CHAPTER
3 TWO PHASE OPERATION OF BLDC MOTOR USING HYSTERESIS CURRENT CONTROL

In this chapter, a simple method of controlling the BLDC motor in closed loop using a hysteresis current control technique (HCCT) with Two-Phase Operation (2-ΦO) is developed and implemented. The modeling and simulation is performed using MATLAB®/SIMULINK. To validate the simulation results the hardware implementation is executed using a simple technique to control the speed of the BLDC motor using a 32-bit arm core ARM controller by interfacing it with MATLAB®/SIMULINK WAIJUNG blocks. This method does not require complex coding. WAIJUNG block set is user-friendly similar to SIMULINK blocks.

3.1 Introduction

The disadvantage with permanent magnet motor is increased complexity in the drive operation due to the use of power electronics converters to drive them. Various control strategies such as PWM techniques, current control techniques, voltage control techniques, etc. for the control of BLDC motor drive have been incorporated [12-14]. The stator winding inductance prevents the phase current commutation instantly with the change in rotor position resulting in a spike in non-conducting phase current with only two conducting phases in the BLDC motor. This is the major cause of producing torque ripple in BLDC motor drive along with the inverter DC link voltage. The performance and dynamic response of the motor can be improved by incorporating current control, voltage control, flux control, or torque control methods. An advanced simulation model using hysteresis current control to predict and monitor the dynamic performance of the BLDC motor drive is proposed by [56]. The author [57] has implemented the closed-loop control of the BLDC motor using a PID controller and tested the performance of the drive before and after the addition of the load. A low-cost control for BLDC motor drive with one current sensor and a PI controller to regulate the speed using Arduino Mega controller is suggested by [58].

Today most of the techniques applied for controlling the speed of BLDC motors use high-speed computers, fast ARM controllers, dedicated controllers, and digital signal processors, which require immense coding ability.

To control the motor, complicated programming and skilled programmers are required. In some industrial applications constant speed drives are required and when the load on the motor is variable closed-loop operation is compulsory. The mathematical modeling of the BLDC motor is accomplished and the modeled motor is used for performing speed control of the BLDC motor using HCCT and a PI controller so that a comparison can be made with hardware results.

This chapter discusses the detailed modeling of closed-loop controlled BLDC motor drive incorporated with a PI controller and hysteresis current controller using in section 3.2. The simulation results are discussed in section 3.3. The detailed hardware setup is discussed in section 3.4. The hardware results are discussed in sections 3.5. This chapter also provides a platform to integrate the hardware implementation with the MATLAB®/SIMULINK WAIJUNG blocks to make the drive operation simple without using any complicated programming.

3.2 System description with hysteresis current control technique

The basic block diagram of BLDC motor drive using hysteresis current control technique for closed-loop control is as shown in Fig.3.1. Most BLDC motor have an arrangement of electronics commutator to supply the three-phase stator winding. At any instant of time, only two phases are conducting. To ensure the correct rotation of the motor, the selection of the switching devices should be taken care of. In this technique, the commutation logic for exciting the three-phase stator winding is generated using the hysteresis current controller to obtained improved motor performance than the conventional six-step control. The speed of the motor depends on the magnitude of the applied voltage. Hence speed control can be achieved using various pulse width modulation techniques. The closed-loop incorporates an inner current loop and an outer speed loop. The motor speed can be controlled by employing pulse width modulation techniques. For closed-loop speed control operation, every time the motor speed is determined and correlated with the set value to find the lapse in the speed. The error speed is processed in the PI controller, which in turn adjusts the duty cycle to reduce error. The commutation logic circuit will decide which of the two switches of the inverter bridge are to be switched on for proper commutation from the Hall sensor signal.

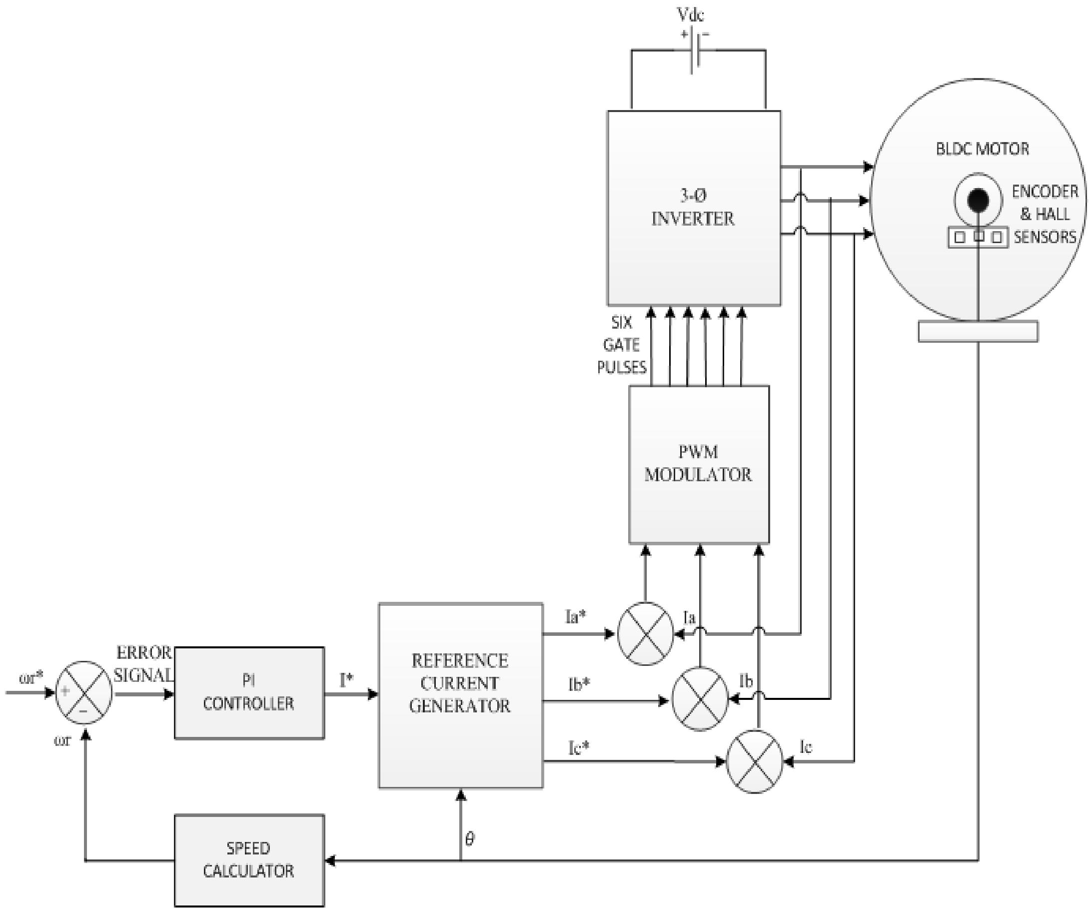


Fig. 3.1 Block diagram of hysteresis current control technique

For current control, a hysteresis controller is used. The controller input is the error signal produced by comparing the actual current with the reference current. The output of the controller is the PWM signals provided to the six inverter switches.

3.2.1 Mathematical modeling using hysteresis current controller

The overall SIMULINK model of a hysteresis current controlled BLDC motor drive consists of a modeled BLDC motor with back emf generation block and three-phase current generation block. The basic equations for motor modeling are taken from chapter 2. Further it includes, a reference current generation block, a PI controller and a hysteresis current controller block to generate switching logic for the three-phase VSI. The trapezoidal back emf being a function of rotor position is formulated using eq. (2.3) is modeled using eq. (3.1) as derived from Fig. 3.2.

The line voltages and back emf are used to generate the three-phase currents which are quasi square waves in nature as shown in Fig.3.2 with only two phases conducting at any time from eq. (2.2).

$$e_a = \begin{cases} \left(\frac{6\mathrm{E}}{\pi}\right)\theta_r & \left(0<\theta_r<\frac{\pi}{6}\right) \\ \mathrm{E} & \left(\frac{\pi}{6}<\theta_r<\frac{5\pi}{6}\right) \\ -\left(\frac{6\mathrm{E}}{\pi}\right)\theta_r+6\mathrm{E} & \left(\frac{5\pi}{6}<\theta_r<\frac{7\pi}{6}\right) \\ -\mathrm{E} & \left(\frac{7\pi}{6}<\theta_r<\frac{11\pi}{6}\right) \\ \left(\frac{6\mathrm{E}}{\pi}\right)\theta_r-12\mathrm{E} & \left(\frac{11\pi}{6}<\theta_r<2\pi\right) \end{cases}$$

$$e_b = \begin{cases} -\mathrm{E} & \left(0<\theta_r<\frac{\pi}{2}\right) \\ \left(\frac{6\mathrm{E}}{\pi}\right)\theta_r-4\mathrm{E} & \left(\frac{\pi}{2}<\theta_r<\frac{5\pi}{6}\right) \\ E & \left(\frac{5\pi}{6}<\theta_r<\frac{9\pi}{6}\right) \\ -\left(\frac{6\mathrm{E}}{\pi}\right)\theta_r+10\mathrm{E} & \left(\frac{9\pi}{6}<\theta_r<\frac{11\pi}{6}\right) \\ -E & \left(\frac{11\pi}{6}<\theta_r<2\pi\right) \end{cases} \qquad 3.1$$

$$e_c = \begin{cases} \mathrm{E} & \left(0<\theta_r<\frac{\pi}{6}\right) \\ -\left(\frac{6\mathrm{E}}{\pi}\right)\theta_r+2\mathrm{E} & \left(\frac{\pi}{6}<\theta_r<\frac{\pi}{2}\right) \\ -E & \left(\frac{\pi}{2}<\theta_r<\frac{7\pi}{6}\right) \\ \left(\frac{6\mathrm{E}}{\pi}\right)\theta_r-8\mathrm{E} & \left(\frac{7\pi}{6}<\theta_r<\frac{9\pi}{6}\right) \\ E & \left(\frac{9\pi}{6}<\theta_r<2\pi\right) \end{cases}$$

The electromagnetic torque is a function of back emf, current, and speed as represented in eq. (2.5). The electrical rotor position is obtained by integrating the actual motor speed as given by eq. (2.7). This theta is used to model the back emf. The actual motor speed is calculated by integrating the rotor position. The back emf is modeled using the eq. (3.1) The ideal back-emf and three-phase stator currents are as shown in Fig. 3.2. It can be seen that the back emf magnitude is constant for 120° and each stator current matches with the back emf flat portion. Six regions each of span 60º with six commutation instances can be observed in one electrical cycle. The rise rate and fall rate of current in the commutation region is equal for ideal case but in actual the rise rate and fall rate being mismatched results in a dip in the non commutating current and hence torque ripple.

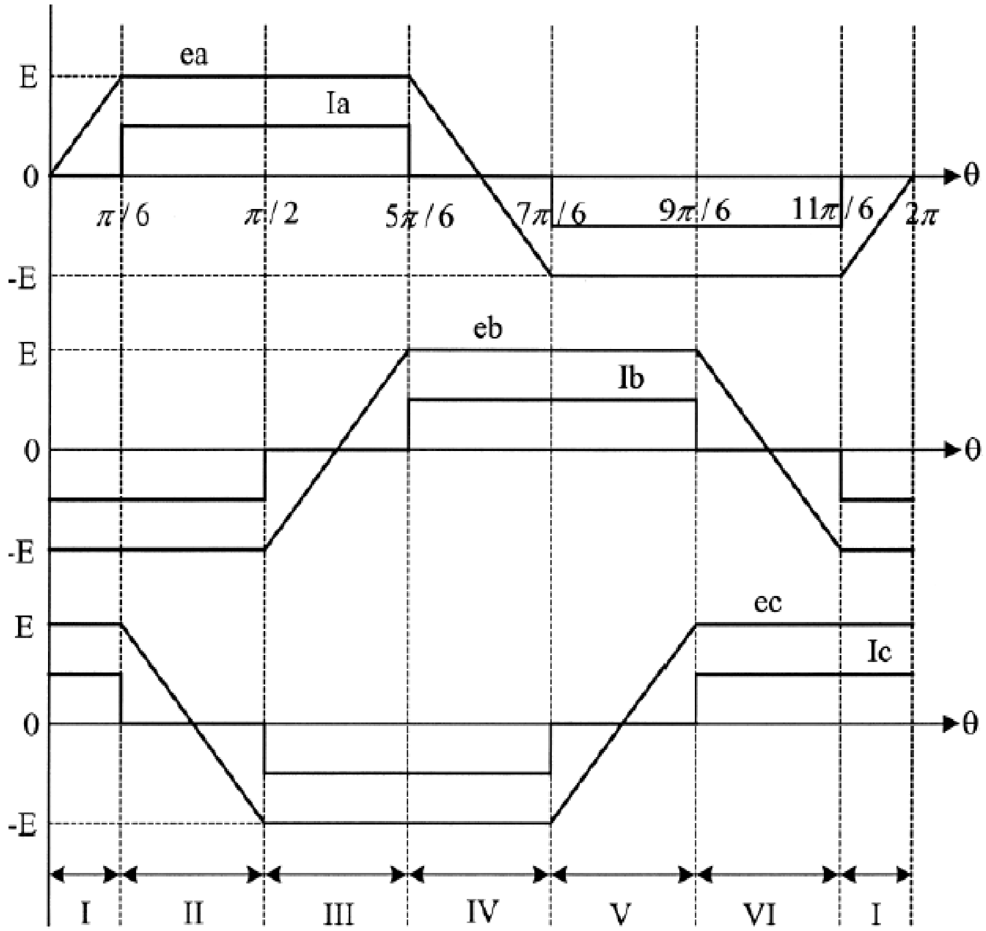


Fig. 3.2 Ideal back emf and current waveforms

3.2.2 PI controller

PI controller block is used to process the error produced by comparison of the set speed with the actual speed as shown in Fig.3.3. The output of the PI controller is torque reference which when divided by torque constant Kt gives the maximum current I* which is used to generate three reference currents.

A simple way to tune the drive is discussed using the mechanical and electrical time constants as given in eq. (3.2) and eq. (3.3). These time constants depend on the motor parameters. Using this time constant a transfer function G(s) is obtained as given in eq. (3.4).

$$\tau_m = \frac{J3R}{K_e K_t} \qquad 3.2$$

$$\tau_e = \frac{L}{3R} \qquad 3.3$$

$$G_s = \frac{\frac{1}{Ke}}{\tau_m \tau_e S^2 + \tau_m S + 1} \qquad 3.4$$

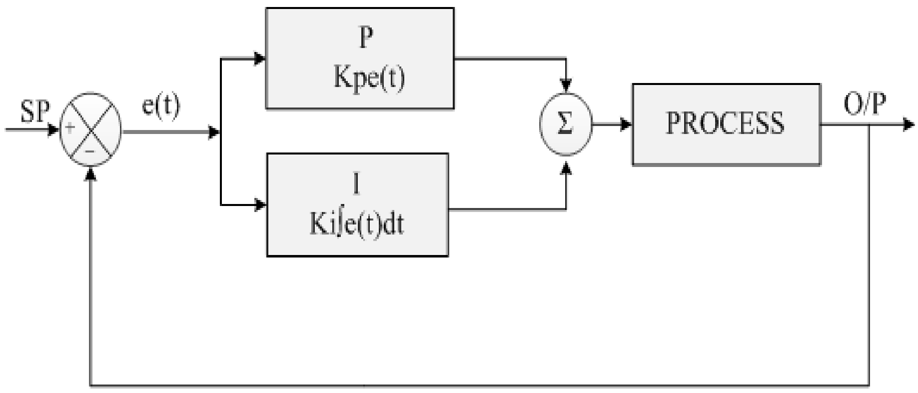


Fig. 3.3 Tuning of PI controller

This BLDC motor transfer function is utilized to tune the PI controller. The tuned block is then placed as a PI controller which processes the error signal from which maximum torque is obtained. The PI controller can be represented as

$$Y(t) = K_p e(t) + K_i \int e(t) d(t) \qquad 3.5$$

Where Y(t) is the process output, SP is the set point or reference value, e(t) is the error produced due to the mismatch between the set value and the output, Kp and Ki are the proportional and integral constants respectively as given by eq. (3.5). The proportional gain introduces steady state error and decreases the rise time which leads to high system gains but introduces steady-state error. The system may become unstable with large proportional gain. The steady state error can be removed with suitable value of integral gain but may result in decrease in the system response. It may lead to oscillations and overshoot. Proper selection of this constants leads to fine tuning of the PI controller results in smooth operation of the drive.

3.2.3 Reference current generation

In the motor operation with the hysteresis current control technique, three reference currents are generated. To generate the maximum current I*, the maximum torque is divided by torque constant Kt. Using the knowledge of the maximum current I*, three reference currents are generated based on the rotor position as given in Table 3.1

Table 3.1 Generation of three reference currents

Rotor Position θ(degree)	**Three Reference Currents**		
	I_{aref}	I_{bref}	I_{cref}
0-60	-I*	+ I*	-
60-120	-I*	-	+ I*
120-180	-	-I*	+ I*
180-240	+ I*	-I*	-
240-300	+ I*	-	-I*
300-360	-	+ I*	-I*

The reference currents are compared with the actual currents using a hysteresis current controller.

3.2.4 Hysteresis current controller

A hysteresis controller is modeled with a narrow hysteresis band of ± 0.1. The actual motor current is compared with the reference current within the narrow hysteresis band. The error signal is used to decide the switching logic for the three phase VSI. The commutation logic should be such that the upper and lower switches of the same phase leg should not be switched ON at the same instant as it would result in the short circuit. The logic used for modeling the hysteresis current controller for the phase current Ia is as given below:

1. **Case I: Ia > 0**

- If Ia < LL → Switch S1 is turned ON
- If Ia > UL → Switch S1 is turned OFF and D2 is conducted
- If LL < Ia < UL and dIa/dt > 0→ Switch S1 is turned ON
- If LL < Ia< UL and dIa/dt < 0→ Switch S1 is turned OFF and D2 is conducted

2. **Case II: Ia < 0**

- If Ia > UL → Switch S2 is turned ON
- If Ia < LL → Switch S2 is turned OFF and D1 is conducted
- If LL < Ia < UL and dIa/dt < 0→ Switch S2 is turned ON
- If LL < Ia < UL and dIa/dt > 0→ Switch S2 is turned OFF and D1 is conducted

Similarly, for phase current Ib

1. **Case I: Ib > 0**

- If Ib < LL → Switch S3 is turned ON
- If Ib > UL → Switch S3 is turned OFF and D4 is conducted
- If LL < Ib < UL and dIb/dt > 0→ Switch S3 is turned ON
- If LL < Ib< UL and dIb/dt < 0→ Switch S3 is turned OFF and D4 is conducted

2. **Case II: Ib < 0**

- If Ib > UL → Switch S4 is turned ON
- If Ib < LL → Switch S4 is turned OFF and D3 is conducted
- If LL < Ib < UL and dIb/dt < 0→ Switch S4 is turned ON
- If LL < Ib < UL and dIb/dt > 0→ Switch S4 is turned OFF and D3 is conducted

For phase current Ic

3. **Case I: Ic > 0**

- If Ic < LL → Switch S5 is turned ON
- If Ic > UL → Switch S5 is turned OFF and D6 is conducted
- If LL < Ic < UL and dIc/dt > 0→ Switch S5 is turned ON
- If LL < Ic< UL and dIc/dt < 0→ Switch S5 is turned OFF and D6 is conducted

3. **Case II: Ic < 0**

- If Ic > UL → Switch S6 is turned ON
- If Ic < LL → Switch S6 is turned OFF and D5 is conducted
- If LL < Ic < UL and dIc/dt < 0→ Switch S6 is turned ON
- If LL < Ic < UL and dIc/dt > 0→ Switch S6 is turned OFF and D5 is conducted

Based on the above logic, detailed investigation of the phase current Ia is shown in Fig. 3.4. If the error which is the difference between the reference current I* and the actual current Ia is greater than the hysteresis band upper switch is ON and if the error is less than the hysteresis band lower switch conducts to produce pulse width modulated gate pulses for the three-phase inverter. The gate pulses for the top and bottom switches of the same phase leg of the inverter bridge as obtained from the comparison between the actual and reference is shown in Fig 3.4.

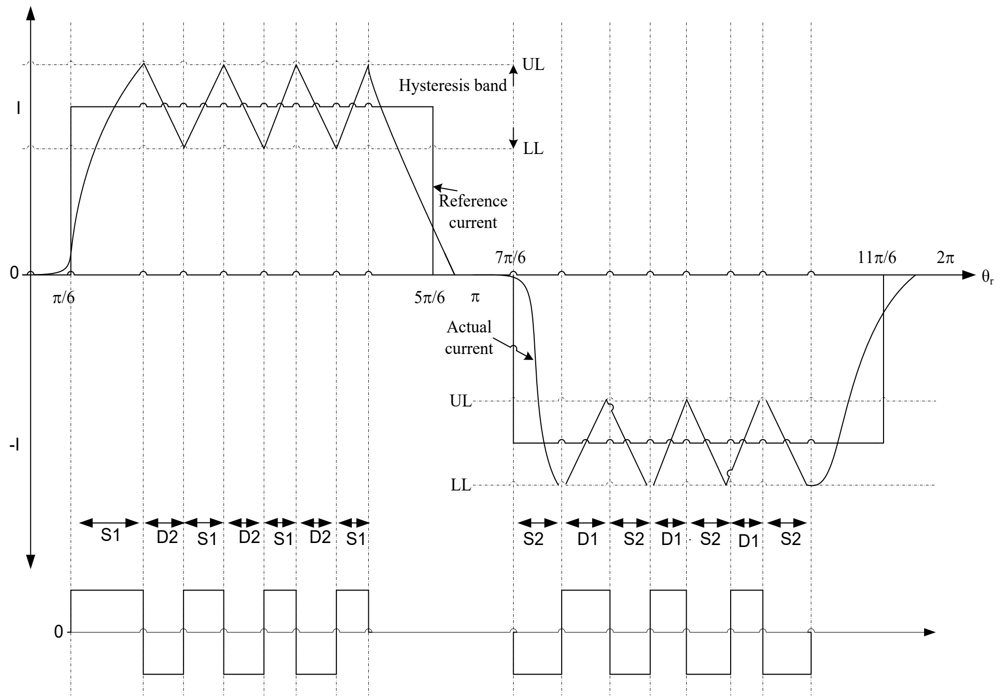


Fig. 3.4 Investigation of hysteresis current control for phase A

3.3 Simulation results and discussion

Simulation of closed-loop control of BLDC motor using hysteresis current control technique is performed to check the effectiveness of the technique on motor performance. The result of the BLDC motor drive with hysteresis current control technique is as shown in Fig.3.5 to Fig.3.13. Initially, the motor speed is set at 1000 rpm. At time t=0.8 sec the set speed is changed to 2000 rpm with a constant load of 0.1 Nm. The change in back emf, stator current, and torque is observed in the results. Again at time t= 1.5 sec, the motor speed is reduced to 1500 rpm. Proper tuning of the PI controller leads the motor to run at the set speed with some speed ripple during high-speed operation. It is observed that the time taken by the controller to reach the set speed is very less and is not noticeable for higher speed. Figure 3.5. and 3.6 shows the variation in the three-phase back emf with applied speed change. The variation in the three-phase stator current is as shown in Fig. 3.7 and Fig. 3.8. The variation in speed according to the set speed can be observed in Fig. 3.9. The speed is changed from 1000 rpm to 2000 rpm and again reduced to 1000 rpm. The behavior of electromagnetic torque with the applied speed is shown in Fig. 3.10. The effect of speed can be easily noticed on the electromagnetic as shown in Fig. 3.11 to Fig 3.13. The torque ripple increases with the increment in speed owing to the mismatch in the incoming and outgoing current rate as discussed in chapter 2 (2.5). The torque ripple at a speed of 2000 rpm is higher than that at 1000 rpm and 1500 rpm which affects motor performance for high-speed applications.

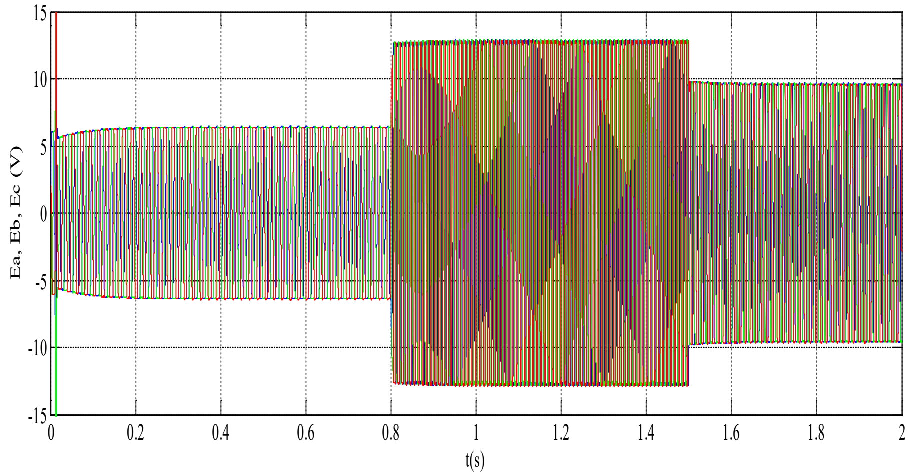


Fig. 3.5. Variation in back emf with a speed change at 0.8 sec and 1.5 sec with a constant load of 0.1 Nm

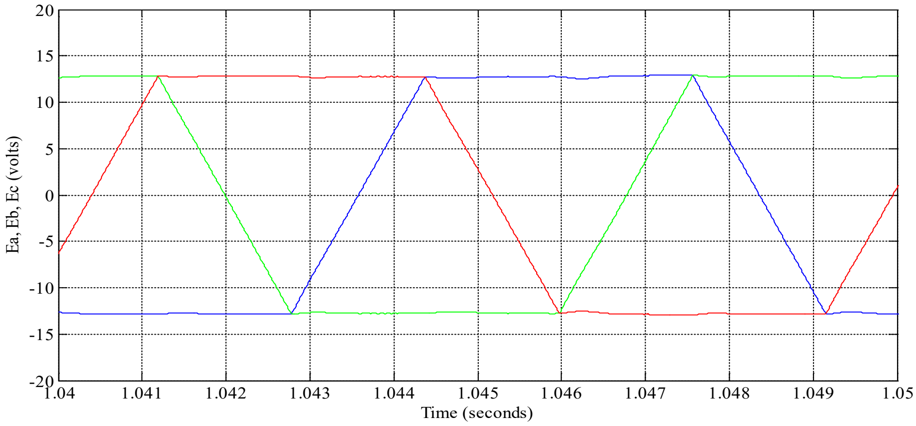


Fig. 3.6. Zoomed view of three-phase back emf at a speed of 2000 rpm

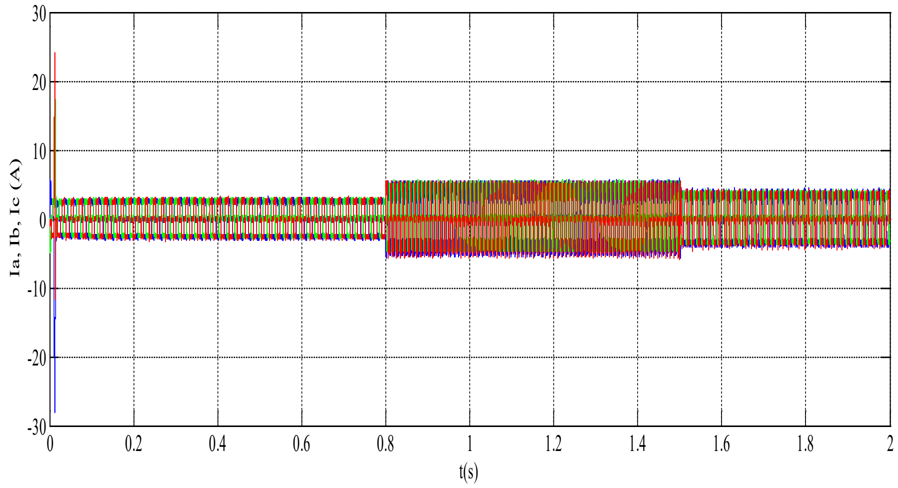


Fig. 3.7 Variation in stator current with a speed change at 0.8 sec and 1.5 sec with a constant load of 0.1 Nm

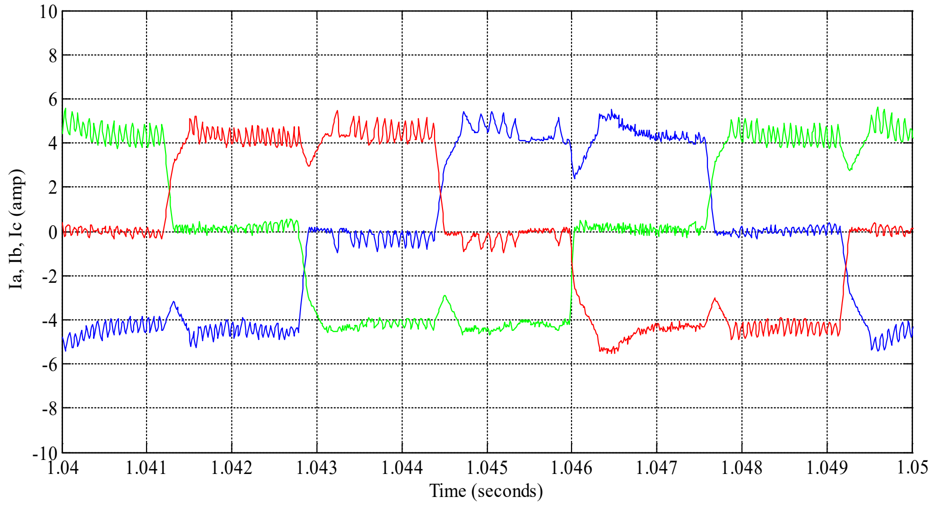


Fig. 3.8. Zoomed view of three-phase stator current at a speed of 2000 rpm

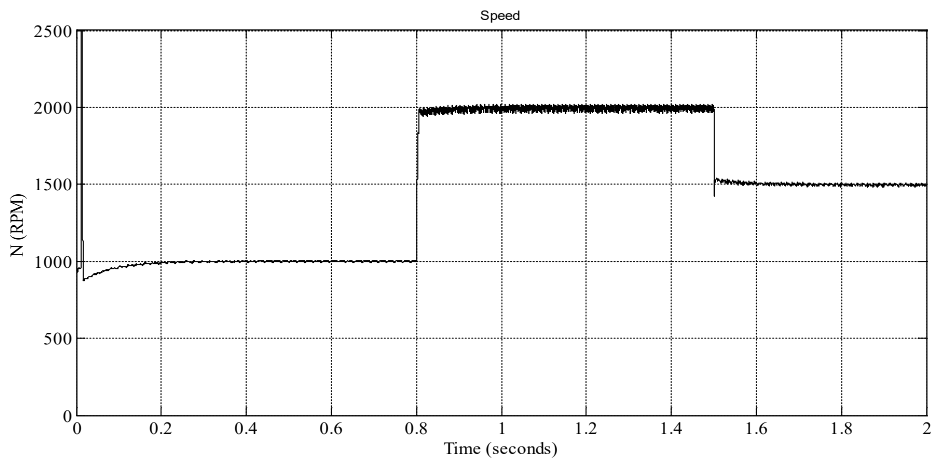


Fig. 3.9. Speed change at 0.8 sec and 1.5 sec with a constant load of 0.1 Nm

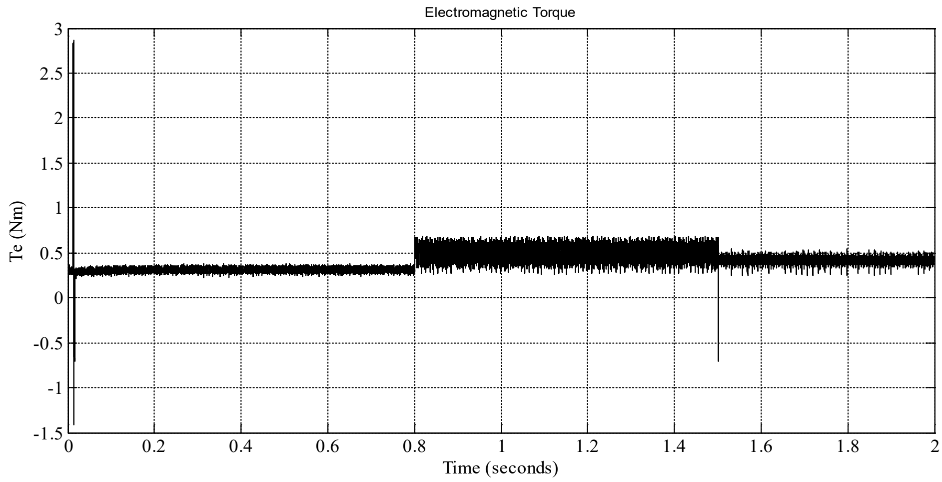
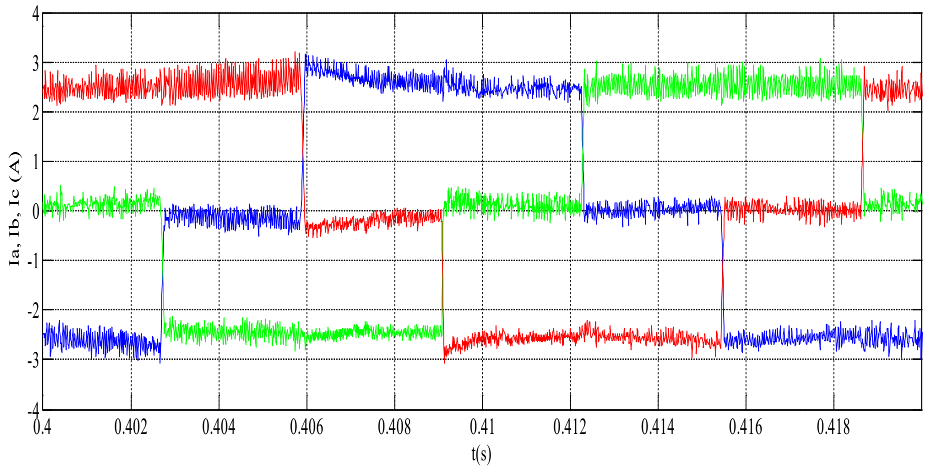
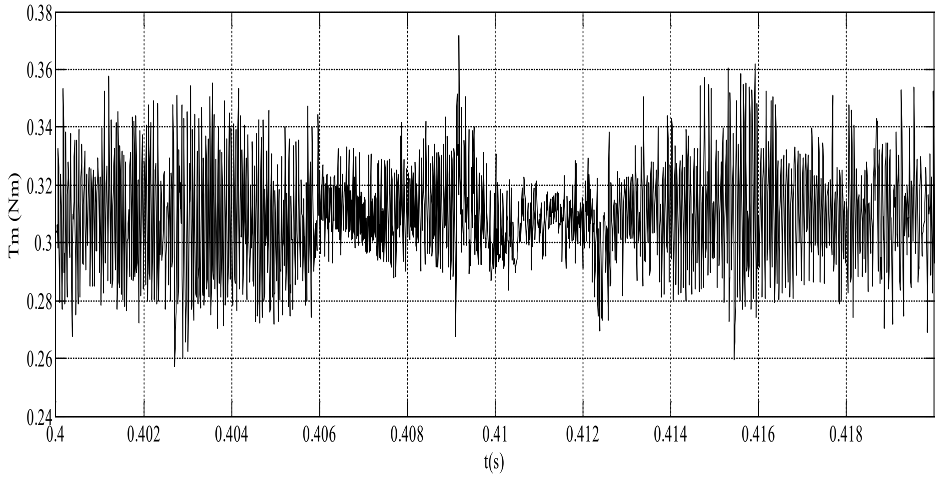


Fig. 3.10 Variation in motor torque with a speed change at 0.8 sec and 1.5 sec with a constant load of 0.1 Nm

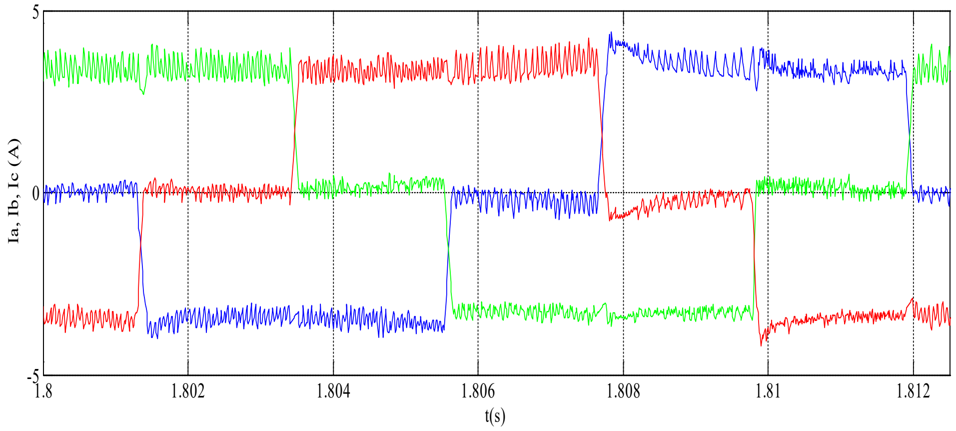


(a)

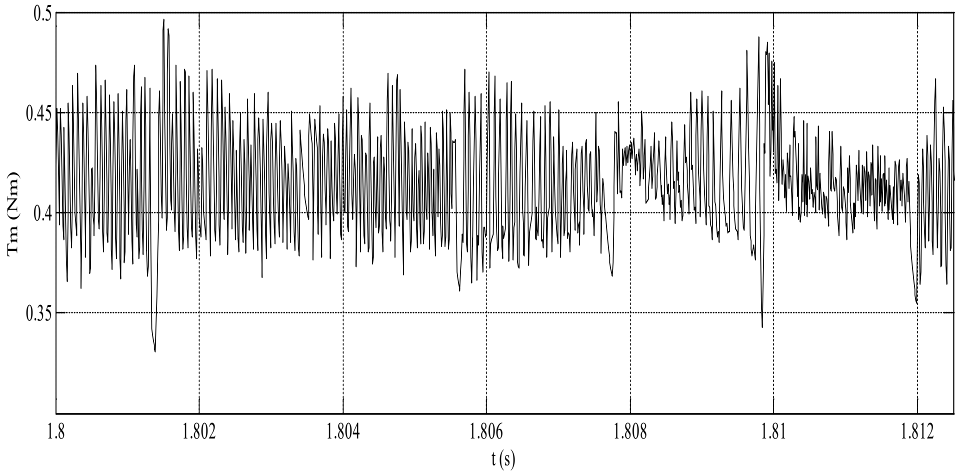


(b)

Fig. 3.11 Zoomed view of (a) stator current (b) electromagnetic torque at 1000rpm

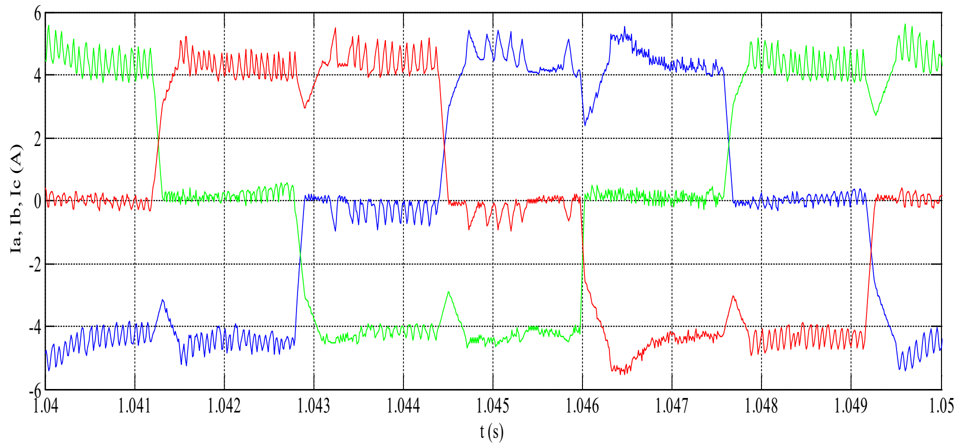


(a)

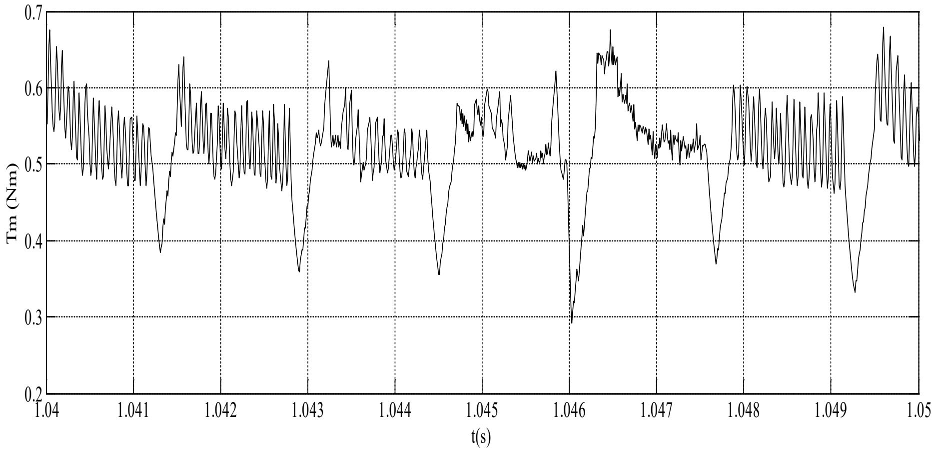


(b)

Fig. 3.12 Zoomed view of (a) stator current (b) electromagnetic torque at a speed of 1500rpm



(a)



(b)

Fig. 3.13 Zoomed view of (a) stator current (b) electromagnetic torque at a speed of 2000 rpm

3.4 Hardware Description & Design

The overall schematic block diagram of the system used to carry out the closed-loop operation of the BLDC motor is as shown in Fig.3.14. The hardware setup of closed-loop control of the BLDC motor is implemented as shown in Fig. 3.15. The Hardware setup consists of a current sensor card, driver card, 3-phase voltage source inverter card acting as electronic commutator, BLDC motor with hall sensor/shaft encoder, STM32 discovery card. The same hardware setup is used for operating the motor with 2-ΦO, 2-3ΦO, and 3-ΦO. The interfacing of the hardware circuit with MATLAB®/SIMULINK WAIJUNG block set has to be done to generate gate pulses for the IGBT switches of the three-phase bridge inverter circuit using different control strategies.

The closed-loop speed control of the BLDC motor requires the following hardware as well as software for experimental implementation:

- BLDC motor set with Encoder/Hall sensors.
- IGBT Driver and Power Card
- STM32 Discovery Card
- ARM Cortex M4 - WJ - 32 Bit Kit
- Matlab 2014a
- MDK516a (Keil ARM IDE)
- STM32 ST-LINK Utility_v2.3.0
- WAIJUNG14_12a or higher

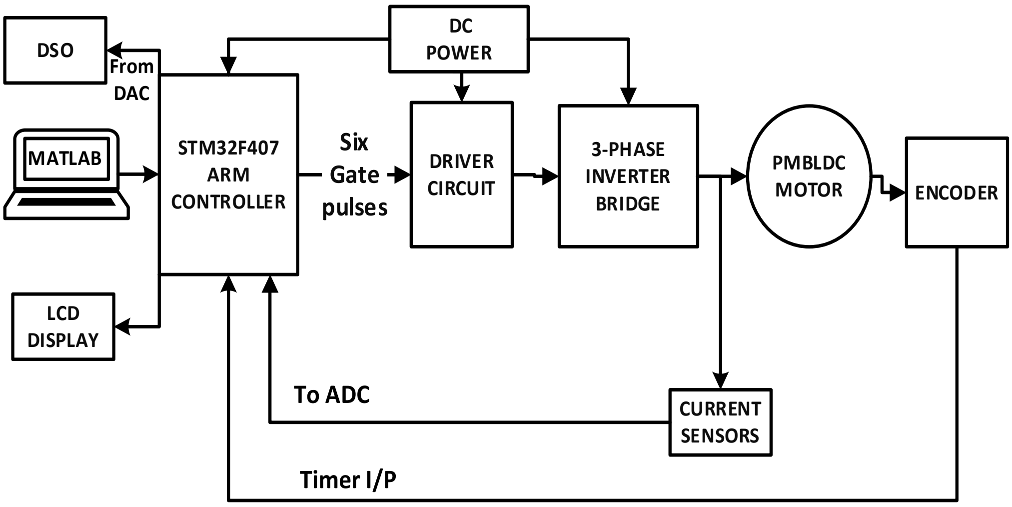


Fig. 3.14 Schematic block diagram of overall system

BLDC motor with the parameters as given in Table 3 is used for experimentation. The experimental setup for closed-loop control is as shown in Fig. 3.15. This experimental setup is used to test the BLDC motor operation with different control techniques operating the motor with 2-ΦO, 2-3ΦO, and 3-ΦO. A detailed discussion of the hardware setup is discussed here. The same is referred to in chapters 4 and 5.

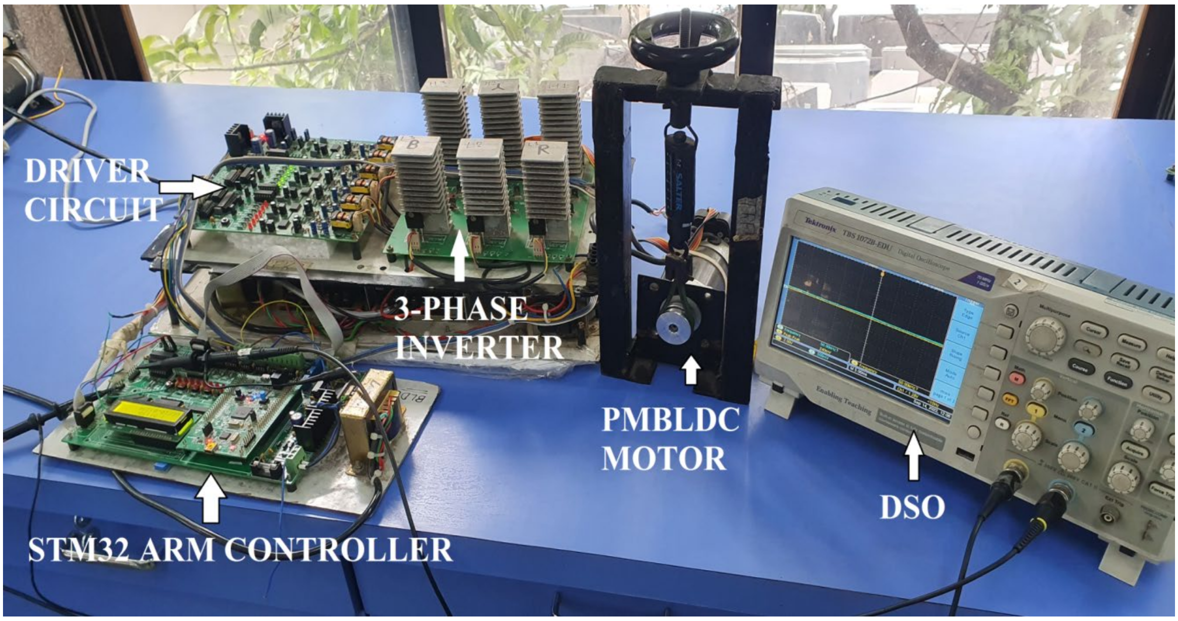


Fig. 3.15 Hardware System with STM32 Discovery Card

3.4.1 BLDC motor with a shaft encoder

A permanent magnet brushless dc motor of 36 volts, with concentrated winding producing a trapezoidal back emf and quasi square wave currents with only two phases excited at a time, is used for hardware implementation. The BLDC motor has three hall sensors for providing the accurate rotor position as well as a shaft encoder with a resolution of 1250 PPR mounted on the shaft. The speed of the motor is calculated from the output of the encoder using the controller, Unlike the DC motor which has a mechanical commutator BLDC motor requires an electronic commutator. The switching of the electronic commutator depends on the rotor position. The rotor position required for producing the commutation logic for the inverter circuit is also obtained using an encoder. The motor is provided with a loading arrangement which helps to study motor behavior with changes in load.

Table 3.2 BLDC Motor Parameters

Parameters	**Values**
Voltage (V)	36 volt
Poles (P)	4
Rated Speed (N)	4000 rpm
Rated Torque (T)	0.32 Nm
Resistance per phase (R)	0.5 Ω
Inductance per phase (L-L)	1.65 mH
Moment of inertia (J)	$17.3*10^{-6}$ Kg-m^2
Torque Constant (Kt)	0.061 Nm/A

3.4.2 Operation of 3-Φ inverter as electronic commutator

For operating BLDC motor the key factor is the acknowledgment of rotor position and energizing two-phase windings. In sensored BLDC motor magnetic hall sensors are used for acknowledgment of correct rotor position. Here a shaft encoder is used for detecting the rotor position. In some motors, hall sensors are available to provide rotor position. Generally, 120̊ hall sensing technique is used. Three different hall sensors are located at three different places in the stator winding. When they are moved in a bipolar magnetic field they generate TTL compatible output as discussed by [57] can be interfaced with logic circuits or controllers. For every hall sequence, there are two possible combinations of inverter switches that with one combination the motor will rotate in a clockwise (forward) direction and for the other, it will rotate in a counter-clockwise(reverse) direction. The hall sequence and the respective switching combinations are discussed in brief in this paper. For changing the direction of rotation of motor the only thing required is the selection of proper switches at a particular hall sequence as suggested by[60].

The output of the incremental shaft encoder is given to the QEI pins of the controller to collect the information of rotor position. There are six commutation states in one electrical cycle depending on the rotor position. To obtain correct switching combinations for the electronic commutator a simple method is discussed for the motor with hall sensors. For example, if the present hall sensor state is 101 and the possible switch combinations are 1&4, 1&6, 3&6, 3&2, 5&2, 5&4. Out of these six combinations only two combinations are correct. For every commutation state or hall sequence, there are two possible combinations of inverter switches that with one combination the motor will rotate in the forward direction and for the other, it will rotate in the reverse direction. The switching combination for the forward direction is shown in Table 3.3. The switching pattern matches that of the 3-Φ, 120° conduction mode VSI.

Table 3.3 Switching combination for forward operation

Commutation States			**Switch Combinations**
H_A	H_A	H_C	Forward direction
1	0	1	S1, S4
1	0	0	S1, S6
1	1	0	S3, S6
0	1	0	S3, S2
0	1	1	S5, S2
0	0	1	S5, S4

3.4.3 Driver circuit

The driver circuit is used for driving the IGBTs of the Inverter card as well as to provide isolation between the control circuit and power circuit. Here an Infineon-made driver card is used for driving the six IGBT switches. Each switch is provided separate isolation from a coreless transformer to isolate their supply and ground terminals.

3.4.4 Current sensor

Current sensors are used to measure the three-phase stator current. For this current transformers are used. The measured current is fed to the ADC of the STM32 discovery card. A unipolar signal is to be provided to the ADC whose operating range is from 0-3.3 volts.

3.4.5 STM32 Discovery card & MATLAB WAIJUNG

STM32F407VGT6 microcontroller is a 32-bit ARM controller with FPU core[61]. The controller is featured with 1MB Flash memory for coding large data, 192-Kbyte RAM which helps in storing large data with LQFP100 package is used in hardware, which facilitates the closed-loop control of the drive. STM32 bit Arm controller has a generic core that facilitates C coding. The software requirement is MATLAB 2014a which has a WAIJUNG block set. Installation of MDK516a (Keil ARM IDE) and STM32 ST-LINK Utility_v2.3.0 along with MATLAB 2014a to interface STM32Discovery card with WAIJUNG block set is required.

In this hardware implementation of BLDC drive, the hysteresis current controller and PI controller are modeled in MATLAB®/SIMULINK WAIJUNG block set which generates C code automatically for STM32 ARM controller. For this, the MATLAB model is interfaced with the microcontroller to dump the code generated by the WAIJUNG block set.

3.4.6 Generation of gate pulses for 3-phase inverter bridge

Generally, closed-loop operation of BLDC motor using 32 bit ARM Controller requires knowledge of assembly language or C coding. Here, a simple technique is used in which no coding is required. The 32 bit ARM controller is interfaced with the MATLAB®/SIMULINK model. Programming of the microcontroller is done using the WAIJUNG block set in MATLAB®/SIMULINK library [9].

At first, the target set up from the device configuration block is done. The selection of Analog to Digital Converter (ADC), Digital to Analog Converter (DAC), timers, etc. is done from the on-chip peripherals readily available. From hardware modules, the Character LCD is selected to display set speed and actual speed. ADC is used to convert an analog value into a digital value. As ADC is 12 bit, hence 4095 count is available. This count can be calibrated in terms of speed, current, voltage, frequency, modulation index, etc. The maximum voltage to ADC is 3.3 volt. For an analog signal of 0 to 3.3 volt, ADC gives a 0 to 4095 count. Here three-phase currents are measured by the CT. The value of current is calibrated as 5A (RMS) to 3 V_{pp}. DAC converts the digital signal to analog. Here it is used to convert angle theta into analog value. The actual speed of the motor is calculated from theta. The actual speed is compared with the set speed. A PI controller is used to process the error. The output of the PI controller is torque reference from which maximum current is obtained. This signal is used to generate three reference currents based on rotor position as mentioned in Table 3.1. These reference currents are compared with the measured currents within a narrow band of 0.05 using a hysteresis controller to generate gate pulses. Advanced Timer-8 is used to generate the six gate pulses for the three-phase inverter. A dead band of a 1-micro second is kept between the upper and lower switch of the same phase leg, to avoid shoot-through. The execution time of the overall system is approximately 100 μs. WAIJUNG also facilitates the display of set speed and actual speed on the LCD display.

3.5 Experimental results and discussion

Experimental results prove the workability of the method in actual practice. It provides the actual behaviour of the 2-ΦO and a platform to compare the simulation results with the actual results. Hardware results of closed-loop control of BLDC motor are captured using DSO. The control scheme is tested on a 36-volt BLDC motor with motor parameters described in Table 3.2. The main intention of performing the 2-ΦO with the hysteresis current control technique is to test the hardware setup and validate the theoretical waveforms of the stator current, back emf, rotor position, and speed with the simulation results. The waveforms of back emf, stator current, rotor position theta as well as the variation of actual speed with the set speed can be seen in the following figures. The motor is allowed to run at a speed of 2000 rpm with a maximum loading of up to 2.5 ampere.

The loading to the motor is given with the belt and pulley arrangement. It can be seen from Fig. 3.16, that the motor back emf is trapezoidal with quasi square wave currents. The variation of back emf and stator current with rotor position can be seen in Fig. 3.17 and Fig. 3.18. Precise tuning of the PI controller is done to allows the actual speed to vary according to the reference speed as shown in Fig. 3.19. It can be seen that as the load varies, the PI controller takes corrective actions to reduce the error and maintains the actual speed of the motor equal to the set speed.

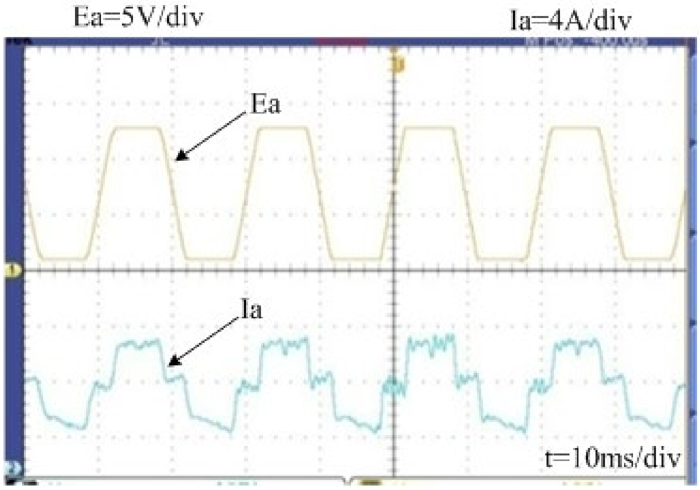


Fig. 3.16 Back emf and stator current at a speed of 2000 rpm

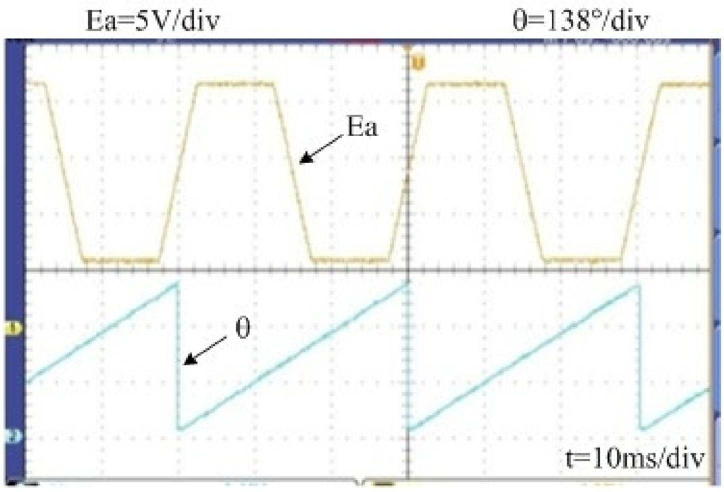


Fig. 3.17 Back emf and theta at a speed of 2000 rpm

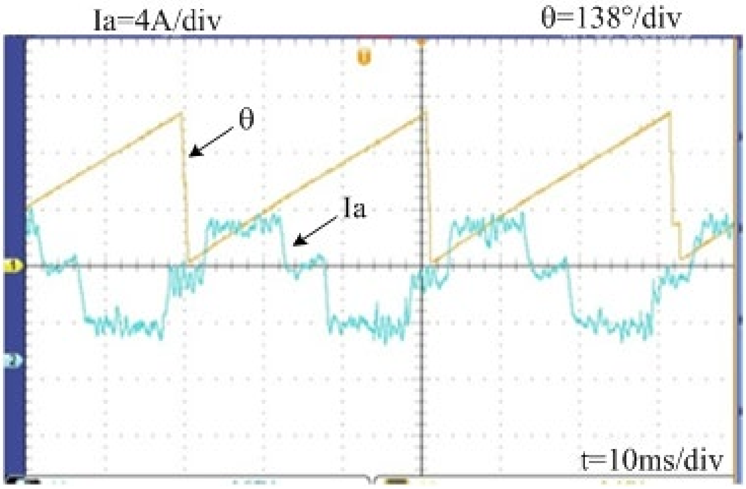


Fig. 3.18 Theta and stator phase current at a speed of 2000 rpm

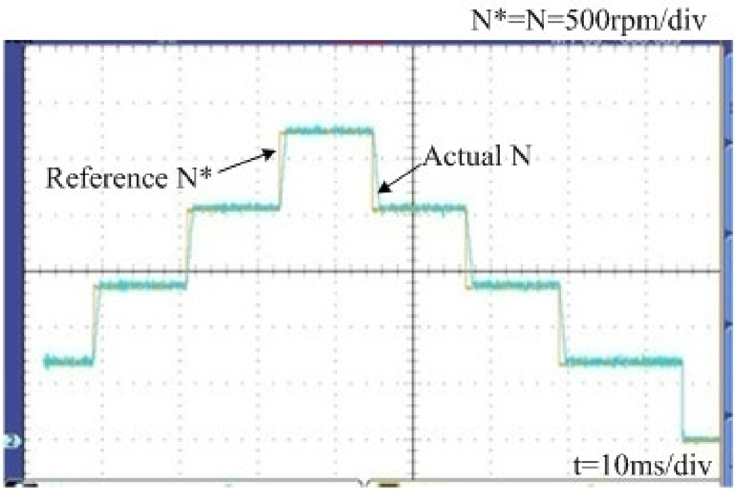


Fig. 3.19 Variation in actual speed with set speed

3.6 Conclusion

The popularity of the BLDC motor in the international drive market is increasing very fast especially in automobile engineering, the textile industry, and home appliances. This work depicts a simple and effective speed controlling technique using HCCT with MATLAB WAIJUNG block set which does not require 32bit coding or C programming. The BLDC motor is first modelled with actual motor parameters and then simulation of the same is carried with a hysteresis current control technique in MATLAB®/SIMULINK. To validate the simulation results, hardware implementation is carried out on a 36-volt motor by interfacing the hardware modules with MATLAB®/SIMULINK WAIJUNG blocks. This is used to produce six gate pulses for the 3-phase inverter bridge which drives the motor using correct switching logic. It can be observed from hardware results that fine-tuning of PI controller makes the motor follow the set speed despite variation in the load.