

CODEBUSTER

OPERATING MANUAL

MARSHVIEW SOFTWARE

MARSHVIEW SOFTWARE



PRESENTS

CODEBUSTER



COPYRIGHT 1985 MARCEL LEBLANC & RON SMITH

TABLE OF CONTENTS

CODEBUSTER INTRODUCTION.....	1
HOW IT WORKS.....	1
REQUIREMENTS.....	2
SET UP PROCEDURE.....	3
DESCRIPTION OF TERMS USED.....	3
DESCRIPTION OF FUNCTION KEYS.....	4
CAPABILITIES.....	5
USING CODEBUSTER.....	7
EXTRA FEATURE.....	9
COPYRIGHT NOTICE.....	12
LIMITED WARRANTY.....	13
DISCLAIMER.....	14

NOTE

We realize that some of you won't be able to wait long enough to read this manual thoroughly before trying out your newest acquisition. So for those of you who fall into this category, GO DIRECTLY TO PAGE 7 and read the section called 'USING CODEBUSTER'.

CODEBUSTER INTRODUCTION

Congratulations and thank you for your purchase of Codebuster. We hope that you find it useful and are confident that you will be very satisfied with it.

To make full use of your Codebuster system (and to save you needless frustration), please read the documentation carefully. A few minutes should be enough to totally familiarize yourself with the correct procedures to operate the system. Many hours have been spent to make Codebuster as simple and user friendly as possible. So to help yourself (and make us believe that we didn't waste money on this paper) please spend a few moments and read this manual.

HOW IT WORKS

The premise of Codebuster is unique and really very simple (the premise, not the programming). When a program loads, the protection scheme is USUALLY (see CAPABILITIES) executed during the load. If the protection is found then the program continues to load. If it is not, well, we all know what happens then. The program that is put into memory is usually free of the protection scheme. This is the point where Codebuster comes in. Although Codebuster is a cartridge that plugs into the 64, until it is activated it is completely invisible.

SET UP PROCEDURE

Not even the computer knows that it is there! With the simple push of a button and the press of a key, Codebuster is activated. It immediately seizes control of the computer and begins to analyze the code contained therein.

Its first function is to locate all of the relevant code necessary to the proper functioning of the program. When this is accomplished it downloads the selected memory into a small number of convenient files complete with an auto-loading boot. These files can then be loaded from any 64 compatible drive.

The broken program will load faster than the original especially if used with one of the commercially available 'fast loaders'. The program can also be saved on a disk with other programs thus allowing you to make more efficient use of your available disk space.

REQUIREMENTS

Codebuster requires a Commodore 64 and a compatible disk drive. Usually a 1541 works best as many copy-protected programs will not load on anything but a 1541. Codebuster also works with an SX-64. The package that you received should contain the following components:

Codebuster cartridge
Documentation
Template
Warranty registration card

Here is where you will really begin to appreciate the ease of use of this product. Simply TURN OFF your 64 and then insert the Codebuster cartridge. Now turn on your computer and then the disk drive. That's all there is to it! There are no special programs to load or switches to check. You're ready for your first encounter.

DESCRIPTION OF TERMS USED

The following is a list of some of the terms used that initially may be found to be confusing:

- 1) Codebuster...the cartridge that you paid that ridiculously small amount for.
- 2) Original disk...the original program disk.
- 3) Target disk...the disk that will contain the broken version of the program.
- 4) Template...the cardboard overlay that came with your package. This is designed to fit over your function keys and to remind you of the function of the various keys.

DESCRIPTION OF FUNCTION KEYS

Before you go racing off to tackle that first conquest you should familiarize yourself with the purpose of the function keys and the button on the cartridge. The following is a brief overview:

Cartridge button--This 'freezes' the computer at the screen currently displayed.

F1--Pressing this key instructs Codebuster to begin the process of analyzing and saving the program.

F3--This key clears the computer's memory which is necessary for the minimizing function.

F5--Fondly referred to as the panic button. It is used when you have begun the Codebuster process by loading in the original program and realize that, much to your horror, you do not have a blank formatted disk and are not sure that you have a disk lying around that has at least 273 blocks free (the most space that the total Codebuster files could take). Pressing this key will format a disk with a pre-determined header and a random ID. Press this button instead of F1.

F7--This is the abort/resume key. It is used when you mistakenly push the cartridge button before the desired screen has appeared. Most of the time the program will continue after this key

is pressed. However, sometimes this will not work due to the way that the original was written and you will have to start all over again.

CAPABILITIES

Codebuster will work on virtually all software programs that load entirely into the computer's memory. There are some programs that cannot be defeated by it, however. Problems will be encountered with the following types:

- 1) Programs that do a protection check well into the game.
- 2) Programs that store code in the 1541's memory.
- 3) Programs that are dongle, cartridge or key protected.
- 4) Programs that refer back to the disk and expect to find the information at a specific track and sector. There IS a cure for this. Simply do a straight backup of the original and place it on another disk or on the flip side of the disk that has the files produced by Codebuster.

If the program simply refers back to the disk for file information and does not care where it is found, all you have to do is copy those files onto the target disk after it has the files produced by Codebuster. To do this, use any of the

many file copy utilities that are available commercially or public domain. You could use the file copier that came on your demo disk with your disk drive.

It should be pointed out again that it is a very small percentage of programs that cannot be successfully copied using Codebuster.

If you wish to find out if a particular program contains a type of protection that Codebuster cannot get past, try using the following procedures.

Protection that is checked after the program has loaded and begun can be detected in the following manner. After the program has loaded and the drive is INACTIVE, replace the original disk with a straight backup of the program. DO NOT attempt to duplicate the protection of the original. Continue using the program and watch for the first time that the program accesses the drive. If the program hangs up or does not act correctly then you can assume that a protection scheme has been checked for and not found. On the other hand, if everything appears to work normally, then you know that there is no protection scheme left.

Protection that resides in the RAM of the disk drive can be determined with equal ease. First load the original program as you would normally. When the program stops, at an option screen for instance, try turning off your drive and then turning it on again. Choose an option and watch to see if the program

can successfully access the drive. If something strange happens, such as the computer hanging up or the screen displaying garbage, you can safely assume that the program is using drive protection. If the program accesses the drive successfully there should be no problem copying the program using Codebuster.

USING CODEBUSTER

The following steps should be used when copying a program with Codebuster:

- 1) Place the Template over the function keys.
- 2) Make sure you have a blank formatted disk or a disk with at least 273 blocks free. This is your target disk.
- 3) With the cartridge inserted (see SETUP PROCEDURE) press the button on the Codebuster cartridge. You will notice that the screen goes blank.
- 4) Press F3. The screen should come back with the message 'CLEARED' and a blinking cursor.
- 5) Load in the original disk as you would normally. *NOTE* Please ensure that your original program disk is write protected as you may inadvertently insert the wrong disk or simply forget to change disks when prompted. Mistakes do happen.

6) Press the button on the Codebuster cartridge when the program has loaded and the first option screen appears (preferably one with little music or movement although this is not crucial and is for simple aesthetics). The screen will blank to the current border color. At this time remove the original disk and turn your drive off and then on again. If your using an SX push the drive reset.

7) Insert your target disk and then press the F1 key. The drive will come on and after approximately 25 seconds the screen will turn dark blue and the message 'WORKING...' will appear. If you have forgotten to format a new disk press F5 instead of F1. In this case the drive will activate and format a disk first. Then the blue screen and message will appear.

8) Within two to three minutes (depending upon the size of the program) a message will appear on the screen prompting you for two characters by which to identify the new files. Type in the two characters and then press return. The drive will again activate. The message '00,ok,00,00' should now appear. This simply tells you that the drive channel is clear and drive functions will be successful. Now the message 'RENAMING FILES...' will appear on the screen.

9) In approximately twenty seconds another message will appear asking for a name for the new boot. This name must be sixteen characters or less. We would suggest that you use the original name of the program. Once you have typed in the

desired name press return. The drive will come on again and the drive channel status will be displayed. Shortly thereafter the drive will stop and the message 'CODEBUSTER PROCESS COMPLETED' appear.

YOU'RE DONE!!

Codebuster has now finished and again is invisible. At this point you can load in the new program by typing-- LOAD "name of program",8,1-- and pressing return (if there are no extra files to be copied by way of a separate file copier).

EXTRA FEATURE

There is a special program built into Codebuster that we call CODE INSPECTOR. This will only be of benefit to those familiar with machine language (hereafter called ML). It is meant to be a tool that can be used to tackle those programs that cannot be broken by Codebuster. It will provide the experienced user with the ability to interrupt a program at any time. The only segments of code that will be corrupted will be screen memory and bytes between \$0808 and \$0D00.

This feature is accessed by first pressing the button on the cartridge. The screen will freeze and default to the border color. Pressing the RETURN key will turn the screen black and the current status of the following will be displayed:

- 1) Program Counter (PC)
- 2) IRQ Vector
- 3) NMI Vector
- 4) Processor Status Register (SR)
- 5) Accumulator (AC)
- 6) X Register (XR)
- 7) Y Register (YR)
- 8) Stack Pointer (SP)

Also displayed will be the contents of the following:

- 9) 6510 on-chip I/O registers (0&1)
- 10) Vic II Chip IRQ Mask & Raster Compare latch value
- 11) CIA ICR mask values and timer latch values

There will be four options displayed underneath this information:

The first (L) will allow you to load and execute a machine language file from disk. This could be used to load in a ML monitor or perhaps a custom program. It will ask for the name of the file and then for the jump address. So, if for example, you wanted to load a monitor called MYMON and the entry point was 49152 you would press L. When the first prompt appeared you would type MYMON and press return. On the second prompt you would type C000 (hex for 49152). To ensure that your ML monitor will function properly, Code Inspector will restore memory between \$0000 and \$0400 to its normal state after a reset (power up) BEFORE it loads in your ML program.

The second option (J) will be to jump to any address point in memory. This could be used to test for proper entry points for program start up. You would type J and then enter the desired address when prompted. If you simply want to resume at the last address executed, enter 'R' at the 'JUMP ADDRESS?' prompt. Code Inspector will restore as much memory as it can before resuming execution, including the VIC II chip and the two CIA's. Keep in mind that some low memory had to be destroyed to make room for Code Inspector, so your program may not resume normally.

Option number three (D) will allow you to display memory in hex. Enter the start and end addresses when prompted. Memory displayed in the address ranges \$A000 to \$BFFF and \$D000 to \$FFFF will depend on the state of byte \$01 at the time that your program was interrupted. Similarly, the values of the IRQ vector and the NMI vector that are displayed on the screen when Code Inspector appears will be taken from addresses \$FFFE and \$FFFA respectively if the Kernal ROM is not banked in at the time your program is interrupted. All memory dumps are to screen only.

The fourth option (B) will return you to basic.

The possibilities that this feature opens up should be apparent by now. For example, if there is a program that is using disk protection that is accessed during the running of the program, you can quickly determine where it is in

memory by interrupting at the point of the protection check. The information displayed on the screen will indicate where the code in question is located. Based on this you can load a monitor into another area of memory and jump to it. From there you can study the code that is involved in your 'problem'.

This is only one of the possible uses of this feature. Experiment with it. You will find that it can be of invaluable benefit.

GOOD LUCK!!

COPYRIGHT NOTICE

Codebuster concepts by Marcel LeBlanc and Ron Smith
Programming by Marcel Leblanc
Codebuster board layout by Cadmi
Microelectronics Ltd., Fredericton, N.B.
Manual by Ron Smith
Software and hardware Copyright (c) 1985
Marcel LeBlanc and Ron Smith

Thanks to Marshview Software for the support and help.

And you too, Rob!

LIMITED WARRANTY

Neither the authors nor the distributors of this product shall be liable for any damages which may be caused by any errors or omissions in this product. Should the product be defective, the distributor shall replace it upon return of the defective product, postage paid, within ninety days of the date of delivery. There are no other warranties implied or expressed, including but not limited to, any implied warranties of merchantability or fitness for a particular use.

The warranty registration card must be on file for repair or replacement under warranty. It is assumed that any Codebuster package that is returned without proof of purchase is void of warranty. This warranty shall be void if, in the opinion of the authors, this product has been misused, improperly installed, modified or otherwise tampered with.

Your Codebuster package should be sent to:

Marshview Software Limited
P.O. Box 1212
Sackville, New Brunswick
Canada EOA3C0

Please make sure that your Codebuster package is protectively packaged as damage due to shipping is not covered by warranty. We would suggest that you insure your Codebuster package.

DISCLAIMER

This product is meant expressly for the archival backup of your legitimate software.

Neither the authors nor the distributors of Codebuster condone the use of this product to assist in software piracy.

