THEORY OF OPERATION

09826-66514 CPU BOARD
& 09826-66515

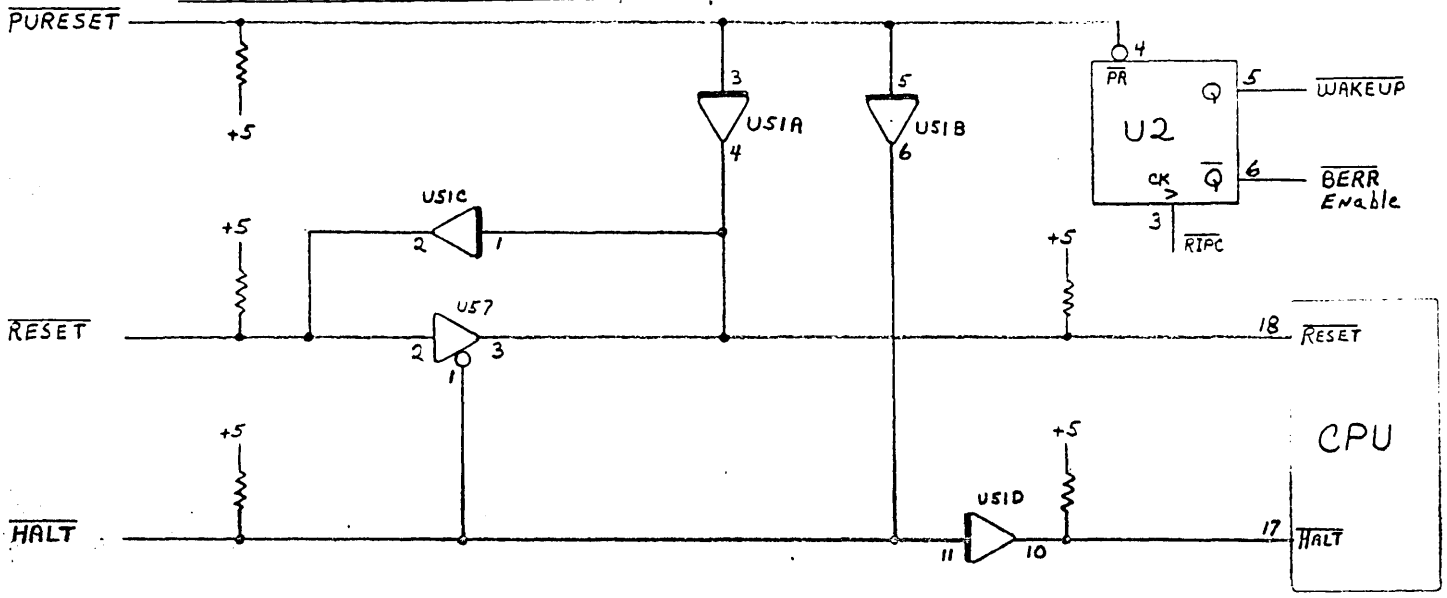
This document assumes that the reader is familiar with the operation and requirements of the Motorola MC68000 Microprocessor.

When reading this Theory of Operation, the schematic (HP Drawing 09826-90300-4), Timing diagrams (HP Drawing 09826-90300-7), Block diagram (HP Drawing 09826-90300-6,) and the Motorola MC68000 Data Manual are useful references.

GENERAL DESCRIPTION:

The CPU Board contains the CPU chip (68000), control circuitry, Boot Roms, and 64K of Auto-Located RAM. In addition to the standard 68000 control signals, the CPU Board also supplies two special purpose control signals, $\overline{IOR}$ and $\overline{IOW}$. These two signals are used by sub-systems which utilize Intel compatible peripheral chips for controlling reads and writes.

II. POWER UP CONTROL, RESET and HALT



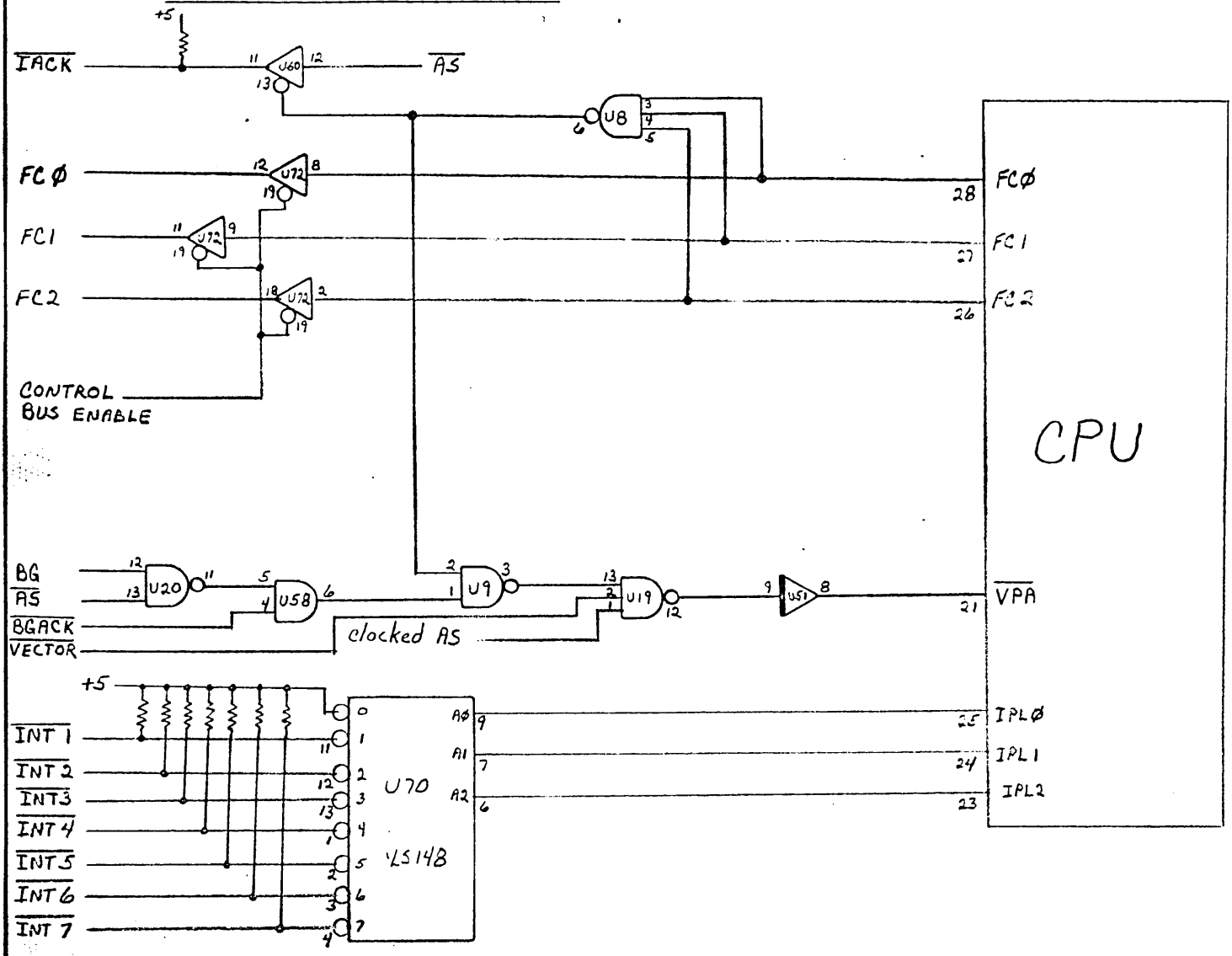
During power-up, $\overline{\text{PURESET}}$ is asserted by the power supply board. This asserts $\overline{\text{RESET}}$ and $\overline{\text{HALT}}$ to the system bus and the processor through U5, A + B + C and presets U2. $\overline{\text{WAKEUP}}$ from U2 is negated, inhibiting chip select (See RAM Access section). $\overline{\text{BERR}}$ Enable is negated allowing suppression of the first $\overline{\text{BERR}}$. (See RAM Auto Locate section). After voltages stabilize, $\overline{\text{PURESET}}$ is negated.

To the CPU, $\overline{\text{RESET}}$ and $\overline{\text{HALT}}$ are both inputs and outputs. If the CPU is generating a reset due to a $\overline{\text{RESET}}$ instruction, $\overline{\text{RESET}}$ is gated through U51 C to the system bus. This resets all external devices while the state of the processor is not affected. CPU $\overline{\text{HALT}}$ output is not used, and is prevented from being applied to the bus by U51 D.

If $\overline{\text{RESET}}$ and $\overline{\text{HALT}}$ are both asserted by an external device, $\overline{\text{HALT}}$ enables U57 so that $\overline{\text{RESET}}$ is applied to the CPU along with $\overline{\text{HALT}}$. This initiates a total system reset of the CPU and external devices.

A $\overline{\text{RESET}}$ from the system will reset the external devices but not the processor because U57 is not enabled since $\overline{\text{HALT}}$ is not asserted.

II. INTERRRUPT DECODE AND DETECT



Seven levels of interrupt are decoded into a 3 line binary by U70. Interrupt level 7 is the highest priority, level 0 is no interrupt. Internal circuitry in the CPU inhibits all interrrupts with priority levels less than or equal to the current processor priority.

FC0 - FC2 outputs indicate the type of cycle being executed by the processor. During an interrupt cycle, all three outputs are high. These are gated through U8, enabling U60. When $\overline{AS}$ is asserted, it is gated through U60, asserting $\overline{IACK}$ (Interrrupt Acknowledge) to the interrrupting device.

LTR	PC NO	APPROVED	DATE
	See page 1 for Rev.'s		
	REVISIONS		

MODEL 9826A STK NO

THEORY OF OPERATION

BY Carl Anderson DATE May 29, 1981

APPD SHEET NO 4 OF 26

SUPERSEDES DWG NO A-09826-90320-9

An interrupt requires that the processor go to some exception handling routine. The addresses of these routines are in memory locations called EXCEPTION VECOTRS. The vector numbers can be generated internally by the CPU or externally.

If the $\overline{VPA}$ input is asserted, it tells the CPU to generate the vector number internally, based upon the interrrupt level (Auto-Vector). If $\overline{VPA}$ is not asserted, the interrrupting device must supply the vector number on D0- D7. Control of the $\overline{VPA}$ input is accomplished through the $\overline{VECTOR}$ input. During the interrrupt, U19 is enabled by the interrrupt detect circuitry explained above (U8), U9 and the clocked AS. If $\overline{VECTOR}$ is high $\overline{VPA}$ is asserted indicating auto-vector. If $\overline{VECTOR}$ is asserted, the interrrupting device must supply the vector number.

See the 68000 manual for more detailed information concerning exception processing.

M6800 PERIPHERAL CONTROL

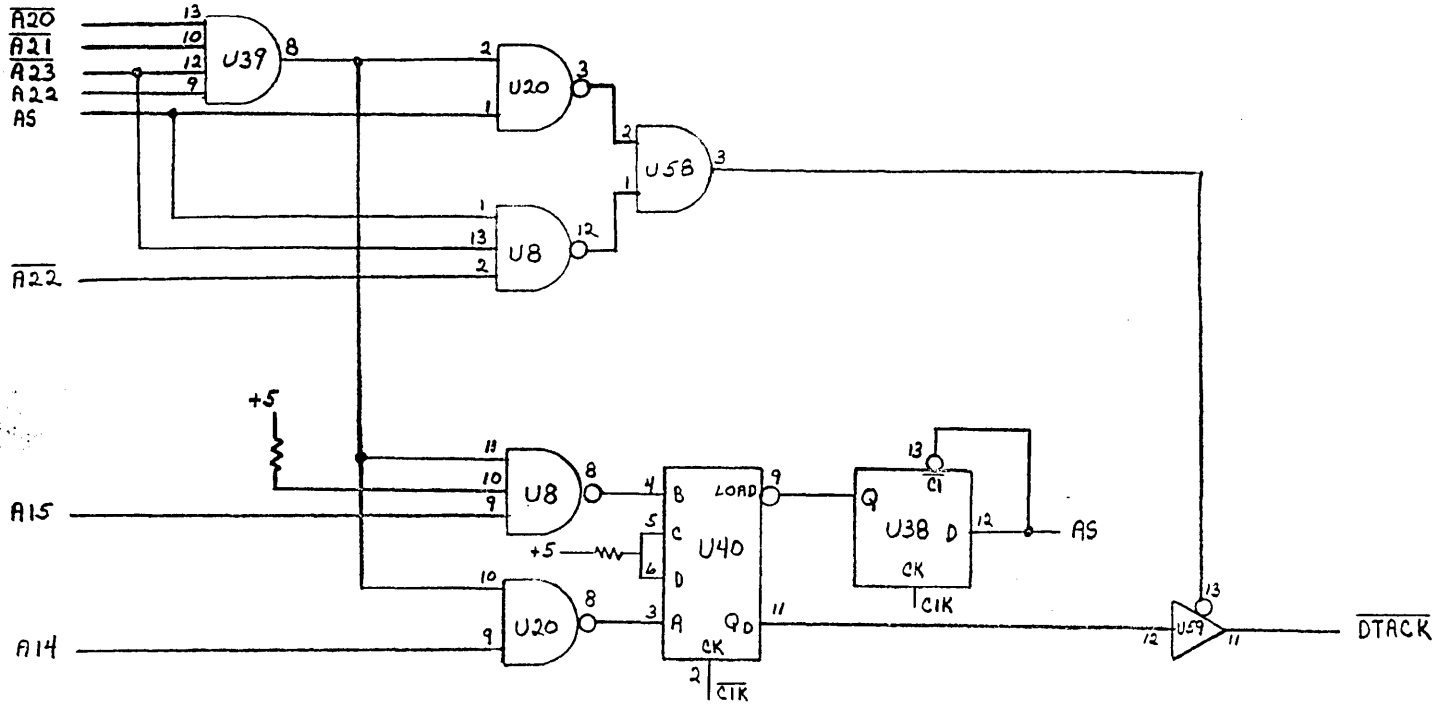
The 68000 CPU has three control signals called E (enable), VPA (Valid peripheral address), and VMA (Valid memory address), used for interfacing synchronous M6800 peripheral devices with the asynchronous MC68000.

The E output is the standard enable signal common to all M6800 peripheral devices.

The VPA input indicates that the device addressed is a M6800 family device and it also tells the processor to use automatic vectoring for an interrupt (See interrupt decode and defect).

The VMA output tell M6800 peripheral devices that there is a valid address on the address bus and the processor is synchronized to enable.

AUTO DTACK



$\overline{\text{DTACK}}$ signals the CPU that a transfer is complete. Auto DTACK is generated on the CPU board and is used to supply the $\overline{\text{DTACK}}$ for the ROM address space and for internal synchronous I/O devices.

The states of A20 through A23 determine the use of Auto DTACK in the following manner:

A20 = $\overline{\text{synchronous}}$ (if internal)

A21 = $\overline{\text{internal}}$

A22 and $\overline{\text{A23}}$ = I/O

U39 and U20 and U8 decode A20 through A23. If an internal, synchronous I/O device is addressed, the output of U58 is low, enabling the transfer of $\overline{\text{DTACK}}$ through U59.

				MODEL 9826A	STK NO
				THEORY OF OPERATION	
				BY Carl Anderson	DATE May 29, 1981
	See page1 for Rev's			APPD	SHEET NO 67 OF 326
LTR	PC NO	APPROVED	DATE		
	REVISIONS			SUPERSEDES	DWG NO A-09826-90320-9

Since different devices require different access times, A14 and A15 are used to determine the number of wait states after AS before U40 enables $\overline{\text{DTACK}}$ as shown in the following truth table:

A15	A14	Wait (cycles)	Access time (clock cycles)
0	0	0	5
0	1	1	6
1	0	2	7
1	1	3	8

$\overline{\text{AS}}$ and $\overline{\text{DATA STROBES}}$ are negated by the processor after it sees $\overline{\text{DTACK}}$ asserted. When AS is negated, U38 is cleared, allowing U40 to load and dissabling the count and U59 is inhibited negating $\overline{\text{DTACK}}$.

The following examples show how the auto DTACK wait states correspond with the processor access cycle:

Access cycle with 0 wait states (5 cycle access)

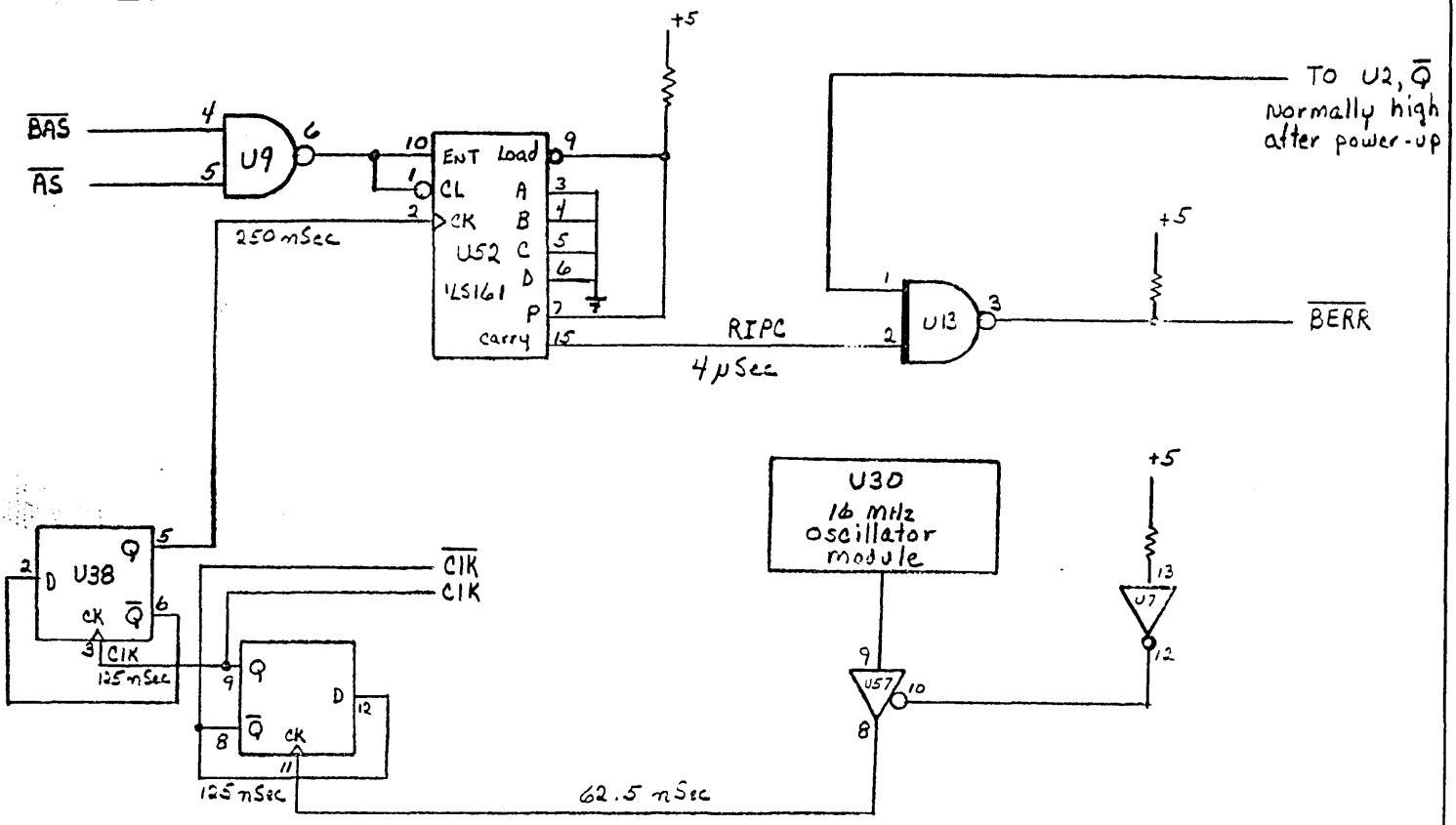
Cycle 1	Place address on bus
Cycle 2	Assert $\overline{\text{AS}}$ and $\overline{\text{Data strobes}}$
Cycle 3	Assert $\overline{\text{DTACK}}$
Cycle 4	Transfer data
Cycle 5	Terminate access by negating $\overline{\text{AS}}$, $\overline{\text{Data strobes}}$, and $\overline{\text{DTACK}}$

Access cycle with 3 wait states (8 cycle access)

Cycle 1	Place address on bus
Cycle 2	Assert $\overline{\text{AS}}$ and $\overline{\text{Data strobes}}$
Cycle 3	Wait
Cycle 4	Wait
Cycle 5	Wait

Cycle 6	Assert DTACK
Cycle 7	Transfer data
Cycle 8	Terminate access by negating $\overline{AS}$, $\overline{Data\ Strobes}$ and $\overline{DTACK}$

BUS ERROR TIMEOUT AND CLOCK



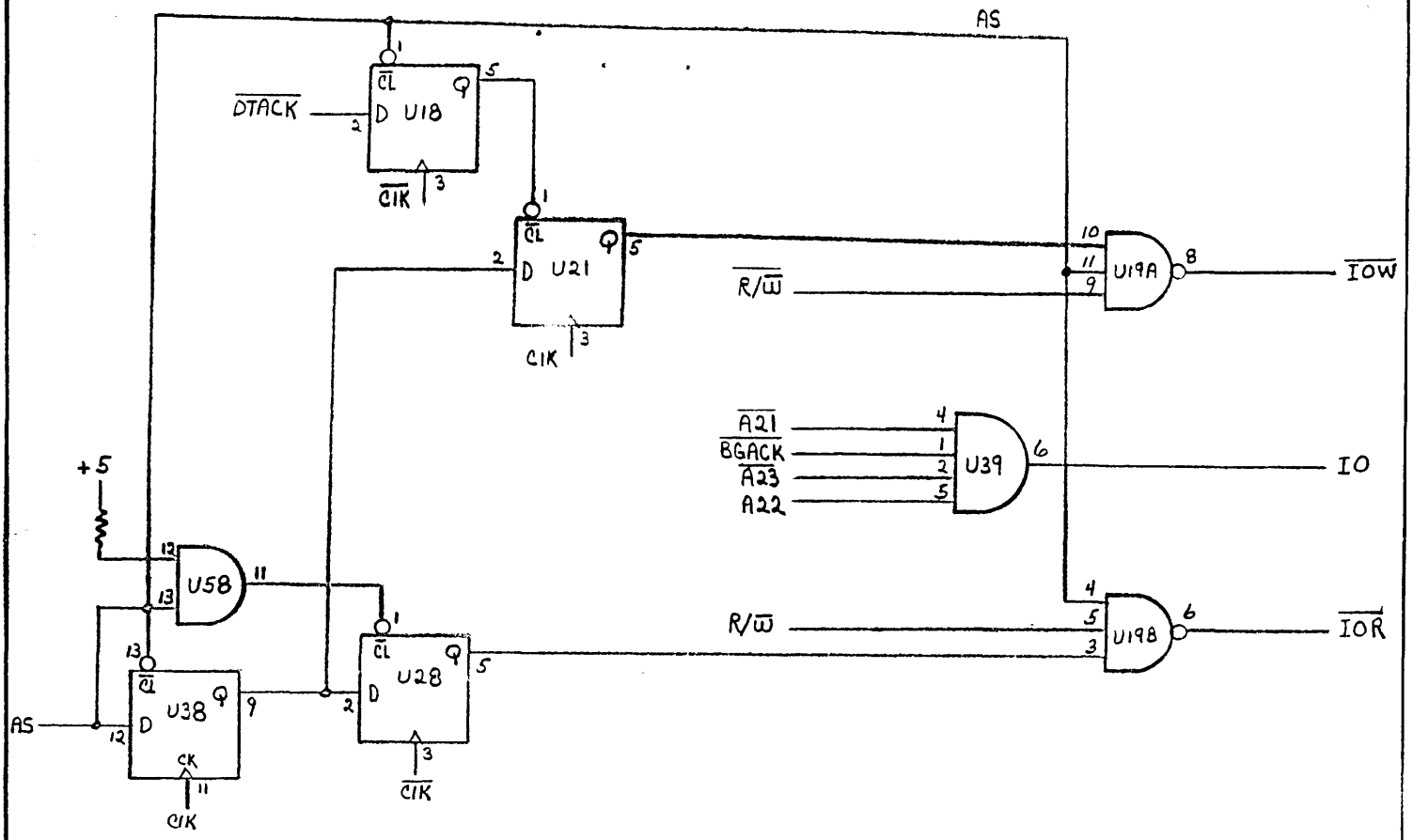
DTACK signals the processor that a data transfer is complete so that the CPU can terminate the cycle. BERR tells the CPU that DTACK has not occured during a pre-determined amount of time and it should initiate an exception processing cycle.

U30 generates a 16 MHZ clock. This is divided by 2 by U28 to produce the 8 MHZ C1K and C1K system clocks. C1K is again dividied by 2 by U38 to produce a 4MHZ clock to the bus error timeout counter, U52.

When AS and BAS are negated the clear and enable T inputs of U52 are held low, disabling the counter. When AS and/or BAS are asserted, enable T and clear go high allowing U52 to count. Since U52 is clocked

at a 250 nS rate, it takes 4 µS until RIPC is asserted. If $\overline{AS}$ and $\overline{BAS}$ are negated during this time, indicating a data transfer acknowledge has been asserted, U52 will be cleared and inhibited preventing RIPC. If $\overline{DTACK}$ is not asserted during the 4 µS that U52 is counting, RIPC is asserted and, except at power-up (see RAM Auto Locate section), is gated through U13 to produce $\overline{BERR}$. Upon seeing $\overline{BERR}$, the microprocessor goes into an exception processing sequence.

I/O CONTROL



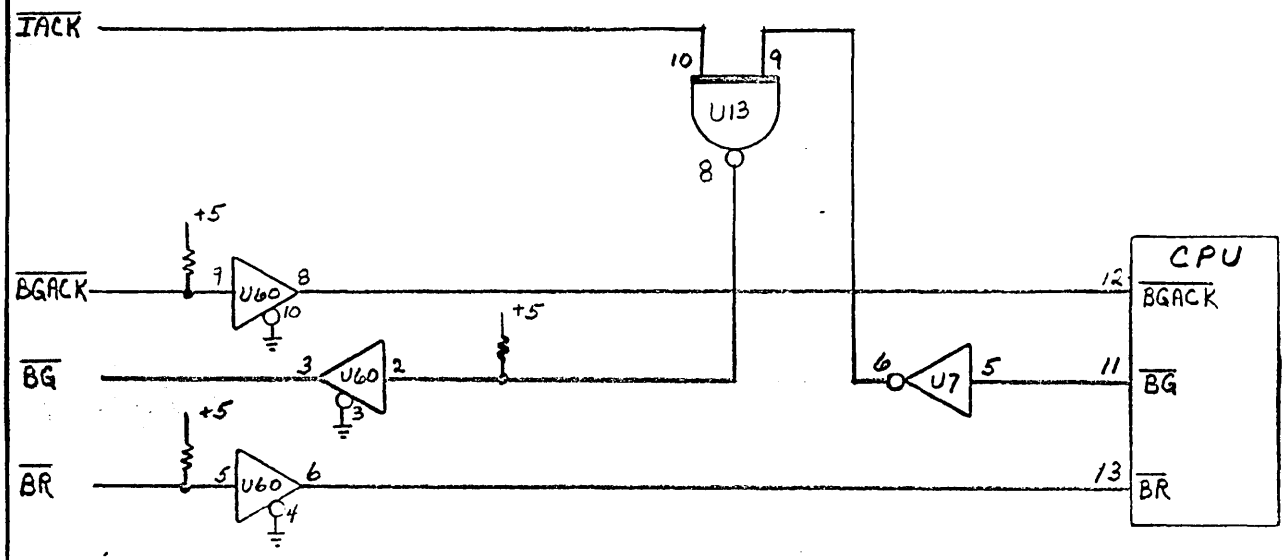
The two control signals $\overline{IOW}$ and $\overline{IOR}$ are used as control signals for Intel compatable peripheral chips. The signal IO is used to aid in internal chip select decoding.

U39 decodes A23, A22, and A21 to determine an I/O address. If the CPU has control of the bus, indicated by the negation of $\overline{BGACK}$, IO is asserted.

Before $\overline{AS}$ is asserted, U38, U28, U18, and U21 are cleared by AS low. When $\overline{AS}$ is asserted, either U19A or U19B is enabled depending upon the state of R/W. The first rising edge of the clock after AS is asserted clocks AS through U38. On the falling edge it is clocked through U28. If the operation is an I/O read, $\overline{IOR}$ is asserted at this

time. The next rising edge of the clock, U21 transfers AS to U19. If the operation is an I/O write, $\overline{IOW}$ is asserted. $\overline{IOW}$ is negated by $\overline{DTACK}$ being clocked through U18, clearing U21. $\overline{IOR}$ is negated by the negation of AS at U19.

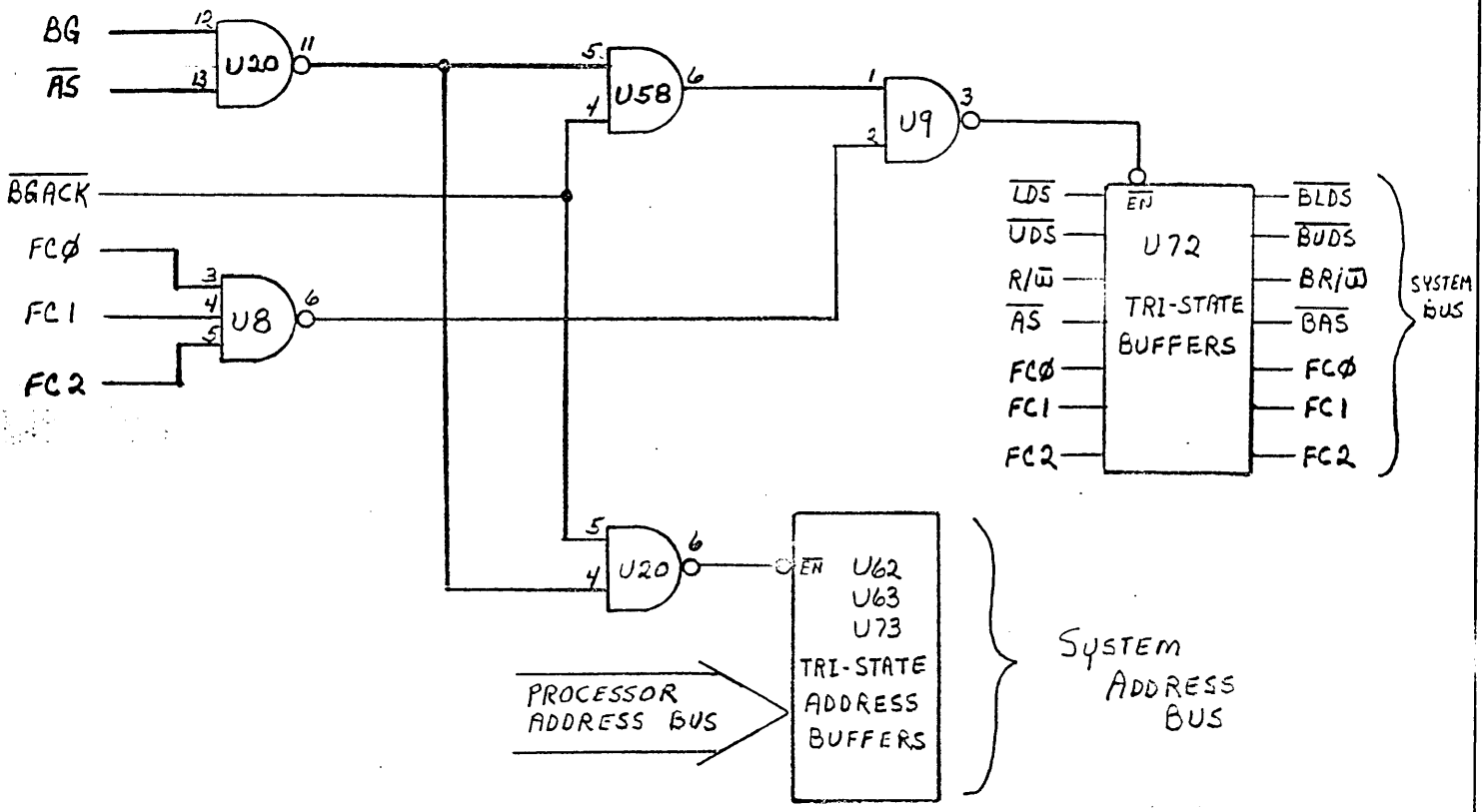
EXTERNAL BUS CONTROL



An external device capable of controlling the bus requests the bus by asserting $\overline{BR}$ (Bus Request). When the processor sees $\overline{BR}$, it asserts $\overline{BG}$ indicating that it will release bus control at the end of the current bus cycle.

U13 is enabled by $\overline{IACK}$ negated so that a bus request cannot be honored until after the processing of an interrupt. When the requesting device receives the $\overline{BG}$ and sees the negation of $\overline{AS}$, indicating the termination of the current processor cycle, it asserts $\overline{BGACK}$ (Bus Grant Acknowledge). The external device then has control of the bus until it negates $\overline{BGACK}$.

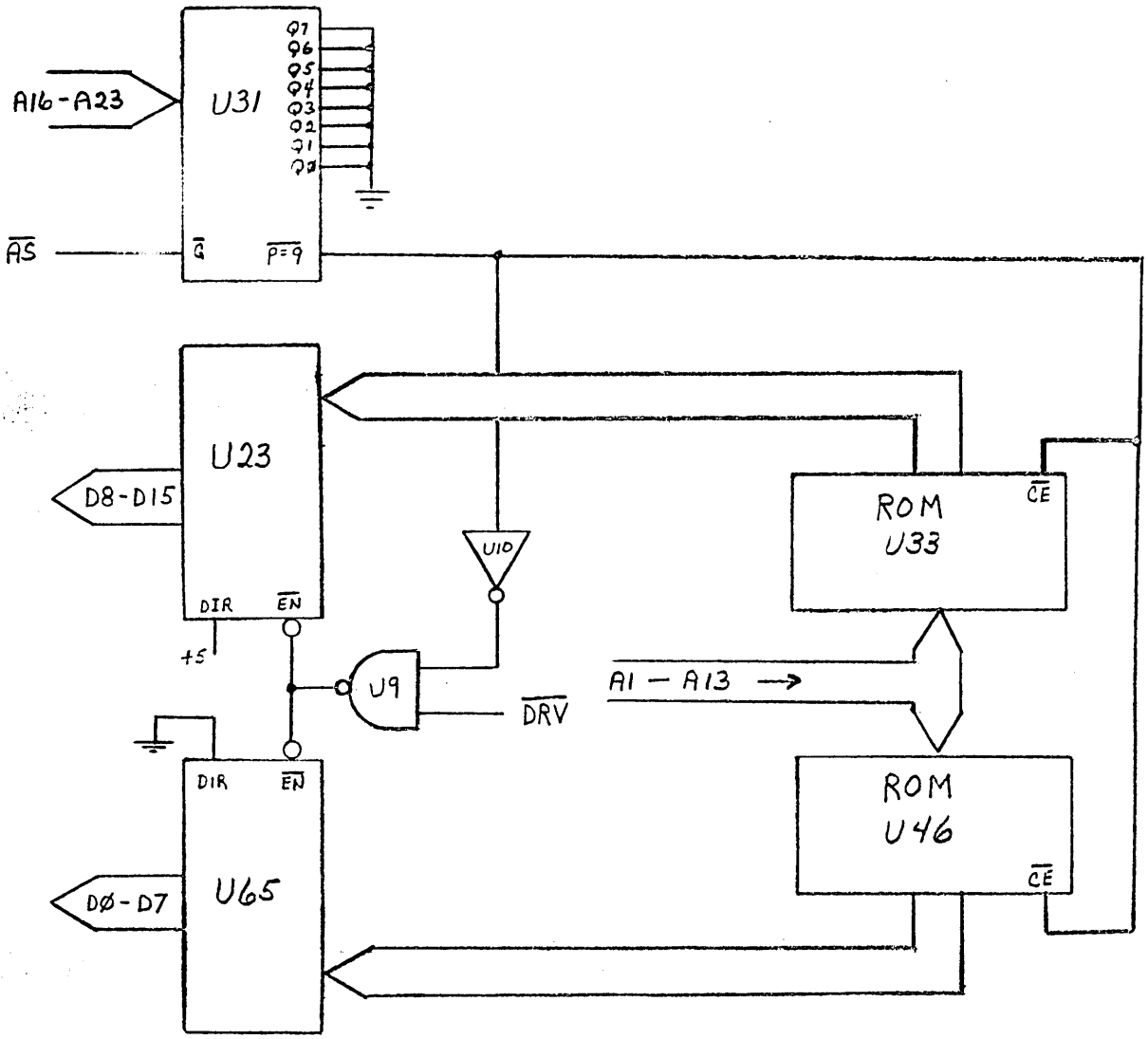
BUS CONTROL, CPU STATUS, AND ADDRESS BUFFERS



During a DMA cycle the CPU gives up control to an external device. The system address bus and control bus must be tri-stated to that the controlling device can apply signals to them. This is accomplished by decoding $\overline{BG}$, $\overline{BGACK}$, and $\overline{AS}$ with U20, U58, and U9. If the states of $\overline{BG}$, $\overline{BGACK}$, and $\overline{AS}$ indicate that an external device has control, U72 and the address buffers are tri-stated by bringing their enable inputs high.

During an interrupt cycle, the CPU is active during vector number acquisition (see Interrupt Decode and Detect), but the control signals are isolated from the system control bus, so that the I/O card responds to the $\overline{IACK}$ access not the RAM in the top of memory. U8 and U9 decode the states of FC0-FC2 (all high during interrupt) and tri-state U72.

ROM

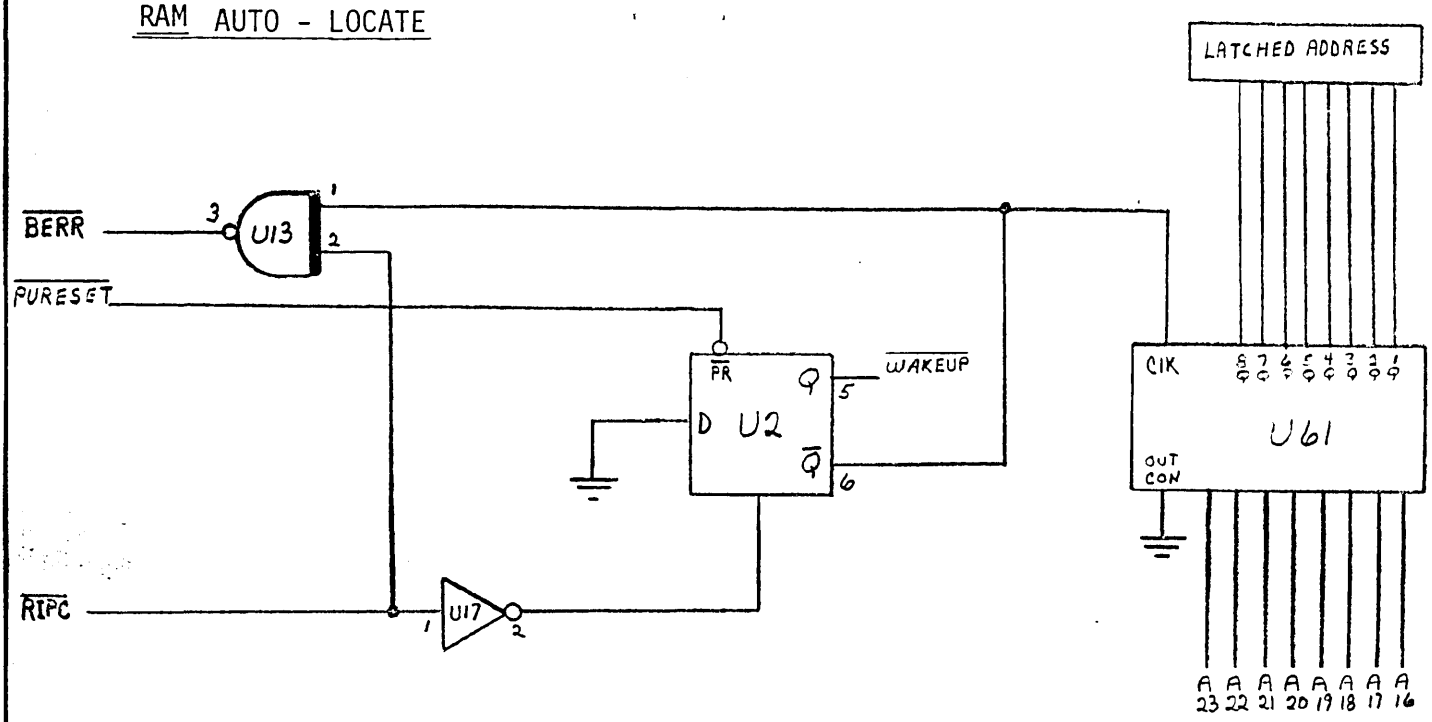


U31 decodes a low on A16 through A23 to enable $\overline{CE}$ to ROM. $\overline{DRV}$ is normally high so $\overline{p=q}$ is gated through U9 to enable the data buffer U23 and U65. Data at the location addressed by the address bus is transfered through the buffers onto the processor data bus.

$\overline{DRV}$ is used to isolate the ROM data output from the CPU data bus for test purposes. If $\overline{DRV}$ is asserted, the data buffers are never enabled.

				MODEL 9826A	STK NO
				THEORY OF OPERATION	
				BY Carl Anderson	DATE May 29, 1981
See page 1 for Rev.'s				APPD	SHEET NO 16 OF 26
LTR	PC NO	APPROVED	DATE	SUPERSEDES	DWG NO A-09826-90320-9
REVISIONS					

RAM AUTO - LOCATE



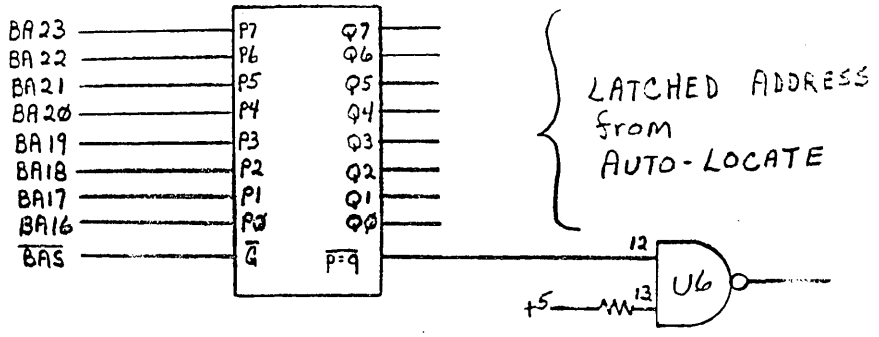
The on board RAM locates itself at the bottom of RAM memory. This auto-locate feature eliminates the need for user set switches on an internal board.

U2 is preset by $\overline{\text{PURESET}}$ at turn on. The low Q output inhibits U13. During wake up, the processor writes to and reads from each block of memory to determine how much memory is available. When there is no response from a memory card indicating end of memory, RIPC is asserted by the Bus Error Timeout circuit. Since U13 is inhibited, $\overline{\text{BERR}}$ is not asserted. Instead, U2 is clocked by RIPC. This enables U13 so that subsquent Bus Errors will be asserted, and clocks the address that is present when this first RIPC accurs into latch U61. Because the D input

of U2 is tied low, U2 and U61 outputs will remain in this state until power is cycled.

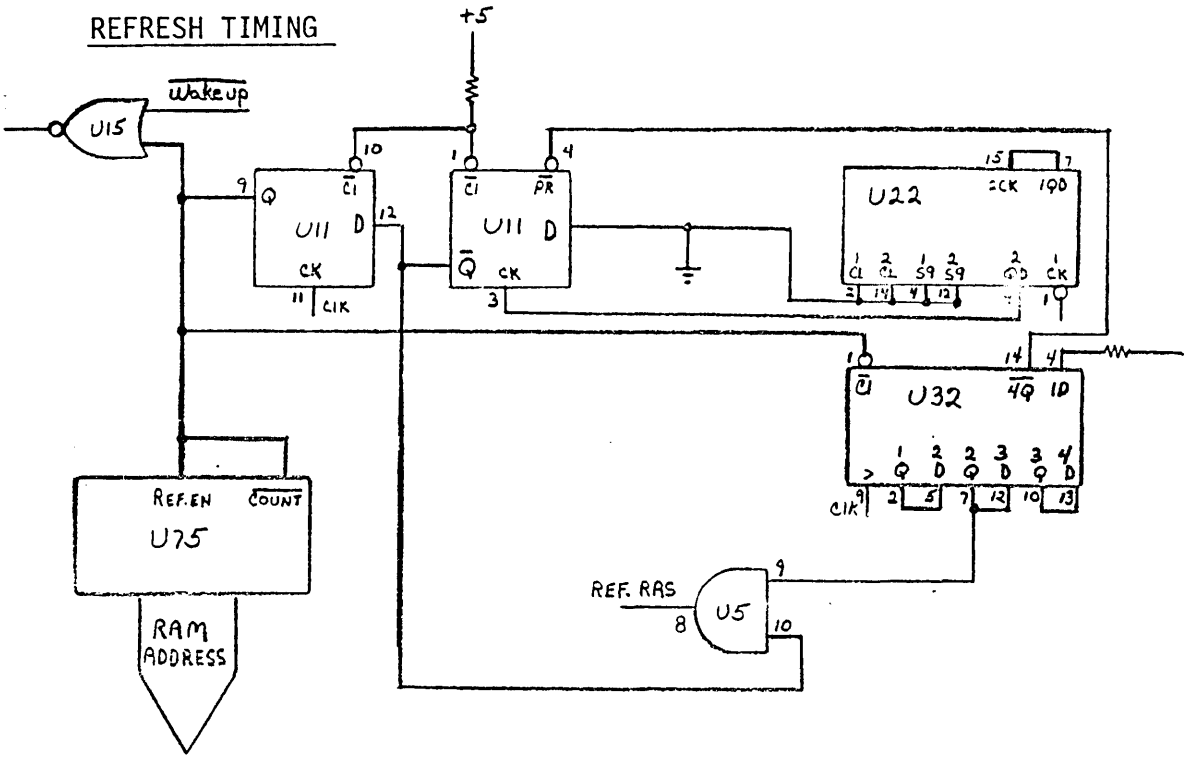
The latched address from U61 is used by the RAM Location Decoder to provide chip select to the RAM.

RAM LOCATION DECODER



The outputs of the RAM Auto Locate circuit (U61) are tied to Q inputs of comparator U71. When the address on BA16 through BA23 matches the latched address from Auto Locate and when U71 is enabled by $\overline{BAS}$, the output of U71 is asserted. If Ram is not being refreshed this output is gated to become chip select.

REFRESH TIMING



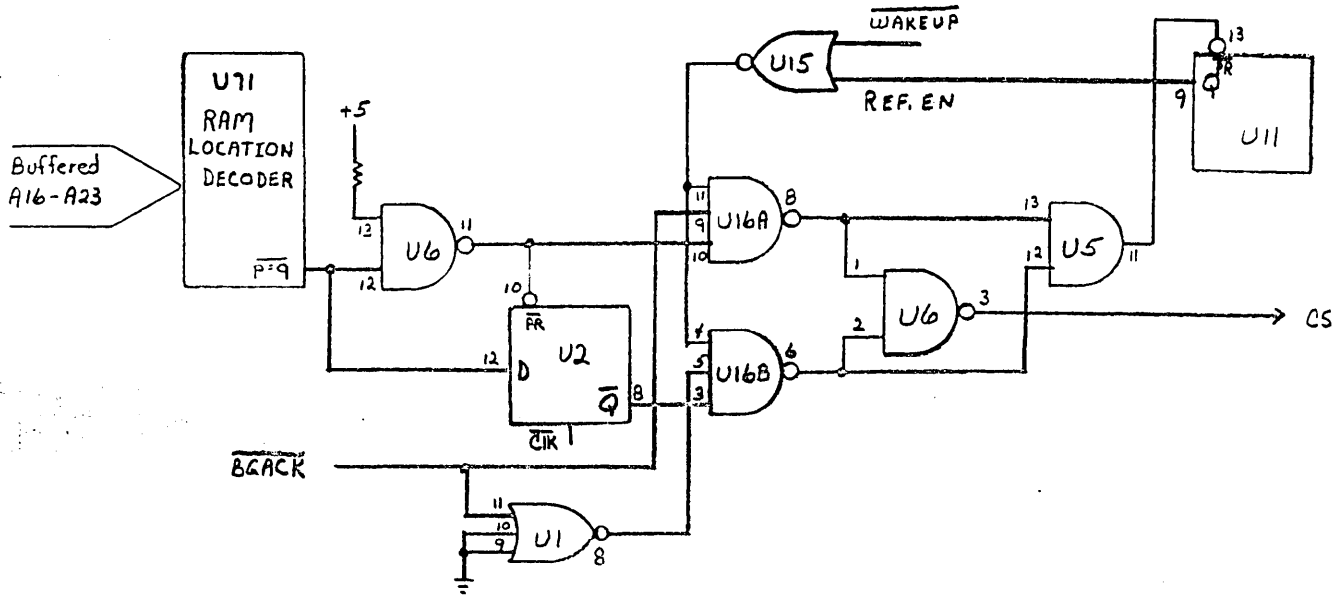
The 2118-4 RAM requires that each of the 128 rows be refreshed at least every 2 mS. This is accomplished by the dual decade counter, U22. The $\overline{CLK}$ input is 125 nS. U22 divides this by 100 producing an output which clocks U11 pin 3 at a 12.5 µS rate. Because the D input (pin 2) of U11 is tied low, when it is clocked by U22 it's $\overline{Q}$ output (pin 6) goes high. On the next rising edge of the clock Ref. en (U11 pin 9) is asserted. This occurance of Ref. en every 12.5 µS enables each row to be refreshed every 1.6 mS.

When Ref. en is asserted the internal counter of U75 supplies the refresh address to RAM, the output of U15 goes low disabling chip select, and the clear input of U32 is brought high allowing it to start counting.

				MODEL 9826A	STK NO
				THEORY OF OPERATION	
				BY Carl Anderson	DATE May 29, 1981
	See page 1 for Rev.'s			APPD	SHEET NO 20 OF 26
LTR	PC NO	APPROVED	DATE		
	REVISIONS			SUPERSEDES	DWG NO A-09826-90320-9

250 nS (2 clocks) later, 2Q of U32 (pin 7) goes high. At this time, both inputs of U5 are high, asserting the Ras for Refresh. After another 250 nS, U32's $\overline{4Q}$ output (pin 14) goes low. This negates the refresh Ras by reseting U11 bringing U5 pin 10 low. The next rising edge of the clock negates ref.en,U15's output (pin 10) goes high enabling chip select, U32 is cleared, and the internal counter of U75 is advanced by asserting count.

RAM ACCESS



If RAM is being refreshed, the Ref.en input to U15 is high. The output of U15 is low inhibiting both gates of U16. This causes CS to remain low. When the refresh is complete, Ref.en is negated. The output of U15 then enables both gates of U16.

If RAM is being accessed by the CPU, $\overline{BGACK}$ is high, $\overline{p=q}$ is inverted by U6 and fed to U16 pin 10. Since this gate is enabled by $\overline{BGACK}$ high and Ref.en low, U16 pin 8 goes low allowing U6 to assert CS. At the same time, U5 is inhibited causing its output to go low. This presets the Ref.en flip-flop (U11) preventing refresh from occuring during a memory access.

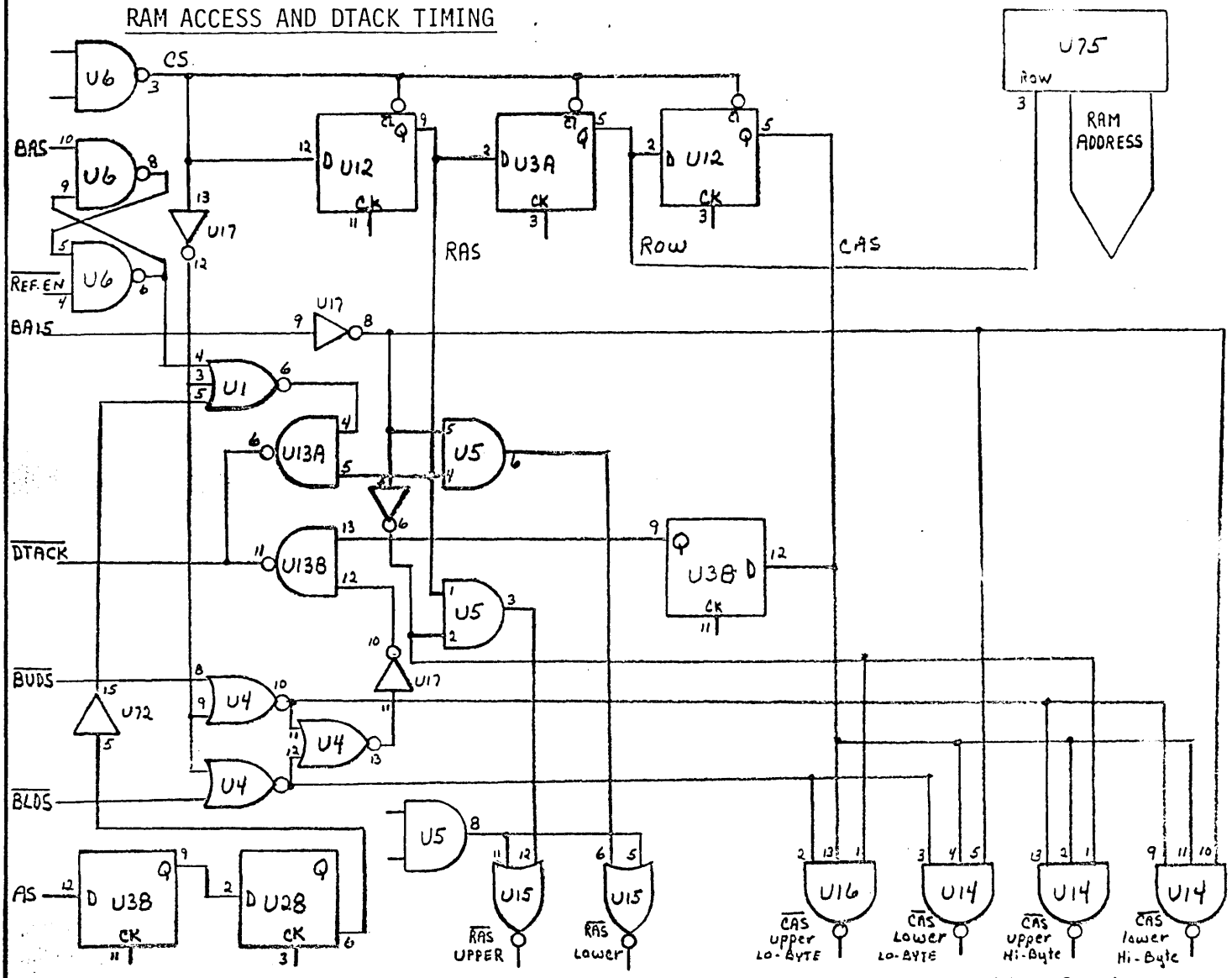
Because DMA is asynchronous with the clock and RAM control is synchronous, a means must be provided to synchronize a DMA access of RAM to prevent possible timing problems. This is accomplished with

U2 and U16B.

During DMA $\overline{\text{BGACK}}$ is asserted. This inhibits U16A. U1 inverts $\overline{\text{BGACK}}$ and enables U16B. When $\overline{\text{p=q}}$ is asserted, it is clocked through U2 by $\overline{\text{clk}}$ and inverted, gated through U16B and U6, asserting CS. At the same time, U5 is inhibited presetting U11 to prevent refresh.

$\overline{\text{WAKEUP}}$ is normally low except during power-up. At power-up, on board RAM could be located anywhere, depending upon the state of the RAM location decoder. If $\overline{\text{WAKEUP}}$ is high, CS is negated and stays that way until the first bus error. (See RAM Auto-Locate).

RAM ACCESS AND DTACK TIMING



A normal CPU access of RAM is a 5 cycle access. However, if Refresh occurs during the access, time must be allowed for the refresh before the transfer of data can occur. This is accomplished by controlling the time at which $\overline{\text{DTACK}}$ occurs. During a normal access, $\overline{\text{ENDT}}$ is gated to provide $\overline{\text{DTACK}}$. When RAM is accessed during a refresh cycle, $\overline{\text{ENDT}}$ is ignored, and since CAS won't occur until after refresh, it is gated to provide $\overline{\text{DTACK}}$.

				MODEL 9826A	STK NO
				THEORY OF OPERATION	
				BY Carl Anderson	DATE May 29, 1981
	See page 1 for Rev.'s			APPD	SHEET NO 24 OF 26
LTR	PC NO	APPROVED	DATE	SUPERSEDES	DWG NO A-09826-90320-9
	REVISIONS				

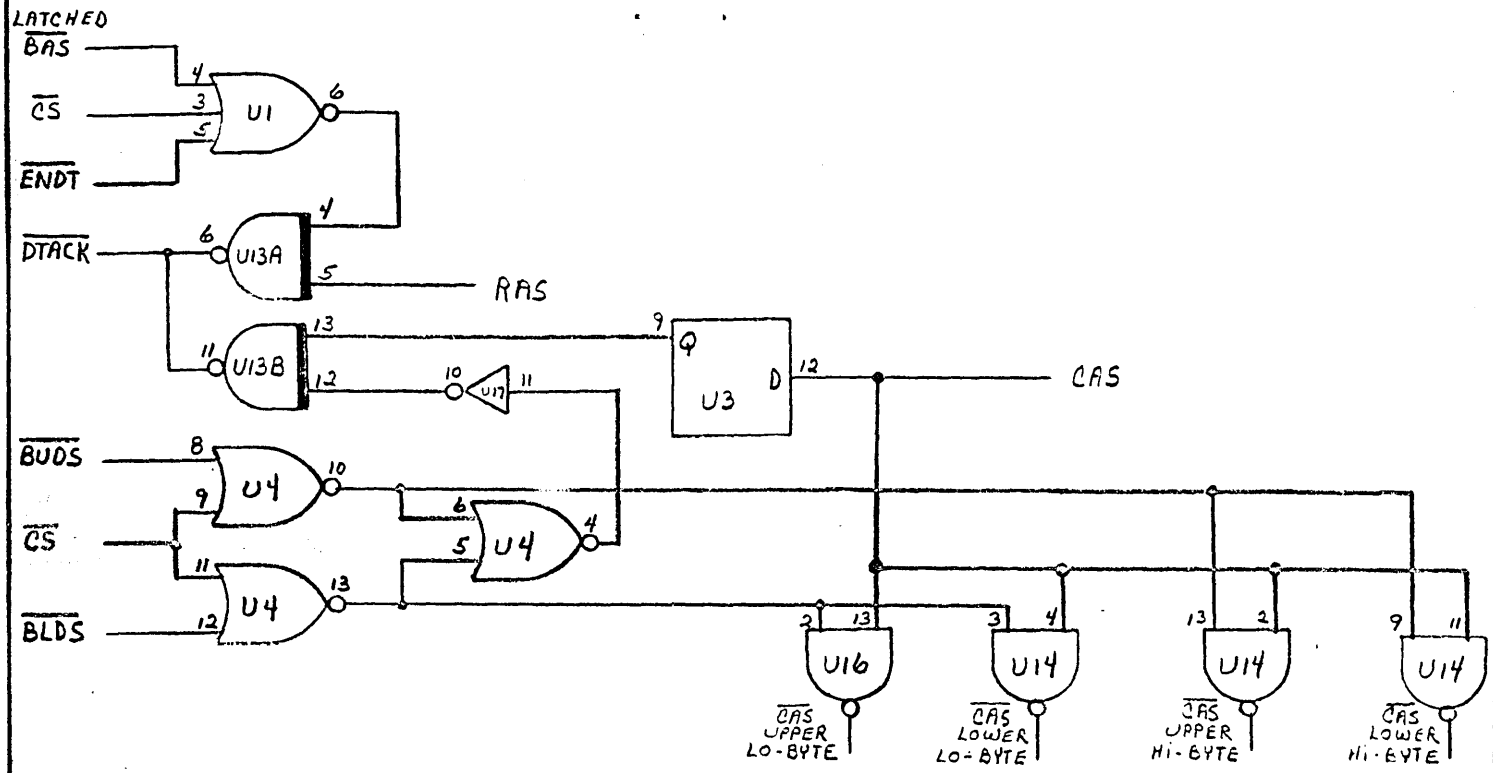
RAM Access with no Refresh

While CS is negated, U12 and U3 are cleared negating RAS, enabling row address to RAM by pulling the row input to U75 low, and negating CAS. When CS is asserted, it is clocked through U12 asserting RAS. RAS is gated to U5 pin 6 or pin 3 depending upon the state of BA15. If BA15 is high, RAS is gated to U15 pin 12 asserting $\overline{RAS}$ upper. If BA15 is low, it is gated to U15 pin 6 asserting $\overline{RAS}$ lower. At the same time, RAS enables U13A (EDTACK) and AS has been clocked through U38. The next clock falling edge clocks RAS through U3 asserting Row which enables the column address to RAM through U75, and AS is clocked through U28 asserting $\overline{ENDT}$. $\overline{ENDT}$ is inverted by U1 and applied to U13A, asserting $\overline{DTACK}$. The next clock transition clocks Row through U12 asserting CAS. CAS is gated to RAM through U14 and U16 depending upon the states of UDS, LDS, and BA15. BA15 determines high or low address, UDS and LDS determine high or low byte or both.

RAM Access with a Refresh

During a refresh cylce $\overline{Ref.en}$ is asserted, inverted and latched by U6, CS is inhibited and RAS remains low. These signals combine to inhibit U13A. When Ref.en is negated, indicating end of refresh, CS is asserted, $\overline{CS}$ goes low and $\overline{BUDS}$ and/or $\overline{BLDS}$ have already been asserted. These are gated through U4 and U17 enabling U13B (LDTACK). RAS, ROW, and CAS are clocked through U12 and U3 in the manner described above. CAS is clocked through U3B asserting LDTACK which is gated through U13 asserting $\overline{DTACK}$.

DATA STROBE HOLD OFF OF A WRITE CYCLE



During a DMA write to RAM, the amount of time necessary to set up data to RAM is unknown. Because of this the DMA controller starts the RAM access by placing the RAM address on the bus and asserting $\overline{BAS}$. While data is being set up, the DMA controller holds off the data strobes until data is valid. $\overline{ENDT}$ is inhibited because the control bus is tri-stated when the CPU gives up control, so U13A (EDTACK) is inhibited.

RAS, ROW, and CAS are asserted in the normal manner, but $\overline{CAS}$ is not applied to RAM, because U16 and U14 are inhibited by the negation of the data strobes.

Once data is valid, the DMA controller asserts $\overline{BUDS}$ and/or $\overline{BLDS}$, which are gated through U4 and U16 or U14 to assert $\overline{CAS}$, and through U4, U17, and U13 to assert $\overline{DTACK}$.

				MODEL 9826A	STK NO
				THEORY OF OPERATION	
				BY Carl Anderson	DATE May 29, 1981
	See page 1 for Rev.'s			APPD	SHEET NO 26 OF 26
LTR	PC NO	APPR	DATE	SUPERSEDES	DWG NO A-09826-90320-9
	REVISIONS				