Journal of Theoretical and Applied Information Technology
10th July 2014. Vol. 65 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EFFICIENT FPGA-BASED INVERSE PARK TRANSFORMATION OF PMSM MOTOR USING CORDIC ALGORITHM

[1] ANIS SHAHIDA MOKHTAR, [2] MAMUN BIN IBNE REAZ, [2]MOHD ALAUDDIN MOHD ALI

[1] Department of Electric and Electronics. Faculty of Engineering, National Defence University of Malaysia, Sungai Besi Camp, 57000 Kuala Lumpur, Malaysia

[2] Department of Electrical, Electronic and Systems Engineering, Universiti Kebangsaan Malaysia, Bangi, 43600 UKM, Selangor, Malaysia

E-mail: [1] adniksda2@yahoo.com, [2]mamun.reaz@gmail.com

ABSTRACT

A real-time field programmable gate array (FPGA) debugging system design is presented in this work. Inverse Park Transformation, which is an important module in Field Oriented Control (FOC) focuses on fast computation with good accuracy. The integrated system designed in this work provides a precise and highly accurate performance. It contributes to optimal efficiency and stability of permanent magnet synchronous motor (PMSM). The result shows that the performance of the proposed system is 142 ns execution time with 20 MHz frequency and 0.04% of error. This result shows significant improvement compared to other recent similar researches.

Keywords: *CORDIC, FOC, Inverse Park transformation, PLL, PMSM*

1. INTRODUCTION

In recent years, permanent magnet synchronous motors (PMSM) had experienced rapid growth and had been increasingly applied in many automation control field mainly due to the advantages of better power density, high performance motor control with fast speed and better accuracy. The vector control of the motor, which is known as Field Oriented Control (FOC) of the PMSM drive, is employed to achieve this high performance. The FOC is represented by a vector used to control the stator currents, based on projections to transform three-phase time and speed dependent system to two-phase in time variant. FOC relies on two algorithms, Clarke Transform and Park Transform. The Clarke algorithm converts the stator-winding phase currents from a three-axis vector to a two-axis vector. The Park transform indeed converts the two-axis currents into a rotating system. This transformation requires a complex and intensive computation. Therefore, a prevailing method to calculate these transformations is needed to ensure the performance of the motor is optimized.

CORDIC (Coordinate Rotation Digital Computer) is a significant technique used to achieve such transformation. CORDIC was introduced by Volder [1] and later generalized by Walther [2]. It is an iterative algorithm used to calculate a large class of mathematical functions using repetition of addition, shifting and small look-up table operations. Due to its simplicity of implicated operations, CORDIC suits very well for realization FPGA (Field-programmable gate array). Considerable research was carried-out on the development of FPGA-based adaptive control strategies for motor control and other applications [3]. FPGA technology allows development of architecture specific equipment in a flexible programmable environment. Currently, due to the advantages of high computation power and parallel processing, FPGA brought more attention and became one of the best resolutions to accomplish complex algorithm. Therefore, a lot of practical application in motorized control adopting FPGA was studied [4]–[8].

The architecture of whole PMSM motor is shown in fig 1. FOC modules consist of Park Transformation, Clarke Transformation and their inverses. The Inverse Park transform modifies the flux, torque (d, q) rotating frame into a two phase orthogonal system.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The Inverse Park Transform uses 2 main equations: [9]

$$V_\alpha = V_d \cos\theta - V_q \sin\theta \quad (1)$$

$$V_\beta = V_d \sin\theta + V_q \cos\theta \quad (2)$$

Where $\mathbf{V_{alpha}}$ is the alpha voltage, $\mathbf{V_d}$ is the direct voltage, $\mathbf{V_q}$ is the quad voltage, $\mathbf{V_{beta}}$ is the beta voltage, and $\boldsymbol{\theta}$ is the angle of rotation for the transformed reference frame.

The value of cos θ and sine θ from the equation need to be evaluated prior to CORDIC utilization to realize this transformation. Due to the simplicity of the hardware structure, speed and resources, CORDIC is selected to solve FOC task [10]. The aim of this research is to comprehend the inverse Park Transformation in a minimal cycle so that the motor speed can be increased with steady precision.

Ghariani et. al used a modified CORDIC to compute the Park transformation with a frequency of 8 Mhz [11]. The total operation time for the CORDIC algorithm alone was 125 ns. Research group, Rachid et. al. used two different frequencies to perform their operations,100MHz and 8MHz. Both of frequencies need *6* clock cycles to produce the output [12]. Furthermore, rotor flux estimator were used to calculate *cos θ* and *sine θ*. The calculated angle was used in both Park transformation and inverse Park transformation. Ying-Shieh Kung et. al. accomplished an inverse Park sub-module in 5 steps [13] using Finite state machine (FSM) method. FSM interpret each step equals to 1 clock cycle. Each clock cycles is equal to 40 ns at 25 MHz. Thus, the execution time for whole inverse Park transformation is 0.2 µs. However, the values of cos θ and sin θ were pre-calculated in the look-up table. Ricardo de Castro et. al. reported that the combination of Park transformation and Clarke transformation produces 30 clock cycles with a maximum frequency of 78MHz [14]. This CORDIC implementation takes 20 clock cycles to complete. A summary of the frequency, execution time and clock cycles of the research discussed tabulated in Table 1.

Table 1: Comparison With Other Works

	Frequency	Execution time	Clock cycle
Ghariani et. al.[11]	8 MHz		-
Rachid. et. al.[12]	100MHz and 8MHz	-	6
Ying-Shieh Kung et. al. [13]	25Mhz	0.2µs	5
Ricardo de Castro et.al. [14]	78MHz		30

2. METHODOLOGY

CORDIC is a simple algorithm to compute trigonometric, linear, hyperbolic and logarithmic functions. CORDIC has two (2) operational modes: rotation and vectoring where rotation mode rotates the input vector by a specific angle, which has been used in this research. The standard CORDIC algorithm can be describe as

$$x_{i+1} = x_i.\cos\theta - y_i.\sin\theta = \cos\theta_i.(x_i - y_i.\tan\theta_i) \quad (3)$$

$$y_{i+1} = x_i.\sin\theta + y_i.\cos\theta = \cos\theta_i.(y_i + x_i.\tan\theta_i) \quad (4)$$

Where x_{i+1} and y_{i+1} are the final vector and θ is the target angle of rotation. To obtain the rotated vector, the angle of rotation θ is decomposed into a sequence of fixed predefined elementary rotations with variable direction which known as a micro-rotation. These micro rotations, which are on a sequence of elementary angles α_i, can be expressed as follows:

$$\theta = \sum_{i=0}^{b} \mu_i \alpha_i \quad (5)$$

$$\alpha_i = \tan^{-1}\left(2^{-i}\right), \text{ i=0, 1, 2… n-1} \quad (6)$$

Where $\mu_i \in \{1,-1\}$ decides the direction of vector rotation for i^{th} iteration. n is the number of iterations. The overall scaling-factor of CORDIC iterations is given by (5). The scaling factor K is a constant value, K = 0.60725.

$$K = \prod_{i=0}^{n-1} K_i = \prod_{i=0}^{n-1} \frac{1}{\sqrt{1+2^{-2i}}} \quad (7)$$

The rotation mode of CORDIC algorithm has three inputs that are initialized to the coordinate components of the vector x_i, y_i and the angle of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

rotation $z_i=\theta$ and it is described by the following iteration equations

$$x_{i+1} = x_i - y_i \mu_i 2^{-i} \quad (8)$$

$$y_{i+1} = y_i + x_i \mu_i 2^{-i} \quad (9)$$

$$z_{i+1} = z_i - \mu_i \tan^{-i} 2^{-i} \quad (10)$$

Where $\mu_i=-1$ if $z_i<0$, $+1$ otherwise. The complete system architecture is shown in Fig. 2, with the aims to completely transform two phase rotating frame (d, q) into two phase stationary frame (α, β) with a minimum clock cycle and execution time.

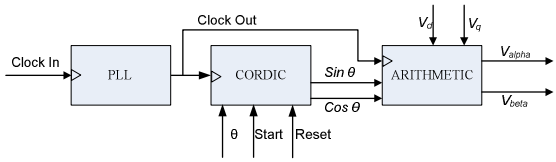

Fig. 1. Architecture of whole system

Fig 2 shows the architecture of inverse Park transformation, which consists of PLL, CORDIC and ARITHMETIC module. This architecture is used to compute equation (1) and (2). CORDIC module is used to calculate *sine* and *cos* of the rotated angle. This is the most important part in order to get an accurate result as it affect the rest of the system performance. The ARITHMETIC module that comes after the CORDIC module is used to complete the transformation. It consists of adder, subtractor, multiplexer and shifter. This design has been simulated in [15], but in [15] the research is done until gate level simulation. In this research, it is further improved by the real-time hardware implementation using logic analyzer. In the previous experiment [15], the frequency used was 24 MHz with 160 ns of execution time while in this research the maximum frequency was 20 MHz with 142 ns of execution time.

.

3. RESULTS AND DISCUSSION

The architecture is realized by using Verilog hardware description language (HDL) in Quartus II and the simulation is done using ModelSim simulator. Altera Cyclone® II 2C35 FPGA is used as an FPGA platform and for the real-time implementation was facilitated by logic analyzer and pattern generator. The simulation is done for the entire coordinate system. The input angle θ is represented in radian and it ranges for entire coordinate space. Six set of inputs are tested throughout the process as in **Error! Reference source not found.**. The numerical calculation of this table is done based on equation (1) and (2). For a realization in digital environment, fixed point is used for implementation

The overall design requires 48 bits of input data and 64 bits of output data. However, the LA has limited IO ports. In order to validate the proposed design in FPGA, value of V_q is fixed as 0.5 V in pattern generator (PG), so that results can be tested by using the LA. Altera Cyclone® II EP2C20F484C7N FPGA offers 2 GPIO (General Purpose Input Output) with 40 pins each. Values of V_d randomly selected in certain range based on the fixed point format used and value of θ varies for whole coordinate space. V_d and θ consume 16 bits each while output V_α consumes 32 bits. The result in table 2, shows that the highest percent error between simulated and calculation values are 0.04%. The motor accuracy and precision can be assured by the proposed result.

The simulation values obtained from Fig. 3 for data number 1 and 2, Fig. 4 for data number 3 and 4, and Fig. 5 for data number 5 and 6. The system operates on 20 MHz frequency. Logic analyzer clock signal generates the *outclock* signal, whereas *outreset* signal is a signal used to reset the system and *outstart* signal indicates the start of the operations. The calculations begin on the first rising edge of clock signal after *outstart* signal triggered. As shown in fig. 3, 4 and 5, it shows that the maximum execution time of the system is 142 ns and the output is obtained within 3 clocks cycles which is certainly gives better performances compared to other similar research works in this era

4. CONCLUSION

FOC in PMSM motor needs high efficient performance in order to be fully utilized. The proposed inverse Park Transformation presents an excellence results in term of execution time and delay where the execution time of the system takes about 142 ns with error of 0.04% and has fast as 3 clock cycles. This small execution time and fast output will play an important role in achieving optimal efficiency and stability of the PMSM motor drive.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REFRENCES:

[1] J.E. Volder, "The Cordic trigonometric Computing Technique", *IRE Transaction Electronics Computer,* vol. EC-8, pp. 330-334, 1959.

[2] J. S. Walther, "A Unified Algorithm for Elementary Functions," *Proc. AFIPS Spring Joint Computer Conf.*, vol. 38, pp. 379-385, 1971.

[3] Leena Vachhani,K. Sridharan, and Pramid Kumar Meher, "Efficient FPGA Realization of CORDIC With Application to Robotic Exploration", *IEEE Trans. Industrial Electronics*, vol. 56(12),pp. 4915–4929.

[4] B. Bossoufi, M. Karim and A. Lagrioui, "FPGA-based implementation with simulation of structue Direct Torque Control of a PMSM," in *Int. Conf. Multimedia Computing and Systems*, 2011 pp. 1-6.

[5] A. Sathyan, N. Milivojevic, L. Young-Joo, M.Krishnamurthy and A. Emadi, "An FPGA-Based Novel Digital PWM Control Scheme for BLDC Motor Drives," *IEEE Trans. Industrial Electronics*, vol. 56, pp. 3040-3049, 2009.

[6] K. Ying-Sheih and T. Ming-Hung, "FPGA-Based Speed Control IC for PMSM Drive with Adaptive Fuzzy Control", *IEEE Trans. Power Electronics,* vol. 22, pp. 11, 2007.

[7] L. Idkhajine, E. Monmasson, M. W. Naouar, A. Prata and K. Bouallaga, "Fully Integrated FPGA-Based Controller for Synchronous Motor Drive," *IEEE Trans. Industrial Electronics,* vol. 56, pp. 4006-4017, 2009.

[8] Z. Da and L. Hui, "A Stochastic-Based FPGA Controller for an Induction Motor Drive with Integrated Neural Network Algorithms," *IEEE Trans. Industrial Electronic*, vol. 55, pp. 551-561, 2008.

[9] H. A. T. Steven Campbell, "Clarke's and park's transformations," in *DSP-Based Electromechanical Motion Control*, CRC press, 2004, pp. 209–222.

[10] P. K. Meher, J.Valls, J. Tso-Bing, K.Sridharan and Maharatna, "50 Years of CORDIC: Algorithms, Architectures, and Applications," *IEEE Trans. Circuits and Systems I: Regular Papers*, vol. 56, pp. 1893-1907, 2009.

[11] R. Beguenane, J. Mailloux, S. Simard and A. Tisserand "Towards the System-on-Chip Realization of a Sensorless Vector Controller with Microsecond-order Computation Time", *Canadian Conference on Electrical and Computer Engineering,* 2006, 1073–1077.

[12] R. De Castro, R. E. Araujo and H. Oliveira, "Control in Multi-Motor Electric Vehicle with a FPGA platform", in *IEEE International Symposium on Industrial Embedded Systems,* Lausanne, 2009, pp. 219–227.

[13] M. Ghariani, N. Masmoudi, M.W. Kharrat and L. Kamoun," Design and chip implementation of modified CORDIC algorithm for Sine and Cosine functions application: PARK transformation", in *Proceedings of the Tenth International Conference Microelectronics*, 1998. pp. 241–244.

[14] K. Ying-Shieh and T. Ming-Hung, " FPGA-Based Speed Control IC for PMSM Drive With Adaptive Fuzzy Control", *IEEE Transactions on Power Electronics*, vol. 22, pp. 2476–2486.

[15] A.S.N. Mokhtar, M.B.I. Reaz, Mohd Marufuzzaman, M.A.M.Ali, "Hardware Implementation of A High Speed Inverse Park Transformation Using CORDIC and PLL for FOC Brushless Servo Drive," *Eelectronika ir Elektrotechnika (Electronics and Electrical Engineering),* No 3. pp. 23–26, 2013

.

Journal of Theoretical and Applied Information Technology
10th July 2014. Vol. 65 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

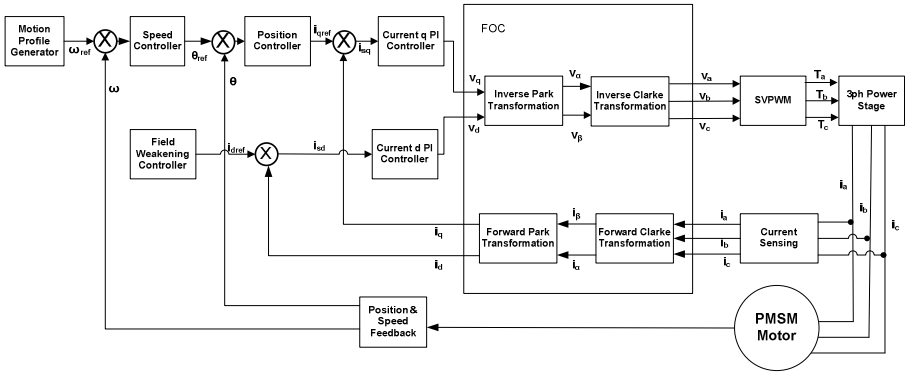

Fig. 2. Architecture of PMSM Motor

Table 2:Simulation results

No	V_d	V_q	θ	V_{beta}	V_{alpha}	V_{alpha} (Calculated value)	V_{alpha} (Simulated value)	% error
1	-0.55	0.5	200	-0.28174	0.687841	7018BA2 H	701C857 H	0.0132
2	-0.7	0.5	220	0.066929	0.857625	8BC41C8 H	8BC0CA4 H	0.0093
3	0.25	0.5	260	-0.33303	0.448992	492BF07 H	4935000 H	0.0484
4	0.5	0.5	300	-0.18301	0.683013	6F4F4A3 H	6F4E000 H	0.0045
5	0.62	0.5	320	-0.01551	0.796341	81C75C8 H	81C6158 H	0.0038
6	0.91	0.5	345	0.247438	1.008402	A456884 H	A4475BB H	0.0361

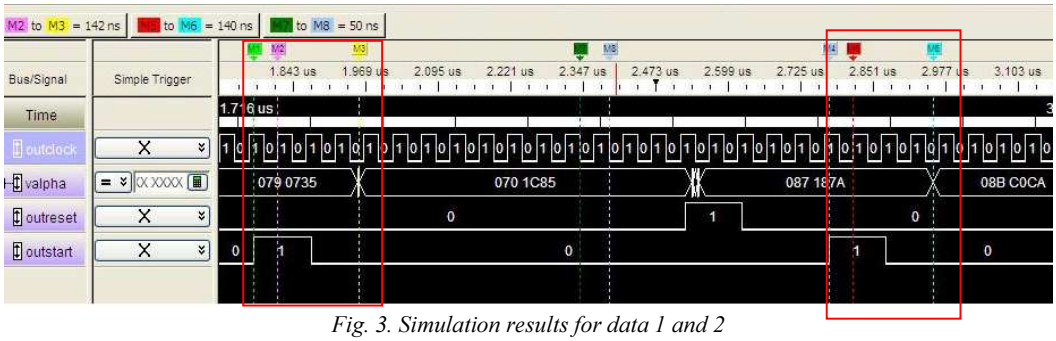

Fig. 3. Simulation results for data 1 and 2

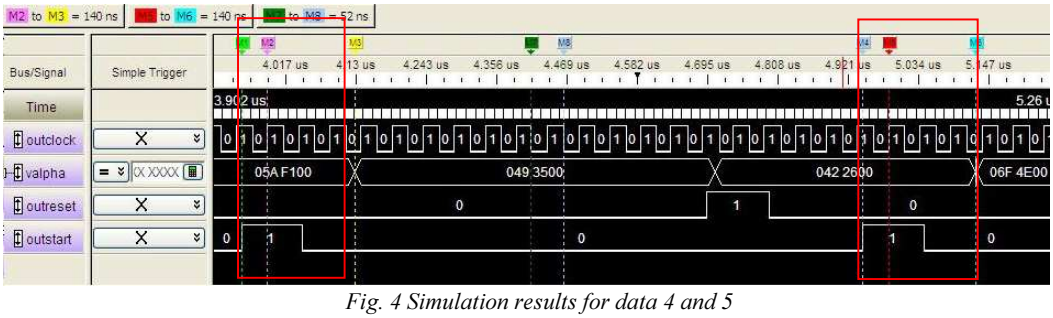

Fig. 4 Simulation results for data 4 and 5

Journal of Theoretical and Applied Information Technology
10th July 2014. Vol. 65 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

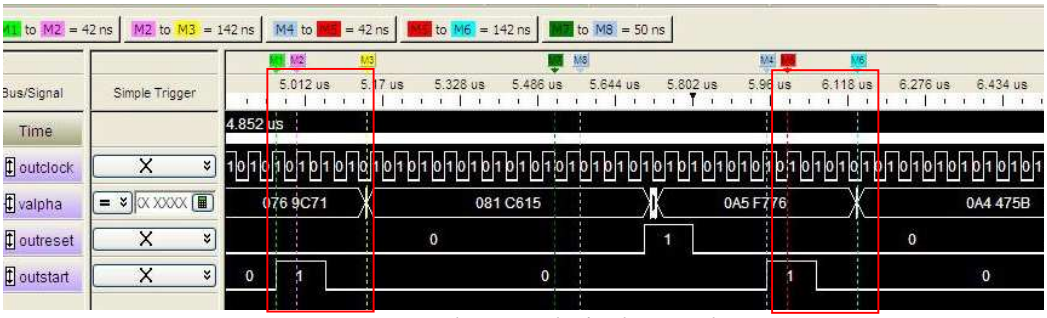

Fig. 5 Simulation results for data 5 and 6