

Apple II Technical Notes



Developer Technical Support

Apple IIGS

#25: *Apple IIGS Firmware Reference Updates*

Revised by: Dave Lyons and Jim Luther
September 1990
Written by: Rilla Reynolds
October 1987

This Technical Note includes updates to the May 1987 edition of the Apple IIGS Firmware Reference, published by Addison-Wesley (Part Number 030-3121-A). The new Monitor commands require an Apple IIGS revised ROM (Part Number 342-0077-B), which is available without charge from an authorized Apple dealer. Please contact Apple II Developer Technical Support at the address listed in Apple II Technical Note #0 if you have additional corrections or suggestions for this manual.

Changes since May 1990: Added a section about register sizes for several \$E1 vectors (Appendix D).

Contents

Page vii, Chapter 7 SmartPort Firmware: Change “Generic SmartPort calls 121” to “Standard and Extended SmartPort calls 121.”

Chapter 2: Notes For Programmers

Page 11, Environment for the Firmware Routines: Refer to Apple IIGS Technical Note #88, The Page One Stack in a 16-Bit World for more information on manipulating the stack pointer.

Chapter 3: System Monitor Firmware

Page 24, Table 3-1 (continued), Monitor commands grouped by type: Add a miscellaneous-type and a debugging-type Monitor command to the table, as follows:

Command type	Command format
...	
Quit Monitor	Q
<i>Install Visit Monitor and MemoryPeeker desk accessories#</i>	
...	
Enter mini-assembler	!
<i>Set flags (e, m, x) for full-native mode</i>	<i>Control-N</i>

Page 43, Back to BASIC: The last paragraph should read: “If you are using DOS 3.3 or ProDOS®, use the Monitor Q (Quit) command to return to the language you were using with your program and variables intact.”

Page 48, Table 3-6, Commands for program execution and debugging: Add a Monitor command to the table:

Command type	Command format
...	
Enter mini-assembler	!
<i>Set flags (e, m, x) for full-native mode</i>	<i>Control-N</i>

Page 66, after final paragraph: Add a new Monitor instruction heading and description:

Native Mode Set Control-N (Native Mode)

Control-N sets the m, x, e flags to 0 for full-native mode. All other registers are unchanged.

Page 67, after final paragraph: Add a new Monitor instruction heading and description:

Turn on ROM Desk Accessories, #

Enables the currently available ROM desk accessories, Visit Monitor and Memory Peeker. These desk accessories remain active in the desk accessory menu until power is shut off. Control-Open Apple-Reset has no affect on these items. To exit the Visit Monitor desk accessory, press Control-Y then press Return. To exit the Memory Peeker desk accessory, press Q.

Chapter 4: Video Firmware

Page 77, Table 4-4, Control characters with 80-column firmware on: Change the actions taken by Control-E and Control-F to read (they were reversed):

Control character	Action taken by C3COUT1
Control-E	Turns cursor on
Control-F	Turns cursor off

Chapter 5: Serial-Port Firmware

Page 82, Compatibility: The second half of the third sentence in the first paragraph should read: "...the Apple IIGS hardware *is different* from that used on the SSC."

Page 91, Input buffering, BE and BD: This heading should be “Input/Output buffering, BE and BD.”

Page 94, Table 5-6: The Extended Interface footnote which states, “If the function call returns with the carry bit set...” is incorrect. For Apple IIGS ROM 01, the Extended Serial Interface does not return the error condition in the carry bit. Programs using the Extended Serial Interface should check for a non-zero result value in the result code rather than the carry bit to determine if an error has occurred. For additional error handling information using the Extended Interface, see Apple IIGS Technical Note #50, Extended Serial Interface Error Handling.

Page 95, Error handling: The second sentence should read: “If the character has a framing or parity error (assuming that the parity option is not set to None), the character is deleted from the input stream and the appropriate *mode bit* is set.”

Page 96, Note: The Note should read: “The InQStatus elapsed-time counter functions correctly only if *a* heartbeat interrupt task has been started. *A* heartbeat interrupt task is a set of functions called by interrupt code that run automatically at one-thirtieth of a second intervals.

Page 96, Interrupt notification: The fourth sentence in the first paragraph should be: “The system interrupt handler will transfer control to the user’s interrupt vector *at* \$03FE in bank \$00.”

Page 97, Interrupt notification: The last three paragraphs should be replaced with this paragraph: “The interrupt completion routine executes as *part of the firmware interrupt handler* and must be run in that environment. The interrupt completion routine must preserve the DBR, speed, 8-bit native mode, D register, stack pointer (or just use the current stack), and MSLOT for proper operation. A/X/Y need not be preserved.”

Page 100, SetModeBits: The first sentence in the paragraph following the CMDLIST should read: “Use this call to alter any of the mode bits whose function is described *below*.”

Page 105, GetIntInfo: The command list should read:

```
CMDLIST    DFB      $03                ;Parameter count
           DFB      $0C                ;Command code
           DW       $00                ;result code (output)
           DW       $00                ;interrupt setting (output)
           DL       Completion address ; (output)
```

The following should be added after the command list: “Note: The Parameter count of \$03 is correct even though there are four parameters.”

The following should be added after the last paragraph: “Note: Before making this call from an interrupt completion routine, you must set the operating environment to look and act exactly like a 6502 in all respects. During interrupt completion routines, you must preserve the DBR, speed, 8-bit native mode, D register, stack pointer (or just use the current stack), and MSLOT for proper operation. A/X/Y need not be preserved. See “Environments for the Firmware Routines” in chapter 2, Notes for Programmers for details about setting and restoring the operating environment.

Page 106, SetIntInfo: The command list should read:

```
CMDLIST    DFB      $03                ;Parameter count
           DFB      $0D                ;Command code
           DW       $00                ;result code (output)
           DW       Interrupt setting  ; (input)
           DL       Completion address ; (input)
```

The following should be added after the command list, “Note: The Parameter count of \$03 is correct even though there are four parameters.”

Chapter 7: SmartPort Firmware

Page 120, Issuing a call to SmartPort: The standard and extended SmartPort call examples should be:

This is an example of a standard SmartPort call:

```
SP_CALL      JSR      DISPATCH      ;Call SmartPort command dispatcher
              DC       i1 'CMDNUM'    ;This specifies the command type
              DC       i2 'CMDLIST'   ;Word ptr to param list in bank $00
              BCS      ERROR          ;Carry is set on an error
```

This is an example of an extended SmartPort call:

```
SP_EXT_CALL  JSR      DISPATCH      ;Call SmartPort command dispatcher
              DC       i1 'CMDNUM+$40' ;This specifies the ext cmd type
              DC       i4 'CMDLIST'   ;Pointer to the parameter list
              BCS      ERROR          ;Carry is set on an error
```

Page 121, Generic SmartPort calls: Change occurrences of “Generic SmartPort Calls” to “Standard and Extended SmartPort Calls” in the header and the first sentence. Refer to SmartPort Technical Note #2, SmartPort Calls Updated, for updated information on the SmartPort STATUS call.

Page 122, Statcode = \$00: Change the function of bit 0 of the first device status byte to: “1 = Device currently open (character devices only) or disk switched (block device only).”

Page 124: SmartPort device types should be same as those documented in SmartPort Technical Note #4, SmartPort Device Types.

Page 125, SmartPort driver status: See SmartPort Technical Note #2, SmartPort Calls Updated, for the correct format of the status list for unit 0, status code 0.

Vendors must request a Vendor ID Assignment from Developer Technical Support before using a specific value in bytes two and three.

Page 125, Possible errors: Add the following:

\$1F	No interrupt. Interrupts not supported.
\$2B	No write. Disk write-protected.
\$2F	Offline. Disk off-line or no disk in drive.

Page 126, ReadBlock: Add a sentence at the end of the first paragraph which reads, “On return, the X and Y registers indicate the number of bytes transferred.”

Page 131, Open: The following changes apply for the CMDNUM:

	Standard call	Extended call
CMDNUM	\$06	\$46

Page 132, Read: Add a sentence at the end of the first paragraph which reads, “On return, the X and Y registers indicate the number of bytes transferred.”

Page 140, Figure 7-8, Disk-sector format: Change to the following:

13 5-Nibble SelfSync Fields	FF	D5	AA	96	Track	Sector	Side	Format	AdrsLRC	DE	AA	FF	1 5-Nibble SelfSync Field	FF	D5	AA	AD	Sector	699 GCR Nibbles	4 Checksum	DE	AA	FF
--------------------------------------	----	----	----	----	-------	--------	------	--------	---------	----	----	----	------------------------------------	----	----	----	----	--------	--------------------	------------	----	----	----

A SelfSync Field is four 20us selfsync nibbles written as a sequence of five 16us nibbles.

Page 140, ResetHook: The Control code and Control list should be:

Control Code	Control list
\$06	Count low byte \$04
	Count high byte \$00
	Hook reference number \$xx, \$00, \$00, \$00

Page 141, SetInterleave: The Control code and Control list should be:

Control Code	Control list
\$0A	Count low byte \$01
	Count high byte \$00
	Interleave \$01 to \$0C

Page 143, UniDiskStat: The Status code and Status list should be:

Status Code	Status list
\$05	Byte \$04
	Soft error \$00
	Retries \$xx
	A register after execute \$xx
	Y register after execute \$xx
	P register after execute \$xx
	Byte \$xx

Page 152, Passing parameters to a ROM disk: Add a sentence to the end of the second paragraph which reads: "These locations will not be preserved between SmartPort calls."

Page 156, Table 7-6, SmartPort error codes: Add the following error code:

Acc value	Error type	Description
\$69	IOTERM	I/O terminated due to new line

Page 166, Table 7-8, Standard command packet contents":

Byte 3 descriptions should read "Byte 2 of param list."
 Byte 4 descriptions should read "Byte 3 of param list."
 Byte 5 descriptions should read "Byte 4 of param list."
 Byte 6 descriptions should read "Byte 5 of param list."
 Byte 7 descriptions should read "Byte 6 of param list."
 Byte 8 descriptions should read "Byte 7 of param list."
 Byte 9 descriptions should read "Byte 8 of param list."

Chapter 8: Interrupt-Handler Firmware

Page 184, Serial-port interrupt notification: The last three paragraphs should be replaced with this paragraph: “The interrupt completion routine executes as *part of the firmware interrupt handler* and must be run in that environment. The interrupt completion routine must preserve the DBR, speed, 8-bit native mode, D register, stack pointer (or just use the current stack), and MSLOT for proper operation. A/X/Y need not be preserved.”

Chapter 9: Apple DeskTop Bus Microcontroller

Page 191, Sync, \$07: The first sentence should read: “This command performs *the three preceding commands in sequence*.”

Page 194, Receive Bytes, \$48: The fourth sentence should read: “The second byte value is a combination of *the device address in the high nibble and the ADB command in the low nibble* (see the *Apple IIGS Hardware Reference*).”

Chapter 10: Mouse Firmware

Page 201: Mouse button positions should be changed as follows:

- **X data byte**
If bit 7 = 0, then mouse button 1 is *down*.
If bit 7 = 1, then mouse button 1 is *up*.
- **Y data byte**
If bit 7 = 0, then mouse button 0 is *down*.
If bit 7 = 1, then mouse button 0 is *up*.

Page 205, Figure 10-1, Position and status information:

Bit 7 description should be: “Currently, *button 0* is up/down (0/1).”

Bit 6 description should be: “Previously, *button 0* was up/down (0/1).”

Appendix B: Firmware ID Bytes

Page 223, Table B-2, Register bit information: Change the table to show that Bits 7-0 of the *Y* register hold the ROM version number, and the *X* register is reserved. In addition, the table description should read: “The *Y* register contains the machine ID and the ROM version number. The *X* register is reserved.”

Page 249, COUT1: In the third sentence, change the value of line feed from \$8C to \$8A.

Page 277, RDALTZP: Change the comment to read: “Bit 7 = 1 if *alt* zp enabled.”

Appendix D: Vectors

Page 272: At the end of the introductory paragraph, add “The vectors TOWRITEBRAM through TOPRINTMSG8 must be called in eight-bit native mode.”

Further Reference:

- *Apple IIGS Firmware Reference*
- *Apple IIGS Firmware Reference 1MB Apple IIGS Update*
- Apple IIGS Technical Note #50, Extended Serial Interface Handling
- SmartPort Technical Note #2, SmartPort Calls Updated