
EUROGAM Project
Real Time Kernels

Edition 1.0
June 1990

. . .

.

M.M. Aléonard
CENBG
Le Haut Vigneau
33175 GRADIGNAN CEDEX, France

Real time Kernels and Operating Systems implementations in EUROGAM

m.m. Aleonard

The EUROGAM acquisition setup has a distributed processors structure. The whole architecture is built around an Ethernet network (see EUROGAM docs) with VXI and VME crates for electronics and fast data handling, and a network of graphics stations and apropriate servers for disk files and storage on tapes.

The SUN based workstations network use a UNIX non real time Operating System. Although in a first stage GEC computers will be in use at Daresbury in conjunction with the SUN stations, in a latter stage, in Strasbourg these GEC will be replaced by UNIX stations.

As a summary the architecture of the EUROGAM acquisition is a UNIX based network and real-time processes are running in several VME and VXI crates. The main requirement of this architecture is to have network tools to be able to download, test and run tasks in all the CPU implemented are they directly on the Ethernet link or reachable through the VME backpanel.

A question is to be answered : do we need (mainly in the VXI crates) an operating system or a kernel system ? This will be examined after a summary of available solutions.

1. Kernels

Several kernels are available with nearly the same performances : VRTX, PSOS, PSOS+, VxWorks. They can be bought for various CPU VME boards from several manufacturers. Among these PSOS+ and VxWorks are very well suited for our application since they have been designed to operate under a UNIX environment on Ethernet (see VME Exec and VxWorks tools). As such they have tools (debuggers, board packages...) : see development tools.

Advantages of a Kernel

A Kernel brings :

- a software interface hiding the hardware peculiarities of a VME board. Each has its own Board Support System (BSP) either in a PROM (VxWorks) or a data file (VME Exec). Activating the same Directive will so be independant of the VME board in use. This is very usefull since processors are upgrading so quickly, both architecture RISC or CISC could be used as soon as the BSP are available. This feature should be very efficient in EUROGAM where several kinds of cards will be used (MVME147, MVME167, FIC8230,...).

- as a software interface it brings homogeneity in high level (C, Fortran) modular programmation, the application being divided in tasks running if necessary in several processors. These tasks are real time processes in communication : multiprocessors will surely be used in Event Builder and Sort engine, the Resource Manager in the VXI crate is may be alone in his crate.

- VME boards from main manufacturers (Motorola, Force Computer,...) are provided with several kernels and their BSP as soon as they are available. This is an advantage for upgrading quickly a configuration. For small companies (CES, Struck one needs to check which Kernel is available). See Table 1 .

- Messages and general I/O processes are UNIX driven. For example in VME Exec calls are typical UNIX commands.

Development tools

They should afford a real improvment for software implementation and with appropriate debugging facilities. These are XRAY68k debugger for VME Exec, DbXworks in the catal of SUN. Debugging with XRAY is performed directly in assembly langage or higher level langage (C).

Table 1 shows on which processors various kernels can run and some indication of prices.

An illustration : VME Exec

The evaluation of a product is related to its knowledge. VME Exec has been bought by CENBG, as an illustration of tools and possibilities of a kernel here are some remarks on VME Exec and PSOS+ :

The basic package is :

-VMEXGEN (system generation utility) : with loading and starting application tools, make file and assembly langage from Motorola, interface for multiprocessors, sharable memory driver.

-The VME Exec kernel is structured around RTEID (Real Time Executive Interface Definition) which gives all real time primitives.

- The kernel itself of VME Exec is now PSOS+. PSOS+ is a monitor with all usual primitives : tasks managment, resources allocations, communication...

- SVIDlib (System V Interface Definition Library) is a software layer interfacing VMEexec and Unix System V AT&T (used on Motorola stations), it allows transportation of Unix software towards a real time application. This layer has 2 file systems : Fast File System and System V file System. Fast File is a real time file manager System V is not.

Equivalent tools exists for VxWorks.

2. Real time Operating System.

This aspect is at contrario of the Kernels notion a whole real time system either promed or running with disc VME based processors. This deterministic system optimize the real time versus messages and network communications. The following discussion will involve the OS9 and OS9000 Operating Systems, both having applications at CERN. OS9 is for the 680xx processors and OS9000 for RISC processors. One is attracted by the possible implementations issued from the CERN equipments, however new implementations on new boards on new OS release suffer from the lack of support from a manufacturer (i.e implementations of Xwindows, Internet...) and release on VME boards may be delayed versus a Kernel version. The migration from OS9 to OS9000 needs about 20 percent recoding, that means that if we do want to keep open some RISC processors (this may be useful in the event builder or the sort engine), we should start directly with OS9000 new applications but Ethernet is not yet available (see table).

No detailed description of OS9 is to be given it is known as a real time multitasks Operating System its real time performances are less efficient than those of a kernel because it is more general and supports more primitives. OS9 is said to be UNIX-like that means that its links to UNIX are less efficients than VxWorks and VME Exec which have been designed in a UNIX environment.

One advantage of OS9 is that it (may) have been practiced by ingeniors for quite a time.

3. What do we need ?

In the main architecture of EUROGAM (see EUROGAM overview) we are interested in real time applications in VXI and VME crates in communication with an Ethernet then FDDI network.

With these 2 functionalities one needs CPU handling efficient protocols on the network, this may be provided either as a standard from VME Exec, VxWorks where RPC from Unix could be reached, or implemented for OS9 or OS9000. The MVME147 seems well fitted for those interested in a SCSI bus. Anyhow even if more expensive the use of the same board for all Ethernet communication seems less software consuming, what about the next generation with FDDI ?

So besides this communication task the MVME 147 may be used as a support for the Resource Manager in the VXI crates (see other documents). Other VME boards are expected to be used : FIC CPU 8230 (68020), MVME165 (68040), and possibly some RISC processors.

Which are the tasks to run in the MVME147 ? :
-cold start
-Ethernet driver
-download task in the board or in the VME space of the crate
-start, stop, resume... a task
-messages transfer
-bloc transfer (spectra, tables...)
-... see RPC and other documents...

For most of these tasks we just need a kernel. Use of standard RPC calls would avoid rewriting drivers both for the UNIX part and the Kernels. More complicated tasks can be down loaded through the network previously compiled. This is already expected for the sort engine for example.

The network facility afford the availability of quickly increasing powerfull workstations with standard processing software (graphics, fortran, a whole history of analysis software...). In such an environment the real time processes should be kept as small as possible. The Resource Manager and Event builder need these small tasks. The Sort Engine is more difficult except if in a latter version a powerfull UNIX (or transputer ?) station is used (as a slightly off-line sort engine) to "crunch" Megawords of spectra and matrices.

4. Conclusion

The Ethernet network is not really a real time data transfer, specially if there are colliding riscs.

In the network context of the EUROGAM acquisition and its UNIX environment the Kernel solution is well suited, it fits the general tandency of manufacturers designing more powerfull UNIX stations.

VME Exec is really fitted to UNIX but up to now limited to Motorola Unix stations. The Ethernet communication is due these days. Intertasks communications is under test. These are proceeding through the VME back panel. VxWorks is well suited to the SUN environment but the task communication is through TCP/IP protocol and involves several ms for intertask communication (however this should evolve to standard VME exchange through the backpanel).

Typically one expects for VME Exec and Vxworks about 40 microseconds for task switch, and 20 microseconds for interrupt services for 68010 due to the efficiency of the kernel.

The standard prices for VxWorks from Wind River quoted in table 1 have to be lowered to 70kF for a 1st licence and 35kF for n additional licences and 3.8kF for n Runtimes on CPU boards. This offer is valuable for one month.
Vxworks and VME Exec are both well suited for designing real time applications tasks. The BSP available for several VME boards manufacturers, as well as the VxWorks SUN environment and debugging facilities in high level langages (C, Fortran...), makes VxWorks more attractive, besides VxWorks is available on VAX's, Appolo... stations. Anyhow in some (future) time both kernels will meet the ORKID (Open Kernel Interface Definition) specifications now in definition for the VITA.

The use of OS9000 should be better than OS9 to leave open future RISC opportunities. An advantage of OS9 is the knowledge of some ingeniors however OS9 is not well designed for networks neither UNIX; but mainly it is lacking development tools which are most efficients in the UNIX stations context these tools should lower development time as soon as people are used to them.

It may be the right time to invest in these tools to be more efficient in the 3 years to come.

	PSOS+			VXWORKS
MOTOROLA 68K	2-20 Mips	CES 8230 MOTOROLA FORCE...	A	A
MOTOROLA 88K	17 Mips		D	
INTEL 386/486	1,5-15 Mips		A	
INTEL i960/i860	30-40 Mips 66-80 MFlops	HEURIKON CSPI MERCURY		A (2)
SUN SPARC	12,5-15-40 Mips 2,4 à 5 MFlops	MIZARD SUN		A (3)
MIPS R3000/R6000	12 à 60 Mips $\sim$ 10 MFlops	CES 8235		
RESEAU ETHERNET	PNA TCP/IP			400-500 Ko/s TCP/IP Xlib client
RESEAU FDDI	pas prévu actuellement			PRONET 80 →FDDI
PRIX	Note (5)			Note (6)

	VRTX	OS 9000
MOTOROLA 68K	A	A
MOTOROLA 88K	A	D septembre
INTEL 386/486		A
INTEL i960/i860	septembre	fin de l'année
SUN SPARC	A (4)	D
MIPS R3000/R6000	octobre (1)	
RESEAU ETHERNET	VELOCITY TCP/IP	en cours
RESEAU FDDI	pas prévu actuellement	pas prévu actuellement
PRIX	Note (7)	Note (8)

A : Available
D : Development

(1) CES développe une carte à base de MIPS R3000 FIC 8235. VRTX sera porté dessus (octobre). VxWorks est en discussion. Il n'y a pas de Cross-Compilateur pour MIPS. Il faut disposer d'une station MIPS (R2000). La carte 8235 de CES disposera d'un driver PSOS et pourra travailler comme co-processeur d'une carte FIC 8230 avec PSOS.

(2) VxWorks tourne sur la carte 1960 HEURIKON HK 80/V960. Coût environ 30 kF pour la carte. La société CSPI propose une carte i860 (environ 75 kF à 33 MHz → 66 MFlops) diposant d'un driver lui permettent de travailler dans l'environnemet VxWorks comme co-processeur. Cross-Compilateur GNU FSF (Free Software Foundation)

(3) SUN France proposera un package SUN 4E/VxWorks.

(4) La chaîne VELOCITY sur SPARC ne fonctionne que pour cible 68000. Pas de prévision pour cible SPARC. Au niveau de la chaîne de développement, il est toujours plus simple de travailler avec une boîte et une cible homogène → Cross Compilateur, debugger symbolique.

(5) 45 kF pour 10 licences PSOS+. Prix PNA ? Rajouter outils de développement; environ 50 kF.

(6) 170 kF pour une licence puis 85 kF pour les suivantes. 5kF par cible.

(7) 190 kF pour une licence puis 80 kF pour les suivantes (remise de 50 %). 6 kF par cible.

(8) 20 kF plus outils complémentaires à préciser.