Automatic Programmer: A Software Tool

Altaf Hussain[1], Mudassar Ali Riaz[2], Zain ul Abdin[3], Afshah Saeed[4]
Department of Computer Science
Dr A. Q. Khan Institute of Computer Science and Information Technology, KICSIT
Kahuta Rawalpindi, Pakistan
{saddamaltaf, mudassar9287, zraja15, afshanghouri57}@gmail.com

***Abstract*—Nowadays, Microcontroller has great impact in our life which can't be ignored. Many automatically controlled products i.e. automobile engine control systems, and embedded systems etc. are being programmed using microcontroller. Any person working in the field of electronics has knowledge to use a microcontroller but it's difficult for him/her to write code for microcontroller because every person may not have good programming skills to write code for microcontroller. So it's really tough task to program a microcontroller for such person. Our work aims to provide a solution in the form of Automatic Programmer for electronic engineers with such problems. It is basically software solution that allows the user to generate the code for microcontroller by providing a desired block diagram as an input. This tool allows its user to create the block diagram of desired circuit as input and generate the code for it and also generates its HEX file of the respective generated code.**

Keywords— microcontroller, Hex file generator for PIC, mikroC code generator tool, PIC microcontroller code generator

I. INTRODUCTION

Microcontroller are task-specific chips cheaper to be build and reliable in the field. Most microcontrollers have the ability to control program, data and input- output ports [1]. Micro means small in size and controller means a type of device that controls some processes, objects, or events etc. so microcontroller is considered to be a computer-on-a-chip or it's a single- chip computer. It is also known as an embedded controller as being embedded in the devices that it control. A microcontroller is available in different word lengths like microprocessors (4bit, 8bit, 16bit, 32bit, 64bit and 128 bit microcontrollers are available today) [2]. Consumer products that contain microcontroller include remote control, cameras, recorders, compact-disk players, oven, vending machines etc. but these are a few examples. The largest use of microcontrollers is in automobiles as every car manufactured today contains at least one embedded Controller for engine control and often more microcontrollers for other additional systems like power control, temperature sensor and speed control etc. Similarly we find microcontrollers in scanners, printers, mouse, keyboard and other peripheral devices of computer systems as well. Simply any product/device that measures, calculates, stores, displays information may be assumed as a candidate for putting microcontroller inside it [3]. Consumer appliances are not the only contain microcontroller; instead machinery, aerospace design, robots and other high-tech devices are also built with these embedded controllers. Robots have been designed to do everything from assisting students in their learning for controlling motors, reading sensors and communication other computers, to mowing lawn, and to solving complex mechanical problems such as autonomous remote flight robot [4]. In today's world, almost every electronic product in the household or place of business contain embedded controller. Moreover, just only the product being used in one's house without microcontroller is the light bulb [5].

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 [6]. 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 [6].

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. 0vercoming 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.

978-1-5090-0436-2/15/$31.00 ©2015 IEEE

These tiny microcontrollers are small enough and cheap enough to be used in products such as toys and toothbrushes [6].

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. 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 [7]. For the verification of the respective generated mikroC code, the Proteus Software is being used in Automatic Programmer. 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 [8].

The remainder of this paper is organized as follows: Section 2 gives a detail of automatic programmer development detail. Section 3 describes the demonstration of this tool for constructing block diagram and generating code. Section 4 presents the Conclusion and future work.

II. Automatic Pprogrammer Development

This tool is developed on .Net platform using C#.Net and micro C for creating HEX file. This software tool named as Automatic Programmer has a visual interface comprised of Component Toolbox, Block Diagram Window containing microcontroller, and Display Code Window. Automatic Programmer enables its user to create the block diagram of desired electronic circuit by drag and drop components from provided toolbox. This toolbox enables the user to build a block diagram using LED, LCD-4bit, LCD- 8bit, DC motor, Stepper motor, seven segment anode, and seven segment cathodes. The microcontrollers used in this software tool are PIC18F442, PIC18F4520, and PIC18F542 and it is already placed on the Block Diagram Window. After creating the block diagram, the user has to make connections between components and can set the properties for it. On plotting the desired block diagram, the code is generated as per block diagram and shown on the screen in the Display Code Window.

Initially, Block Diagram Window has a work area with a microcontroller placed on it along with a toolbox on one side containing the mentioned components as in Figure 1. For constructing the block diagram, user selects the desired component from the toolbox and drops it on the work area. Then this component is attached to the microcontroller. On selecting and dropping a component on the working area, a dialog box appeared for setting the properties of this component. In this manner, the whole block diagram can be plotted by user. Now the user can generate code for this block diagram by using the "Display Code" option on the screen. When the code generates, meanwhile a HEX file of that code also gets generated.

Using "Display Code" option, three files generated at a time; i) C file, ii) HEX file and iii) text file. These files can be saved in the same folder which is initially created by user for new project. When the user saves this code, first the user has to select the PIC microcontroller type (PIC18F442, PIC18F452, or PIC18F4520), set its frequency and name the project for saving this project. On saving the project, the Proteus we have integrated in our software tool launches for verifying the generated code by constructing same block diagram and then loads the generated Hex file for verification purpose.

III. Automatic Programmer Demonstration

Using "New" option in the File menu, the new project window is appeared as shown in Figure 2. Left side of the window shows a toolbox named Components comprises of components available for constructing a block diagram. Right side of the window shows a working area named Block Diagram. This working area already contains a microcontroller on it and other components are needed to be added from the toolbox by the user as per requirement. Constructing a block diagram, the user drags any of desired components from the provided toolbox. When a component is being dropped on the work area, meanwhile the properties window of that component appear as popup menu. Here the example of stepper motor is shown in Figure 3. It allows the user to set the properties of the component through this window as required. User is required to fill all the fields of properties window and click Apply button. Then the block diagram after adding stepper motor and setting its properties will be shown as in Figure 4.

For generating code of the respective block diagram, the user must have to select the Display Code option from the provided menu which is on the top of the window. By selecting this, the code generates for the respective block diagram and displays on a separate code window. On the Code Window there is a Setting section present on the right side. This allows user to select the type of PIC microcontroller (PIC 18F 442, PIC 18F452, or PIC 18F4520), and set its frequency and naming the project, then click on Save button. All the fields are required to be filled by the user with valid input.

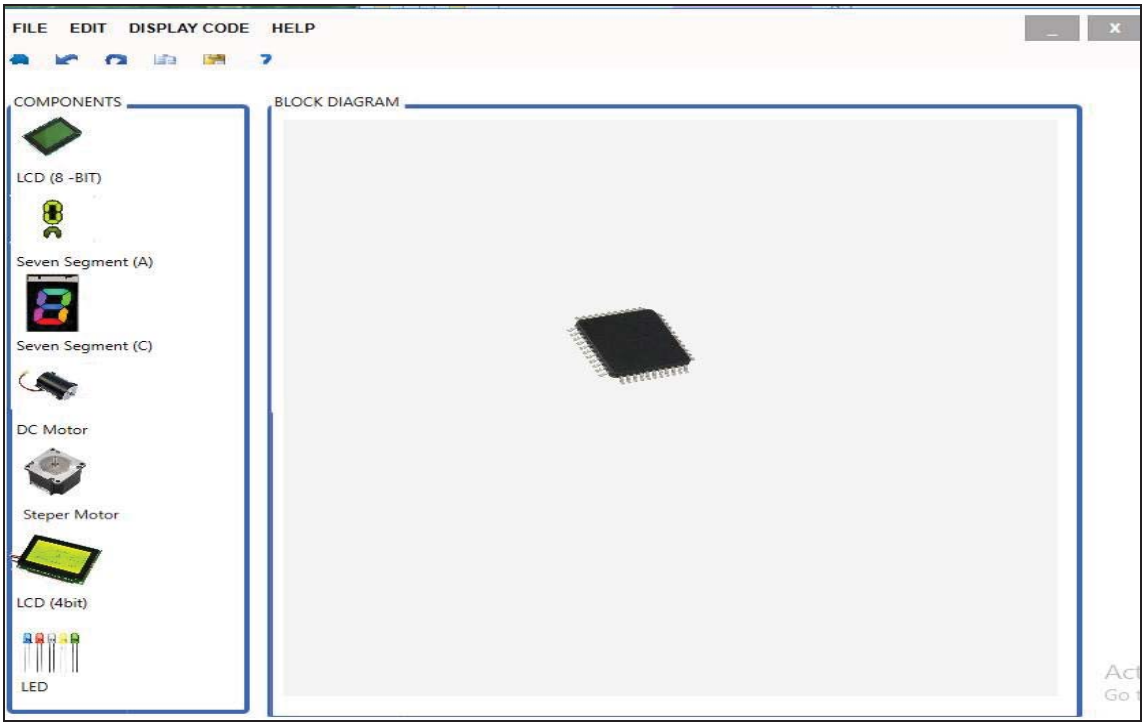

Figure 1: New project window for constructing block diagram

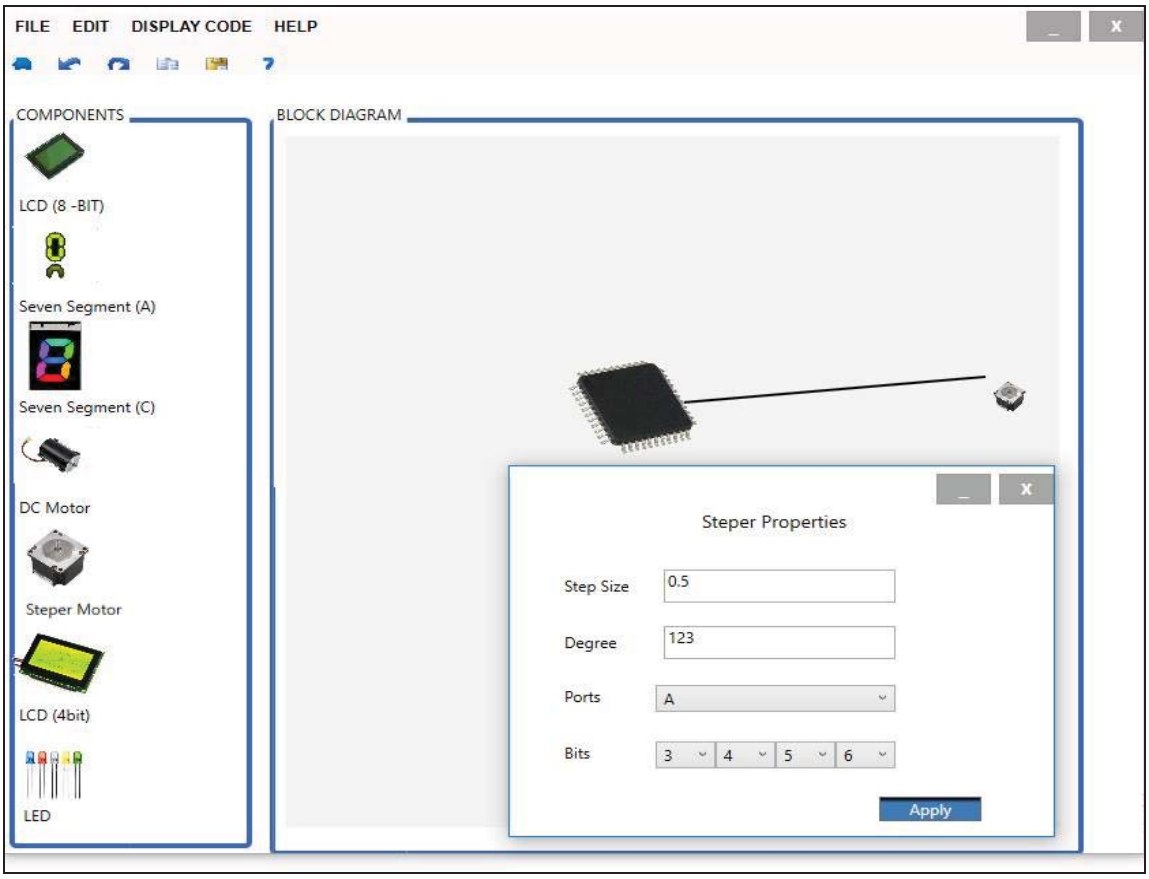

Figure 2: Adding stepper motor and setting its properties

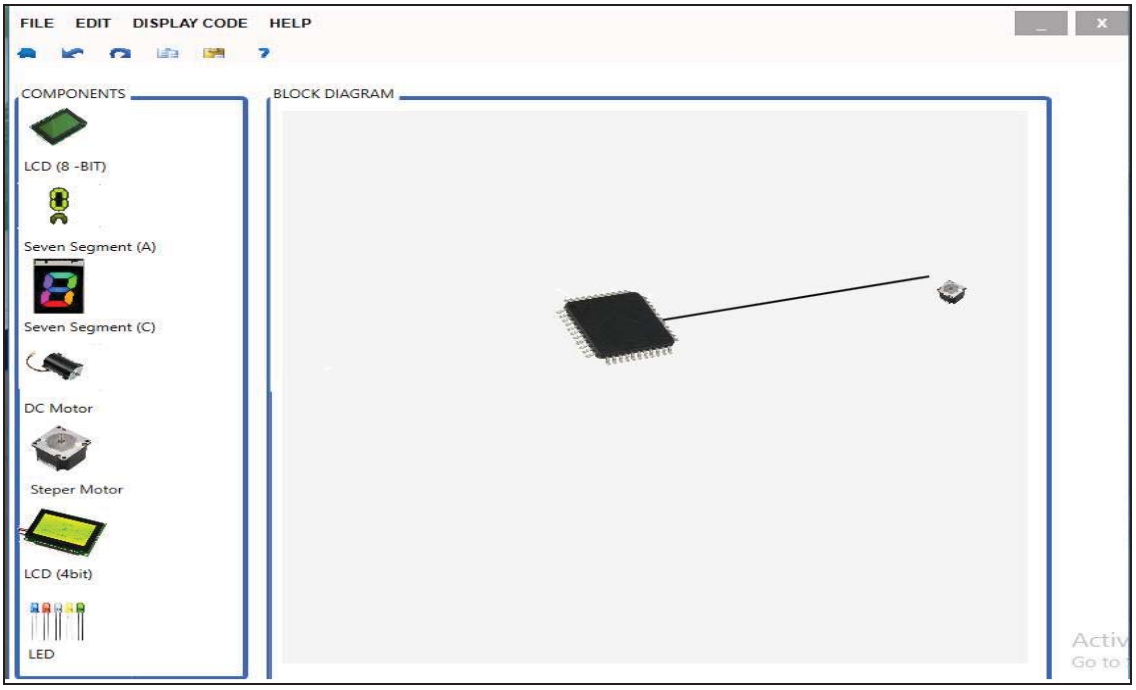

Figure 3: Block diagram with stepper motor shown in working area

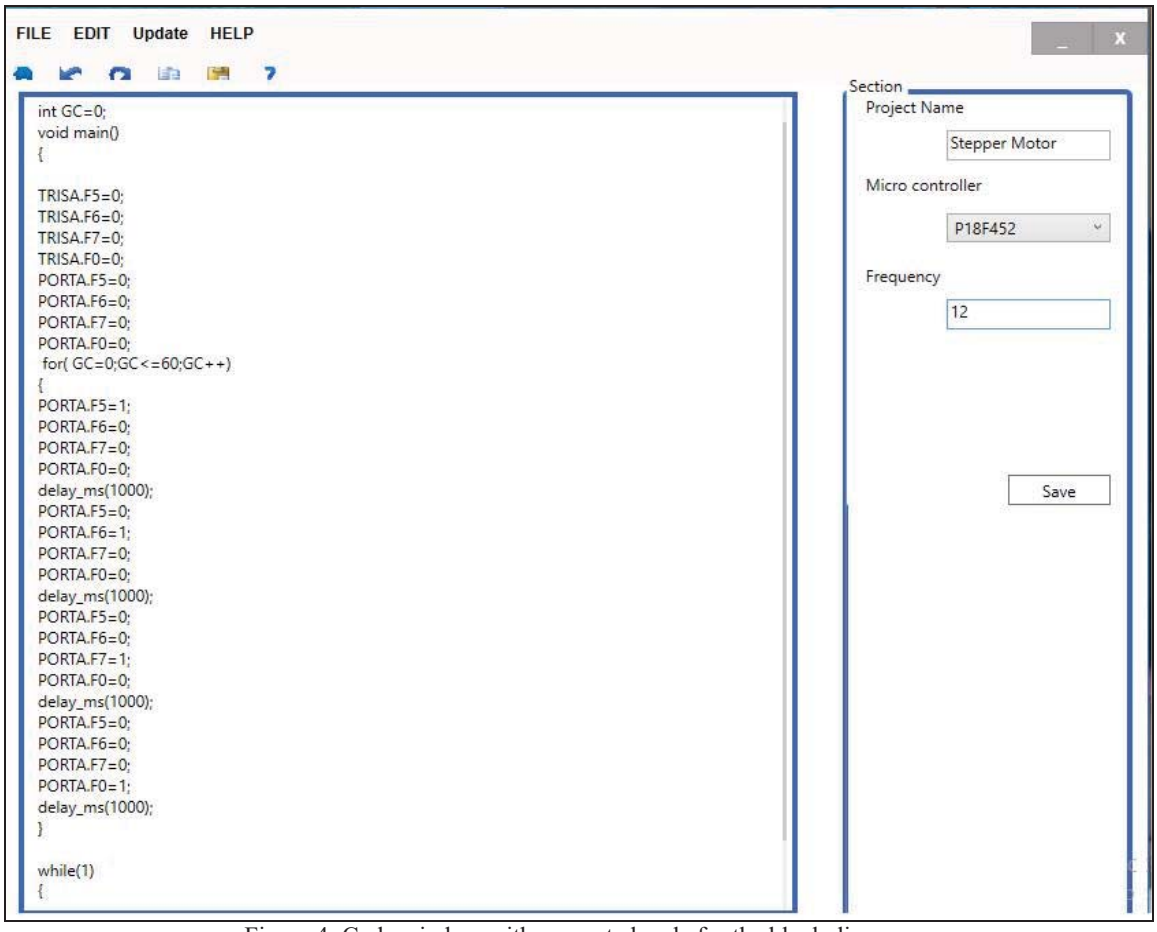

Figure 4: Code window with generated code for the block diagram

A user can also make some amendment in the generated code of a saved project. When the generated code is displayed in the Display Code window, there is an "Update" option present in the menu on the top of the Display Code window. Using Update option, the user can amend the block diagram of the code of saved project. Now user can also add component to or remove component from the block diagram and can again generate the code as like earlier. For updating the stepper motor properties, the user clicked on the stepper motor that is attached with microcontroller and the properties window of stepper motor will reopened for modification. In addition, the user can remove a component in this process of modifying diagram. After block diagram modification and then using the Display Code option will create new code for it.

IV. Future Enhancement and Conclusion

This tool is developed with the intention to provide a solution for creating PIC microcontroller code using a few of initial component in block diagrams. It facilitates its user by providing a way to generate code without having the required programming skills. The tool has a user friendly interface and creates a code along with verification by Proteus SVM. Being its first version, this tool has components in scarce but in the later versions more components can be considered to add more components for constructing block diagram and strengthening it as a tool for more enhanced applications. Moreover, it can be modified for other microcontrollers.

Acknowledgment

We pay our gratitude to the National Grassroots ICT Research Initiative project for FYP funding by Ministry of Information Technology, Govt. of Pakistan. This work was approved for FYP funding as per the list Approved Projects 2013-2014 on the website of mentioned project of ministry of IT. We appreciate the project for promoting the involvement of young graduates to contribute in R&D based projects as their FYP.

References

[1] J. Guilford. The World of Microcontrollers: A Beginning Guide to the HC11 Microkit.

[2] Vysakh. Basics of Microcontrollers. *CT Circuit Today*. [Online], Available: http://www.circuitstoday.com/basics-of-microcontrollers

[3] J. Axelson. "The Microcontroller Idea Book Circuits." *Programs & Applications featuring the 8052-BASIC single-chip computer.* ISBN 0-965081907, Madison, 1994.

[4] R. A. Khan. "Workhorses of the electronic era [microcontrollers]." *Spectrum, IEEE. Vol.* 33.10, pp. 36-39, 1996.

[5] S. Aycock. The History of the Microcontroller. *eHow contributor*. [Online], Available: http://www.ehow.com/info_10018768_history-microcontroller.html

[6] C. Mei. "On teaching the simplification of block diagrams.", *International Journal of Engineering Educatio.* Vol. 18.6, pp. 697-703, 2002.

[7] J. C. Maxwell. A Treatise on Electricity and Magnetism. *Dover Publications Inc.* vol. 2, pp.68-73. 1954.

[8] A.V. Padgavhankar and S. W. Mohod. "Experimental Learning of Digital Power Controller For Photovoltaic Module Using Proteus VSM." Journal of Solar Energy, *power* 4: 6,(2014)