# 7 Calorimeter trigger demonstrator programme

## 7.1 Introduction

The ATLAS level-1 calorimeter trigger group has used the ATLAS test beam every year from 1992 to 1998 as a test bed for processes and technologies which will be necessary in order to design and build the final calorimeter trigger for ATLAS. These test-beam operations form an overall programme of performance testing which demonstrates the feasibility of all the demanding parts of the system. Initially, design and test-beam work was performed as part of the R&D programme RD27, before this developed into the current ATLAS and CMS trigger programmes.

As already described, the ATLAS calorimeter trigger requires pipelined processing of a large number of trigger towers, and extensive data fan-out in order to perform the physics algorithms. All this has to be performed in a very short time (typically ~40 bunch-crossings from raw data to final result). In order to develop hardware which could achieve these goals, it was decided to test out ideas and electronics in the harsh reality of a test-beam environment, using prototypes of the final calorimeters as data sources.

The philosophy of the demonstrator programme [7-1] was to produce a small-scale version of the trigger system which contained all of the most important elements of the final design. In particular, each of the stages of the process which were identified as being critical should be performed in the demonstrator system. Alternative technologies were tried for various parts of the system throughout the programme in order to assess which would form the best basis for the final solution, so it was envisaged that some ideas would have to be abandoned, but that by the end we would have enough pointers to make the most reliable design, and sufficient evidence that this design would provide the performance needed for the final ATLAS level-1 trigger.

## 7.2 Brief history

### 7.2.1 Data source

For the test-beam runs from 1992 to 1995, data were mainly recorded using the RD3 liquid-argon (LAr) accordion calorimeter prototypes in the North Area at CERN. The signals were summed to form a convenient $6 \times 6$ array of detector towers, which were similar in arrangement to the final trigger towers in the ATLAS detector. This was a very useful geometry for testing the cluster-finding aspect of the trigger algorithms. The major difference from final ATLAS is that the area covered by these towers was smaller than the eventual tower sums of the LAr calorimeter, meaning that the noise seen in the test-beam set-up is about one-half of that which may be expected at ATLAS.

In 1996 and 1997 signals from the module-0 Tile Calorimeter were used. In this case, the geometry was not so convenient for the hard-wired trigger algorithms, but it still served as a useful insight into how the system responded to different calorimeter signal pulse shapes.

There was also significant testing performed in the laboratory at RAL during 1998 using pseudo-random input data.

### 7.2.2 Data acquisition technique

The means to investigate the performance of the hardware was based on a fairly simple data acquisition setup. Data flowed through a sequence of modules, and wherever appropriate, were clocked into circular memories on each ‘bunch-crossing’ tick. When an interesting event occurred (usually triggered by the standard beam-line scintillators), the whole system was stopped in order to be read out. This meant that the progress of a particular sequence of data could be followed through the various elements of the processing chain. Clearly this is not the way readout will be performed in the final system, but it gave a simple way of testing the integrity of data processing without developing buffered readout.

Online analysis could quickly check for some errors, but the data were also recorded to tape for later offline analysis. Thus the data integrity throughout the system could be determined, allowing measurement of bit-error rates and latency stability. Unfortunately, in terms of statistics for very low bit-error rates, this comparison of data offline only provides a fairly limited verification. Most of the results quoted come from this sort of analysis, and so are statistics-limited. Other techniques, as detailed later, were developed to test some aspects of the system in real time and obtain a better estimate of the system performance.

### 7.2.3 Test-beam summary

The first RD27 test-beam occurred in November 1992 [7-2], although at that stage very little custom-built hardware was available. LAr calorimeter pulses were digitized and recorded in order to study pulse shapes for BCID. In April / May 1993 [7-3] custom-built FADCs were used and cluster-finding was performed on the 36 channels using a module containing nine cluster-finding (CF) ASICs. This system was further enhanced in November 1993 [7-4], when BCID was performed on a single channel via a new BCID module. This system was used again in September 1994 [7-5] when data from prototype tile-calorimeter modules (RD34) were also taken. Most of the important results from the earlier years of test-beam running were summarized in a NIM paper [7-6]. A typical system set-up is illustrated in Figure 7-1.

In 1995, use of BCID was extended to all channels in a newly designed module using FPGAs, and this was used to study the performance of different algorithms. In the same year new purpose-built FADCs were used to produce the digitized data. Again the 36 channels were passed to the cluster-finding modules [7-7]. This was the end of the phase-1 demonstrator programme.

The phase-2 demonstrator was designed to closely emulate the final system, and in particular simulate the various data transmission and connectivity problems that will eventually be encountered. Figure 7-2 shows the full phase-2 system that was built by the end of the test-beam work. Completely new BCID modules and electromagnetic (e.m) cluster-finding modules had to be designed for this phase. The first tests of the new system were made in June 1996, but at that stage the most useful results came from investigations of analogue data transmission systems. More data were taken in September 1996 [7-8], when problems were discovered in the backplane technology used. This led to some redesign work before the October 1997 run and subsequent laboratory tests. These runs included several other modules to exercise new aspects

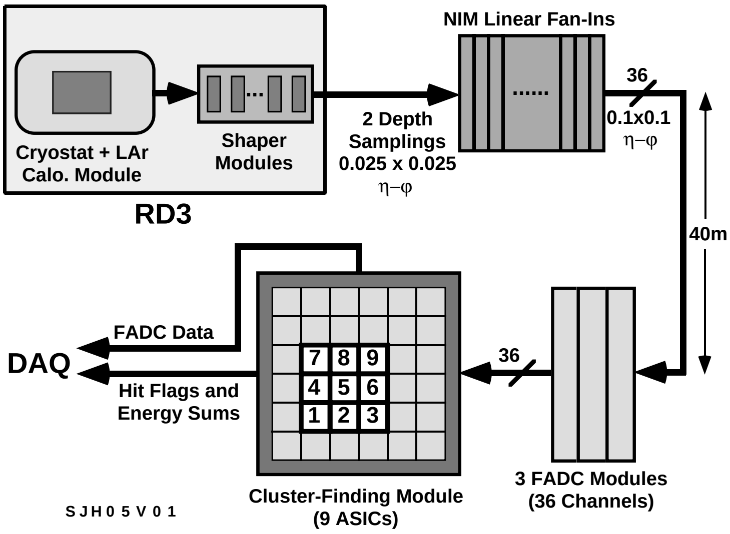

**Figure 7-1** Block diagram of the liquid-argon calorimeter and demonstrator trigger system in the test-beam runs of early 1993.

of the system, such as jet processing and integration with the Central Trigger Processor demonstrator.

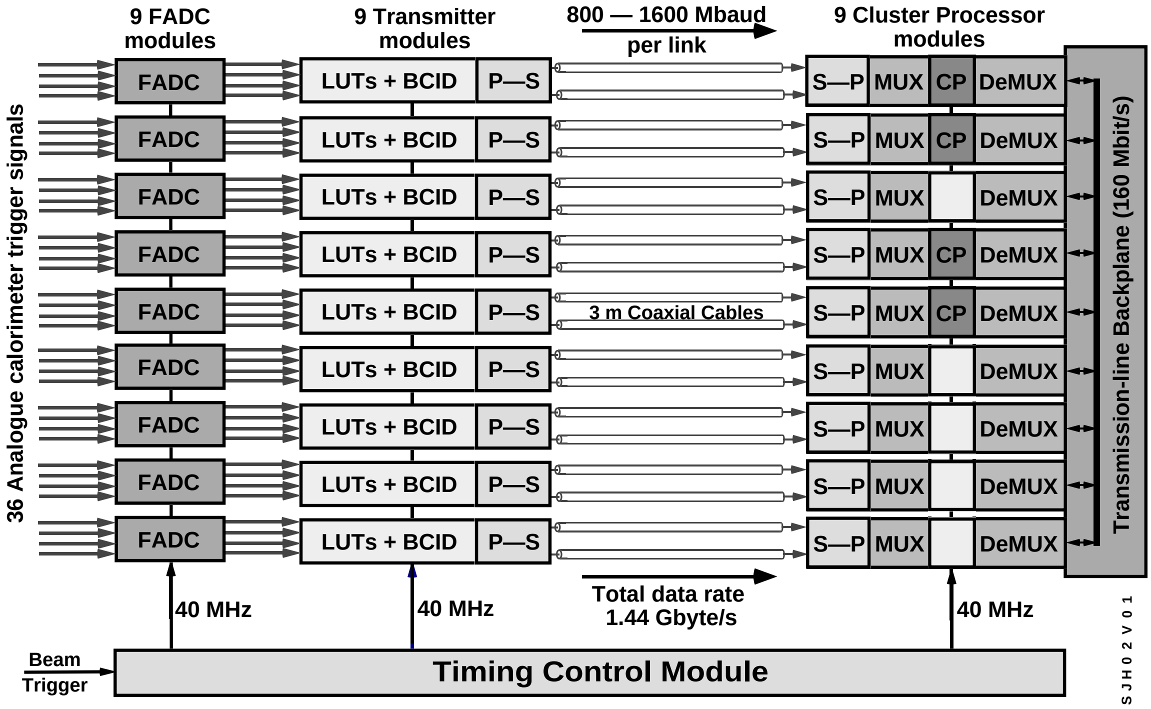

**Figure 7-2** Schematic diagram of the full 36-channel phase-2 demonstrator.

## 7.3 Pipelined trigger algorithms

The trigger algorithms described in Chapter 4 need to be applied to all the data at 40 MHz to generate trigger flags for each individual bunch-crossing. The two most intensive processing tasks are the e.m. cluster-finding algorithm and the jet algorithm. Both of these require information from many neighbouring trigger towers and combine the data in a relatively complex way to produce several triggers. It was necessary to show that this can be done in parallel on all channels reliably at 40 MHz with a small latency. This was done in pipelined logic with either custom-designed ASICs or FPGAs, as described below.

### 7.3.1 Cluster-finding ASIC

One of the first hardware tasks was to implement the e.m. cluster algorithm in a CF ASIC. A cut-down algorithm, using only 16 inputs (one layer of towers) and only two thresholds for each energy requirement, was implemented in ASIC RAL114 [7-9], and was ready for testing in the 1993 test-beam periods. At the time that the ASIC was designed, the LHC bunch-crossing time was foreseen as being 15 ns, so the ASIC was designed to work up to at least 67 MHz. In fact, it was shown to work correctly at speeds above 70 MHz, limited only by other on-board components. The latency of the process was six ticks for central cell energy summing, and seven ticks for the final trigger decision.

### 7.3.2 Phase-1 cluster module

The original test module for the CF ASICs was a 9U VME board which housed nine ASICs in order to fully process the nine $4 \times 4$ arrays contained within the full $6 \times 6$ trigger-tower environment of the early test-beam work. Data were brought into the module across the backplane, but with only one CF module, backplane traffic was not an issue for this stage of the demonstrator. In Figure 7-3a, the energy sum computed by the CF ASIC is plotted against the expected sum (using the LAr data) for a typical run. As can be seen, the two agree perfectly. Note the bunching of data around three typical values, corresponding to pedestal energy, pions, and electrons. Figure 7-3b shows the sharpness of the cluster-finding algorithm (with the isolation threshold switched off), as should be expected with a digital threshold.

Analysis concentrated on the correct performance of the ASICs, which operated at speeds in excess of 70 MHz [7-6]. So with the real LHC clock frequency of 40 MHz, and the progress in ASIC technology since the design of this ASIC in 1992, it should be possible to implement the full algorithm with a shorter tick latency than with this prototype ASIC by performing more complex operations at each step. The CF ASIC was also used in the phase-2 demonstrator, but the focus was by then on backplane issues, as the correct functioning of the ASIC had already been established. Its hit outputs were however used in the full-slice tests (see Section 7.6).

### 7.3.3 Jet processor module demonstrator

In 1997, work began at Stockholm on the jet trigger. This differs from the cluster-finding architecture in that it is proposed to use programmable logic chips rather than ASICs. The JPMD was designed to test FPGA implementation of the jet logic and to integrate with the existing demonstrator modules so that the full processing chain could be demonstrated as a complete ‘slice’ of the final system.

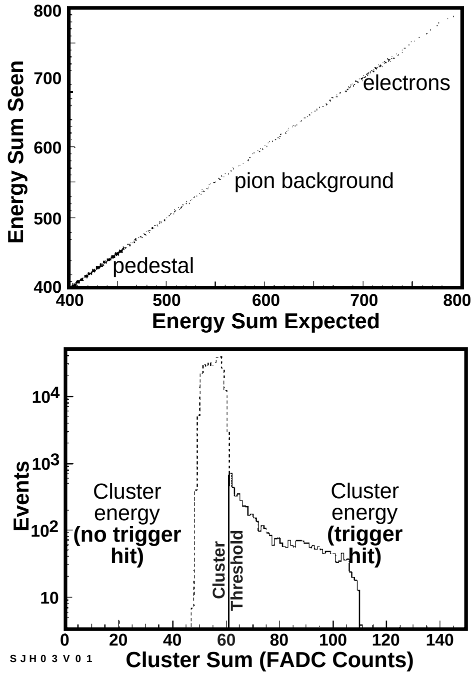

**Figure 7-3** Analysis of the cluster-finding ASIC performance: (a) observed vs. expected energy sum, (b) sum for events failing and passing threshold.

The JPMD was a 6U VME board which received 16 8-bit tower sums. It analysed these sums using a pipelined processing algorithm implemented on a commercial FPGA. The JPMD used a 411-pin Xilinx XC4000EX FPGA to process the 128 bits of input data and produced an 8-bit result to be sent to the CTP demonstrator (CTPD). In addition to the input and output data bits, eight address and eight data pins are also assigned to accommodate up to 256 internal registers for setting thresholds, configurations, etc. The processor FPGA could be configured by a download cable from the front panel, by an on-board EPROM, or remotely through the VME interface.

The baseline processor algorithm used at the test beam was implemented using four sets of adder trees, whose results were each compared with three thresholds which are individually programmable by the VME interface. The main elements of the design (adders and comparators) will also be the main elements of the final jet processor algorithm implementation. The input and output data lines of the JPMD were shared by 12 ns RAMs, which are used to capture and read out event information for diagnostic purposes. The RAMs could also be used to program and play back simulated data for testing various elements of the trigger.

The JPM demonstrator board was successfully integrated into the level-1 slice test during the 1997 ATLAS test-beam runs, and again at RAL in February 1998. JPM software for the DAQ, diagnostic, and offline analysis programs were written and merged into the level-1 software, and the full data chain from the CPMs to JPM to CTP was shown to work.

A more detailed study of the JPM functions, however, showed an unacceptably high bit-error rate (~$10^{-6}$) in the JPMD jet algorithm. Analysis of the data to pinpoint the source of the errors is still in progress, but some design issues have emerged which have influenced the planning of the next jet / energy-sum demonstrator:

- The board-level timing and control of the JPMD by an FPGA, which produced all read / write strobes from a single 40 MHz external clock. This gave inadequate control over the relative timing of different components on the board, and some devices (such as the jet processor FPGA) received clock signals with an asymmetric duty cycle.
- The jet processor FPGA (Xilinx 4036EX-4) was selected before the full programming and routing software was available for it. Timing analysis of the jet algorithm showed a maximum speed of 40 MHz, with no margin for error. This, combined with the timing issues above, may account for the error rates observed.

The jet / energy-sum ‘minus-2’ demonstrator, to be produced in June 1998, will resemble the current jet / energy-sum proposal much more closely than the JPMD, and is being designed with the above issues in mind. The clock distribution will be performed using discrete, high-speed components, and will feature independently-adjustable 40 MHz clocks which will be locally refreshed and multiplied with discrete PLL-based devices. Test configurations have already been written for the Altera devices selected to perform the jet algorithms, and simulations show that the 80 MHz algorithms should be able to run at over 100 MHz, giving a far better timing margin than that of the JPMD.

## 7.4 Bunch-crossing identification

Although the exact raw data format was not clear at the start of the programme, it was always foreseen that the level-1 trigger would have to perform bunch-crossing identification (BCID). Analogue techniques have been used at previous experiments, but it seemed a good idea to consider digital filtering as an alternative. Therefore the programme included the digitization of the prototype calorimeter signals and an investigation into implementing digital BCID. The digitized signals were useful for offline analysis and optimization of digital filtering techniques, and the BCID hardware was built to demonstrate the reliability of real-time BCID.

All of the demonstrator BCID implementations were based on an FIR filter combined with a simple peak-finder. This algorithm has been implemented in several modules over the course of the test-beam programme. It should be noted that in each case, the full matched filter has not been implemented — only a simplified version with a limited number of components, and with each component having a limited accuracy.

### 7.4.1 FADC module

In the early test-beam runs, various FADCs developed from previous experiments were used, but from 1995 onwards, new FADC modules custom-built by the Heidelberg group were used [7-10]. They ran at 40 MHz, with four channels per 6U module and generating 8-bit ECL outputs for each channel. They also contained a 256-byte memory for data capture, playback, and test-data injection. These have been used as the digital data source from 1995 to the present.

Much data was gathered with these modules over the course of three years, and we now have considerable confidence in their reliability. There was one genuine problem observed in 1995 and 1996 data where there would be a latency slip of one tick on about 1% of all events, but this problem was fixed in early 1997.

### 7.4.2 Single-channel BCID module

The first BCID board was built in Birmingham in 1993. It simply implemented the BCID algorithm in discrete hard-wired logic. Variable FIR components were implemented via lookup tables (LUTs) so that different filters could be used. Analysis documented in [7-4] shows that this initial attempt to implement BCID performed at or near 100% efficiency at the required LHC frequency.

### 7.4.3 Phase-1 BCID demonstrator module

The next stage was to build a full 36-channel system and implement the BCID logic in Xilinx FPGAs incorporated on a BCID demonstrator. Modules to do this were built at RAL, and a detailed description can be found in [7-7]. Each 9U board contained three Xilinx chips, each of which could process four data streams. Thus each board dealt with 12 channels, so that three modules were needed for the full system. The Xilinx logic could not cope with an infinite variety of filters, but several simple filters were implemented and tested. One useful feature of the Xilinx implementation was the capability to switch the Xilinx into a ‘transparent’ mode, where the filter and peak-finding were disabled. This allows data to flow unchanged through the system, making data comparison simpler.

Analysis of the test-beam results showed that the Xilinx implementation was, within limits of low statistics, 100% reliable. This meant that BCID algorithm performance and subsequent cluster-finding performance could be analysed from the test-beam data [7-7].

### 7.4.4 Phase-2 developments of the BCID module

The main phase-2 BCID module was the Transmitter Module (TXM), which did not develop the BCID implementation further, but concentrated instead on data transmission. This aspect will be described in more detail later. The internal BCID performance of the TXM in the 1996 and 1997 test-beam periods was entirely satisfactory. Within the limits of statistics available, there were no errors in clocking the data into the TXMs, or with the Xilinx and BCID performance. Results from these tests suggest bit-error rates at less than $10^{-8}$. Figure 7-4 shows the TXM in BCID mode correctly flagging (with the same latency) a particle which deposited energy in two adjacent trigger towers.

BCID implementation was developed further towards a final ATLAS solution first in a Heidelberg BCID test ASIC [7-11], then in the Heidelberg Front-end Module (FEM) [7-12]. This module encapsulated most of the functionality of the TXM, and added the extra feature of buffered data readout, which is closer to the readout mechanism in the final system. On this module, the BCID algorithm and data buffering were implemented in an ASIC, called the front-end ASIC [7-13], again designed by the Heidelberg group. The FEM was commissioned in 1997 and used in the October test-beam set-up. Results confirmed that the data transfer and BCID processing of the system was very reliable, with bit-error rates measured at less than $10^{-7}$ (limited by statistics). Some data were also recorded using the buffered readout mechanism. In this case verification of data integrity was difficult, since very few elements of the system had the capability to perform buffered readout, so there was little scope for cross-checking. However, some results were obtained by running with the Central Trigger Processor Demonstrator, as described in Section 7.6. Earlier laboratory tests in 1997 [7-14] showed that the fast readout mode had no bit errors in $10^9$ bits when tested at level-1 accept rates up to 128 kHz.

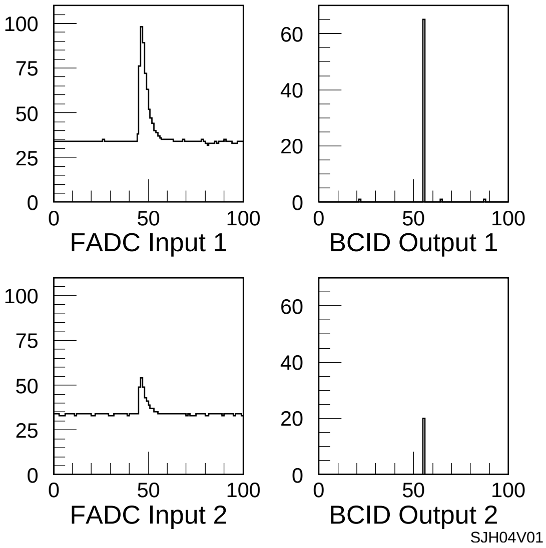

**Figure 7-4** Two TXMs performing BCID on the same particle, seen in two adjacent channels of the Tile Calorimeter.

Occasional errors started to occur at higher rates, but 128 kHz is already higher than the ATLAS specifications (75–100 kHz).

## 7.5 High-speed data transmission

The trigger processor must be split over many crates of electronics, meaning that fast digital data transmission between modules and crates is a potential problem area. For crate-to-crate transmission, several different digital options were considered, including both electrical and optical solutions [7-15]. In order to minimize input pins, serial links running at both 800 MBd and 1600 MBd were investigated at the test beam, the results confirming that the option of G-links running at these rates with copper cables is a safe baseline.

For the hardware that performs the trigger algorithms, the module-to-module interconnectivity within a crate becomes an issue. Both the cluster and jet-finding algorithms require massive fan-out of data between modules. For Cluster Processor crates, it is foreseen that this will be done using a fast backplane. In order to keep pin-counts down, the data will be fanned out serially using single-ended 160 Mbit/s bit-streams between boards. This will therefore require both a reliable multiplexing/demultiplexing process, and a dense backplane that will work with high data integrity at these speeds. One of the major tasks of the phase-2 demonstrator was to demonstrate that this is feasible. In fact, the current processor architecture means that the backplane will now be simpler than envisaged at the time of the demonstrator programme. In the final design, modules will only have to communicate with two nearest neighbours, whereas the demonstrator backplane had to transport signals between modules up to four slots apart. Therefore, proof that the demonstrator backplanes work means that the ATLAS trigger backplane should be easily achievable.

### 7.5.1 Inter-crate serial links

In the ATLAS system, a fast serial link will be needed from the Preprocessor Module to the Cluster Processor Module. To investigate various options for this link, a Transmitter Module (TXM) was designed at RAL. This implemented four-channel BCID on a single-slot 6U VME board. The data transmission aspect was separated off onto a replaceable single-slot daughter-board, allowing for different options to be tested. The original specifications can be found in Ref. [7-16]. The data were received on the phase-2 Cluster Processor Modules (CPM) described below.

The original daughter-board, used in 1996, used the HP G-link chip-set (HDMP-1014D) for data transmission. This could be run at either 800 MBd or1600 MBd, allowing the four channels of 8-bit data to be transmitted on two or one link(s) respectively. The link could be either a coaxial cable or an optical fibre via a commercial optical converter. The details of the implementation of the various options are shown in Figure 7-5.

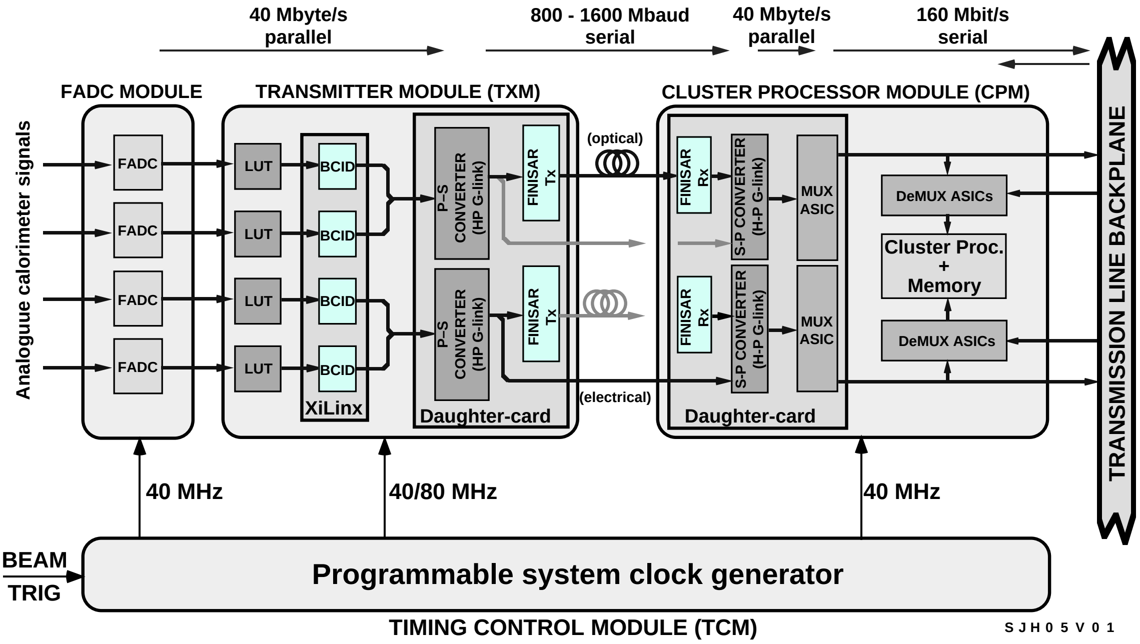

**Figure 7-5** Schematic diagram of the 1996 demonstrator.

Early tests in 1996 suggested that only optical transmission at 800 MBd performed well. All other modes had unacceptably large bit-error rates, or failed to work at all. However, as the problems of reliably driving G-links came to be understood, particularly the importance of reducing power and ground noise by good isolation, better results were obtained. Probably the best performance seen in the September 1996 test-beam run was no errors in about $10^{13}$ bits with optical transmission at 800 MBd. The experimental set-up in shown in Figure 7-6. Electrical transmission was best when no VME access was being made to the modules for the purposes of data readout. With event readout taking place, the best performance gave bit errors at one in $10^{10}$. After the test-beam period, the 1600 MBd mode was also made to work, and electrical isolation of the G-links gave confidence that there was no need for optical transmission. For more detail on results see Ref. [7-8].

Using insight gained in the 1996 run, new daughter-boards were designed for the 1997 run. These were built with the possibility of the two speed modes, but no optical link capability. This

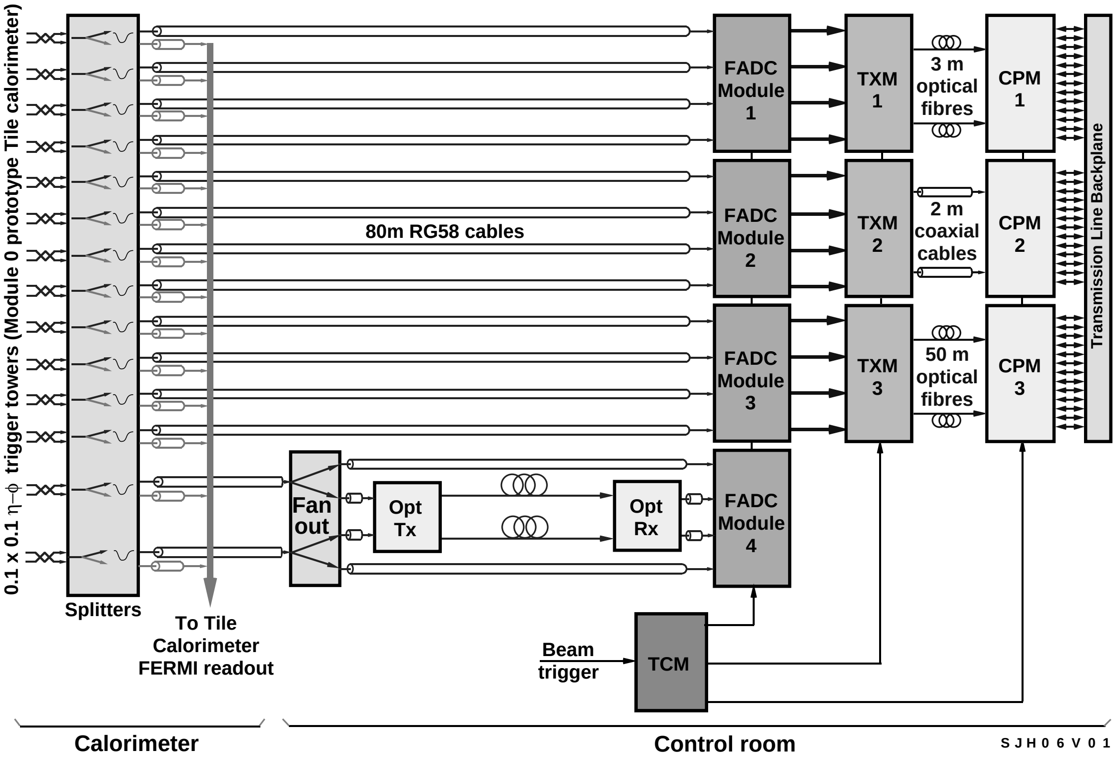

**Figure 7-6** Experimental set-up in 1996, showing the optical test and the reduced 12-channel demonstrator system.

system was seen to work very reliably in the 1997 test-beam run, with typical bit-error rates of less than $10^{-15}$ in the 800 MBd mode, as measured by the G-link internal error-checking. Some low-level problems were observed with the 1600 MBd mode, but again these were associated with events being read out. From the behaviour of the system with different cables, it was inferred that the choice of connectors and cables was also paramount to the reliable performance of G-links, so careful decisions will have to be made with the choice of connectors in the final system.

### 7.5.2 Cluster Processor fast backplane

The feasibility of the Cluster Processor crate backplane communication was a critical feature that needed to be demonstrated, to which end the backplane density in the phase-2 demonstrator was made to be approximately the same as that then envisaged for the final system. In order to achieve the level of backplane fan-out required without having an enormous number of pins, a single 40 MHz 8-bit signal was multiplexed into two 160 Mbit/s streams. These were transmitted as single-ended signals across the backplane. This is a factor of eight saving over differential 40 Mbit/s signals. However this format clearly requires more care and a custom-designed PCB backplane. Careful timing, signal isolation, and good grounding were needed to prevent high levels of data corruption and crosstalk. The full cluster-finding crate consisted of nine Cluster Processor Modules (CPMs) communicating via the backplane. Each module had to output data from a maximum of 21 trigger towers onto the backplane, and receive up to 21 trigger towers from the backplane. This adds up to 84 signal pins per CPM connecting with the custom backplane, plus several ground and power lines.

Another aspect of the backplane problem was how to perform the required multiplexing and demultiplexing. Again, custom-built components in the final system would be needed for this task. To demonstrate this operation, another ASIC was designed, known as the dual-function ASIC RAL163 [7-17]. As the name suggests, two of these could be used as a pair to perform both the 40 MHz to 160 MHz multiplexing, and also the conversion back down to 40 MHz. It also contained internal memory for test purposes and timing self-calibration to make the bit alignment easy to handle. These ASICs had been tested stand-alone in 1995, and were ready for integration with the CPMs and backplane in 1996.

### 7.5.3 Phase-2 Cluster Processor Module

The first CPM boards were two-slot 9U VME modules designed by Birmingham and RAL in 1995/1996 [7-18]. They had to receive data from the phase-2 TXMs so, in an equivalent way to the TXMs, they used daughter-boards to receive front-panel signals to allow testing of several protocol options. The signals received via the front panel, called the ‘core cells’, were first multiplexed up to 160 Mbit/s to be passed onto the motherboard, which then fanned them out onto the backplane and also into the core-cell CF ASICs. The motherboard also received fanned-out signal data from other CPMs via the backplane, which were also sent to the CF ASICs. Only four channels were received directly onto the CPMs, the others having to come from the backplane. In all 25 channels were available on each CPM, meaning that four full versions of the cluster-finding algorithm could be performed. Each CPM had four CF ASICs, and the whole system fully covered the $6 \times 6$ tower matrix. The layout of the channels meant that the maximum distance travelled by a signal over the backplane was four double-width VME slots. Trigger-flag output bits and the four core-cell trigger-tower data bytes were available on the front panel.

Three of these CPMs were tested in the test-beam run in September 1996. The set-up has already been shown in Figure 7-6. After stable G-link operation was established, the performance of the multiplexing and the backplane was investigated. Deficiencies in the timing set-up led to less than perfect performance in these first CPMs. The best-optimized performance seen at the 1996 test-beam for an individual trigger tower was no errors in about $10^8$ bits [7-8], ignoring occasional latency slips due to the timing set-up. This relatively high data integrity was only observed on the trigger towers that were entirely internal to a CPM, confirming that the dual-function ASICs were performing well. A typical event, showing the flow of data through the whole system from FADC to CPM, is shown in Figure 7-7a.

The backplane, which used BTL signal protocol, was far more of a problem in the 1996 test-beam run. The best error rates observed were at the level of 5%, which was completely unacceptable. Despite this, many of the calorimeter pulses appeared quite clean across the backplane, since only errors in the higher-order bits have an obvious effect on the pulse shape, as is shown in Figure 7-7b. The signals themselves suffered bad reflections and crosstalk, and in general the BTL logic was found to have signal rise and fall times that were too long for 160 Mbit/s bit-streams arriving from multiple sources.

Because of the failure of the BTL version of the backplane logic to reach the required specifications, a second version of the CPM was designed in 1997. ECL logic was used for the backplane protocol, and timing improvements were included on the board. The other major difference was the capability to use an MCM on the CPM daughter-board to receive the serial signals from the TXM. This would form our first demonstration of the use of MCM technology (Section 8.4.4).

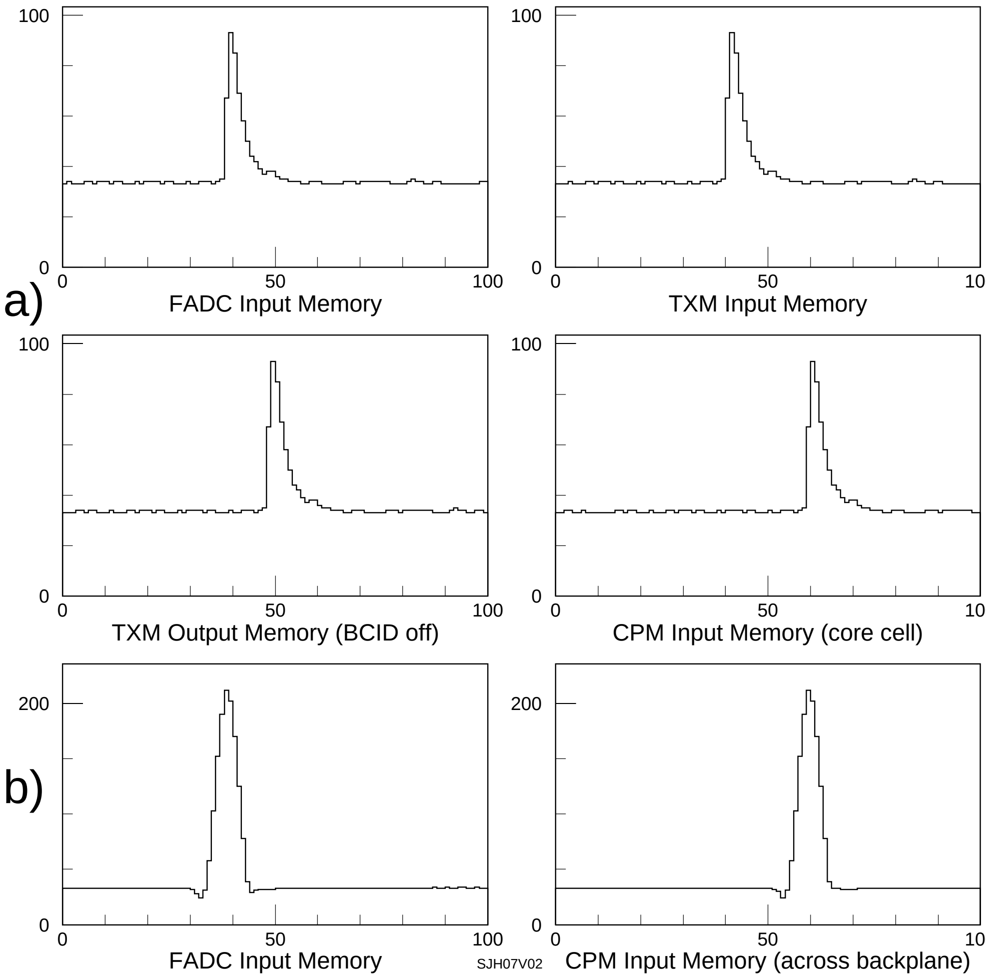

**Figure 7-7** Data flow through the complete system. (a) Tile Calorimeter pulse flowing from FADC to TXM input, through the BCID (switched off) to TXM output, and finally to CPM — note the latency at each stage. (b) a test pulse seen across the backplane.

A full system of nine TXMs and nine of the new CPMs was tested at CERN in October 1997. This meant that the full chain of processing could be tested for 36 channels from FADC through to CPM output, and measurements made of errors between each stage. Also the full latency of the demonstrator system could be directly measured. A typical pulse flowing with no errors onto both the on-board CPM memories, and across the backplane, can be seen in Figure 7-8.

All parts of the chain up to the CPM worked perfectly, as far as could be seen from the data recorded. Error rates in the whole of the chain from FADC through BCID in the TXMs and G-link data transmission at 800 MBd worked with bit-error rates of less than $10^{-8}$. However, low-level problems were observed with the CPM system. For the ‘core cells’, i.e. those not going across the backplane, bit errors were observed at a rate of about $3 \times 10^{-5}$ averaged over all channels. Also, occasional latency slips were observed (at a rate of about one in 2,000 events). It was concluded that these errors were due to control problems rather than difficulty with the multiplexing process, and the rate was very board-dependent. There were also problems in timing the backplane data since the ECL parts were slightly slower than expected. This meant that the timing for some data coming across the backplane was difficult or impossible to optimize. This, and other minor problems, caused the bit-error rate on backplane data to be about $5 \times 10^{-5}$.

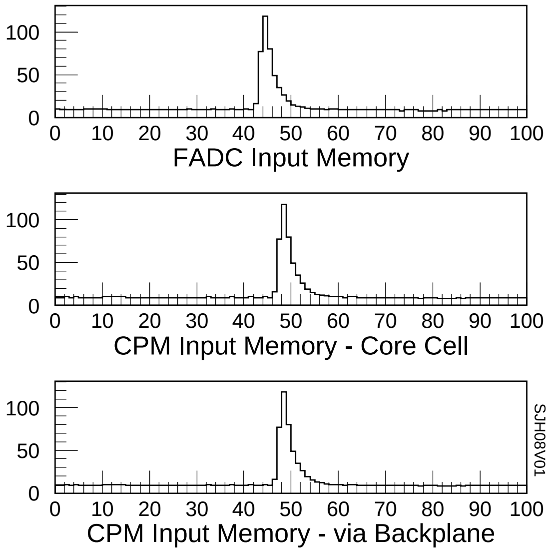

**Figure 7-8** Tile Calorimeter data from one FADC flowing perfectly onto two CPM boards, one directly and one via the backplane.

Since there were clearly curable problems in the test beam, it was decided to reconstruct the whole system in the laboratory for final tests. This has the advantage that pseudo-random or particularly difficult data patterns could be used to exercise the system far more stringently than test-beam data. The system was put together in February 1998 after several minor faults had been discovered and fixed in the CPMs. Running the system in the usual way suggested a very high level of data integrity.

Analysing events taken by the DAQ system requires huge numbers of events to prove that the system is working to the level demanded for the final ATLAS system, i.e. bit-error rates less than about $10^{-10}$. In order to bypass that problem, an alternative mechanism was devised using a hardware module to perform the necessary comparisons in real time. This custom-built module compared data from the input and output of the system and produced an error count that could simply be scaled. The system is illustrated in Figure 7-9. The results obtained via the DAQ system could be duplicated using this module in a matter of seconds, with no extra analysis required. Data passing through the whole system were measured to have bit error rates less than $10^{-13}$, showing that each stage of the processing was performing to this level. Data going across the backplane were measured to have equivalent error rates showing that negligible data corruption and crosstalk was taking place. A full series of tests will be performed at RAL to give confidence in the choices of technology made for the trigger system.

## 7.6 Full-slice test

An additional important aspect of the demonstrator programme was to show that once all the individual components were working, they could actually all be put together and work in a stable fashion as a genuine trigger. That is to say that using a common clock (equivalent to the final LHC clock), the entire system would work properly to produce a reliable and correct trigger with a constant latency. The test-beam environment, and working with the CTPD, was a vital aspect to showing that the final system acted as a working trigger.

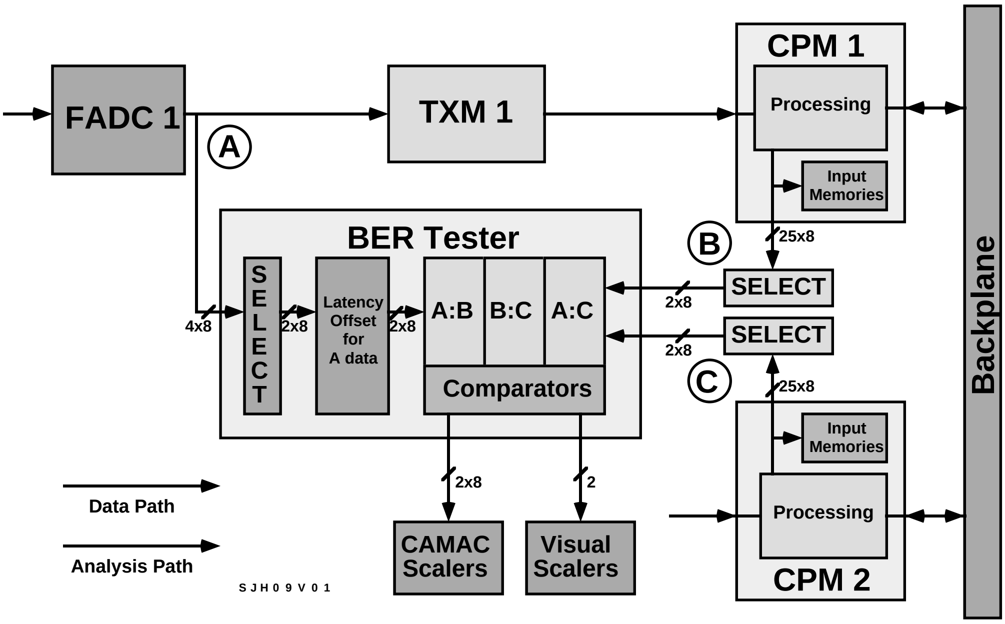

**Figure 7-9** Block diagram of the bit-error rate tester and experimental setup.

### 7.6.1 Timing calibration and stability

The timing control in ATLAS will be provided by the TTC system. This will provide a very stable clock locked to the phase of the LHC bunch-crossings. However, this was not available for use at the test beam, so a module was designed and built at RAL to provide all the necessary timing and synchronization signals. This was the Timing Control Module (TCM) [7-19] used in the phase-2 demonstrator.

The TCM provided fan-out for all the 40 MHz clocks needed by each module in the system, and also generated the stop signals and VME interrupts for event readout purposes. It could be programmed to accept different triggers and also has the flexibility to either use an external clock or generate its own clock. The phase of clocks and stop signals to each of the different types of modules could be programmed, which was necessary to ensure that data were being strobed at the optimum time throughout the system. The success of the TCM is seen by the integrity and latency stability of the data in all parts of the full system.

Along with the programmable delays on the TCM, several other points in the data stream required timing delays. For example, to allow for the possibility of different length cables between crates, the data transmitted from the TXMs could be delayed by more than 1 µs in steps of 1 ns. Delays of this sort are generally quantized in units of 25 ns, corresponding to the time between LHC bunch-crossings. Clearly, the 1 ns steps below 25 ns are needed to align the data to be strobed correctly into the next stage, whereas the ‘tick’ delays (in steps of 25 ns) are needed to equalize the latency of the data.

With the TCM delays and all the individual module delays, the full phase-2 demonstrator had about 100 parameters to calibrate. In principle, many of these delays could be set automatically via software, but in fact most were done by hand as experience was gained with how to control the timing. In designing the system, the policy with delays was to err on the side of caution and provide a delay mechanism in most conceivable places. Much was learnt about which delays were necessary, and how much delays were likely to vary between data streams. Generally, it was encouraging that once a stable timing regime was found for one data chain, it was usually easy to time the other modules in with the same, or very similar, parameters. This led to a full system with data transfer stable over the full 36 channels for several thousand events. As for the latency stability of individual channels, the bit-error rate measurements given above would have been dominated by latency errors had they been present, so the conclusion is that we could, by use of a few timing parameters, tune the system to be completely stable.

### 7.6.2 Full-slice tests with FEM and CTPD

The availability of the CTPD module [7-20] during the beam tests allowed the assembly and operation for the first time of a complete slice through the calorimeter trigger. The organization of modules for this is illustrated in Figure 7-10. Data digitized in the 36 FADC modules were passed via TXMs and FEMs to the cluster-processing system, and 16 core-cell energy values were passed to the JPMD. The nine hit outputs from the cluster-processing system and the JPMD were sent to the CTPD, where a trigger decision was taken and a level-1 accept (L1A) signal was formed. Since this was the first SPS run following the SPS fire, beam availability was below average, so the complete slice tests were made using simulated calorimeter pulses derived from a pulse generator. The system was operated in three modes:

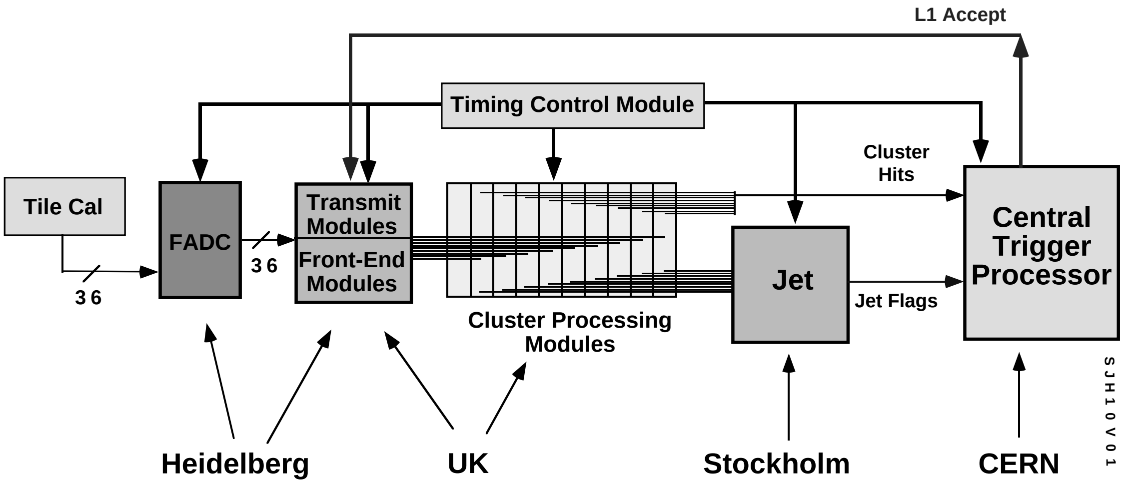

**Figure 7-10** Schematic diagram of the complete slice tests using JPMD and CTPD.

- An output from the pulse generator was used as an 'event' input to the TCM, stopping the scrolling memories and initiating computer readout of all modules after each analogue pulse was applied to the FADC input. This allowed checks that the data passed correctly through the system, that the correct data were sampled in the FEM pipeline, and that an L1A was generated correctly for each analogue input pulse. The timing relationship between the different parts of the system is shown in Figure 7-11a.
- The system was modified so that the L1A signal provided the 'event' signal. In this mode, the system operated as a true trigger for the first time, so that, for example, all readout activity could be suspended by lowering the input analogue pulse height. At this stage, the frequency of L1A was restricted to the maximum rate at which the online computer could record data from all modules. The altered timing relationships are shown in Figure 7-11b.

- The L1A signal was used to initiate the fast pipeline readout in the FEM. Running in this mode, data from the eight time-slices surrounding the L1A were captured in an FEM double buffer without stopping the trigger pipeline. A signal indicating FEM buffer full was used to initiate readout of the FEM only, again without stopping the trigger pipeline. Running in this mode, the frequency of L1A could be raised to many kHz, finally being limited by the ability of the computer to empty the FEM buffer. The contents of the buffer memory are illustrated in Figure 7-12. The analogue input pulse produced a digitized FADC output after pedestal subtraction of around 60 counts. However, since the analogue signal was not phase-locked to the 40 MHz system clock, successive pulses might be digitized at the peak or on rising or falling edges. After BCID, this led to digitizations ranging from 55 to 70 counts. It may be seen that a regular sequence of one in eight time slices is occupied with a value in the expected range, other time-slices having been set to zero by the BCID [7-14].

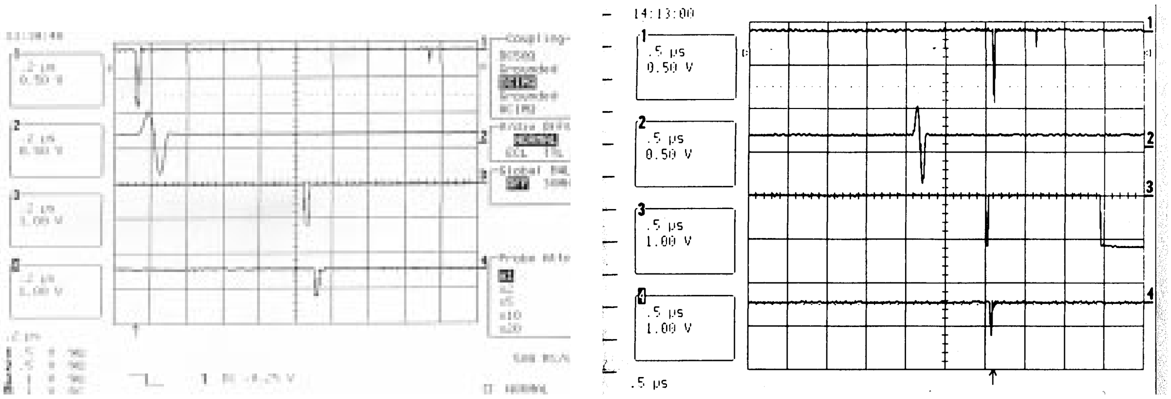

**Figure 7-11** Captured oscilloscope traces showing the progress of a trigger pulse through the system. (a) using external trigger, (b) using self-generated trigger. From top to bottom, the traces are: system trigger, generated pulse, CPM hit output, and level-1 accept from CTPD.

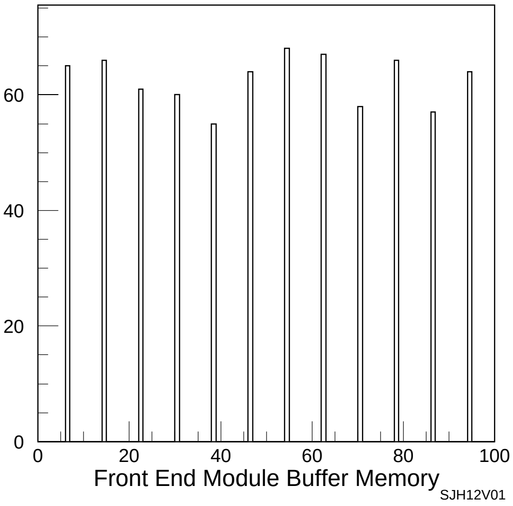

**Figure 7-12** BCID processed data captured in FEM memory using level-1 accept to trigger data buffering.

## 7.7 Conclusions from the test-beam programme

The group has performed a sequence of successful test-beam runs using signals from prototype ATLAS calorimeters. The initially-simple system has evolved into one of considerable complexity which contains most of the essential elements of the proposed final trigger system. During this time many options have been studied and decisions regarding preferred solutions have been made. The final options, as built into the full phase-2 demonstrator, were tested to a high level, and found to perform in a satisfactory manner. There have been some difficulties encountered, as was only to be expected when building such a large integrated system, but these have been overcome in the final system. The experience gained by this programme has led to the current proposal for the level-1 calorimeter trigger, and there can be considerable confidence in the feasibility of each part of this design.

## 7.8 Computing

The main focus of our demonstrator programme has been to investigate the technologies, such as ASICs, MCMs, fast backplanes and serial links, which are essential components of the trigger. This has also involved developing software, some of which can be considered as a prototyping and testing exercise, particularly in regard to object oriented (OO) techniques. In this section we briefly review the software from our demonstrator programme and summarize what relevant experience has been gained for the design and implementation of software for the final trigger system.

### 7.8.1 Data acquisition and online monitoring

The DAQ software used in the laboratory and beam tests of the demonstrator system evolved from the CERN Spider system. It has been ported from OS9 to LynxOS and considerably upgraded. In addition to the usual readout and data logging, the DAQ software provides simple histogramming and limited online analysis of the data. While this was essential for our development programme, any further DAQ work should be done in the context of the ATLAS DAQ framework. Discussion will be needed to determine how the ATLAS DAQ should be used for laboratory tests outside CERN on our installed processor base.

Our existing DAQ system provides fast VME readout but has limited computing speed and memory. It allows us to monitor histograms of pulse shapes and the comparisons of pulses from different stages in the processor chain. It also tracks a variety of errors in the readout and the transmission of data across the system. We will still require this kind of low-level monitoring in the final system, but far more sophisticated checking of the performance of the trigger will also be required. Some of the analysis we currently only perform offline should be moved online. Better graphical display of status and errors will also be desirable. The higher level online monitoring will have to be written afresh as a suitable component of the ATLAS DAQ system.

### 7.8.2 Diagnostics and test package

For the most recent phase of our demonstrator programme, we invested considerable effort in providing a user-friendly diagnostics package to allow physicists and engineers an easy way to manipulate and test the various modules we built. The package presents a graphical view of the whole configuration, and of the registers and memories on each board and ASIC. The behaviour of modules and their individual registers can be verified in detail. The software has proven to be very useful, and we will certainly require something similar for our future prototyping and testing phase.

The diagnostic software was written mostly in C++, with the front-end user interface written in the Tcl/Tk scripting language which provides a Motif-like ‘look and feel’. The software has been described in an ATLAS note [7-21] and more detailed documentation is also available.

In the light of our experience, some redesign of the diagnostics software will probably be desirable. In particular the use of Tcl/Tk to provide the user interface part should be reconsidered as it has become rather cumbersome to maintain the scripts (which are not OO in style). We have also experienced problems (due to limited memory size) in running a large X-based application on ‘front-end’ processors such as the Motorola MVME167 under LynxOS. Alternative architectures, where the user interface and hardware access are split, will also need to be considered for the future. Thirdly, the existing software provides a detailed model of the behaviour of the hardware. This has been very useful for the debugging of individual cards, but adds a large overhead in memory when a complete configuration consisting of many modules is being used. Some way of dynamically enabling and disabling the detailed model would be useful. Finally, the diagnostic software was written assuming a unified design philosophy for the modules. However, one prototype module contained an internal ‘firmware’ layer which does not fit naturally into the existing software design.

The diagnostics software was well developed for interactive diagnosis and configuration of individual modules. The architecture also provided for testing complete chains of modules, and there were also facilities for more automated testing of the complete functionality of boards or the links between them. However these aspects have not yet been fully exploited. In the final system, much more emphasis on automated tests will be required.

### 7.8.3 Calibration

During the demonstrator programme we have developed a number of small stand-alone calibration programs. These have addressed issues such as the tuning of positive ECL (PECL) voltage thresholds and backplane timing between CPMs. However, in practice most of these calibrations have been performed by hand. This proved to be easier in the laboratory during debugging and also in the test beam environment. This will not be feasible in the final system, so much more work on calibration software will be required.

### 7.8.4 Test-beam data analysis

We have written a number of programs to analyse the data we have recorded in successive beam tests. The latest of these is fully object-oriented and is written in C++. This program unpacks the data read out from each module and calculates the data transformation from input data to expected output performed by each module in a processor chain. It can then compare the actual and expected data at each stage to verify the correct operation of the boards and their various cable and backplane connections. The simulation of each module is simpler than the detailed model implemented in the diagnostic package, but allows for a quick analysis of error sources. This functionality is a large part of what will be required for the online monitoring of the DAQ data from the trigger in ATLAS.

## 7.9 References

7-1 *A demonstrator programme for the ATLAS level-1 calorimeter trigger,* RD27 note 35, ATLAS note DAQ–NO–031, 1995.

7-2 R. E. Carney et al., *The level-1 bit-parallel electromagnetic trigger processor — CERN beam tests, November 1992,* RD27 note 12, 1993.

7-3 U. Dickow, *An analysis of the RD27 level-1 prototype calorimeter trigger processor test beam data — April/May 1993,* RD27 note 23, 1994.

7-4 I. P. Brawn et al., *Beam tests of a single-channel bunch-crossing identification module for the level-1 trigger — November 1993,* RD27 note 31, 1994.

7-5 I. P. Brawn et al., *Beam tests of the bit-parallel trigger processor with the tile hadronic and liquid argon electro-magnetic calorimeters — September 1994,* Internal note, July 1995.

7-6 I. P. Brawn et al., *Beam tests of a prototype level-1 calorimeter trigger processor for LHC experiments,* Nucl. Instrum. Methods A349 (1994) 356–366.

7-7 I. P. Brawn, *Bunch-crossing identification for the ATLAS level-1 calorimeter trigger,* Ph.D. Thesis, University of Birmingham, 1996.

7-8 S. J. Hillier, *Summary of September 1996 test-beam,* Internal note, October 1996.

7-9 V. Perera, *Data on the LHC electromagnetic cluster-finding ASIC RAL114,* RAL hardware write-up, February 1992.

7-10 P. Hanke et al., *A flash analog-to-digital converter for trigger tests with the ATLAS calorimetry,* RD27 note 41, 1995.

7-11 A. Mass et al., *Design and test of an ASIC for bunch-crossing identification in the ATLAS level-1 Trigger.* CERN / LHCC / 95–36, First Workshop on Electronics for LHC Experiments, Lisbon, 1995.

7-12 C. Schumacher, Diplomarbeit Universität Heidelberg: *Der ATLAS Level-1 Trigger, Auslese des Front-End,* IHEP 97–05, 1997.
http://www.ihep.uni-heidelberg.de/Veroeffentlichungen/ps/schumacher.diplom.ps.gz

7-13 A. Mass et al., *A front end digitization and readout system for the ATLAS level-1 calorimeter trigger.* CERN / LHCC / 96–39, Second Workshop on Electronics for LHC Experiments, Balatonfüred, Hungary, September 1997.
http://wwwasic.ihep.uni-heidelberg.de/atlas/PAPERS/lerb-96.ps.gz

7-14 U. Pfeiffer et al., *Performance of the front-end demonstrator system of the ATLAS level-1 calorimeter trigger,* CERN / LHCC / 97–60, Third Workshop on Electronics for LHC Experiments, London, UK, September, 1997.
http://wwwasic.ihep.uni-heidelberg.de/atlas/PAPERS/LEB97.ps.gz

7-15 U. Pfeiffer, Diplomarbeit Universität Heidelberg: *Analoge optische Signalübertragung vom Kalorimeter zum ATLAS Level-1-Trigger.* IHEP 96–16, 1996.
http://www.ihep.uni-heidelberg.de/Veroeffentlichungen/ps/pfeiffer.diplom.ps.gz

7-16 G. C. Lewis, *ATLAS transmitter module demonstrator phase-2,* RAL hardware write-up, January 1996.

7-17 J. Edwards and V. Perera, *User guide for the first level calorimeter trigger processor demonstrator dual function ASIC RAL163,* Hardware write-up, 1994.

7-18 R. Staley, *Cluster processing module — user guide,* Hardware write-up, University of Birmingham, 1996.

7-19 B. Hatley and J. Leake, *Programming model for the trigger timing control module used in the phase-2 ATLAS bit-parallel calorimeter trigger demonstrator*, RAL hardware write-up, 1996.

7-20 G. Schuler, *Central Trigger Processor*, RD27 working document, August 1996.

7-21 J. Leake et al., *ATLAS level-1 calorimeter trigger diagnostic software*. ATLAS note DAQ–NO–72, August 1997.