The Design of a Rapid Prototype Platform for ARM Based Embedded System

Rui Wang and Shiyuan Yang, Senior Member, IEEE

Abstract — *Hardware prototype is a vital step in the embedded system design. In this paper, we discuss our design of a fast prototyping platform for ARM based embedded systems, providing a low-cost solution to meet the request of flexibility and testability in embedded system prototype development. It also encourages concurrent development of different parts of system hardware as well as module reusing.*

Though the fast prototyping platform is designed for ARM based embedded system, our idea is general and can be applied to embedded system of other types. [1].

Index Terms — Rapid prototyping, Embedded system, ARM, FPLD, Reconfigurable Interconnection.

I. INTRODUCTION

Embedded systems are found everywhere, including in cellular telephones, pagers, VCRs, camcorders, thermostats, curbside rental-car check-in devices, automated supermarket stockers, computerized inventory control devices, digital thermometers, telephone answering machines, printers, portable video games, TV set-top boxes -- the list goes on. Demand for embedded system is large, and is growing rapidly.

In order to deliver correct-the-first-time products with complex system requirements and time-to-market pressure, design verification is vital in the embedded system design process. A possible choice for verification is to simulate the system being designed. If a high-level model for the system is used, simulation is fast but may not be accurate enough, with a low-level model too much time may be required to achieve the desired level of confidence in the quality of the evaluation. Since debugging of real systems has to take into account the behavior of the target system as well as its environment, runtime information is extremely important. Therefore, static analysis with simulation methods is too slow and not sufficient. And simulation cannot reveal deep issues in real physical system.

A hardware prototype is a faithful representation of the final design, guarantying its real-time behavior. And it is also the basic tool to find deep bugs in the hardware. For these reasons, it has become a crucial step in the whole design flow. Traditionally, a prototype is designed similarly to the target system with all the connections fixed on the PCB (printed circuit boards).

[1] Rui Wang is with the Department of Automation, Tsinghua University, Beijing, China (e-mail: wr01@mails.tsinghua.edu.cn).

Shiyuan Yang, was with the Department of Automation, Tsinghua University, Beijing, China (e-mail: ysy-dau@mail.tsinghua.edu.cn).

Contributed Paper
Manuscript received March 30, 2004

As embedded systems are getting more complex, the needs for thorough testing become increasingly important. Advances in surface-mount packaging and multiple-layer PCB fabrication have resulted in smaller boards and more compact layout, making traditional test methods, e.g., external test probes and "bed-of-nails" test fixtures, harder to implement. As a result, acquiring signals on boards, which is beneficial to hardware testing and software development, becomes infeasible, and tracking bugs in prototype becomes increasingly difficult. Thus the prototype design has to take account of testability. However, simply adding some test points is not enough. If errors on the prototype are detected, such as misconnections of signals, it could be impossible to correct them on the multiple-layer PCB board with all the components mounted. All these would lead to another round of prototype fabrication, making development time extend and cost increase.

Besides testability, it is important to maintain high flexibility during development of the prototype as design specification changes are common. Nowadays complex systems are often not built from scratch but are assembled by reusing previously designed modules or off-the-shelf components such as processors, memories or peripheral circuitry in order to cope with more aggressive time-to-market constraints. Following the top-down design methodology, lots of effort in the design process is spent on decomposing the customers' requirements into proper functional modules and interfacing them to compose the target system.

Some previous research works have suggested that FPLDs (field programmable logic device) could be added to the final design to provide flexibility as FPLDs can offer programmable interconnections among their pins and many more advantages. However, extra devices may increase production cost and power dissipation, weakening the market competition power of the target system. To address these problems, there are also suggestions that FPLDs could be used in hardware prototype as an intermediate approach[1]-[3], whereas this would still bring much additional work to the prototype design. Moreover, modules on the prototype cannot be reused directly. In industry, there have been companies that provide commercial solutions based on FPLDs for rapid prototyping[4]. Their products are aimed at SOC (system on a chip) functional verification instead of embedded system design and development.

In this paper, we discuss our design of a Rapid Prototyping Platform for ARM based Embedded System, providing a low-cost solution to meet the request of flexibility and testability in embedded system prototype development. It also

encourages concurrent development of different parts of system hardware as well as module reusing.

The rest of the paper is organized as follows. In section 2, we discuss the details of our rapid prototyping platform. Section 3 shows the experimental results, followed by an overall conclusion in section 4.

II. The Design of a Rapid Prototyping Platform

A. Overview

ARM based embedded processors are wildly used in embedded systems due to their low-cost, low-power consumption and high performance. An ARM based embedded processor is a highly integrated SOC including an ARM core with a variety of different system peripherals[5]. Many arm based embedded processors, e.g.[6]-[8], adopt a similar architecture as the one shown in Fig. 1.

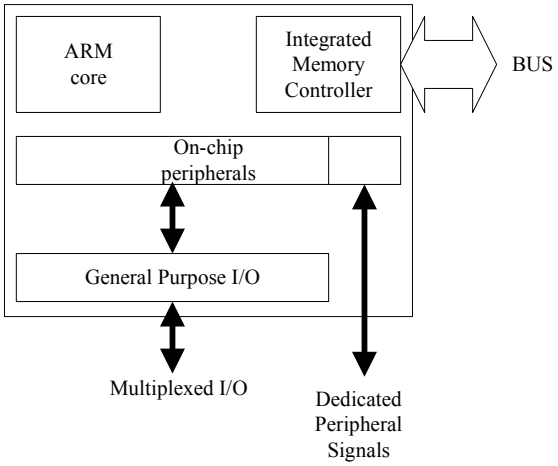


Fig. 1. Arm-based embedded processor block diagram

The integrated memory controller provides an external memory bus interface supporting various memory chips and various operation modes (synchronous, asynchronous, burst modes). It is also possible to connect bus-extended peripheral chips to the memory bus. The on-chip peripherals may include interrupt controller, OS timer, UART, I2C, PWM, AC97, and etc. Some of these peripherals signals are multiplexed with general-purpose digital I/O pins to provide flexibility to user while other on-chip peripherals, e.g. USB host/client, may have dedicated peripheral signal pins. By connecting or extending these pins, user may use these on-chip peripherals. When the on-chip peripherals cannot fulfill the requirement of the target system, extra peripheral chips have to be extended.

The architecture of an ARM based embedded system is shown in Fig. 2. The whole system is composed of embedded processor, memory devices, and peripheral devices. To enable rapid prototyping, the platform should be capable of quickly assembling parts of the system into a whole through flexible interconnection. Our basic idea is to insert a reconfigurable interconnection module composed by FPLD into the system to provide adjustable connections between signals, and to provide testability as well. To determine where to place this

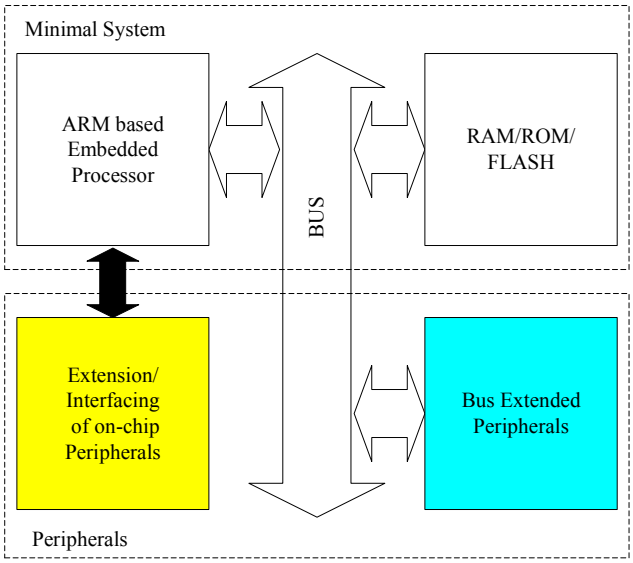


Fig. 2. The architecture of an ARM based embedded system

module, we first analyze the architecture of the system.

The embedded system shown in Fig. 2 can be divided into two parts. One is the minimal system composed of the embedded processor and memory devices. The other is made up of peripheral devices extended directly from on-chip peripheral interfaces of the embedded processor, and specific peripheral chips and circuits extended by the bus.

The minimal system is the core of the embedded system, determining its processing capacity. The embedded processors are now routinely available at clock speeds of up to 400MHz, and will climb still further. The speed of the bus connecting the processor and the memory chips is exceeding 100MHz. As pin-to-pin propagation delay of a FPLD is in the magnitude of a few nanoseconds, inserting such a device will greatly impair the system performance.

The peripherals enable the embedded system to communicate and interactive with the circumstance in the real world. In general, peripheral circuits are highly modularized and independent to each other, and there are hardly needs for flexible connections between them.

Here we apply a reconfigurable interconnection module to substitute the connections between microcomputer and the peripherals, which enables flexible adjusting of connections to facilitate interfacing extended peripheral circuits and modules. As the speed of the data communication between the peripherals and the processor is much slower than that in the minimal system, the FPLD solution is feasible.

Following this idea, we design the Rapid Prototyping Platform as shown in Fig. 3. We define the interface ICB between the platform and the embedded processor core board that holds the minimal system of the target embedded system. The interface IPB between the platform and peripheral boards that hold extended peripheral circuits and modules is also defined. These enable us to develop different parts of the target embedded system concurrently and to compose them into a prototype rapidly, and encourage module reusing as

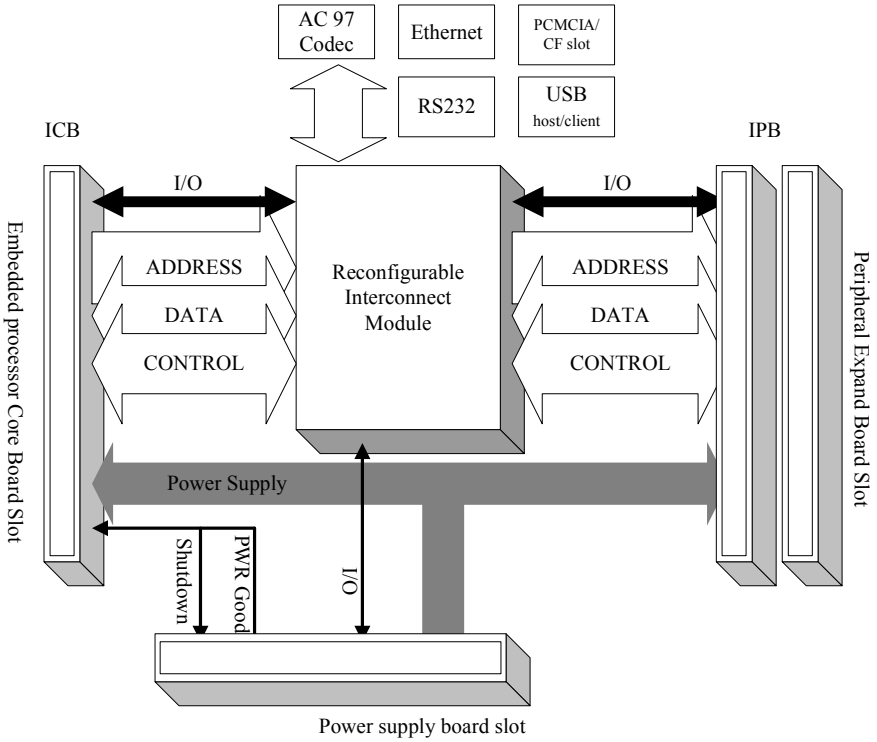


Fig. 3. Schematic of the Rapid Prototyping Platform

well. The two interfaces are connected by a reconfigurable interconnect module. There are also some commonly used peripheral modules, e.g. RS232 transceiver module, bus-extended Ethernet module, AC97 codec, PCMCIA/CompactFlash Card slot, and etc, on the platform which can be interfaced through the reconfigurable interconnect module to expedite the embedded system development.

B. Reconfigurable Interconnect Module

With the facility of state-of-arts FPLDs, we design a reconfigure interconnection module to interconnect, monitor and test the bus and I/O signals between the minimal system and peripherals.

As the bus accessing obeys specific protocol and has control signals to identify the data direction, the interconnection of the bus can be easily realized by designing a corresponding bus transceiver into the FPLD, whereas the interconnection of the I/Os is more complex. As I/Os are multiplexed with on-chip peripherals signals, there may be I/Os with bi-direction signals, e.g. the signals for on-chip I2C[9] interface, or signals for on-chip MMC (MultiMedia Card[10]) interface. The data direction on these pins may alter without an external indication, making it difficult to connect them via a FPLD. One possible solution is to design a complex state machine according to corresponding accessing protocol to control the data transfer direction. In our design we assign specific locations on the ICB and IPB interfaces to these bi-direction signals and use some jumpers to directly connect these signals when needed. The problem is circumvented at the expense of losing some flexibility.

The use of FPLD to build the interconnection module not only offers low cost and simple architecture for fast prototyping, but also provides many more advantages. First, interconnections can be changed dynamically through internal logic modification and pin re-assignment to the FPLD. Second, as FPLD is connected with most pins from the embedded processor, it is feasible to detect interconnection problems due to design or physical fabricate fault in the minimal system with BST (Boundary-Scan Test, IEEE Standard 1149.1 specification). Third, it is possible to route the FPLD internal signals and data to the FPLD's I/O pins for quick and easy access without affecting the whole system design and performance. It is even possible to implement an embedded logical analyzer into the FPLD to smooth the progress of the hardware verification and software development.

C. *Power Supply*

Power dissipation has a great impact on system cost and reliability. It is an increasingly important problem in embedded systems designs not only for the portable electronics industry but in other areas including consumer electronics, industry control, communications, etc. In order to facilitate the design of power supply for the target embedded system, power supply issues have also been considered in the design of the platform.

First, the power supplies to the devices on the platform are separated from those to the core board and peripheral expand boards, which makes it more realistic to measure and verify the power dissipation on the platform for the target embedded system. Second, the power supplies for the core board and

TABLE I
EXAMPLE OF POWER SUPPLY REQUIREMENTS FOR SOME WILDLY USED ARM BASED EMBEDDED PROCESSORS

Embedded processor	Power supply requirements
Intel XScale PXA255	+0.85~1.3V for Core and PLL Power Supply +2.5V or +3.3V for Memory Bus +3.3V for Other CMOS I/Os
Samsung S3C2410	+1.8V for Core Supply +3.3V for I/O Block and Analog Core
Cirrus Logic Maverick EP7312	+2.5V for Core and PLL Power Supply +2.5V~3.3V for I/O input/output

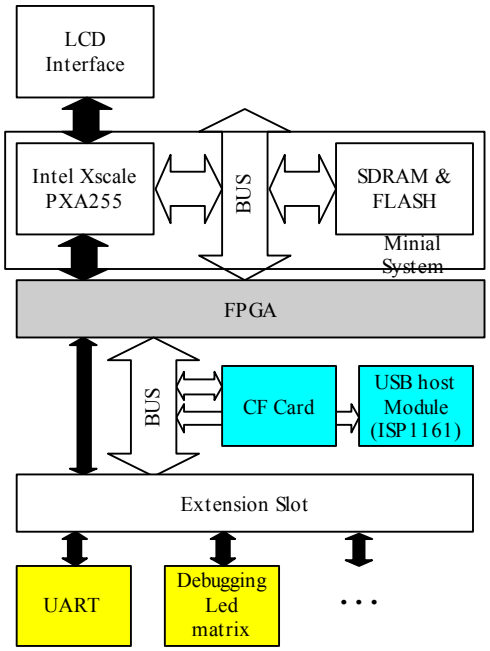
peripheral expand boards are built on a separate board and connected to the platform through a slot. As a result, it provides flexibility for power system design while speeding up the whole design process.

To meet the demand for higher system speed and lower power consumption in data communications and processing, embedded processor vendors use increasingly advanced processing technologies requiring lower core operating voltages, and keep the interface voltage compatible with most low voltage semiconductor devices on market. Consequently, almost every embedded processor requires more than one power supply, such as power supply for internal logic, for PLLs and oscillators, for memory bus interface, and for other I/Os. Further, different embedded processors may have different requirements on power supply, such as different power supply voltage, power-up sequence, and different strategies to adjust the core voltage in different CPU run mode for minimizing power dissipation.

A survey of some widely used ARM based embedded processor suggests that most of them need no more than 3 groups of separated power supply, as shown in Table 1. As the peripherals may require different supply voltages for special purpose, such as +5V for powering the USB ports, we divide the power supply from the power supply slot into 4 separated channels, which is connected to both the core board slot and the peripheral board slot. Each channel of power supplies has a "power good" signal to indicate power output status of the channel, and a shutdown signal to shut the power supply of the channel down. And these signals are directly connected to the core board slot to accommodate the embedded processor's requirement of power-up sequence. In order to enable dynamic voltage adjusting, some control signals are routed to the power supply board by the reconfigurable interconnect module.

III. EXPERIMENTAL RESULTS

As the Rapid Prototyping Platform is still under development, we present an example applied with the same considerations in the Rapid Prototyping Platform. It is an embedded system prototype based on Intel XScale PXA255, which is an ARM based embedded processor. The diagram of the prototype is illustrated in Fig. 5(a). The photo is shown in



(a) Block diagram

(b) Photo

Fig. 5. The diagram and the photo of the prototype.

Fig. 5(b), where a Bluetooth module is connected to the prototype USB port and a CF LAN card is inserted.

The FPGA (an Altera Cyclone EP1C6F256) here offers the same function as the reconfigurable interconnection module shown in Fig. 3. Most of the peripheral devices are expanded to the system through the FPGA, and more peripherals can be easily interfaced when needed. As both of the FPGA and PXA255 support the BST, we can detect faults, e.g. short-circuit and open-circuit faults, on the connections between the two devices by chaining their JTAG ports and performing BST. Here, we use an open source software package [11] to perform the BST.

The FPGA internal signals can be routed to the debugging LED matrix for easy access, which is helpful in some simple testing and debugging. We also insert an embedded logical

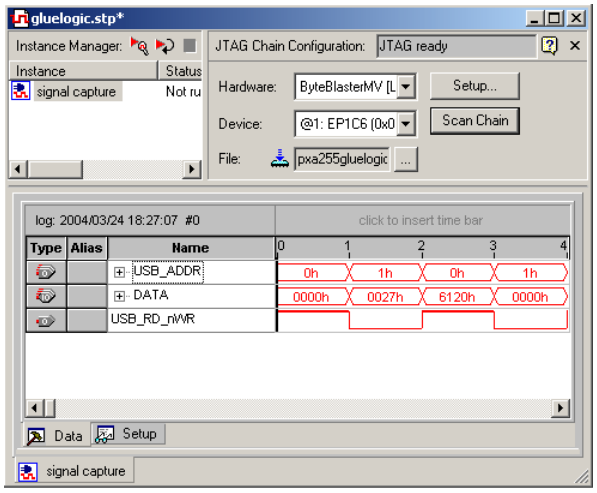


Fig. 6. Logical analyzer capture result

analyzer, the SignalTap II embedded logic analyzer [12] provided in Altera's Quartus II software, into the FPGA for handling more complicated situations. With the help of the logical analyzer, we are able to capture and monitor data passing through over a period of time, which expedites the debugging process for the prototype system. Fig. 6 shows the captured data communication between the embedded processor and the USB host module during the initialization process of the Philips ISP1161 USB host chip[13]. It can be seen clearly from the figure that the embedded processor writes the command code of 0027H to address 01H (the ISP1161's host controller command port) to access the HcChipID register, and reads 6120H (the chip's ID) from address 00H (the ISP1161's host controller data port).

The power supply module of the prototype system is held on a separate board connected to the system via a socket. We designed two power supply modules for the prototype system (shown in Fig. 7). One is a large module providing fixed output composed with simple but low-efficiency linear regulator (the upside one in the Fig. 7), the other is a compact module, capable of dynamic voltage adjusting, made up of

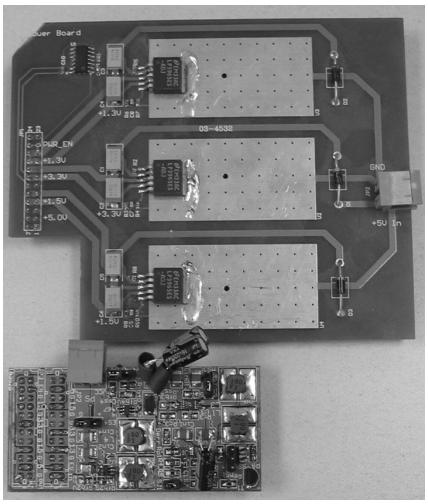


Fig. 7. Power supply modules.

complex high-efficiency switch regulator(the downside one in the Fig. 7). The former module is first used to accommodate the basic power supply requirements during hardware test and relative software development. During the process, the later is developed and refined, and replaced the former one finally. The separation of power supply module from prototype allows refinement of the power supply module without affecting development of other parts of the system.

With the help of this flexible prototype, we finished our hardware development and related software development, including boot-loader development, OS (Arm-Linux) transplant, and driver development in about one week.

IV. Conclusion

In this paper, we discuss the design of a fast prototyping platform for ARM based embedded systems to accommodate the requirements of flexibility and testability in the prototyping phase of an embedded system development.

With the aid of the platform, modules of the target embedded system can be developed simultaneously, and previous modules can be applied to future designs. As a result, develop process is greatly accelerated.

Though the fast prototyping platform is designed for ARM based embedded system, our idea is general and can be applied to embedded system of other types.

References

[1] S. Trimberger, "A Reprogrammable Gate Array and Applications," Proc. IEEE, Vol. 81, No. 7, July 1993, pp. 1030-1041.
[2] Hauck, S, "The roles of FPGAs in reprogrammable systems", Proc. IEEE , Vol. 86 , Issue: 4 , April 1998, pp. 615 – 638
[3] Cardelli, S.; Chiodo, M.; Giusto, P.; Jurecska, A.; Lavagno, L.; Sangiovanni-Vincentelli, A. "Rapid-Prototyping of Embedded Systems via Reprogrammable Devices," Rapid System Prototyping, 1996. Proceedings., Seventh IEEE International Workshop on.
[4] Product Briefs for System Explorer Reconfigurable Platform with debugging capabilities, http://www.aptix.com/literature/productbriefs/sysExpl.pdf
[5] Steve Furber, Stephen B., ARM system-on-chip architecture, Addison-Wesley, 2000.
[6] Intel® PXA255 Processor Developer's Manual, http://www.intel.com/design/pca/applicationsprocessors/manuals/278693.htm
[7] Samsung S3C2410x User Manual, http://www.samsung.com/Products/Semiconductor/SystemLSI/MobileSolutions/MobileASSP/MobileComputing/S3C2410X/um_s3c2410s_rev12_030428.pdf
[8] Cirrus Logic Maverick EP7312 Manual, http://www.cirrus.com/en/pubs/proDatasheet/EP7312-5.pdf
[9] I2C bus specification, http://www.semiconductors.phiips.com/acrobat/literature/9398/39340011.pdf
[10] The MultiMediaCard System Summary, http://www.mmca.org/tech/MMC-System-Summary-v3.31.pdf
[11] JTAG Tools, http://openwince.sourceforge.net/jtag/
[12] Altera Application note, Design Verification Using the SignalTap II Embedded Logic Analyzer, http://www.altera.com/literature/an/an280.pdf
[13] Philips ISP1161 datasheet, http://www.semiconductors.philips.com/acrobat/datasheets/ISP1161-02.pdf

Rui Wang Rui Wang received the B.S degree in Industry Automation from Hunan University in 1997, and the M.S degree in Control Theory and Control Engineering from Guangdong University of Technology in 2001. Now he is pursuing his Ph.D. degree at Tsinghua University. His main research interests are embedded system design and implement, multimedia signal processing and communication.

Shiyuan Yang Shiyuan Yang was born in Shanghai, China. He received the B.S. and M.S. degrees from Tsinghua University, Beijing, China, in 1970 and 1981, respectively. Since 1993, he is a professor in Automation Department at Tsinghua University, Beijing, China. He is an Associate Director of the FTC committee, China. His main research interests are home automation network, test technology, electronic technology application, system fault diagnosing.