Standard Blocking Repeater in VME64x Format

Functional Specifications

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

September 21, 2011

Acknowledgements This document agregates the functional specifications of the TTL, inverted-TTL and Standard Blocking [1] to Standard Blocking output repeater for the substitution of the following boards: 8 and 16 Channel Repeaters, Level Converter and LASB-TTL-BLO.

System Description and Purpose The Standard Blocking Pulse Converter is a set of two VME64x boards which has 6 channel repeaters. Each channel repeater is a device able to translate TTL and inverted-TTL signals into Standard Blocking pulses and regenerate Standard Blocking ones. Differing from previous versions, a FPGA is included in the Pulse Converter to add control to the repeater and communication through a serial interface in the VME64x connector. The aim of this document is setting the specifications needed for the renovation of these devices due to the difficulties faced when it comes to maintenance and repairing.

History of changes

This document version has been checked by:

This document version has been approved by:

Date	Pages	Changes
September 14, 2011	All	Initial submission
September 21, 2011	All	Added references to Standard Blocking

Contents

1 General schema

The Standard Blocking Pulse Converter consists on two boards, CTDAH and CTARA. The decision of splitting up the Pulse Converter lies in easing substition and repairing of damaged parts and help the operator when interconnecting the inputs and the outputs.
A Pulse Converter can host a maximum of 6 channel repeaters due to the size limitations of the Rear Panel dimensions.
The Front Board, CTDAH is the board that holds the majority of the active parts, such as the FPGA and the Pulse Converter Units. This board is responsible of the control and monitoring of the Pulse Converter activity.
The Rear Board, CTARA, holds mainly passive parts, such as the inputs and outputs.

Board	Location	Parts	Connectivity
CTDAH	FRONT	Mainly Active	VME64x Serial Control Lines
CTARA	REAR	Mainly Passive	Input\Output Connectors

This can be better understood with the following schema:

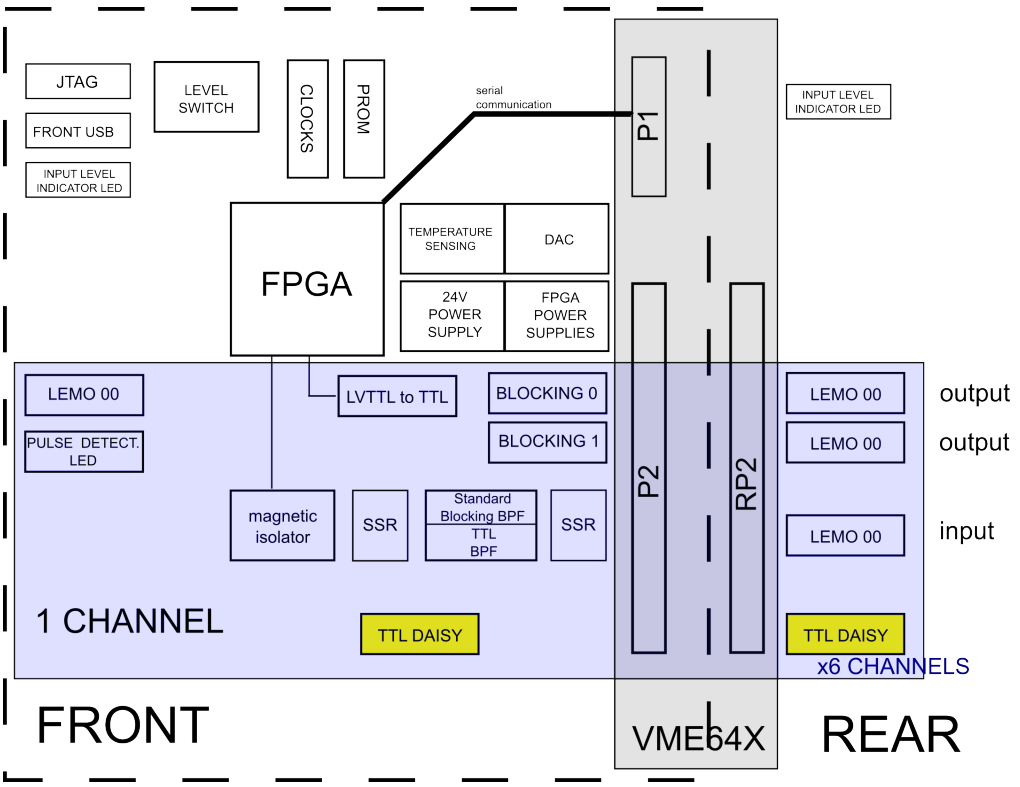

Figure 1: Functional block schema

2 Panels

Following the previous schema, the Front and Rear Panels are as follows: [RED PARTS TO BE DISCUSSED]

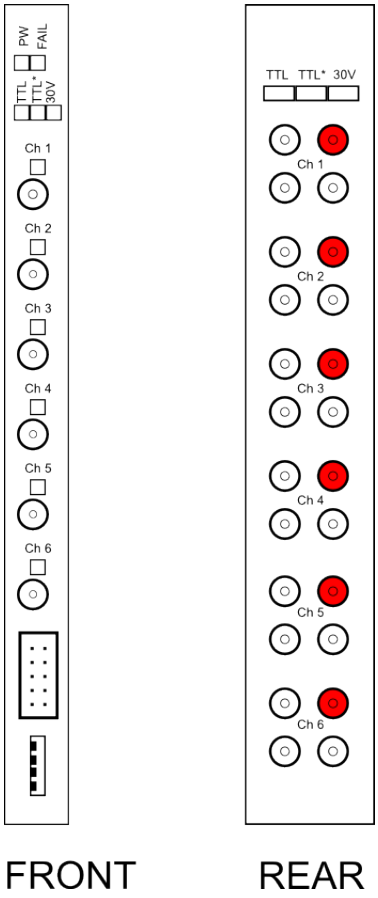

Figure 2: Front and Rear Panel Sketch

2.1 Front Panel

The front board, CTDAH, offers in the Front Panel:

- Power OK and System Fail LEDs
- An input signal level indicator: three LEDs.
- For every channel: a pulse activity indicator and a TTL output level pulse connector. It can be used either as an output or for monitoring purposes.

- A JTAG interface for programming the FPGA
- An USB interface for allow communication with the FPGA

2.2 Rear Panel

The Rear Board, CTARA, offers in the Rear Panel:

- As in the Front Panel, the input signal level must be issued though LEDs so as to help the operator.
- The Rear Panel must hold six channels. Each channel can have either 3 or 5 LEMO 00 connectors [RED COLOR IN FIGURE 2 TO BE DISCUSSED].

  Three LEMO 00 connectors: one input, two outputs.

  Five LEMO 00 connectors: one input, two outputs and two daisy-chain -one input and one output- connectors.

3 Power-up and System Monitoring

The System Monitor Unit consists on a HDL core that control critical parts of both the front and rear boards.

3.1 Power Supplies

The power supply domains must be monitored by the FPGA trough a DAC and a power-up sequence must be specified so as to improve reliability. Reports must be available through VME64x serial lines.

3.2 Board Temperature

The board temperature must be monitored by the FPGA. Reports must be available through VME64x serial lines.

3.3 Daisy-chain detection

Daisy-chain detection is a solution offered to the problem of bypassing the same signal to different channels.

How it works By physically configuring the Rear Board it can be specified whether a channel is bypassing its signal or not. Two possible implementations are studied:

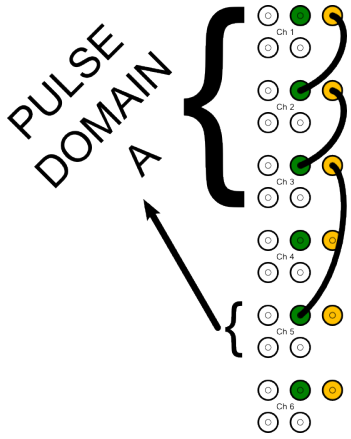

Figure 3: Daisy-chain: Option A

Option	Connectors per channel	Activate daisy-chain
A	5 LEMO 00	Wired
B	3 LEMO 00	Switch

In **Option A** there are two additional connectors in every channel of the Rear Panel. One of this connectors is a daisy-chain input, the other is a daisy-chain output. So as to enable the daisy-chain operation, the operator must connect one daisy-chain output of one channel with a daisy-chain input of a different channel. It is not necessary that the channels are contiguous, thus providing more flexibility to the operator when it comes to the installation process.
After the installation, a daisy-chain detection process must be carried out by the FPGA. Instead of outputting the signal through the daisy-chain connectors, it is internally loopbacked between the detected daisy-chained channels. By internally loopbacking, the associated delay is reduced. A maximum FPGA bypassing delay must be specified.
Option B represents a different approach to the same problem. In the Rear Panel, instead of having two dedicated daisy-chain connectors, there is a switch in every channel. This switch allows loopbacking the signal of the current channel to, and only to, the next channel.
After setting up the switch in the desired position, the FPGA will detect its state and will proceed -or not- to internally loopback the signal to the next channel.

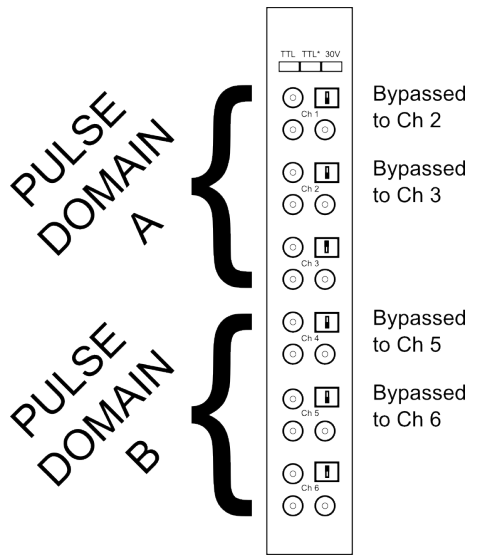

Figure 4: Daisy-chain: Option B

What it gets rid of Input signal integrity is improved. Reflections are reduced because no stubs are added and line is matched at the end.

Added value This solution is simple and more economic than the existing

one. The most important added value is that an effective **improving in the remote detection of failures** can be addressed. If the daisy-chain topology of all the channels is known for each repeater, a complete network schema can be inferred from this. Periodic requests of the pulse repeater state will be gathered through the serial VME64x interface. Then they can be centrally analyzed to determine where a problem is happening.
For instance, if a central machine detects a problem in one repeater of the network, it can report the error as a quadruple {VME crate ID, VME Pulse Repeater Board ID, VME Pulse Repeater Channel ID, Problem ID}.
As this remote implementation is not trivial, it must be carefully studied. The hardware part should offer the basis for a prospective implementation of the system.

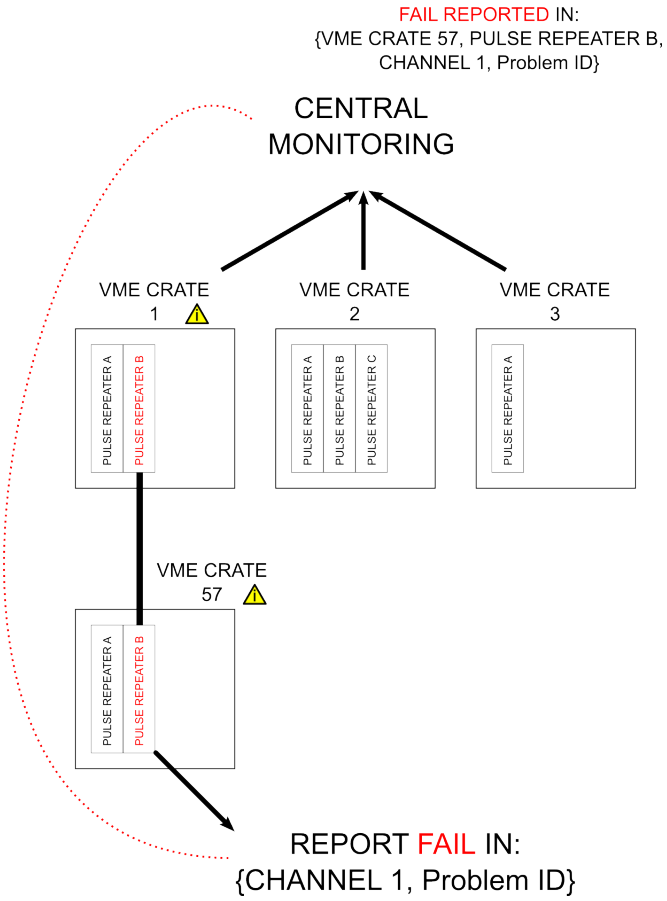

Figure 5: Remote Failure Detection Capability

4 FPGA Control and Pulse Converter Unit

FPGA control will handle all the system logs of the input pulses and parameters of the output ones: pulse width and time between pulses. The configuration of this two paramters is important in two aspects:

Pulse width Initially the output pulse width is set to 1.2 μs, following the Standard Blocking definition [1]. Input pulses wider than 1.2 μs will be cropped to 1.2 μs output ones. However, as different applications could require different pulse width, a configuration register in the FPGA will be used to add this functionality.
In electrical terms, changing the pulse width out of bounds will produce that the current in the transformer will be higher than the saturation current. In this case, irreversible damage will be produced. HDL control must take care of this issue.

Time between pulses Initially the time between pulses is set to 2 μs. Any pulse received within this time frame after a pulse was outputted will not be regenerated and registered as "Time between pulses violation" in the HDL control.
It is important to monitor this parameter because it can produce physical damage to the transformer. Due to the remaining magnetizing current built up during the pulse on state, the circuit needs time to draining it off thanks to a snubber circuit. Subsequently, time between pulses is dependent of the pulse width and must be set so as to not leave too much remaining current in the transformer. If the remaining current is not low and input pulses are received too often, the transformer current can build above the saturation one producing irreversible damage.

4.1 Board ID

A Board ID must be specified to distinguish between different hardware versions of the Front and Read Board. By reading the value of the board ID, different approaches can be taken in the HDL control. For instance, if we consider that a manufacturer is not able to dispatch a 1mH primary inductance transformer so that it must be replaced for a 470 μH one, both the *pulse width* and *time between pulses* parameters must be adapted to this different part.

4.2 HDL control

The FPGA control is governed by a HDL core which must calculate the bounds of the *pulse width* and *time between pulses* parameters depending on the Board ID.

Then, as previously stated, it must monitor the *input pulse width* and eventually correct the *time between pulses* so as to avoid damage to the output pulse transformer due to the remaining magnetizing current.
Furthermore, an event log should be carried out so that it can be reported back through either serial VME64x P1 lines or front USB connector.

4.3 Pulse Converter Unit

The Pulse Converter Unit is the part of the circuit that receives and adapt the input signal to the FPGA and converts a control signal from the FPGA to Standard Blocking output -it goes to the Rear Panel- and TTL –it goes to the Front Panel.

4.3.1 Input

TTL, inverter-TTL and Standard Blocking levels must be translated to pass the signal to the FPGA. *Due to attenuation produced for the long distance connections made, signals that were outputted as Standard Blocking levels will be accepted if the received high level is above 10V.* Protection Circuitry must be used. External line termination is recommended [TO BE DISCUSSED WHY NOT].

4.3.2 Output

The output circuity must generate a Standard Blocking signal [1] compliant:

Pulse Width	**Pulse Height**	**Rise Time**	**Fall Time**
1.2 μs	24V	150 $\pm$ 75 ns	350 $\pm$ 75 ns

it should be noted that the *Pulse Width* can be configured within the Standard Blocking pulse width range.

5 On board memory and FPGA reprogramming

A PROM memory must be installed to hold the FPGA bitstream. The size of the PROM memory should be at least the required to install two uncompressed FPGA bitstreams.
FPGA can be reprogrammed through the external JTAG interface or by means of the VME64x serial connection.

6 Testing methodologies

The testing process of the FPGA should cover the following areas:

- Testbench of the HDL power-up related code –power domains, daisy-chain configuration.
- Testbench of the HDL control circuit.
- Verification of the input reflection.
- Verification of the output pulse shape.

References

[1] C. Gil Soriano. Standard Blocking Output Signal Definition for CTDAH board, September 2011. http://www.ohwr.org/documents/109.