DRAFT 1027 AC

Bell Laboratories

subject: PDP-11 32-bit time-of-year clock

date: March 6, 1974

from: Rudd H. Canaday

MEMORANDUM FOR FILE

The Time-of-Year (TOY) clock is a 32-bit binary clock with a variable tick rate. It is designed to connect to a PDP-11 UNIBUS and be controlled and read by a PDP-11. It occupies one standard system unit (24 card slots) and is built of standard DEC M-series logic cards. The TOY clock has three addressable registers. Two 16-bit read/write registers comprise the 32-bit binary clock. The third register is an 8-bit write-only register containing the Tick Count (TC). The 32-bit binary clock will be incremented every TC/30 seconds for $1 \le TC \le 2^8-1$. If TC = 0 the tick rate will be 1/60 sec. For use with UNIX, the TC register should be loaded with the value 30 to produce one "tick" per second.

The clock is set or read by writing or reading the two 16-bit clock registers. These are byte or word addressable. The TOY clock will continue to keep time as long as power is up (it uses the existing PDP-11 power supplies (+5V) and derives its time signal from the 60 cps line. The TOY clock is not affected by system initialization.

The parts for the clock cost about $600. A complete wirelist is available from the author. The parts cost includes a wire-wrappable backplane (the BB-11 system interfacing-unit) from DEC. The prototype is scheduled for completion in early May, 1974.

Word Count: Contains the 2s - complement of the number of words to transfer

Data: Data written into this register can be read by "B". Reading this register is actually reading "B"s data register.

Status & Control Register - "A" side

Bit	15	14	13	12	11	10	09	08
R/W	R	R/C	R	R/W	R	R	R	R/W
Name	ERROR	A-NEX	B-ATTN	A-MAINT	B-INTR	B-RCV	B-IDLE	A-CYCLE

Bit	07	06	05	04	03	02	01	00
R/W	R	R7W	R/W	R/W	R/W	R/W	R/W	W
Name	A-IDLE	A-IE	AA17	AA16	A-B-IE	A-RCV	SIGTOB	GO

ERROR

ERROR - Set by interrupt request from B or error in A. Causes interrupt of A if Bit 06 = one. Resets itself

A-NEX - Set by attempt to address invalid device by A. Sets ERROR. Must be reset by loading with zero.

B-ATTN- Set by interrupt request from B or invalid operation (A&B both attempt to receive or to transmit at the same time). Sets ERROR. Resets itself.

A-MAINT-Puts A in maintenance mode. Must be zero for normal operation.

B-INTR -Set by explicit interrupt request from B (=Bs control bit 01). Sets B-ATTN and ERROR. Can be cleared only by clearing Bs control bit 01.

B-RCV - Reports whether B is in receive or transmit mode. Equals B's control bit 02

B-IDEE -Reports whether B is idle or active.

A-CYCLE - If set, causes A to do immediate bus access when GO is set. Normally must be zero.

A-IDLE - Reports whether A is idle or active. Set by GO (A control bit 00). Reset by completion of transfer, by error condition in A, or by interrupt request from B.

A-IE - Enables A to be interrupted. If set, A will be interrupted by:

1. A word count increments to zero during transmission.
2. A addresses invalid device
3. A address overflows out of address bit 15 during transmission
4. B sets its control bit 01
5. B goes idle with B control bit 03 set
6. A and B are both active (Control bits 09 = zero and 07 = zero) and are both trying to receive or both trying to transmit (Control bits 10 and 02 have the same value).

AA17 Bits 16 and 17 of the bus address (the data source or
AA16 destination address) used by A.

A-B-IE (A to B interrupt enable) If set, B's control bit 15 (ERROR) will be set when A goes from active to idle state.

A-RCV If set, A will receive data from B. If zero, A will transmit to B. B's bit 02 must be set the opposite of A's bit 02.

SIGTOB When set, causes an immediate interrupt of B if B's control bit 06 (interrup enable) is set. Also forces B to remain in the idle state until reset. If SIGTOB is set and B's control bit 06 is set, every attempt by B to issue GO will cause B to interrupt.

GO Causes A to go from idle to active (unless an error condition exists or B's control bit 01 is set). When both A and B become active, transmission will begin.

III. How to control the MMTU:

Data may be transferred between A and B via the DATA registers in two ways.

1. A and B may read each other's data registers by a normal bus reference. This provides a two-way path (A and B's data registers may contain different data).

2. The "transmit" command may be set up on one side and the "receive" command on the other side.

The following discussion will show how to set up a transmission from A to B. This requires a command in A to transmit and in B to receive. We assume that initially both sides are "idle". The two commands may be issued in either order. The transfer will begin when both sides are non-idle.

To set up A to transmit:

1. Load the low order 16 bits of the beginning address of the buffer to be read from in A's address register

2. Load the buffer size into As word count register (2s-complement)

3. Load A's status and control register as follows:

Bit 06 - one - to enable interrupts

Bits 05-04 - XX - high order bits of B's address

Bit 03 - Controls B interrupting A when B is done. See "A" discussion above

Bit 02 - one - to make B receive

Bit 01 - zero

Bit 00 - one - "GO" - B becomes non-idle

IV. Error Conditions and other interrupts:

The following conditions will cause A to generate an interrupt on its bus if its control bit 06 = one:

1. A address non-existent
2. A address overflow (Bit 16 of A-address will not increment during transfer - attempt to generate a carry out of bit 15 generates error).
3. A word count increments to zero (normal completion)
4. B control bit 01 = one
5. B terminates transfer with B control bit 03 = one
6. A and B both try to transmit or both try to receive at the same time.

The cause of an A bus interrupt can be determined by looking at A's registers:

	Status bit				Address	Word
Condition Code	15	14	11	10-02	Register	Count
1	1	1	0	≠	≠0	X
2	1	0	0	≠	0	X
3	0	0	0	≠	≠0	0
4	X	0	1	X	≠0	X
5	X	0	0	≠	≠0	≠0
6	X	0	0	=	≠0	X

V. Additional Considerations

1. Word counts on A and B sides need not be the same. Incomplete tr transfers on either side may be aborted, or may be left hanging ready to continue. (control bit 03)

2. Transfer will begin when both sides are ready. Timing synchronization for issuing "GO" commands need not be considered.

3. Receive - transmit synchronization must be effected by the software (i.e. - software must keep track of who is transmitting and who is receiving). Three features of the MMTU aid in this:

   a. If both sides initiate the same operation an interrupt is generated

   b. Each side may observe the state (rcv/xmit, idle/non-idle) of the other side

   c. Whenever transmission is not actually taking place the two-way data path may be used for additional status information.

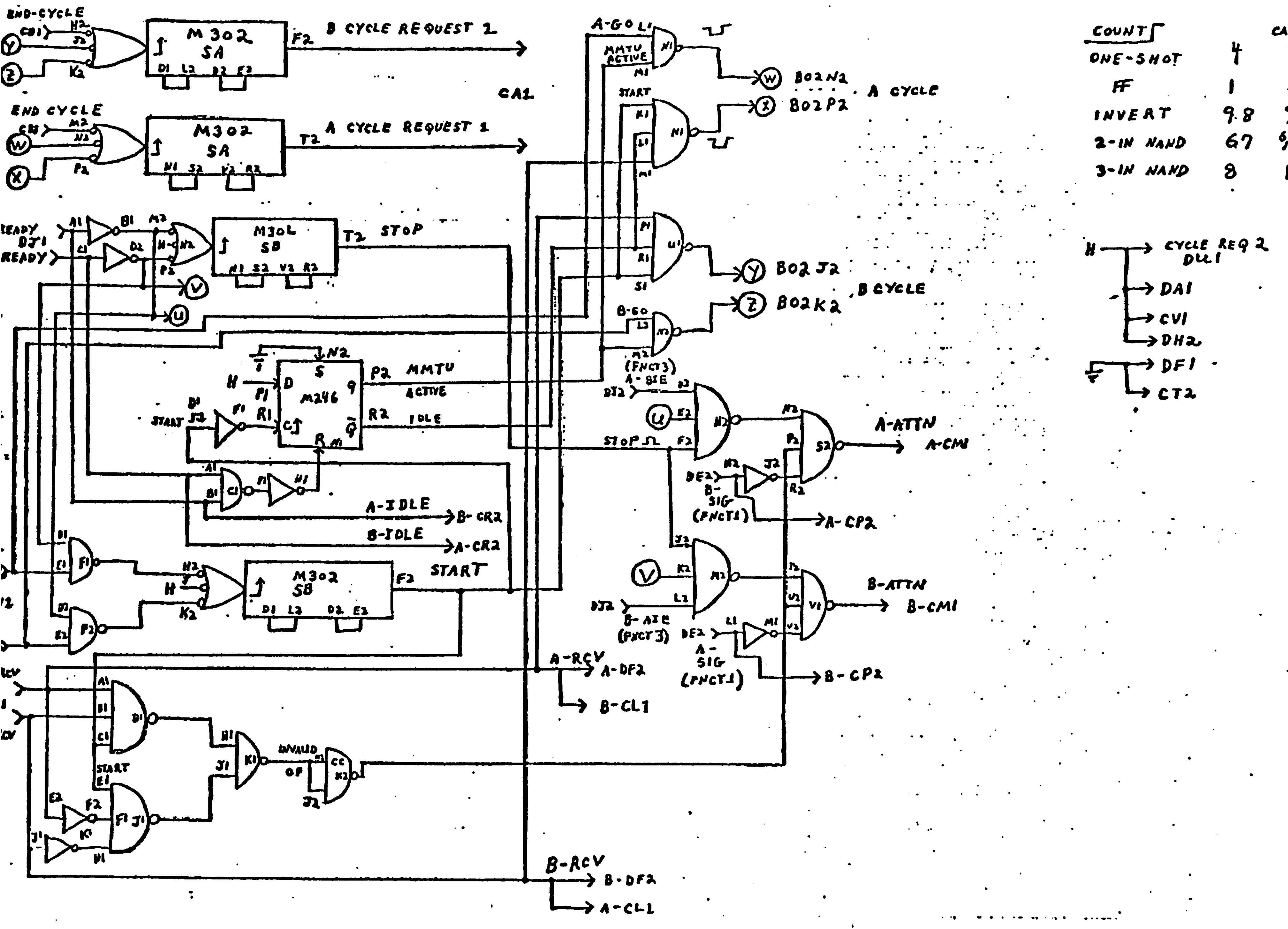
   d. While transmitting from A to B the data in B's data word is not affected. (This is the data word that B writes and A reads). Thus it can be used to hold status information. It is unclear whether it can be read or written during a transfer, but probably it can.

4. Setting A's control bit 01 (SIGTOB) will interrupt B (if interrupts enabled) and will prevent B from going active, or will force B to become idle immediately if it is active when SIGTOB is set.

VI. Timing

A complete bus cycle on one side (read a word from buffer or write a word to buffer) takes 0.5-1.0 sec. The two sides are never active at the same time (Thus they can both be connected to the same bus).

A complete transfer of one word takes 1.0-2.0 sec assuming different memory banks are used. Internal delays in the MMTU will be designed to force a minimum delay of 5.0 sec per transfer to avoid hogging the bus. Thus in the worst case with both A and B connected to the same bus the MMTU will occupy 40% of the bus capacity. Actually it will never be this high, unless the bus is otherwise completely idle.



COUNT		CARDS
ONE-SHOT	4	2
FF	1	1/5
INVERT	9.8	9/16
2-IN NAND	67	6/13
3-IN NAND	8	1