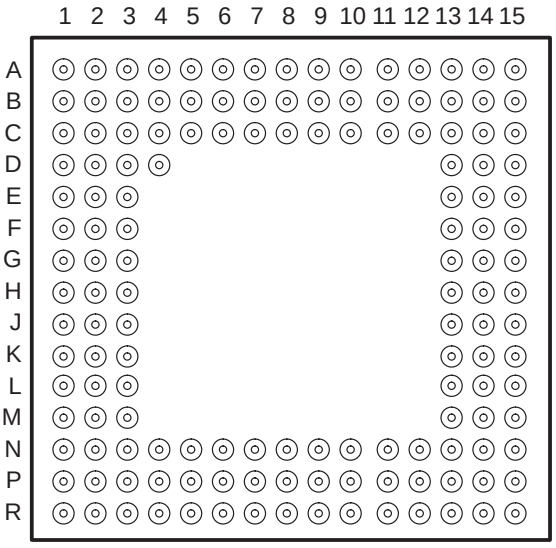
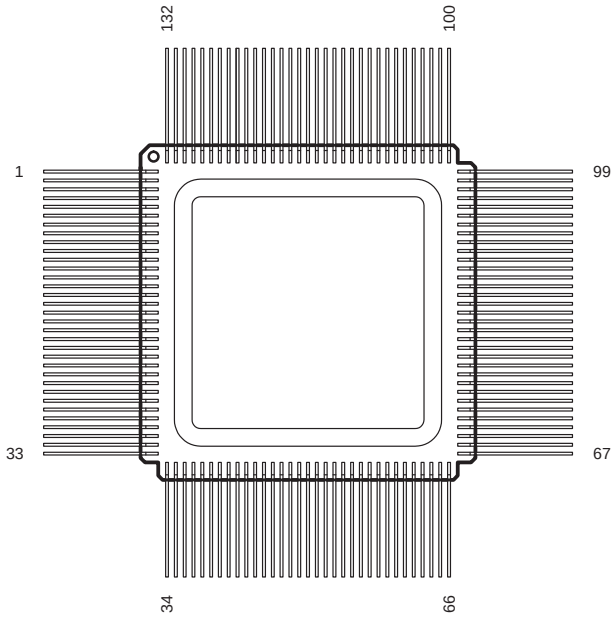
SGUS011D − APRIL 1991 − REVISED SEPTEMBER 2004

- **Class B High-Reliability Processing**
- **1-μm CMOS Technology**
- **Military Operating Temperature Range −55°C to 125°C**
- **SMJ34020A-32/40 125/100-ns Instruction Cycle Time**
- **Fully Programmable 32-Bit General-Purpose Processor With 512-Megabyte Linear Address Range (Bit Addressable)**
- **Second-Generation Graphics System Processor**
  - **Object-Code Compatible With the SMJ34010**
  - **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 Six Arithmetic Pixel Processing Options (Raster Ops)**
- **512-Byte LRU On-Chip Instruction Cache**
- **Optimized DRAM/VRAM Interface**
  - **Page-Mode for Burst Memory Operations**
  - **Dynamic Bus Sizing (16-Bit and 32-Bit Transfers)**
  - **Byte-Oriented $\overline{\text{CAS}}$ Strobes**
- **Flexible Host Processor Interface**
  - **Supports Host Transfers**
  - **Direct Access to All of the SMJ34020A Address Space**
  - **Implicit Addressing**
  - **Prefetch for Enhanced Read Access**
- **Programmable CRT Control**
  - **Composite Sync Mode**
  - **Separate Sync Mode**
  - **Synchronization to External Sync**
- **Direct Support for Special Features of 1M VRAMs**
  - **Load Write Mask**
  - **Load Color Mask**
  - **Block Write**
  - **Write Using the Write Mask**
- **Flexible Multi-Processor Interface**
- **Packaging Options**
  - **145-Pin Grid Array Ceramic Package (GB Suffix)**
  - **132-Pin Ceramic Quad Flat Pack (Unformed Lead) (HT Suffix)**

145-PIN GRID ARRAY PACKAGE
(TOP VIEW)



132-PIN QUAD FLATPACK
(TOP VIEW)



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

description

The SMJ34020A graphics system processor (GSP) is 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 SMJ34020A 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 SMJ34020A is a CMOS 32-bit processor with hardware support for graphics operations such as pixel block transfers (PIXBLTS) during raster operations 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 SMJ34020A 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 SMJ34020A directly interfaces to DRAMs and VRAMs and generates raster control signals. The SMJ34020A can be configured to operate as a standalone processor, or it can be used as a graphics engine with a host system. The host interface provides a generalized communication port for any standard host processor. The SMJ34020A 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 SMJ34020A provides single-cycle execution of general-purpose instructions and most common integer arithmetic and Boolean operations from its instruction cache. Additionally, the SMJ34020A 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 SMJ34020A 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 pixel transfer (PIXT) instructions or on two-dimensional arrays of arbitrary size (PIXBLTS).

The SMJ34020A'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 SMJ34020A 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 SMJ34020A.

Pin Assignments − 145-Pin Grid Array Package

PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME
A1	V_{SS}	C9	RCA8	J1	EMU0	N15	LAD17
A2	$\overline{\text{ALTCH}}$	C10	RCA12	J2	$\overline{\text{GI}}$	P1	V_{CC}
A3	$\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$	C11	LAD30	J3	EMU1	P2	$\overline{\text{HWRITE}}$
A4	HSYNC	C12	V_{SS}	J13	LAD4	P3	$\overline{\text{HCS}}$
A5	$\overline{\text{TR}}$/$\overline{\text{QE}}$	C13	V_{SS}	J14	V_{CC}	P4	HA30
A6	RCA2	C14	V_{CC}	J15	LAD5	P5	HA27
A7	RCA3	C15	LAD26	K1	EMU2	P6	HA24
A8	V_{CC}	D1	$\overline{\text{RAS}}$	K2	$\overline{\text{RESET}}$	P7	HA22
A9	RCA6	D2	$\overline{\text{CAS2}}$	K3	$\overline{\text{LINT2}}$	P8	HA18
A10	RCA7	D3	V_{SS}	K13	V_{SS}	P9	HA14
A11	RCA10	D4†	NU†	K14	LAD3	P10	HA13
A12	SCLK	D13	LAD28	K15	LAD20	P11	HA10
A13	LAD15	D14	LAD11	L1	$\overline{\text{LINT1}}$	P12	HA7
A14	LAD29	D15	LAD10	L2	CAMD	P13	HA5
A15	V_{SS}	E1	$\overline{\text{R1}}$	L3	LRDY	P14	HBS0
B1	$\overline{\text{CAS3}}$	E2	V_{CC}	L13	LAD1	P15	LAD0
B2	$\overline{\text{WE}}$	E3	$\overline{\text{CAS1}}$	L14	LAD2	R1	$\overline{\text{HREAD}}$
B3	V_{SS}	E13	LAD27	L15	LAD19	R2	HA31
B4	$\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$	E14	LAD25	M1	BUSFLT	R3	HA28
B5	VSYNC	E15	LAD9	M2	$\overline{\text{PGMD}}$	R4	HA26
B6	RCA0	F1	HRDY	M3	VCLK	R5	HA23
B7	RCA1	F2	$\overline{\text{R0}}$	M13	V_{SS}	R6	HA20
B8	RCA5	F3	V_{SS}	M14	LAD16	R7	HA19
B9	RCA9	F13	LAD24	M15	LAD18	R8	HA17
B10	RCA11	F14	LAD8	N1	$\overline{\text{SIZE16}}$	R9	HA16
B11	LAD31	F15	V_{SS}	N2	V_{CC}	R10	HA15
B12	LAD14	G1	$\overline{\text{HINT}}$	N3	CLKIN	R11	HA11
B13	V_{CC}	G2	$\overline{\text{HOE}}$	N4	V_{SS}	R12	HA9
B14	LAD13	G3	HDST	N5	HA29	R13	HA8
B15	LAD12	G13	LAD7	N6	HA25	R14	HBS3
C1	$\overline{\text{CAS0}}$	G14	V_{SS}	N7	HA21	R15	V_{SS}
C2	V_{CC}	G15	LAD23	N8	V_{SS}		
C3	$\overline{\text{DDOUT}}$	H1	LCLK1	N9	V_{SS}		
C4	DDIN	H2	EMU3	N10	HA12		
C5	V_{SS}	H3	LCLK2	N11	HA6		
C6	SF	H13	LAD22	N12	HBS2		
C7	RCA4	H14	LAD21	N13	HBS1		
C8	V_{SS}	H15	LAD6	N14	V_{CC}		

† This pin is provided for device orientation purpose only. Make no external connection.

Pin Assignments − 132-Pin Ceramic Quad Flatpack Package

PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME
1	$\overline{\text{CAS3}}$	34	$\overline{\text{HCS}}$	67	LAD0	100	LAD29
2	$\overline{\text{CAS2}}$	35	HA31	68	LAD16	101	LAD14
3	$\overline{\text{CAS1}}$	36	HA30	69	LAD1	102	LAD30
4	$\overline{\text{CAS0}}$	37	HA29	70	LAD17	103	LAD15
5	V_{CC}	38	HA28	71	LAD2	104	LAD31
6	$\overline{\text{RAS}}$	39	HA27	72	LAD18	105	SCLK
7	V_{SS}	40	HA26	73	V_{SS}	106	RCA12
8	$\overline{\text{R0}}$	41	HA25	74	LAD3	107	RCA11
9	$\overline{\text{R1}}$	42	HA24	75	LAD19	108	RCA10
10	$\overline{\text{HOE}}$	43	HA23	76	V_{CC}	109	RCA9
11	HDST	44	HA22	77	LAD4	110	RCA8
12	HRDY	45	HA21	78	LAD20	111	RCA7
13	$\overline{\text{HINT}}$	46	HA20	79	LAD5	112	RCA6
14	EMU3	47	HA19	80	LAD21	113	RCA5
15	LCLK1	48	HA18	81	LAD6	114	V_{CC}
16	LCLK2	49	HA17	82	LAD22	115	V_{SS}
17	EMU1	50	V_{SS}	83	LAD7	116	RCA4
18	EMU0	51	HA16	84	LAD23	117	RCA3
19	EMU2	52	HA15	85	V_{SS}	118	RCA2
20	GI	53	HA14	86	V_{SS}	119	RCA1
21	RESET	54	HA13	87	LAD8	120	RCA0
22	$\overline{\text{LINT2}}$	55	HA12	88	LAD24	121	SF
23	$\overline{\text{LINT1}}$	56	HA11	89	LAD9	122	TR/$\overline{\text{QE}}$
24	CAMD	57	HA10	90	LAD25	123	VSYNC
25	BUSFLT	58	HA9	91	LAD10	124	HSYNC
26	SIZE16	59	HA8	92	LAD26	125	$\overline{\text{CBLNK}}$ / $\overline{\text{VBLNK}}$
27	PGMD	60	HA7	93	LAD11	126	$\overline{\text{CSYNC}}$ / $\overline{\text{HBLNK}}$
28	LRDY	61	HA6	94	LAD27	127	V_{SS}
29	V_{CC}	62	HA5	95	V_{CC}	128	V_{SS}
30	VCLK	63	HBS3	96	LAD12	129	ALTCH
31	CLKIN	64	HBS2	97	LAD28	130	DDIN
32	HWRITE	65	HBS1	98	V_{SS}	131	DDOUT
33	HREAD	66	HBS0	99	LAD13	132	WE

Terminal Functions

TERMINAL NAME	TERMINAL TYPE†	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 available on LAD. A transparent latch (such as a 54ALS373) maintains the current address and status as long as $\overline{\text{ALTCH}}$ remains low.
BUSFLT	I	Bus fault. External logic asserts BUSFLT high to the SMJ34020A 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 LAD. In the 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. DDIN is used to drive the active-high output enables on bidirectional transceivers (such as the 54ALS623). The transceivers buffer data input and output on LAD0−LAD31 when the SMJ34020A is interfaced to several memories.
$\overline{\text{DDOUT}}$	O	Data bus direction output enable. $\overline{\text{DDOUT}}$ drives the active-low output enables on bidirectional transceivers (such as the 54ALS623). The transceivers buffer data input and output on LAD0−LAD31.
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 SMJ34020A system. LAD0 is the LSB and LAD31 is the MSB.
LRDY	I	Local ready. External circuitry drives LRDY low to inhibit the SMJ34020A from completing a local-memory cycle it has initiated. While LRDY remains low, the SMJ34020A waits unless the SMJ34020A loses bus priority or is given an external RETRY request (through BUSFLT). Wait states are generated in increments of one full LCLK1 cycle. LRDY can be driven low to extend local memory-read and memory-write cycles, VRAM serial-data-register-transfer cycles, and DRAM-refresh cycles. During internal cycles, the SMJ34020A ignores LRDY.
$\overline{\text{PGMD}}$	I	Page mode. The memory-decode logic asserts $\overline{\text{PGMD}}$ 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 can pull $\overline{\text{SIZE16}}$ 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 the emulation mode, $\overline{\text{SIZE16}}$ is used to select the size of mapped memory.
DRAM AND VRAM CONTROL		
CAMD	I	Column-address mode. CAMD 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	Four column-address strobes. $\overline{\text{CAS}}$ outputs drive the $\overline{\text{CAS}}$ inputs of DRAMs and VRAMs. $\overline{\text{CAS0}}$ − $\overline{\text{CAS3}}$ 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. $\overline{\text{RAS}}$ output drives the $\overline{\text{RAS}}$ inputs of DRAMs and VRAMs. $\overline{\text{RAS}}$ strobes the row address on RCA0−RCA12 to memory.
RCA0−RCA12	O	Thirteen 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 pin. SF is the special-function signal to 1M VRAMs that allows the use of block write, load write mask, load color mask, and write using write mask. SF is also used to differentiate instructions and addresses for the coprocessor as part of the coprocessor interface.
$\overline{\text{TR}}/\overline{\text{QE}}$	O	Transfer/output-enable. $\overline{\text{TR}}/\overline{\text{QE}}$ drives the $\overline{\text{TR}}/\overline{\text{QE}}$ input of VRAMs. During a local memory-read cycle, $\overline{\text{TR}}/\overline{\text{QE}}$ functions as an active-low output enable to gate from memory to LAD0−LAD31. During special VRAM function cycles, $\overline{\text{TR}}/\overline{\text{QE}}$ controls the type of cycle that is performed.

† I = input, O = output

Terminal Functions (Continued)

TERMINAL NAME	TERMINAL TYPE†	DESCRIPTION
DRAM AND VRAM CONTROL (CONTINUED)		
$\overline{WE}$	O	Write enable. The active low $\overline{WE}$ drives the $\overline{WE}$ inputs of DRAMs and VRAMs. $\overline{WE}$ can also be used as the active low write enable to static memories and other devices connected to the SMJ34020A local interface. During a local-memory read cycle, $\overline{WE}$ remains inactive high while $\overline{CAS}$ is strobed active low. During a local-memory write cycle, $\overline{WE}$ is strobed active low before $\overline{CAS}$. During VRAM serial-data-register transfer cycles, the state of $\overline{WE}$ at the falling edge of $\overline{RAS}$ controls the direction of the transfer.
HOST INTERFACE		
HA5−HA31	I	Twenty-seven host address input signals. A host can access a long word by placing the address on these lines. HA5−HA31 correspond to LAD5−LAD31 that output the address to the local memory.
HBS0−HBS3	I	Four host byte selects. HBS0−HBS3 identify which bytes within the long word are being selected.
$\overline{HCS}$	I	Host chip select. A host drives $\overline{HCS}$ low to latch the current host address present on HA5−HA31 and the host byte selects on HBS0−HBS3. $\overline{HCS}$ also enables host access cycles to the SMJ34020A I/O registers or local memory. During the low-to-high transition of $\overline{RESET}$, the level on $\overline{HCS}$ determines whether the SMJ34020A is halted ($\overline{HCS}$ is high for host-present mode) or whether it begins executing its reset service routine ($\overline{HCS}$ is low for self-bootstrap mode).
HDST	O	Host data-latch strobe. The rising edge of HDST latches data from the SMJ34020A local address space to the external host data latch on host read accesses. HDST can be used in conjunction with HRDY to indicate that data is valid in the external data latch.
$\overline{HINT}$	O	Host Interrupt. $\overline{HINT}$ allows the SMJ34020A to interrupt a host by setting the INTOUT bit in the HSTCTLL I/O register. $\overline{HINT}$ can also be used to interrupt the host if a BUSFLT or RETRY occurs due to a host access cycle.
$\overline{HOE}$	O	Host data latch output enable. $\overline{HOE}$ enables data from host data latches to the SMJ34020A local address space on host write cycles. $\overline{HOE}$ can be used in conjunction with HRDY to indicate data has been written to memory from the external data latch.
HRDY	O	Host ready. HRDY is normally low and goes high to indicate that the SMJ34020A is ready to complete a host-initiated read or write cycle. If the SMJ34020A is ready to accept the access request, HRDY is driven high and the host can proceed with the access. A host can use HRDY logically combined with HDST and $\overline{HOE}$ to determine when the local bus access cycles have completed.
$\overline{HREAD}$	I	Host read strobe. $\overline{HREAD}$ is driven low during a read request from a host processor. This notifies the SMJ34020A that the host is requesting access to the I/O registers or to local memory. $\overline{HREAD}$ should not be asserted at the same time that $\overline{HWRITE}$ is asserted.
$\overline{HWRITE}$	I	Host write strobe. $\overline{HWRITE}$ is driven low to indicate a write request by a host processor. This notifies the SMJ34020A that a write request is pending. The rising edge of HWRITE is used to indicate that the host has latched data to be written in the external data transceivers. $\overline{HWRITE}$ should not be asserted at the same time $\overline{HREAD}$ is asserted.
SYSTEM CONTROL		
CLKIN	I	Clock input. CLKIN generates LCLK1 and LCLK2, to which all processor functions in the SMJ34020A are synchronous. A separate asynchronous input clock (VCLK) controls the video timing and video registers.
LCLK1, LCLK2	O	Local output clocks. LCLK1 and LCLK2 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 SMJ34020A (except the CRT timing signals) are synchronous to LCLK1 and LCLK2.
$\overline{LINT1}$, $\overline{LINT2}$	I	Local interrupt requests. Interrupts from external devices are transmitted to the SMJ34020A on $\overline{LINT1}$ and $\overline{LINT2}$. 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. $\overline{LINT1}$, $\overline{LINT2}$ should remain low until the SMJ34020A recognizes it. $\overline{LINT1}$, $\overline{LINT2}$ can be applied asynchronously to the SMJ34020A as they are synchronized internally before use.
$\overline{RESET}$	I	System reset. During normal operation, $\overline{RESET}$ is driven low to reset the SMJ34020A. When $\overline{RESET}$ is asserted low, the SMJ34020A's internal registers are set to an initial known state and all output and bidirectional pins are driven either to inactive levels or to the high-impedance state. The SMJ34020A's behavior following reset depends on the level of the $\overline{HCS}$ input just before the low-to-high transition of $\overline{RESET}$. If $\overline{HCS}$ is low, the SMJ34020A begins executing the instructions pointed to by the reset vector. If $\overline{HCS}$ is high, the SMJ34020A is halted until a host processor writes a 0 to the HLT bit in the HSTCTLL register.

† I = input, O = output

Terminal Functions (Continued)

TERMINAL NAME	TERMINAL TYPE †	DESCRIPTION
POWER		
V_{CC} ‡	I	Nominal 5-V power supply inputs. Five pins on QFP; Nine pins on PGA.
V_{SS} ‡	I	Electrical ground inputs. Nine pins on QFP; 17 pins on PGA.
EMULATION CONTROL		
EMU0−EMU2	I	Emulation pins 0−2
EMU3	O	Emulation pin 3
MULTIPROCESSOR INTERFACE		
$\overline{GI}$	I	Bus grant input. External bus arbitration logic drives $\overline{GI}$ low to enable the SMJ34020A to gain access to the local-memory bus. The SMJ34020A must release the bus if $\overline{GI}$ is high so that another device can access the bus.
$\overline{R1}$, $\overline{R0}$	O	Bus request and control. $\overline{R1}$ and $\overline{R0}$ indicate a request for use of the bus in a multiprocessor system; they are decoded as shown below: $\overline{R1}$ $\overline{R0}$ **Bus Request Type** L L High-priority bus request L H Bus-cycle termination H L Low-priority bus request H H 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 the request as soon as possible by asserting $\overline{GI}$ 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 can 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.
VIDEO INTERFACE		
$\overline{CBLNK}$ / $\overline{VBLNK}$	O	Composite blanking/vertical blanking. $\overline{CBLNK}$ / $\overline{VBLNK}$ can be programmed 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, $\overline{CBLNK}$ / $\overline{VBLNK}$ is configured as a $\overline{CBLNK}$ output.
$\overline{CSYNC}$ / $\overline{HBLNK}$	I/O	Composite sync/horizontal blanking. $\overline{CSYNC}$ / $\overline{HBLNK}$ can be programmed 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: As an input, extracts $\overline{HSYNC}$ and $\overline{VSYNC}$ from externally generated horizontal sync pulses As an output, $\overline{CSYNC}$ / $\overline{HBLNK}$ generates active-low composite-sync pulses from either externally generated $\overline{HSYNC}$ and $\overline{VSYNC}$ signals or signals generated by the SMJ34020A's on-chip video timers Horizontal blank (output only) for blanking the display during horizontal retrace in separate-sync-video mode. Immediately following reset, $\overline{CSYNC}$ / $\overline{HBLNK}$ is configured as a $\overline{CSYNC}$ input.

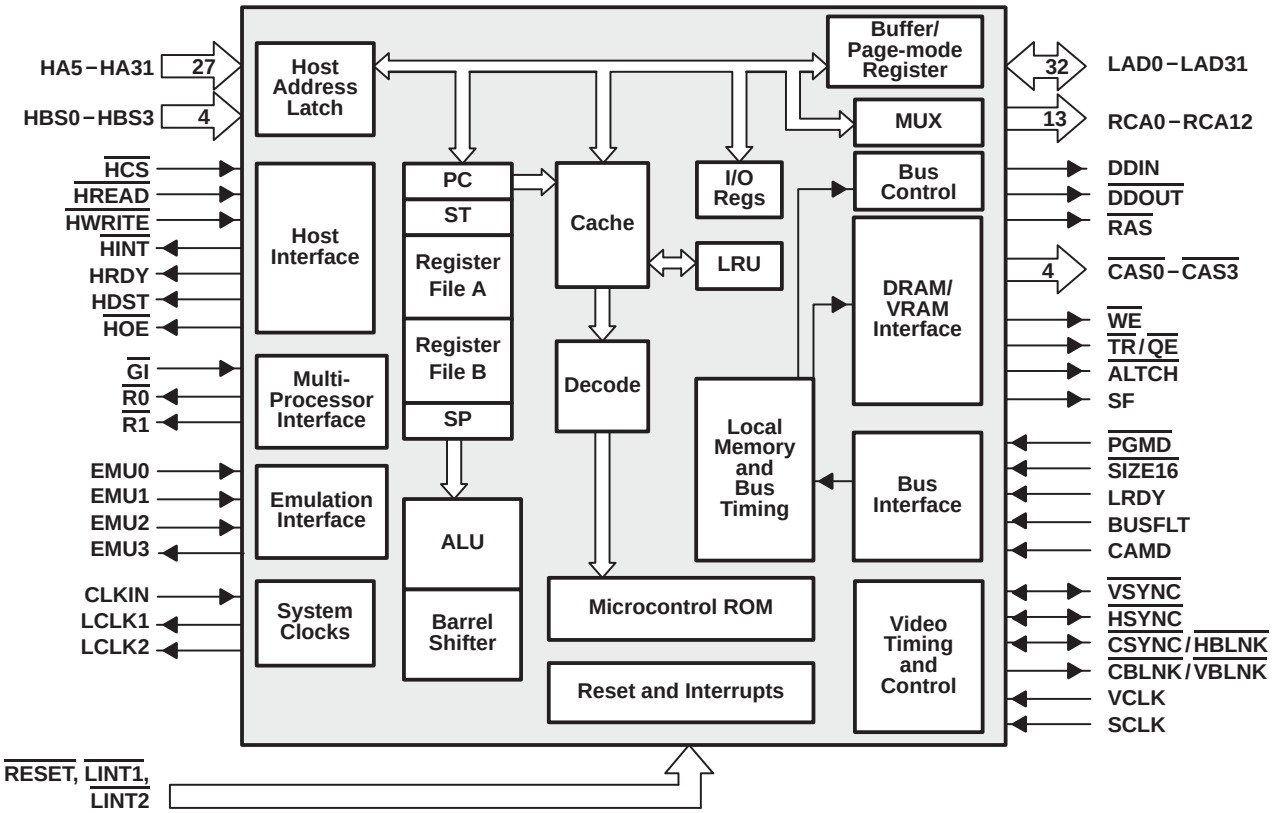
† I = input, O = output
‡ For proper SMJ34020A operation, all V_{CC} and V_{SS} pins must be connected externally.

Terminal Functions (Continued)

TERMINAL NAME	TERMINAL TYPE †	DESCRIPTION
VIDEO INTERFACE (CONTINUED)		
$\overline{\text{HSYNC}}$	I/O	Horizontal sync. $\overline{\text{HSYNC}}$ is the horizontal sync signal that controls external video circuitry. $\overline{\text{HSYNC}}$ can be programmed 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 SMJ34020A's on-chip video timers. As an input, $\overline{\text{HSYNC}}$ synchronizes the SMJ34020A 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. SCLK is the same as the signal that drives VRAM serial data registers. SCLK allows the SMJ34020A to track the VRAM serial-data-register count, providing serial-register transfer and midline-reload cycles. (SCLK can be asynchronous to VCLK; however, it typically has a frequency that is a multiple of the VCLK frequency).
VCLK	I	Video clock. VCLK is derived from a multiple of the video system's dot clock 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. $\overline{\text{VSYNC}}$ can be programmed 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 SMJ34020A's on-chip video timers. As an input, $\overline{\text{VSYNC}}$ synchronizes the SMJ34020A 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.

† I = input, O = output

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 SMJ34020A 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 using a complete set of field *move* instructions with selectable field sizes.

The 15 general-purpose registers in register file A are used for high-level language support and assembly-language programming. The 15 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 SMJ34020A'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 an associated present (P) flag to indicate whether or not the subsegment contains valid data.

The cache is loaded only when an instruction requested by the execution section of the SMJ34020A 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. During an interrupt, when the IX bit in the ST is placed on the stack, it indicates 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 SMJ34020A outputs a 28-bit address on LAD4−LAD31 that 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 SMJ34020A 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 SMJ34020A include bytes, words, long words, pixels, two independent fields of from 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) can 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 can 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 the 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 SMJ34020A 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 SMJ34020A contains an on-chip block of sixty-four 16-bit locations (mapped into the SMJ34020A'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

host interface registers

The host interface registers (HSTDATA, HSTADRL, HSTADRH, HSTCTLL, and HSTCTLH) are provided to facilitate communications between the SMJ34020A and a host processor and maintain compatibility with the SMJ34010. The registers are mapped into five of the I/O locations accessible to the SMJ34020A.

Two of these registers (HSTCTLL and HSTCTLH) are used to provide control by the host. This control consists of the passing of interrupt requests, flushing the instruction cache, halting the SMJ34020A, transmitting a non-maskable interrupt request to the SMJ34020A, enabling emulation interrupts, and setting host access modes and configurations.

The other three registers are simple read/write registers to allow the SMJ34020A software to leave addresses for the host at a known location and allow compatibility with some SMJ34010 software.

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 SMJ34020A can be configured to drive composite sync or separate sync displays.

In composite sync mode, the SMJ34020A 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 SMJ34020A 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

video timing and screen refresh (continued)

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 SMJ34020A directly supports 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 SMJ34020A supports various display resolutions and either interlaced or noninterlaced video. The SMJ34020A can optionally be programmed to synchronize to externally generated sync signals so that images created by the SMJ34020A can be superimposed upon images created externally. The external sync mode can also be used to synchronize the video signals generated by two or more SMJ34020As in a multiple-SMJ34020A graphics system.

CPU control registers

Five of the I/O registers (CONVDP, CONVMP, CONVSP, CONTROL, and PSIZE) provide CPU control to configure the SMJ34020A 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 SMJ34020A, 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 nonmaskable 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 SMJ34020A to execute instructions (that is, before the stack pointer is initialized). A dedicated terminal controls the SMJ34020A reset function.

memory controller/local-memory interface

The memory controller manages the SMJ34020A'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 those 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 SMJ34020A forced to defer execution of the subsequent instruction.

The SMJ34020A directly interfaces to standard DRAMs and in particular, to standard video RAMs (VRAMs) such as the SMJ44C25x multiport VRAMs. The SMJ34020A 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 LAD. 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-$\overline{\text{RAS}}$ (CBR) refresh cycles.

memory controller/local-memory interface (continued)

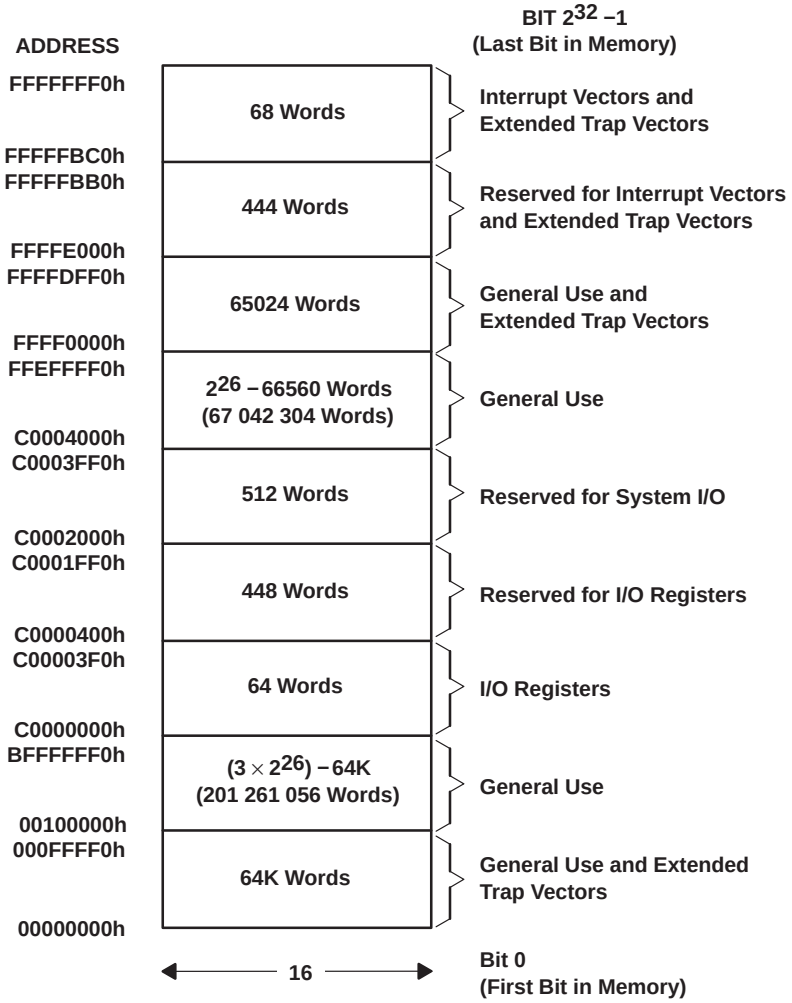


Figure 1. Memory Map

reset

Reset puts the SMJ34020A into a known initial state. This state is entered when the input signal at $\overline{\text{RESET}}$ 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.

The state of the $\overline{\text{HCS}}$ input on the CLKIN cycle before the low-to-high transition of $\overline{\text{RESET}}$ determines whether the SMJ34020A is halted or begins executing instructions. The SMJ34020A can be in one of two modes, host-present or self-bootstrap mode.

Host-present mode: if $\overline{\text{HCS}}$ is high at the end of reset, SMJ34020A instruction execution halts and remains halted until the host clears the HLT (halt) bit in HSTCTLH (host control register). Following reset, the $\overline{\text{RAS}}$ cycles required to initialize the dynamic RAMs are performed automatically by the GSP memory control logic. The host can request a memory access after the eight $\overline{\text{RAS}}$ initialization cycles have completed. The SMJ34020A automatically performs DRAM refresh cycles at regular intervals although the SMJ34020A remains halted until the host clears the HLT bit. Only then does SMJ34020A fetch the level-0 vector address from location FFFFFFE0h and begin executing the reset service routine.

Self-bootstrap mode: if $\overline{\text{HCS}}$ is low at the end of reset, the SMJ34020A first performs eight refresh cycles to initialize the DRAMs. Immediately following the eight refresh cycles, the GSP fetches the level-0 vector address from location FFFFFFE0h and begins executing the reset service routine.

At the time the SMJ34020A 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 nonmaskable interrupts if the NMIM bit in HSTCTLH is set to a 1) because the value of the stack pointer just before a reset is generally not valid. 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 $\overline{\text{RESET}}$ to its active-low level. To reset the SMJ34020A at powerup, $\overline{\text{RESET}}$ 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 powerup, the SMJ34020A can be reset by holding $\overline{\text{RESET}}$ low for a minimum of four local clock periods; the GSP enters 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 SMJ34020As 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 can be done by using CLKIN to clock the $\overline{\text{RESET}}$ as received by the SMJ34020As. All SMJ34020As 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 powerup, 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 SMJ34020A to a particular quarter cycle by adding Q1 cycles. All SMJ34020As 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 SMJ34020A 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 $\overline{\text{RAS}}$ be driven inactive-high for 1 millisecond 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 $t-(10t_Q)$ microseconds, t_Q being the quarter-cycle time as defined by the input clock period, $t_{c(CHI)}$. The SMJ34020A memory controller automatically inserts the required eight $\overline{\text{RAS}}$ cycles after all resets (after powerup 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 SMJ34020A is set to perform a refresh sequence every eight cycles.

At times other than powerup, 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 SMJ34020A to refresh the memories. On reset, the SMJ34020A is set to do a refresh cycle every eight local clock periods. A 30-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 $\overline{\text{RESET}}$ remains low longer than the maximum refresh interval specified for the memory, the previous contents of the local memory can not be valid after the reset.

initial state following reset

While $\overline{\text{RESET}}$ is asserted low (or while in the internal reset state), the SMJ34020A's output and bidirectional pins are forced to the states in Table 1.

Table 1. Initial State of Pins Following a Reset (With $\overline{\text{GI}}$ Low)†

OUTPUTS DRIVEN HIGH	OUTPUTS DRIVEN LOW	BIDIRECTIONALS DRIVEN TO HIGH IMPEDANCE
$\overline{\text{RAS}}$	HRDY	$\overline{\text{VSYNC}}$
$\overline{\text{CAS0}}-\overline{\text{CAS3}}$	$\overline{\text{CBLNK}}/\overline{\text{VBLNK}}$	$\overline{\text{HSYNC}}$
$\overline{\text{WE}}$	DDIN	$\overline{\text{CSYNC}}/\overline{\text{HBLNK}}$
$\overline{\text{TR}}/\overline{\text{QE}}$		LAD0−LAD31
$\overline{\text{DDOUT}}$		
$\overline{\text{ALTCH}}$		
$\overline{\text{HINT}}$		
$\overline{\text{R0}}$		
$\overline{\text{R1}}$		
$\overline{\text{HOE}}$		
HDST		
EMU3		
RCA0−RCA12		
SF		

† If $\overline{\text{GI}}$ is high, then all $\overline{\text{GI}}$-controlled pins are high-impedance. $\overline{\text{GI}}$-controlled pins are $\overline{\text{RAS}}$, $\overline{\text{CAS0}}-\overline{\text{CAS3}}$, $\overline{\text{WE}}$, $\overline{\text{TR}}/\overline{\text{QE}}$, $\overline{\text{DDOUT}}$, 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 1 if $\overline{\text{HCS}}$ is high just prior to the low-to-high transition of $\overline{\text{RESET}}$; otherwise, it is set to 0.

reset and DRAM/VRAM initialization (continued)

Just prior to the execution of the first instruction in the reset routine, the SMJ34020A's internal registers are in the following states:

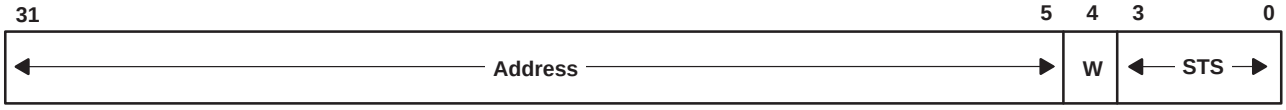
- General-purpose register files A and B are uninitialized.
- The ST is set to 0000 0010h.
- The PC contains the most-significant 28 bits of the vector fetched from memory address FFFF FFE0h (the least significant four bits of the PC are set to zero).
- The BEN bit in the I/O register CONFIG is set to 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) can 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 SMJ34020A local memory interface consists of an address/data multiplexed bus on which addresses 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 SMJ34020A 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 LAD while the ROW address is output on the row/column address (RCA) bus. See Figure 2. $\overline{\text{ALTCH}}$ and $\overline{\text{RAS}}$ are used to latch the address/status and ROW address, respectively, on these two buses. LAD is then used to transfer data to or from the memory while the RCA bus is set to the column address for the memory. (LAD31 is the most significant bit of the address or data).



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

Figure 2. LAD During the Address Cycle

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 column-address mode (CAMD) during each memory cycle (see Table 2). The CAMD 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.

local memory and DRAM/VRAM interface (continued)

Table 2. 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 can be supported within the basic VRAM mode

Table 3 lists the actual logical address bits output on each of the RCA lines during row and column intervals for each of the four VRAM modes and states of CAMD.

Table 3. Logical Address Bit Output

ROW TIME					COLUMN TIME				
					CAMD = 0	CAMD = 1			
RCA BIT	64K	256K	1M	4M		64K	256K	1M	4M
12	24	25	26	27	16	23	26	15	28
11	23	24	25	26	15	22	14	14	14
10	22	23	24	25	14	13	13	13	13
9	21	22	23	24	13	12	12	12	12
8	20	21	22	23	12	11	11	11	11
7	19	20	21	22	11	10	10	10	10
6	18	19	20	21	10	9	9	9	9
5	17	18	19	20	9	8	8	8	8
4	16	17	18	19	8	7	7	7	7
3	15	16	17	18	7	6	6	6	6
2	14	15	16	17	6	5	5	5	5
1	13	14	15	16	5	4	4	4	4
0	12	13	14	15	4	4	4	4	16

In the 64K mode with CAMD=0, any eight adjacent RCA0−RCA12 pins output 16 contiguous logical address bits. The eight most significant addresses are output during row-address time while the least significant addresses are output during column-address time. Logical addresses 12 through 16 are output twice during a memory cycle (during both $\overline{RAS}$ and $\overline{CAS}$ falling edges) but at different pins. This allows a variety of VRAM memory organizations and decoding schemes to be used. When CAMD = 1, the addresses output during column-address time are changed such that a new logical address mapping occurs, allowing connection of RCA directly to 256K or 1M DRAMs.

local memory and DRAM/VRAM interface (continued)

Similarly, for each of the other VRAM modes, direct connection is provided for other DRAM modes requiring larger matrices than the configuration mode. Table 4 gives examples of the connections using this feature.

Table 4. Connections to RCA for CAMD = 1

RCA	64K†		256K†		1M†	4M
12		1M × 32		4M × 32	4M × 32	16M × 32
11		1M × 16	1M × 32	4M × 32	4M × NN	16M × 32
10	256K × 32	1M × 32	1M × NN	4M × 32	4M × NN	16M × 32
9	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
8	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
7	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
6	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
5	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
4	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
3	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
2	256K × NN	1M × NN	1M × NN	4M × 32	4M × NN	16M × 32
1	256K × 16	1M × 16	1M × 16		4M × 16	
0						16M × 32

† NN is used for either 16-bit (× 16) or 32-bit (× 32) memory connections.

status codes

Status codes are output on LAD0−LAD3 at the time of the falling edge of $\overline{\text{ALTCH}}$ and can be used to determine the type of cycle that is being initiated. Table 5 lists the codes and their respective meanings.

Table 5. Status Codes Output on LAD0−LAD3

CODE	STATUS	TYPE
0000	Coprocessor code	
0001	Emulator operation	OTHER
0010	Host cycle	(00XX)
0011	DRAM refresh	
0100	Video-generated DRAM serial register transfer	
0101	CPU-generated VRAM serial register transfer	VRAM
0110	Write mask load	(01XX)
0111	Color latch load	
1000	Data access	
1001	Cache fill	
1010	Instruction fetch	
1011	Interrupt vector fetch	CPU
1100	Bus locked operation	(1XXX)
1101	Pixel operation	
1110	Block write	
1111	− RESERVED −	

dynamic bus sizing

The SMJ34020A 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 SMJ34020A then performs another memory access to the next 16-bit (odd) word in memory. The SMJ34020A samples $\overline{\text{SIZE16}}$ at the start of Q2 in the second cycle (access to odd word address) to determine to which half of LAD the port or memory is aligned. If the port is on LAD0−LAD15, $\overline{\text{SIZE16}}$ should be low during the second cycle access (odd word); otherwise, if the port is on LAD16−LAD31, $\overline{\text{SIZE16}}$ must be high at this time. The SMJ34020A always performs two memory cycles to access the 16-bit wide memories, even when attempting only a 16-bit transfer.

The SMJ34020A 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 LAD 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 LAD 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 transfer the most significant word first, then the least significant word.

The second memory cycle forced by $\overline{\text{SIZE16}}$ is performed as a page mode access if $\overline{\text{PGMD}}$ was low during the first access. A read-write cycle to the 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 LAD the memory is connected (low for connection to LAD0−LAD15 or high for connection to LAD16−LAD31).

special 1-M VRAM cycles

The SMJ34020A provides control for special function VRAM cycles that are available in the 1-M devices. These cycles are obtained by the appropriate timing control of SF, $\overline{\text{CAS}}$, $\overline{\text{TR}}/\overline{\text{QE}}$, and $\overline{\text{WE}}$ of the VRAMs at the falling edge of $\overline{\text{RAS}}$. 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 can 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 bus grant in $\overline{\text{GI}}$ and the priority request signals $\overline{\text{R0}}$ and $\overline{\text{R1}}$ 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 SMJ34020A to another. Synchronization of multiple SMJ34020As in a system occurs at reset with the rising edge of $\overline{\text{RESET}}$ meeting the setup and hold requirements to CLKIN, so all SMJ34020As are certain to respond to $\overline{\text{RESET}}$ during the same quarter cycle. $\overline{\text{RESET}}$ is not required to be synchronous to CLKIN except to allow synchronization of multiple SMJ34020As in a system.

multiprocessor arbitration (continued)

The $\overline{GI}$ priority for multiprocessing environments is determined by arbitration logic external to the SMJ34020A. If $\overline{GI}$ goes inactive-high, the SMJ34020A releases the bus on the next available cycle boundary. If the cycle in progress has not successfully completed, the SMJ34020A restarts the cycle upon regaining control of the bus. Normally, if the SMJ34020A asserts both $\overline{R0}$ and $\overline{R1}$ low, it should be given the control of the bus by the arbitrator.

host interface

The SMJ34020A host interface allows the local memory to be mapped into the host address space. The SMJ34020A acts as a DRAM controller for the host. The address for the host access is latched within the SMJ34020A; however, the data for the access is transferred using external transceivers. The host selects the address of a 32-bit long word for an access using the 27 host address lines, HA5−HA31. If the host desires byte addressability, it can select the active bytes for the access by using HBS0−HBS3. The SMJ34020A always reads 32 bits from memory; however, on host writes, it uses the host byte selects to enable $\overline{CAS0}$−$\overline{CAS3}$ to memory. The address and byte selects are latched at the falling edge of $\overline{HCS}$ within the SMJ34020A. The host indicates a read or write by asserting $\overline{HREAD}$ or $\overline{HWRITE}$ (as appropriate) either before or after $\overline{HCS}$. (Note that $\overline{HREAD}$ and $\overline{HWRITE}$ must never be asserted at the same time.)

The SMJ34020A responds to a host read request by latching the requested data in the external latches and providing HRDY to the host, indicating that the read cycle is completing. The rising edge of HDST with HRDY high indicates data is latched in the external transceivers.

The host indicates that a write to a particular location is required by providing the address and asserting $\overline{HWRITE}$. The host must maintain both $\overline{HCS}$ and $\overline{HWRITE}$ asserted until valid data is in the transceivers. (The rising edge of $\overline{HOE}$ with HRDY high indicates that the data previously stored in the external transceivers has been written to memory.) Typically, the rising edge of $\overline{HWRITE}$ is used to strobe the data into the latches and signal the SMJ34020A that the write access can start. The SMJ34020A uses its byte-write capability to write only to the selected bytes.

The SMJ34020A always accesses the required location as latched at the falling edge of $\overline{HCS}$; however, in order to increase the data rate, a look ahead mechanism is implemented. The host increment enable (HINC) and host prefetch after write enable (HPFW) bits in the host control register (HSTCTLH) must be appropriately set to make optimum use of this feature. These bits provide four modes of operation as indicated in Table 6.

Table 6. Modes of Operation

HINC	HPFW	HOST ACCESS MODE	DESCRIPTION
0	0	Random/Same	No increment, no prefetch
0	1	Random/Same	No increment, no prefetch
1	0	Block	Increment after read or write, prefetch after read
1	1	Read-Modify-Write	Increment after write, prefetch after write

When the SMJ34020A is programmed for block mode or read-modify-write accesses, the host can still do random accesses because the SMJ34020A always uses the address provided at the falling edge of $\overline{HCS}$; however, there is a prefetch to the next sequential address. The prefetch occurs after reads in block mode and after writes in read-modify-write mode. The SMJ34020A compares the address latched by $\overline{HCS}$ on host reads to see if it is the same as that of the last prefetched data. If the addresses match, data is not re-accessed but HRDY is set high to indicate that the data is presently available.

dynamic bus sizing on host accesses

If the host makes a read access to a 16-bit wide memory, the SMJ34020A automatically does the second cycle required to read the rest of the 32-bit word (even if the host did not require a 32-bit cycle). The external logic must comprehend the sense of $\overline{\text{SIZE16}}$ or the $\overline{\text{CAS}}$ strobes during the accesses in order to route the data into the proper external host data transceivers. The SMJ34020A uses the host byte selects $\overline{\text{HBS0}}$−$\overline{\text{HBS3}}$ to enable the $\overline{\text{CAS}}$ strobes when doing a host write.

coprocessor interface

Support for coprocessors is provided through special instructions and bus cycles that allow communication with the coprocessor. A coprocessor can be register based, depending on the SMJ34020A to do all address calculations, or it can operate as its own bus controller, using the multiprocessor arbitration scheme. Five basic cycles are provided for direct communication and control of coprocessors:

- SMJ34020A to coprocessor
- Coprocessor to SMJ34020A
- Move memory to coprocessor
- Move coprocessor to memory
- 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 can be sent to the coprocessor and operated upon without an explicit coprocessor command cycle.

instruction set

The SMJ34020A instruction set can be divided into five categories:

- Graphics instructions
- Coprocessor instructions
- Move instructions
- General-purpose instructions
- Program control and context switching

Specialized graphics instructions manipulate pixel data that is accessed using memory addresses or XY coordinates. These instructions include graphics operations, such as array and raster operations, 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 instructions allow the user to control flow and to save and restore information using instructions with both register-direct and absolute operands.

clock stretch

The SMJ34020A supports a clock stretching mechanism.

With advances in semiconductor manufacturing, newer versions of the SMJ34020A can be made, each supporting a higher CLKIN frequency. The increase in CLKIN frequency means that the SMJ34020A 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.

clock stretch (continued)

The clock-stretch mechanism allows the SMJ34020A 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 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 memory devices than the designer could use if no stretch mechanism was 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, Q4a, and Q4b.

When clock-stretch mode is enabled, the fourth machine quarter cycle can be stretched to twice its original length. See Figure 3 for an example. This stretching takes place only when the SMJ34020A attempts certain types of memory cycles.

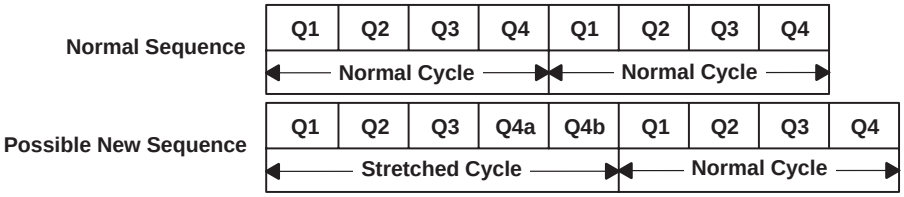
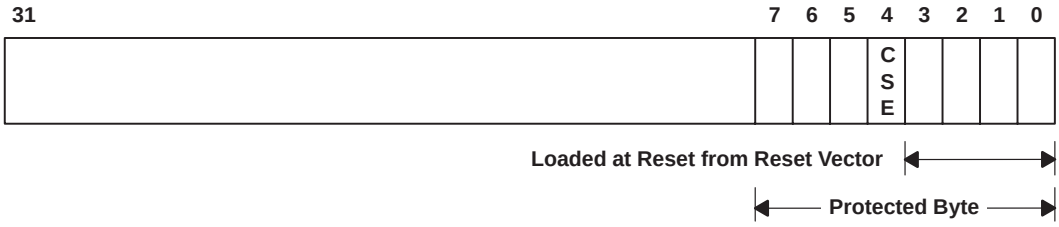


Figure 3. Stretched Machine Quarter Cycle

The stretch is achieved by holding the internal SMJ34020A clocks in the Q4 state for an extra quarter cycle so all of the device outputs remain unchanged during Q4a and Q4b. The SMJ34020A 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 using a bit in the CONFIG register memory mapped to location C00001A0h, see Figure 4.



CONFIG register

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

Figure 4. Stretch Mode Enable

Bit 4 of the CONFIG register is the clock-stretch-enable mode bit. A zero in this bit disables stretch mode and a one in this bit enables stretch mode. The bit is cleared during reset; that is, stretch mode is disabled by default.

enabling clock stretch (continued)

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

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

Notes:

a) The host default cycle shown in the *TMS34020 User's Guide* is not stretched because it is not a true address cycle; that is, $\overline{RAS}$, etc., do not go low.
b) The CPU default cycle, which is similar to the host default cycle in that $\overline{RAS}$, etc., do not go low, 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 SMJ34020A is running with the CSE bit set to 1.

The advantage of this implementation of clock-stretch mode is that the SMJ34020A can execute code at maximum speed, slowing down only 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 SMJ34020A (with the exception of the video logic, which is clocked independently by VCLK) effectively slows down during such a stretched cycle.

Figure 5 through Figure 8 show examples of stretch-mode memory operations.

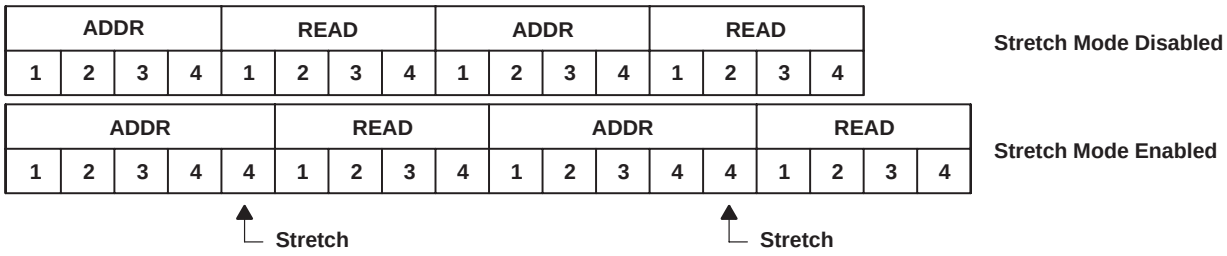


Figure 5. Two 32-Bit Nonpage-Mode Reads

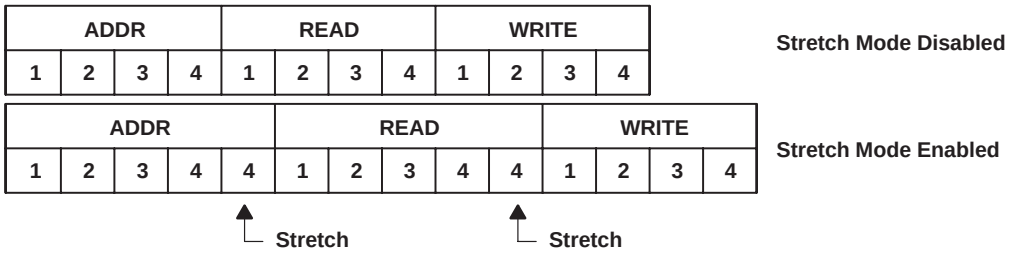


Figure 6. One 32-Bit Page-Mode Read-Modify-Write

enabling clock stretch (continued)

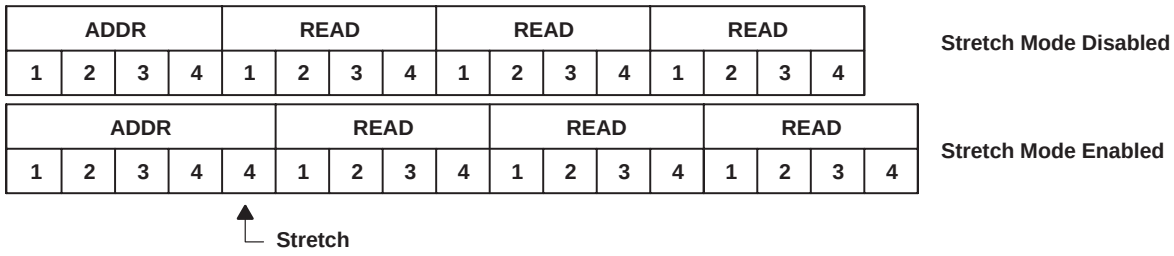


Figure 7. Three 32-Bit Page-Mode Reads

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 can be generated. For example:

Figure 8. 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.

clock-stretch timing example, SMJ34020A-32 and 150-ns DRAMs

This example analyzes a memory interface timing parameter. It shows that the clock-stretch mechanism can be used to allow the SMJ34020A-32 to avoid a timing violation when interfaced to 100-ns VRAMs.

Consider a system with:

- A SMJ34020A-32, which has a 32-MHz clock input frequency and hence a 125-ns cycle time, so t_Q = 31 ns. Timing parameters are taken from this data sheet.
- A SMJ44C251-10 1 megabit × 1 bit DRAM. Timing parameters are taken from the corresponding Texas Instruments data sheet.

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

Without clock stretch

SMJ4C1024	$t_{h(RA)}$	Hold time, row address valid after $\overline{RAS}$ low	Min = 20 ns
SMJ34020A	Parameter 88	Hold time, row address valid after $\overline{RAS}$ low	Min = t_Q − 5 ns = 26 ns

If $\overline{RAS}$ is passed through a PAL™ with a delay of 7 ns, then $t_{h(RA)}$ seen by the DRAM is 26 ns − 7 ns = 19 ns. This violates the 20 ns minimum.

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

With clock stretch

SMJ34020A	Parameter 88 $t_{h(ADV-REL)}$	Hold time, row address valid after $\overline{RAS}$ low	Min = $2t_Q$ − 5 ns = 57 ns

With the same 7-ns PAL delay, the DRAM sees $t_{h(RA)}$ as 57 ns − 7ns = 50 ns, which does not violate the 20 ns minimum.

cycle timing examples

The following figures show examples of many of the basic cycles that the SMJ34020A uses for memory access, VRAM control, multiprocessor bus control, and coprocessor communication. These figures should not be used to determine specific signal timings, but can be used to see signal relationships for the various cycles. The Q4 phases that *could* be stretched are marked with an * on the diagrams. The conditions required for the stretch are:

- The design uses a SMJ34020A.
- The CONFIG register's CSE bit is set to 1.
- The SMJ34020A is doing either:
  a) Any address cycle, or
  b) A read data cycle in a read-modify-write sequence

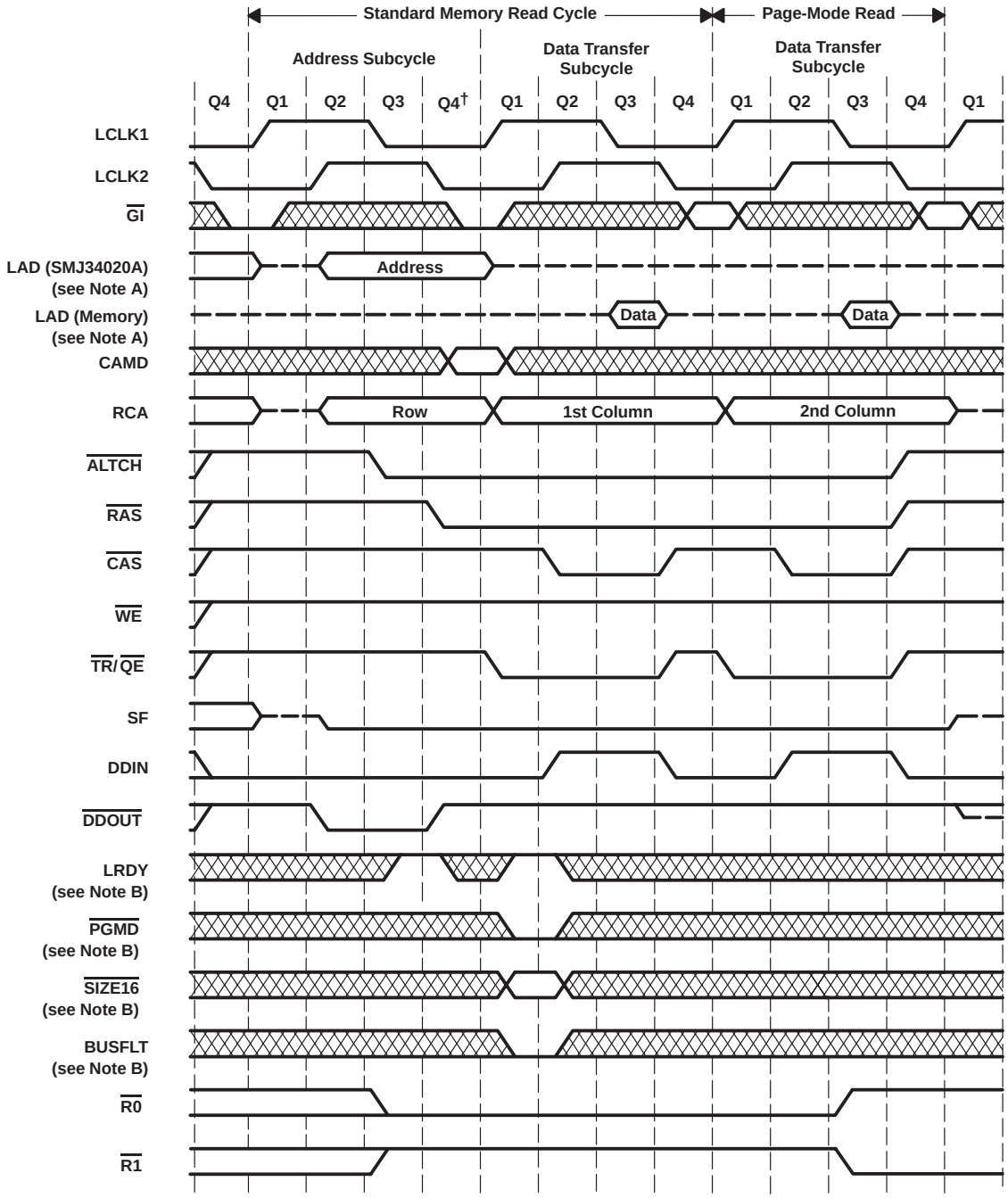
The following remarks apply to memory timing in general. A row address is output on RCA0−RCA12 at the start of a cycle along with the full address and status on LAD0−LAD31. These remain valid until after the fall of $\overline{ALTCH}$ and $\overline{RAS}$. The column address is then output on RCA0−RCA12, 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 can require external buffering of the address and data lines. DDIN and $\overline{DDOUT}$ coordinate these external buffers with LAD.

During the address output to LAD by the SMJ34020A (Figure 9), the least significant four bits (LAD0−LAD3) contain a bus-status code. $\overline{PGMD}$ low at the start of Q2 after $\overline{RAS}$ low indicates that this memory supports page-mode operation. LRDY high at the start of Q2 after $\overline{RAS}$ low indicates that the cycle can continue without inserting wait states. $\overline{DDOUT}$ returns high after the initial address output on LAD (during Q4), indicating that a memory read cycle is about to take place.

PAL is a trademark of Advanced Micro Devices, Inc.

cycle timing examples (continued)



† See clock stretch, page 21.

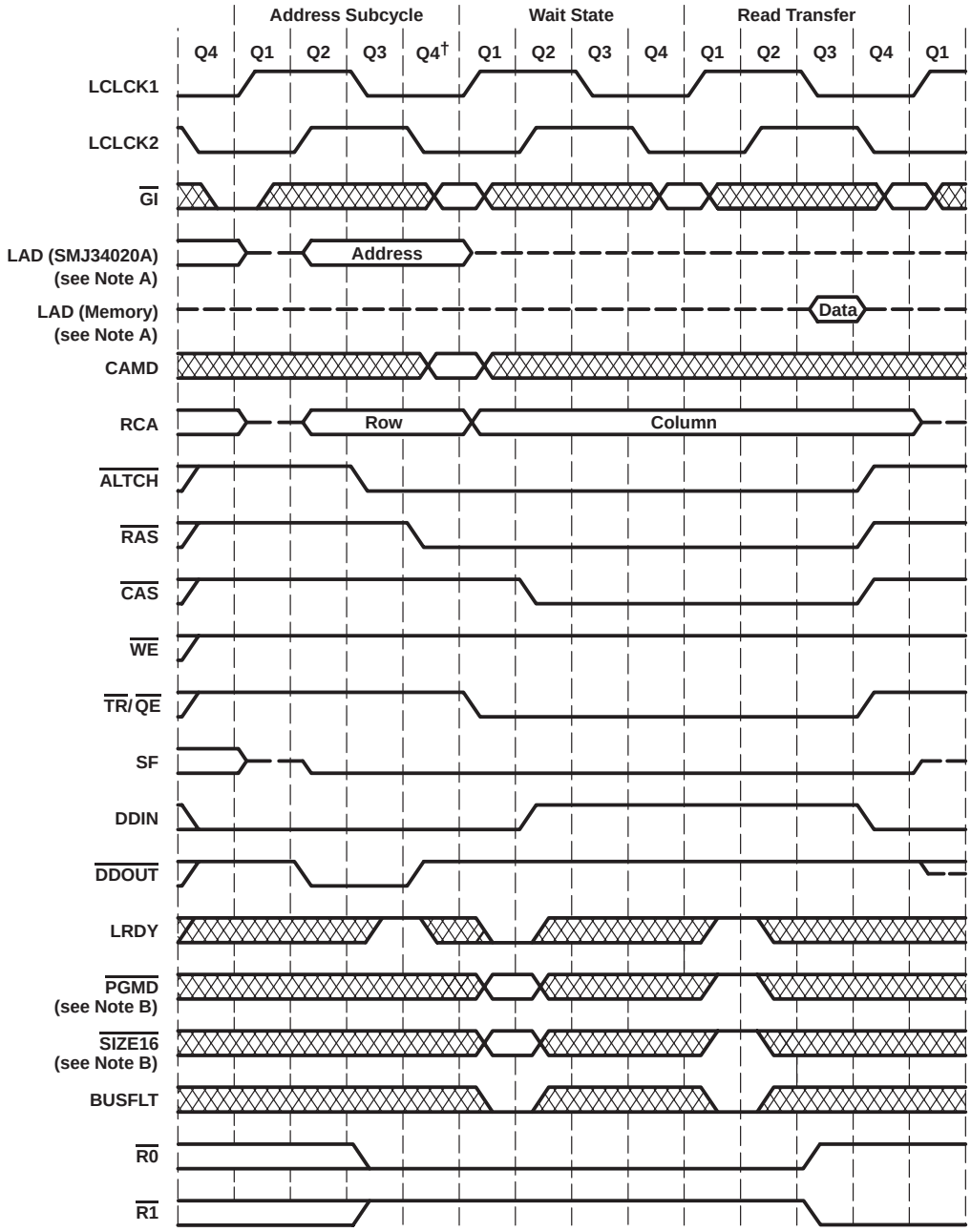
NOTES: A. LAD (SMJ34020A): Output to LAD by the SMJ34020A
LAD (memory): Output to LAD by the memory.

B. LRDY, PGMD, SIZE16, and BUSFLT are not sampled on subsequent page-mode cycle accesses to 32-bit-wide memory space.

Figure 9. Local-Memory Read-Cycle Timing (With Page Mode)

cycle timing examples (continued)

LRDY low at the start of the first Q2 after $\overline{\text{RAS}}$ low (Figure 10) indicates that the memory requires the addition of wait states. LRDY high at the next Q2 indicates the cycle can 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 21.

NOTES: A. LAD (SMJ34020A): Output to LAD by the SMJ34020A
LAD (memory): Output to LAD 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 LRDY is sampled high.

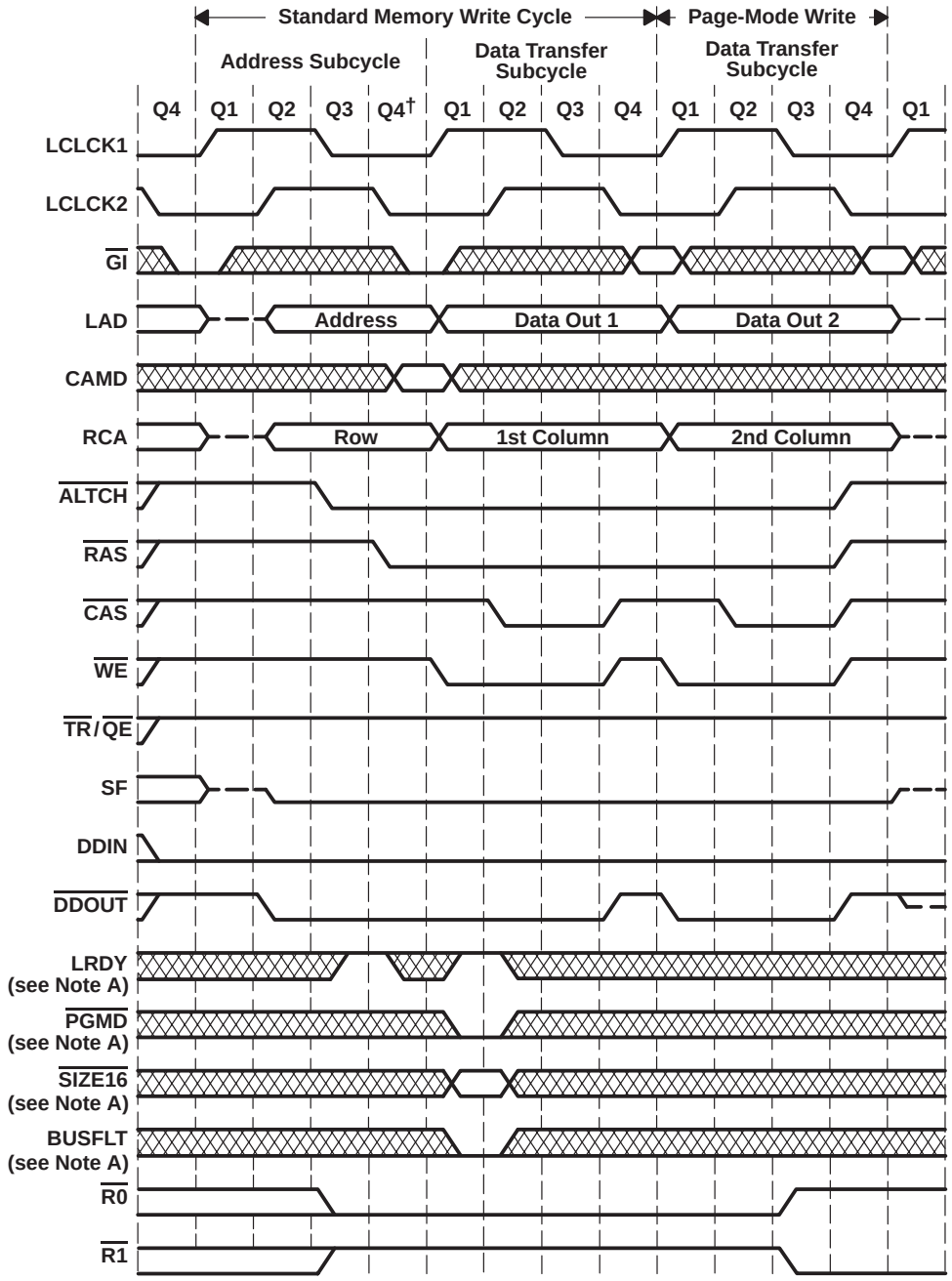
Figure 10. Local-Memory Read-Cycle Timing (Without Page Mode, With One Wait State)

cycle timing examples (continued)

During the address output to LAD by the SMJ34020A (Figure 11), the least significant four bits (LAD0−LAD3) contain a bus-status code. $\overline{\text{PGMD}}$ low at the start of Q2 after $\overline{\text{RAS}}$ low indicates that this memory supports page-mode operation. LRDY high at the start of Q2 after $\overline{\text{RAS}}$ low indicates that the cycle can continue without inserting wait states.

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

cycle timing examples (continued)



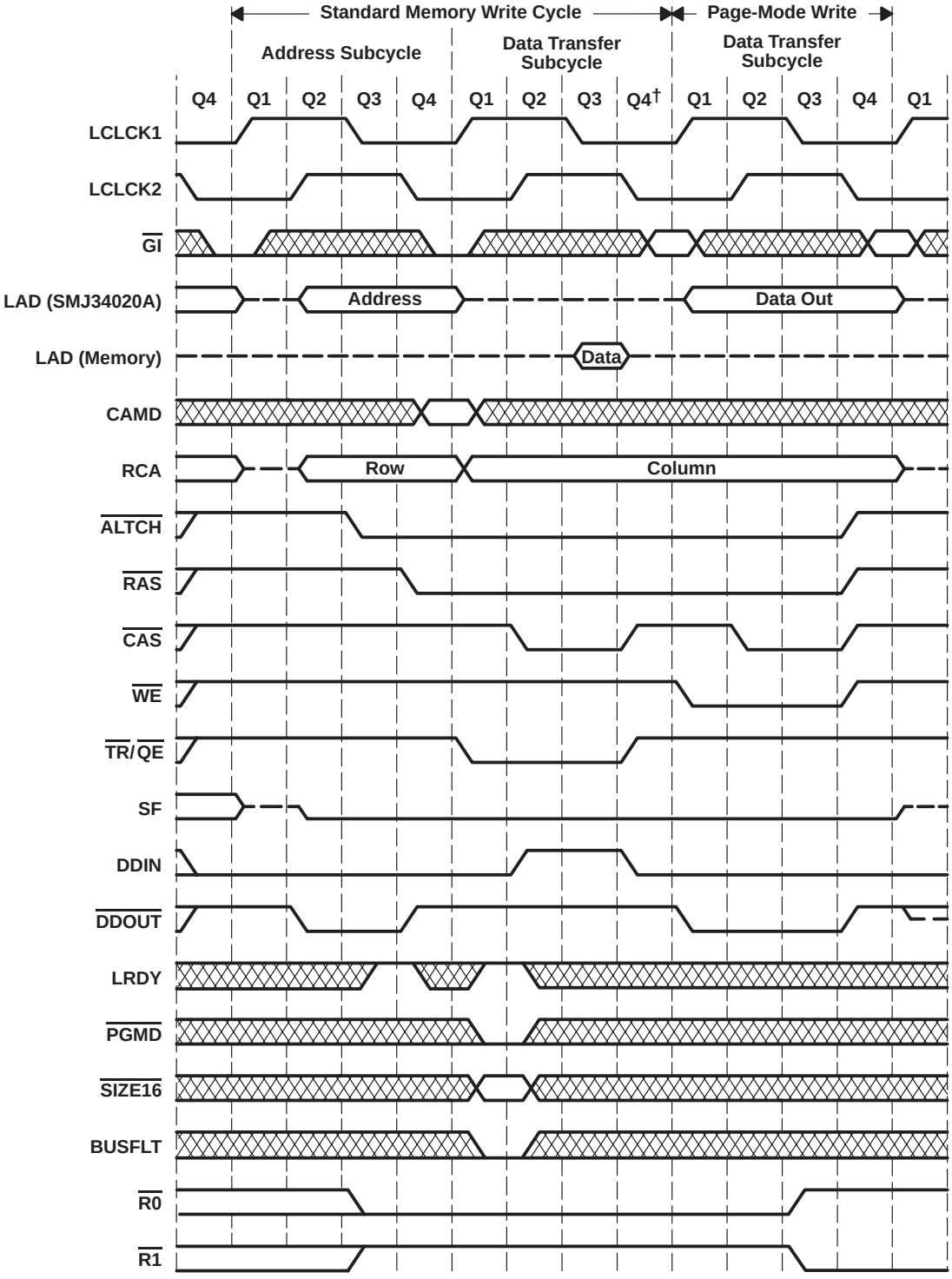
† See clock stretch, page 21.

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

Figure 11. Local-Memory Write Cycle Timing (With Page Mode)

cycle timing examples (continued)

The local memory read-modify-write cycle (Figure 12) 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 21.

Figure 12. Local-Memory Read-Modify-Write Cycle Timing

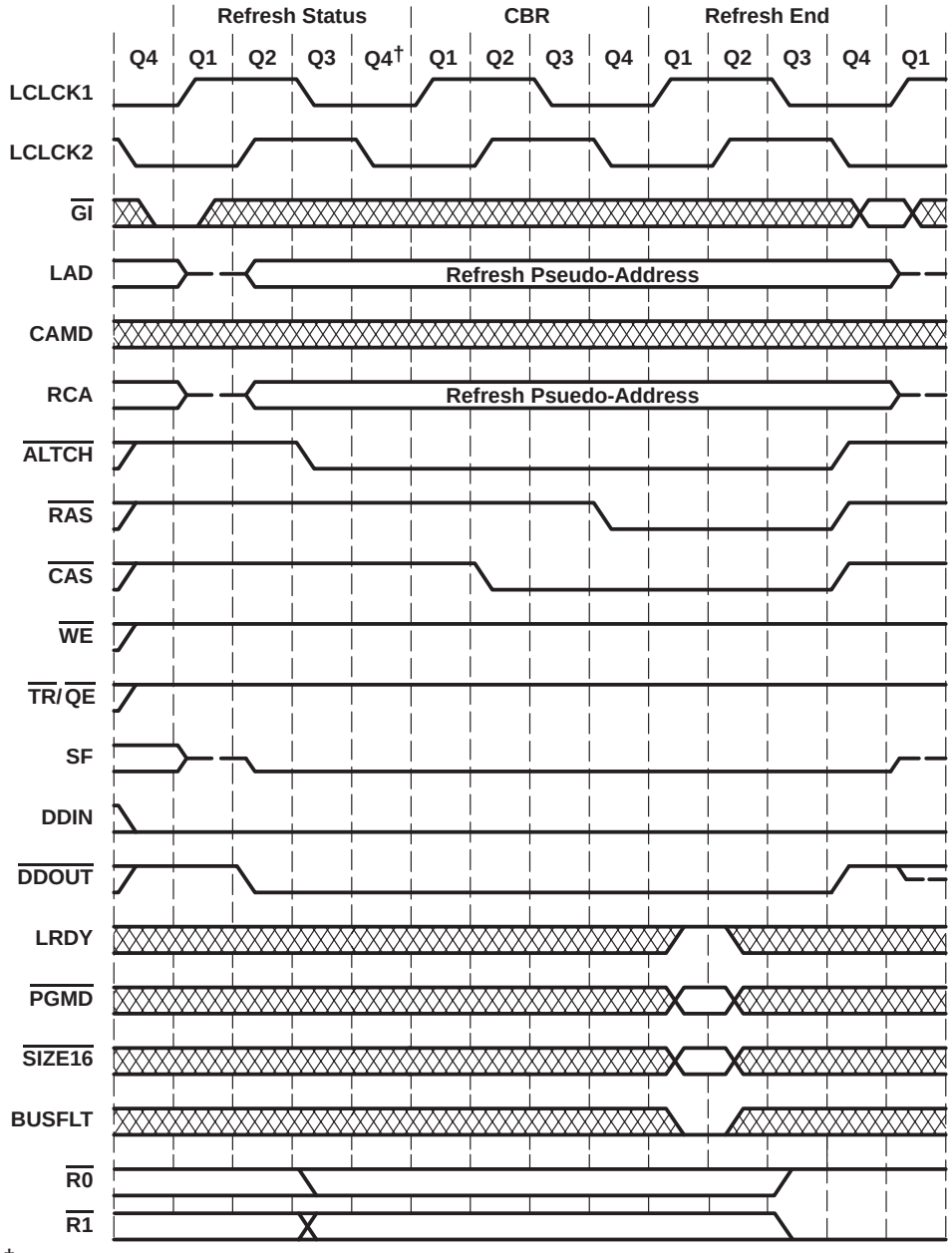
cycle timing examples (continued)

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 (Figure 13). The 16 bits of address are placed on LAD16−LAD31; all other LAD bus lines are 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 is not decremented and the same pseudo-address is reissued when the refresh is restarted.

cycle timing examples (continued)



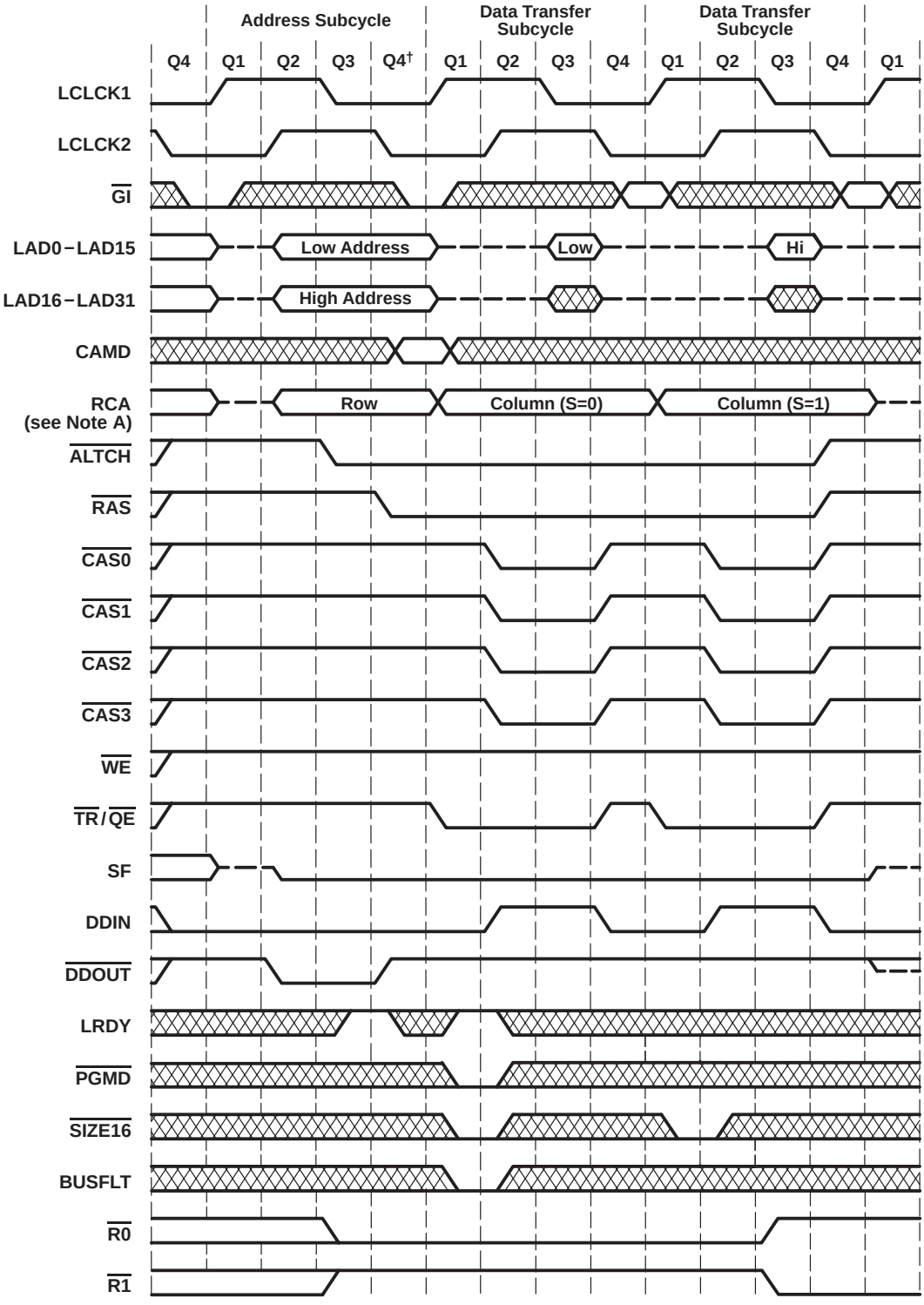
† See clock stretch, page 21.

Figure 13. Refresh Cycle Timing

cycle timing examples (continued)

When $\overline{\text{SIZE16}}$ is selected low (Figure 14), the SMJ34020A 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 SMJ34020A 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.

cycle timing examples (continued)



† See clock stretch, page 21.

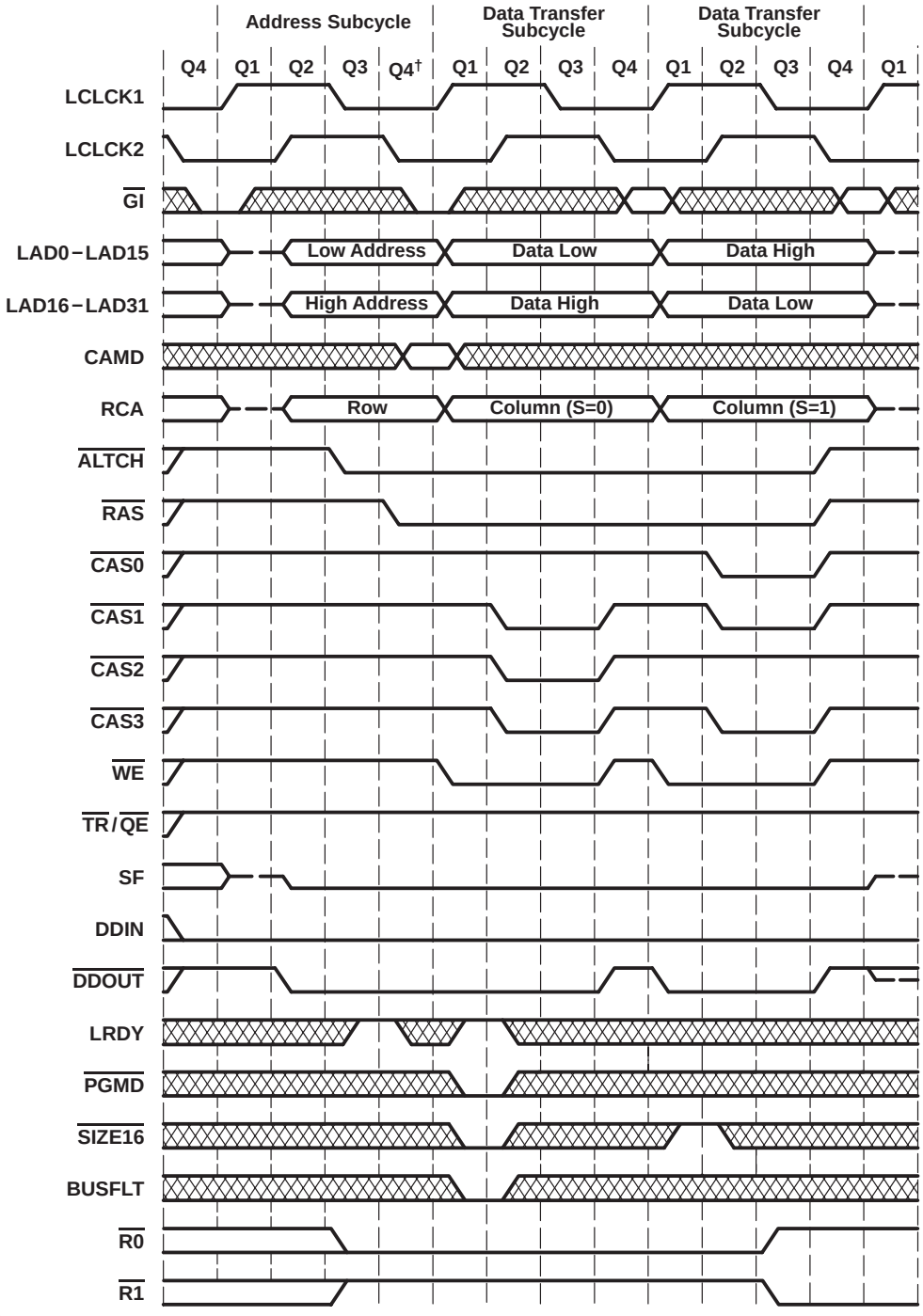
NOTE A: RCA0 can be used to determine accesses to odd or even words because it outputs the least significant bit of the word address during the column-address time (except in 4-M mode with CAMD = 1).

Figure 14. Dynamic Bus Sizing, Read Cycle Timing

cycle timing examples (continued)

Write accesses to 16-bit memory are performed by swapping the data on upper and lower words of LAD and exchanging data on $\overline{\text{CAS0}}$ and $\overline{\text{CAS1}}$ for data on $\overline{\text{CAS2}}$ and $\overline{\text{CAS3}}$, respectively (Figure 15). During the first cycle, data is placed on LAD0−LAD31 as in a normal write. The sampling of $\overline{\text{SIZE16}}$ low during the first access indicates that this is 16-bit-wide memory, so the SMJ34020A swaps data on the upper and lower halves of LAD. Notice that during the first cycle, $\overline{\text{CAS0}}$ is inactive (because this byte was not selected), and during the second cycle, $\overline{\text{CAS2}}$ is inactive due to the exchange of $\overline{\text{CAS0}}$ for $\overline{\text{CAS2}}$ and $\overline{\text{CAS1}}$ for $\overline{\text{CAS3}}$.

cycle timing examples (continued)

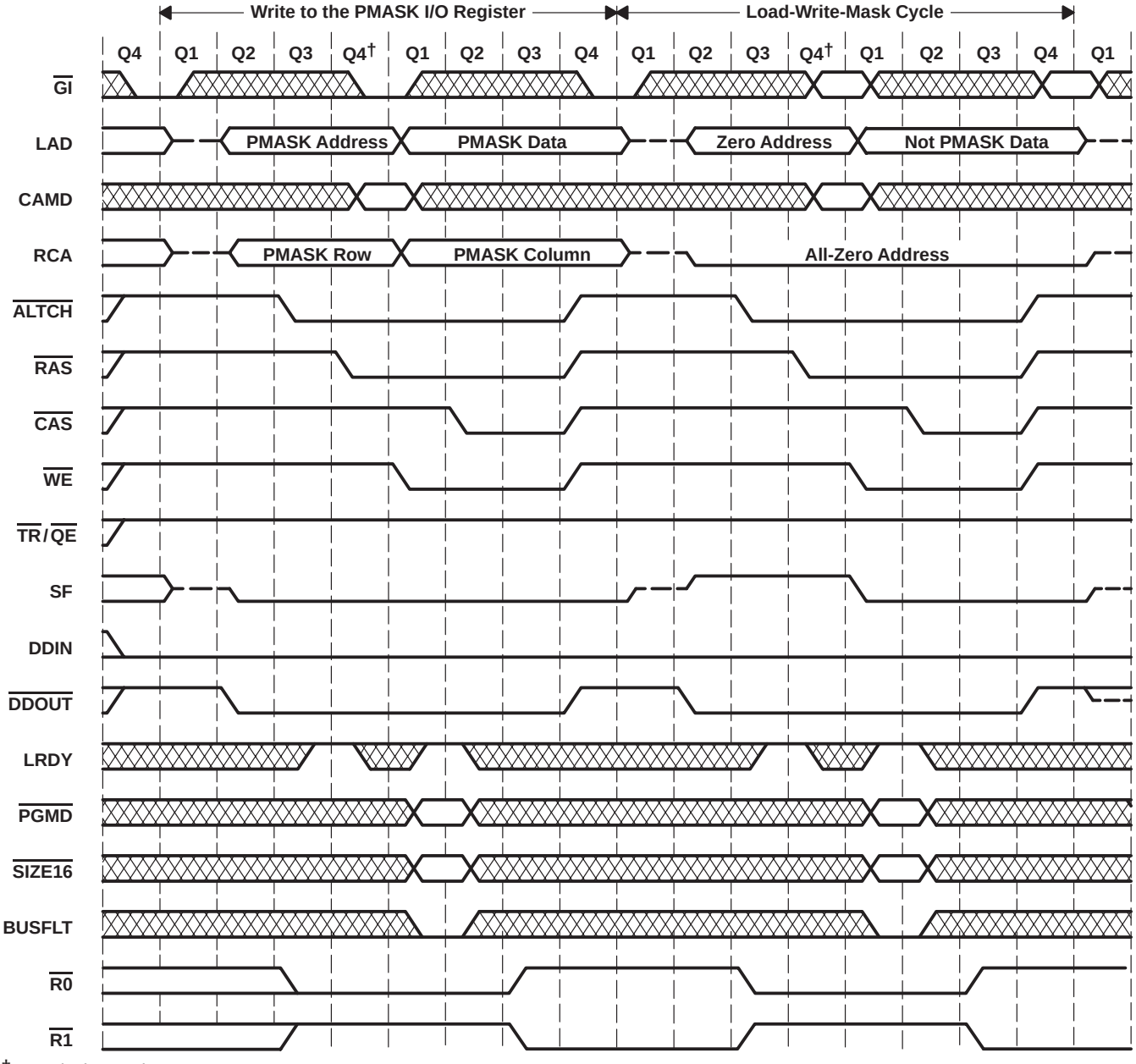


† See clock stretch, page 21.

Figure 15. Dynamic Bus Sizing, Write-Cycle Timing

cycle timing examples (continued)

Clock stretch is a special 1-megabit VRAM control cycle that is executed when VEN in the CONFIG I/O register is set and PMASKL and/or PMASKH are written (Figure 16). This cycle is indicated by $\overline{CAS}$, $\overline{WE}$, $\overline{TR}/\overline{QE}$, 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 LAD 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.

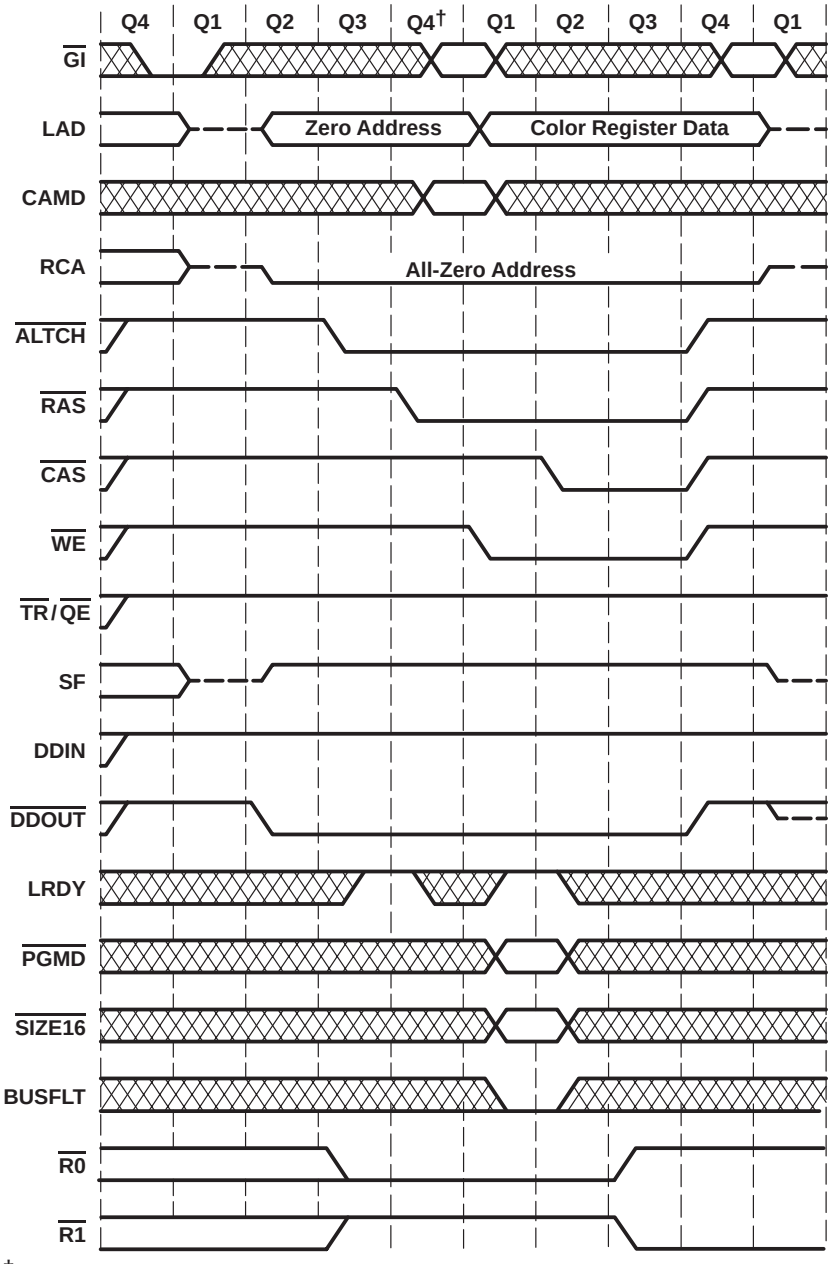


† See clock stretch, page 21.

Figure 16. Load-Write-Mask-Cycle Timing

cycle timing examples (continued)

The clock stretch is generated by the VLCOL instruction and is indicated by $\overline{CAS}$, $\overline{WE}$, $\overline{TR}/\overline{QE}$, and SF high at the falling edge of $\overline{RAS}$ and SF high at the falling edge of $\overline{CAS}$ (Figure 17). 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.

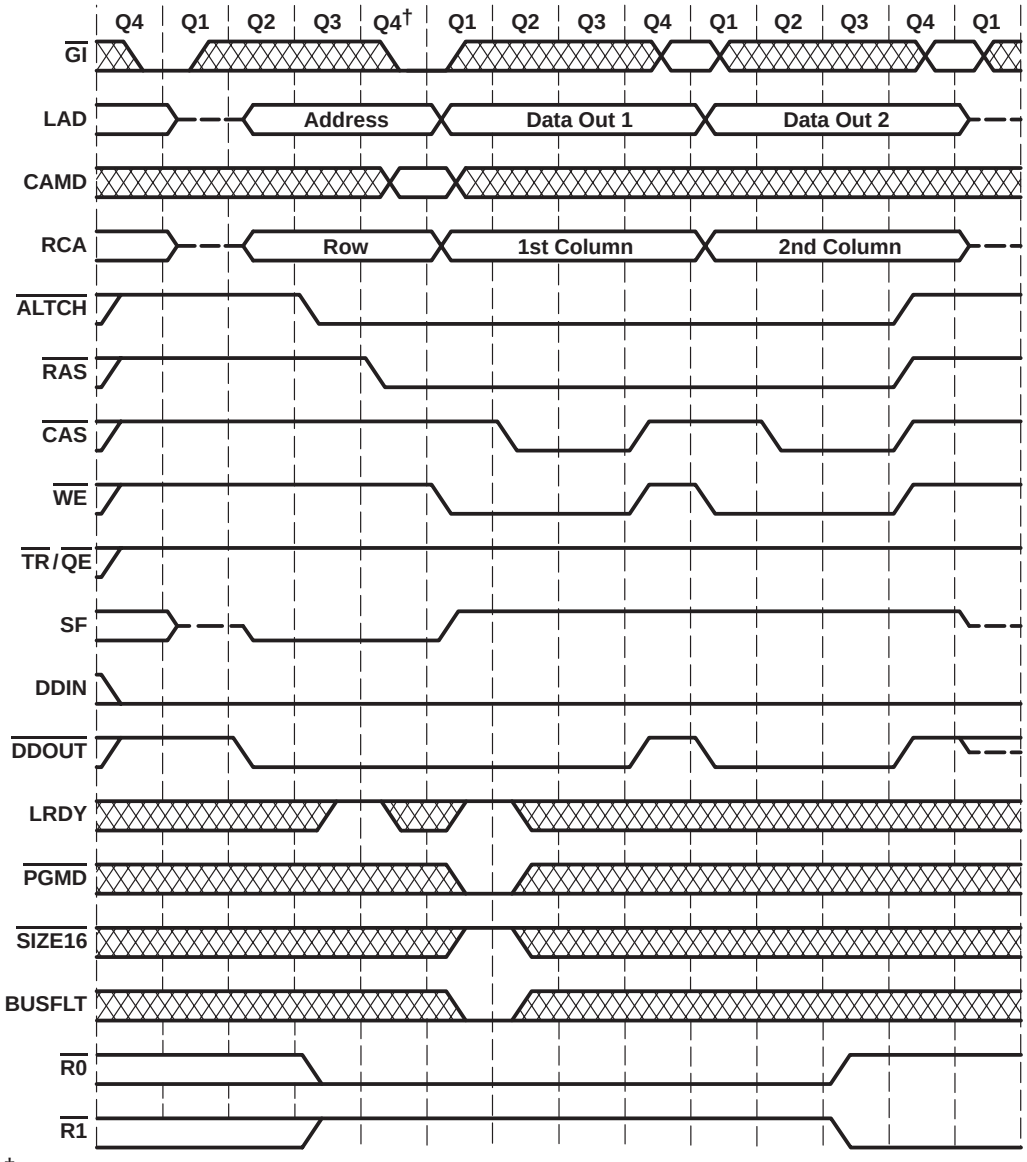


† See clock stretch, page 21.

Figure 17. Load-Color-Latch-Cycle Timing

cycle timing examples (continued)

The clock stretch is also performed when a VBLT or VFILL instruction is executed and PMASKL and PMASKH are set to zero (Figure 18). It is indicated by $\overline{CAS}$, $\overline{WE}$, $\overline{TR}/\overline{QE}$ 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.

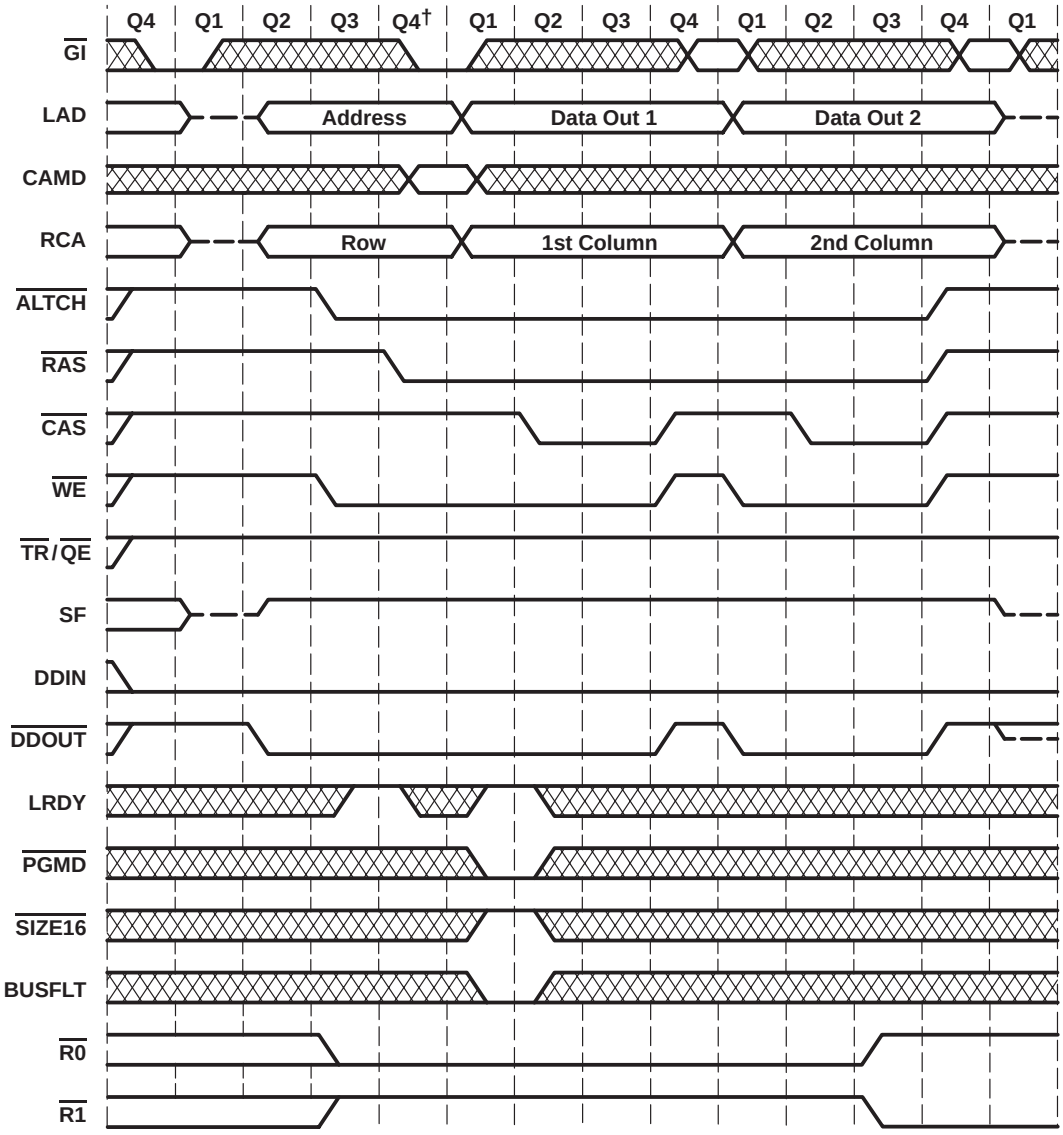


† See clock stretch, page 21.

Figure 18. Block-Write-Cycle Timing (Without Mask)

cycle timing examples (continued)

The clock stretch is also performed when a VBLT or VFILL instruction is executed and PMASKL and PMASKH are set to nonzero values (Figure 19). It is indicated by $\overline{CAS}$, $\overline{TR}/\overline{QE}$, 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 the block-write cycle 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.

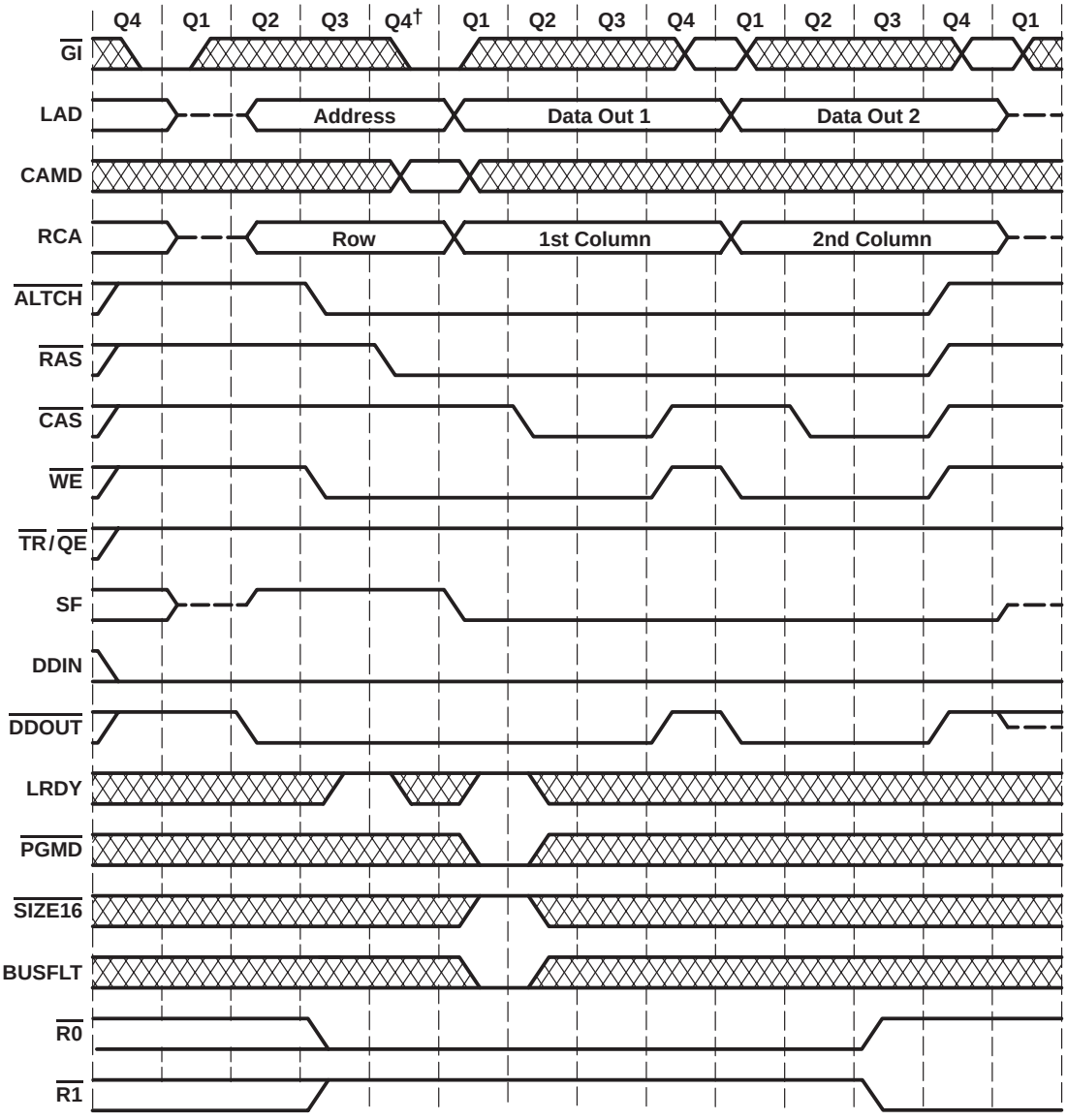


† See clock stretch, page 21.

Figure 19. Block-Write-Cycle Timing (With Mask)

cycle timing examples (continued)

As a special 1-megabit VRAM control cycle, the clock strech is also performed when the PMASKL and PMASKH registers are set to nonzero values, CST in DPYCTL is cleared, VEN in CONFIG is set, and the byte-aligned pixel-write instruction is executed (Figure 20). This cycle is indicated by $\overline{CAS}$, $\overline{TR}/\overline{QE}$, 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 DQs that are written to memory. During the address portion of the cycle, the status on LAD0−LAD3 indicates that a pixel operation is being performed (status code = 1101).



† See clock stretch, page 21.

Figure 20. Write-Cycle Timing Using Mask

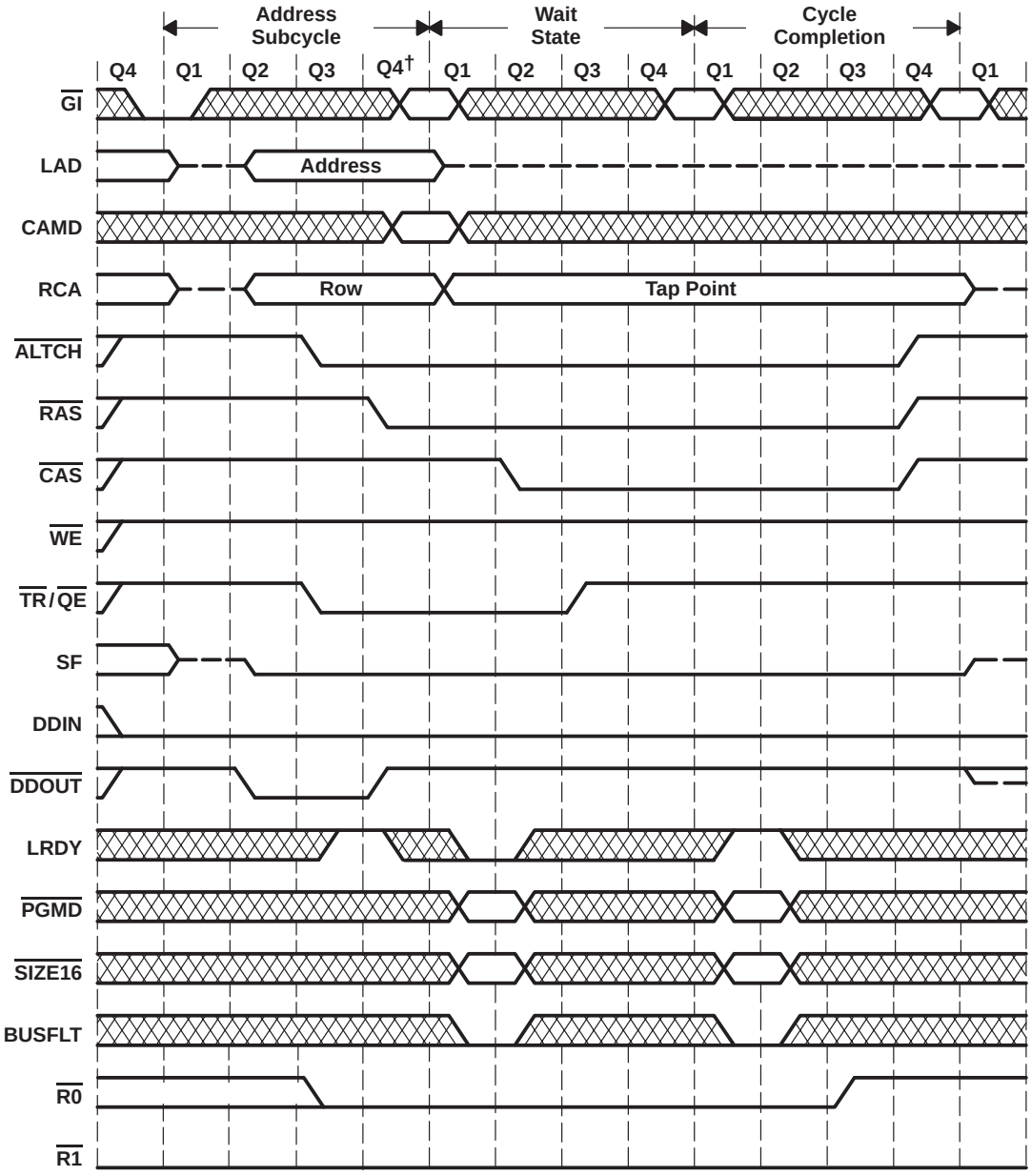
cycle timing examples (continued)

The VRAM cycle shown in Figure 21 is issued in any of three ways:

- Pixel operation instruction with CST in DPYCTL set
- Horizontal blank reload cycle requested by the video-control logic with VCE in DPYCTL cleared
- Video timeout due to SCOUNT match with the value in MLRNXT and VCE and SSV 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 midline 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).

cycle timing examples (continued)

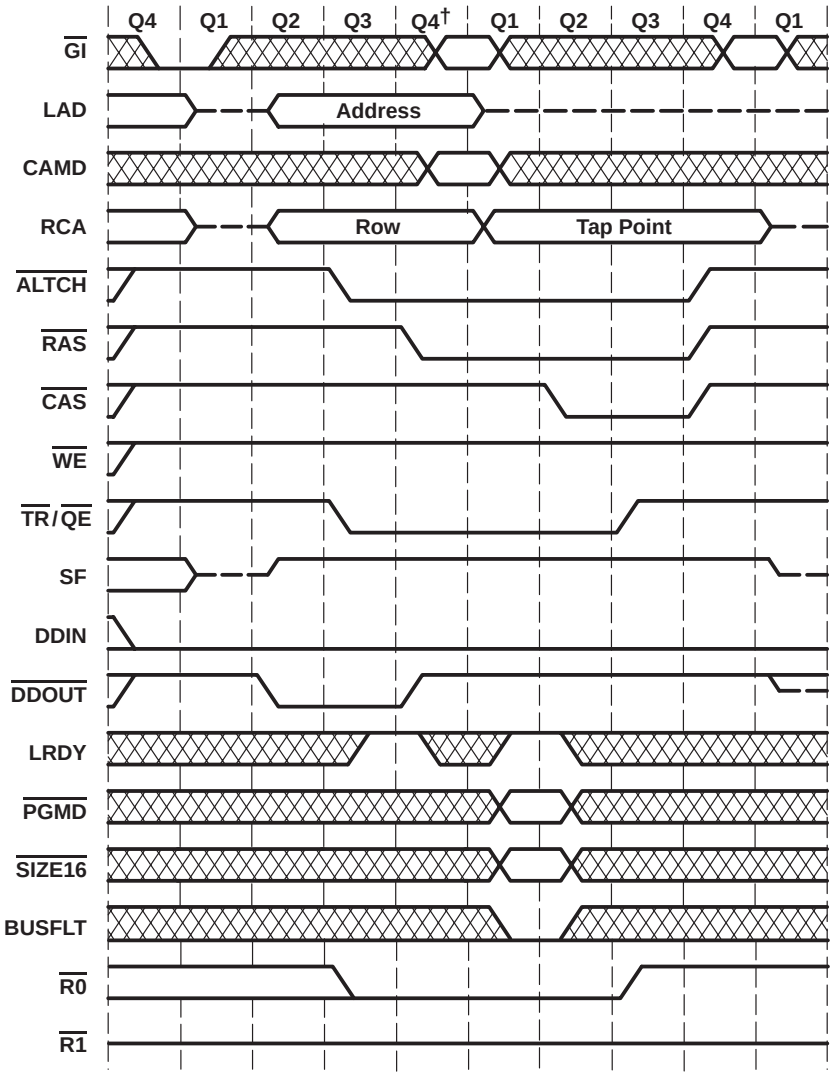


† See clock stretch, page 21.

Figure 21. Memory-to-Serial-Data-Register-Cycle Timing (VRAM Read Transfer)

cycle timing examples (continued)

This VRAM cycle shown in Figure 22 is performed when a video timeout occurs due to a match of the MLRNXT register, VCE in DPYCTL is cleared, and SSV 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 midline 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.

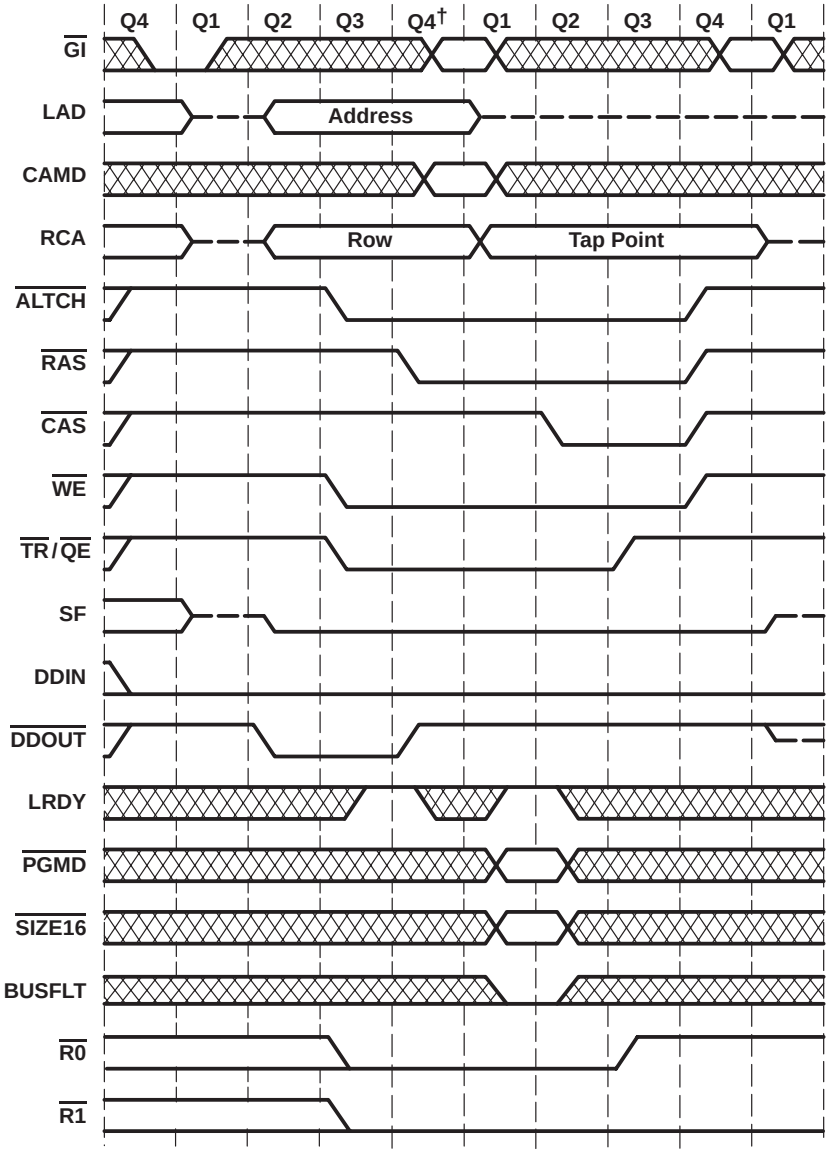


† See clock stretch, page 21.

Figure 22. Memory-to-Split-Serial-Data-Register-Cycle Timing (VRAM Split-Register Read Transfer)

cycle timing examples (continued)

Figure 23 shows the VRAM cycle performed when a horizontal blank reload is requested by the video-control logic and VCE and SRE 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 SMJ34020A). 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.

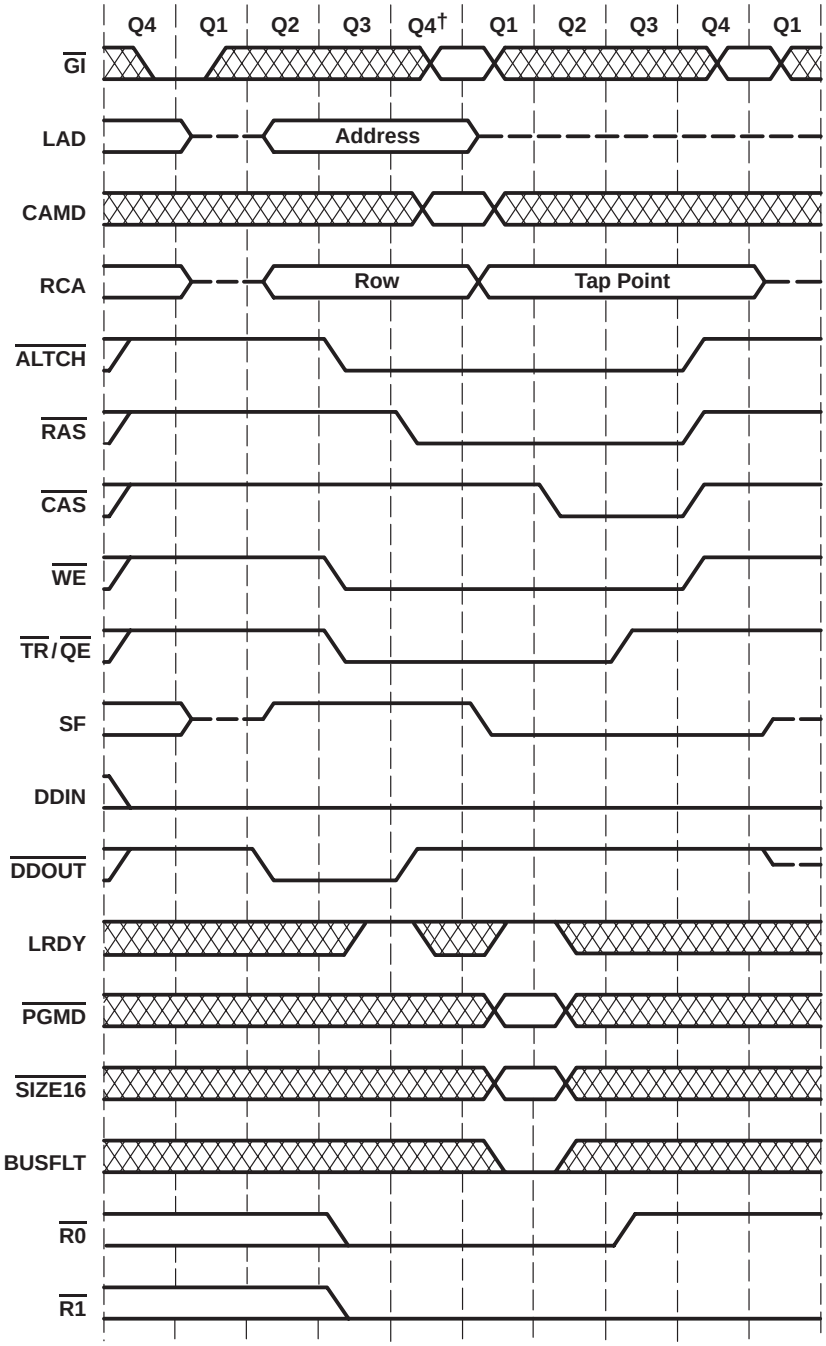


† See clock stretch, page 21.

Figure 23. Serial-Data-Register-to-Memory-Cycle Timing (VRAM-Write Transfer, Pseudo-Write Transfer)

cycle timing examples (continued)

This VRAM cycle (Figure 24) is performed when a pixel-write instruction is executed with the CST bit in DPYCTL set. This cycle is indicated by $\overline{TR}/\overline{QE}$ 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 $\overline{SOE}$ 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.



† See clock stretch, page 21.

Figure 24. Serial-Data-Register-to-Memory Cycle Timing (VRAM-Alternate-Write Transfer)

cycle timing examples (continued)

In Figure 25, transition points are shown for $\overline{R0}$ and $\overline{R1}$ to indicate where they occur relative to the other signals.

This example indicates that the SMJ34020A has control of the bus, yields control, and then regains control. The SMJ34020A regains bus mastership as soon as $\overline{GI}$ is driven active (low). $\overline{R0}$ and $\overline{R1}$ 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 SMJ34020A clock stretch not be used in multiprocessor systems.

cycle timing examples (continued)

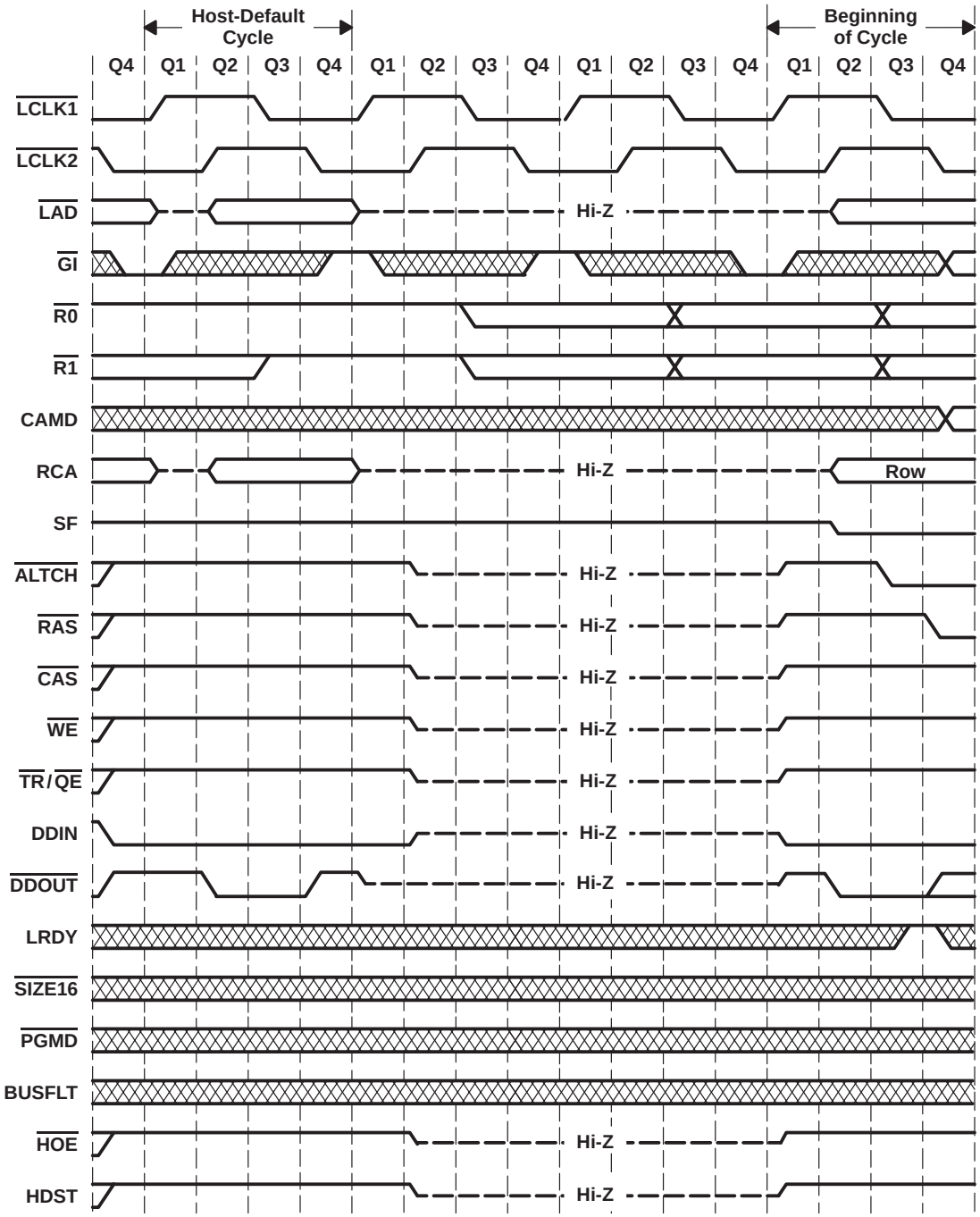


Figure 25. Multiprocessor-Interface-Cycle Timing (High-Impedance Signals)

cycle timing examples (continued)

Two SMJ34020As use the multiprocessor interface to pass control of local memory from one to the other (Figure 26). GSP1 completes a read cycle to the local memory and, although desiring another read, loses the bus to GSP2, which does a single write cycle (perhaps a host-write access). GSP1 then regains control and completes the read cycle (shown with a single wait state). Since no further memory-access requests are present, GSP1 maintains control of the bus and holds all of the local-memory control signals at their inactive levels. LRDY is a common input to both GSP1 and GSP2.

The host cycle timing diagrams shown in this data sheet are only a sample. For more information, see the *TMS34020 User's Guide*.

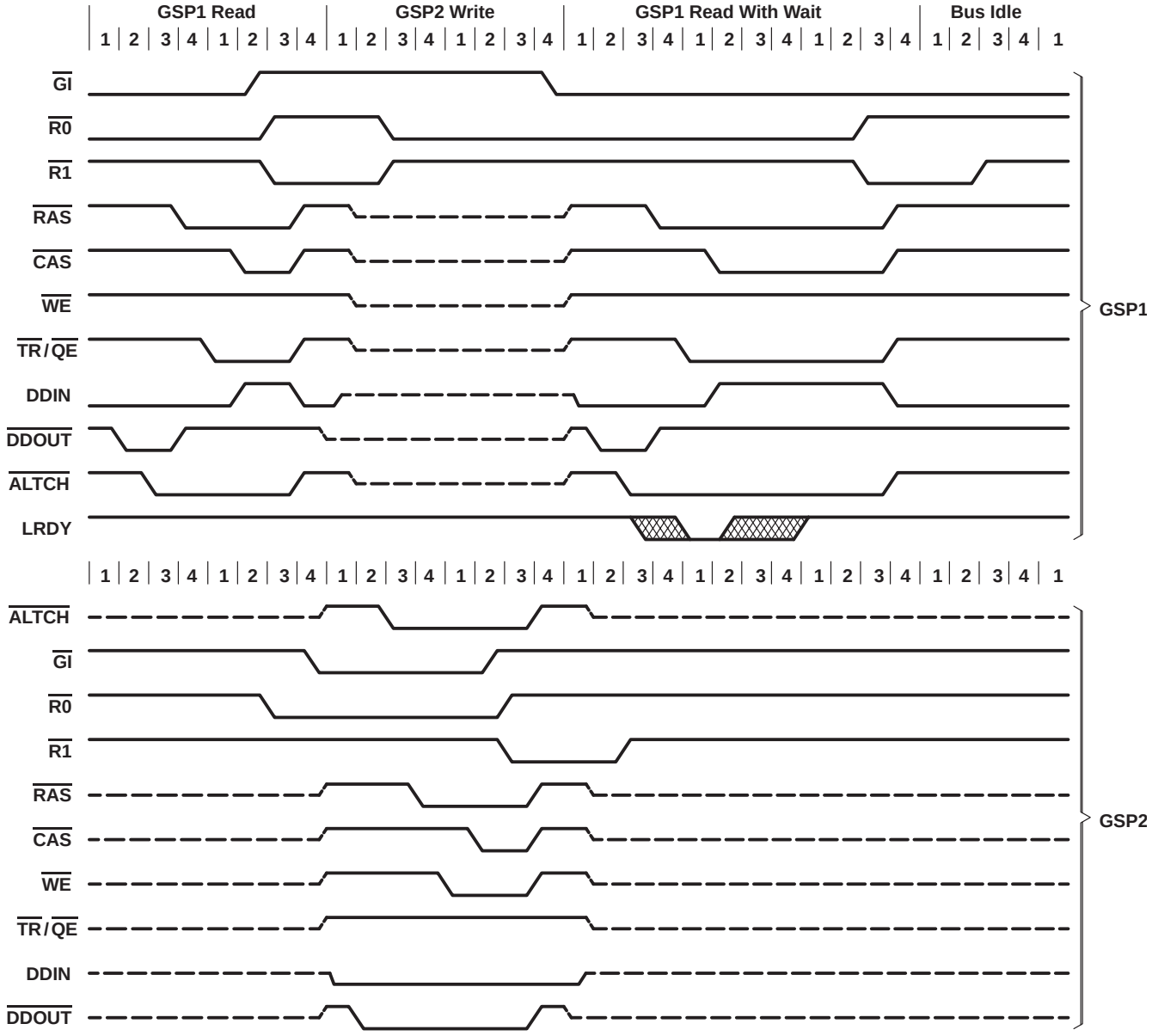
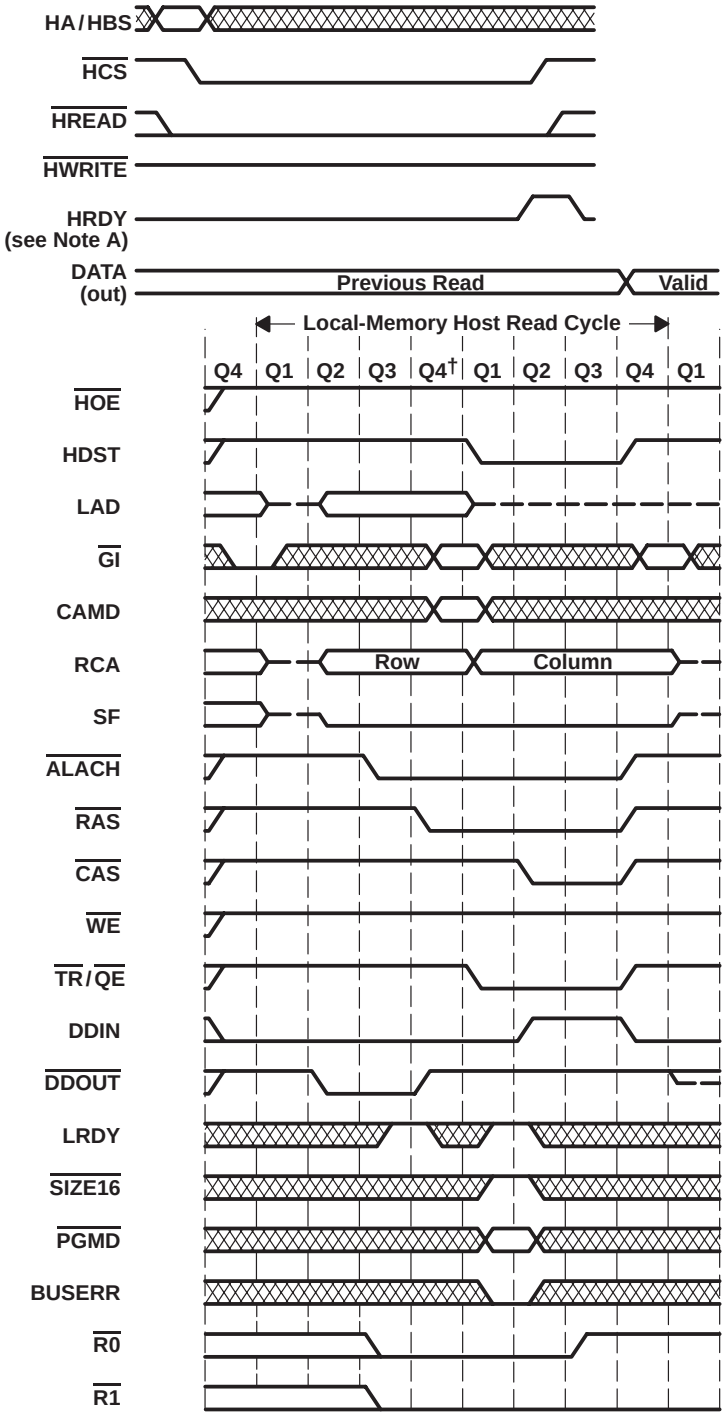


Figure 26. Multiprocessor-Interface Cycle Timing (Passing Control)

cycle timing examples (continued)

In Figure 27, the host-access request is synchronized to the SMJ34020A at the beginning of Q4 so that the local-memory cycle can begin in Q1. If the external host-access request occurs after the setup time requirement before Q4, the request is not considered until the next Q4 cycle. In order to provide back-to-back accesses as indicated in this example, the host must remove $\overline{\text{HCS}}$ on receipt of HRDY and reassert it before Q4 (it can also remove and reassert $\overline{\text{HREAD}}$ with $\overline{\text{HCS}}$).

cycle timing examples (continued)



† See clock stretch, page 21.

NOTE A: HRDY goes high at the start of Q2; however, data is not strobed into the external latches until the start of Q4 when HDST goes high.

Figure 27. Host-Read-Cycle Timing (Random/Same Accesses, not From SMJ34020A I/O Registers)

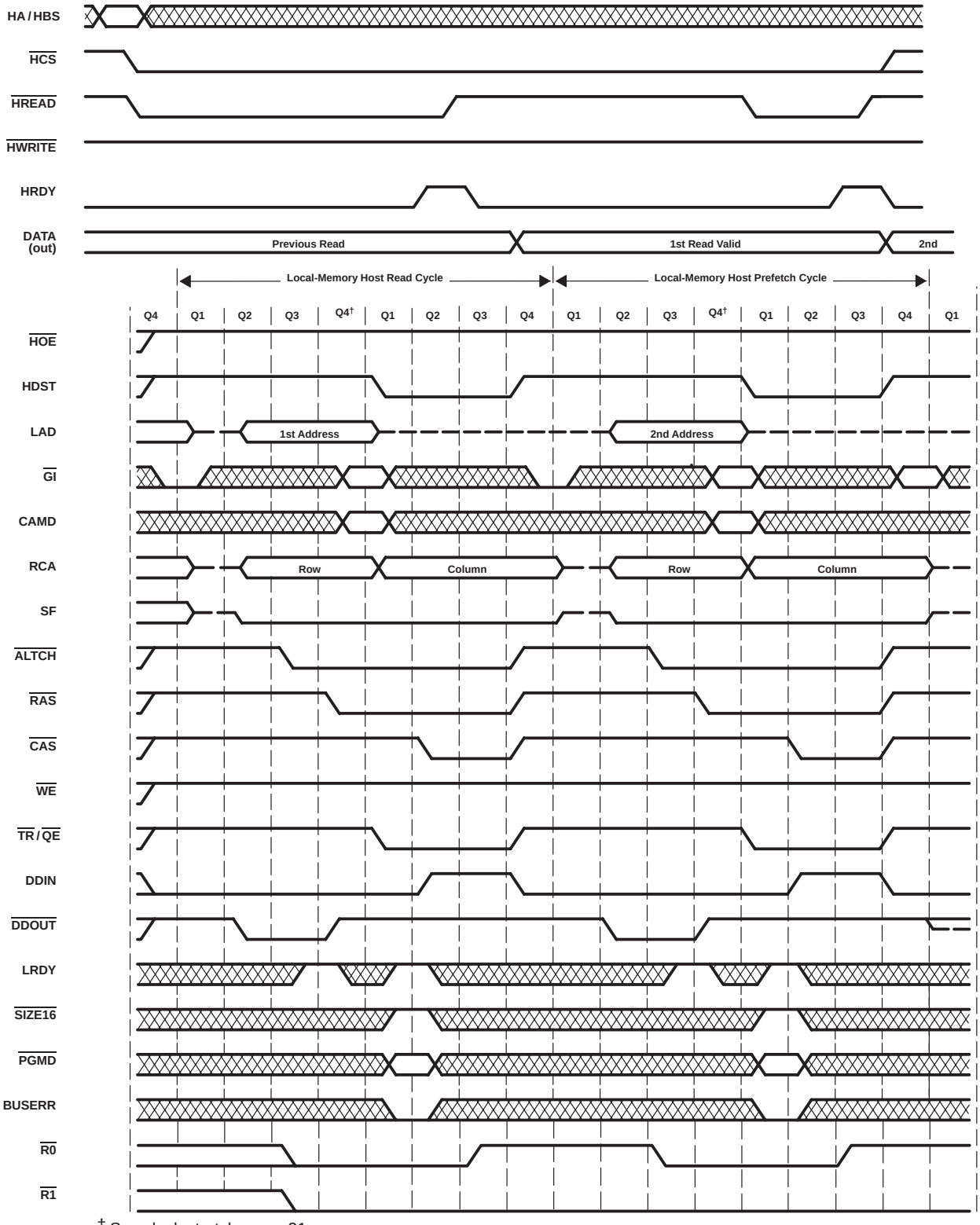
cycle timing examples (continued)

The host-access request is synchronized to the SMJ34020A at the beginning of Q4 so that the local memory cycle can begin in Q1.

In block mode (prefetch after read), the SMJ34020A automatically initiates sequential read accesses as soon as the host deasserts the current read request. In this example, the host reads a location and must wait for the first access to complete. When the host removes $\overline{\text{HREAD}}$ (Figure 28), indicating the end of the first read, the SMJ34020A starts to prefetch the next sequential location. When the host makes the next request, the SMJ34020A has prefetched the data so that the host reads with no delay. While in block mode, the SMJ34020A continues to prefetch data for the host read each time the host removes either $\overline{\text{HREAD}}$ or $\overline{\text{HCS}}$. If the address present and latched at the falling edge of $\overline{\text{HCS}}$ matches the previously prefetched address, HRDY is asserted high so that the host can read with no delay.

In read-modify-write mode (prefetch after write), the SMJ34020A initiates the read access as soon as the current write request is deasserted.

cycle timing examples (continued)



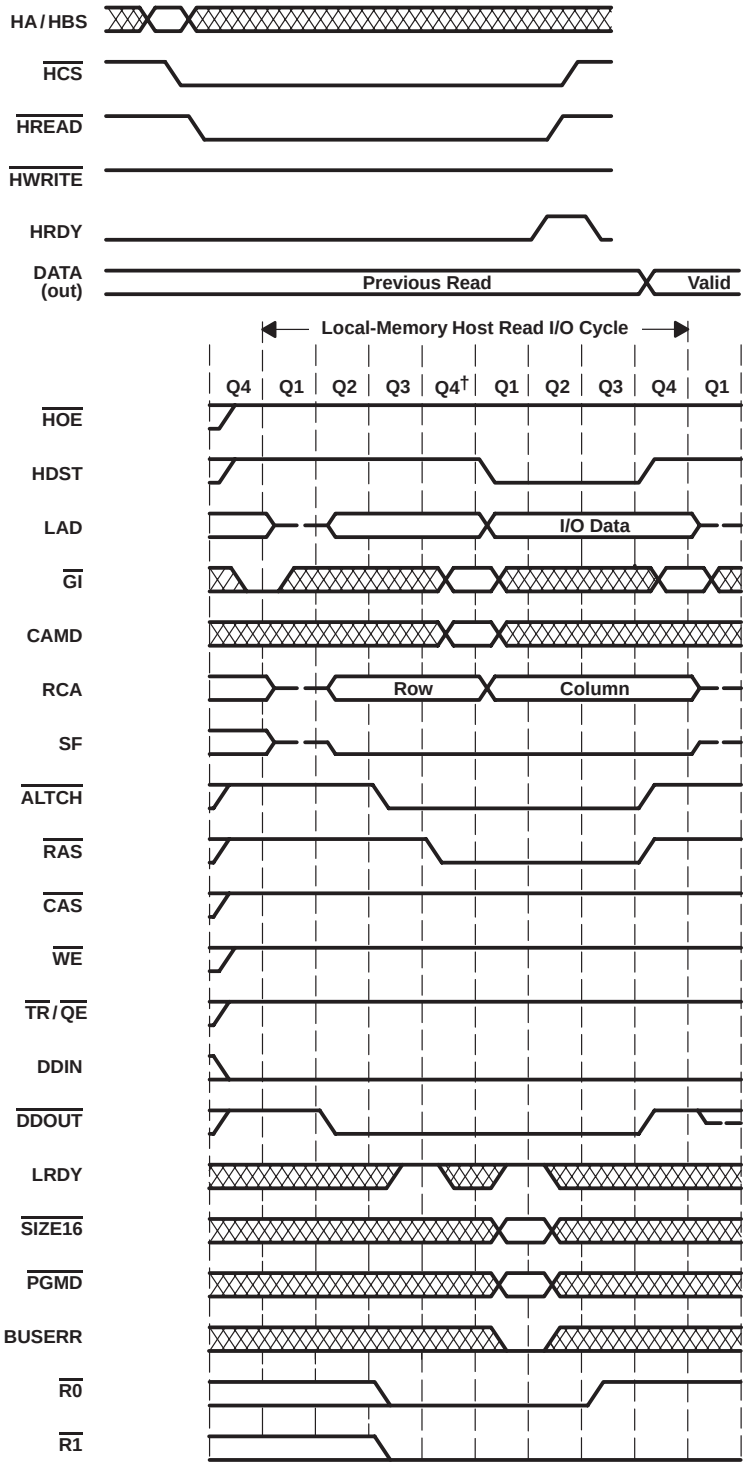
† See clock stretch, page 21.

Figure 28. Back-to-Back Host-Read Cycles With Implicit Addressing; $\overline{\text{HREAD}}$ as Strobe

cycle timing examples (continued)

The host read of the SMJ34020A I/O registers (Figure 29) suppresses the generation of $\overline{TR}/\overline{QE}$ and $\overline{CAS}$ so that data is read from the SMJ34020A rather than from memory. $\overline{DDOUT}$ is enabled so that data can flow through external buffers on LAD to the host data latches. The SMJ34020A I/O registers can be accessed in any of the host access modes (random/same, block, or read-modify-write).

cycle timing examples (continued)



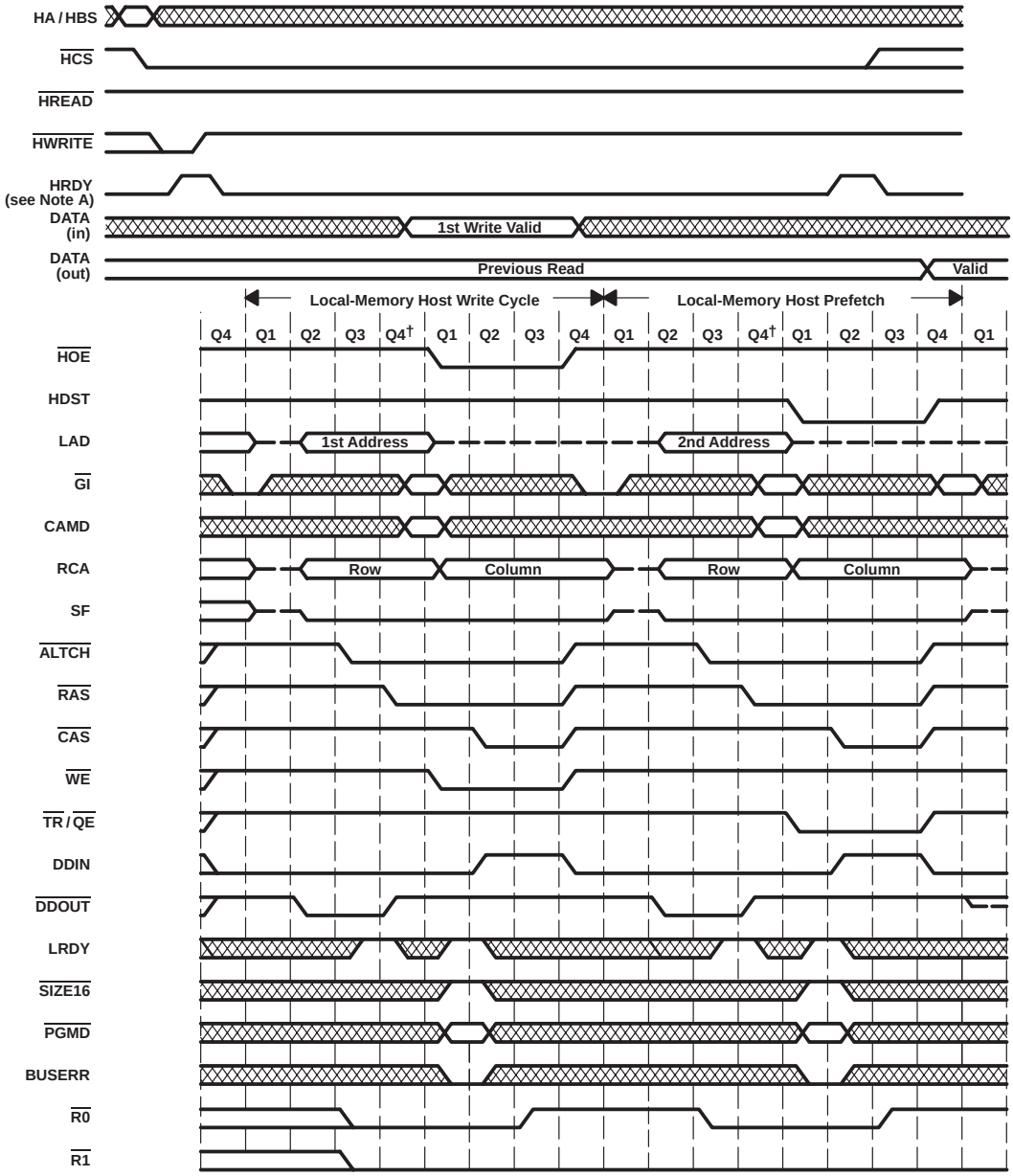
† See clock stretch, page 21.

Figure 29. Host-Read Cycle Timing From SMJ34020A I/O Registers

cycle timing examples (continued)

In Figure 30, SMJ34020A provides HRDY as soon as it recognizes the host write cycle (if no other host write cycle is in progress), allowing the host to latch the data in the external data latches. The host then attempts a second write but does not get an immediate HRDY because the SMJ34020A is still writing the first data to memory. As soon as the memory write completes, HRDY goes high so that the host can latch the new data. The SMJ34020A then writes the second data while the host continues other processing. The host access request is synchronized to the SMJ34020A at the beginning of Q4 so that the local memory cycle can begin in Q1. If the external host access request occurs after the setup time requirement before Q4, the request is not considered until the next Q4 cycle. During a host write cycle DDIN is active so that if the write is to the SMJ34020A I/O registers, the data can be required within the GSP.

cycle timing examples (continued)



† See clock stretch, page 21.

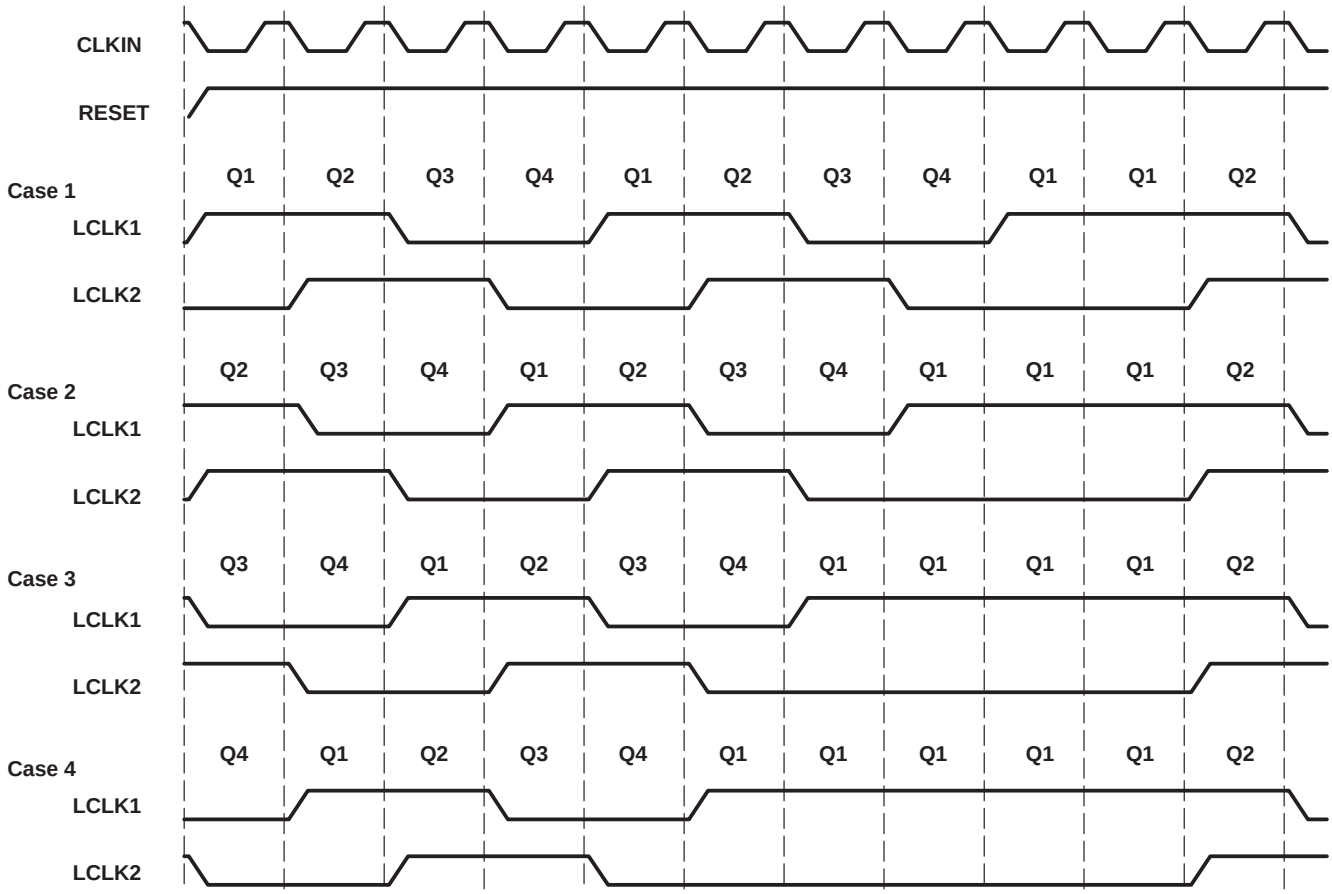
NOTE A: HRDY goes high at the start of Q2; however, the memory cycle writing data to memory is not completed until the start of Q4 when ALTCH, CAS, and HOE return high. The host must not strobe new data into the external latch until just after the start of Q4.

Figure 30. Host-Write Cycle Back-to-Back With Prefetch of Next Word and Implicit Addressing; HREAD and HWRITE Used as Strobes

cycle timing examples (continued)

Although $\overline{\text{RESET}}$ is not *normally* required to be synchronous to CLKIN, in order to facilitate synchronization of multiple SMJ34020As 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 (Figure 31). The four possible conditions for the state of the SMJ34020A at the time $\overline{\text{RESET}}$ goes high are shown below. Quarter cycle 1 is extended accordingly to provide synchronization of the GSPs. All SMJ34020As to be synchronized must share a common CLKIN and $\overline{\text{RESET}}$. Within 10 CLKIN cycles after $\overline{\text{RESET}}$ goes high, all GSPs are synchronized to the same quarter cycle through the extension of Q1 cycles.

It is recommended that SMJ34020A stretch mode not be used in multiprocessor systems.

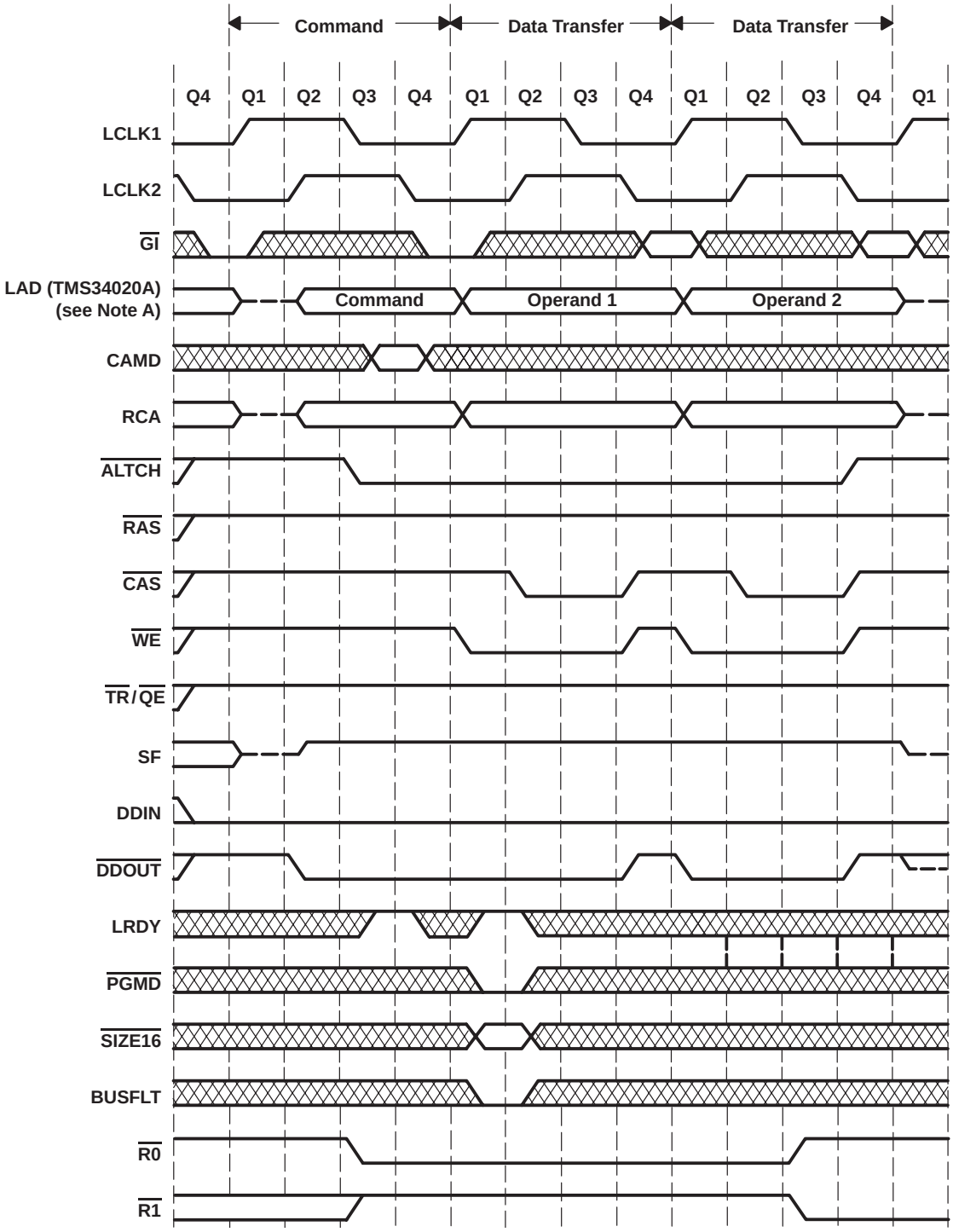


NOTE A: No timing dependencies of LCLK1 and LCLK2 relative to CLKIN or $\overline{\text{RESET}}$ are to be implied from this figure.

Figure 31. Synchronization of Multiple SMJ34020As

cycle timing examples (continued)

The timing example in Figure 32 is like a memory write cycle except that $\overline{RAS}$ and SF are high.

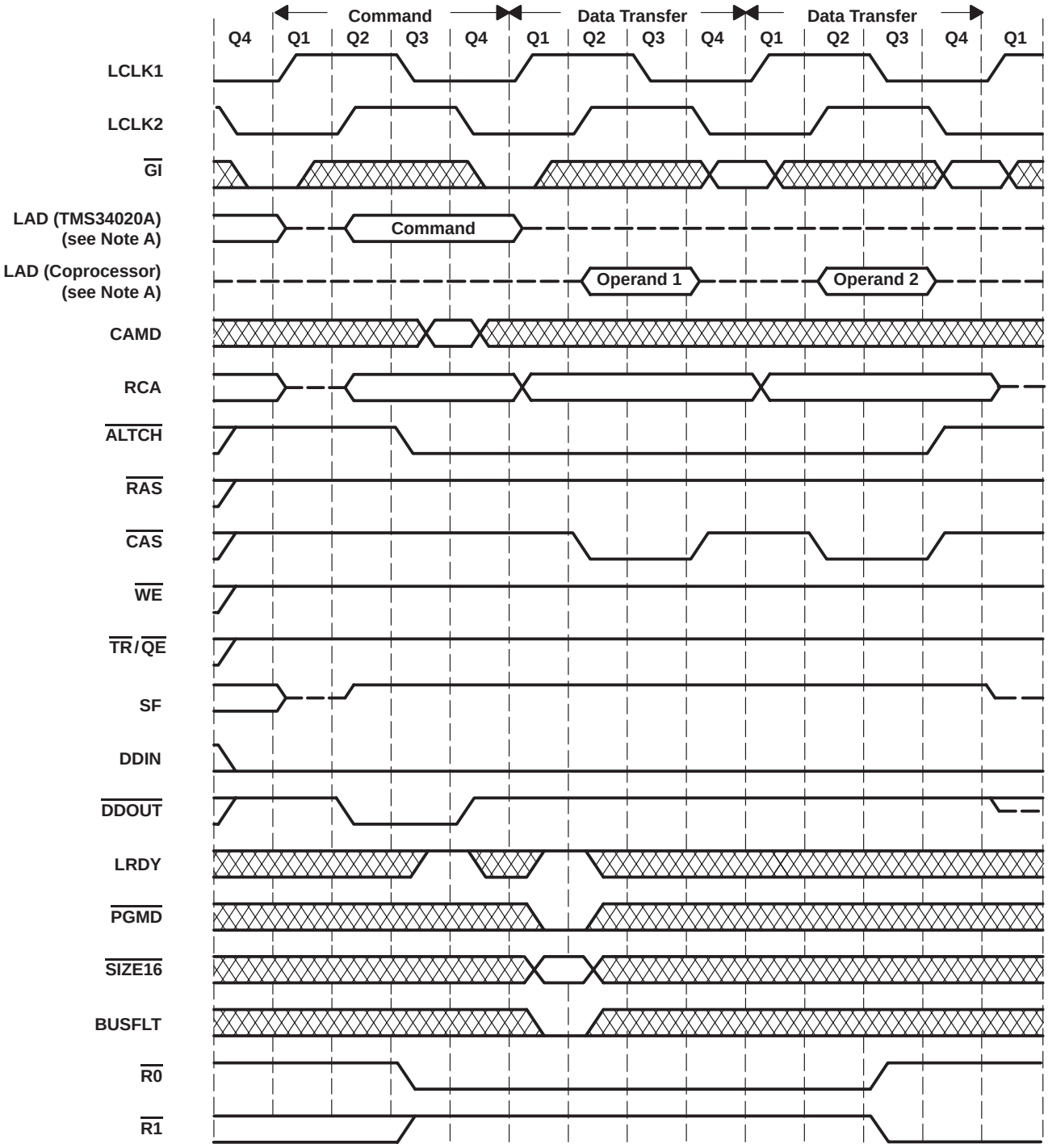


NOTE A: LAD (SMJ34020A): Output to LAD by the SMJ34020A
Command: Coprocessor ID, instruction and status code present on LAD
Operand n: Data to or from the coprocessor

Figure 32. Transfer SMJ34020A Register(s) to Coprocessor (One or Two 32-Bit Values)

cycle timing examples (continued)

The timing example in Figure 33 is like a memory write cycle except that $\overline{RAS}$ and SF are high.

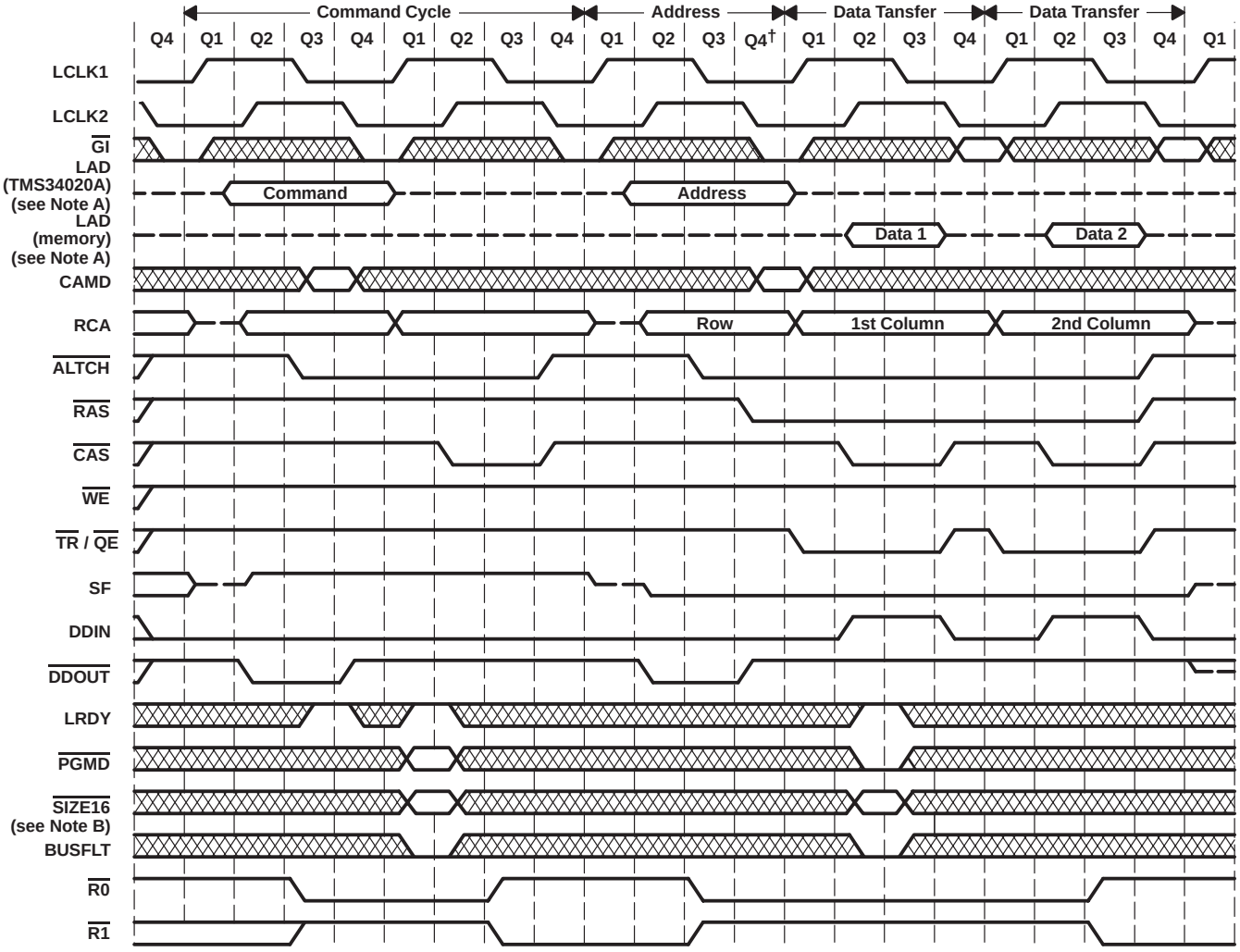


NOTE A: LAD (SMJ34020A): Output to LAD by the SMJ34020A
LAD (coprocessor): Output to LAD by the coprocessor
Command: Coprocessor ID, instruction and status code present on LAD
Operand n: Data to or from the coprocessor

Figure 33. Transfer-Coprocessor Register to SMJ34020A (One or Two 32-Bit Values)

cycle timing examples (continued)

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 (Figure 34). The coprocessor can place status information on LAD during the initialization cycle for the SMJ34020A. 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 SMJ34020A to be used for the count. Both instructions specify a register to be used as an index into memory. The index can be postincremented or predecremented on each transfer cycle.



† See clock stretch, page 21.

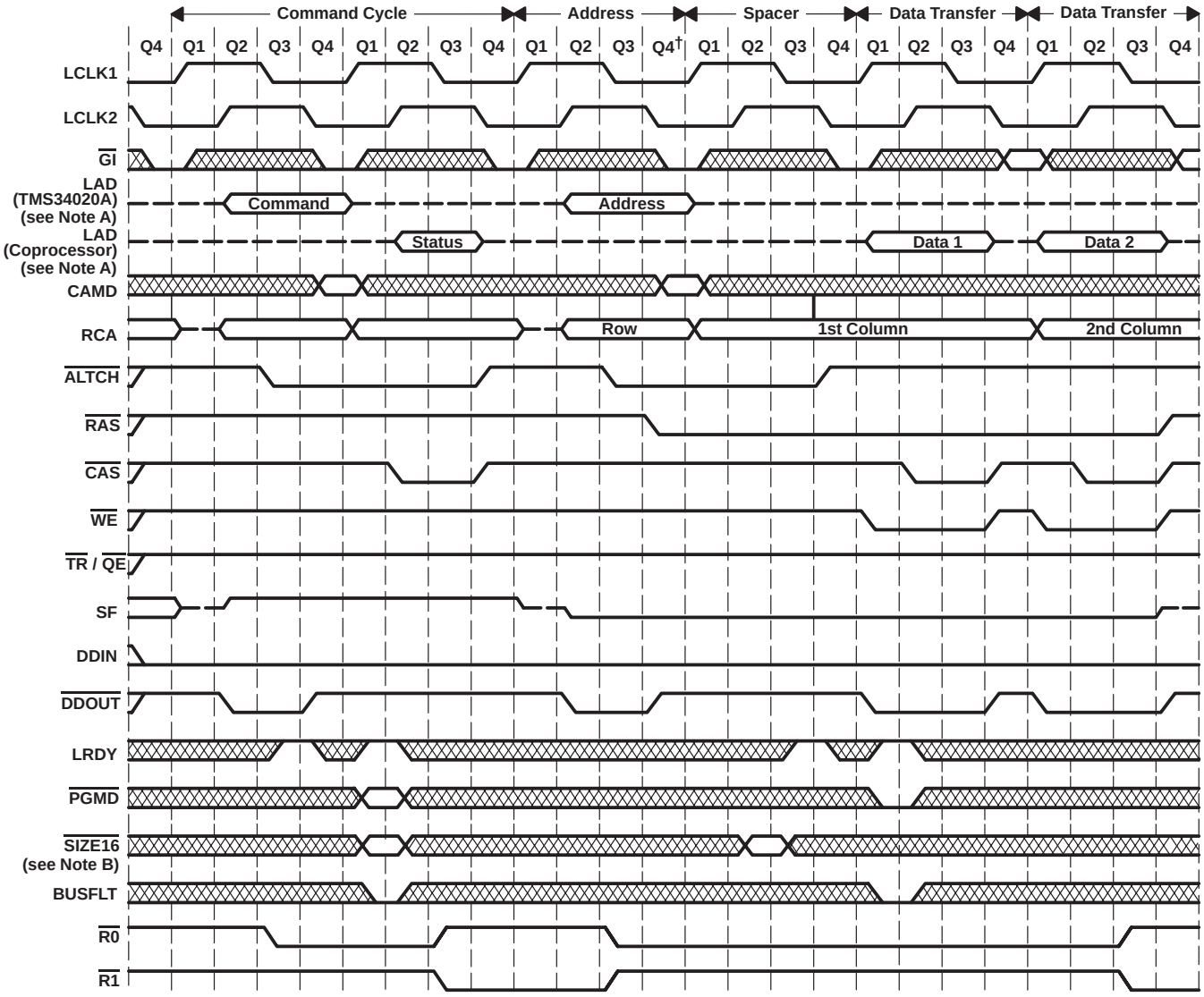
NOTES: A. LAD (SMJ34020A): Output to LAD by the SMJ34020A
LAD (memory): Output to LAD by the memory
Command: Coprocessor ID, instruction and status code present on LAD
Address: Memory address for the data transfer with coprocessor status code
Data n: Data to or from the coprocessor (number of values transferred depends on a value in a register or count in the instruction)

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

Figure 34. Transfer Memory to Coprocessor Register(s)

cycle timing examples (continued)

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 (Figure 35). The coprocessor can place status information on LAD during the initialization cycle for the SMJ34020A. The memory cycle includes a dead cycle to enable the SMJ34020A to take LAD drivers to the 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 postincrement and the other uses it with a predecrement after each transfer cycle.



† See clock stretch, page 21.

NOTES: A. LAD (SMJ34020A): Output to LAD by the SMJ34020A
LAD (coprocessor): Output to LAD 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

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

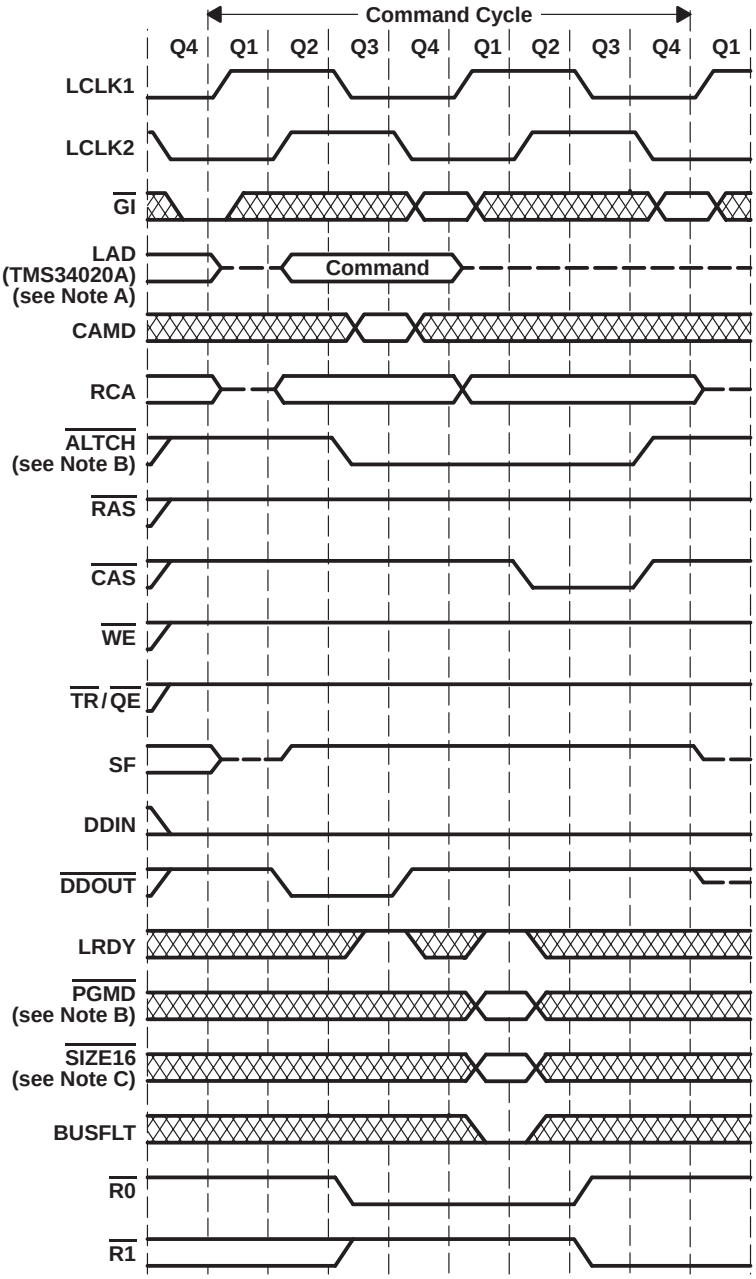
Figure 35. Transfer-Coprocessor Register(s) to Memory ($\overline{\text{ALTCH}}$ High During Data Transfer)

cycle timing examples (continued)

The timing example in Figure 36 is like a memory write cycle except that $\overline{\text{RAS}}$ 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 can place status information on LAD during the cycle for the SMJ34020A.

cycle timing examples (continued)



NOTES: A. LAD (SMJ34020A): Output to LAD by the SMJ34020A
LAD command: Coprocessor ID, instruction and status code present on LAD

B. Although the coprocessor internal command never requires the use of page mode cycles, $\overline{\text{PGMD}}$ should be held at a valid level during the start of Q2 after $\overline{\text{ALTCH}}$ has gone low.

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

Figure 36. Coprocessor-Internal Operation Command Cycle

absolute maximum ratings over operating case temperature range†

Maximum supply voltage, V_{CC} (see Note 1) . . . . . 7 V
Input voltage range . . . . . − 0.3 V to 7 V
Off-state output voltage range . . . . . − 2 V to 7 V
Operating case termperature range, T_C . . . . . − 55°C to 125°C
Storage temperature range, T_{stg} . . . . . − 65°C to 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

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

recommended operating conditions

			MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	34020A-32	4.5	5	5.5	V
		34020A-40	4.75	5	5.25	
V_{SS}	Supply voltage (see Note 2)			0		V
I_{OH}	High-level output current				400	µA
I_{OL}	Low-level output current				2	mA
T_C	Operating case temperature (see Note 3)		− 55		125	°C

NOTE 2: A minimum inductance path between the V_{SS} pins and system ground must be provided to minimize noise on V_{SS}.

NOTE 3: T_C MAX at maximum rated operating conditions at any point on case. T_C MIN at initial (time zero) power up.

dc electrical characteristics over recommended range of supply voltage (see Note 4)

PARAMETER				TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{IH}	High-level input voltage	BUSFLT, LRDY, VCLK, PGMD, SIZE16, CSYNC, VSYNC, HSYNC	GB PKG		2.2		V_{CC}+0.3	V
			HT PKG		2.3		V_{CC}+0.3	
		HWRITE, HREAD	GB PKG		2		V_{CC}+0.3	
			HT PKG		2.3		V_{CC}+0.3	
		HA5−HA31, HCS, HBS0−HBS3	GB PKG		2		V_{CC}+0.3	
			HT PKG		2.3		V_{CC}+0.3	
		CLKIN only			3		V_{CC}+0.3	
		All other inputs			2		V_{CC}+0.3	
V_{IL}	Low-level input voltage, HT only: HCS V_{IL} = − 0.3 min, 0.7 V max				−0.3		0.8	V
V_{OH}	High-level output voltage			V_{CC} = MIN, I_{OH} = MAX	2.6			V
V_{OL}	Low-level output voltage		GB PKG	V_{CC} = MAX, I_{OL} = MIN			0.60	V
		DDIN, HINT, HRDY, R0, R1, EMU3	HT PKG				0.8	
		HYSNC, VSYNC					0.8	
		All other outputs					0.6	
I_O	Output current, leakage (high impedance)		GB PKG	V_{CC} = MAX, V_O = 2.8 V			20	μA
			HT PKG				20	
			GB PKG	V_{CC} = MAX, V_O = 0.6 V			− 20	
			HT PKG				−20	
I_I	Input current (All inputs except EMU0−EMU2, HREAD, HWRITE‡)			V_I = V_{SS} to V_{CC}			±20	μA
I_{CC}	Supply current		34020A-32	V_{CC} = MAX, Freq = MAX			265	mA
			34020A-40				280	
C_i	Input capacitance					10	18	pF
C_o	Output capacitance					18	25	pF

† All typical values are at V_{CC} = 5 V, T_A (ambient-air temperature)= 25° C.
‡ EMU0−EMU2 are not connected in a typical configuration. Nominal pullup current for EMU0−EMU2 and HREAD, HWRITE is 600 μA.
NOTE 4: HDST and HOE (output terminals) have internal pullup resistors that allow high logic levels to be maintained when the SMJ34020A is not actually driving these pins.

signal transition levels

NOTE A: 2.2 V for BUSFLT, VCLK, LRDY, PGMD, SIZE16. 3V for CLKIN.

Figure 37. TTL-Level Inputs

For high-to-low and low-to-high transitions, the level at which the input timing is measured is 1.5 V.

signal transition levels (continued)

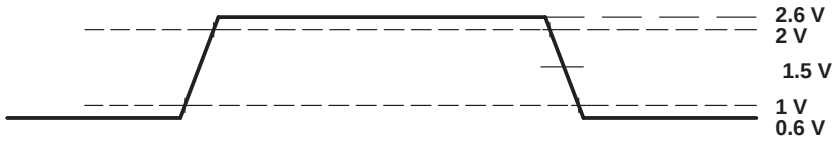
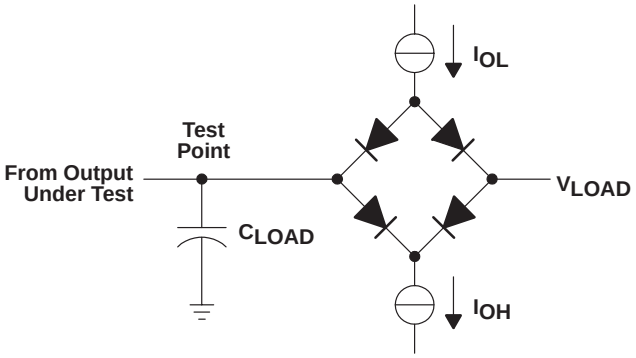


Figure 38. TTL-Level Outputs

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 1 V. For a low-to-high transition, the level at which the output is said to be no longer low is 1 V, and the level at which the output is said to be high is 2 V. A V_{OL} trip level of 1.5 V is used for timing requirements for testing at − 55°C.

test measurement

The test load circuit shown in Figure 39 represents the programmable load of the tester pin electronics that is used to verify timing parameters of SMJ34020A output signals.



Where: I_{OL} = 2 mA (all outputs)
I_{OH} = 400 µA (all outputs)
V_{LOAD} = 1.5 V
C_{LOAD} = 80 pF typical load circuit capacitance

NOTE: The load applied may be set higher than the values indicated for I_{OL} and I_{OH} during timing tests in order to reduce signal bounce induced by the tester hardware. However the timing performance is assured at the stated load values.

Figure 39. Test Load Circuit

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:

A	HA5−HA31 and $\overline{\text{HBS0}}$−$\overline{\text{HBS3}}$	LINT	$\overline{\text{LINT1}}$, $\overline{\text{LINT2}}$
AD	LAD0−LAD31 and RCA0−RCA12	OE	$\overline{\text{HOE}}$
AL	$\overline{\text{ALTCH}}$	RC	RCA0−RCA12
BC	Any of the bus control input signals (LRDY, $\overline{\text{PGMD}}$, $\overline{\text{SIZE16}}$, or BUSFLT)	RD	$\overline{\text{HREAD}}$
CE	$\overline{\text{CAS0}}$−$\overline{\text{CAS3}}$	RE	$\overline{\text{RAS}}$
CK	LCLK1 and LCLK2	RQ	$\overline{\text{R0}}$ or $\overline{\text{R1}}$
CK1	LCLK1	RS	$\overline{\text{RESET}}$
CK2	LCLK2	RY	HRDY
CKI	CLKIN	S	$\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, or $\overline{\text{CSYNC}}$
CM	$\overline{\text{CAMD}}$	SC	EMU3
CS	$\overline{\text{HCS}}$	SCK	SCLK
CT	Any of the bus control output signals ($\overline{\text{ALTCH}}$, $\overline{\text{CAS0}}$−$\overline{\text{CAS3}}$, $\overline{\text{RAS}}$, $\overline{\text{WE}}$, $\overline{\text{TR}}$/$\overline{\text{QE}}$, $\overline{\text{HOE}}$, or HDST)	SF	SF
DI	DDIN	SG	Any output signal
DO	$\overline{\text{DDOUT}}$	SGV	Signal valid
EM	EMU0, EMU1, EMU2	ST	HDST
HI	$\overline{\text{HINT}}$	TR	$\overline{\text{TR}}$/$\overline{\text{QE}}$
HS	$\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$, or $\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$	VCK	VCLK
GI	$\overline{\text{GI}}$	WR	$\overline{\text{HWRITE}}$
LA	LAD0−LAD31		

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 level
L	Low level
V	Valid level
X	Unknown, changing or don’t care level
Z	High-impedance state of 3-state output

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 equal to the input clock period, $t_{c(CKI)}$.

All output signals from the SMJ34020A 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.

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

CLKIN and $\overline{\text{RESET}}$ timing requirements (see Figure 40)

NO.				SMJ34020A-32		34020A-40		UNIT
				MIN	MAX	MIN	MAX	
1	$t_{c(CKI)}$	Cycle time, period of CLKIN ($4t_Q$)		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		15†		12‡		ns
6	$t_{su(RSH-CKI)}$	Setup time, $\overline{\text{RESET}}$ high to CLKIN no longer low		10†		6‡		ns
7	$t_{w(RSL)}$	Pulse duration, $\overline{\text{RESET}}$ low	Initial reset during powerup	$160t_Q - 40$‡		$160t_Q - 40$‡		ns
			Reset during active operation	$16t_Q - 40$‡		$16t_Q - 40$‡		
8	$t_{su(CSL-RSH)}$	Setup time, $\overline{\text{HCS}}$ low to $\overline{\text{RESET}}$ high to configure self-bootstrap mode		$8t_Q + 55$		$8t_Q + 55$		ns
9	$t_{d(CSH-RSH)}$	Delay time, $\overline{\text{HCS}}$ no longer low 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 timings are required only to synchronize the SMJ34020A 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 SMJ34020A has been initialized need to be present only long enough to be recognized by the internal logic; the internal logic maintains 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 assure that the SMJ34020A is configured to run in the self-bootstrap mode (HLT bit = 0) following the end of reset.

* The parameter is not production tested.

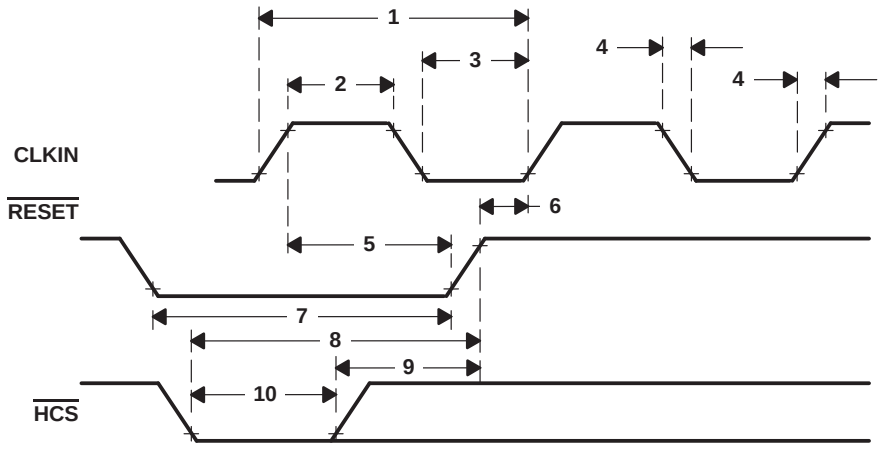


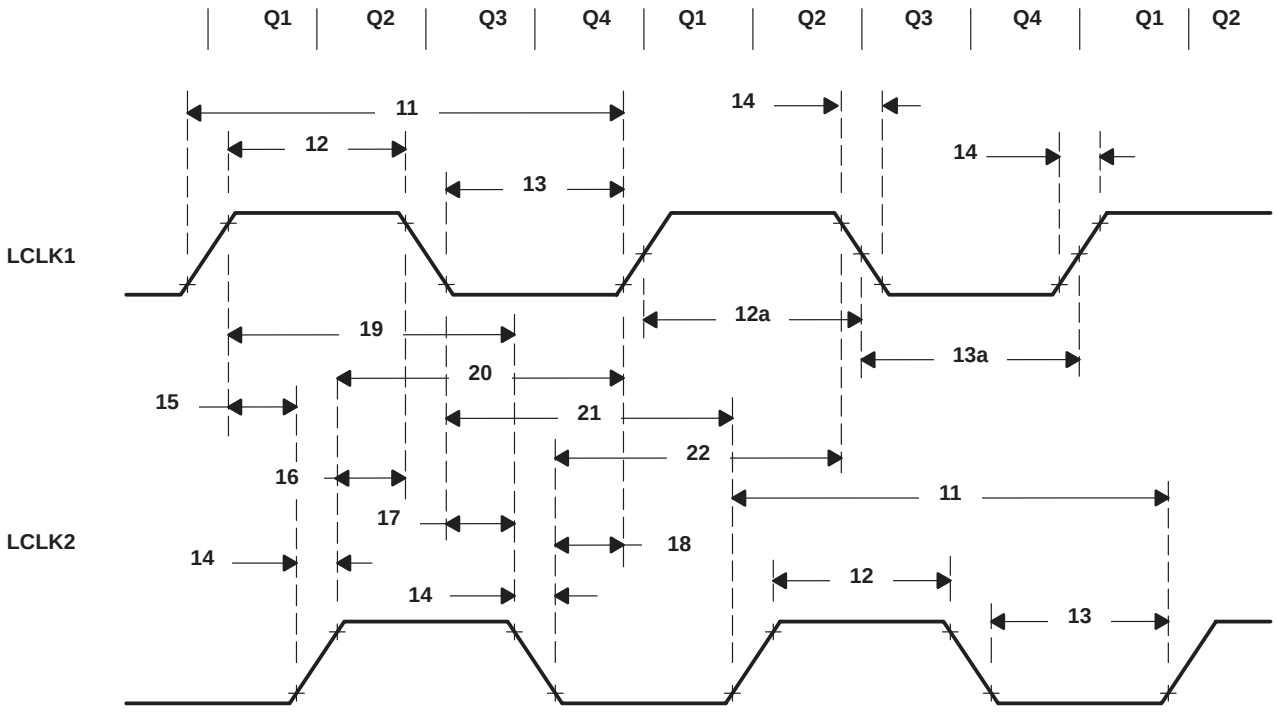
Figure 40. CLKIN and $\overline{\text{RESET}}$ Timing Requirements

local-bus timing: output clocks (see Note 5 and Figure 41)

NO.			SMJ34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
11	$t_{c(LCK)}$	Cycle time, 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	$t_{w(LCKH)}$	Pulse duration, LCLK1 high (see Note 6)	$2t_Q$ −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	$t_{w(LCKL)}$	Pulse duration, LCLK1 low (see Note 6)	$2t_Q$−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 parameter can also be specified as $4t_Q$.
* The parameter is not production tested.

NOTES: 5. s = t_Q if using the clock stretch;
s = 0 otherwise
6. Parameters 12a and 13a are specified with 1.5 V timing levels (parameters 12 and 13 are specified with standard timing voltage levels).



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 41. Local-Bus Timing: Output Clocks

output signal characteristics (see Notes 7 and 8)

The following general parameters are common to all output signals from the SMJ34020A unless otherwise stated. They are intended as an aid in estimating the timing requirements. See the specific numbered parameters for actual times. In the minimum and maximum values shown, “n” is an integral number of quarter cycles.

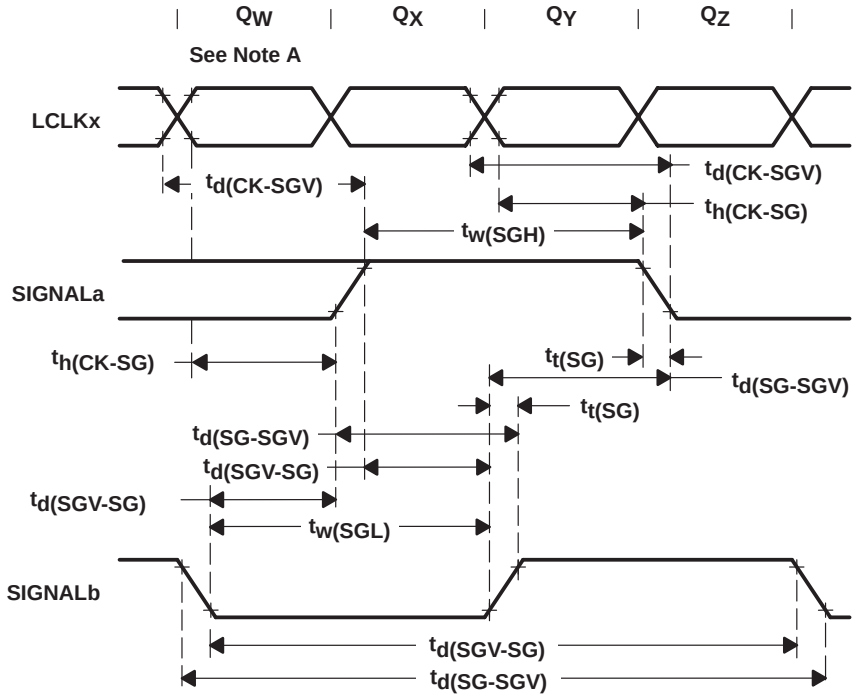
PARAMETER			SMJ34020A-32 MIN	SMJ34020A-32 MAX	UNIT
$t_{h(CK-SGNV)}$	Hold time, LCLKx to output signal not valid		$t_Q - 15$		ns
$t_{d(CK-SGV)}$	Delay time, LCLKx start of transition to output signal valid	Fast: $\overline{RAS}$, $\overline{CAS}$, $\overline{ALTCH}$, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$		$t_Q + 15$	ns
		Slow: LAD, RCA, SF		$t_Q + 22$	ns
$t_{d(SGNV-SGV)}$	Delay time, output signal started transition to output signal valid	Fast: $\overline{RAS}$, $\overline{CAS}$, $\overline{ALTCH}$, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$		$nt_Q + 15$	ns
		Slow: LAD, RCA, SF		$nt_Q + 22$	ns
$t_{d(SGV-SG)}$	Delay time, output signal valid to output signal not valid	Fast: $\overline{RAS}$, $\overline{CAS}$, $\overline{ALTCH}$, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$	$nt_Q - 15$ $nt_Q - 16$†		ns
		Slow: LAD, RCA, SF	$nt_Q - 22$		ns
$t_{t(SG)}$	Output signal transition time	Fast: $\overline{RAS}$, $\overline{CAS}$, $\overline{ALTCH}$, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$		15	ns
		Slow: LAD, RCA, SF		22	ns
$t_{w(SGH)}$	Pulse duration, output signal high	Fast: $\overline{RAS}$, $\overline{CAS}$, $\overline{ALTCH}$, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$	$nt_Q - 15$		ns
		Slow: LAD, RCA, SF	$nt_Q - 22$		ns
$t_{w(SGL)}$	Pulse duration, output signal low	Fast: $\overline{RAS}$, $\overline{CAS}$, $\overline{ALTCH}$, $\overline{TR/QE}$, $\overline{DDOUT}$, DDIN, EMU3, $\overline{HOE}$, $\overline{R0}$, $\overline{R1}$, HDST, $\overline{WE}$	$nt_Q - 15$		ns
		Slow: LAD, RCA, SF	$nt_Q - 22$		ns

† See parameter 73 in “local-bus timing: bus control inputs” table.

NOTES: 7. Also see Figure 34 on following page.

8. For parameters on this page specifying minimum or maximum times between two output signals, the word fast or slow in column 2 refers to the signal with a subscript of 1, regardless of the other signal. For example, if you are using the spec $t_{h(SG2NV-SG1V)}$, use the slow value if the signal becoming valid (SG_1) is RCA, LAD, or SF; use the fast value otherwise. The pin referred to as SG_2 does not determine fast or slow signal time.

output signal characteristics (continued)



NOTE A: Any of these quarter phases could be $2t_Q$ if they are stretched (see clock stretch, page 21).

Figure 42. Output Signal Characteristics

example of how to use the general output signal characteristics

Assume a system is using a SMJ34020A-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 LAD for a local-memory write cycle.

From the local-memory read-modify-write-cycle timing diagram (Figure 12), the time from the falling edge of $\overline{\text{ALTCH}}$ to valid data on LAD 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 $t_{d(SG\text{-}SGV)}$. In Figure 42, there are two representations of $t_{d(SG\text{-}SGV)}$ 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 LAD. By definition, the signal becoming valid (SGV) determines whether the fast value or the slow value from the table is used.

In this case, for parameter $t_{d(SG\text{-}SGV)}$, SGV is LAD. LAD is in the slow group, so the maximum value for $t_{d(SG\text{-}SGV)}$ is nt_Q + 22. The value for n is 2 from the analysis of the diagram on page 28. Thus, the maximum time from the start of the falling edge of $\overline{\text{ALTCH}}$ to the time when data must be valid on LAD for a local-memory write cycle is $2t_Q$ + 22 ns.

host-interface-cycle timing requirements (see Note 9 and Figure 43)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
23	$t_{su(AV-CSL)}$	Setup time, address prior to $\overline{HCS}$ no longer high	12		10		ns
24	$t_{h(CSL-AV)}$	Hold time, address after $\overline{HCS}$ low	12		10		ns
25	$t_{w(CSH)}$	Pulse duration, $\overline{HCS}$ high	28		25		ns
26	$t_{w(RDH)}$	Pulse duration, $\overline{HREAD}$ high	28		25		ns
27	$t_{w(WRH)}$	Pulse duration, $\overline{HWRITE}$ high	28		25		ns
28	$t_{su(RDH-WRL)}$	Setup time, $\overline{HREAD}$ high to $\overline{HWRITE}$ no longer high	28		25		ns
29	$t_{su(WRH-RDL)}$	Setup time, $\overline{HWRITE}$ high to $\overline{HREAD}$ no longer high	28		25		ns
30	$t_{w(RDL)}$	Pulse duration, $\overline{HREAD}$ low	18		15		ns
31	$t_{w(WRL)}$	Pulse duration, $\overline{HWRITE}$ low	18		15		ns
32	$t_{su(CSL-WRH)}$	Setup time, $\overline{HCS}$ low to $\overline{HWRITE}$ no longer low	18		15		ns
33	$t_{su(RDL-CK2L)}$	Setup time, $\overline{HCS}$ low or $\overline{HREAD}$ low to LCLK2 no longer high	30†		25†		ns
34	$t_{su(WRH-CK2L)}$	Setup time, $\overline{HWRITE}$ high or $\overline{HCS}$ high to LCLK2 no longer high	30†		25†		ns
35	$t_{h(CK2L-RDH)}$	Hold time, $\overline{HREAD}$ high after LCLK2 no longer high	0‡		0‡		ns
36	$t_{h(CK2L-WRL)}$	Hold time, $\overline{HWRITE}$ low after LCLK2 no longer high	0‡		0‡		ns
37	$t_{su(RDH-CK2L)}$	Setup time, $\overline{HREAD}$ high to LCLK2 no longer high, prefetch read mode	30†§		25†§		ns
38	$t_{su(CSL-RDH)}$	Setup time, $\overline{HCS}$ low to $\overline{HREAD}$ no longer low	18		15		ns

† Setup time to ensure recognition of input on this clock edge.
‡ Hold time required to assure response on next clock edge. These values are based on computer simulation and are not tested.
§ When the SMJ34020A is set for block reads, use the deassertion of $\overline{HREAD}$ to request a local memory cycle at the next sequential address location.

NOTE 9: Although $\overline{HCS}$, $\overline{HREAD}$, and $\overline{HWRITE}$ can be totally asynchronous to the SMJ34020A, cycle responses to the signals are determined by local memory cycles.

host-interface-cycle timing requirements (continued)

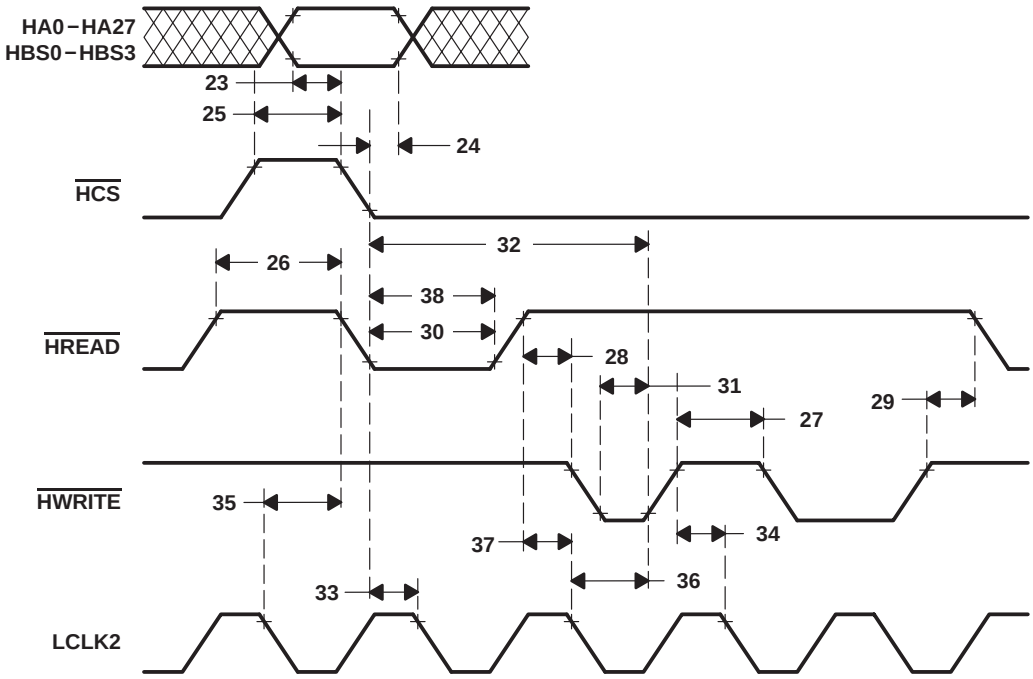


Figure 43. Host-Interface-Cycle Timing Requirements

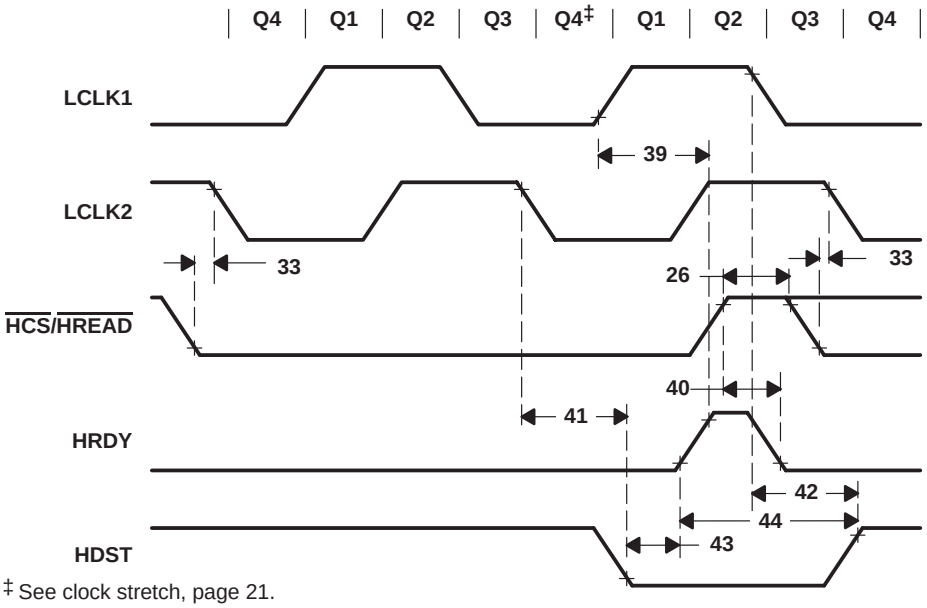
host-interface-cycle timing responses (random read cycle) (see Note 5 and Figure 44)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
26	$t_{w(RDH)}$	Pulse duration, $\overline{\text{HREAD}}$ high	28		25		ns
33	$t_{su(RDL-CK2L)}$	Setup time, $\overline{\text{HCS}}$ low or $\overline{\text{HREAD}}$ low to LCLK2 no longer high	30†		25†		ns
39	$t_{d(CK1H-RYH)}$	Delay time, LCLK1 going high to HRDY high (end of read cycle)		t_Q+20		t_Q+18	ns
40	$t_{d(RDH-RYL)}$	Delay time, $\overline{\text{HREAD}}$ or $\overline{\text{HCS}}$ high to HRDY low		20		18	ns
41	$t_{d(CK2L-STL)}$	Delay time, LCLK2 no longer high to HDST low	*s*+	t_Q+15		$t_Q+13.5+s$	ns
42	$t_{d(CK1L-STH)}$	Delay time, LCLK1 no longer high to HDST high		t_Q+15		$t_Q+13.5$	ns
43	$t_{su(STL-RYH)}$	Setup time, HDST low to HRDY no longer low	t_Q-15		$t_Q-13.5$		ns
44	$t_{d(RYH-STH)}$	Delay time, HRDY no longer low to HDST high		$2t_Q+15$		$2t_Q+13.5$	ns

† Setup time to ensure recognition of input on this clock edge

NOTE 5: *s* = t_Q if using the clock stretch;
s = 0 otherwise

.



‡ See clock stretch, page 21.

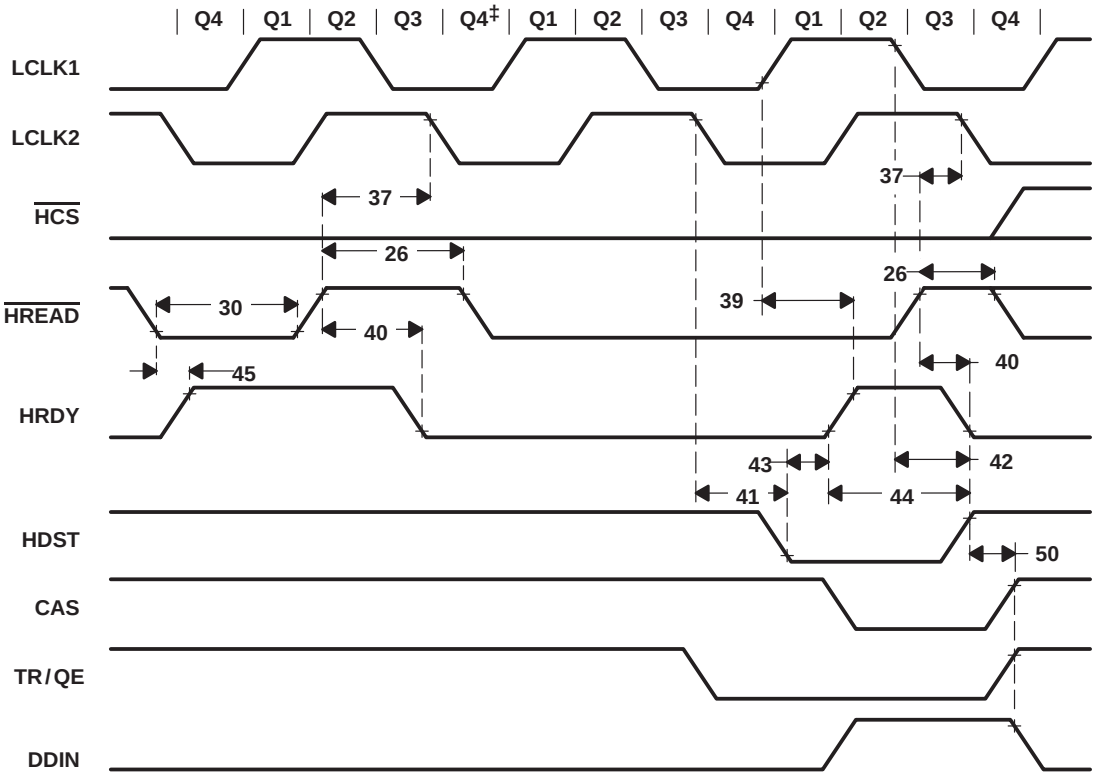
Figure 44. Host-Interface-Cycle Timing Responses (Random Read Cycle)

host-interface-cycle timing (block-read cycle) (see Notes 5 and 9 and Figure 45)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
26	$t_{w(RDH)}$	Pulse duration, $\overline{\text{HREAD}}$ high	28		25		ns
30	$t_{w(RDL)}$	Pulse duration, $\overline{\text{HREAD}}$ low	18		15		ns
37	$t_{su(RDH-CK2L)}$	Setup time, $\overline{\text{HREAD}}$ high to LCLK2 no longer high, prefetch read mode	30†		25†		ns
39	$t_{d(CK1H-RYH)}$	Delay time, LCLK1 no longer low to HRDY high		t_Q+20		t_Q+18	ns
40	$t_{d(RDH-RYL)}$	Delay time, $\overline{\text{HREAD}}$ or $\overline{\text{HCS}}$ high to HRDY low		20		18	ns
41	$t_{d(CK2L-STL)}$	Delay time, LCLK2 no longer high to HDST low		t_Q+15+s		$t_Q+13.5+s$	ns
42	$t_{d(CK1L-STH)}$	Delay time, LCLK1 no longer high to HDST high		t_Q+15		$t_Q+13.5$	ns
43	$t_{su(STL-RYH)}$	Setup time, HDST low to HRDY no longer low	t_Q-15		$t_Q-13.5$		ns
44	$t_{d(RYH-STH)}$	Delay time, HRDY no longer low to HDST high		$2t_Q+15$		$2t_Q+13.5$	ns
45	$t_{d(RDL-RYH)}$	Delay time, $\overline{\text{HREAD}}$ or $\overline{\text{HCS}}$ low to HRDY high after prefetch		25		20	ns
50	$t_{h(STH-CTV)}$	Hold time, $\overline{\text{CAS}}$, $\overline{\text{TR}}/\overline{\text{QE}}$, DDIN valid after HDST high	− 2		− 2		ns

† Setup time to ensure recognition of input on this clock edge. When the SMJ34020A is set for block reads, the deassertion of $\overline{\text{HREAD}}$ is used to request a local memory cycle at the next sequential address location.

NOTES: 5. $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise
9. Although $\overline{\text{HCS}}$, $\overline{\text{HREAD}}$, and $\overline{\text{HWRITE}}$ can be totally asynchronous to the SMJ34020A, cycle responses to the signals are determined by local memory cycles.



‡ See clock stretch, page 21.

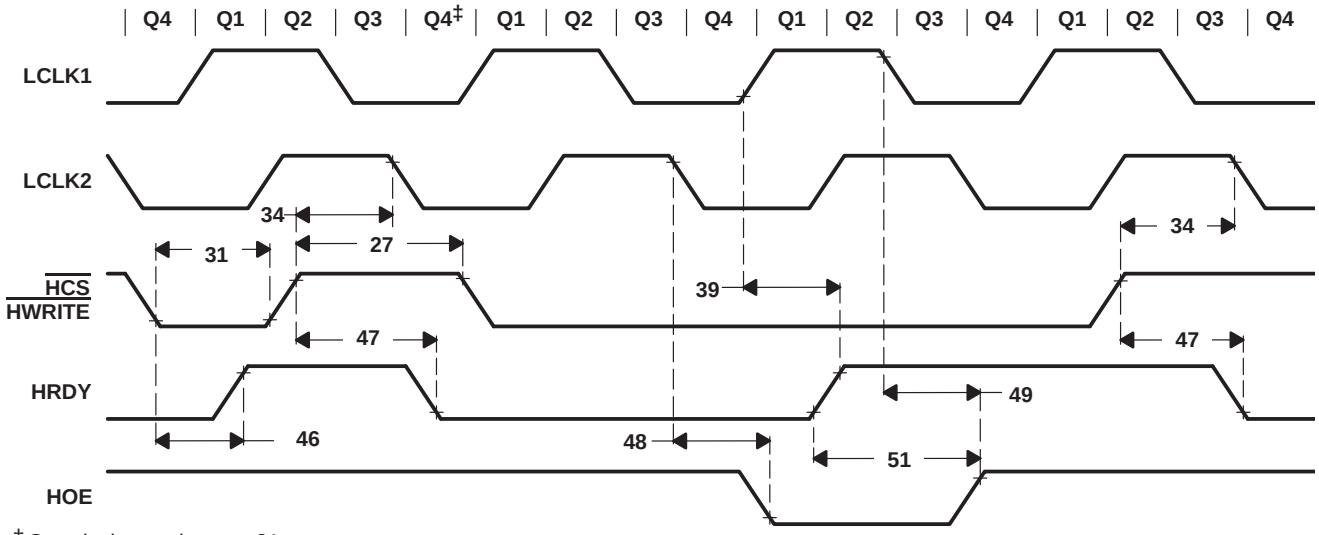
Figure 45. Host-Interface-Cycle Timing (Block-Read Cycle)

host interface timing responses (write cycle) (see Note 5 and Figure 46)

NO.	PARAMETER		34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
27	$t_{w(WRH)}$	Pulse duration, $\overline{\text{HWRITE}}$ high	28		25		ns
31	$t_{w(WRL)}$	Pulse duration, $\overline{\text{HWRITE}}$ low	18		15		ns
34	$t_{su(WRH-CK2L)}$	Setup time, $\overline{\text{HWRITE}}$ high or $\overline{\text{HCS}}$ high to LCLK2 no longer high	30†		25†		ns
39	$t_{d(CK1L-RYH)}$	Delay time from LCLK1↑ to HRDY high		t_Q+20		t_Q+18	ns
46	$t_{d(WRL-RYH)}$	Delay time from later of $\overline{\text{HCS}}$ or $\overline{\text{HWRITE}}$ low to HRDY high (TMS34020 ready)		25		20	ns
47	$t_{d(WRH-RYL)}$	Delay time from earlier of $\overline{\text{HCS}}$ or $\overline{\text{HWRITE}}$ high to HRDY low (end of write)		25		20	ns
48	$t_{d(CK2L-OEL)}$	Delay time from LCLK2↓ to $\overline{\text{HOE}}$ low		t_Q+15+s		$t_Q+13.5+s$	ns
49	$t_{d(CK1H-OEH)}$	Delay time from LCLK1↓ to $\overline{\text{HOE}}$ high		t_Q+15		$t_Q+13.5$	ns
51	$t_{d(RYH-OEH)}$	Delay time from HRDY↑ to $\overline{\text{HOE}}$ high		$2t_Q+15$		$2t_Q+13.5$	ns

† Setup time to ensure recognition of input on this clock edge.

NOTE 5: $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise



‡ See clock stretch, page 21.

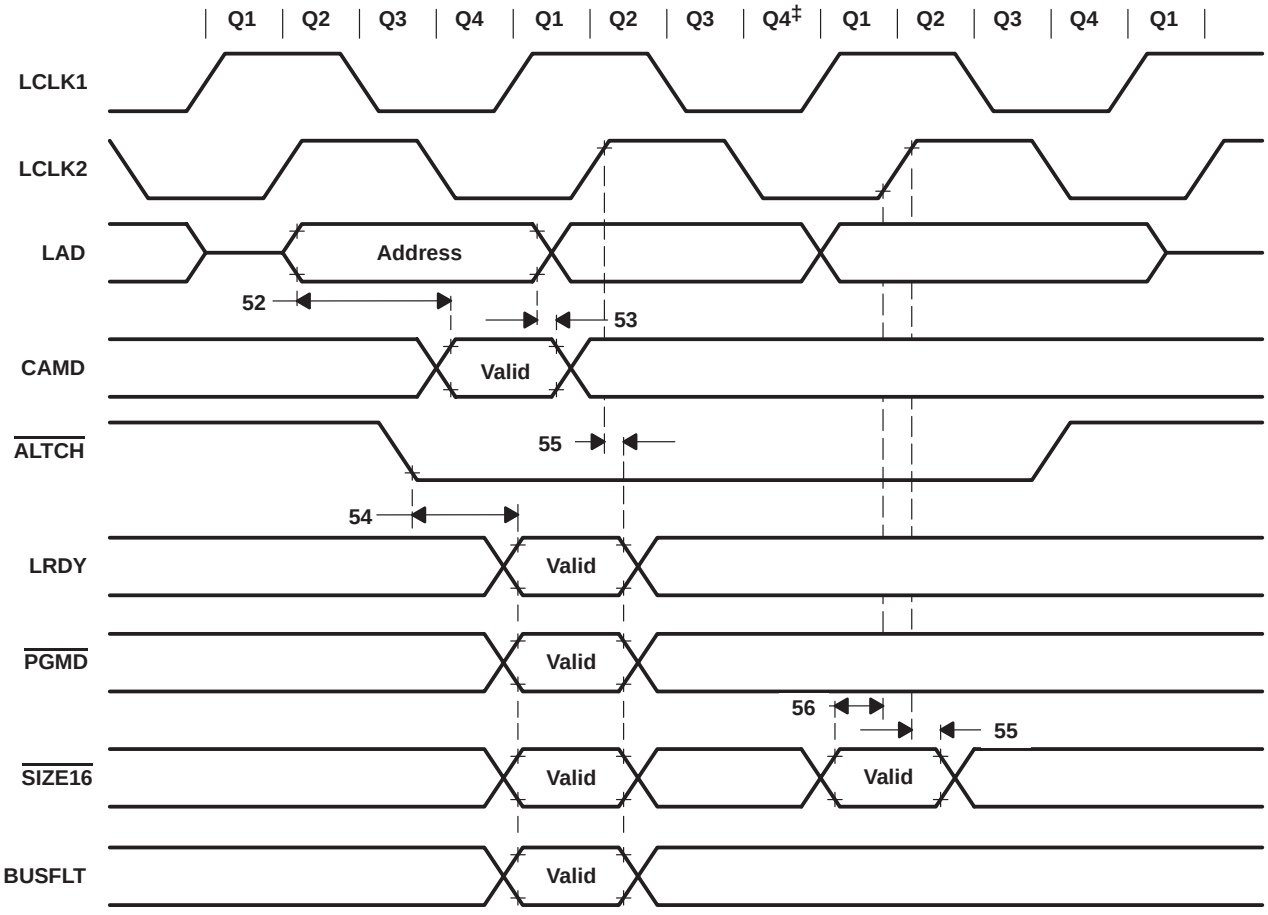
Figure 46. Host-Interface-Cycle Timing Responses (Write-Cycle)

local-bus timing: bus control inputs (see Note 5 and Figure 47)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
52	$t_{a(CMV-LAV)}$†	Access time, $\overline{CAMD}$ valid after address valid on LAD		$3t_Q-45$		$3t_Q-37$	ns
53	$t_{h(LA-CMV)}$†	Hold time, CAMD valid after address no longer valid on LAD	0		0		ns
54	$t_{a(BCV-ALL)}$†	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	$t_{su(BCV-CK2H)}$†	Setup time, $\overline{SIZE16}$ valid before LCLK2 no longer low	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.

NOTE 5: $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise



‡ See clock stretch, page 21.

Figure 47. Local-Bus Timing: Bus Control Inputs

local-bus timing: bus control inputs (see Note 5 and Figure 48)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
57	$t_{d(CK2H-ALL)}$	Delay time, $\overline{ALTCH}$ low after LCLK2 no longer low		t_Q+15		$t_Q+13.5$	ns
58	$t_{d(CK1L-ALH)}$	Delay time, $\overline{ALTCH}$ high after LCLK1 no longer high		t_Q+15		$t_Q+13.5$	ns
59	$t_{d(CK1H-LAV)}$	Delay time, LAD0−LAD31 address valid after LCLK1 no longer low		t_Q+22		t_Q+20	ns
60	$t_{h(LAV-CK2L)}$	Hold time, LAD0−LAD31 address valid after LCLK2 low	t_Q-15+s		t_Q-12+s		ns
61	$t_{d(CT-LAD)}$	Delay time, LAD0−LAD31 driven after earlier of DDIN no longer high or $\overline{CAS}$ no longer low or $\overline{TR/QE}$ no longer low	t_Q-5+s *		t_Q-5+s *		ns
62	$t_{h(LAV-CTV)}$	Hold time, LAD0−LAD31 read data valid after earlier of DDIN low or $\overline{RAS}$, $\overline{CAS}$, or $\overline{TR/QE}$ low	3.5		3.5		ns
63	$t_{d(CK2L-LAV)}$	Delay time, LAD0−LAD31 data valid after LCLK2 no longer high (write)		t_Q+22+s		t_Q+20+s	ns
64	$t_{h(CK2L-LAV)}$	Hold time, LAD0−LAD31 data valid after LCLK2 low (write)	t_Q-15		$t_Q-13.5$		ns
65	$t_{d(CK1H-RCV)}$	Delay time, RCA0−RCA12 row address valid after LCLK1 no longer low		t_Q+22		t_Q+22	ns
66	$t_{d(CK2L-RCV)}$	Delay time, LAD0−LAD31 column address valid after LCLK2 no longer high		t_Q+22+s		t_Q+20+s	ns
67	$t_{h(RCV-CK2L)}$	Hold time, RCA0-RCA12 address valid after LCLK2 low	t_Q-15		$t_Q-13.5$		ns
68	$t_{d(CK1H-DIH)}$	Delay time, DDIN high after LCLK1 no longer low		t_Q+15		$t_Q+13.5$	ns
69	$t_{d(CK1L-DIL)}$	Delay time, DDIN low after LCLK1 no longer high		t_Q+15		$t_Q+13.5$	ns
70	$t_{d(CK1H-DOL)}$	Delay time, $\overline{DDOUT}$ low after LCLK1 no longer low		t_Q+15		$t_Q+13.5$	ns
71	$t_{d(CK1L-DOH)}$	Delay time, $\overline{DDOUT}$ high after LCLK1 no longer high		t_Q+15		$t_Q+13.5$	ns
72	$t_{d(CK2L-DOL)}$	Delay time, $\overline{DDOUT}$ low after LCLK2 no longer high		t_Q+15+s		$t_Q+13.5+s$	ns
73	$t_{su(LAV-ALL)}$	Setup time, LAD0−LAD31 data valid before $\overline{ALTCH}$ no longer high	t_Q-16		$t_Q-13.5$		ns
74	$t_{en(DAV-DIH)}$	Enable time, data valid after DDIN high (see Note 10)		$2t_Q-20$		$2t_Q-17$	ns
75	$t_{dis(DAV-DIL)}$	Disable time, data in the high-impedance state after DDIN low (see Note 10)		t_Q-12+s *		t_Q-10+s *	ns

* The parameter is not production tested.

NOTES: 4. $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise

10. DDIN is used to control LAD bus buffers between the SMJ34020A and local memory. Parameter 74 references the time for these data buffers to go from the 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.

local-bus timing: bus control inputs (continued)

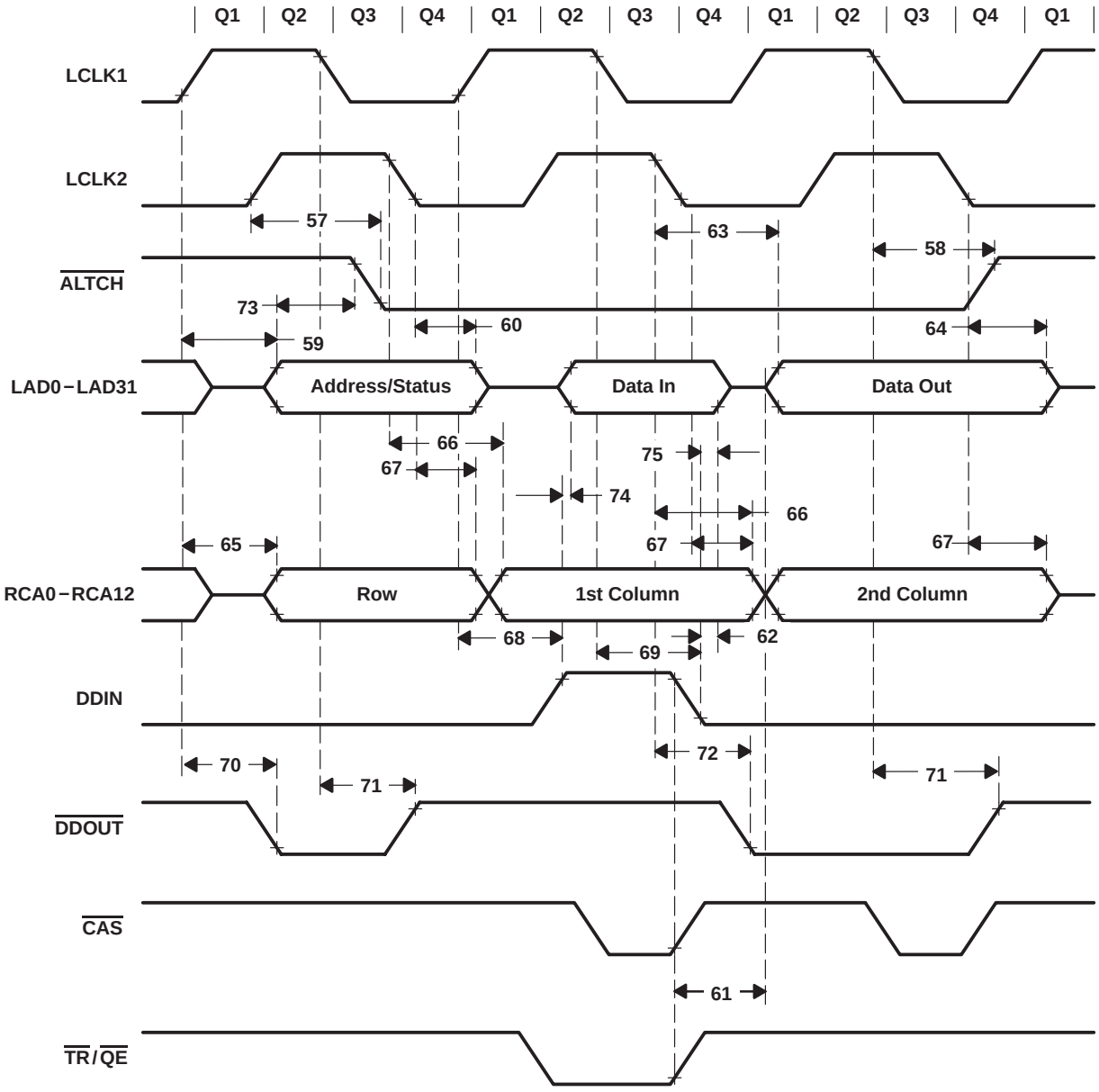


Figure 48. Local-Bus Timing: Bus Control Inputs (Continued)

local-bus timing: $\overline{RAS}$, $\overline{CAS0}$−$\overline{CAS3}$, $\overline{WE}$, $\overline{TR}$/$\overline{QE}$, and SF (see Notes 5 and 8 and Figure 49)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
62	$t_{h(LAV-CTV)}$	Hold time, LAD0−LAD31 read data valid after earlier of DDIN, low after $\overline{RAS}$, $\overline{CAS}$, or $\overline{TR}$/$\overline{QE}$ high	3.5		3.5		ns
76	$t_{d(CK1L-REL)}$	Delay time, $\overline{RAS}$ low after LCLK1 no longer high		t_Q+12+s		t_Q+10+ s	ns
77	$t_{d(CK1L-REH)}$	Delay time, $\overline{RAS}$ high after LCLK1 no longer high		t_Q+12		t_Q+10	ns
78	$t_{d(CK1H-CEL)}$	Delay time, $\overline{CAS}$ low after LCLK1 no longer low		t_Q+12		t_Q+10	ns
79	$t_{d(CK1L-CEH)}$	Delay time, $\overline{CAS}$ high after LCLK1 no longer high		t_Q+12		t_Q+10	ns
80	$t_{d(CK2L-WEL)}$	Delay time, $\overline{WE}$ low after LCLK2 no longer high		t_Q+15+s		t_Q+13.5+ s	ns
81	$t_{d(CK1L-WEH)}$	Delay time, $\overline{WE}$ high after LCLK1 no longer high		t_Q+15		t_Q + 15	ns
82	$t_{d(CK2L-TRL)}$	Delay time, $\overline{TR}$/$\overline{QE}$ low after LCLK2 no longer high		t_Q+15+s		t_Q+13.5+ s	ns
83	$t_{d(CK1L-TRH)}$	Delay time, $\overline{TR}$/$\overline{QE}$ high after LCLK1 no longer high		t_Q+15		t_Q+13.5	ns
84	$t_{d(CK1H-SFV)}$	Delay time, SF valid after LCLK1 no longer low		t_Q+22		t_Q+20	ns
85	$t_{d(CK2L-SFV)}$	Delay time, SF valid after LCLK2 no longer high		t_Q+22+s		t_Q+20+ s	ns
86	$t_{d(CK2L-SFZ)}$	Delay time, SF in the high-impedance state after LCLK2 no longer high		t_Q+22 *		t_Q+20 *	ns
87	$t_{su(ADV-REL)}$‡	Setup time, row address valid before $\overline{RAS}$ no longer high	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	$t_{su(RCV-CEL)}$	Setup time, column address valid before $\overline{CAS}$ no longer high	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	$t_{su(CAV-CEL)}$	Setup time, write data valid before $\overline{CAS}$ no longer high	t_Q−22		t_Q − 20		ns
92	$t_{h(CAV-CEH)}$	Hold time, write data valid after $\overline{CAS}$ no longer low	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 $\overline{CASL}$ no longer high		2t_Q−8		2t_Q −8	ns
95	$t_{a(LAV-RCV)}$	Access time, data-in valid after column address valid		3t_Q−20+s		3t_Q − 12	ns

‡ Parameters 87 and 88 also apply to $\overline{WE}$, $\overline{TR}$/$\overline{QE}$, and SF relative to $\overline{RAS}$.

* This parameter is not production tested.

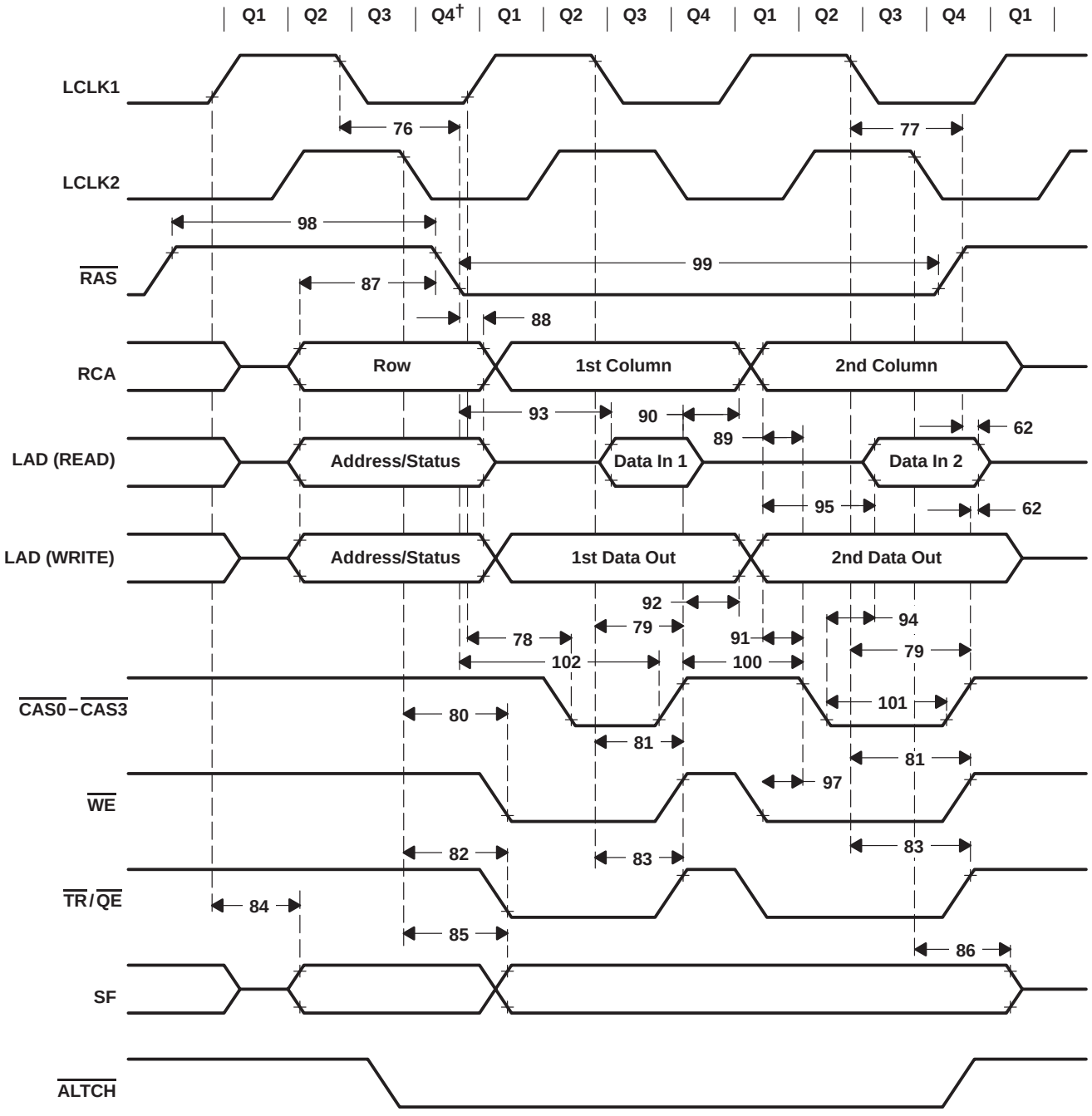
NOTES: 5. s = t_Q if using the clock stretch;
s = 0 otherwise
11. Parameter 96 has been eliminated.

local-bus timing: $\overline{RAS}$, $\overline{CAS0}$−$\overline{CAS3}$, $\overline{WE}$, $\overline{TR}$/$\overline{QE}$, and SF (see Notes 5 and 11 and Figure 49) (continued)

NO.			34020A-32		34020A-40		UNIT
			MIN	MAX	MIN	MAX	
97	$t_{su(WEL-CEL)}$	Setup time, write low before $\overline{CAS}$ no longer high (on write cycles)	$t_Q - 15$		$t_Q - 13.5$		ns
98	$t_{w(REH)}$	Pulse duration, $\overline{RAS}$ high	$4t_Q - 12 + s$		$4t_Q - 10 + s$		ns
99	$t_{w(REL)}$	Pulse duration, $\overline{RAS}$ low	$4nt_Q - 12 + s'$		$4nt_Q - 4 + s'$		ns
100	$t_{w(CEH)}$	Pulse duration, $\overline{CAS}$ high	$2t_Q - 12$		$2t_Q - 10$		ns
101	$t_{w(CEL)}$	Pulse duration, $\overline{CAS}$ low	$2t_Q - 12$		$2t_Q - 8$		ns
102	$t_{d(REL-CEH)}$	Delay time, $\overline{RAS}$ low to $\overline{CAS}$ no longer low	$4t_Q - 12 + s$		$4t_Q - 4 + s$		ns

NOTES: 5. $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise
s' is 2tQ when using the clock stretch since both the address cycle and read cycle of a Read-Modify-Write will be stretched.
11. Parameter 96 has been eliminated.

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



† See clock stretch, page 21.

Figure 49. Local-Bus Timing: $\overline{RAS}$, $\overline{CAS0}$−$\overline{CAS3}$, $\overline{WE}$, $\overline{TR}$/$\overline{QE}$, and SF

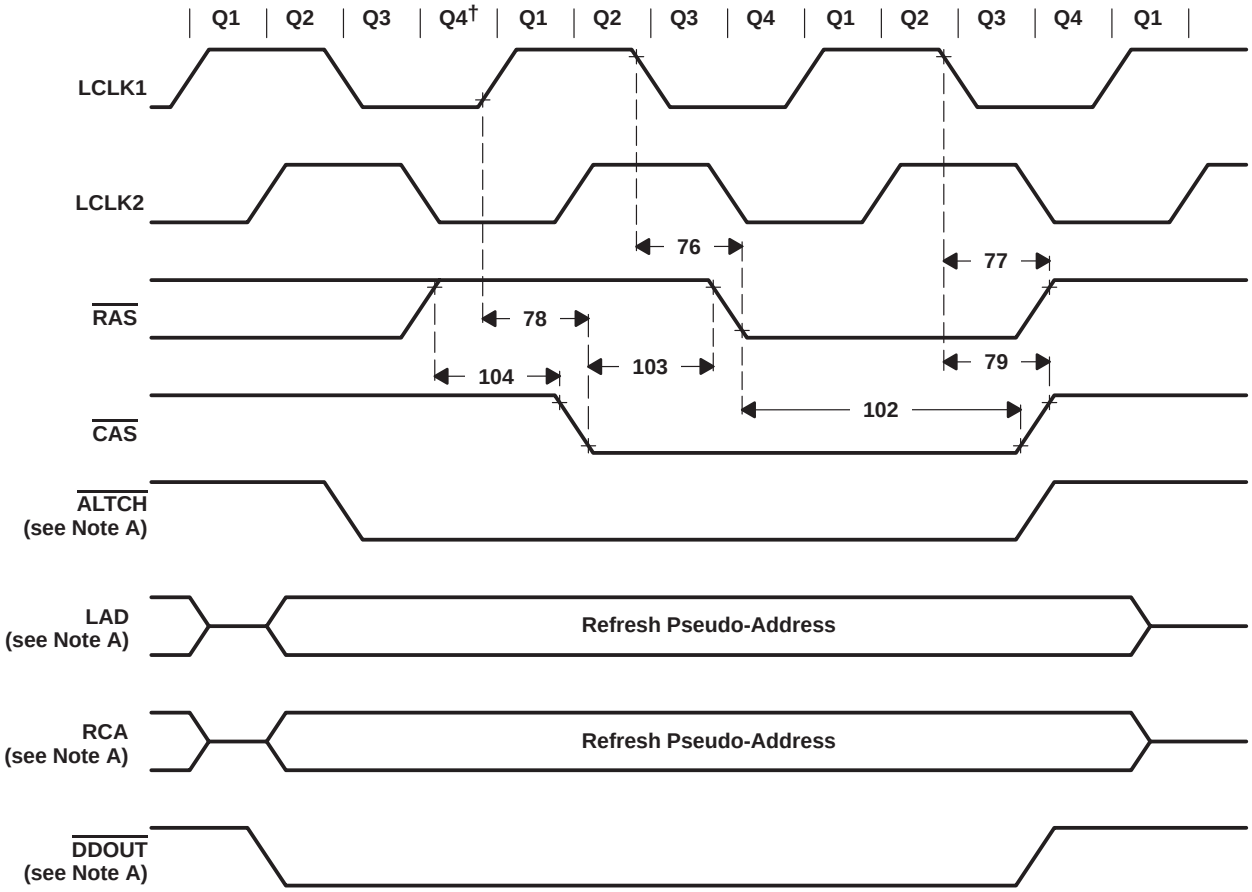
CBR refresh: $\overline{RAS}$ and $\overline{CAS0}$−$\overline{CAS3}$ (see Note 5 and Figure 49)

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.

NO.			'34020A-32		'34020A-40		UNIT
			MIN	MAX	MIN	MAX	
76	$t_{d(CK1L-REL)}$	Delay time, $\overline{RAS}$ low after LCLK1 no longer high		$t_Q + 12 + s$		$t_Q + 10 + s$	ns
77	$t_{d(CK1L-REH)}$	Delay time, $\overline{RAS}$ high after LCLK1 no longer high		$t_Q + 12$		$t_Q + 10$	ns
78	$t_{d(CK1H-CEL)}$	Delay time, $\overline{CAS}$ low after LCLK1 no longer low		$t_Q + 12$		$t_Q + 10$	ns
79	$t_{d(CK1L-CEH)}$	Delay time, $\overline{CAS}$ high after LCLK1 no longer high		$t_Q + 12$		$t_Q + 10$	ns
102	$t_{d(REL-CEH)}$	Delay time, $\overline{RAS}$ low to $\overline{CAS}$ no longer low	$4t_Q - 12 + s$		$4t_Q - 4 + s$		ns
103	$t_{d(CEL-REL)}$	Delay time, $\overline{CAS}$ low to $\overline{RAS}$ no longer high	$2t_Q - 15$		$2t_Q - 13.5$		ns
104	$t_{d(REH-CEL)}$	Delay time, $\overline{RAS}$ high to $\overline{CAS}$ no longer high	$2t_Q - 15 + s$		$2t_Q - 13.5 + s$		ns

NOTE 5: $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise

CBR refresh: $\overline{RAS}$ and $\overline{CAS0}$−$\overline{CAS3}$ (continued)



† See clock stretch, page 21.
NOTE A: $\overline{ALTCH}$, LAD, RCA, and $\overline{DDOUT}$ are shown for reference only.

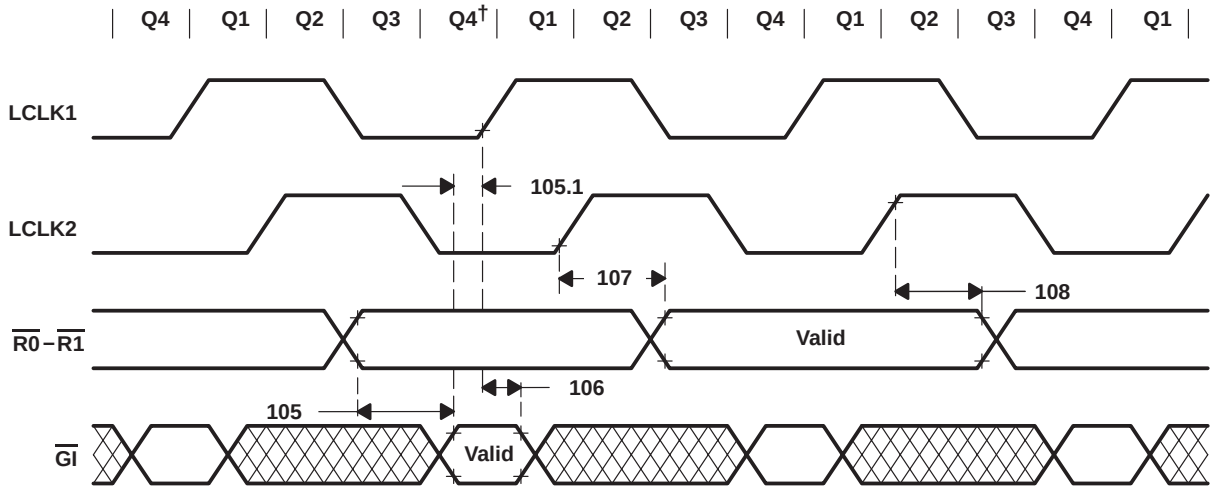
Figure 50. CBR Refresh: $\overline{RAS}$ and $\overline{CAS0}$−$\overline{CAS3}$

multiprocessor-interface timing: $\overline{GI}$, $\overline{ALTCH}$, $\overline{RAS}$, $\overline{R0}$ and $\overline{R1}$ (see Figure 51)

NO.			’34020A-32		’34020A-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 12)		$2t_Q-40$		$2t_Q-30$	ns
105.1	$t_{su(GIV-CK1H)}$	Setup time, $\overline{GI}$ valid before LCLK1 no longer low (see Note 12)	40		35		ns
106	$t_{h(CK1H-GIV)}$	Hold time, $\overline{GI}$ valid after LCLK1 no longer low	0		0		ns
107	$t_{d(CK2H-RQV)}$	Delay time, LCLK2 no longer low 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 12: These timings must be met to ensure that $\overline{GI}$ is recognized on this clock cycle.

For a SMJ34020A to gain control of the local bus during a given cycle, $\overline{GI}$ must be low at the start of Q1 (indicating that the bus arbitration logic is granting the bus to this processor).



† See clock stretch, page 21.

Figure 51. Multiprocessor-Interface Timing: $\overline{GI}$, $\overline{ALTCH}$, $\overline{RAS}$, $\overline{R0}$ and $\overline{R1}$

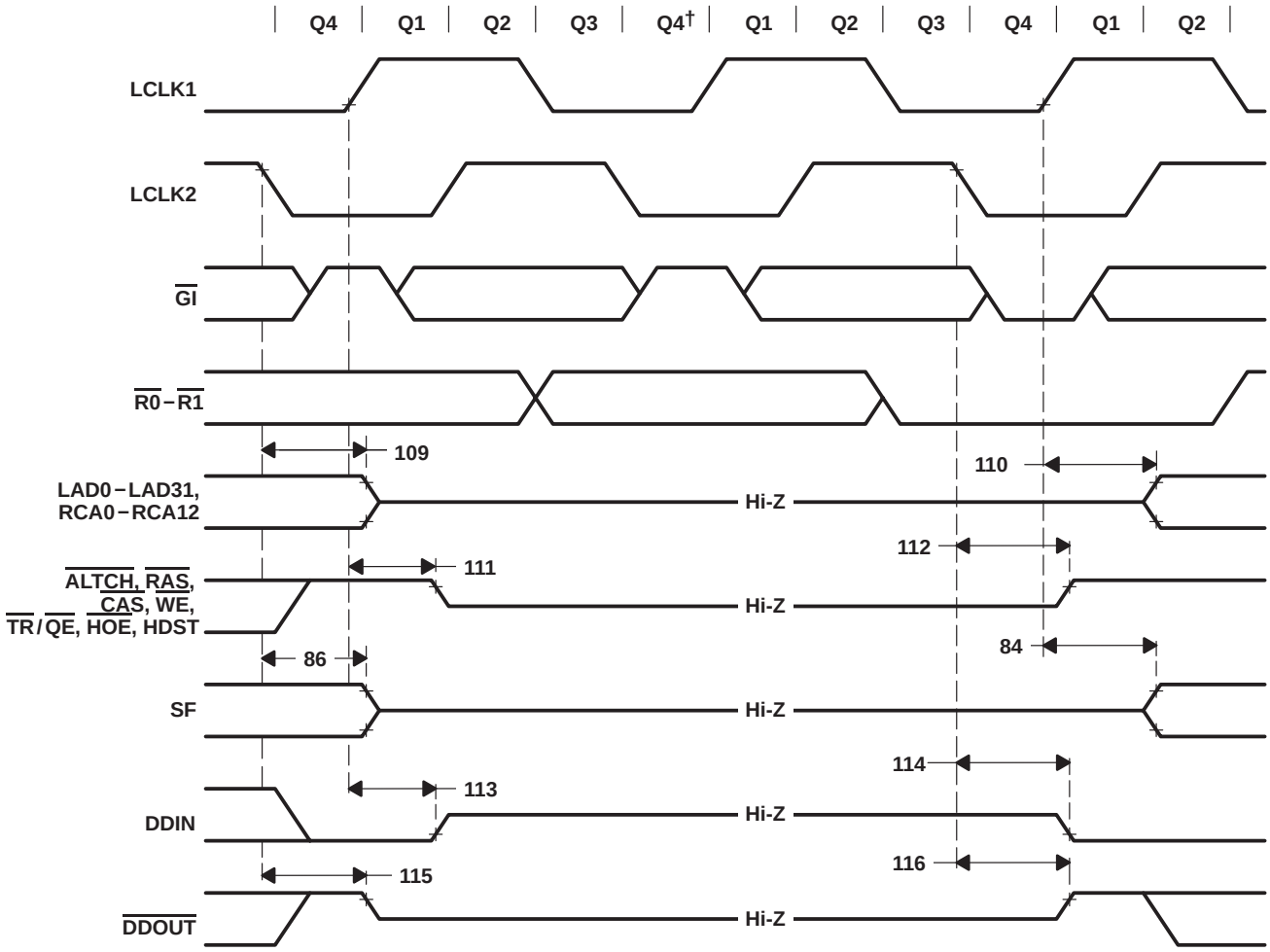
multiprocessor-interface timing: high-impedance signals (see Note 5 and Figure 52)

NO.			’34020A-32		’34020A-40		UNIT
			MIN	MAX	MIN	MAX	
84	$t_{d(CK1H-SFV)}$	Delay time, SF valid after LCLK1 no longer low		t_Q+22		t_Q+20	ns
86	$t_{d(CK2L-SFZ)}$	Delay time, SF in the high-impedance state after LCLK2 no longer high		t_Q+22+s *		t_Q+20+s*	ns
109	$t_{d(CK2L-ADZ)}$	Delay time, LAD and RCA in the high-impedance state after LCLK2 no longer high		t_Q+22+s *		t_Q+20+s *	ns
110	$t_{d(CK1H-ADV)}$	Delay time, LAD and RCA valid after LCLK1 no longer low		t_Q+22		t_Q+20	ns
111	$t_{d(CK1H-CTZ)}$	Delay time, $\overline{\text{ALTCH}}$, $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, $\overline{\text{WE}}$, $\overline{\text{TR}}/\overline{\text{QE}}$, $\overline{\text{HOE}}$, and HDST in the high-impedance state after LCLK1 no longer low		t_Q+15 †		$t_Q+13.5$ *	ns
112	$t_{d(CK2L-CTH)}$	Delay time, $\overline{\text{ALTCH}}$, $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, $\overline{\text{WE}}$, $\overline{\text{TR}}/\overline{\text{QE}}$, $\overline{\text{HOE}}$, and HDST in the high-impedance state after LCLK2 no longer high		t_Q+15+s		$t_Q+13.5+s$	ns
113	$t_{d(CK1H-DIZ)}$	Delay time, DDIN in the high-impedance state after LCLK1 no longer low		t_Q+15 *		$t_Q+13.5$ *	ns
114	$t_{d(CK2L-DIL)}$	Delay time, DDIN low after LCLK2 no longer high		t_Q+15+s		$t_Q+13.5+s$	ns
115	$t_{d(CK2L-DOZ)}$	Delay time, $\overline{\text{DDOUT}}$ in the high-impedance state after LCLK2 no longer high		t_Q+15+s *		$t_Q+13.5+s$ *	ns
116	$t_{d(CK2L-DOH)}$	Delay time, $\overline{\text{DDOUT}}$ high after LCLK2 no longer high		t_Q+15+s		$t_Q+13.5+s$	ns

* This parameter is not production tested.

NOTE 5: $s = t_Q$ if using the clock stretch;
$s = 0$ otherwise

multiprocessor-interface timing: high-impedance signals (continued)



† See clock stretch, page 21.

Figure 52. Multiprocessor-Interface Timing: High-Impedance Signals

video-shift-clock timing: SCLK (see Figure 53)

NO.			’34020A-32		’34020A-40		UNIT
			MIN	MAX	MIN	MAX	
117	$t_{c(SCK)}$	Cycle time, period of video serial clock SCLK	35	50	25	50	ns
118	$t_{w(SCKH)}$	Pulse duration, SCLK high	12		10		ns
119	$t_{w(SCKL)}$	Pulse duration, SCLK low	12		10		ns
120	$t_{t(SCK)}$	Transition time, (rise and fall) of SCLK	2 *	5 *	2 *	5 *	ns

* This parameter is not production tested.

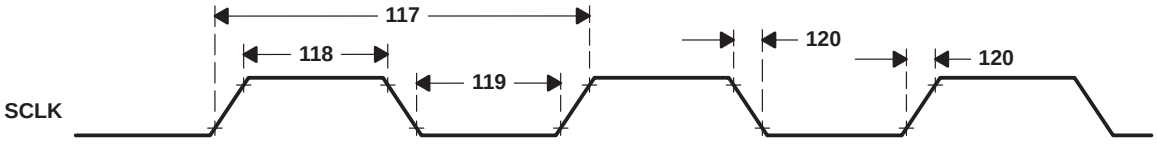


Figure 53. Video-Shift-Clock Timing: SCLK

video-interface timing: VCLK and video outputs (see Figure 54)

NO.			’34020A-32		’34020A-40		UNIT
			MIN	MAX	MIN	MAX	
123	$t_{c(VCK)}$	Cycle time, period of video input clock VCLK	62.5	100	62.5	100	ns
124	$t_{w(VCKH)}$	Pulse duration, VCLK high	28		28		ns
125	$t_{w(VCKL)}$	Pulse duration, 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 $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$/$\overline{\text{VBLNK}}$ or $\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$ low		40		40	ns
128	$t_{d(VCKL-HSH)}$	Delay time, VCLK low to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$, or $\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$ high		40		40	ns
129	$t_{h(VCKL-HSL)}$	Hold time, VCLK no longer high to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$, or $\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$ no longer high	0*		0 *		ns
130	$t_{h(VCKL-HSH)}$	Hold time, VCLK no longer high to $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$, or $\overline{\text{CBLNK}}$/$\overline{\text{VBLNK}}$ no longer low	0 *		0 *		ns

* This parameter is not production tested.

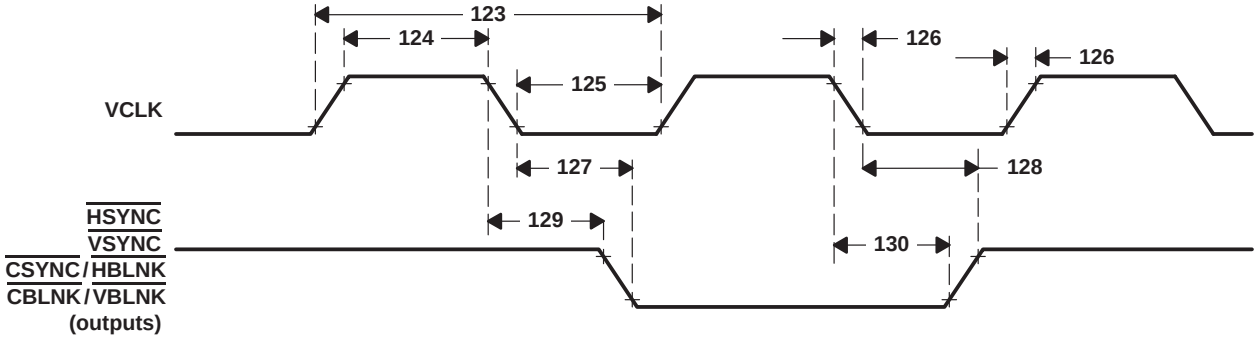
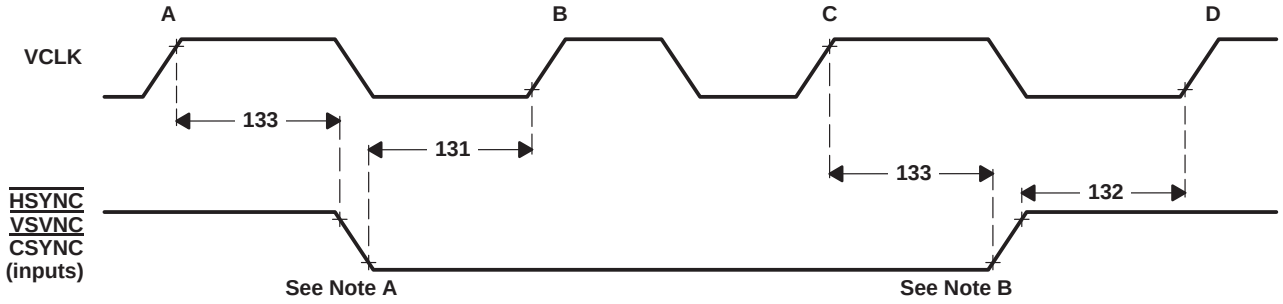


Figure 54. Video-Interface Timing: VCLK and Video Outputs

video-interface timing: external sync inputs (see Note 13 and Figure 55)

NO.			'34020A-32		'34020A-40		UNIT
			MIN	MAX	MIN	MAX	
131	$t_{su(SL-VCKH)}$	Setup time, $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$ low to VCLK no longer low	20		20		ns
132	$t_{su(SH-VCKH)}$	Setup time, $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$ high to VCLK no longer low	20		20		ns
133	$t_{h(VCKH-SV)}$	Hold time, $\overline{\text{HSYNC}}$, $\overline{\text{VSYNC}}$, $\overline{\text{CSYNC}}$ valid after VCLK high	20		20		ns

NOTE 13: Setup and hold times on asynchronous inputs are required only to assure recognition at indicated clock edges.



NOTES: A. If the falling edge of the sync signal occurs more than $t_{h(VCKH-SV)}$ after VCLK edge A and at least $t_{su(SL-VCKH)}$ before edge B, the transition is detected at edge B instead of edge A.
B. If the rising edge of the sync signal occurs more than $t_{h(VCKH-SV)}$ after VCLK edge C and at least $t_{su(SH-VCKH)}$ before edge D, the transition is detected at edge D instead of edge C.

Figure 55. Video-Interface Timing: External Sync Inputs

interrupt timing: $\overline{\text{LINT1}}$ and $\overline{\text{LINT2}}$ (see Figure 56)

NO.			'34020A-32		'34020A-40		UNIT
			MIN	MAX	MIN	MAX	
134	$t_{su(LINTL-CK2H)}$	Setup time, $\overline{\text{LINT1}}$ or $\overline{\text{LINT2}}$ low before LCLK2 no longer low	t_Q+45 †		t_Q+40 †		ns
135	$t_{w(LINTL)}$	Pulse duration, $\overline{\text{LINT1}}$ or $\overline{\text{LINT2}}$ low	$8t_Q$‡		$8t_Q$‡		ns

† Although $\overline{\text{LINT1}}$ and $\overline{\text{LINT2}}$ can be asynchronous to the SMJ34020A, this setup ensures recognition of the interrupt on this clock edge.
‡ This pulse duration minimum ensures that the interrupt is recognized by internal logic; however, the level must be maintained until it has been acknowledged by the interrupt service routine.

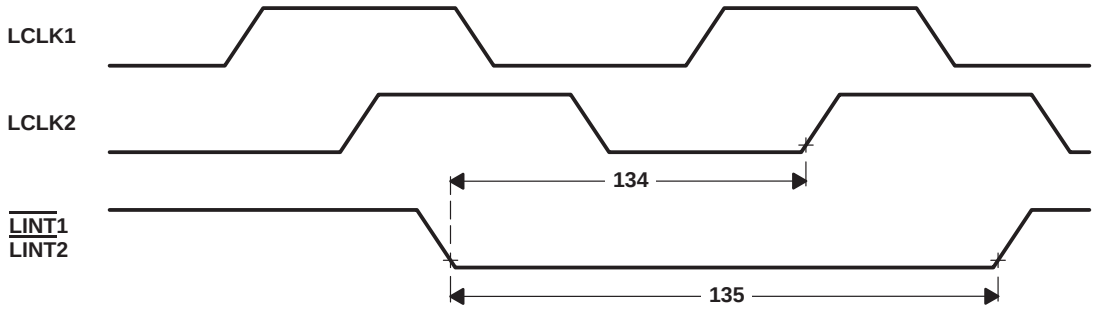


Figure 56. Interrupt Timing: $\overline{\text{LINT1}}$ and $\overline{\text{LINT2}}$

host-interrupt timing: $\overline{\text{HINT}}$ (see Figure 57)

NO.			’34020A-32		’34020A-40		UNIT
			MIN	MAX	MIN	MAX	
136	$t_{d(CK1H-HINTV)}$	Delay time, LCLK1 no longer low to $\overline{\text{HINT}}$ valid		30		25	ns

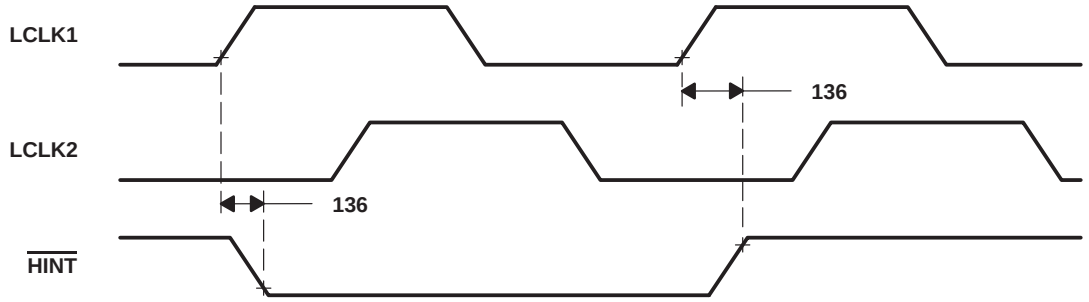


Figure 57. Host-Interrupt Timing: $\overline{\text{HINT}}$

emulator-interface timing (see Figure 58)

NO.			’34020A-32		’34020A-40		UNIT
			MIN	MAX	MIN	MAX	
137	$t_{su(EMV-CK1H)}$	Setup time, EMU0−EMU2 valid to LCLK1 no longer low	30		25		ns
138	$t_{h(EMV-CK1H)}$	Hold time, EMU0−EMU2 valid after LCLK1 no longer low	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_Q−15		t_Q−13.5		ns

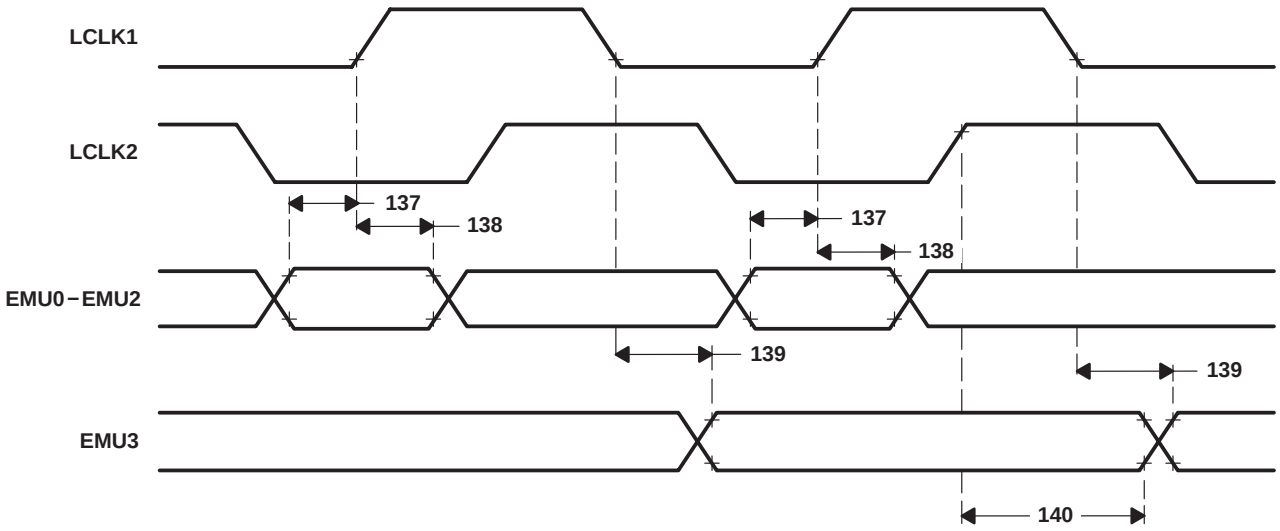
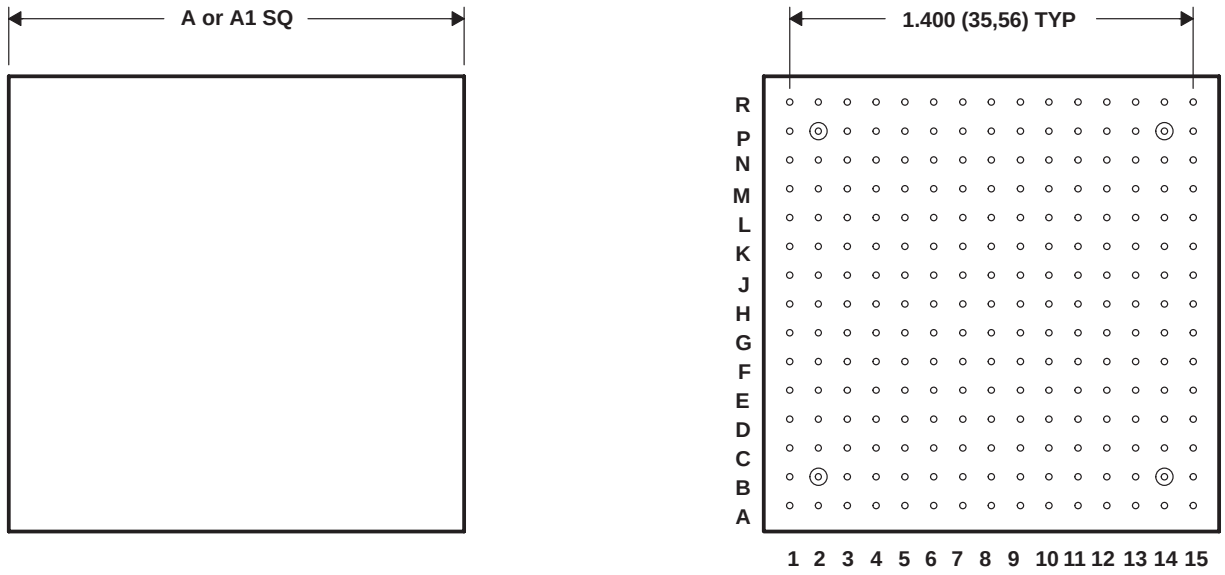
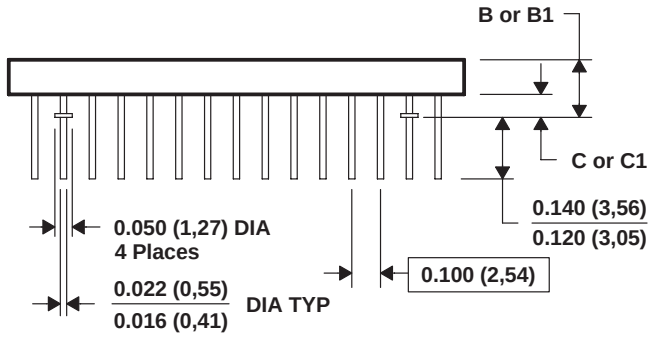


Figure 58. Emulator-Interface Timing

MECHANICAL DATA

GA-GB (S-CPGA-P15 X 15) **CERAMIC PIN GRID ARRAY PACKAGE**



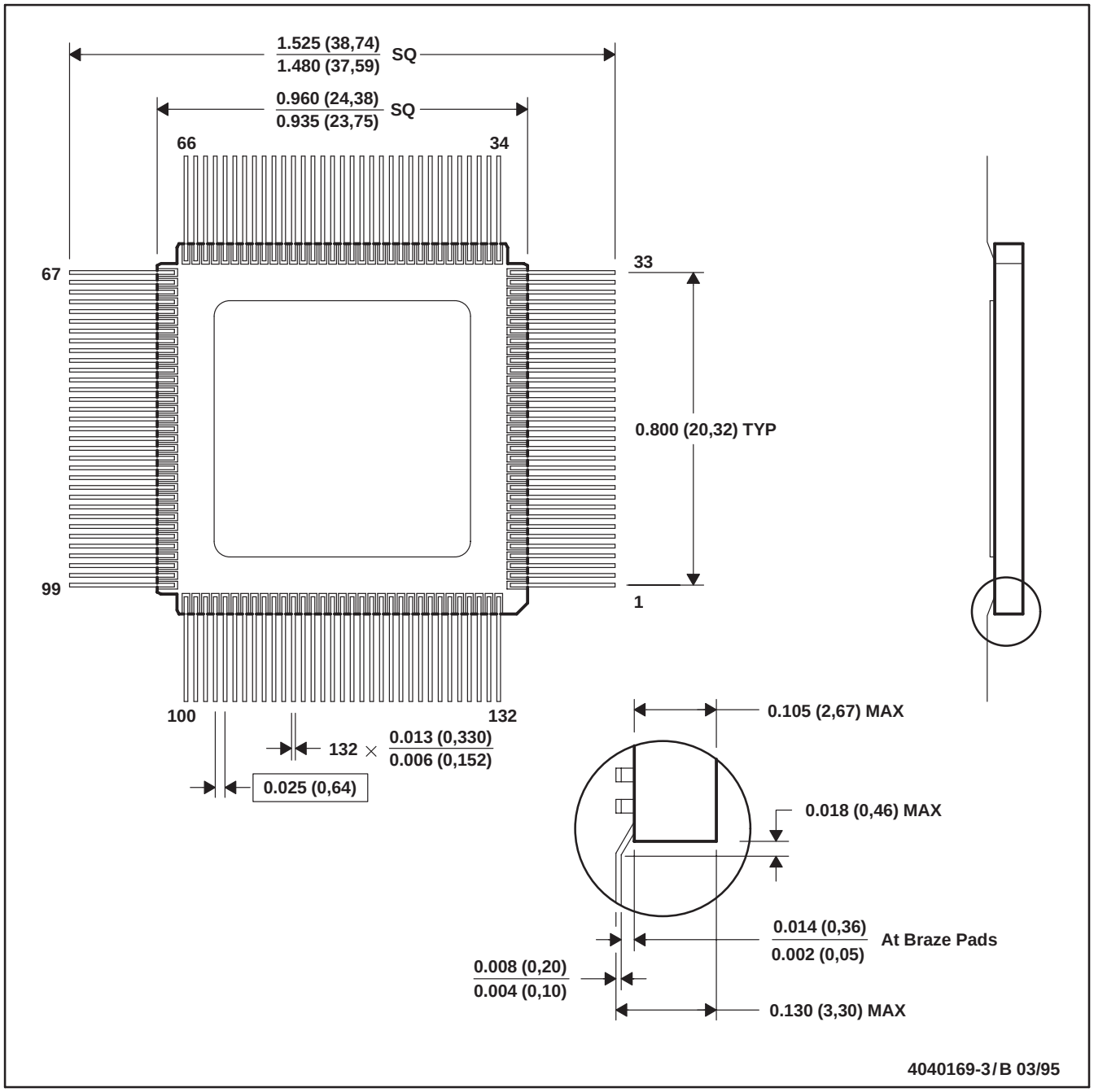


DIM	MIN	MAX	Notes
A	1.540 (39,12)	1.590 (40,38)	Large Outline
A1	1.480 (37,59)	1.535 (38,99)	Small Outline
B	0.110 (2,79)	0.205 (5,21)	Cavity Up
B1	0.095 (2,41)	0.205 (5,21)	Cavity Down
C	0.040 (1,02)	0.060 (1,52)	Cavity Up
C1	0.025 (0,63)	0.060 (1,52)	Cavity Down
MAXIMUM PINS WITHIN MATRIX − 225			

4040114-8/B 10/94

NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. Index mark may appear on top or bottom depending on package vendor.
D. Pins are located within 0.005 (0,13) radius of true position relative to each other at maximum material condition and within 0.015 (0,38) radius relative to the center of the ceramic.
E. This package can be hermetically sealed with metal lids or with ceramic lids using glass frit.
F. The pins can be gold plated or solder dipped.
G. Falls within MIL-STD-1835 CMGA7-PN and CMGA19-PN and JEDEC MO-067AG and MO-066AG, respectively

HT (S-CQFP-F132) **CERAMIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. This package is hermetically sealed with a metal lid.
D. The terminals are gold plated.
E. Falls within JEDEC MO-090 AB

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265