National Semiconductor

PRELIMINARY
October 1990

NS32CG821A microCMOS Programmable 1M Dynamic RAM Controller/Driver

General Description

The NS32CG821A dynamic RAM controller provides a low cost, single chip interface between dynamic RAM and the NS32CG16. The NS32CG821A generates all the required access control signal timing for DRAMs. An on-chip refresh request clock is used to automatically refresh the DRAM array. Refreshes and accesses are arbitrated on chip. If necessary, a $\overline{WAIT}$ output inserts wait states into memory access cycles, including burst mode accesses. $\overline{RAS}$ low time during refreshes and $\overline{RAS}$ precharge time after refreshes and back to back accesses are guaranteed through the insertion of wait states. Separate on-chip precharge counters for each $\overline{RAS}$ output can be used for memory interleaving to avoid delayed back to back accesses because of precharge.

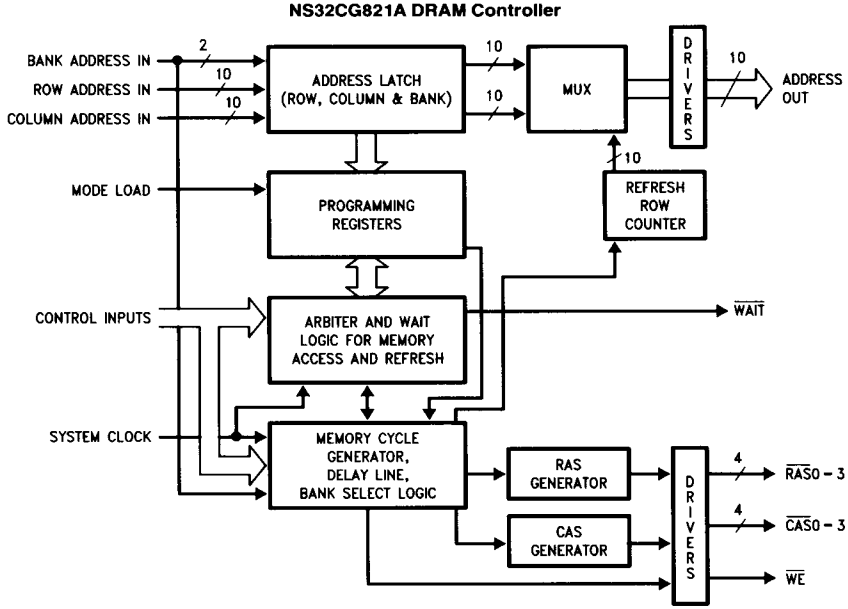
Features

- Allows zero wait state operation
- On chip high precision delay line to guarantee critical DRAM access timing parameters
- microCMOS process for low power
- High capacitance drivers for $\overline{RAS}$, $\overline{CAS}$, $\overline{WE}$ and DRAM address on chip
- On chip support for page and static column DRAMs
- Byte enable signals on chip allow byte writing with no external logic
- Selection of controller speeds: 20 MHz and 25 MHz
- On board access refresh arbitration logic
- Direct interface to the NS32CG16 microprocessor
- 4 $\overline{RAS}$ and 4 $\overline{CAS}$ drivers (the $\overline{RAS}$ and $\overline{CAS}$ configuration is programmable)

Control	# of Pins (PLCC)	# of Address Outputs	Largest DRAM Possible	Direct Drive Memory Capacity
NS32CG821A	68	10	1 Mbit	8 Mbytes

Block Diagram

NS32CG821A DRAM Controller



TL/F/10126-1

TRI-STATE® is a registered trademark of National Semiconductor Corporation.

RRD-B20M100/Printed in U. S. A.

Table of Contents

1.0 Introduction

The NS32CG821A is a CMOS Dynamic RAM controller that incorporates many advanced features including the capabilities of address latches, refresh counter, refresh clock, row, column and refresh address multiplexer, delay line, refresh/access arbitration logic and high capacitive drivers. The programmable system interface allows the NS32CG16 microprocessor to directly interface via the NS32CG821A to DRAM arrays up to 8 Mbytes in size. (See *Figure 3* for an example.)

After power up, the NS32CG821A must first be reset and programmed before accessing the DRAM. The chip is programmed through the address bus.

Resetting

Due to the differences in power supply characteristics, the internal reset circuit may not always reset correctly. Therefore, an external (hardware) reset must be performed before programming the chip.

There are two methods of programming the chip. The first method, mode load only, is accomplished by asserting the signal mode load, $\overline{ML}$. A valid programming selection is presented on the row, column, bank and $\overline{ECAS0}$ inputs, then $\overline{ML}$ is negated. When $\overline{ML}$ is negated, the chip is programmed with the valid programming bits on the address bus.

The second method, chip selected access, is accomplished by asserting $\overline{ML}$ and performing a chip selected access. When $\overline{CS}$ and $\overline{TSO}$ are asserted for the access, the chip is programmed. During this programming access, the programming bits affecting the wait logic become effective immediately, allowing the access to terminate. After the access, $\overline{ML}$ is negated and the rest of the programming bits take effect.

Once the NS32CG821A has been programmed, a 60 ms initialization period is entered. During this time, the NS32CG821A controller performs refreshes to the DRAM array so further DRAM warm up cycles are unnecessary.

To access the DRAM, the signal ALE is asserted along with $\overline{CS}$ to ensure a valid DRAM access. ALE asserting sets an internal latch and only needs to be pulsed and not held throughout the entire access. Once $\overline{CS}$ and ALE are both asserted, $\overline{WAIT}$ is asserted, unless $\overline{WAIT}$ is programmed as OT and a non-delayed access occurs, and follows $\overline{CS}$ until the rising clock edge. This is not a problem since the 32CG16 will not process the $\overline{WAIT}$ signal until the end of state T2. $\overline{WAIT}$ only has to guarantee that it meets the setup time to this edge of CLK on which it is sampled. On the next rising clock edge, $\overline{RAS}$ will be asserted for that access. The NS32CG821A will place the row address on the DRAM address bus, guarantee the programmed value of row address hold time of the DRAM, place the column address on the DRAM address bus, guarantee the programmed value of column address setup time and assert $\overline{CAS}$. $\overline{TSO}$ can be asserted anytime after the clock edge which starts the access $\overline{RAS}$. $\overline{RAS}$ and $\overline{CAS}$ will extend until $\overline{TSO}$ is negated.

The NS32CG821A has greatly expanded refresh capabilities compared to other DRAM controllers.

When using internal automatic refreshing, the NS32CG821A will generate an internal refresh request from the refresh request clock. The NS32CG821A will arbitrate between the refresh requests and accesses. Assuming an access is not currently in progress, a refresh will occur and on the next positive clock edge, refreshing will begin. If an access had been in progress, the refresh will begin after the access has terminated.

The controller has two types of refreshing available: conventional and staggered. Any refresh control mode can be used with internal refreshing. In a conventional refresh, all of the $\overline{RAS}$ outputs will be asserted and negated at once. In a staggered refresh, the $\overline{RAS}$ outputs will be asserted one positive clock edge apart.

The NS32CG821A has wait support available as programmable $\overline{WAIT}$, which connects directly to the NS32CG16 $\overline{CWAIT}$ pin. This signal is used by the on-chip arbiter to insert wait states to guarantee the arbitration between accesses and refreshes or precharge.

$\overline{WAIT}$ is asserted during the start of the access (ALE and $\overline{CS}$) and will negate a number of clock edges from the event that starts the access $\overline{RAS}$. After $\overline{WAIT}$ is negated, it will stay negated until the next access. $\overline{WAIT}$ can also be programmed to toggle with $\overline{ECAS}$ inputs during a burst/page mode access.

$\overline{WAIT}$ can be dynamically delayed further through the $\overline{WAITIN}$ signal to the NS32CG821A.

The NS32CG821A has address latches, used to latch the bank, row and column address inputs. Once the address is latched, a column increment feature can be used to increment the column address. The address latches can also be programmed to be fall through.

The $\overline{RAS}$ and $\overline{CAS}$ drivers can be configured to drive a one, two, four or eight bank memory array. The $\overline{ECAS}$ signals can then be used to select one of four $\overline{CAS}$ drivers for byte writing with no external logic.

When configuring the NS32CG821A for more than one bank, memory interleaving can be used. By tying the low order address bits to the bank select lines, B0 and B1, sequential back to back accesses will not be delayed since the NS32CG821A has separate precharge counters per bank.

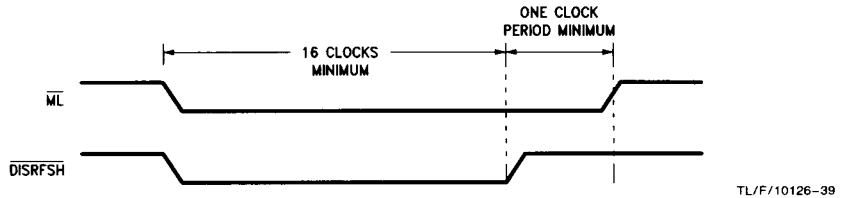
The following explains the terminology used in this data sheet. The terms negated and asserted are used. Asserted refers to a "true" signal. Thus, "$\overline{ECAS0}$ asserted" means the $\overline{ECAS0}$ input is at a logic 0. The term "COLINC asserted" means the COLINC input is at a logic 1. The term negated refers to a "false" signal. Thus, "$\overline{ECAS0}$ negated" means the $\overline{ECAS0}$ input is at a logic 1. The term "COLINC negated" means the input COLINC is at a logic 0. The table shown below clarifies this terminology.

Signal	Action	Logic Level
Active High	Asserted	High
Active High	Negated	Low
Active Low	Asserted	Low
Active Low	Negated	High

External Reset

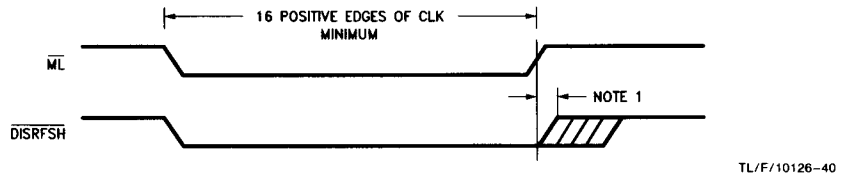
The power up state can again be entered by asserting $\overline{ML}$ and $\overline{DISRFSH}$ for a minimum of 16 positive edges of CLK. After resetting if the user negates $\overline{DISRFSH}$ at least one CLK period before negating $\overline{ML}$ as shown in *Figure 1A*, $\overline{ML}$ negated will program the chip. If $\overline{ML}$ is negated before $\overline{DISRFSH}$ as shown in *Figure 1B*, the chip will not be programmed. After the chip is programmed, the 60 ms initialization period will be entered into if this is the first programming after power up or reset.

1.0 Introduction (Continued)



TL/F/10126–39

FIGURE 1A. Chip Reset and Programmed

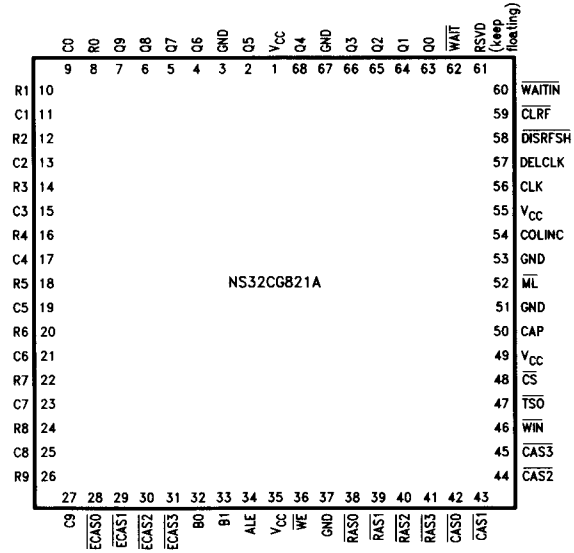


TL/F/10126–40

FIGURE 1B. Chip Reset but Not Programmed

Note 1: The positive going edge of DISRFSH needs to be delayed a few nanoseconds after the positive going edge of ML. Any gate delay is sufficient.

Connection Diagram

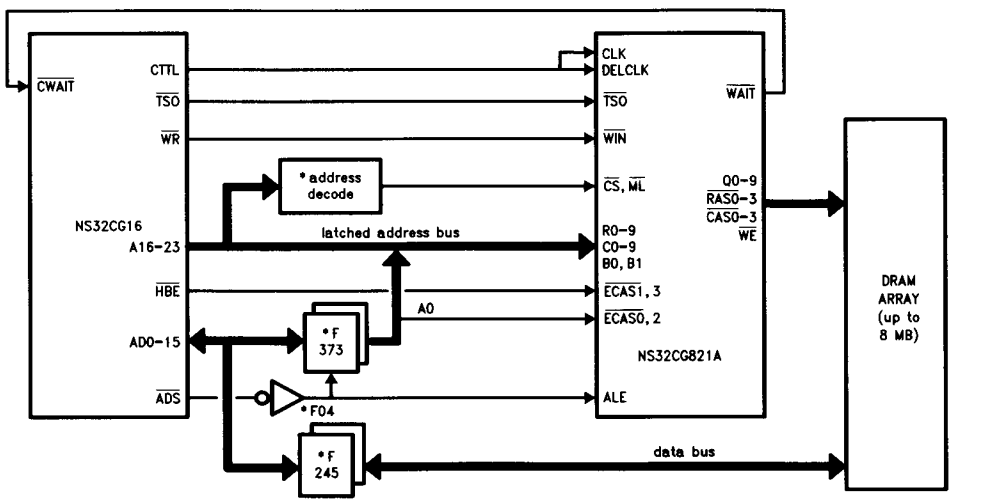


TL/F/10126–2

Top View

FIGURE 2

Order Number NS32CG821AV-20 or NS32CG821AV-25
See NS Package Number V68A



TL/F/10126–48

*Standard components in any NS32CG16 design

FIGURE 3. NS32CG16–NS32CG821A Connection Diagram (Note 1)

Note 1: This is only one possible way to connect the NS32CG821A. See Sections 7 and 8 for additional options and configurations for hookup.

2.0 Signal Descriptions

Pin Name	Input/ Output	Description
2.1 ADDRESS, R/W AND PROGRAMMING SIGNALS		
R0–9	I	**ROW ADDRESS:** These inputs are used to specify the row address during an access to the DRAM. They are also used to program the chip when $\overline{ML}$ is asserted (except R10).
C0–9	I	**COLUMN ADDRESS:** These inputs are used to specify the column address during an access to the DRAM. They are also used to program the chip when $\overline{ML}$ is asserted (except C10).
B0, B1	I	**BANK SELECT:** Depending on programming, these inputs are used to select a group of $\overline{RAS}$ and $\overline{CAS}$ outputs to assert during an access. They are also used to program the chip when $\overline{ML}$ is asserted.
$\overline{ECAS}$0–3	I	**ENABLE $\overline{CAS}$:** These inputs are used to enable a single or group of $\overline{CAS}$ outputs when asserted. In combination with the B0, B1 and the programming bits, these inputs select which $\overline{CAS}$ output or $\overline{CAS}$ outputs will assert during an access. The $\overline{ECAS}$ signals can also be used to toggle a group of $\overline{CAS}$ outputs for page mode accesses. They also can be used for byte write operations.
$\overline{WIN}$	I	**WRITE ENABLE IN:** This input is used to signify a write operation to the DRAM. The $\overline{WE}$ output follows this input. This input asserted will also cause $\overline{CAS}$ to delay to the next positive clock edge if address bit C9 is asserted during programming.
COLINC	I	**COLUMN INCREMENT:** When the address latches are used, this input functions as COLINC. Asserting this signal causes the column address to be incremented by one.
$\overline{ML}$	I	**MODE LOAD:** This input signal, when low, enables the internal programming register that stores the programming information.
2.2 DRAM CONTROL SIGNALS		
Q0–9	O	**DRAM ADDRESS:** These outputs are the multiplexed output of the R0–9 and C0–9 and form the DRAM address bus. These outputs contain the refresh address whenever refreshing is taking place. They contain high capacitive drivers with 20Ω series damping resistors.
$\overline{RAS}$0–3	O	**ROW ADDRESS STROBES:** For an access, these outputs are asserted to latch the row address contained on the outputs Q0–9 into the DRAM. For refreshing, the $\overline{RAS}$ outputs are used to latch the refresh row address contained on the Q0–9 outputs in the DRAM. These outputs contain high capacitive drivers with 20Ω series damping resistors.
$\overline{CAS}$0–3	O	**COLUMN ADDRESS STROBES:** These outputs are asserted to latch the column address contained on the outputs Q0–9 into the DRAM. These outputs have high capacitive drivers with 20Ω series damping resistors.
$\overline{WE}$	O	**WRITE ENABLE:** This output asserted specifies a write operation to the DRAM. When negated, this output specifies a read operation to the DRAM. This output has a high capacitive driver and a 20Ω series damping resistor.
2.3 REFRESH SIGNALS		
$\overline{CLRF}$	I	**CLEAR REFRESH:** This pin, in conjunction with $\overline{DISRFSH}$ is used to clear the internal refresh counter.
$\overline{DISRFSH}$	I	**DISABLE REFRESH:** When asserted with $\overline{ML}$ asserted for 16 positive edges of clock, the entire chip is reset and when negated with $\overline{CLRF}$ asserted clears the internal refresh address counter.
2.4 MEMORY ACCESS		
ALE	I	**ADDRESS LATCH ENABLE:** When ALE asserted along with $\overline{CS}$ causes an internal latch to be set. Once this latch is set and precharge time has been met an access will start from the positive clock edge of CLK as soon as possible. If Address Latch (B_0 = 0) is programmed, the low going edge of this signal latches the bank, row, and column address.
$\overline{CS}$	I	**CHIP SELECT:** This input signal must be asserted to enable an access.
$\overline{TSO}$	I	**TIMING STATE OUTPUT:** This input signal must be asserted some time after the first positive clock edge after ALE has been asserted. When this signal is negated, $\overline{RAS}$ is negated for the access.

2.0 Signal Descriptions (Continued)

Pin Name	Input/ Output	Description
2.4 MEMORY ACCESS (Continued)		
$\overline{WAIT}$	O	**WAIT:** This output can be programmed to insert wait states into a CPU access cycle. This signal can be delayed by a number of positive clock edges or negative clock levels of CLK, depending on how it is programmed, to increase the microprocessor's access cycle through the insertion of wait states.
$\overline{WAITIN}$	I	**WAIT INCREASE:** This input can be used to dynamically increase the number of positive clock edges of CLK until $\overline{WAIT}$ will be negated during a DRAM access.
2.5 POWER SIGNALS AND CAPACITOR INPUT		
V_{CC}	I	**POWER:** Supply Voltage.
GND	I	**GROUND:** Supply Voltage Reference.
CAP	I	**CAPACITOR:** This input is used by the internal PLL for stabilization. The value of the ceramic capacitor should be 0.1 μF and should be connected between this input and ground.
2.6 CLOCK INPUTS There are two clock inputs to the NS32CG821A, CLK and DELCLK. These two clocks may both be tied to the same clock input, or they may be two separate clocks, running at different frequencies, asynchronous to each other.		
CLK	I	**SYSTEM CLOCK:** This input may be in the range of 0 Hz up to 25 MHz. This input is generally a constant frequency but it may be controlled externally to change frequencies or perhaps be stopped for some arbitrary period of time. This input provides the clock to the internal state machine that arbitrates between accesses and refreshes. This clock's positive edges and negative levels are used to extend the $\overline{WAIT}$ signal. This clock is also used as the reference for the $\overline{RAS}$ precharge time and $\overline{RAS}$ low time during refresh. All memory accesses are assumed to be synchronous to the system clock CLK.
DELCLK	I	**DELAY LINE CLOCK:** The clock input DELCLK, may be in the range of 6 MHz to 20 MHz and should be a multiple of 2 (i.e., 6, 8, 10, 12, 14, 16, 18, 20 MHz) to have the NS32CG821A switching characteristics hold. If DELCLK is not one of the above frequencies the accuracy of the internal delay line will suffer. This is because the phase locked loop that generates the delay line assumes an input clock frequency of a multiple of 2 MHz. For example, if the DELCLK input is at 7 MHz and we choose a divide by 3 (program bits C0–2) this will produce 2.333 MHz which is 16.667% off of 2 MHz. Therefore, the NS32CG821A delay line would produce delays that are shorter (faster delays) than what is intended. If divide by 4 was chosen the delay line would be longer (slower delays) than intended (1.75 MHz instead of 2 MHz). (See Section 10 for more information.) This clock is also divided to create the internal refresh clock.

3.0 Memory Access

An access to DRAM is initiated by two input signals: ALE and $\overline{CS}$. The access is always terminated by one signal: $\overline{TSO}$. These input signals should be synchronous to the input clock, CLK. Once an access has been requested by $\overline{CS}$ and ALE, the NS32CG821A will guarantee the following:

The NS32CG821A will have the row address valid to the DRAMs' address bus, Q0–9 given that the row address set-up time to the NS32CG821A was met;

The NS32CG821A will bring the appropriate $\overline{RAS}$ or $\overline{RAS}$s low;

The NS32CG821A will guarantee the minimum row address hold time, before switching the internal multiplexer to place the column address on the DRAM address bus, Q0–9;

The NS32CG821A will guarantee the minimum column address setup time before asserting the appropriate $\overline{CAS}$ or $\overline{CAS}$s;

The NS32CG821A will hold the column address valid the minimum specified column address hold time.

The memory access shown in *Figure 4* is selected by negating the input B1 during programming. This access mode al-

3.0 Memory Access (Continued)

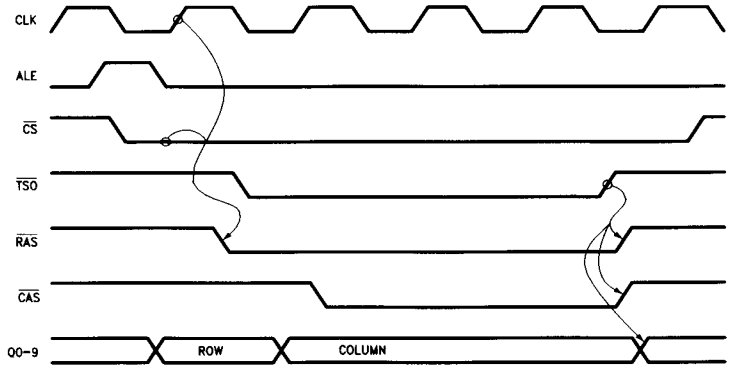


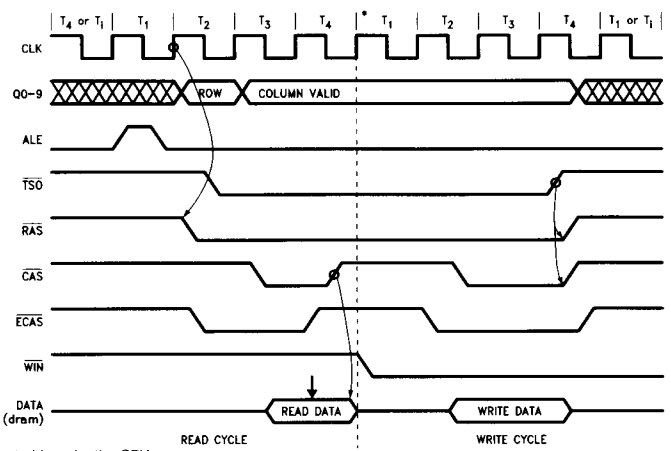
FIGURE 4. Memory Access

lows accesses to DRAM to always be initiated from the positive edge of the system input clock, CLK. To initiate an access, ALE is pulsed high and $\overline{CS}$ is asserted. Pulsing ALE high and asserting $\overline{CS}$, sets an internal latch which requests an access. If the precharge time from the last access or DRAM refresh had been met and a refresh of DRAM was not in progress, the $\overline{RAS}$ or group of $\overline{RAS}$s would be initiated from the first positive edge of CLK. If a DRAM refresh is in progress or precharge time is required, the controller will wait until these events have taken place and assert $\overline{RAS}$ on the next positive edge of CLK.

Once ALE and $\overline{CS}$ are both asserted, $\overline{WAIT}$ is asserted, unless $\overline{WAIT}$ is programmed as OT and a non-delayed access occurs, and follows $\overline{CS}$ until the rising clock edge. This is not a problem since the 32CG16 will not process the $\overline{WAIT}$ signal until the end of state T2. $\overline{WAIT}$ only has to guarantee that it meets the setup time to this edge of CLK on which it is sampled.

Sometime after the first positive edge of CLK after ALE and $\overline{CS}$ have been asserted, the input $\overline{TSO}$ must be asserted. Once $\overline{TSO}$ has been asserted, $\overline{CS}$ can be negated. Once $\overline{TSO}$ is negated, $\overline{RAS}$ and $\overline{CAS}$ will be negated. ALE can stay asserted several periods of CLK. However, ALE must be negated before or during the period of CLK in which $\overline{TSO}$ is negated.

There are 2 methods by which this chip can be used to do read-modify-write access cycles. The first method involves doing a late write access where the $\overline{WIN}$ input is asserted some delay after $\overline{CAS}$ is asserted. The second method involves doing a page mode read access followed by a page mode write access with $\overline{RAS}$ held low (see *Figure 5*).



*There may be idle states inserted here by the CPU.

TL/F/10126-5

FIGURE 5. Read-Modify-Write Access Cycle

3.0 Memory Access (Continued)

$\overline{CAS}$n must be toggled using the $\overline{ECAS}$n inputs and $\overline{WIN}$ has to be changed from negated to asserted (read to write) while $\overline{CAS}$ is negated. This method is better than changing $\overline{WIN}$ from negated to asserted in a late write access because here a problem may arise with DATA IN and DATA OUT being valid at the same time. This may result in a data line trying to drive two different levels simultaneously. The page mode method of a read-modify-write access allows the user to have transceivers in the system because the data in (read data) is guaranteed to be high impedance during the time the data out (write data) is valid.

4.0 Refresh Options

The NS32CG821A supports automatic internally controlled refresh. Different types of refreshes can be performed. These different types include all $\overline{RAS}$ refresh and staggered refresh.

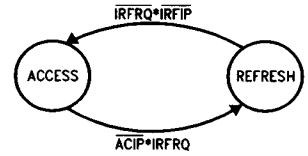
There are two inputs, $\overline{DISRFSH}$ and $\overline{CLRF}$, associated with refresh. There are also ten programming bits; R0–1, R9, C0–6 and $\overline{ECAS}$0 used to program the various types of refreshing.

The NS32CG821A will increment the refresh address counter automatically. The refresh address counter will be incremented once all the refresh $\overline{RAS}$s have been negated.

In every combination of internal refresh and refresh type, the NS32CG821A is programmed to keep $\overline{RAS}$ asserted a number of CLK periods. The values of $\overline{RAS}$ low time during refresh are programmed with the programming bits R0 and R1.

4.1 AUTOMATIC INTERNAL REFRESH

The NS32CG821A has an internal refresh clock. The period of the refresh clock is generated from the programming bits C0–3. Every period of the refresh clock, an internal refresh request is generated. As long as a DRAM access is not currently in progress and precharge time has been met, the internal refresh request will generate an automatic internal refresh. If a DRAM access is in progress, the NS32CG821A on-chip arbitration logic will wait until the access is finished before performing the refresh. The refresh/access arbitration logic can insert a refresh cycle between two accesses. If the two accesses are back to back, the arbitration logic can insert a refresh cycle into the beginning of the next access. The CPU will wait to complete that access until the refresh cycle is completed. However, the refresh arbitration logic can not interrupt an access cycle in progress to perform a refresh. To enable automatic internally controlled refreshes, the input $\overline{DISRFSH}$ must be negated.



TL/F/10126–6

Explanation of Terms

IRFRQ = Internal ReFresh ReQuest of the NS32CG821A. IRFRQ has the ability to hold off a pending access.

IRFIP = Internal ReFresh In Progress

ACIP = ACcess In Progress. This means that either RAS is low for an access or is in the process of transitioning low for an access.

FIGURE 6. NS32CG821A Access/Refresh Arbitration State Program

4.2 REFRESH CYCLE TYPES

Two different types of refresh cycles are available for use. The two different types are mutually exclusive and can be used with the internal refresh control. The two different refresh cycle types are: all $\overline{RAS}$ refresh and staggered $\overline{RAS}$ refresh. In all refresh cycle types, the $\overline{RAS}$ precharge time is guaranteed: between the previous access $\overline{RAS}$ ending and the refresh $\overline{RAS}$0 starting; between refresh $\overline{RAS}$3 ending and access $\overline{RAS}$ beginning; between burst refresh $\overline{RAS}$s.

4.2.1 Conventional $\overline{RAS}$ Refresh

A conventional refresh cycle causes $\overline{RAS}$0–3 to all assert from the first positive edge of CLK after refresh begins as shown in *Figure 8*. $\overline{RAS}$0–3 will stay asserted until the number of positive edges of CLK programmed have passed. On the last positive edge, $\overline{RAS}$0–3 will be negated and the refresh cycle will end. This type of refresh cycle is programmed by negating address bit R9 during programming.

4.2.2 Staggered $\overline{RAS}$ Refresh

A staggered refresh staggers each $\overline{RAS}$ or group of $\overline{RAS}$s by a positive edge of CLK as shown in *Figure 9*. The number of $\overline{RAS}$s, which will be asserted on each positive edge of CLK, is determined by the $\overline{RAS}$, $\overline{CAS}$ configuration mode programming bits C4–C6. If single $\overline{RAS}$ outputs are selected during programming, then each $\overline{RAS}$ will assert on successive positive edges of CLK. If two $\overline{RAS}$ outputs are selected during programming then $\overline{RAS}$0 and $\overline{RAS}$1 will assert on the first positive edge of CLK after refresh cycle begins. $\overline{RAS}$2 and $\overline{RAS}$3 will assert on the second positive edge of CLK after refresh cycle begins. If all $\overline{RAS}$ outputs were selected during programming, all $\overline{RAS}$ outputs would assert on the first positive edge of CLK after refresh cycle begins. Each $\overline{RAS}$ or group of $\overline{RAS}$s will meet the programmed $\overline{RAS}$ low time and then negate.

4.0 Refresh Options (Continued)

TL/F/10126–7

FIGURE 7. Automatic Internal Refresh with Internal Refresh Request (3T of $\overline{RAS}$ low during refresh programmed)

TL/F/10126–8

FIGURE 8. Conventional $\overline{RAS}$ Refresh

TL/F/10126–9

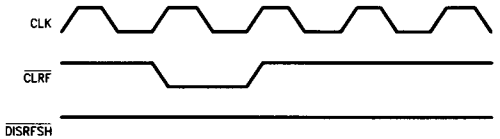
FIGURE 9. Staggered $\overline{RAS}$ Refresh

*$\overline{IRFRQ}$ and $\overline{IRFIP}$ are internal variables, they are only shown in the timing diagrams to make the diagrams easier to understand.

4.0 Refresh Options (Continued)

4.3 CLEARING THE REFRESH ADDRESS COUNTER

The refresh address counter can be cleared by asserting $\overline{CLRF}$ while $\overline{DISRFSH}$ is negated as shown in *Figure 10*. This can be used prior to a burst refresh of the entire memory array. An end-of-count signal can be generated from the Q DRAM address outputs of the NS32CG821A and used to negate $\overline{CLRF}$.



TL/F/10126–10

FIGURE 10. Clearing the Refresh Address Counter

5.0 Wait State Support

Wait states allow a CPU's access cycle to be increased by one or multiple CPU clock periods. By increasing the CPU's access cycle, all signals associated with that access cycle are extended. The CPU samples a wait line to determine if another clock period should be inserted into the access cycle. If another clock period is inserted, the CPU will continue to sample the input every CPU clock period until the input signal changes polarity, allowing the CPU access cycle to terminate. The user determines which value to select for $\overline{WAIT}$ depending upon the CPU speed used and where the user wants the CPU to sample its wait input during an access cycle.

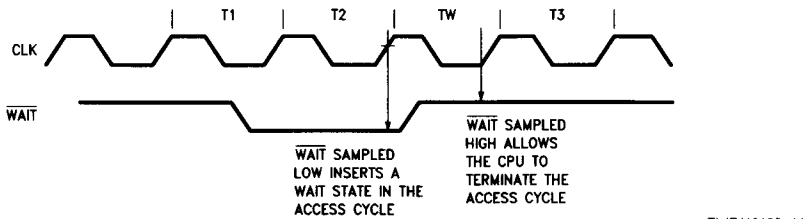
The decision to terminate the CPU access cycle is directly affected by the speed of the DRAMs used. The system designer must ensure that the data from the DRAMs will be present for the CPU to sample or that the data has been written to the DRAM before allowing the CPU access cycle to terminate.

The insertion of wait states also allows a CPU's access cycle to be extended until the DRAM access has taken place. The NS32CG821A inserts wait states into CPU access cycles due to; guaranteeing precharge time, refresh currently in progress, user programmed wait states, and the $\overline{WAITIN}$ signal being asserted. If one of these events is taking place and the CPU starts an access, the NS32CG821A will insert wait states into the access cycle, thereby increasing the length of the CPU's access. Once the event has been completed, the NS32CG821A will allow the access to take place and stop inserting wait states.

There are six programming bits, R2–R7; an input, $\overline{WAITIN}$; and an output that functions as $\overline{WAIT}$.

5.1 WAIT OUTPUT

If $\overline{WAIT}$ is sampled asserted by the CPU, wait states (extra clock periods) are inserted into the current access cycle as shown in *Figure 11*. Once $\overline{WAIT}$ is sampled negated, the access cycle is completed by the CPU. $\overline{WAIT}$ is asserted at the beginning of a chip selected access and is programmed to negate a number of positive edges and/or negative levels of CLK from the event that starts the access. $\overline{WAIT}$ can also be programmed to function in page/burst mode applications. Once $\overline{WAIT}$ is negated during an access, and the $\overline{ECAS}$ inputs are negated with $\overline{TSO}$ asserted, $\overline{WAIT}$ can be programmed to toggle, following the $\overline{ECAS}$ inputs. Once $\overline{TSO}$ is negated, ending the access, $\overline{WAIT}$ will stay negated until the next chip selected access.



TL/F/10126–11

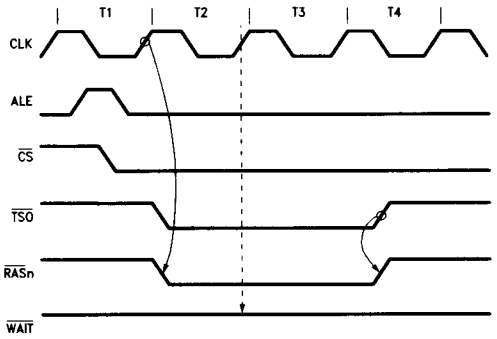
FIGURE 11. $\overline{WAIT}$ Type Output

5.0 Wait State Support (Continued)

5.1.1 Wait during Single Accesses

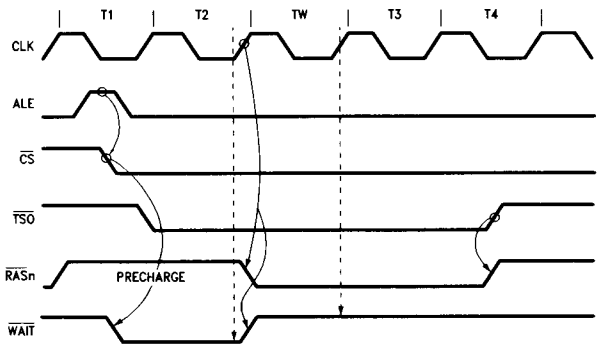
$\overline{WAIT}$ can be programmed to delay a number of positive edges and/or negative levels of CLK. These options are programmed through address bits R2 and R3 at programming time. The user is given four options described below.

0T during non delayed and delayed acceses: $\overline{WAIT}$ will stay negated during a non-delayed access as shown in *Figure 12*. During an access that is delayed, $\overline{WAIT}$ will assert at the start of the access ($\overline{CS}$ and ALE) and negate from the positive edge of $\overline{CLK}$ that starts $\overline{RAS}$ for that access as shown in *Figure 13*.



TL/F/10126–12

FIGURE 12. Non-Delayed Access with $\overline{WAIT}$ 0T ($\overline{WAIT}$ is Sampled at the End of the "T2" Clock State)

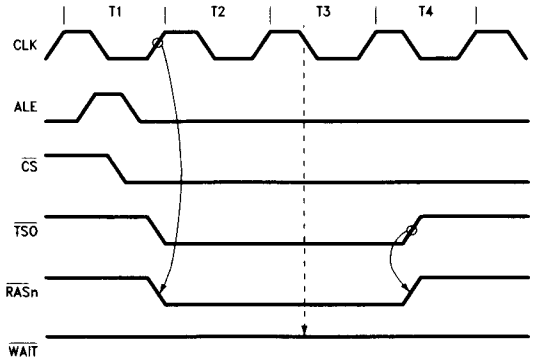


TL/F/10126–13

FIGURE 13. Delayed Access with $\overline{WAIT}$ 0T
("2T" $\overline{RAS}$ Precharge, $\overline{WAIT}$ is Sampled at the End of the "T2" Clock State)

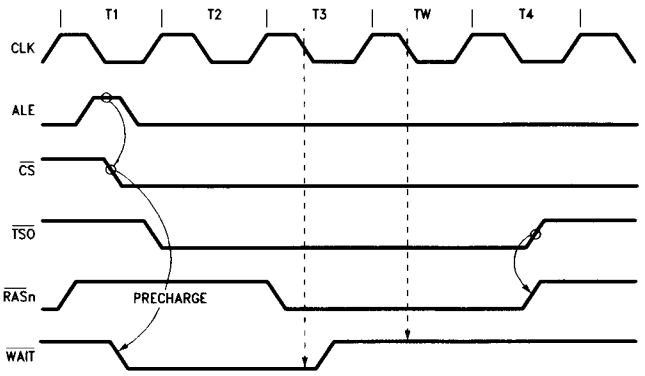
5.0 Wait State Support (Continued)

0T during non-delayed accesses and ½T during delayed accesses: $\overline{WAIT}$ will stay negated during a non-delayed access as shown in *Figure 14*. During an access that is delayed, $\overline{WAIT}$ will assert at the start of the access ($\overline{CS}$ and ALE) and negate on the negative level of CLK after the positive edge of CLK that asserted $\overline{RAS}$ for that access as shown in *Figure 15*.



TL/F/10126–14

FIGURE 14. Non-Delayed Access with $\overline{WAIT}$ 0T ($\overline{WAIT}$ is Sampled at the "T3" Falling Clock Edge)

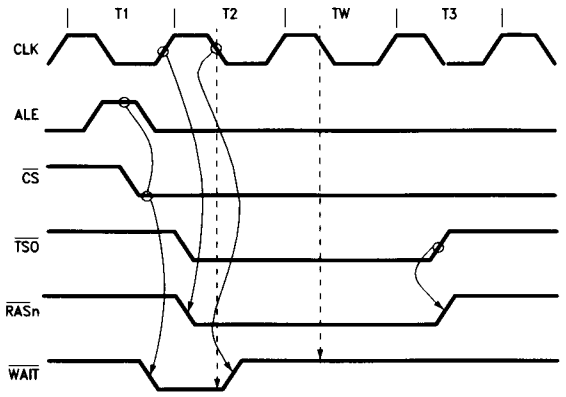


TL/F/10126–15

FIGURE 15. Delayed Access with $\overline{WAIT}$ ½T ($\overline{WAIT}$ is Sampled at the "T3" Falling Clock Edge)

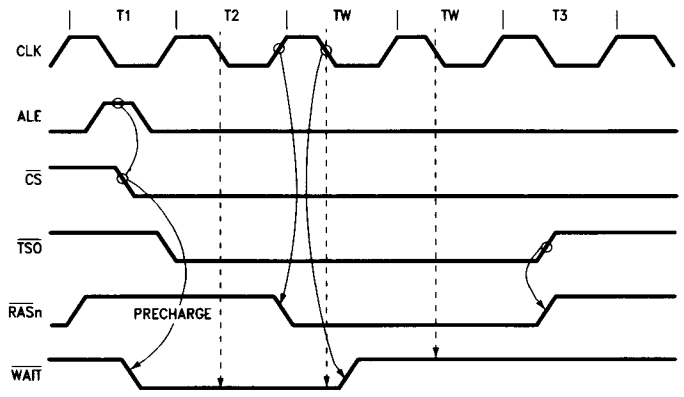
5.0 Wait State Support (Continued)

½T during non-delayed and delayed accesses: $\overline{WAIT}$ will assert when ALE is asserted and $\overline{CS}$ is asserted. $\overline{WAIT}$ will then negate on the negative level of CLK after the positive edge of CLK that asserts $\overline{RAS}$ for the access as shown in *Figure 16*. During delayed accesses in both modes, $\overline{WAIT}$ will assert at the start of the access and negate on the negative level of CLK after the positive edge of CLK that started $\overline{RAS}$ for that access as shown in *Figure 17*.



TL/F/10126–16

FIGURE 16. Non-Delayed Access with $\overline{WAIT}$ ½T ($\overline{WAIT}$ is Sampled at the "T2" Falling Clock Edge)

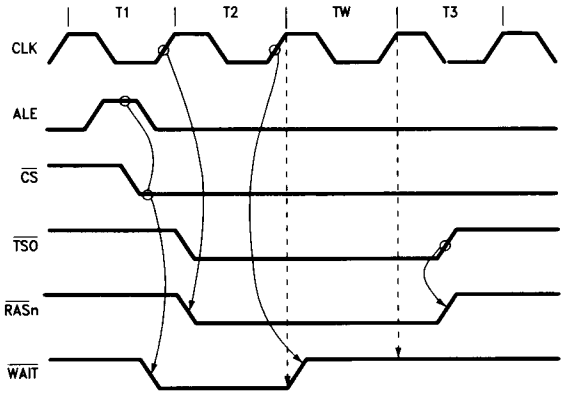


TL/F/10126–17

FIGURE 17. Delayed Access with $\overline{WAIT}$ ½T ($\overline{WAIT}$ is Sampled at the "T2" Falling Clock Edge)

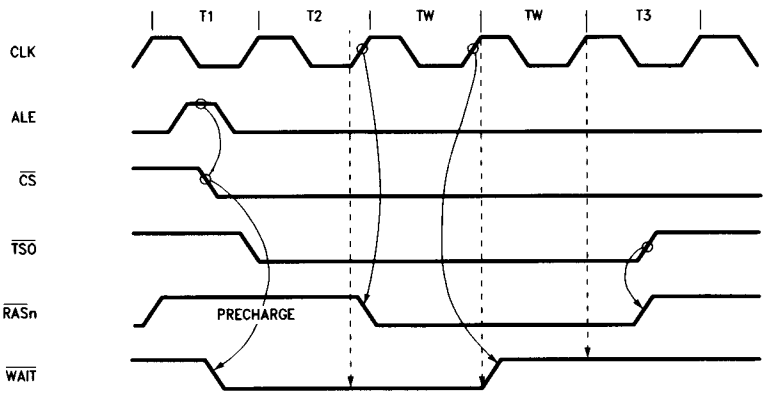
5.0 Wait State Support (Continued)

1T during non-delayed and delayed accesses. $\overline{WAIT}$ will assert from ALE asserted and $\overline{CS}$ asserted. $\overline{WAIT}$ will negate from the next positive edge of CLK that asserts $\overline{RAS}$ for the access as shown in *Figure 18*. During delayed accesses, $\overline{WAIT}$ will assert at the beginning of the access and will negate on the first positive edge of CLK after the positive edge of CLK that starts $\overline{RAS}$ for the access as shown in *Figure 19*.



TL/F/10126–18

FIGURE 18. Non-Delayed Access with $\overline{WAIT}$ 1T ($\overline{WAIT}$ is Sampled at the End of the "T2" Clock State)



TL/F/10126–19

FIGURE 19. Delayed Access with $\overline{WAIT}$ 1T ($\overline{WAIT}$ is Sampled at the End of the "T2" Clock State)

5.0 Wait State Support (Continued)

When ending $\overline{WAIT}$ from a negative level of CLK; if $\overline{RAS}$ is asserted while CLK is high then $\overline{WAIT}$ will negate from the negative edge of CLK; if $\overline{RAS}$ is asserted while CLK is low then $\overline{WAIT}$ will negate from $\overline{RAS}$ asserting. When ending $\overline{WAIT}$ from a positive edge of CLK, the user can think of the positive edge of CLK that starts $\overline{RAS}$ as 0T and the next positive edge of CLK as 1T. In a delayed access, the positive edge of CLK that starts $\overline{RAS}$ can be thought of as 0T and the next positive edge as 1T.

5.1.2 Wait during Page Burst Accesses

$\overline{WAIT}$ can be programmed to function differently during page/burst types of accesses. During a page/burst access, the $\overline{ECAS}$ inputs will be asserted then negated while $\overline{TSO}$ is asserted. Through address bits R4 and R5, $\overline{WAIT}$ can be programmed to assert and negate during this type of access. The user is given four programming options described below.

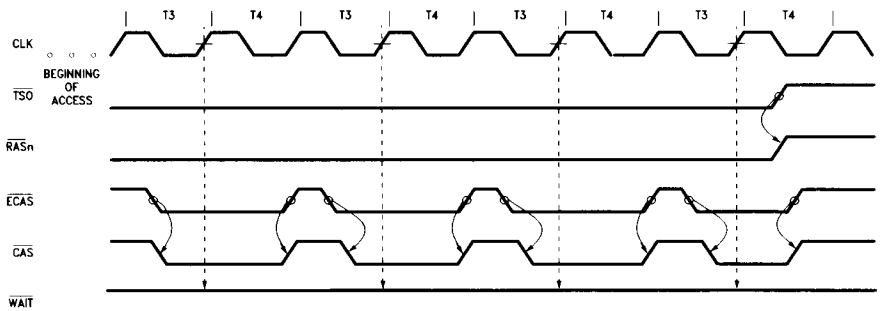
No Wait States: In this case, $\overline{WAIT}$ will remain negated even if the $\overline{ECAS}$ inputs are toggled as shown in *Figure 20*.

0T: $\overline{WAIT}$ will be asserted when the $\overline{ECAS}$ inputs are negated with $\overline{TSO}$ remaining asserted. When a single or group of $\overline{ECAS}$ inputs are asserted, $\overline{WAIT}$ will be negated as shown in *Figure 21*.

½T: $\overline{WAIT}$ will be asserted when the $\overline{ECAS}$ inputs are negated with $\overline{TSO}$ remaining asserted. When a single or group of $\overline{ECAS}$ inputs are asserted again, $\overline{WAIT}$ will be negated from the first negative level of CLK after a single $\overline{ECAS}$ or group of $\overline{ECAS}$s are asserted as shown in *Figure 22*.

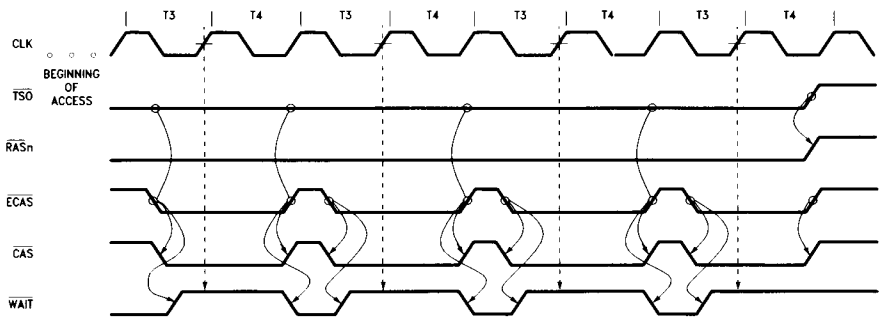
1T: $\overline{WAIT}$ will be asserted when the $\overline{ECAS}$ inputs are negated with $\overline{TSO}$ remaining asserted. When a single or group of $\overline{ECAS}$ inputs are asserted again, $\overline{WAIT}$ will be negated from the first positive edge of CLK after a single $\overline{ECAS}$ or group of $\overline{ECAS}$s are asserted as shown in *Figure 23*.

When ending $\overline{WAIT}$ from a negative level of CLK; if the $\overline{ECAS}$s are asserted while CLK is high then $\overline{WAIT}$ will negate from the negative edge of CLK, if the $\overline{ECAS}$s are asserted while CLK is low then $\overline{WAIT}$ will negate from the $\overline{ECAS}$s asserting. When ending $\overline{WAIT}$ from a positive edge of CLK, the positive edge of CLK that $\overline{ECAS}$ is setup to can be thought of as 1T.



TL/F/10126–20

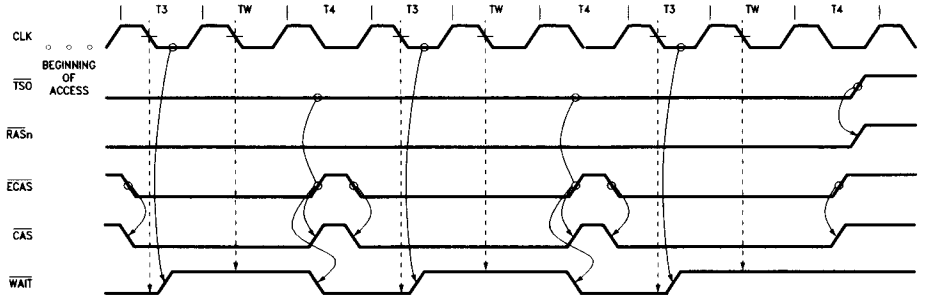
FIGURE 20. No Wait States during Burst ($\overline{WAIT}$ is Sampled at the End of the "T3" Clock State)



TL/F/10126–21

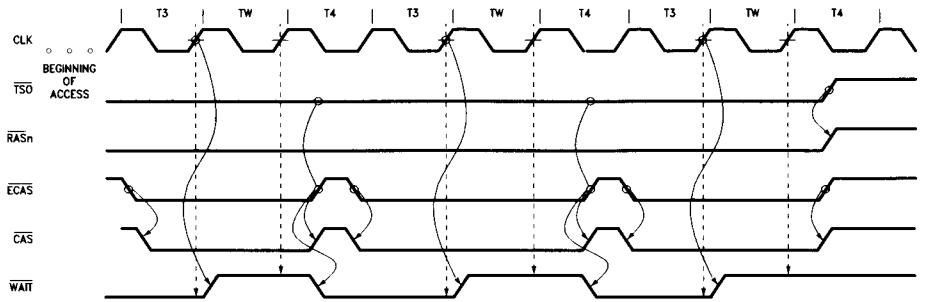
FIGURE 21. 0T during Burst ($\overline{WAIT}$ is Sampled at the End of the "T3" Clock State)

5.0 Wait State Support (Continued)



TL/F/10126–22

FIGURE 22. ½T during Burst Access (WAIT is Sampled at the "T3" Falling Clock Edge)

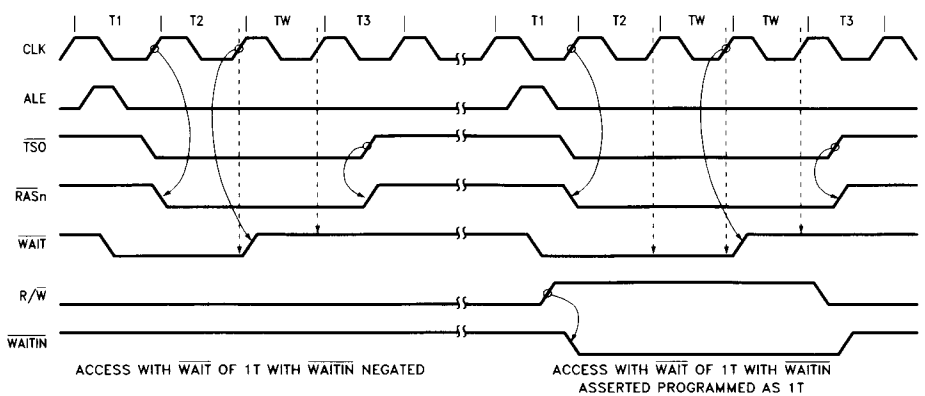


TL/F/10126–23

FIGURE 23. 1T during Burst Access (WAIT is Sampled at the End of the "T3" Clock State)

5.2 DYNAMICALLY INCREASING THE NUMBER OF WAIT STATES

The user can increase the number of positive edges of CLK before WAIT is negated. With the input WAITIN asserted, the user can delay WAIT negating either one or two more positive edges of CLK. The number of edges is programmed through address bit R6. If the user is increasing the number of positive edges in a delay that contains a negative level, the positive edges will be met before the negative level. WAITIN can increase the number of positive edges in a page/burst access. WAITIN can be permanently asserted in systems requiring an increased number of wait states. WAITIN can also be asserted and negated, depending on the type of access. As an example, a user could connect the DDIN output from the NS32CG16 to the WAITIN input. This could be used to perform write accesses with 1 wait state and read accesses with 2 wait states as shown in *Figure 24*.



TL/F/10126–24

FIGURE 24. WAITIN Example (WAIT is Sampled at the End of "T2")

5.0 Wait State Support (Continued)

5.3 GUARANTEEING $\overline{RAS}$ LOW TIME AND $\overline{RAS}$ PRECHARGE TIME

The NS32CG821A will guarantee $\overline{RAS}$ precharge time between accesses; between refreshes; and between access and refreshes. The programming bits R0 and R1 are used to program combinations of $\overline{RAS}$ precharge time and $\overline{RAS}$ low time referenced by positive edges of CLK. $\overline{RAS}$ low time is programmed for refreshes only. During an access, the system designer guarantees the time $\overline{RAS}$ is asserted through the NS32CG821A wait logic. Since inserting wait states into an access increases the length of the CPU signals which are used to create ALE and $\overline{TSO}$, the time that $\overline{RAS}$ is asserted can be guaranteed.

Precharge time is also guaranteed by the NS32CG821A. Each $\overline{RAS}$ output has a separate positive edge of CLK counter. $\overline{TSO}$ is negated setup to a positive edge of CLK to terminate the access. That positive edge is 1T. The next positive edge is 2T. $\overline{RAS}$ will not be asserted until the programmed number of positive edges of CLK have passed. Once the programmed precharge time has been met, $\overline{RAS}$ will be asserted from the positive edge of CLK. However, since there is a precharge counter per $\overline{RAS}$, an access using another $\overline{RAS}$ will not be delayed. Precharge time before a refresh is always referenced from the access $\overline{RAS}$ negating before $\overline{RAS}0$ for the refresh asserting. After a refresh, precharge time is referenced from $\overline{RAS}3$ negating, for the refresh, to the access $\overline{RAS}$ asserting.

6.0 Additional Access Support Features

To support the different modes of accessing, the NS32CG821A have multiple access features. These features allow the user to take advantage of CPU or DRAM functions. These additional features include: address latches and column increment for page/burst mode support; and delay $\overline{CAS}$, to allow the user with a multiplexed bus to ensure valid data is present before $\overline{CAS}$ is asserted.

6.1 ADDRESS LATCHES AND COLUMN INCREMENT

The address latches can be programmed, through programming bit B0, to either latch the address or remain permanently in fall-through mode. If the address latches are used to latch the address, the rising edge of ALE places the latches in fall-through. Once ALE is negated, the address present on the row, column and bank inputs is latched.

Once the address is latched, the column address can be incremented with the input COLINC. With COLINC asserted, the column address is incremented.

If COLINC is asserted with all of the bits of the column address asserted, the column address will return to zero. COLINC can be used for sequential accesses of static column DRAMs. COLINC can also be used with the $\overline{ECAS}$ inputs to support sequential accesses to page mode DRAMs as shown in *Figure 25*. COLINC should only be asserted during an access.

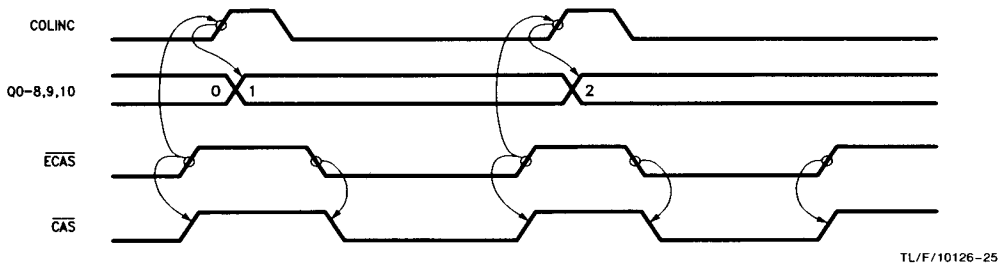


FIGURE 25. Column Increment

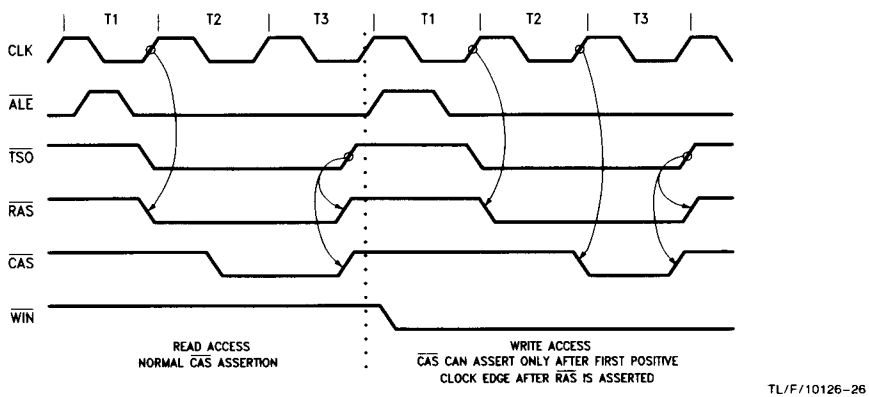


FIGURE 26. Delay $\overline{CAS}$

6.0 Additional Access Support Features (Continued)

6.2 DELAY $\overline{CAS}$ DURING WRITE ACCESSES

Address bit C9 asserted during programming will cause $\overline{CAS}$ to be delayed until the first positive edge of CLK after $\overline{RAS}$ is asserted when the input $\overline{WIN}$ is asserted. Delaying $\overline{CAS}$ during write accesses ensures that the data to be written to DRAM will be setup to $\overline{CAS}$ asserting as shown in *Figure 26*. If the possibility exists that data still may not be present after the first positive edge of CLK, $\overline{CAS}$ can be delayed further with the $\overline{ECAS}$ inputs. If address bit C9 is negated during programming, read and write accesses will be treated the same (with regard to $\overline{CAS}$).

7.0 $\overline{RAS}$ and $\overline{CAS}$ Configuration Modes

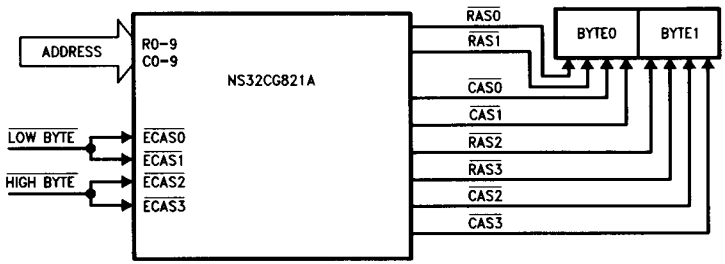
The NS32CG821A allow the user to configure the DRAM array to contain one, two, four or eight banks of DRAM. Depending on the functions used, certain considerations must be used when determining how to set up the DRAM array. Programming address bits C4, C5 and C6 along with bank selects, B0–1, and $\overline{CAS}$ enables, $\overline{ECAS}$0–3, determine which $\overline{RAS}$ or group of $\overline{RAS}$s and which $\overline{CAS}$ or group of $\overline{CAS}$s will be asserted during an access. Different memory schemes are described. The NS32CG821A is specified driving a heavy load of 72 DRAMs, representing four banks of DRAM with 16-bit words and 2 parity bits. The NS32CG821A can drive more than 72 DRAMs, but the AC timing must be increased. Since the $\overline{RAS}$ and $\overline{CAS}$ outputs are configurable, all $\overline{RAS}$ and $\overline{CAS}$ outputs should be used for the maximum amount of drive.

7.1 BYTE WRITING

By selecting a configuration in which all $\overline{CAS}$ outputs are selected during an access, the $\overline{ECAS}$ inputs enable a single or group of $\overline{CAS}$ outputs to select a byte (or bytes) in a word. In this case, the $\overline{RAS}$ outputs are used to select which of up to 4 banks is to be used as shown in *Figure 29*. In systems with a word size of 16 bits, the byte enables can be gated with a high order address bit to produce four byte enables which gives an equivalent to 8 banks of 16-bit words as shown in *Figure 30*. If less memory is required, each $\overline{CAS}$ should be used to drive each nibble in the 16-bit word as shown in *Figures 27* and *28*.

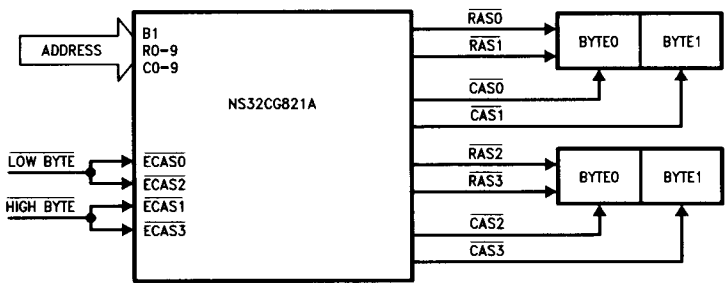
7.2 MEMORY INTERLEAVING

Memory interleaving allows the cycle time of DRAMs to be reduced by having sequential accesses to different memory banks. Since the NS32CG821A have separate precharge counters per bank, sequential accesses will not be delayed if the accessed banks use different $\overline{RAS}$ outputs. To ensure different $\overline{RAS}$ outputs will be used, a mode is selected where either one or two $\overline{RAS}$ outputs will be asserted during an access. The bank select or selects, B0 and B1, are then tied to the least significant address bits, causing a different



TL/F/10126–27

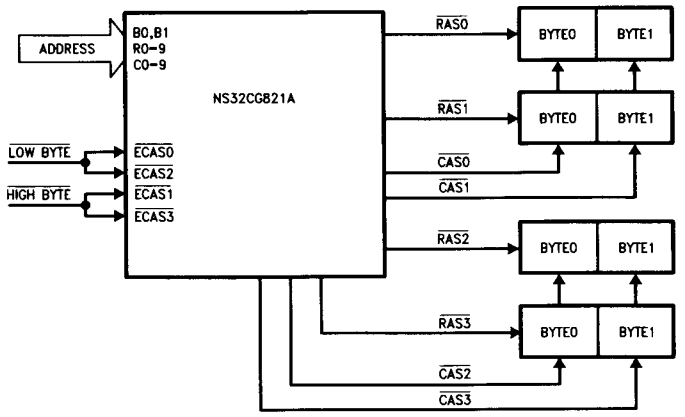
FIGURE 27. 1 Bank DRAM Array Setup for 16-Bit System (C_6, C_5, C_4 = 011 during Programming)



TL/F/10126–28

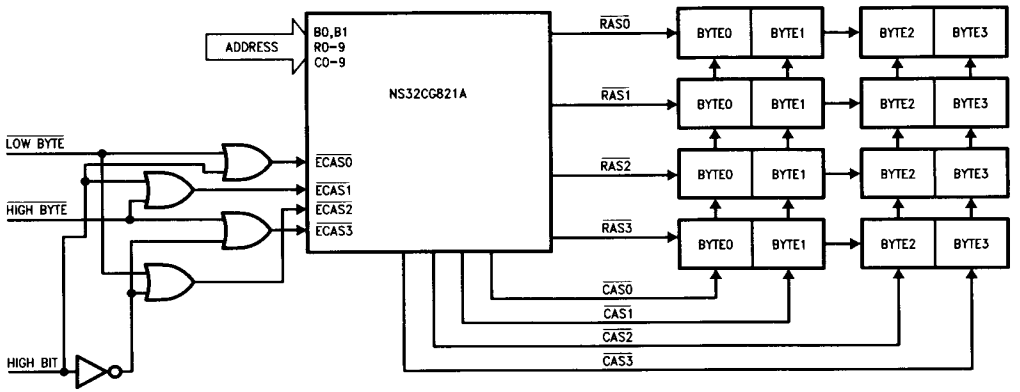
FIGURE 28. 2 Banks DRAM Array Setup for 16-Bit System (C_6, C_5, C_4 = 101 during Programming)

7.0 $\overline{RAS}$ and $\overline{CAS}$ Configuration Modes (Continued)



TL/F/10126–29

FIGURE 29. 4 Banks Array Setup for 16-Bit System (C_6, C_5, C_4 = 110 during Programming)



TL/F/10126–30

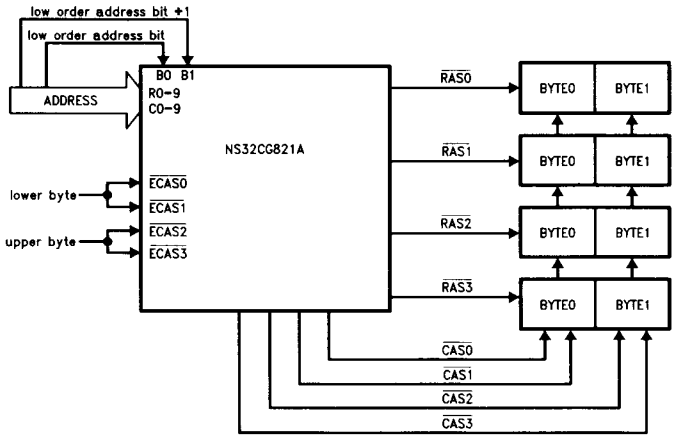
FIGURE 30. 8 Bank DRAM Array for 16-Bit System (C_6, C_5, C_4 = 1, 1, 0 during Programming)

7.0 $\overline{RAS}$ and $\overline{CAS}$ Configuration Modes (Continued)

group of $\overline{RAS}$s to assert during each sequential access as shown in *Figure 31*. In this figure there should be at least one clock period of all $\overline{RAS}$'s negated between different $\overline{RAS}$'s being asserted to avoid the condition of a $\overline{CAS}$ before $\overline{RAS}$ refresh cycle.

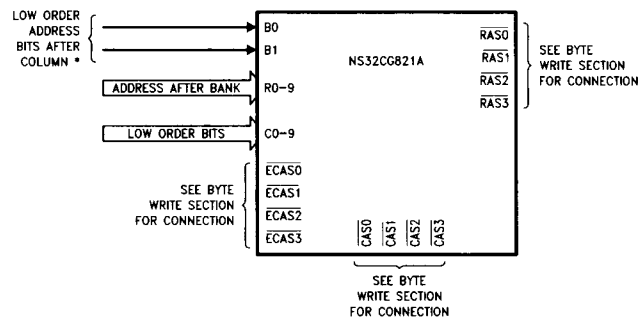
7.3 PAGE/BURST MODE

In a static column, page or burst mode system, the least significant bits must be tied to the column address in order to ensure that the page/burst accesses are to sequential memory addresses, as shown in *Figure 32*. The $\overline{ECAS}$ inputs may then be toggled with the NS32CG821A's address latches in fall-through mode, while $\overline{TSO}$ is asserted. The $\overline{ECAS}$ inputs can also be used to select individual bytes. In page or static column modes, the two address bits after the page size can be tied to the bank select inputs to select a new bank if the page size is exceeded.



TL/F/10126-31

FIGURE 31. Memory Interleaving (C6, C5, C4 = 1, 1, 0 during Programming)



TL/F/10126-32

*See table below for row, column & bank address bit map. A0 is used for byte addressing in this example.

Addresses	Page Mode/Static Column Mode Page Size		
	256 Bits/Page	512 Bits/Page	1024 Bits/Page
Column Address	C0-7 = A1-8 C8-9 = X	C0-8 = A1-9 C9 = X	C0-9 = A1-10
Row Address	X	X	X
B0 B1	A9 A10	A10 A11	A11 A12

X = DON'T CARE, the user can do as he pleases.

FIGURE 32. Page, Static Column, Mode System

8.0 Programming and Resetting

The NS32CG821A must be programmed by one of two possible programming sequences before it can be used. Once the chip is programmed, the bits effecting the wait logic become effective immediately, thus allowing the programming bus cycle to end. At power up, the NS32CG821A must be externally reset before programming. Initially, the programming bits are in an undefined state. After programming, the NS32CG821A enters a 60 ms initialization period. During this initialization period, the NS32CG821A performs refreshes about every 15 μs; this makes further DRAM warmup cycles unnecessary. The chip can be programmed as many times as the user wishes. After the first programming, the 60 ms initialization period will not be entered into unless the chip is reset. During the 60 ms initialization period, internal refreshes are taking place and the CPU is not allowed to enter into a memory access cycle. If a memory access is attempted, the NS32CG821A will send out wait states into the processor until initialization is complete. The actual initialization time period is given by the following formula:

T = 4096*(Clock Divisor Select)
*(Refresh Clock Fine Tune)
/(DELCK Frequency)

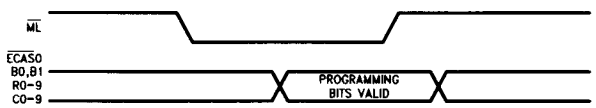
8.1 MODE LOAD ONLY PROGRAMMING

MODE LOAD, $\overline{ML}$, asserted enables an internal 23-bit programmable register. To use this method, the user asserts $\overline{ML}$, enabling the internal programming register. After $\overline{ML}$ is asserted, a valid programming selection is placed on the address bus (and $\overline{ECAS0}$), then $\overline{ML}$ is negated. When $\overline{ML}$ is negated, the value on the address bus (and $\overline{ECAS0}$) is latched into the internal programming register and the NS32CG821A is programmed, as shown in *Figure 33*. After $\overline{ML}$ is negated, the NS32CG821A will enter the 60 ms initialization period only if this is the first programming after power up or reset.

Using this method, a set of transceivers on the address bus can be put at TRI-STATE® by the system reset signal. A combination of pull-up and pull-down resistors can be used on the address inputs of the NS32CG821A to select the programming values, as shown in *Figure 34*.

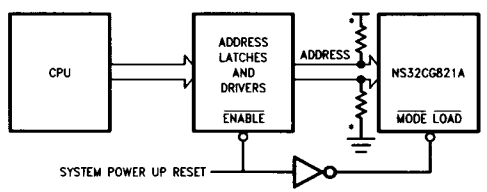
8.2 CHIP SELECTED ACCESS PROGRAMMING

The chip can also be programmed by asserting $\overline{ML}$ and performing a chip selected access. ALE is disabled internally until after programming. To program the chip using this method, $\overline{ML}$ is asserted. After $\overline{ML}$ is asserted, $\overline{CS}$ is asserted and a valid programming selection is placed on the address bus. When $\overline{TSO}$ is asserted, the chip is programmed with the programming selection on the address bus. After $\overline{TSO}$ is negated, $\overline{ML}$ can be negated as shown in *Figure 35*.



TL/F/10126-33

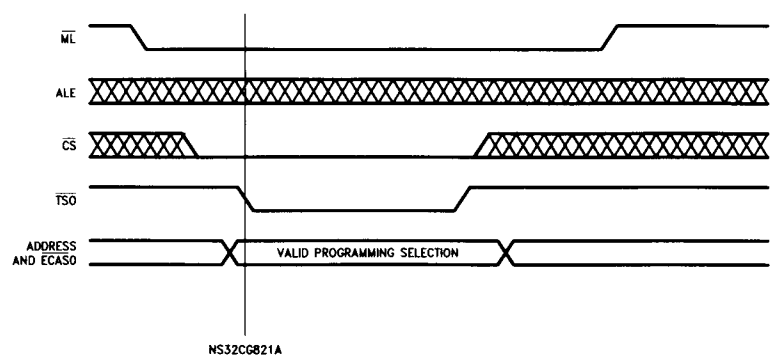
FIGURE 33. Mode Load Only Programming



TL/F/10126-34

*Pull-Up or Pull-Down Resistors on Each Address Input

FIGURE 34. Programming during System Reset

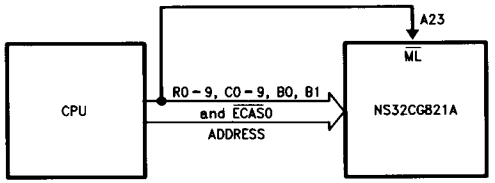


TL/F/10126-35

FIGURE 35. $\overline{CS}$ Access Programming

8.0 Programming and Resetting (Continued)

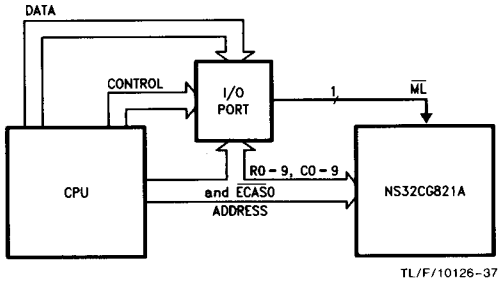
Using this method, various programming schemes can be used. For example if extra upper address bits are available, an unused high order address bit can be tied to the signal $\overline{ML}$. Using this method, one need only write to a page of memory, thus asserting the high order bit and in turn programming the chip as shown in *Figure 36*.



TL/F/10126–36

FIGURE 36. Programming the NS32CG821A through the Address Bus Only

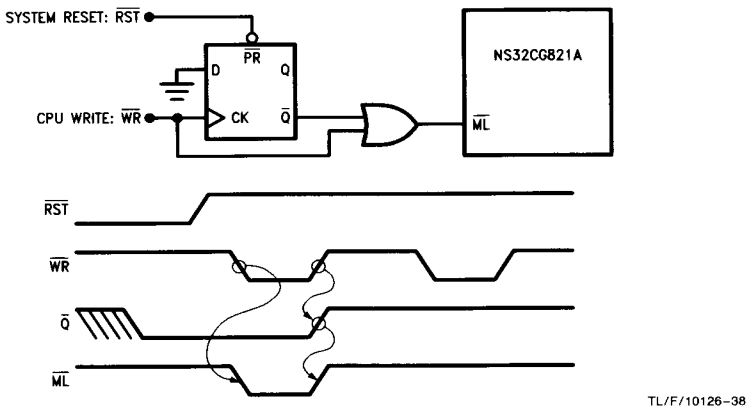
An I/O port can also be used to assert $\overline{ML}$. After $\overline{ML}$ is asserted, a chip selected access can be performed to program the chip. After the chip selected access, $\overline{ML}$ can be negated through the I/O port as shown in *Figure 37*.



TL/F/10126–37

FIGURE 37. Programming the NS32CG821A through the Address Bus and an I/O Port

Another simple way the chip can be programmed is the first write after system reset. This method requires only a flip-flop and an OR gate as shown in *Figure 38*. At reset, the flip-flop is preset, which pulls the $\overline{Q}$ output low. Since $\overline{WR}$ is negated, $\overline{ML}$ is not enabled. The first write access is used to program the chip. When $\overline{WR}$ is asserted, $\overline{ML}$ is asserted. $\overline{WR}$ negated clocks the flip-flop, negates $\overline{ML}$, and programs the NS32CG821A with the address and $\overline{ECAS0}$ available at that time. $\overline{CS}$ does not need to be asserted using this method.



TL/F/10126–38

FIGURE 38. Programming the NS32CG821A on the First CPU Write after Power Up

8.0 Programming and Resetting (Continued)

8.4 PROGRAMMING BIT DEFINITIONS (Notes 1 and 2)

Symbol	Description
B0	**Address Latch Mode**
0	ALE asserted latches the input row, column and bank address.
1	The row, column and bank latches are fall through.
C9	**Delay $\overline{CAS}$ during WRITE Accesses**
0	$\overline{CAS}$ is treated the same for both READ and WRITE accesses.
1	During WRITE accesses, $\overline{CAS}$ will be asserted by the event that occurs last: $\overline{CAS}$ asserted by the internal delay line or $\overline{CAS}$ asserted on the positive edge of CLK after $\overline{RAS}$ is asserted.
C8	**Row Address Hold Time**
0	Row Address Hold Time = 25 ns minimum
1	Row Address Hold Time = 15 ns minimum
C7	**Column Address Setup Time**
0	Column Address Setup Time = 10 ns minimum
1	Column Address Setup Time = 0 ns minimum
C6, C5, C4	**$\overline{RAS}$ and $\overline{CAS}$ Configuration Modes**
0, 1, 1	$\overline{RAS}$0–3 and $\overline{CAS}$0–3 are all selected during an access. $\overline{ECAS}$n must be asserted for $\overline{CAS}$n to be asserted. B1, B0 are not used during an access.
1, 0, 1	$\overline{RAS}$ and $\overline{CAS}$ pairs are selected by B1. $\overline{ECAS}$n must be asserted for $\overline{CAS}$n to be asserted. B1 = 0 during an access selects $\overline{RAS}$0–1 and $\overline{CAS}$0–1. B1 = 1 during an access selects $\overline{RAS}$2–3 and $\overline{CAS}$2–3. B0 is not used during an access.
1, 1, 0	$\overline{RAS}$ singles are selected by B0–1. $\overline{CAS}$0–3 are all selected. $\overline{ECAS}$n must be asserted for $\overline{CAS}$n to be asserted. B1 = 0, B0 = 0 during an access selects $\overline{RAS}$0 and $\overline{CAS}$0–3. B1 = 0, B0 = 1 during an access selects $\overline{RAS}$1 and $\overline{CAS}$0–3. B1 = 1, B0 = 0 during an access selects $\overline{RAS}$2 and $\overline{CAS}$0–3. B1 = 1, B0 = 1 during an access selects $\overline{RAS}$3 and $\overline{CAS}$0–3.
C3	**Refresh Clock Fine Tune Divisor**
0	Divide delay line/refresh clock further by 30 (If DELCLK/Refresh Clock Clock Divisor = 2 MHz = 15 μs refresh period).
1	Divide delay line/refresh clock further by 26 (If DELCLK/Refresh Clock Clock Divisor = 2 MHz = 13 μs refresh period).
C2, C1, C0	**Delay Line/Refresh Clock Divisor Select**
0, 0, 0	Divide DELCLK by 10 to get as close to 2 MHz as possible.
0, 0, 1	Divide DELCLK by 9 to get as close to 2 MHz as possible.
0, 1, 0	Divide DELCLK by 8 to get as close to 2 MHz as possible.
0, 1, 1	Divide DELCLK by 7 to get as close to 2 MHz as possible.
1, 0, 0	Divide DELCLK by 6 to get as close to 2 MHz as possible.
1, 0, 1	Divide DELCLK by 5 to get as close to 2 MHz as possible.
1, 1, 0	Divide DELCLK by 4 to get as close to 2 MHz as possible.
1, 1, 1	Divide DELCLK by 3 to get as close to 2 MHz as possible.
R9	**Refresh Mode Select**
0	$\overline{RAS}$0–3 will all assert and negate at the same time during a refresh.
1	Staggered Refresh. $\overline{RAS}$ outputs during refresh are separated by one positive clock edge. Depending on the configuration mode.
R6	**Add Wait States to the Current Access if $\overline{WAITIN}$ is Low.**
0	$\overline{WAIT}$ will be delayed by one additional positive edge of CLK.
1	$\overline{WAIT}$ will be delayed by two additional positive edges of CLK.

8.0 Programming and Resetting (Continued)

8.4 PROGRAMMING BIT DEFINITIONS (Continued)

Symbol	Description
R5, R4	**$\overline{\text{WAIT}}$ during Burst (See Section 5.1.2)**
0, 0	NO WAIT STATES; $\overline{\text{WAIT}}$ will remain negated during burst portion of access.
0, 1	1T; $\overline{\text{WAIT}}$ will assert when the $\overline{\text{ECAS}}$ inputs are negated with $\overline{\text{TSO}}$ asserted. $\overline{\text{WAIT}}$ will negate from the positive edge of CLK after the $\overline{\text{ECAS}}$s have been asserted.
1, 0	½T; $\overline{\text{WAIT}}$ will assert when the $\overline{\text{ECAS}}$ inputs are negated with $\overline{\text{TSO}}$ asserted. $\overline{\text{WAIT}}$ will negate on the negative level of CLK after the $\overline{\text{ECAS}}$s have been asserted.
1, 1	0T; $\overline{\text{WAIT}}$ will assert when the $\overline{\text{ECAS}}$ inputs are negated. $\overline{\text{WAIT}}$ will negate when the $\overline{\text{ECAS}}$ inputs are asserted.
R3, R2	**$\overline{\text{WAIT}}$ Delay Times (See Section 5.1.1)**
0, 0	NO WAIT STATES; $\overline{\text{WAIT}}$ will remain high during non-delayed accesses. $\overline{\text{WAIT}}$ will negate when $\overline{\text{RAS}}$ is negated during delayed accesses.
0, 1	½T; $\overline{\text{WAIT}}$ will negate on the negative level of CLK, after the access $\overline{\text{RAS}}$.
1, 0	NO WAIT STATES, ½ T; $\overline{\text{WAIT}}$ will remain high during non-delayed accesses. $\overline{\text{WAIT}}$ will negate on the negative level of CLK, after the access $\overline{\text{RAS}}$, during delayed accesses.
1, 1	1T; $\overline{\text{WAIT}}$ will negate on the positive edge of CLK after the access $\overline{\text{RAS}}$.
R1, R0	**$\overline{\text{RAS}}$ Low and $\overline{\text{RAS}}$ Precharge Time**
0, 0	$\overline{\text{RAS}}$ asserted during refresh = 2 positive edges of CLK. $\overline{\text{RAS}}$ precharge time = 1 positive edge of CLK.
0, 1	$\overline{\text{RAS}}$ asserted during refresh = 3 positive edges of CLK. $\overline{\text{RAS}}$ precharge time = 2 positive edges of CLK.
1, 0	$\overline{\text{RAS}}$ asserted during refresh = 2 positive edges of CLK. $\overline{\text{RAS}}$ precharge time = 2 positive edges of CLK.
1, 1	$\overline{\text{RAS}}$ asserted during refresh = 4 positive edges of CLK. $\overline{\text{RAS}}$ precharge time = 3 positive edges of CLK.

Note 1: During programming $\overline{\text{ECAS}}_0$, B_1, R_7 have to be set to low, and R_8 has to be set high.

Note 2: $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ configuration modes C_6, C_5, C_4 = 000, 001, 010, 100 and 111 are reserved.

9.0 DRAM Critical Timing Parameters

The two critical timing parameters, shown in *Figure 39*, that must be met when controlling the access timing to a DRAM are the row address hold time, tRAH, and the column address setup time, tASC. Since the NS32CG821A contains a precise internal delay line, the values of these parameters can be selected at programming time. These values will also increase and decrease if DELCLK varies from 2 MHz.

9.1 PROGRAMMABLE VALUES OF tRAH AND tASC

The NS32CG821A allow the values of tRAH and tASC to be selected at programming time. For each parameter, two choices can be selected. tRAH, the row address hold time, is measured from $\overline{RAS}$ asserted to the row address starting to change to the column address. The two choices for tRAH are 15 ns and 25 ns, programmable through address bit C8.

tASC, the column address setup time, is measured from the column address valid to $\overline{CAS}$ asserted. The two choices for tASC are 0 ns and 10 ns, programmable through address bit C7.

9.2 CALCULATION OF tRAH AND tASC

There are two clock inputs to the NS32CG821A. These two clocks, DELCLK and CLK can either be tied together to the same clock or be tied to different clocks running asynchronously at different frequencies.

The clock input, DELCLK, controls the internal delay line and refresh request clock. DELCLK should be a multiple of 2 MHz. If DELCLK is not a multiple of 2 MHz, tRAH and tASC will change. The new values of tRAH and tASC can be calculated by the following formulas:

If tRAH was programmed to equal 15 ns then tRAH = 30*(((DELCLK Divisor)* 2 MHz/(DELCLK Frequency)) − 1) + 15 ns.

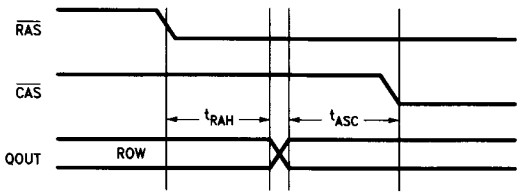
If tRAH was programmed to equal 25 ns then tRAH = 30*(((DELCLK Divisor)* 2 MHz/(DELCLK Frequency)) − 1) + 25 ns.

If tASC was programmed to equal 0 ns then tASC = 15* ((DELCLK Divisor)* 2 MHz/(DELCLK Frequency)) − 15 ns.

If tASC was programmed to equal 10 ns then tASC = 25* ((DELCLK Divisor)* 2 MHz/(DELCLK Frequency)) − 15 ns.

Since the values of tRAH and tASC are increased or decreased, the time to $\overline{CAS}$ asserted will also increase or decrease. These parameters can be adjusted by the following formula:

Delay to $\overline{CAS}$ = Actual Spec. + Actual tRAH − Programmed tRAH + Actual tASC − Programmed tASC.



TL/F/10126–41

FIGURE 39. tRAH and tASC

10.0 Absolute Maximum Ratings (Note 1)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Temperature under Bias ..... 0°C to +70°C
Storage Temperature ..... −65°C to +150°C
All Input or Output Voltage with Respect to GND ..... −0.5V to +7V
Power Dissipation @ 20 MHz 0.5W
ESD Rating to be determined.

11.0 DC Electrical Characteristics T_A = 0°C to +70°C, V_{CC} = 5V ±10%, GND = 0V

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_{IH}	Logical 1 Input Voltage	Tested with a Limited Functional Pattern	2.0		V_{CC} + 0.5	V
V_{IL}	Logical 0 Input Voltage	Tested with a Limited Functional Pattern	−0.5		0.8	V
V_{OH1}	Q and $\overline{WE}$ Outputs	I_{OH} = −10 mA	V_{CC} − 1.0			V
V_{OL1}	Q and $\overline{WE}$ Outputs	I_{OL} = 10 mA			0.5	V
V_{OH2}	All Outputs except Qs, $\overline{WE}$	I_{OH} = −3 mA	V_{CC} − 1.0			V
V_{OL2}	All Outputs except Qs, $\overline{WE}$	I_{OL} = 3 mA			0.5	V
I_{IN}	Input Leakage Current	V_{IN} = V_{CC} or GND	−10		10	μA
$I_{IL\ ML}$	$\overline{ML}$ Input Current (Low)	V_{IN} = GND			200	μA
I_{CC1}	Standby Current	CLK at 8 MHz (V_{IN} = V_{CC} or GND)		6	15	mA
I_{CC1}	Standby Current	CLK at 20 MHz (V_{IN} = V_{CC} or GND)		8	17	mA
I_{CC1}	Standby Current	CLK at 25 MHz (V_{IN} = V_{CC} or GND)		10	20	mA
I_{CC2}	Supply Current	CLK at 8 MHz (Inputs Active) (I_{LOAD} = 0) (V_{IN} = V_{CC} or GND)		20	40	mA
I_{CC2}	Supply Current	CLK at 20 MHz (Inputs Active) (I_{LOAD} = 0) (V_{IN} = V_{CC} or GND)		40	75	mA
I_{CC2}	Supply Current	CLK at 25 MHz (Inputs Active) (I_{LOAD} = 0) (V_{IN} = V_{CC} or GND)		50	95	mA
C_{IN}*	Input Capacitance	f_{IN} at 1 MHz			10	pF

*Note: C_{IN} is not 100% tested.

12.0 AC Timing Parameters: NS32CG821A

Two speed selections are given, the NS32CG821A-20 and the NS32CG821A-25. The differences between the two parts are the maximum operating frequencies of the input CLKs and the maximum delay specifications. Low frequency applications may use the "−25" part to gain improved timing.

The AC timing parameters are grouped into sectional numbers as shown below. These numbers also refer to the timing diagrams.

1–36 Common parameters to all modes of operation

50–56 Difference parameters used to calculate;
$\overline{RAS}$ low time,
$\overline{RAS}$ precharge time,
$\overline{CAS}$ high time and
$\overline{CAS}$ low time

200–212 Refresh parameters

300–315 Memory access parameters used in both single and dual access applications

500–506 Programming parameters

Unless otherwise stated V_{CC} = 5.0V ±10%, 0 < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

Two different loads are specified:

C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or

C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

12.0 AC Timing Parameters: NS32CG821A (Continued)

Unless otherwise stated V_{CC} = 5.0V ±10%, 0°C < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

Two different loads are specified:

C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or

C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

Number	Symbol	Common Parameter Description	NS32CG821A-20				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
1	fCLK	CLK Frequency	0	20	0	20	0	25	0	25
2	tCLKP	CLK Period	50		50		40		40	
3, 4	tCLKPW	CLK Pulse Width	15		15		12		12	
5	fDCLK	DELCLK Frequency	5	20	5	20	5	20	5	20
6	tDCLKP	DELCLK Period	50	200	50	200	50	200	50	200
7, 8	tDCLKPW	DELCLK Pulse Width	15		15		12		12	
9a	tPRASCAS0	$\overline{RAS}$ Asserted to $\overline{CAS}$ Asserted (tRAH = 15 ns, tASC = 0 ns)	30		30		30		30	
9b	tPRASCAS1	$\overline{RAS}$ Asserted to $\overline{CAS}$ Asserted (tRAH = 15 ns, tASC = 10 ns)	40		40		40		40	
9c	tPRASCAS2	($\overline{RAS}$ Asserted to $\overline{CAS}$ Asserted (tRAH = 25 ns, tASC = 0 ns)	40		40		40		40	
9d	tPRASCAS3	($\overline{RAS}$ Asserted to $\overline{CAS}$ Asserted (tRAH = 25 ns, tASC = 10 ns)	50		50		50		50	
10a	tRAH	Row Address Hold Time (tRAH = 15)	15		15		15		15	
10b	tRAH	Row Address Hold Time (tRAH = 25)	25		25		25		25	
11a	tASC	Column Address Setup Time (tASC = 0)	0		0		0		0	
11b	tASC	Column Address Setup Time (tASC = 10)	10		10		10		10	
12	tPCKRAS	CLK High to $\overline{RAS}$ Asserted following Precharge		27		32		22		26
13	tPARQRAS	$\overline{TSO}$ Negated to $\overline{RAS}$ Negated		38		43		31		35
14	tPENCL	$\overline{ECAS}$0–3 Asserted to $\overline{CAS}$ Asserted		23		31		20		27
15	tPENCH	$\overline{ECAS}$0–3 Negated to $\overline{CAS}$ Negated		25		33		20		27
16	tPARQCAS	$\overline{TSO}$ Negated to $\overline{CAS}$ Negated		60		68		47		54
17	tPCLKWH	CLK to $\overline{WAIT}$ Negated		39		39		24		31
19	tPEWL	$\overline{ECAS}$ Negated to $\overline{WAIT}$ Asserted during a Burst Access		44		44		36		36
20	tSECK	$\overline{ECAS}$ Asserted Setup to CLK High to Recognize the Rising Edge of CLK during a Burst Access	24		24		19		19	
23	tSWCK	$\overline{WAITIN}$ Asserted Setup to CLK	5		5		5		5	
24	tPWINWEH	$\overline{WIN}$ Asserted to $\overline{WE}$ Asserted		34		44		27		37
25	tPWINWEL	$\overline{WIN}$ Negated to $\overline{WE}$ Negated		34		44		27		37
26	tPAQ	Row, Column Address Valid to Q0–9 Valid		29		38		26		35
27	tPCINCQ	COLINC Asserted to Q0–9 Incremented		34		43		30		39
28	tSCINEN	COLINC Asserted Setup to $\overline{ECAS}$ Asserted to Ensure tASC = 0 ns	18		19		17		19	

12.0 AC Timing Parameters: NS32CG821A (Continued)

Unless otherwise stated V_{CC} = 5.0V ±10%, 0°C < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

Two different loads are specified:
C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or
C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

Number	Symbol	Common Parameter Description	NS32CG821A-20				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
29a	tSARQCK1	$\overline{TSO}$ Negated Setup to CLK High with 1 Period of Precharge	46		46		37		37	
29b	tSARQCK2	$\overline{TSO}$ Negated Setup to CLK High with >1 Period of Precharge Programmed	19		19		15		15	
31	tPCKCAS	CLK High to $\overline{CAS}$ Asserted when Delayed by $\overline{WIN}$		31		39		25		32
32	tSCADEN	Column Address Setup to $\overline{ECAS}$ Asserted to Guarantee tASC = 0	14		15		14		16	
33	tWCINC	COLINC Pulse Width	20		20		20		20	
34a	tPCKCL0	CLK High to $\overline{CAS}$ Asserted following Precharge (tRAH = 15 ns, tASC = 0 ns)		81		89		72		79
34b	tPCKCL1	CLK High to $\overline{CAS}$ Asserted following Precharge (tRAH = 15 ns, tASC = 10 ns)		91		99		82		89
34c	tPCKCL2	CLK High to $\overline{CAS}$ Asserted following Precharge (tRAH = 25 ns, tASC = 0 ns)		91		99		82		89
34d	tPCKCL3	CLK High to $\overline{CAS}$ Asserted following Precharge (tRAH = 25 ns, tASC = 10 ns)		101		109		92		99
35	tCAH	Column Address Hold Time (Interleave Mode Only)	32		32		32		32	
36	tPCQR	$\overline{CAS}$ Asserted to Row Address Valid (Interleave Mode Only)		90		90		90		90

12.0 AC Timing Parameters: NS32CG821A (Continued)

Unless otherwise stated V_{CC} = 5.0V ±10%, 0°C < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

Two different loads are specified:
C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or
C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

Number	Symbol	Difference Parameter Description	NS32CG821A-20				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
50	tD1	($\overline{TSO}$ Negated to $\overline{RAS}$ Negated) Minus (CLK High to $\overline{RAS}$ Asserted)		16		16		14		14
51	tD2	(CLK High to Refresh $\overline{RAS}$ Negated) Minus (CLK High to $\overline{RAS}$ Asserted)		13		13		11		11
53	tD3b	(CLK High to $\overline{RAS}$ Asserted Minus ($\overline{TSO}$ Negated to $\overline{RAS}$ Negated)		4		4		4		4
54	tD4	($\overline{ECAS}$ Asserted to $\overline{CAS}$ Asserted) Minus ($\overline{ECAS}$ Negated to $\overline{CAS}$ Negated)	−7	7	−7	7	−7	7	−7	7
55	tD5	(CLK to Refresh $\overline{RAS}$ Asserted) Minus (CLK to Refresh $\overline{RAS}$ Negated)		6		6		6		6

Unless otherwise stated V_{CC} = 5.0V ±10%, 0°C < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

Two different loads are specified:
C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or
C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

Number	Symbol	Refresh Parameter Description	NS32CG821A-20				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
207	tPCKRFRASH	CLK High to Refresh $\overline{RAS}$ Negated		35		40		29		33
208	tPCKRFRASL	CLK High to Refresh $\overline{RAS}$ Asserted		28		33		23		27

12.0 AC Timing Parameters: NS32CG821A (Continued)

Unless otherwise stated V_{CC} = 5.0V ±10%, 0°C < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

Two different loads are specified:
C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or
C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

Number	Symbol	Memory Access Parameter Description	NS32CG821A-20				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
300	tSCSCK	$\overline{CS}$ Asserted to CLK High	14		14		13		13	
301a	tSALECKNL	ALE Asserted Setup to CLK High Not Using On-Chip Latches or if Using On-Chip Latches and B0, B1, Are Constant, Only 1 Bank	16		16		15		15	
301b	tSALECKL	ALE Asserted Setup to CLK High, if Using On-Chip Latches if B0, B1 Can Change, More Than One Bank	29		29		29		29	
302	tWALE	ALE Pulse Width	18		18		13		13	
303	tSBADDCK	Bank Address Valid Setup to CLK High	20		20		18		18	
304	tSADDCK	Row, Column Valid Setup to CLK High to Guarantee tASR = 0 ns	11		15		11		16	
305	tHASRCB	Row, Column, Bank Address Held from ALE Negated (Using On-Chip Latches)	10		10		8		8	
306	tSRCBAS	Row, Column, Bank Address Setup to ALE Negated (Using On-Chip Latches)	3		3		2		2	
307	tPCKRL	CLK High to $\overline{RAS}$ Asserted		27		32		22		26
308a	tPCKCL0	CLK High to $\overline{CAS}$ Asserted (tRAH = 15 ns, tASC = 0 ns)		81		89		72		79
308b	tPCKCL1	CLK High to $\overline{CAS}$ Asserted (tRAH = 15 ns, tASC = 10 ns)		91		99		82		89
308c	tPCKCL2	CLK High to $\overline{CAS}$ Asserted (tRAH = 25 ns, tASC = 0 ns)		91		99		82		89
308d	tPCKCL3	CLK High to $\overline{CAS}$ Asserted (tRAH = 25 ns, tASC = 10 ns)		101		109		92		99
309	tHCKALE	ALE Negated Hold from CLK High	0		0		0		0	
310	tSWINCK	$\overline{WIN}$ Asserted Setup to CLK High that starts access $\overline{RAS}$ to Guarantee $\overline{CAS}$ is Delayed	−21		−21		−16		−16	
311	tPCSWL	$\overline{CS}$ Asserted to $\overline{WAIT}$ Asserted		26		26		22		22
312	tPCSWH	$\overline{CS}$ Negated to $\overline{WAIT}$ Negated		30		30		25		25
314	tPALEWL	ALE Asserted to $\overline{WAIT}$ Asserted ($\overline{CS}$ is Already Asserted)		35		35		29		29
315		$\overline{TSO}$ Negated to CLK High That Starts Access $\overline{RAS}$ to Guarantee tASR = 0 ns	41		45		34		39	
316	$t_{tPCKCV0}$	CLK to Column Addr. Valid (t_{RAH} = 15 ns, t_{ASC} = 0 ns)		78		87		60		75

12.0 AC Timing Parameters: NS32CG821A (Continued)

Unless otherwise stated V_{CC} = 5.0V ±10%, 0°C < T_A < 70°C, the output load capacitance is typical for 4 banks of 18 DRAMs per bank, including trace capacitance (see Note 2).

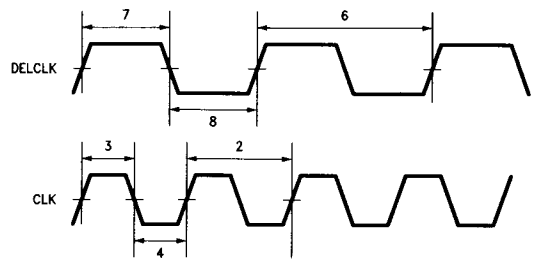
Two different loads are specified:
C_L = 50 pF loads on all outputs except
C_L = 150 pF loads on Q0–9 and $\overline{WE}$; or
C_H = 50 pF loads on all outputs except
C_H = 125 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3 and
C_H = 380 pF loads on Q0–9 and $\overline{WE}$.

Number	Symbol	Programming Parameter Description	NS32CG821A-20				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
500	tHMLADD	Mode Address Held from $\overline{ML}$ Negated	8		8		7		7	
501	tSADDML	Mode Address Setup to $\overline{ML}$ Negated	6		6		6		6	
502	tWML	$\overline{ML}$ Pulse Width	15		15		15		15	
503	tSADAQML	Mode Address Setup to $\overline{TSO}$ Asserted	0		0		0		0	
504	tHADAQML	Mode Address Held from $\overline{TSO}$ Asserted	51		51		38		38	
505	tSCSARQ	$\overline{CS}$ Asserted Setup to $\overline{TSO}$ Asserted	6		6		6		6	
506	tSMLARQ	$\overline{ML}$ Asserted Setup to $\overline{TSO}$ Asserted	10		10		10		10	

Note 1: "Absolute Maximum Ratings" are the values beyond which the safety of the device cannot be guaranteed. They are not meant to imply that the device should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

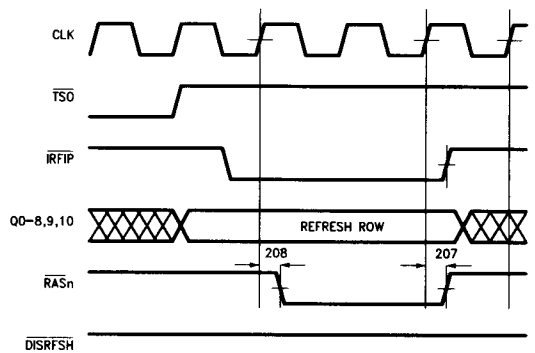
Note 2: Input pulse 0V to 3V; tR = tF = 2.5 ns. Input reference point on AC measurements is 1.5V. Output reference points are 2.4V for High and 0.8V for Low.

Note 3: AC Production testing is done at 50 pF.



TL/F/10126–42

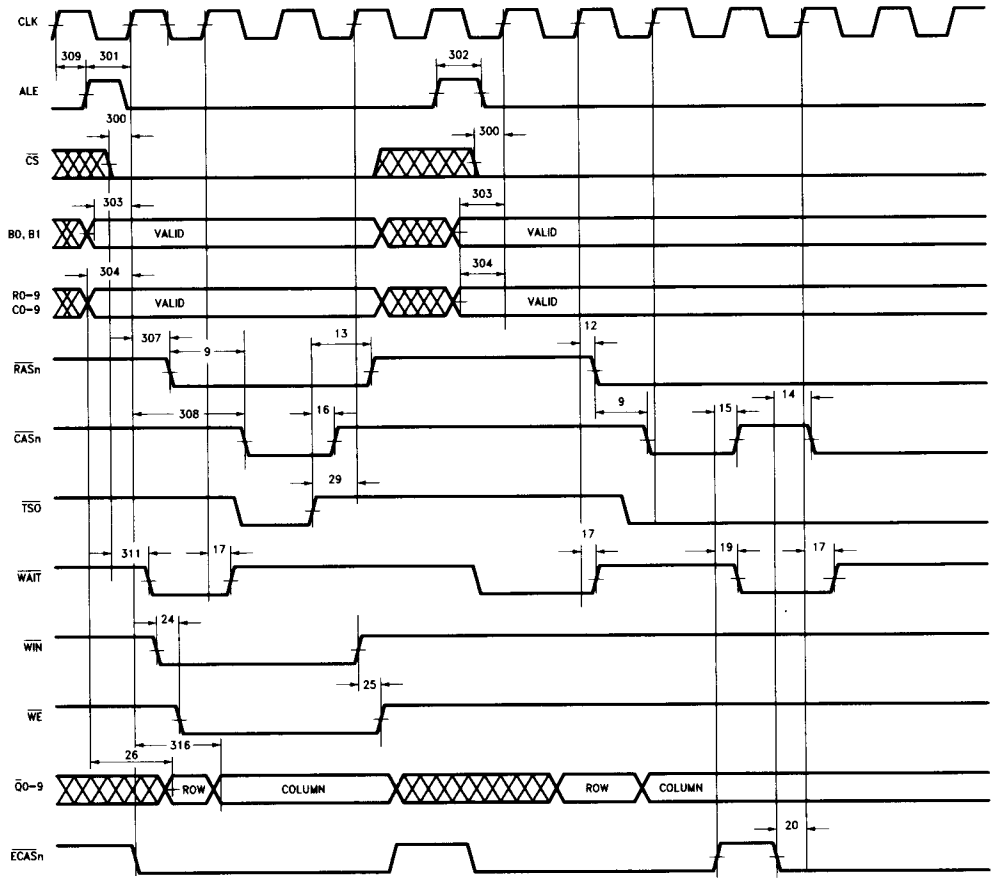
FIGURE 40. Clock, DELCLK Timing



TL/F/10126–43

FIGURE 41. 200: Refresh Timing

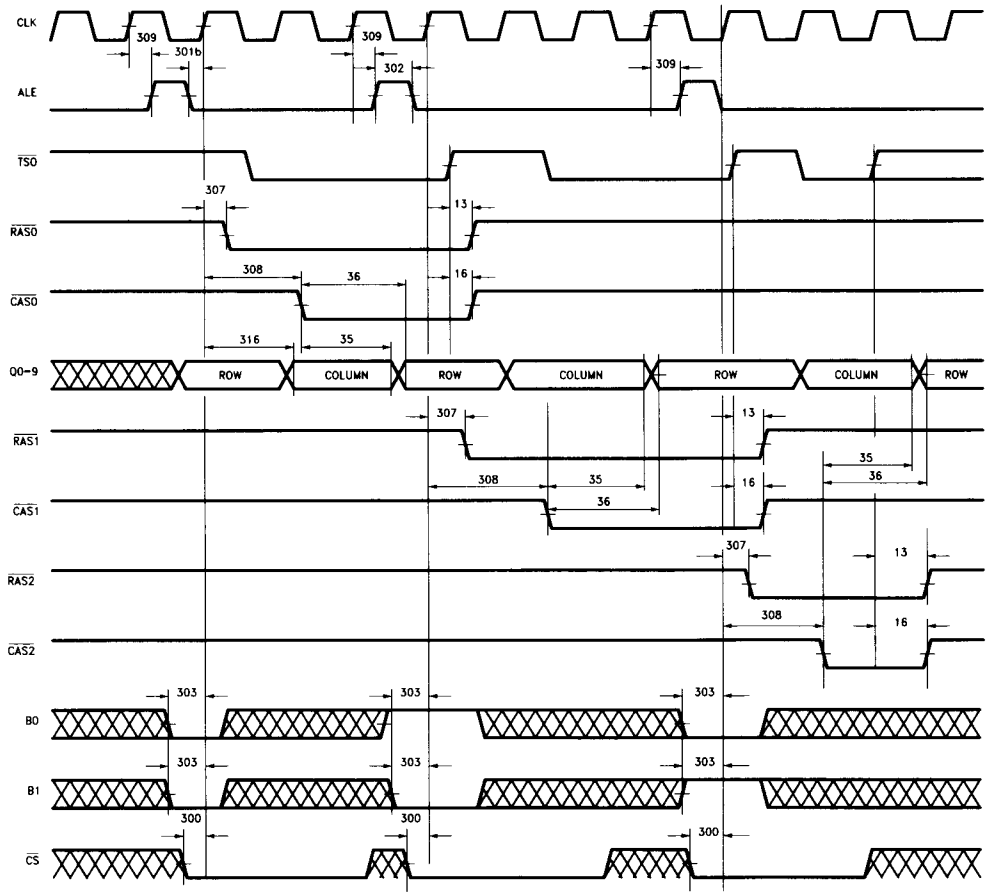
12.0 AC Timing Parameters: NS32CG821A (Continued)



TL/F/10126-44

FIGURE 42. 300: Memory Access Timing

12.0 AC Timing Parameters: NS32CG821A (Continued)

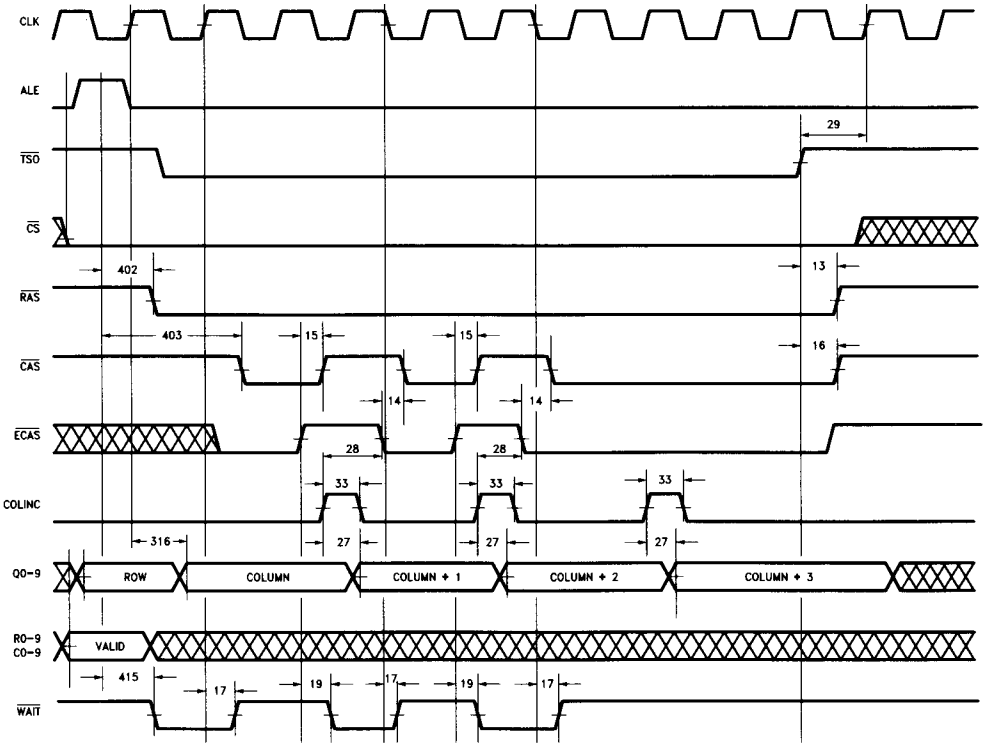


TL/F/10126–45

(Programmed as C4 = 1, C5 = 1, C6 = 1)

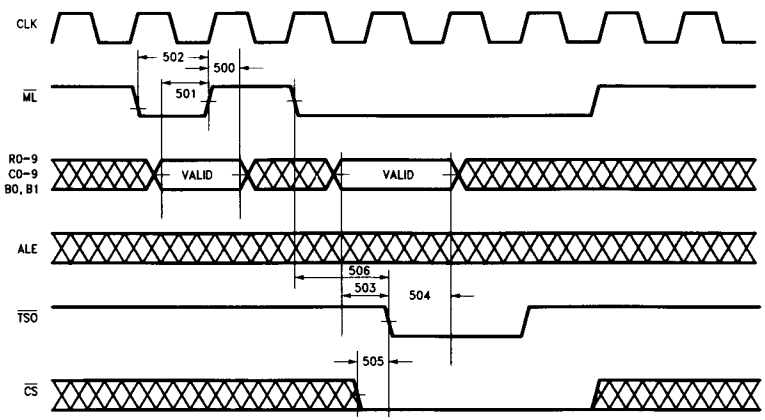
FIGURE 43. 300: Access Interleaving

12.0 AC Timing Parameters: NS32CG821A (Continued)



TL/F/10126-46

FIGURE 44. 400: COLINC Page/Static Column Access Timing



TL/F/10126-47

FIGURE 45. 500: Programming

13.0 Timing Differences Between the NS32CG821A and the NS32CG821

The only differences between the NS32CG821A and the NS32CG821 are AC timing. The 4 tables below list the differences.

TABLE I

Number	Symbol	Common Parameter Description	NS32CG821-25				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
10a	t_{RAH}	Row Address Hold Time (t_{RAH} = 15)	17		15		15		15	
10b	t_{RAH}	Row Address Hold Time (t_{RAH} = 25)	27		25		25		25	
11a	t_{ASC}	Column Address Setup Time (t_{ASC} = 0)	2		0		0		0	
11b	t_{ASC}	Column Address Setup Time (t_{ASC} = 10)	12		10		10		10	
17	t_{PCLKWH}	CLK to $\overline{WAIT}$ Negated		39		39		24		31
19	t_{PEWL}	$\overline{ECAS}$ Negated to $\overline{WAIT}$ Asserted During a Burst Access		34		34		36		36
24	$t_{PWINWEH}$	$\overline{WIN}$ Asserted to $\overline{WE}$ Asserted		34		44		27		37
25	t_{WINWEL}	$\overline{WIN}$ Negated to $\overline{WE}$ Negated		31		41		27		37
28	t_{SCINEN}	COLINC Asserted to Q0–9 Incremented	15		17		17		19	
29a	$t_{SARQCK1}$	$\overline{TSO}$ Negated Setup to CLK High with 1 Period of Precharge	34		34		37		37	

TABLE II

Number	Symbol	Difference Parameter Description	NS32CG821-25				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
55	t_{D5}	(CLK to Refresh $\overline{RAS}$ Asserted) Minus (CLK to Refresh $\overline{RAS}$ Negated)		5		5		6		6

TABLE III

Number	Symbol	Memory Access Parameter Description	NS32CG821-25				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
312	t_{PCSWL}	$\overline{CS}$ Negated to $\overline{WAIT}$ Negated		22		22		25		25
314	t_{PALEWL}	ALE Asserted to $\overline{WAIT}$ Asserted ($\overline{CS}$ is Already Asserted)		39		39		29		29
316	t_{PCKCVO}	CLK to Column Address Valid (t_{RAH} = 15 ns, t_{ASC} = 0 ns)		66		75		60		75

TABLE IV

Number	Symbol	Programming Parameter Description	NS32CG821-25				NS32CG821A-25			
			C_L		C_H		C_L		C_H	
			Min	Max	Min	Max	Min	Max	Min	Max
500	t_{HMLADD}	Mode Address Held from $\overline{ML}$ Negated	5		5		7		7	
504	$t_{HADAQML}$	Mode Address Held from $\overline{TSO}$ Asserted	29		29		38		38	

14.0 NS32CG821A User Hints

1. All inputs to the NS32CG821A should be tied high, low or the output of some other device.

   Note: One signal is active high. COLINC should be tied low to disable.

2. Each ground on the NS32CG821A must be decoupled to the closest on-chip supply (V_{CC}) with 0.1 μF ceramic capacitor. This is necessary because these grounds are kept separate inside the NS32CG821A. The decoupling capacitors should be placed as close as possible with short leads to the ground and supply pins of the NS32CG821A.
3. The output called "CAP" should have a 0.1 μF capacitor to ground.
4. The NS32CG821A has 20Ω series damping resistors built into the output drivers of $\overline{RAS}$, $\overline{CAS}$, address and $\overline{WE}$. Space should be provided for external damping resistors on the printed circuit board (or wire-wrap board) because they may be needed. The value of these damping resistors (if needed) will vary depending upon the output, the capacitance of the load, and the characteristics of the trace as well as the routing of the trace. The value of the damping resistor also may vary between the wire-wrap board and the printed circuit board. To determine the value of the series damping resistor it is recommended to use an oscilloscope and look at the furthest DRAM from the NS32CG821A. The undershoot of $\overline{RAS}$, $\overline{CAS}$, $\overline{WE}$ and the addresses should be kept to less than 0.5V below ground by varying the value of the damping resistor. The damping resistors should be placed as close as possible with short leads to the driver outputs of the NS32CG821A.
5. The circuit board must have a good V_{CC} and ground plane connection. If the board is wire-wrapped, the V_{CC} and ground pins of the NS32CG821A, the DRAM associated logic and buffer circuitry must be soldered to the V_{CC} and ground planes.
6. The traces from the NS32CG821A to the DRAM should be as short as possible.
7. $\overline{ECAS0}$ should be held low during programming if the user wishes that the NS32CG821A be compatible with NSCG821 design.
8. PARAMETER CHANGES DUE TO LOADING

   All A.C. parameters are specified with the equivalent load capacitances, including traces, of 64 DRAMs organized as 4 banks of 18 DRAMs each. Maximums are based on worst-case conditions. If an output load changes then the A.C. timing parameters associated with that particular output must be changed. For example, if we changed our output load to

   C = 250 pF loads on $\overline{RAS}$0–3 and $\overline{CAS}$0–3

   C = 760 pF loads on Q0–9 and $\overline{WE}$

   we would have to modify some parameters (not all calculated here)

   $308a Clock to $\overline{CAS}$ asserted
   (t_{RAH} = 15 ns, t_{ASC} = 0 ns)

   A ratio can be used to figure out the timing change per change in capacitance for a particular parameter by using the specifications and capacitances from heavy and light load timing.

$$\text{Ratio} = \frac{\$308a\ \text{w/heavy load} - \$308a\ \text{w/light load}}{C_H(\overline{CAS}) - C_L(\overline{CAS})}$$

$$= \frac{79\ \text{ns} - 72\ \text{ns}}{125\ \text{pF} - 50\ \text{pF}} = \frac{7\ \text{ns}}{75\ \text{pF}}$$

$$\$308a\ (\text{actual}) = (\text{capacitance difference} \times \text{ratio}) + \$308a\ (\text{specified})$$

$$= \left(250\ \text{pF} - 125\ \text{pF}\right)\frac{7\ \text{ns}}{75\ \text{pF}} + 79\ \text{ns}$$

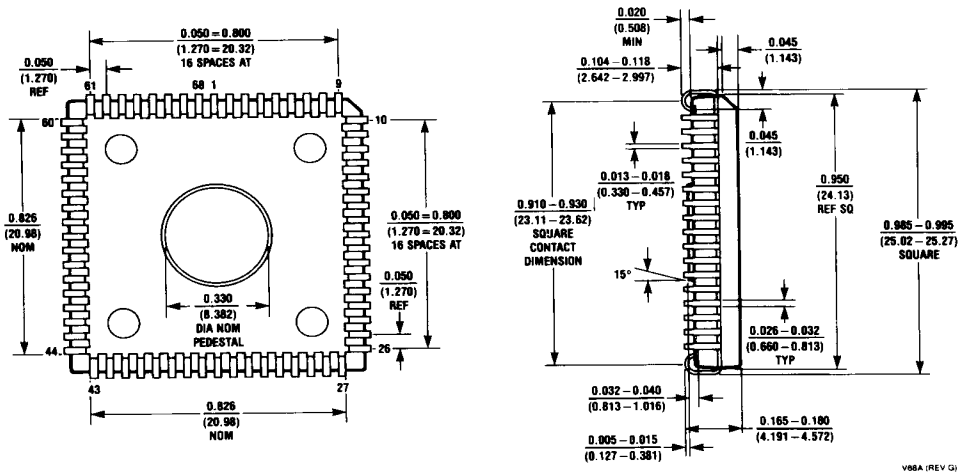
$$= 11.7\ \text{ns} + 79\ \text{ns}$$

$$= 90.7\ \text{ns @ 250 pF load}$$

9. It is recommended that the user perform an external hardware reset of the NSCG821A before programming and using the chip. A hardware reset consists of asserting both $\overline{ML}$ and $\overline{DISRFSH}$ for a minimum of 16 positive edges of CLK.

Physical Dimensions inches (millimeters)

Lit. # 114291



Plastic Chip Carrier (V)
Order Number NS32CG821AV-20 or NS32CG821AV-25
NS Package Number V68A

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.

NOV 1 9 1990 020729 ✓

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: (408) 721-5000
TWX: (910) 339-9240

National Semiconductor GmbH
Industriestrasse 10
D-8080 Furstenfeldbruck
West Germany
Tel: (0-81-41) 103-0
Telex: 527-649
Fax: (08141) 103554

National Semiconductor Japan Ltd.
Sanseido Bldg. 5F
4-15 Nishi Shinjuku
Shinjuku-Ku,
Tokyo 160, Japan
Tel: 3-299-7001
FAX: 3-299-7000

National Semiconductor Hong Kong Ltd.
Suite 513, 5th Floor
Chinachem Golden Plaza,
77 Mody Road, Tsimshatsui East,
Kowloon, Hong Kong
Tel: 3-7231290
Telex: 52996 NSSEA HX
Fax: 3-3112536

National Semicondutores Do Brasil Ltda.
Av. Brig. Faria Lima, 1383
6.0 Andor-Conj. 62
01451 Sao Paulo, SP, Brasil
Tel: (55/11) 212-5066
Fax: (55/11) 211-1181 NSBR BR

National Semiconductor (Australia) PTY, Ltd.
1st Floor, 441 St. Kilda Rd.
Melbourne, 3004
Victory, Australia
Tel: (03) 267-5000
Fax: 61-3-2677458