MP/M II™ Operating System Release 2.0
Application Note 01, 9/14/81

Compiled September 1981

SUPPRESSING THE MP/M™ LOADER DISPLAY

Applicable products and version numbers: MP/M II™ Release 2.0

Program: MPMLDR.COM

When the MP/M II loader reads the MPM.SYS file, it displays a load map on console #0. In some applications you might want to suppress this display.

To suppress the load map display on console #0, type the following RET instruction into the LDRBIOS.ASM file using any standard editor. The RET instruction replaces the console output code.

```
; Loader BIOS jump vector:

        ...
        jmp     conout
        ...

conout:
        ret
```

Assemble LDRBIOS.ASM to create LDRBIOS.HEX. Integrate the new LDRBIOS.HEX file into the MPMLDR.COM file according to instructions provided in the MP/M II Operating System System Guide. Then, update the system tracks of the boot disk with the new loader.

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 02, 9/14/81

SETTING AND RESETTING THE RAW CONSOLE I/O MODE

Applicable products and version numbers: MP/M II™ Release 2.0

Some application programs require raw input from the console. Raw input implies that the operating system takes no action on special characters, such as CTRL-C.

Execute the following code to place an application program into a raw console input mode.

```
MVI  C,9CH
CALL XDOS     ; get process descriptor address
LXI  D,6
DAD  D
MOV  A,M
ORI  80H      ; turn 'on' the high-order bit of first
MOV  M,A      ;  character in the process name
...
```

Execute the following code to exit the raw console input mode.

```
...

MVI  C,9CH
CALL XDOS     ; get process descriptor address
LXI  D,6
DAD  D
MOV  A,M
ANI  7FH      ; turn 'off' the high-order bit of first
MOV  M,A      ; character in the process name
...
```

Functions 3, 4, and 6 place the system into raw console input mode. All other console I/O functions reset the system to normal console input mode.

Raw console input mode can cause problems. You cannot abort a process running in raw mode because the system ignores all control characters. To abort a process, use Function 11 before using any disk I/O functions. Function 11 returns the system to normal console mode.

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 03, 9/14/81

CHANGING PRL FILE MINIMUM BUFFER SIZE REQUIREMENTS

Applicable products and version numbers: MP/M II™ Release 2.0

You might want to allocate a larger default buffer for a program such as the editor. You can change the minimum buffer size requirements for PRL files. The following procedure demonstrates how to change the minimum buffer size requirements for ED from 4k to 8k bytes.

```
0A>ddt ed.prl
[MP/M] DDT VERS 1.1
NEXT  PC
2300 0100
-s104
0104 00 00
0105 10 20
0106 .
-v2300
0044
-ied.prl
-w44
-g0
```

Bytes 4 and 5 of the PRL header record (relative to the base) contain the low- and high-order bytes for the minimum buffer size specification.

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II T.M. Operating System Release 2.0
Application Note 04, 9/14/81

ACCESSING THE INTERNAL MP/M II T.M. TOD

Applicable products and version numbers: MP/M II Release 2.0

Some application programs might require access to the internal MP/M II time and date fields to set initial values. Execute the following code sequence at the end of your MP/M T.M. XIOS system initialization procedure. Place the code at the end because the XDOS call to obtain the system data page address might cause interruptions.

```
...

MVI  C,9AH
CALL XDOS       ;obtain the system data page address
                ;*** warning ***
                ;the XDOS call enables interrupts

LXI  D,00FCH
DAD  D          ; hl -> pointer -> TOD
MOV  E,M
INX  H
MOV  D,M        ; de -> TOD

...
```

The assembly language subroutine TODCNV.ASM distributed on the MP/M II release disk converts from ASCII string representation of the time and date to MP/M II internal time and date format.

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 05, 9/14/81

DMA DISK CONTROLLERS WITH BANKED MEMORY SYSTEMS

Applicable products and Version Numbers: MP/M II™ Release 2.0

Be extra careful with bank switched memory systems that have Direct Memory Access disk controllers. Bank switching is not allowed during a transfer of data from the disk controller to a target bank.

DMA from the disk controller is obtained through common memory, then copied from common memory into the user buffer that you wanted. Sectors larger than 128 bytes are placed in a common memory buffer. The specified sector is then transferred to the target buffer. This is a reasonable technique in systems where deblocking is required.

Use the following procedure if DMA is to occur directly into the user buffer bypassing common memory. Set a DMA active flag to true before each DMA operation. Reset the flag following each operation.

```
...

MVI  A,FFH
STA  DMACTVE

;  initiate DMA operation

;  perform flag wait or poll for operation complete

XRA  A
STA  DMACTVE
...
```

Place the following code sequence in the XIOS select memory procedure to ensure that the bank cannot be switched during a DMA operation:

```
SELMEMORY:
 ...

LDA  DMACTVE
ORA  A
JZ   OKTOSWITCH      ; jump if not in DMA operation
 ; Next, the bank to be switched can be
 ; compared with the current bank.  If
 ; it matches, the DMA operation will not be affected.
JZ   OKTOSWITCH      ; no bank change required
```

```
        ; A  new  bank is  specified and a DMA operation is  in
        ;  progress.  A busy wait must now be performed to wait
        ;  until the DMA operation is complete.

        ; *** warning ***

        ; The selmemory call is made from inside the dispatcher
        ;   therefore interrupts are disabled and nothing  must
        ;   be done that could force a dispatch.

BUSYWAIT:
IN    DMASTATUSPORT   ; This is a "BUSY-WAIT" !
ANI   DMADONE
JZ    BUSYWAIT        ; loop until the DMA is complete
```

Place the following code into the remaining select memory procedure.

```
OKTOSWITCH:
   ...

   ...

RET
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 06, 9/14/81

USING THE SEND CLI COMMAND XDOS FUNCTION

Applicable products and version numbers: MP/M II™ Release 2.0

Use of the Send CLI Command XDOS Function can effectively implement a menu driven application program. The following steps outline use of the SEND CLI XDOS Function.

1) Change the priority of the calling process so that it is higher (actually a lower value) than the TMP.

2) Obtain the console number of the calling process.

3) Assign the console to the Command Line Interpreter.

4) Issue the send CLI command function call.

5) Issue an ATTACH console function to get the console back after the initiated process has terminated.

6) Restore the priority of the calling process to its original value (usually 200).

Segments of a menu driven program named MENU appear in the following example.

```
;
; XDOS Function Equate Table
;
setpriority     equ     145
attachconsole   equ     146
assignconsole   equ     149
sendCLIcommand  equ     150
getconsole      equ     153

MENU:
        ...

        mvi     e,190
        mvi     c,setpriority
        call    BDOS            ;set priority to 190
        mvi     c,getconsole
        call    BDOS            ;get console # in A reg
        sta     AssignPB        ;fill in
        sta     CLIcommand+1    ;       console fields
        lxi     d,AssignPB
        mvi     c,assignconsole
```

```
            call    BDOS            ;assign console to CLI
            inr     a
            jz      cannotassign    ;assign failed
            lxi     d,CLIcommand
            mvi     c,sendCLIcommand
            call    BDOS            ;send CLI command
            mvi     c,attachconsole
            call    BDOS            ;attach console
            mvi     e,200
            mvi     c,setpriority
            call    BDOS            ;set priority back to 200
            ...

AssignPB:
            db      $-$             ;console number
            db      'cli     '      ;name (cli is lower case)
            db      0
            ...

CLIcommand:
            db      0               ;default disk / user code
            db      $-$             ;console number
            db       this is an ASCII string terminated with a
                     null that is exactly as you would run the
                     program from the console. e.g.
                     'PIP LST:=MYPROG.LST[PT8]',0
            ...
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 07, 9/14/81

CREATING A SUBMIT FILE FROM AN APPLICATIONS PROGRAM

Applicable products and version numbers: MP/M II™ Release 2.0

The following procedure shows you how to create a submit file from an applications program and force its execution. The procedure to terminate a submit file job is included.

1) Obtain the temporary file drive from the system data page.

2) Obtain the console number at which the program is executing.

3) Create the n.SUB file. Use n to specify the console number.

4) Set the appropriate submit flag in the array to on. The array is contained in the system data page.

```
;
; BDOS / XDOS Function Equate Table
;
closefile       equ     16
searchfirst     equ     17
deletefile      equ     19
makefile        equ     22
getconsole      equ     153
getsysdatadr    equ     154
subflgofst      equ     128
    ...
    ...
   mvi     c,getsysdatadr
   call    BDOS
   lxi     d,196           ; temp file drive offset
   dad     d
   mov     a,m
   sta     FCB
   mvi     c,getconsole
   call    BDOS
   sta     console
   adi     '0'
   sta     FCB+2           ; put console # in fname
   lxi     d,FCB
   mvi     c,searchfirst
   call    BDOS            ; see if file there
```

```
        inr     a
        jz      nofile
        lxi     d,FCB
        mvi     c,deletefile
        call    BDOS            ; delete old version first
                nofile:
        lxi     d,FCB
        mvi     c,makefile
        call    BDOS            ; make the $n$.SUB file
         ...
;
; Now, write the records into the the $n$.SUB file as
;  follows:
;               -one line of the submit file per record
;
;               -last record first (i.e. in reverse order
;                 that they are to be executed
;
;               -each record in the following form:
;
;               [CNT][ASCII command line][NULL]
;                  where:  CNT = # chrs in cmnd ln, 1 byte
;                          ASCII command line <= 125 chrs
;                          NULL = zero, 1 byte
;
         ...
         ...
        lxi     d,FCB
        mvi     c,closefile
        call    BDOS            ; close the $n$.SUB; file
        mvi     c,getsysdatadr
        call    BDOS            ; get system data page adr
        lda     console         ; retrieve the saved con #
        adi     subflgofst      ; add offset to base of
        mov     e,a             ;  submit flag array
        mvi     d,0
        dad     d               ; DE = .submitflag(console);
        mvi     m,0ffh          ; set 'on' submit flag
         ...
; terminate the program
        ...
FCB:
        DB      1               ; disk drive, usually A:
        DB      '$n$     '      ; filename
        DB      'SUB'           ; filetype
        DB      0               ; file extent
        DS      20              ;  remainder of FCB
console:
        ds      1               ; temp loc for console #
         ...
```

Terminate the operation of a submit job by zeroing a submit flag located in the SYSTEM DATA PAGE region of memory. To locate and zero the submit flag for a console use the following code procedure.

```
;
;  XDOS Function Equate Table
;
getconsole      equ     153
getsysdatadr    equ     154
subflgofst      equ     128
         ...
         ...
        mvi     c,getconsole
        call    BDOS            ; get console #
        push    psw             ; save console #
        mvi     c,getsysdatadr
        call    BDOS            ; get system data page adr
        pop     psw             ; restore console #
        adi     subflgofst
        mov     l,a             ; hl = address of sub. flag
        mvi     m,0             ; zero submit flag
         ...
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 08, 9/14/81

FILE SHARING

Applicable products and version numbers: MP/M II™ Release 2.0

Multiple users can share files using the MP/M II file system. An applications program such as Wordstar® requires that files be open while the program is running. Multiple users of the application will need to share the open files. Usually under MP/M II, sharing of files causes problems if the applications program is not structured to open files in Read-Only mode. The default mode for the open function is locked mode which prevents the sharing of files. Files are opened in locked mode for earlier versions of both CP/M® and MP/M™ as well.

To enable file sharing, place all files to be shared under USER 0 on the default disk. Using the SET utility, assign the attributes SYS (System) and RO (Read-Only) to the files. The BDOS opens the file in Read-Only mode regardless of which mode the open function specified. An example is shown below.

```
OA>set wsmsgs.com [SYS,RO]
```

Licensed users are granted the right to inlude these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 09, 9/14/81

PROGRAM CONTROL OF THE CONTROL-P SWITCH

Applicable products and version numbers: MP/M II™ Release 2.0

An applications program might need to echo console I/O to the printer while under program control. Use the following procedures to set and clear the CTRL-P flags. The array of flags is located at the SYSTEM DATA PAGE address + 126.

Setting CTRL-P Flag

```
        mvi     c,9ah       ; Get System Data Page address
        call    BDOS
        lxi     d,126
        dad     d           ; add 126 to Sys. Data Page addr.
        mov     e,m
        inx     h
        mov     d,m         ; DE = addr. of CTRL-P array
        push    d
        mvi     c,0a4h      ; Get List Number
        call    BDOS
        mov     e,a
        mvi     d,0
        pop     h
        dad     d
        mvi     m,0ffh      ; set CTRL-P flag
        ...                 ; cons. I/O is echoed from now on
```

Clearing CTRL-P Flag

```
        mvi     c,9ah       ; Get System Data Page address
        call    BDOS
        lxi     d,126
        dad     d           ; add 126 to Sys. Data Page addr.
        mov     e,m
        inx     h
        mov     d,m         ; DE = addr. of CTRL-P array
        push    d
        mvi     c,0a4h      ; Get List Number
        call    BDOS
        mov     e,a
        mvi     d,0
        pop     h
        dad     d
        mvi     m,0         ; reset CTRL-P flag
        ...                 ; console I/O echo is now off
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 10, 9/14/81

COLD BOOT STARTUP

Applicable products and version numbers: MP/M II™ Release 2.0

MP/M II can execute one command upon cold boot. However, the system can execute any number of commands upon cold boot if the initial command is SUBMIT.

To execute the Startup command place the Startup command singularly into a file using standard command format. Name this file n.SUP where n is the console number that executes the command. The n.SUP file resides on the system drive at the desired USER number or at USER 0 with a SYS (SYSTEM) attribute. Examples are shown below.

Startup file:	0.SUP
Command in the Startup file:	SUBMIT START0

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 11, 9/14/81

SUBMIT ENHANCEMENTS

Applicable products and version numbers: MP/M II™ Release 2.0

Enhancements to SUBMIT include the following new features and facilities.

INCREASED n.SUB FILE SIZE: SUBMIT file size is now unlimited. The n.SUB file originally was limited to one extent, 128 lines.

CHANGING THE USER NUMBER: To change the current USER number in SUBMIT, include the USER command in the SUBMIT file.

INCLUDE FILES: An include file is a standard SUBMIT file subject to all SUBMIT rules and features. Format for the INCLUDE command is demonstrated below.

```
$INCLUDE filename parm1 parm2 parm3 ...
```

The filename in the INCLUDE command must have the filetype .SUB to indicate a SUBMIT file and parameters are standard SUBMIT parameters. An INCLUDE file can nest up to four SUBMITs in a SUBMIT command.

EMBEDDED CONTROL CHARACTERS: Control characters can be embedded in a SUBMIT file by preceding the capitalized character with an ASCII up arrow ^. For example, type ^X to embed a CTRL-X. Embedded control characters are not interpreted by MP/M II, but can be of use to programs that SUBMIT executes.

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 12, 9/14/81

SPOOL UTILITY MODIFICATIONS

Applicable products and version numbers: MP/M II™ Release 2.0

SPOOL can return an error message if the file to be spooled is not found. To utilize this modification, SPOOL is divided into a transient portion (SPOOL.PRL) and a resident portion. The transient portion parses the command tail, opens the file, passes the file to the spool queue (named SPOOLQ), and displays an error message if the open sequence on the file fails. Then, the transient portion ends itself.

Issue a SPOOLQ command if you do not want to use a memory segment to spool a file. Error messages are not returned, however. Sample commands to spool a file are shown below.

```
SPOOL file1.typ,file2.typ ...
```

The SPOOL process passes the command tail, checks for errors, and sends the file to the spool queue (SPOOLQ).

```
SPOOLQ file1.typ,file2.typ ...
```

The command tail is sent to the spool queue (SPOOLQ) bypassing error checking or error reporting.

The SPOOL utility sets its priority to 201. Most processes execute ahead of the SPOOLER. To change the SPOOLER priority, the SPOOL.BRS file is modified. Make sure you have a back-up copy of SPOOL.BRS before using RDT to make the following changes.

```
A>rdt spool.brs
   ...
-s3b5
03B5 C9 c8
03B6 00 .
-ispool.brs
-w14
-g0
A>gensys
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 13, 9/14/81

RECORD LOCKING/UNLOCKING

Applicable products and version numbers: MP/M II™ Release 2.0

Record locking/unlocking allows multiple processes to share access of one file. Files are opened in the UNLOCKED mode. A record locked by one process can only be read by a different process, however, a locked record can be modified by the initial process. Avoid reading locked records to prevent reading data that is being updated. To avoid reading locked records let the process try to lock the record. If the attempt fails, do not read the record. The following code segment demonstrates how to lock records.

```
        mvi     c,2ch       ;set multi-sector cnt.
        mvi     e,#         ;# = num. of sectors
        call    bdos        ;1<= # <=16
        mvi     c,2ah       ;lock record
        lxi     d,fcb       ;record to be locked
        call    bdos
```

The following code segment demonstrates how to unlock records.

```
        mvi     c,2ch       ;set multi-sector cnt.
        mvi     e,#         ;# = num. of sectors
        call    bdos        ;1<= # <=16
        mvi     c,2bh       ;unlock record
        lxi     d,fcb       ;record to be unlocked
        call    bdos
         ...
fcb:
        db      0,'DATA',A0H,20H,20H,20H,'DAT',0
        ds      20
        db      10,0,0      ;begining at record 10
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II[T.M.] Operating System Release 2.0
Application Note 14, 9/14/81

GENSYS ENHANCEMENTS

Applicable products and version numbers: MP/M II[T.M.] Release 2.0

Enhancements to GENSYS include the following new features and facilities.

AUTOMATIC RESIDENT SYSTEM PROCESS INCLUSION FACILITY: The GENSYS automatic system generation facility can be modified to include all default disk .RSP files. Type GENSYS $AR to include the .RSP files automatically. The R option must be used in conjunction with the A option. Change the filetypes for files that you want to exclude from GENSYS.

ERROR RECOVERY: If an error is encountered running in automatic mode ($A option), GENSYS restarts in manual mode.

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

MP/M II™ Operating System Release 2.0
Application Note 15, 12/1/81

CHANGING THE PRIORITY OF SPOOL.PRL

Applicable products and version numbers: MP/M II™ Release 2.0

The SPOOL utility sets its priority to 201, therefore, most other processes execute ahead of the SPOOLER. Modify the SPOOL.PRL file to change the SPOOLER priority. If your product serial number is between 4-000-00001 and 4-000-00464, install MP/M II Patch 11 before changing the SPOOL.PRL default priority.

Make sure you have a back-up copy of SPOOL.PRL before using DDT to make the following changes.

```
A>ren spool.sav=spool.prl
A>ddt spool.sav
[MP/M II] DDT VERS 2.0
NEXT PC
0980 0100
-s269
0269 C9 c8
026A CD .
-ispool.prl
-w11
-g0

A>
```

Licensed users are granted the right to include these enhancements in MP/M II Release 2.0 software.

RMAC™ 1.1 Application Note 01, 6/19/81

INCLUDING LOCAL SYMBOLS

Products: PL/I-80™ 1.3 and RMAC™ LINK-80™ LIB-80™

This application note shows how to patch RMAC so that local symbols as well as publics are put into the object file produced by RMAC, and hence into the SYM file produced by LINK.

```
A>ddt rmac.com
DDT VERS 2.2
NEXT  PC
3600 0100
-s1167
1167 08 18
1168 32 .
-g0
A>save 53 rmac.com
A>
```

Licensed users are granted the right to include these changes in RMAC software. LINK-80, LIB-80, PL/I-80 and RMAC are trademarks of Digital Research.