DIGITAL EQUIPMENT CORPORATION

COMPONENT COMPUTERS
APPLICATION NOTE

COMPONENTS GROUP

JANUARY 1976

Using The LSI-11 As A Remote Node In A Computer Network

SCOPE

This application note presents the range of information required to use the LSI-11 microcomputer as a remote node in a computer network.

- Technical discussions in this note begin with a comprehensive description of Digital Equipment Corporation's real time disk operating system RT-11. This system is resident in the host PDP-11/35 and serves as the principle operating software supporting this network application.
- Following this is a discussion of the network functions implemented by REMOTE-11. This system, which operates under control of RT-11, is specifically designed to support such network functions as down-line loading and remote operation of application programs, remote debugging and editing, and communication line protocol.
- Technical discussions conclude with the detailed instructions for configuring an LSI-11 system for operation in a network remote environment. These instructions cover the configuring of all the modules comprising the LSI-11 along with the backplanes which mount the system.
- Altogether, the information contained in this application note is at a level sufficient to actually implement a network system having LSI-11 microcomputers operating in a remote environment.

INTRODUCTION

Efficient, economic implementation of computer based networks requires a level of computational power not provided by most in-production microcomputers. With rare exception, modern LSI microcomputer design represents an economic/technological compromise characterized by restricted instruction sets, limited I/O capability, and low computation speed. As a consequence, the ability of the microprocessor to efficiently perform the functions necessary for network processing, namely down-line program loading, interrupt processing, and high speed computation, has inhibited the development of low cost, microprocessor based, network systems.

Digital Equipment's LSI-11 microcomputer represents a significant departure from conventional microprocessor design. First, the LSI-11 is an LSI microcircuit version of the PDP-11 minicomputer, without compromise in instruction repertoire, I/O capability, or computational speed. Second, direct software compatability between the PDP-11 family and LSI-11 and PDP-11/03 microcomputers provide a range of support software unavailable to conventional microprocessors. This support software, specifically the PDP-11 family RT-11 and REMOTE-11 systems, includes sophisticated assemblers, utility, debugging, and network support modules, as well as resident executives which execute FORTRAN IV, FOCAL, and BASIC object programs.

Finally, the availability of a powerful software operating system, coupled with widespread familiarity with PDP-11 programming principles, makes new application program development a far easier task when compared to developing programs using microcode or without supporting software tools.

Taken together, the characteristic low cost and high computational power of the LSI-11 and PDP-11/03, provide the level of cost effectiveness necessary to development of consumer and industrial network system applications previously considered economically unfeasible. These new network system application areas range from banking transactions, credit verification, security, and reservation processing to numerical tool control, process control, general and special purpose robots, message switching and inventory control.

APPLICATION OVERVIEW

The intent of this application note is two-fold:

1. To demonstrate that the high computational power and ease of application program development characteristic of LSI-11 systems, when coupled with an extremely low system cost, opens areas for LSI-11 based network applications previously not considered feasible or economic.
2. To present the range of background information on LSI-11 hardware configuration and supporting software necessary to the analysis of specific LSI-11 network system applications.

Computer-based networks are distributed computer systems consisting of a number of processor/peripheral configurations linked by communication lines to form a functionally cohesive system. The network system described in this application is a point-to-point, hierarchical network implemented by a PDP-11/35 and up to seven LSI-11 microcomputers.

This generalized network, which is illustrated by the overall block diagram shown in Figure 1, is supported at the host processor, in all operational aspects, by the PDP-11 family REMOTE-11 system. REMOTE-11 is designed specifically for network support, and operates under executive control of the PDP-11 family RT-11 high performance real-time disk operating system. The system consists of a multiuser text editor LETTER, communications handling routines based on the Digital Data Communication Message Protocol [DDCMP], and routines for down-line loading programs to remote processors.

Under control of RT-11 the REMOTE-11 system provides for the sharing of system resources with all network nodes, operation of application programs in a multiuser foreground/background environment, and complete translation, compilation, interpretation, linking, debugging, and file maintenance support.

Operation at the remote LSI-11 is supported by the RT-11 simulator RT SIM, which is down-line loaded with the as-

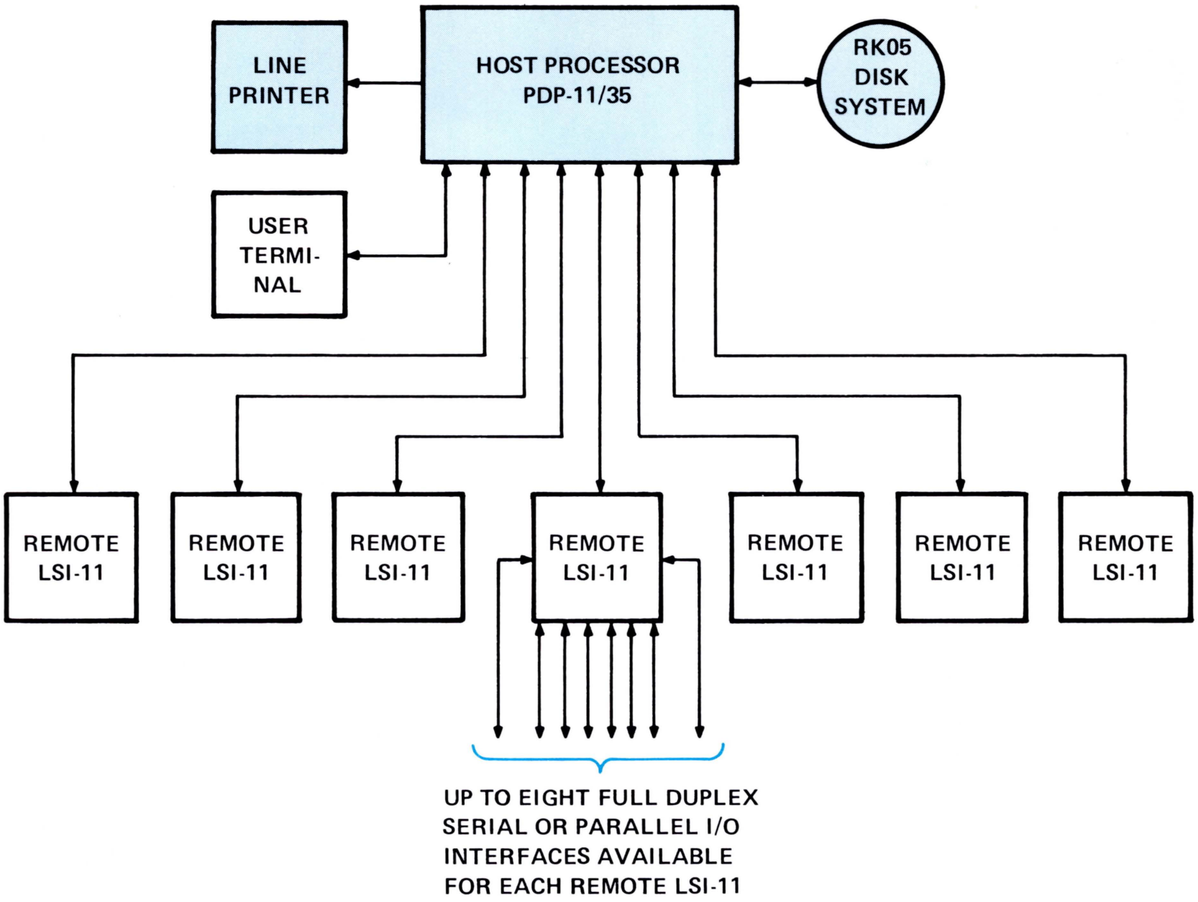


Figure 1 System Block Diagram

sembled or compiled application program by REMOTE 11. Once loaded, control is passed to the starting location of the application program for execution under control of RTSIM, which simulates RT-11 at the remote node. Application tasks, along with communication with the host processor is conducted in a single job mode. However, the LSI-11 interrupt priority and vectoring capability would permit a user to implement foreground/background processing at the remote LSI-11 level. By assigning the lowest interrupt priority to host system communication, this function could comprise the background processing with high priority I/O processing forming the foreground.

Further, by enabling the real-time clock function integral to the KD11-F processor module, I/O interrupts can be handled by the remote LSI-11 on a real-time basis. When enabled, the real-time clock, which has the highest external interrupt priority, would synchronously interrupt the application program to define the boundaries of a real-time processing frame. As shown in Figure 1, a remote LSI-11 can handle up to eight serial or parallel full duplex communication lines. This capacity is based on an LSI-11 system configured on two backplanes and equipped with an 8K dynamic MOS, random access memory.

In addition to the software development and operation support provided by RT-11, REMOTE-11, and RTSIM, the LSI-11 user has access to the vast collection of network application software available through the Digital Equipment Computer Users Society [DECUS] library. Any network application programs developed for use on PDP-11/04 through PDP-11/40 systems are executable on the LSI-11 with only the most minor modification primarily involving program references to the Processor Status Word (PSW).

TECHNICAL DESCRIPTION

Implementing a computer-based network application using LSI-11 microcomputers as remote processors, requires detailed information on the following topics:

1. System operation as supported by RT-11.
2. Development and operation of remote processor application programs as supported by REMOTE-11 and RTSIM.
3. Communication Protocol.
4. Configuring an LSI-11 to serve as a remote processor in a network.

This detailed information is presented in the paragraphs that follow.

SYSTEM OPERATIONS SUPPORTED BY RT-11

Primary software support for the host PDP-11/35 in this generalized LSI-11-PDP-11/35 network system is provided by the RT-11 disk operating system. RT-11 is a low cost, low overhead, real time, application program development and operating system designed specifically for the single user involved in application program development and operation of systems such as computer networks. This software system provides simple, direct, and fast on-line access to the host PDP-11/35 and to remote LSI-11 processors through the REMOTE-11 system. RT-11 can support the operation of REMOTE-11 on the host PDP-11/35 through one of two system monitors; specifically the single job (SJ) monitor or the foreground/background (F/B) monitor.

SJ MONITOR

The RT-11 SJ monitor is a single user, single job executive which requires approximately half the memory space used by the F/B monitor. This high speed executive is designed to support extensive application program development activity as well as for the operation of those user programs characterized by extremely high data rates. With respect to application program development, the smaller size of the monitor in combination with full access to all RT-11 resources, provides for larger memory space for application program development along with the full range of modern software tools. Also, the high speed interrupt servicing capability of the SJ monitor facilitates the handling of high data-rate application programs.

Under control of the SJ monitor, REMOTE-11 can function as a multi-user editor, a down-line loader of application programs to remote processors, or a handler of file maintenance and updating requests between the remote and host processors. Under SJ monitor control, such operations as assemblies, linkages, compilations interpretations, remote communications, or debugging functions must be performed separately by REMOTE-11 with the exclusion of all other operations.

The RT-11 SJ monitor will operate REMOTE-11 on a PDP-11 host computer having a memory capacity of 16 K words. REMOTE-11 and the monitor would occupy 8 K words plus 1 K words for each user and perform on-line editing, down-line loading, and operational control of remote processing by REMOTE-11. Execution of application program assemblies, linkages, interpretations, and compilations by RT-11 would require either removal of REMOTE-11 or additional memory capacity. With this memory capacity, REMOTE-11 under SJ monitor control, can handle up to eight single users with each user being serviced on a queuing basis. Figure 1 depicts an on-line terminal at the host processor with the remaining seven users being up to seven remote LSI-11 microcomputers.

Operation of REMOTE-11 under control of the RT-11 SJ monitor is ideal for use in those network applications which, by their nature, do not require a background processing capability. Such applications could include both real and non-real time networks and networks performing batch processing at the remote level and data logging at the host level. In addition, in the user development of turnkey networks, REMOTE-11 under the SJ-monitor, could be used along with the full range of RT-11 resources at the host computer to develop and install all network operating programs.

Upon completion of the application program development phase REMOTE-11 monitor could then down-line load and operate the completed system. It should be noted that use of RT-11 resources under the supervision of the SJ monitor requires that REMOTE-11 be removed from the host processor programming environment. Conversely, when REMOTE-11 is operating, it has access only to its own resources; specifically the on-line text editor LETTER, the ability to down-line load application programs with RTSIM, and the capacity for conducting network intercommunications.

F/B MONITOR

The RT-11 F/B monitor is designed to execute time dependent program tasks, such as real-time input interrupts, in the foreground, and time independent tasks using RT-11 resources, such as application program development, in the background.

In addition, the F/B monitor along with the SJ monitor provides a wide range of "programmed requests" in the form of assembler MACRO calls for use by host level application programs executing under RT-11 control. These programmed requests support such user program functions as file maintenance, data transfer control, I/O device control, I/O operations, and general program housekeeping tasks.

The F/B monitor, however, has several unique programmed request functions not available from the SJ monitor. These are:

1. MARK TIME — This programmed request provides for the scheduling of elapsed time processes such as DMA transfers by permitting the application program to time a given function. When the specified time has elapsed, a user specified routine such as a completion routine is entered to terminate the timed processes.
2. TIME WAIT — This programmed request permits the application program to suspend operation until a preset time period has elapsed. This feature is useful where a program is required to periodically perform some function such as sampling data where the time periods can vary from seconds to minutes. While the initiating program is idle, another program task can be executed, with a return being made to the initiating program at the end of the preset elapsed time.
3. SEND DATA/RECEIVE DATA — These two programmed requests permit the transfer of data and messages between foreground and background application programs. For example, data could be passed from a foreground data collection program to a background analysis program.

Within the context of network processing, these F/B monitor features serve to facilitate such host computer network related functions as the building and maintaining of data files based on input from remote network nodes, and processing of data received in the foreground by a low overhead background. REMOTE-11, under control of the F/B monitor, resides in the foreground with the background available for assemblies, linkages, compilations, interpretations, debugging, and other such activity connected with application program development. For example, through an interactive terminal at the host computer, which would be one of the eight single users serviced by REMOTE-11, a user could apply the resources of REMOTE-11 to down-line load, debug, and execute application programs at selected remote processors. In addition, through appropriate REMOTE-11 on-line commands, the user can leave the foreground to access those resources of RT-11 residing in the background.

In the role of supporting the operation of REMOTE-11 at the host level, the F/B monitor requires a nominal host processor memory capacity of at least 28 K words. Out of this 28K total, 10 K plus 1 K for each user is reserved for foreground programs (REMOTE-11 and the F/B monitor) and 10 K minimum for background programs (application programs and RT-11 resources).

SYSTEM OPERATIONS SUPPORTED BY REMOTE-11

REMOTE-11 is a PDP-11 family software system designed specifically to support the development and operation of computer networks systems at both the host and remote processor levels. This support software is formed by three operational components:

1. A powerful multi-user text editor designated LETTER.
2. A set of routines with associated commands, which control down-line loading, configuration, and execution of application programs at the remote processor (LSI-11) level.
3. A set of routines which facilitate communications between the remote processor and the host.

The user has the option to structure REMOTE-11 in three versions:

1. A version consisting of the text editor LETTER only. This version optimizes the use of host processor memory space during application program development and would be run with the RT-11 SJ monitor.
2. A version supporting networks processing only. This version optimizes host processor memory and execution of application programs at the remote processor (LSI-11) level.
3. A set of routines which facilitate communication between the host processor and the remote processor, including terminals operational at the remote station.

Each of the components comprising REMOTE-11 are described in the paragraphs that follow.

LETTER

LETTER is a highly flexible on-line text editor whose function is to create, open, modify, manipulate, update and close

ASCII data files. These files can consist of source application programs, data base files, or any special-case user file consisting of ASCII coded data. The major user advantage provided by LETTER is the ability to edit application source programs and data bases on-line, at either the host processor, or at the remote LSI-11. Editing with LETTER can be performed in two modes: data can be called for editing, either one line at a time, or as a page or block.

In the line mode, lines of text from a source file are read one line at a time from an internal I/O buffer, operated on by the user, and when applicable, written back into a holding buffer on the disk. If the user aborts a file editing operation, the source file will remain unchanged, irrespective of the number of lines edited. Upon completion of a file editing operation, the user can replace the old source file on the disk with the edited source file. Since this mode does not require the larger text buffer, it is ideal for use where memory space is restricted.

In the page or block mode, a page of text, delimited by form feeds or a pre-specified number of lines, is read from the disk source file and written into the text buffer space reserved in host processor memory. The page of text is then edited as necessary by the user and upon completion, is stored by LETTER for subsequent updating of the source file. In this mode, LETTER assigns a minimum 256 character host processor text buffer to each user. The actual size of this text buffer depends on the available memory space. If less than 256 characters are available for a given user, the program will inform that user.

LETTER executes more than fifty editing commands which can be grouped into five functional categories. These are:

1. File input/output operations — These commands are used to read selected files as lines or pages of text from the disk, save or restore sections of text, and open secondary files for merging with a primary file.
2. Altering the line pointer — The commands in this category permit the user to access new single lines of interest within a text grouping or to move the line pointer to new lines of interest within a block of text. User specified LETTER editing commands are executed with respect to a reference text line called the "current line" which is the line of interest to the user and the last line located by LETTER. When operating in the line mode, once a new line has been located it becomes the current line and the prior line becomes unaccessible unless the file containing the past line is closed and again reopened. In the block mode, the user can move the line pointer at will, with respect to the text lines contained in the text block currently stored in the pertinent buffer. In this manner the user can designate new lines of interest.
3. Search for test strings — Commands in this category permit the user to search for and locate a specific line of text as identified by a string of alphanumerical characters associated with the command. The line pointer is positioned at line of text so located, and that line becomes the current line. That line can then be modified on a word or character basis, or have additional text added as necessary. Searches for strings can be made in two ways; for specified character strings located either at the beginning of a line or at any position within a line.
4. Modify text strings — The text modification commands implemented by LETTER permit the user to add characters to the current line, change portions of that lines, and retype list or overlay the current line with one or more new lines.
5. Utility operations — LETTER also provides the user with a command set which, although not necessary for text editing, does contribute to the overall task of application program development. For example, the user can define new MACROs for later use, erase a block, or alter the size of the block buffer, define new concatenating characters and enable automatic display of text lines during searches and after modification.

DOWN-LINE LOADING

Under REMOTE-11, down-line loading of application programs from the host processor to the remote LSI-11 is implemented by two commands; specifically GET or RUN. The GET command will down-line load an application program in absolute address form, but will not start that program. The RUN command will also initiate down-line loading of an absolute program and will automatically transfer control to the loaded program for execution if that program contains an absolute transfer address. If the loaded program does not contain such an address the command executes as a GET command. A program which is down-line loaded by a GET command can be subsequently started by the REMOTE-11 START, or REENTER commands providing an absolute transfer address is embedded in the down-line loaded program.

An LSI-11 microcomputer can, for example, be configured so that when power is applied, control is transferred to a user defined bootstrap program. This program would then issue one of the REMOTE-11 instructions GET or RUN to the host processor in order to initiate down-line loading. If a series of application programs are to be executed at the REMOTE node, the first program, upon completion, would transmit down-line loading commands to the host processor to initiate the loading of the second program. This sequence would continue until all required application programs have been executed. Alternately, a remote LSI-11 can be configured so that a local terminal could be used to initiate the down-line loading process. In this case, the terminal would use one of the up to eight communication lines available at the remote processor.

NETWORK COMMUNICATIONS

Under REMOTE-11, communication activities between a host processor and a remote LSI-11 can be conducted in two ways; in the terminal mode, or under control of Digital Data Communications Message Protocol (DDCMP).

In the terminal mode, a terminal connected to a remote LSI-11 can be operated as though it was connected directly to the host processor. The user can then operate the remote terminal on an interactive basis with the host processor and access the resources provided by REMOTE-11 and RT-11 at the host processor. In this manner, the user can perform all the activity necessary to application program development and checkout at the remote node. The terminal mode can be automatically entered at start-up time through a user defined bootstrap program which connects the terminal through the LSI-11 to the host over the host/remote communications line.

With RT-11 and REMOTE-11 resident at the host, and with the remote node one of the REMOTE-11 users, the terminal operator can then use the host resources as required. Note that for terminal I/O operations between the host and the remote LSI-11, REMOTE-11 automatically monitors data rates and halts transmission when a buffer overrun is imminent, and restores transmission when buffer content is reduced to a designated level.

In addition, data input through the remote terminal is transmitted to the host as character strings with parity checking conducted optionally, only when specified as part of the system configuration. Similarly, responses from the host processor are received at the remote terminal with optional parity checking.

Digital Equipment Corporation's communications line protocol, DDCMP, is invoked by REMOTE-11 for all down-line loading transfers and for file transfers between the host processor and the remote LSI-11. This protocol can be implemented at the remote level by the REMOTE-11-supported remote operating system RTSIM which is linked with applications and/or specific RT-11 modules at the host processor prior to down-line loading.

This subset of REMOTE-11 is described in this section under the heading REMOTE PROCESSOR OPERATIONS SUPPORTED BY RTSIM. The use of DDCMP for down-line loading and for file data transfers insures error-free receipt of transmitted data since error detection results in automatic retransmission of the data subject to transmission error. Data transmission between a host processor and the remote LSI-11 is supported under DDCMP by REMOTE-11 for both serial and parallel transmission.

REMOTE PROCESSOR OPERATIONS SUPPORTED BY RTSIM

RTSIM is a module within REMOTE-11 which implements a range of RT-11 monitor and support functions at the remote processor level. RTSIM, then, simulates those functions necessary to development and checkout of network application programs at the remote LSI-11.

Specific RT-11 modules such as the on-line debugging module, and the BASIC/FOCAL interpreters are executed on the remote processors, while others such as the various file manipulation modules and FORTRAN compilers are executed at the host level with only resultant program or data being down-line loaded. However, this interaction between the host and remote processors is transparent to the user.

At a remote LSI-11 equipped with an interactive terminal, the user can access files stored on the host disk system, perform file operations and restore files on the disk. File operations include changing, updating, revising or copying existing files; merging existing files to form new files; and creating new files from external data. In addition, down-line loading for execution by the remote LSI-11 can be initiated for application programs which were FORTRAN compiled or assembled by the RT-11 MACRO assembler, as well as for BASIC or FOCAL statements. FORTRAN compilations and absolute object programs are down-line loaded as core images with RTSIM embedded in the image. Similarly, a BASIC or FOCAL interpreter is down-line loaded accompanied by RTSIM. Programs written in BASIC or FOCAL are accessed through DDCMP file transfers for execution by the pertinent interpreter in the remote LSI-11

In actual practice, RTSIM and the application program to be down-line loaded are linked by the RT-11 linker at the host processor. However, the user at a remote terminal operating in the terminal mode, can perform this linking process as though he was operating a terminal connected directly to the host processor. The user at a remote LSI-11, has access to all the software resources available to the larger and more powerful host processor.

To demonstrate the use of RTSIM at a remote LSI-11, consider this example of the linking, down-line loading, and execution of the FORTRAN-coded application program XYZ stored on the host disk system. The user at the remote terminal would type in the following statements to link the program XYZ for execution on an 8 K remote processor:

```
.  R LINK
*  XYZ = XYZ, RTSIM/F/L/I
$  8K
carriage return
*
```

Where the linker switch F causes a search of the FORTRAN disk library for application program XYZ, switch L specifies that this linked program be down-line loaded in absolute address (LDA) format and switch I causes inclusion of the appropriate memory space parameters. The reference to the 8 K memory size allows the linker to designate a link address appropriate to an 8 K memory space.

With the link between the application program XYZ established, the user can then initiate down-line loading and execution of program XYZ by keying in the following REMOTE-11 commands at the remote terminal:

```
FILE: @GET XYZ.LDA
FILE: @START
      or simply:
      @ R XYZ
```

Both of these command strings would cause REMOTE-11 to down-line load the application program linked with the RTSIM monitor and to be automatically executed. When the FORTRAN STOP statement in XYZ is executed, REMOTE-11 returns the remote terminal to the terminal mode.

Consider the case of the application program UVW, which is a machine language program assembled by the RT-11 MACRO assembler, but not yet debugged. The user can link the program UVW with RTSIM and the RT-11 on-line debugging module ODT, and down-line load the linked programs to debug the application program on-line at the remote terminal. The command strings keyed in at the remote terminal to perform this linking would be:

```
.  R LINK
*  UVW = UVW, RTSIM, ODT/L
$  8K
```

Program UVW would then be down-line loaded in the same manner as with FORTRAN program except that control would be turned over to RT-11 module ODT after down-line loading, with the remote terminal remaining in the terminal mode. The user could then key in ODT debugging commands to check out the application program. Once debugged, correction to the source code could be made from the remote terminal using the REMOTE-11 editor LETTER.

In the case of user BASIC or FOCAL programs, contained on the host disk system, the appropriate interpreter is linked with RTSIM and down-line loaded to the remote processor along with the ASCII statements comprising the program.

CONFIGURING A REMOTE LSI-11

LSI-11 systems are formed by a processor module, the specific memory and interface modules required by the application, and one or more backplanes to mount and connect the system. In order to obtain the widest possible latitude in adapting the LSI-11 microcomputer to specific applications, strategic functions in each module circuit can be enabled or disabled through connection or disconnection of jumpers.

Preparing an LSI-11 to serve as a remote processor in a network system then, requires that the jumpers on each module be properly configured. Also, the two backplanes mounting the system modules must be configured in terms of bus interconnection and termination, power supply connection and module placement in the backplane slots.

Each of the configuration tasks necessary to preparing an LSI-11 to serve in a network application are described under the topics that follow. Since the network application described herein is necessarily of a very general nature, these descriptions should be viewed as guiding examples rather than specific implementing instructions.

CONFIGURING THE KD11-F PROCESSOR MODULE

A KD11-F Processor module is a complete central processor characterized by PDP-11 architecture and equipped with 4 K of semiconductor memory. This module serves as the central processor for an LSI-11 system and occupies the first two slots in the primary backplane. In this position it is at the electrical head of the system bus and therefore has the highest system priority. The six jumpers on the KD11-F processor modules are configured at the factory for general applications. This factory configuration is listed in Table 1.

TABLE 1
FACTORY INSTALLED JUMPER CONFIGURATION
FOR THE KD11-F PROCESSOR MODULE

W1	W2	W3	W4	W5	W6	Function
X	X	X	R	X	X	Memory Refresh Enabled
X	X	R	X	X	X	Line Time Clock Enable
X	X	X	X	R	R	Power-Up Mode 0
R	I	X	X	X	X	Resident Memory Bank 0 Selected

NOTE
X = Don't Care
I = Installed
R = Removed

RESIDENT MEMORY BANK ASSIGNMENT

As shown in Table 1, the 4 K integral memory is assigned to memory bank zero (locations 0 to 17777) so that jumpers W1 and W2 do not have to be altered.

REAL-TIME CLOCK

If a user application at the remote level is event driven in real-time, or operates within a real-time processing frame, the line time clock interrupt on the KD11-F processor module should be enabled. As shown in Table 1, jumper W3, which enables the KD11-F line time clock when removed, has the line time clock interrupt enabled. To disable this function the user merely has to install the jumper.

When incorporating the line time clock in a specific application the user must provide a 3.6V (nominal), 60 Hz square wave input to drive the clock. This input can be the line voltage properly transformed and clipped, with the negative going excursion eliminated. For implementing details, refer to the LSI-11, PDP11/03 User's Manual (see Reference Documents).

MEMORY REFRESH

A KD11-F processor module is equipped with a 4 K dynamic MOS random access memory contained on the module PC board. The nature of this memory is such that it requires periodic refreshing in order to maintain the stored content. Further, the additional memory module recommended for this application, specifically the MSV11-B 4 K by 16-bit MOS RAM, also requires periodic refreshing. The KD11-F processor module is factory configured with jumper W4 removed to enable the memory refresh logic contained on the module. This refresh signal is propagated along the H9270 backplane bus structure for use by the MSV11-B memory module. Since the refresh function has the highest internal interrupt priority, the refreshing of memory data is assured irrespective of the current processor task.

POWER-UP MODES

KD11-F jumpers W5 and W6 serve to implement one of four power-up modes, which are summarized in Table 2.

TABLE 2
KD11-F POWER-UP MODE OPTION SUMMARY

Mode	Jumpers W6	Jumpers W5	Power-Up Mode Selected
0	R	R	PC@24 and PS@26, or Halt mode
1	R	I	ODT Mode
2	I	R	PC @ 173000 for user bootstrap
3	I	I	Special processor microcode (not implemented)
NOTE: R = Jumper Removed, I = Jumper Installed			

A KD11-F is factory configured to implement power-up mode 0 (jumpers W5 and W6 not connected) which is inappropriate for network applications. The actual power-up mode selected depends on the operational environment of the remote LSI-11. If the remote LSI-11 is to be operated by the user in the terminal mode with that terminal serving as the system console, then power-up mode 1 should be implemented by connecting jumper W5.

With the power-up mode 1 selected, halting the LSI-11 immediately places it in the console ODT mode causing the processor to execute the ODT console microcode which operates the console terminal and is integral to the KD11-F processor module. Executing this code causes the LSI-11 to address memory location 177560 which is the assigned address of the console terminal. At this point the user can input instructions to initiate startup and communication with the host processor to begin down-line loading of application programs.

If the remote LSI-11 is to operate under host processor program control, or under control of users located at host processor terminals with little or no user intervention at the remote level, then power-up mode 2 should be selected. This option is selected by connecting jumper W6, and removing jumper W5.

With power-up mode 2 selected, pressing a user-installed initializing switch causes the LSI-11 to execute an integral microcode sequence which transfers control to location 173000; a reserved location in the seventh memory bank. Typically, this location would be the first location of an MRV11-AA PROM/ROM module minimally configured as a 256 X 16 bit read-only memory and hard-wire assigned to the seventh memory bank. This memory would contain a user defined bootstrap program which would initiate communication with the host processor and subsequent down-line loading of application programs for remote execution.

CONFIGURING THE MSV11-B DYNAMIC RAM

The second 4 K of memory for this generalized network application is specified as an MSV11-B dynamic MOS memory module. This module is equipped with four jumpers; three to control bank assignment (W1, W2, W3) and one to provide for disabling the reply to refresh signal from the module (W4).

MEMORY BANK ASSIGNMENT

Since this application requires an 8 K memory, and therefore only one MSV11-B memory module, this module can occupy memory bank 1 which requires a wirewrap connection on jumper W3. However, if a user wishes to configure a system for additional memory or assign this module to a different bank, Table 3 below describes the jumper configuration for various bank assignments.

TABLE 3
MSV11-B MEMORY BANK ASSIGNMENT
JUMPER CONFIGURATION

Jumper		Function
W1	I	This jumper configuration assigns the 4 K module to memory bank zero.
W2	I	
W3	I	
W4	R	With this jumper removed, the module will reply to a refresh cycle.
NOTE: I = Installed, R = Removed		

MEMORY REFRESH

LSI-11 Microcomputers, configured with MSV11-B dynamic MOS RAM modules, require that the memory module, which is electrically the farthest away on the bus from the processor module, be configured to provide the reply to refresh signal. This arrangement assures that during a refresh cycle, all MSV11-B RAM modules will be refreshed and that the refresh logic on the processor module will receive an acknowledgement of that fact. Since each MSV11-B module is factory configured to supply this acknowledge signal (W4 removed) this jumper does not have to be altered for use in an 8 K system.

CONFIGURING THE MRV11-AA PROM/ROM MEMORY

When a user chooses not to use his remote LSI-11 in a terminal mode, he must have provisions for incorporating a bootstrap routine in his memory space. Since the memory space specified for this generalized network application is implemented by dynamic MOS RAMs, an additional non-volatile memory for the bootstrap is provided by the MRV11-AA PROM/ROM. This module is equipped with 32 16-pin IC sockets permitting a 4 K ROM space made up of 32 512 X 4 bit memory circuits, or up to a 2 K word space made up of 32 256 X 4 bit memory circuits. For the purposes of this application, four 256 X 4 bit PROM circuits, obtained from the sources listed below, should be installed in the sixth row of four sockets which is the third row down from the top or handle end of the module.

Manufacturer	Type
Intersil	IM5623
Monolithic Memories	6031
Signetics	8259

This module is equipped with 18 jumpers W0 through W17 which can be configured for a memory capacity ranging from 256 to 4 K X 16-bit words in either 256 or 512 word increments. In addition, by configuring these jumpers, the module can be assigned to any equivalent memory space within 32 K words. The MRV11-AA must be configured so that the user defined bootstrap resides in the 256 word address space 173,000 to 173,776. To this end, Table 4 compares the MRV11-AA factory-installed jumper configuration, which is that of a 4 K word ROM, with the jumper configuration pertinent to this application.

TABLE 4
MRV11-AA FACTORY VS NETWORK
JUMPER CONFIGURATION COMPARISON

Factory Installed Jumper Configuration		Network Application Configuration	
Jumper Desig- nation	Jumper Installed = I Jumper Removed = R	Jumper Desig- nation	Jumper Installed = I Jumper Removed = R
W0	I	W0	R
W1	I	W1	R
W2	I	W2	R
W3	I	W3	R
W4	I	W4	R
W5	I	W5	I
W6	I	W6	R
W7	I	W7	R
W8	I	W8	R
W9	I	W9	R
W10	I	W10	R
W11	R	W11	I
W12	R	W12	I
W13	R	W13	R
W14	R	W14	I
W15	I	W15	R
W16	I	W16	R
W17	I	W17	R

Inspection of Table 4, and reference to Figure 2, will show that to configure an MRV11-AA for this generalized network application requires that jumpers W0 through W4, W6 through W10, and W15 through W17 be removed or cut. Also jumpers must be installed on W11, W12, and W14. Removing the chip select jumpers W0 through W4 and W6 through W7 places the 256 word address space within the absolute address space 173,000 to 173,776. Removing jumpers W8, W9 and W10 and installing W11 and W12, the module is configured for the four 256 x 4 bit PROM circuits installed for this application. Installing W14 assigns the 256 word memory space to the upper 2 K of an overall 4 K space. Finally, installing jumpers W15, W16, and W17 completes assignment of the 256 word address space to the upper 2 K section of the 7th memory bank, which is the recommended memory space for device addresses.

As an alternative to a user defined bootstrap contained in an MRV11-AA PROM/ROM module, Digital Equipment Corporation furnishes two optional bootstrap modules; one to bootstrap the RXV11 floppy disk and a second containing a serial communications line bootstrap which can be used for a paper tape reader/punch. The device address for both is 173000 which is the same address specified for the user-defined bootstrap, implemented by power-up mode 2.

These optional modules also refresh the MSV11-B memory independent of the CPU thereby increasing CPU efficiency.

CONFIGURING DLV11 SERIAL LINE UNITS

A DLV11 Serial Line Unit (SLU) is equipped with thirty jumpers which can be configured to satisfy specific application requirements. The DLV11 SLU is factory configured to serve as a console terminal interface. This configuration is summarized in Table 5. As shown in Table 5, the 30 SLU jumpers fall into five categories; specifically, device address assignment, interrupt vector address assignment, data word configuration, baud rate selection and communication line type. Specific details concerning configuring jumpers in each category are described under the topics that follow.

DEVICE ADDRESS ASSIGNMENT

SLU jumpers implementing device address assignment are designated A3 through A12 which correspond to bus address bits 03 through 12. These jumpers serve to hardwire the assigned device address on the specific SLU module. The format for a hardwired device address is shown in Figure 3.

A remote LSI-11 with a dual backplane configuration can control up to nine DLV11 SLU's with one dedicated to the host processor. Excluding an SLU assigned as a console terminal, all user SLU's will occupy any one of the 32 device address slots in the address space 164000 to 167770. (Note that each SLU requires four device addresses as indicated in Table 5.)

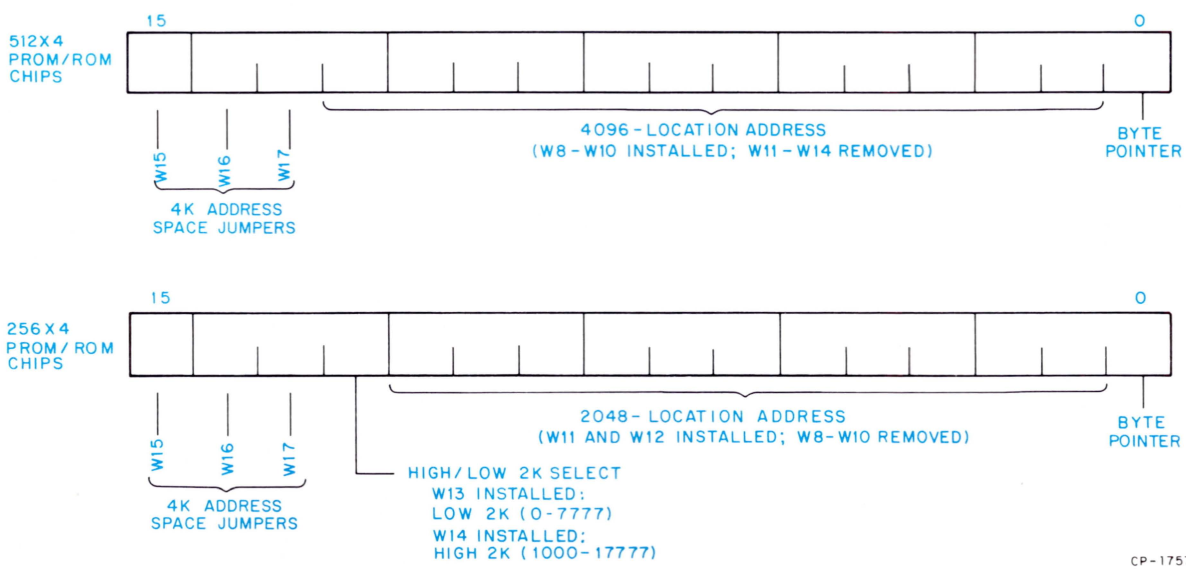


Figure 2 MRV11-AA Addressing

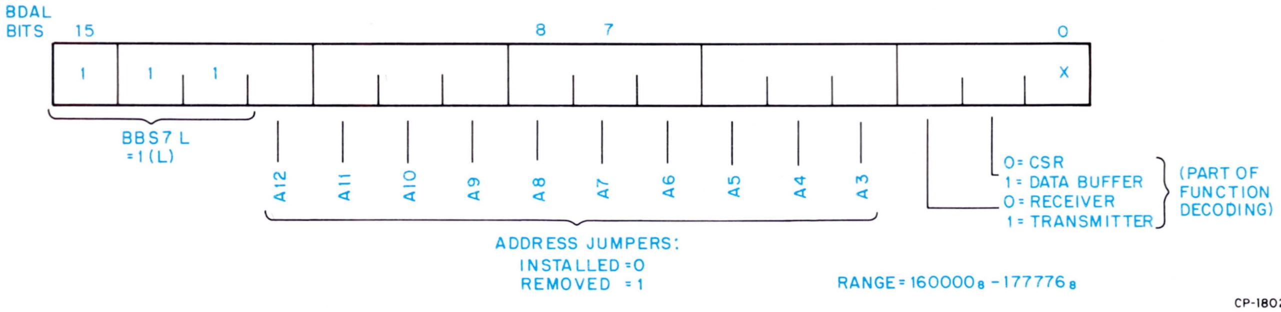


Figure 3 DLV11 Device Address Format

TABLE 5
DLV11 SLU FACTORY JUMPER CONFIGURATION AS A CONSOLE TERMINAL

Jumper Designation	Jumper State	Function Implemented
A3	I	This arrangement of jumpers A3 through A12 implements the octal device address 17756X, which is the assigned address for the console terminal SLU. The least significant digit is hardwired on the module to address the four SLU devices as follows: X = 0, RCSR address; X = 2, XCSR address; X = 4, Receive data register address; X = 6, Transmit data register address
A4	R	
A5	R	
A6	R	
A7	I	
A8	R	
A9	R	
A10	R	
A11	R	
A11	R	
A12	R	
V3	I	This factory set arrangement implements the interrupt vector addresses 60 for received data and 64 for transmitted data.
V4	R	
V5	R	
V6	I	
V7	I	
NP	R	No parity
2SB	R	Two stop bits
NB2	R	Eight data bits
NB1	R	
PEV	R	Even parity if NP installed
FEV	I	Halt on framing error. This jumper must be installed for the console device and removed for all other devices.
FR0	R	
FR1	R	110 baud rate selected
FR2	R	
FR3	R	
EIA	R	12 V EIA operation disabled
CL1	I	20 mA current loop active receiver and transmitter selected
CL2	I	
CL3	I	
CL4	I	

NOTE: R = Removed, I = Installed
Removed = 1, Installed = 0

INTERRUPT VECTOR ADDRESSES

The SLU jumpers implementing interrupt vector address assignments are designated V3 through V7 to correspond to the vector address field bits V3 through V7. These jumpers serve to hardwire the assigned vector address on the specific SLU module. The format for a hardwired vector address is shown in Figure 4.

Vector addresses are available for 32 DLV-11 SLU modules starting at the memory bank 0, locations 0 to 374 with two locations reserved for each vector.

DATA WORD CONFIGURATION

The configuration of the data word received and transmitted by a specific DLV11 SLU is programmed by a set of six jumpers on the module. Since the data word configuration for a given network application is determined by user requirements and the specific nature of the network, no specific recommendations concerning data word configuration can be made. However, on the DLV11 SLU connecting the console terminal jumper FEV must be installed on that module. In addition, jumper FEV must be removed on all other DLV11 SLU's configured with a remote LSI-11. The range of possible data word configurations is summarized in Table 6.

TABLE 6
DLV11 DATA WORD CONFIGURATION SUMMARY

Operating Function	Jumper Configuration	Function Implemented
Number of Data Bits	NB1 installed NB2 installed	5-bit data word
	NB1 removed NB2 installed	6-bit data word
	NB1 installed NB2 removed	7-bit data word
	NB1 removed NB2 removed	8-bit data word
Number of Stop Bits	2SB installed 2SB removed	One stop bit Two stop bits
Parity	NP removed	No parity bit issued or checked
	NP installed	Parity bit issued and checked
Framing Error	FEV installed FEV removed	Halt on framing error No halt on framing error

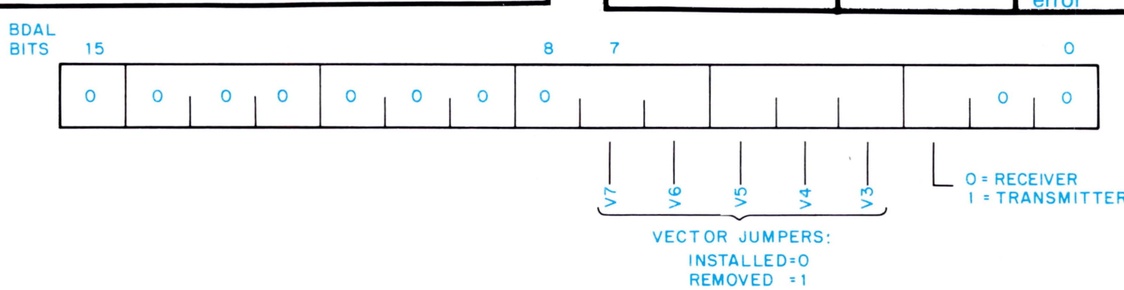


Figure 4 DLV111 Vector Address Format

BAUD RATE SELECTION

Four jumpers, FR0 through FR3 permit the user to select baud rates ranging from 50 to 9600 baud. Since baud rates are determined by a combination of user requirements and the specific peripheral devices used, it is not practical to specify baud rates for this generalized network application. However, the range of baud rates which can be implemented on the DLV11 SLU are summarized in Table 7.

TABLE 7
BAUD RATE JUMPER CONFIGURATION

Baud Rate	Jumper FR3	Jumper FR2	Jumper FR1	Jumper FR0
50	I	I	R	I
75	I	I	R	R
110	R	R	R	R
134.5	I	R	I	I
150	R	R	R	I
200	I	R	I	R
300	R	R	I	R
600	I	R	R	I
1200	R	I	R	R
1800	R	I	R	I
2400	I	R	R	R
2400	R	R	I	I
4800	R	I	I	R
9600	R	I	I	I
External (via pin BH1)	I	I	I	X
NOTE I = installed R = removed X = don't care				

COMMUNICATION LINE CONFIGURATION

A DLV11 SLU can be configured to interface to an EIA communications line or a 20 mA current loop communications line by jumpers CL1 through CL4. EIA interface drivers on the module are enabled when jumper EIA is installed. When a 20 mA current loop is selected, this jumper must be removed.

Jumpers CL1 through CL4 are configured for 20 mA current loop interface operation as follows:

1. Active Current Loop
   Transmit: CL3 and CL4 installed
   Receive: CL1 and CL2 installed
2. Passive Current Loop
   Transmit: CL4 and CL3 removed
   Receive: CL1 and CL2 removed

CONNECTION TO USER DEVICES

The DLV11 Serial Line Unit module connects to a user device through one of the two cable assemblies listed below:

EIA Interface	BC05C-25 Modem Cable
20 mA Current Loop	BC05M-01 Cable Assembly

The number at the end of the cable part number designates the length of the cable. For 20 mA current loop applications, the BC05M cable assembly is terminated with a Mate-N-Lock connector which is pin compatible with the following Digital Equipment Corporation Terminals:

LA36 DECwriter (read/write)
LA35 DECwriter (read only)
VT05B Alphanumeric Terminal
VT50 DECscope (12 line)
VT52 DECscope (24 line)
RT02 Alphanumeric Terminals
DF01-A Acoustic Telephone Coupler
LT33 Teletypewriter
LT35 Teletypewriter

The BC05C modem cable is used to connect the DLV11 through one integral 40-pin connector to an EIA RS232C specified communications line. The device end of this cable is terminated with a Cinch DB25P connector which is pin-to-pin compatible with Bell 103 and 113 modems.

CONFIGURING THE DRV11 PARALLEL LINE UNIT

The DRV11 Parallel Line Unit (PLU) is a general purpose parallel interface module permitting connection of a wide range of general and special purpose user I/O devices to the LSI-11. This module is designed to minimize interfacing problems through inclusion of all the buffering and interrupt control logic necessary to the connection of virtually any parallel I/O device.

The PLU is equipped with 15 jumpers which can be configured to select device addresses and interrupt vector addresses. In addition, the PLU module is equipped with split lugs for installation of an optional capacitor to adjust control pulse widths to the requirements of specific user devices. With reference to Table 8, the PLU is factory configured so that device and vector addresses reside in address space specifically reserved for user device interfaces.

TABLE 8
DRV11 PLU FACTORY JUMPER CONFIGURATION

Jumper Designation	Jumper State	Function Selected
A3 A4 A5 A6 A7 A8 A9 A10 A11 A12	R R R R R R R R R I	This arrangement of jumpers A3 through A12 assigns the device address 16777X to the PLU. This address is the starting address of a reserved block in memory bank 7 which is recommended for user device address assignments. The least significant digit X is hardwired on the module to implement the three PLU device addresses as follows: X = 0 CSR address X = 2 Output buffer address X = 4 Input buffer address
V3 V4 V5 V6 V7	I I I R R	This factory installed jumper configuration implements the two interrupt vector addresses 300 and 304 for use as defined by application requirements.
NOTE: R = Removed, I = Installed Removed = 1, Installed = 0		

DEVICE ADDRESS ASSIGNMENT

PLU jumpers implementing device address assignment are designated A3 through A12 which correspond to bus address bits 03 through 12. Those jumpers serve to hardwire the assigned device address directly on the PLU module. The format for a hardwired device address is shown in Figure 5.

A remote LSI-11 with a dual backplane configuration can control up to eight PLU modules, with one device slot being dedicated to a DLV11 SLU module for use in host processor/remote processor communication. All user PLU modules can occupy any one of 32 device address slots in the user assigned address space 167770 to 170770 (note that each PLU requires three device addresses as indicated in Table 8).

INTERRUPT VECTOR ADDRESSES

The PLU jumpers implementing interrupt vector address assignments are designated V3 through V7 to correspond to the vector address field bits V3 through V7. These jumpers serve to hardwire the assigned vector address on the specific PLU module. The format for the vector address field is shown in Figure 6. There are vector addresses available for PLU modules starting at the memory bank 0, locations 0 to 374, with two locations reserved for each vector.

The same vector addresses available to DLV11 SLU modules are also available for use by DRV11 PLU modules, and pose no conflict.

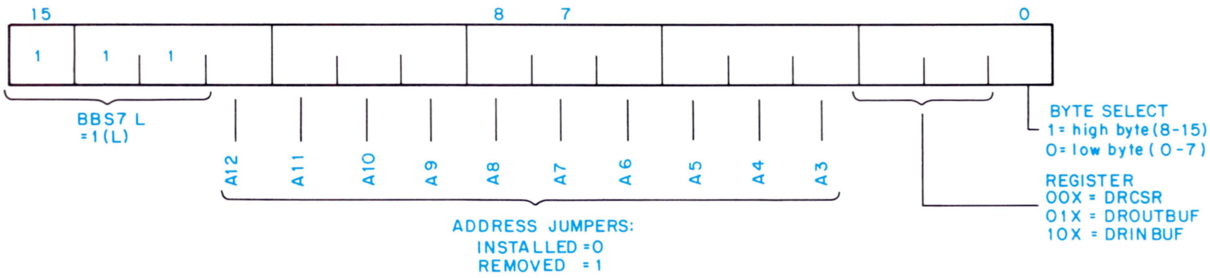


Figure 5 DRV11 Device Address Format

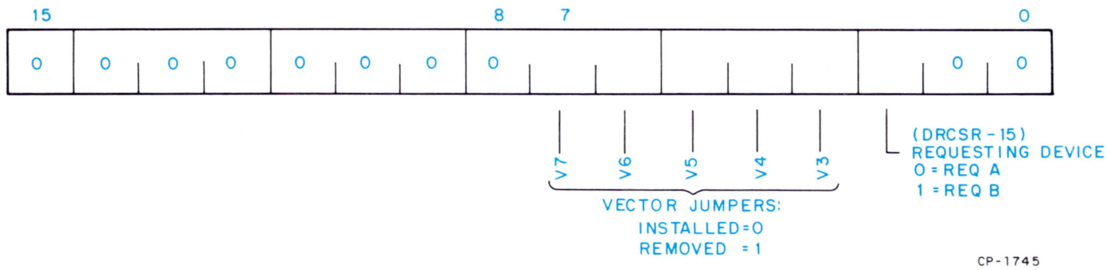


Figure 6 DRV11 Vector Address Format

CONTROL SIGNAL PULSE WIDTH MODIFICATION

A user can install an optional capacitor on the DRV11 module in order to extend to pulse width of both the NEW DATA READY and DATA TRANSMITTED pulses. Without the optional capacitor, the module, as shipped, will produce 300 nsec (nominal) pulses. The pulse widths obtained by various values of capacitance are listed in Table 9. An optional capacitor is installed on split lugs SL1 and SL2, located on the component side of the module PC board at the lower right hand corner, just above the fingers.

TABLE 9
DRV11 PULSE WIDTH MODIFICATION

Optional External Capacitance (pF)	Nominal Pulse Width (nsec)
None	300
1200	500
1800	600
6000	1200

CONNECTION TO USER DEVICES

The DRV11 Parallel Line Unit Module connects to a user device through two BC11K-X signal cables (the letter X designates one of five available lengths; 1, 6, 10, 20, and 25 feet). This cable is equipped with an H856 40-pin female connector on one end with the other end being terminated by the user.

CONFIGURING A REMOTE LSI-11 BACKPLANE

As shown in Figure 7, the LSI-11 system serving as the remote processor in this generalized network application can be obtained on either one or two H9270 wirewrap backplanes. With one backplane, a remote LSI-11 can be configured for an 8 K memory, an optional PROM module containing a user bootstrap, one DLV11 SLU dedicated to the host communication line, one DLV11 SLU dedicated to an optional console terminal, and depending on options, up to three DLV11 SLUs or DRV11 PLUs available for user I/O devices.

BACKPLANE CONNECTION AND TERMINATION

If a dual backplane configuration is chosen, the two backplanes must be connected by a BCV1B-X Bus Expander Cable/Terminator Assembly. I/O bus data and signals from and to the KD11-F on the primary backplane are available to all modules along the bus. On single backplane configurations the bus does not require termination and the slot occupied by a BCV1B-X assembly on multi-backplane configurations is available for use by a device interface module or an additional memory module. On dual backplane configurations, bus signals are propagated to the second backplane by a BCV1B-X assembly, and the bus must be terminated at the second backplane by a TEV11 Terminator Module.

POWER CONNECTION

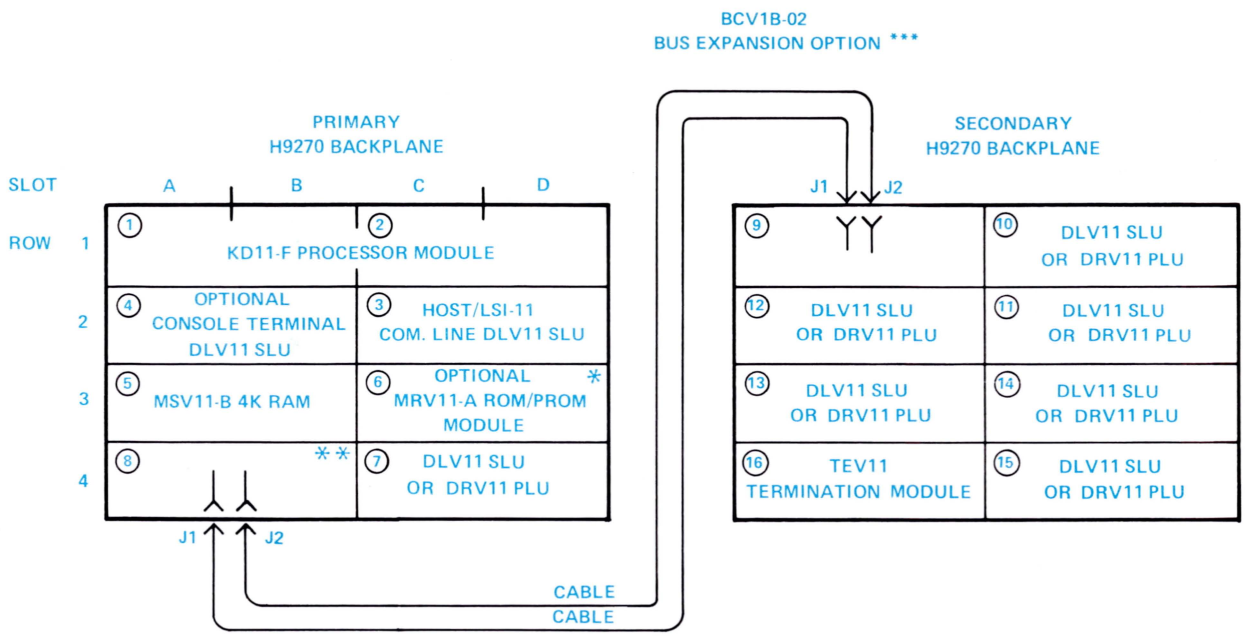
Power connections from a user provided power supply for both single and dual backplane configurations are shown in Figure 8. Wire sizes for power supply connection are also shown in Figure 8.

Figure 9 describes the connection of the externally generated power status bus signals, BDCOK H and BPOK H, as well as the line time clock signal BEVNT L and the halt signal BHALT L.

It is recommended that H9270 frame/castings be electrically connected to system/power supply ground.

DEVICE PRIORITY SELECTION

In LSI-11 systems, device priority is determined by the physical position of the device interface module on the bus and hence on the backplane. The closer the backplane slot occupied by a device interface module is to the processor module the higher the internal priority, and the further away, the lower the internal priority. Referring to Figure 7, the DLV11 SLU serving the host/remote communication line occupies the third backplane slot on the backplane, allowing that function the highest internal device priority. Similarly, the placement of the Console terminal DLV11 SLU in the fourth backplane slot provides that function with the second highest device priority. Note that memory modules are slave to the processor so that memory access is performed by the processor without priority arbitration. As a consequence, the position of RAM and PROM memory modules on the bus is not critical and has no effect on the priority of subsequent device interface modules.



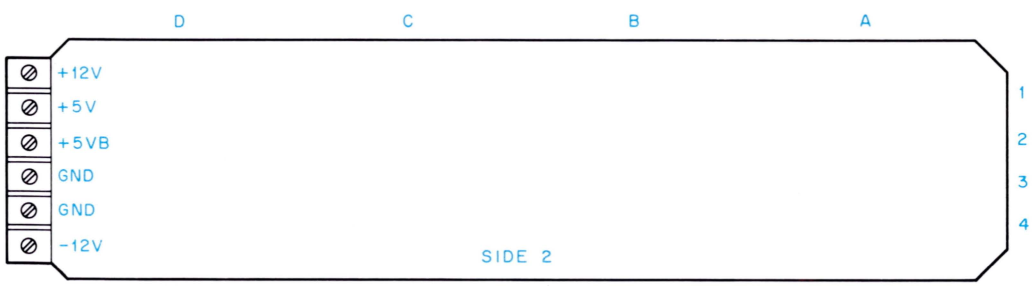
*This slot can be used for a DLV11 SLU or a DRV11 PLU when a terminal is included in a remote LSI-11 configuration.

**In single backplane configurations, this slot can be used by a DLV11 SLU, a DRV11 PLU, an MSV11-B RAM or an MRV11-AA PROM.

***This option includes two modules, shown installed in option slots 8 and 9, and two cables.

CP-2052

Figure 7 LSI-11 Backplane Configuration



CP-1762

Power Source (From)	H9270 Connector Block (To)		Recommended Wire Sizes
+ 12V	+ 12V		18 gauge
+ 5V	+ 5V	Connect	18 gauge 5-strand min.
	+ 5B		18 gauge 5-strand min.
GND	GND	Connect	18 gauge 6-strand min.
GND	GND		18 gauge 6-strand min.
-12V	-12V	(This voltage is not required. The connection is available for custom interfaces.)	18 gauge

Figure 8 H9270 Backplane Power Connections

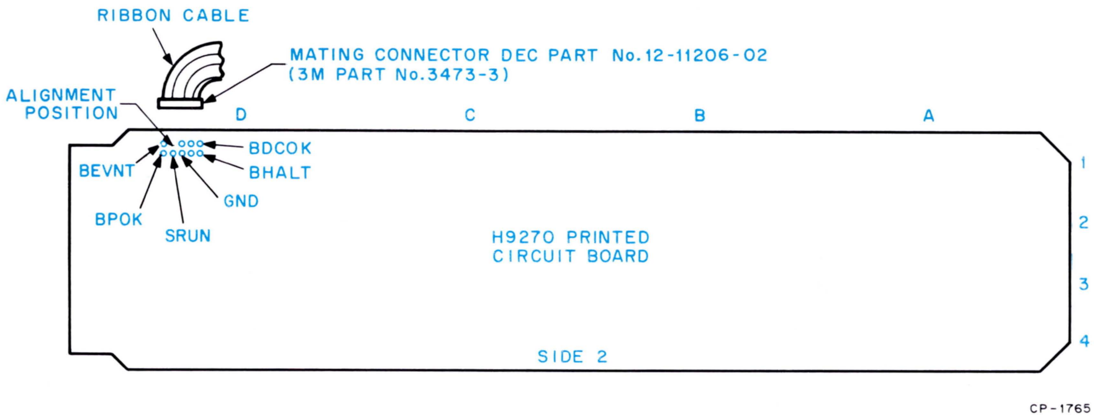


Figure 9 H9270 Signal Connections

MODULE INSERTION

The proper insertion orientation for LSI-11 modules is with the component side of each module facing backplane row 1 as shown in Figure 10.

CAUTION

All modules in an LSI-11 system can be installed or removed from the backplane only when power is removed from the backplane.

REFERENCE DOCUMENTS

Title	Document Number
LSI-11, PDP-11/03 Configuration and Installation Guide	EK-LSI11-IN-001
LSI-11, PDP-11/03 Processor Handbook	FB 04870 75 060/-09/02 25
LSI-11, PDP-11/03 User's Manual	EK-LSI11-TM-002
PDP-11 Peripheral Handbook - 75	2002 20175 4526RD 09 40
PDP-11/04/05/10/35/40/45 Processor Handbook - 1975-76	EB 05138 75 070/20 9 50
REMOTE-11 Reference Manual	Preliminary
RT-11 System Reference Manual	DEC-11-ORUGA-C-D

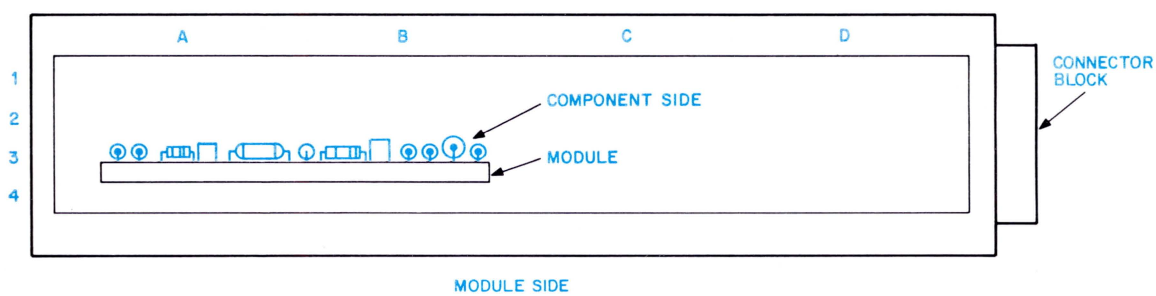


Figure 10 Module Installation

PRINTED IN U.S.A. ED 07154 53/76 11A 03 40

digital
DIGITAL EQUIPMENT CORPORATION
COMPONENTS GROUP HEADQUARTERS
ONE IRON WAY, MARLBOROUGH, MASSACHUSETTS 01752
(617) 481-7400 TWX: 710-347-0348
For detailed information about products and policies, call 800-225-9480 toll-free (USA only); Massachusetts residents call (617) 481-7400.