Apple II Miscellaneous
#7: Apple II Family Identification

Revised by: Jim Luther May 1991
Written by: Cameron Birse and Matt Deatherage December 1986

This Technical Note describes the ROM identification bytes in the Apple II family.
Changes since November 1988: Added the identification bytes needed to identify the Apple IIe Card for Macintosh LC.

To identify which computer of the Apple II family is executing your program, you must check the following identification bytes. These bytes are in the main bank of main ROM (shadowed on the Apple IIGS), and you should make sure that this bank is switched in before making decisions based on the contents of these locations.

Machine	$FBB3	$FB1E	$FBC0	$FBDD	$FBBE	$FBBF
Apple][	$38		[$60]			[$2F]
Apple][+	$EA	$AD	[$EA]			[$EA]
Apple /// (emulation)	$EA	$8A				
Apple IIe	$06		$EA			[$C1]
Apple IIe (enhanced)	$06		$E0			[$00]
Apple IIe Option Card	$06		$E0	$02	$00	
Apple IIc	$06		$00			$FF
Apple IIc (3.5 ROM)	$06		$00			$00
Apple IIc (Org. Mem. Exp.)	$06		$00			$03
Apple IIc (Rev. Mem. Exp.)	$06		$00			$04
Apple IIc Plus	$06		$00			$05
Apple IIGS	(see below)					

Note: Values listed in square brackets in the table are provided for your reference only. You do not need to check them to conclusively identify an Apple II.

The Apple IIe Card for Macintosh LC uses the same identification bytes ($FBB3 and $FBC0) as an enhanced Apple IIe. Location $FBDD allows you to tell the difference between the Apple IIe Card and an enhanced Apple IIe because $FBDD will always contain the value $02 on the Apple IIe Card. Location $FBBE is the version byte for the Apple IIe Card (just as $FBBF is the version byte for the Apple IIc family) and is $00 for the first release of the Apple IIe Card.

The ID bytes for an Apple IIGS are not listed in the table since they match those of an enhanced Apple IIe. Future 16-bit Apple II products may match different Apple II identification bytes for compatibility reasons, so to identify a machine as a IIGS or other 16-bit Apple II, you must make the following ROM call:

```
SEC                     ;Set carry bit (flag)
JSR $FE1F               ;Call to the monitor
BCS OLDMACHINE          ;If carry is still set, then old machine
BCC NEWMACHINE          ;If carry is clear, then new machine
```

In all the current, standard Apple II ROMs, $FE1F contains an `RTS`. In the Apple IIGS, there is a routine that returns compatibility information in the A, X, and Y registers:

Bit	Accumulator	X Register	Y Register
Bit 15	Reserved	Reserved	Machine ID Number (0 = Apple IIGS)
Bit 14	Reserved	Reserved	Machine ID Number
Bit 13	Reserved	Reserved	Machine ID Number
Bit 12	Reserved	Reserved	Machine ID Number
Bit 11	Reserved	Reserved	Machine ID Number
Bit 10	Reserved	Reserved	Machine ID Number
Bit 9	Reserved	Reserved	Machine ID Number
Bit 8	Reserved	Reserved	Machine ID Number
Bit 7	Reserved	Reserved	ROM version number
Bit 6	1 if system has memory expansion slot	Reserved	ROM version number
Bit 5	1 if system has IWM port	Reserved	ROM version number
Bit 4	1 if system has a built-in clock	Reserved	ROM version number
Bit 3	1 if system has desktop bus	Reserved	ROM version number
Bit 2	1 if system has SCC built-in	Reserved	ROM version number
Bit 1	1 if system has external slots	Reserved	ROM version number
Bit 0	1 if system has internal ports	Reserved	ROM version number

Note: In emulation or eight-bit mode, only the lower eight bits are returned.

This ROM call is enough to determine if a machine is an Apple IIGS or equivalent.

Note: The original Apple IIGS ROM returns a faulty value in the accumulator. The value returned is $xx1F and should be $xx7F. If you see a $0000 in the Y register (i.e., Apple IIGS, ROM version $00), you should assume that the accumulator value is $xx7F.

The current Apple IIGS ROM (ROM version $01) sets all the registers correctly before returning from this call.

Further Reference

- Miscellaneous Technical Note #2, Apple II Family Identification Routines 2.1