New Technical Notes

Macintosh

®

Developer Support

Our Checksum Bounced
Devices

Written by: Jim Reekes   October 1989

This Technical Note discusses a fix to a SCSI Manager bug which concerns all developers working with SCSI and NuBus™ device drivers.

A Bit of History

The boot code contained in the ROM has a feature used by the Start Manager to perform a checksum on the SCSI driver being loaded. *Inside Macintosh*, Volume V-573, The SCSI Manager, documents this being performed on the Macintosh SE and later models for volumes using the new partitioning method. The truth, however, is that that checksum verification was never performed due to a bug in the ROM, and because of this, all drivers loaded regardless of validity.

That was the case until recently. On new Macintosh computers, the checksum verification works. That's the good news: we've fixed the bug. Now the bad news: this fix causes a number of third-party SCSI drivers to fail to load.

Some SCSI drivers improperly implement the new partitioning scheme. If the partition map entry name begins with the four letters "Maci" (case sensitive) and is of type "Apple_Driver", the driver now has its checksum verified with the entries in the partition map. If this checksum fails, the driver is not loaded. This checksum algorithm is documented in *Inside Macintosh*, Volume V-573, The SCSI Manager.

Drivers That Check In, But Don't Check Out

The checksum routine tests the number of bytes specified in `pmBootSize`, beginning at the start of the driver boot code. Only drivers contained within the new partition map have this test performed. If you are using the old partition map scheme documented in *Inside Macintosh*, Volume IV-283, The SCSI Manager, the driver does not have its checksum validated. The following is the startup logic in the new Macintosh ROMs:

```
    IF
        pmSig = $504D
```

```
AND
    pmPartName = Maci
AND
    pmPartType = Apple_Driver
```

```
    AND
         pmBootChecksum = ChecksumOf(bootCode, pmBootSize)
    THEN
         Load the driver
    ELSE
         Do not load the driver
```

Just When You Thought It Was Safe To Call _SysEnvirons

The call `_SysEnvirons` was created for compatibility reasons. It allows an application to make a single call to the system to determine its characteristics. It keeps the application from reading ROM addresses and low memory. This trap is now in the ROM of new machines. But, before you get excited about this addition to ROM, there is something that *Inside Macintosh*, Volume V-5, Compatibility Guidelines, states that must be understood by those writing SCSI drivers:

> *"All of the Toolbox Managers must be initialized before calling SysEnvirons."…"SysEnvirons is not intended for use by device drivers, but can be called from desk accessories."*

This statement means that neither SCSI nor NuBus device drivers can use `_SysEnvirons`. The earliest possible moment to call `_SysEnvirons` is at INIT time. Some SCSI drivers call `_SysEnvirons,` and this causes the Macintosh to crash at boot time.

To Sum Up

Check if your partition map is of the version described in the SCSI Manager chapter of *Inside Macintosh*, Volume V, and contains the `pmPartName` and `pmPartType` as mentioned earlier in this Note. If it does, then verify that the `pmBootChecksum` is correct. If the checksum is not correct, the new Macintosh computers will not load your driver.

The solution to this problem is to have a valid partition map entry in all cases and to expect the Start Manager to perform the checksum verification regardless of the `machineType`. `_SysEnvirons` is not available until the system has been initialized.

Further Reference:

- *Inside Macintosh*, Volume IV-283, The SCSI Manager
- *Inside Macintosh*, Volume V-5, Compatibility Guidelines
- *Inside Macintosh*, Volume V-573, The SCSI Manager
- Technical Note M.OV.GestaltSysenvirons —
    Gestalt and Sysenvirons : a Never Ending Story

NuBus is a trademark of Texas Instruments