bc630AT
Real Time Clock Module

User's Guide

CHAPTER ONE
INTRODUCTION

1.0 GENERAL

The bc630AT Real Time Clock 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 bc630AT Real Time Clock include:

- Real Time Clock operation with microsecond precision.
- Decodes commonly used time code formats: IRIG B, IRIG A, NASA 36 and 2137.
- Master/Slave operation allows digital synchronization of multiple bc630AT's.
- Time base can be synchronized to external 1PPS or 1, 5, 10 MHz reference frequency.
- Continues to provide time, 'flywheels', during loss of the external time code source.
- Battery backed Real Time Clock maintains settable time during power loss and can be synchronized to an external time code.
- Battery backed control values allow one-time setup.
- Supports external event time capture input.

- Provides a rate programmable heartbeat pulse/interrupt (1-2000 Hz) synchronized to the time source.

- Provides programmable propagation delay compensation.

- It's 16 byte I/O block can be located on any one of 48 I/O blocks in the 100H - 3FFH AT I/O address space.

- Drives Datum's optional LED display modules with the decoded time.

1.2 PHYSICAL AND FUNCTIONAL OVERVIEW

The bc630AT is a half height IBM XT/AT bus board designed to provide a precision real time clock, decode serial time code signals, digitally synchronize multiple PC's and provide a number of other valuable timing features. The bc630AT also incorporates a battery backed real time clock IC which maintains settable time during power loss and can be synchronized to an external time code signal. The real time clock can also be decoded with microsecond precision.

The operation of the bc630AT is controlled by sixteen 8-bit registers written and read by the host XT/AT computer. These registers are used for defining the time source the bc630AT decodes, mode of operation, setting and reading the real time clock, activating time capture operations, holding the captured time and status; defining the heartbeat frequency, and defining the propagation delay compensation value.

1.3 PERFORMANCE SPECIFICATIONS

The principal performance characteristics are listed in Table 1-1, on the following page.

Table 1-1
Performance Specifications

Item		Description
Time Code Reader	Time Code Formats	IRIG A, B, NASA 36, 2137.
	Carrier Range	IRIG A - 10 kHz ± 2%. IRIG B, NASA 36, 2137 - 1 kHz ± 2%.
	Flywheel Accuracy	<3.6 milliseconds per hour (with Filter enabled).
	Modulation Ratio	3:1 to 6:1.
	Input Amplitude	100 mVrms to 1 Vrms.
	Input Impedance	10K Ω (AC Coupled).
Real Time Clock	Format	24 hour.
	Accuracy	± 10 PPM.
XT/AT Bus Interface	Address Space	1 block of 16 bytes in the PC I/O map. Range: 100H - 3FFH in an AT computer. 200H - 3FFH in an XT computer.
	Interrupts	Jumper selectable for IRQ3-7, 9, 10, 11, 12, 14 or 15 interrupt lines.
	Power	+5VDC @ 500mA.
TTL/CMOS Input Signals	Event Capture	TTL/CMOS, positive or negative edge triggered, 50ns min width, 500 µs min period.
TTL Output Signals	1 Pulse Per Second	TTL/CMOS, positive edge on time.
	Heartbeat Pulse	TTL/CMOS, positive edge on time.
RS-422 Input/Output Signals	1PPS	1PPS (+), 1PPS (-), positive edge on time.
	Serial Time/Mssg	TX/RX (+), TX/RX (-) 9600 Baud, 8 bits, 1 stop bit, no parity. TX (-) compatible with most RS-232 devices.
Operating and Storage Environments	Temperature Operating Non-Operating	 0^{o} to 70^{o} C. -50^{o} to 125^{o} C.
	Relative Humidity Operating Non-Operating	 10% to 80% (non-condensing). 5% to 95% (non-condensing).
	Altitude Operating Non-Operating	 1,000 feet below sea level to 10,000 feet above sea level. 1,000 feet below sea level to 20,000 feet above sea level.

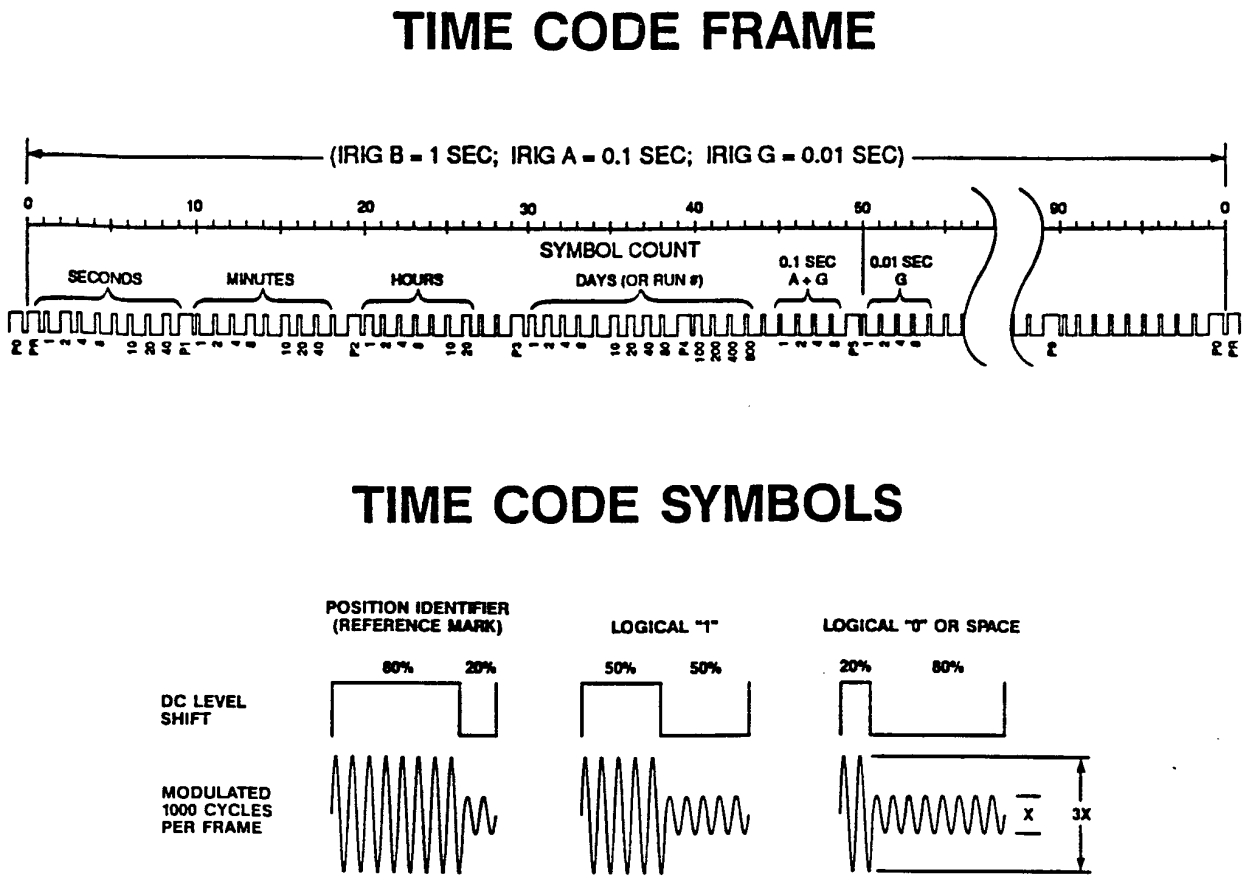
1.4 TIME CODE FORMATS

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 InterRange Instrumentation Group (IRIG) promoted a series of 'standard' time code formats now loosely referred to as 'IRIG Time Codes'. The bc630AT decodes two of these formats: IRIG A and 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 1-1 illustrates a frame of IRIG A, B or G time code.

Figure 1-1
IRIG Time Code Frame



1.5 IBM XT/AT COMPATIBILITY

AT type computers have an extended memory addressing capability over XT type computers. The bc630AT Real Time Clock is a port mapped device and does not make use of this extended memory capability but does give the user the option of using the additional interrupts found on the AT bus. Therefore, the bc630AT can be utilized in an XT computer with complete functionality and performance with the exception of a lower number of interrupt selections.

Physically, the bc630AT mounts in either an XT or AT type computer. In the case of the XT, the AT's second connector is not accessed resulting in a lower number of interrupt selections available to the PC.

CHAPTER TWO

INSTALLATION AND SETUP

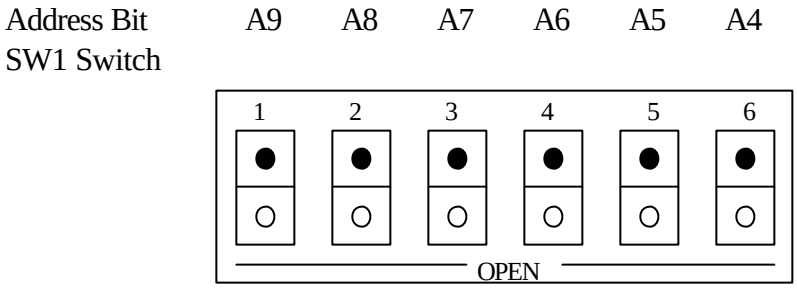
2.0 GENERAL

The bc630AT is a half height IBM XT/AT bus board. The bc630AT registers are 8-bits wide. This section details the steps required to setup the module for operation.

2.1 BASE ADDRESS SELECTION

Before installing the module in the computer, the address select DIP switch (SW1) must be set. The bc630AT occupies 16 bytes in the PC I/O address space and can be freely located on any 16 byte boundary. The 6 DIP switch positions of SW1 correspond to address bits A9 - A4 as shown in Figure 2-1, and determine the base address for the module (switch positions 7 and 8 are not used). The base address is defined as the address selected by the SW1 DIP switch when A3 - A0 are 0.

Figure 2-1
DIP Switch SW1



To select a base address, set each of the 6 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.

In an AT computer the base address can be set in the range of 100H - 3F0H. In an XT computer the base address can be set in the range of 200H - 3F0H.

Note: Setting a base address that is out of these ranges will probably prevent your computer from booting.

For example, to set the bc630AT base address to 300H, set switches 1 and 2 to the OFF position and set switches 3, 4, 5, and 6 to the ON position. Refer to Table 2-1 for a list of all the available base addresses in the PC bus and the corresponding switch settings.

Table 2-1
PC Bus Base Address Selection (SW1)

Base Addr	S1 A9	S2 A8	S3 A7	S4 A6	S5 A5	S6 A4	A3	A2	A1	A0
100H	ON	OFF	ON	ON	ON	ON	0	0	0	0
110H	ON	OFF	ON	ON	ON	OFF	0	0	0	0
120H	ON	OFF	ON	ON	OFF	ON	0	0	0	0
130H	ON	OFF	ON	ON	OFF	OFF	0	0	0	0
140H	ON	OFF	ON	OFF	ON	ON	0	0	0	0
150H	ON	OFF	ON	OFF	ON	OFF	0	0	0	0
160H	ON	OFF	ON	OFF	OFF	ON	0	0	0	0
170H	ON	OFF	ON	OFF	OFF	OFF	0	0	0	0
180H	ON	OFF	OFF	ON	ON	ON	0	0	0	0
190H	ON	OFF	OFF	ON	ON	OFF	0	0	0	0
1A0H	ON	OFF	OFF	ON	OFF	ON	0	0	0	0
1B0H	ON	OFF	OFF	ON	OFF	OFF	0	0	0	0
1C0H	ON	OFF	OFF	OFF	ON	ON	0	0	0	0
1D0H	ON	OFF	OFF	OFF	ON	OFF	0	0	0	0
1E0H	ON	OFF	OFF	OFF	OFF	ON	0	0	0	0
1F0H	ON	OFF	OFF	OFF	OFF	OFF	0	0	0	0

Table 2-2
PC Bus Base Address Selection (SW1)

Base Addr	S1 A9	S2 A8	S3 A7	S4 A6	S5 A5	S6 A4	A3	A2	A1	A0
200H	OFF	ON	ON	ON	ON	ON	0	0	0	0
210H	OFF	ON	ON	ON	ON	OFF	0	0	0	0
220H	OFF	ON	ON	ON	OFF	ON	0	0	0	0
230H	OFF	ON	ON	ON	OFF	OFF	0	0	0	0
240H	OFF	ON	ON	OFF	ON	ON	0	0	0	0
250H	OFF	ON	ON	OFF	ON	OFF	0	0	0	0
260H	OFF	ON	ON	OFF	OFF	ON	0	0	0	0
270H	OFF	ON	ON	OFF	OFF	OFF	0	0	0	0
280H	OFF	ON	OFF	ON	ON	ON	0	0	0	0
290H	OFF	ON	OFF	ON	ON	OFF	0	0	0	0
2A0H	OFF	ON	OFF	ON	OFF	ON	0	0	0	0
2B0H	OFF	ON	OFF	ON	OFF	OFF	0	0	0	0
2C0H	OFF	ON	OFF	OFF	ON	ON	0	0	0	0
2D0H	OFF	ON	OFF	OFF	ON	OFF	0	0	0	0
2E0H	OFF	ON	OFF	OFF	OFF	ON	0	0	0	0
2F0H	OFF	ON	OFF	OFF	OFF	OFF	0	0	0	0

Table 2-3
PC Bus Base Address Selection (SW1)

Base Addr	S1 A9	S2 A8	S3 A7	S4 A6	S5 A5	S6 A4	A3	A2	A1	A0
300H	OFF	OFF	ON	ON	ON	ON	0	0	0	0
310H	OFF	OFF	ON	ON	ON	OFF	0	0	0	0
320H	OFF	OFF	ON	ON	OFF	ON	0	0	0	0
330H	OFF	OFF	ON	ON	OFF	OFF	0	0	0	0
340H	OFF	OFF	ON	OFF	ON	ON	0	0	0	0
350H	OFF	OFF	ON	OFF	ON	OFF	0	0	0	0
360H	OFF	OFF	ON	OFF	OFF	ON	0	0	0	0
370H	OFF	OFF	ON	OFF	OFF	OFF	0	0	0	0
380H	OFF	OFF	OFF	ON	ON	ON	0	0	0	0
390H	OFF	OFF	OFF	ON	ON	OFF	0	0	0	0
3A0H	OFF	OFF	OFF	ON	OFF	ON	0	0	0	0
3B0H	OFF	OFF	OFF	ON	OFF	OFF	0	0	0	0
3C0H	OFF	OFF	OFF	OFF	ON	ON	0	0	0	0
3D0H	OFF	OFF	OFF	OFF	ON	OFF	0	0	0	0
3E0H	OFF	OFF	OFF	OFF	OFF	ON	0	0	0	0
3F0H	OFF	OFF	OFF	OFF	OFF	OFF	0	0	0	0

2.2 INSTALLATION PROCEDURE

To install the bc630AT module in your computer chassis:

1. Remove the computer chassis cover.

2. Select a vacant expansion slot and remove the blank rear panel bracket. Save the screw.

3. Slide the bc630AT Real Time Clock Module straight down to engage the motherboard connectors.

4. Fasten the top of the bracket to the chassis using the screw that was saved from Step 2.

5. Replace the chassis cover.

2.3 DEMONSTRATION DISKETTE

This disk contains two executable programs that demonstrate the use of the bc630AT Real Time Clock Module.

630DEMO1.EXE
Reads IRIG B, IRIG A, NASA 36, 2137 modulated and DC level shift time codes.

630DEMO2.EXE
General Purpose Demo Program for all time sources, time code types and features of the bc630AT.

630CODE.SYS
Device drivers that, when installed, replace the usual clock driver of 630RTC.SYS the PC with the bc630AT.

Instructions on the use of these programs is contained in the diskette file - README.DOC. At the DOS prompt simply type 'README'.

CHAPTER THREE

OPERATION AND SOFTWARE INTERFACE

3.0 GENERAL

The bc630AT occupies one block of 16 bytes in the XT/AT I/O address space. Refer to Section 2.1 for details on base address selection. Data transfers between the XT/AT bus can be accomplished with 8 bit I/O instructions. This chapter describes the bc630AT registers and their use.

3.1 REGISTERS

This section describes the registers used on the bc630AT for controlling its operation and transferring time data. Section 3.2 details the use of these registers. The Control Register memory map for the bc630AT is listed in Table 3-1. The memory map for the rest of the bc630AT 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 power on or software reset is shown. A value of '--' indicates that the register contents are undefined. A label for each register is listed as is a brief description of the register's function.

3.1.1 CONTROL REGISTERS

The Control Registers govern the operation of the bc630AT. To control the operation of the bc630AT, first write the appropriate values to the control registers, then execute an 'Initialize' command. The control registers are described below. All values are in base hexadecimal.

Table 3-1
bc630AT Control Register Memory Map

Offset (HEX)	Default Value	Label	Description
07H	00	STATUS	Status Register.
08H	* 00	TMSEL	Time Code Select.
09H	* 00	MASKS	Masks Register.
0AH	* D0H	HBRATE0	Heartbeat Rate LSB.
0BH	* 07H	HBRATE1	Heartbeat Rate MSB.
0CH	* 00	PROPDEL0	Propagation Delay LSB.
0DH	* 00	PROPDEL1	Propagation Delay MSB.

* Denotes battery backed value loaded at power on.

3.1.2 STATUS REGISTER (STATUS - offset 07H)

The bc630AT provides a read/write status byte containing the status of the Real Time Clock Module. The status byte is the only control register that is written to by the bc630AT.

1PPS	EVENT	TSTAT1	TSTAT0	TC3	TC2	TC1	TC0

STATUS - Offset 07H

TC3, TC2, TC1, and TC0 each designate the Last Known Time Code Detected.

Table 3-1
Status - Offset 07H

TC3	TC2	TC1	TC0	Description
0	0	0	0	Initialization or reset is complete, time code not yet found.
0	0	0	1	IRIG B - Modulated Time Code.
0	0	1	0	IRIG A - Modulated Time Code.
0	0	1	1	2137 - Modulated Time Code.
0	1	0	0	Battery Backed Real Time Clock IC.
0	1	0	1	bc630AT Master.
0	1	1	0	NASA 36 - Modulated Time Code.
0	1	1	1	IRIG B - DC Level Shift Time Code.
1	0	0	0	IRIG A - DC Level Shift Time Code.
1	0	0	1	NASA 36 - DC Level Shift Time Code.
1	0	1	0	External 1PPS.
1	0	1	1	External 1, 5, 10 MHz Frequency.

TSTAT1, TSTAT0
Time Code Tracking Status.

00 = time code present.
01 = flywheeling to the internal crystal.
10 = flywheeling to an external 1PPS.
11 = flywheeling to an external 1, 5, 10 MHz frequency reference.

Note: The term 'flywheeling' means that the input time source has been lost or is unusable, but the bc630AT is still providing time, heartbeats, etc. as if the time source is still present.

EVENT
Event Capture Flag.

0 = No event.
1 = Event captured.

1PPS
1PPS Flag.

0 = On-time mark has not occurred.
1 = On-time mark has occurred.

Note: This flag is set each second approximately 150 :secs after the on-time mark occurs. The bc630AT clears this flag only at Power On and with a software 'Reset' or 'Initialize' command. If utilized the user is responsible for clearing the flag.

3.1.3 TIME CODE SELECT REGISTER (TCSEL - offset 08H)

--	MOD/DC	IM1	IM0	TC3	TC2	TC1	TC0

Time Code Select Byte - offset 07H
TC3, TC2, TC1, and TC0 designate the time code to be decoded.

0000 = Auto detect IRIG B, IRIG A, NASA 36 and 2137 modulated & DC level shift time codes.
0001 = IRIG A.
0010 = IRIG B.
0011 = 2137.
0100 = Battery Backed Real Time Clock IC.
0101 = bc630AT Master.
0110 = NASA 36.
1010 = External 1PPS Synchronization.
1011 = External 1, 5, 10 MHz Synchronization.

Note: When auto detect is selected, the MOD/DC bit must be 0.

IM1, IM0
Designates the Initialization Mode upon power on or after an 'Initialize' command is issued.

00 = Normal initialization.
01 = Battery Backed RTC IC time and RTC time code period.
10 = Battery Backed RTC IC time and battery backed time source period.
11 = Battery Backed RTC IC time and user defined time source period specified in DAT00 (LSB) and DAT01 (MSB).

Initialization modes 1, 2, and 3 synchronize the bc630AT with the battery backed RTC IC time and the respective time code period of the mode during initialization. When initialization completes, the bc630AT searches for the designated time code and if a valid time code is not found, enters a 'flywheel' state maintaining time with the mode specified time code period.

The bc630AT continues flywheeling until a valid time code specified by TC3-TC0 is found at which time the bc630AT's clock is re-synchronized to the time code.

Note: Initialization takes approximately 3 seconds when Initialization Modes 1, 2 and 3 are selected.

MOD/DC
Designates the time code type specified by TC3 - TC0 as either Modulated or DC Level Shift.

0 = DC Level Shift Time Code.
1 = Modulated Time Code.

Note: In auto detect mode the MOD/DC bit must be 0. When non-time codes are selected (i.e. battery backed RTC IC, Master bc630AT, etc.) this bit is not used.

3.1.4 MASKS REGISTER (MASKS - offset 09H)

The Masks Register controls the disabling (masking) and enabling (unmasking) of the event capture, digital filtering of incoming time signals, heartbeat pulses, external flywheel synchronization and selects the interrupt source. If a change to the default conditions is desired a new value is written to the Masks Register and followed by an 'Initialize' or 'Process Masks Register' command.

ISRC2	ISRC1	ISRC0	HBEN	FILTER	XFW	EVN1	EVN0

MASKS - offset 09H

EVN1, EVN0
External Event Capture Control

00 = Disable External Event Capture.
01 = Enable External Event Capture on the falling edge of the external Event Capture Input.
10 = Enable External Event Capture on the rising edge of the External Event Capture Input.
11 = Enable External Event Capture on both the rising and falling edges of the External Event Capture Input.

XFW
Enable/Disable External Flywheel Synchronization

0 = Disable External Flywheel Synchronization.
1 = Enable External Flywheel Synchronization.

Enables disciplining of the bc630AT's time base during 'flywheel' periods with an external 1PPS or 1, 5, 10 MHz frequency reference if present.

FILTER
Enables Digital Filtering of the time source signals.

0 = Disable filter
1 = Enable filter

Enabling the Digital Filter invokes an algorithm which allows the bc630AT to decode all available time sources with maximum accuracy and increases the accuracy of the 'flywheel' during drop outs of the time source. The filter should be disabled if decoding time from an unstable time code or tape source (i.e. time frames fluctuate more than " 15 :secs).
HBEN
Heartbeat Pulses/Interrupts Control.

0 = Disable heartbeat pulses/interrupts (pulse output remains high)
1 = Enable heartbeat pulses/interrupts at the rate defined by HBRATE1 (MSB) and HBRATE0 (LSB).

Note: The heartbeat feature is not available for IRIG A.

ISRC2, ISRC1 and ISRC0
Interrupt Source Selection.

	Interrupt Register Value	Interrupt Source
000	- -	No Interrupts Selected.
001	Ready Byte Value	Ready Byte.
010	40H	Event.
011	60H	Heartbeat.
100	80H	1PPS.

The bc630AT generates an interrupt to the host PC by writing the Int Reg Val to the Interrupt Register (offset 0FH) upon the occurrence of the selected interrupt source. The interrupt is asserted until the host PC reads the Interrupt Register. The PC interrupt IRQ channel that the bc630AT asserts is controlled by jumper JP-5 (See Table 4-9). The host can optionally use the Interrupt Register in a polled mode by clearing the register and polling for the proper Interrupt Register Value.

Note: Only one Interrupt source can be selected. Maximum polling rate of Interrupt Register not to exceed 500 kHz.

3.1.5 HEARTBEAT RATE (HBRATE1, HBRATE0 - offset 0AH and 0BH)

The rate at which the heartbeat pulses/interrupts are generated is controlled by these registers which combine to produce a 16-bit unsigned value. The heartbeat pulse rate represents the number of pulses/interrupts generated each second. The frequency range of the heartbeat rate is 1 to 2000 Hz. The rising edge of the heartbeats pulses are synchronized to on-time. For example, a heartbeat rate of 1000 would result in a generated pulse/interrupt each millisecond whose rising edge would correspond to exactly 1 millisecond. (i.e. 0.000 000 , 0.001 000, 0.002 000 ... 0.999 000, 1.000 000 seconds.)

The bc630AT processes the heartbeat rate data when it receives an 'Initialize' or 'Synchronize Heartbeats' command. (See Section 3.3.)

Note: The bc630AT must be decoding time or flywheeling before the Heartbeat Rate is programmed.

3.1.6 PROPAGATION DELAY (PROPDEL1, PROPDEL0 - offset 0CH and 0DH)

The propagation delay compensation function can offset the effects of long cables between the time code source and the bc630AT. These two registers combine 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 :sec. Positive values advance the 1PPS (Pulse Per Second) and heartbeat pulses relative to the time code. Negative values retard the 1PPS and heartbeat pulses.

3.2 DATA TRANSFER ADDRESS SPACE

The data transfer address space consists of a sequence of 7 data bytes starting at the base address and one command byte at offset 0EH. The command byte also serves as a 'data ready flag byte' to feedback ready flag values which indicate the bc630AT has successfully completed a command task.

The PC XT/AT computer requests specific data from the bc630AT by writing a coded request to the Command Register at offset 0EH. The baseboard must then wait for the command byte to be modified by the bc630AT firmware to indicate that the requested data is available at offset locations 00H through 06H. The data format is predefined in this section.

Note: Polling rates of the Command Register must not exceed 500 kHz.

Table 3-2
bc630AT Data Transfer, Command and Interrupt Register Memory Map

Offset (HEX)	Reset Value	Label	Description
00H	- -	DAT00	Data Byte 0.
01H	- -	DAT01	Data Byte 1.
02H	- -	DAT02	Data Byte 2.
03H	- -	DAT03	Data Byte 3.
04H	- -	DAT04	Data Byte 4.
05H	- -	DAT05	Data Byte 5.
06H	- -	DAT06	Data Byte 6.
0EH	10H/1EH	COMMAND	Command Register.
0FH	- -	INTERRUPT	Interrupt Register.

3.3 SOFTWARE COMMANDS

The following command bytes have been defined for version OS 3.0. Additional command bytes may be added in future versions of the bc630AT firmware.

com byte	ready flag	data offset / bits 7-4 / bits 3-0

<Request Time> Current time is logged as the time when 80H was written to offset 0EH.

80H	00H	0 / seconds tens / seconds units
		1 / minutes tens / minutes units
		2 / hours tens / hours units
		3 / days tens / days units
		4 / DH1 DH0 -- C20 / C19 C18 C17 C16
		5 / C07 C06 C05 C04 / C03 C02 C01 C00
		6 / C15 C14 C13 C12 / C11 C10 C09 C08

C20-C00 = binary 0.5 microsecond count
DH1-DH0 = days hundreds value (0, 1, 2, 3)
-- = bit undefined

<Request Event Time> Last event time logged on programmed edge, positive or negative or both.

81H	01H	0 / seconds tens / seconds units
		1 / minutes tens / minutes units
		2 / hours tens / hours units
		3 / days tens / days units
		4 / DH1 DH0 -- C20 / C19 C18 C17 C16
		5 / C07 C06 C05 C04 / C03 C02 C01 C00
		6 / C15 C14 C13 C12 / C11 C10 C09 C08

C20-C00 = binary 0.5 microsecond count
DH1-DH0 = days hundreds value (0, 1, 2, 3)
-- = bit undefined

<Clear Event Capture> Event capture is disabled after each event until the command is executed.

82H	02H	0 / Time code Period mod 65536 LSB
		1 / Time code Period mod 65536 MSB
		2 / AUXA1 data byte
		3 / AUXA2 data byte

The time code period and AUX analog data are returned to facilitate data logging applications which use the event strobe.

com byte	ready flag	data offset / bits 7-4 / bits 3-0
<Output auxiliary data>		
83H	03H	0 / Time code Period mod 65536 LSB
		1 / Time code Period mod 65536 MSB
		2 / AUXA1 data byte
		3 / AUXA2 data byte
		4 / AGC level
		5 / reserved
		6 / reserved

<Set RTC Chip IC Time> Set the time of the battery backed Real Time Clock IC.

84H	04H	0 / seconds tens / seconds units
		1 / minutes tens / minutes units
		2 / hours tens / hours units
		3 / days tens / days units
		4 / months tens / months units
		5 / year tens / year units
		6 / 0000 / day of week
		(Sunday = 0 ... Saturday = 6)

<Request RTC IC Time> Current time of day obtained from the battery backed Real Time Clock IC as the time when 85H was written to offset 0EH.

85H	05H	0 / seconds tens / seconds units
		1 / minutes tens / minutes units
		2 / hours tens / hours units
		3 / days tens / days units
		4 / months tens / months units
		5 / year tens / year units
		6 / 0000 / day of week
		(Sunday = 0 ... Saturday = 6)

<Request Power Off Time> Time when computer was last turned off.

86 06H 0 / seconds tens / seconds units
1 / minutes tens / minutes units
2 / hours tens / hours units
3 / days tens / days units
4 / DH1 DHO -- -- / -- -- -- --
5 / -- -- -- -- / -- -- -- --
6 / -- -- -- -- / -- -- -- --
(-- denotes bit undefined)

com byte	ready flag	data offset / bits 7-4 / bits 3-0

<Set major time> Allows the user to set major time on the board without effecting the battery backed RTC.

87H 07H 0 / seconds tens / seconds units
1 / minutes tens / minutes units
2 / hours tens / hours units
3 / days tens / days units
4 / days hundreds / -- -- -- --
5 / -- -- -- -- / -- -- -- --
6 / -- -- -- -- / -- -- -- --
(-- denotes bit undefined)

Note: This command is primarily intended for use when the bc630AT is programmed to synchronize to an external 1PPS or frequency reference. It can also load time in other modes (e.g. time code mode) when the bc630AT is flywheeling. The time loaded will not be used until the one second epoch following the load. The bc630AT increments the time before loading it to the output buffer registers. The time is incremented at approximately 850 milliseconds into the current frame. The buffer registers are transferred to a set of holding registers synchronously with the 1PPS output. The time loaded by 87H command should be input well in advance of the 850 millisecond point in the frame and should reference the current frame.

<Initialization command>

90H 10H no associated data

<Reset to Default Values>

91H 10H no associated data

<Reset to Battery Backed Values>

92H 10H no associated data (Successful reset to battery backed values)
1EH no associated data (Checksum failure of battery backed values - Default control values loaded)

<Process Masks Register Only>

93H 13H no associated data

com byte	ready flag	data offset / bits 7-4 / bits 3-0

<Synchronize RTC IC> Synchronizes the battery backed RTC to the external time source being decoded.

94H 14H no associated data

<Synchronize Heartbeat Pulses/Interrupts> Loads and synchronizes HBRATE1, HBRATE0 to the current time source.

95H 15H no associated data

3.4 FUNCTIONAL DESCRIPTION

Section 3.1 and 3.2 provided an overview of the bc630AT control, command, and data registers. This section provides a description of how these registers are used to achieve the desired functions.

3.4.1 CONTROLLING THE OPERATING MODE OF THE bc630AT

To control the operating mode of the bc630AT, first write the appropriate data to the Control Registers. Then, to cause the bc630AT to take action on the Control Register data, issue the 'Initialize' command by writing 90H to the Command Register (offset 0EH). This action will cause the bc630AT to initialize and act on the data in the Control Registers. When the bc630AT is initialized and ready to accept another command, it will return the ready flag value of 10H to the Command Register (offset 0EH).

Note: Polling rates of the Command Register must not exceed 500 kHz.

In Version 3.0 firmware commands 94H ('Process Masks Register') and 95H ('Synchronize Heartbeats') are available which allow the Masks Register and Heartbeat Register data to be processed without reinitializing the bc630AT.

3.4.2 REQUESTING AND READING TIME

To request time from the bc630AT over the XT/AT bus, issue the 'Time Request' command by writing 80H to the Command Register (offset 0EH). This action causes the bc630AT to freeze the time and transfer it to the Data Transfer Address Space (offset 00H - 06H). Following the transfer of time, the bc630AT will write the ready flag value of 00H to the Command Register indicating the requested time is ready to be read by the XT/AT bus and the bc630AT is ready to receive another command.

The bc630AT requires approximately 150 :secs to load the Data Transfer Address Space with the time in response to a Time Request. The time, however, is frozen at the instant the 'Time Request' is issued.

3.4.3 EXTERNAL EVENT CAPTURE

The bc630AT has the ability to capture time in response to an external event. Use bits 0 and 1 in the Masks Register to enable the external event function and to select the active edge for capture. When the bc630AT detects the appropriate edge of the event signal it will:

- Freeze time and load it into an internal buffer.
- Set the Event bit in the Status register.
- Generate an interrupt to the host PC (if selected as the interrupt source in the Masks register) by writing the value 40H to the Interrupt register. (See section 3.1.4.)

The Event Time is transferred to the Data Transfer Address Space by issuing the 'Request Event Time' command 81H. Following the Event Time transfer, the bc630AT will write the ready flag value of 01H to the Command Register indicating the requested time is ready to be read by the XT/AT bus, and the bc630AT is ready to receive another command. At this point the event capture circuitry is disabled. To capture another edge with the parameters defined in the Masks register issue a 'Clear Event Capture' command. This reactivates the event capture circuitry.

The bc630AT requires approximately 150 :secs to load the Event Time to the Data Transfer Address Space in response to 'Request Event Time' command. The time, however, is frozen at the instant that the External Event Input is activated.

3.4.4 RATE PROGRAMMABLE HEARTBEAT PULSES

It is often useful to generate a periodic pulse/interrupt (heartbeat) which is synchronized to the time source signal. The bc630AT has the ability to generate heartbeat pulses/interrupts between a 1-2000 Hz rate. The Masks Register bit 4 enables the heartbeat, and the registers HBRATE0 and HBRATE1 determine the rate of the pulses/interrupts (in pulses per second). Following a change to the Heartbeat Rate registers the user must 'Initialize' the bc630AT or issue the 'Synchronize Heartbeats' command.

Heartbeat rates above 2000 pulses per second are not recommended. The Heartbeats are synchronized to on-time, therefore a heartbeat rate of 1000 would generate a positive pulse each millisecond. (i.e. at 1.000 000, 1.000 001, ...1.999 999 , 2.000 000 seconds). The rising edge of the heartbeat pulse occurs on-time. This edge is generated by hardware and should be used as the reference. The falling edge is controlled by software resulting in a small variance of the pulse width. The heartbeat pulses are available on J1 pin 15 and the BNC connector (if jumpered for heartbeats).

The heartbeats can generate XT/AT bus interrupts if selected as the interrupt source in the Masks Register (See Section 3.1.4). The bc630AT generates heartbeat interrupts to the PC by writing 60H to the Interrupt Register, offset 0FH. The interrupt is cleared when the host PC reads the Interrupt Register. The host can optionally use the Heartbeat Interrupt in a polled mode.
The bc630AT achieves synchronization of the heartbeats to the time source by computing the heartbeat pulse/interrupt period by dividing the Time Code Period (see section 3.5) by the programmed heartbeat rate (HBRATE1, HBRATE0). The bc630AT does the heartbeat period calculations in response to the 'Initialize' and 'Synchronize Heartbeats' commands and at power on. The bc630AT must be decoding time code or 'flywheeling' with a valid Time Code Period to insure accurate heartbeat synchronization.

The 'Initialize' command is valid for heartbeat synchronization only when the 'Initialization mode' is set to normal in the Time Code Select register (see section 3.1.3). When other 'Initialization modes' are selected the initial heartbeat period will be calculated based on the time code period specified by the mode.

The 'Synchronize Heartbeats' command allows on the fly heartbeat programming without disruption of the bc630AT's clock. It is a valuable command for changing the heartbeat rate or re-synchronizing an existing heartbeat rate to the time code. The heartbeats become valid at the on-time mark after the command completes.

Varying temperature changes to the bc630AT may result in a small drift of the Time Code Period measurement. When this occurs the heartbeats will become slightly out of sync near the end of the one second time period (possibly a few microseconds). To assure the highest synchronization accuracy’s of the heartbeat pulses/interrupts, the host XT/AT computer may issue the 'Synchronize Heartbeats' command when a drift in the time code period is detected or at a periodic rate.

Note: The heartbeat feature is not available for the IRIG A time code format.

3.4.5 PROPAGATION DELAY COMPENSATION

When the time code source is located an appreciable distance from the bc630AT, a significant propagation delay will be introduced. The bc630AT 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 'Initialize' the bc630AT.

The PROPDEL0 (MSB) and PROPDEL1 (LSB) registers are loaded with a 16-bit signed value

between -2048 and +2047. Each unit represents 0.5 :sec of delay, giving a range of delay values between -1000 :sec and +1000 :sec. Positive values advance the 1PPS and heartbeat relative to the time code; negative values retard the 1PPS and heartbeat. For example, if the cable between the time code source and the bc630AT introduces a delay of 50 :sec, and you want to make the system act as if the time code source is sitting right next to the bc630AT, then load the value +100 into the propagation delay registers.

The propagation delay adjustment is also useful to maximize the synchronization accuracy of the bc630AT to an external time source or other bc630AT's. The procedure is summarized as follows:

- With a dual channel oscilloscope connect the 1PPS output from the bc630AT to channel. Trigger the scope with the rising edge of the 1PPS.

- Connect the external time code or 1PPS of another bc630AT to the channel 2 input of the oscilloscope.

- Adjust the propagation delay until the 1PPS on channel 1 aligns perfectly with the on- time mark of the channel 2 time signal.

3.5 SUBSECOND COUNT AND TIME CODE PERIOD

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

The subsecond count in conjunction with the Time Code Period are used to derive an accurate subsecond time. The subsecond count is an unsigned 20-bit binary number, and the Time Code Period is an unsigned 16-bit binary number. The subsecond count multiplied by 0.5 :sec 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 bc630AT'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. The frequency error between the clock frequencies is computed by analyzing the Time Code Period bytes.

If the bc630AT 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 Time Code Period bytes contain the value 33920 decimal (8480 hex) which is 2E6 modulo 65536. If for example, the Time Code Period bytes contain the value 34000 decimal (33920 + 80), then the bc630AT 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). Likewise, if the Time Code Period bytes contain the value 33840 decimal (33920 - 80) then the bc630AT 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 + (Time Code Period - 33920))

When scaled by the factor on the bottom of the previous page, the subsecond count will always vary between 0 and 2E6-1, giving an accurate number of 0.5 :sec counts. This methodology allows the user to completely remove the effects of the bc630AT crystal clock frequency offset and aging.

The user may also simply divide the subsecond count by the factor (2E6 + (Time Code Period - 33920)) to produce a real number which varies between 0.0000000 and 0.9999995.
The programming examples in Section 6 provide some useful data structures and algorithms for working with the subsecond count and Time Code Period values.

3.6 XT/AT BUS INTERRUPTS

The bc630AT interrupts the XT/AT computer by writing a value to the Interrupt register. (Refer to section 3.1.3.) When this value is written, the jumper selected interrupt (IRQ3-7, IRQ10-12 or IRQ14-15) is pulled high. The interrupt is cleared by reading the interrupt register. The host PC can use the interrupt feature in a polled mode by clearing the interrupt register and polling for the respective interrupt value. Polling rates must not exceed 500 kHz.

3.7 REAL TIME CLOCK IC

The bc630AT has an independent battery backed Real Time Clock (RTC) IC which maintains time during power down of the XT/AT computer. The RTC IC time can be set and requested with the 'Set RTC IC Time', 'Request RTC IC Time' and 'Synchronize RTC IC' commands. The RTC IC maintains time with month/day/year, hour/minutes/seconds and day of week format. As an option in the Time Code Format Select register, the RTC IC can be selected for decoding by the bc630AT and read with microsecond precision through the standard Time Request commands.

3.7.1 REQUESTING TIME FROM THE RTC IC

Command 'Request RTC Time' retrieves the time of day directly from the independent RTC IC and loads it into the Data Address Space. The time is in month/year/day format and has a resolution of one second.

3.7.2 DECODING THE RTC IC TIME WITH MICROSECOND RESOLUTION

The bc630AT Real Time Clock Module can optionally supply time with microsecond precision without an external time code. This is accomplished by selecting the battery backed RTC IC for decoding in the TCSEL register (See Section 3.1.3) followed by an 'Initialize' command. The bc630AT now retrieves the time from the battery backed RTC IC. The battery backed RTC IC disciplines the bc630AT clock with its 1PPS output. All bc630AT features are available in this mode.

3.7.3 SETTING THE RTC IC TIME

Setting the time of the RTC IC is accomplished by loading the desired time into DAT00 - DAT06 followed by the 'Set RTC IC Time' command. The format of the Day of Week data is: Sunday = 0, Monday = 1 ... Friday = 5, Saturday = 6. The RTC IC maintains this time with automatic leap year adjustment independent of the time decoded by the bc630AT. The RTC IC time can also be set (synchronized) to the time currently being decoded by the bc630AT. (Refer to section 3.7.4).

3.7.4 SYNCHRONIZING THE RTC IC TO EXTERNAL TIME CODE

Synchronization of the RTC IC to external time code is achieved by issuing the 'Synchronize RTC IC' command. The ready status value of 14H is returned after the synchronization of the RTC IC is initiated by the bc630AT. However, the synchronization process requires two seconds of time code therefore the synchronization is complete approximately two seconds later.

To properly sync the month/day data of the RTC IC, the year data must be loaded into the RTC IC before synchronization. This allows the bc630AT to do the proper conversion of the Julian Time Code Format to the month/day/year format of the RTC IC with automatic leap year adjustment. The bc630AT automatically detects an invalid Julian date. (i.e. number of days = 0 or greater than 365 [366 in a leap year]). If an invalid number of days is detected, the bc630AT synchronizes only the hour/min/second time in the RTC IC. The user is responsible for providing the proper day-of-week data. However, this data is optional.

3.8 EXTERNAL 1PPS OR 1, 5, 10 MHz PRECISION REFERENCE

The Time Code Select register has options to decode an external 1PPS or 1, 5, 10 MHz frequency references. When these modes are selected the bc630AT does a one time load of the time from the battery backed RTC IC and maintains that time with the selected external reference signal.

When the 1, 5, 10 MHz mode is selected, the bc630AT:

- Automatically detects the external frequency source as 1, 5, or 10 MHz.
- Performs a one time load of the major time from the battery backed RTC IC.
- Disciplines the bc630AT clock with the external frequency.

The 1PPS mode disciplines the bc630AT clock with the rising edge of the external pulse.

3.8.1 FLYWHEELING WITH AN EXTERNAL 1PPS OR 1, 5, 10 MHz REFERENCE

The bc630AT enters a 'flywheeling' mode whenever the decoded time source drops out or becomes corrupted. While flywheeling, the bc630AT continues to provide uninterrupted time based on the Time Code Period measurement obtained from the decoded time source.

The highest 'flywheel' accuracy's are maintained with the filter enabled. (Refer to section 3.1.4.) When enabled, the bc630AT averages the decoded time code period measurements over time which results in higher flywheeling accuracy's. Flywheeling accuracy's of better than 1 part in 10E6 (less than 3.6 milliseconds per hour of drift) are typical with stable ambient temperatures. Flywheel accuracy, however, degrades with fluctuating temperatures.

The external 1PPS and 1, 5, 10 MHz reference inputs provide a means to maintain higher flywheeling accuracy's over temperature. When the external flywheel is enabled, the bc630AT automatically detects a valid 1PPS or 1, 5, 10 MHz frequency and, if found, adjusts the bc630AT's Time Code Period to match the one second time period of the external reference.

3.9 MASTER/SLAVE OPERATION

The bc630AT Real Time Clock Module has the capability of operating in a Master/Slave mode in which Slave bc630AT's synchronize digitally to the time base of a Master bc630AT. This allows computer synchronization of two or more PC's without external time code. Synchronization accuracy's of " 1 microsecond (with filter enabled) can be achieved between the Master and Slaves. The Master/Slave mode of operation is summarized as follows:

- A Master bc630AT decodes the time source selected in it's Time Code Select register. A 1PPS synchronization pulse and serial 9600 Baud major time message (days - seconds) is broadcast via two RS-422 lines approximately each second through the RJ11 connector (J2). The Master broadcasts these digital time signals when decoding time or flywheeling. The necessary signals for synchronization are the 1PPS synchronization pulse and the corresponding "buf2 ddd:hh:mm:ss" time message. (Refer to Section 4.2.1.)

- Slave bc630AT's select the Master bc630AT as the time source in their Time Code Select register. The RJ11 connectors (J2) of the Master and Slave are linked with 6 conductor telephone cable to the Master bc630AT. When a valid 1PPS and time message is received from the Master, synchronization takes place. All features of the bc630AT are available with this mode including "flywheeling". (Refer to Figures 3-1 and 3-2). No signals are transmitted to the RJ11 connector from the Slave bc630AT.

- Terminate the RS-422 line with the bc630AT's termination resistors by moving the existing jumper on JP3 pins 1, 2 to JP3 pins 3, 4 and move the jumper on JP3 pins 5, 6 to JP3 pins 7, 8 (Refer to Table 4-7). The two bc630AT's on the end of the network should be terminated.

Note: On a Master\Slave bc630AT network only two modules should be terminated.

Any one of the bc630AT's depicted in Figure 3-1 and 3-2 can serve as a Master. The Master is the only board in the network that transmits Time and Synchronization pulses on the RS-422 lines. If more than one bc630AT is operating as a Master, the time signals are corrupted. There is no hardware damage when this occurs but network synchronization is not possible. When a bc630AT is designated as a slave (refer to step 2) all RS-422 ports are placed in receive mode.

Figure 3-1
Serial Daisy Chain of Master/Slave bc630AT's

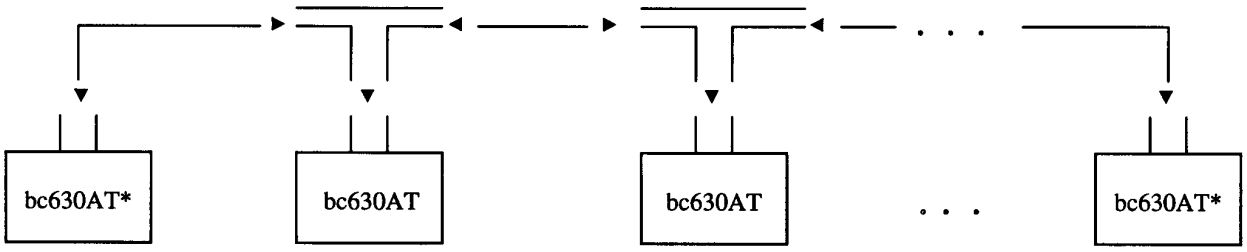
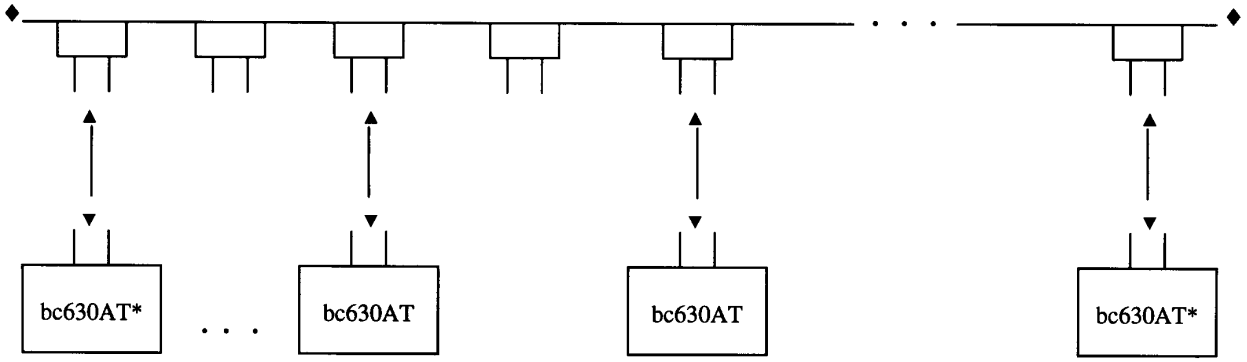
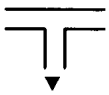


Figure 3-2
Bc630AT Master/Slave Serial Distribution Bus with Multiple Drop Points



Wall mount junction connector with female RJ11 mating connector.



"T" connector: 2 female and 1 male RJ11 connectors.

6 conductor phone cable with RJ11 male connector.

bc630AT RJ11 Port (Female mating connector)

Termination: Choice of <u>one</u> of the following methods:

♦ Denotes direct 100 ohm termination on RS-422 line.

***** Denotes termination by on-board 100 ohm resistors.
(Necessary if RS-422 line is not directly terminated)

CHAPTER FOUR
INPUT/OUTPUT CONNECTORS

4.0 GENERAL

All I/O signals (except the PDC signals) are available on the rear panel connectors. Time code input is available on the rear panel 15 pin DS connector and, if jumpered, the BNC connector. The PDC signals are available on a 20 pin rectangular connector located near the top edge of the module. The location of all connectors is shown in Figure 4-1.

4.1 J1 DS 15 CONNECTOR INPUT/OUTPUT

All I/O signals (except the PDC signals and serial output) are connected to the front panel 15 pin DS connector labeled 'J1' and are connected in parallel with the other connectors which carry these signals. The pinout assignments for the J1 connector are shown in Table 4-1.

4.2 J2 SERIAL RS-422 DIFFERENTIAL OUTPUT

A serial RS-422 differential output is available on the rear panel phone jack connector labeled 'J2'. A 6 wire phone jack can be used with J2. The pinouts for J2 are listed in Table 4-2. Figure 4-2 shows the orientation of the J2 pinouts. This output is used for linking bc630AT's used in a Master/Slave configuration. Regular telephone wire is used connecting the Slave bc630AT's in parallel to the master.

This output port can also be used for diagnostic purposes. The bc630AT broadcasts time, measured code period, code detection status, code search status, software version and date code, AGC slice level, and control byte information. Data is sent at 9600 baud, 8 data bits, one stop bit, no parity. Each line is terminated by a carriage return (0DH) and a line feed (0AH).

The Tx(-) signal will drive most RS-232 interfaces such as a serial port to a PC. The data pacing signals RTS, CTS, DSR and DTR may need to be jumpered depending on the specific terminal emulation software in use.

4.2.1 bc630AT RS-422 SERIAL OUTPUT MESSAGES

When a valid time code signal is being decoded, a packet of data is sent every second, about 1/2 second after the 1PPS signal occurs. The following are examples of message data when decoding the respective time code signals, real time clock and other time sources.

Initialization Message:

```
SLAMR bc630AT 3.0  10/29/91
initialization complete
control bytes 0/1/2 = XX/XX
hb period remainder = XXXX XXXX
```

Decoding modulated time code:

```
(Time code type)              {iriga, irigb, nasa36, 2137}
lev = XX                      {X represents a hexadecimal digit}
dif = XXXX
buf2 ddd hh:mm:ss
etim ddd hh:mm:ss :: :: ::
```

Decoding RTC, 1PPS, External Reference and DC Level Shift time code

```
(Time Source)                 {iriga - dcls, irigb - dcls, nasa36 - dcls, RTC,
                               RTC 1pps, RTC xref}
dif = XXXX
buf2 ddd hh:mm:ss
etim ddd hh:mm:ss :: :: ::
```

Searching for time code with Time Code Select register set for Autodetect:

```
checking (time code type)     {Modulated  time  code:  irigb,  iriga,  nasa36,
2137}
lev = cannot find (time code type)

checking (time code type)     {Non-modulated time codes: irigb - dcls, iriga -
                               dcls, nasa36 - dcls}
cannot find (time code type)
```

Searching for time code with the Time Code Select register set for for a specific time source:

```
(time code type)              {iriga, irigb, nasa36, 2137, iriga - dcls, irigb
-                              dcls, nasa36 - dcls,}
decode error !!               {RTC, RTC xref, RTC 1pps}
```

Time code error message:

```
decode error !!
```

Flywheeling message:

```
flywheeling !!
(flywheel source)             {int xtal, ext 1PPS, 1 MHz, 5 MHz, 10 MHz}
buf2 ddd hh:mm:ss
etim ddd hh:mm:ss :: :: ::
```

Notes: **1)** "lev" is a hex representation of the "slice" level of a modulated time code. It is used to determine whether a cycle is a high or low cycle.

2) "dif" is hexadecimal representation of the time code period measurement. It is used to scale the internal time. The "dif" value reflects the difference between the internal crystal and the external time signal.

3) No data is transmitted out the serial port when the bc630AT is set up as a Slave to decode a Master bc630AT.

4) RS-422 data is only received by the bc630AT when the card is configured as a Slave. The normal message data and a 1PPS signal broadcast from the Master is decoded by the Slave for time synchronization.

5) No control codes or commands are accepted by the user through the RS-422 port.

6) The RS-422 messages were provided primarily for diagnostic purposes. The user, however, may use the data to obtain time externally from the board. Once a valid time code is being decoded, time/message data is broadcast once each second about 1/2 second after the 1PPS signal. The time being broadcast, following the "buf2" label, is valid at the rising edge of the preceding 1PPS pulse.

4.3 J3 BNC CONNECTOR INPUT/OUTPUT

The BNC connector J3 can be chosen to input or output 1 of 10 signals from the DS 15 connector J1. The signal desired is selected with jumper JP2. Table 4-6 lists the jumper field options for JP2.

4.4 J6 PDC CONNECTOR

The 20 pin connector J6 carries the PDC signals which can be used to drive Datum Inc time display modules. The pin assignments for the J6 connector are shown in Table 4-3 and the PDC data format in Table 4-4.

4.5 PDC SIGNAL DESCRIPTION

The signals carried on the PDC (Peripheral Data Connector) are generally used to drive other Datum Inc products which require decoded time such as the M80D-LU Computer Top LED Display. These signals can, however, be used by the user for a variety of applications.

Decoded time is transmitted over the PDC once per second using the PDC ENABLE* and D0-D7 lines in a byte serial fashion (i.e a burst of 9 bytes are sent) just after the on time mark. The D0-D7

encoding is shown in Table 4-4. The lower nibble contains the BCD encoded time digit and the upper nibble determines which digit has been transmitted. The D0-D3 data lines are valid at the rising edge of PDC ENABLE*. The D4-D7 data lines are valid anytime PDC ENABLE* is low.

EACH CYCLE is a TTL representation of the time code carrier (high during positive half cycles, low during negative half cycles). 1PPS is a TTL signal whose rising edge occurs once per second on time.

Figure 4-1
bc630AT Real Time Clock Module Connector Layout

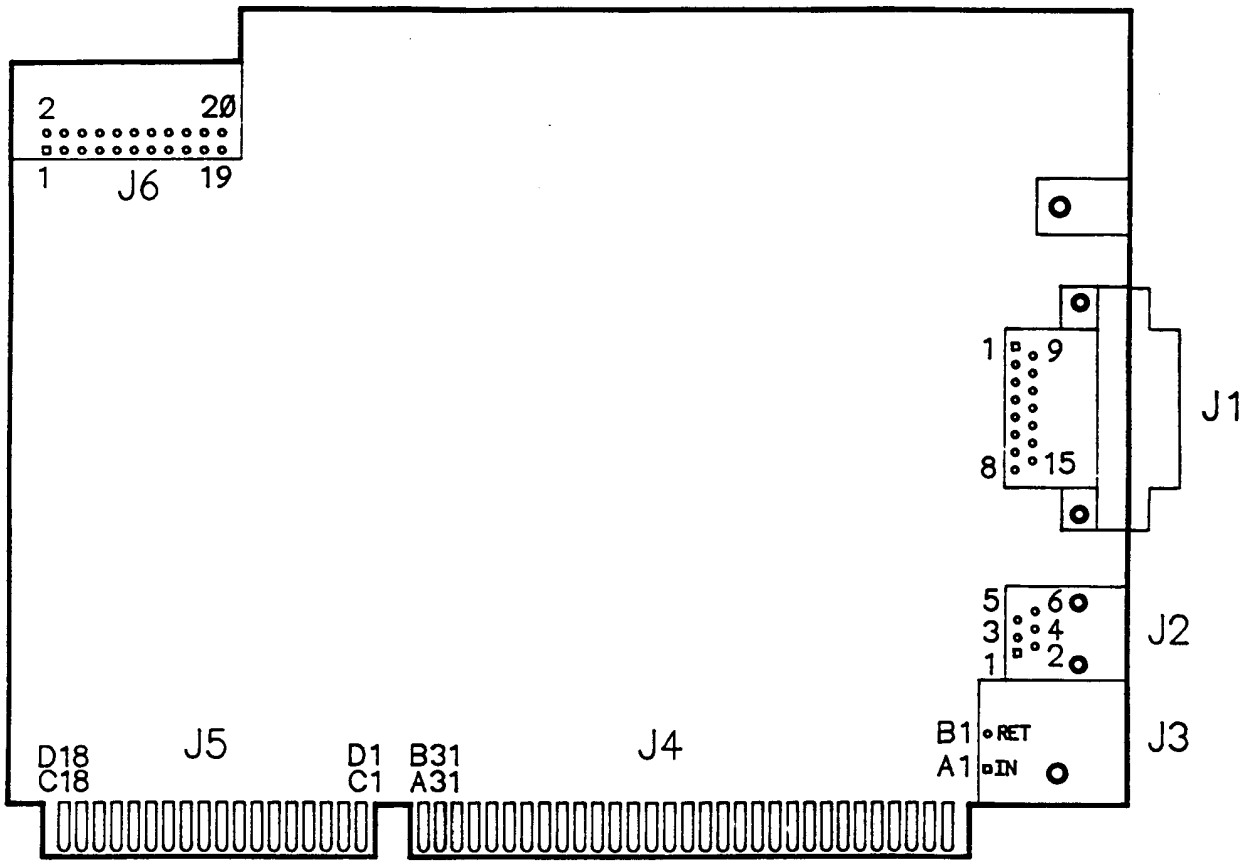


Table 4-1
J1 I/O Connector Pinouts

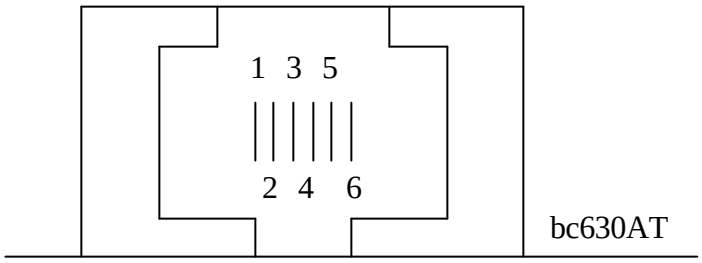
J1	Signal Description
1	Time Code Input
2	Ground
3	(Future Implementation)
4	(Future Implementation)
5	Event Trigger Input
6	1PPS Output
7	Ground
8	DC Level Shift Input
9	Isolated Return
10	External 1, 5, 10 MHz Input
11	Auxiliary 1 Input
12	Ground
13	Auxiliary 2 Input
14	Heartbeat Pulse Output
15	External 1PPS Input

Table 4-2
J2 Serial RS422 I/O Pinouts

J2	Signal Description
1	1PPS TX/RX (+)
2	Ground
3	DATA TX/RX (+)
4	DATA TX/RX (-)
5	Ground
6	1PPS TX/RX (-)

Note: The above pinouts apply to the bc630AT version C hardware. The pinouts differ from the version A and B boards.

Figure 4-2
J2 Phone Jack Pinout Orientation



Front View

Table 4-3
J6 PDC Connector Pinouts

J6	Signal Description
1	Ground
2	PDC ENABLE*
3	D0
4	D1
5	D2
6	D3
7	D4
8	D5
9	D6
10	D7
11	EACH CYCLE / HEARTBEAT
12	Not Used
13	1PPS
14-17	Not Used
18	Ground
19	+5 VDC
20	+5 VDC

Table 4-4
J6 - PDC Data Format

D7	D6	D5	D4	D3 - D0 (BCD)
0	0	0	0	Days Hundreds
0	0	0	1	Days Tens
0	0	1	0	Days Units
0	0	1	1	Hours Tens
0	1	0	0	Hours Units
0	1	0	1	Minutes Tens
0	1	1	0	Minutes Units
0	1	1	1	Seconds Tens
1	0	0	0	Seconds Units

Table 4-5
JP1 Jumper Positions for Isolated Time Code

JP1 Position	Time Code Input Jumper For J1
* 1 to 2	Non-isolated Time Code Input
3 to 4	Isolated Time Code Input (Opt)
* 5 to 6	Non-isolated Time Code Input
7 to 8	Isolated Time Code Input (Opt)
* 9 to 10	Grounded Return J1
11 to 12	Isolated Time Code Return (Opt)

* Factory Setting, Total of 3 Jumpers.

Table 4-6
JP2 Jumper Positions for BNC (J3) Signal Selection

JP2 Position	BNC Signal Selection
1 to 2	No Connection
3 to 4	Reserved
5 to 6	Reserved
7 to 8	Event Trigger Input
9 to 10	Heartbeat Pulse Output
11 to 12	1PPS Output
13 to 14	External 1PPS Input
15 to 16	DC Level Shift Time Code
17 to 18	External 1, 5, 10 MHz Input
* 19 to 20	Time Code Input (Modulated)

* Factory Setting.

Table 4-7
JP3 Jumper Positions for Master/Slave Termination

JP3 Position	Master/Slave Termination
* 1 to 2	No Termination
3 to 4	100 ohm Termination
* 5 to 6	No Termination
7 to 8	100 ohm Termination

* Factory Setting.

Table 4-8
JP4 Jumper Positions for Time Code Return

JP4 Position	J3 Time Code Return Jumper
* 1 to 2	Grounded Return
3 to 4	Isolated Time Code Return (Opt)

* Factory Setting.

Table 4-9
JP5 Jumper Positions for XT/AT Bus Interrupt Selection

JP5 Position	XT/AT Interrupt
* 1 to2	IRQ2 (XT), IRQ9 (AT)
3 to 4	IRQ3
5 to 6	IRQ4
7 to 8	IRQ5
9 to10	IRQ6
11 to 12	IRQ7
13 to 14	IRQ10
15 to 16	IRQ11
17 to 18	IRQ12
19 to 20	IRQ14
21 to 22	IRQ15

* Factory Setting.

Table 4-10
JP6 Jumper Positions for J6 PDC Pin #13

JP6 POSITION	J6 PDC PIN #13 JUMPER
1 to 2	Heartbeat Pulse
* 3 to 4	Each Cycle Clock

* Factory Setting.

CHAPTER FIVE
THEORY OF OPERATION

5.0 GENERAL

This section describes the theory of operation for the bc630AT Real Time Clock Module. Reference Schematic Diagram 11450.

5.1 TIME CODE DECODER

The heart of the bc630AT utilizes Datum Inc's reduced chip set decoder circuitry. Details on the operation of this circuitry are not available at this time.

5.2 XT/AT bus INTERFACE

The XT/AT bus interface consists of the usual assortment of bus transceivers, buffers, and decoders commonly found on any microprocessor based system. The bc630AT uses an 8 bit dual port RAM for the I/O interface.

CHAPTER SIX
PROGRAMMING EXAMPLES

6.0 GENERAL

This section provides programming examples to illustrate the operation of the bc630AT. The examples are written in Turbo C. Hexadecimal constants are specified in C by using the prefix '0x'.

The BASE constant refers to the base address setting of SW1 (refer to Chapter Two). 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.

6.1 SETTING UP THE CONTROL REGISTERS

This example shows how to program the bc630AT control registers. The bc630AT is setup to automatically detect an IRIG A, IRIG B, NASA 36 or 2137 modulated or DC level shift time code input signal, disable external event time capture, disable heartbeat pulses, disable the filter, disable external flywheel synchronization and select no interrupt sources. The example manually sets up the bc630AT to the default conditions.

```
                                    /* Set up control registers       */
outportb(BASE+TCSEL, 0);            /* Select automatic time code     */
                                    /* selection                      */
                                    /* Initialization mode = normal   */
outportb(BASE+MASKS, 0);            /* Disable event capture, disable */
                                    /* filter, external flywheel      */
                                    /* disabled, disable heartbeats, no */
                                    /* interrupts                     */
outportb(BASE+HBDIV0, 0xE8);        /* Set heartbeat pulse/interrupt  */
outportb(BASE+HBDIV0, 0x03);        /* frequency 1000 Hertz           */

outportb(BASE+PROPDEL0, 0);         /* Set propagation delay to 0     */
outportb(BASE+PROPDEL1, 0);

outportb(BASE+COMMAND, 0x90)        /* Initialize bc630AT to above    */
do                                  /* control settings               */
{
     time_delay();                  /* User defined time delay function */
                                    /* that slows polling rate of     */
                                    /* COMMAND byte to less than 500 KHz */
}
while (inportb(BASE+COMMAND) != 0x10); /* Wait for ready value of 0x10 from */
                                    /* bc630AT, indicating 'Initialize' */
                                    /* complete                       */

time_delay()
{
     int i;
     for (i=100; i != 0; i--);      /* Initial value for i is dependent */
}                                   /* on host PC's clock speed to slow */
                                    /* polling rate to < 500 KHz      */
```

6.2 CAPTURING AND READING TIME

This example shows how to capture time across the XT/AT bus using the 'Request Time' command. The bc630AT COMMAND register is polled to determine when the time data is valid. Before time is requested the Timecode Period data is obtained with the 'Output Auxiliary Data' command. This data is used to compute the frequency error. The time read corresponds to the time code selected in the TCSEL register.

```
char time[9];                          /* Major BCD time array secs - days  */
int freq_error;                        /* hundreds                          */
unsigned int tc_period;
long unsigned int sub_seconds;


                                       /* Get timecode period and compute   */
                                       /* frequency error                   */
outportb(BASE+COMMAND, 0x83);          /* Request auxiliary data            */
do
{
     time_delay();                     /* Function slows polling rate to    */
}                                      /* < 500 KHz                         */
while(inportb(BASE+COMMAND) != 3);     /* Wait for ready flag of 3 (Polling */
                                       /* rate not to exceed 500 kHz)       */

tc_period =  inportb(BASE+1) * 0x100;  /* Get MSB of time code period and   */
                                       /* shift                             */
tc_period += inportb(BASE+0);          /* Get LSB of time code and sum      */
freq_error = tc_period - 33920;        /* Figure frequency error            */

outportb(BASE+COMMAND, 0x80);          /* Request time                      */
do
{
     time_delay();                     /* Function slows polling rate to    */
}                                      /* < 500 KHz                         */
while(inportb(BASE+COMMAND) != 0);     /* Wait for ready flag of 0 (Polling */
                                       /* rate not to exceed 500 kHz)       */


                                                /* Load major time into  */
                                                /* time[] array          */
time[0] = (inportb(BASE+0) & 0xF) | 0x30;       /* Get second units and  */
                                                /* convert to BCD        */
time[1] = ((inportb(BASE+0) & 0xF0) >> 4) | 0x30;  /* Get second tens and   */
                                                /* convert to BCD        */

time[2] = (inportb(BASE+1) & 0xF) | 0x30;       /* Get minute units and  */
                                                /* convert to BCD        */
time[3] = ((inportb(BASE+1) & 0xF0) >> 4) | 0x30;  /* Get minute tens and   */
                                                /* convert to BCD        */

time[4] = (inportb(BASE+2) & 0xF) | 0x30;       /* Get hours units and   */
                                                /* convert to BCD        */
```

```
time[5] = ((inportb(BASE+2) & 0xF0) >> 4) | 0x30;  /* Get hours tens and    */
                                                   /* convert to BCD        */

time[6] = (inportb(BASE+3) & 0xF) | 0x30;          /* Get days units and    */
                                                   /* convert to BCD
*/time[7] = ((inportb(BASE+3) & 0xF0) >> 4) | 0x30;   /* Get days tens and  */
                                                   /* convert to BCD        */

time[8] = ((inportb(BASE+4) & 0xC0) >> 6) | 0x30); /* Get days hundreds and */
                                                   /* shift and convert to  */
                                                   /* BCD                   */
                                                   /* Major BCD time now in */
                                                   /* time[]                */


                                                      /* Get sub second  */
                                                      /* time            */
sub_seconds  = (inportb(base+DAT04) & 0x1F) * 0x10000L;  /* Get sub second  */
                                                      /* time bits 16-20 */
sub_seconds += inportb(base+DAT05);                   /* Get & sum sub   */
                                                      /* second time bits*/
                                                      /* 0-7             */
sub_seconds += inportb(base+DAT06) * 0x100L;          /* Get & sum sub   */
                                                      /* second time bits*/
                                                      /* 8-15            */

sub_seconds  = (sub_seconds / (2E6+freq_error)) * 1E6;   /* Compute real    */
                                                      /* :secs with     */
                                                      /* offset          */
                                                      /* Sub second time */
                                                      /* now in          */
                                                      /* sub_seconds     */
```

6.3 SETTING THE REAL TIME CLOCK CHIP

The following example shows how to set the time of the independent, battery backed Real Time Clock Chip.

```
                                                   /* Set time in RTC chip
to 01/22/91, */
                                  /* Tuesday, 13:23:45                 */
outportb(BASE+0, 0x45);           /* Load seconds data into Data       */
                                  /* Transfer Address Space            */
outportb(BASE+1, 0x23);           /* Load minutes data into Data       */
                                  /* Transfer Address Space            */
outportb(BASE+2, 0x13);           /* Load hours data into Data         */
                                  /* Transfer Address Space            */
outportb(BASE+3, 0x22);           /* Load days data into Data Transfer */
                                  /* Address Space                     */
outportb(BASE+4, 0x01);           /* Load months data into Data        */
                                  /* Transfer Address Space            */
outportb(BASE+5, 0x91);           /* Load year data into Data Transfer */
                                  /* Address Space                     */
outportb(BASE+6, 0x02);           /* Load day of week data into Data   */
                                  /* Transfer Address Space            */
```

```
outportb(BASE+COMMAND, 0x84);          /* Issue 'Set RTC Chip Time' command */
do
{
     time_delay();                     /* Function slows polling rate to    */
}                                      /* < 500 KHz                         */
while (inportb(BASE+COMMAND) != 4);    /* Wait for ready flag status of 4   */
                                       /* Polling rate not to exceed 500 kHz*/
                                          /* RTC Chip time successfully set */
```

6.4 SYNCHRONIZING THE RTC IC TO EXTERNAL TIME CODE

The following example illustrates how to synchronize the Battery Backed RTC IC time to the external time code. Before synchronizing be sure RTC IC is loaded with the valid year. The example assumes the year is accurately set in the RTC chip.

```
outportb(BASE+COMMAND, 0x93);            /* Initialize bc630AT              */
do
{
     time_delay();                       /* Function slows polling rate to  */
}                                        /* < 500 KHz                       */
while (inportb(BASE+COMMAND) != 0x13);   /* Wait for ready value of 0x13    */
                                         /* from bc630AT, indicating        */
                                         /* synchronization process         */
                                         /* initiated.  Process will be     */
                                         /* complete in approximately two   */
                                         /* seconds                         */
                                         /* Polling rate not to exceed      */
/* 500 KHz                          */
```

CHAPTER SEVEN
ADJUSTMENTS

7.0 GENERAL

The bc630AT 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 R6.

7.1 PHASE LOCK LOOP ADJUSTMENT

The following steps assume that the bc630AT is correctly mounted on a baseboard and is exposed to allow access by an oscilloscope probe. A source of IRIG B time code is required.

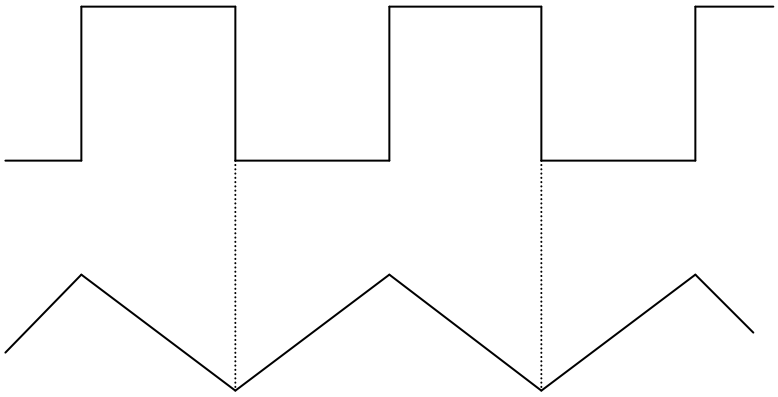
- Select an IRIG B timecode by writing a 'B' (42H) to offset location 08H or in demonstration program 630DEMO2 select "IRIG B ONLY" for the time code select register. This step forces the bc630AT to search for a 1000Hz carrier frequency only.

- Input the IRIG B time code through the BNC connector (J3) or pin #1 of J1.

- Use pin #16 of U2 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 R6 so that the negative going transitions on pin #16 are centered on the negative "peaks" of the carrier. See figure 7-1 below.

Figure 7-1
Phase Lock Loop Adjustment



CHAPTER EIGHT
DRAWING SET

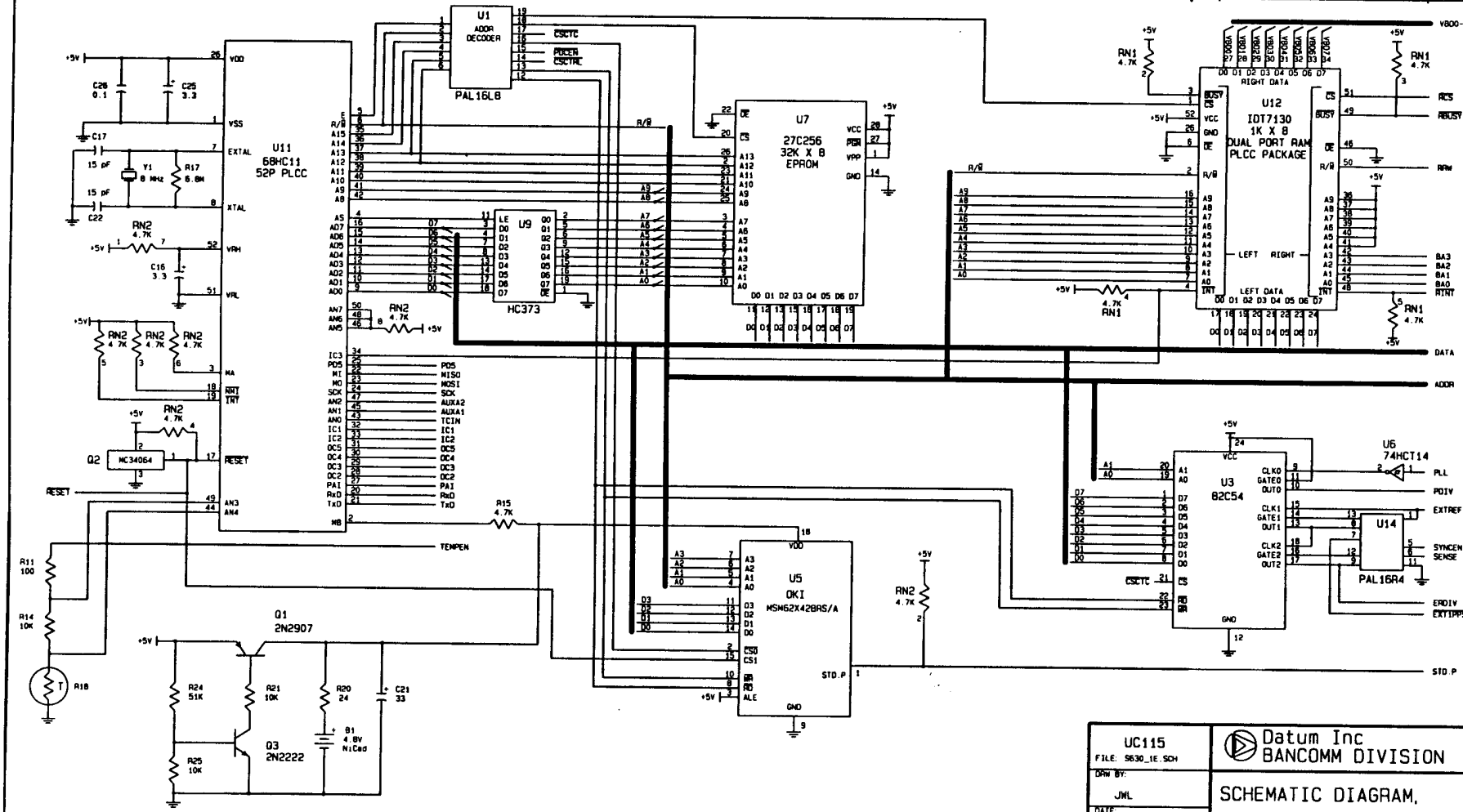
8.0 GENERAL

This chapter contains the schematic diagram, assembly drawing, and parts list for the bc630AT.

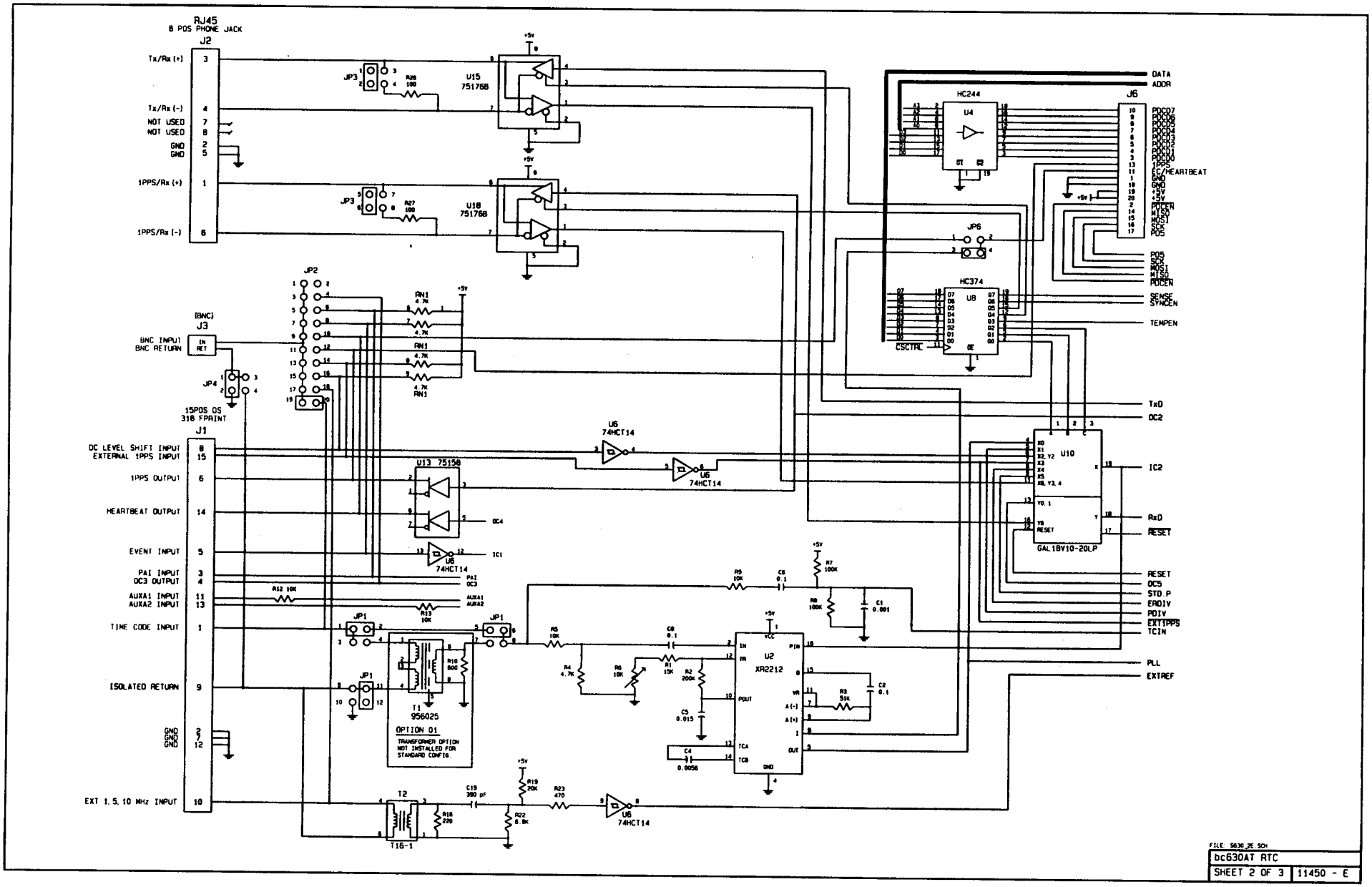
NOTES: UNLESS OTHERWISE SPECIFIED

1. ALL COMPONENT VALUES ARE IN OHMS AND MICROFARADS.

REVISIONS			
REV	DESCRIPTION	DATE	APPROVED
A	FIRST RELEASED	10-09-90	JF
B	REV PER ECO: 183	11-18-90	JF
C	REV PER ECO: 191	07-12-91	RH
D	REV PER ECO: 254	04-21-95	RH
E	REV PER ECO: 279	11-06-95	MB

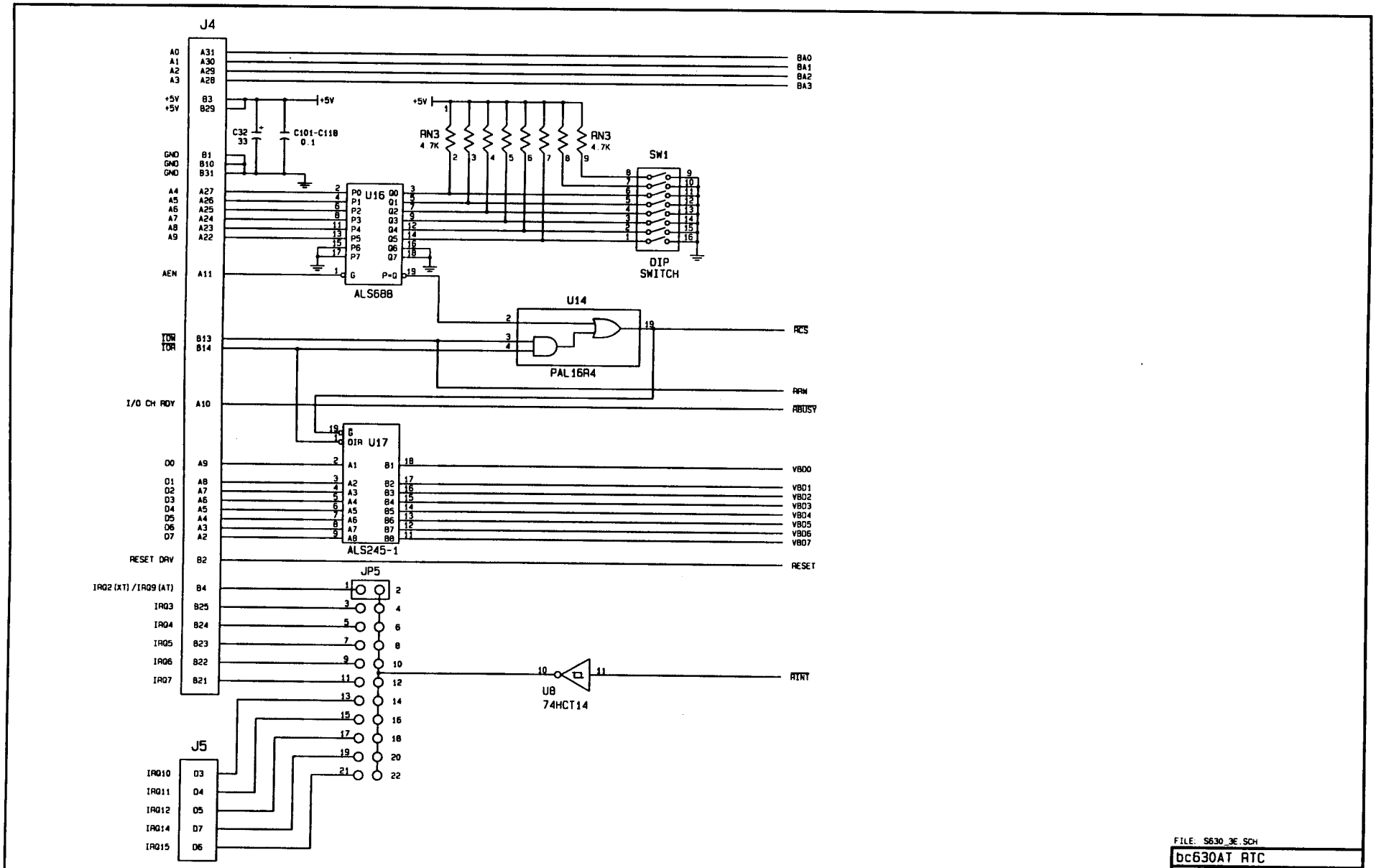


UC115 FILE: 5630_1E.SCH	Datum Inc BANCOMM DIVISION
DRW BY: JWL	SCHEMATIC DIAGRAM, bc630AT REAL TIME CLOCK
DATE: Sept 06, 1995	
APPROVED BY:	SHEET 1 OF 3 | 11450 - E



FILE: 5630_2E.SCH

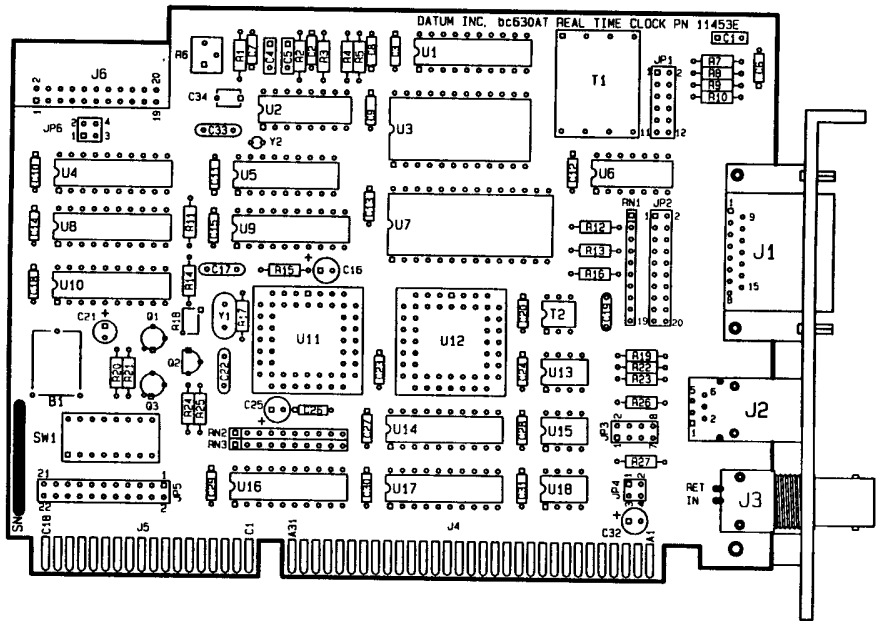
bc630AT RTC	
SHEET 2 OF 3	11450 - E



FILE: S630_3E.SCH

bc630AT RTC	
SHEET 3 OF 3	11450 - E

REVISION			
LTR	DESCRIPTION	DATE	APP.
A	PRODUCTION RELEASED	10/09/90	JF
B	REV PER ECO: 183	11/18/90	JF
C	REV PER ECO: 191	07/12/91	RH
D	REV PER ECO: 254	04/21/95	RH
E	REV PER ECO: 278	11/06/95	MB



UC115 BCM072.ZIP FILE: A630_1E.SCH	Datum Inc BANCOMM DIVISION	
DRN BY: JWL	ASSEMBLY DIAGRAM, bc630AT REAL TIME CLOCK	
DATE: Sept 6, 1995		
CHECK BY:	SHEET: 1 OF 2	11453 - E

Assembly, Parts Listing
bc630AT Real Time Clock

Ref: Drawing No. 11453 E
Ref: UC 115
Sep 6, 1995

AS	OPT	BC P/N	MANF P/N	MANUFACTURE	VALUE	DESCRIPTION	QTY#	REF DESIG.
		1419025	11455E	BANCOMM DIV, DATUM	bc630AT	IBM PC BRKT	1.00	BKT1
		1501150	CM05CD150J03	CDE	15 PF,500V	DIPPED MICA CAPACITOR	2.00	C17,22
		1501391	CM04FA391JD	ARCO	390 PF,100V	DIPPED MICA CAPACITOR	1.00	C19
		1503335	335RMR035M	IC	3.3 MF,35V	ALUMINUM ELECTROLYTIC CAP.	2.00	C16,25
		1503336	336RMR025M	IC	33 MF,35V	ALUMINUM ELECTROLYTIC CAP.	2.00	C21,32
		1506102	SR211C102KAA	AVX	1000 PF,50V	MONO CERAMIC CAPACITOR .2 R/L	1.00	C1
		1506153	SR211C153KAA	AVX	15000 PF,100V	MONO CERAMIC CAPACITOR .2 R/L	1.00	C5
		1506562	SR211C562KAA	AVX	5600 PF,100V	MONO CERAMIC CAPACITOR .2 R/L	1.00	C4
		1515104	MD015E104MAA	AVX/67349	0.1 MF, 50V	DIP GUARD CAPACITOR	22.00	C2,3,6-15,18,20,23,24,26-31
		1701115	11452C	BANCOMM DIV, DATUM	bc630AT RTC	PRINTED CIRCUIT BOARD	1.00	PCB1
		2109020	3592-5002	3M	20 POS	RTANG HEADER	1.00	J6
		2117061	TSW-130-07-G-D	SAMTEC	2X30 POS	STRAIGHT TERMINAL STRIP	1.00	JP1=2x6, J2=2x10, JP3=2x4,
		2117061	TSW-130-07-G-D	SAMTEC	2X30 POS	STRAIGHT TERMINAL STRIP	1.00	JP4,6=2x2, JP5=2x11
		2124215	869521-1	AMP	15 POS	'D' SKT, .318 RTANG PCMNT B/L	1.00	J1
		2150020	10620-01-445	ANDON/SPECIRA	20 POS	DIP SOCKET	3.00	U1,10,14
		2150028	10628-01-445	ANDON/SPECIRA	28 POS	DIP SOCKET	1.00	U7
		2152052	641748-2	AMP	52 POS	PLCC REC CHIP CARRIER	2.00	U11,12
		2190004	SS-6466S-NF	STEWARD	RJ11 1 PT, 6 POS	SHIELDED PHONE JACK, PCMNT.	1.00	J2
		2302005	DS-800	SEIKO	8MHz	CRYSTAL	1.00	Y1
		2802001	3341-1L	3M		JACK SCREW KIT	1.00	BKT1
		3902001	PMB4.8-15-H3	PLAINVIEW INC.	4.8V	NICAD PCMNT BATTERY	1.00	B1
		4701101	RC07GF101J	ALLEN BRADLEY	100 OHM,1/4W	FIXED RESISTOR	3.00	R11,26,27
		4701103	RC07GF103J	ALLEN BRADLEY	10 K OHM,1/4W	FIXED RESISTOR	7.00	R5,9,12-14,21,25
		4701104	RC07GF104J	ALLEN BRADLEY	100 K OHM,1/4W	FIXED RESISTOR	2.00	R7,8
		4701153	RC07GF153J	ALLEN BRADLEY	15 K OHM,1/4W	FIXED RESISTOR	1.00	R1
		4701203	RC07GF203J	ALLEN BRADLEY	20 K OHM,1/4W	FIXED RESISTOR	1.00	R19
		4701204	RC07GF204J	ALLEN BRADLEY	200 K OHM,1/4W	FIXED RESISTOR	1.00	R2
		4701221	RC07GF221J	ALLEN BRADLEY	220 OHM,1/4W	FIXED RESISTOR	1.00	R16
		4701240	RC07GF240J	ALLEN BRADLEY	24 OHM,1/4W	FIXED RESISTOR	1.00	R20
		4701471	RC07GF471J	ALLEN BRADLEY	470 OHM,1/4W	FIXED RESISTOR	1.00	R23
		4701472	RC07GF472J	ALLEN BRADLEY	4.7 K OHM,1/4W	FIXED RESISTOR	2.00	R4,15
		4701513	RC07GF513J	ALLEN BRADLEY	51 K OHM,1/4W	FIXED RESISTOR	2.00	R3,24
		4701682	RC07GF682J	ALLEN BRADLEY	6.8 K OHM,1/4W	FIXED RESISTOR	1.00	R22
		4701685	RC07GF685J	ALLEN BRADLEY	6.8 M OHM,1/4W	FIXED RESISTOR	1.00	R17
		4703103	72P103	BECKMAN	10 K OHM,1/2W	SINGLE TURN POTENTIOMETER	1.00	R6
		4705472	710A472	ALLEN BRADLEY	4.7 K OHM,1/8W	C-SIP RESISTORS, 10 PIN 'X'	3.00	RN1-3
		4801002	2N2222			NPN SWITCHING/AMPLIFIER (TO18)	1.00	Q3
		4802002	2N2907A	MOTOROLA	3P D PKG	PNP SWITCHING/AMPLIFIER (TO18)	1.00	Q1
		5108002	76SB08	GRAYHILL		8PST DIP SWITCH	1.00	SW1
		5601001	MCL T16-1 (X65)	MINI-CIRCUITS	6P DIP PKG	R.F. TRANSFORMER 10:1 INPUT	1.00	T2
		9002710	74HCT14	RCA	14P DIP PKG	HEX SCHMITT INVERTER	1.00	U6
		9006658	74HC374	MOTOROLA	20P DIP PKG	OCTAL D FLIP FLOP	1.00	U8
		9008657	74HC373	VARIOUS	20P DIP PKG	OCTAL D TRANSPARENT LATCH, T/S	1.00	U9
		9015940	SN74ALS688N	TI	20P DIP PKG	8 BIT MAGNITUDE COMPARATOR	1.00	U16
		9102002	68HC11A1FN	MOTOROLA	52P PLCC PKG	MICROCOMPUTER	1.00	REF: U11 (SKT)
		9103008	82C54	AMD	24P DIP PKG .6W	CMOS COUNTER TIMER	1.00	U3
		9103035	MSM62X42BRS/A	OKI	18P DIP PKG	REAL TIME CLOCK/CALENDAR	1.00	U5
		9201030	MC34064P-5	MOTOROLA	3P CASE29-04 PKG	UNDERVOLTAGE SENSING DEVICE	1.00	Q2
		9207070	SN75158P	TI	08P DIP PKG	DUAL 50 OHM TTL LINE DRIVER	1.00	U13
		9207076	SN75176B	TI	08P DIP PKG	DIFF BUS TRANS	2.00	U15,18
		9207615	74HC244	NATIONAL	20P DIP PKG	OCTAL BUFFER/LINE DRIVER	1.00	U4
		9207920	SN74ALS245A-1N	TI	20P DIP PKG	OCTAL BUS TRANSCEIVER	1.00	U17
		9307030	XR2212CP	EXAR	16P DIP PKG	PHASE LOCKED LOOP	1.00	U2
		9405015	PAL16L8B	MMI	20P DIP PKG .3W	PAL	1.00	U1 (SKT)
		9405020	PAL16R4ACN	MMI	20P DIP PKG .3W	PAL 35 NS	1.00	U14 (SKT)
		9405064	GAL18V10-20LP	LATTICE	20P DIP PKG	GAL-20NS	1.00	U10 (SKT)
		9406040	27C256		28P DIP PKG .6W	32 K BYTE EPROM	1.00	U7 (SKT)
		9407660	IDT7130LA70J	IDT	52P PLCC PKG	1K X 8 DUAL PORT RAM	1.00	U12 (SKT)
	BNC					BNC CABLE OPTION		
	BNC	2106001	31-317	AMPHENOL	50 OHM	BNC JACK, STRIGHT	4.00	EXT. CABLE PARTS
	BNC	2123015	DAE15P	CANNON	15 POS	'D' PLUG	1.00	EXT. CABLE PARTS
	XFM					TRANSFORMER OPTION		
	XFM	2105003	227161-7	AMP	50 OHM	ISOLATED RTANG PCMNT BNC RECPT	1.00	J3
	XFM	4701681	RC07GF681J	ALLEN BRADLEY	680 OHM, 1/4W	FIXED RESISTOR	1.00	R10 (OPT)
	XFM	5603002	956025	DATUM INC	08P DIP PKG	TRANSFORMER	1.00	T1 (OPT)
00						STANDARD ASSEMBLY		
00		2105002	413879-1	AMP	50 OHM	RTANG PCMNT BNC RECPT.	1.00	J3