Macintosh Technical Notes

#108: _AddDrive, _DrvrInstall, and _DrvrRemove

See also: Technical Note #36, Drive Queue Elements
SCSI Development Package (APDA)

Written by: Jim Friedlander March 2, 1987
Revised by: Pete Helme December 1988

_AddDrive, _DrvrInstall, and _DrvrRemove are used in the sample SCSI driver in the SCSI Development Package, which is available from APDA. This Technical Note documents the parameters for these calls.
Changes since March 1, 1988: Updated the _DrvrInstall text to reflect the use of register A0, which should contain a pointer to the driver when called. Also added simple glue code for _DrvrInstall and _DrvrRemove since none is available in the MPW interfaces.

_AddDrive

_AddDrive adds a drive to the drive queue, and is discussed in more detail in Technical Note #36, Drive Queue Elements:

```
FUNCTION AddDrive(DQE:DrvQEl;driveNum,refNum:INTEGER):OSErr;
```

A0 (input)	→	pointer to DQE
D0 high word(input)	→	drive number
D0 low word(input)	→	driver RefNum
D0 (output)	←	error code
		noErr (always returned)

_DrvrInstall

_DrvrInstall is used to install a driver. A DCE for the driver is created and its handle entered into the specified Unit Table position (–1 through –64). If the unit number is –4 through –9, the corresponding ROM-based driver will be replaced:

```
FUNCTION DrvrInstall(drvrHandle:Handle; refNum: INTEGER): OSErr;
```

A0 (input)	→	pointer to driver
D0 (input)	→	driver RefNum (–1 through –64)
D0 (output)	←	error code
		noErr badUnitErr

_DrvrRemove

_DrvrRemove is used to remove a driver. A RAM-based driver is purged from the system heap (using _ReleaseResource). Memory for the DCE is disposed:

```
FUNCTION DrvrRemove(refNum: INTEGER):OSErr;
```

D0 (input)	→	Driver RefNum
D0 (output)	←	error code noErr qErr

Interfaces

Through a sequence of cataclysmic events, the glue code for _DrvrInstall and _DrvrRemove was never actually added to the MPW interfaces (i.e., "We forgot."), so we will include simple glue here at no extra expense to you.

It would be advisable to first lock the handle to your driver with _HLock before making either of these calls since memory may be moved.

```
;---------------------------------------------------------------
; FUNCTION DRVRInstall(drvrHandle:Handle; refNum:INTEGER):OSErr;
;---------------------------------------------------------------

DRVRInstall    PROC   EXPORT
       MOVEA.L (SP)+, A1     ; pop return address
       MOVE.W  (SP)+, D0     ; driver reference number
       MOVEA.L (SP)+, A0     ; handle to driver
       MOVEA.L (A0), A0      ; pointer to driver
       _DrvrInstall          ; $A03D
       MOVE.W  D0, (SP)      ; get error
       JMP     (A1)          ; & split
       ENDPPROC


;---------------------------------------------------------------
; FUNCTION DRVRRemove(refNum:INTEGER):OSErr;
;---------------------------------------------------------------

DRVRRemove     PROC   EXPORT
       MOVEA.L (SP)+, A1     ; pop return address
       MOVE.W  (SP)+, D0     ; driver reference number
       _DrvrRemove           ; $A03E
       MOVE.W  D0, (SP)      ; get error
       JMP     (A1)          ; & split
       ENDPPROC
```