LOGSYS: A simple tool for complex student projects

Béla Fehér, Tamás Raikovich, György Dancsi, Péter Laczkó
Department of Measurement and Information Systems
Budapest University of Technology and Economics
{feher, rtamas, dancsi}@mit.bme.hu

Abstract

The LOGSYS Development Environment is used as a versatile tool in different levels of the education of the B.Sc. and M.Sc. Embedded System courses. The motivation at the introduction was to provide an affordable platform to every student, which offers compatibility with the existing industry standard solutions, while can support practical design work from the basic examples, through simple systems, up to the modern model based methodologies. The main components of the solution include the FPGA board, the development cable and the user application. The USB powered compact board makes possible its usage during class work, allowing direct experience on the presented topics. The LOGSYS tool set is compatible with the major Xilinx software components, like ISE, EDK, ChipScope, and offers the usability of MATLAB Simulink/System Generator and LabVIEW FPGA Module. These two later model based methodologies are very useful in case of the DSP and control type applications.

1. Introduction

By the introduction of the Bologna process in the higher education in Europe, one important aspect was to increase the ratio of the practical work during the courses. All subjects were reviewed and more examples and laboratory practice were assigned. In our university, both the EE and the CS (Technical Informatics in our terms) branches have strong laboratory courses, with good equipments, but these tasks are mainly pre assigned measurement activities, limiting the possibilities of the self made development experience. As it is well known, that the practice makes perfect, or at least gives support to extend the knowledge, students own experience with their selected projects could significantly improve the efficiency of the learning process. Every educational tool should support the interested student to implement their own idea and should be easy to use to allow stepping over the painful introduction of the fundamentals, offering learning through discovery. Usually the first real sense of achievements can lead to strong dedication to the required commitment to the engineering courses.

The modern embedded computing system design includes both microelectronic digital hardware and programmable software components. With the availability of the FPGA devices, these fields are closer to each other, and a unified platform can be used to introduce most aspect of systems on a (programmable) chip design, architecture exploration, HW/SW partition, debug technologies, soft core development and model based developments.

The LOGSYS Development Environment has been introduced to cover these requirements. It started as a student project, and extended after positive reception. In addition of its affordable cost, the main advantage is the simplicity of the tools, a laptop (or PC) is enough to set up the environment. This allows its use not only in laboratory environment, but even during classroom practice. For student project works, the system is open for hardware extension module design and software application development.

2. Curriculum support

The LOGSYS demonstration tools are introduced early during the courses and used throughout the final years. The CS curriculum in the first two semesters includes Digital Design (1 & 2), which is a good subject to use the platform for presentation of examples and offer it for interested students. The tasks of first semester of the Measurement Laboratory are based mainly on Xilinx FPGA technology, so a priori knowledge is an advantage. The main area, where the LOGSYS boards are used is the Embedded Systems specialization. The students got their own development boards, and they can use it during the laboratory work, and for their own projects. Three elective subjects offer

978-1-4244-4406-9/09/$25.00 ©2009 IEE

more detailed knowledge. These elective subjects are the Logic Design, Development of Microsystems on FPGA Circuits, and DSP Applications. All three subjects put emphasis on applications, which means realization of real world solutions to the presented problems on the LOGSYS platform. Above the classroom examples, students are preparing a home work project, which are frequently based on ideas presented by them.

In some cases the platform is used also during thesis work, when the task is to develop new application programs or extensions to the current system.

The following paragraphs introduce the components, and some typical applications.

3. LOGSYS development environment

The LOGSYS Development Environment [1] is a versatile, easy-to-use and cost-effective educational and development tool mainly for FPGA based systems. It has three main parts (Figure 1): the development cable, the FPGA board and the user application.

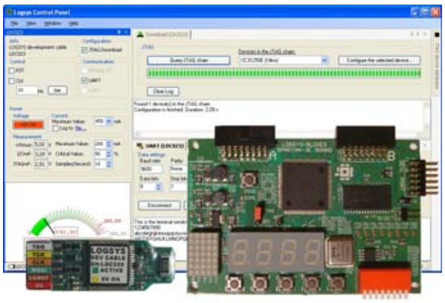


Figure 1. The LOGSYS system.

The USB based development cable connects the target system with the PC and provides configuration, control and serial communication interfaces and 5V power supply output. The development cable functions can be accessed through the graphical user interface. The serial communication interface can be used from Windows applications as a virtual COM port.

The FPGA board contains a Xilinx XC3S250E device, which enables to implement complex logics and smaller microprocessor systems, as well as simple signal and image processing functions. The board contains simple peripherals, which are suitable for the beginners. Expansion modules containing more advanced peripherals (VGA, ADC, Ethernet, etc.) can be connected to the board also.

The LOGSYS system is completed with a special application called Xilinx Platform Cable USB emulator, which enables all Xilinx design tools (ISE, EDK, ChipScope and System Generator) to access the JTAG interface of the development cable. This way, the LOGSYS tool can mimic the Xilinx download cables and allow the students to practice in classrooms using laptops or even at home.

4. Basic Applications

The approach in a modern Digital Design curriculum is typically RTL-focused. After the presentation of the basic knowledge and the most important synthesis elements of the used HDL language (in our case Verilog), the firsts entry level designs, like counters, shift registers, pattern recognizer FSMs, etc. are implemented by the students. To demonstrate the operation of different functional units and to realize standard or custom functions using RTL HDL descriptions, the Xilinx ISE WebPACK software environment is very appropriate. The ISE built in modules, like language templates, syntax checker, simulator and RTL schematic viewer makes easier the first steps. The debug of the real circuit is supported by the LOGSYS Bit Bang control panel, with single or burst clock pulses, arbitrary serial input and direct output on different displays. An example design task from the text book [2] used in one of the courses is a simple soda dispenser with the classical controller–datapath decomposition. The states of the FSM (INIT, WAIT, ADD, DISPENSE), the soda price and the current total value of the coins are displayed on the dot matrix and seven segment units, see (Figure 2).

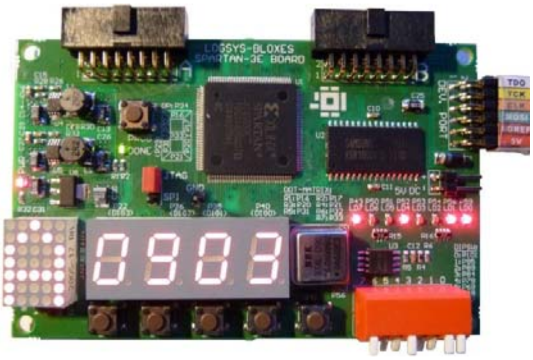


Figure 2. The soda dispenser in the ADD state.

Development of complex design tasks are supported by high level register/FIFO/memory interfaces, realized through the USB serial channel. This makes easier the setup and test of wide variety of projects.

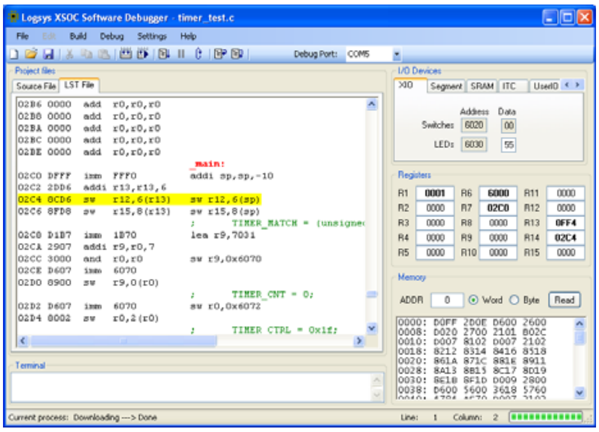
5. SoPC development

Selecting the Development of Microsystems with FPGA Circuits elective class, students can learn about SoPC systems, the design of soft core microprocessors, on chip bus systems, peripheral components and other related topics. This subject is a good extension to the Embedded and Ambient Systems Laboratory, where students already have experiences programming a MicroBlaze processor to realize a basic calculator with PS/2 keyboard input and VGA output. The class is mainly based on the XUP Embedded System curriculum, it covers the general CPU design, architecture and instruction set selection and offer a good coverage of different soft processor implementations from the simple 8/16 bit controllers to 32 bit CPUs.

5.1. XSOC

The XSOC [3] is a simple and open-source SoPC design. The core of the XSOC is the XR16 CPU, which is a 16-bit RISC processor with sixteen 16-bit general purpose registers and a high level C compiler support based on LCC [4]. Because of its simplicity, this processor is well suitable for introducing the basics of the microprocessor systems without any special software requirements. Another advantage of the XSOC is its small size: the original version was implemented in a Xilinx XC4005XL FPGA and utilized only 196 CLBs.

The XR16 CPU has been ported to the LOGSYS Spartan-3E FPGA board. A graphical environment is available for software development and debugging purposes (Figure 3).



```
02B6 0000  add  r0,r0,r0
02B8 0000  add  r0,r0,r0
02BA 0000  add  r0,r0,r0
02BC 0000  add  r0,r0,r0
02BE 0000  add  r0,r0,r0
                              _main:
02C0 DFFF  imm  FFF0          addi sp,sp,-10
02C2 2DD6  addi r13,r13,6
02C4 8CD6  sw   r12,6(r13)    sw r12,6(sp)
02C6 8FD8  sw   r15,8(r13)    sw r15,8(sp)
                              ;      TIMER_MATCH = (unsigne
02C8 D1B7  imm  1B70          lea r9,7031
02CA 2907  addi r9,r0,7
02CC 3000  and  r0,r0         sw r9,0x6070
02CE D607  imm  6070
02D0 8900  sw   r9,0(r0)
                              ;      TIMER_CNT = 0;
02D2 D607  imm  6070          sw r0,0x6072
02D4 0002  sw   r0,2(r0)
                              ;      TIMER CTRL = 0x1f;
```

	Address	Data
Switches	6020	00
LEDs	6030	55

R1	0001	R6	6000	R11	0000
R2	0000	R7	02C0	R12	0000
R3	0000	R8	0000	R13	0FF4
R4	0000	R9	0000	R14	02C4
R5	0000	R10	0000	R15	0000

ADDR 0 ● Word ○ Byte Read

```
0000: D0FF 2D0E D600 2600
0008: D020 2700 2101 802C
0010: D007 8102 D007 2102
0018: 8212 8314 8416 8518
0020: 861A 871C 881E 8911
0028: 8A13 8B15 8C17 8D19
0030: 8E1B 8F1D D009 2800
0038: D600 5600 3618 5760
```

Current process: Downloading ---> Done

Line: 1 Column: 2

Figure 3. The LOGSYS XSOC debugger.

5.2. Xilinx EDK

The main development platform during the course is the Xilinx EDK toolkit to design more complex FPGA based microprocessor systems. It supports the hard PowerPC (for selected Virtex devices only) and the soft MicroBlaze cores. The EDK provides wide range of hardware peripherals and the necessary tools for software development (C/C++ compiler, debugger).

The board support package (XBD file) required by the Base System Builder Wizard is available for the LOGSYS Spartan-3E FPGA board. Using this wizard allows the user to create a ready-to-use base system including the on-board peripherals with only a few mouse clicks.

The laboratory projects of the Embedded Design Flow Workshop course of the Xilinx University Program [5] has been adapted to the LOGSYS Spartan-3E FPGA board. Only a little modification was required in the original design, Figure 4 shows the block diagram of the new system.

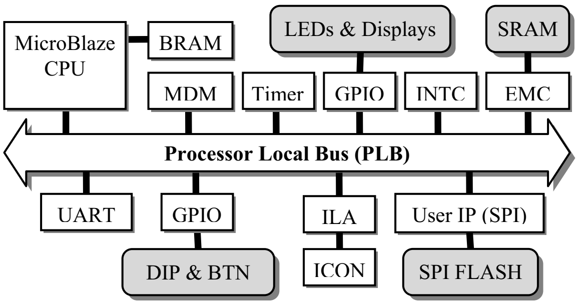


Figure 4. The XUP MicroBlaze reference design

Xilinx offers advanced verification tools. The GNU Debugger is available in the EDK for debugging software applications. With the ChipScope on-chip logic analyzer (Figure 5), it is also possible to monitor internal hardware signals. These two tools can be used together when HW/SW co-debugging is required.

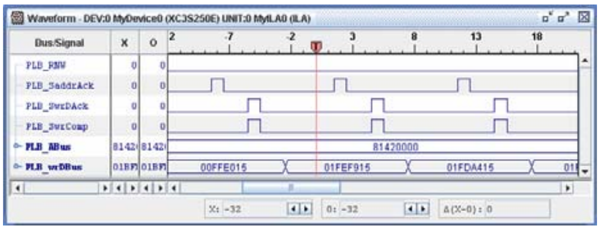


Figure 5. Monitoring PLB on chip bus transfers

The LOGSYS development cable not only provides power, and supports all the JTAG based functions necessary during the EDK or ChipScope debugging work, but offers direct consol interface through the built in USB COM port for the MicroBlaze program.

6. Model based development

Model based application development is a modern, high-level development method. Usually the design is entered using a graphical programming language and the development tool generates the C/C++ or HDL code from the system model. The underlying hardware is hidden from the user therefore no deep hardware knowledge is required. This method has become more and more popular to implement DSP and control systems.

Implementing simple DSP functions with System Generator is one of the lab exercises at Embedded and Ambient Systems Laboratory. The DSP Applications elective class offers a more detailed review of the FPGA based DSP applications and the students can choose such homework that is realizable with System Generator.

The necessary software extensions are available for MATLAB Simulink/System Generator and LabVIEW FPGA Module that makes possible to use these model based development tools together with our LOGSYS FPGA board. Although the XC3S250E FPGA device complexity is relatively limited, simple designs can be easily implemented. The familiar hardware environment makes easier to focus to the design methodologies, speeding up the learning curve.

6.1. MATLAB Simulink

Simulink is a graphical block diagramming tool for modeling, simulating and analyzing multidomain dynamic systems. System Generator for DSP is a Xilinx blockset for Simulink that enables to model and create complex DSP systems, and generate HDL code targeting the Xilinx FPGA devices.

The DSP for FPGA Primer course of the Xilinx University Program [5] covers a wide range of theoretical DSP topics and go in depth into how such algorithms can be implemented in the Xilinx FPGA technology using System Generator. Many examples presented in this course can be implemented on the LOGSYS Spartan-3E FPGA board. Figure 6 shows such an example design: an IIR filter based sine wave oscillator and its hardware co-simulation.

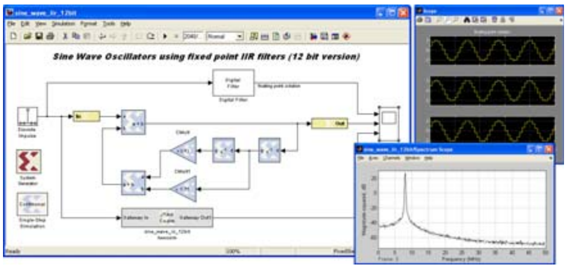

Sine Wave Oscillators using fixed point IIR filters (12 bit version)
In
Out
Digital Filter
System Generator

Figure 6. An oscillator using System Generator.

6.2. LabVIEW FPGA Module

The NI LabVIEW visual programming environment is an efficient engineering tool to develop a wide range of applications. The LabVIEW FPGA Module enables the implementation of the different signal processing or control functions directly in digital hardware. All advantage of the high level graphical programming can be exploited and intuitive visual user interfaces can be generated. The LOGSYS board is completed with the LabVIEW FPGA Module required parameter description data bases. The communication between the PC and the FPGA VI is based on the standard JTAG interface. An example robot arm control is shown on Figure 7. The five joint mechanism is controlled by TowerPro SG5010 servos, the fingers are activated by a smaller SG91R. The servos require external +5VDC power source, the limited USB power operates only the FPGA board. The two boards have galvanic isolation. All servos are controlled by simple PWM modules, using an NCO.

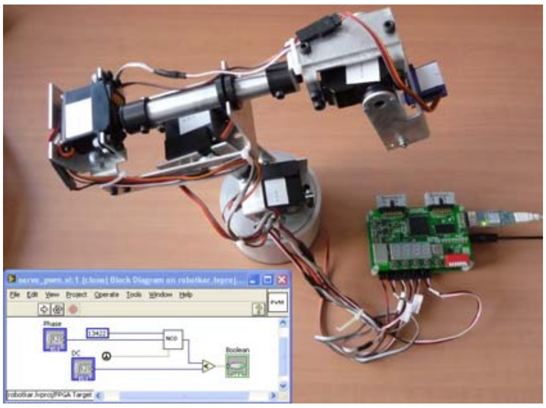


Figure 7. Robot arm control by LabVIEW.

7. Summary

The easy to use LOGSYS educational development board and its typical applications in the different courses are presented. The main advantage of the platform is its applicability for all levels of education, especially in our Embedded Systems specialization. It can be used not only in laboratory projects developments, but even in classroom examples and demonstrations. External hardware modules expand its applicability to special application areas.

8. References

[1] B. Fehér, T. Raikovich, P. Laczkó, "LOGSYS Development Environment", Proc. of EWME, Budapest, May, 2008, pp. 94–95.
[2] F. Vahid, "Digital Design", John Wiley & Sons, 2006 pp. 227–230.
[3] J. Gray, "FPGA CPU News", www.fpgacpu.org/
[4] http://www.cs.princeton.edu/software/lcc
[5] Xilinx University Program at http://www.xilinx.com/univ
[6] LabVIEW FPGA at http://www.ni.com/fpga/