Professional paper

Informacije MIDEM
Journal of Microelectronics,
Electronic Components and Materials
Vol. 44, No. 2 (2014), 168 – 173

FPGA-based EtherCAT Microcontroller circuit design of SPI communication for real-time systems

Mingxin Hou[1], Li Jiang[1], Fenglei Ni[1], Minghe Jin[1], Hong Liu[1], Zhaopeng Chen[2]

[1]School of Technology State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin, China
[2]Institute of Robotics and Mechatronics, German Aerospace Center, Munich, Germany

Abstract: In this study a novel Microcontroller unit (MCU) circuit based on FPGA for EtherCAT system is presented. The resource utilization statistics of the MCU circuit are provided and the performance of the MCU circuit is analyzed. The first objective is to understand the feasibility of the approach, i.e., whether it is possible for the MCU to drive the EtherCAT slave interface, be reasonable for real-time performance and be stable at different communication SPI frequencies. The second objective is to give the MCU circuit developers some valuable guidelines. Furthermore, it is verified that the proposed MCU circuit based on FPGA is stable, reliable and real-time tested using the online debugging tool SignalTap II.

Keywords: MCU circuit, EtherCAT, SPI, FPGA, online real-time system

FPGA EtherCAT mikrokontroler SPI komunikacij za isteme v realnem času

Izvleček: V študiji je predstavljen novo vezje mikrokontrolne eneote (MCU) na osnovi FPGA za EtherCAT sisteme. Predstavljena je statistika koriščenja MCU vezij ter analiza učinkovitosti MCU vezij. Prva naloga je razumevanje zmožnosti pristopa, npr. ali lahko MCU krmili podrejen EtherCAT vmesnik, je sprejemljiv za sisteme v realnem času, ali je stabilen za različne frekvence SPI komunikacije. Druga namen je ponuditi razvijalcem MCU vezij pomembna vodila. Dokazano je, da je predlagan FPGA mikrokontroler stabilen, zanesljiv in testiran v realnem času z online SignalTap II razhroščevalnim orodjem.

Ključne besede: MCU vezhe, EtherCAT, SPI, FPGA, online sistem v realnem času

** Corresponding Author's e-mail: jiangli01@hit.edu.cn*

1 Introduction

Ethernet for Control Automation Technology (EtherCAT) is an open real-time Ethernet network, which is typical for a transfer of 125 byte over 100 Mbit/s Ethernet [1]. The EtherCAT has many practical applications, which includes industrial robots [2], precision motion control [3-5], real-time networks communication [6-9], etc. As pointed out in [10], the EtherCAT circuit coupler accuracy is 23 ns. This may be associated to the concept of MCU slave circuit delay defined by M. Sung [11], where to be essentially used for the design of EtherCAT communication-based circuit are crucial [12-14].

However, presently most of the research has focused only on the EtherCAT master or slave applications in networks [15, 16]. Much remains to be done to understand and exploit the EtherCAT system design using MCU. For instance, to enable the compliance with real-time requirements as well as to provide data transparent security, the actual communication delay is being considered as a practical problem. If there is no estimated precisely the circuit power consumption, then the suitable power supply for the EtherCAT MCU circuit will be unacceptable in a real situation.

To present the performance and power optimization of the SPI communication used for EtherCAT system, as shown in Fig.1, the EtherCAT system structure is firstly analyzed. The typical process of establishing a communication is initiated by the EtherCAT master by sending a broadcast to the EtherCAT slave, which relies on Ethernet wire. The SPI master block and SPI slave block, which are integrated into the MCU circuit and EtherCAT Slave, take care of the communication as a crucial interface between the field bus(MCU users apply) and the EtherCAT slave application. Furthermore, the power consumption of the EtherCAT user circuit design will be determined by the SPI_CLOCK signal. Therefore, the optimization of the SPI communication will play a critical role in the whole real-time EtherCAT system.

The Field Programmable Gate Array (FPGA) is a programmable digital logic device by software, which has advantages include the ability to re-program in the field to fix bugs, and may include a shorter time to market and lower non-recurring engineering costs [17-20]. To design and evaluate the EtherCAT slave MCU circuit effectively, we choose Altera FPGA (EP3C25F256) as a develop platform, for it is especially useful for complex EtherCAT protocol algorithm [21-24]. High-precision and real-time synchronous operation is important for EtherCAT MCU circuit design. With an EtherCAT-based real-time system, the development of such synchronized operations relies on enough Static Random Access Memory (SRAM). Currently there are various types of SRAM-based schemes in practice [25-27]. We chose SRAM CY7C1380D for the MCU circuit design because it has desirable features for realizing enough memory cells with advanced synchronous peripheral circuitry and a two-bit counter for internal burst operation.

The objective of this research is to provide a novel MCU structure circuit based on FPGA and give some guidelines to the developers who are actively engaged in designing the EtherCAT circuit. This paper will be arranged as follows: Section 2 describes the architecture of the MCU design, and section 3 evaluates the MCU performance. Section 4 presents the real-time characteristics through online experiments. Finally, a conclusion of this paper is addressed in section 5.

2 MCU circuit for EtherCAT

The real-time EtherCAT system is composed of three basic blocks: EtherCAT master block, EtherCAT slave block and MCU circuit [28] (see Fig.1). Moreover, two basic design principles of the MCU circuit are how to communicate with the EtherCAT slave by SPI blocks and what's the MCU circuit Electrical characteristic. Fig. 2 (a) shows a MCU circuit architecture that achieves

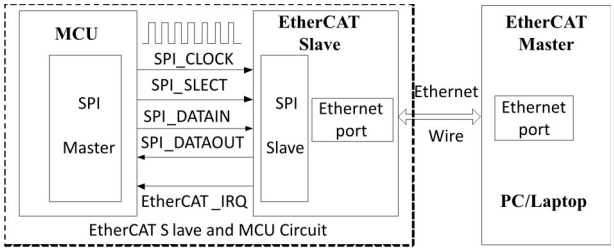


Figure 1: EtherCAT system structure

real-time communication of a cable-driven EtherCAT when opens and runs. The MCU block will be signaled by the EtherCAT AL Event Register using the Process Data Interface (PDI) Interrupt Request signal (SPI_IRQ). For IRQ generation, the AL Event Request register (0x0220:0x0223) is combined with the AL Event Mask register (0x0204:0x0207) using a logical AND operation, then all resulting bits are combined (logical OR) into one interrupt signal. The FPGA PIO core is configured to input ports for capturing the IRQ signal from the EtherCAT slave interface. Whenever the MCU synchronously detects a falling-edge from the IRQ signal, an internal MCU interrupt request should be generated. The SPI logic block of the MCU is synchronous to the clock input provided by SOPC PLL. Meanwhile, the Nios II CPU also shares the same clock with the SPI core. Because the MCU has been configured as a SPI master, the Avalon-MM clock is divided to generate the SPI_Clock. The JTAG UART circuitry is built into the MCU. Therefore, the hosts PC can connect to the MCU using a JTAG download cable by a USB-Blaster. A Tristate Bridge and CY7C1380D SRAM are created into the MCU circuit for large external volatile memory. The CY7C1380D SRAM integrates 524,288 × 36 and 1,048,576 × 18 SRAM cells with advanced synchronous peripheral circuitry and a two-bit counter for the internal burst operation. 2 M physical memory sizes will make sure that the MCU has enough memory to run the EtherCAT protocol program adequately, reliably and efficiently. The MCU based on Nios II system can use the EPCS device controller to store program code, keep non-volatile program data and manage the MCU configuration data.

To simultaneously achieve good speed and circuit performances, we connect 5 signals to the EtherCAT slave interface: the SPI_Clock, SPI_Slect, SPI_Data_In, SPI_Data_OUT and Interrupt Signal for IRQ generated. From a signal integrity perspective, to ensure the consistency and stability of the SPI data transmission, all the five signal lines should have the same length when laying out the PCB as shown in Fig.2 (b). The architecture of the MCU based on Altera Cyclone III is shown in Fig.2(c), and offers an unprecedented combination of low power, high functionality, and low cost. The architecture consists of up to 120 K vertically arranged logic elements (LEs) and 200 18x18 embedded multipliers.

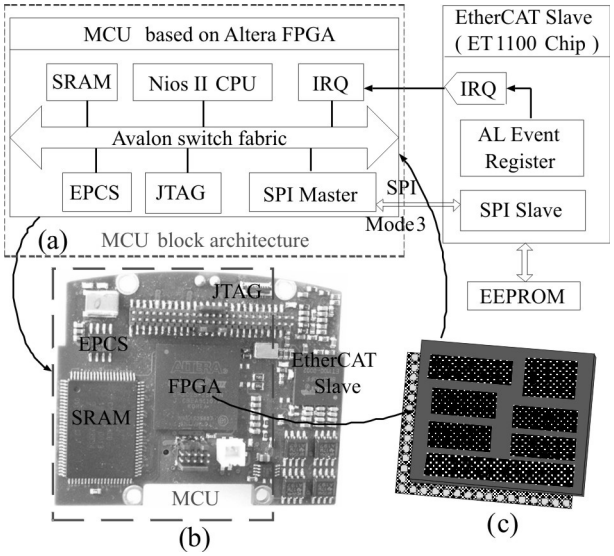


Figure 2: MCU circuit Block Diagram

3 Flow summary and timing specifications

3.1 Flow summary

The MCU circuit based on the Altera Cyclone® III FPGA has been compiled by the Quartus II software platform and these characteristic are reported in Table 1.

Table 1: Accurate MCU based on FPGA compilation resource report

SPI Clock	TR	TMB	DLR	TCF	TLE
1 MHz	2,389	57,600	2,264	2,954	3,542
5 MHz	2,387	57,600	2,262	2,954	3,541
10 MHz	2,386	57,600	2,261	2,958	3,547
15 MHz	2,385	57,600	2,260	2.959	3,548
20 MHz	2,385	57,600	2,260	2,954	3,530
25 MHz	2,384	57,600	2,259	2,939	3,524
30 MHz	2,384	57,600	2,259	2,943	3,535

First, we chose a 1 MHz SPI clock driven by the MCU master to the EtherCAT slave, used it to synchronize the data bit. Total registers (TR) shows a total of 2,389 registers were used. Total memory bits (TMB) illustrate that a total of 57,600 memory bits were used. Total logic elements (TLE) show that a total of 3,542 logic elements were used. Dedicated Logic Registers (DLR) and Total Combinational Functions (TCF) indicated that 2,264 dedicated logic registers and a total of 2,954 combinational functions were used, respectively. Similarly, we get the other SPI clock information which has also been compiled by Quartus II under different rates of frequencies.

Secondly, Table I shows the measured throughput comparison. At 1 MHz the communication between the MCU and EtherCAT slave, the Total Registers column decrease slowly from 2,389 to 2,384 registers. The Dedicated Logic Registers column shows that the data gradually reduces as the SPI clock frequency increases. By contrast, there are no evident consistent tendencies in the columns of the Total Combinational Functions and Total Logic Elements. It is interesting to note that there is no change in the Total Memory Bits no matter the difference in the SPI clock.

3.2 Timing specifications

The MCU circuit is consisted mainly of SPI block, Nios CPU block, external memory block (SRAM), flash block (EPCS), etc (see Fig.2). In order to realize the data communication between the MCU circuit and the EtherCAT Slave chip (ET1100), a reasonable SPI connection mode must be selected synchronously and consistently between the master and the slave side. So, links are established in this case and EtherCAT data communication begins from the master side (MCU circuit) access to the EtherCAT slave registers (EtherCAT slave chip ET1100) by SPI protocol mode 3.

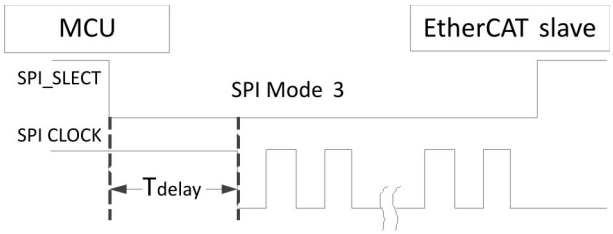


Figure 3: MCU communication time delay

However, the problem of the real-time performance of the MCU circuit is affected by SPI time delay in the case of Fig.3, it is T_{delay}. The rectangles of the MCU and the EtherCAT slave in Fig.3 also correspond to MCU block and EtherCAT slave block in Fig.1. The MCU delay generation logic uses a granularity of half the period of the SPI clock. And the actual delay achieved does not take place during the same time. Instead, it is actually rounded up to the nearest multiple of the falling edge clock, as shown in equation (1) and equation (2).

$$T_{delay} \geq \frac{1}{2}(\text{Period of SPI clock}) \quad (1)$$

$$T_{delay} \leq (\text{Period of SPI clock}) \quad (2)$$

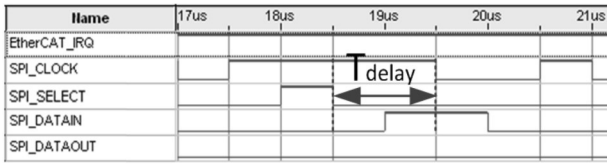
For the SPI master/slave pair must use the same mode to communicate and SPI Mode 3 is selected as the communication between the MCU and the EtherCAT.

Therefore, we setup the Clock Polarity equal to 1, and the Clock Phase equal to 1.

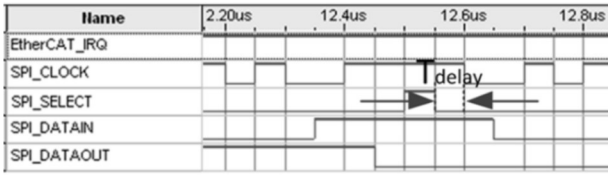
4 MCU circuit for EtherCAT

4.1 Online real-time waveforms

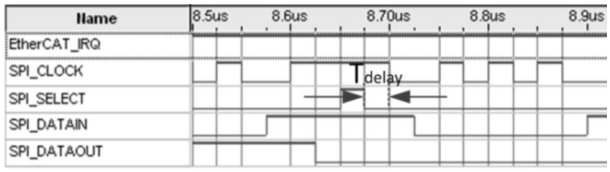
The MCU with online signals generated by a system-on-a-programmable-chip (SOPC) Builder, which is monitored by a SignalTap II, have been shown in Fig.4. By using a SignalTap II Embedded Logic Analyzer (ELA) in the MCU circuit system, we can observe the online behavior of this hardware circuit in real-time operating waveforms more practical than the Quartus II or ModelSim simulation.



(a) 1 MHz MCU circuit SPI clock



(b) 10 MHz MCU circuit SPI clock



(c) 20 MHz MCU circuit SPI clock

Figure 4: MCU circuit online real-time signals by SignalTap II

The MCU transmission clock directly influences the MCU circuit communication speed. Three different frequencies of the SPI clock are observed at 1 MHz, 10 MHz and 20 MHz, respectively, where the five signals are connected to the EtherCAT slave interface: EtherCAT_IRQ, SPI_CLOCK, SPI_SELECT, SPI_DATAIN, and SPI_DATAOUT. The interrupt signal EtherCAT_IRQ is generated by the EtherCAT AL Event register dedicated to the MCU circuit, and typically has low signal polarity. The MCU can synchronously capture the falling edge while an internal interrupt requirement will be generated. The MCU master circuit starts the EtherCAT SPI access by asserting the SPI_SELECT signal and terminates it by taking back the SPI_SELECT, and generally has low signal polarity.

During the communication we tested its online performance. Because the EtherCAT SPI slave device needs additional time for initialization in real situations, the actually maximum time delay (T_{delay}) is about 1 clock cycle. By contrast, the minimum T_{delay} is about half of the clock period. This has been analyzed in section 3.2 and the actual results are exactly the same.

4.2 Electrical characteristics

To maintain the highest possible performance and reliability of the MCU circuit, we must consider the power consumption in a real situation. As depicted in Fig.5, the power consumption of the MCU (µController) circuit grows sharply from 610 mA to 637 mA with the communication SPI clock increases from 1 MHz to 20MHz. It is worth noting that the changing MCU current value is independent of power supply (3.3 V provided). Even so, it only depends on the communication frequency, which is determined by the SPI clock.

Therefore, when starting a new MCU circuit design, the developer could try to follow the two guidelines:

- Choose a suitable DC to DC voltage converter chip, for the absolute maximum output current must be provided.
- Select a satisfactory battery supply, for the power consumption of the MCU circuit clearly rises as the communication clock increase, as shown in Fig.5.

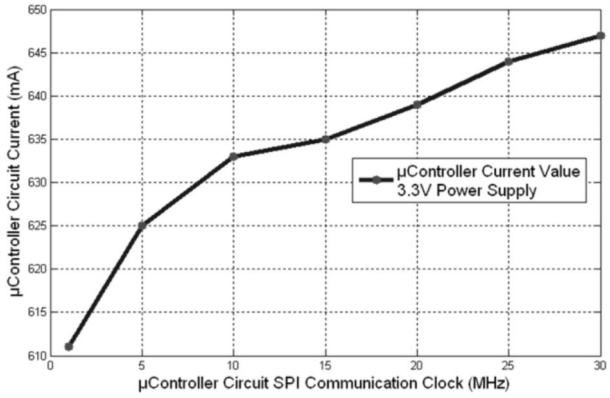


Figure 5: Electricity consumption

5 Conclusions

In this study, our primary objective is to propose the architecture of the MCU circuit block based on FPGA for EtherCAT that are applied in a real-time system. The proposed MCU circuit integrates CPU, SRAM and Flash into the structure in order to investigate its whole behavior under different communication frequencies (1

MHz, 10 MHz and 20 MHz). In particular, the power consumption of the MCU circuit has been tested, which revealed the maximum output current and the tendency for MCU current under corresponding communication frequencies. Furthermore, the technical benefits and practical operation are as follows:

1) MCU circuit block: The EtherCAT MCU circuit can be described as several system blocks, which could be easily found by researchers,
2) Transmission speed: The MCU interface communication speed will be determined by SPI block frequencies. So, the developers could choose the proper SPI clock to satisfy their requirements. However, the maximum communication speed for the EtherCAT MCU is 20 M/S,
3) Power consumption: The procedure allows designers to choose suitable power supply chips conveniently and estimate the battery supply time easily.

The online real-time signals have been captured and displayed by SignalTap II. Meanwhile, the performance of the MCU circuit has been discussed. Furthermore, it was verified that the merits of the MCU circuit performs in real-time and is stabile for EtherCAT technology by online observed experiments under different communication frequencies.

6 Acknowledgments

This work is supported by National Program on Key Basic Research Project of China (973 Program). No.2011CB013306

The authors would thank the reviewers and the editors for their careful review and very helpful comments.

7 References

1. Marco Cereia, Ivan Cibrario Bertolotti, Stefano Scanzio, "Performance of a Real-Time EtherCAT Master Under Linux", IEEE Transactions on Industrial Informatics, vol.7, no. 4, pp. 679-687, 2011.
2. You Wei, Kong Minxiu, Sun Lining, Diao Yanbin, "Control system design for heavy duty industrial robot", Industrial Robot-An International Journal, vol.39, no.4, pp. 365-380, 2012.
3. Kim Kanghee, Sung Minyoung, Jin Hyun-Wook, "Design and Implementation of a Delay-Guaranteed Motor Drive for Precision Motion Control", IEEE Transactions on Industrial Informatics, vol.8, no. 2, pp.351-365, 2012.
4. Vitturi Stefano, Peretti Luca, Seno Lucia, Zigliotto Mauro and Zunino Claudio, "Real-time Ethernet networks for motion control", Computer Standard and Interfaces, vol.33, no.5, pp.465-476, 2011.
5. Kim Jung-Hoon, Lim Sun, Jung Il-Kyun, "EtherCAT based parallel robot control system",Advances in Intelligent Systems and Computing, vol.208, pp.375-382, 2013.
6. Brugger Florian, Kreiner Christian, Thurner Thomas, "Runtime Reconfigurable Communication Concept for Real-Time Measurement and Control", Instrumentation and Measurement Technology Conference, 2012 IEEE International , pp. 2351- 2356, 2012.
7. Toh Chuen Ling, Norum Lars, "A Performance Analysis of Three Potential Control Network for Monitoring and Control in Power Electronics Converter", 2012 IEEE INTERNATIONAL CONFERENCE ON INDUSTRIAL TECHNOLOGY,pp.224-229, 2012.
8. Ju Kyung Leel, Young Hun Song, Suk Lee, Kyung Chang Lee and Young Jin Lee, "Implementation of Multi-axis Smart Driver System via EtherCAT Network based on IEC61800 standard", 2011 11TH INTERNATIONAL CONFERENCE ON CONTROL, AUTOMATION AND SYSTEMS, pp. 1871-1874, 2011.
9. Jansen Dirk, Buttner Holger, "Real-time Ethernet - The EtherCAT solution", COMPUTING AND CONTROL ENGINEERING JOURNAL, 16-21, 2004.
10. Gianluca Cena, Ivan Cibrario Bertolotti, Stefano Scanzio, Adriano Valenzano, Claudio Zunino, "Evaluation of EtherCAT Distributed Clock Performance", IEEE Transactions on Industrial Informatics, vol.8, no.1, pp.20-29. 2012.
11. Minyoung Sung, Ikhwan Kim, Taehyoun Kim, "Toward a Holistic Delay Analysis of EtherCAT Synchronized Control Processes", International Journal of Computers Communications and Control, vol.8, no.4, pp.608-621, 2013.
12. Liu Xiaosheng, Ren Huifen, Zhao Zhenfeng, Zhang Pengyu, "EtherCAT technology for the network of smart substation", 2012 IEEE 7th International Power Electronics and Motion Control Conference, pp.2300-2304, 2012.
13. Park Jee Hun, Lee Suk, Lee Kyung Chang, Lee Yong Jin, "Implementation of IEC61800 based EtherCAT slave module for real-time multi-axis smart driver system", ICCAS 2010 International Conference on Control, Automation and Systems, pp.682-685, 2010.
14. Liu Yanqiang, Song Yongli, "EtherCAT based functional safety integrated communication", International Conference on Automatic Control and Artificial Intelligence, pp. 1005-1008, 2012.
15. Jiang Li, Hou Mingxin, Wei Fanshao, Jin Minghe, Liu Hong, Chen Zhaopeng, "Evaluation of the MCU networks communication for EtherCAT pro-

cess data interface", WSEAS Transactions on Communications, vol.12, no.10, pp. 509-518, 2013.
16. Qi Junyan，Zhao Jianggui, Wang Lei, "Response times evaluation for embedded EtherCAT networks", International Journal of Advancements in Computing Technology, vol. 4, no. 17, pp. 435-442, 2012.
17. Rok Tavčar, Jože Dedič, Drago Bokal, Andrej Žemva, "Transforming the LSTM training algorithm for efficient FPGA-based adaptive control of nonlinear dynamic systems", Informacije MIDEM, Journal of Microelectronics, Electronic Components and Materials, vol.43, no.2, pp.131-138, 2013.
18. Uroš Legat, "On-line Testing and Recovery of Systems on SRAM-based FPGA", Informacije MIDEM, Journal of Microelectronics, Electronic Components and Materials, vol.42, no.3, pp. 144-151, 2012.
19. Thao Tran Phuong, Ohishi Kiyoshi, Yokokura Yuki, et al, "FPGA-Based High-Performance Force Control System With Friction-Free and Noise-Free Force Observation", IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, vol.61, no.2, pp. 994-1008, 2014.
20. Bahri Imen, Idkhajine Lahoucine, Monmasson Eric, et al, "Hardware Software Codesign Guidelines for System on Chip FPGA-Based Sensorless AC Drive Applications", IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, vol.9, no.4, pp. 2165-2176, 2013.
21. Ristovic Milica, Lubura Slobodan, Jokic Dejan, "Implementation of CORDIC Algorithm on FPGA Altera Cyclone", 2012 20TH TELECOMMUNICATIONS FORUM, pp. 875-878, 2012.
22. Koyuncu Ismail, Ozcerit Ahmet Turan, Pehlivan Ihsan, "An analog circuit design and FPGA-based implementation of the Burke-Shaw chaotic system", OPTOELECTRONICS AND ADVANCED MATERIALS-RAPID COMMUNICATIONS, vol. 7, no. 9, pp. 635-638, 2013.
23. Hace Ales, Franc Marko, "FPGA Implementation of Sliding-Mode-Control Algorithm for Scaled Bilateral Teleoperation", IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, vol. 9, no. 3, pp. 1291-1300, 2013.
24. Das Joydip, Wilton Steven, "Towards Development of an Analytical Model Relating FPGA Architecture Parameters to Routability",ACM TRANSACTIONS ON RECONFIGURABLE TECHNOLOGY AND SYSTEMS, vol.6, no.2, 2013.
25. Kretzschmar Uli, Astarloa Armando, Jimenez Jaime, et al, "Compact and Fast Fault Injection System for Robustness Measurements on SRAM-Based FPGAs",IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, vol.61, no.5, pp. 2493-2503, 2014.
26. Agrawal Nidhi, Kimura Yoshie, Arghavani Reza, et al, "Impact of Transistor Architecture (Bulk Planar, Trigate on Bulk, Ultrathin-Body Planar SOI) and Material (Silicon or III-V Semiconductor) on Variation for Logic and SRAM Applications", IEEE TRANSACTIONS ON ELECTRON DEVICES, vol.60, no.10, pp.3298-3304, 2013.
27. Okumura Shunsuke, Kagiyama Yuki, Nakata Yohei, et al, "7T SRAM Enabling Low-Energy Instantaneous Block Copy and Its Application to Transactional Memory", IEICE TRANSACTIONS ON FUNDAMENTALS OF ELECTRONICS COMMUNICATIONS AND COMPUTER SCIENCES, vol.E94A, no.12, pp. 2693-2700, 2011.
28. Il-Kyun Jung, Sun Lim, "An EtherCAT based Real-time Centralized Soft Robot Motion Controller", 2012 International Symposium on Instrumentation and Measurement, Sensor Network and Automation, vol.1, pp. 117-120,2012.

Arrived: 05. 12. 2013
Accepted: 09. 02. 2014