TRIGGER MANUAL

Sibiryak Y. 28.03.2011
Vin 11.05.2011

Dong & Lijiao

The triggers are processed in PHOS by the 24 TRUs inside of modules and 1 TOR outside of PHOS. The **TRU**s can be controlled by the DCS boards

alidcsdcb1584, 1587, 1586, 1585 for M2,
alidcsdcb1595, 1594, 1593, 1592 for M3,
alidcsdcb1588, 1589, 1590, 1591 for M4,

whereas the **TOR** can be controlled by the DCS board ***alidcsdcb1572***.

Before configuring, you need to go to ***alidcscom075***, and then log on to these DCS boards. To log on to a DCS board, you just need to type the two last digits, e.g. "**84**" for ***alidcsdcb1584***, except for the **TOR**, where you type "**tor**". Before you configure the TRU, make sure that the low voltage and the FEE are on.

1. *Check if the TRUs are on or off*

For example: on ***alidcsdcb1584***, type "**rcu-sh r 0x5100**" and you will get "**0x7fff7fff**" if both the TRUs and the FECs are on. The highest "**7fff**" represent the B branch, and the lowest "**7fff**" are for the A branch.
The last bit of each"**7fff**" represents the TRU, the rest one – operable FECs.
Actual values are following:

Module	**M2**		**M3**		**M4**
Brunch	**B A**		**B A**		**B A**
alidcsdcb1584	**0x7fff7fff**	***…1595***	**0x7fff0001**	***…1588***	**0x17fff**
alidcsdcb1587	**0x7fff7fff**	***…1594***	**0x7fff7fff**	***…1589***	**0x7fff7fff**
alidcsdcb1586	**0x7fff7fff**	***…1593***	**0x10001(or0)**	***…1590***	**0x7fff7fff**
alidcsdcb1585	**0x7fff7fff**	***…1592***	**0x7fff7fff**	***…1591***	**0x7fff7b7f**

2. *TRUs initialization*

Go to **/mnt/dcbrw/tru-dcs-share/tru_script**. Type "**./TRU_initial.sh**" to initialize the TRUs. This should be done on every DCS board on which the TRUs are going to be used. After initialization, the screen looks like this:

```
A:0x8002:0x2033
0x8003:0
A:0x8002: 0x4
0x8003:0
B:0x8002:0x102033
0x8003:0
B:0x8002: 0x4
0x8003:0
```

This step takes some time, so please **wait until the terminal prompt comes back**.
After the initialization the threshold is set to **0x100**, the pedestal and mask are set too.

Note: For some brunches the screen looks like shown below:

for M4 …1588:
A:0x8002:0x2033
0x8003:0
A:0x8002: 0x4
0x8003:0
B:0x8002: 0x4
0x8003: 0x2
B:0x8002: 0x4
0x8003: 0x2

for M3 …1593, if both TRUs were not switched:
A:0x8002:0
0x8003:0x1
A:0x8002:0
0x8003:0x1
B:0x8002:0
0x8003:0x1
B:0x8002:0
0x8003:0x1

3. TOR configuring

□ Start Tinserver
Type "ps" on ***alidcsdcb1572*** to check whether the **tinserver** process is running or not, if not, run "**./mnt/dcbrw/starttinserver**" and check again.

□ Check firmware
Go to **/mnt/dcbrw** and check if the firmware is there. Type "**rcu-sh r 0x27**", and you will get the version number. If you get "**no target answer**", the FPGA doesn't have firmware yet, you need program it by typing "**./program_tor tor_fpga4_080311.bit**". Actual version number is **0x53**.

□ Initialize TOR
Run "**rcu-sh b /mnt/dcbrw/set_register.scr**" to initialize the TOR. After the initialization, the default mode is normal mode.

4. Set mask on the TOR side

Actually the mask is already set to **0xfffa, 0xff** by **set_register.scr**.
But because at the time **95_A**, **93_B**, and **93_A** are not available for **M3** and **88_B** is not available for **M4**, it is needed to write "**rcu-sh w 0x1b 0xabfe**" to ignore them. Then if the trigger rate is reasonable (reading register **0x1f**), no more mask is need to set. Otherwise, you need to mask off the TRUs that have unreasonable triggers.

5. TOR busy time 800μs setting

Number **0x8340** should be written in the readout_ mask_register **0xaf** by type "**rcu-sh w 0xaf 0x8340**". It correlates to 800usec busy time of the PHOS.

6. Check TRU trigger rate on TOR inputs

Check the trigger rates for each TRU on ***alidcsdcb1572***.
Run > **cd /mnt/dcbrw/** and then
“**rcu-sh b read_counter_M2_in. scr**” to check Module 2,
“**rcu-sh b read_counter_M3_in. scr**” to check Module 3 and
“**rcu-sh b read_counter_M4_in. scr**” to check Module 4.
Which value is concerned?
After you type “**rcu-sh b read_counter_M2_in. scr**”, you will get this:

```
executing: r 0x38
0x38: 0
executing: wait 100 us
executing: r 0x39
0x39: 0
executing: wait 100 us
executing: r 0x3a
0x3a: 0
executing: wait 100 us
executing: r 0x3b
0x3b: 0
executing: wait 100 us
executing: r 0x3c
0x3c: 0x1
executing: wait 100 us
executing: r 0x3d
0x3d: 0x1
executing: wait 100 us
executing: r 0x3e
0x3e: 0
executing: wait 100 us
executing: r 0x3f
0x3f: 0
executing: wait 100 us
```

The value following “0x** :” is what you need to pay attention to.
For instance, “**0x3c: 0x1**” means the trigger rate for this TRU is **1 Hz**.
If the LED system is off, all the values you get should be around **0x00**.
If the value is very large, this TRU is not configured properly or the FEEs need to be reset. Try to reset the electronics in M2 and M3 by “**GoReady**” on the GUI for the FEE.

!! For M4 do “Configure” and then “TriggerEnable”.
!! If you have done “GoReady” for M4, please reinitialize TRUs in M4, then do “Configure” and “TriggerEnable” anyway.

Now try to read trigger rate again. You should mask off TRUs that have large values (see p.8).

7. Check L0 counting rate

Check if L0 trigger to be sent to the CTP is reasonable or not.
Type "**rcu-sh r 0x1f**", the value you get should be around **0x01**.
If not, please check the trigger rate for each TRU again as described in **step 6**.
(Command "**rcu-sh b read_counter.scr**" also read some TOR registers,
the beginning of **0x1f –** TOR output.)

8. How to mask (if it is not done yet or new mask is needed)

After you type "**rcu-sh b read_counter_M2_in.scr**", you can get the number of triggers in 1 second for each TRU. The counter address, the corresponding TRUs and the bit you need to set to mask off (the corresponding bit is set to **0** to mask **off**) are described as follows:
The TRU address is corresponding to bit number for the mask.

For **M2**, the mask register is **0x1c** in TOR firmware:

		#Bit in 0x1c reg.	
0x38	**84_B**	**0**	
0x39	**87_B**	**1**	
0x3a	**86_B**	**2**	
0x3b	**87_B**	**3**	**M2**
0x3c	**84_A**	**4**	
0x3d	**87_A**	**5**	
0x3e	**86_A**	**6**	
0x3f	**85_A**	**7**	

For **M3** and **M4**, the mask register is **0x1b** in TOR firmware:

		#Bit in 0x1b reg.	
0x30	**95_B**	**8**	
0x31	**94_B**	**9**	
0x32	**93_B**	**10**	
0x33	**92_B**	**11**	**M3**
0x34	**95_A**	**12**	
0x35	**94_A**	**13**	
0x36	**93_A**	**14**	
0x37	**92_A**	**15**	

		#Bit in 0x1b reg.	
0x28	**88_B**	**0**	
0x29	**89_B**	**1**	
0x2a	**90_B**	**2**	
0x2b	**91_B**	**3**	**M4**
0x2c	**88_A**	**4**	
0x2d	**89_A**	**5**	
0x2e	**90_A**	**6**	
0x2f	**91_A**	**7**	

For example, if you get answer like this for **M2**:

		Mask register **0x1c**		
0x38 : 0x3ee	**X**	**0 bit**	**0**	**“6_{16}”**
0x39 : 0x1		**1 bit**	**1**	
0x3a : 0x2		**2 bit**	**1**	
0x3b : 0x789	**X**	**3 bit**	**0**	
0x3c :0x2		**4 bit**	**1**	**“**7_{16}”
0x3d :0x1		**5 bit**	**1**	
0x3e :0x3		**6 bit**	**1**	
0x3f :0x689	**X**	**7 bit**	**0**	

It means TRUs **84_B**, **85_B**, and **85_A** are not working well, you need to mask them out by writing “**rcu-sh w 0x1c 0x76**”.

Note1. Take in mind, that all TRUs are powered by according module LV, while **TOR** is powered by **M2** LV, so if M2 will re-powered, the TOR must be reconfigured after that.

Note2. Take in mind, that TRUs consume big currents from LV PS. Therefore **during of TRUs configuring**, as well as **during of Global runs** you can see dramatic increasing (and jumping) several LV channels current (mostly **ch2 = 3.3V**) on DCS screen LV plots.