Addressing Future Space Challenges using Reconfigurable Instruction Cell Based Architectures

Ahmed O. El-Rayis, Tughrul Arslan, Ahmet T. Erdogan
School of Engineering and Electronics,
University of Edinburgh, Edinburgh EH9 3JL, Scotland, UK
A.El-Rayis@ed.ac.uk

Abstract

Space exploration is evidence to the human desire to better understand our world and the universe that surrounds us. As NASA, ESA and other space agencies design and deploy missions to return to Moon and explore Mars and beyond, the realization is emerging that intelligent payloads must be developed that can respond to the uncertain surroundings autonomously. Space computation challenges have continuously increased in the last decade, and expected to increase further for future space missions. Rapid developments in semiconductor technologies have lead to progression in sensors technology and enormous increase in their capability and accuracy. Rich sensor data encourages researchers to investigate the addition of sophisticated and advanced algorithms in space vehicles to deal with the enormous amount of received data. This has increased the demand on the onboard processing capabilities. In addition, the new trend of shifting the data processing from on-ground to onboard has increased the request on the flexibility and adaptability for space processors. In this paper we discuss the possibility of applying the reconfigurable instruction cell architecture (RICA) in payload data processing. We propose a RICA based processor as a candidate for future space missions. A study of the suggested modifications for the RICA processor to meet future space computation challenges has also been discussed.

1. Introduction

A space mission type is defined by the payload/ instrument installed on-board. Space missions vary from civilian such as remote sensing, space exploration, climate observation, communication, scientific experiments, etc., to military such as communication, remote sensing, radars, etc. New space applications demand high resolution images and more accurate data. The bottleneck for all space missions is the bandwidth limitation. To overcome this problem, a new set of algorithms is being developed to process this information onboard before transmitting to the ground. These new algorithms require high processing capability and low power, in addition to maximizing flexibility and availability of a high quality software development environment [5].

The type of processors that can be thought of to meet such a challenge are GPP (general purpose processor), DSP (digital signal processor), or FPGA (field programmable gate array). GPPs are very general and power hungry and hence they are not efficient for payload data processing. FPGAs are a good candidate and they have proved their suitability in various missions [1-3]. Remote configuration was one of the recent advancements for FPGAs as in [4] but their power consumption raises major concerns due to the limitation of onboard power budget. DSPs have been widely used by various space agencies such as ESA [5]. However, they are facing a great challenge, as new algorithms and future complex functions demand very high processing capability, and current DSPs cannot satisfy these demands.

Reconfigurable array based architectures are recent developments in reconfigurable computing. However, for space applications, reconfigurable architectures are currently not mature enough, due to either the lack of comprehensive software development tools or being dependent on a processor for control (considered as a co-processor for computationally intensive applications) as discussed in [6]. Recent coarse-grain reconfigurable array architectures are required to be combined with a RISC processor such as Morphosys and Garp [13-14]. We introduce here a payload data processor candidate based on a dynamic reconfigurable instruction cell array, RICA [7].

The paper is organized as follows. Section 2 addresses the architectural requirements and processing demands for future space missions. The new processor is introduced in section 3. Section 4 presents some of the processing capabilities of the new processor while section 5 concludes the paper.

This research is funded by the UK Engineering and Physical Sciences Research Council (EPSRC) under grant EP/C546318/1.

978-0-7695-3166-3/08 $25.00 © 2008 IEEE
DOI 10.1109/AHS.2008.66

2. Future Space Applications Requirements and Architecture Specifications

The classical approach for payload data processing for space missions includes the following steps. There is a preliminary processing to take place on-board which is generally followed by data compression to limit the transferred data size. Thereafter, the data is downloaded to the ground reception station, where post processing takes place with the decompressed data. This may include geographical corrections and mapping for image data. Finally, the received data is analysed, as shown in Figure 1. This process is time consuming, especially when an immediate action need to be taken based on certain observations on the received data. Constrained downlink resources and long latencies impose severe limits on spacecraft operations; reducing performance and directly limiting science return [9]. For example, the THEMIS instrument on Mars odyssey is turned on 100 percent of the time but only collects data in the range of 5-10 percent of the time due to downlink limitations [8].

In remote sensing systems, a major problem lies in the limited availability of bandwidth and resources necessary for acquisition, processing and transmission of the images related to a given terrestrial area, as acquired by the sensors mounted on an airborne or space borne remote platform. Since the spatial, spectral and radiometric resolutions of optical and radar sensors are getting finer and finer, the amount of collected data is huge. New-generation sensors aim at increasing the resolution even further. Because of the huge amount of data, powerful compression algorithms are required to match the available channel resources. Moreover, many Earth Observation satellites are required to transmit data to the ground in real time, which means that compression devices with very high throughput are required.

The recent and future direction for space missions is the shifting of on ground data processing to be carried out onboard, as presented in Figure 2. This is mainly due to the bandwidth limitations. Introducing intelligence to the

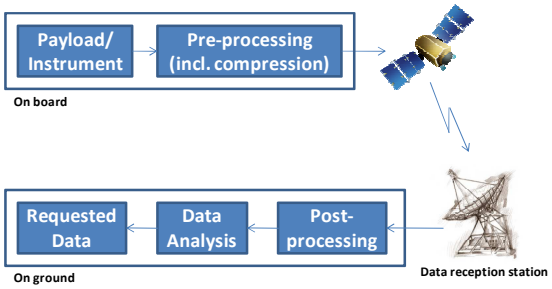

Figure 1. Classical payload data processing locations.

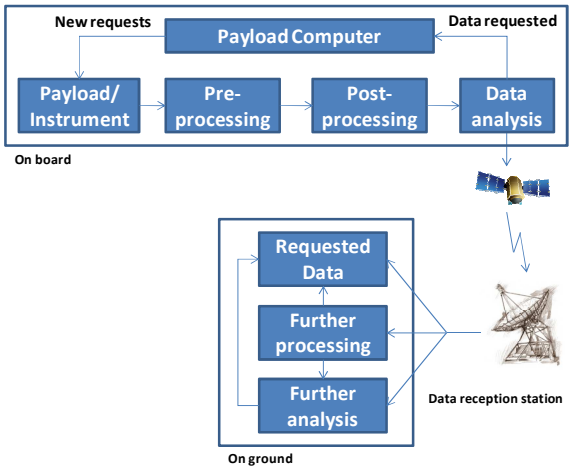

Figure 2. Payload data processing flow with on-board embedded intelligence as a future trend for space missions.

onboard processing equipments is one of the key elements for future missions for immediate decisions that need to be taken and for continuing learning process. The role of a satellite will no longer be just gathering of information and passing it to the ground, but process, analyse and take decisions as well. This will reduce the time wasted waiting for the revisit window followed by on-ground decisions; also this will enhance and maximize the efficiency of the spacecraft and its data quality and reliability. This new direction will require higher processing capability that supports such complex and demanding processes.

Due to the rapid developments in the sensors technology, higher quality data can be gathered. To deploy novel sensors there should be a suitable way to manipulate such amount of data. Recent research work focuses on the data compression methods in order to increase the data volume downloaded on the limited bandwidth links. This opened a wide research area for investigation in both lossy and lossless compression, which is mainly dependant on the type of gathered information. This direction is not the only solution for the limited bandwidth but it should be accompanied by data processing capabilities as mentioned earlier.

For realization of this new trend in space missions, new processors are required which satisfy various requirements. Floating point support with higher precision is one of main requirements for the future processors, as emphasised by ESA [5]. High throughput capability is thought to be an essential requirement due to the higher capabilities of future space instruments. Software development environments and the debugging capabilities are also to be considered for future space processors. As a default, radiation harden proven is a basic requirement for space systems.

3. Reconfigurable Instruction Cell based Architecture

The reconfigurable instruction cell architecture, hereafter RICA, addresses all the desired requirements for future devices. RICA architecture is based on the dynamic reconfiguration of heterogeneous processing elements connected to each other through a reconfigurable interconnect structure. The architecture has characteristics, such as pure ANSI-C programmability, ultra low power, and high performance. The processing elements within the architecture can support primitive operations, such as addition, multiplication, shift, logic, multiplex, etc (Figure 3). Additional processing elements are provided to handle control and branch operations, in addition to some memory elements to handle intermediate results. The current architecture is based on 32-bit operands with a memory distributed in 16 banks. The architecture supports up to four memory read and memory write accesses simultaneously.

The main distinguishing and novel features of RICA architecture are as follows [7]: An array of customizable instruction cells (ICs) that can be dynamically reconfigured on every clock cycle enabling the mapping of data-paths of both dependent and independent instructions. RICA has special ICs to deal with execution flow control (conditional branches). Therefore, it does not require coupling to a general purpose processor, since it provides these features. The cell array configuration can be tailored towards different application domains, thus providing additional power optimizations and performance. A reconfiguration rate controller is provided

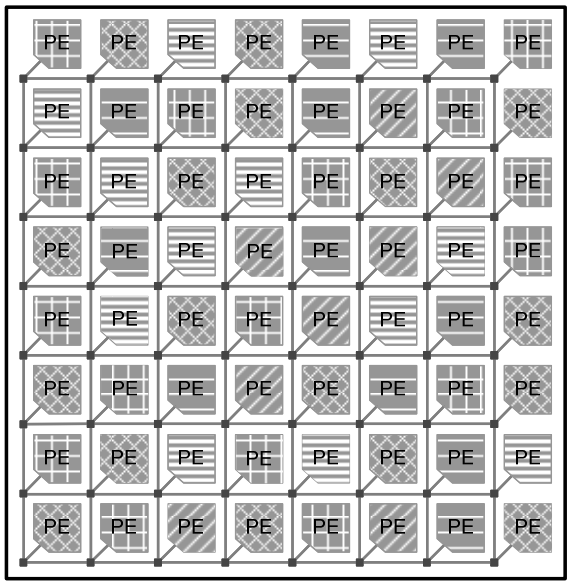

Figure 3. Heterogeneous processing elements (PEs) as the base for the proposed platform for future space missions.

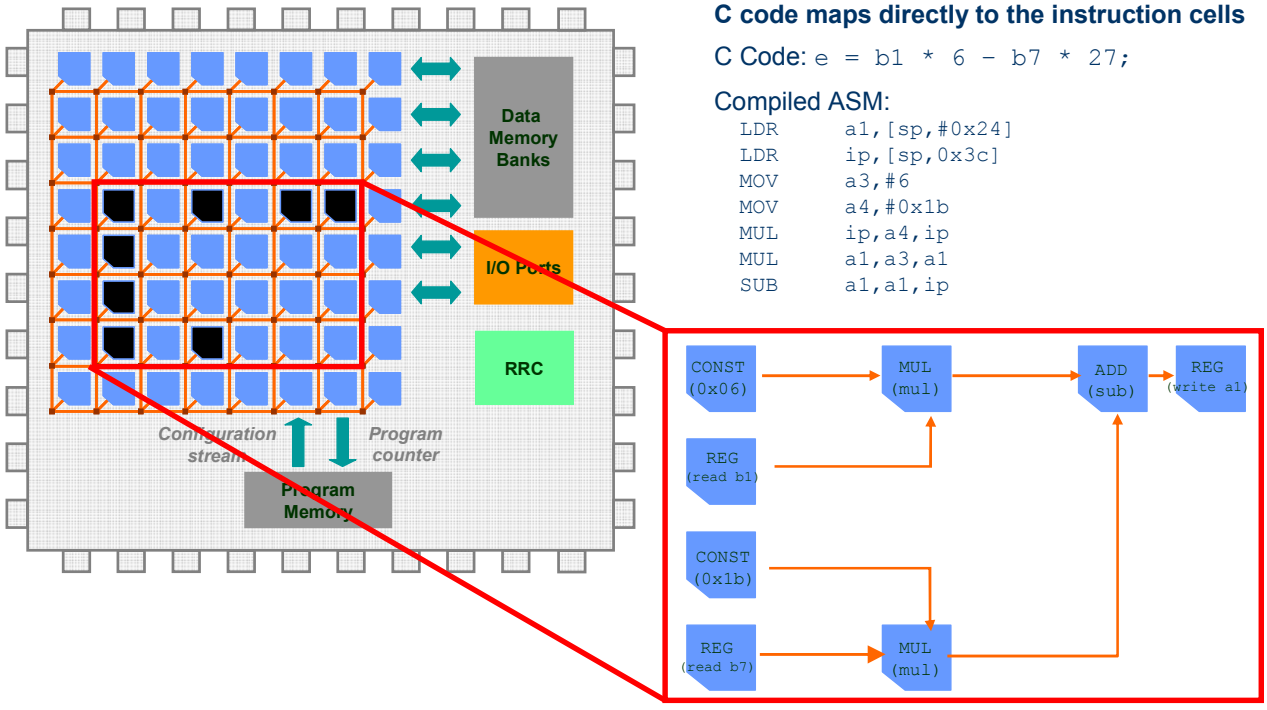

Figure 4. Example of a C-code mapping on the RICA architecture.

Table 1: Performance comparisons between RICA and other processors

Processors	OpenRISC – 112MHz		ARM7 – 110MHz		TI C64x – 600MHZ		RICA	
Benchmark functions	Minimum execution time (us)	Energy consumed per Operation (nJ)	Minimum execution time (us)	Energy consumed per Operation (nJ)	Minimum execution time (us)	Energy consumed per Operation (nJ)	Minimum execution time (us)	Energy consumed per Operation (nJ)
DCT (1-D)	0.91	10.20	0.95	9.40	0.11	34.68	0.12	4.70
DCT (2-D)	44.39	497.00	34.18	338.00	2.94	899.10	3.01	159.30
IIR	1.61	18.00	1.60	15.80	0.07	19.89	0.17	16.33
DCT: Discrete cosine Transform, IIR: Infinite Impulse Response								

to minimize the impact of varying critical paths, which is an intrinsic characteristic of supporting dependent instructions. The architecture is programmable with a high level language (i.e., ANSI-C). An example of a C-code mapping on RICA processor is presented in Figure 4.

For RICA to be deployed in space, there are some aspects need to be considered. RICA as a reconfigurable array of processing elements (PEs) has a built-in fault tolerance capability. For example, if a PE gets faulty, RICA can reconfigure around it without affecting the system operation. In addition, RICA processor can be developed in a Rad-hard manufacturing process if required to provide more resistance to radiation effects. Another issue that has been mentioned earlier as one of the key demands for future space processors is to have a floating point capability. The current version of RICA does not support floating point applications, however, this can be provided by including a number of floating point PEs. It is important to mention that the RICA processor has a complete software tool set that supports the development of applications to run efficiently on RICA, including automatic parallelism extraction in order to maximize the performance. In addition, the RICA tool set includes a simulator that provides the execution performance of the implemented application algorithms on RICA. The information generated by the simulator can also be used to identify various optimisations that could be applied to the algorithms run on RICA in order to further improve the performance characteristics such as throughput and energy consumption.

4. Processing capabilities achieved by RICA architecture

There are various applications being developed and tested on the RICA architecture. For example, the real time LDPC encoder for the WiMAX (IEEE 802.16e) standard achieved a throughput of 47Mbps and could be optimised further to achieve 79Mbps [10]. The implementation of the De-Blocking filter of the recent H.264/AVC video compression allowed the processor to attain an average of 81,405 filtered MBs per second [11]. A recent development of the real time Reed Solomon encoder using GF-Mul (Galois Field multiplier) processing element on RICA achieved an impressive 99 Mbps [12].

A comparison of the RICA processor with some other processors including low power DSPs is presented in Table 1. The benchmark functions are used widely in payload processing, and are known to be computationally intensive. From Table 1, it can be seen that RICA achieves the minimum consumed energy for all benchmarks. In terms of the execution time, RICA has a similar performance to TI C64x and much better performance compared to other two processors. These results put RICA as a strong candidate for future payload data processing.

5. Future work

For the RICA processor to be deployed in space applications a number of enhancements are needed. As discussed in Section 3, the current version of RICA supports fixed point applications only, and therefore another version will need to be developed which can support floating-point applications required by space missions. Another future research work is to consider the precision requirements and to balance between power consumption and data accuracy. Furthermore, there is an ongoing work for the development of a multi-core based RICA array. This multi-core RICA will be able to execute complete systems or complex functions, with the capability to reconfigure and adapt in real-time whenever required. This is one of the crucial requirements for future complex space missions.

6. Conclusion

In this paper, an overview of the current and future space payload processing trends has been provided. We have presented the reconfigurable instruction cell array architecture as a candidate for future space applications. Next generation instruments to be carried on-board spacecraft will collect large quantity of information at increasing rates, due to the recent technological improvements in sensor technology. The huge amount of data generated on-board is competing with the limited channel resources available for the transmission of data to the ground. The solution is to increase the processing capabilities in order to be able to process and analyze rich sensors data. Future directions of space missions, and their computational demands and requirements have been highlighted. Results of various applications and algorithms running on a RICA processor have been reviewed demonstrating that this processor can provide better performance compared to some other processors, and is superior in consuming minimum energy. Therefore, RICA is a very promising candidate for future space mission payload processing.

7. References

1. A.S. Dawood, J.A. Williams, S.J. Visser, "On-board satellite image compression using reconfigurable FPGAs", *Proceedings of IEEE International Conference on Field-Programmable Technology*, 2002, pp. 306- 310.

2. M.E. Dunham, M.P. Caffrey, P.S. Graham, "Challenges and Successes in Space Based Reconfigurable Computing", *Field-Programmable Logic and Applications,* Springer, Berlin, 2003, pp. 948-951.

3. A.O. El-Rayis, A. Melnyk, "Localized Payload Management Approach to Payload Control and Data Acquisition Architecture for Space Applications", *Proceedings of Second NASA/ESA Conference on Adaptive Hardware and Systems*, Scotland, 2008, pp. 263-272.

4. M. Surratt, H.H. Loomis, A.A. Ross, R. Duren, "Challenges of Remote FPGA Configuration for Space Applications", *IEEE Aerospace Conference Proceeding*, 2005, pp. 1-9.

5. M. Suess, "Next Generation Processor for On-board Payload Data Processing Application-Synthesis", *ESA Round Table*, ESTEC Netherlands, 2007.

6. F. Hannig, J. Teich, "Design Space Exploration for Massively Parallel Processor Arrays", *Proceedings of the 6th international Conference on Parallel Computing Technologies*, 2001, Springer-Verlag, London, pp. 51-65.

7. S. Khawam, I. Nousias, M. Milward, Y. Yi, M. Muir, T. Arslan, "The Reconfigurable Instruction Cell Array", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2008, pp.75-85.

8. K. Wagstaff, R. Castano, S. Chien, A.B. Ivanov, T.N. Titus, "An Onboard Data Analysis Method to Track the Seasonal Polar Caps on Mars", *Proceedings of the International Symposium on Artificial Intelligence, Robotics, and Automation in Space*, 2005.

9. A. Jonsson, R.A. Morris, L. Pedersen, "Autonomy in space: current capabilities and future challenges", *AI Magazine*, 2007.

10. Z. Khan, T. Arslan, "Pipelined implementation of a real time programmable encoder for low density parity check code on a reconfigurable instruction cell architecture", *Proceedings of the conference on Design, automation and test in Europe*, 2007, France, pp. 349 – 354.

11. A. Major, I. Nousias, S. Khawam, M. Milward, Y. Yi, T. Arslan, "H.264/AVC In-Loop De-Blocking Filter Targeting a Dynamically Reconfigurable Instruction Cell Based Architecture", *Proceedings of the Second NASA/ESA Conference on Adaptive Hardware and Systems*, Scotland, 2007, pp. 134-138.

12. A.O. El-Rayis, T. Arslan, A.T. Erdogan, "Real Time Programmable Reed Solomon Encoder for Multi-Standard Wireless Communication Systems on a Reconfigurable Instruction Cell Architecture", *Submitted to the International Conference on Field Programmable Logic and Applications*, Germany, 2008.

13. H. Singh, L. Ming-Hau, L. Guangming, F.J. Kurdahi et al., "MorphoSys: an integrated reconfigurable system for data-parallel and computation-intensive applications", *IEEE Transactions on Computers*, May 2000, pp. 465-481.

14. J.R. Hauser, J. Wawrzynek, "Garp: a MIPS processor with a reconfigurable coprocessor", *Proceedings of the 5th Annual IEEE Symposium on FPGAs for Custom Computing Machines*, 1997, pp. 12-21.