

Parallel computer's

R.W. Moorey and P.R. Teetsie

Bit serial arithmetic

"The arithmetic was serial with one bit carrying every clock period of 1 μ s"

NB { (Pilot ACE, Wilkins on 1953)
English Electric Sence (Haley 1956) - it
commercial derivative

"But parallel arithmetic became a practical part of computer design with the availability of static random-access memories from which all the bits of a word could be read conveniently in parallel"

NB "The first experimental machine to use parallel arithmetic was finished at the Institute of Advanced Studies (IAS) in 1948 and this was followed in 1953 by the first commercial computer with parallel arithmetic the IBM 701"

"In the IBM 704, along with other machines of its time, all data read by the input

equipment is written to the output equipment had to pass through a register in the arithmetic unit, thus preventing useful arithmetic from being performed at the same time as input ~~and~~ or output."

"... However these tape speeds were still approximately 1000 times slower, than the processor could manipulate the data, and the input/output could be a major bottle-neck in the overall performance of the ~~I/O~~ IBM 704 and of the computer installation as a whole."

"The I/O problem was at least partially solved by allowing the arithmetic and logical unit of the computer to operate in parallel with the reading and printing of data.

A separate computer, called an I/O channel, was therefore added whose sole job was to ~~the~~ transfer data to or from the slow

NA

29.VII.83 Friday -2-

Abstracts

peripheral equipment, such as card readers, magnetic tapes, or line printers, and the main memory of the computer. Once initiated by the main control unit, the transfer of large blocks of data could proceed under the control of the I/O channel whilst useful work was continued in the arithmetic unit.

The I/O Channel had its own instruction set, especially suited for I/O operations and its own instruction processing unit and registers.

Six such channels were added to the IBM 704 in 1958 and the machine was renamed the IBM 705.

This is therefore an early case of multi-processing".

"The IBM 705 was ~~was~~ re-engineered in transistor technology and marketed in 1955 as the IBM 7090. This machine, together with

30 VII.83. Saturday -2a

upgraded versions (IBM 7054 and 7054 II)
was extremely successful and some 400 were
built and sold

X X
X

Stretch (Dunnell 1956, Blieh 1955)

"Its principal novel features were a look ahead
facility to pick up, decode, calculate
addresses and fetch operands several instructions
in advance, and the division of memory
into two independent banks, that could
send data to the arithmetic units
in parallel. The maximum transfer
rate of data to and from memory was
thereby increased by a factor equal to
the number of memory banks. This was
the first use of parallelism in memory and
enabled a relatively slow magnetic core
memory to be matched more satisfactorily
to the faster processor. Almost
all subsequent large computers have used

30.VII.83 Saturday -3-

Abstracts

banned (sometimes called interleaved) memory
of this kind

"The first ~~&~~ STRETCH was delivered to Los
Alamos in 1961, but did not achieve
its design goals ... the computer was
withdrawn from the product range"

"The dramatic ^{xxx} success of the CDC6600 in
replacing IBM 7050's and converting
most large scientific centres to a rival
company, made IBM respond. It was not
until 1967, however, that the IBM 360/91
(Anderson ~~et~~ al 1967) arrived with a
performance of about twice that of CDC6600.
This machine has the even-ahead facility
of STRETCH and like the CDC6600 had separate
execution units for floating point
and integer address calculation, each of which
were pipe lined and could operate in
parallel. The principle of pipe
pipelining was also introduced to speed up

30.VII.85 Saturday -39-

Abstracts

the processing of instructions, the successive operations of instruction fetch, decode, address calculation and operand fetching being overlapped on successive instructions

In this way several instructions were simultaneously in different phases of their execution as they flowed through the pipe line.

The IBM 360^{X₁X₂}/195 (Murphy and Wade 1970) combined the architecture of 360/91 with the cache memory

The idea of introducing a high-speed buffer memory (or cache) between the slow main memory and the arithmetic registers goes back to the Ferranti Atlas computer (Fotheringham 1961)

The cache, 32768 words of 162 ns semiconductor memory in 360/85, held the most recently used data in blocks

30.VII.83 Saturday -4-

Abstracts S.C. Hoare

of 64 bytes. If the data required by an instruction were not in the cache, the block containing it was obtained from the slower main memory (4M bytes of 756 ns core storage, divided into 16 independent banks) and replaced the ~~to~~ least frequently used block on the cache. It is found in many large-scale calculations, that memory ~~can~~ references tend to concentrate around limited regions of the address space. In this case most references will be to data in the first cache memory and the performance of the 4M byte slow memory will be effectively that of the faster ~~of~~ cache memory."

"Gene Amdahl, who was chief architect of the IBM 360 series (Amdahl et al 1964) formed a separate company in 1970 (the Amdahl Corporation) to manufacture a range of computers that were

30.VII.85. Saturday -4a-

P.C. Kromm

compatible with the IBM 360 instruction code and could therefore use IBM software

... the Amdal 470 V/6 was first to use ~~the~~ large scale integration (LSI) technology for the logic circuits of CPU

bipolar emitter coupled logic, ECL, with 100 circuits per chip

and for this reason is sometimes

called a fourth-generation technology computer.

... Although the arithmetic units on this machine were not pipelined, a high throughput of instructions was obtained by pipelining

the processing of instructions. The execution

of instructions was divided into 12 sub-operations that used 10 separate circuits

when flowing smoothly or new instructions could be taken every two clock periods (or 64ns) and therefore

up to six instructions were simultaneously

4e
new name

higher
pipelined
for instructions

NP
not code, but
arithmetic

30.VII.83 Saturday -5-

S.C. Honey

in different phases of execution and could
be said to be in parallel execution

A high speed buffer (or cache) bipolar memory
of 16 Kbytes (65 ns access) improved the
effective access time to the main memory
of up to 8 Mbytes of MOS store (650 ns access) "

~~Pipelined vector~~ $\times \times \times$

Pipe lined vector computer

" The Cray I (Russel 1978, Dangworth 1979)
followed the evolutionary trend from the

6600 and 7600

It provided 12 functional
units, now all pipelined, a fast ^{with} local ~~period~~
12.5 nsec., and a 16-bank one million
word bipolar memory with 950 ns cycle time

The principal novel feature was the provision
of eight vector registers, each capable
of holding 64 floating point numbers
(64 bits long) and a set of about 32
machine instructions for manipulating

30 VIII 85 Saturday - 5a

S.C. Hoerny

and performing arithmetic on these vectors
Three functional units were reserved for
vector operations (shift, logical and addition)
and three shared with scalar instructions
(floating point addition, multiplication
and reciprocal approximation)

31. VII. Sunday ~~6a~~ - 5a -

"The Cray-1 was therefore referred to as a
vector computer

"Two other pipelined computers have
earlier origins than the Cray-1, namely
the CDC STAR 100 and the ~~For~~ Texas Instrum-
ments Advanced Scientific Computer
TIASC (Theis 1974, Mooney 1977)

The STAR 100 (Hintz and Tabe 1972) was
conceived about 1964 as a processor for
vectors with an instruction set based
on Iverson's ~~APX~~ (1962) APX language, that
would with pipelining be able to
sustain rates of 100 Mflop/s

31.VII.83 Sunday -6-

Sc. Mooney

on the long vectors that were common to many scientific problems at the Lawrence Livermore Laboratory (LLNL)

A letter of intent was received in 1967 and design started after a long gestation period of six years the first machine was operational in 1973 and two machines were ~~ordered~~ delivered to LLNL in 1977-78.

There were two major hiatuses, one technological and one in the design itself, which ~~led to~~ ~~lead~~ led to the slow development and when available the STAR 100

~~It~~ suffered from its very start in the mid-1960's.

The 1,2 μ s magnetic-core memory had been surpassed by semiconductor memory, and its clock with an 80ns period was slow compared with its competitors by the time the machine was available

in the mid-1970's. Furthermore the best serially organized computers such as CDC 760 and IBM 360/195 had much faster arithmetic units

(repeated) Insufficient

31.VII.83

-69-

for scalar operations and for operations
on any but the longest vectors

The STAR 100 was only able to outperform
these more general-purpose competitors on
very carefully prepared code, with ~~over~~ vectors
of several hundred or thousand elements

The STAR 100 however has been completely
reengineered in LSI with semiconductor memory
and was introduced to the market in

1979 as the CYBER 203 E (Kasick 1975)

This machine has been ~~renamed~~ renamed
the Cyber 205 and is now highly competitive
with the Cray-1.

The Cyber 205 differs from the Cray-1
in processing all vector instructions
to and from ~~vector~~ main memory

(there ~~are~~ are no vector registers), however

it has multiple general purpose
pipelines, as opposed to the specialised
pipelines of the Cray-1

32-bit and 16-bit instructions

It

31.VII.83.

Sunday -7-

P.C.
Hickey

"A major milestone in the history of parallelism is certainly the paper of Slotnick et al (1962) entitled "The Solomon computer"

The acronym stands for Simultaneous Operations Learned Ordinal Modules Network ~~etc~~, and describes a two-dimensional array of 32×32 processing elements, each with a memory for 128 32-bit numbers and an arithmetic unit working in a bit serial fashion, under control of a single instruction stream in a central control unit. Contrary to the evolutionary development of the serial computers or the vector pipe line computers, the Solomon concept was a radical change in thinking on computer architecture, and had a substantial influence ~~as~~ on computer science research as well as on computer design.

31.VII.83. Sunday -7a-

D. C. Twomey

The 'SOLOMON' computer was never built exactly as described in the 1962, but gave birth not only to the ILLIAC IV and Burroughs PEP floating point processor arrays, but also to the Goddard Aerospace STARAN and TOLAP arrays of one bit processors which are ^{often} called associative processors

The US Department of Defense's Advanced Research Projects Agency (ARPA) awarded a contract to the University of Illinois for the design of a SOLOMON type computer in 1966, and this computer then became known as the ILLIAC IV

It was to comprise four quadrants each with a control unit interpreting a single stream of instructions for 64 floating-point processing elements (PEs)

31.VII.83 Sunday

-8-

P. C. Mooney

Each PE was to have 2000 64-bit words of thin-film memory and the PEs in each quadrant were connected as an 8×8 array. The four quadrants were to be connected by a highly parallel I/O bus and backed up by a large disk as secondary memory from which jobs ~~would~~ would be read and to which results would be written. Each of 256 PEs

was envisaged to produce a floating-point operation in 240ns so that a maximum rate of 2 Gflop/s was planned.

... The fortunate story of the ILLIAC IV is described by Faden (1976) in his article Reaching for the Giga-flop

The machine was the first to use semi-conductor memory chips (256-bit bipolar logic gates from Fairchild) & for all its main memory, after it was discovered, that there was insufficient space

31.VII.83 Sunday -89-

for the thin-film memories that were originally proposed. The bipolar logic gates were originally intended to be packed about 20 per chip (MSI) but this had to be reduced to about 7 gates per chip (SSI).

This was due largely to the decision to pioneer the new and faster Emitter-coupled logic (ECL) rather than established transistor-transistor logic (TTL). Also JLLIAC IV also pioneered the use of 15-layer circuit boards and computer-aided layout methods that proved necessary to wire them.

The manufacture of JLLIAC IV (now reduced to one quadrant) was entrusted to Burroughs and the machine was delivered to NASA Ames Research Center, California in 1972.

... The JLLIAC IV may be regarded as a failure in that it cost four times the original contract figure and did not come even within a factor of 10 of its originally proposed performance.

31.VII.83 Sunday -9-

H.C. Hoernsey

ILLIAC was, ^{like} ~~as~~ many major changes, too ambitious for the technology of its time. The computer software developed ⁱⁿ ~~with~~ association with ILLIAC IV was also considerable, including four computer languages that could express the parallelism of the computer, and much work on the development of suitable algorithms for standard mathematical problems which contain parallelism, such as matrix manipulations and the solution of partial differential equations (see f.e. Kuen 1968).

The computer languages were the ALGOL-line TRANQUI (Abel et al 1967) and GLYPHIK (Lawrie et al 1975), the PASCAL-line ACTUS (Perron 1978) and CF2 FORTRAN (Stevens 1975). The Burroughs Corporation had played a major role in the development of array-line parallel processors. This began with the ILLIAC IV, for which Burroughs was system contractor in the period 1965-75.

continued with the PEPE // processor for the US Army (delivery began 4 in 1976) and culminated with the announcement in 1977 of the Burroughs Scientific Processor (BSP) as a commercial venture for the general scientific market, in competition with the pipe-lined designs from Cray Res. Inc. (Cray-I) and CDC (Cyber 205). We will now describe the relation of the PEPE and BSP designs to the ILLIAC IV.

The ILLIAC IV was designed for the solution of partial differential equations and can be described as an 8×8 array of 64-bit floating point processing elements each with 2K words of memory working in bicontinuous fashion with nearest neighbour connections and controlled by a single instruction stream processed in a central control unit. PEPE, the Parallel Element Processor Ensemble, was designed, on the other hand, to control

31. VII. 83.

-10-

// C. Hoenney

a ballistic missile defense system of radar detectors and missile launchers (Berg et al 1972, Cornell 1972, Vica and Cornell 1978). PEPE had its origins in work at Bell Laboratories, Whippany, on content-addressable distributed memories (Lee and Paolet 1963) combined with floating-point processing (Crane and Githens 1965) which led in about 1972 to the building of an experimental machine (PEPE IC) with 16 32-bit floating point processing elements (Crane ~~and~~ et al 1972)

The full size PEPE was manufactured by Burroughs and comprised loosely coupled system of 288 PES, each containing three processors (one each for input of radar signals, processing of data, and output of control signals) controlled in even step fashion by three control units were connected to three standard I/O channels

31. VII.83

-109-

//C

of a CDC 7600 which acted as host
to the complete systems

When operating, each target ~~was~~ that was
identified became the responsibility of one
PE and, because there were no ordered

connections between the targets, no direct
connections were provided at all between
the PEs. When necessary, ~~communication~~

communication between the PEs took
place between via the memories of
the control units. The array of processors

was then ~~star~~ said to be unstructured
and the word ensemble was ~~coined~~
coined for this arrangement.

Since each PE had a floating-point
processing rate of 1 MFlop/s the complete
PEPE could be rated as a potential
maximum computing of 288 MFlop/s

More realistic estimates for actual
problems ~~can~~ lead to estimates of
about 100 MFlop/s (Vick and Cornell 1978)

31. VII. 83 Sunday -11-

11C Memory

one of the problems with the ILLIAC IV is the delay in routing data long distances across the array, caused by the limited nearest-neighbour connections between the 65 processors and the 64 banks of PE memory. In their commercial design, the DSP (Tenser 1978, Auston 1979), Burroughs have reduced the number of processors to 16 and the number of memory banks to 17. This smaller number makes it possible to provide connections via an "alignment network" between any ~~of~~ processor and any of ^{the} memory banks.

1. VIII. 83 Monday

The choice of a prime number of memory banks different from the number of processors allows the use of mapping algorithms that reduce the number of memory conflicts that may arise in common matrix manipulations.

1 VIII.83. Monday -11a-

The PEs themselves are serially organized floating point processors with an addition or multiplication time of 320 ns for the production of 16 results (one in each PE). The careful overlapping of ~~write~~ the read, write and arithmetic, together with the provision of all memory bank to PE connections, is expected to eliminate most bottlenecks; and the MSP is designed thereby to sustain a large fraction of its maximum processing rate of 50 MFlap/s on the majority of problems. MSP is chosen for detailed study in § 3.4. but was withdrawn from the market in 1980 before any had been sold.

ARPA and NASA jointly established an Institute of Advanced Computations (IAC) to support the ILLIAC and in 1977 this institute published a design proposal for a machine called PHOENIX.

1.VII.85 Monday

-12-

(Feierbach and Stevenson 1975b) to replace the ILLIAC IV in the mid 1980s. IAC foresaw the need for a 10^9 GFlops machine in order to solve three dimensional problems in aerodynamic flow with sufficient resolution. The P-X design can be described as 16 I-C IVs each executing their own instruction stream under ^{the} control of a central control unit. If each PE can produce a result in 100ns (a reasonable assumption with 1980 technology) the total of 1024 PEs could produce the required 10^{10} operations per second.

NASA has also commissioned two other design studies, from Control Data Corp. and Burroughs, for machines to replace the ILLIAC IV and form a ~~numerical~~

Numerical Aerodynamic Simulation Facility (NASF) for the mid 1980s (Stevens 1975). The CDC design is based on an upgraded four pipe C3000 295

1. VIII. 83 Monday -12a-

operating in lockstep fashion, plus a fifth pipe as an on-line spare that can be electronically switched in if an error is detected.

Each pipeline can produce one 64-bit result or two 32-bit results every 8 ns however

each result may be framed from up to three operations, leading to a maximum computing rate of 3 GFlop/s. There

is also a fast scalar processor clocked at 16 ns.

In contrast, the Burroughs design may be regarded as an upgrade to the BSP architecture, being based on 512 PEs connected to

521 memory banks (the next larger prime number).

Unlike the ILLIAC IV each PE has its own instruction

processor. The same instructions are assigned to each processor, but the arrangement does permit them to

3/8/83
by
mm
by
mm

1.VIII.83 Monday - 13-

be executed in different sequences depending on the result of data dependent conditions that may differ from process to process with a ~~the~~ planned floating-point addition time of 290 ns, multiplication time of ~~240~~ 360 ns and 512 processors a maximum processing rate of about 1-2 gFlop/s could be envisaged. EC2 technology is planned with a 40 ns clock period.

The initial SOLOMON computer design (Slobnick et al 1962) was a 32×32 array of one bit processors each with 4096 bits of memory, conducting its arithmetic on 1024 numbers in parallel in a bit serial fashion. This describes quite closely the pilot model of the JCL Distributed Array Processor (DAP)

that was started in 1972 and commissioned in 1976 (Flanders et al 1977, Reddaway 1979). The first production model of the machine was installed at Queen Mary College London in 1980. It comprises a 64×64 array of PEs and forms a memory ~~for~~ module of a host JCL 2980. As with Solomon

1.VIII.83 Monday -139- //C Memory
As with the SOLOMON and the TLLAC IV, the PEs
are connected in a two-dimensional lattice
with nearest-neighbor connections. The original
DAP used small-scale integration with 16
PEs and their memory per circuit board.
Large-scale integration with four PEs per
chip makes larger arrays possible and
 128^2 or 256^2 DAP's are proposed.

NS An important feature of the engineering
design of the TLLAC DAP is that
the PE logic is mounted on the same
printed circuit board as the memory to
which it belongs.

An attractive possible development using
VLSI technology is to include the
PE and its memory on the same chip.
In the von Neumann concept the logic
and memory are both conceptually and materially
in different units (or even separate cabinets)
which can lead to severe bottlenecks
of the transfer rate between the two
units. It is inadequate.

1. VIII, 85 Monday -14- //C Moevery

In the ICL DAP, as its name implies the logic is distributed into the memory where it is adjacent to the data it is to manipulate. The advent of VLSI which allows

approximately 10^5 logic gates to be included on one chip, has made the distribution of logic into memory a practicable proposition. Indeed this

has become almost a necessity because of the problems associated with interconnections between chips

Stone (1970) proposed a logic in memory computer in which each of the 16 Kbyte segments of the cache memory of an IBM 360/85 was to have special-purpose logic that would process the 16 segments in parallel. This could be

used as an associative memory or for simple arithmetic operations on 16 elements in fast parallel. In

a limited sense the proposal was to for a LUP E DAP without the

1. VIII. 83 Monday - 199 -

processors interconnections Kautz (1971)

proposed a more extensive logre-in-memory computer called an Augmented Content Addressed Memory (ACAM) with special circuitry for sorting, matrix inversion, fast Fourier transform, correlation etc.

The engineering of the LSC chips was to be by cellular arrays with a universal logic capability, implementing as far as possible in hardware special algorithms that had been devised for parallel computation; for example fast Fourier transform (Pease 1968) and matrix inversion (Pease 1971)

Another influential paper in the history of parallelism is that of Shuman (1960) entitled "Parallel Computing with Vertical Data" in which he describes an "orthogonal computer" organization (see also Shuman 1970). The arithmetic units of a conventional serial computer take their data serially from memory in

1.VIII.83 Monday -15- H/C. Moxness
the form of words (e.g. 32-bit fl. point numbers)
and by 1960 most scientific computers
would process the bits of the words in parallel.
Such a procedure may be described as word serial
and bit parallel processing.

Shoeman recognized that many problems
involving information retrieval required
searches on only a few bits of each
word and that conventional word-serial
processing was inefficient. He proposed, ~~that~~
therefore, that the memory should ~~be~~ also be
referenced in the orthogonal direction, i.e.
across the words by bit slice. If the
bits in memory are thought of as a two
dimensional array, with the bits of the n -th
word forming the x th horizontal row, then
the l th bit slice is the bit sequence
formed from the l th bit of each
~~memory~~ ~~word~~ number - that is to say
the l th vertical column of the array.
In the orthogonal computer, one PE is provided
for each word of memory and all the
bits of a bit slice can be processed
in parallel.

2. VIII. 83 Monday -15a-

//C Memory

~~This~~ This is called bot-serial and word-parallel processing. The orthogonal computer provided a "horizontal unit" for performing word-serial / bot-parallel operation and a separate "vertical unit" for bot-serial / word-parallel operation.

2. VIII. 83

~~Th~~ Tuesday (01.45)

The idea of performing tasks in parallel on all words has led to the idea of the associative or content-addressable memory, in which an item is referenced by the fact that part of its contents match a given bit pattern (or mask), rather than by the address of its location in store.

In a purely associative memory, there is no facility to address a data item by its position in store. However many systems provide both forms of addressing.

MS The multitude of different processors based on associated memories have been reviewed by Thurber and Wald (1975)

2.VIII.83 Tuesday -16-

and Yan and Fung (1977) and the reader is also referred to the books by Foster (1976) and Thurber (1976) for a more ~~complete~~ complete treatment.

The OMEN (Orth. Mini Embedded) series of computers were a commercial implementation of the orthogonal computer concept, manufactured by Sanders Associates for signal processing applications (Highne 1972).

The OMEN-60 series used a PDP-11 for the conventional horizontal arithmetic unit and an array of 64 PEs for the associative vertical arithmetic unit which operated on byte slices, rather than bit slices. Depending on the model, either bit serial arithmetic with eight bits of storage were provided with each PE or alternatively hardware floating point with ~~eight~~ eight 16 bit registers and five mask registers.

2.VIII.83 Tuesday -16a-

//C Memory

Logic was provided between the PES to reverse the order of the bytes within a slice or perform a perfect shuffle or barrel shift

Another computer arrived from the nThugour computer concept was the Goodyear STARAN (Bader 1975) which was conceived in 1962, completed in 1972 and by 1976 about 400 had been sold. The STARAN

~~typically~~ typically comprised ~~four~~ four array modules, each with 256 one-bit PES and between 64 Kbits and 64 Mbits of total storage, controlled by a sequential PDP-11. Unlike the

SOLOMON, however, the storage was not assigned to specific PES, instead a flexible "FLIP" network was interposed between the PES and the memory

NB computer not available

2.VIII 1983

-17-

HC Mackney

A slice of 256 bits was selected from the memory in ~~off~~ a pattern specified, under program control, by a 256-bit code. The pattern selected may, for example, have treated the store as a multidimensional array with a varying number of dimensions, or shuffled the data in the manner required by the fast Fourier transform and other important numerical algorithms. Connections between the PEs were achieved by passing the 256-bit slices of data through the FLIP network, thus achieving in minimum time a highly flexible effective interconnection pattern that could be varied from problem to problem by the programmer. The STARAN, like other bit-oriented computers, was most effective when performing logic and short ^{word-length} integer arithmetic. A particularly notable application is the digital processing of pictures, in which the image

2.VIII.83. Tuesday -17a-

11C. Twenty

is divided into millions of pixels (picture elements) each of which is represented by 6-12 bits. The first STARAN was

delivered to the Royal Air Force base for such an application and it has also been proposed for air traffic control

The STARAN and DAP concepts have been brought together in proposals from Gerdyn Aerospace Corporation for a Massively Parallel Processor (MPP) which

comprises a ~~132~~ 132 x 128 array of one-bit PES connected two-dimensionally

The machine is designed principally for picture processing of satellite photographs at rates 10⁶ pixels per second

Eight- and twelve-bit integer arithmetic is

designed for execution at the rate of a few Mflop/s and 32-bit floating point arithmetic between 200 and 400 Mflop/s.

2.VIII.83 Tuesday -18-

HC Hoxney

(Personal comment:)

~~Some~~ conjectures and problems

1. The computers, based on orthogonal concept, can be applied to solving some old problems.

In the PIC method, the ^{first} fractional step ~~was~~

~~the conversion~~ for calculation of ^{integrating}

large particles trajectories (Lorenville system)

can be realised by ~~the~~ bit-serial

word-parallel processing (sorting, searching

for lagrangian marked particles, belonging

to the ~~the~~ numerical cell.

The second fractional step ^{to describe} ~~for~~ ^{contouring}

~~the new velocity components~~ ^{of a depression}
of the field ^(pressure, charge, potential a.s. O.) ~~the~~ ^{the} distribution

of the ~~the~~ particles, can be realised
(on scalar computers)

by word-serial bit-parallel processing

In the ~~the~~ pipe - line or other

computer both words and bits can be treated parallel

The alternative ^{particles - cells} ~~the~~ ^{the use of} orthogonal concept can be solved on

2.VIII.83. Tuesday -18a-

HC Moevery.

orthogonal concept is near to Lebesgue
approach : integrating can be realised
not only by Riemann's, but by the
Lebesgue's processing

The mapping Lagrange ^{ensemble} - eulerian mesh
is to be implemented by or on
the basis of orthogonal concept.

NB: Giguier's idea to apply
FFT for solving Boltzmann
equation ~~is to be evaluated~~
can be realised, it seems, effectively by
parallel processing on the basis of
orthogonal
concept

The same is valid for ^{nonlinear} interpolation
operator. To facilitate solving
(search for Lagrangian marked particles)
mesh adaptation is to be applied

Cheap special-purpose computers

Another line of computer evolution involving parallelism has been the development of relatively cheap, special-purpose computers for processing arrays of data, which are therefore often referred to as "array processors".

Note that, in this context, this name does not imply that the computers are architecturally arrays of processors, in fact most designs use the ~~array~~ pipe line principle.

The main application is to signal processing and the analysis of seismological data and

the most ubiquitous algorithm, round which the the hardware is designed is the fast Fourier transform (FFT)

The first such processors were augmented 16-bit minicomputers, which permitted parallel access to both data and instructions. Bipolar technology was used to enhance

2. VIII. 83.

-199-

the performance is rates about 5×10^4 operations per second. The next phase of development was the design of special-purpose "function boxes" for the FFT and similar algorithms. Parallelism was ^{extensively} ~~extremely~~ used in these function boxes, which typically comprised multiple program, data and coefficient memories that could be simultaneously accessed and multiple adders and multipliers interconnected in the way required for the particular algorithm. The function boxes were controlled by microprograms, that were installed at the factory. The machines were thus almost totally inflexible, but they ~~are~~ were able to ~~produce~~ provide a factor of 10 improvement in performance to about 0.5 Mop/sec.

In 1973 a new generation of array processor emerged in which multiple processing units were interconnected

NB:
particular,
not special

2.VIII.83 Tuesday - 20 - //C Mockney
by a limited number of data buses and
run asynchronously under the overall supervision
of a central control unit. Processing speeds of
around 5 Mop/s were achieved, however the
asynchronous operation led to major
difficulties of timing and non-reproducibility
of conditions in program check-out.
By far the most successful of the array
processors is the Floating Point Systems
AP-120B which has returned to the principle
of synchronous operation (Harte 1978)

The company was founded in 1970, the
machine launched in 1976 and by 1980
about 1000 systems had been installed.
The FFS AP-120B, which may be attached
to either minicomputers or main frame
computers as hosts, performs 32-bit
f-p arithm. in separate pipelined
multiplication and addition units
and 16-bit counting and address
calculation in an independent integer
arithmetic unit. Three memories
(for data, tables and program) and
two "scratch pads" of registers are provided
with multiple paths between each memory
and each arithmetic units.

3. VIII.83 Wednesday - 20a -

11c Machinery

All units of the machine are controlled in each clock cycle by a 64-bit instruction that gives precisely reproducible conditions and, unlike many earlier machines, allows the computer to be programmed for a variety of uses.

Thus the special-purpose line of development has evolved in this machine to a cheap general-purpose processor of arrays.

Typically, processing rates of 5-10 MFlop/s may be achieved at costs of approximately £50,000 (1980 prices). This is highly cost effective computing and

may enable many calculations presently prohibitively expensive for industrial use to become commonplace. In many

respects the FPS AP-120B may be considered as the "poor man's" Gray-2 being about one ~~the~~ tenth as fast and one-fiftieth of the price.

The overall architecture of both machines is quite similar, both being based on ~~the~~ multiple independent pipelined functional units.