Proceedings of CECNet 2021
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA210423

RISC-V Based Safety System-on-Chip with Hardware Comparator

EIKE HAHN [1], DOMINIK KALINOWSKI, WALDEMAR MUELLER, MOHAMED ABDELAWWAD and JOSEF BOERCSOEK

ICAS, Institute for Computer Architecture and System Programming, University of Kassel, Kassel, Germany

Abstract. In this paper, a Safety System-on-Chip based on the open-source RISC-V processor SweRV EH1 from Western Digital is presented. A hardware comparator concept is followed. The SSoC is implemented on a Xilinx FPGA system and extended with standard peripherals from the Xilinx IP library and from Cobham Gaisler, so that the overall system has an Ethernet interface in addition to GPIO and UART. The goal is to create a complete redundancy approach with a hardware fault tolerance of nearly 1 from input to output based on the freely available RISC-V instruction set and prove its feasibility.

Keywords. Functional Safety, Miniaturized Safety Systems, Safety SoC, RISC-V

1. Introduction

The ICAS of the University of Kassel has been working for many years on the development of novel structures for embedded systems related to functional safety applications, which are realized specifically for use on programmable hardware or as ASICs. The goal is always the conformity with existing safety standards like IEC 61508 [1], ISO26262 [2], ISO 13849 [3] or similar.

Nowadays, functional safety microcontroller systems are increasingly being designed as a completely integrated system, a so-called Safety System-on-Chip (SSoC). Depending on the application, a hardware fault tolerance (HFT) of at least 1 is targeted for safety-critical architectures. 1-out-of-2 (1oo2), or in general MooN, stands for the degree of redundancy and the tolerance to errors in the system. In the best case, a 1oo2 system achieves a HFT of 1. The overall system consists of two independent systems, one of which is required to perform the task. A 1oo3 system can thus achieve a HFT of 2, a 1oo4 system a HFT of 3 [4].

Current systems mostly contain redundant CPU structures, but the peripherals are singular implemented. This leads to lower HFT than 1 for a 1oo2 redundant system. For most applications this approach is sufficient but there are cases where full redundancy is needed. Examples for such systems are the SSoCs from Texas Instruments [5] or from

[1]Corresponding Author: Research Engineer, Institute for Computer Architecture and System Programming, University of Kassel, Wilhelmshöher Allee 71, 34121 Kassel, Germany; E-mail: eike.hahn@uni-kassel.de.

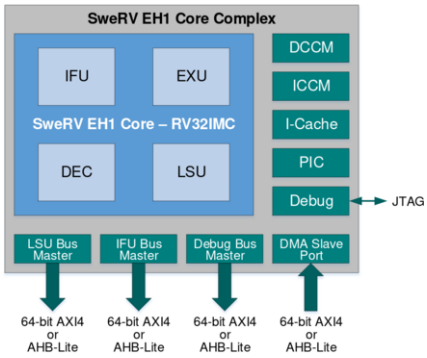

Figure 1. SweRV EH1 Structure [8, p. 1]

Infineon [6]. Also, the new RISC-V safety processor from Fraunhofer IPMS and CAST Inc. only provide a redundant processor architecture with singular peripherals [7].

In the development of functional safety microcontroller systems, microcontroller IPs are used, which are connected to form 1oo2, 1oo2D, 1oo4 or other structures depending on the application. Potentially applicable is a large number of available microcontrollers, which are based on a wide variety of instruction sets. The range extends from freely available and open source to highly optimized and therefore expensive architectures.

Especially for research and for applications with manageable quantities, the IP costs for the microcontrollers used are essential. Furthermore, in the area of functional safety, the use of proven-in-use hardware is important in order to be able to exclude structural errors in the hardware components as far as possible. The SweRV EH1 from Western Digital [8] (Figure 1) fulfils both requirements, since on the one hand it is open source and available under Apache 2.0 license and on the other hand it is already used by Western Digital in millions of hard disk controllers [9].

The paper at hand presents the implementation of a safety structure based on the mentioned SweRV EH1. The resulting ReSC-5 SSoC is based on a full 1oo2 architecture with HFT near to 1 and is implemented on a Xilinx AC701 [10] designed as FPGA. A hardware comparator is implemented as comparison unit. Finally, the SSoC is validated both in simulation and implementation on the FPGA, and the functionality of the hardware comparator is proven by fault injection.

2. Architectural Model of RESC-5 SSoC

In the field of functional safety, several approaches exists describing how a required safety level can be achieved. In addition to the requirements for reliability, theses approaches also include requirements for the availability of a system.

Conventional 1oo2 redundant structures, as used by common safety processors, are based on a hardware comparator principle. Here, the internal system buses of the processing units are permanently compared. This approach triggers a fault condition, and puts the system into a safe state, whenever a difference in the bus systems of the redundant channels occur.

In addition to the development of widely used hardware comparator systems [11–13], ICAS at the University of Kassel has been developing structures with software

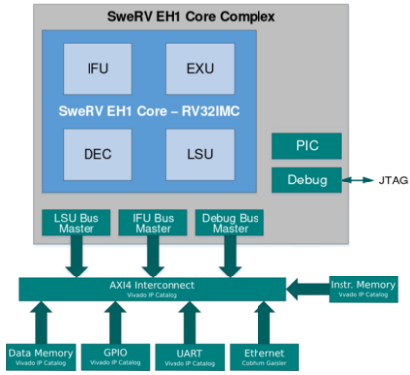

Figure 2. Modified SweRV EH1 Structure

comparators [14–16] for several years. This approach is based on the principle that the processing units of the channels independently compare important values and, in the case of a difference, independently set the system into a safe state. The advantage of this approach is the possibility of executing different software on the processing units, which, however, evaluate safety-critical information independently of each other and thus guarantee the safety of the overall system. In the event of a fault, it is also possible to continue operating the system with a reduced safety level rather than switching it off.

However, here the following architecture for a hardware comparator concept is used: A two-channel system is implemented, which consists of two SweRV EH1 cores. These cores are modified and extended with different peripheral modules, so that two independent and completely redundant processing units are created, each of them represents a complete 1oo1 system. The core of each 1oo1 system is based on original SweRV EH1 core but the directly connected data memory (DCCM), the instruction memory (ICCM) and the instruction cache (I-Cache) are removed. The interrupt controller (PIC) and a debug interface remain in the system. The SweRV EH1 core is extended with a data and an instruction cache from the Xilinx Vivado IP Catalog [17]. Moreover, a GPIO, a UART and an Ethernet module are connected to enable minimal communication. The GPIO and UART modules are also taken from Xilinx, while the Ethernet module from Cobham Gaisler's GRLIB [18] is used. All peripheral modules are connected via an AXI4 interconnect provided by the Vivado library. Figure 2 shows the overall structure of the modified system.

Based on these 1oo1 systems, a 1oo2 system is built up: The single-channel system is instantiated twice for this purpose. The system clock for both channels comes from a common clock source, also called lock-step mode. Since both processing units must execute the same code at the same time, monitored by a hardware comparator, they share a common instruction memory. However, to keep the possibility of common-cause-failures (CCF) low, each processing unit has its own data memory.

The UART interface connections and the GPIO pins are routed separately to the outside in order to establish complete redundancy for them as well. Likewise, each debug interface is routed out such that both processing units can be verified and debugged simultaneously. This redundancy implies nearly no difference in design in contrast to a singular implementation, but a singular implementation would be susceptible for CCF and is therefore avoided.

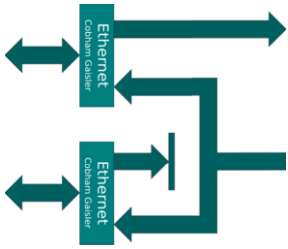

Figure 3. Single Ethernet Interface Concept

An Ethernet interface allows the system to be implemented in a higher-level structure and appears as a singular element. As this is implemented singular, special attention must be paid and is implemented as shown in Figure 3. One of the processing units realizes the communication to the outside while the second processing unit only listens passively for all incoming signals. The second unit does not notice that all its own outgoing signals are not connected, since the first unit sends identical data at the same time and therefore also receives the expected data at the expected moment. The principle represents a common method for the design of singular interfaces of redundant systems. However, this method is more critical for an Ethernet interface than for other standard interfaces, since the base frequency is orders of magnitude faster than, for example, the one of I^2C, which is usually 400 kHz.

The major issue in the conceptual design of the overall system is the avoidance of common-cause-failures. The proposed method to avoid CCF foresees redundancy of the peripherals. A remaining drawback is, that a failure may affect both processing units at the same time. Such a failure can, for example, be caused by external radiation or by changing electro-magnetic fields, and can never be avoided.

To circumvent this, one of the two processing units is delayed by two system clock cycles to realize temporal independence between the two channels. Then similarly affecting failures result in different outcomes in both channels, as both units do not longer perform the same operation at the same time. In case of a bit-flip in both processing units caused by radiation, the comparator structure detects a difference between the processing units and puts the system into a safe state. An overview of the implemented SSoC structure can be seen in Figure 4.

Since the Ethernet interface is implemented singularly, and both processing units need to access it simultaneously, the signals from and to the delayed unit must also be delayed. The delay is realized by D-flipflops in the data path.

3. Implementation of the Hardware Comparator

Finally, a hardware comparator is integrated into the system. The basic task of this is to compare the system buses of the two processing units and, in the event of a difference, to set the system to a safe state. With the approach implemented here, the instructions on the instruction bus are compared, since this carries all CPU instructions. If there are differences, this is a strong indication that an error has occurred in one of the two processing units and the system must be put into safe state. Due to the two delaying clock

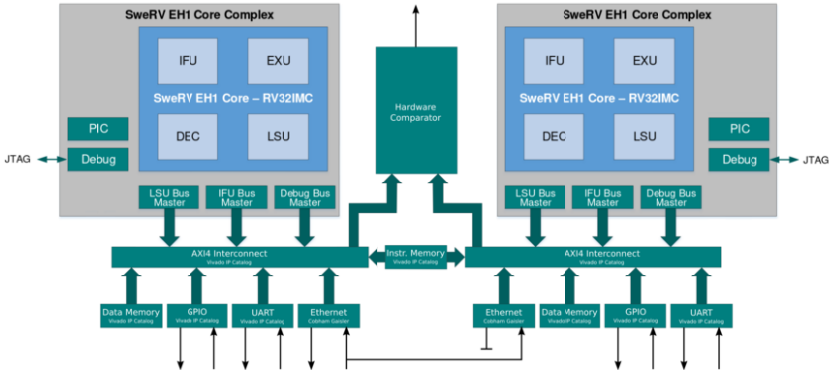

Figure 4. 1oo2 Safety Structure Concept of the ReSC-5 SSoC

cycles between the two processing units, the compared signal must also be delayed such that the comparator compares the same data with regards to content.

The structural design of the comparator can be described with the functionality of an XOR gate. The same signals of both processing units are XORed and in case of a difference a corresponding failure signal is set. This is brought out as a digital output in the design created here. In case of system integration, it can be connected to a special reset input of the SSoC. The reset state is defined as a safe state in most cases. A special reset controller can detect the reason of the last reset at system start-up and distinguish between a normal system start-up or a reset due to a fault condition. A subsequent software-based system diagnosis detects whether the failure had a transient or a permanent cause.

4. Validation of the RESC-5 SSoC

The implemented safety structure is first validated in simulation and then on a FPGA. The environment integrated in Xilinx Vidado is used as the simulation environment. Test programs, which are written in C, are loaded as a .coe file into the Vivado memory model of the instruction memory and executed there when the simulation is started.

First, the GPIO and the UART module are tested. The test program for the GPIO reads an input which is connected to an external push button and controls a LED connected to another pin of the GPIO module. Since both channels have separate GPIO modules, push buttons and LEDs are connected to both channels. Figure 5 shows the FPGA board with the expansion board and the debuggers.

Second, a program is developed, which validates the Ethernet interface. The free LWIP library is used to enable Ethernet communication. On a PC a command line application is used to send and receive single packets.

The communication is shown in Figure 6. It can be seen that the two channels of the safety structure receive a packet via Ethernet and return the data via UART. In addition, it can be seen in the second part of the figure that the tool receives data sent by the interfaces.

Finally, the hardware comparator is validated. For this purpose, a fault is injected intentionally. The wiring diagram for the concept can be seen in Figure 7. The bus signals

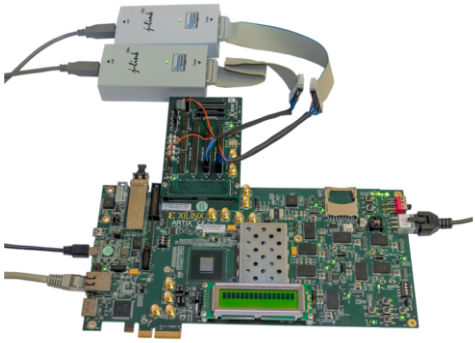

Figure 5. AC701 FPGA Eval Board

of the second processing unit are intentionally changed by an AND gate when a push button is pressed. The comparator applies a high level signal to the designated output directly after the key is pressed. This can be recognized by an illuminated LED.

5. Conclusion

The structure of a 1oo2 safety architecture with two SweRV RISC-V processor cores is presented in this paper. The comparator is based on the hardware comparator principle. The minimal system has been extended with different communication interfaces and represents a ready to use pin redundant safety system. Tests both in simulation and on FPGA have been successfully performed.

The presented method enables the possibility to realize this concept as an ASIC, for example through the Mini@sic program from Europractice [19], and to apply it in real applications – this is matter of future work of the authors. Furthermore, various structural specifications from the generic safety standard IEC 61508 were taken into account in the design generation, such that certification of the structure up to SIL3 is possible. However, a comprehensive calculation of the safety parameters and various qualitative and quantitative analyses, which are also necessary for certification, are still pending.

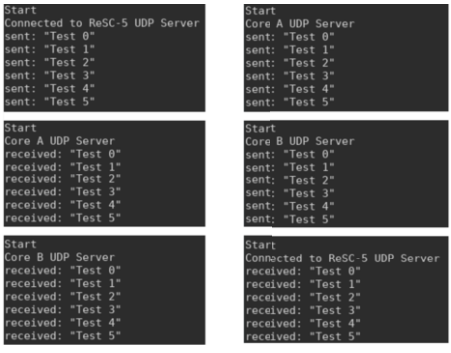

Figure 6. Ethernet Communication

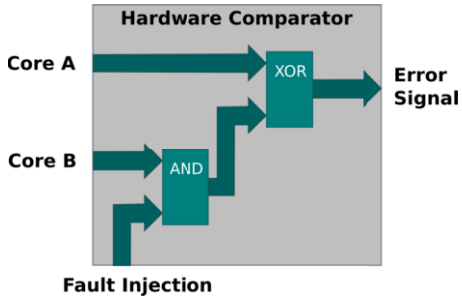

Figure 7. Hardware Comparator Structure

References

[1] Functional safety of electrical/electronic/programmable electronic safety-related systems, IEC 61508, IEC, Apr. 2010.

[2] ISO 26262 Road Vehicles– Functional Safety, ISO, 2018.

[3] ISO 13849 Safety of machinery, ISO, 2015.

[4] J. Börcsök, Functional Safety: Basic Principles of Safety-related Systems, 1st ed. Heidelberg, Neckar: Hüthig Verlag, 2006.

[5] Texas Instruments, Safety Manual for TMS570LS31x and TMS570LS21x Hercules™ and ARM® Safety Critical Microcontrollers: User´s Guide.

[6] Infineon Technologies AG, Highly Integrated and Performance Optimized 32-bit Microcontrollers for Automotive and Industrial Applications. Neubiberg.

[7] Fraunhofer Institute for Photonic Microsystems IPMS, EMSA5-FS – RISC-V Functional safety processor IP Core (accessed: Aug. 24 2021).

[8] Western DigitalCorporation, RISC-V SweRV™ EH1: Programmer's Reference Manual. [Online]. Available: https://raw.githubusercontent.com/westerndigitalcorporation/swerv_eh1/master/docs/RISC-V_SweRV_EH1_PRM.pdf (accessed: Jul. 29 2021).

[9] Zvonimir Z. Bandic, Robert Golla, SweRVCores Roadmap. Accessed: Jul. 29 2021. [Online]. Available: https://riscv.org/wp-content/uploads/2019/12/12.11-14.20a3-Bandic-WD_SweRV_Cores_Roadmap_v4SCR.pdf

[10] Xilinx, Xilinx Artix-7 FPGA AC701 Evaluation Kit. [Online]. Available: https://www.xilinx.com/products/boards-and-kits/ek-a7-ac701-g.html (accessed: Jul. 29 2021).

[11] A. Hayek, B. Machmur, M. Schreiber, J. Börcsök, S. Gölz, and M. Epp, "HICore1: "Safety on a chip" turnkey solution for industrial control," in 2014 IEEE 25th International Conference on Application-Specific Systems, Architectures and Processors, 2014, pp. 74–75.

[12] M. Abdelawwad, A. Hayek, A. Alsuleiman, and J. Börcsök, "FPGA Implementation of a Safety System-on-Chip Based on 1oo4 Architecture Using LEON3 Processor," in 2018 International Conference on Computer and Applications (ICCA), 2018, pp. 231–235.

[13] A. Hayek and J. Börcsök, "On-chip safety system for embedded control applications," in MELECON 2014 - 2014 17th IEEE Mediterranean Electrotechnical Conference, 2014, pp. 315–319.

[14] Josef Börcsök, Waldemar Müller, Eike Hahn, Michael Schwarz, and Mohamed Abdelawwad, "Approach for a Safe-SoC for Cyber-physical Application according to IEC 61508," International Journal of Computers, vol. 14, 2020, doi: 10.46300/9108.2020.14.12.

[15] J. Börcsök, W. Müller, E. Hahn, M. Schwarz, and M. Abdelawwad, "Safe-System-on-Chip for Functional Safety," in 2021 18th International Multi-Conference on Systems, Signals & Devices (SSD), Monastir, Tunisia, 2021, pp. 619–624. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9429321

[16] M. Abdelawwad, M. Drabesch, M. Schwarz, M. I. Hafiz, and J. Börcsök, "Communication SoC based on 1oo2D architecture for industrial human-robot-collaboration," in 2021 18th International Multi-Conference on Systems, Signals & Devices (SSD), Monastir, Tunisia, 2021, pp. 954–959. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9429310

[17] Xilinx, Intellectual Property. [Online]. Available: https://www.xilinx.com/products/intellectual-property.html

[18] Cobham Gaisler AB, SoC Library - Overview. [Online]. Available: https://www.gaisler.com/index.php/products/ipcores/soclibrary/soclibrary-overview

[19] EUROPRACTICE, EUROPRACTICE — MPW & Mini@sic. [Online]. Available: https://europractice-ic.com/mpw-prototyping/general/mpw-minisic/