

CERN 91-05
14 May 1991

ORGANISATION EUROPÉENNE POUR LA RECHERCHE NUCLÉAIRE
CERN EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

1990 CERN SCHOOL OF COMPUTING

Ysermonde, Belgium
2-15 September 1990

PROCEEDINGS

GENEVA
1991

CONTENTS

Preface	V
Advisory Committee	VI
On LANs and MANs: an evolution from Mbit/s to Gbit/s, <i>P. Zafiropulo</i>	1
ATM: solution for B-ISDN, <i>M. De Prycker</i>	9
Standards in computer networking, <i>P. Van Binst</i>	47
An introduction to transputers and OCCAM, <i>R.W. Dobinson, D.R.N. Jeffery and I.M. Willers</i>	52
New developments in program verification, <i>P. Wolper</i>	91
Unix: evolution towards distributed systems, <i>E. Milgrom</i>	97
Computer security within the CERN environment, <i>D.H. Lord</i>	127
Numerical simulation in fluid mechanics, <i>H. Oertel Jr.</i>	131
Digital optical computing, <i>S.D. Smith</i>	167
Introduction to digital image processing, <i>M. Kunt</i>	211
Digital signal processing in high energy physics, <i>D. Crosetto</i>	263
Data acquisition using the 'MODEL' software, <i>P. Vande Vyvre</i>	293
Graphical event analysis, <i>D. Bertrand</i>	316
Simulation of natural fractals by IFS, DLA and L-string systems, <i>R.F. Churchouse</i>	339
List of participants	351

DIGITAL SIGNAL PROCESSING IN HIGH ENERGY PHYSICS

Dario Crosetto(*)

CERN, Geneva, Switzerland

1. SUMMARY

New uses are continually appearing for the type of microprocessor known as the Digital Signal Processor, originally conceived for efficient execution of signal processing algorithms in both the time (convolution) and frequency (FFT) domains.

After a brief introduction to signal analysis and processing, the principal characteristics of DSP's are described and compared with other commercially available microprocessors. In particular, emphasis is given to those specialized instructions which yield the above efficiency.

A survey of applications of the DSP relevant to High Energy Physics is then made in the fields of Accelerator Control and in data acquisition, including the rather complex triggering operations.

2. SIGNAL ANALYSIS.

2.1. Signal Classification.

2.1.1 Describing the signal.

Signal analysis in its widest application simply means the extraction of useful information from input data. Input data can be, by nature, either continuous (temperature) or discrete (number of events) as a function of some independent either continuous (time) or discrete (channel number) variable. According to the methods of analysis chosen, transitions to/from discrete representations are frequently effected. In the narrower sense specific to the analysis of electrical signals, the input will usually be a continuous time variable. The transition from continuous to discrete is then called analog-to-digital conversion (ADC) and the inverse transition is called digital-to-analog conversion (DAC). Quantization of the input signal when done will be effected at regular intervals of the independent variable.

Signals can be classified as continuous or discrete; deterministic or stochastic. In the particular case of signals arising from HEP detectors, these signals can be classified as stochastic with an independent variable (continuous or discrete) in the time or frequency domain and a dependent variable in continuous or discrete form.

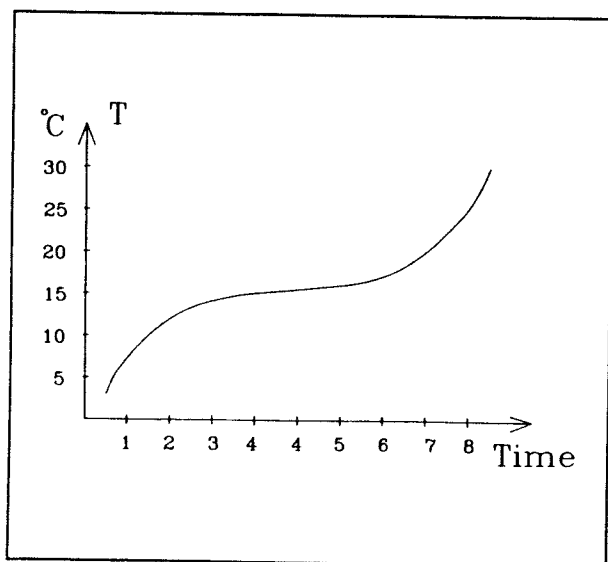


Figure 1a. Example of a signal with a continuous independent variable (time) and a continuous dependent variable (temperature).

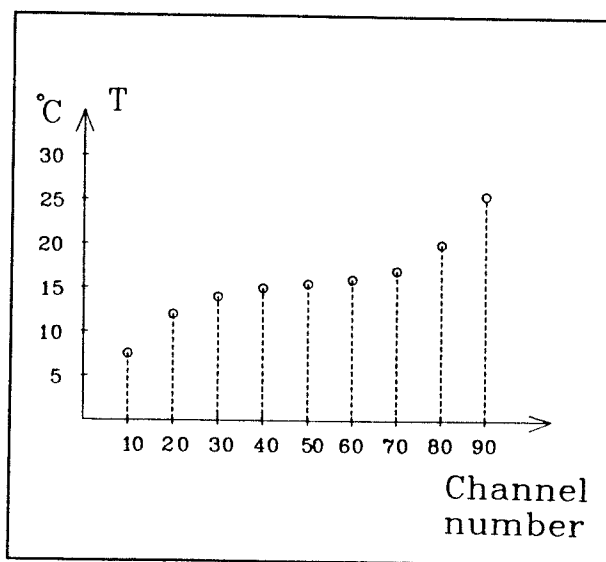


Figure 1b. Example of a signal with a discrete independent variable (channel number) with a continuous dependent variable (temperature).

(*) present address: INFN - Torino, Italy

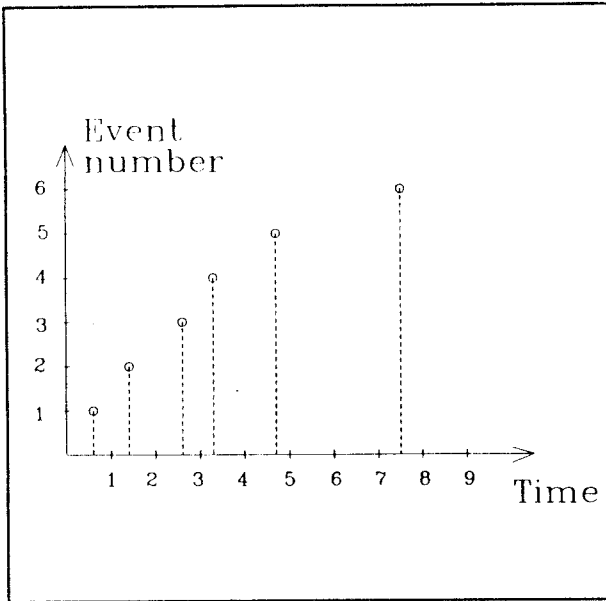


Figure 1c. Example of a signal with a continuous independent variable (time) with a discrete dependent variable (number of events).

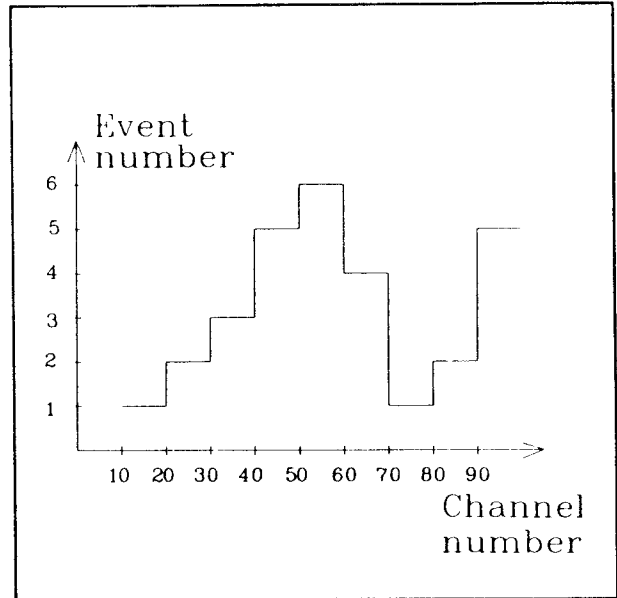


Figure 1d. Example of a signal with a discrete independent variable (channel number) with a dependent (discrete) variable (N events).

Figure 2a illustrates the transformation of a continuous time variable signal in the Digital Signal Processing phases. Figure 2b shows the different units used for these transformations.

- The continuous time variable input signal is first transformed into a Pulse Amplitude Modulated (PAM) version by the Sampling and Hold (S/H) (independent variable = discrete, dependent variable = continuous),
- It is then transformed into a Pulse Code Modulated signal (PCM) by the Analog-to-Digital converter (A/D) (independent variable = discrete, dependent variable = discrete),
- then the signal is processed digitally by the Digital Signal Processor (DSP). Discrete results are then transformed by Digital-to-Analog converter (DAC) in analog signals (independent variable = discrete, dependent variable = continuous),
- finally the analog signal is filtered (independent variable = continuous, dependent variable = continuous).

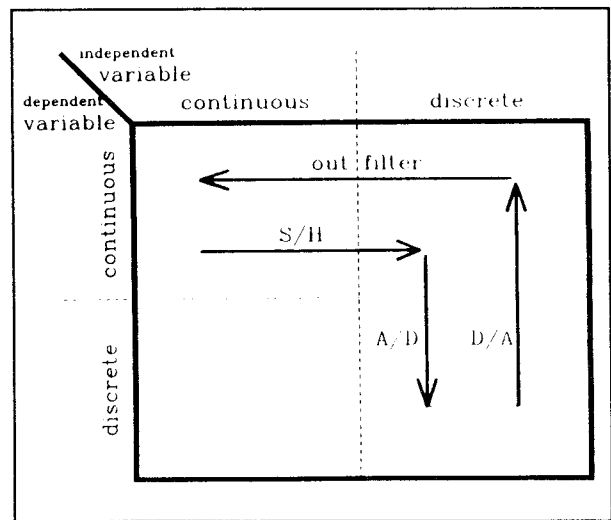


Figure 2a. Transformations of the independent and dependent variables in the Digital Signal Processing phases.

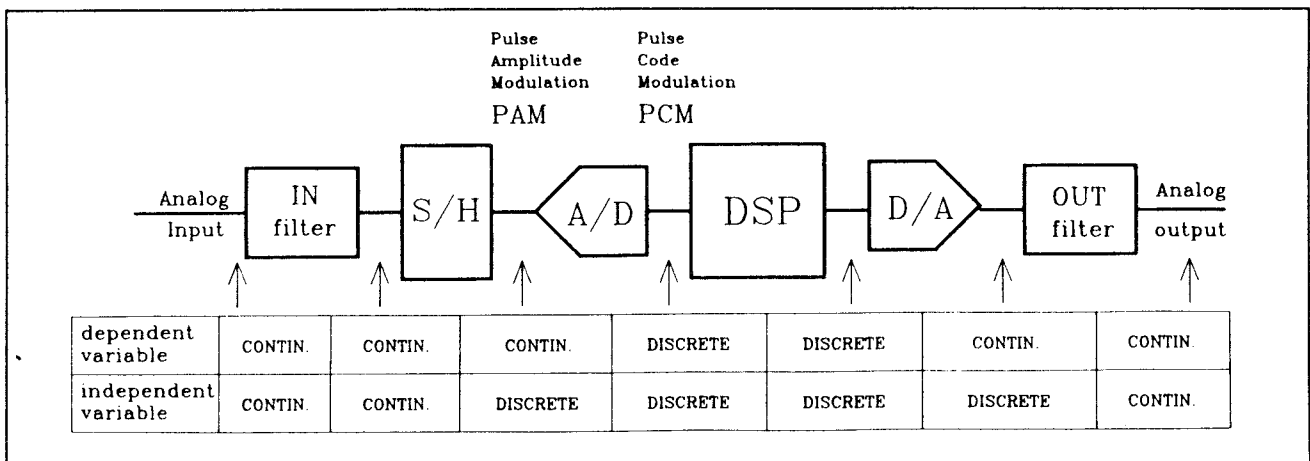


Figure 2b. Block diagram of the units used in a digital processing of analog signals.

2.1.2. Time and frequency domain.

Signal processing requires thinking in two domains, the time domain and the frequency domain. A signal can be described either by its waveform as a function of time $f(t)$ (time domain description) or as a spectrum of amplitudes and phases for each frequency component. The two descriptions are equally valid and either is chosen on the basis of computational convenience.

2.1.3. Deterministic and stochastic Signals.

Signals can be further classified as deterministic or stochastic. The former are useful for testing equipment but do not carry information while the latter carry information. By stochastic we mean here that the characteristic of the signal can only be described by its statistical properties within known limits.

2.2. Linear and non-linear systems.

Analysis and processing of time varying signals is a well established scientific discipline covered in many textbooks [1-10]

In this note I do not intend to present the basic concepts of this discipline, such as convolution, difference equations, continuous and discrete time Fourier transforms, Laplace Transforms, and z-transforms, which can be found in the text books referenced.

The theory of signal analysis is normally divided between linear and non-linear processing. The simpler and better known theory is the linear part which is representable as linear differential (continuous) or difference (discrete) equations with constant coefficients. The main practical consequences of linear processing are two:

1. The output of a sum of inputs is equal to the sum of the outputs due to each input singularly applied.
2. Output frequency components can be modified in amplitude and phase, but no new frequency components can be generated.

An example of a non-linear process is a rectifier which due to its abrupt current cutoff introduces new frequencies with respect to the input spectrum.

In what follows only linear processing is implied.

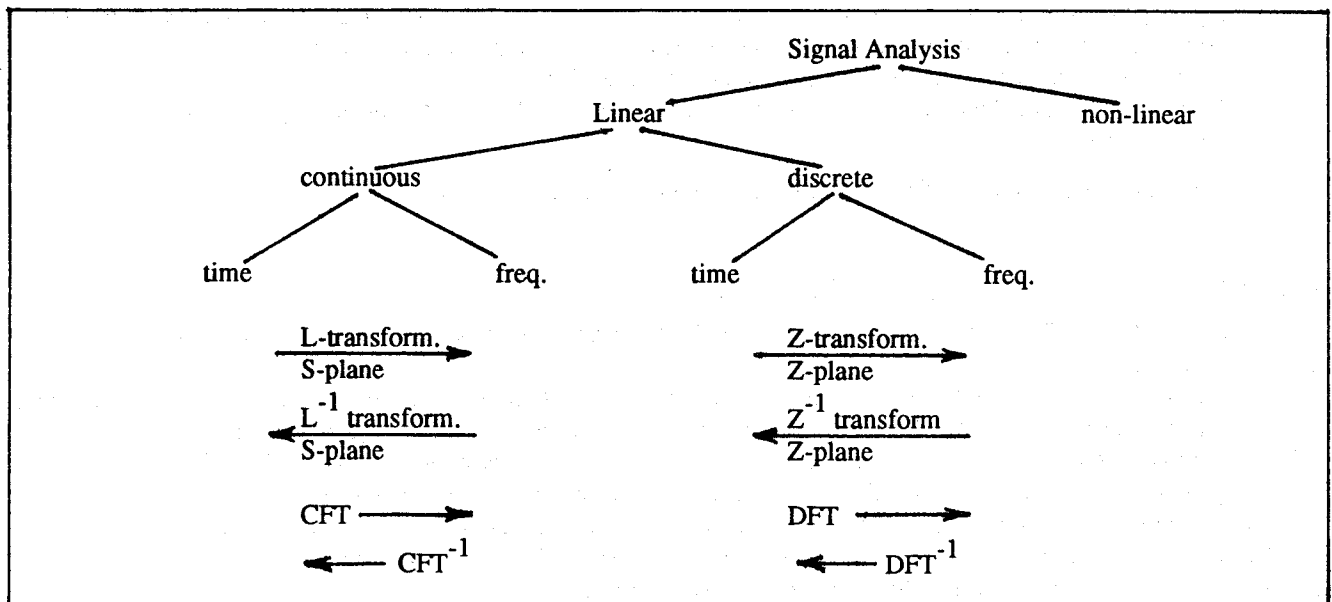


Figure 3. Signal analysis.

2.3. Linear Systems.

2.3.1. Signal representation and transformation from time to frequency domain.

There are representations and transformations which are adapted to the type of signal (continuous or discrete). For the continuous signals the best known transformation from time to frequency and viceversa is the Laplace transform, coupled with the frequency spectrum representation in the complex s-plane. The equivalent for discrete signals is the z-transform and the frequency spectrum representation in the complex z-plane.

Another transform which is widely used and well suited for numerical calculations is the Fourier transform. Thus we have continuous (CFT) and discrete (DFT) Fourier transforms which are limited to real frequencies. The latter can be manipulated so as to provide a very rapid evaluation, known as the Fast Fourier Transform.

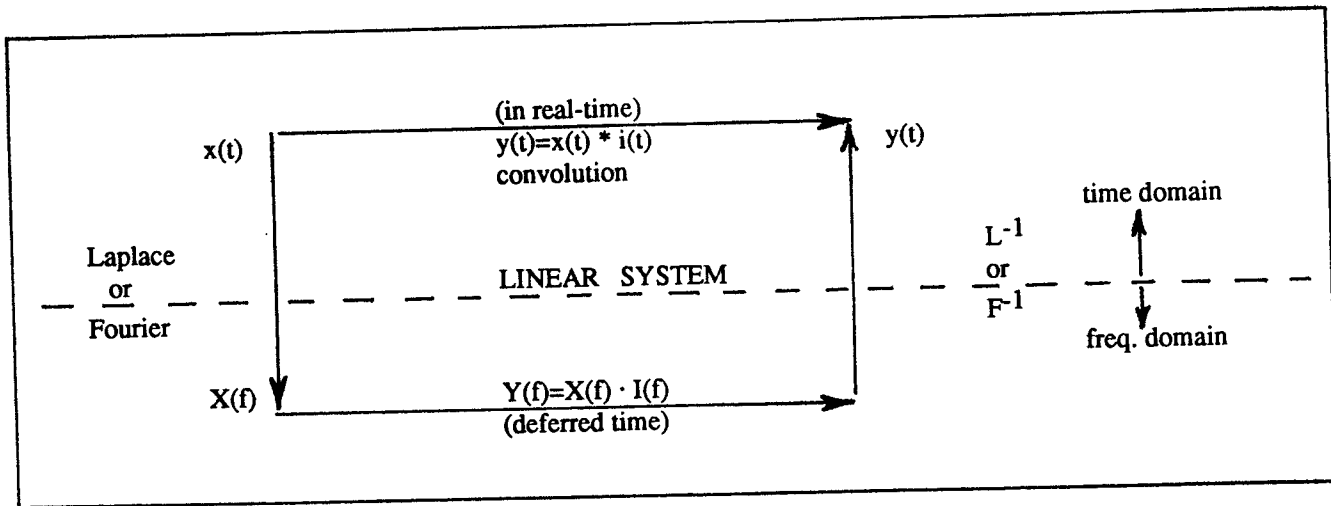


Figure 4. Signal relation between time and frequency domain.

The above figure is very useful in understanding the relation between time and frequency domain processing. It is based on the convolution theorem which states that

- if $X(f)$ is the transform of $x(t)$ and $I(f)$ (transfer function) is the transform of $i(t)$ (impulse response)
- then the product of $X(f)$ times $I(f)$ is the transform of the convolution of $x(t)$ with $i(t)$ {written $x(t) * i(t)$ }.

2.3.2. Signal processing in the time and frequency domain.

Signal processing in the time domain implies convolution of the input $x(t)$ with the impulse response $i(t)$ of the linear system. To effect the same operation in the frequency domain requires three steps.

- 1) transformation from time to frequency of the input signal (FFT).
- 2) multiplication of the input spectrum by the transfer function (both functions of frequency) to obtain the resultant spectrum
- 3) reverse transformation of the resultant spectrum to obtain the resultant waveform (FFT⁻¹).

3. DIGITAL SIGNAL PROCESSING.

Digital Signal Processing is the analysis and transformation of sampled, discrete input signals to yield discrete useful results.

DSP can be effected either in the time or in the frequency domain. In the time domain, there is one output result for each input sample. In the frequency domain, there is at least one output sequence for each input sequence. This distinction implies that time domain processing is in real time (prompt response) while frequency domain processing is in deferred time because the input sequence must be accumulated before the analysis can begin. DSP processing in the time domain is usually described in terms of digital filters characterized by their response to a single impulse; finite impulse response (FIR) or infinite impulse response (IIR). DSP processing in the frequency domain is usually described through transforms (discrete or fast Fourier) or special operations.

3.1. Digital versus analog signal processing.

With the advent of Fast A/D converters and new processors oriented towards signal processing (DSP), there arose the tendency to treat analog signals in digital form, thus using discrete algorithms instead of analog functions. The advantages of the digital versus analog processing are principally perfect stability (no drift due to temperature or aging), repeatability (not dependent on component tolerance) easy design (programming an algorithm), lower cost through programming of the same devices for different functions, no calibration is needed, accuracy is limited only by converter resolutions and processor arithmetic precision (word size and presence or absence of floating point), low power consumption (CMOS), testability and high circuit density. In contrasts upper speed limits of DSP are inferior to those of analog processing and are determined by the clock cycle of the processor.

3.2. Digital Filters.

$$\text{output}(n) = \sum_{i=0}^k a_i \cdot \text{input}(n-i) + \sum_{j=1}^l b_j \cdot \text{output}(n-j)$$

1 tap = ideal amplifier (=one coefficient)

3.2.1. Characteristics of FIR Filters

- Stability (no feedback) zero relative phase distortion possible through symmetry.
- Introduce more delay than IIR.
- Powerful and simple design using the Kaiser-Window Method [8]. In this method, an ideal frequency response is defined, the corresponding ideal impulse response is computed and then truncated to a finite number of non zero samples weighted with the window function.

3.2.2. Characteristics of the IIR filters

- more efficient than FIR filters in frequency cutoff
- introduce relative phase distortion
- the feed-back can introduce instability. One reason of instability is the roundoff of the "b" coefficients.

3.2.3. Coefficient quantization.

Most digital filter design algorithms are implemented with floating point arithmetic and the resulting filter coefficients are generally obtained as floating point numbers. When a filter is to be implemented using fixed point arithmetic it is necessary to represent the filter coefficients as fixed-point binary numbers, thus introducing roundoff and possible instability.

Various software packages for digital filter design are available from: ATLANTA, HYPERCEPTION, FDAS, DISPRO, BURR-BROWN, etc.

3.3. The FFT and spectral estimation

The fast Fourier transform (FFT) is an algorithm for computing the discrete Fourier Transform (DFT). The FFT algorithm provides a significant reduction in the number of arithmetic operations compared to the straight forward DFT.

The DFT of an N-point finite input sequence results in another N-point finite spectrum sequence of harmonically related frequency components.

In order to find the spectrum of a sampled data signal, it is necessary to compute values of a function of the form (DFT) [4]:

$$X_k = \sum_{n=0}^{N-1} x(n)e^{-j(2\pi/N)nk} \quad k = 0, 1, \dots, N-1 \quad (3.31)$$

It is common to denote the complex exponential $e^{-j(2\pi/N)nk}$ of that equation as W_N^{nk} . Thus the equation (3.31) can be written as:

$$X_k = \sum_{n=0}^{N-1} x(n)W_N^{nk}, \quad k = 0, 1, \dots, N-1 \quad (3.32)$$

Evaluation of equation (3.32) requires N^2 complex multiplications.

3.3.1. Butterfly diagram for reverse-order input

The object of the FFT algorithm is to reduce the computational complexity.

If the time domain sequence $x(n)$ is reordered (decimated) in odd and even samples, and due to the symmetry of the factor W_N ($W_N^k = -W_N^{k+N/2}$, $k=0,1, \dots,(N/2)-1$), the equation (3.32) can be written as:

$$X_k = X_e(k) + W_N^k X_o(k) \quad k=0, 1, \dots,(N/2)-1 \quad (3.33)$$

$$X_{k+N/2} = X_e(k) - W_N^k X_o(k), \quad k=0, 1, \dots,(N/2)-1 \quad (3.34)$$

Where X_e and X_o are of the form of equation (3.32), over a selective set of indices.

These equations known as "decimation in time butterfly" can be represented graphically as a butterfly of Figure 5.

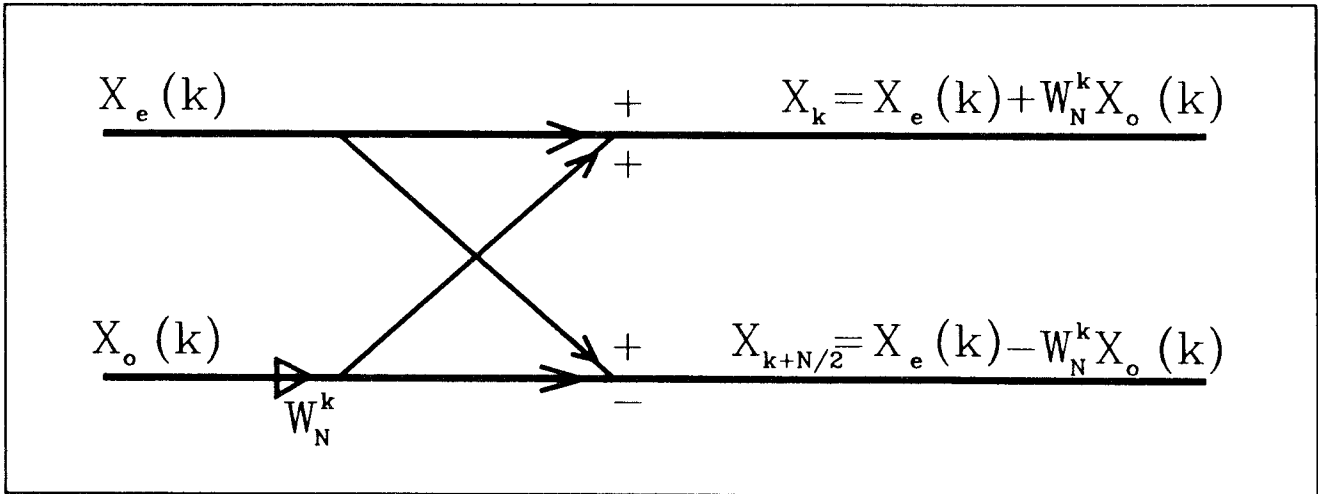


Figure 5. Basic butterfly operation for a decimation in time FFT.

The computational macro that requires one complex multiplication and two complex additions is in general executed in one, or very few cycles, by the Digital Signal Processors.

With this method of computation, the original N point time sequence DFT is reordered into two $N/2$ point sequences.

As X_e and X_o are of the form of (3.32), the computation of X_e and X_o can be accomplished in a similar manner by further decimating the time domain sequence into four $N/4$ point sequence and so on.

This decomposition process is continued until the $N/2$ two-point DFTs are computed directly from $N/2$ two-point sequences, each consisting of two samples of the original sequence spaced $N/2$ samples apart.

- As a consequence the input data must be presented in bit-reversed order.
- The number of passes required in an FFT is $\log_2 N$.

For more details see [4], chapter 4.

Table 1.

NATURAL ORDER				BIT-REVERSE ORDER			
			bit 0				bit 0
0	0	0	0	0	0	0	0
1	0	0	1	4	1	0	0
2	0	1	0	2	0	1	0
3	0	1	1	6	1	1	0
4	1	0	0	1	0	0	1
5	1	0	1	5	1	0	1
6	1	1	0	3	0	1	1
7	1	1	1	7	1	1	1

- A complete flow graph for an 8-point radix-two decimation in time FFT is illustrated in Figure 6.

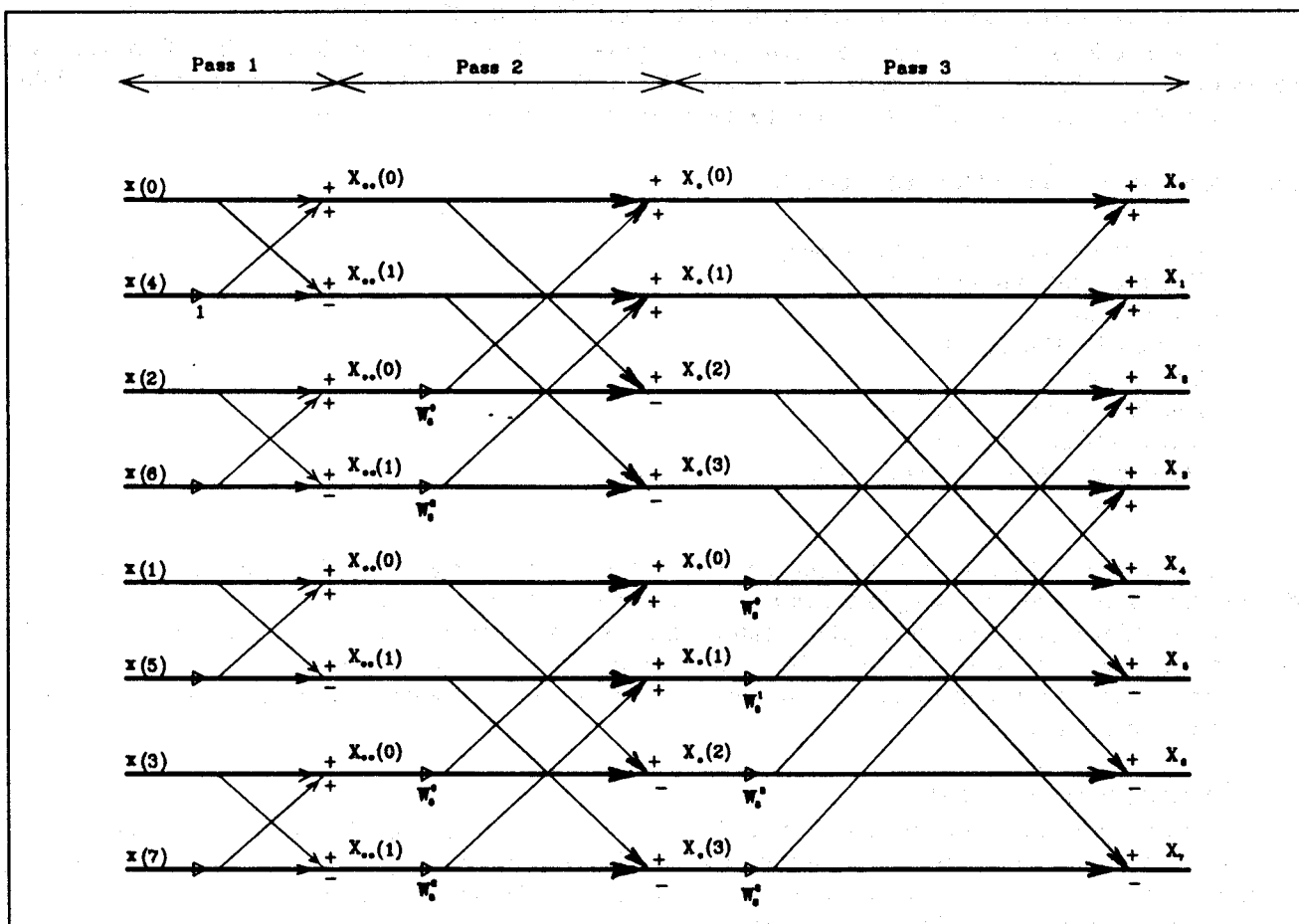


Figure 6. Radix 2 decimation in time, in place 8-point FFT bit reversed input; normal output.

4. DIGITAL SIGNAL PROCESSORS (DSP).

Digital Signal Processors are special purpose microprocessors optimized for the execution of digital signal algorithms. They are traditionally designed for performance, not for extensive functionality nor programmer convenience.

4.1. Historical evolution of DSP.

- 1st DSP	1978	AMI S2811
-	1979	INTEL 2920/21 (Telecommunication)
-	1979	Bell Labs DSP1 (never marketed outside AT&T)
-	1980	NEC uPD7720
-	1980	Analog Devices ADSP-2100
-	1981	Hermes (not marketed outside IBM)
-	1982	Hitachi 61810
-	1982	Texas 32010

The last line represents a great widening of the applicability spectrum due to external reprogrammability, ideal for low volume applications.

In the beginning most DSP's were distinguishable from other microprocessors due to their characteristics of:

- 1) Harvard architecture (separation between Program and Data memories)
- 2) internal and very small Program and Data memory area
- 3) small instruction sets, and mostly executable in one cycle (for this reasons similar to RISC)
- 4) special instructions for treatment of digital signals (such as: parallel multiply, barrel shifting, auxiliary registers for single cycle manipulation of data tables, etc.)

4.2. Characteristics of different chips for Real-Time processing.

In what follows we will compare different types of processors (microcontroller, RISC, CISC, Transputer and high performance embedded controllers) with the Digital Signal Processor, with the aim of assessing their suitability for various real-time applications. We first recall the basic elements of a real-time processor:

Basic elements of a real-time processor.

- ALU (one or more, for Addr and Data)
- Floating Point Unit (optional)
- Control Unit
- Program RAM or ROM
- Data RAM
- Parallel I/O Controller (DMA)
- Serial I/O Controller (DMA)
- (Analog-to-Digital and Digital-to-Analog Converters)
- Fast Interrupt

There are several ways to realize a concurrent system out of the basic elements listed above, each one having a different throughput, and privileging in one case one aspect with respect to another.

In selecting a processor for a certain application it is very important to know the characteristics of all the processors which would possibly solve the problems, in order to make a balanced judgement.

After performance comparison with other processors it will become clearer why the DSP is more suitable for several types of applications.

4.2.1. Characteristics of DSPs.

In recent years the characteristics of the DSP's have improved very rapidly. Not a single feature of the past was dropped (hardware multiplier, special instructions, etc.) but in addition today's DSPs use extensive pipelining, several independent memories with large address capability, parallel function units (one cycle floating point instruction), and hardwired control (not microprogrammed).

Also applications in this field are increasing so rapidly that at present a classification among the hardware of the DSPs must be made (section 4.4).

For the comparison with other types of processors, we select among the several DSPs commercially available today one from the "General Purpose DSP family": the Motorola DSP96000

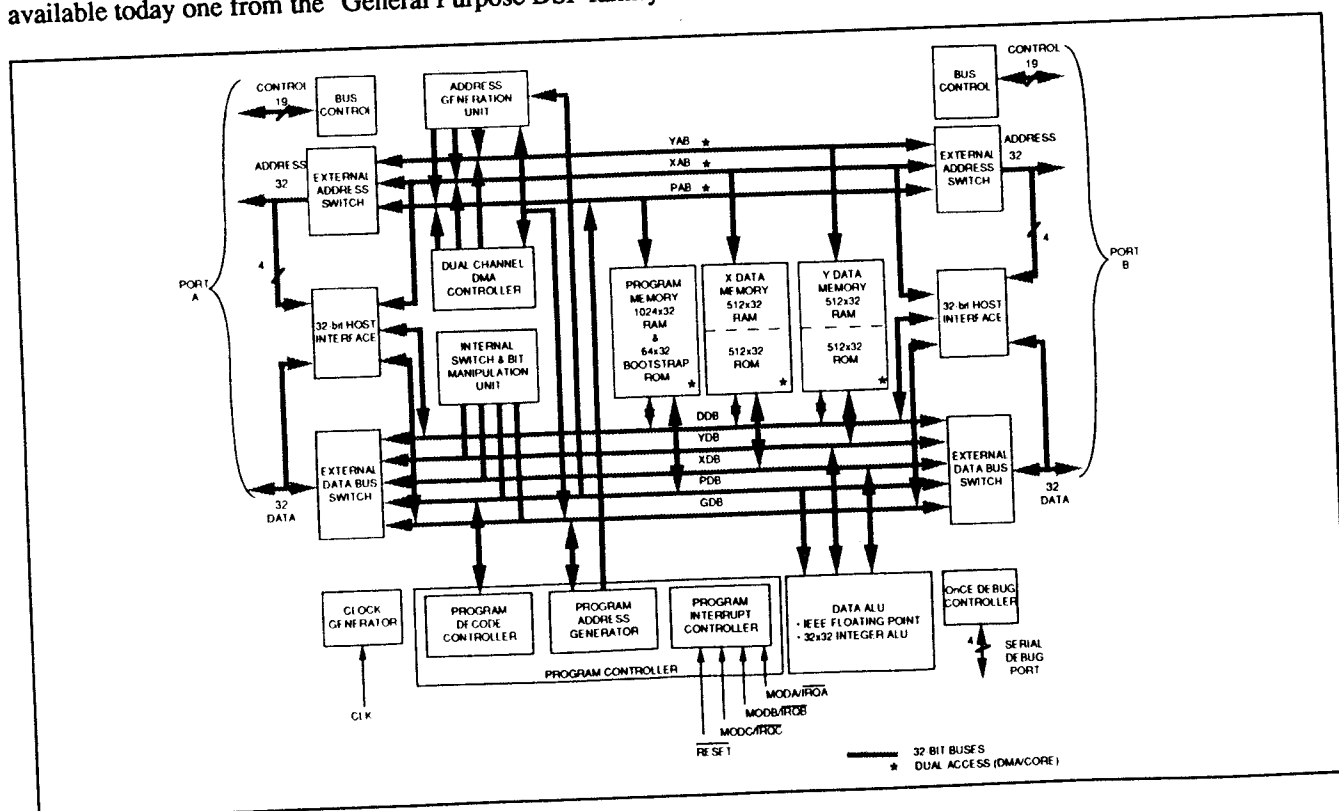


Figure 7. DSP96000 Block Diagram (courtesy of Motorola).

Some of the characteristics that make DSP's particularly suitable to treat discrete signals are found in its instruction set.

Several presently available DSP's have hardware "DO LOOP" instructions, have compare magnitude instructions, and/or can perform a simple operation $y = ax + b$ in one cycle (75 ns) while at the same time performing some operations on addresses by updating pointers. E.g. a single line of assembly code of the Motorola DSP96000 (ideal for Butterfly FFT calculation. See chapter 3.3),

FMPLY D4,D5,D0 FADDSUB.S D0,D1 X:(R0)>N0,D4 Y:(R4)>N4,D5

executed in a single cycle, will generate the results of multiplication, addition, subtraction and will update the pointers to the data in the memory.

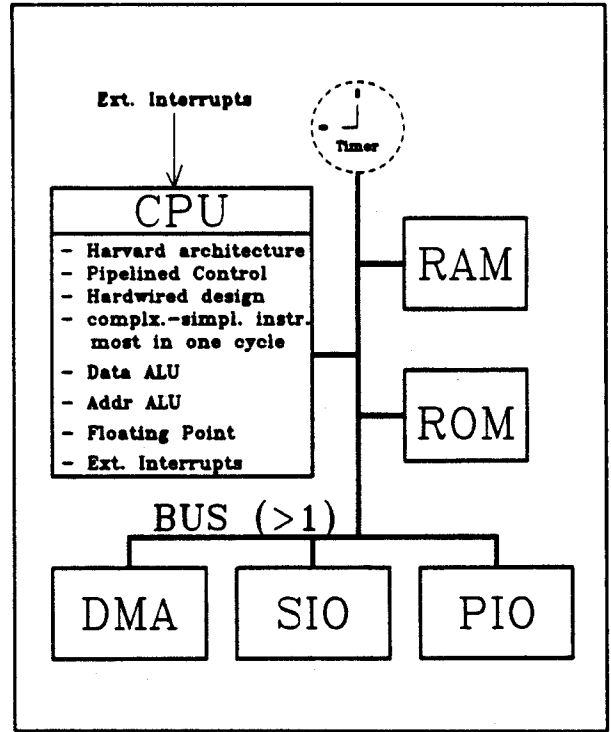


Figure 8. Simplified block diagram of a General Purpose DSP.

4.2.2. MICROCONTROLLERS versus DSP

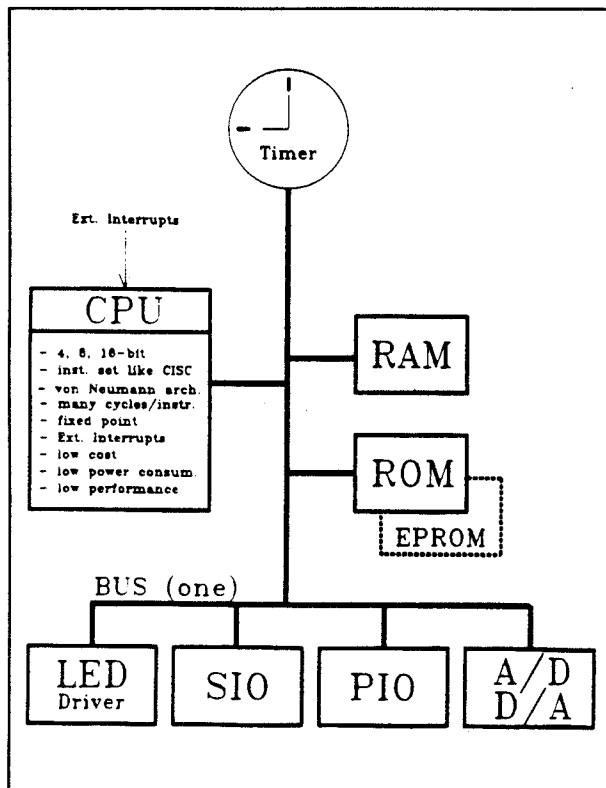


Figure 9. Simplified Block diagram of Microcontroller.

A "microcontroller" contains all the necessary components of a complete system on one piece of silicon (E.g. Intel 8051, Motorola MC6804, MC6805, MC68HC11, etc.).

The microcontroller has lower performance than a DSP, uses 4, 8, or 16-bit data, has an instruction set more like CISC processor (using more than one cycle per instruction). Some extra programmable peripherals on chip, like A/D converters are not available on DSP. The microcontroller is not designed to build multi-processor systems but it is intended to be used for economical applications in embedded systems where is only necessary to have the capability of one of the most common 8-bit or 16-bit microprocessor instruction sets.

Applications:

- industrial control
- device controller (printers, plotters, etc.)
- in an array of front end processors in a High Energy Physics Experiment for slow calculations

DSP is replacing this component in the more sophisticated applications where speed is an important factor.

4.2.3. TRANSPUTER versus DSP.

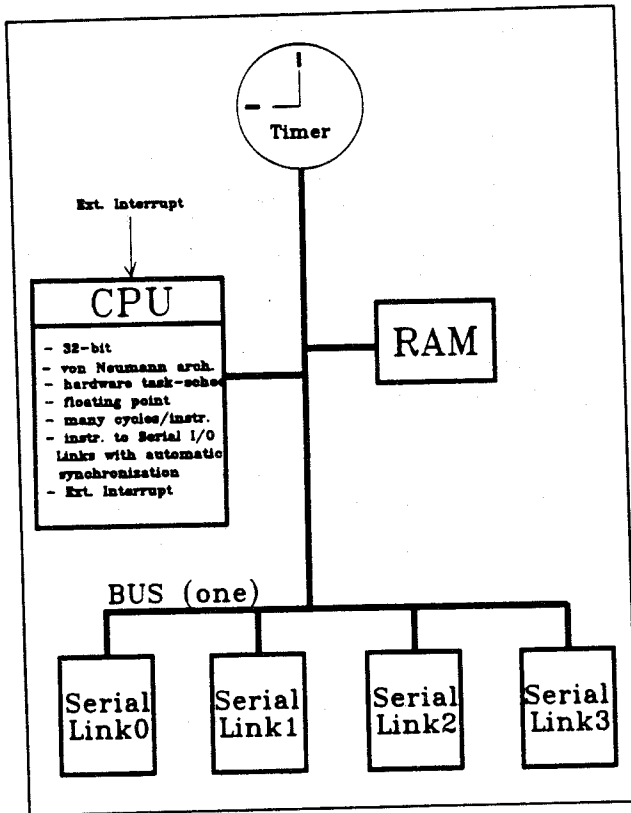


Figure 10. Simplified Block diagram of Transputer.

4.2.4. RISC versus DSP.

- Initial simple concept of a register-intensive cpu design came from Seymour Cray in 1960 for CDC 6600
- modern notion of RISC architectures emerged from John Cocke's project at IBM in 1970.
- Cocke's team goal was to design the best CPU architecture for an optimizing compiler; the machine should be register-to-register with only load and store accessing the memory, the architecture eliminated microcode and microsequencers in favour of simple, hardwired, pipelined, one-instruction-per-cycle CPU design.
- RISC technology created an almost insatiable demand for memory speed. The answer to this problem comes with high performance memory hierarchy, including general purpose registers and cache memories. The instruction set is regular and simple with few addressing modes: indexed and PC-relative [11, 12, 13].
- There are some RISC variations from the common theme.
- IBM (1975) with 801 minicomputer
- BERKELEY (1980) with RISC I and RISC II
- STANFORD (1981) with MIPS (Microprocessor Without Interlocked Pipeline Stages).

IBM and Stanford pushed the state of art in Compiler Technology to maximize the use of registers.

A Transputer contains in a single chip:

- an integer processor
- a Floating Point Unit
- 4 Kbyte of memory
- 4 high speed serial links (20 Mbit/sec)

The Transputer is designed as a programmable component to implement a system with much higher degree of parallelism than is currently common. The formal rules of the programming language Occam provide the design methodology for this family of concurrent systems. Special instructions divide the processor time between the concurrent processes, and perform interprocessor communication.

With the Transputer it is easier to build parallel systems because of the good coordination between hardware and software (Occam). It is easy to transport software between different concurrent systems with different numbers of transputers. The time required for a multiplication is $\sim 2 \mu\text{s}$ for a T414 (500 ns average for a T800). Most DSP's do it in one cycle (75 to 200 ns), while the same is true for the division and for the floating point operations.

DSP's have a performance of 20 to 40 Mflops, the T800 Transputer has 4.5 Mflops.

The BERKELEY team did not include compiler experts, so a hardware solution was implemented to keep register contents synchronized with a scoreboard register.

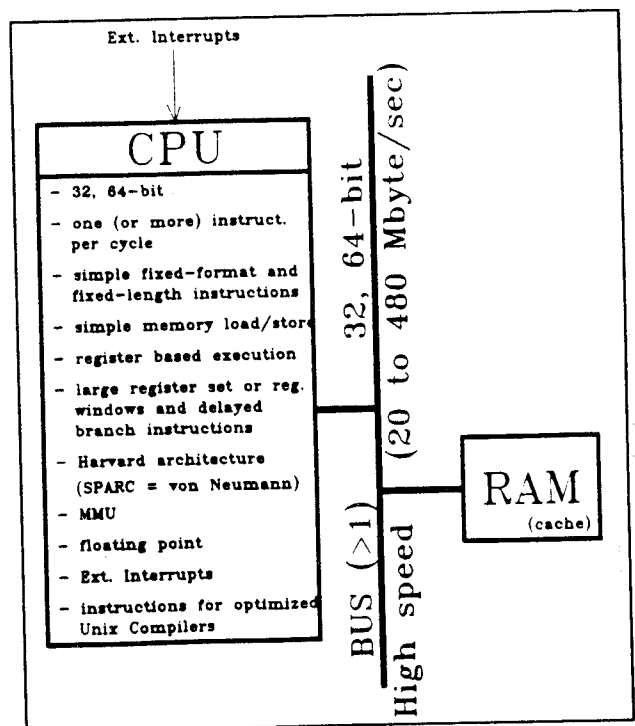


Figure 11. Simplified Block diagram of RISC.

To optimize the procedure calling time they defined many sets or windows of registers (global and local) so that registers would not have to be saved on every procedure call. The disadvantage of register windows is that they use more chip area.

The Fairchild Clipper now available from Intergraph Advanced Processor Division, was the first microprocessor design to recognize the need for improved memory bandwidth. Their solution was to separate the relentless demand of instruction fetches from the Load/Store activity by providing separate instruction and data buses.

The MC88000 from Motorola appears to be very linear. It follows the dogma of simple, one-cycle, fixed-length instructions and load/store architecture. The MC88000 is a dual bus, three-chip layout (CMOS). All the execution units work over the same two source buses and a single destination bus. The most important characteristics of the MC88000 may be the system's ability to incorporate new, specialized execution units.

Other RISC vendors:

- MIPS and SPARC from Fujitsu, Bipolar Integrated Technology, Cypress, LSI Performance Semiconductor, Device Technology. Acorn Sanyo for the VL86C010. Hewlett-Packard with the Apollo Domain 10.000. AMD 29000 family. HARRIS RTX2000 (highly integrated FORTH-executing microcontroller).

Table 2. Comparison between the most common RISC processors.

Type	VLSI chips set	inst. per cycle	Clock rate	Architecture	Regs	Comments
CLIPPER	3	1	33 Mhz	Harvard		8K icache, 8k dcache. On chip 2 x 64 TLB. (A Translation lookaside buffer, is the memory cache of the most recently used page table entries within the MMU).
SPARC	6	1	33 Mhz	Von Neumann	<250	Register windows and delayed branch instr. Need simpler compilers. Supports the Big-endian format.
R3000	2	1	33Mhz	Harvard	48	64k icache, 64k dcache. On chip 64 TLB. Clever Compilers. Microprocessor Without Interlocked Pipeline Stages.
2900	3	1	25 Mhz	Harvard	195	No Direct Cache support, branch cache On chip 64 TLB.
88000	3	1	20 Mhz	Harvard	32	Special bit field instructions Division. SQRT
i860	1	3	30 Mhz	Harward	64	Special instr. for graphic proces, 3D graphic unit MMU (4 Gbyte), FCU, FMU,4kb icache, 8kb dcache.Support byte ordering formats.
IBM RISC	7	5	30 Mhz	Harvard		CPU = 3 processors (ICU, FXU, FPU) Bus transfer rate up to 480 Mbyte/s.

4.2.5. CISC (CRISP) versus DSP

- CISC (Complex Instruction Set Computer) architecture uses a large amount of hardware complexity to provide a high degree of instruction set capability. They are characterized by a large instruction set with some very complex instructions. The length and execution time of instruction can be different from one another. Instructions can manipulate bit, byte, word and long word. The dynamic bus interface allows for simple, highly efficient access to devices of different data bus width. The latest components of this technology support, directly via BUS Monitoring, Multimaster and Multiprocessor applications.

Advantages and Disadvantages

- Some instructions need more then 50 cycles to be executed.
On the other hand, DSP executes most instructions in one cycle.

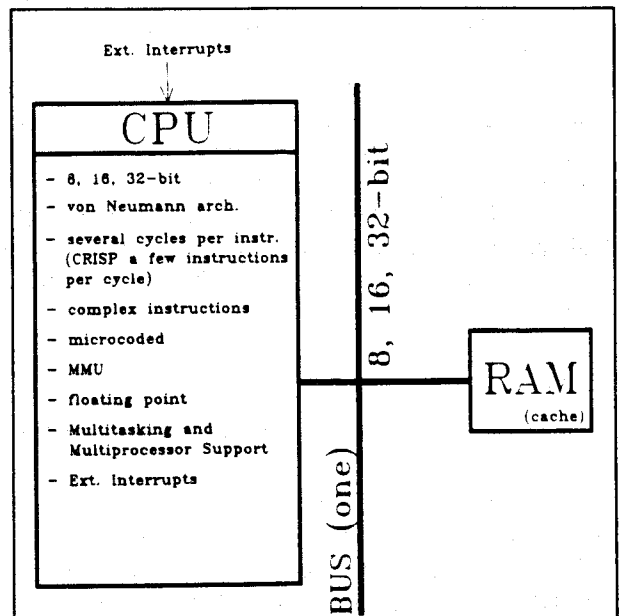


Figure 12. Simplified Block diagram of CISC

- Have control lines to support a multiprocessing environment.
- CISCs are connectable to different bus width devices.

RISC and CISC may become more alike in the future. RISC is a technology, a philosophy of design, not a product. Some design techniques that have been applied to RISC machine can be applied to CISC architecture to improve performance.

An example is the National Semiconductor 32532 general purpose processor which incorporates many RISC features. It has:

- on-chip data and instruction caches
- direct-mapped caches for stack access
- pipelining and branch-prediction logic
- it uses microcode (not used in RISC) only for the most complex instructions and hardwired logic elsewhere.

Processors like the 32532 with Intel 80486 and Motorola 68040, incorporate more RISC-like features to push the number of cycles for most of the instructions below 2. These new features will probably characterize the new type of processor as CRISP (Complexity-Reduced Instruction Set Processor) [14].

4.2.6. High performance EMBEDDED CONTROLLERS versus DSP.

This architecture (new since 1990) has been designed to meet the need of embedded applications such as machine control, robotics, process control, avionics, and instrumentation.

These types of applications require high integration, low power consumption, quick interrupt response time and high performance.

Since time to market is critical, embedded processors need to be easy to use in both hardware and software design.

The newest chips in this family are from Intel and Motorola with some differences in their characteristics.

Intel chips (80960) are based on a RISC core architecture. Each processor in the series will add its own special set of functions to the core to satisfy the need of a specific application or range of applications in the embedded market. For example, future processors may include DMA controllers, timers, or an A/D converter.

Other characteristics are: large register set, fast instruction execution, load/store architecture, simple instruction format, overlapped instruction execution, integer execution optimization.

The Motorola MC683xx family combines the high performance of M68000 family microprocessor with intelligent data-handling peripherals on a single chip.

In one chip (32-bit) besides the CPU, there are: DMA controller, a timer module, a serial I/O module, a system interface module, and a 16-bit data port. Instructions are similar to the M68000 Family and need several cycles per instruction.

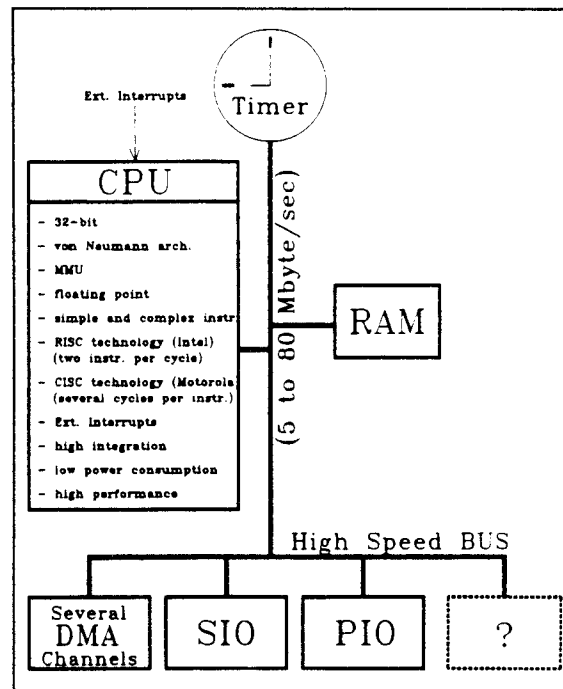


Figure 13. Simplified Block diagram of high performance Embedded Controllers.

4.3 Future trends in microprocessors.

The goal computer architects want to reach up to now seems to be the execution of almost all instructions in a single cycle. Now that this goal has been reached, it seems that the game is not over, but competition will continue in trying to have more than one instruction in a single cycle. An example is the Intel i860 which executes up to three instructions in one cycle and the IBM RISC (a seven chip set) which executes up to five instructions in one cycle. At this stage a more serious problem will then arise: how to write optimized compilers that will take advantage of this hardware capability.

4.4. Classification of DSP hardware.

The DSP chips commercially available, can be classified in four main categories:

4.4.1. High performance general purpose DSP.

These processors have an architecture similar to an MPU/MCU, but in addition may include on chip multiplier, RAM, ROM, DMA, peripherals I/O, hardware Do-loop, pipelining and several internal and external busses.

Some examples of these DSP types are:

- AT&T DSP16, DSP16A, DSP32, DSP32C
- Motorola DSP56116, DSP5600x DSP9600x
- Texas TMS320Cxx
- Analog Devices 2100.

4.4.2. Algorithm specific DSP.

The architecture is configured for the optimum processing of a specific algorithm.

Among the DSP types designed for executing digital filter algorithms (FIR, IIR) there are: INMOS A100, LSI64240, Motorola DSP56200, Zoran.

Among the DSP types designed for executing FFT there are: TRW2310, HDSP66110, UT69532, Zoran.

4.4.3. Application specific DSP.

This type of DSP's are designed to implement specific applications such as a modem or voice encoder/decoder.

4.4.4. Building blocks

Multiplier, adder, registers, RAM, ROM, I/O peripherals, etc. can be used as building block components to configure a complete DSP system with very high performance but with higher costs. (E.g. MaxVideo graphic processor)

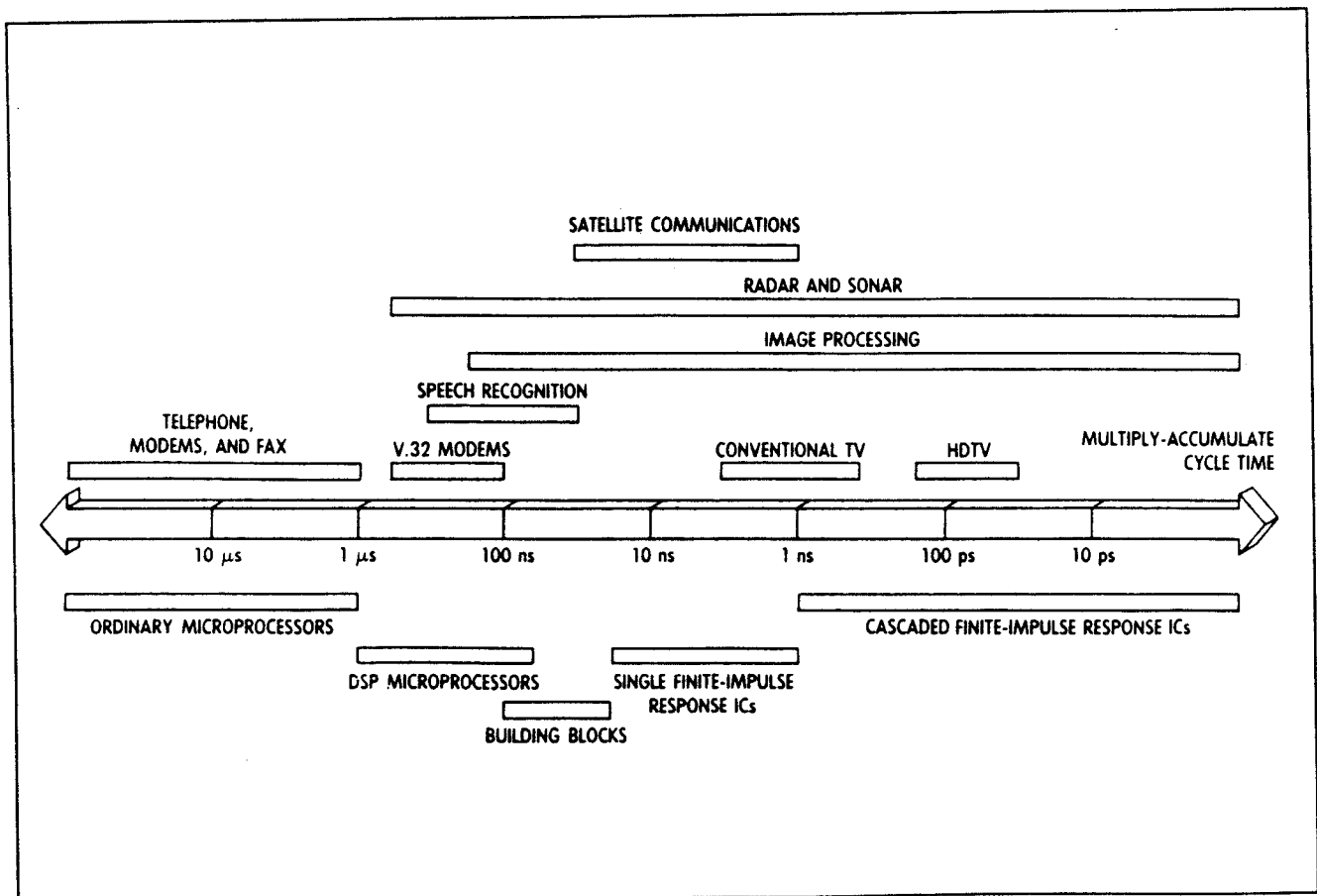


Figure 14. Spectrum of application of microprocessors (source: Electronics. April 1989)

4.5. Comparison between DSP products.

The firms leading in designing and manufacturing DSP's are:

Analog Devices, AT & T, Fujitsu Hitachi, Honeywell Inc., IBM, Inmos, OKI, NEC, Motorola, Philips, Signet, Thomson, Toshiba, Texas Instruments, TRW, Zoran.

Table 3.

Firm	Type	Tec.	Cyc (ns)	I-size	P-mem.	D-mem.	N bus	ALU	Other features
Analog Devices	ADSP2100	CMOS	125	24	32K	16K	1	16-Fix	
AT&T	DSP16A DSP32 DSP32C	CMOS CMOS CMOS	15 40 20	16 32 32		2K 4K		16-Fix 32-Flo 32-Flo	SIO, PIO, DMA
Fujitsu	MB8764	CMOS	100	24	1K	1K	1	26-Fix	
Hitachi	61810 DSPi	CMOS CMOS	250 50	16 16				16-Flo	Fast I/O
Inmos	IMS-A100	CMOS	100	16				16-Fix	Optimized for filters
Motorla	DSP56116 DSP56000 DSP56200 DSP96000	HCMOS HCMOS HCMOS HCMOS	50 75 97.5 37.5	16 24 8 32	2k 256 256 512	2k 256 256 512	6 1 8 8	16-Fix 24-Fix 16-Fix 32-Flo	40-bit accumulator, DMA, SIO DMA, SIO DMA, SIO
National	LM32900	CMOS	100	16				16-Fix	No internal memory
NEC	mPD7281 mPD77230 mPD77220	CMOS CMOS CMOS		32 24	2k	1k		32-Flo 24-Fix	Data Flow SIO subset of 77230
Oki	6992	CMOS	100	22				22-Flo	
Philips	5010	CMOS	125	16	128	128	2	16-Fix	SIO
Thomson	68931	CMOS	360	32				32-Fix	
Toshiba	6386/7	CMOS	250	16				16-Fix	
Texas Instrum.	TMS320C10 TMS320C25 TMS320C30	CMOS CMOS CMOS	160 100	16 16 32		144	1	16-Fix 32-Flo	
TRW	TMC2310	CMOS	50	16					1024-point complex FFT in 514 μ s
Zoran	34161 34322 35325	CMOS CMOS CMOS	100 100 100	16 32 32				16-Flo 32-Flo 32-Flo	IEEE-Floating point

The figures 15, 16, 17 and 18 give examples of performance of some Digital Signal Processors from leading manufacturers.

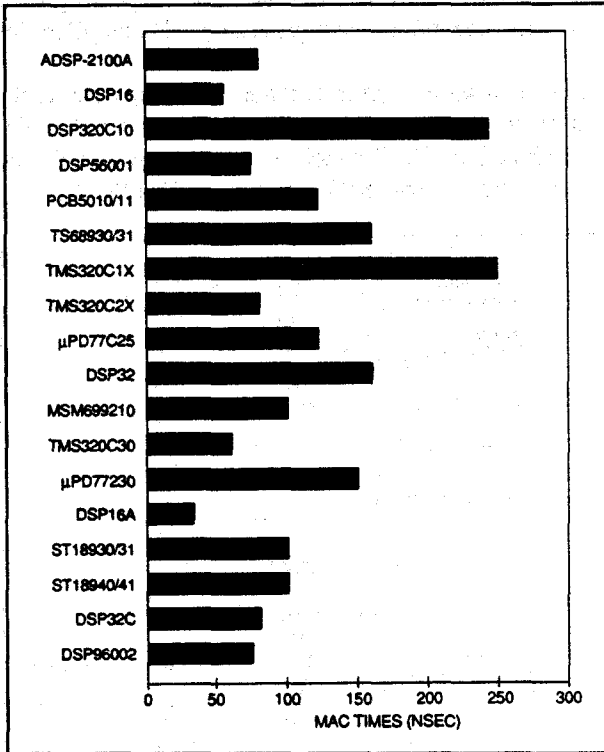


Figure 15. The multiply-accumulate (MAC) time comparison between different DSPs (EDN magazine, September 29, 1988).

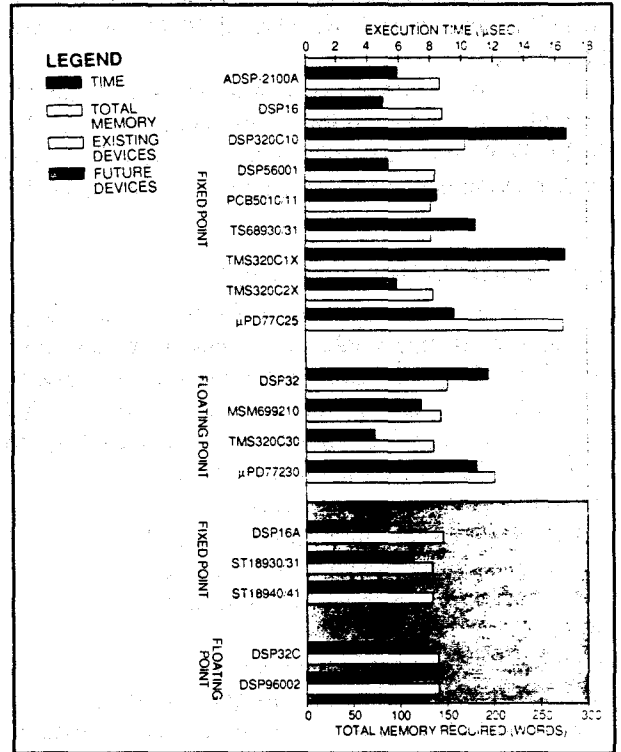


Figure 16. 64-tap filter algorithm time comparison between DSPs (EDN magazine, September 29, 1988).

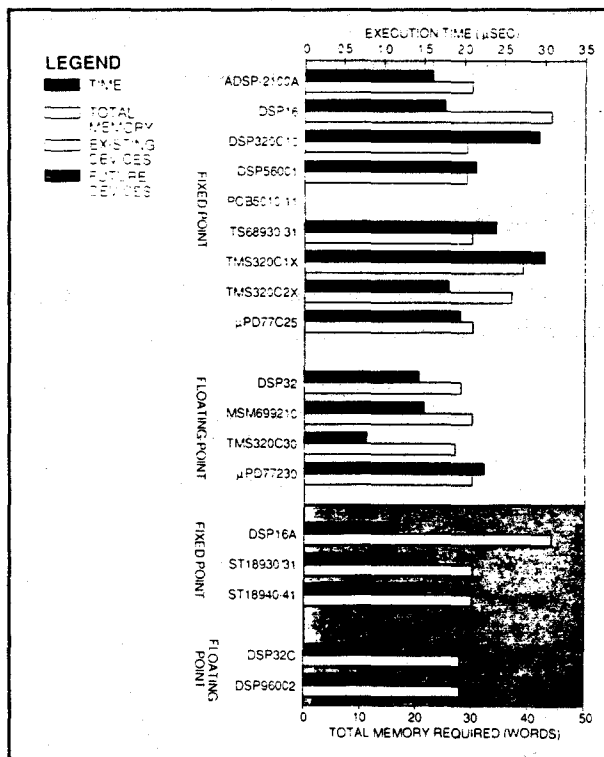


Figure 17. 3x3 times 3x1 matrix multiply time comparison between DSPs (EDN magazine, September 29, 1988).

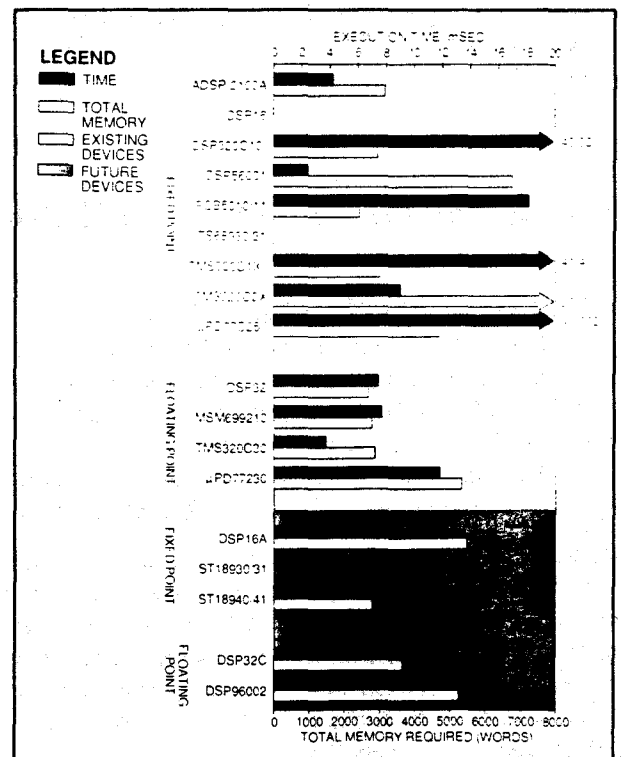


Figure 18. 1024-point FFT (radix-2) algorithm time comparison between DSPs (EDN magazine, September 29, 1988).

4.6. DSP software support.

Assembler language may be convenient to optimize a fast algorithm, but it is a limitation for large programs. Therefore the principal firms (AT & T, Motorola, Philips and Texas Instruments) are also providing "C" compilers for their DSP's.

Various other firms, who do not manufacture DSP chips themselves, offer software development tools. Examples are the Signal Processor Workstation (SPW) from Tektronix that runs on VAX or Apollo Computer Domain, Euclid Tools and DSP-1000 from Datacube. DSP Development introduced DADiSP which is a menu driven software for displaying and analyzing digital waveforms, and STEP Engineering offers Step-4 SDT running on IBM PC AT

4.7. Applications overview

Due to the rapid advances in the technology, new and interesting application areas are being found for DSPs. Many applications are moving from analog to digital processing in the quest for higher performance and lower cost. Low-cost and high-speed favours the use of DPS in these applications.

- instrumentation, spectrum analysis
- telecommunication (high speed modems)
- image processing and pattern recognition
- speech recognition, musical synthesizer
- direction finding in radar,
- target tracking (closed loop systems)
- ultrasound medical imaging, image processing
- automobiles: antiskid braking systems, adaptive suspension, engine control and instrumentation
- vibration analysis
- medical electronics
- disk drives, tape drives
- printers, plotters and consumer products
- digital filters
- digital HI-FI, digital AM/FM radio, digital video
- workstations
- robotics

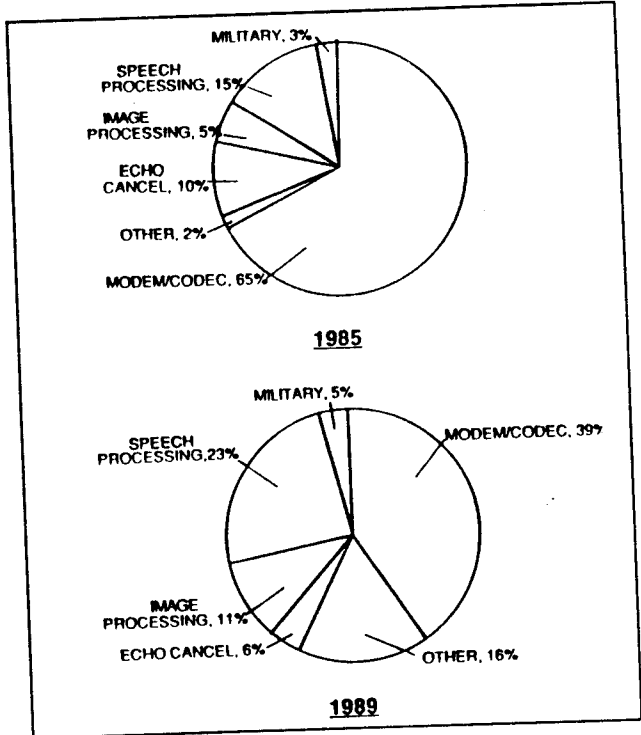


Figure 19. DSPs application areas.

5. THE USE OF DSP IN HIGH ENERGY PHYSICS.

5.1. DSP in Accelerator Control.

Digital Signal Processors are being used in the accelerator control in place of hardwired logic for calculating FFT in the Q-measurements, in closed-loop control, in filtering and signal analysis.

The use of FFT is required to measure Q-values. Q_H and Q_V are respectively the number of complete betatron oscillations per beam revolution in the horizontal and vertical plane (see figure 20). The normal method of measurement with the Q-meter is as follows: coherent betatron oscillations are excited in a bunched or debunched beam, radially, vertically or simultaneously in both planes by triggering the vertical and horizontal Q-kickers [15].

In the Accelerator control, due to slow reaction of the magnetic field, there are no very high speed processing requirements in a closed-loop made of pickup sensors and current regulators of the magnets. The high processing speed that will require use of a

DSP is mainly needed for closed-loops within a measuring system.

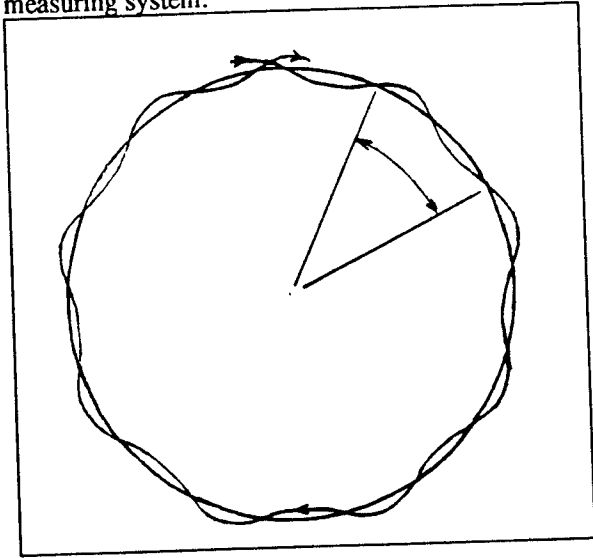


Figure 20. Betatron oscillation of a beam closed orbit.

5.1.1. LEP Q-measurement

The fractional part of the betatron oscillation is measured in LEP by exciting transverse oscillations and measuring the beam motion over many revolutions (LEP/BI group [16]). Data treatment of beam excitation and the analysis of the beam motion is a digital process that can be executed on a DSP. The choice of the excitation mode and the data treatment depends on the machine state and the desired measurement speed and accuracy. Three measurement modes have been implemented into the LEP Q-meter:

- Swept frequency (SF)
- Resonant excitation with phase-locked loop (PLL)
- Random noise excitation and Fast Fourier Transform (FFT)

where only the PLL mode makes full use of the real-time capability of the DSP.

The Q-meter is used in the PLL mode as a constant monitor of the betatron resonance, or with the help of a feed-back system, acting on the currents of quadrupoles as a system to stabilize the resonances at preselected values.

The Phase-Locked Loop (PLL) algorithm must be executed in less than 89 μ s (LEP revolution time), this comprises:

- position calculation (four sensor readings of the pickups, see figure 21)
- beam excitation with sinusoidal kicks
- control 2 digital regulators with proportional and integral action.
- four first order ripple filters.

The other two measurements do not have feed-back, and do not require the digital process to be executed in short time.

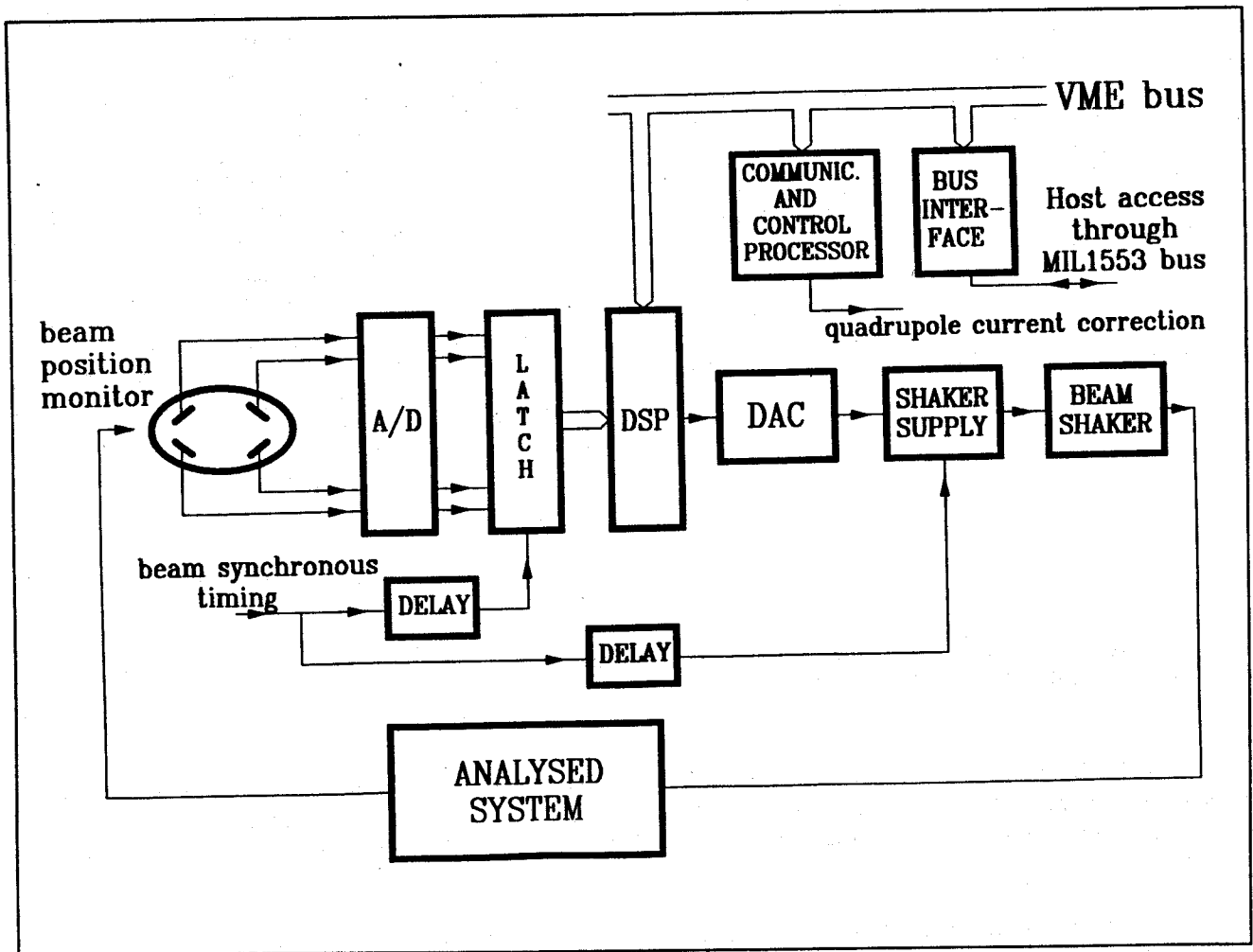


Figure 21. Hardware configuration of the signal processing and shaker driving electronics [16].

5.1.2. SPS Q-measurement

In the monitor group of SPS, a data acquisition system BOSC (Beam OSCillation) [17] has been built. The intensity and the position of up to 8 particle bunches can be measured each revolution. A measurement of up to 1 million subsequent revolutions is possible. The first use of the system is for machine studies, in which it is necessary to measure betatron resonance to high precision, reconstruct phase space plots by using the information of 2 pick-ups (90 degrees apart in phase), find intensity losses and record the lifetime of a coasting beam. BOSC has been constructed to replace a number of single purpose instruments for operation as well as to serve as a tool for specific machine experiments. The system can be fed by signals from any source: position and intensity from directional coupler pick-ups (single bunch) or electrostatic pick-ups (many bunches) with their 20 Mhz and 200 Mhz receivers respectively, intensity signals from the beam current transformer (BCT) for unbunched beams and various signals monitoring the hardware.

At present a Fast Digital Parallel Processing module (FDPP, see figure 22) [18] is used in the BOSC system to provide Real-time Signal Analysis. The use of the FDPP is not limited to the Q-measurement application, but it can be used whenever it is necessary to make intensive calculations on the local acquired data from A/D converters (FFT, Power Spectrum, etc.). Commands can be issued from an Apollo workstation in the SPS control room or from any workstation in the Ethernet or Token ring [19-20].

The FDPP is installed as a daughter board on a VME industrial motherboard IMSB014 from INMOS. Data are fetched through the Transputer serial link at 300 Kbyte/sec and the FDPP operates as follows: while the Transputer is uploading results of FFT (n-1) from dual-port memory X and downloading new data for FFT (n+1) into the same memory, the DSP32C is converting integer inputs to floating, executing FFT (n), calculating the power spectrum and converting DSP32C floating to IEEE floating in dual-port memory Y.

The above operations for 256, 512, 1024, 2048 or 4096 point FFT can be started from an Apollo workstation (or from a Processor Control Assembly, PCA) in the SPS control room.

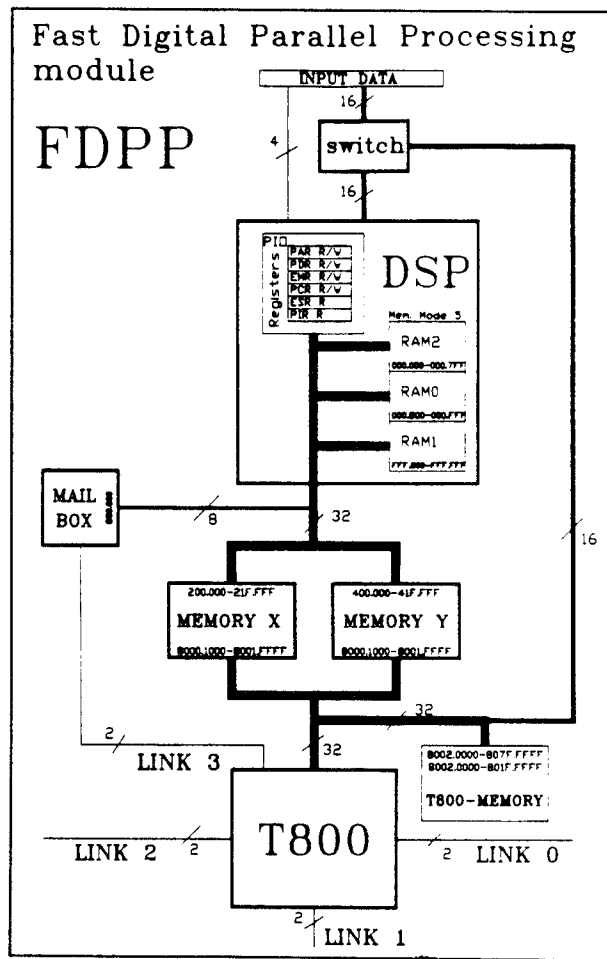


Figure 22. Fast Digital Parallel Processing module block diagram.

Figure 23 illustrates in the lower sections the digitalized input signals from the SPS beam and in the upper ones, the results of the FFT done by the FDPP module.

The FDPP module can be replaced by a FDPP_DM [21], having the same architecture as the FDPP with an additional internal direct-to-memory channel permitting data transfer up to 150 Mbyte/s with either memory bank "X" or "Y". This feature will improve the transfer from the A/D acquisition module and provide a better match with the processing time.

The FFT algorithm used for the SPS Q-measurement first converts integer input data from the A/D converter into floating point format, implements an in-place, decimation-in-time, N/2 point, radix-2, complex FFT. The complex FFT is then post processed to obtain the N-point real data FFT. At the completion of the routine, the N input locations store the complex output data up to Nyquist frequency (see E.O. Brigham, The Fast Fourier Transform, Prentice-Hall, 1974, p. 169).

Table 4. FFT execution time in floating point for real input data of the FDPP module installed in the BOSC system.

256-point FFT	512-point FFT	1024-point FFT	2048-point FFT	4096-point FFT
0.454 ms	0.976 ms	2.085 ms	4.447 ms	9.440 ms

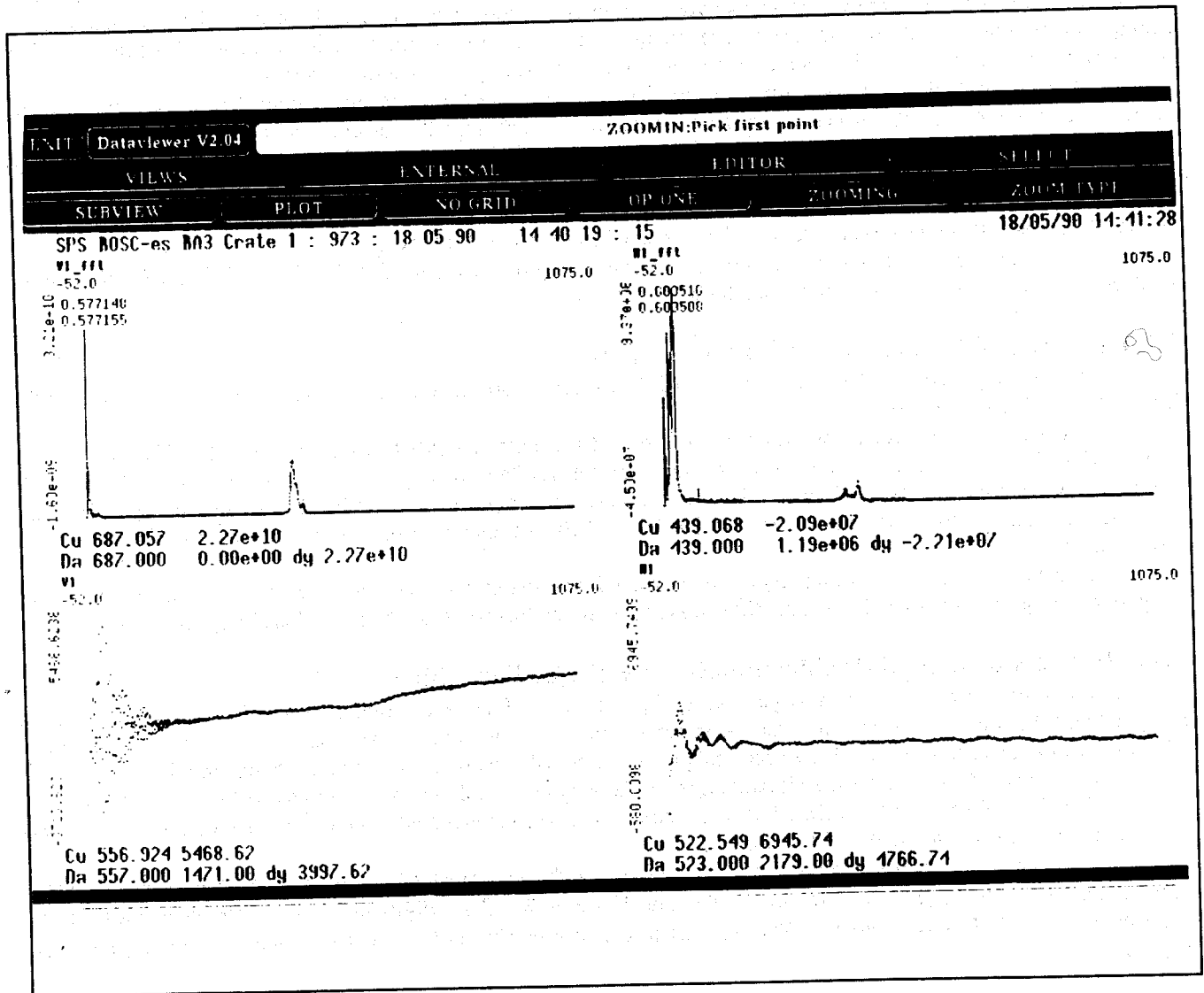


Figure 23. 1024-points FFT executed on FDPP for SPS Q-measurements.

5.1.3. PS Q-measurement

For the Q-measurement of a rapidly cycling accelerator like the PS one has to use sufficiently fast hardware to obtain a fine time resolution of this measurement (PS/PA group [22]). An FFT processor (Computer General VASP-16) containing a TMS 320C25 DSP and 4 Zoran ZR 34161 vector processors has been chosen, which allows a complete acquisition and analysis every 5 msec.

Different modes of operation can be selected:

- calculation of the Q-value by interpolation between the two most important lines of the spectrum. The length of the time window is 512 points resulting in a resolution of the Q-fraction of .001 at best.
- the whole signal spectrum can be output every 5 msec and could be shown in a mountain range display with FFT size of 256, 512 or 1024 points.
- in the so called sliding FFT a time segment of 8 msec can be stored in a fast memory and then analyzed with a fine time step of 1 sample to show fast evolutions of resonance phenomena.

This dedicated FFT card housed in VME crate is working under OS-9. It can be programmed using C or assembler and it will be accessed by the PS-control system.

5.1.4. Schottky measurements

To check the operation of cooling and stacking processes of the antiproton collector and antiproton accumulator schottky measurements [23] are used, as they don't affect the operation of the machines (PS/AR group [24]). Using this technique, it is possible to measure intensity as well as longitudinal and transverse beam profiles.

The distribution of the particles in the three planes produces a spectrum of revolution frequencies. From the shape of these spectra one can deduce the profile of the circulating beam. By integrating the spectra one obtains the beam intensity.

To measure these spectra the schottky noise generated by the beam is sampled. A fast Fourier transform is then made on the data to obtain the noise spectrum. To obtain a useful measurement from the noise it is necessary to average many spectra. At present these measurement are made using commercial laboratory instruments. The drawback to the present system is that the instruments have to be set-up for each measurement.

A new pulse of antiprotons arrives every 2.4 seconds from the antiproton target. To allow to make ten measurements in the cycle it is necessary to sample, transform and average the result in 240 milliseconds.

By using a 12 bit ADC, a signal to quantization noise ratio better than 60 DB is obtained. Taking data at 200 Kilosamples/second, and allowing 40 milliseconds for reading and overheads, 40 Kilosamples of data in the time slice are recorded. A resolution in frequency of 1/1000 is required, thus it is necessary to average 20 FFT's each of 2048 points. This means that a new spectrum must be computed every 10 milliseconds.

To achieve speeds of this order a dedicated specialized processor is needed. The Tekmis TSVME 350 with the DSP from Thomson offers this performance for a modest price.

5.1.5. Filters and Control for LEP Power supply

A feasibility study has been made to use a DSP to solve a problem of oscillation in the power supplies of the SPS (SPS/PCO/PS group [25]).

Because of the capacity between the magnets and the earth, the Main Ring dipoles in the SPS constitute a transmission line. The measured open loop transfer function for the power converters combined with these dipole magnets shows almost constant gain (= 0 DB) between approximately 30 Hz to 60 Hz. During normal operation, any noise which is picked up in the frequency range 30 Hz to 60 Hz will therefore be transmitted as current oscillations to the magnets.

In order to reduce the oscillations in the magnets it was decided to make a digital filter which should reduce the gain for frequencies above 30 Hz. This filter was placed in the forward branch of the closed regulation loop.

5.1.6. Data acquisition of Bhabha monitors for the LEP machine.

For optimizing luminosities and particle backgrounds, the four Interaction Points (I.P.) of LEP have been equipped with 16 compact Silicon-Tungsten calorimeters [26]. Two conjugate calorimeters, placed symmetrically to an I.P. in view of intercepting scattered electrons and positron from Bhabha reactions, form a Bhabha monitor.

The energy deposited by the electromagnetic showers in the Silicon detectors has to be recorded for event analysis. This is performed in the CES 8150 VME modules. Eight independent systems are installed and operational from the LEP control room via the LEP token ring.

For each monitor, the analog charges deposited in the calorimeters are integrated, held and sent via a multiplexer to a 128 channel module containing four 12-bit ADC (3 μ s digitizing time) working in parallel and controlled by the DSP VME module. This module also performs the reading of the memory containing the event tagging.

5.1.7. DESY Accelerator Control.

Several applications have been implemented at the DESY Accelerators, by using industrial or DESY designed IBM PC-boards or VME-boards [27]. They are used for measurement and control in the following fields:

- menu driven FFT Analysis for measuring beam instabilities.
- digital control loops for p- accelerators for the:
 - a) controller for the dipole currents
 - b) controller for dipole current power supplies
 - c) controller for phase loops (phase difference between bunches and RF)
 - d) controller for cavity tuning loop.

For the design of all the digital control loops, commercial packages, such as FDAS and DISPRO have been used.

5.2. DSP in triggers and data acquisition.

5.2.1. FDDP Fast Digital Data Processor

High Energy Physics experiments now use huge detectors to track elementary particles and it is extremely important to be able to make decisions, based on the information of thousands of signals in real time as quickly as possible.

In 1985, the Torino group: D. Crosetto, E. Menichetti, G. Rinaudo and A. Werbrouck decided to use a DSP as the basic element in an intelligent programmable trigger decision parallel processing system in the FEMC Delphi detector to satisfy the requirements of the second level trigger decision within 44 μsec and to have more refined information on clusters in the detector within 200 μsec for the third level trigger. A prototype of the parallel system made of six DSP working in parallel has been designed and build in VMEbus boards in 1986 [28].

The first test on beam H6 at CERN was made in the summer 1987. The performance was as expected and then the same system has been constructed in Fastbus [29-30] and used for high level trigger decision in the Forward ElectroMagnetic Calorimeter (FEMC), in the Small Angle Tagger (SAT) and for the data acquisition and preprocessing in the Very Small Angle Tagger (VSAT).

5.2.1.1. Application of FDDP to FEMC high level trigger.

The organization of the FEMC trigger is described in detail in reference [31]. The main points are: the two end caps, each covering the angular region between 10° and 35° , consist of about 4500 lead glass blocks each, read through phototriodes and organized in an (θ, ϕ) geometry. The amplified and shaped signals are analogically added to form a "superblock" signal in the (θ, ϕ) geometry adopted in the DELPHI trigger. The segmentation of the complete end cap is then made of 24 azimuthal (ϕ) sectors, each containing 8 superblocks (see figure 24). A DSP is associated to each sector (see figure 25). The DSP analysis of the data consists of two procedures, the first as part of the second level trigger, the second as part of the third level. For the second level trigger, within 24 μsec each DSP in parallel acquires data from the A/D of the 8 superblocks, does pedestal subtraction, correction for calibration constants, comparison with a threshold value and calculates the sector Energy. A dynamical pedestal subtraction is also possible and it has in fact been tested, to subtract coherent noise in the detector.

It is performed by reading again the analog signals 5 μs after the beam crossing and by using this reading as pedestal value. The Energy sum processor [32] will then calculate and encode the total end cap Energy.

For the third level trigger, which starts when the event has passed the second level trigger, a fast cluster search is performed and cluster energy and center of gravity coordinates are calculated. The cluster search is based on a fast, one pass algorithm [33], which takes also into consideration data of the two adjacent sectors and associates to the same cluster all superblocks which have a finite common edge and have a signal above threshold. In the first year of LEP running, only the second level trigger part of the DSP analysis has been implemented.

5.2.1.2. Application of FDDP to the second level trigger of the SAT.

The Small Angle Tagger of DELPHI consist of two endcaps, covering the polar angle region between 3° and 8° with the beam direction in the forward and backward hemisphere. Each endcap consist of a Si strip tracker and a scintillating fiber calorimeter. Only the calorimeter contributes to the trigger. The signals are first added analogically to give 24 azimuthal sectors per endcap and three DSPs then acquire 8 channels (converted form A/D) each. Within 22 μsec , each DSP in parallel fetches sequentially the 8 converted data, subtracts pedestals, corrects for calibration constants and adds the corrected values to give the partial energy sum. Partial energy is then transferred to one DSP that calculates the endcap energy, compares it with two threshold values (low and high threshold) and encodes the result in the format expected for the DELPHI trigger data within 4 μs . Up to now the system has not yet been implemented.

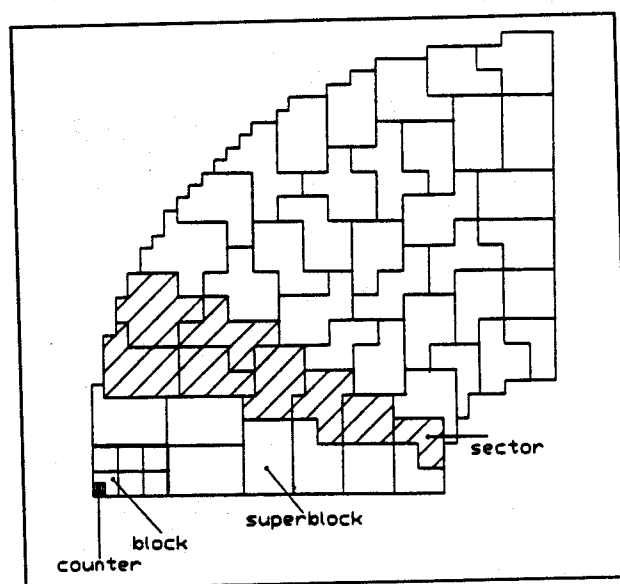


Figure 24. The FEMC endcap segmentation for the 2nd and 3rd level trigger.

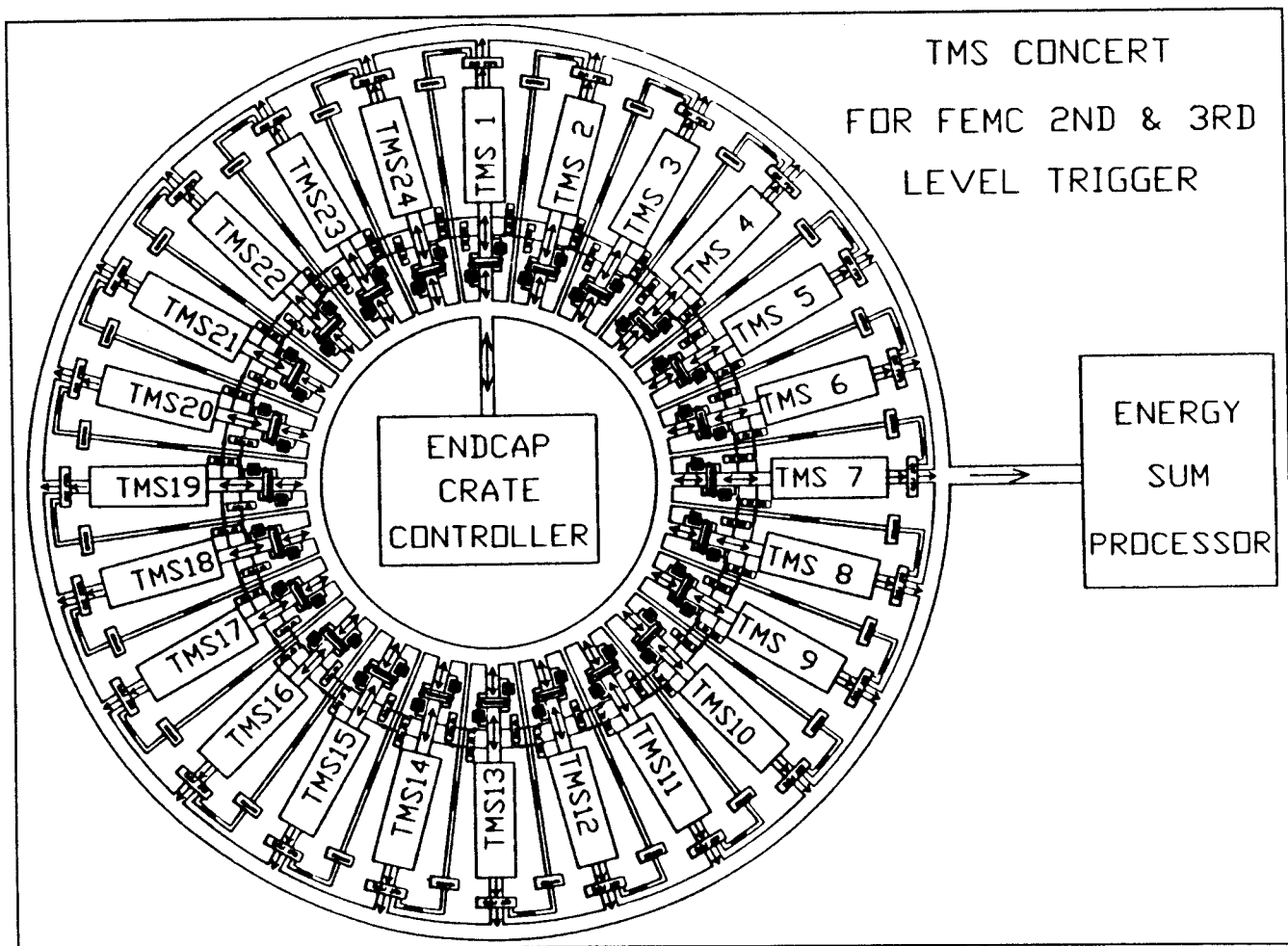


Figure 25. The FEMC 2nd and 3rd level trigger DSP architecture with 24xTMS32010.

5.2.1.3. Application of FDDP for the readout of VSAT

The VSAT is a luminosity monitor for the detection of Bhabha scattering at very small angles (from 5 to 6.5 mrad with respect to the beam direction). It consists of four Si/W sampling quantimeters, two in the front and two in the back arm, positioned beyond the superconducting quadrupole. Each quantimeter covers an azimuthal angle of about 15° around the horizontal direction, for a total coverage of about 1/6 of the azimuth. Longitudinally, the quantimeter consists of 12 Si full area detectors ($3 \times 5 \text{ cm}^2$ FAD), interleaved with W slabs, to measure the electron energy, and three planes of Si strips, with 1 mm pitch (SD), two having vertical strips (x-plane, 32 strips), the third horizontal (y-plane, 48 strips), to measure the electron impact point.

Each quantimeter has data processed by three DSP's. One DSP equipped with a 12-bit A/D converter acquires data from the FAD detector, a second and third DSP equipped with 8-bit A/D converters, acquire data from the y-plane and x-plane SD detector. Besides controlling the timing of the analog module during acquisition and data fetch, the DSPs perform pedestal subtraction, correction for calibration constant and data formatting. The time needed to process the whole y-plane is about $130 \mu\text{s}$, a similar time is required for the x-plane. Time needed to process the 12 FAD data is about $110 \mu\text{s}$. Raw data and total energy are then output to the 4-event-buffer.

5.2.2. Energy Sum Processor (ESP) for the FEMC (DELPHI).

In the DELPHI experiment, the first application of the ESP module is a "total energy" trigger for the end-cap electromagnetic calorimeter [32]. At each new valid beam-crossing, the trigger program prepares some constants and then waits for the first level trigger. Digitized trigger data, corresponding to the energies deposited in 24 sectors of the detector, are then read-out, added and compared with pre-defined thresholds. Results are made available to the Local Trigger Supervisor in the Register of Results, to the other trigger processors on the special ECL bus, and to the read-out system in the Front End Buffer. The trigger algorithms, cross-assembled and tested by the simulator program in an external computer, are recorded on EPROMs and bootstrapped into the DSP internal memory from the "Boot EPROM" at power-on or reset.

The DSP56002 has been chosen as a basis for the design of the ESP Fastbus module. The interfacing to Fastbus and to the detector front-end electronics was made more compact and reliable by an extensive use of XILINX

XC3020 Programmable Gate Arrays. All the circuits could fit in less than one half Fastbus board, at a cost roughly an order of magnitude lower than that of an equivalent processor of the previous generation.

5.2.3. Use of DSPs in the readout of DELPHI microvertex

The microvertex detector of the DELPHI experiment consists of two cylindrical layers of Si-strip detectors. The strips are parallel to the beam axis and have a pitch of 25 mm and a readout pitch of 50 μm , providing a resolution of 5 mm in the θ, Φ plane [34].

The readout of this detector (55000 channels in total) is highly simplified by the use of a front-end VLSI, the Microplex chip, which is directly bonded to the detector strip and handles the charge amplification, double correlated sampling and readout of 128 channels on an analog differential bus. The data acquisition is made in Fastbus by SIROCCO IV [35], which performs at a frequency of 5 MHz and with a 10-bit resolution the digitization of a sequence of up to 2048 analog signals (from a group of up to 16 Microplex chips in series). After digitalization, the individual strip pedestal is subtracted, and a correction for the common mode noise is also made for each group of 128 channels. After these corrections, the real strip signals are finally obtained and could then be compared to a programmable threshold value. In fact, since the total charge produced by a particle crossing the detector spreads over several strips, more complex algorithms have to be implemented to identify hit strips, which also take into account residual noise levels on individual strips after correction. Addresses of hit strips and of their 3-4 neighbors on each side are generated, together with their respective signal amplitudes. Since the track multiplicity in e^+, e^- collisions at DELPHI is low, only a small fraction of the detector receives a useful signal, and a data reduction factor of 20 is typically achieved.

The digital signal processor DSP56001 from Motorola has been chosen to perform the task of pedestal and common mode measurements, signal correction, zero-suppression and final data formatting.

5.2.4. Calorimeter data acquisition system of HERA.

H1 is one of the two experiments which is installed on HERA electron-positron collider at DESY(Hamburg). In the H1 liquid argon calorimeter data acquisition system, there are 45,000 calorimeter channels to be read out, multiplexed 128 to 1. The output of the multiplexer is amplified before being transmitted to the conversion board, 50 meters away. 20 % of the channels are submitted to a double amplification in order to increase the signal dynamic range in an area of the calorimeter where large signals are expected.

Each conversion board deals with 1024 channels, or 512 in case of double amplification. The external board is made of 8 ADC channels running in parallel. Each channel has a sample and hold (1 μsec sample time), a 12 bit ADC at 5.2 μsec conversion rate and a 12-bit output latch. On the analog inputs of the ADC board, there is a 4 to 1 multiplexer to perform test sequences of the ADC board.

The sequencer located on the VME board receives the H1 level-2 trigger approximately 10 μsec after the collision and starts the data conversion. Due to multiplexing, the sequencer will repeat 128 times the conversion sequence which includes: the ADC clock at 5 MHz; a start conversion pulse and a hold signal.

When the sequencer reaches the end of the first conversion sequence, it sends an interrupt to the DSP, announcing that data are ready to be processed in the input FIFO. The DSP then reads the raw data from the FIFO, gets the corresponding gain and offset parameters from its external memory, then normalizes and formats the data, compares to a threshold, inserts an identifier code in the data itself and eventually stores the result in the dual port RAM. Acquisition and signal processing tasks are pipelined, so that the DSP process does not increase significantly the dead time of the detector as it does exceed very much the conversion time of the ADC's.

5.2.5. Data reduction in the OBELIX experiment.

The OBELIX apparatus, installed at LEAR at CERN, is devoted to the high statistics study of antinucleon-nucleon and antinucleon-nucleus interactions [36]. Its tracking device, a drift chamber with 3444, 1.5 m long, sense wires organized in 82 angular sectors, must measure particle momenta and dE/dx . The left and right end of each sense wire is equipped with a 100 MHz FADC channel featuring 8-bit resolution and non-linear response function to increase the dynamical range to 10-bits. The 176 FADC channels are housed in a special purpose front end crate equipped with a programmable zero-suppressor and with 2 AT&T-16A DSPs operating at 70 MHz.

The data filtered by the zero-suppressor (about 60 bytes per hit from 240 original bytes) are fed on flight into the DSPs which linearize and equalize the samples, determine the drift time by using the Differentiation Of Signal (DOS) method, evaluate the pedestals of signals, determine the Z coordinate with the charge division method and estimate the total collected charge.

The reduced data are then read, via the differential VSB-bus, by 40 Front End Processors (MC68030), one per front end crate and sent, via VME-bus to 4 Sub Event Builder processor (MC 68030). A local Event Builder (MC 68030) collects the SEBs data, via differential VSBbus. The throughput of the whole system is about 40, 4-prong events per second, corresponding to 6400 hits per second.

5.2.6. Trigger and data acquisition for LHC.

The recent rapid improvements in performance of commercially available Digital Signal Processors and Data-driven Array Processors, make attractive their efficient integration into the front-end trigger [37-43] and data acquisition electronics of future detectors for High Energy Physics, in place of developing new VLSI processors for this special purpose.

A Parallel-Processing System based on Data-driven Array Processors, Digital Signal Processors and Transputers for trigger decision, data acquisition and compaction is described [21]. The system is modular and suitable for any calorimeter sizes and types such as the ones proposed in the R&D for the LHC experiments. The aim is to give full programmability for the trigger decisions and for the data compaction, over the entire calorimeter at the single-channel granularity, with no boundary limitation. The fast cluster finding system is based on two cooperating levels, the higher one utilizing FDPP_DM, and the lower one, the new data-controlled array processor for video signal processing, DataWave [44]. The proposed architecture [21] provides one DataWave processing element for each calorimeter channel and one FDPP_DM module for every 256 DataWave array processing element. This numbers have been chosen for fastest performance but may be changed. Fast inter-processor communication within the DataWave array overcomes the problem of overlapping areas, and effectively provides a continuous array of processing elements. Local maxima, along with their energy, are found in a very short time from the data-driven array processor, and are easily selected and transferred, with the data of their region of interest, to a higher-level DSP array-processor system even if part of the data fall outside the region originally attributed to a particular FDPP.

5.2.6.1. Fast cluster finding description for calorimeters.

Typical requirements for digital triggers and data acquisition/compaction in calorimetry of High-Energy Physics (HEP) experiments can be summarized as follows:

- a) to load data from the calorimeter and free the front-end electronics for a subsequent acquisition;
- b) to find the geographical address of clusters (or local maxima corresponding to the center of an energy cluster derived from the full granularity), calculate their energies and apply different thresholds;
- c) to calculate the missing energy and the transverse energy;
- d) to isolate the clusters found, and their region of interest, from the empty calorimeter channels, and calculate the cluster shape-factors (e.g. to separate pions from electrons).

The fast cluster finding system is based on two cooperating levels, the higher one utilizing FDPP Fast Digital Parallel Processing module [18] (or FDPP_DM), and the lower one, the new data-controlled array processor for video signal processing, DataWave.

Figure 26 shows an example of a general layout of a trigger and data acquisition scheme with its related data-flow phases and timing for a typical calorimeter of future HEP experiments.

The Fast Digital Parallel-Processing (FDPP) system is a modular system in which each node consist of one DSP tightly coupled to one Transputer. Any number of nodes can be interconnected using the four serial links (1.2 Mbyte/s) provided by the Transputer, while any data source can communicate with the DSP at 10 Mbyte/s.

The Fast Digital Parallel Processing Dual triple-port Memory (FDPP_DM) module has the same architecture as the FDPP module with an additional internal direct-to-memory channel permitting data transfer up to 150 Mbyte/s with either memory bank "X" or "Y".

The DataWave array processing system is made of DataWave chips, each one containing 16 identical, but individually programmable, processors working on the data flow principle and organized in a matrix form. Each cell is a RISC processor with 12-bit architecture which operates on the pipeline principle. The cells communicate with their four nearest neighbors via asynchronous parallel buses. The computing power of one processor is up to 125 MIPS, where an instruction may imply two operations, and the maximum data transfer rate via the processor boundaries is 750 MByte/s. The maximum program length of each processor is 64 words of 48-bit each.

A crossbar switch is provided to switch the parallel I/O ports of the DataWave chips to one of the following:

- the calorimeter input channels,
- the adjacent DataWave chips and
- the FDPP (or FDPP_DM) parallel array.

5.2.6.2. System modularity and flexibility.

The described system is modular in the sense that a "Logical Unit" is made of one FDPP (or FDPP_DM) module and 16 DataWave chips organized in a matrix 4 x 4, thus giving a platform of 256 DataWave PEs organized in a matrix of 16 x 16. This choice does not introduce any boundary limitations in the DataWave array but is simply a practical choice in associating the PE with the closest exit point to a higher-level of processors.

Any number of "Logical Units" can be interconnected in a matrix scheme, regardless of the hardware used (VME, Futurebus, Fastbus, SCI, etc.). All timing given should be independent of the calorimeter size (hundred to hundred thousand channels). As the size of the calorimeter increases, the only time increase is the one required to transfer the data of the clusters found from the transputer to the crate controller.

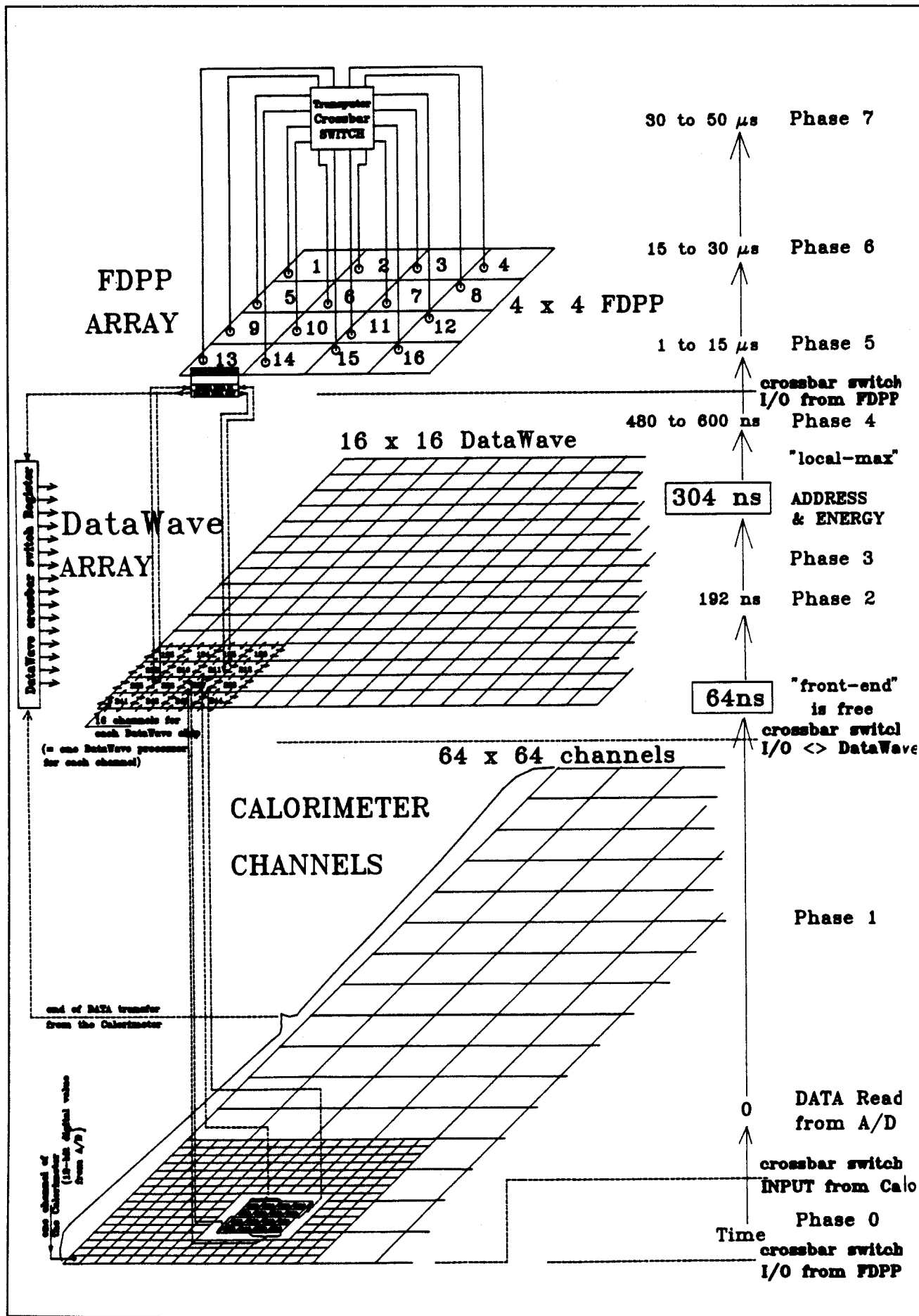


Figure 26. Fast cluster finding system block diagram.

5.2.6.3. Low end to high end programmability.

There are three levels of programmability in the fast cluster system, each one distinguished by: the time available, the capacity of the registers and program memory in the different processors.

Level 1 Each DataWave PE contains at the beginning the program along with calibration constants and transverse energy weights, all contained within a maximum length of 64 words of 48 bit each, loaded from the FDPP array.

Level 2. The programmability of the DSP part of the FDPP is then given by the program written in C or assembly language, having a maximum length of 256 Kbyte. The DSP designed as a Harvard architecture is a 32-bit machine with floating point performance at 25 Mflops. As such it needs the support of at least one host, in this case the Transputer.

Level 3. Transputer are based on Von Neumann architecture and are designed as programmable components to implement a system with a higher degree of concurrency than is currently common, that is:

a) programmable in several languages (OCCAM, Parallel C, FORTRAN, PASCAL, ecc.)

b) interconnected through serial I/O at 1.2 Mbyte/s

In addition, each of them is connected in each FDPP node to a DSP in both tightly coupled and loosely coupled modes.

5.2.6.4. Data reduction during acquisition

To have the possibility to analyze more in detail each local maximum, that is candidate to be a cluster, a relative large area (region of interest) surrounding the local maximum must be saved as useful data for further analysis. On the other hand, there is no need to transfer zeros from empty channels. The DataWave array is therefore capable of achieving a substantial reduction of the amount of data.

5.2.6.5. Performance of the system on real-time algorithms

Each digital value of the calorimeter channel is transferred into a DataWave Processor. In order to calculate the "local maxima", each processor needs to have the data of its neighbors in its own memory. A "local maximum" is defined as a channel value greater than, or equal to, that of its neighbors. This task of routing the data between processors is accomplished in a programmable way by the DataWave processors. The required time for this task is fixed and requires 128 ns. During the execution of the above program, in a pipeline mode partial calculation of the energy is also accomplished, thus only an additional 112 ns are required to have the energy of each "local maxima".

A very useful intermediate result is the ADDRESS of the "local maxima" and the total energy (or/and transverse energy) that are available after 304 ns from A/D data ready time.

For more refined calculations the FDPP array processing system is used. The "local maxima" found in an area of 256 DataWave processing elements are transferred to a FDPP together with the surrounding region. Empty calorimeter data are not transferred. Calculation aiming to separate pions from electrons are accomplished in the FDPP in the time of 15 μ s per cluster found.

A series of algorithms have been selected and coded in the DSP32C assembly code and executed directly on the FDPP module using SPACAL [45] data from the 155-channel prototype as they would be executed during the on-line data taking. Electron, pion and "jet" data were used. Two example results are given in the following section [46].

a) Tests results applied on algorithm 1.

The algorithms 1 (Figure 27), executed in 15 μ s, consists of the following:

- for all local maxima found by the DataWave array, compute, in floating point, the total energy (sum of 19 elements)

- for all local maxima compute, in floating point, I/C and O/C (see Fig. 27).

Figure 28a shows the distribution of I/C for 600 electron events at 40 GeV, while Figure 28b shows the distribution of I/C for jet events at 40 GeV, both obtained from algorithm 1 executed on the FDPP.

Observing the results of Fig. 28a and 28b for the calculation of I/C, one can see that there is a very good separation between electron and pion at the I/C = 0.016.

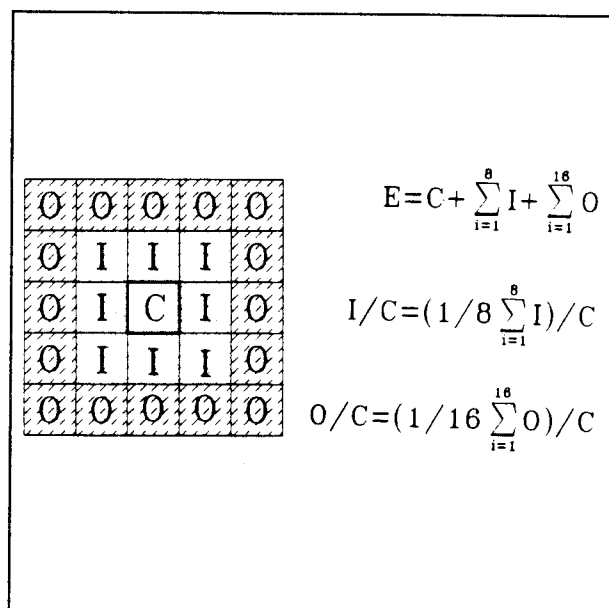


Figure 27. algorithm 1

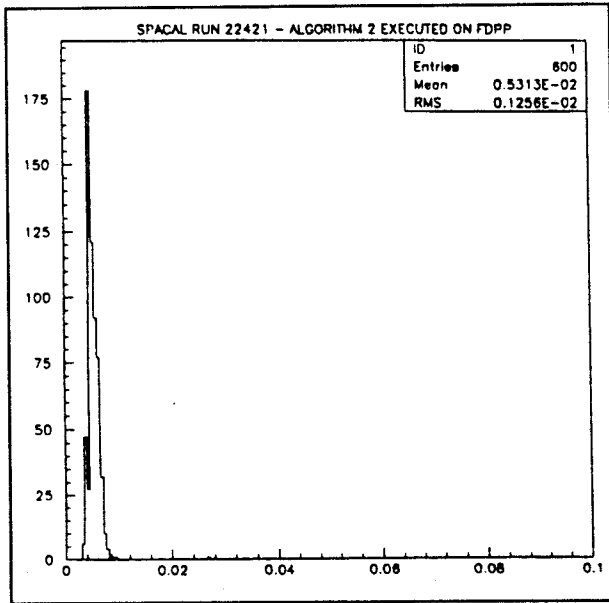


Figure 28a. Distribution of I/C for 600 electron events.

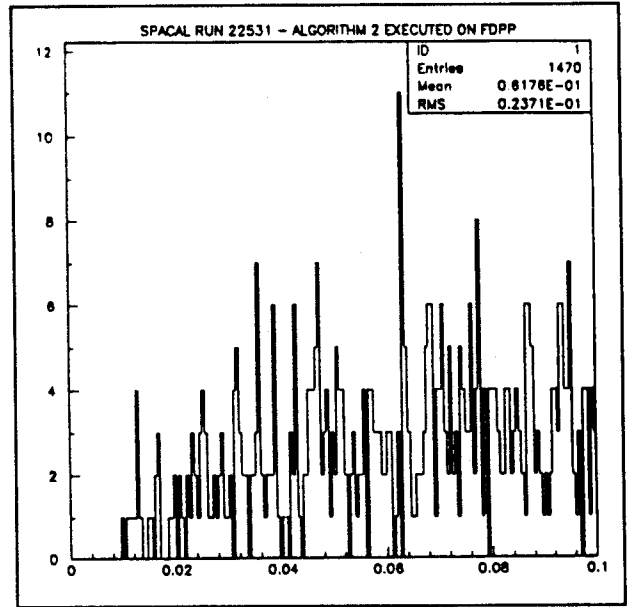


Figure 28b. Distribution of I/C for 1470 jet events.

b) test results applied on algorithm 2.

For all local maxima compute R_p [46] in floating point as reported in figure 29 ($15 \mu s$).

Figure 30a shows the distribution of R_p for electron events, while figure 30b shows the distribution for jet events, both calculated on the FDPP.

Observing the results of figures 30a and 30b for the calculation of R_p , one can see that there is a very good separation between electron and pion.

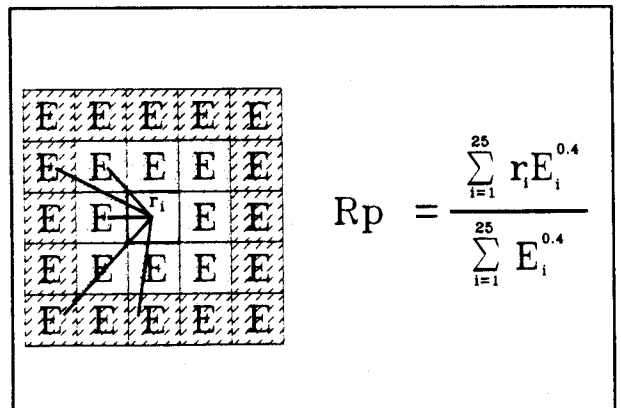


Figure 29. Algorithm 2.

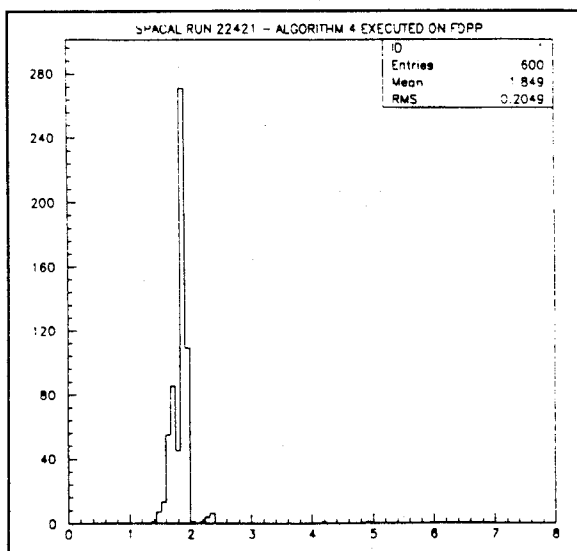


Figure 30a. Distribution of R_p for 600 electron events.

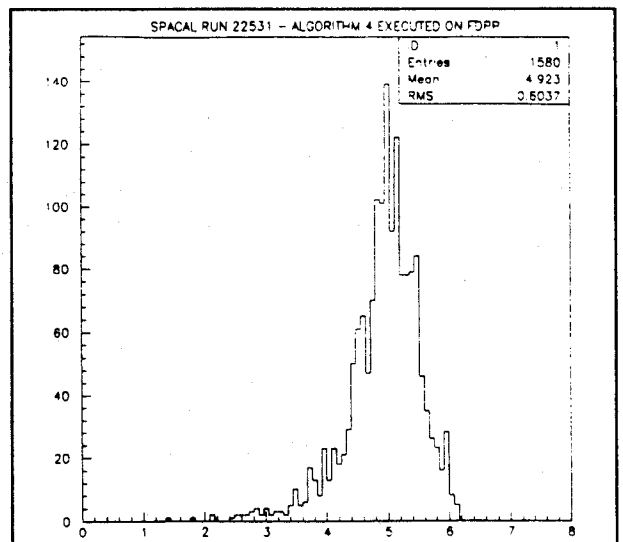


Figure 30b. Distribution of R_p for 1470 jet events.

6. CONCLUSIONS.

The tendency of the market today is to produce DSP's with 16-bit integers and 32-bit floating arithmetic, while in the past DSP of 24-bit or 22-bit or 28-bit were also made.

Generally speaking the use of General Purpose DSPs in place of Special Purpose DSPs, is recommended due to the facility to find software development tools and application libraries.

The DSP is used in High Energy Physics in two main fields:
Accelerator Control and trigger/data acquisition systems.

In the accelerator control there are several types of applications:

- OPEN-LOOP:

- a) simple measurements
- b) set parameters, generate control

- CLOSED-LOOP

- c1) slow reaction time in changing currents of magnets (typically of the order of seconds).
- c2) fast reaction time used in measurement techniques (e.g. PLL of LEP/BI group).

A parallel processing architecture with fast data transfer may be required in case it is necessary to correlate results of measurements, but most of the time DSP applications in accelerator control are of the stand-alone type. Efficient architecture (FDPP_DM) as the one described in chapter 5.1.2. can solve most of the problems listed below without requiring the use of Special Purpose DSPs:

- a) FFT (deferred-time) requires direct parallel transfer to dual-port memories in order to match with the DSP processing time (DMA up to 150 Mbyte/s for the FDPP_DM).
- b) convolutions (FIR, IIR) and closed-loop control, require direct parallel I/O into the DSP.
- c) measurement correlation requires an easy to use operating system from the user point of view, but at the same time a fast data transfer and a low cost processing capability. (In the example described in section 5.1.2., the UNIX like operating system, HELIOS, running on Transputers offers a friendly interface to the user, while the DMA to the memories offers good parallel transfer speed and the DSP offers the calculation capability at very low price).

Examining the applications that are under development in the Accelerator control, the conclusion that can be drawn is that there is a tendency toward digital control of the system. From this inventory, it turns out that experience is being gained on a variety of different components. In the applications mentioned the following DSPs have been used: Analog Devices, AT&T, Motorola, NEC, Texas Inst., Thomson, Zoran, NEC.

In the trigger and data acquisition system DSP's are used for simple signal analysis, signal correlation and data compaction. Algorithms applicable to a variety of situations in the field have been analyzed and published [43].

Trigger and data acquisition Systems in High Energy Physics Experiments require exploiting the field of embedded processors and parallel processing.

- Embedded processors usually must be able to respond to events very quickly, must be very compact and must make code debugging easy, even during real-time operation. In the past these needs have been met by microcontrollers at the low end and bit-slice design at the high end.

- RISC, DSP, CISC, TRANSPUTERS and high performance EMBEDDED CONTROLLERS can be used in embedded systems.

- Even if their throughput is different, the task in some applications can be well satisfied by more than one type of these processors.

- To find the optimal solution for a problem, a careful investigation of the possibilities of each type of processor must be made. The investigation must certainly include the processor instruction set, speed and internal architecture to determine what is best suited for the application algorithms or for the needs of a more general project.

- More decisive in determining the overall throughput in a project, can be the harmonious interaction and intercommunication of the various components rather than the speed of any single one. Thus the real effort in designing a specific application or project should be done in defining

"THE MULTIPROCESSING ARCHITECTURE"

for the best performance solution to the application or project.

In general, resources must be distributed according to necessity. In the earliest applications in High Energy Physics, DSPs have been used as embedded processors in a tightly coupled parallel processing system, for moving data or data compaction. Without having a backbone operating system that can correlate the results, all programs had to be written specifically to communicate from one DSP to another through FIFOs or dual-port memories. The attempt now with the FDPP parallel processing system is to give to the user the backbone of an operating system with the possibility to use in a easy way the calculation power and I/O facility (DMA) of a DSP.

Acknowledgements.

The author wishes to thank the experimental physicists of the SPACAL collaboration for making their data available for this test, as well as the persons of the various accelerators control groups and experimental physics groups, who provided the informations necessary for preparing these lecture notes.

A special thanks to C. Verkerk and A. Werbrouck for proof reading this document and for many helpful suggestions and encouragement in carrying on this work.

References

- [1] A.V. Oppenheim and R.W. Schafer, Digital Signal Processing, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1975.
- [2] L. R. Rabiner and B. Gold, Theory and application of Digital Signal Processing, Prentice-Hall Inc., Englewood Cliffs, N.J., 1975.
- [3] A.V. Oppenheim et al., Signals and Systems, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1983.
- [4] M. Kunt, Traitement numérique des signaux, Georgi, CH-1813 St-Saphorin, 1980.
- [5] V. Cappellini et al., Digital filters and their applications, Academic Press, London, 1978.
- [6] L.B. Jackson, Digital filters and signal processing, Kluwer Academic Publishers, 1986.
- [7] P.A. Lynn, Introductory digital signal processing with computer applications, Chichester, Wiley, 1989.
- [8] R.J. Higgins, Digital signal processing in VLSI, Englewood Cliffs, N.J., Prentice-Hall, 1990.
- [9] B. Gold and C.M. Rader, Digital Processing of Signals, McGraw-Hill, New York, 1969.
- [10] J.F. Kaiser, System analysis by digital computer, J. Wiley and Sons, New York, 1966.
- [11] M.G.H. Katevenis, Reduced instruction set computer architectures, MIT Press, Cambridge, Mass., 1985.
- [12] J. Hennessey et al., Hardware/software tradeoff for increased performance, ACM Proc. Symp. Architectural Support for Programming languages and Operating Systems, Mar. 1982.
- [13] D.W. Wall, Register windows vs. register allocation, Proc. ACM SIGPlan 88 Symp. Programming Language Design and Implementation, June 1988
- [14] D.R. Ditzel and H.R. Mc Lellan, Branch folding in the CRISP Microprocessors: reducing branch delay to zero, Proc. 14th Ann. Symp. Computer Architecture, CS Press, June 1987.
- [15] D. Boussard, Signal analysis, CERN Accelerator School, CAS, Capri, (October 1989).
- [16] K.D. Lohmann, M. Placidi, H. Schmickler, Design and functionality of the LEP Q-meter, CERN/SL/90-32 (BI)
- [17] A. Burns, et. all, The BOSC project, European Particle Accelerator Conference EPAC 90, Nice (12-16 June 1990).
- [18] D. Crosetto, Fast digital parallel processing module (FDPP), CERN/DD 89-33 (AC) (1989).
- [19] P.G. Innocenti, Microprocessors in the LEP Control System, Contrib. to the International conference on the impact of digital microelectronics and microprocessors on particle physics, Trieste, (28-30 March 1988).
- [20] A. Bland, R. Rausch, The controls architecture and equipment access of LEP, CERN SL/90-53 (CO) (1990).
- [21] D. Crosetto, A fast cluster finding system for future HEP experiments, CERN/ECP 90/14 (1990), submitted to Nucl. Instr. and Methods.
- [22] E. Schulte, private communication.

- [23] D. Boussard, Schottky noise and beam transfer function diagnostics, CERN Accelerator School CAS, Oxford, CERN 87-03, 1987.
- [24] A. Chapman-Hatchett, private communication.
- [25] O. Berrig, P. Campiche, H. Dearman, A. Dinius, S. Oliver, private communication.
- [26] G.P. Ferri et al., Commissioning and operating experience with the interaction rate and background monitors of the LEP e^+e^- collider, European Particle Accelerator Conference, Nice (12-16 June 1990).
- [27] H.J. Stuckenberg, DESY - F56, private communication.
- [28] D. Crosetto, Fast Digital Data Processor (FDDP), a modular system for parallel digital processing algorithms. CERN-EP/87-151 (1987).
- [29] D. Crosetto et al., Parallel arrays of digital signal processor as central decision element for upper level triggers in High Energy Physics experiments, IEEE Transaction on Nuclear Science, February 1988.
- [30] F. Bertolino et al., A DSP based FASTBUS board for data acquisition or high level trigger control in DELPHI electromagnetic calorimeter, conf. on real-time computer application, Williamsburg, VA, (15-19 May 1989).
- [31] D. Bulfone et al., Proposal for the FEMC Trigger, DELPHI 86-100 DAS-42.
- [32] P. Battaioetto et al., A FASTBUS module based on a Digital Signal Processor and on programmable gate arrays for triggers applications in high energy physics, submitted to Nucl. Instr. and Methods, (1990).
- [33] V. Roveda, A. Werbrouck, Cluster analysis with a fast microcomputer, Microprocessing and Microprogramming, Vol. 22, 1988, pp. 347-351..
- [34] J.P. Vanuxem, private communication.
- [35] N. Bingerfors and M. Burns, SIROCCO IV, Hardware and software manual, DELPHI/88-48 TRACK 48 (1988).
- [36] F. Tosello, private communication.
- [37] A.J. Lankford, Issues for trigger processing at high-luminosity colliders, ECFA Study Week on Instrumentation Technology for High-luminosity Hadron Colliders, Barcelona (14-21 September 1989).
- [38] S. Cittolin et al., The third level trigger and output event unit of the UA1 data-acquisition system, Contrib. to the International conference on computing in high energy physics, Oxford, (10-14 April 1989).
- [39] I. Gaines, Microprocessors and other processors for triggering and filtering at the SSC, FERMILAB-Conf-89/52 [SSC-207] (1989).
- [40] T. Nash, Computing possibilities in the mid 1990s, Conf. on Future directions in Detector R&D for experiments at pp colliders, Snowmass, Colorado, (5-7 July 1988).
- [41] J. Ballam, Future plans for HEP computing in U.S., Computing in High Energy Physics, Elsevier Science Publisher B.V., 1986.
- [42] C.W. Fabjan, Signal processing for detectors at high-luminosity hadron colliders, CERN-EP/90-85 (1990).
- [43] R. Bock et al., Benchmarking computer architectures for HEP real-time algorithms, CERN-LAA/RT/88-08
- [44] K. Caesar, U. Schmidt, S. Mehrgardt and T. Himmel, Parallel computing made efficient, Elektronik magazine 12 (8 June 1990).
- [45] SPACAL Collaboration, Scintillating fibre calorimetry at the LHC, Proposal CERN/DRDC 90-23, DRDC/P1
- [46] S. Buono and D. Crosetto, Test results of real-time algorithms executed on FDPP with SPACAL data, CERN/ECP 90-6 (1990).