Design and implementation of Vehicle controller

A Project Report Submitted to the College of Science, Baghdad University in Partial Fulfillment of the Requirements for the BSc Degree in Computer Science

BY

Saad Adnan Abd
Abdalhakeem Anan Ali

SUPERVIED BY

LECTURAL

Dr. Asmaa Q. Shareef

2011-2012
(قالوا سبحانه، لا علم لنا إلا ما علمتنا إنك أنت العالم الحكيم)

سورة البقرة (الآية 32)
Acknowledgments

First and foremost we would like to thank and Praise God lord of the world, that without facilitating from him we would not have been able to create this project.

we would like to express our deep thanks and gratitude to our supervisor Dr. Asmaa Qasim for suggesting the project and her continuous guidance and enthusiastic assistance.

Deep thanks are extended to the head and staff of the computer department.

Deep appreciation to the staff of computer laboratories for the facilities offered during this work.

Finally, we would like to express our deep gratitude to our families who stood in our side in every possible way and our thanks to all our friends.
To
My Mother
My Father
My brothers
and My Friends
for their Kindness,
Attention and encouragement.
Contents

abstract

Introduction ..........................................................................................................6

Chapter one (general view)..................................................................................8

Chapter two (Theoretical Sid)...........................................................................10

  2.1 Study of vehicle's parts
  2.2 Theory of wireless controlled vehicle
  2.3 Software
  2.4 Design

Chapter three (design)......................................................................................13

  3.1 Interface
    3.1.1 current use
    3.1.2 Port addresses
    3.1.3 Program interface
    3.1.4 pin outs
  3.2 The Components of Electrical Circuit
    3.2.1 resistance
    3.2.2 Transistor
      3.2.2.1 Transistor as a switch

Chapter four (implementation of the designed controller)..................................20
Chapter three *(Conclusions and Suggestions)*...26

5.1 Conclusions

5.2 Suggestions
Abstract

The project aims to control a vehicle by means of wireless control, this was implemented by using one of the object oriented languages this language is visual basic.

We design specific board to control the transmission circuit electronically.

To interface the designed board to the computer we have used the parallel port ( lpt1).

We use two methods to move the car the first method is implemented by using keyboard, the second method is implemented by drawing desired path to move a car according to this path.

The project contain main form which is the first form that enable us to choose one of the mentioned methods.

The second form used for controlling car using keyboard, in simplicity this part read the pressed arrow then send it the parallel port that interfaced to fixed electronic board.

The third form used for controlling car by drawing path, in this stage each point was buffered each point in an array; the distance between each two points was buffered in another array then when press the star button the serial movements will be sent to the parallel port.
Introduction

The wireless control systems are take a special importance among the other systems as these systems that do not contain computerized controller or a robotic control. This kind of control would avoid the human being the danger of visiting a danger places, also there are places that need the wireless control as the only way to control the system, such as space shuttles and robots.

In this project we design a controller of a vehicle (small toy car) by using the computer. The goals of this project were to control the car by using two different ways of controlling; keyboard and drawing path. Both had been achieved successfully.

The software for this project is designed by using the Visual Basic® language. The software contains two stages; each stage represents one of the control ways that is desired.

The hardware used in this project was a designed circuit that was used to interface the computer with the controlled of a car. The car itself, had to make some changes on the DC motor so that the speed is lessened by connecting a series variable resistance that reduced the high speed to have a better accuracy. This project contains five chapters:

Chapter one is a General View, that will introduce the project to the reader, and explains the goals and the use of the remote controlled vehicle by computer.

Chapter two is the Theoretical side, that explains the parts of the vehicle used and the theory that is working on and gives an introduction to the software and hardware that is used.
Chapter three is *the designed controller*; that contains the details of the circuit used with the data sheets for the components.

Chapter four is *implementation of the designed controller*, that explains to the reader the full software of the project.

Chapter five is *the Conclusions and Suggestions*, that contains our suggestions and ideas for future implementation and a discussion to the problems that we've faced through the design and implementation of this project.
1.1 Introduction

Personal Computers (PC's) are considered as one of the main features of the human civilization nowadays, PC's are used widely in most of applications that needs data processing. One of the most important fields that benefits from PC's capabilities is Control System.

A control system is an interconnection of components forming a system configuration that will provide a desired system response\(^{(1)}\). There are two main categories of control systems: open–loop control system (without feedback) and closed–loop control system (with feedback presence).

In this project we will use the computer as a controller to control a vehicle by designing necessary electronic circuits and software. Our project is an open-loop control system (which is defined as a system that is designed to obtain a desired response without feedback).

1.2 Project work

keyboard controlling is the first stage in our project, we do not include mouse controlling because there is no vital difference from the keyboard, but there is a new goal was to draw a path to the car and then make the car move according to drawn path.

1.3 Project Goals
The main goal of this project is wireless control of a vehicle by using computer. The car position should be controlled by:

1. Keyboard.
2. Drawing a path map.

These goals had been made successfully.

1.4 Project Use

Too many applications and fields would has benefit from Wireless Control of a Vehicle by Computer, for example:

1. It can be used to explore places that are difficult human beings.
2. It can have video tape or take photograph shots (by using a fixed camera on the car) for a different places such as caves.
3. It can be used at warehouses to move stuff from place to another by a pre-determined path.
4. Historical places can use these vehicles to take the tourists in a pre-determined path journey.
Theoretical Background

Theoretical background for this project includes:

1. Study of vehicle's parts.
2. Theory of wireless controlled vehicle.
3. The hardware\ the designed controller.
4. The software\ Implementation of the designed controller.

2.1 Study of vehicle's parts

A small vehicle is used in this project that consists two DC motors and one receiving, transmission: circuit.

The first DC motor is fixed in the forward of the vehicle. This motor is responsible of the left, right turning movements of the front tires. This motor is connected to gearbox that accelerates the response of the tires.

The second DC motor is fixed at the backside of the vehicle. It is responsible for the movement of the back tires forward and backward. It is also connected to a gearbox.

The vehicle also contains a 9 volt source (6×1.5 volt batteries) which supplies the motors and the receiving circuit as followed in the figure below:

Figure 2.1 inside of rc car
The transmission device (Remote Control) contains a 9 volt source (1×9 volt battery) and transmits a 27 MHz frequency. This device contains two switches; one for driving vehicle forward or backward, and the second for turning left or right.

2.2 Theory of wireless controlled vehicle

The wireless controlled vehicle principle of work depends on transmitting known signals from the remote control to the receiver circuit (fixed on the vehicle) these signals will be received by the car through the antenna, each signal will trigger a pre-specified movement.

The remote control contains two switches; one gives the directions forward and backward, the second gives the directions left and right. These switches (which were cancelled) are connected to other switches that activate the first DC motor or the second according to the required movement sequence.

2.3 Software

We've used the oriented object language of Visual Basic® to design the software that will control the vehicle through the computer. This software will send digital signals to the designed electrical circuit through the LPT1 port, these signal will activate the DC motors.

This software contains two main parts:

1. **Keyboard Control**: this part of the program allows the user to control the car by using the keyboard keys. The arrows on the keyboard enable the user to move the car to four directions; forward, backward, left and right. If the user pressed two opposite keys this would be a stop order.

2. **Path control**: this part of software contains two stages:
a. **First stage:** in this stage the user will draw the path that the vehicle will move on, the drawing will be in a special form in the program by clicking the mouse, the 1st click will determine the starting point the 2nd click will draw a horizontal or vertical line between the starting point and the click (even if it was inclined) and so on, till we finish the path drawing; with the capability of moving car backward with side movements. In this way of control the user is limited to draw a vertical or horizontal line. The car cannot move on curve. The forward lines in the path will be in black color. The reverse lines will be in red color.

b. **Second stage:** this stage starts when the user clicks the START button which will initiate the process of analyzing the path and the movements wanted and will send the suitable signals to execute the desired path accurately.

### 2.4 Hardware

We've designed a special circuit to control the car, each element of the circuit had been chosen carefully so that the circuit will work properly as desired. This circuit will initiate the transmitting from the remote control to the car obeying signals from the computer that specifies the position wanted. The produced signal will pass through lpt1 to the resistance then to the transistor that work as switch in our case. All of these parts will be studied in details in chapter 3.
Design

To understand the electrical circuit that we've designed, we have to study the Interface and then knowing the components of the circuit with some information about these components.

3.1 Interface

In one way or another ultimate user of a computer is a man and it may have become apparent that it is not very easy for a man to communicate directly with the computer because of the speed problem, the coding of data. From this point of view we can define the Interface as a term that describes the interaction or data transfer between two essentially autonomous systems\(^{(2)}\) and usually done by using keyboard and monitor.

In this project we've used the parallel port LPT1 as an output port for the signals to the interface circuit that will control the vehicle.

A parallel port is a type of interface found on computers (personal and otherwise) for connecting various peripherals. In computing, a parallel port is a parallel communication physical interface. It is also known as a printer port or Centronics port. The IEEE_1284 standard defines the bi-directional version of the port, which allows the transmission and reception of data bits at the same time.

The printer port have 25 pin, as illustrated in the following Fig 3.1.

![Parallel Port Diagram]

8 bit output 378 4 bit input 379
The pins 18 to 25 are connected to computer ground (0 v), any circuit that is connected to the computer should be connected to the computer ground.

3.1.1 current use
For consumers, the USB interface and sometimes Ethernet has effectively replaced the parallel printer port, for connections both to printers and to other devices.

Many manufacturers of personal computers and laptops consider parallel to be a legacy port and no longer include the parallel interface. The guidelines for Microsoft's Windows Logo Program "strongly discourages" systems builders from including parallel ports. USB-to-parallel adapters are available that can make parallel-only printers work with USB-only systems. There are PCI (and PCI-express) cards that provide parallel ports.

For electronics hobbyists the parallel port is still often the easiest way to connect to an external circuit board. It is faster than the other common legacy port (serial port) and requires no serial-to-parallel converter, and requires far less interface logic and software than a USB target interface therefore we are preferred parallel port for our project.
3.1.2 Port addresses

Traditionally IBM PC systems have allocated their first three parallel ports according to the configuration in the table below\(^{(3)}\).

<table>
<thead>
<tr>
<th>PORT NAME</th>
<th>Interrupt #</th>
<th>Starting I/O</th>
<th>Ending I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPT1</td>
<td>IRQ 7</td>
<td>0x378</td>
<td>0x37f</td>
</tr>
<tr>
<td>LPT2</td>
<td>IRQ 5</td>
<td>0x278</td>
<td>0x27f</td>
</tr>
<tr>
<td>LPT3</td>
<td>IRQ 2</td>
<td>0x3bc</td>
<td>0x3bf</td>
</tr>
</tbody>
</table>

If there is an unused LPTx slot, the port addresses of the others are moved up.

Bit to Pin Mapping for the Standard Parallel Port (SPP)

<table>
<thead>
<tr>
<th>Address</th>
<th>MSB</th>
<th>LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit: 7 6 5 4 3 2 1 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Base (Data port)</td>
<td>Pin: 9 8 7 6 5 4 3 2</td>
<td></td>
</tr>
</tbody>
</table>
In versions of Windows that did not use the Windows NT kernel (as well as DOS and some other operating systems), programs could access the parallel port with simple outportb() and inportb() subroutine commands, therefore we are enforced to add dll library that prepares these procedures. In operating systems such as Windows NT and Unix (NetBSD, FreeBSD, Solaris, 386BSD, etc.), the microprocessor is operated in a different security ring, and access to the parallel port is inhibited, unless using the required driver(4).

3.1.4 pinouts

Pinouts for parallel port connectors are:

<table>
<thead>
<tr>
<th>Pin No (DB25)</th>
<th>Pin No (36 pin)</th>
<th>Signal name</th>
<th>Direction</th>
<th>Register - bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>Strobe</td>
<td>In/Out</td>
<td>Control-0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Data0</td>
<td>Out</td>
<td>Data-0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>Data1</td>
<td>Out</td>
<td>Data-1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>Data2</td>
<td>Out</td>
<td>Data-2</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>Data3</td>
<td>Out</td>
<td>Data-3</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>Data4</td>
<td>Out</td>
<td>Data-4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>Data5</td>
<td>Out</td>
<td>Data-5</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>Data6</td>
<td>Out</td>
<td>Data-6</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>Data7</td>
<td>Out</td>
<td>Data-7</td>
</tr>
</tbody>
</table>

3.2 The Components of Electrical Circuit

The circuit contains two electronic parts are as follows:

3.2.1 resistance
We are used four resistors of $2k\Omega$ each one connected to the one of the serial pins of the interface from the first side and connected to the second part of the circuit by the other side.

3.2.2 Transistor

The transistor used in this circuit is BC33740 npn this transistor is general purpose therefore we used it as switch, transistor pins are arranged as the following figures\(^{(4)}\):

![Figure 3.2 transistor's pins attaching](image)

If we implement the circuit as illustrated in the above figure the circuit will work properly if and only if the the applied voltage from serial port is very small otherwise the transistor will explode, therefore use a resistor to control our base current.

The proper circuit will be as the following figure:
3.2.2.1 Transistor as a switch

When used as an AC signal amplifier, the transistors Base biasing voltage is applied in such a way that it always operates within its "active" region, that is the linear part of the output characteristics curves are used. However, both the NPN & PNP type bipolar transistors can be made to operate as "ON/OFF" type solid state switches by biasing the transistors base differently to that of a signal amplifier. Solid state switches are one of the main applications for the use of transistors, and transistor switches can be used for controlling high power devices such as motors, solenoids or lamps, but they can also used in digital electronics and logic gate circuits\(^5\).

If the transistor used as switch then it has two mode the first mode is off that realized when thevolt of the serial port is low (zero) in this case the circuit will be as the following figure:

![Figure 3.5 opened circuit](image_url)
The second mode of the transistor is active this case realized when the volt of the serial port is high (3v), then the circuit of transistor as following figure:

![Figure 3.6 closed circuit](image)

Then the transistor operates as a switch. With a zero signal applied to the Base of the transistor it turns "OFF" acting like an open switch. With a positive signal applied to the Base of the transistor it turns "ON" acting like a closed switch.
Implementation of the designed controller

The last century had witnessed a very enormous progress in all sciences, especially computers. This progress was in two parallel fields; Software and Hardware. The programming languages were one of the main fields in software benefits from the great changing and progress that occurred.

One of the most popular languages that the programmers used a lot was, GwBasic. This language had witnessed a lot of changing through the next version of it, QBasic, which is considered as a mother language or core for the next version of basic which was, Visual Basic. Visual Basic presented a lot of facilities for the programmers, especially it is working under windows and it is Object Oriented Language (OOL) that deals with objects. It has a very wide capabilities, one of them is interfacing. The flexibility of visual basic made it as one of the most languages that is used for interface.

Interface programs in visual basic faces one problem, which is the definition of external procedures between the port and the program. These procedures are stored into a Dynamic Library Link (DLL) files. These files cannot be written with visual basic, so visual basic designers solved this problem. Visual basic can import such programs from other sources. This kind of files (i.e., DLL) can be written by other languages such as C ++(6).

The ability of visual basic in designing a beautiful Graphical User Interface (GUI) enabling to design the forms of the stages of the program that was designed to control the vehicle. As mentioned the software contains two stages, or specifically two ways of control:

1. Controlling by using keyboard.
2. Controlling by using drawing path.
The following form show these stages

Figure 4.1 main form

The program contain two parts of controlling, each part has distinct approach from other. The first part is key control, as followed in below

Figure 4.2 key-form

in the stage of the dependence, which will be on arrow keys by using key down and key up properties. These properties will verify the pressed key(s) (one key or double) depending on key code for each arrow. The
key code for each arrow is as follow ( up: 37, right: 38, down: 39 and left: 40). After reading the input from keyboard the program will produce a desired signal of motion.

The graphical user interface contain four pictorial representations each one will represent one direction, when we hit any arrow from keyboard this will show up the correspondence arrow picture on form and return back to normal case when we release the key.

The code section of this part very clear, in simplicity is to move the key code argument into a variable then dedicate the motion direction by using select case statement on this variable.

Select case stmt has eight cases for each case we will assign specific value, the first four cases are the four directions while the remain directions are combined for four directions ( up+l, up+r, down+l, down+r).

The values assigned for first four direction ( up, right, down, left) are 16, 2, 4 and 1 respectively, the cause of choosing these values is the corresponding binary values of these decimal values. The correspondence binary values are 10000, 00010, 00100 and 00001 respectively for this only one pin from lpt1 interface will be enabled.

As for the combined directions, the values will be 17, 18, 5 and 6 and the corresponding binary values 10001, 10010, 00101 and 00110 respectively in this case two pins from lpt will be enabled.
The enabled pin(s) (either one pin or two pins) will operate one or two of board switches this in turn operate remote switch(s). The second stage is path control, as followed in the following figure:

![Figure 4.3 path-form](image)

In this type the vehicle will be moved on fixed path (drawn by the user), the mouse down and mouse up events will be used.

This part will analyze the chosen point to recognize the desired movement and to show it on screen if it was forward, reverse, right or left. In this part also the required time for each movement is calculated (by knowing the length) then to pass the calculated value to mv function in order to perform specific movement according to the drawn path. The form contain picture box for drawing purposes and a command.

For each mouse down event on the picture box will draw point and line between this point and previous point, each point is x and y dimensions each dimension will be kept in an array to calculate the
difference between adjacent point then to check the result if no change on x-axis then the change will be on y-axis only and the movement will be either forward or backward, and vice versa for changing on x-axis.

The line between two points represent the movement distance, this distance evaluated by multiplying the length of the line on box by an operator calculated by dividing picture box height on the range of vehicle movement k=p1.height/range in this way the line drawn from bottom to up is the range of vehicle exactly.

When the user finishes his drawing path he will hit a command to begin in calculating needed information (time, direction and offset).

For each calculation of distance the program will record direction signal in an array , many tests on this array will be hold then calling mv function with specific parameter according to the results of the testing operations.

These testing operations are implemented in the following algorithms:

Begin

D= forward

Crd=d //car direction is forward as default

If d=forward then

Begin

If dr(k)= forward then Call mv dist(k),dr(k)

Else if dr(k)= backward then

Begin
Call mv dist(k),dr(k)

D=forward // move to back but the car direction stay forward

End

Else if dr(k)= left then

begin

Call mv dist(k),dr(k)

D=left // car direction will be turn to left

End

Else if dr(k)=right then Begin

Call mv dist(k),dr(k)

D=right // car direction will be turn to right

End

End

End of the main


Conclusions and Suggestions

5.1 Conclusions

During the work to finish this project we face a lot of problems, some of problems were solved, others were not. The following is the most important problems that we've faced.

1. The bad hardware of the vehicle. The car doesn't have a constant speed because of power source supply; also the front tires are not turning with good accuracy. This would be faced by making a feedback from the car and making the car control a close-loop system.

2. We face some problems of using a relay (switch) therefore we use transistor as switch.

3. Rare use of parallel port was the major problem in our project till we found it in old PC.

4. At the part of drawing we face the problem of adding an incline path to the drawing pad. But because of the limited time we couldn't add this capability.

5.2 Suggestions
1. To add feedback to the control system. This will be done by adding a transmitter to the vehicle that will be connected to several sensors and will be useful to determine the car’s speed and location, and this will help to determine the angle of turning that is needed to get better accuracy.

2. Adding a camera to the vehicle, so that it would be able to bring pictures and videos of places, and can avoid any obstacle that would be on its way.

3. Improving the software program by adding an inclined direction to the drawing control.
References

1. Modern Control System Richard C. Dorf and Robert H. Bishop
4. Interfacing the Standard Parallel Port
تصميم وتطبيق مسيطر على مركبة
مشروع
مُقدم إلى قسم علوم الحاسبات في كلية العلوم - جامعة بغداد وهو جزء من متطلبات الحصول على درجة البكالوريوس في علوم الحاسبات

من قبل

الطالب

الطالب

عبدالحكيم عدنان علي

بالإشراف

الدكتورة أسماء قاسم شريف
2012-2011