

# HANDY RELEASE 0.8.3

30 May 1989

CONFIDENTIAL and PROPRIETARY

This document describes the V 0.8.3 release of the Handy development environment. It includes these sections:

- Auto-Timeout
- New CART Rules
- New Cartdefs.i Definitions
- Other 6502: Changes
- HandyAsm 1.04
- Handebug 1.59
- New Hardware Manual Pages
- What You Must Do To Start Using This Release

## Auto-Timeout

The system now provides an auto-timeout capability, which you get simply by defining AUTO\_TIMEOUT\_USER.

With auto-timeout you get 2 timeout durations. If nothing is plugged into the external power jack, the timeout period is approximately 5 minutes. If there's external power, the timeout period is 60 minutes. These durations are subject to change.

For auto-timeout to work, you must either be a GETJOY / GETSWITCH macro user or you must invoke the RESET\_TIMEOUT macro every time button / switch activity is detected (i.e. whenever you find non-zero values in the input bits).

Soon to be implemented will be a PAUSED timeout value, where, if you've set a system flag named PAUSED and no external power jack is plugged in, then the system will use a timeout duration of 15 minutes.

Auto-timeout adds 39 bytes to your code. It lives in the end-of-frame handler, with something like an average of 2.05 opcodes added to the handler execution. The timeout durations are based on 60 frames per second frame rate, so if you run your display at 50 the durations will be a bit longer and if you run at 75 they'll be a bit shorter. You don't need to be a EOF\_USER to be an AUTO\_TIMEOUT\_USER.

## New CART Rules

In order to support the new security system algorithms, there's new rules regarding the cart file data you must provide for the system. Now there are 2 required files. File 0 contains what used to be in file 0. File 0 now must contain a palette and a list of sprites. The palette will be loaded and then the sprites will be displayed by the cartridge boot code while the security check is being performed on the cart. After the cart passes the security check, the normal boot code, now addressed by file 1, will be loaded and executed.

The file 0 sprite list must create a display that presents the name of your game, with its accompanying™ or®. This name will be visible for about 5 seconds or so (we'll know more later) while the security check is in progress.

After the security check is successfully completed, the display will blank to a color of your definition. Also, during the security process if the player presses any of the buttons (except for PAUSE) the screen will blank to this color. You define this color by defining ROM\_SCREENBLANK\_VALUE in your cartdefs.i file. This is described more fully in the *New Cartdefs.i Definitions* section below.

The rules for file 0 data layout:

- The file's data must be .ORG'ed at or above \$2400, can range all the way up to \$FBFF.
- The first 32 bytes must contain palette data in ascending order. The first byte will be written to GREEN0, the 32nd byte will be written to BLUERED15.
- Immediately after the palette bytes should come the first byte of an SCB. Any reasonable SCB list is allowable. All types of sprites are allowable. Start with an EraseSprite, as you don't know what data will be in the display buffer. Presume an HOFF and VOFF of zero.

New rule for file 1 code:

- The file's data must be .ORG'ed at or above \$0400, can range all the way up to \$FBFF.

This release includes a new BOOT CART program, which program will display your file 0 sprite list for a short while before proceeding with the normal boot. You can interrupt this short display and get on with the boot by hitting one of the buttons.

You can freeze your sprite display by loading and executing HANDY:cartgo.bin.

## New Cartdefs.i Definitions

There's 3 new ROMDIR\_FILE1\_mumble definitions. These can be found at the end of the file 6502:examples/cartdefs.i. You should copy these verbatim into your own cartdefs.i file.

If you want, you can define the constant ROM\_SCREENBLANK\_VALUE to define the value to which the screen will blank while loading and executing your game cart file 1. The screen will blank to the color also when the user hits a button during the security check of the cart.

The value that you define will be copied to all of the hardware color registers. This gives you a limited but useful set of colors that can be defined. Here's a table of examples:

<u>If ROM_SCREENBLANK_VALUE is</u>	<u>the display will be blanked to this color</u>
\$00	Black (the default)
\$F0	Fully saturated blue
\$0F	Fully saturated yellow
\$FF	Fully saturated white
\$A3	Blue-grey
\$88	Medium grey

## Other 6502: Changes

Gone: the LIGHT\_OFF definition and the ability to power down the tube.

New: the IODIR / IODAT flag EXTERNAL\_POWER, the state of which reflects whether or not an external power plug is connected. When EXTERNAL\_POWER is set, a plug is plugged in.

There's a new hprint.mac file, with better HPRINT and HPR\_CHARSET macros that use capabilities of the new assembler. For a laugh, have a look at the new HPR\_CHARSET macro.

There are no more LEFT\_ and RIGHT\_FIRE\_BUTTON definitions. From now on you should use OUTER\_ and INNER\_BUTTON definitions.

## HandyAsm 1.04

First cut at improving error reporting and listing capability. Try them out and tell me what you think.

The listing works by combining the source with the binary image. It works like this: you assemble without a listing, and then assemble with one. The listing assembly doesn't really assemble, it just translates.

I'm dubious. But Carl claims that it provides the most complete of all possible listings (there are, sez he, some extremely difficult problems with creating a listing during the single pass).

## Handebug 1.59

The .config file supports the romfile and rom size settings. Too bad the current implementation of the download protocol doesn't support rom size and port. We have to wait to get on Howard Board for that.

When breakpoints are disabled via loading a .config file they are now really disabled.

All breakpoints are disabled (but not cleared) when ROM is downloaded.

The default path for symbols is set when you download a .bin or .rom file, not when you select symbols. This means that the default symbol file will always reflect what you've downloaded most recently.

Supports new BOOT CART program by waiting 10 seconds before timeout.

## New Hardware Manual Pages

There are some new hardware manual pages.

## What You Must Do To Start Using This Release

Make your old file 0 be your new file 1.

Make sure your new file 1 is .ORG'ed at or above \$0400.

Create a new file 0 containing a palette and sprite list.

Copy the 3 new ROMDIR\_FILE1\_mumble definitions from 6502:examples/cartdefs.i into the end of your cartdefs.i file. If you like, define a ROM\_SCREENBLANK\_VALUE (the default, by the way, is black).

Redefine your use of the values LEFT\_ and RIGHT\_FIRE\_BUTTON into INNER\_ and OUTER\_BUTTON.

## CONVENTION.