USER HARDWARE MANUAL
INTERVAL TIMER

GEC COMPUTERS LIMITED
Elstree Way, Borehamwood, Hertfordshire.
Telephone No. 01-953-2030

Holding Company: The General Electric Company p.l.c. of England

April 1982

INTERVAL TIMER

CONTENTS

1 INTRODUCTION

The Interval Timer functions as a real time clock by reporting time at defined intervals. It consists of a single 484 board and communicates with the CPU as a peripheral device using the 16 bit version of the Normal Interface (see Interface Manual).

The Interval Timer can be housed in any Normal Interface slot in GEC 4000 series CPU crate or in any EMC, EMC/4 or Normal Interface extension crate.

The function of the Interval Timer is to inform the CPU when a previously defined time interval has elapsed by performing a Status Break Transfer on the Normal Interface. This function uses the following three Normal Interface transfers:-

(a) Program Output Transfer: used to define the time interval, it may also in addition define a new zero time reference.

(b) Program Input Transfer: used to monitor the time elapsed since the last zero time reference, and to inform the CPU if an interrupt is to be expected.

(c) Status Break Transfer: used to inform the CPU that the time interval defined by a Program Output transfer has elapsed. This transfer operation may define a new zero time reference.

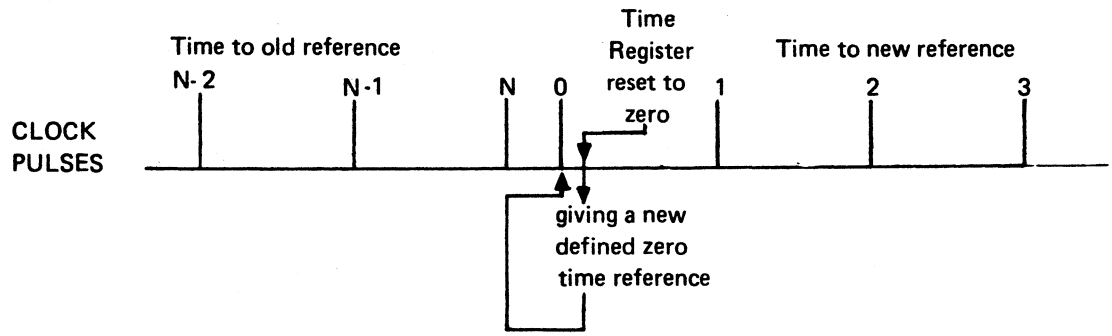
Also situated on the board is a facility for connecting an external clock, normally of a higher order of accuracy.

2.1 REGISTERS

There are three registers in the Interval Timer; the Time, Interval, and Interrupt registers.

(a) *Time Register*

This is a 12 bit register which counts the time elapsed time since the last defined zero-time reference as an integral number of ¼ milliseconds. The first clock after the Time Register is reset being counted as one, as shown below.



The new defined zero reference time can be up to a 1 μsecond before the reset time of the Register, consequently there can be an elapsed time error of up to a microsecond.

(b) *Interval Register*

This is a 12 bit register which contains the time intervals to be measured as an integral number of ¼ milliseconds. The longest interval is that which can be held in 12 bits namely 4095 ¼ milliseconds (1.02375 seconds).

(c) *Interrupt Register*

This is a cyclic 4 bit counter which is incremented every time Status Break Transfer is required. It can only be reset by taking the Active line false (i.e. by resetting the CPU).

2.2 CONTROL STATICISERS

Control of the Interval Timer's functions is by three Control Staticisers; OF, D, and R.

(a) *OF Staticiser*

This staticiser is set if the Time Register overflows by exceeding its capacity of 1.02375 seconds. When the OF Staticiser is set the content of the Time Register is indeterminate. The staticiser is reset when a new zero time reference is defined.

(b) *D Staticiser*

This staticiser is set to indicate that an interrupt is required after the time set into the Interval Register has elapsed. It is reset after an interrupt has occured and no further interrupts can occur until it is set.

(c) *R Staticiser*

This staticiser is set to operate the repeat interrupt feature of the Interval Timer. When set it causes the D Staticiser to remain set after an interrupt so that interrupts occur on every occasion the Time Register Content equals the Interval Register content. (The Time Register is set at zero time by an interrupt or by a Program Output Transfer).

3 PROGRAM OUTPUT TRANSFERS

A Program Output Transfer is used to define an interval to be timed by loading the Interval Register. It may also be used to define a new zero time reference by resetting the Time Register.

3.1 FORMAT

A 16 bit data transfer is used to perform the Program Output; the format on the INFO OUT lines is:--

line number: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

content: M N X X ----------INTERVAL---------- X = unused

If the M bit is 0 the last defined zero time reference is to be used, if the M bit is 1 a new zero time reference is to be defined.

If the N bit is 0 a single time elapsed interrupt is called for, if the N bit is 1 repetitive time elapsed interrupts are called for.

Bits 4 to 15 define the time interval to be measured as an integral number of ¼ milliseconds.

3.2 OUTPUT WITH M = 0 : N = 0

With M = 0: N = 0 a single interrupt occurs when the interval called for in INFO OUT/ 4-15 is equal to the Timer Register content using the previously defined zero time reference.

This Program Output transfer causes:-

INFO OUT / 4-15 => Interval Register

1 => D Staticiser

0 => R Staticiser

Example: INFO OUT = 00000001 00000000

Result: A single interrupt is generated 64 milliseconds after the zero time reference.

Possible errors: the content of the Time Register is greater than the content of the Interval Register, and the Time Register had overflowed setting the OF Control Staticiser. (See section 5. 4)

3.3 OUTPUT WITH M = 1 : N = 0

With M = 1 : N = 0 a single interrupt occurs when the interval called for in INFO/4-15 is equal to the Time register content using the zero time reference defined by this transfer.

This Program Output transfer causes:-

INFO OUT / 4-15 => Interval Register

1 => D Staticiser

0 => R Staticiser

0 => Time Register

Example: INFO OUT = 10000000 01100000

Result: a single interrupt is generated 24 milliseconds after the transfer is obeyed.

3.4 OUTPUT WITH M = 0 : N = 1

With M = 0 : N = 1 an interrupt occurs every time the interval called for in INFO OUT/4-15 is equal to the Time Register content using the previously defined zero time reference for the first interrupt.

This Program Output Transfer causes:-

INFO OUT/ 4-15 => Interval Register

1 => D Staticiser

1 => R Staticiser

Example: INFO OUT = 01001110 00000000

Result: an interrupt occurs every 896 milliseconds after the previously defined zero time reference.

Possible errors: the content of the Time Register is greater than the content of the Interval Register, and the Time Register had overflowed setting the OF Control Staticiser. (See Section 5.4).

3.5 OUTPUT WITH M = 1 : N = 1

With M = 1 : N = 1 an interrupt occurs every time the interval called for in INFO OUT / 4-15 is equal to the Time Register content.

This Program Output Transfer causes:-

INFO OUT/ 4-15 => Interval Register

1 => D Staticiser

1 => R Staticiser

0 => Time Register

Example: INFO OUT = 11000000 11111111

Result: an interrupt occurs every 63¾ milliseconds after the first interrupt which occurs 63¾ milliseconds after the first transfer is obeyed.

4 PROGRAM INPUT TRANSFERS

A Program Input Transfer is used to read the elapsed time since the last zero time reference. This is achieved by inputting the Time Register content.

4.1 FORMAT

A 16 bit data transfer is used to perform the Program Input; the format on the INFO IN lines is:-

```
line number: 0  1  2  3  4 5 6 7 8 9 10 11 12 13 14 15
   content: OF  D  X  X  .....TIME.....          X = Unused
```

4.2 INPUT FLAGS AND TIME

Two flags are input with the Time Register content to indicate the state of the Interval Timer at the transfer time. INFO IN 0 is set to the state of the OF Control Staticiser. INFO IN 1 is set to the state of the D Control Staticiser:

This Program Input Transfer causes:-

(a) Time Register => INFO IN/4-15

(b) D Staticiser => INFO IN 1

(c) OF Staticiser => INFO IN 0

If OF = 1 then the Time Register has overflowed and INFO IN/4-15 is invalid. If OF = 0 then INFO IN/4—15 indicates the elapsed time since the last zero time reference as an integral number of ¼ milliseconds.

If D = 1 the Interval Timer is active. If D = 0 the Interval Timer is inactive.

The Time Register content placed on the INFO IN lines can be up to a ¼ millisecond lower than the real elapsed time because a Program Input Transfer can occur between clock pulses.

INFO IN/4-15 = Elapsed time in ¼ milliseconds (with a tolerance of zero to minus one ¼ millisecond).

Example: INFO IN = 01000001 00000000

Meaning: Either between 64 and 64¼ milliseconds has elapsed since the last zero time reference.

5 STATUS BREAK TRANSFERS

A Status Break Transfer is used by the Interval Timer to inform the CPU that either an error condition has occured or the time interval to be measured has elapsed. With both conditions a 4 bit interrupt count is input to the CPU.

5.1 FORMAT

The status byte format on the INFO IN lines is:-

bit number: 0 1 2 3 4 5 6 7
content: E F 0 0 I_0 I_1 I_2 I_3

5.2 FLAGS AND INTERRUPT COUNT

The flags E and F are coded to indicate the type of Status Break Transfer being performed:-

E F	Meaning
0 0	not defined
0 1	time elapsed
1 0	Time Register overflow error interrupt
1 1	Time Register > Interval Register error interrupt

The contents of the Interrupt Register are loaded into bits 4 to 7 of the status byte. Initially the register is set to 0000 by the CPU. Subsequently the register counts interrupts, after the first interrupt reading 0001 and after the fifteenth1111.The sixteenth interrupt starts the cycle again with the counter at 0000.

5.3 TIME ELAPSED INTERRUPTS

A time elapsed interrupt is initiated if:-

Time Register content = Interval Register content

and OF Staticiser - = 0

and D Statlciser = 1

Under these circumstances the effect on the registers and Staticisers is:-

Initiate Status Break Transfer

0 => Time Register

R Staticiser => D Staticiser

Setting to zero of the Time Register defines a new zero time reference. The transfer of the R Staticiser content to the D Staticiser has the following effect. If R = 0 then D = 0 and further interrupts are inhibited and the Interval Timer is inactive until receipt of the next Program Output Transfer. If R = 1 then D = 1 and further interrupts occur when the three conditions listed at the paragraph head are met.

5.4 ERROR INTERRUPTS

There are two errors, (a) Time Register overflow, and (b) Time Register content greater than the Interval Register content.

(a) The OF Control Staticiser is set if the Time Register overflows after a Program Output Transfer. The effect of an overflow on the Interval Timer is:-

Initiate Status Break Transfer

0 = > D Staticiser

The setting of the D Control Staticiser to zero is performed when the Status Break Transfer is actioned by the IOP inhibiting any further interrupts until the next Program Output Transfer is received. This error interrupt does not define a new zero time reference.

(b) If at any time when the D Control Staticiser is set to 1 the Time Register content exceeds the Interval Register content an error condition exists. The effect on the Interval Timer and the action taken is the same as described for Time Register overflow in (a) above.

6 RESET CONDITION

If the ACTIVE line on the Normal Interface (see Interface Manual) is taken false the Interval Timer enters the reset state. The effect of ACTIVE going false is:-

Time Register content undefined

Interval Register content undefined

1 = > OF Staticiser

0 = > D Staticiser

0 = > R Staticiser

00 00 = > Interrupt Register

7 NORMAL INTERFACE USAGE

7.1 WAY NUMBER

The Interval Timer uses a single Way Number in the range 0 to 255 for all transfers. The actual Way Number is determined by patching on the Interval Timer board. Way Number 255 is not available on an EMC/4.

7.2 READY NUMBER

The Interval Timer uses a single Ready Number to signal a Status Break Transfer request to the IOP. Allocation of the Ready Number is independent of the Way Number and lies in the range 0 to 15. The actual Ready Number is determined by patching on the Interval Timer board.

7.3 INTERACTION OF INTERFACE TRANSFERS

If a Program Output Transfer occurs when the Interval Timer has requested a Status Break Transfer which has not been actioned by the IOP, the interrupt request is cancelled (the READY line is reset to zero if necessary).

All interrupts are inhibited during a Program Output Transfer. After the completion of a Program Output Transfer interrupts are permitted but depend only on the state of the Interval Timer as a result of the transfer and not on any state occurring prior to or during a Transfer.

8 PRECISION CLOCK

The 4 k Hz clock pulses which increment the Time Register are generated by dividing the pulse train from a crystal controlled oscillator. The oscillator may drift due to ageing and temperature.

The total tolerance is equivalent to a loss or gain of less than one second every 24 hours.

The 'on board' 8.192 MHz crystal controlled oscillator may be inhibited, prior to being divided down, and an external clock may be injected in its place.

The external clock may be of higher accuracy, or a different frequency: but, its output needs to be at normal TTL logic level and its frequency within the range 5 to 12 MHz.

Board connections are as follows:

Connect pin 6A31 to 6B31, to inhibit the 'on board' clock.

Connect the external clock to 6B26 (INJECT CK) and 6A26 (OV).