Design and Implementation of a Direct Torque Control of Induction Machine utilizing a Digital Signal Processor and the Field Programmable Gate Arrays

C. L. Toh, N. R. N. Idris and A. H. M. Yatim
Department of Energy Conversion
University Teknologi Malaysia
Johor, MALAYSIA
tohchuenling@yahoo.com

Fazlli Patkar
Faculty of Electrical Engineering,
Kolej Universiti Teknikal Kebangsaan Malaysia,
Melaka, MALAYSIA

Abstract— **Employing both DSP and FPGA in motion control will give advantages of rapid prototyping and higher switching frequency in Direct Torque Control (DTC) drive. This paper presents a high performance DTC induction motor drive with constant switching frequency and low torque and flux ripples. The experimental results prove the feasibility of the proposed controllers, which is achieved using the combination of a DSP and an FPGA.**

Keywords-DTC; DSP; FPGA; Induction Machine

I. INTRODUCTION

Before the microprocessor is widely used in motion control, the control of electrical drives has been dominated by analog technology [1]. Analog circuitry gives an infinite bandwidth with fast response; however it suffers the disadvantages of noise, complex circuitry and difficulty in circuit modification.

Today, the rapid development in high-performance Digital Signal Processors (DSP) not only replaced the analog technology in conventional control method but also provides high computing capabilities. The digital control method has the advantages of simple circuitry. In [1], it is highlighted that, the implementation of complex signal processing and advanced algorithms are possible and the number of sensors used is reduced (in particular position and flux sensor). Nevertheless, a major drawback of DSP is the lack of on chips resources, which is then resulting in an increase of system hardware complexity. Moreover, it requires a high sampling rate to achieve a wide bandwidth performance [2]. Some researchers solved the problem by using dual-DSP to implement the controller; however, this will increase the complexity and cost.

Recent developments in Very-Large-Scale Integration (VLSI) and Application-Specific-Integrated Circuit (ASIC) have made possible to combine complex analog and digital circuits [1], [3], [4]. ASIC methodology offers a reduction in chips count that can lower significantly the fabrication cost and improve the system reliability. However, in motion control systems, ASIC technology is not that popular. This is because the lack of flexibility to modify or to adapt the design to different types of motor drives, once the chip is built [1], [5]. A special class of ASIC known as Field Programmable Gate Arrays (FPGA's) offers a realistic alternative. It can significantly reduce design risk since a design error can be corrected immediately and intensively by re-programming the FPGA.

This paper presents the implementation of the proposed DTC drive controllers, in which the theory and principal was fully discussed in our previous paper [6]. The proposed DTC replaces the hysteresis-based controllers with fixed switching controllers that operate based on the comparison between the compensated signals and the triangular waveforms. In order to avoid the quantization problems, the DSP is used for floating point calculation to estimate the torque and stator flux. The main task of the FPGA is to implement the fixed switching frequency controller. The FPGA is also responsible in selecting a proper voltage vector and implementing the blanking time for the inverter. The rest of the paper is organized as follows. Section II briefly introduces the proposed controllers for DTC drives. Section III describes the hardware implementation while section IV presents the experimental results. Finally, conclusions are given in section V.

II. PROPOSED DTC

DTC is characterized by its simple control structure. Figure 1 shows the hysteresis-based DTC as proposed by Takahashi [7]. It consists of a pair of torque and flux hysteresis controllers, voltage vector selection table, torque and flux estimator and a VSI. Two major problems that are usually associated with DTC drives are 1) the variable switching frequency due to the hysteresis controllers and 2) large torque and flux ripples. This paper proposes a simple yet effective method of overcoming these problems, i.e. by introducing new torque and flux controllers. These controllers are shown in Figure 2.

0-7803-9296-5/05/$20.00

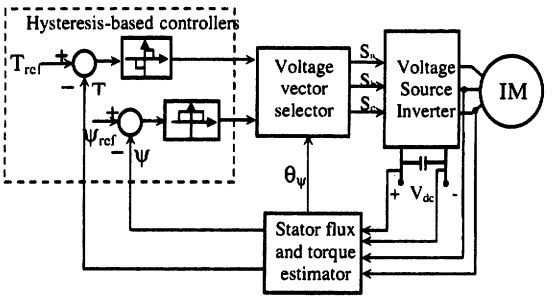


Figure 1. Hysteresis-based DTC

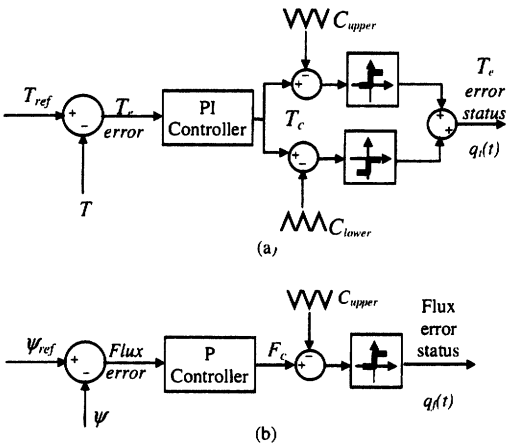


Figure 2. Proposed controllers: (a) torque controller, (b) flux controller

The proposed torque controller consists of two triangular waveform generators, two comparators and a PI controller. The two triangular waveforms (C_{upper} and C_{lower}) are 180° out of phase with each other. The absolute values of the DC offsets for the triangular waveforms are set to half of their peak-peak values. The instantaneous output of the proposed torque controller, $q_t(t)$ is same as the three level hysteresis comparator [6], that can be one of the three states: –1, 0 or 1 with the following conditions.

$$q_t(t) = \begin{cases} 1 & for\ T_c > C_{upper} \\ 0 & for\ C_{lower} < T_c < C_{upper} \\ -1 & for\ T_c > C_{lower} \end{cases} \quad (1)$$

The proposed flux controller works similar to the torque controller. As in the hysteresis-based controller, there are only two levels of output generated from the controller, i.e. 1 to increase the flux and 0 to reduce the flux. This implies that only single triangular waveform is required. The output of the controller, termed as flux error status, is given by:

$$q_f(t) = \begin{cases} 1 & for\ F_c \geq C_{upper} \\ 0 & for\ F_c < C_{lower} \end{cases} \quad (2)$$

The proposed controllers are designed step-by step, starting from modeling, averaging and linearizing the respective loop. Linear control system theory is the fundamental of the design. The inverter switching frequency is mainly determined by the switching frequency of the torque controller. Thus, a high switching frequency can be achieved by setting a high triangular waveform frequency. High switching frequency is desired in order to minimize the torque ripple, reduce the current harmonic contents as well as reduce the audible sound. In this paper, the switching frequency is increased by increasing the triangular carriers' frequency. The sampling frequency is set to twice of the triangular carrier frequency and it is synchronized as shown in Figure 3. For this particular implementation, the switching frequency is set to about 10 kHz. It should be noted however that it is possible to increase the triangular frequency to a much higher value as long as it is synchronized with the sampling frequency of the DSP. The flux loop triangular carrier frequency must be set to a lower value to avoid the stator flux weakening due to the stator flux reduction when zero voltage vectors are selected. In this paper, the triangular frequency for the flux controller is set to about 5 kHz.

III. HARDWARE IMPLEMENTATION

The proposed controllers are simple in concept and easy to implement. Figure 4 shows the block diagram of the experimental set-up. The main components of the set-up consist of a digital signal processor board, DS1102 from dSPACE (TMS320C31 at 60 MHz) and an Altera UP1 Education Board (EPF10K20).

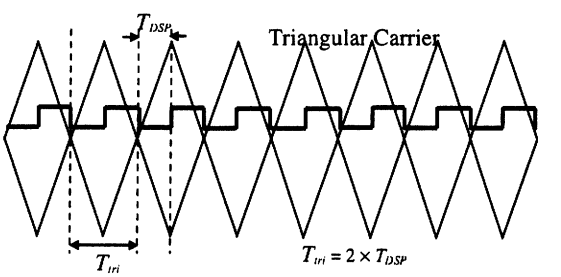


Figure 3. Synchronization of the torque loop triangular carrier frequency and the sampling frequency

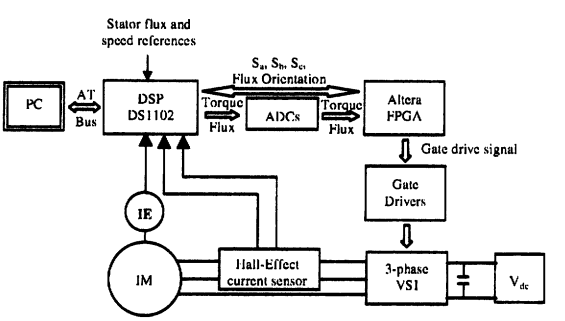


Figure 4. Block diagram of the experiment set-up

A. DSP – DS1102 Controller Board

The DS1102 board is built around the Texas Instruments TMS320C31 third generation floating-point Digital Signal processor (DSP). It is interfaced to the PC via the AT PC slot and is programmed by using C language. The DSP performs two major tasks: 1) estimates the stator flux and torque, and 2) determines the stator flux orientation.

The estimation of stator flux and torque are based on measured 2 phases currents and voltage vectors. The third phase current can be constructed from the other two phase currents due to the isolated neutral of the stator winding in the induction motor.The following equations described the three-phase to two-phase transformation of currents and voltages.

$$i_d = \frac{2}{3}\left(i_a - \frac{1}{2}(i_b + i_c)\right) \tag{3}$$

$$i_q = \frac{1}{\sqrt{3}}(i_b - i_c) \tag{4}$$

$$v_d = \frac{2}{3}V_{dc}\left(S_a - \frac{1}{2}(S_b + S_c)\right) \tag{5}$$

$$v_q = \frac{1}{\sqrt{3}}V_{dc}(S_b - S_c) \tag{6}$$

S_a, S_b and S_c are the switching states (1 or 0), obtained from look–up table implemented using the FPGA. Theoretically, the d-q stator flux may easily be estimated by using the stator voltage model as given by,

$$\psi_d = \int v_d - i_d R_s \, dt \tag{7}$$

$$\psi_q = \int v_q - i_q R_s \, dt \tag{8}$$

However, the pure integrator in equations (7) and (8) create problem of integration drift [9]. Because of this, low-pass filter is normally used to replace the pure integrator, with appropriate cut-off frequency. The discrete form of low-pass filter, which are implemented using the DSP, are given by

$$\psi_{d,n} = \left(\left(v_{d,n} - i_{d,n}R_s\right)T_s + \psi_{d,n-1}\right)\frac{1}{1 + T_s\omega_{cutoff}} \tag{9}$$

$$\psi_{q,n} = \left(\left(v_{q,n} - i_{q,n}R_s\right)T_s + \psi_{q,n-1}\right)\frac{1}{1 + T_s\omega_{cutoff}} \tag{10}$$

where T_s is the sampling time of the DSP and ω_{cutoff} is the cut-off frequency of the low pass filter. The magnitude of stator flux and torque are obtained by using the following,

$$|\psi| = \sqrt{\psi_d^2 + \psi_q^2} \tag{11}$$

$$T_e = 1.5\left(\psi_d i_q - \psi_q i_d\right) \tag{12}$$

The estimated torque and flux are compared with the reference values which are then fed to the respective controllers for compensation. The outputs of these controllers are next fed to the FPGA.

Theoretically the stator flux orientation is calculated by *arctan* (ψ_q/ψ_d). However, processing of *arctan* (ψ_q/ψ_d) in DSP will increase the execution time. It is not the exact position of the stator flux, but the sector in which the stator flux is located is required. Therefore, the sector can be obtained by considering only the signs of the three-phase stator flux components. This allows a simple implementation, which requires only the use of hysteresis comparators. Equations (13) – (15) give the transformation of stator flux from two-phase to three-phase.

$$\psi_a = \psi_d \tag{13}$$

$$\psi_b = -\frac{1}{2}\psi_d + \frac{\sqrt{3}}{2}\psi_q \tag{14}$$

$$\psi_c = -\frac{1}{2}\psi_d - \frac{\sqrt{3}}{2}\psi_q \tag{15}$$

In order to reduce the burden on the DSP, the sign of ψ_b and ψ_c can be obtained by examining the sign of:

$$\psi_b' = 2\psi_b = -\psi_d + \sqrt{3}\psi_q \tag{16}$$

$$\psi_c' = 2\psi_c = -\psi_d - \sqrt{3}\psi_q \tag{17}$$

Figure 5 illustrates how the stator flux orientation is determined. The highlighted sectors in Figure 5 (a), (b), and (c) indicate the positive sign of ψ_a, ψ_b and ψ_c respectively. Theses results are summarized in Table I. Logic '1' indicates the positive sign while Logic '0' represents the negative sign.

B. FPGA – Altera UP1 Education Board

Altera UP1 Education Board is a stand-alone experiment board based on two Altera's leading device families: CPLD – MAX®7000 and FPGA – FLEX®10K. The EPF10K20 device is ideal for advanced designs with a 240-pin RQFP package and has 1152 logic elements and 6 embedded array blocks [10]. Very high-speed integrated circuits, Hardware Description Language (VHDL) is utilized in digital logic designs. The design is then compiled and simulated using MAX + PLUS II and finally downloaded to the EPF10K20 device. The ByteBlaster download cable channels configuration data between the MAX+PLUS II software and the UP1 Education Board. The functional block diagram of the tasks performed by the Altera FPGA is shown in Figure 6.

A synchronous control of for the two ADCs is designed to ensure that the torque and flux error signals are passed to the FPGA simultaneously. A pair of 8-bits register captures the data from ADC once the data is valid. The torque error signal is sent to the torque controller to generate the appropriate torque logic status.

TABLE I. SELECTION OF THE STATOR FLUX SECTOR

Sector / Sign of stator flux	I	II	III	IV	V	VI
Sign of ψ_a	1	1	1	0	0	0
Sign of ψ_b	0	0	1	1	1	0
Sign of ψ_c	1	0	0	0	1	1

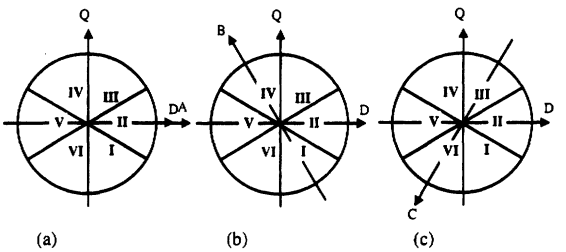


Figure 5. Determining sector of the stator flux. Highlighted sectors indicate the positive sign of (a) ψ_a, (b) ψ_b and (c) ψ_c respectively

The flux controller compares the flux error signals and gives the flux logic status. The DSP passes the flux orientation to the FPGA directly and store it in a 3-bit register. In order to avoid the glitch problem, which may cause instability to the system, a 6-bits register is placed in front of the ROM. It will hold the torque error status; flux error status and stator flux orientation and then pass to the ROM address. The 64-by-3 ROM memories are required to tabulate the Takahashi's voltage vector selection table (Table II). The three bit outputs of the memory element are passed to the blanking time generator before they are fed to the gate drivers.

In order to implement the proposed torque and flux controllers, a pair of triangular carrier generator is designed. For the torque loop, the upper and lower triangular carriers are generated simultaneously. On the other hand, the proposed flux controller will require a single triangular carrier since there are only 2 levels of output is generated by the controller. All carrier signals are synchronized with the DSP sampling time. The frequency of the torque triangular carrier is ½ of the DSP sampling frequency while the flux triangular carrier frequency is set to ¼ of the DSP sampling frequency. The amplitudes of the triangular carriers are set to 100 units.

Both triangular generators are initialized by a LOW signal from the DSP, namely signal 'a', when it starts sampling the first data. For the torque controller, the upper triangular, C_{upper} counts up while the lower triangular, C_{lower} counts down. When the DSP samples the second data, signal 'a' will send a HIGH signal to the FPGA. Then the C_{upper} will count down and the C_{lower} will count up. The flux loop triangular carrier, C counts up during the first and second sampling time. It starts to count down when the third data is sampled by the DSP. Therefore, the signal 'a' is sent to a 2-bit counter in order to generate a control signal, 'b' for the flux loop triangular carrier. Figure 7 shows the flow chart of the triangular generation for both the torque and flux controllers. Figure 8 shows the generated triangular carriers.

IV. Simulation & Experimental Results

The parameters of a ¼ HP induction machine used in the experiment are tabulated in Table III. The widths of the flux and torque hysteresis band are set to 10% of their rated values respectively. For the proposed controllers, the numerical values of the parameters for the torque and flux loops are calculated and listed in Table IV. The triangular frequency of the torque and flux controllers are set to 10.4 kHz and 5.2 kHz respectively, each with a peak-peak of 100 units.

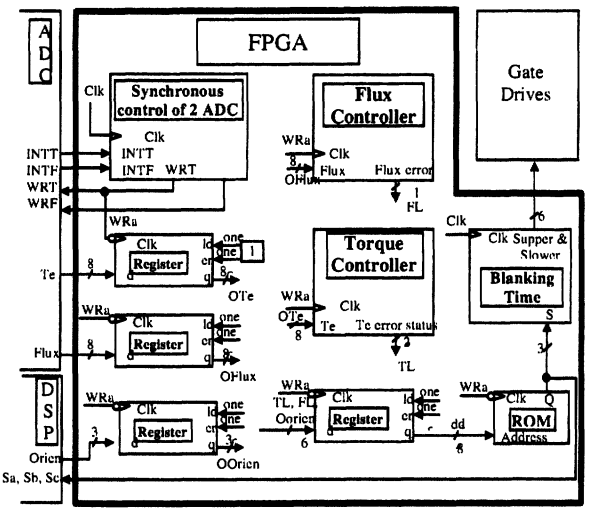


Figure 6. Functional block diagram of Altera FPGA design

TABLE II. Voltage Vector Selection Table

Stator Flux Error Status	Torque Error Status	Sector					
		I	II	III	IV	V	VI
0	1	110	010	011	001	101	100
	0	111	000	111	000	111	000
	-1	011	001	101	100	110	010
1	1	100	110	010	011	001	101
	0	000	111	000	111	000	111
	-1	001	101	100	110	010	011

With the torque reference set to 1.04 Hz ±0.6 Nm, it is clearly shown by Figure 9(a), 9(b), 10(a), and 10(b) that the torque ripple of the proposed controller is reduce drastically. The hysteresis controllers produce an inconsistent and large torque ripple due to the selection of reverse voltage vector. The reverse voltage vector is selected when the torque overshoot and touches the upper band. The torque reduction is faster when the reverse voltage vector is selected rather than zero voltage vector. Consequently, the possibility of torque undershoot is increased. The results also indicate that the dynamic torque response of the proposed controller is as good as the hysteresis-based controller. Figure 11 represents the upper and lower triangular carriers generated by FPGA with the compensated torque error signal, T_c (upper trace) and torque error status (lower trace) for the proposed torque controller. The proposed torque controller eliminates the selection of reverse voltage vector since the torque error status never switches to -1. In the experiment, the condition of the torque error status is set as follows:

$$Torque\ error\ status = \begin{cases} 01 & for\ T_e \geq C_{upper} \\ 00 & for\ C_{lower} < T_e < C_{upper} \\ 10 & for\ T_e \leq C_{lower} \end{cases} \quad (19)$$

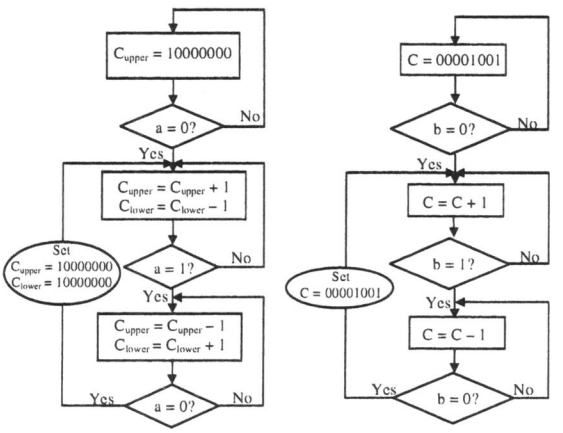


Figure 7. The flow chart of the triangular carrier waveforms generation (a) Torque controller, (b) Flux controller.

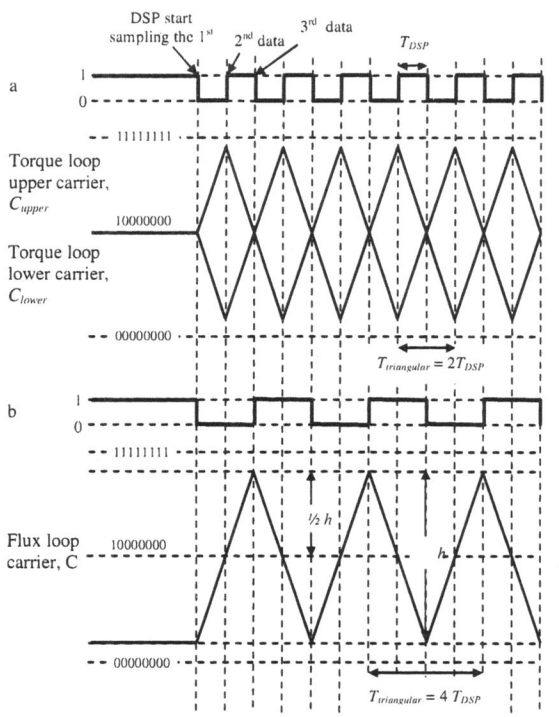


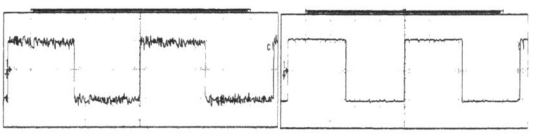
Figure 8. The generated triangular carriers synchronous with the DSP sampling time

TABLE III. PARAMETERS OF INDUCTION MACHINE

Stator resistance	10.9 Ω
Rotor resistance	9.5 Ω
Stator self inductance	0.859 H
Rotor self inductance	0.859 H
Mutual inductance	0.828 H
Rated speed	2880 rpm
Pole pair	2
DC link voltage	120 V
Rated flux	0.495 Wb

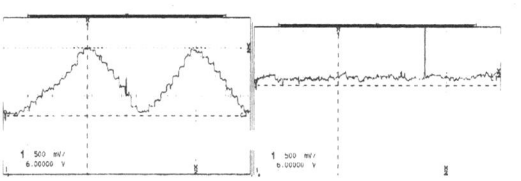
TABLE IV. PARAMETERS FOR THE PROPOSED TORQUE AND FLUX CONTROLLERS

Torque		Flux	
A_t	335	A_ψ	69.282
B_t	12	B_ψ	-69.282
K_t	6	K_ψ	11000
K_{tp}	180		
K_{ti}	60000		



(a) (b)

Figure 9. (a) Torque response for hysteresis-based controller, (b) Torque response for proposed controller, [0.6 Nm/div]



(a) (b)

Figure 10. (a) Zoom in torque response for hysteresis-based controller, (b) Zoom in torque response for proposed controller, [0.05 Nm/div]

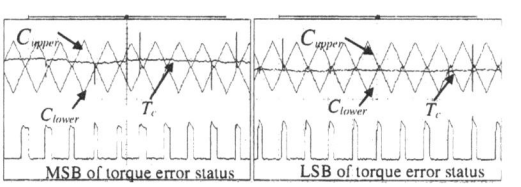
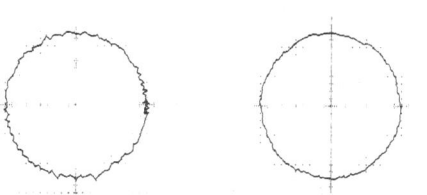


Figure 11. Proposed torque controller – The upper and lower triangular carriers with the compensated torque error signal, T_e (upper trace) and torque error status (lower trace), (a) MSB of torque error status, (b) LSB of torque error status.



(a) (b)

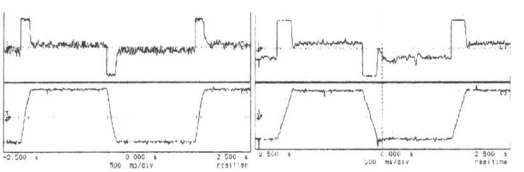
Figure 12. Steady state flux locus, (a) hysteresis-based controller, (b) proposed controller, [0.2 Wb/div]

Figure 12 shows the steady state flux locus. An almost circular locus with smaller ripple is obtained by implementing the proposed flux controller.

In order to examine the speed response of the proposed DTC, a simple PI speed controller is employed. Since a low resolution IE is used (200 p.p.r.), the speed is sampled at every 10ms (100 Hz) by the DSP. As a result, the speed controller gives a low response of speed as shown in Figure 13. The low speed response is reflected to the torque reference, which will cause the torque reference contains ripples. Consequently the torque response also may contain ripple during constant speed.

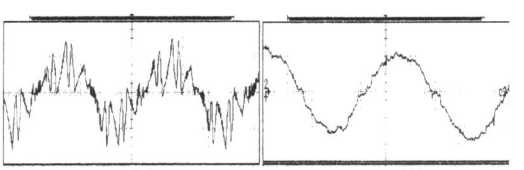
The steady state phase current is demonstrated by Figure 14. The hysteresis controllers produce high ripple current, which contains a lot of harmonic components. The proposed controllers have improved the phase current, as shown by the figure.

The frequency spectrum of the switching pattern, S_b is shown in Figure 15. The switching frequency of the hysteresis-based controller is unpredictable; therefore the harmonic components are widely distributed. Due to the constant switching frequency of the proposed controllers, the harmonic component is concentrated around the carrier frequency and its multiple. The first harmonic is located at 10.4 kHz, which is equivalent to the torque loop triangular carriers frequency - this is because torque switching is more dominant than the flux switching.



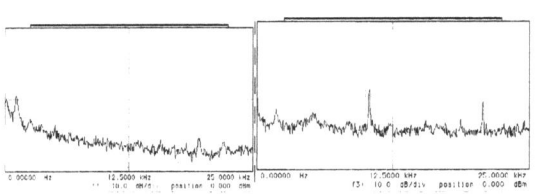
(a) (b)

Figure 13. Upper trace: torque response (1 Nm/div), Lower trace: rotor speed (9.33 rad s^{-1}/div) for (a) Hysteresis controllers, (b) Proposed controllers



(a) (b)

Figure 14. Steady state phase current (0.5714A/div) (a) Hysteresis controllers, (b) Proposed controllers



(a) (b)

Figure 15. Frequency spectrum of the switching pattern, S_b for (a) Hysteresis controllers, (b) Proposed controllers, [Horizontal: 2.5 kHz/div, vertical: 10 dB/div.]

V. CONCLUSIONS

This paper has presented the digital design and implementation of a new torque and flux controllers for DTC induction motor drive system. The implementation, which proved the feasibility of the proposed controllers, was accomplised using the combination of a DSP and an FPGA device. The experimental results showed that the switching frequency of the DTC drive was fixed at 10.4 kHz regardless of the operating speed. The torque and flux ripples were significantly reduced.

REFERENCES

[1] H. Le-Huy, "Microprocessors and digital IC's for motion control," Proc. IEEE, Vol. 82, No.8, pp. 1140-1163, 1994.

[2] Y.Y. Tzou and Hau-Jean Hsu, "FPGA Realization of Space Vector PWM Control IC for Three-Phase PWM Inverters", Trans. IEEE on Power Electronics, Vol.12, No. 6, pp. 953-963, 1997

[3] Y.A. Chapuis, C. Girerd, F. Aubepart, J.P. Blonde, F. Braun, "Quantization problem analysis on ASIC-Based Direct Torque Control of induction machine", Proceeding of the IEEE IECON'98, pp. 1527-1532, 1998.

[4] F. Aubepart, P. Poure, Y.A. Chapuis, C. Girerd and F. Braun, "Design and simulation of ASIC-based system control: application to Direct Torque Control of induction machine", Proceeding of the IEEE International Symposium on Industrial Electronics (ISIE'99), vol 3, pp. 1250-1255, 1999.

[5] F. Aubepart, P. Poure, and F. Braun, "VLSI design approach of complex motor control, Case of Direct Torque Control of AC machine" 7th IEEE International Conference on Electronics, Circuits and Systems, ICECS 2000, Vol. 2, pp. 814-817, 2000.

[6] C. L. Toh, N. R. N. Idris and A. H. M. Yatim, "New torque and flux controllers for Direct Torque Control of induction machines", The 5th International Conference on Power Electronics and Drive Systems 2003 (PEDS 2003), Singapore, Vol. 1, pp. 216-221.

[7] I. Takahashi, T. Noguchi, "A new quick-response and high efficiency control strategy of an induction motor", IEEE Trans. Ind. Appl., Vol. IA-22, No 5, pp. 820-827, 1986.

[8] W. C. Lo, C. C. Chan, Z. Q. Zhu, L. Xu, D. Howe, K. T. Chau, "Acoustic Noise Radiated by PWM-Controlled Induction Machine Drives", IEEE Trans. On Ind. Electronic, Vol. 47, No. 4, pp. 880-889, 2000.

[9] K.D.Hurst, T.G. Habetler, G. Griva, F. Profumo, "Zero speed tacholess IM torque control: Simply a matter of stator voltage integration", IEEE Trans Ind Appl., Vol 34, No. 4, pp. 790-795.

[10] Altera Corporation, (1999), "University Program Design Laboratory Package", User's guide.