

# Analysis of Control System for A 24V PM Brushed DC Motor Fitted with an Encoder by Supplying H-Bridge Converter

Hla Myo Tun, Win Aung

**Abstract**—This project examines the Flexible Inverter (3-phase inverter) that can be controlled in dissimilar kinds of modes such as fixed duty cycle drive (not connected to a load), sinusoidal current generator (connected to star connected impedance) and brushless DC motor drive (connected to a 3-phase brushless DC motor). The use of high-quality power electronic device can acquire many advantages in the applications especially in power conversion. The flexible inverter is one of the most accepted inverter to utilize in the high power electronics applications. In this project, it demonstrates several benefits of power converter by using DSPic. Because of its physical characteristics, Flexible Inverter improves the converter efficiency and reduces the size of the converter. The project investigates the control of a 24V PM brushed DC motor fitted with an encoder by supplying H-bridge converter using PIC microcontroller. However people often deem the microcontrollers as not appropriate for power environment because of their inadequate computing speed or noise immunity problems. This analysis is completed by using C language and practical circuit. This analyze confirms that how the speed control of brush DC motor and the current loop feedback control for flexible inverter system. So, flexible inverter system is more suitable for power electronics drive circuits which need high demand of efficiency and speed control of DC motors. Moreover, the simulation results of the flexible inverter with two different input supplies are presented. The first one is the variable DC supply, the range between 0-24 V, with different duty ratios. The other is the variable AC supply with fixed reference input. The simulation results are evaluated by the C language and the practical results are measured by the Align Oscilloscope and then these data are deliberated by using MATLAB program to get the exact.

**Index-terms**— PM Brushed DC Motor; Control System; H-Bridge Converter; Flexible Inverter, DSPic

## I. INTRODUCTION

The flexible inverter board is one of the most popular power electronic drive system for brushed DC motors. In this Flexible Inverter testing, the inverter operates in three diverse kinds of modes. They are (a) Fixed duty cycle drive (not connected to any load), (b) Sinusoidal current generator (connected to star connected impedance) and (c) Brushless DC motor drive (connected to a 3 phase brushless DC motor).

Hla Myo Tun, Department of Electronic Engineering, Mandalay Technological University, Mandalay, The Republic of the Union of Myanmar, Wing Aung Myanmar Aerospace Engineering University, Meikhtila, The Republic of the Union of Myanmar, Email: [kohlamhotun@gmail.com](mailto:kohlamhotun@gmail.com). Manuscript received January 02, 2014; revised July 12, 2014 and September 17, 2014.

So, there is a lot interest by power electronics drive circuit although it could be available commercially just a few years ago. There are also a lot of scientific research papers scrutinizing the brushed DC motor speed control according to not only to its physical description but also its applications circuits, which could provide high efficiency and other reimbursements.

The aim of this project is to evaluate how to exploit the simulation of MATLAB, which is the most powerful software for electrical and electronics engineering subject. In the experimental test, the developed inverter will be used to generate PWM, and it would be concerned in three phase sine wave generator and brushless DC motor control. Moreover, the inverter will be approved in MATLAB simulation (inverter acts as DC power supply and AC power supply). These tests will be completed in the flexible inverter due to its ease of synthesis configuration to acquire the strict results of speed control and that converter efficiency. The tests will be executed by famous real-time C language and MATLAB software.

The motivation behind choosing the brushed DC motor is that the motor efficiency is increased, motor noise is reduced: the 100-Hz torque ripple is reduced because of the motor current ripple reduction; and the switching frequency is quasi-inaudible and inverter application. It is surprising that the performance of the whole control system and efficiency is related with the speed control process. The superior drive description of flexible inverter system can accomplish this obligation.

Brushed DC motors are extensively used in applications ranging from toys to push-button adaptable car seats. Brushed DC (BDC) motors are economical, uncomplicated to drive, and are voluntarily available in all sizes and shapes. Open-loop operation of the motor can be unsatisfactory in some industrial applications. If the drive requires constant-speed operation under changing load torque, closed-loop control is necessary. The dynamic response of the brushed DC motor angular velocity control depends on the designed control law. A high performance motor drive system must have excellent dynamic responses, although the motor parameters are time shifting. The development of new technologies for DC motor control such as PID, optimal, robust, and other control laws have been proposed in many applications [1]–[7]. Dimitri Mahayana has proposed control law based on robust indirect adaptive control can control angular velocity of brushed DC motor, although the parameters of system are time varying [8]. These high performance control laws depend on the operation conditions. Eaaichoon has discussed about the DC motor speed control using microcontroller PIC 16F877a. It was a closed-loop real time control system, where optical encoder

is coupled to the motor shaft to provide the feedback speed signal to controller [9]. Thierry Castagnet and Jean Nicolai have conversed how a cost effective digital motor drive could be designed by combining a chopper and an 8 bits microcontroller: the speed was simply controlled through direct voltage compensation and motor power limitation [11]. Muñoz-César, J.J et al, have presented the implementation of a conventional PID and a fuzzy PI control schemes in a 16 bits microcontroller. These schemes were designed for the speed control of a DC brush motor [12].

Even though the speed control of brushed DC motor is claim to be easy, however for those who are new in the field will still have a hard time in understanding it let alone in implementing it. Thus, in this project we discuss the control of a 24V PM brushed DC motor fitted with an encoder by supplying H-bridge flexible inverter using DSPic and provide a fundamental feedback control framework. However, the discussion is only limited to the fundamental of brushed DC motor, extensions of speed control of a brushed DC motor such as position control and other are not discussed in the report. Implementation and works of the project are summarized into the flow chart as shown in Fig.1.

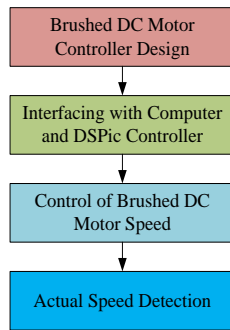


Fig.1. Project Overview

## II. REQUIRED MATERIALS AND DRIVE TECHNIQUES

### A. Investigation of Brushed DC Motor

The geometry of the brushes, commutator contacts, and rotor windings are such that when power is applied, the polarities of the energized winding and the stator magnet(s) are misaligned, and the rotor will rotate until it is almost aligned with the stator's field magnets. As the rotor reaches alignment, the brushes move to the next commutator contacts, and energize the next winding. To perform the simulation of a system, an appropriate model needs to be established. For this paper, the system contains a DC motor. Therefore, a model based on the motor specifications needs to be obtained.

Applying a constant stator current and assuming magnetic linearity, the basic motor equations are

$$T_m = K I \quad (1)$$

$$e_a = K_m \omega \quad (2)$$

Let the switch SW be closed at  $t = 0$ . After the switch is closed,

$$V_t = e_a + I_a R_a + L_{aq} \frac{di_a}{dt} \quad (3)$$

From Equation (2) and (3)

$$V_t = K_m \omega + I_a R_a + L_{aq} \frac{di_a}{dt} \quad (4)$$

$$T_m = K_m I_a = J \frac{d\omega}{dt} + B\omega + T_L \quad (5)$$

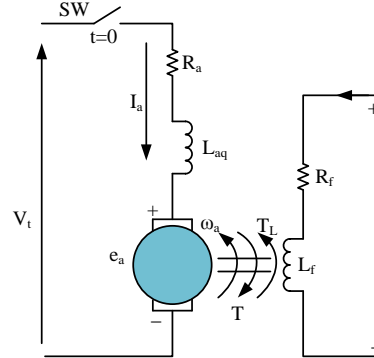


Fig.2. Schematic Diagram of a Brushed DC Motor

The term  $B\omega$  represents the rotational loss torque of the system. The state-space representation is given by the equations:

$$\dot{x} = Ax + Bu \quad (6)$$

$$y = Cx + Du$$

where

$$x = \begin{bmatrix} \omega \\ i_a \end{bmatrix}, A = \begin{bmatrix} -B/J & K_m/J \\ -R_a/L_{aq} & -K_m/L_{aq} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & -1/J \\ 1/L_{aq} & 0 \end{bmatrix}, u = \begin{bmatrix} V_t \\ T_L \end{bmatrix}$$

$$C = [1 \ 0], D = [0]$$

The transfer function between angular velocity and voltage is

$$\frac{\omega}{V_t} = \frac{A_v}{s^2 + B_1 s + B_0} \quad (7)$$

The transfer function between angular velocity and Load Torque is

$$\frac{\omega}{T_L} = \frac{A_L}{s^2 + B_1 s + B_0} \quad (8)$$

where

$$A_v = \frac{K_m}{J L_{aq}}$$

$$A_L = -\frac{K_m}{J^2}$$

$$B_1 = \frac{B}{J} + \frac{K_m}{L_{aq}}$$

$$B_0 = \frac{(B + R_a) K_m}{J L_{aq}}$$

In most DC motors, the rotor inductance and the value of  $B$  are small that can be neglected to lead to reduced order. If  $L_{aq}$  is neglected then Equation (4) becomes

$$V_t = K_m \omega + I_a R_a \quad (9)$$

If B is neglected then Equation (5) becomes

$$T_m = K_m I_a = J \frac{d\omega}{dt} + T_L \quad (10)$$

The current-voltage relationship for the left hand side of the equation can be written and manipulated to relate between voltage and angular velocity.

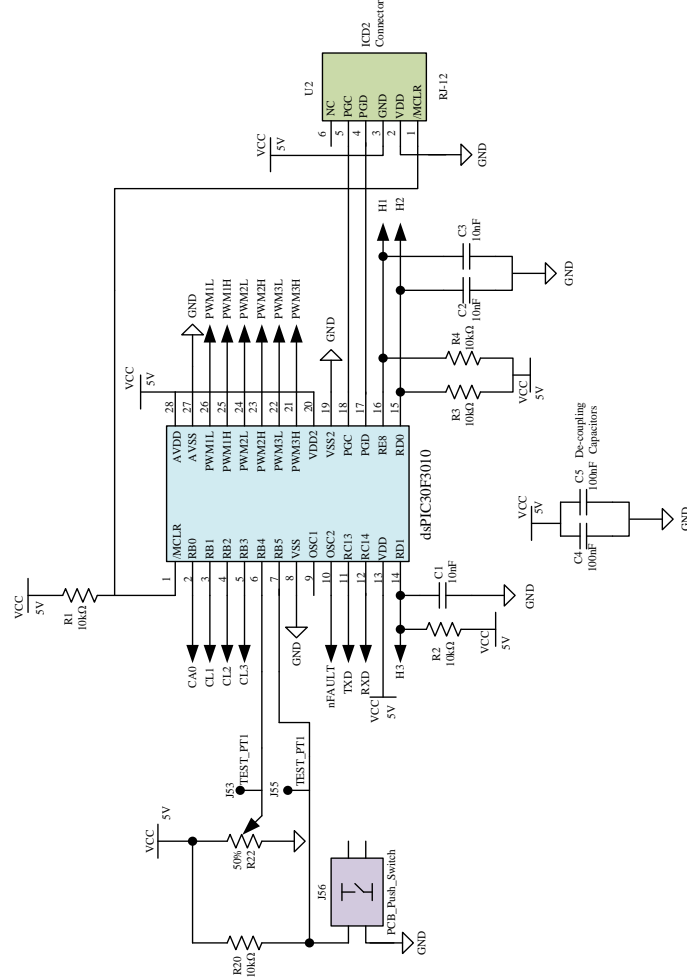


Fig.3. PIC Controller and Ancillary's Schematic Diagram

$$I_a = \frac{V_t - e_a}{R_a} \quad (11)$$

$$\frac{T_m}{K_m} = \frac{V_t - K_m \omega}{R_a} \quad (12)$$

$$\frac{J \frac{d\omega}{dt} + T_L}{K_m} = \frac{V_t - K_m \omega}{R_a} \quad (13)$$

$$\frac{d\omega}{dt} + \left( \frac{K_m^2}{J R_a} \right) \omega = \left( \frac{K_m}{J R_a} \right) V_t - \frac{T_L}{J} \quad (14)$$

## B. Flexible Inverter System

There are four main sections to implement the flexible inverter system. They are PIC controller and ancillary's schematic diagram, MOSFET driver and ancillary's schematic diagram, voltage regulator and ancillary's schematic diagram and 3-phase MOSFET inverter ancillary's schematic diagram.

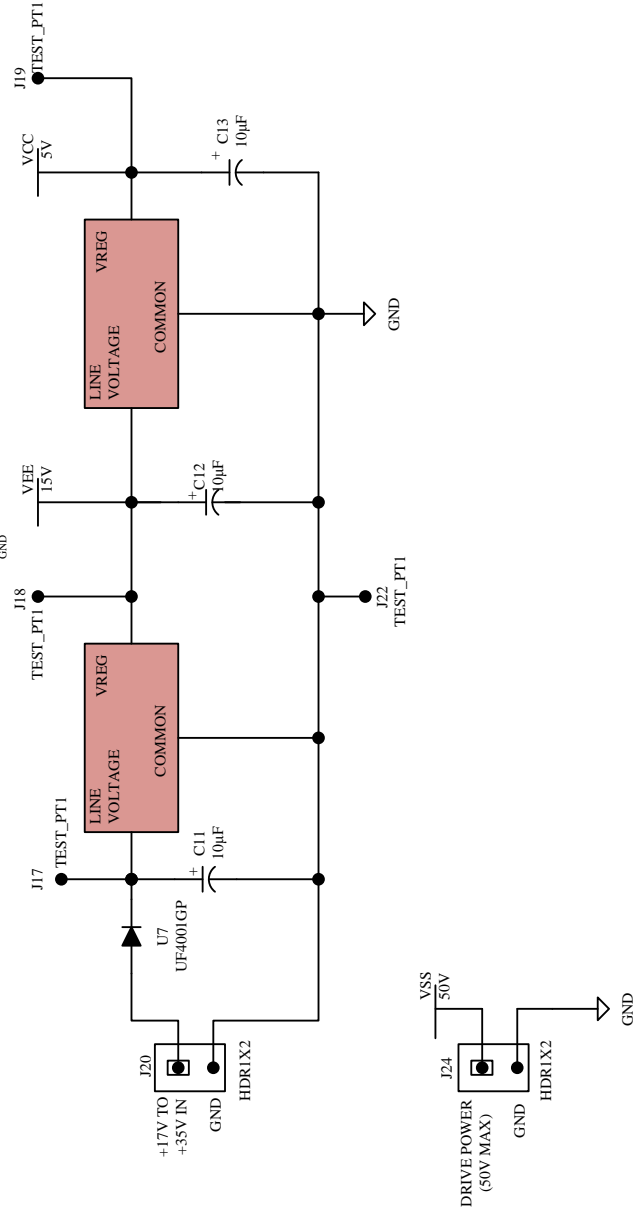


Fig.4. Voltage Regulator and Ancillary's Schematic Diagram

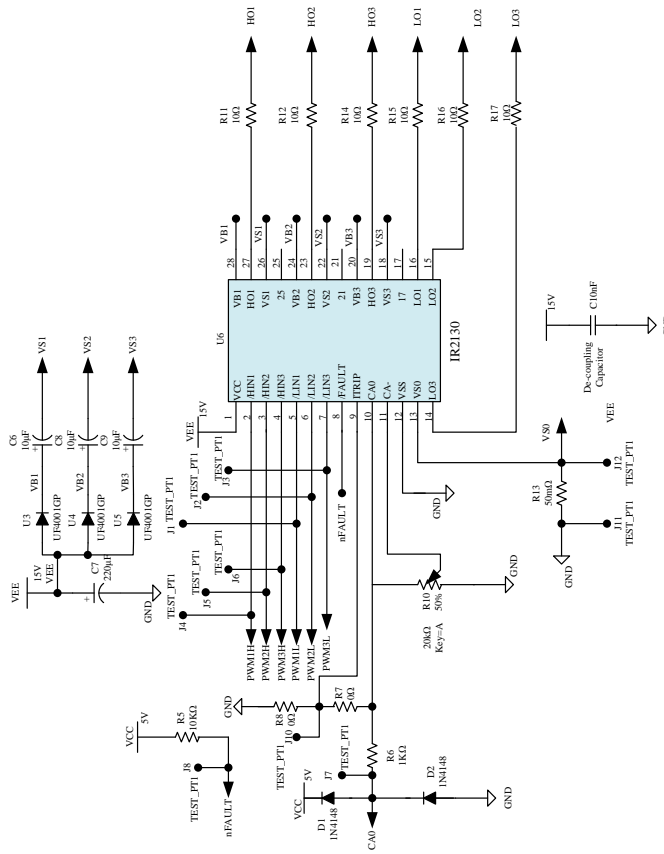


Fig.5. MOSFET Driver and Ancillary's Schematic Diagram

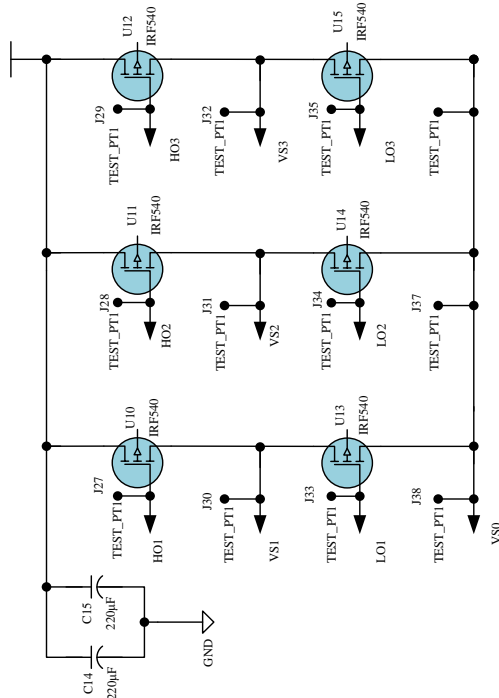


Fig.6. 3-phase MOSFET Inverter Ancillary's Schematic Diagram

The flexible inverter board could be implemented by using above sections. The implemented flexible inverter board is mentioned in Fig.6. The MOSFET switches are very proficient devices in flexible inverter system.

MOSFET is voltage-controlled device, so the DC input resistance is very high. IRF540 MOSFETs are very suitable to implement the inverter section. The component lists for flexible inverter design is given in Table .1.

TABLE I. COMPONENT LISTS FOR FLEXIBLE INVERTER SYSTEM

Components Name	Value	Remarks
MOSFET (U10 to 15)	IRF540	
C14, C15	220 $\mu$ F	
C11, C12, C13	10 $\mu$ F	
U8, U9	LM7805C7	
U7	UF4001GP	
U3, U4, U5	UF4001GP	
C6, C8, C9	1 $\mu$ F	
C7	220 $\mu$ F	
R21	560 $\Omega$	
C10, C26	100nF (decoupling Capacitors)	
R6	1k $\Omega$	
D1, D2	1N4148	
R11, R12, R14, R15, R16, R17	10 $\Omega$	
R13	50 $\Omega$	
R25	39k $\Omega$	
R26	10k $\Omega$	
R2, R3, R4	4.7k $\Omega$	
R9, R19, R18	1k $\Omega$	
R1	10k $\Omega$	
C1, C2, C3	1nF	
R20, R22	10k $\Omega$	
C16	100nF	

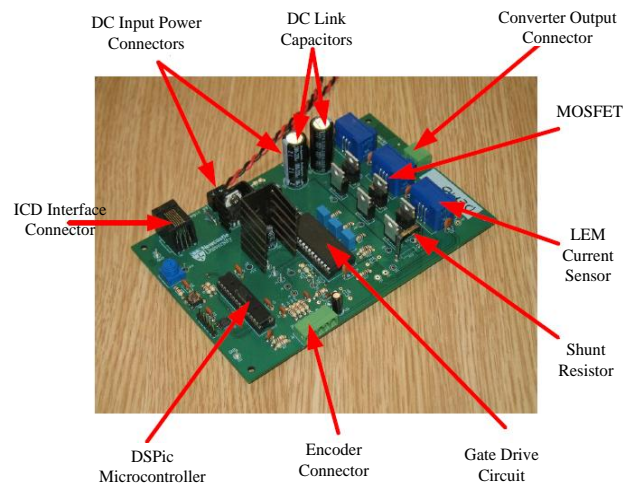


Fig.7.Implemented Flexible Inverter Board

### C. Microcontroller Operation

The dsPIC30F3010 microcontroller is used to employ the digital control. In order to do this the microcontroller

must obtain signals from the armature current sensor and the motor position encoder. The microcontroller then processes these signals and generates suitable PWM output signals to switch the H-bridge MOSFETs. The analogue signal fashioned by the current sensor is provided for to an Analogue to Digital Converter (ADC) which is a sub-system within the microcontroller. An illustration viewing the functions within the microcontroller and how they relate to other devices on the Flexible Inverter Board is shown in Fig.8.

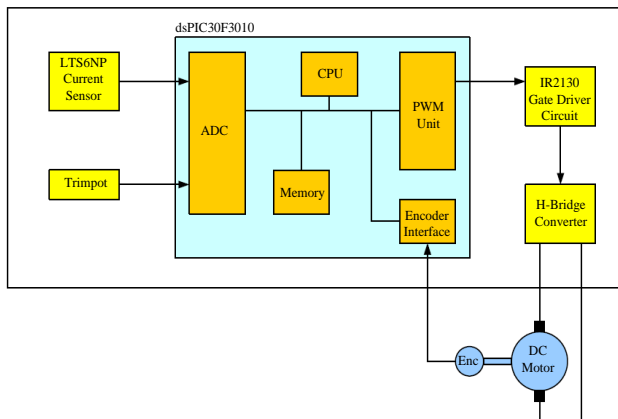


Fig.8. Functions on Flexible Inverter Board

### III. IMPLEMENTATION

In this development, the dsPIC microcontroller is used to simulate the speed control of brushed DC motor with the help of flexible inverter is controlled to the position control system. The project consists of three main parts, dsPIC microcontroller, the brushed DC motor, and the flexible inverter. The dsPIC which is used to control the speed of brushed DC motor and the output is the current component which is used in the speed control situation. The speed control of brushed DC motor consists of twenty steps, which in every time step the speed of brushed DC motor will be adjusted by changing the situation of the experimental set-up. The user can interact both with the speed control and the current component. For example, the user can alter the upcoming step to control the speed of brushed DC motor which is used as input to the next step. The information from the speed of brushed DC motor is necessary since the dsPIC can control the previous speed condition of that motor.

The motor to be controlled has an optical encoder built-in. The other DC machine is used as a load machine and this has a tacho-generator built-in. A tacho-generator generates an output DC voltage proportional to speed. This can be hardened to monitor the speed of the machine using a digital voltmeter or an oscilloscope. The Flexible Inverter Boarden closes a 3-phase MOSFET bridge power circuit, gate drive circuit, 3 current sensors and a dsPIC30F3010 microcontroller.

The H-bridge is fashioned by using two of the three MOSFET legs. The connection diagram is illustrated in Fig.9. The motor armature winding will be connected between terminals A and B on converter output connector (J13). To allow the microcontroller to monitor the armature

current, an LTS6NP current sensor (U19) must be fitted with decoupling capacitor C22.

The dsPIC30F3010 microcontroller is programmed using the C high-level language. The manufacturer of the microcontroller provides software development tools to write modify program and run that programs in the microcontroller. The software development tool for this type of microcontroller is called MPLAB. MPLAB runs on PC and it communicates with the Flexible Inverter Board via an In-Circuit-Debugger (ICD). The ICD allows the developed programs to be downloaded into the microcontroller and also provides control and monitoring features to debug the program code.

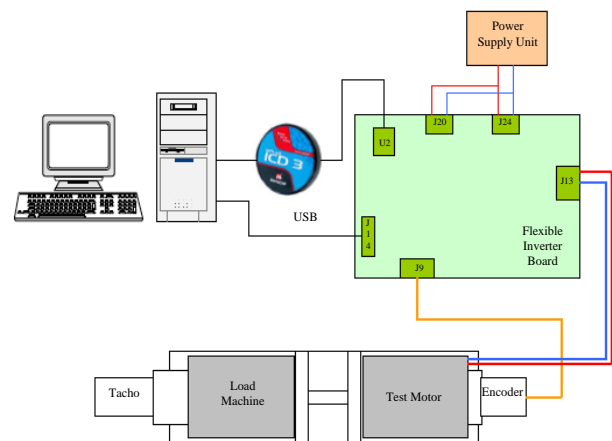


Fig.9.Connection Diagram

### IV. SYSTEM FLOWCHART

The overview of the program flow is shown on the Fig.10 on the next page. The simulation is implemented in C language by using Microsoft Visual C++ 2008 Express Edition. As for the simulation results we use MATLAB together with SIMULINK for the simulation programming. It is very significant to advance the construct of an experimental control system in a series of steps.

Step one to three is detailed procedures for visit of cascade speed control SDL SIMULINK assignment to improve the control problem and C language. Step four is to write a C program to read in a value from the ADC input which has been set of connections for input RB4. The input voltage can be mottled with trimpot R22. Step five is to add program code to use the ADC input to rest the modulation index of output PWM1L/1H and monitor PWM1L on the oscilloscope to verify that the adjustment of the duty ratio of the PWM channel using R22. Step six is with a DC supply connected to the MOSFET Bridge, to check that the MOSFETs U10 and U13 are switching by monitoring waveform at test point J30. Step seven is to substantiate comparable operation for the second leg (i.e. MOSFETs U11 and U14). Step eight is to write code for unipolar PWM.

Step nine is to prove unipolar switching by monitoring test points J30 and J31. Step ten is to connect RL load across H-bridge outputs (i.e. J13 pins A and B) and validate that a DC current flows and the current magnitude can be

proscribed using the trimpot. Step eleven is for a particular modulation index procedure the average and ripple current level in the RL load and to modify the simulation to imitate the RL load and evaluate the results. Step twelve is to write code to sample the armature current and display it on LabVIEW panel and to calibrate the current sensor. Firstly the ADC reading for zero current is established by removing the RL load. Secondly the RL load is re-connected and an ADC reading for a particular current is recognized. These values can be used to set offset and gain values in developed code to standardize the current sensor input samples.

Step thirteen is to remove RL load and connect armature of DC motor and to confirm that motor can be controlled by changing the H-bridge average output voltage with the trimpot. The output voltage from tacho-generator (attached to the load machine) with a DVM is monitored. Step fourteen is to stimulate code to tolerate monitoring of the encoder position on LabVIEW panel and calibrate the output from the tacho-generator against the speed (i.e. volts/RPM). Step fifteen is to write code to implement a PI controller for the armature current loop. In this code there is included to prevent integrator windup and test the current loop with the RL load. A demanded current can be input via LabVIEW panel. Step sixteen is to amend code to create 1s pulses of armature current demand and tune PI controller to acquire an acceptable armature current step response.

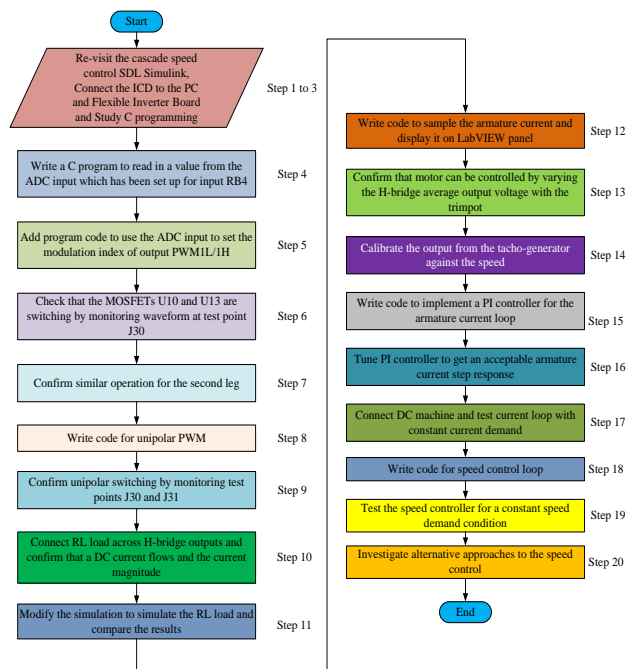


Fig.10. Program Flow

Step seventeen is to connect DC machine and test current loop with constant current demand. As there is no speed controller at this stage the machine will run open loop from the point of view of speed control. This means that a relatively small armature current can produce very high speeds. By connecting an appropriate load resistance across the armature of the load machine it is possible to generate a load torque. This will bound the speed when testing devoid of a speed control loop. Step eighteen is to write code for speed control loop and tune the PI controller to get a suitable step response. Step nineteen is to test the speed controller

for a constant speed demand condition. And it should be subjected the DC machine to a load transient using the load machine and evidenced the speed transient which results and performed further tuning to minimize the magnitude of the speed transient. Final step is to investigate substitute approaches to the speed control.

## V. PROGRAMMING THE FLEXIBLE INVERTER BOARD WITH C PROGRAMMING

The Flexible Inverter Board encloses a dsPIC30F3010 microcontroller. Before programming it is significant to realize the essential peripheral hardware in the dsPIC30F3010 which is used on the Flexible Inverter Board. The C program obtains input data from the Analogue to Digital Converter (ADC) and an Optical Encoder Interface. The C program propels output data to a Pulse Width Modulation (PWM) unit. There is also a cooperative communications interface which permits the Flexible Inverter Board to communicate with a PC. This communication interface is called a Universal Asynchronous Receiver Transmitter (UART). The data flow between the peripheral hardware elements and the C program are shown in Fig.11.

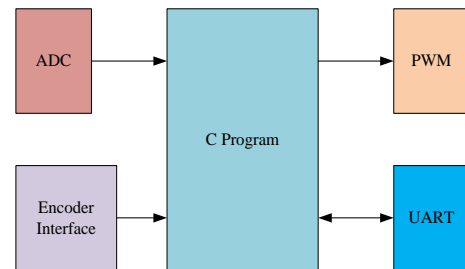


Fig.11. Relationship between Peripheral Hardware and C Program

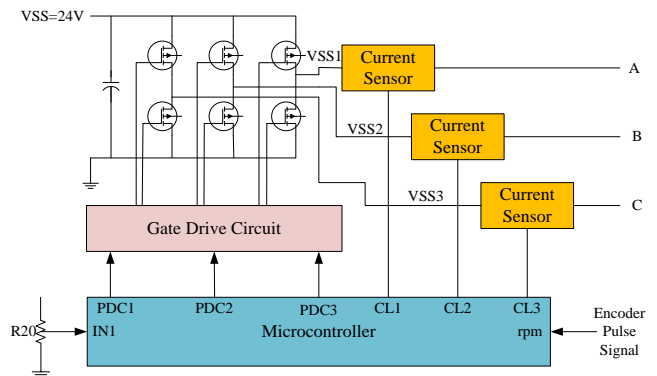


Fig.12. Microcontroller Input and Output Signals

When using a microcontroller it is essential to perform a number of initialization operations before the precise application control code is accomplished. The various peripheral devices such as the ADC, encoder interface logic, UART communications and PWM are compulsory to be set up for explicit modes of operation. It is also necessary to set up the interrupt system so that the sampling of various inputs is finished with the correct timing. The frame arrangement includes various initialization functions and a number of interrupt service routines (ISR). The diagram in Fig.12 demonstrates the signals between the MOSFET inverter, current sensors, trim-pot, encoder and the



microcontroller. The signal labels exposed in the microcontroller block are the names of the variables used in the C program to access the signals revealed.

## VI. RESULTS AND DISCUSSION

According to the experimental set-up, the speed control of brushed DC motor could be accomplished with the help of dsPIC30F3010 microcontroller. There are twenty steps to implement the control of a 24V PM brushed motor fitted with an encoder. The first step through third step relies on the knowledge of the control problems and the programming language skills for control of brushed DC motor.

### A. Development of Step Four

This step is to communicate with a C program to read in a value from the ADC input which has been set of connections for input RB4. The input voltage can be mottled with trimpot R22. There are three conditions to analyze for mottled input voltage. Fig.13 shows mottled input voltage waveform for first condition. The on stage period is very large rather than the two analysis. In Fig.14, the on stage period is normal to compare with the first stage. At that time the mottled shapes are displayed on the scope. For the lowest on stage level (shown in Fig.15), the stippled shape is higher than the other two waveforms.

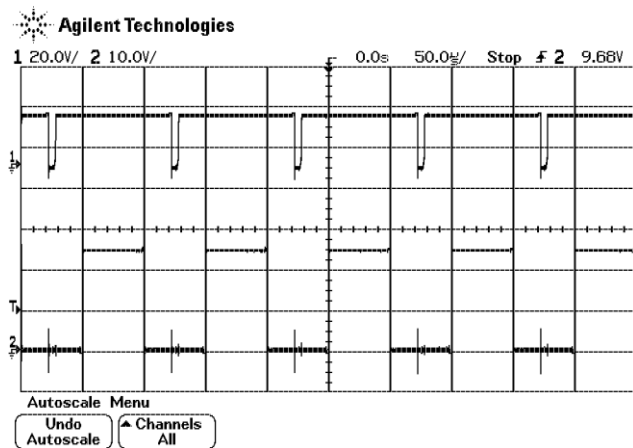


Fig.13. Mottled Input Voltage Waveform for First Condition

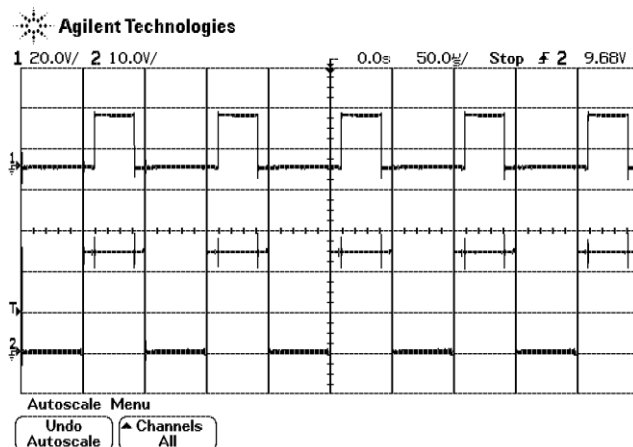


Fig.14. Mottled Input Voltage Waveform for Second Condition

### B. Development of Step Six

This step is with a DC supply connected to the MOSFET Bridge for checking the switching stage of MOSFETs U10 and U13 by monitoring waveform at test point J30.

### C. Development of Step Seven

This step is to substantiate comparable operation for the second leg (i.e. MOSFETs U11 and U14) of the inverter system. Fig.20 illustrates the comparison waveform of the switching stage of MOSFETs U11 and U14.

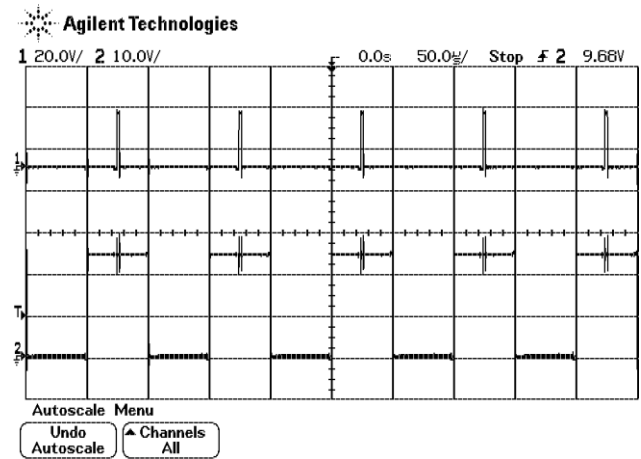


Fig.15. Mottled Input Voltage Waveform for Third Condition

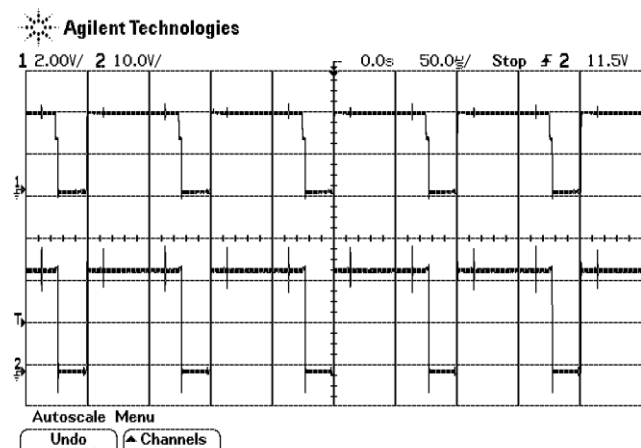


Fig.16. Comparison of the Switching Stage of MOSFETs U10 and U13

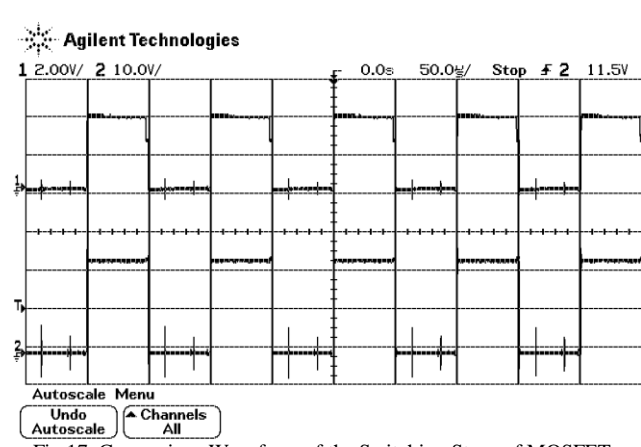


Fig.17. Comparison Waveform of the Switching Stage of MOSFETs U11 and U14

#### D. Development of Step Eight and Nine

The implementation of the step eight is to develop code for unipolar PWM to control the speed of brushed DC motor. The proof of unipolar switching by monitoring test points J30 and J31 is evaluated in step nine. Fig.18 shows the screenshot result of equal width of PDC1 and PDC2. The amplitude level at that situation is 23.8V. Fig.19 mentions the screenshot result of minimum width of PDC1. For this condition the amplitude of waveform is 24.4V level. Fig.19 illustrates the screenshot result of minimum width of PDC2. For the condition of the minimum width of PDC2, the amplitude of waveform is the same level as the result of minimum width of PDC1. Based on the three condition of on stage period, the shapes of the unipolar switching are evaluated by each test points to control the speed of the brushed DC motor system.

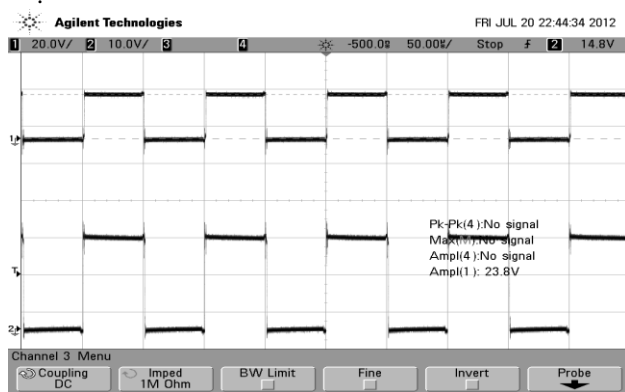


Fig.18. Screenshot Result of Equal Width of PDC1 and PDC2

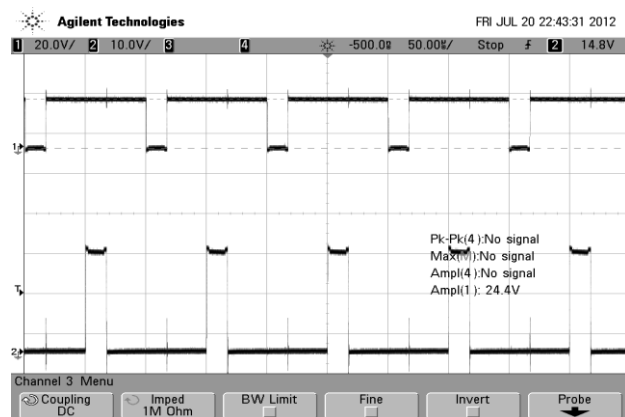


Fig.19. Screenshot Result of Minimum Width of PDC1

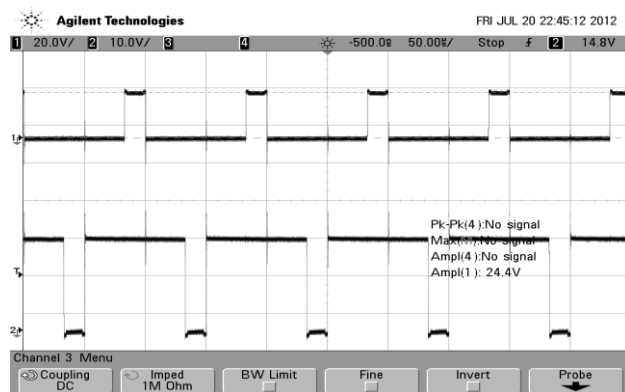


Fig.20. Screenshot Result of Minimum Width of PDC2

#### E. Development of Step Ten

This step is to analyze with the connection between RL load across H-bridge outputs (i.e. J13 pins A and B) and validation of DC current flows and the current magnitude can be proscribed using the trimpot. Fig.20 shows the screenshot results of maximum condition for current (PDC2 Wide). The mean value from the simulation result is 19.574 V. There are three stage conditions for analyzing the maximum condition of current for PDC1 and PDC2 and the equal condition of current for PDC1 and PDC2. Fig.21 mentions the screenshot results of maximum condition for current (PDC1 Wide). Fig.22 demonstrates the screenshot results of equal condition for current (PDC1=PDC2). The mean values from the simulation result are 20.631V and 22.271V respectively.

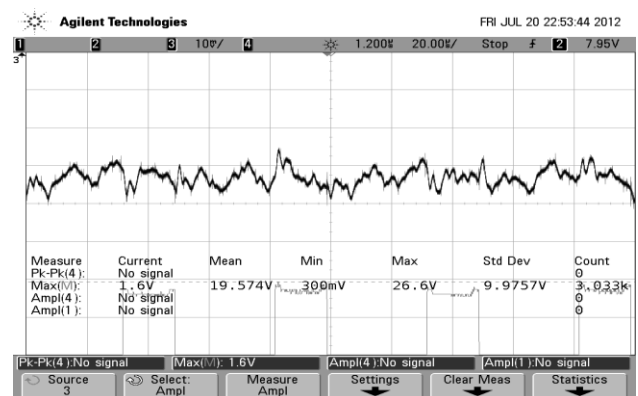


Fig.21. Screenshot Results of Maximum Condition for Current (PDC2 Wide)

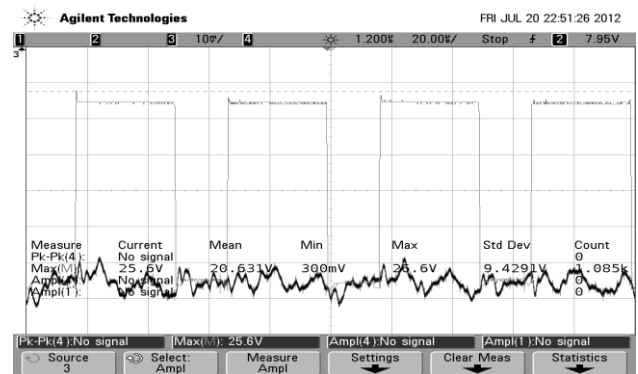


Fig.22. Screenshot Results of Maximum Condition for current (PDC1 Wide)

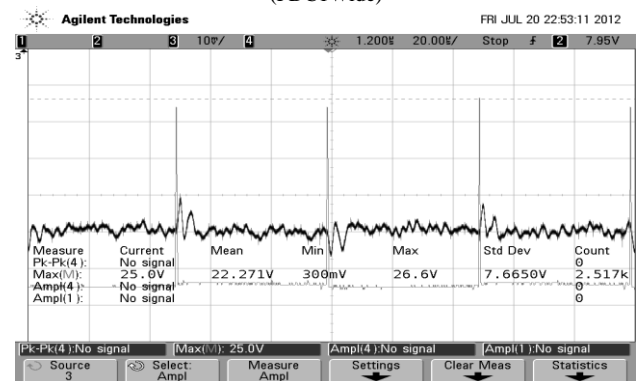


Fig.23. Screenshot Results of Equal Condition for Current (PDC1=PDC2)



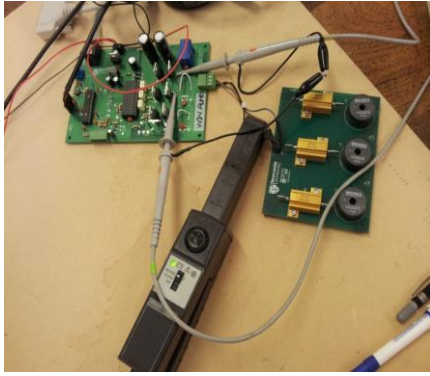


Fig.24. Screenshot Results of Experimental Test on Step Ten

#### F. Development of Step Eleven

This step is to modify the simulation to imitate the RL load and evaluate the results. Fig.24 illustrates the screenshot results of 82.1% PDC2 and 18.1%PDC1 condition.

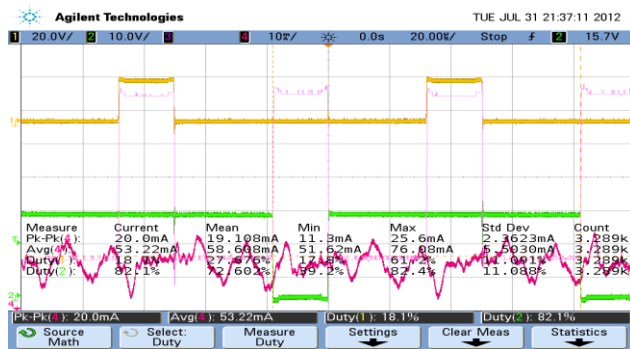


Fig.25. Screenshot Results of 82.1% PDC2 and 18.1%PDC1 Condition

Fig.25 shows the screenshot results of 82.9% PDC2 and 17.3%PDC1 condition and Fig.26 gives the screenshot results of ripple current.



Fig.26. Screenshot Results of 82.9% PDC2 and 17.3%PDC1 Condition

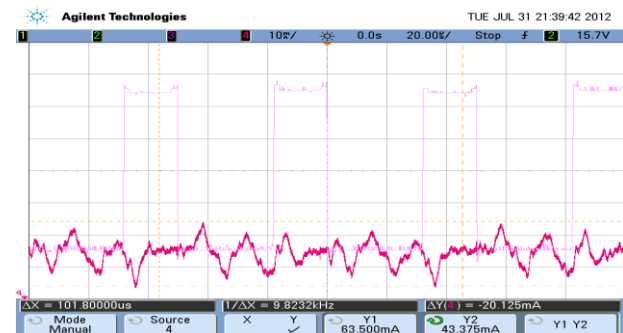


Fig.27. Screenshot Results of Ripple Current

For Ripple Current in Simulation, the following calculation is accomplished to analyze the ripple current of the output waveform.

$$X1 = 246.8, Y1 = 5.107$$

$$X2 = 174.2, Y2 = 5.082$$

$$Y2 - Y1 = 5.082 - 5.107 = -0.025 \text{ A (ripple current)}$$

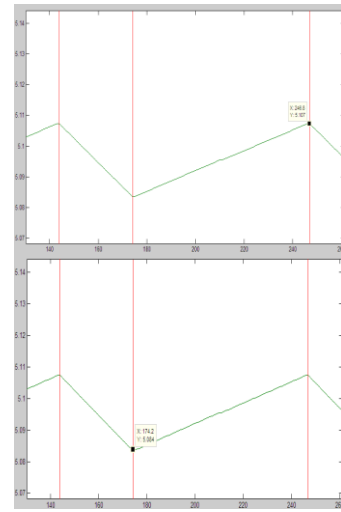


Fig.28. Ripple current in Simulation of the RL load

#### G. Development of Step Twelve

Firstly the ADC reading for zero current was recognized by removing the RL load. Secondly the RL load was re-connected and an ADC reading for a particular current is renowned. These values can be used to set offset and gain values to standardize the current sensor input samples.

Total Error = 66 bits

$$\text{Average bit} = \frac{\text{Total Errors}}{16} = \frac{66}{16} = 4.125$$

TABLE II. COMPARISON RESULTS FOR CURRENT ACROSS RL LOAD

No.	Current	Demand ADC value [(crurent/ 0.0156mA) + 511] bits	Actual ADC value (bits)	Error = Actual ADC value – Demand ADC value
1.	24.16 mA	513	519	6
2.	20.52 mA	512	517	5
3.	17.05 mA	512	517	5
4.	14.05 mA	512	517	5
5.	10.74 mA	512	516	4
6.	7.8 mA	512	515	3
7.	3.93 mA	511	514	3
8.	1.18 mA	511	514	3
9.	1.83 mA	511	514	3
10.	5.14 mA	511	514	3
11.	8.19 mA	512	515	3
12.	11.64 mA	512	517	5
13.	14.92 mA	512	517	5
14.	17.51 mA	512	516	4
15.	21.32 mA	512	515	3
16.	24.42 mA	513	519	6

### H. Development of Step Thirteen

This step is to monitor the output voltage from tachogenerator (attached to the load machine) with a DVM by removing RL load and connecting armature of DC motor and to confirm that motor can be controlled by changing the H-bridge average output voltage with the trimpot. Fig.29 shows the screenshot Results of PDC1 5% condition. At that time the motor speed is 0 rpm for PDC1 and PDC2 are 48 and 1426 respectively.

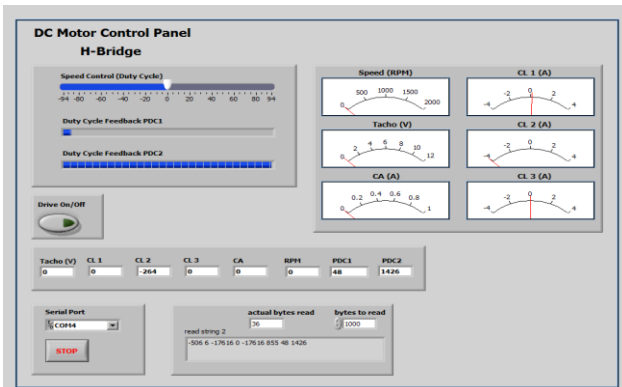


Fig.29. Screenshot Results of PDC15% Condition

Similarly, the screenshot results for PDC1 25%, 45%, 65%, 75% and 95% are mentioned with the appropriate panel shown in Fig.28 through Fig.32.

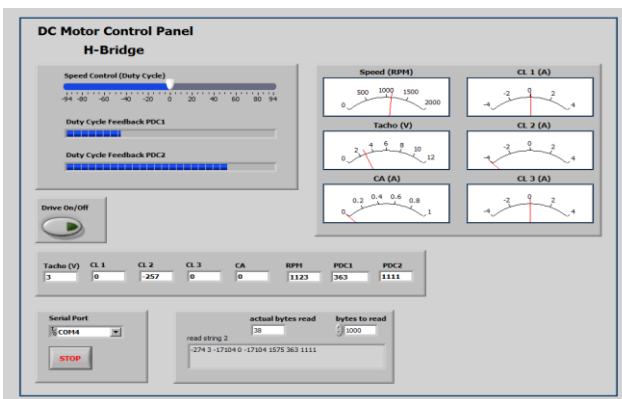


Fig.30. Screenshot Results of PDC125% Condition

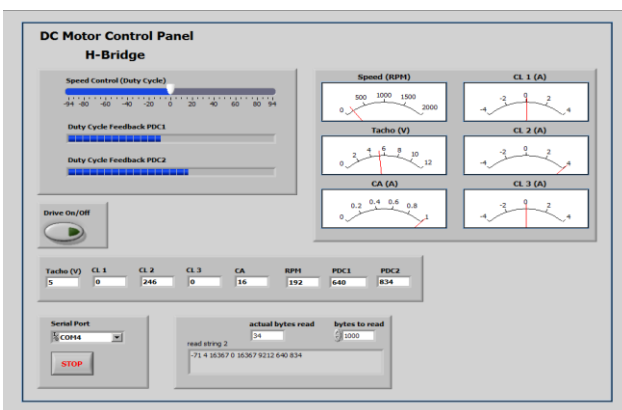


Fig.31. Screenshot Results of PDC1 45% Condition

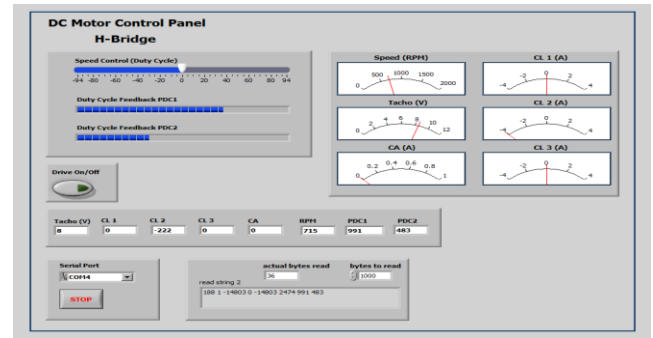


Fig.32. Screenshot Results of PDC1 65% Condition

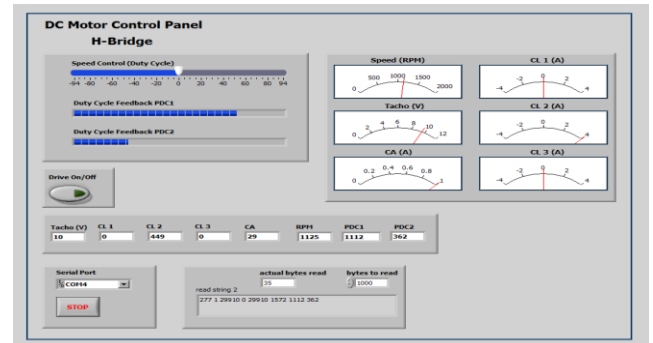


Fig.33. Screenshot Results of PDC1 75% Condition

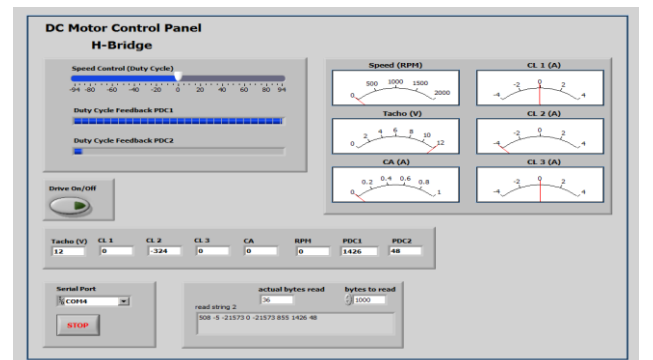


Fig.34. Screenshot Results of PDC1 95% Condition

The comparison results for output voltage from tachogenerator and speed with respect to duty ratio are given in Table 3.

TABLE III. OUTPUT VOLTAGE FROM TACHO-GENERATOR AND SPEED WITH RESPECT TO DUTY RATIO

S. No.	Duty Ratio		Output Voltage from Tacho-generator(V)	Speed (RPM)
	PDC1	PDC2		
1	95%	5%	21.22	0
2	93%	7%	20.36	1996
3	85%	15%	16.47	1615
4	75%	25%	11.72	1147
5	65%	35%	6.92	679
6	55%	45%	2.2	218
7	50%	50%	0	28
8	45%	55%	-1.8	185
9	35%	65%	-6.72	659
10	25%	75%	-11.56	1130
11	15%	85%	-16.45	1615
12	7%	93%	-20.38	1996
13	5%	95%	-21.3	0

### I. Development of Step Fourteen

This step is to analyze the tolerance monitoring of the encoder position on LabVIEW panel and calibrate the output from the tacho-generator against the speed. Table 5.4 shows the calibrated the output from the Tacho-generator against the speed.

TABLE IV. CALIBRATE THE OUTPUT FROM THE TACHO-GENERATOR AGAINST THE SPEED

	Duty Ratio (PDC1)	Output voltage from Tachogenerator(V)	Speed (RPM)
1	93%	20.36	1996
2	88%	17.94	1763
3	83%	15.58	1530
4	78%	13.34	1306
5	73%	10.62	1040
6	68%	8.34	820
7	63%	6.3	618
8	58%	3.65	360
9	53%	1.1	108
10	50%	0	28

According to the results table, the response of voltage vs speed is evaluated with the help of MATLAB. This response says the proportion of speed and voltage level more clearly.

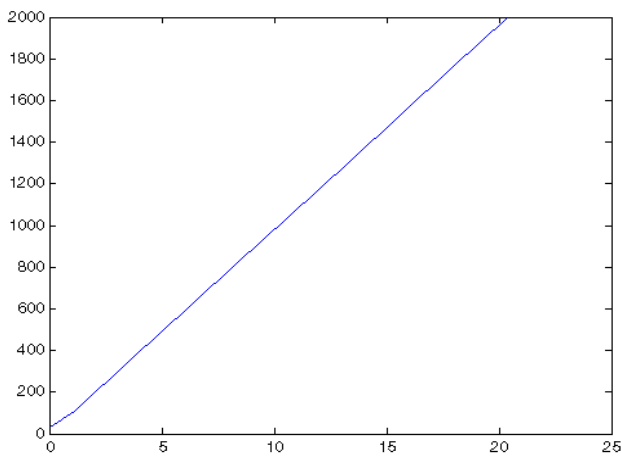


Fig.35. Response of Voltage Vs Speed

### J. Development of Step Sixteen

Step sixteen is to analyze the creation of 1s pulses of armature current demand and tuning PI controller to acquire an acceptable armature current step response. Fig.36 shows the response of initial condition for 1s pulse. The average value is 19.75mA. Fig.37 mentions the response of demand condition for 1s pulse. The average value is 23.18mA for demand condition. Fig.38 mentions the response of after demand condition for 1s pulse. The average value is 10.97mA for that condition. Therefore the average value of each condition is roll off of the initial condition.

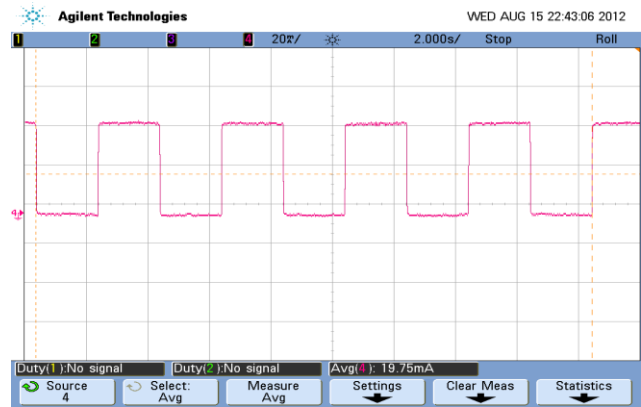


Fig.36. Response of Initial Condition for 1s Pulse

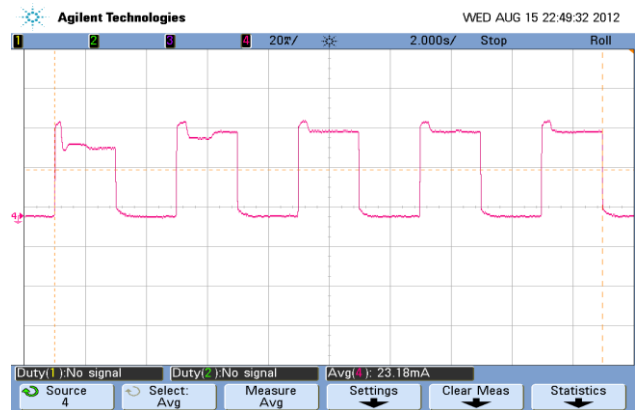


Fig.37. Response of Demand Condition for 1s Pulse

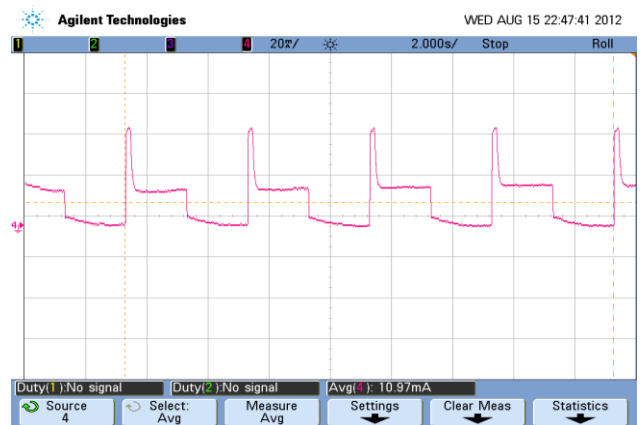


Fig.38. Response of After Demand for 1s Pulse

Set  $ia\_demand = 540$  bits,  
Change the bits value to current value(ampere),  
 $ia\_demand = 540 - 511 = 29 * 0.0156 = 0.45$  A

For Initial Condition

- capture(Oscilloscope) = 19.75 mA
- multi-meter = 0.1 A
- CL1(watch window) = 505 (511-505=6bits,  
 $6 * 0.0156 = 0.0936$  A)

For Demand Condition

- capture(Oscilloscope) = 23.18 mA
- multi-meter = 0.45 A
- CL1(watch window)= 543 (543 - 511 = 32bits,  $32 * 0.0156 = 0.49$  A)

For After Demand Condition

- capture(Oscilloscope) = 10.97 mA
- multi-meter = 0.28 A
- /CL1(watch window) = 512 (512 - 511 = 1bits,  
 $1 * 0.0156 = 0.0156$  A  $\sim 0$  A)

### K. Development of Step Seventeen

This step is to connect with DC machine and test the current loop for constant current demand in speed control. As there is no speed controller at this stage the machine will operate open loop from speed control situation. This means that a somewhat small armature current be able to create very high speeds value. By concerning a suitable load resistance across the armature of the load machine it is possible to spawn a load torque.

Fig.39 mentions the response of initial condition for current loop. Fig.40 shows the response of demand condition for current loop. Fig.41 demonstrates the response of after demand for current loop condition. The average current values are 40.07mA of current loop for initial condition, 41.47mA of current loop for demand condition, and 17.75mA of current loop for after demand condition respectively.

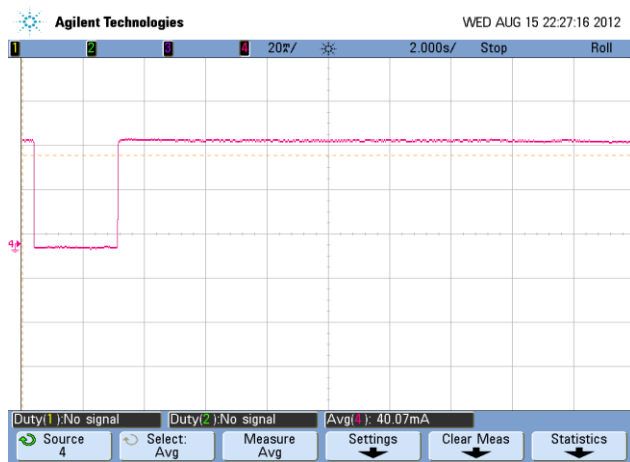


Fig.39. Response of Initial Condition for Current Loop

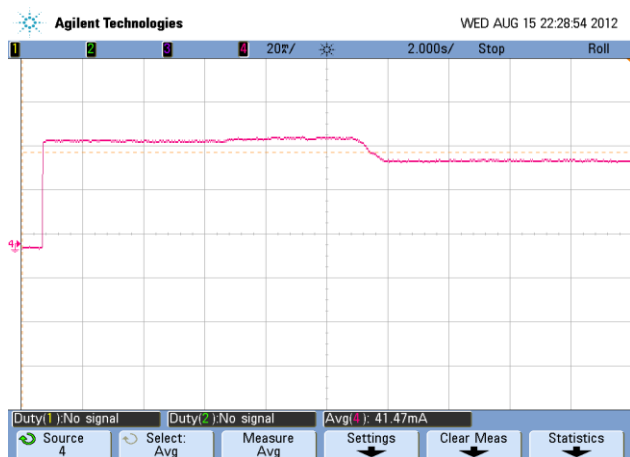


Fig.40. Response of Demand Condition for Current Loop

Set  $i_{a\_demand} = 540$  bits,

Change the bits value to current value(ampere),  
 $i_{a\_demand} = 540 - 511 = 29 * 0.0156 = 0.45$  A

For Initial Condition

- capture(Oscilloscope) = 40.07 mA
- multi-meter = 0.27 A
- CL1(watch window) = 542 (542 - 511 = 31bits,  $31 * 0.0156 = 0.484$  A)

For Demand Condition

- capture(Oscilloscope) = 41.47 mA
- multi-meter = 0.46 A
- CL1(watch window) = 540 (540 - 511 = 29bits,  $29 * 0.0156 = 0.45$  A)

For After Demand Condition

- capture(Oscilloscope) = 17.75 mA
- multi-meter = 0.3 A
- /CL1(watch window) = 527 (527 - 511 = 16bits,  $16 * 0.0156 = 0.25$  A)



Fig.41. Response of After Demandfor Current Loop

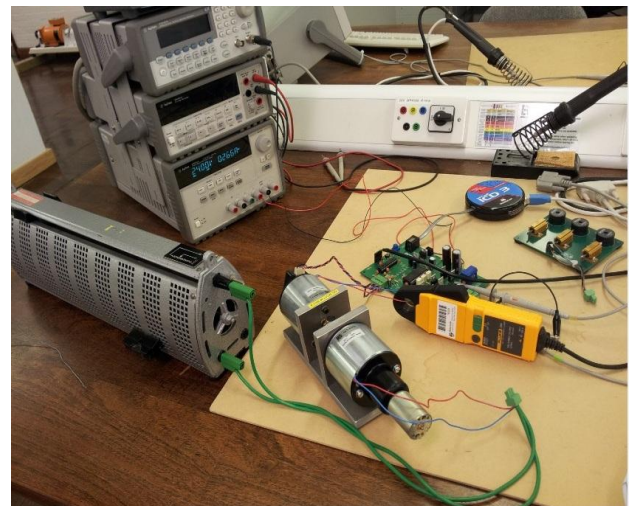


Fig.42. Photo of Load Test Condition

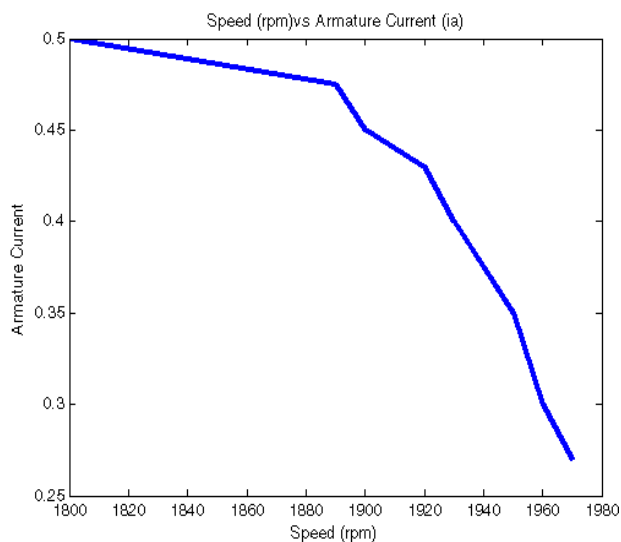
### L. Development of Step Eighteen and Step Nineteen

The final step is to tune the PI controller to obtain a suitable step response and to test the speed controller for a constant speed demand condition. And it should be subjected the DC machine to a load transient using the load machine and confirmed the speed transient which results and carried out further tuning to reduce the magnitude of the speed transient and to investigate substitute approaches to the speed control. Fig.43 illustrates the response of speed (rpm) with respect to armature current ( $I_a$ ).



TABLE V. PERFORMANCE EVALUATION OF BRUSHED DC MOTOR SPEED CONTROL

No.	Voltage	Current	Speed(rpm)
1.	24	0.5	1800
2.	24	0.475	1890
3.	24	0.45	1900
4.	24	0.43	1920
5.	24	0.4	1930
6.	24	0.35	1950
7.	24	0.3	1960
8.	24	0.27	1970

Fig.43. Response Of Speed (Rpm) With Respect To Armature Current ( $I_a$ )

The performance evaluation for speed control of brushed DC motor is given in Table 5. Based on this table, if the speed in rpm is higher condition, the current level will be lower than the previous condition. A control system of a speed control of brushed DC motor has been described in the dissertation. In the control scheme, the speed of the DC motor is controlled by flexible inverter. The laboratory setup will be used to evaluate the advanced drive components and control techniques which are presently being developed to meet the speed required. From these results it is apparent that the circuit depicted in this work is a practical, effective, high-frequency high power driver circuit.

The program does not necessitate much independent power, so it reduces the cost and size of drive circuit and drive the power MOSFET dependability used in the full-bridge inverter circuit called flexible inverter board. During the experiment, it is obvious that if the gate of MOSFET is floated and the voltage to the source is added, the MOSFET was turned on or even breakdown. The reason is that when the driver circuit doesn't work, the gate charge cannot be unconfined. It can be seen from the output waveform that the drive circuit can achieve a reliable MOSFET turn-off with good driving characteristics.

The duty ratio PDC1 is higher, the speed will be very fast and the voltage level is also advanced from step 14. The inner current loop of the cascade control system is tuned first before the outer speed loop is occupied. A good control is achieved by auto-tuning PI controller where there only a small transient deviation in speed when load torque is

applied at very low period, and settle to a demanded speed faster than the conventional PI. The performance evaluation for speed control of brushed DC motor is competent when the speed in rpm is higher condition, the current level would be lower than the previous condition. We have demonstrated the feasibility of brushed DC motors with high animated permanent magnet for low voltage applications. This method offers a cost effective way to control the speed of a Brushed DC Motor.

## VII. CONCLUSION

This control reduces the components count on the board because the microcontroller can integrate in one package all the functions of interface and motor control. The dsPIC plus MOSFET becomes a flexible and adaptable solution for power control in manufacturing system. The presented flexible motor control implementation concepts are confirmed to be highly accurate and effective by the close match that has been attained between simulation and experimental results using a flexible inverter board. The proposed module helps reducing the effort of the inverter construction, and it will contribute to increase the energy efficiency and the performance in the consumer application. The relationship between the motor speed and the current of the driver unit was explained with the figure and illustrated the detail reasons why flexible inverter system is suitable for high performance application circuits. In addition, the describing physical structure of control system and explained how the relationship between current and speed are influenced by its control scheme. The reason for control of brushed DC motor is described with the width of PDC1 and PDC2. The advantage of wider width is related with the higher speed rate for DC motor. The drawbacks of speed control is discussed with its operation current and voltage rating and explain the limitation for power electronics elements are related with its material properties. The evolution and invention of control of PM brushed DC motor using microcontroller are described briefly with the scientists and years. Comparison of various control methods are explained in literature review. Testing the speed control of brushed DC motor in different current values and plotting the respected response are evaluated with the help of MATLAB software. The high performance situation was proved. Explained the important of DC speed control is comparing with digital control techniques and other microcontroller based control techniques. Indicated how could get more efficiency in the speed control by using dsPIC microcontroller. The flexible inverter was designed to test the speed control of brushed DC motor. To reduce the passive components size and increase switching MOSFET efficiency, the flexible inverter is driven with dsPIC microcontroller for brushed DC motor.

In the simulation test, it could be seen more clearly that when the current is high the speed of DC motor could be low. Simulation result did not consider the other switching frequency of DC speed control. However, there was also good power efficiency in the flexible inverter system by evaluating the appropriate response curves.

## ACKNOWLEDGMENT

I would wish to acknowledge my colleagues at Mandalay Technological University who have contributed to the development of this paper. In particular, I would like to thank Myat Su Nwe, my wife, Thet Htar Swe, my daughter, and Zay Yar Tun, my son, for their complete support.

## REFERENCES

- [1] M. Y. Chow and Y. Tipsuwan, "Gain adaptation of networked DC motor controllers based on QOS variations," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 5, pp. 936–943, 2003.
- [2] A. Delibasi, T. Turker, and G. Canserver, "Real time DC position control by fuzzy logic and PID controllers using labview," *International IEEE Conference on Mechatronics & Robotics '04*, Aachen, Germany, September, pp. 9–15, 2004.
- [3] P. Dobra, "Robust PI control for servo DC motor," *International Conference on Control Applications*, vol. 1, pp. 100–110, 2002.
- [4] F. Gurbuz and E. Apkinar, "Stability analysis of a closed loop control for a pulse width modulated DC motor drive," *Turk. J. Elec. Engin.*, vol. 10, no. 3, pp. 427–438, 2002.
- [5] I. B. Kucukdemiral, G. Cansever, and K. Gulez, "Design of a dynamic and robust speed controller for a DC servo motor by using a DSP," *Proc. 2nd International Conference on Mathematical and Computational Applications in Engineering*, Baku, Azerbaijan, Sep., 1–3, 1999.
- [6] D. Y. Ohm, and R. J. Oleksuk, "Influence of PWM schemes and commutation methods for DC and brushless motors and drives," *P.E. Technology 2002 Conference*, Stephens Convention Center (Rosemont, IL), Oct. 27–31, 2002.
- [7] A. Sevinc, "A full adaptive observer for DC servo motors," *Turk. J. Elec. Engin.*, vol. 11, no. 2, pp. 427–438, 2003.
- [8] Dimitri Mahayana, SabatAnwari and SabatAnwari, "Brushed DC Motor Controller Using Robust Indirect Adaptive Control", 2009 International Conference on Electrical Engineering and Informatics, pp. 253–257, 2009.
- [9] Eaaichoon, "DC Motor Speed Control using Microcontroller PIC 16F877a", Malaysia, 2005.
- [10] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Massachusetts: Addison-Wesley Publishing Company, 1995.
- [11] Thierry Castagnet and Jean Nicolai, "Digital Control for Brush DC Motor", *IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS*, VOL. 30, NO. 4, pp. 883–888, 1994.
- [12] Muñoz-César, J.J. et al, "Speed Control of a DC Brush Motor with Conventional PID and Fuzzy PI controllers", *Electronics, Robotics and Automotive Mechanics Conference*, pp. 344–349, 2008.