Design and Implementation of the Arm JTAG Emulator

Chun Zeng*

Chongqing Technology and Business Institute, Chongqing, China

Abstract: ARM processors are widely used in embedded electronic products, especially in communication, industry control, and automatic, has become the major processor in embedded electronic products. Nearly all the high performance emulators are developed by oversea tool providers, which are too expensive to be generally developed by a small or tiny company. Some JTAG emulator with low price can't meet the requirements of large scale software development. This paper focuses on the problems mentioned above, provide a solution to design and implement the ARM JTAG emulator after studing ARM EmbeddedICE technology, GDB debugging technology, RSP protocol, μC/OS-II and LwIP and *etc*. This JTAG emulator is designed based on AT91SAM9260, supporting ARM7 and ARM9 serial processor core and JTAG clock rate programming, with the communication interface including 10/100M auto-adapt Ethernet, USB and RS232.

Keywords: ARM emulator, Embedded ICE, JTAG emulator.

1. INTRODUCTION

ARM-based SOC proportion accounted for 75% of all 32-bit embedded processor, has become the largest number of global accounts for the use of 32-bit RISC processor [1-4]. ARM processors are used in many electronic products, from portable devices (mobile phones, PDA, PND, multimedia players, handheld electronic games) to computer peripherals (hard drives, desktop routers), even in the onboard computer missiles and other military facilities. In addition, according to the market consulting firm, ABI predicts that by 2013, technology based on ARM processor will dominate the mobile Internet device (MID) market [5].

Therefore, how to achieve rapid, efficient development of ARM-based processing products become a major concern. In the traditional method of embedded software development, it is necessary to run in the target system debug monitor (Monitor), provide the necessary software debugging functions such as setting breakpoints, data upload and download applications from the target memory, *etc.* [3, 4]. This means that before you debug your application, you must ensure that the monitoring program (monitor) will be able to work after a hardware is reset to start. The monitoring program code must also be based on the specific target system migration. In different hardware systems, the ARM processor is used not only because of differences in the same peripherals, but it also has a transplant; it introduces additional workload and increases the product development cycle.

ARM processors supporting these traditional debugging methods, have developed a new technology based on boundary scan debug mode to solve the problem that traditional tools can not solve, which is the main content of this research.

Currently, ARM SOC technology support the JTAG debug, through the JTAG interface of the ARM processor core to control the deepest functionality, using the current best software debugging and testing methods [6-9]. Using JTAG debug mode can effectively shorten the software and hardware development cycles, quicker time to market, and can even help users improve product quality.

2. EMULATOR HARDWARE DESIGN AND İMPLEMENTATİON

This chapter introduces the ARM JTAG emulator hardware design and implementation. The entire hardware system uses AT91SAM9260 processor, external use of a wide range of 9 ~ 35V DC power supply, support for Ethernet, USB, serial port communication interface, provides a standard 20-pin JTAG debug interface, and 8M bytes NorFlash board configuration and 64M bytes SDRAM. To support programmable JTAG access clock speed, there are CPLD logic chips on-board. The chip will microinstruction CPU JTAG boundary scan issued convert serial timing signals. CPLD logic chip uses 8 AT91SAM9260 LocalBUS bus width. Hardware board system block diagram shown in Fig. (**1**).

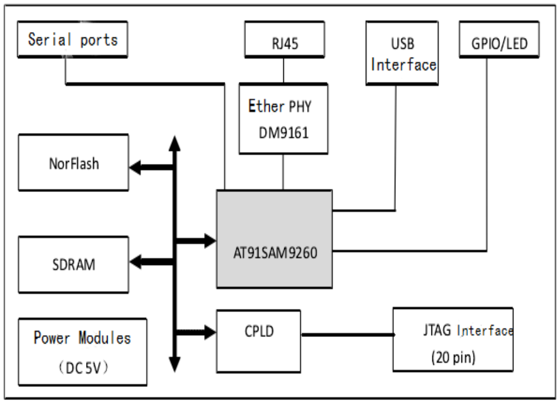

Fig. (1). ARM JTAG emulator hardware system block diagram.

*Address correspondence to this author at the Chongqing Technology and Business Institute, Chongqing, P.R. China; Tel: 18523011233;
E-mail: 76329016@qq.com

1874-4443/15 2015 Bentham Open

2.1. AT91SAM9260 Processor

AT91SAM9260 is released by ATMEL Corporation, based on the ARM926EJ-S processor core SOC chip, including 8K bytes of high-speed data buffer, and 8K bytes of high-speed instruction buffer. It extends the fast ROM, RAM, and a large number of peripherals. 32K bytes of internal ROM, is used to store the power-BOOT boot program. T91SAM9260 uses 6-layer bus matrix structure, effectively raising 6 32 internal bus bandwidth. At the same time, its external bus interface supports a mass storage device [10].

T91SAM9260 integrates Ethernet MAC, integrates a USB 2.0 protocol HOST controller and a USB Device interface, and integrates a variety of standard peripherals, such as UART, SPI, TWI, Timer, counter, ADC, SD / MMC card interface.

2.2. Hardware System Design Based on AT91SAM9260

Emulator hardware board uses ATMEL's AT91SAM9260 as the central processor. For easy welding and commissioning, use PQFP208F package design clocked 180MHz.

(1) MCU Core Module Design

According to the assessment of software size, RAM / ROM of hardware board core part design is as follows:

8M Bytes NorFlash；

64M Bytes SDRAM (Two 16-bit SDRAM chips sub 32-bit interface)

(2) Design of Communication Module

External interface part, according to the needs of communication, commissioning and other aspects of the design of the following interfaces:

Serial port: A DEBU serial port, a common serial port (standard RS232 interface)

Ethernet ports: a 100M Ethernet ports (AT91SAM9260 internal MAC + external PHY), with the sending, receiving and coupling indicator.

USB Interface: a USB HOST (USB 2.0 Full Speed) interface and a USB Device (USB 2.0 Full Speed) interface; USB interface as a follow-up development reserve used temporarily as an ARM emulator communication interface.

(3) JTAG interface

The design of JTAG interface is standard 20Pin JTAG interface.

(4) Status indication LED

3.3V power status indication LED

ARM emulator work status indication LED

JTAG data download status indication LED

(5) Power Interface

Switching power supply, input DC voltage range is + 9 ~ 35V.

2.3. The Design of JTAG Interface

ARM JTAG interface has two forms, namely, 14pin and 20pin. 20pin form is the current mainstream. This paper supports the standard 20Pin JTAG interface, which has anti-misplugging design. Form and signal interface definition are shown in Fig. (**2**).

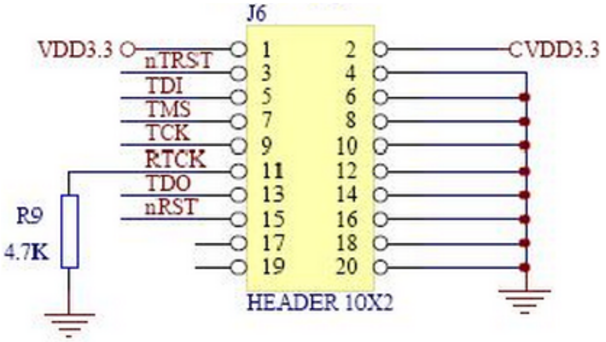

Fig. (2). 20 Pin JTAG signal ınterface.

For detailed description of each pin signal, see Table **1**. It should be noted, that signal direction defined in the table, is based on the perspective of defined ARM JTAG emulator.

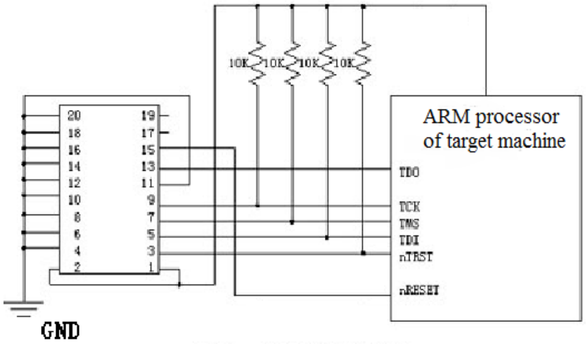

Fig. (3). JTAG ınterface hardware design of target machine.

ARM processor-based hardware board design, which is being debugged as target hardware board. Its JTAG hardware interface circuit is shown in Figs. (**3**, **4**). Interface format and signal definition correspond to definitions of ARM JTAG debugging. JTAG interface target recommended 10K ohm pull-up resistor.

To meet the target ARM chips at different operating voltages, ARM JTAG emulator supports JTAG interface target voltage adaptation, that can adapt to 1.2V, 1.8V, 2.5V, 3.3V and 5V multiple target processor JTAG interface voltages.

To maintainn adaptive voltage between the JTAG debugger interface and CPLD logic chips, was open-drain output voltage converter chip TXS0104 used. The chip supports four bi-level conversions and has the following characteristics:

External circuit drive capability can drive higher than the chip supply voltage of the load. Open drain outputs allow connection to a signal line. A pull_up resistor, without increasing the device's case, generates " and logical."

Because TXS0104 an open-drain chip, it thus requires post-stage circuit, also target JTAG interface circuit board

Table 1. JTAG interface signal descriptions.

Order	Signal name	Direction	Explanation
1	Very	Input	Interface level reference voltage, usually directly connected to the power supply
2	Vsupply	Input	Power
3	Ntrst	Output	(Optional) ARM processor TAP logic reset signal. To prevent the error reset, usually in the ARM processor side plus appropriate pull-up resistor.
4	GND	--	Earthing
5	TDI	Output	Test data input (Test-Data-In), ARM JTAG emulator output this signal as ARM processor TAP controller TDI input signal.
6	GND	--	Earthing
7	TMS	Output	Test Mode Select (Test-Mode-Select), ARM JTAG emulator output this signal, the TAP controller as the ARM processor TMS input signal.
8	GND	--	Earthing
9	TCK	Output	TAP controller clock signal (Test-Clock), ARM JTAG emulator output this signal as ARM processor TAP controller TCK input signal.
10	GND	--	Earthing
11	RTCK	Input	TCK clock feedback acknowledgment signal (Return-Test-Clock). Feedback from JTAG emulator for ARM processor clock signal to synchronize generate TCK signal. Can be connected directly to ground when not in use. This signal is optional, generally used in high-speed JTAG interface.
12	GND	--	Earthing
13	TDO	Input	Test input and output signals (Test-Data-Out), output from the JTAG emulator for ARM processors, ARM processors connected TDO TAP controller output signal.
14	GND	--	Earthing
15	nSTST	Input/Output	ARM processor system reset signal (System Reset), connected to the system reset signal on the target board. JTAG emulator directly on the target system reset by this signal can be detected at the same time to reset the case of the target system. In order to avoid false triggering a reset, the general increase in the ARM processor side proper pull-up resistor. This signal is an optional signal.
16	GND	--	Earthing
17	NC	--	Retention
18	GND	--	Earthing
19	NC	--	Retention
20	GND	--	Earthing

must have a pullup resistor. Pullup resistor supply voltage determines lose TXS0104 out level, in order to achieve the target JTAG adaptive interface level. JTAG interface circuit design schematic section shown in Fig. (**4**).

2.4. Ethernet Interface Design

AT91SAM9260 chip built a 10 / 100M adaptive MAC, only to realize the external Ethernet PHY Interface. On the PHY chip selection, select DAVICOM released DM9161A 10 / 100Mbps Fast Ethernet physical layer, single-chip transceiver. This chip has a single-chip, low-power, 10 / 100M adaptive, support UTP adaptive (AUTO-mix), and supports TCP / IP hardware acceleration features. DM9161A chip functions logic figure is shown in Fig. (**5**).

Through the MII interface DM9161A, AT91SAM9260 DM9161A connects with MAC and the PHY. Ethernet interface design use the integrated isolation transformer RJ45 interface, HR911105A. The interface device meets the IEEE802.3 and 802.3ab standards, supporting a minimum isolation voltage of 1500Vrms.

2.5. Other Interface Design

ARM JTAG emulator also supports RS232 serial and USB interfaces. Due to relatively simple RS232 interface, USB interface design in hardware, due to limited paper space.

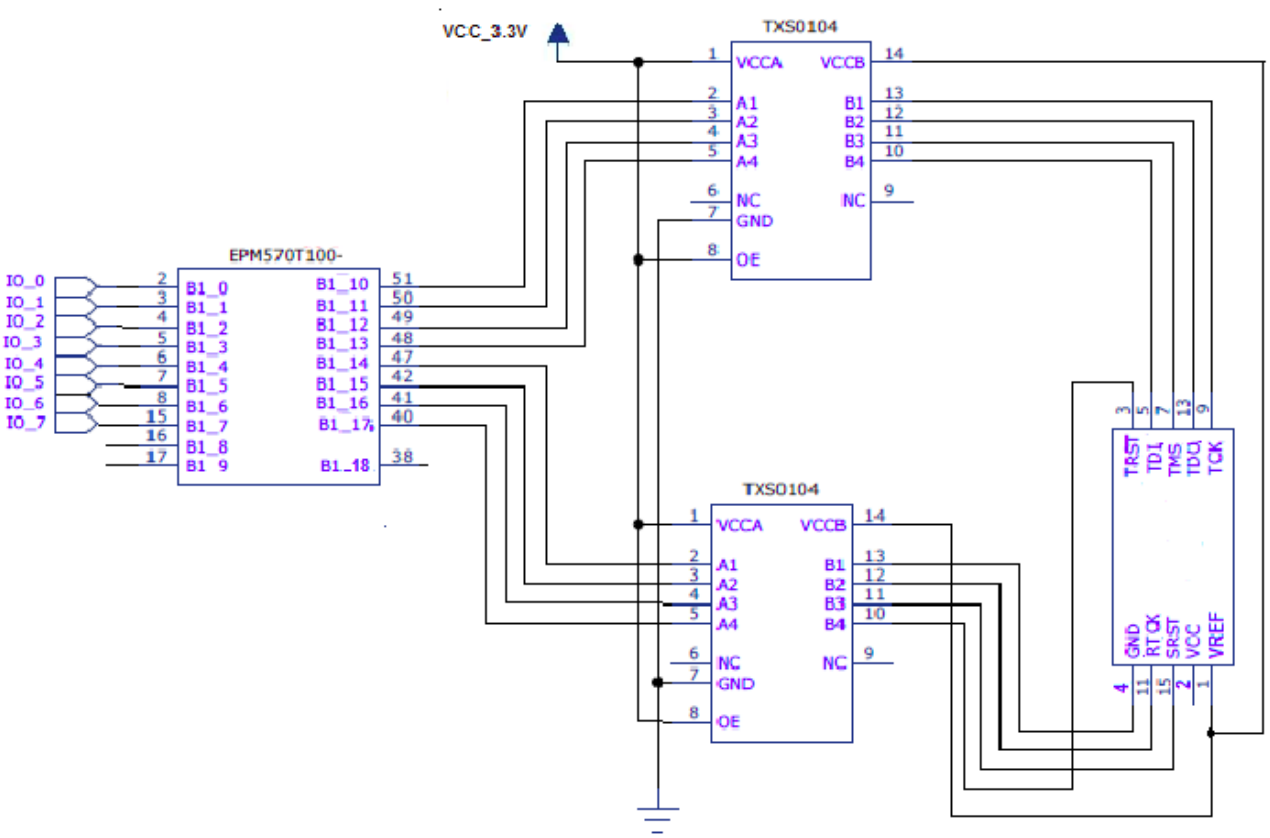

Fig. (4). Adaptive voltage JTAG ınterface design.

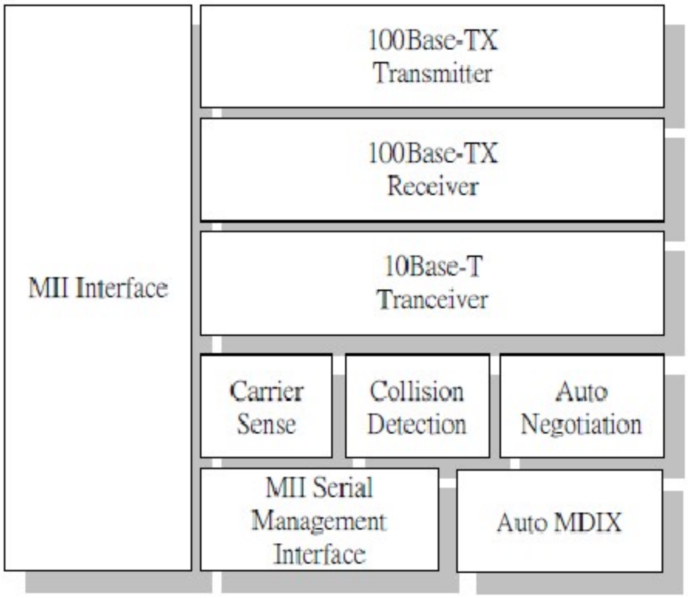

Fig. (5). DM9161A chip functions logic figure.

Table 2. JTAG emulator key test items.

Classification	Test item	Test content	Testing expected Result	Test Result
The basic functional testing	Register to read and write functional tests	For general-purpose registers to read and write functional tests, to see the value of the read and write values are the same.	For R0-R15, CPSR do reading and writing test; Written value is consistent with the read value.	Passed
	Byte memory read and write functional tests	Value in bytes of memory on the target machine to read, write functional tests to see whether the value written and read are the same.	Written value is consistent with the read value.	Passed
	Halfword memory access function test	Halfword value of the target machine memory read, write functional tests to see whether the value written and read are the same.	Written value is consistent with the read value.	Passed
	Word mode memory read and write functional tests	Value of the target word the way memory read, write functional tests to see whether the value written and read are the same.	Written value is consistent with the read value.	Passed
	Software breakpoints set, clear test	Select the software breakpoint mode, set a breakpoint on the program being debugged, clear breakpoints, and other related tests.	Order execution breakpoint is encountered suspended; clear the breakpoint after the program execution is not aborted; unlimited number of breakpoints.	Passed
	Hardware breakpoints set, clear test	Select the hardware breakpoint mode, set a breakpoint on the program being debugged, clear breakpoints related tests.	Suspend program execution breakpoint is encountered; clear the breakpoint after the program execution is not aborted; simultaneously supports up to two hardware breakpoints.	Passed
	Data observation point set, clear test (sub-byte, halfword, word width were tested)	Select the observation point mode, the program being debugged set of data rows outlook Observation point, clear the data related to the observation point test.	When the raw data access addresses corresponding trigger observation point, Cheng Suspend the enforcement order; after clear breakpoints do not abort the program; At the same time the data can support up to two observation points.	Passed
	Source-level debugging features single-step test	Implementation of source-level debugging features single-step test.	You can perform source-level single-step debugging, including source-level single-step for, while, etc. statement.	Passed
	Instruction-level single-step debugging function test	Instruction-level single-step debugging function tests.	You can execute the instruction level single-step debugging.	Passed
	CP coprocessor registers write functional tests	CP15 coprocessor test read and write access.	Written value is consistent with the read value.	
	JTAG emulator software upgrade testing	JTAG emulator software to support online upgrade.	After upgrading to normal operation; false interrupt the upgrade process (e.g. Network interruption) does not lead to a fatal exception; can prevent accidental upgrades.	

(Table 2) contd….

Classification	Test item	Test content	Testing expected Result	Test Result
Key Performance Testing	Download speed test	Program download speed test.	Download speed is not less than 1Mbyte.	Passed
Stability testing	A large network environment, job stability test	The host and JTAG emulators are hung in a large network environment, full functional testing.	You can work, download speed, no significant difference in response time debugging.	Passed
	Large-scale stability testing program download	Scale program download function tests to determine the stability of download function.	The download process is not interrupted; The downloaded data is correct;	Passed
	Stability debug function	Program debugging process, all debugging features integrated use (see the revised register, see the modified memory, review the stack, disassemble, *etc*.), to confirm the stability of debugging features.	Debugging is consistent with expectations, without exception, stable jobs.	Passed
Adaptive testing	Different target JTAG interface voltage stability test	Adaptively target JTAG interface voltage, can work.	Can work at different voltages (1.2V, 1.5V, 1.8, V2.5, V3.3, V5V).	Passed
	ARM processor target automatic recognition test	Target machine can automatically identify the type of processing (ARM7TDMI, ARM720T, ARM9TDMI, ARM920T, *etc*.)	Correct target processor type.	Passed
Environmental Compatibility Test	Debugging different host operating system compatibility testing	Debugger work under different operating host of, JTAG emulator to work properly.	In WindowsXP, Windows2003, Windows7, Linux operating system, were tested, without exception.	Passed

3. SIMULATOR TEST

TAG Emulator is part of an embedded cross-development tool chain, the embedded ARM has a prominent role in the product development process. So for the JTAG emulator, performance and reliability are demanding features. JTAG emulator testing is divided into sub-unit testing and system testing categories. Unit tests aim at key algorithms and control modules, that is before making the system commissioning and testing ensure that the key algorithms and control logic is consistent with the design. System testing is a key work to carry out simulator tests, and most emulators ensure the availability and effectiveness of this most important means. The system testing products are mainly classified according to the following tests carried out in major items:

The basic functional testing;

Key performance testing;

Stability testing;

Environmental compatibility testing;

Adaptive testing of debugging target processor.

CONCLUTION

Table **2** provdes items and test methods with more detailed description of the test. Some problems are also found, such as job insecurity in a large network, the configuration file passing error, ARM9 Cache abnormalities and other problems make debugging process time taking. After careful analysis of these issues and positioning, ultimately problems have to be resolved. In return version of the test, all of the test items are in line with expectations, performing all the basic functions and key performance with JTAG emulator to achieve a full, stable job, to meet the design expectations.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] IEEE. IEEE1149.1-2001 IEEE Standard Test Access Port and Boundary-scan Architecture, pp. 3-32, 2001.

[2] ARM Ltd. The ARM7TDMI Debug Architecture, 2004.
[3] ARM Ltd. ARM7TDMI Technical Reference Manual, 2004.
[4] ARM Architecture Reference Manual.pdf, ARM Ltd
[5] Debugging with gdb, http://sources.redhat.com
[6] R York, "Real Time Debug for System-on-Chip Devices", smartdata.usbid.com
[7] H. Hogl, D. Rath, and F. Augsburg, "Open On-Chip Debugger-OpenOCD", 2006.
[8] ARM Ltd, Application Note 28: The ARM7TDMI Debug Architecture ,www.arm.com
[9] J. Wang, X. Gong, and Q. Su, "Application of JTAG Debugging Technology and ARM Emulator," *Electronic Measurement Technology,* vol. 7, no. 4, pp. 24-25, 2004.
[10] Z. Yu, and J. Wei, "Embedded Emulator Design for ARM System Based on USB Interface," *Chinese Journal of Electron Devices,* vol. 15, no. 4, pp. 1293-1295,1299, 2008.

Received: September 16, 2014 Revised: December 23, 2014 Accepted: December 31, 2014