IMPLEMENTATION OF AHB BUS TRACER WITH DYNAMIC MULTIRESOLUTION FOR LOSSLESS REAL TIME COMPRESSION

A.Mallaiah[1], G.Harish[2], M.Kamaraju[3], T.Subhashini[4]

[1]*Associate Professor in ECE Department, Gudlavalleru Engineering College, Gudlavalleru, A.P.INDIA.*
malli797@gmail.com
[2]*M.Tech in Embedded Systems, Gudlavalleru Engineering College, Gudlavalleru, A.P.INDIA.*
bobbyharish440@gmail.com
[3]*Professor in ECE Department, Gudlavalleru Engineering College, Gudlavalleru, A.P.INDIA.*
profmkr@gmail.com
[4]*Asst.Professor in ECE Department, Sri Vasavi Institute of Engineering & Technology, Pedana, A.P.INDIA.*
subhashini.anagani@gmail.com

Abstract

In the system-on-chip (SoC) debugging and performance analysis/optimization, monitoring the on-chip bus signals are necessary. But, such signals are difficult to observe since they are deeply embedded in a SoC and no sufficient I/O pins to access those signals. Therefore, we embed a bus tracer in SoC to capture the bus signals and store them.The stored trace memory can be loaded to the trace analyzer software for analysis. In this paper a multiresolution AHB On-Chip bus tracer is proposed for system-on-chip (SoC) debugging and monitoring which is capable of capturing the bus trace with different resolutions and efficient built-in compression mechanisms.In addition, it allows the user to switch the trace resolution dynamically so that appropriate resolution levels can be applied to different segments of the trace. It also supports tracing after/before an event triggering, named post-triggering trace/pre-triggering trace, respectively. The SoC can be verified in field-programmable gate array.

Index Terms*— AHB, Compression, Multi resolution, post-T trace, pre-T trace, SoC debugging.*

---------- *** ----------

1. INTRODUCTION

The Major hardware components are connected by the On-Chip bus. The size of the bus trace is said to be increases rapidly as the number of transactions are more. Therefore, in order to reduce the trace size the compression of the trace is necessary. In order to achieve high trace compression ratio the bus tracer adopts trace compression mechanisms. It supports multi resolution tracing by capturing traces at different timing and signal abstraction levels. It also provides the dynamic mode change feature to allow users to switch the resolution on-the-fly for different portions of the trace to match specific debugging/analysis needs. This feature provides a more flexible tracing to focus on the interesting points.

2. RELATED WORK

There are hardware approaches to compress the traces which are the lossy trace compression approach and lossless trace compression approach[1]. Debugging of software applications, distributed over several, possibly heterogeneous processor cores [2], [3].ARM provides the AMBA AHB trace macrocell (HTM) [4] that is capable of tracing AHB bus signals, including the instruction address, data address, and control signals. The bus signals are characterized into three categories: program address, data address/data and control signals. Dictionary based compression reduces the size of repeated data[5]. In the Embedded Trace Macrocell (ETM) [6] of ARM, slice compression takes advantage of the fact that the high-order bits of instruction addresses rarely change. The real-time compression and compression ratio achieved by using typical existing approaches [7]. Tabbara and Hashmi [8] propose the transaction-level SoC modeling and debugging method. The bus checker can check bus transaction if they obey bus protocol or not [10]. FS2 AMBA Navigator [13] supports bus clock mode and bus transfer mode to trace bus signals on every clock and bus transfer respectively.

The transaction-level debugging provides software and hardware designers a common abstraction level to diagnose bugs. The abstraction level is in two dimensions: timing abstraction and signal abstraction. The timing dimension has two abstraction levels which are the cycle level and transaction

level. The cycle level captures the signals at every cycle. The transaction level records the signals only when their value changes. The signal dimension involves grouping of AHB bus signals into four categories: program address, data address/value, access control signals (ACS), and protocol control signals (PCS). Then, we define three abstraction levels for those signals. They are full signal level, bus state level, and master operation level. The full signal level captures all bus signals.

The bus state level further abstracts the PCS by encoding them as states according to the bus-state-machine (BSM).The states represent bus handshaking activities within a bus transaction. The master state level further abstracts the bus state level by only recording the transfer activities of bus masters and ignoring the handshaking activities within transactions. This level also ignores the signals when the bus state is IDLE, WAIT, and BUSY. The BSM is designed based on the AMBA AHB 2.0 protocol to represent the key bus handshaking activities within a transaction. The transitions between BSM states follow the AMBA protocol control signals. Combining the abstraction levels in the timing dimension and the signal dimension, we provide five modes in different granularities.

They are Mode FC (full signal, cycle level), Mode FT (full signal, transaction level), Mode BC (bus state, cycle level), Mode BT (bus state, transaction level), and Mode MT (master state, transaction level). At Mode FC, the tracer traces all bus signals cycle-by-cycle so the detailed bus activities can be observed. At Mode FT, the tracer traces all signals only when their values are changed. At Mode BC, the tracer uses the BSM, such as NORMAL, IDLE, ERROR, and so on, to represent bus transfer activities in cycle accurate level. At Mode BT, the tracer uses bus state to represent bus transfer activities in transaction level Our bus tracer also supports dynamic mode change (DMC) feature which allows designers to change the trace mode dynamically in real-time.

The post-T trace captures signals after a triggering event, while the pre-T trace captures signals before the triggering event. The post-T trace is usually used to observe signals after a known event. The pre-T trace is used to diagnose the cause for unexpected errors by capturing the signals before the errors. In order to overcome the problem, we adopt a periodical triggering technique.

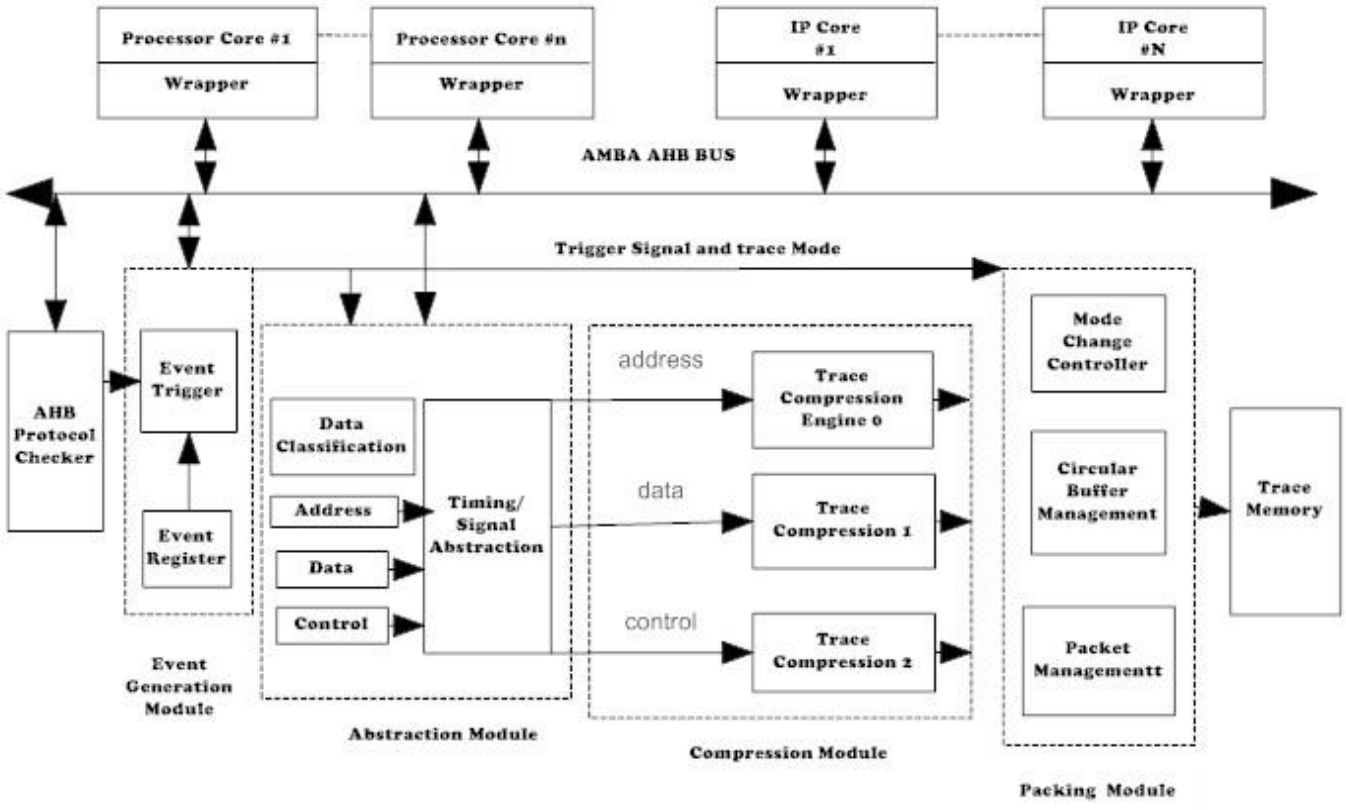

Fig. 1. Multiresolution bus tracer block diagram

We divide the entire trace into several independent small traces. Destroying the initial state of one trace does not affect other traces since every trace has its own initial state. This technique can be accomplished by periodically triggering a new trace. With minor modification to their control circuitry it can be easily accomplished by the existing trace compression engines.

3. BUS TRACER ARCHITECTURE

Fig. 1 is the bus tracer overview. It mainly contains four parts: Event Generation Module, Abstraction Module, Compression Modules, and Packing Module.

3.1 Event Generation Module

The trace and trace mode starting and stopping are decided by event generation module. The triggering events on the bus controlled by event registers. The matching circuit is used to compare bus activities with the events specified in the event registers. We can connect an AHB bus protocol checker (HPChecker) [10] to the Event Generation Module, as shown in Fig.3, to capture the bus protocol related trace.

3.2 Abstraction Module

This Module monitors the AMBA bus and selects/filters signals based on the abstraction mode. Depending on the abstraction mode, some signals are ignored, and some signals are reduced.

3.3 Compression Module

This Module monitors the AMBA bus and selects/filters signals based on the abstraction mode. Depending on the abstraction mode, some signals are ignored, and some signals are reduced.

3.4 Packing Module

This Module receives the compressed data from the compression module. It processes them and writes them to the trace memory. Packet management, circular buffer management, and mode change control are managed by this module.

4. Compression Mechanism

To reduce the size, the compression approaches are necessary. Since the signal characteristics of the address value, the data value, and the control signals are quite different, we propose different compression approaches for them. They are Program address compression, Branch/Target filtering, Dictionary based compression, Slicing, Data address/value compression, Control signal compression. Integrating the bus tracer into a SoC is done by simply tapping the bus tracer to the AHB bus.An on-chip processor or an external debugging host controls the bus tracer. Real-time tracing is achieved when the bus tracer is pipelined to meet the on-chip bus frequency. Since the trace data processing is stream-based, the bus tracer can be easily divided into more pipeline stages to meet aggressive performance requirements.

5. EXPERIMENTAL RESULTS

The bus tracer costs only about 41 K gates, which is relatively small in a typical SoC. The largest component is the FIFO buffer in the packing module. The second one is the compression module. The cost to support both the pre-T and post-T capabilities (periodical triggering module) is only 1024 gates. The major component of the event generation module is the event register, which are roughly 1600 gates per register. The implementation in this paper has two event registers. More registers can be added if necessary. Compared with the previous work [12], the gate count is reduced by 35%. The reason is that this paper optimizes the ping-pong architecture by sharing most of the data instead of duplicating all the hardware components. As for the circuit speed, the bus tracer is capable of running at 500 MHz, which is sufficient for most SoC's. If a faster clock speed is necessary, our bus tracer could be easily partitioned into more pipeline stages due to its streamlined compression/packing processing flow.

6. CONCLUSION

For the process of development, integration, debugging, monitoring and tuning of AHB-based SOC's, an on-chip bus tracer is designed. The bus tracer is attached to the AHB bus which captures and compresses the real-time bus tracer with five modes of resolution. The modes can be switched dynamically while tracing. The bus tracer also supports both directions of traces,trace before the triggering event and trace after the triggering event. Our Bus tracer supports a diverse range of design/debugging/monitoring activities, including module development, chip integration, hardware/software integration and debugging, system behavior monitoring, system performance/power analysis and optimization, etc. The Bus tracer costs only 46 K gates, making it a valuable and economical investment in a typical SoC. It runs at 500 MHz satisfies the requirement of real-time tracing. Experiment results show it achieves high compression ratio ranging from 75% to 95% depending on the trace mode. The bus tracer has been successfully verified in both FPGA and test chip levels. In the future, we would extend this work to more advanced buses/connects such as AXI.

REFERENCES

[1] E. E. Johnson, J. Ha, and M. B. Zaidi, "Lossless trace compression," *IEEE Trans. Comput.*, vol. 50, pp. 158–173, Feb. 2001.

[2] *CoreSight: V1.0 Architecture Specification*, ARM.

[3] R. Leatherman and N. Stollon, "An embedded debugging architecture for SoCs," *IEEE Potentials*, vol. 24, no. 1, pp. 12–16, Feb-Mar 2005.

[4] ARM Ltd., San Jose, CA, "ARM. AMBA AHB Trace Macrocell (HTM) technical reference manual ARM DDI 0328D," 2007.

[5] T. A. Welch, "A technique for high-performance data compression,"*IEEE Trans. Comput.*, pp. 8–19, 1984.

[6] *Embedded Trace Macrocell Architecture Specification*, ARM.

[7] C. MacNamee and D. Heffernan, "Emerging on-chip debugging techniques for real-time embedded systems," *IEE Comput. Contr. Eng. J.*, pp. 295–303, Dec. 2000.

[8] B. Tabara and K. Hashmi, "Transaction-level modeling and debug of SoCs," presented at the IP SoC Conf., France, 2004.

[9] ARM, *AMBA Specification (Rev 2.0) ARM IHI0011A*, May 1999.

[10] C.C.Wang, AHB On-Chip Bus Protocol Checker, 2007

[11] Fu-Ching Yang, Cheng-Lung Chiang and Ing-Jer Huang, "A Reverse-Encoding-Based On-Chip Bus Tracer for Efficient Circular-Buffer Utilization", IEEE Transactions on. Very Large Scale Integration (VLSI) Systems, vol 18, Issue : 5, p732 – 741, May 2010.

[12] Y.-T. Lin, W.-C. Shiue, and I.-J. Huang, "A multi-resolution AHB bus tracer for read-time compression of forward/backward traces in a curcular buffer," in *Proc. Des. Autom. Conf. (DAC)*, Jul. 2008, pp. 862–865.

[13] *AMBA Navigator Spec Sheet*, First Silicon Solutions (FS2) Inc.

BIOGRAPHIES

A.Mallaiah received the M.E degree in Electronic Instrumentation from Andhra University, Visakhapatnam in 2004, B.Tech degree in Electronics and communication Engineering from R.V.R & J.C College of Engineering, Guntur in 2002. He is Associate Professor, Department of Electronics and Communication Engineering at Gudlavalleru Engineering College, Gudlavalleru. He has a total teaching experience (UG and PG) of 7 years. He has guided and co-guided 6 P.G students. His research areas include Embedded Systems, Molecular Electronics, Digital Signal Processing and Digital Image Processing.

G.Harish is pursuing his M.Tech degree in Embedded Systems from Gudlavalleru Engineering College, Gudlavalleru, B.Tech from Electronics & Communication Engineering from St'Anns College of Engineering and Technology, Chirala in 2010.His research interest includes Digital Image processing and Embedded Systems.

M.Kamaraju obtained his Bachelor's Degree & Master's degree from Andhra University. Areas of interest are Microprocessors, Microcontrollers, Digital system Design, Embedded System Design, Low Power VLSI Design. He is an having experience of 19 years in the field of teaching. He published 32 technical papers in national and international journals and conferences. He reviewed number of papers for international journal and international conferences He is a Fellow of IETE and IE. Presently working as Professor & Head of ECE Department, Gudlavalleru Engineering College, Gudlavalleru, India.

T.Subhashini is working as an Asst.Professor in the Department of Electronics & Communication Engineering at Sri Vasavi Institute of Engineering & Technology, Pedana, She has 2 international journal publications to her credit. She has Completed M.Tech degree in Digital Electronics & Communication Systems from Gudlavalleru Engineering College, Gudlavalleru in 2010, B.Tech from Electronics &Communication Engineering from V.R Siddhartha Engineering College , Vijayawada in 2005. Her main research interest includes Digital Signal Processing, Digital Image Processing and Embedded Systems.