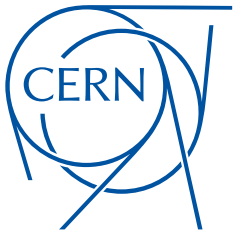
CONV-TTL-BLO HDL Guide

Gateware v4.0
February 17, 2017



Theodor-Adrian Stana (CERN/BE-CO-HT)
Last modified by Denia Bouhired-Ferrag (CERN/BE-CO-HT)

Licensing information

Revision history

Date	Version	Change
04-07-2013	0.1	First draft
26-07-2013	0.2	Second draft
07-08-2013	1.02	Added pulse rejection to *conv_pulse_gen*
14-08-2013	1.02	Changed name of *elma_i2c* to *vbcp_wb*
20-11-2013	1.04	Added MultiBoot module, gateware release v1.0
14-02-2014	2.00	Version 2.0 of gateware, with diagnostics support including: unique board ID and temperature readout, input pulse counters, pulse time-tagging and manual pulse triggering.
08-04-2014	2.10	Version 2.1 of gateware, bringing down the max. allowed input pulse frequency, changing the ERR LED behaviour, adding system errors and changing the pulse timetag FIFO for a timetag ring buffer
25-09-2014	3.00	Version 3.0 of gateware, using the new converter board common gateware [1] **This version changes the memory map**
27-01-2015	3.01	Added repository download commands to Section 1
17-02-2017	4.00	Version 4 modifications, moved LED logic inside the common gateware entity

Contents

List of Figures

List of abbreviations

FPGA	Field-Programmable Gate Array
HDL	Hardware Description Language
LSR	Line Status Register
SR	Status Register

1 Introduction

This document is the HDL guide for the CONV-TTL-BLO board [2]. The HDL for the CONV-TTL-BLO board uses the converter board common gateware [1] as a subproject and adds some external logic to it to adapt for peculiarities on the CONV-TTL-BLO. This short HDL guide explains these peculiarities and the corresponding logic implemented.

The HDL files can be obtained by cloning the repository along with its submodules using the following commands:

```
git clone git://ohwr.org/level-conversion/conv-ttl-blo/conv-ttl-blo-gw.git
cd conv-ttl-blo-gw/
git submodule update --init --recursive
```

1.1 Additional documentation

- Converter board common gateware [1]
- CONV-TTL-BLO User Guide [3]
- CONV-TTL-BLO schematics [4]
- CONV-TTL-BLO OHWR Wiki page [2]

2 Overview

A block diagram of the HDL is shown in Figure 1. This document will detail each of the blocks outside the converter common gateware block in the following sections. The contents of the common gateware block are detailed in the converter common gateware specification [5]. For a more general look at the pulse repetition logic tailored to the CONV-TTL-BLO, refer to the CONV-TTL-BLO User Guide [3].

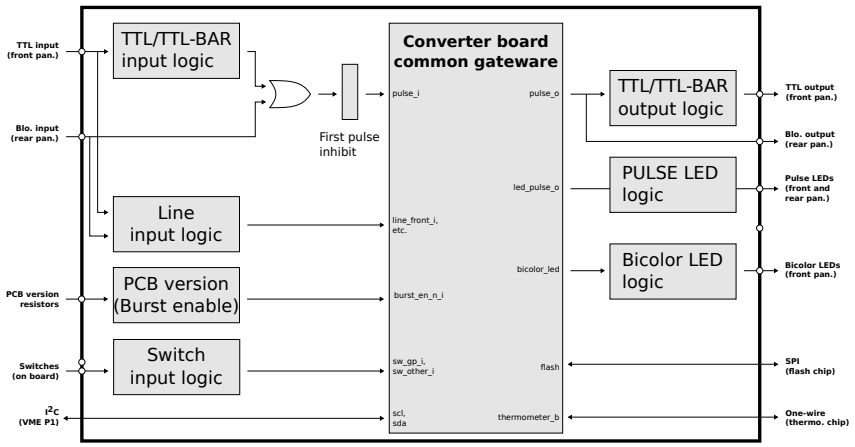


Figure 1: Block diagram of CONV-TTL-BLO gateware

3 Input logic

3.1 TTL/TTL-BAR input logic

The TTL/TTL-BAR input logic is shown in Figure 2. It assures an active-high pulse to the *pulse_i* input of the *conv_common_gw* component and adapts for TTL-BAR pulses that may be input when the TTL switch is on.

In addition, because in TTL-BAR mode a lack of signal on the line is high (due to the on-board Schmitt-trigger buffer), the *no signal detect* block (Figure 3) disables this line if it is high for 100 μs, to allow propagation of blocking pulses arriving on the rear panel while the channel has no cable plugged in while in TTL-BAR mode.

When in TTL-BAR mode, the FRONTFS bits of the lines status register (LSR – see Appendix A.1.42) contain the state of the no signal detect block for each channel and can be used to check if no cable is plugged into the channel. When in TTL mode, the FRONTFS bits are unused.

3.2 First pulse inhibit mechanism

The first pulse inhibit mechanism (Figure 4) is implemented in the form of a counter which waits for 100 μs after reset prior to enabling the line. It is implemented because in TTL-BAR mode, until an inactive line is disabled, the TTL line is high and this may lead to a pulse triggered on the channel, due to reset of modules within the *conv_common_gw* component.

By keeping the line disabled until the no signal detect block in the TTL input logic (Section 3.1) disables the line, no pulse is triggered on the channel. As seen in Figure 4, an extra clock cycle delay is needed before the channel is enabled, to make sure that all reset states inside the *conv_common_gw*

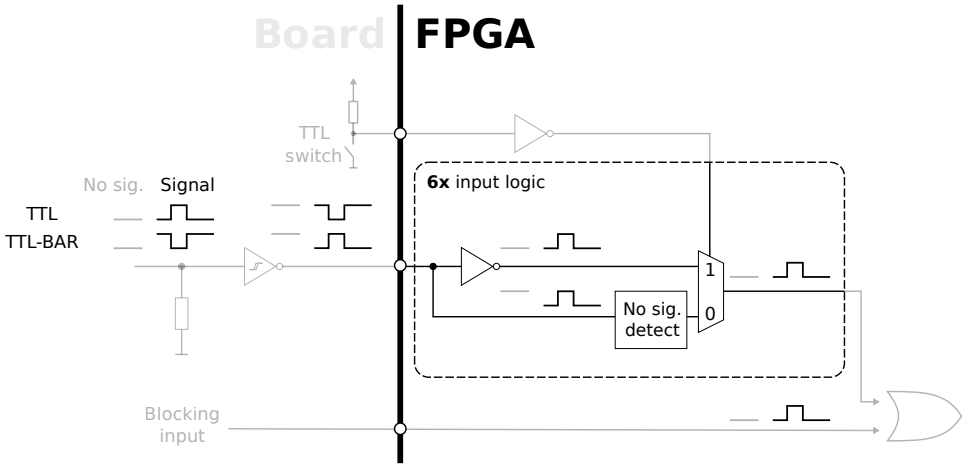


Figure 2: TTL/TTL-BAR input logic

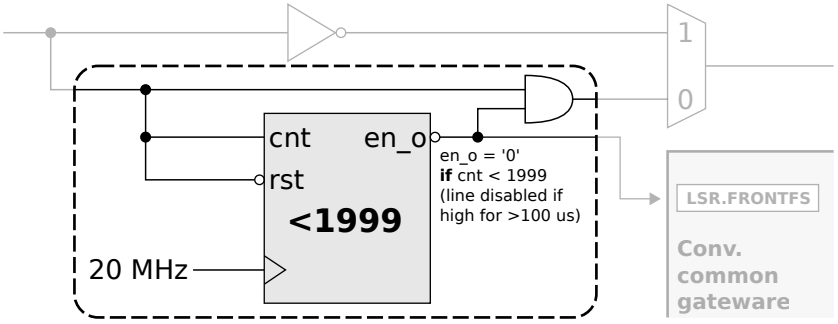


Figure 3: No signal detect block

block have been finished and no pulses are generated.

To keep the logic simple, the pulse inhibit logic disables the line even when the board is in TTL repetition mode. Since in practice the effect it has on the input to the *conv_common_gw* is extending the 100 ms reset by 100 μs, an extra 0.1% delay from reset to full pulse replication capabilities is deemed insignificant in comparison to logic simplification.

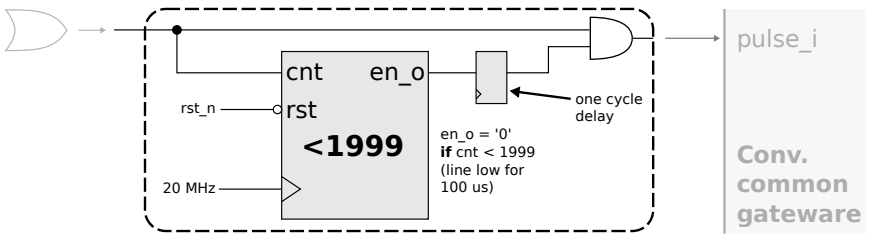


Figure 4: First pulse inhibit mechanism

3.3 Line input logic

The line input logic adapts the levels present at the FPGA inputs due to various on-board circuitry, so that the levels in the line status register (LSR) is active-high. As seen in Figure 5, only the TTL and INV-TTL lines need adaptation in the case of the CONV-TTL-BLO, since the blocking inputs are already adapted on-board for active-high logic.

When in TTL-BAR mode, the FRONTFS bits of the lines status register (LSR – see Appendix A.1.42) contain the state of the no signal detect block for each channel and can be used to check if no cable is plugged into the channel. When in TTL mode, the FRONTFS bits are unused.

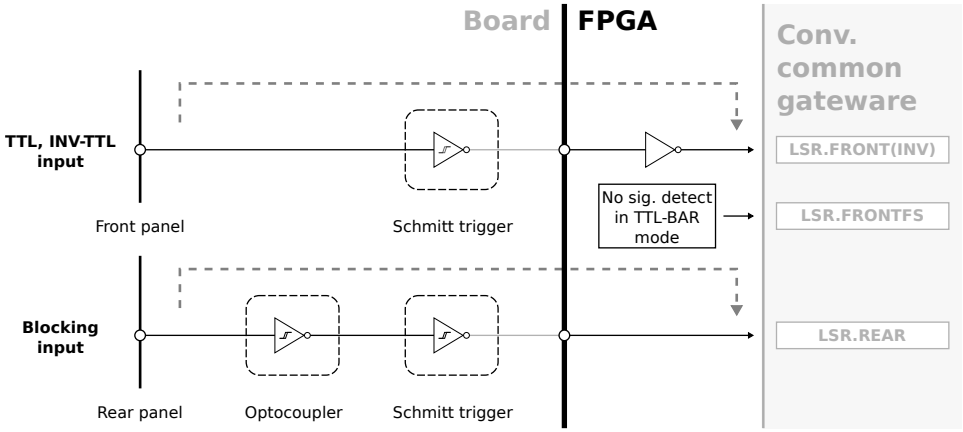


Figure 5: Line input logic

3.4 Switch input logic

Similar to the line input logic (Section 3.3), the general-purpose switch lines must be negated for their active-high reflection in the SR, as shown in Figure 6.

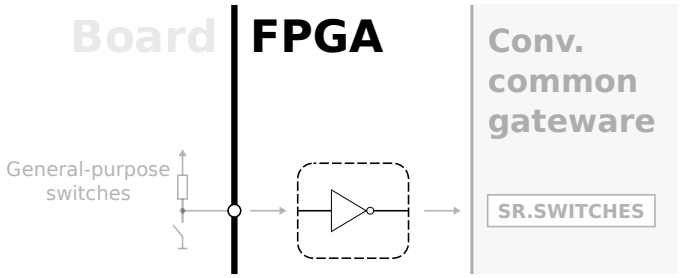


Figure 6: Switch input logic

3.5 PCB Version

CONV-TTL-BLO boards from version 4 onwards, offer the possibility for the FPGA to receive information on the hardware version. This information is hardwired on the board in the form of pulled-up or pulled-down resistors for each bit [6].

As gateware release 4 together with hardware version 4, enables pulse repetition at higher frequencies [3]. Therefore, PCB version input lines are used to enable the so- called *burst mode* enable signal, needed by conv-common-gw component in order to enable pulse width selection and burst mode functionality.

The PCB version I/O is therefore checked, burst mode is enabled via a comparator. if the version is 4 or higher. The functionality is disabled for older PCBs[1].

4 Output logic

4.1 TTL/TTL-BAR output logic

The TTL/TTL-BAR output logic (Figure 7) ensures that TTL pulses are propagated from the *pulse_o* output of *conv_common_o* to the FPGA output when the TTL switch is ON, or that TTL-BAR pulses are propagated when it is OFF.

4.2 Pulse LED output logic

Note that this very same logic has been replicated inside conv-common-gw submodule for inverted channels. This functionality is available on release 4 of the gateware for PCBs v3 onwards.

[1]Since the hardware version is not available on older boards and since the I/O pins now assigned to it were by default pulled- down, board v3 and earlier will show the PCB version as 0)

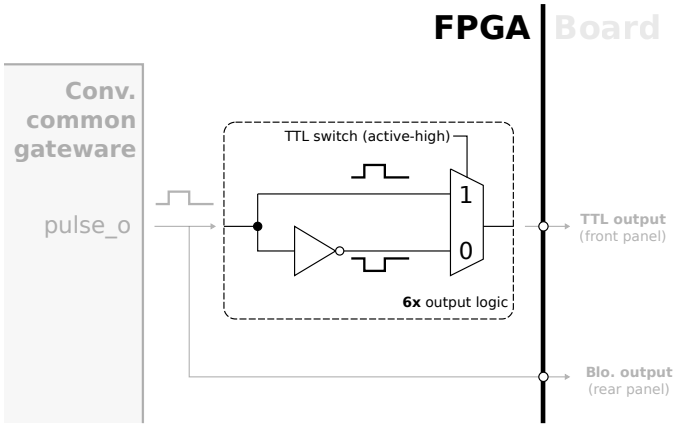


Figure 7: TTL output logic

Since in the CONV-TTL-BLO schematics the pulse LEDs are driven from inverting Schmitt triggers to ground, the active-high pulse LED output from *conv_common_gw* must be inverted prior to driving the Schmitt trigger. This is done via the pulse LED logic (Figure 8).

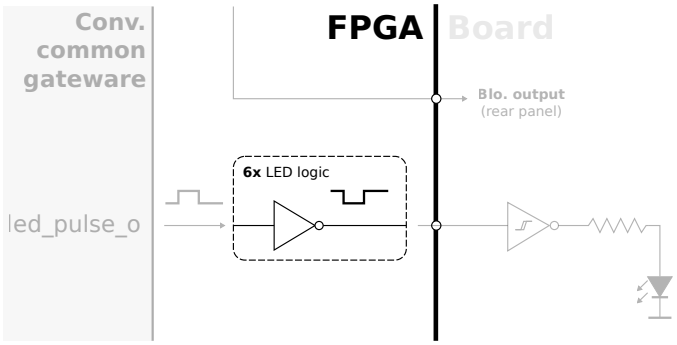


Figure 8: Pulse LED logic

4.3 Bicolor LED output logic

The bicolor LED logic external to the *conv_common_gw* takes the bicolor LED outputs as well as specific control pins (such as, for example, the I^2C LED drive pin, flashing four times on an I^2C transfer) and connects them to the bicolor LEDs, adding multiplexer logic where needed to control the lighting and color of the LED.

The way in which each LED is turned on is described in the CONV-TTL-BLO User Guide [3].

Appendices

A Memory map

Table 1 shows the complete memory map of the gateware. The following sections list the memory map of each peripheral.

In order to convert address values to register index values for SNMP access, the following formula should be used:

$$reg.index = \frac{addr}{4} + 1$$

Table 1: *conv_common_gw* memory map

Peripheral	Address range		Description
Board registers	0x000	0x0ff	Coverter board registers
MultiBoot	0x100	0x11f	MultiBoot module
SDB descriptor	0xf00	0xfff	SDB descriptor (see [7])

A.1 Converter board registers

Base address: 0x000

Offset	Reset	Name	Description
0x0	0x54424c4f	BIDR	Board ID Register
0x4	Note(1)	SR	Status Register
0x8	0x00000000	ERR	Error Register
0xc	0x00000000	CR	Control Register
0x10	0x00000000	CH1TTLPCR	Channel 1 TTL Pulse Counter Register
0x14	0x00000000	CH2TTLPCR	Channel 2 TTL Pulse Counter Register
0x18	0x00000000	CH3TTLPCR	Channel 3 TTL Pulse Counter Register
0x1c	0x00000000	CH4TTLPCR	Channel 4 TTL Pulse Counter Register
0x20	0x00000000	CH5TTLPCR	Channel 5 TTL Pulse Counter Register
0x24	0x00000000	CH6TTLPCR	Channel 6 TTL Pulse Counter Register
0x28	0x00000000	CH1BLOPCR	Channel 1 BLO Pulse Counter Register
0x2c	0x00000000	CH2BLOPCR	Channel 2 BLO Pulse Counter Register
0x30	0x00000000	CH3BLOPCR	Channel 3 BLO Pulse Counter Register
0x34	0x00000000	CH4BLOPCR	Channel 4 BLO Pulse Counter Register
0x38	0x00000000	CH5BLOPCR	Channel 5 BLO Pulse Counter Register
0x3c	0x00000000	CH6BLOPCR	Channel 6 BLO Pulse Counter Register
0x40	0x00000000	TVLR	Time Value Low Register
0x44	0x00000000	TVHR	Time Value High Register
0x48	0x00000000	TBMR	Tag Buffer Meta Register
0x4c	0x00000000	TBCYR	Tag Buffer Cycles Register
0x50	0x00000000	TBTLR	Tag Buffer TAI Low Register
0x54	0x00000000	TBTHR	Tag Buffer TAI High Register
0x58	0x00020000	TBCSR	Tag Buffer Control and Status Register
0x5c	0x00000000	CH1LTSCYR	Channel 1 Latest Timestamp Cycles Register
0x60	0x00000000	CH1LTSTLR	Channel 1 Latest Timestamp TAI Low Register

Offset	Reset	Name	Description
0x64	0x00000000	CH1LTSTHR	Channel 1 Latest Timestamp TAI High Register
0x68	0x00000000	CH2LTSCYR	Channel 2 Latest Timestamp Cycles Register
0x6c	0x00000000	CH2LTSTLR	Channel 2 Latest Timestamp TAI Low Register
0x70	0x00000000	CH2LTSTHR	Channel 2 Latest Timestamp TAI High Register
0x74	0x00000000	CH3LTSCYR	Channel 3 Latest Timestamp Cycles Register
0x78	0x00000000	CH3LTSTLR	Channel 3 Latest Timestamp TAI Low Register
0x7c	0x00000000	CH3LTSTHR	Channel 3 Latest Timestamp TAI High Register
0x80	0x00000000	CH4LTSCYR	Channel 4 Latest Timestamp Cycles Register
0x84	0x00000000	CH4LTSTLR	Channel 4 Latest Timestamp TAI Low Register
0x88	0x00000000	CH4LTSTHR	Channel 4 Latest Timestamp TAI High Register
0x8c	0x00000000	CH5LTSCYR	Channel 5 Latest Timestamp Cycles Register
0x90	0x00000000	CH5LTSTLR	Channel 5 Latest Timestamp TAI Low Register
0x94	0x00000000	CH5LTSTHR	Channel 5 Latest Timestamp TAI High Register
0x98	0x00000000	CH6LTSCYR	Channel 6 Latest Timestamp Cycles Register
0x9c	0x00000000	CH6LTSTLR	Channel 6 Latest Timestamp TAI Low Register
0xa0	0x00000000	CH6LTSTHR	Channel 6 Latest Timestamp TAI High Register
0xa4	Note(2)	LSR	Line Status Register
0xa8	0x00000000	OSWR	Other switch register
0xac	Unique ID	UIDLR	Thermometer ID Low register
0xb0	Unique ID	UIDHR	Thermometer ID High register
0xb4	0x00000000	TEMPR	Board Temperature Register

Note (1): The reset value of the SR cannot be specified, since it is based on the gateware version, the state of the on-board switches and whether an RTM is plugged in or not.
Note (2): The reset value of the LSR cannot be specified, since it depends on whether a cable is plugged into the channel or not.

A.1.1 BIDR – Board ID Register

31	30	29	28	27	26	25	24
BIDR[31:24]							
23	22	21	20	19	18	17	16
BIDR[23:16]							
15	14	13	12	11	10	9	8
BIDR[15:8]							
7	6	5	4	3	2	1	0
BIDR[7:0]							

- **BIDR** [*read-only*]: ID register bits
  Reset value: 0x54424c4f

- **Unimplemented bits**: write as '0', read undefined

A.1.2 SR – Status Register

31	30	29	28	27	26	25	24
-	-	-	WRPRES	HWVERS[5:2]			
23	22	21	20	19	18	17	16
HWVERS[1:0]		RTM[5:0]					
15	14	13	12	11	10	9	8
SWITCHES[7:0]							
7	6	5	4	3	2	1	0
GWVERS[7:0]							

- **GWVERS** [*read-only*]: Gateware version
  Leftmost nibble hex value is major release decimal value
  Rightmost nibble hex value is minor release decimal value
  e.g.
  0x11 – v1.1
  0x2e – v2.14

- **SWITCHES** [*read-only*]: Status of on-board general-purpose switches
  Eg: SW1.1– SR.SWITCHES[0]
  SW1.2– SR.SWITCHES[1]
  SW2.1– SR.SWITCHES[4]
  SW2.4– SR.SWITCHES[7]
  1 – switch is ON
  0 – switch is OFF

- **RTM** [*read-only*]: RTM detection lines citertm-det
  1 bit per RTM output channel
  1 – line active
  0 – line inactive

- **HWVERS** [*read-only*]: Hardware version
  PCB version - Hardwired on the board
  Only meaningful for HW v4.0 and over
  Earlier versions show 0. The register
  uses 4 bits for the version number and
  2 bits for the execution.
  e.g.
  0x010001 – hw v4.1
  0x010111 – hw v5.3
  0x00– hw v3 and earlier

- **WRPRES** [*read-only*]: White Rabbit present
  1 – White Rabbit present
  0 – White Rabbit not present

A.1.3 ERR - Error Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	FWDG_PMISSE[5:0]					
7	6	5	4	3	2	1	0
FLIM_PMISSE[5:0]						I2C_ERR	I2C_WDTO

- **I2C_WDTO** [*read/write*]: I2C communication watchdog timeout error
  1 – timeout occured
  0 – no timeout
  This bit can be cleared by writing a '1' to it

- **I2C_ERR** [*read/write*]: I2C communication error
  1 – attempted to address non-existing address
  0 – idle
  This bit can be cleared by writing a '1' to it

- **FLIM_PMISSE** [*read/write*]: Frequency error
  1 – Input above maximum supported frequency
  0 – idle
  Bit 0 – channel 1
  Bit 1 – channel 2
  etc.
  Each bit can be cleared by writing a '1' to it

- **FWDG_PMISSE** [*read/write*]: Frequency watchdog error
  1 – Pulse over maximum pulse count for given frequency'
  0 – idle

Bit 0 – channel 1
Bit 1 – channel 2
etc.
Each bit can be cleared by writing a '1' to it

A.1.4 CR - Control Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	MPT[7:6]	
7	6	5	4	3	2	1	0
MPT[5:0]						RST	RST_UNLOCK

- **RST_UNLOCK** [*read/write*]: Reset unlock bit
  1 – Reset bit unlocked
  0 – Reset bit locked

- **RST** [*read/write*]: Reset bit - active only if RST_UNLOCK is 1
  1 – initiate logic reset
  0 – no reset

- **MPT** [*write-only*]: Manual Pulse Trigger
  Write the following sequence to trigger a pulse:
  0xde – Byte 1 of magic sequence
  0xad – Byte 2 of magic sequence
  0xbe – Byte 3 of magic sequence
  0xef – Byte 4 of magic sequence
  Number in range 1..6 – trigger a pulse

- **Unimplemented bits**: write as '0', read undefined

A.1.5 CH1TTLPCR - Channel 1 Pulse Counter Register for TTL pulses

31	30	29	28	27	26	25	24
CH1TTLPCR[31:24]							
23	22	21	20	19	18	17	16
CH1TTLPCR[23:16]							
15	14	13	12	11	10	9	8
CH1TTLPCR[15:8]							
7	6	5	4	3	2	1	0
CH1TTLPCR[7:0]							

- **CH1TTLPCR** [*read/write*]: TTL pulse counter value

A.1.6 CH2TTLPCR - Channel 2 Pulse Counter Register for TTL pulses

31	30	29	28	27	26	25	24
CH2TTLPCR[31:24]							
23	22	21	20	19	18	17	16
CH2TTLPCR[23:16]							
15	14	13	12	11	10	9	8
CH2TTLPCR[15:8]							
7	6	5	4	3	2	1	0
CH2TTLPCR[7:0]							

- **CH2TTLPCR** [*read/write*]: TTL pulse counter value

A.1.7 CH3TTLPCR - Channel 3 Pulse Counter Register for TTL pulses

31	30	29	28	27	26	25	24
CH3TTLPCR[31:24]							
23	22	21	20	19	18	17	16
CH3TTLPCR[23:16]							
15	14	13	12	11	10	9	8
CH3TTLPCR[15:8]							
7	6	5	4	3	2	1	0
CH3TTLPCR[7:0]							

- **CH3TTLPCR** [*read/write*]: TTL pulse counter value

A.1.8 CH4TTLPCR - Channel 4 Pulse Counter Register for TTL pulses

31	30	29	28	27	26	25	24
CH4TTLPCR[31:24]							
23	22	21	20	19	18	17	16
CH4TTLPCR[23:16]							
15	14	13	12	11	10	9	8
CH4TTLPCR[15:8]							
7	6	5	4	3	2	1	0
CH4TTLPCR[7:0]							

- **CH4TTLPCR** [*read/write*]: TTL pulse counter value

A.1.9 CH5TTLPCR - Channel 5 Pulse Counter Register for TTL pulses

31	30	29	28	27	26	25	24
CH5TTLPCR[31:24]							
23	22	21	20	19	18	17	16
CH5TTLPCR[23:16]							
15	14	13	12	11	10	9	8
CH5TTLPCR[15:8]							
7	6	5	4	3	2	1	0
CH5TTLPCR[7:0]							

- **CH5TTLPCR** [*read/write*]: TTL pulse counter value

A.1.10 CH6TTLPCR - Channel 6 Pulse Counter Register for TTL pulses

31	30	29	28	27	26	25	24
CH6TTLPCR[31:24]							
23	22	21	20	19	18	17	16
CH6TTLPCR[23:16]							
15	14	13	12	11	10	9	8
CH6TTLPCR[15:8]							
7	6	5	4	3	2	1	0
CH6TTLPCR[7:0]							

- **CH6TTLPCR** [*read/write*]: TTL pulse counter value

A.1.11 CH1BLOPCR - Channel 1 Pulse Counter Register for BLO pulses

31	30	29	28	27	26	25	24
CH1BLOPCR[31:24]							
23	22	21	20	19	18	17	16
CH1BLOPCR[23:16]							
15	14	13	12	11	10	9	8
CH1BLOPCR[15:8]							
7	6	5	4	3	2	1	0
CH1BLOPCR[7:0]							

- **CH1BLOPCR** [*read/write*]: BLO pulse counter value

A.1.12 CH2BLOPCR - Channel 2 Pulse Counter Register for BLO pulses

31	30	29	28	27	26	25	24
CH2BLOPCR[31:24]							
23	22	21	20	19	18	17	16
CH2BLOPCR[23:16]							
15	14	13	12	11	10	9	8
CH2BLOPCR[15:8]							
7	6	5	4	3	2	1	0
CH2BLOPCR[7:0]							

- **CH2BLOPCR** [*read/write*]: BLO pulse counter value

A.1.13 CH3BLOPCR - Channel 3 Pulse Counter Register for BLO pulses

31	30	29	28	27	26	25	24
CH3BLOPCR[31:24]							
23	22	21	20	19	18	17	16
CH3BLOPCR[23:16]							
15	14	13	12	11	10	9	8
CH3BLOPCR[15:8]							
7	6	5	4	3	2	1	0
CH3BLOPCR[7:0]							

- **CH3BLOPCR** [*read/write*]: BLO pulse counter value

A.1.14 CH4BLOPCR - Channel 4 Pulse Counter Register for BLO pulses

31	30	29	28	27	26	25	24
CH4BLOPCR[31:24]							
23	22	21	20	19	18	17	16
CH4BLOPCR[23:16]							
15	14	13	12	11	10	9	8
CH4BLOPCR[15:8]							
7	6	5	4	3	2	1	0
CH4BLOPCR[7:0]							

- **CH4BLOPCR** [*read/write*]: BLO pulse counter value

A.1.15 CH5BLOPCR - Channel 5 Pulse Counter Register for BLO pulses

31	30	29	28	27	26	25	24
CH5BLOPCR[31:24]							
23	22	21	20	19	18	17	16
CH5BLOPCR[23:16]							
15	14	13	12	11	10	9	8
CH5BLOPCR[15:8]							
7	6	5	4	3	2	1	0
CH5BLOPCR[7:0]							

- **CH5BLOPCR** [*read/write*]: BLO pulse counter value

A.1.16 CH6BLOPCR - Channel 6 Pulse Counter Register for BLO pulses

31	30	29	28	27	26	25	24
CH6BLOPCR[31:24]							
23	22	21	20	19	18	17	16
CH6BLOPCR[23:16]							
15	14	13	12	11	10	9	8
CH6BLOPCR[15:8]							
7	6	5	4	3	2	1	0
CH6BLOPCR[7:0]							

- **CH6BLOPCR** [*read/write*]: BLO pulse counter value

A.1.17 TVLR - Time Value Low Register

31	30	29	28	27	26	25	24
TVLR[31:24]							
23	22	21	20	19	18	17	16
TVLR[23:16]							
15	14	13	12	11	10	9	8
TVLR[15:8]							
7	6	5	4	3	2	1	0
TVLR[7:0]							

- **TVLR** [*read/write*]: TAI seconds counter bits 31..0
  Writing this field resets the internal cycles counter.

A.1.18 TVHR - Time Value High Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TVHR[7:0]							

- **TVHR** [*read/write*]: TAI seconds counter bits 39..32
  Writing this field resets the internal cycles counter.

A.1.19 TBMR - Tag Buffer Meta Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	CHAN[5:0]					

- **CHAN** [*read-only*]: Channel mask
  Mask for the channel(s) that triggered time-tag storage:
  bit 0 – channel 1
  bit 1 – channel 2
  ...
  bit 5 – channel 6

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.20 TBCYR - Tag Buffer Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	TBCYR[27:24]			
23	22	21	20	19	18	17	16
TBCYR[23:16]							
15	14	13	12	11	10	9	8
TBCYR[15:8]							
7	6	5	4	3	2	1	0
TBCYR[7:0]							

- **TBCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.21 TBTLR - Tag Buffer TAI Low Register

31	30	29	28	27	26	25	24
TBTLR[31:24]							
23	22	21	20	19	18	17	16
TBTLR[23:16]							
15	14	13	12	11	10	9	8
TBTLR[15:8]							
7	6	5	4	3	2	1	0
TBTLR[7:0]							

- **TBTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.22 TBTHR - Tag Buffer TAI High Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TBTHR[7:0]							

- **TBTHR** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

A.1.23 TBCSR - Tag Buffer Control and Status Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	CLR	EMPTY	FULL
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	USEDW[6:0]						

- **USEDW** [*read-only*]: Buffer counter
  Number of samples in the ring buffer

- **FULL** [*read-only*]: Buffer full
  1 – buffer full
  0 – buffer is not full

- **EMPTY** [*read-only*]: Buffer empty
  1 – buffer empty
  0 – buffer is not empty

- **CLR** [*read/write*]: Clear tag buffer
  1 – clear
  0 – no effect

A.1.24 CH1LTSCYR - Channel 1 Latest Timestamp Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	CH1LTSCYR[27:24]			
23	22	21	20	19	18	17	16
CH1LTSCYR[23:16]							
15	14	13	12	11	10	9	8
CH1LTSCYR[15:8]							
7	6	5	4	3	2	1	0
CH1LTSCYR[7:0]							

- **CH1LTSCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.25 CH1LTSTLR - Channel 1 Latest Timestamp TAI Low Register

31	30	29	28	27	26	25	24
CH1LTSTLR[31:24]							
23	22	21	20	19	18	17	16
CH1LTSTLR[23:16]							
15	14	13	12	11	10	9	8
CH1LTSTLR[15:8]							
7	6	5	4	3	2	1	0
CH1LTSTLR[7:0]							

- **CH1LTSTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.26 CH1LTSTHR - Channel 1 Latest Timestamp TAI High Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TAI[7:0]							

- **TAI** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.27 CH2LTSCYR - Channel 2 Latest Timestamp Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	CH2LTSCYR[27:24]			
23	22	21	20	19	18	17	16
CH2LTSCYR[23:16]							
15	14	13	12	11	10	9	8
CH2LTSCYR[15:8]							
7	6	5	4	3	2	1	0
CH2LTSCYR[7:0]							

- **CH2LTSCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.28 CH2LTSTLR - Channel 2 Latest Timestamp TAI Low Register

31	30	29	28	27	26	25	24
CH2LTSTLR[31:24]							
23	22	21	20	19	18	17	16
CH2LTSTLR[23:16]							
15	14	13	12	11	10	9	8
CH2LTSTLR[15:8]							
7	6	5	4	3	2	1	0
CH2LTSTLR[7:0]							

- **CH2LTSTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.29 CH2LTSTHR - Channel 2 Latest Timestamp TAI High Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TAI[7:0]							

- **TAI** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.30 CH3LTSCYR - Channel 3 Latest Timestamp Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	CH3LTSCYR[27:24]			
23	22	21	20	19	18	17	16
CH3LTSCYR[23:16]							
15	14	13	12	11	10	9	8
CH3LTSCYR[15:8]							
7	6	5	4	3	2	1	0
CH3LTSCYR[7:0]							

- **CH3LTSCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.31 CH3LTSTLR - Channel 3 Latest Timestamp TAI Low Register

31	30	29	28	27	26	25	24
CH3LTSTLR[31:24]							
23	22	21	20	19	18	17	16
CH3LTSTLR[23:16]							
15	14	13	12	11	10	9	8
CH3LTSTLR[15:8]							
7	6	5	4	3	2	1	0
CH3LTSTLR[7:0]							

- **CH3LTSTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.32 CH3LTSTHR - Channel 3 Latest Timestamp TAI High Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TAI[7:0]							

- **TAI** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.33 CH4LTSCYR - Channel 4 Latest Timestamp Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	CH4LTSCYR[27:24]			
23	22	21	20	19	18	17	16
CH4LTSCYR[23:16]							
15	14	13	12	11	10	9	8
CH4LTSCYR[15:8]							
7	6	5	4	3	2	1	0
CH4LTSCYR[7:0]							

- **CH4LTSCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.34 CH4LTSTLR - Channel 4 Latest Timestamp TAI Low Register

31	30	29	28	27	26	25	24
CH4LTSTLR[31:24]							
23	22	21	20	19	18	17	16
CH4LTSTLR[23:16]							
15	14	13	12	11	10	9	8
CH4LTSTLR[15:8]							
7	6	5	4	3	2	1	0
CH4LTSTLR[7:0]							

- **CH4LTSTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.35 CH4LTSTHR - Channel 4 Latest Timestamp TAI High Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TAI[7:0]							

- **TAI** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.36 CH5LTSCYR - Channel 5 Latest Timestamp Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	CH5LTSCYR[27:24]			
23	22	21	20	19	18	17	16
CH5LTSCYR[23:16]							
15	14	13	12	11	10	9	8
CH5LTSCYR[15:8]							
7	6	5	4	3	2	1	0
CH5LTSCYR[7:0]							

- **CH5LTSCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.37 CH5LTSTLR - Channel 5 Latest Timestamp TAI Low Register

31	30	29	28	27	26	25	24
CH5LTSTLR[31:24]							
23	22	21	20	19	18	17	16
CH5LTSTLR[23:16]							
15	14	13	12	11	10	9	8
CH5LTSTLR[15:8]							
7	6	5	4	3	2	1	0
CH5LTSTLR[7:0]							

- **CH5LTSTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.38 CH5LTSTHR - Channel 5 Latest Timestamp TAI High Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TAI[7:0]							

- **TAI** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.39 CH6LTSCYR - Channel 6 Latest Timestamp Cycles Register

31	30	29	28	27	26	25	24
-	-	-	-	CH6LTSCYR[27:24]			
23	22	21	20	19	18	17	16
CH6LTSCYR[23:16]							
15	14	13	12	11	10	9	8
CH6LTSCYR[15:8]							
7	6	5	4	3	2	1	0
CH6LTSCYR[7:0]							

- **CH6LTSCYR** [*read-only*]: Cycles counter
  Value of the 8-ns cycles counter when time tag was taken.

A.1.40 CH6LTSTLR - Channel 6 Latest Timestamp TAI Low Register

31	30	29	28	27	26	25	24
CH6LTSTLR[31:24]							
23	22	21	20	19	18	17	16
CH6LTSTLR[23:16]							
15	14	13	12	11	10	9	8
CH6LTSTLR[15:8]							
7	6	5	4	3	2	1	0
CH6LTSTLR[7:0]							

- **CH6LTSTLR** [*read-only*]: Lower part of TAI seconds counter
  Value of the TAI seconds counter bits 31..0 when time tag was taken.

A.1.41 CH6LTSTHR - Channel 6 Latest Timestamp TAI High Register

31	30	29	28	27	26	25	24
WRTAG	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TAI[7:0]							

- **TAI** [*read-only*]: Upper part of TAI seconds counter
  Value of the TAI seconds counter bits 39..32 when time tag was taken.

- **WRTAG** [*read-only*]: White Rabbit present
  1 - Current time tag generated with White Rabbit
  0 - Current time tag generated with internal counter

A.1.42 LSR - Line Status Register

31	30	29	28	27	26	25	24
REARFS[5:0]						FRONTINVFS[3:2]	
23	22	21	20	19	18	17	16
FRONTINVFS[1:0]		FRONTFS[5:0]					
15	14	13	12	11	10	9	8
REAR[5:0]						FRONTINV[3:2]	
7	6	5	4	3	2	1	0
FRONTINV[1:0]		FRONT[5:0]					

- **FRONT** [*read-only*]: Front panel channel input state
  Line state at board input
  Bit 0 – channel 1
  Bit 1 – channel 2
  etc.

- **FRONTINV** [*read-only*]: Front panel INV-TTL input state
  Line state at board input
  Bit 0 – channel 1
  Bit 1 – channel 2
  etc.

- **REAR** [*read-only*]: Rear panel input state
  Line state at board input
  Bit 0 – channel 1
  Bit 1 – channel 2
  etc.

- **FRONTFS** [*read-only*]: Front panel input failsafe state
  High if line is in failsafe mode (e.g., no cable plugged in)
  Bit 0 – channel 1
  Bit 1 – channel 2
  etc.

- **FRONTINVFS** [*read-only*]: Front panel inverter input failsafe state
  High if line is in failsafe mode (e.g., no cable plugged in)
  Bit 0 – channel 1
  Bit 1 – channel 2
  etc.

- **REARFS** [*read-only*]: Rear panel input failsafe state
  High if line is in failsafe mode (e.g., no cable plugged in)

Bit 0 – channel 1
Bit 1 – channel 2
etc.

A.1.43 OSWR - Other Switch Register

31	30	29	28	27	26	25	24
SWITCHES[31:24]							
23	22	21	20	19	18	17	16
SWITCHES[23:16]							
15	14	13	12	11	10	9	8
SWITCHES[15:8]							
7	6	5	4	3	2	1	0
SWITCHES[7:0]							

- **SWITCHES** [*read-only*]: Switch state
  1 – switch is ON
  0 – switch is OFF

A.1.44 UIDLR - 32 LS bits of 1-wire thermometer ID

31	30	29	28	27	26	25	24
UIDLR[31:24]							
23	22	21	20	19	18	17	16
UIDLR[23:16]							
15	14	13	12	11	10	9	8
UIDLR[15:8]							
7	6	5	4	3	2	1	0
UIDLR[7:0]							

- **UIDLR** [*read-only*]: LS bits of 1-wire DS18B20U thermometer ID

A.1.45 UIDHR - 32 MS bits of 1-wire thermometer ID

31	30	29	28	27	26	25	24
UIDHR[31:24]							
23	22	21	20	19	18	17	16
UIDHR[23:16]							
15	14	13	12	11	10	9	8
UIDHR[15:8]							
7	6	5	4	3	2	1	0
UIDHR[7:0]							

- **UIDHR** [*read-only*]: MS bits of 1-wire DS18B20U thermometer ID

A.1.46 TEMPR - Temperature Resgister

Raw temperature data from the one wire DS18B20U. The register is 2-bytes long; it translates to ^oC as follows: Temp = register value / 16.0

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
TEMPR[15:8]							
7	6	5	4	3	2	1	0
TEMPR[7:0]							

- **TEMPR** [*read-only*]: TEMP
  Current on-board temperature

A.2 MultiBoot controller

Base address: 0x100

Offset	Reset	Name	Description
0x0	0x00000000	CR	Control Register
0x4	0x00000000	SR	Status Register
0x8	0x00000000	GBBAR	Golden Bitstream Base Address Register
0xc	0x00000000	MBBAR	MultiBoot Bitstream Base Address Register
0x10	0x10000000	FAR	Flash Access Register

A.2.1 CR – Control Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	IPROG	IPROG_UNLOCK
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	RDCFGREG	CFGREGADR[5:0]					

- **CFGREGADR** [*read/write*]: Configuration register address
  Address of FPGA configuration register to read.
- **RDCFGREG** [*write-only*]: Read FPGA configuration register
  1 – Start FPGA configuration register sequence.
  0 – No effect.
- **IPROG_UNLOCK** [*read/write*]: Unlock bit for the IPROG command
  1 – Unlock IPROG bit.
  0 – No effect.
- **IPROG** [*read/write*]: Start IPROG sequence
  1 – Start IPROG configuration sequence
  0 – No effect
  This bit needs to be unlocked by writing the IPROG_UNLOCK bit first.
  A write to this bit with IPROG_UNLOCK cleared has no effect.
- **Unimplemented bits**: write as '0', read undefined

A.2.2 SR – Status Register

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	WDTO	IMGVALID
15	14	13	12	11	10	9	8
CFGREGIMG[15:8]							
7	6	5	4	3	2	1	0
CFGREGIMG[7:0]							

- **CFGREGIMG** [*read-only*]: Configuration register image
  Image of the FPGA configuration register at address CFGREGADR (see Configuration Registers section in Xilinx UG380 [8]); validated by IMGVALID bit

- **IMGVALID** [*read-only*]: Configuration register image valid
  1 – CFGREGIMG valid
  0 – CFGREGIMG not valid;

- **WDTO** [*read/write*]: MultiBoot FSM stalled at one point and was reset by FSM watchdog
  1 – FSM watchdog fired
  0 – FSM watchdog has not fired

- **Unimplemented bits**: write as '0', read undefined

A.2.3 GBBAR – Golden Bitstream Base Address Register

31	30	29	28	27	26	25	24
BITS[31:24]							
23	22	21	20	19	18	17	16
BITS[23:16]							
15	14	13	12	11	10	9	8
BITS[15:8]							
7	6	5	4	3	2	1	0
BITS[7:0]							

- **BITS** [*read/write*]: Bits of GBBAR register
  31..24 – Read or fast-read OPCODE of the flash chip (obtain it from the flash chip datasheet)
  23..0 – Golden bitstream address in flash

- **Unimplemented bits**: write as '0', read undefined

A.2.4 MBBAR – MultiBoot Bitstream Base Address Register

31	30	29	28	27	26	25	24
BITS[31:24]							
23	22	21	20	19	18	17	16
BITS[23:16]							
15	14	13	12	11	10	9	8
BITS[15:8]							
7	6	5	4	3	2	1	0
BITS[7:0]							

- **BITS** [*read/write*]: Bits of MBBAR register
  31..24 – Read or fast-read OPCODE of the flash chip (obtain it from the flash chip datasheet)
  23..0 – MultiBoot bitstream start address in flash

- **Unimplemented bits**: write as '0', read undefined

A.2.5 FAR – Flash Access Register

31	30	29	28	27	26	25	24
-	-	-	READY	CS	XFER	NBYTES[1:0]	
23	22	21	20	19	18	17	16
DATA[23:16]							
15	14	13	12	11	10	9	8
DATA[15:8]							
7	6	5	4	3	2	1	0
DATA[7:0]							

- **DATA** [*read/write*]: Flash data field
  23..16 – DATA[2]; after an SPI transfer, this register contains the value of data byte 2 read from the flash
  15..8 – DATA[1]; after an SPI transfer, this register contains the value of data byte 1 read from the flash
  7..0 – DATA[0]; after an SPI transfer, this register contains the value of data byte 0 read from the flash

- **NBYTES** [*read/write*]: Number of DATA fields to send and receive in one transfer:
  0x0 – Send 1 byte (DATA[0])
  0x1 – Send 2 bytes (DATA[0], DATA[1])
  0x2 – Send 3 bytes (DATA[0], DATA[1], DATA[2])

- **XFER** [*write-only*]: Start transfer to and from flash
  1 – Start transfer
  0 – Idle

- **CS** [*read/write*]: Chip select bit
  1 - Flash chip selected (CS pin low)
  0 - Flash chip not selected (CS pin is high)
- **READY** [*read-only*]: Flash access ready
  1 - Flash access completed
  0 - Flash access in progress
- **Unimplemented bits**: write as '0', read undefined

References

[1] "Converter board common gateware project page on OHWR." http://www.ohwr.org/projects/conv-common-gw.

[2] "CONV-TTL-BLO Project Page on OHWR." http://www.ohwr.org/projects/conv-ttl-blo.

[3] T.-A. Stana, "CONV-TTL-BLO User Guide." http://www.ohwr.org/documents/263.

[4] "CONV-TTL-BLO on CERN EDMS." https://edms.cern.ch/nav/EDA-02446.

[5] "Converter board common gateware specification on OHWR." http://www.ohwr.org/documents/352.

[6] T.-A. Stana, "CONV-TTL-BLO Hardware Guide." http://www.ohwr.org/documents/282, 07 2013.

[7] "SDB specification v1.1." http://www.ohwr.org/documents/256.

[8] Xilinx, "UG380 - Spartan-6 Configuration Guide." http://www.xilinx.com/support/documentation/user_guides/ug380.pdf, Jan. 2013. v2.5.