*UDC 621.313.13.519.61*
*Original scientific paper*
*Received: 17.07.2006.*

# Digital speed controller implementation for a switched reluctance motor drive using FPGA

**Subramanian Vijayan**[(1)], **Shanmugam Paramasivam**[(2)], **Rengasamy Arumugam**[(1)], **M. Vasudevan**[(1)] and **Shubhransu S. Dash**[(1)]
[(1)]Department of Electrical and Electronics Engineering, Anna University, Chenai - 600 025, INDIA
e-mail: arumugam@annauniv.edu; vasumame@yahoo.com; munu_dash_2k@yahoo.com
[(2)]ESAB Engineering Services LTD, INDIA
e-mail: paramasivam_s@rediffmail.com and paramsathya@yahoo.com

**SUMMARY**

*In this paper, a Field-Programmable Gate Array (FPGA) based digital speed control scheme is presented, that is developed to overcome the drawbacks existing in the previous speed control schemes, which were proposed for switched reluctance motor (SRM) drives. It is based on discrete P, PI and PID control algorithm, and requires simple mathematical models. The scheme is implemented by using a XC2S300E FPGA. The real-time experimental results given in this paper show that the speed control method proposed could provide accurate speed control and over a wide range of speeds, and can also perform accurately at different operating conditions (steady state/transient operation under soft chopping mode). The closed loop SRM speed control system is seen to achieve 6.2 RPM speed accuracy, depending on the needed operating speed range, with a step response settling time of 0.25 to 1.05 seconds. Complete descriptions of the experimental system along with FPGA implementation are presented.*

***Key words**: Switched Reluctance Motor (SRM), speed control, discrete P, PI and PID controller, FPGA.*

## 1. INTRODUCTION

Switched Reluctance Motors (SRM) have been used for many years in applications, where the simplicity of construction was of primary importance. It is a doubly salient, singly excited motor. That is, the SRM has salient poles on both the rotor and the stator, but only the stator poles carry windings. The rotor tries to get to a position of minimum reluctance by aligning itself with the stator magnetic field when the stator winding is excited [1, 2]. Thus, exciting the stator phase windings of the motor in a particular sequence and consequently, controlling the stepping magnetic field, we can control the movement of the rotor. The availability of low cost power semiconductor switches, dedicated digital controllers and excellent speed torque characteristics of the SRM have paved the way for the applications of those machines to more demanding industrial applications. The advantages like less weight, higher efficiency also favour these machines to be more suitable to electric vehicles and aircrafts [1-10]. Many papers have been reported on the performance simulation of SRM with experimental validation for different control strategies such as a feed back linearization control, variable structure control, fuzzy logic control, self tuning control and four quadrant operation of SRM [11-16]. None of these papers have achieved very low speed error during various operating conditions. Hence, the implementation of a low cost digital speed control of SRM using XC2S300E FPGA with a simplified design, reduced circuit areas and parts count compared to the conventional controllers is desired. This paper is organized as follows. Section 2 reviews the SRM description. Section 3 discusses the controller implementation. Section 4 discusses the experimental setup. Section 5 discusses the Field-Programmable Gate Array (FPGA) implementation. Section 6 discusses the results and discussion. Section 7 discusses the conclusion.

## 2. SRM DESCRIPTION

A prototype SRM has 6 stator poles and 4 rotor poles as shown in Figure 1.

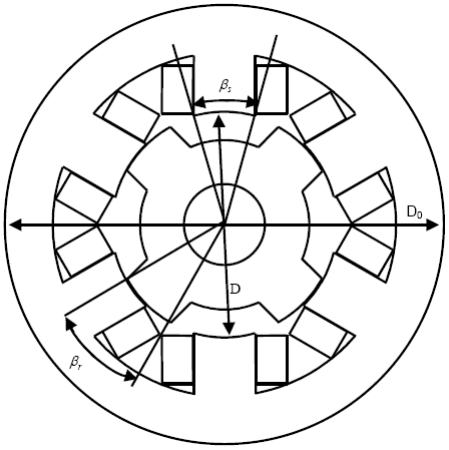



*Fig. 1 6/4 pole prototype SRM*

Its inductance profile is a periodic function with respect to the rotor position and has symmetric structure. The inductance ($L$) variation over one rotor pole pitch with constant current magnitude is shown in Figure 2. The phase inductance is maximum when the rotor pole is aligned with the stator pole and is minimum when the rotor pole is aligned with the inter polar axis of the stator [1, 2]:

$$L(\theta)=\begin{cases} L_{min} & \theta_0 \le \theta p \theta_1 \\ L_{min}+p\theta & \theta_1 \le \theta p \beta_s \\ L_{max} & \beta_s \le \theta p \beta_r \\ L_{max}-p(\theta-\beta_r) & \beta_r \le \theta p \beta_s+\beta_r \\ L_{min} & \theta_2 \le \theta p \theta_3 \end{cases} \qquad (1)$$

where:

$L(q)$ - Inductance variation over one rotor pole pitch
$L_{min}$ - Unaligned inductance ($H$)
$L_{max}$ - Aligned Inductance ($H$)
$b_s$ - Stator pole arc ($rad$)
$b_r$ - Rotor pole arc ($rad$)

$$\beta_r > \beta_s \text{ and } p=\frac{L_{max}-L_{min}}{\beta_s}$$

The per phase equivalent circuit of the SRM can be drawn as shown in Figure 3.


*Fig. 3 Per phase electrical equivalent circuit of SRM*

$$V(t)=Ri(t)+L(\theta,i)\frac{di}{dt}+i\varpi\frac{dL(\theta,i)}{d\theta} \qquad (2)$$

where:

$V$ - is the voltage applied to the phase,
$R$ - is the phase resistance,
$Ri(t)$ - is the resistive voltage drop,

$L(\theta,i)\frac{di}{dt}$ - is the static voltage,

$i\varpi\frac{dL(\theta,i)}{d\theta}$ - is the motional voltage.

Equation (2) is valid only when the mutual inductances are neglected. From the above equation, it is understood that the motional voltage of each phase is proportional to the speed and rate of change of inductance with respect to rotor position [1, 2].

The instantaneous torque ($T$) produced in the SRM is given by the formula [1, 2]:

$$T_d=\frac{1}{2}i^2\frac{dL}{d\theta} \qquad (3)$$

From the above equation, the positive torque is produced when the phase is switched on during the rising inductance, i.e. $\frac{dL}{d\theta}>0$, negative torque is produced when the phase is switched on during the falling inductance, i.e. $\frac{dL}{d\theta}<0$ and zero torque is produced when the phase is switched on during the constant inductance, i.e. $\frac{dL}{d\theta}=0$. Hence it is required to energize the stator phase windings at proper rotor

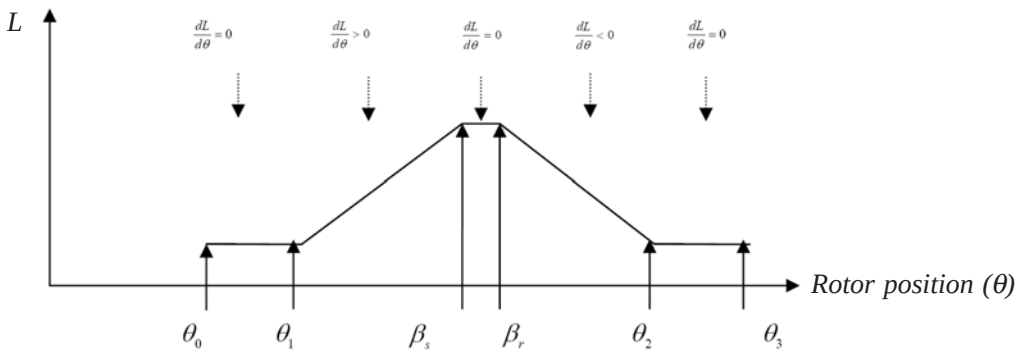



*Fig. 2 Inductance (L) variation over one rotor pole pitch with constant current magnitude*

position. The exact choice of the turn-on and turn-off angles, determine the ultimate performance of the speed control of the SRM. In order to implement the control algorithm effectively, a high performance digital speed controller has to be implemented. The complete basic block diagram of feedback speed control system and SRM drive is shown in Figure 4.

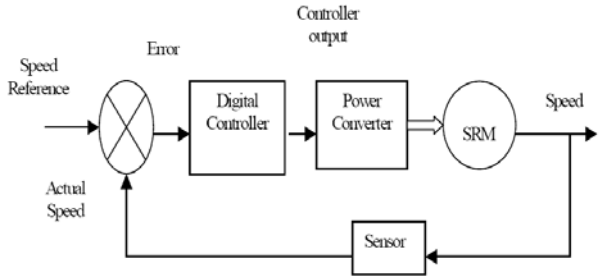



*Fig. 4 Basic speed feed back control system*

## 3. DIGITAL CONTROLLER IMPLEMENTATION

In general, digital controllers can be implemented as digital filters in the following form [17], where $k$ is the current sample in time, for a given sample period $T$:

$$y(k)=\sum_{i=0}^{n} a_i x(k-i)-\sum_{i=0}^{n} b_i y(k-i) \qquad (4)$$

In this form $y(k)$ is the output, $x(k)$ is the input, $a_i$ and $b_i$ are coefficients, or gains, of the controller.

These gains must be selected to produce the desired controller response for a given SRM to be controlled. The structure of this digital controller is illustrated in Figure 5. In the Figure 5, the $Z^{-1}$ blocks represent delays of one sample period.

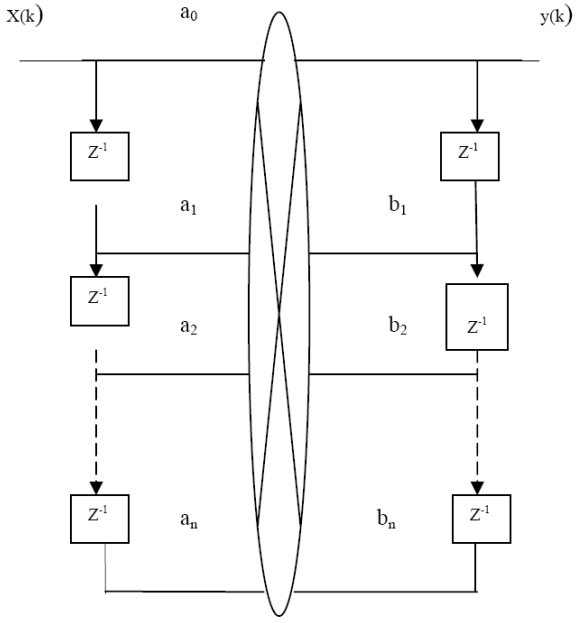



*Fig. 5 Diagram of the digital controller*

When $n=2$, a second order filter is obtained which can be used to implement second order controllers. This representation for a $2^{nd}$ order discrete time controller in the sampled time domain is shown below:

$$Y(k)=a_0x(k)+a_1x(k-1)+a_2x(k-2)-b_1y(k-1)-b_2x(k-2) \qquad (5)$$

The $z$-transform of this gives the following transfer function:

$$D(z)=\frac{Y(z)}{X(z)}=\frac{a_0+a_1z^{-1}+a_2z^{-2}}{1+b_1z^{-1}+b_2z^{-2}} \qquad (6)$$

These kinds of $2^{nd}$ order controller are generally implemented through high-speed digital controllers. In this paper, speed control of SRM utilizes the *P*, *PI* and *PID* algorithm [18-21]. The proportional control mode gives rapid closed-loop transient response to step changes in the speed reference and fast rejection of speed disturbances. The integral control mode assures that the final SRM speed will match the speed reference. Finally, the derivative control mode reduces transient overshoot and oscillation in the speed when the reference is changed from one value to another. The *P*, *PI* and *PID* algorithm can be expressed in the discrete-time domain as:

(i) Discrete Proportional (*P*) Controller

$$u(k)=K_p e(k) \qquad (7)$$

(ii) Discrete Proportional and Integral Controller (*PI*)

$$u(k)=K_p e(k)+\frac{T}{k_1}S(k) \qquad (8)$$

(iii) Proportional, Integral and Derivative Controller (*PID*)

$$u(k)=K_p e(k)+\frac{T}{k_1}S(k)+K_D\frac{e(k)-e(k-1)}{T} \qquad (9)$$

To eliminate the need to calculate the full summation in each time step the summation is expressed as a running sum:

$$S(k)=S(k-1)+\frac{T}{2}\left[e(k)-e(k-1)\right] \qquad (10)$$

where:

$u(k)$ is the control signal,
$e(k)$ is the error signal,
$T$ is the sample period,
$K_p$ is the proportional mode control gain,
$K_i$ is the integral mode control gain,
$K_d$ is the derivative mode control gain.

The $z$-transform of discrete *PID* controller gives the transfer function:

$$D(z)=K_p+\frac{K_1T}{2}\left(\frac{z+1}{z-1}\right)+\frac{K_D}{T}\left(\frac{z-1}{z}\right)$$

$$D(z)=\frac{\left(K_p+\frac{K_1T}{2}+\frac{K_D}{T}\right)+\left(-K_p+\frac{K_1T}{2}+\frac{K_D}{T}\right)z^{-1}+\frac{KD}{T}z^{-2}}{\left(1-z^{-1}\right)} \qquad (11)$$

where:

$$a_0 = K_p + \frac{K_I T}{2} + \frac{K_D}{T}, \quad a_1 = -K_p + \frac{K_I T}{2} - \frac{2K_D}{T},$$

$$a_2 = \frac{K_p}{T}, \; b_1 = -1 \;\; and \;\; b_2 = 0$$

while $a_0$, $a_1$, $a_2$, $b_1$ and $b_2$ are the gains of the digital controller.

The algorithm developed in the discrete time domain has been implemented through a high speed XC2S300E FPGA.

## 4. EXPERIMENTAL SETUP

The SRM drive system under control consists of the components illustrated in Figure 6. The drive system is made up of several distinct subsystems: the 6/4 pole SRM, a personal computer (PC), the driving circuit, classic bridge converter sensing circuitry and the XC2S300E FPGA. The motor is 1.2 hp, 3000 rpm prototype SRM. The SRM is equipped with a rotor position sensor. The three sensors are mounted on the shaft of the SRM, wherein a combination of infrared LED and photo transistor are used to sense the rotor position to indicate which of the three phases of the SRM is to be excited as the motor runs. One of the rotor position sensors out is also used to calculate the actual speed of the SRM. The controller is implemented by verilog coding and executed by an XC2S300E FPGA.

The control algorithm is written and loaded into the XC2S300E FPGA using the PC. The driver circuit is constructed using totem pole configuration, wherein NPN (3904) and PNP (2907) transistors are used. In order to maintain the current at desired value and adjust the controller coefficients, the actual phase current is measured through current sensor (LEM25-NP). The inputs to the FPGA are the rotor position information and DC-bus current. The output of the switching logic section is a sequence of gating signals that are pulse width modulated (PWM) gating signals that are used to drive the classic bridge converter. The power converter is a classic bridge converter utilizing six MOSFETs (IRFP360) and fast recovery diodes (MUR3060). The output of the power converter is chopped to get the desired voltage. The XC2S300E FPGA board is designed with 10 bit ADC to implement different control algorithms.

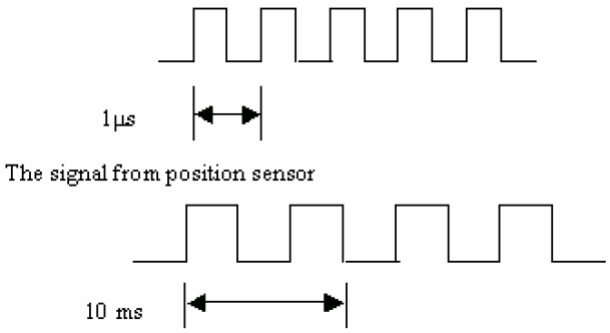
## 5. FPGA BASED CONTROLLER IMPLEMENTATION

***Design criteria:***

Operating frequency: *1 MHz=1 μs*

Frequency of the signal produced from sensor: *100 Hz*

The counter increments itself with every positive edge of the clock. The master clock frequency of the FPGA kit.




Number of counts produced when sensor signal is high (i.e. for *5 ms*): *N*

Time taken for *N* counts : *N μs*

Time for one rotation of the disc : *8N μs*

In *60* seconds number of rotations : *60/8N μs*

Speed in RPM : *7500000 / N*

The reference speed is defined in the software and the error calculation is done. The process repeats every *8* cycles of the sensor output. As the speed of the motor varies the on time the sensor output also changes thereby varying the value of *N*. The difference between actual and reference speeds shows the error.

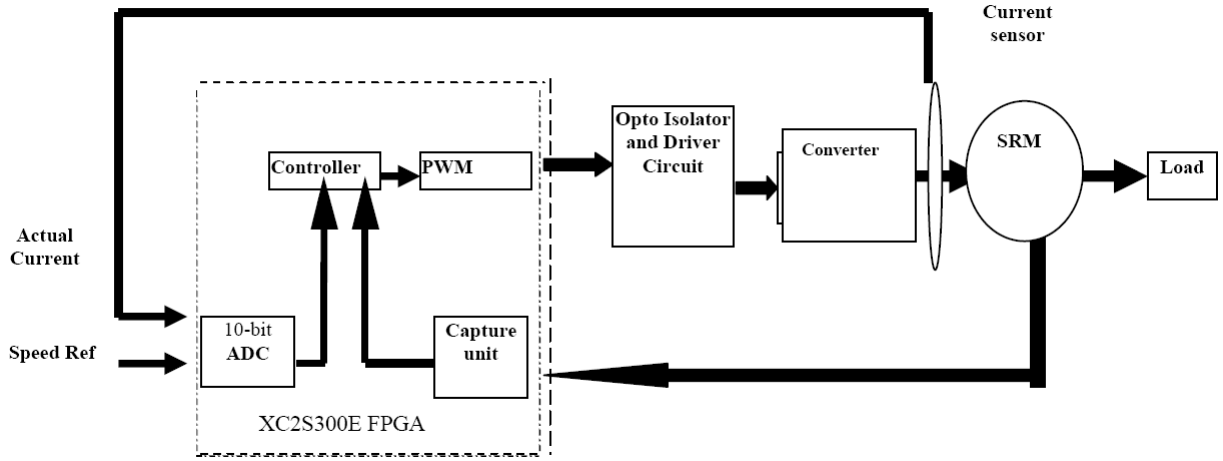



*Fig. 6 Experimental system block diagram of a speed controller for the SRM*

The digital controller consists of a sequence of blocks whose functionalities are realized by Verilog programming. The different parts of the controller are listed below:

- Rotor position signal receiver,
- Registers,
- Positive edge detector,
- Counter for speed measurement,
- Counter for PWM generation,
- Speed comparator,
- Commutation logic,
- PWM output logic generator,
- PID controller.

### *a) Rotor position signal receiver*

It receives the position information of the motor and given to the FPGA the values are stored in the registers:

```
module (s1,out);
input s1;
output out;
reg out;
out =s1;
endmodule;
```

### *b) Registers*

Registers are used to store the data. Register acts as a buffer. It is a device for temporary storage of data:

```
module E_ff(q, data, enable, reset, clock);
output q;
input data, enable, reset, clock; reg q;
always @(posedge clock) // whenever the clock
makes a transition to 1
    if (reset == 0)
    q = 1'b0;
    else if (enable==1)
    q = data;
    // implicitly : else q = q:
Endmodule
```

### *c) Positive Edge Detector*

Detects the positive edge of the sensor signal:

```
input C, D;
output Q;
reg Q;
always @(posedge C)
begin
Q = D;
End
```

### *d) Counter for speed measurement*

In this system the counter is used to measure the speed of the motor. For example, if we measure the speed after some delay means use, the count value is equal to that period. always@ (posedge clk):

```
Begin
if (a=0)
count=0;
else count =count+1;
end
always@(a)
if (a=0)
temp =count;
speed =t/count;
endmodule
```

### *e) Counter for PWM generation*

The pwm is generated according to the duty ratio. The error is calculated from the required speed and the actual speed. According to that error we calculate the duty ratio and pwm is generated. For example pem is generated for the 75% duty cycle, means the counter is generate like this:

```
always@(posedge)
if (count <750)
pwm=1'b1;
else pwm=1'b0;
```

This pwm contains the ¾of the on time and ¼ of the off time. According to that we generate the pwm signals.

### *f) Speed comparator*

It compares the required speed and the actual speed. If the actual speed is less than the required speed means, using the control algorithm the speed is increased. If the actual speed is greater than the required speed means, using control algorithm the actual speed is reduced. The error is maintained as zero:

```
always@(wac)
err=wref- wac;
if (wref >wac)
pwmout=1'b1;
else
pwmout=1'b0;
```

### *g) Commutation logic*

Three positon sensor signals are directly taken in to the FPGA kit. The commutation logic for normal mode is generated with these following statements:

```
Phase 1 = ~s2 and s1;
Phase2 = s3 and ~s1;
Phase 3 = s2 and ~s3;
```

### *h) PWM generation logic*

The value of error is passed in to the discretised equation for PID control action. This generates an actuating signal uk. A fixed PWM at 2 Khz is generated and the value of uk will determine the ON time:

```
er = wref-wn;
err=er/40;
uk=uk_1+(kp+ki+kd)*err-(kp+2*kd)*err_1+kd*err_2;
uk_1=uk;
```

err_2=err_1;
err_1=err;

### *i) PID Controller*

A Proportional + Integral + Derivative controller is implemented in FPGA. $P = Kp$ , $I = Ki$, $D = Kd$.

The Process is:

1) Set value for *Kp*, *Ki*, *kd*
2) Calculate *M1*, *M2*, *M3*
3) Error = Reference speed – actual speed
4) $P = Kp * e(k)$;
   $I = Ki * [I(k-1) + e(k) + e(k-1)]$;
   $D = Kd * [e(k) – e(k-1)]$;
   $U(k) = P + I + D$ ;
5) $Uk-1 = Uk$
   $ek-1=ek$
   $ek-2=ek-1$
6) Out the value of $U(k)$

where:

$u$(k) is the control signal
$e$(k) is the error signal
$e$(k-1) is the previous index error signal
$I$(k-1) is the previous index integral term
$k$ is the discrete-time domain index variable
*Kp* is the proportional mode control gain
*Ki* is the integral mode control gain
*Kd* is the derivative mode control gain

## 6. RESULTS AND DISCUSSION

The speed control systems were built and tested to evaluate the performance of discrete P, PI and PID control algorithm using FPGA. A prototype MOSFET based classic bride converter was fabricated to energize the 6/4 pole motor. Digital controller is implemented through FPGA. Figures 7 and 8 shows the PWM and Commutation logic signals for 6/4 pole SRM by using PID controller. Figure 9 Shows the gate level for PID controller.

*Table 1. Speed RMSE in RPM at different speeds*

| *Speed in RPM* | *Speed RMSE in RPM* | | |
|---|---|---|---|
| | *P control* | *PI Control* | *PID control* |
| *320* | *32.6* | *6.1* | *16.3* |
| *580* | *48.8* | *7.8* | *15.9* |
| *960* | *106.3* | *8.2* | *13.9* |
| *1200* | *49.1* | *9.3* | *17.3* |
| *1740* | *63.6* | *9.8* | *19.6* |

Figures 10-12 shows the experimental results of actual speed, torque and measured phase voltage, phase current at 960 rpm in P, PI and PID control modes during steady state/transient operation under soft chopping operation. In all the cases the dwell angle is kept constant at 30 degrees. In the P control mode, the results have shown that, a steady state error is always present on no load and loaded conditions. In the PI control mode, the results have shown that, the steady state error is very less after 1.09 secs on load conditions. In the PID control mode, the results have shown that, the steady state error becomes less after 0.2 secs in loaded conditions Also, from the Table 1, it is observed that the maximum speed RMSE is 106.3 RPM in P control mode, 9.8 RPM in PI control mode and 19.6 RPM in PID control mode. From the above discussion, it is understood that PI controller is more suitable in applications where very high speed accuracy is required and PID is suitable in applications where quick settling time is required. It is also found from the experimental results, that in PID control mode large chattering effect was present.

## 7. CONCLUSION

In this paper, a robust speed controller for a switched reluctance motor drive using a digital controller has been developed and implemented based on a three-phase classic bridge converter. The controller employs discrete P, PI and PID speed controllers. Experimental results in this paper show that DC-bus voltage can be used to successfully control a three-phase 6/4 pole switched reluctance motor drive system to obtain robust speed holding capability, even under unknown highly dynamic loads. Under load conditions the maximum RMSE speed error was only 9.8 rpm in PI control mode and 19.6 rpm in PID control mode. It is shown that almost perfect robust speed holding could be achieved if the input DC voltage is increased up to 160V (the rated voltage for the motor under test). Most significantly, the implementation of this high performance speed controller needs only high speed FPGA, a few logic IC's, and three current sensors. It can be concluded that discrete PI and PID speed controllers are effective in dealing with the highly nonlinear characteristics of the switched reluctance drive system.

***MOTOR PARAMETERS***

Power : 1.2 KW
Voltage : 160 V
Current : 16 A
Stator outer diameter : 162 mm
Stator core length : 90 mm
Stator inner diameter : 80 mm
Shaft diameter : 25 mm
Number of poles in the stator : 6
Number of turns/pole : 75
Cross-section of the conductor : 1.7 sq-mm
Stator pole arc : 29 degree
Stator pole height : 20 mm
Number of poles in the rotor : 4
Rotor pole arc : 32 degree
Rotor Pole height : 15 mm

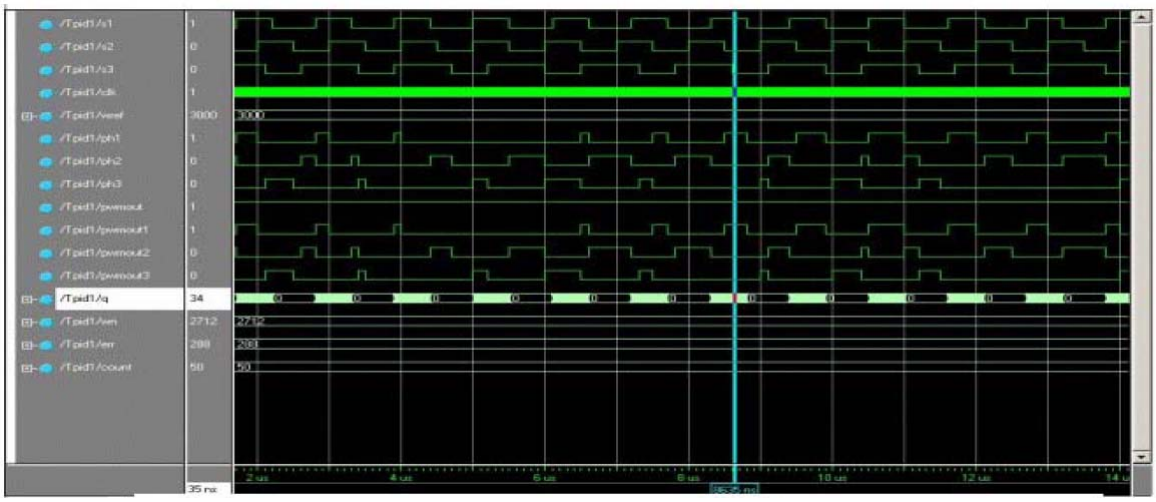


*Fig. 7 PWM and commutation logic signals for 6/4 pole SRM by using PID controller*

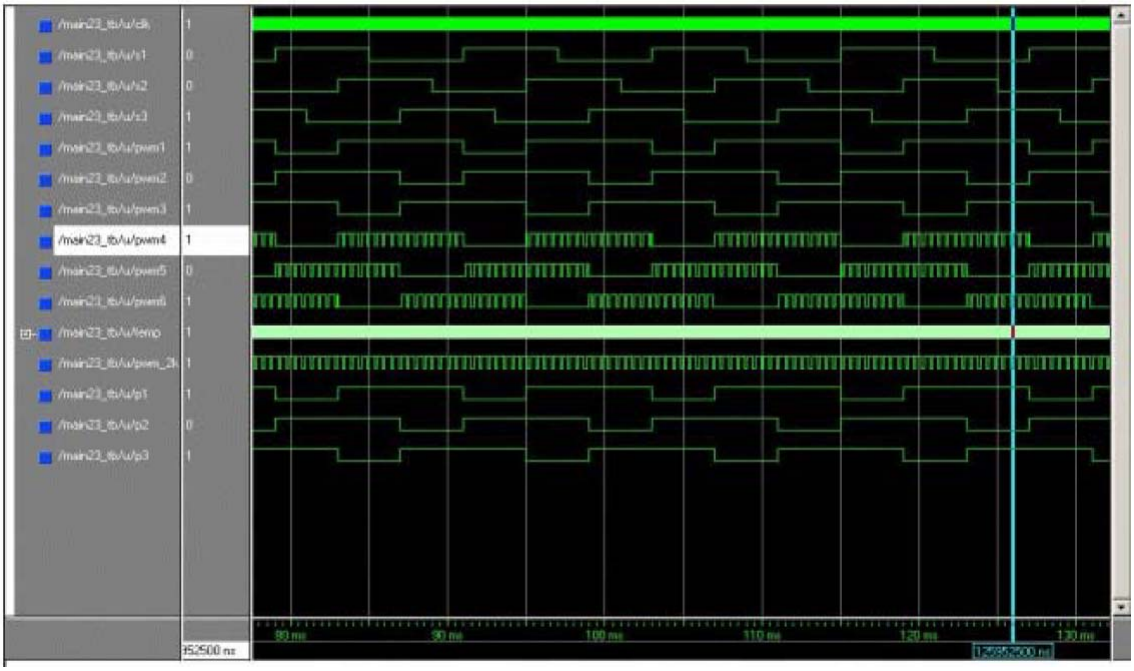


*Fig. 8 PWM and commutation logic signals for 6/4 pole SRM by using PID controllerduring soft chooping*

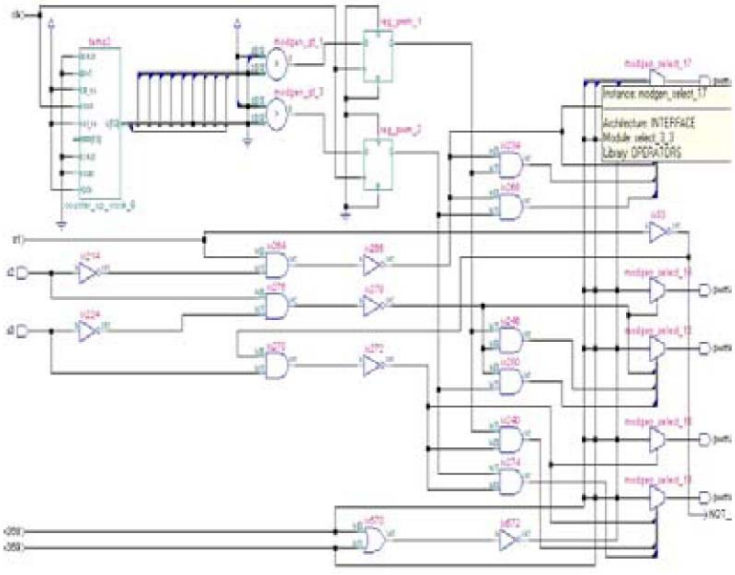


*Fig. 9 Gate level for PID control*

a) b)

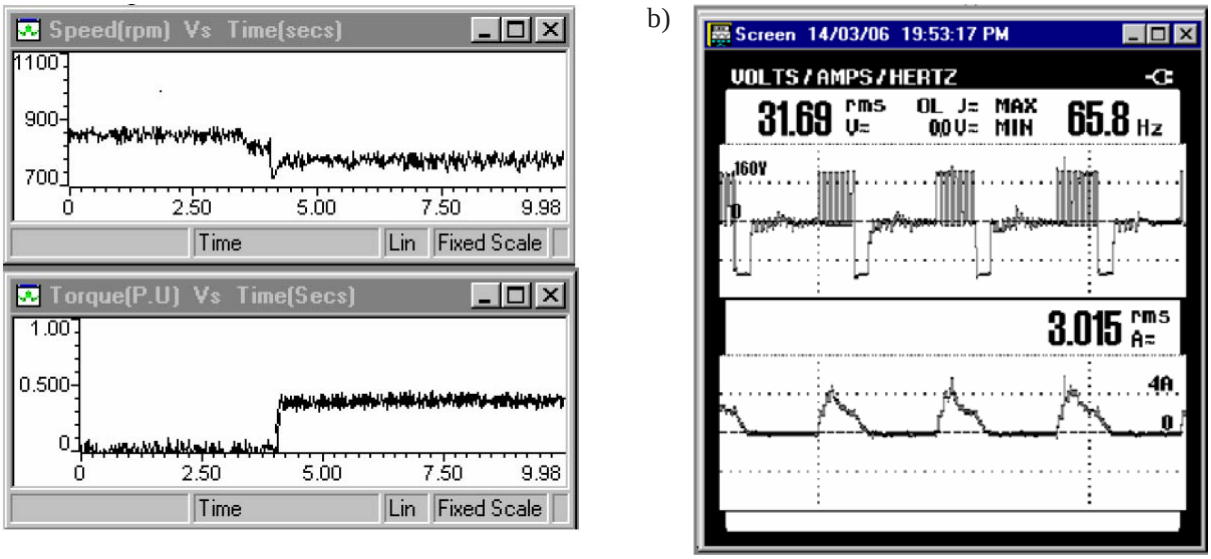



*Fig. 10 Experimental results during P control:*
*(a) Speed (top), Torque (bottom), 0.4 p.u. load, 960 rpm; (b) Phase voltage (upper trace), Phase current (lower trace)*

a) b)

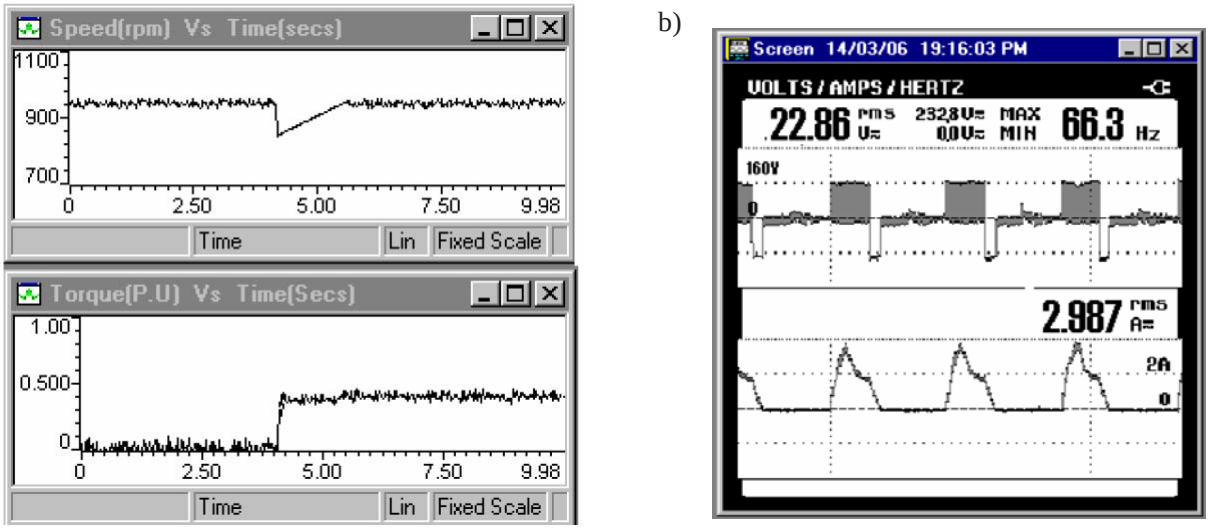



*Fig. 11 Experimental results during PI control:*
*(a) Speed (top), Torque (bottom), 0.4 p.u. load, 960 rpm; (b) Phase voltage (upper trace), Phase current (lower trace)*

a) b)

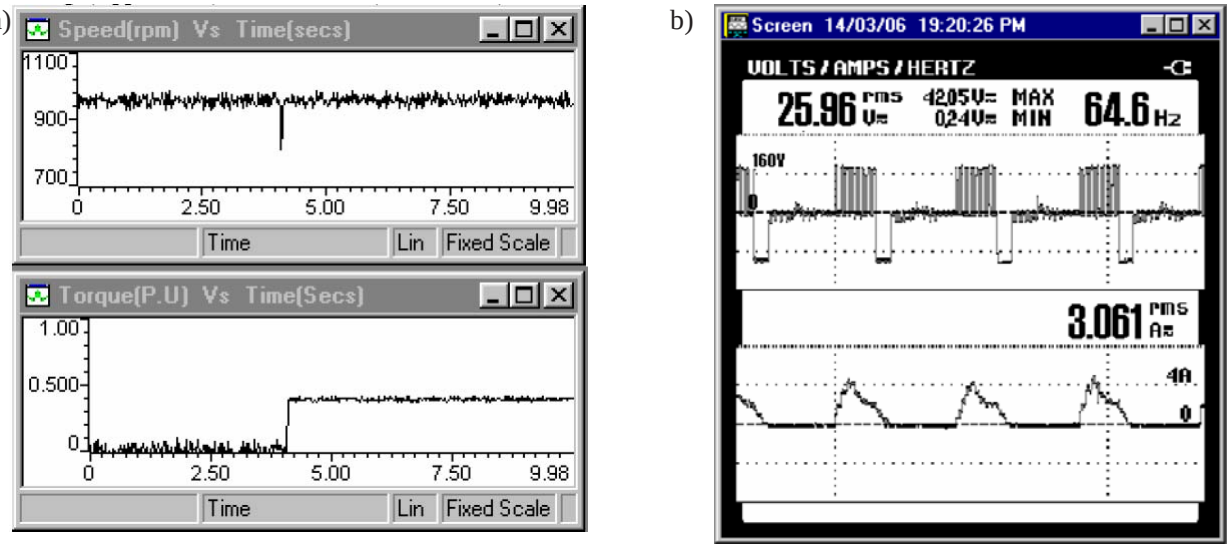



*Fig. 12 Experimental results during PID control:*
*(a) Speed (top), torque (bottom), 0.4 p.u. load, 960 rpm; (b) Phase voltage (upper trace), Phase current (lower trace)*

## 8. REFERENCES


[1] T.J.E. Miller, *Brushless Permanent Magnet and Reluctance Motor Drives*, Oxford University Press, 1989.

[2] R. Krishnan, *Switched Reluctance Motor Drives: Modelling, Simulation, Analysis, Design and Applications*, CRC Press, 2001.

[3] J.V. Byrne and M.F. McMullion, Design of a reluctance motor as a 10 kW spindle drive, Proc. Motorcon'82, Geneva, pp. 10-24, 1982.

[4] B. Powell, *A Low Cost Efficient Motor Driver*, Motornetics Corp., Santa Rosa, 1988.

[5] R. Welburn, *Ultra High Torque Motor System for Direct Drive Robotics*, Motornetics Corp., Santa Rosa, 1989.

[6] W.F. Ray, P.J. Lawrension, R.M. Davis, J.M. Stephension, N.N. Fulton and R.J. Bake, High performance SR brushless drives, *IEEE Transactions on Industrial Applications*, Vol. IA-22, No. 4, pp. 722-730, 1986.

[7] C.A. Ferreira, S.R. Jones, W.S. Heglund and W.D. Jones, Detailed design of a 30-KW switched reluctance starter/generator system for a gas turbine engine application, *IEEE Transaction on Industry Applications*, Vol. 31, No. 3, pp. 553-561, 1995.

[8] C. Hao and X. Guilen, 80C31 single chip computer control of SRM for locomotive in coal mines, China University of Mining and Technology, Xuzhoba, 1984.

[9] C. Hao and X. Guilen, 80C51 single chip control of SRM for locomotive in coal mines, China University of Mining and Technology

[10] R. Krishnan and A.S. Bharadwaj, A comparative study of various motor drive systems for aircraft applications, Motion Control Systems Research Group, Bradley Department of Electrical Engineering, Blacksburg, 1992.

[11] J.C. Moreira and T.A. Lipo, Simulation of a four phase switched reluctance motor including the effects of mutual coupling, *Journal of Electric Machines and Power Systems*, Vol. 16, pp. 281-299, 1989.

[12] A.A. Arkadan and B.W. Kielgas, Switched reluctance motor drive systems dynamic performance prediction and experimental verification, *IEEE Trans. on Energy Conversion*, Vol. 9, No. 1, pp. 36-44, 1994.

[13] S.K. Panda and P.K. Dash, Application of non-linear control to switched reluctance motors: a feed back approach, *Proc. IEE EPA*, Vol. 143, No. 5, pp. 371-379, 1996.

[14] G.S. Buju, R. Menis and M.J. Valla, Variable structure control of an SRM drive, *IEEE Trans. on Industrial Electronics*, Vol. 1, pp. 56-63, 1993.

[15] S. Bologani and M. Zigilotto, Fuzzy logic control for a switched reluctance motor drive, *IEEE Trans. on Industry Applications*, Vol. 32, pp 1063-1068, 1996.

[16] S.A. Hossain, I. Husain, H. Klode, B. Lequesne, A.M. Omekanda and S. Gopalakrishnan, Four-quadrant and zero-speed sensorless control of a switched reluctance motor, *IEEE Transactions on Industry Applications*, Vol. 39, No. 5, pp. 1343-1349, 2003.

[17] Ch.L. Phillips and H.T. Nagle, *Digital Control System Analysis and Design*, Prentice-Hall, Englewood Cliffs, 1984.

[18] F. Blaabjerg, P.C. Kjaer, P.O. Rasmussen and C. Cossar, Improved digital current control methods in switched reluctance motor drives, *IEEE Transactions on Power Electronics*, Vol. 14, No. 3, pp. 563-572, 1999.

[19] H.K. Bae and R. Krishnan, A study of current controllers and development of a novel current controller for high performance SRM drives, Proc. Industry Applications Conference, Vol. 1, pp. 68-75, 1996.

[20] K. Astrom and T. Hagglund, *PID Controllers: Theory, Design and Tuning*, 2nd ed., Instrument Society of America, Research Triangle Park, NC, 1995.

[21] S. Mir, Classification of SRM converter topologies for automotive applications, Proc. SAE World Congress 2000, Detroit, 2000.

## PRIMJENA DIGITALNOG REGULATORA BRZINE KOD RELUKTANTNOG PREKIDAČKOG MOTORNOG POGONA PRIMJENOM FPGA

**SAŽETAK**

*U ovom radu opisana je digitalna shema kontrole brzine koja se zasniva na Field-Programmable Gate Array-u (FPGA) i kojom se žele otkloniti postojeći nedostaci prethodnih shema kontrole brzine, koje su se predlagale za reluktantne prekidačke motorne pogone (SRM). Zasniva se na diskretnom P, PI i PID algoritmu kontrole i zahtijeva jednostavne matematičke modele. Shema se primjenjuje koristeći XC2S300E FPGA. Eksperimentalni rezultati realnog vremena, koji su izneseni u ovom radu, pokazuju da predložena metoda kontrole brzine omogućava preciznu kontrolu brzine kao i široki raspon brzina i to u različitim radnim uvjetima (stacionarno stanje / prijelazni rad u lagano primjenjivom načinu rada). Kontrolni sustav brzine SRM zatvorene petlje postiže 6.2 RPM točnosti brzine, koja ovisi o potrebnom radnom rasponu brzine s obračunatim vremenom od 0.25 do 1.05 sekunda reakcije u fazama. U radu je opisan cjelokupni eksperimentalni sustav uključujući primjenu FPGA.*

***Kjučne riječi**: reluktantski prekidački motor, kontrola brzine, diskretni P, PI i PID kontroleri, FPGA.*