

```

/*****
/* C64Net WiFi Firmware Software *
/* for the Commodore 64 and 128 *
/*****
////////////////////////////////////

```

This disk includes applications written to work with the C64Net WiFi modem running firmware version 2.0 or better. If this is your first time using the modem, you should start with the configure program to get your modem connected to the internet. You can then use the FTP, TELNET, IRC, WGET, and other programs whenever you wish to communicate with the internet. The disk includes numerous machine language binaries (pml64.bin, pml128.bin, pmlvic.bin, x-xfer64.bin, x-xfer128.bin, rds64.bin, and up9600.bin) which are required for some the several programs provided.

You can also use any Commodore compatible terminal programs with the C64Net WiFi modem, such as Novaterm or CCGMS for the C64, Desterm for the C128 in 128 mode, or Mighty Term for the VIC-20. When using one of these programs, you should consult the manual on the C64Net WiFi Firmware command mode for instructions on how to use the Hayes-like commands to connect your modem to the internet.

The disk includes the source assembly code for the pml64, pml128, pmlvic, up9600, rds64, and the demo SwiftLink Kernal Wedge driver binary. These source files are in LADS format. The disk also includes the LADS assembler for the C64 to re-assemble the binaries from the source files. To use the LADS assembler, you should use a C64 or C128 in C64 mode, and load the LADS assembler using LOAD"LADS",8,1. Once the assembler is loaded, clear the screen and type the name of the source file to assemble in the upper left-hand corner of the screen. Once that is done, use SYS11000 to begin the assembly, which will create the .bin files.

The disk lastly includes the program "emutil", which is a C64, C128, or VIC-20 program to creating and dissolving .D64 images, such as those downloaded by the d64wget program discussed below.

Below you will find information on the application programs included on this disk, as well as instructions on using the PML64, PML128, PMLVIC, and UP9600 machine language libraries from BASIC or machine language.

BUT I DON'T HAVE THE DISK?!

```

*****

```

Bootstrap software downloader for Commodore 64 with c64wifi modem with firmware rev 2.4 and better. Before you type in and run this program, you should make sure your modem is connected to a wifi hotspot. You can use any C64/C128/VIC-20 terminal program, set to 1200 baud, in ASCII mode. From inside the terminal mode of your program, you can type ATW to get a list of local wifi hotspots. You can then use ATW"<hotspot name/ssid>,<password>" to connect. You should then enter ATZ&W to save the new settings.

If you don't have access to a terminal program, you can type this little program in, save, and run it, and then enter ATW to get a list of local wifi hotspots, and then use ATW"<hotspot name/ssid>,<password>", and if the connection is successful, enter ATZ&WE1&P1 to save the settings.

```

10 open5,2,0,chr$(8):poke665,73-(peek(678)*30)
15 print#5,"at&p1e1":print#5,"at&p1e1":printchr$(14);
20 get#5,a$:ifa$<>""thenprinta$;
30 geta$:ifa$<>""thenprint#5,a$;
40 goto20

```

Once your modem is connected to wifi, switch your C64 to lowercase mode by pressing shift-C= and then type in the following program and save it to a disk:

```

10 open5,2,0,chr$(8):pa=peek(678)*30:poke665,73-pa
15 a3$=chr$(3):print#5,a3$;
20 input"enter filename";f$:iff$=""thenstop
30 input"enter drive";dv:ifdv<8thenstop
35 print#5,a3$;
40 get#5,a$:ifa$<>""then40
50 dl$=mid$(str$(99+pa),2):print#5,"atze0r0&p1f1s45=2s44="+dl$
60 input#5,a$:ifa$<>"OK"anda$<>"ok"then60

```

```

70 print#5,"at&g";chr$(34);"www.zimmers.net:80/otherprojs/";f$;chr$(34)
75 print#5,chr$(19);
80 input#5,b$:ifb$="["then100
90 printb$:print"reset computer and check wifi":close5:stop
100 input#5,ch,ln,c8,b$:ifb$<>]"thenprint#5,a3$:goto90
110 n8=0:open8,dv,8,"@0:"+f$+",p,w":print"reading...";
120 m=49152:ni=i+(ln/20):print#5,chr$(17);:fori=1toln
130 input#5,a:n8=n8+a:ifn8>255thenn8=n8-256
140 pokem,a:m=m+1:ifm>53240thengosub200:m=49152
145 ifi>nithenni=i+(ln/20):printint(i/ln*100);"%",
150 next:print#5,a3$;"atz":ifn8<c8thenprint"checksum fail. try again."
160 print"100 %",:gosub200:close8:close5:print"done":poke49152,0:poke51200,0
199 end
200 print#5,chr$(19);:print:print"writing...":ti$="000000":tt=ti+200
205 ifti<ttthen205
210 forx=49152tom-1:print#8,chr$(peek(x));:nextx
220 ifi<lthenprint"reading...":tt=ti+200
225 ifti<ttthen225
230 print#5,chr$(17);:return

```

The program should be run 3 times. Once with each of these filenames:

1. pml64.bin (that's pee em el, not pee em one)
2. up9600.bin
3. ftp

If you have any problems or errors here, such as the program doing nothing after the "enter drive" prompt, or it freezing at a certain download percentage, or getting a "file data" error or some other error, then do the following: First, hold down the RUN/STOP key and sharply strike the RESTORE key at the same time. You should see a "READY." prompt. Now enter: "close8" and hit RETURN. This should properly close the file if it was open, and turn the light on your disk drive off. Next, you should turn your computer off and turn it back on to reset it. Then re-load the little downloader program above and run it again. Don't get frustrated. This dumb little program is only to bootstrap you. Once you have the files "ftp", "pml64.bin", and "up9600.bin", things will go much smoother and sweeter.

Next, load and run ftp from either a C64, or a C128 in 64 mode by entering LOAD"FTP",8. RUN this program and use it to connect to ftp.zimmers.net anonymously. Once you are logged in, enter the following commands:

1. cd pub/projects/c64wifi
2. ls
3. get pml128.bin,p
4. get telnet,p
5. ... repeat for all the files you saw in step 2.
6. ... repeat step 2 if you forget which files remain

#### CONFIGURE

\*\*\*\*\*

The Configure program is for setting up or altering the WiFi Hotspot, also known as a Wireless Internet Router, that your modem uses to communicate with the internet. It also allows you to modify the modems built in phone book. It is written in BASIC, but uses the PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below). This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. To run the program, enter LOAD"CONFIGURE",8 and then RUN.

When the program runs, it will initialize the C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, the program will scan the area for WiFi Hotspots and provide you a list to select from. If you do not see your own wireless router listed, or do not know the name, consult your ISP's internet equipment documentation. After you have selected the appropriate hotspot, you will be prompted for the password. After the password is entered, the program will test the settings and, if successful, save the settings to the flash memory in your modem it will be remembered even

after the computer is turned off. When wifi configuration is complete, and the modem passes the connection test, you can modify the modems internal phone book.

The modems internal phone book maintains a list of numbers and associated internet addresses and configurations. These numbers can then be used by third party software to 'dial' to those sites using the modems ATD command. A phone number, in this case, is any large number, preferably 7 digits or more, that is assigned to a particular internet host and port. Each host can then be configured for ASCII-PETSCII translation, TELNET code translation, XON/XOFF flow control, and/or local ECHO (half vs full duplex). Every modification you make is saved to the modem and available after reboots. This system can be used to connect to the online Q-Link service online using the existing Q-Link software.

#### FTP

\*\*\*\*\*

File Transfer Protocol program supports a minimal number of features to connect to FTP Servers all over the internet in order to download and/or upload programs. It is written in BASIC, but uses the UP9600.BIN and PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below).

This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. If you run it on a breadbox Commodore 64 or 64C with UP9600 enabled on the modem (default), it will operate at 9600 baud during your session. If you run it on a Commodore 128 in C128 mode and 80 columns, it will operate at 2400 baud during your session. If you run it on a Commodore 128 in C64 mode, a VIC-20, or C128 mode and 40 columns, it will remain at 1200 baud speed. To run the program, enter LOAD"FTP",8 and then RUN.

When you start the program, it will initialize your C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, you will be given a menu where you can enter the host name of the ftp server to connect to, and the login and password to use. If you plan to login with the name "anonymous", you should use your email address as your password. The last option on the menu is the drive number to upload files from and download files to. When all options are correct, hit RETURN to connect.

At the FTP prompt, you may enter one of the following commands:

DIR : list the files in the current directory  
 CD [DIRECTORY OR PATH] : change to a new directory  
 QUIT : exit the ftp program  
 HELP : show the commands supported by the client and the server  
 GET [FILENAME,P] : download the program with the name filename as a PRG. Use ,S for SEQ.  
 PUT [FILENAME,P] : upload the program with the name filename which is a PRG file. Use ,S for SEQ.  
 DEL [FILENAME] : delete the given file from the server  
 LCD [DEVICE NUMBER or CMD Hard Drive PATH] : change the active local upload/download drive, or change which directory you are using if the active drive is a UIEC or CMD Hard Drive compatible.  
 LDIR [MASK] : get a directory of the active local drive.  
 LDEL [FILENAME] : delete the given local file from the active drive.

#### WGET

\*\*\*\*\*

WGET program is for downloading files and pages from a standard HTTP web address (URL). It is written in BASIC, but uses the UP9600.BIN and PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below).

This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. If you run it on a breadbox Commodore 64 or 64C with UP9600 enabled on the modem (default), it will operate at 9600 baud during your session. If you run it on a Commodore 128 in C128 mode and 80 columns, it will operate at 2400 baud during your session. If you run it on a Commodore 128 in C64 mode, a VIC-20, or C128 mode and 40 columns, it will remain at 1200 baud speed. To run the program,

enter LOAD"WGET",8 and then RUN.

When you start the program, it will initialize your C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, you will be given a menu where you can enter the url of the file or page to download, the unit number of the disk drive to download the file or page to, and the name of the file to write to the disk. Your filename should end with ,P to create a PRG file, and ,S to create a SEQ file. For example, a filename of "mypage,s" will create a file called "mypage" of type SEQ. When all options are correct, hit RETURN to connect and begin the download. When the download completes, the program will exit.

## IRC

\*\*\*\*\*

IRC program is for chatting with people on free internet IRC Servers. It is written in BASIC, but uses the PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below). This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. If you run it on a breadbox Commodore 64 or 64C with UP9600 enabled on the modem (default), it will operate at 9600 baud during your session. If you run it on a Commodore 128 in C128 mode and 80 columns, it will operate at 2400 baud during your session. If you run it on a Commodore 128 in C64 mode, a VIC-20, or C128 mode and 40 columns, it will remain at 1200 baud speed. To run the program, enter LOAD"IRC",8 and then RUN.

When you start the program, it will initialize your C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, you will be asked to enter the host name of the IRC Server to connect to. Next you will be prompted for the port number that the server listens on. The program will then attempt to connect to the host and port. If successful, you will be prompted to enter a nickname, which is the name you will be known by during chatting. If your nickname is already in use by someone else, you will eventually be prompted to enter a new one. Either way, the program will display the servers MOTD and other information. This could take several minutes. It's best to just wait until you see the message "End of MOTD" to begin typing.

After the MOTD message, you may type at any time. If you have joined a chat channel, you will automatically see messages as they appear in the channel. When you begin typing, all messages from the server are suspended until you hit RETURN. Since the IRC client will appear to the server to be idle while you are typing, it is best to spend as little time as necessary in message and command entry mode. You may enter ? followed by RETURN to see the IRC commands supported by this client.

Commands include:

/QUIT : exit the IRC chat program

/JOIN #channelname : join the new channel named "#channelname"

\* Anything else entered will be assumed to be a message to sent to the channel you have already joined.

## TELNET

\*\*\*\*\*

TELNET program is for communicating with a TELNET server as a terminal, or with a TELNET-like server such as a Multi-User Dungeon game (a MUD). It is written in BASIC, but uses the PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below). This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. To run the program, enter LOAD"TELNET",8 and then RUN.

When you start the program, it will initialize your C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, you will be shown a menu with the following options:

**Dial from Phonebook:**

This will let you connect to a server saved in your phonebook on disk. See below for information about terminal mode while connected to a server

**Modify Phonebook:**

This will let you add or modify servers to save in your phonebook on disk. These can be then "Dialed" to quickly later on.

**Quick Connect:**

This will let you manually enter a server host name and port to immediately connect to. See below for information about terminal mode while connected to a server.

**Terminal Mode:**

This will put you into terminal mode without connecting to any server. In this case, you would be in the modem firmware's Command Mode. You should consult the documentation for the C64Net WiFi Firmware for more information on this mode. Use the F1 key or the C128's ESC key to exit terminal mode.

**Quit:**

Exit the TELNET program

When you have connected to a server, you will be in ANSI terminal mode. You can exit the server and disconnect at any time by hitting the F1 key on your C64, or the ESC key on your C128. When connected to a server, the modem will translate as much ASCII/ANSI into mostly viewable PETSCII for your Commodore computer. However, you will notice that underscores look like back-arrows. ANSI colors will be translated to Commodore PETSCII colors wherever possible. Your own keystrokes will also be translated to ASCII/ANSI as you type.

**CBMTERM**

\*\*\*\*\*

C= TERM program is for communicating with a telnet-style Commodore Graphics BBS servers as a terminal. It is written in BASIC, but uses the PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below). This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. To run the program, enter LOAD"CBMTERM",8 and then RUN.

When you start the program, it will initialize your C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, you will be shown a menu with the following options:

**Dial from Phonebook:**

This will let you connect to a server saved in your phonebook on disk. See below for information about terminal mode while connected to a server

**Modify Phonebook:**

This will let you add or modify servers to save in your phonebook on disk. These can be then "Dialed" to quickly later on. You will receive an option for a Host Name, a Port Number, and Local Echo. Consult your online BBS list for hosts and ports of the BBSes you wish to connect to. The Local Echo option should usually be answered with NO, as most BBSes will echo for you, and having the modem do local echo will corrupt X-Modem transfers.

**Quick Connect:**

This will let you manually enter a server host name and port to immediately connect to. See below for information about terminal mode while connected to a server.

**Terminal Mode:**

This will put you into terminal mode without connecting to any server. In this case, you would be in the modem firmware's Command Mode. You will be prompted as to whether you would like Local Echo turned on. If you plan to remain in Command Mode and not connect to anything, you can say Yes. However, if you plan to connect to a BBS, or especially if you plan to do any file transfers, you should definitely answer NO. You should consult the documentation for the C64Net WiFi Firmware for

more information on this mode. Use the F1 key or the C128's ESC key to exit terminal mode. See below for more you can do after exiting terminal mode.

Quit:

Exit the CBMBBS program

When you have connected to a server, you will be in CBM PETSCII/Graphics mode. You can exit the terminal at any time by hitting the F1 key on your C64, or the ESC key on your C128. If you are offline, you will immediately return to the main menu. However, if you are still online, you will have the option to hangup and return to the main menu, upload a file using X-Modem CRC, download a file, or continue back to terminal mode.

D64WGET

\*\*\*\*\*

D64WGET program is for downloading and writing full emulator disk images (such as .D64, .D71, .D81, .DNP, etc) files from a standard HTTP web address (URL) and write them to a formatted, blank diskette. It is written in BASIC, but uses the UP9600.BIN and PML64.BIN, PMLVIC.BIN, or PML128.BIN libraries for speed (see below).

This program requires a C64, C128 in either 64 or 128 mode, or a VIC-20 with 16K expansion or better, and your C64Net WiFi Modem configured for 1200 baud, which is the default factory configuration. If you run it on a breadbox Commodore 64 or 64C with UP9600 enabled on the modem (default), it will operate at 9600 baud during your session. If you run it on a Commodore 128 in C128 mode and 80 columns, it will operate at 2400 baud during your session. If you run it on a Commodore 128 in C64 mode, on a VIC-20, or C128 mode and 40 columns, it will remain at 1200 baud speed. To run the program, enter LOAD"D64WGET",8 and then RUN.

When you start the program, it will initialize your C64Net modem with certain commands. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

After initialization, you will be given a menu where you can enter the url of the disk image file to download and write, and the unit number of the disk drive containing the \*formatted\* blank disk to write the image to. When all options are correct, hit RETURN and answer Y to the confirmation to connect and begin the download and disk write. When the download completes, which could take awhile, the program will exit.

TELNETD64

\*\*\*\*\*

TELNETD program is for controlling your C64 remotely over the internet. It is written mostly in machine language which is contained in the file "RDS64.BIN", with the source code in "RDS64.BAS". The program TELNETD64 is written in BASIC, and is for configuring and launching your server. To run the program, enter LOAD"TELNETD64",8 and then RUN.

This program requires that your C64Net WiFi Modem be configured for 1200 baud on reset, which is the default factory configuration.

When you start the program, it will load "rds64.bin" and prompt you for certain server settings, such as how much time to allow users to remain on your computer in a single session (Time Limit), the amount of time to allow users to remain online when they haven't been typing anything (the Idle Time), remain online when they aren't typing anything, the security flag, the port to listen on, and the modem initialization string. You should be cautious with modifying that last, as it may affect the servers functionality.

After the prompts, the c64 will start up the server itself. The server will initialize your C64Net modem with the above modem initialization string. If you receive an error at this point, or if the program hangs for longer than a minute, try turning your computer off and on again to clear memory and reset the modem, and then load and run it again.

Once the server is started, you will be right back to the normal BASIC "READY." prompt. When a user connects, they will be able to type in commands, which you can watch them do. If the security flag was turned on, then certain BASIC commands, such as "POKE" and "SYS" will be disabled, as will the ability to OPEN channel 2 (the modem). Should the user violate the idle or session time limit, they will be immediately disconnected.

You should be careful with what disk drives you leave connected and which disks you leave in those drives, as online users may be able to modify them. You should also be aware that this program is in an Alpha state. For example, a C64 with JiffyDOS will work with the TELNETD software, but any attempts to use JiffyDos "@" commands will hang up the server. It is also possible there are other ways to hang the server up, even with security turned on. If you discover any new ways, please email the author.

To connect to your server from the internet, you should set your ISP router to forward connections to the TELNETD64 port (default 6400) to the ip address of your modem.

The only way to shut down your server is either to reset the computer, or turn it off and back on again.

#### WEATHER64

\*\*\*\*\*

Weather is a two player online game for the Commodore 64 originally written for the Commodore PET. This program requires that both players have C64Net WiFi Modems configured for 1200 baud on reset, which is the default factory configuration. To run the program, enter LOAD"WEATHER64",8 and then RUN.

When running the program, the system will ask for one player to act as the host, and the other as the client. The host computer then waits for a connection from the client computer, after which the game can commence. The host player must be familiar with opening firewall ports in order to allow the client to connect to their computer.

In the game, both players compete against each other in this artillery like contest that involves attacking each others houses with the forces of nature.

#### PML64.BIN, PMLVIC.BIN, and PML128.BIN

\*\*\*\*\*

These files, for the C64, VIC-20, and C128 in 128 mode respectively, are machine language support files for performing certain common C64Net modem functions from BASIC. PML64.BIN loads to \$C000 (49152), PMLVIC.BIN to \$5701, and PML128.BIN to \$1300 (4864). They are functionally identical.

To use them from BASIC, make sure to follow two important rules: 1. Load the appropriate binary into memory (pml64.bin, pmlvic.bin, or pml128.bin). and 2. Initialize the variable p\$, by giving it some value, as early in your program as possible. For example: P\$="" would be fine.

Throughout this section, I will refer to the variable ML to refer to the starting address of the library. For the C64, this is always 49152, for the VIC-20 it is 22273, and for the C128, this is 4864. So, when I say ML+3, I mean 49155 on the C64, and 4867 on the C128, etc.

-----

#### BUF1LIN

#### Usage:

sys ML

#### Pre-requisites:

P\$ must be initialized in BASIC

Channel 5 must be OPEN to the modem

#### Outputs:

P\$ contains the string line read from the modem, if any

ML + 19 is 0 to denote success, or 255 for a timeout

ML + 26 contains the CRC8 value of the received string

#### Description:

Reads from the open modem channel until a carriage return (13) is seen, or until 255 characters are read, or until a timeout occurs. The characters are returned in the BASIC variable P\$, although character values 10 and 0 are always ignored. The timeout is about 12 seconds.

-----  
BUFXLIN

Usage:  
sys ML+3

Pre-requisites:  
P\$ must be initialized in BASIC  
Channel 5 must be OPEN to the modem  
ML + 18 contains the number of bytes to read

Outputs:  
P\$ contains the X bytes read from the modem if any  
ML + 18 contains the number of bytes NOT read (0 on success)  
ML + 19 is 255 for a timeout, or any other value for the index of the last carriage return (13) in the returned buffer.  
ML + 26 contains the CRC8 value of the received bytes

Reads from the open modem channel until the number of bytes in ML+18 have been read, or until 253 bytes are read, or until a timeout occurs. The bytes are returned in the BASIC variable P\$. The timeout is about 12 seconds.

-----  
GETPACKET

Usage:  
sys ML+6

Pre-requisites:  
P\$ must be initialized in BASIC  
Channel 5 must be OPEN to the modem

Outputs:  
ML + 18 contains the number of bytes read  
ML + 19 is 255 for a timeout or packet formatting error  
ML + 20 is the packet/channel ID that sent the data (0=none)  
ML + 22 is the size of the packet received  
ML + 24 is the EXPECTED CRC8 of the packet data sent by the modem  
ML + 26 contains the CRC8 value of the received string

Reads a special C64Net WiFi Firmware formatted Packet from modem channel. This function will block until it times out, or a packet is received, so it's best used with the ATF3 flow control to ensure that every time XON is sent to the modem, a single packet header is guaranteed to be sent to the computer. When this method returns, ML+19 should be checked for an error, ML+22 for a non-0 to see if any data was received, or ML+20 to make sure it matches the expected channel (this can be 0 for a non-response when using ATF3). You can also optionally compare ML+26 to ML+24 to check for RS232 errors.

-----  
CRCP

Usage:  
sys ML+9

Pre-requisites:  
P\$ contains the string to calculate the CRC of

Outputs:  
ML + 26 contains the CRC8 value of the received string

Description:  
Calculates the CRC8 of P\$ and returns it.

-----  
BUFAP

Usage:  
sys ML+12

Pre-requisites:  
P\$ must be initialized in BASIC  
ML + 32 contains the channel number to read from  
The above channel must be open

Outputs:  
P\$ contains the bytes read from the channel, if any  
ML + 26 contains the CRC8 value of the received bytes

Description:  
Reads from the open channel whose number is contained in ML+32 until no more bytes are available, or until 254 bytes are read. The bytes are returned in the BASIC variable P\$.

#### UP9600.BIN

\*\*\*\*\*

This file for the Commodore 64 or 64C (NOT the Commodore 128 in 64 mode) is a machine language kernal wedge that supports the 9600 baud UP9600 feature of the C64Net WiFi Modem. It resides in memory at \$C800 (51200), and must be initialized before use by calling SYS 51200.

Once initialized, the normal BASIC commands for using the modem will be modified to support UP9600 modems. The OPEN command can now be opened for 9600 baud use by entering OPEN 5,2,0,CHR\$(14), as 14 is the baud rate code for 9600 baud. GET, PRINT, INPUT and other commands can then be used to read and write to the modem at speed. If the modem channel is opened at any other baud rate, the wedge will remain inactive and the normal CBM Kernal will operate.

Machine language KERNAL routines are likewise wedged to support the UP9600 baud rates. OPEN, CLOSE, CHKIN, CKOUT, CHRIN, GETIN, and CHROUT are all supported. As with BASIC, the wedge only operates when the OPEN command has been issued with a baud rate code of 14. Obviously, the modem must also already be set to 9600 baud for any of this to work.

It should be noted that, due to the way the driver works, IEC/Disk access cannot occur at the exact same time as the modem is receiving data. The driver tries to prevent disk issues by disabling itself whenever an IEC/Disk channel is selected, but this should be kept in mind in your application development.

Below is a slightly more detailed explanation of the routines in UP9600.BIN. Throughout this section, I will refer to the variable UM to refer to the starting address of the library, which is 51200 or \$C800. So, if you see UM+3, it means 51203.

#### INIT

-----

Usage:  
sys UM

Pre-requisites:  
None

Outputs:  
None

Description:  
Initializes the wedge by replacing the standard KERNAL vectors for OPEN, CLOSE, CHKIN, CKOUT, CHRIN, GETIN, and CHROUT with its own routines.

#### INSTALL

-----

Usage:

sys UM+3

Pre-requisites:

Memory locations 247 and 248 point to a 256 byte input buffer

Outputs:

Memory location 789 has something other than the value 234 if this call is successful.

Post-requisites:

You should put a non-zero value into memory location UM+19, such as 1, after calling this function, for the KERNAL wedge to recognize that the driver is waiting for 9600 baud data from the modem.

Description:

Checks for the presence of UP9600 hardware, and if found, Installs and Enables the UP9600 baud driver. This should only be called once, ever. It is called by the OPEN BASIC and KERNAL command automatically. If the driver is disabled with the DISABLE call, then call ENABLE below to re-enable it.

ENABLE

-----

Usage:

sys UM+6

Pre-requisites:

INSTALL must have been called

Memory location 247 and 248 point to a 256 byte input buffer

Outputs:

Memory location 789 has something other than the value 234 if this call is successful.

Description:

Enables or Re-Enables the UP9600 baud driver. It is called by INSTALL automatically.

DISABLE

-----

Usage:

sys UM+9

Pre-requisites:

ENABLE or INSTALL must have been called

Outputs:

Memory location 789 has the value 234 when this completes.

Description:

Disables the UP9600 baud driver, allowing IEC/Disk access to be performed.

RSOUT

-----

Usage:

sys UM+12

Pre-requisites:

INSTALL or ENABLE must have called

The accumulator .A contains the byte to transmit

Outputs:

None

Description:

Transmits a byte at 9600 baud, in real time.

RSIN

-----

Usage:  
sys UM+15

Pre-requisites:  
INSTALL or ENABLE must have called

Outputs:  
The accumulator .A contains the byte received, if any.  
The Carry flag is Set if there was an error, Clear if a byte is in the accumulator.

Description:  
Reads a byte from the 9600 baud input buffer.

RDS64.BIN

\*\*\*\*\*

This file is the machine language portion of the TELNETD server. It resides as \$c000 and can be activated with SYS49152.

X-XFER64.BIN

\*\*\*\*\*

This file is the X-Modem protocol used by CBMTERM. It resides at \$c800 with the following jump table:

\$c800 SendFile (upload)  
\$c803 Receive Checksum (download)  
\$c806 SendFile (upload)  
\$c809 Receive CRC (download)

All functions require the modem opened to channel 5, and the appropriate file on channel 2. The C= key can be used to abort.

X-XFER128.BIN

\*\*\*\*\*

This file is the X-Modem protocol used by CBMTERM. It resides at \$c800 in bank 0 with the following jump table:

\$c800 SendFile (upload)  
\$c803 Receive Checksum (download)  
\$c806 SendFile (upload)  
\$c809 Receive CRC (download)

All functions require the modem opened to channel 5, and the appropriate file on channel 2. Also be aware that it will copy a small 16 byte bit of bank-switch code to \$1BE0. During transfers, the C= key can be used to abort.