Interfacing the DP8408A/09A To Various Microprocessors

National Semiconductor
Application Note 309
Chuck Pham, Webster (Rusty) Meier
May 1986

High storage density and low cost have made dynamic RAMs the designer's choice in most memory applications. However, the major drawback of dynamic RAMs is the complex timing involved. First, a $\overline{RAS}$ must occur with the row address previously set up on the multiplexed address bus. After the row address has been held for some minimum time after $\overline{RAS}$ (namely the row address hold time of the dynamic RAMs, t_{RAH}), the column address is set up and then $\overline{CAS}$ occurs. In addition, refreshing must be done periodically to keep all memory cells charged.

With the introduction of the DP8408A Dynamic RAM Controller/Driver, the above complexities are simplified. The DP8408A is housed in a 48-pin package with eight multiplexed address outputs (Q0–7) and six control outputs ($\overline{RAS}$0–3, $\overline{CAS}$, $\overline{WE}$). It consists of two 8-bit address latches and an 8-bit refresh counter. All the output drivers are capable of driving 500 pF loads.

The following discussion demonstrates a typical application of the DP8408A Dynamic RAM Controller/Driver in Z8000™- and Z80®-based systems. The DP8408A basically has six modes of operation: Externally Controlled Refresh, Externally Controlled All-$\overline{RAS}$ Write, Externally Controlled Access, Auto Access (slow t_{RAH}), Auto Access (fast t_{RAH}) and Set End of Count.

The DP8408A, operating in the auto access mode, requires only $\overline{RASIN}$ to initiate a memory access cycle because all the dynamic RAM's control signals are automatically delayed from this input. (Refer to *Figure 1* for the auto access timing sequence.)

In the following applications, the DP8408A operates in either mode 5 or mode 6 Auto Access and mode 1 or 2 Externally Controlled Refresh to provide minimum additional logic.

The DP8408A and Z8000 Interface

MEMORY ACCESS CYCLE

Figure 2a shows the detailed block diagram of Z8000 and the DP8408A interface. Consider a memory cycle of the Z8000; first, the memory address is output on the Address and Data multiplexed bus (AD0–15) during T1 and is latched to the DP8408A by $\overline{AS}$. Simultaneously, $\overline{MREQ}$ goes low and is used to provide $\overline{RASIN}$ to initiate a memory transaction cycle. Then the selected $\overline{RAS}$ output, row address hold time (t_{RAH}), column address set up time (t_{ASC}) and $\overline{CAS}$ output will follow $\overline{RASIN}$ as determined by the auto access modes. A maximum of one wait state is required for 6 MHz and 10 MHz CPUs. This wait state is automatically inserted by the $\overline{CAS}$ output of the DP8408A. For systems using byte-writing, the DM74S158 provides two separate $\overline{CAS}$ outputs for accessing the low and high byte of memory. Note that $\overline{DS}$ from the Z8000 is also gated with the DP8408A's CAS output to generate $\overline{CAS}$L and $\overline{CAS}$H. This guarantees the valid data from the Z8000 being written into memory during memory write cycles. Refer to *Figure 3* for the detailed memory transaction cycle timing.

The following formula allows the designer to determine the proper memory speed in terms of t_{CAC} (access time from $\overline{CAS}$):

$$t_{CAC}\ \text{max.} = 3 \times t_{cC} - \text{tdc(MR)} - t_{RICL} - t_{CASdly} - t_{sDR}(C) - 15.$$

The Z8000 parameters:

t_{cC}: clock cycle time

$t_{sDR}(C)$: read data to clock $\downarrow$ set up time

tdc(MR): clock to $\overline{MREQ}$ delay

The DP8408A, 74S158 and 74LS245 parameters:

t_{RICL}: RASIN to CAS delay

t_{CASdly}: the propagation delay of the 74S158

15 ns: the propagation delay of the 74LS245 (at 50 pF load)

For the 10 MHz CPU and the DP8408A:

t_{CAC} max. = 300 − 40 − 131 − 14 − 10 − 15 = 90 ns.

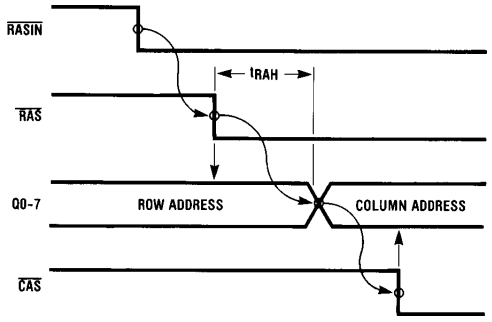
- t_{RICL} max. (mode 6) = 131 ns at 15 pF load.
- t_{CASdly} max. = 14 ns at 50 pF load.

Since $\overline{MREQ}$ is connected directly to $\overline{RASIN}$, t_{RP} ($\overline{RAS}$ precharge time) and t_{RAS} ($\overline{RAS}$ pulse width) are determined by $\overline{MREQ}$ high and low, respectively.

MEMORY REFRESH CYCLE

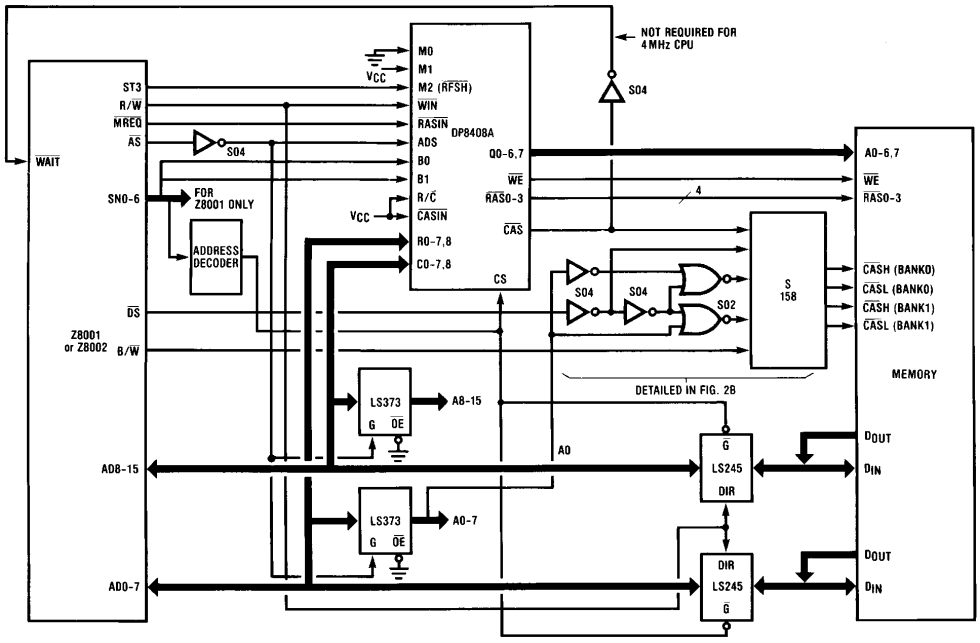
The Z8000 CPU contains a refresh rate counter for automatic memory refresh. This counter should be programmed during the processor initialization to determine the refresh rate. Since memory refresh is automatically inserted by the Z8000, there is no additional refresh arbitration logic allowed. The CPU's STATUS 3 (ST3) output can be directly connected to the M2 ($\overline{RFSH}$) pin of the DP8408A. During the memory refresh cycle, ST3 goes low, setting the DP8408A in the external control refresh mode (mode 2). Then all four $\overline{RAS}$ outputs will follow $\overline{MREQ}$ to strobe the DP8408A's refresh address to all memory banks (the Z8000 refresh address is ignored). As $\overline{MREQ}$ goes high again, the DP8408A increments its refresh counter, preparing it for the next refresh cycle. Refer to *Figure 4* for the refresh cycle timing. Note that ST3 also goes low during the internal cycle, I/O reference cycle and interrupt acknowledge cycle, but the memory will not be refreshed because $\overline{MREQ}$ is not active during these cycles. The DP8408A on-chip refresh counter will not be incremented when M2 goes low unless $\overline{MREQ}$ is inserted.

RRD-B30M115/Printed in U. S. A.



TL/F/5040–1

FIGURE 1. Auto Access Timing Sequence (Mode 5 or Mode 6)



TL/F/5040–2

FIGURE 2a. Z8000 and DP8408A Interface

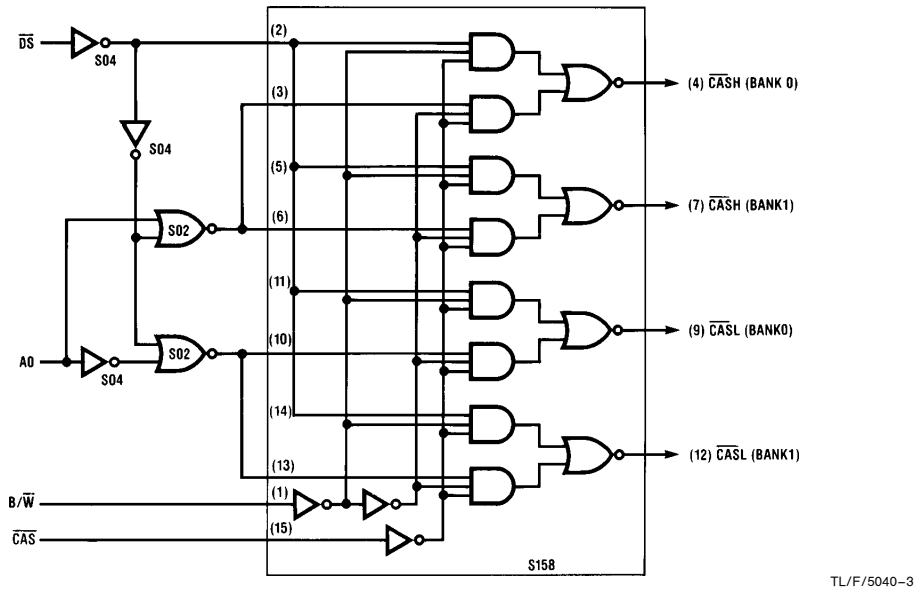


FIGURE 2b. CASH and CASL Decoder

When the processor is in either halt state (by executing the privileged HALT instruction) or single-stepping mode (when $\overline{STOP}$) input is low), it introduces memory refresh cycles. However, care should be taken when the CPU is in either a WAIT state or a Bus Acknowledge cycle, that the dynamic RAM refresh will not take place. If these conditions occur over a long period of time, a burst refresh is recommended. This can be done by toggling $\overline{RASIN}$ while keeping M2 low, until all the rows of the dynamic RAM have been refreshed, then the CPU can resume its operations.

The DP8408A and Z80A® Interface

INSTRUCTION FETCH CYCLE

Figure 5 shows the detailed interconnections between the DP8408A, the Z80 and the Dynamic RAMs. *Figure 6* shows the timing during an M1 cycle (op code fetch). The program counter is output on the address bus at the beginning of the M1 cycle. One-half clock later $\overline{MREQ}$ goes active. This input is used to provide $\overline{RASIN}$ to the DP8408A to access the dynamic memory. Subsequently, the selected $\overline{RAS}$ output, Row to Column Select and then $\overline{CAS}$ output will automatically follow $\overline{RASIN}$ as determined by the Auto Access modes of the DP8408A. The $\overline{RD}$ line also goes active to indicate a memory read cycle is in progress. After t_{CAC} (access time from $\overline{CAS}$), read data becomes valid. This data is sampled on the rising edge of T3, then both $\overline{MREQ}$ and $\overline{RD}$ go inactive. Immediately following this, $\overline{RFSH}$ goes low, putting the DP8408A in the Externally Controlled Refresh mode. The $\overline{MREQ}$ goes active causing all four $\overline{RAS}$ outputs to go active to perform a refresh to all the banks of the dynamic RAMs. Note that during memory refresh cycles, the refresh address from the CPU is output on the address bus. However, the contents of the DP8408A on-chip refresh counter are used instead to provide the row address to the dynamic memory array. Since the Z80 provides only a 7-bit refresh address, it is an advantage to use the DP8408A 8-bit refresh counter to support 64k dynamic RAMs directly. The DP8408A refresh counter is incremented as $\overline{MREQ}$ returns high, ending the memory refresh. The $\overline{RFSH}$ goes inactive returning the DP8408A back to the Auto Access mode, preparing it for the next access cycle.

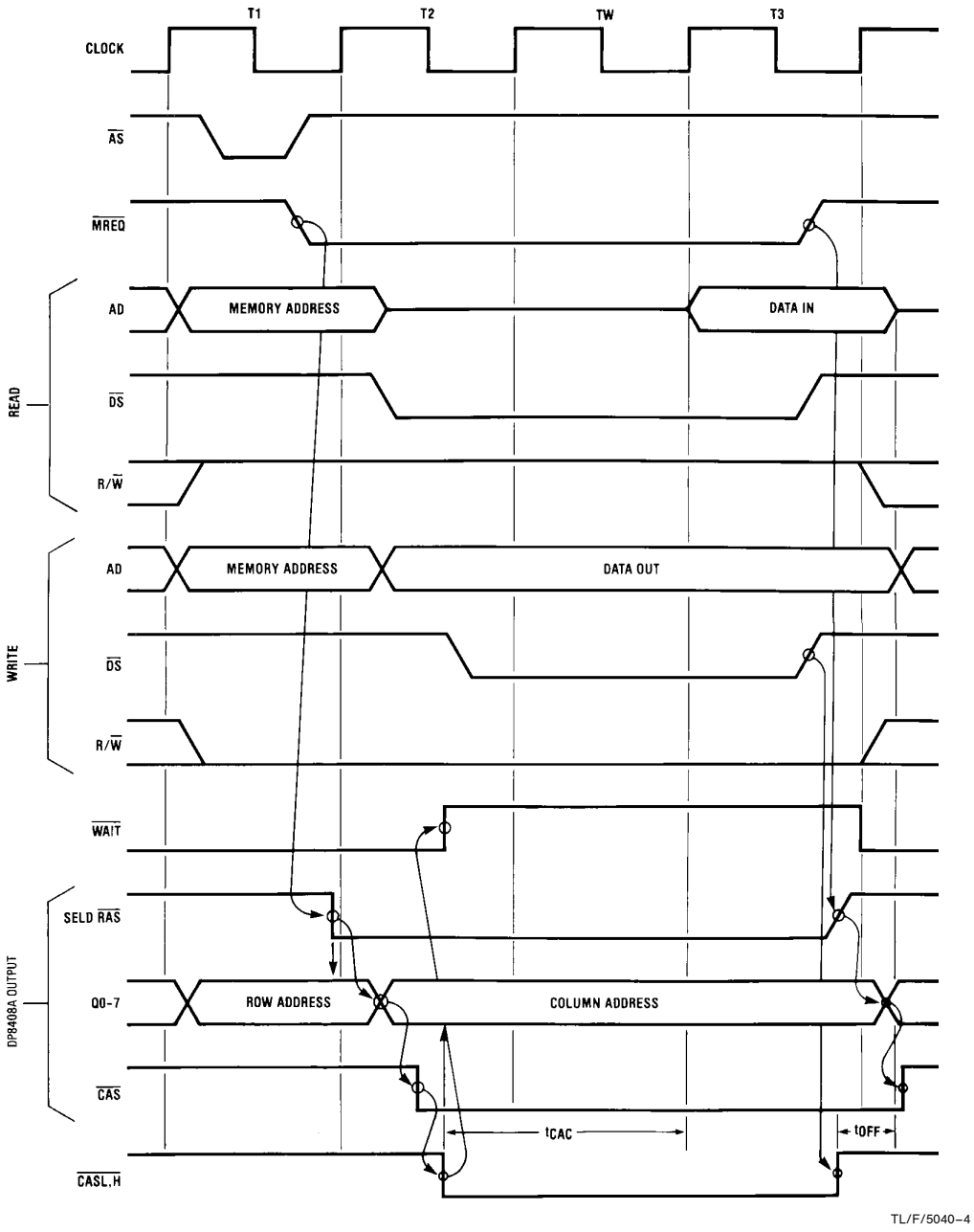
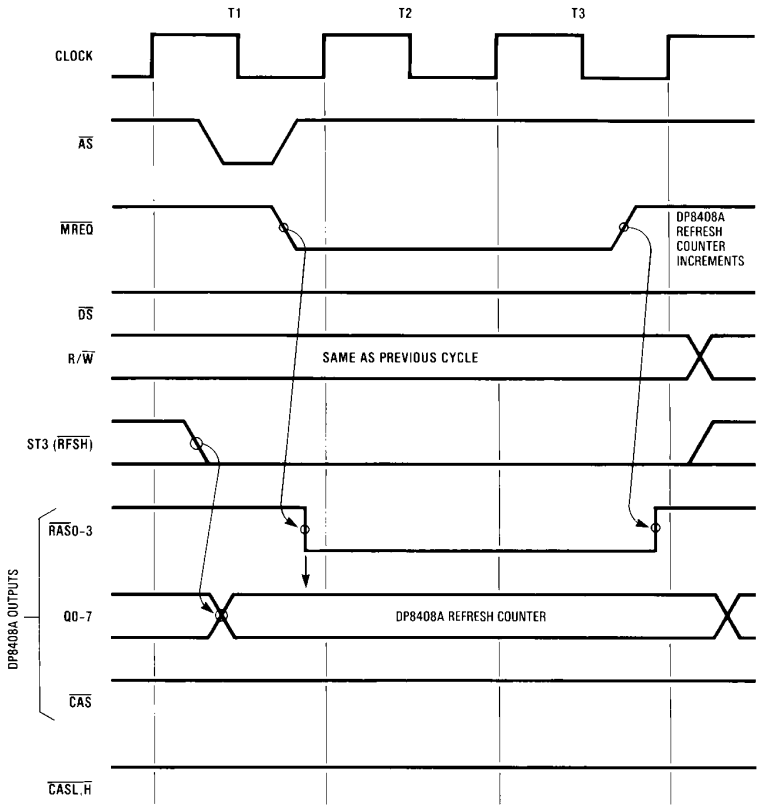
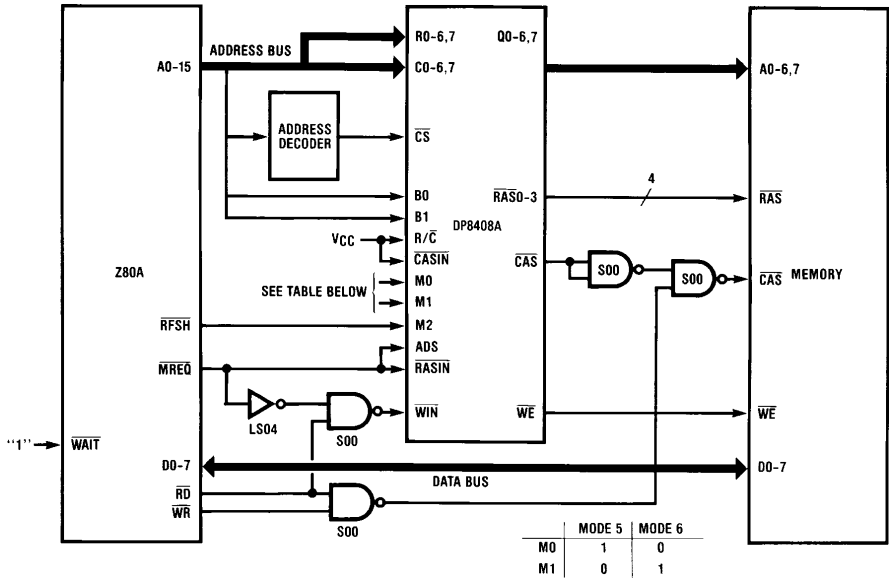


FIGURE 3. Memory Transaction Cycles



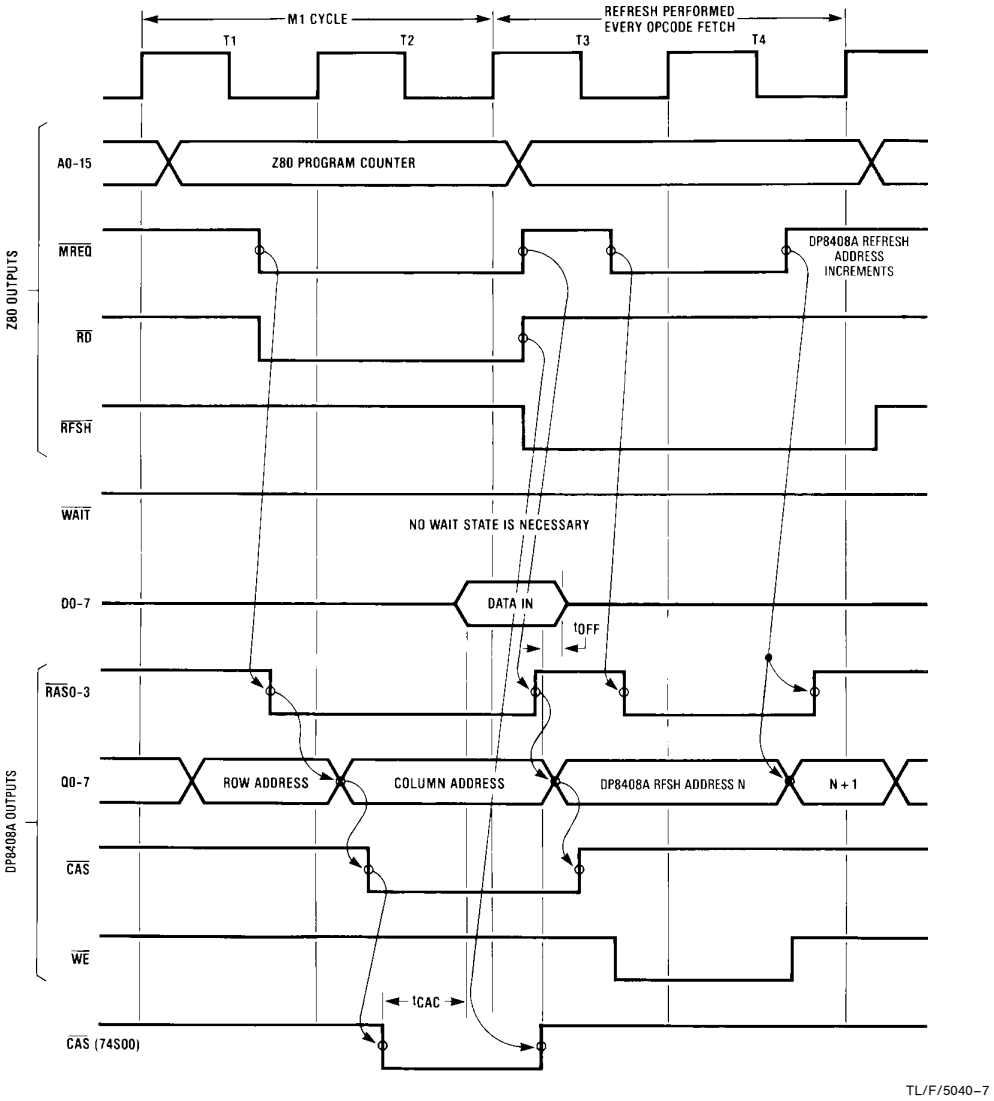
TL/F/5040–5

FIGURE 4. Memory Refresh Cycle



TL/F/5040–6

FIGURE 5. DP8408A and Z80A Interface



TL/F/5040–7

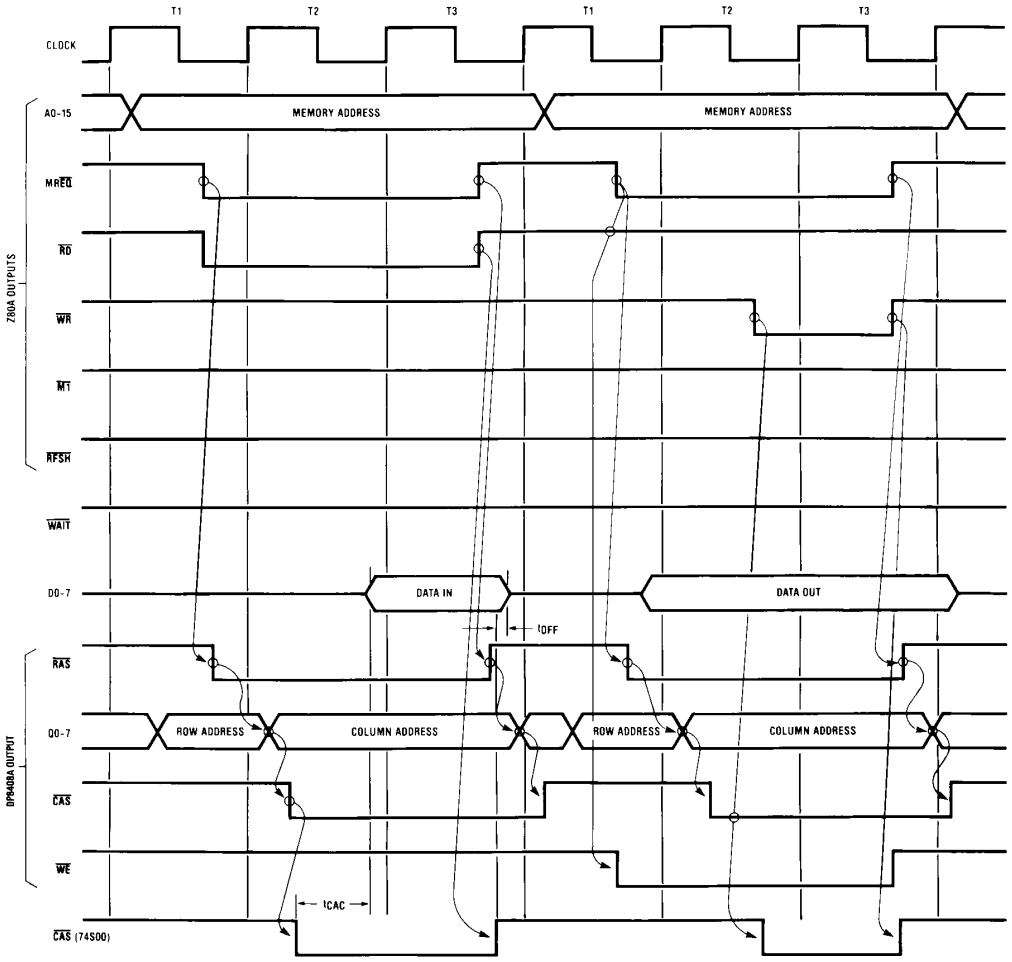
FIGURE 6. Z80A Op Code Fetch Cycle Showing Memory Refresh

MEMORY ACCESS CYCLE

Figure 7 shows the timing of the memory read and memory write cycle other than for the M1 op code fetch cycle. Similar to the op code fetch cycle, $\overline{MREQ}$ is used to provide $\overline{RASIN}$. $\overline{MREQ}$ goes active after the address to the memory has had time to stabilize. Again, $\overline{RAS}$ output, Row to Column Select and then $\overline{CAS}$ output will automatically follow $\overline{RASIN}$ to access the specified memory location. For a memory read cycle, both $\overline{MREQ}$ and $\overline{RD}$ go active, and as a result, $\overline{WIN}$ remains high (refer to *Figure 5*), which allows a memory read operation to occur. On the other hand, only $\overline{MREQ}$ goes active during a write cycle, which forces $\overline{WIN}$ low, indicating an early write cycle. It should be noted that the $\overline{CAS}$ output to the memory array will not go low until $\overline{WR}$ goes low during memory write cycles as this guarantees the valid CPU data will be written into memory.

It is worth mentioning that the Z80 CPU provides powerful block transfer instructions. An example is the LDIR (load, increment and repeat); using only this instruction, the programmer can move any block of data from the location pointed to by the D and E registers. This operation is repeated until the byte counter (B and C registers) reaches zero. Thus, this single instruction can move any block of data from one location to any other. Due to the fact that this instruction is refetched after each data byte transfer, the memory refresh cycle always takes place even though a transfer of up to 64k bytes of data may be performed. Furthermore, when the CPU has executed the software HALT instruction and is waiting for an interrupt before normal CPU operations can resume, the CPU executes NOP instructions to maintain memory refresh activity.

However, care should be taken when the CPU is in either WAIT state or a Bus Acknowledge cycle, the dynamic RAM refresh will not take place. If these conditions occur long enough, a burst refresh is recommended, and it can be done by toggling $\overline{RASIN}$ while keeping M2 low until all the rows of the dynamic RAM have been refreshed before the CPU can resume its operation.



TL/F/5040–8

FIGURE 7. Z80A Memory Read and Memory Write Cycle

The following formulas allow designers to select the appropriate dynamic memory, based on different CPU and DP8408A speed versions, to allow the CPU full speed of operation:

max. t_{CAC}: $1.5 \times t_{Cmin} - t_{DL\phi}(MR) - tRICL - t_{CASDLY} - t_{S\phi}(D)$

min. t_{RP}: $tw(MRH) = tw(\phi H) + t_f - 20$

min. t_{RAS}: $tw(MRL) - 20 = t_C - 50$

Dynamic RAM Parameters:

t_{CAC}: access time from $\overline{CAS}$

t_{RP}: $\overline{RAS}$ precharge time

t_{RAS}: $\overline{RAS}$ pulse width

Z80 Parameters:

t_C: clock period

$tw(\phi H)$: clock pulse width, clock high

tf: clock fall time

$t_{DL\phi}(MR)$: $\overline{MREQ}$ delay from falling edge of clock, $\overline{MREQ}$ low

$t_{S\phi}(D)$: Data set up time to rising edge of clock during M1 cycle

DP8408A and 74S00 Parameters:

t_{RICL}: $\overline{RASIN}$ to $\overline{CAS}$ output delay

t_{CASDLY}: propagation delay of the two 74S00 NAND gates

For example, if the Z80A (4 MHz) and the DP8408A are used, then:

max. t_{CAC}: $1.5(250) - 85 - 132 - 13 - 50 = 95$ ns

min. t_{RP}: $110 + 20 - 20 = 110$ ns

min. t_{RAS}: $t_C - 50 = 200$ ns

t_{RICL} max.

(mode 6): 132 ns at 15 pF load

t_{CASDLY} max.: 13 ns at 50 pF load

Therefore, in this case, the designer should choose a dynamic memory which has maximum t_{CAC} of 95 ns, minimum t_{RP} of 110 ns and minimum t_{RAS} of 200 ns.

DP8409A and MC68B09E Interface

DP8409A OVERVIEW

The DP8409A Dynamic RAM Controller/Driver is designed to control all multiplexed-address dynamic RAMs. It consists of two 9-bit address latches and a 9-bit refresh counter, thus allowing control of all 16k, 64k, and the coming generation 256k dynamic RAMs. More important, all the DP8409A outputs are capable of driving 500 pF loads.

The DP8409A basically has eight modes of operation: Externally Controlled Refresh, Automatic Forced Refresh, Internal Auto Burst Refresh, All $\overline{RAS}$ Auto Write, Externally Controlled Access, Auto Access (slow t_{RAH} and with hidden refresh), Fast Auto Access (fast t_{RAH}) and Set End of Count. Of all these modes, Auto Access (mode 5) and Auto Forced Refresh (mode 1) are the most popular and will be used throughout this application. Mode 5 requires only $\overline{RASIN}$ to initiate a memory access cycle, because all the dynamic RAM's control signals are automatically delayed from this input, as shown in *Figure 1*. To attain maximum system throughput, it is obviously advantageous to perform refreshes without interrupting the system. The DP8409A can do this by monitoring the $\overline{CS}$ input to see if it is high. If $\overline{CS}$ is high, the RAMs are not being accessed. If $\overline{CS}$ is high for one cycle, the DP8409A performs a hidden refresh during this cycle, and stops in time for the system to start another access. But if a hidden refresh does not occur in a specific time slot, a refresh must be forced and this can be done by using Mode 1, Automatic Forced Refresh.

To perform automatic forced refresh, the DP8409A must receive two clock signals: the refresh period clock, RFCK, and RGCK, the $\overline{RAS}$-generator clock; RGCK can be the microprocessor clock. It takes approximately four RGCK clock periods to perform this automatic forced refresh. The DP8409A gives preference to hidden refresh using RFCK as a level reference. The refresh time slot commences as RFCK goes high. If $\overline{CS}$ goes high while RFCK is high, the refresh counter is enabled in the address outputs. All four $\overline{RAS}$ outputs follow $\overline{RASIN}$; so to perform a hidden refresh, $\overline{RASIN}$ must be set low and the refresh counter gets incremented as $\overline{RASIN}$ goes high. The DP8409A allows only one such hidden refresh to occur with a clock cycle of RFCK to minimize power consumption.

If a hidden refresh does not occur the DP8409A must force a refresh before RFCK begins a new cycle on a low-to-high transition. Therefore, as RFCK goes low (and a hidden refresh has not occurred), RF I/O (Refresh Request) goes low, requesting that a refresh be performed. When the system acknowledges the request, it sets M2 low, and prevents further access to the DP8409A. Then two RGCK negative edges after M2 has gone low, all four $\overline{RAS}$ outputs go low and remain low for two RGCK clock periods. After all four $\overline{RAS}$ outputs have gone low, M2 can go high any time to end the Automatic Forced Refresh. The DP8409A allows only one automatic refresh to occur within a clock cycle of RFCK.

MEMORY ACCESS

The MC68B09E starts a memory access cycle when E goes low, then the memory address becomes valid on the Address Bus A0–15. This address is decoded to provide Chip Select ($\overline{CS}$) to the DP8409A. Then Q goes high and sets $\overline{RASIN}$ low from the PAL® Control Logic as shown in *Figure 12*. Note that $\overline{CS}$ must go low for a minimum of 10 ns before the assertion of $\overline{RASIN}$ for a proper memory access. This is important because a false hidden refresh may take place when this 10 ns minimum setup time is not met. $\overline{RASIN}$ goes low initiating the auto access sequence as shown in *Figure 1*. Mode 5 guarantees a 30 ns minimum for row address hold time and a minimum of 8 ns column address set up time. $\overline{RASIN}$ remains low until E goes low at the end of the current access cycle. Using the 16R6A Programmable Array Logic (25 ns PAL), the maximum access time from $\overline{CAS}$ of the selected dynamic RAM is determined as follows:

Max. t_{CAC}: $3 \times 125 - 25 - 160 - 40 = 150$ ns 8409A

t_{CAC}: $3 \times 125 - 25 - 130 - 40 = 180$ ns 8409A-2

Q high to

E low: 3×125 ns (8 MHz clock) $= 375$ ns

Q high to
$\overline{RASIN}$ low: 25 ns (16R6 A PAL Parameter)

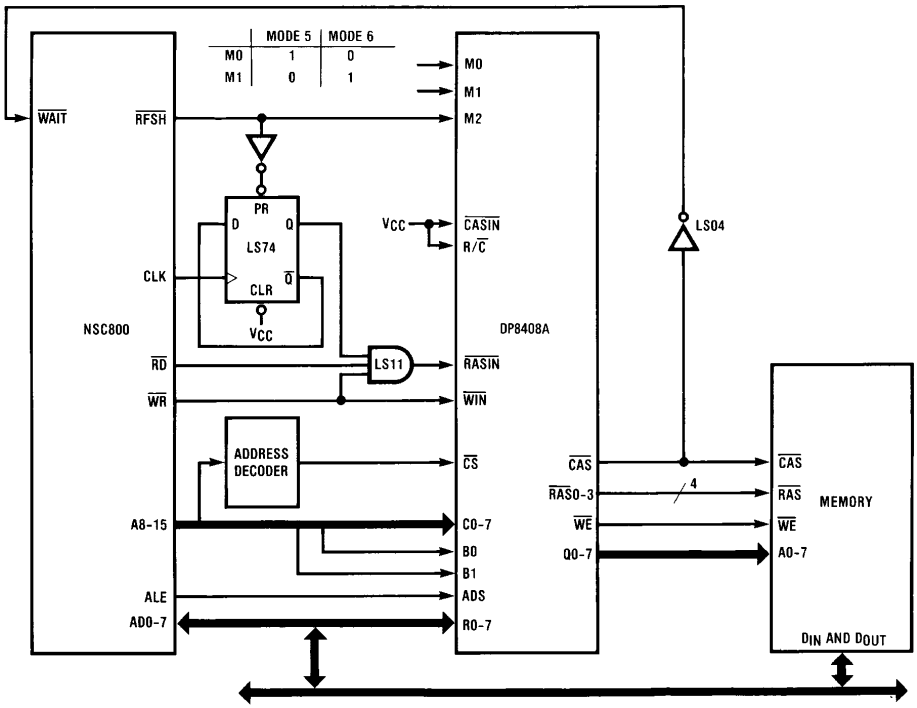
$\overline{RASIN}$ to $\overline{CAS}$
Output low: 160 ns (DP8409A's t_{RICL}, Mode 5, at 500 pF load)
130 ns (DP8409A-2's t_{RICL})

Read data setup time (before E going low): 40 ns

MEMORY REFRESH

As described above, $\overline{RASIN}$ goes active when Q and/or E are high. This scheme, therefore, maximizes chances for hidden refresh because $\overline{CS}$ is high during nondynamic memory cycle. For example, when the CPU is executing internal operation or the CPU is accessing ROM or I/O, $\overline{CS}$ is high during these times. The DP8409A therefore performs a hidden refresh as $\overline{RASIN}$ goes low, assuming that RFCK is high.

However, if no hidden refresh occurs while RFCK was high, RF I/O goes low immediately after the RFCK high-to-low transition requests a forced refresh. The PAL Control Logic samples RF I/O, when E and Q are high and low respectively, to set M2 (RFSH) low, as shown in *Figure 13*. Once M2 has gone low, a forced refresh automatically occurs (as described in the DP8409A Overview). M2 remains low for four system clock periods to allow for this forced refresh. If the current microprocessor cycle is a nondynamic memory cycle ($\overline{CS}$ is high), this refresh is transparent to the microprocessor and $\overline{STRETCH}$ remains high (E and Q are not stretched). Nevertheless, if the current cycle is a dynamic memory access cycle, $\overline{STRETCH}$ goes low stretching E and Q for a maximum of four system clocks. $\overline{RASIN}$ for the pending access will be issued a full system clock after M2 has gone high; this is to allow some $\overline{RAS}$ precharge time for the dynamic RAM. After this, memory will be accessed in the manner as described in the Memory Access Cycle.



TL/F/5040–9

FIGURE 8. NSC800 and DP8408A Interface

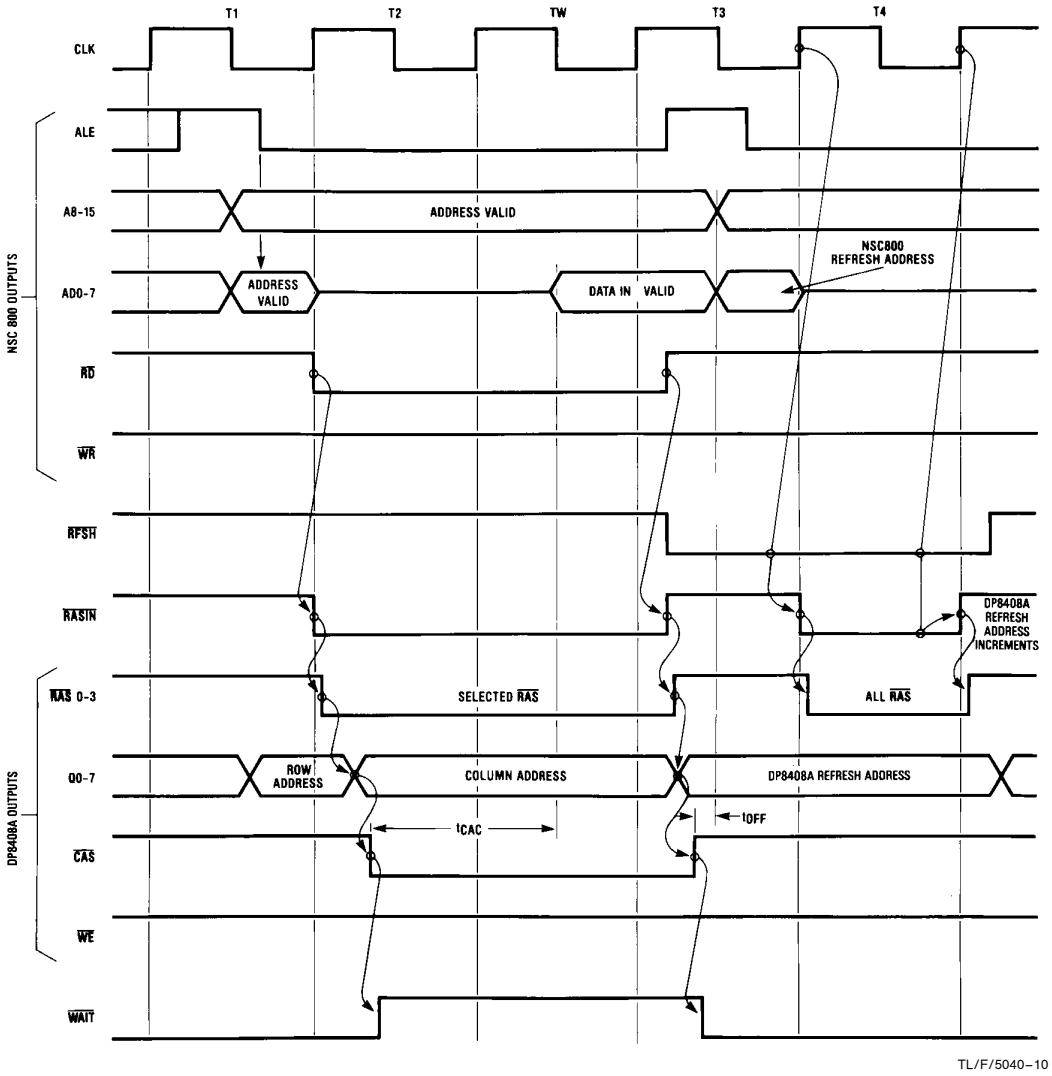
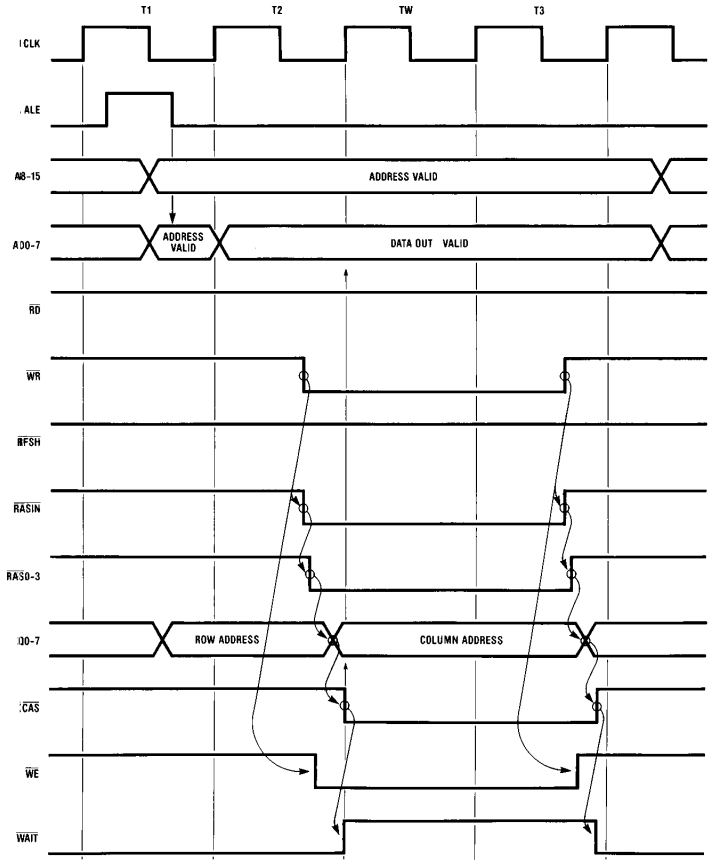
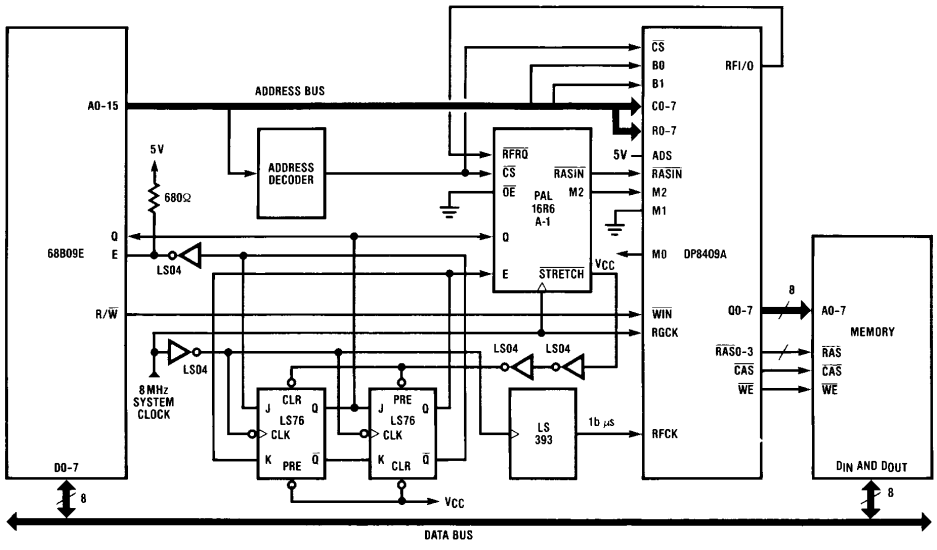


FIGURE 9. NSC800 Op Code Fetch Cycle Showing Memory Refresh



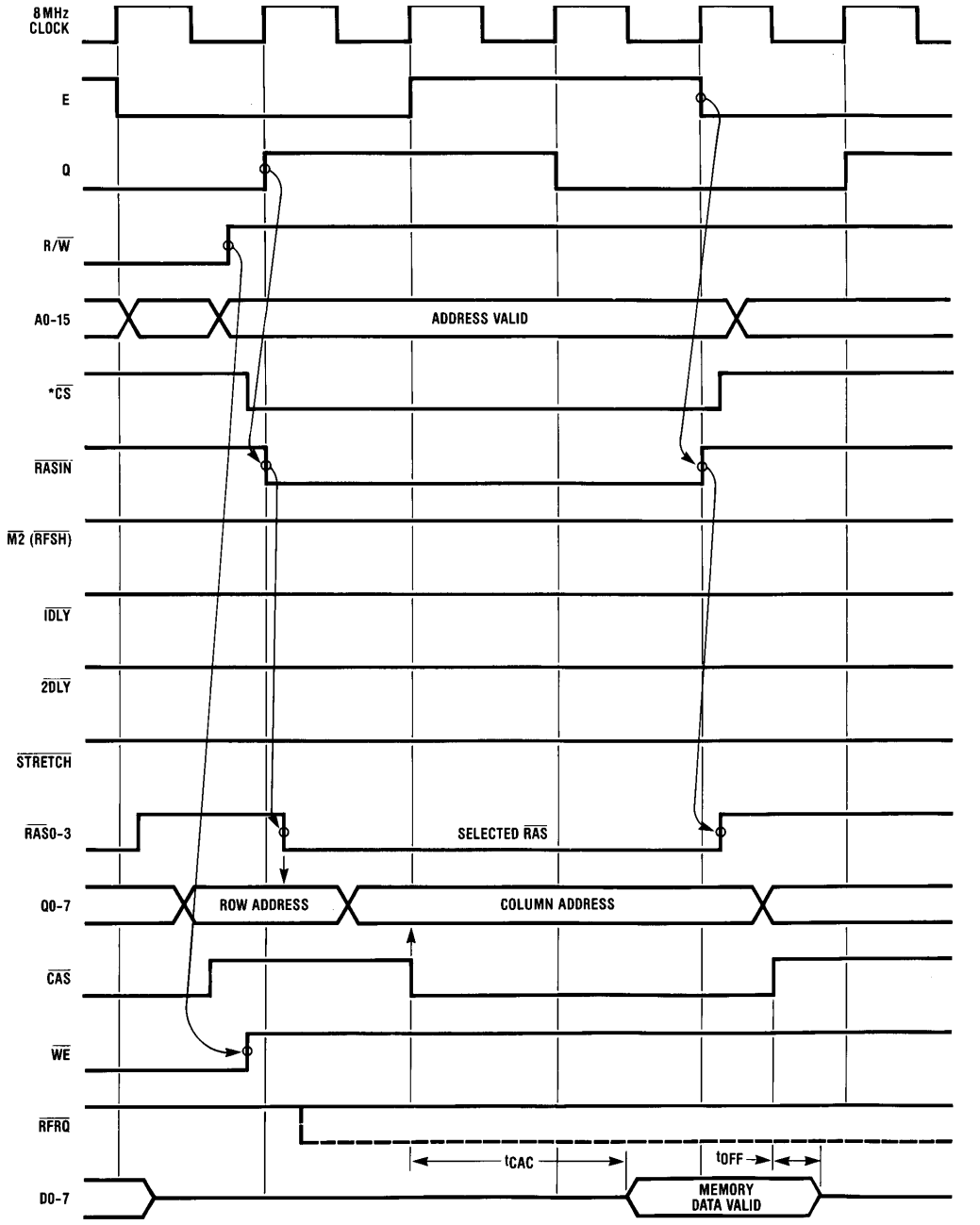
TL/F/5040–11

FIGURE 10. NSC800 Memory Write Cycle



TL/F/5040–12

FIGURE 11. MC68B09E and DP8409A Interface



TL/F/5040–13

*If CS is high throughout this cycle (RFCK is also high), hidden refresh occurs instead of a memory access.

FIGURE 12. MC68B09E Memory Read Cycle

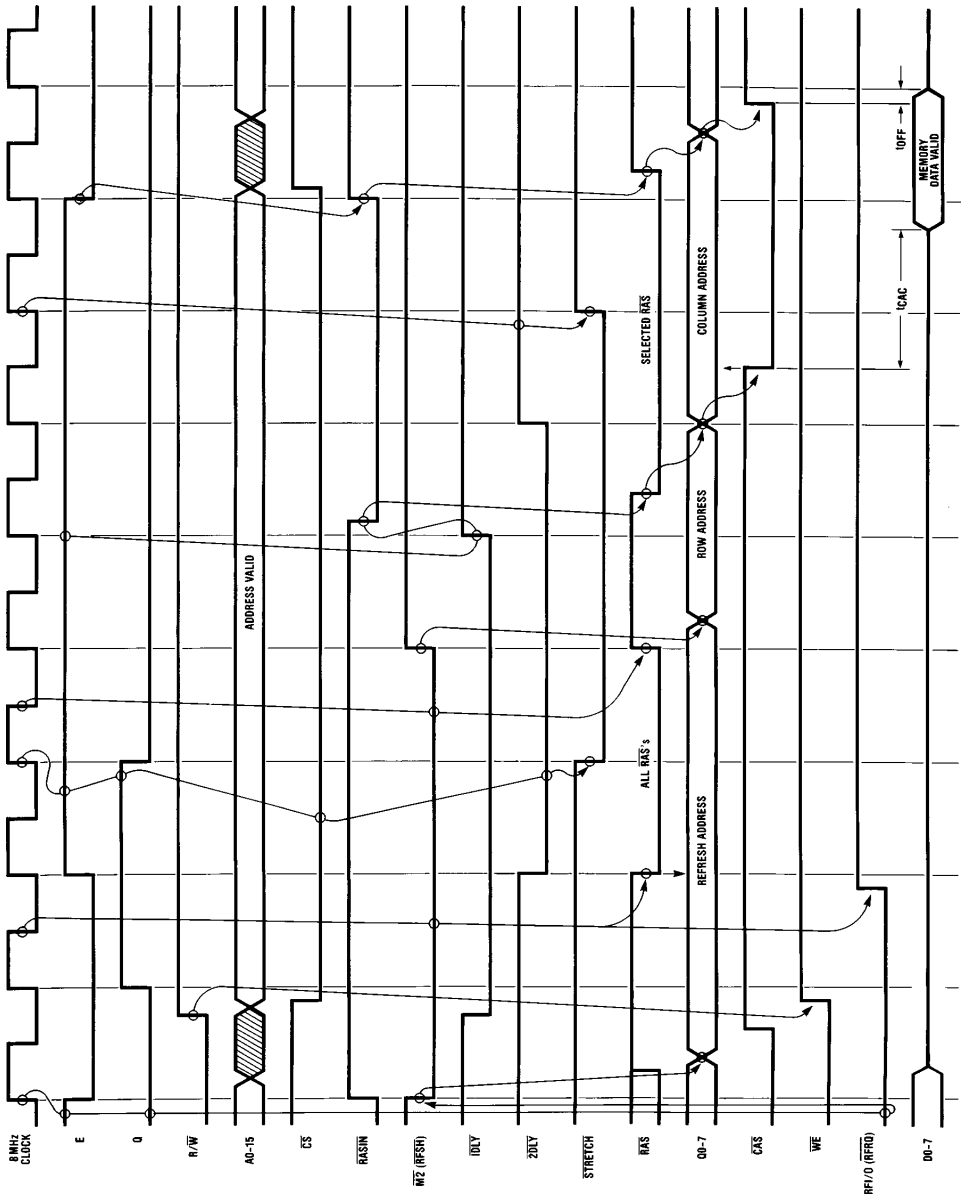
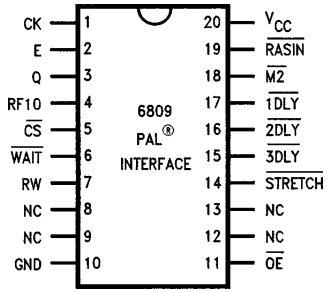


FIGURE 13. MC68B09E Forced Refresh and Memory Read Cycle

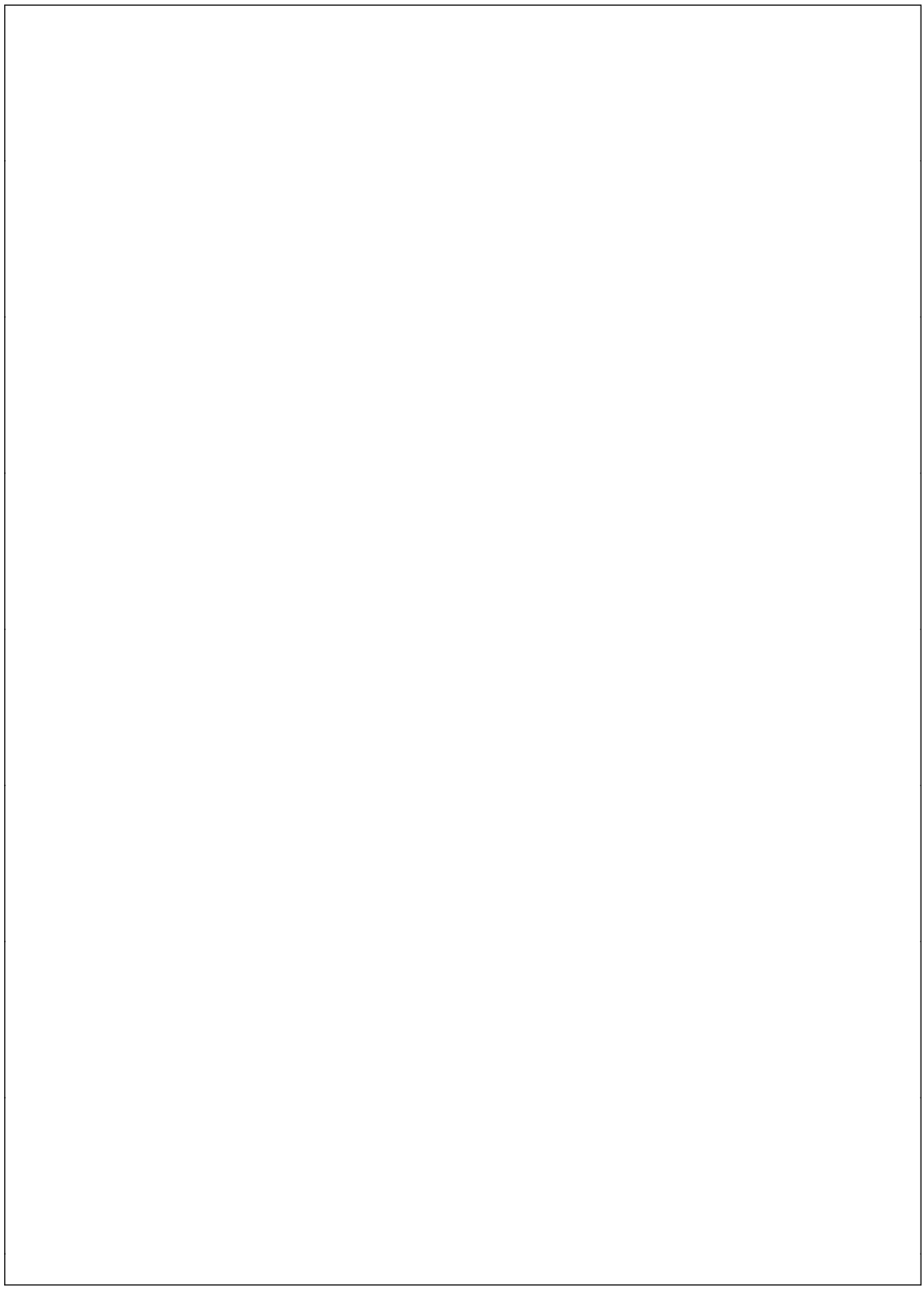
```
PAL16R6A
User Part #
6809/8409A Interface PAL
National Semiconductor
CK E Q RFIO /CS /WAIT RW A B GND /OE C D /STRETCH
/3DLY /2DLY /1DLY /M2 /RASIN VCC
If (VCC) RASIN=CS*E*/M2*/1DLY +
               CS*Q*/M2
M2:=E*/RFIO*/Q +
    M2*/3DLY
1DLY:=M2
2DLY:=1DLY
3DLY:=2DLY
STRETCH:=CS*2DLY*E +
         CS*WAIT*E*Q*RW
;DESCRIPTION:
;The above equations are written in standard PALASM™ format.
;Also included in the logic is a ""/WAIT"" (active low) input. This
;input will allow the insertion of one WAIT state in a READ
;access cycle if it is tied low. If WAIT states are wanted in
;both READ and WRITE access cycles the ""RW"" input in the STRETCH
;equation should be deleted.
;The user should make sure that CS is valid at the DP8409A input a
;minimum of 30 ns before RASIN is valid. If the user does not
;care about the HIDDEN REFRESH feature of the DP8409A, CS can be
;tied permanently low. In this configuration the RASIN term can
;transition whenever is convenient.
```

Dual-In-Line Package



TL/F/5040–15

Top View



LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408