A multi-FPGA high performance computing platform for network-centric applications

Tirumale Ramesh, *Senior Member, IEEE* and John Meier
The Boeing Company

Abstract—The current network technologies and market are pushing the envelope for high-performance computing power at the edge of the network. One such technology support for this computational power is the on-the-field reconfigurable technologies. In this paper we present an overview of a highly-parallel Field Programmable Gate Array (FPGA) system architecture called Super Reconfigurable Fabric Architecture (SuRFA[1]) that combines fine-grain FPGA architecture with coarse-grain Single Instruction Multiple Data (SIMD) parallel processing cells (SPEC). We have demonstrated a multi-FPGA based virtual SuRFA implementation for building a fabric architecture that can support vast number of network-centric applications.

Index Terms—intelligent network, sensor fabric, data stream, reconfigurable computing

I. INTRODUCTION

Boeing is a key player in Network-Centric Operations (NCO), in which shared situational awareness is created by networking operational agents based on common operating information and local agents to make quick decisions. To support NCO, next-generation networks impose new challenges to network management. In particular, Quality of Service (QoS) management presents many complexities to efficient network management at the edge, requiring rapid real-time decisions. Enabling adaptive behavioral management is a growing interest today that drives intelligent mechanisms at the edge of the network such as Distributed Policy Management. Processing steps (algorithms) can easily be upgraded [1] over the network using reconfigurable hardware. Handling high throughput data is a requirement for intelligent reconfigurable routers and switches at edge of the network. Network-centric embedded platforms deploy a wide range of applications requiring image and signal processing tasks. For example, sensor network applications such as sensor fusion, segmentation/automatic target recognition (ATR) image processing, hyper spectral imaging, cryptography, and JPEG 2000 generate high volumes of data and require heavy duty processing that are not well suited to use general-purpose processors because of the fixed architecture and often require dedicated custom hardware co-processor to achieve real time performance. Today's embedded architectural trend is to merge general purpose and special instruction sets (IS) to achieve optimal performance for tasks such as correlation [2].

Computing fabrics [3] is one option for implementing distributed edge network architectures. These fabrics refer to an architecture comprised of processing, memories and interconnect fabric scaling from global network to network-on-a-chip. Fabrics have processors that are flexibly-coupled [4] in which they share many of the attributes of a loosely coupled network, including the ability to be disconnected and reconnected dynamically and they allow for creation of an environment where the address space across the processors can appear to be homogenous to the applications. Fabric-based architectures [5] are particularly attractive for efficient implementation in chips to solve specific computational problems, especially in image and array processing. Fabric architectures can be rapidly prototyped into a solution using combination of general purpose processors (GPP), digital signal processors (DSPs) and FPGAs and/or System-on-a chip.

II. RECONFIGURABLE FABRIC

Reconfigurable fabrics are becoming an enabling information processing platform for many network applications by massively integrating computing fabric on a chip. A new approach to configware reconfigurable computing architecture is being explored today [6]. These coarse-grain arrays of reconfigurable data paths units can offer drastic reduction in reconfigurable time overhead to directly configure high-level parallelism similar to ILP in a conventional processor. SuRFA is a fabric oriented virtual configware architecture with internal VLIW control with slots for configuration and floware control for a given application. Floware is data-stream-driven. Here the definition of the term data streams is adopted from the systolic array scene, which defines, at which time which data item has to enter or leave which port. SuRFA can be used in a hybrid computing implementation along with a general processor or a DSP.

This multi-FPGA architecture has been defined component-based architecture with vendor independent component-to-component interface interconnection and facilitates virtual bus for mapping from any I/O standard bus permitting adaptation to high performance augmented hybrid solution systems or as core architecture within an on-chip embedded computing framework. The platform is designed to have high-end data processing capability for signal and image processing, multimedia, cryptography and wireless communication applications. Smart adaptive processors can be embedded into the fabric of SuRFA. In other words, the fabric will become a generalized Smart Connector that can be configured for intelligent processing and data communication for wide varieties of high performance computing solutions.

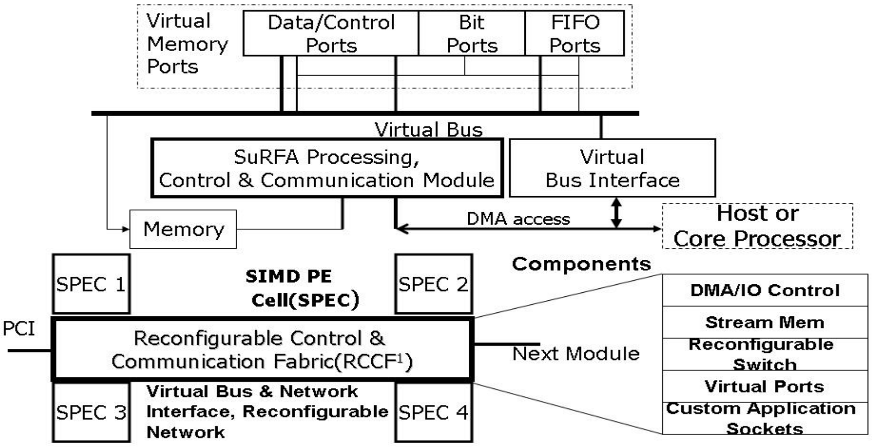


Figure 1 SuRFA Platform

Figure 1 shows a scalable SuRFA platform which is one candidate for reconfigurable fabric solutions. The SuRFA currently designed with two 64 x 64 input FIFOs and two 64 x 64 output for stream buffering to maximize the streaming I/O bandwidth. The FIFO is organized with 16 x 64 FIFO blocks. Each SuRFA SIMD Parallel Processing Cell (SPEC) is a 2 x 2 array of simple *n*-bit and powerful coarse-grain SIMD (Single Instruction Multiple Data) reconfigurable elements with built-in high-speed connectivity built with fine grain logic placed in each. Each cell is connected to its neighboring cell through a reconfigurable fabric (RCCF- Reconfigurable Control & Communication Fabric). RCCF controls the data path unit of cell elements. To make the solution independent of I/O standards and hardware vendors, standard bus protocol is mapped to the Virtual Bus Interface (VBI) via a Virtual Memory Ports (VMP). There are three different types of VMP namely 1) Data/control port that interfaces with any application port for sending and receiving data, 2) Bit controllable port for FPGA application 1-bit variables, and 3) FIFO port for interfacing with internal FIFO (on-chip) for data. Each 64-bit port configuration of VMP is mapped to an address space and are configurable as VBI signal groups as long (64-bit), half (32-bit), (16-bit) or byte (8-bit) data. SuRFA floware control is based on distributed Very Long Instruction Word (VLIW) controller. A configuration memory is built into RCCF for configuration of data path widths, pipeline stages within SPEC elements.

Table I shows how SuRFA compares with some of the other configurable/reconfigurable platforms.

Table 1 SuRFA Comparison to Other Known RCs

System	**Granularity**	**RC Type**	**Use**	**Computational Model**	**Application Domain**
PADDI[7]	Coarse-grain	Static	Remote	VLIW/SIMD	DSP
MorphoSys [8]	Coarse-grain	Dynamic	Local	SIMD	Data-Parallel(CI)
RaPiD[9]	Coarse-grain	Mostly Static	Remote	Pipelined	Systolic/CI
SuRFA	Mixed-grain	Dynamic	Flexible	Streaming/SIMD	Data-Parallel(CI)

PADDI (Programmable Arithmetic Device for DSP) [7] supports computation-intensive DSP data paths. It consists of clusters of arithmetic execution units (EXUs) connected to a central crossbar switch box. MorphoSys [8]) has a MIPS-like “TinyRISC” processor with extended instruction set; and is divided into four quadrants of 4 by 4 16 bit RCs each. RaPiD: The Reconfigurable Pipelined Datapath (RaPiD) [9] aims at speed-up of highly regular, computation-intensive tasks by deep pipelines on its 1-D RC. To implement I/O streams RaPiD includes a stream generator with address generators, optimized for nested loop structures, associated with FIFOs. SuRFA is a multi-FPGA virtual architecture. We mapped SuRFA on to a Virtex-2 6000 four-chip DINI board [10].

Figure 2 shows the SuRFA virtual interface that includes the physical mapping of the SuRFA with an illustration of its interface to standard PCI open core [11]. SuRFA runs on the component-to-component interface not exclusively included in this paper. The component interface is mapped into external I/O virtual interface for standard I/O bus independent implementation and into virtual memory ports for vendor board independent mapping of application I/O. The virtual design tool is an interface for device vendor specific mapping.

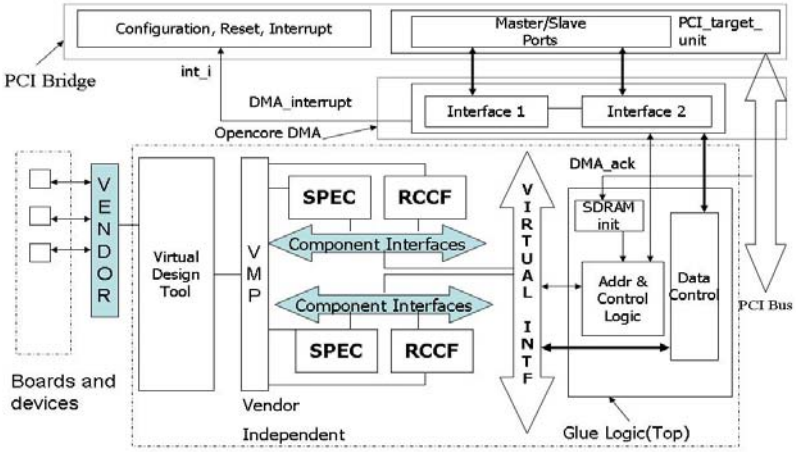
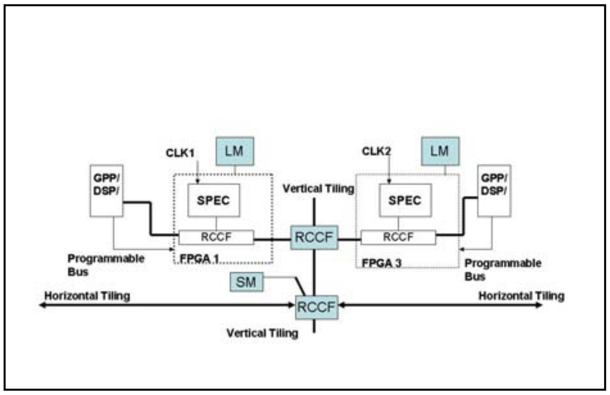


Figure 2 SuRFA Interface

III. MULTI-FPGA IMPLEMENTATION

Figure 3 shows an illustrative generic multi FPGA hybrid mapping and an illustrative mapping on a virtex-2 6000 multi-FPGA board is shown in Figure 4.



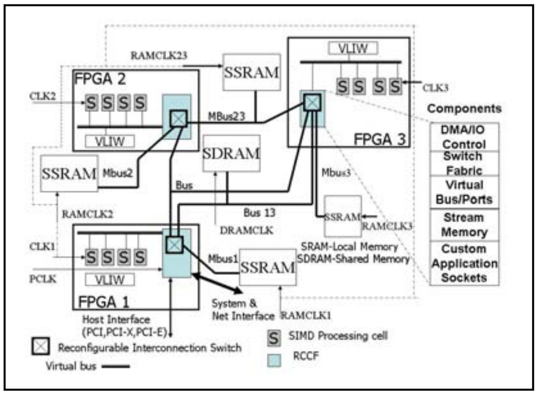


Figure 3 SuRFA Hybrid System

Figure 4 Multi-FPGA Board

“Approved for release to foreign persons. Export Log#IDS-06-DWS272”

SuRFA is used as a validating and rapid prototyping platform for extended formulation of a more generalized fabric model. We applied a graph theoretical model [4] for system shown in Figure 1 for mapping distributed application onto the hybrid system. The use of the reconfigurable fabric is to provide reconfigurable connectivity from one GPP to another. For establishing a path within an RCCF network, we configure the switches within RCCF. By applying SIMD array switch simulation model and mapping function[4] for dynamic reconfiguration at run-time for connectivity to the mapping for RCCF switch elements in a SuRFA hybrid system for a set of entities communicating from one partition of the graph to the other. The total switch availability is shown in Table I.

Table I
Fabric connections availability

Size-n (# of GPP/DSP)	Required Communication Channels[4]	Fabric Connections Available	# of RCCF run-time reconfigurations
8	15	13.84	2
16	30	29.85	1
24	45	41.06	4

Partial reconfiguration is a design process, which allows a limited, predefined portion of an FPGA to be reconfigured while the remainder of the device continues to operate. This is especially valuable where devices cannot be disrupted as per application demand while some subsystems are being redefined. Partially reconfigurable processors though provide a unique ability to directly affect the overall system throughput by reducing reconfiguration bandwidth. In a large fabric computing with a need for partitioned design modules identified for partial reconfiguration, there will be number of reconfigurations of the fabric adding to a large overhead. However, overlapping execution of one temporal partition with partial reconfiguration of another [12] can reduce the overhead time. But, to identify this temporal parallelism for execution and partial reconfiguration requires theoretical models which are beyond the scope of this paper.

IV. VIRTUAL ARCHITECTURE SIMULATION, PERFORMANCE ESTIMATION & APPLICATION

There are vast potential applications in aerospace and defense community such as Airport Baggage Scanning, Biometrics, Homeland Security, Cryptography and High-end Signal and Image Processing needed for data processing and dissemination. Some typical data-intensive applications in spacecraft are filtering, image compression/decompression, object tracking and detection. Of particular interest in compute-intensive processing off-load for space & defense systems is Discrete Wavelet Transform (DWT) algorithm, which has shown significant improved performance for JPEG 2000 compression.

The SuRFA is currently laid out as performing 64-point ordered data FFT on each SPEC (16-point per PE) with data stream from 16 x 64 FIFOs communicated with each 4-point butterfly. This requires k=n/64 streams for an n-point FFT on each SPEC. The initial cycle takes 16 clock cycles to fill the FIFO and subsequent FFT streams takes k cycles. The FIFOs are filled continuously from a data storage FIFO to each of the 16-point stream buffer.

We ran a verilog simulation of an n-point FFT on a SuRFA virtual architecture that would compute the result in approximately 32 + n/64 clock cycles per SPEC. We also extended the split & merge region based operation and used a template image (identifier) to correlate a source image using a modified back track split & merge method [13]. Table II summarizes all the performance estimation for each of the identified functions and the number of SuRFA resource utilization. All our results are very preliminary at this stage. The FIFO required are based on keeping up with a 500 MHz FPGA clock streaming throughput (assumed an Virtex-5 Xilinx) for a PCI-X, 64 bit, 133 MHZ standard I/O interface.

The template image (identifier) is correlated with source image using a back track split & merge method [13]. Table II summarizes all the performance estimation for each of the identified functions and the number of SuRFA resources needed.

Table II SuRFA Performance (Estimated)

Function, Size	**Other application cores**	**SuRFA ,500 MHz w/ 1 SPEC@64-pt BF) (ordered data)**	**Surf Resources per SPEC**		
			FIFO	**MULT**	**ADD/SUB**
FFT, 1024	~160 ns [Ref 14]	136 ns	64*	96*	80*
FFT, 2048	~350 ns [Ref 14]	200 ns	256**	404**	320**
Conv ,512 x 512	~1.5 ms[15]	~ 0.1 ms	64*	80*	72*

** PCI-X, 64 bit, 133 MHZ (Assume 500 MB/s bandwidth) ** PCI-X, 64 bit, 533MHZ (Assume 2 GB/s bandwidth)*

SuRFA application in edge computing

Decentralized peer-to-peer application processing has potential to be the driving force for distributed sensor solutions using next-generation switched fabric architectures. A purely components-based system approach streamlines application development while adding flexibility and scalability. A component-based system approach allows independent subsystems and functional units to be developed and tested separately to allow seamless system integration. Edge computing [18] is one candidate for supporting this peer-to-peer application in which edge nodes are mapped within a physical independent aware computing fabric.

V. CONCLUSIONS

In this paper, we have shown a multi-FPGA based reconfigurable computing fabric architecture. The fabric allows for scaling from system to chips making it more reusable from system built with several board-level components to a chip with integrated components. We envision SuRFA as a major enabler of edge network-computing embedded systems for gaining additional flexibility, lower cost and power.

References

[1] Sproull, T.S.; Lockwood, J.W.; Taylor, D.E., "Control and configuration software for a reconfigurable networking hardware platform," Field-Programmable Custom Computing Machines, 2002.

[2] Todd S. Sproull, John Meier, et. al. “Sensor Fusion and Correlation”, ACM SenSys 2005, San Diego, CA, 11/05.

[3] Wolinski, C.; Gokhale, M.; McCave, K;” A polymorphous computing fabric”, Micro, IEEE, Volume 22, Issue 5, Sept.-Oct. 2002 Page(s):56 – 68.

[4] Tirumale Ramesh, “Reconfigurable Flexibly Coupled Multiprocessor for Parallel Computing,” Ph.D. dissertation, Dept. Computer. Eng., Oakland Univ. MI, 1993., OCLC: 29323502

[5] David Gewirtz, "Massively distributed computing using computing fabric” [online], 1998, available from World Wide Web: http://www.dominopower.com/issuesprint/issue199810/fabric.html

[6] Configware:http://www.configware.org

[7] D. Chen and J. Rabaey: PADDI: Programmable arithmetic devices for digital signal processing; VLSI Signal Processing IV, IEEE Press 1990.

[8] H. Singh, et al.: MorphoSys: An Integrated Re-configurable Architecture; Proceedings of the NATO RTO Symp. On System Concepts and Integration, Monterey, CA, USA, April 20-22, 1998.

[9] C. Ebeling et al., RaPiD: Reconfigurable Pipelined Datapath”, in [17]

[10] http://www.dinigroup.com

[11] PCI wishbone core: http://www.opencores.org

[12] M.Kaul and Ranga Vemuri, “Temporal Partitioning combined with Design Space Exploration for Latency Minimization of Run-lime Reconfigured Designs”, Design and Test in Europe, DATh 98, IEEE Computer Society, Paris, 1998 pp.389-396.

[13] Tirumale Ramesh, “Region growing on a clustered reconfigurable bus system”, Proceedings of the International Conference on Parallel & Distributed Processing Techniques and Applications, Page 530-537, July 1997.

[14] http://www.dilloneng.com

[15] Cloutier, J.; Cosatto, E.; Pigeon, S.; Boyer, F.R.; Simard, P.Y, “VIP: an FPGA-based processor for image processing and neural networks”, Microelectronics for Neural Networks, 1996, Proceedings of Fifth International Conference,12-14 Feb. 1996 Page(s):330–336, Digital Object Identifier 10.1109/MNNFS.1996.493811

[16] http://www.xilinx.com

[17] M. Glesner, R. Hartenstein (Editors): Proc. FPL’96, Darmstadt, Germany, Sept. 23-25. 1996. LNCS 1142. Springer Verlag 1996.

[18] John Meier and Tirumale Ramesh, “Intelligent and Reconfigurable Edge of the Network Computing- Reaping the benefits by moving applications to the network”, 2006 Military & Aerospace PLD conference (MAPLD), September 2006.

Acknowledgement

This work for SuRFA application for edge computing was conducted as part of an edge network computing IRAD project funded by Boeing Phantom Works and also supported by Boeing IDS.

Author's Biography

Tirumale K Ramesh is currently a Senior Scientist and Associate Technical Fellow with the Boeing Space & Intelligence Systems. He has over 25 years of professional experience specifically working in the area of parallel architectures, embedded computing systems, reconfigurable architectures and networks, and system-on-a chip level architecture and solutions. At Boeing, he is currently involved with several key technical activities including leading a major R&D project on high performance embedded intelligent peer-to-peer network computing. Prior to joining Boeing, he has served as Technical Lead at Lockheed Martin and IBM Microelectronics. Dr. Ramesh is a Senior Member of IEEE and IEEE Computer Society and serves on the executive committee of the IEEE Computer Society Technical Activities Board. He served on the technical committees of Military & Aerospace PLD conference (MAPLD 2006) and is presently serving on the program committee of the Mini-Symposium on HPC at HiPC 2006 conference. Dr. Ramesh is widely published and has several patents pending in the area of reconfigurable and embedded computing. Dr. Ramesh earned his BSEE from Bangalore University, India, MSEE from Mississippi State University with concentration in VLSI and a Ph.D degree in Computer Engineering from Oakland University, Michigan.

John L Meier is a Boeing Technical Fellow in the Network Centric Thrust at Phantom Works, NCO Thrust. He has over 27 years of professional experience in avionic technology development specifically working in the area of intelligent networking, reconfigurable computing architectures and wireless network management. At Boeing, he is currently involved with several key technical activities including a major project on edge computing and intelligent distributed system management. Mr. Meier earned his BS from Southern Illinois University - Carbondale (SIU-C), MS University of Missouri - Rolla (UMR), and currently working on his PhD degrees from the Department Computer Science Engineering (CSE) at the Washington University in St. Louis. He is a member of IEEE and Tau Beta Pi.