ISTITUTO NAZIONALE DI FISICA NUCLEARE

Sezione di Perugia

INFN/TC-07/11
July 24, 2007

A MULTI PURPOSE REPROGRAMMABLE NIM MODULE

Andrea Papi, Pasquale Lubrano, Ermanno Imbergamo

INFN sezione di Perugia, Dip. Scienze Fisiche Università di Perugia, 06100 Perugia, Italy

Abstract

We inform about the realization of a NIM interface module thought to replace large variety of old fashioned electronics usually present in experimental setups for Nuclear Physics. In the following note, we discuss the design guidelines along with the most relevant technical choices that we performed.

*Published by **SIS–Pubblicazioni***
Laboratori Nazionali di Frascati

1 THE IDEA

It is usual to find in Nuclear Physics laboratories NIM interface modules working basic operations such as “and-or” coincidences, signal latching, counters, encoders-decoders and many others based on the fundamental concepts of logical gates and flip-flops. On the other hand, the market provides us with commercial CPLD and FPGA which have a large amount of logical resources. The underlying idea of present application is to make use of a reprogrammable device which can be erased then loaded with the logical function that is wished to be performed at that moment, in similar way to a NIM module that can be plugged (unplugged) into (from) a NIM create. In this way, handling many modules and managing with complicate cablings is not necessary any more, but only programming the device is required. Note that, even low cost CPLDs are such that they integrate a so large amount of resources to be able to implement about every logical function may be wanted in common laboratory practice. Instead, limitations can arise from the module user interface, where only a limited number of input-output connectors and controls are feasible due to components space occupancy of the module front panel.

2 THE MODULE USER INTERFACE

Figure 1 shows a picture of the “Glue logic” module front panel. It features:

1. 16 NIM input LEMO connectors
2. 16 NIM output LEMO connectors
3. 1 NIM input LEMO connector dedicated to input clock
4. 10 leds as monitoring of 10 NIM inputs
5. 4 control switches paired to 4 monitoring leds
6. 4 control debounced switches acting as push buttons paired to 4 monitoring leds
7. 4 decimal contraves
8. 4 seven segments displays
9. 40 single ended extension pins, each one paired to ground in order to enhance immunity to noise

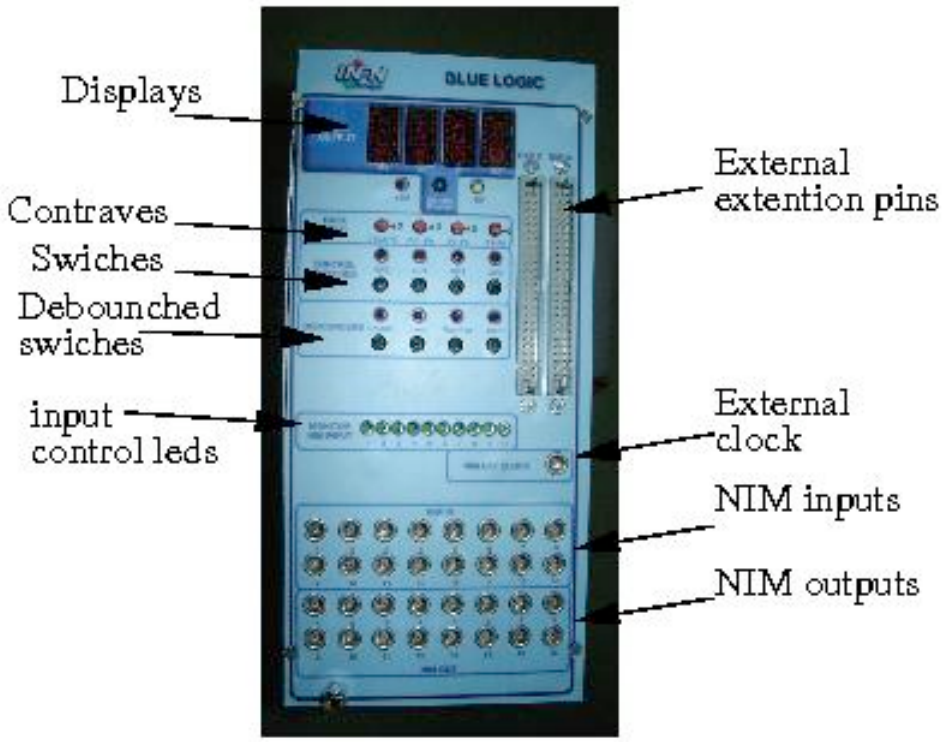

*Figure 1: the modele user interface***.**

3 A LOOK INTO MODULE INTERIOR

Figure II shows a picture of the board inside the module. Among all the components, the most relevant ones are :

1. The EPM7256SQC208, which is an erasable EPROM member of the second-generation MAX family from Altera. It is a low cost CPLD with 208 pins (164 i/o user pins), 5000 usable gates and 256 flip-flops given in a PQFP package. Obviously an FPGA would be richer of both gates and memory elements, but at price of a bigger cost and difficult BGA package to be handled in the PCB.
2. The JTAG connector is the interface between computer and CPLD for programming operation. This step can be performed using the QuartusII software from Altera or any other tool for FPGA project development, i.e. Active HDL, Mentor graphics or other.
3. An on-board quarz oscillator is connected to CPLD dedicated clock input and can be chosen of frequency as closer as possible to the actual operating frequency of the circuit which the user wants to realize (80 MHz in the present design).
4. 16 pairs of MC10H125 and MC10H124 need to translate the input NIM signals not supported by the CPLD I/O pins to TTL and viceversa (not indicated in the picture).
5. A monostable TTL multivibrator 74LS221 for each signal driving a control led (not indicated in the picture).

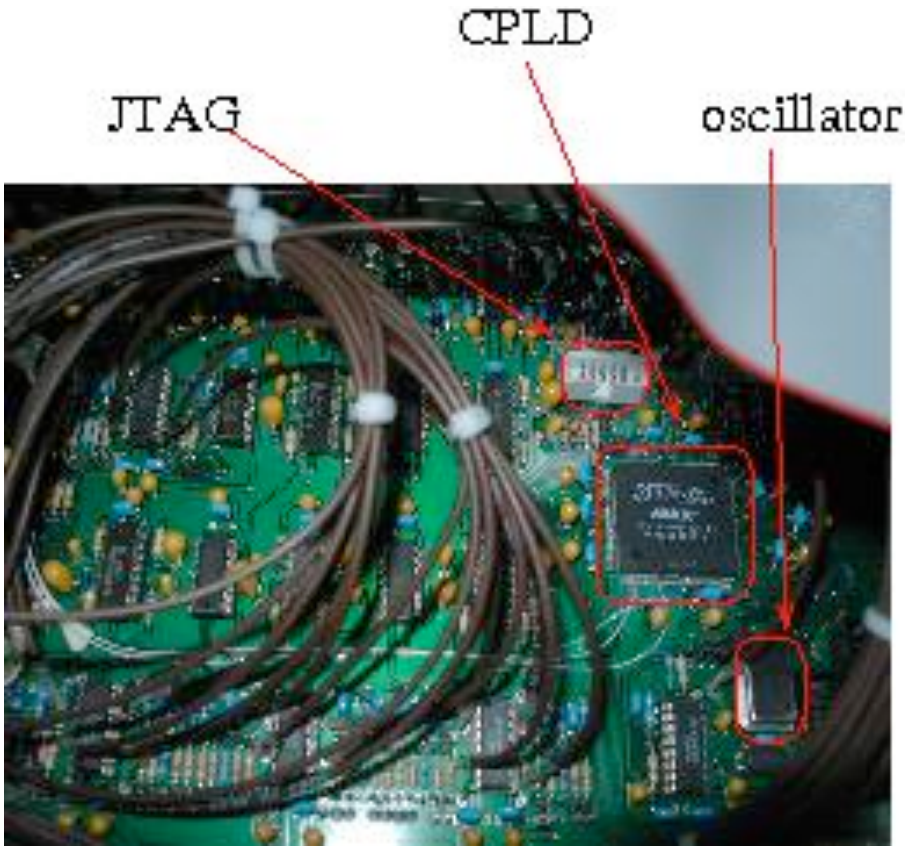

Figure 1: the module interior.

4 GENERALITIES ON MODULE APPLICATIONS

"Glue logic" module can be used as substitute of almost all the commercial NIM module usually present in laboratories for high energy Physics. The only limitation concerns discriminators, where analog signals are required, but they are not valid inputs to the CPLD (nor additional analog circuitry has been arranged in the internal board to manage with them). In addiction, a loss of performance has to be accounted for that functions usually dedicated to analog circuitries such as "timing units", which are designed to respond to an input signal with a corresponding output whose time length can be chosen in a continuous range of values. In the context of current programmable devices, the time length of the output signal can be set only stepwise, i.e. by mean of counters, by one unit of the in-board clock, leading to worse performances respect to analog circuitry. A partial fix to this would come by replacing the CPLD with a MAXII series (at cost of a more complicated board PCB due to the BGA device package), where PLLs are available and then shorter units of clock may be used internally to the device. Another advantage of using a MAXII (or even better an FPGA) would be the possibility of using internal PLLs for generating waveforms of given frequency and duty-cycle instead of counters.

On the other hand, “Glue logic” module gives access to functions that are hardly achievable by mean of commercial NIM modules. For example, the design described in the next section, performs adjustable delays to input signals, ranging from tens of nanoseconds up to in principle infinite amount of time, with the possibility of preserving the input waveform in case of short enough delay and not too fast input.

5 AN EXAMPLE OF APPLICATION

The “Glue logic” module is obviously an ideal solution for projects involving abundant usage of logical gates since the used CPLD is rich of logical gates. Anyway we want to stress that it can be used also when registers are required. In this section, we meet our twofold intent of giving to the reader a feeling of “Glue logic” module usage and its limitations. To do that, we discuss an example of application where registers are critical.

Figure 3 shows the schematic design that can be downloaded inside the CPLD device for an application where an input signal has to be delayed by an adjustable amount of time. From the point of view of “Glue logic” front panel, the implementation looks as follow:

- the input signal is connected to an input LEMO and the output (delayed) signal to an output LEMO.
- the delay time can be set by mean of the available contraves in decimal format
- the delay time can optionally be visualized in the seven segments displays (not shown in the picture).
- one of the switches is used for reset
- the reference clock can optionally be provided from an external device via the dedicated LEMO connector otherwise internal the quarz oscillator is used.

Within the CPLD, the implementation is such that:

- the input signal is rooted along a chain of registers organized in 1 block by 100 D-Flip-Flops, 9 blocks by 10 D-Flip-Flops and 9 blocks by 1 DFFs for a total usage of 199 logic cells among the 256 available ones.
- the actual input to output rooting path along the flip-flops chain is selected by multiplexers controlled by signals external to the CPLD wired to the contraves
- the rooting path is selectable by the user and determines the delay of the output signal at steps of one reference clock unit
- the output signal preserves the input waveform unless it varies faster than the reference clock.

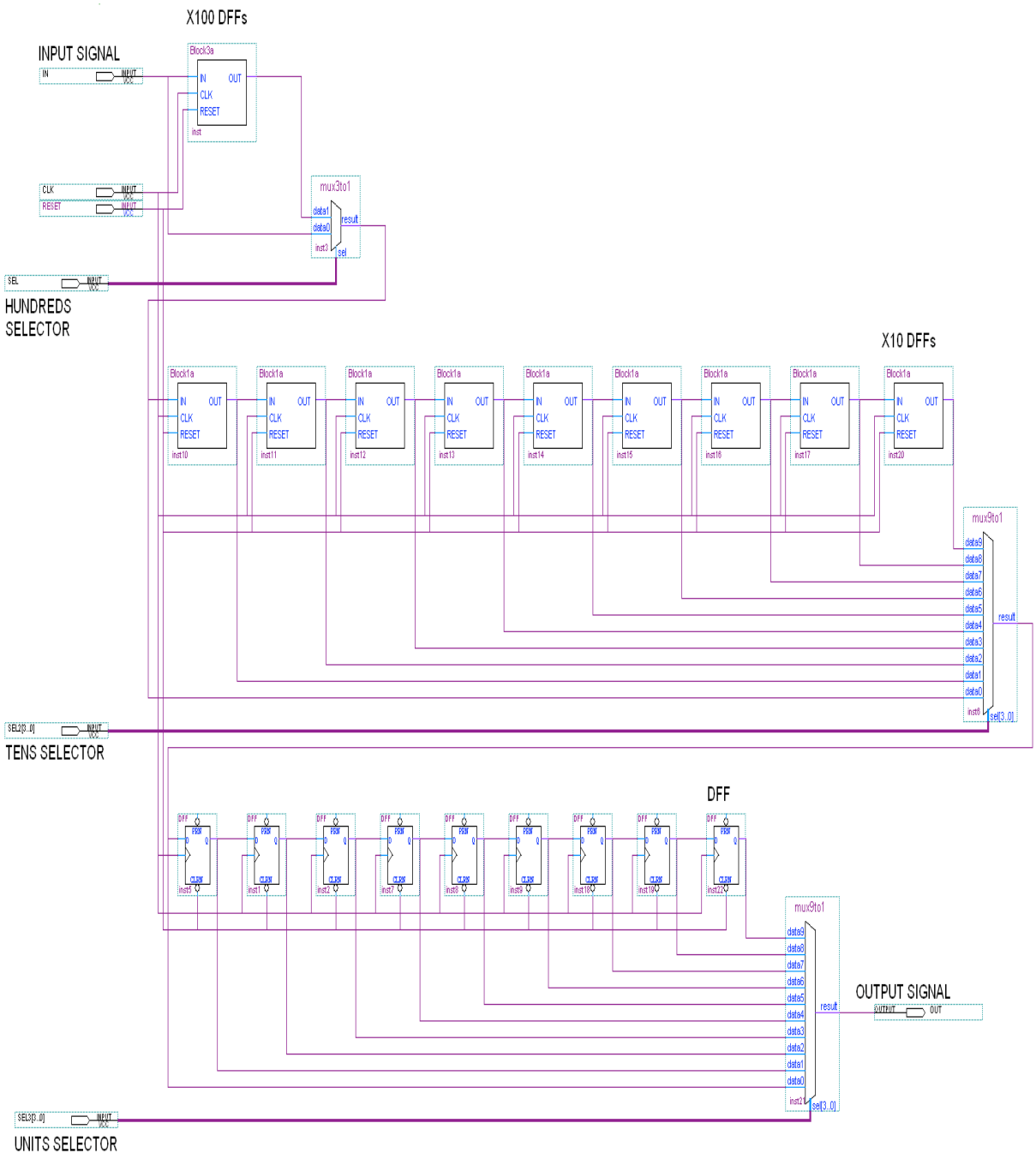

Figure 2: schematic design for an application where an input signal has to be delayed by an adjustable amount of time.

It is important to comment that in this application the poor amount of registers results in a limitation, not only in the programmable delay (199 clock units), but also in the number of

signals that can be processed. On the other hand, if the requirement of preserving the input waveform is relaxed and replaced with the softer requirement of delaying only the rising (or falling) edge of input signals then the few registers would probably not be a limitation any more. As an example, consider that by mean of counters and comparators, the 256 registers may be arranged to form 8 independent binary counters providing the possibility to delay the rising (or falling) edge of 8 independent inputs up to 2^{32} (about 4 billions) clock units with the precision of one clock unit. The inconvenient of using the "Glue logic" module in this way is that there are not enough controls in the front panel and the delays should be hard coded in the design project.

CONCLUSION

We developed a reconfigurable multi-purpose module well suited for implementation of logical operations. It can replace a large variety of usual NIM module, with the only limitation of non-capability to handle with input analog signals and the advantage to allow the development of functionalities hardly achievable by mean of commercial NIM modules. It makes use of a low cost CPLD given in PQFP package, simpler to manage respect to more evolutes programmable devices. Nonetheless, the replacement of the chosen device with an FPGA, rich of both gates and flip-flops and provided with PLLs, would allow for the development of in principle unlimited logical applications.