HIGH PERFORMANCE PPC BASED DPU WITH SPACEWIRE RMAP PORT

Session: SpaceWire Components

Short Paper

Pekka Seppä, Petri Portin

Patria Aviation Oy, Systems/Space, Naulakatu 3, FI-33100, Tampere, Finland

Omar Emam

ENS - Future Mission Systems, EADS Astrium, Gunnels Wood Road, Stevenage - SG1 2NJ, England - UK

Wahida Gasti

European Space Agency, Postbus 299, NL-2200 AG Noordwijk, The Netherlands

Chris McClements, Steve Parkes

University of Dundee, School of Computing, Dundee, Scotland, UK

E-mail: pekka.seppa@patria.fi, petri.portin@patria.fi, omar.emam@astrium.eads.net, wahida.gasti@esa.int, chris@star-dundee.com, sparkes@computing.dundee.ac.uk

ABSTRACT

This paper presents a high performance Data Processing Unit (DPU) with a SpaceWire RMAP Interface. Over 1000 MIPS / 800 MFLOPS performance at 800 MHz CPU clock is achieved with a high performance PowerPC CPU. The DPU has three SpaceWire links operating at up to 100 Mbits/s, two of which with RMAP. These are used to connect the DPU to other SpaceWire nodes. The DPU's 100 Mbits/s RMAP ports can be RMAP Initiator, an RMAP Target or both. The DPU has been developed for use on the MARC demonstrator.

The SpaceWire RMAP port of the DPU is implemented by using ESA's SpaceWire RMAP IP Core. This allows that the RMAP port of the DPU supports both the Initiator RMAP Interface and Target RMAP Interface. The VHDL RMAP IP Core is integrated in the DPU system by VHDL user logic which allows CPU / user software to access all the RMAP IP functions and generates interrupts for the CPU for fluent software execution.

The RMAP IP Core along with the CPU bus interface, memory interfaces with DMA, UART and Interrupt controller are implemented in Actel's Axcelerator AX2000 and ProASIC3E A3PE3000 FPGAs.

1 INTRODUCTION

1.1 CPU

Central Prosessing Unit (CPU) of the DPU is MPC7448, a member of PowerPC 7450 RISC Microprocessor family. The processor has a superscalar architecture that can dispatch and complete three instructions simultaneously, and thus its performance is typically higher than one MIPS / MHz. The CPU contains several execution units including an IEEE-754 compliant double-precision Floating Point Unit (FPU), 32 kbyte Level 1 (L1) instruction and data caches, and a 1 Mbyte Level 2 (L2) cache memory [1]. The MPC7450 implements the 32-bit portion of the PowerPC architecture, which provides 32-bit effective addresses, integer data types of 8, 16, and 32 bits, and floating-point data types of 32 and 64 bits. The CPU interfaces to memory and peripherals by a 64-bit wide bus using 60x protocol [2].

The CPU is set to operate at 800 MHz internal clock frequency in the DPU, although CPU type selected could be clocked up to 1420 MHz. This decision has been made in order to reduce power consumption and avoid thermal cooling by a fan.

1.2 RMAP ON SPACEWIRE

SpaceWire is a communications network for use onboard spacecraft. It is designed to connect sensors, memories, data processing units and a downlink telemetry subsystem providing an integrated onboard data-handling network. SpaceWire links are serial, high-speed (2 Mbits/s to 200 Mbits/s or higher), bi-directional, full-duplex, point-to-point data links that connect together SpaceWire equipment. Application information is sent along a SpaceWire link in discrete packets. Control and time information can also be sent along SpaceWire links. SpaceWire is defined in the European Cooperation for Space Standardization (ECSS) ECSS-E50-12C standard [3].

There is a number of communication protocols that can be used in conjunction with the SpaceWire. The Remote Memory Access protocol (RMAP) is one of these protocols and is specified in ECSS-E-ST-50-52C standard [4]. The aim of RMAP is to support reading from and writing to memory in a remote SpaceWire node. RMAP can be used to configure a SpaceWire network, control SpaceWire nodes, and to transfer data to and from SpaceWire nodes. RMAP can also be used to download and debug software on a remote processor.

DPU’s RMAP SpaceWire links are implemented using the ESA SpaceWire - RMAP IP core developed by University of Dundee for ESA. There are two main function types of the RMAP IP core. The first type is referred to as the Initiator RMAP interface, which sends out RMAP commands and receives any replies. The second type is the Target RMAP Interface, which receives RMAP commands, executes them and sends out any required replies [5].

1.3 DPU

MARC demonstrator DPU is a general purpose 800 MHz high performance space flight (payload) computer providing 1 Dhrystone MIPS (DMIPS) / MHz performance. The DPU high-speed interface to external world is three SpaceWire links, two of which support RMAP protocol.

The breadboard model of the DPU computer will function as a software development board and a demonstrator in MARC SpaceWire - RMAP network. Breadboard models were manufactured using mainly commercial quality grade components. Industry- and military-grade components may be used as well in places where commercial equivalent is not suitable to represent the required function. In general all the core DPU components in the design has space flight equivalent counterparts.

2 DPU ARCHITECTURE

The system architecture of the MARC demonstrator DPU is depicted in **Figure 1**. The DPU system consists of the CPU, System Control Unit (SCU), SpaceWire Control Unit (SpWCU) and memory. DPU's main external data interfaces are the three SpaceWire links capable to 100 Mbits/s data rate full-duplex. The main software debug interface is an EIA-232 Universal Asynchronous Receiver / Transmitter (UART), and the PowerPC COP interface is for low-level software debugger connection from a remote computer. DPU boot-loader and self-test software are controlled via UART.

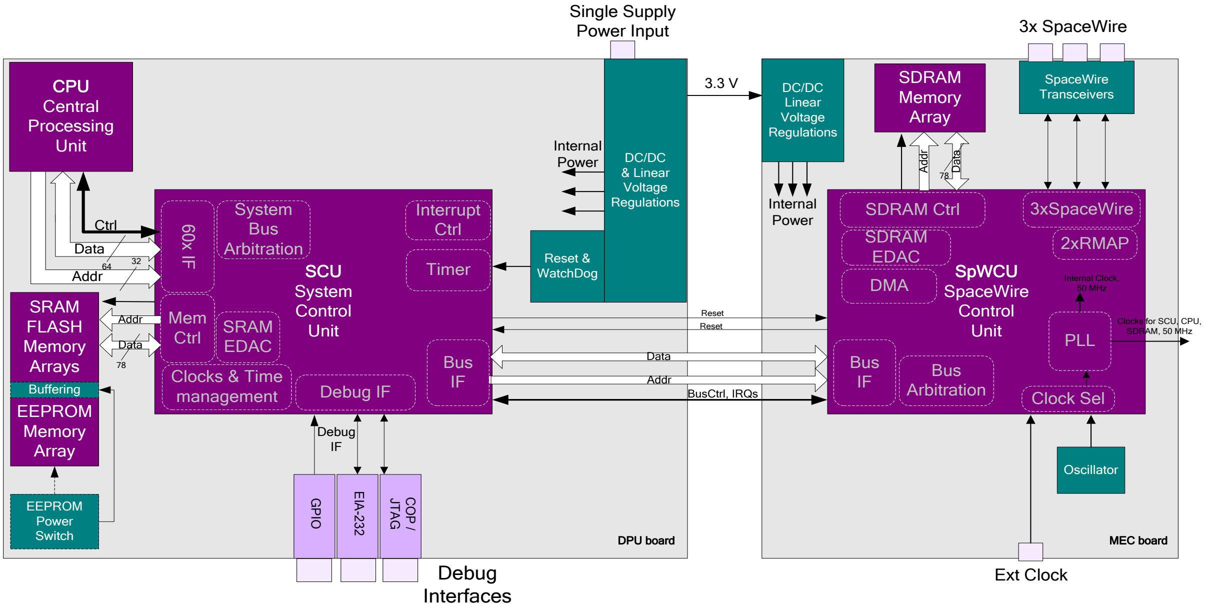

Figure 1 Block Diagram of the DPU

The SCU interfaces the processor with 64-bit PowerPC 60x bus to DPU memory and functions. The SCU controls SRAM, FLASH and EEPROM memory interfaces, and timer, interrupt controller, GPIO, UART and SRAM EDAC are embedded in the SCU. The SCU interfaces also the SpWCU and the CPU by a custom made 32-bit bus.

The SpWCU controls the SDRAM memory interface, and interfaces the Unit to the CPU via SCU. Embedded Direct Memory Access (DMA) unit controls accesses from the CPU and SpaceWire / RMAP functions to the SDRAM memory. The SpWCU and SDRAM memory devices are placed on a separate mezzanine board from the DPU main board.

The DPU memory consists of the 256 kB FLASH for boot program storage, 4 MB EEPROM which can be unpowered when not used, 8 MB EDAC protected SRAM and 512 MB EDAC protected SDRAM. All the memory is accessible by the CPU and can act as storage either for code or data. Data transfers between the DPU and external world through SpaceWire links is routed via SDRAM.

2.1 DMA CHANNEL FOR SPACEWIRE

Purpose of the DMA is to transfer data packets directly between SpaceWire interfaces and SDRAM memory in the dedicated bus independently of the CPU. Meanwhile the CPU can perform other tasks using its local cache or access any other memory space than SDRAM as is illustrated in **Figure 2**. Each SpaceWire receiver and transmitter has it’s own DMA channel to SDRAM memory.

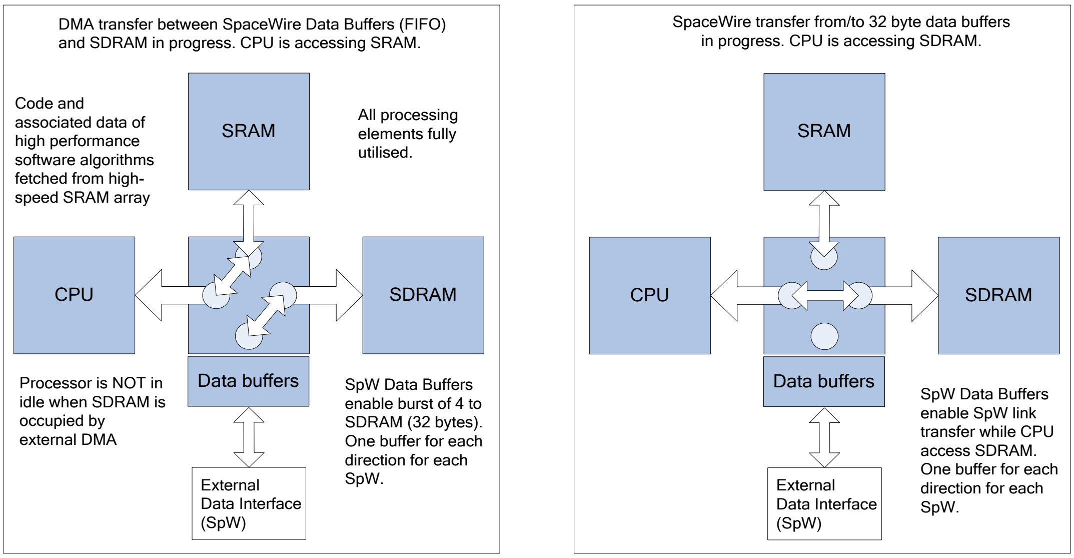
DMA transfer between SpaceWire Data Buffers (FIFO) and SDRAM in progress. CPU is accessing SRAM.
Code and associated data of high performance software algorithms fetched from high-speed SRAM array
SRAM
All processing elements fully utilised.
CPU
SDRAM
Data buffers
Processor is NOT in idle when SDRAM is occupied by external DMA
SpW Data Buffers enable burst of 4 to SDRAM (32 bytes). One buffer for each direction for each SpW.
External Data Interface (SpW)
SpaceWire transfer from/to 32 byte data buffers in progress. CPU is accessing SDRAM.
SRAM
CPU
SDRAM
Data buffers
SpW Data Buffers enable SpW link transfer while CPU access SDRAM. One buffer for each direction for each SpW.
External Data Interface (SpW)

Figure 2 DMA to SDRAM memory

In order to exploit fast data rate SDRAM burst accesses, received data is collected in 32 bytes (4 x 64-bit) buffer before written in SDRAM, and data to be transmitted is read in 32 bytes buffer from SDRAM. SpaceWire data interface buffers are depicted in **Figure 3**. Theoretical data rate is 229 Mbytes/s for SpaceWire access to SDRAM.

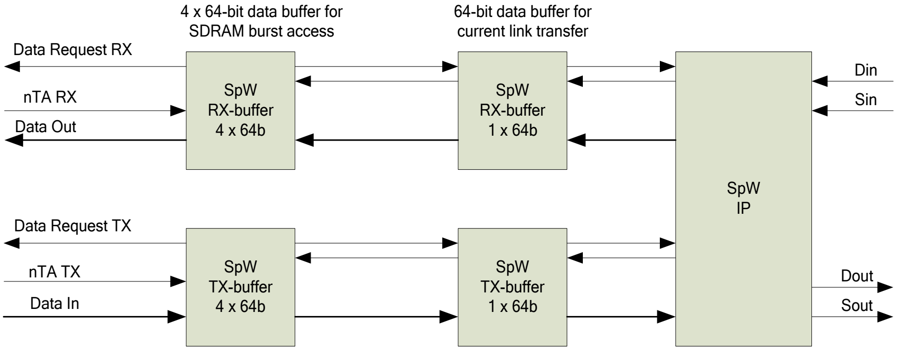

Figure 3 SpaceWire data buffers for memory burst access

2.2 DMA CHANNEL FOR RMAP

RMAP IP core has a common 32-bit bus interface for RMAP Initiator command encoder, Reply decoder and Target functions [7]. In the DPU, each RMAP core has access to SDRAM memory by the bus interface via it’s own DMA channel, as is depicted in **Figure 4**. Memory transfer is one 32-bit word (4 bytes) per an access in the current version of the DPU. This leads to 29 Mbytes/s theoretical data write rate to SDRAM and 50 Mbytes/s data read rate for RMAP access.

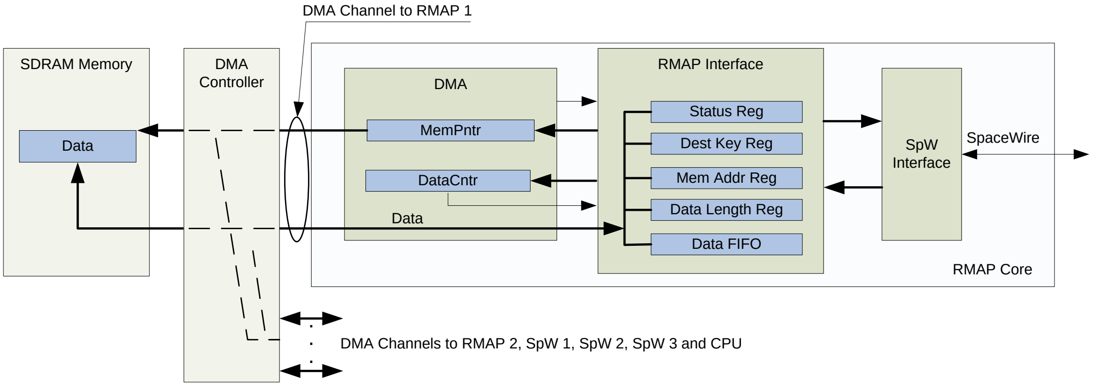

Figure 4 RMAP Core Interface to Memory

3 PERFORMANCE

3.1 CPU PERFORMANCE

CPU performance is highly dependent on whether code and data processed at the moment are located in the CPU cache or not. CPU can reach over 1000 MIPS performance during the code and data are available in the CPU cache. Cache fetches and flushes to memory are done via the 64-bit 60x bus, which operates at 50 MHz in the DPU. These operations may slow down performance considerably. On the other hand CPU has ability for branch prediction for more effective code fetch and SRAM and SDRAM interface functions effectively with burst accesses, 32 bytes are transferred per a burst. With PowerPC, burst accesses are performed when caches are enabled. In current version of the DPU, SDRAM memory is controlled by the SpWCU, which in turn is connected by a 32-bit, 50 MHz bus to the CPU.

Measured DPU power consumption is 10 W in SpaceWire loop-back test, and 8.5 W in idle. Power consumption is dependent on overall activity, i.e. activity in the CPU, SCU, SpWCU, memories and external interfaces.

3.2 SPACEWIRE PERFORMANCE

The three DPU SpaceWire links can connect and reach up to 100 Mbits/s link speed. Transmit speed is limited in the DPU to 100 Mbits/s, wherein receiver adapts to incoming link speed. Receivers have been tested successfully at 200 Mbits/s in a laboratory. Continuous 100 Mbits/s in the SpaceWire link results to 10 Mbytes/s data rate in one direction, and 60 Mbytes/s for three concurrent links full-duplex. The DPU DMA interface between SpaceWire and SDRAM memory is able to handle the rate.

Loop-back test in the laboratory measures continuous 10 Mbytes/s transfer rate between two active SpaceWire links and SDRAM. The CPU initiates every packet transmit and receive; the data rate depends on the software performance. This may explain the measured lower transfer rate than is expected.

3.3 RMAP PERFORMANCE

Every RMAP packet has a header, minimum length of 16 bytes for command and 12 bytes for reply, plus a CRC byte for data bytes in a packet [6]. Protocol overhead may be thus large when a small amount of data bytes is transferred in a packet. However, when an RMAP command is started, CPU needs to write a transaction record and a header information record in the SDRAM memory for the becoming transaction to be read by the RMAP core [7], in addition to actual data transferred in a link.

RMAP loop-back test for variable size data packets (1 byte to 512 bytes) was arranged between two DPU boards, for which 4 Mbytes/s continuous RMAP – SDRAM transfer rate was measured. Initiator board generated packets and verified data correctness of replies. Target board simply wrote the received data in memory for a write command, and read data from memory and sent it in reply packet for a read command. It was not started a new command until the current reply was checked.

RMAP Target latency, 7.9 μs, was measured as time taken from command done to reply received status in the Initiator.

4 CURRENT AND FUTURE WORK

Tests to verify correct behaviour, data rates and reliability are performed for the DPU currently. Future work might include performance enhancement by integrating functions from the Mezzanine board in the DPU main board, which would allow Inter bus i.e. CPU access to SDRAM and RMAP speed up from current 200 Mbytes/s to 400 Mbytes/s. RMAP memory access data rate could be increased by exploiting SDRAM burst transfer mode as is done with the SpaceWire interfaces.

5 ACKNOWLEDGEMENTS

The authors acknowledge the support of ESA, University of Dundee, Astrium Ltd and Patria Aviation Oy for the work.

6 REFERENCES

1. Freescale Semiconductor, “MPC7448 RISC Microprocessor Hardware Specifications”, MPC7448EC, Rev. 4, 3/2007.

2. Freescale Semiconductor, “MPC7450 RISC Microprocessor Family Reference Manual”, MPC7450UM, Rev. 5 1/2005.

3. ECSS, “SpaceWire: Links, nodes, routers and networks”, ECSS-E50-12A, January 2003

4. ECSS, “SpaceWire- Remote memory access protocol”, ECSS-E-ST-50-52C, 5 February 2010.

5. Steve Parkes, Chris McClements, Martin Dunstan, “SpaceWire RMAP IP Core”, International SpaceWire Conference 2008, Nara Prefectural New public Hall, Japan, November 2008.

6. ECSS, “Remote memory access protocol”, ECSS-E-50-11 Draft F, December 2006.

7. Space Technology Centre, University of Dundee, “SpaceNet – RMAP IP Core User Manual Issue 1.5”, RMAP IP WP2-400.3, May 2009.