Table of Contents

How to run the timing systems in few simple (ouch) steps

Link to on-call schedule: https://teamup.com/ks9fj1qdrba84uozmd On-call phone number: +41227665291

Link to firmware repo: https://gitlab.cern.ch/protoDUNE-SP-DAQ/timing-board-firmware

Link to the software repo: https://gitlab.cern.ch/protoDUNE-SP-DAQ/timing-board-software

CERN System setup

The system is meant to be operated from the control pc `np04-srv-012` with the exceptions of operations requiring a JTAG connection to the FPGA (i.e. firmware uploads).

Connections

Board	Usage/Connection	IPBUS ID	IP address
TLU	Timing-trigger link tests	PROD_MASTER	192.168.200.64
Fanout 0	Cryostat and COBs 2-5, 8	PROD_FANOUT_0	192.168.200.65
Fanout 1	-	PROD_FANOUT_1	192.168.200.66
Fanout 2	Timing tests	PROD_FANOUT_2	192.168.200.67
TERTIARY	VST + COB8, cold box + COB 6,7	PTDS_TERTIARY	192.168.200.32
SECONDARY	Test endpoint	PDTS_SECONDARY	192.168.200.17
PRIMARY	CRT	PDTS_PRIMARY	192.168.200.16

Timing system user guide

Quick start guide

The timing software is a C++/python package for operating the protoDune timing hardware from command line. The `pdtbutler` script provides access to the different functionalities of the timing system through an hopefully not-counterintuitive interface.

Loading the work environment at np04

Source the environment file with the following command

```
cd /nfs/sw/timing/
source env_dev.sh
```

Make sure to use a clean session in which `setupDUNEARTDAQ` has not been sourced.

Using the pdtbutler to identify problems

To monitor the status register for the board you can use

```
pdtbutler mst [IPBUS_ID] part [partition number] status
```

Where the status register for the board has the following definitions:

```
        "0000" when W_RST, -- Starting state after reset
        "0001" when W_SFP, -- Waiting for SFP LOS to go low
```

```
"0010" when W_CDR, -- Waiting for CDR lock
"0011" when W_ALIGN, -- Waiting for comma alignment, stable 50MHz phase
"0100" when W_FREQ, -- Waiting for good frequency check
"0101" when W_LOCK, -- Waiting for 8b10 decoder good packet
"0110" when W_RDY, -- Waiting for time stamp initialisation
"1000" when RUN, -- Good to go
"1100" when ERR_R, -- Error in rx
"1101" when ERR_T; -- Error in time stamp check
```

This topic: CENF > TimingSystemBasicOp
Topic revision: r62 - 2019-10-18 - AlessandroThea