ISTITUTO NAZIONALE DI FISICA NUCLEARE

Sezione di Pavia

INFN/TC-89/8
1 Settembre 1989

A. Lanza, M. Caresana:

A DSP Based Processor for High Rate Data Analysis and Reduction

Servizio Documentazione
dei Laboratori Nazionali di Frascati

INFN - Istituto Nazionale di Fisica Nucleare
Sezione di Pavia

INFN/TC-89/8
1 Settembre 1989

A DSP BASED PROCESSOR FOR HIGH RATE DATA ANALYSIS AND REDUCTION

A. Lanza and M. Caresana
Dipartimento di Fisica Nucleare e Teorica, Universita' di Pavia and INFN - Sezione di Pavia

ABSTRACT

In this report we describe a prototype VME board that adopts a DSP (Motorola DSP56001) as central processor. This board has 64K x 16 bits of SRAM data memory, 32K x 24 bits of SRAM program memory and as many bits of EPROM program memory, 5 registers, a 'ROAK' type interface with VME, an external interface 'VME like' to collect data with 8 parallel bit capability and an RS-232 C interface for a terminal. Moreover a very useful monitor has been realized, to debug the user's programs.

1. - INTRODUCTION

The possibility of the very large scale integration technique and the coming of innovative architectures have led the semiconductor industry to realize very specialized microprocessors, able to obtain better results than similar general purpose circuits, in term of execution velocity and easiness of use, with the same clock rate.

The digital signals elaboration is one of the fields that makes a great advantage from this situation: an on-line filtering operation, that needs some dedicated hardware to follow data acquisition before, now can be carried out by a single integrated circuit, able to complete the

elaboration and manage the data latching. Moreover these devices are fully programmable, and therefore it is possible to change the implemented algorithms with simplicity, also on-line.

These developments are assuming a great importance even in High Energy Physics experiments, where in general big amount of data are collected in a very little time, and where the data must be analyzed, at least partially, to be able to make a selection between significant and non significant data, to avoid the writing of large quantities of tapes.

For these reasons the requirements to realize intelligent boards supplied with these devices, and with interfaces with the data acquisition standards employed in the experiments, are risen off. These notes describe a prototype board, called DSPPROT, based on a Motorola digital signal processor, the DSP56001, and interfaced directly with the VME bus; this board has been realized to provide specifically for the data reduction in an experiment of searching solar and galaxies neutrinos[1], but it may be used with little or none modifications in whatever experiment in place of traditional CPUs.

2. - THE MOTOROLA DSP56001

DSP56001[2] is a RISC (Reduced Instruction Set Computer) microprocessor with a Harvard type pipeline architecture, and it is dedicated to digital signal analysis; the DSP56001 is employed in any field of data processing in which is necessary very high speed of instruction execution, great precision calculation and very large throughput.

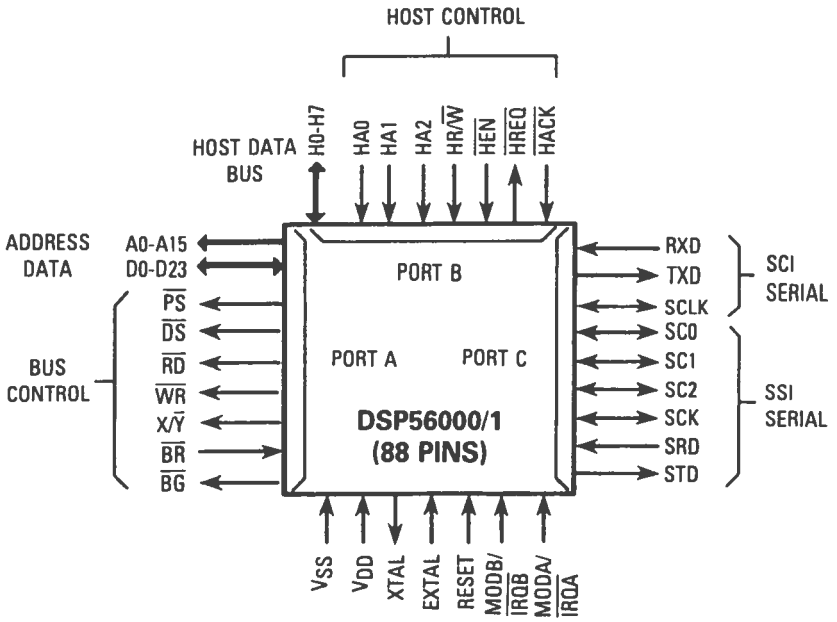

FIG. 1 - DSP56001 Functional Signal Groups.

To give a performance evaluation of the 56001 is sufficient to say that the speed of elaboration reaches 10.25 MIPS (with 20.5 MHz of clock rate), and that the precision of calculation has 144 dB of dynamics, with intermediate results that can reach 336 dB of dynamics, using the two 56 bit accumulators.

Fig.1 gives a panoramic vision of the available groups of I/O, whereas fig.2 represents the internal block diagram of the 56001. Following this figure it is possible to distinguish 3 great blocks: Elaboration, Memory and I/O.

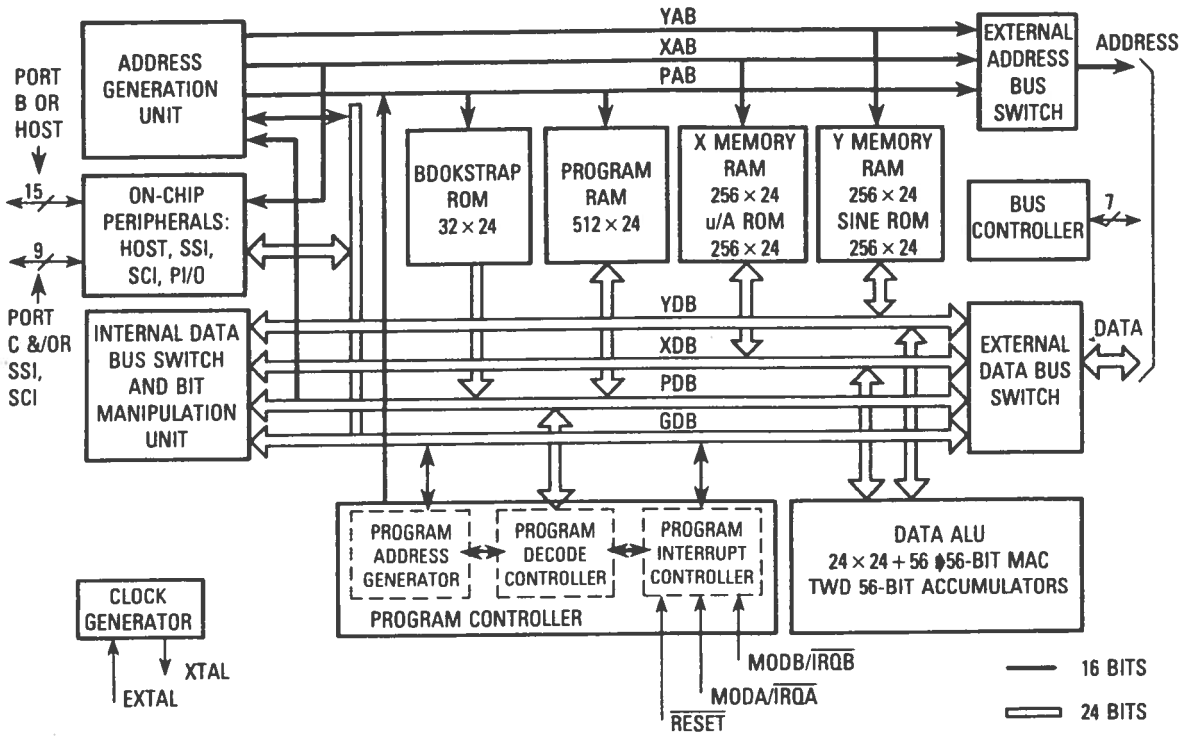

FIG. 2 - DSP56001 Block Diagram.

The Elaboration system is formed by 3 units:

- ALU, that realizes all the logic and arithmetic operations. It is makes up in turn by 4 input registers 24 bit each (X0, X1, Y0 and Y1), 2 accumulator registers 56 bit each (A and B), 1 accumulator shifter, 2 data bus shifter/limiters and 1 parallel single cycle non-pipelined multiply-accumulator (MAC). The ALU is capable of performing operations between 56 bit each operands;
- Address Generation Unit, that contains 24 registers 16 bit each to realize 18 addressing modes, and moreover it is gives the possibility to use 3 types of addressing: linear, modulo and reverse carry;
-Program Controller, to prefetch and decode the instructions, to execute hardware loops and to control interrupts.

The Memory system is composed by 3 internal RAM memories (fig.2): two of these are data memories (X and Y memory) and are organized in 256 x 24 bits, the latter is the program memory (P memory), 512 x 24 bit large. It is possible to use two other internal ROM memories (X and Y), 256 x 24 bit each, preprogrammed by Motorola with 2 look-up tables: μ/A law in X memory (it is a table very useful in telecommunication) and the sine function, calculated over the 4 quadrant, in Y memory.

All the 3 memory types can be expanded externally until 64K x 24 bit each, thanks to the switches shown in fig.2; it is possible to select among 4 different structures for the P memory (completely external, or internal memory enabled and external reset, etc.) properly programming the OMR (Operating Mode Register). The 2 internal ROM memories can become completely transparent via software, to provide more large external addressing space.

The peripheral devices are addressed using memory mapped technique, and are mapped in the later 64 words of the external X memory (the DSP internal devices) and of the external Y memory (the DSP external devices). The memory space of the interrupt vectors occupies the initial 64 words of the internal P memory.

In fig.2 is also shown a particular internal ROM memory, the Bootstrap memory, that is not accessible to the user; it allows of performing the automatic loading of the P internal memory, after the reset, either from an external EPROM or from the DSP Host port.

As one can see in fig.1, there are no acknowledge lines from external memories. The speed of data exchange between DSP and external word is regulated via software, setting an appropriate number of wait cycles, independently for the 3 memory types and for the external peripherals. The wait cycles can vary from a minimum of 0, corresponding to 55 ns of access time, to a maximum of 15, corresponding to 820 ns of access time. These values are valid for a 20 MHz clock rate.

The I/O system provides other 3 ports, more than Data and Address busses for external memory expansion (port A in fig.1):

- a 8 bit parallel port with separated Data and Address busses, to provide a connection with a host computer (Host port or port B). By means of this port operations in DMA mode or in Interrupt mode are possible; moreover the host computer in turn can get up a vectorized interrupt to the DSP, using an appropriate register (Host Command). This port can be programmed via software as 15 bidirectional general purpose I/O lines;
- a serial asynchronous full-duplex port (SCI, Serial Communication Interface) with software programmable baud-rate (up to 2.5 Mbaud!), that can be used as RS232-C line to connect a terminal;
- a serial synchronous full-duplex port (SSI, Synchronous Serial Interface), usually used to connect serial devices, other DSPs, other microprocessors, etc.

Both the serial ports can be programmed via software as 9 bidirectional general-purpose I/O lines (port C).

The principal features of the DSP56001 are:
- very high speed calculations (multiplications and divisions). It can execute a 1024 points FFT in 3.39 ms;
- high parallelism of the elaboration system. Since the ALU, the Address Generation Unit and the Program Controller carry out in parallel, the DSP can execute the next instruction prefetch, a 24 x 24 bit multiplication, a 56 bit addition, two data moves and two address register updates in only one instruction cycle (100 ns);
- great amount of memory, with 128K x 24 bits of Data memory and 64K x 24 bits of Program memory. This characteristic, together with the splitting of the Data memory, allows to analyze a great deal of data without transferring the results to the system memory;
- a RISC instruction set with 62 mnemonics, that offers the possibility of single cycle instruction multiplications and divisions and allows the N times repetition of an instruction and hardware loops, still with a single cycle instruction.

Motorola also provides the development software, that allows a great saving of time to the user. It runs on MS-DOS and is formed by an assembly cross-compiler and a simulator. Program examples (Fourier Transforms and digital filters) and utility software to program EPROM memories are also provided.

3. - THE DSPPROT BOARD

It is a VME slave board that interfaces the DSP56001 with the VME bus, using the 8 bit Host port, with a readout board, called JOB, by means of X memory bus, and with a terminal, by means of SCI port configured like an RS232-C.

The fig.3 shows the block diagram of the board. It can be subdivided in 4 parts: the registers external to the DSP, the VME interface, the JOB interface and the terminal interface.

3.1 - The registers

The registers implemented on board are 5, 1 with 16 bits and 4 with 8 bits: HCR (Host Control Register), used by VME, DCR (DSP Control Register), used by DSP, SR (Status Register), used by both VME and DSP, CCR (Channel Counter Register), used by DSP, and GPR (General Purpose Register), used by VME and DSP.
- HCR: it is an 8 bit register, and can be read or write only by VME. Its address in the VME memory space is $7FFFA. Only the first 3 bits are used:
HCR0 - if it is set, the VME accedes to the first JOB interface;
HCR1 - if it is set, the VME accedes to the second JOB interface;
HCR2 - if it is set, the DMA read/write of the Host port is enabled. It is necessary that the ICR (Interrupt Control Register) of the DSP is set to enable the DMA mode of the Host port. The

ICR value depends on the length of the moving word, and for the right setting the user can refer to the DSP User Manual(2). The DMA procedure is described later;

HCR3...HCR7 - not used.

At power on all the bits are cleared. The reset does not clear the register.

- DCR: it is an 8 bit register, and can be read or write only by the DSP. Its address in the DSP memory space is X:$F000. Only 2 bits are used:

DCR0 - not used;

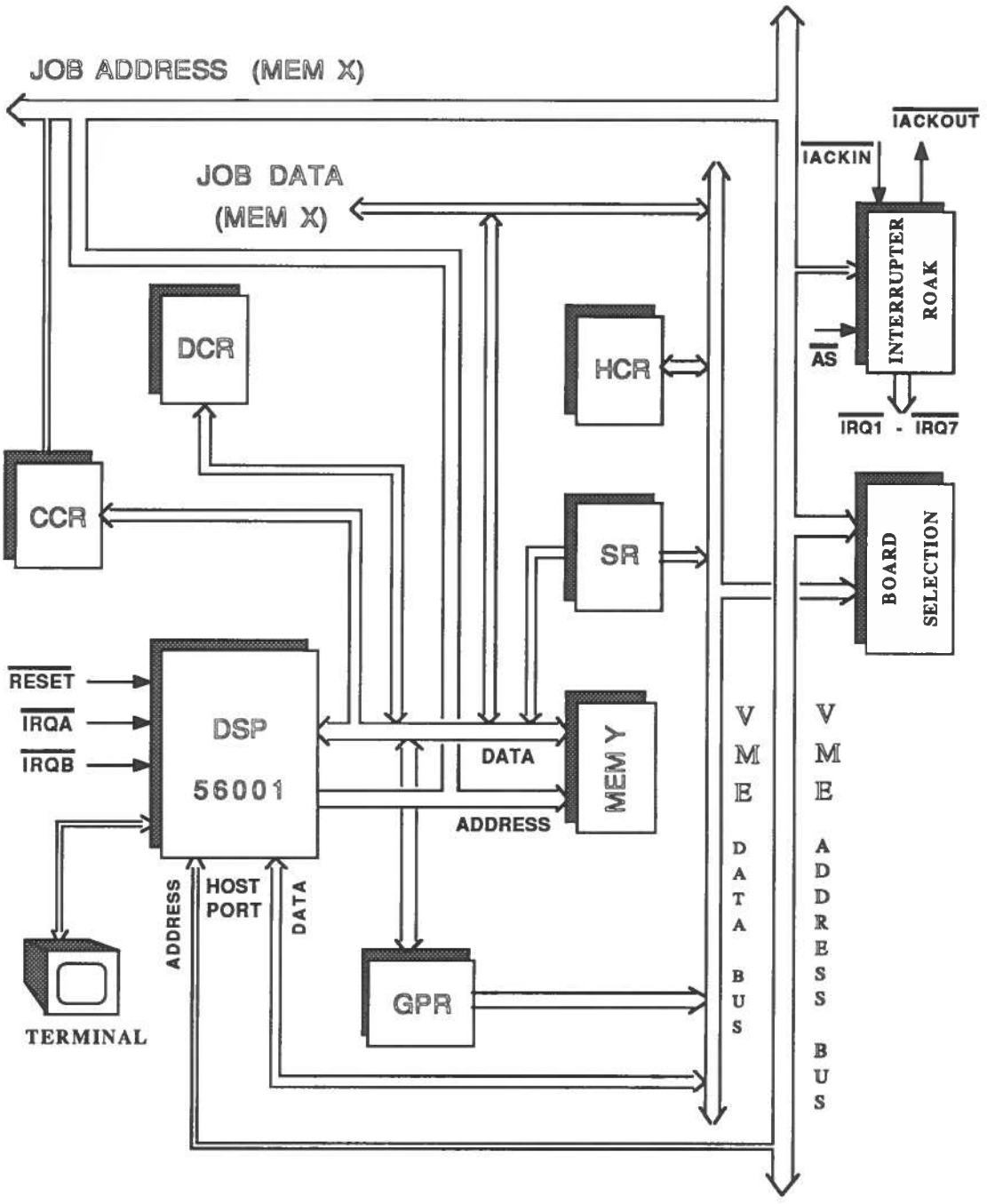

FIG. 3 - DSPPROT Block Diagram

DCR1 - if it is set, the DSP accedes to the first JOB interface;
DCR2 - if it is set, the DSP accedes to the second JOB interface;
DCR3...DCR7 - not used.

The DSP access to the first or to the second JOB interface is privileged compared to the VME access; the latter knows if the DSP is acceding to the JOBs testing the Status Register. It is not possible to accede to the first (second) JOB interface from VME if the DSP is acceding to the second (first) JOB interface.

At power on all the bits are cleared. The reset does not clear the register.

- SR: it is an 8 bit register accessible from VME and DSP, but only for reading. Its address in the VME memory space is $7FFFB, and in the DSP memory space is X:$F001. 5 bits are used:
SR0 - if it is set, the DSP is acceding to the first JOB interface;
SR1 - if it is set, the DSP is acceding to the second JOB interface;
SR2 - if it is set, the VME is acceding to the first JOB interface;
SR3 - if it is set, the VME is acceding to the second JOB interface;
SR4 - if it is set, the DSP is waiting to transfer data (or is transferring data) from the Host port to the VME. If it is cleared, the DSP has terminated the transfer or it is acceding to the JOBs;
SR5...SR7 - not used.

- CCR: it is an 8 bit register, accessible by DSP only for writing. Its address in the DSP memory space is X:$F003. It is used to address the 40 channels of the JOB (8 Kbyte each) taking up only the 8 Kbyte window from X:$2000 to X:$3FFF. The offset address ($2000) is necessary to avoid the first 256 memory locations internal to the DSP, that are not addressed from the Address bus.

The first 13 bits of the DSP Address bus (A0 - A12) are sent directly to the JOB (in this way the address seen by the JOB is $0000 - $1FFF). The remaining 6 bits required to complete the 40 channels addressing (A13 - A18) are provided by the CCR, that is setting via software by the DSP before beginning the channel addressing: the channel #0 corresponds to the CCR value $00, the channel #1 to the value $01, and the channel #39 to the value $27.

- GPR: it is the only 16 bit register, and can be read or write by the DSP, but can only be read by VME. Its address in the DSP memory space is X:$F002, and in the VME memory space is $7FFF8. It can be used to exchange information between DSP and VME, and in particular the DSP can write the dimension of Y memory really occupied by the data, in such way the VME can conveniently manage the reading.

3.2 - The VME interface

The interaction between DSPPROT and VME occurs onto two levels: the former, the selection level, is necessary to make run the board, and the latter is the data transfer using the DSP Host port.

The VME selection of the board is made with a longword type write operation of a number from $00 to $FF, at the base address of the board. For example, the board #7 with base address $200000 is selected with the instruction

MOVE.L #7,$200000

The comparison is made both on the base address and on the selection number, so it is possible to place up to 256 boards with the same base address. The latter holds the address lines from A16 to A23, therefore the minimum addressing step is 64 Kbyte.

The selection has the greater priority compared to whatever operation: a not selected board can't respond to anyone instruction of the master CPU. Moreover the selection is not made if a byte or a word writing is used. The base address and the selection number are set on board with jumpers.

Concerning to the data transfer, the board is seen by VME as a slave module with a 'ROAK' interrupter. The occupied memory space is 512 Kbyte, and the base address is the same described above. The data are moved using the least significant 8 bits of the VME data bus, D0 - D7.

The choice of the memory space has been suggested by the direct compatibility requirement with the readout JOB. The first 312 Kbyte of this memory space are occupied by the JOB data memory (relative address $00000 - $4FFFF), and the registers accessible from VME stay on the last 16 locations, from $7FFF0 to $7FFFF. They are:

- the Host port registers internal to the DSP, with addresses from $7FFF0 to $7FFF7;
- the 3 DSPPROT registers, SR ($7FFFB), HCR ($7FFFA) and GPR ($7FFF8 and $7FFF9);
- the 3 JOB registers, Event Counter ($7FFFD), Control Register ($7FFFE) and Status Register ($7FFFF). These addresses are valid for both the JOB connected to the DSPPROT;
- the address $7FFFC is used for reading the DSP in DMA mode.

The memory space from $50000 to $7FFEF is not used.

The implemented interrupter, as above mentioned, is of 'ROAK' type, that is Release On AcKnowledge, and the priority level is jumper selectable from 1 to 7.

The interrupt request is launched by the DSP by means of HREQ signal; this one is true when the transmission register of the Host port is full, and therefore it must be read by the master CPU. To the arrive of the interrupt, the interrupt handler of the master CPU drives the IACKIN - IACKOUT daisy-chain; the DSPPROT interrupter passes through the Interrupt Acknowledge if the interrupt has not been launched by itself, otherwise it stops the daisy-chain and responds to the Status-Id sent by the master. The interrupter releases the interrupt request after the two Data Strobes are come back false.

It is possible to accede to the 2 JOBs, directly connected to the VME, without passing through to the DSP. In this case the addresses from $00000 to $4FFFF are used for the data (of the

selected JOB) and the addresses from $7FFFD to $7FFFF are used for the registers. The JOB selection is set programming the HCR register.

In fig. 4 are shown 4 photos, to document the performances of the interface connected to a MVME133 master CPU, with a 12.5 MHz 68020 microprocessor. In the photo 4.a the 3 byte (24 bit) data transfer from DSP to VME with interrupt is visible. The time necessary to the VME to read the 3 bytes is 10.9 µs, as we see at the top. The DSP clock frequency is 20 MHz.

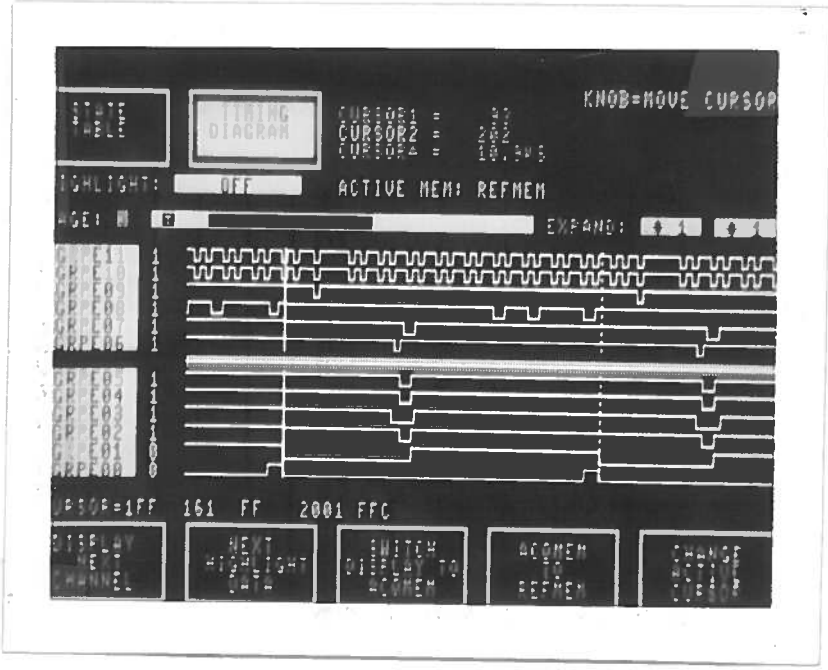

FIG. 4a - 3 byte data transfer from DSP to VME with interrupt: the GRPE00 line is the Interrupt Request signal to the VME, the GPRE03 line is the IACKIN signal on the DSPPROT board, and the GRPE10 and GRPE11 lines are the DS (Data Strobe) and the X/Y (Y memory select) signals of the DSP, respectively.

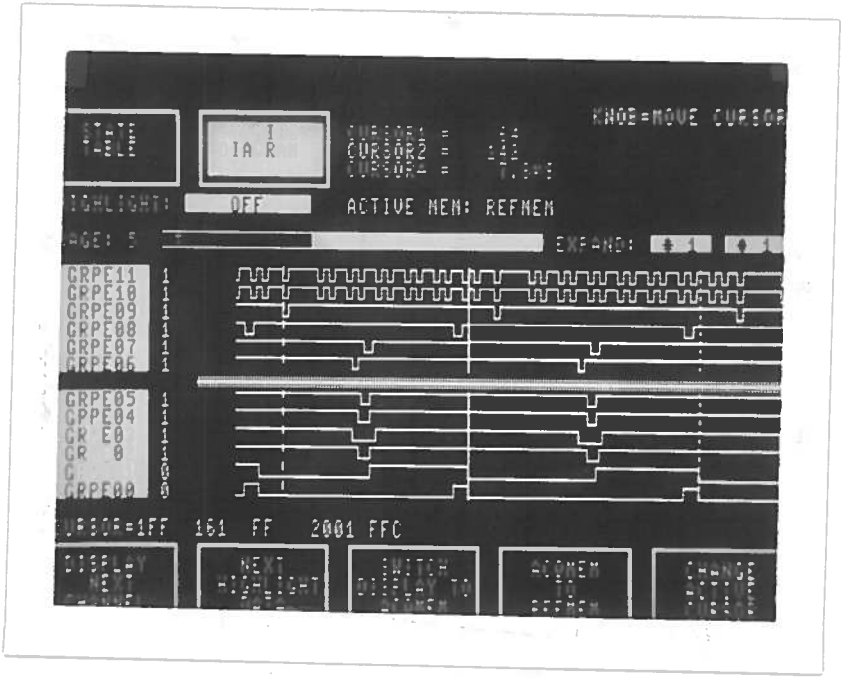

FIG. 4b - 1 byte data transfer from DSP to VME with interrupt: the lines are the same of the previous figure.

In the photo 4.b, instead, is shown the single byte transfer; all the rest is identical to the photo 4.a, and the execution time of the VME read cycle is 7.8 µs.

The photo 4.c illustrates the DSP-VME single byte transfer in DMA (seen by the DSP): the byte reading time in this case is 2.5 µs, and it is due substantially to the time employed by the MVME133 to change the reading address. In all the photos the memory access to the DSP Y memory, that contains the data, is programmed with 2 wait cycles (150 ns).

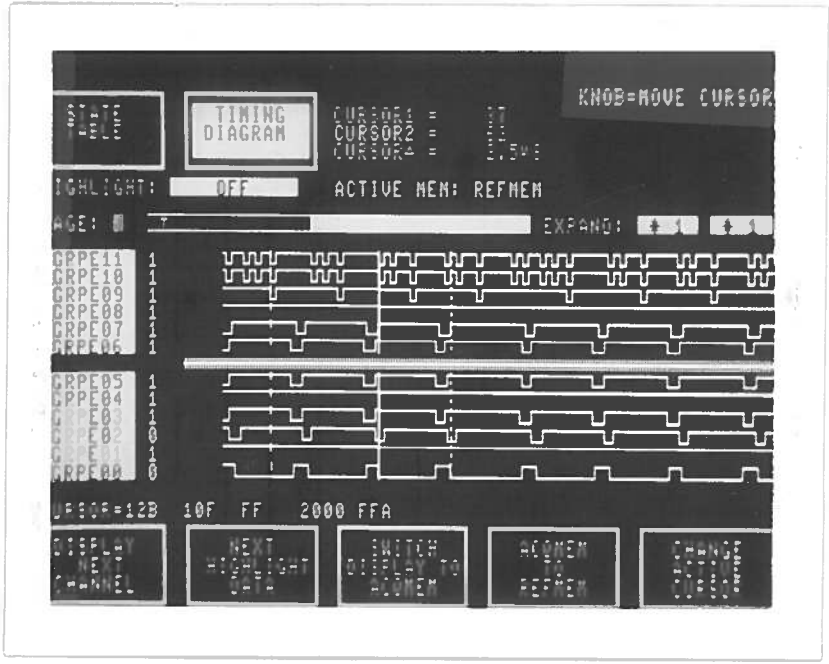

FIG. 4c - 1 byte data transfer from DSP to VME in DMA: the GPRE00 line is the DMA Request signal, and the GRPE10 and GRPE11 lines are as in the previous figure.

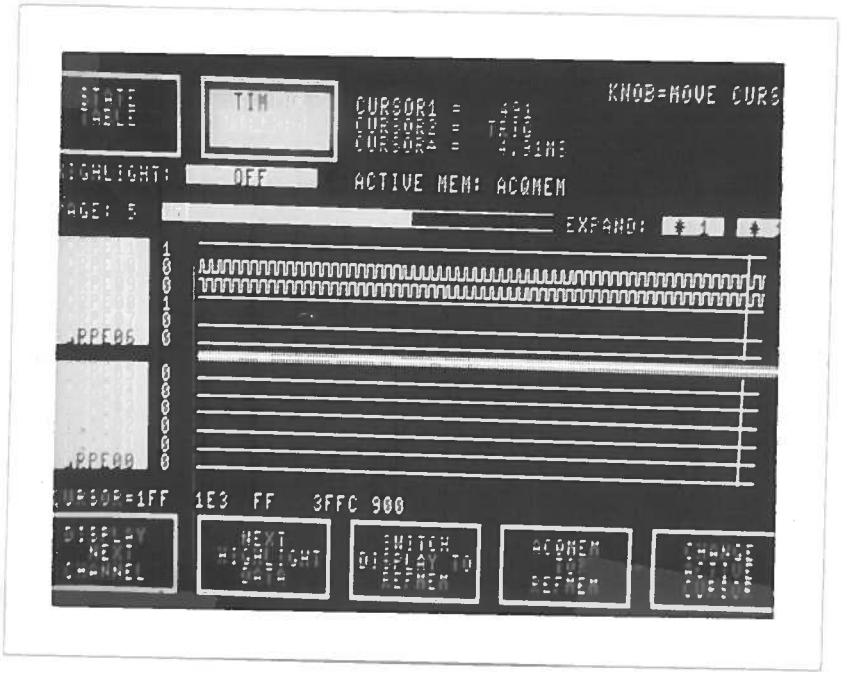

FIG. 4d - The photo shows the time necessary to analyze 1 JOB channel with the test program A(3).

Finally, in the photo 4.d is shown the time necessary to analize one JOB channel with the program A[3]. The measured value, 4.91 ms, is in perfect agreement with the simulation result, taking into account the
positioning error of the logic analyzer cursor, that has a minimum step of 10 μs, and the approximation introduced by the software pointer in the program.

3.3 - The JOB interface

The JOB[4] (John Oliver board, so called from the name of the designer) has been realized to acquire the data come from the front-end of an experiment based on trace detection in liquid Argon, and adopts a 'like VME' protocol for data transfer.

A single board (with dimensions bigger that Fastbus one) is composed by 40 reading channels, and a single channel has a voltage amplifier, a flash ADC (EF8308, constructed by SGS-Thomson), and 2 SRAM
buffered memory banks, 8K x 8 bit each. One of the two memories, the primary memory, is written continuously with the sampled signals of the ADC at a 5 MHz frequency, until the arrive of an external STOP signal.

From this time, the contents of the primary memory, that substantially represents the signal history until 1.6 ms before the STOP signal arrival, is discharged on the other memory bank, and the operation is 3.2 ms long; the latter is the reading dead time.

Successively the secondary memory is read by a VME CPU, when the data discharge is finished, and at this point the primary memory is written again.

The memory space occupied by the JOB is 512 Kbytes, and the 40 channels are addressed sequentially, from $00000 to $4FFFF, for an amount of 312 Kbytes. The 3 board registers are: Status Register, Control Register and Event Counter, and their addresses are $7FFFF, $7FFFE and $7FFFD respectively. The rest of the memory is not used.

The JOB interface allows 2 JOB boards to be connected, and they are seen, one at a time, in the X memory space of the DSP, from $2000 to $3FFF.

The DSPPROT board begins the data reading when the FULL line, driven by both the JOBs, becomes true, indicating that almost one of the boards has completed the data acquisition. This line is driven by open collector ports, and it is connected with the interrupt input IRQA of the DSP.

To know what JOB has activated the FULL line, the DSP reads the Status Register (and exactly the FULL bit) at the JOB address $7FFFF; the latter is obtained setting the CCR register to $3F, and reading to the DSP address $1FFF. Successively, the DCR is programmed to select the interrupting JOB.

The reading of the channel #0 is made programming the CCR to $0 and addressing the JOB from $2000 to $3FFF; the address doesn't begin from $0000 because the first 256 locations of the DSP X memory are internal, and the relative addresses don't appear on the external address

- assigning the RAM area of the P memory used by the monitor;
- receiving the characters coming from the terminal and sending back the echo;
- interpreting the commands and calling the appropriate routines;
- making a series of controls on the commands, for example rejecting the ASCII characters with code higher than $7F, or recognizing the special characters Delete, Back Space, Carriage Return, etc.;
- managing XON - XOFF coming from the terminal.

The characters sent by the terminal while the monitor is executing previous commands are collected in an input temporary buffer, created in the RAM portion of the P memory.

The commands implemented are:

- Memory display: MD <memory type> <addr1> [<addr2>]
- Memory modify: MM <memory type> <addr>
- Register display: RD [<options>]
- Register modify: RM [<options>]
- Program execution: GO [<addr>]
- Program tracing, in single instruction mode or in step mode: TR [<step count>]
- Breakpoint setting: BR <reg> <value>
- Breakpoint deleting: NB [<reg>]
- Program execution with breakpoints: GB [<addr>]
- Block fill: BF <memory type> <addr1> <addr2> <hex data>
- Block move: BM <memory type> <addr1> <addr2> <addr3>
- Memory test: BT <memory type> <addr1> <addr2>
- Software reset: RE
- Help: HE

In case of wrong syntax, the monitor provides diagnostic messages that highlight the error type.

ACKNOWLEDGEMENTS

We wish to thank Prof. S. Centro, that first had the idea to use a DSP (Thomson TS68930) for the data reduction, for the suggestions given to us. We also acknowledge the Pavia group of the Icarus Collaboration, Dr. G. Bressi, Dr. G. Cecchet and Prof. M. Cambiaghi, for the help provided during the whole work, and Mrs. M. Formenti, for the drafting of these notes.

bus. The reading of the channel #39 needs the CCR to be programmed to $27, while the JOB addressing space remains the same.

When the channel reading is finished, the DSP must clear the FULL bit of the Status Register of the JOB it has read, accessing to the Control Register of the same board; if the FULL line remains active after this operation, the DSP must read even the second connected JOB, proceeding with the sequence we have seen.

It is important that the memory access time of the JOB is respected: it is jumper selectable to 150, 200 or 250 ns; the BCR register of the DSP must be programmed coherently with the choice made for the JOB.

3.4 - The RS-232 C interface

The DSP has a SCI port that can be configured via sofware as full duplex RS-232 port. The bit number of the data packet is software selectable: among the asynchronous protocol options, it can be 11 or 10, respectively with or without parity, with 1 start bit, 1 stop bit, and 8 data bits; the parity, if enabled, can be even or odd. The baud-rate is software selectable, up to 2.5 Mbaud.

The SCI port is interfaced with the DSP internal registers through 2 registers, 1 for the parallel-serial conversion (transmit) and 1 for the serial-parallel conversion (receive). The receive mechanism occurs by means of interrupt, the transmit mechanism by means of interrupt and/or polling.

The hardware external to the DSP must only provides to interface the voltage levels, and it is formed by the standard TTL interfaces MC1488 (TTL -> RS-232) and MC1489 (RS-232 -> TTL).

4. - THE MONITOR

To make easy the use of the board and the software debug, a monitor[5] has been written in Assembly language; it traces in the syntax some of the monitors more used [6,7] .

The monitor, DSPBug, is resident in an Eprom in the P memory space, starting from $C600. We must be borne in mind that this memory space can contains even the program to load into the DSP internal memory, in case of boostrap from Eprom, and this latter starts from $C000 and arrives until $C5FF. The DSPBug also occupies a memory P space reserved to RAM memory, from $7A00 to $7FFF.

Its structure is based on a kernel and on many routines that implement the commands execution. The kernel takes care of:

- initializing the DSP registers, and in particular the registers of the SCI port, used by the terminal;

REFERENCES

(1) ICARUS Collaboration: "Searching for new underground phenomena with high resolution visual techniques and magnetic analisys", 1985

(2) "DSP56000 Digital Signal Processor - User's manual", Motorola DSP56000 UM/AD 1986

(3) G.Bressi, G.Cecchet, A.Lanza and F.Panetsos: "Use of a DSP in High Energy Physics Read-out Systems", Pavia University internal report DFNT/AE 07/87, 1987

(4) J. Oliver: "Icarus Waveform Digitizer Board", Harvard University internal note, 1986

(5) M.Caresana and A.Lanza: "A monitor for the Motorola DSP56000 Family", Pavia University internal report, to be published

(6) Motorola MVMEBug/DI debugging package, June 1986

(7) M.Demoulin: "CPUA1Mon User's Manual", UA1-EF-CERN 1985