

Programming a Microcontroller

Altat Hussain
Computer Science Department
KICSIT, Rawalpindi
UET Taxila, Pakistan

Muhammad Hamad
Computer Science Department
KICSIT, Rawalpindi
UET Taxila, Pakistan

Kamran Hafeez
Computer Science Department
KICSIT, Rawalpindi
UET Taxila, Pakistan

Tabinda Zainab
Computer Science Department
KICSIT, Rawalpindi
UET Taxila, Pakistan

ABSTRACT

Microcontroller is a computer on single integrated circuit that includes a CPU, RAM, some form of ROM, and I/O ports. It has great impact in our life which cannot be ignored. Unlike a general purpose computer, microcontrollers are dedicated to perform a specified task and execute single application. Automatically controlled products like automatic engine control systems, remote controls, power tools, toys, and office machines i.e. photo-copier, printer, and fax machines which are used commonly; are being programmed using microcontroller. Any person working in the field of electronics has the knowledge to use a microcontroller but not necessarily be having good programming skills for it. While programing for a microcontroller, one has to interact with different software tools i.e. Proteus for making circuit diagram, mikroC PRO for source coding, and QL-PROGen for burning hex file in microcontroller. So it is really a tough task to program a microcontroller for such persons. Our work aims to extend the automatic programmer tool [1] that generates mikroC code and its hex file from a block diagram. This enhanced tool provides an integrated platform that allows the user to design circuit diagram, generate its code, and burn the hex file directly into the microcontroller. It provides a set of blocks for business logics; that are used to build an algorithm of application as an input for source code generation. In addition, it provides the feature to compile the generated code and then to burn the hex file directly into the chip. The microcontrollers used in this tool are PIC18F442, PIC18F4520, and PIC18F542; placed in the middle of Circuit Diagram Window of this tool.

General Terms

PIC programmer.

Keywords

Microcontroller, PIC18f452, hex file generator, graphical block, OpenBlocks for microcontrollers.

1. INTRODUCTION

A microcontroller is a computer-on-a-chip, or, it may be considered as a single-chip computer. Micro means a small device, and controller identifies it as a device to control objects, processes, or events. An alternate term used for it is embedded controller, because the microcontroller and its support circuits are often built into, or embedded in, the devices they control. Microcontrollers are usually used in devices from all aspect of life. Any device that measures, stores, controls, calculates, or displays information is a candidate for putting a microcontroller inside [1]. The largest use for microcontrollers is in automobiles; just about every

car manufactured today includes at least one microcontroller for engine control, and often more to control additional systems in the car. In desktop computers, you can find microcontrollers inside keyboards, modems, printers, and other peripherals. Similarly, Consumer products that use microcontrollers include cameras, video recorders, compact-disk players, and ovens etc. but these are just a few examples. A microcontroller is similar to the microprocessor inside a personal computer [1]. To make a complete computer, a microprocessor requires memory for storing data and programs, and input/output (I/O) interfaces for connecting external devices like keyboards and displays. In contrast, a microcontroller is a single-chip computer because it contains memory and I/O interfaces in addition to the CPU. Because the amount of memory and interfaces that can fit on a single chip is limited, microcontrollers tend to be used in smaller systems. Examples of popular microcontrollers are Intel's 8052, Motorola's 68HC11, and Zilog's Z8.

In 1971, the first microcontroller was invented by two engineers at Texas Instruments, according to the Smithsonian Institution. Gary Boone and Michael Cochran created the TMS 1000, which was a 4-bit microcontroller with built-in ROM and RAM. The microcontroller was used internally at TI in its calculator products from 1972 until 1974, and was refined over the years. In 1974, TI offered the TMS 1000 for sale to the electronics industry [8]. In addition, Intel also developed many important microcontrollers, two of which are the 8048 and 8051. The 8048 was one of Intel's first microcontrollers introduced in 1976 and was used as the processor in the IBM personal computer keyboard. The 8051 followed in 1980 and became one of the most popular microcontroller families. Variations of the 8051 architecture are still being produced today, making the 8051 one of the most long-lived electronics designs in history [8].

During the 1990s, microcontrollers with electrically erasable and programmable ROM (EEPROM) memories, such as flash memory, became available. These microcontrollers could be programmed, erased and reprogrammed using only electrical signals. Prior to the electrically reprogrammable devices, microcontrollers often required specialized programming and erasing hardware, which required that the device be removed from its circuit, slowing software development and making the effort more expensive. Overcoming this limitation, microcontrollers were able to be programmed and reprogrammed while in a circuit so devices with microcontrollers could be upgraded with new software without having to be returned to the manufacturer. Many current microcontrollers, such as those available from Microchip and Atmel, incorporate flash memory technology.

In addition to general purpose devices, specialized microcontrollers are being produced for areas such as automotive, lighting, communications and low-power consumer devices. Microcontrollers have also become smaller and more powerful. For example, in 2010, Atmel announced a flash microcontroller in a package measuring 2 mm by 2 mm. These tiny microcontrollers are small enough and cheap enough to be used in products such as toys and toothbrushes [8].

On such a demanding use of microcontrollers, the need of writing code for the microcontrollers is becoming an important task. To make this task easier, we have introduced the initial version of automatic programmer [4]. Automatic Programmer is a software tool that allows the user to generate mikroC code for microcontroller PIC after creating his/her desired block diagram; then it generates the Hex file of the following code. A block diagram is basically a group of properly interconnected blocks, with each interconnected block representing and describing a portion of the system. The block diagram is very simple to construct also for complicated system. The function of individual element as well as overall performance of the system can be plotted/ studied from the block diagram. Due to their simplicity and versatility, block diagrams are widely used by control engineers to model all types of dynamic systems [5]. For the verification of the respective generated mikroC code, the Proteus Software is being used in Automatic Programmer. In this work we have enhanced the automatic programmer as explained in Section 4.

The remainder of this paper is organized as follows: Section 2 gives a detail of programming for microcontroller. Section 3 describes the automatic programmer [4]. Section 4 presents enhancements of automatic programmer and the use of this extension. While Section 5 presents a calculator case study using enhanced automatic programmer. Section 6 presents the Conclusion of this work.

2. EXISTING SYSTEM

2.1 Using a Microcontroller

Normally, when we use microcontroller for making some application there are at least 3 steps to follow, first of all design the circuit, write C code for it and generate hex file, and to burn this hex file into the microcontroller. For each step to complete, we have to use separate software. The steps are discussed here.

2.1.1 Circuit Designing

Usually in step-1, the circuit is designed with the help of Proteus software. Proteus developed by Lab center Electronics Ltd. was founded in 1988. The Proteus Design Suite is wholly unique in offering the ability to co-simulate both high and low level microcontroller codes in the context of a mixed-mode SPICE circuit simulation. SPICE (Simulation Program with Integrated Circuit Emphasis) is a general-purpose, open source analog electronic circuit simulator. Proteus's Virtual System Modeling (SVM) has microcontroller programming tool, environment, with its many software features and hardware options [9]. Proteus is a circuit designing software that contains a database of components, which we can use to build the circuit. The components to be used are available in the component library.

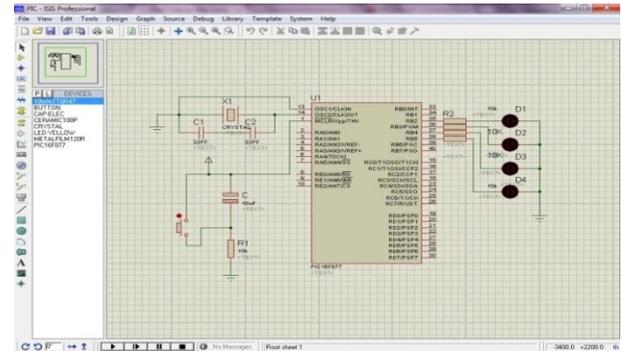


Fig. 1: Proteus: A Software for circuit designing

2.1.2 Programming a Microcontroller

A microcontroller does not know what to do by itself. So in step-2, it is the responsibility of programmer to write a code for microcontroller. The compiler for this purpose is mikroC PRO or MP-Lab. After writing the code, it is compiled for respective microcontroller in the form of a hex file.

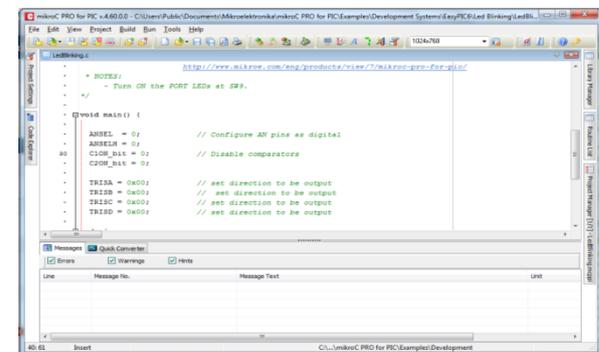


Fig. 2: mikroC PRO: C Compiler for PIC

2.1.3 Burning hex file

The code loading hex process of microcontroller is called dumping. The microcontrollers understand only the machine level language, which contains '0s or 1s'. So we need to load the hex code into the microcontroller. There are many software tools available for loading the code to the microcontroller e.g. QL-PROGen or PICFLASH [3].

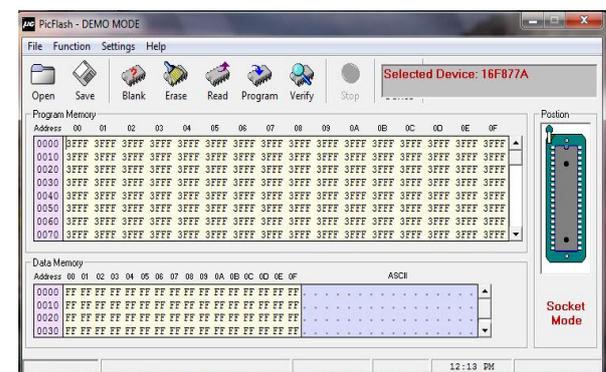


Fig. 3: PicFlash: Software used for dumping

3. AUTOMATIC PROGRAMMER [4]

To make this task easier, a software tool has been introduced by Hussain et al. in the form of Automatic Programmer [4]. It was a software tool that allows the user to generate mikroC code for P*IC microcontroller after creating the desired circuit diagram; then it generates the hex file on compilation. Its

benefit was to automatically generate configuration code i.e. setting of direction register, and user does not have to write code to set direction registers and the hex file is also created.

As discussed earlier, the user has to use three different software tools for circuit designing, programming the chip, and burning hex file onto the chip. Using Automatic Programmer, the user can design the circuit and generate configuration code but there was no mechanism of providing business logic of application that has been introduced in enhanced automatic programmer. In addition, automatic programmer did not provide the mechanism for burning the hex file onto the chip and user has to use separate software for it.

4. ENHANCED PROGRAMMER

Enhanced Automatic Programmer (EAP) is an extension to previously designed software named as Automatic Programmer. EAP is an integrated platform for designing circuit diagram, setting configurations of components (i.e. setting direction registers by mouse click without writing code), making block diagram in which user will provide algorithm of his functionality by joining blocks, burn/load directly this hex file into the microcontroller.

A block diagram is a group of properly interconnected blocks, and each block represents a portion of the system. The block diagram is very simple even to construct for complicated system. The function of individual element as well as overall performance of the system can be plotted/studied from the block diagram. Due to the simplicity and versatility, block diagrams are widely used by control engineers to model all types of dynamic systems [5]. For the verification of the generated micro C code, the same circuit is designed in Proteus software and the generated code is loaded inside the microcontroller to simulate. Proteus developed by Labcenter Electronics Ltd. was founded in 1988. The Proteus Design Suite is wholly unique in offering the ability to co-simulate both high and low level microcontroller codes in the context of a mixed-mode SPICE circuit simulation. SPICE (Simulation Program with Integrated Circuit Emphasis) is a general-purpose, open source analog electronic circuit simulator. Proteus's Virtual System Modeling (SVM) has microcontroller programming tool, environment, with its many software features and hardware options [6].

4.1 Enhanced Programmer Development

It is developed in NetBeans using Java. OpenBlocks is used in designing the interface with which user can make block diagram of the input algorithm. OpenBlocks is an open source java library used for creating block based programming UIs. OpenBlocks consists of two packages: codeblocks and slcodeblocks. Codeblocks is the basic underlying library that is responsible for most of the functionality [7]. Mikro C Pro commands are used for verification and compilation of source code generated by this software. Hex file generated as output of compilation is further used for burning into microcontroller. QL-PROGEN is also integrated for burning code into the microcontroller.

EAP has a visual interface comprised of 1) Component Toolbox, 2) Circuit Diagram Window containing microcontroller, 3) Display Code Window used for displaying generated code and 4) Make Flow Window. User can make circuit diagram using components provided in the component toolbox. The available set of components are LED, LCD- 8bit, Keypad, Stepper motor, and seven segment. The microcontrollers used in our software tool are PIC18F442,

PIC18F4520, and PIC18F542 and it is already placed on the Circuit Diagram Window.

After creating the circuit diagram, the user has to make connections between components and microcontroller. Then the properties of components are set. On plotting the desired circuit diagram, the configuration code is generated for it as shown on the screen in the Display Code Window. Initially user has to create project directory where generated source code is saved as ".c file" with another file named as ".mcppi" which contain configuration setting. These two files are required for compilation. First the user has to select the PIC microcontroller type (PIC18F442, PIC18F452, or PIC18F4520) and then sets its frequency and the project is saved with a specific name. After making circuit diagram user will provide the input algorithm by joining blocks provided in the toolbox and click on update code and this business logic and configuration code will be merged.

4.2 Using Enhanced Programmer

Using "Create Project" option in the File menu, new window will appear that will help the user to create new project directory. On project creation, the main window appears with microcontroller placed on the middle of it. This main window has component toolbox on the left side. This toolbox comprises of components as discussed earlier in section-IV for constructing a circuit diagram. Right side of the main window shows a working area that contains a microcontroller on it and other components are needed to be added from the toolbox as per requirement of input algorithm. Using drag and drop, the user can place any of desired components from the provided toolbox to working area. Three options appeared on the right click on any of the components that are: 1) Delete component 2) Drag component 3) Set component property. With set component property option user can set properties of respective component e.g. to set LED port and pin.

When all components are configured then the basic code is generated by using Generate code button on top of menu bar. This is basic configuration code generated for devices attached to microcontroller. User can view generated code in code tab view. Now user has to make logical flow diagram by using open Make flow diagram window from code tab view. This new window has Component toolbox which contains block design of conditional statements, logical statements, logical operators and block design of functions like "getselectedkey" function used to get keypad selected key, "writeonlcd" function used for writing content on LCD etc. After creating flow diagram click on update code that will generate logical code and merge it with configuration code to make complete final source code.

In order to compile this source code, compilation button from menu bar is used that produced .hex file. At the end, button named "burn code" from menu bar will open a new window. This will ask required information and took .hex file as input to burn it in microcontroller.

5. CALCULATOR CASE STUDY

Create Project directory from "Create Project" option in the File menu. Create Project window will appear, save it with in project folder. Both the project and its folder is named as "Calculator". Now select device name "PIC18F452" from dropdown list and enter device clock 20 Mhz and click create button that will successfully create project directory as shown in Fig. 4.

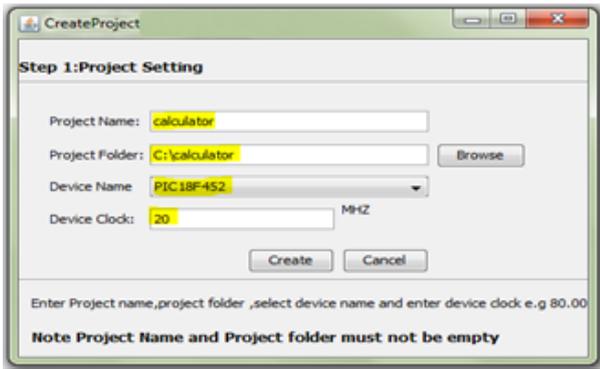


Figure 4: Create Project Window

Now first, make a block diagram for calculator by using LCD and keypad. Drag LCD and keypad from list to viewport and set their properties to make connections with microcontroller. Keypad is connected at port B. Data ports of LCD is connected at port C and RS bit of LCD at bit 0 of port D and E bit at bit 1 of port D as shown in Fig. 5.

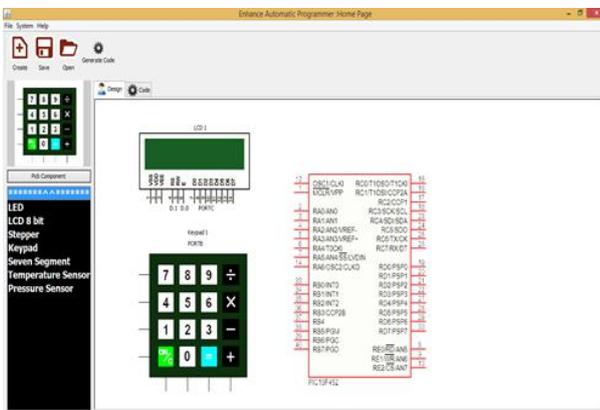


Fig. 5: Designing Circuite Block Diagram

After making connections with microcontroller, use “generate code” button for producing basic configuration code. Configuration code refers to set the direction registers to set as input or output and for keypad it will generate the method to find the key pressed we will simply call this function and can get the value of key clicked. LCD will also be initialized in configuration code as shown in Fig. 6.

Now click on make flow button in the code tab and another window will be opened that will help you to make flow of calculator i.e. to provide algorithm stated earlier in the software. For making flow of algorithms using blocks we have used an open source library openblocks in our software. As we can see there are six different tabs (Control, components, logic operators, number, utilities) on left hand side of screen. Control tab contains conditional statements (if, if else) and loops (for, while), Components tab contains helper methods for hardware components like led on, led off, rotate stepper clockwise, rotate stepper anti-clockwise, iskeypadclicked, keypad. getlickedkey, LCD clear, display on LCD etc. Now start making flow against the calculator algorithm on view port. The algorithm used for calculator is explained in Fig. 7.

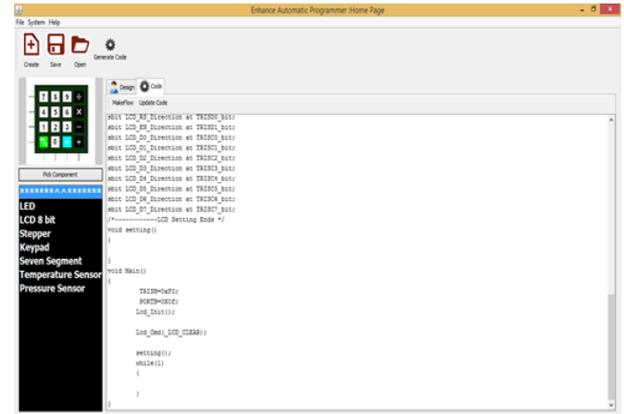


Fig. 6: Configuration Code Window

In the figures 8, 9, 10, 11, and 12, the algorithm in Fig. 7 has been implemented by using respective blocks. In the Fig. 13, hex file of code is generated that will be loaded into microcontroller using this tool and calculator will be ready to use.

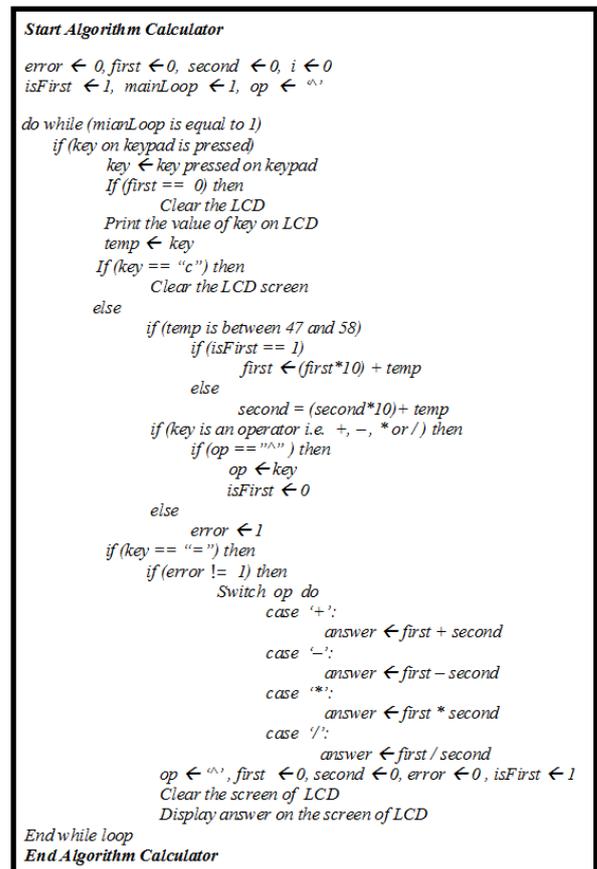


Figure 7: Algorithm for Calculator

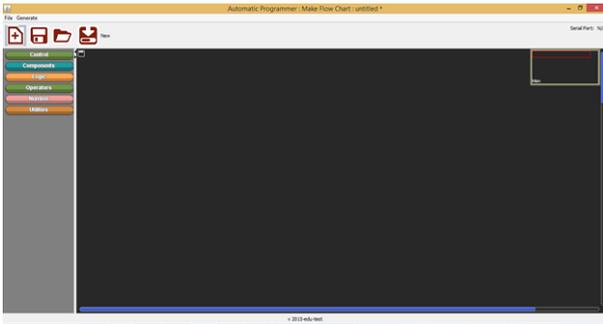


Fig. 8: Make Flow Window

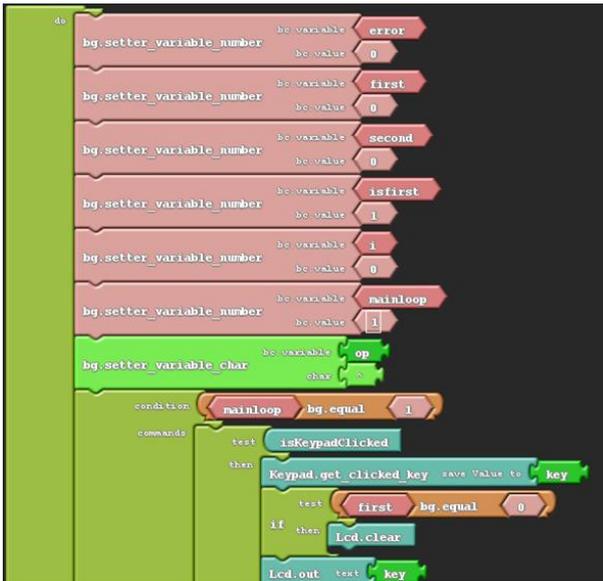


Fig. 9: Calculator Algorithm - Part-01

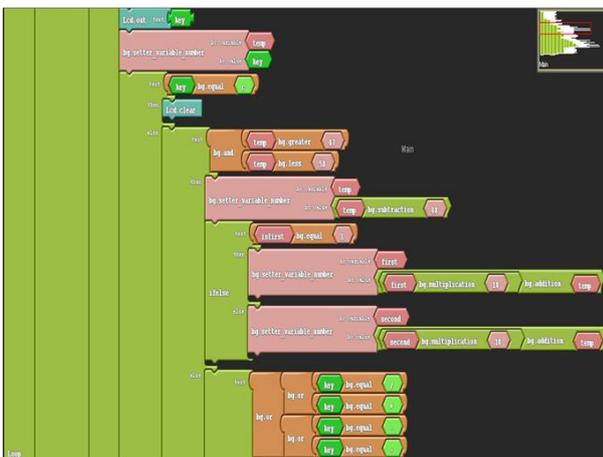


Fig. 10: Calculator Algorithm - Part-02

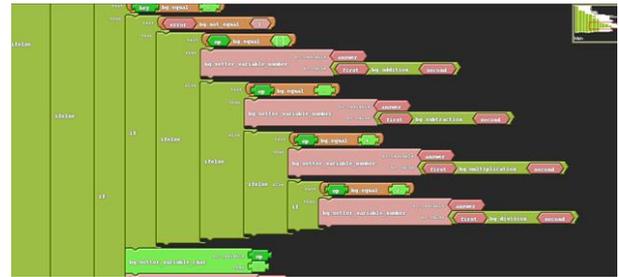


Fig. 11: Calculator Algorithm - Part-03



Figure 12: Calculator Algorithm - Part-04

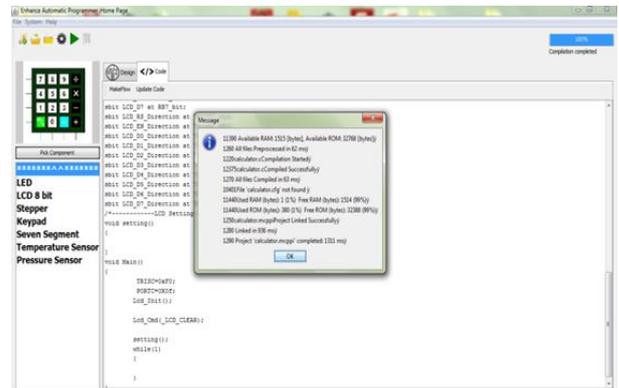


Figure 13: Generate Hex File

6. CONCLUSION

This work is the extension of automatic programmer to provide a friendlier and integrated platform for providing the solution of creating source code for microcontroller from block diagram. The extension to the tool comprise of introducing new components, generation of configuration code and the actual source code. In addition, QL-PROGen is integrated to burn the created hex file in the microcontroller and the same can be pre-simulated within this enhanced tool.

7. REFERENCES

- [1] Axelson, J. (1994). The Microcontroller Idea Book. 2008.
- [2] Verle, M. (2009). PIC Microcontrollers Programming in C: A Complete Guide to Pic Microcontrollers. MikroElektronika.
- [3] How to Program a PIC Microcontroller to Build a Project [online], Avail-able: <https://www.elprocus.com/pic-microcontroller-programming-using-c-language/>
- [4] Hussain, A., Riaz, M. A., ul Abdin, Z., & Saeed, A. (2015, December). Automatic programmer: A software tool. In Emerging Technologies (ICET), 2015 International Conference on (pp. 1-5). IEEE.

- [5] Maxwell, J. C. (1954). *A Treatise on Electricity and Magnetism* Dover Publications. Unabridged Third Edition, Volume one.
- [6] A refactoring branch of MIT OpenBlocks. [online], Available: <http://web.mit.edu/mitstep/openblocks.html>
- [7] Mei, C. (2002). On teaching the simplification of block diagrams. *International Journal of Engineering Education*, 18(6), 697-703.
- [8] Padgavhankar, A. V., & Mohod, S. W. (2014). Experimental Learning of Digital Power Controller for Photovoltaic Module Using Proteus VSM. *Journal of Solar Energy*, 2014.