1. Software & Firmware Requirements

Be sure that your Software and Firmware versions are up-to-date in order to use the different trigger settings explained in this document.

Minimum requirements:

- Firmware: Build >= v0.4 (later than 2022/06/08).
- Software: Build >= v1.1 (later than 2022/06/08).

1.1 Firmware & Software Repositories

- Fw: https://drive.google.com/drive/folders/1dntus9AZZc2OQ27HnMgVEYeuPrHYeTWT?usp=sharing
- Sw: https://drive.google.com/drive/folders/1nuQTUB8zAX0AGc2fjoTY8_rc0dZ8Yss6?usp=sharing

Look for the latest *.pof.svf file to permanently program the FPGA flash memory.
As for the software, it is compiled for Debian-like distributions (Ubuntu), Red Hat Enterprise Linux distributions (CentOS, Scientific Linux), and Windows 7/10. Look for the binary of your favorite OS.

1.2 Checking Firmware & Software versions

- Fw: <beta_binary> fw -v
- Sw: <beta_binary> version

1.3 Upgrading Firmware

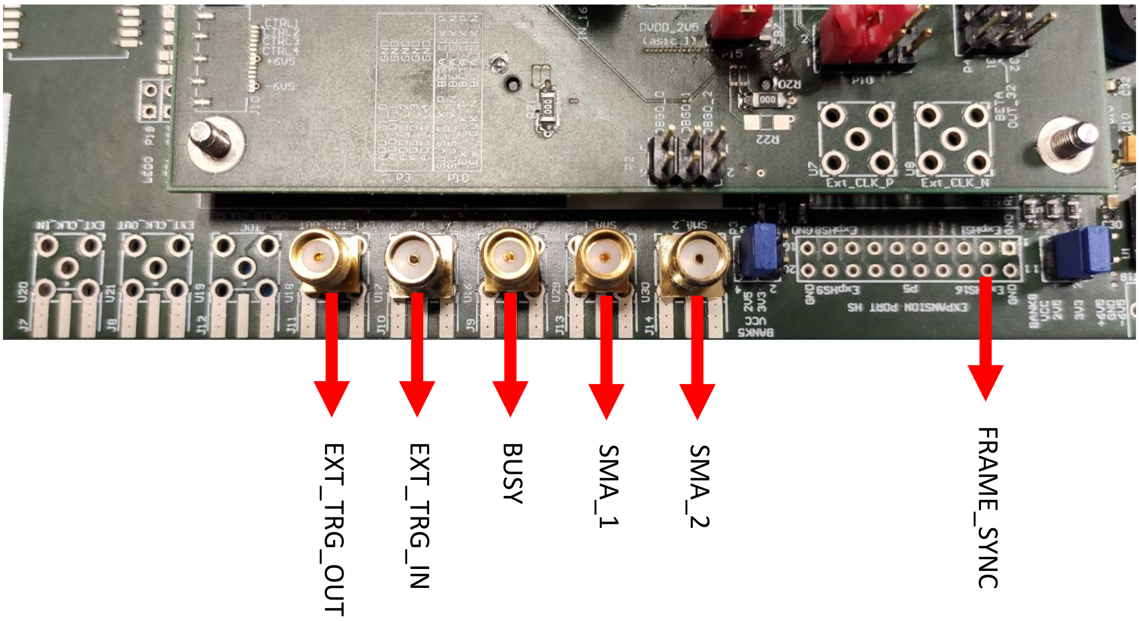
Go to the firmware downloads page and locate the directory *MAXv3.MEZv2.**pof**.svf* and look for the latest firmware version. Sorting the files by date can be very helpful.

The command to program the firmware is:

- <beta_binary> fw -f <path_to_pof.svf>

2. Locating Trigger Signals in BETA test PCB

Figure 1:MAX10 Motherboard & BETA Mezzanine



The previous picture shows the SMA connectors and auxiliary debug pins used in the different trigger modes. The IO standard for all this signals is 2.5V CMOS (single-ended).

It is important to remark that FRAME_SYNC pin is a copy of the differential FRAME_SYNC signal (the one that goes from the FPGA to all the ASICs), and its purpose is merely for debugging purposes only.

2.1 SMA Signals:

- **EXT_TRG_OUT:**
  - Direction: OUTPUT
  - Purpose: Monitors the FPGA trigger signal. It can be helpful to understand what is the selected trigger we are using inside the FPGA.
- **EXT_TRG_IN:**
  - Direction: INPUT
  - Purpose: Allows to trigger the board using an external system (laser, …) as a reference.
- **BUSY:**
  - Direction: OUTPUT
  - Purpose: Tells to an external system that the board is currently processing an event, and therefore no new events can be processed (veto). This signal goes high few ns after the trigger signal and low once the ASIC readout is done.
- **SMA_1 / SMA_2:**
  - Direction: OUTPUT
  - Purpose: Allows the user to individually monitor the trigger output of an ASIC through the SMA connectors. This can be useful for troubleshooting problems.

3. BETA Trigger Modes

The following picture shows the trigger block diagram implemented on the FPGA. For a better understanding of the block, important signals have been highlighted.

- **BOLD_CAPITAL_LETTERS:** Trigger Block input/outputs. Most of them routed to SMA connectors. **BETA_TRGOUT<3:0>** are the Fast-OR Trigger outputs of the 4 BETA ASICs in the board.
- --green_parameters: Correspond to the name of the command option in the software *config* command. To modify any of these parameters:
    - <beta_bin> config --param_1 <param_val> … --param_n <param_val>
        - For more information, type --help option.
- trigger: The internally selected trigger. This signal can be monitored via **EXT_TRG_OUT** SMA, except when trg_mode is 0 (in that case the 4 ASICs Trigger OR is output). The rising edge of this trigger signal produces the **FRAME_SYNC** and **BUSY** sequence by means of three timers. These timers can be adjusted by user.

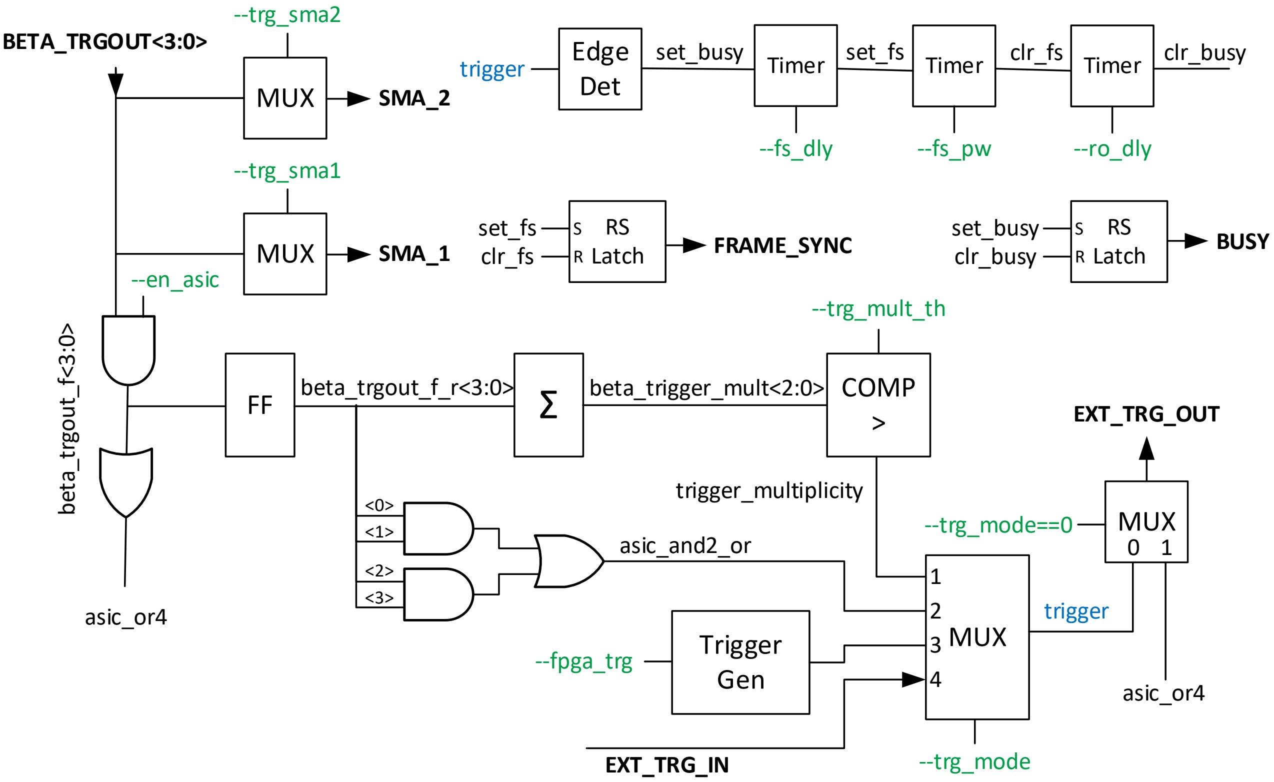


Figure 2: Trigger Block Diagram

3.1 Enabling ASICs’ Trigger output

For those cases where we want to ignore the trigger output a given ASIC (see Figure 2), either because the ASIC is broken or not mounted in the PCB, there is a command option for this purpose:

--en_asic <asic_id> <0/1> [<asic_id> <0/1> … <asic_id> <0/1>]

In the following example we enable Trigger outputs from ASIC #0 & ASIC #1, and disable ASIC #2 & ASIC #3:

- <beta_bin> config [*other_params*] –en_asic 0 1 1 1 2 0 3 0

3.2 SMA1 & SMA2 Trigger debug

This capability allows the user to monitor a certain ASIC Trigger output. You can select 2 ASIC Trigger outputs independently. Parameters:

--trg_sma1 <0..3>

--trg_sma2 <0..3>

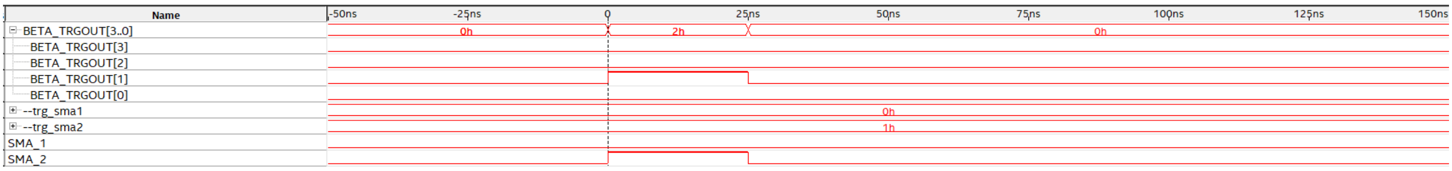


Figure 3: SMA_2 connetor showing the output of BETA_TRGOUT<1>

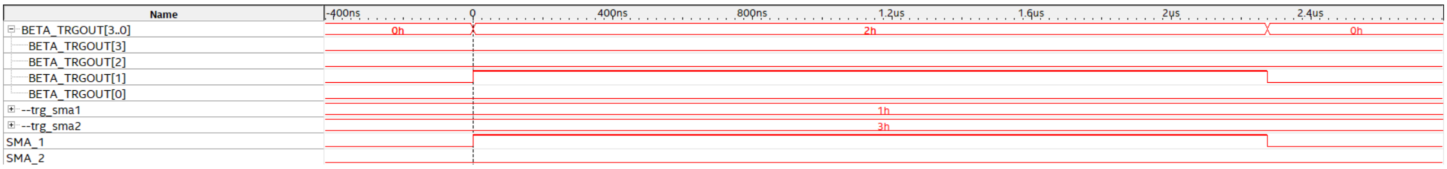


Figure 4: SMA_1 connetor showing the output of BETA_TRGOUT<1>

3.3 Internal Trigger Mode

In this mode, trigger is simply the Fast-OR of the 4 ASICs, and there is no external control. Thus, data conversion will start after the delay time provided by --timer_hold command option. In order to achieve the best performance, it is important to correctly align this signal with the peak of Chanel Shapers’ output. Parameters:

--mode 0

--timer_hold <time_in_ns>

--trg_mode 0

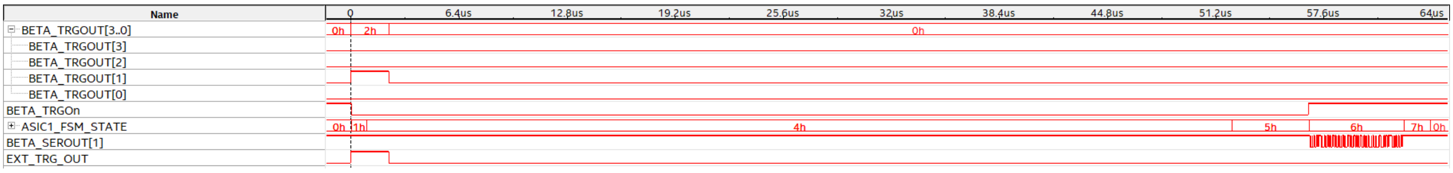


Figure 5: EXT_TRG_OUT SMA showing the Fast-OR between BETA_TRGOUT<3:0> signals

3.4 Multiplicity Trigger Mode (Pseudo-Internal Mode)

In this mode, trigger rises when the number of simultaneous triggering ASICs is higher than a certain threshold (--trg_mult_th). When this occurs, EXT_TRG_OUT SMA connector is high and the internal timers start counting to generate FRAME_SYNC and BUSY signals.

When threshold is set to 0, the behavior is the same as in the Internal Trigger Mode (Section 3.2). However, in this mode ASIC trigger is externally controlled via FRAME_SYNC.

The delay between trigger rising edge condition and FRAME_SYNC rising edge is defined by --fs_dly. The pulse width of FRAME_SYNC (--fs_pw) must be larger enough as to give enough time to the ASICs to perform the ADC conversion. The last timer (--ro_dly) ensures that ASICs have finished data transmission before de asserting BUSY signal. All these values must be provided in nanoseconds. Note: the default values for --fs_pw and --ro_dly should work.

Parameters:

--mode 2

--trg_mode 1

--trg_mult_th <0..3>

--fs_dly <delay_in_ns>

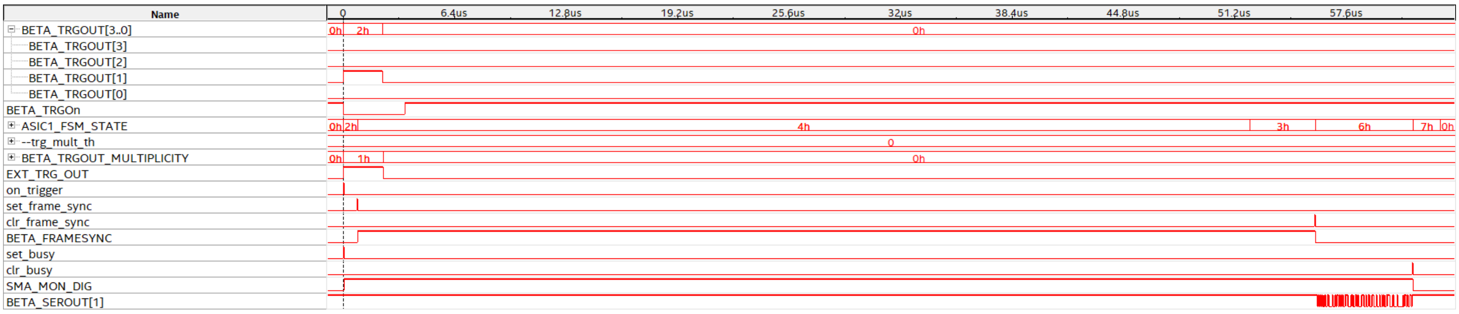


Figure 6: EXT_TRG_OUT SMA goes high when the sum of triggering BETA_TRGOUT<3:0> signals is higher than --trg_mult_th. BUSY signal (SMA_MON_DIG) goes high after trigger and goes low once the data transmission has finished. FRAME_SYNC (BETA_FRAMESYNC) goes high after a certain delay time and goes low sometime after the conversion has finished.

3.5 FPGA Trigger Generator Mode

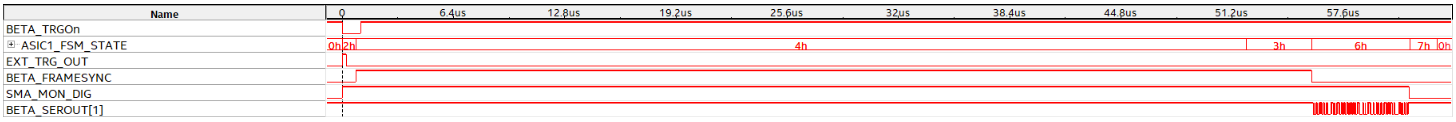
In this mode the FPGA is internally generating the trigger at a certain rate. This rate is provided by --fpga_trg parameter in kHz units. Parameters:

--mode 2

--trg_mode 2

--fpga_trg <0.1…6500>

--fs_dly <delay_in_ns>



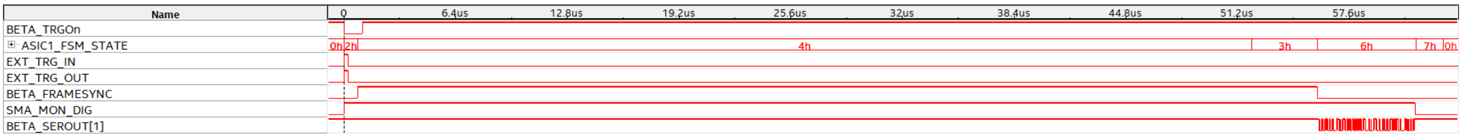
3.6 Exernal Trigger Mode (SMA EXT_TRG_IN)

In this mode, trigger occurs when EXT_TRG_IN SMA input goes high. FRAME_SYNC and BUSY timers works as described in Section 3.3. EXT_TRG_OUT SMA will output the same signal with a certain latency (hundreds of picoseconds).

--mode 2

--trg_mode 3

--fs_dly <delay_in_ns>


Name
BETA_TRGOn
ASIC1_FSM_STATE
EXT_TRG_IN
EXT_TRG_OUT
BETA_FRAMESYNC
SMA_MON_DIG
BETA_SEROUT[1]
6.4us
12.8us
19.2us
25.6us
32us
38.4us
44.8us
51.2us
57.6us
0h
2h
4h
3h
6h
7h
0h