Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

# The Implementation of DMA Controller on Navigation baseband SoC

Abdulraqeb Alnabihi[1]

[1] Guangdong University of Technology

*Received: 6 December 2014 Accepted: 1 January 2015 Published: 15 January 2015*

## Abstract

This article discusses the architecture design of DMA controller on high performance GPS receiver based on RTEMS. We achieve the optimal integration of DMA IP and navigation baseband system. We designed the hardware architecture of DMA IP and make full use of hardware performance with the idea of multiplexing. We use register and FIFO buffer to achieve read-write control. And we design the DMA controller with Verilog HDL. Finally we verify the design on Altera Cyclone4 FPGA. The result demonstrates that DMA controller can ease the CPU?s burden and shorten the acquisition  tracking time thus improving the performance of the whole system.

***Index terms***— DMA controller; AHB

## 1 I. INTRODUCTION

MA controller is the core component of the SoC. Through controlling the data transfer from memory to peripheral independently, it can greatly alleviate the CPU's burden and improve the efficiency of data processing. Therefore the design of DMA controller directly affects the overall performance of SoC chip. Similar to the design of other integrated circuit IP, DMA design should also consider the reusability of DMA IP and balance the operation speed and circuit area, at the same time achieving good integration with SoC. According to the design goal, the domestic and foreign scholars have conducted in-depth research. Literature [2] designs a configurable multichannel DMA controller, but it doesn't consider the size of FIFO BUFFER, which may causes the waste of resources. The design in literature [3] can alleviate the CPU's burden and improve the data transmission rate between peripherals, but it is only a single channel structure thus lack of generality. Literature [4] designs a specific data path for DMA data transfer. Although it avoids the limitation of AHB bus, achieving parallel transmission of multi-path data, but making the internal SoC bus timing complicated and consideration must be given to the arbitration of memory R/W between bus and special data bus. In this paper, according to the large amount of data of high real-time navigation applications, we design the DMA controller implementation in navigation baseband SoC. The design shortens the first positioning time of the chip and improves its whole performance.

Our Navigation Baseband SoC chooses LEON3 processor from Aeroflex Gaisler Corporation and it's based on SPARC V8 architecture. The on-chip bus is the AMBA2.0 of ARM Corporation, which ensuring the data transmission flowing. In order to realize the fast acquisition and tracking of navigation signal, the chip has 2 built-in acquisition and tracking module. The chip uses the on-chip SRAM controller to realize the real-time data storage. We use RTEMS 4.10 as the operation system, making the system with good real-time performance. At the same time, the driver program of RTEMS relative to LINUX and other embedded operating system is simpler. FIFO buffer. The size of the buffer is fixed and connects to a interrupt generating circuit. When any of the buffers is full, it generates corresponding interrupt signal and requires interrupt through the AMBA bus.

## 2 II. Design of the System Architecture

The DMA controller has three groups of control registers. Each group includes a 32 bit control register, source address register, destination address register and data register. It achieves memory W/R operation through

Memory Mapping by AHB bus and AMBA bus controller. Each signal generated by control register is sent to fixed priority arbitrator for arbitration first, the arbiter controls a mux to generate corresponding control signal.

Bus R/W module is made of AMBA writing module and AMBA reading module. It transfers corresponding address data through AMBA bus timing specification.

# 3 Design of the fifo Module

Since the The data generation rate of acquisition and tracking module is much slower than the bus read and write clock, In order to speed up the data transmission rate of DMA, we consider the use of two way asynchronous FIFO as a buffer.

The structure of the FIFO is shown as Figure 3. Because it uses different clocks to read and write, so we use Dual-port RAM as memory module. The word length and depth of storage of dual port RAM is according to the AD parameter RF front-end sampling. In order to solve the problem of metastable clock domain data transmission, FIFO read and write pointer is encoded with the gray code. Through the multi-level register transfer DMA controller can reduce the probability of metastable.

The comparison algorithm of full/empty of FIFO pointer references Clifford E. Cummings' paper. That is to construct a pointer which width is N+1, depth is 2^N bytes. Read and write pointer is represented with the gray code. The first two bits are not the same. When the FIFO is full the two LSB gets the same. When the pointer is exactly equal, FIFO is empty. Initially, the bus is idle, waiting for the DMA transmission request. When the control register Read or Write bit is 1, the DMA controller goes into corresponding reading or writing ready state. According to the setting value of the counter, DMA then decided to adopt the single or burst mode to read and write bus. After read and write data process the controller goes into finish state. Finally it determines whether goes into the read and write cycle according to the value of control register. The workflow is shown as Figure4. We use Verilog HDL to build the DMA controller, and Altera QuartusII for integrated debugging. After downloading the design to Altera Cyclone4 EP4C115F2 9C7 FPGA, we can get the correct gate level netlist. The DMA controller consumes totally 6348 LE after synthesize. We simulate the data transmission process by software interruption on RTEMS. Figure 5 shows the real-time read-burst mode waveform through Altera's Signal Tap on FPGA. We can see that DMA module successfully achieves data R/W in burst mode.

# 4 a) Experiment Design

The first positioning time of navigation baseband SoC is also called cold start time. The system powers on and initialize without any historical information under a cold start condition. And then it attempts to locate and lock the satellite. Because of the lack of prior information, it will take a long time. System uses a method similar to the polling, locking the signals from all the satellites.

The system performs four steps from power on to the output of position data: system initializationacquisition -tracking -position calculation. At the same time it interact with the user. In the four step, the processor is mainly responsible for initializing the operating system, position solution and humancomputer interactive task. The DMA controller can be used instead of the processor, responsible for store the data generated by acquisition and tracking module into RAM. Therefore the processor can focus on the position solution and human-computer interactive task and the first positioning time (TTFF) is shortened.

In order to comprehensively study the performance of the system, the test experiment selects three different IF GPS signal as signal source. The GPS S/N ratio is 42db-Hz under the first condition. The system acceleration is 0g. The GPS S/N ratio is 42db-Hz under the first condition. But the system acceleration is 20g. The GPS S/N ratio is 34db-Hz under the first condition. The system acceleration is 0g. The test results is shown in Table 1 and Table 2 We can draw from Table 1 and Table 2 that: Under the first condition, the system is under a low dynamic environment with high signal-to-noise ratio, since that the TTFF both in DMA and non-DMA mode is relatively short. And the DMA mode takes 1.1 s less time than non-DMA mode. So, DMA controller sets CPU free from busy data transforming task, making it to have more free time to perform other tasks and shortening the TTFF.

Under the second condition, the system is under a high dynamic extreme environment, which greatly increases the acquisition and tracking difficulty. Here the DMA mode takes 2.1 s less time than non-DMA mode. Compared to the first condition, Using DMA method to shorten the TTFF effect is more obvious.

Under the third condition, the system is under a environment with low signal-to-noise ratio, which also increases the acquisition and tracking difficulty. TTFF in DMA mode is 28.6s, in non-DMA mode is 26.8s, which is both longer than the first condition. Here the DMA mode takes 1.8 s less time than non-DMA mode. Implementing DMA also has the obvious effect on reducing TTFF.

The experiment proofs that: The DMA controller can effectively reduce navigation baseband SoC CPU's load, shorten TTFF.

# 5 VIII. Conclusion

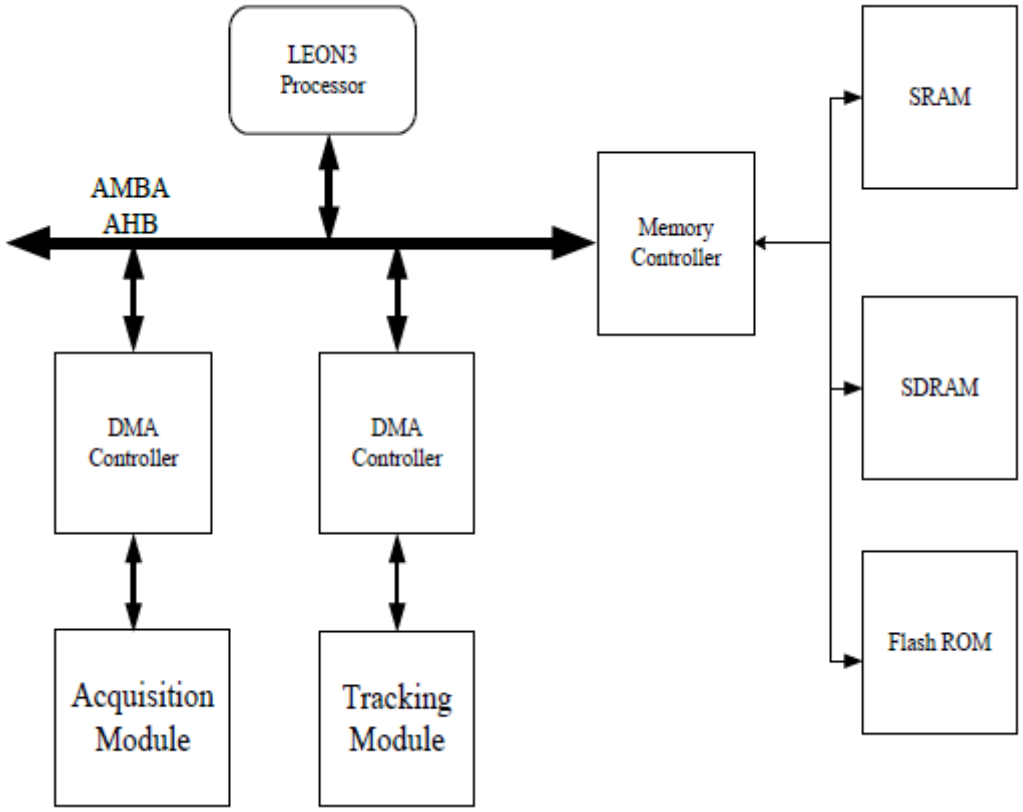
SoC technology has developed rapidly in recent years, represent the future development trend of IC. As a complex system composed by multiple IP, the single component design of GPS baseband SoC must fully consider the overall system hardware and software Global Journal of C omp uter S cience and T echnology Volume XV Issue

II Version I Year ( ) integration in order to play the best performance of the whole system. In this paper, the DMA IP from the system view, it uses FIFO BUFFER and interrupting mechanism to realize the good combination of acquisition and tracking module. With the control register group developer can easily program the DMA controller under the RTMES real-time operating system. The FPGA Experiments show that this design can realize the data read and write control based on AMBA bus, reduce the burden of CPU. [1] [2] [3]


1

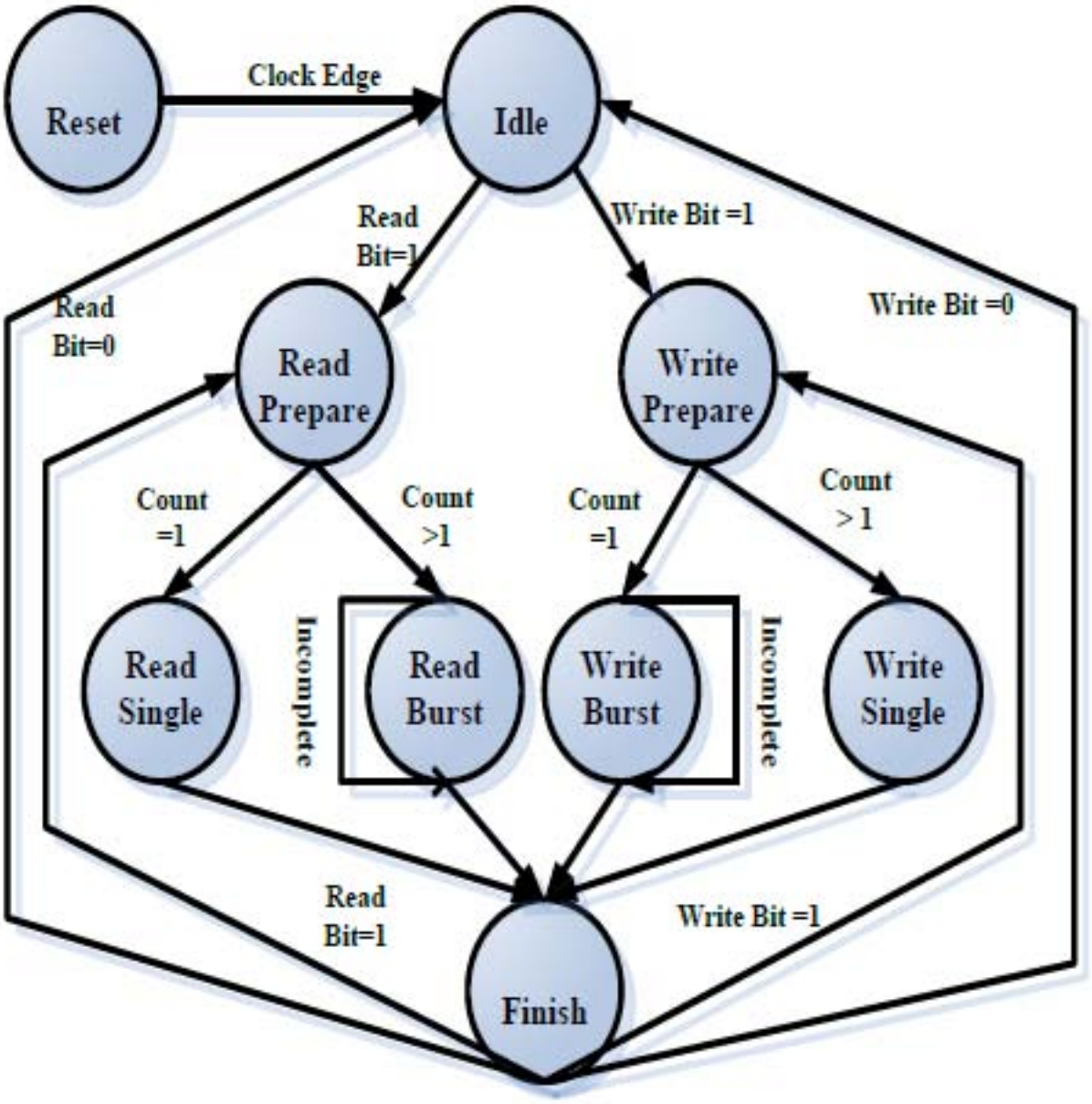
Figure 1: Fig. 1 :

[1] © 2015 Global Journals Inc. (US)
[2] © 2015 Global Journals Inc. (US) 1
[3] The Implementation of DMA Controller on Navigation baseband SoC



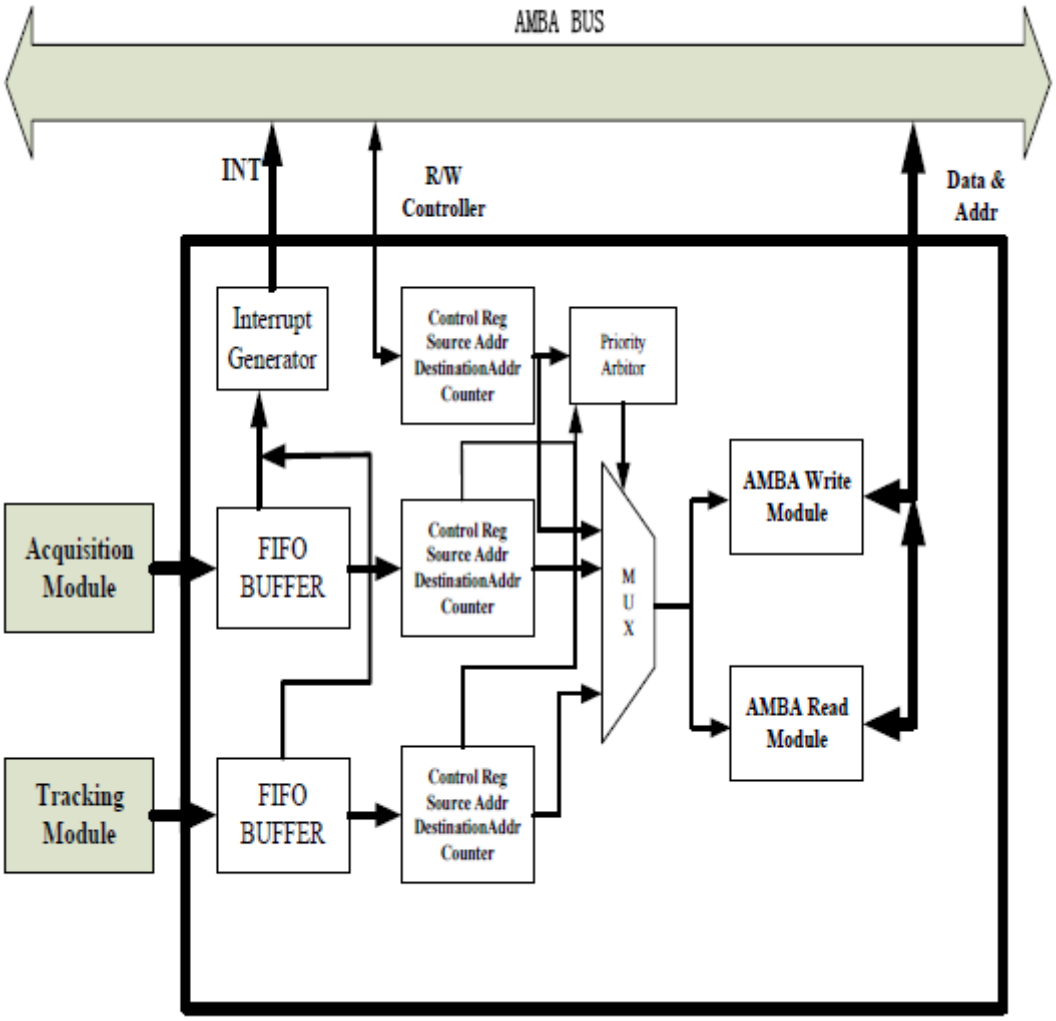

**2**

Figure 2: Fig. 2 :



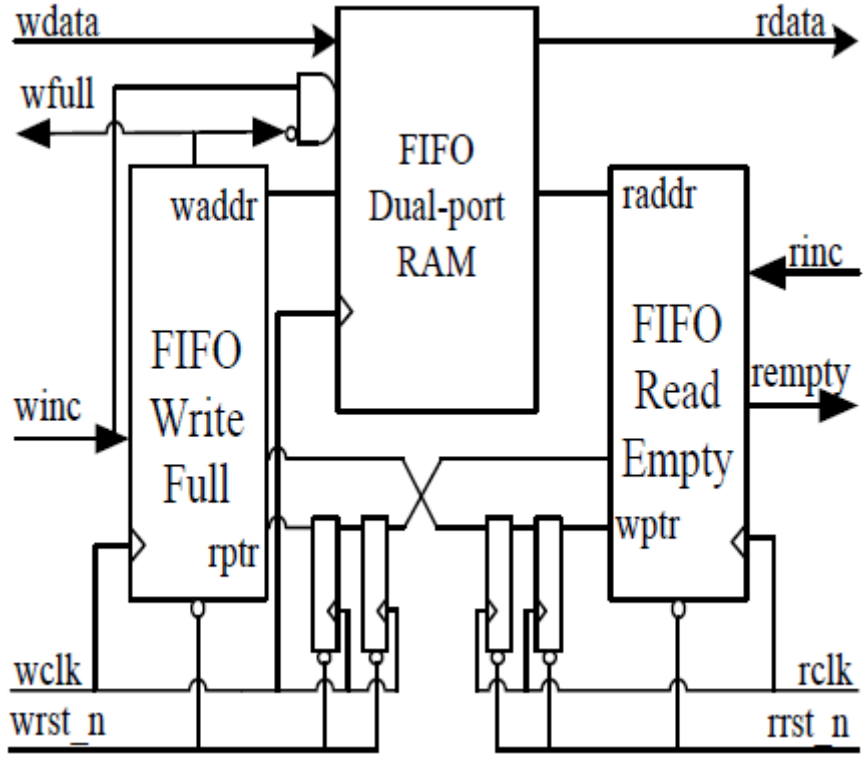

**3**

Figure 3: Fig. 3 :




**4**

Figure 4: Fig. 4 :




**5**

Figure 5: Fig. 5 :

**1**

Figure 6: Table 1 :

**2**

Figure 7: Table 2 :

[Snug ()] , Snug . 2002. 01 p. .

[Zhong-Xiu et al. ()] , Sun Zhong-Xiu , Fei Xiang-Lin , Luo Bin . 2003. Higher Education Press. (3rd Edition)

[Zhong-Ying ()] , Bai Zhong-Ying . 2005. Beijing: Science Press.

[ Microelectronics Computer ()] , *Microelectronics & Computer* 2007. 24 (5) p. .

[Arm L Td ()] *AMBA Bus Specification*, Arm L Td . 1999. p. . (UK: ARM Ltd)

[Chen Shuang-Yan et al.] *Design and Implementation of a Configurable Multi-Channel DMA Controller Based on SystemC*, Wang Chen Shuang-Yan , Dong-Hui , Hou Chaohuan .

[Lu-Yu et al. ()] 'Design of a multi-interface DMA controller based on SoC'. Zhang Lu-Yu , Li Li , Pan Hong-Bing , Wang Kun , Li Wei . *Electronic Measurement Technology* 2014. 37 (9) p. .

[Gaisler ()] *GRLIB IP Library User's Manual [Z]. Sweden: Aeroflex Gaisler Ltd*, Aeroflex Gaisler . 2013. p. .

[Bing et al. ()] 'NAND Flash DMA Application Based on PXA3xx Processor'. Shi Bing , Ding Zhi-Gang , Zhang Wei-Hong . *Journal of Computer Applications* 2009. 29 (8) p. .

[Cummings] *Simulation and Synthesis Techniques for Asynchronous FIFO Design*, Clifford E Cummings .

[Jian-Bo ()] *The Design and Verification of DMA Controller based on AMBA*, Geng Jian-Bo . 2013. Master Dissertation. Xidian University

[Qiang] *The Design of DMA Controller based on AHB Bus Specification*, Zhao Qiang .

[Mu-Guo et al. ()] 'The Design of DMA for PCIe Bus Based on FPGA'. Li Mu-Guo , Huang Ying , Liu Yu-Zhi . *Computer Measurement & Control* 2013. 01 p. .

[Yun] *The Design of DMA Platform for PCIe Bus Based on FPGA*, Wei Yun .

[Yong et al. ()] 'The Hardware Application of High Performance DMA Controller in AMBA Bus Architecture'. Xie Yong , Shen Ming , Zheng Jian-Hong . *Journal of Chongqing Institute of Technology: Computer and Automation Column* 2006. 20 (8) p. .

[Hao Jun] *The Simulation and Verification of DMA Controller*, Hao Jun .

[Yu-Wen ()] Xia Yu-Wen . *Verilog Digital System Design Tutorial*, 2008. BEIHANG UNIVERSITY PRESS. (2rd Edition)