The GBT-SCA, a radiation tolerant ASIC for detector control applications in SLHC experiments

A Gabrielli [a,b] for the GBT project
S. Bonacini [a], K. Kloukinas [a], A. Marchioro [a], P. Moreira [a] A. Ranieri, [c] G. De Robertis [c]

[a] CERN EP/MIC Geneva, Switzerland
[b] Università di Bologna and INFN Bologna, Italy
[c] INFN Bari, Italy

alessandro.gabrielli@bo.infn.it

Abstract

This work describes the architecture of the GigaBit Transceiver – Slow Control Adapter (GBT–SCA) ASIC suitable for the control and monitoring applications of the embedded front-end electronics in the future SLHC experiments. The GBT–SCA is part the GBT chipset currently under development for the SLHC detector upgrades. It is designed for radiation tolerance and it will be fabricated in a commercial 130 nm CMOS technology. The paper discusses the GBT-SCA architecture, the data transfer protocol, the ASIC interfaces, and its integration with the GBT optical link.

The GBT–SCA is one the components of the GBT system chipset. It is proposed for the future SLHC experiments and is designed to be configurable matching different front-end system requirements. The GBT-SCA is intended for the slow control and monitoring of the embedded front end electronics and implements a point-to-multi point connection between one GBT optical link ASIC and several front end ASICs. The GBT-SCA connects to a dedicated electrical port on the GBT ASIC that provides 80 Mbps of bidirectional data traffic. If needed, more than one GBT-SCA ASIC can be connected to a GBT ASIC thus increasing the control and monitoring capabilities in the system. The GBT-SCA ASIC features several I/O ports to interface with the embedded front-end ASICs. There are 16 I2C buses, 1 JTAG controller port, 4 8-bit wide parallel-ports, a memory bus controller and an ADC to monitor up to 8 external analog signals. All these ports are accessible from the counting room electronics, via the GBT optical link system. Special design techniques are being employed to protect the operation of the GBT-SCA against radiation induced Single-Event-Upsets to a level that is compatible for the SLHC experiments.

The paper will present the overall architecture of the GBT-SCA ASIC describing in detail the design of the peripheral controllers for the individual I/O ports, the network controller that implements the connectivity with the GBT ASIC and will discuss the operation modes and the flow of information between the control electronics and the embedded front end ASICs.

I. Introduction

The Gigabit Bidirectional Trigger and Slow Control Adapter (GBT-SCA) is a special purpose integrated circuit built in a standard 130 nm CMOS technology. It is used to implement a dedicated control link system for the control and monitoring of the embedded front-end electronics of a High Energy Physics experiment.

To put this GBT-SCA in the context where it will be used, a brief explanation of GBT system is provided in the next section.

A. Overview of the GBT System

Typical High Energy Physics systems are today composed of three subsystems each of which traditionally implements its transmission system from the control room to the electronics located in the detectors. Figure 1 shows this. The subsystems are:

- a fast timing distribution system responsible to deliver to the experiment the system clock and the fast trigger signals and sometimes some fast signal from the detector to the control room;
- a data acquisition bus carrying the collected data out of the detector into the control room;
- a slow control system carrying bidirectional traffic from and to the control room and the embedded electronics in the detectors.

The GBT project aims at providing a common bidirectional system carrying all three types of traffics mentioned above. Clearly this is achieved by sharing a common medium, which in the GBT system is expected to be a pair of unidirectional optical fibers each one with a capacity of about 4.8 Gbit/s. An appropriate bandwidth is allocated to each of the three tasks in the GBT system.

The slow control part is one of the subsystems served by

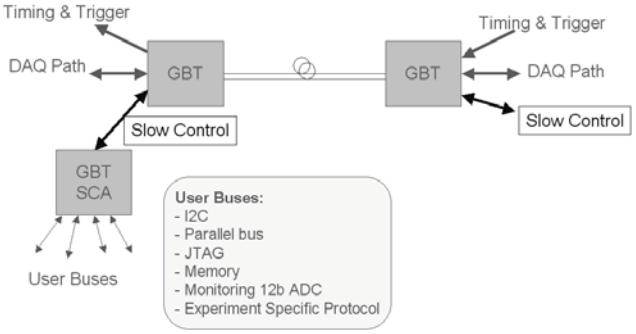

Figure 1: The link

the GBT. The GBT is totally transparent to the slow control protocol. The GBT encoded slow control information in the counting room, carries it along the other traffic on the optical fibers, and delivers the information unmodified to the GBT-SCA in the embedded system. A block diagram of the GBT system is shown in Figure 2. The GBT system consists physically of a dedicated ASIC called GBT13 in the embedded electronics and of an FPGA containing several GBT channels in the counting room. The GBT-SCA is connected physically to the GBT13, which implements the long-haul transmission medium for it.

As the GBT system is based on a point-to-point architecture, the slow control system consists essentially in a local area network using a point-to-point topology. The bandwidth allocated by the GBT system to the slow control function is 80 Mbit/s.

B. Overview of the GBT-SCA Architecture

The communication architecture used by the GBT-SCA is based on two layers. The first layer connects the GBT to the GBT-SCAs; the protocol on this layer is message based and is implemented in a way similar to standard computer LAN networks. The second layer connects the GBT-SCA itself to other chips in the system.

The first layer is unified and common to all GBT-SCAs, and is based on a LAN architecture transporting data packets, to and from the GBT and channel controllers. The second layer is specific to the channel.

The GBT-SCA contains the following blocks as shown in Fig. 2. On the GBT side:

- One MAC Controller;
- One Network Controller (NC). The GBT-SCA control itself is seen as a special channel capable for instance to report the status of the other GBT-SCA channels;
- One SCA Monitor used to control not only the SCA logic itself, but also external front-end alarm signals;
- One arbiter based upon Round-Robin technique to enable the user ports, the monitors or the NC, one at a time, to send data backwards towards the GBT upon reply of previous requests.

On the user side there are 24 I/o ports – one copes with 8 analog inputs:

- 16 I2C master controllers;
- 1 JTAG master controller;
- 1 controller called Detector Control Unit (DCU) that includes an ADC and is used to monitor up to 8 analog signals in the front-end electronic systems;
- 4 I/O like parallel bus controllers such as the ones used in the Motorola PIA etc;
- 1 memory-like bus controller to access devices such

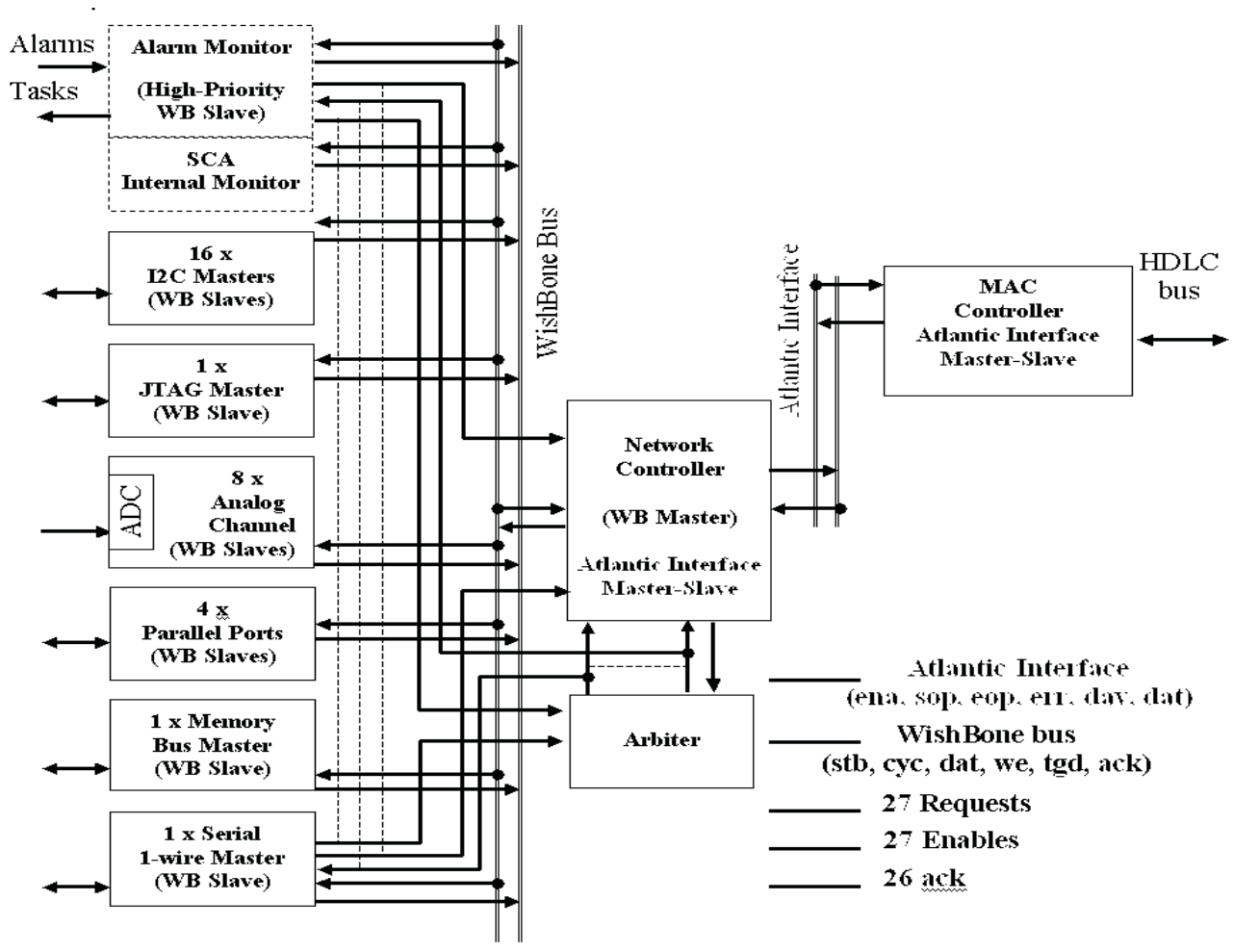

Figure 2: SCA blocks

as static memories, A/D converters etc;

- 1 serial m-wire bus to access simple devices such as temperature sensors and EEPROMS;

All the blocks are synchronous with an external "clock" and have a synchronous "reset". Particularly, each block of the system can be forced into returning to a default state upon execution of a specific reset command. In addition, a hardware reset can reset the whole chip. This latter, is a further asynchronous "reset" added to the system.

II. The Protocol

This architecture assumes that the control is done by sending data packets (messages) to the respective channels, which interpret the messages as commands, execute them on their external interfaces (for example just a read or write operation to a memory bus) and return a status reply to the GBT via another message. The commands can be either addressed to registers located within the channel ports – configuration registers – or to devices located in the far front-end. In this latter case the command interpretation and execution is demanded to the front-end electronics. This protocol assumes that the remote devices controlled by the GBT-SCAs are seen from the GBT as remote independent channels, each one with a particular set of control registers and/or allocated memory locations. The channels operate independently from each other to allow concurrent transactions. The channels can perform transfers to their end-devices concurrently. The high-level network layer, being a local area network-like protocol, is controlled by software running on an appropriate microprocessor through the GBT link. To decouple the operation of the channels with respect to the one of the GBT link, the architecture assumes that all operations on the channels are asynchronous and do not demand an immediate response. Basically this means that all commands carried by the GBT link under the form of network messages are posted to the channel interfaces. This is easy to implement for write operations, where practically one works by posting write operations to the channels. For read operations a read request is sent to the channel; the channel performs the operation on its interface and returns a request of attention to the Arbiter. Then, the Arbiter allows the channels to be activated one at a time through transactions opened by the Network Controller. These transactions send data backwards to the GBT, by including the same transaction identifier that was previously used for the correspondent upwards read command. All upwards packets are acknowledged via either status or data words depending on the command type. Read commands send data backwards, which are auto-acknowledged; write commands send just the status of the channel as a backward reply. Broadcast operations to different GBT-SCA adapters are not supported, as the connection is point-to-point. Only write broadcasts to internal channels are supported. For example, a broadcast operation to several I2C ports proceeds as follows:

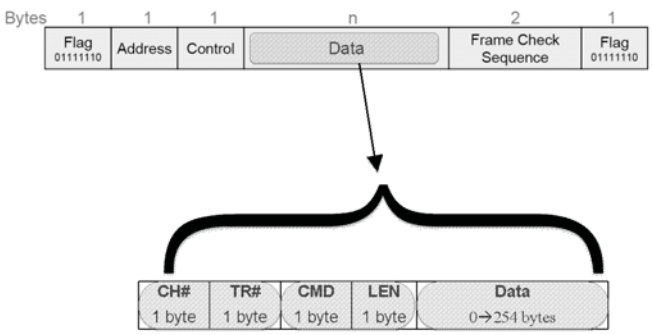

Figure 3: The SCA packet

- a broadcast message is sent to all I2C channels in a given GBT-SCA,
- the I2C channels execute the command concurrently but do not complete it necessarily at the same time,
- if no error occurs, no acknowledgment is sent back.

I2C channels with errors report their status conditions back by sending different error report messages back to the command originator.

MAC controller is addressed via GBT and, if the MAC recognizes a packet whose destination is one of the peripherals of the GBT-SCA, it routes it to the NC. In particular, all the packets from the GBT follow a specific protocol and, through the NC, they are routed to the peripheral (I/O interface) to which they are addressed.

III. The SCA Blocks

The MAC Controller provides two channels, and both according to the Atlantic Interface protocol. These Atlantic Interface channels provide a 1-byte address field and a 2-byte data filed. The address specifies the user-port to be addressed. The Network Controller can be addressed in this way to access its internal configuration registers. The MAC Controller is at the same time a master and a slave device operating according to the Atlantic Interface protocol. It is a master when it sends data to the NC and it is a slave when receives data from the NC.

The Network Controller (NC) routes the data coming from the MAC Controller to the addressed user port. Additionally, when a user port requires sending data backwards to the GBT and when the port, the NC opens a transaction after being allowed by the Arbiter. The transaction correctly closes when acknowledged by the user port. The NC is a master and a slave device with respect to the MAC Controller while it is always a master with respect to the user ports. The NC must require an enable to the Arbiter before opening a transaction through the WishBone bus.

The Arbiter is responsible of enabling the transactions on the WishBone bus. In fact, it allows the blocks, one at a time, to occupy the bus. If the block is the NC, this becomes a transaction required from the GBT while in all other cases the user ports require attention through request signals to the Arbiter. In any case the transactions are master-to-slave from NC to the user ports. In fact, as soon as possible, the Arbiter allows the port to talk and the NC open the transaction. Eventually the data are sent backwards towards to GBT – for example upon reply of previous requests -.

Thus, at any time, each of the user ports can assert its request-signal to indicate that it has data to be delivered backwards to the GBT – for example this occurs as a consequence of a "read" command -. If several ports assert their requests concurrently, the Arbiter provides a quasi-

random priority to the requests: it is based upon the well-know Round-Robin technique. Each request, when served by the Arbiter via an "enable" signal, forces the NC into opening a transaction to the requesting port as soon as possible. Then, the data can flow backwards and the user-port makes the backward bus busy until completion. If the user-port never releases the bus, after a predefined timeout the Arbiter turns off the transaction in any case. The Arbiter is a self-standing device and does not follow the WishBone bus protocol. The "request" and "enable" handshaking signals asserted by any of the user ports are out of the WishBone standards. On the other hand, these signals allow keeping a single-master multi-slave architecture for the WishBone bus. The SCA Monitor is used to monitor not only the SCA logic itself, but also external front-end alarm signals. In fact it is divided into an Internal and an Alarm external part. It continuously monitors the state of the internal machines through counters. It is supposed that, normally, these counters are reset via the NC but, in case of failure, they can reach a given count limit that corresponds to a specific programmable timeout. As a consequence of this, the Monitor can operate one specific task such as an auto-reset of the GBT-SCA or of the channel. A concurrent structure can be applied to monitor external alarm signals and, after a timeout has been reached, further tasks may be activated. In this way the Monitor can make decisions autonomously to handle faulty or abnormal system functioning. This task, for example, will be particularly useful to who wants to switch off an external power supply whenever specific conditions occur. This feature can be seen as if the Network Controller polled continuously external interrupts and, whenever they would require attention, they will be served immediately like high-priority peripherals. This structure allows the implementation of alarm signal through the WishBone bus architecture. In fact, the Monitor is operated and configured through the WB bus - it also contains internal registers to program the timeouts. The network consists of only two devices, the GBT and one embedded GBT-SCA, thus resembling a point-to-point network. Only the GBT is allowed to open a transaction to the GBT-SCA by sending a command via a Data Packet format. If nothing is required to be sent to the GBT-SCA, the GBT to GBT-SCA line is not used except for the clock signal. In fact, this must be sustained in any case to let the GBT-SCA be internally synchronized.

This is a variable-length protocol with a granularity of 1 byte. The MAC might require several cycles to pass an entire packet to the NC. This depends on the LEN field that, as specified below, make the packet length variable from a minimum of 5 bytes to a maximum of 260 bytes.

The WishBone bus standard specifies Single Read, Single Write, Block Read and Block Write commands. Figure 3 shows how each command is identified with a packet – i.e. one command per packet – and contains the following fields:

- 1 mandatory byte for the channel number (CH#),
- 1 mandatory byte for the transaction identifier (TR#),
- 1 mandatory byte for the command type (CMD),
- 1 mandatory byte for the length of the packet (LEN),
- up to 255 optional bytes (DATA) as data field.

In particular:

- CH# specifies the SCA internal port to be addressed – i.e. I2C, JTAG, NC, Monitor, etc. -,
- TR# is a wrap-around byte to identify the packet. This is reported in the backwards reply packet as answer to a previous packet delivered from the GBT to the SCA or to the front-end. This field uses the two dedicated codes 0x00 and 0xFF for internal and external alarm packets,
- CMD is a command code that specifies a given transaction. The operation can refer to a specific internal register of the channel – i.e. a configuration register – or a front-end destination address. In this case an address field follows the command.
- LEN is a field that ranges from 0 to 255 that specifies the DATA field length. For read commands LEN is 0,
- DATA is an optional variable length field upon LEN value.

IV. Conclusion

The GBT and SCA project is aimed at proposing a high-speed general-purpose optical link for the data acquisition chains of the front-end electronics for SLHC experiments and beyond [4]. For this reason many standard user-ports have been proposed along with a Link protocol.

The project is justified because embedded applications in modern large high-energy physics experiments require particular care to assure the lowest possible power consumption and the radiation tolerance, still offering the highest reliability demanded by very large particle detectors.

Within the project, the SCA chip will carry out the slow-control operations for the front-end electronics. In addition, as the SCA will be located in a radiation environment, it will include a robust design to stand SEE.

SCA will interface with front-end electronics via common ports such as JTAG, I2C, parallel and 1-wire and, with the GBT via a Link port.

V. References

[1] P. Moreira, T. Toifl, A. Kluge, G. Cervelli, F. Faccio, A. Marchioro, J. Christiansen., "G-link and gigabit Link compliant serializer for LHC data transmission", Nucl. Sci. Symp. Conf. Record, 2, (2000), pp. 96-99, doi: 10.1109/NSSMIC.2000.949860

[2] M. Rahman, "Super-radiation hard particle tracking at the CERN SLHC", IEEE Trans. Nucl. Sci., 50/6, (2003), pp. 1797-1804, doi:10.1109/TNS.2003.820769

[3] H. F.-W. Sadrozinski, A. Seiden, "Tracking detectors for the sLHC, the LHC upgrade", Nucl. Instr. Meth. A, 541, (2005), pp. 434-440,doi:10.1016/j.nima.2005.01.086

[4] A. Gabrielli, F. Loddo, A. Ranieri, G. De Robertis, "Architecture of a general purpose embedded Slow-Control-Adapter ASIC for future high-energy physics experiments"Nucl. Instr. Meth. A, 596, (2008), pp. 113-116, doi:10.1016/j.nima.2008.07.060