Apple II
Technical Notes

$\square$

Developer Technical Support

SmartPort
#7:   SmartPort Subtype Codes

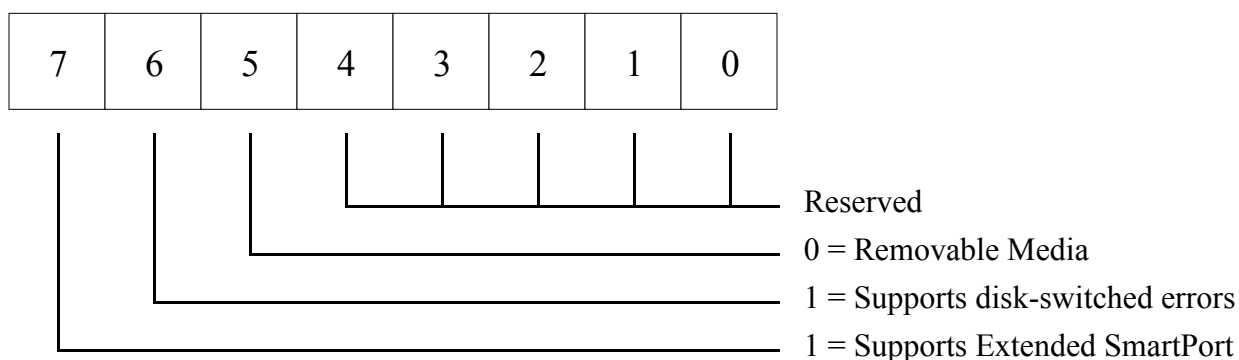Written by: Matt   Deatherage
               November 1988

This Technical Note clarifies information about SmartPort subtype codes.

Following is a definition of the SmartPort subtype code as given in the *Apple IIGS Firmware Reference*:

Figure 1–SmartPort Subtype Byte

Note that the value for subtype is <u>defined</u> for certain characteristics of the device; it is not <u>assigned</u> to the device as with Smartport device types (see SmartPort Technical Note #4, SmartPort Device Types for a complete list).

Attempting to distinguish different kinds of the same device by the subtype field can be confusing. For example, the Apple IIc Plus has an internal 3.5" disk drive. This drive does not support disk-switched errors nor does it support Extended SmartPort, and it has removable media. This

combination of features gives it a subtype definition of $00. However, this is the same subtype returned for a UniDisk 3.5. Any program which finds type $01 (3.5" Disk) and subtype $00 and assumes the drive is a UniDisk 3.5 will be misled by any other 3.5" drive matching the characteristics of the UniDisk 3.5.

Some Apple technical manuals state that the subtype byte may be used for identification purposes, but this cannot be supported if more than one variety of a specific device has the same characteristics and subtype.

To determine if a particular device type is the subtype you want, you may examine the name returned in the Device Information Block (DIB) from a `STATUS` call with `statcode` = 3. For 3.5" drives, however, this is not too helpful (both a UniDisk 3.5 and an Apple 3.5 Drive return `DISK 3.5`).

Because the subtype can not conclusively identify different flavors of 3.5" drives (and perhaps other individual device types), applications **must** look for errors on device specific calls and respond appropriately. Typical errors returned from making a device-specific call to the wrong device are $21 (`BADCTL`) and $22 (`BADCTLPARM`), although these are not the only ones. Also note that error codes in the range $20 - $2F are duplicated as $60 - $6F, the difference being that codes in the latter range are returned if the error was a soft error—a non-fatal error returned when the operation is completed successfully but an abnormal condition is detected.

The Reserved fields in the SmartPort subtype byte are reserved for future expansion. Present peripherals **must** have them set to zero so that they will not appear to support future features which are not presently defined. For this reason, programs checking the status of bits in the subtype byte should do so on a bit-by-bit basis **only**. For example, if you need to know if a device supports Extended Smartport, mask off all bits except bit 7 in the subtype byte before doing any comparisons. Blindly comparing to existing common subtype values (like $00 and $C0) will cause comparisons to fail when future bits in the subtype byte are defined.

Further Reference

- SmartPort Technical Note #4, SmartPort Device Types