

# Speed Control of BLDC Motor using DRV8312EVM in VisSim Environment

Vardi Satya Kumari<sup>1</sup>, R. Anil kumar<sup>2</sup>, Syed Sarfaraz Nawaz<sup>3</sup>

<sup>1</sup>M-Tech Student, GRIET/EEE Department, Hyderabad, India  
satyavardi@yahoo.com

<sup>2</sup>Asst. Professor, GRIET/EEE Department, Hyderabad, India  
rajagiri\_anil04@yahoo.co.in

<sup>3</sup>Assoc. Professor, GRIET/EEE Department, Hyderabad, India  
sarfaraz86nawaz@gmail.com

**Abstract:** The paper presents controlling the Speed of Brushless DC Motor by using VisSim software interfaced with a GUI named InstaSPIN-FOC (Field Oriented Control). VisSim is a software related to Visual Environment for model based exploring the Embedded Systems. It has a highly ingenious diagram-to-code capability, which reduces the complexity of design, time and improves quality of the product. In recent times, Brushless DC Motor has found wide range of Industrial applications for its high reliability, high efficiency, more torque per weight and low maintenance. InstaSPIN-FOC is a complete sensor-less FOC, which provides a solution in on-chip ROM for the selected devices (FAST observer, FOC, speed and current loops), masterfully governs the speed of the Motor without use of any speed sensors.

This paper provides a basic understanding of DRV8312 and F28069. Simulation using VisSim have been executed. Prior to VisSim, simulation is performed with MATLAB Simulink. The speed waveforms for various gain settings are observed and hence studied the drive response. In this paper, the Speed Control of Brushless DC Motor using DRV8312 evaluation module along with F28069 control card in VisSim environment is been implemented.

**Index Terms:** VISSIM, DRV8312EVM, InstaSPIN-FOC, BLDC Motor, FOC, FAST (Flux, Angle, Speed, Torque).

## I. INTRODUCTION

Power Electronics devices find numerous options for conducting simulation and analysis. Among those, MATLAB plays a vital role in simulation with an interface of the National Instruments (NI) hardware for conducting experiments. Use of NI fixtures results in high cost. Alternatively other Microcontrollers such as Arduino (Mega) help in reduction of the cost criteria with proper control algorithms, but dynamic simulation is not possible. Concurrently VISSIM overcomes the problems aroused with above discussed Microcontrollers.

VisSim [1] provides a highly efficient environment covering all phases of developing control systems for Electric Drives. Their diagram-to-code capability frees Electric Drive Engineers from manual coding and allows them to concentrate on developing the most optimal control in the shortest time. Enormous list of Motor control blocks set helps in achieving the in-depth motor modeling. And the model based approach gives a clear overview of the designs.

InstaSPIN - support for Motor Identification and efficient Sensor-less Field Oriented Control (FOC) [2,3]

even at minimum of 1Hz rates. Industrial applications for the BLDC drive address the challenges of achieving a compact solution with higher efficiency and robust protection features like cycle-by-cycle current limiting during overload and short circuit conditions to protect power devices.

## II. BLOCK DIAGRAM FOR INTERFACING BLDC MOTOR WITH VISSIM

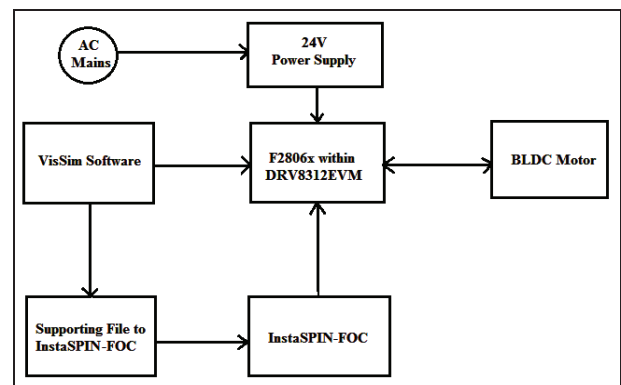


Figure 1. Block diagram of VisSim interfacing with BLDC Motor

Interfacing of BLDC [8] Motor with VisSim using DRV8312 Evaluation Module and InstaSPIN-FOC is shown in Figure 1. The DRV8312 board requires the supply voltage of 24V supplied from AC mains. The code is generated in backend of the VisSim software for designed block diagram, which loads the machine code into the controlCARD (F2806x) placed in the slot provided by the DRV8312 board. The F28069 with DRV8312 and with the supporting files of VisSim initiates InstaSPIN-FOC. InstaSPIN-FOC is sensorless solution which helps in efficient control of BLDC Motor.

## III. MODELING AND RESULTS OF BLDC MOTOR FOR SPEED CONTROL IN MATLAB

The MATLAB [4] simulation of BLDC motor for controlling the speed in a closed loop to obtain the required target speed is shown in Figure 2 where a reference speed is compared with the achieved target speed in a closed loop.

PI controller is used continuously which calculates the error value and applies a correction based on the

proportional and integral terms depending on the error signal at the input. Controlled voltage source converts the input signal to equivalent voltage source. Inverter used is connected to three phase BLDC Motor. As BLDC Motor needs hall sensors to detect rotor position and decoder converts pin positions to EMF values. And finally position EMF's are converted to gate pulses and fed back to inverter.

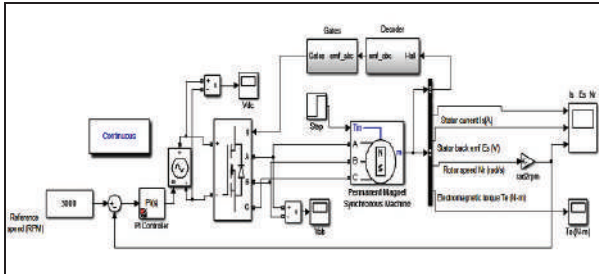


Figure 2. MATLAB simulation scheme of BLDC motor

The Simulation result for Speed Control of BLDC Motor in MATLAB is shown in Figure 3. The graphs represented in the figure indicate Stator current, Stator Back EMF and Speed. From the result shown in graph the set speed is controlled.

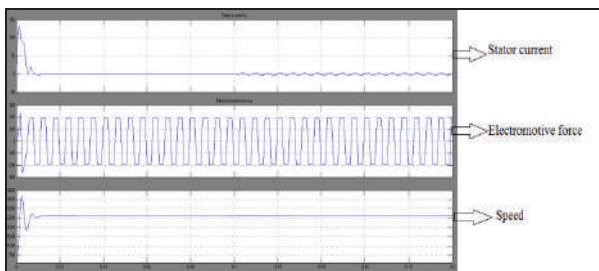


Figure 3. MATLAB simulation results for Stator current, Stator back EMF, Rotor speed

#### IV. VISIM

**VisSim [5]** A Visual Environment software for modeling is based on development of Embedded Systems. Its highly efficient diagram-to-code capability reduces the development time and increases quality of the product. By just swapping a small number of blocks, users can change their models from Software to Processor or Hardware in the Loop. VisSim offers full peripheral support for a large family of Microprocessors, thus making hand coding obsolete and the transition to actual Motor control. Models can be simulated and run in hardware in the Loop simultaneously.

VisSim is built with code efficiency being the top priority, so the simulation-only elements and the hardware specific blocks can be disregarded in code generation and simulation respectively. The VisSim/C-Code add-on generates ANSI C code for the model, and generates target specific code for on-chip devices like PWM, ADC, encoder, GPIO, I2C etc. After the behavior of the controller has been simulated, C-code can be generated, compiled and run on the target. For debugging, VisSim supports an interactive JTAG linkage, called "Hotlink", that allows interactive gain change and plotting of on-target variables. The VisSim

generated code is efficient and readable, making it well suited for development of Embedded Systems.

#### V. DRV8312EVM (Evaluation Module)

The Medium Voltage Digital Motor Control (DMC) module (DRV8312) [6], provides a prominent way to learn and experiment with digital control of medium voltage brushless motors for increasing the efficiency of operation. The DRV8312 board with the control card slot for inserting F28069 control card.

The Motor control board is partitioned into functional groups that enable a complete motor drive system. These functional groups are called Macro Blocks and their labeling as shown in Figure 4.

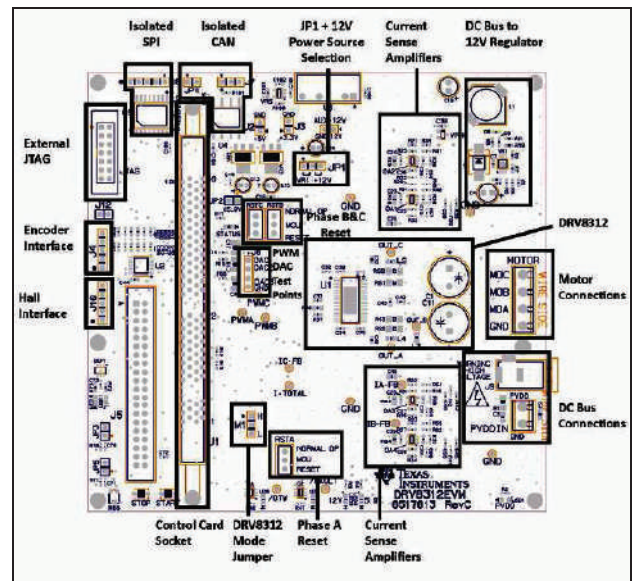


Figure 4. DRV8312 - Evaluation Board Macros

The board is separated into two power domains:

1. The low voltage Controller Power domain powers the controller and the logic circuit present on board
2. The medium voltage supply line that is used to carry the medium voltage and current such as the DC power for the Inverter (also referred to as DC Bus).

##### A. Controller Power

Comprised of the 12 V, 5 V, and 3.3 V that the board uses to power the controller and the logic and sensing circuit present on the board. This power can be sourced from two places (Jumper JP1 selects between the two)

- **+12V DC control power entry:** Connect an external bench supply with 1A current limit here.
- **On board regulator, VR1:** +12V is regulated from DC bus power via an on-board buck regulator

##### B. DC Bus Power

Is the medium voltage line up to 52.5V - that provides the voltage to the inverter stage to generate three phases to control the motor. Connect supplied 24 V.

**VI. PICCOLO F28069 CONTROL CARD**

The F28069 [7] is one of the Piccolo families of microcontroller (MCUs) equipped with the F28069 core and CLA coupled with highly mixed control peripherals in low pin-count devices. An internal voltage regulator, acquiesce for single-rail process. Enhancements have been made to the High-Resolution Pulse Width Modulator (HRPWM) module for enabling dual-edge control (Frequency Modulation). Analog comparators with in-build 10-bit references have been added which can be embedded directly to control the e-PWM outputs. The ADC converts 0 to 3.3V fixed full-scale range and supports ratio-metric VREFHI/VREFLO indications. The ADC interface has been minimized for less complications and inactivity. The device supply voltage is 3.3 V and its clock frequency (system clock) operates between 2MHz - 90MHz.

**VII. INSTASPIN-FOC (GRAPHICAL USER INTERFACE)**

InstaSPIN-FOC [9] is a Graphical User Interface (GUI) with a complete sensor-less FOC solution embedded with on-chip ROM in preferred devices (FAST observer, FOC, speed and current loops). Effectively, it controls the motor without use of any sensors of speed and current.

F2806x peripheral drivers in user code, enable a monitoring InstaSPIN-FOC solution which identify and tune the torque controller resulting in efficiently control the motor within a fraction of time. This complete setup is called InstaSPIN-FOC, which is programmed in ROM.

In case of the F2806x devices, user has an option of working with all FOC functions in user memory (FLASH or RAM), which activates the FAST estimator firmware in ROM. InstaSPIN-FOC is designed for compliance, which accommodate a scope of system software architectures. FAST Estimator module restores the speed control method of motors using mechanical sensors. Field Oriented Control (FOC) of an electric motor propagates-superior torque control, lower torque ripples and improved efficiency when compared with traditional AC control techniques.

For finest dynamic response, rotor flux referenced control algorithms are preferred rather than stator flux referenced techniques. FAST estimator is included to measure rotor flux - magnitude and angle in a sensorless FOC system. The advantage of the FAST estimator is to provide automatic torque or current loop tuning, with preference given for user adjustments. Auto configuration of speed loop gains -  $K_p$  and  $K_i$  provides stable process for most of the applications. Automatic offset calibration assures quality samples of feedback signals.

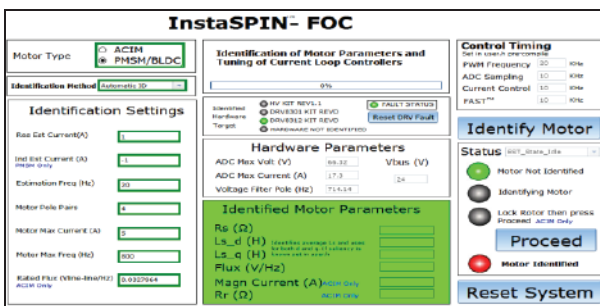


Figure 5. InstaSPIN-FOC initial Identification.

**VIII. BLDC DRIVE MODELLING IN VISSIM**

The VisSim modeling consists of “F28x Config” block and compound blocks like BLDC as shown below.

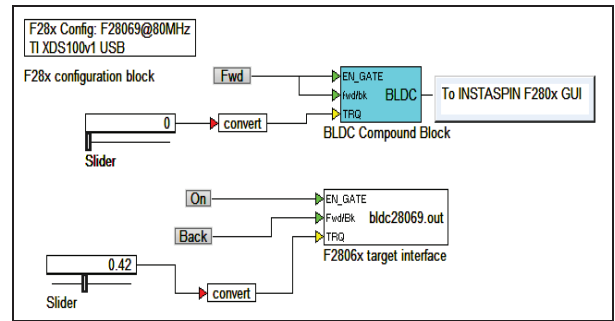


Figure 6. Modeling of BLDC Motor in VisSim.

The “F28x Config” block is used to configure the CPU as “F28069” and JTAG Connection with respective port number is identified. “Fwd” block is used to enable gate of the BLDC [8] motor. The “Slider” block allows the real-time control by moving cursor point in the simulation between lower and upper bounds. The “Convert” block is used to convert data type of input signal to scaled integer.

**IX. EVALUATING INSTASPIN-FOC**

“InstaSPIN\_FOC\_F2806xM” [9] is a GUI Composer evolution, that allows to interact with the model developed in VisSim and to evaluate the same using InstaSPIN-FOC.

*A. Hardware Set-up*

1. Verify the board existing has proper settings for all jumpers and switches
2. Install F28069M control CARD into the control CARD slot
3. Check Board Settings for Critical or Diagnostic states
  - a. TMDSCNCD28069MISO
    - i. SW1: TOP-TOP
    - ii. SW2: BOTTOM-BOTTOM
    - iii. SW3: TOP-TOP
  - b. DRV8312\_REVD
    - i. JP1: VR1 - Center
    - ii. JP2: Projected
    - iii. M1: High-Middle
    - iv. All three toggle switches in the center (MCU) position
    - v. Motor Phase Terminals to MOA, MOB, MOC
    - vi. DC Voltage of 24V to J9/PVDDIN

*B. Launch the GUI*

With low voltage DC power supply, connect USB from PC to J1 on F28069

- a. Run: Go to C drive and search for “web apps InstaSPIN\_FOC\_F2806x.exe” in c://ti/GUI composer location. The GUI Composer application will start as shown in Figure 7

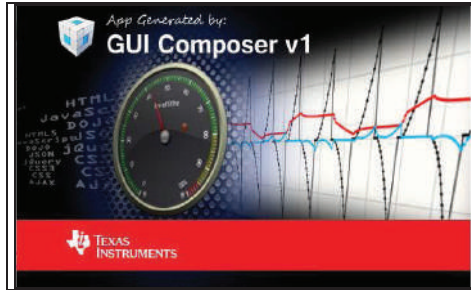


Figure:7. GUI Composer Start-up

- b. After GUI Composer is initialized, connect and then load “vissimblc.out” into the memory of Piccolo F28069M. The successful GUI interfaced window is displayed in Figure 8.

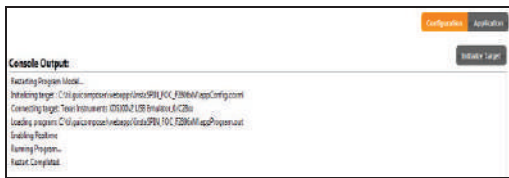


Figure 8. Normal Console Output of a successful GUI Launch

**X. SPEED CONTROL OF BLDC WITH INSTASPIN-FOC**

**A. Motor Identification**

Proceed only if the GUI properly diagnose the target hardware that has been used. Here, the Motor Type, Identified Hardware Target, Hardware Parameters and Control Timing values are automatically identified. Identification settings can be enabled as per requirement. Initially, when the application starts the Motor is not identified. It is indicated with a Green Blink and Motor is identified as Red Blink. When the Motor is under identification, Yellow Blink indicates Identifying Motor. When the Motor is completely identified Green blinks indicate Motor Identified.

Click on “Identify Motor” tab for identification of Motor. During identification of motor the indicator at “Identifying Motor” glows to “Yellow”. The Status changes to consecutive states while identifying and the percentage will be increasing. If the Motor Identification is successful the indicator turns to “Green”. For very low inductance motors, the change of value in “Estimation Frequency” tab of “Identification Settings” between 40 Hz and 60 Hz may result in successful identification. The Start-Up tab for Motor Identification is shown in the Figure 9.

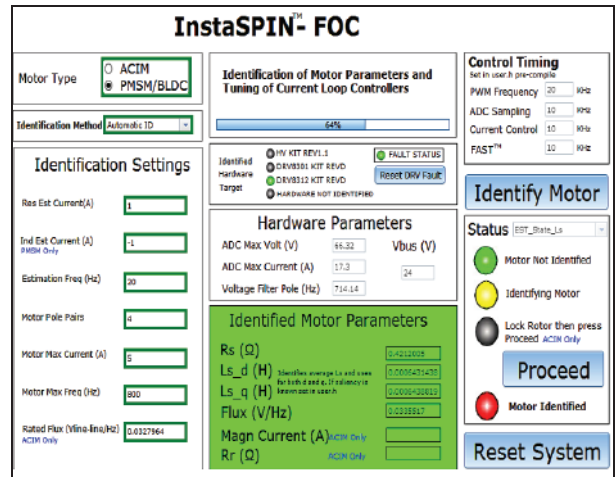


Figure 9. Start-Up Tab : Motor Identification

**B. Speed or Torque**

In Speed or torque tab, the values of Speed Reference and Max Acceleration have been set. In a short span of time Motor achieves the Target Speed with efficient Control Technique. The Display Panel shows

1. Estimated Speed (RPM) Tab from FAST
2. Target Speed (RPM) connected to the speed controller block
3. Voltage Bus (V)
4. Direction: “Blue” indicator represents clock wise direction and “Yellow” represents anti-clock wise direction

In Figure 10, before starting the Motor set the values of “Speed Reference” and “Max Acceleration” and tune for  $K_p$  and  $K_i$  values of the “Speed Controller”.

The first graph represents the Speed curves for both the “Reference Speed” and the “Estimated Speed”, second graph represents the Currents along Direct (d-axis) and Quadrature (q-axis) axes, and third graph represents the Torque produced for the drive operation.

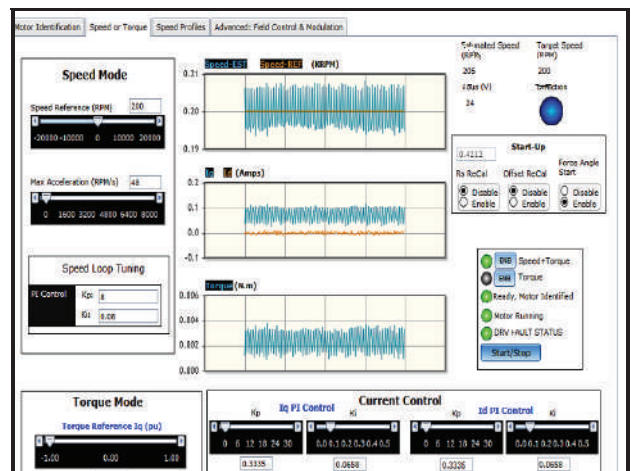


Figure 10. Speed or Torque tab

BLDC Motor operates Stable for  $K_p$  value within a range of 6 – 12 and  $K_i$  value not greater than 0.10. Default Speed Controller used is the PI, which is included in ROM as part of InstaSPIN-FOC.

The  $K_p$  and  $K_i$  gains are calculated from “user.h” selected from “Identification Method” drop down list.

- $K_p = 0.02 * \text{Motor Max. Freq.} * \text{Motor Max. Current (in pu)}$
- $K_i = 2.00 * \text{Motor Max. Freq.} * \text{ctrlPeriod\_sec} * \text{Motor Max. Current (in pu)}$

The current ( $I_q$  and  $I_d$ ) PI controllers can be altered during execution time also.  $K_p$  and  $K_i$  settings are obtained from the motor parameters and other scaling parameters can be fetched from user.h. The values are slightly set so that it can give good performance over wide variant types of hardware boards and motors. To stabilize the control of motor, increase the values of  $K_p$  and  $K_i$ . Further, increase in frequencies (speeds of 1 KHz+) can stiffen the motor control up to multiples of 4x that of the original values.

C. Speed Profiles

The “Speed Profiles” tab shown in Figure 11 is rendered with two different state machine motion profile examples to enable evaluation mode of common variable speed applications. Speed can be positive or negative. “Acceleration” or “Deceleration” mode should always be entered as a positive integer.

Capability is heavily dependent on the motor, design specifications, inertia, power supply, and load. The oscillations in the speed curves is sustainable only for few intervals and can be reduced by proper settings on the Speed Loop tuning block.

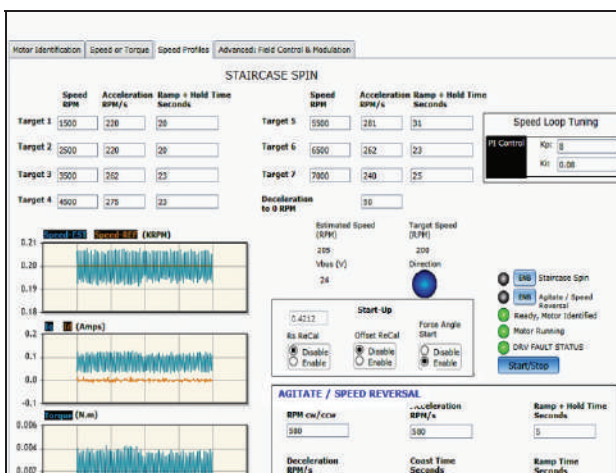


Figure 11. Speed Profiles tab

D. Field Control and Modulation

The tab of “Advanced: Field Control and Modulation” window shown in Figure 12 is used in conjunction with the other tabs, generally Speed and Torque, Staircase Spin (for field weakening), or Torque Mode (for field boosting). High

value of  $-I_d$  may result in demagnetization of a motor. Insure that

$$\sqrt{I_d^2 + I_q^2} < I_{rated} \quad (1)$$

Else, motor will demagnetize. Once operating in those modes, this tab can be used for further field control. Improper use of filed weakening mode may result in permanent damage of the motor and/or inverter.

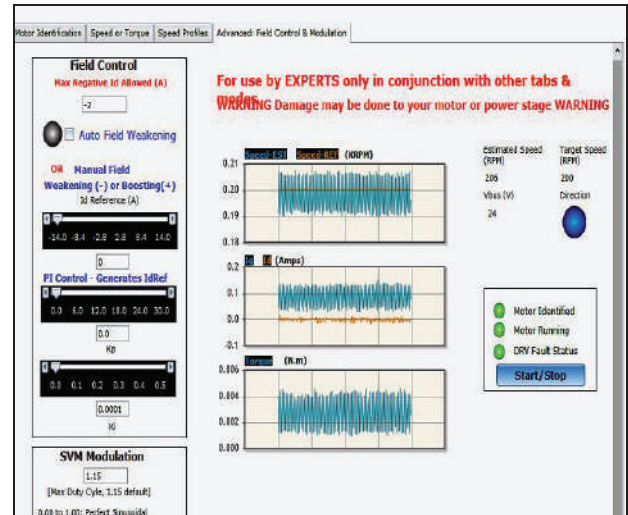


Figure 12. Advanced: Field Control and Modulation tab

XI. RESULT FOR THE SPEED CONTROL OF BLDC MOTOR

Speed control of BLDC Motor with an interface of DRV8312 EVM and InstaSPIN-FOC is shown in the Figure 13.

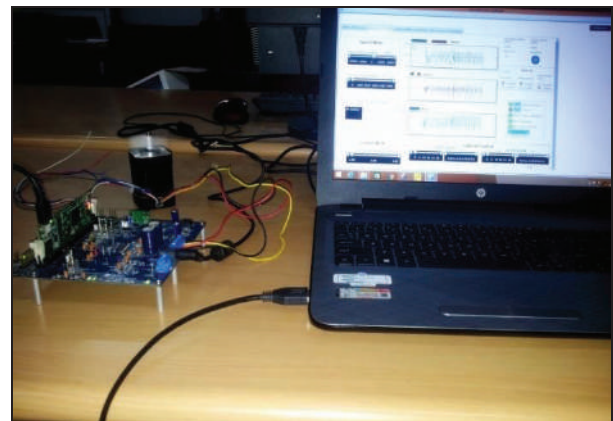


Figure 13. Interface of BLDC Motor with VisSim software

The model designed in VisSim generates a code in the back end and is dumped into the control card F28069 on the DRV8312 Evaluation module. Using the InstaSPIN-FOC GUI, interfacing is done with the BLDC Motor in order to control the speed of Motor. Figure 14 shows the Display panel (Motor tries to reach the target speed both in clockwise and anti clockwise direction.)

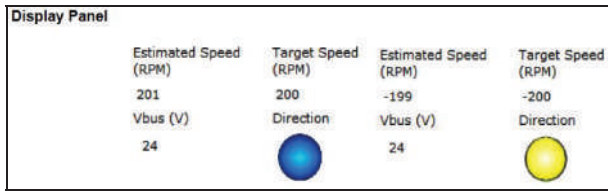


Figure 14. Display panel for Motor Speed Control

## XII. CONCLUSIONS

The paper reports modernization on controlling the Speed of Brushless D.C Motor with VisSim software. Using VisSim Software the design concepts involved and computed can be easily understood. The simulation method has been found to be simple and versatile, since governing equations and functions of each power electronics circuit can easily be represented in blocks. This paper would be useful for obtaining the speed performance for numerous power electronic circuits with wide combinations of motors.

The model has been developed and tested in software. VisSim provides a means of testing it with the real world, thereby delaying the need for prototyping still further. VisSim allows a range of ADC and DAC boards which helps in transferring information from signals and forward them to the hardware in a test plant. This means, function blocks within VisSim may be replaced by the real components and thus allowing the real system to be compared with the mathematical model.

The model may be taken a step further by automatically generating ANSI C program code from the VisSim model. This helps in creating an executable model which can be run

on any platform including transporters, DSP hardware or programmable controllers.

## REFERENCES

- [1] . O. Ustun and M. Yilmaz- "Simulation of power electronic circuits using VisSim software : a study on toolbox development" | IEEE Transaction Industrial Electronics. Istanbul Technical University, April2000, pp.183 – 187.
- [2] . O.Ustun, "Design of permanent magnet linear brushless D.C. Motor with Printed Circuit Armature", Ph.D thesis, Istanbul Technical University, April2000.
- [3] . Carlo Concari, Fabrizio Troni "Sensorless control of BLDC motors at low speed based on differential BEMF measurement" IEEE Conference Publications.
- [4] . MATLAB-Simulink, Math works Incorporated. 2013.
- [5] . VisSim User Guide, Visual Solution Inc., 1998.
- [6] . DRV8312-C2-KIT Hardware Reference Guide, Texas Instruments Incorporated. 2014.
- [7] . TMS320F2806x Piccolo Microcontrollers., Texas Instruments Incorporated., 2016
- [8] . Sensorless Trapezoidal Control of BLDC Motors, Texas Instruments Incorporated. 2013.
- [9] . InstaSPIN-FOC User guide, Texas Instruments Incorporated., 2013.