COMPUTER SYSTEMS INC.

"QUICK & TIMELY"

S-100 CLOCK CALENDAR

INSTRUCTION MANUAL

INTRODUCTION

FEATURES

The ComputerWatch utilizes the popular MSM5832 real time Clock/Calendar chip designed for use in bus-oriented microprocessor applications. In case you are not aware of it the APPLE is a bus oriented microprocessor system. The 32.768 Hz crystal controlled time base will provide addressable 4-bit I/O data of SECONDS, MINUTES, HOURS,DAY OF WEEK, DATE, MONTH, YEAR. The data access is controlled by a 4-bit address, read, write, and hold inputs.

Features include:

- Time in Hours, Minutes, Seconds.
- Program selectable 24 hour military format or 12 hour AM/PM format.
- Date in Month, Day, Year, Day of Week, and Leap year recognition
- Fast time and date setting.
- +-30 second adjust.
- 4 hard interrupts. 1024 Hz (approx 1 millisecond) 1Hz, 1 minute, 1 hour.
- Crystal controlled time base.
- Latched input and output ports.
- On board battery backup power.
- Automatic power off sensing.
- Simple programming interface.

Applicational uses of the ComputerWatch are only limited to the imagination. Anywhere you need to know the time or date you can use the ComputerWatch. Applications would include, Time or date stamping of reports, checks, letters, file updates, calculating time intervals, recording measurements. I am sure you can think of many more uses for the ComputerWatch.

INSTALLTION

This section will show you how to install the ComputerWatch into your system.

Before installing the ComputerWatch in your system the address must be selected by setting the switches on SW1. After deciding which I/O port address will be used to access the board in your system, look up the address you have selected in Appendix B TABLE 5 and set the switches as shown. Inspect the board for shorts or damaged components. Insure that all the IC's are properly installed with pin 1 toward the left side of the board.

Carefully install the board in the system with power off, being certain that pins 1 and 50 line up with the ground pins on the S-100 bus connector. Plugging the board in backwards will cause damage to the board.

There is a Ni-Cad battery mounted on the circuit board that will keep the clock circuit running when power has been removed from the system. The battery will be charging whenever there is power on the system. The battery will keep the clock circuit running for months between chargings.

There is a trimmer capacitor located beside the clock chip U5. This capacitor is for fine tunning the clock time base circuit. If the clock gains or loses time, adjust this trimmer capacitor. You may have to make several adjustments over a period of time.

CHECKOUT

Appendic C contains a listing of a program that will check out the ComputerWatch. If you purchased or were provided with a diskette or a cassette tape then load the program from your media. If you do not have the program on some type of media then if you desire enter the program in Appendix B. The rest of this section deals with the operation of the Demo program. This program demonstrates the capabilities of the ComputerWatch, and allows you to check the operation of the board.

After you eather loaded or entered the program RUN it. The following will appear on the screen:

TYPE T TO DISPLAY TIME/DATE
TYPE R TO RESET THE TIME/DATE
TYPE A TO ADJUST THE TIME +-30 SECONDS
TYPE E TO END PROGRAM

TYPE T and then RETURN, from here on it will be assumed that you will press the RETURN key after each keyboard entry. The following display will be returned.

THE DATE IS M M / D D / Y Y
TODAY IS (Day of WeeK)
THE TIME IS H H : M M : S S AM/PM

The time and date returned from the ComputerWatch should be something reasonable to your time. If you are in a time zone other than Pacific time you will want to reset the time and possibly the date. Before attempting to reset the ComputerWatch TYPE T again and satisfy yourself that the ComputerWatch is working by comparing the second response with the first one. If everthing is as it should be then type R and see what happens.

YEAR TENS - TYPE the tens digit of the year or 8 for 80. The next response will ask for YEAR UNITS - You would respond by entering a Ø. Continue on suppling the requested information until you come to the question: DAY OF WEEK, Ø = SUN, 1 = MON, ETC: The day of week is kept in the ComputerWatch as seven numbers:

0 = SUNDAY
1 = MONDAY
2 = TUESDAY
3 = WEDNESDAY
4 = THURSDAY
5 = FRIDAY
6 = SATURDAY

TYPE the number that applies to today. Continue entering the requested information until you come to the question:
IS THIS A LEAP YEAR - Your response should be a "Y" for yes or a "N" for no. The next question is 12 OR 24 HOUR FORMAT - Answer with a 12 or 24. 12 hour format will act like a normal clock with AM and PM

notation. 24 hour format or military time will keep time up to 23:59:59 with 00:00:00 being midnight and 12:00:00 being noon time. There is no AM or PM notation with this format. The next question AM OR PM you answer with AM or PM depending on the time of day. If 24 hour time was selected this question would not have come up. The following screen will now be displayed:

```
THE DATE IS M M / D D / Y Y
TODAY IS ( Day of Week)
THE TIME IS H H : M M : S S AM/PM

TYPE T TO DISPLAY TIME/DATE
TYPE R TO RESET THE TIME/DATE
TYPE A TO ADJUST THE TIME +-30 SECONDS.
```

Now try the +-30 second adjust routine by typing A. The same screen as above will be returned, except that the minutes may have gained one minute and the seconds are reset to zero. If the seconds are greater than 30 then one would have been added to the minutes units digit and the seconds reset. If the seconds are less than 30 then only the seconds would be reset.

DATA DESCRIPTION	SECONDS		MINUTES		HOUR		DAY OF WEEK	DAY		MONTH		YEAR	
	UNITS	TENS	UNITS	TENS	UNITS	TENS		UNITS	TENS	UNITS	TENS	UNITS	TENS
REG	0	1	2	3	4	5	6	7	8	9	10	11	12
BITS 8	*		*		*			*		*		*	*
BITS 4	*	*	*	*	*		*	*		*		*	*
BITS 2	*	*	*	*	*	*	*	*	*	*		*	*
BITS 1	*	*	*	*	*	*	*	*	*	*	*	*	*

*ACTIVE BITS

FIGURE 1 PROGRAMMING CHART

The next thing you need to know is how to raise the HOLD signal. This signal is used to prevent the internal clock registers from rolling over during a read or write operation. Without this feature it could be possible to read a Units digit of 9 and before the Tens digit could be read the clock could count the Units register causing a rollover to the Tens register. The result would be nine units off. For example a 49 would read as a 59. The use of the HOLD signal will eliminate this undesirable condition. It is not however necessary to raise the HOLD signal to read clock data. In fact it might be undesirable in some cases to do so. You will see an example of this later on. You will need to know more information and it will be supplied as you proceed along. You will learn by example and dicussion of the example. Now type the following routine in BASIC:

```
NEW
010 C1 = 130 : C2 = 129
020 OUT C2,16
030 OUT C1,32
040 X = INP(C1)
050 PRINT X
060 OUT C2,0 : END
```

Run the program and see what happens. Run it a number of times. You will notice that the number printed is changing. This is because you have been reading the seconds units digit. Its value should change every second if the ComputerWatch is working properly. Lets examine the program and see why things happen as thay do.

The first instruction at line 010 sets the Input Port's I/O address into variable C1 and the Output Port's I/O address into variable C2. If you want to use other addresses then these two variables would have to be changed to the new address. Line 020 raises the HOLD signal by writting a 16 to the Output Port. Line 030 sets the address of the register you want to read into the Input Port. Why a 32? You must add 32 to the register address of the digit you want to read. In this case the Seconds Units Digit was wanted and it's address is 0. By adding 32 to 0 the actual value for the address would be 32. If for example you would want to read the Hour Tens Digit then you would add 5 to 32 for an actual address of 37. The chart in FIGURE 2 reflects this new information.

DATA DESCRIPTION	SECONDS		MINUTES		HOUR		DAY OF WEEK	DAY		MONTH		YEAR	
	UNITS	TENS	UNITS	TENS	UNITS	TENS		UNITS	TENS	UNITS	TENS	UNITS	TENS
REG	0	1	2	3	4	5	6	7	8	9	10	11	12
+32	32	33	34	35	36	37	38	39	40	41	42	43	44
BITS 8	•		•		•			•		•		•	•
BITS 4	•	•	•	•	•		•	•		•		•	•
BITS 2	•	•	•	•	•	•	•	•	•	•		•	•
BITS 1	•	•	•	•	•	•	•	•	•	•	•	•	•

*ACTIVE BITS

FIGURE 2 programming chart (expanded)

Line 040 will read the data from the selected register from the Input Port and store it in the variable X. Line 050 will lower the HOLD signal to allow the clock to continue counting by putting a 0 in the Output Port. NOTE - The HOLD signal should never be held up longer than one second. For a variation on the previous seconds routine type the following program:

```
010 C1 = 130
020 OUT C1,32
030 X = INP(C1)
040 IF Y=X GOTO 030
050 Y=X
060 PRINT X : GOTO 030
```

Notice in this program you did not raise the HOLD signal so it was not necessary to reference the Output Port (C2). The program only reads the Seconds Units Digit and is not concerned about digit rollover.

Now that you have mastered the reading of one digit lets try to read all of the time digits, Hours, Minutes, and Seconds. But first you must learn about the OPTION bits associated with time such as 12 or 24 format. If 12 hour format is selected then you have to be concerned with the AM/PM bits. Refer to FIGURE 3.

DATA DESCRIPTION		HOUR UNITS	HOUR TENS
REG		4	5
+32		36	37
BITS	8	•	24
	4	•	AM/PM
	2	•	•
	1	•	•

BIT 8 ON 24 HOUR FORMAT
BIT 8 OFF 12 HOUR FORMAT
BIT 4 ON PM
BIT 4 OFF AM

FIGURE 3 TIME OPTION BITS

The Hours Tens Digit contains the information to determine what format is being used, 12 or 24 Hour. It also contains the AM/PM information. Bit 4 is used to determine AM or PM and bit 8 is used for 12 or 24 Hour format.

The following program will read the time and display it as

H H : M M : S S AM/PM

Now type:

```
NEW
010 C1 = 130 : C2 = 129
020 DIM T(6)
030 OUT C2,16
040 LET I=Ø
050 FOR D = 37 TO 32 STEP -1
060 OUT C1,D
070 T(I) = INP(C1)
080 I=I+1
090 NEXT D
100 B=T(Ø)
110 IF B>7 THEN P$ = " "
120 IF T(Ø)>7 THEN T(Ø) = T(Ø)-8
130 IF B<8 and T(Ø)>3 THEN P$ = "PM"
140 IF B<8 and T(Ø)<4 THEN P$ = "AM"
150 IF T(Ø) >3 THEN T(Ø) = T(Ø)-4
160 PRINT "THE TIME IS ";T(Ø);T(1);":";T(2);T(3);":";T(4);T(5);P$
170 OUT C2,0  : END
```

Check and be sure you entered the program correctly and then RUN it. The time should be displayed on the screen as:

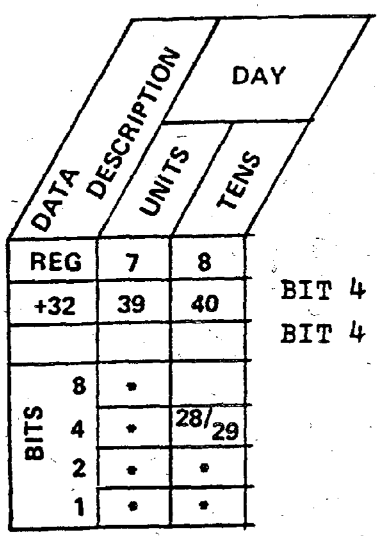
THE TIME IS M M : H H : S S AM/PM

The array T has been set up for six elements at line 020. The FOR loop between 050 and 090 allows the digit register address's to be stepped down each time a digit is read from the Input Port. The digit is stored in the array T and T is subscripted by I. After all clock time digits are read the IF statements between 110 and 150 are checking for 12 or 24 hour format and AM/PM. The PRINT statement will print the contents of array T and insert a : between the hours minutes and seconds. You may want to study is routine for awhile. You must strip the OPTION bits before using them, otherwise the value you print will be wrong, The code at line 120 and 150 does this for the 8 bit and the 4 bit. You may want to add some remarks to the program and save it for future use.

You know how to read the time now you can try the date. But first there are a few more facts you need to know. Refer back to FIGURE 2 and look at register 6 the Day of Week. The Day of Week is stored in this register as on of six digits. For example:

0 = SUNDAY, 1 = MONDAY, 2 = TUESDAY, 3 = WEDNESDAY, 4 = THURSDAY, 5 = FRIDAY, and 6 = SATURDAY.

The other thing you need to know about is how to handle leap year. Refer to FIGURE 4.



BIT 4 = Ø 28 DAYS IN FEB.
BIT 4 = 1 29 DAYS IN FEB.

FIGURE 4 PROGRAMMING CHART

Bit 4 of the Days Tens register control whether or not there are 28 or 29 days in the month of Febuary. If bit 4 is on then it is leap year and Febuary has 29 days. If bit 4 is off then there are 28 days in Febuary. NOTE - once Febuary 29th has been past the clock chip will automaticaly reset bit 4. FIGURE 5 has been updated with the information you have learned so far.

DATA DESCRIPTION	SECONDS		MINUTES		HOUR		DAY OF WEEK	DAY		MONTH		YEAR	
	UNITS	TENS	UNITS	TENS	UNITS	TENS		UNITS	TENS	UNITS	TENS	UNITS	TENS
REG	0	1	2	3	4	5	6	7	8	9	10	11	12
+32	32	33	34	35	36	37	38	39	40	41	42	43	44
BITS 8	•		•		•	24		•		•		•	•
BITS 4	•	•	•	•	•	AM/PM	•	•	28/29	•		•	•
BITS 2	•	•	•	•	•	•	•	•	•	•		•	•
BITS 1	•	•	•	•	•	•	•	•	•	•	•	•	•

*ACTIVE BITS

FIGURE 5 PROGRAMMING CHART (updated)

Enter the following program:

```
NEW
010 C1 = 130 : C2 = 129
020 DIM T(7) : DIM W$(7)
030 DATA "SUNDAY","MONDAY","TUESDAY","WEDNESDAY","THURSDAY",
        "FRIDAY","SATURDAY"
040 FOR A = Ø TO 6 STEP 1
050 READ W$(A)
060 NEXT A
070 OUT C2,16
080 LET I=Ø
090 FOR D = 44 to 38 STEP -1
100 OUT C1,D
110 T(I) = INP(C1)
120 I=I+1
130 NEXT D
140 IF T(4) >3 THEN T(4) = T(4)-4
150 PRINT "THE DATE IS ";T(2);T(3);"/";T(4);T(5);"/";T(Ø);T(1)
160 PRINT "TODAY IS";W$(T(6))
170 OUT C2,0   : END
```

Check the program to be sure all lines have been entered correctly. RUN the program. The display should look like this:

THE DATE IS
TODAY IS (DAY OF WEEK)

The basic difference between this program and the time is the addtion of the array W$ and the definition of the days of the week. Line 140 strips the 4 bit if it was set from the Days Tens Digit. If you dont strip this bit the digit that is read will be wrong.

An interesting exersise would be to combine the date and time routines. Why dont you try it. If you need help check the clock demo program listing in Appendix C.

SUMMARY

1. There are 13 clock registers.
2. All data stored in the clock is numeric.
3. Each digit of information has its own address.
4. The HOLD signal prevents counter rollover.
5. You add a 32 to the register address when doing a read.
6. You must lower the HOLD signal after reading to enable the registers to continue counting.
7. You have to look at the OPTION bits when reading the Hours Tens Digit.
8. You must strip the OPTION bit when reading the Hours Tens Digit.
9. The Day of Week is a value from Ø to 6.
10. The leap year bit is stored in the Day Tens 4 bit.
11. The leap year bit must be stripped when reading the Days Tens Digit.

CHAPTER 2

WRITE A SET ROUTINE

You will now learn to set or write data to the ComputerWatch. The following events must occur in the listed sequence.

1. Raise the HOLD signal by writing a 16 to the Data Output Port.
2. Write the register address to the Address Output Port.
3. Write the data plus 16 to the Data Output Port. The plus 16 is required to keep the HOLD signal active. For example, if you wanted to write a 3, the value sent to the Data Output Port would be 3 + 16 or 19.
4. Write the address plus 16 to the Address Output Port. This will raise the the WRITE signal.
5. Write the register address to the Address Output Port. This will lower the WRITE signal and complete the write operation. The setting of the register may be verified at this time by reading the data back.
6. Repeat steps 2, 3, 4, and 5 until all the desired registers are set.
7. Write a zero to the Data Output Port to lower the HOLD signal.

Type the following program. It is an example of a simple set routine.

```
NEW
010 C1 = 130 : C2 = 129
020 OUT C2,16
030 INPUT "ENTER REG. ADDRESS 0-12";N
040 INPUT "ENTER DATA 0-9";T
050 OUT C1,N
060 OUT C2,T+16
070 OUT C1,N+16
080 OUT C1,N
090 OUT C1,N+32
100 D = INP(C1)
110 PRINT D
120 OUT C2,0 : END
```

This program will allow you set a digit into any register and will then read the data back and display it. RUN the program and play with it by entering different register addresses and data. Try loading the current data and time and then use your read routine to read the data back to verify your settings.

If you feel ambitious try writting a set program similar to the clock demo program. If you have problems refer to the listing in Appendix C.

FIGURE 6 has been updated to reflect the data necessary to set the registers.

DATA DESCRIPTION	SECONDS		MINUTES		HOUR		DAY OF WEEK	DAY		MONTH		YEAR	
	UNITS	TENS	UNITS	TENS	UNITS	TENS		UNITS	TENS	UNITS	TENS	UNITS	TENS
REG	0	1	2	3	4	5	6	7	8	9	10	11	12
+32	32	33	34	35	36	37	38	39	40	41	42	43	44
+16	16	17	18	19	20	21	22	23	24	25	26	27	28
BITS 8	•		•		•	24		•		•		•	•
BITS 4	•	•	•	•	•	AM/PM	•	•	28/29	•		•	•
BITS 2	•	•	•	•	•	•	•	•	•	•		•	•
BITS 1	•	•	•	•	•	•	•	•	•	•	•	•	•

•ACTIVE BITS

FIGURE 6 PROGRAMMING CHART (updated)

ADJUST THE TIME

A feature of the ComputerWatch is an ability to adjust the time by +-30 seconds. if you find that the clock has gained or lost time you can adjust it with a +-30 second routine. This feature will advance the minutes units digit if the seconds are greater than 30 and reset the seconds to zero. If the seconds are less than 30 the minutes units digit will not be advanced and the seconds will be reset to zero.

To reset the time you must first raise the ADJUST signal. This signal must be active for 32 milliseconds. Then you must lower the ADJUST signal. Enter the following program:

```
NEW
010 C1 = 130  :  C2 = 129
020 OUT C2, 32
030 FOR A=1 TO 20 STEP 1
040 NEXT A
050 OUT C2,0  :  END
```

Line 020 will write a 32 to the Data Output Port. This will activate the +-30 second ADJUST signal. Line 030 and 040 provide a time delay of 32 milliseconds and line 050 will write a zero to the Data Output Port. This will lower the ADJUST signal and complete the operation. You will have a complete set routine if you add this code to the routines you have already created.

INTERRUPTS

There are four interrupts provided by jumper options. The read signal and the address of the timers is detected external to the clock chip and enables the hard interrupt signals. These signals can then interrupt the system CPU if an option jumper is activated. The hard interrupts are only enabled during the time that the timer signlas are being read from the clock chip, thus they can be controlled by software. The system CPU can input the timer signals when they are enabled at the clock chip. The four interrupts are:

1024 Hz (approx. 1 millisecond)
1 second
1 minute
1 hour

The 1024 Hz signal is a 50% duty cycle square wave and the 1 second, 1 minute, and 1 hour signals provide a low pulse for 122.1 microseconds at the indicated rate. The pulsed signals are normally high and go low momentarily (122.1us) when the indicated time interval has occurred. The following jumpers must be installed if you want an interrupt at:

	PAD		IRQ		NMI
1024 Hz	D	to	E	or	F
1 Second	C	to	E	or	F
1 Minute	B	to	E	or	F
1 Hour	A	to	E	or	F

To enable the timers you write a 47 to the Address Output Port. To disable the timers write an 0 to the Address Port.

SUMMARY

1. You have to raise the HOLD signal and write the address of the register to be written in the Data Output Port.
2. You have to write the data +16 to the Data Output Port. The 16 keeps the HOLD signal active.
3. You have to write the address +16 to the Address Output Port to raise the WRITE signal, and then write the address again to lower the WRITE signal.
4. You should read the data back for verification.
5. You should lower the HOLD signal by outputting a zero to the Data Output Port.
6. You adjust the time +-30 seconds by outputting a 32 to the Data Output Port. A time delay of 32 milliseconds is required before outputting a zero to the Data Output Port.

CHAPTER 3

FUNCTIONAL DESCRIPTION

The ComputerWatch is interfaced to the system by two TTL latching type output ports and one tristate input port. The ComputerWatch responds to four I/O port addresses in the system. The output port latches are selected by address bit zero or one. When the ComputerWatch is selected for access and an output cycle (write) is taking place with address bit zero active (high) the data port latches data from the system bus. When the ComputerWatch is selected and an output cycle is performed with address bit one active (high) the address port latches data from the system. The data port interfaces to the 4 bit bidirectional data bus of the clock chip by means of a tristate buffer. This buffer is disabled whenever a read of the clock chip is taking place, to allow the clock chip to drive its data onto the bus. The data port latches also interface two control signals to the clock chip. The 30 second +/- adjust signal and the hold signal are controlled by the two most significant bits of the data latches. The address latches provide a four bit address to the clock chip for selection of the desired clock register. The two most significant bits of the address latches control the signals read and write to the clock chip. The read signal is made active after presenting an address to the clock chip. The clock chip will then present the selected data from the clock on the 4 bit data bus. This data may then be read by the system on an input operation.

When data is to be written to the clock chip, the data is first written to the output latches and the address is written in the address latches. The write signal is then raised and lowered to complete the write to the clock chip.

Before reading or writing the clock chip the hold signal should be raised. When the read or write operation is completed the hold signal should be lowered to allow the clock to continue toggling.

The Interval timers are output by the clock when a read is performed to address 15 (hex F). The read signal and the address of the timers is detected external to the clock chip and enables the hard interrupt signals. These signals can then interrupt the system CPU if an optional hard interrupt is enabled (jumper installed). The hard interrupts are only enabled during the time that the timer signals are being read from the clock chip and thus can be controlled by software. The system CPU can input the timer signals when they are enabled at the clock chip. The 1024 HZ signal is a 50% duty cycle square wave and the 1 second, 1 minute, and 1 hour signals provide a low pulse for 122.1 microseconds at the indicated rate. The pulsed signals are normally high an go low momentarily (122.1 us) when the indicated time interval has occured.

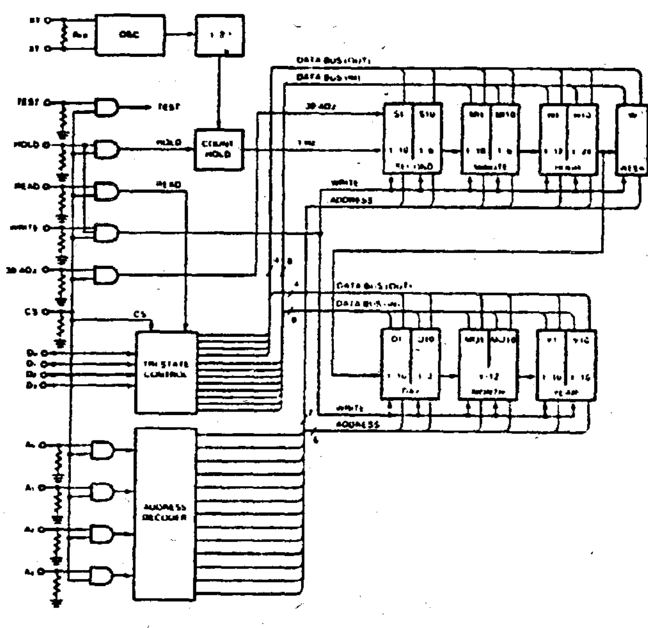
The 30 second +/- adjust signal must be held active (high) for greater than 31.25 milliseconds in order to be effective. When this signal is made active and then lowered the clock will be adjusted. The seconds digits are set to zero and the divider chain is reset so that the next toggle will occur 1 second after the 30 second +/- adjust signal is returned to the inactive state. The minutes are incremented by one if the seconds were equal to 30 or more when the adjust signal is raised. If the seconds were less than 30 at the time the adjust signal was raised then the minutes will not be changed.

The clock chip automatically detects power going off and switches to standby mode. The interface circuits of the clock chip are disabled in standby mode and the power drain is substantially reduced (30uA max.). The clock will continue keeping time while the on board battery supplies power. The clock will still operate properly when the battery voltage drops as low as 2.2 volts. The nominal battery voltage is 3.6 volts. The battery is charging whenever power is applied to the ComputerWatch. The time to fully charge a discharged battery is approximately 12 hours. The ComputerWatch may be used while charging a discharged battery. A fully charged battery can keep the clock running for two months or more.

The ComputerWatch will keep time properly even when removed from your system.

The clock chip contains registers for the time of day in hours, minutes, and seconds. The date is stored as day of week, day of month, month, and year. The clock chip registers are organized as 13 four bit registers. Refer to Table 1 for a layout of these registers. Option bits are contained in the hours ten's register. The two most significant bits of this register are used to select 12 or 24 hour format and indicate AM or PM in the 12 hour format. An optional bit is contained in the day of month ten's register for indicating when 29 days are needed in Febuary (leap year).

FUNCTIONAL BLOCK DIAGRAM



A_0 to A_3: Address Inputs
WRITE: Write Enable
READ: Read Enable
HOLD: Count Hold Enable
CS: Chip Select
D_0 to D_3: Data Input/Output
TEST: Test Input
±30 ADJ: ±30 Second Correction Input
XT & $\overline{XT}$: xtal oscillator connections
Vcc: +5 V Supply
GND: Ground

DISCRIPTION OF OPERATION

The ComputerWatch requires that events occur in a proper sequence and that proper timing is allowed for certain functions to occur. Following is a sequence of events and timing allowances that should be maintained when reading or setting the clock chip. Refer to Tables 1 through 4 for data formats and function codes. All data and address values given are decmal.

READ TIME AND DATE:

1. Raise hold signal by writing 16 to the data output port.
2. Wait 150 microseconds.
3. Activate read operation by outputting to the address port the address of the desired register plus 32. For example, the address of the hours tens digit from Table 1 is 5. The value output would be 32 plus 5 or 37.
4. Wait 6 microseconds
5. Input digit data. Note that the 4 most significant bits of the data are always zero as shown in Table 4. Certain registers have option bits encoded as part of the data. These bits must be examined for active options and masked from the data prior to using it for display purposes. Table 1 has a description of the option bits.
6. Repeat steps 3, 4, and5 untill all desired clock registers are read.
7. Write zero to the data output port to lower the hold signal
8. Write 47 to the address output port to enable the interval timers.

SET TIME AND DATE:

1. Raise hold signal by writing 16 to the data output port.
2. Wait 150 microseconds
3. Write address to the address output port.
4. Write data plus 16 to data output port. The plus 16 is required to keep the hold signal active. For example, if you wanted to output a 3 as data the value output to the port would be 3 plus 16 or 19.
5. Write address plus 16 to the address output port. This will raise the write signal.
6. Write address to the address output port. This will

lower the write signal and complete the write operation. The setting of the register may be verified at this time by reading the data back. Perform steps 3, 4, and 5 of the read time and date procedure to access the clock data.

7. Repeat steps 3, 4, 5, and 6 untill all the desired clock registers are set.
8. Write zero to the data output port to lower the hold signal.
9. Write 47 to the address output port to enable the interval timers.

ADJUST TIME +-30 SECONDS:

1. Write 32 to the data output port. This will activate the +-30 second adjust signal.
2. Wait 32 milliseconds.
3. Write zero to the data output port. This will lower the +-30 second adjust signal and complete the operation.

APPENDIX

APPENDIX A

DATA DESCRIPTION	SECONDS		MINUTES		HOUR		DAY OF WEEK	DAY		MONTH		YEAR	
	UNITS	TENS	UNITS	TENS	UNITS	TENS		UNITS	TENS	UNITS	TENS	UNITS	TENS
REG	0	1	2	3	4	5	6	7	8	9	10	11	12
+32	32	33	34	35	36	37	38	39	40	41	42	43	44
+16	16	17	18	19	20	21	22	23	24	25	26	27	28
BITS 8	*		*		*	24		*		*		*	*
BITS 4	*	*	*	*	*	AM/PM	*	*	28/29	*		*	*
BITS 2	*	*	*	*	*	*	*	*	*	*		*	*
BITS 1	*	*	*	*	*	*	*	*	*	*	*	*	*

*ACTIVE BITS

PROGRAMMING CHART

APPENDIX B

TABLES

Table 1 Clock register layout

Address	Data	Values	Options
0	Seconds units	0 to 9	
1	Seconds tens	0 to 5	
2	Minutes units	0 to 9	
3	Minutes tens	0 to 5	
4	Hours units	0 to 9	
5	Hours tens	0 & 1	AM and 12 hour format (D2=0 and D3=0)
"	" "	4 & 5	PM and 12 hour format (D2=1 and D3=0)
"	" "	8 to 10	24 hour format
6	Day of week	0 to 6	*1
7	Day of month units	0 to 9	
8	Day of month tens	0 to 3	28 days in Feb. (D2=0)
"	" " " "	4 to 7	29 days in Feb. (D2=1)
9	Month units	0 to 9	
10	Month tens	0 & 1	
11	Year units	0 to 9	
12	Year tens	0 to 9	
13	NOT USED	-	
14	NOT USED	-	
15	Interval timers *2	0 to 15	D0=1024 HZ
"	" "	"	D1=1 HZ (1 second)
"	" "	"	D2=1/60 HZ (1 minute)
"	" "	"	D3=1/3600 HZ (1 hour)

*1 0 = Sunday, 1 = Monday, 2 = Tuesday, 3 = WEDNESDAY
4 = Thursday, 5 = Friday, 6 = Saturday.

*2 The interval timers are only active for read operations with hold inactive. All other clock data can be read or written with hold active.

Table 2 Address output port codes

Address port data	Function selected
*1 Address + 0	Setup and hold address for write operation
Address + 16	Generate write signal to clock chip *2
Address + 32	Read data from clock chip to 4 bit data bus
47	Enable interval timers (hold must be inactive

*1 The address portion of the data is obtained from Table 1.

*2 The data from the lower 4 bits of the data output port is provided to the clock chip on the 4 bit data bus. When the write signal transitions from active to inactive the data will be written in the selected clock register. The internal timers cannot be written.

NOTE-The address port is selected when system address bit 1 is active during a system output cycle to the ComputerWatch.

Table 3 Data output port codes

Data port data		Function selected
0		Idle (timers enabled if address port * 47)
Data + 16	*1	Raise hold signal to prepare for read or write
Data + 32	*2	Raise +-30 second adjust signal

*1 This signal must be held active for 150 microseconds before performing a read or write operation.

*2 This signal must be held active for at least 32 milliseconds to function properly.

Note-The data port is selected when system address bit 0 is active during a system output cycle to the ComputerWatch. The data is used as input to the clock chip for write operations. Only the least significant bits are used.

Table 4 Data word layouts

Data to address output latch:

D7 - Not used
D6 - Not used
D5 - Read signal
D4 - Write signal
D3 - Address Bit 3
D2 - Address Bit 2
D1 - Address Bit 1
D0 - Address Bit 0

Data to data output latch:

D7
D6
D5
D4
D3
D2
D1
D0

Table 4 Data Word Formats

ADDRESS OUTPUT LATCH		DATA OUTPUT LATCH	
Data bits	Function	Data bits	Function
D7	Not used	D7	Not used
D6	Not used	D6	Not used
D5	Read signal	D5	+-30 second adjust
D4	Write signal	D4	Hold signal
D3	Address bit 3	D3	Data bit 3 *
D2	Address bit 2	D2	Data bit 2 *
D1	Address bit 1	D1	Data bit 1 *
D0	Address bit 0	D0	Data bit 0 *

* This data is written to the clock during a write operation.

Table 4 Data Input Port

Data bits	Function
D7	Always 0
D6	Always 0
D5	Always 0
D4	Always 0
D3	Data bit 3 *
D2	Data bit 2 *
D1	Data bit 1 *
D0	Data bit 0 *

* The clock data is not stable until 6 microseconds after the address and read signal are stable.

TABLE 5 I/O PORT ADDRESS SELECT

DECIMAL ADDRESS RANGE	SW1 A7 1-16	SW2 A6 2-15	SW3 A5 2-14	SW4 A4 3-13	SW5 A3 5-12	SW6 A2 6-11
0-3	X	X	X	X	X	X
4-7	X	X	X	X	X	
8-11	X	X	X	X		X
12-15	X	X	X	X		
16-19	X	X	X		X	X
20-23	X	X	X		X	
24-27	X	X	X			X
28-31	X	X	X			
32-35	X	X		X	X	X
36-39	X	X		X	X	
40-43	X	X		X		X
44-47	X	X		X		
48-51	X	X			X	X
52-55	X	X			X	
56-59	X	X				X
60-63	X	X				
64-67	X		X	X	X	X
68-71	X		X	X	X	
72-75	X		X	X		X
76-79	X	X	X			
80-83	X		X		X	X
84-87	X		X		X	
88-91	X		X			X
92-95	X		X			
96-99	X			X	X	X
100-103	X			X	X	
104-107	X			X		X
108-111	X			X		
112-115	X				X	X
116-119	X				X	
120-123	X					X
124-127	X					

X = Jumper installed switch on.

TABLE 5 cont. I/O PORT ADDRESS SELECT

DECIMAL ADDRESS RANGE	SW1 A7 1-16	SW2 A6 2-15	SW3 A5 3-14	SW4 A4 4-13	SW5 A3 5-12	SW6 A2 6-11
128-131		X	X	X	X	X
132-135		X	X	X	X	
136-139		X	X	X		X
140-143		X	X	X		
144-147		X	X		X	X
148-151		X	X		X	
152-155		X	X			X
156-159		X	X			
160-163		X		X	X	X
164-167		X		X	X	
168-171		X		X		X
172-175		X		X		
176-179		X			X	X
180-183		X			X	
184-187		X				X
188-191		X				
192-195			X	X	X	X
196-199			X	X	X	
200-203			X	X		X
204-207			X	X		
208-211			X		X	X
212-215			X		X	
216-219			X			X
220-223			X			
224-227				X	X	X
228-231				X	X	
232-235				X		X
236-239				X		
240-243					X	X
244-247					X	
248-251						X
252-255						

X = Jumper installed or switch on.

APPENDEX C

CLOCK DEMO PROGRAM
LISTING

```
010 REM *************************
020 REM *THIS PROGRAM WILL READ *
030 REM *AND DISPLAY TIME/DATE, *
040 REM *RESET TIME/DATE, AND   *
050 REM *ADJUST TIME +-30 SEC.  *
060 REM *************************
070 C1 = 130 : C2 = 129 : DIM T(13)
080 DIM W$(7) : DIM S$(13)
090 DATA "SUNDAY","MONDAY","TYESDAY","WEDNESDAY","THURSDAY",
         "FRIDAY","SATURDAY"
100 REM
110 DATA "YEAR TENS","YEAR UNITS","MONTH TENS","MONTH UNITS",
         "DAY TENS"
120 DATA "DAY UNITS","DAY OF WEEK 0 = SUN, 1 = MON, ETC"
130 DATA "HOUR TENS","HOUR UNITS","MINUTE TENS","MINUTE UNITS"
140 DATA "SECOND  TENS","SECOND UNITS"
150 REM
160 REM *************************
170 REM * LOAD ARRAY W$         *
180 REM *************************
190 REM
200 FOR A = 0 TO 6 STEP 1
210 READ W$(A)
220 NEXT A
230 REM
240 REM *************************
250 REM * LOAD ARRAY S$         *
260 REM *************************
270 REM
280 FOR A = 12 TO 0 STEP -1
290 READ S$(A)
300 NEXT A
```

```
310 PRINT "TYPE T TO DISPLAY DATE/TIME"
320 PRINT "TYPE R TO RESET DATE/TIME"
330 PRINT "TYPE A TO ADJUST THE TIME +-30 SECONDS"
340 INPUT "TYPE E TO END PROGRAM";I$ : PRINT
350 REM
360 REM *************************
370 REM * SELECT CORRECT ROUTINE*
380 REM *************************
390 REM
400 IF I$ = "T" THEN 500
410 IF I$ = "R" THEN 790
420 IF I$ = "A" THEN 1080
430 IF I$ = "E" THEN 1130
440 GOTO 310
450 REM
460 REM *************************
470 REM * TIME/DATE DISPLAY     *
480 REM *************************
490 REM
500 OUT C2,16  : LET I = 0
510 FOR D = 44 TO 32 STEP -1
520 OUT C1,D  : T(I) = INP(C1)  : I = I+1
530 NEXT D
540 REM
550 REM **************************
560 REM * CHECK FOR 12/24 & AM/PM*
570 REM **************************
580 REM
590 IF T(4) > 3 THEN T(4) = T(4)-4
600 B = T(7) : IF B > 7 THEN P$ = " "
610 IF T(7) >7 THEN T(7) = T(7) -8
620 IF B <8 AND T(7) >3 THEN P$ = "PM"
630 IF B <8 AND T(7) <4 THEN P$ = "AM"
640 IF T(7) >3 THEN T(7) = T(7)-4
650 REM
660 REM **************************
670 REM * PRINT ROUTINE          *
680 REM **************************
```

```
690 REM
700 PRINT "THE DATE IS ";T(2);T(3);"/";T(4);T(5);"/";T(0);T(1)
710 PRINT "TODAY IS ";W$(T(6))
720 PRINT "THE TIME IS ";T(7);T(8);":";T(9);T(10);":";T(11);T(12):P$
730 PRINT :  OUT C2,0 : GOTO 310
740 REM
750 REM **************************
760 REM *TIME/DATE SET ROUTINE   *
770 REM **************************
780 REM
790 OUT C2,16  : D = 12
800 FOR D = 12 TO 0 STEP -1
810 PRINT S$(D) : INPUT N : T(D) = N
820 NEXT D
830 INPUT "IS THIS A LEAP YEAR";Y$
840 IF Y$ = "Y" THEN T(8) = T(8)+4
850 IF Y$ = "Y" GOTO 880
860 IF Y$ = "N" GOTO 880
870 GOTO 830
880 INPUT "12 OR 24 HOUR FORMAT";Y
890 IF Y = 24 THEN T(5) = T(5)+8
900 IF Y = 12 GOTO 930
910 IF Y = 24 GOTO 980
920 GOTO 880
930 INPUT "AM OR PM";X$
940 IF X$ = "PM" THEN T(5) = T(5)+4
950 IF X$ = "PM" THEN GOTO 980
960 IF X$ = "AM" THEN GOTO 980
970 GOTO 930
980 FOR D = 12 TO 0 STEP -1
990 N = T(D) : GOSUB 1020
1000 NEXT D
1010 OUT C2,0   : PRINT : GOTO 500
1020 OUT C1,D   :  OUT C2,N+16 : OUT  C1,D+16 : OUT  C1,N : RETURN
1030 REM
1040 REM **************************
1050 REM * ADJUST ROUTINE         *
1060 REM **************************
1070 REM
```

```
1080 OUT C2,32
1090 FOR A = 1 TO 20 STEP 1
1100 NEXT A
1120 OUT C2,0  : GOTO 500
1130 END
```

S-100 CLOCK CALENDAR

BOARD ASSEMBLY INSTRUCTIONS

This KIT is intended for those people who have had some prior experience wit kit building and digital electronics. If you do not fall into this category it is recommended that you find an experienced person to help you with the assembly and checkout of the board.

Although there is nothing sacred in the suggested steps that follow, if you will follow them step-by-step, you will be able to complete your task easier. The time required to complete the assembly should be less than forty five minutes. It will help to place a check mark beside each step as you complete it.

____ 1. Make sure you have the tools you will need to assemble this kit. For this board you will need the following: a soldering iron (20 watts maximum). Rosin core solder, preferably 63/37. diagonal cutters, a small magnifying glass, a screwdriver, and a lead former or a pair of needle-nose pliers.

____ 2. Check the parts received against the parts list. Take special care to correctly identify look-alike parts, i.e., resistors, capacitors, regulators, sockets and IC's. If anything is missing from your kit please call COMPUTIME and report the shortages immediately.

____ 3. Inspect the PC board for shorts by holding it up to a strong light.

****** CAUTION ******

USE EYE PROTECTION WHILE SOLDERING OR CUTTING WIRE

NOTE - When installing sockets DO NOT solder in place untill told to do so.

____ 4. Install 18 pin socket at U5.

____ 5. Install 16 pin sockets at U1, U4, U7, U8, U10, U11.

____ 6. Install 14 pin sockets at U2, U3, U6, U9.

____ 7. A handy trick to help you construct your board is to insert all the above sockets into the board first, then place a flat piece of styrofoam firmly against the top of the board. Turn it over, holding the styrofoam piece tightly against the board. The IC sockets should not be on the bottom of the PC board. You should have installed them on the silk screened side. Press the board down, forcing the sockets into the styrofoam. Now solder alternating corner pins of the IC sockets to hold them in place. Now turn the board over and carefully inspect it to determine that all IC sockets are down flat against the board. If you find any that are not down flat, melt the solder joints at the corners of the IC socket and press it down against the board. When you have determined that all IC sockets are down flat on the board, turn the board back over and solder all the pins. Make sure that all the pins are sticking through the board. IC sockets are very difficult to remove once they are soldered onto the board.

____ 8. Install the 10K SIP resistor networks at RN1 and RN2. These are the ones marked 08-1-103. Solder in place.

____ 9. Install the 1K SIP resistor network at RN3. It is the one marked 08-1-102. Solder in place.

____ 10. Install the glass diodes at CR1 and CR2. Be sure the black bands face the bar of the diode symbol.

____ 11. Install the 100 OHM $\frac{1}{4}$w resistor (brown, black, brown) at R1, and the 220 OHM $\frac{1}{4}$w resistor (red, red, brown) at R2.

____ 12. Install the 4.7uf capacitor at C3. Be sure the positive side of the capacitor faces the + symbol.

____ 13. Install the 1.5uf tantalum capacitor at C4. Be sure to observe the proper polarity.

____ 14. Install the 20pf capacitor at C2. Now solder the diodes, capacitors and resistors.

____ 15. Install the .1uf capacitors at C5, C6, and C7 and solder.

____ 16. Install the heat sink and the 7805 regulator at VR1 using the screw and nut provided. Solder in place.

____ 17. Install the crystall at Y1 and solder in place.

____ 18. Install the trimmer capacitor at C1 and solder in place.

____ 19. Install the 6 position DIP switch at SW1 with the number 1 switch facing the left side of the board.

____ 20. Install the 3.6 volt battery at B1 and solder in place.

____ 21. Examine all solder joints for good clean connections.

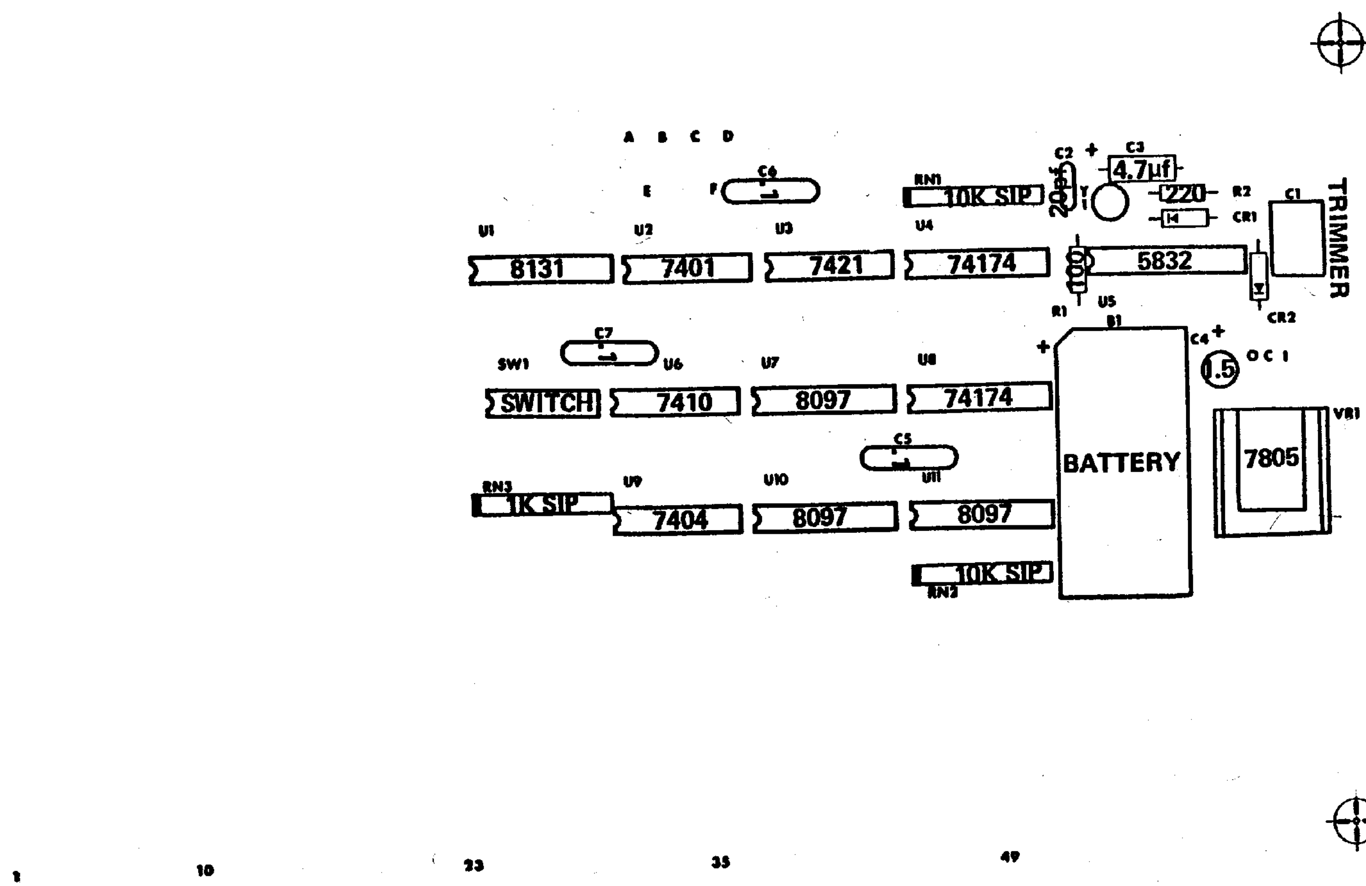
____ 22. Install IC's per location chart. Be carefull not to bend the pins under when inserting into the sockets.

____ 23. Refer to the user manual for programming instructions.

NOTE - The battery will take about eight hours to reach full charge.

PARTS LIST

	LOCATION	PART	DESCRIPTION
___	U1	8131	Hex Buffer
___	U2	7401	QUAD Two Input Gate
___	U3	7421	Dual 4-IN And Gate
___	U4,U8	74174	Hex Flip-Flop
___	U5	MSM5832	Clock/Calendar
___	U6	7410	Triple Three Input Gate
___	U7,U10,U11	8T97/74367	Tristate Hex Buffer
___	U9	7404	Hex Inverter
___	RN1,RN2	10K SIP	Resistor Network 08-1-103
___	RN3	1K SIP	Resistor Network 08-1-102
___	R1	100 OHM	Resistor ¼w
___	R2	220 OHM	Resistor ¼w
___	Y1	CRYSTAL	Crystal 32,768 HZ
___	CR1,CR2	DIODE	Diode glass
___	SW1	SWITCH	Six Position Dip Switch
___	C1	TRIMMER	Trimmer Capacitor 9-35pf JFD TYPE
___	C2	20pf	Capacitor
___	C3	4.7uf	Capacitor Electrlitic
___	C4	1.5uf	Capacitor Tantalum 25 vdc
___	C5,C6,C7	.1uf	Capacitor
___	B1	BATTERY	Battery 3.6 volts GE. NO. DS3SD 3.6V
___	VR1	7805	Regulator +5 volts
___	VR1	HEAT SINK	Heat Sink
___	VR1	HARDWARE	Nut and Screw
___	4	SOCKETS	14 Pin Sockets
___	6	SOCKETS	16 Pin Sockets
___	1	SOCKET	18 Pin Socket
___	1	PC BOARD	Clock/Calendar Circuit Board
___	1	MANUAL	User Manual
___	1	MANUAL	Assembly Instructions



SIZE	CODE IDENT NO	DRAWING NO
C		S100-880
SCALE 2X		SHEET 4 OF 4