

CONV-TTL-BLO

User Guide

Theodor-Adrian Stana
BE-CO-HT

June 7, 2013



Revision history

Date	Version	Change
June 7, 2013	1.0	Created document

Contents

1	Introduction	2
2	Front and rear panels	4
2.1	Front panel	4
2.1.1	System status LEDs	4
2.1.2	SFP connector	6
2.1.3	TTL triggers	6
2.1.4	Repeated TTL pulses	6
2.1.5	General-purpose	6
2.2	Rear panel	6
3	Pulse signal types	8
4	On-board switches	9
5	Communicating with the CONV-TTL-BLO	10
5.1	The VME64x I ² C protocol	10
5.2	Accessing the board through Telnet	10
	Appendices	12
A	Getting started with the CONV-TTL-BLO	12

List of Figures

1	Symplified block diagram of a TTL to blocking conversion system	2
2	CONV-TTL-BLO panel (front panel)	5
3	CONV-TTL-BLO-RTM panel (rear panel)	7
4	Pulse signal characteristics	8
5	Byte order in the VME64x I ² C protocol	10

List of Tables

1	System status LEDs on CONV-TTL-BLO front panels	4
2	Pulse characteristics by type	8
3	Switches on CONV-TTL-BLO	9
4	The <i>readreg</i> and <i>writereg</i> commands	11

List of abbreviations

<i>FPGA</i>	Field-Programmable Gate Array
<i>RTM</i>	Rear Transition Module
<i>RTMM</i>	RTM Motherboard
<i>RTMP</i>	RTM Piggyback
<i>SFP</i>	Small Form-factor Pluggable (connector)

1 Introduction

CONV-TTL-BLO is a board intended for replicating TTL and blocking pulses. The main features of the board are:

- VME64x form-factor
- Six independent pulse replication channels
 - TTL to TTL
 - TTL to blocking
 - Blocking to TTL
 - Blocking to blocking
- Pulse LEDs for each replication channel
- Four general-purpose inverter channels
- SFP connector for White Rabbit [1]
- Can use I²C lines on VME P1 connector for remote monitoring
- Status LEDs

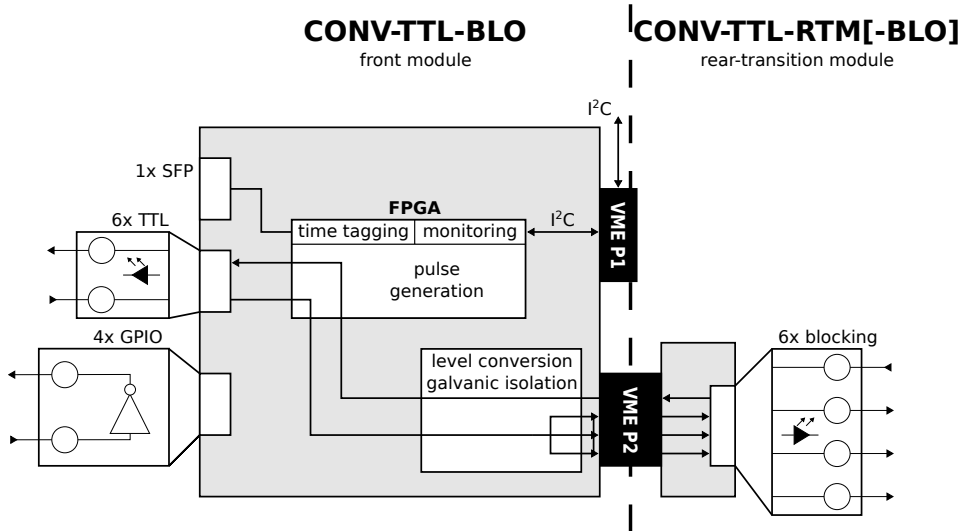


Figure 1: Symplified block diagram of a TTL to blocking conversion system

CONV-TTL-BLO is a VME64x front-module that can be used standalone as a TTL or TTL-BAR pulse repeater using the six replication channels, or as a TTL to TTL-BAR converter using the four general-purpose inverter channels.

By combining the CONV-TTL-BLO with the CONV-TTL-RTM rear-transition module (RTM) and the associated CONV-TTL-BLO-RTM piggyback board in the rear part of the VME crate, a flexible six-channel pulse conversion system can be obtained. Such a system is shown in Fig. 1. TTL pulses arriving on an input TTL channel are regenerated in the FPGA and sent on the channel's TTL output, as well as on the three blocking outputs on the RTM. Similarly, blocking pulses arriving on a blocking input channel are regenerated in the FPGA and replicated on both the TTL and blocking outputs of the channel.

2 Front and rear panels

Two panels exist in the context of the pulse repeater boards. The first of these is the *front panel*, which corresponds to CONV-TTL-BLO boards and offers various status LEDs, as well as various connectors for TTL and TTL-BAR (see Sec. 3) pulses and White Rabbit. The second is the *rear panel*, located on the other side of the VME backplane and corresponding to CONV-TTL-RTM-BLO boards. The rear panel offers blocking pulse connectors and status LEDs for pulse arrival confirmation.

2.1 Front panel

The front panel of CONV-TTL-BLO boards is shown in Fig. 2. It consists of status LEDs and several ports, divided in four sections from top to bottom:

- System status LEDs;
- Small form-factor pluggable (SFP) connector;
- TTL pulse connectors;
- TTL-BAR pulse connectors.

2.1.1 System status LEDs

In the current version of the CONV-TTL-BLO boards, only several of the system status LEDs present on the board are used, due to limited firmware support in the FPGA. The implemented LEDs are presented in Table 1. Unimplemented system status LEDs are **off** by default.

Table 1: System status LEDs on CONV-TTL-BLO front panels

LED	Description
<i>PW</i>	Power LED. Lights <i>green</i> when a valid CONV-TTL-BLO firmware is loaded to the FPGA.
<i>ERR</i>	Error LED. Lights <i>red</i> when no rear transition module board is present.
<i>TTL</i>	TTL status LED. Lights <i>green</i> when TTL logic is selected via the 8 th position of the on-board selection switch.
<i>I2C</i>	I ² C status LED. Lights <i>red</i> until an I ² C transfer has taken place. Once either a read or a write is successfully completed, the I ² C status LED lights <i>green</i> to signal the communication is up.

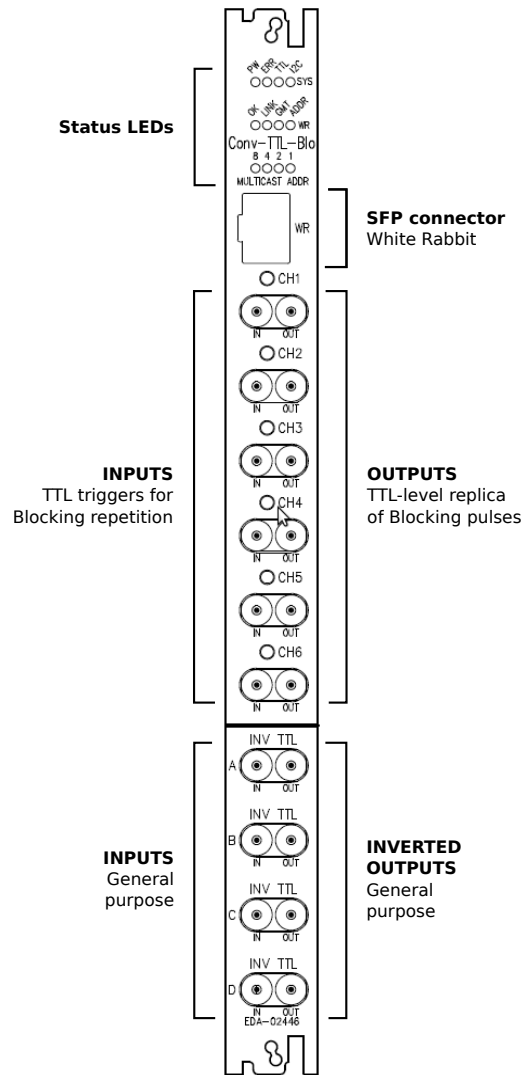


Figure 2: CONV-TTL-BLO panel (front panel)

2.1.2 SFP connector

This connector is used to add White Rabbit support to the CONV-TTL-BLO boards. If an optic fibre cable is connected to this socket, White Rabbit precise time-stamping can be added to CONV-TTL-BLO. Four status LEDs above the connector are provisioned to show the status of the White Rabbit link.

White Rabbit is currently not supported by the firmware.

2.1.3 TTL triggers

One side of the dual LEMO 00 (type EPY) connector on the CONV-TTL-BLO boards are used for the TTL trigger inputs. By connecting an external trigger source to one of these connectors, a blocking pulse is generated at the rear panel and a TTL-level pulse is generated at the front panel. The triggers can be either TTL, or TTL-BAR level.

All input channels are line-terminated with 50Ω resistors.

2.1.4 Repeated TTL pulses

The other side of the dual LEMO 00 connector is used to output a TTL-level replica of the blocking pulse received at the rear panel, or of the trigger signal arrived on the front panel. The pulse width of this output is similar to the pulse output in the rear panel; the rise time and top pulse level are however different from the blocking output.

Note that the TTL outputs always contain TTL signals (see Sec. 3).

When the pulse is output, the LED of the corresponding channel blinks for 96 ms.

TTL output lines are not internally terminated.

2.1.5 General-purpose

Four dedicated inverted-TTL connectors can be found in the lower part of the front panel. The inputs can be either TTL, or TTL-BAR, as selected by the TTL selection switch. The outputs of these connectors are always TTL-BAR level (see Sec. 3).

TTL-BAR inputs are internally terminated with 50Ω resistors.

TTL-BAR outputs are not internally terminated and not replicated on the blocking outputs.

2.2 Rear panel

The rear panel on CONV-TTL-BLO-RTM boards is shown in Fig. 3. It contains the input and output connectors, as well as pulse status LEDs for six blocking-level pulse channels. A blocking-level pulse at the input

connector of a channel is repeated at the three outputs of the same channel in blocking level and TTL level at the output connector of the corresponding channel on the front panel.

When a pulse is repeated on the output connector of a channel, the pulse status LED is lit for 96 ms.

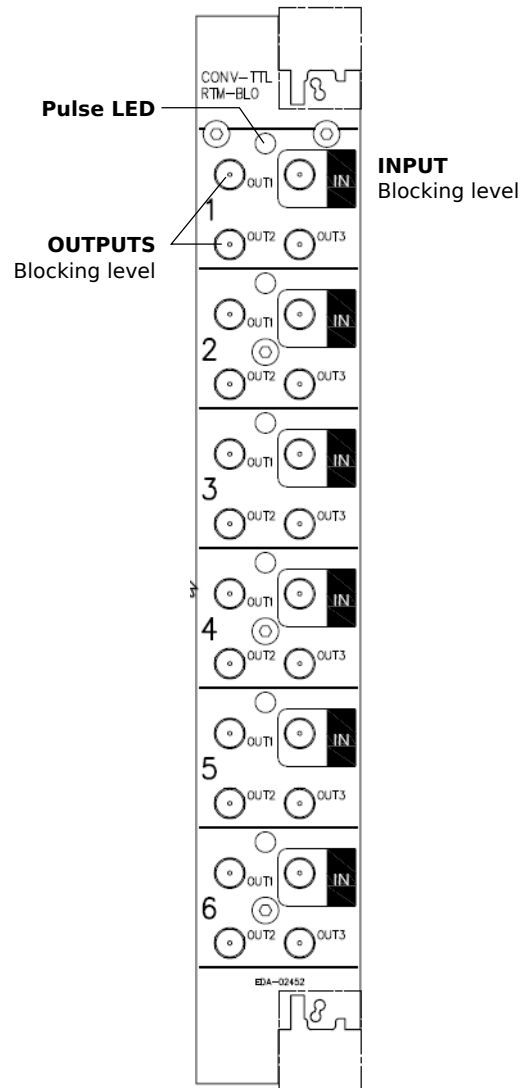


Figure 3: CONV-TTL-BLO-RTM panel (rear panel)

3 Pulse signal types

In order for CONV-TTL-BLO boards to work as repeaters, logic is implemented in the on-board FPGA that reacts to a trigger at either the rear or front panel and generates a pulse at the output. This section presents the pulses generated by the CONV-TTL-BLO board.

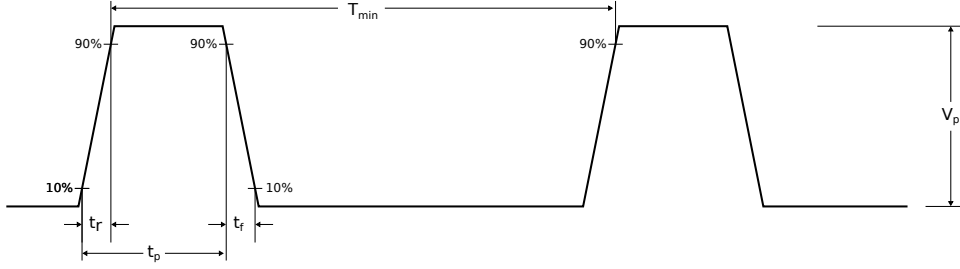


Figure 4: Pulse signal characteristics

The different characteristics of a pulse are defined in Fig. 4. Three pulse types may exist, depending on signal amplitude, rise and fall times; pulse widths and frequencies are the same. TTL and TTL-BAR pulses are input and output on the front panel of the boards. TTL-BAR is essentially an inverted version of TTL signals. Blocking pulses [2] are input and output on the rear panel (via a CONV-TTL-RTM). The characteristics of the three types of pulses are shown in Table 2.

Table 2: Pulse characteristics by type

Symbol	Parameter	TTL	Blocking	Unit
V_p	Pulse amplitude, $V_{p,max} - V_{p,min}$	3.3	24	V
$V_{p,max}$	Pulse max. amplitude	3.3	24	V
$V_{p,min}$	Pulse max. amplitude	0	0	V
t_p	Pulse width	1.2	1.2	μs
T_{min}	Min. period of pulse signal	3.33	3.33	μs
t_r	Rise time	6.4	140	ns
t_f	Fall time	3	160	ns

4 On-board switches

There are eight switches provided on-board the CONV-TTL-BLO board. Of these eight, two are used by the current version of the FPGA firmware, as shown in Table 3.

Table 3: Switches on CONV-TTL-BLO	
Switch	Description
SW1.1	<p>Selects the type of pulse generated at the output</p> <p>ON – <i>Type 1</i>, glitch-sensitive, without output jitter</p> <p>OFF – <i>Type 2</i>, glitch-filtered, with output jitter</p>
SW2.4	<p>TTL/TTL-BAR input selection switch</p> <p>ON – TTL pulses arrive on TTL and TTL-BAR inputs</p> <p>OFF – TTL-BAR pulses arrive on TTL and TTL-BAR inputs</p> <p><i>Note:</i> This switch applies the selection to the whole board, i.e., only TTL or TTL-BAR pulses may be input on all channels of the board, not both</p>

SW1.1 is used to select whether the pulse generated at the output should have jitter or not. Based on application and the environment in which the boards are used, the pulse generated can be one of two types. *Type 1* pulses have no output jitter, but can be sensitive to glitches, i.e., a glitch at the input may trigger the generation of a pulse. *Type 2* pulses are sensitive to glitches up to 40 ns long, but there is jitter at the output.

SW2.4 is used to select the type of signal that would arrive on the TTL and TTL-BAR inputs. The switch is valid board-wide, i.e., if it is set for TTL inputs (**ON**), TTL signals should be input on both TTL and TTL-BAR inputs. Inputting TTL-BAR signals on a channel while the TTL/TTL-BAR selection switch is set to TTL invalidates the operation guaranteed by SW1.1 and introduces a delay on the generated pulse.

5 Communicating with the CONV-TTL-BLO

It is possible to communicate to the CONV-TTL-BLO remotely via the VME P1 I²C interface. This section describes how to connect to the VME64x crate and communicate to the board.

There is currently limited support for this functionality, but work is under way to extend it. The only functionality provided at this point is retrieving the CONV-TTL-BLO ID string (**BLO2**) from the board ID register at address 0x000 (*reg* 1).

5.1 The VME64x I²C protocol

In order to connect to the VME crate the CONV-TTL-BLO board is placed in, a higher-level protocol based on I²C is defined. The protocol uses the I²C lines on the VME P1 connector.

By this protocol, 2¹² (12 address bits) 32-bit registers can be read from or written to in a byte by byte basis. The order in which bytes are sent is shown in Fig. 5. First, as with any I²C transfer, the I²C address of the device is sent. After the I²C address has been acknowledged by the device, the twelve-bit register address is sent with the most significant byte first (little endian address transmission). After the address is sent, four data bytes are sent, starting with the least-significant byte (big-endian data transmission).

byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6
I ² C Addr	A1	A0	D0	D1	D2	D3

Figure 5: Byte order in the VME64x I²C protocol

More information about the protocol, including how the I²C address of the board is obtained, can be found in the protocol specification [3].

5.2 Accessing the board through Telnet

By opening up a telnet session with the VME crate, values can be read or written to registers on the CONV-TTL-BLO. Two telnet commands (see Table) can be used in this purpose. As their names suggest, *readreg* reads a board register, whereas *writereg* writes to a board register. The VME crate handles translating these two telnet commands to I²C accesses on the VME P1 connector.

The register numbers are integer numbers starting from one. The actual addresses sent to the board are word-aligned starting from 0x000. For example, *reg* 1 would translate to address 0x000, *reg* 2 to 0x004 and so on. A simple formula for translating *reg* numbers to on-board addresses is:

$$addr = (reg - 1) * 4$$

Table 4: The *readreg* and *writereg* commands

Command	Description
<code>writereg <i>slot reg val</i></code>	Writes the value <i>val</i> to register number <i>reg</i> of board in slot number <i>slot</i>
<code>readreg <i>slot reg</i></code>	Returns the value of register number <i>reg</i> of board in slot number <i>slot</i>

An example of retrieving the CONV-TTL-BLO ID from *reg* 1 of a CONV-TTL-BLO plugged into VME slot 1 of the crate *some-crate* is given below:

```
tstana@tstana-unit:~$ telnet some-crate
Trying 137.138.192.90...
Connected to some-crate.cern.ch.
Escape character is '^]'.
login:user
password:****
%>readreg 1 1
  Read Data: 424C4F32
%>
```

First, a telnet connection is made with the crate, after which the *readreg* command is issued with address 0x000. The value of the register can be confirmed to be the hex value of the ASCII string **BLO2**, so the board is indeed present in the slot.

Another example of running the same command, this time with the board removed from the crate, is given below. As expected, when the board is removed, it can no longer acknowledge the I²C access, thus the message:

```
Connected to some-crate.cern.ch.
Escape character is '^]'.
login:user
password:****
%>readreg 1 1
  Not Acknowledged!
%>
```

Appendices

A Getting started with the CONV-TTL-BLO

This section provides a description on testing CONV-TTL-BLO boards for basic functionality. The following steps should be followed in order to test the board.

1. (**Optional**) Plug in a CONV-TTL-RTM board with a CONV-TTL-RTM-BLO piggyback into the rear part of the VME crate.
2. Remove the CONV-TTL-BLO board from its ESD-protecting bag;
3. If TTL or TTL-BAR pulses are to arrive on the front panel inputs, set the TTL switch to the appropriate position
 - TTL pulses – set the switch to the **ON** position
 - TTL-BAR pulses – set the switch to the **OFF** position (default)
4. Insert the CONV-TTL-BLO board into the VME crate and power on the crate;
5. Check that the **PW** status LED is lit **green**. If there is no RTM in the rear side of the crate, the **ERR** LED will light **red**. The **TTL** status LED should also be lit **green** if you set the TTL switch to the **ON** position in the previous step.
6. Input a TTL (or TTL-BAR) signal into a front panel input channel. When a pulse arrives on the input, it is replicated on the output of the same channel. If an RTM board is present in the rear part of the VME crate, the pulse will also be replicated on the three blocking outputs of the same channel on the rear-panel. The channel pulse LED on both the front and rear panels flash briefly when a pulse arrives.
7. (**Optional**) Input a blocking signal on a rear panel channel; The pulse LED of the channel will flash and the pulse will be replicated on the three blocking outputs of the same channel, as well as the TTL channel output on the front panel. If the TTL switch is **OFF**, the pulse is replicated in TTL-BAR.

References

- [1] “White Rabbit.” <http://www.ohwr.org/projects/white-rabbit>.
- [2] C. G. Soriano, “Standard Blocking Output Signal Definition for CTDAH board,” Sept. 2011. <http://www.ohwr.org/documents/109>.
- [3] ELMA, “Access to board data using SNMP and I2C.” <http://www.ohwr.org/documents/227>.