MPICds: Modular Application Development System for the Microchip's PIC Microcontroller Family

Miha Smolnikar, Mihael Mohorcic
Department of Communication Systems
Jozef Stefan Institute
Jamova 39
SLOVENIA

Abstract: - In this paper a Microchip PIC microcontroller (μC) application development system consisting of hardware and software tools is described, with special attention given to its applications for educational purposes. It supports the development and transfer of program code from a personal computer to the microcontroller. The first part focuses on the design of the MPIC development system's hardware based on a modular approach, i.e. recomposed for the design of each application, thus ensuring maximum adaptability. The system consists of a programmer, a main board including adapters for a variety of chip packages and plug-in modules. All these hardware components can be fabricated by every potential user. Selected microcontroller is programmed in the standard ICSP mode with the PC software freely available on the Internet. The second part describes why the system is suitable for the pedagogical use. The emphasis is given to its modular and robust design, which is deliberately planned in such a way that even an incorrect use cannot harm any of its components. The paper also gives some suggestions for the system's use at different educational levels.

Key-Words: - PIC microcontrollers,

1 Introduction

A range of devices that we use today in our everyday lives (e.g. telephones, household appliances, cars) contains so called "smart" electronics, which is most often implemented as an embedded system. This is in its fundament a microcomputer designed to control a particular process of the application. The central part of most contemporary embedded systems is a microprocessor system, based upon a microcontroller (μC). These are a special kind of microcomputers integrated on a single chip, which in addition to a central processing unit and a memory consists of numerous application dependent peripherals (e.g. AD and DA converters, timers, communication modules, etc.). Microcontrollers play a pivotal role in electronic control systems and are concerning a broad range of potential users. Being the most widely manufactured processing devices they are a product of interest not only in electrical, but also in mechanical and industrial engineering. With the technology advancing very fast, the progress in this field also has to reflect in the teaching-learning process which calls for novel pedagogical approaches supported by suitable hardware and software tools.

When developing an application we need to be able to easily test and verify the designed system on a real object. Building of prototypes therefore plays an important role in the design process. Aiming to ease and quicken this procedure we designed the MPIC development system (MPICds), destined to μC-based application developers. Its structure is modular, which means we recompose it for each application separately. The main hardware parts are the main board with the attached microcontroller and the programmer, used to transfer the firmware from a personal computer (PC) to a microcontroller. Additionally to these two major hardware parts, several typical plug-in test modules intended for the verification of developed programs have been designed. The system's hardware is designed in such way that every potential user can fabricate it on his/her own. The same can also be said for the proposed accompanying software tools, where only freely available programs were used.

The rest of the paper is organized as follows. We begin with a presentation of assumptions taken into account when designing a MPIC development system. Next a detailed description of the development system's hardware and software is given. Some illustrations of the system's appropriateness for the educational purposes are stated and we conclude with plans for further developments of the system.

2 Background and assumptions

The workflow of designing a microcontroller application can be described as a combination of computer programming and digital circuits constructing [1].

Aiming to offer a system that would ease mastering both fields at the same time the MPIC development system was designed. It is destined to the Microchip's 8-bit PIC microcontroller family [2] which was selected from a range of other microcontroller families from different producers, due to multiple reasons. Firstly, these microcontrollers are widely available on the market at relatively affordable prices. Moreover, we can choose from a wide range of quality development and programming tools (even freely available), while the transfer of the firmware can be always performed using the same interface. Lastly, they are also very popular in hobby electronic, which can also be an important factor for any educational equipment.

When designing the MPIC development system the following assumptions and limitations were taken into account:

- since the application's software development is usually a cyclic process of writing and testing the source code it has to support both, the programming and verification of the source code,
- the system must be structured modularly to assure maximum flexibility,
- not any pre-programmed chip may be used,
- all the needed accompanying software tools should be freely available on the Internet,
- a limitation to the microcontrollers where the transfer of the firmware can be performed using ICSP (In-Circuit Serial Programming) module was considered (a great majority of PIC microcontroller offers this) [3],
- with respect to designing a development system we further restricted the selection to the microcontrollers available in the PDIP type of packages [3],
- programmer must offer a possibility to download a firmware from PC to microcontroller using parallel (LPT) or serial (COM) port,
- programmer must be designed in such way that it can be permanently connected to the main board (i.e. during programming and running of the application),
- all microcontroller's I/O pins must be accessible on the main board plug-in module connectors,
- all the printed circuit boards should be one-sided, so that potential user can fabricate them on his/her own,
- where needed the setting of microcontroller's operation using external signals should be accessible through jumpers or short-circuit connections,
- plug-in modules must be designed in a way that even their incorrect connection does not harm any of the system's components.

These assumptions are based on recognition of problems and difficulties that beginners in the world of microcontrollers are often faced with. The system designed in such way consequently would not result in the most efficient variant, but would on the other hand certainly enable someone to get familiar with the whole application design process.

3 MPIC development system

The intention of the MPIC development system described in this paper is not to compete with the commercially available platforms, but rather to represent a contrast trying to efficiently combine freely available tools and easily accessible components. The system is not meant for a particular curriculum although it can be easily included to secondary school or university courses. Because of its modular concept it is also very appropriate for self-learning, where someone constructs only the components he/she needs, thus reducing the system's costs at the same time.

In the following subsections the key hardware parts of the development system are described more in detail.

2.1 Main board

The main board comprising of a microcontroller represents a central part of the development system, to which all other system components (i.e. power source, programmer, and plug-in modules) are connected when needed. Its electrical scheme is depicted in Fig. 1.

The microcontroller is attached to the main board through a standard 40-pin Textool socket. A limitation to the microcontrollers with up to 40 pins was made due to the fact that a 40-pin package is the largest PDIP package type in the case of PIC microcontrollers. Furthermore, a more frequent variant of pins arrangement was considered, while for microcontrollers with different arrangement or lower number of pins, appropriate adapters were designed. These are connected to the Textool socked instead of a 40-pin microcontroller.

The main board includes a power supply with Greatz bridge whose intention is dual. The first is to be used as a classical full wave bridge, so that input voltage can be either AC or DC. The second is to protect the connected power supply in case the programmer has its own power supply connected. Plug-in modules and programmer are connected to the main board through 10-pin IDC connectors. On each of the IDC connectors for the connection of plug-in modules one microcontroller's port and supply voltage are available, while on the connector of the PORTE additionally the microcontroller's pins for the connection of external reset circuit, external oscillator and external AD converter supply voltage can be accessed. Each microcontroller port is connected to the resistor chain, which is used to perform either pull-up or pull-down function. The operation can be set using appropriate jumper connections. On the IDC connector for the programmer attachment all pins that are necessary for the connection of ICSP compatible programmers are available. Additionally, on the remaining pins, the microcontroller's USART module pins are available, that can be used when downloading the firmware using bootloaders. Programmer's virtual connection-disconnection to the microcontroller is driven manually, using a single switch.

On the printed circuit board also places for the RC and quartz cristal oscillator are foreseen. A simple reset circuit with a key as an execution unit is also designed so that it does not disturb the ICSP programmer activity.

2.2 Programmer

Three particularly important design assumptions were taken into account when designing a programmer's hardware part. These are that the microcontrollers must be programmable in the ICSP mode, that the firmware download can be performed using serial or parallel PC port, and that the accompanying software programming tools are freely available on the Internet. Programmer is therefore designed to be compatible with the JDM [4] and P16PRO [5] programmers, where the download of the firmware is in the case of JDM programmer achieved using COM port, while the P16PRO uses LPT port.

ICSP is a synchronous procedure, specified by Microchip, used to program the PIC microcontrollers [6]. It uses five or six microcontroller pins for writing and reading the firmware. The sixth pin must be driven only at microcontrollers which can also be programmed using a second type of programming procedure, i.e. the Low-Voltage ICSP.

Electrical scheme of the programmer is depicted in Fig. 2. The circuit consists only of standard elements and can be connected to the main board using IDC connector cable. Similar to the main board, the programmer has its own power supply which again has dual intention. The first is to protect the microcontroller against damages during programming using a build in 100 mA current limit, and the second is simply to enable its autonomous use, e.g. for programming the microcontroller already attached to the end application. For such cases an FCC

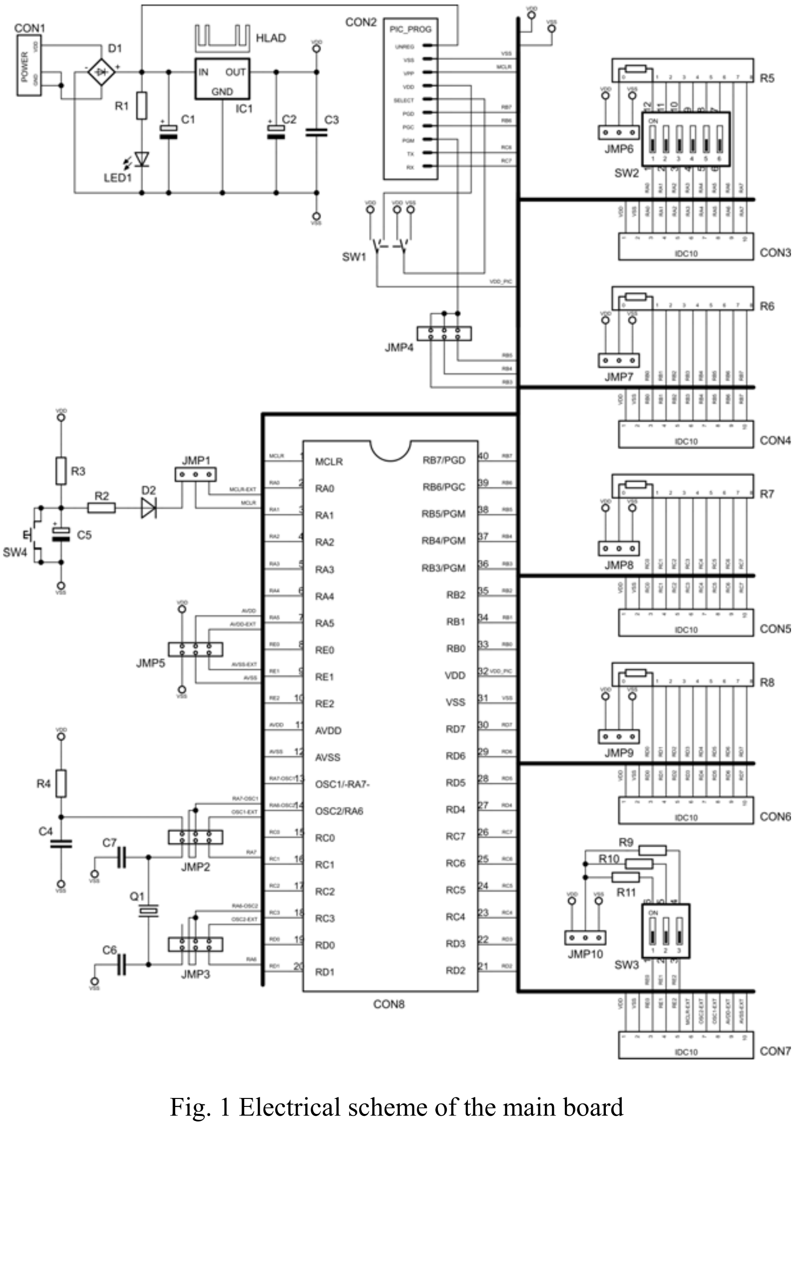

Fig. 1 Electrical scheme of the main board

connector is foreseen on the printed circuit board.

The main programmer's element bridging the PC and microcontroller is the high-speed CMOS integrated circuit 74HCT367. It contains six three-state non-inverting buffers, which are driven by two enable inputs. Using those the firmware is transmitted from the PC to a microcontroller and vice versa. The circuit additionally contains some other elements from which the transistors and zener diodes should be exposed. The latter serve for the adaptation of the COM port voltage levels, which in contrast to LPT port are not TTL compatible. The transistors are used whether to assure appropriate current loads or to convert the voltage levels, since in the ICSP programming mode a voltages higher than 5 V have to be generated.

2.3 Plug-in modules

For the purposes of testing and validating programs written into microcontrollers a set of basic test circuits has been designed. We refer to them as plug-in modules as they are connected to the main board for each application design. Some of those are:

- *PortKEY* – a module to drive a state of a particular microcontroller's input using a key,
- *PortLED* – a module to represent a particular microcontroller's output by lighting a LED,
- *LCDa16* – an LCD module with two 16 characters alphanumeric lines,
- *DS1820* – a module with attached Dallas DS1820 temperature sensor.

The modules are so that that their final function is adapted by setting jumpers or short-circuit connections, thus offering maximum usage flexibility.

An example of fabricated development system is depicted in Fig. 3. A more detailed description of each system's component can be, together with the electrical scheme and a suggestion of a printed circuit board design found in [7].

4 Software tools

During the microcontroller application development several different software tools are used. When working with the PIC microcontroller family we can select among a wide range of quality software tools from different producers, where many of them (including some from Microchip) are in its full functionality freely available on the Internet. We can coarsely divide them into the development tools in which we are developing the source code, and the programming tools used to transfer the compiled firmware from PC to microcontroller.

4.1 Development tools

Microchip offers a full range of development tools for its PIC microcontroller family. These include an integrated development environment (IDE) usually dubbed MPLAB, consisting of assembler, C compiler, simulator product selector guide etc. Integrated with these tools there are also programming tools, which normally support only the Microchip's accompanying hardware. For that reason in the next section some alternative software tools, meant to be used with our programmer, are presented.

Besides the Microchip's development tools we can choose among development tools from different developers. These are typically compilers from higher-level languages (e.g. C, Basic, Pascal) [8], real-time operating systems, and tools adapted for operating systems other than Microsoft Windows (e.g. Linux).

4.2 Programming tools

Hardware part of the MPIC development system can be

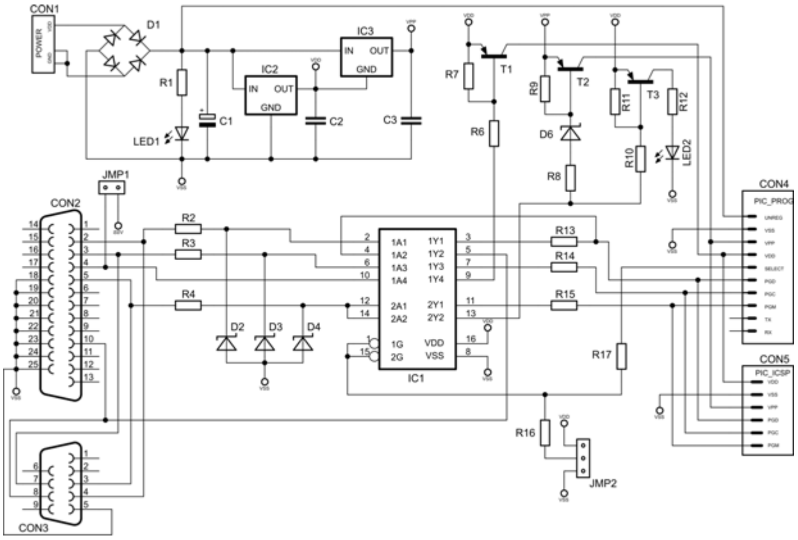

Fig. 2 Electrical scheme of the programmer

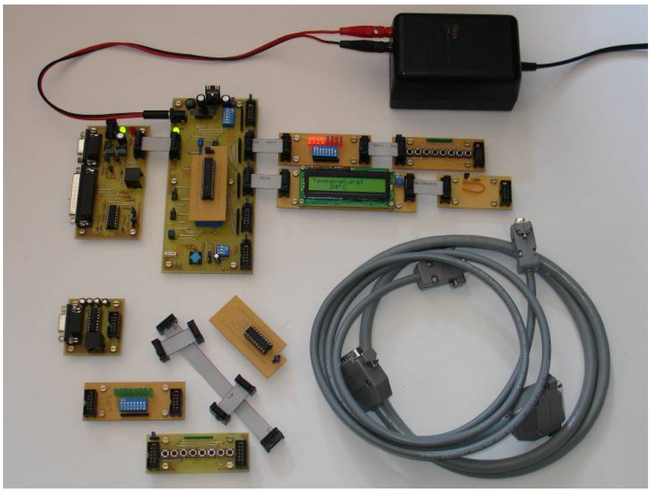

Fig. 3 Photography of fabricated MPICds

accompanied using one of the following freely available software programming tools:

- *IC-Prog* – can be used for programming the microcontroller through PC's COM port, when the programmer works as JDM compatible,
- *PICALL* – when programming using PC's LPT port and the programmer acting as P16PRO compatible,
- *WinPic* – fully customizable programmer that can be used to program the microcontroller either using serial or parallel port.

Settings for the appropriate program operation with the presented programmer can be found in [7].

5 Suitability for educational purposes

Provided that during the design of MPIC development system all the predefined limitations and assumptions were fulfilled, the system resulted in robust educational equipment that can be used either in secondary school or university courses. Moreover, the system is not destined only to electrical engineering education, but can be due its modularity quickly adapted for courses in other engineering specialties. Eventually the system is also efficient enough to serve experts in designing their µC-based applications.

In case of incorporation of the system into secondary school course it can serve as common supporting educational equipment at multiple courses. For instance, the system can be used in courses dealing with basics of microprocessors and microcontrollers, their programming (in assembler or higher-level language), and further in the classes concerning additional plug-in modules design. The most motivated students can also fabricate system (or its particular part) on their own.

From the university's point of view the system can be used as educational equipment in courses dealing with microprocessor, microcontrollers or embedded systems, to cover both the programming and hardware peripheral design support. Due to the growing need of interdisciplinary education the system is also an attractive solution for students of other engineering sciences, e.g. computer, mechanical, chemical or industrial engineering education. Some subjects that the system can help covering are: computer architectures, digital signal processing, mechatronics, automated measuring, etc.

6 Conclusion

In this paper the MPIC development system for the Microchip's PIC microcontroller family was presented together with some proposals for its use in educational courses. The system is designed as open as possible, which means everybody can fabricate it on his/her own and use it with freely available accompanying software. The system is designed generally enough to be used as a whole or as component in other systems. It is not focused on a particular microcontroller device, but it enables to select the most appropriate device for each application design. When using as a whole, the MPICds represents a very robust system, where even an incorrect use does not harm any of the system's components. The development and improvement of the system is still in progress. Additional plug-in modules (e.g. matrix keyboard, 7-seg LED display, stepper motor module, IR communication module, etc.) are also being designed. As such the system is very appropriate for educational purposes, trying to support those entering to the world of microcontrollers.

References:

[1] M. Predko, Programming and customizing PICmicro microcontrollers, McGraw-Hill, 2000.

[2] Microchip, Technical Library CD-ROM 2004/2005 (DS00161Q), Microchip Technology Inc., 2004.

[3] Microchip, Product Selector Guide 2005 (DS00148K2) , Microchip Technology Inc., 2005.

[4] –, PIC-Programmer, http://www.jdm.homepage.dk/newpic.htm, April 2007.

[5] –, P16PRO and PICALL, http://picallw.feniks-pro.com/, April 2007.

[6] Microchip, PICmicro 18C Familiy Refeference Manual – In-Circuit Serial Programming (DS39531A), Microchip Technology Inc., 2004.

[7] M. Smolnikar, MPICds – MPIC development system, http://MPICds.googlepages.com/, April 2007.

[8] –, HI-TECH PICC Compiler for the Microchip PICmicro family, http://www.htsoft.com/products/compilers/picccompiler.php, April 2007.