CONF-810523--17

A PROPOSED DESIGN FOR A FAST (PARALLEL) PREPROCESSOR FOR THE SPIN SPECTROMETER AND OTHER EVENTFUL ALBATROSSES

MASTER

David C. Hensley

Oak Ridge National Laboratory*
Oak Ridge, TN 37830

I. Introduction

Because devices like the Spin Spectrometer described in a previous paper[1] to this conference can produce an extremely fast but fairly simple-to-process data stream, it seems reasonable to consider front-end preprocessors having special characteristics. In general, the kinds of transformations being considered do not require floating point calculations or extensive calculations. In order to be somewhat specific, the particular data acquisition/processing problems posed by the Spin Spectrometer at the Holifield Heavy Ion Facility will be discussed.

The Spin Spectrometer[2] is a 4π multidetector γ-ray spectrometer consisting of up to 72 essentially equivalent NaI detectors. Support electronics for the device provide for each relevant detector the following information:

1) a tag word indicating which detector
2) E - the energy pulse height
3) T - the time relative to the trigger
4) PU - an indication of pulse pile up

A sparse data scan is performed so that only those detectors which fired during the trigger strobe time are processed. Event rates up to 3 kHz can presently be supported and hardware modifications could push the rate close to 10 kHz. The data rate depends on the characteristics of the events being studied, but, as an interesting and unfortunately typical example, many heavy ion reactions of interest produce an average of more than 20 γ rays and from 6 to 10 neutrons. Thus, easily 30 detectors may fire during an event. Since 4 words (8 bytes) can be transmitted per detector, an average event can require 250 bytes or more. Consequently an event rate of 3 kHz implies a data rate ~ 750 kbytes/s. As improvements to the front-end hardware are made a maximum rate of $\lesssim$ 2 Mbytes/s could be contemplated.

Currently, the maximum mass storage spooling rate is ~ 120 kbytes/s imposed by our 1600 BPI, 75 IPS tape drives. Upgrading the drives to 6250 BPI, 75 IPS tape drives should push the spooling rate limit to over 500 kbytes/s, close to the maximum achievable on our branch highway. We presently have the capability of fully saturating our maximum future spooling capability!

Obviously we are not too enthusiastic about writing 8 tapes an hour for the rest of our lives, especially since it may take over 4 hours to process one of these high density tapes. It is not unreasonable to predict that 1 hour of accelerator time could require 30 hours of computer time just to perform the first processing scan. At this stage of our understanding of the instrument, it is not easy to make significant reductions in the data rate or storage requirements by special trigger logic. Most of the obviously less useful data constitute only a few percent of the overall load.

II. Kinds of Preprocessing

In order to correct for certain hardware deficiencies, the following simple transformations must be made for the energy pulse height, E:

$$E = E_0 + E'*I + E''*I^2$$

where I is the channel number from the ADC. E_0 corrects for the ADC pedestal; E" is essentially zero up to 5 MeV or so, though there may be some nonlinearities at higher energies. The gain of each detector is preset to the same value, but drifts of the system may cause E' to be different for some of the detectors.

$$T = T_0 + T'*I + \Delta T(E)$$

The time channels have been hardware adjusted so that T' is the same for all detectors; T_0 compensates for small time differences between the detectors. The term $\Delta T(E)$ corrects for the effect of time walk with amplitude, generally significant only when E is below 200 keV.

$$PU = PU_0 + PU'*I + \Delta PU(E)$$

The pile up indicator is very similar to the time, and $\Delta PU(E)$ is a dynamic compensation. PU' is likely to differ significantly from one detector to another.

An iterative calculation is required to correct for uncertainties in the trigger time. Many of our typical triggering detectors provide triggers with intrinsic timing resolution much poorer than those of the individual NaI detectors. The trick is to ignore the original trigger and to calculate a new timing trigger by averaging over all of the γ-ray times measured in an event.

$$T = T_{NaI} - T_{trigger}$$

defines the time measurement for a given detector. If one averages over all of the detectors, one obtains a new trigger timing base

$$T_t = \frac{1}{K} \sum^{K} T_{NaI} - T_{trigger}$$

where K detectors are averaged over. Since the K measurements are uncorrelated, the uncertainty in T_t is approximately:

$$\sigma(T_t - T_{trigger}) \cong \bar{\sigma}_{T_{NaI}} / \sqrt{K}$$

---

*Operated by Union Carbide Corporation under contract W-7405-eng-26 with the U.S. Department of Energy.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MGW

The average timing resolution of the detectors is ~ 2.1 ns FWHM, so the uncertainty in T_t for 25 γ rays would be < 0.5 ns.

A problem arises since some of the detectors will have triggered on neutrons or random events. T_t must be calculated in an iterative fashion. T_t is first calculated, and a time window of say 10 ns is placed around it to remove some randoms and, by time of flight, some neutrons. T_t is recalculated and a window of 5 ns now excludes most neutrons and randoms. The window should be reduced to where essentially all neutrons are excluded but few γ rays are excluded. To further enhance the quality of T_t special weight could be given to time measurements where the corresponding energy was high, since the timing improves with amplitude.

Now that the data in the event have been separated into γ rays and neutrons and T_t has been calculated, the last correction can be attempted. There is a high probability that with 25 γ rays and 6 to 10 neutrons one or more detectors will detect both a γ ray and a neutron. The timing cannot reject such a neutron by time of flight, but the PU indicator may indicate the presence of a subsequent neutron. In that case, the energy pulse height must be corrected, since it is the sum of two pulse heights. As an approximation one could use:

1) $E = E - \bar{E}_n$, where E_n is an average neutron energy, or
2) $E = \bar{E}_\gamma$, where an average γ-ray energy is assumed.

Assuming that all corrections and calculations have been performed, in what way has the spooling load been reduced? First, it should now be possible to reduce the number of words per detector by a factor of 2.

1. TAG+#		1. TAG+T+#
2. E	==>	2. E
3. T		
4. Pu		

The # is an 8-bit number, T could be reduced to 7 bits. An additional saving would occur if neutrons were dropped from the stream -- this might produce a further reduction of 25%.

Up to this point essentially all of the basic information from the Spectrometer has been retained. On the other hand many experimenters would be very happy to receive only two numbers from the system; K_γ, the total number of detected γ-rays, and E_T, the sum of the measured γ-ray-energy pulse heights. There might be some interest in the less precise measurement, K_N, the number of detected neutrons. If only these two or three numbers were transmitted, the data flow would reduce from ~ 250 bytes + user information to 6 bytes + user information. This works out to a reduction of over an order of magnitude in the size of an event. Even in the case where the experimenter insists on having the information on each individual γ-ray detected, K_γ and E_T would be a useful part of the data stream for future fast processing.

Even if the data stream is reduced only by a factor of 2, it means that our future spooling capability should be just adequate. But, most important, the computer time required to scan this corrected data base would be reduced to less than the actual experiment time. The time to scan 300 tapes would drop from 50 days of round-the-clock work to a manageable 3 days.

How long the next (final ?) stages of processing of the information derived from the preprocessing tape scans would take depends both on the physics and the physicist. Our current experience suggests that this step represents an equally formidable task.

III. Kind of Processor Needed

The data rates we are faced with require a very competent computer capability if preprocessing is to be done in real time without reducing the achievable throughput. The rates apparently exceed the capability of reasonable minicomputers. The projected rates could even exceed the bandwidth of a normal peripheral interface, particularly a CAMAC branch highway. Since it seems most reasonable to reserve the main computers for interactive work as much as possible, the computers appear to be indispensible to high rate data acquistion only for fast spooling to mass storage

If the standard in-house computer is not the answer, what about standard μP or μC packages? They have the double advantage that their hardware and their software is well developed. Their major disadvantage is in speed, but because of their rather low cost they can be ganged into banks of parallel processors. A bank of 5 or more 16-bit μP's could approach minicomputer capability for this application so that a bank of 50 or more μP's might serve to handle the desired rates. Since program and data storage memory requirements are not too large, such a bank of μP's is certainly conceivable, probably practical. But the overall structure becomes annoyingly clumsy; there are so many sources and destinations.

A different approach is to design a stripped μP which is carefully matched to this particular problem where extensive calculations and floating point calculations are not needed but speed is of the essence. A CPU which is stripped of all executive duties and most I/O duties can be built with an optimized and reduced instruction set and can economically use 100 ns memory since its memory requirements should be small. New bit slice devices provide the capability of running most operations at a speed of at least 150-200 ns. If special attention is given to the support hardware, such a preprocessor should have a data processing capability nearly 5 times that of a standard minicomputer. If further capability is needed, several preprocessors could be run in parallel.

IV. Hardware and Firmware

The preprocessor envisaged here contains the Am 29116 16-bit μP as an integral part to handle most arithmetic logic functions, add and subtract accumulators, bit logic, and shifting. A TRW 16-bit fast multiplier would permit multiplications in < 300 ns. The 100 ns memory could be divided into two parts; one block of 4k or 8k would contain program and would be as wide as necessary (perhaps 48 bits) to allow single step operations wherever possible, and the other block would be 16-bit memory for data storage and could be expanded to 48k words. A bit slice microsequencer with a 16-bit address field would handle most addressing and all operations not handled by the Am 29116. Division would probably be implemented in microcode using the multiplier and might have a speed near ~ 5 μs.

With this arrangement all of the computations necessary for preprocessing Spin Spectrometer data could proceed at the highest speed (the fast division required to find T_t could be achieved by inverse multiplication).

A special feature having general applications has to do with specifying free-form gates with bit mapping. This might be useful for detailed specification in E vs T space of the separation between γ rays and neutrons. More generally, this is an excellent way to do particle identification quickly. It requires memory, however, and if the processor memory were not enough, then a separate free-form-gate, bit-map memory device might be included. This device would be available to any of the parallel processors and could cut overall memory requirements considerably.

Of particular interest to the Spin Spectrometer processing would be an external pattern recognition device. Given the 72 bit pattern of triggered detectors, it could immediately display just those which had no contiguous neighbor which had fired. Or if only one neighbor had fired, it could indicate whether that detector pair was free from neighbors which had fired.

V. Input/Output

Input and output within the device would be handled with a multiplexed bus depicted in Fig. 1.

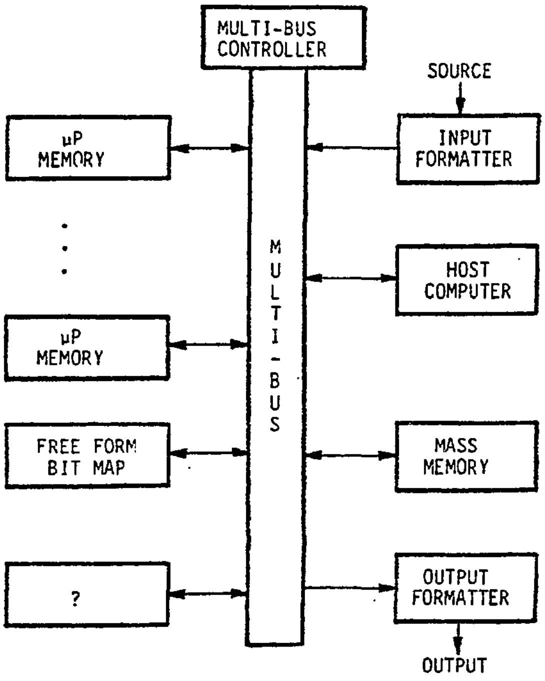

Fig. 1. Schematic parallel processor organization.

Each device connected to the multi-bus could converse with any other (selected) device connected to the multi-bus. The speed of the bus as seen by any device would be $\sim$ 1 MHz, but the bus would be time sliced so that up to 10 separate connections could be handled at the same time, each with the 1 MHz capability. In practice few such connections would be in affect at any one time. The Input-Formatter can connect to only one μP at a time and only one μP can have access to the Output-Formatter. There might, however, be several output devices, a mass memory for on-line histograms, an on-line large disk array handler, and a tape spooling device for storage of the massaged data. It is certainly possible to have more than one input source. But, in no case do I see where a limit of 10 connections between devices on the bus would limit anything. Two or three simultaneous connections would be the most likely combination.

At this point, it is interesting to point out that the data stream source need not be the primary data-acquisition front end. There is another device that is a high rate source of data, a high density magnetic tape drive. From it rates > 500 kbytes/s are typical, and it is a device which is often capable of using up most of the available CPU time of most minicomputers. If it could be switched over to this μP system, much of the load on the interactive computers and the subsequent irritation and frustration could be avoided.

VI. Conclusions

It is difficult to justify the time and effort that would go into developing such a preprocessor system. In most cases, existing computers can handle most of our data acquisition and processing load, even if not in the most speedy fashion. In fact, it is this point that the computers can handle (or almost handle) the load that is so exasperating. If so called spare time and effort are to be expended, any reasonable analysis would strongly suggest that such resources ought to be invested in the computers; more memory, better and bigger disks, utilization of writeable control store, and more and better software. It is this fact that computers are so much the lifeblood of physicists now that seeing them diverted to "simple" tasks is a major argument for trying the approach of distributed processing. The realization that a single CPU has trouble handling the load leads naturally to parallel processing.

It still requires a device like the Spin Spectrometer with possible data rates over 1 Mbyte/s to furnish the final justification. If we do not find some way to solve the major burden of its preprocessing load, then many of the qualities of the spectrometer will be effectively lost. We currently have spooled ourselves into several months of CPU time for simple preprocessing. We would like to trade our albatross in for a parallel preprocessor.

References

1. D. C. Hensley, proceedings this conference.
2. D. G. Sarantites et al., Journal De Physique, Colloque C10, supplement au n° 12, Vol. 41, C10-269 (1980).