

xFOCE Gateway Command Line Tools Reference
xFOCE-CLI-ref-151112 Thu Nov 12 2015

Gateway Management

[xsvc](#)

Start/stop gwrn, sds, sbr

[gwstart](#)

Start/stop gateway and related services

See also: [digicfg](#), [gwdio](#), [gwmeh](#), [xsvc](#)

[gwstop](#)

Stop gateway and related services

See also: [digicfg](#), [gwdio](#), [gwmeh](#), [xsvc](#)

XFHOME/src/gateway/apps

[gq](#)

Gateway query client: get/set/do

[digicfg](#)

Configure Digi virtual serial ports

Archive and Data Tools

[xpres](#)

Archive extraction/expression

[skim](#)

Present recent archive data

[gist](#)

Present archive summary metadata

[qplot](#)

Plot CSV data as PNG, PDF

[qsee](#)

Use xpres and plot to generate CSV data and plot images

Hardware Tools

(use gq when gateway is running)

[gwdio](#)

Read/set gateway DIO

[gwmvc](#)

MV converter control

[gwsn](#)

Sensor Node control

[gwmeh](#)

Main electronics housing control

[gwspi](#)

Read SPI devices

[moco](#)

Thruster motor control

[vaco](#)

ESW valve control

Test Tools

[snTerm](#)

Sensor node terminal

[snTest](#)

Sensor node control/test

[mvTest](#)

MV converter control/test

[subc](#)

Gateway pub/sub client

[subs](#)

Pub/sub test server

[vtest.sh](#)

Valgrind wrapper (Linux VM)

Scripting

[xcycle-start](#)

Run/log a specified profile

[xcycle-stop](#)

Stop a specified profile

[xprofile](#)

List/run a specified profile

[qprofile](#)

Define/run a motor profile

[wuntil](#)

Wait until specified date/time

Configuration Info

[gwv](#)

Show gateway software version

[bbsn](#)

Show Beaglebone hardware serial number

[catgpio](#)

Show Beaglebone GPIO configuration

[catsys](#)

Show results of df, ls (arch, logs) , top

Software Tools

[buildenv](#)

Configure gateway software build environment

[xfenv](#)

Configure gateway environment variables

[gwsetup](#)

Configure gateway SD card

[gwcard](#)

Partition SD card , install gateway rootfs

Utilities

[etx](#)

Timestamp converter

[undup](#)

Remove duplicate CSV records, sort file

[repotar](#)

Generate repository snapshot

Utilities

[sdc](#)

Shared data client

[elmostat](#)

Elmo register decoder

[archi](#)

Archive utility

Expert Tools

[gwrn](#)

gateway main executable

[sds](#)

Shared data server

[sbr](#)

Gateway message handler/service broker

[xrun](#)

xsvc script runner

[testdrv](#)

Gateway driver unit tests

[gwgpio](#)

Gateway Beaglebone GPIO control

[cbit, pbit rbit, sbit, tbit,](#)

clear, pulse, read, set, toggle a Beaglebone GPIO bit
See also gwgpio

[xfoce-init-digi](#)

systemd Digi virtual serial port initialization

[xfoce-init-gpio](#)

systemd GPIO initialization

[xfoce-init-autostart](#)

systemd gateway autostart service
XFHOME/src/scripts

[socat-cn](#)

SystemV socat virtual serial port initialization for central node

xsvc

```
usage: xsvc [start|stop|show|help] <service> [args]
  service  : sds sbr gw gwt
  args     : arguments for service
  -l       : log directory (also passed to gateway)
  -V       : verbose output
  -h       : print this help message
```

Description: [Gateway start utility]

```
usage: gwstart [options]
```

Options:

```
-G           : init logs
-i <id>      : gateway ID (e.g. gw0, gw1, gw2, test; default is hostname)
-I          : start virtual serial IO
-l          : list configurations and exit
-L          : list configurations and continue
-M <spec>    : MVC relay init
-N <spec>    : sensor node power
-P <cfg>     : power up (same as -IGS <cfg> -N all:H -M
-S <cfg>     : start gateway
-T          : test
-V          : verbose output
-h          : print use message
```

Common uses:

Start up system after reboot:

```
${XFHOME}/src/scripts/gwstart -P <cfg>
```

Move (gw/sds/sbr) logs aside (leaves logs of running components):

```
gwstart -G
```

Set sensor node power (not allowed if gw is running):

```
gwstart -N <spec> [see also gwdio -h]
```

Start virtual serial IO:

```
${XFHOME}/src/scripts/gwstart -Ii <id>
```

Description: [Gateway shutdown utility]

usage: gwstop [options]

Options:

- i <id> : gateway ID (e.g. gw0, gw1, gw2, test; default is hostname)
- I : stop virtual serial IO
- P : power down gateway (same as -SHN all:L) [WARNING: requires manual gateway restart]
- S : stop gateway
- N <spec> : set sensor node power [e.g. all:L 0:H,2:L etc.]
- R : reboot gateway [WARNING: requires manual gateway restart]
- H : halt gateway [WARNING: requires manual power cycle to restart]
- l : list configurations and exit
- L : list configurations and continue
- T : test
- V : verbose output
- h : print use message

Common uses:

Prepare for power cycle [WARNING: requires manual power cycle]:

sudo \${XFHOME}/src/scripts/gwstop -P

Stop gateway, leave sensor nodes powered:

gwstop -S

Reboot gateway:

\${XFHOME}/src/scripts/gwstop -R

Stop virtual serial port:

gwstop -Ii <id>

gq

Usage: gq <-a host> [options...] <item> [<item>...]

options:

-a : proxy host addr [localhost]
-p : proxy host port [5555]
-t : timeout (msec) [5000]
-r : retries [3]

-q <op> : Quick Query
 topics[,<topic>] : Topic List
 svstat : Service Status
 svprop[,<service>] : Service Properties
 ioprop[,<filter>,...] : IO Properties
 saprop : Sampling Properties
 debug : Debug Settings

-s <args> : set configuration value

-d <args> : do something
 <topic>=<value>[,...]

<item> : info request items (e.g. /platform/ [<sub-item>])
 platform=[list, all, summary, status, config]
 service=[all,summary,details]
 experiment=[all,summary]service summary

-v : verbose output [N]
-D : debug output [N]
-h : this help message

digicfg

usage: digicfg

-a <ip_addr> : hostname/ip addr [134.89.91.3]
-c <cmd> : command [init]
 (init|start|stop|uninit)
-i <id> : ID (port name suffix) [SW]
-o <own_id> : owner uid [500]
-m <mode> : file permissions [664]
-g <grp_id> : argument with option [20]
-V : verbose output [FALSE]
-h : print this help message

xpres

Usage: xpres [options...] [< config_file]

options:

- s: source ID
- t: stream ID
- b: begin time (YYYY/MM/DDTHH:MM:SS)
- e: end time (YYYY/MM/DDTHH:MM:SS)
- B: begin time (epoch sec)
- E: end time (epoch sec)
- a: archive directory
- n: skip interval
- o: output format

e.g., -o <source>,<stream>,"<format specifiers>"[,<mnemonics>...]
where

<source>,<stream> : source or stream ID or -1 for all

<format specifiers> : printf specifiers

<mnemonics> : mnemonics for data components (see below)

Note:

Quotation marks and escaped characters on the command line must be double escaped:

xpres -a arch -s 300 -o 300,-1,"%lld,%lld\\n\\",ets,seq

-F: output time format (see strftime, strptime)

-D: debug output

-V: verbose output

-h: print help message

Data Component Mnemonics:

ets: epoch time msec	[%llu]
etm: epoch time sec	[%llu]
etf: epoch time fraction	[%d] (3-digit msec)
ftu: local time/date	[%s] (specified by time format using -f option; see strftime)
ftl: UTC time/date	[%s] (specified by time format using -f option; see strftime)
utm: UTC time/date	[%s] (specified by time format using -f option; see strftime)
asc: data buffer ASCII	[%s]
pas: data buffer ASCII characters)	[%s] (printable ASCII only, ignore control characters)
hex: data buffer hex	[%s]
bin: data buffer binary	[%s]
src: source ID	[%u]
pid: platform ID	[%u]
str: stream ID	[%u]
stn: stream ID name	[%u]
seq: sequence number	[%llu]
dln: data length	[%llu]
dsz: data record size	[%llu]

skim

usage: skim

-H <hour> : hours [0]
-M <min> : minutes [0]
-S <sec> : seconds [0]
-s <id> : source id
-t <id> : stream id
-a <arch> : arch directory [-a /home/headley/host/hg/xfoce-31mar2014/xfoce/
arch]
-f <cfg> : config file
-V : verbose output [FALSE]
-h : print this help message

gist

Usage: gist [-ahst]

options:

-a: archive directory
-h: print help message
-s: source ID
-t: stream ID

qplot

qplot: Generate plots from one or more delimited text files

usage: qplot [options] file...

Options:

-D	: debug (keep temporary files)	[]
-C <files>	: combine pdf files (see -o)	[]
-a <path>	: archive directory	[]
-j <fmt>	: output type pdf,png	[png]
-o <ofile>	: output file	[qplot_out]
-O <dir>	: output directory	[./plot-output]
-r <min Y>	: y-range min	[*]
-R <max Y>	: y-range max	[*]
-i <min X>	: x-range min	[*]
-I <max X>	: x-range max	[*]
-p <ptype>	: point type	[7]
-P <psize>	: point size	[0.3]
-T <title>	: plot title	[]
-Y <title>	: Y axis title	[]
-X <title>	: X axis title	["Timestamp (epoch sec)"]
-t <fmt>	: time input format	["%s"]
-x <fmt>	: x axis (time) format	["%m/%d/%y %H:%M:%S"]
-s <sep>	: data separator	[,]
-y <pspec>	: file,xcol,N,col-1,title-1,...col-N,title-N	
-V	: verbose output	[FALSE]
-h	: print this help message	

Tip: store command line options in a file
and pipe/redirect input:

```
qplot < my_plot_cfg  
or  
cat my_plot_cfg | qplot
```

qsee

batch process quick look plots

usage: qsee [options] plots...

```
-f <cfg>      : config file      []
-G <dir>      : log file dir      []
-a <arch>     : arch path         []
-o <dir>      : output directory  []
-q <dir>      : qplot output path []
-p <file>     : plot config
-x <file>     : xpres config      []
-B <ets>      : begin time (epoch sec)
-E <ets>      : end time
-b <iso>      : begin time (iso yyyy/mm/ddThh:mm:ss)
-e <iso>      : end time (iso)
-l <n>        : last n hours
-m <n>        : last n minutes
-t           : test mode (show,don't execute tasks)
-T <task>     : specify task (execute in order specified)
               <type>,opts...
               xdata,src,str[,n=mod,x=xpconf]
               jdata,src,str,alias,[x=xpconf]
               eval,<code>
               plot,qpconf
               cplot,ofile,file,file...
-V           : verbose output    []
-h           : help
```

The easiest way to use qsee is to create config files for qsee, xpres and qplot; each specifying the appropriate command line options (one option/value pair per line). The qsee config file reference the qplot configurations using the -p option.

Then, just invoke qsee like this
qsee < my.conf

Override config file options by passing them on the command line:
qsee -l 24 < my.conf

gwdio

Description: Get/Set Gateway GPIO signals

usage: gwdio [options]

Options:

- r <rlist> : read signal(s)
 <rlist>: <sig>[,<sig>...]
- s <slist> : set signal(s)
 <slist>: <sig>:<state=0|1>[,<sig>:<state>...]
- w <s> : wait <s> msec between switching
- t : test [don't execute commands]
- v : verbose output
- h : print use message

signals:

read:

- node[0-3] : Sensor node power [may also use 0-3]
- eeeprom[0-3] : EEPROM_WP_N
 - env : environmental alarm
 - h2o : water alarm
 - all|a : all signals

set:

- node[0-3]|0-3 : Sensor node power
- all|a : all sensor node power

states:

- H||h|1 : high/ON
- L||l|0 : low/OFF

gwmvc

Usage: gwmvc [options]

Options:

- H <ports> : turn port[s] ON
- L <ports> : turn port[s] OFF
- b <s> : set blinky state [0:off 1:on 2:blink]
- i : get MVC info
- q : query status
- a <n,t> : set adc parms
- g <i,n,t> : set gf parms
- G <mode> : set gf ctrl [0-3|GFC_NORM|GFC_OFF|GFC_ALL|GFC_TEST_ONLY]

- T <dev>[,<term>] : tty port terminal info

where

- <dev > : <device_path>
- <term> : <speed>,<par>,<csz>,<stopbits>

- l : device connection is looped back
- h : print help message
- D <level> : debug level (d,i,w,e,f,[s])
- V : verbose output

Examples:

- gwmvc -T /dev/CN0 -H 0,1 -L 2,3 -b 1 -i -q
- gwmvc -T /dev/USB0,115200,N,8,1 -a 10,10 -g 10,20 -G GFC_NORM -i -q
- gwmvc -T /dev/USB0,115200,N,8,1 -a 10,10 -g 10,20 -G 0 -i -q

gwsn

Usage: gwsn [options]

Options:

- H <ports> : turn port[s] ON
- L <ports> : turn port[s] OFF
- b <s> : set blinky state [0:off 1:on 2:blink]
- i : get sensor node info
- q : query status

-S <i>[,<dev>,<term>] : SensorNode terminal info

where

- <dev > : <device_path>
- <term> : <speed>,<par>,<csz>,<stopbits>
- <i> : resource index

- D <level> : debug level (d,i,w,e,f,[s])
- l : device connection is looped back
- h : print help message
- V : verbose output

Examples:

```
gwsn -S 1 -p 0,38400,E,7,2 -b 2 -i
gwsn -S 1,/dev/ttyUSB0,115200 -p 0,9600 -H 0,2 -L 1,3 -q
```

gwmech

Usage: gwmech [options]

Options:

- s <spec> : set relay states [0:H,3:L...]
- q : query mech status
- S <i>[,<dev>,<term>] : SensorNode terminal info
- p <p>[,<term>] : SensorNode port terminal info
- T <dev>[,<term>] : tty port terminal info

where

- <dev > : <device_path>
- <term> : <speed>,<par>,<csz>,<stopbits>
- <i> : resource index
- <p> : port index [0-3]

- l : device connection is looped back
- h : print help message
- D <level> : debug level (d,i,w,e,f,[s])
- V : verbose output

Examples:

```
gwmech -T /dev/ME00,115200,N,8,1 -r 0/on
gwmech -S 1 -p 0,38400,E,7,2 -q
gwmech -S 1,/dev/ttyUSB0,115200 -p 0,9600 -q
```

gwsapi

Usage: gwsapi [options]

Options:

- D <level> : debug level (d,i,w,e,f)
- h : print help message
- v : verbose output

Examples:

gwsapi -d d

moco

Usage: moco [options]

Options:

- c <counts> : set motor speed counts [-1023:1023]
- q : query motor status
- S <i>[,<dev>,<term>] : SensorNode terminal info
- p <p>[,<term>] : SensorNode port terminal info
- T <dev>[,<term>] : tty port terminal info

where

- <dev > : <device_path>
- <term> : <speed>,<par>,<csize>,<stopbits>
- <i> : resource index
- <p> : port index [0-3]
- l : device connection is looped back
- h : print help message
- D <level> : debug level (d,i,w,e,f,[s])
- V : verbose output

Examples:

```
moco -T /dev/ME00,115200,N,8,1 -q
moco -S 1 -p 0,38400,E,7,2 -c 870
moco -S 1,/dev/ttyUSB0,115200 -p 0,9600 -c 400
```

vaco

Usage: vaco [options]

Options:

- c <v>,<r>,<c>: set valve position (valve[0-1], range[0|1], counts[0-255])
- o <v>,<p> : set valve position (valve[0-1], position[0.0-1.0])
- q : query motor status
- P <msec> : cycle power before start (off time, msec)

- S <i>[,<dev>,<term>] : SensorNode terminal info
- p <p>[,<term>] : SensorNode port terminal info
- T <dev>[,<term>] : tty port terminal info

where

- <dev > : <device_path>
- <term> : <speed>,<par>,<csz>,<stopbits>
- <i> : resource index
- <p> : port index [0-3]

- l : device connection is looped back
- h : print help message
- D <level> : debug level (d,i,w,e,f,[s])
- V : verbose output

Examples:

```
vaco -T /dev/ME00,115200,N,8,1 -o 0,0.25/on
vaco -S 1 -p 0,38400,E,7,2 -c 0,0,128
vaco -S 0,/dev/ttyUSB0,115200 -p 0,9600 -q
```

snTerm

snTerm [-a <slaveAddr>] [-b <baud>] [-d] [-p <modbus port>] [-t <target port>] [-h]

- [-p <port>] Use serial port <port> to communicate with Modbus device
Defaults to COM1
- [-a <slaveAddr>] Communicate with Modbus slave <slaveAddr>, 1-247
Defaults to 1
- [-m <modbus baud>] Set Modbus baud rate. Defaults to 115200
- [-t <target port>] Send/get serial comms to target port <port> (0-3)
Defaults to port 0
- [-b <target baud>] Baud rate for target (virtual) serial port
Defaults to 9600
- [-d <0-3>] Print debug messages
- [-h] Print this message

snTest

snTest [-a <slaveAddr>] [-b <baud>] [-d] [-p <port>] [-h]

- [-p <port>] Use serial port <port> to communicate with Modbus device
Defaults to COM1
- [-a <slaveAddr>] Communicate with Modbus slave <slaveAddr>, 1-247
Defaults to 1
- [-b <baud>] Set Modbus baud rate. Defaults to 115200
- [-d <0-3>] Print debug messages
- [-h] Print this message

mvTest

mvTest [-a <slaveAddr>] [-b <baud>] [-d] [-p <port>] [-h]

- [-p <port>] Use serial port <port> to communicate with Modbus device
Defaults to /dev/ttyS7 (equiv to COM8 in Cygwin)
- [-a <slaveAddr>] Communicate with Modbus slave <slaveAddr>, 1-247
Defaults to 1
- [-b <baud>] Set Modbus baud rate. Defaults to 115200
- [-d] Print debug messages
- [-h] Print this message

subc

Usage: subc [-a <host> -p <port> -s <subtree> -t <key>]

Options:

- a <host> : host address
- p <port> : host service port
- s <subtree> : subscription subtree
- t <topic> : topic filter
- h : print help message
- v : verbose output

Example topics:

- tref
- /data/sbe18.0/ph
- /alarm

subs

Usage: subs

vtest.sh

Usage: vtest.sh [-v] <executable> [options] [> output.file] [&]

xcycle-start

usage: xcycle-start [options] script [arg...]

-w <s> : wait before start (s) [0]
-l <logFile> : log file [<script>-051815-134804.log]
-V : verbose output [FALSE]
-h : print this help message

xcycle-stop

Stop all scripts and sub-scripts running under motorCycle
By default, tries to stop motors before exiting
#

usage: xcycle-stop [-h -k -t -V]

-k : don't kill motors [FALSE]
-t : test mode (echo but do not execute commands)
-h : print this help message
-V : verbose output [FALSE]

xprofile

Display or run motor test profile definitions from a file
#

usage: xprofile [options...] <profileName>

-f <file> : profile data file [/home/headley/host/hg/
xfoce-31mar2014/xfoce/src/scripts/cycle/profile-qfoce.csv]
-a <host> : address/hostname [localhost]
-w <sec> : wait before start
-i : invert profile
-R : prefix commands
-O : postfix commands
-s : show one or all profiles [all]
-T : test mode (echo, don't execute commands)
-h : print this help message
-V : verbose output [FALSE]

qprofile

Description: [Define xFOCE thruster profiles on the fly]

usage: qprofile [options] <motor,speed,delay> [motor,speed,delay>...]

Options:

-t	: test output	
-V	: verbose output	[FALSE]
-h	: print use message	

wuntil

usage: wuntil "YYYY/MM/DD HH:MM:SS"

xfenv

usage: . xfenv [/path/to/XFHOME] (uses `pwd` by default)

buildenv

usage: . buildenv [pclinux|bblinux|mac|oebb|unset]

gwsetup

```
usage: gwsetup [options] -a <hostname> -f <file>
-a <hostname> : gateway hostname
-d <name>      : deployment name          []
-f <file>      : xfoce archive tar.gz
               Creates XFHOME using specified archive
-B            : build distribution
-b           : skip build distribution

-C           : clean bak files           []
-V           : verbose output            [N]
-h           : print help message
```

gwcard

```
usage: gwcard
-i <path>      : image directory
-o <odev>      : output SD Card device
-x <path>      : path to xfoce home [$XFHOME]
-d <file>      : xfoce tar.gz          []
-l <label>     : data volume label    [GWDATA]
-t            : test mode              [N]
-k            : keep mounts            [N]
-v            : verbose output         [FALSE]
-h            : print help message
```

etx

usage: etx [-u] <epoch_sec>

undup

Description: Sort and remove duplicate text records

usage: undup [options]

Options:

-f <file> : verbose output [FALSE]
-t : test, show duplicates, don't change
-v : verbose output [FALSE]
-h : print use message

repotar

usage: repotar <repo-name, e.g. 01jan2015-1200>

makes xfoce-repo-01jan2015-1200.tar.gz

sdcc -h

Usage: sdcc [-hcpt <topic> -a <host_addr> -s <service_port>]

options:

-c : consumer
-p : producer
-t <topic>: topic
-a <addr> : host IP address/name
-s <port> : service (primary) IP port
-S <port> : service (backup) IP port
-u <delay>: get/set delay
-h : print help message
-v : verbose

Usage: archi -a <path> -n <name> -s <id> [options]

Options:

- a <path> : path to archive
- n <name> : device name
- s <id> : device source ID
- h : print help message
- V : verbose output

gwrn

Usage: gwrn [-h][options]

Options:

- a <path> : path to archive directory (default: XFHOME/arch)
- l <path> : path to log directory (default: XFHOME/logs)
- f <file> : configuration file
- x <path> : path to XFHOME (override enviroment setting)
- h : print use message

sds

Usage: sds [options]

Options:

- p : primary
- b : backup
- s <port> : primary port
- S <port> : backup port
- v : verbose

sbr

sbr [-h] | [-d] [-v] [broker url]

- h This help message
 - d Daemon mode.
 - v Verbose output
- broker url defaults to tcp://*:5555

xrun

usage: xrun [-TVhl <log>] <program> [args...]

- l <log> : log file path
- T : test (echo, don't execute)
- V : verbose output
- h : print this help message

testdrv -h

Usage: testdrv [options]

Options:

-t <driver> : driver type

-S <i>[,<dev>,<term>] : SensorNode terminal info

-p <p>[,<term>] : SensorNode port terminal info

-T <dev>[,<term>] : tty port terminal info

where

<dev > : <device_path>

<term> : <speed>,<par>,<csz>,<stopbits>

<i> : snode index

<p> : port index [0-3]

-P <t> : power cycle msec

-l : device connection is looped back

-L : list drivers

-h : help message

-D <c..> : debug level (d,i,w,e,f,[s])

-V : verbose output

Examples:

testdrv -n loop -T /dev/pty0,9600,N,8,1

testdrv -n sbe18s -S 0,/dev/ttyUSB0 -p 0,38400,E,7,2

gwgpio -h

Usage: gwgpio -g <chip, pin> [options]

Options:

-g <chip, pin> : GPIO address

-I : initialize for xfoce

-U : unexport xfoce GPIO

-e : export pin

-l <lh>,<msec> : pulse pin

-u : unexport pin

-a <[l|h]> : set pin active level

-d <[i|o]> : set pin direction

-v <[l|h]> : set pin value

-p : print current values

-t <d,a,v> : toggle one or more parameters

-V : verbose output

-h : print this help message

cbit, pbit rbit, sbit, tbit

cbit clear GPIO bit

sbit set GPIO bit

rbit read GPIO bit

[crs]bit <chip>,<pin>

pbit pulse GPIO bit

pbit <chip>,<pin> <lh>,<msec>

tbit toggle GPIO bit

tbit <chip>,<pin> <a|d|v>

xfoce-init-gpio

Description: xFOCE gateway boot-time GPIO init

usage: xfoce-init-gpio [options]

Options:

-I : Initialize

-U : Un-initialize

-V : verbose output [N]

-h : print use message

xfoce-init-digi

Description: Initialize Digi virtual serial port(s)

usage: xfoce-init-digi [options]

Options:

-I <host>,<id> : Initialize

-U <host>,<id> : Un-initialize

-V : verbose output [N]

-h : print use message

Description: xFOCE gateway autostart

usage: xfoce-init-autostart [options]

Options:

-i <id> : id to pass to gwstart

-U : Up (do autostart, if enabled)

-V : verbose output [N]

-h : print use message

socatd-cn

Description: System V xFOCE Central Node socat virtual serial port service

Usage: /etc/init.d/socatd-cn {start|stop|restart|reload|status}