

EMERGING TECH CONFERENCE – Edge Intelligence

Volume 03, 2024, Pages 85 – 90

Proceedings of Emerging Tech Conference:  
Edge Intelligence 2024

Evaluation and DSP Benchmarking of the  
European Radiation-Hardened NG-ULTRA FPGA

Anastasios Xynos<sup>1</sup>, George Lentaris<sup>2</sup>, and Dimitrios Soudris<sup>1</sup>

<sup>1</sup>National Technical University of Athens, Athens, Greece

<sup>2</sup>University of West Attica, Athens, Greece

[axynos@microlab.ntua.gr](mailto:axynos@microlab.ntua.gr), [glentaris@uniwa.gr](mailto:glentaris@uniwa.gr), [dsoudris@microlab.ntua.gr](mailto:dsoudris@microlab.ntua.gr)

**Abstract**

The advent of advanced space applications has transformed the conventional computing systems used in space missions. Prioritizing reliability alongside enhanced performance-per-Watt, radiation-hardened FPGAs have become the preferred choice. This paper focuses on evaluating NanoXplore's NG-ULTRA, the first European radiation-hardened SoC FPGA, for its capability to accelerate high-performance DSP algorithms in space applications. The proposed development and testing methodologies aim to deliver efficient implementations while also assessing the new NG-ULTRA hardware features. The findings indicate that NG-ULTRA offers competitive resource utilization and performance, making it a highly promising option for European space missions.

**1 Introduction**

The space industry is continually evaluating advanced embedded platforms to handle the growing data and computational demands of modern space applications. FPGAs have emerged as a key solution for on-board data processing [1], offering excellent performance-per-Watt in tasks like Vision-Based Navigation (VBN), Earth Observation (EO), and Satellite Communications (SatCom) [2]. In environments requiring resistance to radiation, thermal fluctuations, and vibrations, radiation-hardened FPGAs are preferred over traditional CPU-based processors like the RAD750 and AT697F due to their higher reliability and processing efficiency.

The high-density space-grade FPGA market is largely controlled by two U.S. companies: AMD/Xilinx, offering devices such as Virtex-4QV, Virtex-5QV, and RT Kintex US, and Microchip/Microsemi, with their RTAX, RTG4, and RT PolarFire models. Notably, only two space grade SoC FPGAs are currently available: AMD's XQR Versal and Microchip's RT PolarFire SoC. Recently, NanoXplore, a European company, has entered the scene with its BRAVE line of radiation hardened FPGAs, some of which have already been deployed in space missions [3], [4]. The newest addition, NG-ULTRA, is Europe's first 28-nm radiation-hardened SoC FPGA[5]. This FPGA integrates a quad-core ARM-R52@600MHz processor and a variety of programmable logic resources, making it suitable for space missions.

NanoXplore's radiation-hardened FPGAs have attracted interest from both industry and academia ([6], [7], [8], [9], [10], [11], [12]), with ESA testing and evaluating these devices. We developed a methodology to

support and test the development of these FPGAs, evaluating both the hardware and the software tools. Our approach systematically explores tool-level functionality and HDL circuits to achieve efficient implementations on NG-ULTRA, with a focus on DSP algorithms for computer vision and signal processing in space applications.

Our work focuses on a new methodology for supporting and testing space-grade FPGAs like NG ULTRA, addressing all the hardware advancements. By developing and testing DSP algorithms from computer vision and signal processing for space applications, we demonstrate the viability of NG ULTRA for use in space. This paper highlights an enhanced testing methodology based on works of previous projects [6], [7], compares NG-ULTRA's performance to established FPGAs, and reports benchmarking results, showing that NG-ULTRA provides competitive resource utilization and sufficient performance for radiation-hardened environments.

## 2 Testing Methodology & Development

The proposed methodology is built around NanoXplore's programmable logic toolchain, utilizing the Impulse software for synthesis and place-and-route operations. The synthesis methodology, depicted in Fig. 1, involves three primary stages: parametric configuration of the DSP kernels, exploration of the tool's settings, and an HDL-level deep dive into the DSP kernels. Initially, the configuration phase focuses on fine-tuning algorithmic parameters like input size, input partitioning, convolution dimensions, bit-width of the datapath, and the degree of parallelization, all tailored to match the architecture and resources of the target BRAVE FPGA. Once configured, the synthesis is performed using third-party vendor tools to generate baseline results for comparison.

In the tool-level exploration stage, we first perform an initial synthesis with the default settings to identify any inherent issues. Following this, we systematically explore all available Impulse synthesis settings, applying them both individually and in various combinations, then assessing the resulting synthesis netlists. During this phase, we investigate settings affecting mapping effort, register duplication, arithmetic, logic, and memory targets, DSP usage, and FSM encoding style. If the tool produces errors or unexpected results, alternative settings or changes in the HDL code are employed to resolve these issues. The resource usage of the most optimized, error-free netlist is then compared with the results obtained from third-party FPGA tools. If any major discrepancies are found, adjustments are made either by tweaking the tool settings or by proceeding to the next stage.

The HDL-level exploration phase involves breaking the DSP kernel into smaller subcomponents for individual testing through a recursive process. This detailed examination helps in identifying optimization challenges or errors that may be difficult to detect when analyzing the kernel as a whole. In this phase, we employ template-based coding practices, using NanoXplore's HDL templates to implement the kernel's building blocks. Each HDL block undergoes post-synthesis simulation to ensure accuracy. Once all issues related to errors or optimizations are resolved, we return to the tool-level exploration to evaluate the entire kernel once again. If no viable solution is obtained after these explorations due to errors, excessive resource use, or failed verification, feedback loops (represented by red dashed lines in Fig. 1) are used to restart the process from the configuration phase with a new kernel setup.

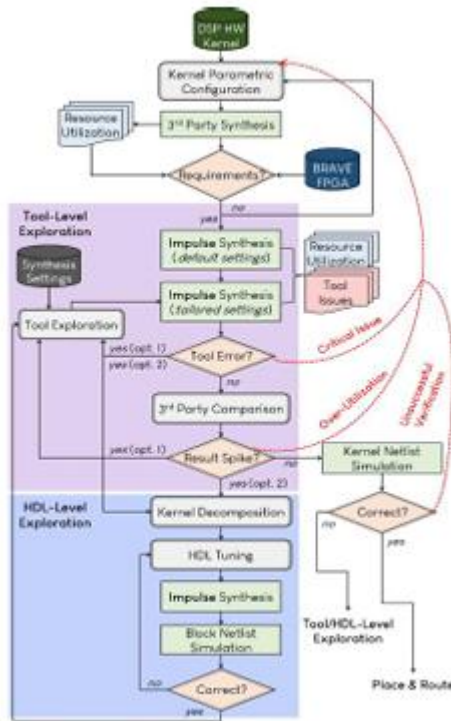


Figure 1: DSP kernel synthesis methodology for BRAVE FPGAs [13]

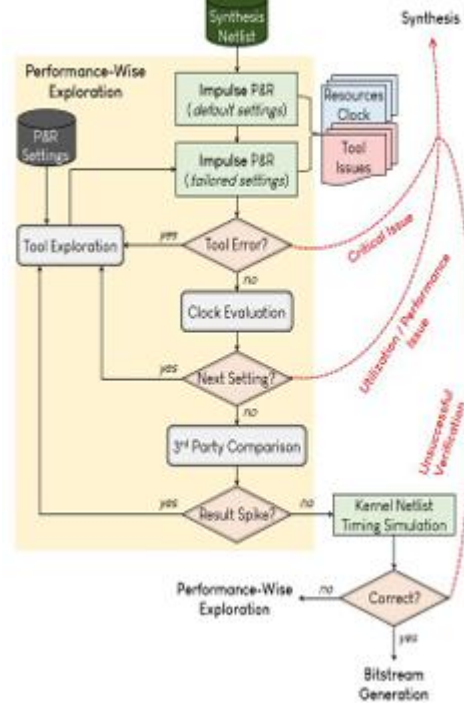


Figure 2: DSP kernel Place & Route methodology for BRAVE FPGAs [13]

The place & route methodology, shown in Fig. 2, starts with the error-free synthesis netlist and focuses on optimizing tool settings to achieve the best possible clock frequency. This involves exploring placement, routing, and physical constraints, while also evaluating specific regions on the FPGA floorplan for mapping. We also fine-tune timing settings, including clock constraints, false paths, maximum delays, and timing-driven placement and routing. As with synthesis, comparisons with third-party tools, simulations, and feedback loops are used to resolve issues like timing violations or inefficient placement, with adjustments made as necessary.

### 3 Experimental Evaluation & Performance Results

For our experimental evaluation, we used several DSP kernels that are computationally and memory-intensive, all commonly found in modern space applications. Specifically, the Harris Corner Detector and Canny Edge Detector are used for feature detection, such as in vision-based navigation and docking systems, GAD-Disparity and SpaceSweep handle stereo matching for depth extraction in 3D scene reconstruction. The QAM Modulator & Demodulator is employed for 16-order modulation and demodulation in 5G satellite communications. The computer vision kernels were tested using 1 Megapixel input images, and the telecom kernel (QAM) was configured for 16-order modulation. For comparison, we ran tests on 3rd-party FPGAs with similar specifications (AMD/Xilinx Zynq UltraScale+ ZU09CG, Microchip PolarFire MPFS460T, and Intel Arria 10 SX480) alongside NanoXplore's smaller FPGAs (NG-MEDIUM, NG-LARGE). We used Impulse v23.5.1.2 for the European FPGA platforms. Table 1 summarizes the results,

comparing NG-ULTRA's performance with the average results of these other FPGAs. For NG-ULTRA, the reported LUT values include both 4 LUTs and carry logic units, while for 3rd-party FPGAs, we converted their 6-LUT and 8-LUT architectures into 4-LUT equivalents for fair comparison. This was done based on approximate conversion rates: one 6-LUT equals about 1.5x 4-LUTs, and one 8-LUT equals around 2x 4-LUTs. NG ULTRA's results reflect its optimized performance after our systematic exploration.

For the Harris Corner Detector, NG-ULTRA slightly exceeds the average 3rd-party LUT usage (+6.6%), with comparable DFF and DSP utilization. NG-ULTRA performs better in DFF efficiency and uses less block memory (108 RAMBs vs. 182), thanks to its larger RAMB size. We also applied a mapping directive to prioritize DSPs for multipliers and carry logic for adders, which improved NG ULTRA's clock frequency by about 8.7%.

For the Canny Edge Detector, none of the FPGAs utilized DSPs. NG-ULTRA, however, used significantly fewer LUTs (about three times less) and DFFs (2.8 times less) compared to the 3rd-party average, mainly because of additional logic in PolarFire, which inflates the average. NG-ULTRA also showed more efficient RAMB usage, consuming around 60% of the average 3rd-party value. Additionally, manual pad placement was used to further optimize NG-ULTRA's clock frequency, resulting in a 2.6% improvement.

**Table 1:** DSP Kernel Resource Utilization: NG-ULTRA FPGA [13] vs 3rd-Party FPGAs (Average Values)

Kernel	I/O	LUT (NG-U)	LUT (3rd-P)	DFF (NG-U)	DFF (3rd-P)	DSP (NG-U)	DSP (3rd-P)	RAMB (NG-U)	RAMB (3rd-P)	MHz (NG-U)	MHz (3rd-P)
Harris	1024x32, 8/32b	16882	15831	19470	21016	92	81	108	182	80	183
Canny	1024x1024, 8/4b	2090	5951	2029	5631	0	0	177	293	67	170
Space Sweep	1024x32, 8/32b	11882	17333	10788	15905	50	36	278	401	58	188
GAD-Disparity	1024x32, 8/10b	1306	12817	3838	14369	494	165	105	169	70	217
QAM16	Nx8, 8/8b	19818	25119	29193	22463	256	162	0	0	92	303

For SpaceSweep, a similar trend emerges: NG-ULTRA shows lower usage in LUTs, DFFs, and RAMBs compared to 3rd-party FPGAs, with the exception of DSP utilization, which is about 38% higher. In the case of GAD-Disparity, NG-ULTRA demonstrates a significant reduction in both LUTs (1306 vs. 12817) and DFFs (3838 vs. 14369), largely due to our design choice of mapping all multipliers and adders to DSPs (494 vs. 165), aimed at improving clock frequency.

In the QAM16 test, NG-ULTRA uses fewer LUTs but higher DFF and DSP resources. By implementing manual pad placement, we boosted clock frequency by 6%. Although QAM16 delivers NG-ULTRA's highest clock speed at 92 MHz, this remains about one-third of the frequency achieved by 3rd-party FPGAs, a trend that persists across most kernels.

**Table 2:** DSP kernel throughput on the three generations of NG-ULTRA FPGAs [13]

Kernel	NG-MEDIUM	NG-LARGE	NG-ULTRA
Harris	--**	4 FPS	11 FPS
Canny	15 FPS	9 FPS	21 FPS
SpaceSweep	92 MPDS	67 MPDS	129 MPDS
GAD-Disparity	--**	7 MPDS	20 MPDS
QAM16	454 MBPS	1415 MBPS	2959 MBPS
Avg. Clock	49 MHz	30 MHz	65 MHz

\* FPS: Frames Per Second, MPDS: Megapixel Disparities Per Second, MBPS: Megabits Per Second.

\*\* Resource overutilization for the targeted accuracy (even for smaller input image).

Table 2 presents the throughput results for the DSP kernels on NanoXplore’s radiation-hardened FPGA models. The kernel configurations mirror those in Table 1, with the exception of NG MEDIUM, where the 1-Megapixel image is split into four 512×512 segments for Canny and sixteen 256×256 segments for SpaceSweep, both processed sequentially. In NG-MEDIUM, the QAM16 parallelization factor is also reduced from 8 to 2. For NG-LARGE, SpaceSweep processes four 512×512 images consecutively. It’s important to note that certain kernels could not be accommodated by NG-MEDIUM and NG-LARGE due to excessive resource utilization (LUTs, DSPs, or cycles) even with reduced input sizes.

In the cases of Canny and SpaceSweep, NG-MEDIUM outperforms NG-LARGE in throughput purely due to its higher clock frequency. NG-MEDIUM completes smaller image runs more quickly than NG-LARGE processes a full-sized image. Meanwhile, NG-ULTRA achieves 2.5 to 3 times the throughput of NG-LARGE. For Canny and SpaceSweep specifically, NG-ULTRA offers a 40% throughput increase compared to the serial runs of NG-MEDIUM. These throughput gains are closely tied to clock frequency, with NG-ULTRA operating at 1.4 times the clock speed of NG-MEDIUM and nearly 2.2 times that of NG-LARGE. Overall, the throughput of these kernels meets the expected performance requirements for space applications. However, further improvements could be made with custom kernel redesigns, such as increased parallelization and manual placement tailored for NG ULTRA, to achieve even greater throughput.

#### 4 Conclusion

In this paper, we introduced a methodical approach for designing, testing, and evaluating high performance DSP algorithms on the latest European radiation-hardened FPGAs. Using our methodology, we achieved efficient and well-balanced implementations on NG-ULTRA, showing substantial performance improvements over previous models. Our future research will target the hardware/software co-design for space applications using the NG-ULTRA SoC FPGA.

## References

- [1] G. Lentaris and others, "High-Performance Embedded Computing in Space: Evaluation of Platforms for Vision-Based Navigation," *AIAA Journal of Aerospace Information Systems*, vol. 15, no. 4, pp. 178–192, 2018, doi: 10.2514/1.1010555.
- [2] O. Kodheli et al., "Satellite Communications in the New Space Era: A Survey and Future Challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 70–109, 2020.
- [3] R. D. Torrijos, "GMV and BRAVE FPGAs: From Studies to Flight Hardware Use," in *NanoXplore's BRAVE Days*, 2023, pp. 1–32.
- [4] NanoXplore, NG-MEDIUM and FUSIO-RT: Mission completed on the dark side of the Moon. 2024. [Online]. Available: <https://nanoxplore.com/index.php/2024/06/07/mediumfusioonmoon/> [
- [5] E. Lepape and M. Le Penven, "New Generation of Rad-Hard SoC FPGA," in *Space FPGA Users Workshop (SEFUW)*, 2023, pp. 1–28.
- [6] K. Maragos and others, "Evaluation Methodology and Reconfiguration Tests on the New European NG-MEDIUM FPGA," in *NASA/ESA Conf. on Adaptive Hardware and Systems (AHS)*, 2018, pp. 127–134. doi: 10.1109/AHS.2018.8541492. [
- [7] V. Leon and others, "Development and Testing on the European Space-Grade BRAVE FPGAs: Evaluation of NG-Large Using High-Performance DSP Benchmarks," *IEEE Access*, vol. 9, pp. 131877–131892, 2021, doi: 10.1109/ACCESS.2021.3114502.
- [8] Y. Barrios and others, "SHyLoC 2.0: A Versatile Hardware Solution for On-Board Data and Hyperspectral Image Compression on Future Space Missions," *IEEE Access*, vol. 8, pp. 54269–54287, 2020, doi: 10.1109/ACCESS.2020.2980767.
- [9] V. Leon and others, "Systematic Evaluation of the European NG-LARGE FPGA & EDA Tools for On-Board Processing," in *European Workshop on On-Board Data Processing (OBDP)*, 2021, pp. 1–8.
- [10] K. Bravhar and others, "SerDes Integrated Into the SpaceWire Interface Helps in Achieving Higher Data Rates," *IEEE Aerospace and Electronic Systems Magazine*, vol. 38, no. 11, pp. 16–27, 2023.
- [11] A. Portaluri and others, "Design Techniques for Multi-Core Neural Network Accelerators on Radiation-Hardened FPGAs," in *Int'l. Symposium on Parallel and Distributed Computing (ISPDC)*, 2023, pp. 16–22.
- [12] E. Danard and A. Comolet-Tirman, "OBC-Ultra, the Rad-Hard NG-Ultra-based On Board Computer for Future Applications," in *European Data Handling & Data Processing Conference (EDHPC)*, 2023, pp. 1–4.
- [13] V. Leon et al., "Development of High-Performance DSP Algorithms on the European Rad-Hard NG-ULTRA SoC FPGA," in *2024 31th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, IEEE, Nov. 2024, pp. 1–4.