

wAx Cartridge Manual Addendum

Included Plug-Ins

Plug-In Installation

The wAx plug-in vector is at memory locations 5 and 6, so you can enable a wAx plug-in with the following command:

@0005:00 page

where *page* is the plug-in's page number. For example, to install Diff (see below), enter

@0005:0079

Included Plug-In Summary

Plug-In Name	Page	Purpose
Code Instrumentation	\$78	Debugging aid
Diff	\$79	Memory comparison
Text	\$7a	Text display
ML2BAS	\$7b	BASIC code generation
Character Helper	\$7d	Custom character generation
wAxScore	\$7e	Music data entry

Code Instrumentation Plug-In

Activation from Cartridge

@0005:0078

Purpose

It might be easier to read posts about this technique at this forum topic:

<http://sleepingelephant.com/ipw-web/bulletin/bb/viewtopic.php?f=2&t=9770>

The Code Instrumentation plug-in automatically generates (and, later, removes) what this forum topic calls a "knapsack." It's basically a breakpoint that can be continued from.

Usage

To create a breakpoint, use

```
'addr
```

Where *addr* is the breakpoint address. The plug-in will generate a "knapsack" of up to ten bytes at \$03ef. Your code will JMP to that location and BRK. You may continue your code with

```
←
```

To remove the breakpoint, and restore your code, use

```
,
```

on a line by itself.

Note: A knapsack may not contain a relative branch instruction. If it the plug-in encounters a relative branch instruction at or immediately after the breakpoint, it will issue ?UNDEF'D STATEMENT ERROR

Diff Plug-In

Activation from Cartridge

@0005:0079

Purpose

Diff provides a list of similarities and differences (comparison) between two blocks of memory of the same size.

Usage

```
'start1 end1 start2
```

Where *start1* is the start of the first block of memory, *end1* is the end of the first block of memory, and *start2* is the start of the second block of memory.

Diff will indicate addresses where the memory matches (in green) and differs (in red), and the number of matching or differing bytes starting at that address.

Text Plug-In

Activation from Cartridge

@0005:007a

Purpose

Text will show 256 bytes (16 rows x 16 columns) of PETSCII text starting at the specified address.

Usage

£addr

Where *addr* is the starting address of the text. You may press STOP to end the listing. You may hold down SHIFT (or engage SHIFT LOCK) to keep the listing going until SHIFT is disengaged. When the listing ends, you may press £ again, or press RETURN, to see the next 256 bytes of text.

ML-to-BASIC Plug-In

Activation from Cartridge

@0005:007b

Purpose

ML2BAS converts the 6502 code between *start* and *end* into a BASIC program in the current BASIC stage. The 6502 code will be appended to the existing program, with line numbers in increments of 5.

Usage

'start end+1 [R,H,T]

Where *start* is the start address, *end* is the end address, and R, H, and T are options.

If R is specified after the end address, uses the relocatable "hermit crab" syntax. Otherwise, uses absolute addresses.

If H is specified after the end address, creates hex data entry lines instead of 6502 code.

If T is specified after the end address, creates assertion test data instead of 6502 code.

If the disassembly would extend beyond the end of the BASIC stage, the existing BASIC program (if any) is restored, and an OUT OF MEMORY error is displayed.

Character Helper Plug-In

Activation from Cartridge

@0005:007d

Purpose

Character Helper assists in the design of custom characters by providing an 8x8 character creation matrix at the top of the screen, and loading custom character designs from that matrix directly into memory.

Usage

' [addr]

When no *addr* is provided, Character Helper draws an 8x8 canvas of colon characters at the top of the screen.

When *addr* is provided, the character in designed in the canvas is entered into memory at the specified address.

Inside the canvas, the colon character is interpreted as a 0 bit. Any other character is interpreted as 1.

wAxScore Plug-In

Activation from Cartridge

@0005:007e

Purpose

wAxScore eases the entry of wAxScore-format music data into memory.

Usage

'addr [R]

Where addr is the address of the score data. If R is included after the address, the wAxScore will be in Record mode. Otherwise, wAxScore will play from the data at the specified address.

Key Bindings for Recording

Notes	2	3	5	6	7			
	Q	W	E	R	T	Y	U	I
	c	d	e	f	g	a	b	c'

Rests SPACE BAR

Durations

f1 - Eighth Note

f3 - Quarter Note

f5 - Half Note

f7 - Whole Note

. - Dot the current duration (may be done multiple times)

Effects RETURN (No Effect)

Plus (Octave Up)

Minus (Octave Down)

Colon (Legato On)

Semicolon (Legato Off)

Undo DEL KEY

Finish STOP KEY

The wAxScore Format

wAxScore is a simple format for one voice. Each byte contains duration and degree data. The high nybble is the length:

1000nnnn = Whole Note

0100nnnn = Half Note

0010nnnn = Quarter Note

0001nnnn = Eighth Note

Note durations can be dotted by setting the bit to the right. For example, 0011 represents a dotted quarter note.

The low nybble is the note number of a chromatic degree from 1 to 13. wAxScore format does not actually specify the temperament that's determined by the table that defines relationships between chromatic degrees and note values.

A low nybble value of zero indicates a rest for the specified duration. If low and high nybble are both 0 (\$00), it indicates the end of the score.

A low nybble value of \$f indicates that the high nybble contains an effect. It is optional for wAxScore players to interpret effects, but they must treat effects as zero-duration. Effects include the following:

- * \$0 No effect - IMPLEMENTED HERE
- * \$1 Octave Up - IMPLEMENTED HERE
- * \$2 Octave Down - IMPLEMENTED HERE
- * \$3 Legato On - IMPLEMENTED HERE
- * \$4 Legato Off - IMPLEMENTED HERE
- * \$5 Crescendo (volume ++)
- * \$6 Decrescendo (volume --)
- * \$7 Accellerando (tempo ++)
- * \$8 Ritardando (tempo --)
- * \$9-\$f User effects

Note that effects may be implemented in IRQ players based on need or available memory. Pressing RETURN during Record will enter \$0f for an effect placeholder, which you can fill in with wAx's data entry tools.

wAxScore players may be developed that can play multiple voices from multiple scores simultaneously.

A sample wAxScore player is in the GitHub repository in src/user_tools.