# DIGITAL SIMULATION OF INDIRECT FIELD ORIENTED CONTROL OF THREE PHASE INDUCTION MOTOR WITH RECURRENT ARTIFICIAL NEURAL NETWORK SPEED ESTIMATION

**Abdulhamid H. Esuri and Alaa Rabah Abonda**

Department of Electric & Electronic engineering
Al-Fateh University, Tripoli, Libya
Essuri_a_h@yahoo.com

الملخص

تعـرض هـذه الورقـة أسـلوب محاكـاة منظومـة الـتحكم في سـرعة المحركـات الحثيـة بـدون استخدام مجسات السـرعة وتعتمـد هـذه الطريقـة على التنبؤ بمقدار السـرعة باستخدام الشبكات العـصبية العقديـة ذات التغذيـة الخلفيـة. حيـث تم اسـتخدام Matlab Simulink لدراسـة وتمثيـل خصائص منظومـة الـتحكم تحـت ظـروف تشغيلية مختلفـة مثل الـتغير السـريع في السـرعة وعـزم الدوران.

## ABSTRACT

This paper simulates the speed sensorless based indirect field oriented control of induct-ion motor, with recurrent neural network speed estimation. This simulation is preformed using matlab simulink to achieve the performance of the drive under different operating conditions such as speed, and torque step changes.

**KEYWORLDS:** Induction Motor; Speed sensorless; Indirect field-oriented; Recurrent neural network; Matlab simulink

## INTRODUCTION

In recent years, a large number of speed sensorless vector control systems for induction motor (IM) have been proposed. Speed information is generally provided by a speed transducer on the motor shaft; recently, low cost and high performance digital signal processors (DSP) become available allowing obtaining speed by means of digital estimators integrated with motor control. This solution represents an advantage in terms of cost, simplicity and mechanical reliability of the drive. Several schemes of speed estimators have been proposed in the literature; among them, the model reference adaptive system (MRAS) approach is very attractive and gives good performance [1-3]. The classical MRAS method is based on the adaptation of the rotor flux [1]; with this scheme, some difficulties in terms of precise and robust speed estimation arise, especially at low speed. The need of pure integration in the speed estimator represents a drawback in the low speed region, due to drift and low frequency disturbances; moreover, parameter sensitivity (in particular to stator resistance) represents a usual disadvantage for all model-based estimators [1]. To overcome these problems, alternative schemes like the recurrent neural network (RNN) speed estimator has been presented, but it seems that the low speed troubles are not solved exactly. The RNN speed estimator is one of advanced topic in the speed sensorless area [2].

The main contribution of this paper is simulink simulation to RNN speed sensorless vector control that is based on: (a) indirect-field oriented control (IFOC), (b) a RNN speed estimation (online estimation), (c) suitable adjustments to improve robustness with respect to parameter variations.

## INDUCTION MOTOR MODEL

In an adjustable speed drive, the machine normally constitutes an element within a feedback loop, and therefore its transient behavior has to be taken into consideration [1,8]. Besides, high-performance drive control, such as vector or field oriented control is based on the dynamic d-q model of the machine. Therefore, a good understanding of the d-q model is important.

Basically, there are three type of dynamic d-q model of induction motor stationary reference frame $d^s - q^s$, rotor reference frame $d^r - q^r$ , and synchronous reference frame $d^e - q^e$.

### a) Synchronously Rotating Reference Frame (Kron Equation)

For the two-phase machine shown in Figure (1). The synchronously rotating reference frame model $d^e - q^e$ can be represented as [1]:

$$v_{qs} = R_s i_{qs} + \frac{d}{dt} \psi_{qs} + \omega_e \psi_{ds} \qquad (1)$$

$$v_{ds} = R_s i_{ds} + \frac{d}{dt} \psi_{ds} - \omega_e \psi_{qs} \qquad (2)$$

Where $\psi_{qs}^s$ and $\psi_{ds}^s$ are q-axis and d-axis stator flux linkages, respectively, and all the variables are in rotating frame. The last term in equations (1) and (2) can be defined as speed emf ($\omega_e$) due to rotation of the axes, that is, when $\omega_e = 0$, the equations revert to stationary form. Note that the flux linkage in the $d^e$ and $q^e$ axes induce emf in the $q^e$ and $d^e$ axes, respectively, with $\pi/2$ lead angle. If the rotor not moving, that is, $\omega_r = 0$, the rotor equations will be similar to equations (1) and (2):
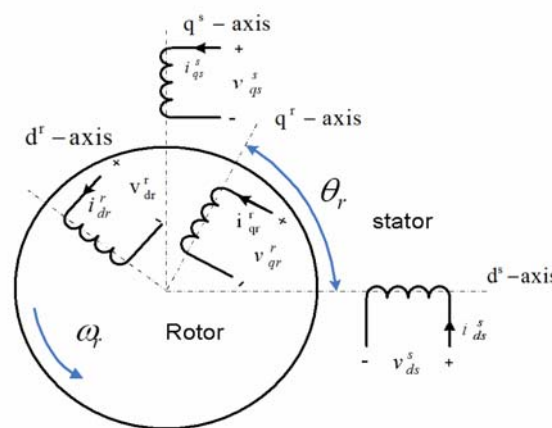


**Figure 1: Equivalent two phase machine**

---

$$v_{qr} = R_r i_{qr} + \frac{d}{dt} \psi_{qr} + \omega_e \psi_{dr} \tag{3}$$

$$v_{dr} = R_r i_{dr} + \frac{d}{dt} \psi_{dr} - \omega_e \psi_{qr} \tag{4}$$

Where the d-q axes on the rotor moves at a speed $\omega_e - \omega_r$ relative to the synchronously rotating frame. Therefore, in $d^e - q^e$ frame, the rotor equations should be written as [1]:

$$v_{qr} = R_r i_{qr} + \frac{d}{dt} \psi_{qr} + (\omega_e - \omega_r) \psi_{dr} \tag{5}$$

$$v_{dr} = R_r i_{dr} + \frac{d}{dt} \psi_{dr} - (\omega_e - \omega_r) \psi_{qr} \tag{6}$$

Where $\psi_{qr}$ and $\psi_{dr}$ are q-axis and d-axis rotor flux linkages, respectively. The speed $\omega_r$ in the above equations cannot normally be treated as a constant. It can be related to the torque as [1]:

$$T_e = T_L + J \frac{d\omega_m}{dt} + \beta \omega_m = T_L + \frac{2}{P} J \frac{d\omega_r}{dt} + \frac{2}{P} \beta \omega_r \tag{7}$$

where $T_L$ = load torque, $J$ = rotor inertia, $P$ = number of pair pole, $\beta$ = friction factor, and $\omega_m$ = mechanical speed.

The more general form of the development torque -by interaction of air gap flux and rotor mmf- relating the d-q components of variables can be expressed in vector form as:

$$T_e = \frac{3}{2}(\frac{P}{2})\overline{\psi}_m \times \overline{I}_r \tag{8}$$

By resolving the variables into $d^e - q^e$ components gives:

$$T_e = \frac{3}{2}(\frac{P}{2})(\psi_{dm} i_{qr} - \psi_{qm} i_{dr}) \tag{9}$$

Equations (1), (2 ),(5),(6) (7) and (9) give the complete model of the electro-mechanical dynamics of induction machine in synchronous frame. The composite system is of the fifth order and nonlinearity of the model is evident.

**b) Stationary Frame- Dynamic Model (Stanley Equation)**

The dynamic machine model in stationary frame $d^s - q^s$ can be derived simply by substituting $\omega_e = 0$ in equations (1), (2), (5) and (6). The corresponding stationary frame equations are given as [1]:

$$v_{qs}^s = R_s i_{qs}^s + \frac{d}{dt} \psi_{qs}^s \tag{10}$$

$$v_{ds}^s = R_s i_{ds}^s + \frac{d}{dt} \psi_{ds}^s \tag{11}$$

$$0 = R_r i_{qr}^s + \frac{d}{dt} \psi_{qr}^s - \omega_r \psi_{dr}^s \tag{12}$$

$$0 = R_r i_{dr}^s + \frac{d}{dt} \psi_{dr}^s + \omega_r \psi_{qr}^s \tag{13}$$

In the stationary frame, the variables appear as sine waves in steady state with sinusoidal inputs.

## INDIRECT ROTOR FIELD ORIENTED CONTROL (IRFOC)

In the indirect rotor field orientation Control The unit vector signals ($\cos\theta_e$ and $\sin\theta_e$) are generated in feed forward manner [1,8]. IRFOC is very popular in industrial applications. Figure (2) explains the fundamental principle of IRFOC with the help of a phasor diagram. The $d^s - q^s$ axes are fixed on the stator, but the $d^r - q^r$ axes, which are fixed on the rotor, are moving at speed $\omega_r$ as shown. Synchronously rotating axes $d^e - q^e$ are rotating ahead of the $d^r - q^r$ axes by the positive slip angle $\theta_{sl}$ corresponding to slip frequency $\omega_{sl}$. Since the rotor pole is directed on the $d^e$ axis and $\omega_e = \omega_r + \omega_{sl}$, then $\theta_e$ can be calculated as [7]:

$$\theta_e = \int \omega_e dt = \int (\omega_r + \omega_{sl}) dt = \theta_r + \theta_{sl} \tag{14}$$

Note that the rotor pole position is not absolute, but is slipping with respect to the rotor at frequency $\omega_{sl}$. The phasor diagram suggests that for decoupling control, the stator flux component of current $i_{qs}$ should be on the $q^e$ axis, as shown.

For complete the decoupling method must be make a derivation of control equations of IFOC with the help of $d^e - q^e$ model which discussed before [1]. Then from equations (5) and (6), the rotor circuit equations can be written as [1, 2]:

$$\frac{d\psi_{dr}}{dt} + \frac{R_r}{L_r}\psi_{dr} - \frac{L_m}{L_r}R_r i_{ds} - \omega_{sl}\psi_{qr} = 0 \tag{15}$$

$$\frac{d\psi_{qr}}{dt} + \frac{R_r}{L_r}\psi_{qr} - \frac{L_m}{L_r}R_r i_{qs} + \omega_{sl}\psi_{dr} = 0 \tag{16}$$

Where $\omega_{sl} = \omega_e - \omega_r$ has been substituted.

For decoupling control, it is desirable that

$$\psi_{qr} = 0 \tag{17}$$



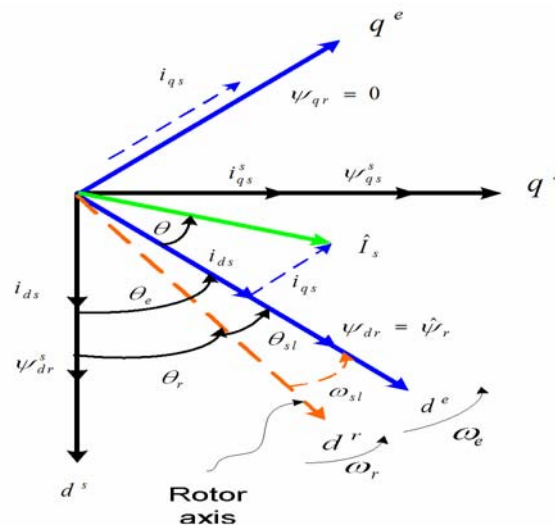**Figure 2: Phasor diagram explaining indirect vector control.**

That gives,    $\dfrac{d\psi_{qr}}{dt} = 0$                                                                                      (18)

In other words, there is no imaginary part and the total rotor flux $\hat{\psi}_r$ is directed on the $d^e$ axis.

By substituting the last two conditions in equations (15) and (16),

$$\dfrac{L_r}{R_r}\dfrac{d\hat{\psi}_r}{dt} + \hat{\psi}_r = L_m i_{ds}$$                                                           (19)

$$\omega_{sl} = \dfrac{L_m R_r}{\hat{\psi}_r L_r} i_{qs}$$                                                                          (20)

where $\hat{\psi}_r = \psi_{dr}$ has been substituted .

   If rotor flux $\hat{\psi}_r$ = constant (i.e. steady state), which is usually the case, then from equation (19),

$\hat{\psi}_r = L_m i_{ds}$                                                                                                        (21)

That means, the rotor flux is directly proportional to current $i_{ds}$ in steady state.

To implement the indirect field orientation control strategy, it is necessary to take equation (14), (19), and (20) into consideration. Figure (3) shows IRFOC block diagram [1]. The power circuit consists of a front-end diode rectifier and a PWM inverter with a dynamic brake in the dc link. A hysteresis-band current control PWM is shown, but the other PWM strategies can also be used. The speed control loop generates the torque component of current $i_{qs}^{*}$, as usual. The flux component of current $i_{ds}^{*}$ for the desired rotor flux $\hat{\psi}_r$ is determined from equation (21), and is maintained constant here in the open manner for simplicity. The variation of magnetizing inductance $L_m$ will cause some drift in the flux. The slip frequency $\omega_{sl}^{*}$ is generated from $i_{qs}^{*}$ in feedforward manner from equation (20) to satisfy the phasor diagram in Figure (2). The corresponding expression of slip gain $K_s$ is given as:
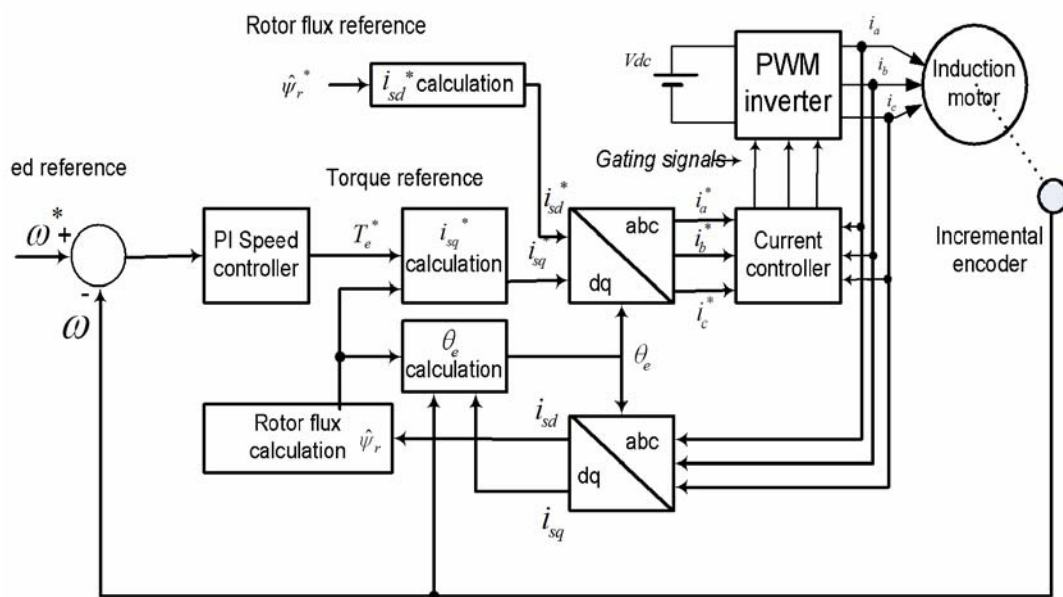


**Figure 3: Indirect field orientation control block diagram with open loop flux control**

$$K_s = \frac{\omega_{sl}^*}{i_{qs}^*} = \frac{L_m R_r}{L_r \hat{\psi}_r} \qquad (22)$$

Signal $\omega_{sl}^*$ is added with speed signal $\omega_r$ to generate frequency signal $\omega_e$. The unit vector signals $\cos\theta_e$ and $\sin\theta_e$ are then generated from $\omega_e$ by integration, as indicated in Figure (3). The speed signal from an incremental-position encoder is mandatory in indirect vector control because the slip signal locates the pole with respect to the rotor $d^r$ axis in feedforward manner, which is moving at speed $\omega_r$. If the polarity of $i_{qs}^*$ becomes negative for negative torque, the phasor $i_{qs}$ in Figure (2) will be reversed, and correspondingly, $\omega_{sl}$ will be negative (i.e. $\theta_{sl}$ is negative), which will shift the rotor pole position ($d^e$ axis) below the $d^r$ axis.

**SPEED ESTIMATION BY RECURRENT NEURAL NETWORK (RNN).**

Speed estimation by the model referencing adaptive system (MRAS) method is shown in Figure (4), which is discussed in the literature (see [1,3]) . The current model flux estimator (that is shown in the lower part of Figure 4) with an adaptive speed signal ($\omega_r$) is a first-order dynamic system, and it can be implemented by the RNN as shown in Figure (5).

In the case of RNN-based speed estimator each output neuron uses the linear activation function. The solution of the voltage model (shown in upper part of Figure (4), and Figure (5) generates the desired flux components. These signals are compared with the RNN output signals and the weights are trained on-line so that the error $\xi(k+1)$ tends to zero. The on-line training can be done by dynamic backpropagation method, and it should be fast enough so that actual speed variations can be tracked well [1,3]. The current model described equation can be discretized and written as [4]:
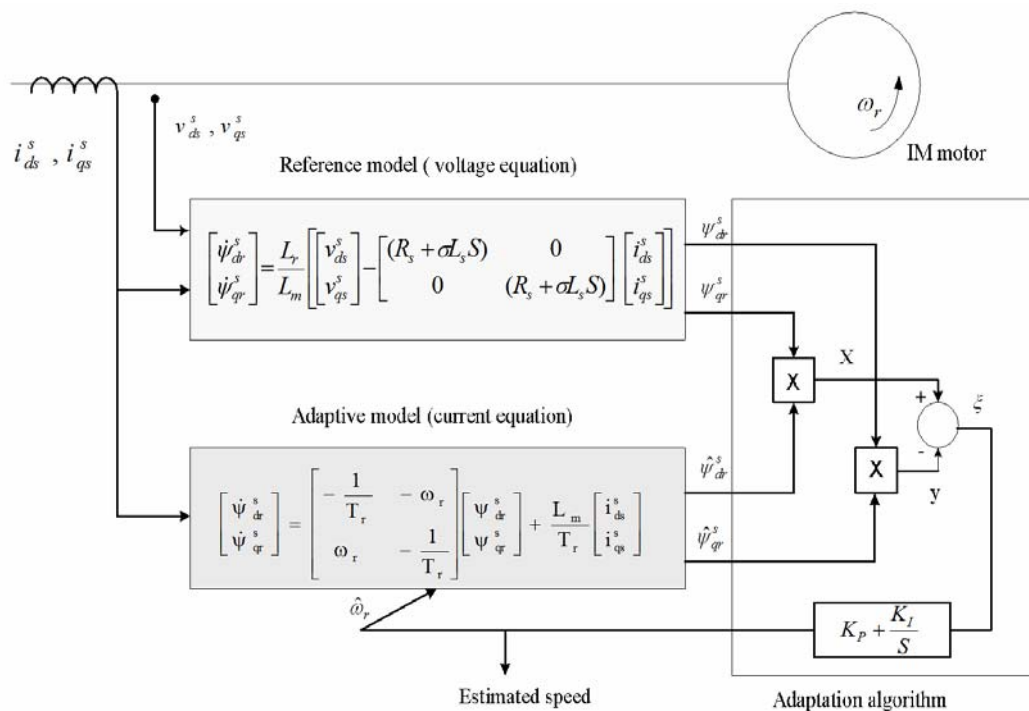


Figure 4: Estimation by the model referencing adaptive system (MRAS) principle.

$$\begin{bmatrix} \psi_{dr}^s(k+1) \\ \psi_{qr}^s(k+1) \end{bmatrix} = \begin{bmatrix} 1-\dfrac{T_s}{T_r} & -\omega_r T_s \\ \omega_r T_s & 1-\dfrac{T_s}{T_r} \end{bmatrix} \begin{bmatrix} \psi_{dr}^s(k) \\ \psi_{qr}^s(k) \end{bmatrix} + \begin{bmatrix} \dfrac{L_m T_s}{T_r} & 0 \\ 0 & \dfrac{L_m T_s}{T_r} \end{bmatrix} \begin{bmatrix} i_{ds}^s(k) \\ i_{qs}^s(k) \end{bmatrix} \qquad (23)$$

Where $T_s$ = sampling time, $L_m$ = magnetizing inductance, and $T_r$ = rotor time constant. The above equation can also be expressed in the form

$$\begin{bmatrix} \psi_{dr}^s(k+1) \\ \psi_{qr}^s(k+1) \end{bmatrix} = \begin{bmatrix} W_{11} & W_{21} \\ W_{12} & W_{22} \end{bmatrix} \begin{bmatrix} \psi_{dr}^s(k) \\ \psi_{qr}^s(k) \end{bmatrix} + \begin{bmatrix} W_{31} & 0 \\ 0 & W_{32} \end{bmatrix} \begin{bmatrix} i_{ds}^s(k) \\ i_{qs}^s(k) \end{bmatrix} \qquad (24)$$

Where $W_{11} = 1 - T_s/T_r$, $W_{21} = -\omega_r T_s$, $W_{12} = \omega_r T_s$, $W_{22} = 1 - T_s/T_r$, $W_{31} = L_m T_s/T_r$, and $W_{32} = L_m T_s/T_r$. The RNN with a bipolar linear transfer function of unity gain satisfies equation (24). Note that out of the six weights in the network, only $W_{21}$ and $W_{12}$ (circled in the figure) contain the speed term; therefore, for speed estimation, it is sufficient if these weights are considered trainable, keeping the other weights constant (assuming $T_r$ and $L_m$ as constants) for speed estimation. Where the speed can be calculated from equation $\omega_r = W_{12}/T_s$. However, if all the weights are considered trainable, the speed as well as the rotor time constant can be tuned [1].
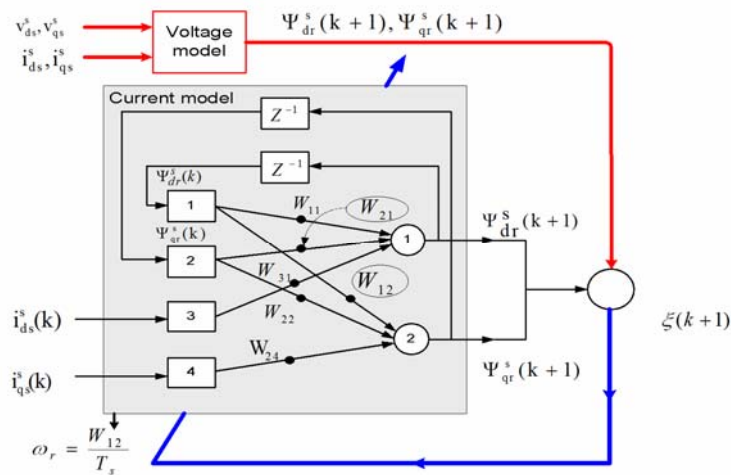


**Figure 5: Recurrent Neural Network (RNN) Speed Estimation.**

**SIMULINK SIMULATION MODEL**

The three phase IM models can be simulated by matlab simulink from SimPowersystems simulink library, where the selected case study data is as below:

- Rotor type (squirrel-cage).
- Reference frame (stationary).
- 4 poles, rated power (50*746 watt).
- L-L voltage 460 V with 60 hz.
- $R_s = 0.087\,\Omega$, $R_r = 0.228\,\Omega$, $L_{ls} = L_{lr} = 0.8 \times 10^{-3}\,H$.
- Mutual inductance $L_m = 34.7 \times 10^{-3}\,H$.
- Moment of inertia $J = 1.662\,kg.m^2$, and friction factor $F = 0.12\,N.m.s$
- Pulse width modulation inverter
  - Hystresis-Band Current Control PWM with IGBT devises

- The voltage rectified value Vdc=780V.
- Snubber resistance $R_s = 1000\Omega$.
- Snubber capacitance $C_s = \infty$ Farad (F)

▪ The synchronously rotating frame angle $\theta_e$ estimation.

$$\theta_e = \int(\frac{34.7\times10^{-3}}{0.1557\psi_{dr}}i_{qs} + \omega_r)dt$$

▪ Rotor Flux estimation.

$$\hat{\psi}^*{}_r = \frac{L_m i_{ds}}{(ST_r + 1)} = \frac{3.4\times10^{-3}i_{ds}}{0.1557S+1}$$

▪ d-axis Stator Current $i_{ds}^*$ estimation.

$$i_{ds}^* = \psi_{dr}^*/L_m = \psi_{dr}^*/(34.7\times10^{-3})$$

▪ q-axis Stator Current $i_{qs}^*$ estimation.

$$i_{qs}^* = \frac{2}{3}\frac{1}{P}\frac{L_r}{L_m\psi_{dr}^*}\cdot T_e^*$$

▪ PI Speed Controller with feed back signals.

Using Ziegler-Nichols tuning criterion and simulink matlab the PI speed controller parameter can be tuned as [4,6]:

$$K_P = 90 \ , \ K_I = 4320.$$

▪ The voltage model is simulated using equations shown upper part in Figure (4).

▪ The RNN speed estimator can be built with the help of equation (24), and the next equation of the training node $W_{12}$ that is training by backpropagation algorithm as [1,3]:

$$\xi = \tfrac{1}{2}(E(k))^2 \tag{25}$$

$$W_{12}(k+1) = W_{12}(k) + \eta\cdot(-\tfrac{\partial\xi}{\partial W_{12}}) + \mu\cdot\Delta W_{12}(k) \tag{26}$$

where $\xi$ is the error between the voltage and current RNN models, $\eta$ is the learning rate, and $\mu$ is the momentum factor. From the last equation and simplified the speed can be estimated as:

$$\hat{\omega}_r(k+1) = \hat{\omega}_r(k) - \frac{\eta}{T_s}[-e_\alpha\hat{\psi}_{qr}^{si}(k) + e_\beta\hat{\psi}_{dr}^{si}(k)] + \frac{\mu}{T_s}[\hat{\omega}_r(k) - \hat{\omega}_r(k-1)] \tag{27}$$

Where $e_\alpha = \hat{\psi}_{dr}^{sv}(k+1) - \hat{\psi}_{dr}^{si}(k+1)$, $e_\beta = \hat{\psi}_{qr}^{sv}(k+1) - \hat{\psi}_{qr}^{si}(k+1)$.

The subscript $i$ refers to RNN model component, and subscript $v$ refers to voltage model component, while the hat subscript refers to the estimated component. Figure (6) shows the simulink model of speed sensorless of IFOC with RNN speed estimation.

## VI. SIMULATION RESLUTS AND DISCUSSION.

The simulation period is 4 (sec) with sampling time $T_s = 2\times10^{-6}$ sec. The machine starts from standstill with 100 (N.m) load torque until the instant t =3(sec) then the load

---

is increased to 150 (N.m). Also a speed step has been applied at t=2(sec) with the value of $[\omega_r = (120 \rightarrow 160)\,(rad/sec)]$.

Figure (7) shows the minimizing speed error caused by the learning rate $\eta$ at constant momentum factor $\mu = 0.1$. The estimated speed error is decreased as $\eta$ is increased until $\eta = 0.4$. The next step is to minimize the error caused by the $\mu$ coefficient as explained in figure (8). Thus the $\mu$ factor has been changed and the learning rate is kept constant at the value of $\eta = 0.4$. It is apparent from Figure (8) that if $\mu$ has a value larger than 0.3 the error is increased rapidly to a large value as in the case of $\mu = 0.9$. From the last discussion the electric and mechanical behaviors of speed sensorless control based IFOC with RNN speed estimator were performed and the best result is obtained at $\eta = 0.4$, and $\mu = 0.1$ as shown in Figure (9) and Figure (10).
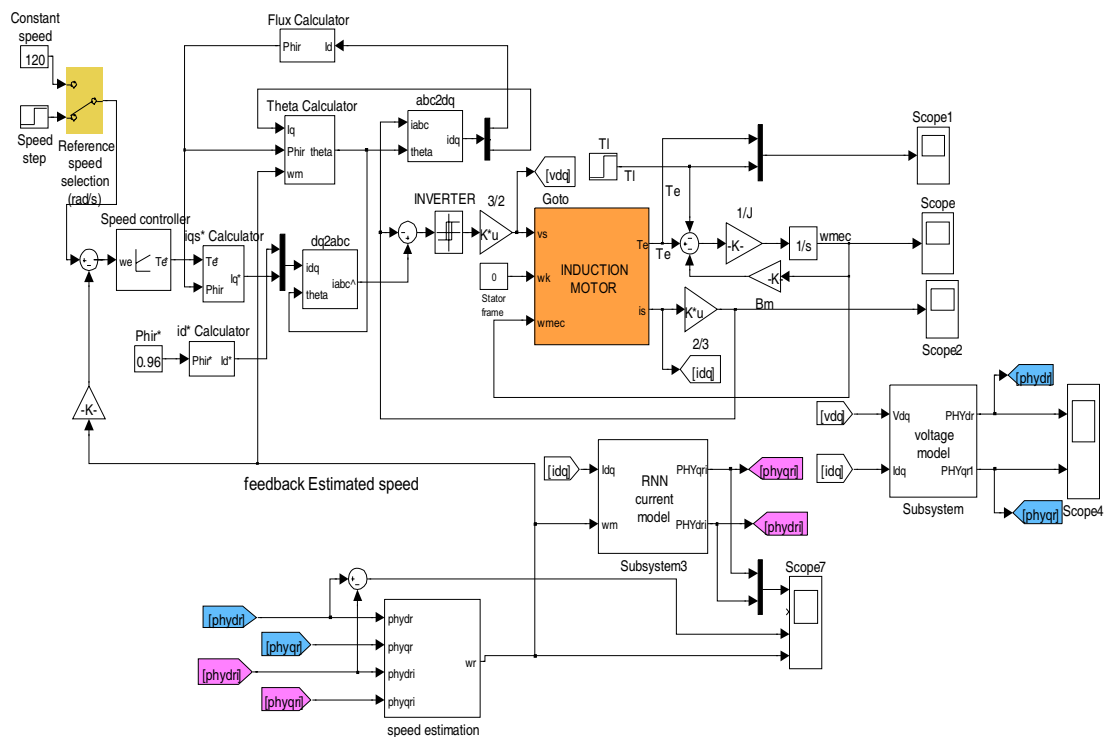


**Figure 6: Simulink model of speed sensorless of IFOC with RNN speed estimation.**

Figure (9) shows the motor phase current, RNN estimated speed response of PI controller, and the torque waveform. Where the motor torque from t=0 sec to t =3 sec is 100 N m. then at t =2 sec when the speed increased from 120 to 150 rad/sec the value of torque increased to it's limit 300N m due to the increase in current phases. The steady state speed of 160rad/sec was attained in a period approach to 1 sec after the disturbance occurred. So it can be observed the pulsating torque which is generated due to hystresis-band current control PWM, and switching harmonics of the converter. The drive response to a torque step change is performed after t =3 sec.

Figure (10) shows the motor speed curve with sensor feedback signal, it can be seen that the overshoot of PI controller which is too soft and fast transit from standstill to final of operating condition. So the pulsating torque effect is negligible due to the inertia filtering of the motor.
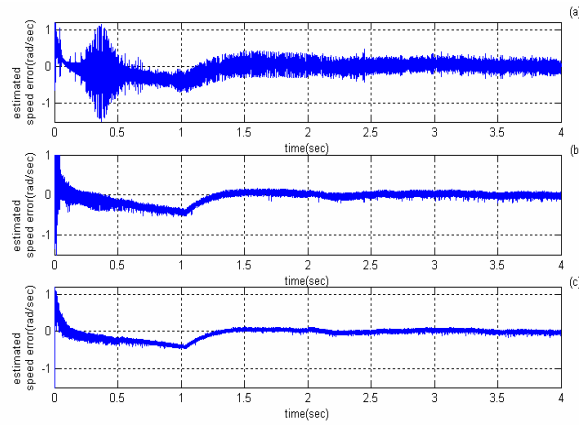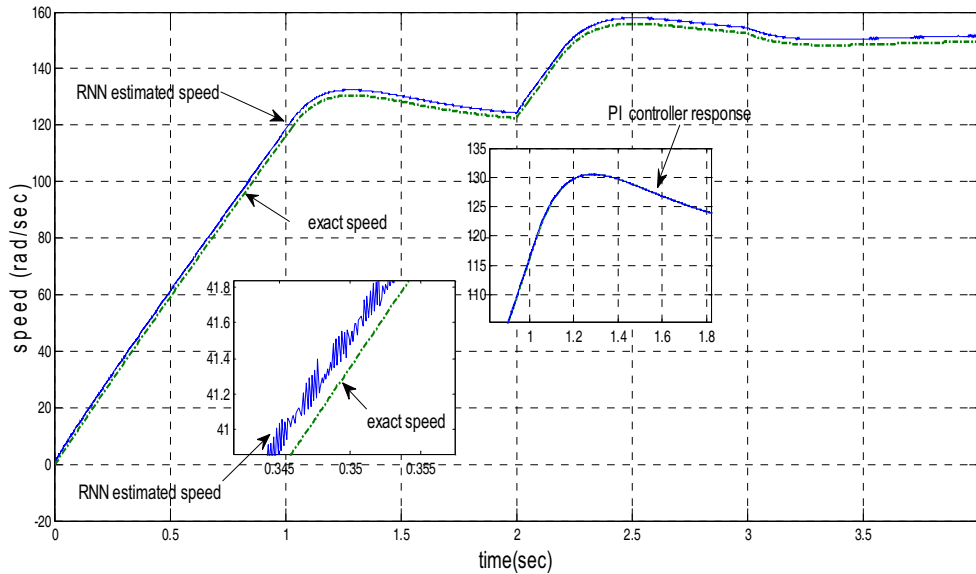
**Figure 7: estimated speed error for constant** $\mu = 0.1$ **and**
**a)** $\eta = 0.004$  **b)** $\eta = 0.04$  **c)** $\eta = 0.4$
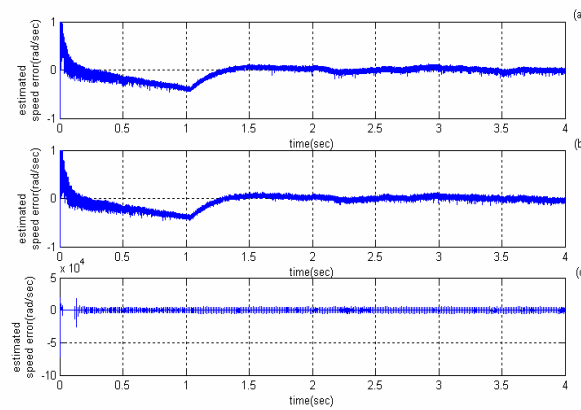


**Figure 8: estimated speed error for constant** $\eta = 0.4$ **and**
**a)** $\mu = 0.01$  **b)** $\mu = 0.3$  **c)** $\mu = 0.9$
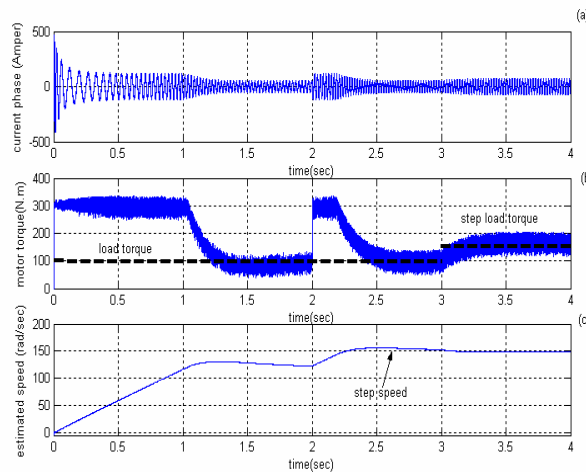


**Fig 9: a) current of phase a, b) motor torque, and c) RNN estimated speed.**

**Figure 10: RNN speed estimation and exact speed with feedback signal**

## CONCLUSION

The objective of obtaining close loop control system using rotor flux oriented theory has been achieved, as demonstrated in this paper. The mathematical model of basic drive with a hysteresis-band PWM inverter was simulated using MATLAB SIMULINK, also all additional identification blocks were then added to this basic RFOC system.

This paper shows that the PI speed controller can be designed using Ziegler-Nichols tuning criterion. This criterion is one of the powerful and easy methods of designing PI controller.

The parameter response of high performance IRFOC of IM has been simulated using SIMULINK under the torque disturbance or speed step changes. Any other response using this software can easily be obtained.

The objective of training an accurate RNN based speed motor estimator using matlab software has been done. The speed estimation using on-line backpropagation RNN training algorithm are simulated and the simulation results are presented, So the RNN training parameters that are generated in this paper are ready to be implemented in real-time applications.

## REFERENCE

[1]  Bimal K.Bos, Modern Power Electronics and AC Drives, Prentice Hall, New Jersey, 2002

[2]  M.N. Cirstea, A. Dinu, J.G. Khor, M. McCormick, Neural and Fuzzy Logic Control of Drives and Power Systems, Newnes, Great Britain,2002.

[3]  Bimal K. Bose, Neural Network Applications in power Electronics and Motor Drives-An Introduction and perspective, Industrial Electronics, IEEE Transactions on, VOL 54, pp 14-33, Feb, 2007.

[4]  Marian P. Kazmierkowski, R. Krishnan, Frede Blaabjerg, Control in Power Electronics Selected Problems, Elsevier Science, USA, 2002.

[5] Howard Demuth, Mark Beale, Martin Hagan, Neural Net-work Toolbox 5 User's Guide, by The MathWorks, 2005–2007.

[6] JOHN CHIASSON, MODELING AND CONTROL OF ELECTRIC MACHINES HIGH-PERFORMANCE, IEEE Press Series on Power Engineering, John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.

[7] M. Jemli, M. Boussak, M. Gossa, M.B.A. Kamoun, Fail-safe digital implementation of indirect field oriented controlled induction motor drive, ELSEVIER, Simulation Practice and Theory , pp 233-252, 2000.

[8] Marcello Montanari, sergei Peresada, Speed Sensorless  Control of Induction Motor based on Indirect Field-Orientation, Supplied by the British Library-" The world's knowledge", IEEE, pp 1858-1865, 2000.

## LIST OF SYMBOLS

$i_{ds}, i_{qs}$        d-axis and q-axis stator current components.

$i_{dr}, i_{qr}$        d-axis and q-axis rotor current components.

$v_{ds}, v_{qs}$        d-axis and q-axis stator voltage components.

$v_{dr}, v_{qr}$        d-axis and q-axis rotor voltage components.

$L_{ls}, L_{lr}$        stator and rotor leakage inductance.

$L_s, L_r$        stator and rotor inductance.

$L_m$        magnetizing inductance.

$R_s, R_r$        stator and rotor resistance.

$\psi_{dm}, \psi_{qm}$        d-axis and q-axis airgap flux linkage d-q component.

$\omega_{sl}$        slip frequency.