 STATUS REPORT 1990

Massively Parallel Processing Collaboration

THE MPPC PROJECT

Presented by François Rohrbach, MPPC Spokesman

This report consists of three parts.

The first part recalls the motivations, the main objectives and the structure of the project. The highlights of the first year of activity are underlined. Initiated in early 1989, the MPPC project, in its present scheme, after formal approval in April 1990, started in July 1990 when the budget was allocated to all the collaborating institutes.

The second part of the report describes in details the R & D work performed on the Hardware by the Main Partners of the Collaboration.

The third part is devoted to the preliminary work done, using simulators, on the applications which are envisaged by every member of the collaboration for this new massively parallel processing architecture and which could be realised in the case of a successful construction of the ASP (Associative String Processors) machine prototypes.

STATUS REPORT 1990

Massively Parallel Processing Collaboration

THE MPPC PROJECT

Presented by François Rohrbach, MPPC Spokesman

This report consists of three parts.

The first part recalls the motivations, the main objectives and the structure of the project. The highlights of the first year of activity are underlined. Initiated in early 1989, the MPPC project, in its present scheme, after formal approval in April 1990, started in July 1990 when the budget was allocated to all the collaborating institutes.

The second part of the report describes in details the R & D work performed on the Hardware by the Main Partners of the Collaboration.

The third part is devoted to the preliminary work done, using simulators, on the applications which are envisaged by every member of the collaboration for this new massively parallel processing architecture and which could be realised in the case of a successful construction of the ASP (Associative String Processors) machine prototypes.

CONTENT

PART I  Highlights of the MPPC Project  7

1.1  Motivations  9
1.2  Principal objectives  10
1.2.1  Main Characteristics of the ASP  11
1.2.2  Hardware development programme  11
1.2.3  ASP evaluation  12
1.3  Structure of the project  13
1.3.1  Collaborating Institutions, Universities and Industries  13
1.3.2  Responsibilities  14
1.3.3  Coordination  15
1.3.4  Milestones  15
1.3.5  Budget and financial control  16
1.3.6  Manpower  18
1.4  Highlights in 1990  19
1.5  Prospect  21

PART II  Progress report of the hardware  23

2.1  ASPEX  25
2.1.1  The chip  25
2.1.2  The hybrid module  28
2.1.3  The operating system  30

2.2  SACLAY  31
2.2.1  Machine architecture  31
2.2.1.1  TRAX-1 machine  33
2.2.1.2  MPPC machine  34
2.2.2  The LAC controller  36
2.2.3  The ASP boards  39
2.2.3.1  Overview  39
2.2.3.2  TRAX-1 ASP card  39
| 2.2.3.3 | MPPC ASP card |
| 2.3 | **ORSAY** |
| 2.3.1 | Interfacing the ASP machine |
| 2.3.1.1 | Introduction |
| 2.3.1.2 | The Interface board |
| 2.3.2 | Minicontroller |
| 2.4 | **CERN** |
| 2.4.1 | CCD acquisition system |
| 2.4.2 | Objectives |
| 2.4.3 | Development |
| 2.4.4 | Next steps |
| 2.4.5 | Collaboration with Thomson-TMS |
| 2.4.5 | Hardware data reduction |
| 2.5 | **THOMSON -TMS, COMPOSANTS MILITAIRES ET SPATIAUX** |
| 2.5 | CCD development of Area Array CCD Image Sensors |
| 2.5.1 | TV mode image sensor |
| 2.5.2 | Full frame image sensors |
| 2.5.3 | Special technologies |
| 2.5.4 | ASICS |

**PART III**  
**Progress report of the applications**

| 3.1 | **LHC/SSC applications** |
| 3.1.1 | **CERN** |
| 3.1.1.1 | Muon selection at LHC |
| 3.1.1.1.1 | Introduction |
| 3.1.1.2 | Track-code simulation |
| 3.1.1.3 | Master point algorithm |
| 3.1.1.4 | Conclusion |
| 3.1.2 | **CERN** |
| 3.1.2 | TRD electron selection at LHC |
3.1.2.1 Progress and ongoing work on the TRD application
3.1.2.2 The TRD triggering algorithm
3.1.2.3 The possible role of the ASP in the TRD trigger
3.1.2.4 Phase I: data distribution
3.1.2.5 Phase II: low-level feature extraction
3.1.2.6 Phase III: decision taking

3.1.3 ORSAY Calorimetry, fast selection on jets at LHC, shower detection
3.1.3.1 Introduction
3.1.3.2 Hypothesis
3.1.3.3 Loading time
3.1.3.4 Rejection criteria
3.1.3.5 Preprocessing
3.1.3.6 Conclusion

3.1.4 SACLA Y First ASP algorithms for SSC/SDC
3.1.4.1 Calorimetry
3.1.4.2 Trigger decision
3.1.4.3 ASP-Trigger algorithms

3.2 Fixed target applications

3.2.1 SACLA Y NA31', second level trigger
3.2.1.1 Abstract
3.2.1.2 Proposal for the level 2 neutral trigger of NA31'
3.2.1.3 Execution time
3.2.1.4 Task parallelism
3.2.1.5 Main Controller
3.2.1.6 Data I/O
3.2.1.7 Final Decision
3.2.1.8 Simulations
3.2.1.9 Conclusions

3.2.2 GENEVA Heavy ions collisions, luminous chamber for tracking with CCD on-line cameras.
Tracks Reconstruction in Heavy Ion Collisions
3.3 Non high energy physics applications

3.3.1 EPFL Image sequence coding, data compaction for HDTV
3.3.1.1 Introduction
3.3.1.2 Activities
3.3.1.3 Results

MPPC Documentation list

Proposal, memorandums, notes and papers.
Part I

Highlights of the MPPC Project
PART I

Highlights of the MPPC Project

1.1 Motivations

The next generation of high luminosity hadronic colliders will require novel detectors, both highly time-sensitive and selective. Potentially, data will be produced at rates that are beyond anything even modern transmission technology can handle. New and unfamiliar techniques will be needed to extract the physics content from the extraordinary amount of data. The high selectivity will require, at some level of triggering, devices which allow to take decisions of a complexity comparable to what is today done in off-line programmes, but with an event frequency of typically 100 KHz. Such decisions will be based on local or global fine-grain data, and will require the execution of algorithms in extremely fast computer-like devices, with some built-in flexibility.

It is at this stage that a potential match can be identified between the needs of executing algorithms on detector data of the future, and industrial efforts to solve much more general classes of computing problems in other domains. The very general base for obtaining the necessary speed of execution is parallelism, i.e. the simultaneous execution of different instructions, or of the same instructions on different data. Both industry and computer science make serious efforts to put different kinds of parallelism to use for various suitable problems of processing.

The MPPC collaboration has concentrated on problems that are likely to benefit from massive parallelism, as aimed at by machines of the SIMD type (SIMD stands for Single-Instruction-Multiple-Data). Such massively parallel machines operate with thousands of processing elements, all maximally integrated and under a single controller, and executing the same instructions on different data.

The data flow through such machines, and the data communication between processors, thus become key elements of the SIMD processor architecture. In the MPCC collaboration, the overall architecture is, in fact, defined and implemented by a group made from both application-oriented physicists and the computer scientists at the origin of the particular SIMD model used. The collaboration targets different applications, dominated by but not exclusively taken from triggering HEP applications; EPFL, as MPPC partner, is indeed working on a first application in image processing for HDTV. More generally, it can be expected that the same basic processing elements will find their way
into quite different application fields.

Due to the open-endedness of the ASP architecture[1], based on a commercial design for a general processing element with application-oriented embedding, the impressive performance figures (in terms of cost, power, achieved density) will attract other suitably parallelized projects (e.g. relational data processing, simulation, computer vision, cellular automata, neural networks) in applications such as, high-definition TV, autonomous guiding vehicles, artificial intelligence, medicine, space science, meteorology, plasma physics, etc.

Taking advantage of the coincidence between technological opportunities (ASP, VLSI/WSI) and application needs, a Research and Development programme "The MPPC Project" has been launched between ASPEX(UK), CERN(CH), CEN-Saclay and LAL-Orsay(F), as main partners, and EPFL-Lausanne(CH), University of Geneva(CH), Brunel University(UK) and Thomson-TMS(F), as associated partners.

The first goal is to evaluate this new kind of parallel architecture, the ASP, in view of solving the bottleneck for event selection in detectors for high luminosity colliders. At this point, let us stress again the fantastic challenge for the detectors at LHC due to:

- **High Rates:** 66 MHz (15 ns between two bunch crossings),
- **High Multiplicities:** = 50 collisions per bunch crossing
  = 1100 charged particles in \pm 2 units of rapidity,
- **High Background:** a rejection of \( 10^8 \) is needed on line .

1.2 Principal objectives

The Massively Parallel Processing Collaboration (MPPC) intends to develop and exploit, in a two-year project, the new concept of associative string processing by combining all the efforts, know-how, and competence of the participating Universities, Research Institutes, and Industries.

The project will provide an assessment of the ASP potentiality in view of further collaborations with experiments in order to demonstrate the real-time performance at the trigger level using detectors in test beams. It is planned to run over a period of 18 months
starting, in each Institute, when the requested support becomes available.

The choice of the ASP, as a R & D platform for the Collaboration, was based on the exceptional potentialities offered by this new architecture.

1.2.1 Main Characteristics of the ASP

The ASP consists in strings of associative processing elements (APE) working as SIMD machines. The strings can be arranged in a loop: the architecture is reconfigurable by programmation.

Each APE is an associative memory cell with processing and communication capabilities.

APEs are addressed by content, which minimises data movement,
Parallel processing is performed on active APEs selected by programme,

The architecture is scalable up to hundred of thousands APEs, due to high integration (VLSI/WSI) and low power consumption. The target cost is low (below 5.- SFr per APE) leading, together with the low power and high integration capability, to the possibility of massive integration,

The system has maximum application flexibility and computational efficiency. It is fault tolerant: blocks of faulty APEs may be disactivated without breaking the string,

The MPPC-Array machines will have Parallel I/O capability,

ASP application programmes can be written in any block-structured language (Modula 2 is the commonly used language). An introductory course to ASP, provided by ASPEX, is useful to reach a good level on the learning curve in parallel algorithms.

1.2.2 Hardware development programme

The main hardware task is to build 4 sets of ASP with 16384 APE array which will be referred to as the "MPPC array". It will be based on the existing VASP-64 processor chip used for the TRAX-1 machine, another ASP project dedicated for off-line image processing [2,3]. The MPPC-array design allows for maximum processor element density and maximum direct parallel interfacing via conventional electronics to the readout of particle detectors. For this task, dense packages of ASP must be constructed. This is based on a modular design, using hybridation on insulators of the VASP-64 chips. These
modules are built by PolyCon(USA) ; they contain a string of 1024 APEs (16 chips) with two parallel I/O per module. These modules will be installed on boards to make 8K strings. Two ASP boards and a low level controller (LAC) in extended VME standard (in order to be compatible with existing industrial modules) are under construction for each 16-K MPPC-Array machine. These machines are built to offer the wide and flexible possibilities required for our applications.

For interfacing the ASP with detectors, it is also intended to evaluate the feasibility of VLSI embedding, taking advantage of the actual state and future development of custom or semi-custom chips (Analogic & digital ASIC).

The MPPC arrays will be used for studying the basic ASP hardware and software concepts. In order to test the ability of the machine to process quickly large amount of data, special hardware interfaces fed with simulated data are developed. In order to test the MPPC-array, a CCD camera appears to be one of the easiest and cheapest input units for large amount of data (100 Mbytes/s) before embedding the ASP in a real test bench detector for high energy physics experiments, for which even larger amount of data will have to be processed (typically 10 Gbytes/s for a LHC calorimeter). Putting together four 16K arrays into one machine will allow to assemble a 65536 array machine, particularly suitable for a demonstration of the on-line image processing power using a CCD camera. Therefore we develop an interface to connect a 1024 by 1024 pixel CCD camera.

1.2.3 ASP evaluation

Apart from the hardware evaluation of the ASP which is done during the construction of the MPPC-arrays, the ASP potentiality in LHC-oriented applications is evaluated through a detailed analysis and modelling of processors embedded in the front-end electronics of characteristic particle detectors (calorimetry and leptons selection). The MPPC-arrays will be used for real time software development of parallel feature extraction algorithms. This will help in determining the best hardware architecture to run event selection algorithms. For fixed target experiments, one of our main partner Saclay studies the use of ASP for the prompt neutral trigger of the approved NA31' experiment and Geneva University will study the power of ASP for fast event selection in an approved experiment for heavy ion physics (WA93), using CCD cameras for imaging luminous chamber planes. ASPs are also used by one of the partners (EPFL) in non H.E.P. applications, for HDTV image sequence compaction and restoration.
1.3 Structure of the project

The collaboration has been established in 1989. A large open discussion on ASP took place in the collaborating institutes, in particular at the EPFL in June 89 [4]. The final MPPC project was approved in April 1990 [5,6]. It formally started at CERN with the allocation of the resources in July 1990.

1.3.1 Collaborating Institutions, Universities and Industries

The Collaboration consists of eight Institutes divided into two groups:

1) The Main Partners are contributing to the development of the hardware and software and to the applications. A 16K processors array machine will be delivered to each Institute at the end of the initial phase of the project:

ASPEX-Microsystems Ltd, United Kingdom, a small company which has invented this massively parallel architecture, represented by
Prof. R.M. Lea, Chairman and Managing Director,
CEA-CEN Saclay, France, represented by
M. Aymar, Chef de la Direction des Sciences de la Matière (DSM),
IN2P3-CNRS/LAL Orsay, France, represented by
P. Lehmann, Directeur de l'IN2P3,
CERN, Geneva, Switzerland, represented by
W. Hoogland, Director of Research,

2) The Associated Partners, who have agreed to participate to the development of the applications only, at least at the initial phase of the project:

Brunel University, United Kingdom represented by
Prof. R.M. Lea,
EPFL - Ecole Polytechnique de Lausanne, Switzerland represented by
Prof. M. Kunt, Laboratoire de Traitement des Signaux, LTS
Thomson - TMS, France represented by
Mrs. M.-N. Gaujour, CCD Marketing Manager,
Université de Genève - Faculté des Sciences, Switzerland represented by
Prof. M. Martin.

The Collaboration is formally hosted by CERN and will remain open to new teams
interested to join the project.

1.3.2 Responsibilities

The sharing of the responsibilities has been decided and is as follows:

**ASPEX in collaboration with Saclay**: construction, operation and maintenance of the four MPPC Arrays using up to 65,536 Associative Processing Elements with the appropriate control system. For this purpose, Saclay has established a strong dedicated hardware platform for the development of the boards to be constructed for MPPC. ASPEX and Saclay, already involved in the construction of TRAX-1, provide to the other partners the necessary documentation on chips, boards, software and simulation tools.

**CERN, Orsay and Saclay**: selection of suitable detectors or detector prototypes, development of associated triggering parallel algorithms ("demonstrators"), for modelling the embedded use of the MPPC array. Construction of the relevant interfacing and demonstration on test benches.

**CERN, University of Geneva, and Thomson-TMS**: construction, operation and maintenance of the CCD camera with associated optics and electronics, readout and interface to the MPPC array. Saclay is interested in the know-how of the CCD electronics and could provide help. Possible implementation of the CCD camera in a demonstrator-experiment is envisaged with assistance of CERN.

**ASPEX**: development of general software tools for using the MPPC array, debugging— including error diagnostic and maintenance. ASPEX also provides upon request the training in ASP technics and programming and the simulation tools.

Two kinds of ASP applications are envisaged by the collaboration:

1) **High Energy Physics:**

**CERN, Geneva, Orsay and Saclay**: application software involving parallel algorithms development and ASP coding. They are intended as triggers in future LHC (and SSC) experiments, but include earlier applications in earlier fixed-target experiments.
2) HDTV:

EPFL, application software involving parallel algorithms development for image sequence representation and coding using ASP.

ASPEX and Brunel will help and support the application teams. Everyone will benefit in this collaboration from the work done for every application. In this view, frequent workshops and meetings are organised for a fruitful exchange of results and ideas.

A training in ASP architecture and programming was scheduled and a two-week course has been organised at the early stage of the project for the scientific staff of the MPPC collaboration. It was held in June 1990 at EPFL and run in collaboration with ASPEX. Fifteen students from the MPPC collaboration and four professors from ASPEX were involved.

1.3.3 Coordination

The project is managed by a board of representatives from the collaborating institutions as listed below:

- ASPEX & Brunel : R.M. Lea
- CEA-CENSaclay : P. Borgeaud, J. C. Brisson
- CERN : R. Bock, F. Rohrbach
- EPFL : M. Kunt
- Geneva : M. Martin
- LAL Orsay : E. Augé, P. Heusse
- Thomson : M.-N. Gaujour

Project Coordinator: F. Rohrbach / CERN

Many collaboration and management meetings have been held [7 to 12].

1.3.4 Milestones

The project, in its initial phase, is scheduled to terminate in the last quarter of 1991. The main milestones and component delivery time of the experimental apparatus are:
<table>
<thead>
<tr>
<th>Milestone</th>
<th>Date</th>
<th>Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ASP course at EPFL</td>
<td>4/90 6/90</td>
<td>EPFL, ASPEX</td>
</tr>
<tr>
<td>2 Start MPPC array prototype design</td>
<td>6/90 7/90</td>
<td>MPPC</td>
</tr>
<tr>
<td>3 Overview of applications modelling</td>
<td>7/90 7/90</td>
<td>MPPC</td>
</tr>
<tr>
<td>4 ASP hybrid prototype module validation</td>
<td>10/90 3/91</td>
<td>ASPEX</td>
</tr>
<tr>
<td>5 Mpixel CCD sensor delivery</td>
<td>10/90 11/90</td>
<td>Thomson-TMS</td>
</tr>
<tr>
<td>6 MPPC array architecture validation</td>
<td>11/90 12/90</td>
<td>Saclay</td>
</tr>
<tr>
<td>7 First results of applications modelling</td>
<td>12/90 12/90</td>
<td>MPPC</td>
</tr>
<tr>
<td>8 Basic software for the prototype test</td>
<td>2/91 3/91</td>
<td>ASPEX</td>
</tr>
<tr>
<td>9 Small scale prototype test</td>
<td>2/91 4/91</td>
<td>Saclay</td>
</tr>
<tr>
<td>10 First design of applications prototypes</td>
<td>7/91</td>
<td>MPPC</td>
</tr>
<tr>
<td>11 Basic software for the MPPC array</td>
<td>10/91</td>
<td>ASPEX</td>
</tr>
<tr>
<td>12 Test of first MPPC array</td>
<td>10/91</td>
<td>Saclay</td>
</tr>
<tr>
<td>13 Delivery of the four MPPC arrays</td>
<td>12/91</td>
<td>Saclay</td>
</tr>
<tr>
<td>14 Full integration of test platform</td>
<td>12/91</td>
<td>MPPC</td>
</tr>
</tbody>
</table>

Minor delays have been incurred in the milestones of the hardware, due to administrative and technical reasons detailed in the ASPEX status report presented below. The main reason in the delays is found in the strong connection existing at the early stage of the project between the starting hardware blocks for the MPPC project (the unpackaged VASP-64 chips which are necessary for the hybrid modules) and the TRAX-1 project (based on the VASP-64 packaged chip). After the successful VASP-64/E1-B1 chip validation in Brunel [13] and the full test end 1990 of the VASP-64/E1-B2 chip: TRAX-1 and MPPC projects are now totally decoupled.

On the other side, the work for the applications has progressed faster than expected and the expertise in ASP simulation has grown tremendously within the Collaboration, at least in the institutes having provided the required manpower on schedule.

1.3.5 Budget and financial control

A total of 1570-KSF, as material cost, has been budgeted for the development and implementation of ASP chips and modules for 4 MPPC arrays of 16K, including chip substrate, assembly, packaging and basic software. The budget is based on a chip price of
£75 (currently quoted by the ES2 foundry) and an estimated £365 for module assembly and packaging (currently quoted by ASPEX) and a rejection factor of 20% on the modules. It also includes a modest contingency on material (7%), the ASP Simulator package for EPFL and ORSAY and the cost of the ASP course for the fifteen students.

It does not include exploitation cost for the teams nor the specific budget for user interfacing. ASP machine related electronics (Crates and control boards for 4 MPPC arrays) are provided in a separate budget by the main partners.

The bulk of the hardware for the construction of the four machines is made by ASPEX and Saclay for the four main partners and a special agreement has been established between these two institutes.

The collaborating Institutions have agreed to fund the project during the fiscal years 1990-1991 according to the following distribution:

<table>
<thead>
<tr>
<th>Institution</th>
<th>Financial contribution (material cost only)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASPEX-Microsystem</td>
<td>650.- KSF</td>
</tr>
<tr>
<td>CEN-Saclay</td>
<td>300.- KSF</td>
</tr>
<tr>
<td>EPFL</td>
<td>60.- KSF</td>
</tr>
<tr>
<td>LAL-Orsay</td>
<td>220.- KSF</td>
</tr>
<tr>
<td>CERN</td>
<td>300.- KSF</td>
</tr>
<tr>
<td>Thomson-TMS</td>
<td>40.-KSF</td>
</tr>
</tbody>
</table>

TOTAL 1570.- KSF

In order to follow the financing of the MPPC project a Finance Commission has been established[14]. It is composed of the following four representatives from the Main Partners Institutes:

For ASPEX Microsystems Ltd.: Mr. John Lancaster
For CERN: Mr. Victor Van Den Berghe
For LAL-Orsay: Mrs. Cécile Caresche
For CEN-Saclay: Mr. Gilbert Burgun
The task of this Commission is to:

1) **Control** the financial development of the MPPC project,
2) **Help** to minimise all kinds of administrative delays in financing the milestones of the project,
3) **Propose** to the MPPC Management board financial solutions in case unexpected difficulties,
4) **Distribute** relevant financial information to the partners.

1.3.6. **Manpower**

The following manpower had been agreed for the Project:

<table>
<thead>
<tr>
<th>Institution</th>
<th>Professional staff</th>
<th>Proposers</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASPEX-Microsystems</td>
<td>4</td>
<td>+</td>
</tr>
<tr>
<td>Brunel University</td>
<td>1</td>
<td>+</td>
</tr>
<tr>
<td>CEN-Saclay</td>
<td>5</td>
<td>+</td>
</tr>
<tr>
<td>EPFL-LTS</td>
<td>2</td>
<td>+</td>
</tr>
<tr>
<td>Geneva University</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAL-Orsay</td>
<td>4</td>
<td>+</td>
</tr>
<tr>
<td>Thomson-TMS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CERN</td>
<td>3</td>
<td>+</td>
</tr>
</tbody>
</table>

At present, the following staff is working on the MPPC project (only part time for most):

- **ASPEX**
  
  John Lancaster, R.M. Lea, D. Boughton, Pete Grimmond, Steve Hedge,
  Tony Higgins, Argy Krikelis,

- **LAL-Orsay**
  
  P. Heusse, E. Augé, R. Ansari, J.-L. Bertrand,
  J.-P. Coulon, A. Ducorps, D. Lavigne

- **EPFL**
  
  T. Reed, Andrea Basso, F. Dufaux,

- **CEA-Saclay**
  
  Pierre Borgeaud, Jean-Claude Brisson, Gilbert Burgun,
  Michel Mur, Jean-Claude Raoul, Bruno Thooris,
1.4 Highlights in 1990

The main issues for 1990 were:

- The MPPC project has been approved by all the Institutes.

- The required and necessary funding for 1990 has been obtained from all the institutes.

- Much of the requested manpower has been assigned, some are working on the project since a short time only.

- The ASP course in Lausanne at EPFL. All participants learnt ASP technology and programming using a VASP-simulator installed on SUN-4 machines.

- The first ASP chips, the VASP-64/E1-B2, have been delivered and validated by MPPC.

- The ASP boards are in the stage of detailed design.

- The MPPC machine architecture is defined.
- The LAC low level controller is in the stage of prototype construction at Saclay.

- The Hybrid modules prototypes incorporating 1024 APEs have been ordered.

- The CCD read out and digitisation for 1 Megapixel CCD sensor has been designed and a prototype constructed.

- **Seven different kind of applications**, where ASPs are used, have been identified and are under detailed study within the Collaboration:

  **Three LHC oriented:**
  - fast algorithms development for feature extraction from calorimeter data at Orsay
  - fast second-level muon selection for a muon compact detector model at CERN,
  - algorithms for low level feature extraction of the TRD at CERN,

  **One SSC oriented:**
  - SDC calorimetry at Saclay, fast selection on isolated electrons, jets and missing \( E_t \),

  **Two for fixed target physics:**
  - Level-2 triggering on clusters at Saclay for NA31',
  - Image processing at Geneva for track reconstruction in WA93.

  **One for HDTV:**
  - Image compaction and restoration at EPFL.

For most of these applications, simulations using the VASP-Simulator have shown that the objectives could be met using ASP. Results of these simulations are presented in Part III of this report. They are very encouraging. The expected breakthrough in on-line applications looks to be a real possibility, at least as found on the basis of ASP simulation. No part of the collaboration has seen a reason why ASP could not be used as a very efficient tool for fast selection of events in future high luminosity experiments.

As a consequence, and last but not least, all the teams are confident to achieve the goals of this project.
1.5 Prospect

The future steps of the programme have been clearly identified within the Collaboration. There is no change in the proposed programme, but it is essential to focus the efforts on the hardware. The expected success of the test of the first ASP prototype, scheduled early 1991, is considered by the whole Collaboration as the major milestone and as the real start for future application proposals. For that purpose, we need full support from the collaborating institutes.

Then, depending on personal orientation, it will be the time at least for some partners to present proposals together with experimental physics teams in order to set up new Collaborations for 1992 and later with the objectives of realising a full test of the ASP in real time and in experimental conditions. This is the final goal of the evaluation phase of the MPPC project before massive integration of ASP in physics experiments or for any industrial application.

The Collaboration remains open to new teams interested to join the project. We concentrate our efforts on the ASP but we already have seen that it is very likely that the final architecture for solving a given experimental problem will have to combine various kinds of processors and technics. This is why we should remain open to all ideas and advice from experts.
- Part II -

Progress report of the hardware
PART II  Progress report of the hardware

2.1  ASPEX – MPPC progress report from Aspex Microsystems Limited.

2.1.1  The chip

The basic chip for the construction of the compact hybrid MPPC modules (HASP) consists of a programmable VLSI SIMD parallel processing device, incorporating 64 associative processing elements (APE, see fig.1 and 2): the Aspex Microsystems Limited VASP-64 VLSI ASP chips (see fig. 3). The development of these devices is an independent project within Aspex. The validation of this chip has been considered by the Collaboration as the starting Hardware Milestone of the Project.

In August 1990 a formal acceptance test of the VASP-64/E1 B1 step VLSI ASP chip for use in the HASP prototypes modules was held at Aspex. The B1 step device was shown to be almost a completely unqualified success and is usable in its current form subject to two algorithmic restrictions. These restrictions are required to overcome a slower than expected Word Read operation and a pattern sensitivity during refresh which lowered APE count. Consequently it was confirmed, at the MPPC September Management Meeting in Brunel, that the device was suitable for making the first prototype hybrid HASP module which is under development at ASPEX for the MPPC-Arrays.

However, it was recommended that the two restrictions be alleviated by amending the design database prior to commissioning the required batch of VASP devices for the HASP. This approach was accepted and it was agreed that 100 B2 step VASP chips should be ordered. Unfortunately, due to administrative and procedural problems at CERN it was not possible to order the chips until early October. They are expected January 2nd 1991. This later than expected delivery of the VASP chips will of course delay the delivery of the HAS prototype.
Fig. 1 – Schematic of the associative processing element (APE)

Fig. 2 – The Associative String Processor
Fig. 3 – The VASP-64/E1 chip manufactured by ES2 for MPP

**Schematic**

**Power**
- Vcc
- GND
- Clock
- LLI
- LLO

**Left Link Port**
- SPI
- SPPOE
- SPPLD

**Test Bus**
- SPCKA
- SPNCKA
- SPCKB
- SPNCKB
- SPO

**Chip Select**
- NCS

**Status Output**
- MRO
- MRT
- LRO
- LRI

**Right Link Port**
- Activity Bus A00…A11
- Control Bus C00…C31
- Data Bus D00…D31

**Chain Connection**

To the VASP Chain Controller
For second-level triggering experiments a faster device will be required. Consideration is currently be given as to how this requirement can best be achieved in a timely manner. The current options are to speed up the current VASP/E2 internal Aspex development programme or to use a 1.2um or 0.8um SOS (silicon on sapphire) device which should become available through a second source deal currently in negotiation. Both routes are being actively pursued in order demonstrate a fully working 25ns slot VASP chip in the Summer of 1991.

2.1.2 The hybrid module

The hybrid ASP module is being design by Aspex Microsystems, in consultation with Saclay and is to be manufactured by the sub-contractor PolyCon Inc. The module has been assigned the interim designator HASP/P1. Work to date has concentrated on the design of a prototype HASP module.

In September a preliminary design for a HASP with 1024 APEs and 4 I/O channels was completed and exposed to the collaboration and PolyCon for comment. During the September Project Management Meeting feedback from PolyCon and Saclay was presented. PolyCon questioned the realisability of the design and indicated that an expensive custom package would be required. The feedback from Saclay's ASP board work indicated that real estate for a 4 channel HASP was not available on the circuit card.

In the absences of a decision on packaging from the September 26 & 27 Project Management Meeting the HASP design was revised during October. The revised design (see fig. 4) has bypass of 64 & 256 APE blocks and 2 I/O ports which can be configured as 2 x 512 APE substrings or 1 substring with a LAC interface and an ADB (ASP data buffer) interface. This design was discussed with Saclay and then issued to PolyCon. The revised design is targeted to a standard 184 pin package 2.5" x 2.5" (3" x 3" with leads).

In addition to the VASP chips the HASP contains glue buffer and PAL (programmable array logic) devices. The design of the PALs was completed in September. A supplier who can provide programmed parts has been identified and the parts will be ordered soon. It has been suggested by PolyCon that more of the APE boards external logic could be included in the HASP, in particular the RAMs. This is currently under investigation.
Fig. 4 – The HASP, Hybrid ASP module organisation
A thermal analysis of the HASP design has been undertaken. The results show that a reasonable airflow (200 linear feet per minute) across the surface of the HASP will hold the device junction temperature below 70 degrees C.

In anticipation of the arrival of HASP devices from PolyCon, work has started on the design of a test jig for the HASP and on the production of test vectors.

2.1.3 The operating system

Work on the operating system and programming tools has now reached the point where further progress cannot be made until the LAC card is delivered. This low level ASP controller is designed by ASPEX and is under construction at Saclay (see 2.2.2), in strong collaboration with ASPEX. It is expected to be ready early 1991.
2. 2  **SACLAY**

2. 2. 1  Machine architecture

*J.-C. Brisson, J.-C. Raoul*

The figure 5 shows the well known general architecture for a microprogrammed processor: SISD (Single Instruction apply to Single Data).

![Diagram: Structure of a microprogrammed processor SISD]

**Fig. 5 – Structure of a microprogrammed processor SISD**

To increase the processing power it is possible to design parallel architecture by using many processing devices (Arithmetic and Logic Unit or ALU) working in parallel on an array of data: SIMD (Single Instruction apply to Multiple Data). In a multi ALU machine two types of data can be considered, scalar and vector. Each ALU works simultaneously on a different vector data, scalar data is broadcast over all ALU.

Some limitation slows down the speed of such machine. The first bottle neck is the competition to get simultaneously vector data in main memory, the second problem is to disable parts of ALU depending of previous result during the processing.
**C. P. U.**

- Microcode memory
- Sequencer

**MEMORY**

- Instruction memory
- Scalar data memory
- Vector

**ALU**

- Vector
- Vector
- Vector
- Vector

**ALU**

- Vector
- Vector
- Vector

**ALU**

- Vector

**ALU**

- Vector

---

*Fig. 6 – Structure of a vectorised processor SIMD*
ASP chip can be used to build SIMD machine in which the ASP string is used as multi ALU. ASP structure encompasses some difficulty of SIMD machine: vector data are distributed and kept in each ALU (APE) in a 64 bits local memory and access of each APE is an associative process which allows to address many APE at a time or works with a subset of the string (selected APE).

The first machine built with ASP chip will be TRAX-1 for NA35 collaboration. Designed by ASPEX this machine is implemented by a ASPEX-SACLAY collaboration and the know-how from this work will be very important for the MPPC collaboration. In addition, the control board and many sub-elements of this design will be used for the MPPC machine.

2.2.1.1 TRAX-1 machine

TRAX-1 is a SIMD machine design for off-line analysis of very complex streamer chamber pictures.

Fig. 7- TRAX-1 Architecture
This machine is built in a look-like VME crate using triple-high eurocard board. The architecture of TRAX-1 machine is given in Fig. 7. A SUN-3 is used as human interface for the machine. It is called HAC (High level ASP Controller) and drives the IAC (Intermediate level ASP Controller) made by a commercial 68030 VME CPU board. It is used to drive the ASP machine by 3 different access VME and VSB bus and a proprietary fast DMA channel. The ASP machine is made by one Low level ASP Control board (LAC) and 8 ASP array boards. Each ASP board contains 32 ASP chips that means 2048 APE per board producing a machine of 16K processors. Microcode memory stores optimised algorithm procedure to be executed by the ASP string.

A CCD camera is used to record and digitise streamer chamber pictures; data are fed from the camera to the array data buffer by VME or VSB bus under control of the Secondary Data Exchanger (SDX). Because the unique path (VME or VSB) boards are loaded in sequence. In each board the Primary Data Exchanger (PDX) will load data in each APE of the string.

It can be noted that a parallel access to each board can be done by a direct connection to the VSB connector on the backplane through a small adaptor card (bus driver and receiver). In this case each connected instrument must act as a master to generate the VSB protocol.

Inter APE network interconnect the LAC and the ASP boards in a double loop (right and left direction). Its purpose is a direct communication between APE; this network works in two modes: shift the contain of APE string over the next string for comparison or remote activation of a selected APE.

2.2.1.2 MPPC machine

The purpose of MPPC collaboration is to design and build a real time machine connected on line on detector prototypes. We planed to build 4 machines, one for each main partner of the collaboration. Each machine will be composed by one LAC and two ASP boards giving a 16 K machine. In concentrating all ASP boards in the same machine we will have the possibility to evaluate the power of a 65 536 processors machine.

The architecture of the MPPC machine is given in figure 8. The LAC board should
be roughly the same as for TRAX-1. For the ASP board we will use the hybrid module described previously (see figure 4) which contains 16 dies of VASP64 circuit. Each ASP board will contain 8 modules giving a number of 8192 APE by board. Each module will have its own PDX and ADB to allow a faster feed for data; 8 connectors will give the possibility to connect external data acquisition logic. ADB memory is a double port memory large enough to store more than one event so the memory can be fed with the next event during the process of the previous one.

---

Fig. 8 – MPPC Architecture
2.2.2 The LAC controller

The Low level ASP Controller card (LAC) provides the environment to execute ASP application programme on one or more associate ASP boards. The LAC is controlled by higher level ASP controllers (see before) over its VME and VSB interfaces and DMA Peripheral Bus (DPB) which connect directly the IAC CPU card to the LAC.

The LAC block-diagram is shown in figure 9. The major parts are a CPU formed by a high speed sequencer and a microprogramme memory to store the LAC operating system and the low level procedures used in application programmes. This Micro Instruction Buffer (MIB) is a 152 bits wide memory 64k words deep at the maximum.

Fig. 9 LAC Architecture
Scalar data can be shift left or right by the Scalar Shift Register (SSR) and can be stored in a high speed scratch pad memory and in two FIFOs, Search/Write FIFO and Read FIFO. The Low level Procedure Control Queue (LPCQ) receives commands from the IAC to be executed by the ASP machine. Data Conversion Register (DCR) provides conversion to fit the ASP data representation (Bits, Bytes or Word). AG bus Data Assembler (AGbus DA) assembles data words to be broadcasted over AG bus to ASP board. AG bus Control Assembler (AGbus CA) and AG bus Activity Assembler (AGbus AA) assemble control word and activity bits to be broadcasted also to ASP cards. The Global Status Inductor (GSI) interface the AGbus to global status indicator. The AGbus Link Controller provide control and monitoring of the inter APE network.

Many of these features may be accessed by the Host; the Host bus connects the host interface to the major parts of the card. Other internal buses interconnect LAC elements. TCbus (Test Condition bus) connects test condition sources to the branch logic of the sequencer; AMbus (Address Modifier bus) connects address modifier sources to the address generation logic of the sequencer. The Scalar Data bus (SDbus) is formed by two 32 bits wide data bus, the read (SDbus.rd) and the search/write (SDbus.sw) buses. LACDbus, AGDbus and CTRLbus carry micro-order fields from the microprogramme store to functional blocks of the LAC and through them to the AGbus. The Scan Path bus connect all register of the board not accessible from the Host bus into a serial bus. Performances monitor and scan path interface allow monitoring and debugging of the machine.

The LAC card layout is shown on figure 10. LAC card is a multilayer board with 6 layers of signal and two layer for ground and Vcc planes. The micro code memory is made in two banks and using 8K or 32K pin compatible memories, it is possible to obtain 8K, 16K, 32K or 64K size. Most of the glue logic is done with Xilinx PGA.
Fig. 10 – LAC card layout
2.2.3 The ASP boards

2.2.3.1 Overview

The ASP board contains ASP chips and there associate Array Data Buffers (ADB). The communication between the LAC and the ASP array is done by the ASP Global bus (AGbus) on the P3 connector of the extended VME card.

Scalar data are transmitted on 32 lines in the form of a 32 bits word or in ternary mode in case of bytes or bits. Activity bits use 12 lines and are always transmitted in ternary mode.

Instructions from the LAC are transferred over the AGbus in a compressed form (20 bits). Two set of look-up memories are used to expand the AGbus instructions in two 16 bits subinstructions send to ASP chips and a 32 bits pattern distributed over the card control bus. AGbus carry also synchronisation signals and status (match for example).

Four daisy chains are used between boards in the AGbus backplane to implement the inter APE network.

2.2.3.2 TRAX-1 ASP card

The TRAX-1 ASP card block-diagram is shown in figure 11. The main part is the ASP Array (ASPA) comprising an array of four ASP channel with 8 VASP-64 chips per channel to give a total of 2048 APE.

The ADB comprises two dual-ported memory planes, providing one output and one input data channel for the ASP array. The memory is divided in two pages, A and B, which allow independent access to process data or to feed new data simultaneously.

ASPA communicates with ADB memory through Global Data bus (GDbus) formed by two 32 bits wide bus, Global Data Input bus (GDIbus) and Global Data Output bus (GDObus) Primary Data eXchange Address Logic (PDX A L) generate ADB address on the PDXbus during sequential 32 bits transfer between ASP array and ADB. The LAC microprogramme may access ADB planes of a selected page for a PDX transfer
independent of an SDX operation.

The host through VME or VSB interface may access the ADB plane of the other page simultaneously for a Secondary Data exchange (SDX) either in random access or in sequential access. SDX Address Logic (SDX A L) is responsible for generating ADB address during sequential transfer. IDbus is an internal data path for VME or VSB interface, LIDbus is the internal data bus for ASPA and ADB. Between this two buses, the ASPA Data Interface (ADI) is used to align data in single byte, word (double byte) or long word (quad byte) for the host access to the ADB and to perform conversion for ASP tertiary mode.

The VASP Debugger (VDB) is a means to the host to monitor the internal state of a selected VASP-64 chip. The content of the selected chip is stored in a double port memory which can be subsequently read by the host.
Fig. 11 – ASP card Architecture for TRAX-1
2.2.3.3 MPPC ASP card

Many elements of the TRAX-1 ASP card will be used in the MPPC ASP card. A draft for the MPPC ASP card block-diagram is shown in fig. 12.

The MPPC ASP array board contains 8 modules of 1024 APE which may work as a string of 8192 APE. One buffer memory (ADB) of 2 k words is associate with each module to speed up the transfer of data between front end detector electronics and APE. Generally this memory will be used only to store new data to be processed. Type of the ADB memory is not yet decided; ADB may be either a FIFO memory or a double port memory. In both case data from front end electronics can be written in the memory simultaneously to an access from the ASPA. SDX A L generate synchronisation signal to external devices. Result are stored in the Output Data Buffer (ODB) to be read out by the data acquisition host through the VME interface.

![ASP card architecture for MPPC](image)

Fig. 12 – ASP card architecture for MPPC
2.3 ORSAY - Progress report on the Hardware

E. Augé, J.-L. Bertrand, A. Ducorps, B. Lavigne, Ph. Heusse

2.3.1 Interfacing the ASP machine

2.3.1.1 Introduction

The aim of our group is to check with the MPPC machine whether an ASP-based processor might fit as a second level trigger local processor for calorimeter data at LHC.

The organisation of the trigger logic of an LHC experiment is not yet precisely known. A "first level trigger" device will have to deal with new data every beam crossing (15 ns). This device will have to be fast enough (< 2 µs). It will use simple criteria, and will probably not be programmable. The full calorimeter resolution might also not be available at that level.

It is therefore of interest to think about processors able to run fast algorithms (< 100 µs) on calorimeter data in order to refine the event selection before sending the data to a general purpose processor for the ultimate selections. The architecture of this "second level trigger" must be such that data movements are minimum. One therefore has to share the work between "local processors" which treat a calorimeter "patch" (i.e. 100-2000 cells) each and send only a few relevant words, and a "global processor" that takes the decision.

A local processor will consist in several "elementary processors" (EP) working in parallel, such that at a given time, one calorimeter cell is associated to one EP. If this EP would be fast enough, it would deal with several calorimeter cells successively for the same event; otherwise, the total number of EPs must be equal to the number of calorimeters cells ( ≈ 10^5). The links between all these EPs must be very powerful to perform the appropriate algorithms. This is the reason of our interest in "massively parallel processors".

2.3.1.2 The Interface board

In Orsay, we are presently designing a test bench based on a MPPC machine, in
order to gain experience on this device, and study in the most detailed way whether ASP arrays might fit as second level local processors. These studies must not only consist in running algorithms, but also in loading the data into the processors, and extracting the relevant results at rates compatible with what we expect from LHC. The "Interface Board" will allow us to repeat the typical sequence:

- Transfer from Interface Memory into MPPC machine buffers data corresponding to "event" number N + 1,

- Transfer from MPPC machine buffers into MPPC machine processors data corresponding to "event" number N,

- Transfer from MPPC machine buffers into Interface Memory results corresponding to "event" N – 1. These three data transfers will be performed simultaneously, at maximum rate = 10Gbit/s (16 or 32 channels in parallel x 20 MHz x 16 bits per word)

- Run algorithm on "event" number N,

- Transfer from processors into MPPC machine buffers results for "event" number N.

The Interface Board will be accessed from a Host computer in order to check the results and load data corresponding to new "events".

Since Orsay is not involved in the hardware of the MPPC machine itself, work dependent on this hardware will only start when the machine specifications are available. However, we manifest special interest in the machine I/O.

2.3.2 Minicontroller

From the test bench, we will get experience on I/O and on "low level control" of the elementary processors. In the future, we hope to be able to propose modifications to simplify the existing "general purpose" ASP controller, the LAC, in order to get a device optimised for an application at LHC. The high level control of the machine, and the operating system would most probably have to be modified accordingly.
2.4 CERN CCD acquisition system

J. Feyt

2.4.1 Objectives

For several applications in high-energy physics, a CCD detector associated with an image intensifier is the basic instrument for recording light imaging detectors. A 1K x 1K CCD with 4 output channels and readout speed of 100ns/pixel produces data flow around 40 Mbytes/s.

For the MPPC project CERN is responsible of the construction of the image acquisition system using a CCD. The objective is the realisation of the CCD readout, using direct digitisation pixel by pixel, and of the interface to the MPPC machine (4 times 16K APEs) see fig.1. The CCD readout (fig. 2) consists of:

A CCD head with:
- 1K x 1K CCD 4 outputs 10Mhz output rate,
- CCD clocking and control,
- Double correlated sampling (DCS),
- Output amplifiers,
- Line drivers of the analog signal to the control card,
- Lines drivers for control signals.

A control card based on a VME(VSB) standard with:

- 10 bit digitisers (one per output of the CCD),
- A one Mbyte bank for the black image storage,
- Hardware substractors and rounder for a full significant 8bits data (dark image correction),
- At least one bank of one Mbyte for local buffering,
- A first simple hardware data reduction,
- Logic interface with the bus,
- Possibility of high speed visualisation for focussing, debugging etc....

Some additional components are necessary in running experiments:

- Cable length up to 20 m between the head and the VME crate.
- Necessity of two read-out versions:
  - full image acquisition,
2.4.2 Development

Several steps of the construction programme have been achieved (see fig.3):

a) Realisation of a first read-out prototype version using a CCD (TH7883/63) in order to gain expertise with CCD clocking, CCD output signal processing, etc...

Two versions of the first prototype are now running at 6 and 8 Mhz. A second generation of the head using higher speed clock buffers, switches and current loop amplifiers is under way and will permit to run these cards at 10 MHz with a efficient dual correlated sampling which bears out the use of a 10bit digitiser (figures 4 and 5).

b) Realisation of a second version using a THX31159 CCD (512 x 512; 2 outputs) using SMD (surface mounting device). This step is needed in order to build and test a prototype for the multi-output system of the 1K x 1K CCD; this prototype can be used in applications with reduced resolution. As the 1K x 1K version cannot be implemented with the use of standard DIL (dual in-line) packages, the use of SMDs package allows to design a PCB (printed circuit board) quite close to the final version. The PCB for the 512 x 512 (THX31159) is actually implemented on PCAD and should be on test at the end of January 1991.

All these versions are fitted out with a 8 bit A/D converter and work with 8 or 16 bits on the VME bus to be compatible with our development and display system (see fig. 6MICRON-MacVee interconnection between a VME crate and a Macintosh using MacSYS developed by UA1).

The heads have been built with discrete components (ECL, TTL, MOS, high speed analog amplifiers and switches). In order to increase the reliability, to decrease the noise and size, a development based on ASIC is viewed for the next realisation.
2.4.3 Next steps

Construction of the final read-out version with the duplication of the acquisition channels, the multiplication by 4 of the memory size and the addition of the complete I/O interface: 32 bits VME bus, VSB bus (figure 1).

The implementation of the 10 bit conversion and the associated hardware and software for individual pixel correction will be soon implemented. Lack of manpower (mainly software development) and money saving consideration (a 10 bit 40 ns converter costs about 1500.- SFr compared to 150.- SFr for 8 bit) have delayed this step until now.

2.4.4 Collaboration with Thomson-TMS

Apart from the current Thomson-TMS products and developments like special UV coating, thinned CCD, EBCCD tube, Short Wave Infrared (SWI) detector, 2048 x 2048 CCD etc... we want to use in the immediate future CCD package with Peltier cell which cools down the sensor by about seventy degrees (see curve of the dark current versus temperature on figure 7).

In collaboration with our Thomson-TMS partner, we envisage the realisation of the circuit for the CCD head by using the bipolar semicustom arrays Thomson-TMS facilities (TSFLxx) with POLY TOOL development system. The development of faster readout has been asked to Thomson: parallel readout line by line with interlaced memory, which is a request for satellite applications, could be of great interest for fast imaging detectors.

THOMSON is also working on a hybrid DCS using the same components (but not encapsulated) than those which are used for our present development (fig. 8 and 9).

2.4.5 Hardware data reduction

In high energy physics experiments, usually a small percentage of data is significant for the event analysis. As the time for file manipulation is very critical (i.e. RICH detector), compression of data must be used. A usual way to decrease the size of data files is to store only useful pixels (position and content). The size of the file in the case of a
512 x 512 or 1024 x 1024 CCD (18 respectively 20 bits for X and Y) is 4Nn where N is the total number of pixels and n the number of useful pixels in %. The number of useful pixels must be significantly less than 25%.

To increase the data reduction a card using another system which we will call "byte page data reduction", is under construction:

The first pixel of each page of 256 is recorded in two bytes, H00 in the first, the digitised value of the pixel content (8 bits) in the second. Between two records of the beginning of each page, only the useful pixels are recorded with their address in the page and their value. The size of the resulting file is:

$$2N/256 + 2Nn$$

The result of these data reductions are shown in figure 10.
**64 K MPPC machine**

Final configuration 4 outputs, 1024 x 1024 CCD data acquisition system

Fig. 1 – Block diagram of the CCD to MPPC machine system (Objective).
Fig. 2 – Block diagram for read-out of the 4 outputs CCD THOMSON-TMS, TH7896 A.
Fig. 3 – Block diagram of the CCD to MPPC machine system (Present status).
CCD acquisition system (TH 7863/7883)

Fig. 4 – Block diagram of the CCD read-out with no data reduction
CCD acquisition system (TH 7863/7883)

Fig. 5 – Block diagram of the CCD read-out with data reduction
Fig. 6 - VME crate configuration at CERN for CCD interface development platform.
Fig. 7 – Dark current versus CCD temperature (From THOMSON - TMS / St Egrève)
SPEED LIMITATIONS

DIFFICULT TO GO FASTER THAN 120 ns/Pixel:

CLOCK BUFFERS: 5 MHz
SWITCHES & AMPLIFIERS for D.C.S.

LIMITATION DUE TO THE CCD ITSELF?

A TH7883/63 is given for 15MHz output frequency but above 8 MHz it is difficult to see the floating phase and above the use of D.C.S. presents no interest.

A realistic evaluation of this parameter needs first a clean clocking of the CCD.

NO ACTUAL LIMITATION ON THE DIGITAL PART:

<table>
<thead>
<tr>
<th>ADC</th>
<th>40 ns</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMORY</td>
<td>50 ns</td>
</tr>
</tbody>
</table>

Fig. 8 - CCD read-out speed considerations.
# SPEED UPGRADING

<table>
<thead>
<tr>
<th>ACTUAL</th>
<th>NEXT</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Clock buffers</strong></td>
<td></td>
</tr>
<tr>
<td>DS0026 (NS) 5 MHz 1 A</td>
<td>DS469A (SILICONIX) 25 MHz 1 A (10 SF)</td>
</tr>
<tr>
<td><strong>Track/Sample &amp;Hold</strong></td>
<td></td>
</tr>
<tr>
<td>HI201HS (HARRIS) 30 ns turnon 40 ns turnoff 30 Ohm</td>
<td>BS12 (FET PHILLIPS) 1 ns turnon ns turnoff 30 Ohm</td>
</tr>
<tr>
<td><strong>Amplifier</strong></td>
<td></td>
</tr>
<tr>
<td>HA2625 (HARRIS) 100 MHz 200 V/μS 100 MΩhm input</td>
<td>Current Feedback Op CLC221 (Comlinear Corporation) 170 MHz 6500 V/μS New device stable with low gain and high bandwidth, the drawback is the input impedance : 250 KΩhm</td>
</tr>
</tbody>
</table>

It is very difficult to use commercial devices (size and price):

<table>
<thead>
<tr>
<th>CLC940 (Comlinear Corporation)</th>
<th>HTS0025 (Analog Device )</th>
</tr>
</thead>
<tbody>
<tr>
<td>22 ns 24 pins dual in line</td>
<td>50 ns 24 pins dual in line</td>
</tr>
<tr>
<td>500SF</td>
<td>750SF</td>
</tr>
</tbody>
</table>

*Fig. 9 - High speed component characteristics*
BYTE PAGE DATA REDUCTION

Fig. 10 – Comparison of data reduction format
2.5 THOMSON COMPOSANTS MILITAIRES ET SPATIAUX

CCD development of Area Array CCD Image Sensors

THOMSON COMPOSANTS MILITAIRES ET SPATIAUX area array CCD image sensors employ n-MOS buried-channel technology for low noise and maximum transfer efficiency. They feature compactness, extremely long life-time, low power drain and good sensitivity, as well as excellent environmental characteristics. In operation, they are free from distortion and burn-in due to overillumination. And as the photosites use a MOS structure, these sensors are also free of lag.

The area array product range is made of two main families: TV mode image sensors and full frame image sensors; around these families TMS has developed complementary technologies for CCD and is able to support the surrounding electronic thanks to its ASIC product range.

2.5.1 TV mode image sensor

They comprise an image zone for integration and an adjacent memory zone for charge storage, and are designed in frame transfer organisation. TH7852, TH7863, TH7864 are CCIR standard. TH7866 is EIA-RS170 standard.

Three other products are in development:

1) THX7868A (ex. THX31162) is a CCIR TV format with 768 useful pixels per line.

This device includes:
- anti blooming device allowing exposure time reduction and fast clear operation,
- windowing feature mode where one part of the image can be thrown out in a drain to get more quickly the useful information,
- two output amplifiers allowing mirror function.

2) THX31163 - THX31167:

These two devices have been designed for HDTV. They have 2 x 576 lines of 1260 useful pixels and are in the 16/9 image format and run up to 25 images/second (50 interlaced frames per second).
Their organisation is frame transfer with two high data rate output registers, one for odd columns and the other for even columns. They include anti blooming device. THX31163 is the front side illumination version, THX31167 is the backside illumination version thinned down to 10 μm, in order to increase the quantum efficiency in the blue region of the visible spectrum.

2.5.2 Full frame image sensors

These area array sensors have been designed without memory zone. Their square pixels have 100% aperture. The use of a specific output amplifier results in a very low noise level (4 electrons of noise at -40° C for 50 KHz pixel frequency).

TH7896A (1024 x 1024) and TH7895A (512 x 512) are available in high speed version and in high sensitivity version (see the organisation in figs. 1 and 2).

The TH7895A is now available in MPP mode allowing to divide by 30 the dark signal.

TMS extends its product range with new products in development:

1) THX7885A designed for spectroscopy application:
   It is a full frame CCD with 128 lines including 1024 pixels 19 x 19 μm size, one readout register allowing addition of 3 lines and two low noise output registers. It will work in MPP mode (see the organisation in fig. 3).

2) THX7897A:
   THX7897A is a 2048 x 2048 pixels with the same type of organisation than the TH7896 (1024 x 1024). The pixel size is 15 x 15 μm. It will be designed for MPP operation, with four low noise output amplifiers.

2.5.3 Special technologies

In order to support the military and scientific applications, TMS has developed some complementary technologies around the CCD.
1) Fiber optic coupling:

The area array image sensors are also available with a fiber optic window. This window is made of low absorption elementary fiber (6 µm in diameter) separated by EMA cement to limit cross talk.

2) Special UV coating:

The special coating is a proprietary phosphor developed in order to improve the sensitivity of the CCD in blue and ultra violet wavelengths, down to 120 nm (fig. 3).

3) Thinned CCD:

TMS has developed a new process consisting in thinning the silicon wafer down to 10 µm allowing back side illumination. This technique drastically increases quantum efficiency in blue and UV spectrum and allows sensitivity of soft X-ray and electrons. This process has been applied on the TH7895 (512 x 512) and on the HDTV sensor (see the spectral response of the thinned TH7882 (test vehicle) in fig. 4).

4) Short Wave InfraRed (SWIR):

SWIR detectors made of InGaAs photodiodes sensitive up to 1.7 µm and multiplexed by CCD readout register have been developed at TMS for the SPOT4 imaging satellite.

2.5.4 ASICS

TMS supports a wide range of ASIC products: CMOS semiconductor with gate arrays or combined cells for digital function integration, for example: sequencer for a CCD.

1) Bipolar semicustom with high frequency low noise linear arrays, for example: CCD output signal processing including double correlated sampling, wide band amplification, black level compensation.

2) BICMOS semicustom mixing analog and digital cells, for example to integrate on the same chip video processing sample and hold, A/D converters and sequencers.

3) Full custom technology for high voltage, high drive requirements, for example CCD driver (see typical application kit around CCD in fig. 5).
TH7896 A (THX 31156)

1024 x 1024 CCD IMAGE SENSOR
4 PHASES FRAME TRANSFER ORGANIZATION

LINE ORGANIZATION

8 PRESCAN PIXELS
8 DARK REFERENCES
2 ISOLATION PIXELS
1024 USEFUL PIXELS
2 ISOLATION PIXELS
8 DARK REFERENCES
8 POSTSCAN PIXELS

FOUR OUTPUTS

MINIMUM READOUT TIME
1 OUTPUT  \(\rightarrow\) 112 ms
2 OUTPUT  \(\rightarrow\) 56 ms
4 OUTPUT  \(\rightarrow\) 28 ms

Fig. 1 - Full frame CCD, four outputs
TH 7895 A (THX 31159 A)

512 x 512 CCD AREA IMAGER ORGANIZATION

- 2 LOW NOISE OUTPUTS: 10 MHz
- 2 HIGHER SPEED OUTPUTS: 15 MHz

READOUT TIME

- 1 OUTPUT (15 MHz)  →  20 ms
- 2 OUTPUTS (15 MHz) →  10 ms

SAME PERFORMANCE AS THX 31156

Fig. 2 – Full frame CCD, two outputs
Fig. 3 – Photo MOS matrix with MPP integration mode
Fig. 4 – Thinned CCD – back side illumination
Fig. 5 – Typical application kit
-- Part III --

Progress report of the applications
3.1 LHC/SSC applications

3.1.1 CERN - Muon selection at LHC

F. Rohrbach, G. Vesztergombi, G. Odor.

3.1.1.1 Introduction

Our current research activity covers the survey of different ASP algorithms for fast muon selection at LHC experiments.

The iconic algorithm is an efficient way to handle on-line the huge information coming out of a LHC muon detector [15]. It uses bit representation of the image of tracks and the image processing can be done bit parallel in the ASP machine. The task to be solved is the fast simultaneous recognition of all muon tracks coming out in the magnetic field of the detector. By recognition we mean triggering: the decision is taken by the on-line analysis of the curvature of the track, for momentum determination, signing all the tracks which are above a given cutoff momentum.

3.1.1.2 Track-code simulation

In the central plane of projection perpendicular to the beam axis, muon trajectories are detected at five radii (see fig. 1). The five consecutive values of the azimuthal angle $\phi$ relative to the innermost $\phi$ value, expressed in unit of $\Delta\phi$ (detecting angle granularity, typically a few mrad) is used for calculating a "track-code", as defined later. A track-code is uniquely representative of a charged particle trajectory through the muon detector.

The aim of this study was to construct track codes from the Geant Monte Carlo data and to explore the feasibility and efficiency of triggering by the proposed Iconic algorithm using ASP [15].

The event simulation data (from the LHC compact muon group, K. Eggert, M. Della Negra [16]) are based on the following detector simulation conditions (fig. 1):
<table>
<thead>
<tr>
<th>layer</th>
<th>R(cm)</th>
<th>B(T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>tracker</td>
<td>0-130</td>
<td>4</td>
</tr>
<tr>
<td>calorimeter</td>
<td>130-180</td>
<td>4</td>
</tr>
<tr>
<td>chamber</td>
<td>180-190</td>
<td>6</td>
</tr>
<tr>
<td>fe-absorber</td>
<td>190-280</td>
<td>6</td>
</tr>
<tr>
<td>chamber</td>
<td>280-290</td>
<td>6</td>
</tr>
<tr>
<td>sc-coil</td>
<td>290-340</td>
<td>6</td>
</tr>
<tr>
<td>chamber</td>
<td>340-350</td>
<td>-6</td>
</tr>
<tr>
<td>fe-absorber</td>
<td>350-556</td>
<td>-6</td>
</tr>
<tr>
<td>chamber</td>
<td>556-566</td>
<td>0</td>
</tr>
<tr>
<td>air</td>
<td>566-576</td>
<td>0</td>
</tr>
<tr>
<td>chamber</td>
<td>576-586</td>
<td>0</td>
</tr>
</tbody>
</table>

**Fig. 1** – Muon tracking simulation modelling
Fig. 2 a – $\phi$ distribution at 20 GeV

Fig. 2 b – $\phi$ distribution at 500 GeV
The sample of muon data obtained consists of 100 tracks at 20 Gev and 100 tracks at 500 Gev. They have been used to study the \( \phi(p) \) distribution function.

Generally the mean value of \( \phi \) on a given detector plane is a non-linear function of the muon momentum \( p \):

\[
\phi = \phi(p_0) \cdot \frac{p_0}{p} + \text{a slowly changing term with } p
\]

The 20 GeV and 500 GeV \( \phi \) distributions for each plane \( j = 0, 1, \ldots, 4 \) can be seen on Fig. 2a and 2b respectively. The variance of the distribution is theoretically inversely proportional to the energy. This behaviour is well reproduced by the simulated data. As one can see on fig 3, the mean value can be fairly well approximated with the \( 1/p \) term. For example on the last detector plane:

\[
\frac{\phi(20\text{GeV})}{\phi(500\text{GeV})} = 24.959
\]

Figure 3. Angle \( \phi \) versus the radius \( r \) for two momenta
This enables one to construct tracks with arbitrary \( p \) by simple scaling the 20GeV tracks.

Track codes are generated by 6.25 mrad binning of the \( \phi (r_j) \) at each \( (j = 0, ..., 4) \) detector layer. \( \phi (r_j) \times 160 \) is rounded to integers \( tc(j) \) within \( (0, ..., 99) \):

\[
\begin{align*}
tc(0) &= \phi (r_0) \times 160 \\
tc(i) &= (\phi (r_j) - \phi (r_0)) \times 160 \quad j = 0, ..., 4
\end{align*}
\]

and rotated by the angle of at first detector layer, exploiting the rotational symmetry. We construct the track codes (trcode) from the \( tc(j) \) in the following way:

\[
\text{trcode} = 10^6 \times tc(4) + 10^4 \times tc(3) + 10^2 \times tc(2) + tc(1)
\]

By sorting and getting the multiplicities of track-codes we can get a track-code distribution for 20 Gev as shown in Fig. 4. The multi-peak structure of code multiplicity is caused by the Markov-like process of the statistical behaviour of Coulomb-scattering when using the diffusion angles: the later points along the track remember the effect of previous deviations, and the codes are grouped into families corresponding to the main branches. Fig.5 shows the five families identifying the main peaks in Fig.4.

Assuming uniform momentum distribution Fig.6 shows, in a typical case, how frequently a muon with momentum \( p \) is able to produce the given "6101111" track-code. The \( \text{FWHM} = 4 \text{ GeV/c} \) implies a resolution better than 10% in momentum at this track-code value.

For triggering purposes one is not using directly a cut in the track-code list, because it is not ordered \( a \) priori according to the momentum due to the smearing caused by the Coulomb-scattering. One can assign, however, to each track-code, a value \( P_{\text{max}}(TC) \) by a simple look-up table. Therefore, an event is accepted only if:

\[
P_{\text{max}}(TC) > P_{\text{threshold}}
\]

For example, if one intends to select 20 GeV/c muons, one can see on Fig. 7. that the track-code "5080909" still has some 20 GeV/c content at its lower tail, and the
the track-code "5080909" still has some 20 GeV/c content at its lower tail, and the "7111212" track-code at its higher end.

The number of track-codes "N_{tc}" depends on the $\phi$ binning. The value of $N_{tc}$ will determine the muon search execution time when using the ASP; let us stress at this point that, on the contrary, the triggering time will not depend upon the muon multiplicity. The granularity $\Delta\phi$ is dictated by the size of the Coulomb-scattering at $P_{\text{threshold}}$. Due to the $1/p$ scaling, one can limit the number of valid track-codes, by applying an appropriate $\Delta\phi$ value, independently of $P_{\text{threshold}}$. In this way one can assure that $N_{tc}$ will be below 100 for all muons above the given threshold, which makes the online decision fast enough.

**Track-code distribution for 20 GeV/c**

![Track-code distribution graph](image)

**Track-code families**

Fig. 4 – Muon track distribution according to track-code values.
Fig. 5 – Track-code families
Fig. 6 – Momentum track-code selectivity for “6101111”

Fig. 7 – Momentum track-code selectivity for 3 track-codes
3.1.1.3 Master point algorithm

In each superlayer the "master-points" with coordinates \((r, \phi)\) are calculated from several number of individual hits. For sake of definiteness we assume, that each superlayer consists of 4 detector layers. These layers are so close to each other that the azimuthal angle change is minimal during the passage of a particle. Consequently:

\[
\alpha_{i+1} = \{ \alpha_i - 1 \quad \text{or} \quad \alpha_i \quad \text{or} \quad \alpha_i + 1 \}
\]

are the only possible coordinates on the subsequent plane.

During the loading procedure all hits in all the 5 times 4 layers are stored in the usual "iconic" way in the APE string. The azimuthal angle \(\phi_j (j=0, 1, 2, 3, 4)\) for the master-point on superlayer \(j\) is calculated using the hits on the detector layers \(k\):

\[
k = j \times 4 + i \quad (i = 1, 2, 3, 4)
\]

The calculation is complicated, however, by the facts that:

a) due to the chamber inefficiency there will be no hit in some layers;

b) the passage of a given particle can produce double or multiple hits in the same detector layer.

We propose an "iconic average" to calculate the master-point coordinates. This iconic average is done in the following way: the inefficiencies are taken into account by "OR-ing" layers #1,2 and #3,4 respectively (see figure 8.)
Then the "AND-ing" of these "OR-ed" results produces $\phi_j$. Assuming detection efficiency $\varepsilon_\alpha$, the master-point detection efficiency will be:

$$\varepsilon_{\text{master}} = (\varepsilon_\alpha^2 + 2\varepsilon_\alpha (1 - \varepsilon_\alpha))^2$$

For $\varepsilon_\alpha = 0.9 (0.95)$ one gets $\varepsilon_{\text{master}} = 0.98 (0.995)$. Due to the fact that for a valid track-code one needs all the five master-points, the total track finding efficiency will be

$$\varepsilon_{\text{track}} = \varepsilon_{\text{master}}^5 = 0.9 (0.975),$$

that is even in case of chambers with very low efficiency one can reconstruct the tracks with reasonable efficiency.

In ASP the master-points as "centers of gravity" are identified by shifting (for taking into account the $\alpha_i$ jumps) and bit-logic operations. In the next we assume that we are looking for positive muons only, where one knows a priori the expected slope direction, determining the shift pattern. Of course, the negative muons are treated in the same way in the parallel system. The procedure is executed in 3 phases for each superlayer separately:

a) The "OR-ing" 1 and 2 is stored in bit 'A' after the appropriate shifts;
b) The "OR-ing" 3 and 4 is stored in bit 'B' after the appropriate shifts;

c) The "AND-ing" 'A' and 'B' is stored in bit 'C' and eventual shrinking from left and right side produces the final point.

The method is illustrated in figure 9, for different hit pattern combinations. The estimated speed of this algorithm is about 1 μs / superlayer.

3.1.1.4 Conclusion

As a summary we conclude that a second level trigger for muons at LHC is feasible. The three phases of the procedure: the loading (detector mapping into the ASP), the preprocessing (Master point determination) and the tracking (track-code algorithm), are well adapted to the ASP architecture.

The preliminary results of the simulation show that the trigger can be worked out within about 20 μs. This timing is fitting the requirements imposed by the expected first level trigger rate at high luminosity LHC.
Fig. 9 - "Iconic average" illustrated in four typical tracking situations

Track #1: Ideal straight track, full efficiency
Track #3: Inclined track with 2 missing hits
Track #2: Inclined track, full efficiency
Track #4: Inclined track with multi-hits and a missing hit
3.1.2 CERN - TRD electron selection at LHC

The ASP in the TRD Second-level Trigger

R.K.Bock and J.Pfennig, J. Toth

3.1.2.1 Progress and ongoing work on the TRD application

For use in a future LHC detector, a transition radiation detector has been proposed [1], whose capability in e/hadron discrimination can be put to use in real time. Since proposing this detector as an application of ASP, major progress has been made in defining the role of the ASP's SIMD architecture in the context of real-time decision making. This is particularly relevant since the problem of the TRD real-time decision is typical for a class of real-time problems as they are encountered around most future LHC subdetector systems. We therefore expand on this problem, and on the ASP's share of the solution, in some detail below.

Over the next months, the ongoing work of simulating data from a TRD in single-channel detail, and on sending them through an overall second-level trigger system, will be continued and refined. Eventually, we plan to demonstrate the system's performance under different physics and luminosity conditions, and to focus on the low-level algorithm which will execute in the ASP array. This can then be coded on the ASP simulator, or even run on the Trax-1 hardware, for understanding in detail the expected performance. We also plan (in the context of the EAST collaboration, see [2]) to build a hardware emulator for a fraction of the TRD readout, which will allow later to demonstrate critical transmission and algorithm components under conditions as close to the future detector specifications as possible.

3.1.2.2 The TRD triggering algorithm

We use for the TRD a straw configuration called the 'halo model' in [1]; it is schematically shown in fig.1. Straws are arranged between 70 and 120 cm of radius around the beam pipe, in multiple planes (a total of about 900), and about 580 straws to a plane. In order to get optimal tracking capabilities from this detector, every alternate plane carries
the straws in a reverse arrangement, thus allowing full stereo views. Multitrack ambiguities can arise in such a tracking arrangement, but due to the non-parallelism of straws they are minimised.

The final readout of the roughly 500 000 straws (on the outer detector surface, one straw end only) is proposed to be largely centralised off the detector, to avoid the various problems encountered by on-detector intelligence: radiation, obstruction, and power dissipation. For the intended trigger, this siting issue is secondary, except that full access to all channels must be possible. We assume that the front-end readout takes care of signal extraction for individual bunch crossings, and remains passive until a (calorimeter-based) signal for a given time slice is received, indicating both that the corresponding bunch crossing establishes one or more electron candidate(s), and where these candidates are to be found ('pointer'). The assumption is made that such bunch crossing candidates do not occur more frequently, on average, than every 10 μsec., i.e. the first-level trigger reduces from the bunch crossing rate by a factor of 1000.

The basis of discriminating electrons from hadrons in a TRD is a statistical analysis of pulse heights of all digitising belonging to a track, so that a measure for the probability of X-ray emission (and detection) can be determined. This probability is a function sensitive to the Lorentz factor \( \gamma \) of the track. Hence the finding of a (stiff) track pointing towards the calorimeter cell signalling an electron candidate is a first local step, its (statistical) pulse height analysis a second local step, and a decision taking into account correlations of (multiple) signals, physics cuts, etc., is a concluding global step of the algorithm. All computations have to be preceded, for practical reasons of bandwidth and performance, by a data selection step under control of the first-level pointers.
Plan de Mesure

Diamètre 2400 mm
Diamètre 1900 mm
Diamètre 1400 mm

288 short straws
288 long straws
576 straws left oriented

Fig.1 The Halo Model of the TRD
3.1.2.3 The possible role of the ASP in the TRD trigger

Due to the fact that tracks can pass through roughly one quarter of the longitudinal extension of the detector and due to the statistical nature of track detection, as outlined above, a single algorithm running in a single hardware unit is not conceivable for the decision rate of 100KHz. We are interested in parallelizing and/or pipelining the process to a maximum, hence we propose to subdivide the 2nd level trigger into three phases, each of them implemented on a different type of hardware and running a dedicated algorithm. The following sections will expand on the proposed practical solutions, as we envisage them at this early state of design.

3.1.2.4 Phase I: data distribution

All data belonging to bunch crossings that passed the 1st level trigger selection, will first be synchronised and then transported from the experiment site to the main trigger electronics. The synchronisation has been introduced to allow simple point to point communication between the readout logic and the trigger, as well as between the phases of the trigger. The TRD trigger, internally, will operate synchronously, or in a data-driven fashion, synchronised by the data flow.

At the input side the full data rate is 100 GBit/s. All incoming data will be kept during a period of some msec in public addressable memory, to allow other components of the experiment to monitor the TRD and the 2nd level trigger performance. The data transport at this level could be based on the HIPPI standard [3].

Only a fraction of the data is relevant for tracking in the trigger. Using the 1st level trigger information and limiting the number of electron candidates to four (independent calorimeter candidates), a fraction of the data (something like 3-4%) will have to be mapped into a data window corresponding to a gross 'road' from the calorimeter area to the vertex zone. Up to four such data windows will be allowed (for simultaneous first-level triggers), and this data fraction will undergo further processing. The retained data
then is mapped from the TRD specific straw representation into plane rectangles of pixels. The plane contains the beam axis and the firing calorimeter cell, and for each potentially interesting track such a plane containing all space points belonging to it can be constructed.

*Due to limitations in transmission rate and the simplicity of the selection mechanism, the ASP is not considered a suitable architecture for this phase I.* Data routing will be based on loadable Look-Up tables, used by DSPs or similar hardware for controlling block transfers (no buses, but of the order of 512 local point-to-point connections).

### 3.1.2.5 Phase II: low-level feature extraction

Every 10 μsec (one time slot) the data of several planes (depending on the calorimeter resolution) generated from each of the four data windows is sent to the phase II hardware. This could be accomplished using the some type of HPPI links mentioned above. Processing now consists of 'histogramming' in parallel along all possible three-dimensional roads that can contain straight tracks from the vertex zone to the calorimeter zone. Massively parallel processing with a modest algorithmic flexibility is required.

For reasons of scalability and expected performance limits, the phase II processors are divided into several groups which work in parallel on different events (time slots); they are used in a pipelined fashion. This is required in order to extend the interval that is available for processing from 10 to 100 μsec. Each time slot is assigned to a processor bank on a round-robin schedule. On the input side, the data can be further reduced by requiring digitising to be inside of roads defined by the calorimeter information from the first-level trigger. See fig.2 for an overview.
ASP in TRD
Second-level trigger architecture

From Detector

First-level trigger

Analog signals pipeline
Flash ADC-s

Data Synchronization
Transfer to latency buffer

Latency buffer
Window selection

Phase I
Transfers & Selections

Phase II
ASP

Phase III
RISCs? ASPs?

Event banks
Low-level algorithm

Multi-bank decisions
High-level algorithm

Access to results
Access to all TRD data

Experiment

Fig. 2 – Overview of the TRD acquisition
The processing of phase II, i.e. low-level trackfinding including the use of pulse height information, can be parallelized to run on a SIMD architecture. *The ASP is a candidate for providing the necessary processing power.* An ASP implementation of phase II might use 10 Fastbus crates with a total of about 640,000 processing elements, on 80 boards, as well as 40 controller/road-builder boards. The processing power of this solution allows to check for about 128 plane areas, each one covering a trapezoid of 16 TRD slices out- and 32 TRD slices inside (per time slot).

In our present thinking, the track finding algorithm is based on shifting (distorting) the trapezoid and histogramming in parallel. The shift (angle) for the highest value is saved for each bin, and should, in an ideal case, coincide with a track. A balance has to be found between calorimeter tower size, vertex area, and histogram bin size, these parameters are under study now.

3.1.2.6 Phase III: decision taking

As a result of Phase II the n (about 64) best peaks would be read out (still under control of the ASP Hardware) and transferred to n corresponding processors in the phase III hardware. Now physics-dependent decisions will be necessary (simple thresholding will not be good under all circumstances), sensitive physics- or luminosity-dependent cuts and limitations are introduced, and global access to data, possibly even including other detector parts may be required. SIMD-type parallelism is no longer possible. This high-level flexibility is not provided by the ASP. Processors in phase III are assumed to be conventional RISCs or transputer-like nodes, programmed under C in a real time operating system environment.

4. References


3.1.3 ORSAY - Calorimetry, fast selection on jets at LHC, shower detection

E. Augé, J.-L. Bertrand, A. Ducorps, B. Lavigne, Ph. Heusse

3.1.3.1 Introduction

An ASP simulation software is used since July 1990 in Orsay to develop selection algorithms coded in ASP language, and to estimate the time to run those algorithms. This work relies on working hypothesis concerning the architecture of the second level trigger and its environment. Some of them can be modified according to our ability to develop faster algorithms. The work described below therefore allows to have a rather precise view of the possible architecture for an ASP-based second level trigger.

3.1.3.2 Hypothesis

At the moment, we have considered only a "barrel" calorimeter, and done nothing specific for possible "end caps". We have supposed that this detector is segmented in four stacks in depth (electromagnetic front and back, hadronic front and back). We have supposed that an hadronic cell is twice as large as an electromagnetic cell both in Front and Back (see figure).

We associate one EP to each e.m. cell and one EP to each had. cell. For each of them, both the front and the back stack energies are stored into the same EP.

We define patches of 32X32 e.m. cells plus the 16X16 corresponding had. cells. This means a sub-string of 1280 EPs.

To avoid complications, each cell at the border of a patch is loaded into two different EPs. The number of "useful cells", where an electron is searched for is only 28X28 (see figure 1).

We have also supposed that the event buffering at the input of the second level trigger is such that the only constraint on the time needed to treat an event is that it should not exceed 100 µs ON AVERAGE.
3.1.3.3 Loading time

We estimate the loading time to be of the order of 10 to 15 $\mu$s. If the Vector Data Buffer feature is available on the ASP, most of this time will be overlapped with the processing of the previous event, and the real cost of loading will be only 1$\mu$s.

The possible presence of Zero-suppression before the second level trigger is an open question. In one case, the data from all cells always come in the same order (50 ns per word, 100 ns per cell-front + back). On the contrary, if the cells with energy below some threshold are not presented to the ASP, each cell must be identified with an address within the patch. This address is also permanently stored in the corresponding EP. Loading an energy in an EP using this address takes 100 ns (i.e. two time slower than previously). However, the gain due to this possible zero-suppression is not uniform along the calorimeter: some patches will be almost empty, but some others might have more than 50% of the cells above threshold. Therefore, we expect comparable loading times in both cases.

3.1.3.4 Rejection criteria

The fraction of truly interesting events at the input of the second level trigger is very small (of the order of $10^{-6}$ at most). Therefore, we have tried to combine the different selection criteria in order to get the fastest possible rejection. We anticipate to reject 90 to 99% of the events, with an average time of 50 $\mu$s per rejected event. For the selected events, more time (300 to 500 $\mu$s) is available in order to diminish the amount of data transmitted to the next step in the data acquisition, and to make higher level decisions faster.

The algorithm described below, and the numbers are preliminary, and we are still working to improve them. We try to select high transverse momentum isolated electrons. Therefore, the selection algorithm implemented so far requires successively:

* an energy in 3X3 cells in the e.m. front stack above some minimum

This takes around 30 $\mu$s (only 1.2 $\mu$s if no individual cell above some low threshold) and is done simultaneously around each cell of the patch.
• an energy in the had. front cells corresponding to some energy deposit in the e.m. front stack below some maximum.
  This takes 1.3 μs

• an energy in the whole e.m. compartment (front + back) above some minimum.
  At the same time, we compute the precise coordinates of the cluster barycentre in the e.m. compartment. The whole operation takes around 30 μs.

• The coordinates of the shower barycentre are transmitted to another device to associate this shower to a possible signal in a preshower detector. This provides a very efficient rejection against π⁰ mesons.

• Simultaneously to this association, we compute the energy deposited in a 5×5 neighbourhood, estimate whether the electron candidate is isolated, and measure the width of the energy deposit.
  This takes around 20 μs.

  The rejection power of each of these criteria has not yet been estimated. It depends strongly on the rejection already obtained by the first level trigger. The average time needed to reject an event is therefore not known. However, we can anticipate it will stay below 50 μs.

3.1.3.5 Preprocessing

No algorithm has been coded yet. We think of computing the energy deposit over a large number of cells (typically 12X12 in the e.m. sector and 6X6 in the had. sector) to identify and measure jets. It seems important that this operation is done (simultaneously) around each cell, because the central cell of the jet might not be easily identified. This would ease a next step in the trigger selecting electron + jets, and computing invariant masses.

We also think of a very simple algorithm to transmit to the next steps of the data acquisition chain only those cells belonging to a significant cluster ("intelligent zero-suppression").
A Calorimeter Tower

Patch arrangement

Fig. 1 – Calorimeter modelling
3.1.3.6 Conclusion

The work presented is far from being complete. In particular, we do not know the jet rejection which could be achieved by the criteria mentioned above. However, we already have evidence that those criteria can be implemented on an ASP-based second level trigger local processor in an LHC experiment.
3.1.4 \textit{SACLAY - First ASP algorithms for SSC/SDC}

\textit{M. Mur, B. Thooris}

We present here some ASP simulations for the SSC/SDC (Solenoidal Detector Collaboration) calorimeter which could be useful in a ASP trigger.

3.1.4.1 Calorimetry

In any SSC detector, the calorimeter is the only component able to measure jet energies, and is essential for the detection of neutrinos and others unseen particles. In the SDC detector, the calorimeter will also be able to distinguish electrons from charged hadrons. The technology for calorimeter is not yet chosen, but several features of the design are invariant to choice. The barrel calorimeter will be installed and fixed within the magnetic iron used for muon identification and it will provide the support for the solenoid coil and the tracking detectors. Using a scintillating tile calorimeter, a preliminary design envisages a barrel calorimeter consisting of 128 wedges evenly distributed in $\phi$ ($\phi$ is the azimuth angle) with absorber plates normal to the particles direction. The endcaps are composed of similar wedges. The SDC set-up is segmented in depth in 5 stacks. (tracking system, shower max., electromagnetic detector, hadronic detector, muon detector).

3.1.4.2 Trigger decision

First purpose of trigger is to make an electrons selection. An isolated electron pixel is characterised by electromagnetic energy value greater then a EM-threshold, hadronic energy value lower than $H$-threshold, and no direct neighbour with EM-energy greater than EM-threshold. A second kind of problem is to calculate transverse energy $E_t$, sum of $E \sin \theta$ on each pixel ($\theta$ is the polar angle).

3.1.4.3 ASP-Trigger algorithms

Let's assume that calorimeter is segmented in 128 in $\phi$ and in 82 in $\theta$. One APE is corresponding to each pixel (it represents a 10496 string) All information about the 5 stacks pixel data and neighbouring EM-energies are stored in each APE. Programmes have been performed on VASP-Simulator and timing estimations are the following :
1. Isolated electrons:

1st threshold = 1 microsecond (to know electrons existence)
2nd threshold = 1 microsecond (to separate from hadron)
3rd threshold on neighbours and counting = 5 microseconds.
Global time = 7 microseconds.

2. Missing Et:

Timing is very dependent on clusters number and geometry. Average time can be estimated about 20 microseconds.
3.2 Fixed target applications

3.2.1 SACLAY - NA31', the level 2 neutral trigger system.

* M. Mur, B. Thooris

3.2.1.1 Abstract

The NA31' experiment is a CERN (SPS) experiment intended to run in 1994-96. It aims at a high precision measurement of the $e'/e$ parameter in order to have a better understanding of CP violation. The experimental set-up is designed to measure charged ($\pi^+\pi^-$) and neutral ($2\pi^0$) decays from $K^0_S$ and $K^0_L$, concurrently. The high flux required to obtain sufficient event statistics during the 3 years running period puts high constraints on the on-line trigger system, to reject physics background and accidentals.

In the neutral channel, the trigger system has to struggle against a very high $3\pi^0$ background, and detect accidental hits which could bias the trigger decision. The target is to obtain a very good signature of $2\pi^0$ candidates in less than 10 $\mu$s, taking as inputs the energy deposits in the 12000 cells of the electromagnetic calorimeter array. Valid events should give exactly four photon shower clusters in the detector. These clusters should have a null energy first momentum, and the vertex calculation, derived from the invariant $K^0$ mass expression, involving the total energy deposit and the energy second momentum, should evaluate to the correct fiducial interval. These calculations should be invalidated if an accidental hit occurred in the calorimeter in the sensitive time window.

We tried to define the best processing candidates for each of the trigger and accidental detection sub-tasks, in order to obtain a compact and flexible system. Processing time was evaluated for the ASP (using either 1D or 2D algorithms), and also for digital signal processors (DSPs). ASP is better suited for the topological processing tasks (find clusters and count them, find accidentals and locate them relative to the normal signal timing), while DSPs are better on the fast, high accuracy, stream arithmetics required in the energy balance and vertex calculations.

In the current state of our investigation, we propose to start from energy sum records in the X and Y projections (1D processing). The system assembles eight concurrent sub-machines, under control of a main controller. Cluster counting (on X and Y projection) and accidental identification are implemented by 3 independent ASP systems.
(a total of 4098 APEs). The arithmetic partial sums \( \Sigma E_i X_i \), \( \Sigma E_i Y_i \), \( \Sigma E_i X_i^2 \), \( \Sigma E_i Y_i^2 \), and \( \Sigma E_i \), used in the first momentum and vertex position evaluation, are each run by an independant DSP. The main controller combines partial results, supports the very high speed dialogue with the full experiment trigger system, and provides initialisation and various services. It may be a DSP or a similar microprocessor.

Investigation continues at the moment to understand the merits, in the available time, of 1D versus 2D processing. Accuracy, signal conversion, and error propagation are also studied with the help of Monte-Carlo events.

This application is an example of the use of ASPs in prompt trigger systems, where real time response performance, and fast parallel loading capability is of prime importance. A simple, fast, and compact controller must be designed, to accommodate system with several independent, concurrent ASP sub-machines. Tradeoffs between simplicity and flexibility must be studied, and the corresponding software environment must be adapted.

3.2.1.2. Proposal for the level 2 neutral trigger of NA31'

1) Introduction

We tried to design the prompt neutral trigger (level 2) of the proposed NA31' experiment (CERN, SPS) using ASP and DSP programmable processors.

This document first summaries the neutral trigger requirements. The performance of the basic trigger algorithms, run on the ASP and DSP simulators are predicted. Then, a possible implementation using multiple, parallel ASP and DSP machines is presented.

2) Neutral trigger requirements

The following figures relevant to the neutral trigger may be derived from the NA31' proposal:

- Level 1 rate is around 46k/s;
- The main neutral physics background is from $3 \pi^0$ events (65k);
- Interesting $2 \pi^0$ events are around 273/s;
- Trigger(2) processing time must be in the 10 µs range.

3) **Trigger tasks**

The following tasks must be performed on the EM calorimeter data to identify the $2 \pi^0$ events (or better, to filter out $3 \pi^0$ events):

4) **Cluster counting**

Four and only four photons must be detected in the EM calorimeter. The task consists in accurately finding and counting shower clusters.

The most natural way of counting is to keep the 2D image structure of the detector, group cells in 2D clusters, and finally count the clusters throughout the image. This algorithm, though very time consuming with traditional methods, may run very efficiently on the ASP processor. However, this approach may turn out to be very costly in this particular context, since it requires fast access to the 12000 cell energy deposit values, through (moderately) fast and accurate digital converters.

Counting may also be done in 1D along X and Y projections, by finding the number of local maxima in each X and Y energy sum record, independently. In that case, the number of event patterns leading to spatial ambiguities is increased: Events with less than four peaks must be accepted, and $3 \pi^0$ events will not all be filtered out. But since the evaluation of local maxima may be done very accurately, this method is rather strong on counting clusters close by each other.
5) **$P_T$-conservation**

Initial zero transverse momentum conservation results in:

$$\sum E_i X_i < \varepsilon$$

$$\sum E_i Y_i < \varepsilon$$

where $X_i$ and $Y_i$ are the algebraic positions (relative to the centre of the calorimeter) of the cells where energy $E_i$ is deposited. This first moment calculation may be performed on individual cell information, or on $X$ and $Y$ energy sums records.

6) **Kaon mass estimate**

The initial kaon mass $m_k$ may be estimated by the following formula, when the fi: moment sums vanish to 0:

$$m_k^2 = \sum E_i \ast (\sum E_i X_i^2 + \sum E_i Y_i^2) / D^2$$

($\sum E_i = E_{\text{tot}}$, is the total energy deposit, $D$ is the vertex distance from the calorimeter).

An interval cut on the reconstructed mass will reject background events faking two $\pi^0$, balanced, four clusters patterns in the calorimeter.

The above expression is only an approximation of the true invariant mass, since the second moment summations are performed on individual pixels (or on $X$ and $Y$ energy sums), whereas they should be done on resolved clusters (the real expression involves accurately located, point-like clusters carrying the full shower energy). The magnitude of the error generated by this approximation needs to be understood.

7) **Accidentals**

The high flux condition may induce computation errors in the above tasks if an accidental hit is taken into account. In order not to reject events under such conditions, a specific circuit is included to detect accidentals occurring in the sensitive trigger time window.
8) **1D versus 2D: discussion**

Computations for the 3 trigger tasks, and for the accidental identification, may be performed on individual cell information (2D), or in 1D after energy summation along lines and columns.

9) **First and second moments**

First and second moment calculations will run faster, cheaper, and probably more accurate\(^1\) in 1D.

10) **Photon counting**

Counting may be performed in 2D or 1D. The rate at which 3 \(\pi^0\) events (6 photons clusters) will fake 2 \(\pi^0\) configurations is not well known yet for both cases, and needs simulation.

2D counting will not be good at resolving clusters which are very close by each other in the 2D image, because local maxima identification inside a 2D cell aggregate seems hardly possible in the available time\(^2\).

1D counting will not resolve clusters which project in the same or neighbour \(X\) (respectively \(Y\)) line (respectively column). However, since both projections are observed together, and due to the better accuracy on the local maxima identification, this method may prove as efficient at reducing the 3 \(\pi^0\) background. In addition, since the \(X\) and \(Y\) energy sums will be made available for the \(P_T\) and mass calculations, counting should anyway be done in 1D (unless further studies show that the accepted background rate is much higher than in 2D).

\(^1\)Higher resolution ADCs may be used, and still be affordable for 2*128 channels.

\(^2\)But could be done later (at level 3) in a 2D ASP machine.
11) **Accidental identification.**

To detect accidentals inside the trigger time window, calorimeter signals must be searched for an out-of-time peak. This must be done with a time resolution in the 10ns range (?). Such a resolution requires waveform recording, peak time finding, and peak time matching for all signals under survey, and might also imply a special analog shaping to be able to resolve cases where an in-time signal and an out-of-time signal appear on the same channel.

Here, we propose a scheme where such an algorithm is applied on 1D projection sums (or at least one of them). This scheme could also apply to each individual cell (or any group of cells), but complexity and cost will certainly be the limiting factor.

12) **Candidate processing devices**

We think that ASP and DSP processors may together provide a very attractive solution to this trigger problem, resulting in a programmable, flexible, testable and compact system at the better cost.

A particular ASP system is a string made of a (normally large) number of elementary processors (APEs), attached to each elementary data element (e.g. one processor per pixel in image processing applications). All APEs run the same algorithm (SIMD machine), but may be dynamically and selectively arranged into various sets depending on the data values they store locally. Vector-vector or scalar-vector operations involving all or part of the array may act on selected APE data fields, and use logic and bit-serial arithmetics capabilities available at each APE location. The programmer may alternately see the APEs as a complete string, or as a set of independent segments.

A typical trigger algorithm will try to search the data set for specific patterns, according to various successive sorting criterions. The ASP will rather naturally apply the sorting criterions on the data set, using an associative matching method. The criterions are applied to all APEs at the same time. For most algorithms, the resulting performance is almost independent of the full data set size, since no scanning or looping operation has to be ever performed on all data. For some algorithms, where most calculations are
performed in place, the resulting performance may also be independent of the number of interesting data in the data set.

Due to its associative capabilities, the ASP is thus very strong on problems involving a small set of interesting data spread in a large raw data array, arranged in a topology involving local vicinity.

13) **Digital signal processors (DSP)**

DSP processors are more conventional microprocessors, which are optimised to be very fast on typical DSP algorithms (digital filters, FFTs, etc...). Multiple internal and external buses, pipelined ALUs, parallel data addressing facilities and inter-chip communication support allow very high stream processing performance. In particular the basic multiply-accumulate primitive may execute in one instruction cycle in most DSPs.

Both fixed-point and floating-point DSPs are currently available.

14) **ASP performance**

The ASP performance on the various trigger tasks was benchmarked with the ASP software simulator, provided by ASPEX, and run on a SUN workstation.

3.2.1.3 **Execution time**

1) **Counting the photons**

On the ASP, counting items matching a specific criterion is achieved with a binary tree reduction algorithm, which results in a \( \log_2 N \) dependence on the number of items to count. In our application, it is possible to indicate very early if there are more than 4 items (photons): First, counting of (up to) 4 items is done in 2 iterations of the pairing algorithm. After 2 iterations, the ASP array is asked if there are still items which were not yet counted. A positive answer indicates that there were more than 4 items.
2) **Cluster counting in 2D**

Steps:
1- Apply global threshold.
2- Find pixel aggregates, and mark only one pixel per aggregate (cluster).
3- Count marked pixels, using the binary tree reduction indicated above.

Time prediction (includes thresholding): 6 µs.

This prediction is slightly dependent on the image width (this estimate is given for a 64*64 grid).

3) **Cluster counting in 1D**

Steps:
1- Apply local threshold.
2- Find local maxima:
   - Compare all data elements to their left neighbour concurrently, and mark if (greater than or equal).
   - Compare all data elements to their right neighbour concurrently, and mark if (already marked and greater than).

Time prediction: 5 µs (runs in parallel on X and Y).

4) **First momentum**

Steps:
1- Calculate EiXi, in place, concurrently for all data elements.
2- Binary tree summation of partial elements, concurrently for all segments.
3- Binary tree summation of segment partial elements.

Time prediction: First momentum calculation (includes thresholding) 8->15 µs.

Depends on the number of clusters and on the size of clusters. Calculated concurrently for each projection.

5) **Second momentum**

Steps:
Same sequence as for first momentum.
Time prediction (includes thresholding): Slightly more than for first moment, due to wider $X^2$ and $Y^2$ bit representations (APE arithmetics is bit-serial).

6) Accidentals

For each line (column) energy sum element, successive waveform time samples are stored in successive APEs inside the ASP string. Each element is then represented as a (8 or 16 APEs deep) segment.

Steps:
1. Apply global threshold.
2. Find local maxima concurrently for all segments.
3. Identify and mark any APE where a local maximum was found out-of-time, i.e. before or after the normal pre-declared peaking region (1, 2, or 3 time slots wide).
4. Flag event if at least one was found out-of-time.

Time prediction: 5 $\mu$s.

7) DSP performance

Although very fast, current DSPs cannot reach the ASP performance level on the counting task, since thresholding, looking for local maxima, and counting would require several instructions for every input data scan. The accidental detection system is also much faster on the ASP, due to parallel thresholding, parallel per-channel and per-time slot maxima identification, and associative identification of out-of-time signals.

However, for the arithmetic summations tasks, DSPs may be faster and more accurate. The latest generation of DSP processors may run the basic multiply-accumulate (MAC) sequence required to evaluate the $\Sigma E_i X_i$, $\Sigma E_i Y_i$, $\Sigma E_i X_i^2$, $\Sigma E_i Y_i^2$, and $\Sigma E_i$ partial elements in the required time.

This sequence will run fixed-length on 128 successive data inputs. Pre-stored coefficient arrays provide $X_i$, $Y_i$, $X_i^2$, and $Y_i^2$, and are indexed automatically for each new input data. Input data are transformed through a look-up table applying pedestal and subtraction and energy scaling (if required).

If the energy look-up is external, the resulting processing time is then simply $(128 \times \text{multiply-accumulate}[\text{MAC}] \text{ time})$. Since the processing time is known in advance,
independent on the data contents and accuracy, DSPs are better candidates than the ASPs for these short run arithmetic sums.

In our context, even the $\Sigma E_i X_i^2$ and $\Sigma E_i Y_i^2$ results will not saturate a 32 bit integer accumulation for 12 bit energy values and 128 cells. Any integer DSP with at least 32 bit accumulation, zero looping overhead and sufficiently fast MAC cycle could be used. To fit the 10 ms constraint, the basic MAC loop element should run in the 75 ns range or less, leaving enough time to convert input data and synchronise before the loop, and some time at the end for the main controller to combine all intermediate results into the trigger 2 response.

Current devices having the the required performance level have been identified\(^3\). Next generation devices\(^4\) might allow to insert the energy look-up operation inside the software loop.

8) System architecture

Following the previous estimates and constraints, a (tentative) structure for the full system is proposed: 8 sub-machines run low-level programmes concurrently, under supervision of a main controller. The arithmetic partial elements $\Sigma E_i X_i$, $\Sigma E_i Y_i$, $\Sigma E_i X_i^2$, $\Sigma E_i Y_i^2$, and $\Sigma E_i$, are each run by an independent DSP. Cluster counting and accidental identification are implemented by 3 independent ASP systems. The Main controller is in charge of combining partial results, supports the very high speed dialogue with the full experiment trigger system, and provides initialisation and various services.

This first approach is based on analog energy sums. This may prove impractical, and very sensitive to (correlated, digital) noise. In the final system, data for the level 2 trigger system will certainly be generated as a by-product of the normal measurement electronics: Individual cell energy digital measurements could be used to generate noise free energy sum values for the trigger.

\(^3\) AT&T DSP16A, Motorola DSP56001, ...
\(^4\) Texas Instruments TMS320C50, ...
Block diagram, the system elements are arranged in the following way:
3.2.1.4 Task parallelism

All subtasks are fully independent. The cluster counting sub-machines aborts all other sub-machines if more than 4 clusters are identified. First moment tasks may be run independently on X and Y projection. Second moment accumulations may be run in parallel on X and Y projections, and have then to be correlated to test the mass estimate. Accidental identification is also run in parallel, and sends a flag to the main controller.

3.2.1.5 Main Controller

The main controller must be a programmable processor capable of fast response to external events, of high speed scalar and floating point calculations, and of easy interface with the rest of the world (and especially the main trigger system of the experiment), even at remote locations. Data i/o throughput is of minor importance in this specific trigger application (data input is to several sub-machines concurrently by special paths, and data output is limited to very few intermediate sub-machine results). The current generation of high-end floating point DSPs seems appropriate.

1) **Interface with the trigger system**
   - Receives Trigger(1) OK, and Trigger(2) Abort;
   - Sends Trigger(2) OK, Trigger(2) Abort;
   - May send Trigger(2) estimates (mass, cluster locations, event class...) for event post-processing and algorithm validation.

2) **Synchronisation of data transfers**
   - Starts/monitors front-end data scan to concurrent sub-machines.

3) **Sub-machines coordination**
   - Starts/monitors/aborts sub-machines pre-stored programmes;
   - Receives attention signals from sub-machines;
   - Has memory-mapped access to specific ASP/DSP dual port memory areas.
4) **Services**
   - Supervises initialisation, programme down-load, test operations.

5) **Sub-machines**
   ASP Controller:
   - Runs pre-stored programme when started by Main Controller;
   - Sends attention signal to Main Controller when programme complete;
   - Ready to abort at any time on Main Controller signal;
   - Provides message i/o space to Main Controller for variables/results.

3.2.1.6 **Data I/O**

1) **Input streams**
   If a 2D machine is needed, loading constraints will be very high and require highly parallel data paths to the ASP machine. 1D loading constraints are much lower.

2) **XY pixels**
   If a 2D machine is required, loading must be performed highly parallel. For the current ASP chips (64 APEs, 32 bit @50ns input bus), the minimum loading time is $16 \times 50$ ns (1.6 ms) for 8 bit wide cell data (requires 1 -> 4 unfolding inside the ASP). For a $64 \times 64$ grid, 64 32 bit parallel, 20 MHz channels must be implemented from the detector electronics to the ASP machine.

3) **Xsum and Ysum energies**
   Energy sums along 128 lines (columns) are built in analog form and converted by moderately fast, high resolution ADCs (e.g.: 1 $\mu$s conversion, 12 bits resolution). 2 parallel X and Y data streams are implemented, resulting in parallel loading for each $\sum E_i X_i$, $\sum E_i Y_i$, $\sum E_i X_i^2$, $\sum E_i Y_i^2$, and $\sum E_i$ sub-machine, at the DSP input rate.

4) **Xsum and Ysum waveform time records**
   Energy sums along 128 lines (columns) are built in analog form and converted by very fast, moderate resolution ADCs (e.g.: 10 ns conversion, 8 bits resolution). A number of samples (e.g. 16) is stored into a fifo buffer memory. Data from 4 neighbour lines (respectively columns) buffer memories are then fed parallel into one single ASP chip.
Loading time into the ASP is 64.50 ns (3.2 μs) in this configuration.

5) **Output streams**

5.1) **Counting sub-machines**
Number of clusters.

5.2) **Arithmetic machines**
First moment sum on X projection.
First moment sum on Y projection.
Second moment sum on X projection.
Second moment sum on Y projection.

3.2.1.7 **Final Decision**

The Main Controller waits for completion signalling from all concurrent sub-machines, and reads partial sub-machine results on a first-come-first-served basis for correlation and logical decision. When sufficient (partial) information is collected to generate the final decision test, the Main Controller may decide to abort other, still running, sub-systems.

1) **Trigger(2) abort**

Trigger(2) abort is decided when the first negative result from any sub-task may be evaluated by the Main Controller, and no accidental has been detected. The trigger system of the experiment is then signalled the Trigger(2) Abort decision from the neutral EM part, and other sub-machine tasks must be aborted and prepared for the next Trigger(1). Abort criterions are very simple:

1- Abort if (number_of_clusters > 4)

2- Abort if \((\Sigma E_i X_i > e_1) \text{ or } (\Sigma E_i Y_i > e_1)\)

3- Abort if not \(( (D^2 \cdot m_k^2) - \varepsilon_2 < \Sigma E_i \cdot (\Sigma E_i X_i^2 + \Sigma E_i Y_i^2) < (D^2 \cdot m_k^2) + \varepsilon_2 )\)

2) **Trigger(2) ok**

Trigger(2) ok Neutral EM part is decided when all sub-machines have signalled
completion to the Main Controller, and all 3 task tests were successful. The trigger system of the experiment is signalled the ok from the EM part, and may obtain event information from the ASP system.

3.2.1.8 Simulations

We have started simulations to adjust the algorithms, understand the effects of ADC accuracy and conversion steps, time resolution and thresholding, and predict time processing distributions in various configurations. Monte Carlo data from the EM calorimeter, in the NA31' experimental environment, are available for this study.

1) Validation of approximations

Mathematical expressions for the second moment were modified to be faster. Residual errors from these approximations must be computed from a batch of Monte-Carlo EM data, in order to fit the $e_2$ value.

2) Cluster counting

Accurate simulations out of MC events are needed to trade 2D (speed, spatial resolution, but cost and complexity) versus 1D (correlation ambiguities but good accuracy, simplicity).

3) Cluster visualisation

The NA31' Trigger(2) tasks variations are coded for the ASP simulator. Array data for various cluster shapes, arrangement and energies are required as input to this simulator. A specific Sunview-based, interactive programme was written to generate and display manual, random, or Monte Carlo clusters in a graphical form. Effects of various ADC widths, energy conversion, and thresholding may be seen on the detector image. The accuracy of the computations is dynamically calculated to see the impact of these parameters. Resulting data may be filed to be further processed by the ASP/DSP simulators.

3.2.1.9 Conclusions

Our current studies show that state-of-the-art, programmable ASP and DSP processors can be used in the NA31' fast neutral trigger system.
Several synchronous load, parallel execution sub-machines run trigger sub-tasks concurrently under control of a main controller. Each sub-machine is fine-tuned to reach the best performance and accuracy level in the given NA31’ framework.

The highly competitive DSP market, and comprehensive ASP support in our community will both guarantee optimal system support and evolution. The current performance level of the processing devices used in the system is currently sufficient to fulfil the NA31’ requirements. Future compatible devices will certainly do more in the same framework.
3.2.2 U. of GENEVA - Heavy ions collisions, luminous chamber for tracking with CCD on-line camera. Tracks Reconstruction in Heavy Ion Collisions

E. Durieux, M.Izycki, M. Martin

Our group at the University of Geneva is going to apply ASP in a heavy ion experiment (WA93) to study Bose-Einstein correlations among charged pions. The reconstruction of the pion momenta with a large acceptance will be done with a tracking system consisting of a spectrometer magnet and new type of light emitting multistep avalanche chambers. Both technical and physical details of the experiment were discussed on the MPPC meetings [1] and can be found in the proposal [2].

Computational tasks in our experiment can be grouped in 4 main conceptual steps:

1. Image preprocessing
2. Image processing
3. Tracking
4. Momenta correlation

The first step is a classical background reduction and subtraction process which aims at finding the most significant pixels (about 1%) in 1 Megapixel image.

In the second step the clean image is corrected for the optical distortions (including grey level interpolation) and cluster analysis is performed to extract impact points of tracks.

The third step consist of two tasks: finding straight tracks from 4 chamber planes and a pointlike target and then calculating the momentum of the track.

Finally the Bose-Einstein correlation requires computation of four-momentum difference

\[ Q^2 = (p_1 - p_j)^2 \]

for all pairs of tracks i.e. \( N_{track}^2 / 2 \times 8 \) floating point operations.

It was estimated that 3 days of running, producing about \( 10^6 \) events, will need 50 GFLOP for S-Au data and about 5TFLOP for the Pb-Pb experiment planned in 1993.

The common feature of all these steps is that they contain a great degree of natural parallelism.
Application of the MPPC is quite obvious and does not pose any special technical requirements, except the online task of image preprocessing. In this case the data flow of 8 images of 110 kilopixel each every 20 msec would require a buffer memory and 4 or 8 parallel I/O channels.

As the proposal was accepted only in November 1990 our activity in the MPPC project were limited to training and feasibility studies. M. Izycki was participating in the ASPEX course at EPFL in May/June 1990. A chamber prototype was successfully tested in the beam in August of this year. Data collected during this test provided an excellent sample to study our cluster finding problems. The MPPC project was formally presented by M. Izycki in the WA93 collaboration meeting 1-3 October 1990.

Recently Eric Durieux, a physicist from our group, have joint our MPPC team, which consist now of 3 physicists working part time on the project. We will also have it the near future a new collaborator from Hungary for a period of 6 months, who will be working full time on this project.

In January 1991 we plan to start writing procedures for the image processing and track finding on the VASP simulator.

References

[1] MPPC CERN meeting 26-27 September 1990,
    CERN/ECP/MPPC 90-8, F. Rohrbach - 1st October 1990
    and Charged Particles
3.3 Non high energy physics applications

3.3.1 EPFL Image sequence coding, data compaction for HDTV

M. Kunt, A. Basso, F. Dufaux, T. Reed

3.3.1.1 Introduction

At the signal processing laboratory (LTS) in EPFL, the image sequence coding group has developed number of techniques for image sequence compression reaching very high compression ratios.

Nevertheless, the limit of these different techniques is the resource consumption in terms of computational time. This constraint is not compatible with practical applications, such as HDTV or videotelephony, which require real time processing.

Because these algorithms are highly parallel in nature, a parallel computational approach may be a solution to this limitation.

Among the different parallel machines, we have chosen ASP. The salient features of this machine are: capability of massively parallel computation thanks to the high number of processors available (65536), and a potential for very fast input/output data exchange. The relatively low price, small physical dimensions and low power dissipation of the ASP machine are characteristics that can be interesting for future applications, such as the installation of this machine inside a television set.

3.3.1.2 Activities

There are three people involved in this project: Todd Reed (group leader), Andrea Basso and Frédéric Dufaux.
Our current research consists of two projects:

The first one is the parallel implementation of the serial pyramidal Gabor (see ref.1 for Gabor expansion and ref.2 for introduction to image processing techniques) compression algorithm developed in the LTS. We are studying a parallel implementation on the ASP of this technique, and the feasibility of real time processing. Our codes are currently developed on the Aspex simulator.

In the Gabor compression algorithm, the image is decomposed in elementary Gabor functions basis. The use of such a decomposition is motivated by the fact that Gabor functions have optimal localisation in both spatial and frequency domains (see ref.3). We can formulate it in matrix form:

\[ I = G X G^T \quad (I: \text{image}, \ G: \text{Gabor matrix}, \ X: \text{coefficients}) \]

A system for coding would be like this:

Fig.1 – System for coding

The coefficients \( X \) and the reconstructed image \( I \) are then obtained by matrix multiplication.

---

Ref.2 - Anil K. Jain, Fundamentals of digital image processing, Prentice Hall.
The realisation of this algorithm involves the study of parallel efficient algorithms for matrix computation. Our interests are in multiplication of large matrices, taking into account their special structures.

Matrix multiplication on ASP:

We have so developed several algorithms for matrix multiplication on ASP.

A second step will be to compute also the matrix A (which is the pseudo-inverse of G) on ASP to allow adaptative Gabor functions.

We can make these general remarks:

- If \( n \) is the dimension of a matrix supposed square for simplicity, we need \( n^3 \) multiplications and additions.

- Typically \( n=256 \) for an image, so with \( n^2 = 65536 \) processors, we need then:

- \( n \) multiplications & additions with parallelism \( n^2 \).

- \( n \) operations write or move to update the data.

Finally, we have obtained the best results with the following algorithm called outer product:

Let say we have two matrices A and B, and we want to compute \( A^*B \):

\[
A = (a_{ij}) \quad B = (b_{ij})
\]

We need \( n \) iterations. At the \( k^{th} \) iteration, we write the \( k^{th} \) column of A and the \( k^{th} \) row of B in all the processors \((k=1..n)\). We have then only to multiply and accumulate.

Here is an example of the first iteration in the case of 3 by 3 matrices:
At this time, two programmes for integer computation have been developed, one using 8 bits and one 16 bits. The next step is the evaluation of floating point performances when the floating point libraries will be available.

The second approach considers parallel implementation on the ASP of artificial neural networks for image compression. There are three different techniques employed for image compression: the Cottrel-Munro-Zipser technique (see ref.4) that uses a backpropagation net; vector quantization based on counterpropagation nets, and Gabor coefficient computation with a relaxation feedback network. At the moment we are evaluating these techniques. We have started with the first one.

The first technique uses a backpropagation network with fewer hidden units than input and output units. The goal is to learn the identity mapping for some arbitrary input data environment. The network is forced to discover a set of good hidden layer weight vectors for representing the input data. The output layer uses these features outputs (the outputs of the hidden layer units) to attempt to reconstruct the input pattern.

---

Ref.4 - G.W. Cottrel, P. Munro, D. Zipser, Image compression by back propagation: an example of extensional programming, ICS report 8702, Feb. 1987, UCSD.
Fig. 3 Cottrel-Munro-Zipser scheme for image compression

Training

For each image blocks of pixels are entered into the network starting at the upper left corner of the image and moving the block to the right one pixel at time.

The training processing is continued until the network mean square error seems to stabilise at a reasonably low value.

Operational use

Blocks are chosen so that they do not overlap and so that they tile the image completely.

As each block is entered the corresponding quantised outputs from the hidden layer units are transmitted and then decoded on the receiving end.
3.3.1.3 Results

The initial results obtained with the first approach are very encouraging.

We have three graphics (fig.4, 5, 6) describing the outer product algorithm performances. The first one (fig.4) shows the time obtained with the ASP simulator in function of the matrix dimension for the 8 bits version and the theoretical time obtained by studying the algorithm complexity. We see a very good concordance between the two curves. The second one (fig.5) is the speed up (= time with one proc. / time with \( n^2 \) proc.) for the 8 and 16 bits version. The last one (fig.6) shows the efficiency (= speed up / nb proc.). The speed up is very high but the efficiency is low for large matrix dimension.

This is due to the string connections which are not the best for matrix multiplication. However the time for 256 by 256 matrix multiplication is quite small.

![Matrix dimension dependency](image)

**Fig.4 - Matrix dimension dependency**
Fig. 5 - Speed up

Fig. 6 - Efficiency
We have then seven images:

- The first one (fig.7) is the original image 64 by 64. The relatively bad quality is due only to the small size of the image (the simulator is too slow to be able to compute 256 by 256 images).

- We have then two sets of three images (fig. 8.1, 8.2, 8.3 & 9.1, 9.2, 9.3), showing the image after expansion and reconstruction for a compression ratio of roughly 10 for two different Gabor basis. The two Gabor basis are elementary Gabor functions with two different spatial partitions. We have two images computed with floating points on a serial machine (fig. 8.1, 9.1), two with 16 bits (fig. 8.2, 9.2) and two with 8 bits (fig. 8.3, 9.3) computed on the ASP simulator.

The 8 bits results are not so good, in particular for the first step where coefficients have large value. But we don't see any difference between the floating points and the 16 bits, and the quality of the image is quite good (in respect with the size effect). This result is quite encouraging. The computation time given by the simulator is about 15 ms and 30 ms for the 8 bits and 16 bits version respectively, in comparison to several seconds on a work-station for the floating point case.

The complexity study let us know that we are able to compress a 256 by 256 image on ASP, up to a ratio of 200:1, with this algorithm in about 100 ms. This is very close to real time processing (25 images per second). We can expect to reach this limit by taking into account the structure of the matrix.
fig. 7 – Original image 64 by 64

fig. 8.1 – Serial reconstruction with floating points (Gabor basis 1)

fig. 8.2 – Asp reconstruction 16 bits (Gabor basis 1)

fig. 8.3 – Asp reconstruction 8 bits (Gabor basis 1)
fig. 9.1 – Serial reconstruction with floating points (Gabor basis 2)

fig. 9.2 – Asp reconstruction 16 bits (Gabor basis 2)

fig. 9.3 – Asp reconstruction 8 bits (Gabor basis 2)
Neural network

As far as the neural network approach is concerned, software for serial machine has been written to be used as reference for further development. Two experiments have been carried out to evaluate the learning performance of identical mapping of a backpropagation net. In the first experiment we have used a network with 64 neurons in the input layer, 16 in the hidden one and 64 in the output layer. The compression ratio obtained is 4. The first image is the original Lena 256x256 pixels (8 bit for pixel) (fig.10.1). The second image has been obtained with a learning of 100 cycles for each block (fig. 10.2)

In the second experiment we have used a network with 64 neurons in the input layer, 8 in the hidden one and 64 in the output layer. The compression ratio in this case is thus 8. Fig 11.1 is the original 64x64 Lena. The second image (fig 11.2) is the reconstructed Lena (the one used for the training) after only 5 cycles for block learning. The third (fig. 11.3) is the result of 20 cycles for block learning.

Comments

The leaning seems to perform well after few cycles of learning (fig 11.2). Due to the slow convergence time of backpropagation the perfect reconstruction of the image requires many cycles of learning (fig. 10.2).

Very soon a parallel version on ASP simulator of the different nets will be available for performance evaluation.
fig. 10.1 – Original image 256 by 256

fig. 10.2 – Reconstruction after 100 cycles learning
fig. 11. 1 – Original image 64 by 64

fig. 11. 2 – Reconstruction after 5 cycles

fig. 11. 3 – Reconstruction after 20 cycles
MPPC Documentation list (Proposal, memorandums, notes and papers):


[2] TRAX-1 proposal, Proposal to develop a physics image processing workstation for Megabytes per event data based on fine and medium grained parallel processors, A. Sandoval et al., CERN, 27 April 1988.


[7] Minutes of the first meeting of the MPPC project held on Friday 27 April 1990 at CERN, CERN/ECP/MPPC 90-1, Minutes 1, 2 May 1990.

[8] Minutes of the second meeting of the MPPC project held on Tuesday 29 May 1990 at Saclay, CERN/ECP/MPPC 90-2, Minutes 2, 7 June 1990.


[10]* Memorandum, 6-7 September Paris Meeting conclusions. MPPC 90-6, 5 September 1990.


[12]* Short Summary and conclusions of the 20 November 1990 Saclay Meeting, MPPC 90-10, 26 November 1990


[14]* Financial structure of the MPPC project, MPPC 90-7, 17 September 1990.


[16] Study of muon triggers and momentum reconstruction in a strong magnetic field for a muon detector at LHC, CERN/DRDC/90-36, DRDC/P7, 30 August, 1990.

Note: [ ]* submitted to restricted distribution.