bc336VME
Time Code Processor
8500 – 0006

User's Guide
Rev. A
March 13, 2000

This Page Intentionally Left Blank

bc336VME
Time Code Processor

TABLE OF CONTENTS

SECTION **PAGE**

CHAPTER ONE
INTRODUCTION

CHAPTER TWO
INSTALLATION AND SETUP

CHAPTER THREE
OPERATION AND SOFTWARE INTERFACE

SECTION **PAGE**

SECTION **PAGE**

CHAPTER SEVEN
PROGRAMMING EXAMPLES

CHAPTER EIGHT
DRAWING SET

This Page Intentionally Left Blank.

CHAPTER ONE
INTRODUCTION

1.0 GENERAL

The bc336VME VMEbus Time Code Processor Operation and Technical Manual provides the following information:

- General Introduction
- Installation and Setup Details
- Operation and Software Interface Details
- I/O Signal Information
- Theory of Operation
- Programming Examples
- Drawing Set

1.1 KEY FEATURES

The salient features of the bc336VME Time Code Processor include:

- Decodes commonly used time code formats (modulated or DC Level Shift): IRIG B, 2137, MILA, NASA 36.

- Continues to provide time during loss of input time code.

- Provides microsecond resolution.

- Allows synchronization of the internal real time clock to the external 1 Pulse Per Second signal and Master/Slave operation using Digital Time Transmission Protocol.

- Allows time capture via an external event trigger input with a rate up to 2000 hertz.

- Provides a rate programmable heartbeat pulse from 1 to 2000 hertz.

- Provides programmable propagation delay compensation.

- Functions as an A16:D08(O) slave with flexible interrupt capabilities.

- It's 4K byte block can be located on any 4K byte boundary in the VMEbus short address space.

- Accommodates 6U racks. Provides both front panel and P2 I/O connections.

- Includes LED display with the decoded time (hours through seconds).

1.2 bc336VME OVERVIEW

The bc336VME is a double height (6U) VMEbus module designed to decode serial time code signals and provide additional capabilities not normally found on a single board time code reader. The module consists of a 6U printed circuit.

The operation of the bc336VME is controlled by registers written and read by the host via VMEbus A16:D08(O) data transfers. These registers are used for:

- Defining the time code translation modes of operation.
- Activating time capture operations.
- Holding the captured time and status.
- Defining on-board interrupt priority levels and vectors.
- Defining the heartbeat interval rate.
- Defining the propagation delay compensation value.

The principal performance characteristics are listed in Table 1-1. The bc336VME is shown in Figure 1-1.

Table 1-1: bc336VME Performance Specifications

Item	Description
Time Code Reader	
Time Code Formats	IRIG B; 2137, NASA 36, MILA (Modulated or DC Level Shift)
Carrier Range (For Modulated Signal)	IRIG B 1 kHz " 2% 2137 1 kHz " 2% NASA 36 1 kHz " 2% MILA 1 kHz " 2%
Flywheel Accuracy	< 3.6 ms per hour
Modulation Ratio	3:1 to 6:1
Input Amplitude	500 mV to 5 V p-p
Input Impedance	>10 K Ohms

Table 1-1: bc336VME Performance Specifications (continued)

Item	Description
VMEbus Interface	
Standardization	IEEE 1014-1987 Revision C.1 of the VMEbus Spec
Address Space	A16, AM Codes $29 and $2D 4K contiguous bytes
Data Transfer	D08(O)
Interrupter	D08(O), I(1-7), ROAK
Power	+5 VDC @ 600 mA
TTL/CMOS Input Signals	
Event Capture	TTL/CMOS, Positive or Negative edge
1 Pulse Per Second	TTL/CMOS, Positive edge on time
TTL/CMOS Output Signals	
1 Pulse Per Second	TTL/CMOS, Positive edge on time, 32 msec width (approx)
Heartbeat Pulse	TTL/CMOS, Positive edge on time, 20 to 30 msec width
Load Rate (recomended)	
Time Request Rate	2000 Hz
Event Capture Rate	2000 Hz
Heartbeat Rate	2000 Hz
Maximum Activity Rate (all of the above combined)	3000 Hz
Operating and Storage Environments	
Temperature	
Operating	0^{o} to 70^{o} C
Non-Operating	-50^{o} to 125^{o} C
Relative Humidity	
Operating	10% to 80% (non-condensing)
Non-Operating	5% to 95% (non-condensing)
Mean Time Between Failure	200,000 hours

1.3 RELIABILITY

Reliability stress analysis and prediction were performed on the bc336VME board. Results were obtained using the part stress analysis prediction method of paragraph 5.1 of MIL-HDBK-217E. The analysis results apply to the board when being utilized in Ground, Benign (G_B) environment where the ambient air temperature is 77^o F. When subject to continuous operation under these conditions, the board has a predicted failure rate of 4.95310 failures/million hours which is equivalent to a predicted mean time between failure (MTBF) of 201,894 hours.

1.4 TIME CODE FORMATS

The widespread use of coded timing signals to assist in the correlation of intercept and test data began in the early 1950's. These signals can be decoded in real time to indicate the current Time of Day (TOD) or recorded along with intercept/test data on magnetic tape recorders for post processing and time correlation. Hundreds of time code formats were developed - one for each agency involved. During the early 1960's the Inter Range Instrumentation Group promoted a series of 'standard' time code formats now loosely referred to as 'IRIG Time Codes'. The bc336VME decodes two of these formats: IRIG B.

More complete details on these and other time code formats is available free of charge, on request from Datum Inc in the form of the 'Datum Inc, Handbook of Time Code Formats'. Figure 1-2 illustrates a frame of IRIG B or G time code.

Figure 1-1
bc336VME VMEbus Time Code Processor

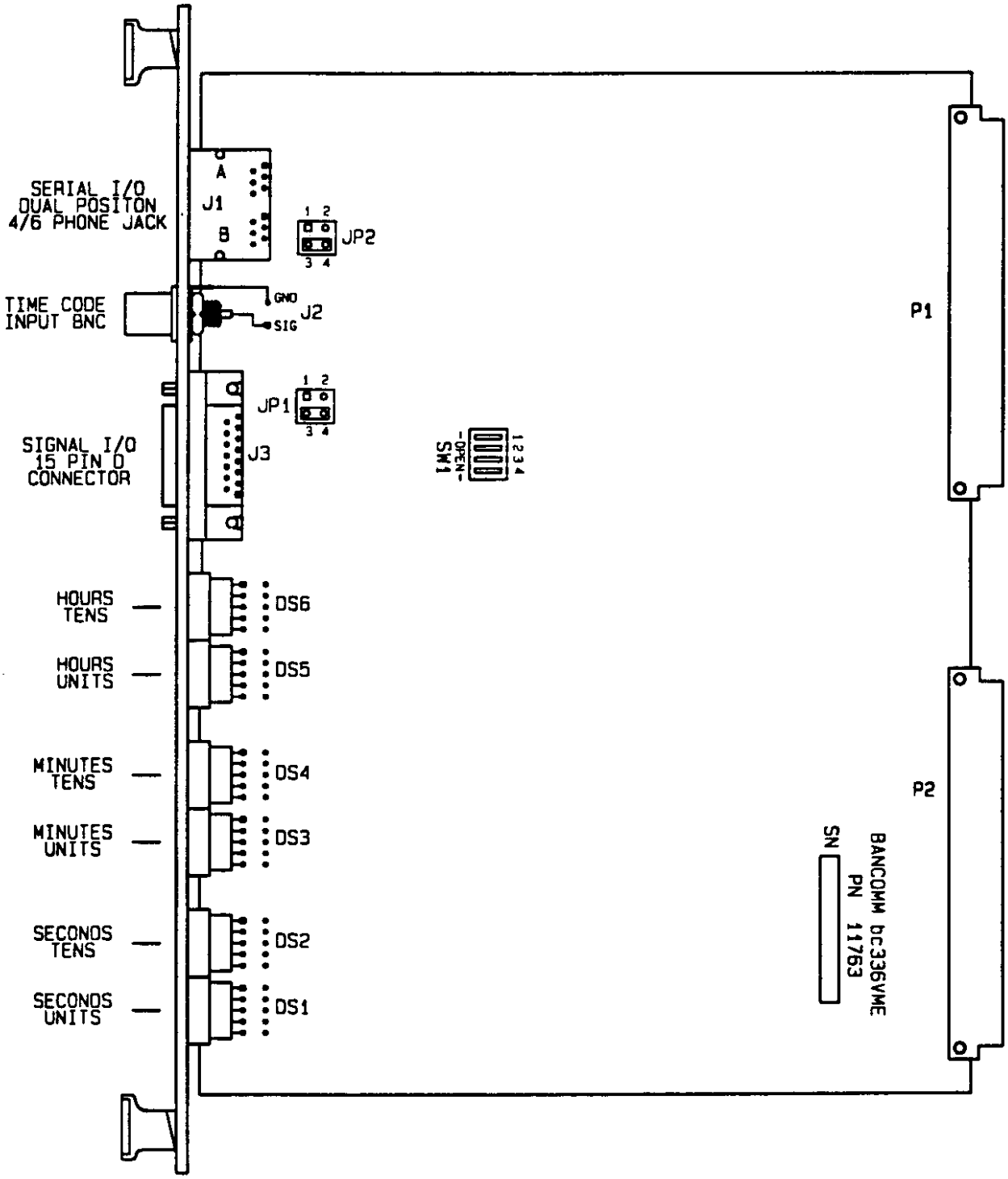
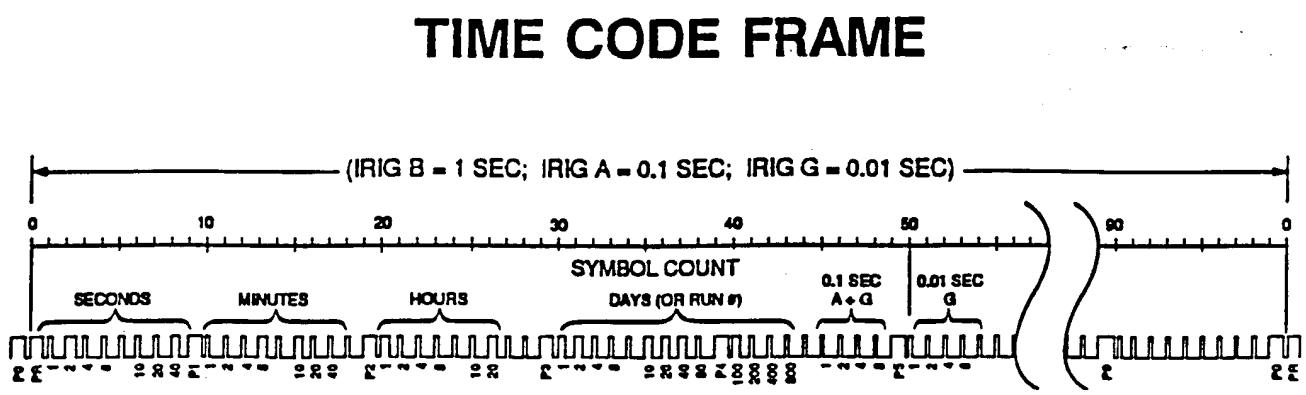


Figure 1-2
IRIG Time Code Frame



CHAPTER TWO
INSTALLATION AND SETUP

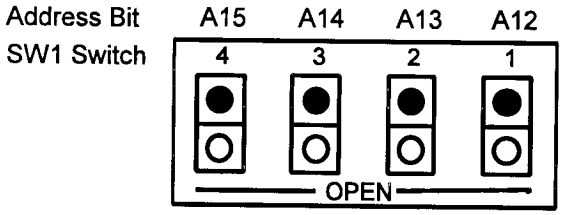
2.0 GENERAL

The bc336VME is a double height (6U) VMEbus board designed to be installed in a standard VMEbus subrack. This section details the steps required to install the module in the subrack. Also discussed are the steps necessary to connect a number of bc336VME boards in the network using the Digital Time Bus.

2.1 BASE ADDRESS SELECTION

Before installing the module in the subrack the address select DIP switch (SW1) must be set. The location of SW1 is shown on Figure 1-1. The bc336VME occupies 4K bytes in the VMEbus short address space and can be freely located on any 4K byte boundary. The 4 DIP switch positions of SW1 correspond to address bits A15 - A12 as shown in Figure 2-1 and determine the base address for the module. The base address is defined as the address selected by the SW1 DIP switch when A11 - A1 are 0.

Figure 2-1
DIP Switch SW1



To select a base address, set each of the 4 DIP switches to the ON (same as CLOSED) or OFF (same as OPEN) position. Setting a DIP switch to the ON position selects a logical 0 for that address bit and the OFF position selects a logical 1.

The bc336VME responds to address modifiers $2D (Short Supervisory Access) and $29 (Short Non-Privileged Access) as decoded by the U17 address modifier decoder PLD. This decoder PLD can be modified by the factory to decode different and/or additional address modifiers. Consult the factory for custom address modifier decoding.

2.2 TIME CODE SELECTION

The bc336VME can decode both modulated and DC level shift time code signals. In order to select the type of the incoming signal, jumper JP1 should be set according to Table 2-2. Location of the jumper JP1 is shown on Figure 1-1.

Table 2-2
Jumper JP1 Setting

1 - 2	DC level shift signal
3 - 4	Modulated signal (Factory Setting)

2.3 INSTALLATION PROCEDURE

Once the base address has been selected the bc336VME is ready to be installed in the VMEbus subrack. Install the bc336VME as follows:

- Remove the IACKIN*/IACKOUT* backplane jumper for the bc336VME slot. This step should be done even if you will not be using interrupts from the bc336VME.

- Verify that the power to the subrack is turned off before inserting the bc336VME module into the subrack.

- Insert the bc336VME into the subrack slot and secure the board in this slot by tightening the 2 front panel screws.

2.4 SERIAL TIME NETWORK INSTALLATION PROCEDURE

The bc336VME has the capability of operating in a Master/Slave mode in which Slave bc336VME's synchronize digitally to the time base of a Master bc336VME using the Digital Time Transmission Protocol. This allows synchronization of two or more boards without an external time source. Boards could be connected in the network using a daisy-chain or bus architecture.

2.4.1 DAISY-CHAIN ARCHITECTURE

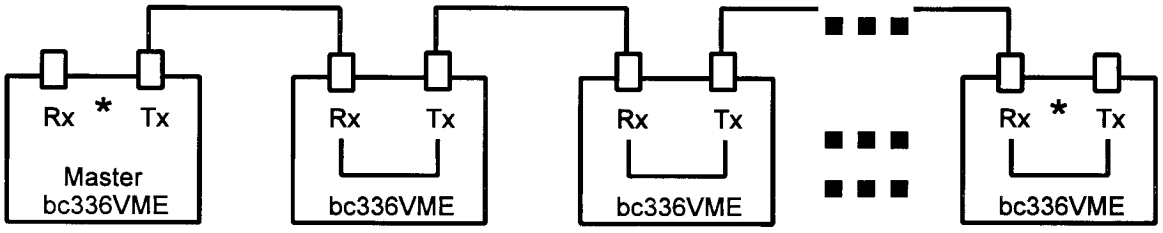
A limited number of devices can be connected in the network using daisy-chaining as shown in Figure 2-1. Connections are made using a 6-conductor RJ11 jack. The cable has an RJ11 plug on each end, wired as described in Table 2-3. This is a wiring of a standard phone cable. Conductors on pins 1 and 6 are not used and they can be omitted from the cable. Daisy-chaining more than five devices is not recommended.

Note that ends of the line should be terminated using internal 100 Ohm resistors as described in the Section 2.4.3.

Table 2-3
Connection Cable Pin Layout

JACK	PIN NUMBERS					
A	1	2	3	4	5	6
B	6	5	4	3	2	1

Figure 2-1
Serial Daisy-Chain Connection of Master/Slave bc336VME's

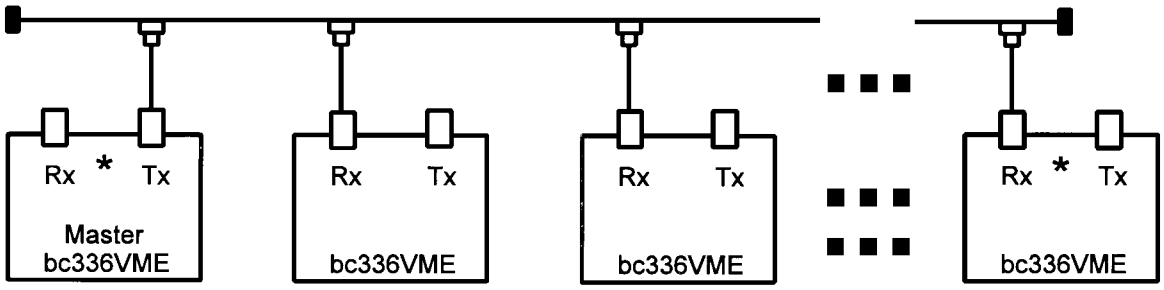


* Denotes termination by on-board 100 ohm resistors

— 4- or 6-conductor phone cable with RJ11 male connectors

bc336VME RJ11 Port (Female mating connector)

Figure 2-2
bc336VME Master/Slave Serial Bus with Multiple Drop Points



bc336VME RJ11 Port (Female mating connector)

4- or 6-conductor phone cable with RJ11 male connectors

Wall mount junction connector with female RJ11 mating connector

Denotes direct 100 ohm termination of the bus

* Denotes termination by on-board 100 ohm resistors

2.4.2 BUS ARCHITECTURE

A large number of devices should be connected using bus architecture as shown on Figure 2-2. Up to 32 devices may be connected this way with a maximum cable length of 4000 feet. The main bus should be a shielded twisted pair of wires to increase noise immunity. Stubs off the main wire should be kept as short as possible.

Ends of the bus should be terminated using external 100 Ohm resistors or using on-board resistors as described in Section 2.4.3 (necessary if bus is not terminated externally). Only one of the described methods should be used.

2.4.3 NETWORK TERMINATION

In order to use the Digital Time Transmission Protocol ends of the network should be terminated with 100 Ohm resistors. The resistor may be external or internal to the bc336VME. To use the internal resistor a jumper should be placed between pins 1 and 2 on JP2. Location of the jumper JP2 is shown on Figure 1-1.

Table 2-4
Jumper JP2 Setting

1 - 2	Digital Time Bus terminated
3 - 4	Digital Time Bus not terminated (Factory Setting)

This Page Intentionally Left Blank.

CHAPTER THREE
OPERATION AND SOFTWARE INTERFACE

3.0 GENERAL

The bc336VME occupies 4K bytes in the VMEbus short address space (2K D08(O) memory locations). See Section 2.1 for details on base address selection. All data transfers between the VMEbus and the bc336VME are D08(O) type (single odd byte transfers).

3.1 REGISTERS

This section describes the registers used on the bc336VME for controlling its operation and transferring time data. Section 3.2 details the use of these registers. The Control Register memory map for the bc336VME is listed in Table 3-1. The memory map for the rest of the bc336VME registers is listed in Table 3-2. The first column of these tables shows the offset from the base address of each register. The value of each register following a VMEbus SYSRESET* is shown where '--' indicates that the register contents is undefined. A label for each register is listed as is a brief description of the register's function.

3.1.1 CONTROL REGISTERS AND WARM START REGISTER

The Control Registers govern the operation of the bc336VME. A write of any non-zero value to the WARM START Register (offset 7FD) causes the bc336VME to perform a warm start reset. During a warm start reset, the bc336VME reads the Control Registers to determine its mode of operation. To control the operation of the bc336VME, first write the appropriate values to the control registers, then write any non-zero value to the WARM START Register. When the bc336VME is ready to accept new Control Register values, it will set the WARM START Register contents to zero. The Control Registers are described below. All values are in base hexadecimal.

TCSEL: Input Time Code Selection (offset 001)

- 00 = Automatically detects an IRIG B, 2137 or NASA 36 time code signal
- 42 = Decodes modulated IRIG B only
- 43 = Decodes 2137 only
- 4D = Decodes MILA only
- 4E = Decodes modulated NASA 36 only
- 52 = Begins flywheeling immediately
- 02 = Decodes DC level shift IRIG B only
- 06 = Decodes DC level shift NASA 36 only
- 03 = Synchronizes to the external 1 Pulse Per Second signal only
- 04 = Synchronizes to the Master Module using Digital Time Transmission Protocol only

OFFSET (HEX)	RESET VALUE	LABEL	DESCRIPTION
001	00	TCSEL	Time Code Format Select
003	00	TVINTEN	Time Valid Int Enable
005	00	EVENT	External Event Control
007	00	MODE	Mode Control
009	00	PROPDEL0	Propagation Delay (MSB)
00B	00	PROPDEL1	Propagation Delay (LSB)
021	00	HBCTRL	Heartbeat Control
023	03	HBRATE0	Heartbeat Rate (MSB)
025	E8	HBRATE1	Heartbeat Rate (LSB)
02D	01	FILTER	Filter Register
02F	02	FRMCNT	Frame Count
037	01	LEAP	Leap Year Flag
7FD	00	WARMSTRT	Warm Start Command

Table 3-1: bc336VME Control Register Memory Map

When TCSEL is set to 52 or to 03, the start time will be the time contained in REQTIME (Requested Time Data Block) when a WARM START is performed. The user should load REQTIME with the desired start time (see Section 3.1.6) and FERR0/1 with frequency adjust data if necessary (see Section 3.1.5) prior to issuing the WARM START command.

TVINTEN: Time Valid Interrupt Enable (offset 003)

00 = Disable time valid interrupt (interrupt source 3)
01 = Enable interrupt on Time Request time valid
02 = Enable interrupt on External Event Capture time valid
03 = Enable interrupt on either Bus Time Request or External Event Capture time valid

EVENT: External Event Control (offset 005)

00 = Disable External Event Capture
01 = Enable External Event Capture on the rising edge of the External Event Capture Input
02 = Enable External Event Capture on the falling edge of the External Event Capture Input

03 = Enable External Event Capture on both the rising and falling edges of the External Event Capture Input

MODE: Mode Control (offset 007)

00 = Read time code mode

01 = Board operates as a Master on the Digital Time Transmission Network. Selecting MODE equal 01 will conflict with the setting TCSEL to 04 meaning that the board can not operate as Master and Slave simultaneously. In this situation bc336VME will operate as a Slave. For details on Master/Slave operation refer to Section 3.5.

PROPDEL0 and PROPDEL1: Propagation Delay (offset 009 and 00B)

The propagation delay compensation function can offset the effects of long cables between the time code source and the bc336VME. These two registers are combined to produce a signed 16-bit quantity. The range of values supported is -2048 to +2047. Each unit represents a delay of 0.5? sec. For example, a value of 1000 would effect a delay of 500 msec. Positive values advance the 1 PPS (Pulse Per Second) and heartbeat pulses relative to the time code; negative values retard the 1 PPS and heartbeat pulses.

HBCTRL: Heartbeat Control (offset 021)

00 = Disable heartbeat pulses (output remains high)

01 = Enable heartbeat pulses at the rate defined by HBRATE0 and HBRATE1. The heartbeat feature is not available for IRIG A.

HBRATE0 and HBRATE1: Heartbeat Rate (offset 023 and 025)

The rate at which the heartbeat pulses are generated is controlled by these registers which are combined to produce a 16-bit unsigned value. After powering on the contents of these registers is 03E8 (1000 decimal). The rate of heartbeat pulses is in pulses per second. For example, a value of 1000 would generate 1000 pulses per second. Heartbeat rates are supported from 1 to 2000 pulses per second. Rates above 2000 are not recommended. The heartbeat pulses are synchronized to the time code. The bc336VME must be decoding time or flywheeling before the heartbeats are generated.

FILTER: Filter Register (offset 02D)

Bit 0

Setting this bit will enable filter that removes jitter from the output signals. By default this bit is set. It is recomended to turn this filter off if incoming time code signal has jitter of more than 20 msec.

Bits 1-7

Reserved for future use

FRMCNT: Frame Count (offset 02F)
The Frame Count Register sets up the number of consequent valid time code frames decoded before bc336VME will synchronize to the selected time source. Default value of this register is 02. Supported values are from 00 to 0F. Setting FRMCNT register to 00 disables the filter. Contents of the register does not affect operation of the bc336VME if TCSEL is set to 4D, 03, or 04.

LEAP: Leap Year Flag (offset 037)

00 = Disable yearly day count (days range from 000 to 999)
01 = Enable yearly day count (days range from 001 to 365)
02 = Enable yearly day count for the leap year (days range from 001 to 366)

3.1.2 TIME REQUEST REGISTER

The Time Request Register is used to request time from the bc336VME over the VMEbus. A read or write to this register causes the bc336VME to freeze the time and transfer it to the Requested Time Data Block so that the time can be read over the VMEbus.

3.1.3 STATUS BYTE

The bc336VME provides a status byte containing the status of the time code decoder. The status byte is organized as two 4-bit fields. The lower order field (bits 0-3) designates the last known time code format detected as follows:

0 = Warm Start complete, time code not yet found
1 = IRIG B Modulated
3 = MILA
4 = 2137
5 = NASA 36 Modulated
6 = IRIG B DC Level Shift
8 = NASA 36 DC Level Shift
9 = 1 Pulse Per Second Synchronization Mode
A = Digital Time Transmission Protocol

The upper field (bits 4-7) denotes the time code tracking status as follows:

0 = currently decoding time
1 = flywheeling

The term 'flywheeling' means that the input time code has been lost or is unusable, but that the bc336VME is still providing time, heartbeats, etc. as if the time code is still present. The bc336VME will indicate flywheeling mode by illuminating the decimal points on the time display.

3.1.4 TIME VALID FLAG REGISTER

The Time Valid Flag Register (offset 41F) is used to indicate when the time data blocks contain valid time data.

TVFLAG: Time Valid Flag Register (offset 41F)

Bit 0	=	1 when Time Request time is valid
Bit 1	=	1 when External Event Capture time is valid
Bit 2	=	1 when WARM START Command is complete
Bits 3-7	=	Not used

Once the user detects that a valid time is available, the time data block can be read. Once the time data is read the user must clear the appropriate bit in the Time Valid Flag Register. The bc336VME will only set these bits (except during a WARM START); the user must clear them. The user should clear bit 0 of the TVFLAG register before performing a time request.

Table 3-2
bc336VME Register Memory Map

OFFSET (HEX)	RESET VALUE	LABEL	DESCRIPTION
401-417	--	REQTIME	Requested Time Data Block
421-437	--	EVTIME	Ext Event Time Data Block
419	--	STATUS	Status Byte
41B	--	FERR0	Frequency Error Byte 0 (MSB)
41D	--	FERR1	Frequency Error Byte 1 (LSB)
41F	00	TVFLAG	Time Valid Flag
1F3	00	HOLDSTAT	MILA Hold Status Register
7FF	--	INT3ACK	INT Source 3 Acknowledge
C01	--	TIMEREQ	Time Request
801	00	INTCR0	INT Control Register 0
803	00	INTCR1	INT Control Register 1
805	00	INTCR2	INT Control Register 2
807	00	INTCR3	INT Control Register 3
809	1F	INTV0	INT Vector Register 0
80B	1F	INTV1	INT Vector Register 1
80D	1F	INTV2	INT Vector Register 2
80F	1F	INTV3	INT Vector Register 3
E01	--	PINTCLR0	Pending Interrupt Clear 0
E03	--	PINTCLR1	Pending Interrupt Clear 1
E05	--	PINTCLR2	Pending Interrupt Clear 2
E07	--	PINTCLR3	Pending Interrupt Clear 3

3.1.5 FREQUENCY ERROR DATA

The Frequency Error Data is used in conjunction with the subsecond count bytes to determine an accurate subsecond time. Section 3.3 describes the meaning and use of the Frequency Error Data. Both the Time Request and External Event Capture times use the same Frequency Error data.

Table 3-3
Time Data Block Format

OFFSET (HEX) REQ/EVENT	DATA
401/421	Days Hundreds (" for MILA)
403/423	Days Tens
405/425	Days Units
407/427	Hours Tens
409/429	Hours Units
40B/42B	Minutes Tens
40D/42D	Minutes Units
40F/42F	Seconds Tens
411/431	Seconds Units
413/433	Subsecond Count Byte 0 (MSB)
415/435	Subsecond Count Byte 1
417/437	Subsecond Count Byte 2 (LSB)

FERR0 and FERR1 must be loaded by the user to set the flywheel rate when the TCSEL byte is set to 52 (FERR0/1 are set by the bc336VME for all other values of TCSEL). A value of 33920 decimal causes the bc336VME crystal frequency to be used directly. A value of 33921 lowers the output rate by 5 parts in 10E7. A value of 33919 increases the output rate by 5 parts in 10E7. The bc336VME firmware does not initialize FERR0/1 at power up; these memory locations will contain undetermined values.

3.1.6 TIME REQUEST/EXTERNAL EVENT TIME DATA BLOCKS

Two separate blocks of data are used to hold time data. One block holds the time for a Time Request (from the VMEbus) and one block holds the time for an External Event Capture. The format of these time data blocks is shown in Table 3-3. The time digits for days hundreds through seconds units contain the values 00 - 09. The subsecond count bytes are described in Section 3.3. The requested time data block is loaded with zeros on power up.

For the MILA Countdown time code format the Days Hundreds digit (offset 401/421) holds the sign bit (MILA does not contain a days hundreds digit). The values for the sign bit are:

00 = + (plus time)
01 = - (minus time)

Table 3-4
Interrupt Control Register Format

BIT	7	6	5	4	3	2	1	0
FUNCTION	FLAG	FAC	X/IN	IRE	IRAC	L2	L1	L0

Table 3-5
Interrupt Level Setting

L2	L1	L0	IRQ LEVEL
0	0	0	DISABLED
0	0	1	IRQ1
0	1	0	IRQ2
0	1	1	IRQ3
1	0	0	IRQ4
1	0	1	IRQ5
1	1	0	IRQ6
1	1	1	IRQ7

3.1.7 MILA HOLD STATUS REGISTER

This register indicates Hold Mode while decoding MILA Time Code. The bc336VME sets it to 01 when Hold Mode is detected.

3.1.8 INTERRUPT CONTROL, VECTOR AND PINTCLR REGISTERS

The bc336VME supports four independent interrupt sources (interrupt source 0-3). Associated with each interrupt source are three registers; one Interrupt Control Register, one Interrupt Vector Register, and one Pending Interrupt Clear Register.

3.1.8.1 INTERRUPT CONTROL REGISTERS

The Interrupt Control Registers govern the operation of the VMEbus interrupts. There is one control register for each interrupt source, i.e. INTCR0 controls interrupt source 0, INTCR1 controls interrupt source 1, etc. The Interrupt Control Register format is shown in Table 3-4.

L2, L1, L0
Interrupt Level. The 3 interrupt level bits determine the level at which an interrupt will be generated as shown in Table 3-5.

IRAC
Interrupt Auto Clear. If the IRAC is set, IRE (Bit 4) is cleared during an interrupt acknowledge cycle responding to this request. This action of clearing IRE disables the interrupt request. To re-enable the interrupt associated with this register, IRE must be set again by writing to the control register.

IRE
Interrupt Enable. This bit must be set to 1 to enable the bus interrupt request associated with the control register.

X/IN
External/Internal Vector. This bit must be cleared to 0 in all cases.

FAC
Flag Auto Clear. If FAC is set to 1, the FLAG bit is automatically cleared during an interrupt acknowledge cycle.

FLAG
Flag Bit. This bit is a flag that can be used for processor-to-processor communication and resource allocation. The FLAG bit has no affect on the operation of the interrupts.

3.1.8.2 INTERRUPT VECTOR REGISTERS

Each of the four interrupt sources has associated with it an interrupt vector register. Interrupt source 0 uses INTV0, interrupt source 1 uses INTV1, etc. The 8 bit interrupt vector is supplied to the VMEbus during an interrupt acknowledge cycle. The four Interrupt Vector Registers are set to 0F at reset which corresponds to the MC68000 vector for an uninitialized interrupt vector.

3.1.8.3 PENDING INTERRUPT CLEAR REGISTERS

Associated with each of the four interrupt sources is a Pending Interrupt Clear Register. Interrupt source 0 uses PINTCLR0, interrupt source 1 uses PINTCLR1, etc. These registers are used to clear any pending interrupt signals before interrupts are enabled with the Interrupt Control Registers. A pending interrupt is cleared by reading or writing any value to the appropriate Pending Interrupt Clear Register. Before enabling interrupts with INTCRX, the user should clear any pending interrupts associated with interrupt source X.

3.1.9 INTERRUPT SOURCE 3 ACKNOWLEDGE REGISTER

Interrupt source 3 requires an acknowledge operation following the interrupt. Before another interrupt from source 3 can take place, the Interrupt Source 3 Acknowledge Register must be read. The contents of this register are meaningless. Interrupt sources 0-2 do not have corresponding interrupt acknowledge registers.

3.2 FUNCTIONAL DESCRIPTION

Section 3.1 provided an overview of the bc336VME registers and their function. This section provides a description of how these registers are used to achieve the desired functions.

3.2.1 CONTROLLING THE OPERATING MODE OF THE BC336VME

To control the operating mode of the bc336VME, first write the appropriate data to the Control Registers. Then, to cause the bc336VME to take action on the Control Register data, clear bit 2 of the TVFLAG Register to 0 and write any non-zero value to the WARM START Register. This action will cause the bc336VME to perform a warm reset and to act on the data in the Control Registers. When the bc336VME is ready to accept another WARM START command, it will clear the contents of the WARM START Register to 0 and set bit 2 of the TVFLAG Register to 1. Do not initiate a warm reset unless bit 2 of the TVFLAG Register is set.

3.2.2 REQUESTING AND READING TIME

To request time from the bc336VME over the VMEbus, first clear bit 0 of the Time Valid Flag, then access (read or write) the Time Request Register (offset C01). This action causes the bc336VME to freeze the time and transfer it to the Time Request Data Block (offset 401 - 417). Following the transfer of time to the Time Request Block, the bc336VME will set bit 0 of the Time Valid Flag Register (offset 41F), and if the Time Valid Interrupt Enable byte in the control block is set to 01 or 03 the bc336VME will generate a VMEbus interrupt (interrupt source 3). The interrupt registers must also be setup for interrupt source 3 for an interrupt to take place. The Time Request Data Block can then be read. Before a subsequent time request, bit 0 of the Time Capture Flag Register must be cleared.

The bc336VME will take 140 msec to load the Time Request Data Block with the time in response to a Time Request. The time, however, is frozen at the instant that the Time Request Register is accessed.

3.2.3 EXTERNAL EVENT CAPTURE

The bc336VME has the ability to capture time and generate an interrupt in response to an external event. Use the External Event Control Register (offset 005) in the control block to enable the external event function and to select the active edge for capture. When the bc336VME detects the appropriate edge of the event signal it will freeze the time and load the Event Capture Time Data Block (offset 421 - 437) with the time. Following the time data transfer, the bc336VME will set bit 1 of the Time Valid Flag Register (offset 41F), and if the Time Valid Interrupt Enable byte in the control block is set to 02 or 03 the bc336VME will generate a VMEbus interrupt (interrupt source 3). The interrupt registers must also be setup for interrupt source 3 for an interrupt to take place. The Event Capture Time Data Block can then be read. After reading the time, bit 1 of the Time Valid Flag Register must be cleared. The bc336VME will not overwrite the Event Capture Time Block unless it detects that bit 1 of the Time Valid Flag Register has been cleared. This prevents Event Time data from being destroyed by a subsequent event before the user has a chance to read it. The user should be certain that bit 1 of the Time Valid Flag is cleared before starting External Event Captures.

The bc336VME will take 140 msec to load the Event Capture Data Block with the time in response to an External Event. The time, however, is frozen at the instant that the External Event Input is activated. The bc336VME supports Event Capturing rates up to 2000 events per second. Using the bc336VME for capturing events with rates above 2000 is not recommended.

3.2.4 RATE PROGRAMMABLE HEARTBEAT PULSES

It is often useful to generate a periodic pulse (heartbeat) which is synchronized to the time code signal. The bc336VME has the ability to generate this heartbeat. The control register HBCTRL enables the heartbeat, and the registers HBRATE0 and HBRATE1 determine the rate of the pulses (in pulses per second). Following a change to the heartbeat control and rate registers, the user must perform a bc336VME warm start as described above. The bc336VME must be decoding time or flywheeling (see Section 3.1.3) before the Heartbeat Pulses are started or else synchronization of the pulses to the time code will not occur. Heartbeat rates are supported from 1 to 2000 pulses per second. Heartbeat rates above 2000 are not recommended. The heartbeat pulses are available on the front panel I/O connector. The heartbeat can generate VMEbus interrupts (interrupt source 2) as well. The rising edge of the heartbeat occurs on time.

3.2.5 PROPAGATION DELAY COMPENSATION

When the time code source is located an appreciable distance from the bc336VME, a significant propagation delay will be introduced. The bc336VME incorporates a propagation delay compensation feature which removes the effects of this delay. To utilize this feature simply load the desired value into the PROPDEL0 and PROPDEL1 registers and then perform a bc336VME warm start.

The PROPDEL0 (MSB) and PROPDEL1 (LSB) registers are loaded with a 16-bit signed value between -2048 and +2047. Each unit represents 0.5 msec of delay, giving a range of delay values between -1000 msec and +1000 msec. Positive values advance the 1 PPS and heartbeat relative to the time code; negative values retard the 1 PPS and heartbeat. For example, if the cable between the time code source and the bc336VME introduces a delay of 50 msec, and you want to make the system act as if the time code source is sitting right next to the bc336VME, then load the value +100 into the propagation delay registers.

Propagation delay also can be used for the fine tuning of the board to the incoming time code. The oscillator on the board introduces some errors that can cause the outgoing 1 PPS signal to be off time (for up to 5msec for IRIG B). You can measure this time with the oscilloscope comparing the on-time mark of the time code against the outgoing 1 PPS signal. For example, if 1 PPS signal goes out 3? sec before the on-time mark, load propagation delay register with value +6 to compensate this error.

3.3 SUBSECOND COUNT AND FREQUENCY ERROR

Note: The notation 2E6 means 2 X 10^6 or 2,000,000.

The subsecond count in conjunction with the frequency error is used to derive an accurate subsecond time. The subsecond count is an unsigned 24-bit binary number and the Frequency Error is an unsigned 16-bit binary number. The subsecond count multiplied by 0.5msec will provide the nominal subsecond time. Therefore, the subsecond count varies nominally between 0 and 2E6 - 1 (1,999,999) each second. However, due to the difference between the bc336VME's crystal clock frequency and the frequency of the time code, the subsecond count can vary between 0 and some number greater than or less than 2E6-1. This variation in subsecond counts in a 1 second interval gives rise to the frequency error bytes.

If the bc336VME clock is exactly on frequency and the input time code time base is also perfect then the frequency error is 0. When the frequency error is 0 the frequency error bytes contain the value 33920 decimal (8480 hex.) If the frequency error bytes contain the value 34000 decimal (33920 + 80), for example, then the bc336VME time base is fast by 80 parts in 2E6, and the subsecond count would need to be scaled down by the ratio of 2E6 / (2E6 + 80).

Table 3-6
bc336VME Interrupt Sources

INT SOURCE	FUNCTION
0	1 Pulse Per Second
1	Future Implementation
2	Heartbeat Pulses
3	Ext Event/Time Request Time Valid

Likewise, if the frequency error bytes contain the value 33840 decimal (33920 - 80) then the bc336VME time base is slow by 80 parts in 2E6, and the subsecond count would need to be scaled up by the ratio of 2E6 / (2E6 - 80). In general, the subsecond count needs to be scaled by the following factor:

2E6 / (2E6 + (Frequency Error - 33920))

When scaled by the above factor, the subsecond count will always vary between 0 and 2E6-1, giving an accurate number of 0.5msec counts. This methodology allows the user to completely remove the effects of the bc336VME crystal clock frequency offset and aging.

The user may also simply divide the subsecond count by the factor (2E6 + (Frequency Error - 33920)) to produce a real number which varies between 0.0000000 and 0.9999995.

The programming examples in Chapter Section 7 provide some useful data structures and algorithms for working with the subsecond count and frequency error values.

3.4 VMEbus INTERRUPTS

The bc336VME provides 4 independent VMEbus interrupt sources as shown in Table 3-4. As described above, associated with each interrupt source are three registers; an Interrupt Control Register, a Vector Register, and a Pending Interrupt Clear Register. Each interrupt source can generate an interrupt on any one of the 7 interrupt request levels (IRQL) on the VMEbus. All four sources could also use the same IRQL. Additionally, each interrupt source can have its own unique interrupt vector. The following sections describe the operation of each interrupt source.

3.4.1 INTERRUPT SOURCE 0 (1 Pulse Per Second)

The bc336VME generates a 1 Pulse Per Second (1 PPS) signal. This signal generates a rising edge which occurs at the on time mark of the time code signal and is available on the J3 I/O connector. This 1 PPS signal can also be used to generate a VMEbus interrupt. Since this signal is always active, there will probably be an interrupt pending associated with it. Therefore, when the Interrupt Control Register is set to enable interrupts for this source, an interrupt would be immediately generated. To clear this pending interrupt, simply read or write the PINTCLR0 register before enabling interrupt source 0 with INTCR0.

3.4.2 INTERRUPT SOURCE 1 (Future Implementation)

3.4.3 INTERRUPT SOURCE 2 (Heartbeat Pulses)

The rate programmable heartbeat pulses can be used to generate VMEbus interrupts. Use INTCR2, INTV2, and PINTCLR2 to control this interrupt source. The interrupt is generated on the rising edge of the heartbeat pulse.

3.4.4 INTERRUPT SOURCE 3 (Time Valid)

Interrupt Source 3 is used to inform the user that a valid time (External Event or Time Request) has been loaded into the Time Data Block without having to poll the Time Valid Flag Register. The Time Valid Interrupt Control Register (offset 003) is used (along with INTCR3, INTV3, and PINTCLR3) to enable this interrupt. The user should clear any source 3 pending interrupts before enabling interrupts with INTCR3 by accessing the PINTCLR3 Register. Also, be sure to read the INT3ACK register after servicing the interrupt or a subsequent interrupt cannot take place.

3.5 DIGITAL TIME TRANSMISSION PROTOCOL

Digital time is a high-precision method of time transmission. With good local clocks synchronization to less than 5 microseconds may be achieved.

3.5.1 MASTER/SLAVE OPERATION

The bc336VME has the capability of operating in a Master/Slave mode in which Slave bc336VME's synchronize digitally to the time base of a Master bc336VME. The Master/Slave mode of operation is summarized as follows:

- A Master bc336VME decodes its selected time source. A 9600 baud serial Time Synchronization Message (synchronization character, days-microseconds) is broadcast each second normally through the TX side of the RJ11 connector (J1B). This message is broadcast automatically while the Master module decodes time or flywheels.

  During flywheeling the synchronization message may be broadcast less frequently (up to once each 3 seconds). The Time Synchronization Message is sent as an asynchronous serial message. Characters are sent at 9600 bps, with one start bit, one stop bit and no parity bit.

- Slave bc336VME's are programmed with "Digital Time" as their time source. The RX side of the Slave's RJ11 connector (J1A) is connected to the TX side of the Masters RJ11 with four-conductor telephone cable. When a valid time message is received from the Master, synchronization takes place. All features of the bc336VME are available with this mode including "flywheeling". The signal received on the Slave's Rx connector is looped through to the Tx connector, allowing signals to be connected in a "daisy-chain" fashion.

- Any of the bc336VME's depicted in Figure 2-1 and 2-2 can serve as a Master. The Master is the only unit in the network that transmits Time Synchronization Messages on the serial lines. If more than one bc336VME tries to operate as a Master, the time signals are corrupted. There is no hardware damage when this occurs, but network synchronization is not possible.

3.5.2 TIME SYNCHRONIZATION MESSAGE FORMAT

Time Synchronization message consist of two messages in the following format:

* <CR><LF>
Time YY+DDD HH:MM:SS.::::::<CR><LF>

" * "
An asterisk synchronization character, which is transmitted as soon as possible after the on-time mark. It is followed by the carriage return ($0D) and line feed ($0A) characters.

" Time"
A message indicating contents of the following data packet. Note that there is a space after the word Time. Case is significant.

" YY "
Two ASCII numeric characters representing year. Currently the bc336VME does not support year keeping and this field is reserved for future use.

" + "
A single character representing sign of the time. Characters "+" or space represent positive time and "-" represent negative time. The bc336VME works only in real-time mode and contents of this character will not effect operation of the board.

" DDD "
Three ASCII numeric characters representing days followed by the space. This can range from 000 to 999 (decimal).

" HH: "
Two ASCII numeric characters representing hours followed by a colon. This can range from 00 to 23 (decimal).

" MM: "
Two ASCII numeric characters representing minutes followed by a colon. This can range from 00 to 59 (decimal).

" SS. "
Two ASCII numeric characters representing seconds followed by a period. This can range from 00 to 59 (decimal).

" μμμμμμ "
Six ASCII hexadecimal characters representing the time tag of the beginning of the stop bit of the synchronization character (the asterisk). The unit of time is a half microsecond.

The message must be transmitted within the 1 second period following the on-time mark.

CHAPTER FOUR
INPUT/OUTPUT CONNECTORS

4.0 GENERAL

All I/O signals are available on the front panel connectors. Time code input is available on the front panel BNC and 15 pin DS connector. The location of all connectors is shown in Figure 1-1.

4.1 J1 SERIAL DIFFERENTIAL INPUT/OUTPUT

A serial differential I/O is available on the front panel dual phone jack connector labeled 'J1'. Either a 4 wire or 6 wire phone jack can be used with J1. The pinouts for J1 are listed in Table 4-1. Figure 4-1 shows the orientation of the J1 pinouts.

The Tx(-) signal will drive most RS-232 interfaces.

4.2 J2 TIME CODE INPUT

The time code input is available on the front panel BNC labeled 'J2' and is connected in parallel with the other connectors which carry this signal.

4.3 J3 SIGNAL I/O CONNECTOR

All I/O signals (except the serial output) are connected to the front panel 15 pin DS connector labeled 'J3' and are connected in parallel with the other connectors which carry these signals. The pinout assignments for the J3 connector are shown in Table 4-2.

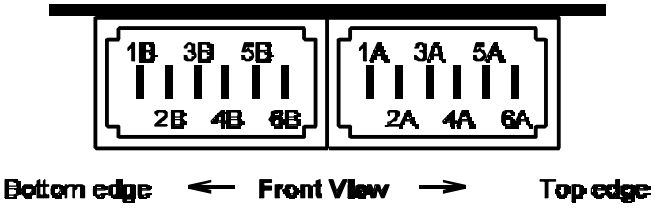
4.4 P2 CONNECTOR

The P2 connector carries signals on the row C. P2 pinouts are shown in the Table 4-3. The ground connections on P2 are on the pins A1-A17.

Table 4-1
J1 Serial I/O Pinouts

J1	SIGNAL DESCRIPTION
1A	NC
2A	Ground
3A	Rx(-)
4A	Rx(+)
5A	Ground
6A	NC
1B	NC
2B	Ground
3B	Tx(+)
4B	Tx(-)
5B	Ground
6B	NC

Figure 4:1



J1 Phone Jack Pinout Orientation

Table 4-2
J3 I/O Connector Pinouts

J3	SIGNAL DESCRIPTION
1	Time Code Input
2	Ground
3	Auxiliary Analog Input 1
4	Ground
5	Auxiliary Analog Input 2
6	Ground
7	External Event Input
8	1 Pulse Per Second Input
9	Heartbeat Pulse Output
10	Future Implementation
11	1 Pulse Per Second Output
12	Ground
13	-6U Option I/O
14	-6U Option I/O
15	Not Used

Table 4-3
P2 Connector Pinouts

P2	SIGNAL DESCRIPTION
C1	Time Code Input
C2	Time Code Return
C3	1 Pulse Per Second Input
C4	Tx(+)
C5	Tx(-)
0	External Event Input
C8	Future Implementation
C9	Heartbeat Pulse Output
C10	1 Pulse Per Second Output
C13	Auxiliary Analog Input 1
C14	Auxiliary Analog Input 2

This Page Intentionally Left Blank.

CHAPTER FIVE
THEORY OF OPERATION

5.0 GENERAL

This section describes the theory of operation for the bc336VME Time Code Processor Module. Reference Schematic Diagram 11760.

5.1 TIME CODE DECODER

The heart of the bc336VME utilizes Datum's reduced chip set decoder circuitry. Details on the operation of this circuitry are proprietary.

5.2 VMEbus INTERFACE

The VMEbus interface consists of the usual assortment of bus transceivers, buffers, and decoders commonly found on any microprocessor based system. Support for the interface is provided by the VME 2000 Slave Module Interface device from PLX Technology and the MC68153 Bus Interrupter Module from Motorola.

This Page Intentionally Left Blank

CHAPTER SIX
ADJUSTMENTS

6.0 PHASE LOCK LOOP ADJUSTMENT

The bc336VME has only a single adjustment which should never require alteration by the user. This section outlines a method for correctly setting the phase lock loop adjustment, variable resistor R10. The following steps assume that the bc336VME is correctly mounted in a rack and is exposed to allow access by an oscilloscope probe. A source of IRIG B timecode is required.

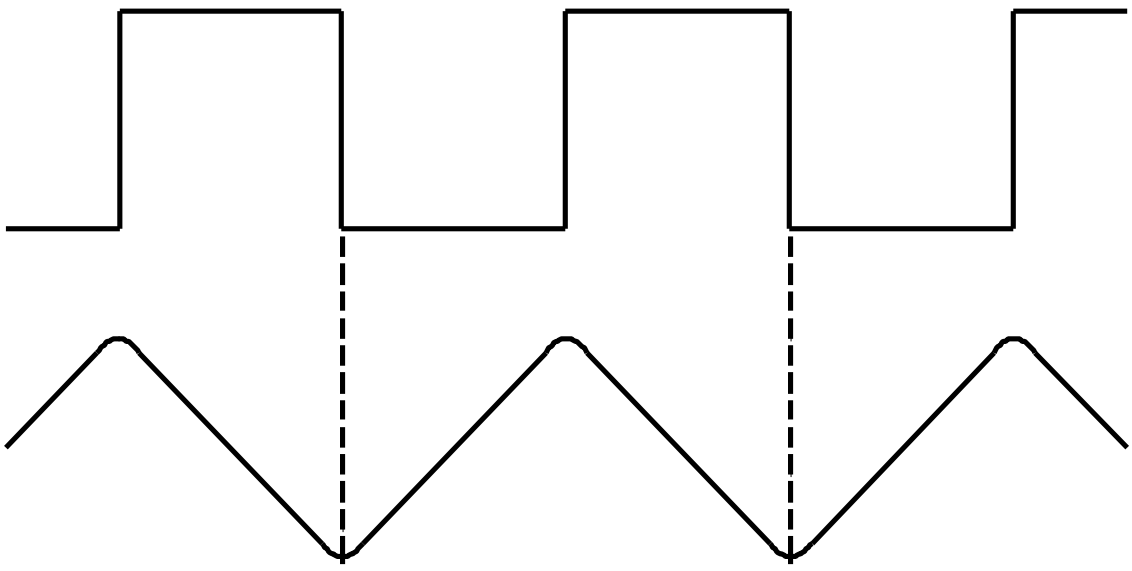
- Select an IRIG B timecode to be decoded by the bc336VME. This step forces the bc336VME to search for a 1000Hz carrier frequency only.

- Input the IRIG B timecode through the BNC connector (J2) or pin #1 of J3.

- Use pin #16 of U3 as the input to channel A of a dual channel oscilloscope. Use the reference carrier selected in Step 2 above as the input to channel B of the oscilloscope. Use channel A as the oscilloscope horizontal synch input.

- Adjust R10 so that the negative going transitions on pin #16 are centered on the negative "peaks" of the carrier. See figure 6-1 below.

Figure 6-1
Phase Lock Look Adjustment



This Page Intentionally Left Blank.

CHAPTER SEVEN
PROGRAMMING EXAMPLES

7.0 GENERAL

This section provides programming examples to illustrate the operation of the bc336VME. The examples are written in the C programming language. Hexadecimal constants are specified in C by using the prefix '0x'. The functions read_byte() and write_byte() are used to indicate a direct access to a physical memory location. In most programming environments, read_byte() and write_byte() can be defined as the following macros, assuming that type char is used for D08(O) data and that 'addr' can be successfully cast as a pointer to type char.

```
#define read_byte(addr)              (*(char *)(BASE + addr))
#define write_byte(addr,data)        *(char *)(BASE + addr) = data
```

A system dependent base address is defined below where 'YYYY' indicates a 64k byte page of memory used for A16 data and 'S' indicates the SW1 dip switch setting.

```
#define BASE 0xYYYYS000
```

Memory locations are referred to with the BASE constant plus an offset constant. The offset constants are referred to with a label that is defined in Table 3-1 and Table 3-2.

7.1 SETTING UP THE CONTROL REGISTERS

This example shows how to setup the bc336VME control registers. The bc336VME is setup to automatically detect an IRIG B time code input signal, disable Time Valid Interrupts, disable External Event Capture, and operate in the 'read time code' mode.

```
while(read_byte(WARMSTRT));          /* the bc336VME is ready to       */
                                     /* accept setup data when         */
                                     /* WARMSTRT = 0                   */
write_byte(TCSEL,0);                 /* setup control registers        */
write_byte(TVINTEN,0);
write_byte(EVENT,0);
write_byte(MODE,0);
write_byte(HBCTRL,0);
write_byte(PROPDEL0,0);
write_byte(PROPDEL1,0);
write_byte(TVFLAG,0);
write_byte(WARMSTRT,1);               /* initiate bc336VME warm reset */
```

7.2 SET MAJOR TIME

The major time will be set in REQTIME block when TCSEL is set to 0x03 or 0x52. To load a desired major time to REQTIME block refer to the following programming example:

```
#define     read_byte(addr)              (*(char *)(base+addr))
#define     write_byte(addr,data)        *(char *)(base+addr)=data

#define     time0       0x401            /* days hundreds                */
#define     time1       0x403            /* days tens                    */
#define     time2       0x405            /* days units                   */
#define     time3       0x407            /* hours tens                   */
#define     time4       0x409            /* hours units                  */
#define     time5       0x40b            /* minutes tens                 */
#define     time6       0x40d            /* minutes units                */
#define     time7       0x40f            /* seconds tens                 */
#define     time8       0x411            /* seconds units                */

#define     WARMSTART          0x07fd/* warmstart                      */

/* Set major time */

write_byte(TCSEL, 0x03);             /* select external 1pps           */
write_byte(time0,1);                     /* setime days hundreds =1        */
write_byte(time1,0);                     /* setime days tens =0            */
write_byte(time2,0);                     /* setime days units =0           */
write_byte(time3,2);                     /* setime hours tens =2           */
write_byte(time4,3);                     /* setime hours units =3          */
write_byte(time5,4);                     /* setime minutes tens =4         */
write_byte(time6,7);                     /* setime minutes units =7        */
write_byte(time7,5);                     /* setime seconds tens =5         */
write_byte(time8,9);                     /* setime seconds units =9        */
write_byte(TCSEL, 0x03);             /* select external 1pps           */

write_byte(WARMSTART,0x01);              /* warmstart command              */
while(read_byte(WARMSTART){};        /* wait until the command is done */
```

7.3 CAPTURING AND READING TIME

This example shows how to capture time across the VMEbus using the REQTIME register. The bc336VME is polled to determine when the time data is valid. A useful data structure is defined for the bc336VME time data block. The union 'btol' converts four (8 bit) bytes to one (32 bit) unsigned long integer. The structure 'time_buf' provides storage for the entire time data block and the frequency error bytes. The structure provides storage for the 9 major time (days - seconds) bytes in the major_time char array. The three subsecond bytes are written to the time.subsec char array. The two frequency error bytes are written to the time.freqerr char array. The subsecond count and frequency error can then be read out as two unsigned long integers.

```
union btol {
      char ba[4];
      unsigned long li;
};
struct time_buf {
      char major_time[9];
      union btol subsec;
      union btol freqerr;
} time = {0};

write_byte(TVFLAG,0);                              /* clear time valid flag */
write_byte(TIMEREQ,0);                             /* request time          */

/* wait for time valid (TVFLAG bit 0 = 1) */
while(!(read_byte(TVFLAG) & 0x01));
/* time is now valid and can be read */
time.major_time[0] = read_byte(BASE + 0x401L)      /* days hundreds         */
time.major_time[1] = read_byte(BASE + 0x403L)      /* days tens             */
                     .
                     .
time.major_time[8] = read_byte(BASE + 0x411L)      /* seconds units         */

/* NOTE: time.subsec.ba[0] = 0 */

time.subsec.ba[1] = read_byte(BASE + 0x413L)       /* subsec MSB            */
time.subsec.ba[2] = read_byte(BASE + 0x415L)       /* subsec                */
time.subsec.ba[3] = read_byte(BASE + 0x417L)       /* subsec LSB            */

/* NOTE: time.freqerr.ba[0] = time.freqerr.ba[1] = 0 */

time.freqerr.ba[2] = read_byte(BASE + 0x41bL)      /* frequency error MSB   */
time.freqerr.ba[3] = read_byte(BASE + 0x41dL)      /* frequency error LSB   */

/* convert subsecond count and frequency error to real number */
/* between 0.0000000 and 0.9999995 (SEE SECTION 3.3)          */
temp = 2000000L + (time.freqerr.li - 33920);
subseconds_real = (float)time.subsec.li / (float)temp;
```

7.4 EXTERNAL EVENT TRIGGER

The following example shows how to setup the bc336VME to receive External Event Triggers on the rising edge. The event is setup to trigger a VMEbus interrupt. Be sure to read the INT3ACK register and write a 0 to the TVFLAG register during the interrupt service routine so that another event can take place.

```
write_byte(EVENT,0x01);                    /* enable event on rising edge  */
write_byte(TVINTEN,0x02);                  /* enable int on event time     */
write_byte(WARMSTRT,1);                    /* initiate bc336VME warm reset */
write_byte(BIMV3,VECTOR);                  /* setup int vector             */
write_byte(TVFLAG,0);                      /* clear time valid register    */
dummy = read_byte(INT3ACK);                /* ack int source 3             */
write_byte(PINTCLR3,0);                    /* clear pending interrupt      */
write_byte(BIMCR3,0x12);             /* enable int on IRQ level 2 */

/* wait for interrupt                          */

/* part of interrupt service routine           */
```

```
/* read event time data block (offset 421 - 437) */
dummy = read_byte(INT3ACK);                 /* ack int source 3            */
write_byte(TVFLAG,0);                       /* clear time valid register   */

/* return from interrupt */
```

7.5 HEARTBEAT PULSE

The following example shows how to setup the bc336VME to generate an interrupt every heartbeat pulse. The heartbeat is setup for a rate of 1000 pulses per second. The time code format is assumed to be IRIG B. Be sure that the bc336VME is decoding time or flywheeling before starting the heartbeat pulses.

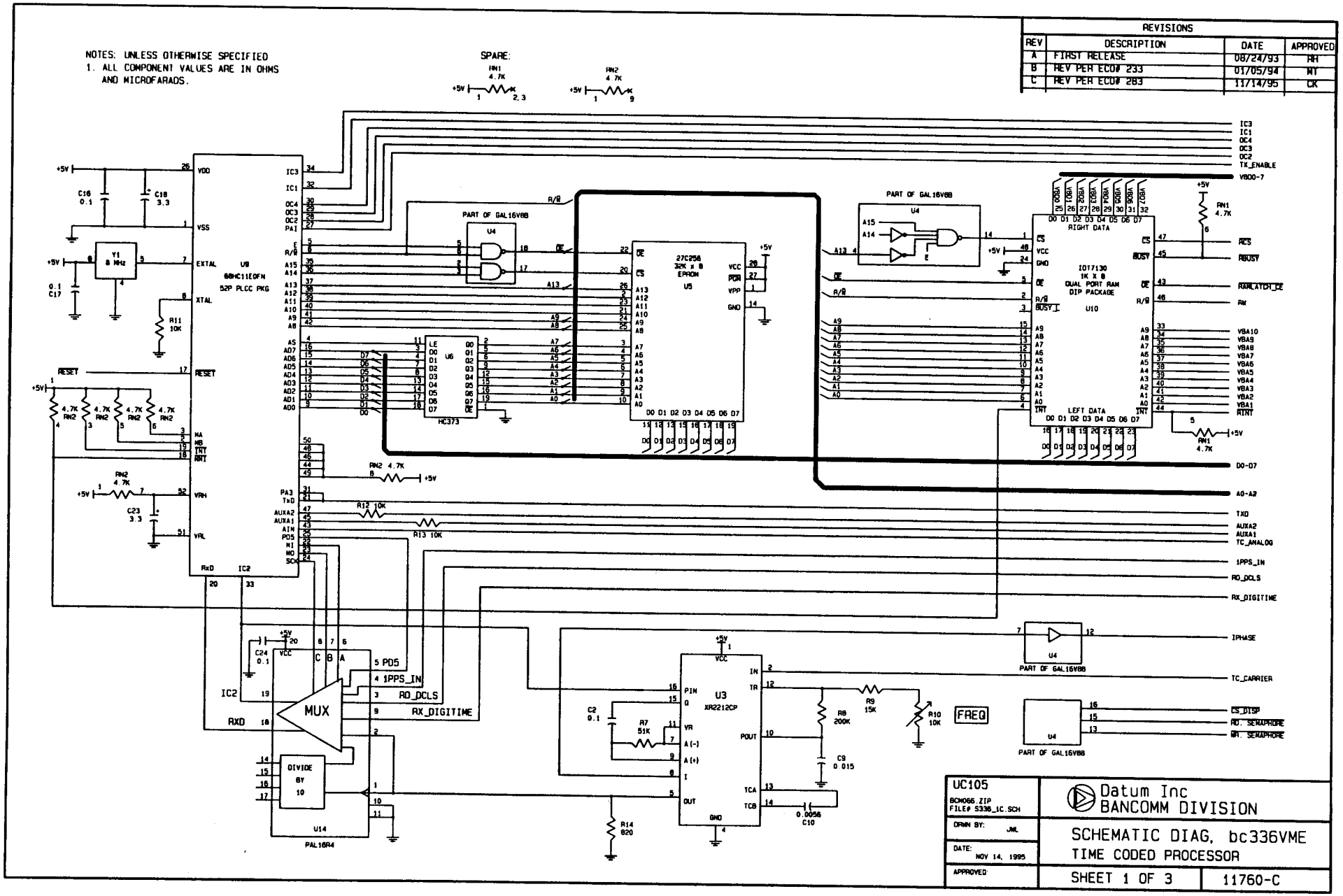
```
while(read_byte(STATUS) != 0x01);           /* wait for decoding           */
write_byte(HBRATE0,0x03);                   /* heartbeat rate (MSB)        */
write_byte(HBRATE1,0xE8);                   /* heartbeat rate (LSB)        */
write_byte(HBCTRL,1)                        /* enable heartbeat            */
write_byte(WARMSTRT,1);                     /* initiate bc336VME warm reset */
write_byte(BIMV2,VECTOR);                   /* setup int vector            */
write_byte(PINTCLR2,0);                     /* clear pending interrupt     */
write_byte(BIMCR2,0x15);              /* enable int on IRQ level 5    */

/* wait for interrupt generated every millisecond */
```

CHAPTER EIGHT
DRAWING SET

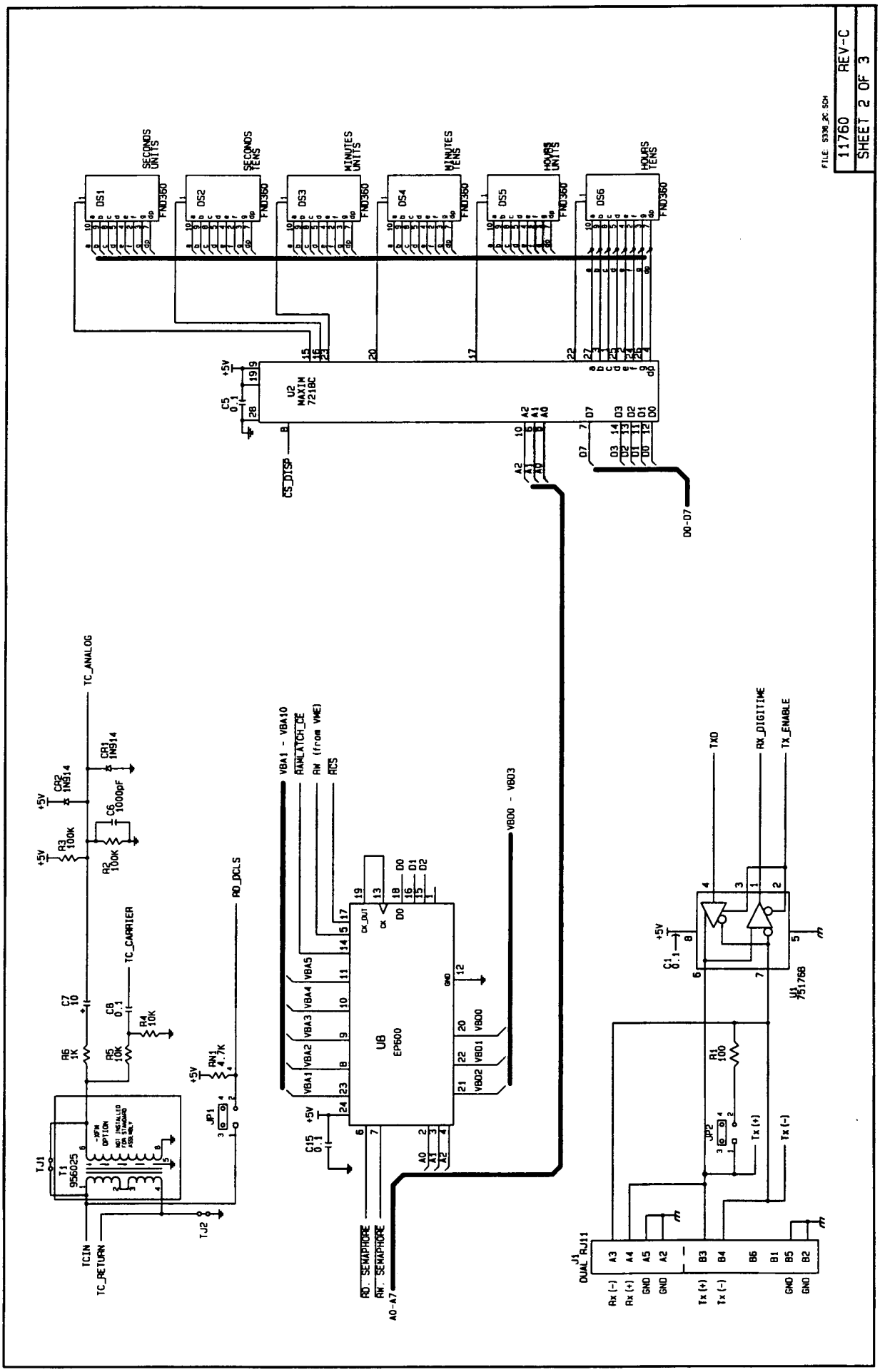
8.0 GENERAL

This chapter contains the schematic diagram, assembly drawing, and parts list for the bc336VME.

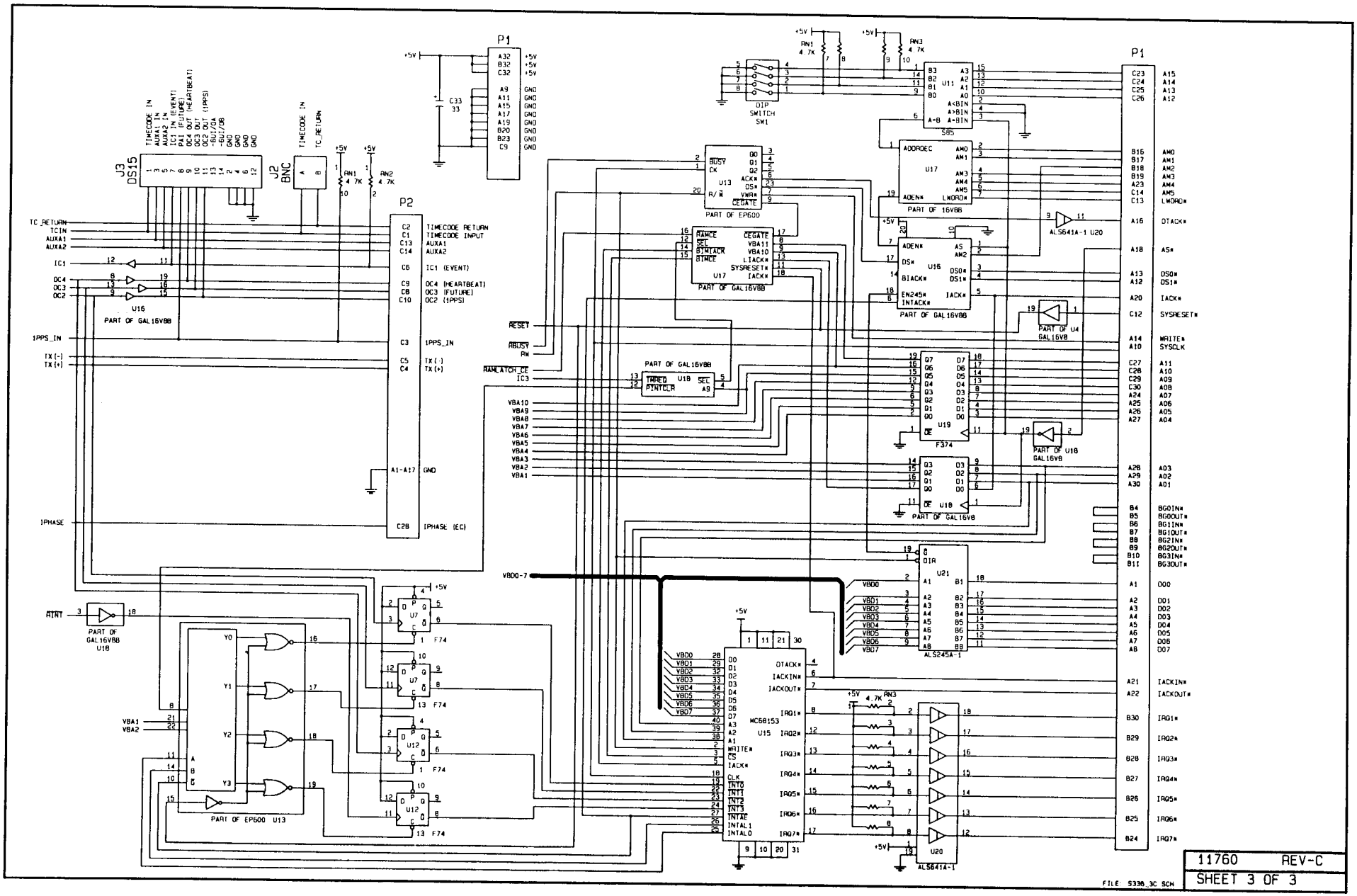
This Page Intentionally Left Blank



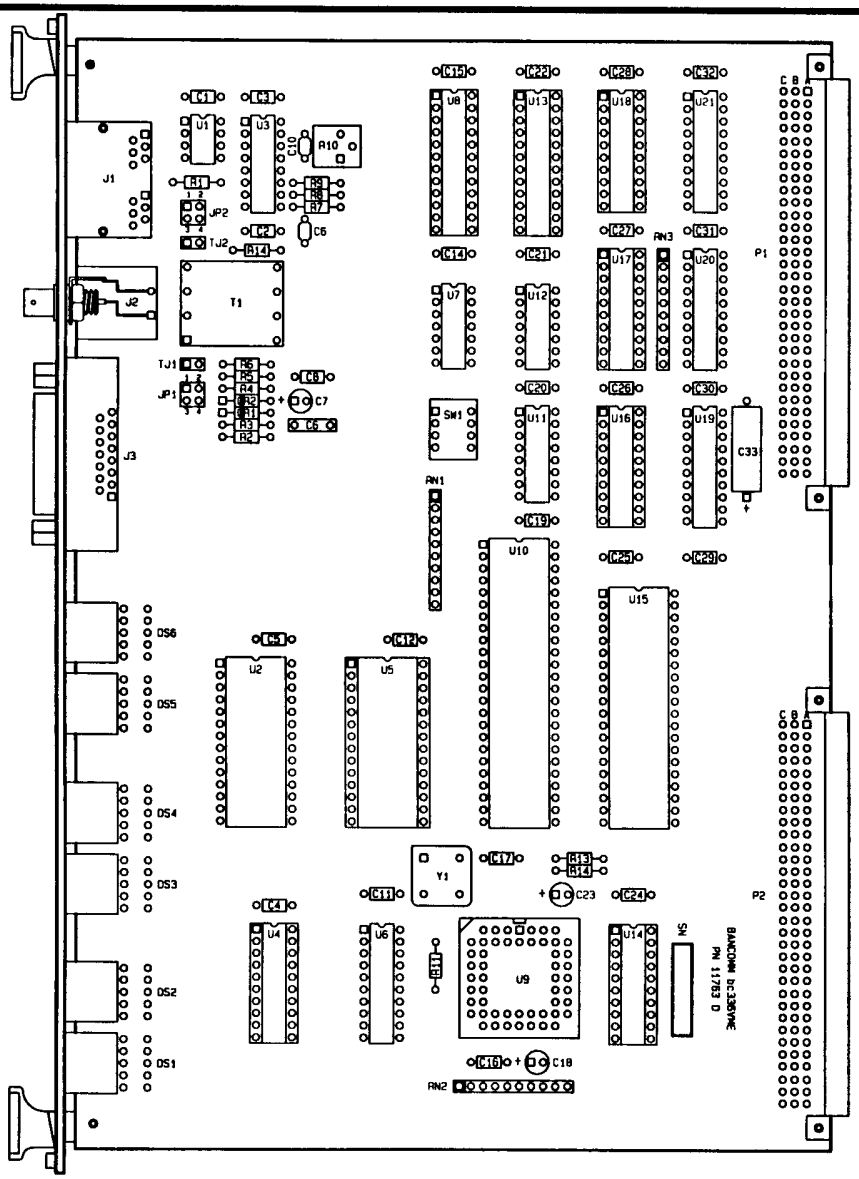
This Page Intentionally Left Blank



This Page Intentionally Left Blank



This Page Intentionally Left Blank



REVISIONS			
LTR	DESCRIPTION	DATE	APVD
A	FIRST RELEASE	08/24/93	R.H.
B	REV PER ECO# 233	01/05/94	M.T.
C	REV PER ECO# 276	09/09/95	C.K.
D	REV PER ECO# 283	11/14/95	C.K.

UC105 BCM066.ZIP FILE: A336_10.SCH	Datum Inc BANCOMM DIVISION	
DRAWN BY: JWL	ASSEMBLY, bc336VME	
DATE: NOV 14, 1995	TIME CODE PROCESSOR	
APP. BY:	SHEET: 1 OF 2	11763-D

This Page Intentionally Left Blank

Assembly, Parts Listing
bc336VME Time Code Processor

Ref: Drawing No. 11763 D
Ref: UC 105
Nov 14, 1995

OPT	BC P/N	MANF P/N	MANUFACTURE	VALUE	DESCRIPTION	QTY#	REF DESIG.
	1503335	335RMR035M	IC	3.3 MF,35V	ALUMINUM ELECTROLYTIC CAP.	2.00	C18,23
	1504106	196D106X9035PE4	SPRAGUE	10 MF,35V	TANTALUM CAP, RADIAL LEADS	1.00	C7
	1506102	SR211C102KAA	AVX	1000 PF,50V	MONO CERAMIC CAPACITOR .2 R/L	1.00	C6
	1506153	SR211C153KAA	AVX	15000 PF,100V	MONO CERAMIC CAPACITOR .2 R/L	1.00	C9
	1506562	SR211C562KAA	AVX	5600 PF,100V	MONO CERAMIC CAPACITOR .2 R/L	1.00	C10
	1508336	336TTA035B	IC	33 MF,35V	ALUM. ELEC. CAPACITOR AXIAL/L	1.00	C33
	1515104	MD015E104MAA	AVX/67349	0.1 MF,50V	DIP GUARD CAPACITOR	25.00	C1-5,8,11,12,14-17,19-22,24-32
	1701763	11762C	BANCOMM DIV, DATUM	bc336V TC PROC.	PRINTED CIRCUIT BOARD	1.00	PCB1
	2101003	31-221	AMPHENOL	50 OHM	FRONT MNT BNC BULKHEAD RECEP.	1.00	J2 Pnl Mnt
	2104001	913346	ERNI	96 POS	DIN CONNECTOR, MALE	2.00	P1,2
	2117061	TSW-130-07-G-D	SAMTEC	2X30 POS	STRAIGHT TERMINAL STRIP	1.00	JP1,2 = 2x2
	2124215	869521-1	AMP	15 POS	'D' SKT, .318 RTANG PCMNT B/L	1.00	J3
	2148010	10-2822-90C	ARIES	10 POS	RTANG PCB MOUNT, LED SOCKET	6.00	Ref DS1-6
	2149024	824-AG31D	AUGAT	24 POS	SLIM DIP SOCKET	2.00	Ref U8,13
	2150020	10620-01-445	ANDON/SPECIRA	20 POS	DIP SOCKET	5.00	Ref U4,14,16,17,18
	2150028	10628-01-445	ANDON/SPECIRA	28 POS	DIP SOCKET	1.00	Ref U5
	2152052	641748-2	AMP	52 POS	PLCC REC CHIP CARRIER	1.00	Ref U9
	2190005	SS-666602S-NF	STEWART	2 PT,6POS	SHIELDED PHONE JACK PCMNT	1.00	J1
	2306008	X209	DIGI-KEY	8MHz	HALF SIZE TTL/CMOS CLOCK OSC	1.00	Y1
	2401613	11765A	BANCOMM DIV, DATUM	bc336V TCP	FRONT PANEL	1.00	FT. PANEL
	2404600	VME-6U-1450	PHILLIPS COMPONENTS		VME EXTRACTOR HANDLES KIT	1.00	Pnl Mnt
	2802002	3341-1S	3M		JACK SCREW KIT	1.00	Pnl Mnt
	3703002	LTS360HR	LITEON	HI-BRIGHT	7 SEGMENT DISPLAY, 0.36 INCH	6.00	DS1-6 (SKT)
	4701101	RC07GF101J	ALLEN BRADLEY	100 OHM,1/4W	FIXED RESISTOR	1.00	R1
	4701102	RC07GF102J	ALLEN BRADLEY	1 K OHM,1/4W	FIXED RESISTOR	1.00	R6
	4701103	RC07GF103J	ALLEN BRADLEY	10 K OHM,1/4W	FIXED RESISTOR	5.00	R4,5,11-13
	4701104	RC07GF104J	ALLEN BRADLEY	100 K OHM,1/4W	FIXED RESISTOR	2.00	R2,3
	4701153	RC07GF153J	ALLEN BRADLEY	15 K OHM,1/4W	FIXED RESISTOR	1.00	R9
	4701204	RC07GF204J	ALLEN BRADLEY	200 K OHM,1/4W	FIXED RESISTOR	1.00	R8
	4701513	RC07GF513J	ALLEN BRADLEY	51 K OHM,1/4W	FIXED RESISTOR	1.00	R7
	4701821	RC07GF821J	ALLEN BRADLEY	820 OHM,1/4W	FIXED RESISTOR	1.00	R14
	4703103	72P103	BECKMAN	10 K OHM,1/2W	SINGLE TURN POTENTIOMETER	1.00	R10
	4705472	710A472	ALLEN BRADLEY	4.7 K OHM,1/8W	C-SIP RESISTORS, 10 PIN 'X'	3.00	RN1,2,3
	4803001	IN914			SILICON DIODE	2.00	CR1,2
	5108001	76SB04	GRAYHILL		4PST DIP SWITCH	1.00	SW1
	9006818	74F74	NATIONAL	14P DIP PKG	DUAL D FLIP FLOP	2.00	U7,12
	9006858	MM74F374N	NATIONAL	20P DIP PKG	OCTAL D FLIP FLOP	1.00	U19
	9008657	74HC373	VARIOUS	20P DIP PKG	OCTAL D TRANSPARENT LATCH, T/S	1.00	U6
	9015222	74S85	NATIONAL	16P DIP PKG	4-BIT MAGNITUDE COMPARATOR	1.00	U11
	9102003	MC68HC11E0FN	MOTOROLA	52P PLCC PKG	MICROCOMPUTER	1.00	U9 (SKT)
	9103031	MX68C153	MACRONIX	40P DIP PKG .6W	CMOS BUS INTERRUPT MODULE	1.00	U15
	9204020	ICM7218CIPI	MAXIM	28P DIP PKG .6W	8 DIGIT LED DISPLAY DRIVER	1.00	U2
	9207076	SN75176B	TI	08P DIP PKG	DIFF BUS TRANS	1.00	U1
	9207920	SN74ALS245A-1N	TI	20P DIP PKG	OCTAL BUS TRANSCEIVER	1.00	U21
	9207925	SN74ALS641A-1N	TI	20P DIP PKG	OCTAL BUS TRANSCEIVER	1.00	U20
	9307030	XR2212CP	EXAR	16P DIP PKG	PHASE LOCKED LOOP	1.00	U3
	9405001	EP600DC-3	ALTERA	24P DIP PKG .3W	EPLD	2.00	U8,13 (SKT)
	9405020	PAL16R4ACN	MMI	20P DIP PKG..3W	PAL	1.00	U14 (SKT)
	9405063	GAL16V8B-25LP	LATTICE	20P DIP PKG	GAL	4.00	U4,16,17,18 (SKT)
	9406040	27C256	VARIOUS	28P DIP PKG .6W	32 K BYTE, CMOS EPROM	1.00	U5 (SKT)
	9407665	IDT7130LA70P	IDT	48P DIP PKG .6W	1K X 8 DUAL PORT RAM	1.00	U10
XFM	5603002	956025	DATUM INC	08P DIP PKG	TRANSFORMER	1.00	T1