AN ORIGINAL APPROACH TO DATA ACQUISITION : CHADAC

HUPPERT-NAYMAN-RIVOAL- LPNHE-PARIS (FRANCE).

ABSTRACT

Many labs try to boost existing data acquisition systems by inserting high performance intelligent devices in the important nodes of the system's structure. This strategy finds its limits in the system's architecture. The CHADAC project proposes a simple and efficient solution to this problem, using a multiprocessor modular architecture. CHADAC main features are :

a- Parallel acquisition of data : CHADAC is fast ;it dedicates one processor per branch ; each processor can read and store one 16 bit word in 800 ns.

b- Original structure : each processor can work in its own private memory, in its own shared memory (double access) and in the shared memroy of any other processor (this feature being particulary useful to avoid wasteful data transfers). Simple and fast communications between processors are also provided by local DMA'S.

c-Flexibility : each processor is autonomous and may be used as an independent acquisition system for a branch, by connecting local peripherals to it. Adjunction of fast trigger logic is possible.

By its architecture and performances, CHADAC is designed to provide a good support for local intelligent devices and transfer operators developped elsewhere,providing a way to implement systems well fitted to various types of data acquisition.

INTRODUCTION

In a data acquisition system the computer workload consists of primary tasks (data readout, event filtering,apparatus control...) and secondary tasks (computations, displays, mass-memory management....).

The continuous trend of growth in detector size of high-energy physics leads to an increasing volume of data, thence primary workload for the acquisition system computer. The acquisition process is slowed down, and primary tasks tend to eliminate the availability of useful secondary ones.

To cope with that situation the widely used strategy is to limit as much as possible the data flow, using specialized more intelligent devices inside the acquisition branches (for instance formating or filtering processors inside CAMAC or ROMULUS branches).

Such a strategy finds its limits in the system architecture: the computer sitting at the root acquisition system tree structure (which tends to be very big) still has to read sequentially every branch and has to manage the whole acquisition.

One elegant and efficient way out of this bottleneck is to decentralize the primary tasks and some of the secondary ones in a simple but high performance multiprocessor structure.

CHADAC is an example of such a structure.

In CHADAC the tree structure is split into independent

branches. On the first hand, each branch is managed by an autonomous processor : global acquisition can thus proceed at high speed in parallel (1.25 M. 16 bit-words per sec per branch ready into memory).

On the other hand, an original architecture provides very optimized links among processors (via double access memory) and enables each processor to use general (shared) ressources or peripherals as well as local (private) ones.

As an end result the system can deliver very compacted data structures (such as filtered, formated, pre-treated and possibly inter-branch correlated) to some computer via high-speed channel (DMA).

From the software point of view CHADAC is basically a multijob real time system. Several levels of software are provided : real-time multitask monitor, debugger, file handling system, shared ressources and communications management, high-level language,editor,etc. Each processor has autonomous development capability for application software development.

II- DESCRIPTION OF CHADAC

1- General structure

The CHADAC system is modular. It is built around a bus of 16 boards capacity, the general bus, which supports basic units called P_i.

Each P_i unit is also built around a bus of 8 boards capacity, called the private or local bus. Processor i get access to the private ressources of unit P_i, such as Memory, I/O, interfaces..., via the local bus. It can also get access to general or shared ressources (memory,I.O, peripherals,....) via the general bus.

Memory sitting at the P_i level is of different kinds : private memory (up to 384 K bytes) called MP_i and shareable secondary memory called MS_i.

2- Inter-processor exchange architecture

The architecture of CHADAC is designed to optimize inter-processor communications. The basic idea is to use double-access memory as letter-box buffers. Such a shareable memory, called secondary memory or MS_i is associated to each P_i. The processor of each P_i can get direct access to its MS_i, viewed as a local ressource, as well as access via the general bus. It also can get access via the genral bus to the MS_j of any other P_j. In that case its MS_i and all of the MS_j are considered general ressources.

Since MS_i is double-access (dual port) access conflicts arise. This is solved by defining direct access (via local bus) as the highest priority access.

From the user's viewpoint at the P_i level the set of all MS_i including its own looks like a unique continuous address space. The phrase "including its own" is true as long as the processor i addresses its MS_i via the general bus. If however, it is addressing its MS_i via its local bus, address space of its MS_i starts at 0, whatever the P_i.

User software may thus be distributed over the whole set of MS_i. Any piece of reentrant software may be executed simultaneously by several processors, avoiding dupplication of code.

Data need not to be transfered from unit P_i to unit P_j. Unit P_j can work upon data in MS_i directly, avoiding wasteful data transfers.

3- Priority management on the general bus

Since the general bus has a multiprocessor structure, it is necessary to provide access priority management in order to ensure equal servicing to each processor.

The ALLOC board, dynamic allocator of priority, is devoted to this task.

Any processor i wanting to gain access to the general bus emits request Req_i to ALLOC which determines the selected candidate SEL_i in about 50 ns.

Three allocation strategies are available with ALLOC, and the strategy is programmable.

1- fixed priority defined by switches S_1,S_2,S_3.

2- fixed priority defined by 3-bits code loaded into ALLOC; Both of these strategies implement geographic priorities.
Only the highest priority P_i is defined, the order of priority allocation is fixed and predetermined(priority of P_i > priority of P_{i+1})

3- dynamic priority allocation using a microprogrammed algorithm.
If P_i is being serviced then P_{i+1} get highest priority and P_i lowest.

4- Standards

For simplicity and ease of implementation the standard for the buses which has been chosen is Intel's MULTIBUS. This choice gives access to a wide variety of ready-to-use boards proposed by constructors.

5- Description of a P_i unit

A P_i unit is built around a bus of 8 slots : the private bus. Into the private bus are plugged some general boards and some specific boards, as necessity arises.

- the processor board : it is built around a 16 bits microprocessor (INTEL 8086) and has access to the general bus and to the private bus. It houses 6 programmable real time clock channels, 2 serial I/O channels (USART), 1 programmable interrupt controller (PIC),8 Kx8 bits PROM.

Three switches (SW_0,SW_1,SW_2) on the board define the size of the memory the processor can directly address as a local ressources(MP_i+MS_i). Any address greater than this limit is considered general memory accessible via the general bus.

SW_0	SW_1	SW_2	MP_i +MS_i
0	0	0	OFF
0	0	1	32 K
0	1	0	64 K
0	1	1	128 K
1	0	0	192 K
1	0	1	256 K
1	1	0	320 K
1	1	1	384 K

- the I/O processor board (IOP) : it is built around the I/O processor INTEL 8089 and is devoted to fast data acquisition. The processor supports 2 fast intelligent DMA , it can read and store into memory a 16-bit word in 800 ns. The board can also be used to make memory to memory transfers (for instance MP_i to MS_i).

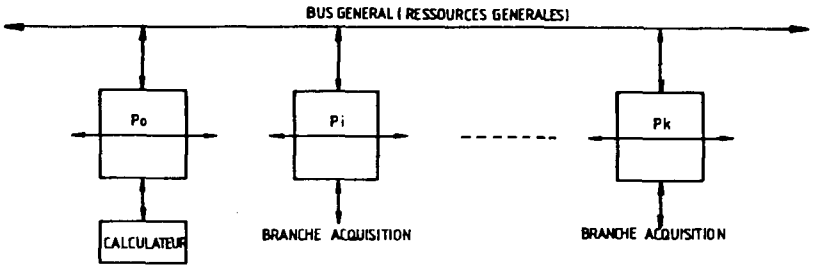
- the decoder board : decodes addresses on the private bus to grant access to secondary memory, or to a special IOP dedicated PROM. This 4Kx8 bits Prom can be used to store resident programs for the I/O processor.

- the memory boards : one or more memory boards can be used. This memory is the P_i private memory.(Any MULTIBUS compatible memory board is usable).

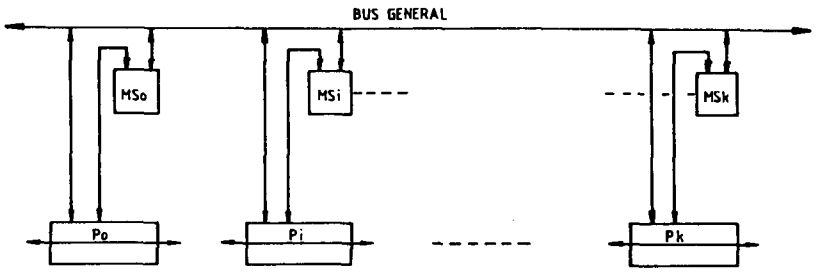
III- CONCLUSION

The CHADAC system is entirely modular, both at the general and local levels. A single branch acquisition system can be fitted with a single processor playing the role of a true autonomous control unit with a variety of peripherals. The system can be extended up to 7 branches. CHADAC is thus well adapted to small as well as large scale data acquisition.

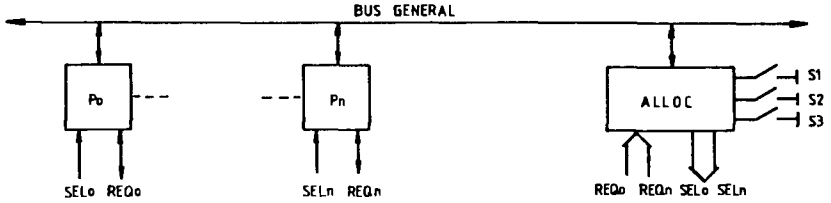
CHADAC is built out of monolithic 16 bit-processors. It necessitates only a small amount of circuits whose information is simple leading to reasonably low cost.



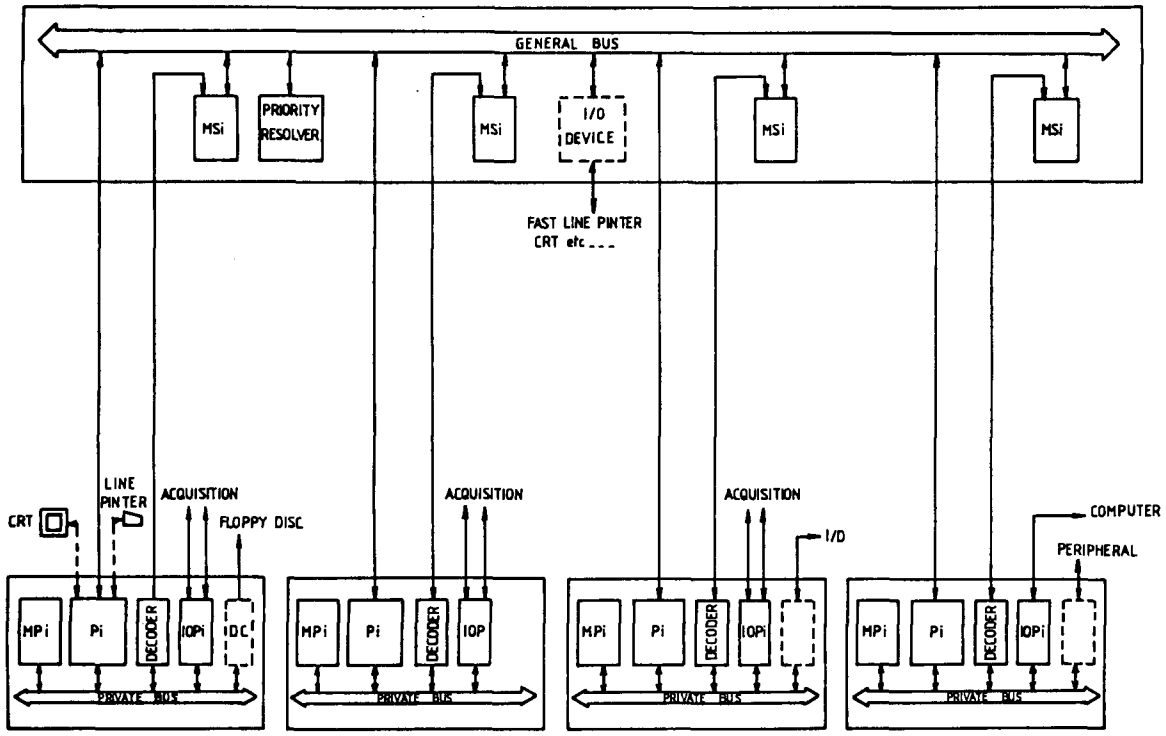
①



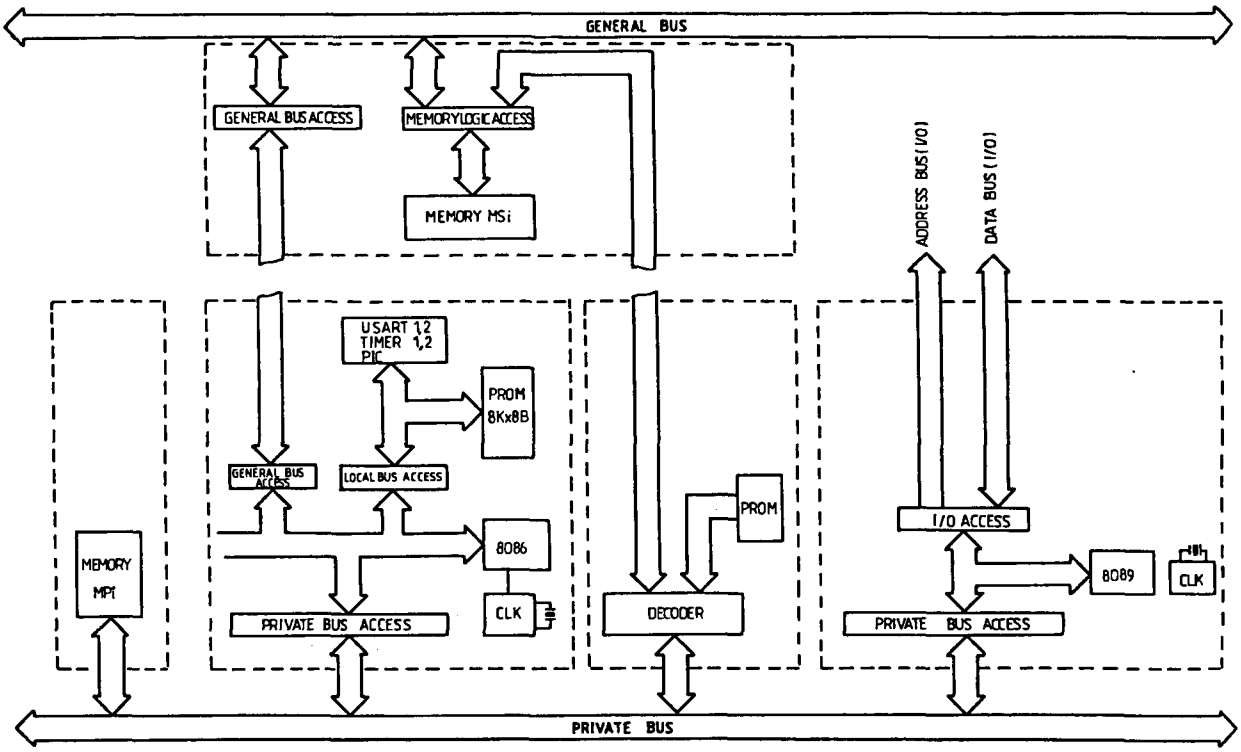
②



③



CHADAC : MULTI Pi ORGANIZATION



CHADAC : Pi ORGANIZATION