HANDY RELEASE 0.6.2
2 January 1989

CONFIDENTIAL and PROPRIETARY

Oh No! The Joystick Buster it is! Every man for himself!

This document describes the V 0.6.2 release of Handy development environment. It includes these sections:

- Joystick and Button Event Recording
- Video Recording
- ~~Handy Tape Recording~~
- 6502: Changes
- Handycraft V1.38
- What You Must Do To Start Using This Release

REQUEST OF ALL: this stuff is being rammed out in a big hurry. Some of it might not work. Please use this stuff right away and report bugs immediately and expect a bug-fix release within an hour from now.

WARNING TO ALL: when setting up your video display frame rate, use only the SETDISP_60 macro. There's some idiosyncratic behavior in the display hardware if you ask for slower frame rates.

Joystick and Button Event Recording

There's a new TAPE_RECORDING capability that we've created. A full description needs to be made for this stuff, but here's the short of it:

- Rather than including controls.src and controls,mac you include tapedeck.src and tapedeck.mac *and* you must include tapedeck.i. We suggest that you wrap all the changes within a constant named TAPE_RECORDING. See the file 6502:examples/testcontrols.src for an example of this usage.
- At the top of your code, invoke the NEWTAPE macro with a RECORD_MODE mode argument to get a "new event tape" into the system software.
- Within the main loop of your program, use TAPEDECK (rather than the new GETINPUT macro) to record any new button/joystick transitions. TAPEDECK acts as a call to GETINPUT, after which you can call GETJOY and GETSWITCH as normal.
- After your program has run and you've recorded the events of interest, get your code table onto the Handebug display and copy down the numbers in the table. Type these numbers into an array in your source code.
- After you've created an event table as described above, you can reassemble your program using the PLAY_MODE argument with the NEWTAPE macro. Then, calls to TAPEDECK (which replace calls to GETINPUT) will result in the input events coming from your table rather than from the actual input devices when you subsequently call GETJOY and/or GETSWITCH.
- See the document HANDY VIDEO TAPE RECORDING PROCEDURES for a description of how to tell Handycraft to do your bidding.

Ultimately, you'll be able to get an input event recording even from real hardware, but currently this stuff works only within the emulator.

Video Recording

Handycraft and now supports the auto-recording stuff described above. See the document HANDY VIDEO TAPE RECORDING PROCEDURES for a description of how to tell Handycraft to do your bidding.

6502: CHANGES

In 6502:examples, video.src, testsprite.src, testcontrols.src and testmath.src now start with a CLI and CLD and LDX #$FF and TXS. If your code doesn't start with at least CLI, you ought to ask yourself why.

There's a new constant that you must define, the HANDYMATH constant. This is used to tell the macros and math source files what sort of math you desire. HANDYMATH can have these definitions:

0 = Use real Handy math hardware
1 = Use normal Handy math software emulation of the hardware, The math registers are decalred to be in RAM locations, not the hardware registers (as before)
2 = Use normal Handy math software emulation of the hardware, with Steve's new fast multiply routine (which routine is much faster and much fatter than the old)

Those who are already using Steve's fast multiply are probably doing a JSR make_square. You don't need to do that anymore, and it is in fact wasteful for you to do so.

To get Larry's new tapedeck input event recording routines, you need to include 6502:src/tapedeck.src, 6502:macros/tapedeck.mac and 6502:include/tapedeck.i. You should use something like what can be found in 6502:examples/testcontrols.src. If I had more time I'd reproduce it here, but for now go look there to see what we mean.

The input macros now correctly return values as they would look when gotten from the real hardware, so all you input code is broken by this release. Also, before you call GETJOY and/or GETSWITCH you must first call the GETINPUT macro. GETJOY returns the value of the JOYSTICK hardware register or a faked facsimile thereof. GETSWITCH does the same with the SWITCHES register. Also, new switches and fire button definitions can be found in harddefs.i.

Here;s some new control bits to note (defined in harddefs.i): EVER_ON and NO_COLLIDE. EVER_ON gets set if your sprite was ever on the display, else it is cleared. NO_COLLIDE is a bit that you can set sprite-by-sprite that allows you to turn off collision processing for that sprite regardless of its TYPE definition.

For you Apple keyboard readers, we support you now too. You can use the normal input macros GETINPUT followed by GETJOY and GETSWITCH *and* read a RAM location named AppleKey to discover which key was actually pressed by the user. If no key was pressed, AppleKey will equal zero.

Handycraft V1.38

Handycraft now supports the auto-recording stuff described above. See the document HANDY VIDEO TAPE RECORDING PROCEDURES for a description of how to tell Handycraft to do your bidding.

Just doing 'x' (without 'f' or 'F') ought to work just fone now.

There are now two out-of-memory messages. Out of Darn Memory is the general one. Out of Memory Manager CHIP RAM is the message you get when the Memory Manager needs more CHIP RAM.

What You Must Do To Start Using This Release

Work hard: 2.5 shopping days left until CES DROP DEAD.