8051 IP Core for FPGA Applications

Rastislav Struharik, *Graduate Member, IEEE*, Ivan Mezei, *Graduate Member, IEEE*

***Abstract* — This paper presents a design of soft-core for industry standard 8051 microcontroller intended for FPGA implementation. It can be used in a wide range of applications such as embedded microcontroller systems, data computation and transfer, communication systems and professional audio and video. This core will be used as a part of video transmission and data acquisition system for water, oil and gas exploration. Core was written using industry standard VHDL language and has an advanced micro-architecture that enables it to be 4.51 times faster than the original 8051 microcontroller while operating at the same clock frequency. Paper presents detailed description of core architecture and interface to the surrounding peripheral devices as well as description of the verification plan that was used to verify the correct operation of the core.**

***Keywords* — 8051 microcontroller, IP core, complex digital systems, VHDL, FPGA, functional verification.**

I. Introduction

THE Intel 8051 is a Harvard architecture, single chip microcontroller (μC) which was developed by Intel in 1980 for use in embedded systems [1]. The 8051 was one of the first microcontroller families, and remains one of the most commonly used. The devices are available from multiple sources, are cheap, have decent tools, and offer a nice upgrade path to larger and more capable parts.

Although the 8051 family was created by Intel, it is now largely driven by other companies, including Atmel, Dallas Semiconductor, and Philips.

Many different C compilers are available for the 8051 microcontrollers, some of them are open-source [2]. This fact together with a long and successful operating history makes the 8051 a popular choice for controller when designing new digital systems even today.

A field-programmable gate array (FPGA) is a semiconductor device containing programmable logic components called "logic blocks", and programmable interconnects. Logic blocks can be programmed to perform the function of basic logic gates such as AND, and XOR, or more complex combinational functions such as decoders or mathematical functions. In most FPGAs, the logic blocks also include memory elements, which may be simple flip-flops or more complete blocks of memory.

FPGAs are usually slower than their application-specific integrated circuit (ASIC) counterparts, cannot handle as complex a design, and draw more power (for any given semiconductor process). But their advantages include a shorter time to market, ability to re-program in the field to fix bugs, and lower non-recurring engineering costs.

Nevertheless, FPGA devices have made significant progress in the recent years, and today devices that can handle designs whose complexity measures several millions of gates are readily available on the market. This fact has made FPGAs a plausible candidate for the implementation of System-On-Chip (SOC) designs consisting from large number of different IP cores.

An IP (intellectual property) core is a block of logic or data that is used in making a FPGA or ASIC for a product. As essential elements of design reuse, IP cores are part of the growing electronic design automation (EDA) industry trend towards repeated use of previously designed components. IP cores fall into one of three categories: hard cores, firm cores, or soft cores. Hard cores are physical manifestations of the IP design. These are best for plug-and-play applications, and are less portable and flexible than the other two types of cores. Like the hard cores, firm (sometimes called semi-hard) cores also carry placement data but are configurable to various applications. The most flexible of the three, soft cores exist either as a netlist (a list of the logic gates and associated interconnections making up an integrated circuit) or hardware description language (HDL) source code.

Developed core will be used in a video transmission and data acquisition system for water, oil and gas exploration. During water, oil and gas exploration, borehole measurement systems are used to collect various data. Measuring methods include use of various sophisticated probes that are descended into boreholes up to 5 kilometers of depth [3]. While probe is descending into borehole, various parameters are being monitored in order to avoid a damage of expensive equipment. These parameters include probe speed, depth, cable tension and probe tip differential tension. Due to harsh operating conditions inside boreholes in which this equipment must operate, possibilities of hazards are increased. In order to prevent hazards additional procedures and equipment are being introduced. One possible solution to this problem is introduction of the system for the visualization of probe's environment.

This research was partially financed by the Ministry of Science and Technological Development within the project "Development of Systems and Instruments for water, oil and gas exploration", Pr. Num. 11006, 2008-2009.

Rastislav Struharik is with the Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovica 6, 21000 Novi Sad, Serbia (e-mail: rasti@eunet.yu).

Ivan Mezei, is with the Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovica 6, 21000 Novi Sad, Serbia (e-mail: imezei@uns.ns.ac.yu).

II. General Description of the Core

A. General Description

ATT8051M256 is a soft core of a single-chip 8-bit embedded microcontroller dedicated for operation with fast (on-chip) and slow (off-chip) memories.

ATT8051M256 soft core is 100% binary compatible with the industry standard 8051 microcontroller. It executes all ASM51 instructions and has the same instruction set as the 8031. The ATT8051M256 serves both software and hardware interrupts, and has standard peripheral units like timers and serial communication system.

ATT8051M256 has an advanced micro-architecture that enables it to be 4.51 times faster than the original 8051 microcontroller. It features 8 times faster multiplication and division operations then original 8051 microcontroller, and 4 times faster addition. Core comes with 256 bytes of internal (on-chip) data memory and can address up to 64 KB of internal (on-chip) or external (off-chip) program memory as well as 64 KB of internal (on-chip) or external (off-chip) data memory. To allow easy connection to external memory and other peripheral units, ATT8051M256 core offers de-multiplexed address/data bus, which is not the case with original 8051 microcontroller.

ATT8051M256 is delivered with fully automated testbench and a compete set of tests allowing easy package validation at each stage of SoC design flow.

ATT8051M256 is a microcode-free design developed for reuse in FPGA implementations. The design is strictly synchronous with positive-edge clocking, no internal tri-states and a synchronous reset.

Besides the CPU unit, cores contains interrupt controller with 2 external and 3 internal sources. Interrupt requests can be divided into two priority groups.

Additionally, core contains two programmable 16-bit timer/counters modules. Each module can work as externally gated event counter, or timer clocked by internal source. When operating in timer mode, user can choose one of four possible timer configurations.

Core also contains full-duplex serial port which can operate in one of the four possible modes.

To allow connection with external peripheral devices, ATT8051M256 core has four 8-bit I/O ports. In contrast with original 8051 microcontroller, ATT8051M256 core uses separate input and output lines for each I/O ports. Also, alternate port functions such as interrupts and serial interface are separated, providing extra port pins in comparison with original 8051 microcontroller implementation.

Core comes with the following deliverables: VHDL source code [4], VHDL test bench environment, tests with reference responses, technical documentation, installation notes, instantiation templates and example application.

B. Interface signals

Symbol of the ATT8051M256 and interface signals are presented on Figure 1.

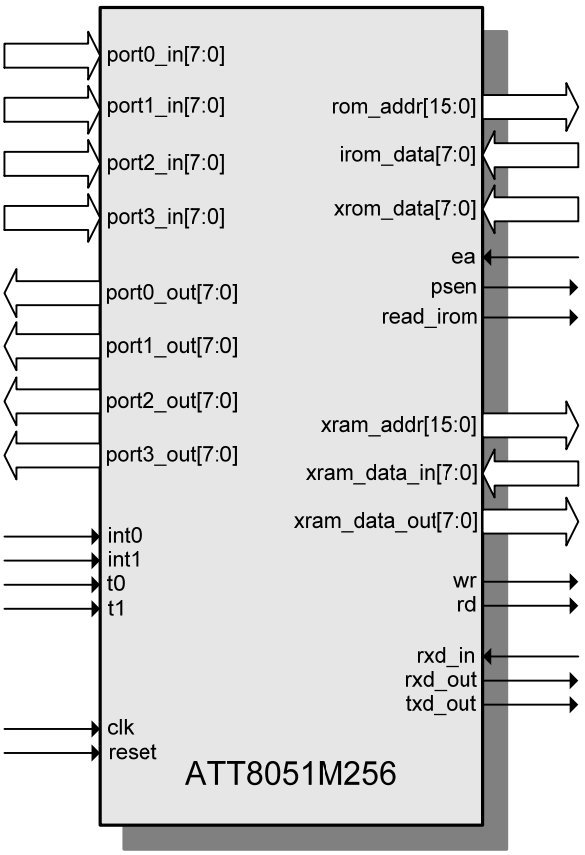

Fig. 1. Symbol and interface signals of ATT8051M256 core.

From Figure 1 we can see that there are in total four 8-bit input (*port1_in*, *port2_in*, *port3_in*, *port4_in*), and for 8-bit output (*port1_out*, *port2_out*, *port3_out*, *port4_ out*) ports.

Interface with internal and external program memory is accomplished using *rom_addr*[15:0], *irom_data*[7:0], *xrom_data*[7:0], *psen*, *irom_read* and *ea* ports.

External data memory interface consists from *xram_addr*[15:0],*xram_data_in*[7:0], *xram_data_out*[7:0], *wr* and *rd* signals.

Besides these signals, there are additional signals for external interrupt requests (*int0*, *int1*), external timer inputs (*t0*, *t1*) and serial port interface (*rxd_in*, *rxd_out*, *txd_out*).

Finally, *clk* and *reset* ports are used to connect global synchronization signal and reset signal respectively.

III. Functional Description and Verification Approach

A. Functional Description

ATT8051M256 core is designed using hierarchical approach. Block diagram showing the structure of the ATT8051M256 core is shown on Figure 2.

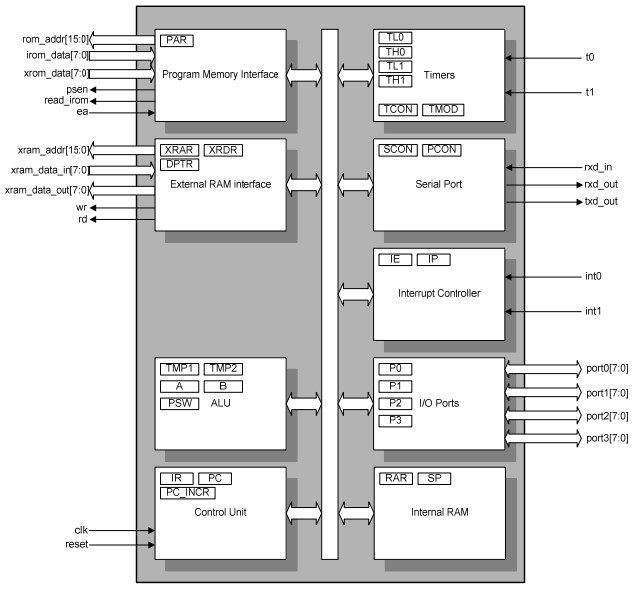

Fig. 2. Top-level block diagram of ATT8051M256 core

Program memory interface contains logic necessary for efficient interfacing to program memories. ATT8051M256 has two program memory spaces, small internal and large external memory. Size of internal program memory is set to 4K bytes, when core is delivered, but this can be modified by user's request to any value up to the 64K bytes. Typically, internal memory is implemented as on-chip memory, and external memory is implemented as off-chip memory. It is important to emphasize that this does not have to be so. Both memories can be implemented as on-chip or as off-chip memories, depending on user preferences.

External RAM interface contains logic needed to implement access to external RAM memory. Typically this memory is implemented as off-chip, but there is no problem if user wants to implement it as on-chip module.

ATT8051M256 contains 256 bytes of internal RAM and control logic that enables easy access. Upper 128 bytes can be accessed using indirect addressing only. Part of this memory is also bit-addressable. This is the same RAM that can be found in original 8051 microcontroller.

ALU unit implements all of the 8-bit arithmetic and logic functions that can be found in 8051 microcontroller. These are: addition, subtraction, multiplication, division, logical AND, OR and XOR operations. ALU also contains a complete Boolean processor that operates with single bit operands.

Control unit performs instruction fetch, decoding and execution. Also it controls the operation of entire system.

There are two almost identical timers, Timer 0 and Timer 1. Both of them have four modes of operation:

- 13-bit Timer/Counter
- 16-bit Timer/Counter
- 8-bit Timer/Counter with auto-reload
- two 8-bit timers

The last mode is available for Timer 0 only. Each timer can serve as pulse counter (transition from 1 to 0) on corresponding inputs t0 and t1. Each of the timers generates one interrupt request when "roll-off" condition occurs.

ATT8051M256 provides means to easy implement serial communication using the serial port. Serial port is capable of both synchronous and asynchronous operation. In synchronous mode, microcontroller generates the clock signal and operates in half-duplex mode. In asynchronous mode, full duplex operation is available. Received data is buffered in a holding register, enabling serial port to receive additional data before software has read the previous value.

Serial port provides four modes of operation:

- Synchronous mode, fixed baud rate
- 8-bit asynchronous mode, variable baud rate
- 9-bit asynchronous mode, fixed baud rate
- 9-bit asynchronous mode, variable baud rate

Serial port also generates two interrupt requests, one to indicate that transmission was completed, and other to indicate that reception was completed.

Interrupt controller has five interrupt sources, two of them external. There are two priority levels that can be assigned to each of the interrupts. Also each source has an independent priority bit, flag bit, interrupt vector and enable bit. There is global enable bit that enables or disables all of the interrupts.

ATT8051M256 provides user with four 8-bit input/output ports, Port0-Port3. Each port is both bit-addressable or can be addressed as a byte.

B. Verification Plan

Since the ATT8051M256 core is a complex digital system, verifying its correct behavior has been a great challenge. To perform comprehensive and efficient functional verification of the ATT8051M256 core, we have followed the methodology presented in [5]. As a first step a detailed verification plan for the ATT8051256 verification has been developed. Verification plan for the ATT8051M256 core can be divided into following phases:

1. Functional verification of the behavioral model for the 8051 microcontroller
2. Functional verification of every module in the synthesizable RTL model
3. Functional verification of the complete RTL model
4. Verification of the synthesized ATT8051M256 model
5. Verification of mapped ATT8051M256 core
6. Verification of placed and routed ATT8051M256 core
7. Verification of external data memory accessing
8. Verification of program memory accessing
9. Verification of the interrupt system
10. Verification of timer operation
11. Verification of serial port operation
12. Random program testing
13. Self-checking testbench for the ATT8051M256 core

First, a behavioral model for the ATT8051M256 was developed and verified. This model was later used as the

reference model for the "correct ATT8051M256 behavior". Next, synthesizable RTL model for ATT8051M256 core was verified. This model was tested in two separate phases, first phase had every major module of the RTL ATT8051M256 model tested individually, and second phase where complete RTL model was tested on a set of test programs and compared with the behavioral model. Next, testing of synthesized, mapped and placed and routed ATT8051M256 core was performed on the same set of test programs, and once again compared to the behavioral model. Then, separate testing of external RAM memory access and internal and external program memory accessing was performed. Next, interrupt system, timers and serial port correct operation were verified. Next, random program testing was performed to verify the correct behavior of the core on test cases that might have been overlooked in the previous phases of verification.. Finally, a self checking testbench was developed to allow easy and quick regression testing.

Complete verification environment was written in VHDL and ModelSim VHDL simulator [6] was used to perform necessary simulations.

IV. Device Utilisation and Performance

In order to estimate the performance of the ATT8501M256 we have implemented it using various families of Xilinx FPGA devices. Implementation was performed using Xilinx ISE Foundation 9.1.03i software [7] with default synthesis and P&R options. Design was optimized for speed and all core I/O signals have been routed off-chip. Table 1 contains the device utilization measured in number of occupied slices and IOBs, together with the maximum operating frequency of the core.

TABLE 1: DEVICE UTILISATION AND PERFORMANCE

FPGA Family	***Device***	***Num of Slices***	***Num of IOBs***	***Performance [MHz]***
Spartan-3	3S1000-4	1472	142	43.5
Spartan-II	2S200-6	1430	141	43.5
Spartan-IIE	2S200E-6	1425	141	45.0
Virtex	XCV200-6	1434	141	41.5
Virtex-E	V200E-8	1425	141	50.0
Virtex-II	2V250-6	1455	142	77.0
Virtex-II Pro	XC2VP4-7	1448	142	100.0

V. Conclusion

This paper presented a design of 8051 microcontroller core intended for FPGA implementation. This core can be used in various application domains, and currently it is planned to use it as a part of the video transmission and data acquisition system for water, oil and gas exploration. In paper, detailed architecture for the developed 8051 core is presented, together with the verification plan that was used to check the correct operation of designed core. In order to estimate core's size and performance, it was implemented on various Xilinx FPGA devices.

References

[1] *MCS 51 Microcontroller Family User's Manual*, Intel Corporation, 1994.

[2] *SDCC - Small Device C Compiler*, Available: http://sdcc.sourceforge.net/

[3] G. Mancic, S. Martinovic, M. Živanov,, "Geofizicki karotaž – osnovni principi", (in Serbian), DIT NIS-Naftagas, Novi Sad, 2002

[4] P. J. Ashenden, *The Designer's Guide to VHDL*, San Francisco, Morgan Kaufmann Publishers, Inc., 1996.

[5] B. Wile, J. C. Goss, W. Roesner, *Comprehensive Functional Verification – The Complete Industry Cycle*, Elsevier Inc., 2005.

[6] *ModelSim SE User's Manual – Version 6.1f*, Menthor Graphics Corporation, 2006.

[7] *Xilinx Development System Reference Guide*, Xilinx Inc., 2007.