DEVELOPING AN EDUCATIONAL SOFTWARE CONTROLLING DATA TRANSFER OF SERIAL AND PARALLEL PORTS

İlhan TARIMER[*]**, Serkan ÖRÜCÜ**[**]**, Rıza GÜRBÜZ**[**]

[*]Technical Education Faculty, Department of Electronics and Computer Education, Kotekli Campus, 48000, Mugla,Turkey
[**]Çankırı Karatekin University, Technical and Business College, Turkey

itarimer@mu.edu.tr, ser_khanor@hotmail.com, gurbuz@cmyo.ankara.edu.tr

Abstract: In this study, it has been examined whether computer-based teaching software is applicable in today's electronic and computer teaching. By the improvement of today's computer teaching, people are able to reach information easily. Both indoor and outdoor of classroom works, teaching software, which supports learning, has been aimed to help both teachers and students. The educational software prepared in this study that can be used in the area of controls with parallel and serial computer ports. In this study, basic properties of parallel-serial ports, to use them in mechatronics and their using facilities have been explained. How to program these ports together with their connections have been embedded into the educational teaching software. Thus, according to its algorithm and frame, this educational software interface developed can be used for general mechatronics training.

1. CAT SOFTWARE

CAT software change according to the area which they are used and according to specifications of the users. For instance, specifications of a lesson software which is prepared for the usage of one's his/her own working style and for himself herself is obviously different from the specifications of a lesson software which will be applied in a lesson time, in the laboratory under the under the control of a teacher (www.aof.edu.tr/kitap/IOLTP/2276/unite10.pdf 2006).

Computer which operates the software leads among the sectors which have an interrelation with teaching softwares. The first student is the person which communicates with software that can be found frequently. There is not an exception; software is controlled by the first user. The second student is the other student who is also going to use the computer. In the simultaneous applications, the software can be used by students or teachers, or by a group. Teacher is also the director of the software in a laboratory environment (www.aof.edu.tr/kitap/IOLTP/2276/unite10.pdf, 2006; www.ceit.metu.edu.tr/~yuksel/cai-1.doc, 2006).

The first part of program which called "What is parallel port", trains students about parallel port basics, and signal types. The code snippet supports learners to understand parallel port fundamentals with several animations as code based and sound records. While reading these explanations and code based animations with sound records, powerful learning effects would occur. The code based animations initially shows how to send data via parallel ports. The results which got from experiment kits and screen definitely seem to be obtained at the same time. Once the lessons completed, learners can use testing part of program to define its learning level. The second part of program which called "What is serial port" has been prepared as similar as to the program of "What is parallel port", but it includes only serial port lessons.

2. CONTROL OF PORT

Electronic ways which are used for getting and sending data are called as communication points (http://www.cs.itu. edu.tr/~gunduz/courses_2005/mikroisl/slides/d8.pdf, 2006). They are known as "ports" as well. These pins are used for programming hardware or sending data to the computer over the motherboard. The ports of a computer are introduced in below lines.

2.1. Parallel communication and parallel port

If a separated line is used for each bit of the data while transporting a datum, such kind of a communication is called as parallel communication. In the Fig. 1, two units are connected each other by using the technique of parallel communication.

On DB-25 pins, there are 3 ports named, data, status and control (Fig. 2). All of these pins are the exit pins at the level of TTL. The exit of parallel port is at the level of TTL Logic. Even though it changes one computer to another that the capacity of port's current absorbing and carrying; it is limited to a few miliamperes their level of current which will be absorbed and carried. In the figure-2 Port of data, status and control are given with their pin numbers.

It has been thought to use Visual Studio.NET 2005 platform (Tuğay, 2005; Yanik, 2004) for preparing the program on controlling the ports in this study. But Visual Studio.NET 2005 platform doesn't include any component for the control of parallel port, so, only way to reach the addresses of parallel ports is installing an external component. Mostly, inpout 32 DLL is preferred which is mostly known. Also this DLL is used for its blocking the operating systems to reach to ports in the systems which are Windows NT based. Inpout 32 DLL's description in the programming table of C# [1, 2] is like that; firstly that code is added to the form. Inpout 32 DLL basically includes two functions:
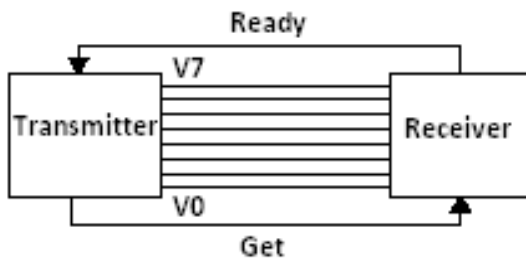
using System.Runtime.InteropServices;

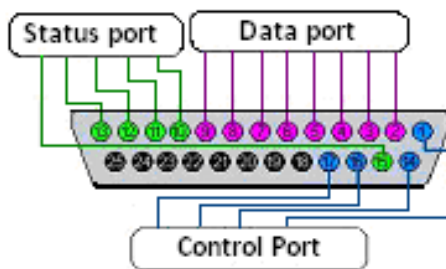**Fig. 1.** Two connected nodes for parallel communication

Fig. 2. Data, Status and Control ports

These are Inp32 and Out32. Inp32 is used to read the data which come from pointed addresses, and on the others hand addresses which will be read are defined with variables from the genre of integer. Out32, similar to Inp32, is defined with variables from the genre of integer which includes the values of data which will be written to that address and the address to write the data (http://www.code project.com/csharp/csppleds.asp, 2006; http://msdn2.micro soft.com/enus/library/system.io.ports.serialport(VS.80).asp x2006).

public class read_port
{[DllImport("Inpout32.dll")]
public static extern short Inp32(int address);}
public class write_port
{[DllImport("inpout32.dll")]
public static extern void Out32(int adress, int value);}

To write data onto the parallel port after the definitions of DLL and function, following code pattern is used (http

://www.csharpnedir.com/makalegoster.asp?MId=576, 2006; Axelson, 1998).

private int baseadres = 0x378;
private void timer1_Tick(object sender, System.EventArgs e)
scan_status_port(baseadres);}
public static int Read_status_port(int base_address)
{int veri =read_port.Inp32(base_address+1);
data= data & 120; return (data >> 3);}

By the help of a timer which is integrated into the form, if a code is written as "read_status port" to scan the ports in a specific time, the data which come from port are scanned automatically in a specific time. Reading data by using data port of parallel port is easier and done as shown below;

write_port.Out32(888, x);

2.2. Serial Communication and Serial Port

In version 2005 of C# programming language which is the one among Visual Studio.NET platforms, for port accesses, one of the components of this software is defined for only serial port (Karagülle, 2004). To be able to use serial port, firstly the component of serial port is added to the form in figure 3.
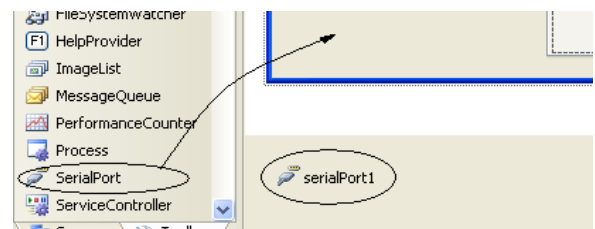

Fig. 3. Adding serial port component

The way to add the component of serial port is shown in the figure 3. This library supplement enables all of the functions related with the serial port to use altogether. Furthermore, an example to form some functions can be defined and can be given as like below lines (http://www. dotnetmagzin.de/itr/online_artikel/psecom,id,397,nodeid,31 .html, 2006).

private void Form1_Load(object sender, EventArgs e)
{serialPort1.StopBits= StopBits.One;
serialPort1.DataBits = 8;
serialPort1.Parity= Parity.None;
serialPort1.BaudRate = 9600;
serialPort1.PortName = "COM1";
serialPort1.Open();}

When data transfer from serial port was being examined, the device developed for experiments must be installed. Thus, a code pattern is obtained to read and to write like
as one in the below lines;

string data_write = textBox1.Text;
serialPort1.WriteLine(data_write);

string data_read;
data_read = serialPort1.ReadLine();

Instead of this way, a delegate might be used for opening data process channel (Tuğay, 2005; Gümüşkaya, 2004). In this case, a code pattern is occurred like as ones in the below lines;

private delegate void run(string value);
private run serial_port_thread;
private void Form2_Load(object sender, EventArgs e)
{serialPort1.StopBits = StopBits.One;
serialPort1.DataBits = 8;
serialPort1.Parity = Parity.None;
serialPort1.BaudRate = 9600;
serialPort1.PortName = "COM1";
serialPort1.Open();
seri_port_thread=new run (port_listen);}
private void button1_Click(object sender, EventArgs e)
{string a = textBox1.Text; serialPort1.WriteLine(a);}
private void get_data(object
sender,System.IO.Ports.SerialDataReceivedEventArgs e)
{string data = serialPort1.ReadExisting();
this.Invoke(this.serial_port_thread, new object[] {data });}

The educational software program has been designed as two separated parts. One of them is for the kits and the other is for the experimental devices. They could be combined constructional controlling of this computer and the software according to port types which people need. A sample has been given in figure 4. It shows the animation of   data transfer which is used in the part of "what is parallel port?" which is one of P-Kod software. The other sample software has been given in the figure 5 which shows transferring data from/to the addresses of parallel ports according to the boxes which are clicked.

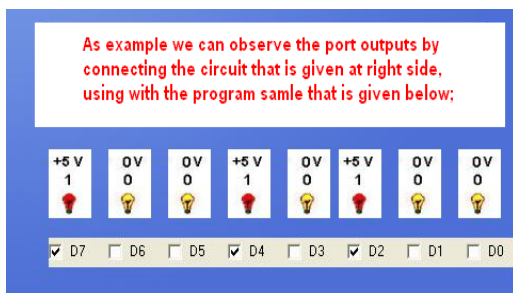**Fig. 4.** An animation written with codes

Fig. 5. A sample software usage panel

By the time first installation, the form view of "what is parallel port" program which is the first part of P-Kod software is shown as in the figure 6. When it is clicked to the parallel port button, the screen view transforms its form into the figure below (Fig. 7). The first part of the software appears to the form that concerns with the teaching of the lecture topic. At this part, the basic functions of parallel ports could be examined as audio–visual. However an audio and video playback dll which is one of the dll of Direct X has been used in this coding section of the program.
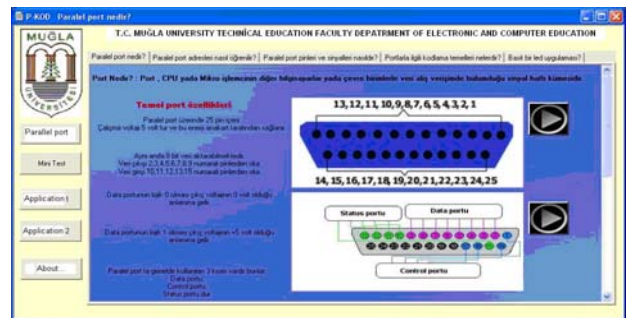

Fig. 6. P-KOD software main page

Fig. 7. P-KOD software teaching part

Fig. 8. The software tutoring part

"What is serial port", which is the second part of P-Kod software is similar to the view of the first installation section of "what is parallel port". When clicked the button "serial port", it transforms its screen view into the view as it seen in figure 8. The view of mini experiment kits are given in the figure 9 which was prepared for P-Kod. The circle of data reading contains 4 units of buttons and resistances, and the circle of data writing contains 8 units of LEDs and 8 resistances.

Fig. 9. Mini experiment kits

3. USING THE SOFTWARE INTERFACE AND COMPARISONS

Once clicked on to the button of parallel port, the initial section of the software application related with teaching lesson appears to our face. In this section, features of parallel port can be examined as audio visually. After that it has been stated pin numbers and basic features of data, control and status ports (Fig. 10). At the next steps of this software interface, there is a sample workable application using with together an experiment kit. To be able to have the sample software interface to connect the experiment kit numbered 1 with the cable belonging to that one, they must connect to each others via normal apparatus (Fig. 9). The first section of serial port tutoring is being related with serial communication. Then general specifications of serial port have been prepared to be taught to learners in a form of charts as audio visual (Fig. 11).
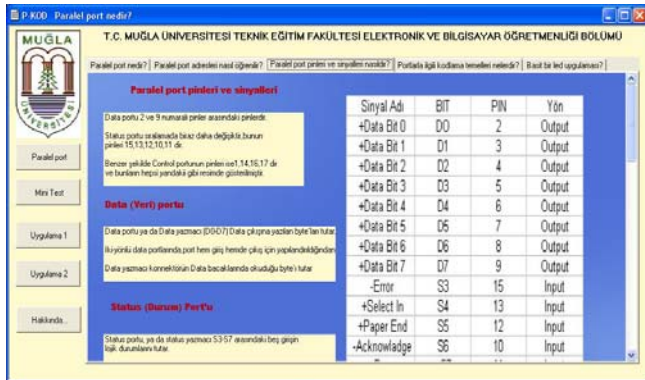

Fig. 10. Pin numbers and basic features

When clicked on to the button of application, a new segment is opened. This segment is the last section of the software interface and it makes the two computers to communicate together (Fig. 12). In both sections there have been several tests in which learning levels can be evaluated. Therefore this software includes open source codes as well and it steers learners to benefit from these codes for ascending their personal experiences (Fig. 13).
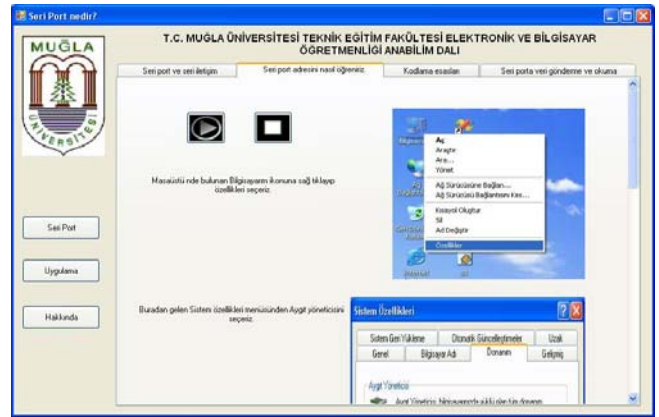

Fig. 11. Experiment kit

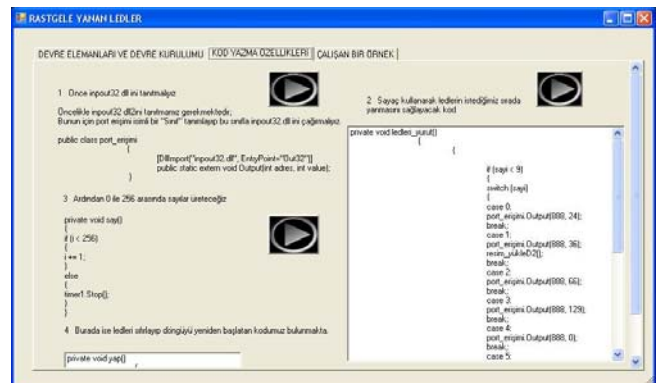**Fig. 12.** Application window of the software interface

Fig. 13. Open source codes

These codes in which any learners interested in using of them have been designed and written according to the system of chart. So, students and learners have got a re-playing chance. Thus, this software interface forms the learning and teaching process more rapid, more retentive and absolutely independent from an experimental environment.

4. CONCLUSIONS

In this study, it has been examined an educational software for teaching of parallel and serial ports in electronics, computer and mechatronics. To do this, a computer based teaching programs have been developed by using C# pro-

gramming language which is one of the Visual Studio.NET 2005 platform programs. The software developed provides the users to learn port control theories both on their own and under control of a teacher. Although the port addresses used in this application are stationary, the user can also choose other addresses that he wants to write. The entire program modules were powered by a graphic user interface. It has been examined within the program, how audio sources support learners besides the visual sources. From this aspect, the results obtained would be a pioneer amongst port control teaching programs. With this technique, both theory and applications in area of electronics, computer and mechatronics would be applied together and there might be no need for installing and running additional hardware equipments.

REFERENCES

1. **Karagülle İ.** (2004), *Visual C#.Net Introduction Guide*, Türkmen Pub. House, Istanbul.
2. **Yanık M.** (2004), *Microsoft Visual C#.Net*, Seckin Publishings.
3. **Tuğay G.** (2005), *Electronics Hobby*, ALFA Publications, Istanbul.
4. **Axelson J.** (1998), *Parallel Port Complete*, Lakeview Research, USA.
5. **Gümüşkaya H.** (2004), *Microprocessors and Computers*, ALFA Publications, Istanbul.
6. Anadolu University, Open Learning Faculty Lecture Notes, M. Emin Mutlu, www.aof.edu.tr/kitap/IOLTP/2276/unite10.pdf, (Access: January 2006).
7. ITU, Lecture Notes, Şule Öğüdücü, http://www.cs.itu.edu.tr/~gunduz/courses_2005/mikroisl/slides/d8.pdf (Access: January 2006).
8. Levent Saltuklaroğlu, http://www.codeproject.com/csharp/csppleds.asp, (Access: Feb. 2006).
9. METU, Lecture Notes, Yüksel Göktaş, www.ceit.metu.edu.tr/~yuksel/cai-1.doc, (Access: January 2006).
10. http://msdn2.microsoft.com/enus/library/system.io.ports.serialport(VS.80).aspx, (March 2006)
11. http://www.csharpnedir.com/makalegoster.asp?MId=576, Fevzi Özgül, (Access: April 2006).
12. http://www.dotnet-magazin.de/itr/online_artikel/psecom,id,397,nodeid,31.html Markus Palme, (Access: January 2006).