CHAPTER 4

Silicon Drift Electronics For ALICE ITS

4.1. Introduction

The work on the A Large Ion Collider Experiment (ALICE) Inner Tracking System (ITS) drift detector development has resulted in good progress in design and test of first prototypes. In the INFN Drift SIlicon (DSI) project several approaches in design and technology were thoroughly investigated [23, 24, 25, 26, 27, 28, 29]. Particularly it has been investigated the front-end electronic approach for data acquisition, packaging and transmission to the storage devices. This front-end electronics has been divided into two main stages: the first is also divided into an analog drift electronics part for data acquisition from detectors, for temporary analog storage and for analog-to-digital conversion and in another digital electronic part for temporary storage and control before data packing and transmission. Then there is a second-stage electronics for collecting data from several data-channels, compression, packaging and transmission upon a give data protocol. It has also been decided to design a first low-speed off-line prototype printed board for testing purposes. The board particularly focuses on the second-stage of the front-end electronics. On the printed board have been mounted two XILINX FPGAs 4025E-4 that consist of a 32x32 Configurable Logic Blocks (CLBs) matrix. This matrix can be imagined as a flexible device with up to 32768 ram bits, or as a up to 2432 logic cells device, or as a component with up to 25000 logic gates, or as a matrix with up to 2560 flip-flops. Moreover this FPGA has up to 256 I/O pads. Obviously any given design generally needs just a subset of some of these features and the rest is left unused.

The two components have been designed one for a first data compression based on the Huffman, threshold and differential encoding and one for packing the data from different channels into a well-defined transmission protocol. It is to be noted that the Huffman code, that is a variable length encoding, provides a reasonable compression if the data statistics is known in advance and if the data are mainly concentrated in a given range of the input domain.

As it is going to be explained in the chapter the printed board receives digital 8x8-bit data from 8 input channels at a rate of 10 MHz. The rate will rise to 40 MHz according to LHC. These 8x8-bit data are firstly analyzed and if recognized as under-threshold values are rejected. On the other hand the selected ones are coded according to the Huffman code. It can be said that the data are packed to a set of 32-bit for each of the 8 input channels. Then, every 16 sets of these 32-bit words, the collected data are joined to a header word that gives specifications on the event and on the channel numbers. Finally these data are sent to the Source Interface Unit (SIU) and to the Data AcQuisition system (DAQ).

4.2. Drift Electronics

The system requirements for the Silicon Drift Detector (SDD) derive from both the characteristics of the detector and the ALICE experiment in general. The power budget is very low, in fact as low as possible. This requirement is due, on one side, to

the high sensitivity of the detector to temperature variations and, on the other side, to the limited material that can be located on the ladder structure, which limits the capabilities of the cooling system and the section of the power lines. The structure is composed of set of silicon detectors (SDD) put together side by side. Several linear structures made of SDDs are then put together in a cylindrical shape: the ladder. The signal generated by the SDD is a gaussian-shaped current signal, with variable sigma and charge (5–30 ns and 4–32 fC), and can be collected by one or more anodes. Therefore the front-end electronics should be able to handle analog signals in a wide dynamic range. Then, the system should be very low noise while being able to handle large signals. The amount of data generated by the drift detector is very large: each half-detector has 256 anodes, and for each anode 256 time samples have to be taken in order to cover the full drift length (see figures 4.1. and 4.2.).

The small space available on the ladder and the constraints on material impose an architecture that minimizes cabling. In order to meet these requirements a two-buffer front-end system has been designed. The first buffer is a low-power and fast analog memory while the second buffer is a static Random Access Memory (RAM). The data from the detector are continuously stored into the first buffer and are transferred to the second buffer (after an ADC) only when the trigger system validates them. In this way the ADC is activated only when necessary. The second buffer can store more than one event. In this way it is possible to reduce the transmission rate to the DAQ system from the peak value to the average value with an acceptable dead time. Before transmission to the DAQ system a data reduction has to be performed in order to match the constraints on the transmission media (optical fiber) bandwidth and, more stringent, on the event size on the tape (1.5Mb/event in Pb–Pb). Since a simple zero suppression does not allow a good reconstruction, more complex algorithms have been studied to perform a better data compression. Those algorithms are based on multi-threshold cluster analysis, differential schemes and Huffman encoding (see Sec. 4.4.).

The front-end system is organized in readout units, and holds four two-chip groups, distributed on the ladder, near the detectors. Each front-end readout unit reads 256 detector anodes (a half-detector). The first ASIC contains the preamplifier, the analog buffer and the ADC, while the second chip contains the digital event buffers. The data compression and transmission are performed in the two FPGA units, located at both ends of the ladder structure. Each unit is based on eight macro-channels. Each macro-channel is connected to a single front-end unit, and thus manages the data of a half-detector. The data from the eight macro-channels are then multiplexed before transmission to the DAQ system. In this way a high degree of modularity has been achieved. The total radiation dose foreseen in the experiment is rather low, but Single Event Effects (SEE) have to be taken into account. Thus we will adopt radiation-tolerant techniques, based on deep sub-micron technologies, for the ASIC design.

4.3. Requirements and Trigger

The SDDs are expected to provide high detection efficiency over the whole detector surface with a spatial precision of the order of 30 μm and a two-track separation down to O(600) μm.

Three trigger levels are foreseen in ALICE, two with fixed latency and one with variable latency. Table 4.1 shows the calculated latency of the levels. Since *L*1 has fixed latency, absence of this signal means event rejected or *L*1*reject*, while the *L*2*reject*

signal will be sent explicitly by the trigger system. A *BUSY* signal for the detector is asserted on *L*0 by the trigger system for about 1–1.3 μs.

Level	**Latency**
L0	fixed 1.2 μs
L1	fixed 2.7 μs
L2	variable: 4 - 100 μs

Table 4.1. Calculated trigger latencies for each trigger level

With this triggering scheme, the SDD readout system should be able to perform the following.

- Start the acquisition after an *L*1 signal. The acquisition has to be delayed with respect to *L*1 in order to take into account the total detector drift time.
- Abort the acquisition if an *L*2*reject* is received and reset as soon as possible. In a standard Pb–Pb operating mode 95% of the events selected by *L*1 will be rejected.
- Reply to the trigger system within 1–1.3 μs after the earliest trigger level.

A new trigger scheme is currently being evaluated. Table 4.2. summarizes the proposed new timing.

Level	**Type**	**Latency**
L0	*L*0*e*	1.2 μs
	*L*0*d*	2.7 μs
L1	*L*1	5.5 μs
L2	*L*2*y*	100 μs
	*L*2*n*	$\leq$ 100 μs

Table 4.2 Trigger latencies for each trigger level in the new proposed scheme

In this scenario the acquisition will be started by a delayed copy of *L*0*d* and aborted immediately by the absence of *L*1 or later by *L*2*n* reject signal. In both cases an efficient way to abort an event acquisition and to reset the readout system has to be implemented.

4.3.1. Radiation effects

The effects of radiation on electronic circuits can be divided into total-dose effects and single-event effects. Total dose modifies the thresholds of MOS transistors and increases leakage currents. This is of particular concern in leakage-sensitive analog circuits, like analog memories. For instance, assuming for the storage capacitors in the

memory a value of 1 pF, a leakage current as small as 1 nA would change the value of the stored information by 0.2 V in about 200 μs. This is of course unacceptable. Radiation-tolerant layout practices prevent this risk and their use in analog circuits is therefore recommended. These design techniques become extremely effective in deep-sub-micron CMOS technologies. Single-event effects can trigger latch-up phenomena or can change the value of digital bits (single-event upset). Latch-up can be prevented with the systematic use of guard rings in the layout. Single-event upset can be a problem if it occurs in the digital control logic, and can be prevented by layout techniques or by redundancy in the system. A radiation-tolerant layout carries with it, of course, a penalty in terms of area. It can be estimated that in a given technology a minimum-size inverter with radiation-tolerant layout is 70% bigger than the corresponding inverter with standard layout. Nevertheless, a radiation-tolerant inverter in a 0.25 μm technology is about eight times smaller than a standard inverter in a 0.8 μm technology. The radiation dose which will be received by the readout electronics is quite low: 13 krad in 10 years for the inner layer. This value is probably below the limit of what a standard technology can afford; however conservative considerations suggest the use of radiation-tolerant techniques for critical parts of the circuit. These techniques have been proven to work up to several Mrad and allow a lower area penalty and lower cost compared to the radiation-hard technologies.

4.4. System Architecture

The basic principle of the SDD readout scheme is to amplify the signal coming from the detector, convert it locally to a digital representation, and send the data directly to the DAQ [30]. This architecture offers two main advantages, compared to schemes in which the analog data are transmitted far from the detector or to schemes with distributed intelligence, both of which have been considered for the ALICE SDDs [31].

- Converting the signal into digital samples immediately after the preamplifier avoids signal degradation during data transmission.

- Sending data directly to the DAQ system, without online data analysis, allows the use of more powerful and flexible offline data-analysis tools. Moreover, early failure detection is easier if raw data are directly sent outside the experimental area.

In principle this architecture would require a low-noise preamplifier, a fast ADC and some data-formatting logic. In practice the architecture is more complex, in order to cope with the limited power and the material budget.

4.4.1. Sampling and Conversion Frequency

The required sampling frequency is 40 MHz. At this speed it is extremely difficult to design an ADC with a power budget of only 5 mW/anode; the adopted solution is based on an analog memory to store temporarily the samples. When a trigger signal validates the data the analog memory content is frozen (after an appropriate delay to account for the total detector drift time) and the conversion process is started. In this way the ADC is activated only when the data are valid and can work at conversion rates lower than the sampling rate. Of course the conversion time should be kept as low as

possible in order to minimize the dead time. The size of the analog memory should be large enough to cover the detector drift time, which is 6 μs; at 25 ns sampling time 240 cells are required. A number of 256 cells has been chosen to be able to accommodate changes in the detector parameters. The current prototypes designed in 0.7 and 0.8 μm technologies feature an ADC conversion rate of 2 MS/s, or 500 ns conversion time and an analog memory settling time of 900 ns. These numbers give 1.4 μs for a single cell conversion, i.e. 336 μs for 240 cells and 358.4 μs for 256 cells: well below the 1 ms required readout time.

Moreover new prototypes are going to be re-designed by means of new sub-micron technologies. Nowadays the state-of-the art sub-micron technologies feature transistors with a minimum gate length of 0.25–0.35 μm and have several layers of interconnections (up to six for a standard 0.25 μm commercial process).These technologies offer a number of advantages for the implementation of the SDD readout electronics.

- The power consumption of the digital blocks is greatly reduced. The density of the digital circuits can be increased by at least a factor of 8 (even if radiation-tolerant layout techniques are applied).

- Because of the thinner gate oxide, the transconductance parameters of MOS transistors are increased. This means higher transconductance for the same current, with an improvement in the performance/power trade-off.

- The many available levels of interconnections allow shielding of critical signals even at the chip level and therefore a reduction of the cross-talk between analog and digital elements in a mixed-mode chip.

- Sub-micron technologies are recent processes and medium-term availability is not an issue.

Preliminary simulations with the new 0.25 and 0.35 μm technologies show that it will probably be possible to increase the conversion rate to 4 MS/s and to decrease the memory settling time to 400 ns, and to use one ADC every two analog memory channels in order to reduce ASIC area and power consumption with a small increase in readout time. The described operations are performed on hybrid circuits called front-end readout units, which are distributed on the ladder near the detectors. These readout units accommodate the front-end circuits and the power-supply filtering capacitors.

4.4.2. Solutions for a First Data Reduction

From the resolution point of view a 10-bit digital information should be required. Nevertheless, in order to reduce the amount of data from the SDDs, a preliminary data compression to reduce the number of bits per sample from 10 to 8 is foreseen in the front-end readout unit. The principle is to decrease the resolution for larger signals with a logarithmic or square-root law. Since the larger signals have a better Signal to Noise (S/N) ratio than the smaller ones, the accuracy of the measurement is not affected. This operation can be done in three ways.

- With a nonlinear response preamplifier. This solution greatly simplifies the design of the analog memory and of the ADC because they need an 8-bit instead of 10-bit resolution; on the other hand the calibration process complexity increases.

- With a two-reference ADC. This solution needs a 10-bit analog memory but an 8-bit converter. On the other hand the ADC needs a reference switch circuit, and problems with the threshold uncertainty can arise.

- With a digital threshold circuit. This is the most simple and flexible solution of the compression problem, but requires a 10-bit analog memory and ADC.

Owing to conservative considerations, the actual prototypes are based on the first solution. Prototypes based on the third solution are under development, while the second solution has been discarded because the design complexity of a double-threshold 8-bit ADC is comparable with that of a 10-bit ADC.

The minimum loss of information for the third solution can be obtained with the multi-threshold encoding shown in Table 4.3.

Input Range	**Output Codes**	**Code Mapping**	**Bit Lost**
0–127	128 → 128	0XXXXXXX	0
128–255	128 → 32	100XXXXX	2
256–511	256 → 32	101XXXXX	3
512–1023	512 → 64	11XXXXXX	3

Table 4.3. Digital compression from 10 to 8 bits

With this encoding the 128 Least Significant Bit (LSB) configurations of the 10-bit input range are directly coded into a 8-bit word with 0 as Most Significant Bit (MSB): all the configurations are mapped and the are no bits lost. On the other hand the second group of 128 input configurations, from 128 to 255, are compressed to only 32 configurations and encoded with 100 as MSBs: this means a compression from 2^7 to 2^5 configurations that gives rise to a 2-bit loss. The same method applies for the input configurations from 256 to 511 encoded with 101 as MSBs and from 512 to 1023 that are encoded with 11 as MSBs.

4.4.3. Event Buffer Strategy

After the conversion, the data are sent to a second readout unit, called end-ladder readout unit, placed at both ends of the ladder structure. This readout unit groups together the data provided by up to eight half-detectors and sends them to the DAQ through an optical fiber. Even with the first data compression, the amount of data generated by a single detector is very high; Table 4.4. summarizes the data volume generated by one SDD detector and the required bandwidth for transmission to the end-ladder readout unit. For simplicity and noise reasons, all the circuits will use the same 40 MHz clock. Assuming that the ADCs generate 512 bytes (one byte per anode) every 900 ns, then 16 8-bit buses per detector are required. The space on the ladder is very

limited and managing 128 data lines for each detector is a very serious problem, especially for the input connections. An alternative option is to slow down the conversion to 3.9 μs per cell, or 1 ms for the entire cell array. In this case only four 8-bit buses per detector are needed, but the dead time rises from 230.4 μs to 1 ms.

This is not a good solution, since increasing the readout time increases the dead time and imposes more stringent requirements on the decay time of the analog memory, and this, in turn, means larger storage capacitors and more power consumption. The adopted solution is to insert a digital multi-event buffer on the front-end readout unit. In this way the data coming from the ADCs are stored into a digital memory placed in the same readout unit, so that a very wide bus is not a problem. The data can be then sent to the end-ladder readout unit at lower speed, because if another event arrives in the meanwhile, another digital buffer is ready for data storage. In this way the readout speed can be tuned on the average event rate (~40 Hz). In the ALICE experiment a very small number of event buffers is sufficient to ensure a very low dead time. Our simulations indicate that with two event buffers and a readout time of 2 ms the dead time due to buffer overrun is only 0.1% of the total time. With the double event buffer, the converter can work at full speed and the data can be transferred to the end-ladder readout unit in less than 1.65 ms with only one 8-bit bus for each half-detector.

Anodes/detector	512
Samples/anode	256
Total data volume	512x256-bit = 128 kbyte
Readout time	1 ms
Required bandwidth	128 Mbyte/s

Table 4.4. SDD data volume and bandwidth requirements

4.4.4. Front-End Power Budget

Table 4.5. summarizes the average power budget for the various parts of the front-end readout unit.
The 1 mW/channel for the ADC is too low to obtain the required performances; however with switched-capacitor techniques it is possible to design converters with very low power consumption during the idle state. Looking at the trigger rates, in the worst case (p–p at minimum bias) the ADC is active, on average, less than 9.2% of the time, while in the normal situations (Pb–Pb or Ca–Ca) the active time is $\leq$ 4.8%. The power budget for the ADC in converting state can increase to 10 mW for the 2 MS/s converter and to 20 mW for the 4 MS/s converter.

Preamplifier	< 1 mW/channel
Analog memory (write)	< 2 mW/channel
Analog memory (read)	< 0.5 mW/channel
ADC	< 1 mW/channel
Event buffer	< 0.5 mW/channel

Table 4.5. Average power budget for the front-end board components

The actual prototype dissipates 10–11 mW/ADC at 2 MS/s including control logic and the external Sample and Hold (S/H) circuits. Similar considerations are valid for the event buffers and the line drivers. Standard digital memories supplied by silicon foundries dissipate half of the power budget, while the rest will be used by the control circuitry (which has very low power consumption) and the line drivers. Detailed analysis on the power consumption of the line drivers has not yet been performed, but since these drivers are shared among all the channels of a half-detector, they should not be critical in terms of power.

4.4.5. Solutions for End-Ladder Data Reduction

In order to meet the data size requirements, the following data reduction algorithms have been studied.

- *Zero sequence encoding*.
  Sequences of zeroes are transmitted as a zero code followed by the number of consecutive zeroes. Since the occupancy is quite low, long zero sequences are highly probable.

- *Simple threshold zero suppression*.
  The data below a certain threshold, which takes into account noise and pedestal, are set to zero. This technique is very easy to implement and increases the number of zeroes by cutting non-zero values due to the noise. Unfortunately this results in information loss.

- *Differential encoding.*
  Instead of the samples, the difference between consecutive samples is transmitted. In this way any channel baseline value is translated into a zero sequence. On the other hand a differential encoding scheme is more sensitive to sample errors during transmission.

- *Simple threshold tolerance.*
  This is a single-threshold zero suppression applied after the differential encoding. It reduces the noise variations over a baseline, at the expense of information loss.

- *Huffman encoding*.
  Since the probability of lower codes is much higher than that of higher ones, using a variable length encoding leads to a loss-less data reduction. This reduction depends on the sample statistics; the implementation is quite demanding in terms of hardware requirements.

- *Multi-threshold zero suppression.*
  A sample is set to zero depending on its value and on the value of neighboring samples. In this way the information loss can be greatly reduced with respect to single-threshold zero suppression.

A loss-less compression system is, of course, the best choice but requires very complex electronics and does not guarantee a sufficient compression ratio in any

situation, since it is based on data statistics. A Field Programmable Gate Array (FPGA)-based prototype of the first five algorithms, with software-tunable parameters, has been developed and is currently under test with the data taken from the ALICE SDD beam tests [27]. The sixth algorithm is currently under evaluation. A detailed description of this algorithm can be found in Ref. [32].

4.4.6. Test and Slow Control

Since the readout units and ladder assembly is very critical, it becomes very important to be able to test the various parts in the different phases of the assembly process. For electronic circuit tests, the IEEE 1149.1 protocol has been adopted, which is the standard protocol for testing at the board level and which has been chosen by the whole ALICE ITS. The same protocol will be used also for downloading configuration information onto the readout units.
Figure 4.1. shows the readout architecture.

4.5. Front-End Readout Unit Design

The front-end readout unit, shown in Fig. 4.2., is a hybrid circuit containing four sub-modules of the pre-amplification, analog storage and ADC architecture (called PASCAL) and a multi-event digital buffer (called AMBRA) integrated-circuit pair. Considerations of space on the ladder and cost lead towards the highest possible level of integration. Owing to the very high switching noise it is not possible to integrate the digital memories on the same substrate with analog parts, so at least two ASICs are needed. For conservative reasons the current prototypes of the PASCAL architecture

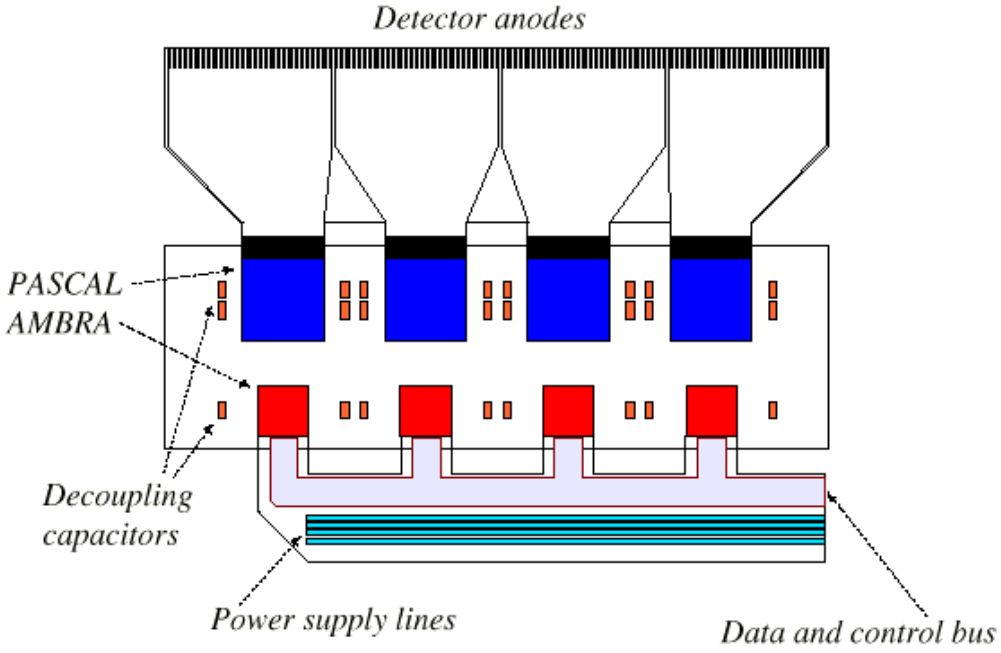

Figure 4.1. The SDD Ladder Readout Architecture

designed in 0.7–0.8 μm technologies are based on three different ASICs: the first with the preamplifier, the second with the analog memory, and the third with the ADC and the control unit. The ADC prototype matches the ALICE requirements, while the nonlinear preamplifier should be improved in terms of dynamic range (limited to 20 fC in this design). Promising results have also be obtained with the linear preamplifier and the analog memory.

Prototypes of the analog memory and the ADC at 10-bit resolution in 0.25 μm technology are currently in production. The simulations show, as expected, significant

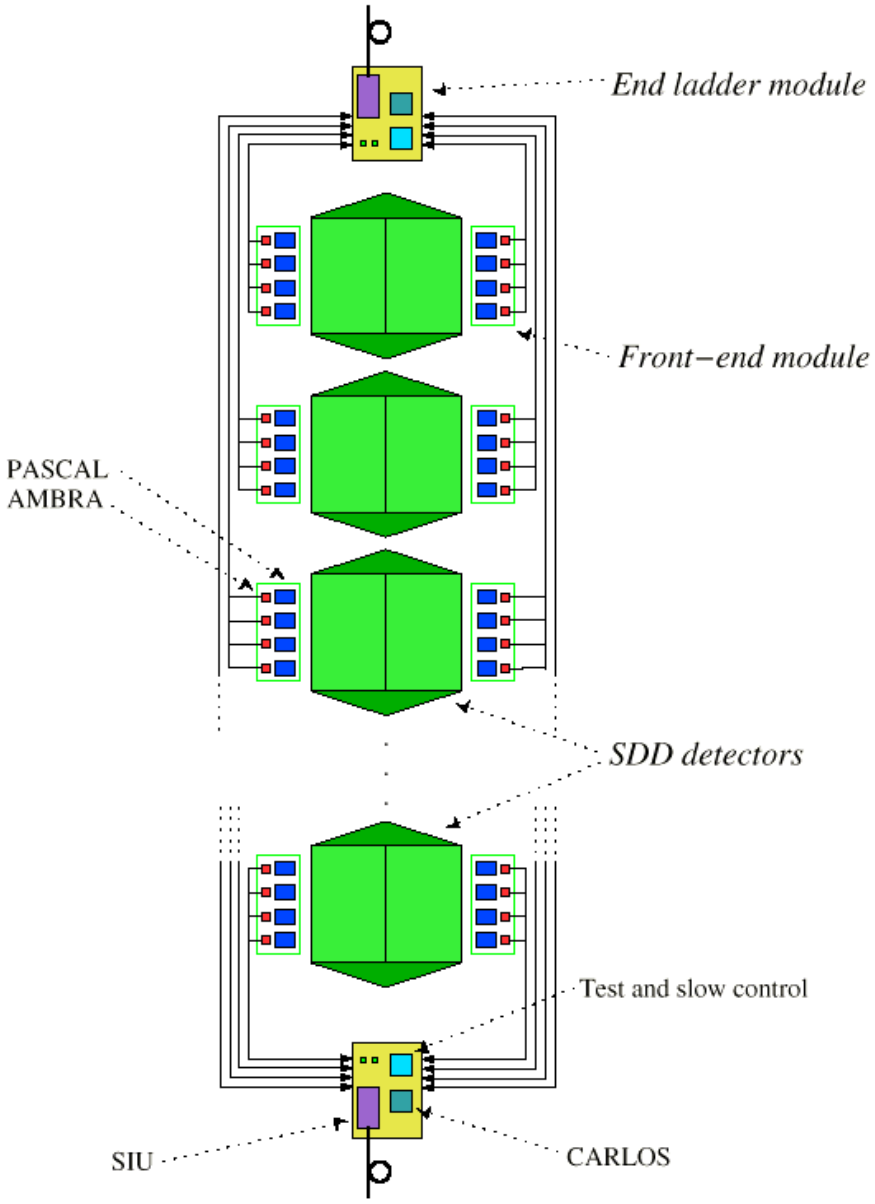

Figure 4.2. The Front-End Readout Unit

improvements in terms of speed, power consumption and area. If the simulation results are confirmed by the prototype tests, the next step will be to design an ASIC which contains the three parts of the PASCAL architecture. Since the preamplifier/analog memory group and the ADC are never active at the same time, we do not foresee insurmountable problems in the integration of these three parts on the same substrate. Both the analog memory and the converter are designed with switched-capacitor techniques and work with the same clock signal, and therefore the effects of the switching noise should be rather low. Special layout techniques will be used in order to shield the preamplifier from the rest of the circuit. ASICs which integrate the three units have already been designed and tested [33, 34]. Concerning the number of inputs per chip, most of the prototypes of the PASCAL functional blocks have been produced with 16 inputs per chip or fewer. However, with the technology scaling down and the improvement of the silicon processes, our design target of 64 inputs per chip is feasible with reasonable yield. The readout unit is placed near each side of the SDDs; it is connected to the rest of the system through the following lines.

- *reset*: global reset.
- *clock*: global clock.
- *data in[255:0]*: 256 input lines from the detector. They are divided into four 64-wide bus lines and are connected to the PASCAL chips.
- *pa cal*: calibration line for the PASCAL preamplifier.
- *trigger*: start of an event acquisition.
- *dis trigger*: tells the end-ladder readout unit that the front end is not ready to accept a trigger signal.
- *abort*: aborts an event acquisition due to a reject signal from the trigger system.
- *data out[7:0]*: data lines to the end-ladder readout unit, driven by the AMBRA chip.
- *data write*: tells the end-ladder readout unit that data on the *data out* lines are valid.
- *data stop*: the end-ladder readout unit is not ready to receive data.
- *data end*: tells the end-ladder readout unit that all data have been transferred.
- *jtag bus[4:0]*: test and slow-control serial bus based on the IEEE 1149.1 standard.

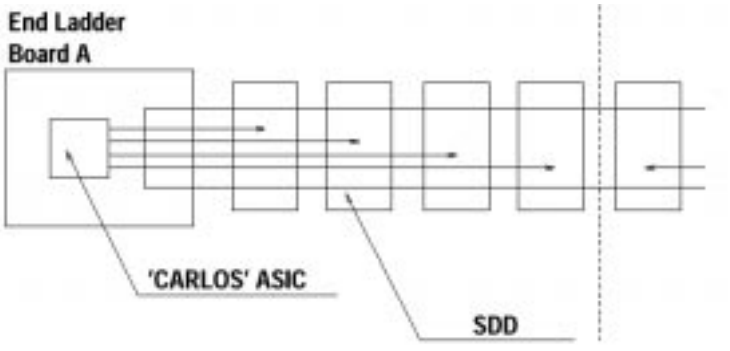

Figure 4.3. Structure of a Ladder

4.6. End-Ladder Design

Each end-ladder (see Fig. 4.4.) board will host an ASIC, which will provide the compression of data coming from the front-end modules, clock fanout and, possibly,

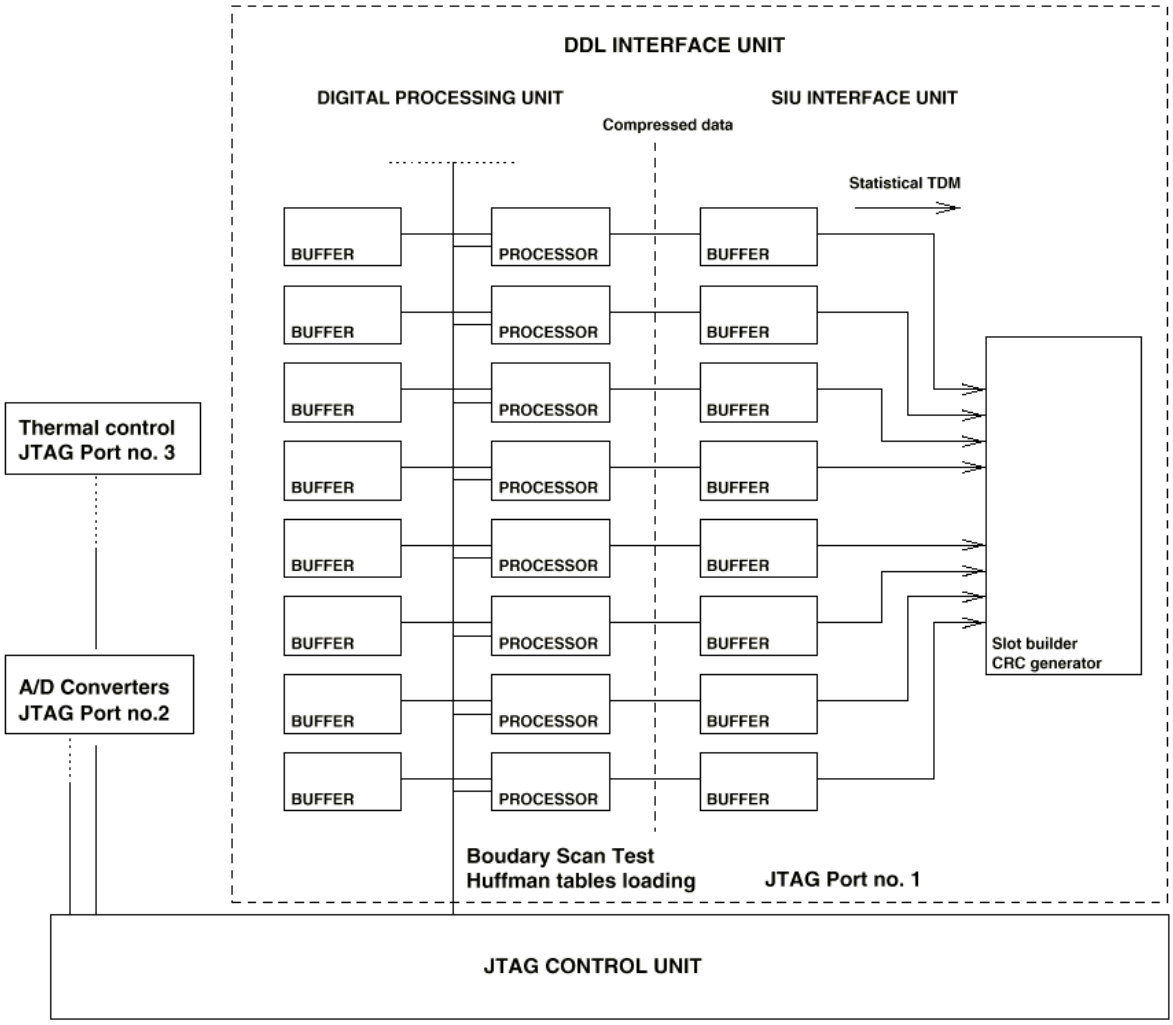

Figure 4.4. Schematic Representation of the Front-End IC

power regulators for the ladder systems. The main architecture of the ASIC named CARLOS is reported in Fig. 4.4. It consists of eight identical channels. Each channel processes data coming from one half-SDD: there are eight half-SDDs per half-ladder, and two SDD compression subsystems on each ladder, see Fig. 4.3. for details. Each compression subsystem has to work on eight channels in parallel to provide a real-time readout and processing of the event data. This is done on an L2 trigger event.

4.6.1. Architecture Description

As mentioned above, CARLOS is primarily a dedicated compression sub-system and provides buffering on the input side and multiplexing on the output. We will call ‘macro-channel’ the hardware pipe beginning at the input buffer and ending at the slot

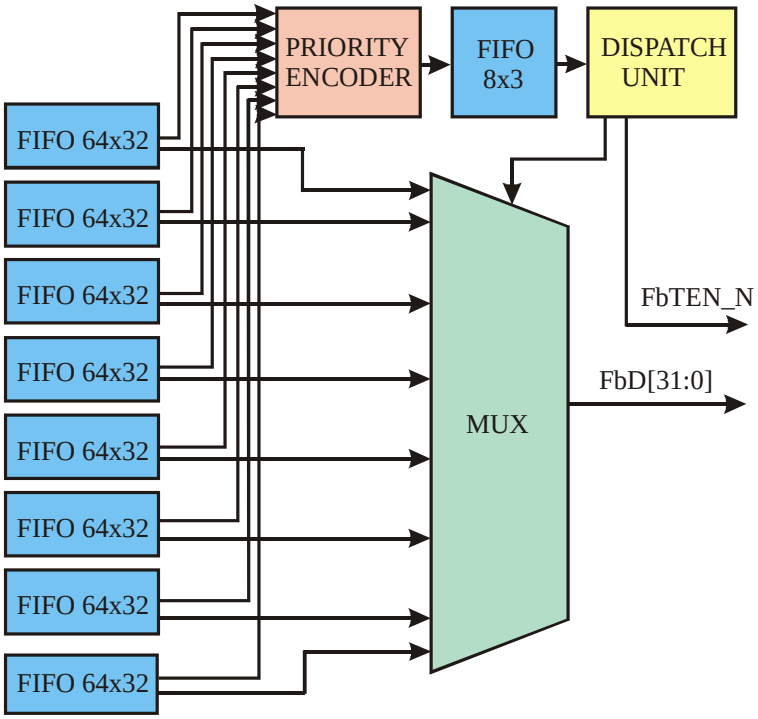

Figure 4.5. Output stage of the front-end IC

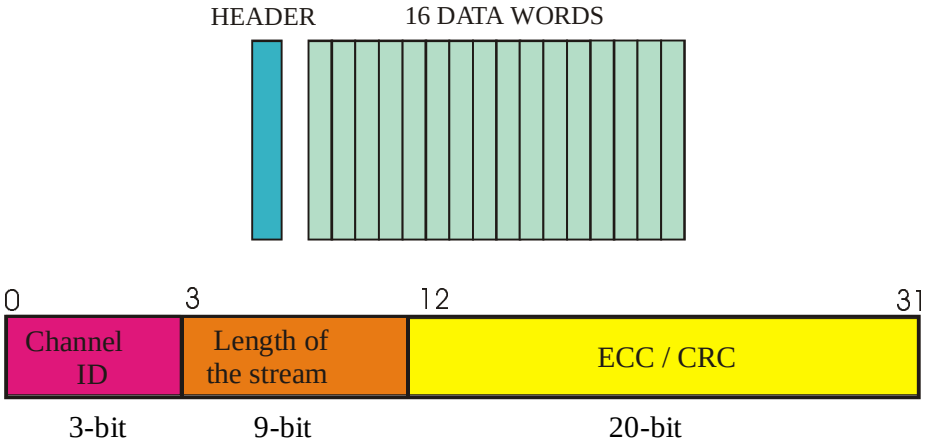

Figure 4.6. Bit Assignment of the Packet Header

builder (see Fig. 4.4.). Each macro-channel is a sequence of a buffer, needed to provide a backward handshake with the front-end electronics, and a processor which is the hardware implementation of the algorithms described in the previous section. CARLOS hosts eight macro-channels and a unit to multiplex the eight data flows that each macro-channel produces. This unit is also responsible for the packing and tagging of data and for the CRC generation. This architecture is scalable, which makes an eventual expansion of the number of macro-channels served per IC very easy.

4.6.2. Bandwidth Allocation

The eight input streams coming from the detectors are basically asynchronous. Because of the compression processing they cannot be considered as eight homogeneous streams to send to the fiber. CARLOS has been designed to perform the statistical multiplexing of the eight channels and to send packets of data coming from the same channel, tagging them with a channel identifier and with redundancy information. A good implementation of this Time Domain Multiplexing (TDM) scheme shall limit the buffer memory needed to make the system work without losses. In a typical case, when the amount of data coming from the detectors is roughly the same, the statistical multiplexing will become uniform between channels providing a highly predictable transfer rate to each channel. In the case of differences between data streams on each channel this scheme will allocate the bandwidth in a rather efficient way. The multiplexing of the eight channels in time domain has been implemented by a mechanism similar to an interrupt request. An overall view of the output stage is shown in Fig. 4.5.

4.6.3. Transmission protocol

The output stage is also responsible for assembling packets containing the following:

- A header containing the channel IDentifier (ID), the length of the information stream in the packet, the event number and other information, e.g. ECC. The allocation of bits inside the header is represented in Fig. 4.6.

- 16 32-bit data words.

The redundancy of the protocol is 1/16, equal to 6.25%.

4.6.4. CRC implementation

The remaining 8 bits in the two control words can be used for error recovery (CRC) and/or for future applications. The choice of the specific polynomial, for the CRC implementation, has not yet been made.

4.6.5. Joint Test Action Group (JTAG) Protocol

The interface-board design contains test structures (design for testability) as well as boundary scan registers, etc. Test procedures, such as detector control, will be implemented according to the JTAG protocol. The design of the JTAG control unit and of the JTAG interfaces is under development.

4.6.6. Detector Control

CARLOS also provides an interface to implement a virtual circuit JTAG channel thanks to the SIU featuring this extra kind of connection. This JTAG channel will be

used for loading and downloading parameters of the ladder systems and the set-up procedure of CARLOS itself.

4.6.7. Status of the Project

At the moment, a full description of the CARLOS architecture in VHDL is available. For a pure digital chip, the generation of the layout from the VHDL code is a highly automated process and does not add a significant time overhead. Implementation on silicon is nevertheless expensive. Therefore, for the prototype field-programmable gate arrays (FPGA) have been used. In particular, two XILINX (TM) chips host a macro-channel of the whole system. In this way the compression algorithms and the communication protocols can be efficiently tested and optimized at very low cost before the actual implementation on silicon takes place. At the moment a simple prototype of CARLOS has been developed by the use of two XILINX (TM) FPGAs. These two ICs together host one macro-channel of the whole system and have been programmed to test the efficiency of the compression pipeline. The two FPGAs have been mounted on a PCB and one of these implements a simple SIU interface so that we can test also the compliance of our hardware with the transmission protocol used by the SIU-DIU system. A picture of the prototype is shown in Fig. 4.7.

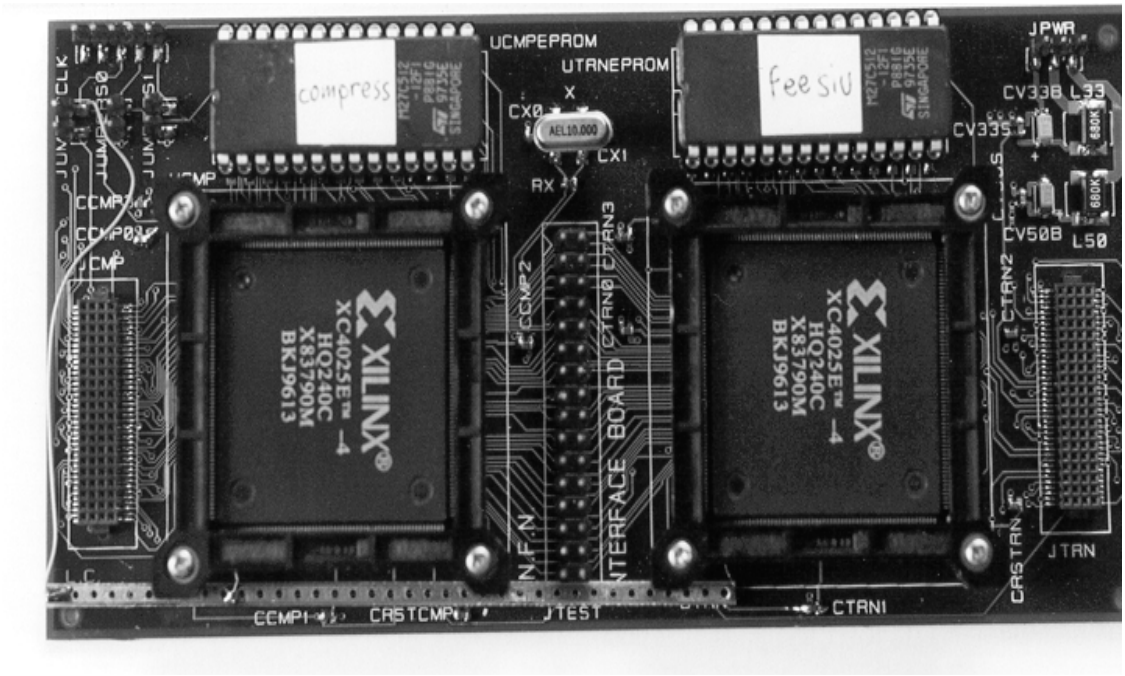

Figure 4.7 Photograph of the CARLOS Prototype (one macro-channel)