ASIC Implementation of Cairo University SPARC "CUSPARC" Embedded Processor

Amr A.Z. Suleiman
amrzahir@ieee.org

Alhassan F. Khedr
alhassan.f.khedr@ieee.org

S. E.-D. Habib
seraged@ieee.org

Electronics and Communications Department
Faculty of Engineering,
Cairo University, Cairo, Egypt.

***Abstract*— Cairo University SPARC "CUSPARC" processor is an IP embedded processor core conforming to SPARC V8 ISA. CUSPARC is fully developed at Cairo University and is the first Egyptian processor. In this paper, the ASIC Implementation and Verification of the CUSPARC processor is described at 130nm technology node. CUSPARC scores a typical clock frequency of 260MHz, power dissipation of 0.11 mW/MHz and power Efficiency of 8.78 DMIPS/mW, which makes it very suitable for embedded and real-time systems.**

I. INTRODUCTION

The market demands are pushing up the complexity of embedded systems at an unprecedented rate. In many application areas, e.g., wireless communications or automotive applications, the answer to these demands is multicore systems. It is expected that processor cores per chip will, in the near future, reach tens to hundreds cores. However, the stringent power constraints on these embedded systems put a premium on the power efficiency of candidate processor cores. By the same token, high performance, high power general purpose processors are not suitable candidates for such systems. The current replacement of the POWERPC core on Xilinx FPGA chips in favor of an ARM processor core points out this trend explicitly [1].

This paper introduces the ASIC design of a low-power medium performance IP core for embedded applications. This IP core is code named CUSPARC for Cairo University SPARC. CUSPARC is the first fully-operational Egyptian processor. It was fully designed at the Electronics and Communications department, Cairo University during the past five years. CUSPARC conforms to SPARC ISA V8 standard [2]. A companion paper presents the RTL design of CUSPARC [3]. This paper is organized as follows: In section II, we discuss the CUSPARC architecture. Section III presents the design flow adopted for the ASIC implementation and verification of this processor on the 130nm CMOS Technology. The performance (area, speed and power) of the CUSPARC processor is detailed in section IV. Finally, in section V, we compare the CUSPARC with some of the famous ARM and MIPS embedded processors indicating that CUSPARC compares favorably against many of these processors.

II. CUSPARC ARCHITECTURE

The CUSPARC Architecture consists of: Integer Unit, Instruction and Data Caches, Cache Controller, Memory Controller, Wishbone Bus and some peripherals. Figure (1) shows the block diagram of the CUSPARC. The following paragraphs give a brief description of each block.

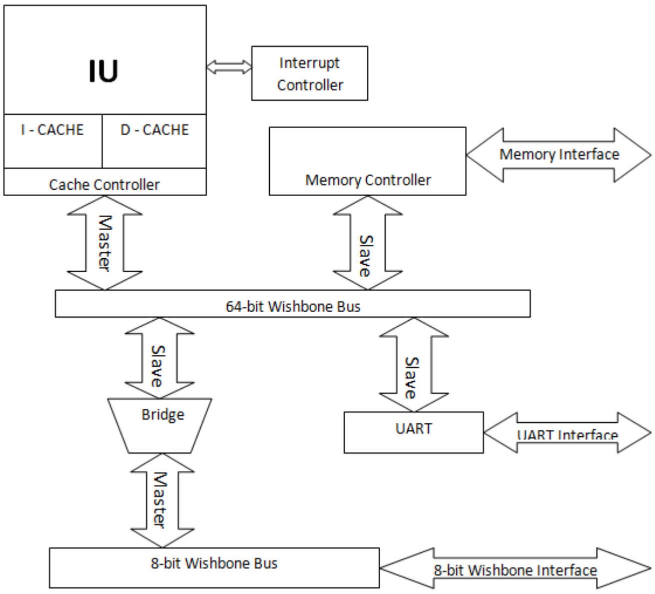

Fig. 1. CUSPARC Block Diagram

A. Integer Unit

The integer unit (IU) supports about 80 instructions of the instruction set of SPARC V8 ISA. CUSPARC IU has four pipeline stages; namely:

- Fetch
- Decode
- Execute
- Write Back

With the Memory read and write instructions specially handled so as not to disrupt the pipeline. This IU implements 4 overlapping register windows conforming to the SPARC windowed register-file [2]. Data forwarding is employed to

978-1-61284-150-2/10/$26.00 ©2010 IEEE

resolve data hazards. For the branch hazards, we follow a branch-always strategy. The pipeline is flushed if the branch is not taken.

B. Cache

CUSPARC has Harvard architecture. Both the instruction Cache (I-Cache) and the data Cache (D-Cache) have a 4 Kbytes size. Both caches are direct mapped.

C. Cache Controller

The cache controller is a simple finite state machine that works as a server responding to requests from the 2 caches in case of a cache miss or an I/O access. The cache controller is the only master on the processor Bus. This makes the Bus implementation simpler as there is no need for a bus arbiter.

D. Memory controller

The memory controller interfaces CUSPARC to the main memory (RAM) or the boot memory (Flash). It maps the address from caches to the external memory. The design of the controller handshaking with external memory is generic and can be controlled by software. This provides the flexibility to use different RAM or FLASH memories with different speeds and sizes.

E. WISHBONE Bus

CUSPARC adopts the open source Wishbone bus [4] as its processor bus. This processor bus is a 64-bit bus that interfaces between the cache controller as a master and memory controller and peripherals as slaves. As is shown in Figure (1), another slow 8 bit Wishbone bus is included to provide I/O interface for CUSPARC. A bridge (FIFO) is used to connect the two buses.

F. Peripherals

CUSPARC has 2 UARTs, 3 timers and a watch dog timer. There's also an interrupt controller, supporting 3 different interrupts.

III. ASIC FLOW

Figure (2) shows the digital ASIC flow adopted to design, implement and verify the CUSPARC processor.

A. Synthesis

We followed mainly a structural approach at RTL level to define the VHDL model of CUSPARC. We synthesized the design using Artisan standard cell library [5] for IBM CMOS 130nm 8RF technology [6]. We maintained the design hierarchy when synthesizing the design so as efficiently floorplan and layout the chip. Static timing verification tools are used to generate timing constraints which are linked into the synthesizer to help optimize certain paths generated during synthesis. Other constraints such as drive strength on various signals and allowable block area are also used to instruct the synthesizer how to generate an efficient netlist. The slow corners of the library cells are used to implement the design so as to guarantee the required performance even under worst case conditions.

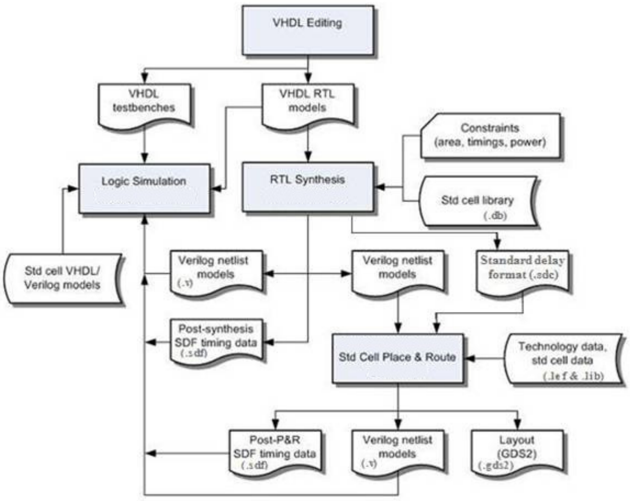
VHDL Editing
VHDL testbenches
VHDL RTL models
Constraints (area, timings, power)
Logic Simulation
RTL Synthesis
Std cell library (.db)
Std cell VHDL/ Verilog models
Verilog netlist models (.v)
Verilog netlist models
Standard delay format (.sdc)
Post-synthesis SDF timing data (.sdf)
Std Cell Place & Route
Technology data, std cell data (.lef & .lib)
Post-P&R SDF timing data (.sdf)
Verilog netlist models (.v)
Layout (GDS2) (.gds2)

Fig. 2. ASIC Digital Flow

B. Place and Route

After successfully synthesizing the design and verifying it, Placement and Routing (P&R) took place. We floorplanned the design so as to have minimum area and lowest power. Figure (3) shows the final layout of the chip.

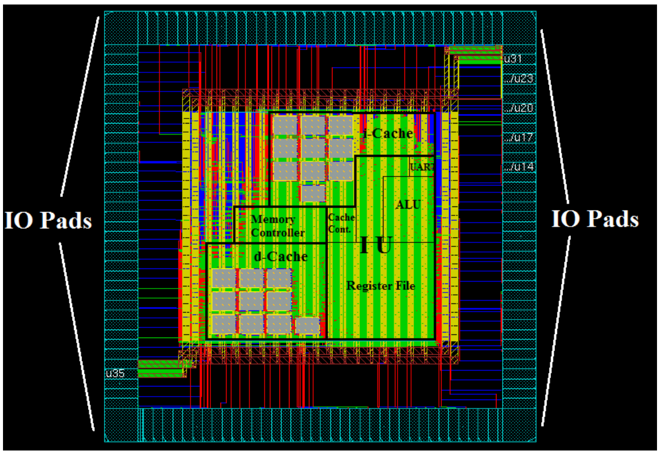

Fig. 3. CUSPARC Layout View

C. Verification

We carried out several verification steps for the CUSPARC processor.

1) Functional Simulation: The functionality of the design was tested using simulation tool in Pre-Synthesis, Post-Synthesis and Post-Layout phases of the design. At each design phase, we have to be sure that the functionality of the design is preserved.

2) *Static Timing Analysis:* A Static Timing Analysis (STA) tool is used to verify the timing of the design and to ensure the absence of any unaccounted paths. Frequently, we had to back annotate the STA results into the synthesis tool so as to improve the timing of the design.

3) *Power Analysis:* A Power Analysis tool is used to calculate the post-layout power dissipated in the design. The following steps are made for calculating the power of the CUSPARC:

- Import the design into the power analysis tool.
- Read the signal activity file (SAIF or VCD) generated from Dhrystone benchmark simulation.
- Read the parasitic data exported from P&R tool after layout.
- Report the power.

4) Formal Verification and layout verification: Common practices for formal verification and layout checking (DRC, ERC, etc.) were followed throughout the CUSPARC design.

IV. AREA AND POWER PERFORMANCE

For the target IBM's 130nm process, the core area of CUSPARC with Caches is 1.96mm^2 (excluding I/O pads). The core area without Caches is only 1.38mm^2. The typical maximum clock frequency is 260MHz. At this frequency, performance is 267 Dhrystone MIPS. Table (I) summarizes these estimates.

CUSPARC	With Cache	Without Cache
Core Area	1.96 mm^2	1.38 mm^2
Power	0.58 mW/MHz	0.11 mW/MHz
DMIPS/MHz	0.9663	0.9663

TABLE I
CUSPARC AREA, POWER AND PERFORMANCE SUMMARY

V. COMPARING CUSPARC TO ARM AND MIPS IP CORES

Table (II) summarizes the features of CUSPARC core and compares it with some famous ARM [7] and MIPS [8] cores. All these core processors are implemented using 130 nm CMOS technology.

All the performance metrics for CUSPARC are based on Post-Layout simulations. The metric values shown in Table (II) for CUSPARC as well as for the ARM and MIPS processors are for the cores only excluding caches. The power estimates of our design represents "typical" consumption when running time-based power calculations based on the Dhrystone benchmark. Figure (4) shows the position of the CUSPARC processor on the DMIPS/MHz vs. Power/MHz chart.

From Figure (4) we can deduce the power efficiency by dividing the DMIPS by the Power. Figure (5) depicts the power efficiency of CUSPARC versus the considered five ARM and MIPS processors.

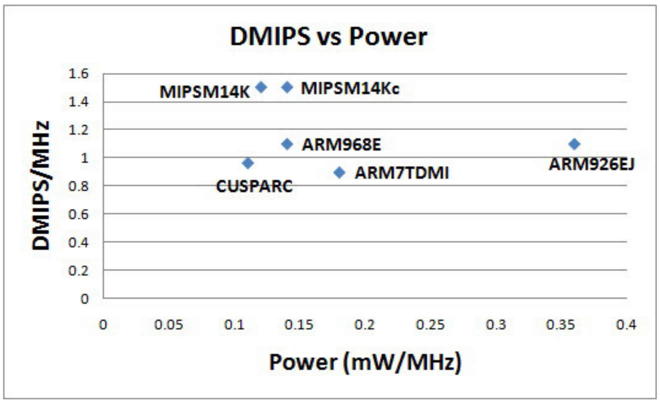

Fig. 4. DMIPS/MHz vs. Power per MHz (mW/MHz)

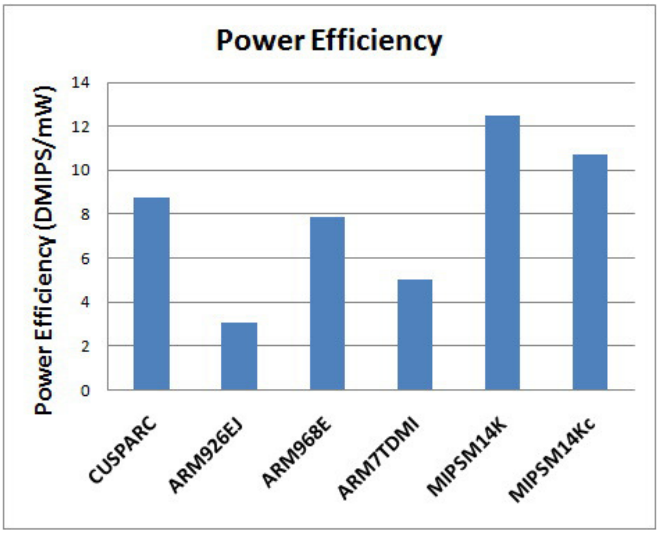

Fig. 5. Power Efficiency (DMIPS/mW)

From figure (5) it's clear that CUSPARC is highly competitive relative to these common embedded processor cores. Admittedly, it's very difficult to fairly compare CUSPARC against these vendor-supplied power and performance estimates, because they assume other unknown variables (exact core configurations, logic-synthesis scripts, cell libraries, etc.). Yet, figures (4) and (5) indicate that CUSPARC fares well against comparable ARM and MIPS cores.

VI. CONCLUSION

The ASIC design, implementation and verification procedures of the Cairo University SPARC (CUSPARC) core are described for a 130nm target technology. The area, power and delay performance of CUSPARC is evaluated. Our results show that CUSPARC competes favorably, especially on power efficiency metric, with several famous ARM and MIPS cores.

Feature	CUSPARC	ARM926EJ	ARM968E	ARM7TDMI	MIPSM14K	MIPSM14Kc
CPU ISA	SPARC	ARM/Thumb	ARM/Thumb	ARM/Thumb	MIPS32 R2	MIPS32 R2
Arch. Width	32 bit	32 bit	32 bit	32 bit	32 bit	32 bit
Pipeline Depth	4 stage	5 stage	5 stage	3 stage	5 stage	5 stage
Core Frequency	260 MHz	238 MHz	297 MHz	184 MHz	100 MHz	100 MHz
Core Area	1.38 mm2	1.45mm2	0.45mm2	0.35mm2	0.35mm2	0.61mm2
DMIPS/MHz	0.9663	1.1	1.1	0.9	1.5	1.5
Power (typical)	0.11 mW/MHz	0.36 mW/MHz	0.14	0.18 mW/MHz	0.12 mW/MHz	0.14 mW/MHz
Efficiency DMIPS/mW	8.78	3.05	7.85	5.22	12.5	10.7

Table II
CUSPARC, ARM AND MIPS COMPARISON (WITHOUT CACHES)

VII. ACKNOWLEDGMENT

The authors acknowledge a MOSIS MEP grant # 4960 through which they got access to the technology files of IBM 0.13μm CMOS 8RF-DM and the corresponding Artisan standard cell library.

REFERENCES

[1] Mike Santarini, "Xilinx Architects ARM-Based Processor-First, Processor-Centric Device", Xilinx Xcell Journal, second quarter2010, p.g. 6-11, available at www.xilinx.com/publications/archives/xcell/issue71/coverstory.pdf

[2] The SPARC Architecture Manual, Version 8. Available at http://www.sparc.com/standards/V8.pdf

[3] Ezz El-Din O. Hussein et al "CUSPARC IP processor: Design, Characterization and Applications", 22 International Conference on Microelectronics (ICM'10) 19 – 22 December 2010, Cairo, Egypt.

[4] Wishbone bus specification, "WISHBONE system-on-Chip (SoC) Interconnection Architecture for Portable IP Cores". Available at http://opencores.org/opencores,wishbone

[5] Refer to http://www.mosis.com/ibm/ibm_rules_libs.html

[6] Refer to http://www.mosis.com/ibm/8rf-dm/

[7] ARM Processors Specifications, http://www.arm.com/products/processors/classic/

[8] MIPS M14K Processors, http://www.mips.com/media/files/M46_MIPS_Reprint.pdf