LA-UR -83-1808

LA-UR--83-1808

DE83 014151

CONF-830577--3

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

MASTER

TITLE RSX SYSTEM DEVELOPMENT UNDER VAX/VMS COMPATIBILITY MODE

AUTHOR(S) Mary A. Fuka

SUBMITTED TO The 1983 Spring DECUS US Symposium
St. Louis, Missouri
May 23-27, 1983

DISCLAIMER

Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO 836 R4
ST NO 2629 5/81

RSX SYSTEM DEVELOPMENT UNDER VAX/VMS COMPATIBILITY MODE*

Mary A. Fuka
Los Alamos National Laboratory
Los Alamos, New Mexico

ABSTRACT

The Control System for the Proton Storage Ring now being built at Los Alamos will use a VAX-11/750 as its main control computer with several LSI-11/23 microprocessors reading and controlling the hardware.

The VMS Compatibility Mode makes it possible to use the VAX as a development system for the LSI-11/23 microprocessors running the RSX-11S (stand-alone) operating system. Digital Equipment Corporation (DEC)-supplied software is used to generate the RSX-11S operating system and DECNET-11S network software. We use the VMS editors to create source files, the Macro-11 assembler and the PDP-11 Fortran-77 compiler to generate object code, and the RSX Task Builder to link the executable RSX task image. The RSX task then can be tested to some extent on the VAX before it is down-line loaded to the LSI-11/23 for further testing.

Several areas of difficulty and "incompatibility" exist in the VMS Compatibility Mode. Areas that can give trouble include file protection, process protection, privilege, directory names, device names, and the Indirect Command File Processor. Understanding these differences between RSX and VMS Compatibility Mode facilitates our system and task development.

WHY USE VAX/VMS FOR RSX-11S SYSTEM DEVELOPMENT?

The RSX-11S operating system is useful for a control-system environment where the only Input/Output (I/O) devices are the interfaces to the actual equipment being controlled. At Los Alamos National Laboratory we are using several LSI-11/23 microcomputers running RSX-11S and networked to a VAX-11/750 to control the Proton Storage Ring at the Clinton P. Anderson Meson Physics Facility (LAMPF). The LSI-11/23s are in an equipment building about 100 m from the building that houses the VAX and the control console. This building does not provide an environment suitable for disks, tape drives, or other mass-storage media. We also plan to have no terminals attached to the microcomputers when the Proton Storage Ring is operating normally. The stand-alone version of RSX provides a memory-only system that allows us to have multiple tasks running in each microcomputer at various priorities and saves us the effort of writing our own scheduling software.

The stand-alone nature of RSX-11S demands that the RSX-11S operating systems and the tasks running on them be created on a host VAX/VMS, RSX-11M, or RSX-11M+ system. Because we have already chosen the VAX-11/750 as the central control computer, it seems natural to use it as the RSX-11S host, thus saving the cost of purchasing, setting up, and maintaining a separate RSX-11M system. This decision also allows all software development to be centralized on one computer, and the software developers find the VAX/VMS environment a pleasant one in which to work.

RSX/VMS DIFFERENCES THAT AFFECT RSX SYSTEM AND TASK DEVELOPMENT

There are some important differences between RSX and VMS that must be kept in mind when developing RSX systems using VMS Compatibility Mode. The VMS Applications Migration Executive (AME) is at MCR Level 3.2; whereas RSX is at Level 4.0. DEC promises that the AME will be brought up to Level 4.0; in the meantime command files designed to run under RSX will not necessarily run under VMS Compatibility Mode.

Another major difference between VMS and RSX is the way the two operating systems define and use privilege. RSX sees a user as either privileged or unprivileged, depending on the User Identification Code (UIC) group number under which that user logged in, and associates this privilege with the user's terminal. Under VMS, a user is given a set of privileges from the wide range of specific privileges that VMS recognizes, and the set of privileges given a particular user is associated with the process created when the user logs in. It is

*Work supported by the US Department of Energy.

possible under VMS to tailor the set of privileges given a particular user to the job he is doing. It requires some rather powerful VMS privileges to do RSX system development and task development under VMS Compatibility Mode. For example, to "SET UIC [group,member]" requires the CMKRNL (Change Mode to Kernel) privilege, which allows a user access to the VMS operating system's highly sensitive and protected data structures!

VMS and RSX use similar file specifications. There are differences, however, in the device and directory fields of the file specification and these differences must be kept in mind when working in Compatibility Mode. VMS allows three formats for directory names:

[group,member]
[name]
[name.name]

RSX uses only the [group,member] format in which "group" and "member" represent group and member numbers that make up the UIC for a particular user. The other two directory formats allow for names made up of combinations of letters and numbers, which can be separated by periods to indicate subfile directories of higher level directories. RSX does not allow subfile directories.

VMS uses device specifications in the format "ddcu:" where dd is a two-character device mnemonic, c is a one-character (alphabetic) controller designation, and u is a unit number. RSX device specifications are similar, but RSX does not use the controller designation character. To associate an RSX device name with a specific VMS device, one can use either a logical name or the automatic mapping of RSX device names to VMS device names in which an RSX unit number is converted to a VMS controller designation and unit number. This mapping is explained in Sec. 2.5.2 of the VAX-11/RSX-11M User's Guide. Logical names are easier to use, but some RSX tasks, the File Transfer Utility (FLX) for example, demand the use of the automatic mapping. FLX also does not understand directory specifications in anything but the [group,member] format, although it can find a file in a named directory or subfile directory if it is the default directory.

Under RSX, the UIC and the directory specification are identical. VMS associates UIC with a particular user process when the user logs in and not with the directory in which the user is working. Files created by a user are given this UIC, no matter the directory in which they are placed, unless the UIC is reset after log-in. Both VMS and RSX use the UIC for file protection. VMS also uses UIC for process protection, allowing a user to affect only other processes with the same group number if the user has the GROUP privilege or all processes if the user has the WORLD privilege. Without either of these privileges, the user can affect only his own subprocesses.

The four-bit RWED file protection fields for SYSTEM OWNER, GROUP and WORLD are the same in VMS and RSX with the exception of the "E" bit. Under RSX, the E bit allows extending the file; under VMS it allows executing an image file, either an RSX or VMS image.

Two other basic differences must be noted for persons developing programs on a VAX to be run on an RSX system: the RSX executive is not present in VAX/VMS nor is the PDP-11 I/O page present. Task images that must be mapped to the RSX executive or to the I/O page can be task built on the VAX, but must be tested on an RSX system. Care must be taken to see that such task images are built using the system libraries for the RSX-11S system.

RSX-11S SYSTEM-DEVELOPMENT TOOLS

Several RSX system development tools are available with RSX-11S, DECNET-11S, VMS, and VMS DECNET.

- It is possible to do RSX SYSGENs on a VAX using the standard RSX-11S distribution.
- Down-line loading and up-line dumping of RSX-11S systems is provided with VMS DECNET.
- The Virtual Monitor Console Routine (VMR) provided with the RSX-11S distribution allows configuring of RSX systems before they are down-line loaded.
- The Virtual Network Processer (VNP) provided with the DECNET-11S distribution allows the network to be loaded into an RSX system image on the VAX before it is down-line loaded.
- It is possible to save an RSX system image on selected media on the VAX in bootable form and then load it through a load device on the LSI-11/23.

SYSGEN AND INITIALIZATION OF RSX SYSTEMS

The RSX-11S Version 4.0 SYSGEN works as distributed on the VAX (at least with Version 3.0 or later) and the documentation provided is relatively easy to follow. This was a pleasant change from our earlier experience with the RSX-11S Version 2.2 (equivalent to RSX-11M Version 3.2) distribution. It is necessary to work from an account with the CMKRNL, LOGIO, SYSPRV, and VOLPRO privileges and to be logged in with MCR as the Command Language Interpreter (CLI=MCR) to do a SYSGEN. Setting up a special account with the necessary privileges and with the default Command Language Interpreter set to MCR facilitates RSX-11S system development. The Login.cmd File for this account can assign logical names for the VAX devices similar to the logical names one expects on an RSX system, for example, SY:, LB:, MP:, and can assign logical names to the task-image files for the RSX-11M Task Builder, VMR, and VNP. Examples of the User Authorization File and Login.cmd File for the RSX system-development account on our system are included at the end of this paper.

Version 4.0 of RSX-11S allows the system developer to choose a group number for the RSX system files. If the SYSGEN is performed on the VAX system disk, group numbers allowed are 3 through 10. This avoids conflicts with files already on the system disk, the Compatibility Mode SYSLIB, for example. Also, this makes it easier to SYSGEN and keep track of several variations of the RSX-11S Executive.

After doing the RSX-11S SYSGEN, the system must be initialized using VMR. Partitions must be set up, drivers loaded, all tasks installed, and any tasks to be fixed in memory must be fixed with VMR. Basic MCR can be included with the RSX-11S system, but it performs only a small subset of the MCR functions. Basic MCR allows one to remove or unfix a task, but does not permit installing or fixing a task. Many of the commands that one would normally do with MCR on an RSX-11M system are done with VMR

for an RSX-11S system. One can also examine the system after it is initialized, using VMR commands DEV (ICES), LUNS, PAR (TITIONS), TAS (KLIST). Finally, any tasks to be run when the system is booted or to be scheduled for some later time can be scheduled with the RUN command. This makes it possible to set up a system that does not require a terminal to run the software installed in the system. The system image can be saved in bootable form, using the VMR SAV (E) command.

We have come across a few problems with the VMR received with our RSX-11S distribution. After doing REM and INS on the same task, the TAS command fails before displaying all the tasks in the system. If one exits from VMR and then enters it again, the TAS command performs as expected. VMR also apparently corrupts the file header of the RSX-11S system image under some error conditions. When giving it a command file that it cannot execute, it gives an error message as expected; then on exiting VMR one finds that the system-image file header has fixed-length records of less than 512 bytes. The system-image file itself seems to be in order; it is the right size, and when compared to a "good" system-image file, no differences are found. However, the down-line-loading software refuses to down-line load the system image with the bad header. It is best to avoid making mistakes when using VMR!

DECNET-11S NETGENS

NETGENs on a VAX for DECNET-11S were not supported when Version 3.1 of DECNET-11S was first released. The command files for NETGEN were at MCR Level 4.0 and included commands not implemented in MCR 3.2, the present level of the VMS AME. I borrowed a friend's PDP-11/60 for the NETGEN and then transferred the files to the VAX. Some time after I did this, a workaround to allow DECNET-11S NETGENs on the VAX became available. I have tried this workaround and it works. It includes the RSX-11M-PLUS Indirect Command Processor and the file NETUSR.EXE to process the GIN$ directives. The only problem is the printing out of strange messages, which the workaround directions assure us are harmless. When the AME is brought up to MCR Level 4.0, the NETGEN command files should work without special workarounds.

NETGEN produces a command file that VMR can use to install and fix in memory the required network tasks and those network tasks selected as options during the NETGEN. This command file can be modified if desired; however it is necessary that the Network Initialization Task (NTINIT) and the Network Ancillary Control Processor (NETACP) be installed and fixed in memory. The Satellite Loader (SLD) is required for down-line loading tasks and the Network Management Driver (NMDRV) is required if network management programs are desired. It can be handy to have the Network Control Program (NCP) installed and fixed in memory in a DECNET-11S system, even though it is rather limited in function.

After generating the network, it must be loaded into the RSX-11S system image using VNP. VNP loads the Communications Executive and Network Processes and is also used to perform many of the functions one might perform with NCP on an RSX-11M system. The DECNET-11S version of NCP implements a small subset of the DECNET-11M NCP commands. One can SET a circuit on or off, LOOP executor and node, SET executor HOST, SET Logging Console on or off, ZERO counters, and SHOW most relevant information with NCP. However, one must use VNP to perform most of the SET, CLEAR, and LOAD commands that DECNET-11M NCP allows. Also one can use either VNP or the Configuration File Editor (CFE) to change network parameters set up during the NETGEN. A sample command file to load the network is included at the end of this paper.

VNP is also useful for examining the network components in the RSX-11S system-image file. One caution must be observed before using VNP: the RSX-11S system image should be backed up. VNP can corrupt the system image if it is aborted or exits in an abnormal fashion. I use a command file that copies the virgin system-image file before using either VMR or VNP. A copy of this command file is at the end of this paper.

DOWN-LINE LOADING OF RSX-11S SYSTEMS

When it comes time to down-line load an RSX-11S system from a VAX, one must stop reading the RSX DECNET manuals and turn to the VMS DECNET Manager's Manual. Down-line loading of RSX-11S systems is built into VMS DECNET. The network database on the VAX is configured to include the RSX-11S node, using the DEFINE NODE or SET NODE commands of VMS NCP. The secondary and tertiary loaders for the down-line loading are included with VMS DECNET; however, I found it necessary to use the loaders built using DLLDAT.CMD on the RSX-11M system, or DLLVMS.CMD included with the workaround, because the ones included with VMS did not work correctly. Once the database is configured to include the RSX-11S node (defining the correct secondary and tertiary loaders, the circuit over which the load is to take place, and the RSX-11S system-image file) the down-line load takes place when the RSX-11S system is booted (provided the remote system is set to boot over the network).

An up-line dump also is possible for an RSX-11S system hosted by a VAX. NETPAN.MAC must be included in the RSX-11S system, directions for doing so are included in Appendix B of the RSX DECNET Network Generation and Installation Guide. The other requirement is specifying a dump file using the VMS NCP DEFINE NODE DUMP FILE command. An up-line dump is generated only if the remote RSX-11S system crashes. An easy way to generate a crash to test this feature is to type "X" to the XDT prompt in an RSX-11S system built to include the executive debugging tool.

SOFTWARE TOOLS FOR RSX TASK IMAGE DEVELOPMENT

VAX/VMS provides several tools for development of programs to run on RSX systems or on the VAX in Compatibility Mode. Among these are

- the MACRO-11 assembler (MAC),
- the Compatibility Mode Task Builder,
- the line text editor (EDI),
- the RSX librarian,
- the Peripheral Interchange Program (PIP),
- the File Transfer Utility,
- the File Patch Utility (ZAP),
- the File Dump Utility (DMP), and
- RMS-11 utilities.

In addition, RSX-11 compilers can be installed and run on the VAX, and the RSX-11M Task Builder can be run in Compatibility Mode to create task images to be run on PDP-11 and LSI-11 machines.

The tools I use regularly for RSX software development are the assembler, the RSX Fortran-77 compiler, and the Task Builder. Occasionally I use the RSX librarian, FLX and DMP. I have seldom used PIP and never use EDI, because I much prefer the DEC standard editor, EDT. Generally, I use the VMS Digital Command Language (DCL) when editing, assembling and compiling and use a special account with CLI=MCR and RSX-type symbols defined when task building and installing a task for down-line loading. It is important to note that calling MCR from DCL is not the same as logging in with CLI=MCR. Calling MCR from VMS DCL allows one to execute an RSX task image, but does not invoke the MCR command language. Logging in with CLI=MCR invokes the MCR command language.

ASSEMBLING AND COMPILING RSX-11S PROGRAMS

I have encountered no problems with the Compatibility Mode Macro-11 assembler. It can be run in several ways, from VMS DCL by typing MACRO/RSX11 filename, or MCR MAC with the usual RSX style output = input string, or from MCR by typing MAC with the standard output = input string. MAC understands VMS directory specifications; thus, it can be run from any directory or subdirectory.

I have used two PDP-11 FORTRAN compilers under VMS: the VMS/PDP-11 FORTRAN-IV cross compiler and the RSX-11 FORTRAN-77 compiler. The FORTRAN-IV cross compiler is a supported VMS product, the RSX-11 FORTRAN-77 compiler is not. (It is supported to run under RSX but not under VMS.) However, we have found both compilers easy to install and use. The FORTRAN-IV cross compiler understands VMS file specifications; the RSX 11 FORTRAN-77 compiler can be installed to operate in the same fashion. To do so, specify the Compatibility Mode SYSLIB as the default system library when task building the compiler. I find it convenient to keep a Compatibility Mode FORTRAN-77 library, which I created by copying the Compatibility Mode SYSLIB and inserting the FORTRAN-77 modules into it, then renaming it. Tasks built with this library understand the VMS file specifications.

I do most of my RSX software development in numbered directories rather than named directories because other necessary tools, VMR and VNP for example, do not understand VMS file specifications. To ensure that the directory specifications and UICs match for these RSX programs, I need to SET UIC to the same [group, member] as the directory when setting a new default. As this requires the CMKRNL privilege and I do not care to work from a highly privileged process when editing and compiling, I work from an account with the SETPRV privilege and use command files to give the process the CMKRNL privilege, set the new default and UIC, then take away the privilege.

There are two ways to set up a logical symbol for the compiler system image under VMS so that it works like an RSX-11M installed program. When working in VMS DCL, assign a logical name to the compiler task-image file in the following manner:

```
F77:==$USERDISK:[1,54]F77.EXE    .
```

The dollar sign before the file specification is important; it signals VMS that the file to which the symbol is assigned is an executable image and allows parameters to follow the command. If the symbol is assigned without the dollar sign, it will not allow the output = input file string to follow the command, and it will be necessary to wait for the F77> prompt to enter output = input. When working in MCR, assign a logical name to the compiler using

```
ASN DR1:[1,54]F77.CXE=F77/GBL    .
```

This also allows the output = input string to follow the F77 command. The /GBL switch puts the symbol in the system logical name table if the user has the SYSNAM privilege. If the symbol is to be made available to all users, it should be in the system Log-in Command File.

TASK BUILDING RSX-11S PROGRAMS

After assembling or compiling a program, I log in, under the same account used for RSX-11S system development, to task build and prepare a task for down-line loading. The default Command Language Interpreter is MCR; the Log-in Command File assigns the symbol TKB to the Task Builder that comes with the RSX-11S distribution, VMR to the Virtual Monitor Routine, and VNP to the Virtual Network Program and also assigns RSX system symbols to VAX devices.

Although the Compatibility Mode Task Builder can be used for linking many RSX programs that will run under RSX-11S systems, I usually use the Task Builder distributed with the RSX-11S system, which gives tasks the .tsk extension and expects an RSX type environment; the system libraries must be in directory [1,1] on the disk to which the symbol SY: is assigned, and any tasks built to be installed in common partitions also must have .tsk and .stb files in [1,1] on this SY: device. It understands only directory specifications in [group,member] format. As long as this Task Builder is given an RSX-type environment, it performs well.

DOWN-LINE LOADING OF RSX-11S TASKS

After Task Building, any task images to be run on the RSX-11S system must be installed in the RSX-11S system image using VMR. If there are old versions of the tasks installed in this image, first they must be removed. It helps to have a system directory in which copies of the RSX-11S system, the system symbol table, and the system and network tasks are kept. Then SET UIC to this directory before using VMR. The tasks to be down-line loaded can be in any numbered directory. After using the commands REM and INS to remove, then reinstall the desired tasks, I copy this system image to the VAX/VMS SYS$SYSTEM directory, where the system image to be down-line loaded is kept, and delete any old versions of this file. It is not necessary to down-line load the RSX-11S system every time a task is removed and reinstalled unless the task is fixed in memory; just be sure the latest copy of the system image with the correct tasks installed is in the file specified as the node load file.

If a task grows larger when a new version is created, the size of the task in the Task Control Block in the RSX-11S system image that has been previously down-line loaded may not be large enough. The message that HLD writes in the the HLD.LOG says that the task is too large for the partition. If one is using a large system partition, this message is confusing, because it is really the task size and not the partition size that is the problem. There are two ways to solve this problem: either down-line load the latest version of the RSX-11S system image or use the GPE command on the remote RSX-11S system to alter the task size in the Task Control Block. I prefer the latter solution because it is simpler and quicker. If a task partition is being used and it really is too small for the task, then VMR will give an error message and it will be necessary to repartition the system with VMR and down-line load the new system.

Each task to be down-line loaded must also have an entry in the Host Loader Mapping Table. Under VMS, all that is required is to enter each task's name and task-image file specification in the file HLD.DAT in the SYS$SYSTEM directory. This is easier to do under VMS than under RSX-11M because the Host Loader Mapping table does not have to be assembled and linked. It is not necessary to change this table when a new version of a task image is created unless the file specification changes (version numbers do not matter). We have found it helpful to have the tasks to be down-line loaded have the same UIC as the nonprivileged network rectory and the HLD.EXE image file. HLD.DAT also must have this UIC or have protection allowing WORLD to read and execute.

CONCLUSIONS

In general, we have found the tools provided by DEC for RSX-11S system and task development under VAX/VMS to be useful, although it has taken more effort than we cared to exert to use these tools efficiently and effectively. The tools have improved greatly since the first RSX-11S distribution for installation under VAX/VMS, but we hope for further improvement; in particular we hope that the AME will be brought up to RSX Level 4.0, that the bug in VMR will be fixed, and that the RSX utilities and system management tools will all understand VMS device and file specifications. It should be possible also to SET UIC with some lesser privilege than CMKRNL. In the meantime, it is convenient to have all of our software and system development on the VAX, and it is possible to develop methods of working, using special accounts and command files that make this reasonably efficient.

EXAMPLES OF COMMAND FILES

Backing up a system image before using VMR or VNP.

File: 11S.CMD

```
DEL RSX11S.SYS;?
COP RSX11S.SYS;1 RSX11S.SYS;?
PU SYSVMR.CMD
```

Partitioning, Loading Drivers, Installing and Fixing Tasks, then Displaying Some Information About the System Image.

```
VMR @SYSVMR
```

File: SYSVMR.CMD

```
RSX11S
SET /POOL=1120
SET /MAIN=CEXPAR:*:60:COM
SET /MAIN=TTPAR:*:164:TASK
LOA TT:
SET /MAIN=SYSPAR:*:137:TASK
SET /MAIN=MCRPAR:*:140:TASK
SET /MAIN=RSDV1H:*:400:TASK
SET /MAIN=DBASE:*:200:COM
SET /MAIN=GEN:*:*:SYS
SET /MAIN=BIRADV:7600:200:DEV
SET /POOL
INS [1,1]BIRADV
INS [7,64]TKN
FIX TKTN
INS [7,64]BASMCR
FIX MCR...
@[350,64]INSNET.CMD
INS [350,64]RSDV1H
FIX RSDV1H
INS [300,1]SIMPLE/PAR=GEN
FIX SIMPLE
INS [300,1]DBASE
INS [300,1]TRCFSA/PAR=GEN
INS [300,1]ISINIT/PAR=GEN
INS [300,1]CATY/PAR=GEN
INS [300,1]LAMMOD/PAR=GEN
INS [300,1]PRVTSK/PAR=GEN
INS [300,1]DLLTRY/PAR=GEN
INS [300,1]READER/PAR=GEN
INS [300,1]WRITDB/PAR=GEN/PRI=60.
SET /TERM=TT0:VT100
SET /TERM=TT1:VT100
RUN MCR... 55
RUN READER 10S
DEV
PAR
```

Installing the Network Tasks

File: INSNET.CMD

```
SET /NETUIC=[350,64]
INS [350,64]NTINIT
FIX NTINIT
INS [350,64]NCP/TASK=NCP/CKP=NO
FIX NCP
LOA NM:
INS [350,64]NETACP
FIX NETACP
INS [350,64]MIR/CKP=NO
FIX MIR...
INS [350,64]SLD
FIX LDR...
LOA DV:
```

Loading the Network

```
VNP @1SSVNP
```

File: 1SSVNP.CMD

```
RSX11S
SET SYSTEM ALL ECHO
SET EXE STATE ON
```

Copying the System Image to the SYS$SYSTEM Directory

File: NEWVERS.CMD

```
SET UIC [1,4]
SET DEF SYS$SYSTEM
DELETE ISS11S.SYS;*
COPY USERDISK:[350,64]RSX11S.SYS ISS11S.SYS
SET PROTECTION=(WORLD:RE) ISS11S.SYS
DIR/FU ISS11S.SYS
SET UIC [300,1]
SET DEF USERDISK
```

Setting UIC From an Account With the SETPRV Privilege

File: S101.COM

```
$ SET PROCESS/PRIV=CMKRNL
$ SET DEF [300,101]
$ SET UIC [300,101]
$ SET PROCESS/PRIV=NOCMKRNL
```

USEFUL MANUALS

VAX-11/RSX-11M User's Guide

VAX-11/RSX-11M Programmer's Reference Manual

DECNET-VAX System Manager's Guide

RSX-11M/M-PLUS System Management Guide

RSX-11M System Generation and Installation Guide

RSX-11S System Generation and Installation Guide

PDP-11 FORTRAN-77 Installation Guide/Release Notes

RSX DECNET Network Generation and Installation Guide