digital

RL11 CONTROLLER TECHNICAL DESCRIPTION MANUAL

EK-ORL11-TD-001

RL11 CONTROLLER
TECHNICAL DESCRIPTION
MANUAL

digital equipment corporation
colorado springs, colorado

First Edition, April 1980

The information in this manual is for informational purposes and is subject to change without notice.

Digital Equipment Corporation assumes no responsibility for any errors which may appear in this manual.

Printed in U.S.A.

The following are trademarks of Digital Equipment Corporation, Maynard, Massachusetts:

- DEC
- DECUS
- DECnet
- DECsystem-10
- DECSYSTEM-20
- DECwriter
- DIBOL
- Digital Logo
- EduSystem
- IAS
- LSI-11
- MASSBUS
- OMNIBUS
- PDP
- PDT
- Q-BUS
- RSTS
- RSX
- UNIBUS
- VAX
- VMX
- VT

CONTENTS

CHAPTER 1 INTRODUCTION

CHAPTER 2 USER INFORMATION

CONTENTS (Cont)

CHAPTER 3 INTERFACE LEVEL DESCRIPTION

CHAPTER 4 FUNCTIONAL DESCRIPTION

CONTENTS (Cont)

CHAPTER 5 COMMAND DESCRIPTIONS

APPENDIX A SPECIAL ICS

APPENDIX B READ-ONLY MEMORIES

FIGURES

Figure No. Title Page

CONTENTS (Cont)

TABLES

CHAPTER 1 INTRODUCTION

1.1 PURPOSE AND SCOPE OF MANUAL

This manual is a technical description of the RL11 Disk Controller. It is intended as an aid in understanding the functionality of the controller to a component level. Prerequisite for an understanding of this manual is a basic knowledge of PDP-11 processors, the UNIBUS, and basic disk principles.

1.2 RELATED DOCUMENTS

This manual is intended to be used in conjunction with the engineering drawings in the Field Maintenance Print Set (MP00153).

Table 1-1 lists documents that contain information necessary for a complete understanding of a disk subsystem that includes an RL11 Controller and RL01 or RL02 Disk Drive.

Table 1-1 RL11/RL01/RL02 Subsystem Documentation

RL01/RL02 Disk Subsystem User's Guide	EK-RL012-UG	(hard copy)
RL01/RL02 Pocket Service Guide	EK-RL012-PG	(hard copy)
RL01/RL02 Disk Drive Technical Manual	EP- RL012-TM	(microfiche)
PDP-11 Bus Handbook	EB-17525	(hard copy)
RL01/RL02 Preventive Maintenance Manual	EP-00008-PM	(microfiche)
RL01 Illustrated Parts Breakdown	EP-00016-IP	(microfiche)
RL02 Illustrated Parts Breakdown	EP-00016-IP	(microfiche)

1.3 DESCRIPTION OF SUBSYSTEM

Figure 1-1 is an overview of an RL11-based RL01/RL02 Subsystem.

An RL11 Subsystem is a mass storage subsystem with a capacity of 5 to 20 megabytes for use on a PDP-11 UNIBUS system. It consists of an RL11 Controller and one to four RL01 or RL02 Disk Drives. The RL11 Controller interfaces with the UNIBUS so that the PDP-11 Processor can access the addressable registers to check status and issue commands. The controller can also use the UNIBUS to perform direct memory access (DMA) data transfers between the system memory and the controller's data buffer.

The controller also interfaces with the drive I/O bus which links the controller with up to four disk drives in daisy chain fashion for the purpose of checking status, issuing commands, or transferring data.

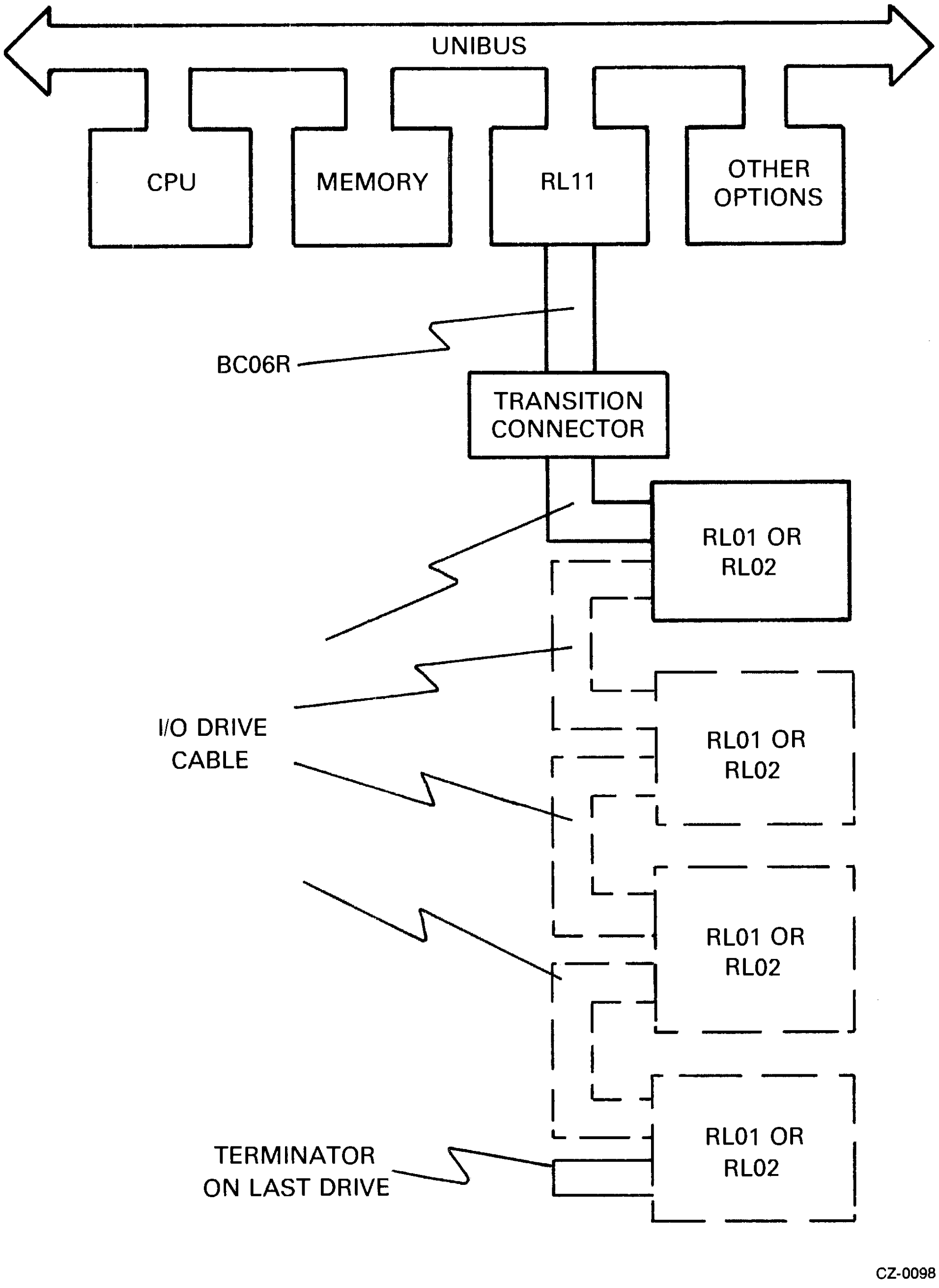


Figure 1-1 RL11-Based RL01/RL02 Disk Subsystem

1.4 DESCRIPTION OF MEDIA

The RL01K-DC Disk Cartridge is a single platter, top loading cartridge that is physically, but not functionally, interchangeable with an industry standard 5440 type cartridge. The RL02K-DC Disk Cartridge is the double track density version of the RL01K-DC. It is physically, but not functionally, interchangable with the RL01K-DC. The RL01K-DC has a formatted capacity of 5.2 megabytes and the RL02K-DC has a capacity of 10.4 megabytes. The RL01K-DC has 256 cylinders while the RL02K-DC has 512 cylinders. Both cartridges use both surfaces for data and use 40 sectors as defined by machined sector notches in the hub. There is no physical index notch in the hub.

Each sector contains the items listed below.

- Servo information that is read by the normal read/write head during sector pulse time while the head is not reading or writing data. This is referred to as "imbedded" servo information.

- Header information that consists of three words. The first contains the address of the surface, cylinder, and sector. The second is a word of all zeroes. The third is a 16-bit cyclic redundancy check (CRC).

- Data that consists of 256 8-bit bytes plus a 16-bit CRC word.

- Idle time until the next sector pulse. This time varies but is nominally 20 microseconds.

Figure 1-2 shows the layout of a sector.

The servo and header areas are factory-written and cannot be reformatted in the field. The data areas are, of course, written by the user.

The last track (last cylinder, last surface) is reserved for use as a bad sector file. Half of this file is reserved for factory-written entries and half for field written entries. This file also contains the cartridge serial number (in octal). Figure 1-3 shows the bad sector file.

The disk surface outside of the data area is prerecorded with unique servo patterns (guard bands) that indicate when the heads are not in the data area. In this case, the drive logic causes the head to move back into the data area (bounce off the guard band).

1.5 DESCRIPTION OF DISK DRIVE UNIT

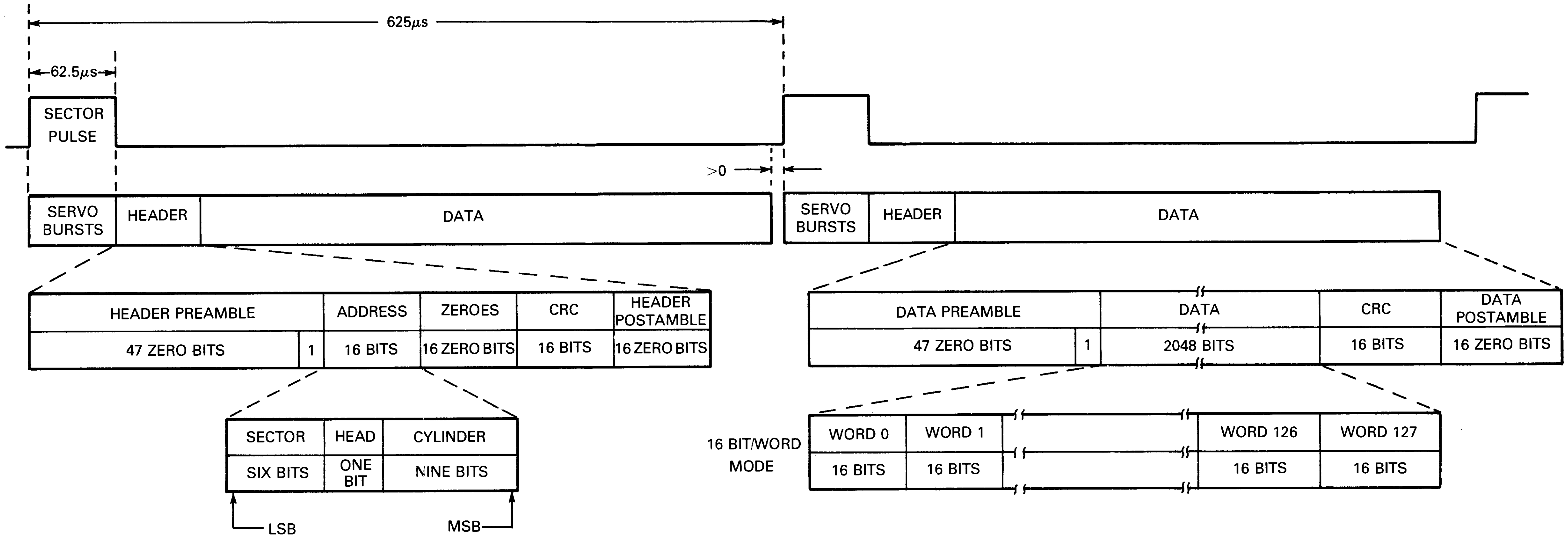
The RL01/RL02 Disk Drive is mounted on chassis slides and contains its own dc power supply; ac power is provided by a power cord. The drive contains two reversible terminal blocks (on the back of the drive) that are used for selection of an appropriate range of ac power. One block selects either 120 or 240 Vac and the other plug selects either low or nominal range. The frequency of the ac power can be either 50 or 60 Hz without any change to the drive.

The drive interfaces to the drive I/O bus through two connectors on the back of the drive that are connected in parallel with each other and also to the drive logic module. One is for I/O bus in, the other is for either I/O bus out or a terminator (if the drive is the last one on the bus). The drive uses the I/O bus to select the drive, receive commands or data from the controller, and to send status information or data to the controller.

On the front of the drive there is an operator control panel that consists of four push button switch/indicators. They are listed below.

- A load/unload switch with a LOAD indicator
- A unit number plug with a READY indicator
- A FAULT indicator
- A write protect switch with a WRITE PROTect indicator

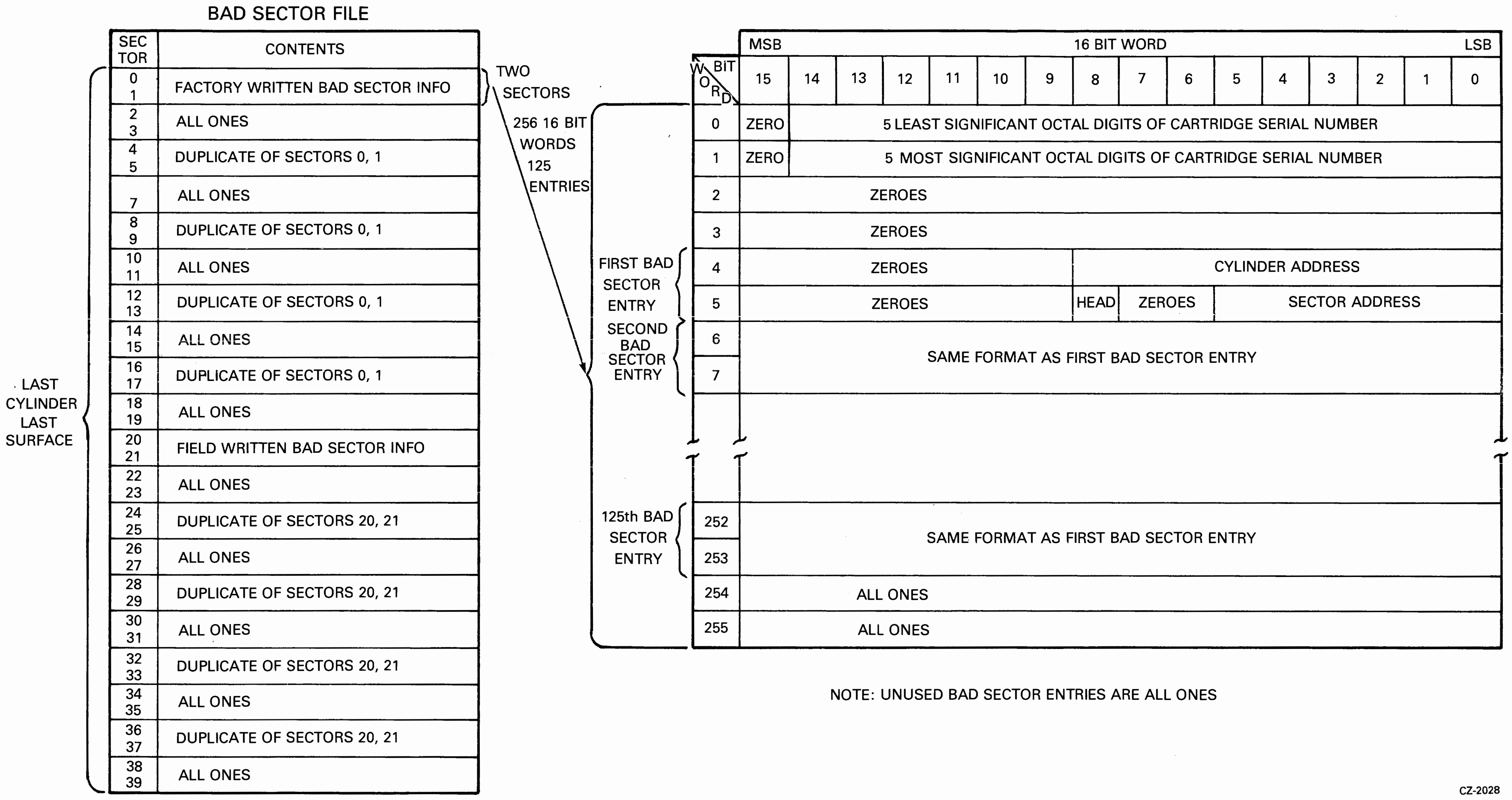
Each drive on the I/O bus compares the drive select bits (unit number) with the unit number of the plug on the operator panel. If they agree, the drive is selected which enables acceptance of signals from the controller as well as provides signals to the controller. The drive always monitors system clock and the ac low signal from the controller even without being selected.



NAME	MNEMONIC	LENGTH
SERVO BURSTS	SV	—
HEADER PREAMBLE	PR1	3 WORDS
HEADER	HDR	3 WORDS
HEADER POSTAMBLE	PO1	1 WORD
DATA PREAMBLE	PR2	3 WORDS
DATA	—	2048 BITS
DATA POSTAMBLE	PO2	1 WORD

CZ-0255

Figure 1-2 Disk Format



BAD SECTOR FILE

SEC TOR	CONTENTS
0 1	FACTORY WRITTEN BAD SECTOR INFO
2 3	ALL ONES
4 5	DUPLICATE OF SECTORS 0, 1
7	ALL ONES
8 9	DUPLICATE OF SECTORS 0, 1
10 11	ALL ONES
12 13	DUPLICATE OF SECTORS 0, 1
14 15	ALL ONES
16 17	DUPLICATE OF SECTORS 0, 1
18 19	ALL ONES
20 21	FIELD WRITTEN BAD SECTOR INFO
22 23	ALL ONES
24 25	DUPLICATE OF SECTORS 20, 21
26 27	ALL ONES
28 29	DUPLICATE OF SECTORS 20, 21
30 31	ALL ONES
32 33	DUPLICATE OF SECTORS 20, 21
34 35	ALL ONES
36 37	DUPLICATE OF SECTORS 20, 21
38 39	ALL ONES

	MSB							16 BIT WORD								LSB
WORD \ BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	ZERO	5 LEAST SIGNIFICANT OCTAL DIGITS OF CARTRIDGE SERIAL NUMBER														
1	ZERO	5 MOST SIGNIFICANT OCTAL DIGITS OF CARTRIDGE SERIAL NUMBER														
2	ZEROES															
3	ZEROES															
4	ZEROES							CYLINDER ADDRESS								
5	ZEROES							HEAD	ZEROES		SECTOR ADDRESS					
6	SAME FORMAT AS FIRST BAD SECTOR ENTRY															
7																
252	SAME FORMAT AS FIRST BAD SECTOR ENTRY															
253																
254	ALL ONES															
255	ALL ONES															

NOTE: UNUSED BAD SECTOR ENTRIES ARE ALL ONES

Figure 1-3 Bad Sector File

Once an operator has loaded a cartridge and pushed the LOAD switch, the drive spins the disk to a speed of 2400 r/min, performs a brush cycle, and loads the heads over the data area. The drive is now ready to receive commands from the controller. The two operations that it can perform are listed below.

1. Get status operation which causes the drive to provide status bits and status clock to the controller.
2. Seek operation which causes the drive to move the head positioner in the specified direction, the specified number of cylinders, and select the specified head.

Read and write operations are controlled entirely by the RL11 Controller and no command is sent to the drive.

1.6 DESCRIPTION OF CONTROLLER

The RL11 Controller consists of a single hex height module designated M7762. It is a Small Peripheral Controller (SPC) and therefore interfaces to the UNIBUS through backplane pins on rows C, D, E, and F using standard SPC pin assignments. See Paragraph 1.6.1 for more details.

The RL11 interconnects to the drive I/O bus through a 40 pin Berg connector on the module, a flat BC06R cable to a transition connector mounted on the cabinet, a round BC20J drive I/O cable to the first drive. BC20J cable is used down the daisy chain to the last drive which has a terminator instead of a cable out. The maximum cable length is 30 meters (100 feet). The maximum number of drives on the I/O bus is four.

The normal UNIBUS address is 77440X and the vector address is 160 (early modules were 330). These can be altered by changing jumpers on the module. The normal interrupt priority level is BR5. To select another level requires changing the priority plug.

The module is powered through the standard power pins on the backplane. It requires +5 Vdc ±5% at 5 amps, +15 Vdc ±10% at 0.5 amps, ±15 Vdc ± 10% at 0.5 amps.

This product is designed to operate in a class C environment as defined in DEC Standard 102. Highlights of this standard are shown in Table 1-2.

Table 1-2 Environmental Specifications

Operating Temperature:	5° C to 50° C (41° F to 122° F). The upper temperature limit is rated at sea level (760 mm Hg). This should be degraded at the rate of 1.8° C/1000 m (1° F/1000 ft) altitude.
Operating Humidity:	Less than 95%. Maximum wet bulb 32° C (90° F) and minimum dewpoint 2° C (36° F).
Non-operating Temperature:	-40° C to 66° C (-40° F to 151° F)
Non-operating Humidity:	Less than 95%
Operating Altitude:	Up to 2.4 km (8000 ft)
Non-operating Altitude:	Up to 9.1 km (30,000 ft)

1.6.1 SPC Considerations

The RL11 is an SPC but does not unconditionally fit into any SPC slot. Early SPCs were always quad-height modules or combinations of smaller (single or dual) modules that involved only four rows. Thus, the standard pin assignments applied only to rows C, D, E and F on a hex height backplane. Many newer options, such as the RL11, are hex-height modules and therefore require that rows A and B be vacant. Some SPC slots use rows A and B for UNIBUS cables or power connectors. Some hex-height options require standard UNIBUS pinning on rows A and B and some require modified UNIBUS device (MUD) pinning. In the case of the RL11, the only connections used on rows A and B are the +5v and ground. Thus, these rows can be either standard UNIBUS or MUD pinning.

The early SPCs did not utilize DMA data transfers to/from memory and therefore those signals were not part of the original SPC pin assignments. Some of the newer options, such as the RL11, do utilize DMA transfers. There is a new pin assignment called SPC PRIME that includes these signals. If the RL11 is to be used in an older (non-SPC PRIME) slot, then it is necessary to ensure that the following signals are wired on the backplane.

- Pin CA1 - NPG In
- Pin CB1 - NPG Out
- Pin FJ1 - NPR
- Pin CV1 - AC LO
- Pin CU1 - +15v

If the slot has SPC PRIME pinning, then another precaution must be taken. NPG continuity is maintained across an empty SPC PRIME slot by a backplane jumper from pin CA1 to pin CB1. This jumper must be removed whenever a DMA-type option is installed, such as an RL11, and the jumper must be added if the module is removed. This consideration is in addition to the normal Bus Grant Continuity card used in row D of all empty SPC slots.

1.6.2 Configuration Considerations

When configuring a UNIBUS system for the best priority assignments, two characteristics of a peripheral option must be taken into consideration. These are the peak word transfer rate and the T1 time (T1 time is a function of the peak transfer rate and the silo size). The RL11 has a peak transfer rate of 256 kHz (3.9 microseconds/word) and a T1 time of 62.4 microseconds. This dictates its position in the priority scheme. Table 1-3 shows the recommended priority scheme.

Table 1-3 UNIBUS Priority

- CPU
  - Memory
  - RK11/RK05
  - TM11/TU10
  - TC11/TU56
  - RL11/RL01-RL02
  - RJS04
  - RM02
  - RJP04
  - RK611/RK06-RK07
  - RJS03
  - TJU16
  - RF11/RS11
- DB11

er general configuration rules are listed below.

- On a PDP-11 UNIBUS, a combination of two disk subsystems and a tape or floppy disk subsystem is considered maximum.
- On a PDP-11/70 system, one UNIBUS disk subsystem is considered maximum if there are MASSBUS disks.
- A disk subsystem should not be installed beyond a bus expander.

CHAPTER 2
USER INFORMATION

2.1 INTRODUCTION

This chapter explains RL11 features that are visible to the programmer.

2.1.1 General

The RL11 Controller can perform one of eight functions when the CPU loads the command and status register of the controller. Each function is described below.

- Read Data – This function causes reads data from the disk and transfers it to the system memory.
- Read Data Without Header Check – This function has the same effect as a read function with the following exception. The transfer starts at the next sector, using the cylinder and head that are selected at the time, instead of waiting for a specified address.
- Read Header – This function reads the next header from the disk and transfers it to the controller where the CPU software can access it through one of the addressable registers.
- Write Data – This function transfers data from the system memory and writes it on the disk.
- Write Check – This function reads data from the disk and compares it with data read from the system memory for the purpose of verification.
- Seek – This function commands the drive to move the head positioner a specified number of cylinders, in a specified direction, and/or select a specified head.
- Get Status – This function commands the drive to transfer its status bits to the controller where the CPU software can access them to monitor the drive conditions.
- No-op – This function is a non-operation for an RL11 Controller. (It is used by other controllers to perform another function.)

Paragraph 2.2 describes these functions in more detail.

There are four addressable registers in the RL11 Controller that the CPU can access for the purpose of issuing commands and checking status. A basic explanation of each register is shown below.

- Control and Status Register (CSR) – The CSR (address 774400) contains various bits that are used by the CPU to command the controller and to monitor the status of the controller and/or the drive(s).
- Bus Address Register (BAR) – The BAR (address 774402) contains the UNIBUS address of the next word of the system memory to be accessed during data transfer functions.

- Disk Address Register (DAR) - The DAR (address 774404) contains the address of the desired area of the disk during data transfers. It has other uses during non-data transfer functions.

- Multipurpose Register (MPR) - The MPR (address 774406) contains status or header information that has been requested from the disk unit. It also holds the word count during data transfer functions.

Paragraph 2.3 describes these registers in more detail.

2.1.2 Implied Seeks

The RL11 Controller does not have the capability of performing an implied seek. Therefore, the CPU software must instruct the drive to position the heads over the desired cylinder and select the desired head by using the seek command prior to starting any data transfer function.

2.1.3 Spiral Seek

The RL11 Controller does not have the capability of performing a spiral (mid-transfer) seek. The hardware cannot initiate a seek to the next track or surface if the end of a track is reached before the end of the desired data range. The CPU software must do one of two things. Anticipate the situation and divide the transfer into alternate data transfers and seeks or be prepared to handle the error condition that will occur when the drive attempts to read or write past the end of track. If the drive attempts to read or write past the last sector on a track, the sector count in the DAR is advanced to 50 octal. Since there is no header with a sector address of 50, there will not be a match, and the operation will time out. The CPU software must recognize this situation, cause a seek to the next track, and initiate a resumption of the original data transfer function on the next track. This latter method is not the preferred method because of the long delay of the timeout.

2.1.4 Recalibrate

The RL11 Controller does not have a recalibrate (return head positioner to cylinder zero) function. To accomplish this function the CPU software should perform read header operation to determine the current position and then perform a seek operation back to cylinder zero.

2.1.5 Overlapped Seeks

Since the drive I/O bus can handle data transfers with only one drive at a time, the controller cannot overlap any operations except seeks. Seeks can be overlapped to a limit of all four drives seeking or one drive performing another operation while three other drives are seeking. Seek completion must be monitored by the CPU software.

2.1.6 Controller Ready

The controller ready (CRDY) condition is indicated by a flip-flop in the CSR (bit 7). When set, it indicates that the controller is ready to accept a command from the CPU. When reset, it indicates that the controller is busy performing a function and the registers should not be accessed. The CRDY is also used as a negative GO bit. When the software resets CRDY, the controller starts the function that is specified by the function bits in the CSR. When a function is terminated (either a normal termination or an error termination), the hardware sets the CRDY bit. The UNIBUS signal INIT sets the CRDY bit.

2.1.7 Interrupt

The RL11 Controller can interrupt the CPU by asserting BR5 bus request. When the request is honored, the controller provides a vector address of 160 (earlier versions were 330). The conditions that are required to assert the request are the operation being completed and interrupt enable (IE) set at the same time. IE is indicated by a flip-flop in the CSR (bit 6). IE is reset by the UNIBUS signal INIT. The CPU software sets the flip-flop to enable the interrupt capacility or resets the flip-flop to clear the interrupt capability.

2.1.8 Seek Interrupt

When a seek function is performed, the cylinder difference number and head address are transferred from the controller to the drive and the controller is free to perform another function while the drive performs the seek operation. The controller becomes ready and therefore interrupts (if IE is set) at this time. There is no end-of-seek interrupt to inform the CPU when the actual mechanical operation is completed. It is the responsibility of the CPU software to keep track of the seek operation that was initiated by periodically using the controller to check the drive unit ready bit of that drive. If the CPU software initiates another operation on a drive that is busy seeking, the controller will suspend the operation until the seek is completed.

2.1.9 Cyclic Redundancy Check (CRC)

A CRC checkword is generated by the controller and written on the disk following the data field every time a sector is written. As each sector is read, the CRC check is performed. Writing and checking of the CRC is invisible to the programmer except that when a CRC error is detected the programmer is notified. There is a CRC checkword appended to the end of each header when the disk is formatted at the factory. A header CRC check is performed after a header comparison if it was successful and after a read header.

2.2 FUNCTIONS

The eight functions are explained below.

2.2.1 Read Data

Prior to commanding the controller to perform the read data function, the CPU software should ensure that the heads are over the desired cylinder and that the desired head is selected.

The CPU software should then load the

- BAR with the starting address of system memory
- DAR with the starting address of the disk
- MPR with the number of words to be transferred (in two's complement)

Then, as the CPU software is loading the CSR with a function code of 5 for a write data operation, it should also load the drive unit number, the memory address extension bits, the IE bit (if desired), and reset the CRDY bit. Resetting the CRDY bit starts the operation.

The controller begins the operation by reading headers and looking for a match between the header and the contents of the DAR. The header CRC check is performed if a match occurs, and any error is flagged. Once a match is found, the data transfer starts. Data is transferred in bit serial fashion over the drive I/O bus into the controller where it is converted to parallel (16-bit words) and inserted into a sixteen word silo buffer. The data bubbles to the highest available level where it causes the logic to request use of the UNIBUS to perform non-processor request (NPR) data transfers. As each word is transferred from the controller silo across the UNIBUS to the system memory (or other destination) the address in the BAR is incremented by one word and the negative word count in the MPR is incremented. The reading of data continues until word count reaches zero.

If the end of the sector occurs before the word count reaches zero, the DAR is incremented and the operation is temporarily suspended until the next sector appears and, assuming that the header and the DAR match, the transfer is resumed.

If the word count reaches zero before the end of the sector, the data transfer into the silo is terminated but the data continues to be read until the end of the sector so that the CRC check can be performed.

As data is read into the silo it also generates a CRC checkword. At the end of each sector, the CRC checkword on the disk is read and compared with the generated CRC checkword. If the two CRC checkwords do not match, the CRC error bit is set.

2.2.2 Read Data Without Header Check

This function performs the same operation as the read function except that no comparison is made between the address in the header and the address in the DAR prior to starting the data transfer. Thus the transfer of data starts with the first word of the next sector that appears under the selected head. The prerequisites for this function are the same as the prerequisites for the read function except that it is not necessary to load the DAR. The function code is 7.

2.2.3 Read Header

Prior to commanding the controller to perform the read header function, the CPU software should load the CSR with the desired drive unit bits, a function code of 4, the IE bit (if desired), and the negative GO bit.

This function causes the next header that appears under the selected head to be transferred to the silo buffer in the controller where it bubbles to the top of the silo. This header information does not cause an NPR request as data would, but instead sets CRDY. The first word is available to the CPU when it accesses the MPR. When the CPU takes the first word out of the MPR, the second header word enters the MPR. After the CPU takes the second word out of the MPR, the third header word enters the MPR.

2.2.4 Write Data

Prior to commanding the controller to perform the write data function, the CPU software should ensure that the heads are over the desired cylinder and that the desired head is selected. The software should then load the BAR, DAR, AND MPR.

Then, as the CPU software is loading the CSR with a function code of 5 for a write data operation, it should also load the drive number, the memory address extension bits, the IE bit (if desired), and reset the controller ready (CRDY) bit. Resetting the CRDY bit starts the operation.

The controller begins the write operation by reading headers and looking for a match between the header and the contents of the DAR. The header CRC check is performed when a match occurs and any error is flagged. During the header search the controller is requesting data from system memory using NPR transfers and filling the silo buffer with data. When the buffer has been filled with sixteen words, the NPR transfers are suspended until data is transferred out of the buffer to the disk. When a header comparison has been found, the data transfer starts. Data is transferred in bit serial fashion over the drive I/O bus from the controller and written on the disk.

As each word of data is transferred over the UNIBUS, the BAR is incremented by one word and the MPR is incremented toward zero. Writing continues until the count in the MPR reaches zero which indicates the end of the transfer.

If the end of sector is reached and the word count has not reached zero yet, the data transfer is suspended until the next sector.

If the word count reaches zero in the middle of a sector, the remainder of the sector is written with all zero data.

As each word is transferred to the disk, the controller generates a CRC checkword and at the end of each sector the CRC checkword is written on the disk. No CRC check can be performed on the data area during a write operation.

2.2.5 Write Check

This function compares data from system memory with data read from the disk. This is done to verify that data previously written on the disk with a write function is correct (agrees with memory).

Prior to commanding the controller to perform a write check function the CPU software should ensure that the heads are over the desired cylinder and the desired head is selected. The software should then load the BAR with the starting address of system memory, load the DAR with the starting disk address, load the MPR with the word count, and load the CSR with the drive unit bits, the high order bus address bits, the IE bit, a function code of 1, and the negative GO bit.

The controller begins the write check operation by reading headers and looking for a match between each header and the contents of the DAR. A header CRC check performed is when a match occurs and errors flagged. During the header search the controller is also requesting (NPR) words from system memory and filling up the silo buffer. When a header match is found, data is read from the disk and compared with data from the silo. Any bits that do not match set bit 11 in the CSR to flag the error. A CRC check is performed on the data read.

As each word is transferred from the UNIBUS, the BAR and the MPR are incremented. The write check operation continues until the word count reaches zero.

After the operation has finished as signified by CRDY being set, the CPU software should check to see if the data compared or not by examining CSR bit 11.

2.2.6 Seek

Prior to issuing the seek command, the CPU software should determine the location of the positioner by either calculating the location or by performing a read header operation. The software should then determine if a seek operation is needed to position the heads to the desired location.

The software can then load the DAR with the cylinder difference number, the direction bit, the head selection bits, and the marker bit (bit 00). The CSR should then be loaded with the drive unit bits, the IE bit, a function code of 3, and the negative GO bit.

The controller sends a seek command to the drive and sets the CRDY bit. The controller is now ready to accept another command even though the seek operation is still in process in the drive.

2.2.7 Get Status

Prior to commanding this operation, the CPU software should load the DAR bits 01:00 and bit 03 (reset bit, if desired), and then load the CSR with the drive unit number, the IE bit, a function code of 2, and the negative GO bit.

The controller sends a get status command to the desired drive. The drive sends the status bits back to the controller where they go into the silo buffer. The CRDY bit is set to indicate that the CPU software can access the status word by addressing the MPR.

If bit 3 of the DAR was set when the operation started, the error conditions in the drive unit are reset before status is gathered.

The status bits are explained in Paragraph 2.3.4.1.

2.2.8 No-op

The function code of 0 in the CSR of an RL11 Controller when the CRDY bit is reset causes the controller to perform an operation that clears the controller error flip-flop and sets CRDY.

2.3 ADDRESSABLE REGISTERS

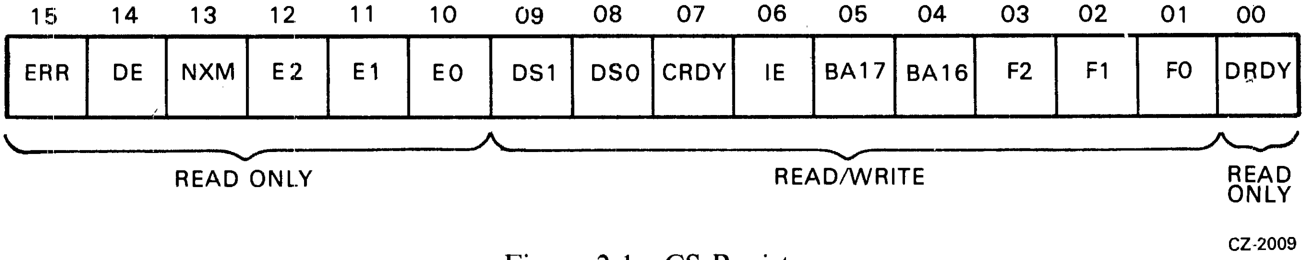
The four addressable registers are illustrated and explained below.

2.3.1 Control Status Register

The Control Status (CS) register (Figure 2-1) is a 16-bit word addressable register with an address of 774400. Bits 1 through 9 can be read or written; the other bits can only be read.

When the controller is initialized, bits 1-6 and 8-13 are cleared and bit 7 is set. Bit 0 is set whenever the selected drive is ready; otherwise, the bit is cleared. Bit 14 is set whenever there is a drive error. It is cleared when the drive error is corrected or the drive error is cleared by a get status command. Bit 15 is set when there is a drive or controller error (indicated in bits 10-14).

CONTROL STATUS REGISTER (CSR)



15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
ERR	DE	NXM	E2	E1	E0	DS1	DS0	CRDY	IE	BA17	BA16	F2	F1	F0	DRDY

Figure 2-1 CS Register

Bit(s)	Name	Function
0	Drive Ready (DRDY)	This bit is tied directly to the drive ready signal from the selected drive. When set, it indicates that the drive is ready to read/write/seek.
1-3	Function Code	These bits are set by software to indicate the command to be executed. Command execution requires that bit 7 (controller ready) be cleared by software. A zero bit being transferred into bit 7 of the CSR can be considered as a Go bit.

Function Codes

F2	F1	F0	Command Code	Octal
0	0	0	No-Op	0
0	0	1	Write Check	1
0	1	0	Get Status	2
0	1	1	Seek	3
1	0	0	Read Header	4
1	0	1	Write Data	5
1	1	0	Read Data	6
1	1	1	Read Data Without Header Check	7

Bit(s)	Name	Function
4-5	Bus Address Extension Bits (BA16, BA17)	The two most significant bus address bits. Read and written as data bits 4 and 5 of the CS register but considered as address bits 16 and 17 of the bus address register (see Paragraph 2.3.2).
6	Interrupt Enable (IE)	When this bit is set by software, the controller is allowed to interrupt the processor at the command completion or error termination.
7	Controller Ready (CRDY)	When cleared by software, this bit indicates that the command in bits 1-3 is to be executed. When set, this bit indicates that the controller is ready to accept a command.
8-9	Drive Select (DS0, DS1)	These bits select the drive that will communicate with the controller via the drive I/O bus.
10	Operation Incomplete (OPI)	When set, this bit indicates that the current command was not completed within 200 milliseconds.
11	Data CRC (DCRC) or Header CRC (HCRC) or Write Check (WCE)	If OPI (bit 10) is cleared and this bit is set, a CRC error has occurred when reading the data (DCRC). If OPI (bit 10) is set and bit 11 is also set, the CRC error has occurred on the header (HCRC). If OPI (bit 10) is cleared and bit 11 is set and the function command was a write check, a write check error (WCE) has occurred. If the function command was not a write check, the error condition was a data CRC error.

<table>
<tr><th>Bit(s)</th><th>Name</th><th>Function</th></tr>
<tr><td>12</td><td>Data Late (DLT) or Header Not Found (HNF)</td><td>This bit is set during a write when the silo is empty but the word count has not yet reached zero (meaning that the bus request was ignored for too long). The OPI bit will not be set.
This bit will be set during a read when the silo is full (meaning that the word being read could not enter the silo and the bus request has been ignored for too long). The OPI bit will not be set.
When this bit and OPI are both set, a 200 millisecond timeout occurred while the controller was searching for the correct sector to read or write (no header compare - HNF).
<table>
<tr><th colspan="3">Bits</th><th>Error</th></tr>
<tr><th>12</th><th>11</th><th>10</th><th></th></tr>
<tr><td>0</td><td>0</td><td>1</td><td>OPI</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>DCRC or WCE</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>HCRC</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>DLT</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>HNF</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>DLT and DCRC or WCE</td></tr>
</table></td></tr>
<tr><td>13</td><td>Non-Existent Memory (NXM)</td><td>This bit is set when the addressed memory does not respond within 20 microseconds of the beginning of a direct memory access (DMA) data transfer.</td></tr>
<tr><td>14</td><td>Drive Error (DE)</td><td>This bit is tied directly to the DE signal on the I/O drive bus. When set, it indicates that the selected drive has flagged an error. (The source of the error can be determined by executing a get status command without reset.)</td></tr>
<tr><td>15</td><td>Composite Error (ERR)</td><td>DE can be cleared by executing a get status command with bit 3 (reset bit) of the DA register set.</td></tr>
</table>

140,201
13 9 6 210

2.3.2 Bus Address Register

The bus address (BA) register (Figure 2-2) is a 16-bit register with an address of 774402. Bits 1 through 15 can be read or written; bit 0 is non-existent since the RL11 is word rather than byte oriented. Bus address bits 16 and 17 are contained in bits 4 and 5 of the CS register (see Paragrapn 2.3.1).

The BA register indicates the memory location involved in the data transfer during a read or write operation. The contents of the BA register are automatically incremented as each word is transferred between the bus and the I/O buffer. This register overflows into CS register bits 4 and 5.

The BA register is cleared by initializing or by loading the register with zeros.

BUS ADDRESS REGISTER (BAR)

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8	BA7	BA6	BA5	BA4	BA3	BA2	BA1	0

READ/WRITE

CZ-2035

Figure 2-2 BA Register

Bit(s)	Name	Function
1-15	BA1 thru BA15	These bits point to the UNIBUS address that data is to be transferred to/from (normally a memory address). BA16 and BA17 are CSR bits 4 and 5.

2.3.3 Disk Address Register

The disk address (DA) register is a 16-bit register with an address of 774404. It is used during three of the operations. During a read/write operation it holds the disk address. During the seek operation and during the get status operation it holds the drive command. This register is cleared by initializing the controller or loading the register with zeros. All 16 bits can be read or written by the processor.

2.3.3.1 DA Register During Read or Write Data Command – For a read or write operation, the DA register is loaded with the address of the first sector to be transferred. As each successive sector is transferred, the DA register is automatically incremented (Figure 2-3).

DAR DURING READING OR WRITING DATA COMMANDS

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CA8	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0	HS	SA5	SA4	SA3	SA2	SA1	SA0

CZ-2011

Figure 2-3 DAR – Read/Write Data Command

Bit(s)	Name	Function
0-5	Sector Address SA 05:00	Address of one of the 40 sectors on a track.

Bit(s)	Name	Function
6	Head Select (HS)	Indicates which head (disk surface) is desired. A one indicates the lower head; a zero, the upper head.
7-15	Cylinder Address CA 08:00	Address of the cylinders being accessed. (Range is 0 through 777, octal)

2.3.3.2 DA Register During a Seek Command – To perform a seek function, it is necessary to provide cylinder address difference, head select, and movement direction information to the selected drive as indicated (Figure 2-4).

DAR DURING SEEK COMMAND

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
DF8	DF7	DF6	DF5	DF4	DF3	DF2	DF1	DF0	0	0	HS	0	DIR	0	1

CZ-2010

Figure 2-4 DAR – Seek Command

Bit(s)	Name	Function
0	Marker	Must be a 1.
1	–	Must be a 0.
2	Direction (DIR)	This bit indicates the direction in which a seek is to take place. When the bit is set, the heads move toward the spindle (to a higher cylinder address). When the bit is cleared, the heads move away from the spindle (to a lower cylinder address). The actual distance moved depends on the cylinder address difference (bits 7-15).
3	–	Must be a 0.
4	Head Select (HS)	Indicates which head (disk surface) to select. A one indicates the lower head; a zero, the upper head.
5-6	–	Reserved.
7-15	Cylinder Address Difference DF 08:00	Indicates the number of cylinders the heads are to move on a seek.

2.3.3.3 DA Register During a Get Status Command – For a get status command, the DA register bits must be programmed as follows (Figure 2-5).

DAR DURING GET STATUS COMMAND

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
X	X	X	X	X	X	X	X	0	0	0	0	RST	0	1	1

CZ-2037

Figure 2-5 DAR – Get Status Command

Bit(s)	Name	Function
0	Marker	Must be a 1.
1	Get Status (GS)	Must be a 1, indicating to the drive that the status word is being requested. At the completion of the get status command, the drive status word is available to the CPU by accessing the multipurpose (MP) register (see Paragraph 2.3.4.1).
2	–	Must be a 0.
3	Reset (RST)	When this bit is set, the drive clears its error register before sending a status word to the controller.
4-15	–	Must be zeros.

2.3.4 Multipurpose Register

The multipurpose (MP) register is the name associated with the address 774406. It is not an actual hardware register. When the CPU accesses the MPR as a source, the output of the silo buffer (data buffer out) is obtained. When the CPU accesses the MPR as a destination, the receiver is the word count register which is not condisered an addressable register because the CPU cannot call it as a source. The MPR uses are explained below.

2.3.4.1 MP Register After a Get Status Command – When a get status command is executed, the status word is returned to the controller and transferred to the MP register (Figure 2-6). In this case, the contents of the MP register are defined as follows.

MPR AFTER GET STATUS COMMAND

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
WDE	CHE	WL	SKTO	SPE	WGE	VC	DSE	DT	HS	CO	HO	BH	STC	STB	STA

CZ-2012

Figure 2-6 MPR – Status Word

Bit(s)	Name	Function
0-2	ST C:A	These bits define the state of the drive. C B A 0 0 0 Load Cartridge 0 0 1 Spin-up 0 1 0 Brush Cycle 0 1 1 Load Heads 1 0 0 Seek 1 0 1 Lock On 1 1 0 Unload Heads 1 1 1 Spin-down
3	Brush Home (BH)	Set when the brushes are home.
4	Heads Out (HO)	Set when the heads are over the disk.
5	Cover Open (CO)	Set when the cartridge access cover is open or the cartridge dust cover is not in place.
6	Head Select (HS)	Indicates the currently selected head. A zero indicates the upper head; a one, the lower head.
7	Drive Type (DT)	A zero indicates an RL01; a one, an RL02.
8	Drive Select Error (DSE)	Set when a multiple drive selection is detected.
9	Volume Check (VC)	Set during cartridge spin-up. Cleared by execution of a get status command with bit 3 (reset bit) asserted. It is used as an indicator to the CPU software that a cartridge may have been mounted since the last time the drive was used.
10	Write Gate Error (WGE)	Set during write gate if one or more of the following conditions occur. • Drive is not "Ready to Read/Write" • Drive is write protected • Sector pulse is occurring • Drive has another error
11	Spin Error (SPE)	Set when spindle has not reached the proper speed in the required time during spin-up or when spindle speed is too high.

Bit(s)	Name	Function
12	Seek Time Out (SKTO)	Set when the heads do not come on track in the required time during a seek command or when "Ready to Read/Write" is lost while the drive is in position (lock-on) mode.
13	Write Lock (WL)	Set when the drive write protect switch is in.
14	Current In Head Error (CHE)	Set if write current is detected in the heads when write gate is not asserted.
15	Write Data Error (WDE)	Set if write gate is asserted but no write data transitions are received within the required time.

2.3.4.2 MP Register After a Read Header Command – When a read header command is executed, the next header will be read and its three words will be stored in the silo data buffer. The first word contains sector address, head select, and cylinder address information. The second word contains all zeros. The third word contains the header CRC checkword. All three words can be read sequentially by the CPU software (Figure 2-7).

MPR AFTER READ HEADER COMMAND

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CA8	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0	HS	SA5	SA4	SA3	SA2	SA1	SA0

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
ZEROES															

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CRC															

CZ-2013

Figure 2-7 MPR – Three Header Words

Bit(s)	Name	Function
0-5	SA	Sector Address
6	HS	Head Select
7-15	CA	Cylinder Address

2.3.4.3 MP Register During Read/Write Data Commands – Before the reading or writing data, the program should load the word count into the MP register in two's complement form. The counter is incremented as each word is transferred. A read or write operation is terminated when the word counter reaches zero (overflows). The word counter can keep track of any number of data words, from one to the full 40-sector count of 5120 words (Figure 2-8).

MPR DURING READ/WRITE COMMANDS FOR WORD COUNT

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
1	1	1	WC12	WC11	WC10	WC9	WC8	WC7	WC6	WC5	WC4	WC3	WC2	WC1	WC0

CZ-2036

Figure 2-8 MPR – Used as a Word Counter

Bit(s)	Name	Function
0-12	Word Count WC 12:00	Contains the two's complement of total number of words to be transferred.
13-15		Must be ones.

MP Register Programming Note – The RL01/RL02 Disk Drive will not do spiral read/writes without software intervention. If data is to be transferred past the end of the last sector of a track, it is necessary to break up the operation into the following steps.

1. **Program the data transfer to terminate at the end of the last sector of the track.**

2. **Program a seek to the next track. This can be either a head switch from the upper to the lower surface on the same cylinder or a head switch from lower to upper surface and move to the next cylinder.**

3. **Program the data transfer to continue at the start of the first sector on the next track.**

CHAPTER 3
INTERFACE LEVEL DESCRIPTION

3.1 INTRODUCTION

This chapter describes the controller logic that drives and responds to the two interfaces. One is the interface to the UNIBUS, the other is the interface to the drive I/O bus. Since this chapter is a guide through the engineering drawings, a brief description of the signal naming conventions used on the circuit schematic (CS) drawings is provided.

The RL11 is a hex-height module (M7762) that interfaces with the UNIBUS through backplane pins on rows C, D, E, and F. A backplane pin is represented on a CS drawing as an arrowhead with a three character designator that indicates the geography of the backplane row/pin/side. The mnemonic for a UNIBUS signal starts with a UB, whereas a signal that is generated by the controller logic starts with the page number of the sheet where the signal originates. The number(s) in the brackets that follow the mnemonic indicates the destination page(s) of the signal. Each page is laid out in coordinates 8 through 1 across the top and D through A down the side.

An example of this can be seen by referring to CS-M7762, Sheet 1 of 11 at coordinates 6-D. There is an arrowhead signal line with EV2 above it and the mnemonic UB A03 L. This refers to pin EV2 and the UNIBUS signal "Address 03 asserted low".

3.2 UNIBUS INTERFACE

The RL11 Controller responds to or drives the UNIBUS in three different situations. One situation is the RL11 responding as a slave to the CPU (master). The second is the RL11 acting as a bus master during a BR interrupt. The third situation is the RL11 acting as a bus master during an NPR transfer. These situations are explained below.

3.2.1 RL11 as a Slave to a CPU Access

The RL11 continuously monitors the UNIBUS address lines and when MSYN occurs (P1, 6-D, E84-13), a comparison is made between the UNIBUS address lines and the RL11 address as defined by jumpers W7-16. (See Table 3-1 for address jumper designations.) The address lines A02:00 are not compared when looking for a match condition. The internal logic of the DC005 and 8641 chips are explained in Appendix A.

If a match occurs, the signal P1 DEV SEL H is generated (P1, 5-D), delayed (P2, 8-BA), and SSYN is asserted (P1, 2-B, E26) as the response that is required of a UNIBUS slave. On Page 2, the signals C1 and A 02:01 are used (P2, 6-AB and 4-AB) to select one of four registers and to select a direction for data flow, either from the UNIBUS to the tristate bus or from the tristate bus to the UNIBUS. The tristate bus connects to the various registers to complete the data path. (C0 is not examined because the RL11 is word oriented and DATIP is not used.) Note that, although the programmer accesses the "MPR" at address 774406, there is no one hardware register called the MPR. When 774406 is used as a destination (as preparation for a read/write function) the information is loaded into a word counter. When 774406 is used as a source (to get the results of a read header or get status function) the information comes from the data output of the silo.

Table 3-1 Address Jumpers

Jumper	Address Bit	Normal Address of 774400
W 7	8	In
W 8	7	Out
W 9 I	6	Out
W 10 I	4	Out
W 11 I	5	Out
W 12	12	In
W 13	3	Out
W 14	9	Out
W 15	10	Out
W 16.	11	In

NOTE

A jumper is in for a one. The physical locations of the jumpers are shown on Print UA M7762, Sheet 1 of 6.

3.2.2 RL11 as a Master During an Interrupt

When an operation has been completed and interrupt is enabled (IE is set), the controller will interrupt the CPU software by asserting a Bus Request (BR).

The controller generates the signal P6 REQ INTR L that activates the request input to the BR chip (P2, 6-C, E79-1). The internal logic of this type of chip (8647) is shown in Appendix A. This chip produces the signal P2 BR L (P2, 5-C, E79-9) that is routed to the socket for the priority plug (P2, 2-D, E59). Normally there is a BR5 plug installed in the socket which routes the BR signal out on the UB BR5 L to assert the UNIBUS BR5 line (P2, 1-C, E59-14). The CPU arbitrator responds to the UB BR5 L signal with a UB BG5 H (P2, 2-D, E59-7) which is routed through the priority plug and on to the BR chip as P2 BG IN H (P2, 6-C, E79-5). The BR chip responds by generating P2 SACK H (or P2 BG OUT H if this controller was not the one making the request) which goes to the UNIBUS as UB SACK L (P1, 2-C, E15-1). The arbitration cycle is now over and requesting and granting can cease because SACK holds this controller to be the next UNIBUS master.

When the current bus master relinquishes ownership of the bus by dropping UB BBSY L (P2, 5-B, E79-10), the BR chip asserts UB BBSY to assume ownership of the bus. The BR chip also generates P2 ENA INTR VECT H which clears the request flip-flop, asserts UB INTR L (P1, 2-B, E26-13), and asserts UB D07:02 L (p1,4-B,E49,48) through jumpers W1-6 to provide vector information. (See Table 3-2 for vector jumpers.) When the CPU responds to UB INTR L with UB SSYN L (P1, 2-C, E26-15) the signal P1 INT SSYN H is produced which goes to the BR chip (P2, 6-C, E79-2) to clear the BBSY and INTR VECT signals. At this point the transaction is complete.

Table 3-2 Vector Jumpers

Jumper	D Bit	Normal Vector 160
W 1	7	Out
W 2	2	Out
W 3	6	In
W 4	5	In
W 5	4	In
W 6	3	Out

NOTE
A jumper is in for a one. The physical locations of the jumpers are shown on Print UA M7762, Sheet 1 of 6.

3.2.3 RL11 as a Master During a Direct Memory Access

When the controller logic determines that it is time to transfer data from the silo to the system memory (or vice versa), it requests use of the UNIBUS for a direct memory access (DMA) by asserting UB NPR L (P2, 5-D, E41-9). The CPU arbitrator responds to UB NPR L by sending UB NPG L into the NPR chip (P2, 6-C, E41-5). This results in P2 SACK OUT H which goes to the UNIBUS driver on Page 1 to produce UB SACK L (P1, 2-C, E15-1). If this controller is not the one making the request, the NPR chip passes the grant by asserting the UB NPG L line instead of the SACK line. The arbitration cycle is over and requesting and granting can now cease because SACK holds this controller to be the next UNIBUS master.

The NPR chip now monitors UB BBSY L (P2, 5-C, E41-10) to wait for the present bus master to relinquish its ownership. When this happens the NPR chip asserts UB BBSY L to indicate that it is now the current owner, and asserts P2 MASTER L which produces MSYN through the logic in the upper right quarter of Page 2. P2 MASTER L also asserts the UB A17:01 L lines, using the contents of the BAR to address the system memory. It also controls UB C1 L line (P1, 2-D, E69-1). The data lines UB D16:00 L are gated to/from the tristate bus (connecting to the silo buffer) to complete the data path. Normally, the slave responds with a UB SSYN L (P1, 1-C, E26-15) which resets the MSYN logic and the NPR chip drops the UB BBSY L and the P2 MASTER signals to relinquish ownership of the UNIBUS. In case the slave does not respond with a UB SSYN L, the MSYN time-out logic (P2, 2-B) allows release of ownership to prevent a "hung bus" situation. The time-out condition sets bit 13 in the CSR to notify the CPU of the problem. The controller logic can hold bus ownership for two DMAs to occur. This is controlled by the logic on Page 2 at coordinates 5-D.

3.2.4 Miscellanous Signals

3.2.4.1 INIT

The signal UB INIT L (P1, 2-B, E26-1) is used to clear the controller to its initialized state.

3.2.4.2 AC LO

The signal UB AC LO L (P1, 2-C, E15-15) does not affect the controller but is passed through to the drive I/O bus as POWER FAIL L (P11, 7-B, Berg connector pin UU). It goes to every drive on the I/O bus where it becomes AC LO L. This signal causes the heads to unload and retract to the home position, even if the drive is not selected.

3.2.4.3 PA, PB, A0, C0, DC LO
The signals PA, PB, A0, C0, and DC LO are not used by the RL11 Controller.

3.3 DRIVE I/O BUS
The drive I/O bus connects the RL11 Controller to as many as four RL01 and/or RL02 drives in a daisy-chain fashion. Physically the bus consists of the following.

- A 40 pin connector on the M7762 module (J1)
- A flat 40 wire BC06R cable
- A transition connector on the rear of the cabinet
- A round BC20J cable that is the drive I/O cable
- A 40 pin connector on the rear of the drive unit (J1) this connects in parallel to the Drive Logic Module (J12) and to another connector (J2) on the rear of the drive that provides for a cable out to the next drive
- Another BC20J cable to the next drive, and so on
- A terminator on J2 of the last drive.

The drive I/O bus components are illustrated in Figure 3-1.

Electrically, the cable contains twelve differentially-driven signals, one single-ended signal, and fifteen grounds. The pin assignments are shown in Table 3-3. The receivers and drivers are shown on CS M7762, Sheet 11. The use of each signal is explained in the following paragraphs.

Drive command bits, status bits, and data bits are transferred in serial fashion while the other signals are transferred in parallel.

3.3.1 Controller Generated Signals

3.3.1.1 Drive Select
The drive select bits DR 01:00 are transferred continuously onto the bus from CSR 09:08. The drive compares DR 01:00 with the bits from the unit number plug. When a match occurs, the drive is "selected" and enables its receivers and drivers.

3.3.1.2 System Clock
This signal is a 4.1 MHz clock provided by the controller to the drives to synchronize the transfer of command bits and write data bits. It is also used internally by the drives for timing.

3.3.1.3 Drive Command
The drive command is sixteen bits that are transferred in serial fashion from the DAR in the controller over the I/O cable into the drive command and status register located in the drive. The first bit shifted (the low order bit) is a marker bit and is always a one. It is used by the drive as notification of both the start and end of a command shift. The drive uses System Clock to synchronize the shifting of each bit received.

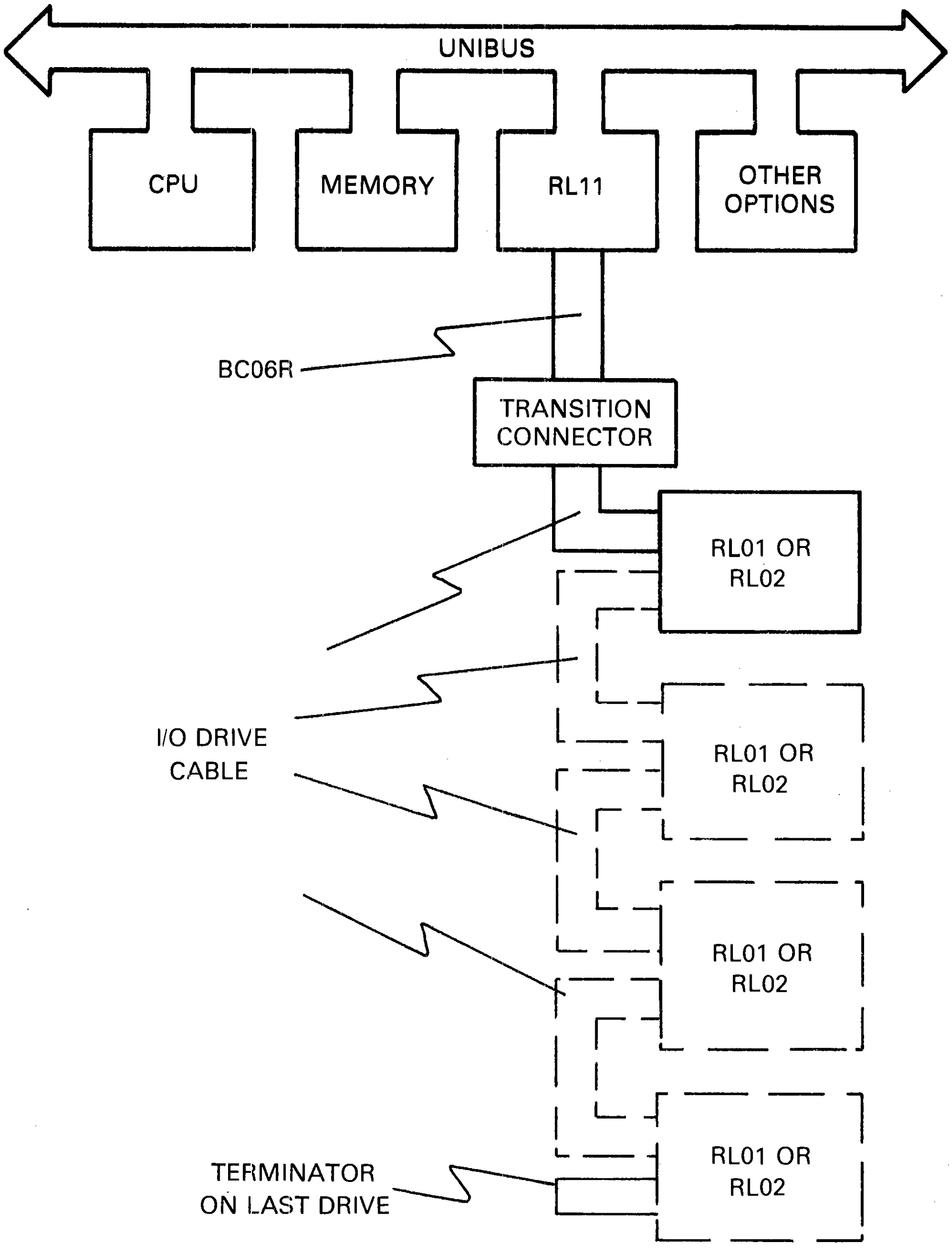


Figure 3-1 Drive I/O Bus Components

Table 3-3 I/O Bus Pin Assignments

Signal Name	RL11 J1 Pin Number	RL01 J1 Pin Number	RL02 DLM J1L Number
GROUND	A	1	VV
GROUND	B	2	UU
DR ERR H	C	3	TT
DR ERR L	D	4	SS
DR RDY H	E	5	RR
DR RDY L	F	6	PP
SEC PLS H	H	7	NN
SEC PLS L	J	8	MM
GROUND	K	9	LL
GROUND	L	10	KK
STATUS CLK H	M	11	JJ
STATUS CLK L	N	12	HH
GROUND	P	13	FF
GROUND	R	14	EE
STATUS H	S	15	DD
STATUS L	T	16	CC
GROUND	U	17	BB
GROUND	V	18	AA
READ DATA H	W	19	Z
READ DATA L	X	20	Y
DR CMD L	Y	21	X
DR CMD H	Z	22	W
GROUND	AA	23	V
GROUND	BB	24	U
WR DATA L	CC	25	T
WR DATA H	DD	26	S
GROUND	EE	27	R
GROUND	FF	28	P
SYS CLK L	HH	29	N
SYS CLK H	JJ	30	M
GROUND	KK	31	L
GROUND	LL	32	K
WR GATE L	MM	33	J
WR GATE H	NN	34	H
DR SEL 1 L	PP	35	F
DR SEL 1 H	RR	36	E
DR SEL 0 L	SS	37	D
DR SEL 0 H	TT	38	C
POWER FAIL L	UU	39	B
GROUND	VV	40	A

There are only two controller functions that result in a command being shifted to the selected drive. They are seek and get status.

During a seek function, the command consists of the following.

Bit(s)	Name	Function
00	Marker	Always a one
01	Function	A zero indicates a seek
02	Direction	A one indicates forward (toward the spindle)
04	Head Select	A one selects the lower head
15:07	Difference	Difference count

The other bits are shifted but not used.

The drive accepts the command and starts a seek operation.

During a get status function, the command consists of the following.

Bit(s)	Name	Function
00	Marker	Always a one
01	Function	A one indicates a get status
03	Reset	A one causes errors to be reset before status is sent to the controller

The other bits are shifted but not used.

The drive responds by loading the status conditions into the drive command and status register and shifting them to the controller.

3.3.1.4 Write Data and Write Gate

Write Gate is sent by the controller and causes the selected drive to turn on the write current in preparation for writing on the media. The write current is turned off when write gate is dropped.

Write data is sent by the controller in serial fashion and is used by the drive to toggle the direction of the write current. The drive uses system clock to sychronize the bits received.

3.3.1.5 Power Fail

This is a single-ended signal that is the result of the controller receiving an AC LO signal from the UNIBUS which indicates that the system power supply is detecting a power failure. All drives on the I/O bus respond by unloading and retracting the heads to their home position, regardless of drive selection.

3.3.2 Drive Generated Signals

The drive generated signals are explained in the following paragraphs.

3.3.2.1 Drive Ready

This signal is sent by the selected drive to indicate that it is ready to read or write or seek. It is sent when the heads are not moving and are locked onto a cylinder. The READY indicator on the unit front panel is lit when drive ready is asserted.

Drive ready is transferred to UNIBUS bit 00 when the CPU reads the CSR (P5, 2-A, E89-10). It is also used to prevent the shifting of a seek command from the DAR to the selected drive until the drive is ready (P6, 7-B, E102-1). Drive ready is also used to suspend the sequencing of the function control ROM program counter (in the controller) until the selected drive becomes ready.

3.3.2.2 Drive Error

This signal is sent by the selected drive to the controller as an indication that there is an error condition. The controller transfers this signal to the CPU as bit 14 when the CSR is read.

3.3.2.3 Sector Pulse

This signal is derived from the sector transducer in the selected drive. It has a duration of 48 ±12 microseconds and occurs every 625 ±6 microseconds. It signifies the start of a sector and is used to:

- Clear the write data shift register (P10, 7-B, E64-1)
- Control the counting of the program counter for the function control ROM (P6, 7-B, E102-15)
- Reset the word counter for the format control ROM (P7, 6-D, E110-1), resets the header compare logic (P7, 7-C, E56-1)
- Count the sector counter (P7, 6-B, E100-1).

3.3.2.4 Read Data

This is the data produced by the read head of the selected drive. It is provided continously except during sector pulse time when the information being read by the head is servo information instead of data. It goes directly to the data separator in the controller.

3.3.2.5 Status and Status Clock

Status bits are shifted from the drive command and status register in the selected drive to the controller where the programmer can access them from the MPR. The sixteen status bits are shifted in serial fashion into the silo (P7, 2-B, E74-9) where they are converted to parallel and bubble to the top to become available at the output data buffer. This output data buffer containing the status word is accessed when the programmer reads the MPR. The status clock signal is provided by the selected drive as a synchronizing clock along with the status bits. Note that the drive can provide status while it is performing a seek operation.

CHAPTER 4
FUNCTIONAL DESCRIPTION

4.1 INTRODUCTION

The RL11 Controller is divided into several functional areas. This chapter explains each area to the level of the functions performed by that area. The explanation does not examine each signal. The text references the circuit schematic (CS) drawings of the Field Maintenance Print Set (MP00153) and the functional block diagram shown in Figure 4-1.

The UNIBUS interface functions and the drive I/O bus functions were explained in Chapter 3 and thus are not covered in this chapter.

4.2 CONTROL STATUS REGISTER (CSR)

The CSR is shown on sheet 5 of the CS drawings. It is not the ɜual flip-flop register. Some of the bit positions are simply logic signals that are gated to the tristate bus when the CSR is addressed as a source. These conditions are not altered when the CSR is treated as a destination. These logic signal bits include 15:10 and 00. Bits 04 and 05 are shown with the BAR since they are a logical extention of the BAR. The remaining bits are flip-flops and can be either source or destination for the tristate bus.

4.3 BUS ADDRESS REGISTER (BAR)

The BAR is shown on sheet 3 of the CS drawings. Bit positions 15:01 are gated to/from the corresponding bits of the tristate bus when the BAR is addressed by the register select logic. Bit positions 17:16 are gated to/from bits 05:04 of the tristate bus when the CSR is addressed by the register select logic. Bit positions 17:01 are gated to the UNIBUS address lines during an NPR transfer to serve as a pointer to the desired destination/ source in the system memory. Bit position 00 is not used because the RL11 Controller is a word oriented device.

The BAR is also a counter and incremented by the UNIBUS interface logic every time an NPR transfer takes place. There is no "inhibit bus address increment" feature in the RL11 Controller.

4.4 DISK ADDRESS REGISTER (DAR)

The DAR is shown on sheet 8 of the CS drawings. It can be either a source or destination of the tristate bus so that the CPU can access it. The output of the DAR is serialized using a multiplexer and a bit counter for transfer over the drive command line to the drive. This feature is used during a get status or a seek operation.

During a read/write operation, the DAR holds the disk address. Its contents are serialized and compared with the header from the disk during a header check. It is incremented by a signal from the format control ROM at the end of each sector so that its contents agree with the current sector being read/written.

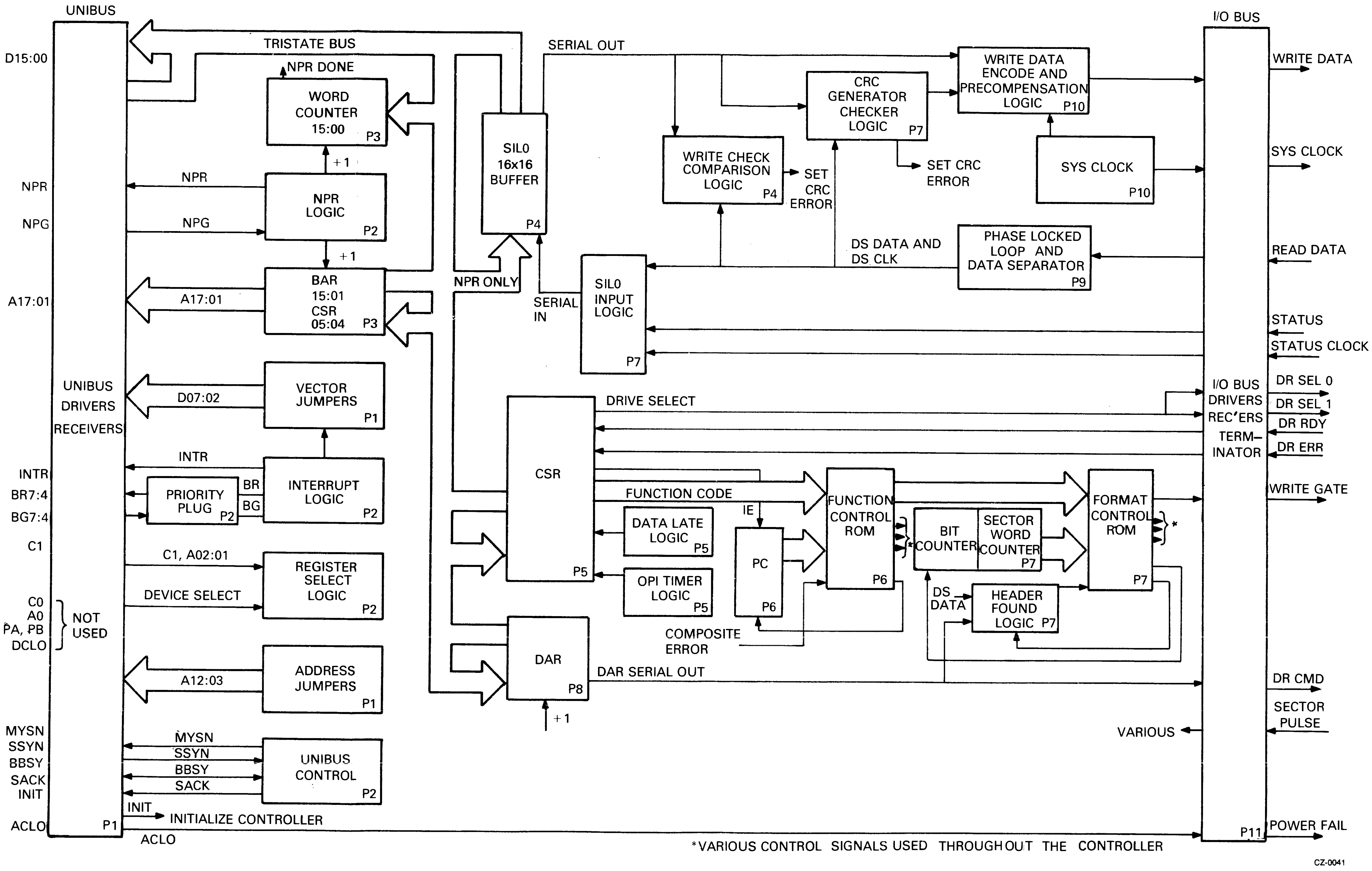


Figure 4-1 RL11 Functional Block Diagram

4.5 WORD COUNTER REGISTER

The word counter register is shown on page 3 of the CS drawings. It receives its input from the tristate bus when the multipurpose register (MPR) is addressed as a destination during a CPU access. It holds the word count as a two's complement number and is incremented by the UNIBUS interface logic every time an NPR transfer takes place. When overflow occurs the signal NPR DONE is generated to cause the operation to be terminated at the end of sector.

The word counter register cannot be a source for the tristate bus and therefore the CPU cannot call it.

4.6 SILO BUFFER

The silo is a 16x16 buffer and is shown on sheet 4 of the CS drawings. It is a source for the tristate bus when the CPU addresses the MPR as a source to get the results of a get status or a read header operation. During a read operation, it is also a source during an NPR transfer of data to the system memory. The silo buffer cannot be a destination of the tristate bus during a CPU transfer to the MPR because the register select logic selects the word counter register as the destination.

The silo receives parallel data from the tristate bus during an NPR transfer of data from the system memory to the buffer during write or write check operations. The parallel data bubbles up to the output of the silo where it is converted to serial data out.

The silo also receives serial data from the drive (data, header, or status) during a read, read header, or get status operation. The data is converted to parallel, bubbling to the output where it is available to the tristate bus. From here it is transferred to the UNIBUS as the result of either an NPR or CPU access.

4.7 SILO INPUT LOGIC

The logic that determines the source of signals for the serial input stream to the silo is shown in the lower right quadrant of sheet 7 of the CS drawings. The logic selects either the data from the drive or the status bits from the drive as the data source along with either DS CLOCK or STATUS CLOCK.

4.8 WRITE CHECK COMPARISON LOGIC

Sheet 4 of the CS drawings shows the logic utilized during write check operations to compare the data from system memory (coming through the silo) with the data from the drive. At this point both data sources are in serial form and are compared using a single EXclusive-OR gate. Any miscomparison sets the write check error flip-flop, which in turn sets the CRC error condition. This is interpreted by the CPU software as a write check error.

4.9 CRC GENERATOR/CHECKER LOGIC

The CRC generator/checker logic is shown on sheet 7 of the CS drawings. It generates a CRC checkword while data is being transferred during a write operation. At the end of the sector this checkword is appended to the data stream being written on the media.

This logic is also used to generate a CRC checkword during a read operation while data is being read from the media. At the end of the sector this generated checkword is compared with the CRC checkword that is read from the media. If they are not the same, the CRC error flip-flop is set.

The algorithim used to generate the CRC checkword is an internal function of the 9401 chip. The S2:0 inputs select one of eight formulas.

4.10 WRITE DATA ENCODE AND PRECOMPENSATION LOGIC

This logic is shown on sheet 10 of the CS drawings. The data that is being transferred to the disk in serial fashion during a write operation passes through this logic where two alterations are performed as explained below.

The data coming from the silo is in non-return to zero (NRZ) format. It has to be translated to modified frequency modulation (MFM or Miller encoded) format. The data is shifted through a register so that the pattern of ones and zeroes can be examined by a multiplexer to generate a series of pulses that are sent over the drive I/O bus to the write amplifier in the drive. These pulses clock a flip-flop to generate MFM format. MFM is defined as a steady state write current that switches direction in the center of the cell for a data one, does not switch in the center of the cell for a data zero, and switches on the leading cell boundary for consecutive zeroes.

While the data is in the shift register, the pattern is also examined by a decoder and an appropriate delay is selected so that the data is written either early, at the normal time, or late to compensate for a recording phenomenon known as peak shift. If peak shift is not compensated for at write time, it causes bit crowding when the data is read back.

4.11 SYSTEM CLOCK

The system clock is shown on sheet 10 of the CS drawings. It consists of an 8.2 MHz oscillator that is divided by two to become a 4.1 MHz timing signal. The 4.1 MHz signal is used throughout the controller and also sent over the drive I/O bus to all drives.

4.12 PHASE LOCKED LOOP AND DATA SEPARATOR

The logic shown on sheet 9 of the CS drawings and is divided into the phase-locked loop (PLL) and the data seperator (DS). The function of each is explained below.

The PLL stabilizes the incoming data by synchronizing a voltage-controlled oscillator (VCO) to the data rate. The PLL receives data from the selected drive in the form of pulses that are the result of the read amplifier sensing a transition (zero crossing). Once the data pulses enter the PLL, the pulse width is standardized to 70 nanoseconds by use of two gates and a delay line. The time relationship between the data pulse and the output of the VCO is examined. Any time difference (error) is fed back to control the frequency and phase of the VCO so that its output becomes locked to the incoming pulses.

The PLL circuitry is enabled by the format control ROM soon after the start of a preamble during the performance of an operation that requires the reading of a header or data area. Since the preamble was written as all data zeroes using MFM format, the pulses that are read back are all at the cell boundaries and represent clock (as opposed to data). Thus the VCO becomes synchronized with the cell boundary clock and remains that way throughout the area being read while smoothing out irregularities in the incoming data stream.

The DS examines the incoming data stream and seperates the pulses into DS data and DS clock. The DS is enabled by the format control ROM after the PLL has had a chance to become locked (but still during the preamble). It then starts looking for the sync bit that occurs at the end of the preamble. The logic consists of five flip-flops in the center right of CS sheet 9. The enable flip-flop synchronizes the start of read enable to the data. The window flip-flop toggles such that it is open during data time and closed during clock time to order to separate the data pulses from the clock pulses. The data flip-flop is set by incoming data one bits. The DS data flip-flop is also set by data ones but is synchronized to the VCO. The marker flip-flop looks for the sync bit and once it is set, the controller is provided with DS DATA and DS CLOCK.

4.13 DATA LATE LOGIC

Sheet 5 of the CS drawings shows the logic that detects the data late condition. This condition can occur during any write function if the silo buffer needs data and the UNIBUS is not able to respond before the silo is empty. It can also occur during any read function if the UNIBUS cannot respond fast enough to keep the silo from filling up.

DATA LATE is an error condition which aborts the operation and sets the ready condition by affecting the address input to the function control ROM (COMPOSITE ERROR).

4.14 OPI TIMER LOGIC

Sheet 5 of the CS drawings shows the operation incomplete timer. The timer is started with each operation and stopped when the operation is completed. If, for some reason, the operation does not complete before the timer times out, the OPI error condition is set. OPI error generates COMPOSITE ERROR which aborts the operation.

4.15 INTERNAL CONTROL

When the controller performs an operation, the various areas of logic have to be coordinated. This coordination is handled by two read only memories (ROM) that have the different control signals stored as data bits at their various addresses. By applying an appropriate address to a ROM, the desired combination of control signals can be activated. The two ROMs are the function control ROM and the format control ROM. Their basic functions are explained below. The ROM maps are given in Appendix B.

4.15.1 Function Control ROM

The function control ROM consists of two 82S114 chips and provides 256 words of 14 bits each (2 bits not used).

The function control ROM modifies the sequence of events according to what operation is being performed. Therefore, CSR 03:01 (function code) is used as part of the address input. The other address input comes from a program counter (PC). The PC starts at 0 and counts sequentially for each operation, but branches to the last count of 16 (or 17) at different points depending upon the operation. For that reason the PC receives a branch condition as an input from the ROM. The input appears when that operation has reached the last count that is necessary for that operation. Whether the end count is 16 or 17 depends on whether or not the CSR IE bit was set. In one case, a BR sequence is initiated. In the other case, it is not. This is the reason the PC receives IE as an input. The function ROM, the PC, and the PC clock logic are shown on sheet 6 of the CS drawings.

The outputs of the function control ROM are various control signals used throughout the controller. Three of these (CON 2:0) are used to select the proper signal to increment the PC, while one (BRANCH) is used to load the end address of 16 or 17 into the PC. Three more (FUNC 2:0) are sent to the format control ROM as address inputs to influence its output according to what operation is being performed.

The control signal REQ INTR goes to the BR logic to request a CPU interrupt.

The NPR IN and NPR OUT signals enable the NPR logic and indicate the direction of data flow.

The control signals PROG CLR GO and PROG CLR ERR clear GO (set ready) and clear the error conditions.

The control signal SEND STATUS enables the output of the DAR to be serialized and sent as a command to the drive over the DR CMD line during a GET STATUS operation.

The control signal RD STATUS enables the status bits from the drive to be shifted into the silo as the result of a get status operation.

4.15.2 Format Control ROM

The format control ROM consists of an 82S114 chip and an 82S115 chip connected to provide 512 words of 15 bits (1 bit is not used).

The format control ROM provides the signals that indicate where in the media format the heads are currently located (such as header, data, CRC, etc.) It uses three operation-dependent signals from the function control ROM as address inputs because different signals are required from this ROM for different operations. It also uses the output of a sector word counter as address inputs. This sector word counter should not be confused with the program addressable "MPR" word counter register. It is, instead, a separate area of logic that keeps track of the word count from the start of the sector (as indicated by the end of sector pulse). It is clocked by the overflow from a 16 count bit counter. The bit counter can be clocked either internally by system clock or externally by DS clock from the drive, depending on the operation.

The header found condition is also used as an address input to the format control ROM since that condition affects what signals are required from the ROM. The format control ROM, the word counter, the bit counter, and the header found logic are all shown on sheet 7 of the CS drawings.

The header found logic compares the header from the drive with the serialized contents of the DAR to determine if the header being read from the disk is the one specified by the DAR.

The control signals ,produced by the format control ROM are explained below.

TRANSFER DONE is generated when the word count indicates the end of the data area of each sector being read/written. It checks for the end of the operation, increments the DAR, and clears the header found condition so it can be used again on the next sector.

STR HDR FOUND occurs at the end of each header area while searching for a header match. It strobes the match condition.

POST occurs during the postamble of both the header and the data areas and sets the CRC error flip-flop if a CRC error had occured.

HDR POST occurs during the postamble of the header only and sets the OPI error condition if the CRC error is set. This is interpreted by the CPU software as a header CRC error.

ENA BC CLK enables the bit counter to count system clock pulses.

RD ENABLE occurs during any operation that is reading either a header or a data area. It enables the data separator (sheet 9), causes the bit counter to be counted by DS CLK, enables the CRC chip, and gates DS DATA and DS CLK as the data and clock sources for the CRC chip.

LOOP LOCK occurs during any operation that is reading either a header or a data area and is used to enable the PLL circuitry on sheet 9 of the CS drawings.

WRT DATA AREA occurs during the data time of a write operation to permit data to be shifted out to the drive I/O bus as well as to the CRC logic. It enables the CRC chip and the CRC input clock (system clock).

ENA WRT CHK occurs during a write check operation and enables the write check logic on sheet 4 as well as the end of operation logic on sheet 8.

ENA CRC OUT occurs during a write operation immediately after writing the data area to append the CRC checkword to the data stream.

ENA SYNC 1 gates a single one bit out on the write data stream at the end of the data preamble to write the sync bit.

ENA SILO SER is used during a write operation to gate the serial data from the silo to the write encode and precompensation logic. This is shown on the gating structure at the output of the CRC logic. It is also used during a write or write check operation to gate system clock as CPSO, which is the clock pulse serial out needed by the silo chips to shift the top of the silo out in serial fashion.

WRITE GATE goes to the I/O bus through the driver on sheet 11 to turn on write current, and also allows checking for the end of operation (via the logic on sheet 8).

ENA DIFF CLK allows system clock to count the bit counter to send the difference number from the DAR to the drive during a seek.

RD DATA AREA occurs during a read when the heads are over the data area. It gates DS DATA and DS CLK to the silo. It is also used during a read header (when the heads are over the header area).

CHAPTER 5
COMMAND DESCRIPTIONS

5.1 GENERAL

Each operation performed by the RL11 Controller accomplishes its task by sequencing through its assigned area of addresses in the function ROM and format ROM. The ROM data outputs become control signals to govern the functions being performed as well as the sequencing of addresses. An overview of the operation follows. Each command is then explained in detail.

The function ROM is divided into eight areas, one for each command. An area is selected by the function code bits from the control and status register (CSR). The program counter (PC) selects specific addresses within the area by sequencing from zero through 17 (octal). At some step during the sequence, the control signal BRANCH causes the PC to be set to an ending address of either 16 or 17 on the next count pulse. The interrupt enable (IE) flip-flop in the CSR governs the choice of 16 (instead of 17) during the BRANCH. Step 16 generates the REQ INT signal that causes the UNIBUS interface to request a BR interrupt. The PC then advances to Step 17. If IE is not set, the PC goes directly to 17 during the BRANCH and interrupt is not requested.

There are several housekeeping chores that are common to all operations. For example, all operations wait at Step 0 for the GO bit to be cleared. When it is cleared, the PC steps to 1. During Step 1 of all operations, the signal PROG CLR ERROR clears the error conditions within the controller. During the last count (17) of each operation, the signal PROG CLR GO clears the GO flip-flop (sets CRDY) in the CSR to indicate to the CPU that the next operation can be started.

During each step a combination of CON 2:0 signals selects one of eight conditions that govern the incrementing of the PC. During most commands a combination of FUNC 2:0 signals selects an area of memory from the format ROM to control the synchronization of events with the rotation of the disk. During that time CON 2:0 will prevent the PC from incrementing until the format ROM has completed its task. The other signals from the ROMs are explained in the text that describes each operation. The ROM maps are contained in Appendix B.

The signal COMPOSITE ERROR is part of the addressing scheme of the function ROM and if this signal occurs, it causes BRANCH to be performed immediately. This has the effect of aborting the operation. Thus any error in the composite error group is considered a fatal error.

5.2 NO-OP

The no-op command uses the function ROM to generate the few signals that this operation requires. It does not require the use of the format ROM. The PC sequences through Steps 0 and 1, and then branches to either 16 or 17. During Step 0, the CON 2:0 signals are equal to 7 which prevents the PC from incrementing until GO is reset. This is true of all operations. When GO is reset by the CPU software, the PC counts to 1 where PROG CLR ERR and BRANCH are generated to clear the errors and branch to 16 or 17 depending upon the setting of IE. Step 16 provides INT REQ and Step 17 provides PROG CLR GO to complete the operation. The PC is advanced to 0 where it waits for the next clearing of GO by the CPU software to start the next operation. The only action performed by the no-op that is visible to the CPU software is the clearing of the controller error conditions.

5.3 GET STATUS

During a get status operation, the PC sequences from 0 through 13 then branches to 16 or 17. During Steps 0 through 12, the signals CON 2:0 are equal to 7 which selects the GO signal as the condition to allow the incrementing of the PC. As a result, the PC is advanced with every other system clock (a flip-flop in the increment logic divides the counter by two). Step 1 clears the errors. Step 2 does not perform any control functions. Steps 3 through 12 generate the SEND STATUS signal which performs two services. First, it enables the bit counter to be counted by system clocks pulses. (The bit counter serializes the contents of the DAR.) Second, it enables the serialized contents of the DAR to be transferred to the DR CMD line on the I/O bus. Step 13 generates the READ STATUS signal which allows the STATUS and STATUS CLOCK signals from the drive to shift STATUS into the silo. During Step 13, the CON 2:0 signals are equal to 1 which causes ST IN SILO to suspend the PC clock until the input buffer of the silo is filled with the sixteen status bits. BRANCH is also produced during this step. When the PC clock is allowed to occur and BRANCH occurs at the same time, the PC is branched to 16 or 17 instead of incrementing. At the end of the operation, the status word has bubbled up to the top of the silo where it is available at the data buffer output ("MPR").

5.4 SEEK

The seek operation sequences the function ROM from Step 0 through 4 and then branches to 16 or 17. Steps 0 and 1 are the same as any other operation. Step 2 generates CON 2:0 = 3 which holds the PC count until DR RDY is provided by the drive (if it is not already present). Step 3 generates CON 2:0 = 4 which holds the PC count until the next sector pulse. Step 4 generates FUNC 2:0 = 3 which enables the format ROM. This step also generates CON 2:0 = 2 which holds the PC count until the format ROM has completed its task. This step also produces BRANCH so that when the PC is allowed to count again it will branch instead of increment.

When the format ROM receives FUNC 2:0 = 3, it provides ENA BC which enables the bit counter to be counted by system clock pulses. The bit counter then counts bits and words starting at the end of the sector pulse. The format ROM uses FUNC 2:0 and the output of the word counter as addresses. Any time the bit counter is counting, the DAR is being serialized by the action of the multiplexer. During word 6, ENA DIFF is generated which enables the serialized contents of the DAR to be transferred over the DR CMD line to the drive. Thus the seek command is sent to the selected drive. When the word counter reaches 7, TRANSFER DONE is produced which is ANDed with FUNC 2 L (on sheet 8 of the circuit schematics) to produce DONE. DONE is used by the PC logic to allow the PC to branch to 16 or 17. At this time the drive has received the seek command and has started the seek operation. The controller is now free to perform another command.

5.5 WRITE CHECK

During a write check operation the PC sequences from 0 through 5 then branches to 16 or 17. Steps 0 and 1 are the same as all other operations. Step 2 generates CON 2:0 = 3 which holds the PC count for DR RDY. Step 3 does not perform any control functions. Step 4 generates CON 2:0 = 4 which holds the PC until the next sector pulse. It also generates NPR IN which allows the UNIBUS interface to fill the silo with data from system memory. After sector pulse occurs, the PC is incremented to 5. This generates FUNC 2:0 = 2 which enables the format ROM. Step 5 also generates NPR IN to allow the UNIBUS interface to keep the silo filled as words are shifted out to the write check compare logic. Step 5 also generates BRANCH and CON 2:0 = 2 which holds up the PC count until the format ROM has completed its task.

The format ROM receives the FUNC 2:0 = 4 which selects the area of the ROM that is assigned to write check operation. LOOP LOCK is generated from word 0 through 3 to gate the data pulses from the drive into the phase-lock loop circuitry. RD ENA is generated during words 2 and 3 after the PLL has had a chance to synchronize itself to the incoming data pulses. RD ENA allows the data separator to function. ENA BC is generated during words 0 and 1 to allow the bit/word counters to be counted by system clock pulses up to a word count of 2, where it waits for the sync pulse at the end of the header preamble. This bit sets the marker flip-flop in the data separator which allows DS DATA and DS CLOCK to be sent to the various places in the controller where they are needed. DS CLOCK now counts the bit/word counters. During word 2 the first word of the header (containing the disk address) is compared with the serialized contents of the DAR to see if this is the desired sector of the desired cylinder and surface. During word 3 the signal ST HDR FND checks the result of the header comparison.

If the result of the header comparison was negative, the word counter is held at 4 until the next sector pulse occurs. Sector pulse resets the header found logic and clears the word counter to zero repeating the header comparison process.

If the result of the header comparison was positive, the signal HEADER FOUND is sent to the format ROM as an address bit to cause a different set of conditions to be read from the ROM. LOOP LOCK is kept asserted through word 7. RD ENA is kept asserted through word 4, allowing the second and third words of the header to enter the CRC logic. POST and HDR POST are generated during word 5 to monitor the result of the header CRC check. ENA BC is kept asserted during words 5, 6, and 7, enabling the word counter to increment up to 8. The count is suspended until the sync bit at the end of the data preamble is encountered and DS CLOCK will then increment the word counter.

If a positive header comparison is not made within 200 milliseconds, the OPI timer will time out and the resulting COMPOSITE ERROR will abort the operation.

The 128 words of data are read during word counts 8 through 135. LOOP LOCK and RD ENA are kept asserted during this time to allow DS DATA and DS CLOCK to be used by the controller. ENA WRT CHK is asserted during this time allowing DS CLOCK to shift SILO SER DATA (data from the system memory) into the comparison logic to be compared with DS DATA from the disk. This comparison is the purpose of the write check operation. A miscomparison results in setting the CRC error flip-flop which is interpreted by the CPU software as a write check miscompare.

During word 136, LOOP LOCK and RD ENA are kept asserted, enabling the CRC checkword to the CRC logic. POST occurs at word 137 to monitor the result of the CRC check. ENA BC occurs during words 137 and 138 to increment the word counter. TRANSFER DONE and ENA WRT CHK occur during word 138 to check for a silo empty condition. It is assumed that if the silo is empty then the end of the operation has been reached. If so, DONE is produced which allows the PC to branch to 16 or 17, completing the operation. If not, the word counter stops at 139 until the next sector pulse resets the word counter to zero and the process is continued during the next sector.

5.6 WRITE

The write operation sequences the PC from 0 through 5 and then branches to 16 or 17. Steps 0 and 1 are the same as all other operations. Step 2 generates CON 2:0 = 3 which holds the PC until the condition DR RDY occurs (if it is not already present). Step 3 allows the PC to step to 4. Step 4 generates CON 2:0 = 4 to make the PC wait until the next sector pulse and generates NPR IN so that the UNIBUS interface can fill the silo with data from the system memory. Step 5 generates the signals necessary to accomplish the four items listed below.

- FUNC 2:0 = 5 to address the format ROM
- NPR IN to keep the silo filled
- CON 2:0 = 2 to wait for the format ROM to complete its task
- BRANCH to cause the PC to branch to 16 or 17 when the PC is released

When the format ROM is addressed by FUNC 2:0 = 5, it performs a header comparison by generating the following signals. ENA BC is generated during word 0 to increment the word counter to 1 where it waits for the sync bit at the end of the header preamble. LOOP LOCK is generated during words 0 through 2 enable to the PLL. RD ENA is generated during words 1 and 2 to allow the first two words of the header into the controller. The first word is compared to the DAR by the header comparison logic. STR HDR FND occurs during word 2 to check the results of the comparison.

If no match occured, the word counter stops at the count of 3 until the next sector pulse resets the header comparison logic and resets the word counter to 0. The cycle repeats until a match is found. If no match is ever found, the OPI timer logic will time out and set the error flip-flop. COMPOSITE ERROR will abort the operation.

When a match is found, RD ENA and LOOP LOCK are kept asserted during word 3, enabling header CRC checkword to be used by the CRC logic. POST and HDR POST are generated during word 4 to monitor the result of the header CRC check. ENA BC is generated from word 4 through word 138 to keep the bit and word counters incrementing with system clock. WRITE GATE is generated from word 4 through 138 to write the entire sector from the start of the data preamble through the end of the data postamble. During words 5 through 7 the all-zeroes preamble is written. During word 7, ENA SYNC 1 is generated which causes a single one-bit to be ORed into the data stream. This becomes the sync bit at the end of the data preamble. WRT DATA AREA and ENA SILO SER are generated from word 8 through word 135 to shift the data to the drive. During word 136, WRITE GATE and ENA CRC OUT are generated, enabling the CRC checkword to be written. WRITE GATE is generated during words 137 and 138 to write the data postamble. During word 138, TRANSFER DONE is generated to check for the end of the operation by monitoring the silo (ORE4). DONE is generated if the silo indicates empty. If the silo is not empty the word counter holds at the count of 139 until the next sector pulse resets the header found logic and resets the word counter to zero. The entire process is repeated (including the header comparison) on the next sector. When the end of operation has been reached, DONE releases the hold on the PC and it branches to 16 or 17 to complete the operation.

5.7 READ

The read operation sequences the function ROM from 0 through 5 and then branches to 16 or 17. Steps 0 and 1 are the same as the other operations. Step 2 generates CON 2:0 = 3 to interlock the PC increment on the DR RDY condition from the drive. Step 3 generates CON 2:0 = 4 to wait for the next sector pulse. Step 4 generates FUNC 2:0 = 6 to enable the format ROM, and generates CON 2:0 = 0 to cause the PC count to wait for READ DONE. BRANCH is generated so that when the PC count is released, the PC will branch to 16 or 17. NPR OUT is generated to cause the UNIBUS interface to transfer data from the silo to the system memory as the data becomes available.

The format ROM performs a header comparison by producing the necessary signals. LOOP LOCK is produced during words 0 through 3 to enable the PLL. RD ENA is generated during words 2 and 3 to enable the data seperator. ENA BC is generated during words 0 through 1 to count the word counter up to 2 where it waits for the sync bit at end of the header preamble. After that, DS CLOCK counts the bit/word counters. During word 2, the first word of the header is compared with the DAR to see if the head is over the desired sector. During word 3 STR HDR FND is produced to check the result of the header comparison. If no match occured, the word counter stays at 4 until the next sector pulse resets the header comparison logic and resets the word counter to zero repeating the process. When a match does occur, the HEADER FOUND condition alters the address of the format ROM so that LOOP LOCK remains asserted until the end of the data CRC. RD ENA remains asserted during words 3 and 4 to allow the rest of the header to be read so that a header CRC check can be made. During word 5 the signals POST and HDR POST are generated to check the result of the header CRC check. ENA BC is generated during words 5 through 7 to count the word counter up to 8 where it waits until the head reads the sync bit at the end of the data preamble. Once the sync bit occurs, DS CLOCK counts the word counter from word 8 through word 136 which corresponds to the data and data CRC checkword.

The data area corresponds to word count 8 through 135. During this word counting period, LOOP LOCK, RD ENA, and RD DATA AREA are generated to shift data from the disk into the silo. During the CRC checkword time (at word 136) RD ENA and LOOP LOCK are generated to shift the data into the CRC logic but not into the silo. During word 137, ENA BC is generated to keep the counters counting and POST is generated to monitor the results of the CRC check.

During word 138, TRANSFER DONE is generated to check for end of operation. This signal checks the NPR DONE signal (shown on sheet 8) to see if the UNIBUS interface logic has reached the end of the desired range. If the end of range has not been reached before the next sector pulse occurs, the word counter is reset to zero and the entire process is repeated on the next sector (including the header check). If the NPR DONE condition does occur, the RD DONE signal releases the PC count and the PC is branched to 16 or 17 to complete the operation.

5.8 READ WITHOUT HEADER CHECK

During a read without header check operation the function ROM sequences from 0 through 5, then branches to 16 or 17. Steps 0 and 1 are the same as all operations. Step 2 generates CON 2:0 = 3 which waits for DR RDY. Step 3 produces no functions. Step 4 generates CON 2:0 = 4 to wait for the next sector pulse and also generates NPR OUT. Step 5 generates FUNC 2:0 = 7 to enable the format ROM as well as NPR OUT to allow to UNIBUS logic to keep the silo from filling up by transferring data from the silo to system memory. CON 2:0 = 0 holds the PC until the format ROM has completed its task. BRANCH is produced to cause the PC to branch to 16 or 17 instead of incrementing when the hold is released.

The format ROM generates LOOP LOCK from word 0 through 136 (through data CRC). ENA BC is generated during words 0 and 1 to count the word counters up to 2 where it stays until the header preamble sync bit allows DS CLOCK to count through word 4. RD ENA is asserted during words 2 through 4. The signal ST HDR FND is not produced, so the result of the header comparison is not examined. The header-found condition is simulated by having the desired bits stored in both the "header-found" and "not-header-found" sections of the ROM for this operation. Neither POST nor HDR POST is produced so the header CRC check result is not examined. ENA BC is asserted during words 5 through 7 to count the counter up to 8 where it waits for the data preamble sync bit. Thus, this sector will be read regardless of the header comparison.

The data area corresponds to words 8 through 135 and the data CRC to word 136. LOOP LOCK, RD ENA, and RD DATA AREA are asserted during the data area to keep the data shifting into the silo. LOOP LOCK and RD ENA are asserted during word 136 to keep the CRC checkword flowing into the CRC logic. ENA BC is asserted during word 137 to keep the counter going. POST is asserted during 137 to check the result of the data CRC check. TRANSFER DONE is asserted during 138 to check for end of operation. It examines NPR DONE to determine if RD DONE should be produced. If so, then the PC branches to 16 or 17 to complete the operation.

5.9 READ HEADER

During a read header operation the PC sequences from 0 through 5 then branches to 16 or 17. Steps 0 and 1 are the same for all operations. Step 2 generates CON 2:0 = 3 to cause the PC to wait on DR RDY (if the drive is not currently ready). Step 3 does not product any control functions. Step 4 generates CON 2:0 = 4 to wait for the next sector pulse. Step 5 generates FUNC 2:0 = 2 to enable the proper set of addresses in the format ROM. It also generates CON 2:0 = 2 to cause the PC to wait for DONE. BRANCH is produced causing the PC to branch to 16 or 17 when the format ROM has finished its task.

The format ROM generates LOOP LOCK from word 0 through 4 to enable the PLL circuitry. ENA BC is generated to count the word counter up to 2 where it waits for the sync pulse at the end of the header preamble. RD ENA is asserted during words 2 through 4 to shift the header into the data separator. RD DATA AREA is asserted during words 2 through 4 to shift the header information (3 words) into the silo as if it were data. ENA BC, POST, and HDR POST are generated during word 5 to check the results of the header CRC check. TRANSFER DONE is generated during word 6 and is changed to DONE when it is ANDed with FUNC 2 L (shown on sheet 8).

The PC is branched to 16 or 17 and the operation is completed.

APPENDIX A
SPECIAL ICS

A.1 UNIBUS TRANSCEIVER DC005 19-13040

The DC005 integrated circuit chip performs some unique functions of interfacing with a UNIBUS. The symbol for the DC005 chip is shown in Figure A-1 and its inputs and outputs are defined below.

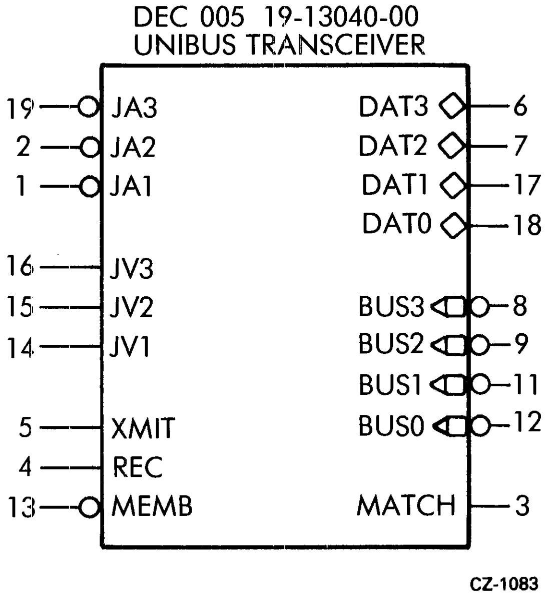


Figure A-1 DC005 UNIBUS Transceiver

JA – These are ternary inputs (gnd, Vcc, open) used for address jumpers and are activated by asserted low levels. When MEMB is enabled (low), the JA inputs 1, 2, 3 are compared with the BUS inputs 1, 2, 3. Ground (one) on JA is compared with low (one) on the BUS input. Open (zero) on JA is compared with high (zero) on the BUS input. Vcc on the JA input is the "don't care" state and causes the comparison to be considered a match for that bit. When the comparison of all three bits results in a match, the MATCH output goes high.

MEMB – This is a high impedance input used to enable the comparison of JA and BUS inputs. When it is asserted low, it enables the MATCH logic explained in the JA definition.

MATCH – This is an open collector output that is asserted high. When MEMB is low, the JA inputs and BUS inputs are compared as explained in the JA definition.

JV – These are TTL level inputs (with pull downs) that are used to transfer vector information to the BUS outputs. Any JV 1, 2, 3 input pulled high (one) will be transferred to the corresponding BUS 1, 2, 3, output as a low (one) without any enabling conditions. Any JV input pulled down or left open does not affect the corresponding BUS output.

XMIT, REC - These are TTL level inputs used to control the information flow between DAT and ·BUS input/outputs. They are activated by asserted high levels. See Table A-1 below.

Table A-1 XMIT, REC Line Functions

REC	XMIT	Function
L	L	Disable BUS, DAT. Both become open outputs.
L	H	DAT in transferred to BUS out and inverted.
H	X	BUS in transferred to DAT out and inverted.

DAT - These are TTL level inputs and tristate outputs that are asserted high. When REC is high, BUS inputs are inverted and transferred to DAT outputs. When REC is low and XMIT is high, DAT inputs are inverted and transferred to BUS outputs. When REC is low and XMIT is low, DAT and BUS are both open outputs and do not affect each other.

BUS - These are high impedance inputs and open collector outputs that are asserted low. When REC is high, BUS inputs are inverted and transfered to DAT outputs. When REC is low and XMIT is high, DAT inputs are inverted and transferred to BUS outputs. When REC is low and XMIT is low, DAT and BUS are both open outputs and do not affect each other. When MEMB is low, BUS 1, 2, 3 inputs are compared with JA 1, 2, 3 inputs as explained in the JA definition. When any JV input is pulled high the corresponding BUS output is forced low.

A.2 8641 TRANSCEIVER CHIP 19-11579-00

The 8641 integrated circuit chip performs the function of a high impedance input receiver and low leakage, open-collector driver for UNIBUS signals. The 8641 chip schematic is shown in Figure A-2.

When both the ENA and the ENB inputs are low, the IN inputs are transferred to the BUS outputs (inverted) and to the OUT outputs. When either or both of the ENA or ENB inputs are high, any BUS input in the low state causes a high on the corresponding OUT output.

A.3 8647 UNIBUS INTERFACE CHIP 19-12083-00

The 8647 is not a preferred chip. The DC013 is a newer version and is recommended over the 8647. (The problems exhibited by the 8647 chip were solved by external logic for the RL11 Controller.) There is no guarantee that a discrete logic implementation of the 8647 will perform satisfactorily. However, the equivalent logic of the 8647 is illustrated in Figure A-3.

The DEC 8647 chip is used to generate and respond to UNIBUS signals when the controller becomes bus master. One chip is used for NPR transfers and one chip is used for BR interrupts. The signals produced and responses of the chip are similar in both cases so only one is explained.

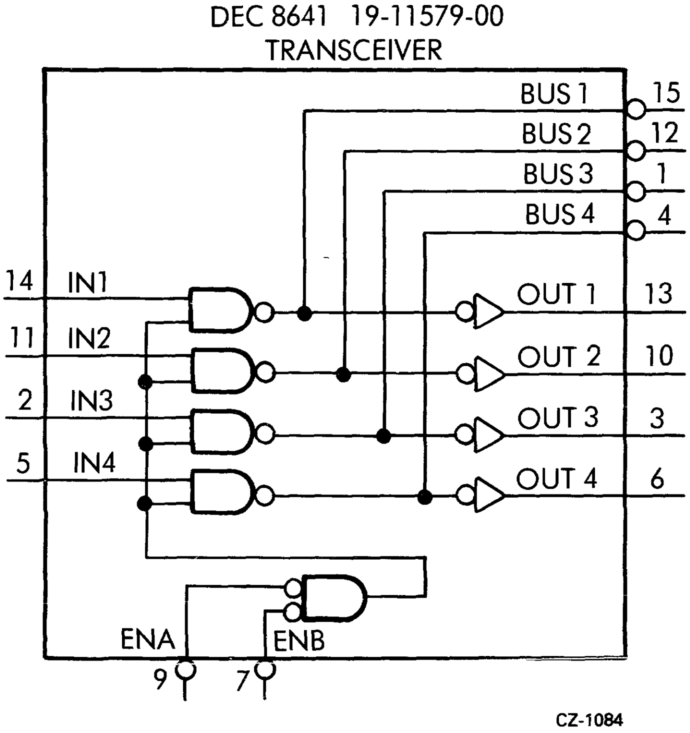


Figure A-2 8641 Transceiver

During an NPR transfer, the sequence of events is started by the controller logic asserting pin input 1 within the signal REQUEST L. This signal is latched internally and causes the chip to assert its output pin 9 NPR L (if the chip is not already asserting SACK or BBSY). When the system arbitrator responds to NPR with an NPG IN H on pin 5, the chip examines the REQUEST condition and either "passes the grant" by asserting NPG OUT H (in the case where this chip was not the one making the NPR) or the chip "takes the grant" and asserts SACK H on pin 12. The SACK condition satisfies the UNIBUS arbitrator so NPR is dropped and the arbitrator will drop NPG. The arbitration portion of the cycle is completed. Meanwhile the UNIBUS is completing the data transfer portion of the cycle for some other bus master.

The chip now monitors BBSY on pin 10 to see when the previous bus master relinquishes ownership (signified by BBSY being dropped). When BBSY is dropped, this chip asserts BBSY to claim ownership of the bus. This action clears the SACK condition to allow the system arbitrator to negotiate the next master. At this time the chip also asserts the MASTER L line (pin 11) which is used by the controller to generate other signals needed to cause the transfer to take place. After a delay, the controller produces the signal MASTER CLEAR H which is sent to pin 14 and is used in conjunction with SSYN L from the slave (system memory) to clear the BBSY and MASTER conditions. This completes the data transfer portion of the UNIBUS cycle.

When the chip is used to control a BR interrupt, the operation of the chip itself is the same except that the "steal grant" feature can be used to improve bus latency. In the case of the RL11 Controller, the STEAL GRANT L input on pin 3 is grounded to enable this feature. If the chip sees NPR and BG both asserted at the same time, SACK is asserted. This is done because obviously the arbitrator has granted a device farther down the bus the right to interrupt (BG is asserted), and after that occured, some other controller has requested an NPR transfer (NPR is asserted). This controller will then block the BG intended for the interrupting controller and assert SACK to allow the arbitrator to honor the NPR first. Not all PDP-11 arbitrators will respond in this manner, but on those systems that do, the RL11 controller can help latency problems.

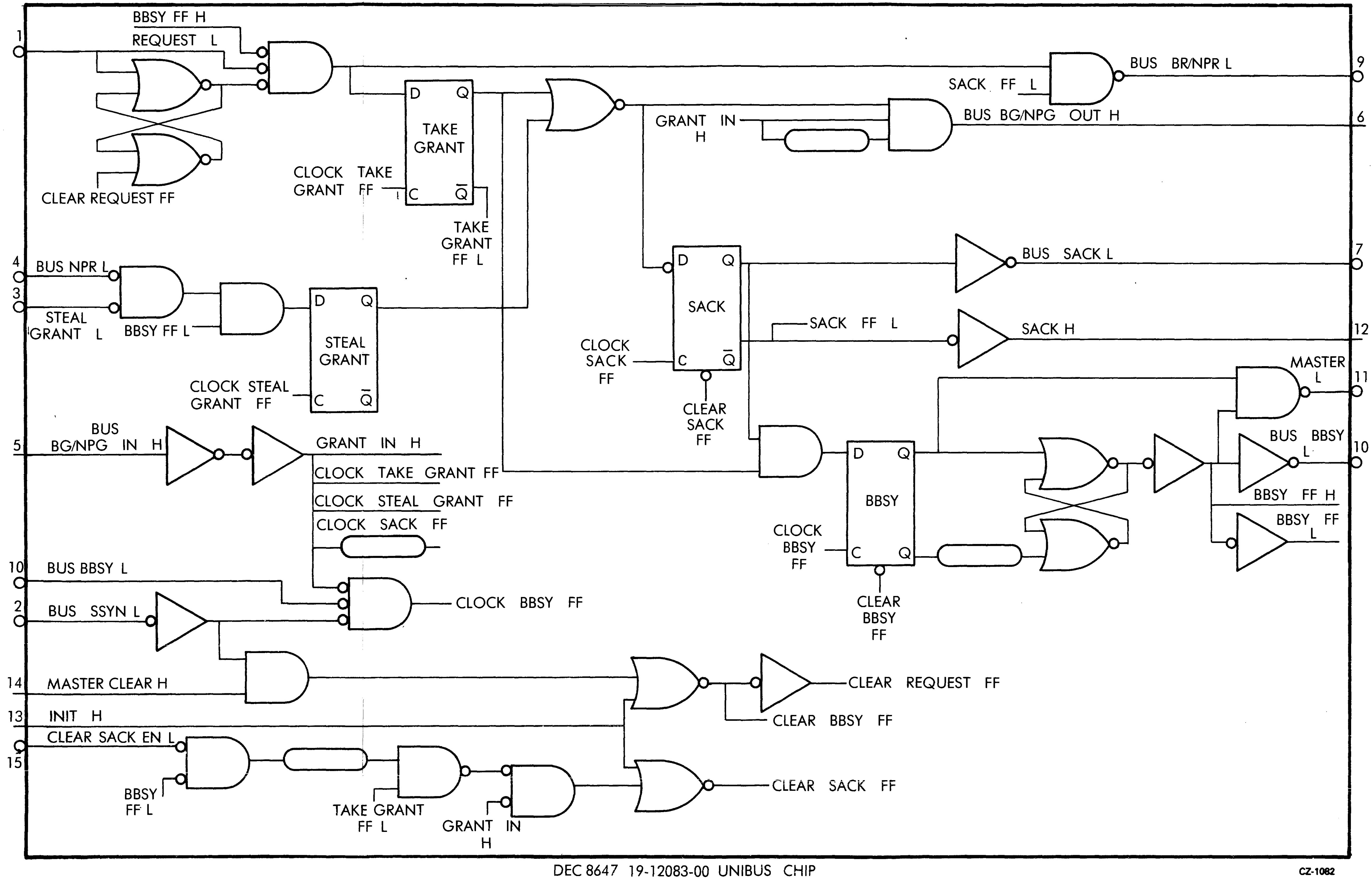


Figure A-3 8647 UNIBUS Interface

APPENDIX B
READ-ONLY MEMORIES

B.1 FUNCTION ROM

The function ROM consists of two 256 x 8 ROM chips. The addressing for both chips is shown below.

A0 =	Program Counter 2^0
A1 =	Program Counter 2^1
A2 =	Program Counter 2^2
A3 =	Program Counter 2^3
A4 =	CSR Function Code F0
A5 =	CSR Function Code F1
A6 =	CSR Function Code F2
A7 =	COMPOSITE ERROR

This addressing scheme uses A7 to divide the memory into two halves. The lower half of memory is used for the normal sequencing through an operation when no error condition exists. If an error occurs, the addressing is changed to access the upper half where the contents of memory cause the operation to abort and branch to the ending address of 16 or 17. All operations contain the same pattern in the upper half (error half) of memory.

The addressing scheme uses A6:4 to divide the memory into eight areas, one for each operation as defined by the function code in the CSR. Each operational area contains 16 locations that are defined by the program counter (PC). Each operation starts at location 00 of its area and accesses sequential locations until a control signal causes the PC to branch to 16 or 17 instead of incrementing.

The data contained in the ROM is shown below.

Upper ROM Chip	Lower ROM Chip
D0 = CON 0 H	D0 = BRANCH L
D1 = CON 1 H	D1 = REQ INTR L
D2 = FUNC 0 H	D2 = NPR IN H
D3 = FUNC 1 H	D3 = NPR OUT H
D4 = FUNC 2 H	D4 = PROG CLR GO H
D5 = SEND STATUS H	D5 = PROG CLR ERR H
D6 = RD STATUS H	D6 = not used
D7 = CON 2 H	D7 = not used

The following ROM map divides the memory into sections by operation, showing the PC count (in octal), the ROM address (in octal) and the data of both the upper and the lower ROM chip.

NO-OP Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	000	10000011	000	00000011
01	001	10000011	001	00100010
02	002	10000011	002	00000010
03	003	10000011	003	00000010
04	004	10000011	004	00000010
05	005	10000011	005	00000010
06	006	10000011	006	00000010
07	007	10000011	007	00000010
10	010	10000011	010	00000010
11	011	10000011	011	00000010
12	012	10000011	012	00000010
13	013	10000011	013	00000010
14	014	10000011	014	00000010
15	015	10000011	015	00000010
16	016	10000001	016	00000001
17	017	10000011	017	00010011

WRITE CHECK Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	020	10000011	020	00000011
01	021	10000011	021	00100011
02	022	00000011	022	00000011
03	023	10000011	023	00000011
04	024	10000000	024	00000111
05	025	00010010	025	00000110
06	026	10000011	026	00000010
07	027	10000011	027	00000010
10	030	10000011	030	00000010
11	031	10000011	031	00000010
12	032	10000011	032	00000010
13	033	10000011	033	00000010
14	034	10000011	034	00000010
15	035	10000011	035	00000010
16	036	10000001	036	00000001
17	037	10000011	037	00010011

GET STATUS Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	040	10000011	040	00000011
01	041	10000011	041	00100011
02	042	10000011	042	00000011
03	043	10100011	043	00000011
04	044	10100011	044	00000011
05	045	10100011	045	00000011
06	046	10100011	046	00000011
07	047	10100011	047	00000011
10	050	10100011	050	00000011
11	051	10100011	051	00000011
12	052	10100011	052	00000011
13	053	01000001	053	00000010
14	054	10000011	054	00000010
15	055	10000011	055	00000010
16	056	10000001	056	00000001
17	057	10000011	057	00010011

SEEK Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	060	10000011	060	00000011
01	061	10000011	061	00100011
02	062	00000011	062	00000011
03	063	10000000	063	00000011
04	064	00001110	064	00000010
05	065	10000011	065	00000010
06	066	10000011	066	00000010
07	067	10000011	067	00000010
10	070	10000011	070	00000010
11	071	10000011	071	00000010
12	072	10000011	072	00000010
13	073	10000011	073	00000010
14	074	10000011	074	00000010
15	075	10000011	075	00000010
16	076	10000001	076	00000001
17	077	10000011	077	00010011

READ HEADER Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	100	10000011	100	00000011
01	101	10000011	101	00100011
02	102	00000011	102	00000011
03	103	10000011	103	00000011
04	104	10000000	104	00000011
05	105	00001010	105	00000010
06	106	10000011	106	00000010
07	107	10000011	107	00000010
10	110	10000011	110	00000010
11	111	10000011	111	00000010
12	112	10000011	112	00000010
13	113	10000011	113	00000010
14	114	10000011	114	00000010
15	115	10000011	115	00000010
16	116	10000001	116	00000001
17	117	10000011	117	00010011

WRITE Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	120	10000011	120	00000011
01	121	10000011	121	00100011
02	122	00000011	122	00000011
03	123	10000011	123	00000011
04	124	10000000	124	00000111
05	125	00010110	125	00000110
06	126	10000011	126	00000010
07	127	10000011	127	00000010
10	130	10000011	130	00000010
11	131	10000011	131	00000010
12	132	10000011	132	00000010
13	133	10000011	133	00000010
14	134	10000011	134	00000010
15	135	10000011	135	00000010
16	136	10000001	136	00000001
17	137	10000011	137	00010011

READ Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	140	10000011	140	00000011
01	141	10000011	141	00100011
02	142	00000011	142	00000011
03	143	10000000	143	00000011
04	144	00011000	144	00001011
05	145	00000000	145	00001010
06	146	10000011	146	00000010
07	147	10000011	147	00000010
10	150	10000011	150	00000010
11	151	10000011	151	00000010
12	152	10000011	152	00000010
13	153	10000011	153	00000010
14	154	10000011	154	00000010
15	155	10000011	155	00000010
16	156	10000001	156	00000001
17	157	10000011	157	00010011

READ WITHOUT HEADER COMPARE Without Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	160	10000011	160	00000011
01	161	10000011	161	00100011
02	162	00000011	162	00000011
03	163	10000011	163	00000011
04	164	10000000	164	00001011
05	165	00011100	165	00001010
06	166	00000000	166	00000010
07	167	10000011	167	00000010
10	170	10000011	170	00000010
11	171	10000011	171	00000010
12	172	10000011	172	00000010
13	173	10000011	173	00000010
14	174	10000011	174	00000010
15	175	10000011	175	00000010
16	176	10000001	176	00000001
17	177	10000011	177	00010011

NO-OP With Error

	Upper ROM Chip		Lower ROM Chip	
PC	ADD	DATA	ADD	DATA
00	200	10000011	200	00000011
01	201	10000011	201	00100011
02	202	10000011	202	00000010
03	203	10000011	203	00000010
04	204	10000011	204	00000010
05	205	10000011	205	00000010
06	206	10000011	206	00000010
07	207	10000011	207	00000010
10	210	10000011	210	00000010
11	211	10000011	211	00000010
12	212	10000011	212	00000010
13	213	10000011	213	00000010
14	214	10000011	214	00000010
15	215	10000011	215	00000010
16	216	10000001	216	00000001
17	217	10000011	217	00010011

WRITE CHECK With Error

Uses octal addresses 220-237. Has the same pattern as 200-217.

GET STATUS With Error

Uses octal addresses 240-257. Has the same pattern as 200-217.

SEEK With Error

Uses octal addresses 260-277. Has the same pattern as 200-217.

READ HEADER With Error

Uses octal addresses 300-317. Has the same pattern as 200-217.

WRITE With Error

Uses octal addresses 320-337. Has the same pattern as 200-217.

READ With Error

Uses octal addresses 340-357. Has the same pattern as 200-217.

READ WITHOUT HEADER COMPARE With Error

Uses octal addresses 360-377. Has the same pattern as 200-217.

B.2 FORMAT ROM

The format ROM consists of two chips. The upper chip is a 512 × 8 ROM and the lower chip is a 256 × 8 ROM. They are addressed as shown below.

Upper ROM Chip	Lower ROM Chip
A0 = SWC 2^0	A0 = SWC 2^0
A1 = SWC 2^1	A1 = SWC 2^1
A2 = SWC 2^2	A2 = SWC 2^2
A3 = SWC $2^3*2^4*2^5*2^6=0$	A3 = SWC $2^3*2^4*2^5*2^6=0$
A4 = SWC 2^7	A4 = SWC 2^7
A5 = HEADER FOUND H	A5 = FUNC 0 H
A6 = FUNC 0 H	A6 = FUNC 1 H
A7 = FUNC 1 H	A7 = FUNC 2 H
A8 = FUNC 2 H	

This addressing scheme uses A8:6 on the upper chip and A7:5 on the lower chip to divide the memory into eight areas, one for each of the eight possible combinations of FUNC 2:0 signals from the function ROM. Basically this corresponds to one area for each of the eight operations. (However, two of the operations do not require use of the format ROM so these two areas are not used). Within the upper ROM there are two areas used with each operation. One of these is selected before the HEADER FOUND condition occurs and the other one is selected when (if) the HEADER FOUND condition is generated. Each area is divided into 32 locations as defined by the Sector Word Counter (SWC). As the SWC counts from 000 toward decimal 143 (although it will never get that far before being interrupted), it selects locations within the area. Note that as it counts from decimal 008 through 127, the ROM addressing repeatedly cycles from octal X10 through X17 (X50 through X57). The map shows the SWC count in decimal and the address in octal.

The following ROM map divides the memory according to areas as defined by FUNC 2:0, and shows the ROM address and data. At any one count of the SWC there is one location of the lower ROM chip selected and one location of the upper ROM chip selected. In order to determine which of the two upper ROM locations shown is the one selected, it is necessary to consider the state of HEADER FOUND.

The data stored in the ROM corresponds to control signals as shown below.

Upper ROM Chip	Lower ROM Chip
D0 = not used	D0 = RD DATA AREA H
D1 = LOOP LOCK H	D1 = ENA DIFF CLK H
D2 = RD ENA H	D2 = WRITE GATE H
D3 = ENA BC CLK H	D3 = ENA SILO SER H
D4 = HDR POST H	D4 = ENA SYNC 1 H
D5 = POST H	D5 = ENA CRC OUT L
D6 = STR HDR FND L	D6 = ENA WRT CHK H
D7 = TRANSFER DONE H	D7 = WRT DATA AREA H

FUNC 2:0 = 0 Not Used

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	000	01000000	040	01000000	000	00100000
001	001	01000000	041	01000000	001	00100000
002	002	01000000	042	01000000	002	00100000
003	003	01000000	043	01000000	003	00100000
004	004	01000000	044	01000000	004	00100000
005	005	01000000	045	01000000	005	00100000
006	006	01000000	046	01000000	006	00100000
007	007	01000000	047	01000000	007	00100000
008	010	01000000	050	01000000	010	00100000
009	011	01000000	051	01000000	011	00100000
010	012	01000000	052	01000000	012	00100000
011	013	01000000	053	01000000	013	00100000
012	014	01000000	054	01000000	014	00100000
013	015	01000000	055	01000000	015	00100000
014	016	01000000	056	01000000	016	00100000
015	017	01000000	057	01000000	017	00100000
*						
*						
128	020	01000000	060	01000000	020	00100000
129	021	01000000	061	01000000	021	00100000
130	022	01000000	062	01000000	022	00100000
131	023	01000000	063	01000000	023	00100000
132	024	01000000	064	01000000	024	00100000
133	025	01000000	065	01000000	025	00100000
134	026	01000000	066	01000000	026	00100000
135	027	01000000	067	01000000	027	00100000
136	030	01000000	070	01000000	030	00100000
137	031	01000000	071	01000000	031	00100000
138	032	01000000	072	01000000	032	00100000
139	033	01000000	073	01000000	033	00100000
140	034	01000000	074	01000000	034	00100000
141	035	01000000	075	01000000	035	00100000
142	036	01000000	076	01000000	036	00100000
143	037	01000000	077	01000000	037	00100000

FUNC 2:0 = 1 Not Used

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	100	01000000	140	01000000	040	00100000
001	101	01000000	141	01000000	041	00100000
002	102	01000000	142	01000000	042	00100000
003	103	01000000	143	01000000	043	00100000
004	104	01000000	145	01000000	044	00100000
005	105	01000000	145	01000000	045	00100000
006	106	01000000	146	01000000	046	00100000
007	107	01000000	147	01000000	047	00100000
008	110	01000000	150	01000000	050	00100000
009	111	01000000	151	01000000	051	00100000
010	112	01000000	152	01000000	052	00100000
011	113	01000000	153	01000000	053	00100000
012	114	01000000	154	01000000	054	00100000
013	115	01000000	155	01000000	055	00100000
014	116	01000000	156	01000000	056	00100000
015	117	01000000	157	01000000	057	00100000
*						
*						
128	120	01000000	160	01000000	060	00100000
129	121	01000000	161	01000000	061	00100000
130	122	01000000	162	01000000	062	00100000
131	123	01000000	163	01000000	063	00100000
132	124	01000000	164	01000000	064	00100000
133	125	01000000	165	01000000	065	00100000
134	126	01000000	166	01000000	066	00100000
135	127	01000000	167	01000000	067	00100000
136	130	01000000	170	01000000	070	00100000
137	131	01000000	171	01000000	071	00100000
138	132	01000000	172	01000000	072	00100000
139	133	01000000	173	01000000	073	00100000
140	134	01000000	174	01000000	074	00100000
141	135	01000000	175	01000000	075	00100000
142	136	01000000	176	01000000	076	00100000
143	137	01000000	177	01000000	077	00100000

FUNC 2:0 = 2 Used During READ HEADER

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	200	01000000	240	01000000	100	00100000
001	201	01000000	241	01000000	101	00100000
002	202	01000000	242	01000000	102	00100000
003	203	01000000	243	01000000	103	00100000
004	204	01000000	244	01000000	104	00100000
005	205	01000000	245	01000000	105	00100000
006	206	01000000	247	01000000	106	00100000
007	207	01000000	247	01000000	107	00100000
008	210	01001010	250	01001010	110	00100000
009	211	01001010	251	01001010	111	00100000
010	212	01000110	252	01000110	112	00100001
011	213	01000110	253	01000110	113	00100001
012	214	01000110	254	01000110	114	00100000
013	215	01111000	255	01111000	115	00100000
014	216	11000000	256	11000000	116	00100000
015	217	01000000	257	01000000	117	00100000
*						
*						
128	220	01000000	260	01000000	120	00100000
129	221	01000000	261	01000000	121	00100000
130	222	01000000	262	01000000	122	00100000
131	223	01000000	263	01000000	123	00100000
132	224	01000000	264	01000000	124	00100000
133	225	01000000	265	01000000	125	00100000
134	226	01000000	266	01000000	126	00100000
135	227	01000000	267	01000000	127	00100000
136	230	01000000	270	01000000	130	00100000
137	231	01000000	271	01000000	131	00100000
138	232	01000000	272	01000000	132	00100000
139	233	01000000	273	01000000	133	00100000
140	234	01000000	274	01000000	134	00100000
141	235	01000000	275	01000000	135	00100000
142	236	01000000	276	01000000	136	00100000
143	237	01000000	277	01000000	137	00100000

FUNC 2:0 = 3 Used During SEEK

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	300	01000000	340	01000000	140	00100000
001	301	01000000	341	01000000	141	00100000
002	302	01000000	342	01000000	142	00100000
003	303	01000000	343	01000000	143	00100000
004	304	01000000	344	01000000	144	00100000
005	305	01000000	345	01000000	145	00100000
006	306	01000000	346	01000000	146	00100000
007	307	01000000	347	01000000	147	00100000
008	310	01001000	350	01001000	150	00100000
009	311	01001000	351	01001000	151	00100000
010	312	01001000	352	01001000	152	00100000
011	313	01001000	353	01001000	153	00100000
012	314	01001000	354	01001000	154	00100000
013	315	01001000	355	01001000	155	00100010
014	316	01001000	356	01001000	156	00100000
015	317	11000000	357	11000000	157	00100000
*						
*						
128	320	01000000	360	01000000	160	00100000
129	321	01000000	361	01000000	161	00100000
130	322	01000000	362	01000000	162	00100000
131	323	01000000	363	01000000	163	00100000
132	324	01000000	364	01000000	164	00100000
133	325	01000000	365	01000000	165	00100000
134	326	01000000	366	01000000	166	00100000
135	327	01000000	367	01000000	167	00100000
136	330	01000000	370	01000000	170	00100000
137	331	01000000	371	01000000	171	00100000
138	332	01000000	372	01000000	172	00100000
139	333	01000000	373	01000000	173	00100000
140	334	01000000	374	01000000	174	00100000
141	335	01000000	375	01000000	175	00100000
142	336	01000000	376	01000000	176	00100000
143	337	01000000	377	01000000	177	00100000

FUNC 2:0 = 4 Used During WRITE CHECK

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	400	01000000	440	01000110	200	01100000
001	401	01000000	441	01000110	201	01100000
002	402	01000000	442	01000110	202	01100000
003	403	01000000	443	01000110	203	01100000
004	404	01000000	444	01000110	204	01100000
005	405	01000000	445	01000110	205	01100000
006	406	01000000	446	01000110	206	01100000
007	407	01000000	447	01000110	207	01100000
008	410	01001010	450	01000000	210	00100000
009	411	01001010	451	01000000	211	00100000
010	412	01000110	452	01000000	212	00100000
011	413	00000110	453	00000110	213	00100000
012	414	01000000	454	01000110	214	00100000
013	415	01000000	455	01111010	215	00100000
014	416	01000000	456	01001010	216	00100000
015	417	01000000	457	01001010	217	00100000
*						
*						
128	420	01000000	460	01000110	220	00100000
129	421	01000000	461	01101000	221	00100000
130	422	11001000	462	11001000	222	00100000
131	423	01000000	463	01000000	223	00100000
132	424	01000000	464	01000000	224	00100000
133	425	01000000	465	01000000	225	00100000
134	426	01000000	466	01000000	226	00100000
135	427	01000000	467	01000000	227	00100000
136	430	01000000	470	01000110	230	01100000
137	431	01000000	471	01000110	231	01100000
138	432	01000000	472	01000110	232	01100000
139	433	01000000	473	01000110	233	01100000
140	434	01000000	474	01000110	234	01100000
141	435	01000000	475	01000110	235	01100000
142	436	01000000	476	01000110	236	01100000
143	437	01000000	477	01000110	237	01100000

FUNC 2:0 = 5 Used During WRITE

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	500	01000000	540	01001000	240	10101100
001	501	01000000	541	01001000	241	10101100
002	502	01000000	542	01001000	242	10101100
003	503	01000000	543	01001000	243	10101100
004	504	01000000	544	01001000	244	10101100
005	505	01000000	545	01001000	245	10101100
006	506	01000000	546	01001000	246	10101100
007	507	01000000	547	01001000	247	10101100
008	510	01001010	550	01000000	250	00100000
009	511	01000110	551	01000000	251	00100000
010	512	00000110	552	00000110	252	00100000
011	513	01000000	553	01000110	253	00100000
012	514	01000000	554	01111000	254	00100000
013	515	01000000	555	01001000	255	00100100
014	516	01000000	556	01001000	256	00100100
015	517	01000000	557	01001000	257	00110100
*						
*						
128	520	01000000	560	01001000	260	10000100
129	521	01000000	561	01001000	261	00100100
130	522	11001000	562	11001000	262	00100100
131	523	01000000	563	01000000	263	00100000
132	524	01000000	564	01000000	264	00100000
133	525	01000000	565	01000000	265	00100000
134	526	01000000	566	01000000	266	00100000
135	527	01000000	567	01000000	267	00100000
136	530	01000000	570	01001000	270	10101100
137	531	01000000	571	01001000	271	10101100
138	532	01000000	572	01001000	272	10101100
139	533	01000000	573	01001000	273	10101100
140	534	01000000	574	01001000	274	10101100
141	535	01000000	575	01001000	275	10101100
142	536	01000000	576	01001000	276	10101100
143	537	01000000	577	01001000	277	10101100

FUNC 2:0 = 6 Used During READ

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	600	01000000	640	01000110	300	00100001
001	601	01000000	641	01000110	301	00100001
002	602	01000000	642	01000110	302	00100001
003	603	01000000	643	01000110	303	00100001
004	604	01000000	644	01000110	304	00100001
005	605	01000000	645	01000110	305	00100001
006	606	01000000	646	01000110	306	00100001
007	607	01000000	647	01000110	307	00100001
008	610	01001010	650	01000000	310	00100000
009	611	01001010	651	01000000	311	00100000
010	612	01000110	652	01000000	312	00100000
011	613	00000110	653	00000110	313	00100000
012	614	01000000	654	01000110	314	00100000
013	615	01000000	655	01111010	315	00100000
014	616	01000000	656	01001010	316	00100000
015	617	01000000	657	01001010	317	00100000
*						
*						
128	620	01000000	660	01000110	320	00100000
129	621	01000000	661	01101000	321	00100000
130	622	01000000	662	11000000	322	00100000
131	623	01000000	663	01000000	323	00100000
132	624	01000000	664	01000000	324	00100000
133	625	01000000	665	01000000	325	00100000
134	626	01000000	666	01000000	326	00100000
135	627	01000000	667	01000000	327	00100000
136	630	01000000	670	01000110	330	00100001
137	631	01000000	671	01000110	331	00100001
138	632	01000000	672	01000110	332	00100001
139	633	01000000	673	01000110	333	00100001
140	634	01000000	674	01000110	334	00100001
141	635	01000000	675	01000110	335	00100001
142	636	01000000	676	01000110	336	00100001
143	637	01000000	677	01000110	337	00100001

FUNC 2:0 = 7 Used During READ WITHOUT HEADER CHECK

SWC	Upper ROM Chip W/O Header Found		Upper ROM Chip With Header Found		Lower ROM Chip	
	ADD	DATA	ADD	DATA	ADD	DATA
000	700	01000110	740	01000110	340	00100001
001	701	01000110	741	01000110	341	00100001
002	702	01000110	742	01000110	342	00100001
003	703	01000110	743	01000110	342	00100001
004	704	01000110	744	01000110	344	00100001
005	705	01000110	745	01000110	345	00100001
006	706	01000110	746	01000110	346	00100001
007	707	01000110	747	01000110	347	00100001
008	710	01001010	750	01001010	350	00100000
009	711	01001010	751	01001010	351	00100000
010	712	01000110	752	01000110	352	00100000
011	713	01000110	753	01000110	353	00100000
012	714	01000110	754	01000110	354	00100000
013	715	01001010	755	01001010	355	00100000
014	716	01001010	756	01001010	356	00100000
015	717	01001010	757	01001010	357	00100000
*						
*						
128	720	01000110	760	01000110	360	00100000
129	721	01101000	761	01101000	361	00100000
130	722	11001000	762	11001000	362	00100000
131	723	01000000	763	01000000	363	00100000
132	724	01000000	764	01000000	364	00100000
133	725	01000000	765	01000000	365	00100000
134	726	01000000	766	01000000	366	00100000
135	727	01000000	767	01000000	367	00100000
136	730	01000110	770	01000110	370	00100001
137	731	01000110	771	01000110	371	00100001
138	732	01000110	772	01000110	372	00100001
139	733	01000110	773	01000110	373	00100001
140	734	01000110	774	01000110	374	00100001
141	735	01000110	775	01000110	375	00100001
142	736	01000110	776	01000110	376	00100001
143	737	01000110	777	01000110	377	00100001

RL11 Controller
Technical Description Manual
EK-ORL11-TD-001

Reader's Comments

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use?

What features are most useful?

What faults or errors have you found in the manual?

Does this manual satisfy the need you think it was intended to satisfy?

Does it satisfy *your* needs? Why?

□ Please send me the current copy of the *Technical Documentation Catalog*, which contains information on the remainder of DIGITAL's technical documentation.

Name
Title
Company
Department

Street
City
State/Country
Zip

Additional copies of this document are available from:

Digital Equipment Corporation
444 Whitney Street
Northboro, Ma 01532
Attention: Communications Services (NR2/M15)
Customer Services Section

Order No. EK-ORL11-TD-001

— — — — — — — — — — — Fold Here — — — — — — — — — — — —

— — — — — — — — Do Not Tear - Fold Here and Staple — — — — — — — —

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD, MA.

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
301 Rockrimmon Boulevard South
Colorado Springs, Colorado 80919

Educational Services
Development and Publishing