NINTH EUROPEAN ROTORCRAFT FORUM

Paper No. 41

A COMPARATIVE ANALYSIS OF PROCESSING STRUCTURES FOR AIRBORNE SYSTEMS

G.A. SCOTTI DI UCCIO

SELENIA S.p.A.
(POMEZIA)

September 13-15, 1983
STRESA, ITALY

ASSOCIAZIONE INDUSTRIE AEROSPAZIALI
ASSOCIAZIONE ITALIANA DI AERONAUTICA ED ASTRONAUTICA

A COMPARATIVE ANALYSIS OF PROCESSING STRUCTURES FOR AIRBORNE SYSTEMS

G.A. SCOTTI DI UCCIO — SELENIA, Industrie Elettroniche Associate, S.p.A.

1. INTRODUCTION

This paper presents an overview of the architecture configuration for the avionic systems. It will first summarize the architecture evolution in the last twenty years and will despict the next generation concept of architecture; then an outline of the family of alternatives is given and some of them will be described in more detail to show advantages and drawbacks.

2. AVIONIC ARCHITECTURE EVOLUTION

Table 1 identifies the key technical characteristics expected to be the foundation for future A/C avionic Systems.

1940-1960	1960-1980	1980-2000
— WIRED PROGRAMS	— STORED COMPUTER PROGRAM	— DISTRIBUTED HIERARCHICAL STORED PROGRAMS
— DEDICATED ANALOG PROC.	— CENTRAL DIGITAL PROCESS	— REDUNDANT CENTRAL PROCESSING
— INDEPENDENT SUB-SYSTEMS — INDEPENDENT DISPLAY	— COMMUNICATIONS THROUGH I/O INTEGRATION AND CENTRAL PROCESSOR STORED PROGRAMS	— DISTRIBUTED DEDICATED FUNCTIONAL PROCESSORS
— INTEGRATION THROUGH OPERATOR	— CENTRALIZED DISPLAY	— COMMUNICATION THROUGH BUS
— NO REDUNDANCY	— SOME REDUNDANCY	— MULTIPATH REDUNDANCY
— LIMITED FAULT DETECTION	— SOME DEGREE OF FAULT TOLERANCE	— FAULT TOLERANCE
— NO RECONFIGURATION	— NO DYNAMIC RECONFIGURATION	— DYNAMIC RECONFIGURATION
— DISCRETE HARDWARE	— MSI LSI HARDWARE	— VHSIC HARDWARE

TABLE 1 — EVOLUTION OF AVIONICS

The first type of «architecture» was circuscribed over a set of indipendent subsystems with very limited capability of data interrelationship between equipment. The technology trend was based on servo's and synchro's, operational amplifiers, direct coupled sensors.

No system automatic fault detection was provided and no integration was really possible.

Even if the unit reliability was quite good, since the system is at a very low level of complexity, the total system performance was not very high.

By the early '60 the need for an board equipment flexibility lead to the introduction of general purpose digital computers into a variety of A/C and avionic systems. The programmability of these machines permitted operational and technical changes via S/W modifications. The IC technique hastened the GP computers introduction because of weight and volume savings.

The first generation of airborne computers were termed «centralized»: i.e. all Operational Flight Programs were contained in a single machine. As consequence, the 1960-1980 airborne systems were based on a mix of digital and analog technology where the architecture is generally centralized.

The «centralization» allows a good automatic data exchange between different equipments; so some capability of integration is possible. The use of IC's reduces dimensions, weight and power consumption. The increase of computation capability improves the system performance but is still limited by power of computers and the size of the available memories.

As the solid-state techniques progressed, the phisical characteristics of the on-board computers lead to the phisical incorporation into various on board subsystems. Thus the term «embedded computer» came out. Eventually these machines were connected together in what is now defined a «federation of resources».

For the future applications improvements in electronic technologies and a more general use of digital equipment will brings to new solutions for a better integration of the avionic system. The intrinsic redundancy of the equipment, in conjunction with the concept of BITE, will allow dynamic reconfiguration both at unit and system level. The overall system operational availability will be certainly better than in the past.

On the other hand, the continuation of proliferation of hardware in absence of a standard, the increasing of speed at which new component were beeing invented led, by early '80, to the establishment of common standard for computer hardware and related Higher Order Languages.

Now it is possible to say that there are still many questions to be answered relative to computer and avionic systems architecture and language standard, but it is also possible to say that the trend for a common standard introduction is going on; so we can see the future with optimism.

2. SYSTEM ARCHITECTURAL STRUCTURES

Table 2 summarize the top-down approach of decomposition of avionic systems structure form an engineering design point of view. The equivalent of the system mission for the A/C is the total A/C avionic systems which contains *the Operational functions performed by the avionic systems (equivalent to A/C definition: fighter, attack,* Airborne Early Warning, Electronic Warfare etc.). The System mission or the total A/C avionic system determines the types, functions, performances of the A/C avionic equipments.

1 — TOTAL AIRCRAFT/AVIONIC SYSTEM
2 — PARTITIONING OF AIRCRAFT/AVIONIC SUBSYSTEMS
3 — INTERCONNECT BUS STRUCTURE
4 — SYSTEM-WIDE PROCESSING ARCHITECTURE
5 — SUBSYSTEM DEFINITION
6 — COMPUTER SYSTEM
7 — EQUIPMENT DEFINITION

TABLE 2 — TOP-DOWN APPROACH OF AVIONIC «DECOMPOSITION»

These equipments must be divided into several groups according to the subsystem in which they are included. For example, the «Vehicle Group» shall contain flight control equipments, pilot's display, electrical generators, etc.; the «Core Group» shall contain navigation, communication and computational resources; the «Mission Group» shall contain specific radars, sensors, EW equipments etc; the «Weapon Group» shall contain stores management equipment.

How to connect all these equipments in the total aircraft/avionic system? Fig. 1 is the «family tree» of the various Processing Architecture alternatives available to the system designers.

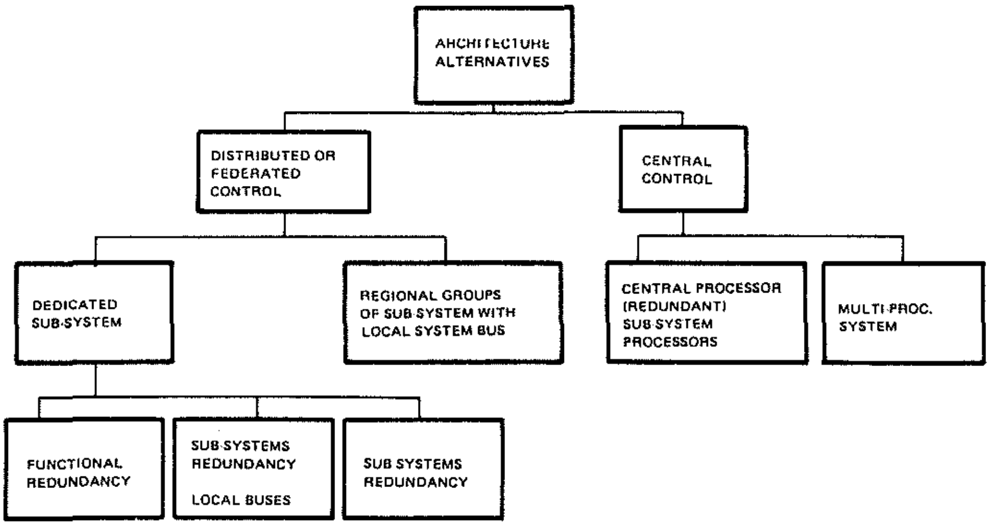

Fig. 1 — ARCHITECTURE ALTERNATIVES

2.2 *CENTRALIZED ARCHITECTURE*

Fig. 2 outlines the configuration of the centralized architecture.

The main characteristics for the hardware for the hardware are:

— A powerful central processor (which can be redundant) should perform all tasks.

— I/O capabilities of the central unit should be very powerful and could be a limitation for the system performance.

— Each peripheral must be connected with a dedicated interface according to its type.

The main characteristics for the software are:

- Executive is resident in the central unit, it shall allow real-time multi-task processing.
- The operational program is unique and covers all the system functions.

Critical areas for growth capability are:

- Processing speed
- Memory size
- Input/output lines.

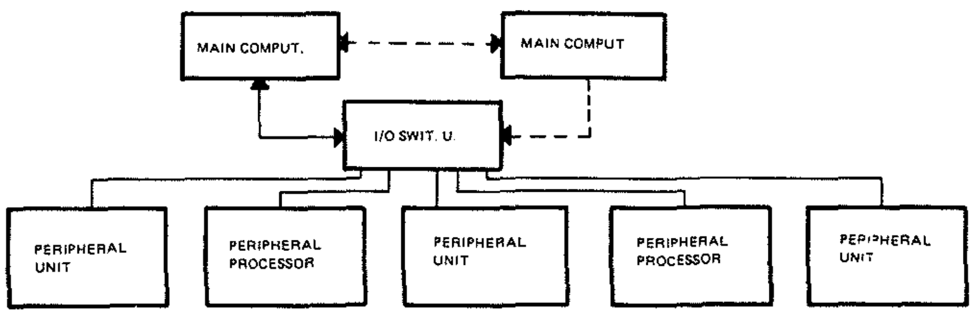

Fig. 2 — CENTRALIZED ARCHITECTURE

2.3 *FEDERATED ARCHITECTURE*

Fig. 3 outlines the configuration of the federated architecture.

The main hardware characteristics are:

- The hardware is tailored for each function.
- Interrelationship between systems are done through a data bus which is treated as another peripheral.
- There is no reconfiguration capability and redundancy is performed at peripheral level.
- Only interfaces and communication protocols must be standardized.

The main characteristics for the software are:

- The executive could be different.
- The operational programs for each function will be indipendent.
- Only one unit will control the system.

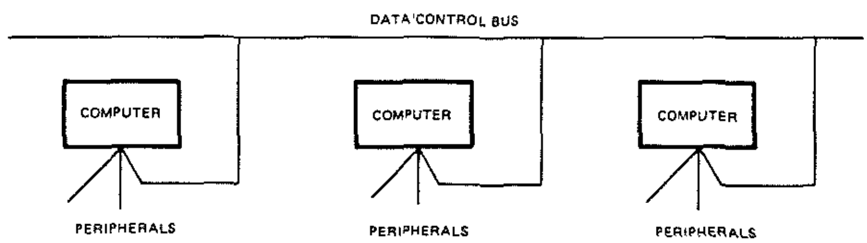

Fig. 3 — FEDERATED ARCHITECTURE

2.4 *DISTRIBUTED ARCHITECTURE*

Fig. 4 outlines the configuration of the distributed architecture.

The main hardware characteristics are:

- All computers have the same architecture.
- A high speed intercomputer bus is necessary.
- I/O with peripheral is done through a multipath bus.

The main software characteristics are:

- The total system control is distributed throughout local executives.
- Function programs are not dedicated to a specific computer.
- Software programming and testing is no complex.

Growth capability is possible by adding other computers. Reconfiguration in case of failure of one unit is possible.

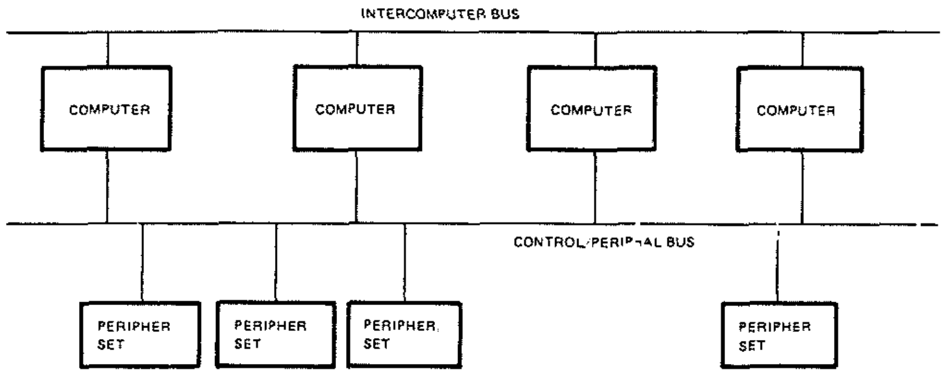

Fig. 4 — DISTRIBUTED ARCHITECTURE

2.5 *HIERARCHICAL ARCHITECTURE*

Fig. 5 outlines the configuration of the hearchical architecture.

The main hardware characteristics are:

— Computers could be the same type.

— Processing speed and memory capability for each computer could be chosen according to the needs.

— Local buses could have a low speed.

The main software characteristics are:

— At local buses level, same possibilities as for federated architecture could be applied with global

— At local buses level, same possibilities as for federated architecture cold be applied with global bus software acting as executive.

The system reconfiguration in case of failure is possible, but requires a high speed intercomputer global bus.

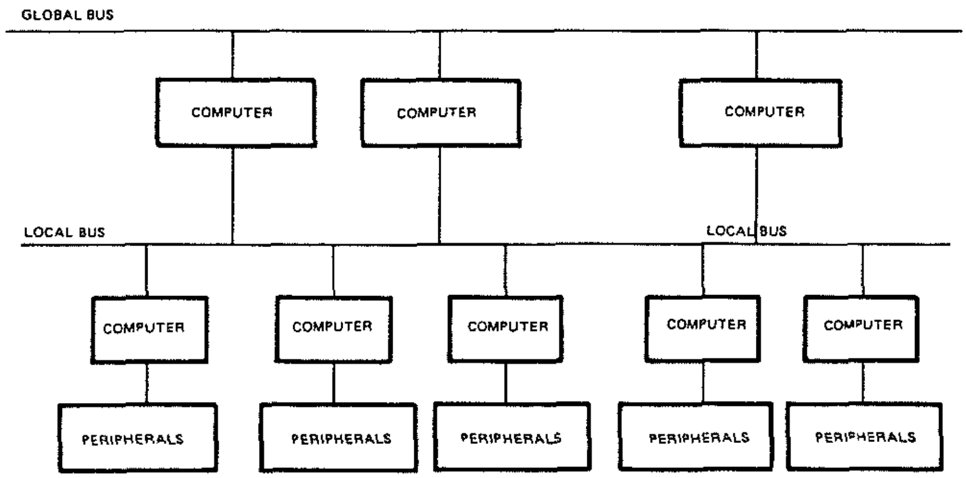

Fig. 5 — HIERARCHICAL ARCHITECTURE

3. DESCRIPTION OF POSSIBLE SOLUTIONS

Making reference to the definitions already given, it is possible to list some of the architectural options available for consideration by system designers.

Hereafter the following configurations are taken into account:

- Full functional redundancy.
- Full functional redundancy plus dedicated subsystems.
- Maximum physical redundancy.
- Full functional redundancy within local group of subsystems.
- Centralized.
- Multiprocessors.

These options offer the payload system designer the capability to maximize those characteristics which are of major importance to the particular application required by the aircraft mission definition.

3.1 FULL FUNCTIONAL REDUNDANCY

Fig. 6 outlines the structure of a system where the redundancy is total. Each processor can manage any sensor according to the actual needs.

The hardware will be based on identical architecture giving the same level of performance and the processors are connected, both to the sensors through a serial data bus and to the other processors through a high speed data bus. So a full interchangeability within processor is possible, but this concept need the standardisation of processor memory and capabilities to the maximum extent.

The software will be organized in such a way that any application program could run in any processor. In case of failure of one processor, the reconfiguration of the system will be actuated by simple processes: in fact, only bus addresses will be changed.

As far as the execution of the tasks is concerned the software executive will be simple, but in order to maximize reliability a new set of routines for complex fault detection, isolation of errors and recovery function shall be provided.

The Input/Output, which will be standardize on all the equipment, will be a significant part of the system and will be certainly very expensive.

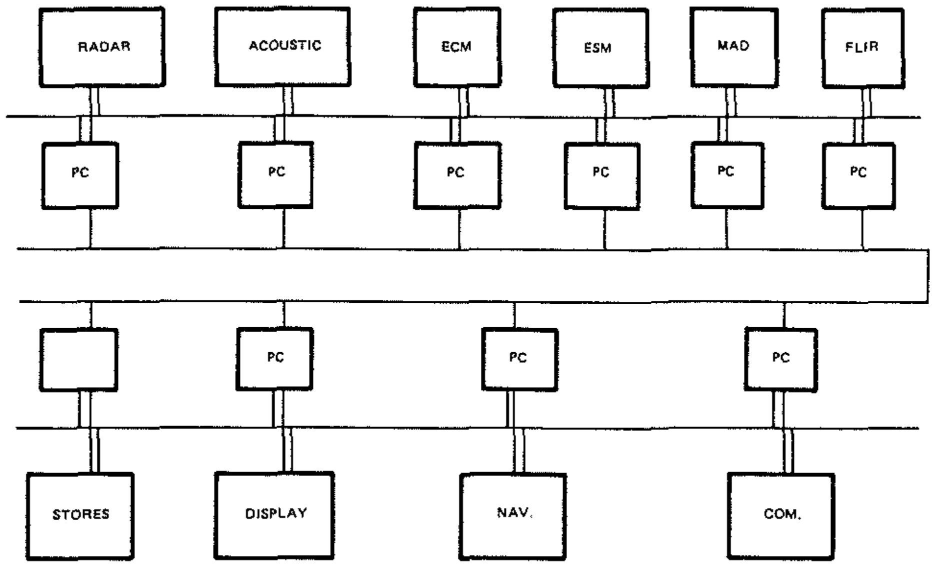

Fig. 6 — FULL FUNCTIONAL REDUNDANCY

3.2 FULL FUNCTIONAL REDUNDANCY PLUS DEDICATED SUBSYSTEMS

Fig. 7 outlines the structure of a system where the redundancy for each function is total. Each processor can manage any sensor of its subsystems according to the actual needs.

The hardware may be based on identical architecture but the level of performance must be optimized to the specific needs of each subsystems. As for the full functional redundancy structure, processors are connected both to the other processors through a global bus and to the sensors of the subsystem by a local bus. In this case there is no need to have a global bus as fast as for the previous solution.

The software will be organized in such a way that any application program of the subsystem could run in any processor of the same subsystem.

In case of failure of one processor, the involved sub-system can be reconfigured once if it has at least two processors.

The software executive will be simple and the fault detection will be easier than the configuration described in the previous paragraph, but the recovery is more problematic.

The Input/Output and bus technolofy can be tailored to the dedicated subsystem.

This configuration, even if not so clean as for the Full Functional Redundancy, is certainly easier to implement.

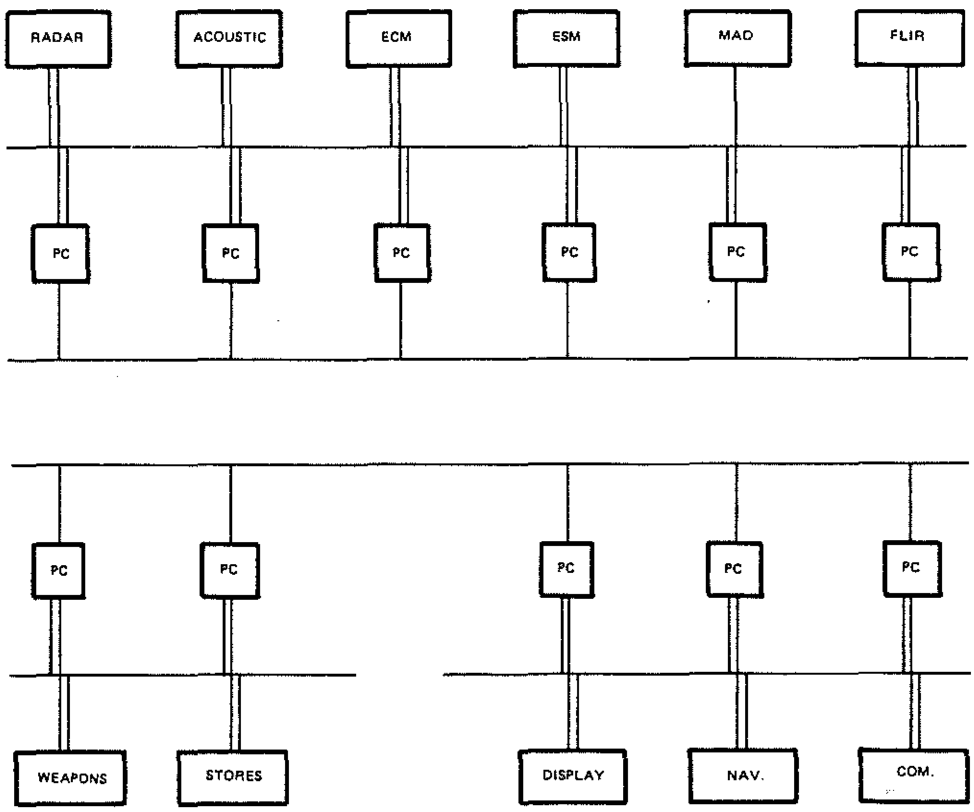

Fig. 7 — FULL FUNCTIONAL REDUNDANCY PLUS DEDICATED SUBSYSTEMS

3.3 *MAXIMUM PHYSICAL REDUNDANCY*

Fig. 8 outlines the structure of a system where the physical redundancy is maximized.

Each function is controlled by at least a couple of processors which will have the same architecture.

There is no need to have the same type of processor for all functions. However, if the choice of a unique type of cheap processor is feasable, then the system cost effectiveness will be improved even if the number of processors is increased.

The software development will be simple and easy to test.

Redundant processors will have unique software.

Sensors Input/Output is not required to be standardized.

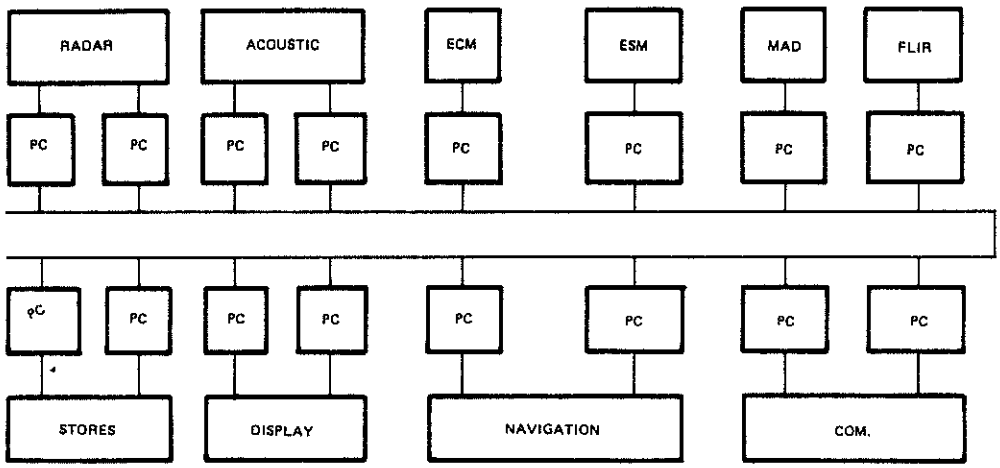

Fig. 8 — MAXIMUM PHYSICAL REDUNDANCY

3.4 *FULL FUNCTIONAL REDUNDANCY WITHIN LOCAL GROUP OF SUBSYSTEMS*

Fig. 9 outlines a typical configuration where full functional redundancy is limited to a local group of subsystems.

This configuration is used when the full functional redundancy becomes impratical due to different performance and technology of local buses.

The hardware is the same as for the full functional redundancy in a group but it is not necessary that the architecture and performance of the processors and the sensors (in terms of I/O) are the same.

The software can be organized as for the full functional redundancy, but, due to the different computer architecture and/or data buses, it can vary between local group.

The overall configuration is clean and feasible and allows a good definition of the specific functions and a good testing capability, but it is probably more expensive than the maximum physical redundancy in terms of total life cycle costs.

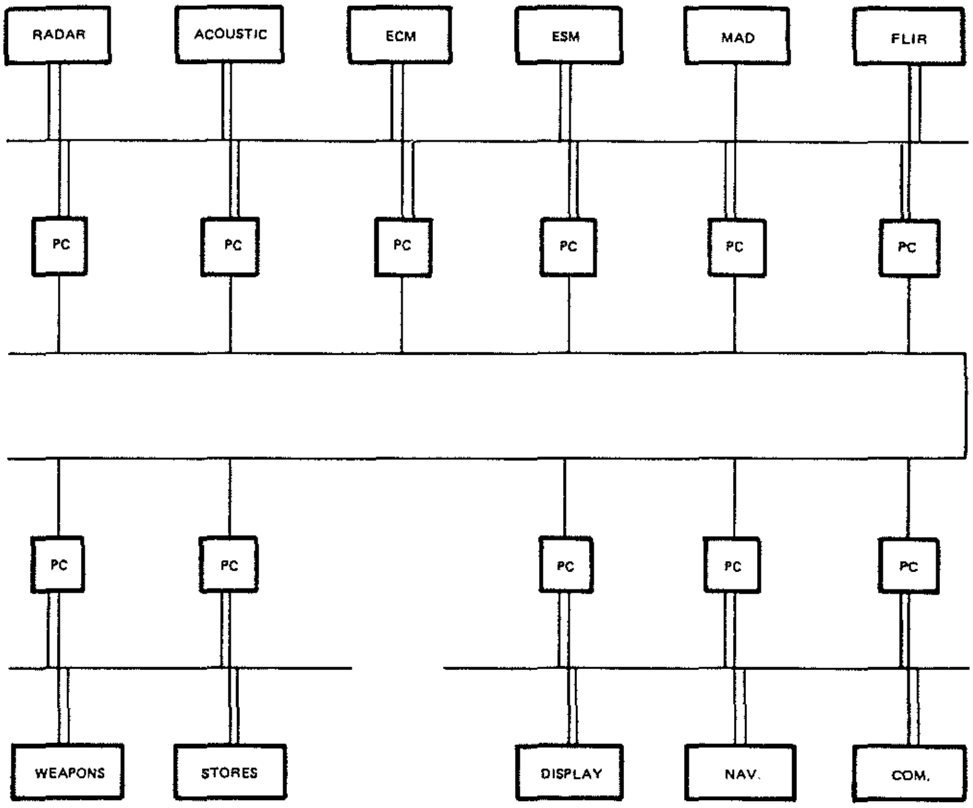

Fig. 9 — FULL FUNCTIONAL REDUNDANCY WITHIN LOCAL GROUP OF SUBSYSTEMS

3.5 *CENTRALIZED*

Fig. 10 outlines the classical configuration where the processing is completely centralized. Fig. 10 shows a version where two main computers can work together (one is on stand-by).

The hardware is mainly based on a very high performance main computer which, under the control of sophisticated executive, performs all tasks according to the mission.

The software is complex, but it is based on a classical concept of executive which is at present very well understood. It allows very simple fault detection and recovery routines implementation.

Input/Output hardware is very complex and will reflect the I/O different characteristics of the sensors: generally this is the least reliable part of the system.

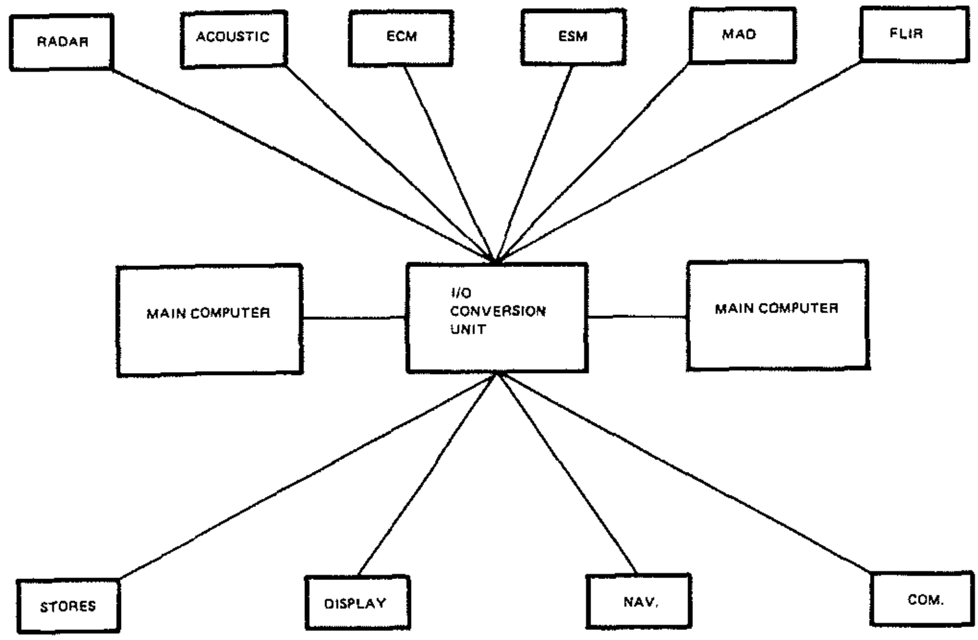

Fig. 10 — CENTRALIZED

3.6 *MULTIPROCESSOR*

Figure 11 outlines a typical configuration where processing is executed through a multiprocessor system.

In multiprocessing, the computer, or the computers, based on a standard high performance microprocessor, is designed to operate on a multiprocessing base, where I to N processors may be tightly coupled in the same structure called «NODE».

Memory and I/O resources are provided either locally at each processor and in common at node level.

As a consequence, the processing power of the system is modular and can be tailored to the operational needs of the application.

Generally the system is configured in such a way that at least one processor in each node is redundant to allow fault tolerance; all sensors are connected to the processing nodes through a high speed direct memory access data bus.

The multiprocessing configuration is the only one which permits dynamic task reconfiguration.

The software is fairly complex, but very well understood executive, together with the use of High Level Programming Languages and System Description Language (SDL) permits an easy configuration of operational software to match different availabilities of processing resources, memory and I/O.

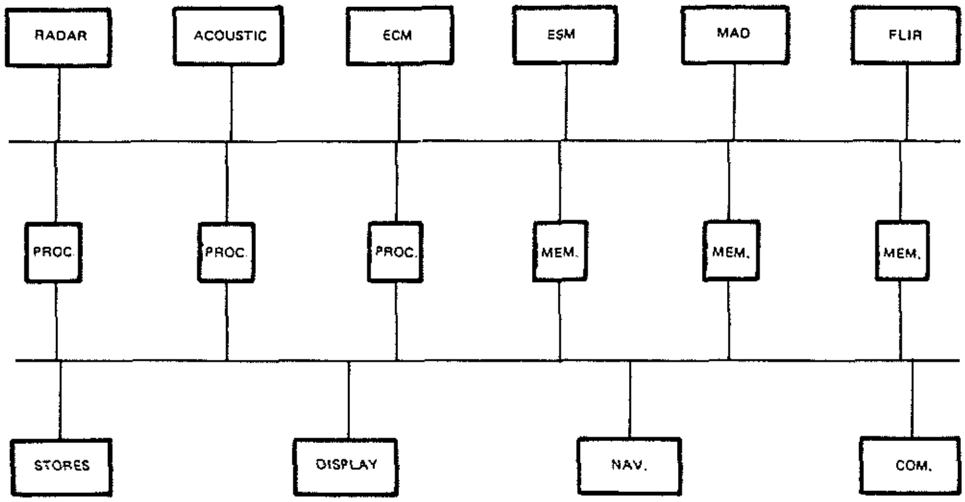

Fig. 11 — MULTIPROCESSOR

CONCLUSIONS

System processing architectures in the avionic systems have been limited in the past to the central control computer arrangement.

With the emergence of microprocessors, the avionic equipments evolved in computing power and the system architecture became more and more distributed.

Processing systems features now modularity and self adaptation to the external environment, critical functions can be isolated and duplicated, flight safety is increased, hight mission success is achieved with relatively low cost equipment; finally a large growth capability in becaming available, reducing the «first step» risk of a poor bad dimensioning of the system.

New software architectures and new HL languages are now available to accept and better exploit these new concepts and capabilities.

We have presented today an overview of the state of the art distributed real time processing systems applied in general aircraft applications and in the Helo environment.

The main definitions and concepts of the evolution of the avionic computing systems have been shown in a step by step sequence.

The following computer architecture have been examined:

— centralized

— federated

— hierarchical

— distributed.

Comparison of the above has been made in respect to:

— H/W implementation

— S/W development and testing

— bus control and organization

— multiple bus structures

— growth capabilities.

Some relationships between A/C mission, computing system architecture and total life cycle cost have been also given.

The final result is that the multiprocessor architecture is now the most suitable for the avionic system in the next decade.