A RISC-V Fault-Tolerant Microcontroller Core Architecture Based on a Hardware Thread Full/Partial Protection and a Thread-Controlled Watch-Dog Timer

Chapter · March 2020

DOI: 10.1007/978-3-030-37277-4_59

CITATION
1

READS
32

6 authors, including:

Luigi Blasi
Sapienza University of Rome
2 PUBLICATIONS **2** CITATIONS
SEE PROFILE

Abdallah Cheikh
Sapienza University of Rome
6 PUBLICATIONS **20** CITATIONS
SEE PROFILE

Antonio Mastrandrea
Sapienza University of Rome
30 PUBLICATIONS **112** CITATIONS
SEE PROFILE

Francesco Menichelli
Sapienza University of Rome
6 PUBLICATIONS **3** CITATIONS
SEE PROFILE

PREPRINT

A RISC-V Fault-Tolerant Microcontroller Core Architecture Based on a Hardware Thread Full-Weak protection and a Thread-Controlled Watch-Dog Timer

Luigi Blasi[1], Francesco Vigli[1], Abdallah Cheikh[1] Antonio Mastrandrea[1], Francesco Menichelli[1], Mauro Olivieri[1]

[1] Dept. of Information Engineering, Electronics and Telecommunications (DIET)
Sapienza University of Rome, Via Eudossiana, 18, 00184 Roma, Italy
{luigi.blasi, abdallah.cheikh, antonio.mastrandrea, mauro.olivieri}@uniroma1.it

Abstract. The electronics devices that operate in the extreme space environment require a high grade of reliability in order to mitigate the effect of the ionizing particles. For COTS components this can be achieved using fault-tolerant design techniques which allow such design to fulfil the space mission requirements. This paper presents the design and the implementation of one of the Klessydra F03x microcontroller soft core family, called the *F03_mini*, which is a RISC-V RV32I compatible fault-tolerant architecture enhanced by a Hardware Thread (HART) full-weak protection and a thread-controlled Watch-Dog Timer module. The core architecture has been synthesized and implemented on an ARTIX-7 A35 FPGA and fault-injection by the meaning of a functional RTL simulation has been performed in order to evaluate the robustness to Single Event Effects (SEE). Experimental results are provided, illustrating the impact and the benefits obtained by the usage of the proposed TMR protection techniques as well as a thread-controlled Watch-Dog Timer.

Keywords: Microcontroller core architecture, fault-tolerance, RISC-V instruction set, interleaved multithreading, single event effects, Watch-dog timer.

1 Introduction

The electronic devices that operate in the extreme space environment require a high grade of reliability in order to mitigate several effects of ionizing particles [1]. In our design, we considered only soft-errors (SE), such as Single Events Effects (SEE), as we focus on low clock speed (25MHz) applications.

The usage of Commercial Off-the-Shelf (COTS) components as well as an open-source Instruction Set Architecture (ISA) allow a reduction in cost due to the low volume demand for aerospace applications. From this point of view, the growing interest for an extendable microprocessor Instruction Set Architectures (ISA) has led many companies to support the RISC-V open standard [2] [3].

Since this kind of components are not intrinsically protected at hardware level, a fault-tolerant architecture design is required in order to fulfil with the severe environment requirements as well as with resource availability [7][11].

This paper describes the design and the implementation of a compact variant of the Klessydra F03x microcontroller soft core family (named F03b or *F03_mini*) which is a RISC-V RV32I compliant, fault-tolerant architecture enhanced by a TMR-based full-weak Hardware Thread (HART) protection and a Thread Controlled Watch-Dog Timer (TC-WDT) module.

In the following, Section 0 provides an overview of the core microarchitecture and its compatibility with the RV32I instruction set. Section 3t describes the proposed HART full-weak protection techniques, as well

as the utilization of the dedicated TC-WDT. Section 4 reports experimental results about the FPGA implementations and the HDL fault-injection simulation, and finally Section 5 provides the conclusions.

2 The Klessydra processing core family and RV32I compliance

The Klessydra processing core family is a set of cores featuring full compliance with the RISC-V instruction set and intended to be integrated within the PULPino microcontroller platform [4]. To date, the Klessydra family includes:

- a minimal gate counts single-thread core, Klessydra S0;
- a set of multi-threaded low-end IoT-oriented cores, Klessydra T0x;
- a set multi-threaded fault-tolerant cores, Klessydra F03x, both available in different implementations;
- a set of multi-threaded cores, Klessydra T1x, supporting vector processing acceleration for high-speed controllers in high-end IoT nodes [10].

The Klessydra core family features can be summarized as follows:

- Full compliance with the RISC-V architecture specification (instruction set, control and status registers, interrupt handling mechanism and calling convention);
- Compliance with the standard RISC-V compilation toolchain;
- Interleaved multi-threaded execution of RISC-V HARTs. In particular, each HART has its own Program Counter (PC), Control and Status Registers (CSR) and Registers File (RF) and every HART can send a software interrupt to another HART. A new instruction is fetched from a different PC at each clock cycle, according to the interleaved multi-threading scheme.
- Easy and standardized multi-threading programming interface;
- Core synthesis on FPGA (presently, Xilinx Series 7 implementations have been tested);
- Hardware compliance with the PULPino microprocessor platform, as pin-to-pin compatible alternative of the PULPino RI5CY and Zero-riscy core.
- Software compliance with the PULPino microprocessor platform, as compatible I/O memory map, interrupt handler memory map, program/data memory map

We focus our discussion on the “mini” version of the Klessydra F03x core, which implements the 32-bit integer RISC-V machine mode instruction set, namely user-level RV32I base integer instruction set version 2.2 [2] and M-mode privileged instruction set version 1.10 [3].

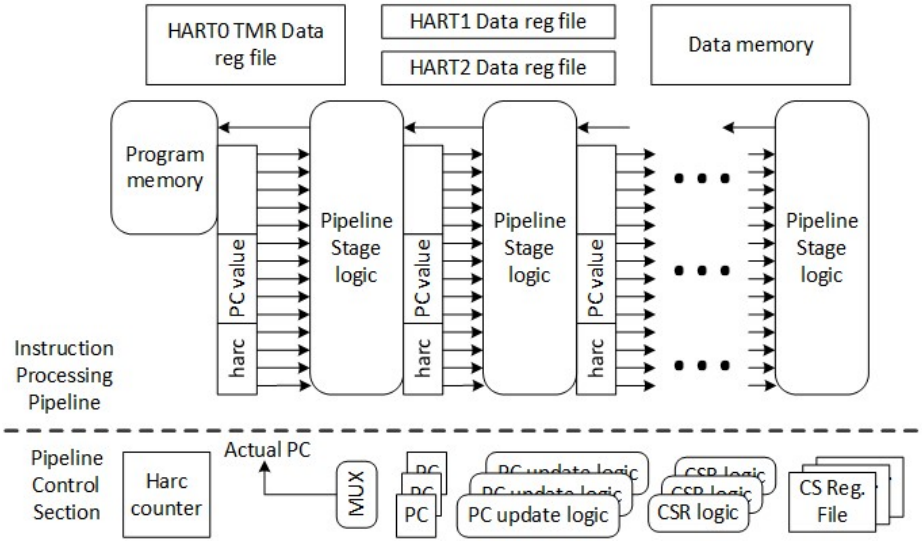

Figure 1 - F03_mini Microarchitecture

3 The F03_mini fault-tolerant microarchitecture

In this section we discuss the architectural choices included in the *F03_mini* core in order to minimize area overhead required by fault-tolerance features. The core shares the same baseline architecture as the T03x [9][6], on which classic Triple Modular Redundancy (TMR) has been applied (Fig. 1). As opposed to the Klessydra F03a core, featuring full TMR protection on all the HARTs supported by the hardware microarchitecture, the *F03_mini* introduces the following general characteristics in order to save hardware resources:

- Different degrees of error protection among the HARTs.
- Reduced set of Counter & Performance Registers.

All the CSRs and PCs are protected using TMR technique, while the non-critical Counter & Performance Registers (CPRs) are not protected at all, in order to reduce the usage of hardware resources.

Klessydra *F03_mini* supports the execution of 3 HARTs. The hardware microarchitecture features a fully-protected datapath for *HART0* and a weakly-protected datapath architecture for *HART1* and *HART*2. Actually, the Processing Pipeline (PP) is completely TMR-protected, while only *HART0* has a TMR-protected register file. In this way it is possible to reduce the use of resources by reducing the reliability of two HARTs. A limited degree of protection is guaranteed on *HART1* and *HART2* by the introduction of the TC-WDT. Moreover, the user can implement software protection techniques to prevent processing failures on weak-protected HARTs, by exploiting the thread-communication features offered by the Klessydra architecture. From the application software point of view, *HART0* will handle the mission critical tasks of the satellite, while *HART1,2* will handle non-critical tasks.

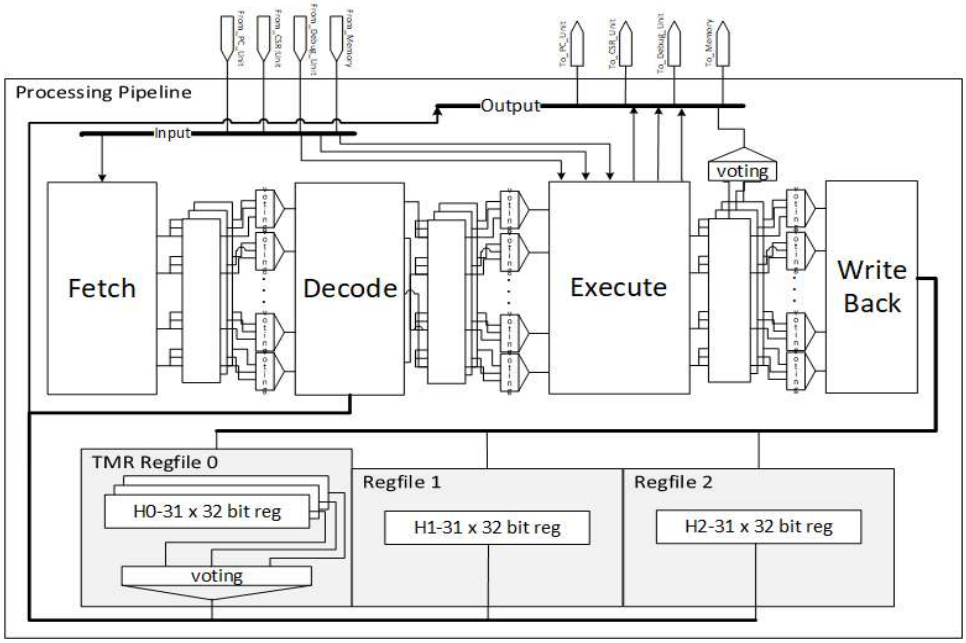

Figure 2 - Processing Pipeline Architecture

The TC-WDT is a critical component for the correct behaviour of the microcontroller core to be used in the space environment, as it provides a limited degree fault-tolerance for weakly protected HARTs whenever a loss of control due to a SEE occurs within the application program flow.

The TC-WDT can be controlled only by *HART0*, i.e. the only one which has full TMR protection. In normal operation, i.e. in the absence of critical SEE on *HART1* and *HART2*, all threads will send their reset request (RST_REQ) to the WDT before its timeout has elapsed. The reset command (RST_CMD) o the WDT can be sent only by *HART0* once it has verified the correctness of the results for weak-protected HARTs (*HART1* and *HART2*). All the requests and commands are performed with write/read access on memory mapped register (accessed by AMBA Peripheral Bus (APB) interface). The complete reset request sequence is described by the following points:

1. *HART1* and *HART2* send their reset requests (WDT_RST command).
2. The WDT enables the flags for *HART1* and *HART2* (in the WDT_CSR register).
3. *HART0* checks periodically both flags in the WDT to check the correct behaviour of HART1 and HART2.
4. HART0 requests the WDT reset by the WDT_RST command.

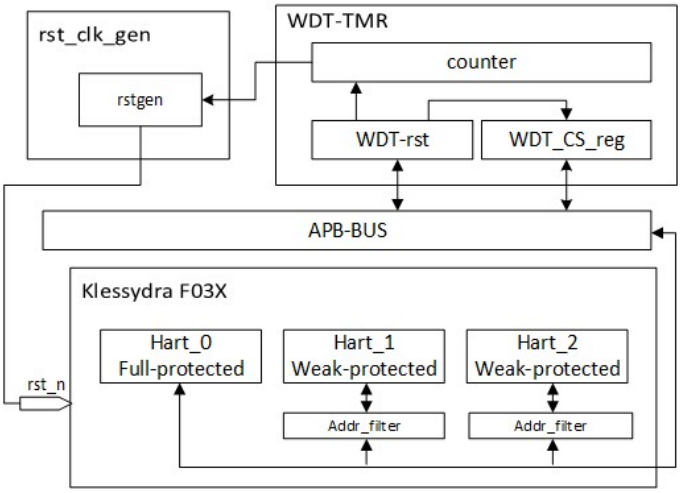

Figure 3: Thread controlled WDT architecture

According to the above description, whenever a SEU causes a loss of control within the program flow of *HART1* or *HART2*, *HART0* will detect a mismatch when checking the WDT flags. In this case, a dedicated software routine will handle the error detection in the proper way, which is application dependent. A software support by means of dedicated error recovery routines is therefore required in order to assure a reliable behaviour.

4 Experimental Results

Klessydra *F03_mini* has been coded in VHDL-2008 HDL language and implemented on a Xilinx ARTIX-7 xc7a35tftg256 device. Here we report the essential results related to area, speed and fault tolerance tests.

Table 1 provides results for the area usage, compared with the fully TMR-protected F03a version.

Architecture	LUT	LUTRAM	FF	IO	BUFG	BRAM	MMCM
F03a	28704	1	20840	28	7	16	1
F03_mini	19465	1	14464	28	7	16	1

Table 1: F03x vs. F03_mini resources usage

Table 2 reports a group of tests have been executed to compare the performance between the *F03_mini* fault tolerant core and non-fault-tolerance T03x core. We can see that the application of the TMR technique in the *F03_mini* cores does not reduce performance in terms of clock cycle.

Test Name	F03_mini	T03x
testALU	123131 cycles	123135 cycles
testCSR	63098 cycles	63098 cycles

testIRQ	316383 cycles	316383 cycles
testException	43949 cycles	43949 cycles

Table 2: T03x vs. F03_mini performances test

To verify the proposed fault tolerant features, we performed several HDL fault-injection simulation. The tests are based on TCL scripts which force random bit flip in each flip-flop inside the core with a rate up to *48-UPSets/100µs*. Table 3 provides the results of fault-injection campaign compared with the fully TMR-protected F03x core.

F03_mini	18 UPS/1µs	24 UPS/0.1µs	48 UPS/100µs
testCSR	123131 cycles	123131 cycles	123131 cycles
testALU	63098 cycles	63098 cycles	63098 cycles
testIRQ	316383 cycles	316383 cycles	316383 cycles

Table 3: F03_mini fault-injection results

5 Conclusions

We illustrated the fault-tolerant microarchitecture used for the implementation of a microcontroller core belonging to the Klessydra F03x processing core family, which is compliant with RISC-V integer 32-bit instruction set and with the widely known PULPino System-on-Chip platform. Performance analysis by the means of FPGA area usage and fault-injection by HDL simulation results was also reported, showing the trade-offs between different micro-architecture organizations (F03x and F03_mini). Future work will be focused on several fault- tolerant techniques based on the intrinsic interleaved multithreading architecture in order to enhance the fault-tolerant of the presented architecture. This work is a fundamental step towards the utilization of RISC-V RV32I microcontroller core as a payload for Nanosatellites (CubeSats, Picosats) which allow academic institutions and small companies to afford space mission research. The first launch of a satellite equipped with a reconfigurable computing sub-system based on F03x cores is expected in spring 2020.

6 References

[1] ESA-ESTEC, “Space product assurance - Techniques for radiation effects mitigation in ASICs and FPGAs handbook,” *ECSS-Q-HB-60-02A*, 2016.

[2] “RISC-V User-Level ISA Specification v2.2, Online: https://riscv.org/specifications/.

[3] “RISC-V Privileged ISA Specification v1.10,” Online: https://riscv.org/specifications/privileged-isa/.

[4] “PULP Platform, Open hardware, the way it should be!,” Online: https://pulp-platform.org//.

[5] A. Reveedran, V. B. Patil, D. Selvakumar and V. Desalphine, “A RISC-V Instruction Set Processor-Microarchitecture Design and Analysis,” in *International Conference VLSI Systems Architectures, Technologies and Applications*, 2016.

[6] Cheikh A., Cerutti G., Mastrandrea A., Menichelli F., Olivieri M. (2019) The Microarchitecture of a Multi-threaded RISC-V Compliant Processing Core Family for IoT End-Nodes. In: Applications in Electronics Pervading Industry, Environment and Society. ApplePies 2017. *Lecture Notes in Electrical Engineering*, Vol 512. Springer.

[7] S. Gupta, N. Gala, G. Madhusudan, V.Kamakoti, “SHAKTI-F: A Fault Tolerant Microprocessor Architecture,” in *IEEE 24th Asian Test Symposium*, 2015.

[8] Klessydra Processing Core Family Technical Manualv8, 2019. Online: http://github.com/klessydra

[9] M. Olivieri, A. Cheikh, G. Cerutti, A. Mastrandrea and F. Menichelli, "Investigation on the Optimal Pipeline Organization in RISC-V Multi-threaded Soft Processor Cores," *2017 New Generation of CAS (NGCAS)*, Genova, 2017

[10] A. Cheikh, S. Sordillo, A. Mastrandrea, F. Menichelli, M. Olivieri, Efficient mathematic accelerator design coupled with an IMT RISC-V microprocessor, In: *Applications in Electronics Pervading Industry, Environment and Society. ApplePies.* 2017.

[11] Blasi, L., Mastrandrea, A., Menichelli, F., Olivieri, M., A space-rated soft IP-core compatible with the PIC® hardware architecture and instruction set, *Advances in the Astronautical Sciences*, Vol. 163, 2018, pp. 581-594