ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV028

CompactRIO CUSTOM MODULE DESIGN FOR THE BEAMLINE'S CONTROL SYSTEM AT SIRIUS

L. S. Perissinotto[†], F. H. Cardoso, M. M. Donatti
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

Abstract

The CompactRIO (cRIO) platform is the standard hardware choice for data acquisition, controls and synchronization tasks at Sirius beamlines. The cRIO controllers are equipped with a processor running a Real-Time Linux and an embedded FPGA, that could be programmed using Labview. The platform supports industrial I/O modules for a wide variety of signals, sensors, and interfaces. Even with many available commercial modules, complex synchrotron radiation experiments demand customized signal acquisition hardware to achieve proper measurements and control systems integration. This work aims to describe hardware and software aspects of the first custom 8-channel differential digital I/O module (compatible with RS485/RS422) developed for the Sirius beamlines. The module is compliant with cRIO specifications and can perform differential communication with a maximum 20 MHz update rate. The features, architecture and its benchmark tests will be presented. This project is part of an effort to expand the use of the cRIO platform in scientific experiments at Sirius and brings the opportunity to increase the expertise to develop custom hardware solutions to cover future applications.

INTRODUCTION

In a typical synchrotron beamline, a wide variety of electronic devices are employed for controlling and monitoring complex scientific experiments. In a general aspect, these equipment acts to control the beamline components such as shutters, mirrors, monochromators, diagnostic elements, motors, vacuum pumps, etc. These devices have different types of communication interfaces or electrical I/Os in different voltage levels or standards that must be proper integrated to the EPICS control system [1]. The CompactRIO (cRIO) [2] platform is the standard hardware controller used for data acquisition, control and synchronization tasks. The platform supports industrial I/O modules for manipulating a wide variety of signals and interfaces and despite the broad portfolio of commercially available I/O modules, custom hardware designs are often needed. To supply the demand for differential signaling communication, the first in-house C-series module was designed and comprises a 20 MHz max. update rate digital I/O compatible with RS485/RS422 [3] standard.

The hardware and software development were guided by the National Instruments cRIO-9951 [4] development kit, which provides guidelines for design implementation. The developed module will be used for long distance differential communication, triggering and synchronization. The knowledge applied in this development will be useful to develop custom hardware for future applications.

HARDWARE

The module has 8 differential channels with configurable direction, compatible with RS485/RS422, and can achieve up to 20 MHz update rate. An external power supply may be required in some conditions, which depends on the channels' directions and data rate. Differential signals and external power supply are available to the user in the front panel through a 37-pin DSUB connector and it has two led indicators that shows the internal powering status and external powering needs. The module is connected to CompactRIO using a high density 15-pin D-SUB connector on the back of the board.

To safeguard the controller from ESD damage, insulation topology [5] was adopted using digital isolators between all internal and external signals to cRIO bus. The circuit is physically separated into the non-insulated side, where compactRIO is attached, and the insulated side where the application signals are available.

The module circuit is divided into three main blocks: power supply, transceivers, and control logic. The power supply block is responsible for insulating, protecting, and managing all power supply sources, internal and external. The transceivers block converts the internal bus single-ended signals into insulated external differential signals, or vice versa, depending on the configured direction. The control logic block identifies compactRIO control signals and implements all internal mode's support logic. The blocks organization in the PCB can be seen in Fig. 1.

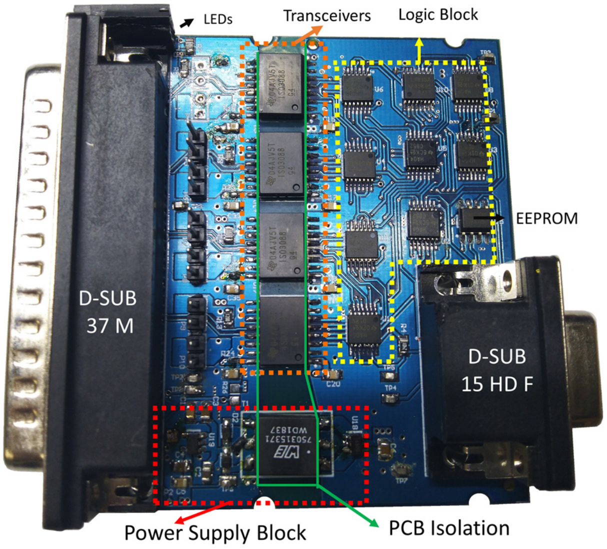

Figure 1: Device blocks on PCB.

[†] lucas.perissinotto@lnls.br

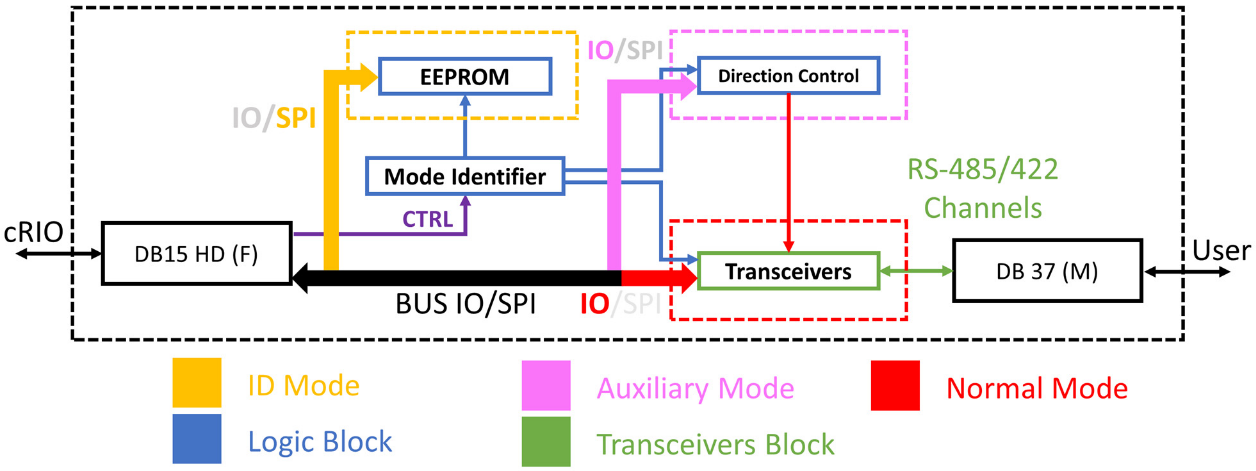

Figure 2: Device block diagram.

The module supports the four internal compactRIO operating modes [6]: Idle, Id, Auxiliary and Normal. In each mode, the bus is used to control a different interface except for the Idle mode, where all module interfaces are disabled. In Id mode, the bus is used to access the embedded EEPROM, allowing the module's identification procedure. The Auxiliary mode allow the user to configure differential channel direction. In Normal mode, the bus is used to access transceivers input or output data channels, depending on the direction set in Auxiliary mode. The operation modes diagram is shown in Fig. 2.

Transceivers

To obtain the required performance and protection, eight insulated half-duplex transceivers IC [7] is used. The transceiver converts single-ended (TTL) signals to differential (RS485) standard with a maximum 20 MHz update rate. Each transceiver's channel is connected directly to the compactRIO bus.

The transceivers have two pins for data signals, one for input (R) and another for output (D). Since the transceivers are half-duplex, both data signals tied together to reduce the number of demanded digital I/O from the controller. This arrangement can be seen in Fig. 6.

In Normal mode, the transceivers have their inputs and outputs set as pre-defined in Auxiliary mode. When other modes are selected, all data pins from the transceivers switch to high impedance state, granting the bus the ability to be used in module identification procedure or channels direction configuration. The differential side are also switched to high impedance to prevent any damage during module initialization.

Regarding the physical aspects of the transceivers block, the differential signals were routed as a pair and designed to have 120 ohms characteristic impedance. The differential traces have termination resistors, that can be enabled by manual jumpers, to provide matched load for tests and specific applications.

Control Logic

The logic block is responsible for identifying each compactRIO internal operation mode and controlling all interfaces that access the bus. The interfaces include the EEPROM, direction control and transceivers. All logic was implemented using logic gates from the 74 IC family. The logic implementation for internal modes identification is shown in Fig. 3.

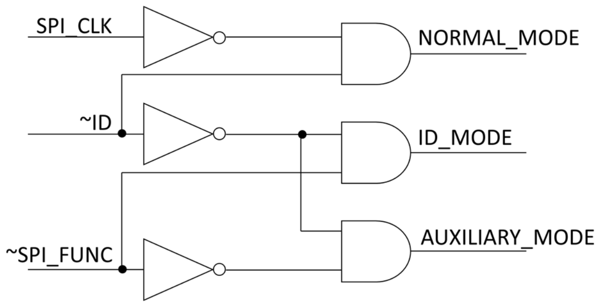

Figure 3: Internal operation mode logic identifier circuit.

The signals ~ID, SPI_FUNC and SPI_CLK comes directly from compactRIO bus in which the first two are intrinsically planned for this purpose. There is no signal intended to determine Normal mode, thus SPI_CLK was selected to be used as a digital I/O since SPI isn't used in this mode.

The EEPROM logic was implemented using the Id mode signal and the chip select SPI signal, as can be seen in Fig. 4. EEPROM SPI signals share the bus with I/O's used in Normal and Auxiliary mode. For this reason, the EEPROM SPI interface is activated only by control signal ~CS during Id mode.

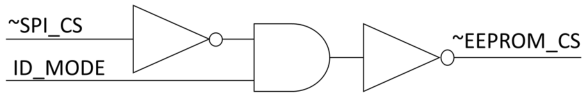

Figure 4: EEPROM ~CS logic control.

An 8-bit shift register [8] with output implemented by flip-flops is used to set and store the data channels logic state directions in Auxiliary mode. Auxiliary mode signal provides access to the shift register for three digital I/O's from the bus. Figure 5 shows the logical scheme implemented to store channels direction that are used in Normal mode.

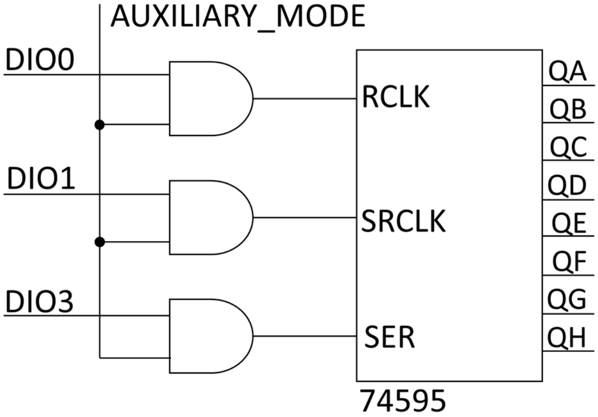

Figure 5: Shift Register and control logic.

The transceiver, on the non-insulated side, has different input and output data pins. Each data pin has its configuration pin, which configures channels direction and high impedance state. Both data pins have been tied together to reduce the need of extra digital I/Os from the bus. The implemented control logic for configuration pins allows it to be used in both directions during normal mode and keeps the outputs at high impedance in other modes. The logic implemented to control each transceiver can be seen in Fig. 6.

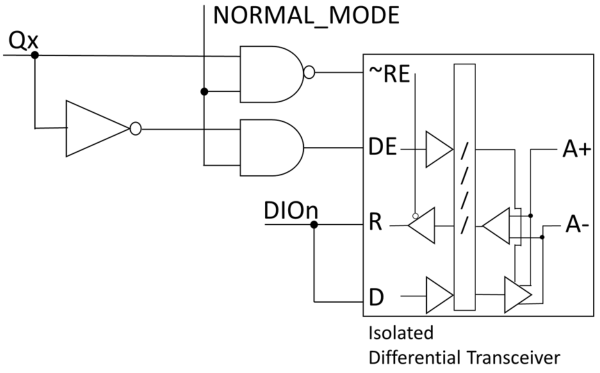

Figure 6: Transceiver control logic.

Power Supply

The power supply block was carefully designed to attend the voltage and current specification and provide protection for the external and internal circuitry. When internal power is used (from cRIO controller), isolators and current limiters are used to guarantee that compactRIO is protected from damaged caused by short-circuit, overcurrent or electrostatic discharge.

The current limiter was placed on the isolated side, thus an overcurrent or short-circuit would not disable the logic block on the non-insulated side. National Instruments documentation restricts the bus current draw to 200 mA.

The current consumption of the non-insulated side was estimated at 45 mA, consequently, the non-insulated side could draw a limit of 155 mA. Considering the transformer conversion factor (1:1.1) and the conversion efficiency (90%), the limiter was configured to limit current draw to 120 mA, in order to limit the input module current in 190 mA. The power supply block diagram can be seen in Fig. 7.

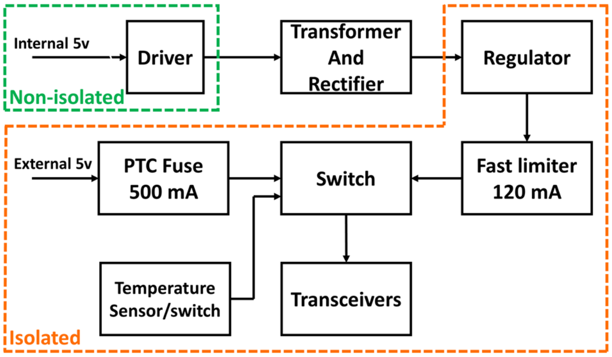

Figure 7: Module's Power Supply.

The internal power supply is adequate for some use conditions, but often an external power supply will be necessary. An electronic power switch is used to manage the use of external/internal power supply. A 500 mA PTC fuse protects the input from short circuits. Two led indicators (LED's) are present in the front panel: a red light indicates the limiter is activated, so external power supply is needed; a green light indicates the module is internally powered.

The external power supply is needed when current consumption exceeds the 200mA bus limit, but the internal module total power dissipation could be monitored to avoid temperature rising. The design rules limit the internal power dissipation in 1.5 W and for all transceivers used at same time (at the highest update rate) the board dissipates nominally 2.4 W. A temperature sensor and switch were used to monitoring the internal temperature in the maximum dissipation power condition and turn off the external power supply in case temperature rise above 60 ºC. The temperature monitor is configurable and can be used in any desired protection logic defined by the user.

SOFTWARE

The module software supports comprises the development of configuration files (XML) and Labview codes (VI). The elaboration of such items involves a series of steps and development modes, and all must be done using Labview environment and is detailed described in the development kit software documentation.

CompactRIO users interact with modules in FPGA environment [9] through Nodes (specific labview block) [10], which types can be I/O's, Methods or Properties. Each module Node usually has codes scripted behind them or can be mapped directly to digital I/O available in compactRIO (D-Sub-15-HD).

The strategy adopted to achieve an update rate close to FPGA clock speed was to map the digital I/O's behind I/O Node. As described in the transceiver's subsection, the controller digital I/O's are connected to the module's internal bus just when the module is in Normal Mode. The differential signal works as an extended digital I/O's mapped in the I/O Node. This arrangement can be seen in Fig. 8.

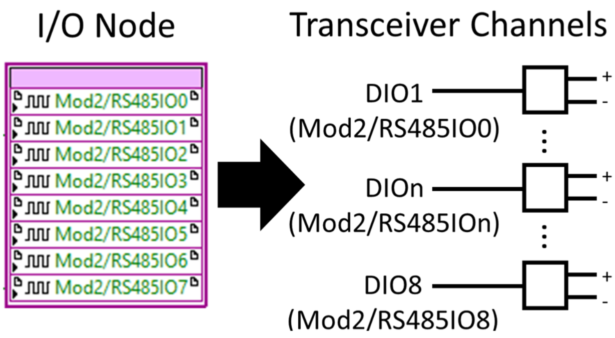

Figure 8: I/O's Node mapped direct in module bus.

In module support block, there is the module core code that works as a "main" function in a textual programming language. The core code executed in this module is responsible for module initialization and channel direction change procedure. This code runs in a loop, waiting for a direction change to be indicated by a Method Node. When a Method Node is used, a byte is passed as an argument, and each bit has, respectively, a differential channel associated, as shown in Figure 9.

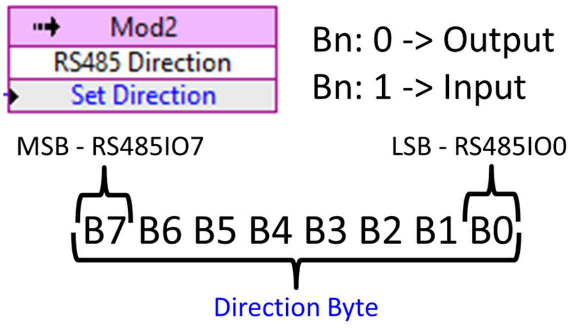

Figure 9: Change direction Method Node.

The core code switches the module to Auxiliary mode at the beginning of each channel direction change procedure. It transfers a byte (bit-by-bit) the to the shift register over DIO3 during the rising edge of DIO1. In the end, the shift register output is updated by a rising edge on DIO0, and the module returns to Normal Mode. The block diagram of direction change is shown at Fig. 10.

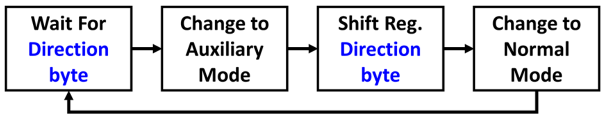

Figure 10: Change direction core code diagram.

TESTS

The module was initially tested using an in-house developed test jig based on LPC1768 [11] microcontroller. All compactRIO modes, functions and I/O's were emulated and tested before board connection at cRIO data bus. With a mature software version ready, performance, consumption and temperature tests were performed.

Performance tests were conducted and the module was able to reach the maximum specified 20 MHz output data rate, as shown in Fig. 11. Communication tests were performed using three strategies: a loopback connection, a commercial NI-9753 [12] module and another developed prototype. In these communication tests, long cables (20 meters) and termination resistors (120 ohms) were used with all simultaneous channels transmitting or receiving data in both directions.

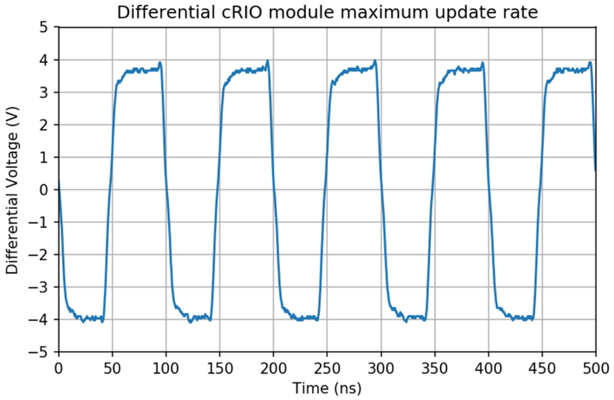

Figure 11: Module maximum update rate output waveform.

The current consumption was measured with the channels configured as input and output. In the input direction test, the module was supplied through the compactRIO connector. Differential signals were applied in the external connector by another module, while the current and limiter indication were verified for different update rates. The results are shown in Table 1.

Table 1: Module Internal Current Consumption

Update Rate Input (MHz)	Internal Current (mA)	Limiter indicator
20	215	1
10	190	0
5	179	0
2.5	173	0
1.25	170	0
0.1	167	0

For the transmitting data (output direction), the module was powered by both sides (internal and external power supply), and the current consumption was measured in the external connector for different data rates. The results are summarized in Table 2.

Table 2: Module External Current Consumption

Update Rate Input (MHz)	External Current (mA)
20	403
10	372
5	342
2.5	311
1.25	280
0.1	169

The maximum measured consumed power, in the transmission test (output direction) was 2W. During this test, the internal current was not monitored but is estimated to be less than half of the values presented in Table 1. In this case, the total dissipated power reaches approximately 2.5W, above the stipulated limit at design rules. The effects of excess power dissipations were verified by keeping three adjacent modules under such conditions, while the temperature was being monitored until stabilization. The observed module temperature was 47.5ºC, working at the room temperature about 23±1ºC and stabilization took two hours.

CONCLUSIONS

The first custom 8-channel differential digital I/O module (compatible with RS485/RS422) in a C-series form factor for the Sirius beamlines was designed. Using high-performance isolated half-duplex differential line transceivers, the module can perform differential communication with a maximum 20 MHz data rate. The module is pin-compatible with the commercial NI-9753 module [12], and it has eight channels independently configurable as input or output. A selectable 120 Ohms resistor is placed to guarantee proper termination in receiver mode.

The designed module is compliant with NI cRIO specification in certain conditions and it has some operational limitations related to the current draw, which can be overcome by the use of external power supplies.

In terms of power dissipation, the module could exceed the defined 1.5 Watts limit in some situations. When the cRIO crate is fully populated, the module temperature must be monitored, and the use of temperature interlock must be evaluated.

Due to the use of low-cost transceivers, the module was able to be manufactured for less than one-tenth the price of the commercial benchmark, which makes it a cost-effective alternative for a wide variety of applications, in special for long-distance digital signal transmission.

FUTURE PLANS

This project is part of an extensive effort to expand the use of the cRIO platform at Sirius beamlines. The developed prototype fulfilled its objective, allowing the developers to obtain the necessary knowledge in hardware and software implementation. The experience acquired by the development team will contribute for the design of more complex modules. Future developments aim to provide solutions for high-resolution digitizers, low latency/low jitter triggering modules, and other scientific experiment-oriented demand.

ACKNOWLEDGEMENTS

The authors would like to gratefully acknowledge the Brazilian Ministry of Science and Technology as funding agency and all the engineering and science teams members working for the Sirius project.

REFERENCES

[1] EPICS, https://epics-controls.org/

[2] National Instruments, CompactRIO Platform Overview, https://www.ni.com/pt-br/shop/compactrio.html

[3] M. Soltero *et al.*, RS-422 and RS-485 Standards Overview and System Configurations, SLLA070 Application Report, https://www.ti.com/lit/an/slla070d/slla070d.pdf

[4] National Instruments, cRIO-9951, https://www.ni.com/pt-br/support/model.crio-9951.html

[5] Texas Instruments, How to Isolate Signal and Power for an RS-485 System, Oct. 2018, https://www.ti.com/lit/an/slla416c/slla416c.pdf

[6] National Instruments, NI cRIO-9951, CompactRIO™ Module Development Kit User Manual, Software User Manual, Nov. 2017, https://www.ni.com/pdf/manuals/375951b.pdf

[7] Texas Instruments, ISO3088 Datasheet, https://www.ti.com/lit/ds/symlink/iso3088.pdf

[8] Texas Instruments, SN74HC595 Datasheet, https://www.ti.com/lit/ds/symlink/sn74hc595.pdf

[9] National Instruments, FPGA Module, https://zone.ni.com/reference/en-XX/help/371599P-01/lvfpgamain/lv_fpga_main_title/

[10] National Instruments, FPGA I/O Node, https://zone.ni.com/reference/en-XX/help/371599P-01/lvfpga/fpga_io_node/

[11] NXP, Arm LPC1768 Board, https://www.nxp.com/products/processors-and-microcontrollers/arm-microcontrollers/general-purpose-mcus/lpc1700-cortex-m3/arm-mbed-lpc1768-board:OM11043

[12] National Instruments, NI-9753 information page, https://www.ni.com/pt-br/support/model.ni-9753.html