IEEE Nuclear Science Symposium, San Francisco, Nov. 1993. IEEE Trans. on Nucl. Sci. Vol. 41, No. 4, August 1994, p1187

A 64-ch Time Memory Cell Module with a DSP and a VME Interface

Yasuo Arai and Masahiro Ikeno
KEK, National Laboratory for High Energy Physics
1-1 Oho, Tsukuba, Ibaraki 305, JAPAN

Abstract

A new 64-channel Time Memory Cell (TMC) module has been developed for high-rate wire-chamber applications. A combination of the TMC chip and associated FIFO memories digitize and store input signals at 1 nsec/bit resolution for a period of 64 μsec. To handle the large data size, a digital signal processor (DSP56002) is implemented in the module. The size of the module is 9U x 400 mm Euroboard size, and has a VME interface using the P1 and P2 connectors. The P3 connector has been assigned for the output of trigger signals.

I. Introduction

High-density time-to-digital conversion modules with around 1 ns resolution are in demand for wire-chamber readout. Especially, multi-hit and dead-time-less capabilities are required in high-rate experiments.

The Time Memory Cell (TMC) [1, 2] is a custom chip that digitizes input signals every 1 ns and stores the data inside memories. The TMC has unique features of low-power (< 10 mW/ch), high density (4 ch/chip), good timing resolution (σ = 0.52 ns), and high stability. A 32-channel CAMAC module having 1 μsec depth TMC's that we built [3] has been used in many test experiments. One of the problem concerning the CAMAC module is the data-readout scheme. Since the module has no zero-suppression logic, it takes a relatively long time to read out all of the data. Furthermore, since the output data from the module is encoded, users need to process the data in order to extract required information. At the same time, a wider timing range would be required in some applications.

We have developed a new 64-channel TMC-VME module (Fig. 1). To extend the time range from 1 μsec, external FIFO buffers have been connected to the output of the TMC. The FIFO, having a word depth of 2 k, can store up to 64 μsec of data. This large data size (64 ch x 2 kw = 128 kw) requires a high-speed processor to be on the board. We implemented a high-speed digital signal processor (DSP) of the Motorola DSP56002. A monitor program and data-processing routines have also been developed and are described in section III.

The board size is 9U x 400 mm Euroboard size, and the power consumption is less than 30 W. The module has a VME slave interface and Dual Port Memories (DPM) and several registers can be accessed from VME. Connector P3 has been assigned for the output of trigger signals.

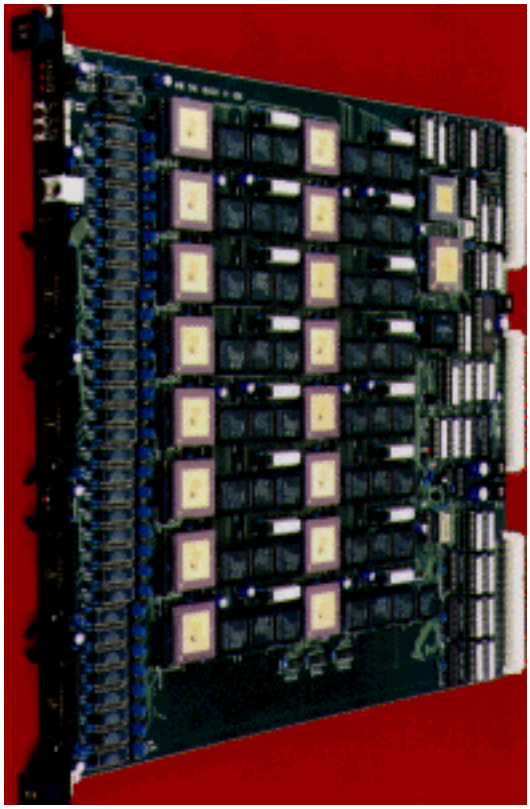

Fig. 1 Photograph of the 64 ch TMC-VME module.

II. Circuit Description

A. Overview

The specifications of the TMC-VME module are summarized in Table 1. The module has 16 chips (TMC1004) which are produced by NTT Electronics Technology Co. using a 0.8 μm CMOS process. The TMC records input signals at 1 ns/bit resolution and stores past 1 μsec of data. The TMC has 4 channels in a chip and encodes the data-transition point within 32 ns to 6-bit data. Thus, each chip has 24-bit data outputs.

The DSP56002 can perform at 20 MIPS with a 40 MHz clock, and up to 3 instructions can be executed in a single instruction cycle. While the DSP has 56-bit accumulators, it has a 24-bit external data bus. This matches with the data size

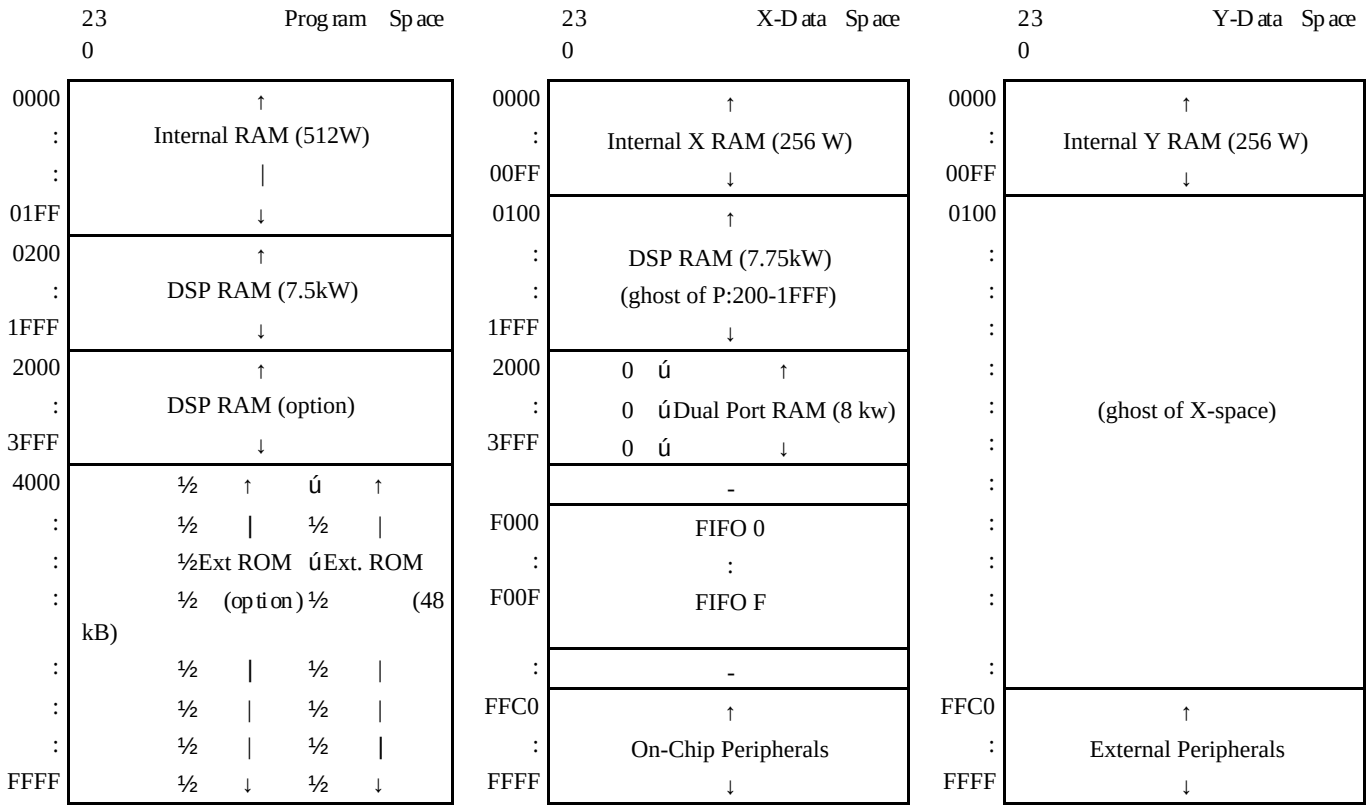

Fig.2 DSP memory address map

of the TMC. The DSP has three address spaces: Program (P-), X-data, and Y-data space. Figure 2 shows a memory map of the DSP.

There is an external RAM (DSPRAM: 8 k x 24 bit); an additional RAM socket is provided for future expansion. Since there is plenty of address space, the RAM is mapped to all three spaces. The user can assign the address space of the RAM as either P-, X- and/or Y- data space in the software. This flexibility of mapping is useful for improving the software performance. The monitor software is programmed in a PROM (64 k x 8 bit). There are two PROM sockets in total.

A block diagram of the TMC-VME module is shown in Fig. 3. Since the TMC operates at 3 V, there are 3V/5V level converters between the TMC and DSP buses. Some of the control signals are clamped by Schottky diodes in order to protect the TMC inputs. Since the output signals of the TMC chip have a sufficient voltage swing to drive the TTL device, these are connected directly. Because the same clock of 31.25 MHz is used for both the TMC and DSP chips, both chips run synchronously.

The module has a serial (RS-232C) port at the front panel and an 8-bit parallel host port in VME space. A dual port memory (8 k x 16 bit), which can be accessed from both the DSP and VME, is provided as a data buffer.

Input signals are received by fast comparators (MAX 9698) which receive ECL or small differential signals (> Δ100 mV) in a common mode voltage range of ±3 V.

B. TMC and Start/Stop Control

Each TMC channel comprises 32 rows by 32 columns of memory. The 32-bit data of a row are encoded as 6-bit data. Since the TMC memories are dual-ported, signal recording

Table 1 Specifications of the TMC-VME module

TMC	TMC1004 (31.25 MHz)
DSP	DSP56002 (31.25 MHz)
DSP RAM	MCM56824 (8k x 24b) x 2 (no wait)
ROM	27512 (64k x 8b) x 2 (4 waits)
Dual Port Memory	IDT7025 (8k x 16b) (no wait)
I/O	Serial (front) and 8-bit host port (VME)
FIFO	74ACT7808 (2k x 9b) x 49 (no wait)
Signal Inputs	64 ch (ECL) differential. Four 34-pin Flat Cable Connectors.
Start/Stop Input	NIM level. Lemo-type connectors.
Trigger Output (P3)	TTL low assert signals. Stretched to 32ns. 64 signals.
Least Time Count	1 ns/bit
Max. Time Range	64 μsec
Two Hit Resolution	~40 nsec
Time Measur. Error	σ = 0.72 ns (start/stop hit)
VME I/F	A24/D16
(P1 and P2)	Interrupt to/from VME
Packaging	9U x 400 mm, single-width module.
Power	+5 V x 4.2 A, -5.2 V x 1 A,
(P1 and P2)	3.3 V x 1.2 A, < 30 W

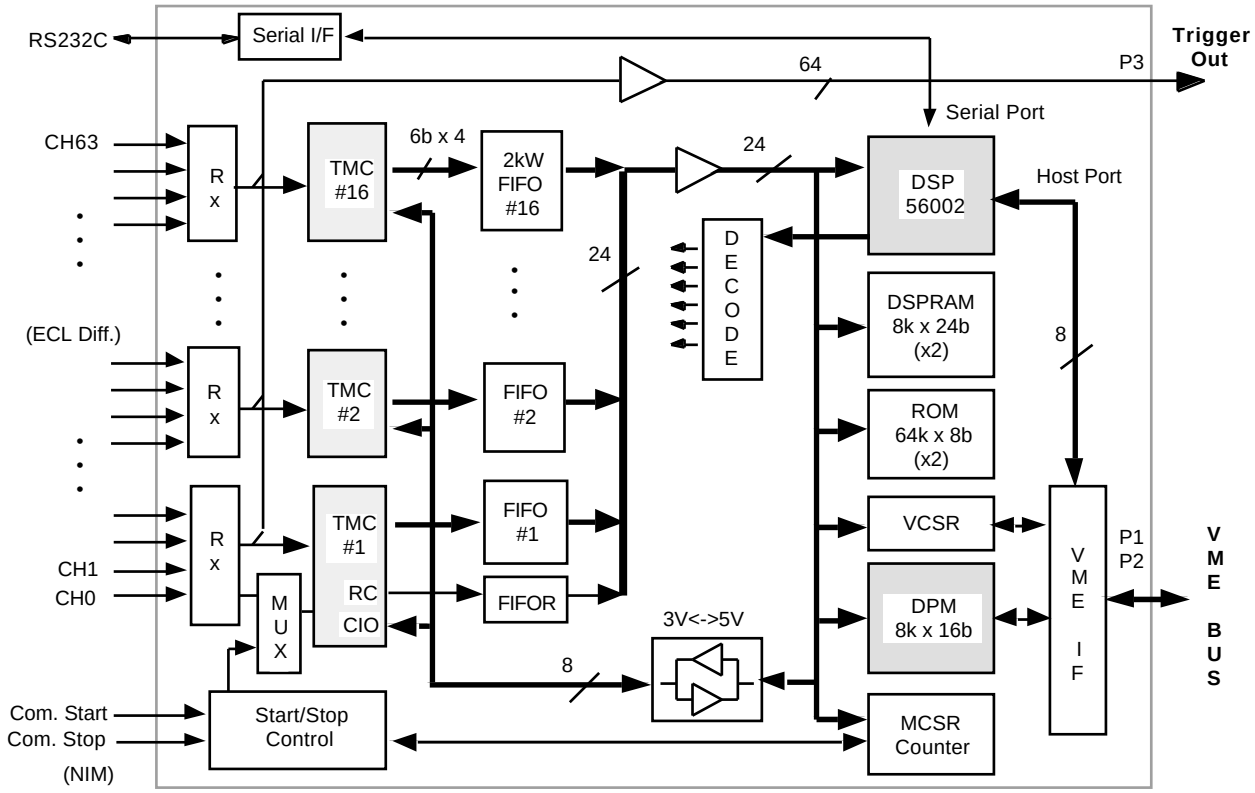

Fig. 3 Block Diagram of the 64-ch TMC-VME module.

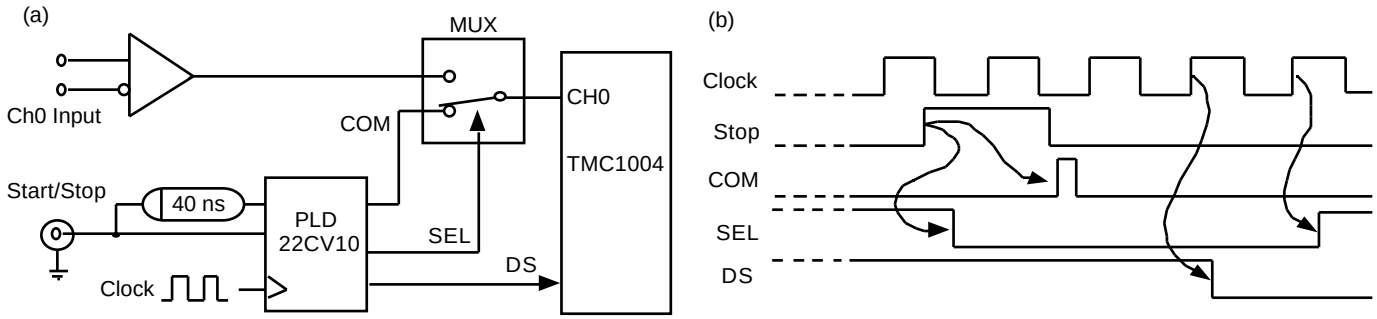

Fig. 4 (a) Schematics of an input multiplexer for channel 0 and start/stop signals. (b) Timing diagram of the channel 0 multiplexer in the common stop mode.

and data readout can be carried out simultaneously. Then, the TMC output is continuously written to the FIFO so as to extend the time range. Three FIFO's are used to cover 4 channels. There is another kind of FIFO, called the FIFOR, which records a 3-bit event number and the 5-bit row number (RC) of the TMC. The FIFOR is used to distinguish events in the multi-event mode and to know the actual row number of the TMC.

There is a multiplexer at the input of channel 0. It selects the Ch0 input signal or the common start/stop signals. The common start (stop) signal is recorded as the first (last) two data of channel 0. Schematics and a timing diagram in the common stop mode are shown in Fig. 4. 'DS' is a signal used to initiate a data movement from the TMC to the FIFO. A sequencer and some random logic for the start/stop control are implemented in the PLD's. An external signal is recorded in the TMC until a stop signal arrives. When the stop signal arrives, the multiplexer is changed to select the stop signal. A signal 'COM' is generated from the stop signal and is then recorded as the last two data. This signal is delayed by 40 ns with an external delay line and is clipped to have a narrow width. After recording the 'COM' signal, the 'DS' is negated to stop the data-taking.

In the common start mode, the sequence is similar but the common start signal is recorded first. Although the recording to the FIFO starts after receiving the common start signal, we start the TMC before that. This enables us to record input signals which arrive at the same (or earlier) time as the common start signal.

There is a 16-bit counter which is used for counting the recording time (number of cycles). In the common start mode, the counter counts the number of recorded cycles. The data acquisition stops when the count reaches zero. In the common stop mode, data-taking begins by the start signal or a DSP

command. After spending the preset number of cycles, an unload clock is generated to discard any unnecessary data from the FIFO. This cycle continues until the stop signal arrives.

Table 2 shows the bit assignment of the module CSR register (MCSR). The WSTRT bit starts the data recording into the TMC chip. This does not initiate writing to the FIFO. FIFO writing begins by the DSTRT bit or the external start signal. Bits 4 through 7 indicate the status of the FIFO. Since all the FIFO's run synchronously, only the status of the first FIFO is indicated. The OVWEN bit is used to control the unload clock of the FIFO. Bits 9 and 10 reset the FIFO and the TMC respectively. Bit 11 is used for debugging purposes.

Table 2. Module CSR (MCSR) Bit Assignment

Bit	Name	R/W	Contents
0	WSTRT	R/W	Start TMC
1	DSTRT	R/W	Start FIFO Recording
2	DSTOP	R/W	Stop FIFO Recording
3	CSTSP	R/W	Common Start (=1) / STOP (=0) mode
4	EMPTY	R	FIFO Empty Flag
5	AFAE	R	FIFO Almost Full/Empty Flag
6	HFULL	R	FIFO Half Full Flag
7	FULL	R	FIFO Full Flag
8	OVWEN	R/W	FIFO Overwrite Enable
9	FRST	R/W	FIFO Reset
10	TRST	R/W	TMC Reset
11	DBG	R/W	Debug Status read / Indicator write

Table 3. VME CSR (VCSR) Bit Assignment

Bit	Name	from VME	from DSP	Contents
0	MRST	W	W	Reset module
1	INTM	R/W	R/W	INTM status
2	INTMEN	R/W	R/W	INTM Interrupt Enable

VME Addr	15 Upper byte 8 7 Lower byte 0
x+0 : x+3FFE	Dual Port Memory
:	-
x+FF00	| VCSR .
:	-
x+FFF0	| Interrupt Control Register
x+FFF2	| Command Vector Reg.
x+FFF4	| Interrupt Status Register
x+FFF6	(Host Port) | Interrupt Vector Register
x+FFF8	| -
x+FFFA	| Rx/Tx Byte Register H
x+FFFC	| Rx/Tx Byte Register M
x+FFFE	| Rx/Tx Byte Register L

Fig. 5 VME space address map. The module occupies a 64 k byte address space. Base address 'x' is set by switches for the upper 8-bit address (A16 to A23).

C. DSP and Interrupt Control

The expansion bus timing for the P-, X-, Y-, and I/O space can be programmed individually in the DSP56002. In this module, all the memory, except for the I/O space and ROM, can be accessed without a wait cycle. One wait cycle is required for access to the I/O space, and several wait cycles for accessing the ROM.

A crystal oscillator of 62.5 MHz is used to generate a 31.25 MHz symmetric clock. This clock is distributed within the module through clock buffers and is terminated by a 73 W to 2 V Thévenin equivalent terminator to minimize any signal reflection. The high-speed signal lines are carefully routed to operate the module at 31.25 MHz stably.

There are several interrupt sources to the DSP. In addition to the interrupt created within the DSP chip (stack error, software interrupt, etc.), the serial port and the host port can cause interrupts to the DSP.

For an interrupt to the VME, there is an interrupt signal (INTM) which is controlled by bits 1 and 2 of the VME CSR register (VCSR). This interrupt is provided to communicate with the VME master for handling event data. The interrupt vector of the module interrupt is written in the last 8 locations of the DPM.

Table 3 shows the bit assignment of the VCSR. A reset signal is asserted by setting the MRST bit of the VCSR. This register can be accessed from both DSP and VME.

The host port can also generate a VME interrupt from the DSP. The interrupt vector is set in a host port register.

D. VME Interface

The VME interface has A24/D16 address/data size. The base address of the module is set by switches for the upper 8-bit address (A16-A23). Thus, the module uses a 16-bit address space. An address map of the VME space is shown in Fig. 5.

The lower 16 k byte is assigned to the DPM and the higher 16 addresses are used for the host port registers. Address of the VCSR is assigned $FF00. Since the power pins assigned in the VME specification were not sufficient for our purpose, we assigned additional power of 5.0 V, 3.3 V, -5.2 V and GND to pins of the P2-A and C rows by referencing the VXI standard.

III. DSP SOFTWARE

We have used a cross assembler of Motorola to develop DSP programs. The assembler runs on a Macintosh or an IBM PC, and produces object module format (OMF) files. These files are downloaded to the TMC-VME module through the

serial or the host port controlled with the DSP monitor program.

A. Monitor Program

A monitor program called "TMCBUG" has been developed based on "DSPBUG" [4] which was developed by Motorola Inc. The monitor program handles read(write) from(to) memory and registers, sets breakpoints, traces, downloads programs, and handles terminal and host port I/O's. The TMCBUG also supports user-accessible subroutines. The TMCBUG is programmed in a ROM, and a bootstrap program moves the main codes from the ROM to the DSPRAM at the startup time. To minimize the required RAM area, several unimportant commands are removed and a program overlay technique is also used. The program overlay loads only the requested command codes in the RAM at the execution time. Several custom commands used to check the contents of FIFO's and TMC's have been added to ease the debugging. The present "TMCBUG" occupies only 3 k words of the program RAM space.

B. Data Acquisition Program

A common data-acquisition program has been developed. The program runs in both common start and common stop modes. Although the module can handle multiple events, the present program does not yet support such a function. The program obtains experimental parameters from the terminal or DPM and waits for an event. When an event is available, the DSP searches for the Start/Stop timing from the Ch0 data. Then, the DSP searches for the transition point from the other data. If the DSP finds a transition point, the DSP writes the data to the DPM. Optionally the user can subtract any offset values channel by channel. At last, the DSP sets a flag (or causes an interrupt) to notify the VME master.

IV. PERFORMANCE

A. Resolution

The timing resolution of the TMC chip is 0.52 ns for a single hit [1]. Since we need both the start/stop time and the hit time to measure the detector signal timing, the timing resolution is cumulative of the two measurement errors. Figure 6 shows the timing resolution of the TMC-VME module for measuring the time between a common start/stop signal and a hit. The error includes a digitization error of two data (s = 0.4 ns). Although the distribution is not pure Gaussian, we obtain a timing resolution of σ = 0.72 ns from the fit of the data to a Gaussian distribution.

B. Data Processing Time

The data-processing speed depends not only on the hardware, but also on the software. Especially, in the DSP program, the level of the software affects its execution time. The data-processing time becomes shorter and shorter as we become accustomed to its programming. Thus, we can discuss only the present performance here. We expect that a better performance will be obtained by carefully designing the program.

Fig. 6 Time measurement error for a measurement of the time between a common start/stop and a hit signal.

We locate most of the time-critical routines in internal P-RAM to avoid any access to the external bus. The data-taking speed depends on the time range and number of hits. The processing time was measured and can be expressed as follows:

$$\text{Time} \approx 0.45\ \mu\text{s} \times (\#\text{Ch}) \times (\#\text{Rows}) + 5\ \mu\text{s} \times (\#\text{Hits})$$

One row corresponds to a 32 ns time range. Since one DSP instruction cycle is 64 ns, we are using 7 instruction cycles for processing one row of data and about 78 cycles for a hit.

For example, if we handle 64 channels, a 4 μsec (128 rows) time range, and each channel has one hit, the data taking time will be about 4 msec.

ACKNOWLEDGMENTS

We acknowledge T. K. Ohska and T. Kondo for their continuing support. We also thank to the members of SDC straw and muon group for their encouragements.

REFERENCES

[1] Y. Arai and T. Matsumura and K. Endo; "A CMOS 4 ch x 1 k Time Memory LSI with 1 ns/bit Resolution", IEEE Journal of Solid-State Circuits, Vol.27, No.3, March 1992, p359-364.

[2] Y. Arai, F. Sudo and T. Emura; "Developments of Time Memory Cell VLSI's", to be published in proceedings of Third Annual Conference on Electronics for Future Colliders. May, 1993. LeCroy Corporation.

[3] Y. Arai, M. Ikeno and T. Matsumura; "Development of a CMOS Time Memory Cell VLSI and a CAMAC Module with 0.5 ns Resolution", IEEE Trans. on Nucl. Sci. Vol. 39, No. 4 (1992)784-788.

[4] DSPBUG version 2. DSPBUG is a Motorola debugger for the DSP56000 series.