PROJECT-BASED MICROCONTROLLER SYSTEM LABORATORY USING BK300 DEVELOPMENT BOARD WITH PIC16F887 CHIP

Lukman A. Ajao [1], Olayemi M. Olaniyi [2], Jonathan G. Kolo[3], Abdulazeez O. Ajao[4]

[1,2,3]Department of Computer Engineering, Federal University of Technology, Minna, Nigeria

[3]Department of Computer Engineering, Federal Polytechnic, Offa, Nigeria

Abstract

Microcontroller system is one of the vital subjects offered by students during the sequence of study in universities and other colleges of science, engineering and technology in the world. In this paper, we solve the problem of student comprehension and skill development in embedded system design using microcontroller chip PIC16F887 by demonstration of hands-on laboratory experiments. Also, developments of software code, circuit diagram simulation were carried out. This is to help students connect their theoretical knowledge with the practical experience. Each of the experiments was carried out using BK300 development board, PICKit3 programmer, Proteus 8.0 software. Our years of experience in the teaching of microcontroller course and the active involvement of students as manifested in complete in-depth hands-on laboratory projects on real life problem solving. Laboratory session with the development board and software demonstrated in this article is unambiguous. Future embedded system laboratory session could be designed around ATMel lines of Microcontrollers.

Keywords

Microcontroller, Embedded system, Hands-on lab experiments, simulation, and PICKit3 programmer

1. INTRODUCTION

The impression of studying microcontroller system is globally increasing in various branches of science, engineering and technology department of Universities and Polytechnics [1-2]. The use of microcontroller chips for controlling the embedded system functions are increasing exponentially in everyday technology design activities in colleges and technological research institutes [3].

A microcontroller is an entire computer integrated on a single chip that incorporates all the features that are found in microprocessor. For instance, microcontrollers are used as controllers in automobiles and as system exposure and focus controllers in camera. For the purpose of these applications, they have high concentration of on-chip facilities such as built in ROM, RAM, I/O ports, Serial Port, Parallel I/O ports, Timers, Counters, Interrupts controllers, Analog-to-Digital Converters, and Clock circuit. Since microcontrollers are powerful digital processors, the degree of control and programmability they provide significantly enhances the effectiveness of the applications.

DOI : 10.5121/ijesa.2015.5302   15

Teaching microcontroller system as a course for department like computer engineering, telecommunication engineering, electrical and electronics engineering and computer science students in a conventional way is inefficient and insufficient, if the teaching methods overlook hands-on laboratory sessions[4]. In addition, embedded systems development, microcontroller system interconnection and configurations are of paramount importance to learn practicality. Simply because such embedded systems are widely used in several fields, and the demand for trained personnel in this field is increasing fast. However, the teaching methods and laboratory training in embedded systems is lagging behind (5). One of the major problems in teaching microcontroller system course is how to help students make the reasoning that connects their theoretical knowledge with practical experiences (6).

Experiments descriptions on microcontroller system teaching are continuously monitored and improved based on changes taking place in the architecture of microcontroller chips [7]. In general, students acquired more knowledge and skills when they take an active role in both practical and theoretical learning. But, any student who involved passively in both lecture courses will have great difficulty in learning embedded system designing and microcontroller programming materials [8]. Therefore, microcontroller system course should be organized in such a way that students will obtain not only a purely theoretical experience, but include adequate practical understanding of the topics lectured. By this approach and student start learning microcontroller system at early period of their academic studies with significance hands-on laboratory experiments will assure the best high-tech way of understanding microcontroller system course with comfort. This will complement the theoretical knowledge of the learner; it will provide opportunity to modern teaching practices, international project collaboration and cooperative learning processes [9-11].

The systematic process of hands-on laboratory work is to build confidence in students, improve their skills, self-reliance, and creative thinking on embedded system design and development with the use of microcontroller chips [12]. The Criteria for selection of microcontroller in any embedded system are as follows:

- The computing requirements of task cost effectively based on
- Speed of system operation.
- Packaging of the system.
- System power consumption.
- Memory capacity (RAM and ROM on chip).
- Number of I/O pins and on-chip timers.
- Availability of software development tools such as compiler, assembler and debugger [13-15].

2. RELATED WORK

There exists numerous works in literatures that involve the design and development of laboratory project-based microcontroller and embedded systems. A project-based embedded systems design course using a reconfigurable SoC platform was put forward by authors in [17]. A laboratory challenges associated with embedded system was solved, and variety applications were employed ranging from little unmanned aerial vehicle control to image processing. The students are able to learn and gain experience based on configuration of peripherals without circuit design and simulation knowledge of an embedded system, and also is an elective course in the curriculum. A research in [3] focuses on design and development of a project-based embedded system laboratory using LPC1768. These help students in learning building block of embedded system with sequential increase in hardware design and it cover just one third semester laboratory course.

With this research, it does not cover microcontroller concepts and embedded circuit simulation. A details demonstration of low-cost embedded system laboratory was design and developed using TI MSP430 [18]. The significant feature of experiments carried out here is that each hands-on lab was demonstrated with both hardware description and software development. The shortcoming of the proposition of this research is personally designed and developed for their training and consumption. Therefore, it will difficult for others to adopt the system for their research and experiment hands-on lab embedded system development in their regions. Embedded systems home experimentation is a research effort by [19]. He gave pave way and complimentary approach on embedded system development laboratory exercises, by increasing the number of workplaces in the lab, using simulators instead of real embedded hardware and provides means for distance learning of hands-on lab exercises. Finally, authors in [20] put forward microcontroller system from concept to the printed board. In this work, microcontroller system was used for experiment analog signal to digital signal conversion from the simulated circuit to the design of printed board layout. Therefore, the system was developed and tested in open loop and closed loop to improve students' skills in microcontroller systems.

Many of the above review work are embedded system project-based design and developments. In this paper, we try to simplify the problem in embedded system design from microcontroller project based circuit design, circuit simulation and development concept. This is to complete in-depth hands-on lab exercises and improve the learner's skills from simple embedded system design to the robotic system.

3. Hardware, Software System Description And Their Architecture

3.1 HARDWARE SYSTEM DESCRIPTION AND ITS ARCHITECTURE

In this paper, the hardware systems involved to programming the microcontroller chip are PICKit3 programmer, BK300 PIC development board, PIC16F887 chip, USB extension cord, personal computer system, and some electronic components. The hardware system was powered and setup by the following principles.

- Interface BK300 PIC development board to PCs using USB cable,
- Attach the PICKit3 programmer to development board using jumper selector cable,
- Interface PICKit3 programmer to PCs using USB cable,
- Insert PIC16F887 microcontroller chip on to the 40 pins ZIP socket located on BK300 PIC development board and pull down the notch to hold the chip firmly.
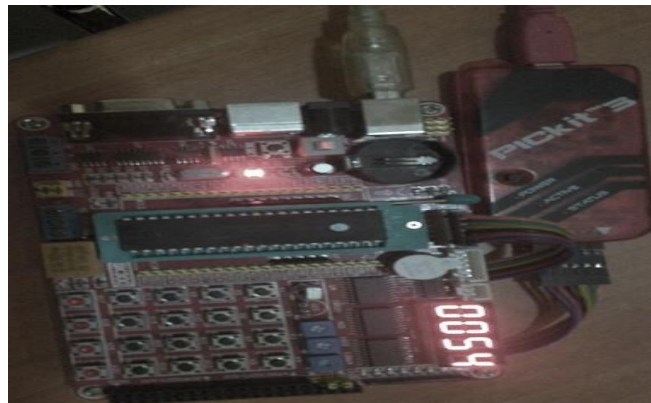- Then, press the switch button SW1 to power the board.

Figure 1. Snapshot of hardware system (BK300 development board, PIC16F887 chip and PICKit 3 programmer)
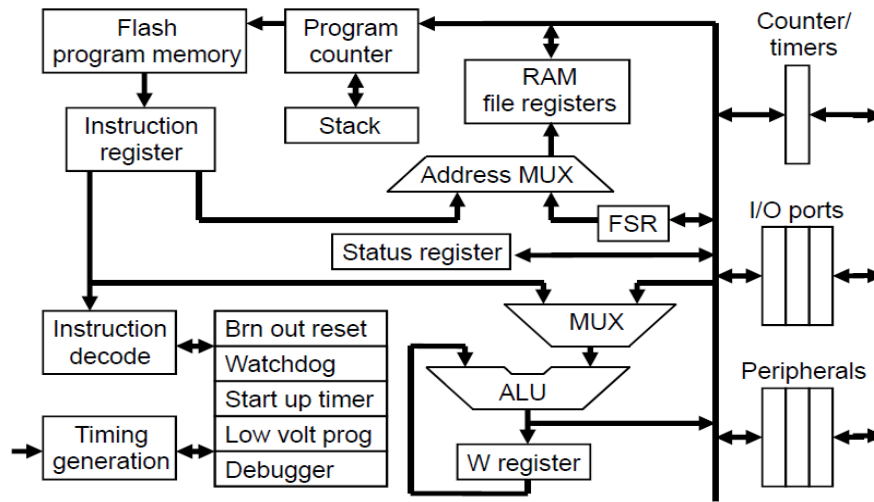
Figure 2. Block diagram depict internal architecture of microcontroller chip (PIC16F887)

3.2 SOFTWARE SYSTEM DESCRIPTION AND THEIR ARCHITECTURE

The compiler used for the program code and simulation in this project is mikroC Pro for PIC software, proteus 8.0 (VSM) and the PICKit3 software debugger. Details of software use with program code compilation and development stages discuss in this paper are presented in block diagram and flowchart shown in the figure 3 and 4 respectively.
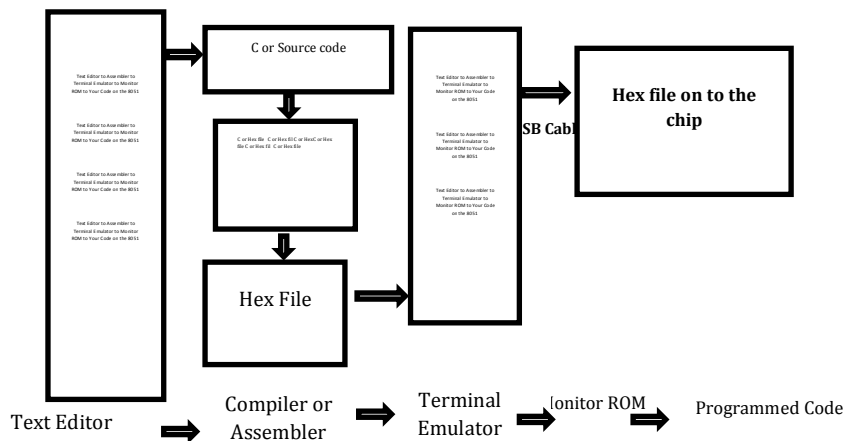

Figure 3: Block Diagram of Program Code Development Stages
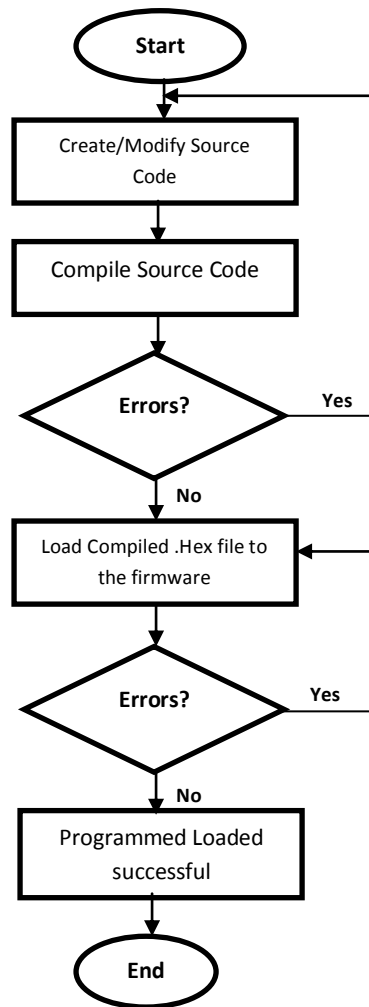
Figure 4: Flowchart of Program Code Development Stages

3.3 DETAILS OF PROGRAMMING PROCEDURE AND SIMULATION USING MIKROC PRO PIC SOFTWARE

Here are the stages involved in the programming procedure and simulation program overviews as existing in this paper.

3.3.1 Stage One: System Configuration

This is to create a New Project File, writing the source code and compile the code to .Hex file using mikroC PRO for PIC software (Text Editor). Therefore, this stage involved the following configuration setting.

- Project name and optional description,
- Target device,
- Device flags (config word),
- Device clock,
- List of the project source files with paths such as header files (*.h), binary files (*.mcl), image files, and other files.

To complete stage one configuration, here are the steps.

Step 1: From the main menu bar, click on Project › New Project › click on Next button, or by click on the New Project Icon [icon] from Project Toolbar.

Figure 5: Screen short of a new project create

Step 2: On the project setting, the followings task will be perform;

Select the Project Name, Project folder, the device name and the device clock in MHz from the each drop-down list, for instance (LED Blink, C:\Users\Desktop\Computer Lab work, P16F887 and 4.000000MHz respectively) and click on the Next button.
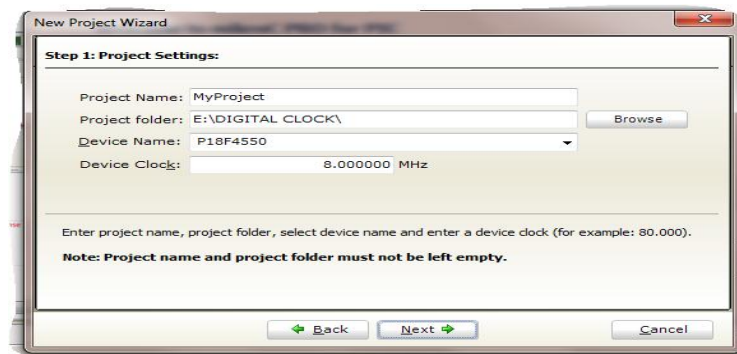
Figure 6: Screen short of the project setting 2

Step 3: Select the file you want to add to the file and click on Next button.
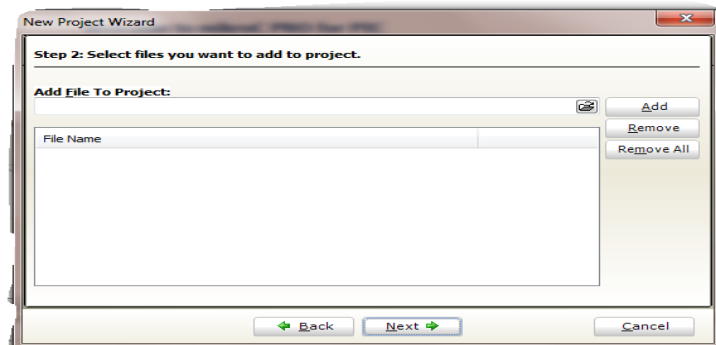

Figure 7: Screen short of ad file to project

Step 4: Select the initial state for library manager. Hint; select all by default is recommended for the beginners.
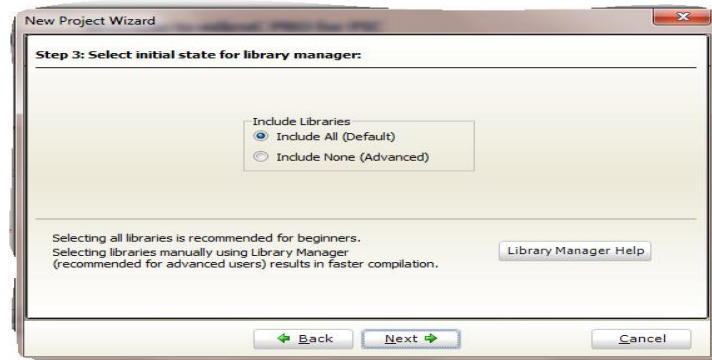

Figure 8: Screen short of setting library manager

Step 5: Check the edit projects window button to set configurations bit, and click on Finish button to be successful creating project file and folder.
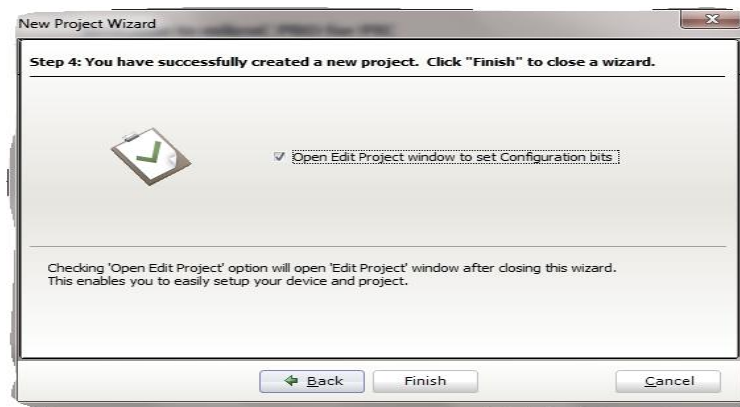

Figure 9: Screen short of setting configuration bit

Step 6: Here, you can finally edit the MCU pins configuration to either enable or disable by select each drop-down list level and click OK button to finalize settings.
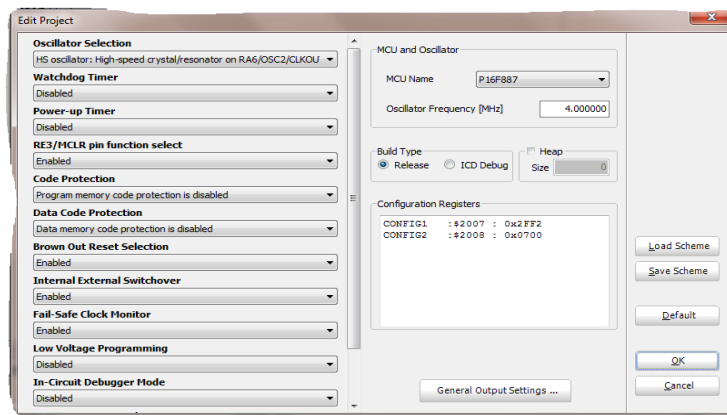

Figure 10: Screen short of configuring MCU pins

3.3.2 STAGE TWO: SOFTWARE DEVELOPMENT

This is the second stages involved in the programming procedure and simulation program, from the source code design to an object code (.hex file) development for hardware programming using software compiler or assembler tools. The compiler interface (mikroC code editor) is shown in the figure 12; it encompasses main menu, tool bar, cold folding, code assistance and parameter assistance etc. To accomplish this stage
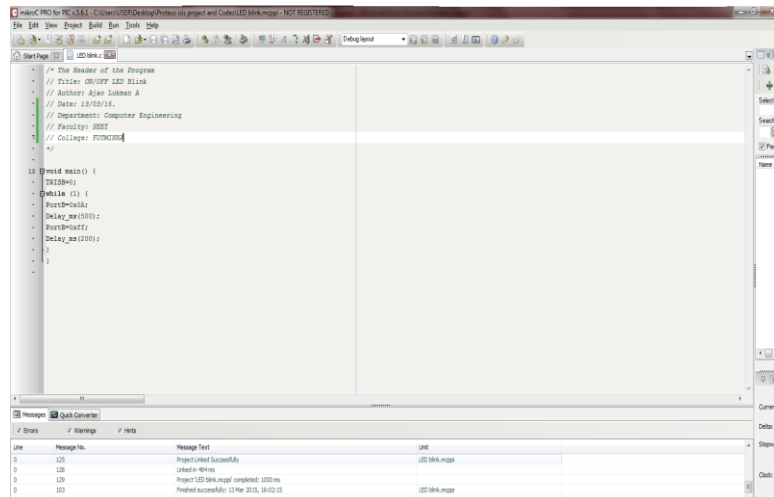

Figure 11: Screen shot of MicroC PRO for PIC features

3.3.3 Stage Three: Software Design and Implementation

This is the final stages where the hardware system and software program are been implemented to make a complete functioning of an embedded system. The development board is connected to the PICKit3 programmer and finally interfaces with PC (Terminal Emulator, Monitor ROM, and Programmed Code). Here are the simple procedures to accomplished software design implementation on microcontroller chip.

- a.Connect PICKit 3 programmer to the PC, when power and the active light display, then you open PICKit 3 programmer software installed.
- b.On the main menu of PICKit 3 software, click on Device family to select MCU architecture (Midrange)
- c.On the Device Configuration, select the MCU you want to download the .hex file on (PIC16F887).
- d.Click on File, point to Import to import object code (.Hex file).
- e.Click on Programmer on the menu bar, and select Write Device to erase the content of MCU and finally deploy .hex code to the device for programmed.
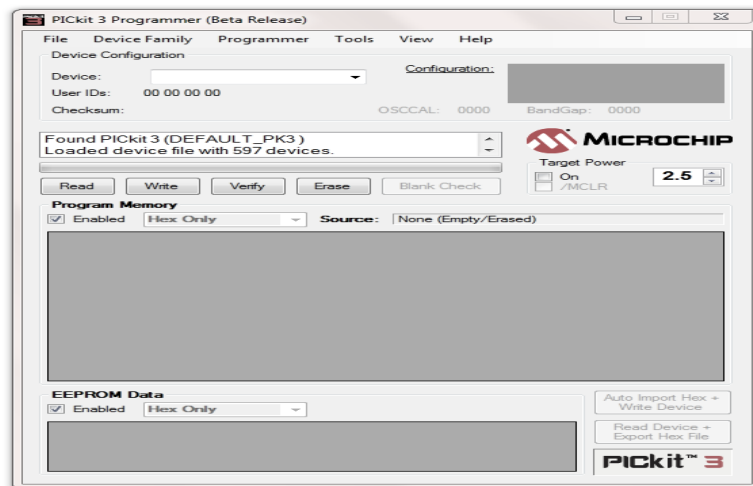
Figure 12: Screen shot of PICKit 3 software interface

4. DESCRIPTION OF CIRCUIT SIMULATION EXPERIMENTS AND INDIVIDUAL HAND-ON LABORATORY EXERCISES

In this paper, four different experiments are demonstrated with the use of available hardware and software tools. The circuit simulation diagram for each experiment is design in proteus 8.0 software and individual expertise hand-on laboratory experiment are carried out with the use of hardware system described in 3.1, which all are shown together in each experiment performed. We started the experiment with ON/OFF LED display, LCD module message display, Seven-segment count-down timer, and temperature sensor with LCD module interface. With the following experiments performed and demonstrated in this paper, it aid theoretical understanding of students in learning microcontroller system as a course, improve their practical skills on embedded system design and interfacing microcontroller chip with different electronic components. It helps them to configure it for laboratory research, industry applications, world consumption and development [16].

4.1. EXPERIMENT 1: PARALLEL I/O ON/OFF LED INTERFACING

In the first experiment, on/off LED Interfacing; Eight LED was used and arranged in parallel before interface with the PIC16F887 using Port B (RB0-RB), pin 33-40 of microcontroller chip. The following are the steps involved in programmed the PIC16F887chip using both hardware and software tools to accomplish the experiments and demonstration.

- Connect the USB cable between the PC and the BK300 PIC development board for the power.
- Connect the PICKit3 programmer to the ICSP port J7.
- Connect the USB cable between the PC and the PICKit3 programmer for the power.
- Press down the power switch at SW2 to power the board.
- Place the PIC 16F887 chip on the 40-pins ZIF socket of the board. If it is already in place, then proceed to the next step.
- Run the MikroC PRO for PIC complier on your PC. Create a new project and folder with the name 'LED Blink'. Choose PIC 16F887 as your target device and

4MHz as the clock frequency. Write the code on the code editor window and click build to compile to the project (.HEX) code.

- Run the PICKit 3 programmer GUI on the PC. On the device family select Midrange, Under File, click 'Import Hex' and navigate to the project folder. Load in the project (.HEX) Code. Click on 'Write' to erase and program your chip.
- Observe the blinking of 8 LEDs on the board ON/OFF at the same time with accurate interval, and display continuously based on the instruction code.
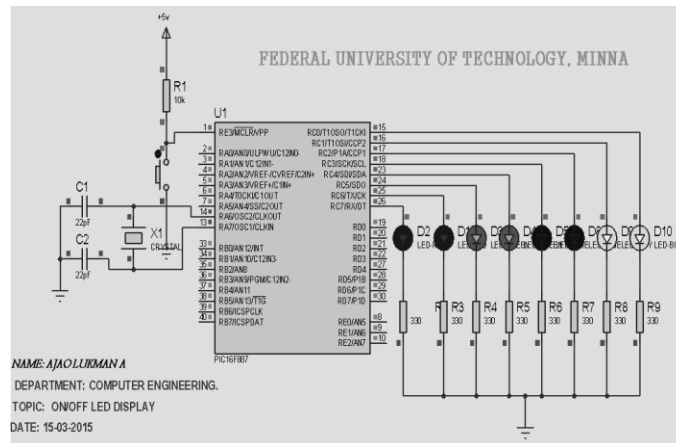

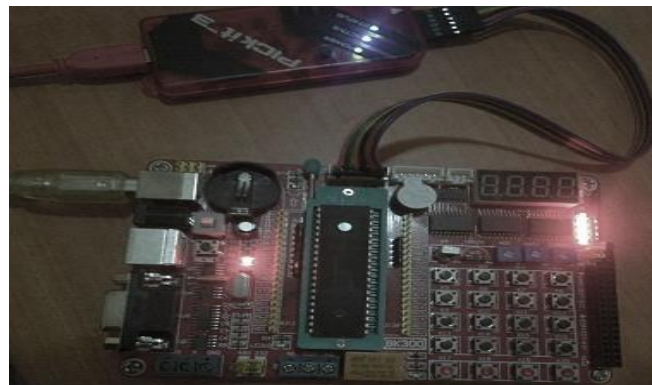Fig 13a: Circuit simulation diagram of I/O LED interfacing with PIC16F887

Fig 13b: Photograph of I/O LED interfacing with PIC16F887 using BK300 Board

4.2 EXPERIMENT 2: LCD MODULE INTERFACING

At this point of experiment 2, the LCD and LED are used as an output device which connected to Port A and Port C of the controller system respectively. To program the chip for this experiment 2, follow the same steps as itemized in the experiment 1, from step i-vii. You just need to change your code at step (vi) and compile it to generate .Hex file. This procedure is applicable to program PIC16F887 chip for further experiments 3 and 4 presented in this paper respectively.

Observe the message display on the 16x2 LCD modules "**FUTMINNA**" "**COMPUTER ENGR. LAB**". This message displays continuously until any external alteration or power failure, and the eight LED also glow simultaneously.
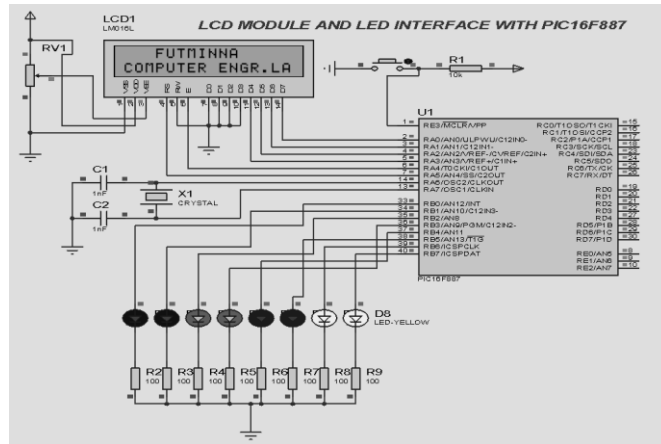
Fig 14a: Circuit simulation diagram of I/O LED and LCD module interfacing with PIC16F887
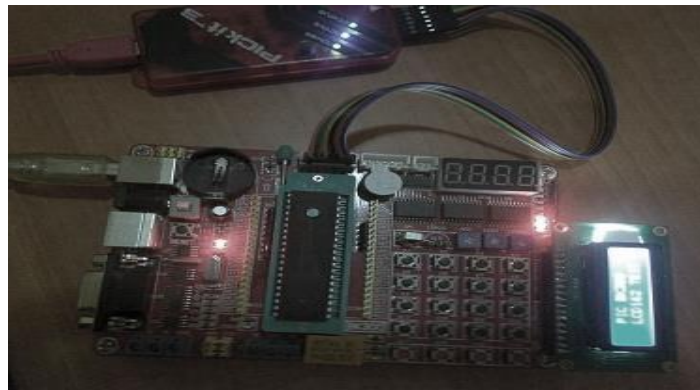

Fig 14b: Photograph of the I/O LED and LCD module interfacing with PIC16F887 using BK300 Board

4.3 EXPERIMENT 3: FOUR-DIGITS MULTIPLEXED SEVEN-SEGMENT DISPLAY INTERFACING

In this experiment, four-digit multiplexed seven segment display is interfaced with PIC16F887 chip using Port B to perform simple operation known as count-down timer (0-99) at a specified interval.
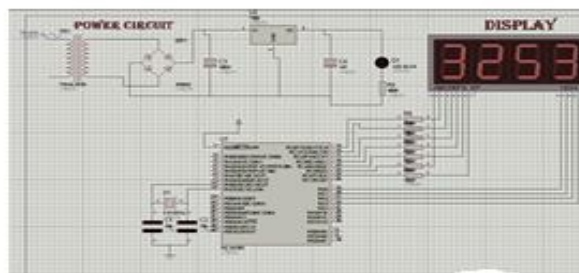

Fig 15a: Circuit simulation diagram of four-digit multiplexed 7-segment count down timer interfacing with PIC16F887
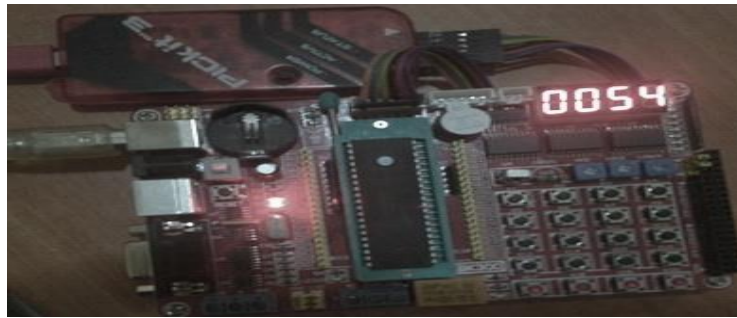
Fig 15b: Photograph of the 7-segment display count up timer interfacing with PIC16F887usin BK300 board

4.4 EXPERIMENT 4: TEMPERATURE SENSOR AND LCD INTERFACE WITH PIC16F887

The application of the experiment 4 is to measure the value of room temperature using digital thermometer (LM35). The LCD and LM35 temperature sensor is used and connected to Port B and Port A pin 4 respectively. Then, the measure value in ^0C is converted to the equivalent value in ^0F and both values are display on the LCD screen.

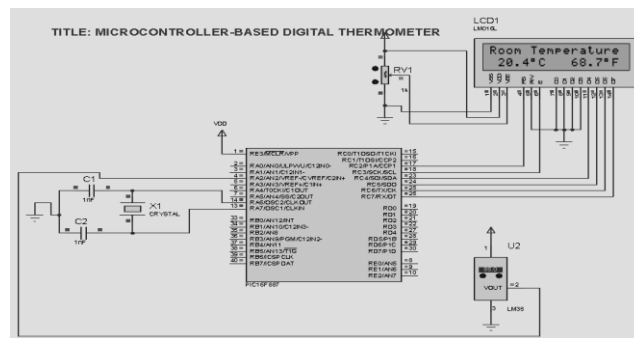Fig 18a: Circuit simulation diagram of temperature sensor and LCD interface with PIC16F887
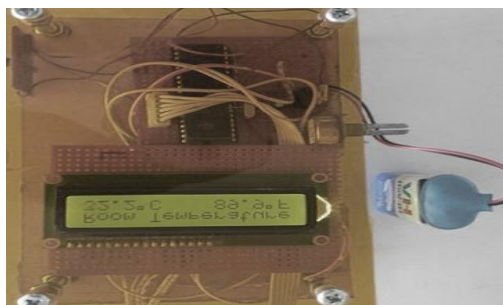

Fig 5a: Photograph of the final package of Room Temperature sensor LM35 interface with PIC16F887

5. COMPARATIVE STUDY WITH SIMILAR WORKS IN LITERATURE

The development of project-based microcontroller system laboratory using Bk300 development board with PIC16F887 chip was compared with other existing similar works in literature from

perspective such as features, functions, cost and architecture of the development of low-cost embedded system like Aruna et al., 2013, Naveen et al., 2013, Daniel et al., 2015 and the concept of microcontroller system development like Wilfried et al., 2003 and Edgard et al., 2010. From the comparative study, it can be concluded that Bk300 development board and PIC16F887 chip used has better merits over others existing similar works as described in the table i.

Table I: Comparison of the BK300 development board and PIC16F887 with adopted works in literature.

S/N	Parameter of Comparison	Aruna et al., 2013, Naveen et al., 2013 and Daniel et al., 2015.	Project-based microcontroller system laboratory using Bk300 development board with PIC16F887 chip
1.	Described Features	This laboratory development board has limited features and support little applications as described in authors' paper.	This has common features of most laboratory development board with additional structures like electronic clock experiment (using DS1302 Electronic Clock), buzzer (MCU sound, music play), 12864 LCD interface (characters display, image display and man-machine interface), infrared receiver interface, PS2 keyboard interface and ICSP simulation programming interface.
2.	Functions	As described in the authors work (Aruna et al., 2013, Naveen et al., 2013), the boards are suitable to carried out some laboratory experiments exercises based on their features but limited.	This Bk300 development board laboratory tools has many features and support applications more than the existing boards in the Aruna et al., 2013, Naveen et al., 2013, and also as mentioned in parameter 1 of the table I.
3.	Cost	Board considered in these works are expensive than the proposed board in our contribution. As NXP LPC1768 is $49 and MSP430 LaunchPad and the development board is over $45.	This tools is considered to be less expensive compared to the existing one ($33.69)
4.	Microcontroller Architecture	In the authors work (Aruna et al., 2013, Naveen et al., 2013), the microcontroller used for programming are expensive, scarce and not generally accessible to the users compared to the BK300 board. Each pin has little functions compared to the microchip PIC16F887.	PIC16F887 is the microcontroller used for programming from microchip company. This chip is considered to be less expensive ($2.8), compact, and easy to study. It's generally available in the market for the user consumption and applications. Each pin is configured to perform more than one function in the capacity of applications.

6. CONCLUSION

The hands-on lab experience microcontroller-based experiment presented in this paper will increase the students' learning skills, gives self confidence in any attempt of microcontroller simulation and embedded system design projects. It will guide students to write an embedded system program from simple to the complex program code, and build different simple stand-alone systems with complex hardware circuit for industry consumption and research applications.

REFERENCES

[1]   Crespo. A, Vila. J, Blanes. A and Ripoll. I "Real Time Education in a Control Engineering Curriculum", 3rd IEEE Real Time Systems Education Workshop. pp. 112-116 1998.

[2]   Ricks K. G. Jackson D. J. and Stapleton W. A., "Incorporating Embedded programming skills in to an ECE Curriculum", SIGBED Rev., Vol .4, No.1, pp. 17-26, Jan. 2010.

[3]   Aruna K, Raghavendra R. K. "Design and Development of a Project-Based Embedded System Laboratory Using LPC1768", American Journal of Embedded Systems and Applications. Vol. 1, No. 2, pp. 46-53 2013.

[4]   Bachnak R. "Teaching Microcontrollers with Hands-on Hardware Experiments", Journal of Computing Sciences in Colleges, 20(4):207-213, 2005.

[5]   Jean-Samuel Chenard, Zeljko Zilic, Milos prokic, " A Laboratory Setup and Teaching Methodology for Wireless and Mobile Embedded Systems", "IEEE Transactions on Education", vol.51, No.3, pp. 378-384, August 2008.

[6]   Yehezkel C. Yurcik W., Pearson M. and Armstrong, D. "Three Simulator Tools for Teaching Computer Architecture: EasyCPU, Little Man Computer, and RTLSim," ACM Journal of Educational Resources in Computing, Vol. 1(4), pp. 60-80, 2001.

[7]   Jiang Xiaoluo and Li Han. "CDIO-Based Embedded Systems Training Mode in Graduate Teaching", 5th International Conference on Distance Learning and Education, vol. 12, pp. 78-82, 2011.

[8]   Carpinelli J. D. and Jaramillo F. "Simulation Tools for Digital Design and Computer Organization and Architecture", 31st ASEE/IEEE Frontiers in Education Conference, Vol. S3C, pp. 1-5, 2001.

[9]   Bruce J. W. Harden J. C. and Reese R. B. "Cooperative and Progressive Design Experience for Embedded Systems", IEEE Transactions on Education, vol.47, no.1, pp.83-92, Feb. 2004.

[10]   Davcev D, Stojkoska B., Kalajdziski S., and Trivodaliev "Project Based Learning of Embedded Systems", Proceedings of the 2nd WSEAS Telecommunications, 2008.

[11   ]Nooshabadi S. and Garside J., "Modernization of Teaching in Embedded Systems Design–An International Collaborative Project", IEEE Transactions on Education, vol. 49, no. 2, pp. 254-262, May, 2006.

[12]   Silik. C, Nagvaara, P. Taskin. B, "A Microcontroller Based Embedded System design Course with PSoC3", "IEEE International Conference on Microelectronic Systems Education (MSE), pp.28-31, June. 2-3, 2013.

[13]   http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dv164131

[14]   http://ww1.microchip.com/downloads/en/DeviceDoc/51795B.pdf

[15]   http://www.mikroe.com/chapters/view/14/chapter-1-world-of-microcontrollers/.

[16]   Wilfried Elmenreich, Christian Trodhandl and Bettina weiss, "Embedded System Home Experimentation", Proceedings of the 2003 IEEE International Conference on Microelectronic Systems Education (MSE" 03).

[17]   Daniel Roggow, Paul Uhing, Phillip Jones, and Joseph Zambreno "A Project-Based Embedded Systems Design Course Using a Reconfigurable SoC Platform", Proceedings of the 2015 IEEE International Conference 978-1-4799-9915-6/15

[18]   Naveen Kumar Uttarkar, Raghavendra Rao Kanchi "Design and Development of a Low-Cost Embedded System Laboratory using TI MSP430 LaunchPad" American Journal of Embedded Systems and Applications; 1(2): 37-45, November 20, 2013

[19]   http://www.ecs.tuwien.ac.at/Projects/SCD

[20]   Edgard J. Guimarães and Neusa M. F. Oliveira "Microcontroller System: From Concept to Printed Board" 40th ASEE/IEEE Frontiers in Education Conference T3C-1, October 27 - 30, 2010.