DEEP UNDERGROUND NEUTRINO EXPERIMENT

Penn
University of Pennsylvania

WIB Firmware Updates Timing System

Adrian Nikolica for the WIB team (Alex Madorsky, Ben Land)

7 June 2021

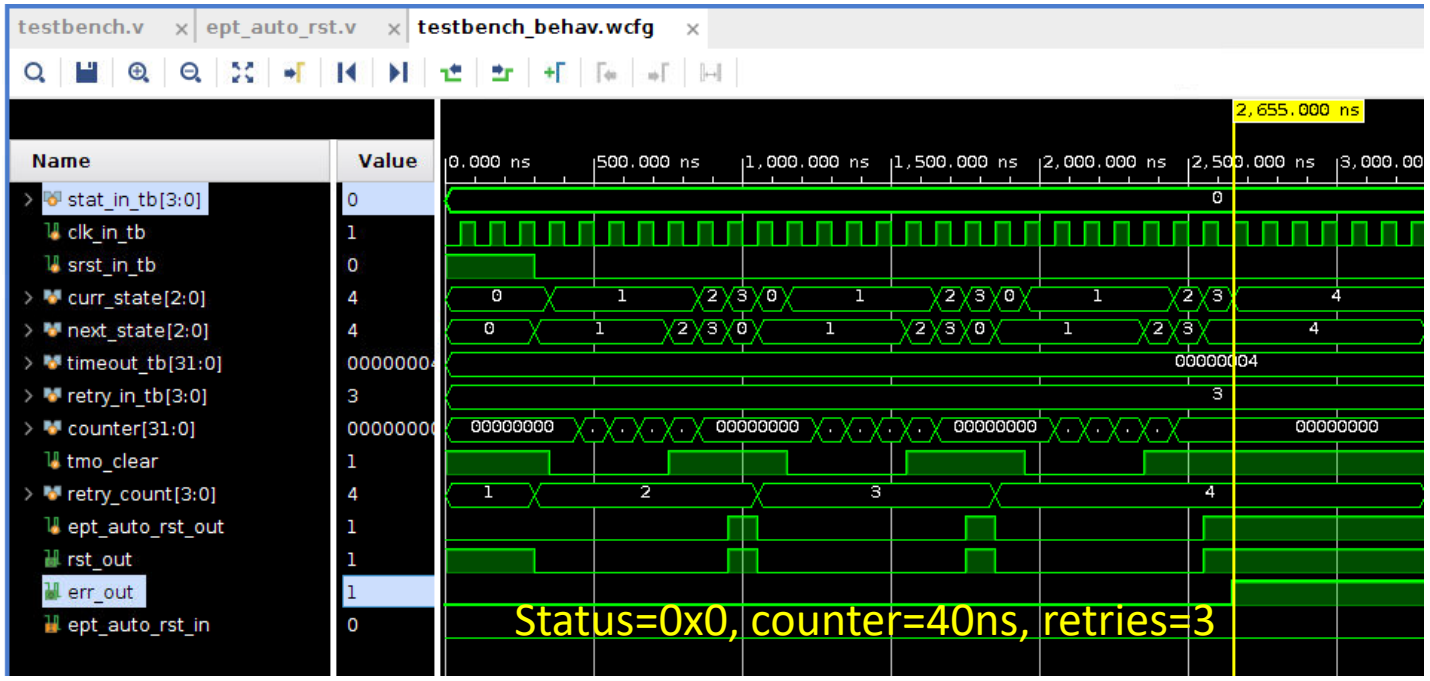
Priorities

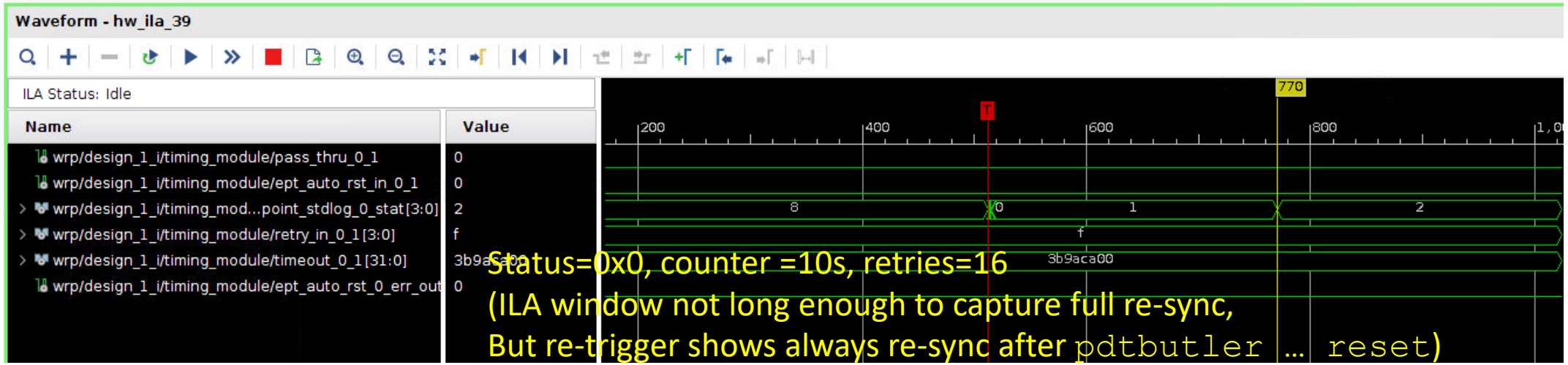
2. Updates to synchronization and checking of FEMB timestamps
   - Changes in response to timing endpoint changes in UK
   - WIB needs to *self-synchronize* without "central intervention"
     - Doesn't use synchronous commands from timing system (which may not exist)
     - It can use periodic SYNC or PPS from timing system
   - WIB needs to measure differential delays to FEMBs
   - WIB needs to provide differential delays to database
   - WIB needs to be updated for new COLDATA version
   - WIB needs to report timestamp errors
   - WIB needs to report synchronization status and update metadata

Implemented auto-reset and tested

Auto-reset based on endpoint lock status

- After endpoint reset, a programmable 32-bit counter is started (10ns – 42.95s)
- After counter rolls over, endpoint status is continuously monitored for when stat != 0x8
- If stat !=0x8, endpoint is reset, counter is restarted, monitor again
- If a programmable 4-bit #re-tries is exceeded, ERROR bit is generated





Notes / Questions

- This FW block mitigates against single bit errors in the timing stream that cause unlock
  - When this block operates, timestamps will go to all zeros, so DAQ will know timestamps are invalid
  - If the timing endpoint cannot be recovered, we can insert the error bit in the datastream
- Is auto-reset a reasonable way to do this?
- Limits of operation?
  - How long does a typical re-sync take?
  - How many re-tries should we allow?
- Are there plans to make the endpoint auto-recover by re-syncing on the data stream?

- WIB needs to *self-synchronize* without "central intervention"
  - Doesn't use synchronous commands from timing system (which may not exist)
  - It can use periodic SYNC or PPS from timing system

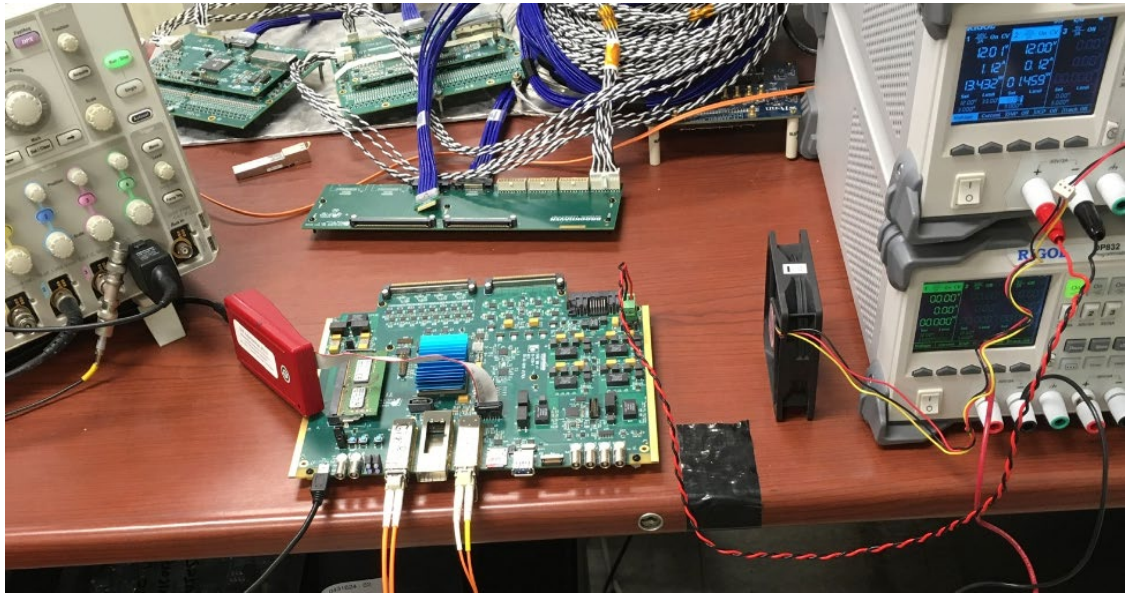
Try this next?
Any advice?

Backup

Setup

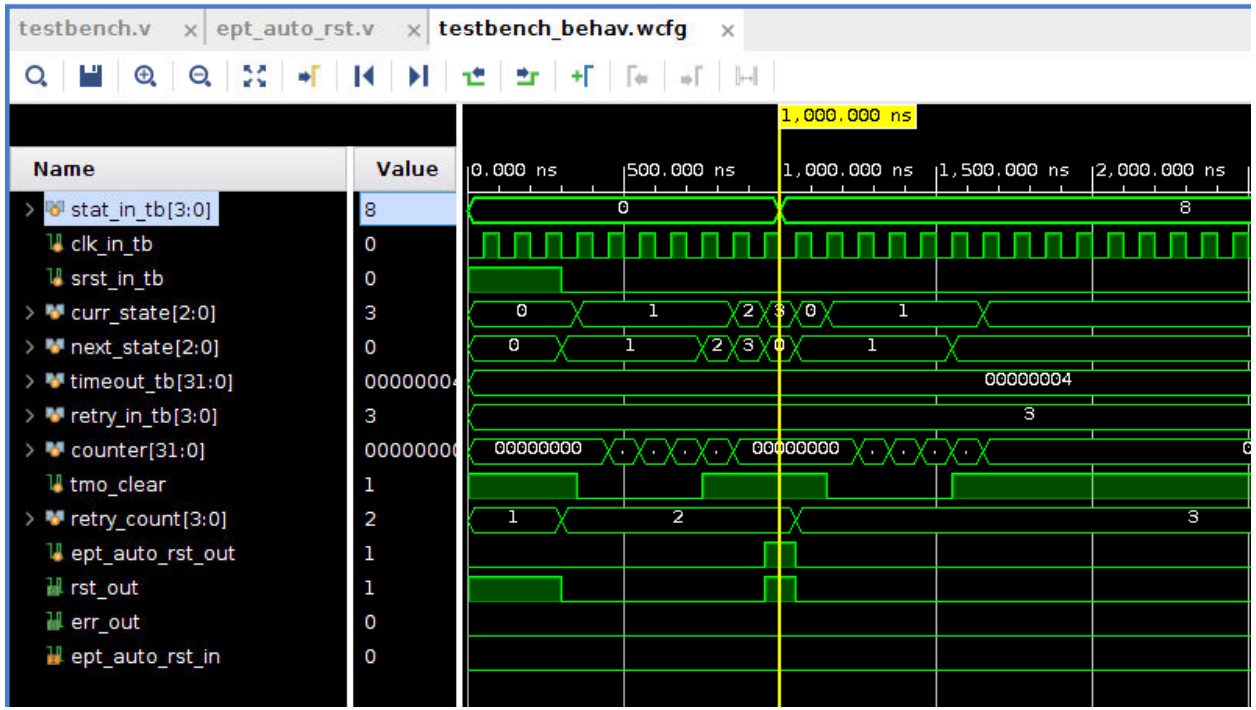
- Everything tested with rev A FMC card
- Below `pdtbulter io PRIMARY reset` command is used to trigger a “fail” condition

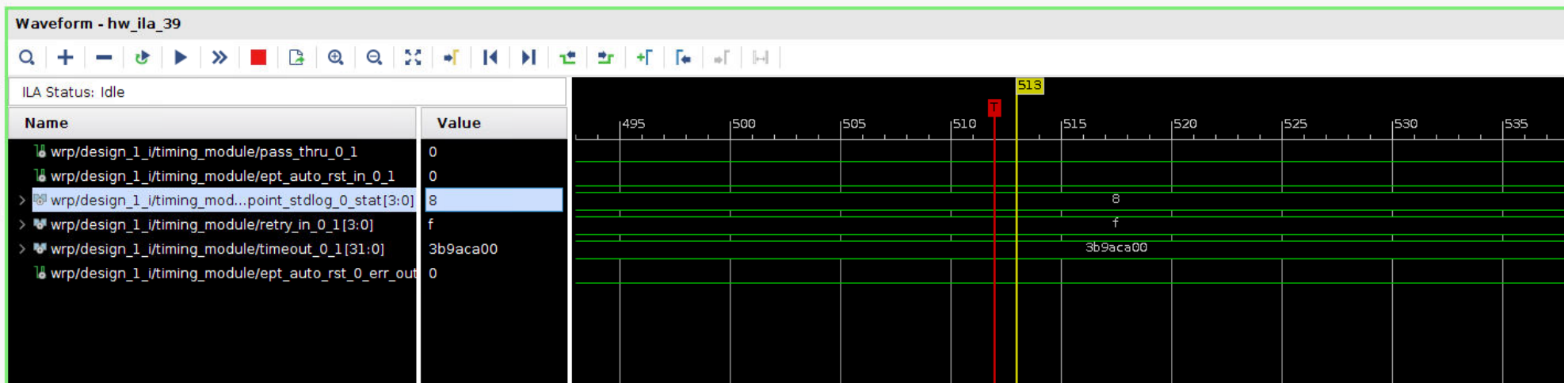
```
[hep@localhost timing-board-software-v5.2.1]$ pdtbutler mst PRIMARY synctime
Created device PRIMARY
ID: design 'ouroboros' on board 'fmc' on carrier 'enclustra-a35'
Master FW rev: 0x50100, partitions: 4, channels: 5
Old Timestamp 0x4c7ca9f8
New Timestamp 0x11dc540a642a81a
-0.000204086303711
Wed, 23 Dec 2020 12:36:36 +0000
[hep@localhost timing-board-software-v5.2.1]$ pdtbutler io PRIMARY reset --force-pll-cfg ../Si5344-053master_312
.5_mhz-Registers.txt
```



Auto-reset based on endpoint lock status

- Endpoint remains running if no 0x8 status change





In Xilinx project

- Block is independent of endpoint