A 167-processor 65 nm Computational Platform with Per-Processor Dynamic Supply Voltage and Dynamic Clock Frequency Scaling

Dean Truong, Wayne Cheng, Tinoosh Mohsenin, Zhiyi Yu, Toney Jacobson, Gouri Landge, Michael Meeuwsen, Christine Watnik, Paul Mejia, Anh Tran, Jeremy Webb, Eric Work, Zhibin Xiao, Bevan Baas

University of California, Davis
{hottruong,bbaas}@ucdavis.edu

Abstract

A 167-processor 65 nm computational platform well suited for DSP, communication, and multimedia workloads contains 164 programmable processors with dynamic supply voltage and dynamic clock frequency circuits, three algorithm-specific processors, and three 16 KB shared memories, all clocked by independent oscillators and connected by configurable long-distance-capable links.

Keywords: many-core, DSP, DVFS, GALS.

Overview

Recent chips containing a large number of programmable processing elements are proving to be efficient and flexible platforms for computing DSP and multimedia applications [1-3]. Several significant remaining challenges include: reducing power dissipation of processors when lightly loaded or unused, achieving low energy while maintaining performance on common demanding tasks such as FFTs and Viterbi decoders, providing access to shared on-chip memories, and efficiently communicating between processors.

The computational platform shown in Fig. 1 efficiently addresses the aforementioned challenges. Similar to the AsAP chip [1], each of the simple programmable Globally Asynchronous Locally Synchronous (GALS) processors contains a 16-bit datapath with a 40-bit accumulator and an independent haltable local clock oscillator. The remainder of this paper presents new features.

The homogeneous array of 164 programmable processors reduces design effort and enables applications to be mapped in multiple ways for yield enhancement and self-healing. To achieve even greater energy/performance for three common tasks, dedicated processors designed for signal processing (Fast Fourier Transform (FFT)), communication (Viterbi decoder), and multimedia (video motion estimation) tasks are integrated into the array. Three 16 KB shared memories provide high speed storage to the array and support port priorities, port request arbitration, and multiple addressing modes including programmable address generators [5]. The dedicated-purpose processors and shared memories contain independent individual local oscillators, and communicate through dual-clock FIFOs [6] to seamlessly communicate with the array.

A. Homogeneous Processors: Each processor has a 128x35-bit instruction memory, 128x16-bit data memory, and two 64x16-bit FIFO memories. The in-order, single-issue, six-stage RISC pipeline shown in Fig. 2 executes over 60 basic instructions including new: byte-add/sub, min/max, absolute value, jump/return, and new features such as conditional execution; zero overhead looping, and block floating point. A CORDIC square root task executes in 216 cycles on this processor compared to 628 cycles on AsAP [1]—a 2.9x speedup.

B. Dedicated-Purpose Processors: The FFT processor is capable of dynamically switching between 16 to 4096-point complex FFT/IFFT transforms with a continuous throughput of one complex radix-4 or radix-2 butterfly per cycle. The configurable Viterbi processor contains 8 ACS units and can decode codes up to constraint length 10. The Motion Estimation processor supports several fixed and programmable search patterns, all H.264-specified block sizes, and computes over 14 billion SADs/sec at 880 MHz.

Intra-chip Communication

Processor tiles communicate through configurable nearest-neighbor and long-distance links. Connections are circuit-switched and statically configured resulting in very low overhead and are well suited for GALS clocking. Fig. 4 shows the interconnect structure including the processor's two inputs and single output which is dynamically maskable to the tile's eight outputs. Each link contains a 16-bit data bus, a source-synchronous clock, a valid signal, and a reverse-direction request signal used for flow control. Limited only by clock duty cycle distortions, links may be configured to pass data across the chip in a dedicated channel without disturbing intermediate processors and without regard to their clock or voltage domains. The data can be pipelined at each tile to achieve full-rate communication over long distances, or un-pipelined if the distance is short or the source clock's frequency is low, resulting in lower total communication latency.

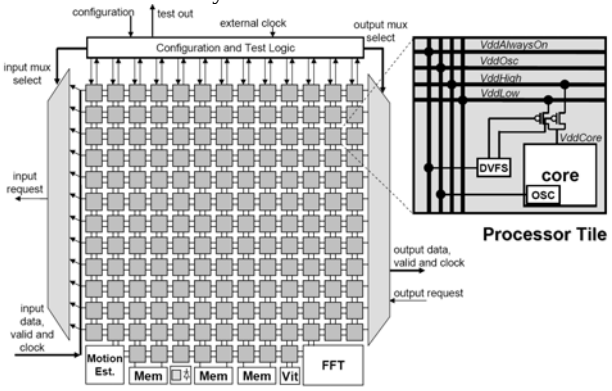


Fig. 1: Chip and tile block diagram

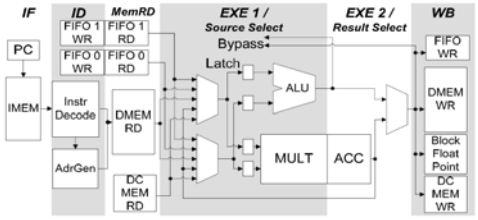


Fig. 2: The processor core's 6-stage single-issue pipeline

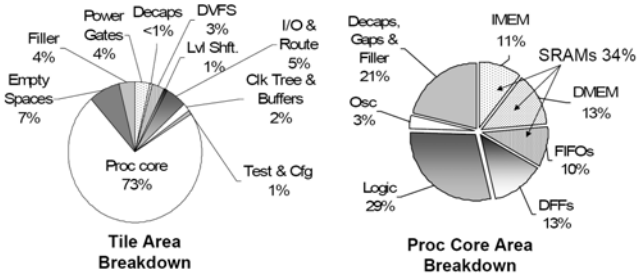


Fig 3: Tile and processor core area breakdowns

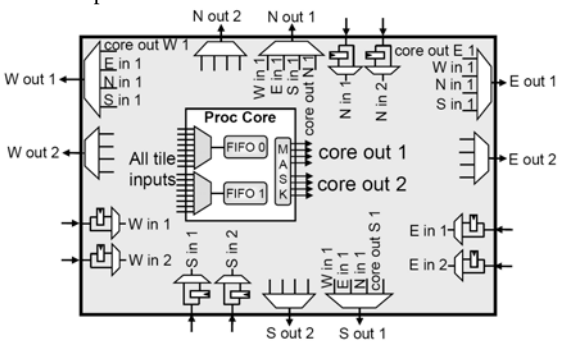


Fig. 4: Communication block diagram at the tile level

Dynamic Voltage and Frequency Scaling

To reduce power dissipation when processors are not fully active, processors can dynamically change their local supply voltage and local clock frequency. The 0.168 mm^2 processors change their supply voltage by connecting their local power grid (*VddCore*) to one of two global supply voltages [4]. The power grid *VddOsc* powers all local oscillators and enables smooth oscillator operation while switching power supplies. The power grid *VddAlwaysOn* powers configuration and communication circuitry as well as dynamic voltage and frequency control circuits. It is also possible to disconnect the *VddCore* of unused processors from the power grids for over 100x reduction in local processor core leakage currents.

A block diagram of the Dynamic Voltage and Frequency Scaling (DVFS) hardware controller is shown in Fig. 5. Supply voltage and clock frequency settings may be controlled by three methods: a configurable HW controller, local processor software, or statically defined configuration. The key idea is that the HW controller is driven by the utilization or "fullness" of the input FIFO(s) and the processor's stall signal. Thus, fuller FIFOs or infrequent stalling indicate the processor is too slow, and emptier FIFOs or frequent stalling indicate it is operating faster than necessary. Because the FIFO and stalling characteristics are different for different tasks, a configurable FIR/IIR low-pass filter smoothes FIFO utilization values to better estimate long-term utilization. Stall counter thresholds are similarly configurable. The hazards of dynamically switching a processor's power grid connection in a many-core system include local supply voltage droop and global power grid noise. To alleviate these, in normal operation the controller shuts off the oscillator during transitions as illustrated in Fig. 6. In addition, the PMOS power gates for each grid connection are organized as 48 parallel power gates with individual control signals and configurable relative timing. Fig. 6 contains processor SPICE waveforms of three example settings that tradeoff faster switching speeds for lower global power grid noise—in this case droop in *VddHigh* when the processor switches from *VddLow* to *VddHigh*. Figure 7 is a layout plot of a processor tile highlighting the memory blocks and local oscillator. The DVFS controller, inter-processor communication circuitry, and PMOS power gates are placed into a wrapper around the core and are powered by *VddAlwaysOn*, which makes voltage level converters necessary only between core and wrapper.

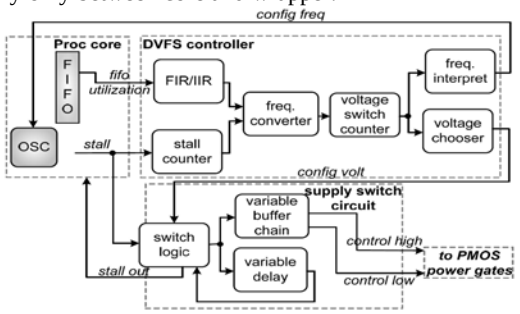


Fig. 5: DVFS circuit block diagram

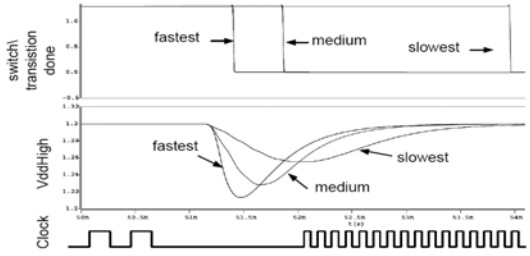


Fig. 6: Voltage switch speed effect on voltage grid (Low → High)

Fabricated 65 nm Chip

The presented chip has been fabricated in a 65 nm low-leakage ST Microelectronics CMOS process using only standard cells (except for the power gates) and is shown in Fig. 8 along with key simulated statistics. Programmable processors operate at a maximum clock frequency of 1.07 GHz @ 1.3 V and dissipate 31 mW when 100% active and executing a typical application instruction mix. Early measurements of a simple program of 39.2 mW @ 1.212 GHz and 1.3 V support the accuracy of the simulated results.

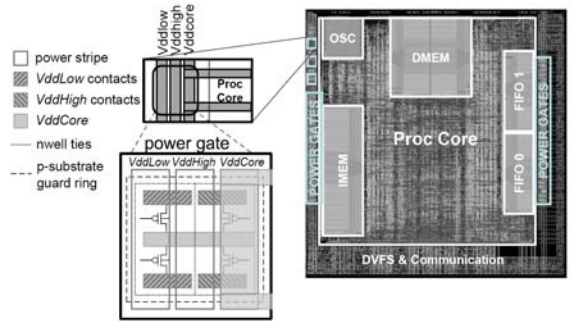
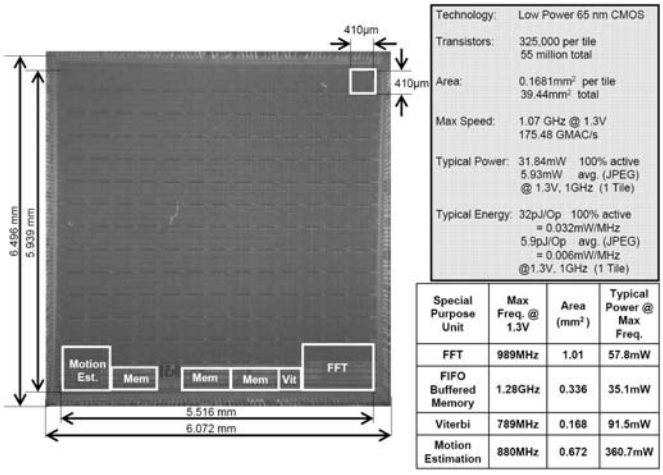


Fig. 7: Tile and core layout, and power gate distribution



Special Purpose Unit	Max Freq. @ 1.3V	Area (mm^2)	Typical Power @ Max Freq.
FFT	989MHz	1.01	57.8mW
FIFO Buffered Memory	1.28GHz	0.336	35.1mW
Viterbi	789MHz	0.168	91.5mW
Motion Estimation	880MHz	0.672	360.7mW

Fig. 8: Chip die photo and performance summary

Application Results

A 9-processor JPEG encoder operating with supply voltages of 1.3 V and 0.8 V achieves a simulated 48% reduction in energy dissipation with an 8% reduction in throughput compared to the same encoder running at 1.3 V. A complete lightly-optimized fully-compliant IEEE 802.11a/g wireless LAN receiver has been implemented using 39 processors (plus FFT and Viterbi) using only nearest-neighbor interconnect. Using long-distance interconnect, it requires only 27 processors—a 31% decrease. In simulation, the latter version dissipates 75 mW when operating at 690 MHz in 54 Mbps mode at full rate in real time, which includes 2.7 mW for the FFT and 5.5 mW for the Viterbi. This implementation is 34 times faster than an implementation on a TI C62x [7] and 19 times faster and 28x lower energy than an implementation on LART [8].

Acknowledgments

The authors gratefully acknowledge fabrication by ST Microelectronics; support from Intel, UC Micro, NSF Grant 430090 and CAREER award 546907, SRC GRC, Intellasys, and SEM; and thank J.-P. Schoellkopf, K. Torki, R. Krishnamurthy and M. Anders.

References

[1] Z. Yu et al., "An Asynchronous Array of Simple Processors for DSP Applications," *ISSCC* 2006.

[2] S. Vangal et al., "An 80-Tile 1.28TFLOPS Network-on-Chip in 65nm CMOS," *ISSCC* 2007.

[3] M. B. Taylor et al., "A 16-issue multiple-program-counter microprocessor with point-to-point scalar operand network", *ISSCC* 2003.

[4] K. Agarwal and K. Nowka. "Dynamic power management by combination of dual static supply voltages". *ISQED* 2007.

[5] M. Meeuwsen et al., "A Shared Memory Module for Asynchronous Array of Processors," *EURASIP Journal on Embedded Systems*, vol. 2007, Art. ID 86273.

[6] R. Apperson et al., "A Scalable Dual-Clock FIFO for Data Transfers Between Arbitrary and Haltable Clock Domains," *IEEE TVLSI*, Oct. 2007.

[7] M. Tariq et al., "Development of an OFDM based High Speed Wireless LAN Platform using the TI C6x DSP," *ICC* 2002.

[8] J. D. Bakker and F. C. Schoute, "LART: Design and implementation of an experimental wireless platform," *Vehicular Tech. Conference* 2000.