Qualification of the Front End Cards for the CMS ECAL.

E. Corrin, M Gastal, M. Hansen, and P. Petit

CERN, CH-1211 Geneva 23, Switzerland

Abstract

The Front End (FE) card is a critical component of the new readout architecture for the CMS ECAL. The module hosts the Front End New Intermediate data eXtractor (FENIX) ASIC that contains the DSP for trigger primitive generation and RAM for the digital pipeline and the primary event buffers.

The Front end Automatic Tester (FAT) allows fast fault detection to ensure that the manufacturing process of the FE cards remains in control. In less than two minutes, the response of the device under test is then recorded and analysed to detect any unexpected behaviour.

I. Introduction

The high-resolution crystal calorimeter ECAL[1] is a crucial sub-detector within the CMS experiment for the Higgs quest. Lead tungstate crystals which have high density, a small Moliere radius, and a short radiation length have been chosen allowing for a very compact calorimeter system. Figure 1 shows the position of ECAL in the overall architecture of CMS.

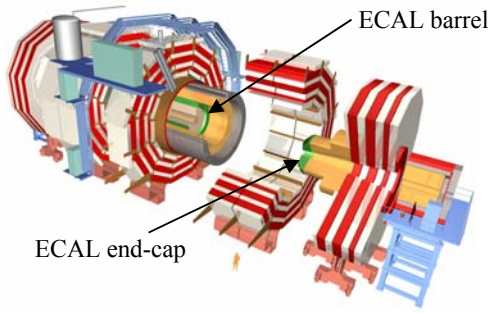

Figure 1: The ECAL detector within CMS

Even though the architecture of the CMS ECAL trigger and data acquisition system has changed in 2002, the functional requirements remained unchanged[1]. The FE card is just one of the new set of modules recently developed to fit the new needs of the CMS experiment. It serves one trigger tower in the ECAL barrel or one super-crystal in the end cap. Each trigger tower consists in five Very Front End (VFE) cards, one FE card, one Low-voltage Regulation (LVR) card, and one mother board. Each VFE card communicates with the FE card through five 16-bit channels. Figure 2 shows how the FE card is included into a trigger tower.

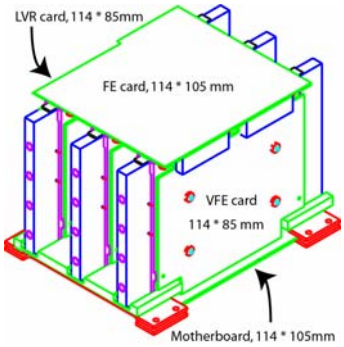

Figure 2: FE card in a Trigger Tower

The FE card was designed to carry the currently produced FENIX ASIC whose function is to perform complex Digital Signal Treatment (DSP) tasks on the output data of the VFE cards. The output data from the FE card is sent to Data Concentrator Card (DCC) and Trigger Concentrator Cards (TCC) through dedicated optical links. The optoelectronic component Gigabit Optical Link[2] (GOL) is in charge of serialising the data and driving the laser transmitting it through an optical fibre. It is carried by a mezzanine card called Gigabit Optical Hybrid (GOH) plugged into connectors on the FE card. Two models of the FE card will eventually be available. For the barrel, the FE card has two GOH (1 TCC and 1 DCC: trigger tower) while in its end-cap version it has six of them (5 TCCs and 1 DCC: trigger strip). Each optical link consists in 16 bits of data plus 1 bit of handshake. Figure 3 shows the FE card in its barrel version connected to two GOH cards.

Figure 3: FE card with two GOH cards

The production of up to 4,000 FE cards will start in the second half of year 2004. To ensure that the manufacturing process remains in control, a fast qualification system was built. The Front end Automatic Tester (FAT) was designed to fully qualify a FE card within two minutes and with a minimum amount of handling. The characterisation procedure verifies all the functionality of the FE card by downloading data patterns into the FAT and using them as stimuli for the card under test. The slow control protocol between the FAT and the FE card is a perfect replica of the one implemented in the experiment (Token Ring). The response of the card to the selected stimulus is then recorded and analysed to detect any unexpected behaviour. Figure 4 presents a block diagram of the system.

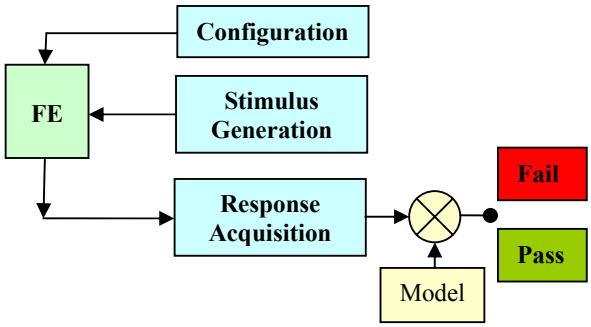

Figure 4: FAT functional block diagram

Because the FAT reproduces the real life environment of the FE card when operating in the CMS experiment, it can also be used to accurately simulate a particular event and monitor the corresponding FE card response. Potentially any event can be reproduced through a data flow and a trigger generation.

II. SYSTEM ARCHITECTURE

A. Hardware

The FAT system consists in two Printed Circuit Boards (PCBs) called respectively FAT (Front end Automatic Tester) and FATI (Front end Automatic Tester Interface) cards.

The FAT card is a custom 12-layer PCB, designed and produced at CERN. It holds all the Integrated Circuits (ICs) of the system as well as connectors for its interface. Because of its size (300mm × 550mm) the production price of this card is significant and therefore only a few units should be manufactured. The amount of costly components soldered onto the board also contributes to the willingness to minimise the production. When the time comes to start the production test only 3 FAT card should be used (1 at PSI, 1 at PH-CME, and 1 at building 867 Integration centre) and an additional unit should be kept as backup. Figure 5 shows the FAT card plugged in a VME rack.

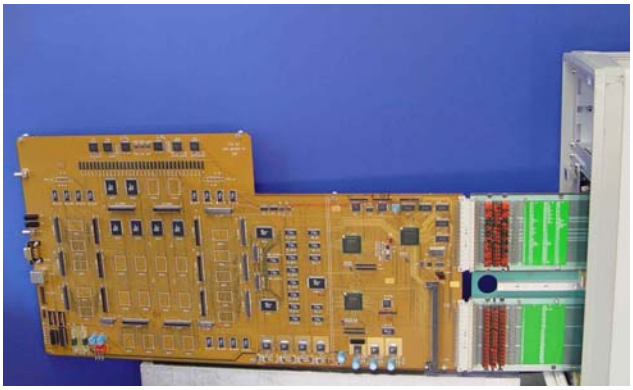

Figure 5: VME configuration for FAT

The FATI card is designed to be plugged into the FAT card and to serve as holder for the FE card under test. Its smaller size (300mm x 250mm) and limited amount of layers (6) make it much cheaper to produce. The reason for designing an interface card arose when considering the risks of fast aging of the FAT card under intensive use. Plugging and un-plugging FE cards can lead to damage the interface connectors and compromise the functionality of the FAT system. Because of its reduced costs the FATI appeared like a convenient alternative. The FE cards to be tested will be plugged into the FATI, which if need be, will be frequently replaced. In total 15 units of this card are planned to be produced. Additionally, boundary scan buffers have been implemented on the card to a complete boundary scan check of the FE card. Figure 6 shows the FATI card and points out some its main connectors.

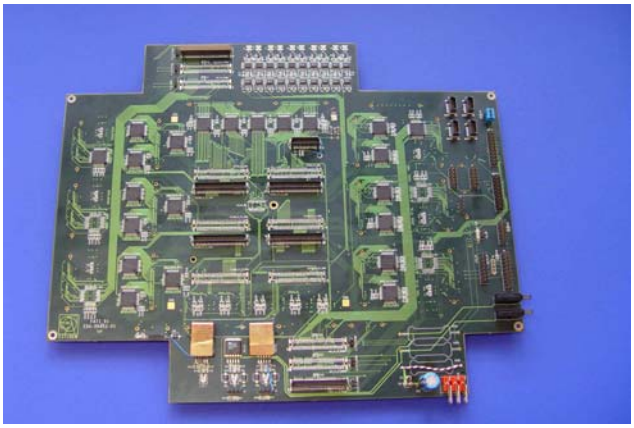

Figure 6: The FATI card

B. VME or TINI controlled system

To interface the FAT system with its environment two equally performing solutions have been designed and implemented. The first one makes use of a VME interface and implies to plug the FAT card into an appropriate rack. Alternatively, the FAT can be operated through a Tiny InterNet Interface (TINI, Dallas Semiconductor) so avoiding the space consuming VME environment.

The VME interface allows operating the FAT system in a well known and extensively used environment. Hardware

such as rack and rack controller are conveniently available at CERN. The chosen operation mode uses a 32-bit address and 16-bit data protocol. As will be presented later a software interface has been design to this extent. The main drawback of this interfacing solution is the large space requirement to operate the system.

It can also be noticed that in this configuration the FAT card has to be held vertically which does not make the handling of the cards under test easy (see Figure 5). This is why this configuration is mainly used for debugging purposes and is not considered for the production test phase.

The TINI interface provides a cost efficient and practical alternative to the VME interface. It can be connected to a PC using a simple Ethernet connection and allows the FAT card to be positioned horizontally. Figure 7 shows the FAT system in this configuration.

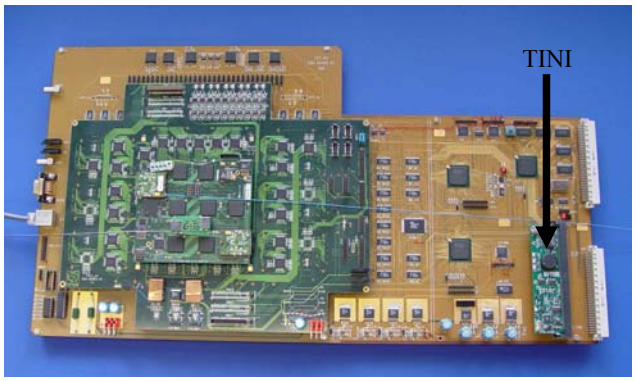

Figure 7: TINI configuration for FAT

Thanks to the Ethernet connection the FAT system could theoretically even be operated from a remote location. This might come in handy should the production tests be outsourced to an outside company. In this case the results of the tests could be monitored and parameters updated from CERN. This configuration is therefore the preferred one for production tests.

In order to introduce the functioning of the FAT system the key components present on board will now be described.

C. Memories and programmable logic

The data handling is managed by three Xilinx Field Programmable Gate Array (FPGA) devices of type xc2s200-6fg456. Each of them has a dedicated role to play in the system operation.

The first FPGA (FPGA1) is in charge of bidirectional data flow between the FAT system and the VME/TINI interface. It also manages the FAT local 16-bit address and data bus. Additionally, behaving as a master, it handles the communication with the two other FPGAs on board. One of its key functionality is to read from and write to the memories on board.

The second FPGA (FPGA2) receives it commands from FPGA1 and essentially drives the stimulus-release process to which the FE card under test is exposed as well as the recording of its response.

The last FPGA (FPGA3) also receives its commands from FPGA1 and implements the slow control of the FE card through a Token Ring type of communication.

Three groups of Dual Port Random Access Memories (DPRAM) are also implemented on the FAT card. One can read and write to these devices through both of their ports. For example port A is used by the FPGA1 to read from and write to the DPRAMs on board the FAT system.

The Data DPRAMs (25 components of 16 bit × 64k) are used to simulate the outputs from the VFE cards in a real life environment. Readout at a 40MHz rate, they will reproduce the data flow to which the FE cards are exposed during ECAL operations. Each of the 25 VFE channels is therefore replaced by a DPRAM.

The Real Time DPRAM provides a simple way to generate synchronous commands to the FE card. To do so clock edges will be removed from the continuous clock link between FPGA3 and the FE card (missing pulse) upon reception of a logical '1' at the output of the DPRAM. Several synchronous commands can be sent depending of the pattern loaded in the RT DPRAM. For example the sequence "100" will be interpreted as a level 1 trigger by the FE card, while "110" will be understood as a resynchronisation request.

Seven additional DPRAMs (REC DPRAMs) are used to record the data from the FE card. They receive the signals normally sent through the optical links. Since the end-cap version of the FE card has six 17-bit channels to be sent through the optical links seven 16-bit DPRAMs will substitute them.

D. Data Flow

In order to better visualise the path followed by the data during the test transactions each case will be looked at separately.

1. Loading of the DPRAM

The data to be loaded in the DPRAMs is sent through the VME/TINI interface on to FPGA1. The values are then presented on the local bus and loaded into each specific memory location. In a similar way a readout process is implemented following the same data path. Figure 8 shows the dataflow.

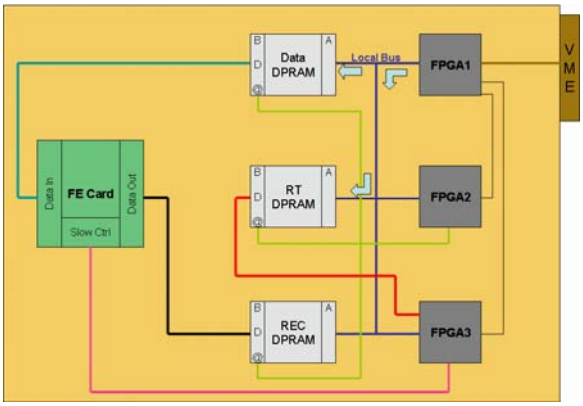

Figure 8: Data flow – Writing to, Reading from DPRAMs

2. *Configuration of the FE card*

To set-up parameters inside the FE card a Token Ring type of communication is used. The command is received by FPGA1 and then forwarded to FPGA3. From then on communication between the FPGA3 and the FE card occurs exactly as in the real life system. The only difference is that electrical signals are used instead of optical ones. This allows saving the cost of expensive opto-electronic transceivers. Figure 9 shows the dataflow.

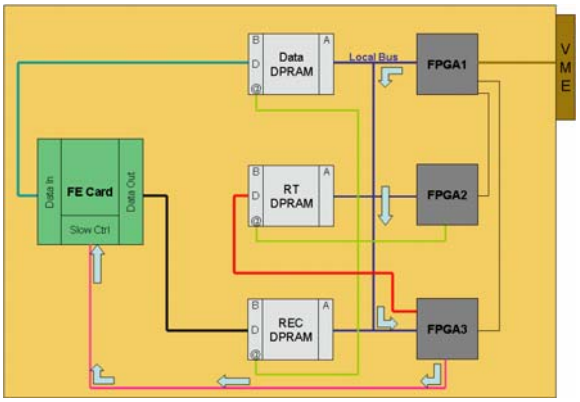

Figure 9: DATA flow – Configuration of the FE card

3. *Stimulus sending*

When both the FE card is configured and the DPRAMs are loaded, the stimulus can be sent and the response of the card under test recorded. The stimulus release is triggered by a command sent to FPGA1 through the VME bus. In a first step FPGA2 broadcasts addresses simultaneously to all the DPRAMs. This triggers the sending through port B of the data loaded in the DATA DPRAMs to the FE card and the recording by the REC DPRAM of the FE output. The data from the RT DPRAM reaches FPGA3 in the process. The FE card then sees a flow of incoming data similar to what would be experienced in a real life environment with the VFE cards. When a "100" pattern is broadcasted by the RT DPRAM and received by FPGA3, a level 1 trigger (or its missing edge equivalent) is sent to the FE card using the Token Ring. Figure 10 shows the corresponding dataflow.

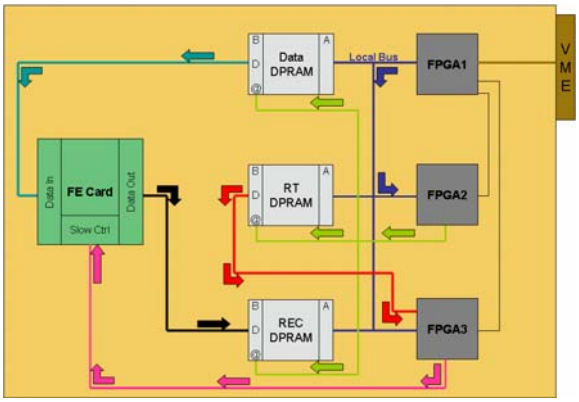

Figure 10: Data flow – Stimulus sending

At the end of the sequence the data collected by the REC DPRAMs is readout by FPGA1.

E. *Digital design*

In this section the functionalities of each of the three FPGAs will be detailed. The design tools that were used are Cadence for the simulation, Leonardo for the synthesis, and Xilinx ISE for the Place and Route operation.

1. *Board management (FPGA 1)*

This device is the master of every communication occurring on the FAT board. This status results from the data centralization occurring on the board. To have a command executed, to read from or write to the board, one needs to go through FPGA1.

FAT Local Bus:

The 16-bit data sent through the VME data bus is passed on to the local 16-bit data local bus.

The least significant 16 bits (Bits<16..1>) of the 31 bits of the VME address bus (Bits <32..1>) are copied onto the local address bus of the FAT board. When reading from or writing to a DPRAM these bits specify a memory location to access. When passing a command to FPGA2 or 3 they are used in combination with handshake signals to specify a register location.

DPRAMs access:

Used to identify the DPRAM targeted by the communication these bits (Bits<21..17>) are decoded and select the corresponding DPRAM.

Figure 11 shows how the DPRAMs are identified.

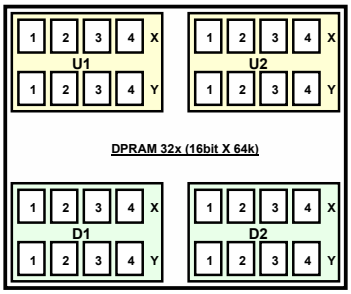

Figure 11: DPRAMs identification

Among these 32 devices 25 are data DPRAMs and 7 are REC DPRAMs. From address "00000" to "11000" the data DPRAMs are targeted while from address "11001" to "11111" access is given to the REC DPRAMs. In this later case D2-X-4 is used to record the handshake signals from the optical links and the remaining devices record the 16-bit data sent out. As presented below the RT DPRAM is treated independently.

When loading the RT DPRAM one particular address should be used and corresponds to Bits<22..17> = 100000.

Local Display Register:

Designed to easily notify the user of the system, this 8-bit LED array can be driven by the VME bus when using the following address: Bits <22..17> = 100001. In this case the least significant 8 bits on the VME data bus will be loaded into the register. This feature can be particularly useful to notify an operator that the test is complete and successful.

Communication with FPGA2:

Since any communication goes through FPGA1, one particular address had to be selected to indicate that the destination of the data is another FPGA. To access FPGA2 the selected address is: Bits<22..17> = 100010. Additional handshake signals are also required as described in the FPGA2 section.

Communication with FPGA3:

Similarly, an address has been chosen to target FPGA3, Bits <22..17> = 100011.

Because the communication on the board makes use of a common local bus, many buffers, some of them bi-directional, have been implemented. They allow setting the communication direction and isolating lines when required. The control ports of these components (direction and enable) are exclusively handled by FPGA1.

2. *Pattern Generation (FPGA 2)*

The core function of FPGA2 is to manage the FE card stimulus sending process. The user has the possibility through FPGA1 to set a start address that will define from where in the DPRAMs the data should be broadcasted. The second parameter is the length of the broadcasting sequence. These two parameters make it possible for the user to select a restricted amount of data from the whole range loaded in the DATA DPRAMs.

Two extra commands have also been encoded to start and stop the stimulus sending process.

3. *FE Configuration (FPGA3)*

This device is dedicated to the communication with the FE card through the slow control link implemented as a Token Ring with a single node. In order to reproduce as precisely as possible the real life environment of the FE card, existing digital blocks have been reused as they are implemented in the Clock and Control System (CCS). Figure 12 shows the block diagram of the system.

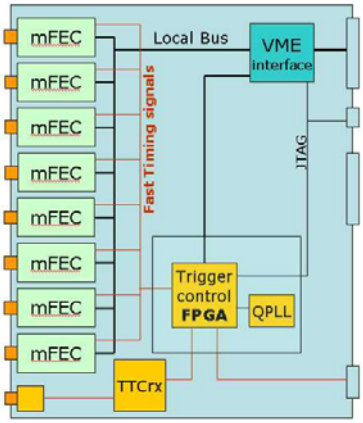

Figure 12: Schematic view of the CCS board

By instantiating the Verilog blocks from the mFEC (mezzanine Front End Control) and the VME interface FPGA, a top level code was built. A third block (ADD) is designed to interface the local bus of the FAT with these two former blocks. Figure 13 shows how the core of FPGA3 is assembled.

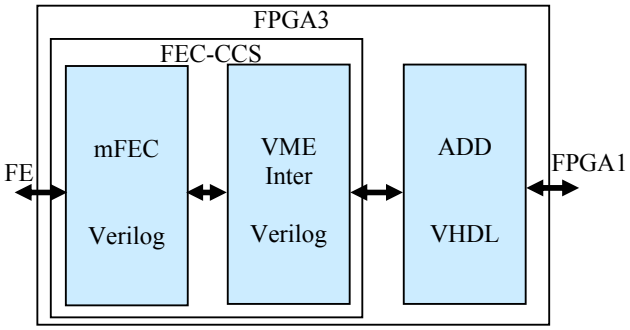

Figure 13: FPGA3 core

Using the handshake signals between FPGA1 and FPGA3 as well as the FAT local bus, the user can send commands to the FE card through the exact same protocol as the one implemented in the CSS system.

An additional functionality has been implemented to remove a clock edge when receiving a logical '1' from the RT DPRAM.

F. Software

Both flexible and user friendly, the user interface offers a practical solution to communicate with the on-board logic implemented on the FAT system. A typical test commences by loading the data DPRAM and RT DPRAM with a predetermined set of stimulus test-vectors, and initiating the test sequence. Once the run is complete, the data from the REC DPRAM (containing the response from the FE card) is read back and compared to the expected model. If any differences are encountered, they are summarised and reported to the user.

The initial version of the software is written in C++ and runs under Linux. It uses the HAL library to communicate with the FAT via the VME interface. The next generation of the software will not use the VME interface, but will instead communicate with the TINI module via the Ethernet interface. This will allow the test to be run either locally or remotely, from any internet-connected PC, using just a web browser.

III. FE CARDS TESTS

A. Objectives

The FAT system was design to allow a simple and quick qualification of the FE cards. The main idea is to check that the crucial connections exist and that the FE card operates successfully at 40 MHz.

Limiting the tests to a reasonable amount of checks is necessary considering the complexity of an FE card. A few objectives were defined to guarantee the compliance of the cards under test with the requirements set by the CMS collaboration. One test cycle should include:

- 1 boundary scan check
- 1 I^2C test (check presence of FENIX)
- 1 set of data loading in the DATA DPRAMs and 100 triggers in the RT DPRAM.
- 1 set-up of the FE card using the slow control
- 1 stimulus sending and data recording

The time budget for one full test sequence should not exceed two minutes.

B. TEST PATTERNS AND SIMULATION

The six FENIX chips present on every FE card are performing complex DSP on the data coming from the VFEs. To make the tests relevant, the stimulus should be close to the data generated by the VFE cards in reality. To do so it was decided to reuse the simulation test vectors generated when designing the FENIX chip. This stimulus has been evaluated close enough to the physics to validate the FENIX design and should therefore be just as appropriate to qualify the FE cards.

One of the great assets of the FAT system is its ability to simulate the real life environment of the FE card. Should the response to a particular event be studied, the corresponding stimulus could be loaded in the DATA DPRAMs and sent on to the FE card. This functionality extends the use of the FAT system beyond a test system on to a simulation tool. This may be useful even after the ECAL installation.

IV. CONCLUSION

Because of the flexibility implemented in its design the FAT system appears to be both a versatile characterization and a simulation tool. Its prototype version has been tested with the first version of the FE cards (barrel). Operations with the end-cap version of the FE cards have yet to come but should occur sometimes before the end of the year. As soon as the system will have been fully tested a new version of the FAT board will be manufactured to replace the existing prototype. Plans are to have a fully mature system by the time the production of the FE cards starts. In addition, it is intended to make use of the FAT system during the irradiation tests on the FE cards that will take place later this year.

V. ACKNOWLEDGEMENTS

I would like to thank Christer Ljuslin from CERN-PH/MIC for his help with the Verilog blocks he designed and that were re-used in the FAT system. His in-depth knowledge of the designs allowed to speed up the instantiation process and greatly contributed to the success of the project.

VI. REFERENCES

[1] “The Electromagnetic Calorimeter Project”, Technical Design Report, CERN/LHC 97-33, CMS TDR 4, Dec. 1997

[2] “The New Readout Architecture for the CMS ECAL”, M. Hansen, Proceedings of the 9th Workshop on Electronics for the LHC Experiments, Amsterdam, Sept., 2003.

[3] “A Radiation Tolerant Gigabit Serializer for LHC Data Transmission”, P. Moreira, G. Cervelli, J. Christiansen, F. Faccio, A. Kluge, A. Marchioro and T. Toifl, Proceedings of LEB 2001, Stockholm, Sept., 2001.

[4] “The CMS ECAL Readout Architecture and the Clock and Control System, R. Benetta, M. Gastal, M. Hansen, K. Kloukinas, C. Ljuslin, A. Marchioro, J. Nash, P. Sharp, proceedings of CALOR 2004, Perugia March-April, 2004.