Application of the programmable logic devices for implementation of in-circuit tester for microprocessor system

Boyko Baev Petrov

Abstract - **The article examines the use of a CPLD for the realization of in-circuit tester of microprocessor system. The behavior is designed to be analogous to the behavior of the target MPU in write and read mode to and from an external address via parallel MPU bus. By setting the appropriate address and data from a PC based application via USB-connected external microprocessor are performed various test procedures on system and peripheral resources of the target system.**

Keywords – **CPLD, Verilog, Emulator, Tester, In-circuit**

I. Introduction

Testing of the microprocessor systems is up to date technical problem early in the development of microprocessors. Given the complex nature of the deficiencies that may arise in microprocessor systems have been developed various methods and means to perform test procedures designed to detect defective electronic component. In the 1980s, the Joint Test Action Group (JTAG) developed a specification for boundary-scan testing that was later standardized as the IEEE Std. 1149.1 specification [1]. After 1992, virtually all of microprocessors have the functional capacity to perform the test procedures resulting from embedded JTAG or some similar in functionality a method of internal arrangements test. In the industry are still microprocessor systems based on microprocessors, which do not possess any built-in-circuit test method. In these cases, it is convenient to use a special unit - the emulator connected in the microprocessors socket. The operation of this emulator is to generate data, address and control buses in time for reading and writing, which is generated by microprocessor superseded. Thus, all the resources available through this system bus are available from the emulator. The implementation of a variant of such unit, realized by CPLD, microcontroller and a PC based program is the subject of this paper.

B.Petrov - Department of Electronics, Technical University Sofia, Branch Plovdiv. E-mail: bpetrov@tu-plovdiv.bg

II. Tester Architecture

The Figure 1 describes architecture of the present tester unit. As can be seen, it is MPU based unit, connected to the PC over USB compatible connection and connected to the target PCB via ribbon cable directly to the socket of the removed MPU Intel 80C186 [4][6].

The main functions of the in-circuit tester are separated between three main units - CPLD, embedded MCU and PC-based software.

The CPLD is directly connected to the target system. Figure 2 describes a method of connection between two pins of the CPLD and single pin of target system. The current limit resistor protects the CPLD output pin from short connection to the power or ground planes or logical output in damage condition, that can be happened in particular system-over-test. The input pin, connected to the same output signal is used to recognize these types of cases and via tester interface can be seen on the PC display. Based on described connection, the three types of digital ports are used - output-only digital port, input-output digital port and input-only port. These three types of ports are semantically connected to the corresponding signal lines that are under 80C186 MPU control normally. In CPLD two types of command interfaces with embedded MPU is used - read and write to ports made over embedded MCU software and automatic read and write bus cycles according 80C186 timing requirements [4][6] made by embedded in CPLD finite state machine (FSM). The using of embedded FSM significantly increase performance to the memory and I/O spaces of the target system. The start of read and write cycles are under embedded MCU control.

The embedded MCU (ATmega162) is connected to the PC and CPLD. The connection to the PC is over USB interface using well-known FT232 [8]. The connection to CPLD is over parallel data and address bussed of the ATmega162. In this type of architecture, every port organized in CPLD is writable and readable without any delays. The main function of embedded MCU is waiting of incoming PC-commands via USB bus. The set of these based commands is designed for using only read and writes of data to the ports of the CPLD.

The PC-software is main part of the tester. It integrates all features of the CPLD and embedded MCU software. The main idea of PC-software design is to make a useful tool that can help to find a defective component in target system. The main functions of the PC-software are:

- Emulation of the every output pin behavior of the target MPU to stay in high or low logical state,
- Read the status of the every input pin of the target MPU,
- Realization of single-mode writing of the 16-bit data to the particular address and auto-modification mode of increment or decrement of writing memory address
- Realization of single-mode reading of the 16-bit data from the particular address and auto-modification mode of increment or decrement of reading memory address,
- Functional test of the system RAM,
- Functional test and verification of content of the system FLASH includes - erase, program, read and verify,
- Iinitialization of the user mode of the peripheral timer,
- Initialization of the user mode of the peripheral asynchronous adapter.

All of listed modes of test can be started in random order according the desire of the qualified operator.

III. Tester Realization

The implementation of the tester is based on the following key components: CPLD - XC95288, MCU-ATmega162 and FT232.

The following features are essential for the choice of them:

XC95288 CPLD [2] is powered from 3.3V device and has 5V compatible inputs. Mounted in 208-pins package allowing a duplication for current limitation technique shown in Fig.2. of each terminal used for MPU emulation. The released of the outputs and reading from the inputs are done by MCU embedded software via program accessible registers. The FSM for read and write cycles implementation of external MPU parallel bus is build-on in CPLD. The timing diagram on Fig.3 shows a simulation results after test-bench Verilog code of FSM is executed.

ATmega162 is a MCU [3] used for middle level realization of tester control path. The internal FLASH memory is used for firmware executing. The quick access to the CPLD status and control registers was realized over parallel data and address busses. ATmega162 has a serial interface UART that allows connection to a higher hierarchical level via UART-to-USB converter FT232 [8]. Table 1 describes a brief of the command list that MCU can execute.

TABL.1. BRIEF OF PC-SOFTWARE-TO-MCU COMMANDS

Command Code	**Command description**
0x00 <CPLD_address>	internal read from CPLD status register
0x10 <CPLD_address> <CPLD_data>	internal write to CPLD control register
0x20<Ext_mem_address>	external memory read command
0x30<EXT_mem_address> <EXT_data>	external memory write command
0x40<EXT_page_address>	external memory read page
0x50<EXT_page_address> <PAGE>	external memory write page
0x60	upload page
0x70	download page
0x80	page to flash program
0x90<EXT_mem_address>	RAM page test
0xA0<EXT_IO_address>	external I/O read
0xB0<EXT_IO_address> <IO_data>	external I/O write
0xC0<FLASH_page_address>	flash page erase
0xD0<FLASH_page_address>	set flash page address
0xE0<active_FLASH_code>	set active FLASH memory chip
0xF0	reset tester

The PC-software is Windows XP based application. The main functions are organized in six tabs. The main features are:

- Find a Tester via existing USB serial virtual communication ports;
- Execution of low level test of capability of every output pin to set in high and low logical level. The views of low-level tests are shown on Figures 4,5,6,7.
- Execution of high-level functional test of memory bus includes make timing with auto increments or auto decrements capabilities with user data bus contends, shown on Figure 8,
- Execution of high-level functional test of RAM, shown on Figure 8,
- Execution of maintenances, shown on Figure 8, of two FLASH devices, that includes the next services - erase, write FLASH page, read FLASH page, write FLASH device with content of particular data file,
- Execution of high-level functional test of I/O bus includes free read and write to I/O device, shown on Figure 9,
- Initialization of timer UPD71054 [5] with user constants, shown on Figure 9,
- Initialization, read and writes of Multi-protocol serial controller UART LD8174 [7], shown on Figure 9.

The Fig.10 shows the view of the Tester to Target system connection.

FIG.1. BLOCK SCHEME OF THE CPLD BASED IN-CIRCUIT MPU TESTER

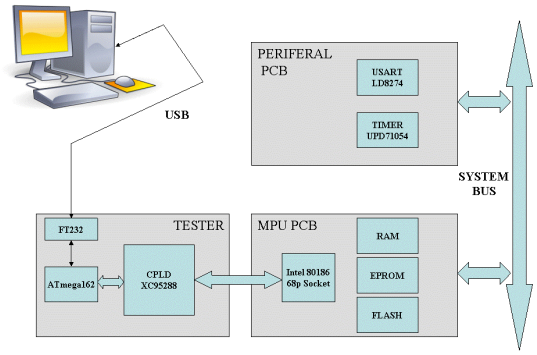


FIG.2. ELECTRICAL SCHEME OF THE CPLD TO TARGET SYSTEM CONNECTION OF THE SINGLE PIN.

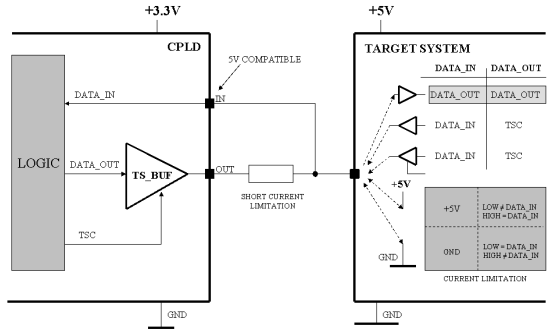


FIG.3. TIMING OF READ AND WRITE BUS CYCLES SIMULATION IN AUTO MODE.

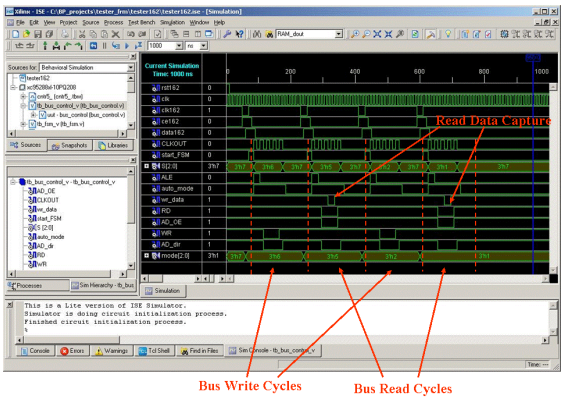


FIG.4. THE VIEW OF SOFTWARE MENU FOR LOW LEVEL TESTS OF TWO DIRECTIONAL 16-BIT MULTIPLEXED AD BUS OF THE INTEL 80C186 MCU.

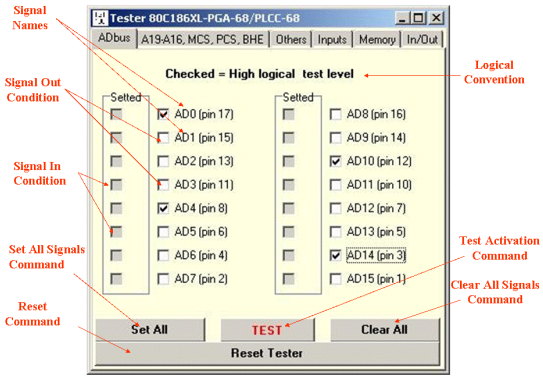


FIG.5. THE VIEW OF SOFTWARE MENU FOR LOW LEVEL TEST OF SINGLE DIRECTION HIGH-ADDRESS BUS AND OTHER CONTROL SIGNALS OF THE INTEL 80C186 MCU.

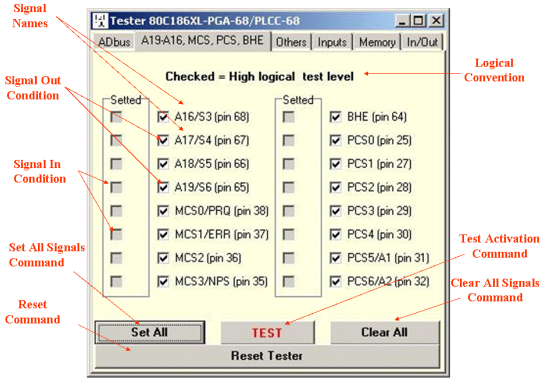


FIG.6. THE VIEW OF SOFTWARE MENU FOR LOW LEVEL TEST OF SINGLE DIRECTION CONTROL SIGNALS OF THE INTEL 80C186 MCU.

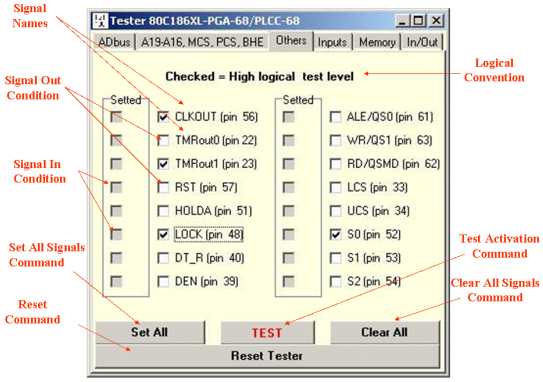


FIG.7. THE VIEW OF SOFTWARE MENU FOR LOW LEVEL READ OF SINGLE DIRECTION INPUT SIGNALS OF THE INTEL 80C186 MCU.

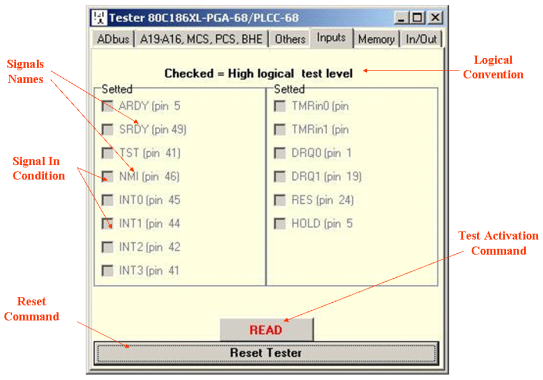


FIG.8. THE VIEW OF SOFTWARE MENU FOR HIGH LEVEL TESTS OF THE MEMORY SPACE, RAM AND FLASH MEMORY OF THE INTEL 80C186 MCU.

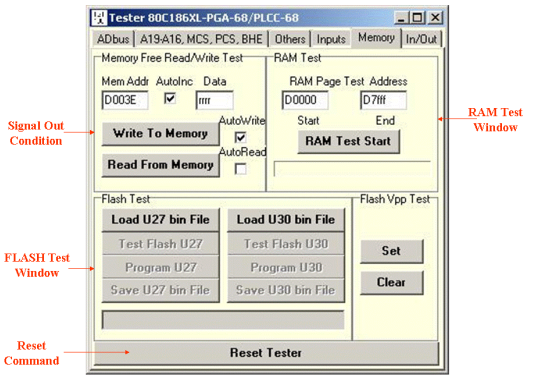


FIG.9. THE VIEW OF SOFTWARE MENU FOR HIGH LEVEL TESTS OF THE I/O SPACE, TIMER, SERIAL INTERFACE AND WDT SYSTEM.

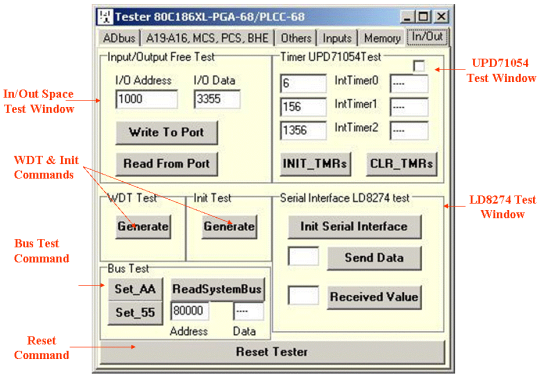
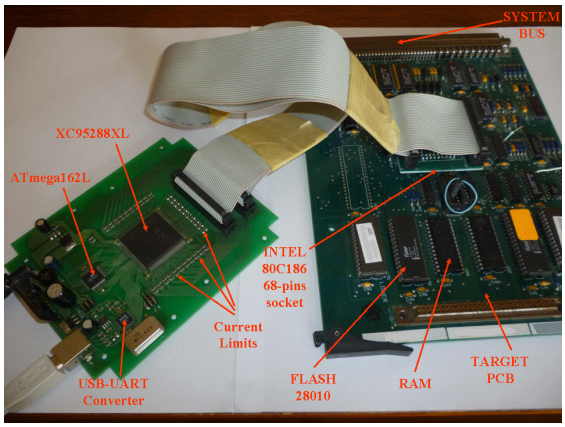


FIG.10. THE VIEW OF THE TESTER-TO-TARGET-SYSTEM CONNECTION.



IV. CONCLUSION

The present paper describes the using of CPLD in MPU-bus emulation application. According with the presented results, it can be successfully used for low and middle level of bus emulation when a particular timing of emulation unit is required. A high level of test procedures can also be implemented in PC-software. The proposed method is useful only for fix of defects in some devices and cannot be used for debugging and other in-circuit emulation of MCU. For these types of emulation, a FPGA based system with embedded MPU core will be more suitable.

REFERENCES

[1]http://www.jtag.org/education/JTAG_Tutorial.htm
[2]http://www.xilinx.com/support/documentation/xc9500xl.htm
[3]http://www.atmel.com/devices/ATMEGA162.aspx
[4]http://download.intel.com/design/intarch/specupdt/27289401.pdf
[5]http://www.ambex.com.pl/archiwum/82C54_82C55/NEC_71054.pdf
[6]http://www.bitsavers.org/pdf/intel/_dataBooks/1991_80C186EA_80C188EA_Users_Manual.pdf
[7]http://www.datasheetarchive.com/intel%208274-datasheet.html
[8]http://www.ftdichip.com/Products/ICs/FT232B.html