Operations

Overview

This document describes procedures for CYBIS running on NOS 2.8.7 on the DtCyber emulator by Paul Koning which is structurally different than Tom Hunter's dtCyber.

Paul's implementation separates the display logic from the running emulator so NOS can be operated “headless” or by other users remotely.

Procedures

1 - Local and remote control of DtCyber

DtCyber is controlled via two X-based applications, called “dd60” (the Cyber console emulation) and “dtoper” (the DtCyber operator control interface).

If you’re using local access, i.e., you’re at the X display of the machine where DtCyber runs, you would start DtCyber with the “dtycyber.sh” script. Among other things, that script will start dd60 and dtoper.

If you’re using remote access (you’re coming in via SSH) do not use dtycyber.sh but instead use “dtycyber-remote.sh”. That script also does dtycyber startup handling but doesn’t run the control applications.

For remote access, you’ll need to specify SSH tunneling on the SSH command line, or in the tunneling settings if you’re using a GUI-based SSH utility such as PuTTY. I use this command:

```
ssh -L 5104:localhost:5004 -L 5105:localhost:5005 
  -L 8105:localhost:8005 paul@cyberserv.org
```

The tunnel entries for 5004 and 8005 are for the classic and ASCII pterm respectively. 5005 is the port for the pterm at station 0-1; that is occasionally needed because 0-1 is a “special station”. 5006 is the port for dtoper; 5007 is the port for dd60. I map all these to different port numbers locally so I don’t get conflicts if I run dtycyber on my own machine while at the same time tunneling to Cyber1.

1.1 - Starting and Stopping dtOper

You can start dtoper and stop at any time. It’s ok to have more than one copy running.

For local access, simply start it in the background:

```
./dtoper & <enter>
```
For remote access, specify the TCP port number to access. That will have to be a port number you
gave to SSH for tunneling to the machine on which DtCyber is running. My convention is to use the
standard port number plus 100; for dtoper that means 5106:

```
./dtoper 5106 & <enter>
```

To exit dtoper, enter the command:

```
end. <enter>
```

dtoper will exit. Note that this doesn’t affect DtCyber.

### 1.2 - Starting and Stopping dd69

You can start dd60 and stop at any time. It’s ok to have more than one copy running.

For local access, simply start it in the background:

```
./dd60 & <enter>
```

For remote access, specify the refresh interval, and the TCP port number to access.

The refresh rate controls how quickly dd60 updates. The default is every 60 milliseconds, which
makes it nicely responsive for local use, but that isn’t a good idea for WAN access. I usually pick 3-5
seconds for the refresh rate.

The TCP port number will have to be a port number you gave to SSH for tunneling to the machine on
which DtCyber is running. My convention is to use the standard port number plus 100; for dd60 that
means 5107:

```
./dd60 4 5107 & <enter>
```

To exit dd60, enter ALT-Z (Command-Z on the Mac). dd60 will exit. Note that this doesn’t affect
DtCyber.

### 2 - Startup

Normally you will want to start DtCyber with a Level 0 Deadstart, and autostart everything. There are
two startup scripts: dtcyber.sh (if you’re local, i.e., sitting at the X display of the machine where
DtCyber runs) and dtcyber-remote.sh (if you’re remote, i.e., connected via SSH). Either way, if you
invoke the script without arguments, it will start DtCyber in autostart mode, which will start DtCyber in autostart mode, which will start
DtCyber in autostart mode, which will start DtCyber in autostart mode, which will start
DtCyber in autostart mode, which will start DtCyber in autostart mode, which will start
DtCyber in autostart mode, which will start DtCyber in autostart mode, which will start
DtCyber in autostart mode, which will start DtCyber in autostart mode, which will start
DtCyber in autostart mode, which will start DtCyber in autostart mode, which will start
DtCyber

```
./dtcyber.sh
```

After a few seconds, two windows appear, one of these is the Cyber console. (If you’re using dtcyber-
remote.sh, the script just exits once dtcyber has started, and you then have to start dd60 and/or
dtoper manually, at your local system.)

Click on the dd60 window to see the console display so you can see what's going on. When it first appears, NOS is in the middle of the deadstart, which shows a display like this:

![Deadstart status](image)

When it is done, you'll see the A and B displays on the console, and lots of things will start to run. After all the activity stops, PLATO is running; the display will look roughly like this:

![System status](image)

### 3 - Shutdown

First back out PLATO (1 3 from author mode). Then, at the console:
**MASTOR K-DISPLAY**

**SUBMIT FILE** = CYBIS

**SECUR** = OFF

```
K.STOP
```

```
k,mas1. <enter>
k.stop <enter>
[ (to erase the "k.")
```

```
ab.<enter>
```
idle,nam.<enter>

The word “UNLOCK” appears in bold on the top line of the display.
Important! Wait for “check point complete” message (right side of the display, bottom line, “31 SYS”).

The word “STEP” appears in bold on the top line of the display.

Now go to the DtCyber operator window. At that window, enter:
UNLOCK.

It appears as if the word “UNLOCKED” appears in the top line, on the right hand side.

shutdowm. <enter>
The message “DtCyber shut down.” appears at the bottom, indicating that DtCyber has exited.

This will close the operator window.

### 3.1 - Shutdown Failures

Sometimes, if a PPU program is misbehaving, the “check point system” step does not complete normally. (Give it 30 seconds or so, but it should complete fairly quickly. Completion is indicated by the fact that the B display – right side of the standard display – has only one line on it, for “SYS” and that line has a message “checkpoint complete.”) If “check point system” does not complete, do the following.

**Important:** do this right away, you get only one chance to do this and you wouldn’t want to wait till later and forget...
Go to the DtCyber operator window. At that window, enter:

```
unlock. <enter>
shutdown. <enter>
end. <enter>
```

Then go to the console window and use the Deadstart Options display to do a level 3 deadstart:

```
INITIAL OPTIONS
A OS LOAD AUTOMATIC
O OS LOAD WITH INTERVENTION
U UTILITIES
H HELP
(CR) OS LOAD AUTOMATIC

CONTROL DATA PROPRIETARY PRODUCT
COPYRIGHT CONTROL DATA 1980.
CIP- V11
739

O

OPERATOR INTERVENTION
(NOS OR NOS/BE DEADSTART)
A - CONTINUE OS LOAD
V - HARDWARE VERIFICATION
M - HARDWARE RECONFIGURATION
P - DEADSTART PANEL PARAMETERS
S - SELECT DEADSTART DEVICE

(CR) - CONTINUE OS LOAD
(ES) - PREVIOUS DISPLAY
```

P
The Level 3 deadstart options display will appear. Enter:

```
LEVEL 3 OPTIONS

ABORT.       NO
ABORT.B.     NO
AUTLOAD.     YES
GRENADE.     NO
AUTO.        YES
```

```
I=3 <enter> <enter>
```

```
abort. <enter>
go. <enter>
```

You should get a display like this. Don’t be mislead by the “recovery aborted” message, the right thing has happened at this point.
Now go to the DtCyber operator window. At that window, enter:

```
unlock. <enter>
shutdown. <enter>
end. <enter>
```

Finally, restart DtCyber using the autostart (which is a level 0 deadstart with ECS initialization).

### 4 - Crashes

If PLATO (CYBIS) dies, you will get the dreaded “plato off” display on all the terminals. A memory dump will be generated and sent to the “printer”. Mastor will automatically restart things.

If Conden dies, you may be able to restart it from the menu from 1 at author mode.

If Framat dies, go to the console window and enter:

```
drop,pla1. <enter>
```

(which will produce “plato off”) and things should restart.

If Mastor dies, shut down the system and restart, as follows:

On the console, enter:

```
CDC Terminal
IDLE,NAM. UNLOCK. CHECK POINT SYSTEM.
```

```
...........
**Important!**
.
   Wait for “check point complete” message (at the B display line for “SYS”)
............
```
STEP.

Now go to the DtCyber operator window. At that window, enter:

CDC Terminal
UNLOCK. SHUTDOWN. END.

Then restart DtCyber using the autostart (level 0 deadstart with ECS initialization).

4.1 - Dump File Preservation

In any of the above crash cases, wait for any line printer output to stop (BIO status on the B display is IDLE). Then enter this command into the DtCyber operator window:

```
unload,12,5. <enter>
```

You’ll get a message saying that LP was unloaded to some long file name. That’s the line printer output up to this point; save it for analysis. If DtCyber dies, hopefully we’ll have a corefile; save that for analysis. In all cases, if you do a restart, make it a level zero deadstart, with ECS initialized.

5 - The "Black Box"

Some PLATO systems had a black box (earlier, an 8-track tape, if I remember right) that would transmit a fixed message periodically to all terminals. This is useful if the system is down for some reason; that might be backups, or some planned (or unplanned) downtime of another kind.

The DtCyber equivalent is the utility program “blackbox”. You can run this when DtCyber is not running. It takes one argument, which is the message to be displayed. It accepts connections from pterms, just as DtCyber does. It will ignore any input, and send the supplied message to every connected terminal every 5 seconds.

For example:

```
./blackbox "Backups in progress, expect PLATO back at 9:00 PST"
```

If no argument is given, the default is: “PLATO is down for the moment”.

Blackbox will reply “Current message is: message” and “Enter a new message at any time, or Ctrl/D to stop blackbox”. This allows you to change the message if you need to; just type the new message and Enter. You can change it as often as needed.

The black box runs until terminated (by control/D). It takes very little CPU time, so it can easily be used while the system is busy doing backups or the like.

If you use “blackbox” be sure to stop it before restarting DtCyber.

If you forget, you should see some messages saying “can’t bind to socket” from DtCyber. If that happens, let startup complete normally, shut down DtCyber, kill blackbox, and restart. Since the NIU
(PLATO terminal interface) couldn't start up in this situation, you'll need the console PLATO terminal (`x.console.`) to shut down PLATO if you had done an autostart.

From: https://codex.retro1.org/ - Retro1.org

Permanent link: https://codex.retro1.org/cdc:nos2:cybis_op_guide_pk

Last update: 2023/08/05 15:36