Acta Scientiarum. Technology
ISSN: 1806-2563
eduem@uem.br
Universidade Estadual de Maringá
Brasil

Acta Scientiarum

http://www.uem.br/acta
ISSN printed: 1806-2563
ISSN on-line: 1807-8664
Doi: 10.4025/actascitechnol.v39i3.28416

dSPACE real time implementation of fuzzy PID position controller for vertical rotating single link arm robot using four-quadrant BLDC drive

Manikandan Ramasamy[1*], Rajendran Somasundram[1] and Arulmozhiyal Ramasamy[2]

*[1]Department of Electrical and Electronics Engineering, Dhirajlal Gandhi College of Technology, Sikanampatty, 636309, Salem, Tamil Nadu, India. [2]Department of Electrical and Electronics Engineering, Sona College of Technology, Salem, Tamilnadu, India. *Author for correspondence. E-mail: electricmani@yahoo.co.in*

ABSTRACT. Automation has been growing in recent years for the manufacturing industries to increase productivity. Multiple robotic arms are used to handle materials for lifting in flexible directions. The vertical rotation of a 360 degree single arm is considered in this research on a position servo drive with brushless DC motor. The load torque of an arm varies depending upon the angular displacement due to gravity, so it requires four-quadrant operation of the drive with a robust feedback controller. This paper deals with the design and performance comparison of a conventional PID feedback controller with a fuzzy-based PID controller and suggests the most suitable controller. The design was implemented in real time through the dSPACE DS1104 controller environment to verify the dynamic behaviors of the arm.

Keywords: BLDC motor, dSPACE DS1104, four quadrant drive, fuzzy PID controller, position servo control, vertical rotating single arm robot.

Implementação dSPACE em tempo real do controlador de posição PID Fuzzy para robô giratório de braço único vertical utilizando unidade BLDC de quatro quadrantes

RESUMO. A automação vem crescendo nos últimos anos para que as indústrias transformadoras aumentem a produtividade. Robôs com múltiplos braços são usados para manusear materiais e levantar em direções flexíveis. A rotação vertical de um único braço em 360 graus é considerada nesta pesquisa em uma posição servo com motor DC sem escova. O torque de carga de um braço varia dependendo do deslocamento angular devido à gravidade, portanto, requer operação em quatro quadrantes da unidade com um robusto controlador de feedback. Este artigo trata da comparação de design e desempenho de um controlador de feedback PID convencional e um controlador PID fuzzy e sugere o controlador mais adequado. O design foi implementado em tempo real através do ambiente controlador dSPACE DS1104 para verificar os comportamentos dinâmicos do braço.

Palavras-chave: motor BLDC, dSPACE DS1104, drive de quatro quadrantes, controlador PID fuzzy, controle de posição servo, robô giratório de braço único vertical.

Introduction

Brushless direct current (BLDC) motors are widely used in position servo drive applications. They have less moment of inertia, so are more suitable for quick start and stop operations. In recent years, BLDC motors have been available in different milli-wattage to kilo-wattages as per position servo drive requirements, for example in the automotive, aerospace, consumer and medical fields, for industrial automation robotic arms (Hernandez, Santibanez, & Campa, 2008). Especially industrial automation requires more robust position control in different operating environments and this requires four-quadrant operation, which allows bidirectional speed control with regenerative braking capabilities (Manoharaprasad & Raja, 2014). The single arm of the robot moves vertically or horizontally with one degree of freedom (DOF) around 360°. During its horizontal motion, there is no change in the torque of the payload because gravitational force acts on the center of the arm at a constant magnitude. But during vertical motion, the payload torque varies depending on its position due to the gravitational force in the centric load (Klafter, Chmeliewski, & Negin, 2009). A varying load torque in nature is required to maintain constant speed to attain the desired position without delay, so the designs of feedback controllers are an important consideration in this respect. The conventional PID feedback controller depends on the accuracy of the mathematical model of the system and the expected performance is not met due to the load disturbances. Fuzzy logic is a technique to embody human-like thinking into a control system. The fuzzy-based PID controller gives better dynamic performance as well as error reduction. This paper discusses the position control drive of a vertical rotating single arm using conventional PID and fuzzy-based PID controllers (Rodriguez & Emadi, 2007).

Material and methods

Four-quadrant drive of single arm

Figure 1 shows the arm rotating vertically in a 360° schematic and the relevant load torque curve. Whenever the arm moves from 0° to 180° in the clockwise direction, the motor should rotate in the forward direction to lift the payload. The load torque tries to pull down the arm and its magnitude increases with respect to the change of displacement. Hence the motor needs to oppose the load torque and work in forward motoring mode to reach the desired position. When the arm reaches 90°, then the load torque is at its maximum and will reduce to zero when it reaches 180°, so its speed will reduce with increasing load. However the load varies, we need to maintain constant speed; otherwise, the time of reaching the desired position will be delayed.

When the arm rotates in a downward direction of 180° to 360° in a clockwise direction the gravitational force pulls down the arm and hence the speed and current may exceed the safe limit. This will mean the desired arm position is faster than motoring mode. The load torque during this period is negative but the motor wants to rotate in a forward direction with maximum negative torque at 270°. However, a negative torque motor should operate in forward braking mode; otherwise, the arm may drop the payload due to gravity, which may leads to awkward incidents (Joice, Paranjothi, & Kumar, 2013).

Similarly, when the arm rotates in an anticlockwise direction, the same scenario occurs. With reverse rotation of the arm from 0° to -180° (360° to 180°), the motor turns in a negative direction to lift the payload and this gives negative torque. This requires the motor to operate in reverse motoring mode. In anticlockwise rotation of the arm from -180° to -360° (180° to 0°), the load torque will act in a positive direction. But the motor rotates in the reverse direction and it is required to operate the motor in reverse braking mode. Hence the curve between the arm position and the load torque resembles a sine waveform in both directions. So the load applied to the motor shaft (T_m) is defined as the trigonometric sin function of the actual position multiplied by the payload torque (T_L) as shown in Equation (1).

$$T_m = T_L \sin(\theta) \tag{1}$$

where:

T_m - Mechanical shaft torque given to motor in Nm

T_L- Pay load torque added with arm in Nm

θ- Arm position in degrees.

The block diagram of the proposed position servo drive is shown in Figure 2.

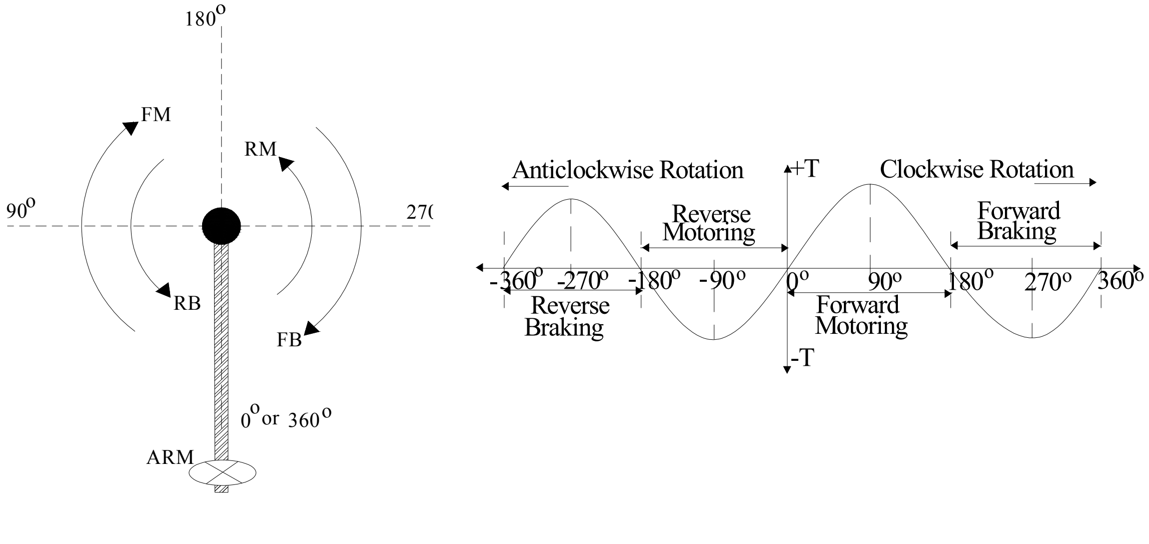


Figure 1. Single link vertical rotating arm and torque with effect of gravity as a function of angle.

A single arm robot is directly coupled with a motor shaft and a quadrature encoder is coupled with the BLDC motor, which can be used to measure the actual position. Some applications require a self-locking gear arrangement to provide the holding torque. Quadratic encoder pulse (QEP) signals Q_a and Q_b are decoded through the quadrature pulse decoder, which will give actual position information. The actual position and set position are compared to generate the position error and then it is given to the position controller. This may be a conventional PID or fuzzy PID controller as required. The output of the position controller gives the magnitude of the reference DC link current I_{dc}* and it is used in the range of safe limits. The reference current is compared with the actual measured I_{dc} to generate the current error to feed the current controller. The current controller may be a conventional PID or fuzzy PID as required. The current controller will act in the inner loop and the position controller will act in the outer loop of the control system.

The output of the current controller will give the duty cycle of the gate pulse and the direction to be run. The BLDC motor is self-controlled through Hall position information. The Hall signals are decoded to generate the gate pulse through the Hall decoder and this will be modulated with the duty cycle information coming from the current controller to generate six pulse width modulated (PWM) pulses. The generated PWM pulses are fed to a six MOSFET or IGBT switches based inverter to drive the three-phase BLDC motor as shown in Figure 3. The available AC supply is converted to constant DC through an uncontrolled diode rectifier.

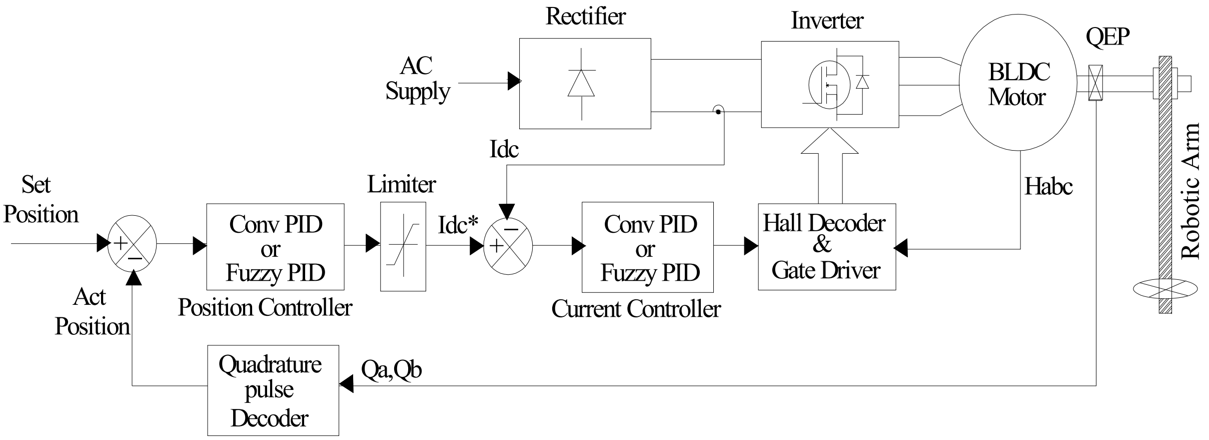


Figure 2. Block diagram of arm position servo drive.

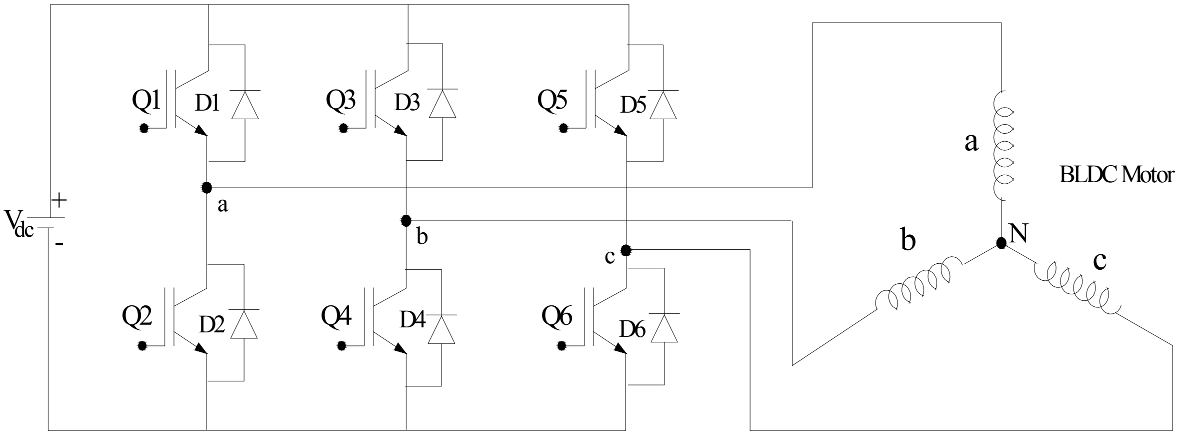


Figure 3. Six switch based three-phase inverter.

dSPACE implementation

The laboratory hardware test bench is developed through the dSPACE DS1104 DSP controller with a 1H.P brushless DC motor. A TETRA 85TR2.2 series BLDC motor from the Motor Power Company Italy is considered for a directly coupled vertically rotating single link arm and the specifications are shown in Table 1.

Table 1. Motor specification.

Parameter	Value
Voltage	310V dc
Speed	4600 rpm
Torque	2.2 Nm
Stall current	4.52 A
Poles number	4
Voltage constant (Ke)	51 V krpm^{-1}
Torque constant (Kt)	0.49 Nm A^{-1}
Ph/ph resistance	3.07 Ω
Ph/ph inductance	6.57 mH
Moment of inertia (Jm)	1.8 kg cm^{2}
Encoder	2000 ppr

A three-phase intelligent power module (IPM) is used to drive the motor, built with a six- IGBT isolation and gate driving unit, a diode rectifier with protection circuits and signal conditioning units. A single-phase auto transformer is used to supply the driving power from a 230V AC line.

Conventionally, the motor driving signals are generated by using programmable integrated chips, in which the coding is complex, very costly, time-consuming and therefore difficult to handle. In recent years, researchers have used dSPACE-based DSP controllers for integrating the hardware-software platform for MATLAB-Simulink based graphical programming. A dSPACE DS1104 controller board is used with a connector panel to control the real-time interfaces (Rubaai, Castro-sitiriche, & Ofoli, 2008).

The hardware implementation block diagram is shown in Figure 4. A host PC connected with dSPACE is used to program through MATLAB-Simulink and monitor the motor's real- time performance, including current, speed and position through a control-desk software virtual panel (Mingliang, Jianqi, & Cenwei, 2011). A DS1104 controller with an inbuilt Texas Instruments C24x based 16-bit slave DSP is used for PWM generation, ADC, digital I/O capture and QEP modules. The motor's Hall signals are fed to the controller via a Hall sensor signal interface card to match the voltage level. A dSPACE DS1104 adaptor card is used to link the connector panel and the IPM to proper specified connector standards. Hall signals are acquired by the high-speed slave capturing unit, and gate PWM pulses are generated in the slave DSP. The motor's scaled current is acquired through the ADC unit and motor coupled encoder signals are acquired through the QEP unit. The DAC is used to generate the position and speed equivalent voltage for monitoring through a digital storage oscilloscope (DSO).

Simulink-based graphical programming of the proposed drive is shown in Figure 5. Figure 6A shows the Hall sensor subsystem used to acquire Hall signals via slave capture. An adaptor card inbuilt with logic inverters is used for buffering the signals. In order to maintain the original signals, every acquired and generated signal is programmed for logic inversion to resemble the original signals.

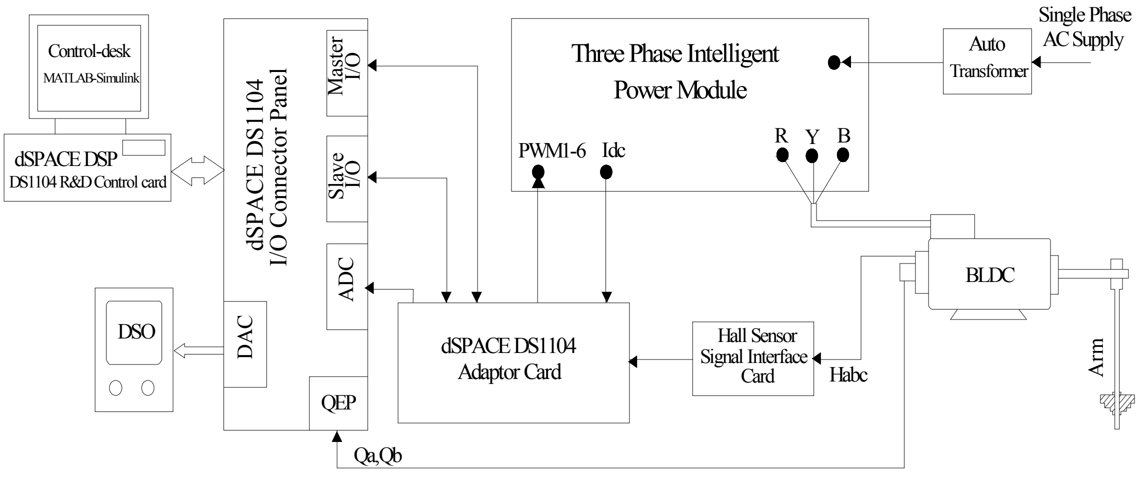


Figure 4. dSPACE hardware implementation block diagram.

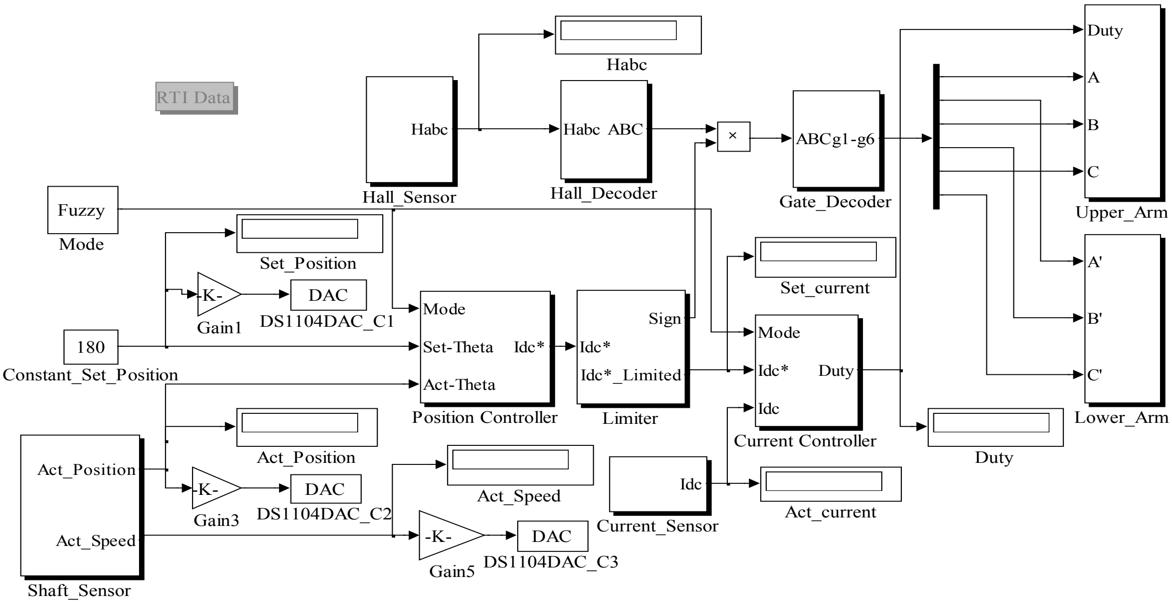


Figure 5. MATLAB-Simulink based graphical programming.

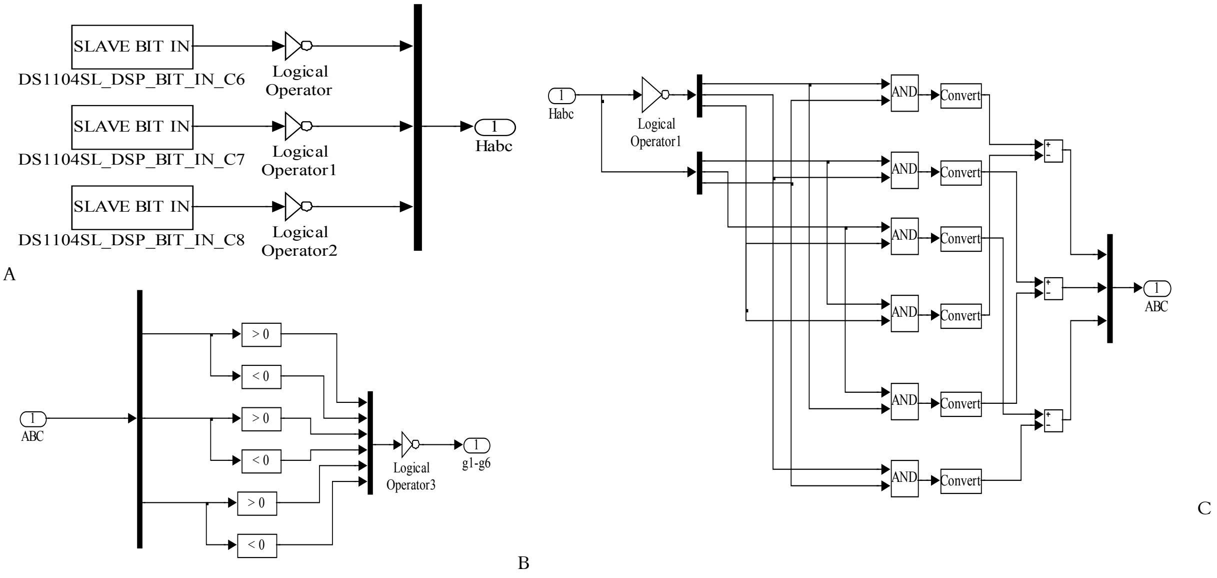


Figure 6. Subsystems: A. Hall sensor; B. gate decoder; C. Hall decoder.

Hall signals decoded in the subsystem are used to generate six- gate pulses shown in Figure 6B and C as per the commutation Table 2, where ~1 indicates switches in chopping and 1 indicates the ON (HIGH) and 0 the OFF (LOW) condition. In general, for the 120° conduction mode of the inverter two switches are kept in the ON position at any instant, one from the upper arm and another from the lower arm, so that one switch chopping is enough to control the voltage in order to reduce the switching voltage stress. Upper arm switches are chopped as shown in Figure 7A at 10 kHz chopping frequency and the lower arm switches are kept at ON and OFF through master digital outputs as shown in Figure 7B.

Table 2. Commutation table.

Rotor position (electrical θ)	Hall sensors states			Switch states						Phase current		
	Ha	Hb	Hc	Q1	Q2	Q3	Q4	Q5	Q6	Ia	Ib	Ic
0° – 60°	0	1	0	0	0	0	1	~1	0	0	-	+
60° – 120°	1	1	0	0	1	0	0	~1	0	-	0	+
120° – 180°	1	0	0	0	1	~1	0	0	0	-	+	0
180° – 240°	1	0	1	0	0	~1	0	0	1	0	+	-
240° – 300°	0	0	1	~1	0	0	0	0	1	+	0	-
300° – 360°	0	1	1	~1	0	0	1	0	0	+	-	0

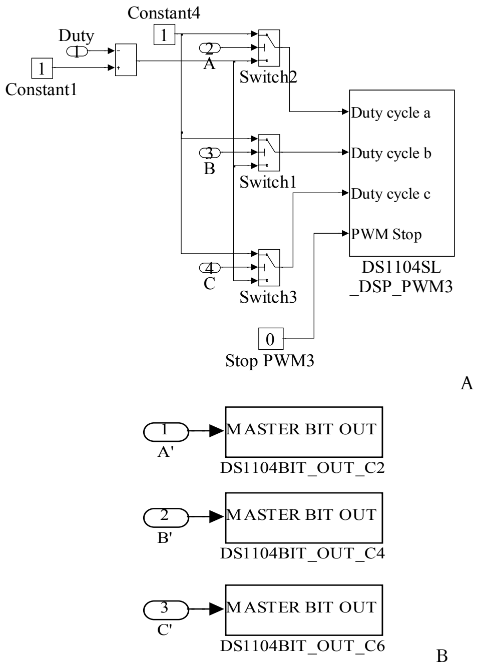


Figure 7. Subsystems: A. upper arm; B. lower arm.

The shaft sensor subsystem shown in Figure 8A is used to measure the position and speed of the motor. The quadrature encoder model will increment or decrement 0.25 pulse counts as per the phase shift of quadrature signals Qa and Qb in each and every edge. The encoder has 2000 slots and every slot gives one rising and falling edge. By combining the two signals Qa and Qb, four edges are possible per slot, which makes 4 × 0.25 = 1 pulse count. Hence, one revolution of the motor makes 2000 pulse counts, and thus increases the resolution of the drive. The pulse counts are multiplied with a gain of 360/2000 to get the actual position in degrees. The speed can be calculated from the delta position, which provides the difference of the pulse count value from the last to the current sample step, measured in encoder lines and multiplied with a gain of 60 / 0.0001 × 2000 as shown in Equation 2. The average value of around 10 speed information samples is taken to avoid noise at low speed. The measured speed is used for monitoring only and it is not used for any controlling purpose.

$$N = \frac{60}{T_s \times ppr} \times \Delta PulseCounts \qquad (2)$$

where:

N - speed in rpm

T_s - sampling time 0.0001 seconds

ppr - 2000 pulses per revolution of encoder

Δ - difference of pulse count value between two samples.

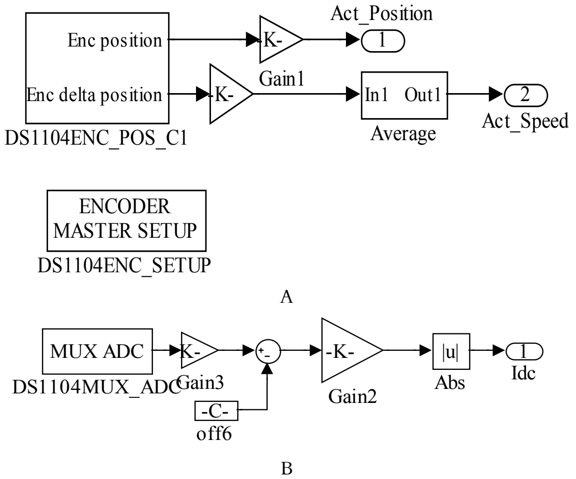


Figure 8. Subsystems: A. shaft sensor; B. current sensor.

The Hall current sensor with a signal conditioning unit in IPM is used to measure the DC link current. The analog equivalent voltage of the DC link current, acquired through the ADC as shown in Figure 8B, is offset and scaled to get the original current magnitude. Figure 9 shows the complete picture of the hardware implementation.

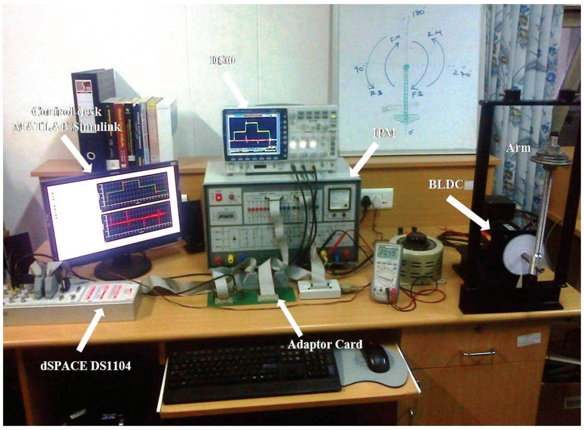


Figure 9. Real-time implementation photo view.

Design of controllers

The feedback controller is the heart of the servo driver. The design of the feedback controller requires two feedback closed loops, of which one is the inner loop current controller to limit the maximum current for protection of the motor and the second one is the position controller in the outer loop to maintain the arm at the set desired position. In common industries, conventional PID controllers are used for the feedback system due to their simplicity and easy operation. But the conventional

PID controller shows poor performance during load disturbance and during changes of command. Also, the design of the control gain requires more mathematical modeling. To overcome these drawbacks, researchers have been using fuzzy-based PID controllers in recent years. Fuzzy PID does not require complex mathematical modeling and is more suitable in all working environments (Hongli, Duan, Cai, & Jia, 2008). It also reduces the error and the responses are faster.

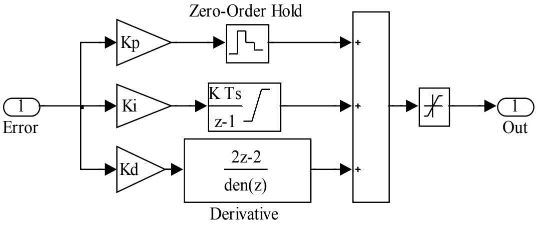


Figure 10. Conventional PID controller.

The Simulink model of the conventional PID controller as shown in Figure 10 is used in both the position and current controllers, but their gain values K_p, K_i and K_d differ according to the position and current magnitude of the working ranges. According to PID control law, the mathematical formulation of a discrete controller is formed as per Equation (3). The control gain values are determined using the Ziegler–Nichols method of tuning.

$$C(z) = K_p + K_i T \frac{z}{z-1} + \frac{K_d}{T}\frac{z-1}{z} \quad (3)$$

where:

K_p- proportional gain

K_i- integral gain

K_d- derivative gain

T- sampling time in sec.

In order to improve the performance of the feedback controller, the conventional PID controller was replaced with a fuzzy-based PID controller. Figure 11 shows the combination of fuzzy PI with fuzzy PD to implement the fuzzy PID controller. Error and change of measurement are the input to the controllers. Using a change of measurement instead of a change of error is in order to prevent a step change in the reference signal from directly triggering the derivative action.

The range of membership is a function used as unity ± 1, and hence it is required to scale the control signals before and after fuzzy inference. The reference current magnitude I_{dc}* is the output of the outer loop position controller, and the gate pulse duty cycle is the output of the inner loop current controller. Inputs are scaled through the error normalization factor GE and change in the measurement normalization factor GCE. The output response signal is scaled through the response de-normalization factor GU and change in the response de-normalization factor GCU. These scaling factors play a vital role in the design of the fuzzy PID controller.

The scaling factors are derived from conventional PID controller gains as per Equations 4–7. One feed forward path is provided to ensure the working of proportional action when the fuzzy PID controller is linear.

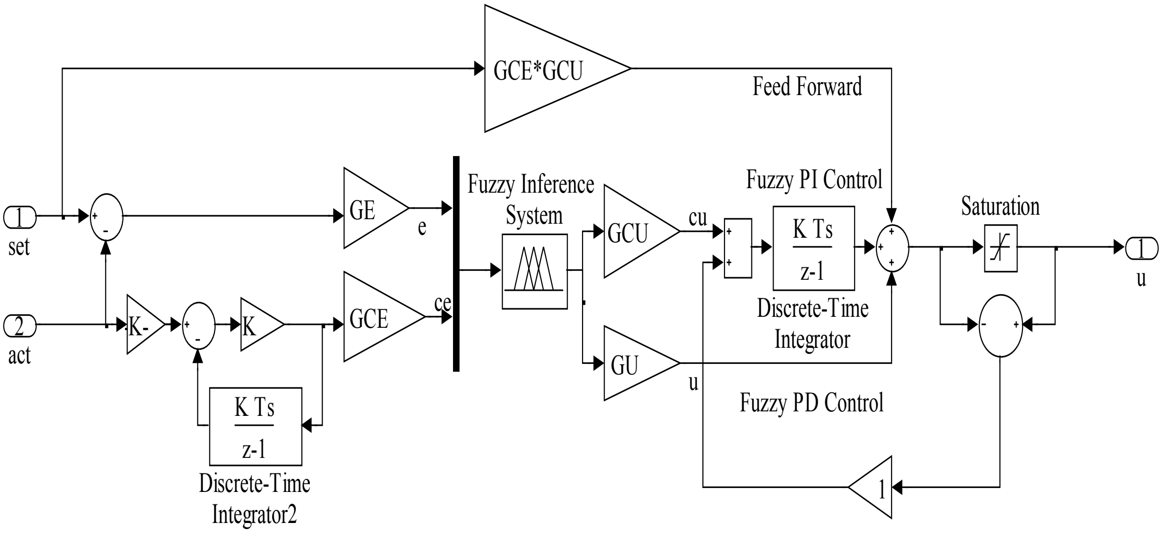


Figure 11. Fuzzy PID controller.

An anti-windup feature is included to control the saturation level of the output response signal, so as to improve the quick response of the fuzzy PID controller.

$$GE = \frac{1}{\max.error} \tag{4}$$

$$GCE = GE * \left(K_p - \sqrt{K_p^2 - 4K_i K_d}\right)\frac{K_i}{2} \tag{5}$$

$$GU = \frac{K_d}{GCE} \tag{6}$$

$$GCU = \frac{K_i}{GE} \tag{7}$$

where:

GE - error normalization factor;

GCE - change in measurement normalization factor;

GU - response de-normalization factor;

GCU - change in response de-normalization factor. The input and output variables are mapped via membership functions and they are triangular as shown in Figure 12.

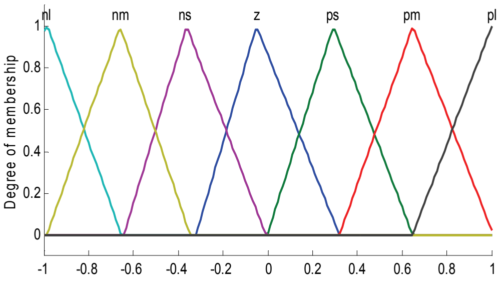


Figure 12. Membership functions of 'e', 'ce' and 'u'.

Mamdani-type inference is used for the inference engine and the center of gravity method is used for defuzzification. The linguistic variables are divided into seven groups, which are nl - negative large, nm - negative medium, ns - negative small, z-zero, ps- positive small, pm - positive medium and pl - positive large. The inference engine works based on rules shown in Table 3, where 7 × 7 = 49 rules are possible in the matrix.

Table 3. Fuzzy rule-based matrix.

ce \ e	nl	nm	ns	z	ps	pm	pl
nl	nl	nl	nl	nl	nm	ns	z
nm	nl	nl	nl	nm	ns	z	ps
ns	nl	nl	nm	ns	z	ps	pm
z	nl	nm	ns	z	ps	pm	pl
ps	nm	ns	z	ps	pm	pl	pl
pm	ns	z	ps	pm	pl	pl	pl
pl	z	ps	pm	pl	pl	pl	pl

Results and discussion

Several tests were conducted with both the conventional PID and fuzzy PID controllers in the laboratory with a 35 cm long single arm carrying a constant 1 Nm payload torque equivalent weight in all four quadrants. The set position and actual position of the arm and the corresponding speed equivalent voltages are captured in the DSO. Channels 1 and 2 show 25 mV per 1° of set and actual position respectively. Channel 3 shows 6.666 mV per rpm of speed.

When the arm moves upward in a clockwise direction from 0° to 180°, the motor rotates in the forward direction. To make the arm reach 90°, it requires maximum torque due to gravitational force, so that the arm pulls down the payload to reach 180° at reduced speed and thereby takes longer to reach the desired position. The conventional PID controller takes 0.37 seconds to reach the desired position, as shown in Figure 13 A. The fuzzy PID controller takes 0.25 seconds settling time (Ts), as shown in Figure 13B. The comparative study results relating to the dynamic behaviors, like the rising time (Tr), maximum overshoot (Mp) and ± percentage of steady state error listed in Table 4, prove that the fuzzy PID gives a better performance than the conventional PID controller.

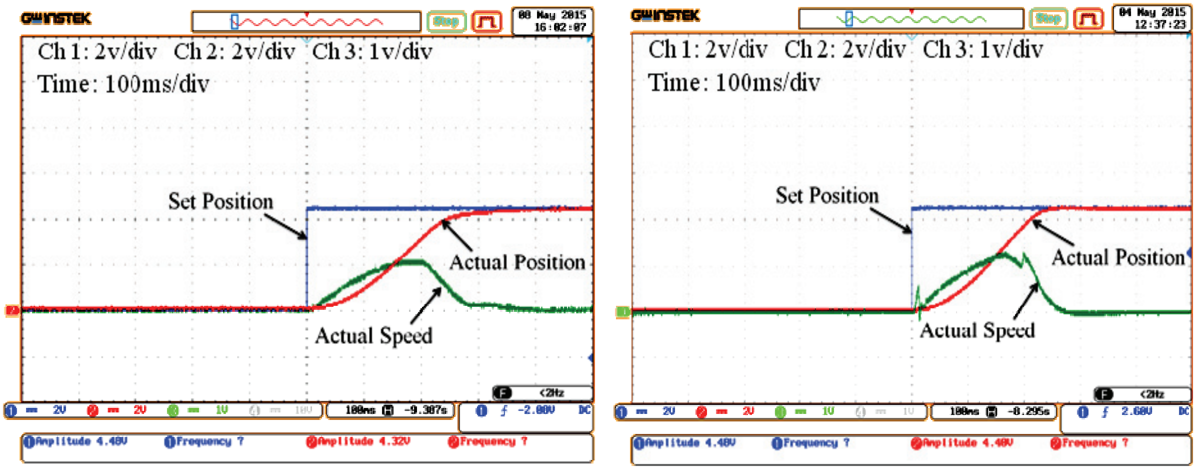
When the arm moves downward from 180° to 360° in a clockwise direction, it moves faster than when in the upward motion due to gravitational force. It operates in the second quadrant of forward braking at a safer speed in order to avoid accidents. With the conventional PID the arm reaches 360° in 0.21 seconds as shown in Figure 14A, which is faster than the time taken in the motoring mode. But with fuzzy PID it takes 0.25 seconds as shown in Figure 14B, the same time as that taken in motoring mode.

When the arm rotates in the reverse direction from 360° to 180° upward and anticlockwise, the motor needs to operate in the third quadrant of reverse motoring mode. The time taken by the conventional PID is 0.37 seconds as shown in Figure 15A whereas the fuzzy PID controller takes 0.25 seconds as shown in Figure 15B, which is the same time as in forward motoring mode.

When the arm moves downward in an anti-clockwise direction of 180° to 0° in the fourth quadrant of reverse braking mode, the settling time taken is the same as in forward braking mode with both the controllers, as shown in Figures 16

A and B. The conventional PID controller produces the maximum overshoot of around 13.88% and more oscillations. The results presented in Table 4, show that the fuzzy PID controller has a smaller percentage of steady state error and takes a uniform settling time in all four quadrants.

To compare the performance of the controllers, a load impact analysis is conducted with steady state conduction. When the arm is at 90°, it needs the maximum torque with a 1 Nm connected load. Further 1 Nm load torque equivalent weight is added and removed after 4 seconds. The sudden addition and removal of load makes the arm pull down and push up respectively, with some percentage of position error. The alteration of the load to the arm, with the conventional PID controller, impacts the position up to ± 25° and the arm reaches steady state after 1 second with an error of 3% as shown in Figure 17A. However, the impact and load alteration to the arm with the fuzzy PID controller to the position is ± 15° and it takes 0.5 seconds to reach the steady state as shown in Figure 17B, without any error. Thus the load impact analysis proves that the fuzzy PID controller is robust and a better controller than the conventional PID controller.



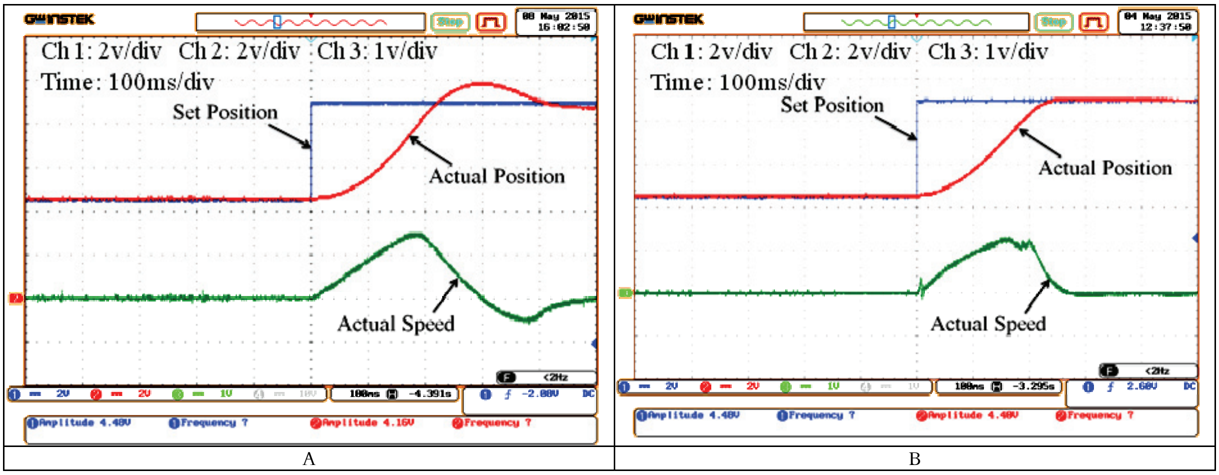
A

B

Figure 13. Forward motoring mode [0° to 180°]: A. conventional PID controller response; B. fuzzy PID controller response.

Table 4. Results comparison.

Controller / Parameter / Set Position	Conventional PID				Fuzzy PID			
	Tr (sec.)	Mp (%)	Ts (sec.)	± Error (%)	Tr (sec.)	Mp (%)	Ts (sec.)	± Error (%)
0° to 180°- [Forward Motoring]	0.25	0	0.37	-3.77	0.15	2.77	0.25	-0.22
180° to 360°- [Forward Braking]	0.15	13.88	0.21	-5.55	0.15	0	0.25	-3.55
360° to 180°- [Reverse Motoring]	0.25	0	0.37	-3.77	0.15	0	0.25	-3.55
180° to 0°- [Reverse Braking]	0.15	13.88	0.21	-5.55	0.15	1.77	0.25	-2.44



A

B

Figure 14. Forward braking mode [180° to 360°]: A. conventional PID controller response; B. fuzzy PID controller response

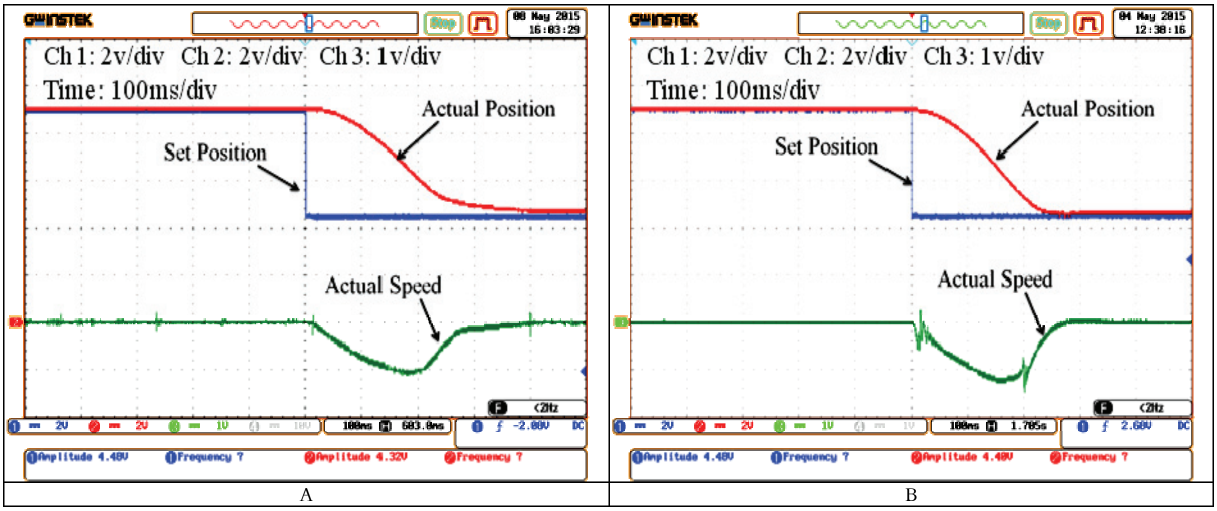


Figure 15. Reverse motoring mode [360° to 180°]: A. conventional PID controller response; B. fuzzy PID controller response

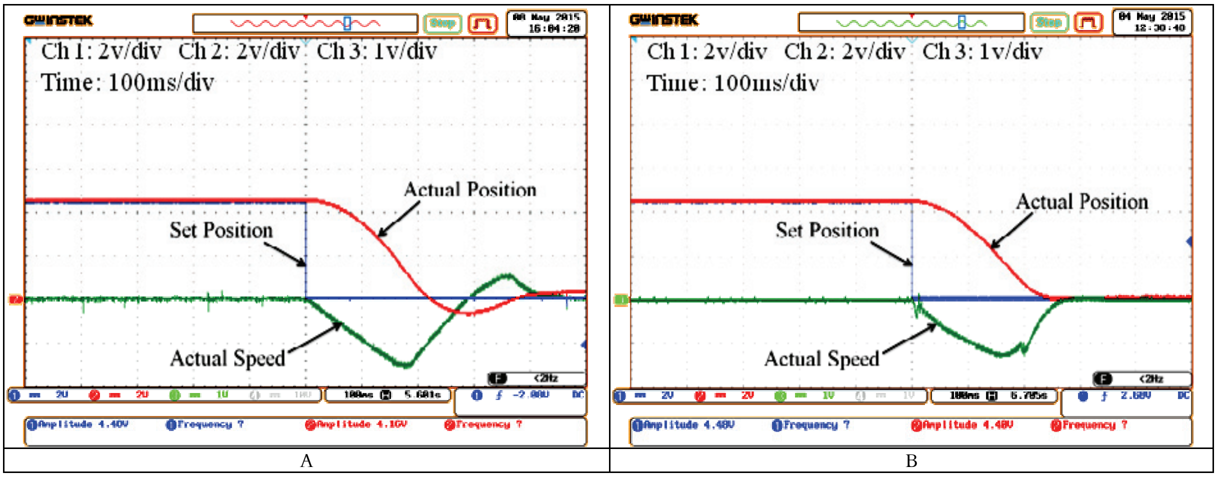


Figure 16. Reverse braking mode [180° to 0°]: A. conventional PID controller response; B. fuzzy PID controller response.

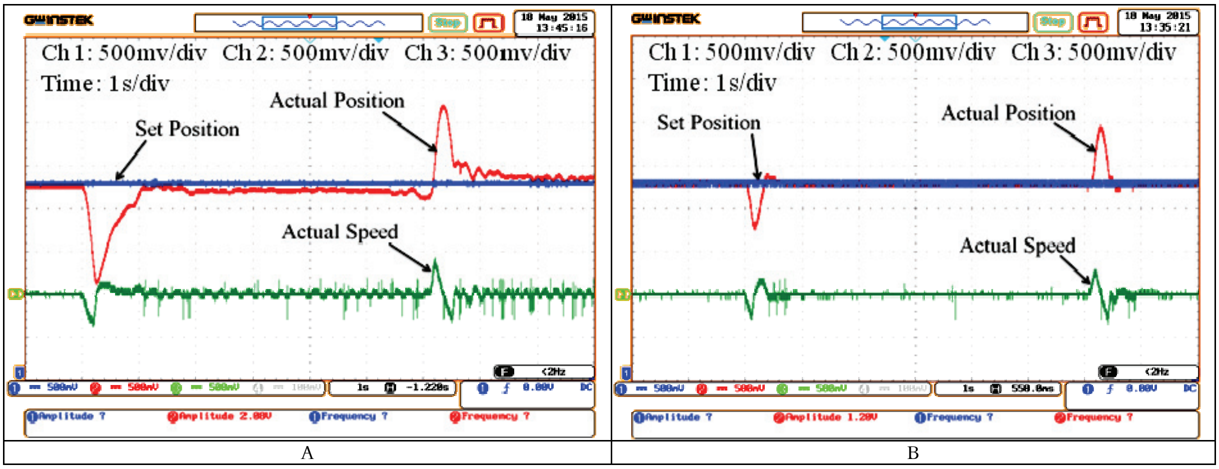


Figure 17. Load impact analysis: A. conventional PID controller response; B. fuzzy PID controller response.

Conclusion

In this paper, a control scheme is proposed for the position servo drive of a vertical rotating single-arm robot. The motor is made to operate in all four quadrants and also under varying load conditions. From the evidence of the dSPACE DS1104-based hardware implementation results, the fuzzy-based PID controller will reduce the settling time and maintain the same interval during all modes. By comparison of other dynamic behaviors with conventional PID controllers, the fuzzy PID controller will give robust control of the drive.

References

Hernandez, G., Santibanez, V. M., & Campa, R. (2008). PID control of rigid robots actuated by brushless DC motors. *American Control Conference, 1*(1), 1430-1435.

Hongli, L. V., Duan, P., Cai, W., & Jia, L. (2008). Direct conversion of PID controller to fuzzy controller method for robustness. *IEEE Conference on Industrial Electronics and Applications, 1*(1), 790-794.

Joice, C. S., Paranjothi, S. R., & Kumar, V. J. S. (2013). Digital control strategy for four quadrant operation of three phase BLDC motor with load variations. *IEEE Transactions on Industrial Informatics, 9*(2), 974-982.

Klafter, R. D., Chmeliewski, T. A., & Negin, M. (2009). *Robotic engineering: an integrated approach*. India, IN: PHI Learning Pvt.

Manoharaprasad, A., & Raja, C. V. N. (2014). Control of two-link Scara robot using IMC PID and fuzzy controllers. *International Journal of Emerging Trends in Engineering and Development, 2*(4), 248-258, 2014.

Mingliang, H., Jianqi, Q., & Cenwei, S. (2011). A comparative analysis of fuzzy PI and PI speed control in Brushless DC motor based on dSPACE. *International Conference on Electrical Machines and Systems, 1*(1), 1-5.

Rodriguez, F., & Emadi, A. (2007). A novel digital control technique for brushless DC motor drives. *IEEE Transactions on Industrial Electronics, 54*(5), 2365-2373.

Rubaai, A., Castro-sitiriche M., & Ofoli, A. DSP-based laboratory implementation of hybrid fuzzy-PID controller using genetic optimization for high-performance motor drives. *IEEE Transactions on Industry Applications, 44*(6), 1977-1986.

Received on 3 september, 2015.
Accepted on 27 november, 2015.