Testing Level Shifter Boards

Arun Tadepalli
Rutgers University

Introduction

What are level shifter boards?

- Provides voltages to the ASDQ cards
- Receives the signals from ASDQ cards
- Can test ASDQ based from end cards by pulsing them

How to control them?

- We can give commands via telnet
- If we give one command to the master board, then all the boards which are daisy chained can receive the signal.
- Powered by 24 V DC power supply
- Florian's slides on readout electronics is a good source of reference

Parts

- Master board connected with blue Ethernet cable
- Other boards daisy chained to the master board
- Send port and receive ports on all level shifter board

Ribbon cables that connect the ASDQ cards and Level shifter boards

Master board connected with a blue Ethernet cable

Trigger

Send port

Receive port

Daisy chaining of level shifter boards

How to connect to LSB?

- We can talk to the Level Shifter Boards using telnet

```
[arunts@e906-gat2 ~]$ telnet 192.168.24.61 5002
Trying 192.168.24.61...
Connected to levelshift1.fnal.gov (192.168.24.61).
Escape character is '^]'.
```

- Level shifter boards listen to ports 5000, 5001, 5002
- Example 192.168.24.6x 5000 where x = 1 – 4 for St 1, 2, 3+, 3-

192.168.24.61 5001
192.168.24.61 5002
192.168.24.61 5003

All used to connect to St on different ports

If one of the ports do not connect, keep trying other ports

Help command (he)

he command displays all the commands available to use telnet

```
he

SeaQuest Level Adapter Command List
Main
 WT ch d     Write DAC0 Threshold, ch-0-7 (8=all),  d(0-4095),(0-4095mV @ DAC0)
 WC ch d     Write DAC1 ch0=IBLR, 1=TREFE, 2=TREF0, d(0-4095),(0-4095mV @ DAC1)
 ID n        Display Network Setup and Firmware Ver,(n=1 info)(n=0 Add hBus ##)
 MO n        Monitor, Temperature and ADC Channels. (n=1 info)(n=0 Add hBus ##)
 RDAC n      Read DACs (shadow), data as cnts(dec), (n=1 info)(n=0 Add hBus ##)
RJ11 Link
 HC b cmd    hBus command, b=(brd 1-20, 0=all), cmd(any valid command)
 HI          hBus Initialize, returns board count, (cmd 'HM' does this)
 HM          hBus sets board to hBus Master (reset heart beat LED rate)
 HS n        hBus status information, (n=1 info)
 H3V d       hBus Enable -3Volt power, (d=1=ON) (0 OFF)
Socket
 SET         Set Network Regs, Sockets, Ports, and serial number (SAVE)
 SOCK        Display Network Socket Status, Enabled 0-2, Disabled 3-7
 QUIT s      Close Network Socket, s=socket 0-2, if no param close this sock
Flash
 DREC s      Device Recall, from Flash,(s=Setup page 0-10, 0=Constants)
 DSAV s      Device Save,   to Flash,  (s=Setup page 1-10, 1=Boot page)
 FL          USB Flash Load (to ATMEL MEMORY) using Altera Binary File(*.rbf)
 FD          USB FPGA direct load over USB using Altera Binary File(*.rbf)
 FMR page    Flash Memory reads (DSAV page 1,,10), 11=Network, 0=Consants
 FMR adr 1   Flash Memory Reads 256 bytes ATMEL FLASH, (Addr(D) * 256)
Misc
 RD adr      Read Addr (hex even addr), returns data16(H)
 WR a d      Wr Addr(hex even addr), writes d=Data16(H)
 Help        HE and H1
 RESET       Hardware Reset of Ethernet, FPGA, then soft restart
 USB         If USB fails after uC reboot. Disconnect/Connect USB cable
```

Commands display h1

```
h1
SeaQuest Level Adapter FPGA Address Map, BaseAddr(0xA0000000)
  OFFSET     Description
   FPGA TEST
    0x0      Test Reg Hi/Lo byte, for r/w tests
    0x6      Pulser, ~1Hz per count. (0=stop)
   FPGA DAC0
    0x0A     DAC0 Hi/Lo byte of 8-ch Threshold dac, data= nXXX (n=0-7)
             DAC0 data as 0340,1340,2340,3340,4340,5340,6340,7340
   FPGA DAC1
    0x0B     DAC1 Hi/Lo byte 4-ch dac, d= nXXX, n=2(IBLR),6(TREFE),A(TREF0)
             DAC1 data as 2930,6C00,AC00

    0x0D     Link CSR(8..0)  txf rfff rffe dcrf dcre prtBsy dsWr dsErr bErr
    0x0E     Link ADDR(3..0) dir 3vEna term dir(1=xmt)
```

Some helpful commands

Hm command makes the board connected with Ethernet as the buss master and also returns the number of boards connected to the master (1-N, with 1 furthest from masterboard & N closest).

```
hm
hBus Master, (reset to exit this mode), Sending hBus Init
Hm
hBus Master, (reset to exit this mode), Sending hBus Init
hBusCfg: BrdCnt= 11
```

No of boards connected to the level shifter crate

```
hc 0 rdac
0099 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0001 2400 2400 2400 2400 2400 2400 2400 2400 2700 3072 3072
0003 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0002 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0004 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0005 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0006 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0007 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0008 1400 1400 1400 1400 1400 1400 1400 1400 2700 3072 3072
0009 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0010 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0011 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
```

Some commands

hc is the HAPI command.. Basically we can use this command to make changes, read out thresholds, voltages, etc..

```
hc 0 rdac
0099 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0001 2400 2400 2400 2400 2400 2400 2400 2400 2700 3072 3072
0003 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0002 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0004 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0005 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0006 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0007 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0008 1400 1400 1400 1400 1400 1400 1400 1400 2700 3072 3072
0009 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0010 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0011 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
```

Hc command

0 to talk to include all the boards

8 readout points on the level shifter board

12 boards including the master board

St 1 LSB testing

```
q
Connection closed by foreign host.
[arunts@e906-gat2 ~]$ telnet 192.168.24.61 5001
Trying 192.168.24.61...
Connected to levelshift1.fnal.gov (192.168.24.61).
Escape character is '^]'.
hm
hBus Master, (reset to exit this mode), Sending hBus Init
hBusCfg: BrdCnt= 17
```

17 boards connected

Hc 0 rdac returns the threshold at all the check points

99 is masterboard

```
hBusCfg: BrdCnt= 17
hc 0 rdac
0099 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0001 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0002 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0003 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0004 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0005 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0006 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0007 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0008 1400 1400 1400 1400 1400 1400 1400 1400 2700 3072 3072
0009 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0010 1200 1500 1200 1200 1200 1200 1200 1200 2700 3072 3072
0011 1200 1200 1200 1400 1200 1200 1200 1200 2700 3072 3072
0012 1500 1500 1500 1500 1500 1500 1200 1200 2700 3072 3072
0013 1200 1200 1200 1200 1200 1200 1500 1500 2700 3072 3072
0014 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0015 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0016 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
0017 1200 1200 1200 1200 1200 1200 1200 1200 2700 3072 3072
```

hc 0 rdac 0 command returns the voltages at each of the 8 checkpoints and the applied threshold to all these checkpoints

```
hc 0 rdac 0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
```

```
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
```

```
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1400 cnts  ~ 350mV @ card
  ch1=1400 cnts  ~ 350mV @ card
  ch2=1400 cnts  ~ 350mV @ card
  ch3=1400 cnts  ~ 350mV @ card
  ch4=1400 cnts  ~ 350mV @ card
  ch5=1400 cnts  ~ 350mV @ card
  ch6=1400 cnts  ~ 350mV @ card
  ch7=1400 cnts  ~ 350mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
```

```
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1500 cnts  ~ 375mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1400 cnts  ~ 350mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
```

```
dac_0 registers
  ch0=1200 cnts  ~ 300mV @ card
  ch1=1200 cnts  ~ 300mV @ card
  ch2=1200 cnts  ~ 300mV @ card
  ch3=1200 cnts  ~ 300mV @ card
  ch4=1200 cnts  ~ 300mV @ card
  ch5=1200 cnts  ~ 300mV @ card
  ch6=1200 cnts  ~ 300mV @ card
  ch7=1200 cnts  ~ 300mV @ card
dac_1 registers
  ch0=2700 cnts  IBLR
  ch1=3072 cnts  TREFE
  ch2=3072 cnts  TREF0
```

17 such readouts for all the seventeen boards for St 1

Hc 0 mo 0 command returns all the monitored voltages and temperatures at the readout points.

Fuse warning but it is still ok since there is a voltage on it. If it blows out, one has to go and solder a new element. That's it!

If the fuse is blown out, it will give a warning with 0.00 v reading

```
hc 0 mo 0
 TempC     = 26.0
 5.0v__Pos=  4.94v          (Ch00) 558 *  8.86mV/bit
 3.3v__Pos=  2.84v FUSE     (Ch01) 480 *  5.92mV/bit
 0.85v_Pos=  0.72v FUSE     (Ch04) 241 *  2.98mV/bit
 5.0v__Neg=  5.01v          (Ch02) 420 * 11.94mV/bit
 3.3vA_Neg=  2.85v FUSE     (Ch05) 319 *  8.92mV/bit
 3.3vB_Neg=  2.85v FUSE     (Ch14) 320 *  8.92mV/bit
 3.3v_Digi=  3.31v          (Ch15) 562 *  5.89mV/bit
 1.2v_Digi=  1.22v          (Ch03) 406 *  3.00mV/bit
 TempC     = 28.0
 5.0v__Pos=  4.97v          (Ch00) 561 *  8.86mV/bit
 3.3v__Pos=  2.86v FUSE     (Ch01) 483 *  5.92mV/bit
 0.85v_Pos=  0.72v FUSE     (Ch04) 243 *  2.98mV/bit
 5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
 3.3vA_Neg=  2.77v FUSE     (Ch05) 311 *  8.92mV/bit
 3.3vB_Neg=  2.85v FUSE     (Ch14) 320 *  8.92mV/bit
 3.3v_Digi=  3.30v          (Ch15) 560 *  5.89mV/bit
 1.2v_Digi=  1.21v          (Ch03) 403 *  3.00mV/bit
 TempC     = 28.0
 5.0v__Pos=  4.97v          (Ch00) 561 *  8.86mV/bit
 3.3v__Pos=  2.75v FUSE     (Ch01) 465 *  5.92mV/bit  Warning
 0.85v_Pos=  0.68v FUSE     (Ch04) 228 *  2.98mV/bit
 5.0v__Neg=  5.01v          (Ch02) 420 * 11.94mV/bit
 3.3vA_Neg=  2.73v FUSE     (Ch05) 306 *  8.92mV/bit
 3.3vB_Neg=  2.85v FUSE     (Ch14) 319 *  8.92mV/bit
 3.3v_Digi=  3.33v          (Ch15) 565 *  5.89mV/bit
 1.2v_Digi=  1.23v          (Ch03) 410 *  3.00mV/bit
 TempC     = 29.0
 5.0v__Pos=  4.98v          (Ch00) 562 *  8.86mV/bit
 3.3v__Pos=  2.85v FUSE     (Ch01) 482 *  5.92mV/bit
 0.85v_Pos=  0.71v FUSE     (Ch04) 237 *  2.98mV/bit
 5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
 3.3vA_Neg=  2.85v FUSE     (Ch05) 320 *  8.92mV/bit
 3.3vB_Neg=  2.86v FUSE     (Ch14) 321 *  8.92mV/bit
 3.3v_Digi=  3.33v          (Ch15) 565 *  5.89mV/bit
 1.2v_Digi=  1.21v          (Ch03) 402 *  3.00mV/bit
 TempC     = 24.0
```

Readout for St 2 – same procedure

```
hc 0 mo 0
 TempC    = 26.0
 5.0v__Pos=  4.94v          (Ch00) 558 *  8.86mV/bit
 3.3v__Pos=  2.84v FUSE     (Ch01) 480 *  5.92mV/bit
 0.85v_Pos=  0.72v FUSE     (Ch04) 241 *  2.98mV/bit
 5.0v__Neg=  5.01v          (Ch02) 420 * 11.94mV/bit
 3.3vA_Neg=  2.85v FUSE     (Ch05) 319 *  8.92mV/bit
 3.3vB_Neg=  2.85v FUSE     (Ch14) 320 *  8.92mV/bit
 3.3v_Digi=  3.31v          (Ch15) 562 *  5.89mV/bit
 1.2v_Digi=  1.22v          (Ch03) 406 *  3.00mV/bit
 TempC    = 28.0
 5.0v__Pos=  4.97v          (Ch00) 561 *  8.86mV/bit
 3.3v__Pos=  2.86v FUSE     (Ch01) 483 *  5.92mV/bit
 0.85v_Pos=  0.72v FUSE     (Ch04) 243 *  2.98mV/bit
 5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
 3.3vA_Neg=  2.77v FUSE     (Ch05) 311 *  8.92mV/bit
 3.3vB_Neg=  2.85v FUSE     (Ch14) 320 *  8.92mV/bit
 3.3v_Digi=  3.30v          (Ch15) 560 *  5.89mV/bit
 1.2v_Digi=  1.21v          (Ch03) 403 *  3.00mV/bit
 TempC    = 28.0
 5.0v__Pos=  4.97v          (Ch00) 561 *  8.86mV/bit
 3.3v__Pos=  2.75v FUSE     (Ch01) 465 *  5.92mV/bit  Warning
 0.85v_Pos=  0.68v FUSE     (Ch04) 228 *  2.98mV/bit
 5.0v__Neg=  5.01v          (Ch02) 420 * 11.94mV/bit
 3.3vA_Neg=  2.73v FUSE     (Ch05) 306 *  8.92mV/bit
 3.3vB_Neg=  2.85v FUSE     (Ch14) 319 *  8.92mV/bit
 3.3v_Digi=  3.33v          (Ch15) 565 *  5.89mV/bit
 1.2v_Digi=  1.23v          (Ch03) 410 *  3.00mV/bit
 TempC    = 29.0
 5.0v__Pos=  4.98v          (Ch00) 562 *  8.86mV/bit
 3.3v__Pos=  2.85v FUSE     (Ch01) 482 *  5.92mV/bit
 0.85v_Pos=  0.71v FUSE     (Ch04) 237 *  2.98mV/bit
 5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
 3.3vA_Neg=  2.85v FUSE     (Ch05) 320 *  8.92mV/bit
 3.3vB_Neg=  2.86v FUSE     (Ch14) 321 *  8.92mV/bit
 3.3v_Digi=  3.33v          (Ch15) 565 *  5.89mV/bit
 1.2v_Digi=  1.21v          (Ch03) 402 *  3.00mV/bit
 TempC    = 24.0
```

```
TempC    = 27.0
5.0v__Pos=  4.87v          (Ch00) 550 *  8.86mV/bit
3.3v__Pos=  2.99v FUSE     (Ch01) 505 *  5.92mV/bit
0.85v_Pos=  0.82v FUSE     (Ch04) 275 *  2.98mV/bit
5.0v__Neg=  4.93v          (Ch02) 413 * 11.94mV/bit
3.3vA_Neg=  2.93v FUSE     (Ch05) 328 *  8.92mV/bit
3.3vB_Neg=  2.94v FUSE     (Ch14) 330 *  8.92mV/bit
3.3v_Digi=  3.27v          (Ch15) 555 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 403 *  3.00mV/bit
TempC    = 28.0
5.0v__Pos=  4.98v          (Ch00) 562 *  8.86mV/bit
3.3v__Pos=  3.01v FUSE     (Ch01) 509 *  5.92mV/bit
0.85v_Pos=  0.83v FUSE     (Ch04) 280 *  2.98mV/bit
5.0v__Neg=  5.00v          (Ch02) 419 * 11.94mV/bit
3.3vA_Neg=  2.98v FUSE     (Ch05) 334 *  8.92mV/bit
3.3vB_Neg=  2.98v FUSE     (Ch14) 334 *  8.92mV/bit
3.3v_Digi=  3.31v          (Ch15) 562 *  5.89mV/bit
1.2v_Digi=  1.20v          (Ch03) 399 *  3.00mV/bit
TempC    = 22.0
5.0v__Pos=  4.94v          (Ch00) 558 *  8.86mV/bit
3.3v__Pos=  2.82v FUSE     (Ch01) 477 *  5.92mV/bit
0.85v_Pos=  0.71v FUSE     (Ch04) 239 *  2.98mV/bit
5.0v__Neg=  4.96v          (Ch02) 415 * 11.94mV/bit
3.3vA_Neg=  2.81v FUSE     (Ch05) 315 *  8.92mV/bit
3.3vB_Neg=  2.84v FUSE     (Ch14) 318 *  8.92mV/bit
3.3v_Digi=  3.32v          (Ch15) 563 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 402 *  3.00mV/bit
TempC    = 27.0
5.0v__Pos=  5.00v          (Ch00) 564 *  8.86mV/bit
3.3v__Pos=  2.84v FUSE     (Ch01) 480 *  5.92mV/bit
0.85v_Pos=  0.73v FUSE     (Ch04) 246 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.87v FUSE     (Ch05) 322 *  8.92mV/bit
3.3vB_Neg=  2.92v FUSE     (Ch14) 327 *  8.92mV/bit
3.3v_Digi=  3.33v          (Ch15) 566 *  5.89mV/bit
1.2v_Digi=  1.22v          (Ch03) 406 *  3.00mV/bit
TempC    = 24.0
5.0v__Pos=  4.98v          (Ch00) 562 *  8.86mV/bit
3.3v__Pos=  2.83v FUSE     (Ch01) 478 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 243 *  2.98mV/bit
5.0v__Neg=  5.07v          (Ch02) 425 * 11.94mV/bit
3.3vA_Neg=  2.87v FUSE     (Ch05) 322 *  8.92mV/bit
3.3vB_Neg=  2.88v FUSE     (Ch14) 323 *  8.92mV/bit
3.3v_Digi=  3.30v          (Ch15) 561 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 404 *  3.00mV/bit
```

```
TempC    = 26.0
5.0v__Pos=  4.97v          (Ch00) 561 *  8.86mV/bit
3.3v__Pos=  2.85v FUSE     (Ch01) 482 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 240 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.81v FUSE     (Ch05) 315 *  8.92mV/bit
3.3vB_Neg=  2.85v FUSE     (Ch14) 319 *  8.92mV/bit
3.3v_Digi=  3.33v          (Ch15) 565 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 403 *  3.00mV/bit
TempC    = 26.0
5.0v__Pos=  4.95v          (Ch00) 559 *  8.86mV/bit
3.3v__Pos=  2.82v FUSE     (Ch01) 477 *  5.92mV/bit
0.85v_Pos=  0.71v FUSE     (Ch04) 239 *  2.98mV/bit
5.0v__Neg=  5.00v          (Ch02) 419 * 11.94mV/bit
3.3vA_Neg=  2.83v FUSE     (Ch05) 317 *  8.92mV/bit
3.3vB_Neg=  2.85v FUSE     (Ch14) 320 *  8.92mV/bit
3.3v_Digi=  3.28v          (Ch15) 557 *  5.89mV/bit
1.2v_Digi=  1.20v          (Ch03) 399 *  3.00mV/bit
TempC    = 25.0
5.0v__Pos=  5.02v          (Ch00) 567 *  8.86mV/bit
3.3v__Pos=  2.89v FUSE     (Ch01) 489 *  5.92mV/bit
0.85v_Pos=  0.75v FUSE     (Ch04) 250 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.69v FUSE     (Ch05) 302 *  8.92mV/bit
3.3vB_Neg=  2.93v FUSE     (Ch14) 328 *  8.92mV/bit
3.3v_Digi=  3.36v          (Ch15) 570 *  5.89mV/bit
1.2v_Digi=  1.23v          (Ch03) 411 *  3.00mV/bit
TempC    = 26.0
5.0v__Pos=  4.92v          (Ch00) 555 *  8.86mV/bit
3.3v__Pos=  2.82v FUSE     (Ch01) 477 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 240 *  2.98mV/bit
5.0v__Neg=  4.94v          (Ch02) 414 * 11.94mV/bit
3.3vA_Neg=  2.80v FUSE     (Ch05) 314 *  8.92mV/bit
3.3vB_Neg=  2.81v FUSE     (Ch14) 315 *  8.92mV/bit
3.3v_Digi=  3.28v          (Ch15) 557 *  5.89mV/bit
1.2v_Digi=  1.19v          (Ch03) 397 *  3.00mV/bit
TempC    = 26.0
5.0v__Pos=  4.92v          (Ch00) 555 *  8.86mV/bit
3.3v__Pos=  2.83v FUSE     (Ch01) 478 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 241 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.79v FUSE     (Ch05) 313 *  8.92mV/bit
3.3vB_Neg=  2.83v FUSE     (Ch14) 317 *  8.92mV/bit
3.3v_Digi=  3.29v          (Ch15) 559 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 403 *  3.00mV/bit
```

Blown out fuse!

```
hc 17 mo 0
 TempC     = 34.0
 5.0v__Pos=  4.98v          (Ch00) 562 *  8.86mV/bit
 3.3v__Pos=  3.20v FUSE     (Ch01) 540 *  5.92mV/bit
 0.85v_Pos=  0.77v FUSE     (Ch04) 257 *  2.98mV/bit
 5.0v__Neg=  5.07v          (Ch02) 425 * 11.94mV/bit
 3.3vA_Neg=  3.11v FUSE     (Ch05) 349 *  8.92mV/bit
 3.3vB_Neg=  0.00v FUSE     (Ch14)   0 *  8.92mV/bit  Warning
 3.3v_Digi=  3.31v          (Ch15) 562 *  5.89mV/bit
 1.2v_Digi=  1.21v          (Ch03) 404 *  3.00mV/bit
mo
0099 30.0  5.01  3.14  0.68  4.98  3.12  3.11  3.35  1.21
hc 17 mo
0017 34.0  4.99  3.21  0.77  5.09  3.12  0.00  3.31  1.22
```

If fuse blows out, then it should show 0.00 v

Note blown fuse on board 17!

Taken from David Christian's command summary

We could be selective about which board we would like to look at.

```
hc 1 mo 1
 TempC     = 28.0
 5.0v__Pos=  4.98v          (Ch00) 562 *  8.86mV/bit
 3.3v__Pos=  2.87v FUSE     (Ch01) 484 *  5.92mV/bit
 0.85v_Pos=  0.72v FUSE     (Ch04) 242 *  2.98mV/bit
 5.0v__Neg=  5.04v          (Ch02) 422 * 11.94mV/bit
 3.3vA_Neg=  2.82v FUSE     (Ch05) 316 *  8.92mV/bit
 3.3vB_Neg=  2.84v FUSE     (Ch14) 318 *  8.92mV/bit
 3.3v_Digi=  3.33v          (Ch15) 566 *  5.89mV/bit
 1.2v_Digi=  1.22v          (Ch03) 405 *  3.00mV/bit
```

Hc 0 mo 0 for St2

```
TempC    = 22.0
5.0v__Pos=  4.94v          (Ch00) 558 *  8.86mV/bit
3.3v__Pos=  2.75v FUSE     (Ch01) 465 *  5.92mV/bit  Warning
0.85v_Pos=  0.73v FUSE     (Ch04) 245 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.80v FUSE     (Ch05) 314 *  8.92mV/bit
3.3vB_Neg=  2.85v FUSE     (Ch14) 319 *  8.92mV/bit
3.3v_Digi=  3.32v          (Ch15) 564 *  5.89mV/bit
1.2v_Digi=  1.20v          (Ch03) 401 *  3.00mV/bit
TempC    = 26.0
5.0v__Pos=  5.01v          (Ch00) 565 *  8.86mV/bit
3.3v__Pos=  2.90v FUSE     (Ch01) 490 *  5.92mV/bit
0.85v_Pos=  0.75v FUSE     (Ch04) 253 *  2.98mV/bit
5.0v__Neg=  5.01v          (Ch02) 420 * 11.94mV/bit
3.3vA_Neg=  2.91v FUSE     (Ch05) 326 *  8.92mV/bit
3.3vB_Neg=  2.85v FUSE     (Ch14) 320 *  8.92mV/bit
3.3v_Digi=  3.33v          (Ch15) 566 *  5.89mV/bit
1.2v_Digi=  1.23v          (Ch03) 409 *  3.00mV/bit
TempC    = 30.0
5.0v__Pos=  5.02v          (Ch00) 567 *  8.86mV/bit
3.3v__Pos=  2.87v FUSE     (Ch01) 484 *  5.92mV/bit
0.85v_Pos=  0.73v FUSE     (Ch04) 245 *  2.98mV/bit
5.0v__Neg=  5.04v          (Ch02) 422 * 11.94mV/bit
3.3vA_Neg=  2.88v FUSE     (Ch05) 323 *  8.92mV/bit
3.3vB_Neg=  2.91v FUSE     (Ch14) 326 *  8.92mV/bit
3.3v_Digi=  3.34v          (Ch15) 567 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 404 *  3.00mV/bit
```

```
hc 0 mo 0
TempC    = 28.0
5.0v__Pos=  4.94v          (Ch00) 558 *  8.86mV/bit
3.3v__Pos=  2.84v FUSE     (Ch01) 479 *  5.92mV/bit
0.85v_Pos=  0.70v FUSE     (Ch04) 234 *  2.98mV/bit
5.0v__Neg=  4.93v          (Ch02) 413 * 11.94mV/bit
3.3vA_Neg=  2.77v FUSE     (Ch05) 311 *  8.92mV/bit
3.3vB_Neg=  2.86v FUSE     (Ch14) 321 *  8.92mV/bit
3.3v_Digi=  3.32v          (Ch15) 563 *  5.89mV/bit
1.2v_Digi=  1.22v          (Ch03) 406 *  3.00mV/bit
TempC    = 28.0
5.0v__Pos=  4.99v          (Ch00) 563 *  8.86mV/bit
3.3v__Pos=  2.87v FUSE     (Ch01) 484 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 241 *  2.98mV/bit
5.0v__Neg=  5.04v          (Ch02) 422 * 11.94mV/bit
3.3vA_Neg=  2.82v FUSE     (Ch05) 316 *  8.92mV/bit
3.3vB_Neg=  2.84v FUSE     (Ch14) 318 *  8.92mV/bit
3.3v_Digi=  3.34v          (Ch15) 567 *  5.89mV/bit
1.2v_Digi=  1.22v          (Ch03) 406 *  3.00mV/bit
TempC    = 25.0
__Pos=  4.93v          (Ch00) 556 *  8.86mV/bit
__Pos=  2.79v FUSE     (Ch01) 471 *  5.92mV/bit  Warning
v_Pos=  0.70v FUSE     (Ch04) 236 *  2.98mV/bit
__Neg=  4.92v          (Ch02) 412 * 11.94mV/bit
A_Neg=  2.77v FUSE     (Ch05) 311 *  8.92mV/bit
B_Neg=  2.83v FUSE     (Ch14) 317 *  8.92mV/bit
_Digi=  3.27v          (Ch15) 555 *  5.89mV/bit
_Digi=  1.22v          (Ch03) 405 *  3.00mV/bit
C    = 30.0
__Pos=  4.96v          (Ch00) 560 *  8.86mV/bit
__Pos=  2.86v FUSE     (Ch01) 483 *  5.92mV/bit
v_Pos=  0.71v FUSE     (Ch04) 238 *  2.98mV/bit
__Neg=  4.99v          (Ch02) 418 * 11.94mV/bit
A_Neg=  2.80v FUSE     (Ch05) 314 *  8.92mV/bit
B_Neg=  2.81v FUSE     (Ch14) 315 *  8.92mV/bit
```

```
TempC    = 24.0
5.0v__Pos=  4.94v          (Ch00) 557 *  8.86mV/bit
3.3v__Pos=  2.88v FUSE     (Ch01) 486 *  5.92mV/bit
0.85v_Pos=  0.73v FUSE     (Ch04) 246 *  2.98mV/bit
5.0v__Neg=  5.00v          (Ch02) 419 * 11.94mV/bit
3.3vA_Neg=  2.91v FUSE     (Ch05) 326 *  8.92mV/bit
3.3vB_Neg=  2.83v FUSE     (Ch14) 317 *  8.92mV/bit
3.3v_Digi=  3.27v          (Ch15) 555 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 402 *  3.00mV/bit
TempC    = 26.0
5.0v__Pos=  4.96v          (Ch00) 560 *  8.86mV/bit
3.3v__Pos=  2.82v FUSE     (Ch01) 477 *  5.92mV/bit
0.85v_Pos=  0.71v FUSE     (Ch04) 239 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.81v FUSE     (Ch05) 315 *  8.92mV/bit
3.3vB_Neg=  2.84v FUSE     (Ch14) 318 *  8.92mV/bit
3.3v_Digi=  3.34v          (Ch15) 567 *  5.89mV/bit
1.2v_Digi=  1.21v          (Ch03) 404 *  3.00mV/bit
TempC    = 31.0
5.0v__Pos=  5.02v          (Ch00) 567 *  8.86mV/bit
3.3v__Pos=  2.85v FUSE     (Ch01) 482 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 241 *  2.98mV/bit
5.0v__Neg=  5.06v          (Ch02) 424 * 11.94mV/bit
3.3vA_Neg=  2.85v FUSE     (Ch05) 320 *  8.92mV/bit
3.3vB_Neg=  2.92v FUSE     (Ch14) 327 *  8.92mV/bit
3.3v_Digi=  3.36v          (Ch15) 571 *  5.89mV/bit
1.2v_Digi=  1.22v          (Ch03) 407 *  3.00mV/bit
TempC    = 27.0
5.0v__Pos=  5.02v          (Ch00) 567 *  8.86mV/bit
3.3v__Pos=  2.88v FUSE     (Ch01) 486 *  5.92mV/bit
0.85v_Pos=  0.73v FUSE     (Ch04) 245 *  2.98mV/bit
5.0v__Neg=  5.09v          (Ch02) 426 * 11.94mV/bit
3.3vA_Neg=  2.86v FUSE     (Ch05) 321 *  8.92mV/bit
3.3vB_Neg=  2.90v FUSE     (Ch14) 325 *  8.92mV/bit
3.3v_Digi=  3.38v          (Ch15) 574 *  5.89mV/bit
1.2v_Digi=  1.22v          (Ch03) 406 *  3.00mV/bit
TempC    = 25.0
5.0v__Pos=  4.97v          (Ch00) 561 *  8.86mV/bit
3.3v__Pos=  2.83v FUSE     (Ch01) 478 *  5.92mV/bit
0.85v_Pos=  0.72v FUSE     (Ch04) 240 *  2.98mV/bit
5.0v__Neg=  5.03v          (Ch02) 421 * 11.94mV/bit
3.3vA_Neg=  2.84v FUSE     (Ch05) 318 *  8.92mV/bit
3.3vB_Neg=  2.86v FUSE     (Ch14) 321 *  8.92mV/bit
3.3v_Digi=  3.31v          (Ch15) 562 *  5.89mV/bit
```

Write command

wt ch d (wt 8 1000)

- Write threshold
- Ch=0-7 is ASDQ #; 8 is all
- d is a base-10 number 0-4095
  - Nominal threshold is 832.

```
hc 0 rdac
0099 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0001 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0003 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0002 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0004 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0005 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0006 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0007 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0008 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0009 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0010 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0011 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
```

Before wt command

After wt command with threshold changed for channel 10

```
hc 10 wt 8 2000
hc 0 rdac
0099 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0001 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0003 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0002 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0004 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0005 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0006 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0007 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0008 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0009 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
0010 2000 2000 2000 2000 2000 2000 2000 2000 2700 3072 3072
0011 1600 1600 1600 1600 1600 1600 1600 1600 2700 3072 3072
```

To pulse the LSB

```
hc 3 wr 6 0
```

To pulse channel 3 with some frequency

```
hc 3 wr 6 1000
hc 3 wr 6 0
hc 2 wr 6 1000
hc 2 wr 6 0
hc 2 wr 6 1000
hc 2 wr 6 0
```

1000 Hz pulse to channel 3. To stop it, just repeat the command with a pulse of freq 0 Hz

```
hc 0 DSAV 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
Setup saved to ATMEL Page 1
```

Finally, to save settings

Use of wc command

Telnet commands

- hc 0 wt 8 4000

This command writes a threshold of 4000 to all ASDQ cards

- hc 0 wc 3 0

This command makes the amplitude of all pulses to 0 units

- hc 7 wt 1 2000

This command writes the threshold of LSBID 7 ASDQ ID 1 to a threshold of 2000

- hc 7 wc 1 1000

This command sends a pulse of 1000 units to all even channels of the chosen LSB

wc command is used to send a pulse to a particular Level Shifter board

THE END!

REFERENCES:

- Florian's collaboration meeting slides on read out electronics about LSB and ASDQ cards
- Dave Christians slides on Level shifter command summary