Design and Implementation of AMBA Bridge Protocol in System on Chip Design

Wang Hang Suan¹, Asral Bahari Jambek², Mohd Nazrin Bin Md Isa³, Azizi bin Harun⁴, Shaiful Nizam Bin Mohyar⁵, Zulfiqar Ali Bin Abd Aziz⁶

¹, ², ³, ⁴, ⁵ School of Microelectronic Engineering, University Malaysia Perlis, Perlis, Malaysia.
⁶School of Electrical and Electronic Engineering, University Sains Malaysia, Nibong Tebal, Penang.

ABSTRACT

The Advanced Microcontroller Bus Architecture (AMBA) is widely used in modern technology device. The design of bridge in the system is due to increase demand of power consumption and functionality. The bridge help to reduce power by separate the system into high bandwidth and low bandwidth. The goal of this paper is to design and implement the AMBA bridge into a SoC design which consists of a processor, RAM, ROM, watchdog and LED module. These peripherals are connected separately based on different bandwidth with a bridge as the medium. The result shown the bridge transfer a correct data from the ROM into the RAM. The experiment was carry out using Synopsys 2017 and Keil uVision.

Keywords:
ARM
Cortex-M0 SDK
AMBA
APB bridge
System-on-Chip

1. INTRODUCTION

The Advanced Microcontroller Bus Architecture (AMBA) is widely used in many SoC microcontroller and devices. This type of AMBA has a standard interface [1] which is efficient for System on Chip interconnect [2]. The designer just need to design the IP that support AMBA protocol and able to use the buses that will reduce the time to market. The AMBA also can support burst and sequential mode which is suitable for fast memory access.

There are many type of buses in AMBA family such as Advanced Extensible Interface (AXI), Advanced High-performance Bus (AHB), Advanced System Bus (ASB) and Advanced Peripheral Bus (APB). Bridge is one of the important component to connect a high bandwidth bus to a low bandwidth bus for minimum power consumption [3]. The bridge not only need to ensure no data loss during the transfer but the bridge also need to make sure the data transfer between the buses are correct. The bridge will act as master for the APB bus and control the transfer sequence after receive command from AHB bus as shown in figure 1. The bridge is using finite state machine to determine the stage of transfer and only one peripheral will be selected at one time [4]. There are three stages in the AMBA bridge which are idle stage, setup stage and enable stage.

In this paper, an AMBA bridge has been design to ensure the bridge can support different frequency. The APB frequency using half frequency of the AHB. This two clock is synchronize, in other word, the transmit stage can only be occurred when positive edge of clock. The design was tested using one RAM and ROM in the APB side. The value of the ROM is read and write into the RAM to show the result in simulation.
In paper [5], the author design a different phase and frequency AMBA bridge. The design of the bridge is follow AMBA protocol which the control signal from AHB is delay to APB. The design using handshake signaling method to communicate between AHB and APB. However, this two bus has its own clock frequency, therefore the design include double stage synchronizer inside the bridge to avoid setup and time violation as shown in figure 2.

The design is verify by using module AHB driver and monitor to act as a master for those buses. These control signals and clock are inserted through an internal memory that act as a bus master. This bus master will write some data to the bridge and receive it after passing through bridge and compare the result. While the APB driver and monitor also insert the signals received from the bridge. The result was successfully synthesized.

In paper [6], the author design a low power AMBA bridge. The design consists of a bridge, and a secure UART as the APB slave. The design also have a few states to determine the data transfer stage as shown in figure 3.

The bridge will keep monitor the signal in the idle state. The signal will go write state or read state depend on the HWRITE signal. If the states is in read states, PSEL will insert. The next state will be READOK allow the data to fetch back to the processor. Then the state will go back to idle state. The same process will go through in the write state. The result also shown that the AMBA 3 specification manage to reduce 6% power and 10% area.
Apart from that, the author in paper [7] had design an IIC IP with Avalon bus. The design combine with other IPs in Quartus II using System on Programmable Chip (SOPC) to become a complete system. The design was verify using Atmel IIC memory.

The SOPC will handle and generate the best Avalon bus for the design. The design include a bus controller, state machine and read and write controller as shown in figure 4. The design system include processor, memory, PIO, and JTAG. The author then assign the base address and interrupt settings for the peripheral. Next, the SOPC system was generated. A software was included to verify the functionality of the IIC IP core by read and write operation on the IIC memory. The result show that the design can be fast generate using SOPC and shorten the development cycle.

![Figure 4. The design of the IIC IP [7].](image)

In paper [8], the author proposed a two layer of buses working based on the Wishbone structure. The double bus interconnect with the peripherals based on the speed of the device. The main purpose for this type of architecture is because the convention wishbone bus does not differentiate high speed bus and low speed bus. This will cause the whole system to run at the speed of the slowest peripheral.

The design then was connect with different devices to form a system based on the double bus. The interconnect structure for the bus is a interconnect matrix which utilize the crossbar switch interconnection structure. It allow different master and different slave communicate simultaneously based on the priority.

The design include a bus bridge between the wb_conmax to encode the address and synchronize the clock. The system was then targeted to Xilinx for verification as shown in figure 5. The result show that the proposed design increase the logic gate but reduce power consumption.

![Figure 5. The two layer bridge [8].](image)

The other author [9], had propose an off-the-shelf microcontroller SoC system for system control and support function for complex computer node. The system consists of PowerPC 440 processor, digital Input Output, IIC, JTAG and etc. The system use IBM CoreConnect as internal bus communication. The System have the capability to connect Ethernet based and run the application in order to access the system management units.

The design also include hot plug for the exchange of this microcontroller without affect the system operation. The system can operate simultaneously or as master and slave mode. The slave will replace the master to continue the task if the master fault. The bus use in the design is Processor Local Bus (PLB) which based on CoreConnect architecture as shown in figure 6.
Figure 6. The off-shelf microcontroller system [9].

The bus has a high bandwidth with 166MHz, 128 bits data bus and a round-robin scheme to select the devices. The design ware verified using Cadence Simulator and Cadence Palladium II hardware accelerator.

In conclusion, after a few paper have been reviewed, some aspect of the bus was compared as shown in Table I. Most of the buses is an open architecture, which mean anyone is allow to build the code. The design of SoC require a standard bus signal in order to achieve time-to-market. Besides, the bus architecture need transfer data without any error for different frequency.

The buses use to compare are AMBA, Avalon, Wishbone and IBM CoreConnect. The AMBA has 32 bits data bus and can support up to 100MHz which is suitable for microcontroller. Meanwhile, Coreconnect has 128 bits and can support 166MHz is suitable for high end device. It is found that the AMBA hold more advantage than other type of buses for certain application. Therefore, for this paper, AMBA bus and bridge was chosen as the research purposes for small application. Design such as AES [10] with AMBA interface can be added easily as they have standard interface.

<table>
<thead>
<tr>
<th>Features</th>
<th>[5, 6]</th>
<th>[7]</th>
<th>[8]</th>
<th>[9]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of bus</td>
<td>AMBA</td>
<td>Avalon</td>
<td>Wishbone</td>
<td>CoreConnect</td>
</tr>
<tr>
<td>Topology</td>
<td>Shared bus</td>
<td>Avalon MM</td>
<td>Crossbar switch</td>
<td>PLB</td>
</tr>
<tr>
<td>Arbitration</td>
<td>Not used</td>
<td>Not used</td>
<td>Parameterized</td>
<td>Round Robin</td>
</tr>
<tr>
<td>Frequency</td>
<td>100MHz</td>
<td>100MHz</td>
<td>100MHz</td>
<td>166MHz</td>
</tr>
<tr>
<td>Development cycle</td>
<td>Fast</td>
<td>Fast</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>License</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

2. RESEARCH METHOD

In this paper, a system with a processor, an AMBA bus and a bridge are designed. The system using 2 APB module act as RAM and ROM to test the bridge [11][12]. The bridge is design to handle two different frequency, which the APB frequency is half of the AHB.

The bridge has been modify from the ARM bridge with the decoder is build inside on it. The SRAM will later store the application program to verify the design. The LED IP in this design also act as a verification reference to check whether the bridge select the intended IP. The system design is illustrated in figure 7.
Next, this IP is integrated into the system. The memory map is designed as shown in Table 2. The RAM is designed with only one address register with 32 bits. The RAM will store result temporarily which can be increased in width if needed. Besides, the ROM is designed with only four address registers which store some data which is 0x40, 0x30, 0x20, and 0x10 as shown in Table 3. The bus will choose the address based on the application program. It is written using ARM assembly instruction set [13].

Table 2. IP base address

<table>
<thead>
<tr>
<th>Module</th>
<th>Base Address</th>
<th>End Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM</td>
<td>0x0000_0000</td>
<td>0x4FFF_FFFF</td>
</tr>
<tr>
<td>LED</td>
<td>0x5000_0000</td>
<td>0x50FF_FFFF</td>
</tr>
<tr>
<td>RAM</td>
<td>0x5100_0000</td>
<td>0x51FF_FFFF</td>
</tr>
<tr>
<td>ROM</td>
<td>0x5200_0000</td>
<td>0x52FF_FFFF</td>
</tr>
<tr>
<td>Watchdog</td>
<td>0x5300_0000</td>
<td>0x53FF_FFFF</td>
</tr>
</tbody>
</table>

Table 3. ROM address

<table>
<thead>
<tr>
<th>Module</th>
<th>Base Address</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROM</td>
<td>0x5200_0000</td>
<td>0x40</td>
</tr>
<tr>
<td></td>
<td>0x5200_0004</td>
<td>0x30</td>
</tr>
<tr>
<td></td>
<td>0x5200_0008</td>
<td>0x20</td>
</tr>
<tr>
<td></td>
<td>0x5200_000C</td>
<td>0x10</td>
</tr>
</tbody>
</table>

The next step is using a Hex file to verify the design using simulation waveform [14]. The Hex file for the processor to run the program is created using Keil software. The Hex file is only binary code which can only be understood by the processor. There are two steps to run the program. First, the program writes a value and a delay for the LED IP to act as a blinking LED. Then, the RAM stores the data from SRAM in AHB side which is written in the program. While in next step the ROM data that pre-stored in the Verilog is stored into the RAM.

The design of this system uses AMBA bus as the internal communication. The design of RAM and ROM contains APB interface to communicate with AMBA.

3. RESULTS AND ANALYSIS

The simulation result using Synopsys is shown in figure 8. The result shows the RAM stores the result 0xF and 0x8 in the first two data sets. This store is to prove that the bridge is capable to write data to the intended peripherals. The RAM then stores the value from ROM which are 0x40, 0x30, 0x20 and 0x10. It shows that the bridge is able to read data from its peripheral and transfer to other device.

It can be noticed that the PCLK is half of the frequency of clk or HCLK. It can also be noticed that the address is point to the intended IP while transferring the data. For example, the 0x30 was read based on the address which is 0x5200_0004 (red circle). The bridge basically controls the PENABLE, PSEL, and PWRITE. These signals are purposely extended until the next cycle for synchronous purposes.

The result shows that the data transfer as intended even with different clock frequency. The processor Cortex-M0 is obtained from ARM official website [15][16] through ARM University Program.
4. CONCLUSION

This paper has presented the design of a system with a bridge. The main goal of this effort is to design a bridge that can transfer data between two buses with different frequency. This will reduce some power consumption. The result also shown that the system had transfer the data accurately. In future study, the RAM and ROM IP can be substitute with a more specific purpose IP for a small application.

ACKNOWLEDGEMENTS

This research was part of 3D (Demand, Driven, and Development) program supported by TalentCorp Malaysia.

REFERENCES

BIOGRAPHIES OF AUTHORS

Wang Hang Suan is graduated with B.Eng (Hons.) Electronic Engineering from University Malaysia Perlis (UniMAP) in 2016. Currently, he is continuing his M.Sc in the same university. He previously did his internship at Inari Sdn Bhd, and was responsible to help engineer to setup an electronic test machine (ATE). His research interest is on full chip RTL integration for system-on-chip design.

Asral Bahari Jambek is a member of the School of Microelectronics Engineering, Universiti Malaysia Perlis (UniMAP), and was a Programme Chairperson for the Electronics Engineering Degree Programme, UniMAP. He has more than 15 years experience in integrated circuit and system design in both the industry and academic sectors, and has been involved at various levels of VLSI design such as transistor modelling, digital circuit design, analogue circuit design, logic synthesis and physical place and route, architecture design and algorithm development. Currently, he is actively researching new techniques to minimize power consumption in portable system-on-chip design. His research interests include integrated circuits and systems design, digital signal processing (DSP), low power algorithms and architectures design, and image and video processing.