

VM/370 CP Unit-Record Devices

Moving Data Off of VM

From within any virtual machine, you may direct your virtual punch and/or printer (output devices) to any other virtual machine, or to the SYSTEM queues. The CP (VM/370 Control Program) command to manipulate the destination is the SPOOL command. Output (and Input) is assigned a "Class". For more information about Unit Record Device Classes - please refer to the VM/370 CP Operator Manual.

As installed, the SixPack configuration defines the system unit record devices as:

General User

```
q ur RDR 00C DRAINED SYSTEM PUN 00D DRAINED SYSTEM CLASS = P SEP PRT 00E DRAINED SYSTEM  
CLASS = A SEP PRT 00F DRAINED SYSTEM CLASS = A SEP Ready; T=0.01/0.01 17:48:15
```

Therefore, you must spool your devices accordingly ensure that your output will be routed correctly. The following commands are "case insensitive" but the lower-case characters may be omitted, and the desired effect will still be attained.

1. `cp SPOOL PUN TO SYSTEM CLASS P`
2. `cp SPOOL PRT TO SYSTEM CLASS A`

When you punch or print data, they will now immediately flow out of the virtual VM/370 environment onto the HERCULES folders. There are two primary methods to perform a "punch" operation:

1. Using the PUNCH command
2. Using the DISK DUMP command.

PUNCH is best because the output will be readable but only works for original data which does not exceed 80 columns in width. For records longer than 80 characters will require use of the DISK DUMP command.

The downside of DISK DUMP is that it "compresses" the data so it won't easily be usable outside of CMS. May might consider using the printer for content which is longer than 80 columns, but less than the width of the virtual printer's carriage width (usually, 120 characters).

Examples:

- `PUNCH PROFILE EXEC A`
- `DISK DUMP MY MODULE A`
- `PRINT USER LISTING A`

To perform any operation on multiple files, the Listfile command can be used to make an EXEC that you can use to issue a command on all the files in the list.

1. `cp SPOOL PUN Cont` (spool the punch 'continuously'; which keeps the punch from closing after each file is punched)
2. Create the CMS EXEC file using the LISTFILE command
 - `LISTFILE * EXEC A (EXEC`
3. Use the CMS EXEC file to issue the command for each of the files

- CMS PUNCH
- 4. Close the punch
 - cp CLOSE PUN

Other examples:

1. Listfile * LISTING A (EXEC
CMS PRINT
cp CLOSE PUN
2. Listfile * * A (EXEC
CMS DISK DUMP
cp CLOSE PUN

NOTES:

As you can see above, the System has the I/O Devices in a DRAINED state. This means that the devices are not enabled for any activity. Unless you START the devices, your files will just sit there in the queue until the OPERATOR turns on the system card punch. **And YOU are the operator!**

The OPERATOR privilege is assigned to the user MAINT and the user OPERATOR.

The HERCULES operator console is automatically logged on as OPERATOR so since the HERCULES console needs to determine whether a command is destined for HERCULES or to the GUEST operating system (OPERATOR in this case). Hercules uses the forward slash "/" character to make this distinction:

So go to the OPERATOR console and type:

- /cp START PUN
- /cp START PRT

These will start all the printers and punches. You can also issue the CP START command using the Device Address instead of the generic token of PRT or PUN.

If you have a punch at 00D and another at 01D:

- cp START 01D

will start the ONE device. You may want to do this so you have one that outputs in ASCII and one in binary (or EBCDIC since there is no translation).

This will start the system punch and printers. If anything is queued for them, the files will be output to your computer as specified in the HERCULES configuration file. For example:

- 000D 3525 /herc370/vm2/punch00d.txt ascii
- 000E 1403 /herc370/vm2/print00e.txt crlf

Moving Data On to VM

1. Add a record to the start of the file you want to transfer showing what

userid it should be sent to:

- USERID CMSUSER

1. Attach the file you want to import to your system's card reader.

Type this on the Hercules console:

- devinit 000c myfile.txt (note: no preceding "/" (slash) mark) *(This assumes the file is in the same path from where you started. If not, give the entire path in the above command.*
- Response: HHCPN098I Device 000C initialized

The file should be delivered to the user you specified. There will be a message about this on the OPERATOR console:

- RDR 00C INPUT FOR CMSUSER FILE = 0051

as well as on the console of the receiving user

- RDR FILE 0051 HAS BEEN READ

If the file is NOT sent, the reader is probably drained. You will need to start it up!

On the operator console, type:

- cp START RDR
- CP Responds: RDR 00C STARTED SYSTEM

1. On the receiving user, read in the file. If you have more than one file,

ORDER the one you want to the top. Example:

General User

```
q r all ORIGINID FILE CLASS RECDS CPY HOLD DATE TIME NAME TYPE DIST CMSUSER 0053 A RDR
000008 01 NONE 08/26 05:17:17 CMSUSER 0051 A RDR 000008 01 NONE 08/26 05:13:20 R; order rdr
51 0001 FILE ORDERED R; read my file a R;
```

Notice that the files have no name. You could have given it one by adding an additional card after the USERID card in step 1:

- USERID CMSUSER
- :READ MY FILE A

I've seen evidence that you can shorten the first record to ID CMSUSER but I haven't tested this. You can probably find a way to automate this on your system. This page has some samples: http://open360.copyleft.de/VM370/SysOpas_little_Helpers.html Or another way is to create a small file with just the header information in it:

```
USERID CMSUSER
:READ MY FILE A
```

```
•
•
•
```

and save it as something like `usercms.txt`. Then issue the `devinit` statement:

- `devinit 00c usercms.txt myfile.txt ascii multifile`

The “multifile” part means the files should be read as if they were one.

To transfer EBCDIC files this way (or binary files), you'll need to carefully create your header file in EBCDIC with exactly 80 bytes for each line and no line-end characters and name it something like `usercms.ebcdic`. Then issue the `devinit` statement like this:

- `devinit 00c usercms.ebcdic mybinary.data ebcdic multifile`

VMARC is a file compression/decompression tool used to pack many of the VM/CMS downloads you may find on the internet. It is installed on the four-pack system on the MAINT 19E disk as VMARC370 MODULE.

After all this experience, here's a sample PROFILE EXEC file for OPERATOR to make our lives easier:

```
&CONTROL OFF
CP SET PF12 RECALL
CP SET EMSG ON
CP TERM MODE VM
CP ENABLE ALL
CP START PUNCH
CP START PRINT
CP START READER
```

If you don't set `TERM MODE VM`, the OPERATOR console seems to only accept CP commands. Another thing is that OPERATOR won't IPL CMS automatically, so you'll need to do that on startup, then let it's PROFILE EXEC run.

Reading Tapes

1. Place the tape image files into the `vm1` directory (with the other VM system files).
2. Attach a tape image file to the system.

From the Hercules console, type (for example):

- `devinit 480 vmgames.aws`

1. Attach the tape drive to a user.
2. From the OPERATOR console attach CP System Unit 480 to userid MAINT as 181 using the command:

- `cp ATTach 480 MAINT 181`

1. Scan the tape

- From MAINT, type `VMFPLC2 SCAN`

This will list files before the next tape mark. Repeat until you get an empty list.

1. Rewind the tape

- REWind 181

1. Create a temp disk to hold the data (optional)

General User

```
cp DEFINE T3350 as 123 cyl 10 DASD 123 DEFINED 010 CYL R; FORMAT 123 B DMSFOR603R FORMAT
WILL ERASE ALL FILES ON DISK 'B(123)'. DO YOU WISH TO CONTINUE? (YES|NO): yes DMSFOR605R
ENTER DISK LABEL: TMPDSK FORMATTING DISK 'B'. '10' CYLINDERS FORMATTED ON 'B(123)'. R;
```

You may want to create a quick TDISK EXEC to do this (for future use):

```
&CONTROL OFF
&IF &1 NE ? -SKIPHELP
&TYPE FORMAT: TDISK [cuu [mode [size]]]
&TYPE DEFAULT: TDISK 123 B 10
&EXIT 1
-SKIPHELP
&IF .&1 EQ . &1 = 123
&IF .&2 EQ . &2 = B
&IF .&3 EQ . &3 = 10
CP DEFINE T3350 AS &1 CYL &3
&IF &RETCode NE 0 &EXIT &RETCode
&STACK YES
&STACK TMPDSK
FORMAT &1 &2
&EXIT &RETCode
```

Note: You may need to change T3350 to T3330, depending on your DASD type.

1. Load the tape files: VPFPLC2 LOAD * * B
2. Copy files to your A-disk COPYfile * * B = = A (NEWDate

I've found some tape files load with blank dates. This could be a Y2K issue. In any case, the NEWDATE option fixes this. Copying the files also seems to fix some other problems I've seen with files loaded from tapes. Modules that wouldn't run now run after copying. Files that could be TYPEd but not edited can now be edited.

1. Detach the tape drive cp DETach 181

NOTE: There are two utilities for dumping CMS files to and from tape VMFPLC2 and TAPE. If you receive errors, the files may be loaded in TAPE format instead of VMFPLC2

From:

<https://codex.sjzoppi.com/> - Wizard of Odd

Permanent link:

https://codex.sjzoppi.com/ibm360-370:vm_370_cp_system_unit-record_devices

Last update: 2021/01/01 19:06



