TATU: A FLEXIBLE FPGA-BASED TRIGGER AND TIMER UNIT CREATED ON CompactRIO FOR THE FIRST SIRIUS BEAMLINES

J. R. Piton, D. Alnajjar, D. H. C. Araujo, J. L. Brito Neto,
L. P. do Carmo, L. C. Guedes, M. A. L. de Moraes,
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

Abstract

In the modern synchrotron light sources, the higher brilliance leads to shorter acquisition times at the experimental stations. For most beamlines of the fourth-generation source SIRIUS (the fourth-generation particle accelerator in Campinas, Brazil), it was imperative to shift from the usual software-based synchronization of operations to the much faster triggering by hardware of some key equipment involved in the experiments. As a basis of their control system for devices, the SIRIUS beamlines have standard CompactRIO controllers and I/O modules along the hutches. Equipped with an FPGA (Field Programmable Gate Array) and a hard processor running Linux Real-Time, this platform could deal with the triggers from and to other devices, in the order of ms and µs. TATU (Time and Trigger Unit) is an FPGA/real-time software combination running in a CompactRIO unit to coordinate multiple triggering conditions and actions. TATU can be either the master pulse generator or the follower of other signals. Complex trigger pattern generation is set from a user-friendly standardized interface. EPICS (Experimental Physics and Industrial Control Systems) process variables [1], by means of LNLS Nheengatu [2], are used to set parameters and to follow the execution status. The concept and first field test results in at least four SIRIUS beamlines are presented.

INTRODUCTION

LNLS Tatu (Timing and Trigger Unit), being mostly FPGA code running in a CompactRIO (CRIO) device (NI CRIO models 9045/9049/9035/9039), combines C-Series I/O modules to work as trigger detectors, pulse generators and a recorder of analog and digital input readouts. Tatu manages digital signals to detect events and to produce actions for synchronized operations at a beamline. That comprises sequences of operations on shutters, motorized devices and detectors, allowing the data acquisition at beamlines to be as fast as possible (also known as "fly-scan"). An example of signal combination to produce different outputs is shown in Figure 1.

Some premises that have been considered in the pursuing of this project are as follows:

- it was already decided that each hutch in a SIRIUS beamline would count on at least one device to make available TTL signal input/output and CompactRIO was the selected standard.
- additional small pieces of software, either in FPGA or Real-Time, could be included for dedicated jobs (like filtering analog readouts), in parallel to the triggering management.
- the access to the configuration parameters would happen through Nheengatu [2], so their manipulation would be immediately made available under EPICS.
- at least for the initial operation of the first beamlines, a maximum pulse rate of 4 kHz would fully meet the requirements.
- it should be highly flexible to let different conditions and outputs to be combined, just by software configuration (no FPGA recompilation needed).

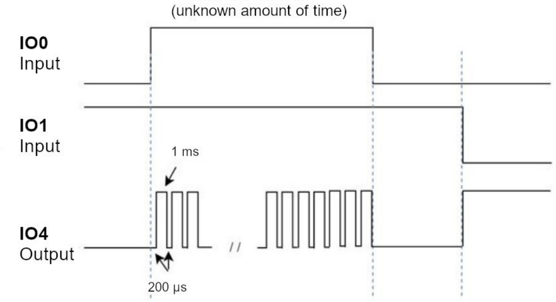


Figure 1: Two cases of output in IO4 port (pulses or "on") depending on the combination of two input ports IO0 and IO1.

THE HARDWARE

Tatu is embedded in an 8-slot CompactRIO controller, under a Real-Time Linux operating system and a programmable FPGA (through LabVIEW). Tatu has been implemented in four different models:

9045 (or 9035) – 1.30 GHz Dual-Core CPU, 2 GB DRAM, Kintex-7 70T FPGA

and the larger

9049 (or 9039) – 1.60 GHz Quad-Core CPU, 4 GB DRAM, Kintex-7 325T FPGA.

The chosen CompactRIO chassis can host C-series modules, a suite of interfaces that provide several analog or digital I/O channels, in a range of different sample rates.

One of two modules can be used under Tatu:

9401 – a 5 V/TTL digital module with 8 bidirectional I/O channels. Four channels are set to input (IO0 to IO3) and four channels to output (IO4 to IO7). In this configuration, the maximum signal switching frequency for the module is 16 MHz per input channel and 10 MHz per output channel [3].

or the larger (but 70 times slower) module

9403 – a 5 V/TTL digital module with 32 bidirectional I/O channels. The first block of four channels is set to input and the next four channels are set to output and so on (IO0 to IO3, IO8 to IO11 and IO16 to IO19 as inputs, IO4 to IO7, IO12 to IO15 and IO20 to IO23 as outputs). Only 12 of 16 input ports and 12 of 16 output ports have been implemented so far, to save FPGA resources. The signal switching frequency is about 150 kHz. [4]

The nominal maximum switching frequency as described in the specification of 9401 and 9403 modules is not reached by Tatu, as its flexibility in the signal management consumes FPGA some clock cycles. Also, our selection of the CRIO chassi model depends on the I/O modules to be used, as they require different demands for FPGA resources. Resource usage and achieved performance in each case is listed in Table 1.

Table 1: CompactRIO FPGA Use and Module Rates

CRIO model	**I/O model**	**FPGA use (Total slices)**	**I/O rate**
9049	9401	24.7 %	10 MHz
	9403	49.2 %	140 kHz
9045	9401	86.8 %	10 MHz

SIGNAL MANAGEMENT

Tatu as the Master Signal Generator

There are some cases where the synchronization of the devices can be provided by Tatu as the signal manager, identifying the status of each involved device and providing trigger signals at the right time, to coordinate their operations as previously set by the user. In this situation, Tatu is said to be in the *Master* mode. Tatu produces a series of internal "master" trigger signals (defined by a given number of pulses, with a duration and periodicity, all previously set by the user). That determines the pace of the operation, being followed by Tatu itself and/or replicated to other devices involved in the data acquisition.

Tatu as a Signal Follower

There are cases where some devices mandatorily require the use of its proprietary software. In such scenarios, the device is the "master" giving the pace of the acquisition operation and Tatu is just another device in the chain. Tatu is set to wait for some external signal and occasionally to replicate it to third parties, other devices in the acquisition. This is the case of Tatu being in the *Follower* mode.

TATU OPERATIONS

When Tatu is deployed, an EPICS IOC (Input Output Controller) is configured and it runs under the Linux RT CompactRIO operating system. It provides several EPICS PVs (process variables) to set and monitor the Tatu operations.

For each input port (IO0...IO3) a trigger state can be detected. The Input Trigger detection modes currently available are:

- an *Up* or *Down* state
- a *Falling edge* or a *Rising edge*
- *under* or *over an Analog threshold (*of a channel in an associated C-series analog module)
- just following the internal *Tatu Master Pulse*
- otherwise, the input is ignored

If one of the *edge* trigger modes is selected, a secondary parameter can be used – *EdgestoTrig*. Then, the triggering condition for input port *n* is satisfied only after the detection of a given number of the specified edge (falling or rising) at that port.

When the input readout meets the trigger mode defined for an input port *n*, the corresponding P*n* state goes true.

Three different cases *c0, c1* and *c2* can be defined for operations on each output port (IO4...IO7). This way, an output port can be turned on in a specific situation and turned off only when another specific condition is present. Each case for an output port has its own *Condition* parameter, whose values are:

- a *Forced* action, as the defined operation should happen regardless of any condition
- the operation should happen as soon as the *Master Pulse* is present
- the operation should happen if any of the P states is true (*P0 ~ P19*)
- the condition is given by a logical combination of *Ps*, so the additional parameter *ConditionCombo* should be inspected to determine the condition; the *ConditionCombo* parameters can be set to one of the logical operations, limited to P0 to P3:
  - *P0 and P1 / P0 and P2 / P0 and P3*
  - *P1 and P2 / P1 and P3*
  - *P2 and P3*
  - *P0 and P1 and P2*
  - *P0 and P1 and P2 and P3*
  - the same combinations with an *or* operator
- otherwise, no output happens

The output will happen after a specific amount of time that can be set (in microseconds) by the *Delay* parameters, for each case and each output port. The type of output or action (also for each case and each output port) is defined by the *Output* parameters, which can be set to one of the following choices:

- set this output port to 0 *(Off)*
- set this output port to 1 *(On)*
- produce a pulse by setting the port to 1, then to 0 for a given time and to 1 again *(Falling pulse)*
- produce a pulse by setting the port to 0, then to 1 for a given time and to 0 again *(Rising pulse)*
- *Copy* a given value to the output port
- *Copy the inversion* of a given value to the output port
- *Trigger* the scalers readouts and keep them

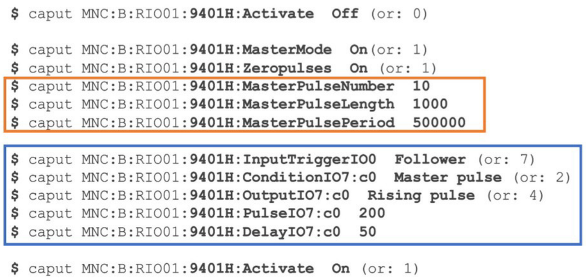
Whenever the *copy* or the *copy the inversion* is selected, yet another parameter is taken in consideration to define the output operation: which input condition state is to be copied. The *OutputCopy* parameter can bear one of the following values:

- copy the *master pulse* value to the output port
- copy the *P0* state or any of the other 19 states to the output port (P0 ~ P19
- otherwise, no output should be produced

Finally, whenever the output port should produce a falling pulse (or a rising pulse), the duration in microseconds of its low (respectively high) state is given by the *Pulse* parameter.

Tatu operates and produces output only when it is put in the *active* state, through an EPICS PV. From then on, any changes in the PVs for Tatu settings are ignored and the input raw data PVs are not updated anymore. All efforts are put in the Tatu execution, and the follow-up must happen through Tatu status Pvs.

The *master pulse* is an internally produced reference. A user example of Tatu parameters set (in CompactRIO1 of hutch B, Manacá beamline) is shown in Figure 2. That would produce a train of four 250-microsecond pulses in the output port IO7 every half second, 10 times.



```
$ caput MNC:B:RIO01:9401H:Activate  Off (or: 0)

$ caput MNC:B:RIO01:9401H:MasterMode  On(or: 1)
$ caput MNC:B:RIO01:9401H:Zeropulses  On (or: 1)
$ caput MNC:B:RIO01:9401H:MasterPulseNumber  10
$ caput MNC:B:RIO01:9401H:MasterPulseLength  1000
$ caput MNC:B:RIO01:9401H:MasterPulsePeriod  500000

$ caput MNC:B:RIO01:9401H:InputTriggerIO0  Follower (or: 7)
$ caput MNC:B:RIO01:9401H:ConditionIO7:c0  Master pulse (or: 2)
$ caput MNC:B:RIO01:9401H:OutputIO7:c0  Rising pulse (or: 4)
$ caput MNC:B:RIO01:9401H:PulseIO7:c0  200
$ caput MNC:B:RIO01:9401H:DelayIO7:c0  50

$ caput MNC:B:RIO01:9401H:Activate  On (or: 1)
```

Figure 2: EPICS PVs used to configure Tatu in the *master mode* (time units in microseconds).

As soon as Tatu is in the *activate* state, that produces pulses in the output port IO7 (Figure 3).

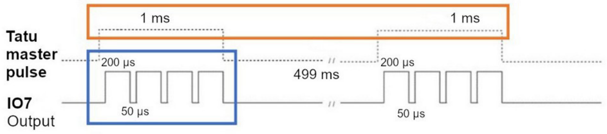


Figure 3: The port IO7 producing output pulses at the *master pulse* pace.

There are watchdog PVs to show that RT-side and FPGA-side software is running. As Tatu gains new features after each new beamline, it may be helpful for the users to check the version control identifier embedded in the FPGA. That can be inspected through an *ai* EPICS PV, a longint number consisting of eight digits *yyyymmdd (*representing *year, month, day)*; for instance, the number **20210131** represents "**January 31st, 2021**".

A small piece of software can be connected to the CRIO to provide a debugging tool, with live values shown (Figure 4).

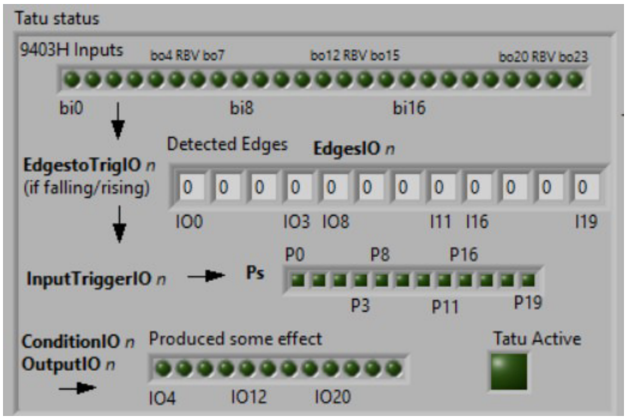


Figure 4: Debug tool for Tatu status.

APPLICATION AT THE BEAMLINES

All beamlines now in commissioning at SIRIUS have one or more Tatu units.

In the commissioning of *Tarumã*, an experimental station at *Carnaúba* (Coherent X-ray Nanoprobe) beamline [5], Tatu has been used for instance in the scanning of areas as small as 80 µm x 80 µm and 800 nm steps. Tatu gives 500 µs pulses along 10100 points, in a total time of 30 seconds for the whole acquisition. [6]

Manacá (Macromolecular Micro and Nanocrystallography) beamline has Tatu in the follower mode, coordinating the shutter opening (1 ms pulse) and the trigger to start the detector in the beginning of an acquisition. This beamline was the first open to user projects [7].

As of Oct. 2021 *Ema* (Extreme condition Methods of Analysis) beamline will start to use Tatu during the energy scanning operations of its high dynamic double-crystal monochromator [8].

CONCLUSION

The premises for such a solution at SIRIUS beamlines were satisfied and Tatu was available in time for the initial tests at SIRIUS *Mogno* beamline (X-ray Micro- and Nanotomography). In sequence, it has been applied for fast data acquisition during the commissioning of the next released SIRIUS beamlines – *Manacá*, *Cateretê*, *Carnaúba*. The flexibility to set parameters and operation modes in Tatu allowed the technical staff to quickly include new input signals and distribute triggering signals to additional devices on demand. An improvement is to admit one or more additional digital I/O modules to increase the number of available ports by keeping the performance observed with just one 100 kHz module. The solution proved successful given the time requirements. Where a better time performance will be needed, the Tatu can be purportedly redesigned, giving up flexibility for the sake of speed. Furthermore, in the future some beamlines will certainly face time requirements too strict to be met by the CompactRIO hardware platform and novel solutions will be selected.

ACKNOWLEDGEMENTS

The authors would like to thank their colleagues among the Brazilian Synchrotron Light Laboratory (LNLS) staff, specially at the groups: SOL-Beamline Software, COI-Beamline Controls and Integration, IEA-Beamline Automation, GIE-Electronics Instrumentation and PLL-Beamline Design for the technical support and advice. During the tests, the contributions of Fernando Henrique Cardoso (GIE), Cassiano Correa Bueno (PLL), Gabriel Schubert (Mogno), Lucca Campoi (Mogno), Andrey Nascimento (Manacá), Tiago Araújo Kalile (Cateretê), João Paulo Zerba (Cateretê), Marcio Paduan Donadio and Gabriel Fedel were much appreciated.

REFERENCES

[1] EPICS-The Experimental Physics and Industrial Control System, `http://www.epics-controls.org`

[2] D. Alnajjar, G. S. Fedel and J. R. Piton, “Project Nheengatu: EPICS support for CompactRIO FPGA and LabVIEW-RT,” in *Proc. 17th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'19)*, N. York, USA, Oct. 2019, pp. 997-1000. `doi:10.18429/JACoW-ICALEPCS2019-WEMPL002`

[3] *NI 9401 Datasheet (p/n 374068A-02),* National Instruments, Austin, TX, USA, Dec. 2015, pp. 4-5. `http://www.ni.com/pdf/manuals/374068a_02.pdf`

[4] *NI 9403 Datasheet (p/n 374069A-02),* National Instruments, Austin, TX, USA, Dec. 2015, pp. 4-5. `http://www.ni.com/pdf/manuals/374069a_02.pdf`

[5] Hélio C. N. Tolentino, Renan R. Geraldes, Gabriel B. Z. L. Moreno, Artur C. Pinto, Cassiano S. N. C. Bueno, Leonardo M. Kofukuda, Anna P. S. Sotero, Antonio C. P. Neto, Francesco R. Lena, Willian H. Wilendorf, Giovanni L. Baraldi, Sergio A. L. Luiz, Carlos S. B. Dias, Carlos A. Pérez, Itamar T. Neckel, Douglas Galante, Veronica C Teixeira, and Dean Hesterberg, "X-ray microscopy developments at Sirius-LNLS: first commissioning experiments at the Carnauba beamline", Proc. SPIE 11839, X-Ray Nanoimaging: Instruments and Methods V, 1183904 (8 September 2021). `doi:10.1117/12.2596496`

[6] C. S. N. C. Bueno, G. N. Kontogiorgos, L. Martins dos Santos, L. C. Guedes, L. G. Capovilla, J. R. Piton, A. C. Piccino Neto, R. R. Geraldes, G. B. Z. L. Moreno, M. A. L. Moraes, “Position Scanning Solutions at the TARUMÃ Station at the CARNAÚBA Beamline at Sirius/LNLS”, presented at the *18th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'21)*, Beijing, China, Oct 2021, paper WEPV002, this conference.

[7] G.D. Noske, A.M. Nakamura, V.O. Gawriljuk, R.S. Fernandes, G.M.A. Lima, H.V.D. Rosa, H.D. Pereira, A.C.M. Zeri, A.F.Z. Nascimento, M.C.L.C. Freire, D. Fearon, A. Douangamath, F. von Delft, G. Oliva, A.S. Godoy, “A Crystallographic Snapshot of SARS-CoV-2 Main Protease Maturation Process”, Journal of Molecular Biology, Volume 433, Issue 18, 2021, 167118, ISSN 0022-2836. `doi:10.1016/j.jmb.2021.167118`

[8] R. R. Geraldes, M. A. L. Moraes, R. M. Caliari, E. P. Coelho, L. P. do Carmo, A. Y. Horita and S. A. L. Luiz, “The FPGA-based Control Architecture, EPICS Interface and Advanced Operational Modes of the High-Dynamic Double-Crystal Monochromator for Sirius/LNLS“, presented at the *18th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'21)*, Beijing, China, Oct 2021, paper TUPV004, this conference.