



**Faculty of Electronics,
Telecommunications and
Information Technology**



TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA
ROMANIA

MICROPROCESSORS

Connection of a smart phone to a computer using
the USB and the Hayes protocol in order to send /
receive SMSs

Coordinator:

Mircea Giurgiu, PhD

Student:

Gherman Teodora Ștefania

Group 2331

Year 3

1. Introduction

For this project it is required to connect a phone to a computer by USB to send and receive short message service using the Hayes protocol. Essentially, this means that the phone provides GSM modem capabilities. Then I use the computer / PC and AT commands to get the received SMS messages from the mobile phone or GSM/GPRS modem.

A GSM modem is a specialized type of modem which accepts a sim card and operates a subscription to a mobile operator. When a GSM modem is connected to a computer, this allows the computer to use the GSM modem to communicate over the mobile network. While these GSM modems are most frequently used to provide mobile internet connectivity, many of them can also be used for sending and receiving SMS and MMS messages.[4]

This can be implemented using an already existing emulator or creating one using a programming language (for example C# or Python). I could not find a phone that my laptop recognizes as a modem, so I will speak purely theoretical and I will use videos from a streaming platform to explain the process and what I understood. They will be referenced in the bibliography at point number [5].

2. Theoretical background

2.1. What are AT commands?

AT commands are commands which are used to control the modems where AT stands for Attention. These commands were derived from Hayes commands which were used by the Hayes smart modems. Every wireless, as well as the dial up modems, require an AT command to interact with a computer machine. These AT commands along with other extended commands also require Hayes command set as a subset.[2]

2.2. Usage

The AT commands can be used with GSM module and GPRS MODEMs or phone to access these services and information[2]:

- SMS
- MMS
- Fax
- Voice link and other data over mobile network
- Information and configuration concerning the mobile devices or MODEM and SIM card.

2.3. Hayes protocol

The Hayes command set (also known as the AT command set) is a specific command language originally developed by Dennis Hayes for the Hayes Smart modem 300 baud modem in 1981. [1]

The command set consists of a series of short text strings which can be combined to produce commands for operations such as dialing, hanging up, and changing the parameters of the connection. The vast majority of dial-up modems use the Hayes command set in numerous variations. [1]

The command set covered only those operations supported by the earliest 300 bit/s modems. When new commands were required to control additional functionality in higher speed modems, a variety of one-off standards emerged from each of the major vendors. These continued to share the basic command structure and syntax but added any number of new commands using some sort of prefix character – & for Hayes and USR, and \ for Microcom, for instance. Many of these were re-standardized on the Hayes extensions after the introduction of the SupraFAXModem 14400 and the market consolidation that followed. [1]

The Hayes command set includes commands for various phone-line operations such as dialing and hanging-up. It also includes various controls to set up the modem, including a set of register commands which allowed the user to directly set the various memory locations in the original Hayes modem. The command set was copied largely verbatim, including the meaning of the registers, by almost all early 300 baud modem manufacturers, of which there were quite a few. [1]

3. Implementation

The connection of a smart phone to a computer using the USB and the Hayes protocol in order to send / receive SMSs is most of the time implemented by downloading an emulator which can inherently be built by the user themselves with the help of a programming language (for example C# or Python).

3.1. C# implementation

For the step-by-step implementation one can use the program Visual Studio.

Firstly, the user would need to design their own user interface (UI) similar to the one in Figure 1. User Interface [4]Figure 1.

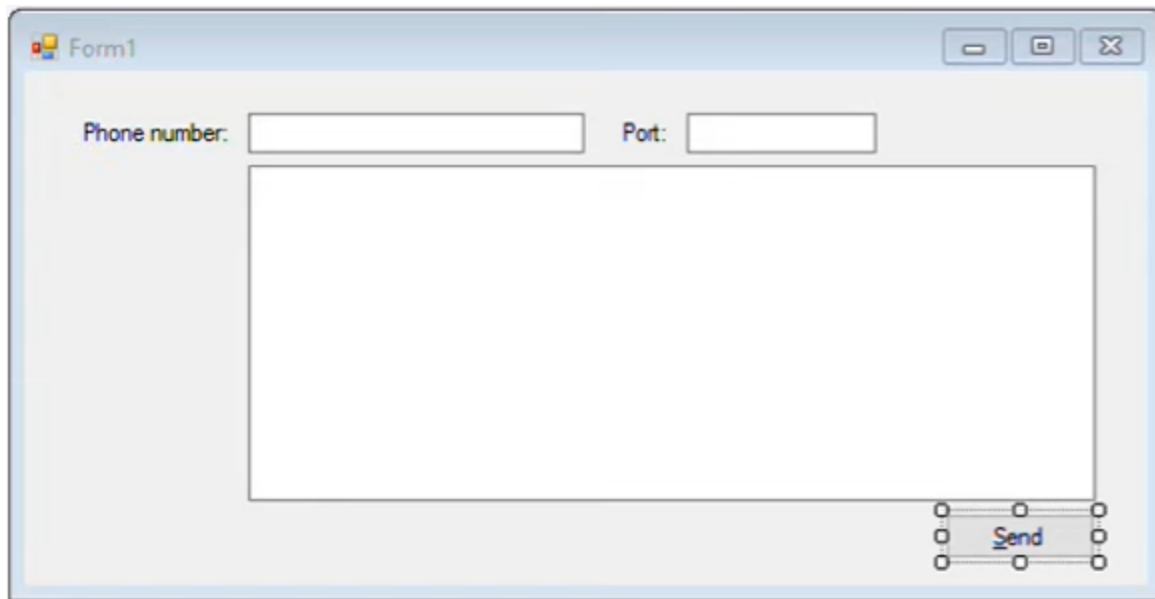


Figure 1. User Interface [4]

It looks similarly to the phone's messaging app screen: a text box in which one will add the phone number they want to communicate with and a text box for the intended message.

Additional to that, the user also should have a button of type text box in which they will add the name of the port their modem is using. This can be found in "Device manager" section "Modems", right click on the available modem and open "Properties". In the window that will open one will access the "Modem" tab and read the name of the port as seen in Figure 2. The name of the port is needed for an already existing emulator as well and the same procedure can be used, although some emulators can find it automatically.

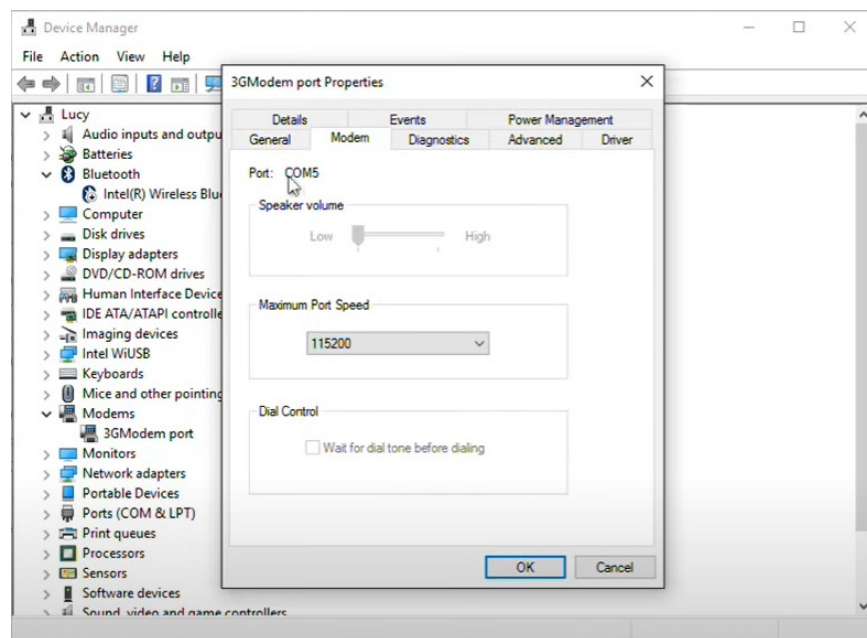


Figure 2. How to find the name of a modem port and the maximum port speed [4]

Further on, the user would need to implement, for example, your C# code using AT commands to send the SMS message. AT (“attention”) commands are instructions used to control a modem. Every command line starts with AT.

```
1 reference
private void btnSend_Click(object sender, EventArgs e)
{
    try
    {
        SerialPort sp = new SerialPort();
        sp.PortName = txtPort.Text;
        sp.Open();
        sp.WriteLine("AT" + Environment.NewLine);
        Thread.Sleep(100);
1 sp.WriteLine("AT+CMGF=1" + Environment.NewLine);
        Thread.Sleep(100);
2 sp.WriteLine("AT+CSCS=\"GSM\"" + Environment.NewLine);
        Thread.Sleep(100);
3 sp.WriteLine("AT+CMGS=\"" + txtPhoneNumber.Text + "\"" + Environment.NewLine);
        Thread.Sleep(100);
        sp.WriteLine(txtMessage.Text + Environment.NewLine);
        Thread.Sleep(100);
        sp.Write(new byte[] { 26 }, 0, 1);
        Thread.Sleep(100);
        var response = sp.ReadExisting();
        if (response.Contains("ERROR"))
            MessageBox.Show("Send failed !", "Message", MessageBoxButtons.OK, MessageBoxIcon.Information);
        else
            MessageBox.Show("SMS Sent !", "Message", MessageBoxButtons.OK, MessageBoxIcon.Information);
        sp.Close();
    }
}
```

Figure 3. Example of code for creating an emulator [4]

Line 1: The AT command +CMGF (command name in text: Message Format) is used to select the operating mode of the GSM/GPRS modem or mobile phone. It takes one parameter. The value of the parameter can either be 0 or 1. The values 0 and 1 refer to SMS PDU mode and SMS text mode respectively. SMS PDU mode is the default mode if it is implemented on the mobile device. [3] Therefore, in our case it sets the operating mode to SMS text mode.

Line 2: The AT command +CSCS is used to select the character set of the mobile equipment. Some possible values are "GSM", "HEX", "IRA", "PCDN", "UCS2", "UTF-8" etc. An SMS can contain up to 140 bytes. The GSM character set is encoded using 7-bits, rather than the usual 8-bits that make a byte. This means there can be 160 characters in an SMS. This 7-bit limitation means only 128 standard characters can be encoded. The GSM standard gets round this by also having the Extended GSM character set. These are another 10 characters which are actually sent by sending two 7-bit characters, an escape (ESC) character followed by another character. This means that 160 ‘£’ symbols can fit in a single SMS, but only 80 ‘{’ symbols. [6]

Line 3: The AT command +CMGS is used to *send* the SMS message to a GSM phone. Here the user should insert the phone number and type in the message. It might also display the number of characters in the message they want to send if they are using an emulator.

3.2. Emulator implementation

The chosen emulator for this part is HyperTerminal.

Upon opening, the emulator will open a small window (as seen in Figure 4) in which the user is able to create a “New connection”. This feature allows the user to save their input information such as the modem they are using, the phone number and the messaging history.



Figure 4. Creating a new project in HyperTerminal [5]

Further on, the user is asked to connect to his modem of choice and to input the phone number afferent to the modem (see Figure 5). As I said above, this information shall be saved by creating a new connection.



Figure 5. Connecting to desired GSM modem [5]

In order to do this, one will need to search for the name of the modem they want to use as explained in 3.1. C# implementation fourth paragraph, Figure 2.

3.2.1. Sending SMS from computer to phone

The same AT commands used in the C# implementation can be used in the emulator as well.

```

AT
OK
AT+CMGF=1
OK
AT+CSCS="GSM"
OK
AT+CMGS="9800220107"
> WWW.NBCAFE.IN
> 
OK
-

```

Figure 6. Example for sending SMS [5]

There are more variations of AT commands that can be used to send text messages. Depending on the command combinations, one can optimize their program. The user can check Table 1 or the link (AT Commands, GSM AT command set, 2018) for a complete list of AT commands.

AT command	Meaning
+CMGS	Send message
+CMSS	Send message from storage
+CMGW	Write message to memory
+CMGD	Delete message
+CMGC	Send command
+CMMS	More messages to send

Table 1. AT commands used for sending SMS [7]

For example, in Figure 7 the user chose +CMGW instead of +CMGS as it was above in Figure 6. This command is used to *write* an SMS text message to the message storage of the GSM/GPRS modem. The AT command +CMSS is used to *send* the SMS text message from the

message storage of the GSM / GPRS modem. "+CMSS: 20" tells us that the reference number assigned to the SMS text message is 20. [7]

```
AT
OK
AT+CMGF=1
OK
AT+CMGW="+85291234567"
> A simple demo of SMS text messaging.
+CMGW: 1

OK
AT+CMSS=1
+CMSS: 20

OK
```

Figure 7. Example 2 for sending SMS

3.2.2. Receiving SMS from phone to computer

For reading SMS text messages sent from the user's phone, they need to use the AT command +CMGL is used to list all SMS text messages in the message storage of the GSM / GPRS modem. "REC READ" indicates both SMS text messages have been read before. [8]

```
AT
OK
AT+CMGF=1
OK
AT+CMGL="ALL"
+CMGL: 1,"REC READ","+85291234567",,"06/11/11,00:30:29+32"
Hello, welcome to our SMS tutorial.
+CMGL: 2,"REC READ","+85291234567",,"06/11/11,00:32:20+32"
A simple demo of SMS text messaging.

OK
```

Figure 8. Example for receiving SMS[8]

The user can check **Error! Not a valid bookmark self-reference.** or the link (AT Commands, GSM AT command set, 2018) for a full list of AT commands one can use.

Table 2. AT commands for receiving and reading SMS [8]

AT command	Meaning
+CNMI	New message indications
+CMGL	List messages
+CMGR	Read messages
+CNMA	New message acknowledgement

4. Conclusion

Connection of a smart phone to a computer using the USB and the Hayes protocol to send / receive SMSs can be easily done with a couple of AT commands and an emulator, but it can also be built from scratch using a programming language C#. I now have a better understanding of the things I have used 10 years ago.

Bibliography

- [1] (Hayes command set, 2020)
https://en.wikipedia.org/wiki/Hayes_command_set#Syntactical_definitions
- [2] (All You Wanted To Know About AT And GSM AT Commands, 2019)
<https://www.electronicsforu.com/resources/gsm-at-commands>
- [3] (20.2. Selecting the Operating Mode (AT+CMGF), 2021)
<https://www.developershome.com/sms/operatingMode2.asp>
- [4] (C# Tutorial - How to Send SMS Message C#.NET Winforms | FoxLearn, 2016)
<https://www.youtube.com/watch?v=pZXJ492Rzy4>
- [5] (send sms by GSM module using hyperterminal, 2014)
<https://www.youtube.com/watch?v=fHqqwSiIHVs>
- [6] (The GSM character set, 2017)
<https://www.clockworksms.com/blog/the-gsm-character-set/#:~:text=The%20GSM%20character%20set%20is,the%20Extended%20GSM%20character%20set.>
- [7] (10. How to Send SMS Messages from a Computer / PC?, 2021)
<https://www.developershome.com/sms/howToSendSMSFromPC.asp>
- [8] (11. How to Receive SMS Messages Using a Computer / PC?, 2021)
<https://www.developershome.com/sms/howToReceiveSMSUsingPC.asp>
- [9] (AT Commands, GSM AT command set, 2018)
<https://www.engineersgarage.com/tutorials/at-commands-gsm-at-command-set/>