can be called servo motor which include motor and its driver. The servo motor model can be found by using the motor and driver parameters supported by the Panasonic's manual and servo motor modeling technique by Yamazaki (2003). The motor driver can be set the control mode to either velocity or torque control mode. The simulations are divided into two categories, i.e. velocity control mode and torque control mode.

Velocity Control Mode

1. Motor control with P-Controller
2. Motor control with FFC and P-Controller
3. Motor control with PFC, FFC and P-Controller

Torque Control Mode

1. Motor control with P-Controller for position loop and PI-Controller for velocity loop
2. Motor control with P-Controller and FFC with PI-Controller
3. Motor control using PFC with P-Controller and FFC with PI-Controller

Velocity Control Mode

Before describe the detail of each simulations, the plant model or motor with driver model is described. The motor with driver model in velocity mode is shown in Figure 8.

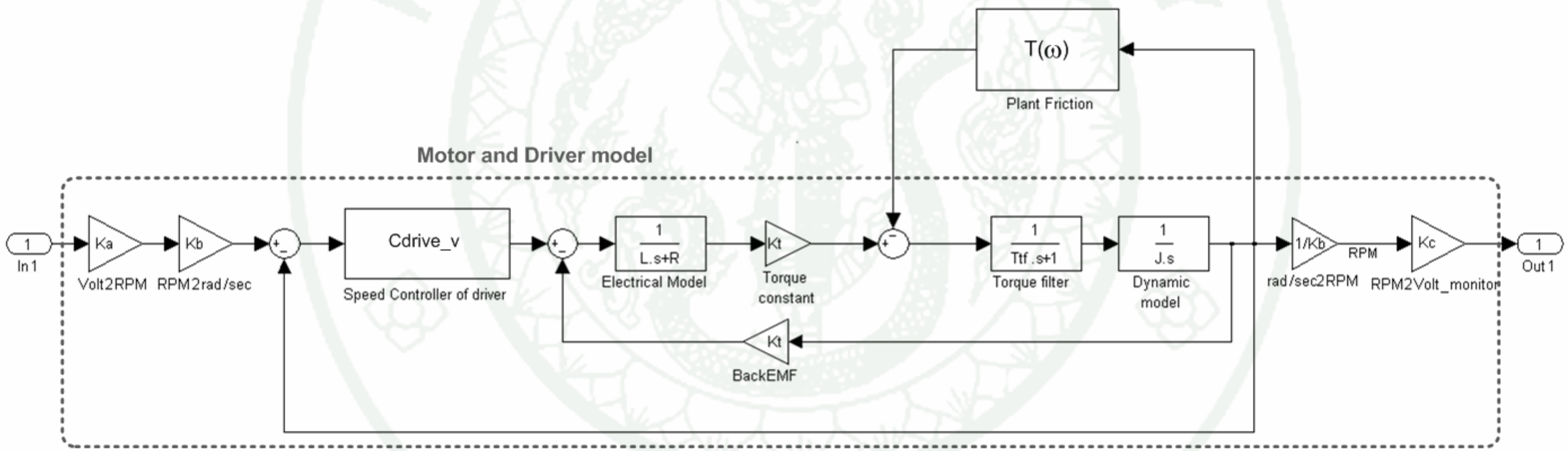


Figure 8 The model of motor and its driver set in velocity control mode with plant friction

From Figure 8, the input of the block is the voltage used to control the motor and the output is the voltage referred to the speed measurement value of the motor. The input voltage is converted to velocity in revolution per minute (RPM) unit by gain K_a that can be adjusted on the motor driver and, then converted to radian per second (rad/s) unit by the gain K_b. The velocity error, the input velocity or desired velocity is subtracted by the output velocity, is an input of internal speed controller of the motor driver, which has the model shown in Figure 9, where K_v is proportional parameter and T_v is time constant of integral part of the velocity controller. The electrical part of the motor model composes of the inductance, L, and resistance, R. The output of this electrical block is the current that is converted to torque by motor torque constant K_t. The torque generated by the electrical part is subtracted by the friction torque shown in Figure 10. The subtracted torque is the input to the torque filter of the motor driver. The dynamic block diagram of the motor composed of moment of inertia, J, is used to convert from torque to velocity. Moreover, the friction of the motor and linear guide composed of viscous friction, K_{vc}, and coulomb friction, K_{cl}, is considered. The parameters of motor, motor driver, and friction model are listed in Table 1.

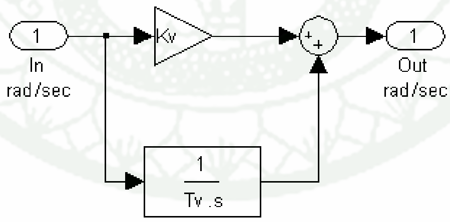


Figure 9 The model of internal speed controller of the motor driver, $Cdrive\_v$

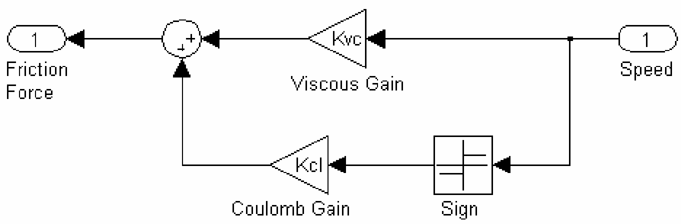


Figure 10 The model of plant friction, $T(\omega)$

Table 1 The parameters of motor, motor driver, and friction model.

Detail	Symbol	Value	Unit
Unit conversion from Volt to RPM	K_a	1000	*RPM/Volt*
Unit conversion from RPM to rad/sec	K_b	0.1047	*rad/s/RPM*
Unit conversion from RPM to Monitor Voltage	K_c	500	*Volt/RPM*
Motor Inductance	L	0	*Henry*
Motor Resistance	R	0.55	*Ohms*
Moment of Inertia of Motor	J	0.000131	*Nm/rad/s*^{2}
Torque Constant and Back EMF	K_t	0.558	*Nm/A and Volt/rad/s*
Time Constant of Torque Filter	T_{tf}	0.00025	*s*
Proportional Gain of Velocity Controller	K_v	0.178486	*Nm/rad/s*
Time Constant of Velocity Controller	T_v	0.02	*s*
Viscous Friction	K_{vc}	0.0017	*Nm/rad/s*
Coulomb Friction	K_{cl}	0.229	*Nm*

1. Motor Control with P-Controller

The motor control with P-Controller is sufficient for position control. From Figure 13, the velocity reference is converted to position reference by integrating the velocity reference. In this simulation, the reference position is in both forward and backward directions in order to observe position error at near zero. The trapezoid velocity profile shown in Figure 11 is used.

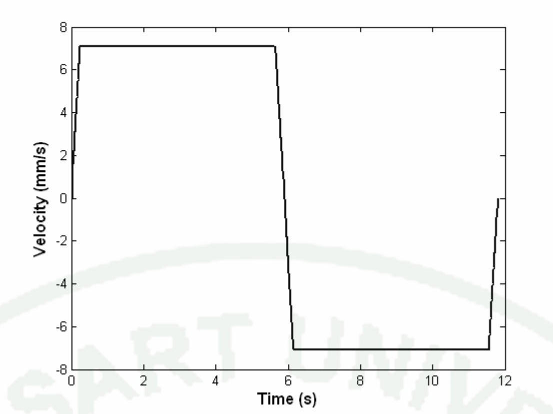


Figure 11 The velocity reference for position control simulation with P-controller

After pass the velocity reference through the integral block, the position reference shown in Figure 12 is obtained.

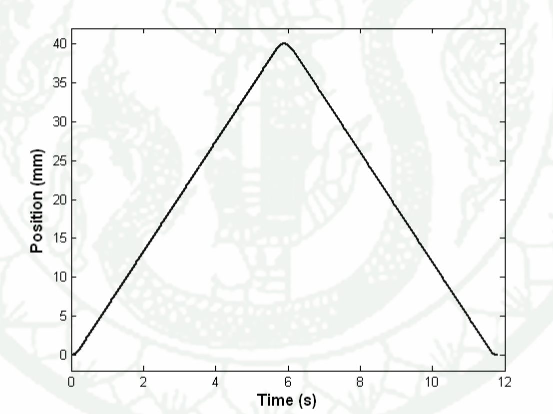


Figure 12 The velocity reference for position control simulation with P-controller

The block diagram of the position control with P-Controller shown in Figure 13 uses proportional parameter K_{pp}, of 100. Input of the simulation is velocity but it must to be changed to position by integration for position control. The gain RPM2Volt, $1/K_a$, can be adjusted on the motor driver. The linear guide has transmission ratio 6 mm per revolution, then the parameter, K_{lg}, used to convert from RPM to *mm/s* unit is 0.1.

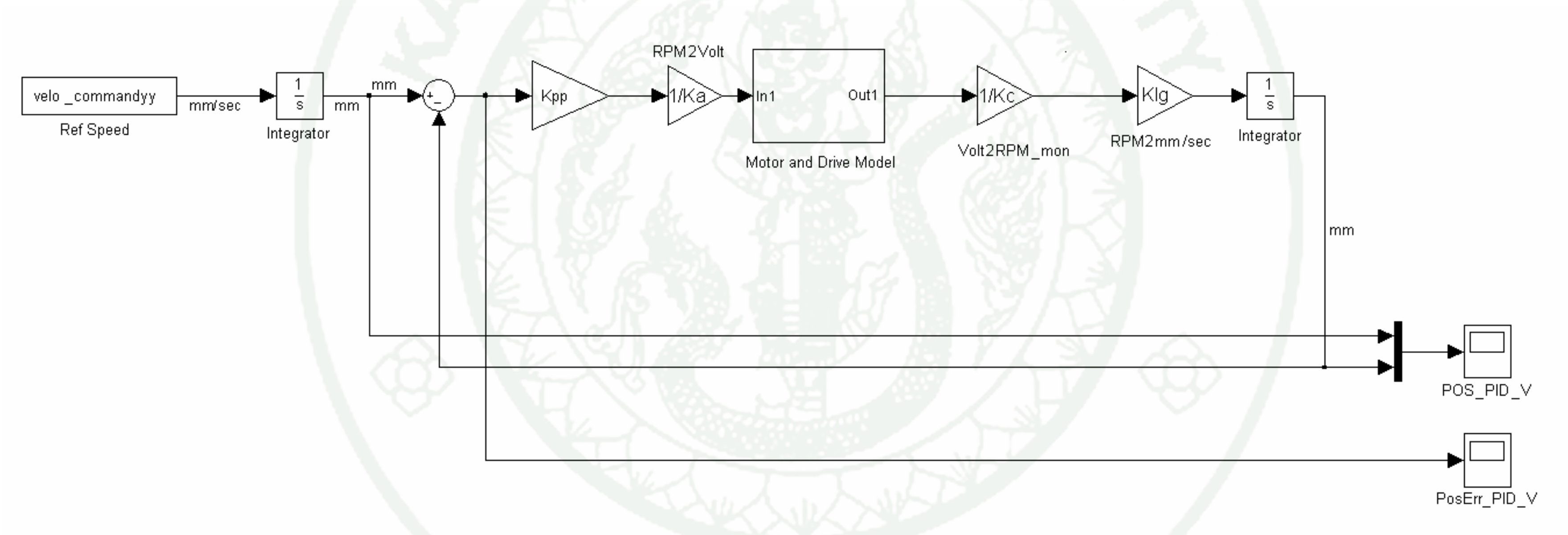


Figure 13 The block diagram of motor control system with P-Controller

2. Motor Control with FFC and P-Controller

In this section, the simulation of motor control with Friction Feedforward Compensataion (FFC) and P-Controller is illustrated. The block diagram of this simulation is shown in Figure 14. Adding the friction force to compensate the friction of the plant, which includes motor, motor driver and linear guide, cannot be done directly because the motor driver is set in velocity control mode. To compensate friction of the plant, the simple moving block diagram technique must be applied. The final block diagram of system with FFC and P-Controller is shown in Figure 14. The friction model of this block diagram is same the friction model of motor and driver in Figure 10. The block G_{sys} is close-loop transfer function of P-Controller and motor with driver model when FFC is not applied. TF_e is the inverse transfer function of electrical part of motor model and TF_s is the inverse transfer function of the speed controller of the driver model. This simulation uses the same velocity profile as the first simulation. The result is better than the control with only P-Controller as shown in Table 2.

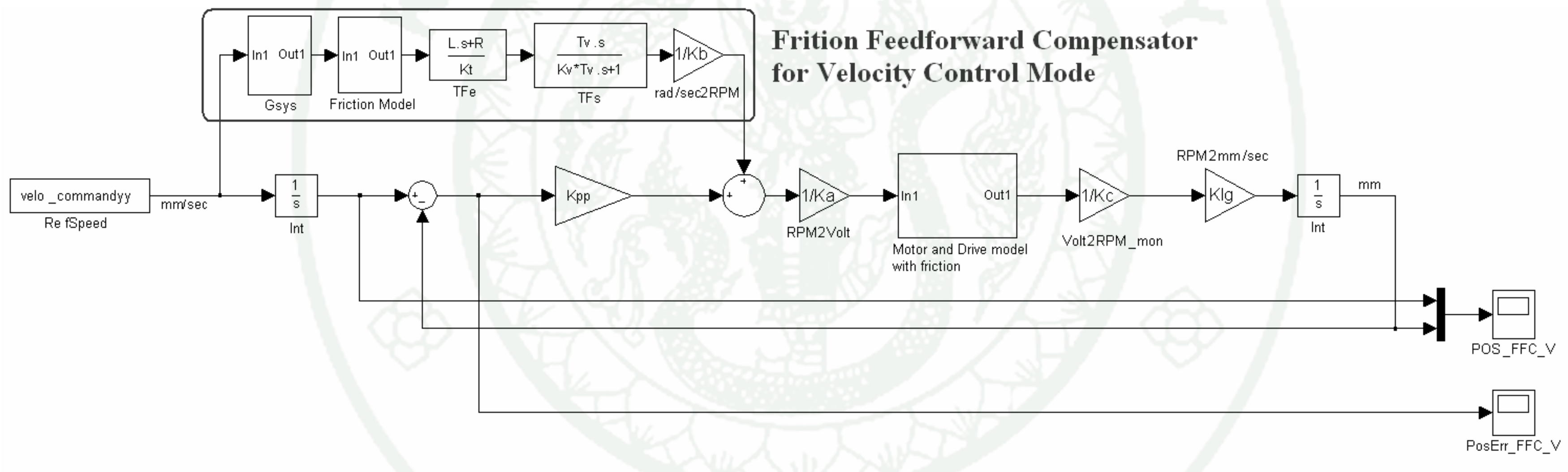


Figure 14 The block diagram of motor control system with FFC and P-Controller

3 Motor Control with PFC, FFC and P-Controller

In this simulation, Position Feedforward Controller (PFC) is added to increase tracking performance of the motor control. The placement of PFC is same as FFC position that is shown in Figure 16. The position controller has parameters that K_{pp} is 100 and K_{ip} is 0. FFC block diagram is same as previous simulation, motor control with FFC and P-controller simulation. The transfer function of Position Feedforward Controller, PFC_v, in Figure 16 is inverse transfer function of motor with driver model which is multiplied by integral block diagram. Position Feedforward Controller can be obtains as

$$PFC_v(s) = \frac{1}{\frac{1}{s}G_{md}(s)} \tag{15}$$

where $G_{md}(s)$ is transfer function of motor with driver model shown in Figure 15. To obtain this PFC, the m-file code is

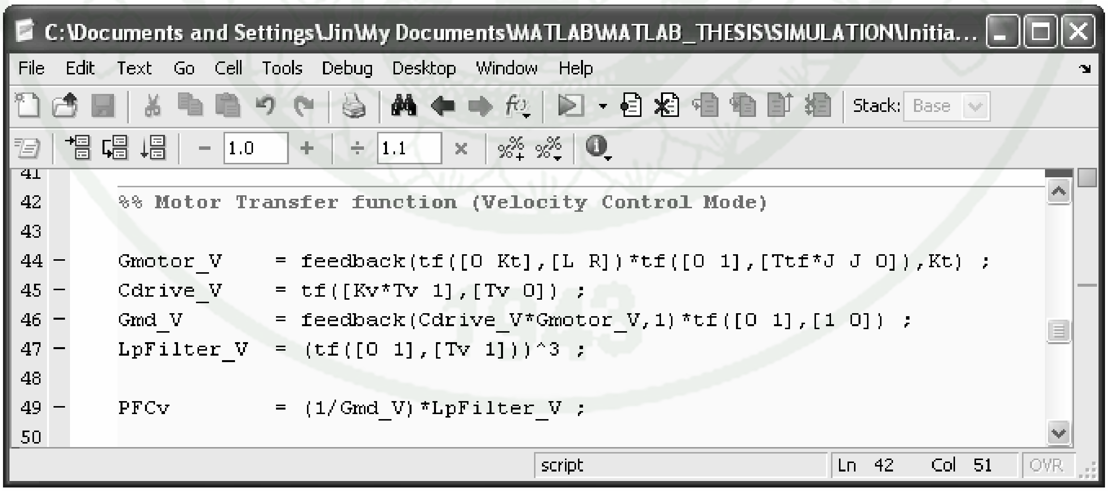


Figure 15 M-File code for obtaining PFC transfer function

where $Gmotor\_V$ = motor transfer function

$Cdrive\_V$ = speed controller of driver transfer function

Gmd_V = motor and driver transfer function

$LpFilter\_V$ = Speed controller low-pass filter transfer function

$PFCv$ = Position Feedforward Controller transfer function

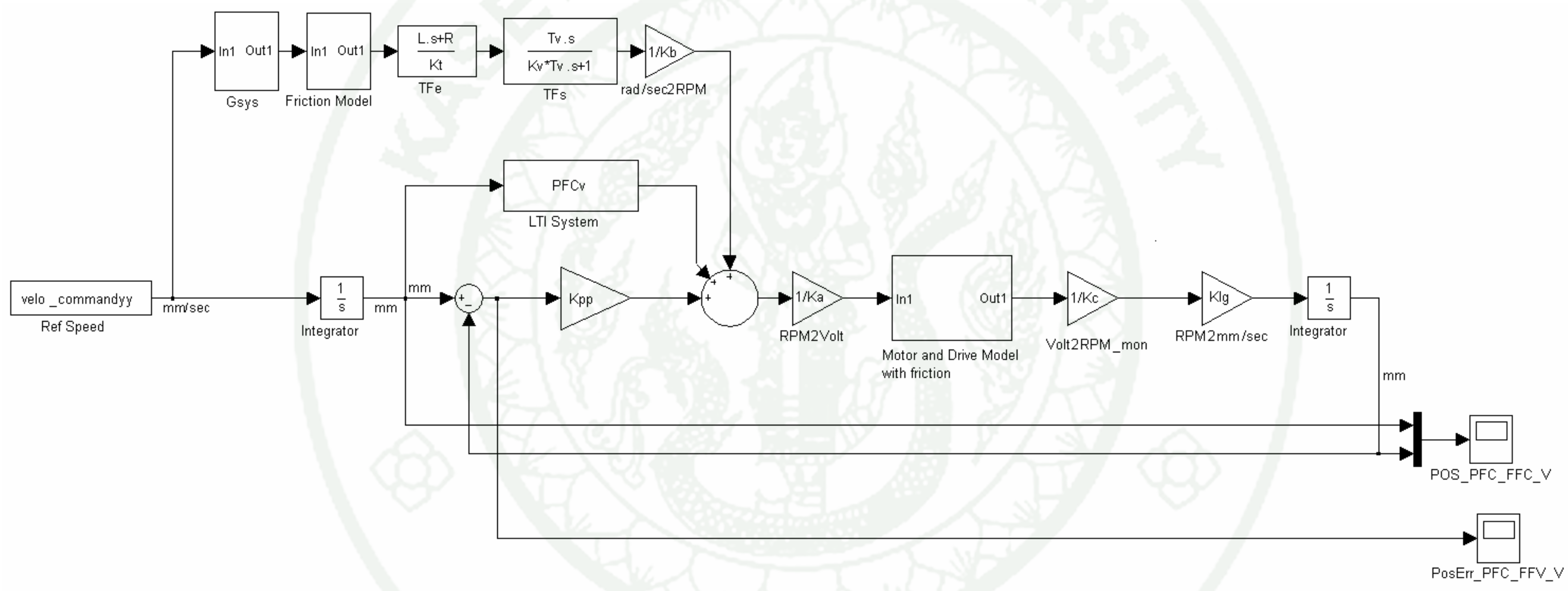


Figure 16 The block diagram of motor control system with PFC, FFC and P-controller

Torque Control Mode

In this simulation, the motor model is changed because the motor driver is set to torque control mode. The motor model is remained only dynamic part which is included torque filter. The motor model for simulation in torque control mode is shown in Figure 17 where Kvt is a driver parameter used to convert voltage to torque.

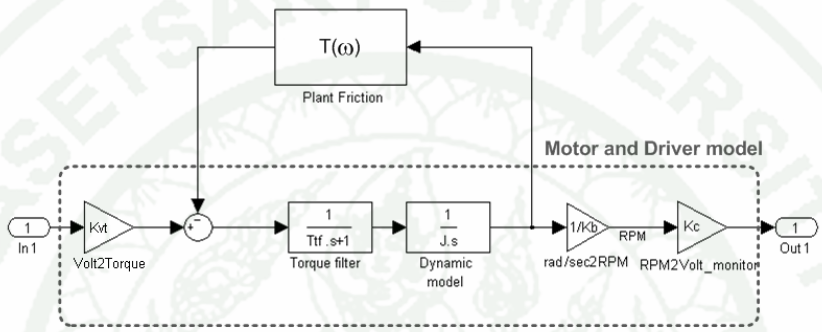


Figure 17 The motor model for simulation in torque control mode with friction

1. Motor Control with P-Controller for Position Loop and PI-Controller for Velocity Loop

The simulation use P-Controller to control motor position and PI-Controller to control motor speed. The position controller use P-Controller that proportional gain, K_{pp}, is 100 and the velocity controller use PI-Controller that proportional gain, K_{pv}, is 0.2 and integral gain, K_{iv}, is 0.01. The parameters of position and velocity controller are obtained by experimental. The simulation block diagram is shown in Figure 18. The velocity controller, C_v, compose of both proportional and integral term that can be obtained as

$$C_v(s) = K_{pv} + \frac{K_{iv}}{s} \tag{16}$$

And position controller, C_p, can be obtained as

$$C_p(s) = K_{pp} \tag{17}$$

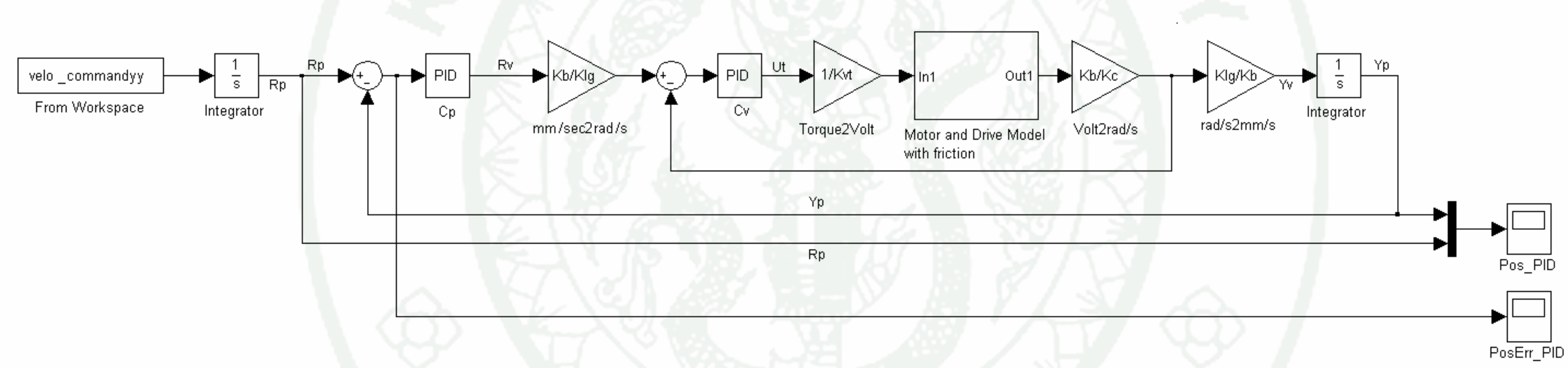


Figure 18 The block diagram of motor control using P-controller in position loop and PI-controller in velocity loop

2. Motor Control with P-Controller and FFC with PI-Controller

This simulation shown in Figure 19 uses P-Controller for position loop and FFC with PI-controller for the velocity loop. The parameters of position and velocity controller are same as the previous simulations of both torque and velocity control mode. The input of FFC of this simulation is velocity output from position loop and the output of FFC is sum with the output of velocity controller, C_v. The FFC uses velocity as the input and torque is the output. Moreover, the FFC is composed of the plant friction model in equation (14) and close-loop transfer function in (11), G_{sys}, of the velocity controller, C_v, and motors with drivers model, G_{mv}, in Figure 3.

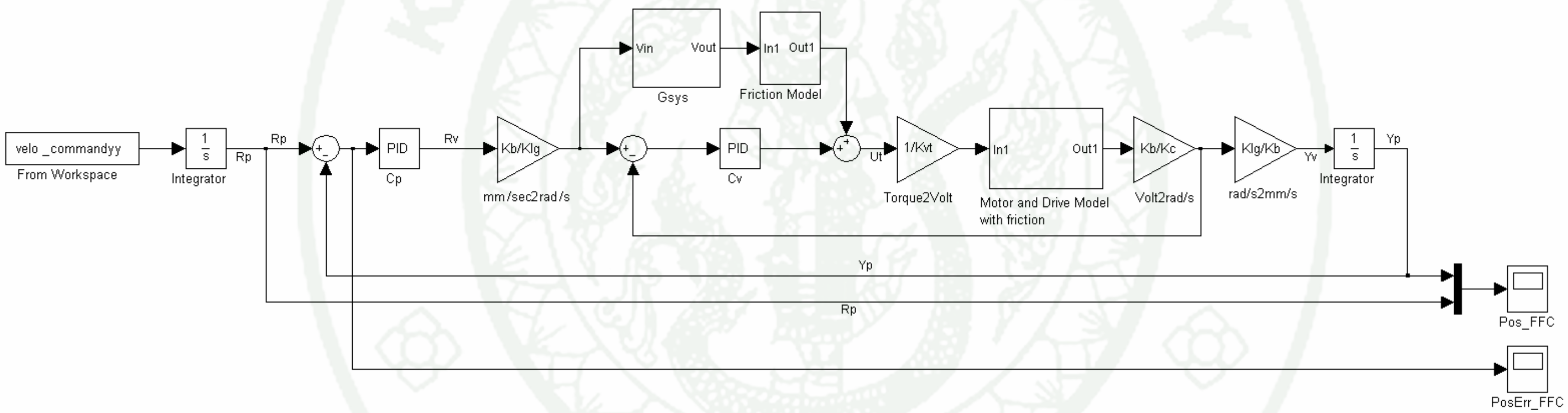


Figure 19 The simulation block diagram of motor control with P-controller and FFC with PI-controller

3. Motor Control Using PFC with P-Controller and FFC with PI-Controller

To simulate position control using Position Feedforward Controller (PFC) and Friction Feedforward Compensation (FFC) technique, FFC and the velocity controller which is a PI-Controller are used for the velocity loop. For the position loop, PFC and P-Controller is used. The block diagram of this simulation is shown in Figure 21. To obtain Position Feedforward Controller, PFC_p, the M-File code is shown in Figure 20 is processed. Other block diagram is same as the last simulation before.

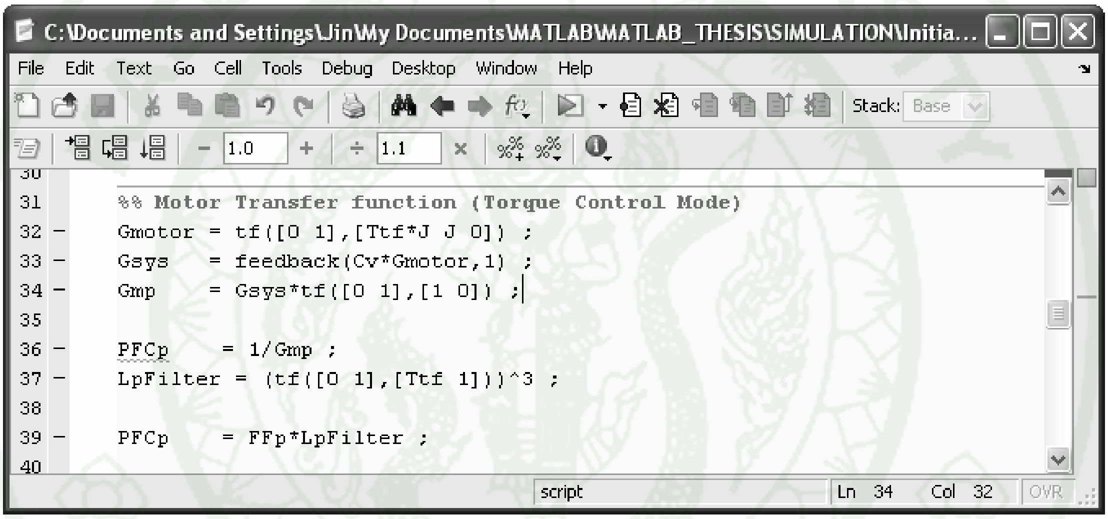


Figure 20 M-File code used to obtain Position Feedforward Controller, PFC_p

Where $Gmotor$ = Motor transfer function

$Gsys$ = Close-loop transfer fucntion of motor and velocity controller

Gmp = Close-loop transfer fucntion of motor and velocity controller with integration.

$LpFilter$ = Low pass filter transfer function

PFC_p = Position Feedforward Controller transfer function for position loop

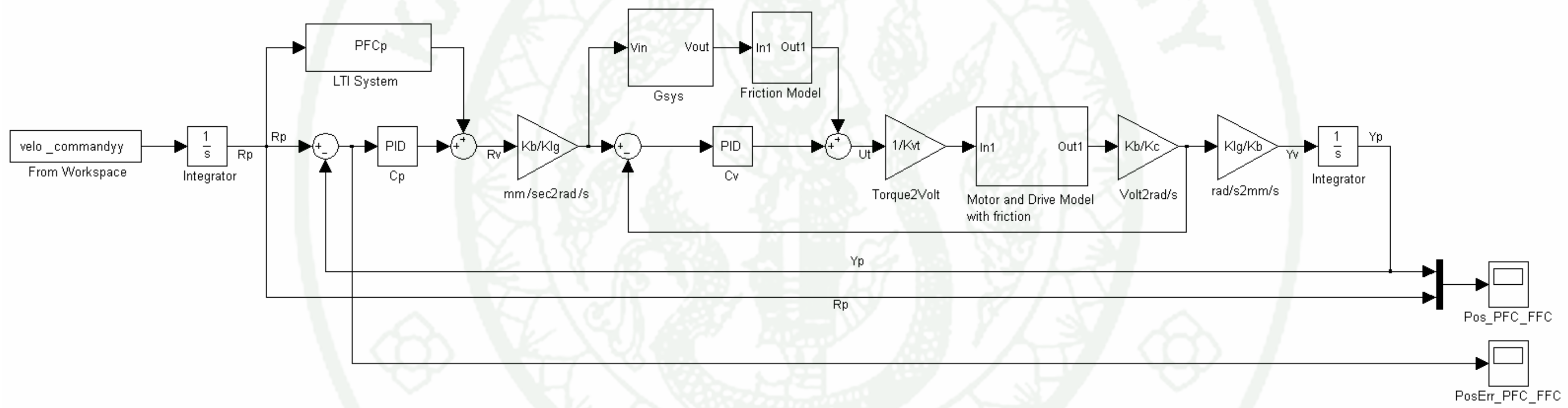


Figure 21 The simulation of motor control using PFC with P-controller and FFC with PI-controller

Implementation

The control system is implemented on the CNC milling machine designed by IMARC laboratory that the X-Y table consists of the MSMA082A1C Panasonic AC servo motor and ball screw with the transmission ratio of 6 mm per motor revolution. The AC servo motor is connected to the MSDA083A1A Panasonic AC servo motor driver. The servo motor is controlled via the PCI-6251 NI Data Acquisition Card (DAQ) with interface to Simulink in MATLAB software. The measured velocity signal is sent to the interface card through the motor driver, and the torque command signal is sent from the interface card to the motor driver to control motor torque. The hardware setup diagram and the system configuration are shown in Figure 22.

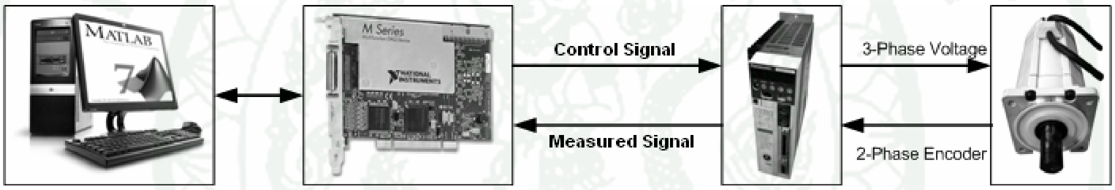


Figure 22 Hardware interface block diagram

From the hardware interface block diagram, PCI-6251 sends the position command to the motor driver, and it receives the measured voltage that refers to the speed from the motor driver.

The algorithm used in the implementation is considered by selection a technique which has best result of simulation that is position control using PFC with P-Controller and FFC with PI-Controller in torque control mode. Moreover, the technique is compared to motor control using PFC, FFC and P-Controller in velocity control mode.

The two implementation block diagrams used with Simulink in MATLAB are shown in Figure 23 and Figure 24. Figure 23 shows the implementation block diagram of motor control using PFC, FFC and P-Controller in velocity control mode. Figure 24 shows the implementation block diagram of motor control using PFC in position loop and FFC in velocity loop. The motor and drive model in simulations is changed to block diagram of an analog output and an analog input of PCI-6251. The analog output connected to command input of the motor driver and the analog input is connected to speed output signal of the motor driver. All of parameters using in this implementation is same as in the simulation, including the controller C_p andC_v.

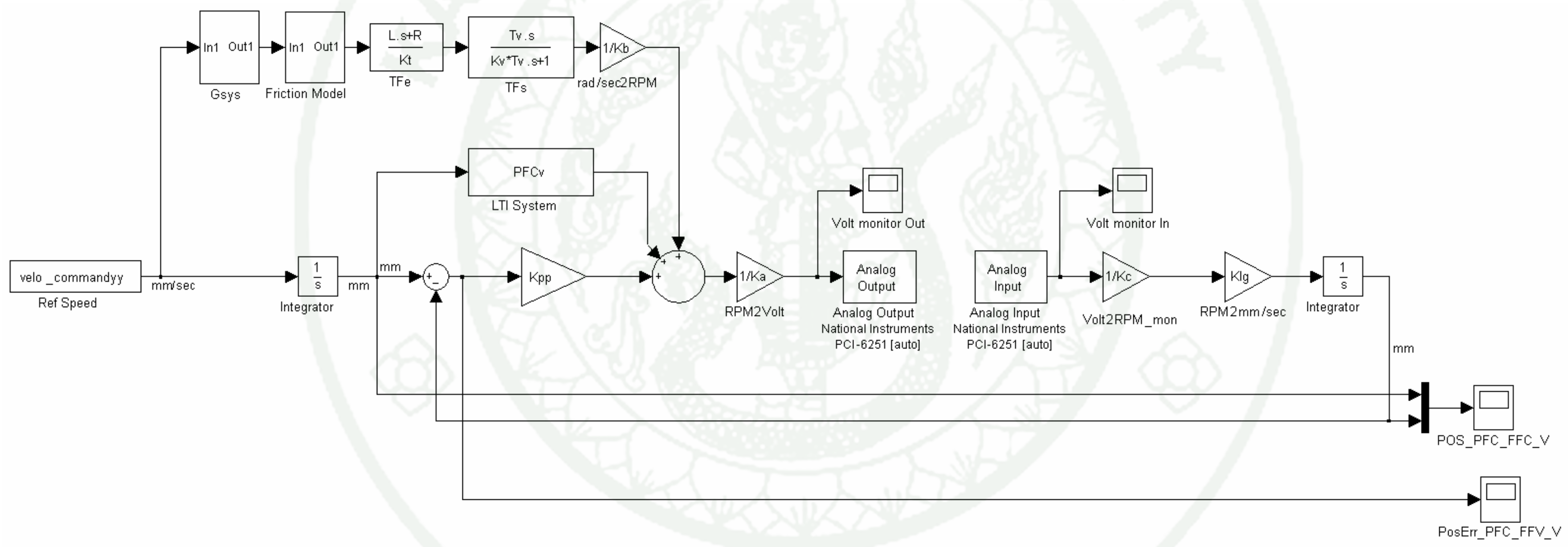


Figure 23 The implementation of PFC, FFC and P-Controller in velocity control mode

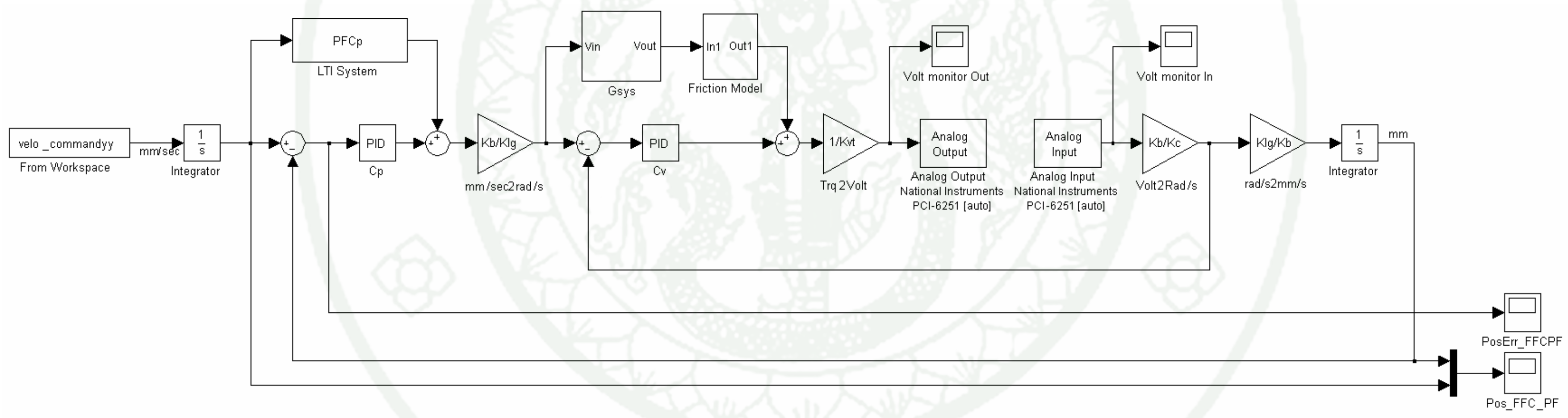


Figure 24 The implementation of PFC with P-Controller and FFC with PI-Controller in torque control mode

Control Unit With Digital Signal Ccontroller (DSP) Design

DSP Circuit Board Design

In this section, the idea and method for designing the prototype board for control CNC milling machine is illustrated. A CNC milling machine used in this research is composed of two servo motors for drive its X-Y table and one servo motor for drive Z-axis linear guide. The motors have their encoders. The prototype board receive the signal from encoder to measure both speed and position to control the motors via their drivers. The prototype board has main functions using in motor control, i.e.,

1. +/-10 Volt Analog output 3 channel
2. 0 -10 Volt encoder input 3 channel

The prototype board use Freescale DSP, MC56F8345, which is specifically designed for motor control. Designing the prototype board, DAC is an important component used to generate the control signal. The range of signal from DAC is 0-5 Volt but the motor driver require input in range +/-10 Volt then the converter circuit is designed using Op-Amp. The encoder input channel used in receiving pulse from motor encoder is designed using opto-isolators to separate signal between motor drivers and the DSP because it is a method to protect the DSP from over voltage in the signal line.

The prototype board can be divided into 7 modules

1. Processing Unit (DSP)
2. Analog Output
3. Encoder Input
4. USB-RS232 Interface
5. Digital Input
6. Relay Output
7. Power Supply

The block diagram of the prototype board is shown in Figure 25.

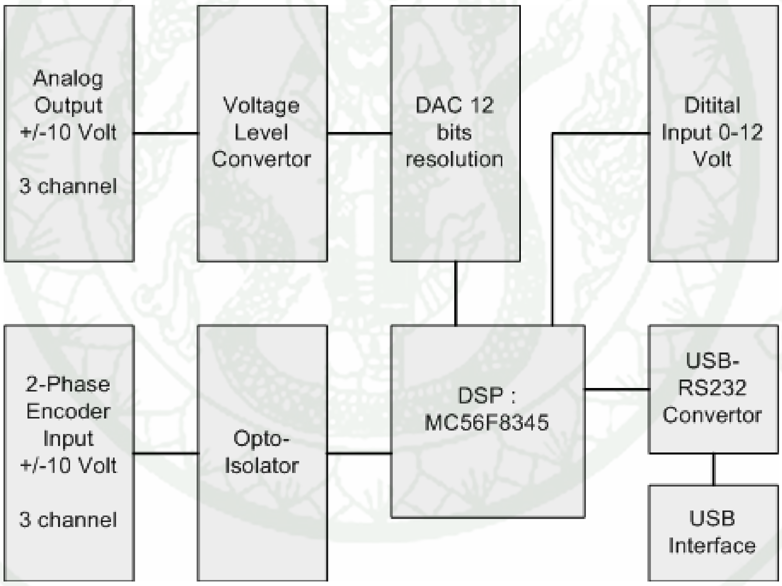


Figure 25 The block diagram of the prototype board

The prototype board is shown in Figure 26.

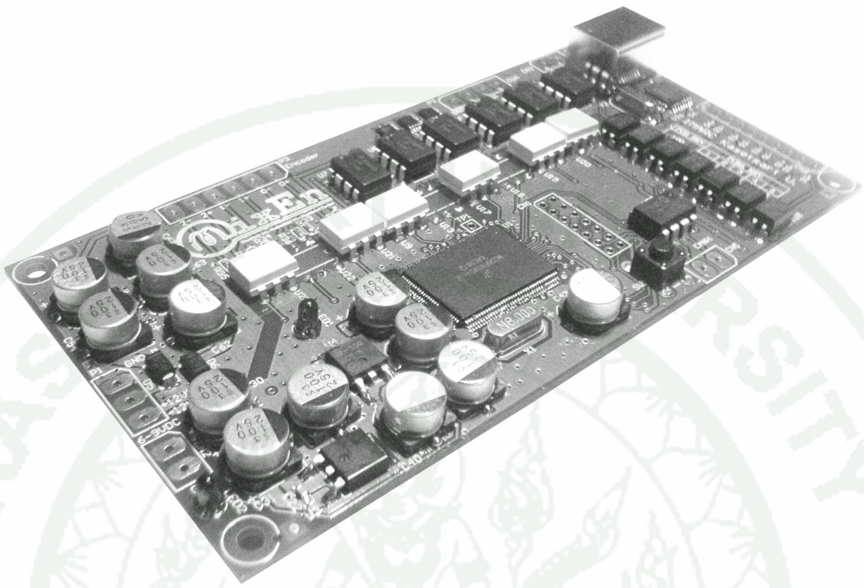


Figure 26 The prototype board

1. Processing Unit (DSP)

The processing unit, DSP, which is the heart of the prototype board containing the software algorithm for motor control. The DSP circuit use 8 MHz crystal and phase lock loop for increasing the clock speed to 60 MHz for processing algorithm. The features of the DSP are:

1.1. Two Pulse Width Modulator modules (PWMA and PWMB), each with six PWM outputs, three current sense inputs, and three fault inputs for PWMA/PWMB; fault-tolerant design with dead time insertion, supporting both center-aligned and edge-aligned modes.

1.2. Two 12-bit Analog-to-Digital Converters (ADCs), supporting two simultaneous conversions with dual 4-pin multiplexed inputs; the ADC can be synchronized by PWM modules.

1.3. Two Quadrature Decoders (Quad Dec0 and Quad Dec1), each with four inputs, or two additional Quad Timers, A and B.

1.4. Two dedicated general purpose Quad Timers totaling three pins: Timer C with one pin and Timer D with two pins CAN 2.0 B-compatible module with 2-pin ports used to transmit and receive.

1.5. Two Serial Communication Interfaces (SCI0 and SCI1), each with two pins, or four additional GPIO lines.

1.6. Serial Peripheral Interface (SPI), with configurable 4-pin port, or four additional GPIO lines.

1.7. Computer Operating Properly (COP) / Watchdog timer.

1.8. Two dedicated external interrupt pins.

1.9. 61 multiplexed General Purpose I/O (GPIO) pins.

1.10. External reset pin for hardware reset.

1.11. JTAG / On-Chip Emulation (OnCE).

1.12. Software-programmable, Phase Lock Loop-based frequency synthesizer for the hybrid controller core clock.

1.13. Temperature Sensor system.

The DSP in the prototype board is MC56F8345, which has the following memory configuration:

1. Program Flash 64K x 16-bit
2. Data Flash 4K x 16-bit
3. Program Ram 2K x 16-bit
4. Data Ram 4K x 16-bit
5. Boot Flash 4K x 16-bit

The DSP circuit is shown in Figure 27.

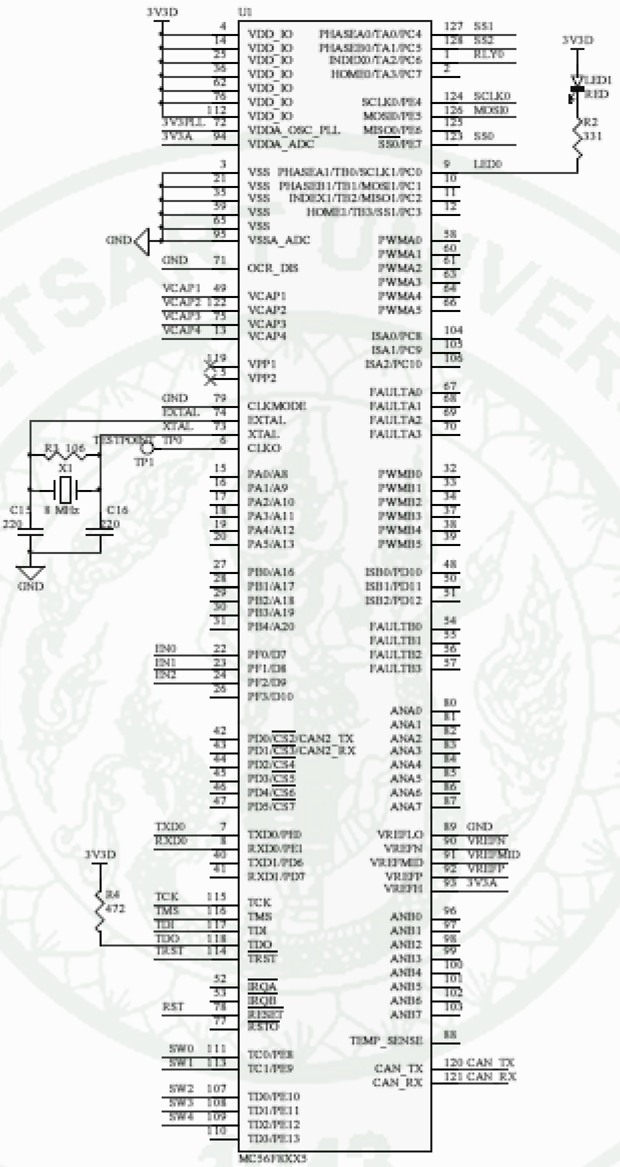


Figure 27 The DSP circuit design

However, The DSP has one quadrature encoder input but the prototype board for 3-axis motor control has to use three quadrature encoder inputs. Therefore, the quadrature encoder peripheral of the DSP is not be used. It has to use General Purpose Input/Output (GPIO) as three channel quadrature encoder inputs.

2. Analog Output

To drive the motor, the control signal in range of +/-10 Volt is needed. The designed circuit use Digital to Analog Converter (DAC) generating the signal in range 0-5 Volt and converting to +/-10 Volt using 2-stage operation amplifier. First stage of Op-Amp convert signal from 0-5 Volt to +/-5 Volt using positive feedback amplifiers with gain 2. The circuit in second stage is same as first stage that can amplify the signal from +/-5 Volt to +/-10 Volt. The analog output circuit is shown in Figure 28.

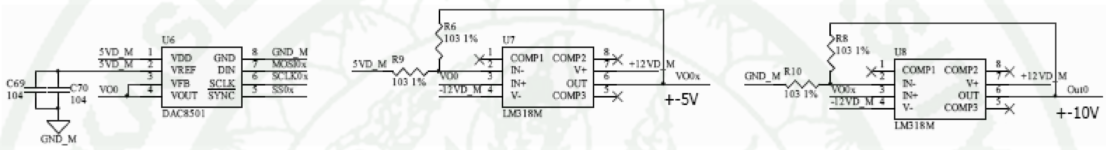


Figure 28 The analog output circuit for one channel

For the symbol of the Op-Amp circuit in Figure 28 is shown in Figure 29.

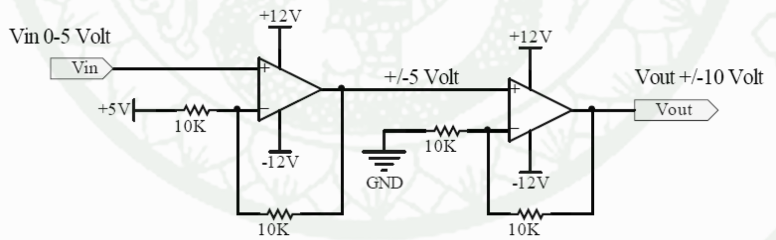


Figure 29 The Op-Amp circuit for changing voltage from 0-5 Volt to +/-10 Volt.

3. Encoder Input

The encoder inputs designed are single phase inputs. The encoder input circuit is shown in Figure 30.

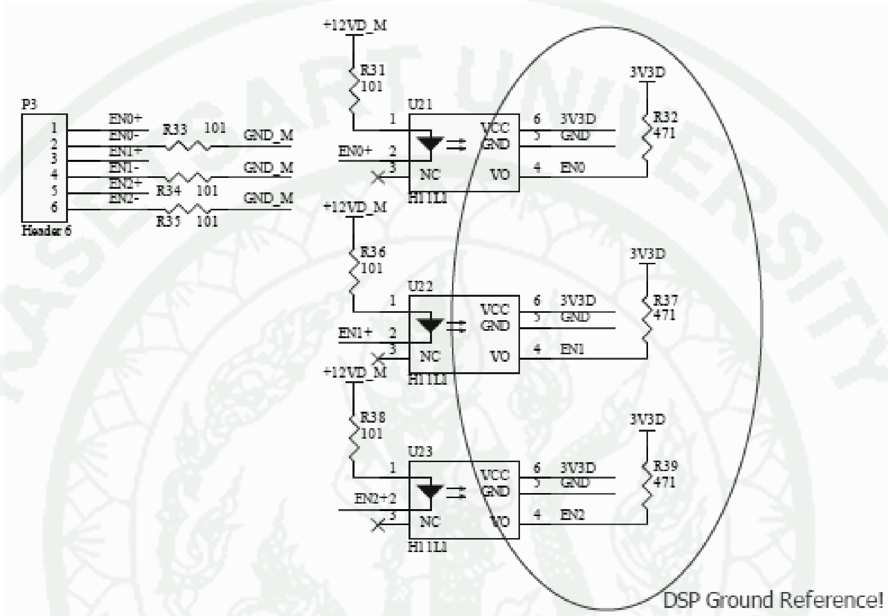


Figure 30 The encoder input circuit

Because the encoder signal from the motor is over 3.3 Volt, the encoder input circuit is designed by using opto-isolators to receive and separate the signal for protection and reduction noise generated from the motor.

However, the prototype board is not support for the servo motors. It has to modify the encoder inputs because the encoder signal is in range of +/-10 Volt but the prototype board can receive voltage between 0 to10 Volt. Since the prototype board has single phase encoder inputs but the pulse signal from the servo motors are 2-phase encoder signal then another signal board must be design to receive one more phase signal.

4. USB-RS232 Interface

The interface between a PC and the prototype board is USB-RS232 that is a serial interface type. The advantages of this interface are:

1. Easy to connect to a PC that does not have any COM port.
2. The USB wire used for this interface is designed for high speed data transfer. therefore the maximum data transfer rate depend on circuit and PCB design only.

The USB-RS232 circuit design is shown in Figure 31.

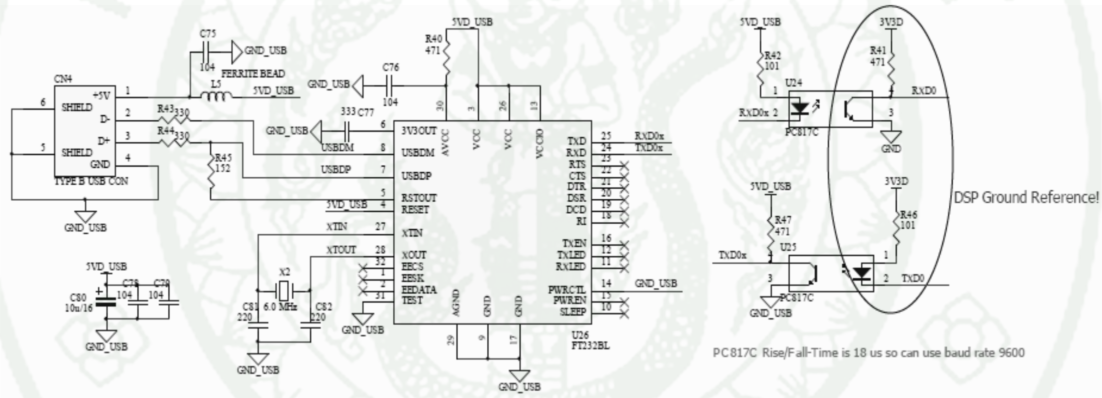


Figure 31 USB-RS232 circuit

CN4 is a type-B USB connector and U26 is the IC used to convert USB to RS232 and vise versa. To protect the DSP from overvoltage or noise, opto-isolator, PC817, is used to separate the data transmission lines. The maximum data transfer rate of the prototype board is 9600 bit per second (bps) because of limited of DSP and circuit design.

5. Digital Input

The digital input circuit is designed for using with external switch such as emergency switch or command switches. The digital inputs of the prototype board have 5 channels. The Figure 32 shows a channel of the digital input circuit.

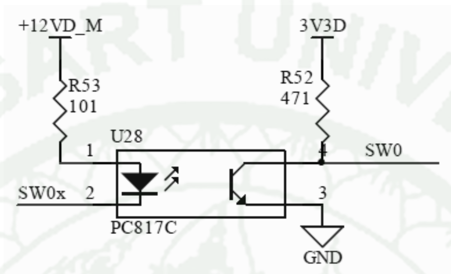


Figure 32 The digital input circuit

6. Relay Output

The relay output on the prototype board is used to turn on or turn off the spindle of the CNC milling machine. The circuit forward current through the coil of an external relay that the coil voltage of the relay has to be 12 Volt. The relay output circuit is shown in Figure 33.

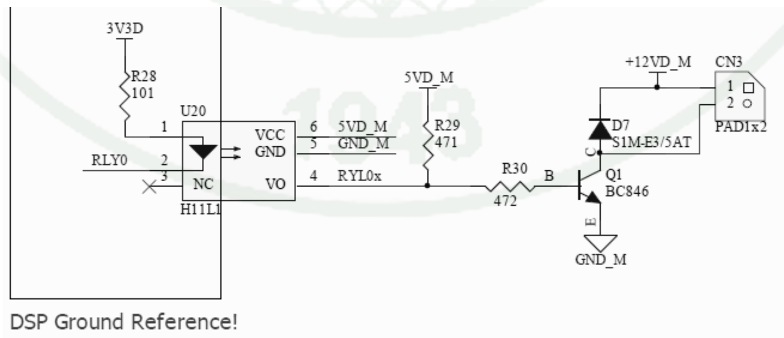


Figure 33 The relay output circuit for driving spindle

7. Power Supply

The power supply unit is a +/-12 voltage switching supply and a 12 voltage switching supply. The +/-12 voltage supply is used with the motor related circuits such as analog outputs, and encoder inputs. For the 12 voltage supply, the buck regulator circuit is used to change the voltage level from 12 Volt to 5 Volt. However, the DSP has to use more than one level of voltage supply. The 3.3 voltage regulators are used to regulate from 5 Volt to 3.3 Volt that is separated the power by using ferrite bead for noise protection. The block diagram of the power unit is shown in Figure 34.

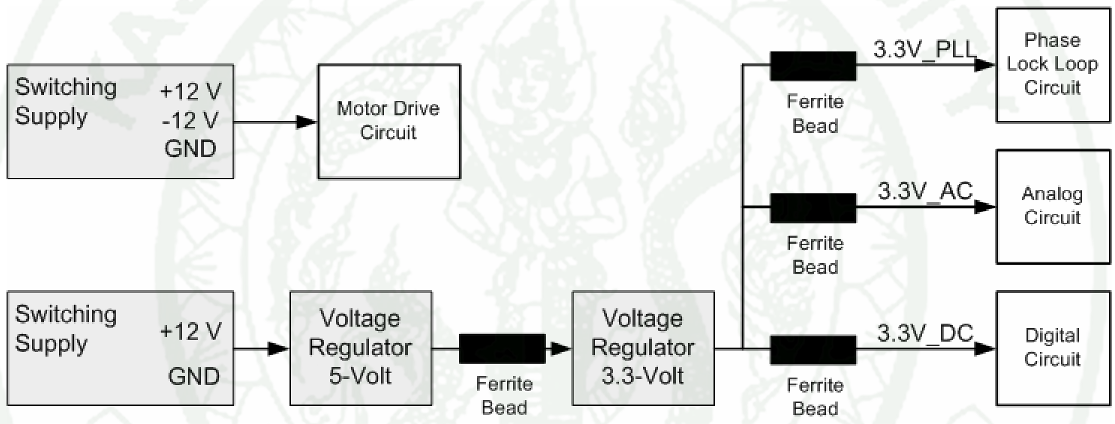


Figure 34 The block diagram of power supply circuit

Control Algorithm Design

A programming for control the CNC milling machine is developed by using C language. The algorithm is divided into 2 parts, first part is a control algorithm such as PID controller, Friction Feedforward Compensation (FFC), and Position Feedforward Controller (PFC). Second part is a pulse counter for measuring speed and position of the motors.

1. Pulse Counter Algorithm

The pulse counter algorithm is designed to count the pulse which has 2-phase signal per axis generated by the motors. The pulse signals has a phase shift of 90 degree.It can be divided into 4 states for determining the motor revolution direction, speed, and position. The 2-phase pulse signal is shown in Figure 35. The general purpose input pins of the DSP are used to capture the pulse signal every 20 microseconds and to update the speed and position every 10 microseconds. Thus, Nyquist frequency of state changing is 25 kHz, therefore the maximum speed of the motor can be found by

$$MiximumSpeed(RPM) = \frac{60}{(2 \times SampleTime) \times NumState \times PulsePerRev}, \quad (18)$$

The maximum speed of the motor is 150 RPM, where sample time is 20 microseconds, NumState is 4, and PulsePerRave is 2500.

The flowchart of the pulse counter algorithm is shown in Figure 36.

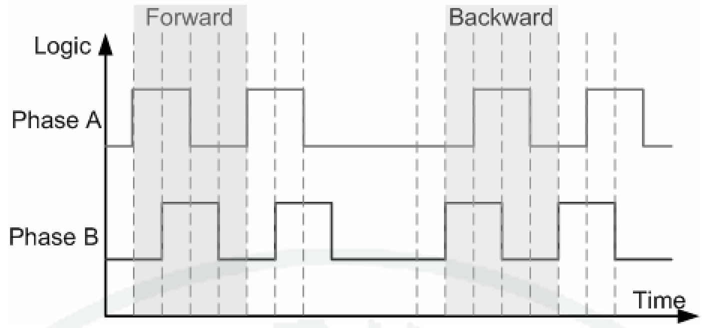


Figure 35 The 2-phase pulse signal

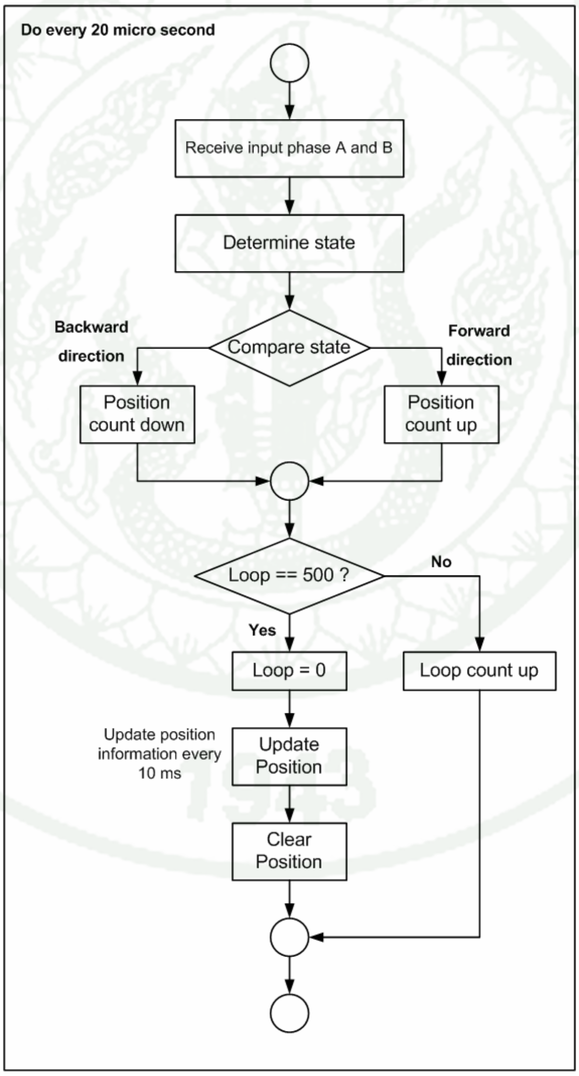


Figure 36 The pulse counter flowchart

2. Control Algorithm

For the control algorithm, the sample period is 10 millisecond which is sufficient for the motors control. To control the position of motor, the reference position or desired position, and the actual position measured from the motor encoder are required input signal, and the control signal is the output. The difference between the reference signal and actual position value is an error which is an input to the controller of the motor control system. The flow chart of motor control is shown in Figure 37.

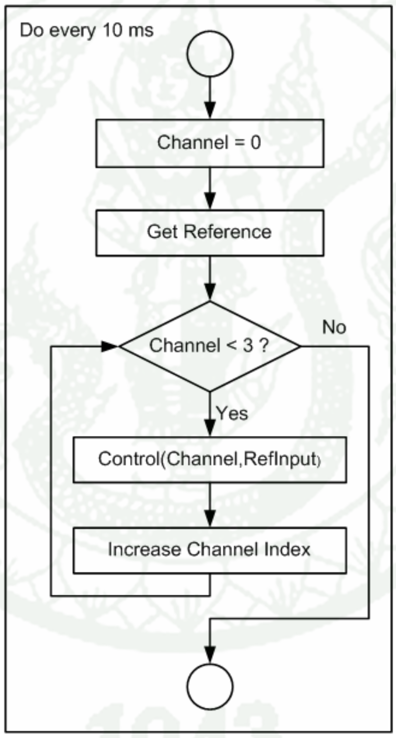


Figure 37 The flow chart of motor control

The function control is depended on the control method such as PI controller, Friction Feedforward Compensation, and etc. The example of PI controller designed is illustrated. A block diagram of a controller is shown in Figure 38.

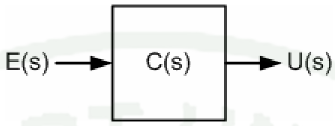


Figure 38 The block diagram of a controller

Where $E(s)$ represent error signal and $U(s)$ means control signal. The transfer function of PI controller, $C(s)$, can be obtained as

$$C(s) = K_p + \frac{K_i}{s}, \quad (19)$$

where K_p is a proportional parameter and K_i is an integral parameter of the controller. To convert from continuous time domain to discrete time domain, Shift-Tustin method in (20) is used.

$$s = \frac{2(z-1)}{T(z+1)} \quad (20)$$

The transfer function of the controller in discrete time domain can be obtained as

$$C(z) = K_p + \frac{K_i T(z+1)}{2(z-1)}. \quad (21)$$

From the block diagram in Figure 38. After convert to discrete time domain, the transfer function is

$$C(z) = \frac{U(z)}{E(z)} \quad (22)$$

Then,

$$\frac{U(z)}{E(z)} = \frac{(2K_p + K_iT) + (-2K_p + K_iT)z^{-1}}{2(1 - z^{-1})} \tag{23}$$

$$2(1 - z^{-1})U(z) = (2K_p + K_iT)E(z) + (-2K_p + K_iT)z^{-1}E(z) \tag{24}$$

It can convert equation (24) to

$$2u[k] - 2u[k-1] = (2K_p + K_iT)e[k] + (-2K_p + K_iT)e[k-1] \tag{25}$$

The controller output used to control the motors can be obtained as

$$u[k] = u[k-1] + (K_p + \frac{K_iT}{2})e[k] + (-K_p + \frac{K_iT}{2})e[k-1] \tag{26}$$

For programming on DSP, the variables are arrays of integer type i.e., e for error and u for control signal. The algorithm is

1. $e[1] = e[0]$
2. $u[1] = u[0]$
3. $u[0] = u[1] + (K_p + \frac{K_iT}{2})e[0] + (-K_p + \frac{K_iT}{2})e[1]$

Where index $[1]$ is refer to previous value of the variables. The output of the controller is $u[0]$ which is an input of function that generate output voltage to the motor driver. The flow chart of control function is shown in Figure 39.

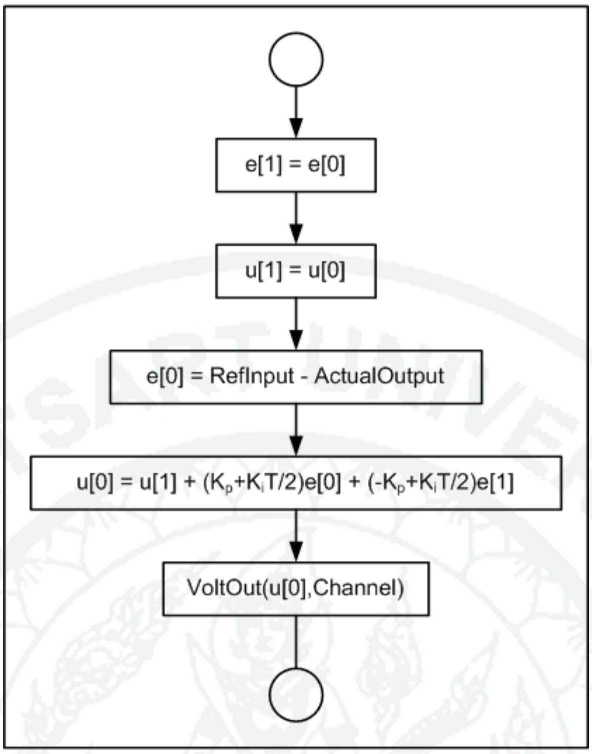


Figure 39 The flow chart of PI control algorithm

RESULTS AND DISCUSSION

Results

Simulation Results

Simulation results shown in this section are divided into two parts which are the result of velocity and torque control mode.

Simulation Result of Velocity Control Mode

The simulation results in velocity control mode are shown in Figure 40, Figure 41, and Figure 42. Figure 40 shows the position control results in both forward and backward directions. The position errors are shown in Figure 41 and the position absolute errors are shown in Figure 42.

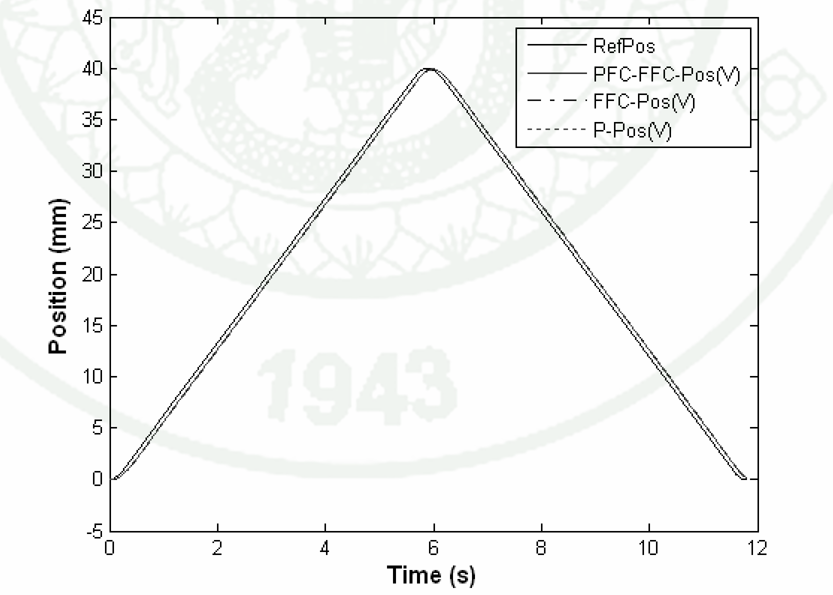


Figure 40 The position control result comparison of simulation in velocity control mode

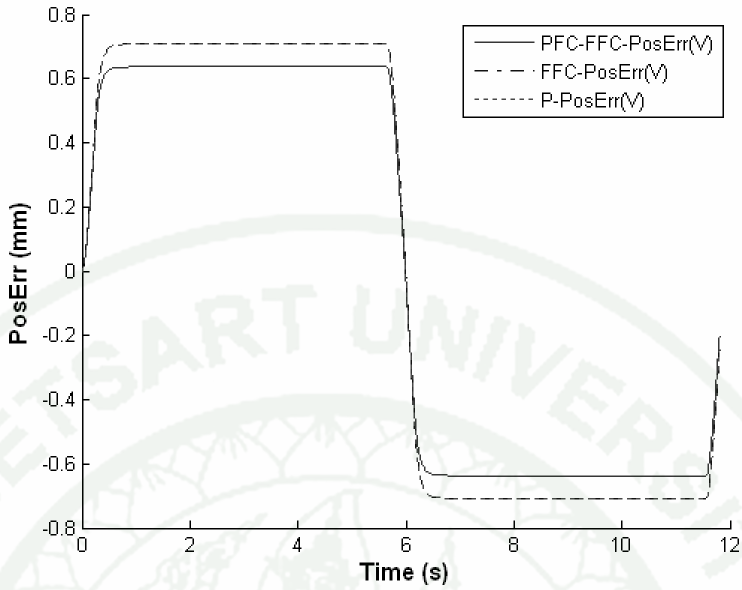


Figure 41 The position error comparison of simulation in velocity control mode

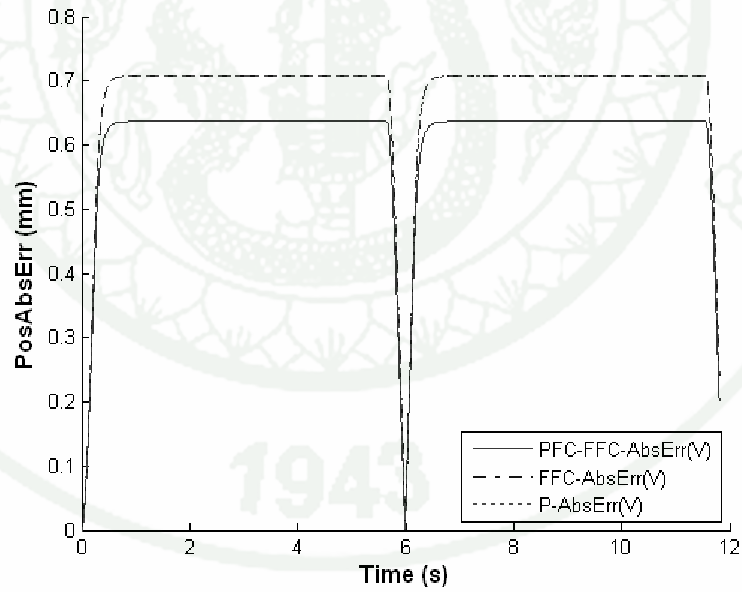


Figure 42 The position absolute error comparison of simulation in velocity control mode

The maximum absolute error and mean absolute error of simulations in velocity control mode are shown in Table 2. The best control method in velocity control mode is the control method using PFC, FFC and P-Controller. The maximum error is only 636.4 μm and mean absolute error is 574.6093 μm.

Table 2 Comparison of errors of velocity control mode

Controller Structure	**Error (μm)**	
	Maximum Error	**Mean Absolute Error**
P-Controller	707.1084	638.1195
FFC and P-Controller	707.1107	638.0696
PFC, FFC and P-Controller	636.4000	574.6093

Simulation Results of Torque Control Mode

Simulation results in torque control mode are shown in Figure 43, Figure 44, Figure 45. Figure 43 shows the position control results of three controller structures. Figure 44 shows the position errors of position control with three controller structures. For Figure 45, the absolute errors are plot.

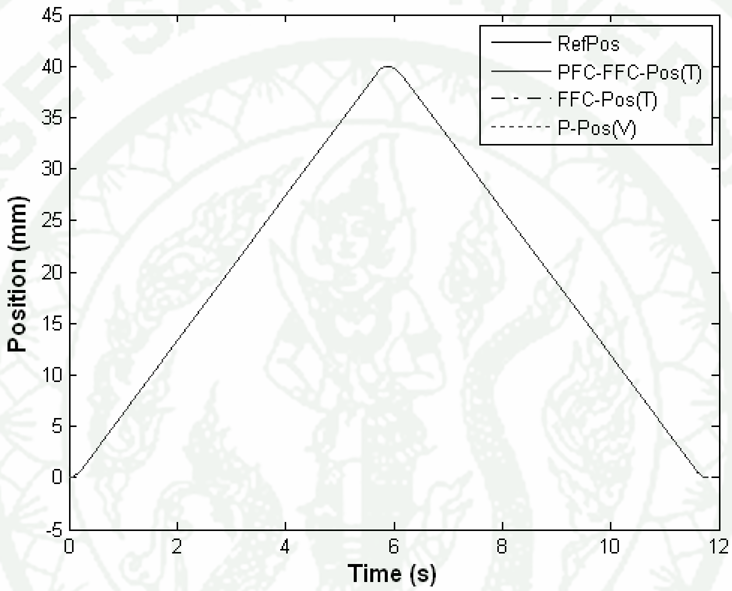


Figure 43 The position control result comparison of simulation in torque control mode

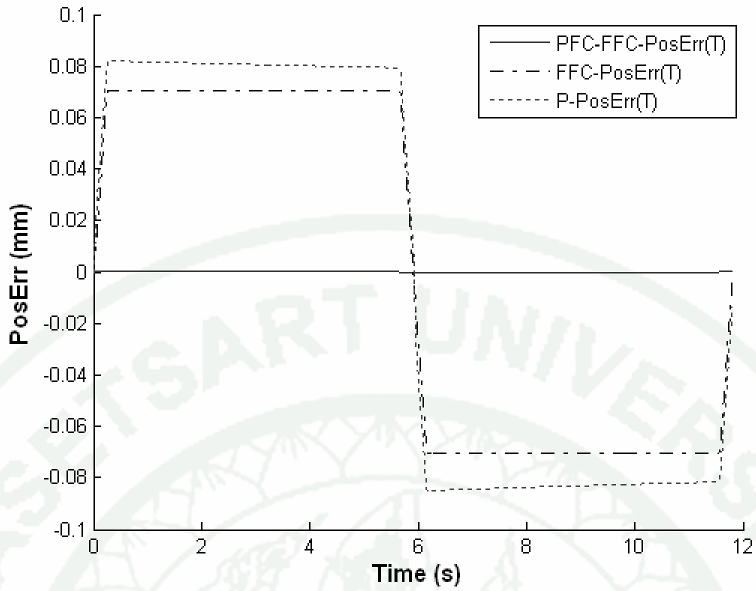


Figure 44 The position error comparison of simulation in torque control mode

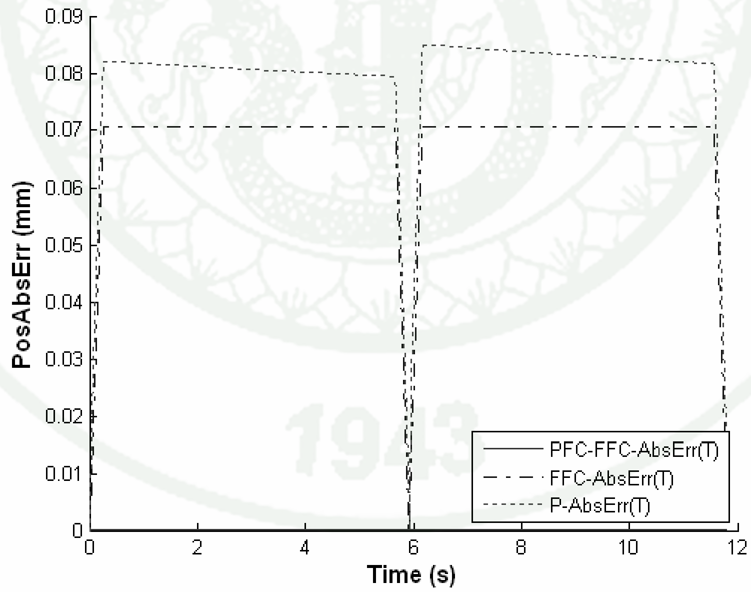


Figure 45 The position absolute error comparison of simulation in torque control mode

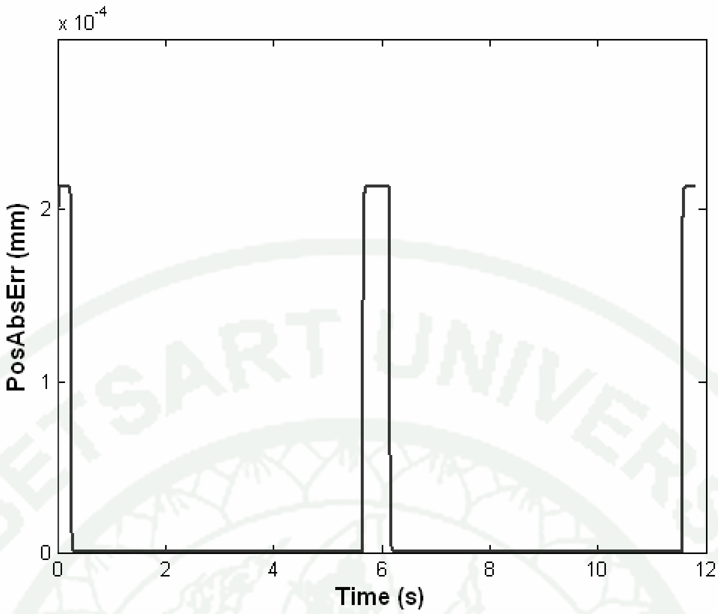


Figure 46 The position absolute error of motor control using PFC with P-Controller and FFC with PI-controller

Figure 46 shows the absolute error of motor control using PFC with P-Controller and FFC with PI-controller. To compare what control mode is better, the comparison results in velocity and torque control mode are shown in Figure 47 and Figure 48. Figure 47 shows the comparison of velocity and torque control mode using P-Controller in position control loop. For the position errors, the comparison is shown in Figure 48. The simulation results in Figure 47 and Figure 48 show that control in torque control mode yields the better result than control in velocity control mode. Because in torque control mode, it has to design velocity controller. The algorithm of this research is using FFC with PI-Controller in velocity loop for torque control mode, while velocity control mode uses the internal driver velocity controller.

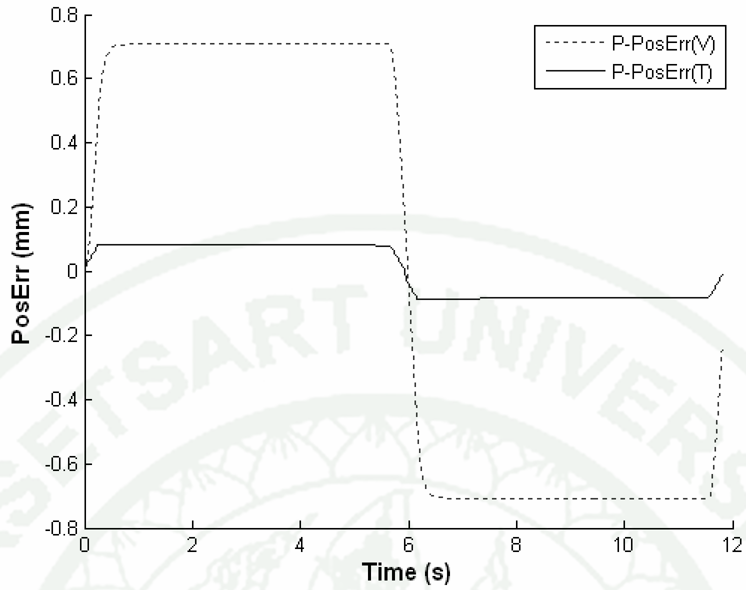


Figure 47 The position error comparison of velocity and torque control mode using P-Controller in position control loop

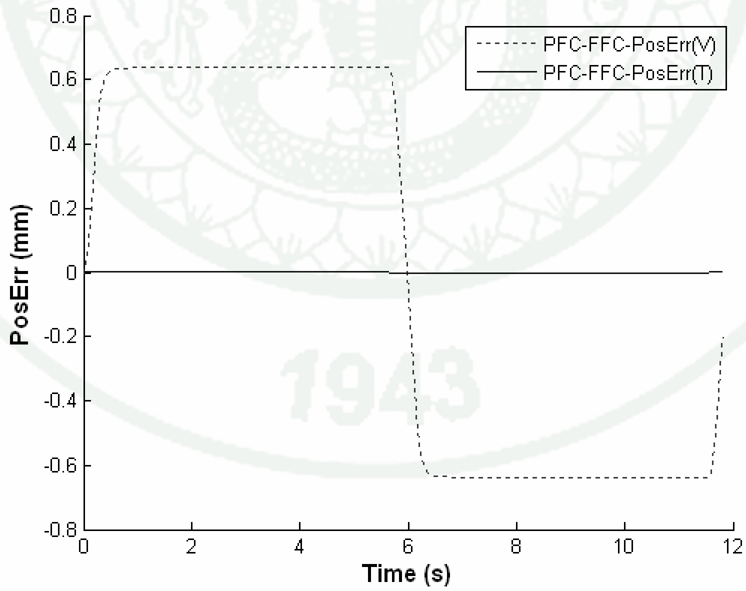


Figure 48 The position error comparison of velocity and torque control mode using PFC with P-Controller in position control loop

The maximum absolute error and mean absolute error of simulations in torque control mode are shown in Table 3. The minimum error of the simulations in torque control mode is the result of using PFC with P-Controller and FFC with PI-Controller that The maximum error is only 0.2138 µm and mean absolute error is 0.0276 µm.

Table 3 Comparison error of torque control mode

Controller Structure	Error (µm)	
	Maximum Error	**Mean Absolute Error**
P-Controller and PI-Controller	84.9933	75.1922
P-Controller and FFC with P-Controller	70.7089	64.3546
PFC with P-Controller and FFC with PI-Controller	0.2138	0.0276

The mean absolute error comparison between velocity and torque control mode are shown in Table 4. The best result is the motor control using PFC and FFC in torque control mode.

Table 4 Mean absolute error comparison

Mean Absolute Error (µm)		
Controller Structure	**Velocity Control Mode**	**Torque Control Mode**
P-Controller	638.1195	75.1922
FFC	638.0696	64.3546
PFC and FFC	574.6093	0.0276

Implementation Result

This section, the implementation results are presented. For implementations, PCI-6251 DAQ Card is used. The comparisons between using PFC combine to FFC in velocity and torque control modes are shown in Figure 49 and Figure 50.

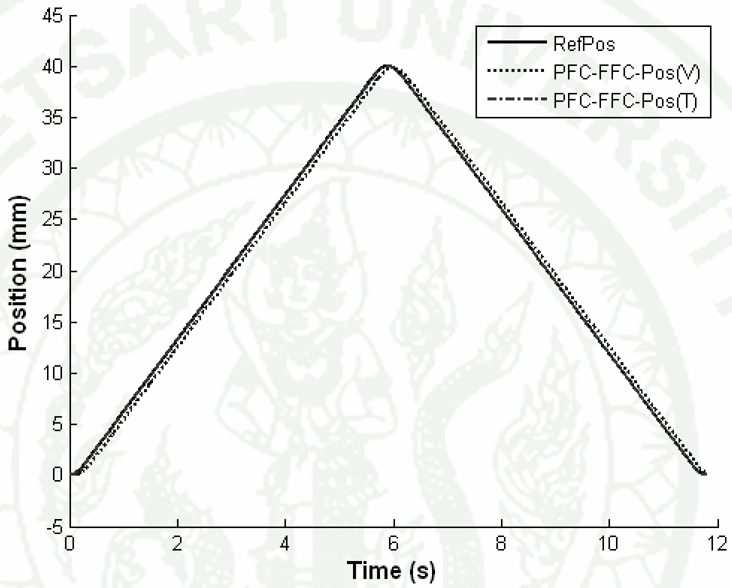


Figure 49 Position control result

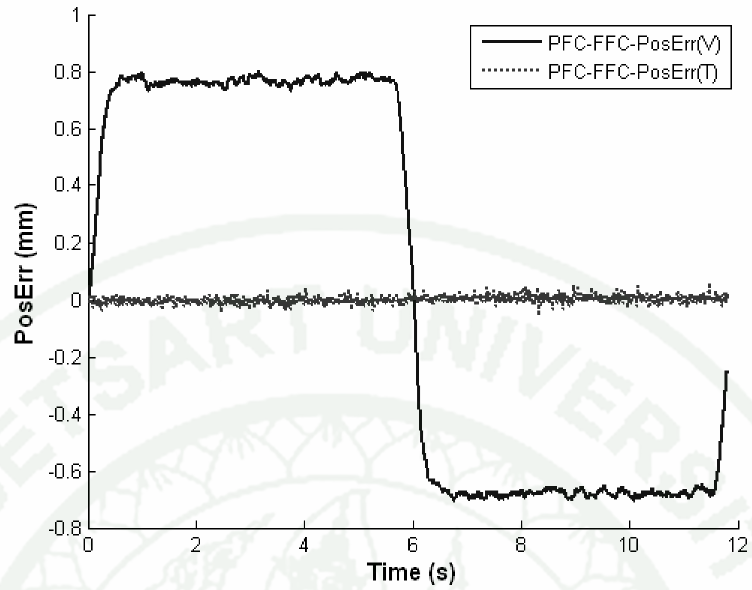


Figure 50 Position error of using PFC and FFC between velocity and torque control mode

The implementation result, position error in Figure 50 is very similar to the simulation result in Figure 48. The error of using PFC and FFC in toque control mode is very low compared to using same algorithm in velocity control mode as shown the mean absolute errors in Table 5.

Table 5 Mean absolute error comparison between using PFC and FFC in velocity and torque control mode

Mean Absolute Error (µm)		
Controller Structure	**Velocity Control Mode**	**Torque Control Mode**
PFC and FFC	689.5422	7.8694

The absolute error of implementation using PFC and FFC in torque control mode shown in Figure 51 is very noisy because of electromagnetic interfereance.

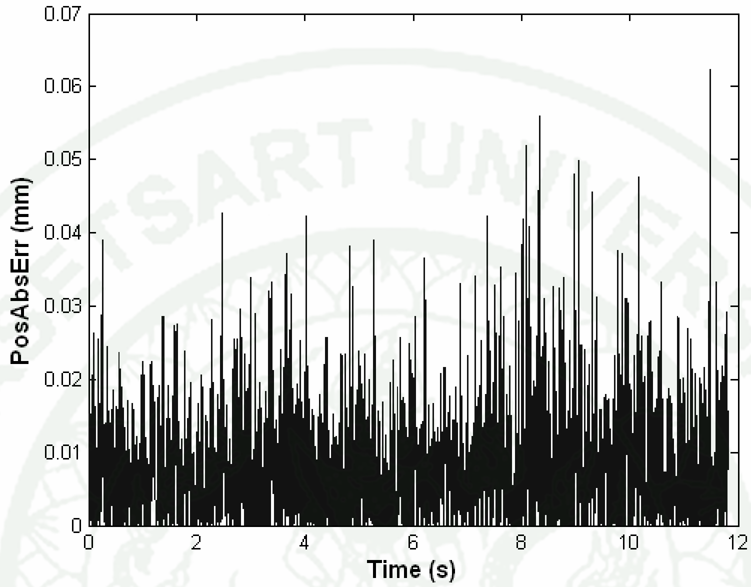


Figure 51 The absolute error of motor control using PFC and FFC in torque control mode

Discussion

The simulations in torque control mode have better results than the simulations in velocity control mode. The simulations using PFC and FFC give the best results when control in both velocity and torque control modes. However, control using PFC and FFC in torque control mode yields the best result, i.e., the mean absolute error is only 0.0276 micrometer .

The implementation result is conducted by control in torque control mode. The result is similar to the simulation that using PFC and FFC yield the best result. The mean absolute error of implementation result is only 7.8694 micrometer.

CONCLUSION AND RECOMMENDATION

Conclusion

Using PFC with P-Controller combined with FFC and PI-Controller in torque control mode is a good technique to reduce tracking error, and eliminate the friction effect on the mechanical components. The simulation and implementation results are similar since the friction model is similar to the real friction on the plant. After implementing P-Controller with the FFC-PFC, the mean absolute error is 7.8694 μm or 0.0098 % of the total moving distance. The next step of the present research work is to apply the motor control using PFC and FFC in torque control mode by C language on DSP circuit board which designed by IMARC laboratory.

Recommendation

The friction model used in this research is identified in one direction. It should be identified again both forward and backward direction using same friction model, Armstrong's model. To obtained the best result for minimizing position error, parameters of velocity and position controller in torque control mode should be improved again. Moreover, The DSP circuit board should be redesigned because the present DSP circuit board have to be modified significantly. Moreover, The present communication interface is not support high speed communication.

LITERATURE CITE

Erkorkmaz, E. and A. Yusuf. 2001. High Speed CNC System Design. Part II: Modeling and Identification of Feed Drives. **International Journal of Machine Tools & Manufacture**. 1487-1509

Industrial and Appliance Motor Division, Motor Co., Matsushita Electric Industrial Co., Ltd. **AC Servo Motor Driver MINAS A-series Operating Manual**.

Lee, D.-H., and J.-W. Ahn. 2008. Dual Speed Control Scheme of Servo Drive System for a Nonlinear Friction Compensation. **IEEE Transaction On Power Electronics**. Vol.23. No.2. 959-965.

Lee, H.S., and M. Tomizuka. 1996. Robust Motion Controller Design for High-Accuracy Positioning Systems. **IEEE Transaction on Industrial Electronics**. Vol. 43. No.1. 48-55.

Lee, S.-W. and J.-H. Kim. 1995. Robust Adaptive Stick-Slip Friction Compensation. **IEEE Transaction on Industrial Electronics**. Vol. 42. No.5. 474-479.

Mcintosh, T. 2004. **Calculating the Axis Level Closed Loop Gains**. Integrated Industrial Technologies, Inc.

Olsson, H., K. L. Aström, C. Canudas de Wit, M. Gäfvert, and P. Lischinsky. 1998. Friction Models and Friction Compensation. **European Journal of Control**. Vol. 4. 176-195.

Rafael, K., and L. Jeśus. 1999. Determination of Viscous and Coulomb Friction by Using Velocity Response to Torque Ramp Inputs. **IEEE International Conference on Robotics & Automation**. 1740-1745.

Tsai, M.-C., I.-F. Chiu and M.-Y. Cheng. 2004. Design and implementation of command and friction feedforward control for CNC motion controllers. **IEE Proc.-Control Theory Appl**. Vol. 151. No. 1. 13-20.

Tung, E. D., G. Anwar, and M. Tomizuka. 1993. Low Velocity Friction Compensation and Feedforward Solution Based on Repetitive Control. **Journal of Dynamic Systems, Measurement, and Control, Transaction of the ASME**. Vol. 115. 279-284.

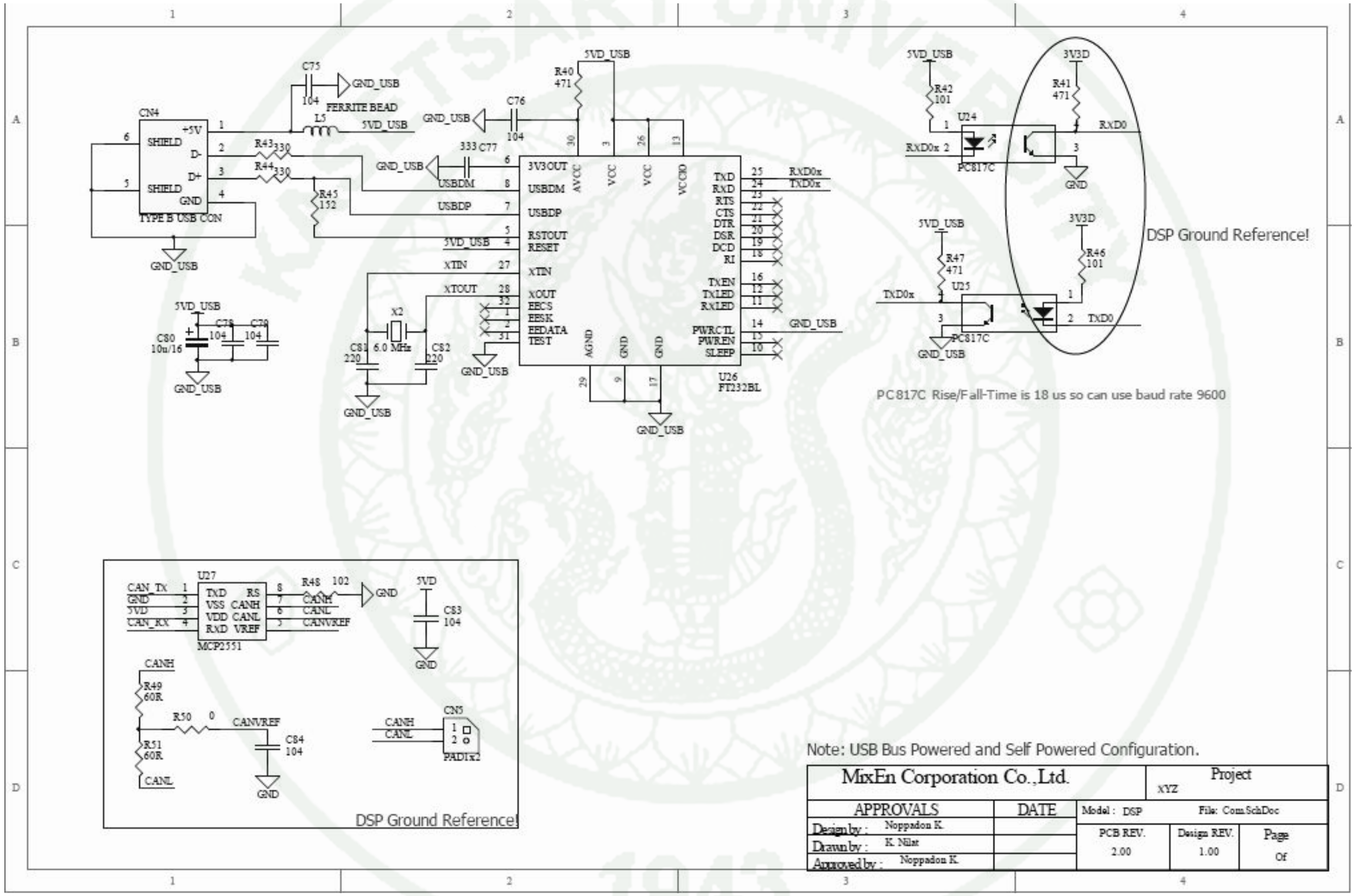
Wijin, S., R. Rapeepong, L. Chutima, S. Peerayot, C. Nattapon, T. Kanokvate, and H. Kunieda. 2010. Eliminating Friction Affect of CNC Machines Using Friction Feedforward Compensation. **The International Conference on Information and Comuication Technology for Embedded Systems**.

Yamazaki, T. 2003. Mathematical Model for Feed Drive System in Microscopic Motion Area. **Archives of Materials Science and Engineering**. Vol. 33. 35-38.

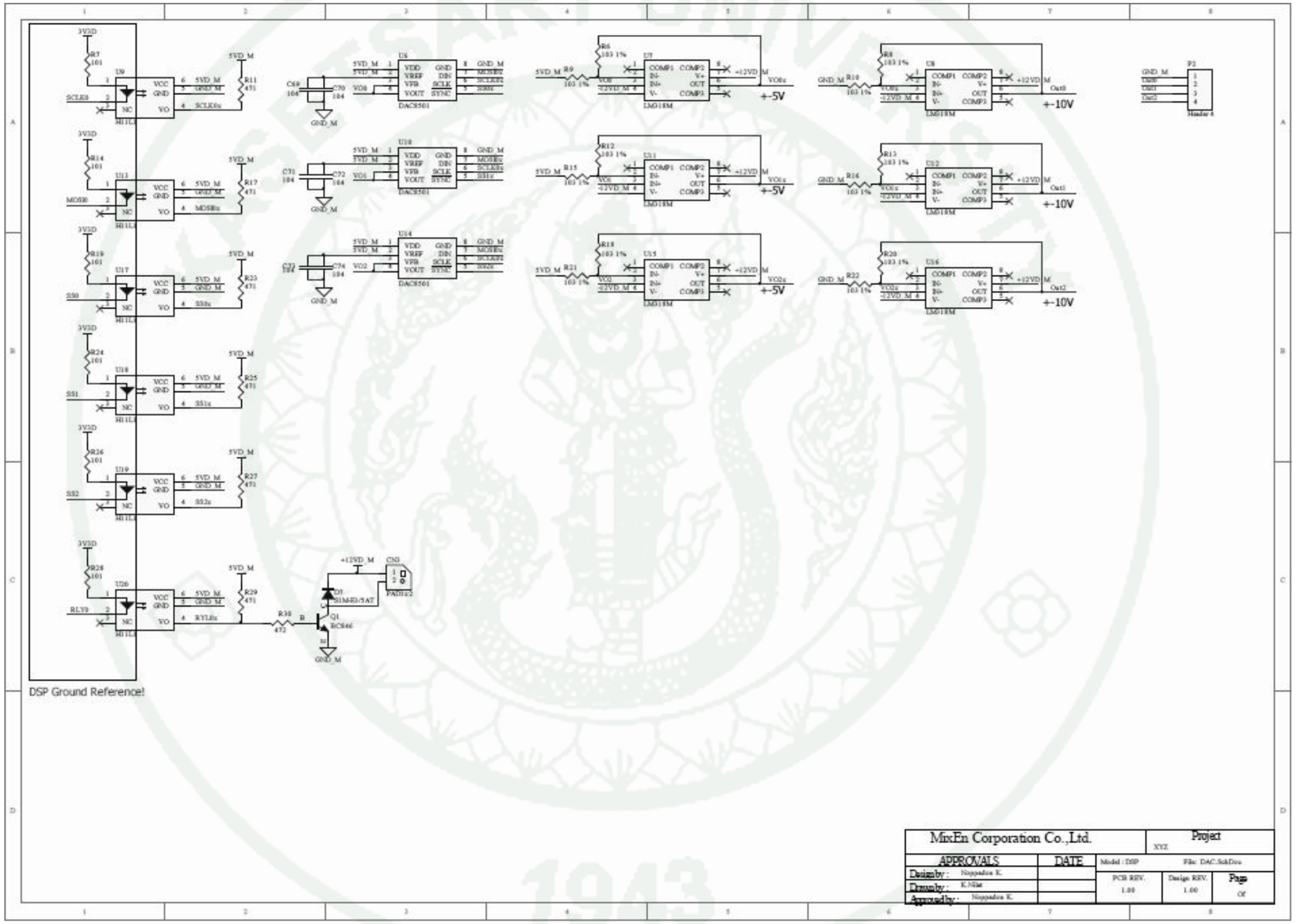
APPENDICES

Appendix A

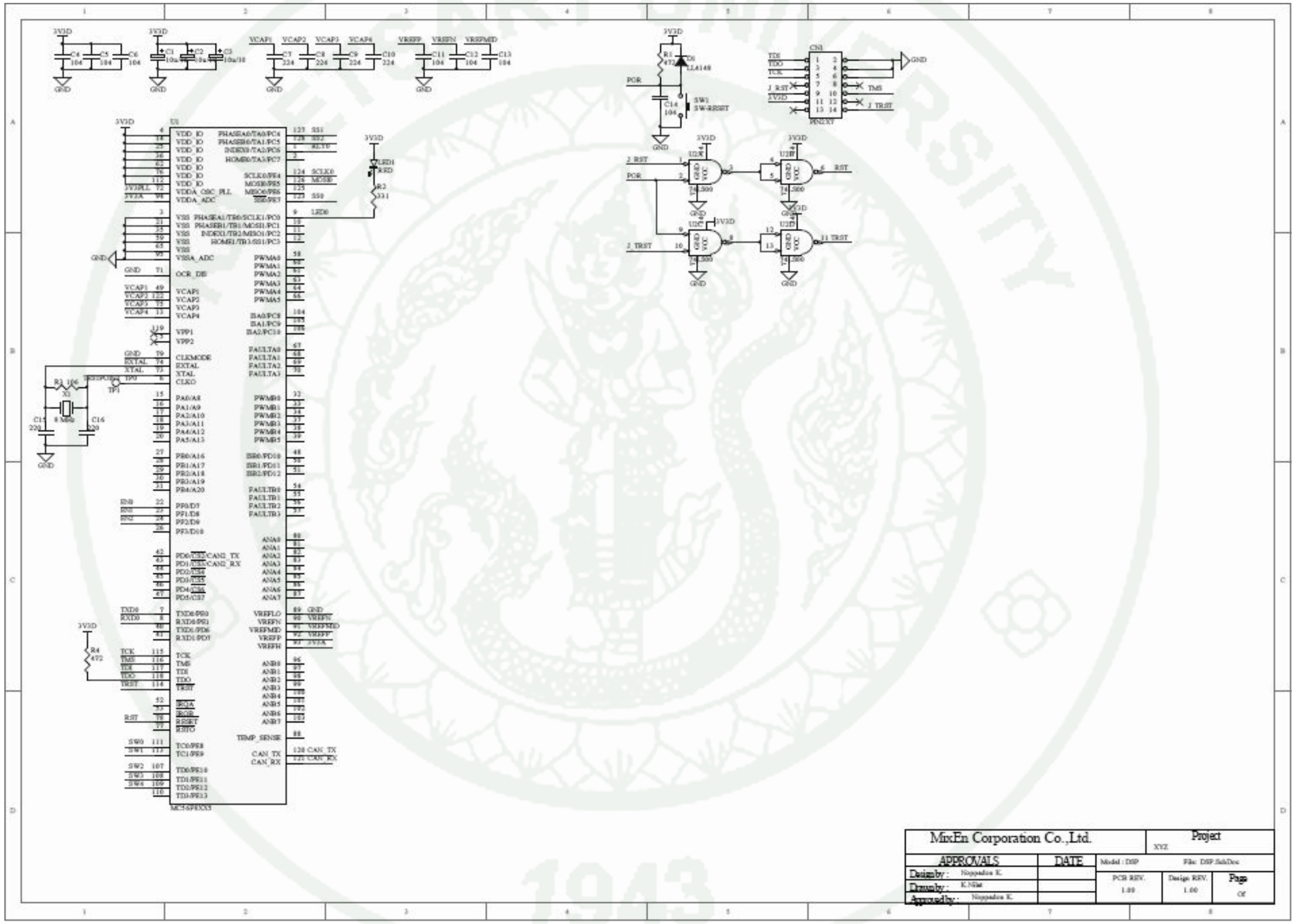
Schematic and PCB of the DSP circuit board



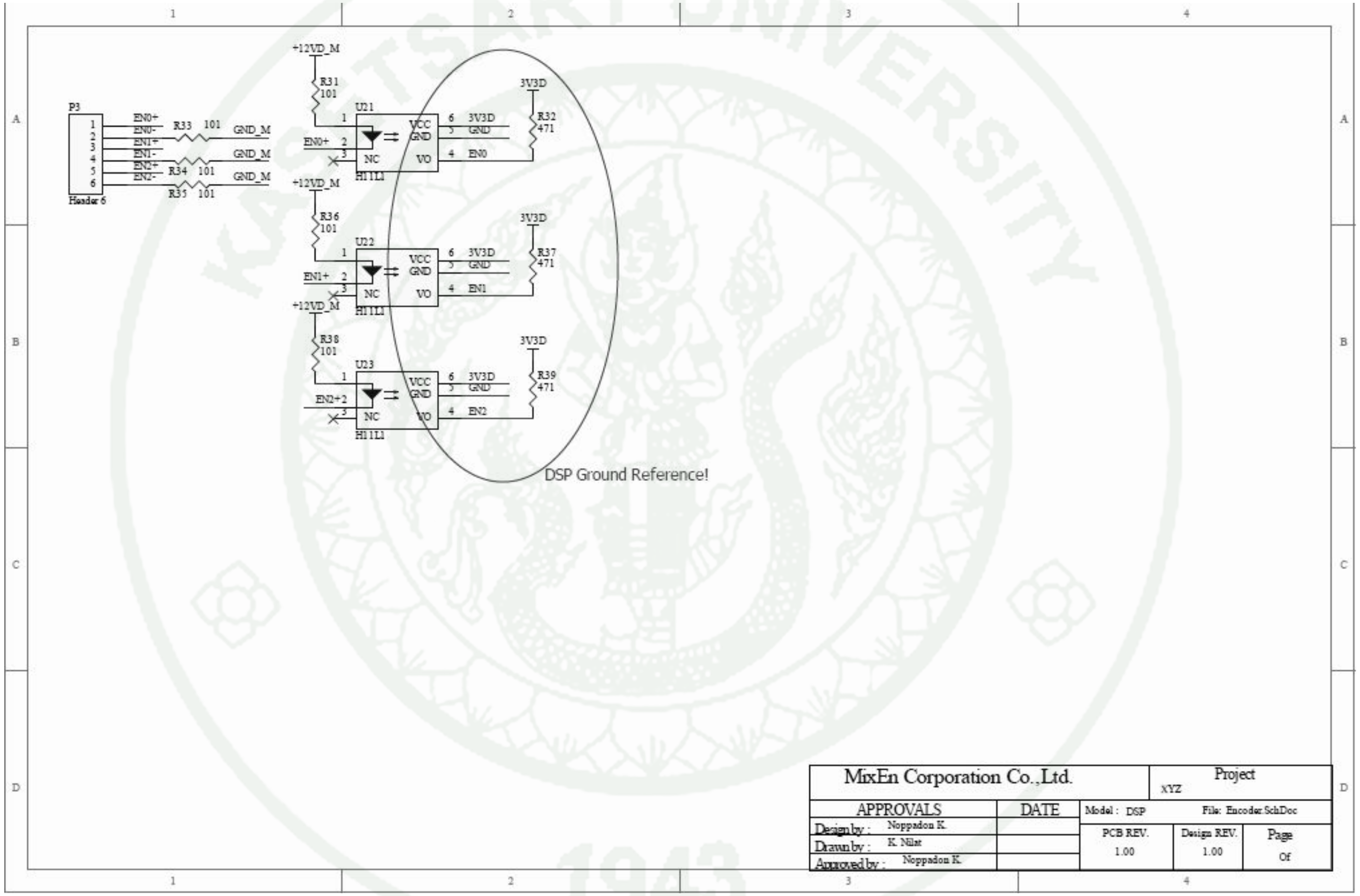
Appendix Figure A1 USB-RS232 interface schematic



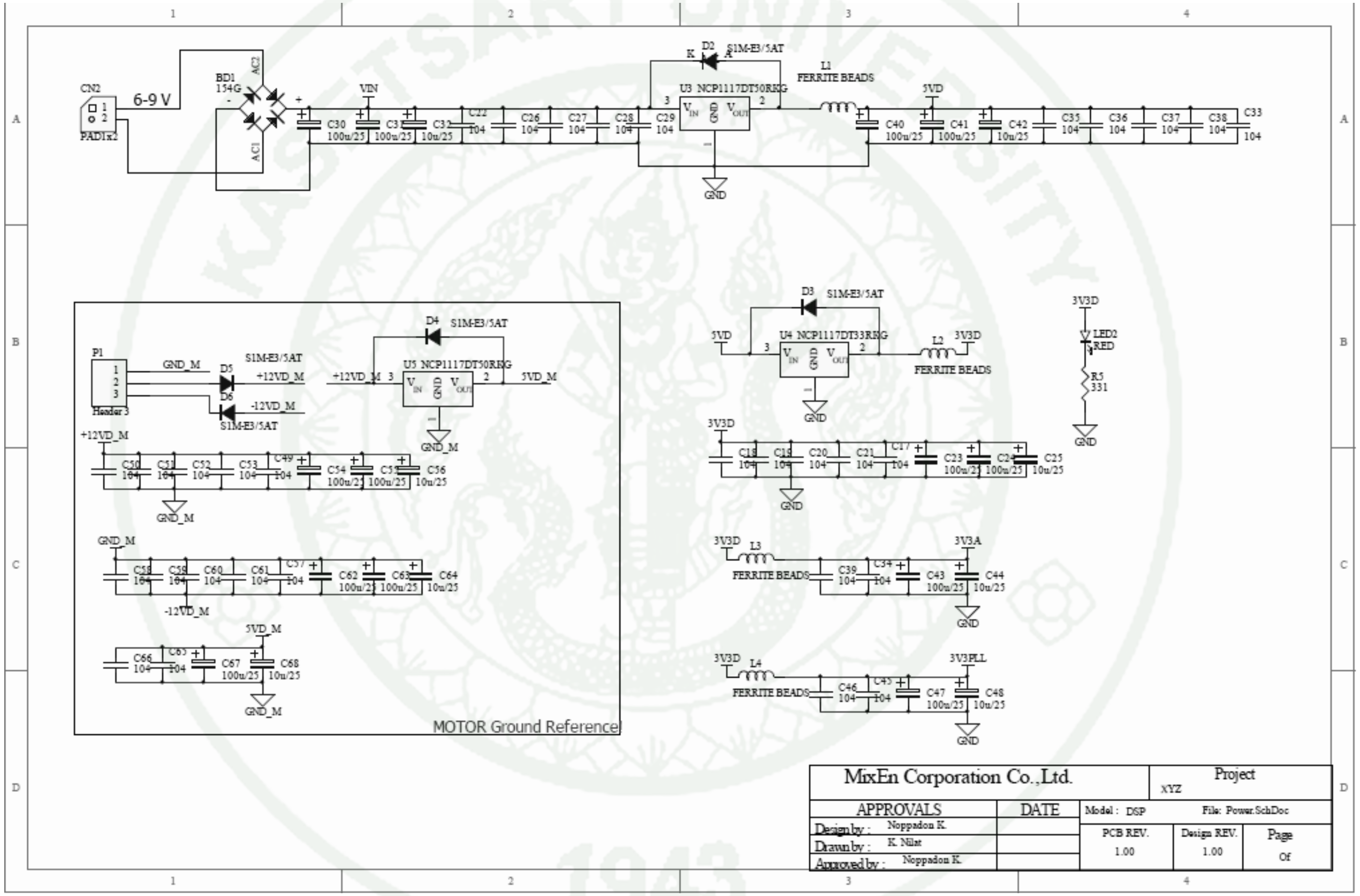
Appendix Figure A2 Digital to analog schematic



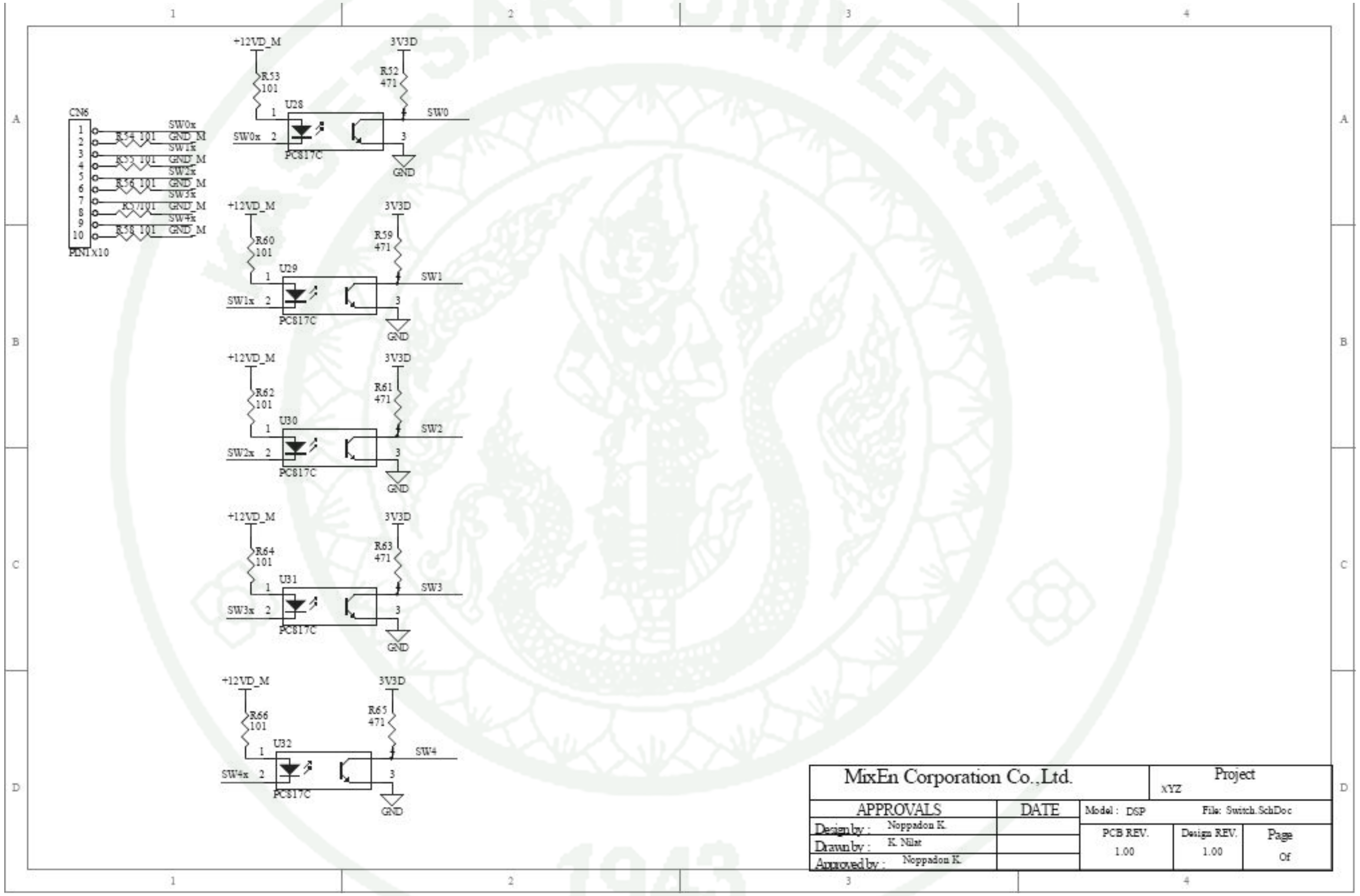
Appendix Figure A3 Digital signal processor schematic



Appendix Figure A4 Encoder input schematic



Appendix Figure A5 Power supply schematic



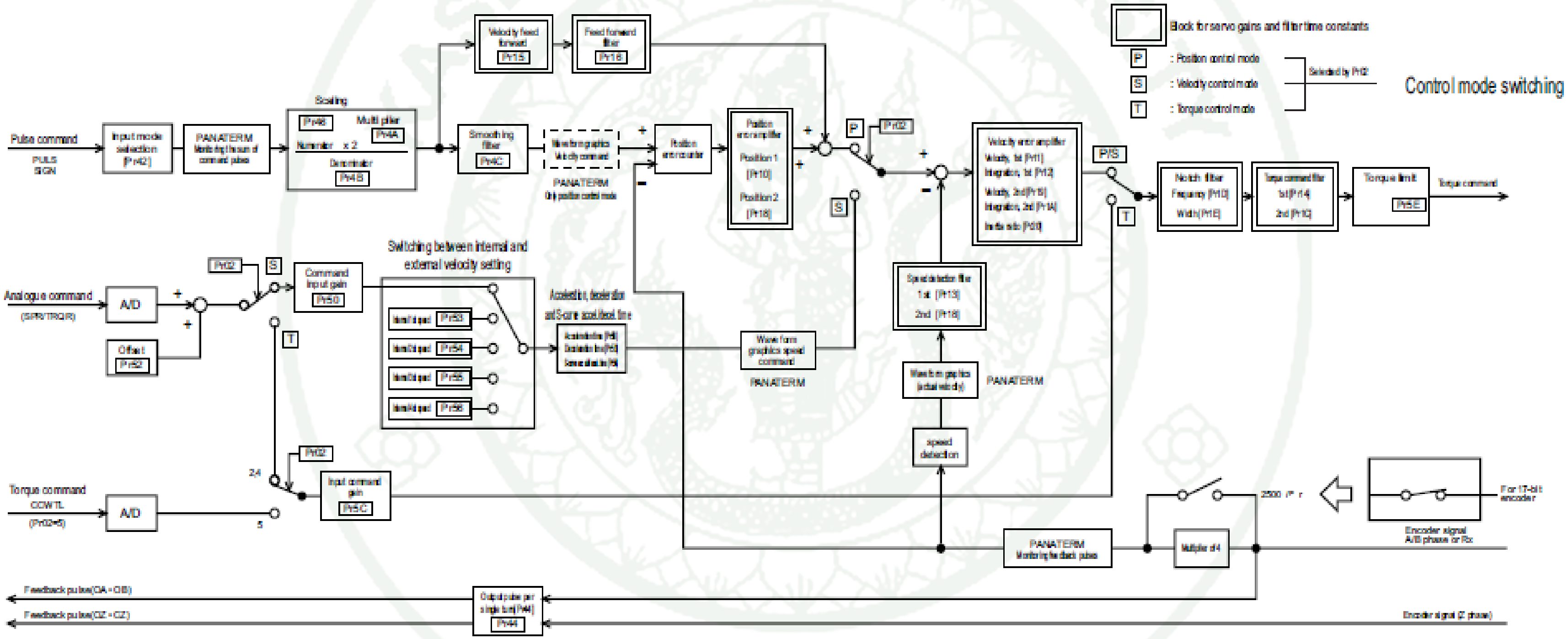
Appendix Figure A6 Digital input schematic

Appendix B

Motor and motor driver specification

Appendix Table B1 Motor and Driver Specification

Detail	Value	Unit
Servo Motor Product No.	MSMA082A1C	-
Rated Output	750	*Watt*
Rated Torque	2.4	*Nm*
Rated RPM	3000	*RPM*
Rated Current	4.3	*A*
Moment of rotor inertia	0.000131	$Nm/rad/s^2$
Brake	No	-
Oil Seal	Yes	-
Key Shaft	No	-
Quadrature Encoder	2500	*pulse/revolution*
Driver Product No.	MSDA083A1A	-



Appendix Figure B1 The control block diagram of the motor drive

CIRRICULUM VITAE

NAME : Mr. Wijin Suteeratanapirom

BIRTH DATE : February 5, 1984

BIRTH PLACE : Bangkok, Thailand

EDUCATION :

YEAR	INSTITUTE	DEGREE/DIPLOMA
2006	Kasersart Univ.	B.Eng. (Electrical Engineering)
2010	Kasetsart Univ.	M.Eng. (Information and Communication Technology for Embedded Systems)

POSITION/TITLE : -

WORK PLACE : -

SCHOLARSHIP/AWARDS : TAIST ICTES Master Degree Scholarship