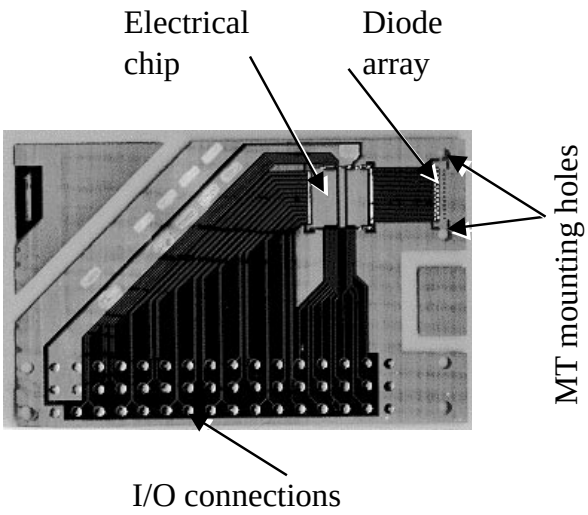

Fig. 5.28: Electro-optical package.

5.9 Upper-Level Readout

5.9.1 Introduction

As mentioned in Section 5.3, data will be transported from the detector to the counting room directly after digitization by fibre-optic link. This relieves the requirement for large quantities of radiation-hard digital electronics, and eases the time-scale for upper-level readout production. In this section, the upper-level readout, interface to the trigger and data acquisition are described.

The upper-level readout stores the floating-point ADC values in a pipeline, and creates tower sums for the trigger. As described below, several circuits have been developed to perform these functions:

- the linearizer, which subtracts the pedestal and multiplies by the gain;
- the pipeline, which stores the data while waiting for the trigger decision;
- the adder, which creates the tower sums;
- filter-1, which extracts the trigger signal from the stream of trigger sums;
- filter-2, which allows hardware processing of the data for upper-level triggers.

Figure 5.29 shows the basic layout of the upper-level readout with the optical receiver, data linearizer, trigger- and DAQ-path. The optical receiver provides the deserialization of the data from the VFE and, by using the link error detection protocol and synch. information, provides a ‘Data integrity check’. This check generates a flag bit which is added to the data and, in the trigger path zeroes the data in that channel, while in the DAQ path the data is unchanged but accompanied by the flag information, thus allowing the later processes to take into account the uncertain state of the data.

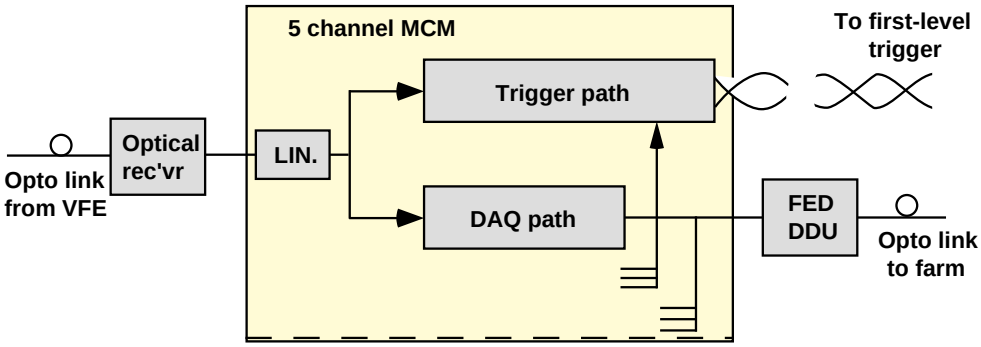

Fig. 5.29: Upper-level readout.

The linearizer transforms the floating-point representation to a linear 18-bit representation which allows subsequent processes, like the trigger summation, to use the data without further conversions. Once the data is linearized it is applied to individual programmable thresholds and phi-strip trigger sums are formed by adding the five channels of the strip. This sum is applied to both an energy extraction filter and to a bunch-crossing identifier as shown in Fig. 5.30. The extracted energy is conditioned by the bunch-crossing identifier in order to provide information which is formatted and sent to the trigger primitives generator.

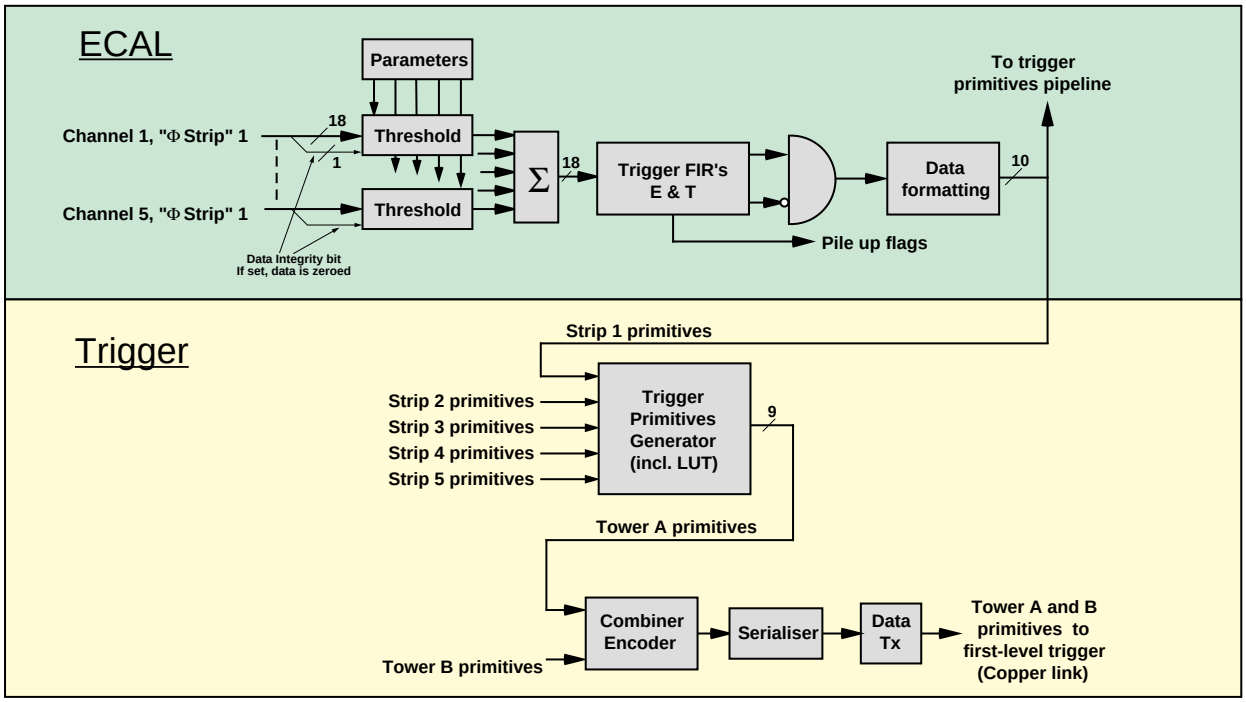

Fig. 5.30: Trigger path.

In parallel, the linearized data is also applied to a pipeline of programmable length where the data is stored during the Level-1 trigger latency. At each Level-1 Yes a set of consecutive samples, a time frame, is extracted from the output of the pipeline, i.e. the data corresponding to the Level-1 decision, and stored in one of the eight derandomizing buffers, eventually together with the BCI information given by the TTC system. The readout controller takes the data from the individual channels, formats the data, adds the Event ID and stores the data block in the output buffer. The data can either be the complete time frames or the result of the LVL-2 filter process.

The upper-level Readout is built around 9U VME crates, see Fig. 5.31, each containing 18 readout modules, a readout master, a VFE controller and a local control and readout CPU. This arrangement allows one crate to contain all the electronics required to read out and control a barrel supermodule. For the endcaps a similar arrangement is being developed. The major part of the digital electronics for the individual channels could be implemented in the form of a Multi-Chip-Module (MCM) containing five complete channels. This corresponds, for the barrel, to the trigger requirement to provide a sum for each phi-strip in the trigger tower. The organization of the MCM substrate is shown in Fig. 5.32.

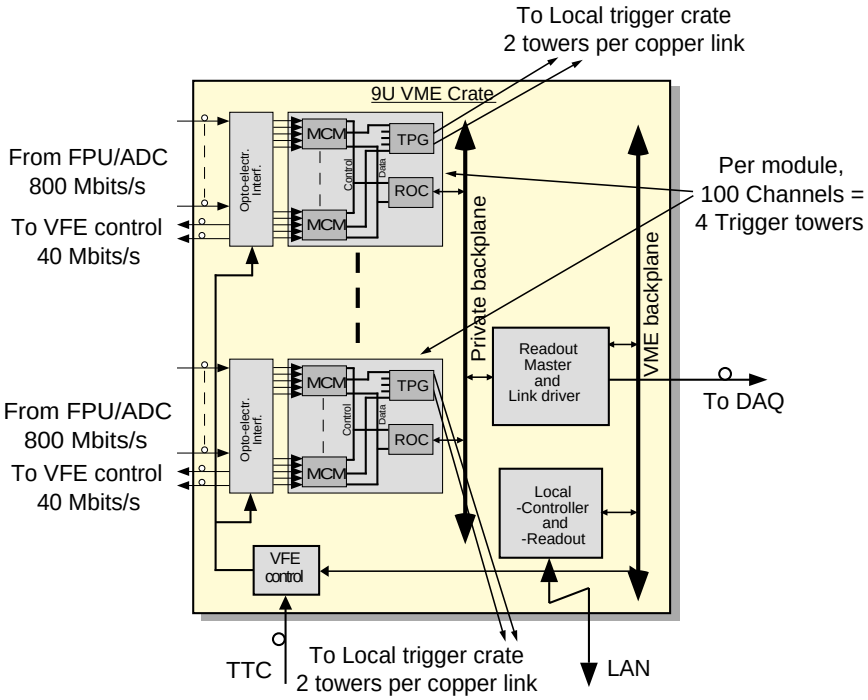

Fig. 5.31: Readout module.

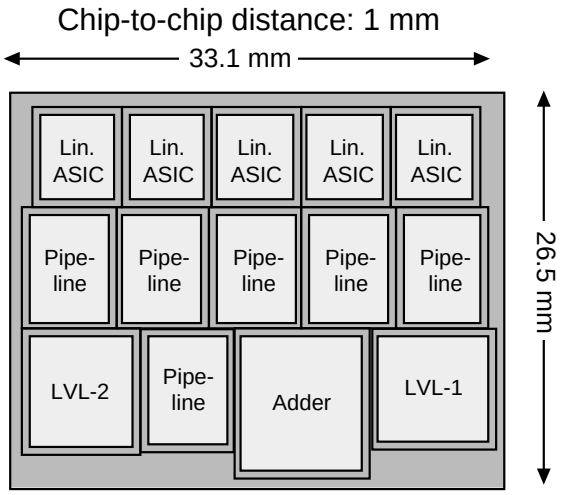

Fig. 5.32: MCM organization.

All functional blocks have been prototyped and tested in different forms and below is a short summary of each physical ASIC development and its status. There are within all ASICs a certain number of programmable features like coefficients for filters, constants for thresholds, status registers, length of pipeline, etc. An internal control bus is routed within the MCM to perform these operations under the control of the MCM readout controller.

Also, each functional block has error-detection or -correction circuits built in and, depending on the type of error the ASIC will raise an error flag indicating either that the error was successfully corrected (non-fatal error) or that the data has been corrupted or that the process failed (fatal error). In the latter case information sent to the trigger process will be zeroed while the information read out by the central DAQ system will be flagged as incorrect.

5.9.2 Linearizer ASIC

The Linearizer ASIC, see Fig. 5.33, processes and combines the data from the four slopes of the FPU. Data from the slopes are joined into one single linear 18-bit representation, controlled by the two range bits, by adding a (programmable) term corresponding to the step created by the change of slope and thereafter multiplied in two steps with the gain ratio. The first multiplication coefficient (programmable) has a value close to 1 with a 10-bit resolution and the second operation is done by a barrel shifter where the data is multiplied with a programmable integer binary value between 1 and 32. These two operations provide a result where the input data is multiplied with a composite coefficient between 1 and 63.999, thus giving a precision of the order of 1 in 1024.

A bypass function exists for the data-integrity bit generated by the optical link from the VFE circuit and a delay is provided in order to keep the synchronism between the data and the integrity bit.

Also, a look-up table is included and, for the ECAL application, it serves as the means to load and inject a predefined set of data that is injected into the system with the 40 MHz clock. This allows a powerful check of the entire system including trigger primitives generation, derandomizer loading and Level-2 filter functions.

Figure 5.34 shows the layout of the ASIC which is currently under fabrication and expected back from foundry in December 1997.

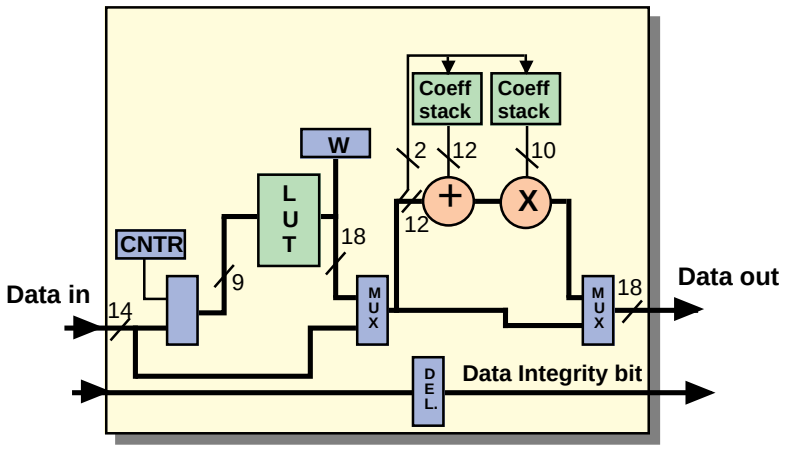

Fig. 5.33: Linearizer block diagram.

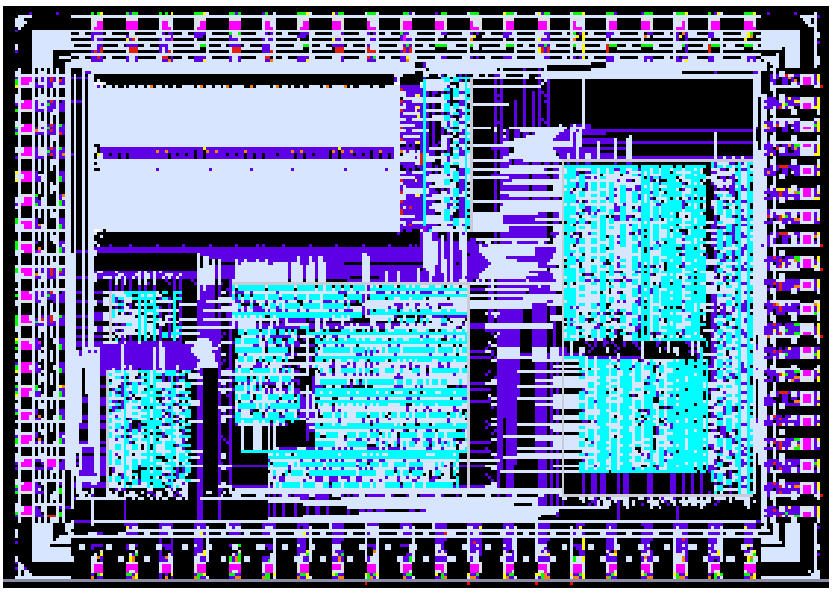

Fig. 5.34: Layout of the linearizer ASIC.

5.9.3 Pipeline ASIC

The pipeline ASIC consists of a programmable length pipeline, built as a rotating buffer, and a set of derandomizing buffers.

Data from the linearizer is combined with the flags from the Filter 1, encoded in a ECC envelope and written into the buffer at each machine clock (Fig. 5.35). The length, i.e. the delay, is programmable between 4 and 256 clocks. A bypass path is built in for test purposes.

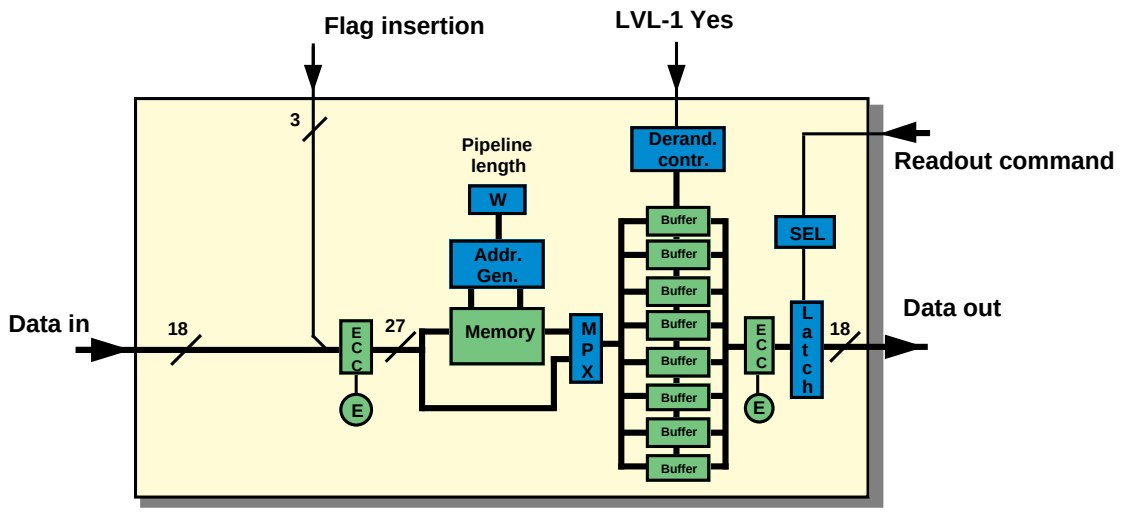

Fig. 5.35: The pipeline ASIC.

At each first-level Yes a time frame, 16 samples long, is written into the next free derandomizing buffer. If a new Yes arrives within the 16 clocks a new buffer is opened and another time frame is written into it. This creates complete frames for each trigger even in cases of severe pileup.

At the output, the data passes an ECC decoder which will correct any transient bit error occurring within the storage elements. In case of a correction the ECC decoder will issue a non-fatal error.

Three versions have been implemented (Fig. 5.36) the first of which, implemented in ES2 0.7 µm technology (155 mm^2 area), has been used in H4 beam tests. The second version (0.8 µm AMS technology, 85 mm^2) is under test with the MCM V2 and the last one (0.8 µm AMS technology, 40 mm^2) is used for the *Proto97* readout.

The final version will be implemented in a 0.5 µm CMOS technology with an estimated chip surface of 20 mm^2.

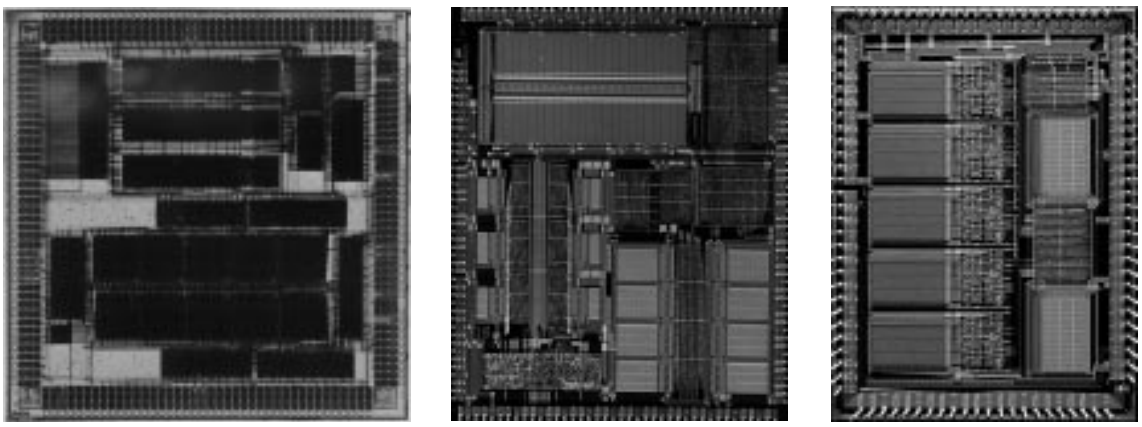

Fig. 5.36: Microphoto of the three pipeline ASIC versions.

5.9.4 Adder ASIC

The Adder ASIC, see Fig. 5.37, generates the sum of the five channels in a phi-strip.

Each input has an individually programmable ‘threshold’ function which zeroes data below a defined value, as well as in case of a data error flagged by the data integrity bit. Also, when the value $FFFFF_h$ is loaded the channel is switched off, i.e. the value is always zero. The adder is protected by a residue 3-code process which generates a fatal error in case of mismatch between the reside sum and the encoded 3-code of the output value.

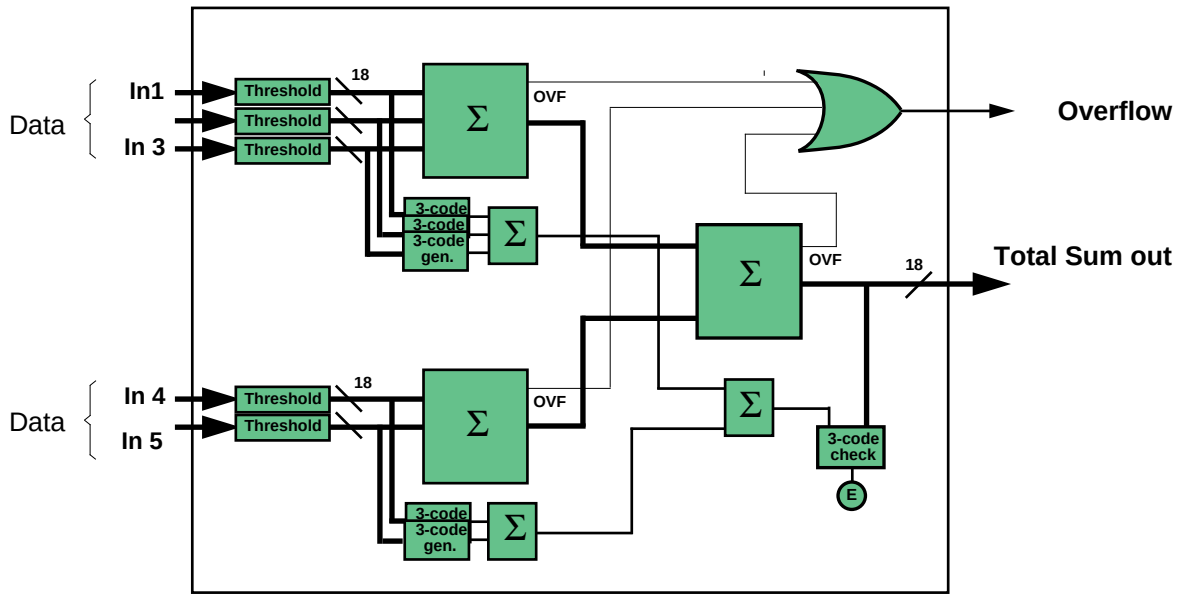

Fig. 5.37: Adder ASIC block diagram.

The two first versions of the above-mentioned channel ASIC also contained the adder function but to improve flexibility and adaptability the function has been extracted and is now a separate ASIC. The above-described version is implemented in AMS 0.8 µm CMOS and, as can be seen in Fig. 5.38, the surface is far from optimized as it was necessary to use classical wire bonding into PGA packages for the first prototypes. The final version will be adapted to flip-chip bonding, thereby eliminating all unnecessary surface. The ASIC is fully tested in the packaged version.

The conversion to the final 0.5 µm technology will shrink the chip even more and it will have an estimated surface of 18 mm^2.

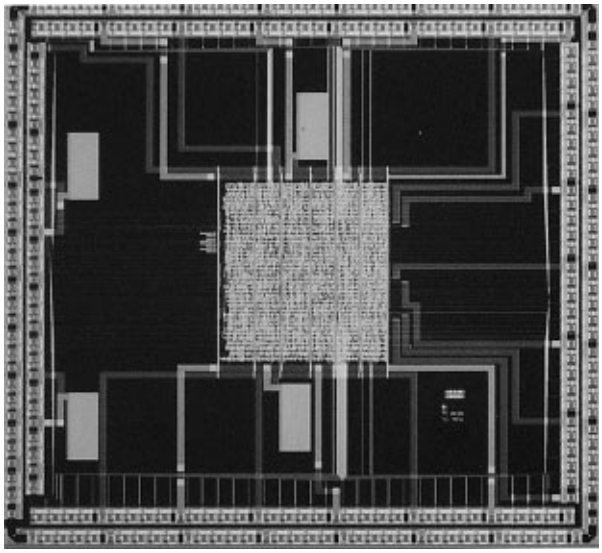

Fig. 5.38: Microphoto of the adder ASIC.

5.9.5 Level-1 filter ASIC

The Level-1 filter ASIC, see Fig. 5.39, takes the linear data summed by the adder ASIC and performs the feature extraction required for the trigger process. It consists of two 6-tap FIR filters, one optimized for energy extraction and the other for bunch crossing identification (BCI). Each tap in the two filters has loadable coefficients with a width of 7 signed bits. The energy FIR processes with full resolution in order not to bias the results, while the timing FIR truncates two LSBs. At the input of each filter is a 3-code generator followed by a 3-code filter and, at the output, a 3-code comparison between the results of the filter and its 3-code filter. In case of mismatch a fatal error is generated. Also, the corresponding 3-code coefficient and a parity bit are added to each tap coefficient, the latter to assure that no transient errors have occurred in the stored value.

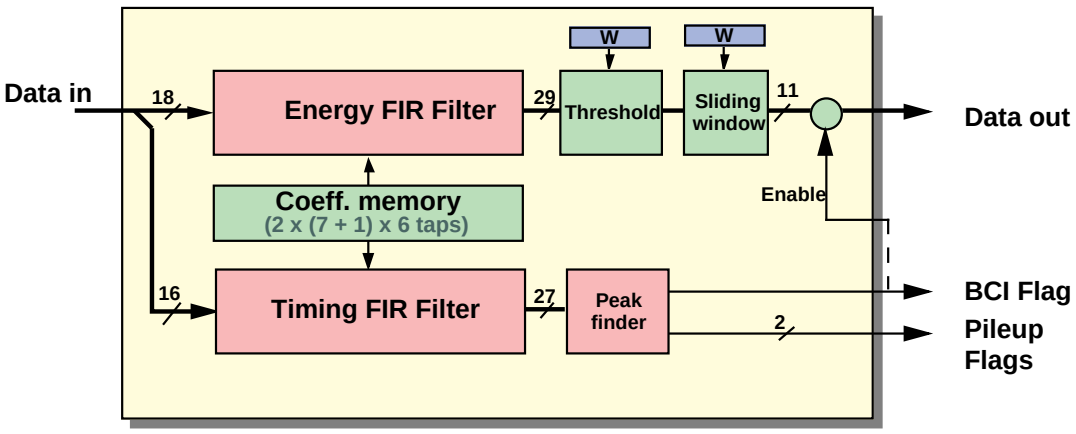

Fig. 5.39: Level-1 filter ASIC.

Following the energy filter is a programmable threshold function and an equally programmable sliding window which selects up to 11 bits of the energy. The timing FIR is followed by a peak or leading edge finder that identifies the time origin of the signal, called BCI flag. Also, this circuit detects possible pileup conditions by counting the number of clocks between adjacent BCI flags and two values can be loaded to define pileup and severe pileup. The corresponding flags are generated for insertion into the pipeline. The BCI flag is also used to condition the energy output in order to produce zero energy except at the correct bunch crossing.

A first version of the ASIC was implemented in the ES2 0.7 μm technology (65 mm^2) and used in the H4/96 trigger tests, together with the above-mentioned channel ASIC. A second version using AMS 0.8 μm technology (58 mm^2), has been produced and tested in the laboratory.

In the final technology the surface of this chip is estimated to be 20 mm^2. Figure 5.40 shows the microphoto of the ES2 version and Fig. 5.41 the AMS version of the ASIC.

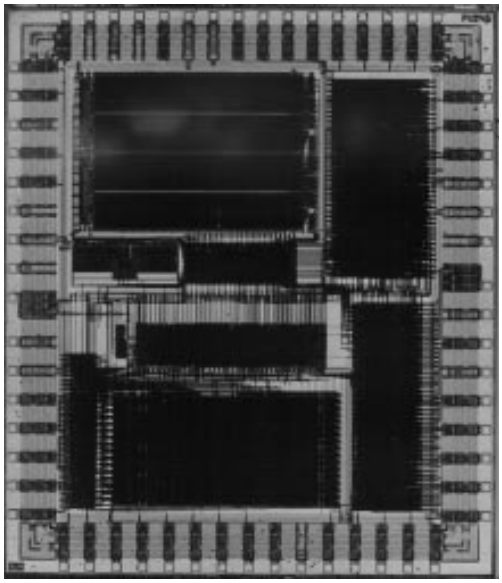

Fig. 5.40: Level-1 filter ASIC.

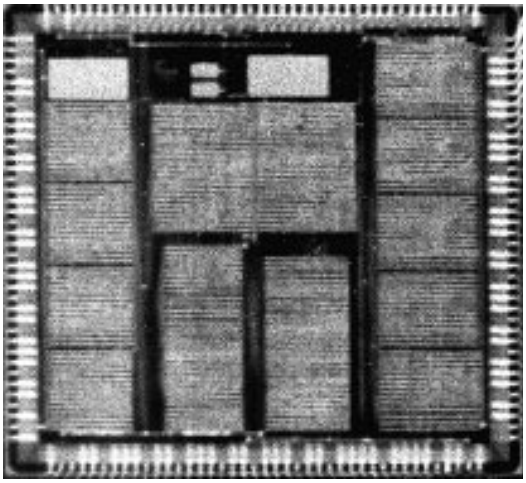

Fig. 5.41: Level-1 filter ASIC.

5.9.6 Level-2 filter ASIC

The Level-2 filter ASIC, see Fig. 5.42, consists of three pipelined FIR taps and an order statistics operator supervised by the readout controller. Each FIR filter has an associated coefficient memory consisting of eight banks of 16 locations which can be selected according to event conditions (flags) or by external commands. From the insert it can be seen that the coefficient word contains a 10-bit signed FIR coefficient, the corresponding 3-code coefficient and a parity bit.

Each FIR is, by its coefficients, optimized for a particular condition, like low signal-to-noise ratio, pileup conditions or to the fluctuations in detector response. The OS operator selects the FIR giving the best result by sorting them according to max., min. or median criteria. This creates a similarity with ‘loops’ or ‘case’ statements in a pattern-recognition routine.

Two different error checks are done, parity check on the coefficients and a 3-code check of the filter function, both giving Fatal errors.

The Filter 2 has been designed and is expected back from foundry in early December 1997. Figure 5.43 shows the layout of the filter.

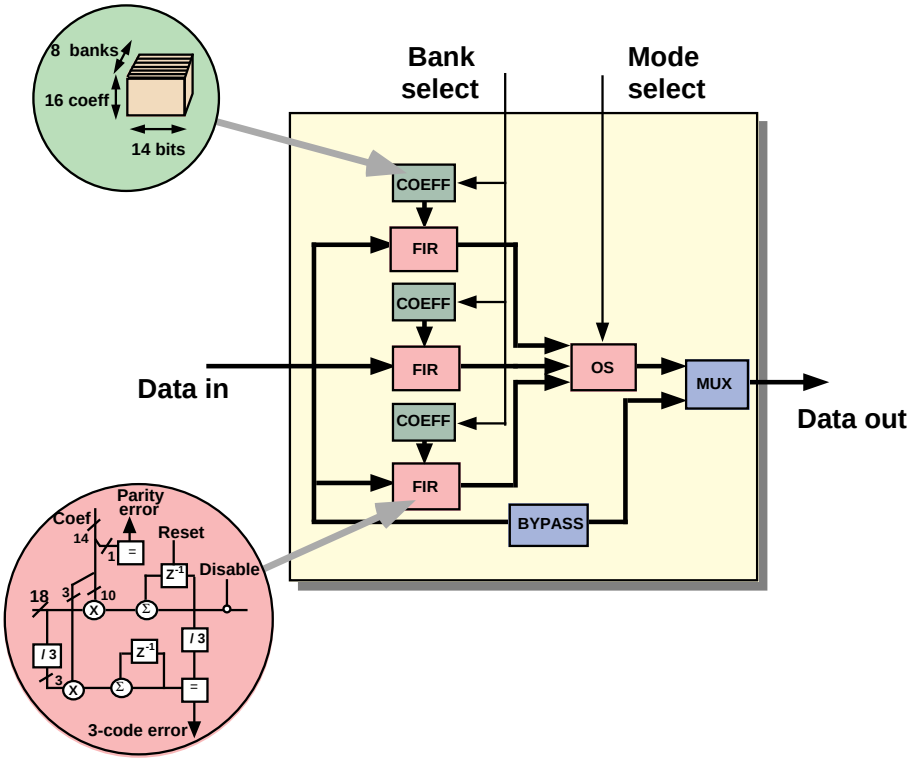

Fig. 5.42: The filter-2 ASIC.

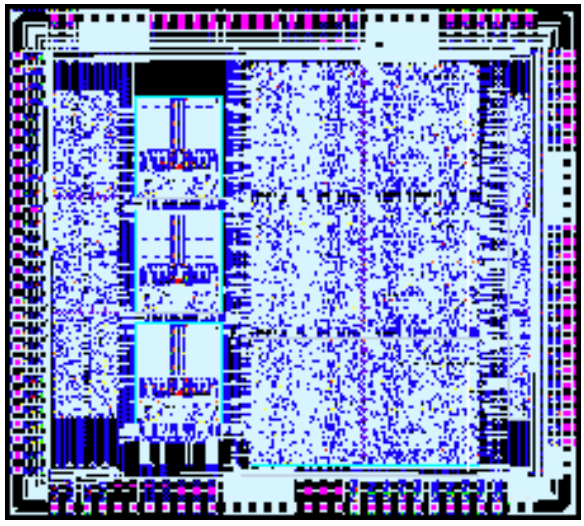

Fig. 5.43: Layout of the Level-2 filter ASIC.