Low Power System-on-Chip Platform Architecture for High Performance Applications

W.-C. Lo[1], A.T. Erdogan[2] and T. Arslan[1,2]

[1]*Institute for System Level Integration, The ALBA Campus, Livingston, EH64 7BH, Scotland, United Kingdom.*

[2]*University of Edinburgh,*
Department of Electronics & Electrical Engineering,
Edinburgh EH9 3JL, Scotland, United Kingdom.

Abstract: This paper describes work on the development of a scheme for implementation of low power high performance Digital Signal Processing intensive AMBA based System-On-Chip platforms. The scheme is based on a novel interfacing scheme which utilises the bus hierarchy within AMBA in order to allow single and multiple high performance DSP Intellectual Property cores to be integrated into the SoC platform. The paper describes the overall AMBA SoC architecture, the integration scheme, and an SoC generic wrapper which allows multiple low power high performance DSPs to be connected to the platform. Initial results are provided with a low power FIR filter core demonstrating impact of power on various sections of the platform.

Key words: Low power, System on a Chip, DSP

1. INTRODUCTION

Today's System-On-Chip (SoC) devices are targeting complex multi-media applications where there is a need for significant amount of Digital Signal Processor (DSP) computing power in order to achieve the various data processing tasks which could include both video and audio. In order to complete these tasks various DSP Intellectual Property (IP) cores, which are optimised for the application in hand, are utilised. In order to satisfy the constraint of fast time to market effective platform architectures are required. High performance IP cores can be connected to the above platforms on a plug-in basis in order to maintain performance constraints as well as that of the fast time to market [1][2].

Although numerous researchers in the literature have considered the development of interfaces and wrappers [3][4], only a few discuss implementation issues and impact at the SoC level. This paper will discuss the implementation of an efficient interface strategy and its effect on the overall system. The paper describes the development of an SoC platform for low power wireless applications. The platform utilises the LEON Processor, which is a general SPARC compatible processor [5].

In order to integrate a number of DSP IP cores, an interface is required. This interface connects the DSP IPs directly to the Advance Microcontroller Bus Architecture (AMBA) [6] of the LEON Processor. In comparison to the bus interface design that contains a Virtual Component Interface (VCI) [7], the AMBA interface described in this paper eliminates the VCI overhead. As discussed in [8], it is less efficient to design extra VCI for both the bus interface and DSP IPs. Therefore, the AMBA interface discussed in this paper connects the DSP IPs directly to the processor without any additional control signals. This is achieved through gated clock and look ahead features. At the same time, these features have made the interface consumed less power.

Gated clock has been employed normally to selectively stop the clock in portions of circuits where active computation is not being performed [9] [10]. In [9], the Finite State Machine (FSM) is partitioned into several sub-FSMs and these sub-FSMs are clocked when necessary. However, the AMBA interface discussed in this paper is partitioned into Bus Address and Control Decoders, a AHB Data FSM and a Clock Controller. The AHB Data FSM is clocked when there is data available from the processor and the selected DSP IP is clocked when the data is available from the interface.

The DSP IP core used in this paper is an FIR filter that has a transpose direct form structure [11][12]. This structure reduces the switching activity at data inputs of the multiplier. The data input of the multiplier remains unchanged until it is multiplied by all the coefficients. In addition, the transpose direct form structure can be exploited by different numerical ordering algorithms for manipulating filter coefficients in order to reduce the switching activity at the coefficient input of the multiplier.

2. PLATFORM ARCHITECTURE

The overall SoC platform architecture is demonstrated in Figure 1. The architecture is built around the LEON processor, and includes a number of high performance DSP IP cores.

The LEON processor is connected to the AMBA interface through the AHB (Advance High Performance Bus) and the APB (Advance Peripheral Bus). Within the LEON processor, the AHB is intended for high clock frequency system modules and enables access to high bandwidth memory devices, while the APB is optimised for minimal power consumption and suits low bandwidth peripheral modules. Any data that the LEON processor sends or receives (from the AMBA interface and to the DSP IP core) will be through these buses. The AMBA interface functions as a middleman that monitors the AHB and APB buses for any possible transaction being directed to the DSP Core. If a read or write operation is required, then the AMBA interface will load or store data (from the AHB or APB buses) to the DSP IP respectively. The DSP IP only accepts input data, processes this data and outputs the resulting data. Therefore, the AMBA communication protocol with the LEON processor is transparent to the DSP IP core as these protocols are handled by the AMBA interface. Likewise, the DSP IP's clock is actively controlled by the AMBA interface to save power.

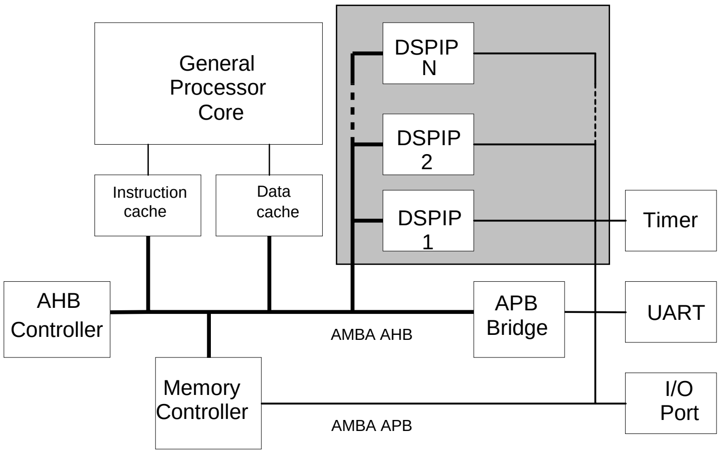

Figure 1. The Low Power Wireless Processor

3. THE AMBA INTERFACE

As illustrated in Figure 1, the scheme in this paper can allow multiple high performance IP cores to be attached to the platform. In this paper only a single DSP IP core is used to demonstrate the above scheme. The DSP IP core is an FIR filter. An interface is required to handle the AMBA communication protocols on behalf of this core. This interface is implemented as both an AHB slave as well as an APB device. Inputs into the interface must go through the AHB, while outputs from the interface can go through AHB or APB, depending on the urgency of output data required by the LEON processor.

An AMBA-based low power clocking scheme is devised for the interface to reduce the switching activity in the AMBA interface and the connecting DSP IP Cores. This clocking scheme is based on a 3 level hierarchy of clocks, where the relevant module is clocked when required**.**

The AMBA interface is first partitioned into 4 modules, consisting of the AHB Address and Control Decoder, AHB Data FSM, APB Address and Control Decoder, and the Clock Controller. This is shown in Figure 2.

The AHB address and control decoder will decode all the address and control signals from the LEON processor and send a signal to the AHB FSM to indicate its next state operation for the next clock cycle. This is possible due to the pipelined nature of the bus where the address and data are overlapped to allow high performance operation. The decoded address also enables the decoder to decide whether to clock the FSM and the connecting cores, by sending signals to the clock controller.

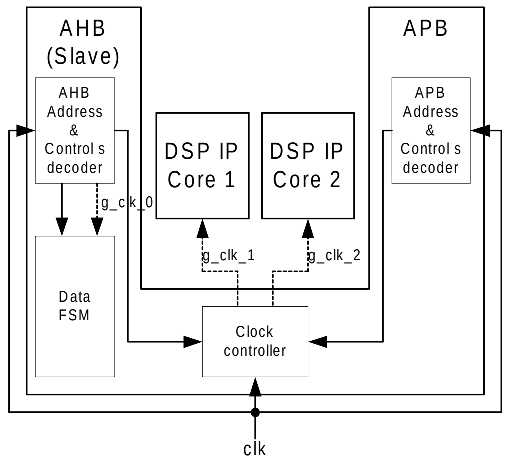

Figure 2. Partitions of the AMBA interface with 2 DSP IPs

The AHB FSM reacts to the signals send by the AHB address and control decoder, and routes the AHB write data bus to the input of the selected DSP IP core, or routes the data from the output of the selected core back to the AHB read data bus. This AHB FSM is clocked based on the gated clock signal from the clock controller.

The APB address and control decoder decodes the address and control signals send by the APB Bridge. It will only respond to read operations while write operations are ignored. During a read operation, the output data from the selected core will be routed back to the APB read data bus, and transferred back to the APB Bridge. Since this decoder only handles output data from the cores, a FSM is therefore not required.

The Clock controller will determine whether to clock the cores based on the signals from the AHB or APB address and control decoders. An AHB read or write operation will trigger the AHB address and control decoder to send signals to the Clock controller, indicating which DSP IP core has to be clocked. Meanwhile, an APB read operation will trigger the APB address and control decoder to send signals to the Clock controller to clock the selected core.

4. RESULTS AND DISCUSSION

The whole system was simulated at RTL level with Mentor Graphics' ModelSim simulator. A section of the timing diagram from the system is shown in Figure 3. Due to the pipelined nature of the AHB operations, the data arrives on the next clock cycle following the address. Therefore, it gives the AHB Address and Control decoder 1 clock cycle time to anticipate and decide the next clock operation as well as knowing which relevant DSP IP Core to clock.

Using Figure 3 as an example, the address of 0xA3000000 indicates that on the next clock cycle, the data of 0x000042DE has to be transfer from the AHB write data bus to the DSP IP input. The AHB address and control decoder clocks AHB FSM twice. First time is to load the FSM's next state, while the second time is to transfer the next state to current state. At the same time, the AHB or APB address and control decoder sends signal to the Clock controller to request for 1 clock cycle for the selected DSP IP core to load the data to its input.

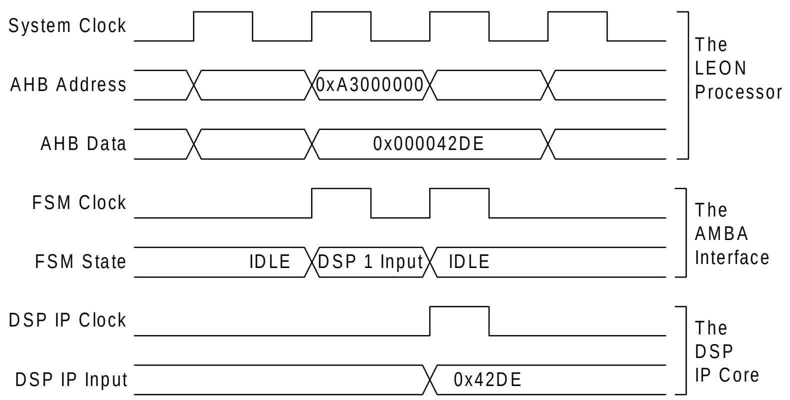

Figure 3. Timing diagram for data transfer from the LEON Processor to the DSP IP via the AMBA interface

The power analysis was then performed with Synopsys Power tools. The power distribution of the main blocks of the system is shown in Figure 4. Clearly, the most power is consumed by the LEON processor, followed by memory controller and the timers.

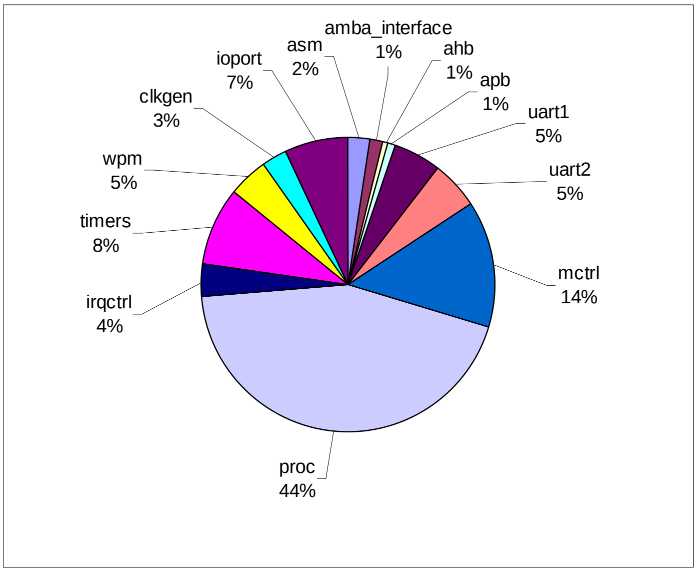

Figure 4. The Power Distribution of the Main Blocks

proc:	*Processor and cache sub-system*
mctrl:	*Memory controller*
ahb:	*AHB arbiter/decoder*
apb:	*AHB/APB bridge*
uart:	*Universal receiver/transmitter*
asm:	*AHB status register*
wpm:	*AHB write protection register*
irqctrl:	*Interrupt controller*

5. CONCLUSION

The authors have presented a low power platform for the implementation of high performance DSP intensive tasks. In addition, a generic wrapper architecture was demonstrated to integrate DSP IP cores to the platform.

Simulations were carried out with an FIR filtering IP core integrated on to the platform. Results were provided for timing and power consumption of the whole SoC platform. The work is subject of on going research. Current work involves synthesis and simulation of complete systems with multiple high performance DSP IP cores which include blocks for audio and image manipulation together with IP cores for transmitting these over a range of channels.

6. REFERENCES

[1] HUANG, J.R., IYER, M.K., CHENG, K.T.: 'A self-test methodology for IP cores in bus-based programmable SoCs', VLSI Test Symposium, 19th IEEE Proceedings on VTS 2001, 2001 pp. 198 –203.

[2] WILTON, S.J.E., SALEH, R.: 'Programmable logic IP cores in SoC design: opportunities and challenges', *Custom Integrated Circuits, 2001, IEEE Conference on 2001* pp. 63 –66.

[3] KIM, K.W., KWANG, H.B., SHANBHAG, N., LIU, C.L., KANG S.M.: 'Coupling-driven signal encoding scheme for low-power interface design', *Computer Aided Design, 2000. ICCAD-2000. IEEE/ACM International Conference on*, 2000, pp. 318 -321.

[4] YOO, S.J.; NICOLESCU, G., LYONNARD, D., BAGHDADI, A., JERRAYA, A.A.: 'A generic wrapper architecture for multi-processor SoC cosimulation and design', *Hardware/Software Codesign, 2001. CODES 2001. Proceedings of the Ninth International Symposium on*, 2001 pp. 195 -200.

[5] GAISLER J.: 'The LEON Processor User's Manual', Version 2.3.7, August 2001, Gaisler Research, www.gaisler.com.

[6] AMBA™ Specification, Revision 2.0, May 1999, © ARM Ltd., www.arm.com/Pro+Peripherals/AMBA

[7] HELLMICH, H.H.H., ERDOGAN, A.T., ARSLAN, T. 'Re-Usable Low Power DSP IP embedded in an ARM based SoC Architecture', *IEE Coloquim on Intelectual Property*, Edinburgh, UK, July 2000, pp. 10/1 - 10/5.

[8] YEUNG, C., MATTHEWS, G., MORRIS, J., HAVERINEN, A., ZAIDI, J.: 'Standard bus vs. bus wrapper: what is the best solution for future SoC integration?', Design, Automation and Test in Europe, 2001. Conference and Exhibition 2001. Proceedings, 2001 pp. 776 - 776.

[9] OELMANN, B., O'NILS, M.: 'Asynchronous control of low-power gated-clock finite-state-machines', *Electronics, Circuits and Systems, 1999. Proceedings of ICECS '99. The 6th IEEE International Conference on*, Volume: 2, pp. 915 -918.

[10] WU, Q., PEDRAM, M., WU, X.W.:'Clock-gating and its application to low power design of sequential circuits', Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on, Volume: 47 Issue: 3, March 2000, pp. 415 -420.

[11] ERDOGAN, A.T., ARSLAN, T.: 'Low power multiplication scheme for FIR filter implementation on single multiplier CMOS DSP processors', *Electronics Letters,* Volume: 32 Issue: 21, 10 Oct. 1996, pp. 1959 -1960.

[12] ERDOGAN, A.T., ARSLAN, T., HORROCKS, D.H.: 'Low power multiplication schemes for single multiplier CMOS based FIR digital filter implementations', *Circuits and Systems, 1997. ISCAS '97., Proceedings of 1997 IEEE International Symposium on* , Volume: 3 , 1997 pp. 1940 -1943 vol.3.