

A Many-Core Platform Implemented for Multi-Channel Seizure Detection

Jordan Bisasky, Darin Chandler Jr., and Tinoosh Mohsenin
Dept. of Computer Science & Electrical Engineering
University of Maryland, Baltimore County

Abstract—This paper presents a reconfigurable many-core platform performing fixed point DSP applications supporting up to 64 cores routed in a hierarchical network. To demonstrate an application, electroencephalogram (EEG) seizure detection and analysis is mapped onto the cores. The individual cores are based on a 5 stage RISC pipeline architecture optimized to support communication to other cores on the platform. To reconfigure the platform, programs are loaded onto each of the cores. Communication between cores is implemented using low-area routers that partitions computational cores into hierarchical clusters resulting in a low network diameter. The routers use a packet-switched protocol that minimizes circuitry which further reduces circuit size in comparison to the computational circuitry. A globally asynchronous, locally synchronous (GALS) architecture is implemented to eliminate global clock routing which consumes high levels of power due to long propagation and thus high capacitive loading from many cores. Additionally, cores not configured for an application has its local clock disabled which turns off unused cores. The overall result is a platform with lower power consumption than a traditional single core DSP with the reconfigurability lacking in an ASIC. Applications tested within the mapping include the Fast Fourier Transform (FFT) and Finite Impulse Response (FIR) filter. The seizure detection and analysis algorithm, when mapped onto the many-core platform, takes 5663 cycles to execute in 14.45 μ s. The prototype SoC is implemented in 65 nm CMOS which contains 64 cores and occupies 8.41 mm².

Index Terms—65 nm CMOS, DSP, many-core, biomedical, seizure detection

I. INTRODUCTION

With constraints in today's DSP applications, there is a greater need for a low power, low area, high speed platform. Types of applications vary widely from communication error correction to portable medical devices. With limits to the maximum frequency due to energy consumption concerns, the efficiency of an algorithm now relies on parallelization which is a fundamental limitation of general-purpose processors. Traditional single core DSPs and FPGAs carry some advantages in performing DSP applications, but both also have their limitations. Traditional single core DSP processors are not suited for parallelization whereas FPGAs are cumbersome to program. ASICs are the optimal platform in terms of area, speed, and power, but the long development time and high manufacturing costs are prohibitive. A growing area of interest is using many-core platforms to bridge the gap between ASICs and FPGA/DSP processors.

In this paper, we propose a many-core platform supporting 64 low power modified RISC cores with an emphasis on low

power and low area. Communication between cores or nodes is accomplished via a simple, scalable hierarchical network that reduces the number of hops in communication compared to a flat topology. Common kernels including Fast Fourier Transform (FFT), Finite Impulse Response (FIR) filters and sort are used for performance evaluation. To reduce area and power, each core has minimum pipeline stage and instruction and data memory. Despite the restriction in memory, performing fixed point DSP is possible through fine grained task level parallelization of the cores. To demonstrate an application of the platform, the cores are programmed to implement seizure detection and analysis.

The following sections begin by discussing previous work that lead to the proposed architecture. Then, the architecture and its features are described including simulations results from the seizure detection and analysis application mapping. Finally, the implemented hardware is described and discussed.

II. BACKGROUND

A. Many-core Architecture and Routing

There have been advancements in many-core applications necessitated by the increased need for low power DSP applications. Previous work has varied in both the design of the processing core and the design of the network interconnect. Core designs have comprised of a wide range of architectures varying in instruction sets, pipeline design, and complexity. SmartCell uses a simplified four stage pipeline limited to a low number of instructions [1]. However, the 64 bit instruction word in Smartcell and VLIW architectures [2] [3] results in a larger memory and overall area. Another core design uses a more complex architecture including a floating point block and built in self testing [4]. However, many applications using FFT can achieve the desired precision using fixed point instead of floating point.

Network topologies have varied equally as much. AsAP [5] and Morphosys [6] use a flat topology where the nodes are arranged in a 2D network. An advantage of a flat topology is simple routing circuitry that leads to a small router area. Alternatively, SmartCell [1] partitions cores into clusters, where cores within a cluster can communicate directly and distant clusters use routers to communicate. This type of topology and other hierarchical networks [7] [8] [9] [10] generally lowers the network diameter or number of communication hops but increases routing complexity [11]. This may result

in more hardware and thus higher area occupation and power dissipation.

B. EEG Seizure Detection and Analysis

The primary tool for diagnosis of an epileptic seizure is an electroencephalogram (EEG) which measures brain activity. Detection and analysis requires the placement of a minimum of 16 electrodes on the scalp with each electrode being a channel. Previously proposed algorithms and implementations have targeted low power, portable detection in a non-clinical setting [12] [13] [14] [15]. The proposed work has been implemented on the many-core platform combining seizure detection and analysis [16] to increase programmability. Detection is performed in the time domain by comparing EEG inputs to a predetermined threshold value. A seizure is only detected if there are multiple inputs over the threshold value in order to remove random noise in the data. After detection of a seizure, the analysis is performed by converting to the frequency domain and separating the energy of the data into four frequency bands - Theta (4-7 Hz), Alpha (8-12 Hz), Beta (13-29 Hz). The results are transmitted to an external device to determine the type and location of the seizure.

III. PROPOSED ARCHITECTURE

A. RISC Core

Each core in the proposed architecture consists of a modified five stage RISC pipeline for executing the instructions. The stages are Fetch (IF), Decode (ID), Execute (EX), Memory (MEM), and Write Back (WB). The other hardware blocks are instruction memory, data memory, and input/output FIFO's as seen in Fig. 1. The instruction memory stores up to 128 words and the data memory stores up to 128 16-bit words. Additionally, there are 15 registers that each store 16 bits of data. Finally, the input/output FIFOs each store 16 words. To reconfigure the cores, an assembly code is used to write programs for each core. The instructions are assembled to 17-bit words and are loaded onto each core's memory. Each instruction is 17 bits consisting of 5 bits identifying the operation and the remaining 12 bits are used for input and output reference. Instructions include memory load/store, ALU operations, conditional branching, and core-to-core communication.

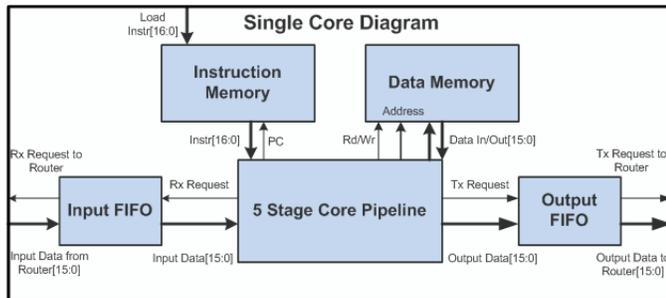


Fig. 1. Single core block diagram consisting of the 5 stage pipeline, instruction and data memory, and input/output FIFO's.

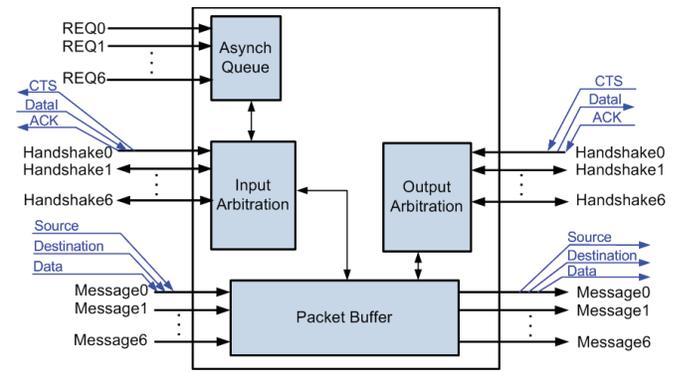


Fig. 2. Block diagram of the cluster router.

B. Difficulties for Network Routing

One challenge with many-core implementations is sending data long distances across the platform. Some implementations are more efficient than others in terms of speed over distance. For example, the worst case number of hops in a 2-D mesh network (side-to-side) is $(N - 1)$ where N is the network diameter. Conversely, in a hypercube network every node is at most $\log_2 N$ hops away from another node. There are, however, tradeoffs with each implementation. Whereas the 2-D router node is very simple, its performance does not scale well to large applications. A hypercube network is fast but the router and wire complexity is much higher. The proposed architecture targets metrics such as scalability, low hardware overhead, and energy efficiency.

C. Network Architecture

The proposed network architecture takes inspiration from the 4-ary tree architecture [17]. The advantage of a tree network is that the worst-case message is $(\log_2 N - 1)$ hops away, where N is the number of network nodes. For a tree architecture, leaf nodes must talk up in hierarchy in order to communicate with any other node. Since most applications on a many-core chip tend to map to a local region, bursting with only one path to all other nodes creates severe bottleneck. Our architecture cures this problem because any core in a cluster of four can communicate to any other core within its cluster. This alleviates bottlenecking at a parent router. Routers are only necessary when a core must send a message outside of its own cluster.

D. Network Implementation

The network is designed with the potential to run any core at a different clock rate through a globally asynchronous, locally synchronous (GALS) architecture [18]. This eliminates global clock routing which consumes high levels of power due to long propagation and thus high capacitive loading from many cores. Using the GALS paradigm, cores not configured for an application would have its local clock disabled which turns off unused cores. Each core is wrapped with an input FIFO and an output FIFO creating an asynchronous interface to other cores on chip. Unlike the design in [5] where the source clock

is shared to another core's FIFO, this design hides the clock and uses asynchronous 4-phase handshaking. Metastability in request signals is dealt with by using an asynchronous queue implemented by adapting a priority encoder and registers to an asynchronous design. Other handshaking signals and data are synchronized using flip-flops, as seen in Fig. 2. These circuits are simple and translate to a router that occupies very little area. This strategy of asynchronous communication eliminates the need to share a clock which is a major issue with clock tree complexities rising as CMOS technologies shrink [19].

The information necessary to successfully send a message is the ID of the sender, the ID of the destination core, and the data. This information is immediately stored into an output FIFO. As soon as the information is entered into the buffer, the buffer attempts to send the message toward the destination. If the message is destined to a core outside of the sender's cluster, a router is the next hop. The router is implemented with an input queue and a buffer. The input queue solves the issue of simultaneous requests and contention—nodes will have messages passed in the order in which they request to send. The buffer protects against a situation when the router receives messages faster than it can send them.

The routing pattern in this network is statically wired—it will always send messages from one node to another through the same path. The benefit of this design is that the routing is transparent to the programmer—the programmer states where the message should go and the hardware executes the request.

IV. MAPPING EEG SEIZURE DETECTION AND ANALYSIS

To demonstrate the functionality of the many-core platform for fixed point DSP applications, the multi-channel seizure detection and analysis is mapped onto the platform. The low power seizure detection architecture was initially proposed by the authors and more details are found in [16]. Fig. 3 depicts the high level block diagram of the hardware block. Seizure data is converted from analog to digital across the 16 channels and is serially passed to the many-core platform. Each channel has its own dedicated seizure detection block which also includes a 33 tap high pass FIR filter to remove any DC offset. The detection block compares data to a preset threshold. If the input data is greater than the threshold more than once in a preset time period, then the channel flags the detection of a seizure to the multi-channel detection block. Upon detection, the seizure data is passed to the 128-point FFT for frequency analysis, and energy calculations in the four frequency bands are performed. The 128-point FFT, as depicted in Fig. 4, is divided into two main stages. The first stage consists of eight 16-point FFT's while the second stage consists of sixteen 8-point FFT's. Between the two stages, there is address shuffling to write the data points to the appropriate memory location for the next FFT. After the FFT is completed, the bands are filtered. To compute the energy in each band, the real and imaginary components at each frequency are squared and summed. Then the energies of the frequencies are summed and outputted to an external device for storage. Table I summarizes the cycle counts of each of the

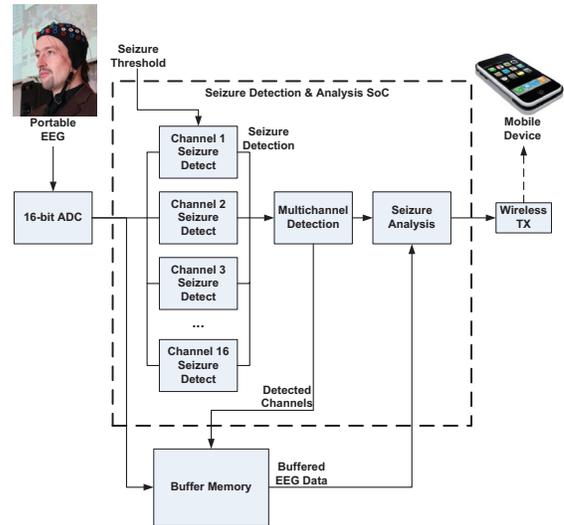


Fig. 3. Proposed multi-channel seizure detection architecture. 16 single channel detection circuits are instantiated and passed to a threshold detector to confirm a seizure. Subsequently, analysis circuitry is enabled.

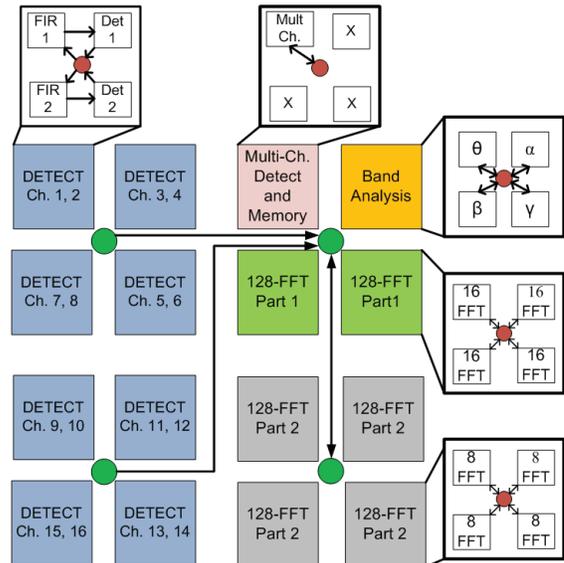


Fig. 4. Mapping of the seizure detection and analysis hardware block onto the many core platform. The implementation supports 16 EEG channels mapped onto 61 cores.

blocks used to map the seizure detection and analysis onto the many-core platform. The seizure detection block completes in $14.45 \mu\text{s}$ at the maximum clock frequency on 392 MHz based on the implementation results from Table II.

V. CMOS IMPLEMENTATION AND RESULTS

The many core platform is implemented in 65 nm CMOS technology with a nominal supply voltage of 1.0 V. We used a standard-cell RTL to GDSII flow using synthesis and automatic place and route. The hardware was developed using Verilog to describe the architecture, synthesized with Cadence RTL Compiler, and placed and routed using Cadence SOC Encounter. Fig. 5 shows the layout of a single core and a

TABLE I
CYCLE COUNT RESULTS FOR SEIZURE DETECTION AND ANALYSIS.

Application	Cycle Count
128-Pt. FFT	2511
8-Pt. FFT	524
16-Pt. FFT	1,867
33 tap FIR Filter	631
Energy Band MAC	130
Total	5663

TABLE II
IMPLEMENTATION RESULTS

Technology	65 nm, 1V
Logic Utilization	96%
Processor Area	280 μm x 280 μm
Router Area	110 μm x 110 μm
Processor Speed	392 MHz
Router Speed	405 MHz
Processor Power	15.7 mW*
Router Power	1.9 mW*

*All results are from place & route except power which is from synthesis

single router. Table II summarizes the post-layout results. The prototype design routes 64 cores divided into 16 clusters. Every four clusters is connected to a router and every four routers connects to a router in a similar fashion. A single core occupies 0.078 mm² and each router occupies 0.012 mm² and the entire prototype design results in a total area of 8.41 mm². The total area of a many core platform is $[2^n \times L_{core} + (2^n - 2) \times L_{router}]^2$, where $n \geq 1$ (heirarchy level), L_{core} is length of one side of a core, and L_{router} is the length of one side of a router.

VI. CONCLUSION

This paper presents the design and implementation of a many-core platform capable of performing DSP applications including seizure detection and analysis. The low area, low power, high speed single-core processors perform the DSP computations when programmed in assembly language. The core processors are connected in clusters which are networked by routers to support parallel processing. As proof of concept, seizure detection and analysis is programmed and mapped onto

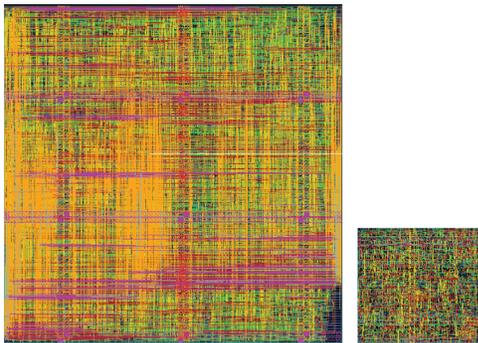


Fig. 5. Size comparison of one core (left) and one router (right).

the processor. The initial prototype maps 64 cores onto the network-on-chip with an area of 8.41 mm² at 1.0 V and 65 nm CMOS.

REFERENCES

- [1] C. Liang and X. Huang, "Smartcell: A power-efficient reconfigurable architecture for data streaming applications," in *Signal Processing Systems, 2008. SiPS 2008. IEEE Workshop on*, oct. 2008, pp. 257–262.
- [2] T. Wada, S. Ishiwata, K. Kimura, K. Nakanishi, M. Sumiyoshi, T. Miyamori, and M. Nakagawa, "A vliw vector media coprocessor with cascaded simd alus," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 17, no. 9, pp. 1285–1296, sept. 2009.
- [3] K. Kim, S. Lee, J.-Y. Kim, M. Kim, and H.-J. Yoo, "A configurable heterogeneous multicore architecture with cellular neural network for real-time object recognition," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, no. 11, pp. 1612–1622, nov. 2009.
- [4] G. Zhong, F. Xu, and J. Willson, A.N., "A power-scalable reconfigurable fft/ift ic based on a multi-processor ring," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 2, pp. 483–495, feb. 2006.
- [5] D. Truong *et al.*, "A 167-processor computational platform in 65 nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 44, no. 4, pp. 1130–1144, Apr. 2009.
- [6] J. H. Bahn, J. Yang, and N. Bagherzadeh, "Parallel fft algorithms on network-on-chips," in *Information Technology: New Generations, 2008. ITNG 2008. Fifth International Conference on*, april 2008, pp. 1087–1093.
- [7] D. Gohringer, O. Oey, M. Hubner, and J. Becker, "Heterogeneous and runtime parameterizable star-wheels network-on-chip," in *Embedded Computer Systems (SAMOS), 2011 International Conference on*, july 2011, pp. 380–387.
- [8] F. Sibai, "A two-dimensional low-diameter scalable on-chip network for interconnecting thousands of cores," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 2, pp. 193–201, feb. 2012.
- [9] A. Bouhraoua and M. Elrabaa, "Addressing heterogeneous bandwidth requirements in modified fat-tree networks-on-chips," in *Electronic Design, Test and Applications, 2008. DELTA 2008. 4th IEEE International Symposium on*, jan. 2008, pp. 486–490.
- [10] P. Sahu, N. Shah, K. Manna, and S. Chattopadhyay, "An application mapping technique for butterfly-fat-tree network-on-chip," in *Emerging Applications of Information Technology (EAIT), 2011 Second International Conference on*, feb. 2011, pp. 383–386.
- [11] Y. Salah, M. Atri, and R. Tourki, "Design of a 2d mesh-torus router for network on chip," in *Signal Processing and Information Technology, 2007 IEEE International Symposium on*, dec. 2007, pp. 626–631.
- [12] S. Raghunathan, S. K. Gupta *et al.*, "A hardware-algorithm co-design approach to optimize seizure detection algorithms for implantable applications," *Journal of Neuroscience Methods*, vol. 193, no. 1, pp. 106–117, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0165027010004504>
- [13] S. Sridhara, M. DiRenzo *et al.*, "Microwatt embedded processor platform for medical system-on-chip applications," *Solid-State Circuits, IEEE Journal of*, vol. 46, no. 4, pp. 721–730, april 2011.
- [14] N. Verma, A. Shoeb *et al.*, "A micro-power eeg acquisition soc with integrated feature extraction processor for a chronic seizure detection system," *Solid-State Circuits, IEEE Journal of*, vol. 45, no. 4, pp. 804–816, april 2010.
- [15] N. Salleh, K. Lim *et al.*, "Ar modeling as eeg spectral analysis on prostration," in *Technical Postgraduates (TECHPOS), 2009 International Conference for*, dec. 2009, pp. 1–4.
- [16] D. Chandler, J. Bisasky, J. Stanislaus, and T. Mohsenin, "Real-time multi-channel seizure detection and analysis hardware," in *Biomedical Circuits and Systems Conference (BioCAS), 2011 IEEE*, nov. 2011, pp. 41–44.
- [17] N. K. Kavaldjiev, "A run-time reconfigurable network-on-chip for streaming dsp applications," Ph.D. dissertation, Enschede, 2006. [Online]. Available: <http://doc.utwente.nl/57687/>
- [18] M. Krstic, E. Grass, F. Gurkaynak, and P. Vivet, "Globally asynchronous, locally synchronous circuits: Overview and outlook," *Design Test of Computers, IEEE*, vol. 24, no. 5, pp. 430–441, sept.-oct. 2007.
- [19] A. Iyer and D. Marculescu, "Power and performance evaluation of globally asynchronous locally synchronous processors," in *Computer Architecture, 2002. Proceedings. 29th Annual International Symposium on*, 2002, pp. 158–168.