

Using the Modem

Originally published in NL #114

by George Johnson

[Using the Modem](#) | [Programming with the Modem](#) | [User Interface](#) | [Open the Modem](#) | [COM Number](#) | [Sending Commands to the Modem](#) | [Result of a Command](#) | [Closing the Modem](#) | [Demo](#)

Programming with the Modem

I know it's a semi-random subject, but have you ever wanted to create a program that involved your modem in some way? This program could be an answering machine, a phone dialer, an automated calling system, a highly complex piece of software that reveals the identity of the lady on the error recordings from the phone company, etc. Well with LB, It's actually quite easy to design programs of this type. In the next few paragraphs, I will be explaining the basics of modem manipulation through Liberty Basic 3.

User Interface

The first element necessary to build your program is to think about the user interface. Most programs, no matter what they do, are going to need user input eventually. Although console applications are acceptable, Graphical User Interfaces are much more appealing to the eye and easier to use. FreeForm, which is included with Liberty Basic, can help you design an interface.

Open the Modem

The next step is opening the modem so that you can send commands to it. Most of the time, the following line of code is acceptable for opening the modem.

```
open "COM3:9600,n,8,1,ds0" for random as #comm
```

Usually, the only thing that you would have to change is the COM number. (The identifying number of the communications port the modem is installed on.) Please note that a modem cannot be opened without the correct COM number, or if it is already in use by another program. COM numbers can be different on different computers, so you will probably want to prompt the user for their COM number and contain it in a variable. Example:

```
prompt "Please enter your modem's COM number.";comnum$
open "COM" + comnum$ + ":9600,n,8,1,ds0" for random as #comm
```

COM Number

Other forms of user input are also acceptable. To find your modem's COM number, go to your Windows Control Panel, click on the modem's icon, single click on the modem you want to use, then click properties. That modem's COM settings should be listed.

Sending Commands to the Modem

Next is the process of sending commands to the modem. This is just like saving to a text file with Liberty Basic. The format is:

```
print #handle, "command"
      or,
print #comm, "ATZ"
```

I'm not going to list all the commands in the **Hayes Command Set** since there are so many, but I will list some. Every command must start with the prefix "AT", which means attention. ATZ resets the modem to its default state. ATH1 takes the modem off the hook, while ATH0 hangs up. Another useful command is ATDT(number). This command uses tone dialing to dial the number immediately following the command. ATM0 mutes the modem speaker, and ATL(0-3) changes the speaker's volume level. AT&F followed by ATS0=(any number) is a great command. It tells the modem to scan for incoming calls, then answer after the phone rings the number of times specified in the second command. This is useful for answering machine software. For a complete list of modem commands, please refer to the Hayes Command Set by [\[clicking here\]](#). With these commands, the modem can do virtually everything a modem's good for.

Result of a Command

When you send commands, your program is probably going to need to know the outcome. To find the result of a command, use the following code.

```
if lof(#comm) > 0 then result$ = input$(#comm, lof(#comm))
```

If there is any information about the preceding commands, this one line of code will get it for you. The result will be contained in the variable result\$, which is usually in the following format:

ATZ

OK

or,
ATH2

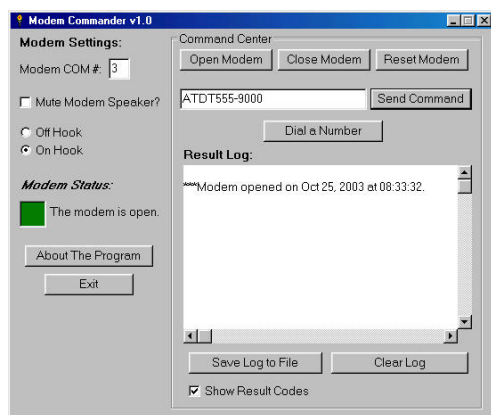
ERROR

Closing the Modem

Closing the modem is also like a file. You simply insert this code when you want to close it:

```
close #handle
```

This will free the modem for use by other programs. For more information on this subject, see the LB help file, and terminal.bas which is included with LB. One good development tool for this type of software, is Modem Commander v1.0 by, well... me. I designed it as a development and diagnostic tool, and it has served me well. It can be obtained by clicking [here](#). You can send commands via the controls in the command center groupbox, or play with some other, more automated, options. This is a good way to try out program structures and new commands. If you have any questions about this subject, email me at george@gasoftware.net. - George Johnson (www.gasoftware.net)



Demo

MODEM COMMANDER

```
'Modem Commander v1.0
'A Product of G.A.Software
'Copyright G.A.Software 2003
'www.gasoftware.net
'support@gasoftware.net
```

```
'The code in this program is pretty easy to understand, but if you need any help
'with the program or code, just email me.
```

```
NOMAINWIN
```

```
'Setup main window with handle of #main
WindowWidth = 550
WindowHeight = 466
UpperLeftX=int((DisplayWidth-WindowWidth)/2)
UpperLeftY=int((DisplayHeight-WindowHeight)/2)
graphicbox #main.g, 10, 192, 30, 30
texteditor #main.log, 195, 152, 325, 200
groupbox #main.groupbox7, "Command Center", 180, 2, 350, 425
statictext #main.statictext1, "Modem COM #:", 10, 37, 92, 20
textbox #main.com, 110, 33, 25, 25
checkbox #main.checkbox3, "Mute Modem Speaker?", [mute], [unmute],
10, 72, 165, 20
checkbox #main.cb4, "Show Result Codes", [show], [noshow], 200, 397,
165, 20
radiobutton #main.off, "On Hook", [onhook], [reset], 10, 127, 77,
20
radiobutton #main.on, "Off Hook", [offhook], [reset], 10, 107, 78,
20
button #main.button9, "Open Modem", [open], UL, 190, 22, 105, 25
button #main.button8, "Close Modem", [close], UL, 300, 22, 105, 25
button #main.button9, "Reset Modem", [reset], UL, 410, 22, 105, 25
button #main.button13, "Exit", [quit], UL, 38, 274, 100, 25
button #main.button15, "About The Program", [about], UL, 17, 243,
145, 25
textbox #main.send, 190, 67, 210, 25
button #main.default, "Send Command", [send], UL, 405, 67, 115, 25
statictext #main.statictext22, "Result Log:", 195, 132, 85, 20
button #main.button24, "Save Log to File", [savelog], UL, 200, 362,
150, 25
button #main.button25, "Clear Log", [clearlog], UL, 360, 362, 155, 25
statictext #main.statictext26, "Modem Settings:", 10, 7, 120, 20
statictext #main.statictext27, "Modem Status:", 10, 167, 155, 20
statictext #main.stat, "Modem is not open.", 45, 197, 130, 20
button #main.button30, "Dial a Number", [dial], UL, 285, 102, 135, 25
open "Modem Commander v1.0" for dialog_nf as #main
```

```
print #main.g, "down; fill red; flush"
print #main, "font ms_sans_serif 10"
print #main.statictext22, "!font ms_sans_serif 10 bold"
print #main.statictext26, "!font ms_sans_serif 10 bold"
print #main.statictext27, "!font ms_sans_serif 10 bold italic"
PRINT #main.off, "set"
PRINT #main.cb4, "set"
PRINT #main, "trapclose [quit]"
PRINT #main.com, "3"
PRINT #main.send, "!setfocus"
ONCOMERROR [error]
show=1
' variable used to determine whether the user wants the program to show
result codes.
call EnableWindow hwnd(#main.default),0
call EnableWindow hwnd(#main.button8),0
call EnableWindow hwnd(#main.button9),0
call EnableWindow hwnd(#main.button30),0
call EnableWindow hwnd(#main.cb4),0
call EnableWindow hwnd(#main.checkbox3),0
call EnableWindow hwnd(#main.off),0
call EnableWindow hwnd(#main.on),0
WAIT

[about]
NOTICE "Modem Commander v1.0"+chr$(13)+
"Modem Commander is a control and diagnostic tool for all modems on COM"
+chr$(13)+_

+chr$(13)+_
"+chr$(13)+_
"Copyright G.A.Software 2003"+chr$(13)+_
"www.gasoftware.net"
PRINT #main.send, "!setfocus"
WAIT

[error]
'When a COM error is detected, the program will automatically come to
this branch label.
NOTICE "Modem Commander v1.0"+chr$(13)+
"There was a problem with the last command sent. Please try this command
again,"+chr$(13)+
"and if it does not work, email G.A.Software at: support@gasoftware.net"
+chr$(13)+"+chr$(13)+
```

```
"(Note: Make sure you have your modem's correct COM number.)"  
PRINT #main.send, "!setfocus"  
WAIT
```

```
[show]  
'Code to show the result codes to the user.  
IF com<>1 THEN PRINT #main.cb4, "reset":PRINT #main.send,  
"!setfocus":WAIT ' If COM is not open, WAIT.  
show=1  
PRINT #com, "ATQ0"  
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))  
' Get rid of any remaining result codes.  
PRINT #main.send, "!setfocus"  
WAIT
```

```
[noshow]  
'Code to prevent the user from seeing result codes.  
IF com<>1 THEN PRINT #main.cb4, "set":PRINT #main.send,  
"!setfocus":WAIT ' If COM is not open, WAIT.  
show=0  
PRINT #com, "ATQ1"  
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))  
'Get rid of result codes from the command, "ATQ1"  
PRINT #main.send, "!setfocus"  
WAIT
```

```
[offhook]  
IF com<>1 THEN PRINT #main.off, "set":NOTICE  
"Modem Commander v1.0" + chr$(13) +  
"You must first open the modem.":PRINT #main.send,  
"!setfocus":WAIT ' If COM is not open, WAIT.  
PRINT #com, "ATH1"  
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))  
'Get rid of result codes from the command, "ATH1"  
PRINT #main.send, "!setfocus"  
WAIT
```

```
[onhook]  
IF com<>1 THEN PRINT #main.send, "!setfocus":WAIT  
' If COM is not open, WAIT.  
PRINT #com, "ATH0"  
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))  
'Get rid of result codes from the command, "ATH0"  
PRINT #main.send, "!setfocus"  
WAIT
```

```
[mute]
IF com<>1 THEN PRINT #main.checkbox3, "reset":PRINT #
main.send, "!setfocus":WAIT ' If COM is not open, WAIT.
PRINT #com, "ATM0"
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))
'Get rid of result codes from the command, "ATM0"
PRINT #main.send, "!setfocus"
WAIT

[unmute]
IF com<>1 THEN PRINT #main.send, "!setfocus":WAIT
' If COM is not open, WAIT.
PRINT #com, "ATM2"
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))
'Get rid of result codes from the command, "ATM2"
PRINT #main.send, "!setfocus"
WAIT

[reset]
IF com<>1 THEN PRINT #main.send, "!setfocus":WAIT
' If COM is not open, WAIT.
PRINT #com, "ATZ"
PRINT #main.log, ""
IF show=1 THEN PRINT #main.log, ""+chr$(13)+
"***Modem reset on " +date$() + " at "+time$()+". "
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))
'Get rid of result codes from the command, "ATZ"
PRINT #main.send, "!setfocus"
WAIT

[close]
IF com<>1 THEN PRINT #main.send, "!setfocus":WAIT
' If COM is not open, WAIT.
CLOSE #com
com=0 'variable that shows the computer that the modem is not open.
PRINT #main.g, "fill red;flush"
PRINT #main.stat, "Modem is not open."
PRINT #main.off, "set"
PRINT #main.checkbox3, "reset"
IF show=1 THEN PRINT #main.log, ""+chr$(13)+
"***Modem closed on " + date$() + " at " + time$() + ". "
PRINT #main.send, "!setfocus"
call EnableWindow hwnd(#main.default),0
call EnableWindow hwnd(#main.button8),0
call EnableWindow hwnd(#main.button9),0
call EnableWindow hwnd(#main.button30),0
```

```
call EnableWindow hwnd(#main.cb4),0
call EnableWindow hwnd(#main.checkbox3),0
call EnableWindow hwnd(#main.off),0
call EnableWindow hwnd(#main.on),0
WAIT

[savelog]
' Save the contents of #main.log into a txt file
FILEDIALOG "Save log file as...", "*.txt",file$
IF file$="" THEN WAIT
IF file$ = "*.txt" THEN WAIT
PRINT #main.log, "!contents? text$"
IF right$(file$,3) <> "txt" THEN OPEN file$ + ".txt" FOR OUTPUT AS #
log ELSE OPEN file$ FOR OUTPUT AS #log
PRINT #log, text$
CLOSE #log
text$=""
file$=""
PRINT #main.send, "!setfocus"
WAIT

[clearlog]
'clears #main.log
PRINT #main.log, "!cls"
PRINT #main.send, "!setfocus"
WAIT

[open]
IF com=1 THEN WAIT
PRINT #main.com, "!contents? com$"
OPEN "COM"+com$+":9600,n,8,1,ds0" FOR random AS #com
PRINT #main.g, "fill darkgreen; flush"
PRINT #main.stat, "The modem is open."
IF show=1 THEN PRINT #main.log, ""+chr$(13)+
"***Modem opened on " + date$() + " at " + time$() + "."
com=1 'variable that shows the program that the modem is open.
PRINT #main.send, "!setfocus"
call EnableWindow hwnd(#main.default),1
call EnableWindow hwnd(#main.button8),1
call EnableWindow hwnd(#main.button9),1
call EnableWindow hwnd(#main.button30),1
call EnableWindow hwnd(#main.cb4),1
call EnableWindow hwnd(#main.checkbox3),1
call EnableWindow hwnd(#main.off),1
call EnableWindow hwnd(#main.on),1
WAIT
```



```
[dial]
'Code to dial a phone number using a modem. See terminal.bas for a more
detailed description.
IF com<>1 THEN NOTICE
"Before you can dial a number, you must open the modem.":PRINT #main.send, "!setfocus":WAIT
' If COM is not open, WAIT.
PROMPT "What number would you like to dial?";dial$
IF dial$<>" " THEN PRINT #com, "ATDT"+dial$
PRINT #main.log, ""+chr$(13)+"***Dialing "+dial$
if lof(#com) > 0 then trash$ = input$(#com, lof(#com))
'Get rid of result codes from the command, "ATDT"
PRINT #main.send, "!setfocus"
WAIT

[send]
' Code to send a custom command entered into the command box by the user.
IF com<>1 THEN NOTICE "Modem Commander v1.0" + chr$(13) +
"First you must open the modem.":PRINT #main.send,
"!setfocus":WAIT ' If COM is not open, WAIT.
PRINT #main.send, "!contents? command$"
PRINT #main.send, " "
PRINT #com, command$
IF command$ = "ATH1" THEN PRINT #main.on, "set"
IF command$ = "ATH0" THEN PRINT #main.off, "set"
IF command$ = "ATM0" THEN PRINT #main.checkbox3, "set"
IF command$ = "ATM1" THEN PRINT #main.checkbox3, "reset"
IF command$ = "ATM2" THEN PRINT #main.checkbox3, "reset"
IF command$ = "ATM3" THEN PRINT #main.checkbox3, "reset"

'If the command is one of these, the main window updates its controls.
IF command$ = "ATL0" THEN PRINT #main.checkbox3, "reset"
IF command$ = "ATL1" THEN PRINT #main.checkbox3, "reset"
IF command$ = "ATL2" THEN PRINT #main.checkbox3, "reset"
IF command$ = "ATL3" THEN PRINT #main.checkbox3, "reset"
if lof(#com) > 0 then response$ = input$(#com, lof(#com))
'Get the modems response to the command.
PRINT #main.log, " "
IF show=1 THEN PRINT #main.log, response$
'print the response to the main window if the user wants the program to do so.
PRINT #main.send, "!setfocus"
WAIT
```

```
[quit]
'Code to exit the program.
CONFIRM "Are you sure you want to quit?";ans$
IF ans$="no" THEN PRINT #main.send, "!setfocus":WAIT
IF com=1 THEN PRINT #main.g, "fill red; flush":PRINT #
main.stat, "Modem is not open.":PRINT #main.off, "set":PRINT #
main.checkbox3, "reset":CLOSE #com
CLOSE #main
END

Sub EnableWindow hWnd, flag
  'Enable=1 Disable=0
  CallDLL #user32, "EnableWindow",_
  hWnd As Long, flag As long, result As long
End Sub
```

[Using the Modem](#) | [Programming with the Modem](#) | [User Interface](#) | [Open the Modem](#) | [COM Number](#) | [Sending Commands to the Modem](#) | [Result of a Command](#) | [Closing the Modem](#) | [Demo](#)