ULTRA SHEAR SOFTWARE CONFIGURATION ADDENDUM
20 JUNE 1995

This document describes changes in Ultra Shear configuration relative to the Quanterra SHEAR SOFTWARE CONFIGURATION GUIDE Revision B.

Page 4:

Aqdacserv is only required to be running if there is a "**dacon=**y" somewhere in the configuration file. There are three possible command line options:

- ***-V=n*** sets the verbosity level.
- ***-X*** will use a XVME-505 board rather than a MZ8610.
- ***-D=path*** will use a DAC480 connected to *path*.

Page 5:

There are five possible command line options:

- ***-C=name*** determines the configuration file *name*. If not specified, /r0/aqcfg will be used.
- ***-V=n*** sets the verbosity level.
- ***-M=n*** enables multiple aqsamples, this one with sequence *n*.
- ***-X=mem*** puts *mem* bytes into aqcommon.
- ***-S*** sets the Shell default display mode to SEED names, default is in the config file.
- ***-N*** sets the Shell default display mode to component/stream, default is in the config file.

Page 6:

There are four additional command line options:

- ***-X=mem*** puts starts aqsample with a *-X=mem* option.
- ***-L*** enables display of logical channels from the digitizer in addition to the physical channels.
- ***-S*** forces displaying seed names in the status monitor rather than component/stream.
- ***-N*** forces displaying component/stream names in the status monitor rather than seed names.

If the S option is not used and this is the master shell, then component/stream names will be displayed. If this is not the master shell, then the default will be the setting for the master shell, or you can use the S or N options to force the mode you want. The configuration file name is not required, and if not specified, uses /r0/aqcfg.

Page 8:

Aqwrite is no longer used. Tape operations now use the Tapeserv program. The tapeserv program has only one command line option :

- ***name*** is the *name* of the configuration file. This can be the main aquisition file, or a separate file. The format of this file is outlined later in the "Tape Configuration Section" portion of this manual, later. If not specified, then /r0/aqcfg will be used.

Page 11:

The buffer configuration is now normally stored in the main aquisition file instead of separate files in the data directory. If the ***-C=name*** command line option is not specified, /r0/aqcfg is used. In the configuration file, the parameters following the **[bcfg_C]** keyword is used if ***-q=c***, or **[bcfg_E]** is used if ***-q=e***. The format of the configuration specifications are unchanged.

Page 15:

There is one additional command line option:

- ***-m=name*** creates a data module "*name*" instead of "dacombuf". This allows multiple com links. The keyword in the configuration file for this com link will be "*name*" instead of "comlink".

  If the configuration file is not specified, then /r0/aqcfg is used.

Page 17:

There is one command line option:

- ***name*** will link to name instead of "dacombuf".

Page 18:

There is one additional command line option:

- ***-M=n*** will set the number of message log entries to *n*.

  If the configuration file is not specified, /r0/aqcfg will be used.

Page 20:

There are five additional command line options:

- ***-S=n*** sets the sequence number, it's purpose is unclear.
- ***-S*** sets the display/entry mode to Seed names, the default is in the config file.
- ***-N*** sets the display/entry mode to component/stream, the default is in the config file.
- ***-F=path*** makes retrieve read commands from the *path* and disables STP and Kermit transfers (used by fingerd).
- ***-T=path*** makes retrieve read commands from the *path* (used by tentacle).

If the configuration file is not specified, /r0/aqcfg will be used.

Page 25:

Ignore command line options listed. There is only one command line option:

- ***name*** is the *name* of the configuration file, normally the same one used by **aqsample**. If not specified, then /r0/aqcfg will be used.

Page 27:

Serv_start now resets the server when it starts. This allows it to be used from the comlink (shell command) to reset and then restart the digitizers.

Page 29:

Ignore references to TICKS, all calculations are done in microseconds. Ignore command line options shown in the book. There is only one command line parameter, this is the name of the configuration file. If not specified, then /r0/aqcfg is used. See the clock configuration section at the end of the document.

Page 31:

Lockrep has many command line options. If no command line options are used, then lockrep will write the status to the standard output path and then exit. The following options are available :

- ***-f=secs*** where *secs* is the number of seconds to set "lockable seconds in hold", whatever that is.
- ***-l*** will cause lockrep to stay in a loop, and it will report status whenever there is a change.
- ***-p*** will enable the phase lock function.
- ***-x*** will disable the phase lock function
- ***-d=mins*** will set the external delay to *mins* minutes.
- ***-g=opt*** will send the *opt* string to the gps clock. The most common string is "2" which causes a status dump.
- ***-n=ms*** will set the server timemark noise filter to *ms* milliseconds.
- ***-j=mins*** will set the server timemark jump value to *min* minutes, zero to disable the jump filter.

Lockrep is normally started after the clock has been started with the ***-l*** option to provide periodic reports :

```
clock <command line options> &
sleep -s 5
lockrep -l <>>/nil >-/pipe/log.lr &
```

Additional processes are servtest, bufwait, loadiir, servwait, servmsg, clockwait, tapestat, chart, rcman, two$qvi, sockserv, nmxp, rddso, tentacle, fingerd, and sysmon.

Servtest has been modified significantly since revision B of the documentation and is beyond the scope of this document.

Bufwait is run after starting a buffer server to wait for it to initialize. Bufwait has as it's parameters the name of the buffer service module, and the maximum number of seconds to wait. A typical use would be :

```
shell "bufserv -d=/h0/DAT -c=bcfg_C -q=c -s=1000 -l=3 ^2000 >>-/pipe/log.B
sleep -s 3
bufwait bufsvcq_c 60
```

Loadiir has as it's command line parameter the name of a IIR text data file. It will create a data module called "IIRDATA". Loadiir should be one of the first processes started, a typical command line would be :

```
loadiir /h0/shear/filters/iirfilters <>/nil >>-/pipe/log.loadiir
```

Servwait has no parameters and should be started after starting server. Servwait will finish when server has created a new "SERVDATA" data module. Typically :

```
server </nil >/niil >>-/pipe/log.server ^2500 &
sleep -s 1
servwait
```

Servmsg has as it's first parameter the digitizer channel, and the rest of the command line is the message to send.

Chart is used to provide helicorder-like plots on selected printers. It's command line parameter is the name of the configuration file (see chart preliminary documentation) and if not specified, defaults to /r0/aqcfg. Typically :

```
chart <>/nil >>-/pipe/log.chart &
```

Rcman creates a pipe that dacommo programs write to when they receive a range of commands from the a DP. These commands such functions as calibrations, changing recording status, and processing selected OS9 shell commands. The following command line options are available :

- ***-v=n*** where *n* is the verbosity level. Normally 1 is a good value unless you are trying to debug remote commands from the DP.
- ***-s*** will modify old Martin-Marietta style calibration commands so that the duration is expressed in seconds, rather than cycles or other units.
- ***-l*** will lockout calibrations during event detections.

A typical command line is :

```
rcman -v=1 -s >>-/pipe/log.rcman <>/nil &
```

two$qvi must be loaded if you have a 9 or more channel Q380 type system.

clockwait should be run in foreground after you have started lockrep, but before starting aqsample. Clockwait waits until the clock quality reaches at least 3 before exiting. An optional command line parameter a timeout value in seconds, after which it will exit regardless of the clock quality. A typical command line is :

```
clockwait 500 >-/pipe/log.clockwait <>>/nil
```

tapestat reports on the status of each of the tape drives if tapeserv is running. This command was created to allow tape manipulation over the com link. If no command line perameters are present, it simply reports the status. The tape it is currently writing on is indicated as the "active" drive. If command line parameters are used, there must be two of them. The first is the command letter, and the second is the tape path name (such as mt0). The following commads are allowed :

- ***C*** - change to the tape specified. If this is the active tape, then this indicates a tape change operation.
- ***E*** - erase volume header, this can only be done on an inactive tape.
- ***F*** - flush logs to tape, this can only be done on the active tape.
- ***H*** - hide tape from tapeserv, this can only be done on an inactive tape. This closes the path to the tape, so that you can use other tape utilties on that drive.
- ***L*** - load and analyze new tape, this can only be done on an inactive tape.
- ***O*** - overwrite confirmation, if you do a tape change to a tape that has previously recorded data on it, you will need to send this command when you see the message from tapeserv (FROM TAPE:).
- ***R*** - rewind tape, this can only be done on an inactive tape.
- ***S*** - show tape to tapeserv, this can only be done on an inactive tape. This opens the path to the tape, so that tapeserv can monitor it's activity.
- ***T*** - retension tape, this can only be done on an inactive tape, and on a drive that supports retension.
- ***U*** - unload tape, on DAT drives this will eject the tape.

As an example, to do a tape change on mt0 :

```
tapestat c mt0
```

Sockserv is a server process that provides TCP/IP communications through a data module and event. The first command line parameter is the server number (1-n) which is used in the NXMP configuration section. The optional second command line parameter of "close" will close a open socket if it exists. The close option is normally used after the client and the Sockserv have been killed. Sockservs are started as background processes before any application that requires it. Example :

```
sockserv 1 <>>>/nil &
```

NMXP is an application that sends data out through a TCP/IP socket (using sockserv) once per second. This data is used by systems that are part of the Taiwan project for real-time motion detection. Although this application may not be of interest to anyone else, it is an example of obtaining near-real time output.

Rddso is an application that sends 100Hz data through a serial port in 8 byte packets. This is another example of real-time output.

Tentacle and **fingerd** are described in separate documents. **Sysmon** is described later.

Page 53:

The structure of the config file has changed significantly. Each section of the config file has a specific starting string enclosed in square brackets. The "headers" must start in the first column. Blank lines, and those that do not start with a letter are ignored in the configuration file. There is no terminator (as in the old "endcfg"), the current section stops when either a new section starts, or end of file is reached. The configuration file is used by other programs as well (such as server and clock). Their sections can be anywhere in the file, but any section that starts with "AQ" must be in the order described. Each section can have multiple parameters on a line, separated by spaces, except the two signon sections, and within the LCQs. Within those sections, a line starting with an asterisk "*" is significant, as this is still used as a LCQ definition terminator.

A sample layout of the config file is :

```
(aqterm)
printinit=d;1b;14;d  alpha=18;14;d  graph=d ansi=y wy50=n tv910=n pe550=n tty=n pc=n

(aqdirs)
filtdir=/H0/ushear/FILTERS tempdir=/R0 datadir=/H0/DAT
aqdir=/H0/USHEAR/LOGS alog=AQLog mlog=MsgLog

(aqstation)
Station=RAND lat=43.5392 lon=72.3758 elev=386 naz=0 eaz=90 Netid=VT

(aqsignon)
Quanterra Q680 Very Broad Band Seismic Processor
(retsignon)
Quanterra VBB Data Retrieval System

(aqcomps)
z n e u v w zm nm em it p fp cd fc

(aqglobal)
level=2 freq=1500 resync=21600 defseed=y pass=4.6687725272870111d+025
det=500 tim=1000 cal=100 data=200 verbosity=2 showseed=yes showlevel=yes
chartname=/pipe/chart chartsize=500

(aqdetect)
zvbbcom nvbbcom evbbcom

(aqlcqs)
*
cmp=z str=vbb adc=hz20 fixed=1 offset=0 reversed=n
dacf=spwwss
mkmhh=spww detf=spwwss fhi=2 flo=40 iw=160 nht=4 x1=33 x2=23 x3=8 xx=13 tc=500 wa=507 av=8
mkmhh=remote detf=spwwss fhi=2 flo=40 iw=160 nht=4 x1=33 x2=23 x3=8 xx=13 tc=500 wa=507 av=8
mkthr=thrvbb fhi=1000000 flo=-1000000 iw=500 nht=15 wa=100
detector=z,vbb:spww+z,vbb:remote+z,vbb:thrvbb
comlink cpri=20 epri=24 wccom=y wecom=y detcom=z,vbb:spww+zvbbcom
chart ch_filt=SPWWSS ch_gain=2500 ch_mag=40000 ch_hours=6 ch_traces=72 ch_freq=1
wcseq=y webuf=y wcbuf=y rdebug=n avglen=0 dacnum=1 scale=100 dacon=n gap=0.33
*
More LCQ's
* Terminating asterisk required.
```

(comlink)
levels=32 mprio=16 cprio=4 port=/t3 resend=10 pkts=300 delay=20 fmt=QSL

(chart)
outpath=/p format=hpgl
size=letter maxpoints=12000 pre_delay=30 post_delay=60
pre_string=^(&l32259.1057J post_string=^(&l1057.32259J
text_pen=SP1,3; minute_pen=SP1;PT1; trace_pen=SP1,2; event_pen=SP1,4;
cal_pen=SP1,3;

(tape)
det=50 cal=50 data=50 tim=50 msg=50
changesec=60 mindays=60 logdir=/h0/ushear/logs
path=/dat
warnat=900 fullat=1000 retry=10 throttle=5 autounload=y timeout=10
append=y verify=y
path=/car
warnat=110 fullat=145 throttle=5 autounload=y retension=y timeout=10
append=y verify=y cache=y

(server)
q380=1,1-3,0,supercal,1-8,r
auxerr=y
gpsserial=y
throttle=0
retrysecs=3600

(clock)
verbosity=1 model=6 modulus=0.0125 phaselock=y format=qts
path=/t2 timeout=300 losslevel=1 delaymin=5 plldelaymin=10

(nmxclient)
ipaddr=128.103.105.12 tcpport=33333 station=rand startchan=1 measure=60
socknum=1

(rddso)
port=/t1 startchan=1 sync=d

(bcfg_c)
vbb 500
lp 500
vlp 500
ulp 500

(bcfg_e)
vsp 1000
lg 1000
vbb 1000

Page 65 & 66:

The ctlch and ttype sections are combined into the new **[aqterm]** section. ***Pass*** is moved to the [aqglobal] section, described later.

Page 67:

The dirs section uses the header name **[aqdirs]**.

Page 69:

An additional keyword is provided. ***netid=xx*** will set the Seed network ID to *xx*. If not defined the Seed network ID will be two spaces and a warning will be generated. *XX* must be one or two characters. The header name is **[aqstation]**.

Page 70:

This is split into two sections, the local signon has the header **[aqsignon]** and the retrieval signon has the header **[retsignon]**.

Page 71:

There are 7 additional components, default names are **R3, R4, R5, R6, R7, R8, and R9**. The header name is **[aqcomps]**.

Page 72:

The digs section is removed.

Page 75:

The pipes section is removed.

Page 77:

The tapes section is removed.

There are two new sections before we get to the LCQs. **[aqglobal]** combines old sections that were after the LCQs and a few other misc. flags. The following flags can be set in this section :

- ***level=n*** where *n* is 1, 2, 3F, 3S, or 3A. This is the default compression level if not specified in a LCQ. If not specified it is 1. 3F is Steim level 3, first differences. 3S is level 3, second differences. 3A is level 3, adaptive, which means that the first frame of each com record (normally 7 data frames) is compressed using both first and second differences, and which is best is used for the rest of the frame.
- ***freq=n*** where *n* is between 0 and 4095. This is the starting TCXO frequency control value, if not specified, it is 1500.
- ***resync=n*** where *n* is the number of seconds between resynchronizing the OS9 clock. If not specified, it is 86400 (daily). To disable this function, set n to zero.
- ***defseed=y*** will enable generation of default SEED names, which can be overriden in each LCQ. The default seedname is created by concatenating a one letter code for the stream, followed by a two letter code for the component. The codes for streams are :

  "B" = VBB "E"=VSP "H"=LG "M"=MP "L"=LP
  "V"=VLP "U"=ULP "C"=UBB "D"=USP "F"=SBB "G"=SSP

  The two letter codes are in this order :

  HZ, HN, HE, XZ, XN, XE, MZ, MN, ME, CI, TI, PI, X1, X2, X3, X4, X5, X6, X7, X8, Y1, Y2, Y3, Y4, Y5, Y6, and Y7.
- ***pass=n*** where *n* is encoded password (moved here from the old ctlch section).
- ***det=n*** where *n* is the number of records for the detection log file. If not specified, then no detection log is used.
- ***tim=n*** where *n* is the number of records for the timing log file. If not specified, then no timing log is used.
- ***cal=n*** where *n* is the number of records for the calibration log file. If not specified, then no calibration log is used.
- ***data=n*** where *n* is the number of records for the data log file (data to be written to tape, using tapeserv). If not specified, then the archive bit is set for the records written to the buffer servers.
- ***verbosity=n*** sets the default verbosity level for Aqsample, can be overridden on command line.
- ***showseed=y*** sets the default display mode for Aqshell and Retrieve, can be overridden on command line.
- ***showlevel=y*** changes the seed-name/frame separator character in the Aqshell status display to show the compression level in use. If *showlevel=n* (default) then the character is always a colon. If *showlevel=y* then level 1 is shown as a period, level 2 is a colon, level 3 first differences is a single quote, and level 3 second differences is a double quote.
- *blockettes=y* inserts calibration and detection blockettes into 4K records written to disk and tape.
- ***chartname=pipename*** sets the default chart pipe name. In an LCQ, if you specify "chart" then this name is substituted for the name.
- ***chartsize=size*** sets the size of the default chart pipe. Aqsample is designed to not write to the pipe if doing so would overflow the pipe, thereby hanging aqsample.

The **[aqdetect]** section is used to redefine the names of the 32 available com events. The default names are COMM:1 through COMM:32. You can replace these with more descriptive names by just listing the names. The first name listed renames COMM:1, the second COMM:2, etc.

Page 80:

The header name for this section is **[aqlcqs]**. The ordering of LCQ definitions is now relaxed. The only restriction are:

- There must be one and only one definition for a unique combination of stream and component.
- The components in a stream should all have the same sampling rate, a warning is issued for a mismatch.
- No forward definitions are allowed. If you are deriving data from another LCQ, it must already be defined. Likewise, detectors must be defined before being referenced by a LCQ.
- When using the "*source=*" or "*detector=*" keywords, searching will be done in reverse order, so the closest match is used.

Page 81:

The *queue* keyword is removed. endcfg is no longer needed, the end of the file, or the start of a new section will suffice.

Page 82:

There are four additional streams available :

- **UBB** - Ultra Broad Band.
- **USP** - Ultra Short Period.
- **SBB** - Super Broad Band.
- **SSP** - Super Short Period.
- All stream specific restrictions have been removed.
- **ADC** must specify the sampling the sampling frequency in Hz (such as 80). The letters "hz" may appear at the beginning or end if desired. "hz80" and "80hz" are interpreted as "80". Auxiliary channels are referenced by using the keyword ***AUX*** and will be considered 1Hz data. If ***STATUS*** is used as the ADC, the following status channels are available as 1Hz data :
  fixed=1 through ***fixed=10*** will be the number of free packets in the first ten dacommo links.
  fixed=11 is the clock drift in microseconds.
  fixed=12 is the clock frequency control value (0 - 4095).

Page 83:

- ***FIXED*** is the channel number from the server, the first channel is 1. The number of channels available for any particular frequency are digitizer and server configuration dependent.
- ***RANGED*** is no longer allowed.

Page 84:

On Q380 systems, a detector may be run on the digitizer, instead of on the main processor using the keyword ***REMOTE*** as the detector name. Only one remote detector may be run on any channel, so it is not possible to have detectors on 80 and 20Hz on the same channel. Although multiple non-remote detectors may be defined per LCQ, you may not want to have all of them enabled at the same time. The keyword "**off**" may be used to make the detector disabled until enabled by a command over the comm link. If the "off" keyword is not used, the detector will be initially enabled, until disabled by a comm link command.

Parameters are defined for a threshold detector. This type of detector would normally be used for ***AUX*** and ***STATUS*** channels. To define a threshold detector instead of a Murdock-Hutt detector, replace ***mkmhh*** with ***mkthr***. The following parameters are used :

- fhi is the high limit.
- flo is the low limit.
- iw is the hysterisis. This is the number of counts that the samples must decreased below the high limit to turn off the event, or increase above the low limit to turn off the event.
- nht is the number of samples in sequence that must exceed a limit to be considered an event.
- wa is the number of samples to wait after the samples are back within limits before the detection flag is turned off.

Page 85:

DETECTOR now allows logical combinations of detectors, plus some more detector types. Logical operations available are:

- **&** Logical AND - Precedence Level 2
- **+** Logical OR - Precedence Level 3 (lowest)
- ***!*** Logical NOT - Precedence Level 1 (highest)
- ***** Logical Exclusive OR - Precedence Level 3 (lowest)
- *(....)* Parenthesis can be used to modify the precedence ordering.

For instance, "*Detector=(z,vbb:spww+z,vsp:vspww)&n,vbb:spww*" would only trigger event recording if either the first two detectors are on, and the third detector is on. Detection can be triggered by a flag from the DP side over the com link. There are 32 available flags, ***COMM:1*** through ***COMM:32*** (or the equivalent names defined above). Q380 systems allow doing event detection in the digitizer. The output of this detector is referenced by using the keyword ***REMOTE***, signifying a detection on this channel. Detection can also be triggered by a digitizer calibration using the keyword ***CALON***, signifying a calibration on this channel**.**. All of these can be used just like any other detector, so that "*Detector=REMOTE+CALON+COMM:5+z,vbb:spww"* is valid.

DETECTOR=<equation> specifies the "general detector". This "general detector" is used if a output destination does not define it's own detector. To specify a detector for use only on buffered (disk) output, use ***DETBUF=<equation>***. To specify a detector for use only on sequential (tape) output, use ***DETSEQ=<equation>***.

WCGOES is no longer supported.

Page 86, 87, 88:

WCCOM, WECOM, CPRI, and ***EPRI*** are now used with a specific com link, and there may be multiple comlinks defined. Formally you only needed to use these flags, and the system assumed the ***DACOMMO*** comlink, as it was the only one available. Multiple *DACOMMO* type comlinks are now allowed, as well the ***NSNCOM*** type comlink. For the NSNCOM type comlink the new definition ***NSNID=n*** now must be specified. *N* must be between 1 and 255 and is assigned by the NSN folks to identify a channel/rate combination.

The new definition ***COMLINK[=name]*** now must be specified, followed by the above flags on the same line, to send data through the *name* socket. If you wish to write to the standard *DACOMMO* (with no command line option), then use *comlink=dacombuf* or *comlink*. The default name for the *NSNCOM* comlink would be *comlink=nsncombuf*". Both programs allow setting a different name.

The ***CLEV*** parameter allows setting the compression level: ***CLEV=n*** will use compression level *n* on this LCQ, rather than the default value specified earlier. A control detector for this comlink can be specified by using ***DETCOM=<equation>***.

The ***WCCOM, WECOM, WCSEQ, WESEQ, WCBUF, WEBUF*** can have a flag of *Y* (yes), *N* (no), or *A* (available). Available means that the recording is available, but not enabled as a default. It can later be enabled by a command through dacommo by the DP.

SEED=[location-]name can be used to override the default seed name (the file of names method is no longer supported). If the optional one or two character location is not specified, then the location will remain as two spaces. Examples : "seed=bhz", "ab-bhn".

Gap=samples will set the number of samples that will cause a flushing of the 4K records on the tape or disk. The default is 0.5 samples.

Caldly=seconds will set the amount of time after a calibration ends and the detection flag goes off. The default is 60 seconds.

Cal0dB=value will define the voltage or current that corresponds to a 0dB calibration. This information is passed in the Calibration Blockettes.

Automatic dispatching of chart commands can be accomplished by specifying the parameters in the LCQ description if desired. Refer to the chart preliminary documentation for more complete descriptions of the parameters. Note that specifying minutes in addition to hours is not supported for automatic dispatch. All of the required parameters must be on the same line in the config file and ***chart=*** must be the first one :

- ***chart[=pipenam*e]** where *pipename* is the pipe used to write the commands to. If the pipename is not specified, then the default pipename defined in the aqglobal section is used.
- ***ch_filt=iirname*** where *iirname* is the name of the optional iir filter to use.
- ***ch_gain=system-gain*** where *system-gain* represents the total gain including seismometer, digitizer, and any filter gains.
- ***ch_mag=magnification*** where *magnification* is at the specified frequency.
- ***ch_freq=hz*** where *hz* is the frequency at which magnification is valid.
- ***ch_hours=hours*** where *hours* is the number of hours (and therefor the interval) to be printed on one chart.
- ***ch_traces=count*** where *count* is the number traces to print on one chart.

If none of the dacommos used by this LCQ are running in fixed length mode (see below), then you can select the number of frames in each record by using the ***COMFR=count*** option. *Count* is the total number of frames (one is always used by the header) and must be between 2 and 8. If one of the dacommos is running in fixed length mode, then this option will be ignored and the count dictated by the dacommo will be used instead. This valud defaults to 8. (8 • 64 = 512 bytes).

The maximum number of frames in a 4K disk or tape record can also be limited if desired to increase response time on low rate data by using the ***4KFR=count*** option. *Count* must be between 2 and 64 (64 • 64 = 4096 bytes). The actual number of data frames will always be an integral multiple of the number of data frames in a 512 byte dacommo record. Typical values are for Q512 or QSL dacommo format would be 7 data frames per dacommo record times 9 dacommo records is 63 data frames in a 4K record (plus one header frame). For MM256 dacommo format there are 3 data frames per dacommo record times 21 dacommo records is 63 data frames per 4K record. If you are running dacommo in QSL mode (or not running dacommo on this LCQ at all) and set *COMFR=4* and *4KFR=20*, then there would be 3 data frames per dacommo record and 6 dacommo records for a total of 18 data frames and one header frame in a 4K record.

The maximum number of outstanding frames in a 4K record before writing to disk can now be set. By default, if the rate is greather than or equal to 80hz, the flush threshold is 60 frames. Between 40 and 79Hz it is 30, between 20 and 39Hz it is 15, and below 20Hz it is 5 frames. You can adjust this value by using the ***FLUSH=count*** option, where *count* must be between 2 and 64.

Page 93:

The logs section is moved to the [aqglobal] section.

Page 94:

These clock parameters were moved to the [aqglobal] section.

Page 96:

The standard *DACOMMO*, with no change in module name, is referred to as **[comlink].** Equivalent sections can be defined for a dacommo type com link that uses the M command line option to specify the module name. In this case, the keyword is the same as the module name specified (in brackets).

Page 97:

There are sixteen additional parameters that may be specified:

- ***DETPRIO=n*** where *n* is the priority level of detection packets, if not specified, defaults to message priority.
- ***TIMEPRIO=****n* where *n* is the priority level of time correction packets, if not specified, defaults to message priority.
- ***CALPRIO=n*** where *n* is the priority level of calibration packets, if not specified, defaults to message priority.
- ***TIMEOUT=n*** where *n* is the time-out in seconds for the serial port, purpose unknown.
- ***WS=n*** where *n* is the window size. The window size must be between 2 and 64, with a default if not specified of 8.
- ***RCE=y*** will echo remote commands (excluding the remote commands that have responses, such as ULTRA_REQ). *RCE=n* to disable echo, which is the default.
- ***IPADDR=address*** sets DACOMMO to communicate via TCP/IP rather than a serial port.
- ***PORT=n***. If the *IPADDR* parameter is set then this parameter is redefined to specify the TCP/IP port number instead of the serial port and must be between 5000 and 65535.
- ***MAXRESENDS=n*** limits how many times packets can be resent without getting a acknowledge when using a TCP/IP connection. If this limit is exceeded the network connection will be closed and then re-opened.
- ***SYNCTIME=n*** sets the sync packet interval. If there are no packets waiting to be sent after this number of seconds since the last packet was sent, a sync packet will be set (regardless of grouping). The default is 10 seconds.
- ***RESENDPKTS=n*** sets how many packets are sent out a time when resending packets. The default is 7, or WS-1, whichever is lower. When packets are resent, they are grouped into smaller units and each unit must be completely acknowledged before the next unit. When all packets that were in the window at the time resending began are acknowledged normal streaming mode is continued.
- ***NETDLY=n*** sets the number of seconds that Dacommo waits in between closing the last network connection and opening a new one. The default is 120 seconds.
- ***NETTO=n*** sets the network connection timeout. If no connection is made during this timeout, the connection is aborted and a new connection attempt is made after *NETDLY* seconds.
- ***GRPSIZE=n*** sets the size of packet transmission grouping. The default is 1 in which case there is no grouping, packets are sent as soon as they are available. If n > 1 then that many packets must be available before the whole group is sent. This parameter must be no higher than the window size - 1.
- ***GRPTIME=n*** sets a timeout so that if *GRPSIZE* packets are not available, after this timeout, whatever packets are available are sent. The default is 0, so this parameter must be set if *GRPSIZE* is > 1.
- ***FMT=<format>*** will set the communications format, which defaults to ***QSL*** (Quanterra Smart Link). This format is based on the standard dacommo format in use for a number of years (now called ***Q512***) in that it uses 512 byte records. QSL expands on this by only sending the amount of data required, rather than always sending 512 bytes of data. This reduces comlink loading as well as CPU loading on both ends. The packet sequence will also have a larger modulus than the window size and will be the largest multiple of the window size less than or equal to 256. This is to provide more secure link aknowledgments, should some old packets get stuck in a modem somewhere, you won't be acknowledging data from a previous window. In order to use this format the dpcommo must be modified to :
    1) Check for the "ETX" byte anywhere within the record transmission, indicating end of record. It must not interpret an ETX preceeded by a "DLE" escape character as end of record however. The end of the record can be found by subtracting the size of the error control packet (6 bytes) from the location of the ETX.
    2) Must stop decompressing frames when the number of samples is used up, rather than decompressing all possible frames within a record.
    3) Before processing any records, must send a "LINK_REQ" packet, and then wait for the "LINK_PKT" for details on format and sequence modulus. See the "Ultra Shear DP↔DA Data Structures documentation for details on these packets.

Two other formats are supported for backward compatibility. ***Q512*** is the normal Quanterra format, sending fixed length 512 byte packets, and with a sequence modulus the same as the window size. ***MM256*** is the Martin Marietta format currently used by GSN. It is fixed length 256 byte packets with a different header structure from Q512 or QSL.

Page 99:

This section is removed.

Page 100:

The default compression parameter was moved to **[aqglobal]**.

Server Configuration Section

A new section is added to configure the digitizer server. The available parameters depend on the digitizer used. In all digitizers, the configuration file is ignored until the keyword **[server]**, and parameters are processed until a new keyword is found.

Q380 DIGITIZER

The available parameters are :

- ***AUXERR=y*** - write error messages to standard error path.
- ***AUTOSTART=y*** - release reset automatically (so doesn't require serv_start or servtest) after 5 seconds.
- ***GPSON=y*** - enables polling of GPS through SPI connection.
- ***GPSSERIAL=y*** - GPS is connected through serial port.
- ***AUXCAL=y*** - write calibration status messages to standard error path.
- ***INTCAL=y*** - use internal timemarks during calibration.
- ***DATABUF=nn*** - where *nn* is the number of seconds of data to buffer.
- ***ERRBUF=nn*** - where *nn* is the number of error messages to buffer.
- ***SVCQ=nn*** - where *nn* is the number of service queues to allocate.
- ***VCO=nn*** - where *nn* is the initial VCO control value.
- ***SOH=path,low-high*** - where *path* is the serial port for the remote SOH board. This parameter may be repeated two times. *Low* is the aux channel number where the first channel is mapped to. *High* is the highest aux channel to write to.
- ***QPWR=low-high*** - *Low* is the aux channel number where the first A/D on the power board is mapped to. *High* is the highest aux channel to write to.
- ***Q380=slot,low-high,freq[,caltype,alow-ahigh{,options}]*** - Defines a QDP380 board as installed as board "*slot*". Slot must be between 1 (1st board on QVI number 1) and 4 (2nd board on QVI number 2). "*Low-high*" defines the channels where the data from the QDP will be reported. "*freq*" is 0 for 80/40/20/10/1 Hz output, and 1 for 160/20/1Hz output. If a calibrator board is installed then caltype is either "*QAPCAL*" or "*SCAL*". "*Alow-ahigh*" are the aux channel number range where the aux data from that calibrator board will be mapped to. Available "options" for **QAPCAL** are :
  - ***C*** - Common enable output line (STS-2).
  - ***Ennn*** - Calibration enabled for channels *n*, *n*, and *n*. Default is E123.
  - ***R[ddd[/lll[/hhh[/iii]]]]*** - Recentering available, optional override of default, low, high, and increment in ms.

  Available options for **SCAL** are :
  - ***Ann*** - Maximum calibration amplitude (-96 to 0) in dB.
  - ***C*** - Coupling option available.
  - ***Dnn*** - Maximum calibration duration in seconds.
  - ***Ennn*** - Calibration enabled for channels *n*, *n*, and *n*. Default is E123.
  - ***Fn*** - Always use filter *n.*
  - ***Mnn*** - Calibrator output is monitored on channel *nn*
  - ***R[ddd[/lll[/hhh[/iii]]]]*** - Recentering available, optional override of default, low, high, and increment in ms.
- ***GAIN=channel,gain*** - *Channel* is the digitizer channel (1-n) and *gain* is a floating point gain, the default gain for each channel is 1.0.
- ***THROTTLE=value*** - Defaults to 20, lower values (down to 0) will reduce latency (desirable for real-time output such as NMXP) but will increase CPU usage.
- ***RETRYSECS=seconds*** - How many seconds to wait before trying to put a digitizer channel back into the map. The default is 0 which means no retry.

Q4120 DIGITIZER

- ***PHYSICAL=n*** - where *n* is the maximum number of physical channels, normally either 4 or 8.
- ***AUXERR=y*** - write error messages to standard error path.
- ***AUTOSTART=y*** - release reset automatically (so doesn't require serv_start or servtest) after 5 seconds.
- ***DATABUF=nn*** - where *nn* is the number of seconds of data to buffer.
- ***ERRBUF=nn*** - where *nn* is the number of error messages to buffer.
- ***SVCQ=nn*** - where *nn* is the number of service queues to allocate.
- ***SAFETY=nn*** - where *nn* is the number of data buffers to reserve for catch-up.
- ***MSGS=nn*** - where *nn* is the number of DSP to host message buffers.
- ***VCO=nn*** - where *nn* is the initial VCO control value.
- ***DELAY=n*** - where *n* is the number of seconds after the TMS320 starts before it starts any 2105s, defaults to 2 seconds.
- ***SOH=path,low-high*** - where *path* is the serial port for the remote SOH board. There may be up to 2 SOH boards. *Low* is the aux channel number where the first channel is mapped to. *High* is the highest aux channel to write to.
- ***2105=path*** - file name for 2105 object code. Different files may be available for different filters.
- ***TMS320=path*** - file name for TMS320 object code.
- ***FIRFILT=path*** - file name of file containing TMS320 FIR filters.
- ***MAPFILE=path*** - *path* to write optional channel/memory map.
- ***GPS=magnavox|motorola|gps1|gps2*** -***Magnavox*** indicates that there is a Magnavox GPS Engine attached to /x5. ***Motorola*** is not currently supported. ***GPS1*** indicates that there is an external GPS1 connected to some serial port (controlled by clock). ***GPS2*** indicates that there is an external GPS2 connected to some serial port.
- ***GPSSTAT=daily|hourly|10min*** - Sets automatic status reporting interval.
- ***GPSAUTO=y*** - Enable reporting GPS status when changing clock states.
- ***GPSOPTS=[reset],[defon]*** - If *reset* is specified, the GPS Engine will be reset whenever the server starts, else the GPS Engine is only reset if the RF power was off when the server starts. If *defon* is specified it indicates that the 4120 is jumpered so that the default state is RF power on.
- ***CHECKGAP=y*** - Server checks for data gaps (this is always done in Aqsample).
- ***NOXFER=y*** - The DSP section transfers messages only to server, no data.
- ***AUTOGAIN=n,counts*** - Set the gain on channel *n* so that the output value of stage 1 is *counts* when the calibration voltage is applied. This is not supported on current Analog input boards. If this option is used, the GAIN option is ignored.
- ***GAIN=n,value*** - Set manual gain on channel *n* to floating point *value*.
- ***TICKRATE=n*** - Sets sampling rate, defaults to 20000. 32000 is the other supported rate.
- ***STAGE=1,frequency,delay[,>channels[*mul]][,>>channels]***
- ***STAGE=n,frequency,filter_name,<source[,>channels[*mul]][,>>channels]*** - where *n* is the TMS320 stage number, *delay* is the 2105 filter delay in seconds, *filter_name* is the name of the TMS320 filter, and *source* is the stage from which this stage derives it's input data. Each stage can have two outputs, one (defined by the >) that goes to the server as a fixed point value, and the other which is passed through to the input of other stages as a floating point value. *channels* are the channels to output or pass through and *mul* is an optional scaling multiplier to bring the fixed point output into the desired range. For instance "*>1234*0.25 >>5678*" will output to the server channels 1, 2, 3, and 4 with the value multiplied by 0.25 and allow channels 5, 6, 7, 8 and to be passed through to other stages. You cannot use a stage as a source for a channel unless that channel is defined as a pass through on another stage. You cannot pass data through unless there is another stage to receive it. The maximum number of available output frequencies is "**MAXFREQ**" (defined in servdata.i) and the maximum number of TMS320 filter stages is "**MAXSTAGE**" (defined in server.i). Output is not allowed for frequencies above 1Khz. If a stage defines a channel/frequency combination already defined, it is assumed to be a detector filter, and its channels are mapped into logical channels to avoid conflict. The server "get_logical" call can be used to determine the logical channel based on the physical channel number and filter name.

DSP Data Flow

Raw A/D data and a quantagrator bit are input to an AD2105 fixed point DSP (one per channel) at 20Khz. The 2105 DSP does the quantagration processing and filters the result down to a 1Khz data stream, with an absolute maximum range of ±26 bits. Data from all of the 2105's is multiplexed into a single TMS320 floating point DSP. During startup the 320 calculates both a fixed point channel offset and floating point channel gain factor (for automatic calibration, or manual gain setting in config file). Before the 2105 data is processed the offset is subtracted, converted to floating point, and multiplied by the channel gain factor. Data is passed from filter stage to filter stage in floating point format. Data to output to the server is multiplied by a stage multiplier (*mul, or 1.0 if not specified) and then converted to fixed point.

HRDCU/LRDCU DIGITIZERS

The available parameters are :

- ***AUXERR=y*** - write error messages to standard error path.
- ***AUTOSTART=y*** - release reset automatically (so doesn't require serv_start or servtest) after 5 seconds.
- ***VERBOSITY=y*** - Generates messages for HRDCU/LRDCU initialization.
- ***GPSSERIAL=y*** - GPS clock is connected to a serial port.
- ***DATABUF=nn*** - where *nn* is the number of seconds of data to buffer.
- ***ERRBUF=nn*** - where *nn* is the number of error messages to buffer.
- ***SVCQ=nn*** - where *nn* is the number of service queues to allocate.
- ***VCO=nn*** - where *nn* is the initial VCO control value.
- ***RETRIES=nn*** - where *n*n is the number of times the server will re-initialize the digitizers if a LRDCU starts without finding the external clock from the HRDCU.
- ***SOH=path,low-high*** - where *path* is the serial port for the remote SOH board. This parameter may be repeated two times. *Low* is the aux channel number where the first channel is mapped to. *High* is the highest aux channel to write to.
- ***HRDCU=path,delay*** - where *path* is the serial port for the HRDCU. *delay* is the HRDCU filter delay in seconds. The HRDCU must be defined before any LRDCUs. The HRDCU is assigned channels physical channels 1 through 4 (4 is the calibration monitor channel).
- ***LRDCU=path,delay,freq,channels*** **- where *path* is the serial port for a LRDCU. *delay* is the LRDCU filter delay in seconds. *freq* is the output sample rate in Hz.** *Channels* are the number of normally active channels (1-6). The server always adds one for the calibration channel.

Example :

```
HRDCU=/X1,0.045
LRDCU=/X2,0.03,100,3
LRDCU=/X3,4.537,1,6
```

Defines a HRDCU as channels 1 through 4 on port "/x1" with a filter delay of 45 milliseconds, a LRDCU as channels 5 through 8 on port "/x2" at 100Hz with a filter delay of 30 milliseconds, and another LRDCU as channels 9 through 15 on port "/x3" at 1Hz with a filter delay of 4.537 seconds.

Tape Configuration Section

The tape server is a background process that monitors tape activity, log activity, and writes the new data in all logs to the tape when any of the logs reaches the threshold for that log. The configuration file is ignored until the keyword "**[tape]**", and parameters are processed until a new keyword. The first section is the parameters global to all tapes :

- ***LOGDIR=path*** - specifies the directory where to search for log files to archived to disk. Any valid log file must have the string "log" somewhere in it's name, such as aqlog, data_log, etc.
- ***TIM=percent*** - specified at what percentage of the timing log being full where the logs will be written to tape. If not specified or *percent* is zero then the timing log will not be written to tape.
- ***MSG=percent*** - specified at what percentage of the message log being full where the logs will be written to tape. If not specified or *percent* is zero then the message log will not be written to tape.
- ***DATA=percent*** - specified at what percentage of the data log being full where the logs will be written to tape. If not specified or *percent* is zero then the data log will not be written to tape.
- ***MINDAYS=days*** - specifies how old a tape volume header must be (previous to the current day) for it to be automatically considered a blank tape.
- ***CHANGESEC=seconds*** - specifies how many seconds tapeserv will wait for you to change a tape or wait for a command confirmation.
- ***CACHEFILE=path*** - this optional parameter is the name of a disk cachefile created using the "makecache" command. To use the cachefile, "cache=y" must be specified for each drive to use the file. Note that cacheing is only available for Archive Viper tape drives.
- ***VERBOSITY=yes*** - enables more detailed messages when writing logs to disk.
- ***BOT=filename{,filename}*** - Defines a list of files that will be written to tape as log records at the beginning of the tape, immediately following the volume header record.
- ***EOT=filename{,filename}*** - Defines a list of files that will be written to tape as log records at the end of the tape, followed by an "END OF TAPE" log message, and then the 40 end of file marks.
- ***PATH=path*** - specifies a new tape drive. All of the following commands reference only to the last tape drive specified using this parameter. Up to 4 path parameter may be specified.

These commands only apply to the last "path" specified :

- ***WARNAT=size*** - specifies when tape changing warning messages will start appearing. The size is in megabytes. If not specified no warning messages will be issued.
- ***FULLAT=size*** - specifies when a tape change will be forced. If not specified the tape change will not be forced until the tape reaches End of Media.
- ***RETRY=count*** - the number of times tapeserv will try to write to a tape if there is an error when starting a write. This is intended for DAT tapes which sometimes think their offline after a week or two.
- ***THROTTLE=ticks*** - this inserts sleeps (for the specified number of ticks) between writing SEED records to the tape. If this value is not set, it may cause the system to lose digitizer data during tape writes.
- ***AUTOUNLOAD=y*** - indicates that you want to do an unload command when a tape is full. This has no effect on a cartridge tape, but will eject a DAT tape.
- ***RETENSION=y*** - indicates that the drive has a retension command available. Cartridge tapes do, DAT tapes don't.
- ***OVERWRITE=y*** - indicates that the program may overwrite data on this tape if there are not other available tapes and this tape has the oldest volume header.
- ***TIMEOUT=count*** - specifies how many times to retry a tape SCSI command if the drive returns a "busy" status. There is a one second delay between each attempt.
- ***VERIFY=y*** - indicates that the data written to a tape should be verified.
- ***CACHE=y*** - indicates that data should be first written to the cachefile on disk (or ramdisk) and then copied to the tape directly from the disk. Only cartridge tapes support this feature.
- ***APPEND=y*** - indicates that if a valid SEED volume header is found, that the drive should seek to end of data and check for filemarks. If no filemarks are found, then tapeserv will append data to this tape. If filemarks are found, or the APPEND flag is off, then the tape will be considered full.
- ***SEMILOAD=y*** - indicates that the drive should be put into a "semi-loaded" state between writing to logs. This is only valid on Hewlett Packard drives.
- ***PREVENT=y*** - indicates that removal of the tape should be prevented on an active drive. Has no effect on cartridge drives.
- ***COMPRESS=y*** - enables compression on Hewlett Packard DAT drives. To allow this option to take effect, SW1 must be OFF and SW2 must be ON.
- ***REPORT=y*** - enables writing extended status information after an error condition.
- ***FATALTICKS=ticks*** - where *ticks* is the maximum number of systems ticks (normally 10ms) before a SCSI command is considered hopeless. The default is 360000 ticks (or about 1 hour).

Clock Configuration Section

The clock now uses the configuration file, rather than command line parameters. The configuration file is ignored until the keyword "**[clock]**" and parameters are processed until another keyword is found. The parameters available are :

- ***FORMAT=string_format*** - where string format can be :
  - ***QTS*** - Quanterra Time & Space format for GPS1 (Magnavox) clocks.
  - ***QTS2*** - Quanterra Time & Space format for GPS2 (Motorola) clocks.
  - ***UA31S*** - Meinberg UA31S DCF77 Clock.
  - ***OMDC*** - Kinemetrics OMDC, 432DC, or 77DC.
  - ***OS9*** - operation using the OS9 system clock.
- ***YEAR=yrsrc*** - a parameter only used with an OMDC format clock, which doesn't know what year it is. yrsrc can either be the current year (such as 1994), or OS9 to use the year from the OS9 system clock.
- ***VERBOSITY=n*** - where n is 0 for no messages, 1 for errors and offset setting messages, and 2 for messages every timemark.
- ***PHASELOCK=y*** - enables phase locked frequency control.
- ***M0DULUS=n*** - where n is the fraction of a second to which digitization times are kphase locked. Default is 1/80 second, or 0.0125.
- ***PATH=port*** - sets the serial port from which the timestring is to be read. "SERVER" is used if the Quanterra GPS clock is connected to the system via the SPI port.
- ***MODEL=n*** where n specifies the clock model that will be reported in the timing logs. The following models are defined :
  0 = unknown clock.
  1 = OS9 internal clock.
  2 = Kinemetrics Goes clock.
  3 = Kinemetrics Omega clock.
  4 = Kinemetrics DCF clock.
  5 = Meinburg UA31S clock.
  6 = Quanterra GPS1 (or Magnavox) with QTS format.
  7 = Quanterra GPS1 with Goes emulation.
  8 = Quanterra GPS1 with UA31S emulation.
  9 = Quanterra GPS2 (or Motorola) with QTS2 format.
- ***LOCKLEVEL=n*** - allows adjusting the minimum reception quality for phase lock, at the default of 1, the highest quality is required.
- ***DELAYMIN=minutes*** - allows setting the delay in minutes after a clock loses the accurate time lock before changing quality from 0 to 2. This is designed to allow momentary losses of satellite to not interfere with phase lock operation while it operates on it's internal clock. Default is 60 minutes.
- ***PLLDELAYMIN=minutes*** - allows setting the delay in minutes after external lock is lost and when the PLL stops tracking.
- ***TIMEOUT=seconds*** - Sets the timeout, in seconds, after which loss of the clock string will generate an error. If the clock string is not received for this timeout, the quality level will be set to the following parameter until the clock string returns. A value of zero (default) disables this feature.
- ***LOSSLEVEL=n*** - Sets the quality level if the clock string is not received.
- ***LOSSINTERVAL=seconds*** - How often an error message will be generated while the clock string is missing, default is 3600 (1 hour).
- ***ZONEOFFSET=seconds*** - Sets an offset (plus or minus) from UTC for those people who wish to time data using local time rather than UTC.
- ***SYSTOLERANCE=seconds*** - Sets how close the difference between sucessive minute strings and the OS9 clock must be (plus or minus) when waiting for clock string acceptance, default is 10 seconds. If the difference is greater than this number of seconds, the acceptance time will be recalculated using the new difference.
- ***MINSYSDIFF=seconds*** - Determines how close the time string has to be to the OS9 clock to be accepted on the first time. If there is greater than this many seconds, it will take 2 or more minutes to accept the clock string as valid. Default is 180 seconds.
- ***SYSRATIO=n*** - Sets the ratio of time between how far apart the clock string and OS9 clocks to the number of minutes that will elapse with a consistant difference in those two clocks. The number of minutes is calculated as truncate((<clock difference> + (<sysratio> - <minsysdiff>) / <sysratio>) + 1. Initial clock differences less than MINSYSDIFF will result as the first clock string being accepted, larger differences require more minutes for acceptance. The number of minutes for acceptance will be clipped at *MAXWAIT* minutes. Default ratio is 1000.
- ***MAXWAIT=minutes*** - Sets maximum time for clock acceptance. Default is 180 minutes (3 hours).

Chart Configuration Section

The Chart server is a background process accepts commands through a pipe to create plots on a HPGL printer. The configuration file is ignored until the keyword **[chart]** and parameters are processed until another keyword is found.

pipename=path where path is the name of the pipe that chart will create and other programs will write their commands to. No default. Can also be specified in the [aqglobal] section as chartname=path.

pipesize=bytes The pipe must be large enough to hold the worst case number of commands that may be sent at the same time, commands require approximately 80 bytes each. Default is 256 bytes. Can also be specified in the [aqglobal] section as chartsize=bytes.

outpath=path This is the port name for the printer/plotter.

format=HPGL The format to use. The default (and currently the only supported format) is HPGL.

size=letter | legal | A4 The paper size. Default is legal.

dpi=resolution Specifies the resolution (in dots per inch) you want to represent, defaults to 300. Higher resolutions produce more data to the printer and may overrun internal memory of HP printers due to bad firmware design in their HPGL translators.

maxpoints=samples This is the maximum number of samples per trace. This must be the worst case of any of the charts, calculated as HOURS * 60 * SAMPLE_RATE / TRACES. Default is 10000, absolute maximum is 100000. Note that system memory equal to MAXPOINTS * 4 bytes is obtained by Chart on startup.

pre_delay=seconds This command allows delaying the specified number of seconds after sending "pre_string" and before issuing any HPGL commands. Default is 0. 20 was a good value for the LJ2, and 15 for the LJ4.

post_delay=seconds This command allows delaying the specified number of seconds after sending "post_string" and before processing the next chart. Default is 0. 45 was a good value for the LJ2, and 30 for the LJ4.

pre_string=string This is typically used on HPGL emulation's on printers to setup the printer before accepting HPGL commands. Some users may also wish to have their printers left in normal mode between charts. For instance, to change from HPCL and perform other setups required, the following commands can be used on the LaserJet Series 2 with a Pacific Data Products Emulation Cartridge : **^[&l32259.1057J** while on a LaserJet 4 additional commands are required to also set landscape mode, set margins, and for non-letter paper, set the multipurpose tray as the input. For legal on the LJ4 : **^[E^[&l4H^[&l3A^[&l1O^[9^[&l0E^[%0B**. For letter on the LJ4 : **^E^[&l2A^[&l1O^[9^[&l0E^[%0B**. And for A4 on the LJ4 : **^[E^[&l4H^[&l26A^[&l1O^[9^[&l0E^[%0B**. Note that the **^** symbol is used indicate a control character follows, **^[** is Escape. **l** is lower case letter L, **1** is numeral one, **O** is uppercase letter O, and **0** is numeral zero.

post_string=string This is typically used on HPGL emulation's to put the printer back into HPCL mode. For the LJ2 with PDP cartridge : **^[&l1057.32259J** and for the LJ4 : **^[%0A^[E**.

text_pen=string HPGL was designed initially for pen plotters, so you could select between different pens as the plot was drawn. Chart allows setting up to 5 different pens depending on the item to be drawn. For a real plotter, the commands would be simply SPn where n specifies the pen to use. For emulation's there are various methods used to select the width of the pseudo-pen. On the LJ2 with PDP cartridge a command of the form **SP1,w** sometimes works, other times **SP1;PTw** works, depending on which pen is being specified, where w is the width of the pen. For the text pen, **SP1,3** is recommended on the LJ2 and **SP1;PW0.2** for the LJ4. The text pen is used for writing all text. Default is **SP1**.

minute_pen=string This pen is used to draw the minute lines on the chart. **SP1PT1** is recommended on the LJ2, and **SP1;PW0.05** on the LJ4. Default is **SP1**.

trace_pen=string This pen is used to draw the wave form traces when neither an event or calibration is in progress. **SP1,2** is recommended on the LJ2, and **SP1;PW0.1** on the LJ4. Default is **SP1**.

event_pen=string This pen is used to draw the wave form traces during events. **SP1,4** is recommended on the LJ2, **SP1;PW0.3** on the LJ4. Default is **SP1**.

cal_pen=string This pen is used to draw the wave form traces during calibrations. **SP1,3** is recommended on the LJ2, **SP1;PW0.2** on the LJ4. Default is **SP1**.

top=mm Sets the top margin in millimeters. The default of 0 is recommended for the LJ2, 20 for the LJ4.

left=mm Sets the left margin. The default of 0 is recommended for the LJ2, 2 for the LJ4.

right=mm Sets the right margin. The default value of 0 is recommended for the LJ2, -2 for the LJ4.

bottom=mm Sets the bottom margin. The default value of 0 is recommended for the LJ2, -2 for the LJ4.

Sysmon

Sysmon provides a simple interface for checking system operation and doing tape changes where there is no terminal available. The interface consists of two bi-color LEDs plus a pushbutton. This interface is currently only available for Q4120 systems but may be available for Q380 systems in the future.

One LED and the pushbutton are used for changing tapes. The LED has the following states :

- ***OFF*** - Tape running normally, no errors. A tape change can be started by pushing the pushbutton.
- ***FLASHING RED*** - Error, such as full tape or write protected tape. The tape may or may not be in the drive, in either case, a tape change can be started by pushing the pushbutton.
- ***STEADY RED*** - Pushbutton has been pushed, you may release the button.
- ***FLASHING RED/GREEN*** - Sysmon has requested a tape change be started. Tapeserv will flush logs to tape before writing EOF and ejecting the tape, so this could take quite a while.
- ***STEADY GREEN*** - Tape should now be ejected, you can remove the tape and put a new one in. After checking that the tape can be written the LED will change to either *OFF* indicating the change has been sucessful, or *FLASHING RED* indicating that there is something wrong with the tape. If there is a problem with the tape, push the button again, remove the tape when you get *GREEN*, and try another tape.

The other LED shows general acquisition operation :

- ***OFF*** - Acquisition not operating.
- ***FLASHING RED*** - Acquisition operating, with poor clock (quality is 1 or 2).
- ***FLASHING RED/GREEN*** - Acquisition operating with degrading clock (quality is 0).
- ***FLASHING GREEN*** - Acquisition operating with locked clock (quality is 3, 4, or 5).