International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 20 (2017) pp. 9636-9643

RFID based Technology: university Student attendance system, Design and Implementation

Dr. Tarik H. Baldawi

Department of Electrical Engineering, Applied Science Private University, Amman, Jordan

Orcid: 0000- 0001-7315-9088

Abstract

RFID (1) is a Dedicated Short Range Communication (DSRC) technology. The term RFID is used to describe various technologies that use radio waves to automatically identify people or objects. RFID technology is similar to the bar code identification systems we see in retail stores everyday; however one big difference between RFID and bar code technology RFID does not rely on the line-of-sight reading that bar code scanning requires to work.

The objective of this work is design and implement a student attendance system based on the RF-ID (Radio Frequency Identification Technology), the student will have an RF-ID Card that will allow him to proof the attendance in each subject that he is register in lecture by lecture.

Keywords: RFID identification, PIC applications.

1. System BLOCK DIAGRAM: Fig (1.1) shows the main parts of this work as a block diagram.

The main components used in this work are the following :

1. PIC microcontroller (PIC16F877A)
2. Serial interface chip(this chip will handle the voltage levels between the microcontroller and the PC)
3. RF ID-reader
4. LCD that will be used to display the user number and any desired notes .
5. keypad
6. buzzer

Software needed:

1) Mikrobasic compiler , this will be used to write the code for the microcontroller.
2) visual basic.NET , this software will be used to write the program that will receive the data from the microcontroller and display it on a table.

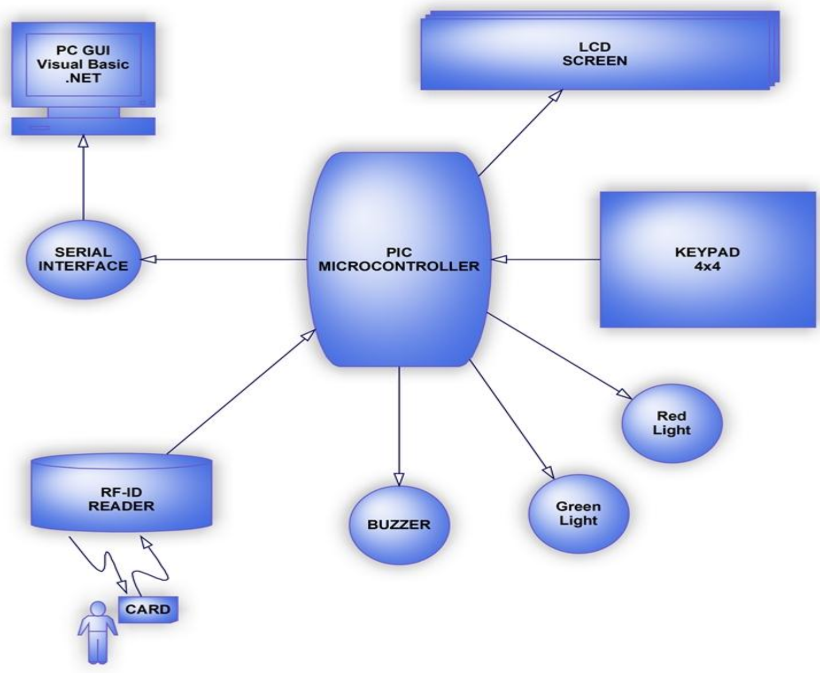

Figure 1.1: System Block Diagram

2. The RFID Technology, an overview, (2) ,(3) ,(4), (5), (6) With RFID, the electromagnetic or electrostatic coupling in the RF (radio frequency) portion of the electromagnetic spectrum is used to transmit signals. An RFID system consists of an antenna and a transceiver, which read the radio frequency and transfer the information to a processing device (reader) and a transponder, or RF tag, which contains the RF circuitry and information to be transmitted. The antenna provides the means for the integrated circuit to transmit its information to the reader that converts the radio waves reflected back from the RFID tag into digital information that can then be passed on to computers that can analyze the data.

In RFID systems, the tags that hold the data are broken down into two different types. Passive tags use the radio frequency from the reader to transmit their signal. Passive tags will generally have their data permanently burned into the tag when it is made, although some can be rewritten. Active tags are much more sophisticated and have on-board battery for power to transmit their data signal over a greater distance and power random access memory (RAM) giving them the ability to store up to 32,000 bytes of data.

RFID Frequencies Much like tuning in to your favorite radio station, RFID tags and readers must be tuned into the same frequency to enable communications. RFID systems can use a variety of frequencies to communicate, but because radio waves work and act differently at different frequencies, a frequency for a specific RFID system is often dependant on its application. High frequency RFID systems (850 MHz to 950 MHz and 2.4 GHz to 2.5 GHz) offer transmission ranges of more than 90 feet, although wavelengths in the 2.4 GHz range are absorbed by water, which includes the human body, and therefore has limitations.

RFID Tags Past and Present

RFID technology has been used since 1970, but until recently, it was too expensive to be used on a large scale. Originally, RFID tags were used to track large items, like cows, railroad cars and airline luggage, that were shipped over long distances, These original tags, called inductively coupled RFID tags, were complex systems of metal coils, antennae and glass were used.

Inductively coupled RFID tags were powered by a magnetic field generated by the RFID reader. Electrical current has an electrical component and a magnetic component, it is electromagnetic. Because of this, a magnetic field is created by electricity, and an electrical current can be created with a magnetic field. The name "inductively coupled" comes from this process, the magnetic field inducts a current in the wire.

Capacitive coupled tags

The next step was an attempt to lower the technology's cost. These were meant to be disposable tags that could be applied to less expensive merchandise and made as universal as bar codes. Capacitively coupled tags used conductive carbon ink instead of metal coils to transmit data. The ink was printed on paper labels and scanned by readers. Motorola's BiStatix RFID tags were the pioneers in this technology. They used a silicon chip that was only 3mm wide and stored 96 bits of information. This technology didn't catch on with retailers, and BiStatix was shut down in 2001

Newer innovations in the RFID industry include active, semi-active, and passive RFID tags. These tags can store up to 2 kilobytes of data and are composed of a microchip, antenna, and, in the case of active and semi-passive tags, a battery.

At a basic level, each tag works in the same way:

- Data stored within an RFID tag's microchip waits to be read.
- The tag's antenna receives electromagnetic energy from an RFID reader's antenna.
- Using power from its internal battery or power harvested from the reader's electromagnetic field, the tag sends radio waves back to the reader.
- The reader picks up the tag's radio waves and interprets the frequencies as meaningful data. Inductively coupled and capacitively coupled RFID tags aren't used as commonly today because they are expensive and bulky. In the next section, we'll learn more about active, semi-passive and passive RFID tags.

Active, Semi-passive and Passive RFID Tags

Active, semi-passive and passive RFID tags are making RFID technology more accessible and prominent in our world. These tags are less expensive to produce, and they can be made small enough to fit on almost any product.

Active and semi-passive RFID tags use internal batteries to power their circuits. An active tag also uses its battery to broadcast radio waves to a reader, whereas a semi-passive tag relies on the reader to supply its power for broadcasting. Because these tags contain more hardware than passive RFID tags, they are more expensive. Active and semi-passive tags are reserved for costly items that are read over greater distances -- they broadcast high frequencies from 850 to 950 MHz that can be read 100 feet or more away. If it is necessary to read the tags from even farther away, additional batteries can boost a tag's range to over 300 feet (100 meters)

Passive RFID tags rely entirely on the reader as their power source. These tags are read up to 20 feet away, and they have lower production costs, meaning that they can be applied to less expensive merchandise. These tags are manufactured to be disposable, along with the disposable consumer goods on which they are placed. Whereas a railway car would have an active RFID tag, a bottle of shampoo would have a passive tag. Another factor that influences the cost of RFID tags is data storage. There are three storage types: read-write, read-only and WORM (write once, read many). A read-write tag's data can be added to or overwritten. Read-only tags cannot be added to or overwritten -- they contain only the data that is stored in them when they were made. WORM tags can have additional data (like another serial number) added once, but they cannot be overwritten.

Most passive RFID tags cost between 7 and 20 cents each Active and semi-passive tags are more expensive, and RFID manufacturers typically do not quote prices for these tags without first determining their range, storage type and quantity. The RFID industry's goal is to get the cost of a passive RFID tag down to 5 cents each once more merchandisers adopt it. In the next section, we'll learn how this technology could be used to create a global system of RFID tags that link to the Internet

HARDWARE IMPLEMENTATION

1- PIC microcontroller
2- RFID reader
3- Liquid crystal display
4- Keypad
5- Serial interface IC

Microcontroller architecture: Memory unit (7)

Memory is part of the microcontroller whose function is to store data. The easiest way to explain it is to describe it as one big closet with lots of drawers. If we suppose that we marked the drawers in such a way that they cannot be confused, any of their contents will then be easily accessible. It is enough to know the designation of the drawer and so we will know its contents for sure, as shown in figure 3.1

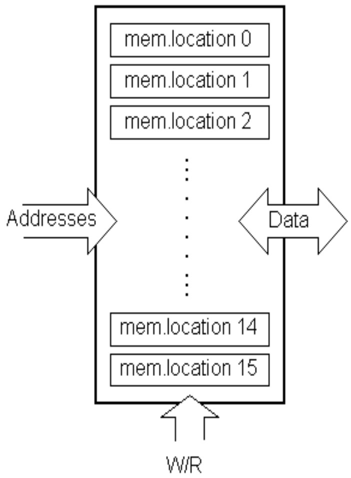

Figure 3.1: Memory unit

Memory components are exactly like that. For a certain input we get the contents of a certain addressed memory location and that's all. Two new concepts are brought to us: addressing and memory location. Memory consists of all memory locations, and addressing is nothing but selecting one of them. This means that we need to select the desired memory location on one hand, and on the other hand we need to wait for the contents of that location. Besides reading from a memory location, memory must also provide for writing onto it. This is done by supplying an additional line called control line. We will designate this line as R/W (read/write). Control line is used in the following way: if r/w=1, reading is done, and if opposite is true then writing is done on the memory location. Memory is the first element, and we need a few operation of our microcontroller.

Central processing unit (CPU)

Let us add 3 more memory locations to a specific block that will have a built in capability to multiply, divide, subtract, and move its contents from one memory location onto another. The part we just added in is called "central processing unit" (CPU). Its memory locations are called registers, as shown in figure 3.2

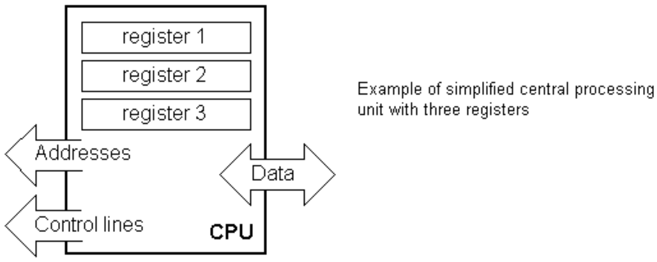

Figure 3.2: CPU

Registers are therefore memory locations whose role is to help with performing various mathematical operations or any other operations with data wherever data can be found. Look at the current situation. We have two independent entities (memory and CPU), which are interconnected, and thus any exchange of data is hindered, as well as its functionality. If, for example, we wish to add the contents of two memory locations and return the result again back to memory, we would need a connection between memory and CPU. Simply stated, we must have some "way" through data goes from one block to another.

Bus: Physically, it represents a group of 8, 16, or more wires. There are two types of buses: address and data bus. The first one consists of as many lines as the amount of memory we wish to address and the other one is as wide as data, in our case 8 bits or the connection line. First one serves to transmit address from CPU memory, and the second to connect all blocks inside the microcontroller, as shown in figure 3.3

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 20 (2017) pp. 9636-9643
© Research India Publications. http://www.ripublication.com

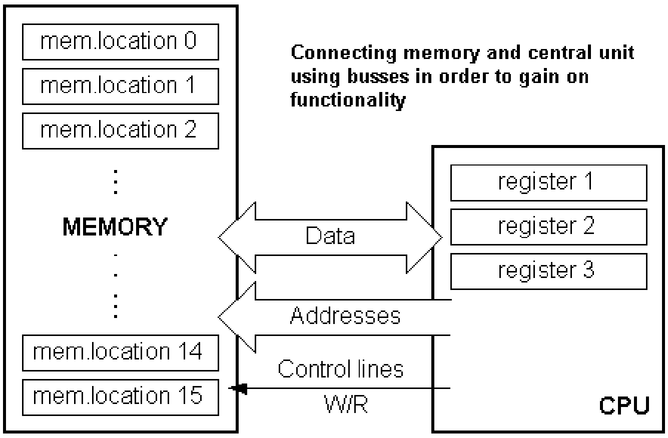

Figure 3.3: DATA Bus

As far as functionality, the situation has improved, but a new problem has also appeared: we have a unit that is capable of working by itself, but which does not have any contact with the outside world, or with us. In order to remove this deficiency, let's add a block that contains several memory locations whose one end is connected to the data bus, and the other has connection with the output lines on the microcontroller that can be seen as pins on the electronic component.

Input-output unit: Those locations we've just added are called "ports". There are several types of ports: input, output or bi-directional ports. When working with ports, first of all it is necessary to choose which port we need to work with, and then to send data to, or take it from the port, as shown in fig. 3.4. When working with it the port acts like a memory location. Something is simply being written into or read from it, and it could be noticed on the pins of the microcontroller.

Serial communication: Beside stated above we've added to the already existing unit the possibility of communication with an outside world. However, this way of communicating has its drawbacks. One of the basic drawbacks is the number of lines that need to be used in order to transfer data. What if it is being transferred to a distance of several kilometers? The number of lines times number of kilometers doesn't promise the economy of the project. It leaves us having to reduce the number of lines in such a way that we don't lessen its functionality. Suppose we are working with three lines only, and that one line is used for sending data, other for receiving, and the third one is used as a reference line for both the input and the output side. In order for this to work, we need to set the rules of exchange of data. These rules are called protocol. Protocol is therefore defined in advance so there wouldn't be any misunderstanding between the sides that are communicating with each other.

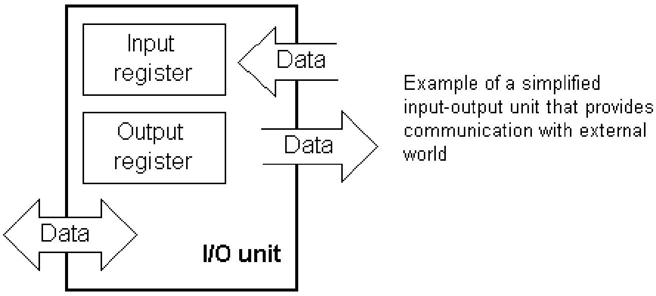

Figure 3.4: input –output unit

For example, if one man is speaking in French, and the other in English, it is highly unlikely that they will quickly and effectively understand each other. Let's suppose we have the following protocol. The logical unit "1" is set up on the transmitting line until transfer begins. Once the transfer starts, we lower the transmission line to logical "0" for a period of time (which we will designate as T), so the receiving side will know that it is receiving data, and so it will activate its mechanism for reception. Let's go back now to the transmission side and start putting logic zeros and ones onto the transmitter line in the order from a bit of the lowest value to a bit of the highest value. Let each bit stay on line for a time period which is equal to T, and in the end, or after the 8th bit, let us bring the logical unit "1" back on the line which will mark the end of the transmission of one data.

The protocol we've just described is called in professional literature NRZ (Non-Return to Zero), as shown in figure 3.5.

As we have separate lines for receiving and sending, it is possible to receive and send data (info.) at the same time. So-called full-duplex mode block that enables this way of communication is called a serial communication block. Unlike the parallel transmission, data moves here bit by bit, or in a series of bits what defines the term serial communication comes from.

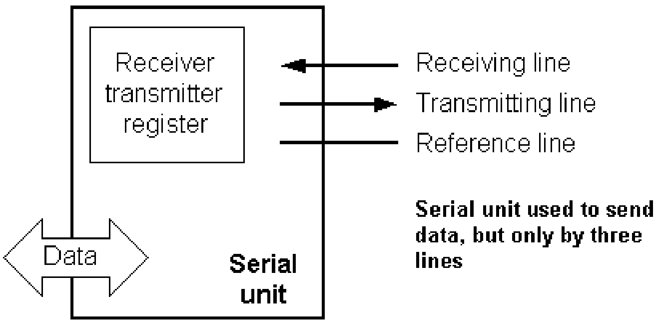

Figure 3.5: Serial unit

After the reception of data we need to read it from the receiving location and store it in memory as opposed to sending where the process is reversed. Data goes from memory through the bus to the sending location, and then to the receiving unit according to the protocol.

RF-ID reader RDM3600 (125 kHz)

Why the RF-ID reader for this project :

- requires 5V supply (it can be supplied from the same supply as PIC)
- 125kHz read frequency (compatible with most kinds of RF cards)
- EM4001 64-bit RFID tag compatible
- 9600bps TTL and RS232 output (can be connected directly to PIC microcontroller using the USART module).
- 100 mm read range (good reading distance). Description: RDM6300 125KHz card mini-module is designed for reading code from 125KHz/ 134.2KHz card compatible read-only tags and read/write card . It can be applied in office/home security, personal identification, access control, anti-forgery, interactive toy and production control systems etc.

RDM6300 Series non contact RFID module, use the advanced RF receiving circuit and embedded MCU design, combing with high efficient decoding algorithm, to finish the data receive of EM4100 and all the compatible cards. This is a very popular RFID card read module for microcontroller users, it can read 125K EM4100 series RFID card.

This module is easy to use, serial port connect to microcontroller MCU after power up, when RFID card enter into card reading range, this module will send the card number to microcontroller by UART You can further connect relay microcontroller shield to your microcontroller in order to enable the simple door keeper controlling system.

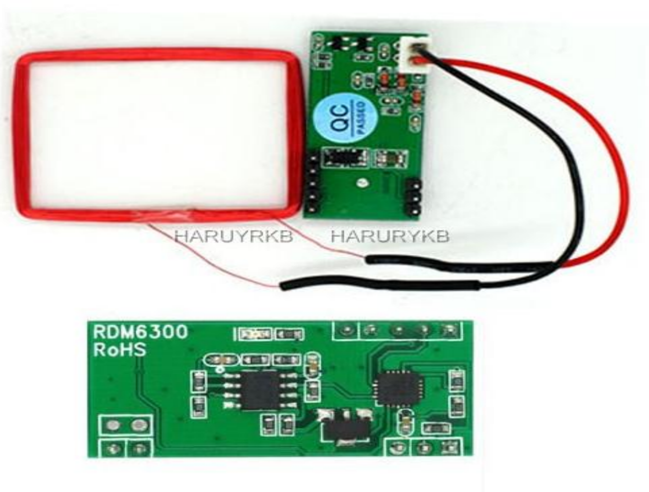

Figure 3.6: RF ID RDM3600

Features: 5V supply , 125kHz read frequency , EM/TK 4100, EM/TK4102, EM4200 or compatible 9600bps TTL and RS232 output , Magnetic stripe emulation output , 100mm read range serial interfaces (8) : In telecommunications, RS-232 (Recommended Standard 232) is a standard for serial binary data signals connecting between a DTE (Data terminal equipment) and a DCE (Data Circuit-terminating Equipment). It is commonly used in computer serial ports. A similar ITU-T standard is V.24.

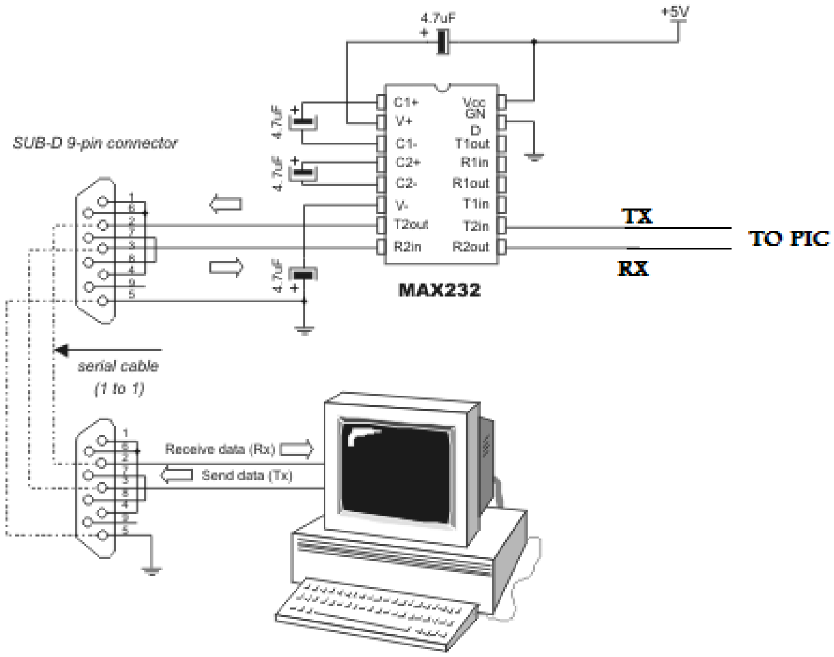

Figure 3.7

Project Schematic Diagram

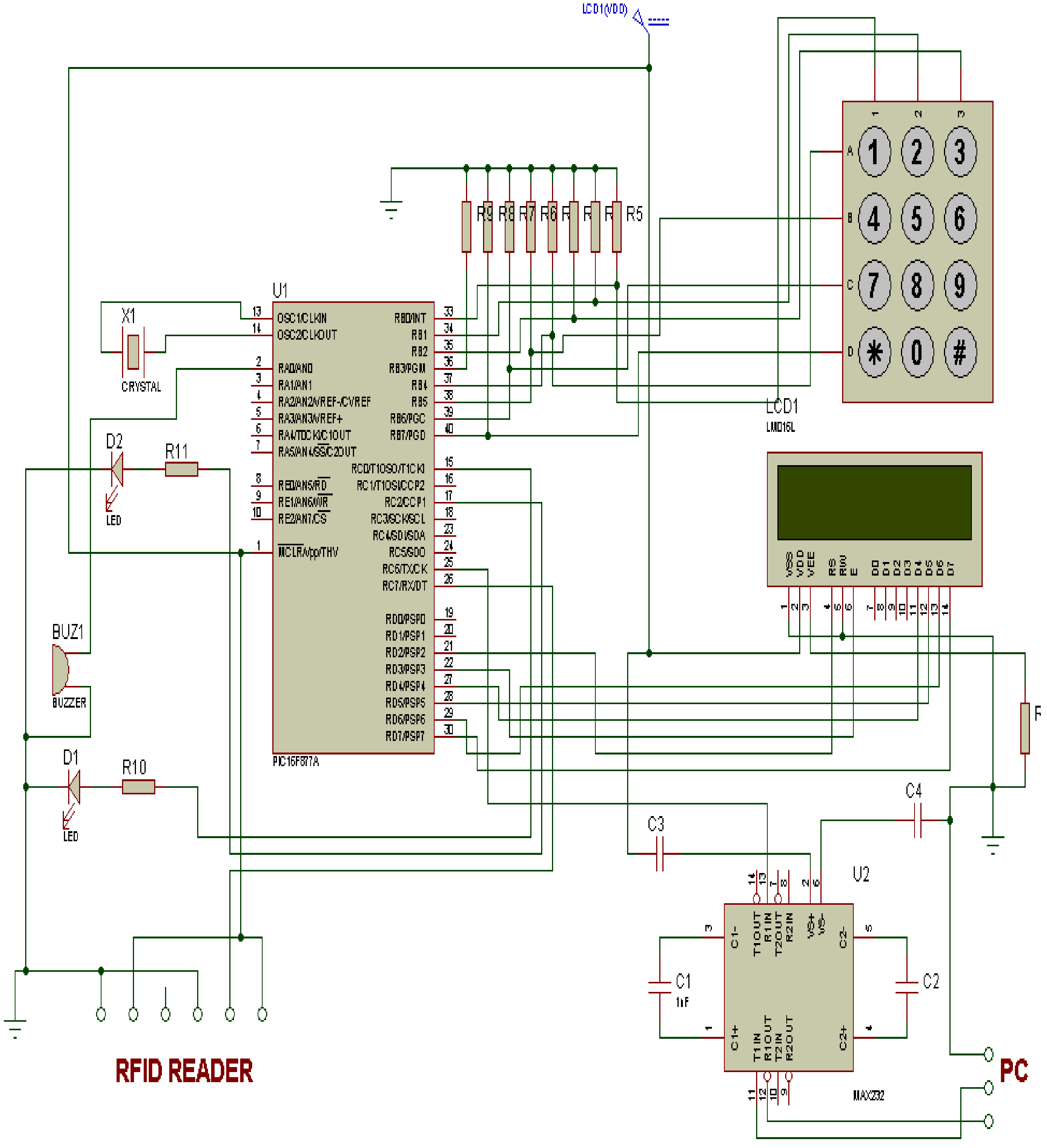

Figure (3.8) Project Schematic Diagram

SOFTWARE IMPLEMENTATION

This section will introduce the software part in this work, this project is divided into two main parts , PIC programming and visual basic programming

Stages to Program a PIC Microcontroller:

STAGE 1: In order to program a PIC Microcontroller ,the programmer must be able to use LOGICATOR software, using a computer, the flow chart approach is used. The flow chart approach means that there is no need to learn difficult and complex programming languages.

STAGE 2: Once the program has been tested using a compiler on a computer, it can be downloaded to the PROGRAMMER which is attached to the computers "common port". Any mistakes to the program are usually found during testing on the computer; however, the PIC can be reprogrammed easily. It is not thrown away as it can be reused any number of times.

STAGE 3: The PIC Microcontroller (or PIC as it is usually known) is then taken out of the programmer and placed in the desired project board. The project is controlled by the downloaded program on the PIC's ROM.

Software

As mentioned earlier, the PIC microcontroller 16f876A was used in this work . In order to create the code for the PIC, a compiler called MikroBasic was used. MikroBasic is a rich development tool for PIC microcontrollers supplied by MikroElectronika . It is designed to provide the easiest solutions for developing applications for embedded systems, without compromising performance or control. MikroBasic allows you to quickly develop and deploy complex applications like writing BASIC source code using highly advanced Code Editor. The Code Editor features adjustable Syntax Highlighting, Code Assistant, Parameters Assistant, Auto Correct for common typos, and the Code Templates.

Features of MikroBasic

- MikroBasic allows to develop and deploy complex applications:
- Write BASIC source code using the built-in Code Editor (Code and Parameter Assistants, Syntax Highlighting, Auto Correct, Code Templates)
- Use the included MikroBasic libraries to dramatically speed up the development: data acquisition, memory, displays, conversions, communications… Practically all P12, P16, and P18 chips are supported.
- Monitor program structure, variables, and functions in the Code Explorer.
- Generate commented, human-readable assembly, and standard HEX compatible with all programmers.
- Inspect program flow and debug executable logic with the integrated Debugger.
- Get detailed reports and graphs: RAM and ROM map, code statistics, assembly listing, and calling tree.

After using the Code Editor in order to write the code, compiling the code will create a hex file meaning that the file is in hexadecimal format. This file is then transferred into the ROM of the PIC via the programmer described above using simple serial bit transference.

Built-in Routines : MikroBasic compiler provides a set of useful built-in utility functions. Built-in functions do not have any special requirements; they can be used in any part of this project. Some built-in routines are implemented as "inline"; i.e. code is generated in the place of the call, so the call doesn't count against the nested call limit

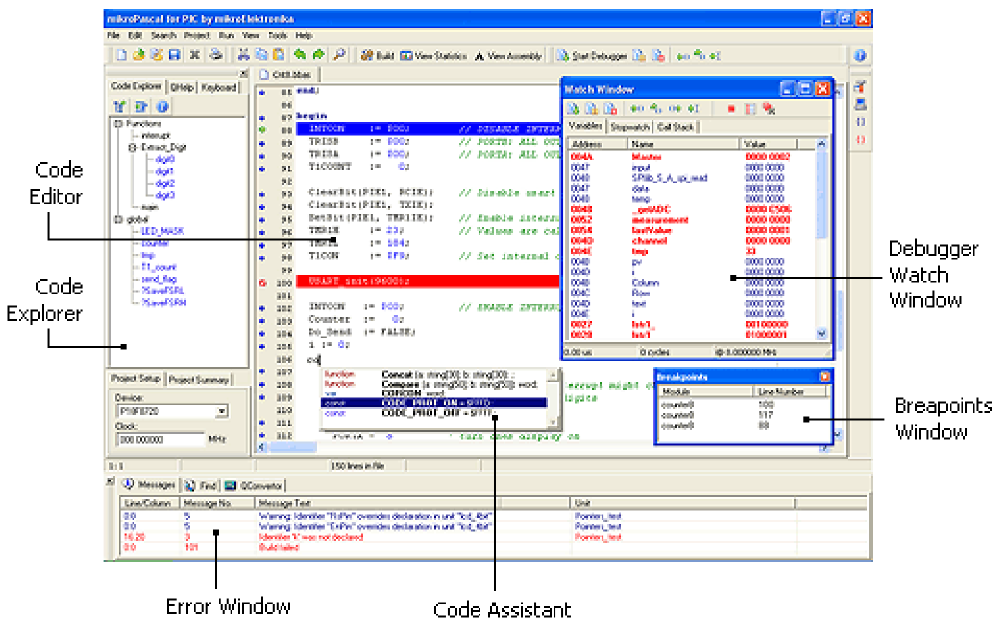

Figure 4.1: Screenshot of the software MikroBasic

SYSTEM TEST AND EVALUATION

This system has been used and tested at the department of electrical engineering, Applied Science Private University, Amman , It shows a very good results without any errors . The staff members using this system reflect good idea about this system . It makes student check procedure in a classroom containing more than 60 or 70 students much easier and precise , with minimum time requirements. The LCD screen and its interface have given us an easy, flexible design in both software and hardware sides, since it is much clear and better from the seven-segments displays for example, and it can be controlled easily through the PIC microcontroller. Another conclusion is that such a system is almost impossible to be done without using a microcontroller , and the PIC microcontroller has been used in this project for its advantages over other microcontrollers and because it is available with its programming tools unlike the other microcontrollers.

ACKNOWLEDGMENT

This work has been done at the Applied Science Private University, Amman, JORDAN, Faculty of Engineering, department of Electrical Engineering. The author would like to thank this university for their strong support to this work.

REFERENCES

[1] Mark Brown, Sam Patadia & Sanjiv Dua “Mike Meyers CompTIA RFID + Certification Passport” McGraw Hill, 2007.

[2] Perdro M. Reyes “RFID: A Guid to Radio Frequency Identification “ McGraw Hill , 2011 .

[3] Jerry Banks, David Hanny, Manuel A. Pachano & Les G. Thompson “RFID Applied “ Wiley 2007

[4] Ian Robertson “Improving Supply Chains Using RFID & Standards” September 2011.

[5] Syed A. Ahson & Mohammad Ilyas “ RFID Handbook : Applications , Technology, Security and Privacy “ CRC Press, 2008 .

[6] Albert Lozano- Nieto “ RFID Design Fundamentals and Applications “ CRC Press , 2010 .

[7] B.P. Singh “Advanced Microprocessors and Microcontrollers “New Age Press, 2009 .

[8] B.P. Singh “Microprocessors interfacing and applications “ New Age Press , 2009