Multi-core DSP sub-system IP

G. Rauwerda, K. Sunesen, T. Bruintjes, T. Hoang, J. Potman

Recore Systems B.V., P.O. Box 77, 7500 AB Enschede, The Netherlands

Gerard.Rauwerda@RecoreSystems.com

Abstract

Next generation digital signal processors for space applications have to be programmable, high performance and low power. Moreover, the digital signal processors have to be tightly integrated with space relevant interfaces in System-on-Chip (SoC) solutions with the required fault tolerance.

We present DSP and Network-on-Chip IP technology to create multi-core architectures for payload data processing. The IP complements existing general purpose processing solutions and can be seamlessly integrated to extend processing and interconnect capabilities in next generation DSP multi-cores.

I. Introduction

On scientific missions to deep space a wealth of data is gathered, analysed and compressed on-board before being relayed back to earth. The data cannot be sent to earth in its entirety since modern instruments gather much more data than can be communicated back to earth. For a correct interpretation of what is going on in space, and valid answers to exciting questions it is key that the compressed and processed data is correct.

Next generation digital signal processors for space applications have to be programmable, high performance and low power. Moreover, the digital signal processors have to be tightly integrated with space relevant interfaces in System-on-Chip (SoC) solutions with the required fault tolerance.

With the planning for the Cosmic Vision programme in mind, ESA plans to have a standard ASIC with a space qualified rad-hard Digital Signal Processor and a performance of at least 1000 MFLOPS in its portfolio. In this paper, we present multi-core DSP sub-system IP, built of fixed-/floating-point Xentium DSP cores connected in a Network-on-Chip [6][7][8].

This paper is organized as follows: Section II presents the architectural aspects of a heterogeneous multi-core DSP system. Section III provides an overview on the Xentium DSP processor. In Section IV the software development process for mulit-core DSP architectures is discussed. Section VI concludes with ideas towards realization of the next-generation many-core DSP for space.

II. Multi-Core DSP Architecture

We present a multi-core DSP architecture for streaming Digital Signal Processing for on-board payload data processing (OPDP) applications. In the Massively Parallel Processor Breadboarding (MPPB) study [2][5] and in the Scalable Sensor Data Processor (SSDP) [10] a Network-on-Chip (NoC) based multi-core DSP sub-system is integrated together with a conventional general purpose processor (LEON) sub-system in a System-on-Chip (SoC).

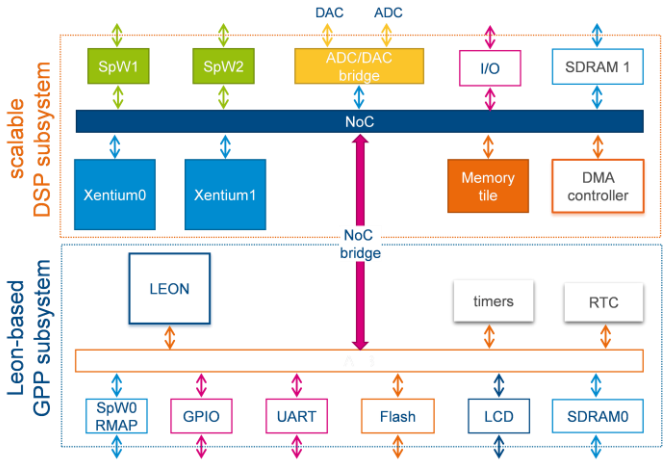

Figure 1: Multi-core processor comprising a NoC sub-system (*scalable DSP subsystem*) and AMBA sub-system (*GPP subsystem*)

Figure 1 shows the multi-core DSP processor architecture comprising two main parts: the NoC sub-system and the AMBA sub-system. Generally, the LEON sub-system acts as the host processor, initializing and controlling the multi-core DSP sub-system. After initialization by the host processor, the multi-core DSP sub-system will autonomously run compute-intensive DSP functions. The architecture combines the AMBA legacy subsystem with the performance of the DSP subsystem. Existing AMBA-based hardware IP components can be easily integrated and legacy software can be easily ported.

The multi-core DSP subsystem comprises the following key building blocks:

- The Xentium® is a programmable high-performance DSP processor core that is efficient and offers high-precision;
- Network-on-Chip (NoC) technology provides sufficient bandwidth, flexibility and predictability which are required for interconnecting DSP cores and I/O interfaces in streaming DSP applications.

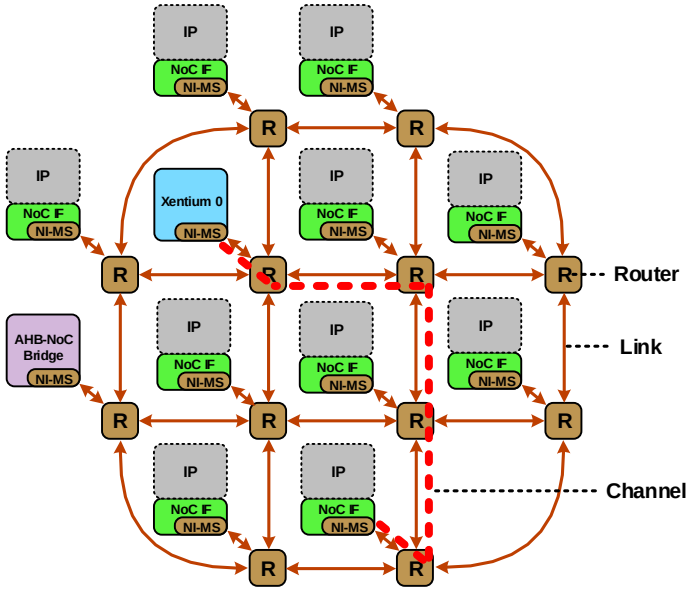

Figure 2: NoC-connected multi-core sub-system

Mainly the high bandwidth peripherals are connected to the NoC while the others are connected to the AMBA system. The AMBA system also provides the ability to attach the well-known LEON core processor to support execution of existing software with minimal changes to the source code.

A. Network-on-Chip

Tomorrow's many-cores for (streaming) DSP applications will be interconnected by a network-on-chip (NoC) instead of a bus. Currently, most multi-core architectures rely on a central bus for interconnecting the (digital signal) processor cores. Such a central bus creates a major bottleneck and impedes performance, scalability and composability of such systems. A NoC approach does not suffer from these limitations. A NoC scales with the number of cores in the design. The more cores there are, the larger the network, and, hence, the more aggregate bandwidth is available in the SoC. Other advantages include that a NoC inherently supports short and structured wires, enabling increased clock rates and easier link optimization. NoCs allow disabling inactive parts of the network, which is essential for energy-efficiency and dependability. Finally, a key feature of NoCs is their predictable performance.

Using transparent I/O interfaces it is even possible to extend the NoC across the chip boundaries creating a network-of-chips. Hence, NoC technology enables true scalability of many-core systems-of-chips.

The NoC sub-system is connected with the AMBA sub-system through an AHB-NoC Bridge (as depicted in Figure 2). All components connected on the NoC use memory-mapped communication, and, hence, are available as memory-mapped components in the AMBA sub-system. So, NoC-connected components are accessible by devices on the AMBA and vice versa. This makes it possible for every master on the system to read and write data anywhere in the system. The LEON can for example read and write in local Xentium memories and the Xentium can read and write directly in the AMBA peripherals.

1) XY-routing and QoS

The NoC consists of a set of 5-port packet-switched routers that use service flow control. One port is the local port connected to the NoC peripherals; the other ports are connected to the neighbouring routers.

The services are used to provide Quality of Service (QoS) for individual data transfers between two communicating entities (i.e. NoC-connected devices) in the NoC architecture. The NoC interface consists of 32-bit data in both directions.

The NoC employs XY-routing, i.e. the direction in each router is determined by the router coordinates and the destination coordinates. Hence, the routing is fixed and depends on the topology of the 2D mesh. The use of a fixed XY-routing scheme ensures in-order delivery of transfers and prevents deadlocking.

2) NoC Transactions and Performance

The NoC links are full-duplex bidirectional. Each network link can handle 32 bit concurrently in each direction. The NoC supports burst transfers with a maximum bandwidth of 32 bits per clock cycle.

The NoC protocol supports single read/write, block read/write and (2D) stride-based transfers. With (2D) stride support data transformations can be done efficiently as part of the data transfer..

3) Network Interface

A Network Interface (NI) is a component to connect IP components (including internal/external IO interfaces) to the NoC. For the connected IPs, the NI hides the implementation details of a specific interconnect. NIs translate packet-based NoC communication on the NoC side into a higher-level protocol that is required on the IP side, and vice versa, by packetizing and de-packetizing the requests and responses.

Using the transparent NI it is even possible to extend the NoC across the chip boundaries. Several I/O interfaces are available on the multi-core DSP architecture, such as SpaceWire bridge interfaces, bridges to external Analog-to-Digital Convertor (ADC) and Digital-to-Analog Convertor (DAC) devices. Through the NI, all these I/O interfaces become memory-mapped interfaces in the multi-core processor system.

III. XENTIUM DSP

The Xentium is a programmable high-performance 32/40-bit fixed-point DSP core for inclusion in multi-core systems-on-chip. High-performance is achieved by exploiting instruction level parallelism using parallel execution slots. The

Very Long Instruction Word (VLIW) architecture of the Xentium features 10 parallel execution slots and includes support for Single Instruction Multiple Data (SIMD) and zero-overhead loops. The Xentium is designed to meet the following objectives: high-performance, optimized energy profile, easily programmable and memory mapped I/O.

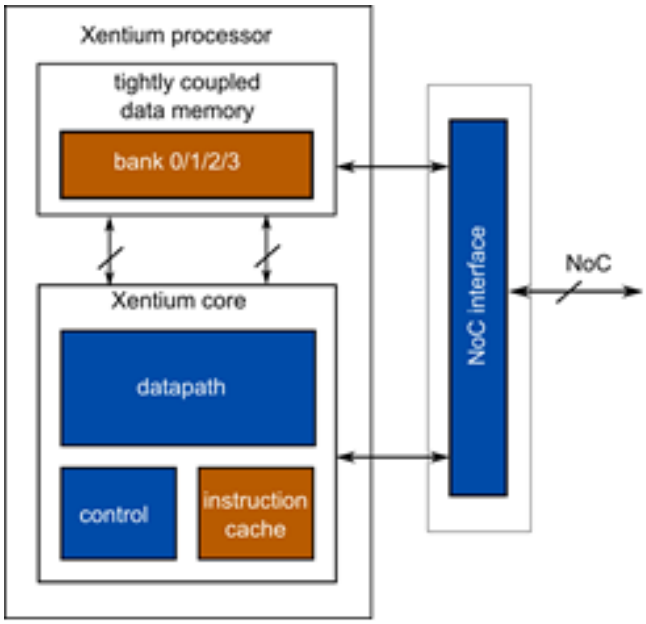

Figure 3: Top-level of the Xentium DSP

The core modules of the Xentium DSP are the Xentium core, tightly coupled data memory, and a NoC interface as shown in the block diagram in Figure 3. The size of the data and instruction memories is configurable at design-time of the multi-core processor SoC. A default instance of the Xentium DSP contains 32 kB tightly coupled data memory and 16 kB instruction cache.

A. *Xentium Datapath*

The Xentium datapath contains parallel execution units and register files. The different execution units can all perform 32-bit scalar and vector operations. For vector operations the operands are interpreted as 2-element vectors. The elements of these vectors are the low and high half-word (16-bit) parts of a 32-bit word. In addition several units can perform 40-bit scalar operations for improved accuracy. Most operations can be executed conditionally.

The Xentium datapath provides powerful processing performance:

- 4 16-bit MACs per clock cycle, or
- 2 32-bit MACs per clock cycle, or
- 2 16-bit *complex* MACs per clock cycle

The Xentium architecture has two M units, four S units, two P units and two E units. The M units can perform multiply operations. The S and P units perform ALU operations (e.g. additions and subtractions) including shift and pack instructions, respectively. The E units are responsible for load and store operations.

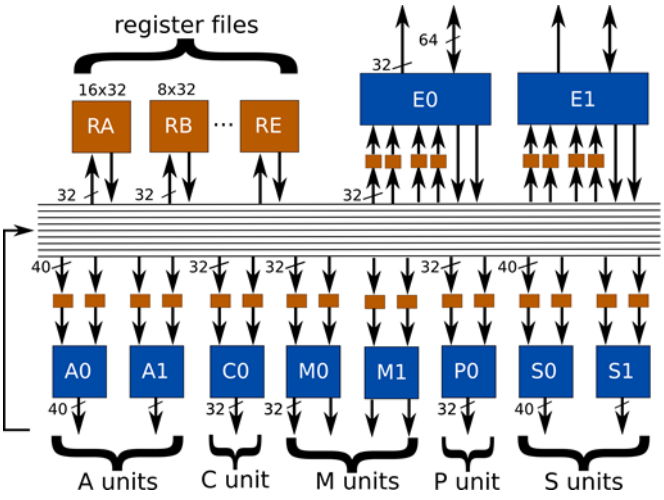

Figure 4: The Xentium datapath

B. *Xentium Control*

The control block in the Xentium core performs instruction fetching and decoding, and controls the execution units in the datapath. Instructions are fetched from Xentium-external memory (e.g. on-chip or off-chip memory in the NoC sub-system) and are stored in the Xentium instruction cache. The programmer can indicate that a section of a Xentium program has to be pre-fetched by the control to ensure that the instructions of that section of the program are cached. This prevents cache misses during execution, which makes the execution time of the pre-fetched section of the program predictable.

C. *Tightly-coupled data memory*

The tightly coupled data memory is organized in parallel memory banks to allow simultaneous access by different resources. The data memory can be simultaneously accessed by the Xentium core as well as by the Xentium NoC interface (i.e. other components in the NoC sub-system have access to the Xentium memories).

The size of the memory banks is parametrizable at design-time. By default the data memory in the Xentium tile is organized in 4 banks of 8 kBytes each, implemented using SRAM cells.

The memories in the Xentium processor are protected by Error Detection and Correction (EDAC) logic.

D. *Debug Support*

Xentium processor IP includes debug hardware for remote debugging. The Xentium debug units supports stepping, watch points, break points, back tracing, and full access to registers and memory.

E. *Application Profiling*

In order to facilitate profiling of Xentium DSP programs, a number of software configurable counters (i.e. performance counters) are integrated in the Xentium processor IP. Through a configuration register, these counters can be configured to monitor different events in the Xentium including events such as cache misses and load/store wait cycles.

IV. Software Development

Xentium Software Development Environment (SDE) is C-based and includes a standard tool chain consisting of a C-compiler with standard C-library, an assembler, a linker, a simulator, and a debugger.

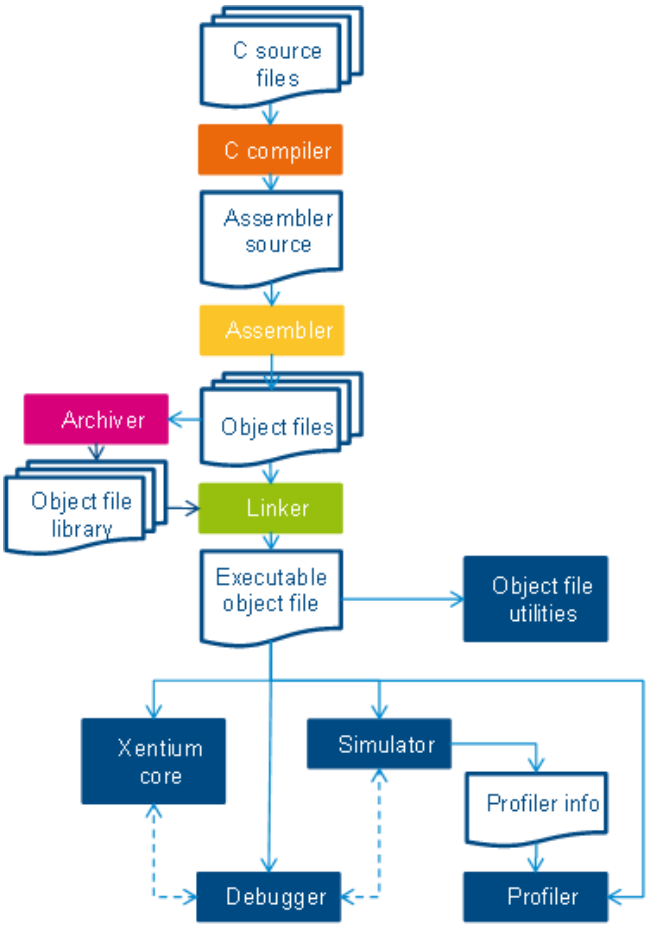

Figure 5: The Xentium toolchain

The tools in the tool chain are based on well-known tools such as the LLVM compiler infrastructure, the GNU binary utilities, and the GNU debugger, offering a familiar user interface to allow a quick start. A Xentium Eclipse plug-in integrates the Xentium tool chain in the Eclipse C/C++ IDE to provide a familiar graphical user interface for editing, building, simulating and debugging Xentium programs.

- C-compiler supports C99 and built-in functions for Xentium instructions. It comes together with a Newlib-based standard C-library.
- The assembler has clean and readable assembly syntax and a pre-processor with macro functionality to facilitate optional hand-programming.
- The Linker, which is based on the GNU linker, has support for linker scripts which allow developers to describe the memory layout of executables.
- The archiver lets developers create reusable code libraries.
- With the Xentium Instruction Set Simulator, developers can test, time and trace execution of Xentium executables on their PC.
- The Xentium Debugger allows debugging a Xentium executable running in the Xentium Simulator or on the Xentium hardware. The debugger is based on GDB, the GNU debugger, and therefore offers a familiar user interface.
- The Xentium Profiler allows the user to get detailed cycle information of an application running on the Xentium Simulator.

V. Towards Multi- And Many-Core Next-Generation DSP Architectures

We have presented the Xentium DSP and Network-on-Chip technology to create multi-core SoC architectures for on-board payload data processing. We integrated the Xentium DSP and NoC IP the Massively Parallel Processor Breadboard (MPPB) [2][5], tested the IP in XentiumDARE IC [11] and improved the IPs in the scope of the Scalable Sensor Data Processor (SSDP) [10].

The journey of integrating more Xentium DSP cores with advanced features, such as floating-point support, continues to further increase the performance of the next-generation data processor for space. Moreover, fault-tolerant features to protect against permanent and transient errors due to radiation effects will be added to the NoC technology.

Using CMOS65Space we estimate a many-core DSP architecture with 8 floating-point Xentium DSP will provide a performance of at least 8 GFLOPS, opening new opportunities for advanced data processing in for example scientific instruments.

VI. References

[1] European Space Agency, “JUICE Definition Study Report”, ESA/SRE(2014)1, 2014

[2] Recore Systems, “Massively Parallel Processor Breadboarding Study”, 2012

[3] European Space Agency, “Next Generation Space Digital Signal Processor Software Benchmark”, TEC-EDP/2008.18/RT, 2008

[4] R. Trautner, “Next Generation Processor for On-Board Payload Data Processing Applications – ESA Round Table Synthesis”, TEC-EDP/2007.35/RT, 2007

[5] G. Rauwerda, “Massively Parallel Processor Breadboarding (MPPB) Study – Final Presentation”, ESA DSP Day, ESTEC, Noordwijk, The Netherlands, August 2012

[6] K. Sunesen, “Multi-core DSP architectures”, Adaptive Hardware and Systems Conference, Torino, Italy, June 2013

[7] K. Walters et al., “Multicore SoC for On-board Payload Signal Processing”, Adaptive Hardware and Systems Conference, San Diego, USA, 2011

[8] K.H.G. Walters, “Versatile Architectures for Onboard Payload Signal Processing”, PhD thesis, University of Twente, The Netherlands, 2013

[9] .M. Souyri et al., “NGDSP European Digital Signal Processing Trade-off and Definition Study – Final Presentation”, ESA DSP Day, ESTEC, Noordwijk The Netherlands, August 2012

[10] R. Pinto et al., “Scalable Sensor Data Processor: Development and Validation”, DASIA, Tallinn, Estonia, 2016

[11] K. Sunesen, “XentiumDARE IC DSP SoC Demonstration ASIC”, DARE+ Final Presentation, ESTEC, Noordwijk, The Netherlands, December 2014