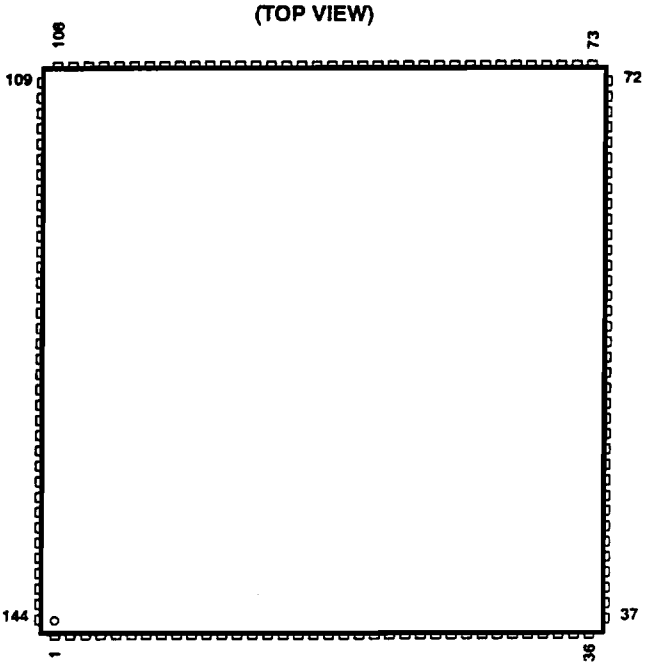
SPVS008 – JUNE 1992

- **Instruction Cycle Time:**
  - **100 ns . . . 340X- 40**
  - **125 ns . . . 340X-32**
- **Fully Probrammable 32-Bit Processor for Low Cost X-Terminals and Other Embedded Applications**
- **512-Megabyte (4-Gigabit) Linear Address Range (Bit Addressable)**
- **Second-Generation Graphics System Processor**
  - **Object Code Compatible With the TMS34010 and TMS34020**
  - **Enhanced Instruction Set**
  - **Optimized Graphics Instructions**
  - **Coprocessor Interface**
- **Pixel Processing, XY Addressing, and Window Checking Built into the Instruction Set**
- **Programmable 1-, 2-, 4-, 8-, 16-, or 32-Bit Pixel Size With 16 Boolean and 6 Arithmetic Pixel Processing Options (Raster-Ops)**
- **512-Byte LRU On-Chip Instruction Cache**
- **Optimized DRAM/VRAM Interface**
  - **Page Mode for Burst Memory Operations up to 40 Megabytes per Second**
  - **Dynamic Bus Sizing (16-Bit and 32-Bit Transfers)**
  - **Byte-Oriented CAS Strobes**
- **Flexible Multi-Processor Interface**

144-PIN QUAD FLATPACK†
(TOP VIEW)



- **Programmable CRT Control**
  - **Composite Sync Mode**
  - **Separate Sync Mode**
  - **Synchronization to External Sync**
- **Direct Support for Special Features of 1M and 4M VRAMs**
  - **Load Write Mask**
  - **Load Color Mask**
  - **Block Write**
  - **Write Using the Write Mask**

description

The TMS340X is a graphics processor designed for use in low cost x-terminals. The TMS340X and the Graphics System Processors (GSPs) are the second generation of an advanced high-performance CMOS 32-bit microprocessor optimized for graphics display systems. With a built-in instruction cache, the ability to simultaneously access memory and registers, and an instruction set designed to expedite raster graphics operations, the TMS340X provides user-programmable control of the CRT interface as well as the memory interface (both standard DRAM and multiport video RAM). The 4-gigabit (512-megabyte) physical address space is addressable on bit boundaries using variable width data fields (1 to 32 bits). Additional graphics addressing modes support 1-, 2-, 4-, 8-, 16- and 32-bit wide pixels.

architecture

The TMS340X is a CMOS 32-bit processor with hardware support for graphics operations such as PixBlts (raster ops) and curve-drawing algorithms. Also included is a complete set of general-purpose instructions with addressing modes tuned to support high-level languages. In addition to its ability to address a large external memory range, the TMS340X contains 30 general-purpose 32-bit registers, a hardware stack pointer, and a 512-byte instruction cache. On-chip functions include 64 programmable I/O registers that control CRT timing, input/output control, and parameters required by some instructions. The TMS340X directly interfaces to

dynamic RAMs and video RAMs and generates raster control signals. The TMS340X also accommodates a multiprocessing or direct memory access (DMA) environment through the request/grant interface protocols. Virtual memory systems are supported through bus fault detection and instruction continuation.

The TMS340X provides single-cycle execution of general-purpose instructions and most common integer arithmetic and Boolean operations from its instruction cache. Additionally, the TMS340X incorporates a hardware barrel shifter that provides a single-state bidirectional shift and rotate function for 1 to 32 bits.

The local memory controller is designed to optimize memory access operations. It also supports pipeline memory write operations of variable-sized fields and allows memory access and instruction execution in parallel.

The TMS340X graphics processing hardware supports pixel and pixel-array processing capabilities for both monochrome and color systems at a variety of pixel sizes. The hardware incorporates two-operand and three-operand raster operations with Boolean and arithmetic operations, XY addressing, window clipping, window checking operations, 1 to *n* bits-per-pixel transforms, transparency, and plane masking. The architecture further supports operations on single pixels (PIXT instructions) or on two-dimensional arrays of arbitrary size (PixBlts).

The TMS340X's flexible graphics processing capabilities allow software-based graphics algorithms without sacrificing performance. These algorithms include clipping to arbitrary window size, custom incremental curve drawing, two-operand raster operations, and masked two-operand raster operations.

The TMS340X provides for extensions to the basic architecture through the coprocessor interface. Special instructions and cycle timings are included to enhance data flow to coprocessors without requiring the coprocessor to decode the instruction stream, generate system addresses, or move data for the coprocessor through the TMS340X.

pin assignments – quad-flatpack package

PIN NO.	FUNCTION	PIN NO.	FUNCTION	PIN NO.	FUNCTION	PIN NO.	FUNCTION
1	V_{SS}	37	V_{SS}	73	V_{SS}	109	V_{SS}
2	V_{CC}	38	$\overline{HCS}$	74	V_{CC}	110	V_{CC}
3	$\overline{CAS3}$	39	NC1	75	LAD0	111	LAD29
4	$\overline{CAS2}$	40	NC1	76	LAD16	112	LAD14
5	$\overline{CAS1}$	41	NC1	77	LAD1	113	LAD30
6	$\overline{CAS0}$	42	NC1	78	LAD17	114	LAD15
7	V_{CC}	43	NC1	79	LAD2	115	LAD31
8	$\overline{RAS}$	44	NC1	80	LAD18	116	SCLK
9	V_{SS}	45	NC1	81	V_{SS}	117	RCA12
10	$\overline{R0}$	46	NC1	82	LAD3	118	RCA11
11	$\overline{R1}$	47	NC1	83	LAD19	119	RCA10
12	NC3	48	NC1	84	V_{CC}	120	RCA9
13	NC3	49	NC1	85	LAD4	121	RCA8
14	NC3	50	NC1	86	LAD20	122	RCA7
15	$\overline{HINT}$	51	NC1	87	LAD5	123	RCA6
16	EMU3	52	NC1	88	LAD21	124	RCA5
17	LCLK1	53	NC1	89	LAD6	125	V_{CC}
18	LCLK2	54	V_{SS}	90	LAD22	126	V_{SS}
19	EMU1	55	V_{SS}	91	LAD7	127	RCA4
20	EMU0	56	NC1	92	LAD23	128	RCA3
21	EMU2	57	NC1	93	V_{SS}	129	RCA2
22	$\overline{GI}$	58	NC1	94	V_{SS}	130	RCA1
23	$\overline{RESET}$	59	NC1	95	LAD8	131	RCA0
24	$\overline{LINT2}$	60	NC1	96	LAD24	132	SF
25	$\overline{LINT1}$	61	NC1	97	LAD9	133	$\overline{TR/QE}$
26	CAMD	62	NC1	98	LAD25	134	$\overline{VSYNC}$
27	BUSFLT	63	NC1	99	LAD10	135	$\overline{HSYNC}$
28	$\overline{SIZE16}$	64	NC1	100	LAD26	136	$\overline{CBLNK/VBLNK}$
29	$\overline{PGMD}$	65	NC1	101	LAD11	137	$\overline{CSYNC/HBLNK}$
30	LRDY	66	NC1	102	LAD27	138	V_{SS}
31	V_{CC}	67	NC1	103	V_{CC}	139	V_{SS}
32	V_{CC}	68	NC1	104	LAD12	140	$\overline{ALTCH}$
33	VCLK	69	NC1	105	LAD28	141	DDIN
34	CLKIN	70	NC1	106	V_{SS}	142	$\overline{DDOUT}$
35	NC2	71	NC1	107	LAD13	143	$\overline{WE}$
36	NC2	72	V_{SS}	108	V_{SS}	144	V_{SS}

NC1 No connect pins that should be tied to V_{SS}.
NC2 No connect pins that should be tied to V_{CC}.
NC3 No connect pins that must not be connected to anything. (They must not be tied together.)
$\overline{HCS}$ Host chip select. $\overline{HCS}$ no longer initiates Host cycles. $\overline{HCS}$ is still sampled during reset, as described on page 2-14 of the TMS34020 User's Guide. In a uniprocessor X-terminal the $\overline{HCS}$ pin will normally be tied to V_{SS} (self bootstrap mode).

signal descriptions

NAME	I/O	DESCRIPTION
		LOCAL MEMORY INTERFACE
$\overline{\text{ALTCH}}$	O	**Address latch.** The high-to-low transitions of $\overline{\text{ALTCH}}$ can be used to capture the address and status present of the LAD signals. A transparent latch (such as a 74ALS373) will maintain the current address and status as long as $\overline{\text{ALTCH}}$ remains low.
BUSFLT	I	**Bus fault.** External logic asserts BUSFLT high to the TMS340X to indicate that an error or fault has occurred on the current bus cycle. BUSFLT is also used with LRDY to generate externally requested bus cycle retries so that the entire memory address is presented again on the LAD pins. In emulation mode, BUSFLT is used for write protecting mapped memory (by disabling $\overline{\text{CAS}}$ outputs for the current cycle).
DDIN	O	**Data bus direction in enable.** This active-high output is used to drive the active-high output-enables on bidirectional transeivers (such as the 74ALS623). The transceivers buffer data input and output on the LAD0-LAD31 pins when the TMS340X is interfaced to several memories.
$\overline{\text{DDOUT}}$	O	**Data bus direction output-enable.** This active-low signal drives the active-low output-enables on bidirectional transceivers (such as the 74ALS623). The transceivers buffer data input and output on the LAD0-LAD31 pins.
LAD0-LAD31	I/O	**32-bit multiplexed local address/data bus.** At the beginning of a memory cycle, the word address is output on LAD4-LAD31 and the cycle status is output on LAD0-LAD3. After the address is presented, LAD0-LAD31 are used for transferring data within the TMS340X system. LAD0 is the LSB and LAD31 is the MSB.
LRDY	I	**Local ready.** External circuitry drives this signal low to inhibit the TMS340X from completing a local-memory cycle it has initiated. While LRDY remains low the TMS340X will wait, unless the TMS340X loses bus priority or is given an external RETRY request (through the BUSFLT signal). Wait states are generated in increments of one full LCLK1 cycle. LRDY can be driven low to extend local-memory read and write cycles, VRAM serial-data-register transfer cycles, and DRAM refresh cycles. During internal cycles, the TMS340X ignores LRDY.
$\overline{\text{PGMD}}$	I	**Page mode.** The memory decode logic asserts this signal low if the currently addressed memory supports burst (page mode) accesses. Burst accesses occur as a series of $\overline{\text{CAS}}$ cycles for a single $\overline{\text{RAS}}$ cycle to memory. LRDY is used with BUSFLT to describe the cycle termination status for a memory cycle. $\overline{\text{PGMD}}$ is also used in emulation mode for mapping memory.
$\overline{\text{SIZE16}}$	I	**Bus size.** The memory decode logic may pull this signal low if the currently addressed memory or port supports only 16-bit transfers. $\overline{\text{SIZE16}}$ can also be used to determine which 16 bits of the data bus are used for a data transfer. In emulation mode, $\overline{\text{SIZE16}}$ is used to select the size of mapped memory.
		DRAM and VRAM CONTROL
CAMD	I	**Column-address mode.** This input dynamically shifts the column address on the RCA0-RCA12 bus to allow the mixing of DRAM and VRAM address matrices using the same multiplexed address RCA0–RCA12 signals.
$\overline{\text{CAS0}}$-$\overline{\text{CAS3}}$	O	**4 column-address strobes.** The $\overline{\text{CAS}}$ outputs drive the $\overline{\text{CAS}}$ inputs of DRAMs and VRAMs. These signals strobe the column address on RCA0-RCA12 to the memory. The four $\overline{\text{CAS}}$ strobes provide byte write-access to the memory.
$\overline{\text{RAS}}$	O	**Row-address strobe.** The $\overline{\text{RAS}}$ output drives the $\overline{\text{RAS}}$ inputs of DRAMs and VRAMs. This signal strobes the row address on RCA0-RCA12 to memory.
RCA0-RCA12	O	**13 multiplexed row-address/column-address signals.** At the beginning of a memory-access cycle, the row address for DRAMs is present on RCA0-RCA12. The row address contains the most significant address bits for the memory. As the cycle progresses, the memory column address is placed on RCA0-RCA12. The addresses that are actually output during row and column times depend on the memory configuration (set by RCM0 and RCM1 in the CONFIG register) and the state of CAMD during the access. RCA0 is the LSB and RCA12 is the MSB.
SF	O	**Special-function.** This is the special-function signal to 1M VRAMs which allows the use of block write, load write mask, load color mask, and write using write mask. This signal is also used to differentiate instructions and addresses for the coprocessor as part of the coprocessor interface.
$\overline{\text{TR/QE}}$	O	**Transfer/output-enable.** This signal drives the $\overline{\text{TR/QE}}$ input of VRAMs. During a local-memory read cycle, $\overline{\text{TR/QE}}$ functions as an active-low output-enable to gate from memory to LAD0-LAD31. During special VRAM function cycles, $\overline{\text{TR/QE}}$ controls the type of cycle that is performed.
$\overline{\text{WE}}$	O	**Write-enable.** The active low $\overline{\text{WE}}$ output drives the $\overline{\text{WE}}$ inputs of DRAMs and VRAMs. $\overline{\text{WE}}$ can also be used as the active-low write-enable to static memories and other devices connected to the TMS340X local interface. During a local-memory read cycle, $\overline{\text{WE}}$ remains inactive high while $\overline{\text{CAS}}$ is strobed active low. During a local-memory write cycle, $\overline{\text{WE}}$ is strobed active low before $\overline{\text{CAS}}$ is. During VRAM serial-data-register transfer cycles, the state of $\overline{\text{WE}}$ at the falling edge of $\overline{\text{RAS}}$ controls the direction of the transfer.

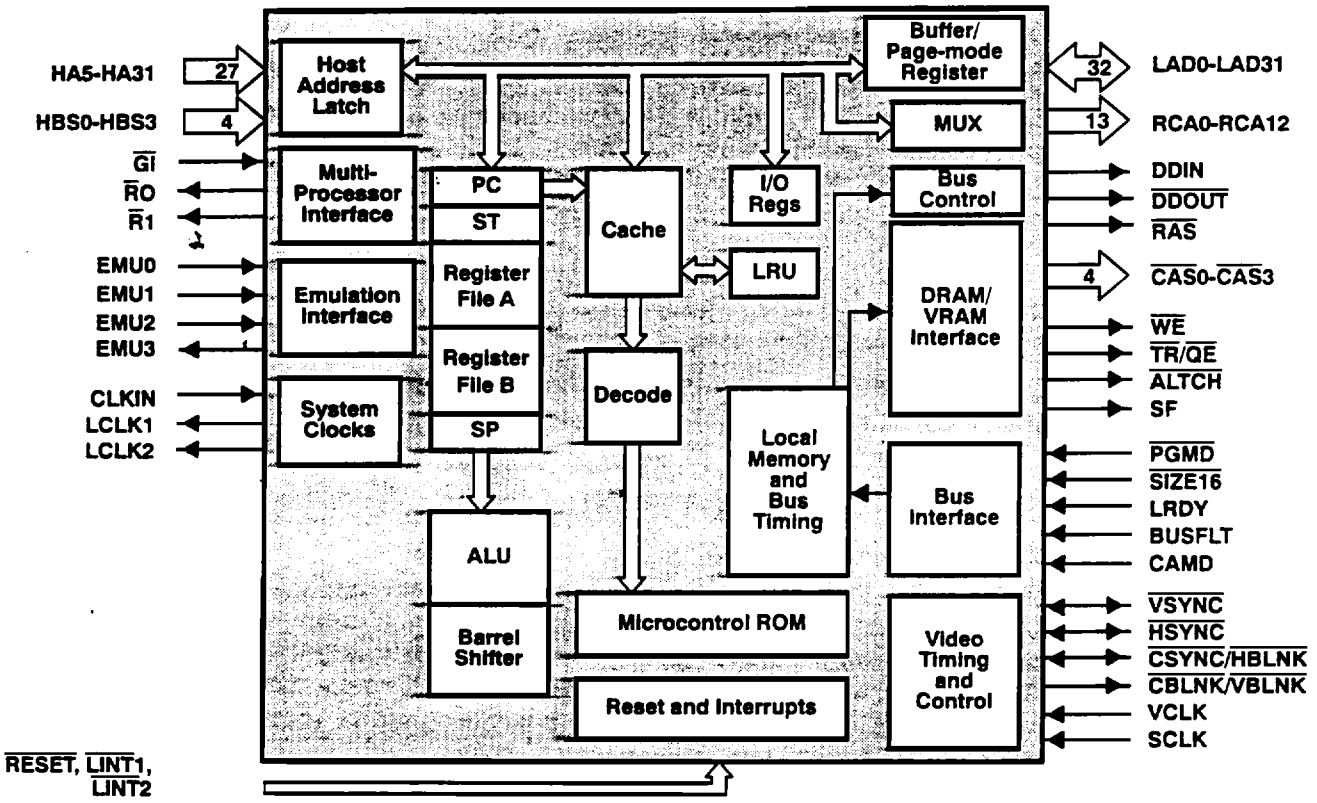
signal descriptions (continued)

HOST INTERFACE		
NAME	**I/O**	**DESCRIPTION**
SYSTEM CONTROL		
CLKIN	I	**Clock input.** This system input clock generates the LCLK1 and LCLK2 outputs, to which all processor functions in the TMS340X are synchronous. A separate asynchronous input clock (VCLK) controls the video timing and video registers.
LCLK1, LCLK2	O	**Local output clocks.** These two clocks are 90 degrees out of phase with each other. They provide convenient synchronous control of external circuitry to the internal timing. All signals output from the TMS340X (except the CRT timing signals) are synchronous to these clocks.
$\overline{\text{LINT1}}$, $\overline{\text{LINT2}}$	I	**Local interrupt requests.** Interrupts from external devices are transmitted to the TMS340X on $\overline{\text{LINT1}}$ and $\overline{\text{LINT2}}$. Each local interrupt signal activates the request for one of two interrupt request levels. An external device generates an interrupt request by driving the appropriate interrupt request pin to its active-low state. The signal should remain low until the TMS340X recognizes it. These signals can be applied asynchronously to the TMS340X as they are synchronized internally before use.
$\overline{\text{RESET}}$	I	**System reset.** During normal operation, $\overline{\text{RESET}}$ is driven low to reset the TMS340X. When $\overline{\text{RESET}}$ is asserted low, the TMS340X's internal registers are set to an initial known state and all output and bidirectional pins are driven either to inactive levels or to a high impedance state. The TMS340X's behavior following reset depends on the level of the $\overline{\text{HCS}}$ input just before the low-to-high transition of $\overline{\text{RESET}}$. If $\overline{\text{HCS}}$ is low, the TMS340X begins executing the instructions pointed to by the reset vector. If $\overline{\text{HCS}}$ is high, the TMS340X is halted until a host processor writes a 0 to the HLT bit in the HSTCTLL register.
POWER		
V_{CC}†	I	Nominal 5-V power supply inputs. 5 pins on QFP; 9 pins on PGA.
V_{SS}†	I	Electrical ground inputs. 9 pins on QFP; 17 pins on PGA.
EMULATION CONTROL		
EMU0-2†	I	Emulation pins 0-2.
EMU3	O	Emulation pin 3.
MULTIPROCESSOR INTERFACE		
NAME	**I/O**	**DESCRIPTION**
$\overline{\text{GI}}$	I	**Bus grant input.** External bus arbitration logic drives $\overline{\text{GI}}$ low to enable the TMS340X to gain access to the local-memory bus. The TMS340X must release the bus if $\overline{\text{GI}}$ is high so that another device can access the bus.
$\overline{\text{R1}}$, $\overline{\text{R0}}$	O	**Bus request and control.** These two signals indicate a request for use of the bus in a multiprocessor system; they are decoded as shown below: $\overline{\text{R1}}$ $\overline{\text{R0}}$ **Bus Request Type** 0 0 High-priority bus request 0 1 Bus cycle termination 1 0 Low-priority bus request 1 1 No bus request pending A high-priority bus request provides for VRAM serial-data-register transfer cycles (midline or blanked), DRAM refresh (when 12 or more refresh cycles are pending), or a host-initiated access. The external arbitration logic should grant request as soon as possible by asserting $\overline{\text{GI}}$ low. A low-priority bus request is used to provide for CPU-requested access and DRAM refresh (when less than 12 refresh cycles are pending). Bus cycle termination status is provided so that the arbitration logic can determine that the device currently accessing the bus is completing an access and other devices may compete for the next bus cycle. A *no bus request pending* status is output when the currently active device does not require the bus on subsequent cycles.

signal descriptions (concluded)

LOCAL MEMORY INTERFACE		
NAME	**I/O**	**DESCRIPTION**
VIDEO INTERFACE		
$\overline{\text{CBLNK}}$ / $\overline{\text{VBLNK}}$	O	**Composite blanking/vertical blanking.** You can program this signal to select one of two blanking functions: **Composite blanking** for blanking the display during both horizontal and vertical retrace periods in *composite-sync video mode.* **Vertical blanking** for blanking the display during vertical retrace in *separate-sync-video mode.* Immediately following reset, this signal is configured as a $\overline{\text{CBLNK}}$ output.
$\overline{\text{CSYNC}}$ / $\overline{\text{HBLNK}}$	I/O	**Composite sync/horizontal blanking.** You can program this signal to select one of two functions. **Composite sync** (either input or output as set by a control bit in the DPYCTL register) in *composite-sync video mode:* *Input:* Extracts $\overline{\text{HSYNC}}$ and $\overline{\text{VSYNC}}$ from externally generated horizontal sync pulses. *Output:* Generates active-low composite-sync pulses from either externally generated $\overline{\text{HSYNC}}$ and $\overline{\text{VSYNC}}$ signals or signals generated by the TMS340X's on-chip video timers. **Horizontal blank** (output only) for blanking the display during horizontal retrace in *separate-sync-video mode.* Immediately following reset, this signal is configured as a $\overline{\text{CSYNC}}$ input.
$\overline{\text{HSYNC}}$	I/O	**Horizontal sync. $\overline{\text{HSYNC}}$** is the horizontal sync signal that controls external video circuitry. You can program this signal to be either an input or an output by modifying a control bit in the DPYCTL register. As an **output,** $\overline{\text{HSYNC}}$ is the active-low horizontal sync signal generated by the TMS340X's on-chip video timers. As an **input,** $\overline{\text{HSYNC}}$ synchronizes the TMS340X video-control registers to externally generated horizontal sync pulses. The actual synchronization can be programmed to begin at any VCLK cycle; this allows for any external pipelining of signals. Immediately following reset, $\overline{\text{HSYNC}}$ is configured as an input.
SCLK	I	**Serial data clock.** This signal is the same as the signal that drives VRAM serial data registers. This allows the TMS340X to track the VRAM serial data register count, providing serial register transfer and midline reload cycles. (SCLK may be asynchronous to VCLK; however, it typically has a frequency that is a multiple of the VCLK frequency).
VCLK	I	**Video clock.** This clock is derived from a multiple of the video system's dotclock and is used internally to drive the video timing logic.
$\overline{\text{VSYNC}}$	I/O	**Vertical sync. $\overline{\text{VSYNC}}$** is the vertical sync signal that controls external video circuitry. You can program this signal to be either an input or an output by modifying a control bit in the DPYCTL register. As an **output,** $\overline{\text{VSYNC}}$ is the active-low vertical sync signal generated by the TMS340X's on-chip video timers. As an **input,** $\overline{\text{VSYNC}}$ synchronizes the TMS340X video-control registers to externally generated vertical sync pulses. The actual synchronization can be programmed to begin at any horizontal line; this allows for any external pipelining of signals. Immediately following reset, $\overline{\text{VSYNC}}$ is configured as an input.

functional block diagram



architecture (continued)

register files

Boolean, arithmetic, pixel-processing, byte, and field move instructions operate on data within the general-purpose register files. The TMS340X contains two register files of fifteen 32-bit registers and a system stack pointer (SP). The SP is addressed in both Register File A and Register File B as a sixteenth register. Transfers between registers and memory are facilitated via a complete set of field MOVE instructions with selectable field sizes.

The fifteen general-purpose registers in Register File A are used for high-level language support and assembly-language programming. The fifteen registers in Register File B are dedicated to special functions during PixBlts and other pixel operations but can be used as general-purpose registers at other times.

stack pointer (SP)

The stack pointer is a dedicated 32-bit internal register that points to the top of the system stack.

program counter (PC)

The TMS340X's 32-bit program counter register points to the next instruction-stream word to be fetched. Since instruction words are aligned to 16-bit boundaries, the four LSBs of the PC are always zero.

instruction cache

An on-chip cache contains 512 bytes of RAM and provides unimpeded access to instructions. The cache operates automatically and is transparent to software. The cache is divided into four 128-byte segments. Associated with each segment is a 22-bit segment start address register (SSA) to identify the addresses in

memory corresponding to the current contents of the cache segment. Each cache segment is further partitioned into eight subsegments of four long-words (32-bits) each. Each subsegment has associated with it a present (P) flag to indicate whether the subsegment contains valid data.

The cache is loaded only when an instruction requested by the execution section of the TMS340X is not already contained within the cache. A least-recently-used (LRU) algorithm determines which of the four segments of the cache is overwritten with new data. For this purpose, an internal four-by-two LRU stack keeps track of cache usage. Although the cache is loaded so as to always fill a subsegment completely, not all eight subsegments within a segment are necessarily filled (this is dependent upon the instruction stream).

status register

The status register (ST) is a special-purpose 32-bit register dedicated to status codes set by the results of implicit and explicit compare operations and parameters used to specify the length and behavior of fields 0 and 1.

In addition, during an interrupt, the IX bit in the ST placed on the stack provides indication that execution of an interruptable instruction (PIXBLT, FILL or LINE) was halted to service the interrupt.

The single step bit causes a TRAP to the single step vector (located at address FFFF FBE0h) after the execution of one instruction when the bit is set high. Normal program execution occurs when the bit is set low.

fields, bytes, words, long-words, pixels, and pixel arrays

The TMS340X outputs a 28-bit address on LAD4-LAD31, which is valid at the falling edge of $\overline{\text{ALTCH}}$. The most significant 27 bits (LAD5-LAD31) define a 32-bit long-word of physical memory; logically, however the TMS340X views memory data as fields addressable at the bit level. The least significant bit of the 28-bit address (LAD4) is used to select the odd or even word when accessing 16-bit memories (indicated by $\overline{\text{SIZE16}}$ asserted low). Primitive data types supported by the TMS340X include bytes, words, long-words, pixels, two independent fields of 1 to 32 bits, and user-defined pixel arrays.

Words and long-words, respectively, refer to 16- and 32-bit values that are aligned on 32-bit boundaries.

The two independent fields are referenced as Field 0 and Field 1. The attributes of these fields (field size and sign extension within a register) are defined in the status register as FS0, FE0, FS1, and FE1. Fields 0 and 1 are specified independently to be signed or unsigned and from 1 to 32 bits in length. Bytes are special 8-bit cases of the field data type, while pixels are 1, 2, 4, 8, 16, or 32 bits in length. In general, fields (including bytes) may start and terminate on arbitrary bit boundaries; however, pixels must pack evenly into 32-bit long-words.

pixel operations

Pixel arrays are two-dimensional data types of user-defined width, length, pixel depth (number of bits per pixel), and pitch (distance between rows). A pixel or pixel array may be accessed by means of either its memory address or its XY coordinates. Transfers of individual pixels or pixel blocks are influenced by the pixel processing, transparency, window checking, plane masking, pixel masking, or corner adjustment operations selected. For further information, see the *TMS32020 User's Guide*, literature number SPVU019.

transparency

Transparency is a mechanism that allows surrounding pixels in an array to be specified as invisible. This is useful for ensuring that only the object and not the rectangle surrounding it are written to the display. The TMS340X provides four transparency modes:

- No transparency
- Transparency on result equal zero
- Transparency on source equal COLOR0
- Transparency on destination equal COLOR0
- Refer to the *TMS34020 User's Guide* for more information.

I/O registers

The TMS340X contains an on-chip block of sixty-four 16-bit locations (mapped into the TMS340X's memory address space) that are used for I/O control registers. Eight of these are used by the host interface logic and are not available to the user. Forty-seven I/O registers control parameters necessary to configure the operation and report status of the following interfaces:

- host interface
- local memory
- video timing
- screen refresh
- external interrupts
- internal interrupts

memory interface control registers

Some of the I/O registers are used to control various local memory interface functions, including:

- Frequency of DRAM refresh cycles
- Masking (read/write protection) of individual color planes
- DRAM row/column addressing configuration
- Accessing mode (big endian/little endian)
- Bus fault and retry recovery

video timing and screen refresh

Twenty-eight I/O registers are dedicated to video timing and screen refresh functions. The TMS340X may be configured to drive composite sync or separate sync displays.

In composite mode, the TMS340X can be set to extract $\overline{\text{VSYNC}}$ and $\overline{\text{HSYNC}}$ from an external $\overline{\text{CSYNC}}$, or it can be used to generate $\overline{\text{CSYNC}}$ from separate $\overline{\text{VSYNC}}$ and $\overline{\text{HSYNC}}$ inputs. Internally, the TMS340X can be set to preset the horizontal and vertical counts on receipt of an external sync signal. This allows compensation for any combination of internal and external delays that occur in the video synchronization process. The HCOUNT register is loaded from SETHCNT by an external $\overline{\text{HSYNC}}$, VCOUNT is loaded from SETVCNT on an external $\overline{\text{VSYNC}}$, and an external $\overline{\text{CSYNC}}$ loads both HCOUNT and VCOUNT from SETHCNT and SETVCNT, respectively.

The TMS340X directly supports multiport video RAMs (VRAMs) by generating the serial data register transfer cycles necessary to refresh the display. The memory locations from which the display information is taken, as well as the number of horizontal scan lines displayed between serial data register transfer cycles, are programmable.

The TMS340X supports various display resolutions and either interlaced or noninterlaced video. The TMS340X can optionally be programmed to synchronize to externally generated sync signals so that images created by the TMS340X may be superimposed upon images created externally. The external sync mode may also be used to synchronize the video signals generated by two or more TMS340Xs in a multiple-TMS340X graphics system.

CPU control registers

Five of the I/O registers (CONVDP, CONVMP, CONVSP, CONTROL, and PSIZE) provide CPU control to configure the TMS340X for operation with specific characteristics. These characteristics include pitches for pixel transfers, window checking mode, Boolean or arithmetic pixel processing operation, transparency mode, PixBlt direction control, and pixel size.

interrupt interface registers

Two dedicated I/O registers (INTENB and INTPEND) monitor and mask interrupt requests to the TMS340X, including two externally generated interrupts and three internally generated interrupts. An internal interrupt request can be generated on one of the following conditions.

- Window violation: an attempt has been made to write a pixel to a location inside or outside a specified window boundary
- Host interrupt: the host processor has set the interrupt request bit in the host control register.
- Display interrupt: a specified horizontal line in the frame has been displayed on the screen.
- Bus fault
- Single-step emulator

A non-maskable interrupt occurs when the host processor sets a control bit in the host interface register (NMI in HSTCTLH). The host-initiated interrupt is associated with a mode bit (NMIM in HSTCTLH) that enables and disables saving of the processor state on the stack when the interrupt occurs. This is useful if the host wishes to use the host interrupt before releasing the TMS340X to execute instructions (i.e., before the stack pointer is initialized). The TMS340X reset function is controlled by a dedicated pin.

memory controller/local memory interface

The memory controller manages the TMS340X's interface to the local memory and automatically performs the bit alignment and masking necessary to access data located at arbitrary bit boundaries within memory. The memory controller operates autonomously with respect to the CPU. It has a write queue one field (1 to 32 bits) deep that permits it to complete the memory cycles necessary to insert a field into memory without delaying the execution of subsequent instructions. Only when a second memory operation is required before completion of the first operation is the TMS340X forced to defer execution of the subsequent instruction.

The TMS340X directly interfaces to standard dynamic RAMs and, in particular, to standard video RAMs such as the TMS44C251 Multiport VRAMs. The TMS340X memory interface consists of the Local Address/Data bus (LAD), the DRAM Row/Column Address (RCA) bus, and associated control signals. The currently selected Word Address (28 bits) and Status (4 bits) are multiplexed with data on the LAD bus. The RCA bus allows direct connection to address/address multiplexed DRAMs from 64K to 16M. Refresh for DRAMs is supported by $\overline{\text{CAS}}$-before-$\overline{\text{RAS}}$ refresh cycles.

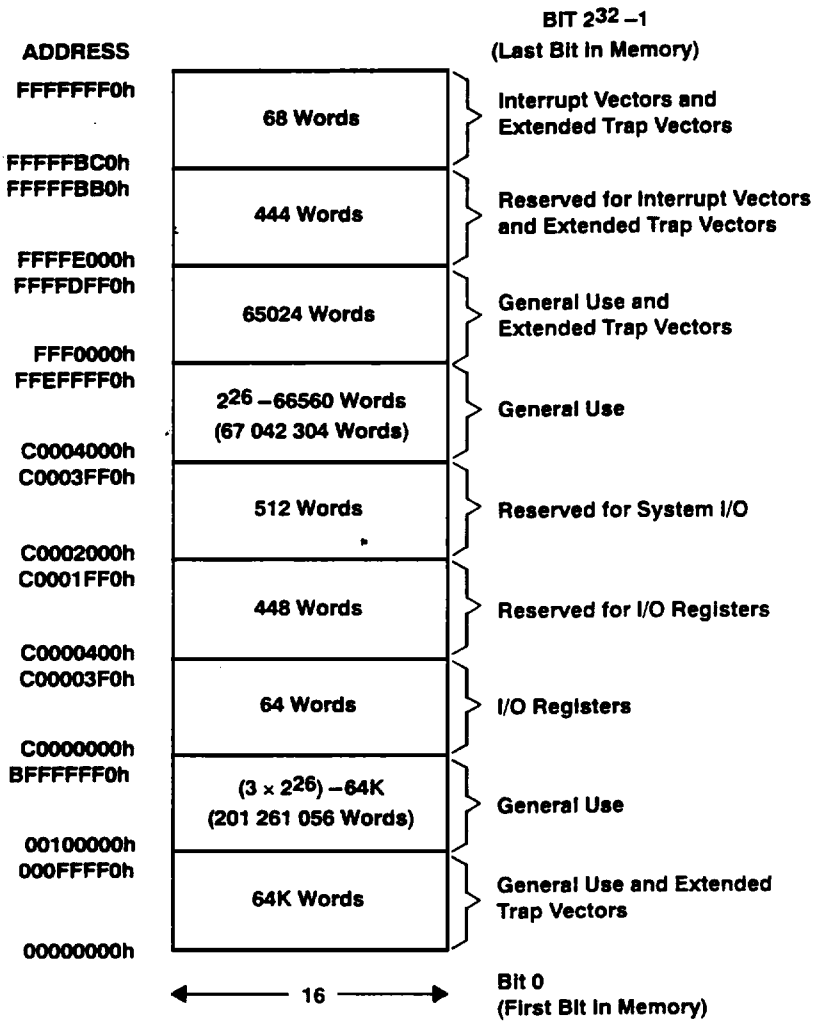


Figure 1. Memory Map

reset

Reset puts the TMS340X into a known initial state. This state is entered when the input signal at the $\overline{\text{RESET}}$ pin is asserted low. While $\overline{\text{RESET}}$ remains asserted, all outputs are in a known state, no DRAM refresh cycles take place, and no screen refresh cycles are performed.

At the time the TMS340X fetches the level-0 vector address (the reset vector), the least significant four bits (bit address part) are used to load configuration data that establishes the initial condition of the *big endian/little endian* mode and the current RCA bus configuration bits in the CONFIG register as described in the I/O register section.

Unlike other interrupts and software traps, reset does not save the previous ST or PC values (this can also occur on host-initiated non-maskable interrupts if the NMIM bit in HSTCTLH is set to a 1), since the value of the stack pointer just before a reset is generally not valid, and saving these values on the stack could contaminate valid memory locations. A TRAP 0 instruction, which uses the same vector address as reset, similarly does not save the ST or PC values.

asserting reset

A reset is initiated by asserting the $\overline{\text{RESET}}$ input pin to its active-low level. To reset the TMS340X at power up, $\overline{\text{RESET}}$ must remain active low for a minimum of 40 local clock periods (LCLK1 and LCLK2) after power levels have become stable. At times other than power up, the TMS340X may be reset by holding $\overline{\text{RESET}}$ low for a minimum of 4 local clock periods; the GSP will enter an internal reset state for 34 local clock cycles. While in the internal reset state and $\overline{\text{RESET}}$ is high, memory refresh cycles occur.

reset and multiprocessor synchronization

The synchronization of multiple TMS340Xs sharing a local memory is done using the $\overline{\text{RESET}}$ input. In systems where the Multiprocessor Interface is used to control the access to a common memory, the processors must be synchronized. Synchronization is achieved by taking $\overline{\text{RESET}}$ high within a specific interval relative to CLKIN. This may be done by using CLKIN to clock the $\overline{\text{RESET}}$ as received by the TMS340Xs. All TMS340Xs to be synchronized should use the same CLKIN and $\overline{\text{RESET}}$ inputs. All of the local memory and bus control signals should be connected in parallel (without buffers) between the processors. After power up, the processors are not necessarily synchronized with respect to the particular quarter cycle in progress. The rising edge of $\overline{\text{RESET}}$ is used to set the TMS340X to a particular quarter cycle by adding Q1 cycles. All TMS340Xs in a multiprocessor environment operate on the same quarter cycle within 10 quarter cycles after the rising edge of $\overline{\text{RESET}}$.

reset and DRAM/VRAM initialization

The TMS340X drives its $\overline{\text{RAS}}$ signal inactive high as long as $\overline{\text{RESET}}$ remains low. The specifications for certain DRAM and VRAM devices require that the $\overline{\text{RAS}}$ signal be driven inactive-high for 1 ms after power is stable to provide the proper conditions for the DRAMs. Typically, eight $\overline{\text{RAS}}$ cycles are also required to initialize the DRAMs for proper operation. In general, holding $\overline{\text{RESET}}$ low for *t* microseconds ensures that $\overline{\text{RAS}}$ remains high initially for *t–(10tQ)* microseconds. The TMS340X memory controller automatically inserts the required eight $\overline{\text{RAS}}$ cycles after all resets (after power up or after the internal reset state) by issuing $\overline{\text{CAS}}$-before-$\overline{\text{RAS}}$ refresh cycles before it allows the CPU access to memory. A host must delay requests to memory until the initialization cycles have had sufficient time to complete. Immediately following reset, the TMS340X is set to perform a refresh sequence every eight cycles.

At times other than power up, to maintain the memory in DRAMs and do a reset, the $\overline{\text{RESET}}$ pulse must not exceed the maximum refresh interval of the DRAMs minus the time for the TMS340X to refresh the memories. On reset, the TMS340X is set to do a refresh cycle every eight local clock periods. A 32 MHz (CLKIN) system with one (refresh) bank of D/VRAM would be completely refreshed in one-sixteenth of the total memory refresh interval. The reset pulse then should not exceed about fifteen-sixteenths of the total refresh interval required by the DRAMs to maintain memory integrity.

If the $\overline{\text{RESET}}$ signal remains low longer than the maximum refresh interval specified for the memory, the previous contents of the local memory may not be valid after the reset.

initial state following reset

While the $\overline{\text{RESET}}$ pin is asserted low (or while in the internal reset state), the TMS340X's output and bidirectional pins are forced to the states below.

INITIAL STATE OF PINS FOLLOWING A RESET (WITH $\overline{\text{GI}}$ LOW)		
OUTPUTS DRIVEN HIGH	**OUTPUTS DRIVEN LOW**	**BIDIRECTIONAL DRIVEN TO HIGH-IMPEDANCE**
$\overline{\text{RAS}}$	$\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$	$\overline{\text{VSYNC}}$
$\overline{\text{CAS0}}$-$\overline{\text{CAS3}}$	DDIN	$\overline{\text{HSYNC}}$
$\overline{\text{WE}}$		$\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$
$\overline{\text{TR}}$/$\overline{\text{QE}}$		LAD0-LAD31
$\overline{\text{DDOUT}}$		
$\overline{\text{ALTCH}}$		
$\overline{\text{R0}}$		
$\overline{\text{R1}}$		
EMU3		
RCA0-RCA12		
SF		

NOTE: If $\overline{\text{GI}}$ is high, then all $\overline{\text{GI}}$-controlled pins will be high-impedance. The $\overline{\text{GI}}$-controlled pins are $\overline{\text{RAS}}$, $\overline{\text{CAS0}}$-$\overline{\text{CAS3}}$, $\overline{\text{WE}}$, $\overline{\text{TR}}$/$\overline{\text{QE}}$, DDOUT, DDIN, $\overline{\text{ALTCH}}$, $\overline{\text{HOE}}$, HDST, RCA0-RCA12, LAD0-LAD31, and SF.

Immediately following reset, all I/O registers are cleared (set to 0000), with the exception of the HLT bit in the HSTCTLH register. The HLT bit is set to 0.

Just prior to the execution of the first instruction in the reset routine, the TMS340X's internal registers are in the following states:

- General-purpose register files A and B are uninitialized.
- The ST is set to 0000 0010h.
- The PC contains the most-significant 28 bits of the vector fetched from memory address FFFF FFE0h (the least significant four bits of the PC are set to zero).
- The BEN bit in the I/O register CONFIG is set to the least significant bit read from the vector fetched from memory address FFFF FFE0h.
- The CBP, RCM0, and RCM1 bits in the I/O register CONFIG are set to the corresponding bits read from the vector fetched from memory address FFFF FFE0h. The Configuration Byte Protect bit (CBP) may be set high to prevent further modification of the lower eight bits of the I/O register CONFIG.

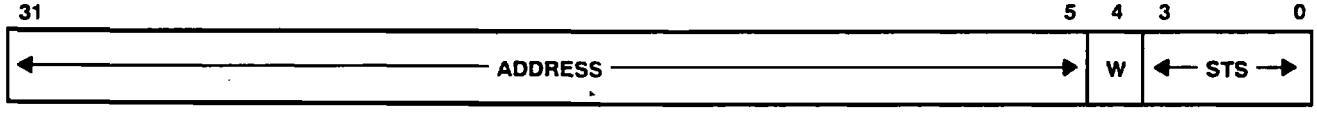
The state of the instruction cache at this time is as follows:

- The SSA (segment start address) registers are uninitialized.
- The LRU (Least Recently Used) stack is set to the initial sequence 0, 1, 2, 3, where 0 occupies the most-recently-used position and 3 occupies the least-recently-used position.
- All P (Present) flags are cleared to 0s.

local memory and DRAM/VRAM interface

The TMS340X local memory interface consists of an address/data multiplexed bus on which address and data are transmitted. The associated control signals support memory widths of 16 or 32 bits, burst (page-mode) accesses, local memory wait states, and optional external data bus buffers. The TMS340X DRAM/VRAM interface consists of an address/address multiplexed bus and the control signals to interface directly to both DRAMs and VRAMs. The local memory interface and the DRAM/VRAM interface are interrelated and therefore considered together for this description. At the beginning of a typical memory cycle, the address and status of the current cycle are output on the LAD bus while the ROW address is output on the Row/Column Address (RCA) bus. $\overline{\text{ALTCH}}$ and $\overline{\text{RAS}}$ are used to latch the address/status and ROW address, respectively, on these two buses. The LAD bus is then used to transfer data to or from the memory while the RCA bus is set to the column address for the memory. (NOTE: LAD31 is the most significant bit of the address or data).

LAD DURING THE ADDRESS CYCLE



ADDRESS — Memory address (select for 128M 32-bit long-words)
W = 0 — Access to lower 16-bit word (even-addressed word or 32-bit boundary)
W = 1 — Access to upper 16-bit word (odd-addressed word)
STS — Bus cycle status code

The address output on the Row/Column Address (RCA) lines is determined by the Row/Column Mode bits (RCM0 and RCM1 in the I/O registers CONFIG) and the state of the Column Address Mode pin (CAMD) during each memory cycle. The CAMD signal is sampled on the internal Q4 clock phase, which allows CAMD to be generated by static logic wired to the Local Address/Data (LAD) bus.

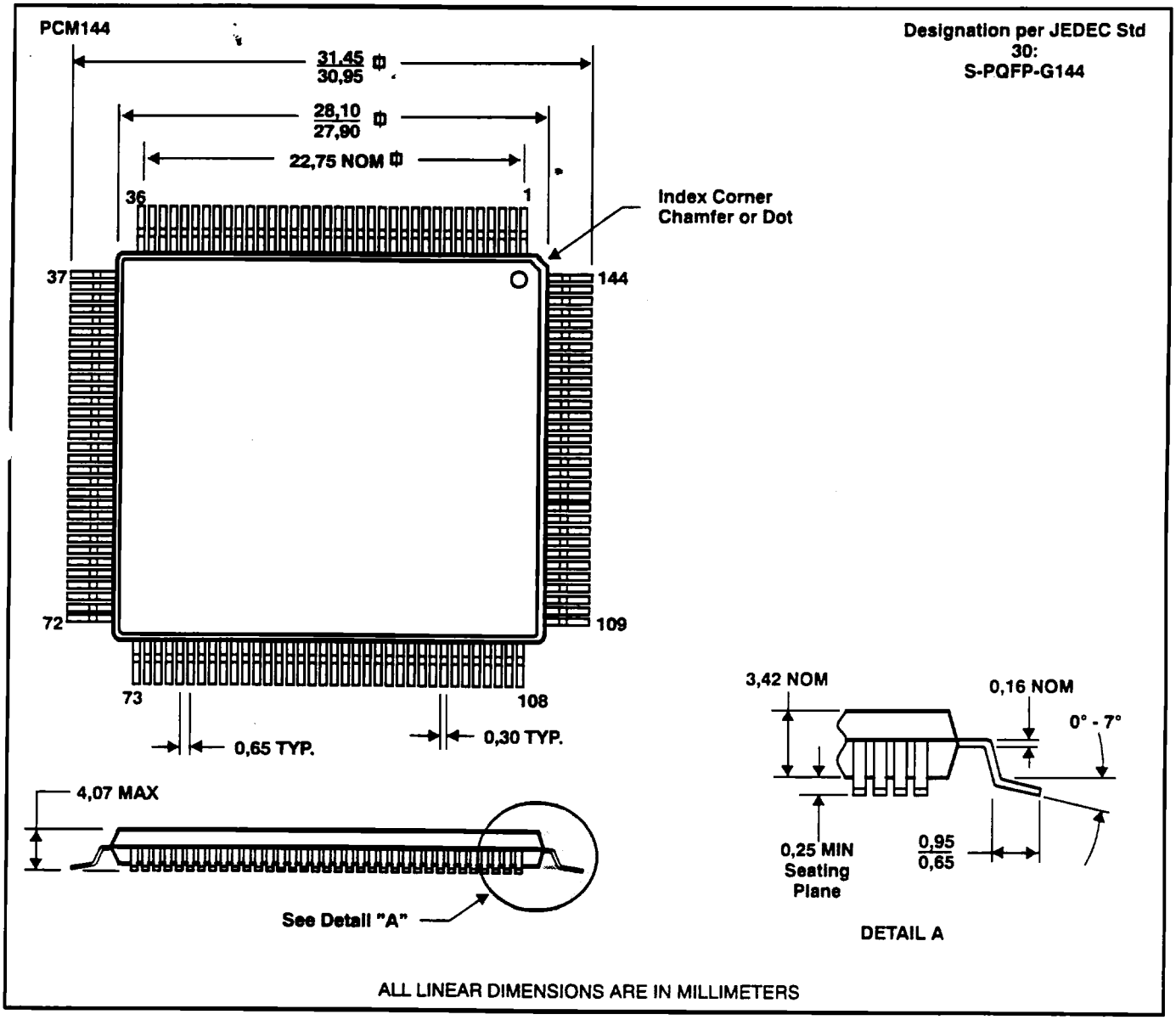
BASIC MEMORY ROW/COLUMN ACCESS MODES					
RCM1	RCM	VRAM MODE	ADDRS	BANKS	CAMD SUPPORT MATRICES
0	0	64K × N	8	16	64K × 16, 64K × 32, 256K × 16, 256K × 32, 1M × 16, 1M × 32
0	1	256K × N	9	8	2564K × 16, 256K × 32, 1M × 16, 1M × 32, 4M × 32
1	0	1M × N	10	4	1M × 16, 1M × 32, 4M × 16, 4M × 32
1	1	4M × N	11	2	4M × 16, 4M × 32, 16M × 32

VRAM Mode = basic size of VRAM addressing supported with CAMD = 0.
Addrs = number of RCA signals required to provide row/column addressing.
Banks = number of possible interleaved 32-bit wide memory spaces.
CAMD Support = possible sizes and configurations of DRAMs that may be supported within the basic VRAM mode.

MECHANICAL DATA

PCM144
JEDEC metric plastic quad flatpack

This plastic package consists of a circuit mounted on a lead frame and encapsulated within an electrically nonconductive plastic compound. The compound withstands soldering temperatures with no deformation, and circuit performance characteristics remain stable when the devices are operated in high-humidity conditions. The package is intended for surface mounting and leads are spaced on 0,65 mm centers with an 0,80 mm foot length. Leads require no additional cleaning or processing when used in soldered assembly.



NOTES: A. Maximum deviation from coplanarity is 0,1 mm.
B. All dimensions and notes for JEDEC outline MO-xxxxx apply.

status codes

Status codes are output on LAD0-LAD3 at the time of the falling edge of $\overline{\text{ALTCH}}$ and may be used to determine the type of cycle that is being initiated. The following table lists the codes and their respective meanings.

CODE	STATUS	TYPE
0000	Coprocessor Code	Other (00XX)
0001	Emulator Operation	
0010	Host Cycle	
0011	DRAM Refresh	
0100	Video-generated DRAM Serial Register Transfer	VRAM (01XX)
0101	CPU-generated VRAM Serial Register Transfer	
0110	Write Mask Load	
0111	Color Latch Load	
1000	Data Access	CPU (1XXX)
1001	Cache Fill	
1010	Instruction Fetch	
1011	Interrupt Vector Fetch	
1100	Bus Locked Operation	
1101	Pixel Operation	
1110	Block Write	
1111	– Reserved –	

dynamic bus sizing

The TMS340X supports dynamic bus sizing between 16 and 32 bits on any local memory access. Any port/memory that is only 16 bits wide must assert $\overline{\text{SIZE16}}$ low during Q1 (to be valid at the start of Q2) of the bus cycle accessing the even memory word (LAD4 = 0) corresponding to its address.The TMS340X then performs another memory access to the next 16-bit (odd) word in memory. The TMS340X samples the $\overline{\text{SIZE16}}$ pin at the start of Q2 in the second cycle (access to odd word address) to determine to which half of the LAD bus the port or memory is aligned. If the port is on LAD0-LAD15, the $\overline{\text{SIZE16}}$ input should be low during the second cycle access (odd word); otherwise, if the port is on LAD16-LAD31, the $\overline{\text{SIZE16}}$ input must be high at this time. The TMS340X always performs two memory cycles to access the 16-bit wide memories, even when attempting only a 16-bit transfer.

The TMS340X outputs the four $\overline{\text{CAS}}$ strobes and LAD bus initially aligned for a 32-bit bus. If the memory is 16 bits wide, the two most significant $\overline{\text{CAS}}$ strobes are swapped with the two least significant strobes when it accesses the second word, and the halves of the LAD bus are also swapped; therefore, 16-bit memories need to respond only to the two $\overline{\text{CAS}}$ strobes corresponding to the upper or lower 16 bits of the LAD bus to which they are connected.

Note that devices connected to LAD0-LAD15 transfer the least significant word during the first cycle and the most significant word during the second cycle. Data accesses on LAD16-LAD31 transfers the most significant word first, then the least significant word.

The second memory cycle forced by the $\overline{\text{SIZE16}}$ pin is performed as a page mode access if $\overline{\text{PGMD}}$ was low during the first access. A read-write cycle to 16-bit page-mode memory requires five bus cycles that occur as address, read0, read1, write0, write1. If a 16-bit transfer is interrupted due to a bus fault, the restart causes the entire access to be restarted.

For memory that supports page mode accesses ($\overline{\text{PGMD}}$ low), $\overline{\text{SIZE16}}$ is sampled during each access to memory. If $\overline{\text{SIZE16}}$ is high on the even word access, then a 32-bit transfer occurs over LAD0-LAD31. If $\overline{\text{SIZE16}}$ is low on the even word access (16-bit wide memory), then it is sampled again on the odd word access to determine to which half of the LAD bus the memory is connected (low for connection to LAD0-LAD15 or high for connection to LAD16-LAD31).

special 1M VRAM cycles

The TMS340X provides control for special function VRAM cycles that are available in the 1M devices. These cycles are obtained by the appropriate timing control of the SF, $\overline{CAS}$, $\overline{TR}/\overline{QE}$, and $\overline{WE}$ pins of the VRAMs at the falling edge of $\overline{RAS}$. The cycles include:

- Load Write Mask
- Load Color Mask
- Block Write (no mask)
- Block Write (current mask)
- Write Using Mask
- Alternate Write Transfer

In addition, other special modes may be implemented by using external logic.

multiprocessor arbitration

The multiprocessor interface allows multiple processors to operate in a system sharing the same local memory. The use of the Grant In ($\overline{GI}$) and the Priority Request signals ($\overline{R}0$ and $\overline{R}1$) allows a flexible method of passing control from one processor to another. The control scheme allows local memory cycles to occur back-to-back, even when passing control from one TMS340X to another. Synchronization of multiple TMS340Xs in a system occurs at reset with the rising edge of $\overline{RESET}$ meeting the setup and hold requirements to CLKIN, so all TMS340Xs are certain to respond to the $\overline{RESET}$ during the same quarter cycle. $\overline{RESET}$ is not required to be synchronous to CLKIN, except to allow synchronization of multiple TMS340Xs in a system.

The $\overline{GI}$ priority for multiprocessing environments is determined by arbitration logic external to the TMS340X. If $\overline{GI}$ goes inactive (high), the TMS340X releases the bus on the next available cycle boundary. If the cycle in progress has not successfully completed, the TMS340X restarts the cycle upon regaining control of the bus. Normally, if the TMS340X asserts both $\overline{R}0$ and $\overline{R}1$ low, it should be given the control of the bus by the arbitrator.

coprocessor interface

Support for coprocessors is provided through special instructions and bus cycles that allow communication with the coprocessor. A coprocessor may be register-based, depending on the TMS340X to do all address calculations, or it may operate as its own bus controller, using the multiprocessor arbitration scheme. Five basic cycles are provided for direct communication and control of coprocessors.

1. TMS340X to coprocessor
2. Coprocessor to TMS340X
3. Move memory to coprocessor
4. Move coprocessor to memory
5. Coprocessor internal command

The first four of these cycles provide for command of the coprocessor in addition to the movement of parameters to and from the coprocessor. In this manner, parameters may be sent to the coprocessor and operated upon without an explicit coprocessor command cycle.

Instruction set

The TMS340X instruction set may be divided into five categories:

1. Graphics instructions
2. Coprocessor instructions
3. Move instructions
4. General-purpose instructions
5. Program control and context switching

Specialized *graphics instructions* manipulate pixel data that is accessed via memory addresses or XY coordinates. These instructions include graphics operations, such as array and raster ops, pixel processing, windowing, plane masking, pixel masking, and transparency. *Coprocessor* instructions allow for the control and data flow to and from coprocessors that reside in the system. *Move* instructions comprehend the bit addressing and field operations, which manipulate fields of data using linear addressing for transfer to and from memory and the register file. *General-purpose* instructions provide a complete set of arithmetic and Boolean operations on the register file, as well as general program control and data processing. Program control and context switching allows the user to control flow and to save and restore information using instructions with both register-direct and absolute operands.

clock stretch

The TMS340XA supports a clock stretching mechanism, which is described in outline below.

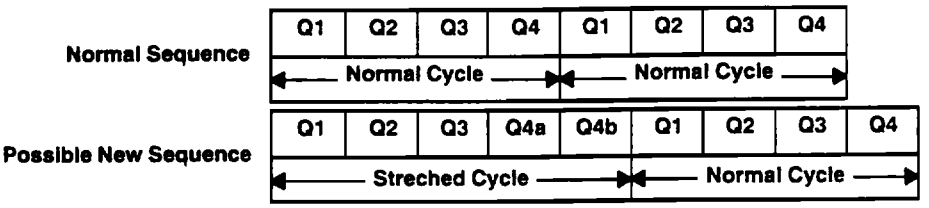
With advances in semiconductor manufacturing, new versions of the TMS340X can be made, each supporting a higher CLKIN frequency. The increase in CLKIN frequency means that the TMS340X machine cycles execute more quickly, with a consequent increase in code execution speed. However, there comes a point when, as the machine cycle time becomes shorter, the local memory control signals begin to violate DRAM and VRAM timing parameters for certain types of memory access.

The clock stretch mechanism allows the TMS340X to slow down and execute those critical local memory cycles, while still benefiting from the accelerated processing allowed by higher CLKIN frequencies during noncritical memory access cycles.

Exact timing issues will vary from system to system, reflecting differences in bus buffering, etc., but broadly speaking the clock stretch mechanism allows the system designer to interface to slower (and hence cheaper and more available) memory devices than the designer could use if no stretch mechanism were available.

A normal, unstretched machine cycle consists of four quarter cycles, Q1, Q2, Q3, and Q4. A stretched cycle consists of five "quarter" cycles Q1, Q2, Q3, and Q4a, and Q4b.

When clock stretch mode is enabled, the fourth machine quarter cycle may be stretched to twice its original length. This stretching takes place only when the TMS3402X attempts certain types of memory cycle.

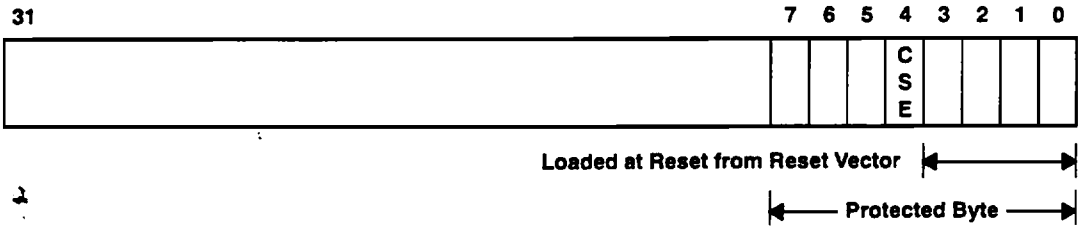


The stretch is achieved by holding the internal TMS340X clocks in the Q4 state for an extra quarter cycle so all the device outputs remain unchanged during Q4a and Q4b. The TMS340X stretches only certain machine cycles so that the execution of code is not slowed unnecessarily.

The TMS340X stretches only certain machine cycles so that the execution of code is not slowed unnecessarily.

enabling clock stretch

Clock stretch mode is enabled and disabled via a bit in the CONFIG register (C00001A0h).



CONFIG register C00001A0h

CSE = 0: Disable stretch mode (normal operation)
CSE = 1: Enable stretch mode

Bit 4 of the CONFIG register is the clock stretch enable mode bit. A 0 in this bit will disable stretch mode and a 1 in this bit will enable stretch mode. The bit is cleared during reset; i.e., stretch mode is disabled by default.

When stretch mode is enabled, the following machine cycles are stretched:

1. All address cycles of all memory access sequences
2. Read data cycles in read-modify-write sequences

Notes:

A. The host default cycle shown on page 8-49 of the *TMS34020 User's Guide* is not stretched because it is not a true address cycle; i.e., $\overline{RAS}$, etc., do not go low.

B. The CPU default cycle, which is similar to the host default cycle in that $\overline{RAS}$, etc., does not go low, and is also not stretched.

C. Clock stretch mode disregards the page-mode input, so that read data cycles in nonpage-mode read-modify-write sequences are stretched, even though there are no timing constraints that require a stretch.

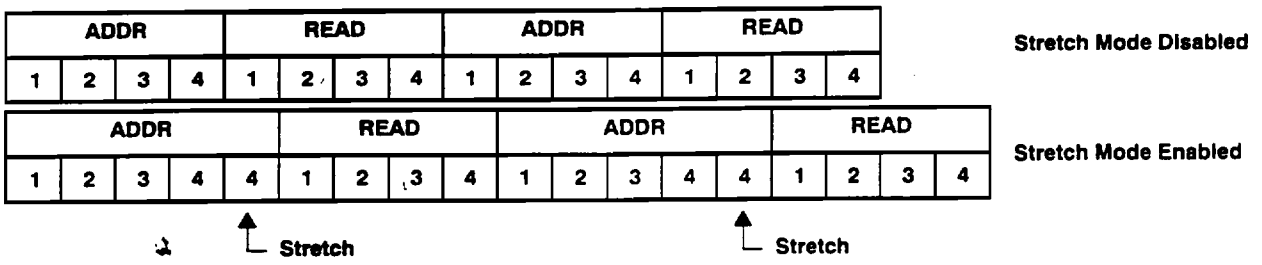
D. All other memory subcycles are *not* stretched, even if the TMS340X is running with the CSE bit set to 1.

The advantage of this implementation of clock stretch mode is that the TMS340X can execute code at maximum speed, only slowing down during certain parts of memory access sequences.

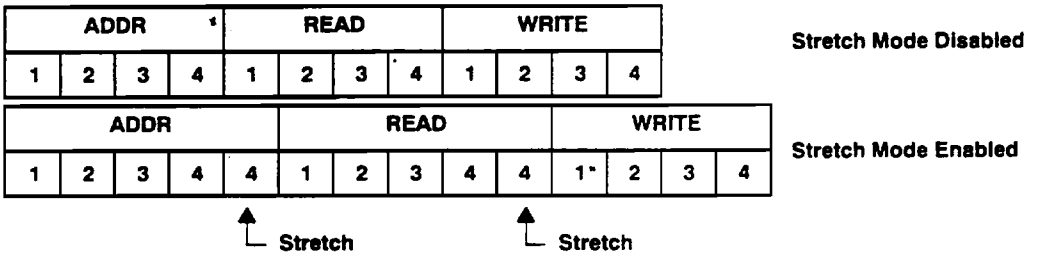
It is important to remember that a stretched cycle is 25% longer than a normal cycle, and that the TMS340X (with the exception of the video logic, which is clocked independently by VCLK) will effectively slow down during such a stretched cycle.

The following are examples of stretch mode memory operations.

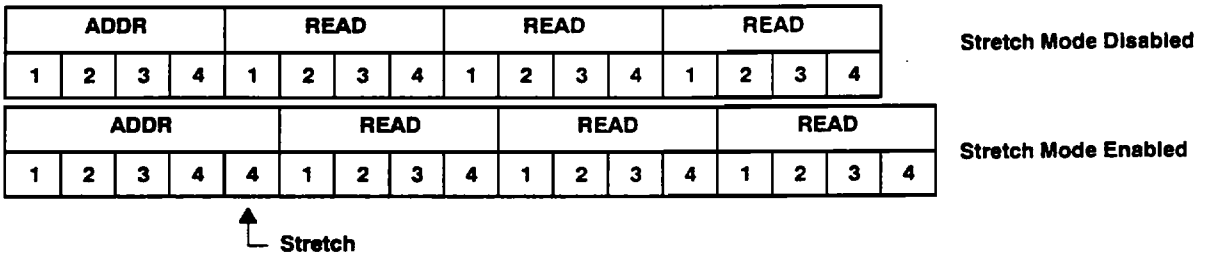
two 32-bit nonpage-mode reads



one 32-bit page-mode read-modify-write

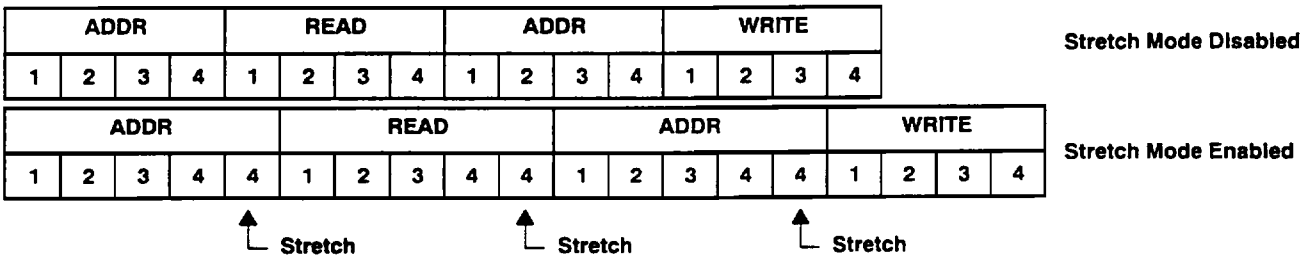


three 32-bit page-mode reads



The stretched cycles are designed to accommodate worst-case 32-bit page-mode accesses, so during some nonpage-mode memory accesses, stretches that are not essential may be generated. For example:

one 32-bit nonpage-mode read-write



Stretches are inserted in read-modify-write accesses to help ease bus "turn-around" timings. In the above example, the second stretch is not needed to help these timings because the read/write "turn-around" has the whole of the address cycle to evaluate.

a clock stretch timing example, TMS340X-40 and 100-ns VRAMs

This example analyzes a memory interface timing parameter. It shows that the clock stretch mechanism can be used to allow the TMS340X-40 to avoid a timing violation when interfaced to 100-ns VRAMs.

Consider a system with:

1. TMS340X-40, which has a 40-MHz clock input frequency and hence a 100-ns cycle time, so t_Q = 25 ns. Timing parameters are taken from the TMS340X data sheet.
2. TMS44C251-10 256K × 4-bit VRAM. Timing parameters are taken from the appropriate section in the Texas Instruments *1989 MOS Memory Data Book.*

row address hold data after $\overline{RAS}$ low, $t_{h(ADV\text{-}REL)}$

Without clock stretch

TMS44C251	$t_{h(RA)}$	Row address hold time after $\overline{RAS}$ low.	Min = 15 ns
TMS340X	Parameter 88	Hold time, row address valid after $\overline{RAS}$ low.	Min = t_Q – 5 ns = 20 ns.

If $\overline{RAS}$ is passed through a PAL with a delay of 7 ns, then $t_{h(RA)}$ seen by the VRAM is 20 ns – 7 ns = 13 ns. This violates the 15 ns minimum.

With clock stretch

TMS340X Parameter 88 $t_{h(ADV\text{-}REL)}$ Hold time, row address valid after $\overline{RAS}$ low. Min = $2t_Q$ – 5 ns = 45 ns.

With the same 7 ns PAL delay, the VRAM sees $t_{h(RA)}$ as 45 ns – 7 ns = 38 ns, which does not violate the VRAM minimum of 15 ns.

cycle timing examples

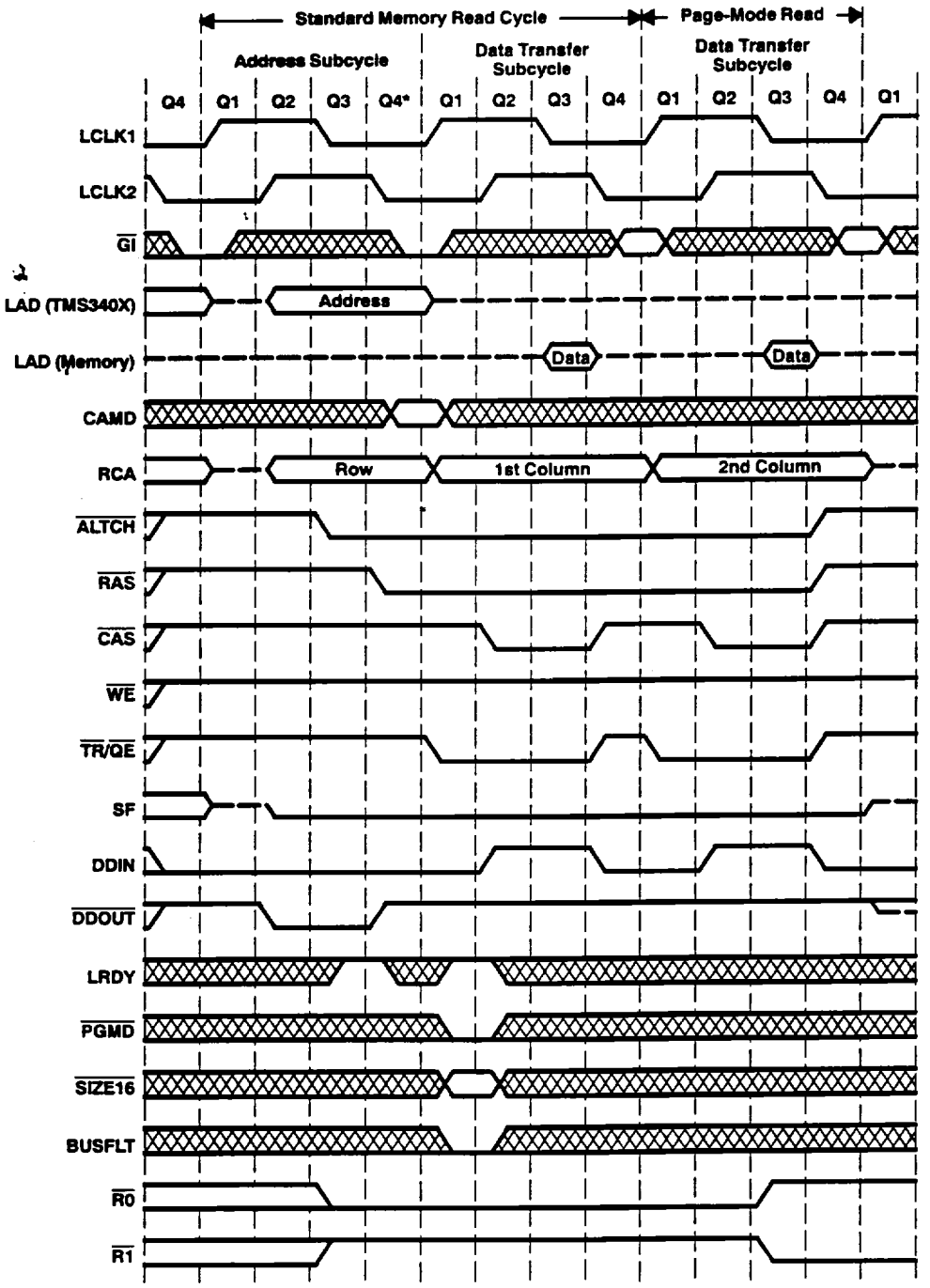
The following figures show examples of many of the basic cycles that the TMS340X uses for memory access, VRAM control, multiprocessor bus control, and coprocessor communication. These figures should not be used to determine specific signal timings, but may be used to see signal relationships for the various cycles. Q4 phases that *could* be stretched are marked with a * on the diagrams. The conditions required for the stretch are:

1. The design uses a TMS340X.
2. The CONFIG register's CSE bit is set to 1.
3. The TMS340X is doing either
   a) Any address cycle, or
   b) A read data cycle in a read-modify-write sequence.

The following remarks apply to memory timing in general. A row address is output on RCA0-RCA12 at the start of a cycle, along with the full address and status on LAD0-LAD31. These remain valid until after the fall of $\overline{ALTCH}$ and $\overline{RAS}$. The column address is then output on RCA0-RCA12, and LAD0-LAD31 are set to read or write data for the memory access. During a write, the data and $\overline{WE}$ are set valid prior to the falling edge of $\overline{CAS}$; the data remains valid until after $\overline{WE}$ and $\overline{CAS}$ have returned high.

Large memory configurations may require external buffering of the address and data lines. The DDIN and $\overline{DDOUT}$ signals coordinate these external buffers with the LAD bus.

ADVANCE INFORMATION

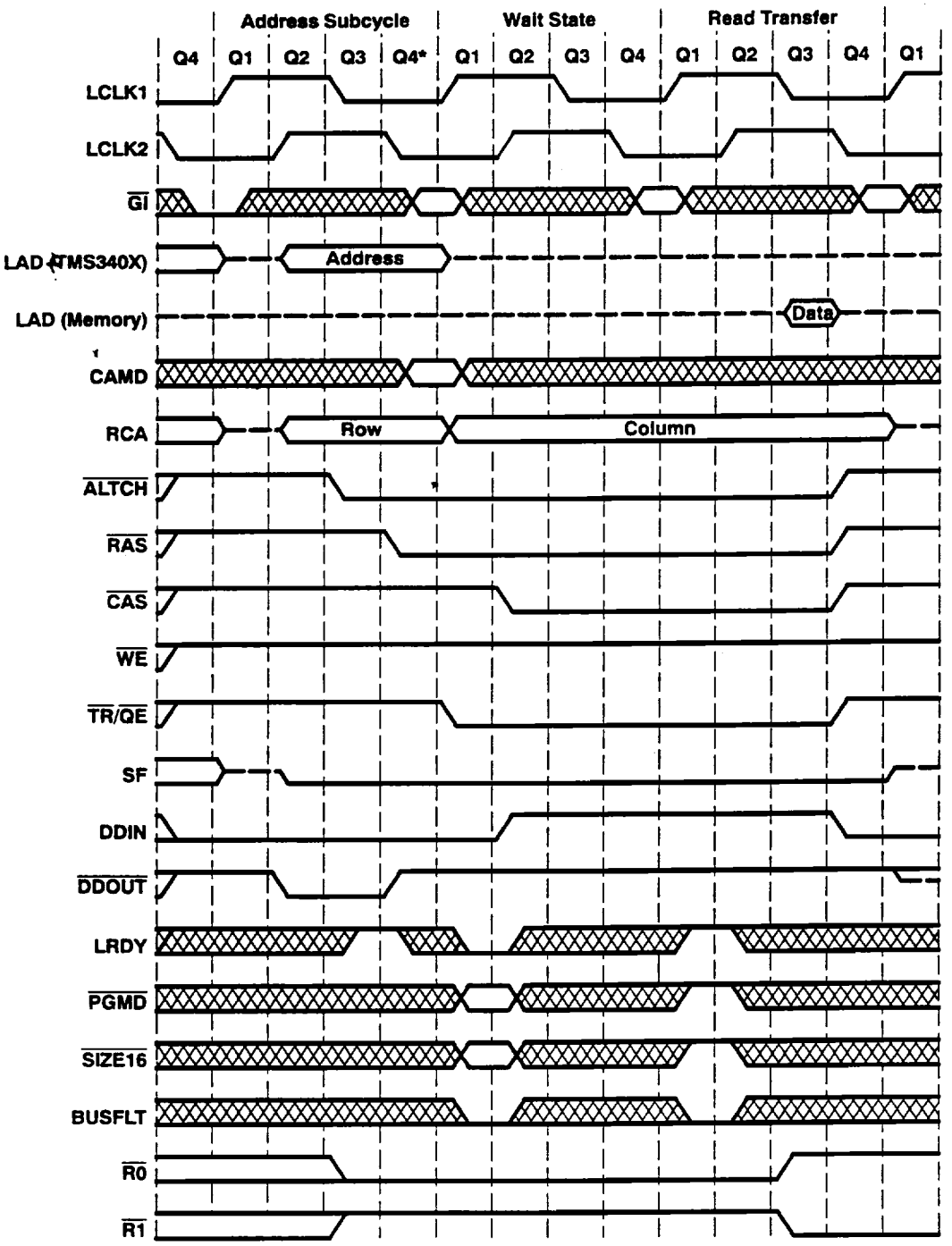


*See "clock stretch", page 18.

NOTES: A. LAD (TMS340X): Output to the LAD bus by the TMS340X.
LAD (memory): Output to the LAD bus by the memory.
B. LRDY, PGMD, SIZE16, and BUSFLT are not sampled on subsequent page mode cycle accesses to 32-bit wide memory space.

Figure 2. Local Memory Read Cycle Timing (With Page Mode)

During the address output to the LAD bus by the TMS340X, the least significant four bits (LAD0-LAD2) contain a bus status code. PGMD low at the start of Q2 after RAS low indicates that this memory supports page-mode operation. LRDY high at the start of Q2 after RAS low indicates that the cycle can continue without inserting wait states. DDOUT returns high after the initial address output on LAD (during Q4), indicating that a memory read cycle is about to take place.



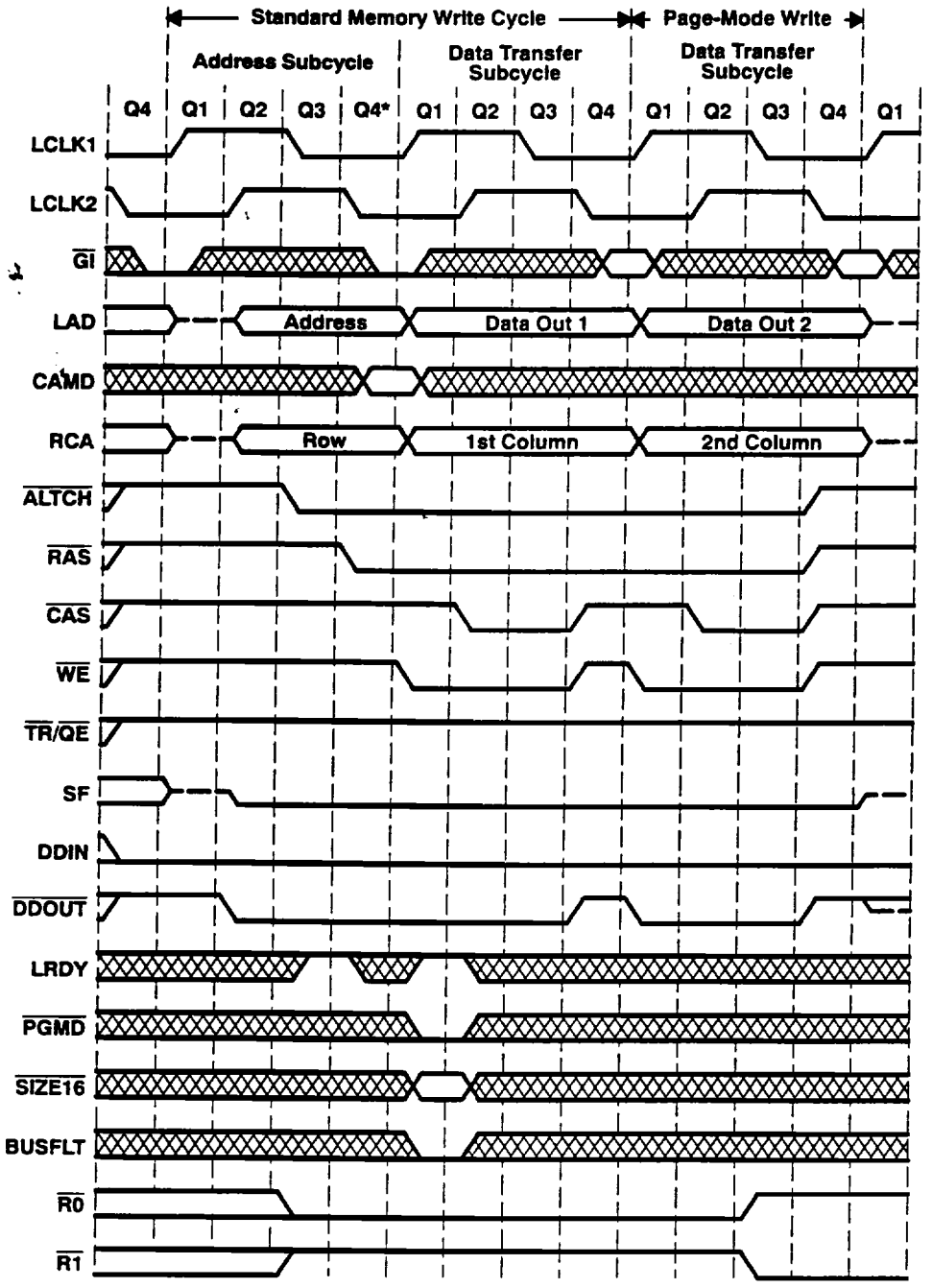
*See "clock stretch", page 18.

NOTES: A. LAD (TMS340X): Output to the LAD bus by the TMS340X.
LAD (memory): Output to the LAD bus by the memory.

B. Although they are not internally sampled, $\overline{\text{PGMD}}$ and $\overline{\text{SIZE16}}$ must be held at a valid level at the start of each Q2 until the LRDY is sampled high.

Figure 3. Local Memory Read Cycle Timing (Without Page Mode, With One Wait State)

LRDY low at the start of the first Q2 after $\overline{\text{RAS}}$ low indicates that the memory requires the addition of wait states. LRDY high at the next Q2 indicates the cycle may continue without inserting more wait states. $\overline{\text{PGMD}}$ high at the start of Q2 where LRDY is sampled high indicates that this memory does not support page-mode operation.

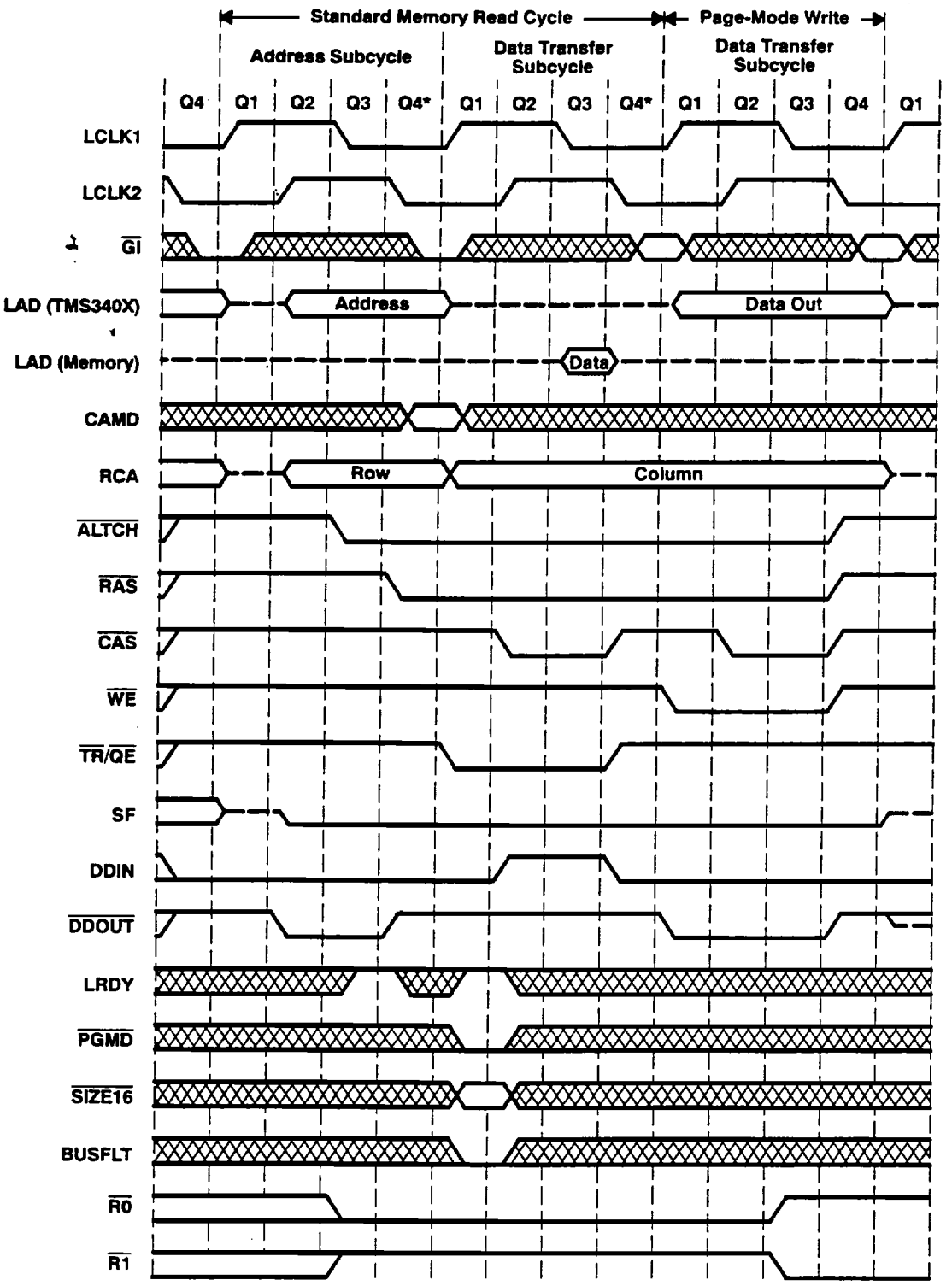


*See "clock stretch", page 18.
NOTE A: LRDY, PGMD, SIZE16, and BUSFLT are not sampled on subsequent page mode cycle accesses to 32-bit wide memory space.

Figure 4. Local Memory Write Cycle Timing (With Page Mode)

During the address output to the LAD bus by the TMS340X, the least significant four bits (LAD0-LAD3) contain a bus status code. PGMD low at the start of Q2 after RAS low indicates that this memory supports page-mode operation. LRDY high at the start of Q2 after RAS low indicates that the cycle can continue without inserting wait states.

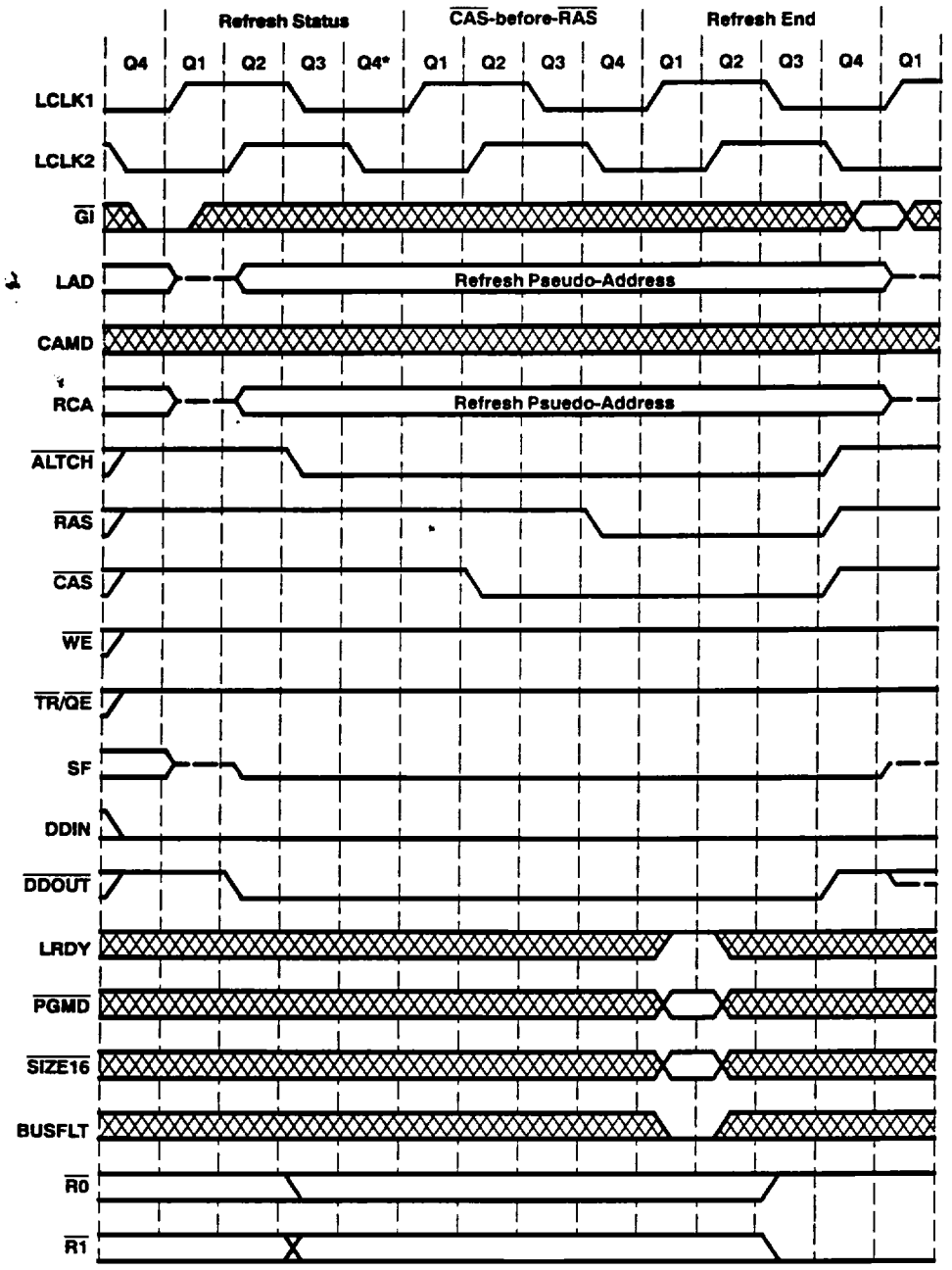
DDOUT remains low after the initial address output on LAD (during Q4 after RAS goes low), indicating that a memory write cycle is about to take place.



*See "clock stretch", page 18.

Figure 5. Local Memory Read/Modify/Write Cycle Timing

The read/modify/write cycle is used when inserting a field into memory that crosses byte boundaries. This cycle is actually performed as a read access followed by a page-mode-write cycle.



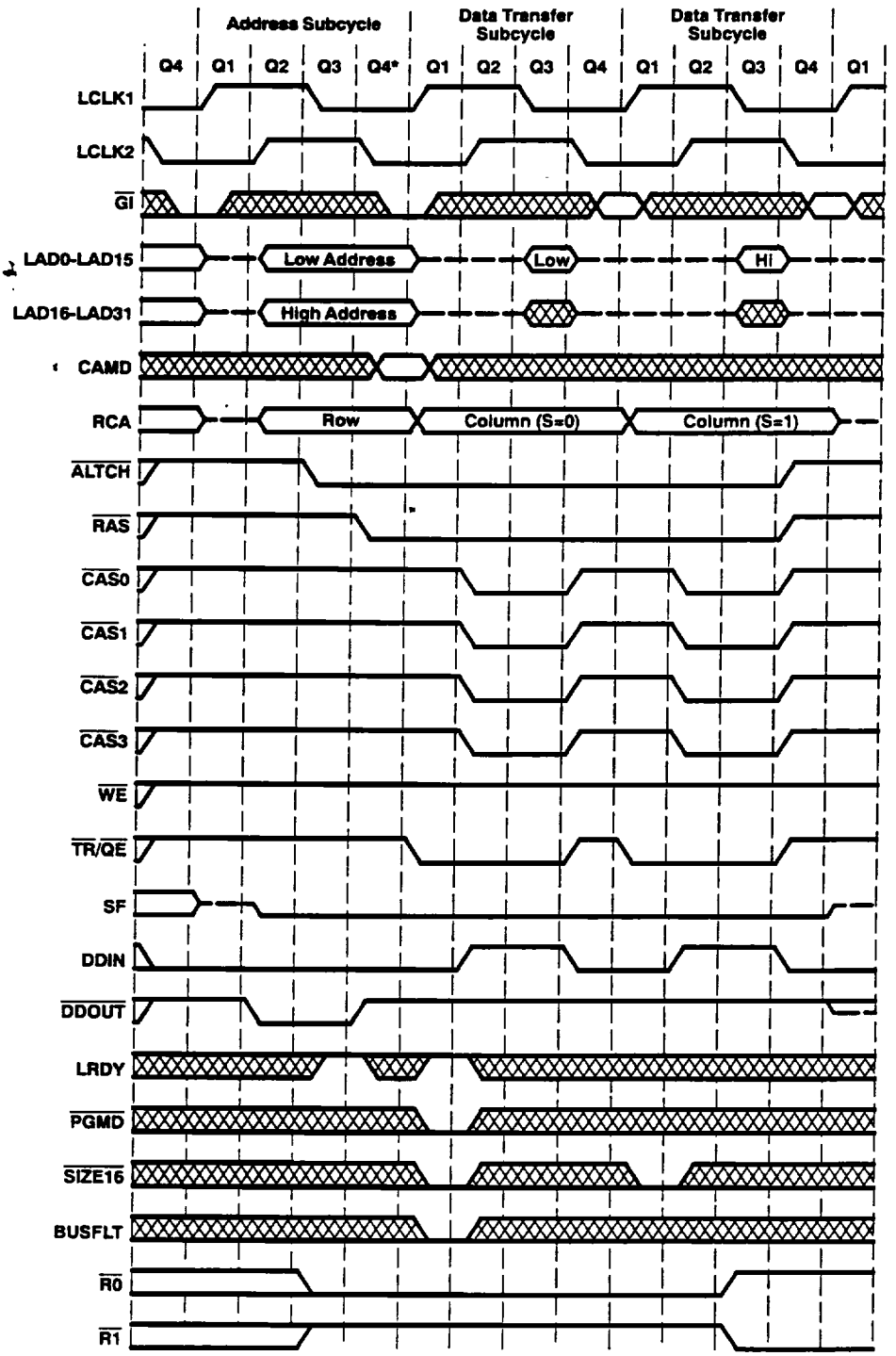
*See "clock stretch", page 18.

Figure 6. Refresh Cycle Timing

The refresh pseudo-address output to RCA0-RCA12 and LAD0-LAD31 comes from the 16-bit refresh address register (I/O register C000 01F0h) that is incremented after each refresh cycle. The 16 bits of address are placed on LAD16-LAD31; all other LAD bus lines will be zero. The logical addresses on RCA0-RCA12 corresponding to LAD16-LAD31 also output the address from the refresh address register.

Although $\overline{\text{PGMD}}$ and $\overline{\text{SIZE16}}$ are ignored during a refresh cycle, they should be held at valid levels. LRDY and BUSFLT are not sampled until the start of the first Q2 cycle after $\overline{\text{RAS}}$ has gone low.

If a refresh cycle is aborted due to a high-priority bus request (assuming LRDY is low at Q2 after $\overline{\text{RAS}}$ low), a bus fault, or an external retry, then the count of refreshes pending will not be decremented and the same pseudo-address is reissued when the refresh is restarted.

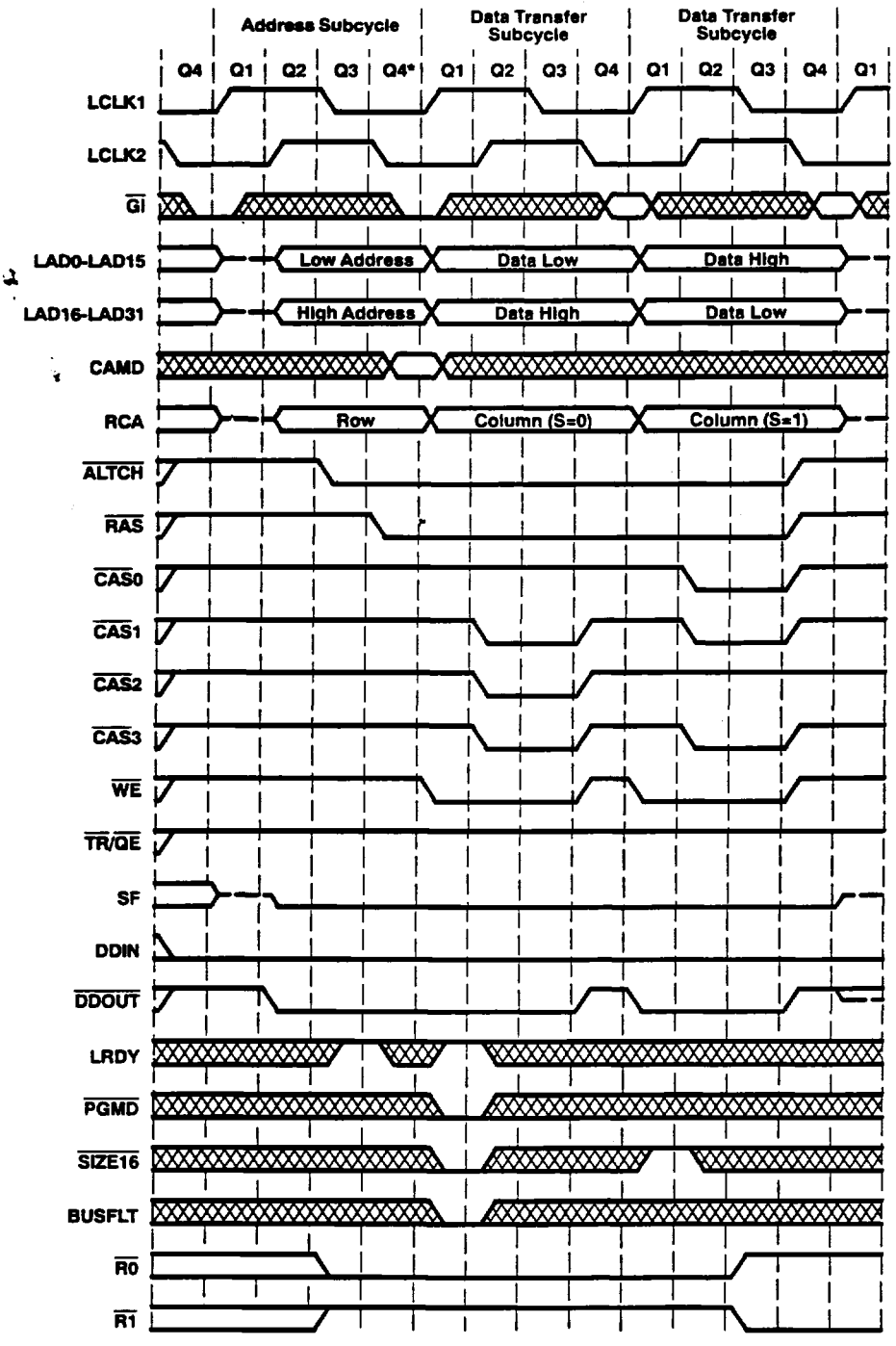


*See "clock stretch", page 18.

NOTE A: RCA0 may be used to determine accesses to odd or even words because it outputs the least significant bit of the word address during the column address time (except in 4M mode with CAMD = 1).

Figure 7. Dynamic Bus Sizing, Read Cycle

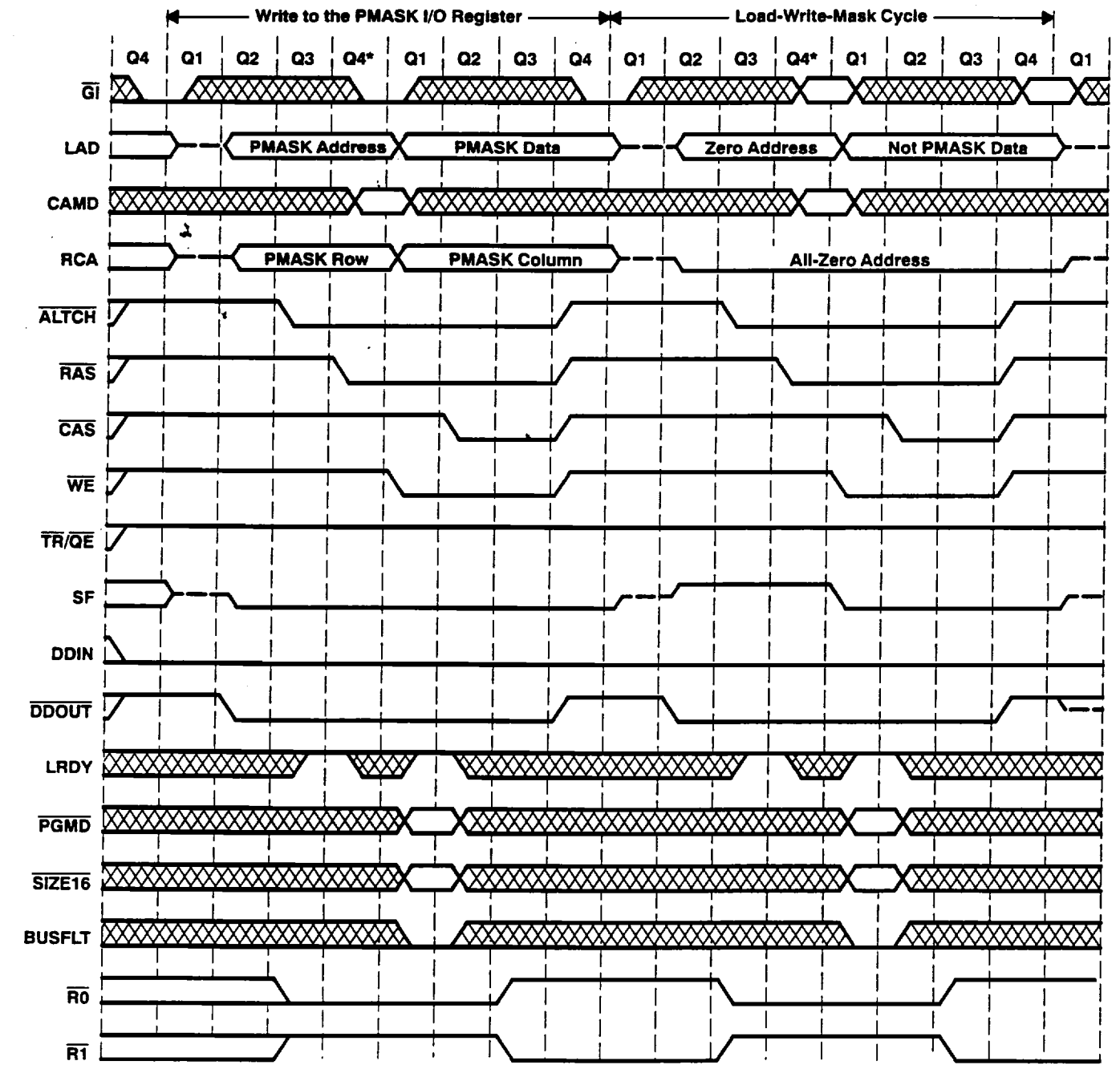
When $\overline{\text{SIZE16}}$ is selected low, the TMS340X performs a second cycle to read (or write) the remaining 16 bits of the word. Reads always access all 32 bits (all $\overline{\text{CAS}}$ strobes are active). Internally, the TMS340X latches both the high and the low words obtained on the first read cycle. The sense of $\overline{\text{SIZE16}}$ on the second (odd word) access is used to determine which half of the bus is to be sampled to replace the data word latched during the first cycle.



*See "clock stretch", page 18.

Figure 8. Dynamic Bus Sizing, Write Cycle

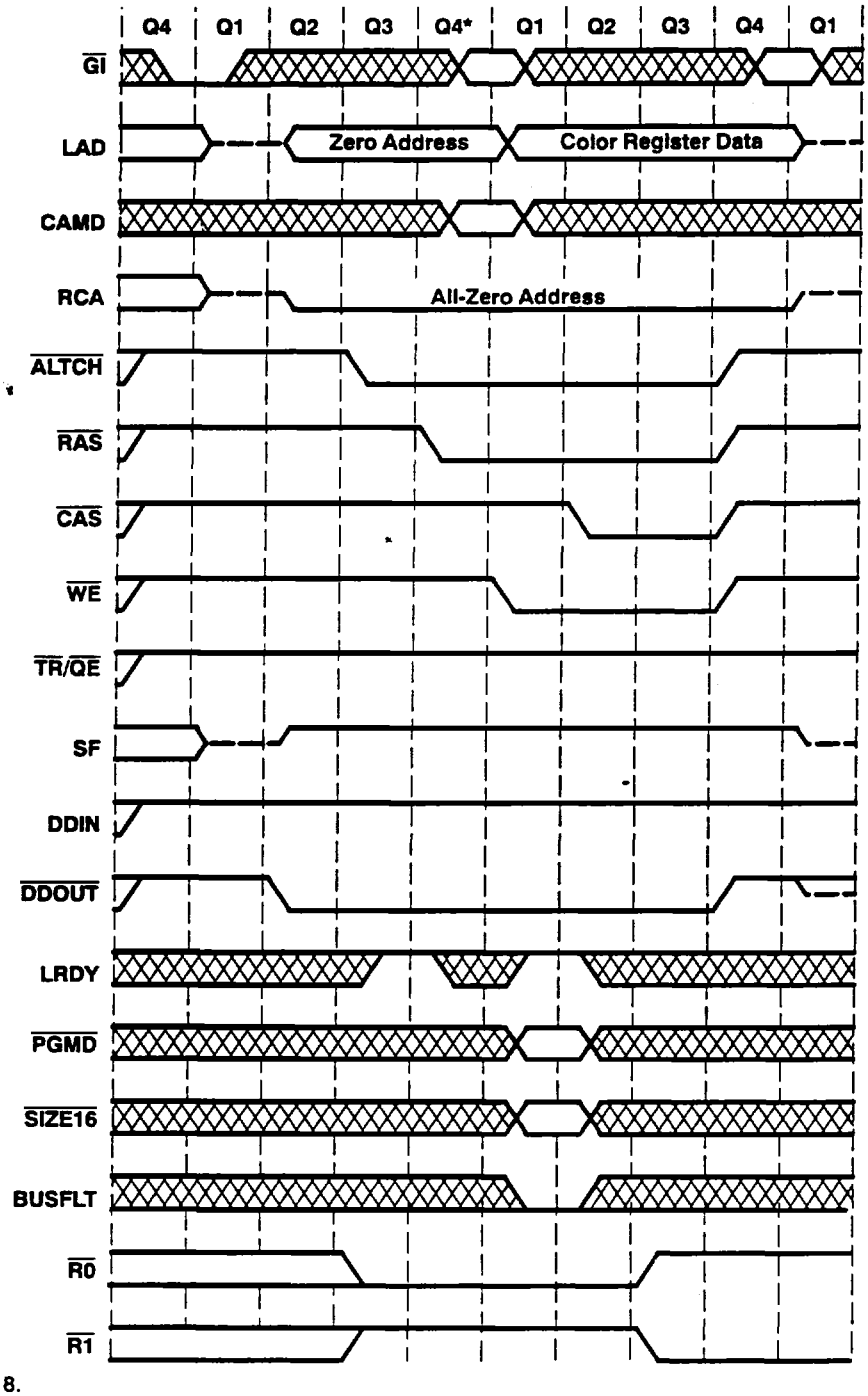
Write accesses to 16-bit memory are performed by swapping the data on upper and lower words of the LAD bus and exchanging data on $\overline{CAS0}$ and $\overline{CAS1}$ for data on $\overline{CAS2}$ and $\overline{CAS3}$, respectively. This is illustrated in the above example where the TMS340X is writing the upper 24 bits (LAD8-LAD31) of data to memory. During the first cycle, data is placed on LAD0-LAD31 as in a normal write. The sampling of $\overline{SIZE16}$ low during the first access indicates that this is 16-bit wide memory, so the TMS340X swaps data on the upper and lower halves of the LAD bus. Notice that during the first cycle $\overline{CAS0}$ is inactive (since this byte was not selected) and during the second cycle $\overline{CAS2}$ is inactive due to the exchange of $\overline{CAS0}$ for $\overline{CAS2}$ and $\overline{CAS1}$ for $\overline{CAS3}$.



*See "clock stretch", page 18.

Figure 9. Load Write Mask Cycle

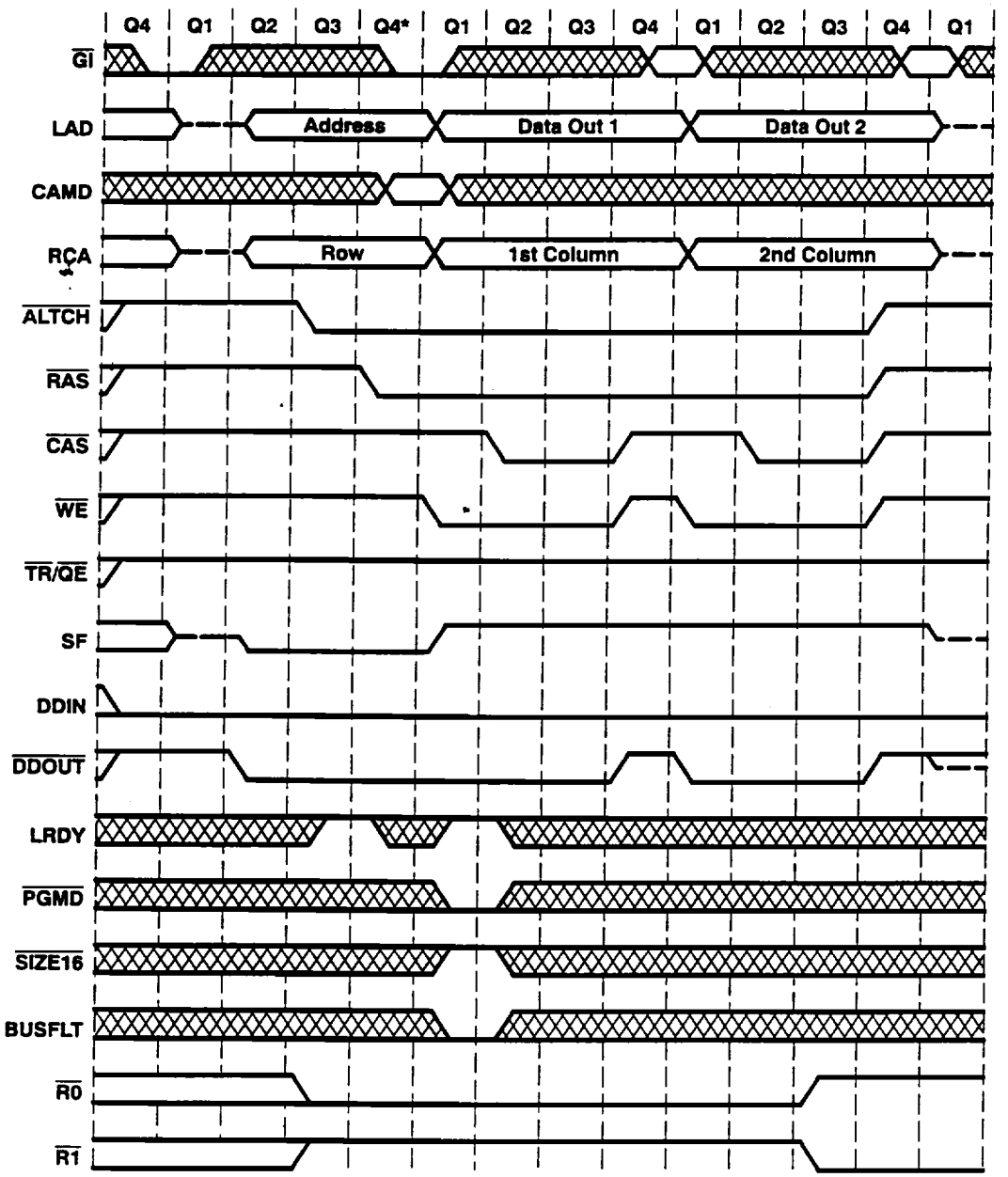
This special 1M-VRAM control cycle is executed when the VEN bit in the CONFIG I/O register is set and PMASKL and/or PMASKH are written. This cycle is indicated by $\overline{CAS}$, $\overline{WE}$, $\overline{TR/QE}$, and SF high at the falling edge of $\overline{RAS}$ and SF low at the falling edge of $\overline{CAS}$. As the plane mask is copied to the PMASK register(s), it is also output on the LAD bus to be written to a special register on the VRAM that is used in subsequent cycles requiring a write mask. During the address portion of the cycle, the status on LAD0-LAD3 indicates a write mask load is being performed (Status Code = 0110). Although CAMD, $\overline{PGMD}$, and $\overline{SIZE16}$ are ignored on this cycle, they should be held at valid levels as shown.



*See "clock stretch", page 18.

Figure 10. Load Color Latch Cycle

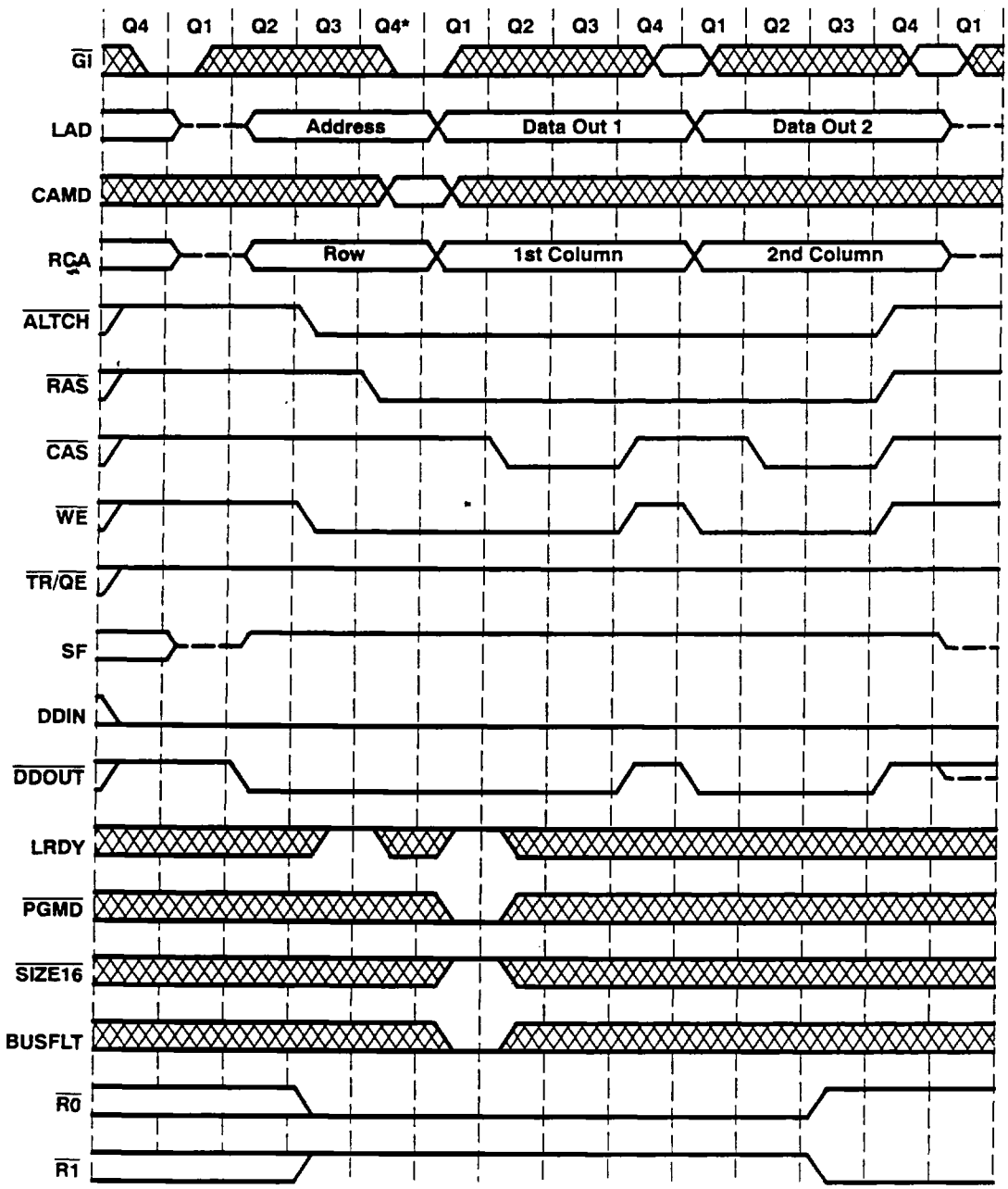
This special 1M-VRAM control cycle is generated by the VLCOL instruction and is indicated by $\overline{CAS}$, $\overline{WE}$, $\overline{TR/QE}$, and SF high at the falling edge of $\overline{RAS}$ and SF high at the falling edge of $\overline{CAS}$. The data in the COLOR1 register is output on LAD to be written to a special register on the VRAM that is used in subsequent cycles requiring a color latch. During the address portion of the cycle, the status on LAD0-LAD3 indicates a color mask load is being performed (Status Code = 0111). Although CAMD, $\overline{PGMD}$, and $\overline{SIZE16}$ are ignored on this cycle, they should be held at valid levels as shown.



*See "clock stretch", page 18.

Figure 11. Block Write Cycle Without Mask

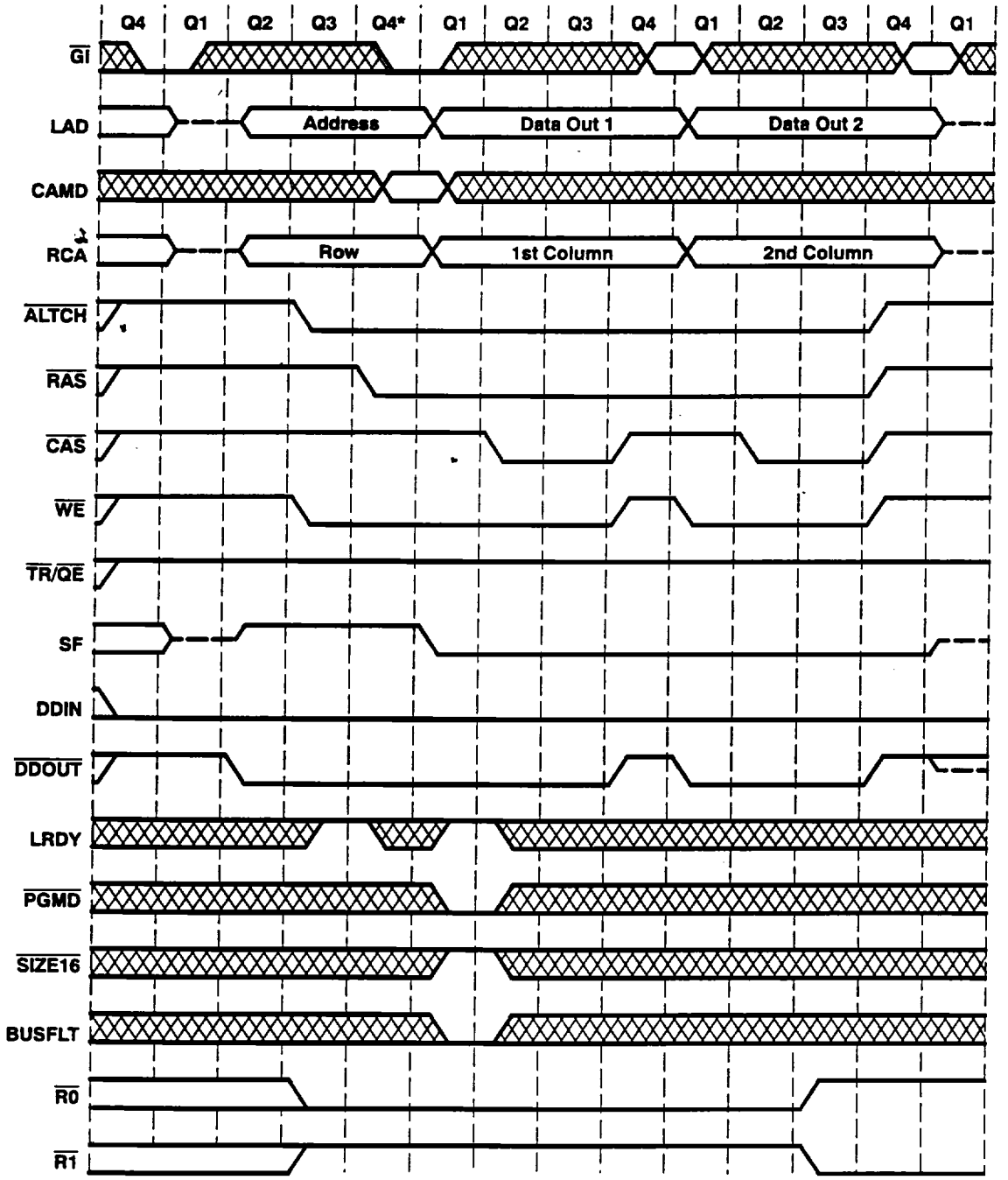
This special 1M-VRAM control cycle is performed when a VBLT or VFILL instruction is executed and PMASKL and PMASKH are set to zero. It is indicated by $\overline{CAS}$, $\overline{WE}$, $\overline{TR/QE}$ high and SF low at the falling edge of $\overline{RAS}$, and by SF high at the falling edge of $\overline{CAS}$. The data on LAD is used as an address mask, and the data stored in the color latch is written to the VRAM. The address selects chosen by the two least significant bits of the column addresses within the VRAM are replaced with the four DQ bits latched on the falling edge of $\overline{CAS}$. A logic 1 on each bit enables that nibble to be written, while a logic 0 disables the write from occurring. This cycle allows up to 16 bits to be written into each VRAM (four adjacent nibbles, each set to the value in the color latch) for a total of 128 bits. During the address portion of the cycle, the status on LAD0-LAD3 indicates a block write is being performed (Status Code = 1110). $\overline{SIZE16}$ can be used with this cycle, but external multiplex logic is required to map the data correctly to appropriate memories.



*See "clock stretch", page 18.

Figure 12. Block Write Cycle With Mask

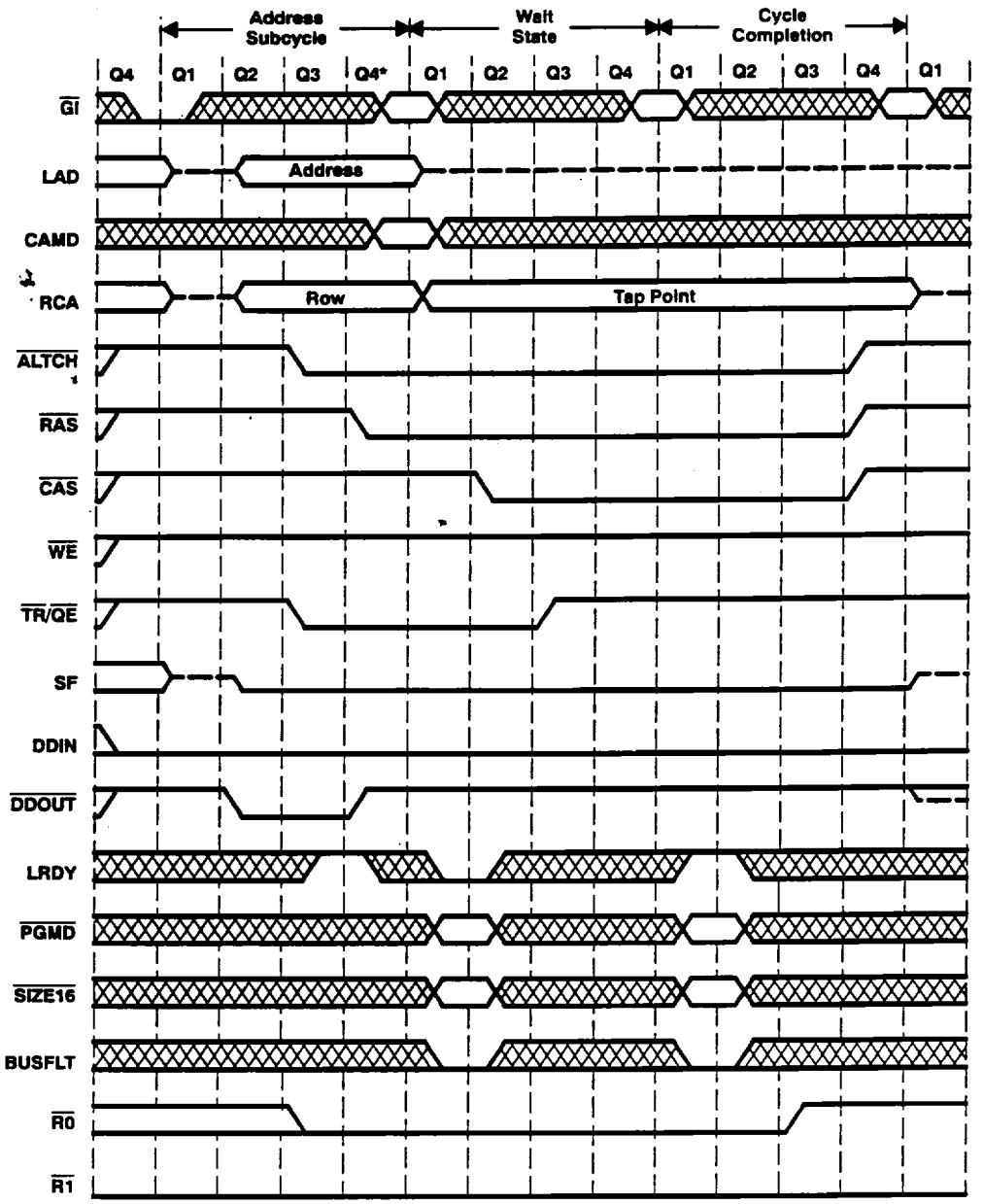
This special 1M-VRAM control cycle is performed when a VBLT or VFILL instruction is executed and PMASKL and PMASKH are set to nonzero. It is indicated by $\overline{CAS}$, $\overline{TR/QE}$, and SF high and $\overline{WE}$ low at the falling edge of $\overline{RAS}$, and by SF high at the falling edge of $\overline{CAS}$. The data on LAD is used as an address mask, and the data stored in the color latch is written to the VRAM, just as in block write without mask, except that the data in the write mask is used to enable the bits from the color latch that are written to memory. This cycle allows up to 16 bits to be written into each VRAM (four adjacent nibbles, each set to the value in the color latch as enabled by the write mask) for a total of 128 bits. During the address portion of the cycle, the status on LAD0-LAD3 indicates a block write is being performed (Status Code = 1110). $\overline{SIZE16}$ can be used with this cycle, but external multiplex logic is required to map the data correctly to appropriate memories.



*See "clock stretch", page 18.

Figure 13. Write Using Mask Cycle

This special 1M-VRAM control cycle is performed when the PMASKL and PMASKH registers are set to nonzero, the CST bit in DPYCTL is cleared, the VEN bit in CONFIG is set, and the byte-aligned pixel write instruction is executed. This cycle is indicated by $\overline{CAS}$, $\overline{TR/QE}$, and SF high and $\overline{WE}$ low at the falling edge of $\overline{RAS}$, and by SF low at the falling edge of $\overline{CAS}$. The data on LAD is written to memory just as a normal DRAM write, except that data in the write mask is used to enable the DQ bits that are written to memory. During the address portion of the cycle, the status on LAD0-LAD3 indicates that pixel operation is being performed (Status Code = 1101).



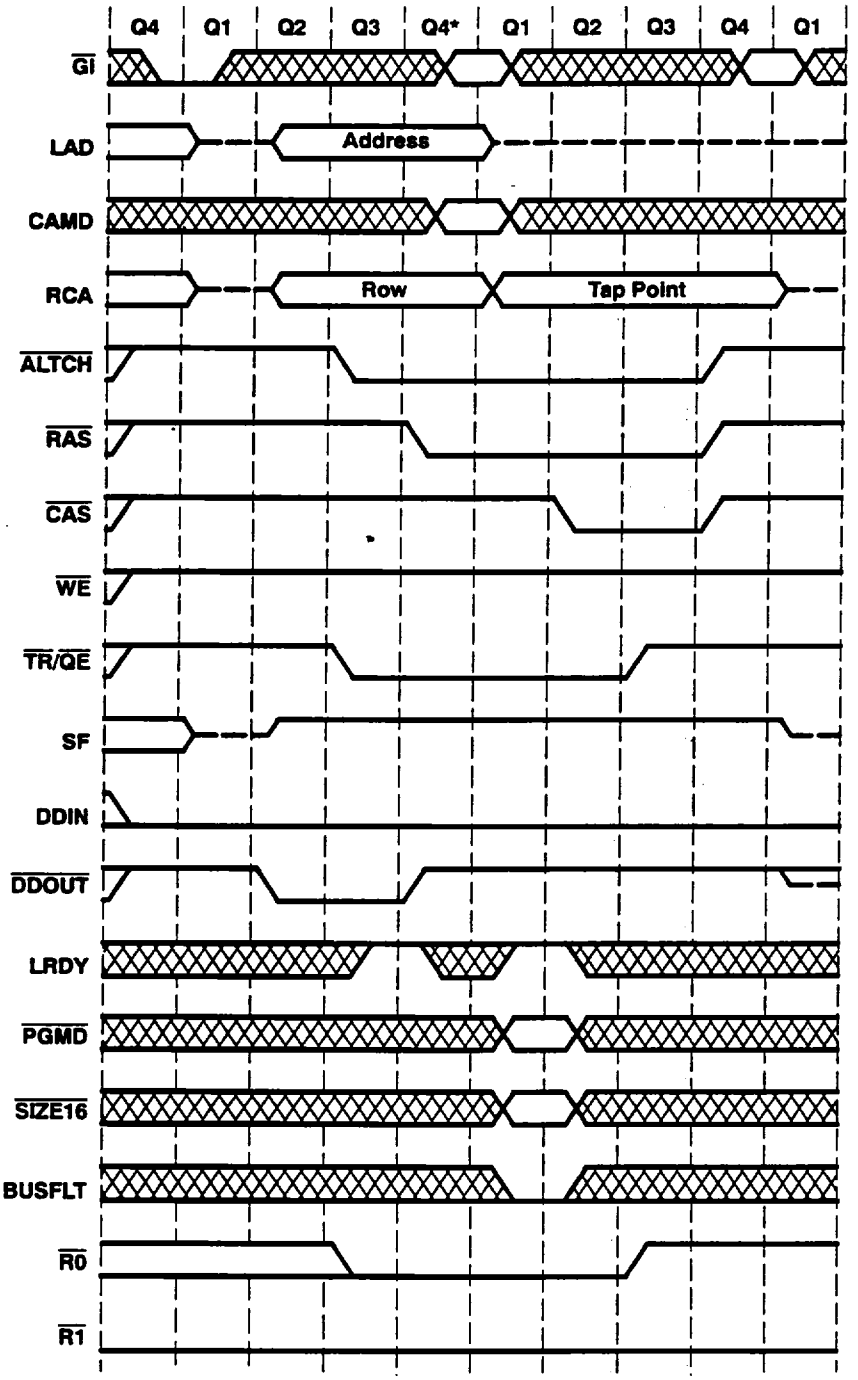
*See "clock stretch", page 18.

Figure 14. Memory to Serial Data Register Cycle (VRAM Read Transfer)

This VRAM cycle is issued in any of three ways:

1. Pixel operation instruction with the CST bit in DPYCTL set.
2. Horizontal blank reload cycle requested by the video control logic with the VCE bit in DPYCTL cleared.
3. Video timeout due to SCOUNT match with the value in MLRNXT and the VCE and SSV bits in DPYCTL cleared.

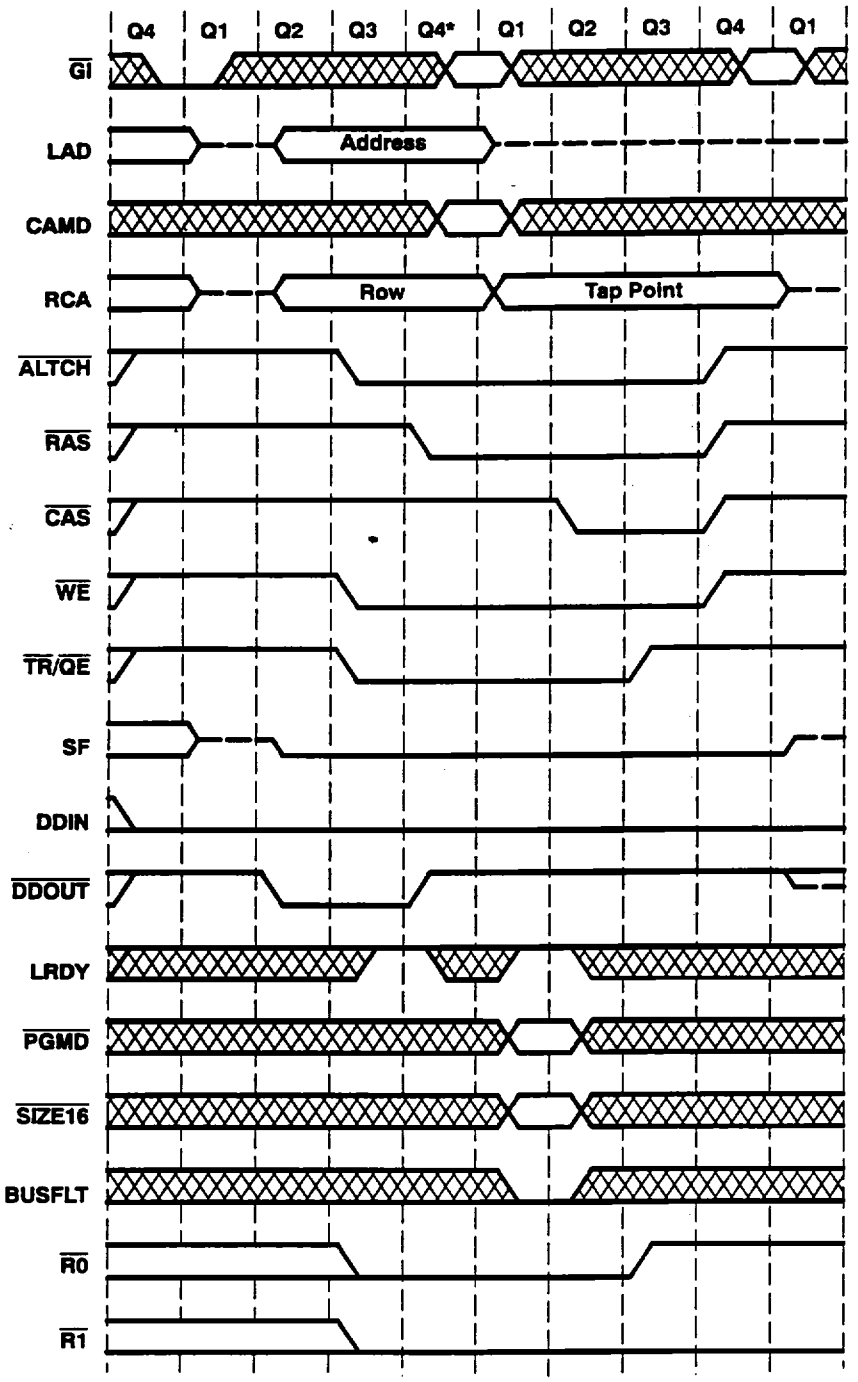
This cycle is indicated by $\overline{TR}/\overline{QE}$ and SF low and $\overline{CAS}$ and $\overline{WE}$ high at the time $\overline{RAS}$ goes low. The timing of the low-to-high transition of $\overline{TR}/\overline{QE}$ is dependent upon the timing of SCLK when doing a mid-line reload cycle. During the address portion of the cycle the status on LAD0-LAD3 indicates either a video-initiated VRAM memory-to-register transfer (Status Code = 0100) or a CPU-initiated VRAM memory-to-register transfer (Status Code = 0101).



*See "clock stretch", page 18.

Figure 15. Memory to Split Serial Data Register Cycle (VRAM Split Register Read Transfer)

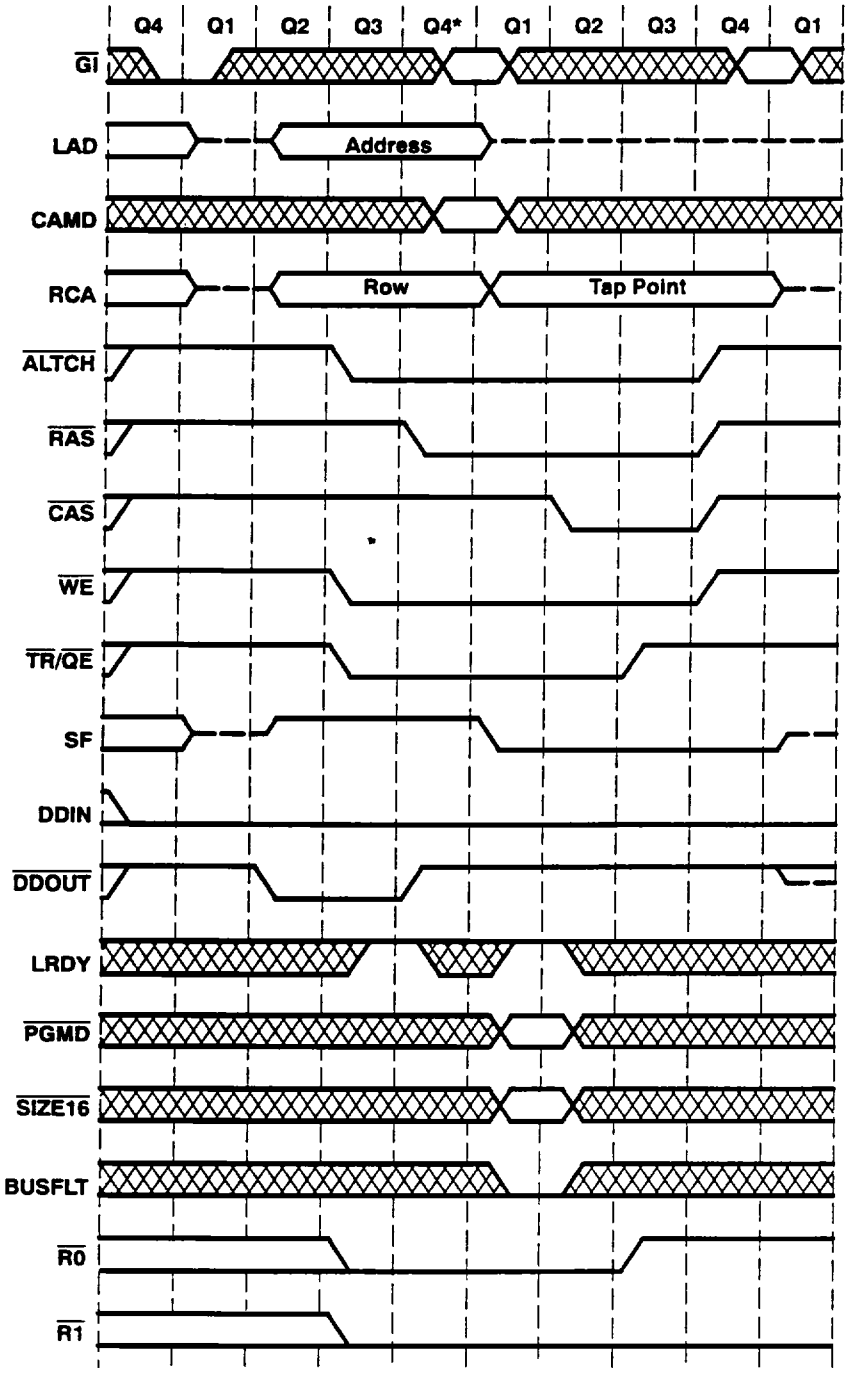
This VRAM cycle is performed when a video timeout due to a match of the MLRNXT register occurs, the VCE bit in DPYCTL is cleared, and the SSV bit in DPYCTL is set. This cycle is indicated by $\overline{TR}/\overline{QE}$ low and $\overline{CAS}$, SF, and $\overline{WE}$ high at the time $\overline{RAS}$ goes low. The timing of the low-to-high transition of $\overline{TR}/\overline{QE}$ is not dependent upon the timing of SCLK because there is not as great a timing constraint to position the cycle as in mid-line reload. During the address portion of the cycle, the status on LAD0-LAD3 indicates a video-initiated VRAM memory-to-register transfer (Status Code = 0100). Although $\overline{PGMD}$ and $\overline{SIZE16}$ are ignored on this cycle, they should be held at valid levels as shown.



*See "clock stretch", page 18.

Figure 16. Serial Data Register-to-Memory Cycle (VRAM Write Transfer, Pseudo-Write Transfer)

This VRAM cycle is performed when a horizontal blank reload is requested by the video control logic, and the VCE bit and the SRE bit in DPYCTL are both set. This cycle is indicated by $\overline{TR}/\overline{QE}$, $\overline{WE}$ and SF low and $\overline{CAS}$ high at the time $\overline{RAS}$ goes low. The $\overline{SOE}$ pin of the VRAMs is used to select between write transfer and pseudo-write transfer cycles. ($\overline{SOE}$ must be generated by logic external to the TMS340X). During the address portion of the cycle, the status on LAD0-LAD3 indicates that a video-initiated VRAM register-to-memory transfer (Status Code = 0100) is being performed. Although $\overline{PGMD}$ and $\overline{SIZE16}$ are ignored on this cycle, they should be held at valid levels as shown.



*See "clock stretch", page 18.

Figure 17. Serial Data Register-to-Memory Cycle (VRAM Alternate Write Transfer)

This VRAM cycle is performed when a pixel write instruction is executed with the CST bit in DPYCTL set. This cycle is indicated by $\overline{TR/QE}$, and $\overline{WE}$ low, and SF and $\overline{CAS}$ high at the time $\overline{RAS}$ goes low. This cycle does not require the use of the $\overline{SOE}$ pin of the VRAM and does not affect the status of the serial I/O pins. During the address portion of the cycle, the status on LAD0-LAD3 indicates that a CPU-initiated VRAM register-to-memory transfer (Status Code = 0101) is being performed. Although $\overline{PGMD}$ and $\overline{SIZE16}$ are ignored on this cycle, they should be held at valid levels as shown.

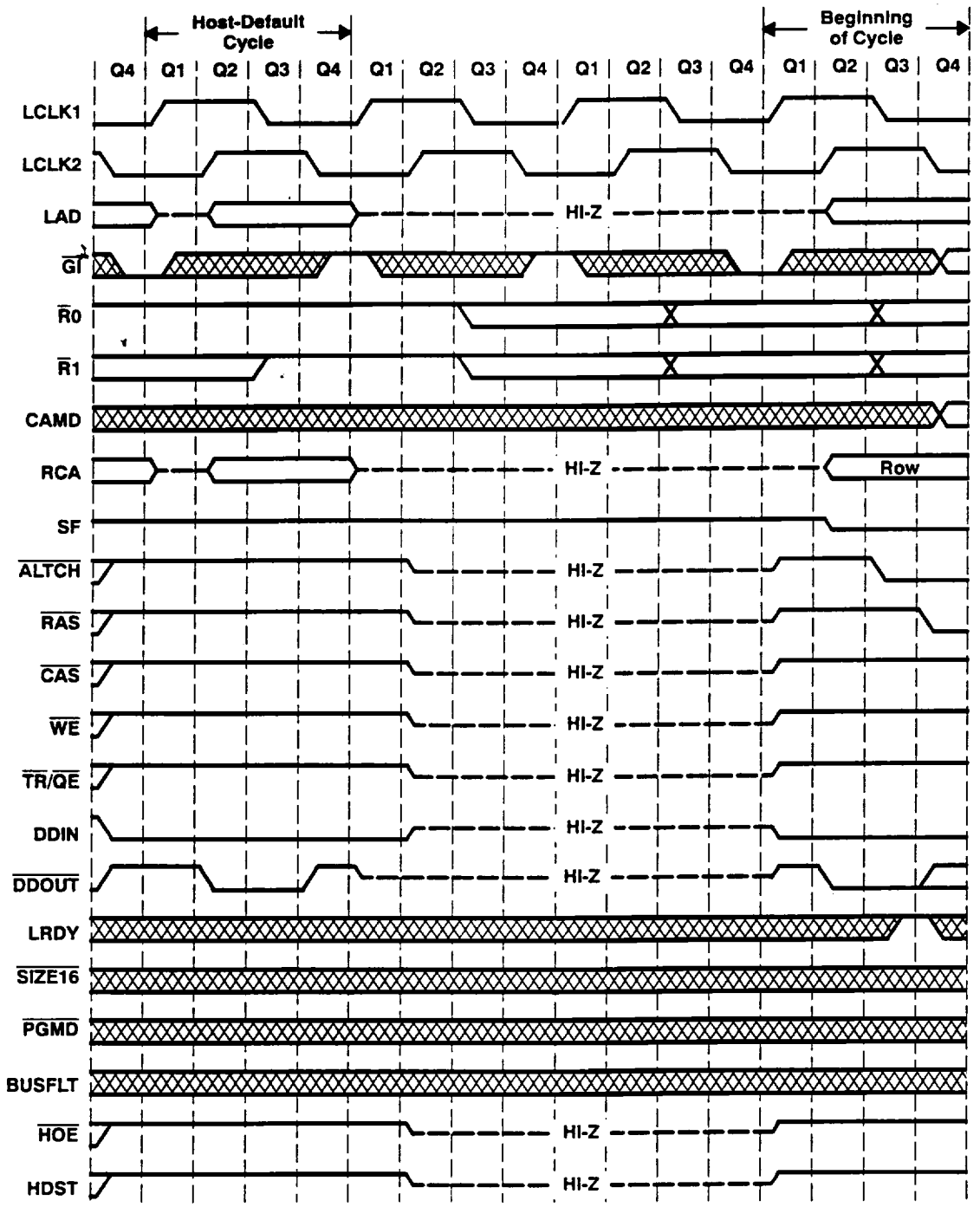
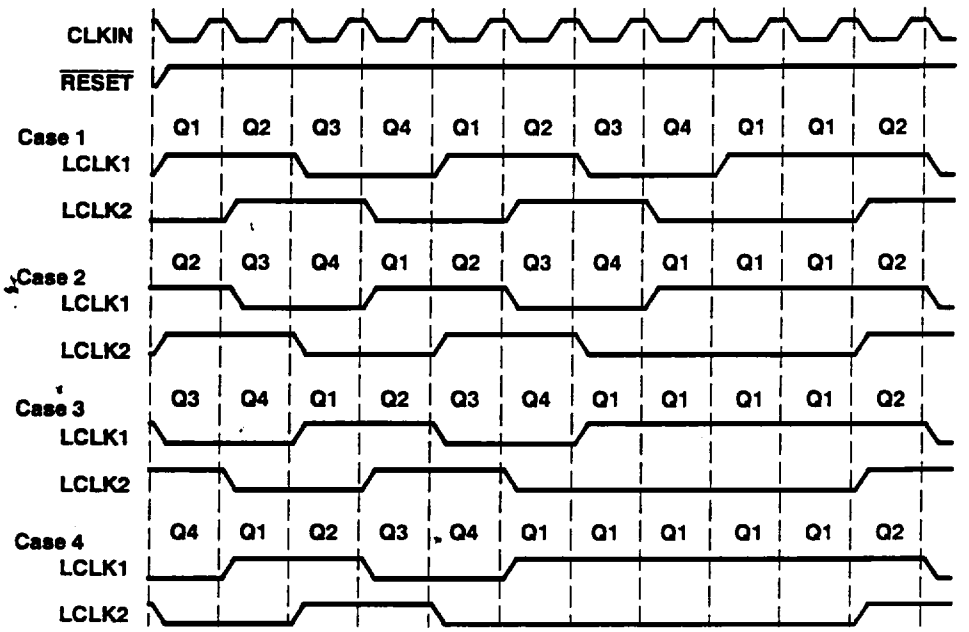


Figure 18. Multiprocessor Interface Timing (High-Impedance Signals))

Transition points are shown for $\overline{R}0$ and $\overline{R}1$ to indicate where they occur relative to the other signals.

This example indicates that the TMS340X has control of the bus, yields control, and then regains control. The TMS340X regains bus mastership as soon as its $\overline{GI}$ pin is driven active low. R0 and R1 could be outputting any of the codes with the exception of the access-termination code. The bus arbitration logic must control the timing of $\overline{GI}$ to all of the processors requiring the bus.

It is recommended that TMS340X clock stretch not be used in multiprocessor systems.

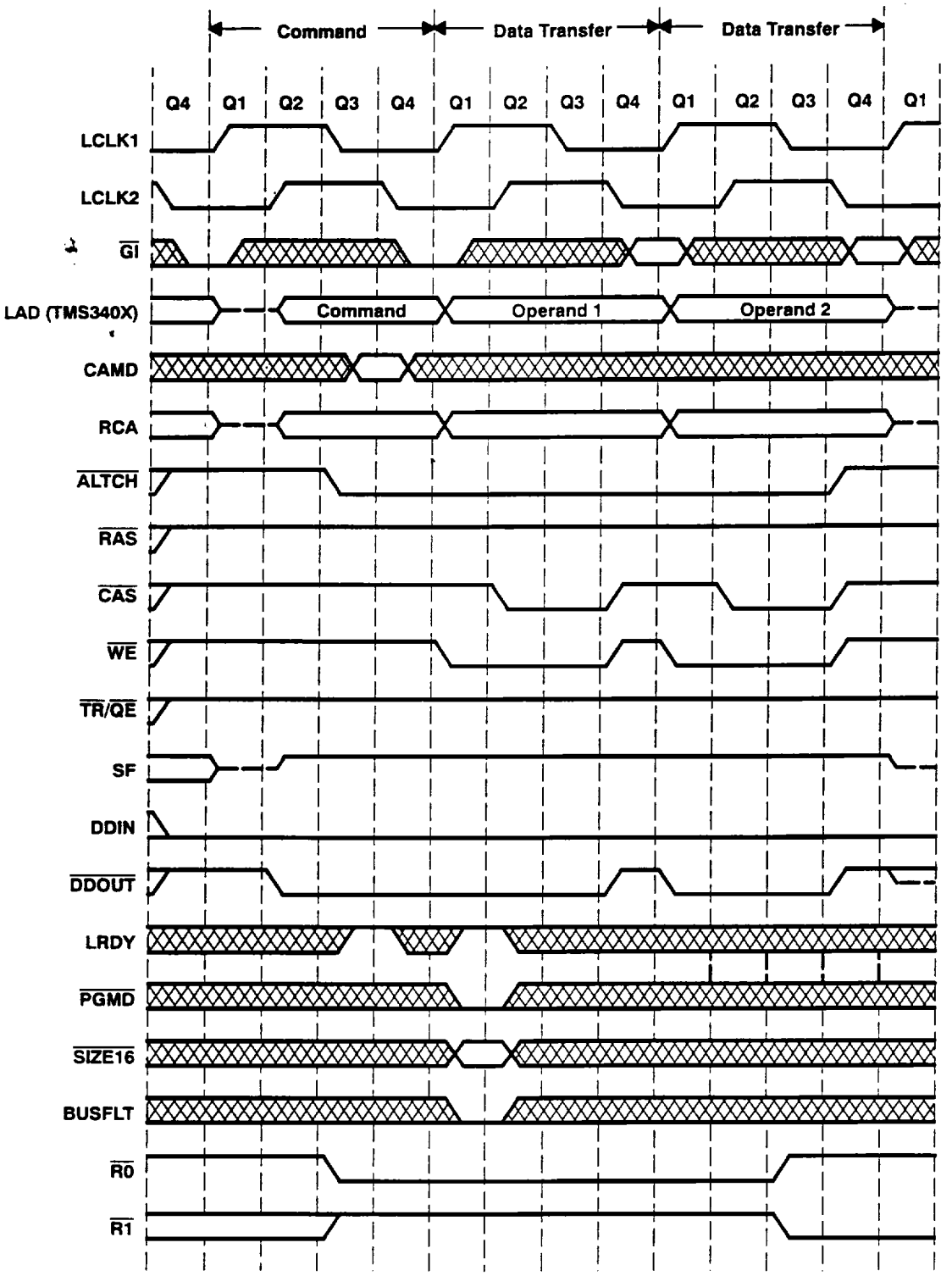


NOTE A: No timing dependences of LCLK1 and LCLK2 relative to CLKIN or $\overline{\text{RESET}}$ are to be implied from this figure.

Figure 19. Synchronization of Multiple TMS340Xs

Although $\overline{\text{RESET}}$ is not *normally* required to be synchronous to CLKIN, in order to facilitate synchronization of multiple TMS340Xs in a system, the rising edge of $\overline{\text{RESET}}$ must meet the setup and hold requirements to CLKIN so that all GSPs are certain to respond to the $\overline{\text{RESET}}$ on the same quarter cycle. The four possible conditions for the state of the TMS340X at the time $\overline{\text{RESET}}$ goes high are shown above. Quarter cycle 1 is extended accordingly to provide synchronization of the GSPs. All TMS340Xs to be synchronized must share a common CLKIN and $\overline{\text{RESET}}$. Within 10 CLKIN cycles after $\overline{\text{RESET}}$ goes high, all GSPs will be synchronized to the same Q cycle through the extension of Q1 cycles.

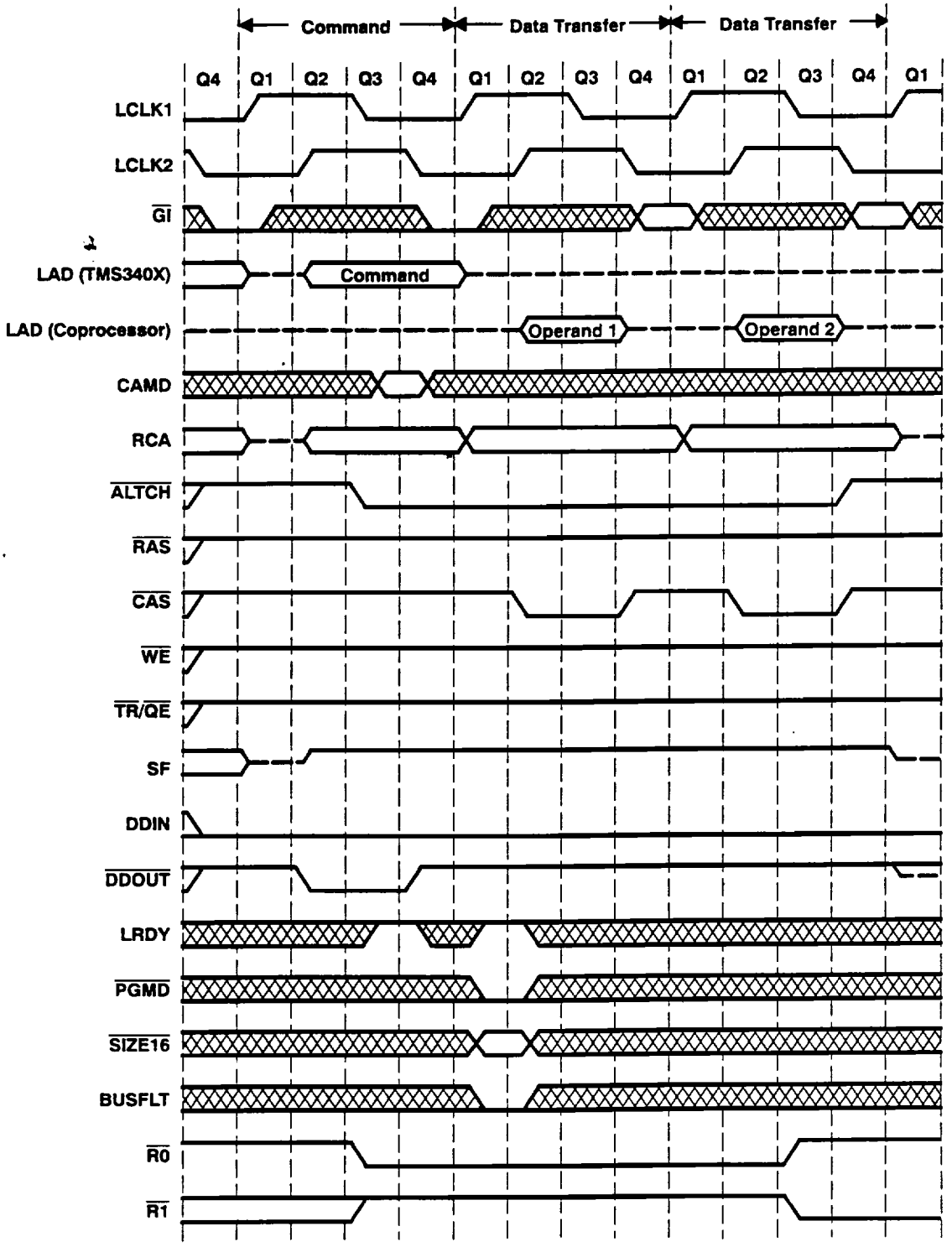
It is recommended that TMS340X stretch mode should not be used in multiprocessor systems.



NOTE A: LAD (TMS340X): Output to the LAD bus by the TMS340X
Command: Coprocessor ID, instruction and status code present on LAD
Operand n: Data to or from the coprocessor

Figure 20. Transfer TMS340X Register(s) to Coprocessor (One or Two 32-Bit Values)

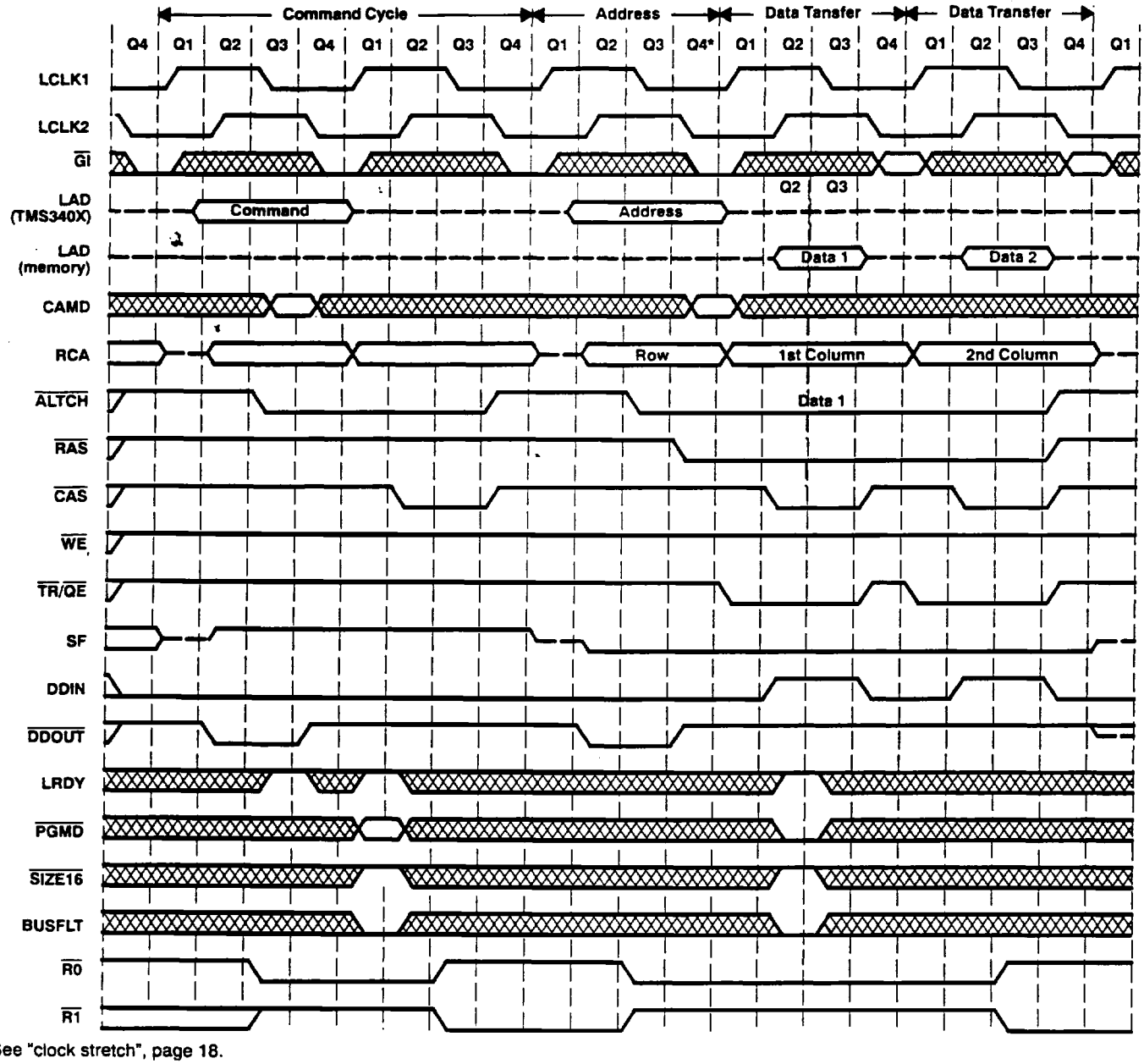
This timing example is like a memory write cycle, except that $\overline{RAS}$ and SF are high.



NOTE A: LAD (TMS340X): Output to the LAD bus by the TMS340X
LAD (coprocessor): Output to the LAD bus by the coprocessor
Command: Coprocessor ID, instruction and status code present on LAD
Operand n: Data to or from the coprocessor

Figure 21. Transfer Coprocessor Register to TMS340X (One or Two 32-Bit Values)

This timing example is like a memory write cycle, except that $\overline{RAS}$ and SF are high.



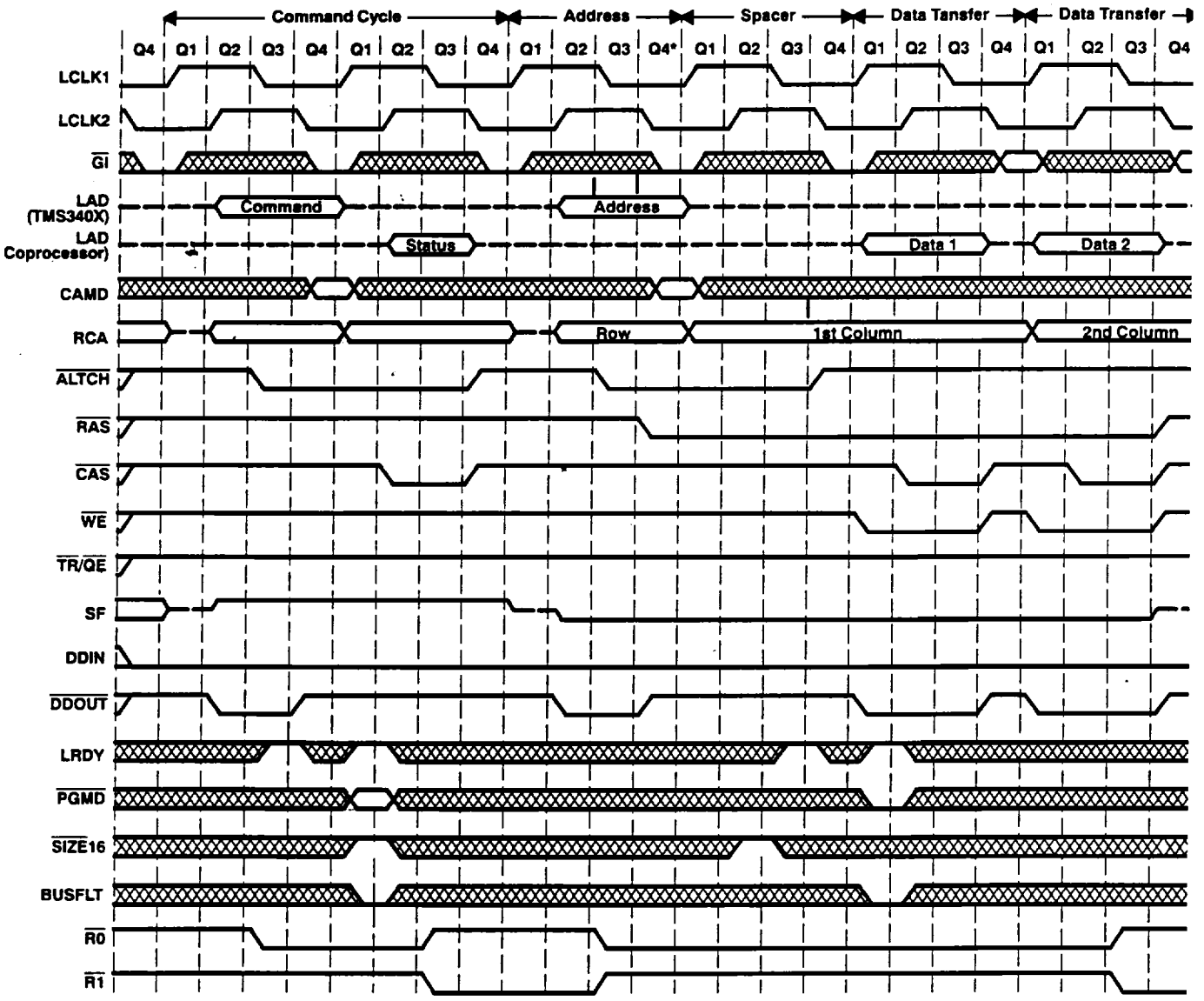
*See "clock stretch", page 18.

NOTES: A: LAD (TMS340X): Output to the LAD bus by the TMS340X
LAD (memory): Output to the LAD bus by the memory
Command: Coprocessor ID, instruction and status code present on LAD
Address: Memory address for the data transfer, with coprocessor status code
Data n: Data to or from the coprocessor (number of values transferred depends on a value in a register or count in the instruction)

B. All coprocessor cycles are implemented as 32-bit operations and therefore $\overline{\text{SIZE16}}$ should be high during these cycles.

Figure 22. Transfer Memory to Coprocessor Register(s)

Data transfer from memory to a coprocessor requires an initialization cycle to inform the coprocessor what is to be transferred and then a memory cycle to perform the actual transfer. The coprocessor may place status information on the LAD bus during the initialization cycle for the TMS340X. Two types of memory to coprocessor instructions are supported: one provides a count (from 1 to 32) of data to be moved in the instruction, the other specifies a register in the TMS340X to be used for the count. Both instructions specify a register to be used as an index into memory. The index may be post-incremented or pre-decremented on each transfer cycle.



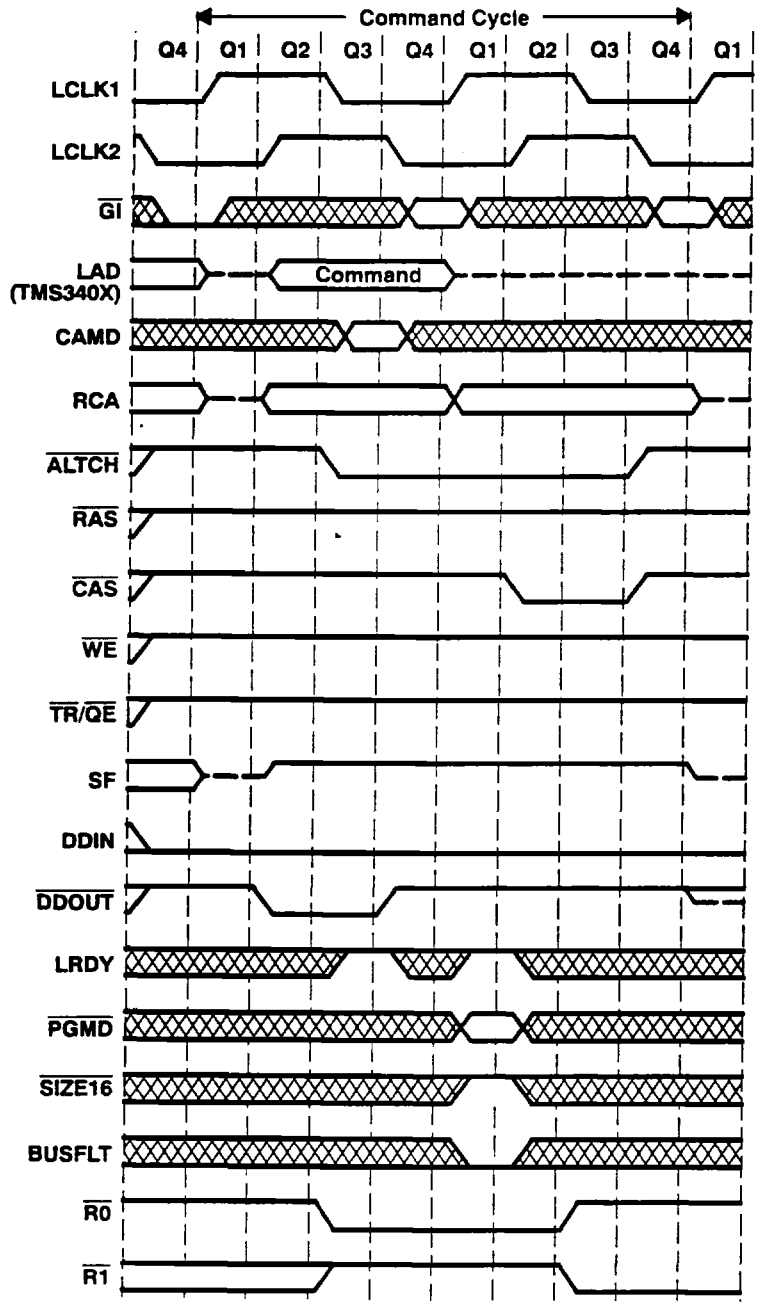
*See "clock stretch", page 18.

NOTES: A. All coprocessor cycles are implemented as 32-bit operations and therefore SIZE16 should be high during these cycles.

B. LAD (TMS340X): Output to the LAD bus by the TMS340X
LAD (coprocessor): Output to the LAD bus by the coprocessor
Command: Coprocessor ID, instruction and status code present on LAD
Address: Memory address for the data transfer, with coprocessor status code
Data n: Data from the coprocessor (number of values transferred depends on a count in the instruction)
Status: Optional coprocessor status register output to LAD bus

Figure 23. Transfer Coprocessor Register(s) to Memory (ALTCH High During Data Transfer)

Data transfer from a coprocessor to memory requires an initialization cycle to inform the coprocessor what is to be transferred and then a memory cycle to perform the actual transfer. The coprocessor may place status information on the LAD bus during the initialization cycle for the TMS340X. The memory cycle includes a dead cycle to enable the TMS340X to take the LAD bus drivers to a high-impedance state before the coprocessor activates its LAD bus drivers to the memory. Two types of memory-to-coprocessor instructions are supported. Both provide a count (from 1 to 32) of data to be moved in the instruction. Both also specify a register to be used as an index into memory. One uses this index register with a post-increment and the other uses it with a pre-decrement after each transfer cycle.



NOTES: A. All coprocessor cycles are implemented as 32-bit operations and therefore SIZE16 should be high during these cycles.
B. LAD (TMS340X): Output to the LAD bus by the TMS340X
command: Coprocessor ID, instruction and status code present on LAD
C. Although the coprocessor internal command never requires the use of page mode cycles, PGMD should be held at a valid level during the start of Q2 after ALTCH has gone low.

Figure 24. Coprocessor Internal Operation Command Cycle

This timing example is like a memory write cycle, except that RAS and SF are high.
A coprocessor internal command assumes no transfer of operands or results, but causes the coprocessor to execute some internal function. The coprocessor may place status information on the LAD bus during the cycle for the TMS340X.

absolute maximum ratings over operating free-air temperature range (see Note 1)†

Maximum supply voltage, V_{CC} ..... 7 V
Input voltage range ..... – 0.3 V to 7 V
Off-state output voltage range ..... – 2 V to 7 V
Operating free-air termperature range ..... 0°C to 70°C
Storage temperature range ..... – 10°C to 150°C

† Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: Voltage values are with respect to V_{SS}.

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage‡	0	0	0	V
V_{IL}	High-level output current			400	μA
I_{OL}	Low-level output current			2	mA
T_A	Operating free-air temperature	0		70	°C

‡ Take care to provide a minimum inductance path between the V_{SS} pins and system ground in order to minimize noise on V_{SS}.

DC electrical characteristics over full ranges of recommended operating conditions

PARAMETER		TEST CONDITIONS	MIN	TYP§	MAX	UNIT
V_{OH}	High-level output voltage	V_{CC} = MIN, I_{OH} = MAX	2.6			V
V_{OL}	Low-level output voltage	V_{CC} = MAX, I_{OL} = MIN			0.6	V
		V_{CC} = MAX, V_O = 2.8 V			20	
		V_{CC} = MAX, V_O = 0.6 V			–20	
I_I	Input current (All input pins except EMU0-EMU2¶)	V_I = V_{SS} to V_{CC}			±20	μA
I_{CC}	Supply current	V_{CC} = MAX			250	mA
C_I	Input capacitance			10		pF
Co	Output capacitance			10		pF

§ All typical values are at V_{CC} = 5 V, T_A = 25° C.

¶ EMU0 – EMU2 will not be connected in a typical configuration. Nominal pullup current will be 500 μA.

NOTE 2: HDST and $\overline{HOE}$ (output pins) have internal pullup resistors that allow high logic levels to be maintained when the TMS340X is not actually driving these pins.

PARAMETER MEASUREMENT INFORMATION

signal transition levels

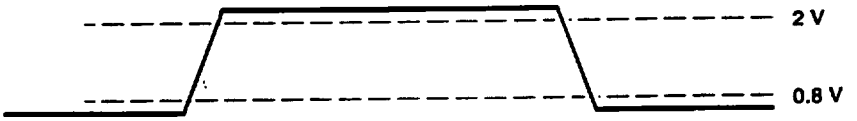


Figure 25. TTL-Level Inputs

For a high-to-low transition on a TTL-compatible input signal, the level at which the input is said to be no longer high is 2 V, and the level at which the input is said to be low is 0.8 V. For a low-to-high transition, the level at which the input is said to be no longer low is 0.8 V, and the level at which the input is said to be high is 2 V.

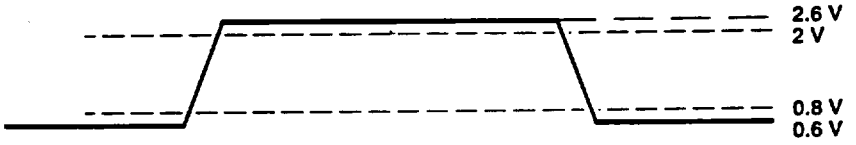


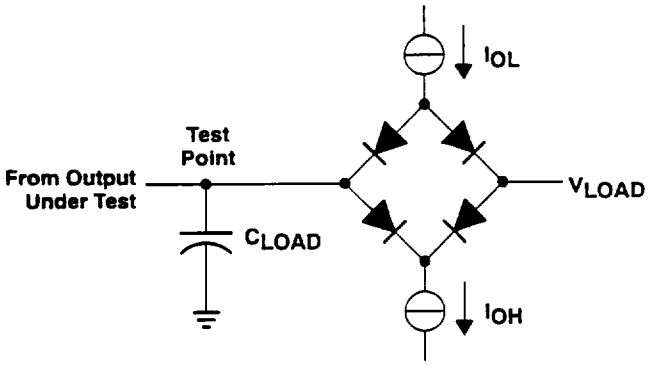
Figure 26. TTL-Level Outputs

TTL-level outputs are driven to a minimum logic-high level of 2.6 V and to a maximum logic-low level of 0.6 V. For a high-to-low transition on a TTL-compatible output signal, the level at which the output is said to be no longer high is 2 V, and the level at which the output is said to be low is 0.8 V. For a low-to-high transition, the level at which the output is said to be no longer low is 0.8 V, and the level at which the output is said to be high is 2 V.

Timing parameters 12a and 13a do not follow these conventions. They are measured at a 1.5-V level.

test measurement

The test load circuit shown in Figure 27 represents the programmable load of the tester pin electronics, which are used to verify timing parameters of TMS340X output signals.



Where: I_{OL} = 2 mA (all outputs)
I_{OH} = 400 μA (all outputs)
V_{LOAD} = 1.5 V
C_{LOAD} = 65 pF typical load circuit distributed capacitance

Figure 27. Test Load Circuit

PARAMETER MEASUREMENT INFORMATION

timing parameter symbology

Timing parameter symbols used herein were created in accordance with JEDEC Standard 100. In order to shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

AD	LAD0-LAD31 and RCA0-RCA12
AL	ALTCH
BC	Any of the bus control input signals (LRDY, PGMD, SIZE16, or BUSFLT)
CE	CAS0-CAS3
CK	LCLK1 and LCLK2
CK1	LCLK1
CK2	LCLK2
CKI	CLKIN
CM	CAMD
CT	Any of the bus control output signals (ALTCH, CAS0-CAS3, RAS, WE, TR/QE)
DI	DDIN
DO	DDOUT
EM	EMU0, EMU1, EMU2
HS	HSYNC, VSYNC, CSYNC/HBLNK, or CBLNK/VBLNK
GI	GI
LA	LAD0-LAD31
LINT	LINT1, LINT2
RC	RCA0-RCA12
RE	RAS
RQ	R0 or R1
RS	RESET
S	HSYNC, VSYNC, or CSYNC
SC	EMU3
SCK	SCLK
SF	SF
SG	Any output signal
TR	TR/QE
VCK	VCLK

Lowercase subscripts and their meaning are:

a	access time
c	cycle time (period)
d	delay time
h	hold time
su	setup time
t	transition time
w	pulse duration (width)

The following letters and symbols and their meaning are:

H	High
L	Low
NV	Not valid
V	Valid
Z	High impedance
↑	No longer low
↓	No longer high
Driven	

s = 0	For (i) TMS340X (ii) TMS340X, if CSE bit in CONFIG register is set to 0 (iii) TMS340X, if CSE bit in CONFIG register is set to 1 *and* the cycle is *not* stretched. See page 18.
s = t_Q	For TMS340X, if CSE bit in CONFIG register is set to 1 D *and* the cycle is stretched. See page 18.

general notes on timing parameters

The period of the local clocks (LCLK1 and LCLK2) is four times the period of the input clock (CLKIN).

The quarter cycle time (t_Q) that appears in the following tables is one quarter of a local output clock period, or is equal to the input clock period, $t_{c(CKI)}$.

All output signals from the TMS340X are derived from an internal clock such that all output transitions for a given quarter cycle occur with a minimum of skewing relative to each other. In the timing diagrams, the transitions of all output signals are shown with respect to the local clocks (LCLK1 and LCLK2). The local clock edge used as a reference occurs one internal clock cycle before the transition specified.

PARAMETER MEASUREMENT INFORMATION

The signal combinations shown in the timing parameters are for timing reference only; they do not necessarily represent actual cycles. For actual cycle descriptions, please refer to the cycle timings section of this specification.

CLKIN and $\overline{\text{RESET}}$ timing requirements

NO.	PARAMETER			TMS340X-32 MIN	TMS340X-32 MAX	TMS340X-40 MIN	TMS340X-40 MAX	UNIT
1	$t_{c(CKI)}$	Period of CLKIN (t_Q)		31.25	50	25	50	ns
2	$t_{w(CKIH)}$	Pulse duration, CLKIN high		10		8		ns
3	$t_{w(CKIL)}$	Pulse duration, CLKIN low		10		8		ns
4	$t_{t(CKI)}$	Transition time, CLKIN		2†	5†	2†	5†	ns
5	$t_{h(CKI-RSL)}$	Hold time, $\overline{\text{RESET}}$ low after CLKIN high		10‡		10‡		ns
6	$t_{su(RSH-CKI)}$	Setup time, $\overline{\text{RESET}}$ high to CLKIN↑		4‡		4‡		ns
7	$t_{w(RSL)}$	Pulse duration, $\overline{\text{RESET}}$ low	Initial reset during power up	$160t_Q - 40$§		$160t_Q - 40$§		
			Reset during active operation	$16t_Q - 40$§		$16t_Q - 40$§		
8	$t_{su(CSL-RSH)}$	Setup time of $\overline{\text{HCS}}$ low to $\overline{\text{RESET}}$ high to configure self-bootstrap mode		$8t_Q + 55$		$8t_Q + 55$		ns
9	$t_{d(CS-RSH)}$	Delay time, $\overline{\text{HCS}}$↑ to $\overline{\text{RESET}}$ high to configure self-bootstrap mode			$4t_Q - 50$¶		$4t_Q - 50$¶	ns
10	$t_{w(CSL)}$	Pulse duration, $\overline{\text{HCS}}$ low to configure GSP in self-bootstrap mode		$4t_Q + 55$		$4t_Q + 55$		ns

† These values are based on computer simulation and are not tested.
‡ These timings are required only to synchronize the TMS340X to a particular quarter cycle.
§ The initial reset pulse on powerup must remain valid until all internal states have been initialized. Resets applied after the TMS340X has been initialized need to be present only long enough to be recognized by the internal logic; the internal logic will maintain an internal reset until all internal states have been initialized (34 LCLK1 cycles).
¶ Parameter 9 is the maximum amount by which the $\overline{\text{RESET}}$ low-to-high transition can be delayed after the start of the $\overline{\text{HCS}}$ low-to-high transition and still guarantee that the TMS340X is configured to run in the self-bootstrap mode (HLT bit = 0) following the end of reset.

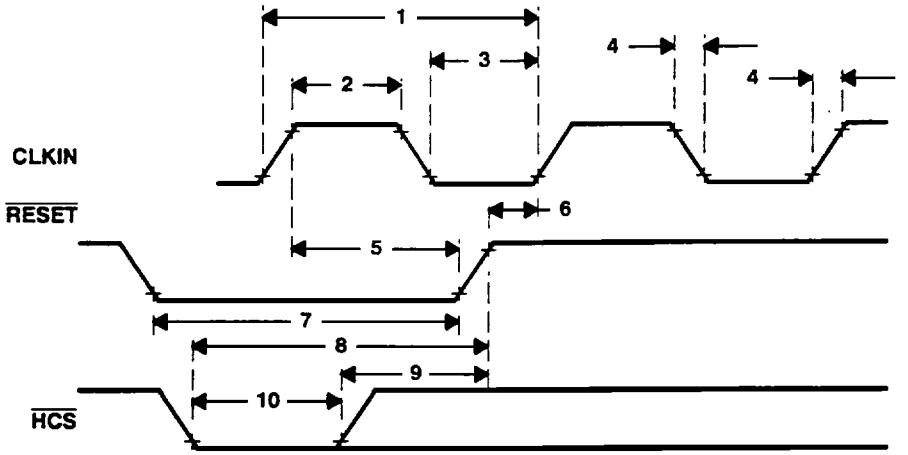


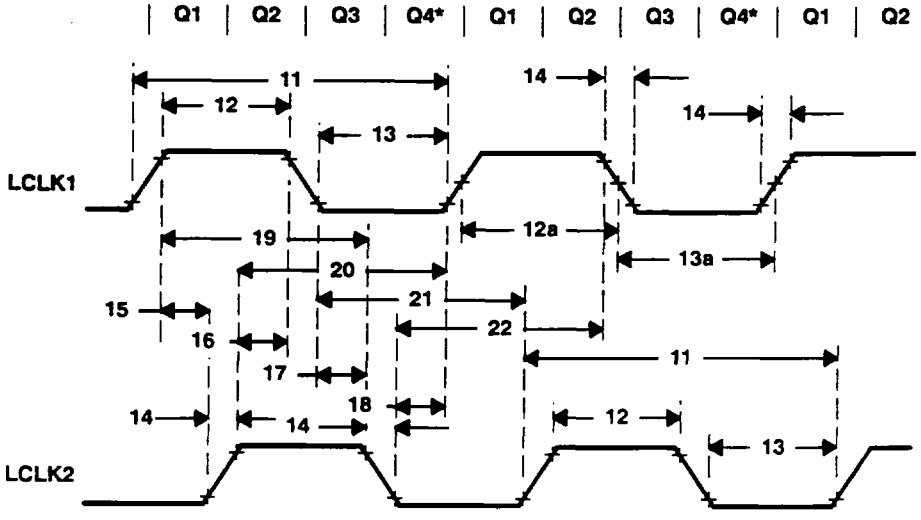
Figure 28. CLKIN And $\overline{\text{RESET}}$ Timing Requirements

PARAMETER MEASUREMENT INFORMATION

local bus timing: output clocks

NO.	PARAMETER		TMS340X-32 MIN	TMS340X-32 MAX	TMS340X-40 MIN	TMS340X-40 MAX	UNIT
11	$t_{c(LCK)}$	Period of local clocks LCLK1, LCLK2	$4t_{c(CKI)}+s$ †		$4t_{c(CKI)}+s$†		ns
12	$t_{w(LCKH)}$	Pulse duration, local clock high	$2t_Q-15$		$2t_Q-13.5$		ns
12a	$t_{w(CK1H)}$	Pulse duration, LCLK1 high	$2t_Q-10$‡		$2t_Q-7$‡		ns
13	$t_{w(LCKL)}$	Pulse duration, local clock low	$2t_Q-15+s$		$2t_Q-13.5+s$		ns
13a	$t_{w(CK1L)}$	Duration of LCLK1 low	$2t_Q-10$‡$+s$		$2t_Q-7+s$		ns
14	$t_{t(LCK)}$	Transition time, LCLK1 or LCLK2		15		13.5	ns
15	$t_{h(CK1H-CK2L)}$	Hold time, LCLK2 low after LCLK1 high	t_Q-15		$t_Q-13.5$		ns
16	$t_{h(CK2H-CK1H)}$	Hold time, LCLK1 high after LCLK2 high	t_Q-15		$t_Q-13.5$		ns
17	$t_{h(CK1L-CK2H)}$	Hold time, LCLK2 high after LCLK1 low	t_Q-15		$t_Q-13.5$		ns
18	$t_{h(CK2L-CK1L)}$	Hold time, LCLK1 low after LCLK2 low	t_Q-15+s		$t_Q-13.5+s$		ns
19	$t_{h(CK1H-CK2H)}$	Hold time, LCLK2 high after LCLK1 high	$3t_Q-15$		$3t_Q-13.5$		ns
20	$t_{h(CK2H-CK1L)}$	Hold time, LCLK1 low after LCLK2 high	$3t_Q-15+s$		$3t_Q-13.5+s$		ns
21	$t_{h(CK1L-CK2L)}$	Hold time, LCLK2 low after LCLK1 low	$3t_Q-15+s$		$3t_Q-13.5+s$		ns
22	$t_{h(CK2L-CK1H)}$	Hold time, LCLK1 high after LCLK2 low	$3t_Q-15+s$		$3t_Q-13.5+s$		ns

† This is a functional minimum and is not tested. This parameter may also be specified as $4t_Q$.
‡ These parameters are specified with 1.5-V timing levels (parameters 12 and 13 are specified with standard timing voltage levels as detailed on page 53).



*See "clock stretch", page 18.
NOTE A: Although LCLK1 and LCLK2 are derived from CLKIN, no timing relationship between CLKIN and the local clocks is to be assumed, except the period of the local clocks is four times the period of CLKIN.

Figure 29. Local Bus Timing: Output Clocks

PARAMETER MEASUREMENT INFORMATION

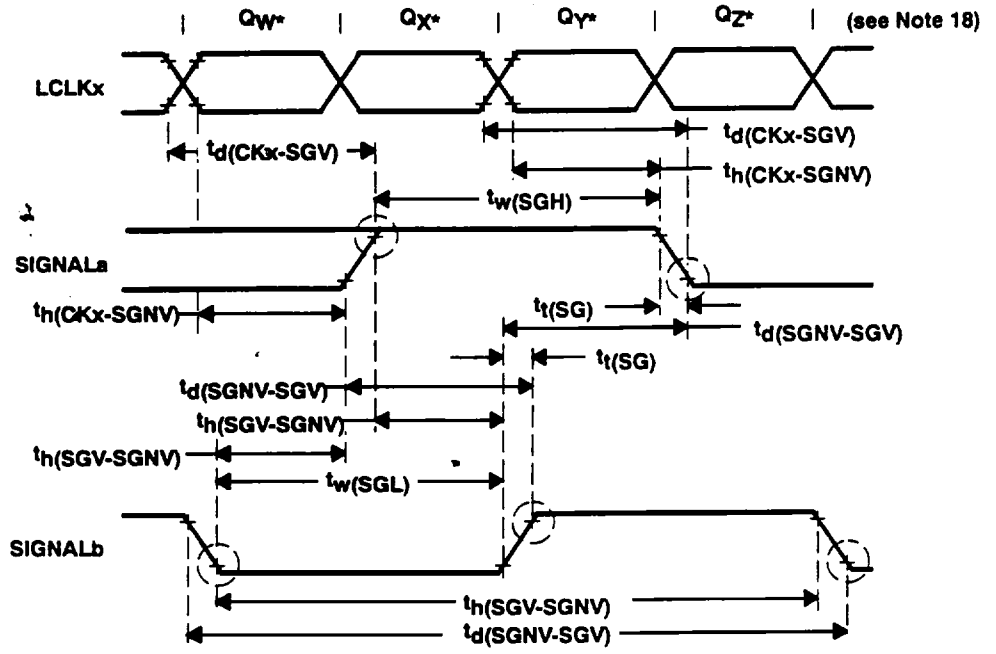
output signal characteristics

The following general parameters are common to all output signals from the TMS340X unless otherwise specifically given. In the minimum and maximum values given, *n* is an integral number of quarter cycles.

PARAMETER			TMS340X-32 MIN	TMS340X-32 MAX	TMS340X-40 MIN	TMS340X-40 MAX	UNIT
$t_{h(CKx\text{-}SGNV)}$	Hold time, LCLKx to output signal not valid		$t_Q - 15$		$t_Q - 13.5$		ns
$t_{h(CKx\text{-}SGNV)}$	Delay time, LCLKx start of transition to output signal valid	Fast: $\overline{RAS}$, $\overline{CAS}$, ALTCH, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$		$t_Q + 15$		$t_Q + 13.5$	ns
		Slow: LAD, RCA, SF		$t_Q + 22$		$t_Q + 20$	ns
$t_{h(SGV\text{-}SGNV)}$	Hold time, output signal valid to output signal not valid	Fast: $\overline{RAS}$, $\overline{CAS}$, ALTCH, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$	$nt_Q - 15$		$nt_Q - 13.5$		ns
		Slow: LAD, RCA, SF	$nt_Q - 22$		$nt_Q - 20$		ns
$t_{d(SGNV\text{-}SGV)}$	Delay time, output signal started transition to output signal valid	Fast: $\overline{RAS}$, $\overline{CAS}$, ALTCH, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$		$nt_Q + 15$		$nt_Q + 13.5$	ns
		Slow: LAD, RCA, SF		$nt_Q + 22$		$nt_Q + 20$	ns
$t_{t(SG)}$	Output signal transition time	Fast: $\overline{RAS}$, $\overline{CAS}$, ALTCH, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$		15		13.5	ns
		Slow: LAD, RCA, SF		22		20	ns
$t_{w(SGH)}$	Pulse duration, output signal high	Fast: $\overline{RAS}$, $\overline{CAS}$, ALTCH, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$	$nt_Q - 15$		$nt_Q - 13.5$		ns
		Slow: LAD, RCA, SF	$nt_Q - 22$		$nt_Q - 20$		ns
$t_{w(SGL)}$	Pulse duration, output signal low	Fast: $\overline{RAS}$, $\overline{CAS}$, ALTCH, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$	$nt_Q - 15$		$nt_Q - 13.5$		ns
		Slow: LAD, RCA, SF	$nt_Q - 22$		$nt_Q - 20$		ns

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION



*See "clock stretch", page 18.

( ) indicates the point at which the signal has attained a valid level.

NOTE A: Any of these quarter phases could be $2t_Q$ if they are stretched. See "clock stretch", page 19.

Figure 30. Output Signal Characteristics

All timing parameters relative to the circled points on the above diagram have a fast and a slow value. Determine which value to use for any parameter by the name of the signal that has the circle on it.

example of how to use the general output signal characteristics

Assume a system is using a TMS340X-32. Determine the maximum time from the start of the falling edge of $\overline{\text{ALTCH}}$ to the time when data must be valid on the LAD bus for a local memory write cycle

From the local memory write cycle diagram on page 26, the time from the falling edge of $\overline{\text{ALTCH}}$ to valid data on the LAD bus is roughly Q3 + Q4; i.e., $2t_Q$. A more precise value can be obtained by using the table of output signal characteristics.

The parameter of interest is $t_{d(SGNV-SGV)}$. Note that in the diagram above, there are two representations of $t_{d(SGNV-SGV)}$ that relate SIGNALa and SIGNALb (the third representation of this parameter relates SIGNALb to itself and is not useful in this example). Let SIGNALa represent $\overline{\text{ALTCH}}$ because $\overline{\text{ALTCH}}$ is making a transition first. Let SIGNALb represent the LAD bus. By definition, the signal becoming valid (SGV) determines whether the fast value or the slow value from the table is used. On the diagram, the SGV points are marked with a circle.

In this case, for parameter $t_{d(SGNV-SGV)}$, SGV is the LAD bus. LAD is in the slow group, so the maximum value for $t_{d(SGNV-SGV)}$ is $nt_Q + 22$. The value for n is 2 from the analysis of the diagram on page 26. Thus, the maximum time from the start of the falling edge of $\overline{\text{ALTCH}}$ to the time when data must be valid on the LAD bus for a local memory write cycle is $2t_Q$ + 22 ns.

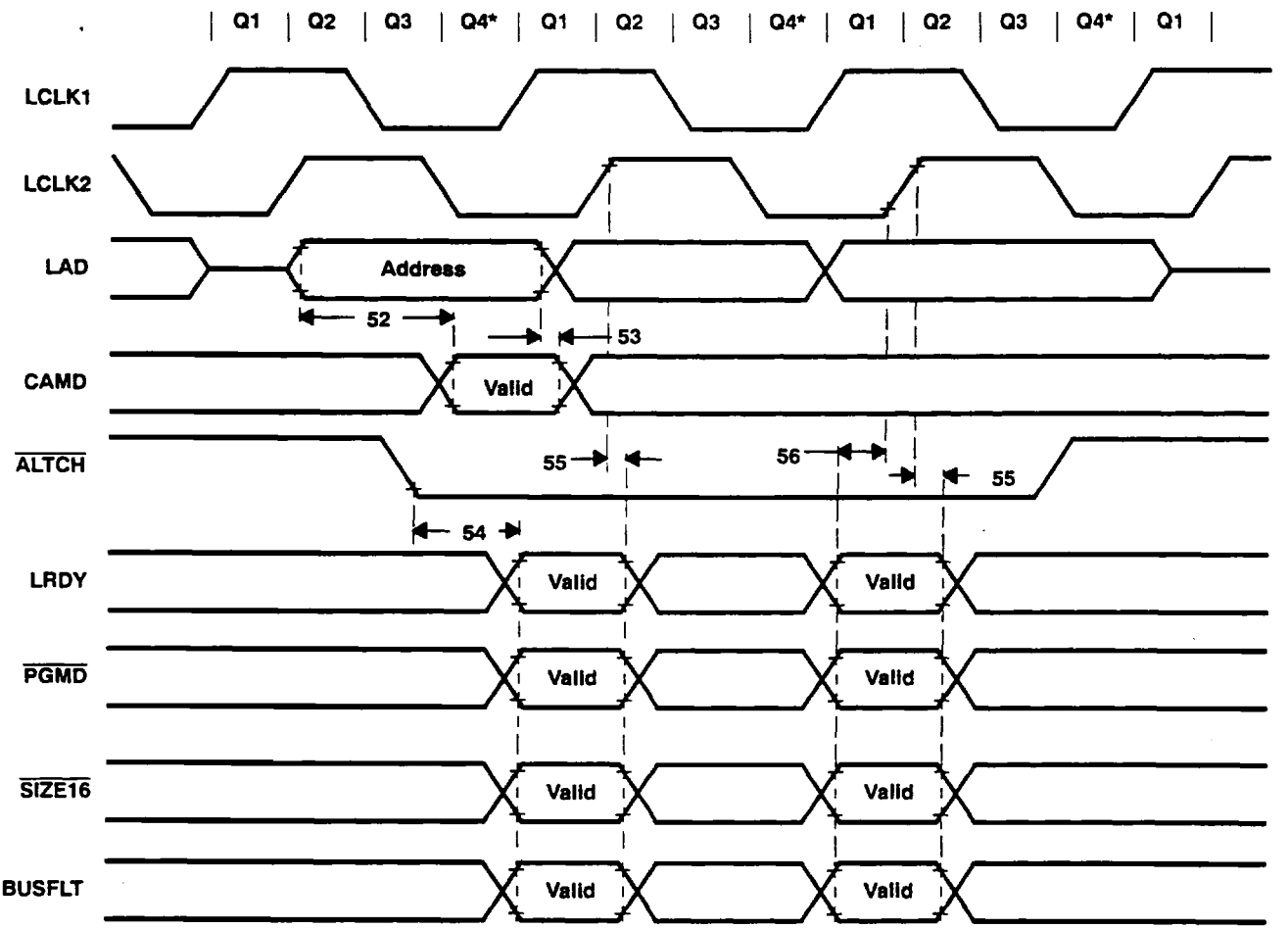
ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

local bus timing: bus control inputs

NO.	PARAMETER		TMS340X-32 MIN	TMS340X-32 MAX	TMS340X-40 MIN	TMS340X-40 MAX	UNIT
52	$t_{a(CMV-LAV)}$†	Access time, $\overline{CAMD}$ valid after address valid on LAD		$3t_Q-45$		$3t_Q-37$	ns
53	$t_{h(LANV-CMV)}$†	Hold time, CAMD valid after address no longer valid on LAD	0		0		ns
54	$t_{a(BCV-A LL)}$†	Access time, control valid (LRDY, $\overline{PGMD}$, $\overline{SIZE16}$, BUSFLT) after $\overline{ALTCH}$ low		$3t_Q-35+s$		$3t_Q-27+s$	ns
55	$t_{h(CK2H-BCV)}$†	Hold time, control (LRDY, $\overline{PGMD}$, $\overline{SIZE16}$, BUSFLT) valid after LCLK2 high	0		0		ns
56	$t_{su(BCV-CK2\uparrow)}$†	Setup time, $\overline{SIZE16}$, LRDY, $\overline{PGMD}$, BUSFLT valid before LCLK2↑	20		15		ns

† CAMD, LRDY, $\overline{PGMD}$, $\overline{SIZE16}$, and BUSFLT are synchronous inputs. The specified setup, access and hold times must be met for proper device operation.



*See "clock stretch", page 18.

Figure 31. Local Bus Timing: Bus Control Inputs

ADVANCE INFORMATION

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

PARAMETER MEASUREMENT INFORMATION

local bus timing: bus control inputs

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
57	$t_{d(CK2\uparrow\text{-}ALL)}$	Delay time, $\overline{ALTCH}$ low after LCLK2↑		t_Q+15		$t_Q+13.5$	ns
58	$t_{d(CK1\downarrow\text{-}ALH)}$	Delay time, $\overline{ALTCH}$ high after LCLK1↓		t_Q+15		$t_Q+13.5$	ns
59	$t_{d(CK1\uparrow\text{-}LAV)}$	Delay time, LAD0-LAD31 address valid after LCLK1↑		t_Q+22		t_Q+20	ns
60	$t_{h(LAV\text{-}CK2L)}$	Hold time, LAD0-LAD31 address valid after LCLK2 low	t_Q-15+s		t_Q-12+s		ns
61	$t_{d(CTNV\text{-}LAD)}$	Delay time, LAD0-LAD31 driven after earlier of DDIN↓ or $\overline{CAS}$↑ or $\overline{TR/QE}$↑	t_Q-5+s †		t_Q-5+s †		ns
62	$t_{h(LAV\text{-}CTV)}$	Hold time, LAD0-LAD31 read data valid after earlier of DDIN low or $\overline{RAS}$, $\overline{CAS}$, or $\overline{TR/QE}$ high	0		0		ns
63	$t_{d(CK2\downarrow\text{-}LAV)}$	Delay time, LAD0-LAD31 data valid after LCLK2↓ (write)		t_Q+22+s		t_Q+20+s	ns
64	$t_{h(CK2L\text{-}LAV)}$	Hold time, LAD0-LAD31 data valid after LCLK2 low (write)	t_Q-15		$t_Q-13.5$		ns
65	$t_{d(CK1\uparrow\text{-}RCV)}$	Delay time, RCA0-RCA12 row address valid after LCLK1↑		t_Q+22		t_Q+20	ns
66	$t_{d(CK2\downarrow\text{-}RCV)}$	Delay time, RCA0-RCA12 column address valid after LCLK2↓		t_Q+22+s		t_Q+20+s	ns
67	$t_{h(RCV\text{-}CK2L)}$	Hold time, RCA0-RCA12 address valid after LCLK2 low	t_Q-15		t_Q-12+s		ns
68	$t_{d(CK1\uparrow\text{-}DIH)}$	Delay time, DDIN high after LCLK1↑		t_Q+15		$t_Q+13.5$	ns
69	$t_{d(CK1\downarrow\text{-}DIL)}$	Delay time, DDIN low after LCLK1↓		t_Q+15		$t_Q+13.5$	ns
70	$t_{d(CK1\uparrow\text{-}DOL)}$	Delay time, $\overline{DDOUT}$ low after LCLK1↑		t_Q+15		$t_Q+13.5$	ns
71	$t_{d(CK1\downarrow\text{-}DOH)}$	Delay time, $\overline{DDOUT}$ high after LCLK1↓		t_Q+15		$t_Q+13.5$	ns
72	$t_{d(CK2\downarrow\text{-}DOL)}$	Delay time, $\overline{DDOUT}$ low after LCLK2↓		t_Q+15+s		$t_Q+13.5+s$	ns
73	$t_{su(LAV\text{-}AL\downarrow)}$	Setup time, LAD0-LAD31 data valid before $\overline{ALTCH}$↓	t_Q-16		t_Q-13		ns
74	$t_{en(DAV\text{-}DIH)}$	Enable time, data valid after DDIN high (see Note NO TAG)		$2t_Q-20$		$2t_Q-17$	ns
75	$t_{dis(DAV\text{-}DIL)}$	Disable time, data high-impedance after DDIN low (see Note NO TAG)		t_Q-12+s†		t_Q-10+s†	ns

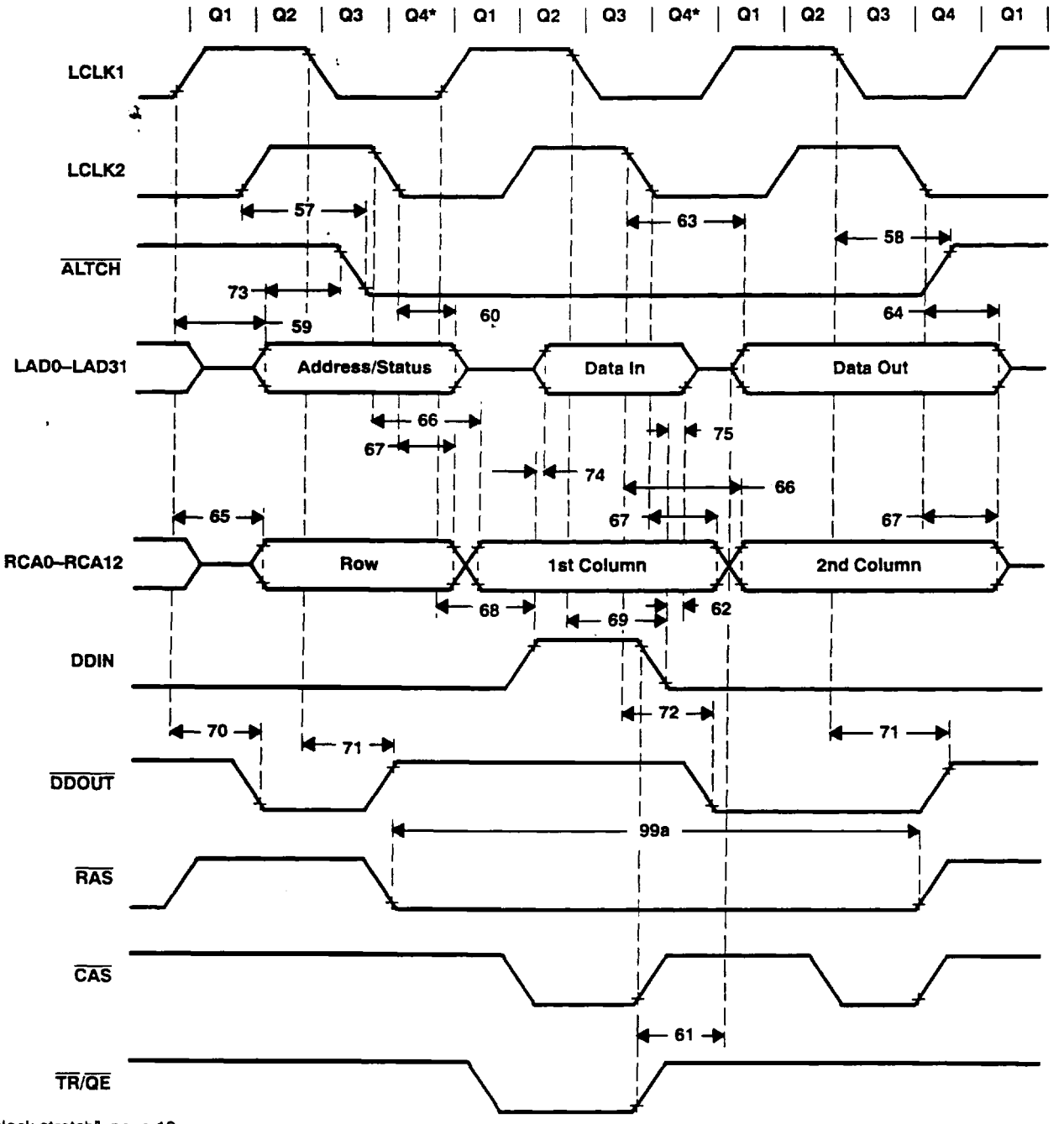
† This value is not guaranteed.

NOTE 3: DDIN is used to control LAD bus buffers between the TMS340X and local memory. Parameter 74 references the time for these data buffers to go from a high-impedance state to an active level. Parameter 75 references the time for the buffers to go from an active level to the high-impedance state.

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

local bus timing (continued)



*See "clock stretch", page 18.

Figure 32. Local Bus Timing (Continued)

PARAMETER MEASUREMENT INFORMATION

local bus timing: $\overline{RAS}$, $\overline{CAS0}$-$\overline{CAS3}$, $\overline{WE}$, $\overline{TR/QE}$, and SF

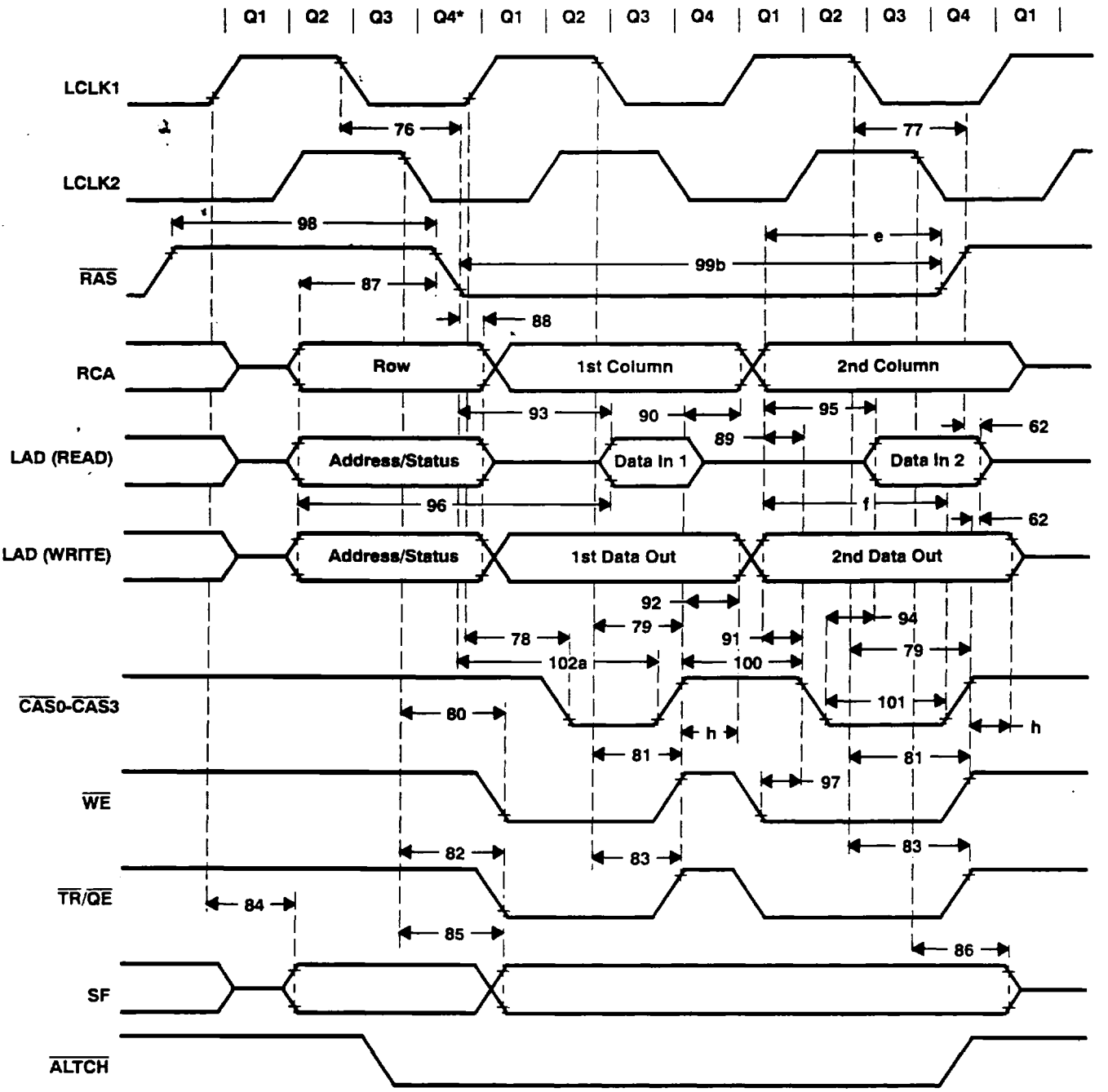
NO.	PARAMETER		TMS340X-32		TMS340XA-40		UNIT
			MIN	MAX	MIN	MAX	
62	$t_{h(LAV-CTV)}$	Hold time, LAD0-LAD31 read data valid after earlier of DDIN, LCLK2 low or $\overline{RAS}$, $\overline{CAS}$, or $\overline{TR/QE}$ high	0		0		ns
76	$t_{d(CK1\downarrow-REL)}$	Delay time, $\overline{RAS}$ low after LCLK1↓		t_Q+12+s		t_Q+10+s	ns
77	$t_{d(CK1\downarrow-REH)}$	Delay time, $\overline{RAS}$ high after LCLK1↓		t_Q+12		t_Q+10	ns
78	$t_{d(CK1\uparrow-CEL)}$	Delay time, $\overline{CAS}$ low after LCLK1↑		t_Q+12		t_Q+10	ns
79	$t_{d(CK1\downarrow-CEH)}$	Delay time, $\overline{CAS}$ high after LCLK1↓		t_Q+12		t_Q+10	ns
80	$t_{d(CK2\downarrow-WEL)}$	Delay time, $\overline{WE}$ low after LCLK2↓		t_Q+15+s		$t_Q+13.5+s$	ns
81	$t_{d(CK1\downarrow-WEH)}$	Delay time, $\overline{WE}$ high after LCLK1↓		t_Q+15		$t_Q+13.5$	ns
82	$t_{d(CK2\downarrow-TRL)}$	Delay time, $\overline{TR/QE}$ low after LCLK2↓		t_Q+15+s		$t_Q+13.5+s$	ns
83	$t_{d(CK1\downarrow-TRH)}$	Delay time, $\overline{TR/QE}$ high after LCLK1↓		t_Q+15		$t_Q+13.5$	ns
84	$t_{d(CK1\uparrow-SFV)}$	Delay time, SF valid after LCLK1↑		t_Q+22		t_Q+20	ns
85	$t_{d(CK2\downarrow-SFV)}$	Delay time, SF valid after LCLK2↓		t_Q+22+s		t_Q+20+s	ns
86	$t_{d(CK2\downarrow-SFZ)}$	Delay time, SF high-impedance after LCLK2↓		t_Q+22 †		t_Q+20 †	ns
87	$t_{su(ADV-RE\downarrow)}$‡	Setup time, row address valid before $\overline{RAS}$↓	$2t_Q-22$		$2t_Q-20$		ns
88	$t_{h(ADV-REL)}$‡	Hold time, row address valid after $\overline{RAS}$ low	t_Q-5+s		t_Q-5+s		ns
89	$t_{su(RCV-CE\downarrow)}$	Setup time, column address valid before $\overline{CAS}$↓	t_Q-22		t_Q-20		ns
90	$t_{h(RCV-CEH)}$	Hold time, column address valid after $\overline{CAS}$ high	t_Q-15		$t_Q-13.5$		ns
91	$t_{su(CAV-CE\downarrow)}$	Setup time, write data valid before $\overline{CAS}$↓	t_Q-22		t_Q-20		ns
92	$t_{h(CAV-CEH)}$	Hold time, write data valid after $\overline{CAS}$ high	t_Q-15		$t_Q-13.5$		ns
93	$t_{a(LAV-REL)}$	Access time, data in valid after $\overline{RAS}$ low (assuming maximum transition time)		$4t_Q-8+s$		$4t_Q-8+s$	ns
94	$t_{a(LAV-CEL)}$	Access time, data in valid after $\overline{CAS}$↓		$2t_Q-8$		$2t_Q-8$	ns
97	$t_{su(WEL-CE\downarrow)}$	Setup time, write low before $\overline{CAS}$↓ (on write cycles)	t_Q-15		$t_Q-13.5$		ns
98	$t_{w(REH)}$	Pulse duration of $\overline{RAS}$ high	$4t_Q-12+s$		$4t_Q-10+s$		ns
99a	$t_{w(REL)}$	Pulse duration of $\overline{RAS}$ low	$4nt_Q-12+s'$ †‡		$4nt_Q-4+s'$ †‡		ns
99b	$t_{w(REL)}$	Pulse duration of $\overline{RAS}$ low	$4nt_Q-12+s$ ‡		$4nt_Q-4+s$ ‡		ns
100	$t_{w(CEH)}$	Pulse duration of $\overline{CAS}$ high	$2t_Q-12$		$2t_Q-10$		ns
101	$t_{w(CEL)}$	Pulse duration of $\overline{CAS}$ low	$2t_Q-8$		$2t_Q-8$		ns
102	$t_{d(REL-CE\uparrow)}$	Delay time $\overline{RAS}$ low to $\overline{CAS}$↑	$4t_Q-12+s$		$4t_Q-4+s$		ns

† These values are derived from characterization data and are not tested.
‡ Parameters 87 and 88 also apply to $\overline{WE}$, $\overline{TR/QE}$, and SF relative to $\overline{RAS}$.
§ s' is $2t_Q$ since both the address cycle and the read data cycle of a read-modify-write will be stretched. See "clock stretch", page 18.
¶ n = number of GSP data cycles in in the memory access.
NOTE 4: Parameters 95 and 96 have been eliminated.

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

local bus timing (continued)



*See "clock stretch", page 18.

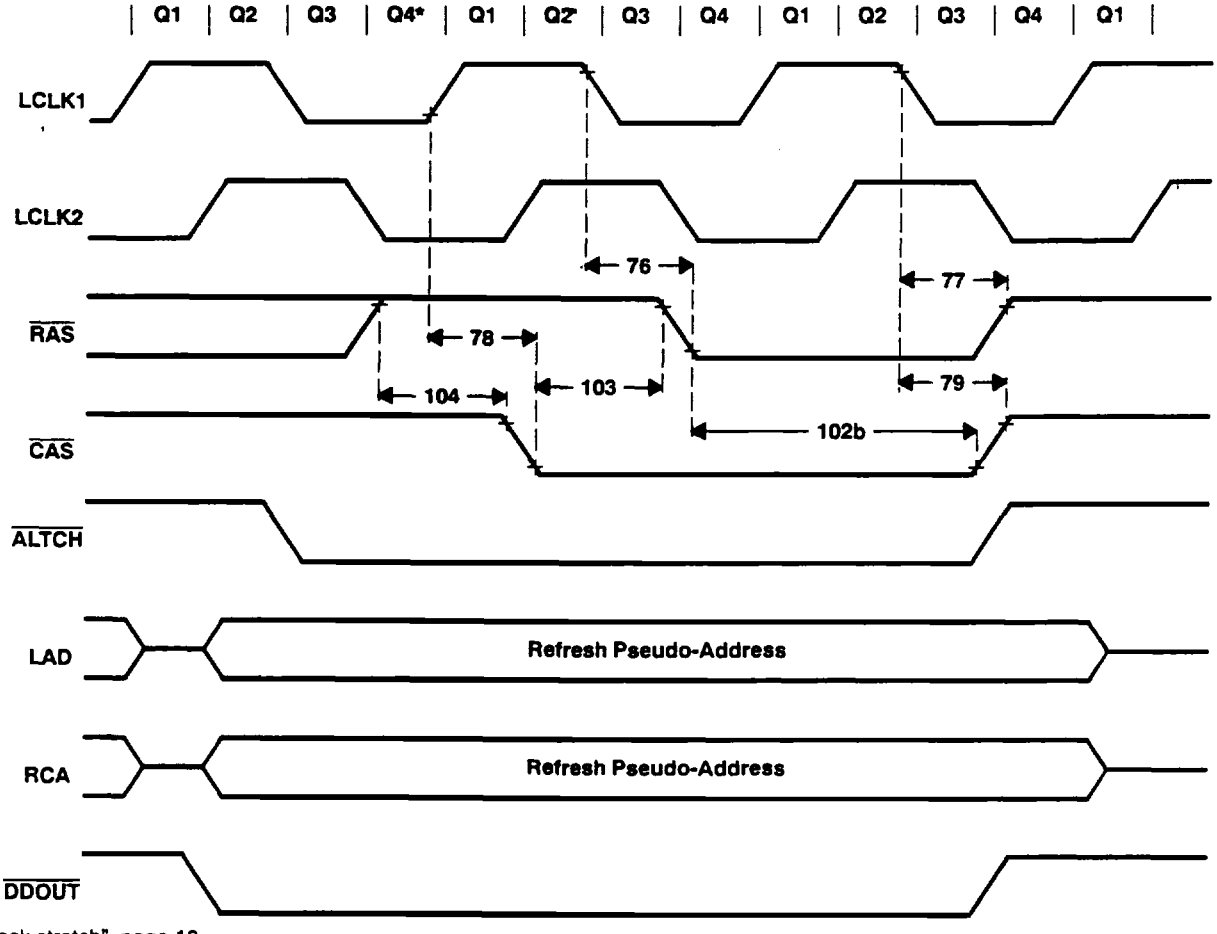
Figure 33. Local Bus Timing (Continued)

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

CAS-before-RAS refresh: RAS and CAS0-CAS3

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
76	$t_{d(CK1\downarrow\text{-}REL)}$	Delay time, RAS low after LCLK1↓		t_Q+12		t_Q+10	ns
77	$t_{d(CK1\downarrow\text{-}REH)}$	Delay time, RAS high after LCLK1↓		t_Q+12		t_Q+10	ns
78	$t_{d(CK1\text{-}\uparrow CEL)}$	Delay time, CAS low after LCLK1↑		t_Q+12		t_Q+10	ns
79	$t_{d(CK1\downarrow\text{-}CEH)}$	Delay time, CAS high after LCLK1↓		t_Q+12		t_Q+10	ns
102a	$t_{d(REL\text{-}CE\uparrow)}$	Delay time, RAS low to CAS↑	$4t_Q-12+s$		$4t_Q-4+s$		ns
102b	$t_{d(REL\text{-}CE\uparrow)}$	Delay time, RAS low to CAS↑	$4t_Q-12$		$4t_Q-4$		ns
103	$t_{d(CEL\text{-}RE\downarrow)}$	Delay time, CAS low to RAS↓	$2t_Q-15$		$2t_Q-13.5$		ns
104	$t_{d(REH\text{-}CE\downarrow)}$	Delay time, RAS high to CAS↓	$2t_Q-15+s$		$2t_Q-13.5+s$		ns



*See "clock stretch", page 18.
NOTE A: ALTCH, LAD, RCA, and DDOUT are shown for reference only.

Figure 34. CAS-Before-RAS Refresh: RAS and CAS0-CAS3

The refresh pseudo-address present on LAD0-LAD31 is the output from the 16-bit refresh address register (IO) register located at C000 01F0h) on LAD16-LAD31. LAD0-LAD3 have the refresh status code (Status Code = 0011) and LAD4-LAD15 are held low.

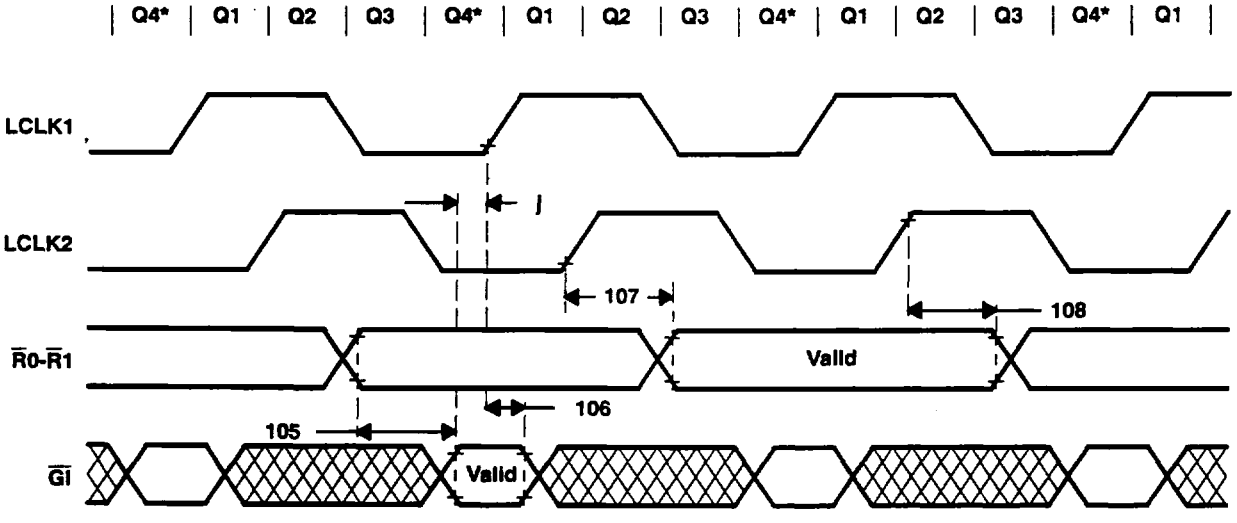
ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

multiprocessor interface timing: $\overline{GI}$, $\overline{ALTCH}$, $\overline{RAS}$, $\overline{R0}$ and $\overline{R1}$

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
105	$t_{a(GIV-RQV)}$	Access time, $\overline{GI}$ valid after $\overline{R0}$ and $\overline{R1}$ valid (see Note 5)		$2t_Q-40$		$2t_Q-30$	ns
j	$t_{su(GIV-CK1)}$	Setup time, $\overline{GI}$ valid before LCLK1 no longer low (see Note 5)	40		35		ns
106	$t_{h(CK1\uparrow-GIV)}$	Hold time, $\overline{GI}$ valid after LCLK1↑ (see Note 5)	0		0		ns
107	$t_{d(CK2\uparrow-RQV)}$*	Delay time, LCLK2↑ to $\overline{R0}$ or $\overline{R1}$ valid		t_Q+15		$t_Q+13.5$	ns
108	$t_{d(CK2H-RQNV)}$	Delay time, LCLK2 high to $\overline{R0}$ or $\overline{R1}$ no longer valid	t_Q-15		$t_Q-13.5$		ns

NOTE 5: These timings must be met to insure that the $\overline{GI}$ input is recognized on this clock cycle.



*See "clock stretch", page 18.

For a TMS340X to gain control of the local bus during a given cycle, its $\overline{GI}$ pin must be low at the start of Q1 (indicating that the bus arbitration logic is granting the bus to this processor).

Figure 35. Multiprocessor Interface Timing: $\overline{GI}$, $\overline{ALTCH}$, $\overline{RAS}$, $\overline{R0}$ and $\overline{R1}$

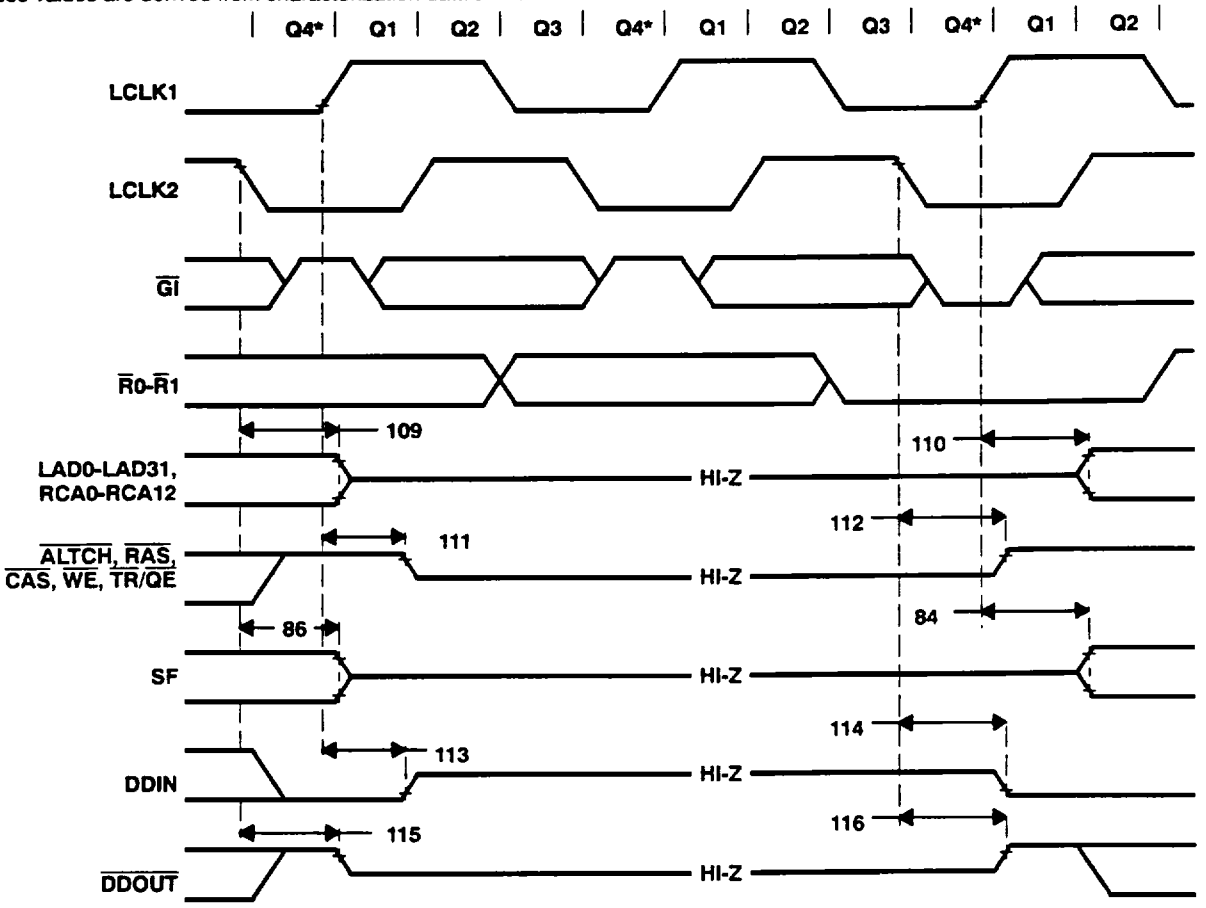
ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

multiprocessor interface timing: high-impedance signals

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
84	$t_{d(CK1\uparrow\text{-}SFV)}$	Delay time, SF valid after LCLK1↑		t_Q+22		t_Q+20	ns
86	$t_{d(CK2\downarrow\text{-}SFZ)}$	Delay time, SF high-impedance after LCLK2↓		t_Q+22+s†		t_Q+20+s†	ns
109	$t_{d(CK2\downarrow\text{-}ADZ)}$	Delay time, LAD and RCA high-impedance after LCLK2↓		t_Q+22+s†		t_Q+20+s†	ns
110	$t_{d(CK1\uparrow\text{-}ADV)}$	Delay time, LAD and RCA valid after LCLK1↑		t_Q+22		t_Q+20	ns
111	$t_{d(CK1\uparrow\text{-}CTZ)}$	Delay time, $\overline{ALTCH}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{WE}$, $\overline{TR/QE}$, $\overline{HOE}$, and HDST high-impedance after LCLK1↑		t_Q+15†		$t_Q+13.5$†	ns
112	$t_{d(CK2\downarrow\text{-}CTH)}$	Delay time, $\overline{ALTCH}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{WE}$, $\overline{TR/QE}$, $\overline{HOE}$, and HDST high-impedance after LCLK2↓		t_Q+15+s		$t_Q+13.5+s$	ns
113	$t_{d(CK1\uparrow\text{-}DIZ)}$	Delay time, DDIN high-impedance after LCLK↑		t_Q+15†		$t_Q+13.5$†	ns
114	$t_{d(CK2\downarrow\text{-}DIL)}$	Delay time, DDIN low after LCLK2↓		t_Q+15+s		$t_Q+13.5+s$	ns
115	$t_{d(CK2\downarrow\text{-}DOZ)}$	Delay time, $\overline{DDOUT}$ high-impedance after LCLK2↓		t_Q+15+s †		$t_Q+13.5+s$†	ns
116	$t_{d(CK2\downarrow\text{-}DOH)}$	Delay time, $\overline{DDOUT}$ high after LCLK2↓		t_Q+15+s		$t_Q+13.5+s$	ns

† These values are derived from characterization data and are not tested.



*See "clock stretch", page 18.

Figure 36. Multiprocessor Interface Timing: High-impedance Signals

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

video shift clock timing: SCLK

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
117	$t_{c(SCK)}$	Period of video serial clock SCLK	35	50	25	50	ns
118	$t_{w(SCKH)}$	Pulse duration of SCLK high	12		10		ns
119	$t_{w(SCKL)}$	Pulse duration of SCLK low	12		10		ns
120	$t_{t(SCK)}$	Transition time (rise and fall) of SCLK	2†	5†	2†	5†	ns

† This value is determined through computer simulation and is not tested.

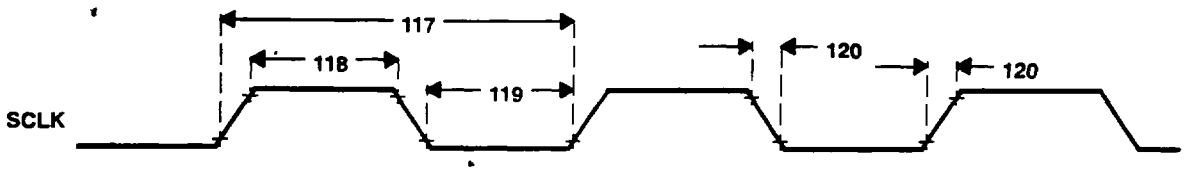


Figure 37. Video Shift Clock Timing: SCLK

video interface timing: VCLK and video outputs

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
123	$t_{c(VCK)}$	Period of video input clock VCLK	62.5	100	62.5	100	ns
124	$t_{w(VCKH)}$	Pulse duration of VCLK high	28		28		ns
125	$t_{w(VCKL)}$	Pulse duration of VCLK low	28		28		ns
126	$t_{t(VCK)}$	Transition time (rise and fall) of VCLK	2†	5†	2†	5†	ns
127	$t_{d(VCKL-HSL)}$	Delay time, VCLK low to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC/VBLNK}}$ or $\overline{\text{CBLNK/VBLNK}}$ low		40		40	ns
128	$t_{d(VCKL-HSH)}$	Delay time, VCLK low to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC/HBLNK}}$, or $\overline{\text{CBLNK/VBLNK}}$ high		40		40	ns
129	$t_{h(VCK\downarrow-HS\downarrow)}$	Hold time, VCLK↓ to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC/HBLNK}}$, or $\overline{\text{CBLNK/VBLNK}}$↓	0†		0†		ns
130	$t_{h(VCK\downarrow-HS\uparrow)}$	Hold time, VCLK↓ to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC/HBLNK}}$, or $\overline{\text{CBLNK/VBLNK}}$↑	0†		0†		ns

† This value is determined through computer simulation and is not tested.

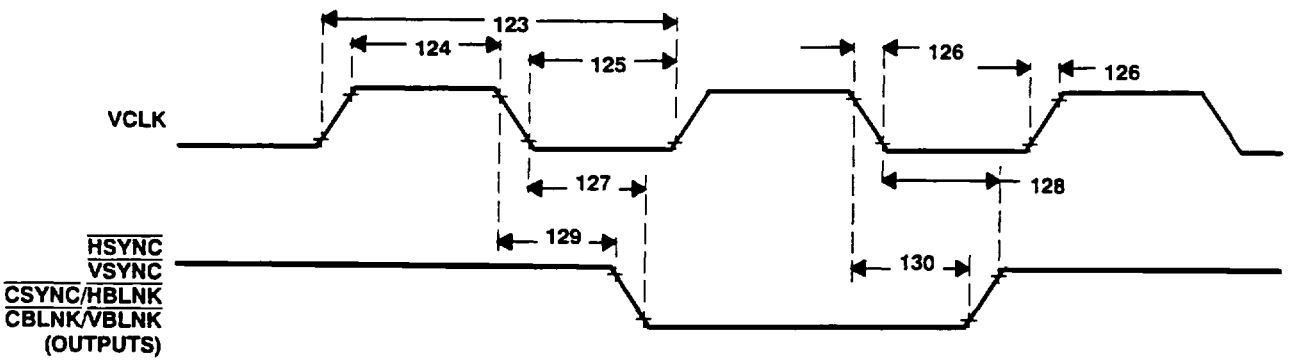


Figure 38. Video Interface Timing: VCLK and Video Outputs

ADVANCE INFORMATION

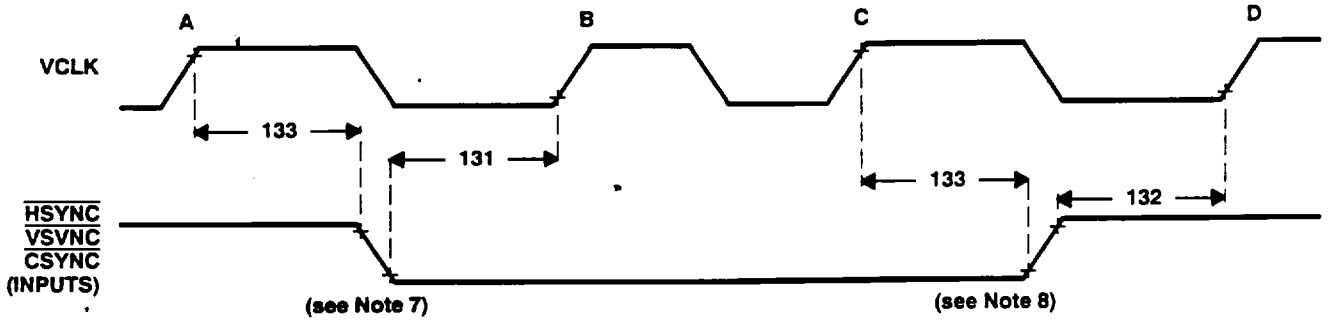
TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

PARAMETER MEASUREMENT INFORMATION

video interface timing: external sync inputs (see Note 6)

NO.	PARAMETER		TMS340X-32 MIN	TMS340X-32 MAX	TMS340X-40 MIN	TMS340X-40 MAX	UNIT
131	$t_{su(SL-VCK\uparrow)}$	Setup time, $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$ low to VCLK↑	20		20		ns
132	$t_{su(SH-VCK\uparrow)}$	Setup time, $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$ high to VCLK↑	20		20		ns
133	$t_{h(VCKH-SV)}$	Hold time, $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$ valid after VCLK high	20		20		ns

NOTE 6: Setup and hold times on asynchronous inputs are required only to insure recognition at indicated clock edges.



NOTES: 7. If the falling edge of the sync signal occurs more than $t_{h(VCKH-SV)}$ after VCLK edge A and at least $t_{su(SL-VCKH\uparrow)}$ before edge B, the transition will be detected at edge B instead of edge A.

8. If the rising edge of the sync signal occurs more than $t_{h(VCKH-SV)}$ after VCLK edge C and at least $t_{su(SH-VCKH\uparrow)}$ before edge D, the transition will be detected at edge D instead of edge C.

Figure 39. Video Interface Timing: External Sync Inputs (see Note 6)

interrupt timing: $\overline{LINT1}$ and $\overline{LINT2}$

NO.	PARAMETER		TMS340X-32 MIN	TMS340X-32 MAX	TMS340X-40 MIN	TMS340X-40 MAX	UNIT
134	$t_{su(LINTL-CK2\uparrow)}$	Setup time, $\overline{LINT1}$ or $\overline{LINT2}$ low before LCLK2↑	t_Q+45 †		t_Q+40 †		ns
135	$t_{w(LINTL)}$	Pulse duration of $\overline{LINT1}$ or $\overline{LINT2}$ low	$8t_Q$‡		$8t_Q$‡		ns

† Although $\overline{LINT1}$ and $\overline{LINT2}$ may be asynchronous to the TMS340X, this setup insures recognition of the interrupt on this clock edge.

‡ This pulse duration minimum insures that the interrupt is recognized by internal logic; however, the level must be maintained until it has been acknowledged by the interrupt service routine.

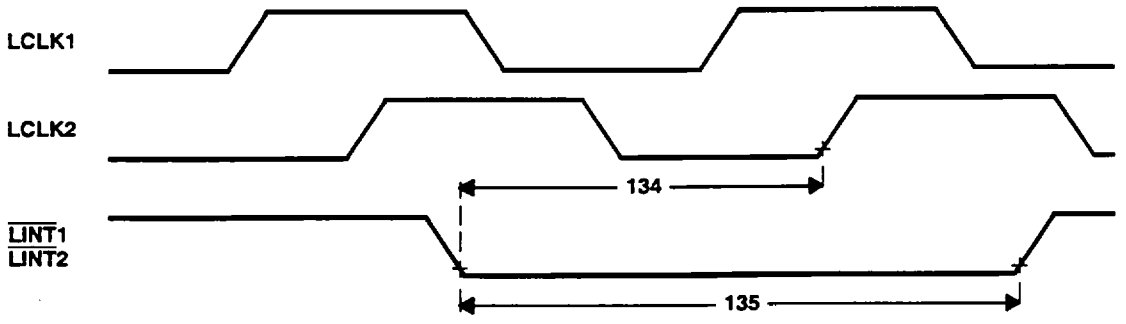


Figure 40. Interrupt Timing: $\overline{LINT1}$ and $\overline{LINT2}$

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

emulator interface timing

NO.	PARAMETER		TMS340X-32		TMS340X-40		UNIT
			MIN	MAX	MIN	MAX	
137	t_{su}(EMV-CK1↑)	Setup time, EMU0-EMU2 valid to LCLK1↑	30		25		ns
138	t_h(EMV-CK1↑)	Hold time, EMU0-EMU2 valid after LCLK1↑	0		0		ns
139	t_d(CK1L-SCV)	Delay time, EMU3 valid after LCLK1 low		25		20	ns
140	t_h(CK2H-SCNV)	Hold time, LCLK2 high before EMU3 not valid	t_O−15		t_O−13.5		ns

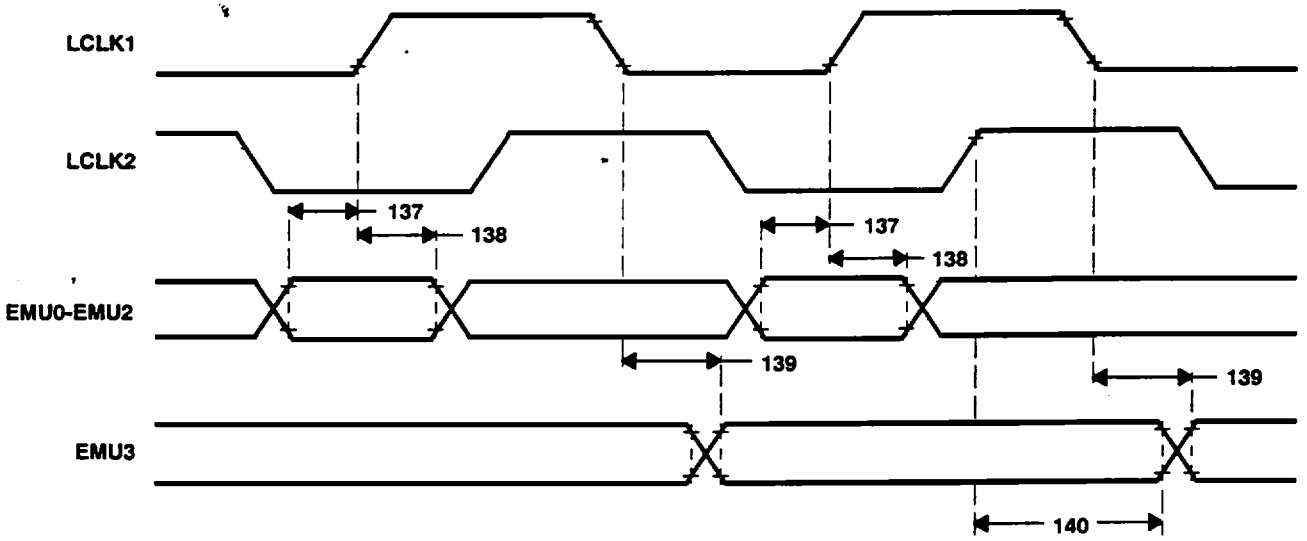
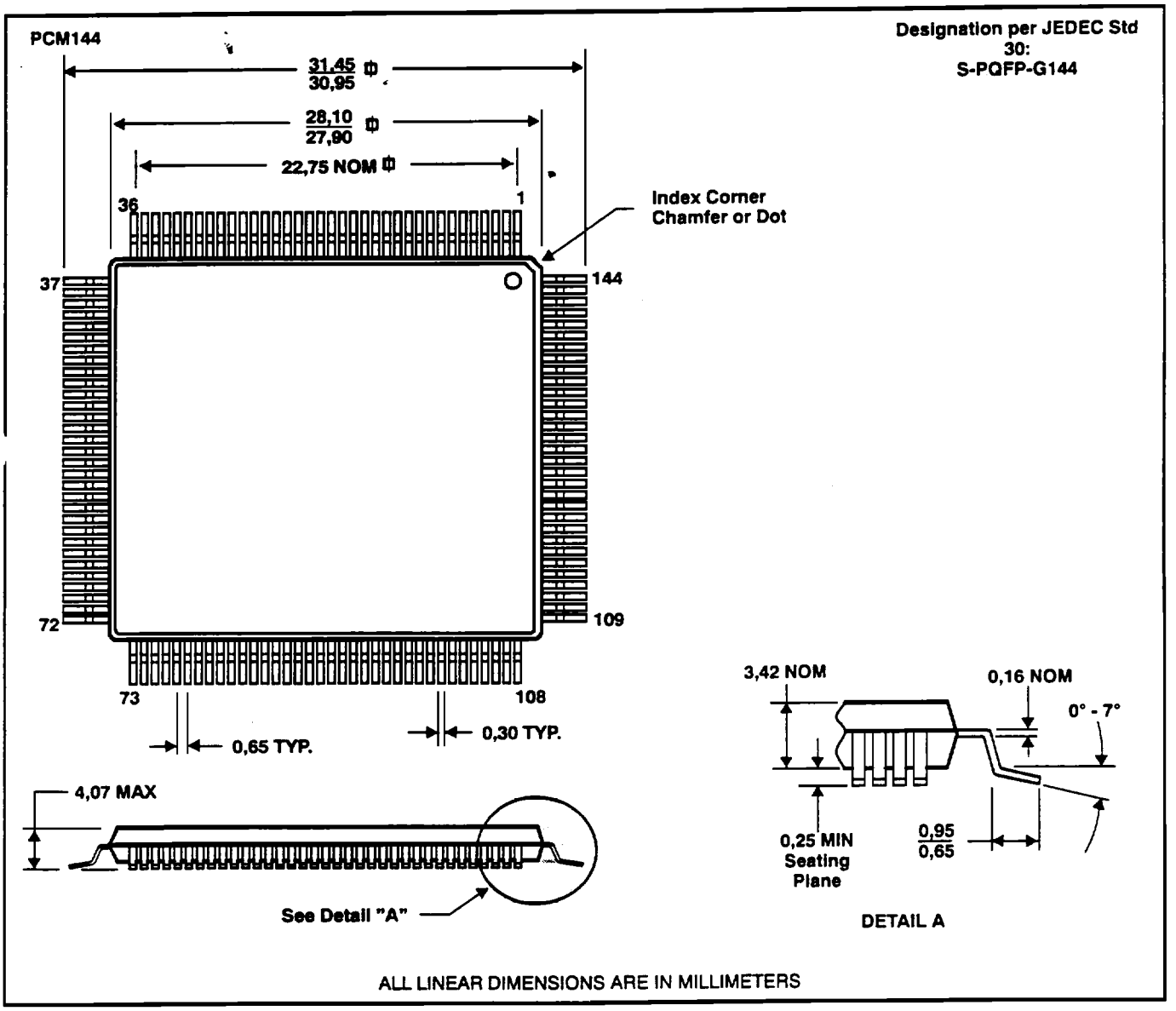


Figure 41. Emulator Interface Timing

MECHANICAL DATA

PCM144
JEDEC metric plastic quad flatpack

This plastic package consists of a circuit mounted on a lead frame and encapsulated within an electrically nonconductive plastic compound. The compound withstands soldering temperatures with no deformation, and circuit performance characteristics remain stable when the devices are operated in high-humidity conditions. The package is intended for surface mounting and leads are spaced on 0,65 mm centers with an 0,80 mm foot length. Leads require no additional cleaning or processing when used in soldered assembly.



NOTES: A. Maximum deviation from coplanarity is 0,1 mm.
B. All dimensions and notes for JEDEC outline MO-xxxxx apply.