Apple II
Technical Notes

$\square$

Developer Technical Support

Apple II Miscellaneous
#11:   Examining the $C800 Space from AppleSoft

Revised by: Matt Deatherage   May 1989

Written by: John Bennett   August 1987

This Technical Note discusses examining the $C800 space from AppleSoft BASIC with PEEK statements.

Changed since January 1989: Corrected the revision author name.

Both the 6502 and 65816 microprocessors perform a false read during absolute-indexed instructions. When AppleSoft interprets a PEEK statement, it performs an absolute-indexed LDA instruction with a base address such that a false read from $CFxx is performed. This read takes place during the formula translation of the expression passed to PEEK, not during the actual loading of the value.

Some peripheral cards have been designed to deselect their $C800 ROM space any time a $CF value is placed on the high-order address lines of the address bus. Therefore, if you use the AppleSoft PEEK statement to examine an address in the $C800 space of such a peripheral card, the $C800 space will be turned off when the statement is interpreted, and the value returned by the statement will not reflect the actual value in the $C800 ROM.

The 65C02, on the other hand, has been designed so that a false read is not performed for an absolute-indexed `LDA` instruction. As a result, if the `PEEK` statement is used to examine the $C800 space of the same peripheral card on an enhanced Apple IIe (or any other Apple II with a 65C02 installed), the $C800 space will not be deselected, and the value returned by the statement will accurately reflect the value in the $C800 ROM.

If it is absolutely necessary to examine the $C800 space from an AppleSoft BASIC program, it is safer to use a assembly-language routine to examine the addresses and pass the results to the BASIC application.