Learning Embedded Systems with Multi-Microcontroller Evaluation Tool

R.S.K. Selvakumar[1], K.H. Ghazali[2], N.M.K. Nik Yusoff[3], H.AbdulAziz[4]

Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang,

26300 UMP, Kuantan, Pahang, Malaysia; Phone: +6012-9631397

E-mail: inderjit_1791@yahoo.com[1], kamarul@ump.edu.my[2], nik@ump.edu.my[3] ,
hazizulden@ump.edu.my[4]

Corresponding Author: inderjit_1791@yahoo.com[1]

Abstract - **Embedded systems have an everyday presence and direct impact in every day's lives. Therefore, learning institutions are continuously improving their courses in microcontroller and embedded system programming. Although the diversity of curriculums, the availability of learning tools, where the student can practice and improve their skills, is a key factor to the success of the learning process. The platform developed and presented in this paper results from author's experience in teaching embedded systems. From the analysis of teaching/learning needs, a learning environment based on the Freescale HC11, MCS51, and Microchip PIC 16 Series families was designed. This tool can be expanded by modules and adjusted, at a specific time, to student's real needs. All modules can be interconnected by an IDE bus, allowing expanding the capabilities of the platform. The developed modules allow the practice of subjects related with digital IO, analogue interface, user interface, wireless communications, and energy management and conservation.**

Keywords - **Embedded system, learning, laboratory, programming, microcontroller.**

I. INTRODUCTION

Embedded systems courses are today included in most of the graduate and undergraduate curriculums of Electric & Electronics engineering and Computer Science from all over the world [1-5]. Increasing the impact of microcontrollers in our daily lives justifies this option. These devices can be found either in very simple products, like toasters or toys, or in highly technological products, as for examples, cellular phones, camcorders or cars. The complexity of an embedded system can change from product to product, depending on the task that they must perform. Therefore, embedded system designers must have knowledge in different areas that, sometimes, are separately handled. Hardware project requires knowledge related with digital and/or analogue electronics and at the same time, with electromagnetic compatibility issues that cannot be forgotten in high frequency operation or in products that must work in very restrictive environments as the ones found in hospitals.

In turn, the designer must project the software, required by the hardware, allowing it to work as expected. Subjects like operating systems, real time systems, fixed and floating point arithmetic, digital signal processing, and programming languages as assembly, C/C++ or Java are of major relevance to the development of embedded systems. As stated previously, the skills that an embedded systems designer must hold are, it highly complex and spread [6-7].

If beyond the previously knowledge, the student need to learn how to work with a complex development kit, then he/she probably will fail. Even if the student is adequately prepared in the required subjects, the time taken to obtain visible results is, sometime, responsible for student demotivation and consequent failure. This paper presents a platform designed from scratch to support embedded systems teaching.

With the improvement of student's skills, tool complexity can also be increased, adding more functionality to it. In its bare configuration, the student can practice only with the digital IO. Functionalities normally found in development kits are not present, avoiding this way student distraction. The complexity of the associated modules can be increased with student progress. Functionalities continuity between modules is granted by an IDE bus. The developed communication protocol allows data exchange between all modules.

In this paper, we are going to propose such a systematic approach for hardware implementation and discuss how it could be used successfully in embedded system education of this evaluation tool in class. This paper presents the importance of microcontrollers course in education syllabus and the design of multi microcontroller evaluation tool that is based on 8-bit microcontroller Philips P89C51RD2HBA, Microchip® PIC 16F877 and Freescale® MC68HC11. It begins with the discussion of the hardware implementation of the system. It proceeds then with software implementation of the system to be interfaced with the PC creating a user-

friendly environment, followed by a discussion. Finally, a conclusion is given in section 6.

II. SYSTEM ARCHITECTURE

Research and education activities at Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang involve embedded systems design and development. There are currently running around 30 research projects and student courses ranging from semester 3 to semester 7 [8]. Every year, there are Embedded Control Technology course available for third year diploma and degree and also for students from our faculty who taking final year projects [9]. All these activities require different microcontroller types, each with its own development board, but structure of such boards is very similar, only microcontroller and its associated circuitry being different [10].

The aim of presented work is to design a development board for embedded systems suitable for different microcontrollers that are connected through several types of application board [11]. Experience gained in last research projects (MCS51 Educational Development System [10] and MC68HC11A1 Microcontroller Laboratory Hardware Platform – MINI11 [12]) lead to a new design – the evaluation tool for 8 bit microcontrollers, Freescale HC11, MCS51, and Microchip PIC 16 Series.
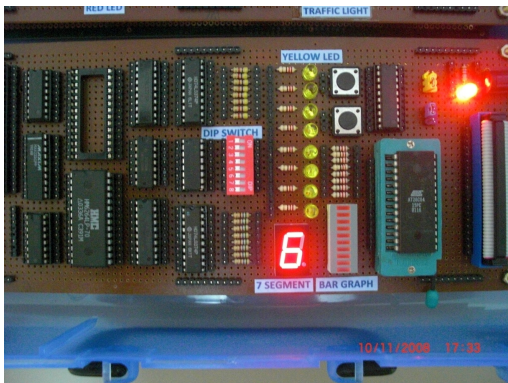

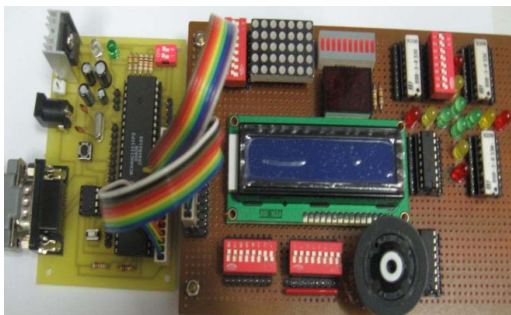Fig 1: MCS51 Educational Development System

Fig 2: MC68HC11A1 Microcontroller Laboratory Hardware Platform – MINI11

The idea of many available peripherals from first project was combined with flexibility of application board plug in second design [10]. The resulted system is expected to have 2 qualities: cheap (no more than RM 1000 per mainboard plus one application board) and flexible (open structure for application board, future expandable for easy to handle new types of application devices).

Figure 3 shows the system block diagram for evaluation tool. The evaluation tool has a similar layout with MCS51 Educational Development System. Other two more microcontrollers and various associates component were added. The boards contain a Freescale MC68HC11E1 device, Philips P89C51RD2HBA device and Microchip PIC16F877A device. Each board contain microcontroller, clock circuit with crystal, reset circuit and programming connector. Only flash-type microcontrollers were chosen because the goal is a development system so software changes are often.
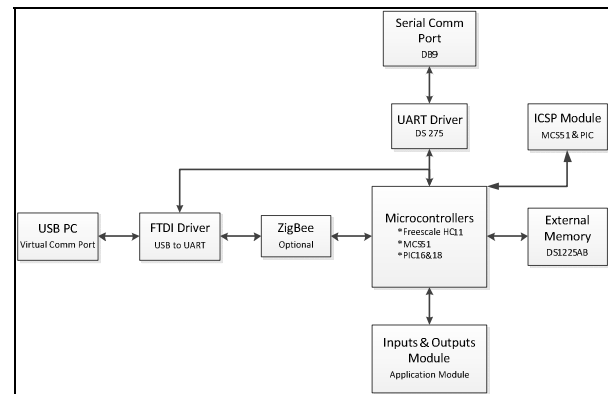

Fig 3: System Block Diagram

The evaluation tool should contain all required peripherals for standard embedded systems applications as shown in figure 4; 5 Volt power supply, RS232 interfaces, open drain power outputs, analogue buffered inputs and outputs, LEDs, input and output interface. Each application board contains remaining parts – obviously the microcontroller, its external memory circuits, as well as its input and output interface.

These voltages are available on different pins of the connector between boards, so no switch is required for voltage selection. On board voltage regulators are able to supply 1.5A on 5V rail, more than enough for a modern microcontroller. Programming connector is dependent on microcontroller family. However, all the microcontroller system pre-programmed with monitor program which enable user to load and execute the file using ZigBee interfacing. Every microcontroller has its own monitor program pre-program in internal and external memory as shown in figure 5.
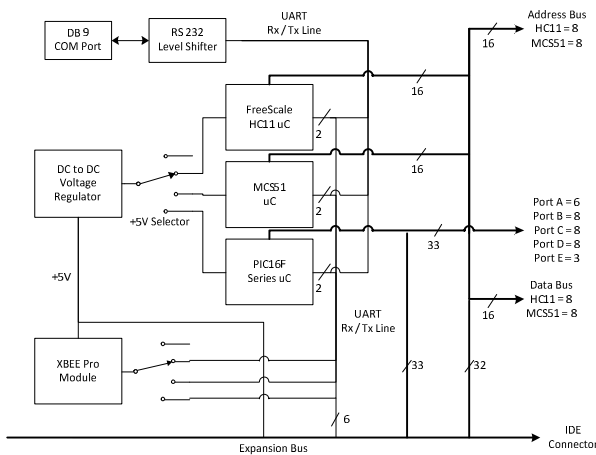
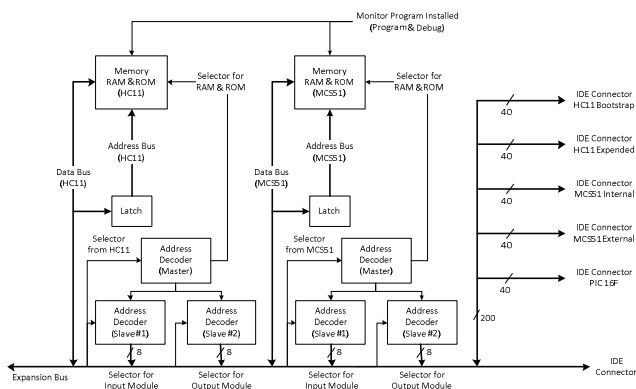Fig 4: Structure of the Basic interface and power module.

Fig 5: Structure of the Memory interface.

Final circuitry of the evaluation tool is designed for interfacing to RS232 serial ports is provided by EIA232 circuit utilizing Dallas Semiconductor's DS275 while power circuit regulates supplied voltage to 5Vdc for TTL application. I/O ports of microcontrollers are connected to connector sockets for ease of interfacing with various I/O devices. Since the development board is designed for on-chip mode, all ports are available to be explored. The prototype of the evaluation tool is shown in figure 6; architecture of application boards is very simple, mainly microcontroller and some small parts for external memory interfacing circuits.
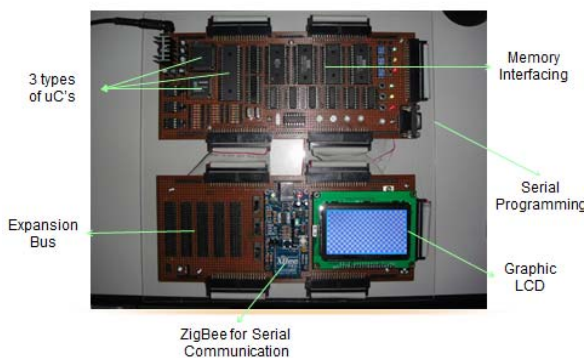

Fig 6: Prototype of the system with GLCD interfacing

III.   THE SOFTWARE IMPLEMENTATION

Development of evaluation tool software is divided into two stages; the monitor program development and the IDE program development. Monitor program controls the entire interfacing between a computer and the evaluation tool. Written in assembly language and stored in ROM, monitor program is used to perform object file loading and executing without the hassle of burn-erase-burn method usually associated with EPROM programming. This program will be stored into microcontroller internal EEPROM.

The conceptualization of the monitor program algorithm (used in this evaluation tool) is shown in the Figure 7. The basic function of a monitor program is to read user's input from keyboard. Based on the input, it determines which procedure, either upload S19 or HEX file or execute program, to be executed. Once selected, the instruction is processed by sending the information to the development board for execution. Status of execution is feedback to user on a computer screen.
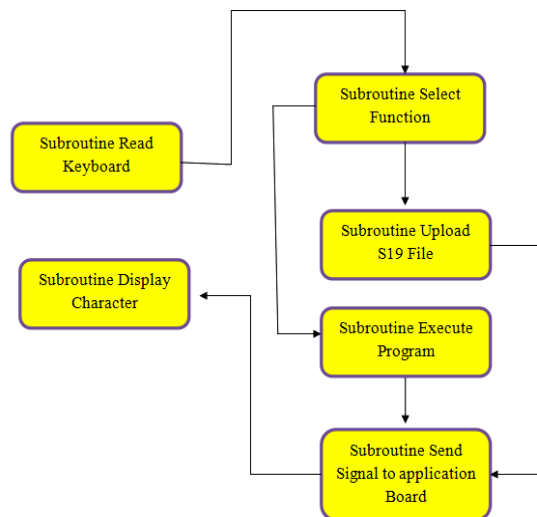

Fig 7: Concept Diagram of Monitor Software

Usually running a monitor program requires the use of communication program such as Hyper-Terminal. Likewise, developing application software for microcontroller requires a text editor and an assembler to write and assemble the code respectively. An IDE program integrating programs such as communication program, cross-assembler and text editor is developed to create user-friendly development environment. This new software is developed to allow users to perform all development activities in the ever familiar window working environment without needing to exit any program. Instead of entering instruction on command line, IDE user interface allows execution of instruction by clicking the appropriate buttons. Figure 8 shows a screenshot of the IDE program.
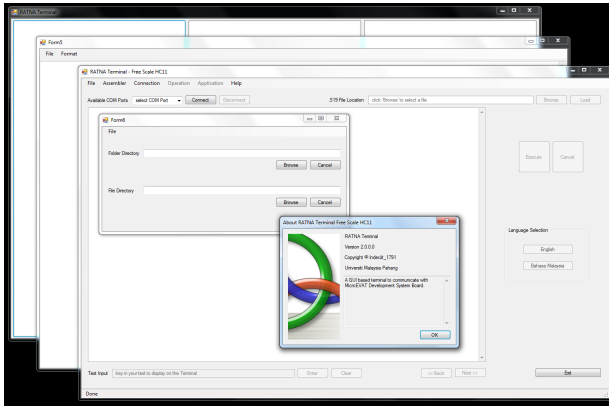
Fig 8: Screenshot of IDE program

IV. DISCUSSION

Likewise, a monitor program will be developed, assembled, and downloaded into the EEPROM or NVRAM. Several tests of mini operating system have been tested several times and it been proven that the system is reliable and sufficiently stable. It shows that the monitor program is able to communicate with the computer and capable of performing several commands issued from the host.

A survey on student's understanding in embedded systems was conducted to detect the level of understanding in this course and the feedback from the students if there are develop in-house microcontroller educational development system based on the MCS51, HC11, and PIC 16 series microcontroller were developed. So far, 54 responses from the students in FKEE who replied the surveys and preliminary results are encouraging. The majority of the students found the evaluation tool system will helpful in understanding the knowledge in embedded system.

The evaluation tool system will also helped them to better understand how the microcontrollers worked and how hardware and software worked together if the evaluation tool are build. The instructors teaching this lab noticed that the need of this evaluation tool system could make a significant improvement in students' performance and understanding of the lab. For more detail on this evaluation tool system's survey, please log on into this webpage, "http://www.kwiksurveys.com/online-survey.php? surveyID=BIEIM_5d8a6dc9".

V. RESULT AND TESTING

After designing the evaluation tool, the mainboard and each application board were tested by running simple applications for each of the microcontroller. These programs were elaborated with monitor program of each microcontroller. All boards are functional as well as all devices from mainboard. Testing of different microcontroller configurations was helped by the existence of a common set of software modules written in C language [13]. These modules have a high level of portability and can be easily adapted to new microcontroller families. From the costs point of view, the solution is cheap – about RM 458 for mainboard and RM 100 – RM 200 for each daughter board, depending on microcontroller.

The flexibility of this solution made it very attractive for replacing other more expensive development systems. After a survey on electronic component distributors, development systems with similar peripherals cost more than RM 5000 for all three families of microcontrollers and offer less functionality. Another advantage of presented solution become important in education activities – a flash microcontroller has a limited number of erase cycles for its program memory, catalogue data mentioning usually 10'000 cycles. Unfortunately this number is valid in some special temperature and voltage conditions that can be easily exceeded in an educational environment. If microcontroller becomes damaged, replacing the device is a cheap option compared with replacement of the entire board like in case of single board solution.

During development and testing stages a number of improvement ideas occurred. First, a USB connection is important for working with modern laptops that have no more RS232 interfaces. Second, connector socket on the motherboard is too much for normal applications and consume too much board space. Third, a buzzer or small speaker is a good option for generating sounds and alarms. Fourth, more high-range devices were required for application module, like high voltage devices being considered. All these new ideas are will be implementing in this evaluation tool system and results are expected in end of April this year 2011. Presented work is a part of a research project of Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang.

VI. CONCLUSION

Modern system-on-chip microcontrollers offer simple solutions for control applications at affordable prices. Software development tools for different microcontroller families tend to have similar interfaces, especially if coming from the same software house, but hardware tools (development boards) are specific to microcontroller families. General trend presented in software industry towards open architectures and systems is for moment not valid for development boards. This work tries to make a step further in this direction of making development platform independent of selected microcontroller

solution. Presented solution for development in the field of embedded systems is a part of an electronic project for evaluation tool systems and has great potential for education – is low cost, flexible and has an open architecture. It is also worth to mention that was elaborated together with a number of lecturers within same faculty of Universiti Malaysia Pahang.

ACKNOWLEDGEMENT

REFERENCES

[1] Diane T. Rover, *et al.*, "Reflections on Teaching and Learning in an Advanced Undergraduate Course in Embedded Systems," in *IEEE Transactions on Education*, Vol. 51, No. 3, August 2008.

[2] Kenneth G. Ricks, David Jeff Jackson, and William A. Stapleton, "An Embedded Systems Curriculum Based on the IEEE/ACM Model Curriculum," in *IEEE Transactions on Education*, Vol. 51, No. 2, May 2008.

[3] Hiroyasu Mitsui, Hidetoshi Kambe, and Hisao Koizumi, "Use of Student Experiments for Teaching Embedded Software Development Including HW/SW Co-Design," in *IEEE Transactions on Education*, Vol. 52, No. 3, August 2009.

[4] Saeid Nooshabadi, and Jim Garside, "Modernization of Teaching in Embedded Systems Design—an International Collaborative Project," in *IEEE Transactions on Education*, Vol. 49, No. 2, May 2006.

[5] Ken Ferens, Marcia Friesen, and Sandra Ingram, "Impact Assessment of a Microprocessor Animation on Student Learning and Motivation in Computer Engineering," in *IEEE Transactions on Education*, Vol. 50, No. 2, May 2007.

[6] Darko Hercog, *et al.*, "A DSP-Based Remote Control Laboratory," in *IEEE Transactions on Industrial Electronics*, Vol. 54, No. 6, December 2007.

[7] Chien-Ying Chen, *et al.*, "EcoSpire: An Application Development Kit for an Ultra-Compact Wireless Sensing System," in *IEEE Embedded Systems Letters*, Vol. 1, No. 3, October 2009.

[8] Databeans (2010, March), 2009 Microcontroller Market Share [online], Available: http://www.electronics.ca/publications/products/2009-Microcontrollers-Market-Share.html

[9] L.C. Ling, "*The MC68HC11 Development Board with Integrated Development Environment (IDE)*", Thesis University Malaysia Pahang 2008

[10] R.S.K Selvakumar, N.M.K Nik Yusoff, H. Abdul Aziz, "MCS51 Educational Development System with IDE Approach," in *Proc International Conference on Software Engineering and Computer Systems 2009 (ICSECS09)*, Kuantan, Pahang, 2009, pp. 434-437

[11] A.K.M Azad & V.K. Lakkaraju, "Development of a Microcontroller Laboratory Facility for Directing Students towards Application Oriented Projects," in *Proc American Society for Engineering Education 2003 IL/IN sectional conference*, Valparaiso, IN, 2003, pp. 145-151

[12] M.Z.B.M.Sapein, "*MINI 11-Microcontroller Laboratory Hardware Platform with IDE Approach*", Thesis University Malaysia Pahang, 2009

[13] T. Stoltz, M. Paulik & N. Al-Holou, "A Microcontroller Laboratory Hardware Platform for the Academic Environment: The UDM-EVB," in *Proc Frontier in Education 35th Annual Conf, FIE'05*, Indianapolis, IN, 2005, pp. S2G-1–S2G-6