This document provides a description of the features of the Event Builder data acquisition sub-system. Conceptually, this sub-system can be thought of as having an input section, a transformation section and an output section.

Input will consist of blocks of event data from each of the acquisition readout controllers.

The transformation section is to some extent optional and consists of one or more cpus processing the incoming event data stream. Transformations include event formatting, gain adjustment,etc.

The output will be blocks of complete events in the agreed event format (see elsewhere) transferred by a network interface to the online Sorter. It is assumed that most data transformations are to be applied in the Sorter. The only criteria for processing data in the Event Builder is assumed to be for requirements more easily accomplished prior to event formatting, or where significant processing or data reduction of a fixed nature can be performed.

Requirements

The Project Scientists Committee has asked for the system to be able to cope with a minimum of 200 KWords/second, with feedback required on the possibility of upgrading to 2 MWords/second. For the purposes of the Event Builder, it is assumed that these defined rates apply to output and not input rates.

Other requirements that have been assumed include ease of programming and production phase maintainability. All components suggested are readily available from stable and internationally known companies.

Background

The Event Builder lies between the data acquisition hardware and the Sorting and Storage components of the whole system. It has to accept event by event data from the Readout Controllers at high rate and with little or no deadtime. It has a primary task of re-organising the format of events ready for passing forward to the Sorting and Storage components. Other transformations could be applied to the data stream at this stage and so sufficient processing power needs to be provided.

The hardware group have agreed to organise event readout from each data acquisition crate sequentially. This results in one data stream arriving at the Event Builder crate. This data stream will consist of sub-event contributions from each acquisition crate. All sub-event sections of one event will therefore arrive together and separated from those of other events.

The connection between the Readout Controllers and the Event Builder crate will be a 32bit FERA-like link with Strobe, Acknowledge, Busy and Readout Inhibit signals. This definition is compatible with the CES HSM8170 fast memory board with the 32bit input option.

A further requirement is that the link from the Readout Controllers to the Event Builder be as short as possible. This forces the Event Builder crate to be positioned close to the acquisition hardware. In handshake mode, any increase of the cable length will reduce the overall bandwidth. For the system envisaged this should not be a problem.

The final requirement relates to the connection between the Event Builder and the Sorting and Storage components. The latter will be placed in the Control room some 100 metres from the acquisition hardware. At Daresbury Laboratory this would mean connecting two different electrical grounds. An optical solution would surmount this difficulty.

Components

It is proposed that the Event Builder consist of boards contained in one VME crate. VME is the most sensible choice for this application as it supports the necessary processing power and data transfer bandwidth. For use at Daresbury Laboratory, the crate will conform to their standard specification.

The crate will support an Ethernet link to the system Control and Setup network. This will be provided by the Motorola 147 cpu board which has an onboard Ethernet interface. See elsewhere for details of the network interface.

The interface with the acquisition Readout Controllers will be the CES HSM8170 fast memory module. This module has a 10MHz 32bit FERA-like input port to fast static RAM. The module also has a VSB interface.

In order to ensure a correct understanding of the interface, the hardware aspects of the link betwen the ROCs and the HSM 8170 will be the subject of a document by the electronics subgroup.

The original proposal for the connection between the Event Builder crate and the Sorting and Storage components was Ethernet. This is now discarded as not having sufficient bandwidth. It is now proposed to use a fibre optic point-to-point link. There are two versions readily available in the catalogues of Struck and CES. Both have maximum quoted rates of 6 Mbytes/second, and both have VSB interfaces. On first sight either product would be satisfactory and some further research will be necessary to make a choice.

The cpus involved in the event builder will be standard VME cpu modules utilising MC68020/30/40 microprocessors. The Motorola 165 (MC68040 at around 15-20 mips) seems a good choice as a general processing engine with sufficient performance. It has been designed specifically for embedded controller functions and is provided with 4 Mbytes of memory ported to both VMEbus and VSBbus. This board is orderable now with production availability in Q3 1990. The 68040 has a quasi-risc design which provides high processing power whilst keeping the powerful instruction set of the 68k series.

Data Rates

Tests have been carried out to quantify the performance of the chosen components. As the MVME165 is not yet available a MVME147 was used for the first test. A DMA transfer initiated by the FIC8230 from HSM8170 memory via VSB to MVME147 dual-ported memory via VME occupied about 50% of VME bandwidth at 1 MWords/second rate. A similar test transferring between two HSM8170 via VSB achieved a rate of nearly 1.5 MWords/second.

The intended approach would be to setup DMA transfers between HSM8170 and MVME165 via VSB. This is estimated to be nearer the case of the first test as the MVME165 dual ported memory would presumably be similar to that of the MVME147. There may be some improvement due to better designed VME and VSB interface chips reportedly used in the MVME165.

A conservative approach, therefore, would say that a HSM8170 and FIC8230 combination would cope with 1 MWord/second throughput.

A comparison with the existing Sort Engine can serve as a guide to the power required for event-reformatting. The current task of event unpacking is thought to be equivalent to the required task of event formatting. A 16 MHz 68020 cpu currently performs event unpacking at about 200 KWords/second. A 25MHz 68040 is thought to be approximately 6 times more powerful, which would indicate that one cpu may be able to re-format events at 1MWord/secomd. Some reduction would be expected due to memory bandwidth contention during data block transfers. Any further processing would require additional processing power. Further processing will include reducing the data volume by calculating the total energy collected in the event, possibly gain matching, filtering on TDCs etc. However, some of these transformations could be applied in the Sorter.

Modus Operandi

One HSM/FIC pair of boards has been shown to be capable of an input rate to the Event Builder of 1 MWord/second. To achieve this data rate then VMEbus and VSBbus will both have to be used. Two data transfers are necessary per block of event data. The first involves a transfer from the HSM8170 memory to dual-ported memory of a processor. The second transfer is from the dual-ported memory of the processor to the output interface controller. The proposal is to use VSB for the first and VME for the second transfer.

The maximum number of slots on a VSBbus is 6, which can be filled by one HSM8170, one FIC8230 and upto four processing cpus. The first stage would be to use one VSB backplane. This would support 1MWord/second input and 1.5 MWord/second output to the Sorter.

Adding a second VSB backplane would allow 2 MWords/second to be processed. This would have to be reduced to 1.5 MWords/second on output. The two HSM8170 boards would then be used in flip-flop mode.

There would be space in the VME crate for a third VSB backplane if necessary. Three HSM8170 connected together in cascade would work in principle.

The amount of processing power required for Event Building depends on two parameters - the data volume and the nature of the processing involved. A maximum of 4 cpus processing data at 1MWord/second would allow a reasonable amount of computation to be accomplished.

Data is strobed into fast memory in the HSM8170 in an automatically incrementing mode. The base address and count are settable. The memory would accumulate events until a programmed count point had been reached (say 16 kbytes). This would signal an interrupt to the controller cpu. This interrupt does not inhibit readout until the BUSY input is cleared. Event data would still arrive until all data for that particular event had been transferred.

An *event-readout-in-progress* signal would need to be supplied and connected to the BUSY input. This signal would be derived from the REN/PASS signals of the FERA-like readout. The cpu would then activate the *readout-inhibit* signal while it changed the address pointers to a free area of memory and reset the counter to re-activate the readin. With buffered Readout Controllers, this will not contribute any deadtime to event collection since it should be accomplishable very quickly.

Upon re-activating readin, the already collected buffer will be DMAed to a processor for event building.

Software

A task running in the network interface cpu will accept control commands from the user and act on them as necessary. These will involve program halt/go/reset/flush and download commands. For a definition of the network interface software and the available command set, see other documents.

The software running in the FIC8230 controls the HSM8170 and initiates DMA transfers to one of several processors. This software is not very complicated, but has to operate as quickly as possible when a memory block in the HSM8170 has filled. For these reasons it is proposed not to use an operating system but to code a simple task in C and assembler as necessary.

There will also be code running in the controller for the optical fibre link. This will be an independent subsystem, accepting requests for data block transfers and initiating block transfers down the link. It will receive acknowledgements from the link, and send acknowledgements to the sending cpu.

The processing cpus for the Event Builder and the Sorter are proposed to be the same. There are advantages to this approach for the software. Both will execute a single task, and the foundation software to enable this should be identical. In principle this would allow either a version of the sort software or purpose written code to be executed.

The transformations to be applied to the input data stream include event re-formatting, total energy calculation, possibly gain matching, TDC filtering and some Compton suppression related tasks. The nature of these transformations do not fit neatly onto the general sort language, and would be better accomplished by purpose written code. One must always remember that some transformations may be better left to the Sorter.

Hence it is proposed to code the necessary transformations in C as a single task. It is proposed that some options be allowed. If the GNU C compiler on the SUN were used then it would be possible create a code module that could be downloaded onto 68k series microprocessors. The same code could be compiled to run on the SUN in a test form. It is proposed to download the code as opposed to being fixed as it may change from time to time, and maybe from experiment to experiment for particular requirements.

Limitations

It should be noted that a single crate VME-based Event Builder would saturate at around 1.5MWords/second output rate using one fibre optic interface.

The input saturation rate will be somewhat higher. Serial readout on a single cable into one HSM8170 will saturate at 3MWords/second for a 20 metre cable assuming signal handshaking is used. If strobing were used instead of handshaking then a similar limit will still apply, due to the number of HSM8170 modules and associated builder cpus that would fit into a crate.

To increase the input rate above these limits, then it would probably be necessary to revert to parallel sub-event readout, in which case the event merging would have to be accomplished in purpose built hardware. There would still be one input stream to the Event Builder crate, but at a rate that would be too high to disperse to builder cpus. Therefore other crate technologies would be necessary, and we may have to wait for something like Futurebus.

It can be seen that any increase required over this rate will incur significant penalties in cost, development time and availability.