

Sensorless Control of Induction Motor with Extended Kalman Filter on TMS320F2812 Processor

R. Gunabalan¹, V. Subbiah² and B. Rami Reddy³

¹Sri Ramakrishna Engineering College/EEE (PG), Coimbatore-641 022, TamilNadu, India

Email: gunabalanr@yahoo.co.in

²Sri Krishna College of Engineering and Technology/EEE, Coimbatore-641 009, TamilNadu, India

Email: subbiah42@yahoo.com

³Pondicherry Engineering College/EEE, Puducherry, India

Email: ramireddy@yahoo.com

Abstract–This paper describes discretized Extended Kalman Filter (EKF) for speed estimation of squirrel cage induction motor. Speed and rotor fluxes are estimated from the measured terminal voltages and currents without anymore additional filter. There is a mathematical basis that yields excellent simulation and real time results. In this, speed is considered as a parameter and state. The speed of an induction motor is estimated by MATLAB simulation under low speed regions and it is verified using TMS320F2812 DSP Code Composer Studio. Estimated values along with other measured states are used for closed loop control. The current control is performed under synchronously rotating reference frame and the estimated speed information is used for the reference frame transformation of the current controller. At low speeds, where noise is predominant, performance is improved due to the filtering action of the EKF.

Index Terms– Extended Kalman Filter, Sensorless control

I. INTRODUCTION

Sensorless vector control of an induction motor drive essentially refers to vector control without any speed sensor. Speed sensor is not suitable for the environmental conditions, which suffers due to large shocks. Sensorless vector control is suitable from the point of reliability of the equipment, cost effectiveness and less maintenance. The speed is estimated from the measured terminal voltages and currents. Full order observers in [5] – [7] estimate all the state variables and are sensitive to any noise in the measurements of currents, voltages and speed. They require filters to remove the noise signals.

EKF is employed to estimate the motor state variables by only using the measurements of stator voltages and currents [1], [3]. During the estimation procedure, these signals are not filtered. The accuracy of the speed estimation depends on the motor parameter variations and the accuracy of measured voltages and currents. The EKF is a recursive optimum stochastic state estimator which can be used for the joint state and parameter estimation of a non-linear dynamic system in real time by using noisy monitored signals that are disturbed by random noise.

The main contributions of this paper are organized as follows. Section II reviews the EKF algorithm and implementation is given in section III. Induction motor drive

system configuration is discussed in section IV. Simulation and CCS results are discussed in section V.

II. EKF ALGORITHM

The main design steps for a speed sensorless induction motor drive implementation using the discretized EKF algorithm are as follows:

- A. Selection of the time domain induction machine model.
- B. Discretization of the machine model.
- C. Determination of the noise and state covariance matrices Q, R and P.
- D. Implementation of the discretized EKF algorithm.
- E. Tuning of the covariance matrices.

A. Time Domain Induction Machine Model

To estimation the rotor speed of an induction machine using EKF, various machine models have been used. Generally the model is expressed in stator flux oriented reference frame. In EKF, the rotor speed is considered as a state variable and parameter. This makes the system matrix A nonlinear i.e. $A = A(X)$.

The state space representation of the system is as follows:

$$\frac{dX}{dt} = AX + BU \quad (1)$$

$$Y = CX \quad (2)$$

$$X = \left[i_{ds}^s \quad i_{qs}^s \quad \phi_{dr}^s \quad \phi_{qr}^s \quad \omega_r \right]^T$$

$$Y = \left[i_{ds}^s \quad i_{qs}^s \right]^T = i_s$$

$$U = \left[v_{ds}^s \quad v_{qs}^s \right]^T$$

$$A = \begin{bmatrix} \frac{-1}{T_s^*} & 0 & \frac{L_m}{L_s' L_r \tau_r} & \frac{\omega_r L_m}{L_s' L_r} & 0 \\ 0 & \frac{-1}{T_s^*} & \frac{-\omega_r L_m}{L_s' L_r} & \frac{L_m}{L_s' L_r \tau_r} & 0 \\ \frac{L_m}{\tau_r} & 0 & \frac{-1}{\tau_r} & -\omega_r & 0 \\ 0 & \frac{L_m}{\tau_r} & \omega_r & \frac{-1}{\tau_r} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{1}{\sigma L_s} & 0 \\ 0 & \frac{1}{\sigma L_s} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

where,

$$\frac{1}{T_s^*} = \frac{R_s + R_r (L_m / L_r)^2}{L_s'}$$

$$L_s' = \sigma L_s$$

R_s, R_r -stator and rotor resistance (ohm)

L_s, L_r -stator and rotor self inductance (H)

L_m -mutual inductance (H)

σ -leakage coefficient

$$\sigma = 1 - \frac{L_m^2}{L_s L_r}$$

$$\tau_r \text{-rotor time constant} = \frac{L_r}{R_r}$$

ω_r -motor angular velocity (rad/s)

B. Discretized Induction Machine Model

For digital implementation of the EKF, the discretized machine equations are required. These can be obtained from (1) and (2).

$$X(k+1) = A_d X(k) + B_d U(k) \quad (3)$$

$$Y(k) = CX(k) \quad (4)$$

$$A_d \approx I + AT \quad (5)$$

$$B_d \approx BT \quad (6)$$

$$A_d = \begin{bmatrix} 1 - \frac{T}{T_s^*} & 0 & \frac{TL_m}{L_s' L_r \tau_r} & \frac{\omega_r TL_m}{L_s' L_r} & 0 \\ 0 & 1 - \frac{T}{T_s^*} & \frac{-\omega_r TL_m}{L_s' L_r} & \frac{TL_m}{L_s' L_r \tau_r} & 0 \\ \frac{TL_m}{\tau_r} & 0 & 1 - \frac{T}{\tau_r} & -T\omega_r & 0 \\ 0 & \frac{TL_m}{\tau_r} & T\omega_r & 1 - \frac{T}{\tau_r} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$B_d = \begin{bmatrix} \frac{T}{L_s} & 0 \\ 0 & \frac{T}{L_s} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} C_d = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

To achieve adequate accuracy, the sampling time should be appreciably smaller than the characteristic time constants of the machine. The final choice for the sampling time should be based on obtaining adequate execution time of the full EKF algorithm and also satisfactory accuracy and stability.

The system noise is represented by $v(k)$ (v -noise vector of the states) which is assumed to be zero-mean and white Gaussian noise. It is independent of $X(k)$ and its covariance matrix is Q and the system model becomes

$$X(k+1) = A_d X(k) + B_d U(k) + v(k) \quad (7)$$

The measurement noise is represented by $w(k)$ which is assumed to be zero-mean and white Gaussian noise. It is independent of $X(k)$ and $V(k)$ and its covariance matrix is P and the output equation becomes

$$Y(k) = CX(k) + w(k) \quad (8)$$

C. Determination of the Noise and State Covariance Matrices Q, R and P

The purpose of the Kalman filter is to obtain the unmeasurable states (e.g. rotor speed) by using the measured states and also the statistics of the noise and measurements. In general, the computational inaccuracies, modeling errors and errors in the measurements are considered by means of noise inputs. A critical part of the design is to use correct initial values for the covariance matrices.

The elements of Q and R depend on the number of state variables [2], [4]. The system noise matrix Q is a five-by-five matrix and the measurement noise matrix R is a two-by-two matrix. This should require the knowledge of 29 elements. However, by assuming that the noise signals are not correlated (no relation between the covariance noise matrices), both Q and R are diagonal and only 5 elements must be known in Q and 2 elements in R .

Generally, the parameters in the direct and quadrature axes are same. This implies that the first two elements in the diagonal of Q are equal ($q_{11} = q_{22}$) and the third and fourth elements are also equal. So $Q = \text{diag}(q_{11}, q_{11}, q_{33}, q_{33}, q_{55})$ contains only 3 elements. Similarly, the two diagonal elements in R are equal ($r_{11} = r_{22} = r$), thus $R = \text{diag}(r, r)$. It follows that in total only 4 noise covariance elements must be known.

$$Q = \begin{bmatrix} q_{11} & 0 & 0 & 0 & 0 \\ 0 & q_{11} & 0 & 0 & 0 \\ 0 & 0 & q_{33} & 0 & 0 \\ 0 & 0 & 0 & q_{33} & 0 \\ 0 & 0 & 0 & 0 & q_{55} \end{bmatrix} R = \begin{bmatrix} r & 0 \\ 0 & r \end{bmatrix}$$

D. Implementation of the Discretized EKF Algorithm

The structure of the EKF is shown in Fig. 1. The state estimates are obtained by the EKF algorithm in the following steps:

Step1: Initialization of the state vector and covariance matrices

Starting values of the state vector $X_o = X(t_o)$, the noise covariance matrices $Q_o(5 \times 5)$ and $R_o(2 \times 2)$ are set together with the starting value of the state covariance matrix $P_o(5 \times 5)$. The starting value of the state covariance matrix can be considered as a diagonal matrix where all the elements are equal. The initial values of the covariance matrices reflect the degree of knowledge of the initial states.

Step2: Prediction of the state vector

Prediction of the state vector at the sampling instant $(k+1)$ is obtained from the input $U(k)$ and the state vector $\hat{X}(k)$.

$$X^*(k+1) = A_d \hat{X}(k) + B_d U(k) \quad (9)$$

where,

$$\hat{X} = [\hat{i}_{ds}(k) \quad \hat{i}_{qs}(k) \quad \hat{\phi}_{dr}(k) \quad \hat{\phi}_{qr}(k) \quad \hat{\omega}_r(k)]^T$$

$$U(k) = [V_{ds}(k) \quad V_{qs}(k) \quad 0 \quad 0 \quad 0]^T$$

Step3: Covariance estimation of prediction

The covariance matrix of prediction is estimated as

$$P^*(k+1) = f(k+1)\hat{P}(k)f^T(k+1) + Q \quad (10)$$

where, f is the gradient matrix.

$$f(k+1) = \frac{\partial}{\partial x} [A_d X + B_d U]_{x=\hat{x}(k+1)} \quad (11)$$

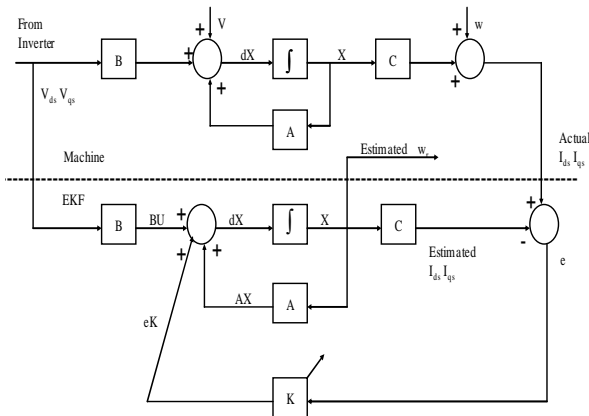


Figure 1. Structure of EKF

$$f(k+1) = \begin{bmatrix} 1 - \frac{T}{T_s^*} & 0 & \frac{TL_m}{L_s L_r \tau_r} & \frac{\omega_r TL_m}{L_s L_r} & \frac{TL_m}{L_s L_r} \phi_{qr} \\ 0 & 1 - \frac{T}{T_s^*} & -\frac{\omega_r TL_m}{L_s L_r} & \frac{TL_m}{L_s L_r \tau_r} & -\frac{TL_m}{L_s L_r} \phi_{dr} \\ \frac{TL_m}{\tau_r} & 0 & 1 - \frac{T}{\tau_r} & -T\omega_r & T\phi_{qr} \\ 0 & \frac{TL_m}{\tau_r} & T\omega_r & 1 - \frac{T}{\tau_r} & T\phi_{dr} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

where,

$$\omega_r = \hat{\omega}_r(k+1) \quad \phi_{dr} = \hat{\phi}_{dr}(k+1) \quad \phi_{qr} = \hat{\phi}_{qr}(k+1)$$

In $f(k+1)$, 17 elements are constants and remaining 8 elements are variables. In a practical DSP application it is useful to compute first $f(k+1)$, since it contains the constant required in $X^*(k+1)$. This leads to reduced memory requirements and reduced computational time.

Step4: Kalman filter gain computation

For induction motor applications, the Kalman gain matrix contains 2 columns and 5 rows.

$$K(k+1) = P^*(k+1)h^T(k+1)[h(k+1)P^*(k+1)h^T(k+1) + R]^{-1} \quad (12)$$

$h(k+1)$ is a gradient matrix, defined as

$$h(k+1) = \frac{\partial}{\partial x} [C_d X]_{x=x^*(k+1)} \quad (13)$$

$$h(k+1) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Step5: State vector estimation

The state vector estimation (corrected state vector estimation, filtering) at time $(k+1)$ is performed as follows:

$$\hat{X}(k+1) = X^*(k+1) + K(k+1)[Y(k+1) - \hat{Y}(k+1)] \quad (14)$$

$$\hat{Y}(k+1) = C_d X^*(k+1) \quad (15)$$

Step6: Covariance matrix of estimation error

The error covariance matrix is obtained from

$$\hat{P}(k+1) = P^*(k+1) - K(k+1)h(k+1)P^*(k+1) \quad (16)$$

Step7: Updation

Put $k = k+1$, $X(k) = X(k-1)$, $P(k) = P(k-1)$ and go to step2.

The EKF described above can be used to estimate the speed of an induction motor under both steady state and transient conditions.

E. Tuning of the Covariance Matrices

The tuning of the EKF involves an iterative modification of the covariance in order to yield the best estimates of the states [6]. Changing the covariance matrices Q and R affects both the transient and steady state operation of the filter. Increasing Q corresponds to stronger system noises or larger uncertainty in the machine model used. If the covariance R is increased, this corresponds to the fact that the measurements of the currents are subjected to a stronger noise and should be weighted less by the filter. Thus the filter gain matrix elements will decrease and this results in slower transient performance. The initial values of Q and R are selected randomly and tuned accordingly. From experience, the value of Q_{55} is chosen higher than remaining elements in Q matrix and the values of R matrix elements are higher than Q matrix elements.

III. EKF IMPLEMENTATION

The EKF algorithm is written in C language and the data required (direct and quadrature axes voltages and currents) for the estimation algorithm are obtained from Matlab. Using M-file program the Matlab simulink block is linked with CCS. The digital values of voltages and currents are stored in a file.

CCS C program takes the input data from Matlab workspace and it estimates the rotor speed. Depending upon the sampling time, the error in the estimated speed is minimized. Due to limitations in the memory size (when Matlab is linked with CCS in the simulator mode), only 32 Kbyte data can be processed. The sampling time is 0.002 sec. The flow diagram for EKF algorithm is shown in Fig.2. Number of samples taken is 4000 (N=4000). The estimated values are stored in the data memory and are plotted. Debugging is done with the help of disassembly window.

IV. SYSTEM CONFIGURATION

The rating of the induction motor is given in Table I. The speed of an induction motor is estimated by EKF algorithm. The system configuration is shown in Fig.3. The estimated speed is used for overall control.

Motor Rating	
Output	3.7 kW
Poles	4
Voltage	160 V
Current	20 A
Motor Speed	1500 rpm
R_s	0.3831 ohm
R_r	0.2367 ohm
L_s	33.34 mH
L_r	33.34 mH
L_m	32.11 mH

The estimated speed is compared with the actual speed and the error is given to PI controller which produces the required torque reference. From the estimated speed and the torque reference, the direct and quadrature axes reference currents are calculated. They are transformed to three phase quantities using inverse Park transformation. The reference three phase currents are compared with the actual currents. The error is given to the hysteresis current controller. Output of the controller produces the required control signal and it is given to the semiconductor switches. The output of the CRPWM drives the induction motor.

V. SIMULATION AND CCS RESULTS

In this work, EKF algorithm is used to estimate the speed of the induction motor. Direct vector control method is applied to calculate the reference currents responsible for torque and flux. Simulations are carried out for speed reversal test, step change in load.

A. Speed Reversal Test

From the measured terminal voltages and currents the speed is estimated using EKF algorithm. Fig.4 shows the simulation and CCS results for a step change in speed. The sampling time is kept at 0.002 sec. Reducing the sampling time increases the accuracy and improves the drive performance. The algorithm is verified under low speed operations. Initially the reference speed is set at 50 rpm. At t = 3 sec, speed is reversed to -50 rpm. The simulation response is measured without a load. The deviation in the estimated speed by Matlab simulation and CCS is shown in Fig. 5 and it is around 4 rpm (8%). Clearly the estimated speed follows the speed command; and, as can be seen in Fig. 4, the estimated speed matches the actual rotor speed of the motor. It ensures that the motor speed is accurately estimated. Stator current and rotor fluxes are also estimated along with speed. Fig. 6 and 7 illustrate the estimated direct and quadrature axes stator currents by both Matlab and CCS. The estimated direct and quadrature axes rotor fluxes by both Matlab and CCS are shown in Fig.8 and 9.

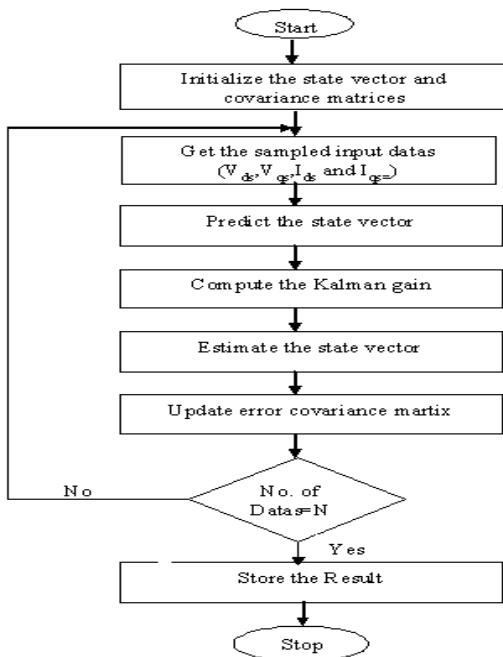


Figure 2. Flow Chart for EKF Algorithm

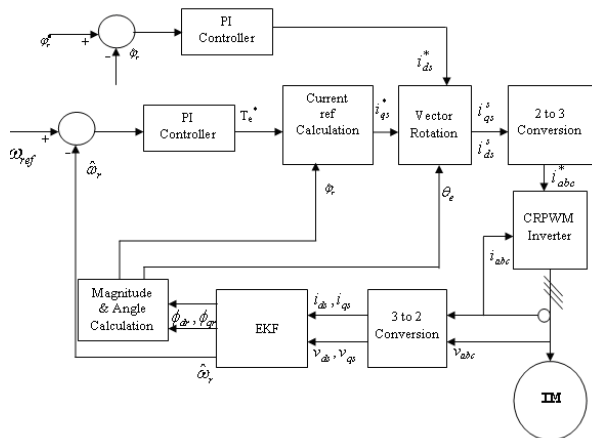


Figure 3. System Configuration

TABLE I
RATING AND PARAMETERS OF INDUCTION MOTOR

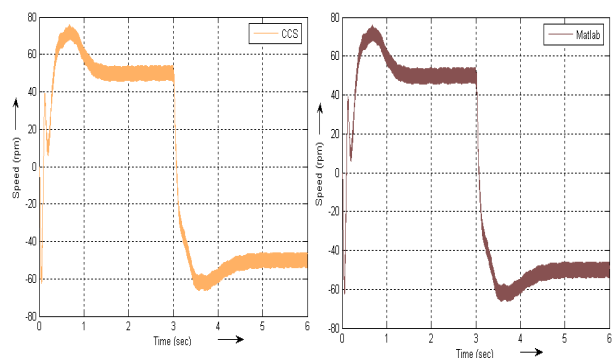


Figure 4. Speed response

Figure 5. Deviation in the estimated speed by Matlab and CCS

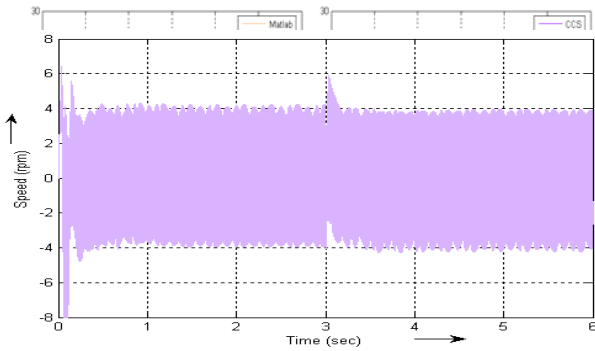


Figure 6. Direct axis stator current

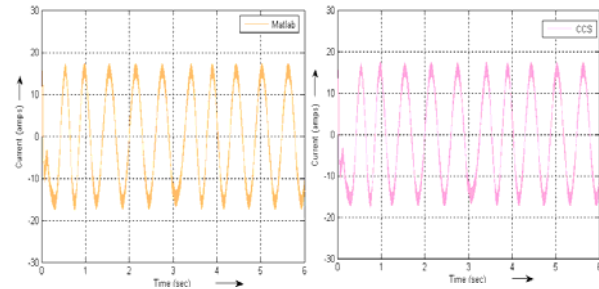


Figure 7. Quadrature axis stator current

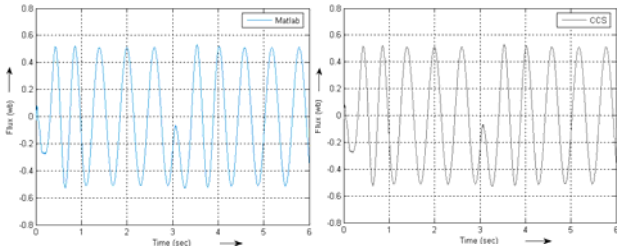


Figure 8. Direct axis rotor flux

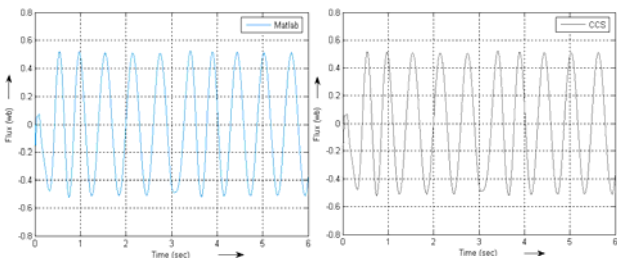


Figure 9. Quadrature axis rotor flux

B. Step Change in Load

The speed of the motor is maintained constant at 50 rpm. At $t = 0$, the load torque is zero and at $t = 3$ sec, a load of 2.5 Nm is applied to the motor. The speed response is shown in Fig. 10 and the deviation in the estimated speed by CCS and Matlab is shown in Fig. 11. The estimated direct axis stator current and the deviation in the estimated current by CCS and Matlab are depicted in Fig.12 and 13. In the CCS program, all the mathematical calculations are performed and values are used directly with three digit approximation to make the program simple but in Matlab the variables are declared directly and there are no hand mathematical calculations involved. This makes some deviation in the estimated speed by Matlab and CCS. Fig. 14 and 15 illustrate the estimated quadrature axis stator current and the deviation in the estimated current by

Matlab and CCS. Fig. 16 and 17 illustrate the estimated direct and quadrature axes rotor flux.

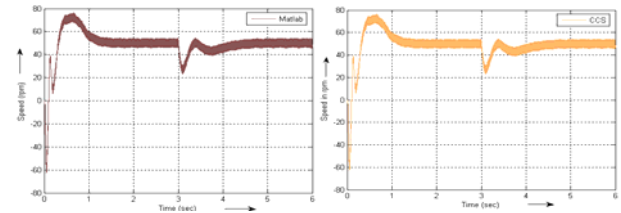


Figure 10. Speed response

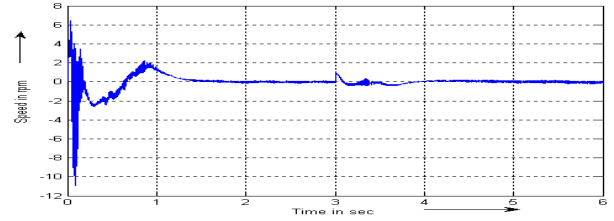


Figure 11. Deviation in the estimated speed by Matlab and CCS

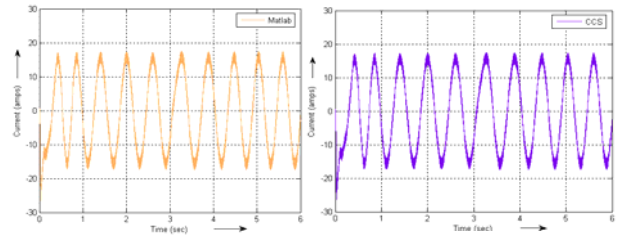


Figure 12. Direct axis stator current

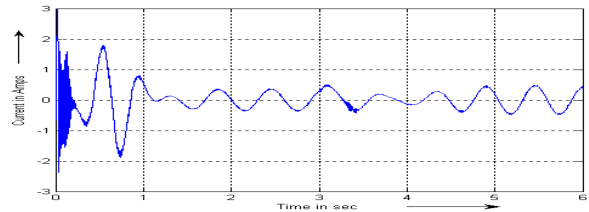


Figure 13. Deviation in d-axis stator current by Matlab and CCS

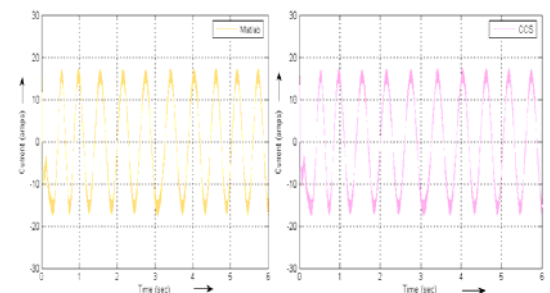


Figure 14. Quadrature axis stator current

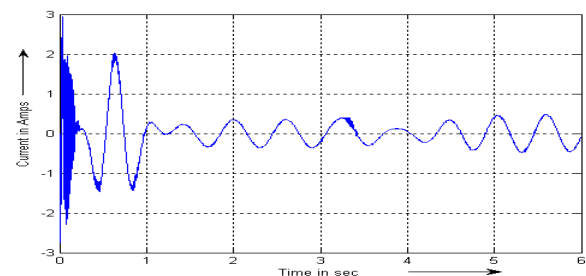


Figure 15. Deviation in q- axis stator current by Matlab and CCS

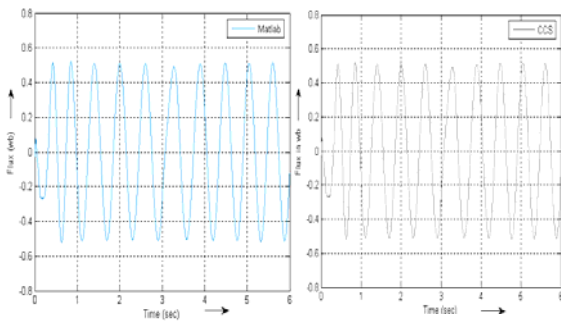


Figure 16. Direct axis rotor flux

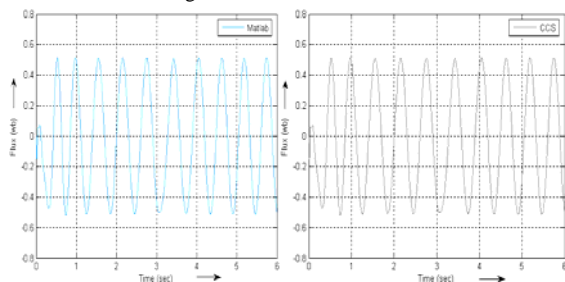


Figure 17. q-axis rotor flux

The difference between the estimated speed by CCS and Matlab is shown in Fig.11 and it is less than 1 rpm (around 0.5%). The difference between the estimated stator current (both d-axis and q-axis) by CCS and Matlab is illustrated in Fig.13 and Fig.15 and its average value is approximately 0.5 amp (error is 2%).

VI. CONCLUSION

As an estimator, the Extended Kalman filtering technique is investigated to observe the rotor speed. For this purpose, appropriate mathematical model of induction machine is studied and discretized for real-time applications. Several simulations are illustrated to examine the performance of EKF. The oscillation in the estimated speed is reduced by decreasing the initial values of the covariance matrices. It is also verified by Code Composer Studio version 2.2 to implement the EKF algorithm in real time. The algorithm can be verified using TMS320F2812 DSP processor. The mathematical complexity and difficulty of EKF is a significant disadvantage when compared to Luenberger observer. And for practical cases where the uncertainties are significant, EKF will provide better state estimation.

REFERENCES

- [1] B. K. Bose, *Modern Power Electronics and AC drives*, Upper Saddle River, N. J. Prentice Hall, Third edition, 2003, pp.388-400 and pp.348-350.
- [2] Bozo Terzic and Martin Jadric, "Design and implementation of Extended Kalman Filter for the speed and rotor position estimation of Brushless DC motor", *IEEE transactions on Industry Electronics*, Vol.48, No.6, 2001, pp.1065-1073.
- [3] Peter Vas, "Sensorless vector and direct torque control", Oxford University Press, 1998, pp.470-491.
- [4] Young-Real Kim, Seung-Ki Sul, Min-Ho Park, "Speed sensorless vector control of an Induction Motor using an Extended Kalman Filter" *IEEE Transactions on Industry Applications*, Vol.30, No.5, 1994, pp.594-599.
- [5] Hisao Kubota, Kouki Matsuse and Takayoshi Nakano, "DSP based speed adaptive flux observer of induction motor", *IEEE transactions on Industry Applications*, Vol.29, No.2, 1993, pp.344-348.
- [6] K.Matsuse, Y.Kouno,H.Kawai and J.Oikawa, " Characteristics of speed sensor-less vector controlled dual induction motor drive connected in parallel fed by a single Inverter", *IEEE Trans. Ind. Applications.*, vol. 40, pp. 153-161, Jan./Feb. 2004.
- [7] K. Matsuse, Y. Kouno, H. Kawai and S. Yokomizo, "A speed-sensor-less vector control method of parallel-connected dual induction motor fed by a single inverter," *IEEE Trans. Ind. Applications.*, vol. 38, pp. 1566-1571, Nov./Dec. 2002.