1.1 Title:

Effect of CLOSE $IO

1.2 MDC Proposer and Sponsor:

Susan A. Schluederberg
Connections Group, Ltd.
33 Wolf Trap Court
Baltimore, Maryland
USA 21236-2556
(410)931-0598 FAX: (410)931-0598

1.3 History:

18 June 1993	X11/SC12/93-33	This document has editorial changes per requests at SC12 meeting February 28, 1993
28 August 1992	X11/SC12/92-27	Proposal formalization modified for RMDS
14 June 1992	X11/SC1/91-46	SC12 elevated proposal to SC12 Type A status (11:1:0)
		Pro: Clarifies ambiguity
		Con: Not backward compatible in some implementations
12 June 1992	X11/SC1/91-46	SC12/TG9 recommended elevating proposal to SC12 Type A status (7:0:0)
31 May 1991	X11/SC1/91-46	Promotion of proposal as amended (to include language not to include it in the next standard) to SC Type A failed. 14:20:6
19 April 1991	X11/SC1/91-46	Amended document to separate out Null Device; To be proposed for SC1 Type A at May/June 1991 meeting
23 January 1991	X11/SC1/90-108	Combined with Null Device proposal (X11/SC1/90-59-4) Motion to elevate to type A failed 18:16:3 (2/3 majority required)
30 August 1990	X11/SC1/90-69	Document amended per June 8 1990 recommendation.
8 June 1990	X11/SC1/90-12	Elevated to Subcommittee 1 Type B Document; committee amended as follows: "If the current device is named in an argument of an executed CLOSE, $IO is *given a value of the empty string*."
3 February 1990	X11/SC1/90-12	New document incorporating changes from 3 October.
16 October 1989	X11/SC1/89-41	MDCC-E vote of support
3 October 1989	X11/SC1/89-41	Subcommittee 1 Type C Document; committee amended as follows: when $IO is CLOSEd the new value of $IO is always null;
28 August 1989	X11/SC1/89-41	original submission by Susan A. Schluederberg

2. Justification of Proposed Change

2.1 Needs

When the current device ($IO) is named in the argument of an executed CLOSE, the current standard gives the implementor a choice: they may either execute the (implicit) code `OPEN P USE P`, or they may give $IO an empty value. The device indicated by the P is some previously defined default device.

This specification has remained unchanged since the original standard (1977).

As the current standard is written, the effect of `CLOSE $IO` is undependable. The programmer cannot tell if the "default device" is available for use by checking the value of $IO.

2.2 Existing Practice in Area of the Proposed Change

In a 1989 survey conducted by the MUMPS Users' Group, twelve (40%) of the thirty implementations surveyed conformed to the 1984 standard. Of these, seven implementations performed the `OPEN P USE P`, two performed a `USE P` (the principal device is not CLOSEable), and three gave $IO a null value.

Of the remaining eighteen (60%) non-standard implementations, twelve gave $IO the value of the principal device without first performing the required `OPEN P USE P` and could produce an error if the principal device was not owned. One implementation does not perform the `OPEN P USE P`, but will allow a write to the device P even if it is not owned by the job. Two implementations do not change $IO ($IO identifies the device just CLOSEd). One implementation makes $IO undefined.

3. Description of Proposed Change

3.1 General Description of the Proposed Change.

$IO should have one of two general values: an empty value (indicating that no current device is specified), or a string value (indicating the current device). If $IO has a non-empty value, the device indicated is owned by the job and Input/Output (READs or WRITEs) will work appropriately.

When the current device is CLOSEd, $IO is given an empty value.

3.2 Annotated Examples of Use

```
DO PRINT                                        ; The subroutine is going to print a report.
IF $IO="" DO                                    ; On return from the print process, check
                                                ; if there is a current device. If not:
. OPEN $PRINCIPAL:1 IF '$TEST DO                ; reopen the principal device
. . FOR  HANG 60 OPEN $PRINCIPAL:1 QUIT:$TEST
                                                ; This wasn't suppose to happen, but since
                                                ; it has, we have to do something about it.
. USE $PRINCIPAL                                ; Make this the current device.
; continue processing                           ; In any case, now we can continue.
```

3.3 Formalization

I. 5.2.2 CLOSE command, second (last) paragraph, fourth and fifth sentences

replace:

"If the current device is named in an argument of an executed CLOSE, the implementor may choose to execute implicitly the commands OPEN P USE P, where P designates a predetermined default device. If the implementor chooses otherwise, SIO is given the empty value."

with:

"If the current device is named in an argument of an executed CLOSE, SIO is given a value of the empty string."

4. Implementation Impacts

4.1 Impact on Existing User Practices and Investments

For those implementations that currently adhere to the standard, there are two possibilities:

1. Currently the system performs the `OPEN P USE P` (30% of the implementations surveyed). For those cases when the principal device is currently owned by another job; processing waits until ownership of the principal device is obtained.

   Under this proposal, processing would continue and could produce an error in those cases where a process performs a `CLOSE $IO` and assumes SIO is then set to the principal device. Under these circumstances, this proposal is not backwards compatible.

   Under this proposal, if the old method is desired (that is, wait until the principal device is obtainable), the following code can be used:

   ```
   CLOSE $IO OPEN $PRINCIPAL USE $PRINCIPAL
   ```

2. The system currently gives SIO the empty value (10% of the implementations surveyed). For those cases when the principal device is currently owned by another job, a process must explicitly OPEN and USE some device before performing any further IO.

   Under this proposal, processing would continue as before. Therefore, there is no backwards compatibility issue.

In those cases where the system does not follow the standard to begin with (60% of the implementations surveyed), these implementations would continue to be non-standard.

4.2 Techniques and Costs for Compliance Verification

The following code can be used to verify compliance. This test will not work on implementations that do not allow CLOSEing the principal device, or on implementations that cannot re-establish ownership of the principal device once it is CLOSEd.

```
CLOSTEST  ;test CLOSE
          FOR  READ !,"test device: ",DEVICE QUIT:DEVICE=""  DO
          . WRITE "  "
          . OPEN DEVICE:1 IF '$T W "can't open" QUIT
          . USE DEVICE
          . CLOSE DEVICE
          . SET TEST=$IO=""
          . OPEN $PRINCIPAL USE $PRINCIPAL
          . WRITE $SELECT(TEST:"passed",1:"failed")
          . QUIT
          QUIT
```

4.3 Legal Considerations

Results from the he MDC sponsored poll were not statistically significant due to small sample size.

5. Closely Related Standards Activities

5.1 Other X11 Proposals Under Consideration:

X11/SC15/93-27 – Initializing Intrinsics
X11/SC12/92-26 –Value of SIO
X11/SC12/93-32 –Null Device

5.2 Other Related Standards Efforts: None.

5.3 Recommendations for Coordinating Liaison: None.