DRAFT August 18, 1998

The FLIP3U QBias card Driver

Steve Furlanetto, Fermilab

1 Introduction

This document describes the basic functionality of the FLIP3U QBias card, and the software tools that perform them. This can be found at
`http://fnphyx-www.fnal.gov/experiments/cdms/software/GPIB/FLIP3UQBias.ps`.

2 Design

The FLIP3U QBias driver performs in conjunction with the `GPIB protocol`. See ref [1] for a detailed description. The basic idea is that commands are issued from a perl script and are sent *via* tcp/ip to a server which issues the GPIB commands to perform the action on the card.

3 Instructions

The following sections list the methods that apply to instances of the FLIP3UQBias class. The program is designed to be used with the `rack.pl` interface program. Instructions for use of this interface can be found in ref [2].

This software also contains a useful persistence feature. Upon executing any command to the GPIB box, it will record the card status in a file called `/tmp/hostname-FLIP3UQBias-subrack-module.out`, where `subrack` is the subrack of the card and `module` is the module number of the card. If `-s` is used, then `hostname` becomes `script`. When `rack.pl` is run again, it will automatically search for this file and reload those values into memory. This feature is designed to ensure that the hardware and software remain synchronized. If the rack is powered down, be sure to re-initialize the settings when the program is run again.

3.1 General Information

help (**String** `topic`)
: Prints the subset of the commands that can be issued to the ZIP card which contain the string `topic`. If no argument, prints all the help information.

print (**void**)
: Prints information about the GPIB box and the card including module number and the status bits of the Control Registers.

cardtype (**void**)
: Prints card type that computer believes to be correct

saveScript (**String** `filename`, **int** `overwrite`)
: Saves all subsequent GPIB commands to `filename`. Commands are saved in format readable by LABView interface and Perl script reading routine. The `overwrite` field is optional (if omitted, and relevant, user will be prompted for it). The possible choices are: `1` to append to existing `filename`, `2` to overwrite `filename`, or `3` to cancel if `filename` exists.

endScript (**void**)
: Ceases saving commands to a file.

enableBox (**void**)
Enables communication with GPIB box. Note that this only works for current box.

disableBox (**void**)
Disables communication with GPIB box. Note that this only works for current box.

accumulate (**void**)
Creates a queue of commands for the GPIB box. Will not execute them until "execute" command is given; then it will do them all at once.

unaccumulate (**void**)
Turns off accumulation feature. Note that the remaining items in the queue are NOT executed.

execute (**void**)
Executes all commands in the GPIB box queue.

powerUp (**void**)
Resets control registers and DACs to the values that are believed by the software to be true. Designed for use when power is shut off to board without reloading program.

toFile (**String** `filename`)
Saves current card settings to file `filename`.

fromFile (**String** `filename`)
Reads card settings from `filename` and sets registers and DACs to those settings.

updateFile (**String** `filename`)
Reads card settings from card (using readback feature) and updates software memory and persistence file.

init (**void**)
Resets registers and DACs to initial values (DACs set at zero, filters selected to be 0, and all channels disabled).

3.2 QBias Control Registers

LEDOn (**int** `LEDnumber`)
Turns LED `LEDnumber` on. If `LEDnumber = 3`, turns both on.

LEDOff (**int** `LEDnumber`)
Turns LED `LEDnumber` off. If `LEDnumber = 3`, turns both off.

LEDWidth (**int** `LEDnumber`, **float** `width`)
Sets LED `LEDnumber` width to nearest value to `width` smaller than `width`. Input should be in units of ms. Will automatically change pulse generator to appropriate setting. If `LEDnumber = 3`, sets both widths to `width`.

LEDPeriod (**int** `LEDnumber`, **float** `period`)
Sets LED `LEDnumber` period to closest possible value for `period`, less than the input value. Input values should be in units of ms; possible settins are in units of 100 ms. If `LEDnumber = 3`, sets both periods to `period`.

gndQI (**int** `state=1`)
Grounds or ungrounds the Qinner bias. `state = 1` grounds it.

gndQO (**int** `state=1`)
Grounds or ungrounds the Qouter bias. `state = 1` grounds it.

LEDShortPulse (**int** `LEDnumber`)
Sets LED `LEDnumber` to short. If `LEDnumber = 3`, sets both.

LEDLongPulse (**int** `LEDnumber`)
Sets LED `LEDnumber` to long. If `LEDnumber = 3`, sets both.

LEDRepMode (**int** `LEDnumber`)
Sets LED `LEDnumber` to repetitive mode. If `LEDnumber = 3`, sets both.

LEDSingleMode (**int** `LEDnumber`)
Sets LED `LEDnumber` to single fire mode. If `LEDnumber = 3`, sets both.

LEDPulseMode (**int** `LEDnumber`)
Sets LED `LEDnumber` to pulse mode. If `LEDnumber = 3`, sets both.

LEDContMode (**int** `LEDnumber`)
Sets LED `LEDnumber` to continuous mode. If `LEDnumber = 3`, sets both.

LEDFire (**int** `LEDnumber`)
If LED `LEDnumber` is set in single fire mode, fires LED `LEDnumber`. Otherwise, prints error message. If $\hat{L}$EDnumber = 3, sets both.

reset (**void**)
Resets card bits.

readLEDModeStatus (**void**)
Reads back first control register in QBias section and updates memory.

readLEDWPStatus (**void**)
Reads back second control register in QBias section and updates memory.

readQBiasCSR (**void**)
Reads back control registers in QBias section and updates memory.

readQBias (**void**)
Reads back control registers and DACs in QBias section and updates memory.

3.3 QBias DACs

setQIDAC (**float** `value`)
Sets Qinner DAC to `value`.

setQODAC (**float** `value`)
Sets Qouter DAC to `value`.

setLED1DAC (**float** `value`)
Sets LED1 DAC to `value`.

setLED2DAC (**float** `value`)
Sets LED2 DAC to `value`.

readQIDAC (**void**)
Reads Qinner DAC.

readQODAC (**void**)
Reads Qouter DAC.

readLED1DAC (**void**)
Reads LED1 DAC.

readLED2DAC (**void**)
Reads LED2 DAC.

readQBiasDAC (**void**)
Reads all DACs in QBias section.

References

[1] Steve Eichblatt, Sae Woo Nam CDMS note 9712002. *The CDMS Net Protocol*

[2] Steve Eichblatt, Steve Furlanetto *General Information on the Perl-based Electronics Interface*