

The μ PP ASIC: Design, Methodologies and Tools for a Pay Phone System-On-a-Chip Based on an ARM Core and Design Reuse

Jacobo Riesco, Juan C. Díaz and Pierre I. Plaza

Hardware Technology Division, Telefónica Investigación y Desarrollo.
Emilio Vargas, 6. 28043 Madrid. Spain

Abstract

This paper describes the μ PP (Microcontroller for Pay Phones) ASIC, a system-on-a-chip solution based on the present Spanish pay phone system. The design integrates an ARM embedded microprocessor, several third party blocks and new custom modules developed in house, using ARM's Advanced Microprocessors Bus Architecture (AMBA). The system has been designed for low power consumption and management. Design reuse, aided by the use of new management tools, and co-design, allowed an important reduction in global design time.

Introduction

Traditionally, microelectronic products for the telecommunications market have been characterised by its long production life and high reliability needs. While this is still true, some new events in the scene, as the European telecommunications deregulation, the Internet era and the multimedia services among others, are driving important transformations. Also technical factors, as the emerging importance of design and Intellectual Property (IP) reuse, new technologies allowing system on a chip integration, and new design and management tools (1), enable significant improvements in design quality, cost and time to market, which are key factors in an ever increasing competitive market.

In this context, Telefónica I+D (TI+D), the Research and Development subsidiary of the multinational telecommunications company Telefónica has undertaken a new redesign of its current pay phone system, the TM (Modular Telephone), developed also inside TI+D nearly ten years ago with more than 300,000 units deployed in Spain and other 18 countries all over the world (2).

This new design, the μ PP microcontroller ASIC, integrates most of the functionality of the TM digital board, providing a system-on-a-chip (SOC) solution for pay phones, ready to support new services, with increased reliability and performance, and with lower power consumption and cost.

To cope with the tight schedule constraints to face competition and to provide enhanced services, the selected solution was based on the use of an ARM embedded microprocessor core and third parties' IP modules. Furthermore, the circuit was developed as a demonstrator within the ESPRIT project 20724 (TOMI, Tools for OMI) whose aim is to supply tools and methods that make practical the OMI (Open Microprocessors Initiative) concept of super-cell based system design.

General Description

The μ PP ASIC is based on the well-proven TM system, a fully autonomous system where even supply is taken from the telephone line. The TM comprises two main boards, an analog one, with the line and mechanical devices interfaces and drivers, and the supply generation subsystem; and a digital board, containing two processors, an 8 bits 8051 microcontroller for power supervision and management, and a 16 bits 8051XA, to control all the pay phone functions and handling the peripherals (keyboard, display, pay elements, communications, etc.) which are interfaced through low power EPLDs (Electrically Programmable Logic Devices) also on the board.

The objective of the μ PP is to replace the functionality of the two microcontrollers in the TM digital board, also integrating the peripheral interfaces (EPLDs) and some memory, to provide an enhanced direct replacement for that board (the analog board has not been considered). Its higher integration level increases reliability and decreases cost, while its higher performance enables an important set of new services.

One main concern in the design was the power consumption. Drawing current from the telephone line imposes serious restrictions both in peak (maximum current) and mean power available to the whole system, so a low power-high performance 32 bits embedded microprocessor was chosen: the ARM7TDMI (6) from Advanced Risc Machines Ltd. This single processor performs the functions of both 8051 microcontrollers in the digital board, with the aid of a custom programmable power and clock control module to allow an efficient and flexible energy handling (7).

Other factors -besides low power and high performance- taken into account when selecting the ARM core were: its small size, its availability from different ASIC vendors (several possible second source providers for μ PP), the increasing availability of commercial software (Real time operating systems, soft modems, web servers, etc.) and hardware (USART, I²C, etc.) modules for it, its compressed 16 bits instruction set (the *Thumb* mode), and that it provides a software debug interface (the *ICEbreaker*) through a JTAG port.

Architectural and Functional Description

The μ PP ASIC is based on the hierarchical ARM's Advanced Microprocessors Bus Architecture (AMBA), a specification for an on-chip bus, that enable macrocells as CPUs, peripherals and memory controllers to be connected together to form a microcontroller or complex peripheral

chip. AMBA specification (8) defines a high-speed, high-bandwidth multimaster bus, the Advanced System Bus (ASB) and a simple low-power, low-speed unlocked bus, the Advanced Peripheral Bus (APB), designed to be a secondary bus to ASB, connected by a bridge (which limits ASB load).

A. Design Reuse

To reduce design time and also increase the confidence in the final results, design reuse has been extensively applied, using hard, firm and soft IP cores (1), (3) from different providers whenever available within the μ PP ASIC.

Hard cores are proprietary design elements supplied by the silicon vendor; only a bus simulation model (4) and an abstract for P&R are provided. The ATMEL ARM7TDMI cell and some generated memories belong to this category.

Firm cores -ATMEL USARTs and Philips I²C interface (9)- are in the form of netlists, so they can be simulated and optimized during synthesis but they are quite difficult to modify, and thus, they have been used as given.

Soft cores are provided as synthesisable VHDL RTL code that can be simulated, synthesised and even modified, so value can be added to the design. ARM provides a set of parameterized RTL modules with their synthesis scripts and compliance test suites: the *micropack*, a simple example AMBA system including all the necessary components to implement the ASB and APB functionality, and some sample peripherals. Some of these soft cores were modified or given proper parameter values to fit the exact circuit needs.

Besides the third party cores or virtual components (VCs) some AMBA compliant custom blocks for specific peripherals were developed from scratch by TI+D's designers.

Fig. 1 depicts the μ PP circuit architecture, its hard, firm, soft and custom blocks and the percentage of final chip area (excluding memory) of each block type.

B. ASB Bus Blocks

The μ PP ASB bus has two bus masters and several slave blocks that may be accessed and controlled by either master. It is the arbiter's function to grant the ASB control to the mas-

ters: the ARM, through the wrapper, or the Test Interface Controller (TIC). This latter is used to allow an external tester to take control of the ASB bus through the external bus/memory interface, offering a fast parallel access to check all internal components separately.

The internal memory interface controls the accesses to an on chip 8Kx32bit SRAM addressable in byte, half-word or word mode. The external memory interface provides a programmable 8/16/32 bits data bus interface with up to 7 external memory devices 256 Mbytes each; wait states may be also programmed, and supports the same address modes as the internal memory.

Address decoding is performed by a centralised decoder block, which selects any of the ASB slave blocks, making them independent of the absolute system memory map, and therefore simplifying their design and enhancing reusability.

C. APB Bus Blocks

The APB bus contains a single master, the APB bridge (an ASB slave in turn) that connects to the ASB bus all peripheral blocks, which act as slaves. Data access is controlled by select and strobe signals (i.e., no clock, and thereby reducing power) generated by the bridge (address sub-decoding is centralised as in the ASB).

The keyboard controller, with debouncing logic, may handle an (up to) 8x8 passive matrix keyboard. The display controller interfaces standard LCD devices. It contains an internal 256x10bits FIFO to allow sending bursts of characters or commands to the display.

Up to 64 digital signals (32 outputs, 16 inputs and 16 bidirectional) may be controlled with the I/Os interface, with a memory mapped access by bit, byte or half word.

For external communications, four USARTs for full duplex serial port accesses and a master/slave I²C interface (9) for bidirectional data transfers up to 400 Kbit/s between the processor and an I²C bus are provided. The Intel interface allows the connection of up to 4 Intel bus-compatible peripherals with a 256 address space each.

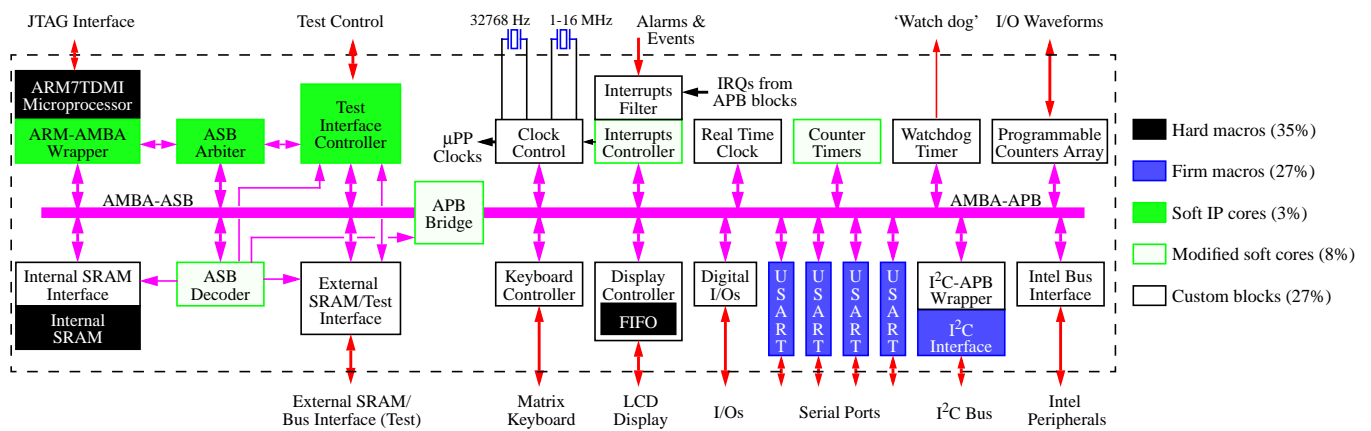


Figure 1: μ PP Circuit Architecture and blocks

The Programmable Counters Array (PCA) contains 8 linkable counters, 8 bits each, that can be programmed to generate output waveforms or to check input waveforms.

Three programmable free running/periodic 16 bit counter timers with selectable clock and prescaling may count for up to 2 s. The watchdog is also a free running counter, clocked at 32 KHz, that generates an external signal that may be connected to the μ PP circuit reset. The real time clock, with alarm and Y2K compliant calendar, uses asynchronous counters to minimise power consumption.

The interrupts controller and filter manages 17 internal and up to 10 external interrupts, with programmable priority level, polarity and enabling.

D. Clock generation

The clock control block (Fig. 2) was designed to cope with the stringent supply restrictions in the pay phone system: a maximum of 1,35 mW power consumption during hang up, and 99 mW when picked up for the whole system.

An on chip 16 MHz low power oscillator, generates a system clock in the 1 to 16 MHz range, with a worst case consumption (100% program activity at 16 MHz) of 300 mW. The output of the oscillator can also be divided to lower frequencies, allowing the microprocessor to run at reduced speed in non critical timing phases. The ARM CPU can be put in stand-by (pause) mode, disabling the processor clock, but minimum consumption (only 2.5 mW at 16 MHz) is obtained by programming the system in halt mode, which disables the clock for all blocks except some peripherals driven by the 32 KHz oscillator (real time clock, e.g.). In both cases the system is reactivated by an interrupt request from one of the active peripherals, or any external interrupt line. Lowest consumption (just 0.6 mW) is obtained by disabling the 16 MHz oscillator and letting the microcontroller run at 32 KHz.

Halt and subclock power is dominated by internal memory leakage current, which was the limiting factor to determine the maximum on chip memory size.

Design Methodology and Tools

In a first step all the needed peripherals and functionality in the present phone system board were identified, and a specification developed, containing both the hardware functionality and interfaces, and the programmer's model. This latter model, was based on the ARM's reference peripherals specifications (6), in which all system elements are memory-mapped. To increase reuse and software portability, the exact

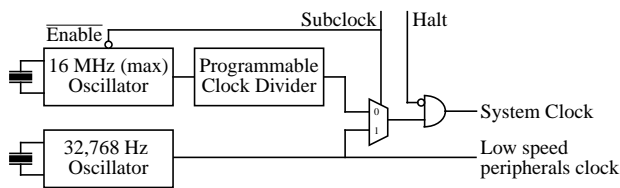


Figure 2: Clock generation and control

definition of the memory map was not fixed initially. However, the registers inside the hardware blocks, have a fixed offset from the block's base address. This allows an independent development of the hardware and software design activities from the initial design phases (see Fig. 3).

A. Hardware design flow

The HW design followed a 'desktop' approach (4) using a standard ASIC design flow based on the use of the VHDL language and logic synthesis. For custom modules and modified soft IP blocks, a VHDL description was written and simulated using the AMBA compliance testbench suites (provided by ARM). The most critical parts of the circuit, in terms of timing constraints (for example clock gating and multiplexing) were described in a nearly structural, but technology independent way, using Synopsys Generic Technology (GTECH) library components. Less critical blocks have higher level RTL descriptions.

Soft IPs and custom blocks synthesis was a quite straightforward task thanks to the scripts available in ARM's Micro-pack. The clock generation block required a specially careful static timing analysis, due to its highly asynchronous nature, to check it met the constraints and had no glitches.

Simulations and regression tests were performed along the design flow for each block and for the whole circuit, with the aid of the AMBA TIC module. Some simple programs were run by simulating, in VHDL, a test system with the μ PP model and external memories containing the ARM code.

The very low power requirements of the circuit forced a latch based design of most blocks, that prevented the use of scan based test techniques for foundry tests. Except for the ARM core and I²C modules that do have scan chains, all the peripheral modules have special program registers to facilitate its ad-hoc testing using the AMBA test methodology (i.e. using the TIC). For the memories, BIST has been used.

B. Software design flow

The software development activities were run in parallel with the hardware design. First, a software analysis and

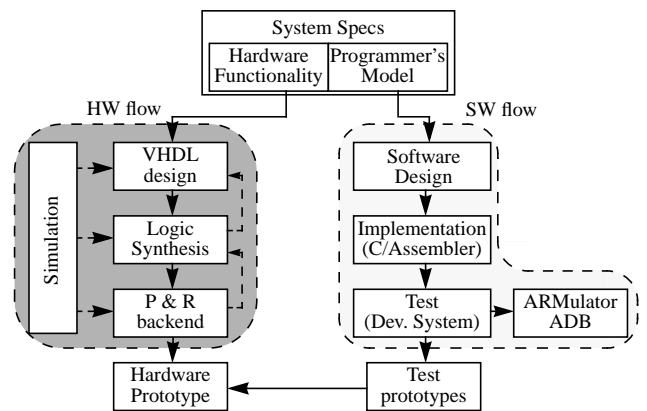


Figure 3: μ PP design flow

design was done, based on the existing pay phone system software. From this analysis resulted a software architecture identifying the software modules to be coded in C, and the hardware drivers for the μ PP peripherals, to be coded in assembler and/or C. This task is currently under development. The compiled software modules are tested using the *ARMulator* software emulator and a hardware prototyping board (the ARM Development Board, ADB).

Design Management

Though with the usage of IP soft blocks designers are released from a lot of burden, they have to analyse, simulate and debug, synthesise, maintain, and even modify blocks they did not design, while ensuring at all times, the whole design consistency and quality. This involves a lot of engineering and reverse engineering work, which demands new design automation tools to aid in the process (3).

For maintenance, analysis, debugging, quality assurance and design complexity management in the μ PP design, the VHDL-ICE (1) and other TOMI Tools (8) have been used. VHDL-ICE is an Integrated Common Environment suitable for design reuse and management. It has a client/server architecture that supports and interoperates across multiple platforms (Win-NT, Unix) over a distributed TCP/IP network.

The VHDL-ICE server manages a centralised design database containing all derived information required to control the different build processes of a VHDL design. Several clients may concurrently connect to the server: the design manager, the navigators and the assessment tools. Any other third party tool may access design data through a provided set of API functions or using import/export facilities.

The design manager allows multiple, flexible and reproducible workspace areas related to different organization units, such as design projects or teamworks. Its role based access control (RBAC) and version and configuration management capabilities coordinate the work done by different designers ensuring data integrity. The navigation tools (design units, hierarchy and simulation model navigators) provide an useful and powerful way to analyse and debug both IP cores and user designed modules, simplifying design reuse. The assessment tools (VHDL style checker and debugger) enhance code quality maintainability and homogeneity.

Results and Conclusions

The circuit has been implemented using ATMEL ecat05 standard cells technology ($0.5 \mu\text{m}$ CMOS). It contains 68K equivalent gates (nand2), 256 Kbit of static RAM, 2.5 Kbit of dual port static RAM, and an ATMEL's application specific hard cell implementing the ARM7TDMI microprocessor. The circuit area is $72.25 \mu\text{m}^2$ and test prototypes are packaged in a 304 CQFP with only 234 functional pins (production will be packaged in a 256 PQFP).

The μ PP circuit is a real example of how design and IP reuse enable fast and efficient System On Chip integration,

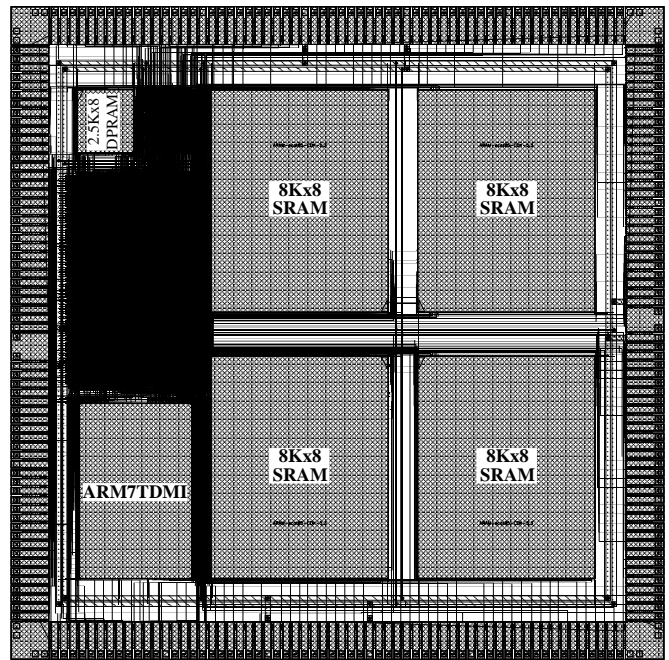


Figure 4: μ PP circuit layout

reducing costs and increasing reliability and performance. The AMBA bus strategy and ARM software development kit and development board, allow the software and hardware design tasks to be performed in parallel, thereby reducing global design time and making a functional system to be ready as soon as the prototypes are available. An estimated 30% reduction in total design time has been obtained thanks to IP design reuse, new management tools, and co-design.

Acknowledgments

This work has been partially funded by the European Union within the TOMI Esprit project 20724.

References

- (1) S. Olcoz et al. "Reusing VHDL soft-cores by means of using appropriate workspace management and navigation tools". Forum on Design Languages 1998 (FDL'98). Proceedings Vol. I. Sept. 6-11, 1998. Lausanne (Switzerland), pp. 201-207.
- (2) E. González, F. Martín. "Public telecommunications". Comunicaciones de Telefónica I+D, Vol. 9, Nr. 1, Jan.-Jul. 1998, pp. 117-122.
- (3) R. K. Gupta, Y. Zorian. "Introducing core-based system design". IEEE Design & Test of Computers, Oct.-Dec. 1997, pp. 15-25.
- (4) A. M. Rincon, C. Cherichetti, J. A. Monzel, D. R. Stauffer, M. T. Trick. "Core design and system-on-a-chip integration". IEEE Design & Test of Computers, Oct.-Dec. 1997, pp. 26-35.
- (5) F. S. Eory. "A core-based system-to-silicon design methodology". IEEE Design & Test of Computers, Oct.-Dec. 1997, pp. 36-41.
- (6) S. Furber. "ARM system architecture". Addison Wesley, 1996.
- (7) T. D. Burd, R. W. Brodersen. "Processor design for portable systems". Journal of VLSI Signal Processing, Vol 13, Nr. 2/3, Aug./Sept. 1996, pp. 203-222.
- (8) "TOMI (Tools for the OMI) technical annex". Esprit project 20724, Sept. 1995.
- (9) "The I2C-bus and how to use it (including specifications)". Apr. 1995. Philips Semiconductors.