**Lantronix Terminal Server on Mark 4**

A Lantronix ETS16PR Terminal Server is now installed on the Mark-4 correlator and connected (approximately) as shown in the tabulation below. The Lantronix ports are numbered 1 through 16; the associated telnet port numbers are 2000 plus the Lantronix port numbers.

A shell script, **vts0** (for view terminal server number zero), creates one or more xterms connected to Lantronix ports. To start any one or all of them, try just:

**vts0**

This script shows a tabulation similar to the one below to choose from. Or, to create just one with a known number, try:

**vts0** *n*

with *n* the port number, 1 through 16, perhaps from the tabulation below. The labels on the xterms from **vts0**, in either case, are from ivex, which might be different from the tabulation below. There are three xterm options: ordinary text, active icon made using the so-called unreadable font, nil2, or ordinary icon. Ctrl-right-click inside an xterm to get a menu to change fonts and box size. If the port is already connected to someone else, you'll see an error message for about 1 millisecond; then this new xterm will disappear. (Read fast!)

An alternative that creates a new xterm connected to one of these ports is:

**Xterm** *n*

where *n* is the port number, 1 through 16, perhaps from the tabulation below. But the xterm in this case does not get a proper label.

Another alternative is to telnet directly:

**telnet tserv0** 200*n*

(that's 2000 plus the port number), but this uses up the term that it's run in and does not get a proper label.

To shut down any of these:

**^]** (Or whatever your telnet uses for escape)

**telnet> close**

or just use your mouse and window manager to close the xterm; this shuts down the telnet also.

There are still some rough edges and some unknowns in the configuration of the Lantronix box, but this server is ready for testing.

# Lantronix Terminal Server Notes

Model:  ETS16PR
Name:  tserv0
Hardware address:  00:80:a3:23:8b:09 (at Haystack)

| Port | Connection |
|---|---|
| 1 | SU0 Port 1/Console |
| 2 | SU0 Port 2 |
| 3 | SU1 Port 1/Console |
| 4 | SU1 Port 2 |
| 5 | SU2 Port 1/Console |
| 6 | SU2 Port 2 |
| 7 | SU3 Port 1/Console |
| 8 | SU3 Port 2 |
| 9 | CU0 RS-232 Port A (Console) |
| 10 | CU0 RS-232 Port B (Debug) |
| 11 | CU1 RS-232 Port A (Console) |
| 12 | CU1 RS-232 Port B (Debug) |
| 13 | RAG's switch (notes below) |
| 14 | Through Micro488/P to HP 53131A counter |

RAG's Switch

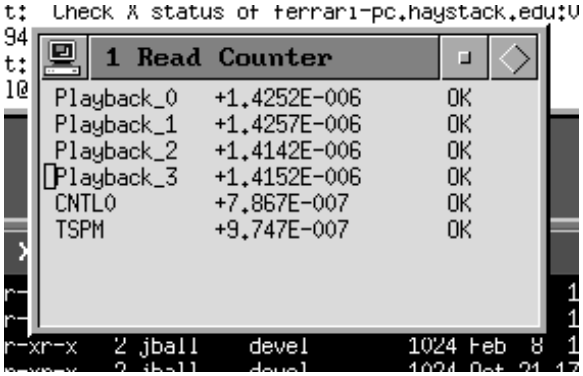
| Port | Connection |
|---|---|
| 0 | SUIM 0 SysTick |
| 1 | SUIM 1 SysTick |
| 2 | SUIM 2 SysTick |
| 3 | SUIM 3 SysTick |
| 4 | SUIM 4 SysTick |
| 5 | SUIM 5 SysTick |
| 6 | SUIM 6 SysTick |
| 7 | SUIM 7 SysTick |
| 8 | TSPM SysTick |
| 9 | CNTL0 one-second pulse |
| A | CNTL1 one-second pulse |

## Count and Counts

**Counts** is an Expect script that talks with Roger Genereux's 16-input switch and the associated HP 53131A counter. This counter is connected through an IOtech Micro488/P-901 RS-232-to-GPIB (IEEE 488) converter, and both this converter and Roger's switch are connected with RS-232 serial through the Lantronix ETS16P Terminal Server (serial-to-ethernet converter) to our LAN. **Counts** uses **telnet** to talk with this Lantronix box, alias tserv0.

Channels 0, 1, 2, 3, 8, and 9 on Roger's switch should have 1-pps signals on them, and the counter is configured by **counts** to measure time offsets with respect to a reference 1 pps from ClockUm. **Counts** uses **pivex** to read ivex to find which channels to monitor and the acceptable limits for the corresponding readings--typically 0.5 to 2.0 µs. **Counts** cycles through these channels and shows the resulting time offsets with channel numbers and an error indication if an offset is outside the prescribed range. **Counts** expects to have an xterm all to itself.

**Count** (no s) is an associated C-shell script that makes an appropriate xterm window and schedules **counts** therein. The attached figure shows an example. End **counts** with ^C or whatever you've set for interrupt. **Count** is normally started by **runall**, and **counts** is normally killed by **endall**.




For various reasons, **counts** is very slow: It takes many seconds to get started, then 11 seconds for each measurement or 55 seconds for a cycle of five measurements. The cursor rests at the beginning of the line after the last completed measurement.

The Micro488/P box is rather fragile. If **counts** doesn't work at all, the Micro488/P probably needs to be reset. Try:

1) Telnet to the Lantronix port to which the Micro488/P is connected. Currently this is tserv0 port 2014. That is:
   **telnet tserv0 2014**
   Then:
   a) Try a half dozen <Enter>s or <Return>s. If you see one or more >s, then it's (probably) fixed. If not,
   b) try several ^Js (ctrl J). See >s? If not,
   c) try ^A then <Enter> or <Return> or ^J. See >?
   Exit out of telnet with ^] and close as usual.

2) If this did not work (no >), then go to the back of the HP 53131A counter. Find the RJ-45 plug on the RJ-45-to-RS-232 adapter that is on the Micro488/P that is on the back of the HP 53131A counter. Disconnect this RJ-45 plug for ten seconds, then reconnect. Try again 1) above.

Revised: 2000 January 5, JAB