Trigger control module for pulse repetition in CONV-TTL-BLO

Carlos Gil Soriano
BE-CO-HT
carlos.gil.soriano@cern.ch

February 6, 2012

Abstract

The task of this HDL core is controlling the pulse replication. Firstly, after a debouncing stage, it detects an input pulse . Once the input is validated, an output signal if generated within the FPGA. This output signal directly controls the pulse generation of the Blocking drivers in CONV-TTL-BLO.

This document shows:

- Parameters used as *generic*
- The registers to control the module.
- Step-by-step instructions for proper use.

Revision history		
HDL version	**Module**	**Date**
0.1	Trigger	February 6, 2012

Contents

1 Structure **3**
- 1.1 HDL cores . . . 3
  - 1.1.1 *trigger_top.vhd* . . . 3
  - 1.1.2 *trigger_regs.vhd* . . . 3
  - 1.1.3 *TT_RAMhandler.vhd* . . . 4
- 1.2 Behaviour . . . 4

2 Parameters **4**
- 2.1 *g_MAX_GLITCH_STAGES* . . . 4
- 2.2 *g_DEFAULT_GLITCH_MASK* . . . 5
- 2.3 *g_MIN_PULSE_LENGTH* . . . 5
- 2.4 *g_MAX_PULSE_LENGTH* . . . 5
- 2.5 *g_DEFAULT_PULSE_LENGTH* . . . 5

3 Registers **5**
- 3.1 STATUS . . . 5
- 3.2 CTR0 . . . 6
- 3.3 CTR1 . . . 6
- 3.4 RAM0 . . . 6
- 3.5 RAM1 . . . 7
- 3.6 RAM2 . . . 7

4 Internal Memory Mapping **8**

5 How to use it **8**
- 5.1 Initialization . . . 8
- 5.2 Changing the deglitch mask and stages length . . . 8
  - 5.2.1 Register writes step-by-step . . . 9
- 5.3 Changing the pulse width . . . 10

1 Structure

The trigger module contains several blocks related the following way:

– trigger_top.vhd
—— trigger_regs.vhd
—— trigger_core.vhd
——— debouncer.vhd
——— monostable.vhd
—— TT_RAMhandler.vhd
——— gc_RAM.vhd (for IDs)
——— gc_RAM.vhd (for TTs)

1.1 HDL cores

1.1.1 *trigger_top.vhd*

The top module interconnects the three basic building blocks: registers, core and generic RAM. Each *trigger_top.vhd* module will control one output Blocking driver.

Inside *trigger_top.vhd* there are some constant that are used as *generic* in both the registers and generic RAMs. The constants are explained later in the section 'Parameters' and are the following:

- *c_RAM_SIZE*
- *c_MAX_GLITCH_STAGES*
- *c_DEFAULT_GLITCH_MASK*
- *c_MIN_PULSE_LENGTH*
- *c_MAX_PULSE_LENGTH*
- *c_DEFAULT_PULSE_LENGTH*
- *c_TAGS_DATA_WIDTH*

1.1.2 *trigger_regs.vhd*

It consist of a core that can be accessed via Wishbone and that contains all the registers which control this trigger core. The minimum and maximum values and proceeding for configuring them will be explained in the two next following sections.

1.1.3 *TT_RAMhandler.vhd*

The TT_RAMhandler is the component that control reads and writes into the RAM space used for time-tagging pulses. It is subdivided into two separated blocks of RAM: one for identification of the input and output pulse shape (so called "ID Block") and the other one for time-tagging ("TT Block").

The idea behind seperating the block memories lies in ease the task of updating the code in case different widths for the ID and TT fields are decided.

1.2 Behaviour

Once a pulse has been deglitched, which translates into a delay of *wb_clk * cycles to match c_DEFAULT_GLITCH_MASK* , a monostable will reproduce a pulse with a length determined by the CPL field in CTR0 register. The duration of the output pulse will be hence, *CTR0[CPL] * wb_clk*. After this time, a preventive action has been taken to not damage the coupled inductors in **CONV-TTL-BLO**. A timeout will be run in which no input pulse will be replicated. The value of this timeout corresponds to *CTR0[CPL] * wb_clk*, which is the same as the outputted pulse.

Min. Deglitch Mask delay *wb_clk * 1*

Max. Deglitch Mask delay *wb_clk * bits of c_DEFAULT_GLITCH_MASK*

Min. input pulse length *c_DEFAULT_GLITCH_MASK * wb_clk*

Output pulse length *CTR0[CPL] * wb_clk*

Inactivity timeout upon output pulse is done *CTR0[CPL] * wb_clk*

2 Parameters

2.1 *g_MAX_GLITCH_STAGES*

It specifies the maximum stages used for debouncing an input signal. The input signal is validated by bit 0 in CTR0. If CRT0[0] is set to 0. Inputs won't be replicated (it won't even be deglitched).

The value of this parameters express the length in bits of the parameter that holds the deglitching mask. The value of the Current deGlitching Mask can be found in CGM in CTR0 register.

2.2 *g_DEFAULT_GLITCH_MASK*

It specifies the default value that it is used for the Current deGlitching Mask for the CGM field in CTR0 register. Only the lower CGM bits specified by the values of *g_MAX_GLITCH_STAGES* will be used as deglitching mask. It should be remarked that the input will be validated against a mask, so values not monotonical can be accepted. Examples are shown at the end of this document.

2.3 *g_MIN_PULSE_LENGTH*

It specifies the default value that it is used for the Minimum Pulse Length for the MinPL field in CTR1 register.

This field overrides the value of CPL in CTR0 in case the user tries to configure an output pulse with a width lower than MinPL. Therefore CTR0 will be MinPL.

BOUNDING MUST BE IMPLEMENTED

2.4 *g_MAX_PULSE_LENGTH*

It specifies the default value that it is used for the Maximum Pulse Length for the MaxPL field in CTR1 register.

This field overrides the value of CPL in CTR0 in case the user tries to configure an output pulse with a width longer than MaxPL. Therefore CTR0 will be MaxPL.

BOUNDING MUST BE IMPLEMENTED

2.5 *g_DEFAULT_PULSE_LENGTH*

It specifies the default value that it is used for the Current Pulse Length for the CPL field in CTR0 register.

It should be noted that its value is bounded by MinPL and MaxPL fields of CTR1 register.

BOUNDING MUST BE IMPLEMENTED

3 Registers

3.1 STATUS

The STATUS register is a read-only register. It shows the basic configuration of the trigger HDL core and the status of the trigger core RAM blocks.

Bits	Field	Meaning
0	EN	General ENable
1	CLR	General CLeaR
3-2	x	Reserved
4	EN_TT	Enable Time-Tagging
5	CLR_TT	CLeaR Time-Tagging
7-6	x	Reserved
8	EMPTY	RAM empty flag
9	FULL	RAM full flag
10	WA	RAM wrapped around
15-11	x	Reserved
31-16	CPL	Current Pulse Length

3.2 CTR0

The CTR0 register is a read-write register. It allows setting up the basic configuration of the trigger HDL core, its RAM blocks and both the deglitching mask and output pulse length to be used.

Bits	Field	Meaning
0	EN	General ENable
1	CLR	General CLeaR
3-2	x	Reserved
4	EN_TT	Enable Time-Tagging
5	CLR_TT	Clear Time-Tagging
7-6	RDM	time-tagging ReaD Mode
15-8	CGM	Current Glitch Mask
31-16	CPL	Current Pulse Length

3.3 CTR1

The CTR1 register is a read-write register. It allows setting up the boundaries that override invalid values of CPL field in CTR0.

Bits	Field	Meaning
15-0	MinPL	Minimum Pulse Length
31-16	MaxPL	Maximum Pulse Length

3.4 RAM0

The RAM0 register is a read-write register. It allows setting up the RAM and the read request to it.

Bits	Field	Meaning
0	EN_TT	Enable Time-Tagging
1	CLR_TT	Clear Time-Tagging
3-2	RDM	time-tagging ReaD mode
4	EMPTY	RAM empty
5	FULL	RAM full
6	WA	RAM Wrapped Around
7	RQT	ReQuesT read
31-8	x	Reserved

3.5 RAM1

The RAM1 register is a read-only register. It shows the current read and write address configured to be accessed.

Bits	Field	Meaning
15-0	CRA	Current Read Address
31-16	CWA	Current Write Address

3.6 RAM2

The RAM2 register is a read-write register. It allows setting up the RAM adress range to be read.

Bits	Field	Meaning
15-0	SA	Starting read Address
31-16	EA	Ending read Address

4 Internal Memory Mapping

The internal registers map is as follow:

Address	Register	Access
0x0	*STATUS*	Read-only
0x1	*CTR0*	Read-write
0x2	*CTR1*	Read-write
0x3	Not used	
0x4	Not used	
0x5	*RAM0*	Read-write
0x6	*RAM1*	Read-only
0x7	*RAM2*	Read-write

5 How to use it

5.1 Initialization

1. Disable trigger core before configuring:
   CTR0: write 0 to EN and EN_TT

2. Set the minimum and maximum pulse lengths:
   CTR1: writes into MinPL and MaxPL

3. Set the deglitching mask and pulse length to be used:
   CTR0: set CGM and CPL

4. Clearing RAM block up before starting:
   RAM0: write 1 to CLR_TT

5. Disable RAM clearup and enable module:
   CTR0: write 1 to EN and EN_TT, write 0 to CLR_TT

5.2 Changing the deglitch mask and stages length

Three examples are given.

Example A

- ***g_MAX_GLITCH_STAGES*** : 6

- **CTR0[CGM]** : 0x0BAA

In this case, only the six less significant bits of CGM field will be used as mask: "11 1010". This mask signal is checked against the sampled input signal. If it matches, it is considered as a pulse and it will be replicated.

Example B The common use will be in the form:

- ***g_MAX_GLITCH_STAGES*** : 6
- **CTR0[CGM]** : 0xFFFF

Which translates into *pulses of length **6 * wb_clk** or greater should be replicated.*

Example C Reducing the Deglitch Mask delay is achieved by configuring CGM properly:

- ***g_MAX_GLITCH_STAGES*** : 6
- **CTR0[CGM]** : 0x0007

which means *pulses of length **3 * wb_clk** or greater should be replicated.*

5.2.1 Register writes step-by-step

1. Disable trigger core before configuring:
   CTR0: write 0 to EN and EN_TT

2. Set the minimum and maximum pulse lengths:
   CTR1: writes into MinPL and MaxPL

3. Set the deglitching mask and pulse length to be used:
   CTR0: set CGM and CPL

4. Reenable module:
   CTR0: write 1 to EN and EN_TT

5.3 Changing the pulse width

1. Disable trigger core before configuring:
   CTR0: write 0 to EN and EN_TT

2. Check/set the minimum and maximum pulse lengths:
   CTR1: read/write into MinPL and MaxPL

3. Set the pulse length to be used:
   CTR0: set CPL

4. Reenable module:
   CTR0: write 1 to EN and EN_TT