Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı 28, S. 130-134, Kasım 2021

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue 28, pp. 130-134, November 2021

Research Article

Tracing Research Using Field Programmable Gate Arrays for Aviation Applications

Ferhat Yüksel[1*,2], Ufuk Sakarya[3]

[1*] Turkish Aerospace Industries, İstanbul, Turkey, (ORCID: 0000-0002-4730-9190) ferhat.yuksel@tai.com.tr
[2] Yildiz Technical University, Graduate School of Science and Engineering, Department of Avionics Engineering, İstanbul, Turkey, (ORCID: 0000-0002-4730-9190), ferhat.yuksel@tai.com.tr
[3] Yıldız Technical University, Faculty of Applied Sciences, Department of Aviation Electronics, İstanbul, Turkey, (ORCID: 0000-0002-8365-3415), usakarya@yildiz.edu.tr

(1st International Conference on Applied Engineering and Natural Sciences ICAENS 2021, November 1-3, 2021)

(**DOI:** 10.31590/ejosat.992431)

ATIF/REFERENCE: Yüksel, F. & Sakarya, U. (2021). Tracing Research Using Field Programmable Gate Arrays for Aviation Applications. *European Journal of Science and Technology*, (28), 130-134.

Abstract

Multicore operating system is becoming more significant topic in the developing aviation industry due to the high performance they provide. However, it also brings some difficulties that slow down these developments. Multicore processors are expected to be deterministic and have high reliability because they take on critical tasks in avionic systems. Authorities working in this direction recommend monitoring real-time CPU performances externally with a different system. Interpreting the information obtained at the time of operation and making inferences about the performance of the system is among the suggestions of this authorities. One of the reasons why multi-core processors are not reliable is the conflict experienced due to the common cache memories they use. Therefore, one of the information that needs to be obtained in the reliability is the occupancy rates of the memories used by the CPU. In addition, inferences can also be made about the performance of the CPU by monitoring the memory. In this paper, a tracing research using field programmable gate arrays (FPGAs) for aviation applications is presented. Successful results were obtained by carrying out studies considering the configuration of using two FPGAs in the simulation environment. One of the FPGAs represents the CPU and the cache memory used by the CPU and the other FPGA has the task of monitoring.

Keywords: Field Programmable Gate Arrays (FPGA), Tracing, Aviation Applications, Reliability, Occupancy rates of the memories.

Havacılık Uygulamaları için Sahada Programlanabilir Kapı Dizilerini Kullanarak İzleme Araştırması

Öz

Çok çekirdekli işletim sistemleri, sağladıkları yüksek performans nedeniyle gelişen havacılık endüstrisinde daha önemli bir konu haline gelmektedir. Ancak bu gelişmeleriyavaşlatan bazı zorlukları da beraberinde getirmektedir. Çok çekirdekli işlemcilerin, aviyonik sistemlerde kritik görevler üstlendikleri için deterministik ve yüksek güvenilirliğe sahip olmaları beklenmektedir. Bu yönde çalışan yetkililer, gerçek zamanlı CPU performanslarının harici olarak farklı bir sistem ile izlenmesini önermektedir. Çalışma anında elde edilen bilgilerin yorumlanması ve sistemin performansı hakkında çıkarımlarda bulunulması bu yetkililerin önerileri arasındadır. Çok çekirdekli işlemcilerin güvenilir olmamasının nedenlerinden biri de kullandıkları ortak önbellek bellekleri nedeniyle yaşanan çakışmadır. Bu nedenle güvenilirlikte elde edilmesi gereken bilgilerden biri de CPU tarafından kullanılan belleklerin doluluk oranlarıdır. Ayrıca belleğin izlenmesiyle CPU'nun performansı hakkında çıkarımlar da yapılabilir. Bu bildiride, havacılık uygulamaları için sahada programlanabilir kapı dizileri (FPGA'lar) kullanan bir izleme araştırması sunulmaktadır. Simülasyon ortamında iki FPGA kullanımının konfigurasyonu dikkate alınarak çalışmalar yapılmış ve başarılı sonuçlar elde edilmiştir. Kullanılan FPGA'lardan biri CPU'yu ve CPU tarafından kullanılan önbelleği temsil ederken, diğer FPGA izleme görevine sahiptir.

Anahtar Kelimeler: Sahada programlanabilir kapı dizileri(FPGA), İzleme, Havacılık Uygulaması, Güvenilirlik, Hafızaların Doluluk Oranları.

[*] Corresponding Author: ferhat.yuksel@tai.com.tr

1. Introduction

Today, Field Programmable Gate Arrays (FPGA) are preferred in many fields with the advantages they provide. Energy consumption, the space they occupy in the system, and the high performance brought by the parallel processing authority are among these advantages.

FPGAs are used in many areas of aviation: Radars, sensors, communication, network systems can be given as examples of these areas. Apart from power usage, performance and longevity of devices to be used in the aviation industry is also taken into consideration. The fact that a system being developed in this area can be reconfigured until the last stage can also be considered as one of the reasons for preference. FPGAs are also suitable for performing complex applications. Their flexibility and tolerance for faults can be added to their advantages.

In this sense, the use of FPGA is remarkable. Other advantages can be shown as the ability to transfer and receive data at high speeds, the increase in the usable logic blocks in them, and the higher integration capacity offered within the framework of lower costs (Anvar et al.,2006).

Real-time data processing is becoming increasingly common in the aviation and avionics fields. As a solution to the high power consumption of the system parts and higher performance requirements, hardware designers are turning to the use of FPGAs, which also have the advantage of being reconfigurable.

In the study of (Atitallah et al., 2018), the roles of simulation, testing and integration steps are defined by mentioning FPGA-based designs for applications that can be used in the field of aviation and that require high signal processing.

In the future, space systems are planned to be dynamically configurable and optimized, with less hardware used to increase human safety. There are reconfigurations in space systems, but these reconfigurations only at the software level and the limited power and volume of the hardware that can be used do not give the expected performance. Providing the hardware for the application's needs gives more successful results than configuring the software. In line with the needs, reconfiguring the hardware during operation and enabling the system to reach sufficient hardware resources are considered as the best solution. A study has been made for the reconfigurable system design (Zheng et al., 2005). It is one of the methods used to increase the modularity of the system and reduce its complexity. The use of reconfigurable devices adds a great deal of modularity to the system.

Functionally, modularization of the system makes the use of the desired number of devices upon request and accordingly the use of power consumption more effective. While dynamically reconfigurable FPGAs provide flexibility in the use of resources, they also directly affect the overall performance of the system and increase it (Lie & Feng-Yan, 2009).

In the aviation industry, the reliability of the devices is important. In addition, it is important to monitor the progress and performance of the devices during operation. One of the information that needs to be obtained in the reliability is the occupancy rates of the memories. In this paper, a tracing research using field programmable gate arrays for aviation applications is presented. The configuration of using two FPGAs is realized for the simulation environment. One FPGA is used to simulate CPU and the other is used for tracing applications. High-speed monitoring is provided by FPGAs.

The rest of paper is designed as follows: The related works are given in Section 2. The proposed system is presented in Section 3. The simulation results and discussions are demonstrated in Section 4. In Section 5, there is a conclusion part.

2. Background

Aurora protocol is a protocol developed by Xilinx that provides high-speed communication between two points. We can list the reasons for using the protocol with the following items (Xilinx, 2010);

- High speed data transfer is possible (480 Mb/s to 84.48 Gb/s).
- Various FPGAs can be supported.
- 8B / 10B or 64B / 66B encoding
- Low resource cost.
- Flow control interfaces (AXI4-Stream based)
- Automatically starts and protects the channel.
- Full duplex or unidirectional operation.

Providing 16-bit or 32-bit Cyclic Residual Check (CRC). Connections can be made in one or more ways. LogiCORE IP, which can be used in the development environment, is provided for the use of this protocol developed by the company. With this IP provided, the control of high-speed data flow between two devices and the packet contents can be examined. FPGA cards that support the protocol have transfer blocks that provide high- speed data transfer. An Aurora protocol may contain one or more data transmission channels. Since each of these channels is bidirectional. This protocol, which allows data flow at high speeds, is a top layer protocol such as Ethernet and TCP/IP. It can be used by everyone without any restrictions (Xilinx, 2010). Fig. 1 shows the general data flow chart of the protocol.

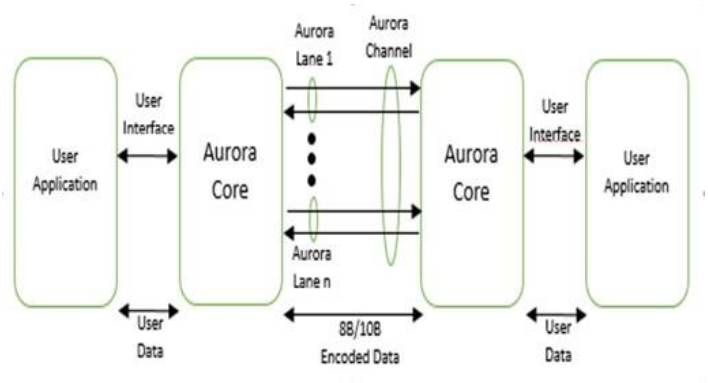

Fig. 1 Aurora protocol (Adapted from [5])

Certification Authorities software team has CAST-32A certificate on ensuring the security, performance and integrity of multi-core processors that will be used in aviation. The philosophy of monitoring the behaviour of a multi-core processor with an external device is within the proposal of the creators of the CAST-32A document (CAST-32A, 2016). The Cast32-A document provides recommendations to detect COTS MCP failures and ensure the continuous safe operation of the airborne system without affecting other components.

There are studies on debugging using the FPGA platform. In the study (Yan, 2012), the system basically consists of 3 basic blocks: the FPGA block, interface and debugging tool. The

monitored system consists of 3 parts: debug block, memory and digital signal processor (Digital Signal Processing - DSP). The debug block provides a way for the system to run while the system is running, and is the block that provides high-speed communication with the outside. The DSP block is the block where real-time error monitoring messages are generated. A memory block is a block used to share memory resources among other blocks. The interface block is the block that provides data exchange between the debugging tool and the FPGA to be monitored. The error monitoring tool is the block that can generate commands such as restarting and stopping the system, as well as performing tasks such as accessing memory, accessing DPS addresses and collecting incoming monitoring messages. In this study, a system that can be monitored from the outside and can make changes on the system has been designed (Yan, 2012).

In a different study (Freitag, 2020), a study was conducted on monitoring the performance counter records of processors using the Aurora protocol. Today, avionics systems are in the process of transition to multi-core processors. The reason for this transition is to realize the design of more performance systems by reducing size, weight and power usage. However, in addition to the advantages it brings, there are also difficulties in using multi-core processors. In the use of multi-core processors, which are planned to be used for real- time systems such as avionics applications, timing problems between cores are encountered. The general aim of the study is the system design that detects the slowdowns due to the use of the multi-core system. A model is created by performing a worst-case execution time analysis of an application running on cores. The slowdown of the observed application can be detected by comparing it with the model created during the execution of more than one application in the same system. Studies have been carried out to reduce the deceleration rate by reducing the impact of low critical applications on kernels if the deceleration exceeds the acceptable level. In the study, data communication was carried out between the processor and the FPGA using the Nexus protocol Aurora interface (Freitag, 2020).

3. Proposed System

The reliability of the devices to be used in the aviation industry is of great importance. While the use of reliable devices is preferred in order to prevent the system from failing, it is also critical to detect errors encountered during use of devices with high reliability. It is also necessary to monitor the progress and performance of the device used in critical systems during operation.

Multi-core processors can run multiple applications simultaneously as they have 2 or more cores. It can cause conflicts over the common memory used by these cores running in different applications.

In such applications, one of the information that needs to be obtained in order to interpret the reliability of the operation of the system is the occupancy rates of the memories. Tracing of the real-time, high-speed occupancy of the memories can be considered as an approach to monitor the memory used by systems using more than one memory.

In our study, a simple CPU that performs various instructions and two different memories that this CPU uses to receive its tasks and write its operations are designed.

By considering the written memory as cache, simulation outputs of transferring the occupancy rate to a different platform with a speed of 5 Gbps were obtained.

The general structure of the system is given in Fig 2.

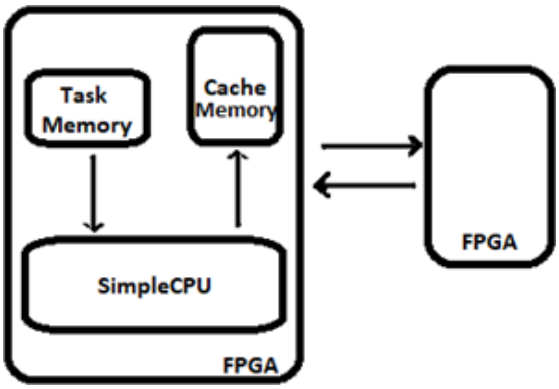

Fig. 2 System architecture

The functions of the components in the system are as follows;

Simple CPU: In line with the data it receives from the task memory component, it performs various instructions and sends it to the cache memory. It performs the information about which instructions to use and in which field it should be written in accordance with the data it receives from the task memory.

Task Memory: It is designed in such a way that 32-bit data can be entered by the user. What operations to do and which data to use is sent to the CPU with 32-bit data. The content of this instruction word is shown in Fig 3.

31 30 29	28	27 14	13 0
Opcode	Immediate	Operand A (27:14)	Operand B (13:0)

Fig. 3 Instruction word

The opcode provides information on which instruction to execute.

While the 14-bit operands A and B are generally used as pointers for the use of data on the cache memory, its value can also be included directly in the processing depending on the content of the immediate bit.

When the Immediate value is 1, the operand B value is processed directly, and if it is zero, the memory uses operand B as a pointer and uses the value at the address it shows in the cache memory.

In line with the incoming opcodes, we can list the CPU instructions in our architecture as follows;

- ADD
- NAND
- SRL (Shift)
- LT (Less or not)
- CP (Copy)
- BZJ (Branch)
- MUL (multiplication)
- DEL (Delete)

Cache Memory: This memory, which has a memory width

of 1GB, is where the data from the CPU is stored.

Interface: Aurora interface, which supports 5GBps speed, is used.

The general structure of the modules used in the project, which are expected to be monitored, and the data transmission interface IP are shown in Fig. 4.

Common clock and reset are used in our system. With the Stop input, it is aimed to stop the data flow through the Aurora interface in the system.

In the simulation environment, it has been observed that error- free data transfer is achieved by making loopback (Tx-Rx) connection in Aurora module.

4. Results and Discussion

According to the CAST-32A position paper, aerospace projects that will use multicore processors may consider using a different "safety net" outside the system for fault detection [8]. CAST-32Agives a set of objectives in order to overcome to multicore certification challenges. And also, it analyses the performance andsecurity effects of multi-core processors used in aviation on the system. Adhering to the objectives of this CAST paper, thegeneral purpose of systems is to monitor CPU behaviour by monitoring the performance and progress of applications in cores without making any changes to applications.

Similarly, the general purpose of our project is to be able to design a system that can approximate CPU behaviour by monitoring the occupancy rates of the cache memories used by the CPU at high speeds.

As an example of instructions realized, the execution of the delete instruction in the simulation environment is shown in Fig. 5. '0110' operation code belongs to delete function. Operand1 shows the value of the address to be deleted. In Fig5, it has been shown that the value 28 of the memory at address 106 is reset withthe incoming delete command. Along with the deletion of the desired address of the memory, the number of the currently used memory amount is also included in the last line of the Fig. 5.

5. Conclusion

Due to the increasing requirements in systems used in aviation, high performance processors are needed. For this reason, single-core processors have begun to leave their place to multi-core processors.

Along with the performance gain of multi-core processors, theyare unsafe due to the use of common resources for aviation applications. Some authorities working on this subject suggest real-time monitoring of system performance external andremotely with a different system.

It is presented that necessary measures can be taken by inferring the performance of the system with the trace results obtained.

In this study conducted in line with these approaches, the general objectives and outputs of the design can be listed with the following items;

- By creating the simple version of the processors used in aviation, it was implemented to the project. In this way, high-cost processors were simulated inthe work.
- In order to make inferences about the performance of the CPU, the cache memories used by the processors have been added.
- Simulation of monitoring the occupancy information of the cache memory used by the CPU by sending it to a different FPGA card at high speed (5Gbps) has been realized.

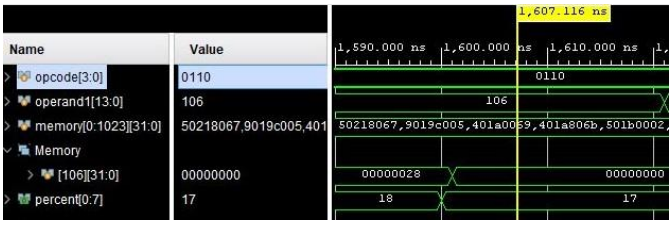

Fig. 5 Simulation of delete instruction

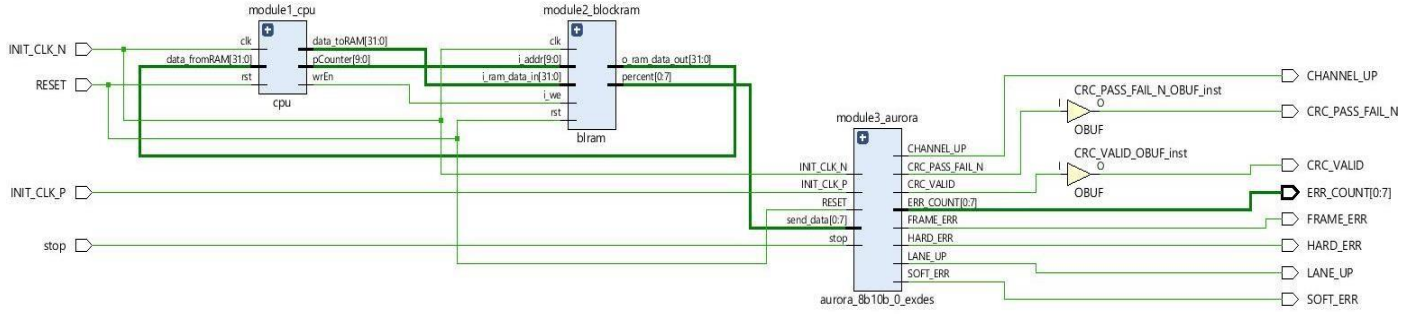

Fig. 4 General structures of the proposed system

References

Anvar, S., Gachelin, O., Kestener, P., Provost, H. L., & Mandjavidze, I. (2006). FPGA-based system-on-chip designs for real-time applications in particle physics. *IEEE Transactions on Nuclear Science, 53*(3), 682-687. doi:10.1109/tns.2006.875076.

Atitallah, R. B., Viswanathan, V., Belanger, N., & Dekeyser, J. (2018). FPGA-Centric Design Process for Avionic Simulation and Test. *IEEE Transactions on Aerospace and Electronic Systems, 54*(3), 1047-1065. doi:10.1109/taes.2017.2733858.

Certification Authorities Software Team, "Multi-core Processors - Position Paper CAST-32A," 2016.

Freitag, J. (2020). Virtual Timing Isolation Safety-Net for Multicore Processors. Dissertation, Universitat Augsburg University.

Lie, W., & Feng-Yan, W. (2009). Dynamic Partial Reconfiguration in FPGAs. *2009 Third International Symposium on Intelligent Information Technology Application*. doi:10.1109/iita.2009.334.

Xilinx, "Aurora 8b/10b protocol specification 2010." http://www.xilinx.com/support/documentation/ip_documentation/aurora_8b10b_protocol_spec_sp002.pdf.

Yan, D. (2012). FPGA Platform for Debug. Master Thesis, Department of Electronic Systems, School of Information and Communication Technology, Kungliga Tekniska Högskolan.

Zheng, W., Marzwell, N., & Chau, S. (n.d.). In-System Partial Run-Time Reconfiguration for Fault Recovery Applications on Spacecrafts. *2005 IEEE International Conference on Systems, Man and Cybernetics*. doi:10.1109/icsmc.2005.1571763.