ABOUT

Striketerm 2013 is Freeware.

It is based on the public domain Novaterm 9.6c

written by Nick Rossi. Nick has released the source code to

the public and Alwyz has modded the routines.

The Author of this software makes no warranty of any kind,

either express or implied, including but not limited to implied

warranties of merchantability and fitness for a particular

purpose, with respect to this software and accompanying

documentation.

In no event shall the author be liable for any damages

(including damages for loss of business profits, business

interruption, loss of business information, or other pecuniary

loss) arising out of the use of or inability to use this

program, even if the author has been advised of the possibility

of such damages.

All rights reserved. No part of this document may be reproduced,

in any form or by any means either manually or electronically,

without written permission from the Author.

Permission to web publish this manual was given to VideoCam

Services by the Author on 6 December 2000 when he Emailed the

Striketerm manual to Gaelyne Gasson.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

INTRODUCTION

Striketerm is the best communications software available for the

Commodore 64. It supports today’s high-speed modems that utilize

a direct internet connection; it supports 80-column emulations as well

as traditional Commodore color/graphics; it supports a variety

of memory expansion devices for capturing text; and it supports

all of the most common file transfer protocols, including

Zmodem.

Striketerm is perfect for the Internet, for major on-line

services, and for local Commodore and non-Commodore bulletin

boards. Striketerm provides for all of your telecommunication

needs!

i.1 Software support

Please feel free to write with comments and bug reports.

alwyz@sceneworld.org

i.2 World Wide Web site

https://1200baud.wordpress.com

i.3 Special thanks!

The following people did amazing work to help the

developement of the original Novaterm 9.6c, which

Striketerm is based on.

\* Nick Rossi

\* Edward Piecewicz

\* Tim Phelps

\* Nate Dannenberg

\* Gunther Richter

\* Gaelyne Moranec

\* Daniel Dallmann

\* Peter Fiset

\* Rod Gasson

REPORT THIS AD

Thanks to Peter Fiset of Performance Peripherals, Inc. for

assistance with the RAMDrive driver.

Thanks to Doug Cotton and Mark Fellows of Creative Micro

Designs, Inc. for assistance with the RAMLink driver.

Thanks to Daniel Dallmann for writing the enhanced user port

driver (UP9600).

Thanks to Nicolas Welte for the RRNet/Silversurfer drivers.

Thanks to George Hug for writing those excellent RS-232 routines

and to Transactor Magazine (RIP) for publishing them.

Thanks to Daniel Fandrich for writing and releasing a framework

for the Zmodem protocol. (Mr. Fandrich’s original protocol was

download-only, and output data to the screen. His code laid the

groundwork for completing the protocol.)

Thanks to Kermit Woodall for his WXmodem protocol source code.

Thanks to Gunther Richter for translating the software and

manual into German.

And thanks to everyone who took the time to report bugs, make

suggestions, and register the software. Every bit has helped to

make this software successful.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. MAIN MENU

This section shows you how to start up Striketerm, goes through

the initial configuration, and explains the main menu.

1.1 Starting up the program / The Main Menu

If you are starting Striketerm for the first time, the program

will run in it’s default configuration, which utilizes WinVice.

This can be changed from the main menu by selecting a new Modem.

1.1.1 Standard Fuctions. The keys F1, F3, F5, and F7 are always

assigned to the following functions, no matter where you are

in the program

F1 – Main Menu

F3 – Terminal Mode

F5 – Dialer

F7 – Buffer Menu

1.1.2 Editing Signatures

In terminal mode, there are 4 keys assigned to send signatures.

These are F2, F4, F6, and F8. All 4 signatures can be defined in

this area.

1.1.3 Disk Commands

You can send a command to the currently selected drive at this

menu. You can also read a directory, delete files,

or read the status of a drive.

Note: All disk operations use the “current” disk device. The

current device is displayed at the top of the menu. The current

device number may be changed using the “Disk command” option

(see below).

1.1.3.1 Commands

REPORT THIS AD

You may send a disk command to the current device by selecting

Disk command. In addition to the normal DOS commands, Striketerm

supports a few others:

Directory: The disk directory may be called up by entering a $

symbol. Pattern matches are also allowed. For example, to get a

directory of files whose names start with prt, you would type:

$:prt.\*

Read error channel: Type @ by itself to read the drive’s error

channel.

Select device: The # symbol selects a different disk drive. For

example, to switch to device 9, drive 0, enter: #9,0

1.1.3.2 Disk directory

This option lists the directory of the disk. Pressing the P key

pauses the listing; pressing the S key stops it.

1.1.3.3 Multiple file delete

This option brings up the file selection screen and allows you

to select files to be scratched. Before bringing up a menu of

files, Striketerm prompts you for a pattern match. This allows you

to enter Commodore DOS wildcards to bring up only certain files.

For example, if you entered per\*, only file names beginning with

the letters per are shown. Pressing RETURN here without an entry

brings up all files in the directory (the pattern match “\*” is

used).

The file selection menu brings up the first 21 files on the disk

and displays them in the left-hand column. To select files, move

the menu bar to the file you want and press either RETURN or F3.

This puts the file name in the right-hand column. Repeat this

until you have selected all of the files you want. To call up

the next 21 files from the directory, press F5. If you select a

file you don’t want, move the menu bar to the file name in the

right-hand column and press F4 to remove it. Once you are

finished, press F7 to start erasing the files you have selected.

With the menu bar in the right-hand column, you may edit or add

file names by typing them in directly rather than paging through

the directory list. Move the menu bar to the file name you wish

to edit, and press RETURN. You may then re-type the file name.

If you move the menu bar to the blank line below the last file

name and press RETURN, you may type in a new file name.

1.1.4 Advanced Configuration

This will offer you a set of options for the advanced user or for

setups with specific needs. This is explained in more detail

in the Advanced Configuration section. (Section 5)

1.1.5 Hang Up

The \* Key hangs up the current connection to a BBS. This

space has been reserved for future development, but for now,

it closes the connection.

1.1.6 Save / Load Configuration

You can either save the current configuration or load the

previously saved configuration. The configuration file is

savable from the Main Menu, the Advanced Configuration menu,

and the Dialer. All configuration options, IP Addresses for the

dialer, and F Key Signatures are saved in this one configuration

file.

REPORT THIS AD

1.1.7 Selecting a modem type

Striketerm uses a device-independent interface to

implement serial communications. A “serial device”, as described

here, is any set of hardware connected to the Commodore 64 that

provides an interface to a serial port. In general, the serial

port itself (usually a 9-pin or 25-pin male RS232 connector) is

also a part of this hardware in some way, so the serial device

is the bridge between the computer and the port.

For information on where to obtain these devices, see Appendix

A, Vendor information. Striketerm currently has drivers for the

following devices:

WinVice: This driver will allow you to use WinVice with tcpser

at speeds up to 9600 Baud. (Full instructions on setting this up

at 1200baud.wordpress.com)

SwiftLink: This driver uses the SwiftLink cartridge. The

SwiftLink cartridge plugs into the cartridge port, and can

operate at speeds up to 38,400 bps.

Turbo232: This driver uses the Turbo232 Cartidridge.

Comet: available from commodoreserver.com, The comet currently

works at 2400 baud in Striketerm. Make sure your comet is set to

2400 baud before dialing out on Striketerm.

HART: This driver uses the HART cartridge. The HART cartridge

plugs into the cartridge port, and can operate at speeds up to

57,600 bps.

User port: This driver should be used for any generic

RS232/TTL -> Serial port or USB Adapter (such as the Strikelink).

It can handle speeds up to 2400 baud.

UP9600: This driver, written by Daniel Dallman, uses a modified

RS232 interface to achieve 9600 bps through the user port. The EZ232

interface by Doppleganger works well with this option.

(See Vendor appendix)

Retrosurf: This RRNet/Silversurfer driver was written by Nicolas

Welte. It is incompatible with the fastloader enhancements.

Therefore these have to be disabled, but without killing the

cartridge hardware registers. The simplest and most straightforward

way is to boot the cartridge with the flashmode jumper set.

This behaviour will change with the next beta version of the firmware.

General incompatibilities have been reported on Striketerm,

which we will try to fix soon.

Since the RetroReplay hardware offers a REU compatible mode,

it would be a shame not to make use of this. Later versions of the

firmware will support this natively, but as of now, I’m supplying

a small BASIC program that will set this mode (reu-retrosurf).

Since the fastloader enhancements have to be disabled anyway, the program

will require you to have booted the computer with the flashmode jumper set.

Then it will ask you to remove that jumper and after that the REU mode

will be enabled, and Novaterm will be loaded. The RetroReplay fastloader

enhancements are not compatible with the SuperCPU, but you can still make

use of your SilverSurfer. Again, the computer has to be booted up with the

flashmode jumper in place, and if you like, you can run the REU mode

program to make use of your REU, as well.

1.1.7.1 Selecting a Turbo232/SwiftLink address

If you specified a SwiftLink or Turbo232 as your serial device,

you must specify its memory address. A SwiftLink is shipped with

an address of $DE00, but it may be moved by physically altering

the cartridge, or by using an AddressFixer device (see Appendix

C, Vendor information). A Turbo232 is shipped with an address of

$DF00, but it may be relocated in a similar manner. The

following addresses are available:

REPORT THIS AD

$DE00: This address should be selected for a standard SwiftLink,

or for a Turbo232 mapped to $DE00.

$DF00: This address should be selected for a standard Turbo232,

or for a SwiftLink mapped to $DF00.

$DE20: This address should be selected for a SwiftLink or

Turbo232 mapped to $DE20.

$DF20: This address should be selected for a SwiftLink or

Turbo232 mapped to $DF20.

$D700: This address should be selected for a SwiftLink or

Turbo232 mapped to $D700. This can only be used on a C128 in 64

mode.

1.1.8 Selecting Baud Rate

Depending on the type of Modem/Serial Driver you are using,

you have access to different baud rates. Using the B command

at the main menu will cycle through your available Baud rates

for the modem you’re using.

1.1.9 Selecting the Terminal

Terminal mode depends on the use of “terminal emulations” to

display incoming text. A terminal emulation is a special driver

(module) that interprets incoming text and displays it

appropriately.

For instance, one terminal emulation shipped with Striketerm is

called “ANSI”. ANSI is the name of a commonly used emulation

where certain sequences of characters are used to position the

cursor, change text color, clear the screen, and do other

display-related things. In ANSI, whenever the driver receives

the ESCape character, the characters immediately following will

always be some sort of special ANSI command. Striketerm’s ANSI

driver knows how to interpret these commands and take the

appropriate action.

So why are they called “emulations”? In the old days, almost all

computers were big mainframes, and they had one or more dumb

terminals attached to them. These terminals interpreted certain

escape sequences in the manner described above. Today, these

escape sequence command have been adopted by the on-line

community. When a desktop computer interprets sequences in this

way, it is said to “emulate” the old dumb terminals.

Typically, for each on-line service you call, you’ll need to

determine which terminal emulation to use. Striketerm includes a

variety of terminal emulations:

ANSI: This emulation displays ANSI color and graphics in

80-column mode. This should almost always be used when calling

IBM-based bulletin boards.

ANSI-40: This emulation displays ANSI color and graphics in

40-column mode. This type of display is limited, as ANSI

graphics are typically laid out for 80-column displays, and will

not look correct in 40 columns. ANSI-40 is intended for users

who want the benefits of ANSI colors (and some graphics) without

using 80-column mode.

VT102: This emulation supports the VT100 and VT102 emulations.

It is typically used when dialing into large computer systems,

such as an Internet provider or a public library’s on-line

catalog.

REPORT THIS AD

VT52: This emulation is typically used when calling large

computer systems that do not support VT100 or VT102. (It is an

older standard.)

Standard-80: The Standard and Commodore emulations are simple

text displays; they do not actually interpret any special escape

sequences. Every character received by the modem is displayed as

is (with ASCII translation, of course). This is appropriate for

any on-line service that does not support terminal emulation at

all.

Standard-80 is, of course, the 80-column version.

Standard-40: This is the 40-column version of Standard.

Commodore: This emulation is for using bulletin boards that can

display Commodore color/graphics. The only difference between

this and Standard-40 is that ASCII translation is turned off.

ANSI-wide, VT102-wide, Std-80-wide: These are all variations of

the above emulations that use the wide 80-column font.

1.1.10 Selecting the Protocol

This is the protocol used for file transfers

How do you choose which protocol to use? Some protocols are

better than others; you want to use the best one possible.

However, you are constrained by the fact that the computer on

the other end of the line must be using the same one. Therefore,

you must choose a protocol that the other computer can also use.

The protcols supported by Striketerm are listed below in order of

performance and convenience.

Zmodem is by far the most popular and superior file transfer

protocol used in the on-line world today. Zmodem supports

excellent error checking and batch file transfers (several files

at a time). File names are transferred from the sender to the

receiver, so the receiver doesn’t have to ask the user to

specify file names. Zmodem transfers in data blocks of up to

1024 bytes. Zmodem doesn’t suffer from the problem of “file

padding” like other protocols (as explained in the section on

Ymodem).

Zmodem also has a nice feature called auto-download, where

Striketerm automatically starts a download when it detects that

the other computer has initiated an upload. (auto download

can be turned on in the Advanced Configuration in the main menu)

Striketerm’s Zmodem also supports crash recovery, which allows you

to resume an interrupted file transfer. If a Zmodem transfer was

aborted early for any reason, you may continue the transfer

where it left off by simply attempting to download the file

again. If Striketerm detects that the file already on the disk –

and if the Auto-replace downloads option is turned off (see

4.4.6) – it notifies the sender to continue where the file left

off. (The only time a crash recovery isn’t possible is when the

file was downloaded to the buffer, and other buffer files have

been created since the download. The aborted file must be the

last file listed in the buffer directory for a crash recovery to

be possible.)

Striketerm’s Zmodem will operate in “streaming” mode if you are

downloading a file directly to the buffer. Zmodem defaults to

streaming mode under the following conditions:

\* The Allow streaming mode option in the Configuration menu is

set to ON.

\* The download device is set to the buffer.

\* The baud rate is 2400 bps or less, or:

\* The baud rate is 4800 bps or higher and hardware flow

control is turned on.

REPORT THIS AD

Zmodem’s streaming mode requires hardware flow control at higher

speeds in order to keep up with the steady flow of incoming

data. If hardware flow control is not turned on and the baud

rate is at least 4800 bps, or if you are downloading to a disk

device, Zmodem defaults to its regular “block” mode.

If you set the Allow streaming mode option in the Configuration

menu to OFF, Zmodem will always default to “block” mode. You

should set this if you have trouble downloading in streaming

mode from a bulletin board or on-line service.

Ymodem is also a protocol that supports batch file transfers. It

is an extension of the original Xmodem protocol (see below). It

uses 1024-byte blocks to transfer data. It is slightly faster

than Zmodem, but it has one disadvantage that makes Zmodem the

preferred protocol.

The main inconvenience of Ymodem is that each block of data

transferred must be of a certain length. If a file’s size is not

an exact multiple of this fixed block length, the end of the

file will be “padded” with filler characters that were used to

achieve the required block length in the last block. Striketerm

can remove these fill characters automatically if you have the

option turned on (see 4.7.9, Chop X/Ymodem padding), but there

is risk of damaging the file when doing this, as explained in

the referenced section.

Ymodem-g is an extension of Ymodem batch that operates in

streaming mode. You may use this protocol only to upload or

download directly to the buffer (because the Commodore 64 cannot

simlutaneously read characters from the modem and write them to

a physical disk drive). There is no provision for error recovery

in Ymodem-g; the first time an error is detected, the transfer

is aborted. Therefore, it is only suitable for downloading over

modem connections with error correction, or across a null modem

connection with another computer.

Ymodem-g holds the prize for being the fastest of the protocols,

since there is no overhead in waiting for acknowledgments from

the receiving computer. However, the lack of error recovery

makes it impractical except over a modem link with error

correction.

Xmodem-1k: This is an extension of the original Xmodem protocol

that uses 1024-byte blocks. It can only transfer a single file

at a time. Unlike Zmodem and Ymodem, both the receiver and sender

must supply a file name, as the protocol will not transfer the name

with the data. This protocol also has the “file padding” problem

described above.

Xmodem-1k-g: A streaming-mode extension of Xmodem-1k, just as

Ymodem-g is the streaming-mode extension of Ymodem. Like

Ymodem-g, you must upload or download directly to the buffer,

and the transfer aborts at the first sign of an error. It is

extremely fast, but must be used over a link with error

correction.

Xmodem-CRC: This is the original Xmodem protocol. It supports both CRC

checksums and additive checksums, so you may use this protocol

even if the receiver does not support a CRC version of Xmodem.

It transfers data in 128-byte blocks; file names are not

transferred; and “file padding” does occur.

Punter (C1): The Punter protocol (also called “C1”) was originally

designed for use on Commodore computers, although it has fallen out of

use in recent years. The Punter protocol can only transfer a

single file at a time, but it does not suffer from “file

padding”. It uses up to 255-byte blocks to transfer data. You’re

likely to find it only on Commodore-based bulletin board

systems. Many systems do not work with Punter at speeds slower

than 9600 baud or with emulators, so you may be best using Xmodem

for C64 transfers if Punter causes your downloads to work

incorrectly.

REPORT THIS AD

Multi-Punter: Punter multi-transfer is an extension to Punter that supports

batch file transfers. It is available on a few Commodore-based

bulletin board systems. Note: This is not the same protocol as Punter’s

C2 batch protocol.

Kermit: The Kermit protocol is designed to work over communication links

where other protocols aren’t able to function for whatever

reason. If control characters are intercepted or changed by a

network switcher or other device before the receiving software

sees them, file transfers with protocols like Xmodem and Ymodem

will not work. Kermit seeks to overcome this problem by encoding

all data characters as text. Because of this, and because it

allows up to only 94 bytes per block, Kermit is very slow. It

should only be used in situations where it is the only protocol

that will work. Kermit does support batch file transfers.

Wxmodem: This is a version of Xmodem designed for use over networks where

the delay between sender and receiver is significant. It is

designed to send up to 512 bytes of data at a time. It is

provided in Striketerm for those situations where it is the only

protocol available to the other

1.1.11 Selecting a block size

Some protocols allow you to vary the size of data blocks that

are transmitted. Basically, a data block is a segment of the

file being transferred. The data block is sent by the sending

computer along with a checksum of the block. When the sender has

transmitted a block, it then waits for the receiver to

acknowledge it as either good or bad before continuing.

The size of that data block affects performance. With larger

sizes, fewer acknowledgements must be sent by the receiver, so

the file transfer is faster. However, if the communication link

is noisy at all, many blocks may need to be retransmitted;

therefore, a smaller block size will alleviate the overhead of

having to retransmit large data blocks that were corrupted.

Only some of the protocols allow you to change the block size:

Zmodem allows block sizes between 32 and 1024 bytes.

Punter allows block sizes between 40 and 255 bytes.

Kermit allows block sizes between 20 and 94 bytes.

To change the block size, use the Block size option on the

Main Menu.

1.1.12 Selecting Disk Devices:

By selecting U on the main menu, you can change your Upload/Download

Drive, and by Selecting F you can select the Buffer Drive.

To set the disk device number for any of these options, enter

the device number, followed by a comma, followed by the

drive/partition number. The drive/partition number (distinct

from the device number) is used when you have a single disk

device containing two or more drives, or a partitioned hard

drive or RAMLink/RAMDrive. If you are using only single disk

drives, the drive number should always be 0.

Striketerm accepts values between 8 and 30 for disk device

numbers. (Striketerm also accepts a disk device of 1, which is

used by some enhancement hardware as a special device.) When you

enter a device number for any of the above options, Striketerm

checks to see if that device is actually connected. If it is

not, the device number is not changed.

REPORT THIS AD

For the Program device, Upload device, and Download device,

typing a letter B in place of the device number directs these

functions to Striketerm’s buffer. In this case, the

drive/partition value specifies the buffer subdirectory number

to use.

1.1.13 Free Bytes

Free Bytes shows how many bytes are available to use in the

current memory device. Striketerm defaults to internal memory.

Changing the memory device can be achieved in the Buffer Menu.

More information on memory devices in the Buffer Section of

this documentation.

1.2.1 How Modules Work

All modules perform the same way. Listed below is a summary

of all of the different types of modules used by Striketerm. If

you don’t know what some of the descriptions mean, you’ll learn

about them in other sections of the manual.

Prefix:

Description:

Manual section:

80col 80-column drivers

asc ASCII translation tables

font 40-column fonts

font80 80-column fonts

modem Modem types

nova Utility programs

prt File transfer protocols

ram Memory expansion drivers

serial RS232 serial drivers

term Terminal emulations

time Time setting modules

tmod Scripts to load emulations

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. TERMINAL MODE

This section takes you through the functions most commonly used

while on-line. It starts with a look at terminal mode, the

center of your on-line activities. It then explains how to set

up the dialing directory, how to use the buffer to capture text,

and how to transfer files (upload and download).

2.1 Exploring terminal mode

Terminal mode is where you interact with a computer on the other

end of your modem. The entire screen is dedicated to displaying

text. In terminal mode, anything you type on the keyboard is

sent to your modem, and the data received by your modem from the

other computer is displayed as it arrives.

In terminal mode, Striketerm can display text in 40- or 80-column

mode. 40-column mode is the normal state for a Commodore 64.

Striketerm also provides an 80-column mode, where each character

is half the width of a 40-column character. See section 3.1.6,

Notes on 80-column mode for more information about it.

This section covers various features of terminal mode, including

the status line, key commands, key reassignments, and terminal

emulations.

2.1.2 Status line

REPORT THIS AD

In terminal mode, Striketerm displays a status line which reports

various conditions. In 40-column mode, the status line looks

like this:

:E:T:K:C:B:X: 2400 00:00:00

In 80-column mode, the status line has some extra information:

:E:T:K:C:B:X: 2400 ANSI Zmodem 3:30pm 00:00

The letters separated by colons represent the status of certain

terminal settings. A letter appears highlighted (reversed) if

its setting is “on” and normal if its setting is “off”. The

function of each letter is described in more detail in the

referenced sections:

E: Local echo. This determines whether keys you type are

displayed on your screen as well as sent to the modem. (see

3.1.3, Commodore key commands, C= E)

T: ASCII translation. This indicates whether ASCII translation

is in effect.

K: Control character display. If this is on, control characters

are printed explicitly. For instance, a CTRL-A would be

displayed with the two characters ^A.

C: Carrier indicator. This indicates whether a modem connection

is established.

B: Buffer capture. This indicates whether data is being captured

to the buffer. (see 3.1.3, Commodore key commands, C= B, C= O)

X: Script file active. This indicates whether a script file is

currently running. (see section 6, Scripts)

The number directly to the right of the status letters displays

the current baud rate.

The clock in the right-hand corner in both 40- and 80-column

modes is an on-line timer. It is reset to zero and begins

counting immediately after a modem connection is established.

When the modem is disconnected, the timer stops and retains its

last value until a connection is made again. The timer is useful

when you are calling long distance or are using an on-line

service that charges by the minute or hour. In 40-column mode,

the clock counts seconds; in 80-column mode, the clock updates

every minute.

In 80-column mode, the extra width available on the status line

is used to display the current terminal emulation (see 3.1.1,

Terminal emulations), file transfer protocol (see 3.4.3,

Selecting a protocol), and the current time (see 1.1.10,

Selecting a real-time clock device).

The status line may be turned off by pressing C= S in terminal

mode.

2.1.3 Terminal Mode Commands

While using terminal mode, a variety of functions are available

to you by pressing logo key combinations. The terminal mode key

commands are summarized below. A key described as C= X means you

should hold down the Commodore logo key while pressing the

letter X.

F3 – Instead of taking you to terminal mode (since you’re now

already in it) F3 displays the terminal menu while in terminal

mode.

REPORT THIS AD

C= A – Toggles ASCII translation. This command determines

whether incoming data should be translated from ASCII to PETSCII

for proper display on the C64. This should always be on unless

you are connected to a Commodore color/graphics BBS. This

command works only with the “Standard” terminal emulations , as

the others require translation to remain on to function properly

(see 3.1.1, Terminal emulations).

C= B – Pause/resume buffer capture. This is only valid when a

buffer capture file has been opened using C= O. Pressing C= B

stops text capture while leaving the capture file open. The

highlighted B in the status line turns off, indicating that text

is no longer being captured. Pressing C= B once more resumes

text capture and highlights the B again.

C= C – Toggle between 40 and 80 columns.

This command works only with the Standard

terminal emulations, as the others require the 80-column display

to function

C= D – Download a file.

C= E – Toggle local echo. When local echo is on, everything you

type is sent to the screen as well as to the modem. This setting

is indicated by the letter E in the status line. Local echo

should be turned on if you are connected to a computer or

bulletin board which does not echo back characters. Most

bulletin boards echo characters, but if you are connected

directly to another terminal (if, say, you and a friend were

linked up to exchange files) and you can’t see what you are

typing, you should turn local echo on.

C= F – Display the amount of free disk space.

C= H – Hang up phone. This hangs up the modem.

C= I – Sends your ID for the current BBS you’re calling.

(based on the dialer settings)

C= N – Display the amount of free buffer space in bytes and

equivalent disk blocks.

C= O – Open/close buffer capture file. Pressing C= O prompts you

for a file name, and then opens a file in the buffer for

receiving text. Any text that is subsequently displayed on the

screen is stored in the buffer file as well. Pressing C= O again

closes the buffer file.

C= P – Sends the password for the current BBS you’re on

(based on the dialer settings)

C= Q – Enter chat mode. This option works in the Standard

terminal emulations only. In a

Standard terminal emulation, this command provides a

159-character keyboard buffer (at the top of the screen in 40

columns; at the bottom in 80 columns) to use in conference chat

modes.

C= S – Toggle status line. In 40 columns, turning the status

line off frees up an extra line at the top of the screen for

text display.

C= U – Upload a file.

C= V – View disk directory.

C= 0 – Enable/disable Commodore keys. When you press C= 0

(zero), a caret symbol appears in the status line. This

indicates that the key commands are disabled and become normal

graphics characters again. This allows you to type graphics

characters, say, for creating a graphics screen in a message on

a BBS. Pressing C= 0 again removes the caret symbol and changes

the keys back to commands.

REPORT THIS AD

2.1.4 Signatures

The F2, F4, F6, and F8 keys on the Commodore 64 keyboard can be

redefined to send strings to the modem when you press them in

terminal mode. These are useful for defining strings that you

find yourself typing often (such as your user name when logging

into an on-line service).

The function keys are defined from the Edit Signatures menu.

2.1.5 Key assignments

There are a few characters that do not exist on the C64

keyboard. These are characters that are part of the standard

character set used by the rest of the world. To accomodate these

characters, some of the graphics keys have been redefined.

To get:

Character:

Type:

escape key left arrow

tab key RUN/STOP or CTRL-I

backspace (8) INST/DEL

true delete (127) CLR/HOME

backslash \ British pound

underline \_ shift-@ (at)

left brace { shift-+ (plus)

right brace } shift-minus

vertical bar/pipe | shift-British pound

back apostrophe (grave) ` shift-\* (asterisk)

tilde ~ shift- (caret/up arrow)

In addition to the above characters, the entire lowercase

European character set is available. In order to use these

characters, you must be using either ANSI or VT102 emulation,

have local echo turned off, and have the Commodore command keys

disabled by pressing C= 0 (zero) (see 3.1.3, Commodore key

commands, C= 0).

Key char Key char Key char Key char

C= Q è C= A à C= X Ð C= \* Ü

C= W é C= S á C= C ç C= 1 ¡

C= E ê C= D â C= V Æ C= 2 ¿

C= R ë C= F ä C= B ß C= 3 ø (Ö)

C= T ý C= G å C= N ñ C= 4 ò

C= Y ÿ C= H ã (Ä) C= M ü C= 5 ó

C= U ì C= J ù C= + æ C= 6 ô

C= I í C= K ú C= – Ä C= 7 ö

C= O î C= L û C= £ £ C= 8 õ

C= P ï C= Z ð C= @ Ö

2.1.6 Notes on 80-column mode

If you run Striketerm on a C128 in 64 mode, and if you have an

80-column RGB monitor, you can use the C128’s 80-column screen

for terminal mode. When you switch to terminal mode with one of

the C128 VDC drivers in place, the 40-column screen goes blank.

At this point, you must switch your monitor to 80-column RGB

mode. When you leave terminal mode, the main menu reappears on

the 40-column screen, so you must switch your monitor back to

40-column mode.

The C128 VDC screen is a true 80-column display with easily

readable text and a fast display. The notes below do not apply

to this mode.

REPORT THIS AD

On a C64, Striketerm simulates an 80-column display by drawing

half-width text characters on the C64’s high-resolution bitmap

screen. Because the C64 was not designed for 80 columns, there

are a few limitations to be aware of. Striketerm compensates for

most of these limitations if you choose the proper settings.

2.1.6.1 Improving readability

First of all, the text can be just plain hard to read. Each

character is only four pixels wide, and vertical lines are one

pixel in width. Your monitor must be in decent shape to read it.

If your monitor is blurry at all — or if you are trying to use

a TV set as a monitor — you’ll have a difficult time. Make sure

your monitor is adjusted properly (by fiddling with the control

knobs).

If you choose to replace your monitor, Commodore-brand color

monitors do an excellent job of displaying 80-column mode.

Monochrome monitors — like the old “green screens” — are even

sharper, but you will lose the color.

You may also try changing the text and background color in

80-column mode (see 4.2.4, 80-column color). The default is

light-gray on black. Your eyes may have an easier time with a

different combination. A black or dark-gray background is needed

for proper color reproduction with ANSI terminal emulation, but

if you primarily use 80-column mode for VT102, VT52, or standard

emulation, a black-on-yellow or black-on-white display may be

preferable. Some people prefer light-green text on a medium-gray

background. Another good combination is black text on medium- or

light-gray background.

Turning up the brightness knob on your monitor helps with any

color combination. It lessens the “color bleed” problem of

single-pixel characters, where dark colors tend to bleed over

into light ones.

One more thing you can do — and you might think this is really

boneheaded, but it works — is keep your screen clear of dust.

You’ll be surprised what a difference a clean monitor makes.

2.1.6.2 Compensating for slow display speed

Because Striketerm must draw each character in the bitmap, the

printing speed of 80-column mode is noticeably slower than

40-column mode. Striketerm’s 80-column mode can keep up with

incoming data at 4800 bps; at 9600 bps, it falls behind and

starts to lose text unless you use one of the options described

below.

If you are using Striketerm at speeds in excess of 4800 bps,

chances are you are using a high-speed modem (9600, 14.4k, or

28.8kbps). All high-speed modems support a feature called

“hardware flow control”. This feature is explained in more

detail in section 4.6.2, Flow control. Using hardware flow

control allows the 80-column display to keep up with incoming

text at all times. You should have hardware flow control turned

on in Striketerm (and in your modem). With hardware flow control,

you can set Striketerm’s baud rate as high as 38,400 bps without

losing any data.

Another option is available if you use a RAM Expansion Unit

(1700/1750/1764). An 80-column driver is provided that

implements a fast scroll using the REU’s direct memory access

(DMA) capability. The first-time configuration program

automatically loads this for you if you specified REU as your

choice of memory expansion. See section 4.2.1, 80-column driver

for information about using the fast-scrolling 80-column driver.

REPORT THIS AD

2.1.6.3 Displaying color

One limitation of the Commodore 64 is that it cannot display

more than two colors in a given 8×8 pixel area. (It can in

multicolor mode, but this mode cuts the horizontal resolution in

half, so it can’t be used for 80 columns.) Therefore, every two

adjacent characters in 80-column mode must be the same color.

This can cause some characters to appear in a different color

than intended by the on-line service. This explains why some

ANSI color/graphics displays will appear distorted when

displayed in Striketerm’s 80-column mode.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. AUTODIALER

Striketerm provides a phone directory for keeping track of on-line

services you call frequently. In the phone directory, you may

create an entry for each number you call. You have the

opportunity to define several configuration options for each

entry. When you dial an entry and make a connection, the

configuration options you specified are automatically set for

you.

You may enter a maximum of 26 entries.

3.1 Creating / Editing a dialing entry

There are 26 entries in the phonebook, called cells. Each cell

is defined with a letter of the english alphabet. To edit a cell,

click it’s letter, followed by the command “3” to

Edit Current Cell.

Each item has the following meaning:

Cell: The Name of the BBS

IP: The IP or WWW address of the system

Port: The system’s port number

Terminal: Selects the terminal the system uses

ID: Your ID for this BBS

Password: Your password for this BBS

3.2 Other commands

Command 1 hangs up the current connection, 2 dials an unlisted

number, and lets you specify the IP or WWW address and port

number, and 4 saves the phone book / configuration file. As per

usual, F1, F3, and F7 hold their same functions as in other

parts of the program.

3.3 Dialing a number

To dial one of your entries, type the letter of the entry

and press RETURN. This starts the dialing process.

Before dialing the number, Striketerm first loads the terminal

emulation for the entry.

Striketerm then starts to dial the number. The screen displays the

number being dialed and will send you to the terminal mode when

connected. The ID and password you specified in the cell configuration

will be assigned to the C= I and C= P commands in the terminal.

REPORT THIS AD

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. THE BUFFER MENU

4.1 Using the buffer

The “buffer” is a section of memory used by Striketerm for the

purpose of storing incoming text. When set up to do so, Striketerm

“captures” text in the buffer at the same time it displays the

text on the screen in terminal mode. The text is stored in a

“capture file”.

Striketerm’s buffer is set up as if it were a

disk drive. It has a directory and keeps track of distinct

files. The only difference is that its data is kept in memory.

(This kind of storage is called a “RAMdisk”.)

Since the buffer isn’t really a disk drive, the Buffer Menu

provides disk-like operations. Most of the standard operations

that work with a disk are available, such as scratching,

renaming, and copying files.

This section covers the operation of the buffer and explains the

various options available on the Buffer Menu.

4.1.1 Specifying the memory device

On the Buffer Menu, selecting K brings up a list of memory device

drivers. You should use the one corresponding to the hardware

you have connected. See Specifying the memory device for a

description of each driver.

Striketerm has a device-independent interface for accessing

memory, which means that Striketerm can use a variety of memory

expansion devices for the buffer with the appropriate driver.

Each driver uses all of the available memory in the device

(except for the RAMLink/RAMDrive driver; see below). Each driver

also has a maximum directory size (number of files) that may be

stored in the buffer directory.

For information on where to obtain these devices, see Appendix

A, Vendor information.

internal: Uses the internal memory of the Commodore 64. With

this driver, your buffer is limited to 4.5K of internal memory

(the RAM located under the kernal ROM). The maximum size of the

directory is 22 files.

VDC+internal: Uses the internal memory of the Commodore 64 as

well as the Commodore 128’s VDC memory. This driver only works

if you are using a Commodore 128 in 64 mode. It uses the 16K or

64K of RAM assoicated with the VDC chip (the driver detects how

much is available), as well as the internal memory of the

Commodore 64, for a total of 20.5K or 68.5K of space. The

maximum size of the directory is 64 files.

If you chose a C128 80-column screen previously (see 1.1.5),

this option will instead configure the VDC-alt+int driver, which

avoids using the first 16K of VDC memory (leaving it for the

80-column screen). If you only have 16K of VDC memory installed

in your computer, this driver only uses the 4.5K of internal

memory. If you have 64K of VDC memory, the driver uses the upper

48K of that memory, for a total of 52.5K of buffer space.

REU: Uses a RAM Expansion Unit (1700/1750/1764). The driver

detects the size of the REU. (The driver will detect REU’s

expanded to 2 MB, and even more if it’s ever done.) The maximum

size of the directory is 144 files. The driver also allocates

three pages (768 bytes) of memory for use with the fast-REU

80-column driver (see 4.2.1, 80-column driver).

REPORT THIS AD

BBG-GEORam: Uses a BBGRam or GEORam. The driver detects the size

of the unit (BBGRam cartridges are available in sizes up to 2

MB). The maximum size of the directory is 144 files.

RAMDrive: Uses a RAMDrive. This driver searches for a direct

access (DACC) partition in your RAMDrive with the name

“Striketerm”. The partition can be any number and any size; the

driver detects its size and allocates the memory in the

partition for the buffer. The maximum size of the directory is

144 files.

RAMLink: Uses a RAMLink. This driver searches for a direct

access (DACC) partition in your RAMLink with the name

“Striketerm”. The partition can be any number and any size; the

driver detects its size and allocates the memory in the

partition for the buffer. The maximum size of the directory is

144 files.

4.2 The buffer directory

Select Buffer directory from the Buffer Menu to list the buffer

directory. The directory is listed just like a disk directory,

with block sizes that indicate how many blocks the file will use

when copied to a physical disk. The directory also lists the

exact byte size of each file, and the number of bytes and

equivalent blocks free.

Files in the buffer directory can have type prg, seq, or scr.

4.3 Working with buffer subdirectories

Because the buffer can potentially have a large number of files,

Striketerm uses a simple subdirectory scheme for organizing files.

Each file can reside in a particular subdirectory, numbered 0

through 30. You use the Change subdirectory option to specify

the current subdirectory. The Buffer directory option will list

only those files in the current subdirectory.

For instance, if you have a number of script files loaded in the

buffer, they could clutter up the directory, and a capture file

you created would show up along with the scripts. Therefore, by

default, Striketerm loads and executes scripts from subdirectory

#1, while buffer captures use the current subdirectory (which is

set to #0 when you start up Striketerm).

Furthermore, by changing the Device Settings for the upload and

download device, you can direct file transfers to use yet a

different buffer subdirectory.

4.4 Buffer directory

This option lists the buffer directory.

4.5 Disk directory

This option lists the directory of the current disk device.

4.6 Load files from disk

This option brings up a list of files on the current disk

device. Select one or more files to be copied into the buffer

directory.

4.7 Save files to disk

This option brings up the list of files in the buffer. Select

one or more files to be copied to the current disk device.

REPORT THIS AD

4.8 Rename file

This option lets you rename a file in the buffer directory.

4.9 Delete Files

This option deletes a file from the buffer directory.

4.10 Clear subdirectory

This option deletes all files from the current buffer

subdirectory.

4.11 Reformat Buffer

This option performs a virtual formatting of the Buffer drive.

4.2 Buffer file processing options

The Buffer Menu also contains a series of options for special

processing of files in the buffer.

4.2.1 View file

This option displays the contents of a buffer file or

other sequential file.

4.2.2 View in 80 columns

This option displays the contents of a file using

80-column mode.

4.2.3 File to modem

This option sends the contents of a file to the modem,

line by line. After each line is sent, Striketerm pauses for the

number of seconds specified in the Line pacing option on the

Advanced Configuration menu

4.2.4 ASCII to PET

This option translates a file from standard ASCII to

Commodore PETSCII. If you downloaded a file and forgot to use

the “Transfer translation” option when you should have, this

command will convert the file for you (see 3.4.8, Transferring

textfiles from non-Commodore systems).

4.2.5 PET to ASCII

This option translates a file from Commodore PETSCII to

standard ASCII, placing carriage return/linefeed combinations at

the end of each line. (This format is appropriate for MS-DOS

text files.)

4.2.6 PET to Unix

This option translates a file from Commodore PETSCII to

standard ASCII, placing only linefeeds at the end of each line.

(This format is appropriate for Unix text files.)

4.2.7 UUdecode file

A “uuencoded” file is a file that has been converted from its

regular form to an encoded form that contains only printable

ASCII characters. This allows the file to be exchanged through a

medium that only allows printable ASCII characters, such as a

public message base or an e-mail message. An end user can

“uudecode” the file to obtain the original file. It is most

commonly used on the Internet to include files in Usenet or

e-mail messages.

REPORT THIS AD

This option will correctly decode uuencoded files that were

either downloaded — with the encoding still in ASCII — or

captured — with the encoding converted to PETSCII.

When the original file name is detected, you are asked to

specify its file type. If you select seq, you are asked whether

to ASCII-translate the original file during the decoding

process. Answer yes if you know that the original file is a

standard ASCII text file. If you’re not sure, just answer no;

you can always go back and translate the original file with the

ASCII to PET option (see 3.3.7.5, ASCII to PET).

4.2.8 UUencode file

This option uuencodes an original file. The encoding is stored

in standard ASCII. This option is appropriate if you intend to

upload the file directly.

4.2.9 UUencode in PET

This option uuencodes an original file, converting the encoding

to Commodore PETSCII. This option is appropriate if you intend

to put the file into a message editor using the File to modem

option (see 3.3.7.4, File to modem).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. ADVANCED CONFIGURATION

The Advanced Configuration menu allows you to modify various

settings for Striketerm if the defaults dont work correctly.

5.1 Word/parity

Most on-line services operate at 8 bit word length and no

parity. Therefore, you should set this option to read 8N1. Some

on-line services may require you to use “even” or “odd” parity

instead, in which case this setting should be 7E1 or 7O1.

However, you should specify the Word/parity setting in the dial

entry for each on-line service in your dialing directory, since

the setting can be different from one service to the next. It’s

best to leave the setting on the Configuration menu set to 8N1,

and just let it change depending on the service you dial.

5.2 40/80-column settings

5.2.1 80-column driver

This option specifies the driver to use for 80-column mode. In

other words, this option determines which set of programming

code to use to display the 80-column screen. The drivers

currently available are:

default: The standard, unassisted 80-column driver that uses the

C64’s high-resolution bitmapped screen.

fast-REU: Uses a RAM Expansion Unit (1700/1750/1764) to

implement a fast scroll on the C64’s hi-res screen. You must use

the REU memory driver in conjunction with this driver (see

1.1.7, Specifying the memory device).

VDC (25): Uses the C128’s 80-column screen in 25-line mode. Use

this driver only if you are running Striketerm on a C128 in 64

mode, and you have an RGB 80-column monitor. This driver also

uses the C128’s 2 MHz fast mode while you are in terminal mode.

VDC (28): Uses the C128’s 80-column screen in 28-line mode. Use

this driver only if you are running Striketerm on a C128 in 64

mode, and you have an RGB 80-column monitor. You may need to

adjust your monitor to make all 28 lines visible. . This driver

also uses the C128’s 2 MHz fast mode while you are in terminal

mode.

REPORT THIS AD

Selecting a C128 VDC screen driver (C128 in 64 mode only)

If you are running a C128 in 64 mode, and if you have an

80-column RGB monitor, you have the option of using the C128’s

80-column screen. You can choose either a 25-line display or a

28-line display. If you choose 28-line mode, you may need to

enter terminal mode later and adjust your monitor to bring all

the lines onto the screen.

If you can’t use the C128’s 80-column mode, just select “Give me

the regular screen”.

If you are running a C64, this option will not be presented to

you.

5.2.2 40-column font

This option specifies the font to use in 40-column mode.

5.3 Carrier type

This option applies only when using the User port serial driver.

It specifies whether the carrier detect line from the modem

indicates a carrier when high (Normal) or when low (Inverted).

(The “carrier detect” indicates whether there is currently a

connection established with another modem.) This must be set

properly, or Striketerm features that depend on a proper carrier

detect will not function, such as redialing and the on-line

timer.

To set this, make sure your modem is hooked up and turned on,

but is not on-line. Go to terminal mode and check the state of

the indicator letter C in the status line. If the C is reversed

(highlighted) even though there is no connection, change the

carrier type to the opposite of its current setting.

5.4 Chop X/Ymodem padding

With this option turned on, Striketerm automatically removes any

“padding” added by the Xmodem and Ymodem family of file transfer

protocols. (See 3.4.3.2, Ymodem batch for a discussion on file

padding.)

5.5 Zmodem auto-download

This option controls whether the Zmodem auto-download feature is

to be used. If you do not have the Zmodem protocol installed,

this option has no effect (see 3.4.3.1, Zmodem).

5.6 Allow Zmodem streaming

With this option turned on, Zmodem will operate in “streaming”

mode if the following conditions are met:

\* The download device is set to the buffer.

\* The baud rate is 2400 bps or less, or:

\* The baud rate is 4800 bps or higher and hardware flow

control is turned on.

Turning off this option disables streaming mode in all cases.

You should do this if you have trouble downloading from a

particular service in streaming mode. (See section 3.4.3.1,

Zmodem for more information about Zmodem’s streaming mode.)

5.7 Existing files policy

This setting determines what happens during a batch download

when a file to be downloaded already exists on disk.

REPORT THIS AD

\* Replace: Deletes the file on disk and creates a new one with

the same name.

\* Rename: Renames the incoming file by incrementing the last

character of the file name.

\* Resume: If the Zmodem protocol is selected, Zmodem attempts

to resume transfer of the file. (See 3.4.3.1, Zmodem) If a

different protocol is selected, the incoming file name is

renamed.

5.8 Flow control

Flow control allows Striketerm to keep up with incoming data. When

Striketerm falls behind in processing incoming data, Striketerm

tells the remote computer to pause until Striketerm catches up.

When Striketerm is nearly caught up, it tells the remote computer

to resume transmission. This prevents characters from being

lost. Flow control is done automatically, but it may be turned

off by toggling this option.

Striketerm supports two types of flow control:

Software (XOFF/XON): In software flow control, Striketerm sends an

XOFF character (usually CTRL-S) to pause and an XON character

(usually CTRL-Q) to resume.

Hardware (RTS/CTS): In hardware flow control, Striketerm tells

your modem to stop delivering data to the computer by lowering

the RTS line (one of the RS232 port pins). In this state, the

modem stores incoming data in a buffer until Striketerm raises the

RTS line again. In addition, the modem can tell Striketerm to stop

giving it data by lowering the CTS line (which can happen if the

remote computer has lowered its RTS line). Hardware flow control

is much more precise and reliable than the software method,

since modems respond immediately to changes in RS232 lines.

However, to use hardware flow control, you must have at least a

9600 bps modem. Most slower modems (2400 bps and below) do not

respond to the RTS and CTS lines, so hardware flow control is

not possible with them.

If you are using a high speed modem (9600, 14.4k, or 28.8k) you

must use hardware flow control to take advantage of data

compression and error correction, and to enable Striketerm to keep

up with incoming data (see 4.1.2,Baud rate).

At speeds of 2400 bps and below, flow control is not necessary,

and it can be set to None.

5.9 Flow tolerance

The “flow tolerance” specifies the number of characters that may

build up in the receive buffer before Striketerm stops the flow

(using either hardware or software flow control). Since the

receive buffer holds a maximum of 256 characters, the tolerance

may be any number between 6 and 256. The default setting is 220,

which is effective for speeds up to 38,400 bps using hardware

flow control. (If you are using a HART cartridge at 57,600 bps,

you may need to reduce this setting to avoid losing data.)

If you are using software flow control, this value should be

much lower, since there is often a significant delay between

when Striketerm sends the XOFF signal and when the remote computer

actually stops sending data.

5.10 XOFF character

With software flow control, this is the ASCII value of the

character used to stop the flow of incoming data. The default is

19, or CTRL-S, which is the standard key used for pausing the

flow of data. Some bulletin boards use a different key for

pausing, so you would need to change this in that situation, if

you were using software flow control.

REPORT THIS AD

5.11 XON character

With software flow control, this is the ASCII value of the

character used to resume the flow of data. The default is 17, or

CTRL-Q, which is the standard key used for this purpose. Some

bulletin boards use a different key for this, so you would need

to change this option in that situation, if you were using

software flow control.

5.12 Set Time From

Striketerm provides several methods of setting the Commodore 64’s

internal clock from an outside source. Select from one of the

following choices:

CMD-RTC: If you have a hard drive or floppy drive from Creative

Micro Designs, Inc., or a RAMLink with a real-time clock

installed, choose this method. This driver scans devices 8

through 30 for a CMD drive with a clock using the T-RB disk

command. Striketerm reads the time from the first drive that does

not report an error from this command.

BBRTC-port1: If you have a BBRTC device from Performance

Peripherals, Inc. plugged into joystick port 1, choose this

method.

BBRTC-port2: If you have a BBRTC device from Performance

Peripherals, Inc. plugged into joystick port 2, choose this

method.

Manual: If you do not have any of the above real-time clocks,

choose Manual. This method asks you to enter the current time in

24-hour format.

All of the above drivers read and set the real-time clock when

you start up Striketerm, except for the Manual option. If you have

selected the Manual driver, you will not be prompted for the

current time when you start up Striketerm; instead, you must

select the driver again from Striketerm’s Configuration menu. At

that time, Striketerm prompts you for the current time.

5.13 Save buffer when full

If you do not have a memory expansion device, you are limited in

the amount of buffer space available. With a stock Commodore 64,

you have only 5.5K of memory for the buffer.

To aid in capturing a large amount of text with a limited buffer

space, Striketerm has the ability to periodically save the buffer

to disk and clear it out when it fills up. This option turns the

feature on or off.

When the buffer becomes full during a text capture, Striketerm

immediately tries to pause the flow of data. How it does this

depends on the type of flow control being used (see 4.6.2, Flow

control). Once the host computer stops sending data, Striketerm

saves the capture file to disk, appending it to the disk file if

it already exists. The capture file in the buffer is erased and

re-opened, making all the memory available again. Then, Striketerm

requests to resume the flow of data (again, depending on the

type of flow control) sends a character to tell the host to

resume transmission (the XON character). The buffer capture

continues after this, and the process repeats when the buffer

fills up again. When the text you want to capture has all been

received, you must manually save the remaining text. Striketerm

allows you to append this text to the existing disk file. The

result is the equivalent of one large capture file saved to

disk.

If a disk error occurs while trying to save the capture file,

Striketerm does not resume transmission. This leaves the remote

system paused so that you may correct the problem, save the

capture file manually, and resume the text capture where it left

off.

If you are using a memory expansion device, and thus any memory

device driver other than internal, you should turn off this

option, since you have a great deal of space available for

capturing text.

5.14 Notes on changing hardware and drivers

If you have more than one serial device and/or memory expansion

device, you may need to take some extra steps when physically

changing hardware and changing Striketerm’s drivers.

If you attempt to load a driver for one device that uses the

same registers as another device, Striketerm may lock up. For

instance, the HART cartridge and the BBGRam cannot coexist

because they share some common I/O registers. Suppose you had

HART cartridge plugged in but not the BBGRam, and you had the

HART serial driver and internal memory driver installed. Then,

you decide to unplug the HART and put the BBGRam in. You do so,

then load up Striketerm again. It locks up! This is because

Striketerm is still loading the HART driver as specified in the

configuration file. When it loads the HART driver, the function

that tests for the presence of the cartridge interferes with the

BBGRam, and the system locks up (or does something else

unpredictable).

To avoid these conflicts, you should:

1. Unplug everything.

2. Load up Striketerm.

3. Change drivers and save the configuration. (Striketerm will not

lock up if a device it is looking for simply isn’t present.)

4. Plug in the new hardware.

5. Start Striketerm again.

As you get used to switching things around, you’ll eventually

determine which erroneous drivers won’t lock up with conflicting

hardware installed, so you won’t have to load up with nothing

installed every time you want to change hardware around.

For your reference, the I/O registers used by common serial and

memory devices are summarized below. Note that a device doesn’t

necessarily use all registers it reserves, but it occupies more

address space due to “mirroring” of registers.

Memory drivers:

internal: No I/O registers used.

VDC+internal: Uses $D600-$D601.

REU: Uses $DF00-$DF0A, mirrored every 32 bytes.

BBG/GEORam: Uses $DE00-$DEFF and $DF80-$DFFF.

RAMLink: Uses $DE00-$DEFF and $DF10-$DFFF.

RAMDrive: Uses $DE00-$DEFF and $DF10-$DFFF.

Serial drivers:

User port: CIA #2 registers used.

UP9600: CIA #1 and #2 registers used.

SwiftLink: Uses $DE00-$DEFF ($DE00-$DE0F with AddressFixer).

Turbo232: Uses $DF00-$DFFF ($DF00-$DF0F with AddressFixer).

SL-DE20: Uses $DE20-$DE2F (with AddressFixer).

SL-DF20: Uses $DF20-$DF2F (with AddressFixer).

SL-D700: Uses $D700-$D7FF ($D700-$D70F with AddressFixer).

HART: Uses $DE18-$DE1F and $DF18-$DF1F.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

APPENDIX A. VENDOR INFORMATION

This section contains addresses and phone numbers of vendors for

the various hardware products listed in this documentation.

A.1 Enhanced RS232 interface (UP9600)

The EZ232 is a serial card made currently by doppleganger for $30

plus $5 shipping. This allows Striketerm to work at 9600 baud

using the UP9600 driver and is the only card in production at

this time for the User Port on a real C64 that can achieve

higher speeds than 2400 baud with Striketerm.

Doppleganger can be reached at irc.newnet.net #c64friends and

by email at dopple@doppleganger.org . Please email him before ordering.

A.2 Strikelink / User Port -> USB Driver

This is a DIY project. You can build your own for less than $15 in

parts, but feel free to email me at alwyz@sceneworld.org

if you are unable to build one.

A.3 Comet / Flyer

Comet and Flyer are wiznet devices that connect your C64

to the net with a direct ethernet connection.

Comet: http://commodoreserver.com/

Flyer: http://www.retroswitch.com/products/flyer/

A.4 Turbo232, BBGRam, RAMDrive, BBRTC

The Turbo232 cartridge contains a 6551 UART chip and a standard

RS232 serial port, enabling speeds up to 38,400 bps.

The BBGRam cartridge is a memory expansion device available in

sizes of 512K, 1 MB, or 2 MB of RAM.

The RAMDrive is an intelligent, transparent memory expansion

device. It is available in sizes up to 8 MB.

The BBRTC is a small, battery-backed real-time clock chip that

plugs into either joystick port.

The above hardware was sold by Performance Peripherals, Inc.,

which is no longer in business.

A.5 SwiftLink, RAMLink

The SwiftLink cartridge contains a 6551 UART chip and a standard

RS232 serial port, enabling communication speeds up to 38,400

bps.

The RAMLink is an intelligent memory expansion device that uses

SIMM chips. It is expandable up to 16 MB.

The SwiftLink and RAMLink were made by Creative Micro

Designs, Inc.

http://www.cmdweb.de/

The Swiftlink and Ramlink are no longer in production

A.6 HART cartridge

The HART cartridge uses an 8250 UART chip connected to a

standard RS232 serial port, achieving speeds up to 57,600 bps.

The HART cartridge is sold by Hatronics.

The HART cartridge is no longer in production.