Microcontrollers in Education: A Case-Study

Andrés Santos, Eduardo Boemo, Julio Faura and Juan Meneses

INTRODUCTION

In this paper are presented the results of an experimental undergraduate laboratory on microcontrollers at third year of the School of Telecommunication Engineering of Madrid, Spain. The objectives been to evaluate the success of a hands-on course on microprocessors that emphasizes a system level perspective including aspects of hardware as well as software. Another additional goal has been to reduce the utilization of the school's laboratories, already quite overcrowded.

Traditionally the lectures on microprocessors are complemented with experiences at laboratories [1]-[2]. The equipment utilized is usually based on a commercial development board that includes a microprocessor, I/O ports, timers, etc, as well as some tools for assembling and debugging the code. Although the development systems allow the students to work with a real circuit and to create programs in assembler language, there are still serious limitations:

- The experiments with hardware are limited to the addition of simple elements (displays, A/D y D/A converters, matrix keyboards, etc.) connected through peripheral ports.

- When the number of student is high, it is not possible to provide them with more adequate instruments like logic analyzer or microprocessor emulators.

- The time assigned to laboratories, usually limited to a semester, reduces the possibility of including complex hardware construction.

- The cost of repairing or replacing commercial microprocessor development boards discards any practical exercise that involves the direct access to the system buses (A solution to this obstacle is described in [3]: an educational μP board specially designed to make the failures detection easy).

As a consequence, although the microprocessor exercises include some development of hardware, they are basically programming exercises. It is not possible face the student with the challenge designing and building a complete microprocessor system with a CPU, the EPROM booting routines, the connection of peripheral circuits, memory banks, communication interfaces, etc. Thus, they can not experiment the common difficulties of any actual electronic project: trade-offs determination among different solutions; the understanding of tools, manuals and databooks; the selection and obtention of sample chips; the isolation and debugging of errors spread in both hardware and software; the pressure of deadline; and the interrelation with economic and business aspects.

Some actions can contribute to make the above described restriction insignificant. These are: to improve the laboratory instruments quality and quantity, to hire a team of skilled and fast-response technicians in charge of lab's maintenance, and to increase of the number of professors at the lab. Now, a realistic question is: How to solve these problems without requiring additional budget and manpower? (a complete diagnostic of universal problems about undergraduate engineering laboratories can be found in [4])

The emergence of low cost powerful microcontrollers constitutes a possible solution to this question. Although they were conceived for industrial applications, their excellent characteristics make them an ideal tool for teaching low-speed digital electronic techniques.

A microcontroller offers for a price extremely low (among US $10 and $30) a complete system that includes an 8-bit microprocessor, memory (RAM, EPROM, an even EEPROM), ports, counters-timers, A/D converter, and serial communication, altogether in just one chip. Using this "component" with few other commercial chips, the students can build a whole microprocessor-based system in a reasonable time. Therefore, they can get training on the process of conceiving and building a complete project (hardware and software) using an contemporary product.

In terms of educational organization, the advantages are notable. Each student can build the circuit even at home; it is not necessary to buy any development system; the minimal set of tools required are just a personal computer, a multimeter, and a d.c. power supply (or a battery). As a consequence, the time spent in the laboratory can be limited to the use of an oscilloscope just to find out start-up problems during the assembly stage.

Finally, there is an additional consequence: the possibility of producing small low-cost development systems constitutes a way, until today unimaginable, of offering a practical course on microprocessors in a distance-education plan.

THE 68HC11 MICROCONTROLLER FAMILY

Today there is a wide offer of microcontrollers of 4, 8 or 16/32 bits. The principal manufacturers (Motorola, Mitsubishi, Intel, NEC, Philips, Hitachi, Matsushita, SGS Thompson, National and Texas) have one or several families of microcontrollers in the market. For this laboratory the 68HC11, an 8-bit family of Motorola, was chosen. The reasons for this selection are multiples:

- It has a traditional architecture (with linear memory), easy to learn and adequate for introductory courses on microprocessors.

- It has an EEPROM memory (writable-non volatile). This allows to store programs without requirement of an EPROM programmer.

- It has several flexible timers/counters, many I/O lines, and analog interfaces, very useful to apply in a great quantity of problems.

- The equipment needed to start working with this chip is reduced to a personal computer with an RS-232-C serial port. It is not necessary to acquire any development system. This was one of the most important reasons for choosing this microcontroller.

- The 68HC11 is widely used and has a good number of available development tools (both freeware and commercial).

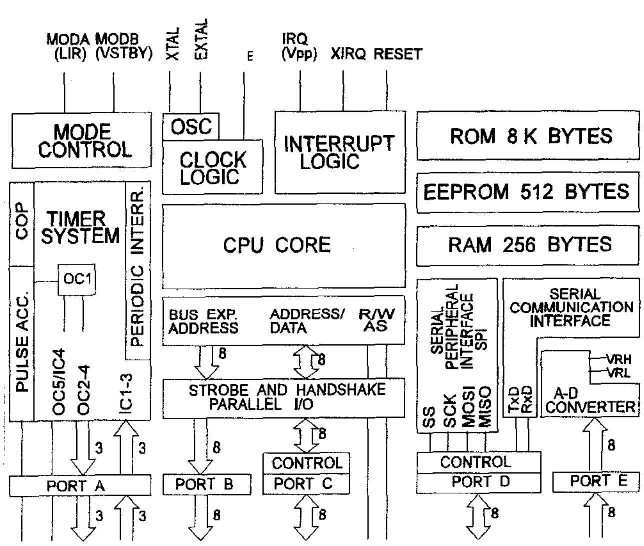

Fig.1: 68HC11 internal architecture

Among the many available versions of this microcontroller, two were selected: a cheap one without EPROM memory (68HC11A1), and another a little more expensive that includes 12 Kbytes of EPROM (68HC711E9) that allow to store more complex programs.

Figure 1 shows the internal architecture of the 68HC11 chip. The main components are [5]:

- CPU: 8 bit processor (an improved version of 6801)

- EPROM: the 68HC711E9 version has 12 Kbytes.

- E^2PROM: 512 bytes of non-volatile memory.

- RAM: 256 bytes in 6811A1, 512 in 6811E9.

- I/O ports: up to 38 independent lines.

- Counters/Time accumulators and real time interruption.

- Serial Interface: two ports, one RS-232-C compatible and another to connect several peripheral circuits.

- A/D Converters: 8-bit converter with an analog 8-input multiplexer.

Figure 2 shows the minimum circuit to use the microcontroller. In addition to the 68HC11, a RS-232 converter (a MAX232 for example), a reduced number of components (resistances and capacitors), and an 8 MHz crystal are necessary to obtain the RS-232-C voltage levels. The total cost is under US $50.

The minimum software needed is an assembler, and a debugging freeware program of Motorola called PCBUG11. It allows to send data and programs to the 68HC11 through the PC serial line, as well as to examine the registers, to execute programs (with break points if needed). It is also able to write in the EEPROM and EPROM memories.

PCBUG11 makes use of one of the operation modes of the 68HC11: when the processor starts at power-up in bootstrap mode, it executes a small routine (stored in ROM) that is in charge of receiving a program through the serial line and storing it in microcontroller internal memory. Using this capability, PCBUG11 sends a small routine (a talker) that allows several operations: to load application programs, to read and to write the internal memories and ports, to execute programs, etc. As a consequence, it is neither necessary to have a monitor program stored in EPROM or ROM nor a development system.

Additionally, there also exist many freeware programs: assemblers, simulators, BASIC interpreters, C compilers, etc.

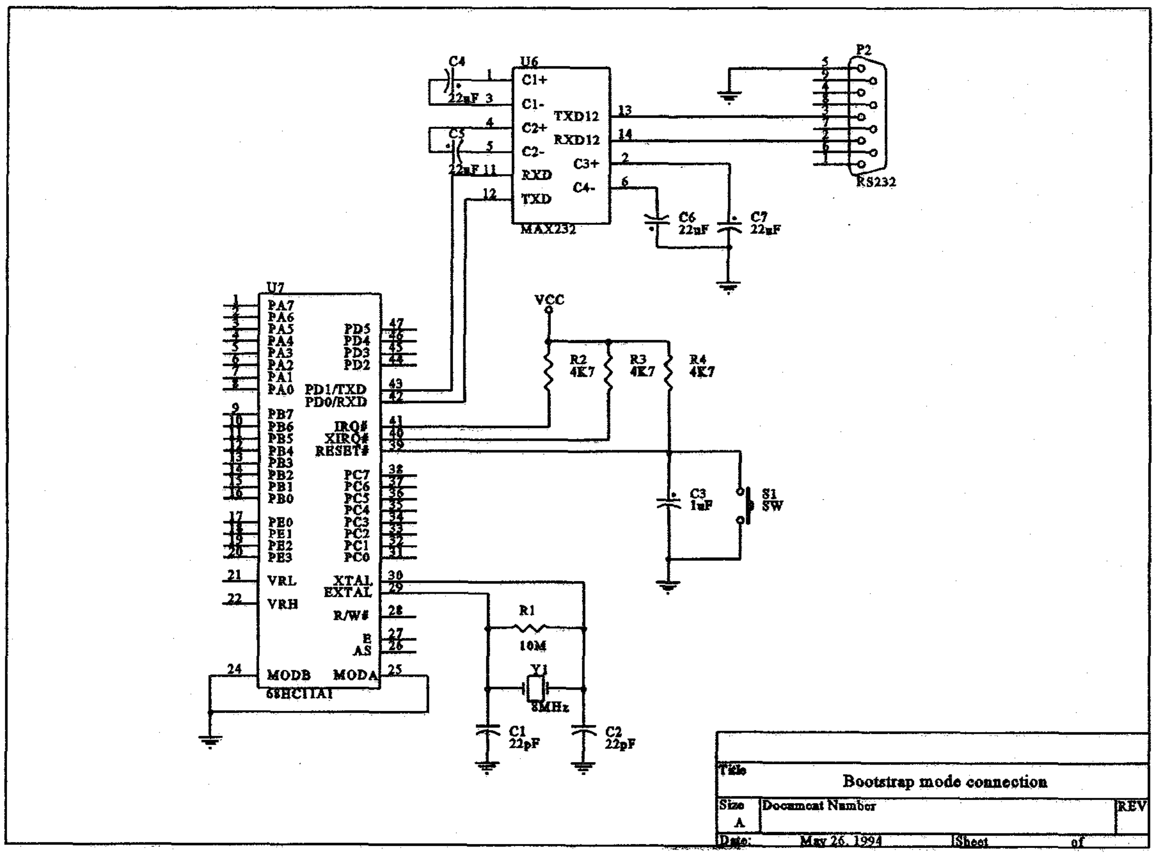

Fig. 2: Minimun configuration circuit for bootstrap mode.

There are lot of FTP sites in the Internet and BBSs that contain public domain software, sample source codes for several applications and documentation.

Finally, another interesting possibility during the initiation stage is the evaluation kit 68HC11EVBU, that includes a small board with a 68HC11 and an area for wrapping; a diskette with the basic programs: assembler, PCBUG11 and user manuals. Some examples about this board utilization in the universities are collected annually by the manufacturer in a publication [6]-[7].

68HC11 VS. 68000 IN EDUCATION

Table I shows the main aspects to evaluate the appropriateness of the 68HC11 as microprocessor teaching tool. It is a comparison between the 68HC11 and the 68000-based development system utilized in our laboratory, that includes an CPU, two VIAs 6522 and a DUART 68681.

The resources of the 68HC11 allow, without any doubt, to achieve a good number of practices including hardware and software. Even though the 68000 CPU is more powerful, the 68HC11 is superior in communication capabilities. In an electronics laboratory, where one of the goals is that students design and build their own systems, the advantages of the 68HC11 are evident.

There are however some topics that would be more adequate to teach with a processor like 68000: direct memory access (DMA), dynamic memories or multiprocessing. Nevertheless, these topics can be considered as more advanced and in fact, are not studied more than superficially (without laboratory practices) in the basic course.

EXPERIMENTS AND RESULTS

In order to verify the potential advantages that microcontrollers can have in education, as well as to detect possible start-up problems, during the 1993/94 course it was offered to 40 students the possibility of building in their own circuits based on a 68HC11 instead use the 68000-based development system.

They received a short introductory lesson (about 4 hours) and copies of the databooks.

	68HC11	68000-based system (TM-683)
Instruction Set	145	56 (122)
Addressing Modes	6	14
Addressing Space	2^{16}	2^{24}
Interruptions	2 external lines 32 vectors	3 external lines 256 vectors
I/O Ports	38	16
Counters/Timers	3 inputs 4 outputs 1 I/O	2 inputs 2 outputs
Serial channels	2	2
Analog inputs	8 channels	External
Possibility of external memories?	Yes	Yes, if the access to the buses is allowed.
Possibility to work at home	Yes	No
Cost of basic equipment	Very small (approx $50)	Medium (approx. $1,000)

Table I: 68HC11 vs. 68000 in Education

The following set of exercises suitable to be solved in about two months were proposed:

- Speed control of a toy's d.c. motor using pulse width modulation (PWM): The input is a frequency signal obtained from the well-known holed wheel with photodiode and phototransistor. The output is a variable duty-cycle periodical signal that is used for exciting the motor coil.

- Digital Voltmeter: It allows to use the A/D converter. The output is presented in a 7-segment display. Additionally, the conjunction of A/D converters and PWM allows the digital control of great variety of systems.

- Time-of-Date clock: The student must divide an input signal in order to obtain a one-second "tic". The subsequent outputs are presented in a 7-segment display.

- Pulse generators with programmable duty-cycle: This exercise is a subset of the motor control project.

- Electronic Piano: inputs are derived by polling a keyboard. The outputs are square waves of musical notes. Another similar exercise is the design of a programmable waveform generator.

The proposed problems are mainly typical real-time applications, that emphasize the use of interruptions, timers, and other internal components of the chip. Some of the exercises are previously solved in the first part of the course, a laboratory based on FPGAs [8]. Thus, the students can see the main differences in speed, cost, and design time of different but complementary technologies. For example, a time-of-day clock, a 32-bit multiplier, counters, or frequency meters admit both FPGAs and microcontroller implementation.

Another goal of the laboratory has been to encourage to students to come up with projects related to their own interest. In this way students feel quite more stimulated to work on what they really like or need for personal use. Some examples of these miscellaneous circuits are a timer for a photograph laboratory, a guitar tuner, a control for model trains, a system of automatic irrigation, an one word speech recognition system for computer login password, a small data acquisition system, a video game for the oscilloscope working in the XY mode, an infrared remote control unit, a bicycle speed meter, etc.

Although the plan was that each student buys his/her own components, for this first experiment a centralized purchase was carried out. Once the chips were available, each student built his/her own circuits at home using some tools for wire-wrapping supplied by the school. Most of the circuits did not work after mounting and needed some debugging. The use of an oscilloscope at lab was often needed but more sophisticated instruments like logical state analyzers or microprocessor emulator were not employed in any case.

The principal problem was, surprisingly, the purchase of the chips. For cost and availability reasons, for quantities of about 50 units, it is prudent to make an agreement with a supplier in advance. An important conclusion is that the teacher must encourage the use of a single type of microcontroller: an excessive variety of chips increases the quantity of problems to solve (incompatible software versions, IC's sockets not available, uncertain delivery periods, reduced inter-changeability, bigger stock, etc.).

Another problem detected was the fact students tend to mount all the components in the board and then try to make the whole circuit run. It is essential to instruct them in the use of decomposition and sequential mounting strategies, and the habit of including testability-oriented elements in the design (a few *leds*, switches and connectors can be enough).

The final balance has shown highly positive results. As usual in hands-on laboratories, students attitude has been very enthusiastic; they have built very interesting circuits and today they are owners of a small microprocessor development system that they can reuse for other designs. Finally, they have obtained more knowledge in comparation with the normal course based on a full pre-designed and builded commercial equipment to which little could be added.

Today we are analyzing the incorporation of microcontroller training for all the students of third year, reserving the systems based on the 68000 for illustrating the students in the particularities of 32-bit microprocessors.

ACKNOWLEDGEMENTS

The authors wish to express their gratefulness to Arsenio Vilallonga for his help solving students' problems, and to the Moya brothers for their assistance with the Internet exploration. Finally, many thanks to Sergio Ocaña and José Pérez of Selco, and Enrique Ruíz of Motorola Spain for their continuous support given to this educational experiment.

REFERENCES

[1] D. Hall, "Adapting Curriculum Materials for Different Course Sequences", *IEEE Micro*, pp.34-37. February 1991. (Special issue on microprocessor education).

[2] R. Chassaing, "A Course in Microprocessors Based on the 16/32 Bit 68000 microprocessor", *IEEE Trans. on Education*, vol.E-30, No.3, pp.194-197, August 1987.

[3] C. Sanz, "Problematic of Practical Teaching of Design of Microprocessor-based System. Some Solutions" (in spanish), *Proc. Conf. on Application of Technologies in Electronic Education (TAEE)*, pp.379-388. Madrid: U.P.M. Press, 1994.

[4] E.W. Ernst, "A New Role for the Undergraduate Engineering Laboratories", *IEEE Trans. on Education*, vol.E-26, No.2, pp.49-51, May 1983.

[5] *"HC11 Reference Manual"*. Motorola Inc, 1993.

[6] *"Motorola University Design Contest"*. Motorola Inc, 1992.

[7] *"68HC11 Application Guidebook"*. Motorola Inc, 1993.

[8] E. Boemo and J. Meneses, "Learning VLSI using Programmable Logic", *Proc. 1993 Frontiers in Education*, pp.422-425. New York: IEEE Press 1993.

*****

Andrés Santos, Ph.D. in Telecommunication Eng. Since 1985 he is Associate Professor at the UPM (Universidad Politécnica de Madrid - Spain). He is in charge of microprocessors teaching at the Electronic Engineering Dept. His areas of interes are also related with speech processing and medical images.

Eduardo I. Boemo received the E.E. degree from the UNMdP (Universidad Nacional de Mar del Plata, Argentina). Currently he is Ph.D. candidate at the UPM. From 1985 to 1987 he was Assistant Professor of the Dept. of Physics (UNMdP). Since 1989 he is Associate Professor at the School of Telecommunication (UPM) where he is in charge on Undergraduate Laboratories at the Electronic Engineering Dept. As consultant, he has participated on about a dozen of industrial project on the area of VLSI and Communication. He has written 20 published papers, 3 technical books and has a patent on Computer Security. His current research interests include VLSI Array Processing, Computer Arithmetics, CAD Tools and E.E. Education.

Julio Faura: received the E.E. degree at the UPM. Currently he is a graduate student at the Electronic Engineering Dept. In 1992 he won the Motorola University Design Contest (Zone Europe).

Juan M. Meneses received the Engineer degree and the Ph.D. degree both in Telecommunications, in 1977 and 1985 respectively by the Technical University of Madrid. At present the is Professor at the Electronics engineering Department of this University. This research activities are focused in: High speed architectures for implementing digital signal processing algorithms (speech and images applications) in real time and high level synthesis methodologies for electronic design and its implementation. During these years he has been director of a group of eleven scientists. He has written more 35 published papers and four technical books in electronics engineering.