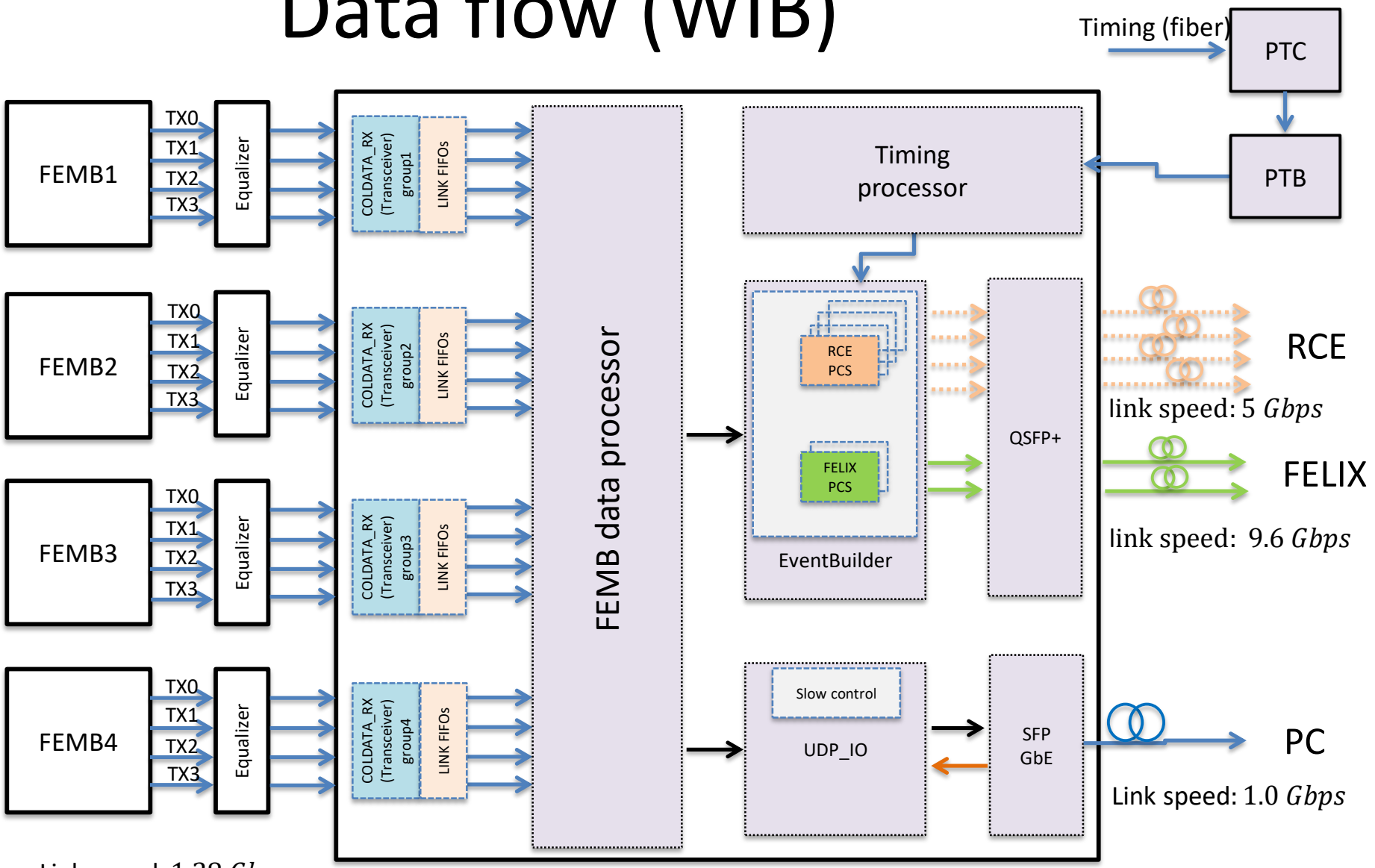
Activities of WIB/FELIX Readout for ProtoDUNE-SP @ BNL

H.Chen, K.Chen, J.Fried, D.Gastler, S.Gao, J.Hugon, J.Zhang

July 4, 2018

Data flow (WIB)



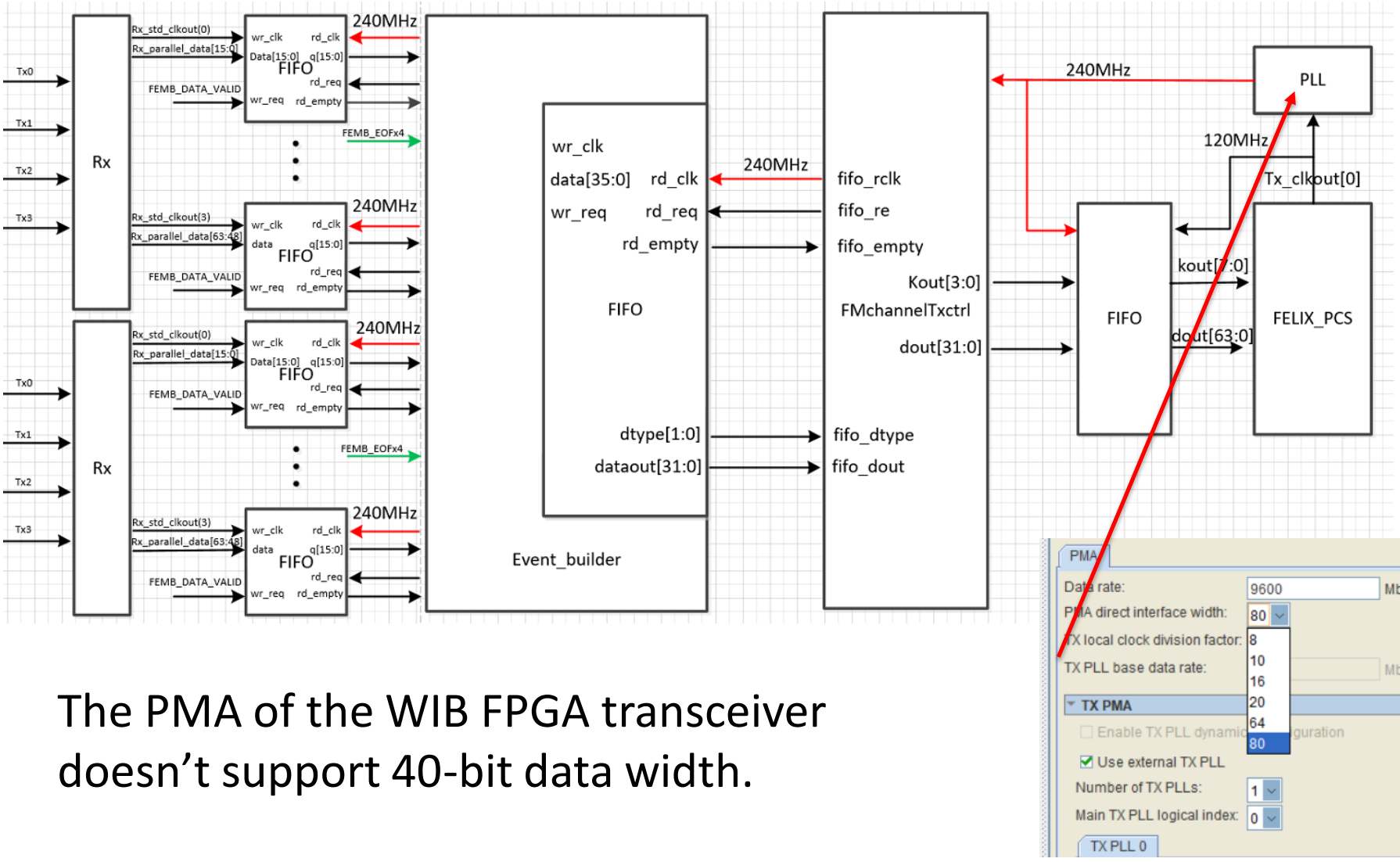
BU firmware 18030101

- Zeros in cold data blocks (corrupted data) we have observed at one link when taking real data (FEMB data).
  - Haven't seen this issue when taking fake data.
- One link is less stable than the other when using fdaq to take data.
  - Sometimes, it stops sending data to FELIX
- Local diagnostic UDP port is not able to collect FEMB data continuously.
- The output data has crc error.

WIB firmware modification

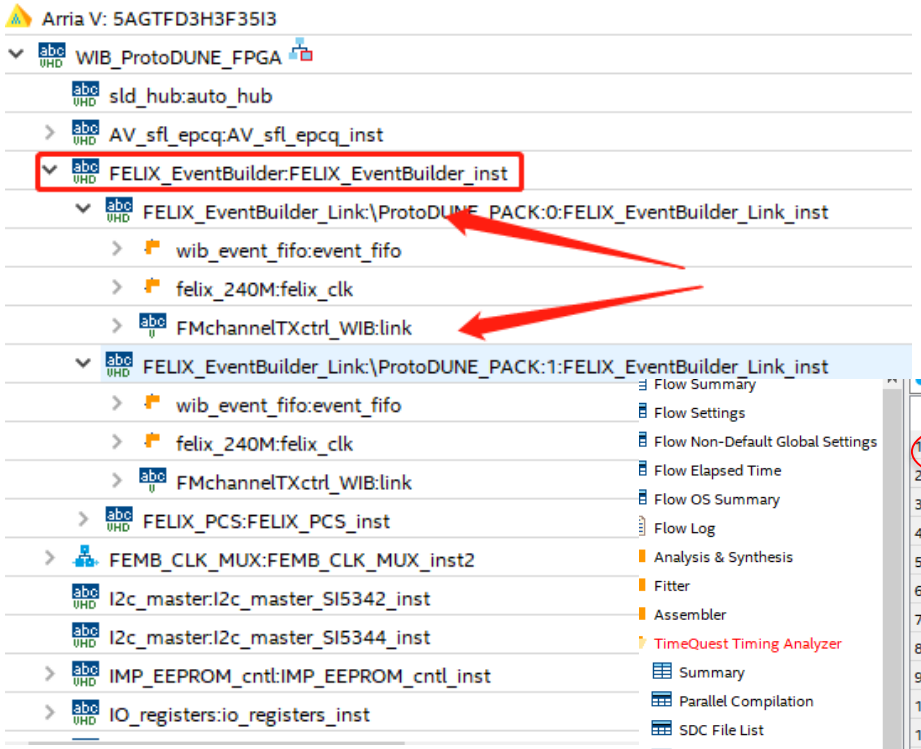
- Revision based on BU firmware
  - Counter mode is functioning well.
  - Zeros in cold data blocks are also observed at one link when taking FEMB data.
- Revision based on BNL firmware
  - Have Integrated FELIX stuff into it.

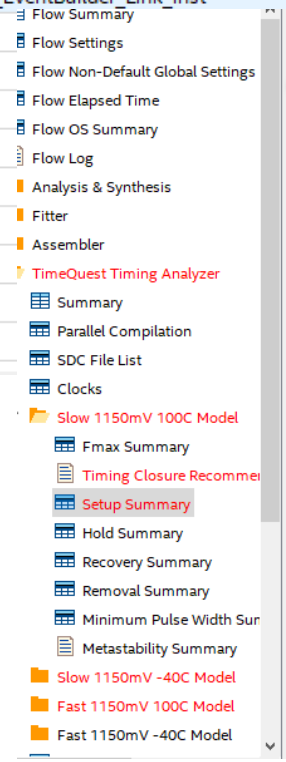
Block diagram

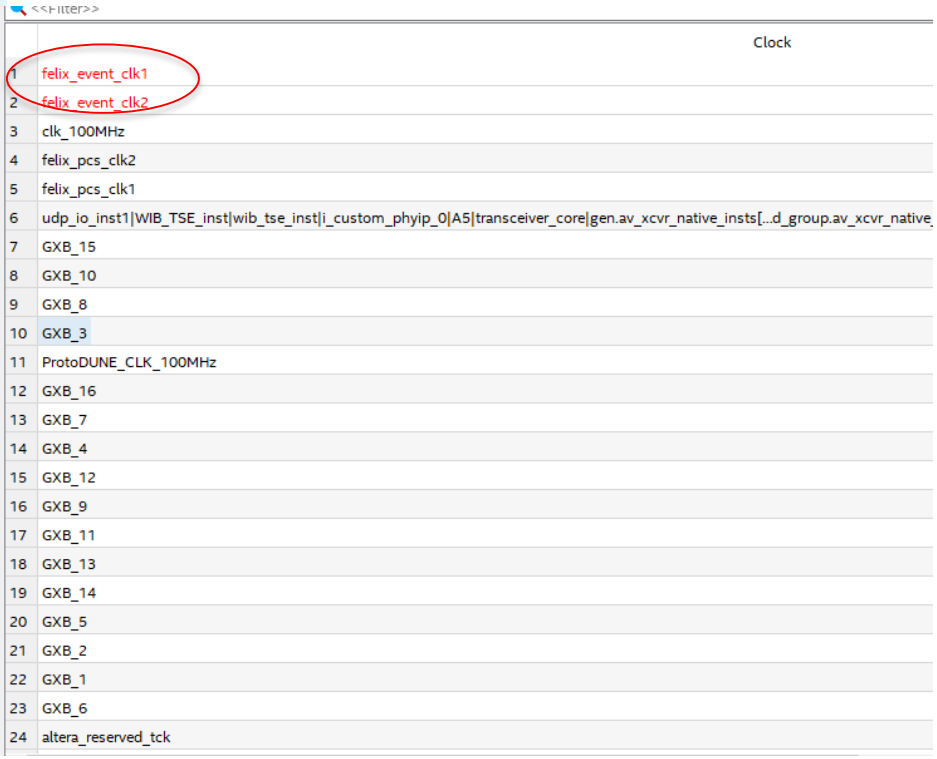


The PMA of the WIB FPGA transceiver doesn't support 40-bit data width.

Timing constraints







All clocks are Ok except 240MHz felix_event clocks. It turns out to be an issue inside FELIX provided CRC module.

FELIX CRC module

FELIX provided CRC module

```
if rising_Edge(Clk) then
    if Reset = '1' then
        if(Calc = '1') then
            for k In 0 to Nbits loop
                if(k = 0) then
                    Reg := (InDirectInitVal);--(CRC_Width-1 downto 0)&dinP(k))xor ('0'&Poly);
                else
                    if Reg(CRC_Width-1) = '1' then
                        Reg := (Reg(CRC_Width-2 downto 0)&din(Nbits - k))xor (Poly);
                    else
                        Reg := Reg(CRC_Width-2 downto 0)&din(Nbits - k);
                    end if;
                end if;
            end loop;
        else
            Reg := InDirectInitVal;
        end if;
    else
        if Calc = '1' then
            for k In 1 to Nbits loop
                if Reg(CRC_Width-1) = '1' then
                    Reg := (Reg(CRC_Width-2 downto 0)&din(Nbits - k))xor (Poly);
                else
                    Reg := Reg(CRC_Width-2 downto 0)&din(Nbits - k);
                end if;
            end loop;
        else
            Reg := Reg;
        end if;
    end if;
```

For loop inside a sequential logic

Deep nesting

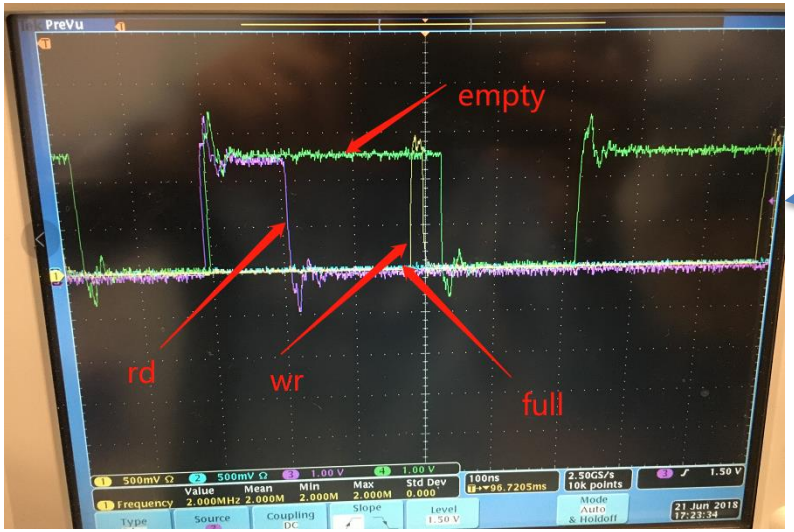
Not a good idea to write HDL code like this although no timing issues found in Xilinx FPGA. It will cause a large delay in data path due to the large combinational logic.
https://its.cern.ch/jira/browse/FLXUSERS-173

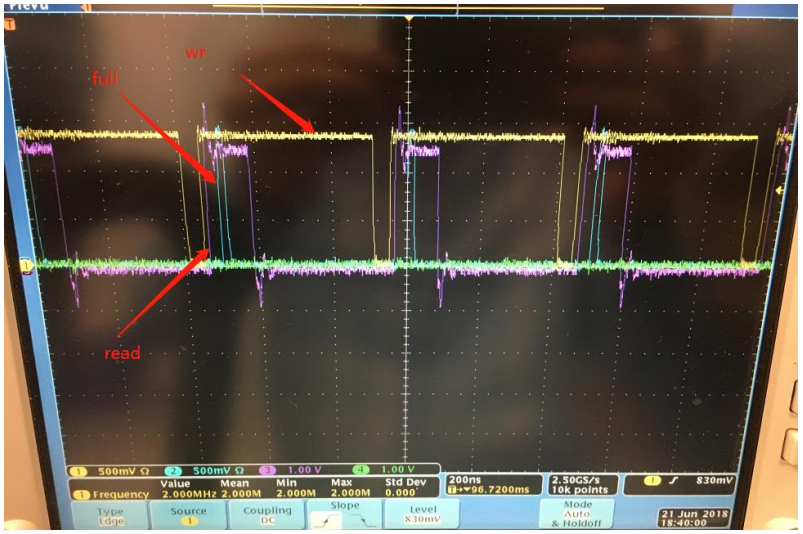
Replacing it with a pipelined crc module generated online, no timing issues reported.
http://outputlogic.com/?page_id=321

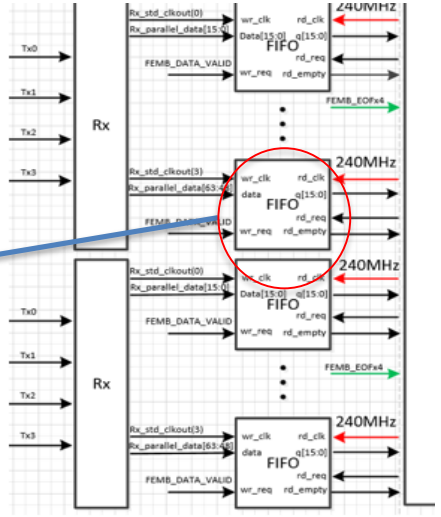
Timing closed

1	felix_event_clk1
2	felix_event_clk2
3	clk_100MHz
4	felix_pcs_clk2
5	felix_pcs_clk1
6	udp_io_inst1|WIB_TSE_inst|wib_tse_inst|i_custom_phyip_0|A5|transceiver_core|gen.av_xcvr_native_insts[...d_group.av_
7	GXB_2
8	ProtoDUNE_CLK_100MHz
9	GXB_1
10	GXB_12
11	GXB_7
12	GXB_13
13	GXB_16
14	GXB_3
15	GXB_14
16	GXB_8
17	GXB_10
18	GXB_4
19	GXB_15
20	GXB_5
21	GXB_11
22	GXB_9
23	GXB_6

Debugging



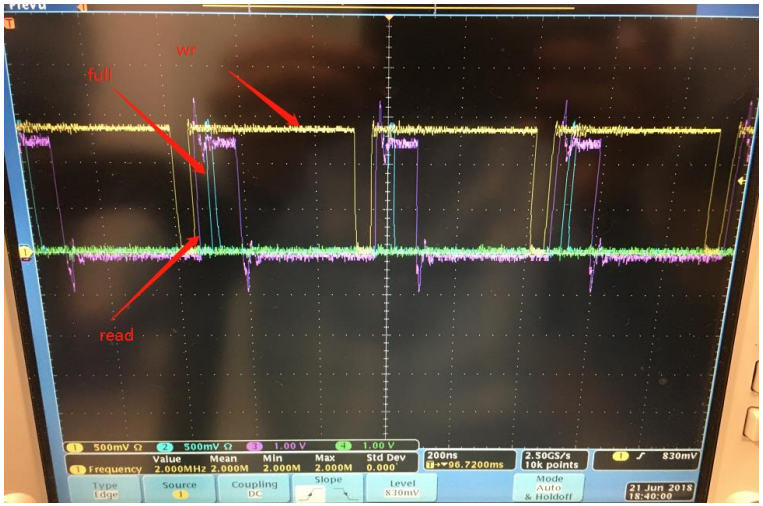




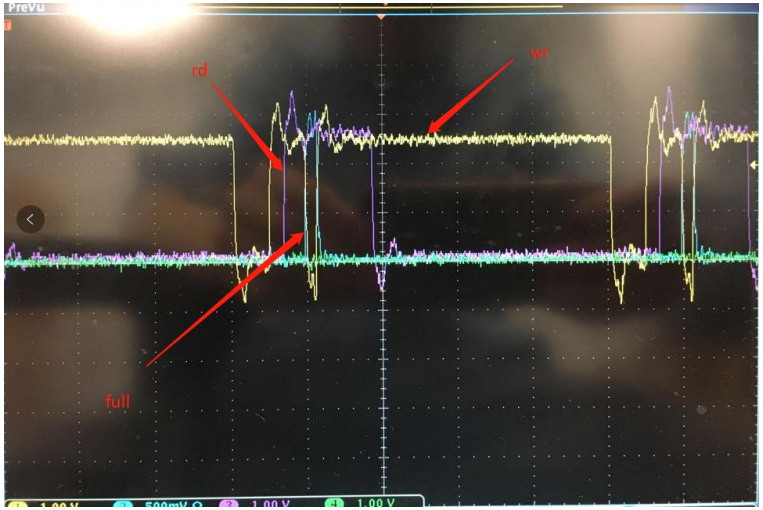
A bug found and fixed.

```
when S_START_OF_FRAME =>
      PKT_LATCH_i        <= '0';
      FRAME_ERROR_i      <= '0';
      CHKSUM_ERROR_i     <= '0';
      HEADER_ERROR_i     <= '0';
      FEMB_DATA_VALID    <= '0'; --bug here.
      UDP_DATA_VALID     <= '0';
      if((DATA_VALID = '1') and ( WORD_CNT <= (Frame_size-1))) then
         WORD_CNT           <= WORD_CNT + 1;
         UDP_DATA_VALID     <= '0';
         if(WORD_CNT /= 0) then
            CHECKSUM_i  <= CHECKSUM_i + DATA_IN;
         end if;
         case WORD_CNT IS
            when 0 =>
               FEMB_DATA_VALID <= '1';
               CHECKSUM_IN     <= DATA_IN;
            when 1 =>
               FEMB_DATA_VALID <= '1'; --added here 0621
               TIME_STAMP_IN   <= DATA_IN;
            when 2 =>
               FEMB_DATA_VALID <= '1'; --added here
               ADC_ERROR_IN    <= DATA_IN;
            when 3 =>
               FEMB_DATA_VALID <= '1'; --added here
               RESERVED_IN     <= DATA_IN;
            when 4 =>
               FEMB_DATA_VALID <= '1'; --added here
               HEADER_IN       <= DATA_IN;
               if(UDP_DISABLE_s = '0')  then
                  UDP_DATA_VALID <= '1';
               end if;
               if(RESERVED_IN(0) = '0') then
                  UDP_DATA_I      <= x"FACE";
               else
```

Debugging



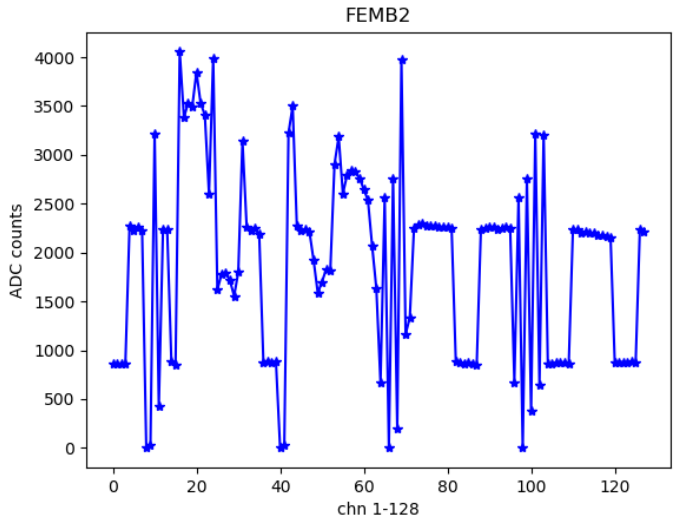
Now the fifo write enable signal becomes normal and we get the ADC data, but the full signal should not appear since read process is more faster than write process.



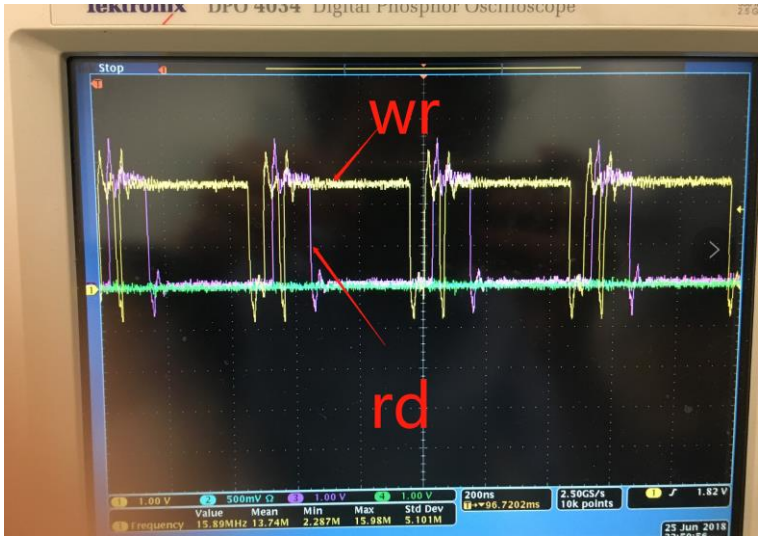
A bug found and fixed

The channel write 29 word into the fifo, but the eventbuilder only read out 28 words. The fifo will full sooner or later.

Debugging

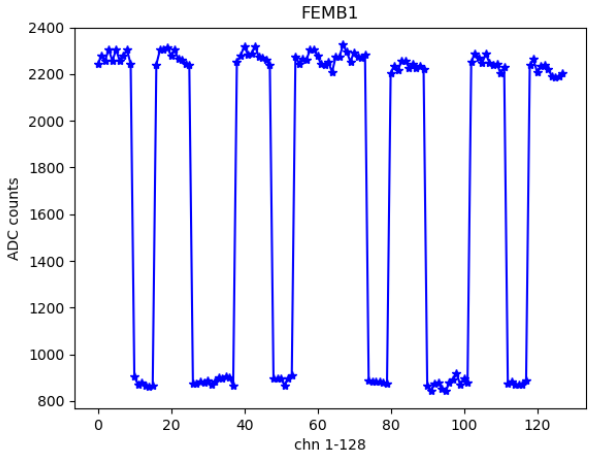


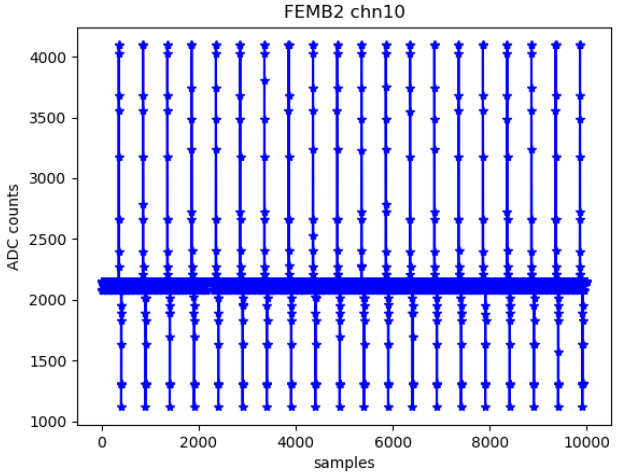
We met this issue before in BU firmware. Data are not aligned. It turns out to be the configuration issue.

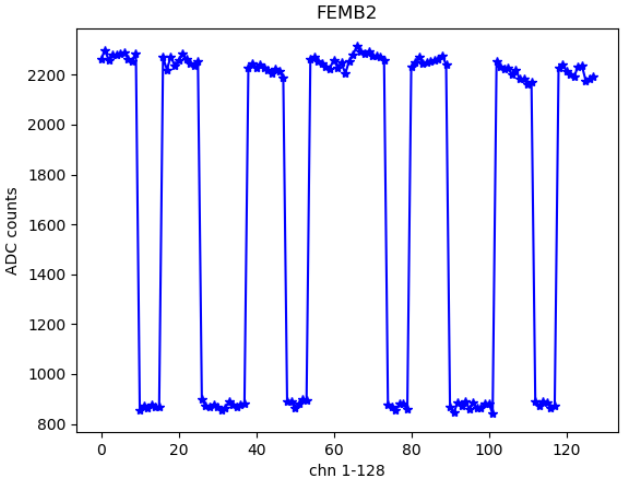


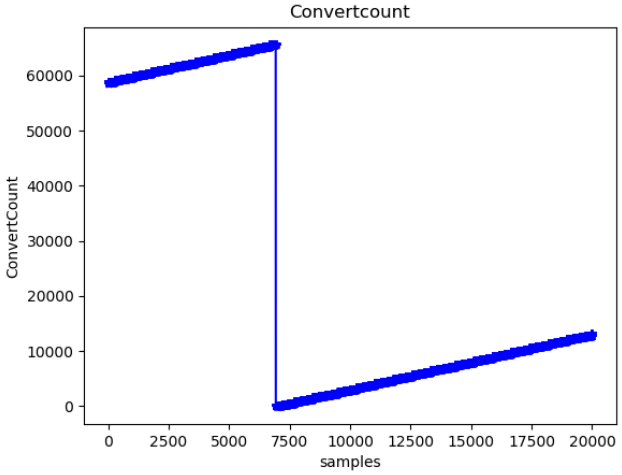
A bug found and fixed. New python scripts developed for configuration. After configuring all 4 FEMBs, a fifo clear operation should be done. Such that, data from all channels are aligned.

Results









All plots are OK

Progress & Next steps

- Both 2 links are continuously taking good FEMB data without CRC error.
- The WIB is freely running for a week, the links are stable and data are good. We are setting up an automatic stability test for the WIB firmware.
- The local diagnostic udp port is working independently with real DAQ. It is able to collect data chip by chip at the same time.
- New python scripts developed for this firmware will be integrated into WIB-FELIX python framework (local diagnostic + FELIX).
- Planning to integrate the timing system into the firmware.