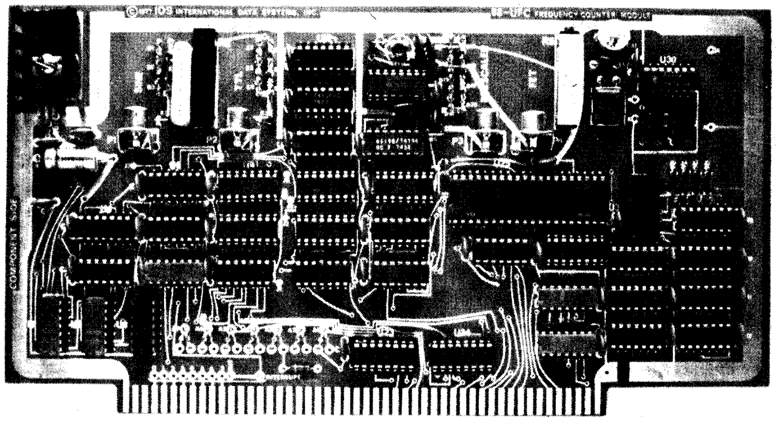
88-UFC UNIVERSAL FREQUENCY COUNTER MODULE

Product Description



The 88-UFC is a universal 9-decade frequency and interval counter. The 88-UFC is completely contained on one pluggable printed circuit board which is compatible with ALTAIR/IMSAI and S100 bus computers. All features and functions are software controlled making remote programming and reading possible for process control applications if the computer is equipped for data communications (such as 88-MODEM).

Features of the counter and timer include:

- Measure frequency from DC to 500 MHz and higher
- Contains built-in prescaler
- Contains four computer selected input channels
- Time base range is 1 second to 100 nanosec.
- Extremely high stability temperature compensated crystal oscillator (TCXO) option available

Function and mode selections under software control include:

- Interval - timebase select
- Port select (1 of 4)
- Counter/Period select
- Interrupt enable for Time/Period
- Stop/Run counter

BASIC Software is provided to compute:

- Totalizing/Accumulation
- Period
- Period averaging
- Time interval
- Frequency
- Frequency ratio
- Fixed or variable frequency offset
- Data storage for later retrieval and/or printout
- ASCII, Binary, or BCD conversion

IDS **INTERNATIONAL DATA SYSTEMS, INC.**

400 North Washington Street, Suite 200, Falls Church, Virginia 22046 U.S.A.

Telephone (703) 536-7373

88-UFC UNIVERSAL FREQUENCY COUNTER MODULE

International Data Systems, Inc.
400 North Washington Street
Suite 200
Falls Church, VA 22046 USA
Telephone (703) 536-7373

SECTION I - THEORY OF OPERATION

1. INTRODUCTION

An electronic digital counter is one of several items that are practically essential in even the most basic analog and/or digit test and experimentation capability. Also, in addition to its use as an item of general purpose test equipment, the digital counter is indispensable in many sensing and control applications.

In the past, digital counters were extremely expensive, and were not generally within the reach of the average hobbyist, experimenter, or ham radio operator. During the past few years, relatively inexpensive digital counters have appeared on the electronics market. However, these devices are extremely simple, normally providing only the capability for frequency measurements. Presently, several test instruments are available that combine the capabilities of a digital counter with the flexibility and computational power of a digital processor. The capabilities of the resulting device are extremely impressive. However, prices generally lie between $2000 and $6,000.

The intention in designing the 88-UFC Frequency Counter Module was to provide a device that, when combined with an appropriate digital processor, would provide a measurement, computation, and display capability competetive with instruments costing several thousands of dollars, at a price well within reach of any experimenter, hobbyist, or ham radio operator owning an S-100 bus computer. This intention has been fulfilled beyond our expectations.

Coupled with the control, computational, and display powers of an S-100 bus computer, the 88-UFC is the most powerful measurement device in its price range.

THEORY OR OPERATION

A digital counter is basically a very simple device. It is, in essence, a comparitor. It compares an unknown frequency or time period to a known frequency or a known time period, depending upon whether frequency or period measurements are being made. The results of the comparisons are presented in an easy-to-read digital format.

A digital counter consists of four basic elements:

1. Time-base generator
2. Decimal counting units with display
3. An electronic switch (gate)
4. Signal input and wave shaping circuitry

The time-base generator generally consists of a crystal oscillator operating either at 1 MHz or 10 MHz, followed by a multi-stage divider (some inexpensive counters use the 60 Hz AC power line frequency as the time base). A high-frequency crystal oscillator is used so that a very accurate time base can be generated. The multi-stage divider reduces the oscillator frequency to one Hertz (one cycle per second).

The decimal counting units (DCU's) are the counting portion of a digital counter. Each DCU counts from zero to nine, resets to zero, and starts over again. As a counting unit goes from nine back to zero, it generates an output signal that is suitable for driving another DCU. Therefore, DCU's can be strung together to allow counting to any desired value. Because the count value must generally be read by humans, each DCU is connected to a digital display that shows the value that the DCU has counted to.

The electronic switch (or gate) is the element that allows the signal being counted to pass into the DCU's or to be disconnected from them. Because the switch must operate very fast (so that high-frequency signals can be measured), it is notmally constructed from a digital logic gate that has turn-on and turn-off times of a few nanoseconds (a few billionths of a second).

The signal input and wave shaping circuitry is required for several reasons. First, it is desirable that the device or circuit generating the signal to be measured be disturbed as little as possible. This requirement is generally satisfied by constructing the counter so that it requires very little signal current to operate. This is another way of saying that the counter has a high input impedance. Second, the signal to be measured may be very small and require amplification before it can cause the DCU's to operate. Third, the DCU's are digital devices that will not operate properly with a sine wave, or other signal that changes amplitudes relatively slowly, as an input. Therefore, a major function of the signal input and wave shaping circuitry is to cause all signal transitions to occur in a few nanoseconds at the input to the DCU's.

There is an additional counter element that was not included in the list above. The element in question controls counter operation and is, in some ways, the most complex portion of the entire device. However, it is not necessary to consider exactly how the counter is controlled to understand the basic modes of operation. In addition, the 88-UFC does not have a control section. Instead, control is accomplished by the S-100 bus computer and its associated software. This method of implementing the counter control function is directly responsible for the great power and flexibility of the 88-UFC. That is, the operation of the counter control section can be changed at will, simply by changing the computer program that drives the counter.

The block diagram below shows the basic configuration of a counter that is set up to measure frequency. The DCU's would first be reset to all zeroes by the control section (not shown). The switch would then be closed under control of the 1 Hz divider output signal. During the one second the switch is closed, the DCU's would count all high-to-low or low-to-high transitions of the input signal. The time between two such transitions is one cycle of the input signal. Therefore, when the switch opened, the DCU's would have recorded the number of cycles that the input signal went through in one second. This is, by definition, the frequency of the input signal in Hertz (Hz).

There are several other modes in which digital counters operate, but two of the most common are "period" and "count". The two block diagrams below show how the counter elements are interconnected for these two modes.

The objective in measuring period is to determine how much time elapses during one cycle of the input signal. Dividing the period (in seconds) into 1, gives the frequency of the input signal. As shown by the diagram, when measuring period, the functions of the oscillator/divider chain and the input and wave shaping circuitry are interchanged. Also, there is one additional difference: the signal from the divider chain is a relatively high frequency such as 1 MHz. Since, at 1 MHz, one cycle represents one microsecond, the period of the input signal is measured in microseconds. It should be pointed out

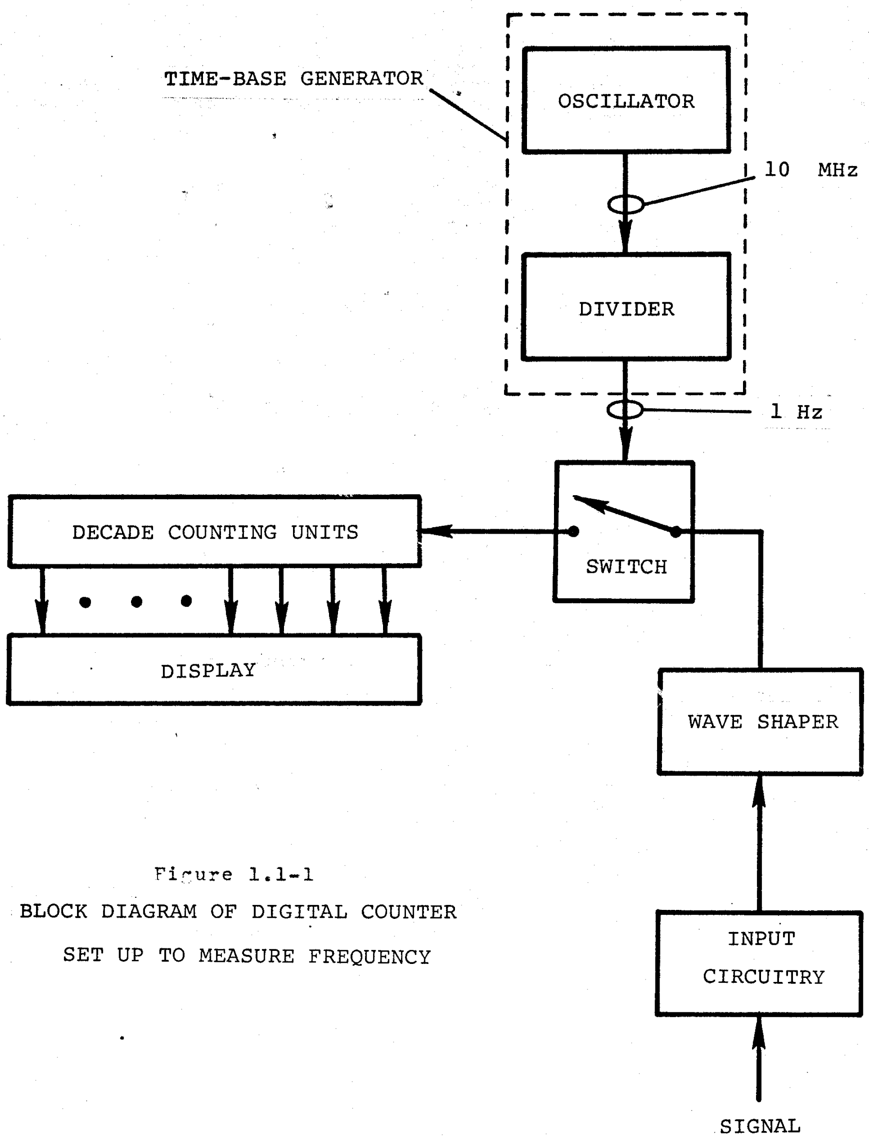


Figure 1.1-1
BLOCK DIAGRAM OF DIGITAL COUNTER
SET UP TO MEASURE FREQUENCY

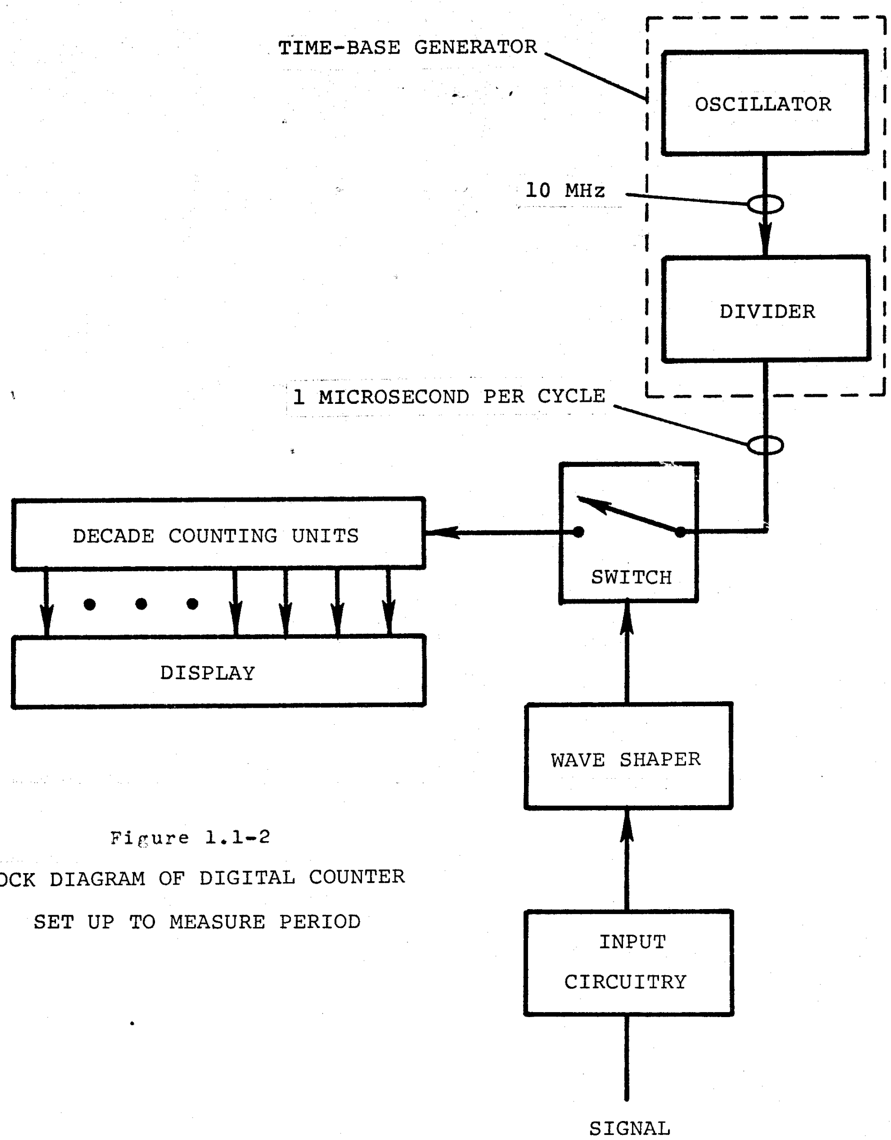


Figure 1.1-2

BLOCK DIAGRAM OF DIGITAL COUNTER

SET UP TO MEASURE PERIOD

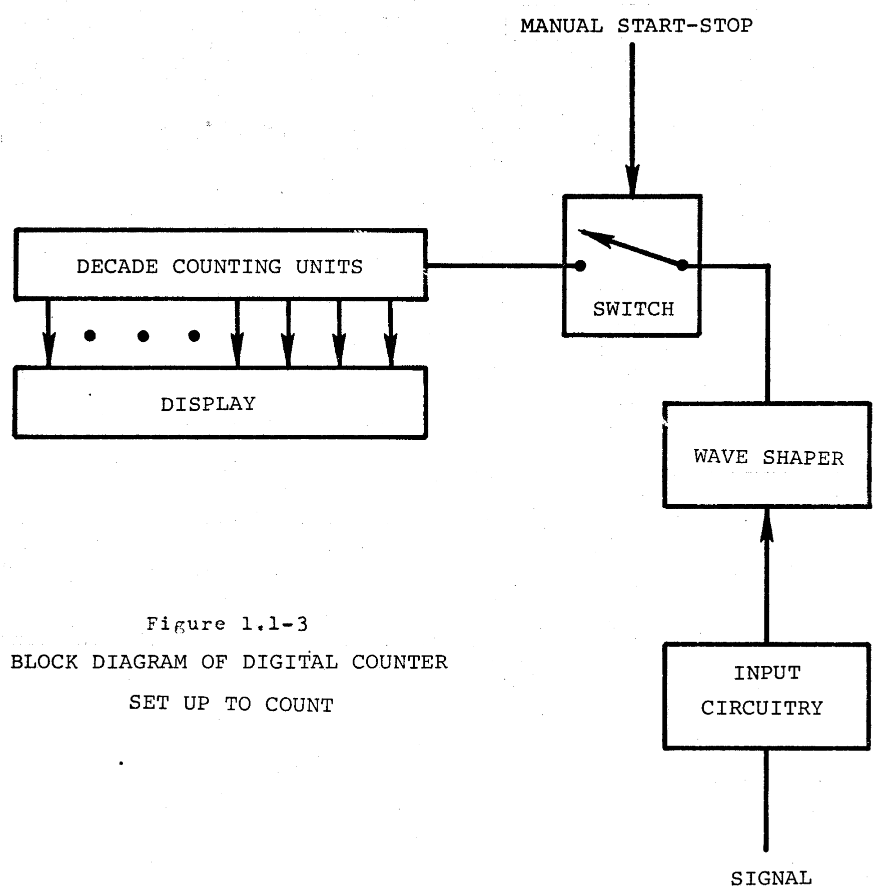
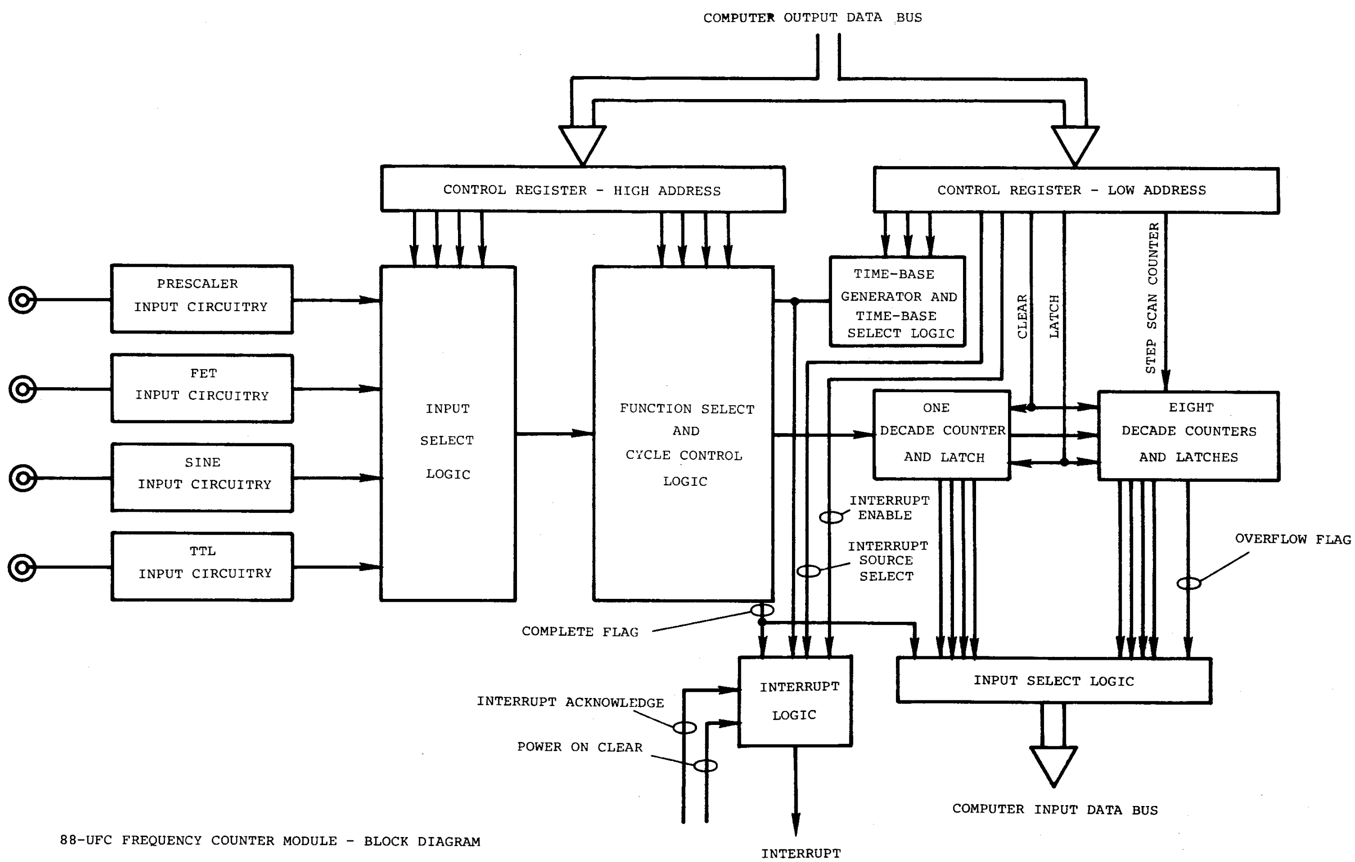


Figure 1.1-3
BLOCK DIAGRAM OF DIGITAL COUNTER
SET UP TO COUNT



88-UFC FREQUENCY COUNTER MODULE - BLOCK DIAGRAM that provisions are generally made in the time base generator (oscillator and divider chain) to obtain all frequencies from the undivided output of the oscillator to 1 Hz, in powers of 10.

In measuring period, the input signal causes the electronic switch to close for one input signal cycle. During this time, the output of the divider chain is counted by the DCU's. If as suggested earlier, the divider chain provided a 1 MHz signal (1 microsecond per cycle), and at completion of the measurement the DCU's registered 100, this would mean that the period of the input signal is 100 microseconds.

As shown by the block diagram, in the "count" mode, the time base generator is not used. The switch is manually opened and closed by the operator.

Now that we have considered the basics of digital counter operation, we can look at how the 88-UFC operates.

The 88-UFC possesses the majority of the desirable features of the most expensive counters presently available. On the surface this does not appear possible, considering the price of the 88-UFC. However, as implied above (the absence of a control section), the 88-UFC is considerably simpler than most high-quality counters. The simplicity results from the fact that many of the counter functions that normally must be implemented in hardware, are software controlled in the 88-UFC. A good example is the ability to measure tha ratio of two input frequencies. In most counters, the lower frequency signal replaces the time base and cycles of the higher frequency signal are counted during one cycle of the lower frequency signal. This of course requir3s considerable switching logic if accomplished in hardware. Under control of an S-100 bus computer, the 88-UFC accomplishes the same function by simply taking two separate frequency or period measurements (one for each signal), and computes the desired answer.

Another reason that the 88-UFC provides so much capability, in such a small package, at such a low price, is that a single MOS-LSI integrated circuit has been used to replace eight counters, eight latches, and a considerable amount of control logic. A basic block diagram of the 88-UFC is shown below.

The 88-UFC has four independent inputs: a divide by 10 prescaler, FET, Sine, and TTL.

The prescaller input accomodates input frequencies up to 600 MHz, has a sensitivity of better than 150 millivolts, and terminates the source in approximately 50 ohms. The other three inputs operate in the range up to approximately 65 MHz. The FET circuit has an input impedance in excess of 100 thousand ohms at low frequencies, and also has a sensitivity on the order of 150 millivolts. Both the Sine and TTL inputs are each coupled through a Schmitt trigger, and can therefore accomodate slowly varying signals without difficulty. The Sine input is AC couplet and biased so that it converts sine waves into symmetrical square waves, to conserve period information. The TTL input is DC coupled and requires an input signal in the TTL range (approximately 1.5 volts). All four inputs are diode protected, and the Sine and TTL inputs include a 47 ohm series limiting resistor.

The 88-UFC input circuitry has been kept very simple. As a result, counting is very reliable and requires no adjustments (except for initial adjustment of the FET and Sine inputs).

The 88-UFC time-base generator consists of a 10 MHz crystal oscillator, a seven decade divider chain, and an 8-input multiplexer. The capability is provided to select any one of eight time-base rates, under computer control, from 100 nanoseconds to 1 second, in decade steps.

The function select logic allows selection, under computer control, of three basic functions: frequency, period, and count. In the period mode, either a whole cycle or the negative half cycle can be selected for measurement (taking both measurements and subtracting yields the period of the positive half cycle).

The 88-UFC provides nine output digits. The first(least significant) digit comes from a single high-speed (Schottky) counter. The remaining eight digits come from an MOS-LSI integrated circuit that includes counters, latches, overflow detect, scan counter, scan multiplexer, and control logic. All nine decades are latched and read under computer control. Therefore, the contents of the counter can be examined by the computer "on-the-fly". This capability is essential, for example, when implementing a preset mode of operation under software control. All counter functions are under direct computer control, including digit counter and scan counter reset, digit scan, and loading of the latches.

The 88-UFC also includes a flexible interrupt capability. The interrupt function if enabled and disabled under software control, and either "measurement complete" or "countdown chain output" can be selected as the interrupt source, also under software control.

The 88-UFC includes a two bit status word that can be read under software control. Bit Ø is a flag that is low while a measurement is in progress, and which goes high when the measurement is completed. Bit 1 goes high if the counter overflows (if the count exceeds 999,999,999).

The 88-UFC requires two computer addresses for both input and output (the nine digits of the reading are multiplexed into one address with multiple reads, so conserve addresses). The two input and two output functions are described in detail below.

COMPUTER INTERFACE DATA FORMATS

As pointed out above, the 88-UFC uses two computer addresses in both the input (to the processor) and output directions. The two 8-bit output words control counter operation. The two 8-bit input words (not all bits are used) allow the reading of counter status and nine digits of measurement results. The input and output word formats are defined below.

DATA OUT - LOW ADDRESS

D7	D6	D5	D4	D3	D2	D1	D0
INTERRUPT ENABLE	STEP SCAN COUNTER	LOAD LATCHES	RESET	INTERRUPT SOURCE SELECT	TIME-BASE SELECT		

Bits DØ, D1, and D2 select the desired tap on the time-base generator divider chain. These three bits specify the period of the selected divider tap in <u>negative</u> powers of 10. That is, $\text{ØØØ}_2 = \text{Ø}_{10}$ selects a divider output period of 1 second, or 10^0. $\text{ØØ1}_2 = 1_{10}$ selects a period of 1/10 second or 10^{-1}, etc.

Bit D3 determines the interrupt source, when the interrupt system is enabled (bit D7 = "1"). A "0" in bit position D3 selects "measurement completion" as the interrupt source. A "1" selects the countdown chain as the interrupt source. In this case, the interrupt rate is determined by the divider chain tap selected by bits DØ, D1, and D2.

Bit D4 resets the nine digit counter chain and the scan counter used to multiplex the nine counter digits into a single computer address.

The counter chain is reset to all zeros and the scan counter is reset to the most significant digit. The reset function requires a pulse that is at least 4 microseconds long. Therefore, bit D4 must be loaded with a "1" then, a minimum of 4 microseconds later, a "0".

Bit D5 loads the contents of the nine counter stages into nine latches so that they can be read. In a manner similar to bit position D4, bit D5 must be driven high for a minimum of 12 microseconds, then low.

Bit D6 steps the scan counter. In a manner similar to bit positions D4 and D5, the "step scan counter" signal must be a pulse with a minimum width of 4 microseconds. Bit position D6 is used in conjunction with multiple reads from the high address to multiplex the nine counter digits into one computer address. That is, the scan counter is steped, using bit D6, following each read.

Bit position D7 is used to enable and disable the 88-UFC interrupt system. A "1" in bit position D7 enables the interrupt system.

DATA OUT - HIGH ADDRESS

D7	D6	D5	D4	D3	D2	D1	D0
SOURCE SELECT				FUNCTION			
TTL	SINE	FET	PRESCALER	FREQUENCY	COUNT	PERIOD	WHOLE CYCLE/ NEGATIVE HALF CYCLE

When the "Period" function is selected (bit D1 = 1) and bit position DØ = "0", the period of the negative half cycle of the input signal is measured. That is, the measurement is made from the first negative transition (of the signal) following counter activation (writing out to the high address) to the next positive transition. If the "Period" function is selected and bit DØ = "1", the period of a whole cycle (time between two positive signal transitions) is measured.

The functions defined by bits D1 through D7 are enabled by a "1" and disabled by a "0".

Bits D1 through D3 select the counter function. If the "Period" of "Frequency" function is selected, the measurement is initiated by the strobe pulse that loads the high address control register (the control register presently being discussed). If the "Count" mode is selected, the presence of the "1" in bit position D2 opens the selected input gate. Multiple modes can be simultaneously selecte However, the user must understand operation of the counter under such circumstances in order to properly evaluate the measurement results. For example, it is possible to use a combination of the "Count" and "Frequency" modes in conjunction with the interrupt capability to extend frequency measurements to any desired number of seconds (the details of this mode of operation <u>are</u> <u>not</u> described in this manual).

Bits D4 through D7 select the input signal source. Multiple sources may be simultaneously selected. However, the user must be aware of th relationships among the several active inputs if the measurement results are to be meaningful. For example, if the 88-UFC is placed in the "Count" mode and two inputs are simultaneously activated, one input can be used to control the counting of events occuring on the other input. If the TTL and Sine inputs are, for example, simultaneously activated, the counting of events on the Sine input will be inhibited as long as the TTL input is at a logic high, whereas events on the Sine input will be counted during intervals when the TTL input is at a logic low. If this mode of operation is used, the

TTL input should normally be used as the control input, since it is the only one of the four inputs that is DC coupled.

DATA IN - LOW ADDRESS

D7	D6	D5	D4	D3	D2	D1	D0
NOT USED						OVERFLOW FLAG	READY FLAG

Bit position DØ indicates when a measurement is in progress (period or frequency - does not operate in the count mode). It DØ is low while a measurement is in progress, and goes high when the measurement cycle is completed and the results can be read.

Bit position D1 indicates counter overflow (reading greater than 999,999,999). A "1" indicates overflow.

Bit positions D2 through D7 are not used, and are always high ("1").

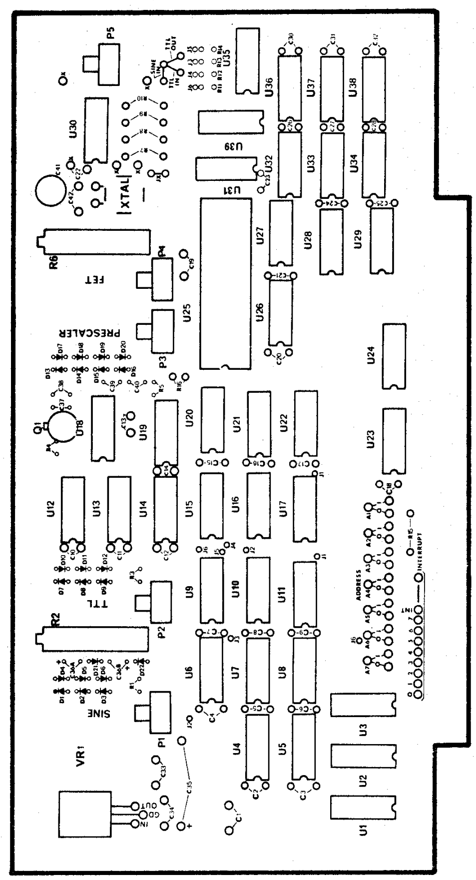
DATA IN - HIGH ADDRESS

D7	D6	D5	D4	D3	D2	D1	D0
NOT USED				BCD DATA			

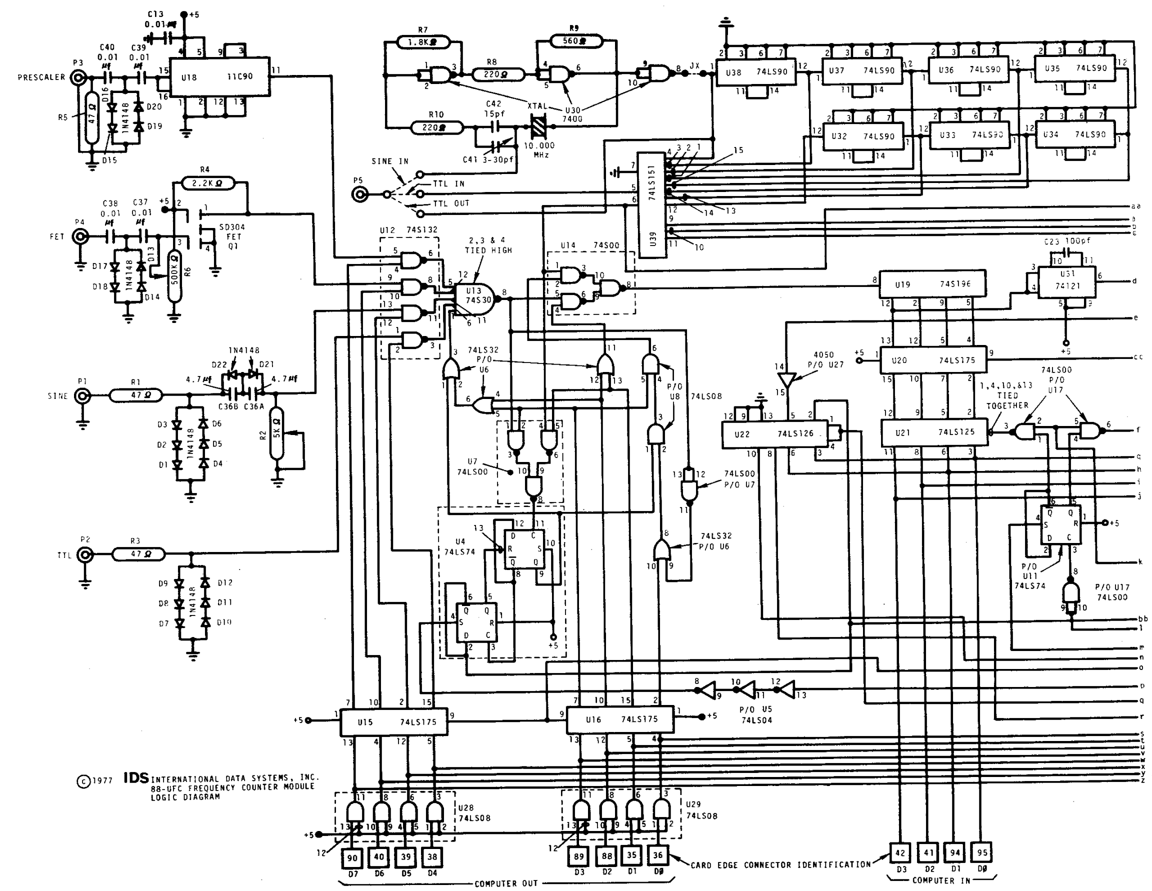
Bit positions DØ through D3 (with DØ least significant) represent a single decimal digit in binary coded decimal (BCD) form.

Bit positions D4 through D7 are not used, and are always high ("1").

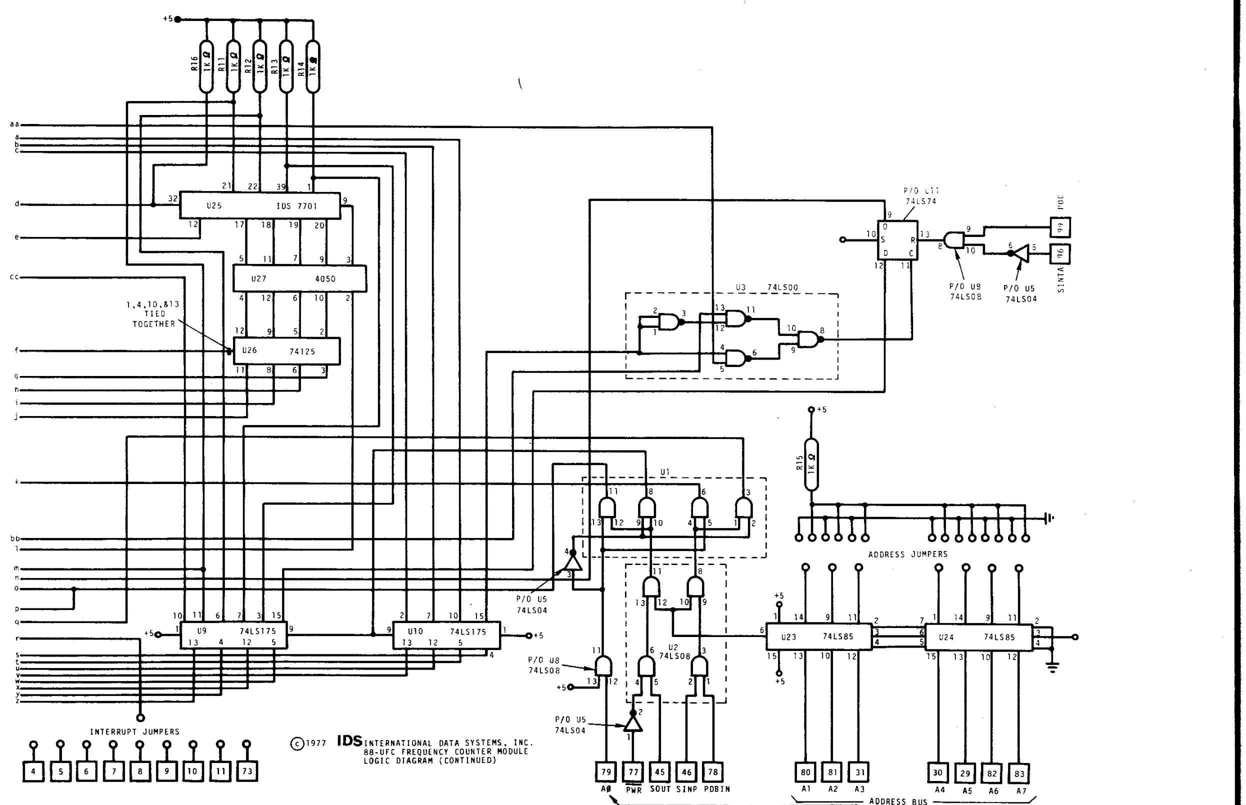
SECTION II - SCHEMATIC DIAGRAM



SINE
TTL
PRESCALER
FET
XTAL
VR1
R2
R6
P1
P2
P3
P4
P5
Q1
ADDRESS
INTERRUPT
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
U15
U16
U17
U18
U19
U20
U21
U22
U23
U24
U25
U26
U27
U28
U29
U30
U31
U32
U33
U34
U35
U36
U37
U38
U39



 IDS INTERNATIONAL DATA SYSTEMS, INC.
88-UFC FREQUENCY COUNTER MODULE
LOGIC DIAGRAM



©1977 IDS INTERNATIONAL DATA SYSTEMS, INC.
88-UFC FREQUENCY COUNTER MODULE
LOGIC DIAGRAM (CONTINUED)

SECTION III - ASSEMBLY INSTRUCTIONS

3. PREFACE

Some experience in kit building is assumed by the assembly instructions for the 88-UFC kit. A minimum of installation directions is given in the text. A diagram of the printed circuit board has been included on page 3-3 for reference when orienting and installing components on the 88-UFC circuit board.

In the unlikely event that the builder has no previous experience in building printed circuit boards for S-100 bus computers, a free booklet "Introduction to Kit Building" is available by writing or calling International Data Systems, Inc.

3.1 STEP 1 Socket Installation

If sockets were included with your kit, install the sockets as indicated below. Orient pin 1 of each socket as shown by the notch in the silk screen representation of the IC.

( ) Install 14-pin sockets for the following IC's:

() U1	() U8	() U21	() U32
() U2	() U11	() U22	() U33
() U3	() U12	() U26	() U34
() U4	() U13	() U28	() U35
() U5	() U14	() U29	() U36
() U6	() U17	() U30	() U37
() U7	() U19	() U31	() U38

( ) Install 16-pin sockets for the following IC's:

() U9	() U16	() U23	() U27
() U10	() U18	() U24	() U39
() U15	() U20		

( ) Install one 40-pin IC socket:

( ) U25

3.2 STEP II Diode Installation

When installing diodes, the banded end of the diode must correspond with the banded end printed on the 88-UFC circuit board. All diodes installed on the 88-UFC must be mounted vertically instead of being mounted flat on the sirface of the circuit board. Figure 3.2-1 gives an example of vertical mounting.

( ) Install 22 diodes:

( ) D1 through D22 - 1N4148

3.3 STEP III Resistor Installation

Install the following resistors, mounting them flat against the surface of the 88-UFC circuit board:

( )	R7 - 1.8K ohms, 1/4 watt	(brown-grey-red)
( )	R8 - 220 ohms, 1/4 watt	(red-red-brown)
( )	R9 - 560 ohms, 1/4 watt	(green-blue-brown)
( )	R10 - 220 ohms, 1/4 watt	(red-red-brown)
( )	R15 - 1K ohms, 1/4 watt	(brown-black-red)

The following resistors must be mounted in a vertical position on the circuit board (see Figure 3.2-1):

( )	R1 - 47 ohms, 1/4 watt	(yellow-violet-black)
( )	R3 - 47 ohms, 1/4 watt	(yellow-violet-black)
( )	R4 - 2K ohms, 1/4 watt	(red-black-red)
( )	R5 - 47 ohms, 1/4 watt	(yellow-violet-black)
( )	R11 - 1K ohms, 1/4 watt	(brown-black-red)
( )	R12 - 1K ohms, 1/4 watt	(brown-black-red)
( )	R13 - 1K ohms, 1/4 watt	(brown-black-red)
( )	R14 - 1K ohms, 1/4 watt	(brown-black-red)
( )	R16 - 1K ohms, 1/4 watt	(brown-black-red)

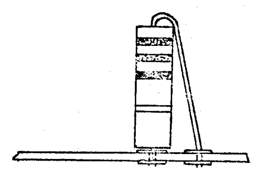
Install the two remaining resistors:

( ) R2 - 5K potentiometer

( ) R6 - 500K potentiometer

Figure 3.2-1

Vertical Orientation for Diodes and Resistors



3.4 STEP IV Capacitor Installation

Carefully observe the polarity markings indicated on the printed circuit board when installing the two tantalum capacitors (C36A and C36B) and the electrolytic capacitor (C35).

( ) C23 - 75 pf, NPO capacitor
( ) C35 - 47 mfd or 50 mfd, 25V electrolytic capacitor
( ) C36A - 4.7 uf, 16 V tantalum capacitor
( ) C36B - 4.7 uf, 16 V tantalum capacitor
( ) C41 - trimmer capacitor
( ) C42 - 20 pf, NPO capacitor

All remaining capacitors on the 88-UFC circuit board are .01 uf ceramic disc capacitors.

( ) Install 36 .01 ceramic disc capacitors

NOTE: C22 uses one of the pads labelled "X" for its upper lead. No connections are made to the other pads labelled "X".

3.5 STEP V Transistor Installation

Install one transistor, aligning the leads of the transistor with the markings on the circuit board:

( ) Q1 - SD304 FET transistor

3.6 STEP VI Crystal Installation

Install one crystal:

( ) XTAL - 10 MHz crystal

3.7 STEP VII Voltage Regulator Installation

Install the voltage regulator using the heat sink provided with the kit:

( ) VR1 - 7805 voltage regulator

3.8 STEP VIII Phono Plug Installation

Install five phono plugs (only one orientation is possible):

( ) P1 through P5 - phono plugs

3.9 STEP IX Integrated Circuit Installation

Standard 74 series, low power (L), and low power shottkey (LS) TTL components are interchangeable for all IC's except U12, U13, U14, U19, and U31 as indicated below. Therefore, substitutions may be provided, based on availability, for the other IC's. In all cases, the set of IC's provided will collectively require less than the bus specification for power requirements for this board.

Install the following IC's, observing the proper orientation of pin 1:

( ) U1 - 7408
( ) U2 - 7408
( ) U3 - 7400
( ) U4 - 7474
( ) U5 - 7404
( ) U6 - 7432
( ) U7 - 7400
( ) U8 - 7408
( ) U9 - 74175
( ) U10 - 74175
( ) U11 - 7474
* ( ) U12 - 74S132 NO Substitutions
* ( ) U13 - 74S30 NO Substitutions
* ( ) U14 - 74S00 NO Substitutions
( ) U15 - 74175
( ) U16 - 74175
( ) U17 - 7400
( ) U18 - 11C90
* ( ) U19 - 74S196 NO Substitutions
( ) U20 - 74175
( ) U21 - 74125
( ) U22 - 74126
( ) U23 - 7485
( ) U24 - 7485
( ) U25 - LS7030
( ) U26 - 74125
( ) U27 - 4050
( ) U28 - 7408

( ) U29 - 7408
( ) U30 - 7400
* ( ) U31 - 74121 NO Substitutions
( ) U32 - 7490
( ) U33 - 7490
( ) U34 - 7490
( ) U35 - 7490
( ) U36 - 7490
( ) U37 - 7490
( ) U38 - 7490
( ) U39 - 74151

3.9A Jumper Wire Installation

Install one jumper on the component side of the printed circuit board using a discarded lead from one of the components. In the lower left corner of the circuit board, install the jumper between the lands in the wide power bus below ICs U2 and U3. Two unlabeled holes are provided.

( ) Install power bus jumper.

Use a sharp knife, such as an "Exacto knife," to cut around the insulation and strip the ends of the wire provided in the kit. Cut the jumpers to 1/2 inch longer than the length required for each connection and strip 1/4 inch on each end. Install all jumpers on the back side of the printed circuit board while referencing the silkscreen designations on the component side of the board. Install the following jumpers:

( ) J1-J1

( ) J2-J2

( ) J3-J3

( ) J4-J4

( ) J5-J5

( ) J6-J6

3.9B Timebase Option Selection

To use the onboard 10MHz crystal timebase install the following jumpers.

( ) . JX-JX Jumper JX connects the two holes nearest the lower righthand corner of the crystal and immediately below the two holes labeled "X". The "JX" pads are unlabeled on some PC boards.

( ) TTL OUT Connect one jumper below jack P5. The TTL OUT jumper connects the two holes connected by the white line labeled TTL OUT.

See section 4.3 for information regarding non-standard timebase connections.

3.10 CALIBRATION

The frequency determining capacitor, C41, for the 10 MHz crystal oscillator (or for the TXCO if this option has been selected) may be set in any of three ways. The three methods are defined below.

3.10.1 Calibration Method I

To use this method, the timebase signal from TP1 is coupled to an external frequency standard, which may be either another counter of known calibration or to a radio receiver tuned to a stable radio frequency broadcast. The timebase rate is set to the desired output rate (10 mc if possible). Trimmer C41 is then adjusted for zero beat if a radio receiver is used, or to the calibration frequency if an external counter is used.

3.10.2 Calibration Method II

Method II requires a stable, highly accurate source at any frequency within the operating range of the 88-UFC. After this frequency is coupled to an input channel on the 88-UFC and the counter is put into frequency mode, trimmer C41 is adjusted until the printed output corresponds to the input frequency of the standard used for input. If highly accurate square waves are available, the frequency mode could also be used with this method.

3.10.3 Calibration Method III

Method III is defined as "dynamic calibration". The 88-UFC trimmer may be set as close as possible with either of the two methods defined above before proceeding, but this is not required before using the dynamic calibration method.

An external stable source must first be coupled to an input channel of the 88-UFC to use this method. An unknown frequency (the one to be measured) is coupled to another input channel. The 88-UFC is then used to measure the standard frequency and the results of the measurement are saved. This is repeated with the unknown frequency and the BCD arithmetic routines provided with the 88-UFC are then used to determine the frequency of the unknown signal by use of the following computations:

```
SF - MV = E
SF / E = FE
FE * MVu + MVu = Fu

        where SF=Standard Frequency
              MV=Measured Value of Standard Frequency
              E=Error in CPS
              FE=Fractional Error
              MVu=Measured Value of Unknown Frequency
              Fu=Frequency of Unknown Signal
```

The dynamic calibration procedure will yield the highest overall accuracy and will give an accuracy equal to that provided by the external stable source. A possible source of this highly stable frequency might be that now being broadcast by some TV stations at given times. A description of this signal and a method for extracting it was described in "Popular Electronics".

If this signal is used with the dynamic calibration procedure described above, it is possible to make frequency measurements that are accurate to one part in 10^{-x}. The signal source derived from the TV signal is supposed to be within one in 10^{x}. The difference stems from the fact that the short term drift (several seconds) of the 88-UFC is better than that, making it possible to make frequency measurments and compare the two sources in about three seconds.

For highest accuracy, several measurements should be made and the results averaged by the use of the BCD arithmetic provided with the 88-UFC.

3.11 Use of 88-UFC Demonstration Program to Adjust the SINE and FET Trimmers

The input trimmer resistors, R2 and R6, need only be set one time. Their purpose is to set the of the Schmidt Trigger to the center of its range, thereby providing even triggering on both the positive and negative portions of the wave shape. Adjusting the trimmer pots to this point also sets these inputs to the most sensitive point.

The program for setting these pots to the correct point will place the 88-UFC in the period measuring mode and print first the whole period and then the half-period measurement. The adjustment procedure is as follows:

1 - Select the SINE input, period measurement within .1 us timebase

2 - Input a stable 1-10 KHz sine wave signal to the SINE input. The level should be about 1V p-p.

3 - With the program running, set the sine input trimmer, R2, so that the half-period measurement is exactly half the value printed for the whole period.

4 - Select the FET input for period measurement and .1 us timebase.

5 - Repeat steps 2 and 3, adjusting the FET trimmer, R6, with the input applied to the FET input.

A sample printout while making this adjustment is shown in Figure 5-4 "Example of Period Measurement". The first printed values show a period of 1001 and a half-period of 454 milliseconds. The adjustment is made and the next set of measurements indicates that the set point was passed as the half-period measurement was equal to 510 ms. Adjustments were continued and the next set of measurements show the final setting. If there is any jitter in the measured signal it will show as a variation in the period and half-period measurements. This calibration procedure is mainly for setting the pots to the maximum sensitivty point of the input circuit for the sine wave type of input signals and may not necessarily represent accurate period measurements for complex wavw type signals. The 88-UFC will provide accurate period measurement only for square wave inputs to the TTL channel. If the period function is used on any input channel other than the TTL channel the user should be aware that the wave shape of the signal may be such that it may not be possible to interpret the measured value correctly for half-periods. Because of the action of the Schmidt Trigger input, the full period measure will generally be that of the dominant frequency component of the complex wave.

SECTION 4 - ADDRESS AND OPTION SELECTION

4.1 I/O ADDRESS SELECTION

The 88-UFC occupies two consecutive I/O addresses beginning on any even integer between 0 and 254 decimal. The address is selected via seven jumpers located on the lower left side of the component side of the 88-UFC printed circuit board.

An address selection chart is provided in Table 4.1-1. Below each pad labelled A1 through A7 on the printed circuit board is a pad labelled "0" and a pad labelled "1". In the jumper connection column of Table 4.1-1 the jumpers are specified for each available address.

All software provided with the 88-UFC is written for the counter to reside at 128 decimal. Referring to Table 4.1-1, A7 is marked "1" and A1 through A6 are marked "0". To select address 128 connect a jumper from pad A7 to the "1" pad below it and jumpers from pads A1 through A6 to the "0" pad below each.

4.2 INTERRUPT SELECTION

The 88-UFC can generate interrupts according to S-100 bus conventions. Interrupts are enabled on the 88-UFC under software control. See Sections 1 and 5 for software details. For the interrupt signal generated by the 88-UFC to reach the processor one interrupt level jumper must be installed. If

Figure 4.1-1

ADDRESS SELECTION TABLE

ADDRESS				JUMPER CONNECTIONS						
DECIMAL	OCTAL			A7	A6	A5	A4	A3	A2	A1
0	0	0	0	0	0	0	0	0	0	0
2	0	0	2	0	0	0	0	0	0	1
4	0	0	4	0	0	0	0	0	1	0
6	0	0	6	0	0	0	0	0	1	1
8	0	1	0	0	0	0	0	1	0	0
10	0	1	2	0	0	0	0	1	0	1
12	0	1	4	0	0	0	0	1	1	0
14	0	1	6	0	0	0	0	1	1	1
16	0	2	0	0	0	0	1	0	0	0
18	0	2	2	0	0	0	1	0	0	1
20	0	2	4	0	0	0	1	0	1	0
22	0	2	6	0	0	0	1	0	1	1
24	0	3	0	0	0	0	1	1	0	0
26	0	3	2	0	0	0	1	1	0	1
28	0	3	4	0	0	0	1	1	1	0
30	0	3	6	0	0	0	1	1	1	1
32	0	4	0	0	0	1	0	0	0	0
34	0	4	2	0	0	1	0	0	0	1
36	0	4	4	0	0	1	0	0	1	0
38	0	4	6	0	0	1	0	0	1	1
40	0	5	0	0	0	1	0	1	0	0
42	0	5	2	0	0	1	0	1	0	1
44	0	5	4	0	0	1	0	1	1	0
46	0	5	6	0	0	1	0	1	1	1
48	0	6	0	0	0	1	1	0	0	0
50	0	6	2	0	0	1	1	0	0	1
52	0	6	4	0	0	1	1	0	1	0
54	0	6	6	0	0	1	1	0	1	1
56	0	7	0	0	0	1	1	1	0	0
58	0	7	2	0	0	1	1	1	0	1
60	0	7	4	0	0	1	1	1	1	0
62	0	7	6	0	0	1	1	1	1	1
64	1	0	0	0	1	0	0	0	0	0
66	1	0	2	0	1	0	0	0	0	1
68	1	0	4	0	1	0	0	0	1	0
70	1	0	6	0	1	0	0	0	1	1
72	1	1	0	0	1	0	0	1	0	0
74	1	1	2	0	1	0	0	1	0	1
76	1	1	4	0	1	0	0	1	1	0
78	1	1	6	0	1	0	0	1	1	1
80	1	2	0	0	1	0	1	0	0	0
82	1	2	2	0	1	0	1	0	0	1
84	1	2	4	0	1	0	1	0	1	0

Figure 4.1-1
Cont'd

ADDRESS SELECTION TABLE

ADDRESS DECIMAL	OCTAL			JUMPER CONNECTIONS A7	A6	A5	A4	A3	A2	A1
86	1	2	6	0	1	0	1	0	1	1
88	1	3	0	0	1	0	1	1	0	0
90	1	3	2	0	1	0	1	1	0	1
92	1	3	4	0	1	0	1	1	1	0
94	1	3	6	0	1	0	1	1	1	1
96	1	4	0	0	1	1	0	0	0	0
98	1	4	2	0	1	1	0	0	0	1
100	1	4	4	0	1	1	0	0	1	0
102	1	4	6	0	1	1	0	0	1	1
104	1	5	0	0	1	1	0	1	0	0
106	1	5	2	0	1	1	0	1	0	1
108	1	5	4	0	1	1	0	1	1	0
110	1	5	6	0	1	1	0	1	1	1
112	1	6	0	0	1	1	1	0	0	0
114	1	6	2	0	1	1	1	0	0	1
116	1	6	4	0	1	1	1	0	1	0
118	1	6	6	0	1	1	1	0	1	1
120	1	7	0	0	1	1	1	1	0	0
122	1	7	2	0	1	1	1	1	0	1
124	1	7	4	0	1	1	1	1	1	0
126	1	7	6	0	1	1	1	1	1	1
128	2	0	0	1	0	0	0	0	0	0
130	2	0	2	1	0	0	0	0	0	1
132	2	0	4	1	0	0	0	0	1	0
134	2	0	6	1	0	0	0	0	1	1
136	2	1	0	1	0	0	0	1	0	0
138	2	1	2	1	0	0	0	1	0	1
140	2	1	4	1	0	0	0	1	1	0
142	2	1	6	1	0	0	0	1	1	1
144	2	2	0	1	0	0	1	0	0	0
146	2	2	2	1	0	0	1	0	0	1
148	2	2	4	1	0	0	1	0	1	0
150	2	2	6	1	0	0	1	0	1	1
152	2	3	0	1	0	0	1	1	0	0
154	2	3	2	1	0	0	1	1	0	1
156	2	3	4	1	0	0	1	1	1	0
158	2	3	6	1	0	0	1	1	1	1
160	2	4	0	1	0	1	0	0	0	0
162	2	4	2	1	0	1	0	0	0	1
164	2	4	4	1	0	1	0	0	1	0
166	2	4	6	1	0	1	0	0	1	1
168	2	5	0	1	0	1	0	1	0	0
170	2	5	2	1	0	1	0	1	0	1

Figure 4.1-1
Cont'd

ADDRESS SELECTION TABLE

ADDRESS DECIMAL	OCTAL			JUMPER CONNECTIONS A7	A6	A5	A4	A3	A2	A1
172	2	5	4	1	0	1	0	1	1	0
174	2	5	6	1	0	1	0	1	1	1
176	2	6	0	1	0	1	1	0	0	0
178	2	6	2	1	0	1	1	0	0	1
180	2	6	4	1	0	1	1	0	1	0
182	2	6	6	1	0	1	1	0	1	1
184	2	7	0	1	0	1	1	1	0	0
186	2	7	2	1	0	1	1	1	0	1
188	2	7	4	1	0	1	1	1	1	0
190	2	7	6	1	0	1	1	1	1	1
192	3	0	0	1	1	0	0	0	0	0
194	3	0	2	1	1	0	0	0	0	1
196	3	0	4	1	1	0	0	0	1	0
198	3	0	6	1	1	0	0	0	1	1
200	3	1	0	1	1	0	0	1	0	0
202	3	1	2	1	1	0	0	1	0	1
204	3	1	4	1	1	0	0	1	1	0
206	3	1	6	1	1	0	0	1	1	1
208	3	2	0	1	1	0	1	0	0	0
210	3	2	2	1	1	0	1	0	0	1
212	3	2	4	1	1	0	1	0	1	0
214	3	2	6	1	1	0	1	0	1	1
216	3	3	0	1	1	0	1	1	0	0
218	3	3	2	1	1	0	1	1	0	1
220	3	3	4	1	1	0	1	1	1	0
222	3	3	6	1	1	0	1	1	1	1
224	3	4	0	1	1	1	0	0	0	0
226	3	4	2	1	1	1	0	0	0	1
228	3	4	4	1	1	1	0	0	1	0
230	3	4	6	1	1	1	0	0	1	1
232	3	5	0	1	1	1	0	1	0	0
234	3	5	2	1	1	1	0	1	0	1
236	3	5	4	1	1	1	0	1	1	0
238	3	5	6	1	1	1	0	1	1	1
240	3	6	0	1	1	1	1	0	0	0
242	3	6	2	1	1	1	1	0	0	1
244	3	6	4	1	1	1	1	0	1	0
246	3	6	6	1	1	1	1	0	1	1
248	3	7	0	1	1	1	1	1	0	0
250	3	7	2	1	1	1	1	1	0	1
252	3	7	4	1	1	1	1	1	1	0
254	3	7	6	1	1	1	1	1	1	1

interrupt mode is not to be employed, an interrupt jumper need not be installed.

If vectored interrupts are not implemented in the system in which the 88-UFC is installed then the only choice is the "INT" bus line. To enable non-vectored interrupts place a jumper from the pad labelled "INTERRUPT" to the pad labelled "INT".

If vectored interrupts are implemented on the host processor then a jumper between the pad labelled "INTERRUPT" and the desired interrupt level pad to its left may be installed. The pads labelled "0" through "7" correspond to interrupt levels 0 through 7.

SECTION 5 - APPLICATIONS SOFTWARE

5. INTRODUCTION

All functions of the 88-UFC counter are completely under software control. This is accomplished by the use of two input addresses and two output addresses. The demonstration software in this section assumes that the address jumpers of the counter are wired for octal 200 and 201 (decimal 128, 129; Hex 80,81). Although the 88-UFC may be addressed at any pair of adjacent addresses available to the 8080 CPU, the software in this section will use octal 200 for Address 0 and octal 201 for Address 1. The user may strap the address jumpers on the board to any valid pair of addresses as discussed in Section 4 of this manual.

5.1 CONTROL WORDS

Because the 88-UFC operates entirely under program control multiple commands are required to set up a measurement. The functional description of the 88-UFC contained in Section i of this manual should be referenced to clarify the software procedures described in this section.

All "OUT" software command bits are provided with latches on the 88-UFC board where the status is "stored" as long as it is needed. Once "stored", it appears that they need not be changed unless the 88-UFC functions are to be changed. Some control bits, however, are dynamic in actual use. Each output word is described below along with any restrictions to its use.

Output word 1 latches are completely static and are set only once for each type of measurement. However, this address also provides the "GO" strobe when an "OUT" is sent to the 88-UFC. Therefore, the "OUT" commands to address 1 must also contain the proper control bit pattern each time it is executed as required for the measurement to be made. Only one "OUT" to address 1 is required to start each measurement. In the case of the "COUNT" measurements, only one "OUT" is required for each entire count sequence since, once started, the 88-UFC will continue to run until stopped by other control sequences to address word 0.

Output word 0 is multi-purpose in structure and several sequence "OUTs" are required for each measurement. Also, three bit positions (bit 4-reset, bit 5-load latch, and bit 6-scan) require an additional consideration in their use. These control bits are also used as software controllable strobes which provide for direct control over the internal functions of the countdown chain and for scanning of the completed measurement data so all nine digits are available at a single input address.

This strobing action is accomplished by an "OUT" to address 0 with the appropriate bit set to "1" followed by an "OUT" with this same bit set to "0". The 88-UFC was designed in this manner to allow full software control over all counter functions so timing would not be dependent upon combinations of one-shot or other timing circuits. It may now appear that by manipulating output bit combinations the user may have simultaneous operation of several counter functions. While this appears feasable it is not recommended for counter use. In addition, changes should not be made to any counter register during a counter operational sequence (except for scan and special features as explained later). For example, executing a load latch sequence in the middle of a count operation will cause the measurement in progress to be latched into the output registers, destroying the previous results with a new partial measurement which has no meaning since the point at which the latch took place has no relationship to the current value of the count.

Although there are several possible control sequences that will satisfy the 88-UFC control requirements, the first time user of the counter should use those discussed below until a thorough understanding of counter operation has been gained.

5.2 HOW TO PROGRAM THE 88-UFC

The following series of examples shows the construction of the control words for initializing the 88-UFC and making the measurement. These are presented in a simplified way to aid the user in understanding the demonstration programs that are included in this manual.

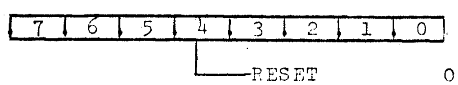
For clarity the control words are shown with only the applicable bits extracted for the current sequence of each operation. Each operation should be performed in the sequence given with only the bits necessary for the current operation being used.

The example used demonstrates setting up the counter for a frequency measurement with a one second timebase and the input signal from the SINE input connection. All other operations are to be demonstrated by the use of the software routines which follow this example.

5.2.1 STEP 1 - RESET

The control parameter bit 4 for address 0 is strobed. To reset the 88-UFC to zero, the following code is used where Q=Output Port 128 as described earlier in this section.

Output Address 0:



Octal 20
Hex 10
Decimal 16

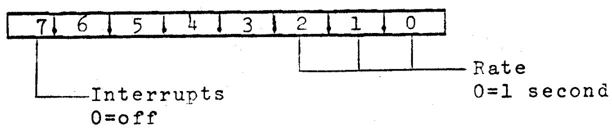
In BASIC: OUT Q,16 STROBE ON
OUT Q,0 STROBE OFF

In assembly:
```
MVI A,10H
OUT 80H
MVI A,0
MVI 80H
```

5.2.2 STEP 2 - SET-UP 0

The following is used to establish the control parameters for interrupts off and a one second timebase.

Output Address 0:



In BASIC: OUT Q,0

In assembly:
```
MVI A,0
OUT 80H
```

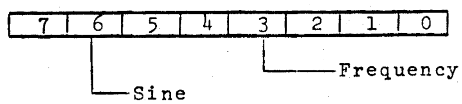
NOTE: Although in this example the "STROBE OFF" command and the set-up command are the same and only one "OUT" is necessary, care must be used if these commands are combined for other types of set-up, in particular, thoses involving the use of interrupts.

The separation is shown to help in understanding how counter registers operate. In actual practice the RESET STROBE OFF command and the SET-UP command may be combined if the system interrupts are not active and the counter is not in count mode.

5.2.3 STEP 3 - SET-UP 1

The next sequence establishes the control parameters for address 1 for frequency measurement from the sine input.

Output Address 1:



Control word: Octal 110
Hex 48
Decimal 72

In BASIC: OUT Q+1,72

In assembly:

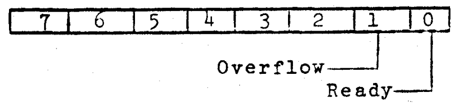
```
MVI A,48H
OUT 81H
```

This step completes the initialization sequences. The "OUT" instruction to address 1 also primes the counter which will start the measurement at the first clock transition from the countdown chain.

5.2.4 STEP 4 - WAIT

The counter will now require one second to compute its measurement (if a different rate had been selected that amount of time would have been required to complete the measurement). Since interrupts are not being used in this example, we must wait for the count to complete before continuing. Only input bits 0 and 1 of address 0 are used (testing for busy requires only bit 0). All other bits will be read in as "1".

Input Address 0:



In BASIC: 100 IF (INP(Q)AND1)=0 THEN 100

In assembly:

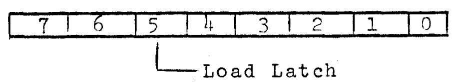
```
WAIT  IN 80H
      ANI 1
      JZ WAIT
```

The overflow bit 1 may be tested if overflow could be expected. This indicates that the count exceeded nine digits (not possible for frequency measurements).

5.2.5 STEP 5 - LOAD LATCH

After the measurement is completed the result is latched into the output holding register for use by the computer.

Output Address 0:



Control word: Octal 040
Hex 20
Decimal 32

In BASIC:

```
OUT Q,32     STROBE ON
OUT Q,0      STROBE OFF
```

In assembly:

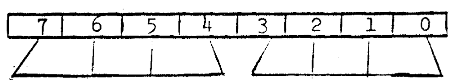
```
MVI A,20H
OUT 80H
MVI A,0
OUT 80H
```

The results are now loaded into the 88-UFC holding register. At this time the counter is free for resetting and establishing parameters for the next measurement if full overlay processing is desired. For this example, however, we will now get and display the results.

5.2.6 STEP 6 - READ

Now that the results are in the holding register it is necessary to scan this register and bring the results into the computer for use or display. This is a two stage process involving first READ then SCAN. The results read from the register may be stored for future use or printed. In this example the register contents will be printed as it is read into the computer. Because the most significant digit is the first to be read in it is input first, then the command is given to scan to the next digit. Scanning takes place from left to right.

Input Address 1:



All "ones"
Value is
240 decimal

BCD Data

In BASIC: XX=INP(Q+1)-240

XX now contains the decimal value of the digit. The four high order bits (which are always "1") are discarded by subtracting their decimal value of 240.

In assembly:

```
IN 81H
ANI F0H 0FH
```

The A register now contains the value of the digit since the ANI instruction masks off the high order bits.

To print the most significant digit that was just extracted from the input address, the following commands are used:

In BASIC:	PRINT XX;	
In assembly:	OUT 1	(for I/O Port 1)

NOTE: In general, the counter is slower than the printer when measuring frequency. If not, as for period measurements, the appropriate test for busy must be made in the assembly OUT program to delay the counter operation for the slower printer.

5.2.7 STEP 7 - SCAN

Output to Address 0:

7	6	5	4	3	2	1	0
	Scan						

Control word:	Octal 100 Hex 40 Decimal 64	
In BASIC:	OUT Q,64 OUT Q,0	SCAN ON SCAN OFF
In assembly:	MVI A,40H OUT 80H MVI A,0 OUT 80H	

The next character is now ready to be read into the computer. The program must loop to pick up all nine digits. A FOR-NEXT loop may be established with all functions from READ to Print within the loop or a program counter may be used with a branch to the READ statement until all digits have been retrieved.

NOTE: As presented above, the SCAN command resets the rate and interrupt latches to zero. This presents no problem unless interrupts are enabled or the counter is in an interrupt mode. Interrupt operation and overlay operation require that the control command established for output word 0 (obtained in Step 2 - SET-UP 0) be ORed into the scan command before the scan is strobed. This value is saved in variable R in the examples. The following changes must be made to the scan command:

```
In BASIC:           OUT Q,64+R
                    OUT Q,R

In assembly:        MVI A,40H
                    ORI R
                    OUT 40H
                    MVI A,R
                    OUT 40H
```

This completes the explanation of setting up parameter words for a simple frequency measurement. The remaining operations for the 88-UFC are set up in a similar way.

5.2 SOFTWARE DESCRIPTION

On the following pages are listed BASIC and assembly language programs that aid in establishing all measurement functions for the 88-UFC. The first program will establish the necessary control words required for the various measurements in response to keyboard entries by the operato and will then execute the required measurement. It is suggested that this program, listed in Figure 5-1 for MITS BASIC, be used as a test and familiarization program for the 88-UFC. A sample output is included in Figure 5-2.

The second example, listed in Figure 5-3, allows the user to compose control words which may be saved for use in single function measurement programs. This programs composes the control words for the user so that it is not necessary to use the procedure outlined in Section 5.1 each time a new set of measurement functions need to be established. Figures 5-5 through 5-9 show uses of this program.

Although both of these programs are described independently, it is necessary for the user to read and understand both this section and the hardware description sections of this manual. The demonstration program is discussed in Section 5.3 and the compose program is discussed in Section 5.4.

In addition to these two programs, listings are also included for the conversion of the nine-digit BCD output of the 88-UFC to internal BASIC floating point representation. The number of significant digits is then reduced to six unless extended BASIC is available. For the user without extended BASIC, subroutines have been included to perform nine digit BCD arithmetic. Figure 5-9 is a programming example showing how to save counter output in BCD. Figure 5-10 contains a listing of the BCD arithmetic subroutines. Applications for these programs are far to numerous to include in the scope of this manual.

Changes required to use the programs with Processor Technology BASIC5 are listed in Figure 5-12. Figure 5-13 contains the assembly language patch required for use with Processor Technology BASIC5.

The last program include in this section is a stand-alone assembly language demonstration program in which the 88-UFC is set in the period-interrupt mode. This program demonstrates the versatility of the 88-UFC by making period measurements on the incoming audio-frequency shift (AFSK) signal at a rate of over 2000 CPS. At this rate period measurements of at least 1000 per second are required. The AFSK signal follows those used for radio teletype communications for narrow band (170 Hz) shift. Software is used to demodulate this signal into its original MARK-SPACE components. A full description and listing of this program complete the software section of this manual in Section 5.5.

5.3 DEMONSTRATION PROGRAM

The demonstration program listed in Figure 5-1 was designed to allow the 88-UFC user to establish all control functions that are required by the hardware registers within the counter and to select an input channel, make measurements and display them.

The program allows the user to enter control parameters in the proper order for setting up and running counter functions without using the procedure described in Section 5.2 each time a new function is desired. The user is prompted for each entry and will receive an error message if a value is entered that is not acceptable to the program. Abbreviated entry is also provided, restricting the prompts to the required values without printing the available choices. After initialization of the control words in this manner, the 88-UFC goes into operation. Measurements will be made and displayed until the program is stopped with a "Control C" entry.

Figure 5-1

88-UFC Demonstration Program
MITS 8K BASIC Source Listing

```
1 REM DEMONSTRATION PROGRAM TO SHOW ALL PROGRAMMIMG
2 REM FEATURES OF THE  88-UFC  UNIVERSAL FREQUENCY
3 REM COUNTER MODULE.
4 REM
5 PRINT"DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO."
6 INPUT K
7 PRINT"ENTER PORT NO IN DECIMAL"
8 INPUT Q
10 PRINT"ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO."
11 INPUT J
12 IF J=1 THEN I1=1
100 PRINT"ENTER COUNTER INPUT CHANNEL CODE"
108 IF K=0 THEN 115
110 PRINT"1= PRESCALER"
111 PRINT"2= FET INPUT"
112 PRINT"3= SINE WAVE INPUT"
113 PRINT"4= TTL INPUT"
115 INPUT J
116 IF J=1 THEN T1=16
117 IF J=2 THEN T1=32
118 IF J=3 THEN T1=64
119 IF J=4 THEN T1=128
120 IF J>0 AND J< 5 THEN 129
121 PRINT"CHANNEL SELECT ERROR"
122 GOTO 100
129 PRINT"ENTER MEASUREMENT FUNCTION"
130 IF K=0 THEN 140
131 PRINT"1= FREQUENCY"
132 PRINT"2= COUNT"
133 PRINT"3= PERIOD"
140 INPUT F
141 IF F=1 THEN T1=T1+8
142 IF F=2 THEN T1=T1+4
143 IF F=3 THEN T1=T1+3
145 IF F>0 AND F<4 THEN 200
146 PRINT"FUNCTION SELECT ERROR"
147 GOTO 130
200 IF F=2 THEN 228
```

Figure 5-1
Cont'd

88-UFC Demonstration Program
MITS 8K BASIC Source Listing

```
205 PRINT"ENTER TIME BASE RATE"
210 IF K=0 THEN 220
211 PRINT"0=  1S  1HZ"
212 PRINT"1= .1S  10HZ"
213 PRINT"2= .01S 100 HZ"
214 PRINT"3= 1MS  1KHZ"
215 PRINT"4= 10MS 10KHZ"
216 PRINT"5= 100MS 100KHZ"
217 PRINT"6= 1US  1MHZ"
218 PRINT"7= .1US 10MHZ"
220 INPUT R
225 IF R>=0 AND R<8 THEN 228
226 PRINT"TIME BASE ERROR"
227 GOTO 200
228 IF I1=0 THEN J=0
229 IF I1=0 THEN 240
230 PRINT"INTERRUPT SOURCE: 1= COUNTDOWN CHAIN,  0= MEAS. COMPLETION"
231 INPUT J
233 REM
236 IF J>=0 AND J<2 THEN 240
237 REM
239 GOTO 230
240 R=R+(J*8)+(I1*128)
253 S=9
300 ON F GOSUB 1000,2000,3000,2020
310 GOTO300
999 REM FREQUENCY MEASUREMENT
1000 GOTO1039
1039 OUT Q,32:OUT Q,0
1040 OUT Q,16:OUT Q,R
1041 OUT Q+1,T1
1042 JM=S-1
1043 IS=0
1050 FOR J=1 TO S
1052 XX=INP(Q+1)-240
1053 PRINTXX;
1054 IS=IS+XX*10↑JM
1055 OUT Q,64+R:OUTQ,R
1057 JM=JM-1
1060 NEXT J
1061 PRINT"    ";IS
1065 IF(INP(Q)AND1)=0 THEN 1065
1080 RETURN
```

Figure 5-1
Cont'd

88-UFC Demonstration Program
MITS 8K BASIC Source Listing

```
1999 REM COUNT
2000 GOTO2010
2010 OUT Q,16:OUT Q,R
2015 OUT Q+1,T1
2020 OUT Q,32
2021 OUT Q,0
2050 FOR J=1 TO S-1
2060 XX=INP(Q+1)
2061 PRINTXX-240;
2070 OUT Q,64+R:OUT Q,R
2090 NEXT J
2100 PRINTINP(Q+1)-240
2119 F=4
2120 RETURN
2999 REM PERIOD MEASUREMENT
3000 GOSUB 1000
3010 T1 = T1 -1
3020 GOSUB1000
3025 T1 =T1+1
3030 RETURN
```

Figure 5-2

88-UFC Demonstration Program
Sample Run

```
DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO.
? 1
ENTER PORT NO IN DECIMAL
? 128
ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO.
? 0
ENTER COUNTER INPUT CHANNEL CODE
1= PRESCALER
2= FET INPUT
3= SINE WAVE INPUT
4= TTL INPUT
? 3
ENTER MEASUREMENT FUNCTION
1= FREQUENCY
2= COUNT
3= PERIOD
? 1
ENTER TIME BASE RATE
0=  1S  1HZ
1= .1S  10HZ
2= .01S 100 HZ
3= 1MS  1KHZ
4= 10MS 10KHZ
5= 100MS 100KHZ
6= 1US  1MHZ
7= .1US 10MHZ
? 0
 0  0  0  0  1  1  0  0  0        11000
 0  0  0  0  2  2  0  0  7        22007
 0  0  0  0  2  2  0  0  6        22006
 0  0  0  0  2  2  0  0  6        22006
 0  0  0  0  2  2  0  0  4        22004
 0  0  0  2  1  5  7  6  8        215768
 0  0  0  2  2  2  8  0  3        222803
 0  0  0  0  2  3  0  4  7        23047
 0  0  0  0  0  0  0  0  0        0
 0  5  0  2  6  4  6  8  5        5.02647E+07
 0  5  0  2  6  5  6  3  7        5.02656E+07
 0  5  0  2  6  6  1  5  3        5.02661E+07
 0  5  0  2  6  7  0  3  1        5.0267E+07
 0  2  2  4  9  6  7  5  1        2.24967E+07
 0  2  2  5  5  5  2  3  3        2.25552E+07
 0  2  2  5  5  6  8  3  7        2.25568E+07
 0  0  4  7  0  6  1  0  0        4.7061E+06
```

Figure 5-2
Cont'd

88-UFC Demonstration Program
Sample Calibration Run

```
DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO.
? 1
ENTER PORT NO IN DECIMAL
? 128
ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO.
? 0
ENTER COUNTER INPUT CHANNEL CODE
1= PRESCALER
2= FET INPUT
3= SINE WAVE INPUT
4= TTL INPUT
? 3
ENTER MEASUREMENT FUNCTION
1= FREQUENCY
2= COUNT
3= PERIOD
? 3
ENTER TIME BASE RATE
0=  1S  1HZ
1= .1S  10HZ
2= .01S 100 HZ
3= 1MS  1KHZ
4= 10MS 10KHZ
5= 100MS 100KHZ
6= 1US  1MHZ
7= .1US 10MHZ
? 7
ENTER SIGNIFICANT DIGITS TO OUTPUT
? 9

 0  0  0  0  0  1  0  0  1
 0  0  0  0  0  0  4  5  4
 0  0  0  0  0  1  0  0  0
 0  0  0  0  0  0  4  5  4
 0  0  0  0  0  1  0  0  0
 0  0  0  0  0  0  4  5  4


 0  0  0  0  0  1  0  0  0
 0  0  0  0  0  0  5  1  0


 0  0  0  0  0  1  0  0  0
 0  0  0  0  0  0  5  0  0
```

5.4 COMPOSE PROGRAM

The COMPOSE program leads the user through a series of prompts requesting the information required to establish the 88-UFC control words for a given measurement function. As in the demonstration program discussed in Section 5.3, error messages are returned if an invalid response is entered. A brief form of prompting may also be chosen.

Once the initialization values have been entered by the user, the COMPOSE program prints out the control word values created as a result of those entries. These values may then be used by the user for entry in single-measurement programs as shown in Figures 5-5 through 5-9.

Figure 5-3

COMPOSE: 88-UFC Control Word Composition Program
MITS 8K BASIC Source Listing

```
1 REM THIS PROGRAM ALLOWS THE USER OF THE 88-UFC FREQUENCY
2 REM COUNTER MODULE TO GENERATE COUNTER CONTROL PRAMETERS
3 REM AND TO USE THEM FOR DIRECT KEYBOARD ENTRY, OR TO USE
4 REM THEM DIRECTLY IN SINGLE FUNCTION PROGRAMS.
5 PRINT"DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO."
6 INPUT K
7 PRINT"ENTER PORT NO IN DECIMAL"
8 INPUT Q
9 J=0
10 PRINT"ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO."
11 INPUT J
12 IF J=1 THEN I1=1
13 I2=I1
100 PRINT"ENTER COUNTER INPUT CHANNEL CODE"
108 IF K=0 THEN 114
110 PRINT"1= PRESCALER"
111 PRINT"2= FET INPUT"
112 PRINT"3= SINE WAVE INPUT"
113 PRINT"4= TTL INPUT"
114 INPUT J
115 C1=J
116 IF J=1 THEN T1=16
117 IF J=2 THEN T1=32
118 IF J=3 THEN T1=64
119 IF J=4 THEN T1=128
120 IF J>0 AND J< 5  THEN 129
121 PRINT"CHANNEL SELECT ERROR"
122 GOTO 100
129 PRINT"ENTER MEASUREMENT FUNCTION"
130 IF K=0 THEN 139
131 PRINT"1= FREQUENCY"
132 PRINT"2= COUNT"
133 PRINT"3= PERIOD"
139 INPUT F
140 F1=F
141 IF F=1 THEN T1=T1+8
142 IF F=2 THEN T1=T1+4
143 IF F=3 THEN T1=T1+3
145 IF F>0 AND F<4 THEN 200
146 PRINT"FUNCTION SELECT ERROR"
147 GOTO 130
```

Figure 5-3
Cont'd

COMPOSE: 88-UFC Control Word Composition Program
MITS 8K BASIC Source Listing

```
200 IF F=2 THEN 228
205 PRINT"ENTER TIME BASE RATE"
210 IF K=0 THEN 220
211 PRINT"0=  1S  1HZ"
212 PRINT"1= .1S  10HZ"
213 PRINT"2= .01S 100 HZ"
214 PRINT"3= 1MS  1KHZ"
215 PRINT"4= 10MS 10KHZ"
216 PRINT"5= 100MS 100KHZ"
217 PRINT"6= 1US  1MHZ"
218 PRINT"7= .1US 10MHZ"
220 INPUT R
221 R1=R
225 IF R>=0 AND R<8 THEN 228
226 PRINT"TIME BASE ERROR"
227 GOTO 200
228 IF I1=0 THEN J=0
229 IF I1=0 THEN 240
230 PRINT"INTERRUPT SOURCE: 1= COUNTDOWN CHAIN, 0= MEAS. COMPLETION"
231 INPUT J
107 REM    XX
232 I3=J
236 IF J>=0 AND J<2 THEN 240
109 GOTO 230
239 GOTO 230
240 R=R+(J*8)+(I1*128)
245 IF I2=0 THENPRINT"INTERRUPTS DISABLED"
246 IF I2=0 THEN 252
251 PRINT" INTERRUPTS ENABLED - SOURCE CODE IS "I3
252 PRINT"INPUT CHANNEL IS ";C1
253 PRINT"MEASUREMENT FUNCTION IS ";F1
254 PRINT"COUNT DOWN CHAIN RATE IS CODE ";R1:PRINT:PRINT
256 PRINT"COUNTER CONTROL VALUES FOR THE ABOVE SELECTION IS:"
257 PRINT"I-O PORT ADDRESS IS Q="Q
258 PRINT"CONTROL WORD 0 IS   R="R
259 PRINT"CONTROL WORD 1 IS  T1="T1
260 PRINT:PRINT
261 GOTO1
```

Figure 5-4

COMPOSE Program: Sample Run

```
RUN
DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO.
? 1
ENTER PORT NO IN DECIMAL
? 128
ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO.
? 1
ENTER COUNTER INPUT CHANNEL CODE
1= PRESCALER
2= FET INPUT
3= SINE WAVE INPUT
4= TTL INPUT
? 2
ENTER MEASUREMENT FUNCTION
1= FREQUENCY
2= COUNT
3= PERIOD
? 1
ENTER TIME BASE RATE
0=  1S  1HZ
1= .1S  10HZ
2= .01S 100 HZ
3= 1MS  1KHZ
4= 10MS 10KHZ
5= 100MS 100KHZ
6= 1US  1MHZ
7= .1US 10MHZ
? 0
INTERRUPT SOURCE: 1= COUNTDOWN CHAIN, 0= MEAS. COMPLETION
? 1
 INTERRUPTS ENABLED - SOURCE CODE IS  1
INPUT CHANNEL IS  2
MEASUREMENT FUNCTION IS  1
COUNT DOWN CHAIN RATE IS CODE  0


COUNTER CONTROL VALUES FOR THE ABOVE SELECTION IS:
I-O PORT ADDRESS IS Q= 128
CONTROL WORD 0 IS    R= 136
CONTROL WORD 1 IS  T1= 40
```

Figure 5-4
Cont'd

COMPOSE Program: Sample Run

```
DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO.
? 0
ENTER PORT NO IN DECIMAL
? 128
ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO.
? 0
ENTER COUNTER INPUT CHANNEL CODE
? 4
ENTER MEASUREMENT FUNCTION
? 3
ENTER TIME BASE RATE
? 7
INTERRUPTS DISABLED
INPUT CHANNEL IS  4
MEASUREMENT FUNCTION IS  3
COUNT DOWN CHAIN RATE IS CODE  7


COUNTER CONTROL VALUES FOR THE ABOVE SELECTION IS:
I-O PORT ADDRESS IS Q= 128
CONTROL WORD 0 IS   R= 7
CONTROL WORD 1 IS  T1= 131

DO YOU WANT COMPLETE INSTRUCTIONS? ENTER 1=YES, 0=NO.
? 0
ENTER PORT NO IN DECIMAL
? 128
ENTER 1 FOR INTERRUPT CONTROL, 0 FOR NO.
? 1
ENTER COUNTER INPUT CHANNEL CODE
? 4
ENTER MEASUREMENT FUNCTION
? 3
ENTER TIME BASE RATE
? 7
INTERRUPT SOURCE: 1= COUNTDOWN CHAIN, 0= MEAS. COMPLETION
? 0
 INTERRUPTS ENABLED - SOURCE CODE IS  0
INPUT CHANNEL IS  4
MEASUREMENT FUNCTION IS  3
COUNT DOWN CHAIN RATE IS CODE  7


COUNTER CONTROL VALUES FOR THE ABOVE SELECTION IS:
I-O PORT ADDRESS IS Q= 128
CONTROL WORD 0 IS   R= 135
CONTROL WORD 1 IS  T1= 131
```

Figure 5-5

Programming Example: Fixed Use of COMPOSE Output

```
1 REM  PROGRAM EXAMPLE TO MAKE A FREQUENCY MEASUREMENT
2 REM INPUT: SINE, TIME BASE: 1 HZ.
10 Q=128
20 R=0
30 T1=72
40 GOSUB 100
50 GOTO10
90 REM
91 REM
92 REM    SUBROUTINE TO MAKE MEASUREMENTS IN A NON-OVERLAP MODE
99 RESET THE 88-UFC COUNTER  TO INITIAL ALL ZERO STATE
100 OUT Q,16 : OUT Q,0
109 REM  OUTPUT COMMAND WORD 0
110 OUT Q,R
119 REM  OUTPUT COMMAND WORD 1 AND START MEASUREMENT
120 OUT Q+1,T1
129 REM  TEST  88-UFC FOR BUSY - LOOP UNTIL MEASUREMENT COMPLETE
130 IF(INP(Q)AND1)=0 THEN 130
134 REM LOAD LATCH WITH RESULTS.
135 OUT Q,32 : OUT Q,0
139 REM PRINT OUTPUT IN INTEGER FORM
140 FOR J=1 TO 9
150 PRINTINP(Q+1)-240;
159 REM  SCAN TO NEXT DIGIT
160 OUT Q,64 : OUT Q,0
170 NEXT J
180 PRINT
190 RETURN
OK
RUN
 0  2  5  1  5  1  9  8  3
 0  2  5  1  5  1  9  8  0
 0  2  5  1  5  1  9  7  7
 0  2  5  1  5  1  9  8  0
```

Figure 5-6

Programming Example: Fixed Use of COMPOSE Output

```
1 REM  PROGRAM EXAMPLE TO MAKE A FREQUENCY MEASUREMENT
2 REM INPUT: SINE, TIME BASE: 1 HZ.
10 Q=128
20 R=0
30 T1=72
31 OUT Q,16 : OUT Q,0
32 OUT Q,R  : OUT Q+1,T1
33 IF(INP(Q)AND1) THEN 33
40 GOSUB 100
50 GOTO 40
90 REM
91 REM
92 REM  SUBROUTINE TO MAKE MEASUREMENTS IN AN OVERLAP MODE.
100 OUT Q,16+R  :  OUT Q,R
120 OUT Q+1,T1
140 FOR J=1 TO 9
150 PRINTINP(Q+1)-240;
160 OUT Q,64+R : OUT Q,R
170 NEXT J
180 PRINT
185 IF (INP(Q) AND1)=0 THEN 185
186 OUT Q,32+R : OUT Q,R
190 RETURN
OK


RUN
 7  7  7  7  7  7  7  7  0
 0  1  0  0  7  4  2  6  7
 0  1  0  0  6  8  5  8  6
 0  1  0  0  7  2  6  9  1
 0  1  0  0  7  0  1  4  3
 0  1  0  0  6  8  4  4  2
```

Figure 5-7

Programming Example: Variable Use of COMPOSE Output

```
200 PRINT"ENTER FUNCTION,Q,R,T1"
201 INPUT F,Q,R,T1
202 S=9
300 ON F GOSUB 1000,2000,3000,2020
310 GOTO300
999 REM FREQUENCY MEASUREMENT
1000 GOTO1039
1039 OUT Q,32:OUT Q,0
1040 OUT Q,16:OUT Q,R
1041 OUT Q+1,T1
1042 JM=S-1
1043 IS=0
1050 FOR J=1 TO S
1052 XX=INP(Q+1)-240
1053 PRINTXX;
1054 IS=IS+XX*10↑JM
1055 OUT Q,64 : OUT Q,0
1057 JM=JM-1
1060 NEXT J
1061 PRINT"    ";IS
1065 IF(INP(Q)AND1)=0 THEN 1065
1080 RETURN
1999 REM COUNT
2000 GOTO2010
2010 OUT Q,16:OUT Q,R
2015 OUT Q+1,T1
2020 OUT Q,32
2021 OUT Q,0
2050 FOR J=1 TO S-1
2060 XX=INP(Q+1)
2061 PRINTXX-240;
2070 OUT Q,64+R:OUT Q,R
2090 NEXT J
2100 PRINTINP(Q+1)-240
2119 F=4
2120 RETURN
2999 REM PERIOD MEASUREMENT
3000 GOSUB 1000
3010 T1 = T1 -1
3020 GOSUB1000
3025 T1 =T1+1
3030 RETURN
```

Figure 5-8

Sample Execution (Figure 5-7 Program)

```
RUN
ENTER FUNCTION,Q,R,T1
? 1,128,0,72
 0  1  0  3  7  6  8  0  6        1.03768E+07
 0  0  0  9  6  7  9  4  3        967943
 0  0  0  9  6  6  2  9  3        966293
 0  0  0  9  6  5  2  6  6        965266
 0  0  0  9  5  9  8  2  6        959826
 0  0  0  0  0  0  0  0  0        0
 0  0  0  0  0  0  0  0  0        0
 0  0  0  6  9  8  5  4  4        698544
 0  0  1  8  0  2  7  8  0        1.80278E+06
 0  0  1  8  0  8  2  4  7        1.80825E+06
 0  0  1  8  0  7  4  1  4        1.80741E+06
 0  0  1  8  0  8  7  7  2        1.80877E+06
 0  0  0  0  0  0  0  0  0        0
 0  0  0  0  0  0  0  0  0        0
 0  0  0  2  1  1  7  0  7        211707
 0  0  0  2  1  2  3  4  7        212347
 0  0  0  2  7  0  0  2  0        270020
 0  0  0  2  7  3  1  7  5        273175
 0  0  0  2  7  3  8  5  5        273855
 0  0  0  2  7  5  7  9  7        275797
 0  0  0  0  1  1  0  0  7        11007
 0  0  0  0  1  1  0  2  8        11028
 0  0  0  0  1  1  0  3  4        11034
 0  0  0  0  1  1  0  3  5        11035
 0  0  0  0  1  1  0  3  5        11035
 0  0  0  0  1  1  0  3  5        11035
```

Figure 5-9

Program Source and Sample Execution
to Save Counter Output in BCD

```
1 REM PROGRAM EXAMPLE TO SAVE RESULTS IN BCD.
5 DIM F(9)
10 Q=128
20 R=0
30 T1=72
40 GOSUB 100
50 GOTO40
100 OUT Q,16 :OUT Q,0
110 OUT Q,R
120 OUT Q+1,T1
130 IF (INP(Q)AND1)=0 THEN 130
135 OUT Q,32 : OUT Q,0
140 FOR J= 1 TO 9
150 F(J)=INP(Q+1)-240
160 OUT Q,64 : OUT Q,0
170 NEXT J
200 REM CONVERT BCD TO INTEGER.
201 JM=8
210 IS=0
220 FOR J= 1 TO 9
230 IS=IS+F(J)*10↑JM
240 JM=JM-1
250 NEXT J
300 FOR J=1 TO 9
310 PRINTF(J);
320 NEXT J
330 PRINT;
340 PRINTIS
350 RETURN
OK
RUN
 0  0  0  0  1  1  0  4  1  11041
 0  0  0  0  1  1  0  4  1  11041
 0  0  0  0  1  1  0  4  0  11040
 0  0  0  0  0  1  1  0  3  1103
 0  0  0  0  0  1  1  0  3  1103
 0  0  0  0  3  7  3  6  0  37360
 0  0  0  1  1  1  6  2  1  111621
 0  0  0  4  7  0  7  5  8  470758
 0  0  1  0  5  2  0  6  4  1.05206E+06
 0  0  1  0  5  1  4  0  1  1.0514E+06
```

Figure 5-10

BCD Arithmetic Subroutines

```
500 DIM A1(9),A2(9),A3(18),A4(18)
4999 REM BCD ADD ROUTINE
5000 K=9
5010 FOR J=1 TO 9
5020 A3(K)=A3(K)+A2(K)+A1(K)
5025 IF A3(K)>9 THEN A3(K-1)=1
5030 IF A3(K)>9 THEN A3(K)=A3(K)-10
5040 K=K-1
5050 NEXT J
5060 RETURN
5999 REM BCD SUB. ROUTINE
6000 K=9
6010 FOR J=1 TO 9
6020 IF A1(K)-A2(K)>=0THEN 6040
6030 A1(K)=A1(K)+10:A1(K-1)=A1(K-1)-1
6040 A3(K)=A1(K)-A2(K)
6050 K=K-1
6060 NEXT J
6070 RETURN
6999 REM BCD MULT. ROUTINE
7000 K=9
7010 FOR J=1 TO 9
7020 X=A2(K)
7030 A=(K+9)
7035 D=9
7037 W=0
7040 FOR M=1 TO9
7050 A4(A)=A1(D)*X
7055 A4(A)=A4(A)+W
7057 W=0
7060 IF A4(A)<10 THEN7080
7070 W=INT(A4(A)/10)
7075 A4(A)=A4(A)-(INT(A4(A)/10)*10)
7080 A=A-1
7085 D=D-1
7090 NEXT M
```

Figure 5-10

BCD Arithmetic Subroutines

```
500 DIM A1(9),A2(9),A3(18),A4(18)
4999 REM BCD ADD ROUTINE
5000 K=9
5010 FOR J=1 TO 9
5020 A3(K)=A3(K)+A2(K)+A1(K)
5025 IF A3(K)>9 THEN A3(K-1)=1
5030 IF A3(K)>9 THEN A3(K)=A3(K)-10
5040 K=K-1
5050 NEXT J
5060 RETURN
5999 REM BCD SUB. ROUTINE
6000 K=9
6010 FOR J=1 TO 9
6020 IF A1(K)-A2(K)>=0THEN 6040
6030 A1(K)=A1(K)+10:A1(K-1)=A1(K-1)-1
6040 A3(K)=A1(K)-A2(K)
6050 K=K-1
6060 NEXT J
6070 RETURN
6999 REM BCD MULT. ROUTINE
7000 K=9
7010 FOR J=1 TO 9
7020 X=A2(K)
7030 A=(K+9)
7035 D=9
7037 W=0
7040 FOR M=1 TO9
7050 A4(A)=A1(D)*X
7055 A4(A)=A4(A)+W
7057 W=0
7060 IF A4(A)<10 THEN7080
7070 W=INT(A4(A)/10)
7075 A4(A)=A4(A)-(INT(A4(A)/10)*10)
7080 A=A-1
7085 D=D-1
7090 NEXT M
```

Figure 5-10
Cont'd

BCD Arithmetic Subroutines

```
7100 B=18
7110 FOR C=1TO18
7120 A3(B)=A3(B)+A4(B)
7130 IF A3(B)>9 THEN A4(B-1)=A4(B-1)+1
7135 IFA3(B)>9 THENA3(B)=A3(B)-10
7140 B=B-1
7150 NEXT C
7160 A4(K+9)=0
7170 K=K-1
7180 NEXT J
7190 RETURN
10000 INPUT A1(1)
10100 INPUT A2(1)
10101 GOSUB 6000
10106 PRINTA1(1),A2(1),A3(1)
```

Figure 5-11

Programming Example: BCD Subroutines

```
3 DIM A4(18)
7 DIM A3(18)
10 INPUTA1(1),A1(2),A1(3),A1(4),A1(5),A1(6),A1(7),A1(8),A1(9)
20 INPUT A2(1),A2(2),A2(3),A2(4),A2(5),A2(6),A2(7),A2(8),A2(9)
60 PRINT"DO YOU WANT TO ADD=1,SUB=2,MULT=3,DIV=4"
70 INPUT F
93 IF F=1 OR F=2 THEN S=9
97 IF F=3 OR F=4 THEN S=18
100 FOR J= 1 TO 9
110 PRINT A1(J);
120 NEXT J
125 PRINT
130 FOR J=1 TO 9
140 PRINTA2(J);
150 NEXT J
160 PRINT
165 ONF GOSUB 5000,6000,7000,8000
170 FOR J=1 TO S
180 PRINTA3(J);
190 NEXT J
200 STOP
```

Figure 5-11
Cont'd

Programming Example: BCD Subroutines

```
RUN
? 1,2,3,4,5,6,7,8,9
? 9,8,7,6,5,4,3,2,1
DO YOU WANT TO ADD=1,SUB=2,MULT=3,DIV=4
? 1
 1  2  3  4  5  6  7  8  9
 9  8  7  6  5  4  3  2  1
 1  1  1  1  1  1  1  1  0
BREAK IN 200
OK
RUN
? 1,2,3,4,5,6,7,8,9
? 9,8,7,6,5,4,3,2,1
DO YOU WANT TO ADD=1,SUB=2,MULT=3,DIV=4
? 2
 1  2  3  4  5  6  7  8  9
 9  8  7  6  5  4  3  2  1
 1  3  5  8  0  2  4  6  8
BREAK IN 200
OK
RUN
? 1,2,3,4,5,6,7,8,9
? 9,8,7,6,,5,4,3,2
DO YOU WANT TO ADD=1,SUB=2,MULT=3,DIV=4
? 3
 1  2  3  4  5  6  7  8  9
 9  8  7  6  0  5  4  3  2
 0  2  1  9  2  6  5  9  5  4  3  3  6  7  7  8  4  8
BREAK IN 200
OK
```

Figure 5-11
Cont'd

Programming Example: BCD Subroutines

```
1 REM BCD COMPARE SUBROUTINE
2 REM
3 REM  FL=1 FOR A1>A2
4 REM  FL=2 FOR A1<A2
5 REM  FL=0 FOR A1=A2
6 REM
9 DIM A1(9),A2(9)
10 FL=0
20 FOR I=1 TO 9
30 IF A1(I)>A2(I) THEN 70
40 IF A2(I)>A1(I) THEN 90
50 NEXT I
60 RETURN
70 FL=1
80 RETURN
90 FL=2
95 RETURN
```

```
1 REM  BCD COUNTER READING IS CONVERTED TO FLOATING POINT
2 REM  BCD ARRAY A5 HAS COUNTER INPUT.
3 REM  A FLOATING POINT NUMBER REPRESENTING THE COUNTER INPUT
4 REM  IN MHZ. WILL BE RETURNED IN -MH-.
5 REM
6 REM   CONTROL INPUT VARIABLE (CT) IS REQUIRED WHICH SPECIFIES THE
7 REM  BCD DIGIT JUST TO THE LEFT OF THE ASSUMED DECIMAL POINT.
8 REM  -- FOR EXAMPLE IF THE COUNTER INPUT IS  0 0 1 2 3 4 5 6 9
9 REM   1.234569     THEN CT SHOULD BE SET TO 3  (I.E. BCD DIGITS
10 REM  ARE NUMBERED FROM LEFT TO RIGHT AND FROM 1 TO 9)
11 REM
100 MH=0
110 LP=CT
120 FOR I=1 TO 9
130 J=LP-I
140 MH=MH+(A5(I)*10↑J)
150 NEXT I
160 RETURN
```

Figure 5-11
Cont'd

Programming Example: BCD Subroutines

```
1 REM FP IS THE FIXED POINT VARIABLE TO BE CONVERTED TO A 9 DIGIT
2 REM BCD ARRAY.
3 REM THE FP ARRAY CANNOT EXCEED 6 DIGITS.
4 REM
10 S=FP
20 FOR I=1 TO 9
30 J=9-I
40 T=INT(S/10↑J+.000001)
50 S=S-INT(T*10↑J+.000001)
60 A5(I)=T
70 NEXT I
80 RETURN
```

Figure 5-11
Cont'd

Programming Example: BCD Subroutines

```
1 REM FP IS THE FIXED POINT VARIABLE TO BE CONVERTED TO A 9 DIGIT
2 REM BCD ARRAY.
3 REM THE FP ARRAY CANNOT EXCEED 6 DIGITS.
4 REM
10 S=FP
20 FOR I=1 TO 9
30 J=9-I
40 T=INT(S/10↑J+.000001)
50 S=S-INT(T*10↑J+.000001)
60 A5(I)=T
70 NEXT I
80 RETURN
```

Figure 5-12

Changes Required to 88-UFC Programs
Previously Listed to Allow Use
with Processor Technology BASIC5

```
1000 REM  TO USE THE 88-UFC WITH PT BASIC, CHANGE  THE
1001 REM  OUT AND INP STATEMENTS AS FOLLOWS:
1002 REM
1003 REM
1004 REM     OUT Q,X   IS CHANGED TO B=ARG(X); B=CALL(6484)
1005 REM     OUT Q+1,X   IS CHANGED TO B=ARG(X); B=CALL(6488)
1006 REM     X=INP(Q)   IS CHANGED TO X=CALL(6492)
1007 REM     X=INP(Q+1)   IS CHANGED TO X=CALL(6498)
1008 REM
1009 REM
1010 REM     PATCH PT BASIC AFTER LOADING WITH THE
1011 REM     ASSEMBLY PROGRAM LISTED ON THE FOLLOWING PAGE.
```

NOTE: Processor Technology BASIC5 users will be required to re-code the test for wait in the examples which use the logic "AND" statement to

```
1065 IF X=254 THEN 1065
```

Figure 5-13

Assembly Patch Required for Use with PTCO BASIC5

```
                      ; PATCH FOR PT BASIC TO ACCESS THE 88-UFC
                              ORG 6484
                      ; I-O PORTS
1954   00 80          T0      EQU     80H
1954   00 81          T1      EQU     81H
                      ;
                      ; WRITE ADDRESS 0
1954   7B             T6484:  MOV     A,E
1955   D3 80                  OUT     T0
1957   C9                     RET
                      ;
                      ; WRITE TO ADDRESS 1
1958   7B             T6438:  MOV     A,E
1959   D3 81                  OUT     T1
195B   C9                     RET
                      ;
                      ; READ ADDRESS 0
195C   DB 80          T6492:  IN      T0
195E   6F                     MOV     L,A
195F   26 00                  MVI     H,0
1961   C9                     RET
                      ;
                      ; READ ADDRESS 1
1962   DB 81          T6498:  IN      T1
1964   6F                     MOV     L,A
1965   26 00                  MVI     H,0
1967   C9                     RET
1968                          END

TOTAL ASSEMBLER ERRORS =    0
```

SYMBOL TABLE

A	0007	B	0000	C	0001	D	0002
E	0003	H	0004	L	0005	M	0006
PSW	0006	SP	0006	T0	0080	T1	0081
T6484	1954	T6488	1958	T6492	195C	T6498	1962

Figure 5-13
Cont'd

Assembly Patch Required for Use with PTCO BASIC5

```
:141954007BD380C97BD381C9DB806F2600C9DB816F2600C9DD
:00

031 124 4 173 323 200 311
031 130 4 173 323 201 311 333 200 157 046
031 140 4 000 311 333 201 157 046 000 311
```

5.5 PERIOD-INTERRUPT MODE DEMONSTRATION PROGRAM

The period interrupts mode program listed in Figure 5-14 operates in the period mode with interrupts enabled to occur at the end of each transition of the source input. The program has been successfully used to demodulate Amateur Radio Audio Frequency Shift Keying for Radio Teletype.

The program assumes that the AFSK signal is a very clean sine wave representation of the received signal. When the signal is received on VHF as a 2-meter RTTY repeater, it may be clean enough to operate the 88-UFC directly and yet provide a fairly low error rate. If AFSK filters and limiters are available, they should be used for improved operation and are necessary if noisy signals such as those from a HF radio receiver are used. Program operation may be verified by using a signal generator as input while varying the input frequency from mark to space and observing the computer output as it follows the frequency shift.

AFSK signals are shifted from 2125 Hz to 2295 Hz at the transmitted baud rate. The counter is set up to measure the period of this incoming signal. For the demonstration program the input is assumed to be at the sine channel. The 88-UFC interrupts the CPU at the end of each full cycle of the input signal and stores the period value in the holding register. Since the timebase rate is set at .1 ms, the period values will then be 000004357 for a space tone of 2295 Hz and 000004705 for a mark tone of 2125 Hz. The counter will measure every other cycle, making over 1000 measurements per second. Since the popular baud rate for currently approved RTTY is 66 baud, the 88-UFC will mak about 66 period measurements for each baud element since each element is about 15 ms in length. Because the counter needs only a few instructions to initialize it for each period measurement, a considerable amount of time (1 ms) is available between each period measurement to analyze the measured value.

The demonstration program provided in this section is relatively simple in that it does not do much more than is necessary to determine if the signal was a mark or a space. A more complicated program could be written if the user wanted to exclude the effects of noise and selective fading from the signal. Since the period of the signal will vary from 4357 to 4705 milliseconds, 4500 milliseconds is the approximate center point. The software program may act as a discriminater by identifying signals from 4500-4899 ms as spaces and signals from 4100-4499 ms as marks while rejecting all other values. A wider range of values may be used to allow for drift of the tones.

The mark or space output is continuously reset to the value determined by the software logic to either a mark or a space. Therefore, the rejected values are not used as output. The output bit remains in its last set position until changed by a valid value of the opposite type. Noise may appear in the output signal as short changes in the mark-space transmissions and as jitter at the start of each mark-space element. Other factors also cause a few milliseconds of jitter at each mark-space transmission. Since these changes are small compared to the interval of a single baud, this effect is easily rejected by the TTY selector magnets or by the action of a UART if used for serial/parallel conversion.

The output from the demonstration program is a single bit at a latched output port. This output is a TTL level serial representation of the original signal and for demonstrtion purposes could be connected to an indicator LED or to a logic probe. For octal TTY operation it would be connected to a switchable TTY level connector and to a teletypewriter. The output could also be connected back into a serial port of the computer with its UART strapped for a 5-bit 1.5 stop bit code, then via software the BAUDOT code could be converted to ASCII for display on the system's display device. Since the 88-UFC is operating in an interrupt driven mode, all programs will operate well within the time available to the CPU.

The program consists of two parts, the main routine, "TEST", and a subroutine to demodulate the FSK signals. For demonstration purposes, the main routine reads input port 0, tests for busy, inputs a character and input port 1 and then output this character to port 1. This reads the keyboard and echos the character to the output port. This could be replaced with a jump to itself to provide an idle loop if a TTY is used or may be replaced by a BAUDOT conversion/display program for use with a teletypewriter display.

The logic for the subroutine used to demodulate the FSK signal works as follows. The routine is entered as a result of an interrupt from location 070 and the period results are latched into the output register of the counter. The counter is then reset to zero (lines 36-46). The five leading zeros of the measured value are rejected by scanning over them without reading them (lines 47-71). A digit is then read in. If it is not a "4" the entire measurement is rejected (lines 73-75). The next digit (7) is read and is rejected if it ia zero or nine (lines 77-85). The digit is tested for a value greater than or equal to five and, if true, the output bit for the serial data is set to zero. If the digit is less than five it is set to one (lines 89-95). The counter control registers are then reloaded and interrupts are enabled. Control is returned to the main program.

Figure 5-14

88-UFC Program to Decode RTTY

```
                          ORG     38C0H
                  ;       I-O PORTS
38C0   00 80      UFCP0   EQU     80H
38C0   00 81      UFCP1   EQU     81H
38C0   00 00      T0      EQU     0
38C0   00 01      T1      EQU     1
38C0   00 06      T3      EQU     6
38C0   00 01      T2      EQU     1
                  ;       FLAGS
38C0   00 01      F1      EQU     1
38C0   00 00      ZERO    EQU     0
38C0   00 01      ONE     EQU     1
38C0   00 40      SCAN    EQU     40H
38C0   00 20      LATCH   EQU     20H
38C0   00 10      RESET   EQU     10H
38C0   00 F4      FOUR    EQU     244
38C0   00 F9      NINE    EQU     249
38C0   00 F5      FIVE    EQU     245
38C0   00 87      SETUP   EQU     87H
38C0   00 43      PERIN   EQU     43H
                  ; LOAD INTERUPT VECTOR
                          ORG     38H
0038   C3 14 38           JMP     BAUD
                  ;       MAIN LINE PROGRAM
                          ORG     3800H
3800   31 FF 03           LXI     SP,3FFH
3803   CD 14 38           CALL BAUD
3806   DB 00      TEST:   IN      T0
3808   E6 01              ANI     F1
380A   C2 06 38           JNZ     TEST
380D   DB 01              IN      T1
380F   D3 01              OUT     T1
3811   C3 06 38           JMP     TEST
                  ;
                  ; SUBROUTINE TO GET MARKS AND SPACES
3814   F5         BAUD:   PUSH    PSW
3815   C5                 PUSH    B
3816   D3 80              OUT     UFCP0
3818   3E 20              MVI     A,LATCH
381A   D3 80              OUT     UFCP0
381C   3E 00              MVI     A,ZERO
381E   D3 80              OUT     UFCP0
3820   3E 10              MVI     A,RESET
3822   D3 80              OUT     UFCP0
3824   3E 00              MVI     A,ZERO
3826   D3 80              OUT     UFCP0
```

Figure 5-14
Cont'd

88-UFC Program to Decode RTTY

```
                        ; DUMP1
3828  3E 40                     MVI     A,SCAN
382A  D3 80                     OUT     UFCP0
382C  3E 00                     MVI     A,ZERO
382E  D3 80                     OUT     UFCP0
                        ; DUMP 2
3830  3E 40                     MVI     A,SCAN
3832  D3 80                     OUT     UFCP0
3834  3E 00                     MVI     A,ZERO
3836  D3 80                     OUT     UFCP0
                        ; DUMP 3
3838  3E 40                     MVI     A,SCAN
383A  D3 80                     OUT     UFCP0
383C  3E 00                     MVI     A,ZERO
383E  D3 80                     OUT     UFCP0
                        ; DUMP 4
3840  3E 40                     MVI     A,SCAN
3842  D3 80                     OUT     UFCP0
3844  3E 00                     MVI     A,ZERO
3846  D3 80                     OUT     UFCP0
                        ; DUMP 5
3848  3E 40                     MVI     A,SCAN
384A  D3 80                     OUT     UFCP0
384C  3E 00                     MVI     A,ZERO
384E  D3 80                     OUT     UFCP0
                        ; GET DIGIT NO. 6
3850  DB 81                     IN      UFCP1
3852  FE F4                     CPI     FOUR
3854  C2 7B 38                  JNZ     EXIT
                        ; GET DIGIT NO. 7
3857  3E 40                     MVI     A,SCAN
3859  D3 80                     OUT     UFCP0
385B  3E 00                     MVI     A,ZERO
385D  D3 80                     OUT     UFCP0
385F  DB 81                     IN      UFCP1
3861  FE 00                     CPI     ZERO
3863  CA 7B 38                  JZ      EXIT
3866  FE F9                     CPI     NINE
3868  CA 7B 38                  JZ      EXIT
                        ;DIGIT NO. 7 MUST NOT BE A ZERO OR NINE
                        ;
                        ;
```

Figure 5-14
Cont'd

88-UFC Program to Decode RTTY

```
386B  FE F5                 CPI   FIVE
386D  DA 77 38              JC    MARK
3870  3E 01         SPAC:   MVI   A,ONE
3872  D3 06                 OUT   T3
3874  C3 7B 38              JMP   EXIT
3877  3E 00         MARK:   MVI   A,0
3879  D3 06                 OUT   T3
387B  3E 87         EXIT:   MVI   A,SETUP
387D  D3 80                 OUT   UFCP0
387F  3E 43                 MVI   A,PERIN
3881  D3 81                 OUT   UFCP1
3883  C1                    POP   B
3884  F1                    POP   PSW
3885  FB                    EI
3886  C9                    RET
3887                        END

TOTAL ASSEMBLER ERRORS =    0
```

Figure 5-15

RTTY Decode Program
Symbol Table, Hex Dump, and Octal Dump

SYMBOL TABLE

A	0007	B	0000	BAUD	3814	C	0001
D	0002	E	0003	EXIT	387B	F1	0001
FIVE	00F5	FOUR	00F4	H	0004	L	0005
LATCH	0020	M	0006	MARK	3877	NINE	00F9
ONE	0001	PERIN	0043	PSW	0006	RESET	0010
SCAN	0040	SETUP	0087	SP	0006	SPAC	3870
T0	0000	T1	0001	T2	0001	T3	0006
TEST	3806	UFCP0	0080	UFCP1	0081	ZERO	0000

```
:03003800C31438B6
:1E38000031FF03CD1438DB00E601C20638DB01D301C30638F5C5D3803E20D3803E00EF
:1E381E00D3803E10D3803E00D3803E40D3803E00D3803E40D3803E00D3803E40D38072
:1E383C003E00D3803E40D3803E00D3803E40D3803E00D380DB81FEF4C27B383E40D305
:1E385A00803E00D380DB81FE00CA7B38FEF9CA7B38FEF5DA77383E01D306C37B383EAC
:0F38780000D3063E87D3803E43D381C1F1FBC905
:00
```

```
070 000   061 377 003 315 024 070 333 000
070 010   346 001 302 006 070 333 001 323
070 020   001 303 006 070 365 305 323 200
070 030   076 040 323 200 076 000 323 200
070 040   076 020 323 200 076 000 323 200
070 050   076 100 323 200 076 000 323 200
070 060   076 100 323 200 076 000 323 200
070 070   076 100 323 200 076 000 323 200
070 100   076 100 323 200 076 000 323 200
070 110   076 100 323 200 076 000 323 200
070 120   333 201 376 364 302 173 070 076
070 130   100 323 200 076 000 323 200 333
070 140   201 376 000 312 173 070 376 371
070 150   312 173 070 376 365 332 167 070
070 160   076 001 323 006 303 173 070 076
070 170   000 323 006 076 207 323 200 076
070 200   103 323 201 301 361 373 311 000
070 210   000 000 000 000 000 000 000 000
```