ISAAC – A Case of Highly-Reusable, Highly-Capable Computing and Control Platform for Radar Applications

Yutao He, Charles Le, Jason Zheng, Kayla Nguyen, and Dmitriy Bekker
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109 USA
Email: {Yutao.He, Charles.T-C.Le, Xin.Zheng, Kayla.Nguyen, Dmitriy.Bekker}@jpl.nasa.gov

***Abstract* — ISAAC is a highly capable, highly reusable, modular, and integrated FPGA-based common instrument control and computing platform for a wide range of instrument needs as defined in the Earth Science National Research Council (NRC) Decadal Survey Report. This paper presents its motivation, technical approach, and the infrastructure elements. It also describes the first prototype, ISAAC I, and its application in the design of SMAP L-band radar digital filter.**

I. INTRODUCTION

The Earth Science National Research Council (NRC) Decadal Survey Report [1] lays out a roadmap of Earth Science instruments that features a wide range of digital electronics requirements, ranging from a simple 8051 micro-controller as in the CLARREO UV-Visible-IR interferometer to a higher-performance radar digital electronics as in the SMAP L-band radar [2] which needs demanding on-board processing capability. Traditional approach in today's instrument development will incur the significantly high non-recurring cost and unnecessary duplication of costly efforts, yet in the meantime will still face daunting challenges in meeting performance and function requirements [3].

The vision of ISAAC (**I**nstrument **S**h**A**red **A**rtifact for **C**omputing) is to provide a highly capable, highly reusable, modular, and integrated FPGA-based common instrument control and computing platform that can be shared by multiple Earth Science and Planetary Exploration instruments. This reusable framework offers an unprecedented combination of adaptability, computation power, I/O bandwidth, digital interface standards, and data processing capability in a single common low mass/power and small form factor platform with significantly reduced non-recurring cost and risk to Earth Science instruments such as SMAP (Soil Moisture Active-Passive) and other future NASA Planetary Exploration instrument of diverse requirements.

II. ISAAC OVERVIEW

ISAAC's unique technical innovations are embodied in its six key components as shown in Figure 1: *iBoard* - the FPGA-based hardware substrate; *iCore* - the library of Register-Transfer-Level (RTL) Intellectual Property (IP) cores implementing common computationally-intensive instrument control and computing functions; *iPackage* – the collection of software functions that implements common non-computationally-intensive instrument control and computing functions; *iBus* – the standard and unified hardware/software interface; *iBench* - the suite of benchmark instrument data streams for performance validation and tuning of a completely-configured system; and *iTool* - the integrated tool-chain providing a familiar and end-to-end design flow for digital system designers. Collectively, *ISAAC* provides instrument electronics designers with a reusable and integrated framework that enables to configure a complete instrument control and computing system on a per-application-basis to match various instrument requirements.

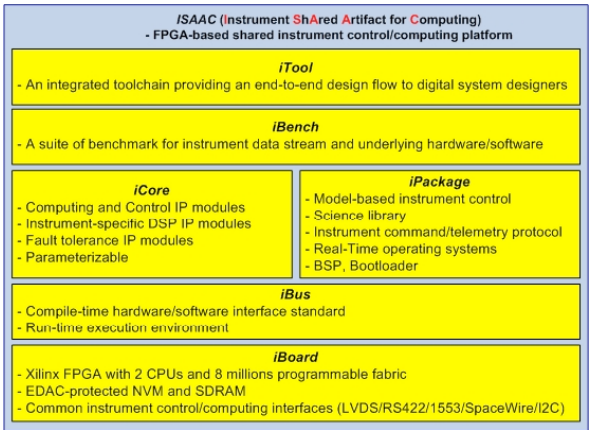

Figure 1 ISAAC Framework

The detailed functional capability of each *ISAAC* component is described as follows:

- ***iBoard**:* is a single board computer (SBC) featuring a high-performance Xilinx Virtex 4 or Virtex 5 FPGA chip with two embedded PowerPC405 processors and over 8 million reprogrammable logic gates and provides the complete set of key instrument control and computing capabilities.
- ***iCore***: is composed of a set of standard and parameterized IP Cores that implements common

computationally-intensive instrument control and computing functions. It is divided into three groups at the top level: *iCore-ctl* - implements control related functions such as commands handling, telemetry collection, and timekeeping; *iCore-ft* - implements fault tolerance functions such as detection of Single-Event-Upset (SEU)-induced errors; and *iCore-dsp* – implements data processing functions common in instruments such as digital filter, Fast Fourier Transform (FFT), Finite-Impulse Response (FIR), Radio-Frequency Interference (RFI) detection and mitigation, presum, block floating point quantizer, convolution, spatial correlation, data compression, and image feature extraction.

- ***iPackage***: is a library of software packages and run-time kernel that augments the *iCore* to provide a complete instrument control and computing solution.
- ***iBus***: defines a set of compile-time hardware/software interface standard called *iBus-ct*, and run-time communication standard called *iBus-rt* to facilitate easy adaptability and integration of components during the system configuration and operation. *iBus-ct* is to unify *iPackage* Application Programming Interface (API)s, standardize the *iCore* interface to external components and the communication bus, organize data for user applications, and expose application developers to common FPGA resources. *iBus-rt* provides a unified integral operating system environment for a complete *ISAAC* configuration by abstracting FPGA fabric as native computational resources. In particular, it will specify whether the resource meets an application's requirements, configure and manage the FPGA, detect/handle hardware faults and interrupts, and transfer data between the FPGA fabric and the processors as well as other resources such as read/write to a mass memory file system.
- ***iBench***: is a benchmark suite that consists of validated historical instrument data streams from different types of instruments (such as radars, radiometers, and imagers), and a common testbench architecture that allows integrated functional testing and validation of ISAAC-based instrument digital electronics system. It will be used to validate and fine-tune the overall performance of one specific *ISAAC* configuration, and to support design-space exploration.
- ***iTool***: is an integrated tool-chain that provides a complete design flow from algorithm development to RTL implementation on the FPGA devices. It provides different front-end by leveraging industrial-strength tools.

III. ISAAC PROTOTYPE – ISAAC I

The first proof-of-concept prototype, ISAAC I, has been developed with the following specific capabilities in its six components.

- ***iBoard***: is a Xilinx ML410 board featuring a Xilinx Virtex 4 FPGA device that contains two embedded PPC405 CPU hardcore and up to 8 millions logic gates programmable fabric. It also provides an IEEE 754 single-precision FPU softcore, 256 MB DDR2 RAM, up to 2GB SystemACE CompactFlash, two 10/100/1000 Mbps Ethernet ports and UART ports, one Interrupt controller, DMA engine, watchdog timer, and I2C RTC and temperature sensors.
- ***iBus***: its prototype design uses CoreConnect as the SoC Bus and supports three models of hardware/software interface: shared memory, FIFO, and memory-based.
- ***iCore***: the prototype IP library includes the off-the-shelf cores from Xilinx and the ISAAC developed digital filter IP core and the function fitting IP core.
- ***iPackage***: the prototype software library includes iBoard BSP, u-boot-based bootloader, v2.4/2.6 Linux kernel, interrupt-based run-time executive, Gnu Science Library (GSL), MBE-based instrument control software framework, and the baseline ICTP (Instrument Command and Telemetry Protocol) for the spacecraft-to-instrument interface.
- ***iBench***: the prototype design includes a suite of synthesized instrument data streams for radar, radiometer, and imager, the *lmbench*-based benchmark suite for profiling the Linux-based iBoard environment, and the iBoard/FPGA-based unified testbench architecture allowing end-to-end functional validation and performance profiling
- ***iTool***: the prototype toolchain includes, Xilinx EDK/ISE as RTL-to-FPGA backend, Impulse C as C-to-RTL front-end, Xilinx AccelDSP as Matlab-to-RTL front-end, MBE as StateChart-to-C front-end, and GNU gcc cross-compile tool-chain for software development.

IV. ISAAC APPLICATION IN SMAP RADAR

A. SMAP Background

The Soil Moisture Active-Passive (SMAP) mission is the successor of the HYDROS mission concept. Its scientific objectives are to provide frequent global maps of the Earth's surface soil moisture and surface freeze/thaw state every 2-3 days, for weather and climate prediction, water, energy and carbon cycle studies, natural hazards monitoring, and national security applications [2].

The SMAP L-band radar requires on-board processing (OBP) to turn 60 MHz high-rate samples into low rate and high rate data ready to be down-linked for further ground processing into radar data products [4].

In particular, each physical receiver channel (H-pol and V-pol) contains three 1 MHz sub-bands corresponding to the co-pol, noise, and x-pol signals. The OBP is essentially a demodulation decimating band-pass channelizer that consists of a quadrature modulation (QM) to move each sub-band to baseband and a three-stage decimation filter (DF) to reduce the sample rate from 60 MHz to a complex (I/Q) output data rate of 1.2 MHz. Details on OBP requirements and algorithm development can be found in [4].

B. ISAAC Newton

ISAAC framework has been applied to the SMAP L-band radar OBP design (codenamed as *Isaac Newton*), including algorithm development and simulation, FPGA design and implementation, and testbench construction and demonstration, as shown in Table 1.

ISAAC Technology	**iBoard**	Xilinx ML410
	iBus	▪ CoreConnect ▪ FIFO
	iCore	▪ Quadrature Demodulation ▪ PolyPhase FIR
	iPackage	▪ BSP ▪ Interrupt-based run-time executive ▪ TCP/IP network stack
	iBench	▪ Synthetic Radar Data ▪ Testbench framework
	iTool	▪ ISE/EDK ▪ AccelDSP ▪ GNU SDE ▪ Matlab

Table 1 ISAAC Newton

Following the digital filter algorithm specification in [4], two IP cores, *QM* and *FIRDEC*, have been designed and implemented under the ISAAC framework. Figure 2 shows the FPGA architecture for one physical receiving channel input that consists of three complex (I/Q) output data.

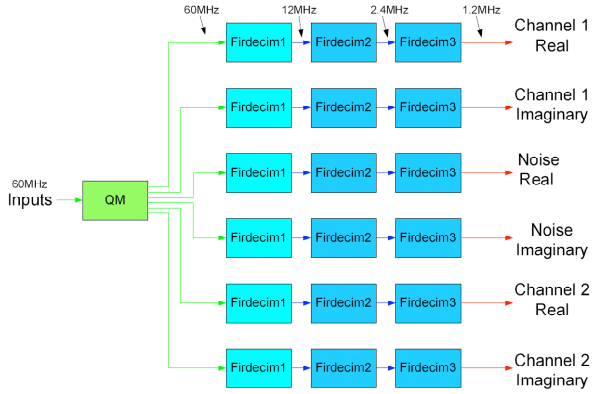

Figure 2 SMAP Digital Filter Core

QM core is implemented using a sine/cosine lookup table and requires 6 multipliers running at 60 MHz.

FIRDEC core is implemented as a three-stage polyphase decimation filter: the first stage is a 15-tap decimation-by-5 filter using 3 multipliers running at 60 MHz; the second stage is a 20-tap decimitation-by-5 filter using 1 multiplier running at 60 MHz; and the third stage is a 50-tap decimitation-by-2 filter using 1 multiplier running at 60 MHz.

Total OBP core requires 36 multipliers running at the 60 MHz input sample rate.

The OBP FPGA core has been synthesized into Xilinx Virtex II and Virtex 4 FPGA devices. Table 2 shows its resource usage, clock speed, and design margin. It can be seen that the Virtex 4 FPGA is capable of running at 112 MHz while the Virtex 2 at about 92 MHz.

	Xilinx XQR2V3000	Xilinx XQR4VFX60
Clock	91.8 MHz	112.3 MHz
	Used / Total	Used / Total
# of Multipliers	36 / 96	36 / 128
# of Slices	10,498 / 14,336	10,533 / 25,280
# of Flips Flops	15,753 / 28,672	15,776 / 50,560
# of 4-input LUTs	11,910 / 28,672	11,817 / 50,560

Table 2 OBP Core Resource Usage

For functional verification and performance validation, the OBP core has been integrated into the *iBench* architecture and programmed into the FPGA on the *iBoard* prototype, a Xilinx ML410 board. Simulated radar data generated in the test station has been produced and fed into the complete testbench system containing the implemented OBP core via the Ethernet. The OBP core, at the prompt of an interrupt, takes the data from the on-chip BRAM into the OBP processing engine, and then sends back the processed data back into the on-chip BRAM, which will then be sent back to the test station, and displayed on a Matlab-based GUI panel software. Figure 3 shows the testbench configuration for SMAP OBP core.

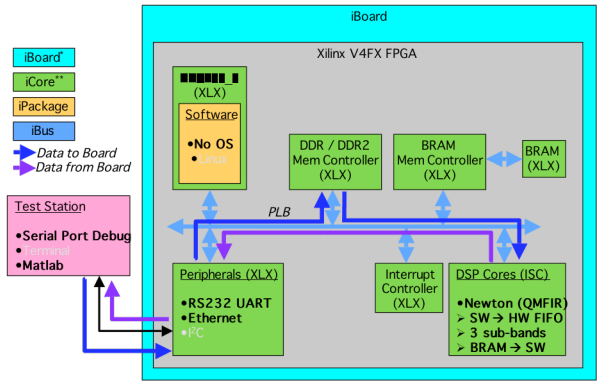

Figure 3 iBench For SMAP OBP

Figure 4 shows the fixed-point error of the OBP core FPGA implementation compared to the floating-point model of the OBP algorithm implemented in Matlab for one receiving channel. It can be seen that all errors are approximately 10^{-4} at this preliminary implementation. Rigorous fixed-point error analysis and measurement will be reported in the future.

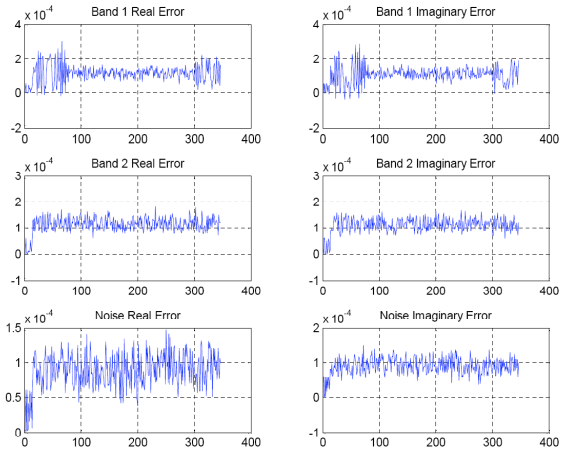

Figure 4 OBP Core Errors

V. SUMMARY AND FUTURE WORK

This paper describes the highly flexible, reusable, and capable ISAAC technology and its application to the onboard processing need of SMAP L-band radar. Its motivation, technical approach, and first prototype ISAAC I are also presented. A demonstrated hardware prototype of SMAP's digital filter was presented and its performance reported.

Future work includes the design of iBoard, and the comprehensive development of the planned capabilities of ISAAC.

ACKNOWLEDGMENT

We would like to thank Pekka Kanglashti, Paula Pingree, Thomas Werne, Will Zheng, and Ian O'Dwyer for numerous technical discussions on the ISAAC framework. We would also like to thank Louise Veilleux, Steven Durden, Kevin Wheeler, Michael Spencer, Wendy Edelstein, Yunling Lou, Samuel Chan, Andy Berkun, and Kyung Pak for their inputs during SMAP radar discussions.

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

REFERENCES

[1] National Research Council, "Earth Science and Applications from Space: National Imperatives for the Next Decade and Beyond", The National Academies Press, 2007.

[2] D. Entekhabi, et. al., "The Hydrosphere State (Hydros) Satellite Mission: An Earth System Pathfinder for Global Mapping of Soil Moisture and Land Freeze/Thaw," *IEEE Trans. Geosci. Remote Sens.,* Vol. 42, no. 10, pp.2184-95, Ocober 2004.

[3] Charles Elachi and Jakob J. van Zyl, "*Introduction to The Physics and Techniques of Remote Sensing"* Wiley-Interscience, 2nd Edition, 2006.

[4] Charles Le, Mike Spencer, Louise Veilleux, Samuel Chan, Yutao He, Jason Zheng, and Kayla Ngyuen, "Algorithm Development for SMAP's Onboard Processor," submitted to *2009 IEEE Radar Conference (RadarCon09),* Pasadena, CA, May, 2009.