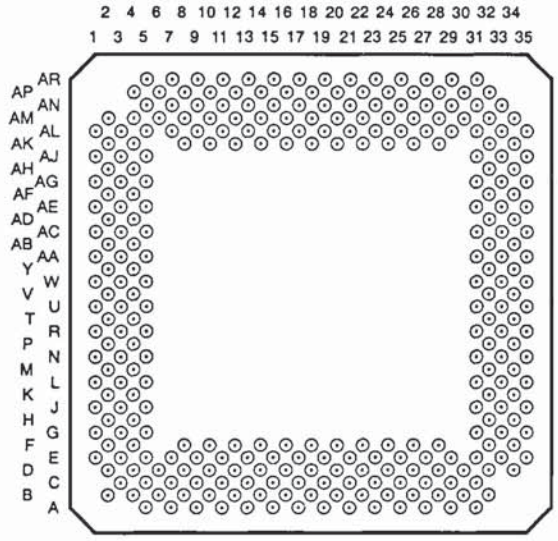
SPRS023 – JULY 1994

- Single-Chip Parallel MIMD DSP
- Over 2 Billion RISC-Like Operations per Second
- Master Processor (MP)
  - 32-Bit RISC Processor
  - IEEE-754 Floating-Point Capability
  - 4K-Byte Instruction Cache
  - 4K-Byte Data Cache
- 4 Parallel Processors
  - 32-Bit Advanced DSP (ADSP) Processors
  - 64-Bit Opcode Provides Many Parallel Operations per Cycle
  - 2K-Byte Instruction Cache and 8K Bytes of Data RAM per ADSP
- Transfer Controller (TC)
  - 64-Bit Data Transfers
  - 320M-Byte/s for '320C80-40
  - 400M-Byte/s for '320C80-50
  - 32-Bit Addressing
  - Direct DRAM / VRAM Interface With Dynamic Bus Sizing
  - Intelligent Queuing and Cycle Prioritization
- Video Controller (VC)
  - Provides Video Timing and VRAM Control
  - Dual Frame Timers for 2 Simultaneous Image Capture and / or Display Systems
- Big- or Little-Endian Operation

GF PACKAGE
(BOTTOM VIEW)



- 50K Bytes of On-Chip RAM
- 4G-Byte Address Space
- 25-ns Cycle Time for '320C80-40
- 20-ns Cycle Time for '320C80-50
- 3.3-V Operation With 5-V I/O
- IEEE-1149.1† Test Port (JTAG)

ADVANCE INFORMATION

description

The TMS320C80 multimedia video processor (MVP) is a single chip, MIMD (multiple instruction / multiple data) parallel processor capable of performing over 2 billion operations per second. It consists of a 32-bit RISC master processor with an 80-MFlop IEEE floating-point unit, four 32-bit advanced DSP (ADSP) processors, a transfer controller with 320M-byte/s for the TMS320C80-40 or 400M-byte/s for the TMS320C80-50 off-chip transfer rate, and a video controller. All the processors are tightly coupled via an on-chip crossbar that provides shared access to on-chip RAM. This performance and programmability make the 'C80 ideally suited for multimedia and imaging applications.

TXIIS023

† IEEE Std 1149.1–1990, IEEE Standard Test Access Port and Boundary-Scan Architecture

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

ADVANCE INFORMATION

PIN ASSIGNMENTS – NUMERICAL LISTING

PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME
A5	CT1	C21	V_{CC}	E33	$\overline{HSYNC0}$	L5	V_{SS}
A7	V_{CC}	C23	$\overline{W}$	E35	TCK	L31	V_{SS}
A9	$\overline{HACK}$	C25	$\overline{DBEN}$	F2	V_{CC}	L33	$\overline{TRST}$
A11	V_{SS}	C27	V_{SS}	F4	V_{SS}	L35	$\overline{XPT1}$
A13	$\overline{CAS7}$	C29	CAREA0	F8	V_{CC}	M2	V_{CC}
A15	$\overline{CAS5}$	C31	$\overline{CBLNK0}$ / $\overline{VBLNK0}$	F10	V_{SS}	M4	V_{SS}
A17	V_{CC}	D2	$\overline{RETRY}$	F12	V_{CC}	M32	V_{SS}
A19	V_{SS}	D4	V_{CC}	F14	PS0	M34	V_{CC}
A21	$\overline{RAS}$	D6	V_{SS}	F16	V_{SS}	N1	V_{CC}
A23	DSF	D8	AS0	F18	V_{CC5}	N3	A8
A25	V_{SS}	D10	$\overline{UTIME}$	F20	V_{CC}	N5	V_{SS}
A27	SCLK1	D12	V_{SS}	F22	V_{SS}	N31	V_{SS}
A29	V_{CC}	D14	$\overline{RESET}$	F24	V_{CC}	N33	TMS
A31	$\overline{EINT1}$	D16	REQ0	F26	V_{SS}	N35	V_{CC}
B2	No Connect	D18	V_{SS}	F28	V_{CC}	P2	A4
B4	BS1	D20	$\overline{CAS0}$	F32	V_{SS}	P4	A9
B6	V_{CC}	D22	FCLK1	F34	V_{CC}	P32	TDO
B8	PS1	D24	V_{SS}	G1	V_{CC}	P34	$\overline{XPT0}$
B10	REQ1	D26	CAREA1	G3	A2	R1	V_{SS}
B12	V_{CC}	D28	SCLK0	G5	A1	R3	V_{CC}
B14	$\overline{CAS6}$	D30	V_{SS}	G31	$\overline{EINT2}$	R5	V_{CC}
B16	$\overline{CAS3}$	D32	V_{CC}	G33	$\overline{CBLNK1}$ / $\overline{VBLNK1}$	R31	V_{CC}
B18	V_{CC}	D34	$\overline{VSYNC0}$	G35	V_{CC}	R33	V_{CC}
B20	$\overline{CAS1}$	E1	AS1	H2	STATUS0	R35	V_{SS}
B22	$\overline{TRG}$	E3	$\overline{FAULT}$	H4	A3	T2	A5
B24	V_{CC}	E5	V_{SS}	H32	$\overline{CSYNC1}$ / $\overline{HBLNK1}$	T4	A13
B26	$\overline{DDIN}$	E7	STATUS2	H34	TDI	T32	D62
B28	FCLK0	E9	READY	J1	STATUS1	T34	EMU0
B30	V_{CC}	E11	BS0	J3	V_{SS}	U1	V_{CC}
B32	$\overline{CSYNC0}$ / $\overline{HBLNK0}$	E13	V_{SS}	J5	V_{CC}	U3	A10
C3	V_{SS}	E15	$\overline{HREQ}$	J31	V_{CC}	U5	V_{CC5}
C5	STATUS3	E17	$\overline{CAS4}$	J33	V_{SS}	U31	V_{CC5}
C7	AS2	E19	$\overline{RL}$	J35	EMU1	U33	D61
C9	V_{SS}	E21	STATUS5	K2	STATUS4	U35	V_{CC}
C11	CT0	E23	V_{SS}	K4	A6	V2	V_{CC}
C13	PS2	E25	CLKOUT	K32	$\overline{VSYNC1}$	V4	V_{SS}
C15	V_{CC}	E27	$\overline{LINT4}$	K34	$\overline{HSYNC1}$	V32	V_{SS}
C17	CLKIN	E29	$\overline{EINT3}$	L1	A0	V34	V_{CC}
C19	$\overline{CAS2}$	E31	V_{SS}	L3	A7	W1	A11

PIN ASSIGNMENTS – NUMERICAL LISTING (CONTINUED)

PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME
W3	A18	AG1	A16	AL17	D20	AN29	D35
W5	V_{SS}	AG3	V_{SS}	AL19	D21	AN31	D45
W31	V_{SS}	AG5	V_{CC}	AL21	D24	AN33	V_{CC}
W33	D59	AG31	V_{CC}	AL23	V_{SS}	AP4	A27
W35	D63	AG33	V_{SS}	AL25	D29	AP6	V_{CC}
Y2	A12	AG35	D57	AL27	D32	AP8	D5
Y4	A19	AH2	A20	AL29	D38	AP10	D8
Y32	$\overline{XPT2}$	AH4	A30	AL31	V_{SS}	AP12	V_{CC}
Y34	D56	AH32	D44	AL33	D48	AP14	D13
AA1	V_{SS}	AH34	D54	AL35	D53	AP16	D17
AA3	V_{CC}	AJ1	V_{CC}	AM2	A24	AP18	V_{CC}
AA5	V_{CC}	AJ3	A31	AM4	V_{CC}	AP20	D26
AA31	V_{CC}	AJ5	V_{SS}	AM6	V_{SS}	AP22	D34
AA33	V_{CC}	AJ31	V_{SS}	AM8	D2	AP24	V_{CC}
AA35	V_{SS}	AJ33	D42	AM10	D6	AP26	D39
AB2	A14	AJ35	V_{CC}	AM12	V_{SS}	AP28	D41
AB4	A21	AK2	V_{CC}	AM14	D14	AP30	V_{CC}
AB32	D55	AK4	V_{SS}	AM16	D19	AP32	D47
AB34	D60	AK8	V_{CC}	AM18	V_{SS}	AR5	D0
AC1	V_{CC}	AK10	V_{SS}	AM20	D23	AR7	V_{CC}
AC3	A22	AK12	V_{CC}	AM22	D25	AR9	D7
AC5	V_{SS}	AK14	V_{SS}	AM24	V_{SS}	AR11	V_{SS}
AC31	V_{SS}	AK16	V_{CC}	AM26	D31	AR13	D11
AC33	D52	AK18	V_{CC5}	AM28	D33	AR15	D15
AC35	V_{CC}	AK20	V_{SS}	AM30	V_{SS}	AR17	V_{SS}
AD2	V_{CC}	AK22	D27	AM32	V_{CC}	AR19	V_{CC}
AD4	V_{SS}	AK24	V_{CC}	AM34	D50	AR21	D30
AD32	V_{SS}	AK26	V_{SS}	AN5	A29	AR23	D36
AD34	V_{CC}	AK28	V_{CC}	AN7	D1	AR25	V_{SS}
AE1	A15	AK32	V_{SS}	AN9	V_{SS}	AR27	D40
AE3	A26	AK34	V_{CC}	AN11	D9	AR29	V_{CC}
AE5	V_{SS}	AL1	A23	AN13	D12	AR31	D43
AE31	V_{SS}	AL3	A25	AN15	V_{CC}		
AE33	D51	AL5	V_{SS}	AN17	D18		
AE35	D58	AL7	D3	AN19	D22		
AF2	A17	AL9	D4	AN21	V_{CC}		
AF4	A28	AL11	D10	AN23	D28		
AF32	D46	AL13	V_{SS}	AN25	D37		
AF34	D49	AL15	D16	AN27	V_{SS}		

ADVANCE INFORMATION

ADVANCE INFORMATION

PIN ASSIGNMENTS – ALPHABETICAL LISTING

PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER
A0	L1	$\overline{CAS0}$	D20	D23	AM20	D62	T32
A1	G5	$\overline{CAS1}$	B20	D24	AL21	D63	W35
A2	G3	$\overline{CAS2}$	C19	D25	AM22	$\overline{DBEN}$	C25
A3	H4	$\overline{CAS3}$	B16	D26	AP20	$\overline{DDIN}$	B26
A4	P2	$\overline{CAS4}$	E17	D27	AK22	DSF	A23
A5	T2	$\overline{CAS5}$	A15	D28	AN23	$\overline{EINT1}$	A31
A6	K4	$\overline{CAS6}$	B14	D29	AL25	$\overline{EINT2}$	G31
A7	L3	$\overline{CAS7}$	A13	D30	AR21	$\overline{EINT3}$	E29
A8	N3	$\overline{CBLNK0}$ / $\overline{VBLNK0}$	C31	D31	AM26	EMU0	T34
A9	P4	$\overline{CBLNK1}$ / $\overline{VBLNK1}$	G33	D32	AL27	EMU1	J35
A10	U3	CLKIN	C17	D33	AM28	$\overline{FAULT}$	E3
A11	W1	CLKOUT	E25	D34	AP22	FCLK0	B28
A12	Y2	$\overline{CSYNC0}$ / $\overline{HBLNK0}$	B32	D35	AN29	FCLK1	D22
A13	T4	$\overline{CSYNC1}$ / $\overline{HBLNK1}$	H32	D36	AR23	$\overline{HACK}$	A9
A14	AB2	CT0	C11	D37	AN25	$\overline{HREQ}$	E15
A15	AE1	CT1	A5	D38	AL29	$\overline{HSYNC0}$	E33
A16	AG1	D0	AR5	D39	AP26	$\overline{HSYNC1}$	K34
A17	AF2	D1	AN7	D40	AR27	$\overline{LINT4}$	E27
A18	W3	D2	AM8	D41	AP28	PS0	F14
A19	Y4	D3	AL7	D42	AJ33	PS1	B8
A20	AH2	D4	AL9	D43	AR31	PS2	C13
A21	AB4	D5	AP8	D44	AH32	$\overline{RAS}$	A21
A22	AC3	D6	AM10	D45	AN31	READY	E9
A23	AL1	D7	AR9	D46	AF32	REQ0	D16
A24	AM2	D8	AP10	D47	AP32	REQ1	B10
A25	AL3	D9	AN11	D48	AL33	$\overline{RESET}$	D14
A26	AE3	D10	AL11	D49	AF34	$\overline{RETRY}$	D2
A27	AP4	D11	AR13	D50	AM34	$\overline{RL}$	E19
A28	AF4	D12	AN13	D51	AE33	SCLK0	D28
A29	AN5	D13	AP14	D52	AC33	SCLK1	A27
A30	AH4	D14	AM14	D53	AL35	STATUS0	H2
A31	AJ3	D15	AR15	D54	AH34	STATUS1	J1
AS0	D8	D16	AL15	D55	AB32	STATUS2	E7
AS1	E1	D17	AP16	D56	Y34	STATUS3	C5
AS2	C7	D18	AN17	D57	AG35	STATUS4	K2
BS0	E11	D19	AM16	D58	AE35	STATUS5	E21
BS1	B4	D20	AL17	D59	W33	TCK	E35
CAREA0	C29	D21	AL19	D60	AB34	TDI	H34
CAREA1	D26	D22	AN19	D61	U33	TDO	P32

PIN ASSIGNMENTS – ALPHABETICAL LISTING (CONTINUED)

PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER	PIN NAME	PIN NUMBER
TMS	N33	$\overline{\text{UTIME}}$	D10	$\overline{\text{W}}$	C23	$\overline{\text{XPT2}}$	Y32
$\overline{\text{TRG}}$	B22	$\overline{\text{VSYNC0}}$	D34	$\overline{\text{XPT0}}$	P34		
$\overline{\text{TRST}}$	L33	$\overline{\text{VSYNC1}}$	K32	$\overline{\text{XPT1}}$	L35		
V_{CC}	A7, A17, A29, B6, B12, B18, B24, B30, C15, C21, D4, D32, F2, F8, F12, F20, F24, F28, F34, G1, G35, J5, J31, M2, M34, N1, N35, R3, R5, R31, R33, U1, U35, V2, V34, AA3, AA5, AA31, AA33, AC1, AC35, AD2, AD34, AG5, AG31, AJ1, AJ35, AK2, AK8, AK12, AK16, AK24, AK28, AK34, AM4, AM32, AN15, AN21, AN33, AP6, AP12, AP18, AP24, AP30, AR7, AR19, AR29	V_{CC5}	F18, U5, U31, AK18	V_{SS}	A11, A19, A25, C3, C9, C27, D6, D12, D18, D24, D30, E5, E13, E23, E31, F4, F10, F16, F22, F26, F32, J3, J33, L5, L31, M4, M32, N5, N31, R1, R35, V4, V32, W5, W31, AA1, AA35, AC5, AC31, AD4, AD32, AE5, AE31, AG3, AG33, AJ5, AJ31, AK4, AK10, AK14, AK20, AK26, AK32, AL5, AL13, AL23, AL31, AM6, AM12, AM18, AM24, AM30, AN9, AN27, AR11, AR17, AR25		

ADVANCE INFORMATION

ADVANCE INFORMATION

Terminal Functions

NAME	I/O/Z†	DESCRIPTION
LOCAL MEMORY INTERFACE		
A31–A0	O	Address bus. These terminals output the 32-bit byte address of the external memory cycle. The address can be multiplexed for DRAM accesses.
AS2–AS0	I	Address shift selection. These signals determine how the column address appears on the address bus. Eight shift values are supported, including zero.
BS1–BS0	I	Bus-size selection. These signals indicate the bus size of the memory or other device being accessed, allowing dynamic bus sizing for data buses less than 64 bits wide, as indicated below: BS1 BS0 BUS SIZE 0 0 8 bits 0 1 16 bits 1 0 32 bits 1 1 64 bits
CT1–CT0	I	Cycle-timing selection. These input signals determine the timing of the current memory: CT1 CT0 CYCLE TIMING 0 0 Pipelined, 1 cycle/column 0 1 Nonpipelined, 1 cycle/column 1 0 2 cycle/column 1 1 3 cycle/column
D63–D0	I/O	Data bus. These signals transfer up to 64 bits of data per memory cycle into or out of the 'C80.
$\overline{\text{DBEN}}$	O	Data buffer enable. This signal drives the active-low output enables of bidirectional transceivers that can be used to buffer input and output data on D63–D0.
$\overline{\text{DDIN}}$	O	Data-direction indicator. This signal indicate the direction of the data that passes through the transceivers. When $\overline{\text{DDIN}}$ is low, the transfer is from external memory into the 'C80.
$\overline{\text{FAULT}}$	I	Fault. This input signal is driven low by external circuitry to inform the 'C80 that a fault occurred on the current memory row access.
PS2–PS0	I	Page-size indication. These signals indicate the page size of the memory device(s) being accessed by the current cycle. The 'C80 uses this to determine when to begin a new row access.
READY	I	Ready. This signal indicates that the external device is ready to complete the memory cycle. This signal is driven low by external circuitry to insert wait states into a memory cycle.
$\overline{\text{RL}}$	O	Row latch. The high-to-low transition of $\overline{\text{RL}}$ can be used to latch the valid 32-bit byte address that is present on A31–A0.
$\overline{\text{RETRY}}$	I	Retry. This signal is driven low by external circuitry to indicate that the addressed memory is busy. The 'C80 will begin the cycle again.
STATUS5–STATUS0	O	Status code. At row time, these signals indicate the type of cycle being performed. At column time, they identify the processor and type of request that initiated the cycle.
$\overline{\text{UTIME}}$	I	User-timing selection. This signal causes the timing of $\overline{\text{RAS}}$ and $\overline{\text{CAS7}}$–$\overline{\text{CAS0}}$ to be modified so that custom memory timings can be generated. During reset, $\overline{\text{UTIME}}$ selects the endian mode in which the 'C80 operates.
DRAM AND VRAM CONTROL		
$\overline{\text{CAS7}}$–$\overline{\text{CAS0}}$	O	Column-address strobes. These outputs drive the $\overline{\text{CAS}}$ inputs of DRAMs and VRAMs. The eight strobes provide byte write access to memory.
DSF	O	Special function. This signal selects special VRAM functions such as block write, load color register, and split-register transfer.
$\overline{\text{RAS}}$	O	Row-address strobe. This signal drives the $\overline{\text{RAS}}$ inputs of DRAMs and VRAMs.
$\overline{\text{TRG}}$	O	Transfer/output enable. During memory-read cycles, $\overline{\text{TRG}}$ is used as an output enable for DRAMs and VRAMs. During VRAM register-transfer cycles, $\overline{\text{TRG}}$ is used as a transfer enable.
$\overline{\text{W}}$	O	Write enable. This signal is driven low before $\overline{\text{CAS}}$ during write cycles. $\overline{\text{W}}$ controls the direction of the transfer during VRAM transfer cycles.

† I = input, O = output, Z = high impedance

Terminal Functions (Continued)

NAME	I/O/Z†	DESCRIPTION
		HOST INTERFACE
$\overline{\text{HACK}}$	O	Host acknowledge. The 'C80 drives this terminal low following an active $\overline{\text{HREQ}}$ to indicate that it has driven the local-memory-bus signals to the high-impedance state and is relinquishing the bus. $\overline{\text{HACK}}$ is driven high asynchronously following $\overline{\text{HREQ}}$ being detected inactive and the 'C80 resumes driving the bus.
$\overline{\text{HREQ}}$	I	Host request. An external device drives this input low to request ownership of the local-memory bus. When $\overline{\text{HREQ}}$ is high, the 'C80 owns and drives the bus. $\overline{\text{HREQ}}$ is internally synchronized to the 'C80's internal clock. $\overline{\text{HREQ}}$ is also used at reset to determine the power-up state of the MP. If $\overline{\text{HREQ}}$ is low at the rising edge of $\overline{\text{RESET}}$, the MP comes up running. If $\overline{\text{HREQ}}$ is high, the MP remains halted until the first interrupt occurrence on $\overline{\text{EINT3}}$.
REQ1, REQ0	O	Internal cycle request. These signals provide a two-bit code indicating the highest priority memory-cycle request that is being received by the TC. External logic can monitor these signals to determine if it is necessary to relinquish the local-memory bus to the 'C80. REQ1 REQ0 INTERNAL REQUEST 0 0 Low-priority packet transfer, trickle refresh, idle 0 1 High-priority packet transfer 1 0 Cache/DEA (direct external access) request, urgent packet transfer 1 1 VC SRT (serial-register transfer), urgent refresh, XPT (external packet transfer) or VCPT (VC packet transfer)
		SYSTEM CONTROL
CLKIN	I	Input clock. This signal generates the internal 'C80 clocks to which all processor functions (except the frame timers) are synchronous.
CLKOUT	O	Local output clock. This signal provides a way to synchronize external circuitry to internal timings. All 'C80 output signals (except the VC signals) are synchronous to this clock.
$\overline{\text{EINT1}}$, $\overline{\text{EINT2}}$, $\overline{\text{EINT3}}$	I	Edge-triggered interrupts. These signals allow external devices to interrupt the MP on one of three interrupt levels ($\overline{\text{EINT1}}$ is the highest priority). The interrupts are rising-edge triggered. $\overline{\text{EINT3}}$ also serves as an unhalt signal. If the MP is powered up halted, the first rising edge on $\overline{\text{EINT3}}$ causes the MP to unhalt and fetch its reset vector (the $\overline{\text{EINT3}}$ interrupt pending bit is not set in this case).
$\overline{\text{LINT4}}$	I	Level-triggered interrupt. This signal provides an active-low level-triggered interrupt to the MP. Its priority falls below that of the edge-triggered interrupts. Any interrupt request should remain low until it is recognized by the 'C80.
$\overline{\text{RESET}}$	I	Reset. This signal is driven low to reset the 'C80 (all processors). During reset, all internal registers are set to their initial state and all outputs are driven to their inactive or high-impedance levels. During the rising edge of $\overline{\text{RESET}}$, the MP reset mode and the 'C80's operating endian mode are determined by the levels of $\overline{\text{HREQ}}$ and $\overline{\text{UTIME}}$ terminals, respectively.
$\overline{\text{XPT2}}$–$\overline{\text{XPT0}}$	I	External packet transfer. These encoded inputs are used by external devices to request a high-priority XPT by the TC.
		EMULATION CONTROL
EMU0, EMU1‡	I/O	Emulation terminals. These terminals are used to support emulation host interrupts, special functions targeted at a single processor, and multiprocessor halt-event communications.
TCK‡	I	Test clock. This signal provides the clock for the 'C80's IEEE-1149.1 logic, allowing it to be compatible with other IEEE-1149.1 devices, controllers, and test equipment designed for different clock rates.
TDI‡	I	Test data input. This signal provides input data for all IEEE-1149.1 instructions and data scans of the 'C80.
TDO	O	Test data output. This signal provides output data for all IEEE-1149.1 instructions and data scans of the 'C80.
TMS‡	I	Test mode select. This signal controls the IEEE-1149.1 state machine.
$\overline{\text{TRST}}$§	I	Test reset. This signal resets the 'C80's IEEE-1149.1 module. When low, all boundary-scan logic is disabled, allowing normal 'C80 operation.

† I = input, O = output, Z = high impedance
‡ This terminal has an internal pullup and can be left unconnnected during normal operation.
§ This terminal has an internal pulldown and can be left unconnnected during normal operation.

ADVANCE INFORMATION

ADVANCE INFORMATION

Terminal Functions (Continued)

NAME	I/O/Z†	DESCRIPTION
		VIDEO INTERFACE
CAREA0, CAREA1	O	Composite area. These signals define a special area such as an overscan boundary. This area represents the logical OR of the internal horizontal and vertical area signals.
$\overline{\text{CBLNK0}}$ / $\overline{\text{VBLNK0}}$ $\overline{\text{CBLNK1}}$ / $\overline{\text{VBLNK1}}$	O	Composite blanking/vertical blanking. Each of these signals provides one of two blanking functions, depending on the configuration of the $\overline{\text{CSYNC}}$/$\overline{\text{HBLNK}}$ terminal: Composite blanking disables pixel display/capture during both horizontal and vertical retrace periods and is enabled when $\overline{\text{CSYNC}}$ is selected for composite sync video systems. Vertical blanking disables pixel display/capture during vertical retrace periods and is enabled when $\overline{\text{HBLNK}}$ is selected for separate-sync video systems. Initially these signals are configured as $\overline{\text{CBLNK0}}$, $\overline{\text{CBLNK1}}$
$\overline{\text{CSYNC0}}$ / $\overline{\text{HBLNK0}}$ $\overline{\text{CSYNC1}}$ / $\overline{\text{HBLNK1}}$	I/O/Z	Composite sync/horizontal blanking. These terminals can be programmed for one of two functions: Composite sync is for use on composite-sync video systems and can be programmed as an input, output, or high-impedance signal. As an input, the 'C80 extracts horizontal and vertical sync information from externally generated active-low sync pulses. As an output, the active-low composite sync pulses are generated from either external $\overline{\text{HSYNC}}$ and $\overline{\text{VSYNC}}$ signals or the 'C80's internal video timers. In the high-impedance state, the terminal is neither driven nor allowed to drive circuitry. Horizontal blank disables pixel display/capture during horizontal retrace periods in separate-sync video systems and can be used as an output only. Immediately following reset, these signals are configured as high-impedance $\overline{\text{CSYNC0}}$ and $\overline{\text{CSYNC1}}$.
FCLK0, FCLK1	I	Frame clock. These signals are derived from the external video system's dotclock and are used to drive the 'C80's video logic for frame timer 0 and frame timer 1.
$\overline{\text{HSYNC0}}$ $\overline{\text{HSYNC1}}$	I/O/Z	Horizontal sync. These signals control the video system. They can be programmed as input, output, or high-impedance signals. As an input, $\overline{\text{HSYNC}}$ synchronizes the video timer to externally generated horizontal sync pulses. As an output, $\overline{\text{HSYNC}}$ is an active-low horizontal sync pulse generated by the 'C80 on-chip frame timer. In the high impedance state, the terminal is not driven and no internal synchronization is allowed to occur. Immediately following reset, these signals are in the high-impedance state.
SCLK0, SCLK1	I	Serial data clock. These clock inputs are used by the 'C80's SRT controller to track the VRAM tap point when using midline reload. SCLK0 and SCLK1 should be the same signals that clock the serial register on the VRAMs controlled by frame timer 0 and frame timer 1, respectively.
$\overline{\text{VSYNC0}}$ $\overline{\text{VSYNC1}}$	I/O/Z	Vertical sync. These signals control the video system. They can be programmed as an inputs, outputs, or high-impedance signals. As inputs, $\overline{\text{VSYNCx}}$ synchronizes the frame timer to externally generated vertical sync pulses. As outputs, $\overline{\text{VSYNCx}}$ are active-low vertical-sync pulses generated by the 'C80 on-chip frame timer. In the high-impedance state, the terminal is not driven and no internal synchronization is allowed to occur. Immediately following reset, this signal is in the high-impedance state.
		POWER
V_{SS}‡	I	Ground. Electrical ground inputs
V_{CC}‡	I	Power. Nominal 3.3-V power supply inputs
V_{CC5}‡	I	5 V power. Nominal 5-V power supply inputs

† I = input, O = output, Z = high impedance
‡ For proper operation, all V_{CC} and V_{SS} terminals must be connected externally.

architecture

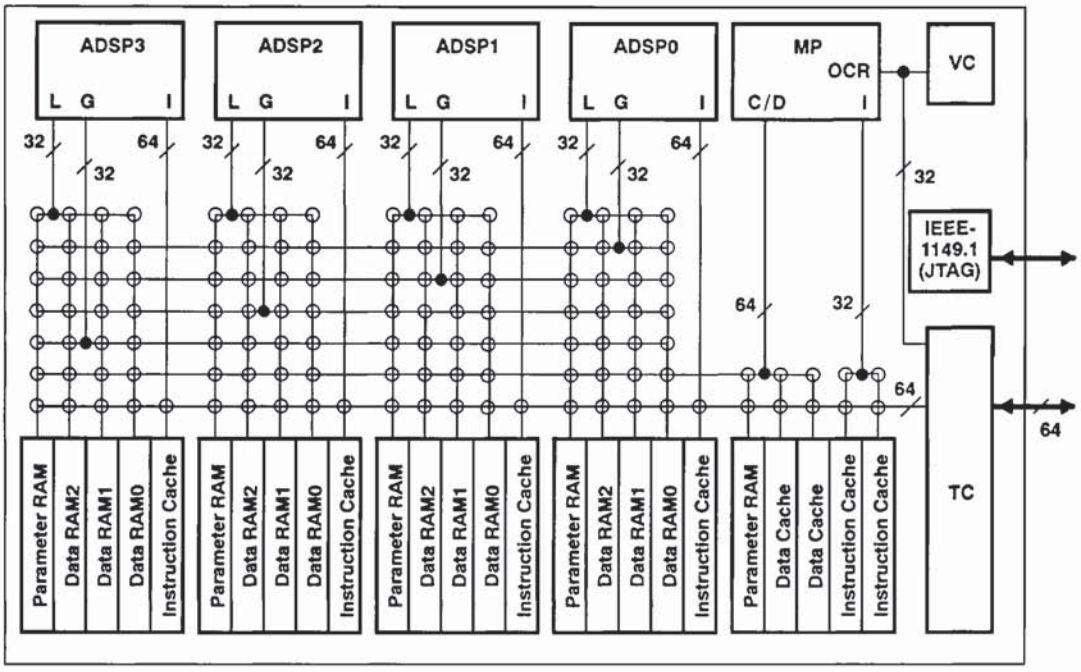


Figure 1. Block Diagram Showing Datapaths

Figure 1 shows the major components of the 'C80: the master processor (MP), the advanced digital signal processors (ADSPs), the transfer controller (TC), the video controller (VC), and the IEEE-1149.1 emulation interface. Shared access to on-chip RAM is achieved through the crossbar. Crossbar connections are represented by ○. Each ADSP can perform three accesses per cycle through its local (L), global (G), and instruction (I) ports. The MP can access two RAMs per cycle through its crossbar data (C/D) and instruction (I) ports, and the TC can access one RAM through its crossbar interface. Up to 15 simultaneous accesses are supported in each cycle. Addresses can be changed every cycle, allowing the crossbar matrix to be changed on a cycle-by-cycle basis. Contention between processors for the same RAM in the same cycle is resolved by a round-robin priority scheme. In addition to the crossbar, a 32-bit datapath exists between the MP and the TC and VC. This allows the MP to access TC and VC control registers that are memory mapped into the MP's memory space.

The 'C80 has a 4G-byte address space as shown in Figure 2. The lower 32M bytes are used to address internal RAM and memory-mapped registers.

architecture (continued)

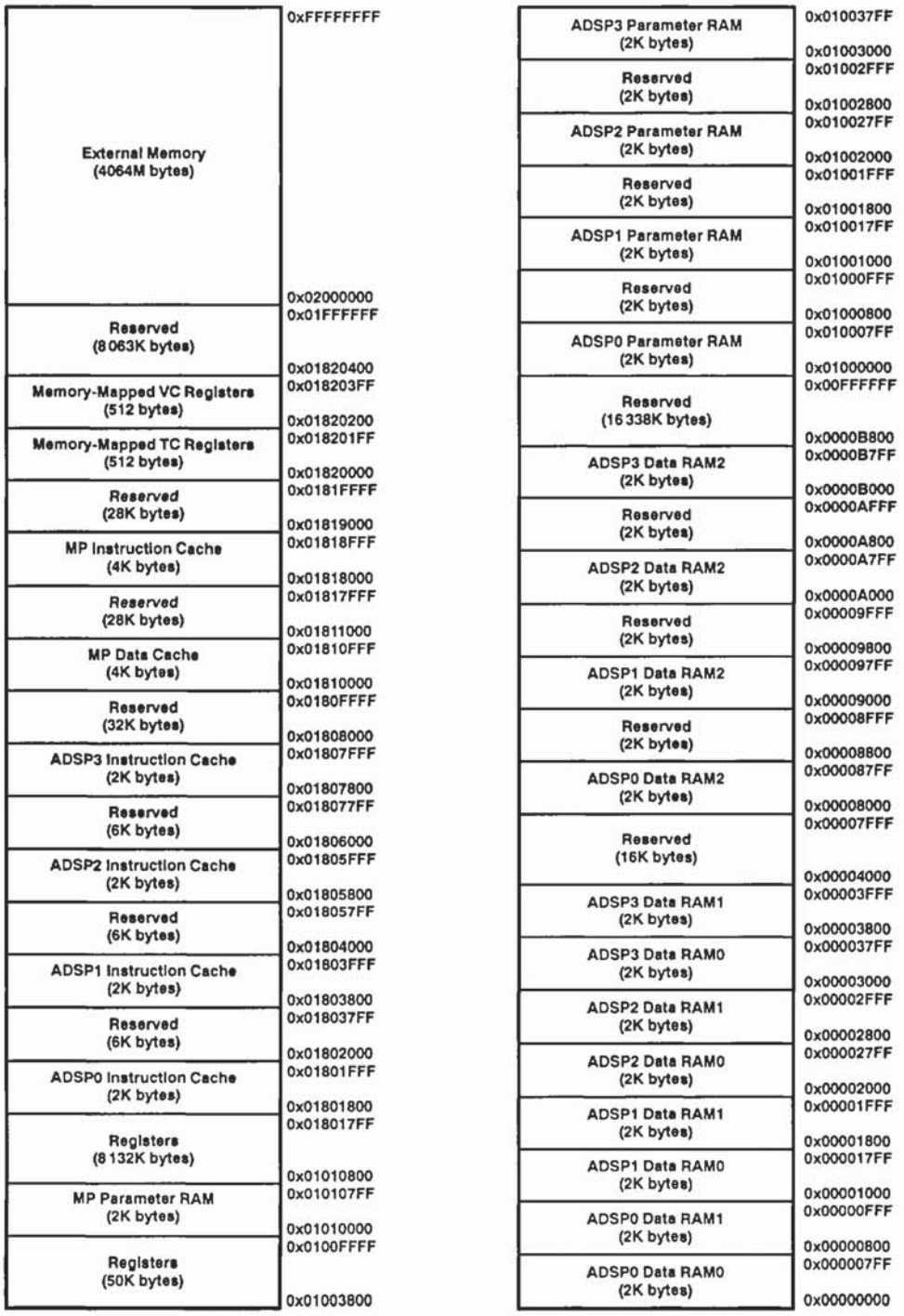


Figure 2. Memory Map

master processor (MP) overview

The MP is a 32-bit RISC (reduced instruction set computer) with an integral IEEE-754 floating-point unit. As with other RISC processors, all accesses to memory are performed with load and store instructions, and most integer and logical operations are performed on registers in a single cycle. The floating-point instructions are pipelined; although a single-precision instruction such as a floating-point multiply takes three cycles to complete, such an instruction can be started on each clock cycle. Likewise, double-precision instructions, such as square root, that take 28 cycles to complete can start on any clock cycle. Floating-point-unit operations use the same register file as the integer and logic unit. A register scoreboard ensures that correct register-access sequences are maintained.

Instructions and data are both fetched from on-chip caches, each 4K bytes in size. The control for these caches is an integral part of the MP design. The MP is able to access the on-chip memories by using the crossbar network.

The MP is structurally designed for efficient execution of C code. As an example, the MP contains an R0 register, often called a zeroing register, used by C. Also, the MP instruction set is tailored to contain many of the C executables found in compiler technology.

Figure 3 shows the block diagram for the master processor.

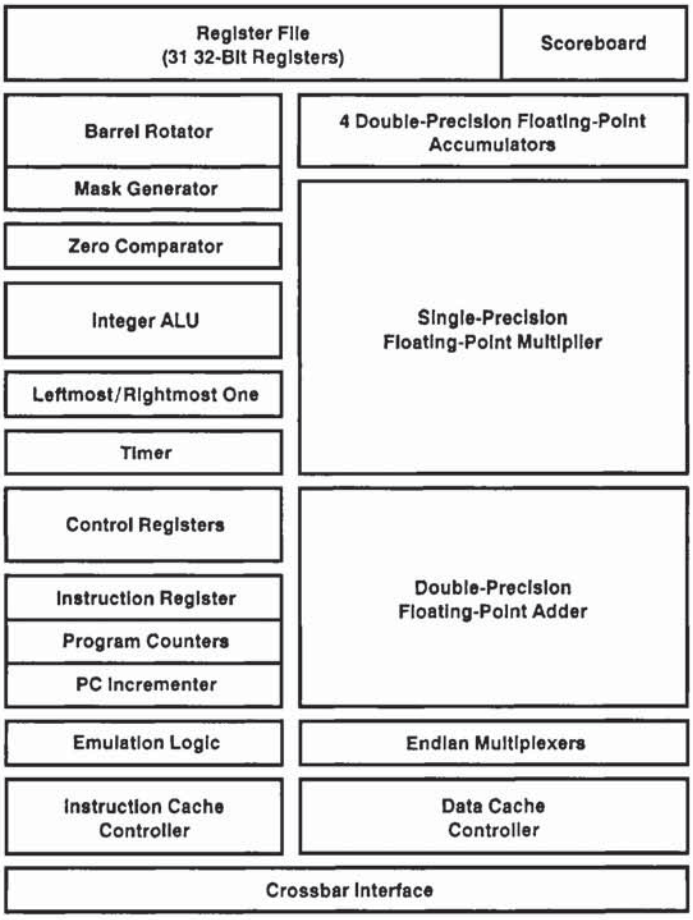


Figure 3. Master Processor Architecture

master processor (MP) overview (continued)

Key features of the MP include:

- A 32-bit RISC processor
  - Load/store architecture
  - 3-operand arithmetic and logical instructions
- Thirty-one 32-bit general-purpose registers
- Four double-precision floating-point vector accumulators
- IEEE-754 floating-point hardware
- 4K-byte instruction cache
- 4K-byte data cache
- Data and instruction cache characteristics that include:
  - 4-way set associative
  - LRU replacement
  - Data writeback
  - No bus snooping or bus watching
- 2K-byte parameter RAM (not cached)
- Delayed branches with option to annul delay-slot instructions
- Explicit compare instructions – no dedicated status register
- Register and accumulator scoreboard
- 15-bit or 32-bit immediate constants
- Vector floating-point instructions
  - Initiate a floating-point operation and a parallel load or store in one instruction
  - Multiply and accumulate
- Scalable timer
- Leftmost-one and rightmost-one logic
- Performance at $30 \leq N \leq 50$ MHz internal frequency:
  - $2 \times N$ MFLOPS peak (80 MFLOPS at 40 MHz)
  - N MIPS (40 MIPS at 40 MHz)
  - Over $2600 \times N$ Dhrystones (104000 Dhrystones at 40 MHz)
- Control registers used for vector data loads or stores, emulation, exceptions, and cache control
- 32-bit address space for bytes

advanced digital signal processor (ADSP) overview

The 'C80's ADSP is a programmable DSP-like 32-bit integer processor with a 64-bit instruction word that is optimized for imaging and graphics applications. It supports the filtering and frequency domain operations required for image processing. The ADSP can execute in parallel a multiply, an ALU operation (such as a shift-and-add), and two memory accesses within a single instruction.

The ADSP has a three-input ALU that supports all 256 Boolean combinations of three inputs and many combinations of arithmetic and Boolean functions. Data merging and bit-to-byte, bit-to-halfword, and bit-to-word translations are supported by hardware along the input data path to the ALU. These merging and translation operations allow the ADSP to accelerate graphics applications such as windowing environments. The internal parallelism allows a single ADSP to achieve over 500 million operations per second for certain algorithms.

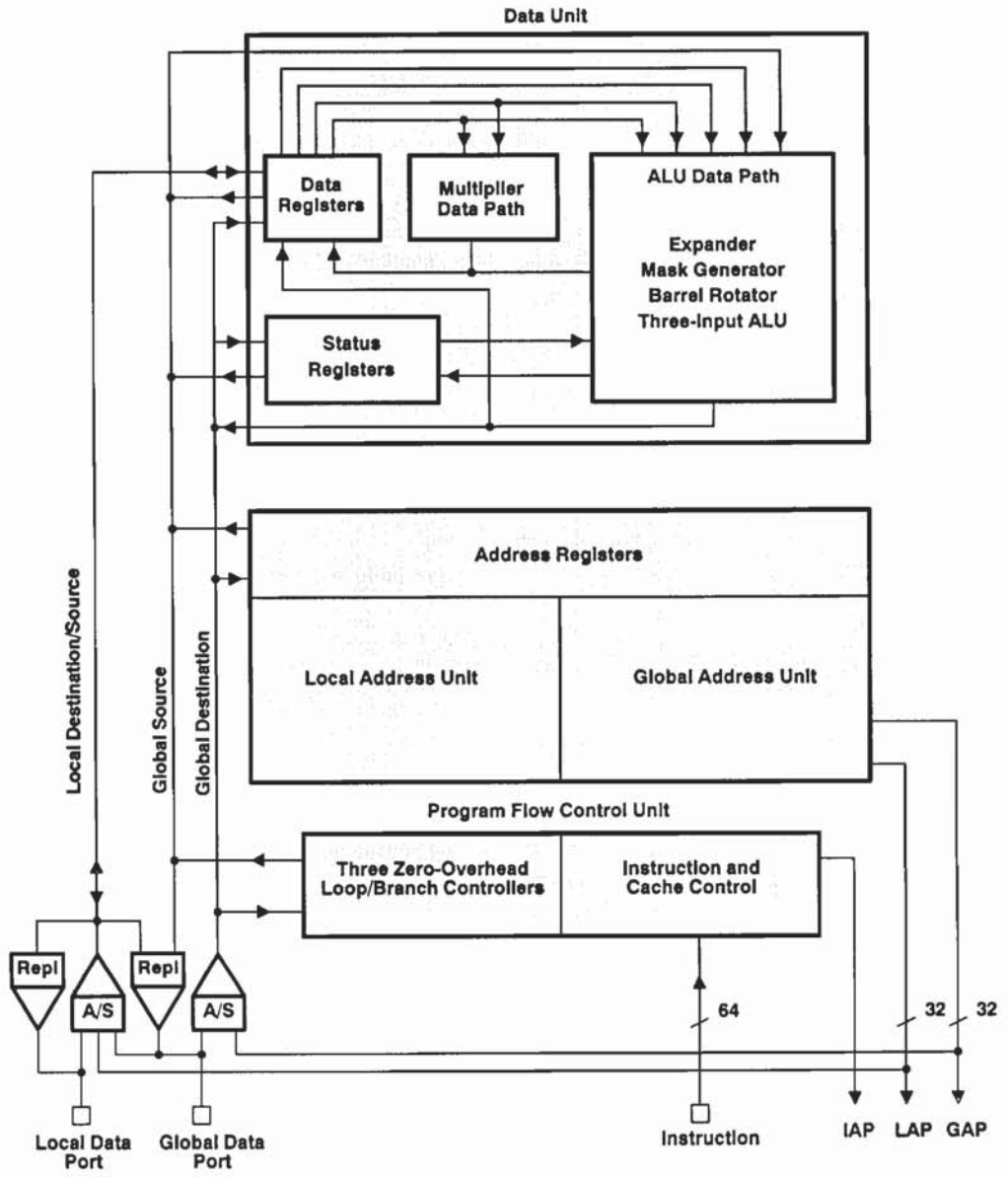
key features of ADSP

Key features of the ADSP include:

- 64-bit instruction word supports many parallel operations, such as a multiply, an ALU operation, and two memory accesses in a single cycle
- 3-stage pipeline provides fast instruction cycle
- Registers
  - 8 data, 10 address, and 6 index registers
  - 20 other user-visible registers
- Data unit
  - 16 × 16 integer multiplier (optional 8 × 8 multiplies)
  - Splittable 3-input ALU
  - 32-barrel rotator
  - Mask generator
  - Multiple-status flag expander facilitates translations to and from 1-bit-per-pixel space. It also supports transparency, max, min, saturation, z-buffering, and patterning.
  - Conditional operations to reduce branch requirements and delays; operations include both conditional assignment of data unit result(s) (16 condition codes) and conditional source selection (based on negative status bit).
  - Special processing hardware such as leftmost 1 and rightmost 1 detection and leftmost-bit-change and rightmost-bit-change detection
- Memory addressing
  - 2 address units (global and local), allowing up to two 32-bit memory accesses in parallel with data unit operations
  - 12 basic address modes (variations of immediate and indexed addressing)
  - Byte, halfword, and word addressability
  - 8-, 16-, and 32-bit data (or pixel) sizes
  - Loads of 8-bit or 16-bit data are either sign or zero-extended to 32 bits
  - Indexed addressing can be scaled according to data size
  - Big- and little-endian addressing supported
  - Conditional assignment for loads (memory-to-register transfers) based on 1 of 16 condition codes
  - Conditional source selection for stores (register-to-memory transfers) based on negative status bit

ADVANCE INFORMATION

key features of ADSP (continued)



Legend:
IAP = Instruction address port
LAP = Local address port
GAP = Global address port
Repl = Replicate hardware
A/S = Align-/sign-extend hardware

Figure 4. ADSP Block Diagram

key features of ADSP (continued)

- Program flow
  - Three hardware loop controllers that support zero-over-head looping and/or branching, three nest loops, one loop with multiple end points, and many other flexible looping combinations
  - The program counter (PC register) is mapped into the register file. Either the ALU or the global address unit can write to the PC register conditionally or unconditionally to cause a branch or subroutine call.
  - Interrupts for message passing and context switching
  - Instruction-cache management for accelerating program execution on the ADSP
- Run-time parallel-programming-environment support
- Algebraic assembly language

typical applications of ADSP

The ADSP serves as a high-speed pixel coprocessor for the RISC-like MP (master processor). Typical tasks performed by the ADSPs are:

- Pixel-intensive processing
  - Motion estimation
  - Convolution
  - PixBLTs
  - Warp
  - Histogram
  - Mean square error
- Domain transforms
  - DCT
  - FFT
  - Hough
- Core graphics functions
  - Line
  - Circle
  - Shaded fills
  - Fonts
- Image analysis
  - Segmentation
  - Feature extraction
- Bit-stream coding/decoding
  - Data merging
  - Table lookups

ADVANCE INFORMATION

transfer controller (TC) overview

The transfer controller (TC) is a combined memory controller and DMA (direct memory access) machine. It handles the movement of data and instructions within the 'C80 system as required by the master processor, parallel processors, video controller, and external devices.

The transfer controller performs the following data-movement and memory-control functions:

- MP and ADSP instruction-cache fills
- MP data-cache fills and dirty-block write-back
- MP and ADSP packet transfers (PTs)
- Externally initiated packet transfers (XPTs)
- VC packet transfers (VCPTs)
- MP and ADSP direct external accesses (DEAs)
- VC shift-register-transfer (SRTs)
- DRAM refresh
- External bus requests

Operations are performed on the cache subblock as requested by the processors' internal cache controllers. DEA operations transfer off-chip data directly to or from processor registers. Packet transfers are the main data transfer operations and provide an extremely flexible method for moving multidimensional blocks of data (packets) between on-chip and/or off-chip memory.

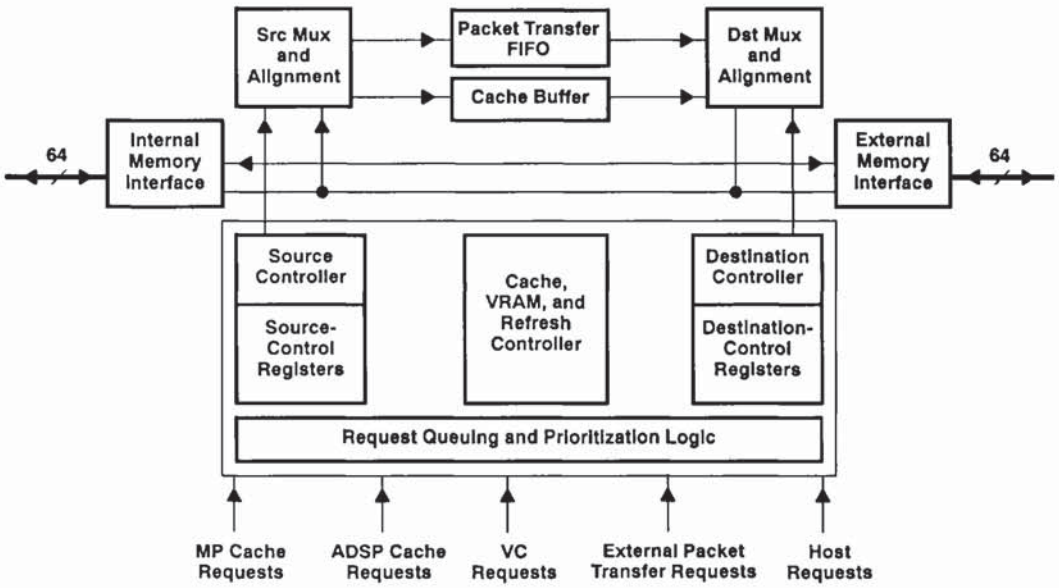


Figure 5. Transfer Controller Architecture

transfer controller (TC) overview (continued)

Key features of the TC include:

- Crossbar interface
  - 64-bit data path
  - Single-cycle access
- External memory interface
  - 4G-byte address range
  - Dynamically configurable memory cycles
    Bus size of 8, 16, 32, or 64 bits
    Selectable memory page size
    Selectable row/column address multiplexing
    Selectable cycle timing
  - Big or little endian operation
- Cache, VRAM, and refresh controller
  - Programmable refresh rate
  - VRAM block-write support
- Independent source and destination addressing
  - Autonomous address generation based on packet transfer parameters
  - Data can be read and written at different rates
  - Numerous data merging and spreading functions can be performed during transfers
- Intelligent request prioritization

overview of the video controller (VC)

The VC is the portion of the 'C80 responsible for the video interface. It provides simultaneous control over two independent frame systems (capture and display) and two frame memories – memory regions coordinated with the systems as either frame-grabber or frame-buffer image storage.

Key features of the VC include:

- Dual frame timers
  - Independent or locked operation
  - Programmable vertical and horizontal timing
  - Seperate or composite sync and blanking control
  - Synchronization to external timing signals
  - Interlaced or noninterlaced frame control
  - Virtually limitless screen resolutions
- SRT controller
  - Generates VRAM serial-register transfer (SRT) requests to the transfer controller
  - Track VRAM tap point and schedules midline reloads
  - Supports display or capture frame buffer transfers
- Timing and control registers programmable via master processor
- Programmable line interrupt to the MP

overview of the video controller (VC) (continued)

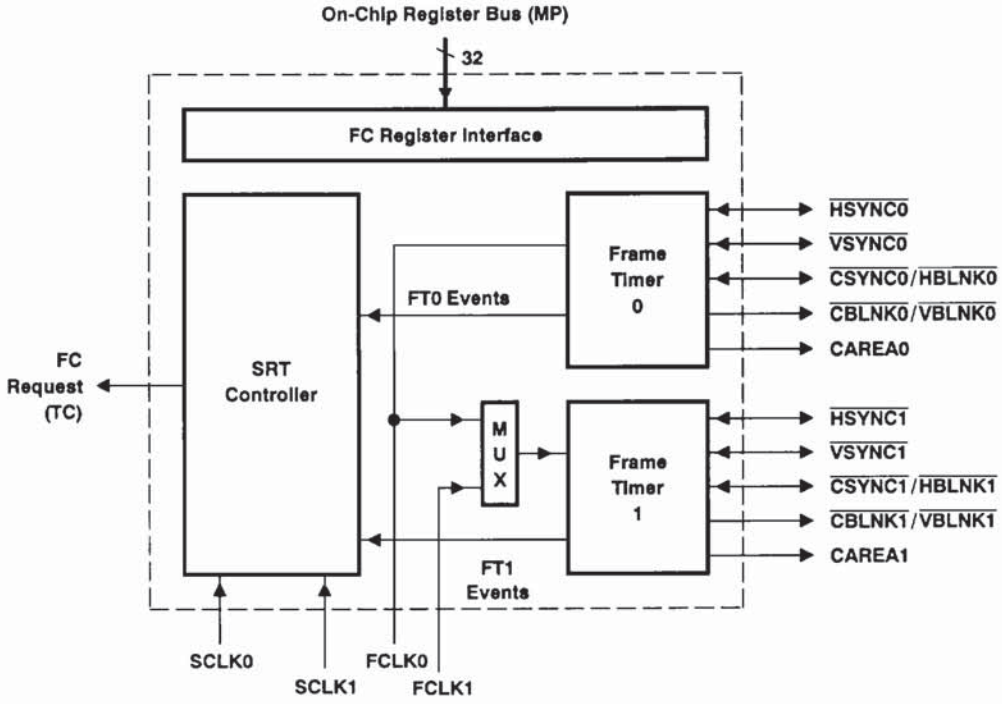


Figure 6. Video Controller Architecture

external memory timing examples

The following sections contain descriptions of various types of external memory cycles and illustrations of the timing for those cycles. For further information about these types of cycles and for explanations of terms used here and for labels used on the illustrations, refer to the *TMS320C80 (MVP) Online Reference*.

read cycle

Local-memory read cycles transfer data and instructions between memory and the 'C80. These cycles can occur as a result of a packet transfer, a cache request, or a DEA (direct external access) request to the transfer controller. The TC outputs 00000 on STATUS[5:0] at the beginning of the cycle to indicate that a read is occurring. During the cycle, $\overline{W}$ is held high. $\overline{TRG}$ is driven low after the fall of $\overline{RAS}$ to enable memory output drivers, and $\overline{DDIN}$ is low during the cycle so that data transceivers drive into the 'C80. The TC switches D[63:0] to the high-impedance state, allowing it to be driven from the memory, and latches input data during the appropriate column state. The TC always reads a 64-bit doubleword and then extracts and aligns the appropriate data bytes. Invalid bytes for bus sizes of less than 64 bits are discarded. Read cycles are generally classified into one of the following categories:

- Pipelined 1-cycle/column read (see Figure 8)
- Nonpipelined 1-cycle/column read (see Figure 9)
- 2-cycles/column read (see Figure 10)
- 3-cycles/column read (see Figure 11)

read cycle (continued)

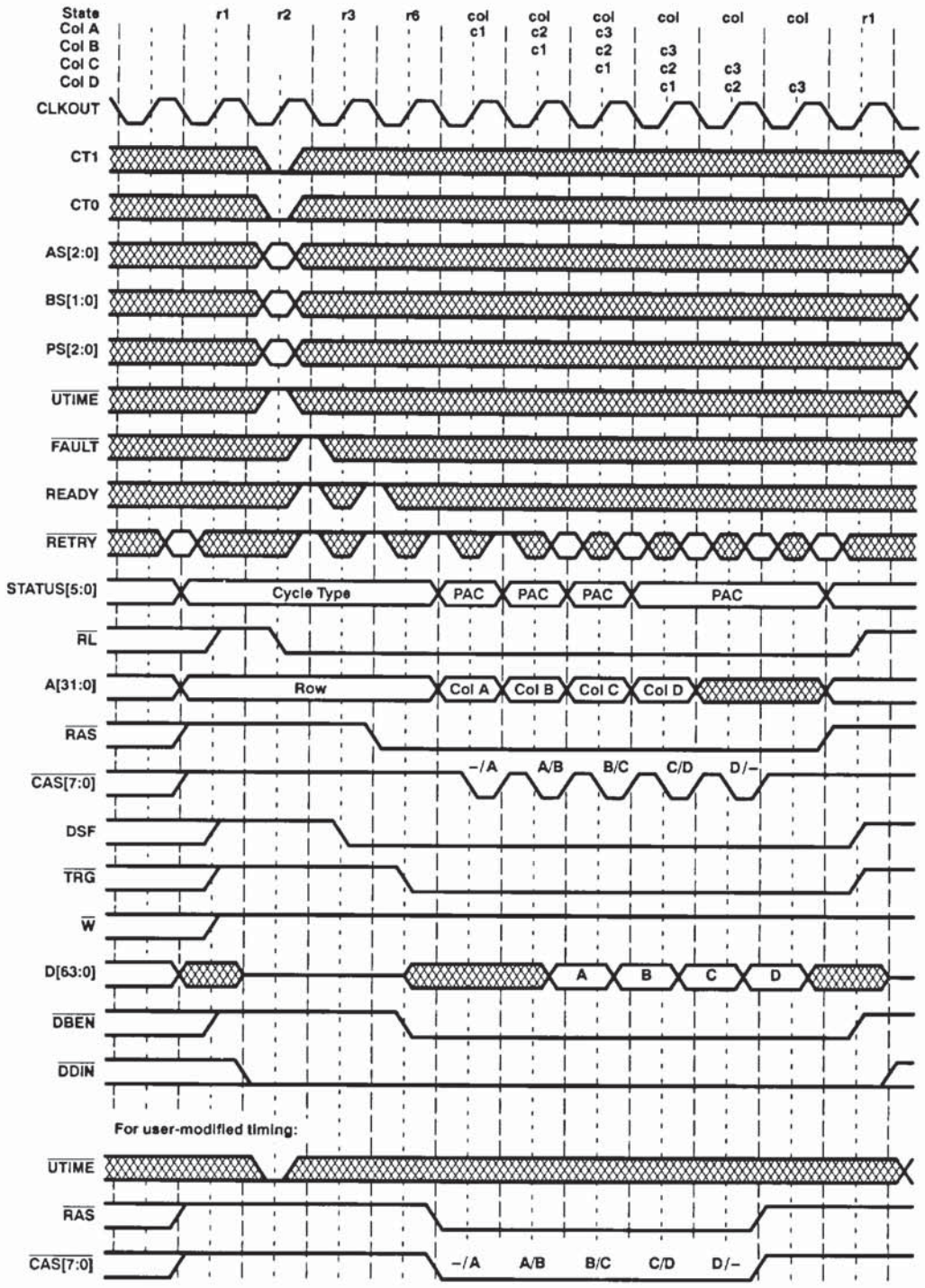


Figure 7. Pipelined 1-Cycle/Column Read-Cycle Timing

ADVANCE INFORMATION

read cycle (continued)

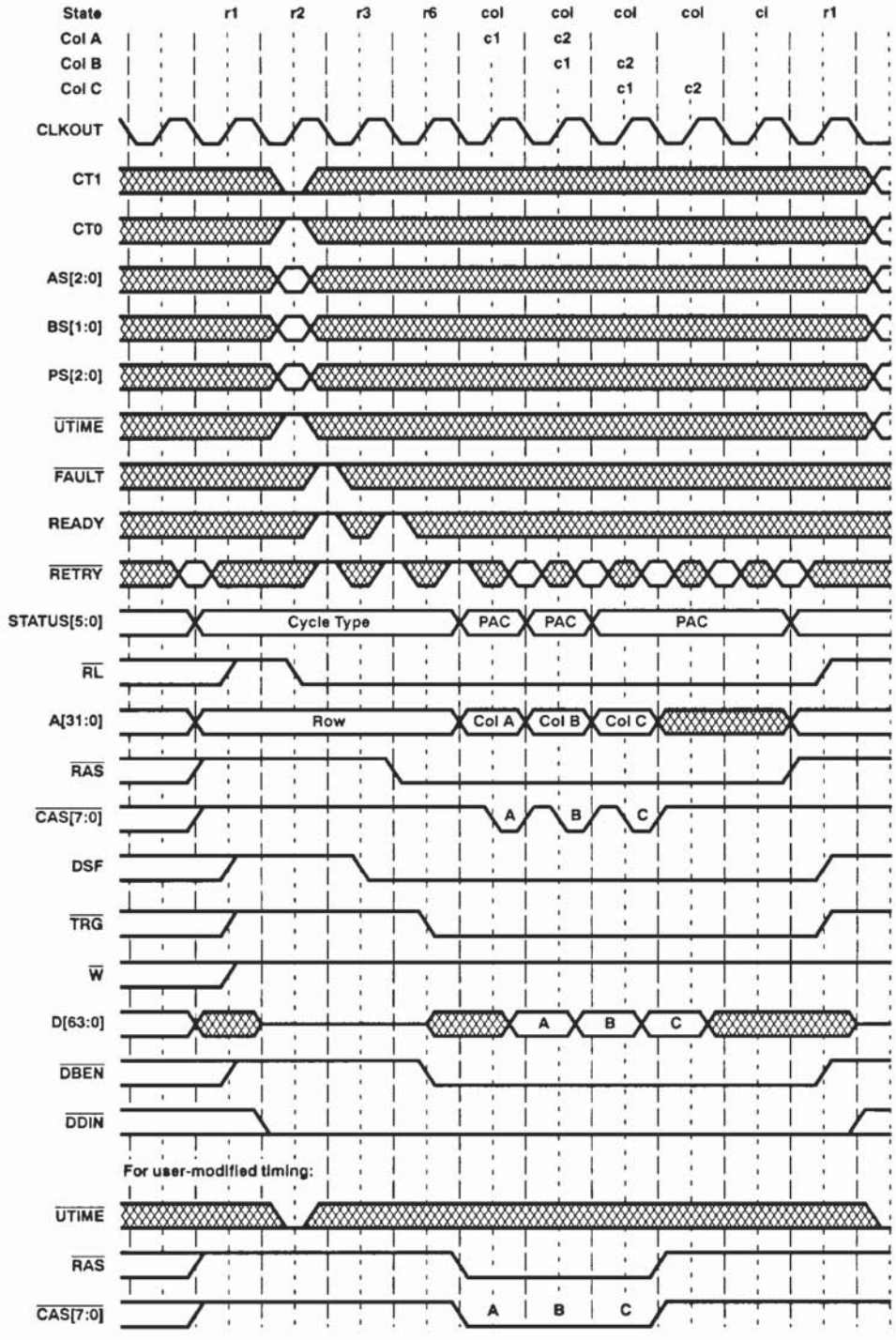


Figure 8. Nonpipelined 1-Cycle/Column Read-Cycle Timing

read cycle (continued)

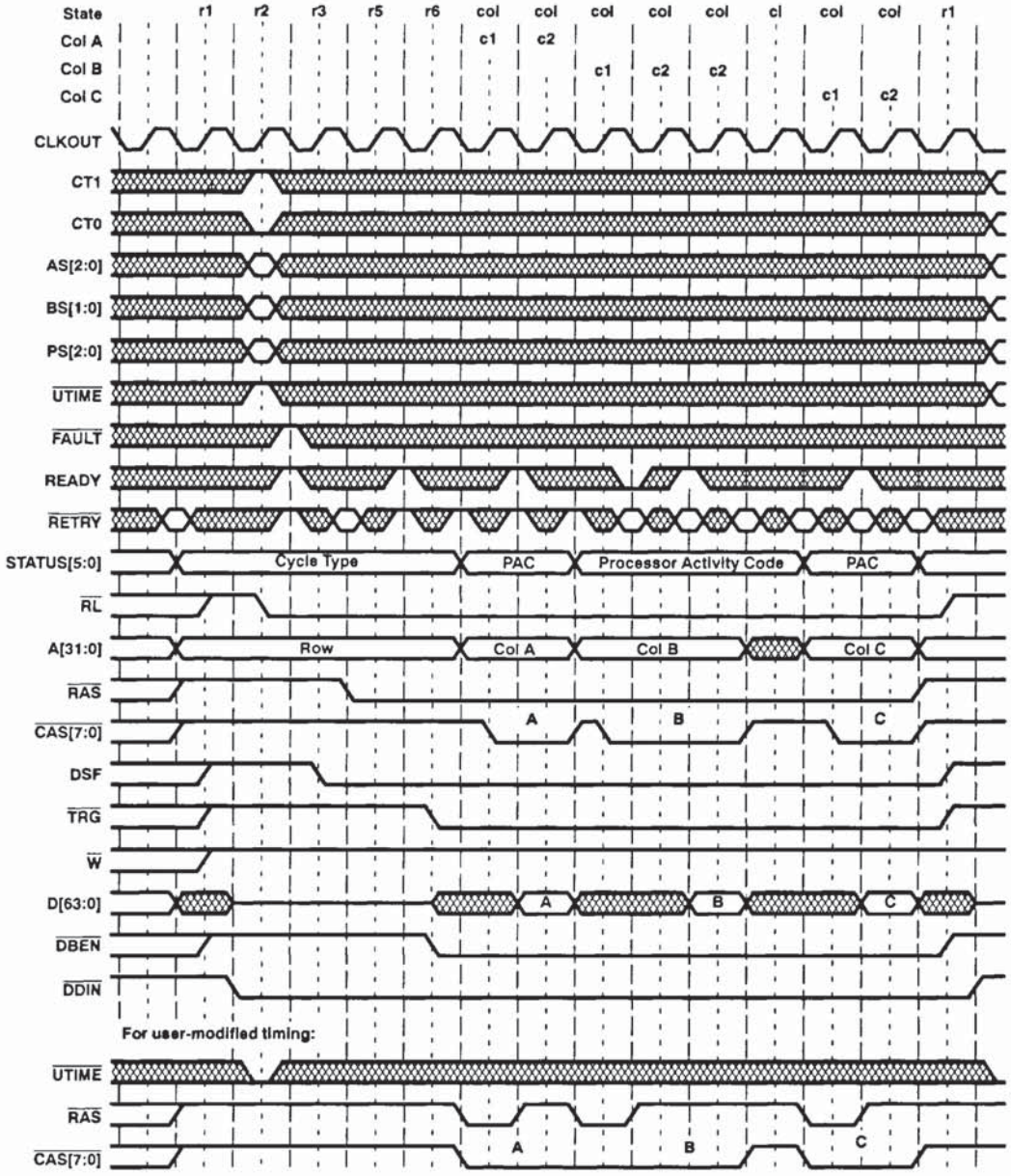


Figure 9. 2-Cycles/Column Read-Cycle Timing

ADVANCE INFORMATION

ADVANCE INFORMATION

read cycle (continued)

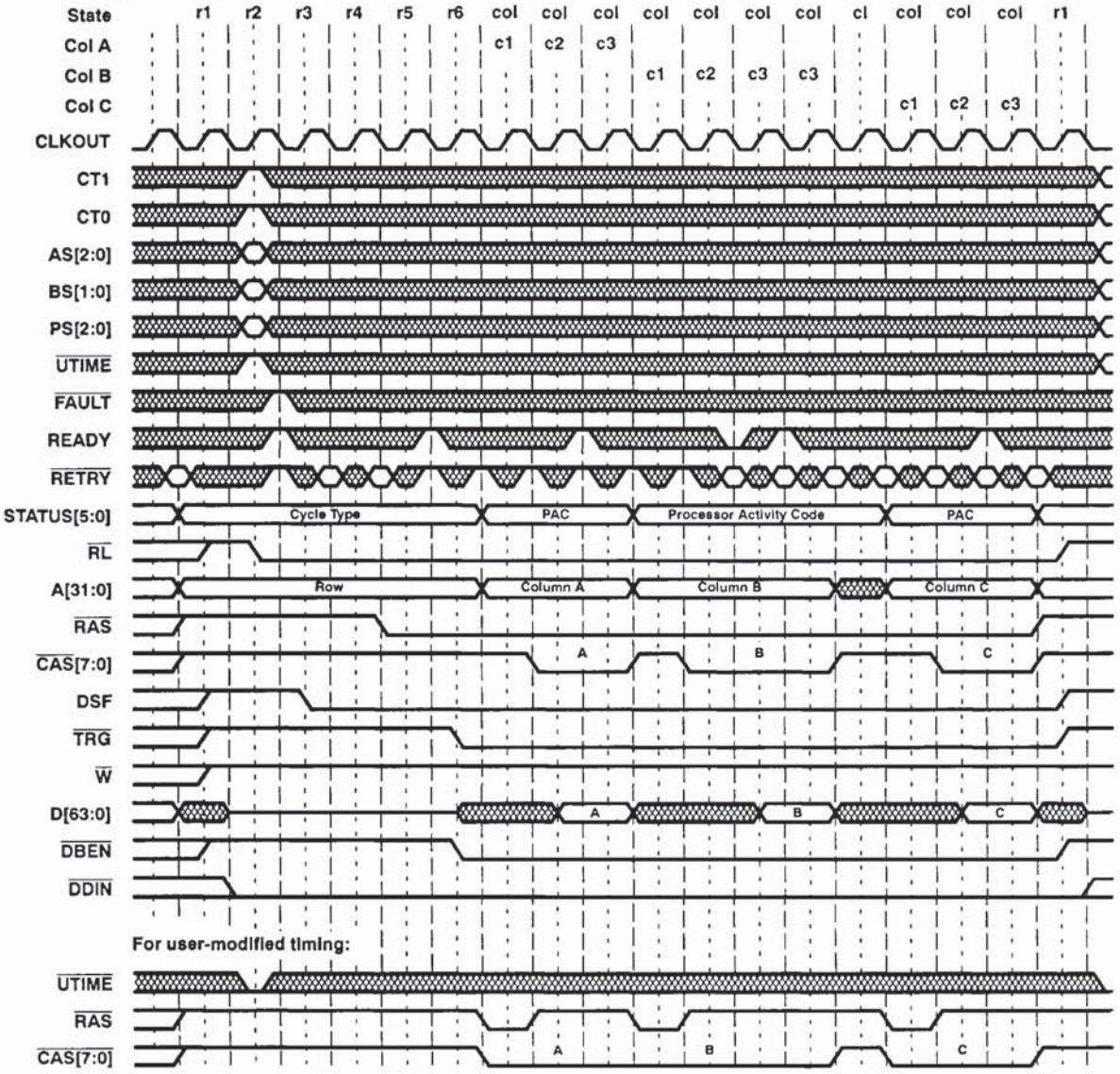


Figure 10. 3-Cycles/Column Read-Cycle Timing

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

write cycle

Write cycles transfer data and instructions from the 'C80 to memory. These cycles can occur as a result of a packet transfer, a cache request, or a DEA request to the transfer controller. The TC outputs 000001 on STATUS[5:0] at the beginning of a cycle to indicate that a write is occurring. During the cycle, $\overline{TRG}$ is held high, $\overline{W}$ is driven low after the fall of $\overline{RAS}$ to enable memory early-write cycles, and $\overline{DDIN}$ is high during the cycle so that data transceivers drive toward memory. The TC drives data out on D[63:0] and indicates valid bytes by activating the valid $\overline{CAS}$ strobes. Write cycles are generally classified into one of the following categories:

- Pipelined 1-cycle/column write (see Figure 11)
- Nonpipelined 1-cycle/column write (see Figure 12)
- 2-cycles/column write (see Figure 13)
- 3-cycles/column write (see Figure 14)

write cycle (continued)

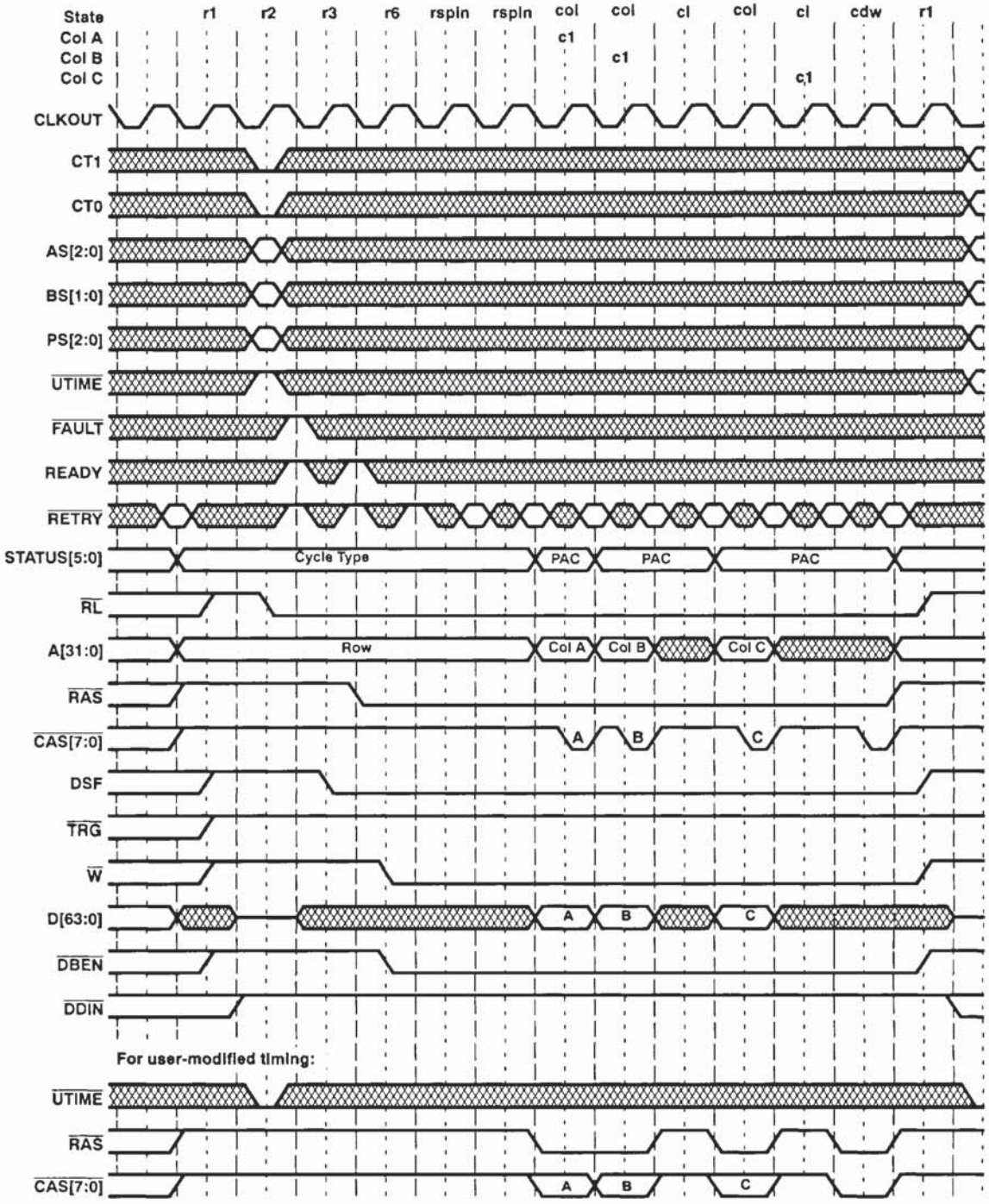


Figure 11. Pipelined 1-Cycle/Column Write-Cycle Timing

write cycle (continued)

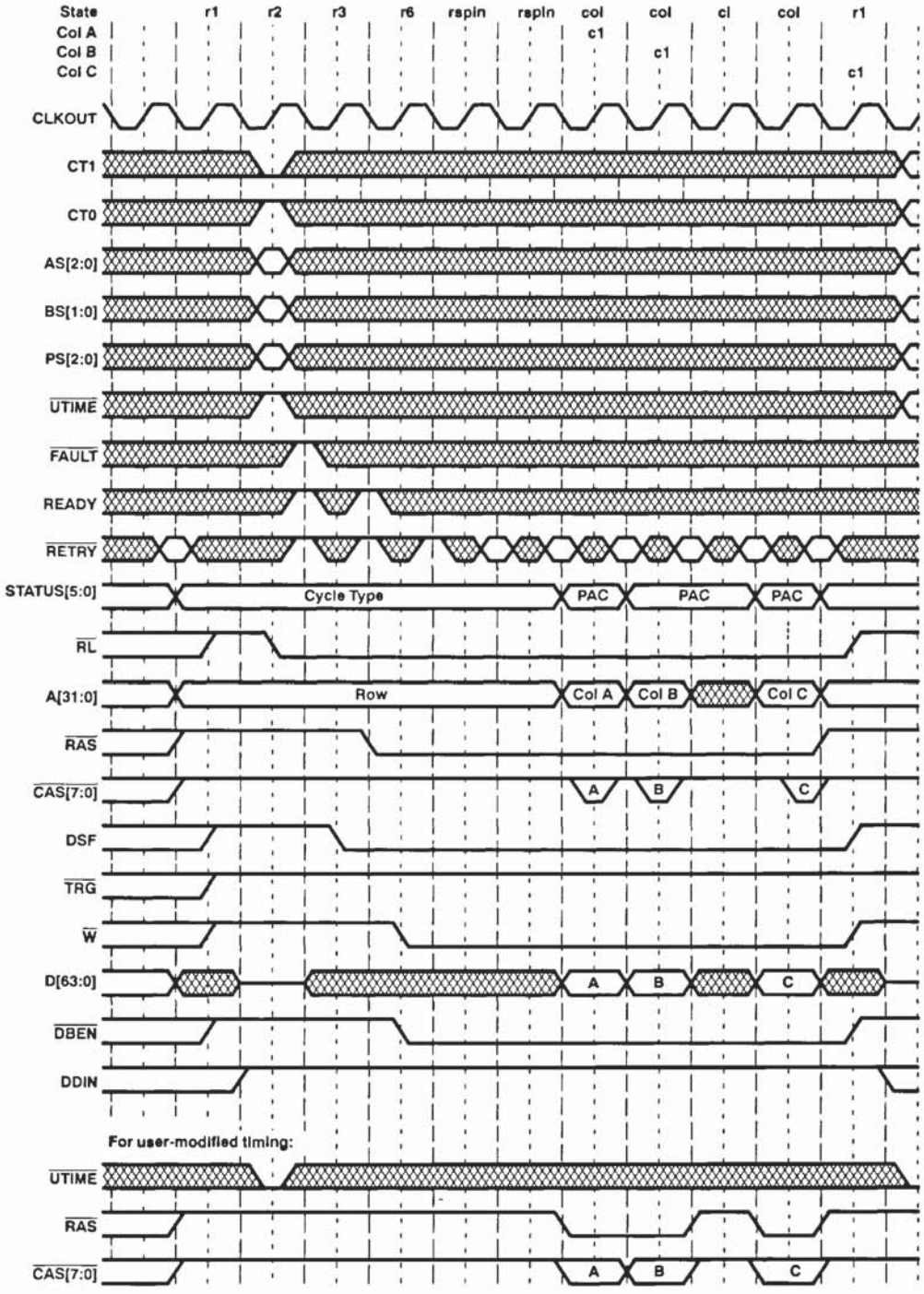


Figure 12. Nonpipelined 1-Cycle/Column Write-Cycle Timing

ADVANCE INFORMATION

write cycle (continued)

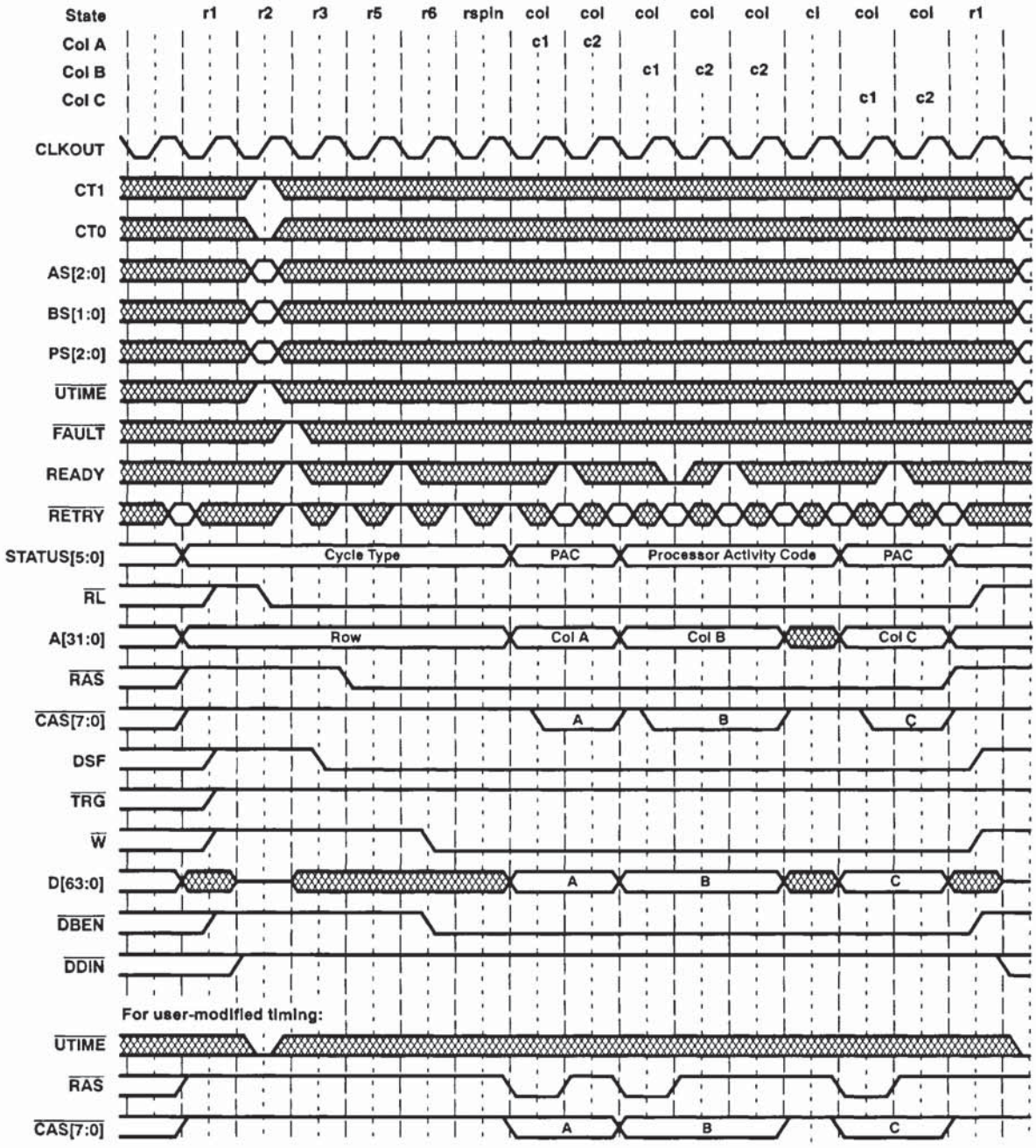


Figure 13. 2-Cycles/Column Write-Cycle Timing

ADVANCE INFORMATION

write cycle (continued)

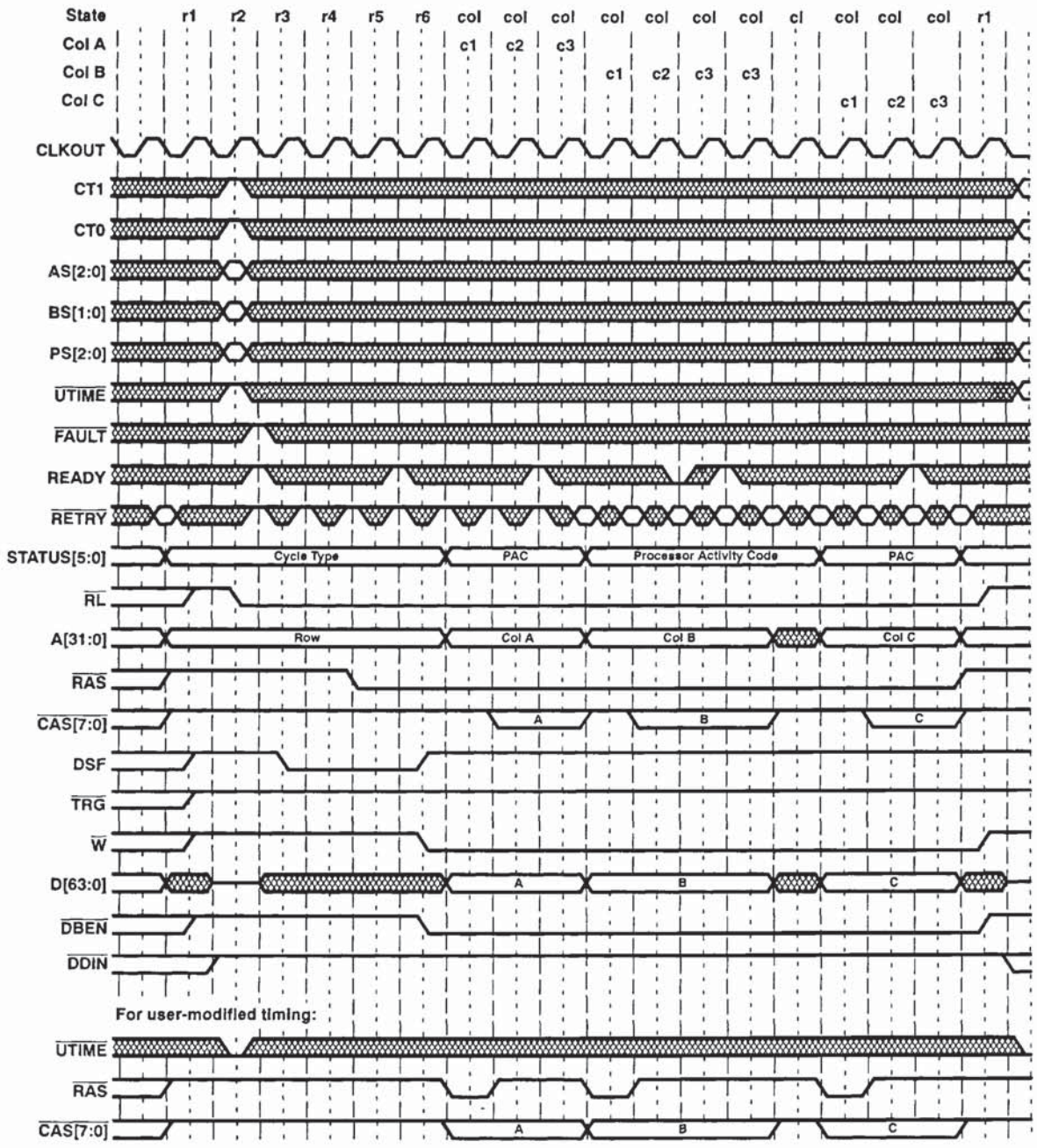


Figure 14. 3-Cycles/Column Write-Cycle Timing

ADVANCE INFORMATION

load-color-register cycle

Load-color-register (LCR) cycles are used to write a value into the VRAM's color register for later use during a block-write cycle. LCR cycles are supported only on 64-bit data buses. A status code of 0001101 indicates an LCR cycle. Because this cycle writes data into the VRAM, its timing closely resembles a standard write cycle. The difference is that DSF is high at both the fall of $\overline{RAS}$ and the fall of $\overline{CAS}$. Also, because the VRAM color register is a single location, page-mode accesses do not occur.

The row address output by the TC is important for memory bank decode information only. In most cases, all VRAM banks must be selected during LCR cycles because another LCR is not performed when a memory page change occurs. The row addressed at the VRAM is refreshed but otherwise is a don't care. Similarly, the column address provided to the VRAM for LCR cycles is irrelevant. All $\overline{CAS}[7:0]$ outputs are active during LCR cycles.

Although the $\overline{RETRY}$ input is sampled and must be valid high or low during column-time states, asserting a column-time retry has no effect because only one column access is performed.

The BS[1:0] inputs during LCR cycles determine the block-write type supported by the addressed memory. If simulated block-write mode is selected, the cycle becomes a standard write rather than an LCR cycle. The following load-color-register cycles are illustrated:

- Pipelined 1-cycle/column LCR cycle (see Figure 15)
- Nonpipelined 1-cycle/column LCR cycle (see Figure 16)
- 2-cycles/column LCR cycle (see Figure 17)
- 3-cycles/column LCR cycle (see Figure 18)

load-color-register cycle (continued)

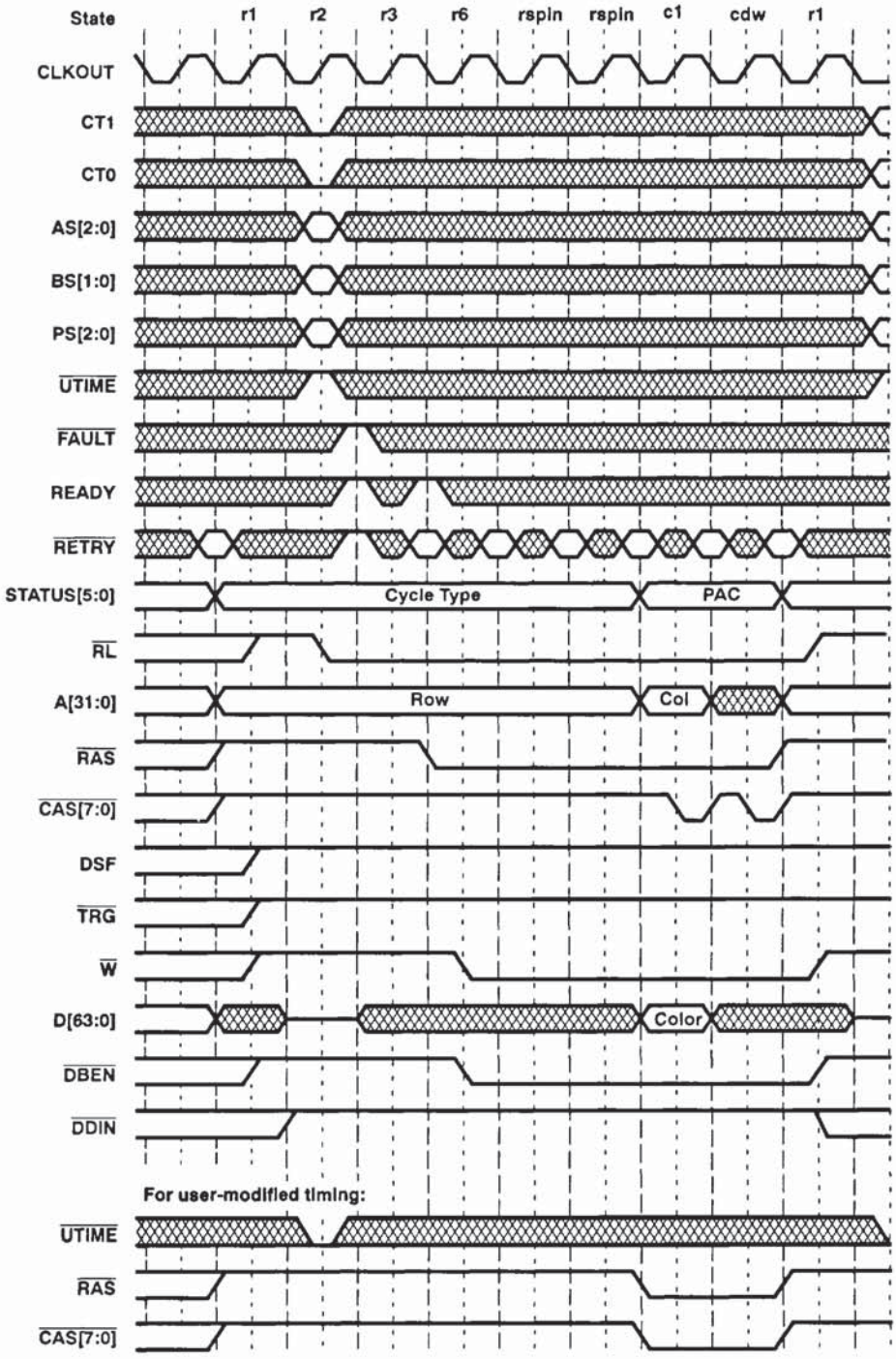


Figure 15. Pipelined 1-Cycle/Column Load-Color-Register-Cycle Timing

ADVANCE INFORMATION

load-color-register cycle (continued)

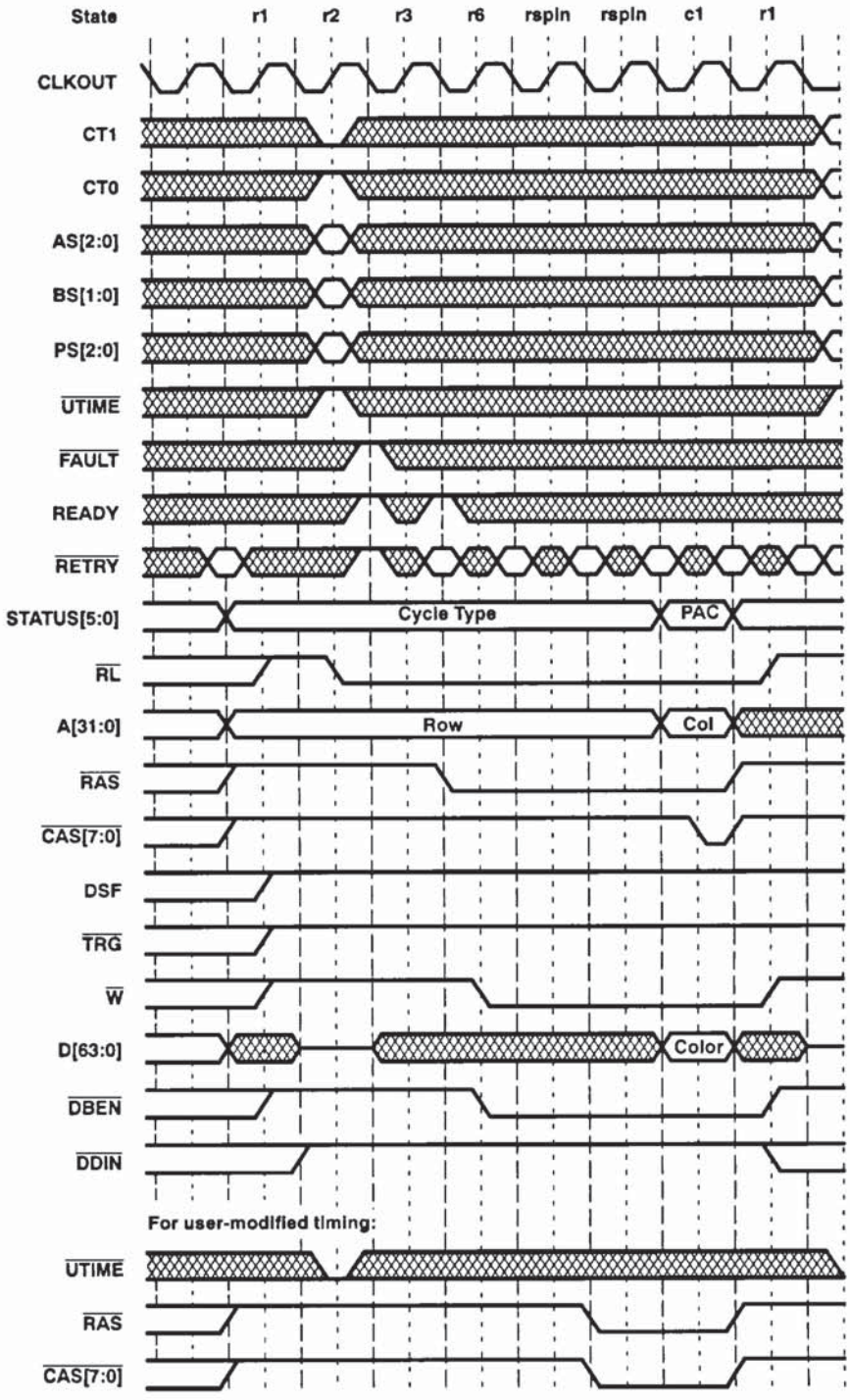


Figure 16. Nonpipelined 1-Cycle/Column Load-Color-Register-Cycle Timing

ADVANCE INFORMATION

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

load-color-register cycle (continued)

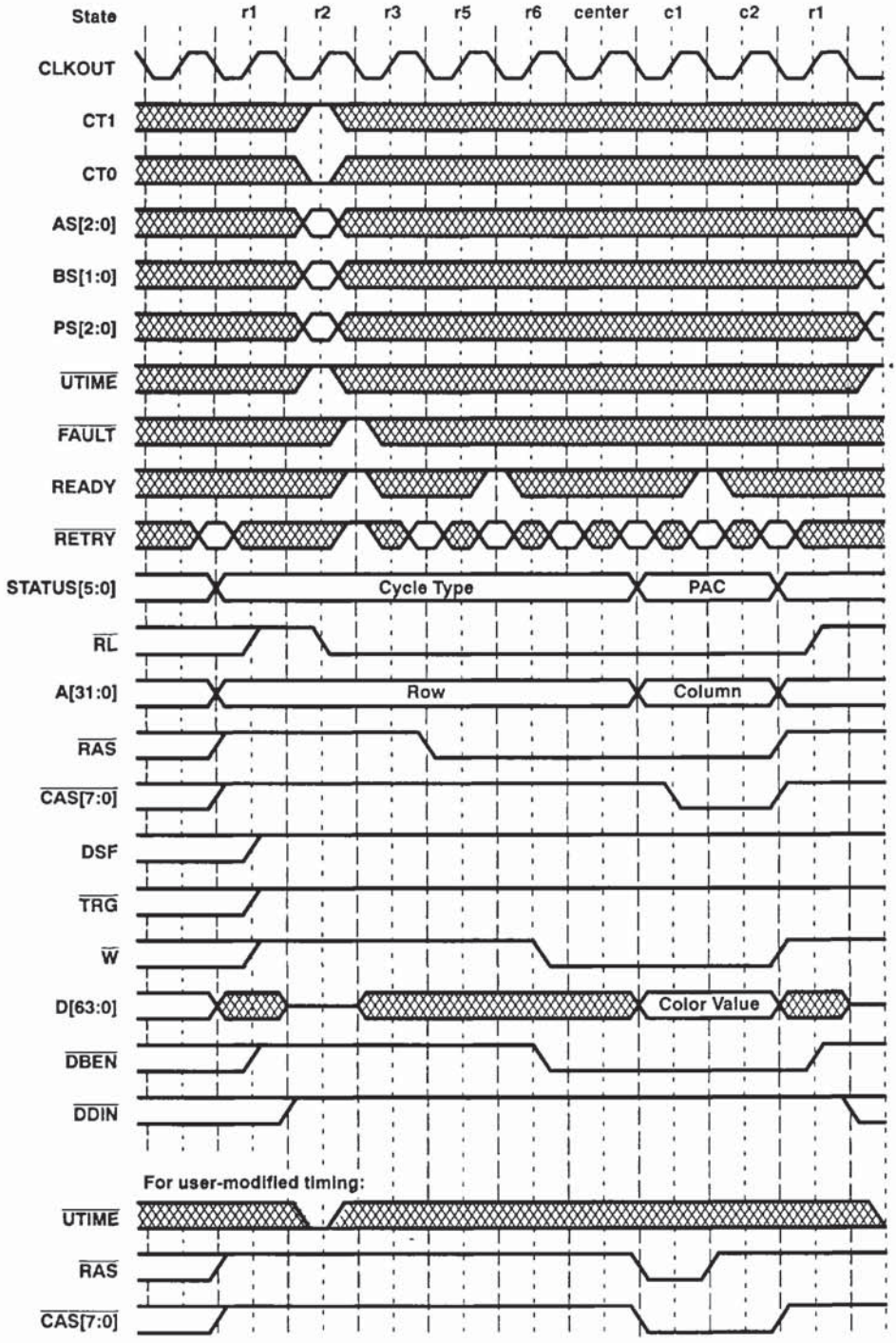


Figure 17. 2-Cycles/Column Load-Color-Register-Cycle Timing

ADVANCE INFORMATION

load-color-register cycle (continued)

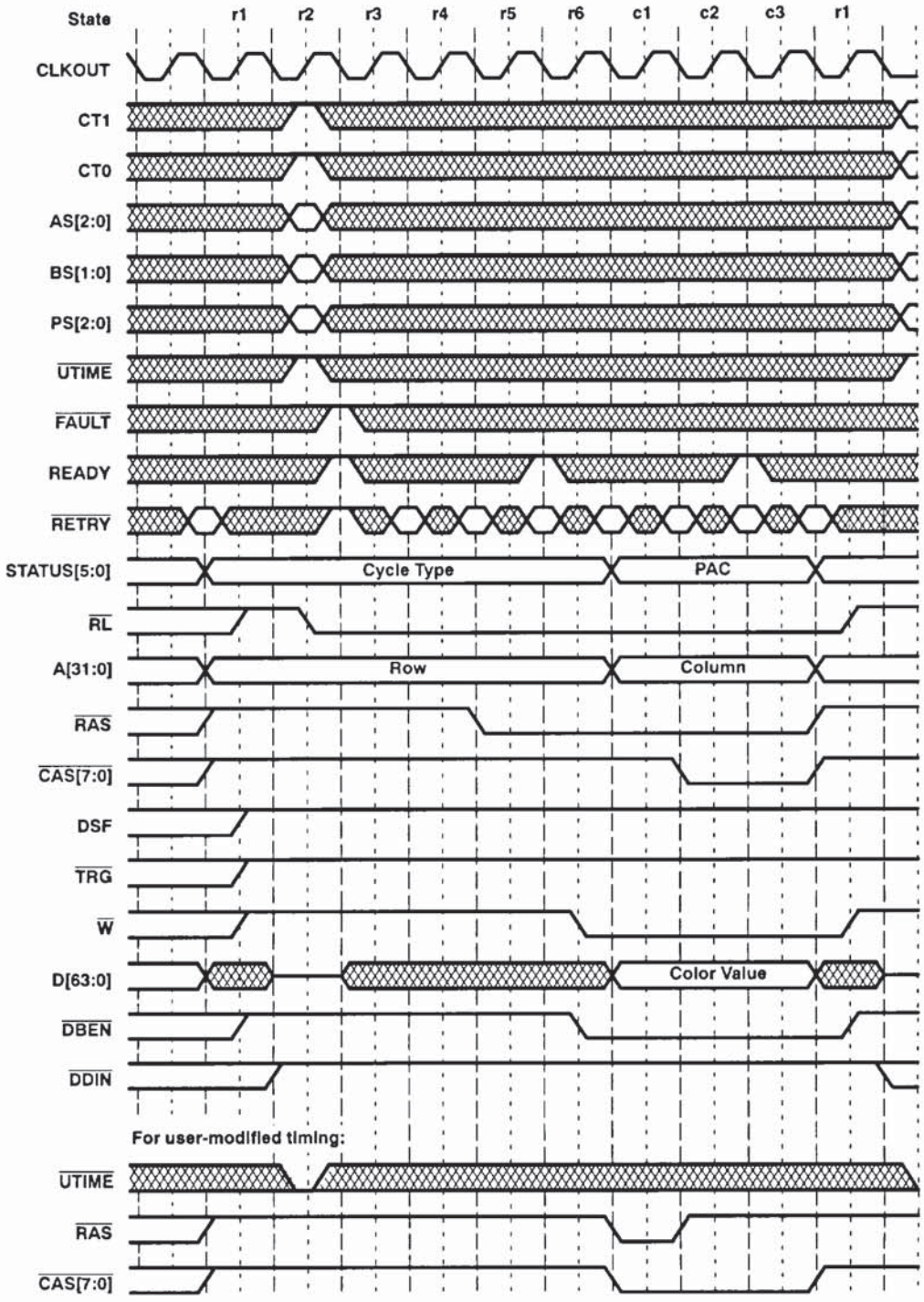


Figure 18. 3-Cycles/Column Load-Color-Register-Cycle Timing

block-write cycle

Block-write cycles cause the data stored in the VRAM color registers to be written to the memory locations enabled by the appropriate data bits output on the D[63:0] bus. This allows up to a total of 64 bytes (depending on the type of block write being used) to be written in a single column access. A status code of 001001 indicates a block-write cycle. This cycle is identical to a standard write cycle with the following exceptions:

- DSF is active (high) at the fall of $\overline{CAS}$, enabling the block-write function within the VRAMs.
- Only 64-bit bus sizes are supported during block write; therefore, BS[1:0] inputs are used to indicate the type of block write that is supported by the addressed VRAMs, rather than the bus size.
- The two or three LSBs (depending on the type of block write) of the column address are ignored by the VRAMs because these column locations are specified by the data inputs.
- The values output by the TC on D[63:0] represent the column locations to be written to using the color register value. Depending on the type of block write supported by the VRAM, all of the data bits may not be used by the VRAMs.
- Block writes always begin with a row access. Upon completion of a block write, the memory interface returns to state r1 to await the next access.

block-write cycle (continued)

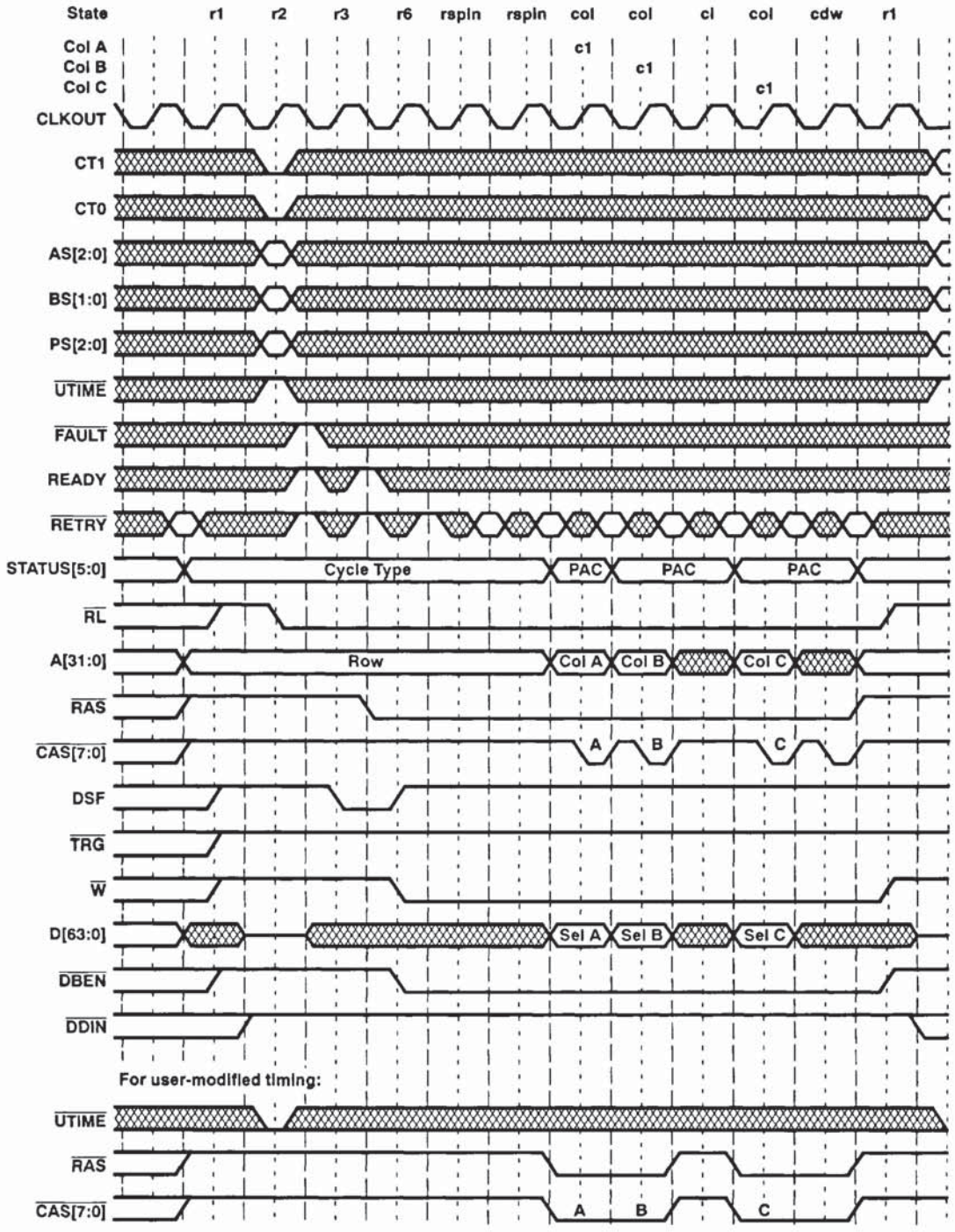


Figure 19. Pipelined 1-Cycle/Column Block-Write-Cycle Timing

block-write cycle (continued)

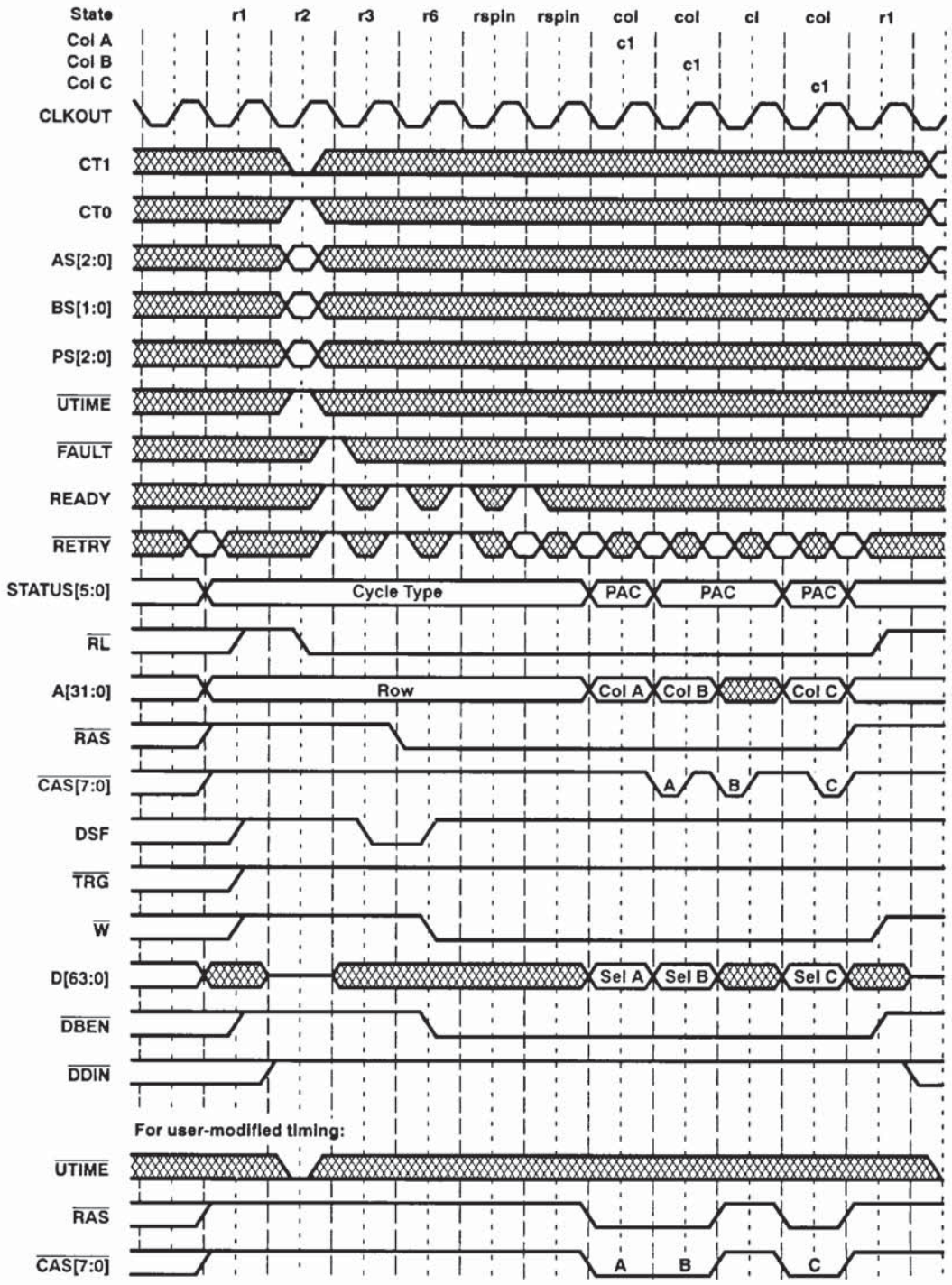


Figure 20. Nonpipelined 1-Cycle/Column Block-Write-Cycle Timing

ADVANCE INFORMATION

ADVANCE INFORMATION

block-write cycle (continued)

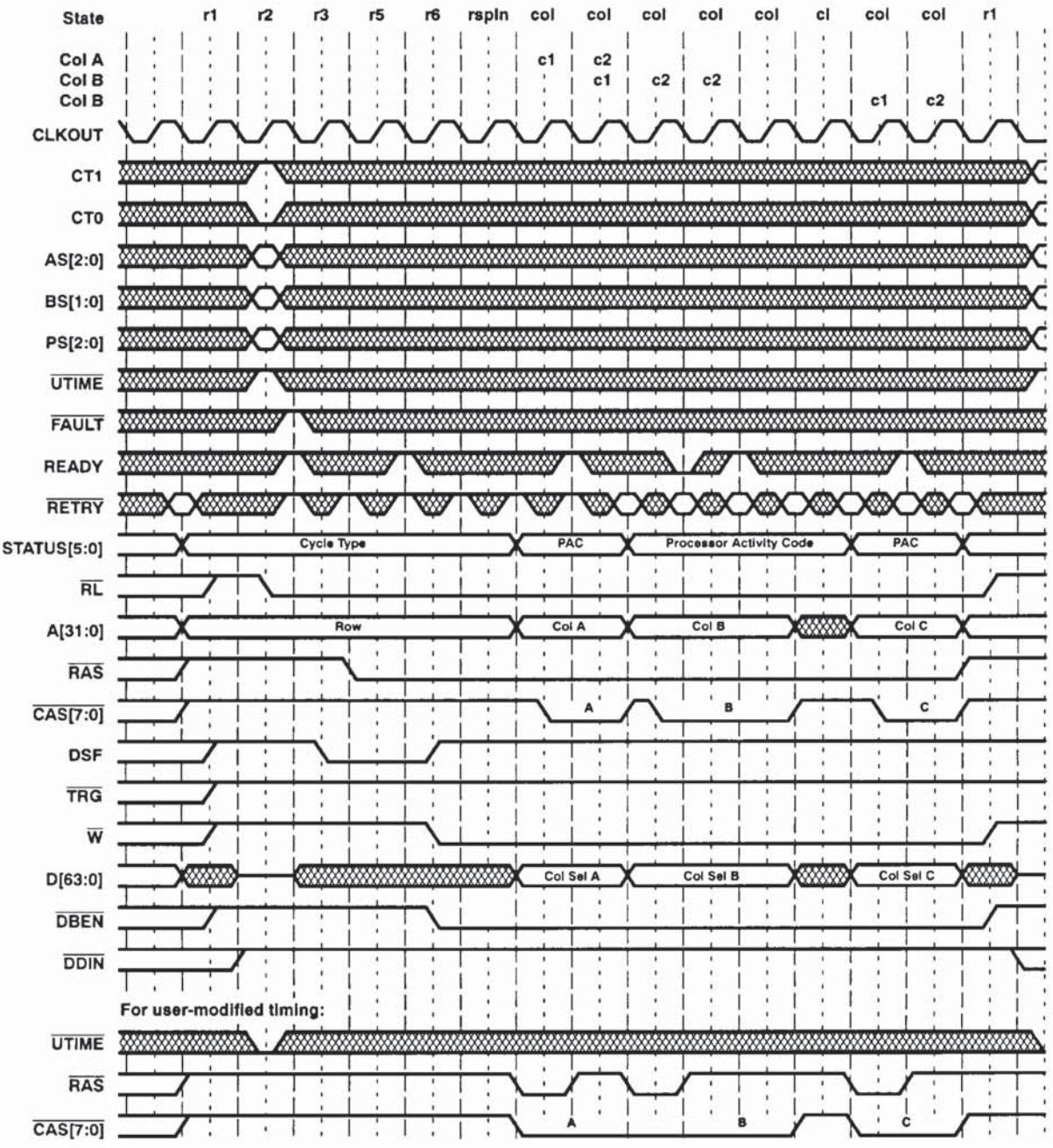


Figure 21. 2-Cycles/Column Block-Write-Cycle Timing

block-write cycle (continued)

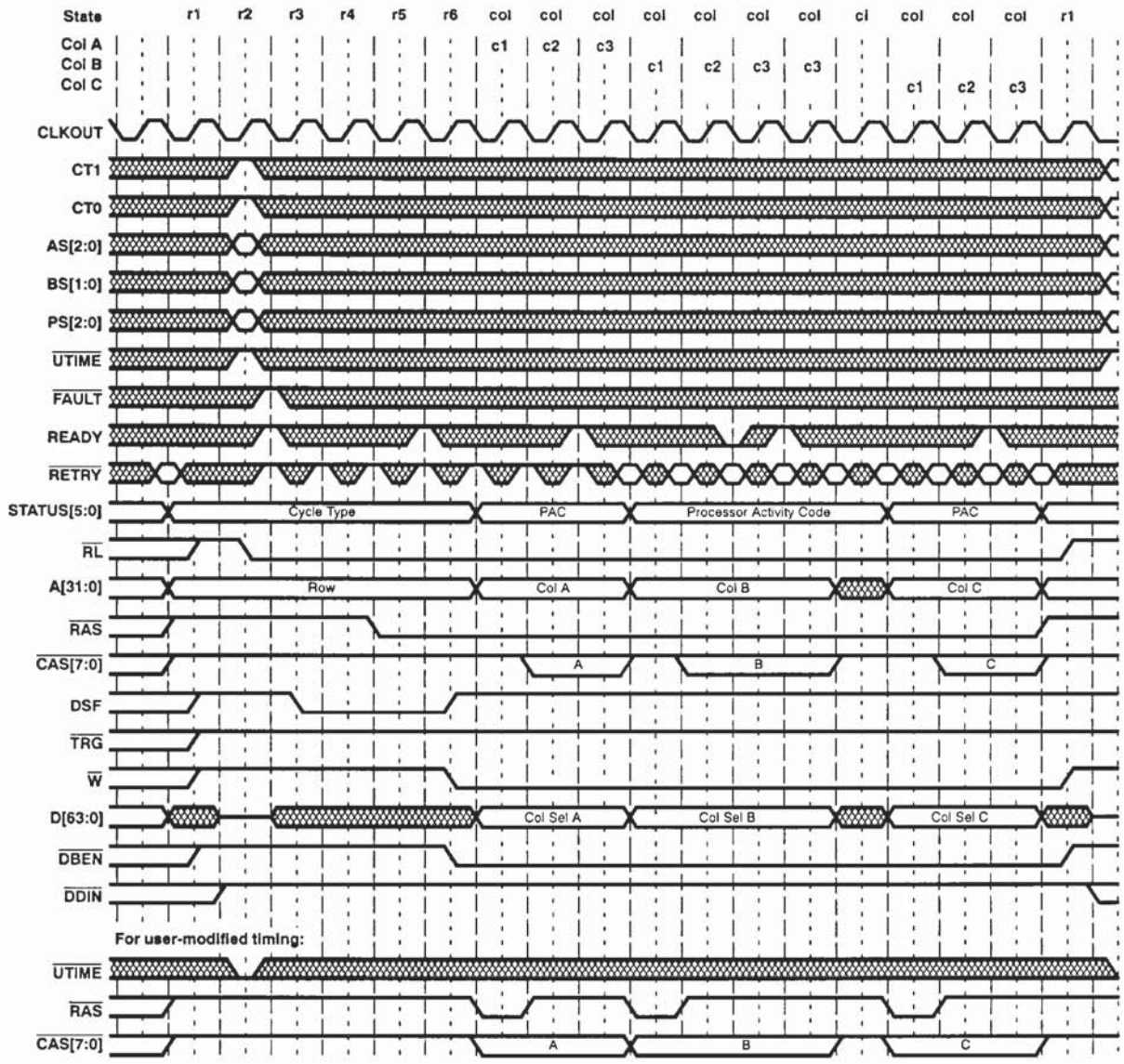


Figure 22. 3-Cycles/Column Block-Write-Cycle Timing

ADVANCE INFORMATION

transfer cycle

Read-transfer (memory-to-register) cycles transfer a row from the VRAM memory array into the VRAM shift register (SAM). This causes the entire SAM (both halves of the split SAM) to be loaded with the array data.

Split-register read-transfer (memory-to-split-register) cycles also transfer data from a row in the memory array to the SAM. However, these transfers cause only half of the SAM to be written. Split-register read transfers allow the inactive half of the SAM to be loaded with the new data while the other active half continues to shift data in or out.

Write-transfer (register-to-memory) cycles transfer data from the SAM into a row of the VRAM array. This transfer causes the entire SAM (both halves of the split SAM) to be written into the array.

Split-register write-transfer (split-register-to-memory) cycles also transfer data from the SAM to a row in the memory array. However, these transfers write only half of the SAM into the array. Split-register write transfers allow the inactive half of the SAM to be transferred into memory while the other (active) half continues to shift serial data in or out.

transfer cycle (continued)

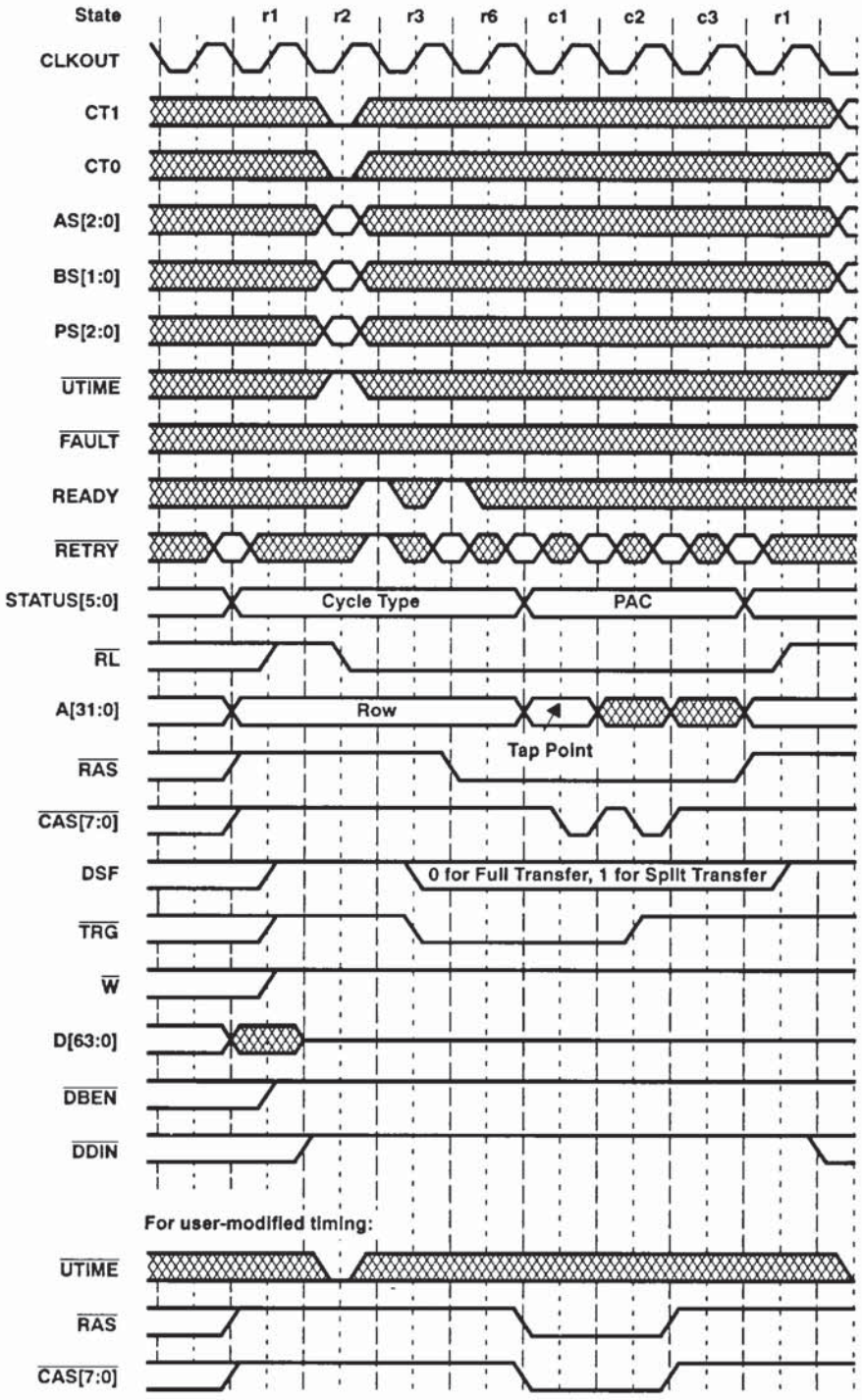


Figure 23. Pipelined 1-Cycle/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing

ADVANCE INFORMATION

transfer cycle (continued)

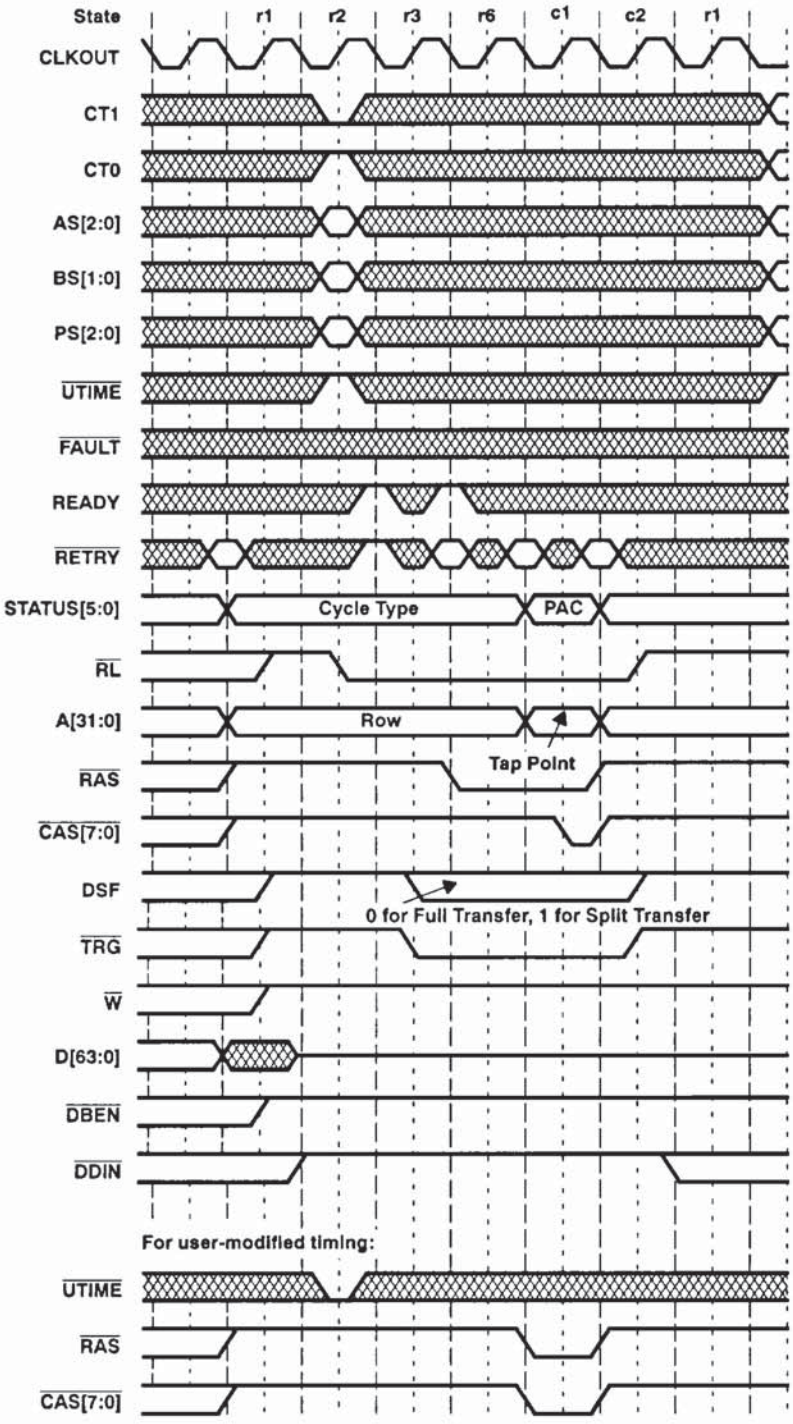


Figure 24. Nonpipelined 1-Cycle/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing

transfer cycle (continued)

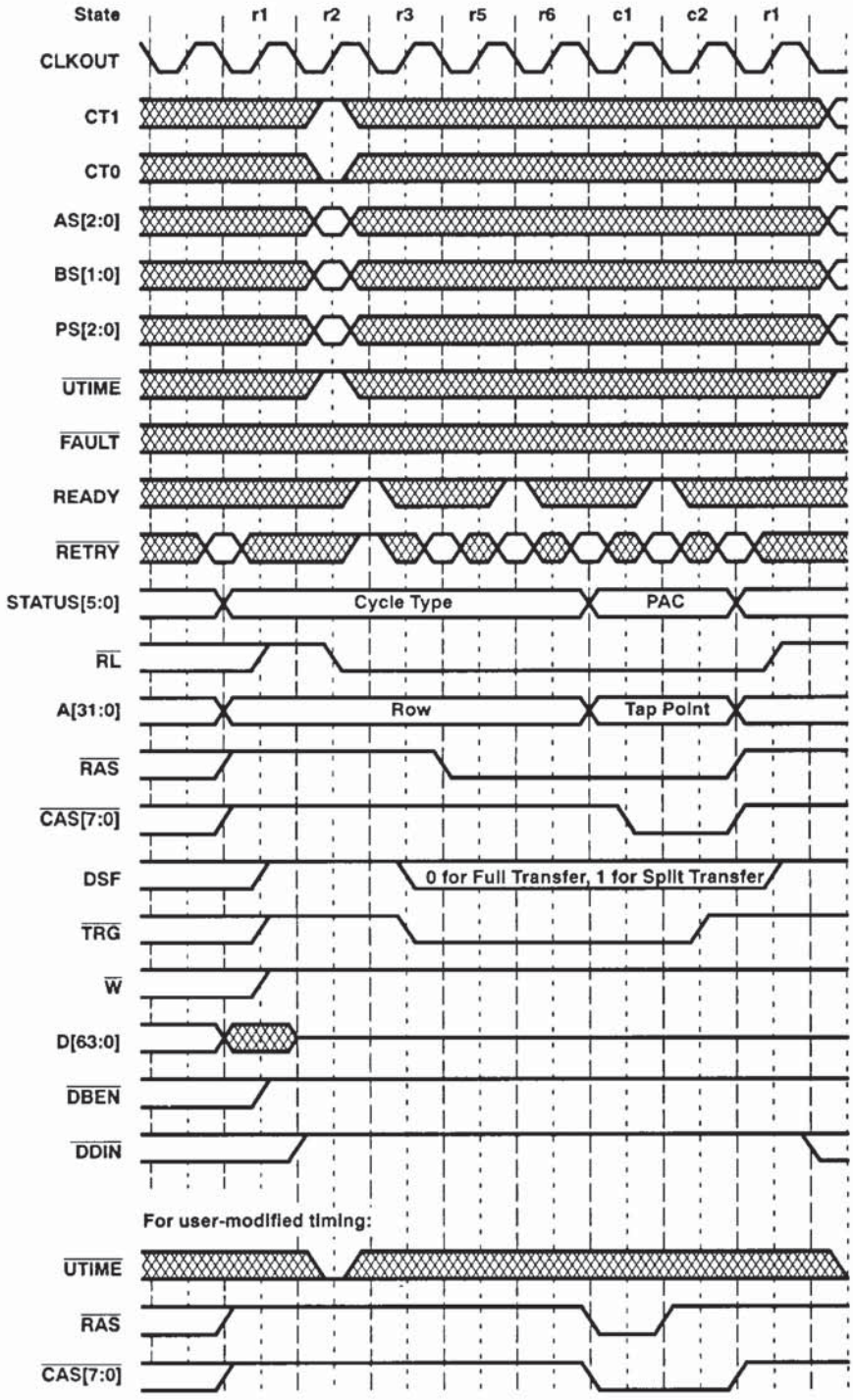


Figure 25. 2-Cycles/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing

ADVANCE INFORMATION

transfer cycle (continued)

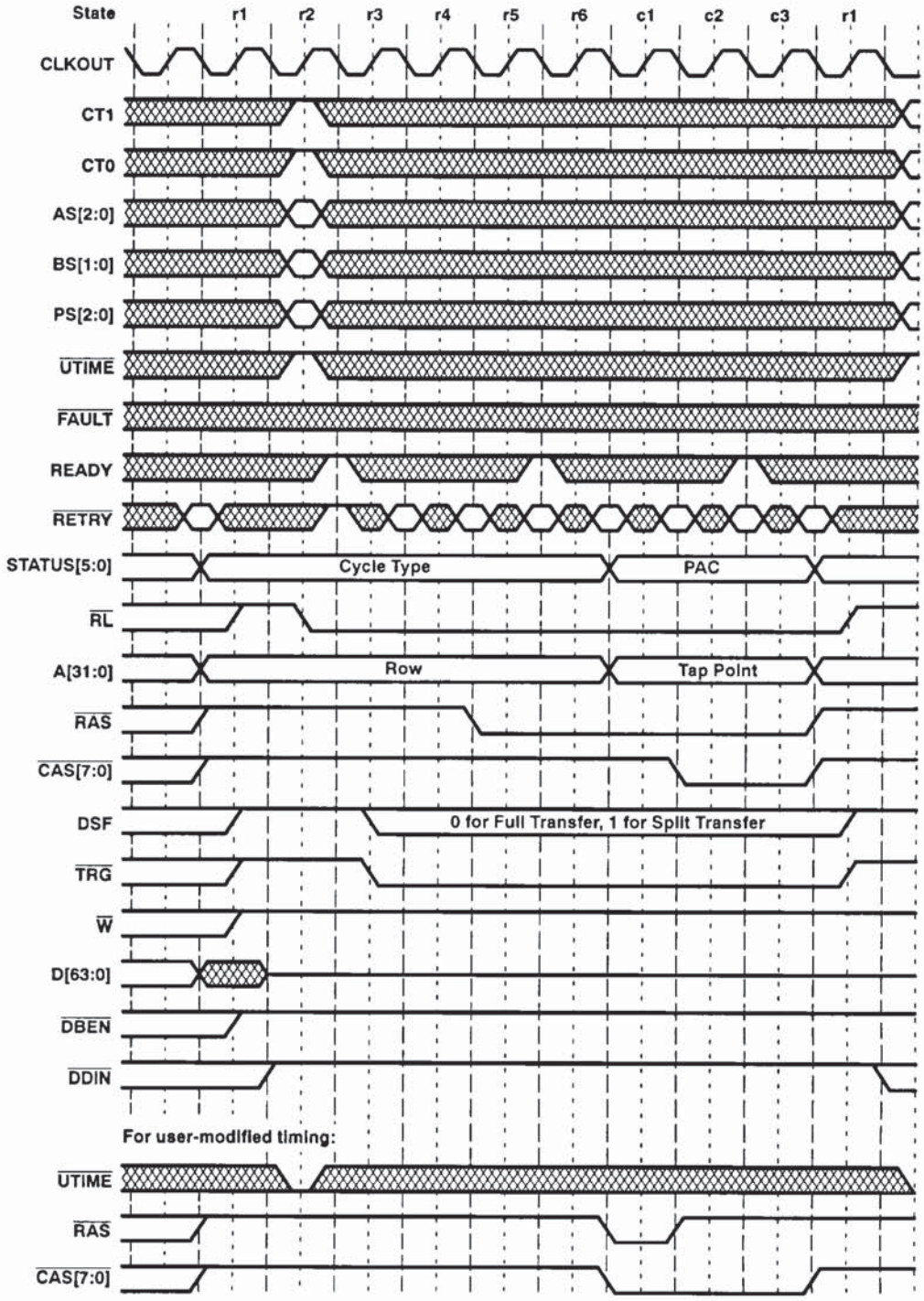


Figure 26. 3-Cycles/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

transfer cycle (continued)

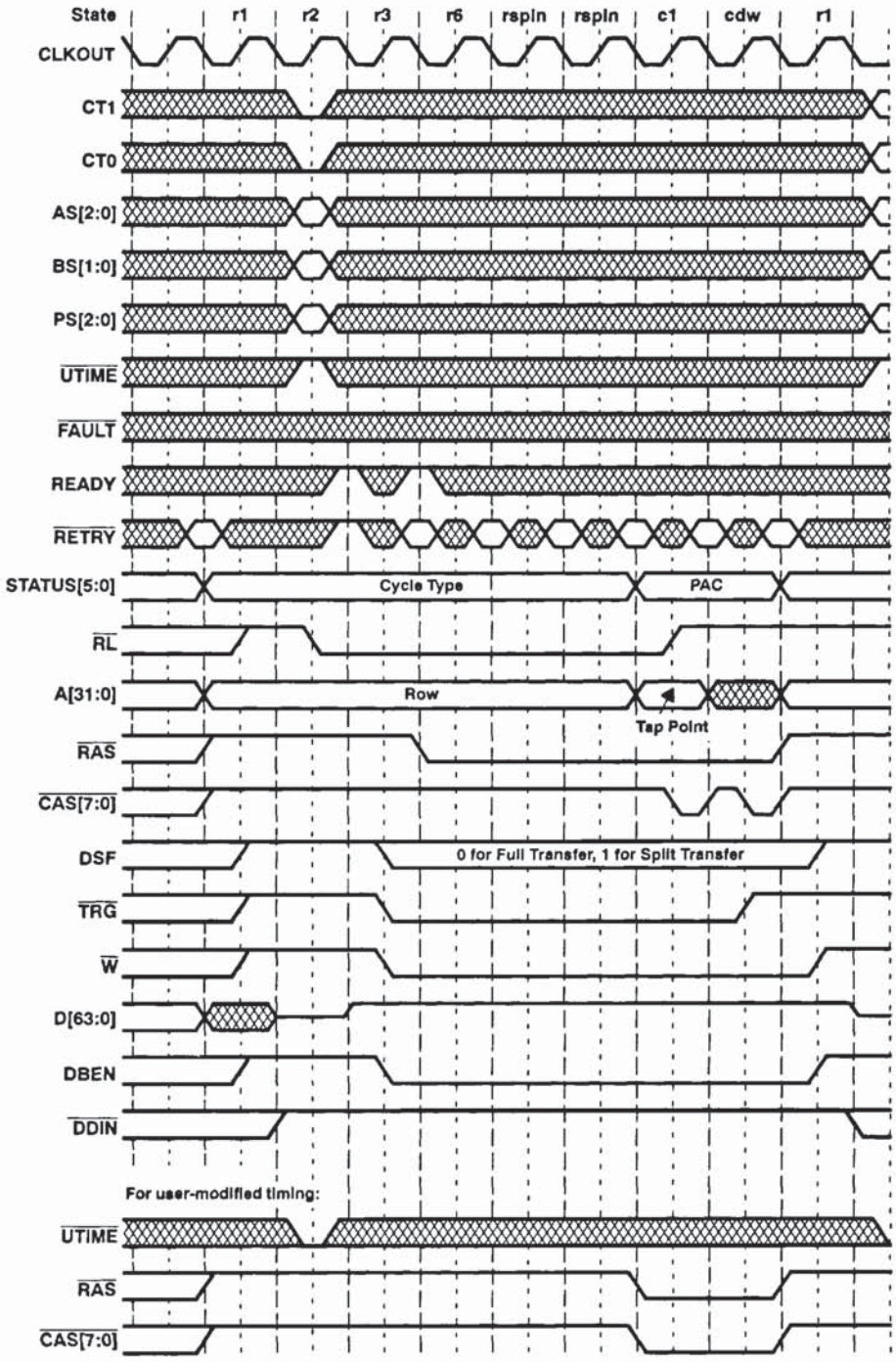


Figure 27. Pipelined 1-Cycle/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

ADVANCE INFORMATION

transfer cycle (continued)

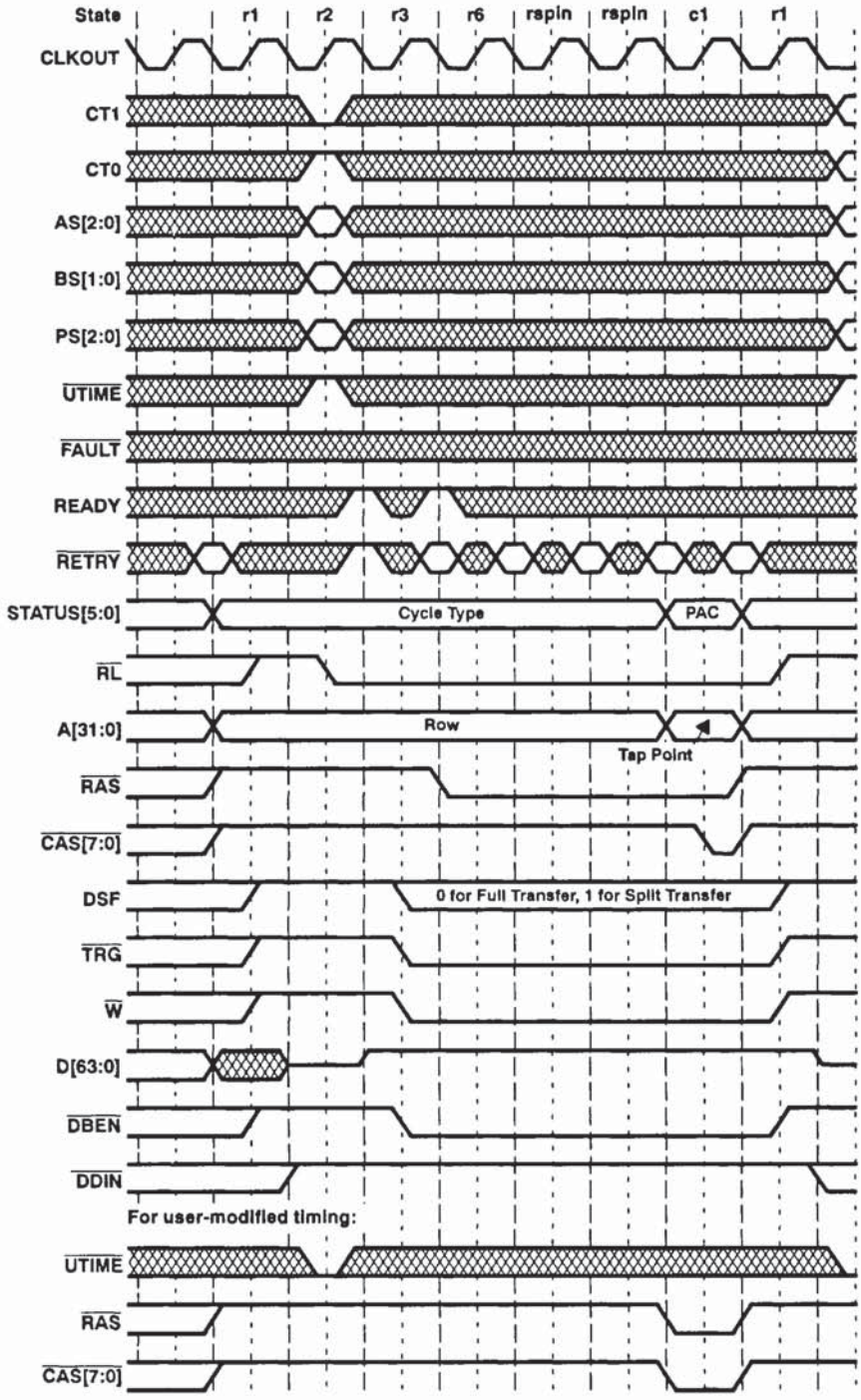


Figure 28. Nonpipelined 1-Cycle/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

transfer cycle (continued)

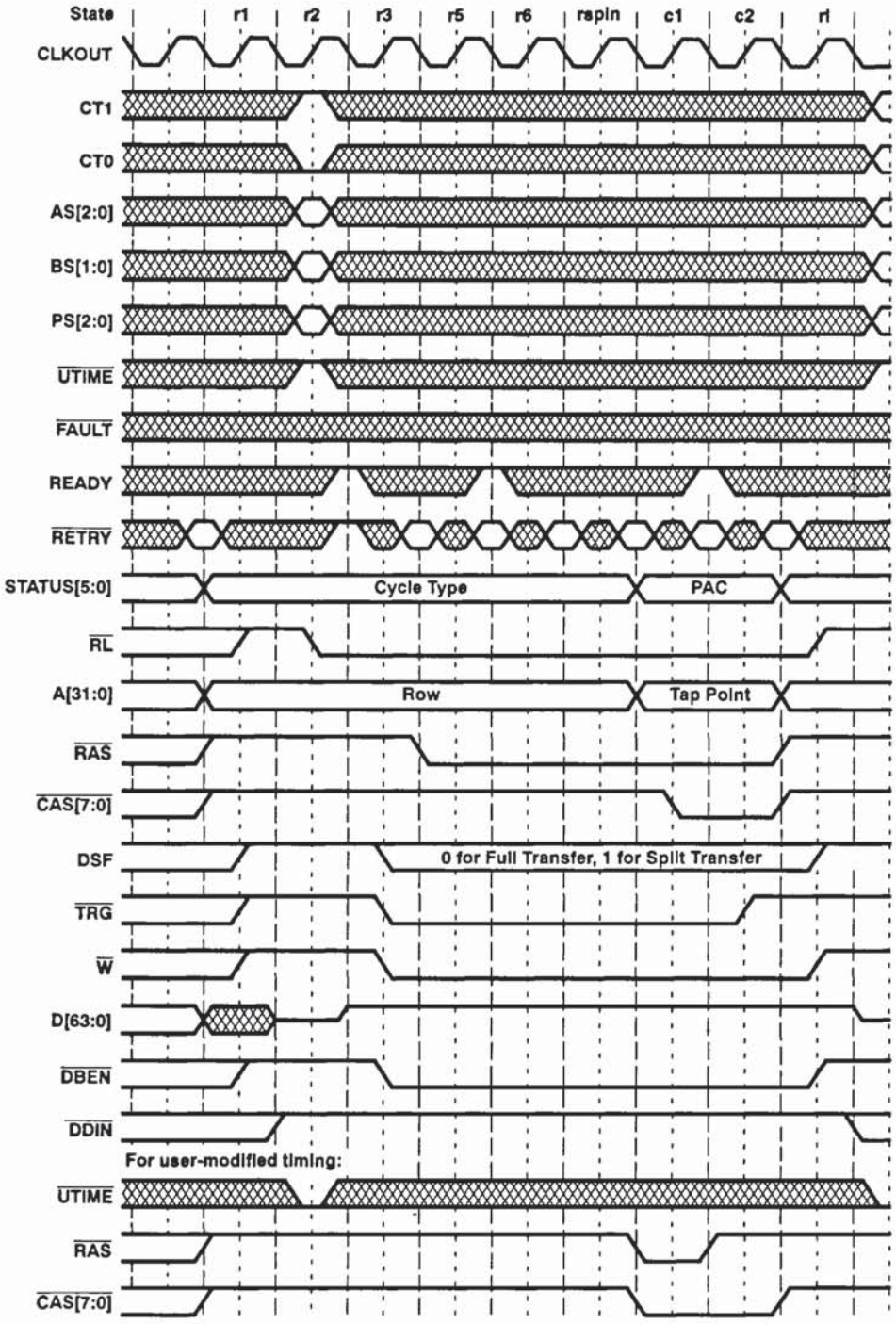


Figure 29. 2-Cycle/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

ADVANCE INFORMATION

transfer cycle (continued)

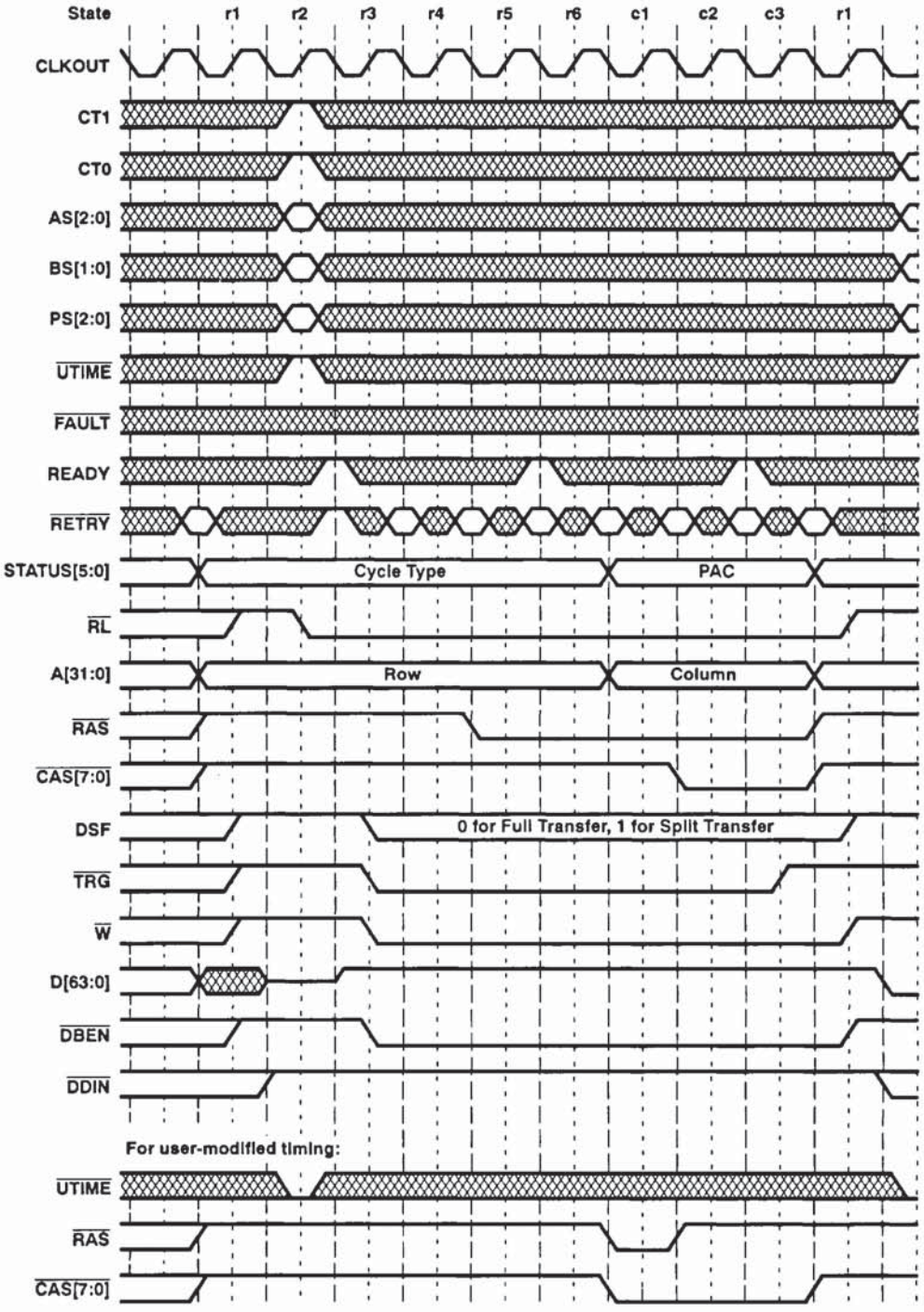


Figure 30. 3-Cycles/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

ADVANCE INFORMATION

refresh cycle

Refresh cycles are indicated on the STATUS[5:0] bus by a value of 000010 at row time. They are also characterized by the following signal activity:

- $\overline{\text{CAS}}$ falls prior to $\overline{\text{RAS}}$.
- All $\overline{\text{CAS}}$[7:0] are active.
- $\overline{\text{TRG}}$, $\overline{\text{W}}$, and $\overline{\text{DBEN}}$ all remain inactive (high) because no data transfer occurs.
- DSF is active (high) at the fall of $\overline{\text{CAS}}$ and driven inactive prior to the fall of $\overline{\text{RAS}}$.
- The data bus is driven to the high-impedance state.
- The upper half of the address bus (A[31:16]) contains the refresh pseudo address and the lower half (A[15:0]) is driven to all zeros.
- If $\overline{\text{RETRY}}$ is asserted at any sample point during the cycle, the cycle timing is not modified. Instead, the pseudo address and backlog counters are simply not decremented.
- Selecting user-modified timing has no effect on the cycles.
- Upon completion of the refresh cycle, the memory interface returns to state r1 to await the next access.

refresh cycle (continued)

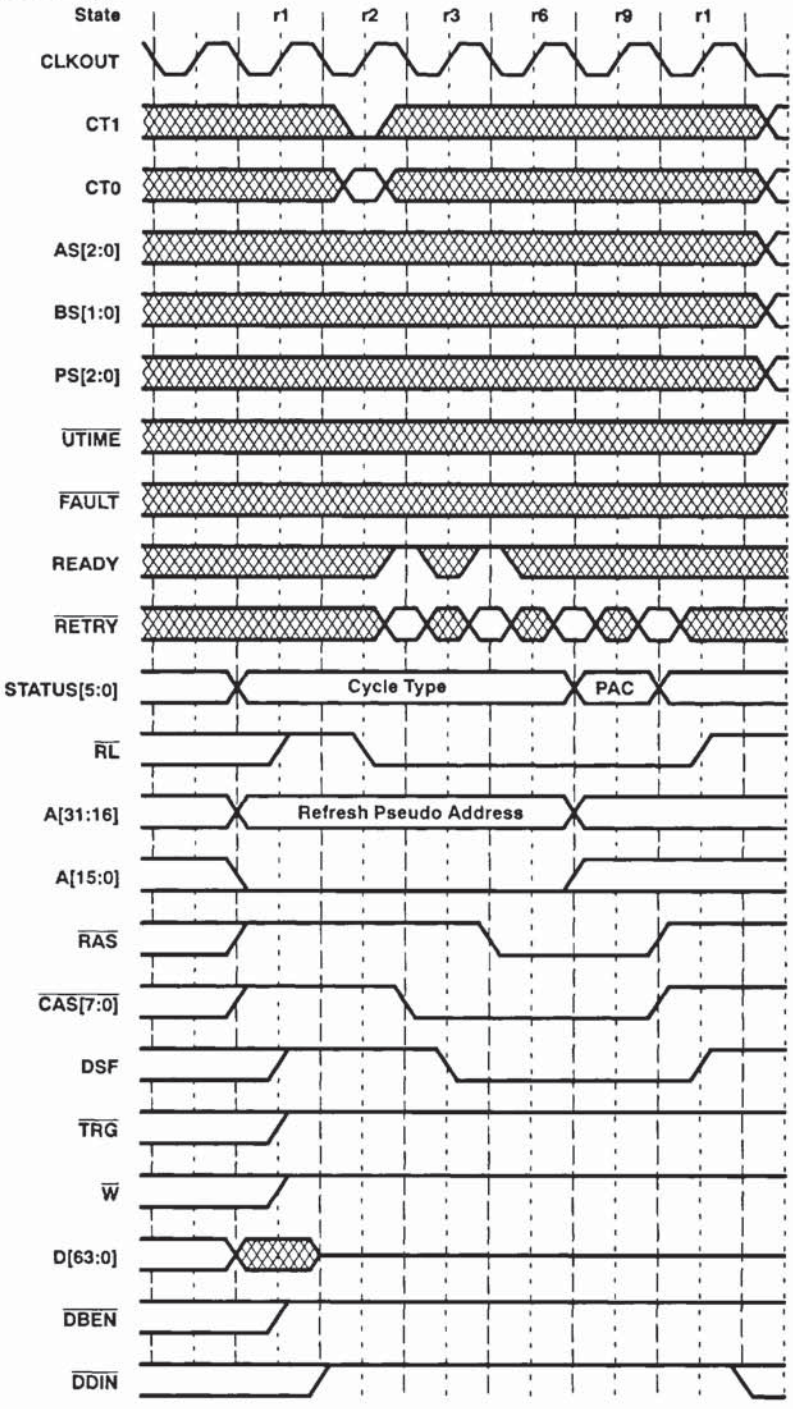


Figure 31. 1-Cycle/Column Refresh-Cycle Timing

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

refresh cycle (continued)

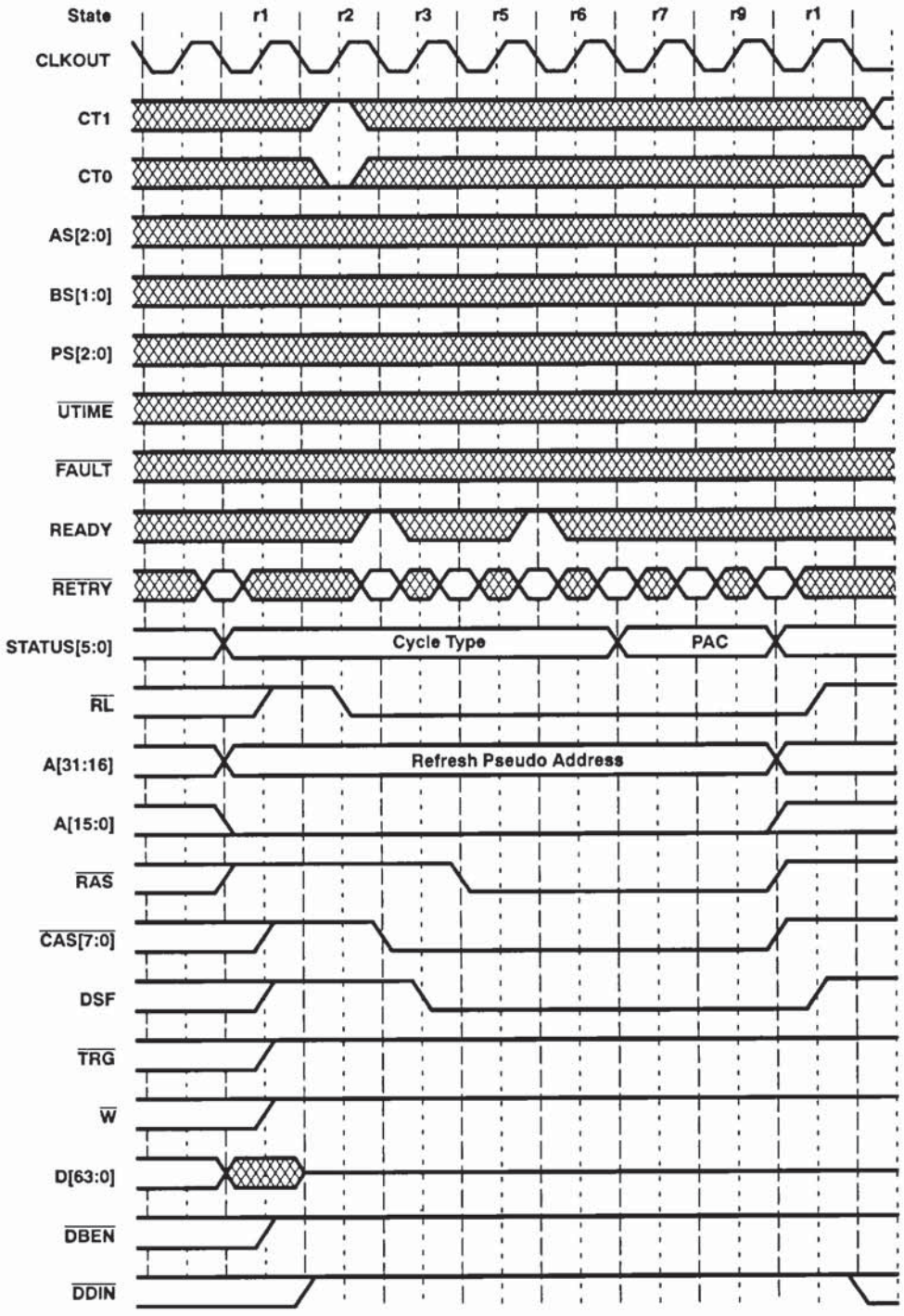


Figure 32. 2-Cycles/Column Refresh-Cycle Timing

refresh cycle (continued)

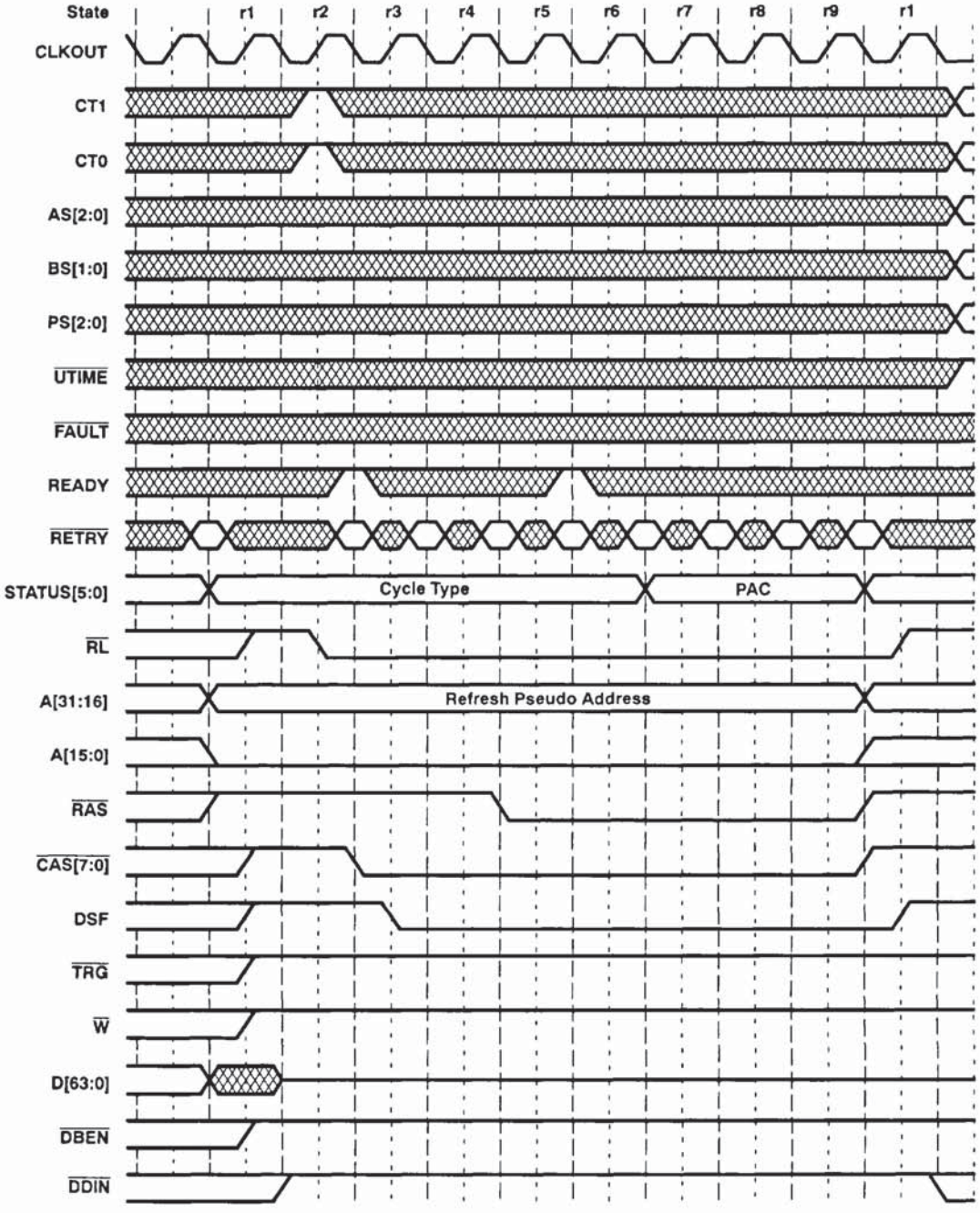


Figure 33. 3-Cycles/Column Refresh-Cycle Timing

absolute maximum ratings†

Supply voltage range, V_{CC} (see Note 1) . . . . . – 0.3 V to 4 V
Supply voltage range, V_{CC5} (see Note 1) . . . . . – 0.3 V to 7 V
Input voltage range . . . . . – 0.3 V to 7 V
Output voltage range . . . . . – 0.3 V to 3.3 V
Operating case temperature range, T_C . . . . . 0°C to 85°C
Storage temperature range . . . . . – 55°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	3.14	3.3	3.47	V
V_{CC5}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage (see Note 2)	0	0	0	V
I_{OH}	High-level output current			400	μA
I_{OL}	Low-level output current			2	mA
T_C	Operating case temperature	0		85	°C

NOTE 2: In order to minimize noise on V_{SS}, care should be taken to provide a minimum inductance path between the V_{SS} terminals and system ground.

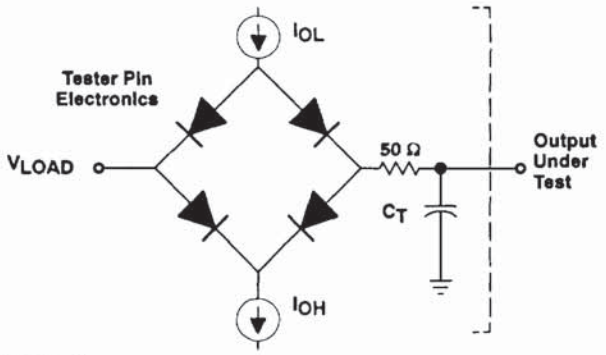
electrical characteristics over recommended ranges of supply voltage and operating case temperature (unless otherwise noted)

PARAMETER		TEST CONDITIONS‡	MIN	TYP§	MAX	UNIT
V_{IH}	High-level input voltage		2		V_{CC5} + 0.3	V
V_{IL}	Low-level input voltage		– 0.3		0.8	V
V_{OH}	High-level output voltage	V_{CC} = MIN, V_{CC5} = MIN, I_{OH} = MAX	2.6	¶		V
V_{OL}	Low-level output voltage	V_{CC} = MAX, V_{CC5} = MAX, I_{OH} = MIN			0.6	V
I_O	Output current, leakage (high impedance)	V_{CC} = MAX, V_{CC5} = MAX, V_O = 2.8 V			20	μA
		V_{CC} = MAX, V_{CC5} = MAX, V_O = 0.6 V			– 20	
I_I	Input current	V_I = V_{SS} to V_{CC5}			±20	μA
I_{CC}	Supply current	V_{CC} = MAX			2.5	A
I_{CC5}	Supply current	V_{CC5} = MAX			250	mA
C_i	Input capacitance			10		pF
C_o	Output capacitance			10		pF

‡ For conditions shown as MIN/MAX, use the appropriate value specified under the recommended operating conditions.
§ All typical values are at V_{CC} = 3 V, V_{CC5} = 5 V, T_A = 25°C
¶ Typical steady-state V_{OH} will not exceed V_{CC}.

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION



Where: I_{OL} = 2.0 mA (all outputs)
I_{OH} = 400 µA (all outputs)
V_{LOAD} = 1.5 V
C_T = 60-pF typical load circuit capacitance

Figure 34. Test Load Circuit

signal transition levels

TTL-output levels are driven to a minimum logic-high level of 2.6 V and to a maximum logic-low level of 0.6 V. Figure 35 shows the TTL-level outputs.

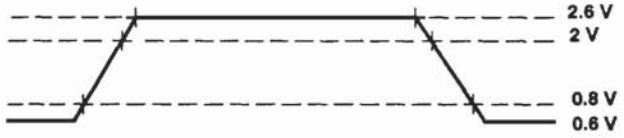


Figure 35. TTL-Level Outputs

TTL-output transition times are specified as follows:

- For a *high-to-low transition*, the level at which the output is said to be no longer high is 2 V, and the level at which the output is said to be low is 0.8 V.
- For a *low-to-high transition*, the level at which the output is said to be no longer low is 0.8 V, and the level at which the output is said to be high is 2 V.

Figure 36 shows the TTL-level inputs.

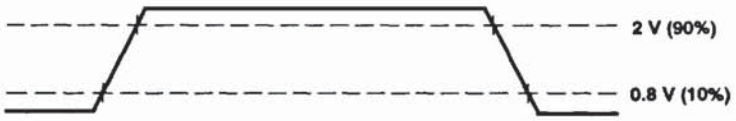


Figure 36. TTL-Level Inputs

TTL-compatible input transition times are specified as follows:

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

PARAMETER MEASUREMENT INFORMATION

timing parameter symbology

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

A	A[31:0]	RAS	$\overline{RAS}$
CA	A[31:0] (column address)	RDY	READY
CAS	$\overline{CAS}$[7:0]	RD	Read cycle
CFG	AS[2:0], BS[1:0], CT[1:0], PS[2:0], $\overline{UTIME}$	RST	$\overline{RESET}$
CKI	CLKIN	RTY	$\overline{RETRY}$
CK0	CLKOUT	REQ	REQ[1:0]
CMP	$\overline{RETRY}$, READY, $\overline{FAULT}$	RL	$\overline{RL}$
D	D[63:0]	SCK	SCLK0, SCLK1
DEN	$\overline{DBEN}$	STS	STATUS[5:0]
DIN	$\overline{DDIN}$	TCK	TCK
DSF	DSF	TDI	TDI
EIN	$\overline{EINT1}$, $\overline{EINT2}$, $\overline{EINT3}$	TDO	TDO
EMU	EMU0, EMU1	TMS	TMS
FLT	$\overline{FAULT}$	TRG	$\overline{TRG}$
FCK	FCLK0, FCLK1	TRS	$\overline{TRST}$
HAK	$\overline{HACK}$	UTM	$\overline{UTIME}$
HRQ	$\overline{HREQ}$	SI	$\overline{HSYNC0}$, $\overline{VSYNC0}$, $\overline{CSYNC0}$, $\overline{HSYNC1}$, $\overline{VSYNC1}$, or $\overline{CSYNC1}$
LIN	$\overline{LINT4}$	SY	$\overline{HSYNC0}$, $\overline{VSYNC0}$, $\overline{CSYNC0}$/$\overline{HBLNK0}$, $\overline{CBLNK0}$/$\overline{VBLNK0}$, $\overline{HSYNC1}$, $\overline{VSYNC1}$, $\overline{CSYNC1}$/$\overline{VBLNK1}$ or $\overline{CBLNK1}$/$\overline{VBLNK1}$
MID	A[31:0], STATUS[5:0]	W	Write cycle or $\overline{W}$
RA	A[31:0] (row address)	t_H	$t_{c(CKI)}$

Lowercase subscripts and their meanings 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 meanings are:

H	High
L	Low
V	Valid
Z	High impedance
X	Unknown, changing, or don't care level

general notes on timing parameters

The period of the output clock (CLKOUT) is twice the period of the input clock (CLKIN), or $2 \times t_{c(CKI)}$. The half cycle time (t_H) that appears in the following tables is one-half of the output clock period, or equal to the input clock period, $t_{c(CKI)}$.

All output signals from the 'C80 (including CLKOUT) are derived from an internal clock such that all output transitions for a given half cycle occur with a minimum of skewing relative to each other.

The signal combinations shown in the following timing diagrams may not necessarily represent actual cycles. For actual cycle examples, refer to the appropriate cycle description section of this data sheet.

ADVANCE INFORMATION

CLKIN timing requirements

		'C80-40		'C80-50		UNIT
		MIN	MAX	MIN	MAX	
$t_{c(CKI)}$	Period of CLKIN (t_H)	12.5		10		ns
$t_{w(CKIH)}$	Pulse duration of CLKIN high	4		4		ns
$t_{w(CKIL)}$	Pulse duration of CLKIN low	4		4		ns
$t_{t(CKI)}$	Transition time of CLKIN		1.5		1.5	ns

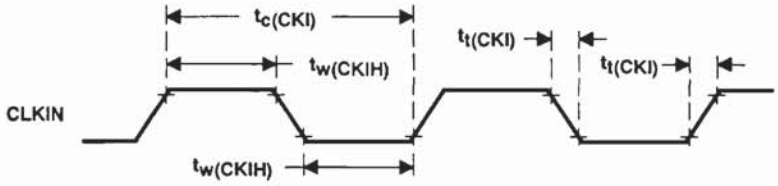


Figure 37. CLKIN Timing

local-bus switching characteristics: CLKOUT

		MIN	MAX	UNIT
$t_{c(CKO)}$	Period of CLKOUT	$2t_{c(CKI)}$ †		ns
$t_{w(CKOH)}$	Pulse duration of CLKOUT high	t_H-5		ns
$t_{w(CKOL)}$	Pulse duration of CLKOUT low	t_H-5		ns
$t_{t(CKO)}$	Transition time of CLKOUT		2	ns

† This is a functional minimum and is not tested. Parameter $t_{c(CKI)}$ may also be specified as t_H.

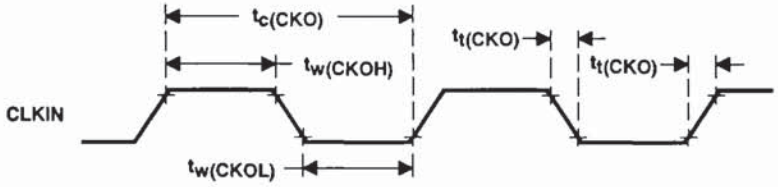


Figure 38. Local-Bus Switching: CLKOUT

ADVANCE INFORMATION

device reset timing requirements

			MIN	MAX	UNIT
$t_{d(UTMH-RSTH)}$	Delay time from $\overline{UTIME}$ no longer low to $\overline{RESET}$ high to configure big-endian operation			TBD	ns
$t_{d(HRQH-RSTH)}$	Delay time from $\overline{HREQ}$ no longer low to $\overline{RESET}$ high to configure self-bootstrap mode			$-2t_H$	ns
$t_{w(HRQL)}$	Pulse duration of $\overline{HREQ}$ to configure self-bootstrap mode		TBD		ns
$t_{su(HRQL-RSTH)}$	Setup time of $\overline{HREQ}$ low to $\overline{RESET}$ high to configure self-bootstrap mode		TBD		ns
$t_{su(UTMH-RSTH)}$	Setup time of $\overline{UTIME}$ low to $\overline{RESET}$ high to configure big-endian operation		TBD		ns
$t_{w(RSTL)}$	Pulse duration of $\overline{RESET}$ low	Reset during active operation	TBD		ns
		Initial reset during power up	TBD		ns
$t_{w(UTML)}$	Pulse duration of $\overline{UTIME}$ low to configure big-endian operation		TBD		ns

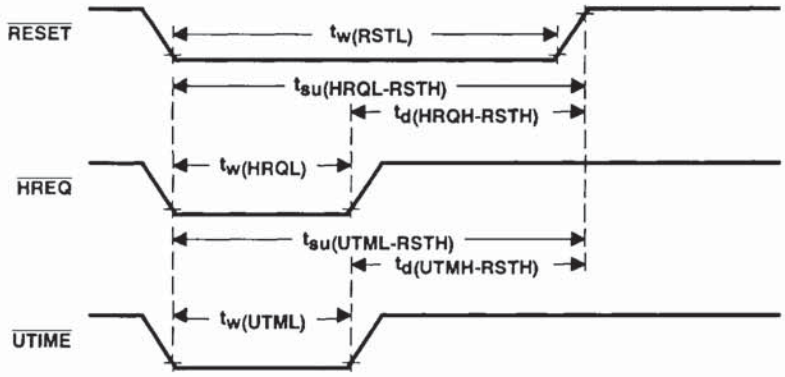


Figure 39. Device Reset Timing

ADVANCE INFORMATION

general output-signal switching characteristics

The following general descriptions and values apply to all TMS320C80 output signals unless otherwise specifically given. The variable *n* as used in the values represents the integral number of half cycles between the transitions of the two outputs in question. The abbreviation SG is used in the parameter symbols to mean any output signal; H and L are used for high and low, as usual.

		MIN	MAX	UNIT
$t_{h(SGL-SGL)}$	Hold time, output signal low after output signal low	$nt_H - 7$		ns
$t_{h(SGL-SGH)}$	Hold time, output signal high after output signal low	$nt_H - 7$		ns
$t_{h(SGH-SGL)}$	Hold time, output signal low after output signal high	$nt_H - 7$		ns
$t_{h(SGH-SGH)}$	Hold time, output signal high after output signal high	$nt_H - 7$		ns
$t_{d(SGH-SGL)}$	Delay time, output signal no longer low to output signal low		$nt_H + 7$	ns
$t_{d(SGH-SGH)}$	Delay time, output signal no longer low to output signal high		$nt_H + 7$	ns
$t_{d(SGL-SGL)}$	Delay time, output signal no longer high to output signal low		$nt_H + 7$	ns
$t_{d(SGL-SGH)}$	Delay time, output signal no longer high to output signal high		$nt_H + 7$	ns
$t_{t(SG)}$	Output signal transition time		2	ns
$t_{w(SGH)}$	Pulse duration, output signal high	$nt_H - 6$		ns
$t_{w(SGL)}$	Pulse duration, output signal low	$nt_H - 5$		ns

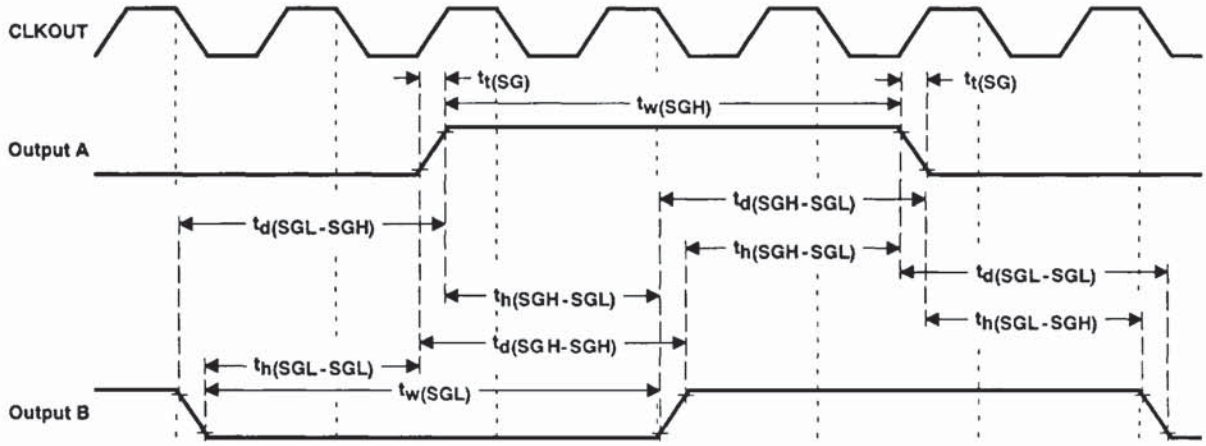


Figure 40. General Output-Signal Switching Characteristics

ADVANCE INFORMATION

local-bus timing requirements: cycle-configuration inputs

		MIN	MAX	UNIT
$t_{su(CFGV-CKOH)}$	Setup time, AS, BS, CT, PS, and $\overline{UTIME}$ valid to CLKOUT no longer low	8		ns
$t_{h(CKOH-CFGV)}$	Hold time, AS, BS, CT, PS, and $\overline{UTIME}$ valid after CLKOUT high	0		ns
$t_{a(MIDV-CFGV)}$	Access time, AS, BS, CT, PS, and $\overline{UTIME}$ valid after memory identification (A, STATUS) valid		$3t_H - 12$	ns
$t_{a(RLH-CFGV)}$	Access time, AS, BS, CT, PS, and $\overline{UTIME}$ valid after $\overline{RL}$ high		$2t_H - 12$	ns

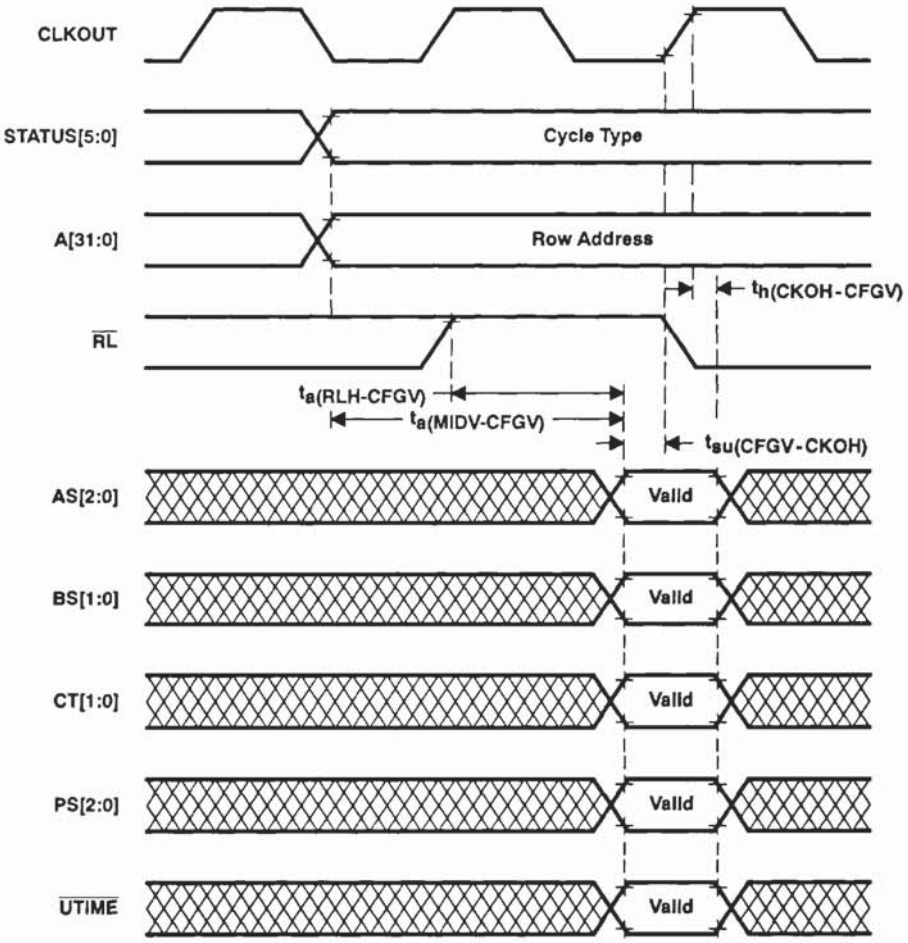


Figure 41. Local-Bus Timing: Cycle-Configuration Inputs

ADVANCE INFORMATION

local-bus timing requirements: $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ inputs (see Note 3)

		MIN	MAX	UNIT
$t_{su(CMPV-CKOL)}$	Setup time, $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ valid to CLKOUT no longer high	8†		ns
$t_{h(CKOL-CMPV)}$	Hold time, $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ valid after CLKOUT low	0†		ns
$t_{a(MIDV-CMPV)}$	Access time, $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ valid (R3 state) after memory identification (A, STATUS) valid		$4t_H - 12$	ns
$t_{a(RLH-CMPV)}$	Access time, $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ valid after $\overline{\text{RL}}$ high		$3t_H - 12$	ns

† $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ are synchronous inputs sampled during the high to low transition of CLKOUT. The specified setup and hold times must be met for each edge where the inputs are sampled for proper device operation.

NOTE 3: $\overline{\text{FAULT}}$ is sampled at the beginning of each r3 state, READY is sampled at the beginning of each r3, r6, and c3 state, and $\overline{\text{RETRY}}$ is sampled at the beginning of each r3 state and every state thereafter until a new page access begins.

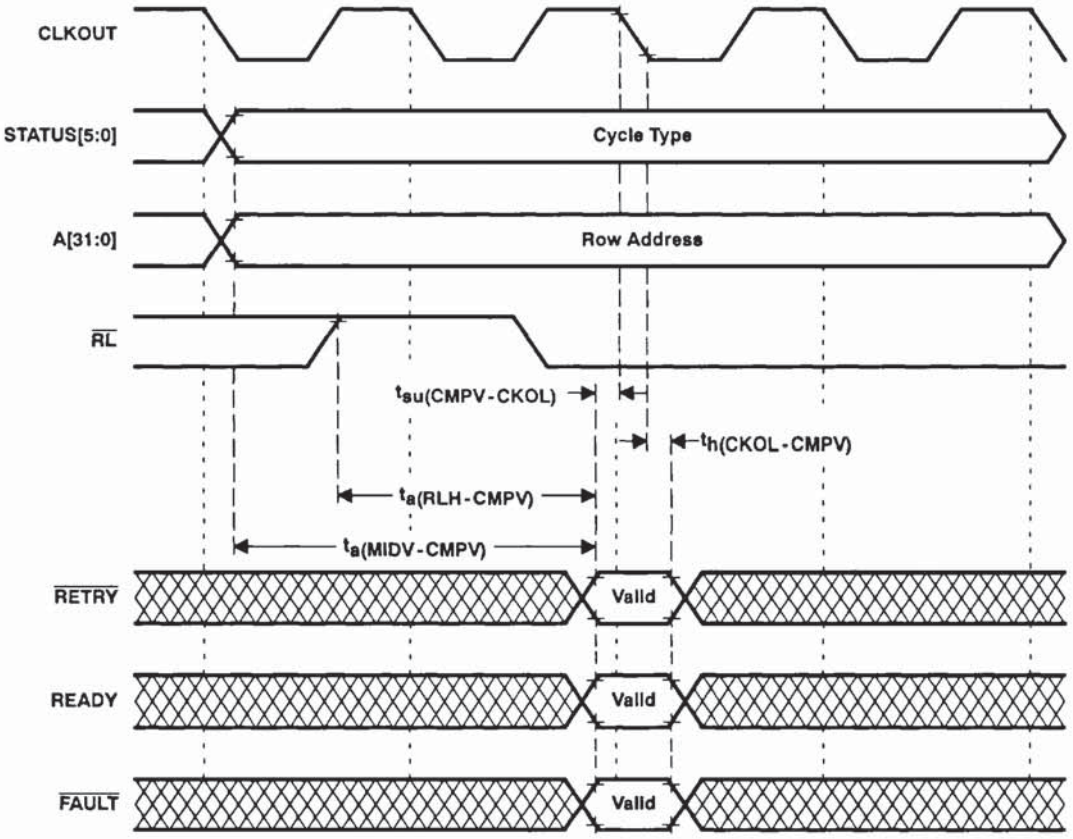


Figure 42. Local-Bus Timing: $\overline{\text{FAULT}}$, READY, and $\overline{\text{RETRY}}$ Inputs

ADVANCE INFORMATION

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

local-bus timing requirements: $\overline{\text{DBEN}}$ and $\overline{\text{DDIN}}$

		MIN	MAX	UNIT
$t_{h(CASH-DENL)}$	Hold time, $\overline{\text{DBEN}}$ low after $\overline{\text{CAS}}$ high	$t_H - 7$		ns
$t_{h(DENH-DINV)}$	Hold time, $\overline{\text{DDIN}}$ valid after $\overline{\text{DBEN}}$ high	$t_H - 7$		ns
$t_{h(RLH-DINV)}$	Hold time, $\overline{\text{DDIN}}$ valid after $\overline{\text{RL}}$ high	$t_H - 7$		ns
$t_{su(DENL-CASL)}$	Setup time, $\overline{\text{DBEN}}$ low to $\overline{\text{CAS}}$ no longer high	$2t_H - 7$		ns
$t_{su(DINV-DENL)}$	Setup time, $\overline{\text{DDIN}}$ valid before $\overline{\text{DBEN}}$ no longer high	$3t_H - 7$		ns
$t_{su(DINV-RLL)}$	Setup time, $\overline{\text{DDIN}}$ valid before $\overline{\text{RL}}$ no longer high	$t_H - 7$		ns

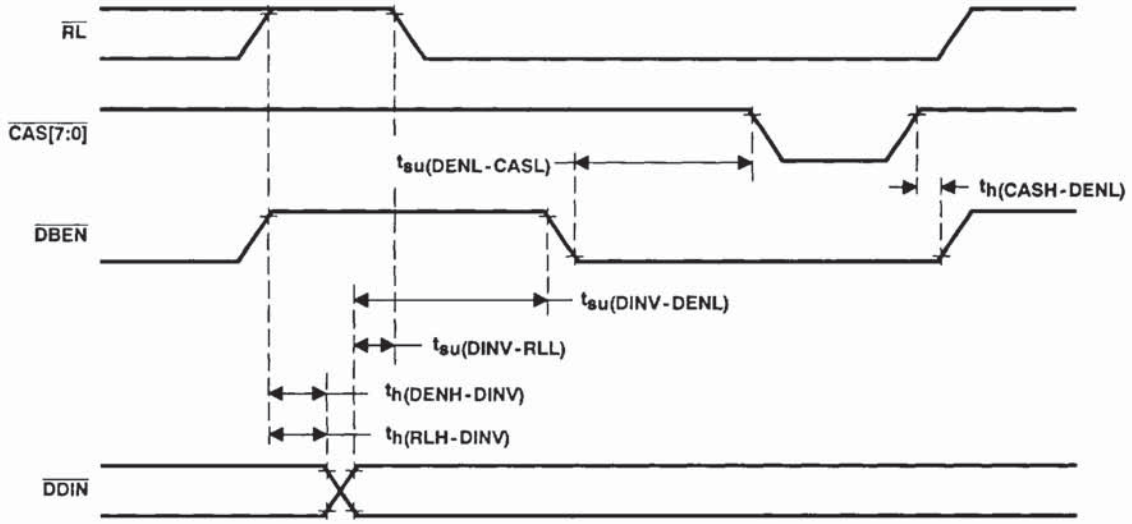


Figure 43. Local-Bus Timing: $\overline{\text{DBEN}}$ and $\overline{\text{DDIN}}$

ADVANCE INFORMATION

local-bus timing: memory control signals – 3-cycles/column

		MIN	MAX	UNIT
$t_{a(CAV-DV)}$	Access time, D (data in) valid from A (column address)		$6t_H - 10$	ns
$t_{a(CASL-DV)}$	Access time, D (data in) valid from $\overline{CAS}$ low		$4t_H - 10$	ns
$t_{a(CASH-DV)}$	Access time, D (data in) valid from $\overline{CAS}$ high (precharge)		$6t_H - 10$	ns
$t_{a(RASL-DV)}$	Access time, D (data in) valid from $\overline{RAS}$ low		$10t_H - 10$	ns
$t_{a(TRGL-DV)}$	Access time, D (data in) valid from $\overline{TRG}$ low		$7t_H - 10$	ns
$t_{a(RASL-RDYV)}$	Access time, $\overline{RAS}$ low to READY valid		$2t_H - 12$	ns
$t_{a(CASL-RDYV)}$	Access time, $\overline{CAS}$ low to READY valid		$2t_H - 12$	ns
$t_{c(RD)}$	Cycle time, read ($\overline{W}$ high) (see Note 4)	$18t_H - 6$		ns
$t_{c(W)}$	Cycle time, write ($\overline{W}$ low) (see Note 4)	$18t_H - 6$		ns
$t_{c(PC)}$	Cycle time, page-mode read or write	$6t_H - 6$		ns
$t_{w(RASL)}$	Pulse duration, $\overline{RAS}$ low (see Note 4)	$10t_H - 5$		ns
$t_{w(RASH)}$	Pulse duration, $\overline{RAS}$ high	$8t_H - 6$		ns
$t_{w(CASL)}$	Pulse duration, $\overline{CAS}$ low	$4t_H - 5$		ns
$t_{w(CASH)}$	Pulse duration, $\overline{CAS}$ high	$4t_H - 6$		ns
$t_{w(CASH)PG}$	Pulse duration, $\overline{CAS}$ high (page-mode cycle)	$2t_H - 5$		ns
$t_{w(WL)}$	Pulse duration, $\overline{W}$ low (see Note 4)	$8t_H - 5$		ns
$t_{w(TRGL)}$	Pulse duration, $\overline{TRG}$ low (see Note 4)	$8t_H - 5$		ns
$t_{w(TRHG)}$	Pulse duration, $\overline{TRG}$ high	$4t_H - 6$		ns
$t_{su(CAV-CASL)}$	Setup time, A (column address) valid before $\overline{CAS}$ no longer high	$2t_H - 7$		ns
$t_{su(RAV-RASL)}$	Setup time, A (row address) valid before $\overline{RAS}$ no longer high	$8t_H - 7$		ns
$t_{su(DV-CASL)}$	Setup time, D valid before $\overline{CAS}$ no longer high	$2t_H - 7$		ns
$t_{su(WL-CASL)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer high	$3t_H - 7$		ns
$t_{su(WL-CASH)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer low	$7t_H - 7$		ns
$t_{su(WH-RASL)}$	Setup time, $\overline{W}$ high before $\overline{RAS}$ no longer high	$7t_H - 7$		ns
$t_{su(WL-RASH)}$	Setup time, $\overline{W}$ low before $\overline{RAS}$ no longer low (see Note 4)	$7t_H - 7$		ns
$t_{su(DSFV-RASL)}$	Setup time, DSF valid before $\overline{RAS}$ no longer high	$3t_H - 7$		ns
$t_{su(DSFV-CASL)}$	Setup time, DSF valid before $\overline{CAS}$ no longer high	$3t_H - 7$		ns
$t_{su(DV-RASL)}$	Setup time, D valid (high) before $\overline{RAS}$ no longer high (write-transfer cycle)	$4t_H - 7$		ns
$t_{su(TRGH-RASL)}$	Setup time, $\overline{TRG}$ high before $\overline{RAS}$ no longer high	$7t_H - 7$		ns
$t_{su(TRGL-RASL)}$	Setup time, $\overline{TRG}$ low to $\overline{RAS}$ no longer high	$3t_H - 7$		ns
$t_{su(RASL-CASH)}$	Setup time, $\overline{RAS}$ low to $\overline{CAS}$ no longer low	$10t_H - 7$		ns
$t_{su(CASH-RASL)}$	Setup time, $\overline{CAS}$ high to $\overline{RAS}$ no longer high	$8t_H - 7$		ns
$t_{su(CASL-RAS))}$	Setup time, $\overline{CAS}$ low to $\overline{RAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns
$t_{su(RASH-CASL)}$	Setup time, $\overline{RAS}$ high to $\overline{CAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns
$t_{su(DZ-CASL)}$	Setup time, D in high-impedance state to $\overline{CAS}$ no longer high	$12t_H - 7$		ns
$t_{su(DZ-TRGL)}$	Setup time, D in high-impedance state to $\overline{TRG}$ no longer high	$9t_H - 7$		ns
$t_{su(WL-RASL)}$	Setup time, $\overline{W}$ low to $\overline{RAS}$ no longer high	$3t_H - 7$		ns
$t_{su(TRGH-D)}$	Setup time, $\overline{TRG}$ high to D driven	$3t_H - 7$		ns
$t_{su(AV-CASL)}$	Setup time, address valid to $\overline{CAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns

NOTE 4: Minimum value for this parameter occurs during cycles with a single column access.

local bus timing: memory control signals – three cycles/column (continued)

		MIN	MAX	UNIT
$t_{h(CASH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ high (precharge)	$6t_H - 7$		ns
$t_{h(CASL-CAV)}$	Hold time, A (column address) valid after $\overline{CAS}$ low	$4t_H - 7$		ns
$t_{h(RASL-RAV)}$	Hold time, A (row address) valid after $\overline{RAS}$ low	$4t_H - 7$		ns
$t_{h(RASL-CAV)}$	Hold time, A (column address) valid after $\overline{RAS}$ low	$10t_H - 7$		ns
$t_{h(CASL-DV)}$	Hold time, D valid after $\overline{CAS}$ low	$4t_H - 7$		ns
$t_{h(RASL-DV)}$	Hold time, D valid after $\overline{RAS}$ low	$10t_H - 7$		ns
$t_{h(CASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{CAS}$ high	$5t_H - 7$		ns
$t_{h(RASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ high	$5t_H - 7$		ns
$t_{h(CASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{CAS}$ low	$5t_H - 7$		ns
$t_{h(RASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{RAS}$ low (write-transfer cycle)	$11t_H - 7$		ns
$t_{h(RASL-DSFV)}$	Hold time, DSF valid after $\overline{RAS}$ low	$3t_H - 7$		ns
$t_{h(CASL-DSFV)}$	Hold time, DSF valid after $\overline{CAS}$ low	$5t_H - 7$		ns
$t_{h(RASL-DSFV)COL}$	Hold time, column-time DSF valid after $\overline{RAS}$ low	$11t_H - 7$		ns
$t_{h(RASL-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ low	$3t_H - 7$		ns
$t_{h(RASL-TRGH)}$	Hold time, $\overline{TRG}$ high after $\overline{RAS}$ low	$3t_H - 7$		ns
$t_{h(RASL-CASL)}$	Hold time, $\overline{CAS}$ low after $\overline{RAS}$ low (refresh cycle)	$10t_H - 7$		ns
$t_{h(CAV-RASL)}$	Hold time, $\overline{RAS}$ low after A (column address) valid	$6t_H - 7$		ns
$t_{h(CAV-CASL)}$	Hold time, $\overline{CAS}$ low after A (column address) valid	$6t_H - 7$		ns
$t_{h(CASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (read cycle)	$5t_H - 7$		ns
$t_{h(CASL-TRG)TR}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (transfer cycle)	$4t_H - 7$		ns
$t_{h(CASL-TRGL)}$	Hold time, $\overline{TRG}$ low after A (column address) valid (transfer cycle)	$5t_H - 7$		ns
$t_{h(RASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{RAS}$ low (transfer cycle)	$9t_H - 7$		ns
$t_{h(RASL-CASH)}$	Hold time, $\overline{CAS}$ high after $\overline{RAS}$ low	$6t_H - 7$		ns
$t_{h(TRGH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{TRG}$ high (transfer cycle)	$t_H - 7$		ns
$t_{h(CASH-DV)}$	Hold time, D (data in) valid after $\overline{CAS}$ high	0		ns
$t_{h(CASL-AV)}$	Hold time, A valid after $\overline{CAS}$ low (refresh cycle)	$14t_H - 7$		ns
$t_{h(RASL-AV)}$	Hold time, A valid after $\overline{RAS}$ low (refresh cycle)	$10t_H - 7$		ns
$t_{h(CASL-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ low	$4t_H - 7$		ns
$t_{d(RASL-CASL)}$	Delay time, $\overline{RAS}$ no longer high to $\overline{CAS}$ low		$6t_H + 7$	ns
$t_{d(RASL-CAV)}$	Delay time, $\overline{RAS}$ no longer high to A (column address) valid		$4t_H + 7$	ns

ADVANCE INFORMATION

ADVANCE INFORMATION

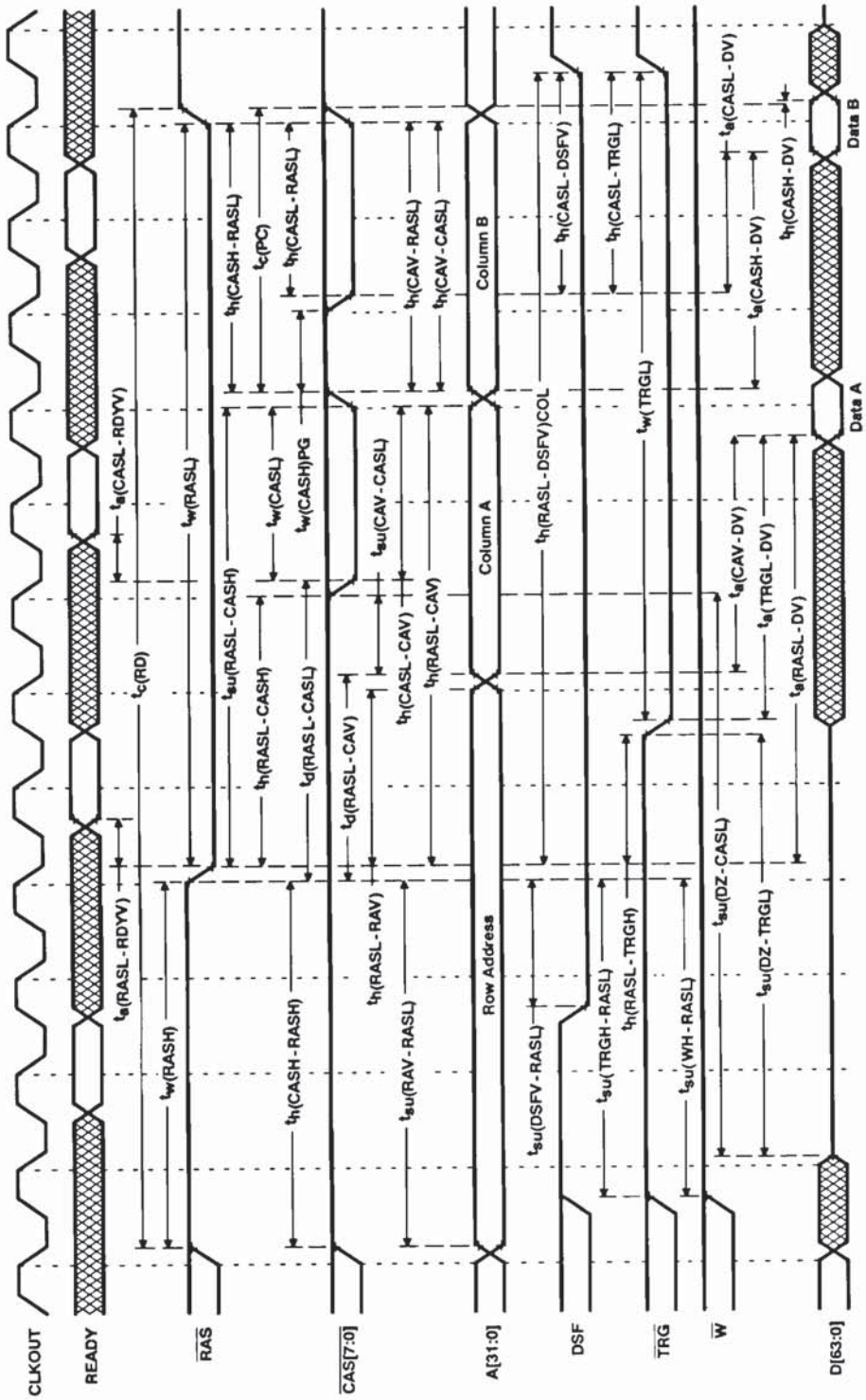
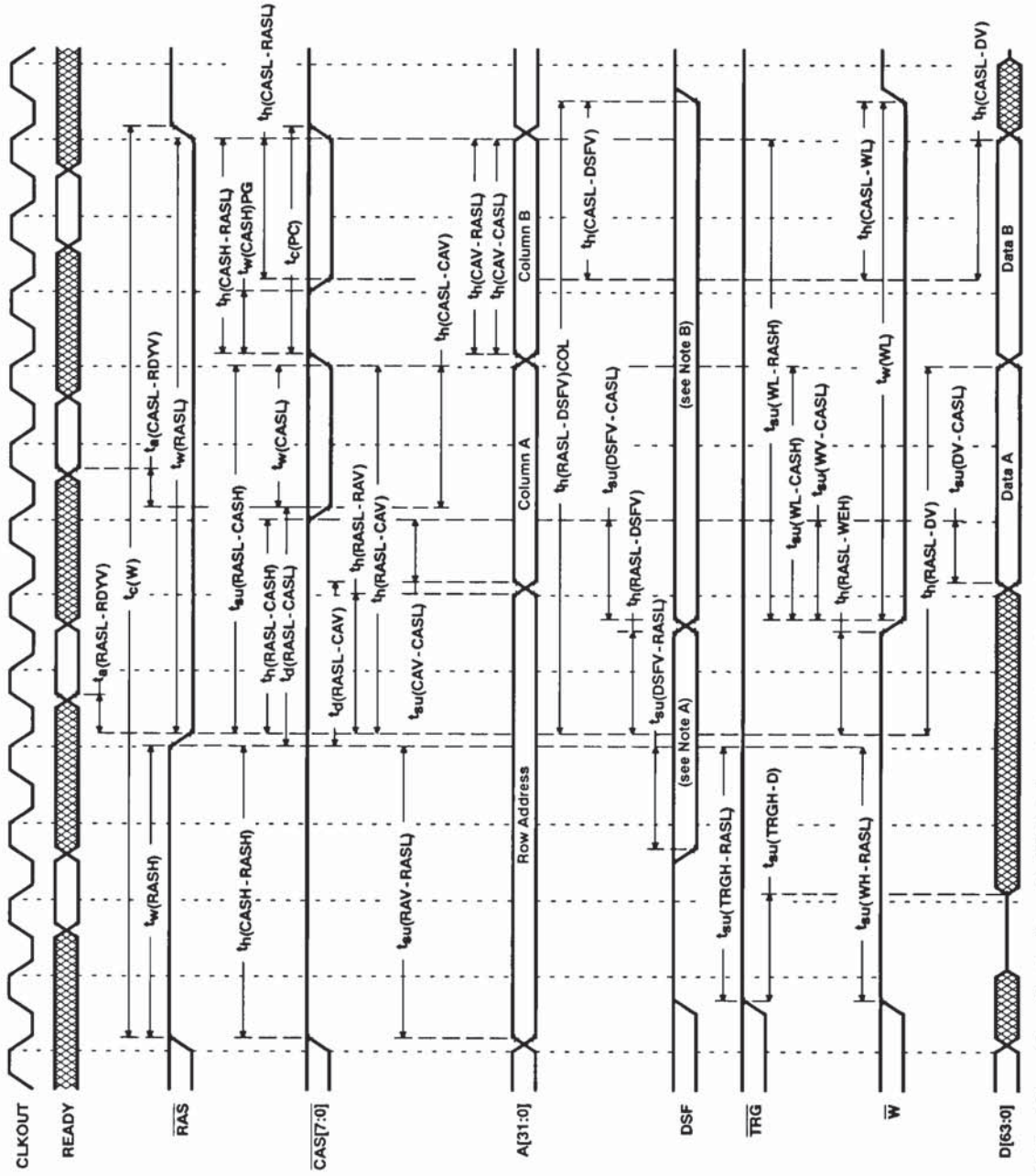


Figure 44. 3-Cycles/Column Read-Cycle Timing

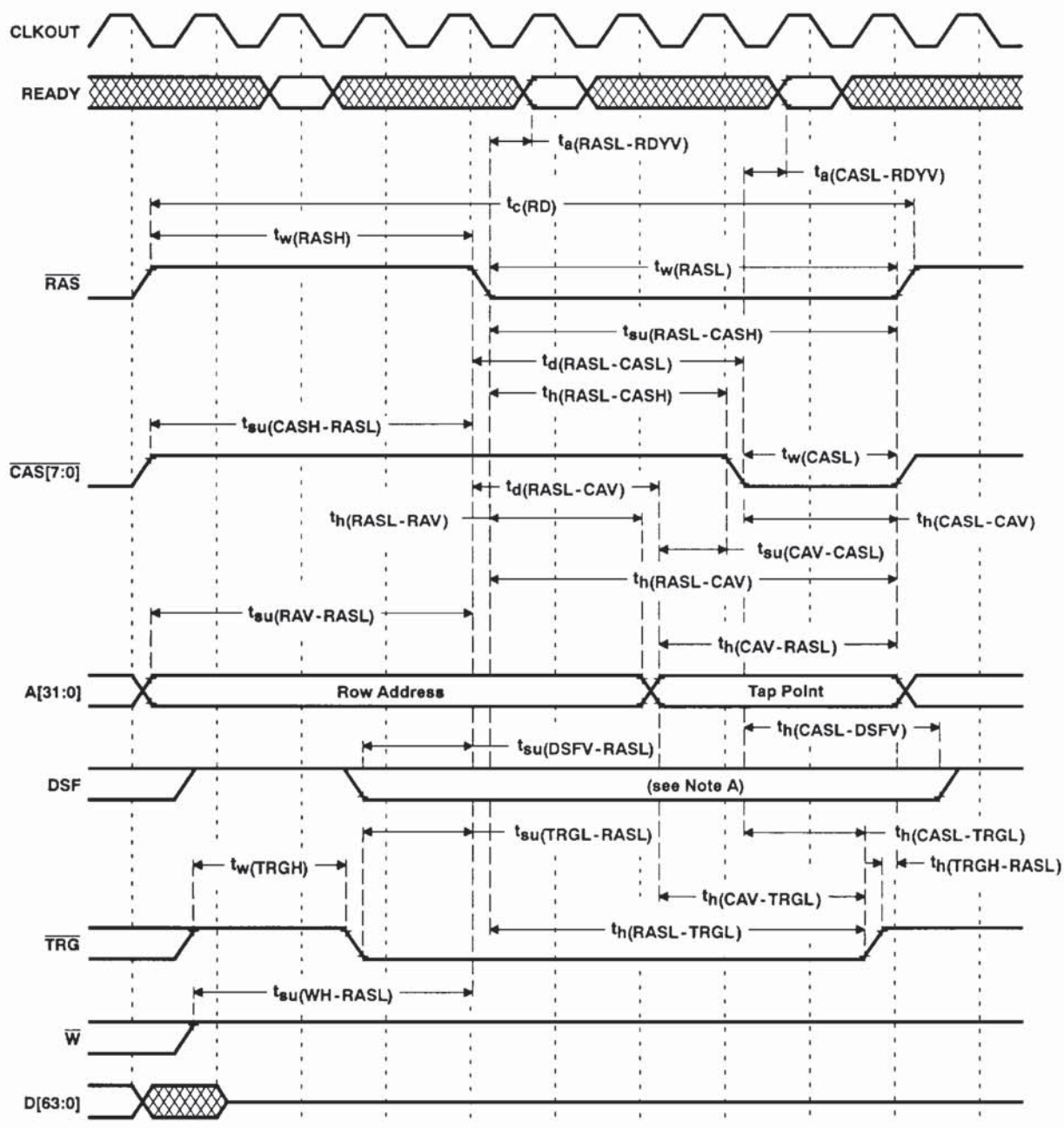
TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443



NOTES: A. DSF = 0 for a write or block-write cycle. DSF = 1 for a load-color-register cycle.
B. DSF = 0 for a write cycle. DSF = 1 for a block-write or load-color-register cycle.

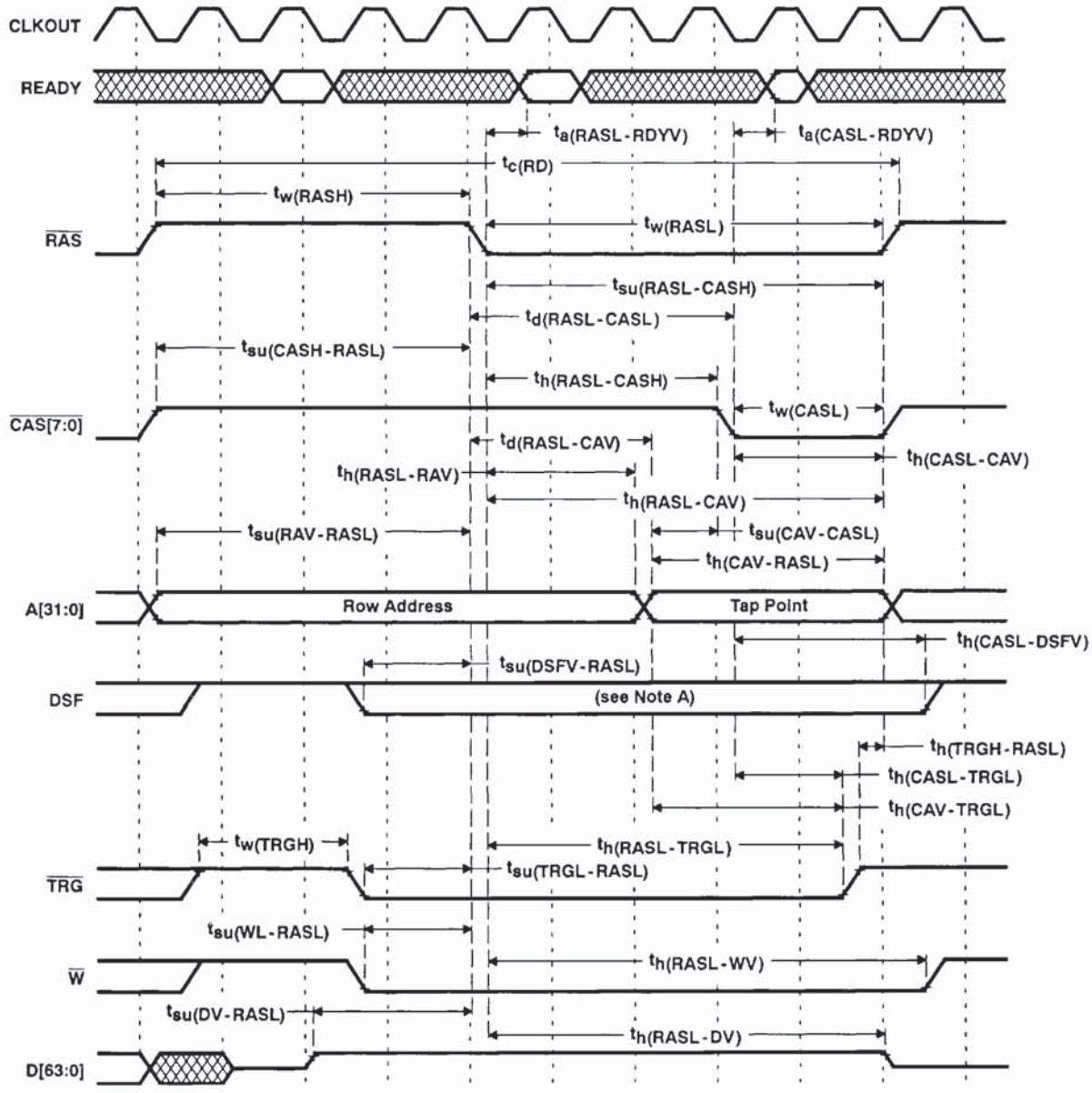
Figure 45. 3-Cycles/Column Write-, Block-Write-, and Load-Color-Register-Cycle Timing

ADVANCE INFORMATION



NOTE A: DSF = 0 for a read-transfer cycle. DSF = 1 for a split-register read-transfer cycle.

Figure 46. 3-Cycles/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing



NOTE A: DSF = 0 for a write-transfer cycle. DSF = 1 for a split-register write-transfer cycle.

Figure 47. 3-Cycles/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

ADVANCE INFORMATION

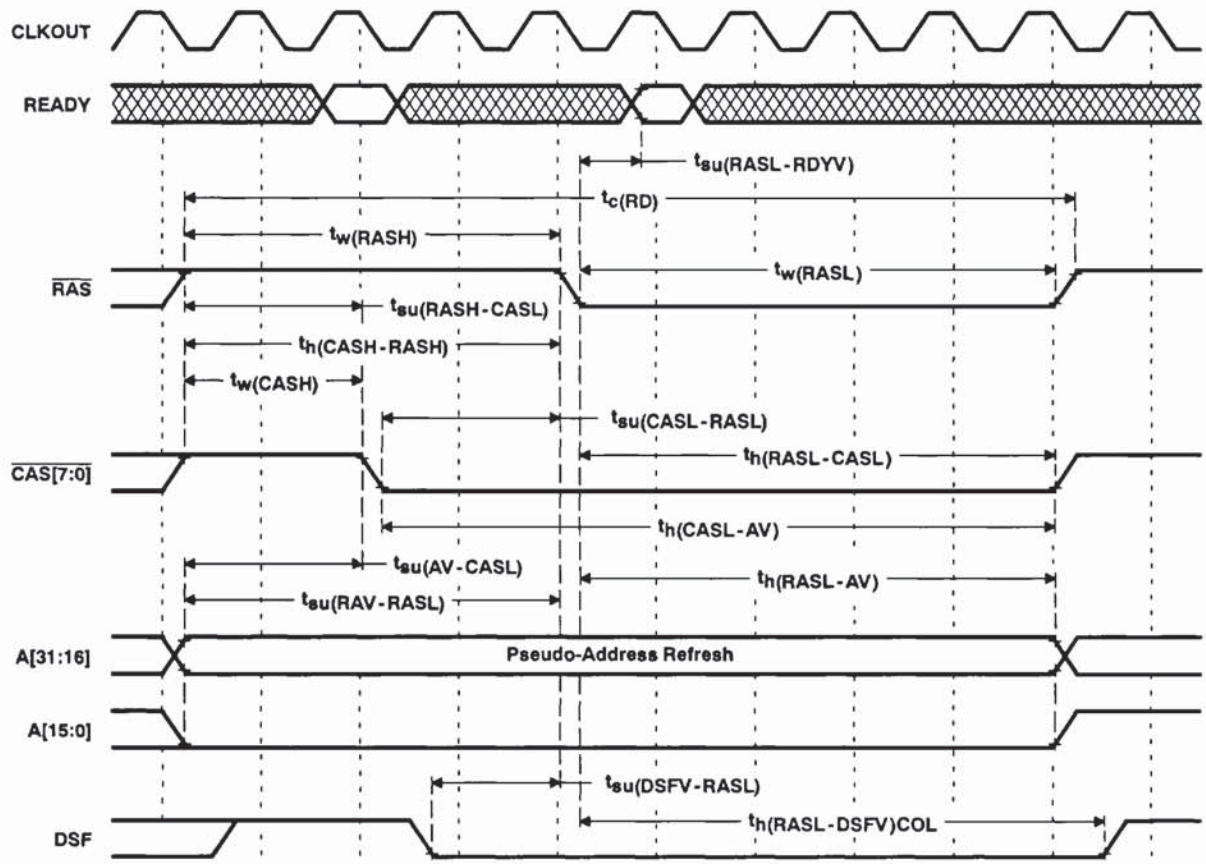


Figure 48. 3-Cycles/Column Refresh-Cycle Timing

local-bus timing: memory control signals – 2-cycles/column

	PARAMETER	MIN	MAX	UNIT
$t_{a(CAV-DV)}$	Access time, D valid (in) from A (column address)		$4t_H - 10$	ns
$t_{a(CASL-DV)}$	Access time, D valid (in) from $\overline{CAS}$ low		$3t_H - 13$	ns
$t_{a(CASH-DV)}$	Access time, D valid (in) from $\overline{CAS}$ high (precharge)		$4t_H - 10$	ns
$t_{a(RASL-DV)}$	Access time, D valid (in) from $\overline{RAS}$ low		$8t_H - 10$	ns
$t_{a(TRGL-DV)}$	Access time, D valid (in) from $\overline{TRG}$ low		$5t_H - 10$	ns
$t_{a(RASL-RDYV)}$	Access time, $\overline{RAS}$ low to READY valid		$2t_H - 12$	ns
$t_{a(CASL-RDYV)}$	Access time, $\overline{CAS}$ low to READY valid		$t_H - 12$	ns
$t_{c(RD)}$	Cycle time, read (see Note 4)	$14t_H - 6$		ns
$t_{c(W)}$	Cycle time, write (see Note 4)	$18t_H - 6$		ns
$t_{c(PC)}$	Cycle time, page-mode read or write	$4t_H - 6$		ns
$t_{w(RASL)}$	Pulse duration, $\overline{RAS}$ low (see Note 4)	$8t_H - 5$		ns
$t_{w(RASL)W}$	Pulse duration, $\overline{RAS}$ low, write cycle (see Note 4)	$10t_H - 5$		ns
$t_{w(RASH)}$	Pulse duration, $\overline{RAS}$ high	$6t_H - 6$		ns
$t_{w(CASL)}$	Pulse duration, $\overline{CAS}$ low	$3t_H - 8$		ns
$t_{w(CASH)}$	Pulse duration, $\overline{CAS}$ high	$4t_H - 4$		ns
$t_{w(CASH)PG}$	Pulse duration, $\overline{CAS}$ high (page-mode cycle)	$t_H - 2$		ns
$t_{w(WL)}$	Pulse duration, $\overline{W}$ low (see Note 4)	$7t_H - 5$		ns
$t_{w(TRGL)}$	Pulse duration, $\overline{TRG}$ low (see Note 4)	$5t_H - 5$		ns
$t_{w(TRHG)}$	Pulse duration, $\overline{TRG}$ high	$4t_H - 6$		ns
$t_{su(CAV-CASL)}$	Setup time, A (column address) valid before $\overline{CAS}$ no longer high	$t_H - 7$		ns
$t_{su(RAV-RASL)}$	Setup time, A (row address) valid before $\overline{RAS}$ no longer high	$6t_H - 7$		ns
$t_{su(DV-CASL)}$	Setup time, data valid before $\overline{CAS}$ no longer high	$t_H - 7$		ns
$t_{su(WL-CASL)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer high	$4t_H - 7$		ns
$t_{su(WL-CASH)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer low	$7t_H - 7$		ns
$t_{su(WH-RASL)}$	Setup time, $\overline{W}$ high before $\overline{RAS}$ no longer high	$5t_H - 7$		ns
$t_{su(WL-RASH)}$	Setup time, $\overline{W}$ low before $\overline{RAS}$ no longer low (see Note 4)	$7t_H - 7$		ns
$t_{su(DSFV-RASL)}$	Setup time, DSF valid before $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(DSFV-CASL)}$	Setup time, DSF valid before $\overline{CAS}$ no longer high	$3t_H - 7$		ns
$t_{su(DV-RASL)}$	Setup time, D valid (high) before $\overline{RAS}$ no longer high (write-transfer cycle)	$2t_H - 7$		ns
$t_{su(TRGH-RASL)}$	Setup time, $\overline{TRG}$ high before $\overline{RAS}$ no longer high	$5t_H - 7$		ns
$t_{su(TRGL-RASL)}$	Setup time, $\overline{TRG}$ low to $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(RASL-CASH)}$	Setup time, $\overline{RAS}$ low to $\overline{CAS}$ no longer low	$8t_H - 7$		ns
$t_{su(CASH-RASL)}$	Setup time, $\overline{CAS}$ high to $\overline{RAS}$ no longer high	$6t_H - 7$		ns
$t_{su(CASL-RASL)}$	Setup time, $\overline{CAS}$ low to $\overline{RAS}$ no longer high (refresh cycle)	$2t_H - 7$		ns
$t_{su(RASH-CASL)}$	Setup time, $\overline{RAS}$ high to $\overline{CAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns
$t_{su(DZ-CASL)}$	Setup time, D in high-impedance state to $\overline{CAS}$ no longer high	$9t_H - 7$		ns
$t_{su(DZ-TRGL)}$	Setup time, D in high-impedance state to $\overline{TRG}$ no longer high	$7t_H - 7$		ns
$t_{su(WL-RASL)}$	Setup time, $\overline{W}$ low to $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(TRGH-D)}$	Setup time, $\overline{TRG}$ high to D driven	$3t_H - 7$		ns
$t_{su(AV-CASL)}$	Setup time, A valid to $\overline{CAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns

NOTE 4: Minimum value for this parameter occurs during cycles with a single column access.

ADVANCE INFORMATION

local-bus timing: memory control signals – 2-cycles/column (continued)

	PARAMETER	MIN	MAX	UNIT
$t_{h(CASH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ high (precharge)	$4t_H - 7$		ns
$t_{h(CASL-CAV)}$	Hold time, column address valid after $\overline{CAS}$ low	$3t_H - 7$		ns
$t_{h(RASL-RAV)}$	Hold time, row address valid after $\overline{RAS}$ low (see Note 5)	$4t_H - 7$		ns
$t_{h(RASL-CAV)}$	Hold time, column address valid after $\overline{RAS}$ low (see Note 5)	$8t_H - 7$		ns
$t_{h(CASL-DV)}$	Hold time, D valid after $\overline{CAS}$ low	$3t_H - 7$		ns
$t_{h(RASL-DV)}$	Hold time, D valid after $\overline{RAS}$ low	$10t_H - 7$		ns
$t_{h(CASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{CAS}$ high	$5t_H - 7$		ns
$t_{h(RASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ high	$5t_H - 7$		ns
$t_{h(CASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{CAS}$ low	$4t_H - 7$		ns
$t_{h(RASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{RAS}$ low (write-transfer cycle)	$9t_H - 7$		ns
$t_{h(RASL-DSFV)}$	Hold time, DSF valid after $\overline{RAS}$ low	$3t_H - 7$		ns
$t_{h(CASL-DSFV)}$	Hold time, DSF valid after $\overline{CAS}$ low	$4t_H - 7$		ns
$t_{h(RASL-DSFV)COL}$	Hold time, column-time DSF valid after $\overline{RAS}$ low (see Note 5)	$9t_H - 7$		ns
$t_{h(RASL-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ low	$3t_H - 7$		ns
$t_{h(RASL-TRGH)}$	Hold time, $\overline{TRG}$ high after $\overline{RAS}$ low	$3t_H - 7$		ns
$t_{h(RASL-CASL)}$	Hold time, $\overline{CAS}$ low after $\overline{RAS}$ low (refresh cycle)	$8t_H - 7$		ns
$t_{h(CAV-RASL)}$	Hold time, $\overline{RAS}$ low after column address valid	$4t_H - 7$		ns
$t_{h(CAV-CASL)}$	Hold time, $\overline{CAS}$ low after column address valid	$4t_H - 7$		ns
$t_{h(CASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (read cycle)	$4t_H - 7$		ns
$t_{h(CASL-TRGL)TR}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (transfer cycle)	$2t_H - 7$		ns
$t_{h(CAVL-TRGL)}$	Hold time, $\overline{TRG}$ low after column address valid (transfer cycle)	$3t_H - 7$		ns
$t_{h(RAS-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{RAS}$ low (transfer cycle)	$7t_H - 7$		ns
$t_{h(RASL-CASH)}$	Hold time, $\overline{CAS}$ high after $\overline{RAS}$ low (see Note 5)	$5t_H - 7$		ns
$t_{h(TRGH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{TRG}$ high (transfer cycle)	$t_H - 7$		ns
$t_{h(CASH-DV)}$	Hold time, D (in) valid after $\overline{CAS}$ high	0		ns
$t_{h(CASL-AV)}$	Hold time, A valid after $\overline{CAS}$ low (refresh cycle)	$10t_H - 7$		ns
$t_{h(RASL-AV)}$	Hold time, A valid after $\overline{RAS}$ low (refresh cycle)	$8t_H - 7$		ns
$t_{h(CASL-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ low	$3t_H - 7$		ns
$t_{d(RASL-CASL)}$	Delay time, $\overline{RAS}$ no longer high to $\overline{CAS}$ low		$5t_H + 7$	ns
$t_{d(RASL-CAV)}$	Delay time, $\overline{RAS}$ no longer high to A (column address) valid		$4t_H + 7$	ns

NOTE 5: During write cycles, the actual value of this parameter is increased by $2t_H$.

ADVANCE INFORMATION

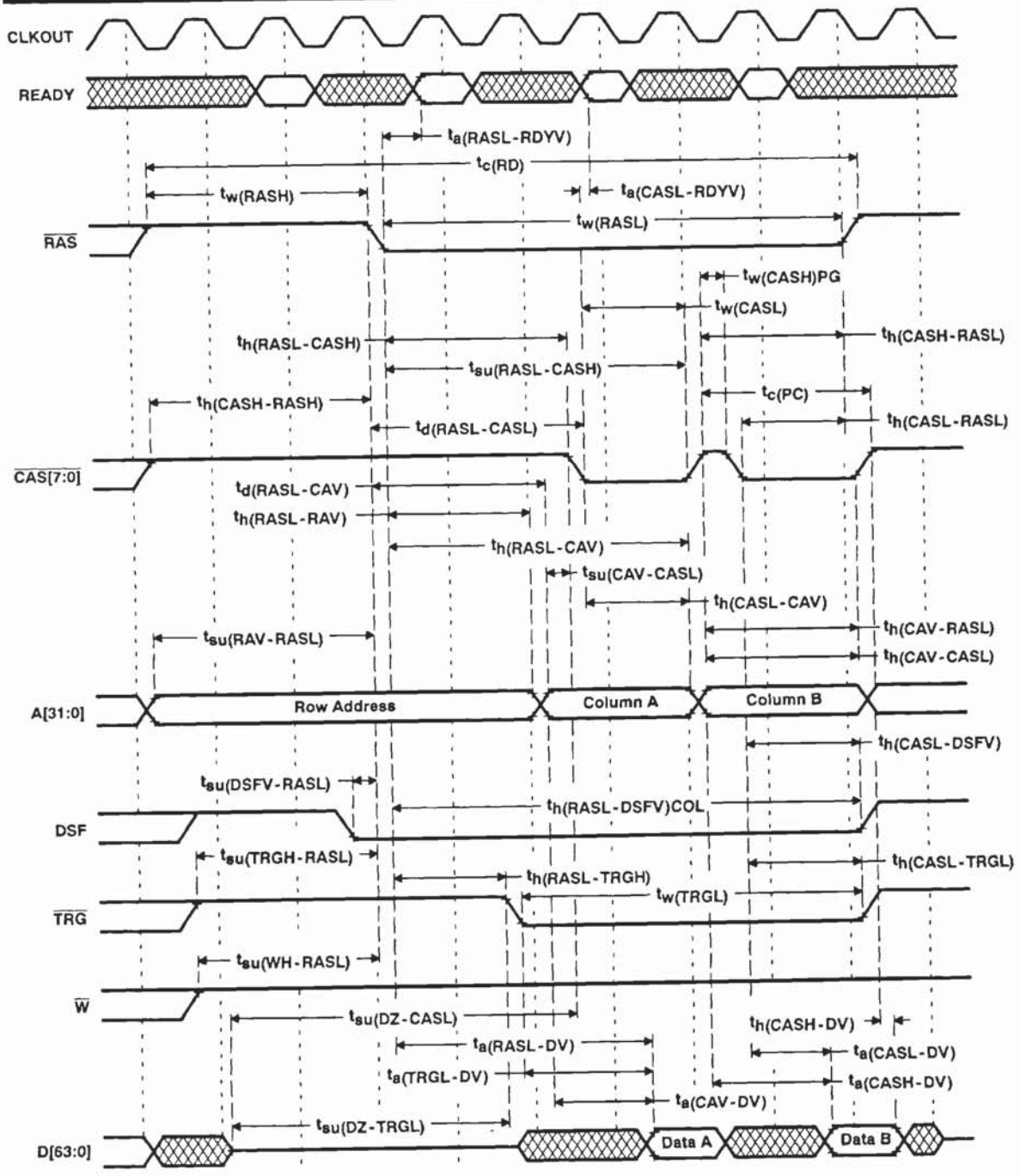
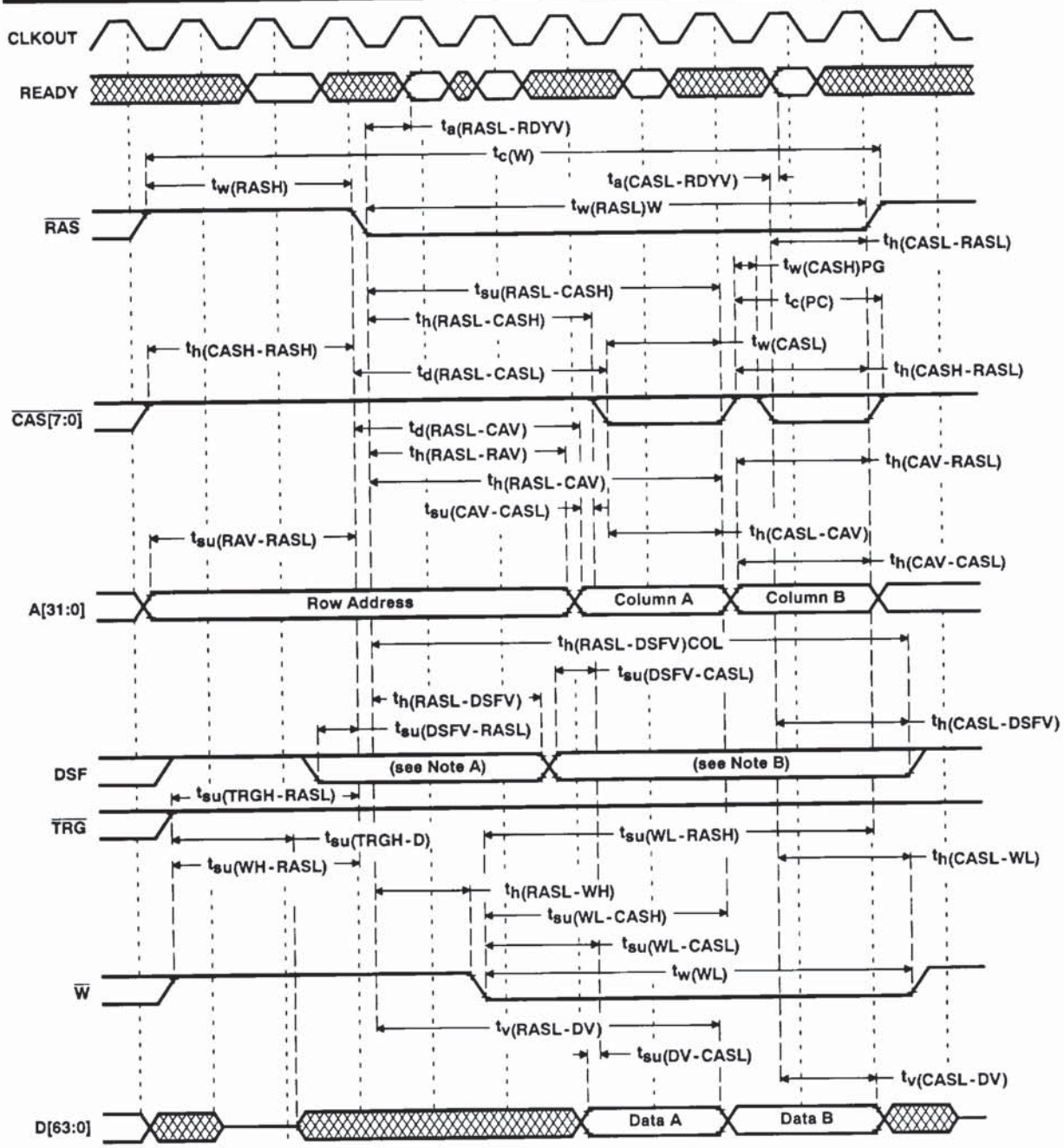


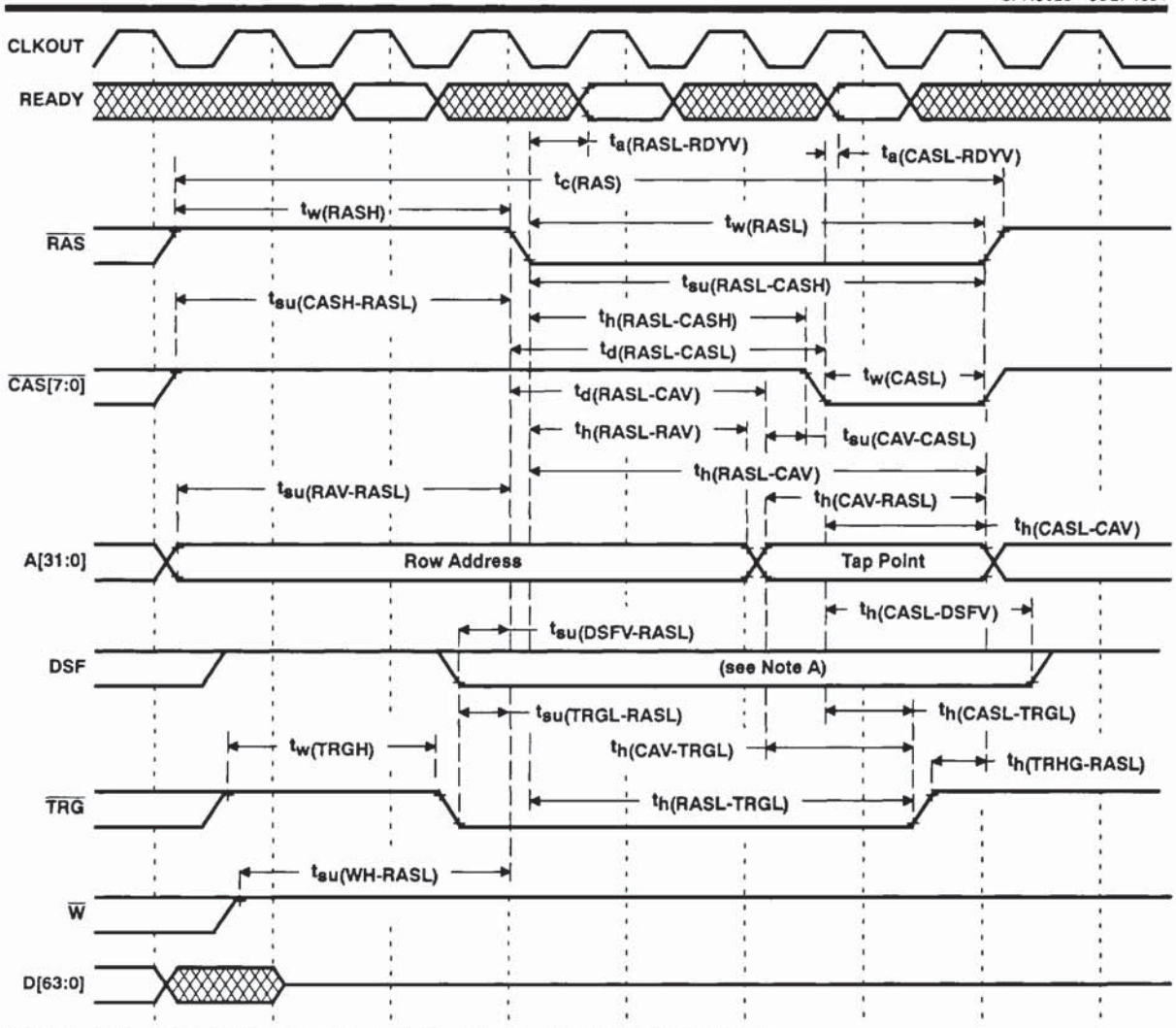
Figure 49. 2-Cycles/Column Read-Cycle Timing

ADVANCE INFORMATION



NOTES: A. DSF = 0 for a write or block-write cycle. DSF = 1 for a load-color-register cycle.
B. DSF = 0 for a write cycle. DSF = 1 for a block-write or load-color-register cycle.

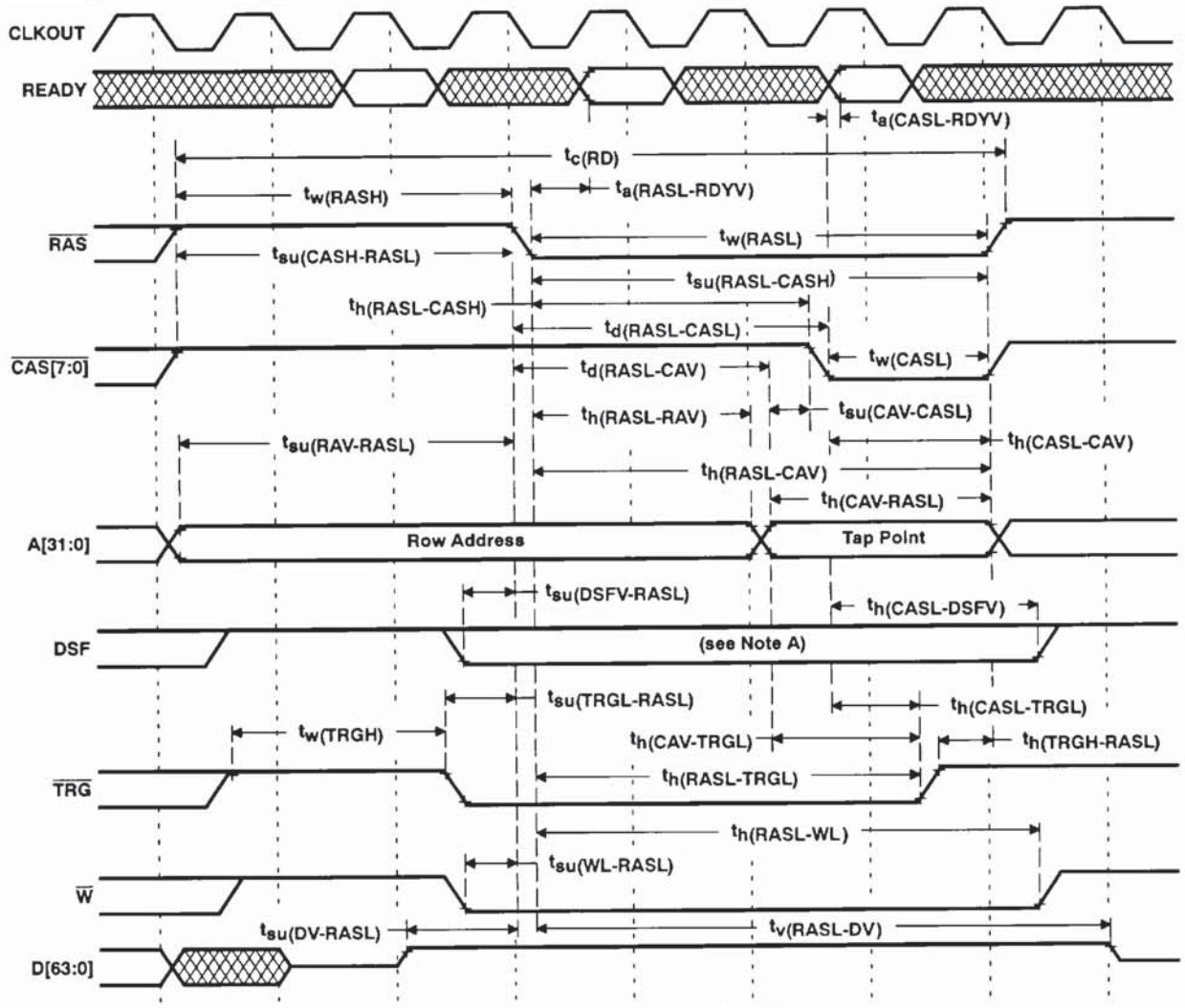
Figure 50. 2-Cycles/Column Write-, Block-Write-, and Load-Color-Register-Cycle Timing



NOTE A: DSF = 0 for a read-transfer cycle and DSF = 1 for a split-register read-transfer cycle.

Figure 51. 2-Cycles/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing

ADVANCE INFORMATION



NOTE A: DSF = 0 for a write-transfer cycle and DSF = 1 for a split-register write-transfer cycle.

Figure 52. 2-Cycles/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

PARAMETER MEASUREMENT INFORMATION

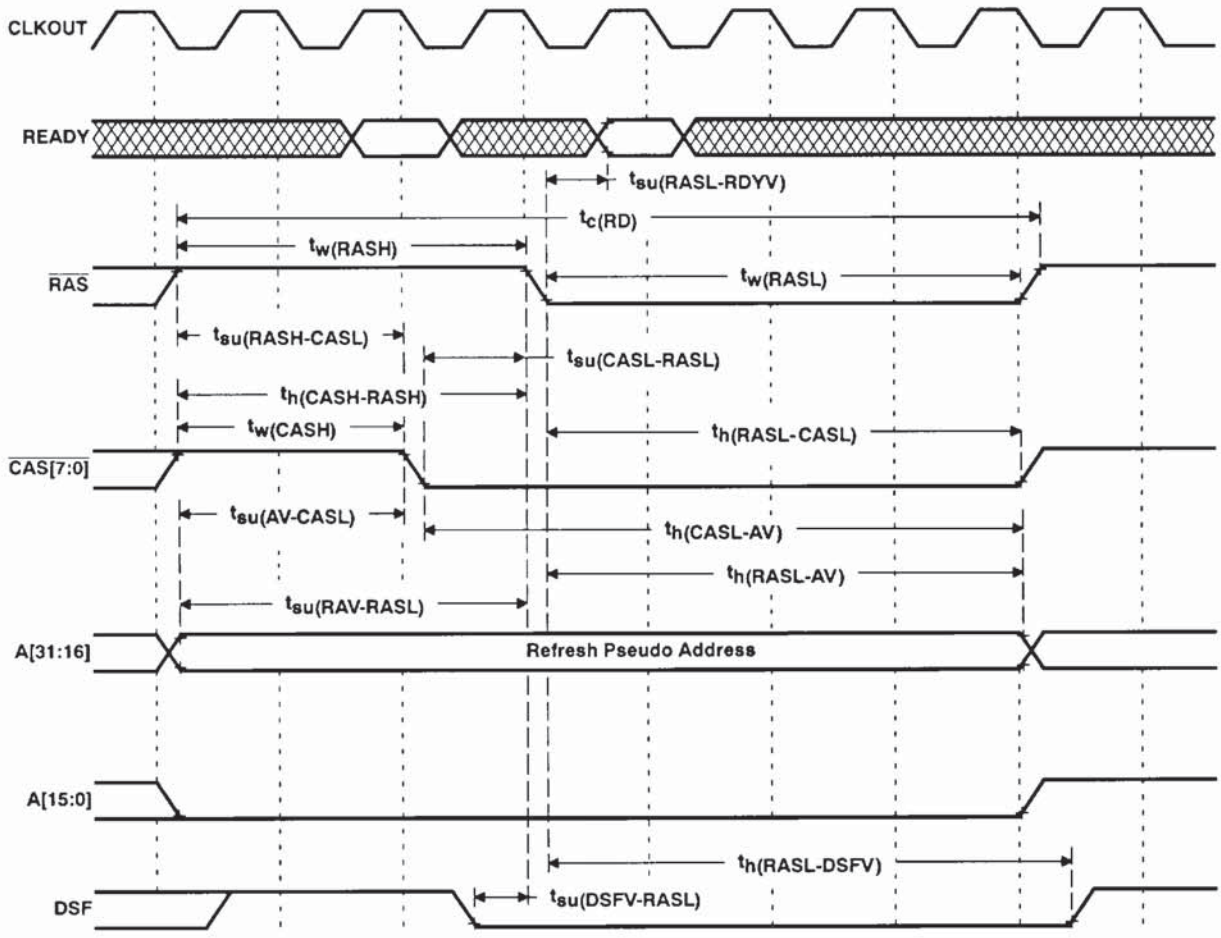


Figure 53. 2-Cycles/Column Refresh-Cycle Timing

ADVANCE INFORMATION

local-bus timing: memory control signals – nonpipelined 1-cycle/column

	PARAMETER	MIN	MAX	UNIT
$t_{a(CAV-DV)}$	Access time, D valid (in) from A (column address)		$3t_H - 10$	ns
$t_{a(CASL-DV)}$	Access time, D valid (in) from $\overline{CAS}$ low		$2t_H - 13$	ns
$t_{a(CASH-DV)}$	Access time, D valid (in) from $\overline{CAS}$ high (precharge)		$3t_H - 10$	ns
$t_{a(RASL-DV)}$	Access time, D valid (in) from $\overline{RAS}$ low		$5t_H - 10$	ns
$t_{a(TRGL-DV)}$	Access time, D valid (in) from $\overline{TRG}$ low		$4t_H - 10$	ns
$t_{c(RD)}$	Cycle time, read (see Note 4)	$12t_H - 6$		ns
$t_{c(W)}$	Cycle time, write (see Note 4)	$14t_H - 6$		ns
$t_{c(W)TR}$	Cycle time, write (transfer cycle)	$10t_H - 6$		ns
$t_{c(PC)}$	Cycle time, page-mode read or write	$2t_H - 6$		ns
$t_{w(RASL)}$	Pulse duration, $\overline{RAS}$ low (see Note 4)	$6t_H - 5$		ns
$t_{w(RASL)W}$	Pulse duration, $\overline{RAS}$ low, write cycle (see Note 4)	$8t_H - 5$		ns
$t_{w(RASL)TR}$	Pulse duration, $\overline{RAS}$ low, write (transfer cycle)	$4t_H - 5$		ns
$t_{w(RASH)}$	Pulse duration, $\overline{RAS}$ high	$6t_H - 6$		ns
$t_{w(CASL)}$	Pulse duration, $\overline{CAS}$ low	$t_H - 5$		ns
$t_{w(CASH)}$	Pulse duration, $\overline{CAS}$ high	$4t_H - 6$		ns
$t_{w(CASH)PG}$	Pulse duration, $\overline{CAS}$ high (page-mode cycle)	$t_H - 6$		ns
$t_{w(WL)}$	Pulse duration, $\overline{W}$ low (see Note 4)	$8t_H - 5$		ns
$t_{w(TRGL)}$	Pulse duration, $\overline{TRG}$ low (see Note 4)	$5t_H - 5$		ns
$t_{w(TRHG)}$	Pulse duration, $\overline{TRG}$ high	$4t_H - 6$		ns
$t_{su(CAV-CASL)}$	Setup time, A (column address) valid before $\overline{CAS}$ no longer high	$t_H - 7$		ns
$t_{su(CAV-CKOH)}$	Setup time, A (column address) valid before CLKOUT no longer low	$t_H - 7$		ns
$t_{su(RAV-RASL)}$	Setup time, A (row address) valid before $\overline{RAS}$ no longer high	$6t_H - 7$		ns
$t_{su(DV-CASL)}$	Setup time, D valid before $\overline{CAS}$ no longer high	$t_H - 7$		ns
$t_{su(DV-CKOH)}$	Setup time, D valid before CLKOUT no longer low	8		ns
$t_{su(WL-CASL)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer high	$6t_H - 7$		ns
$t_{su(WL-CASH)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer low	$7t_H - 7$		ns
$t_{su(WH-RASL)}$	Setup time, $\overline{W}$ high before $\overline{RAS}$ no longer high	$5t_H - 7$		ns
$t_{su(WL-RASH)}$	Setup time, $\overline{W}$ low before $\overline{RAS}$ no longer low (see Note 4)	$7t_H - 7$		ns
$t_{su(DSFV-RASL)}$	Setup time, DSF valid before $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(DSFV-CASL)}$	Setup time, DSF valid before $\overline{CAS}$ no longer high	$6t_H - 7$		ns
$t_{su(DV-RASL)}$	Setup time, D valid (high) before $\overline{RAS}$ no longer high (write-transfer cycle)	$2t_H - 7$		ns
$t_{su(TRGH-RASL)}$	Setup time, $\overline{TRG}$ high before $\overline{RAS}$ no longer high	$5t_H - 7$		ns
$t_{su(TRGL-RASL)}$	Setup time, $\overline{TRG}$ low to $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(RASL-CASH)}$	Setup time, $\overline{RAS}$ low to $\overline{CAS}$ no longer low (see Note 6)	$4t_H - 7$		ns
$t_{su(CASH-RASL)}$	Setup time, $\overline{CAS}$ high to $\overline{RAS}$ no longer high	$6t_H - 7$		ns
$t_{su(CASL-RASL)}$	Setup time, $\overline{CAS}$ low to $\overline{RAS}$ no longer high (refresh cycle)	$2t_H - 7$		ns
$t_{su(RASH-CASL)}$	Setup time, $\overline{RAS}$ high to $\overline{CAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns
$t_{su(DZ-CASL)}$	Setup time, D in high-impedance state to $\overline{CAS}$ no longer high	$7t_H - 7$		ns

NOTES: 4: Minimum value for this parameter occurs during cycles with a single column access.
6. During write cycles, the actual value of this parameter is increased by $4t_H$.

local-bus timing: memory control signals – nonpipelined 1-cycle/column (continued)

	PARAMETER	MIN	MAX	UNIT
$t_{su(DZ-TRGL)}$	Setup time, D in high-impedance state to $\overline{TRG}$ no longer high	$5t_H - 7$		ns
$t_{su(WL-RASL)}$	Setup time, $\overline{W}$ low to $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(TRGH-D)}$	Setup time, $\overline{TRG}$ high to data driven	$3t_H - 7$		ns
$t_{su(AV-CASL)}$	Setup time, A valid to $\overline{CAS}$ no longer high (refresh cycle)	$4t_H - 7$		ns
$t_{h(CASH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ high (precharge) (see Note 7)	$2t_H - 7$		ns
$t_{h(CASL-CAV)}$	Hold time, A (column address) valid after $\overline{CAS}$ low	$t_H - 7$		ns
$t_{h(RASL-RAV)}$	Hold time, A (row address) valid after $\overline{RAS}$ low (see Note 6)	$2t_H - 7$		ns
$t_{h(RASL-CAV)}$	Hold time, A (column address) valid after $\overline{RAS}$ low (see Note 6)	$4t_H - 7$		ns
$t_{h(RASL-DV)TR}$	Hold time, D valid (high) after $\overline{RAS}$ low (write) transfer cycle	$6t_H - 7$		ns
$t_{h(CKOH-DV)}$	Hold time, D valid after CLKOUT high	0		ns
$t_{h(CASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{CAS}$ high	$5t_H - 7$		ns
$t_{h(RASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ high	$5t_H - 7$		ns
$t_{h(CASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{CAS}$ low	$2t_H - 7$		ns
$t_{h(RASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{RAS}$ low (write transfer cycle)	$5t_H - 7$		ns
$t_{h(RASL-DSFV)}$	Hold time, DSF valid after $\overline{RAS}$ low	$t_H - 7$		ns
$t_{h(CASL-DSFV)}$	Hold time, DSF valid after $\overline{CAS}$ low (see Note 7)	$2t_H - 7$		ns
$t_{h(RASL-DSFV)COL}$	Hold time, column time DSF valid after $\overline{RAS}$ low (see Notes 4 and 5)	$7t_H - 7$		ns
$t_{h(RASL-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ low	$t_H - 7$		ns
$t_{h(RASL-TRGH)}$	Hold time, $\overline{TRG}$ high after $\overline{RAS}$ low	$t_H - 7$		ns
$t_{h(RASL-CASL)}$	Hold time, $\overline{CAS}$ low after $\overline{RAS}$ low (refresh cycle)	$4t_H - 7$		ns
$t_{h(CAV-RASL)}$	Hold time, $\overline{RAS}$ low after A (column address) valid (see Note 7)	$2t_H - 7$		ns
$t_{h(CAV-CASL)}$	Hold time, $\overline{CAS}$ low after A (column address) valid	$2t_H - 7$		ns
$t_{h(CASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (read cycle)	$4t_H - 7$		ns
$t_{h(CASL-TRG)TR}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (transfer cycle)	$2t_H - 7$		ns
$t_{h(CAV-TRGL)}$	Hold time, $\overline{TRG}$ low after A (column address) valid (transfer cycle)	$3t_H - 7$		ns
$t_{h(RASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{RAS}$ low (transfer cycle)	$5t_H - 7$		ns
$t_{h(RASL-CASH)}$	Hold time, $\overline{CAS}$ high after $\overline{RAS}$ low (see Note 6)	$3t_H - 7$		ns
$t_{h(TRGH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{TRG}$ high (transfer cycle)	$t_H - 7$		ns
$t_{h(RASH-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{RAS}$ high (write transfer cycle)	$t_H - 7$		ns
$t_{h(CASH-DV)}$	Hold time, D valid (in) after $\overline{CAS}$ high	$t_H + 5$		ns
$t_{h(CASL-AV)}$	Hold time, A valid after $\overline{CAS}$ low (refresh cycle)	$6t_H - 7$		ns
$t_{h(RASL-AV)}$	Hold time, A valid after $\overline{RAS}$ low (refresh cycle)	$4t_H - 7$		ns
$t_{h(CASL-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ low (see Note 7)	$3t_H - 7$		ns
$t_{v(CASL-DV)}$	Valid time, D valid after $\overline{CAS}$ low	$t_H - 7$		ns
$t_{v(RASL-DV)}$	Valid time, D valid after $\overline{RAS}$ low	$8t_H - 7$		ns
$t_{d(RASL-CASL)}$	Delay time, $\overline{RAS}$ no longer high to $\overline{CAS}$ low (see Note 6)		$3t_H + 7$	ns

NOTES: 4. Minimum value for this parameter occurs during cycles with a single column access.
5. During write cycles, the actual value of this parameter is increased by $2t_H$.
6. During write cycles, the actual value of this parameter is increased by $4t_H$.
7. During read, read transfer, and split read transfer cycles, the actual value of this paramter is increased by $2t_H$.

ADVANCE INFORMATION

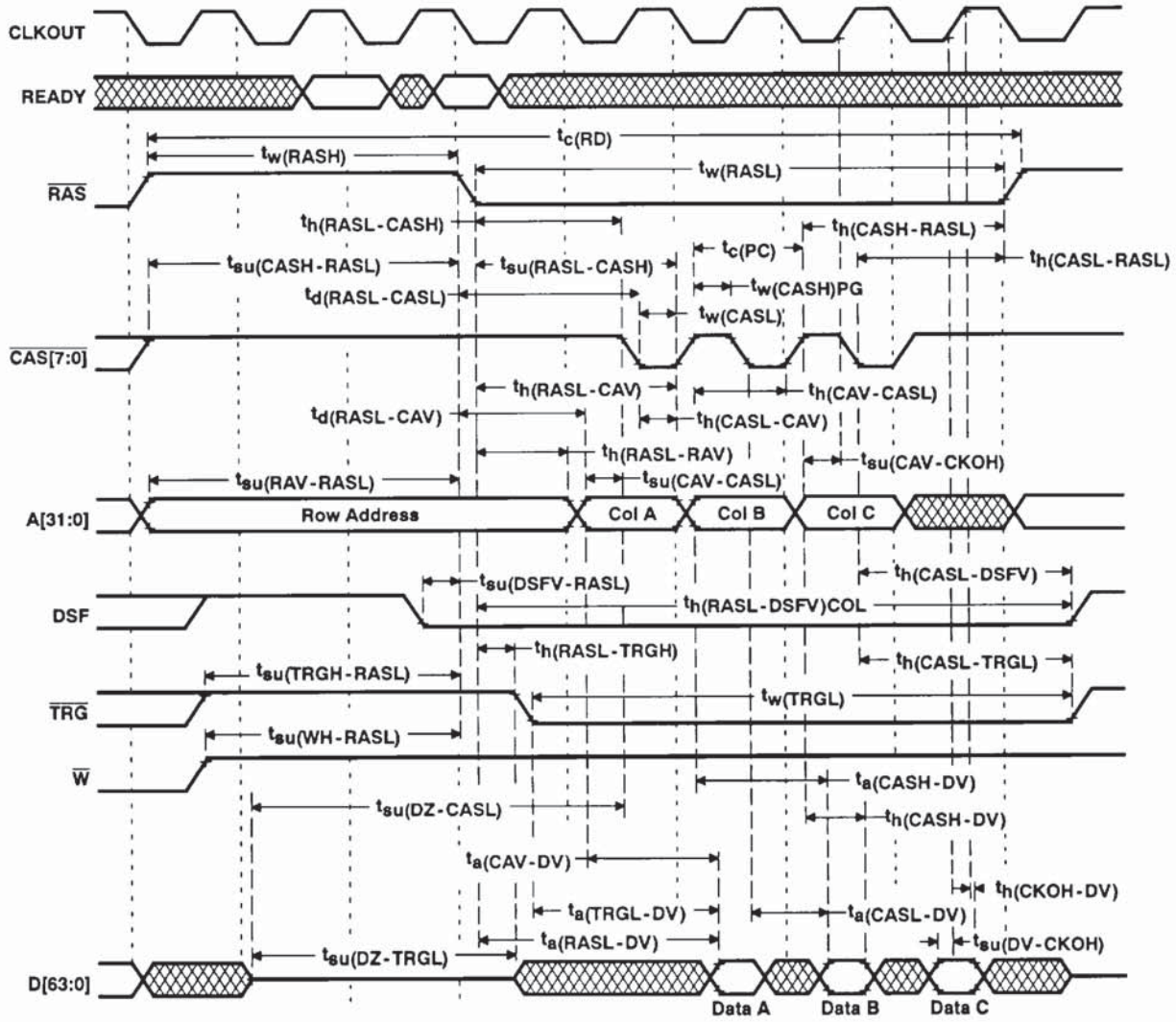
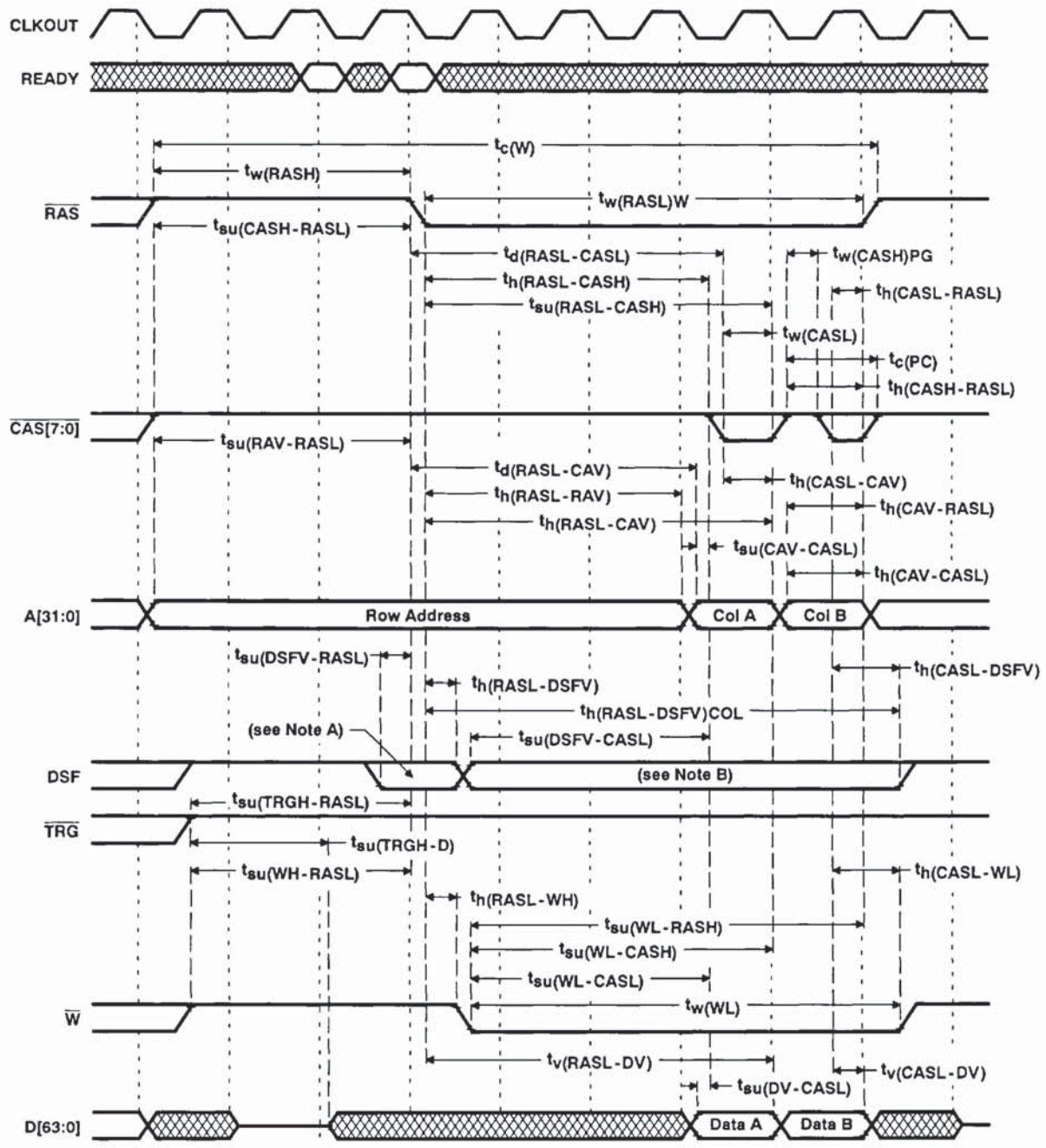


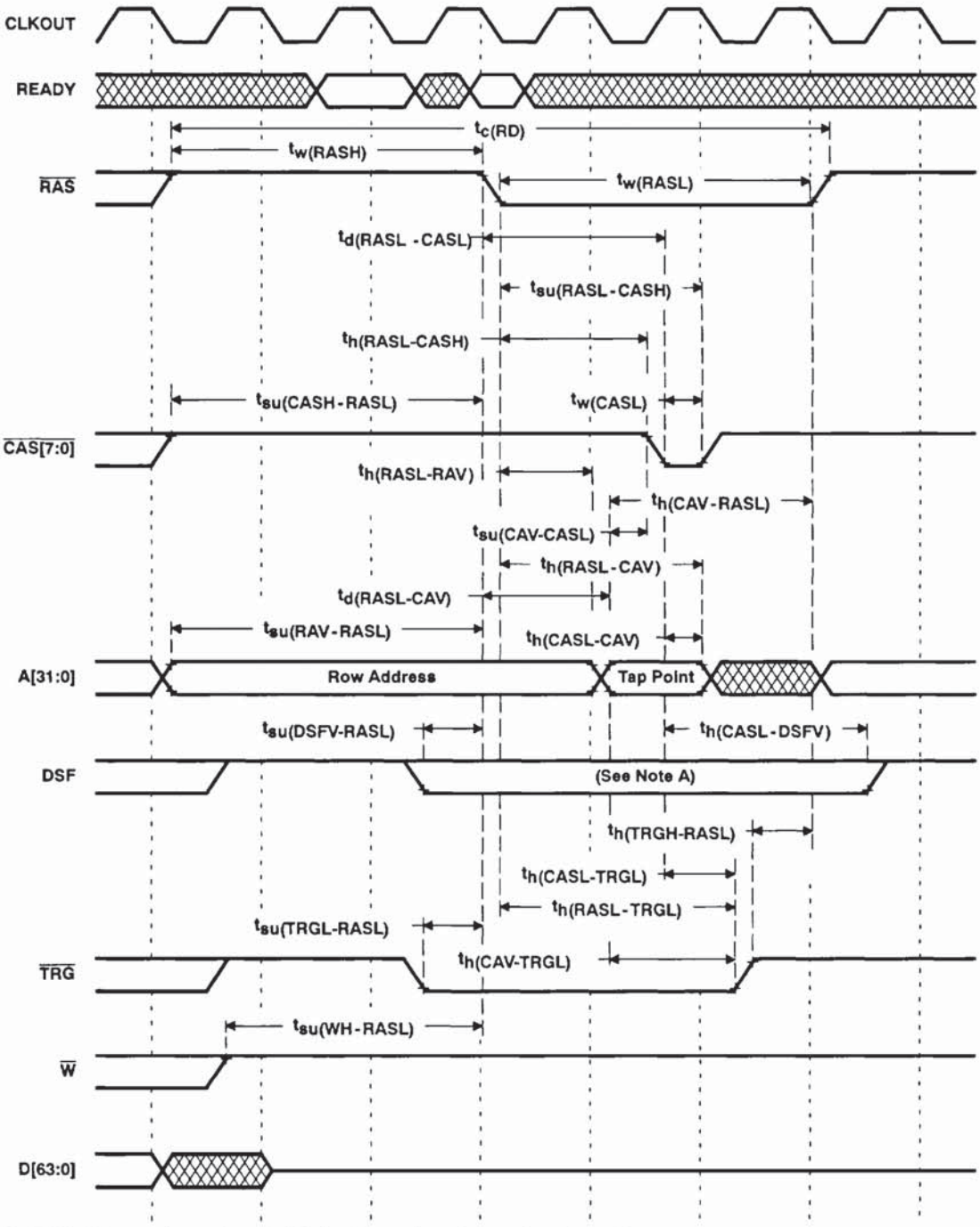
Figure 54. Nonpipelined 1-Cycle/Column Read-Cycle Timing



NOTES: A. DSF = 0 for a write or block-write cycle. DSF = 1 for a load-color-register cycle.
B. DSF = 0 for a write cycle. DSF = 1 for a block-write or load-color-register cycle.

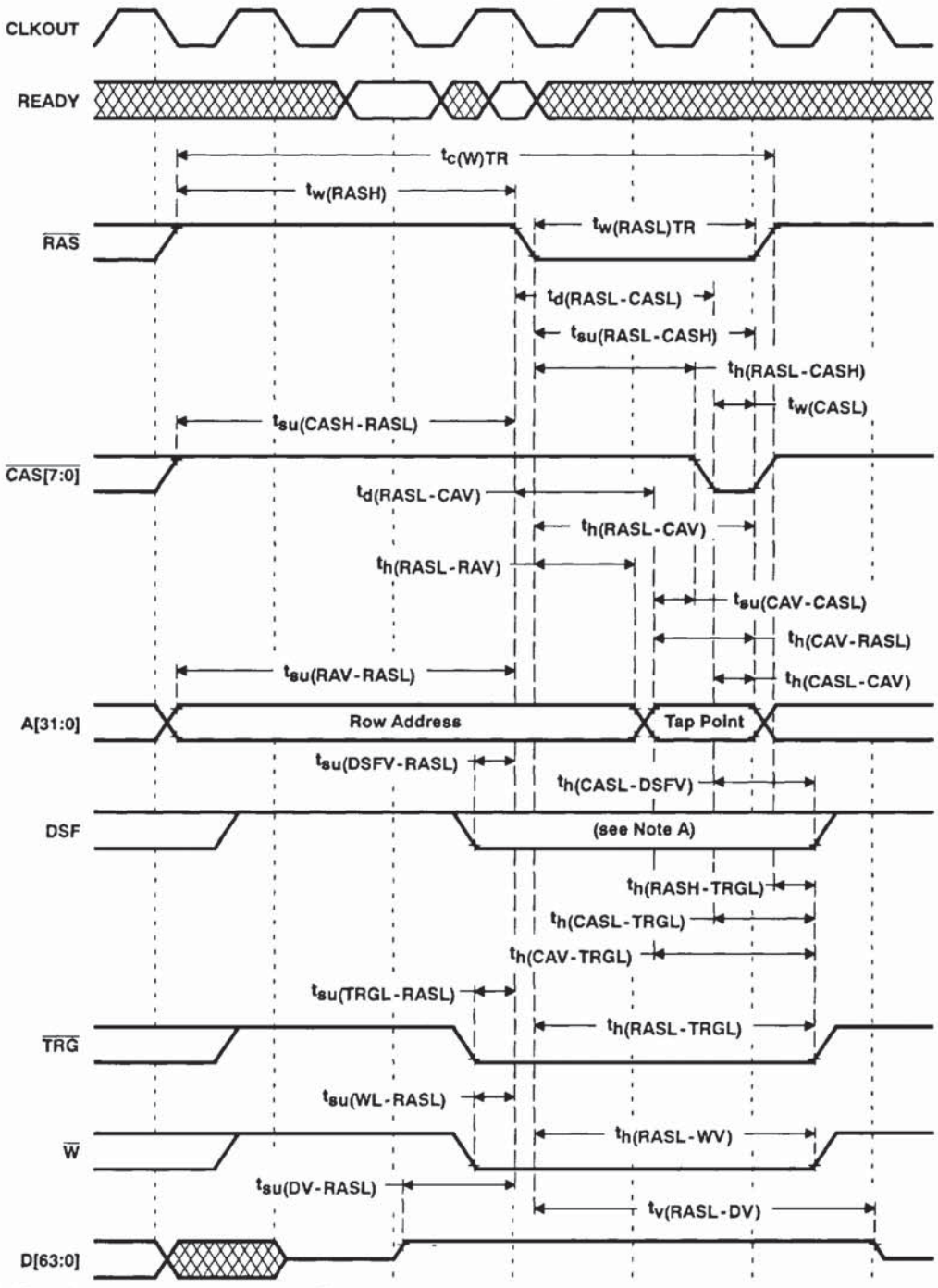
Figure 55. Nonpipelined 1-Cycle/Column Write-, Block-Write-, and Load-Color-Register-Cycle Timing

ADVANCE INFORMATION



NOTE A: DSF = 0 for a read-transfer cycle. DSF = 1 for a split-register read-transfer cycle.

Figure 56. Nonpipelined 1-Cycle/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing



NOTE A: DSF = 0 for a write-transfer cycle and DSF = 1 for a split-register write-transfer cycle.

Figure 57. Nonpipelined 1-Cycle/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

ADVANCE INFORMATION

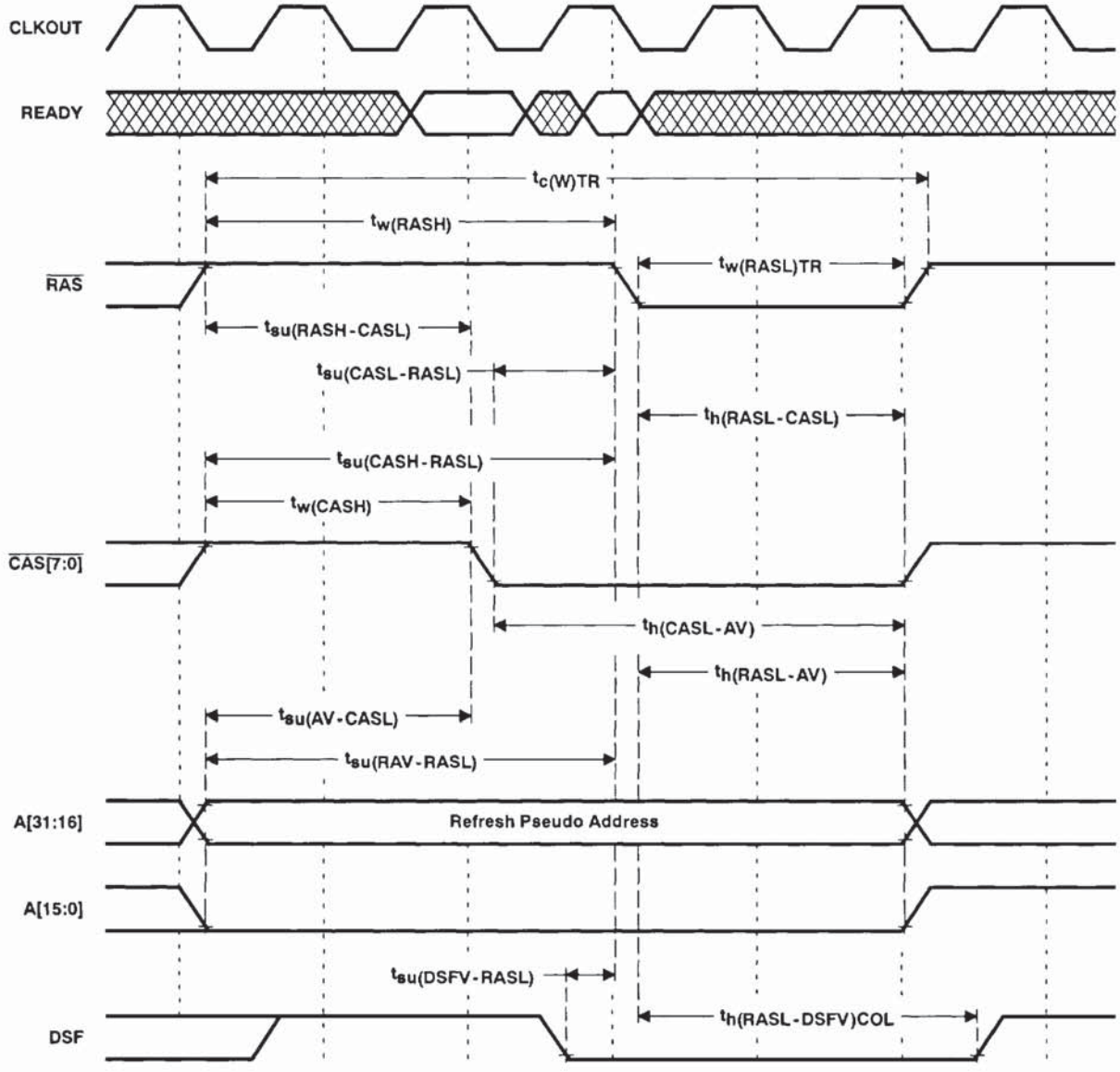


Figure 58. Nonpipelined 1-Cycle/Column Refresh-Cycle Timing

local-bus timing: memory control signals - pipelined 1-cycle/column

PARAMETER		MIN	MAX	UNIT
$t_{a(CASL-DV)1}$	Access time, D valid (in) from $\overline{CAS}$ low		$2t_H - 10$	ns
$t_{a(CASL-DV)2}$	Access time, D valid (in) from $\overline{CAS}$ low		$4t_H - 10$	ns
$t_{a(CASH-DV)}$	Access time, D valid (in) from $\overline{CAS}$ high (precharge)		$3t_H - 10$	ns
$t_{a(RASL-DV)}$	Access time, D valid (in) from $\overline{RAS}$ low		$7t_H - 10$	ns
$t_{a(TRGL-DV)}$	Access time, D valid (in) from $\overline{TRG}$ low		$6t_H - 10$	ns
$t_{c(RD)}$	Cycle time, read (see Note 4)	$14t_H - 6$		ns
$t_{c(W)}$	Cycle time, write (see Note 4)	$16t_H - 6$		ns
$t_{c(W)TR}$	Cycle time, write (transfer cycle)	$12t_H - 6$		ns
$t_{c(PC)}$	Cycle time, page-mode read or write	$2t_H - 6$		ns
$t_{w(RASL)}$	Pulse duration, $\overline{RAS}$ low (see Note 4)	$8t_H - 5$		ns
$t_{w(RASL)W}$	Pulse duration, $\overline{RAS}$ low, write cycle (see Note 4)	$10t_H - 5$		ns
$t_{w(RASL)TR}$	Pulse duration, $\overline{RAS}$ low, write (transfer cycle)	$6t_H - 5$		ns
$t_{w(RASH)}$	Pulse duration, $\overline{RAS}$ high	$6t_H - 6$		ns
$t_{w(CASL)}$	Pulse duration, $\overline{CAS}$ low	$t_H - 5$		ns
$t_{w(CASH)}$	Pulse duration, $\overline{CAS}$ high	$4t_H - 6$		ns
$t_{w(CASH)PG}$	Pulse duration, $\overline{CAS}$ high (page-mode cycle)	$t_H - 6$		ns
$t_{w(TRGH)}$	Pulse duration, $\overline{TRG}$ high	$4t_H - 6$		ns
$t_{su(CAV-CASL)}$	Setup time, A (column address) valid before $\overline{CAS}$ no longer high	$t_H - 7$		ns
$t_{su(CAV-CKOH)}$	Setup time, A (column address) valid before CLKOUT no longer low	$t_H - 7$		ns
$t_{su(RAV-RASL)}$	Setup time, A (row address) valid before $\overline{RAS}$ no longer high	$6t_H - 7$		ns
$t_{su(DV-CASL)}$	Setup time, D valid before $\overline{CAS}$ no longer high	$t_H - 7$		ns
$t_{su(DV-CKOH)}$	Setup time, D valid before CLKOUT no longer low	8		ns
$t_{su(WL-CASL)}$	Setup time, $\overline{W}$ low before $\overline{CAS}$ no longer high	$6t_H - 7$		ns
$t_{su(WH-RASL)}$	Setup time, $\overline{W}$ high before $\overline{RAS}$ no longer high	$5t_H - 7$		ns
$t_{su(DSFV-RASL)}$	Setup time, DSF valid before $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(DSFV-CASL)}$	Setup time, DSF valid before $\overline{CAS}$ no longer high	$6t_H - 7$		ns
$t_{su(DV-RASL)}$	Setup time, D valid (high) before $\overline{RAS}$ no longer high (write-transfer cycle)	$2t_H - 7$		ns
$t_{su(TRGH-RASL)}$	Setup time, $\overline{TRG}$ high before $\overline{RAS}$ no longer high	$5t_H - 7$		ns
$t_{su(TRGL-RASL)}$	Setup time, $\overline{TRG}$ low to $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(RASL-CASH)}$	Setup time, $\overline{RAS}$ low to $\overline{CAS}$ no longer low (see Note 6)	$4t_H - 7$		ns
$t_{su(CASH-RASL)}$	Setup time, $\overline{CAS}$ high to $\overline{RAS}$ no longer high	$6t_H - 7$		ns
$t_{su(DZ-CASL)}$	Setup time, D in high-impedance state to $\overline{CAS}$ no longer high	$7t_H - 7$		ns
$t_{su(DZ-TRGL)}$	Setup time, D in high-impedance state to $\overline{TRG}$ no longer high	$5t_H - 7$		ns
$t_{su(WEL-RASL)}$	Setup time, $\overline{W}$ low to $\overline{RAS}$ no longer high	$t_H - 7$		ns
$t_{su(TRGH-D)}$	Setup time, $\overline{TRG}$ high to data driven	$3t_H - 7$		ns
$t_{h(CASL-CAV)}$	Hold time, A (column address) valid after $\overline{CAS}$ low	$t_H - 7$		ns
$t_{h(RASL-RAV)}$	Hold time, A (row address) valid after $\overline{RAS}$ low (see Note 6)	$2t_H - 7$		ns

NOTES: 4: Minimum value for this parameter occurs during cycles with a single column access.
6. During write cycles, the actual value of this parameter is increased by $4t_H$.

ADVANCE INFORMATION

local-bus timing: memory control signals – pipelined 1-cycle/column (continued)

	PARAMETER	MIN	MAX	UNIT
$t_{h(CKOH-DV)}$	Hold time, D (in) valid after CLKOUT high	0		ns
$t_{h(CASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{CAS}$ high	$5t_H - 7$		ns
$t_{h(RASH-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ high	$5t_H - 7$		ns
$t_{h(CASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{CAS}$ low	$2t_H - 7$		ns
$t_{h(RASL-WL)}$	Hold time, $\overline{W}$ low after $\overline{RAS}$ low (write transfer cycle)	$7t_H - 7$		ns
$t_{h(RASL-DSFV)}$	Hold time, DSF valid after $\overline{RAS}$ low	$t_H - 7$		ns
$t_{h(CASL-DSFV)}$	Hold time, DSF valid after $\overline{CAS}$ low (see Note 7)	$2t_H - 7$		ns
$t_{h(RASL-WH)}$	Hold time, $\overline{W}$ high after $\overline{RAS}$ low	$t_H - 7$		ns
$t_{h(RASL-TRGH)}$	Hold time, $\overline{TRG}$ high after $\overline{RAS}$ low	$t_H - 7$		ns
$t_{h(CASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (read cycle)	$4t_H - 7$		ns
$t_{h(CASL-TRG)TR}$	Hold time, $\overline{TRG}$ low after $\overline{CAS}$ low (transfer cycle)	$2t_H - 7$		ns
$t_{h(CASL-TRGL)}$	Hold time, $\overline{TRG}$ low after A (column address) valid (transfer cycle)	$3t_H - 7$		ns
$t_{h(RASL-TRGL)}$	Hold time, $\overline{TRG}$ low after $\overline{RAS}$ low (transfer cycle)	$5t_H - 7$		ns
$t_{h(RASL-CASH)}$	Hold time, $\overline{CAS}$ high after $\overline{RAS}$ low (see Note 6)	$3t_H - 7$		ns
$t_{h(TRGH-RASL)}$	Hold time, $\overline{RAS}$ low after $\overline{TRG}$ high (transfer cycle) (see Note 7)	$t_H - 7$		ns
$t_{h(CASH-DV)}$	Hold time, D (in) valid after $\overline{CAS}$ high	0		ns
$t_{h(CASL-RASL)1}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ low	$t_H - 7$		ns
$t_{h(CASL-RASL)2}$	Hold time, $\overline{RAS}$ low after $\overline{CAS}$ low	$3t_H - 7$		ns
$t_{d(RASL-CASL)}$	Delay time, $\overline{RAS}$ no longer high to $\overline{CAS}$ low (see Note 6)		$3t_H + 7$	ns
$t_{v(CASL-DV)}$	Valid time, D (out) valid after $\overline{CAS}$ low	$t_H - 7$		ns
$t_{v(RASL-DV)}$	Valid time, D (out) valid after $\overline{RAS}$ low	$8t_H - 7$		ns

NOTES: 6. During write cycles, the actual value of this parameter is increased by $4t_H$.
7. During read, read-transfer, and split-register read-transfer cycles, the actual value of this paramter is increased by $2t_H$.

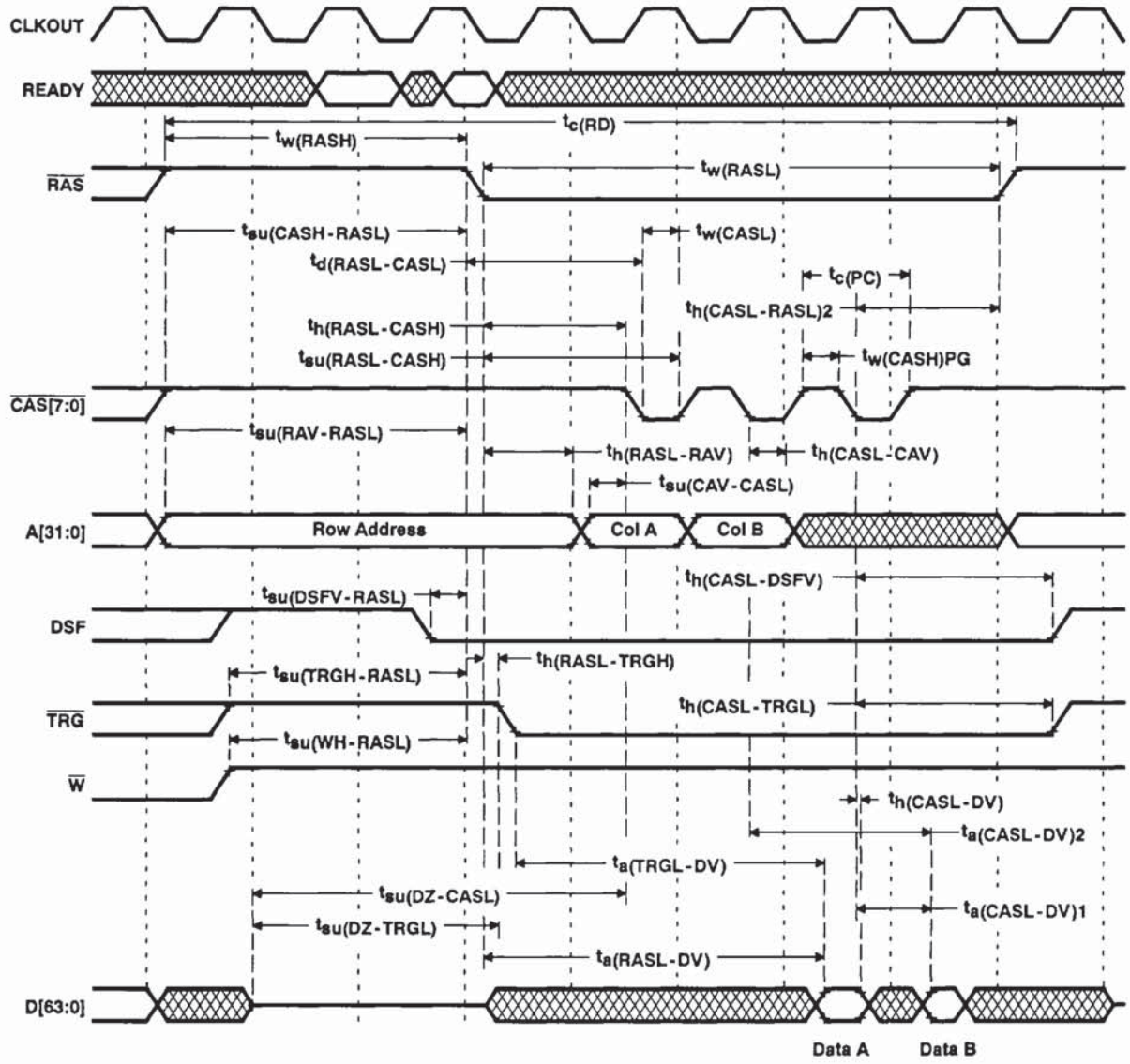
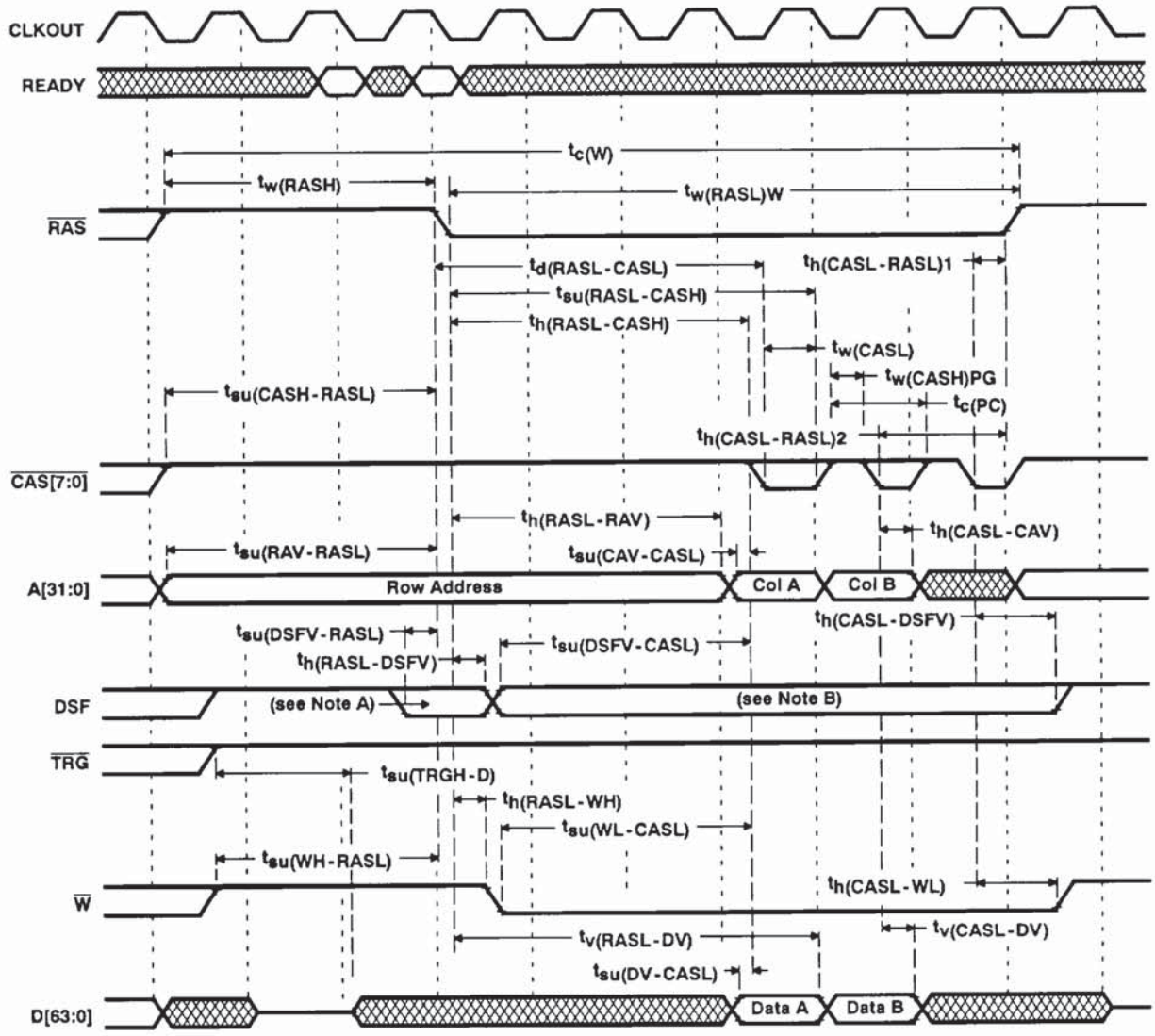


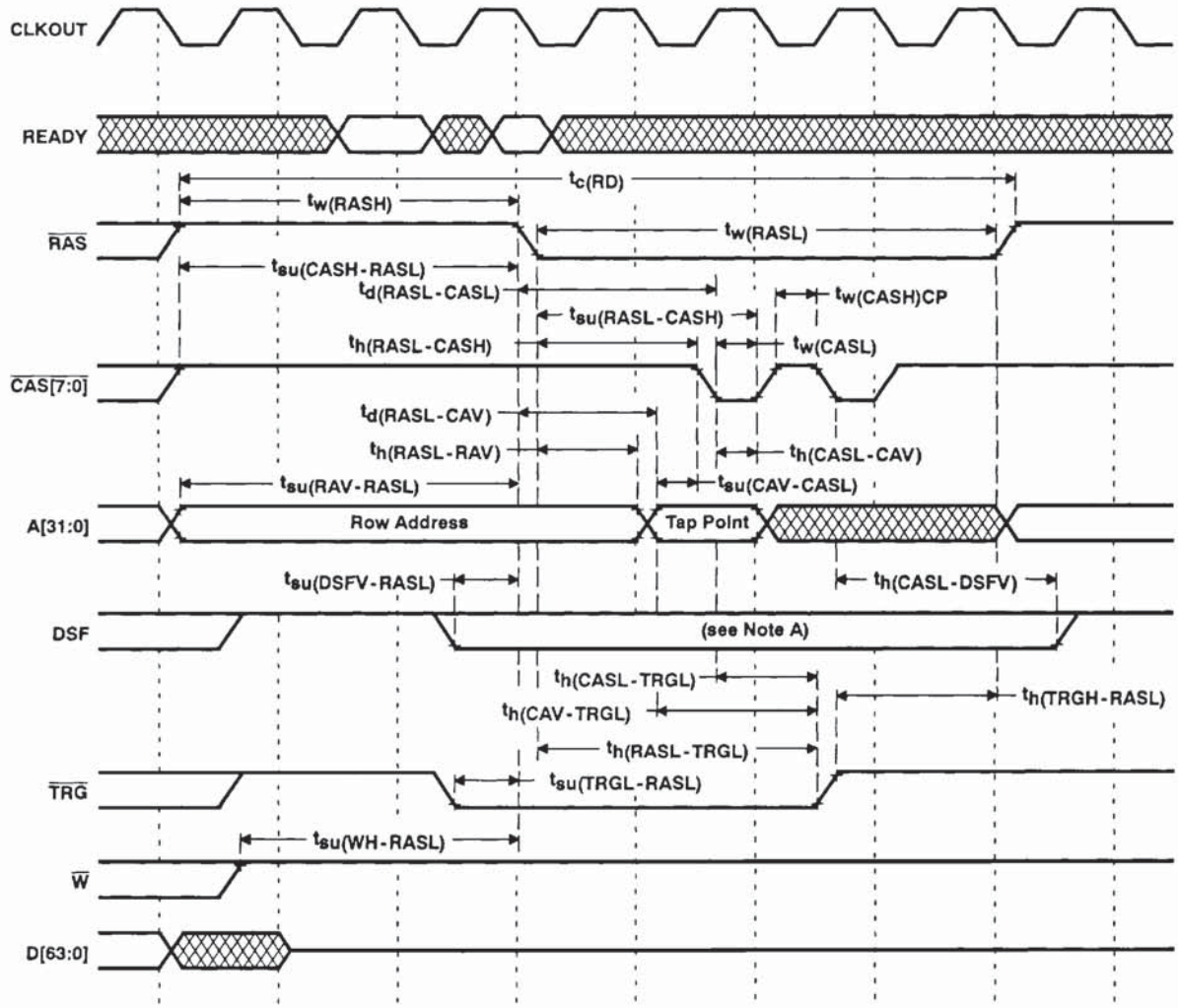
Figure 59. Pipelined 1-Cycle/Column Read-Cycle Timing

ADVANCE INFORMATION



NOTES: A. DSF = 0 for a write or block-write cycle. DSF = 1 for a load-color-register cycle.
B. DSF = 0 for a write cycle. DSF = 1 for a block-write or load-color-register cycle.

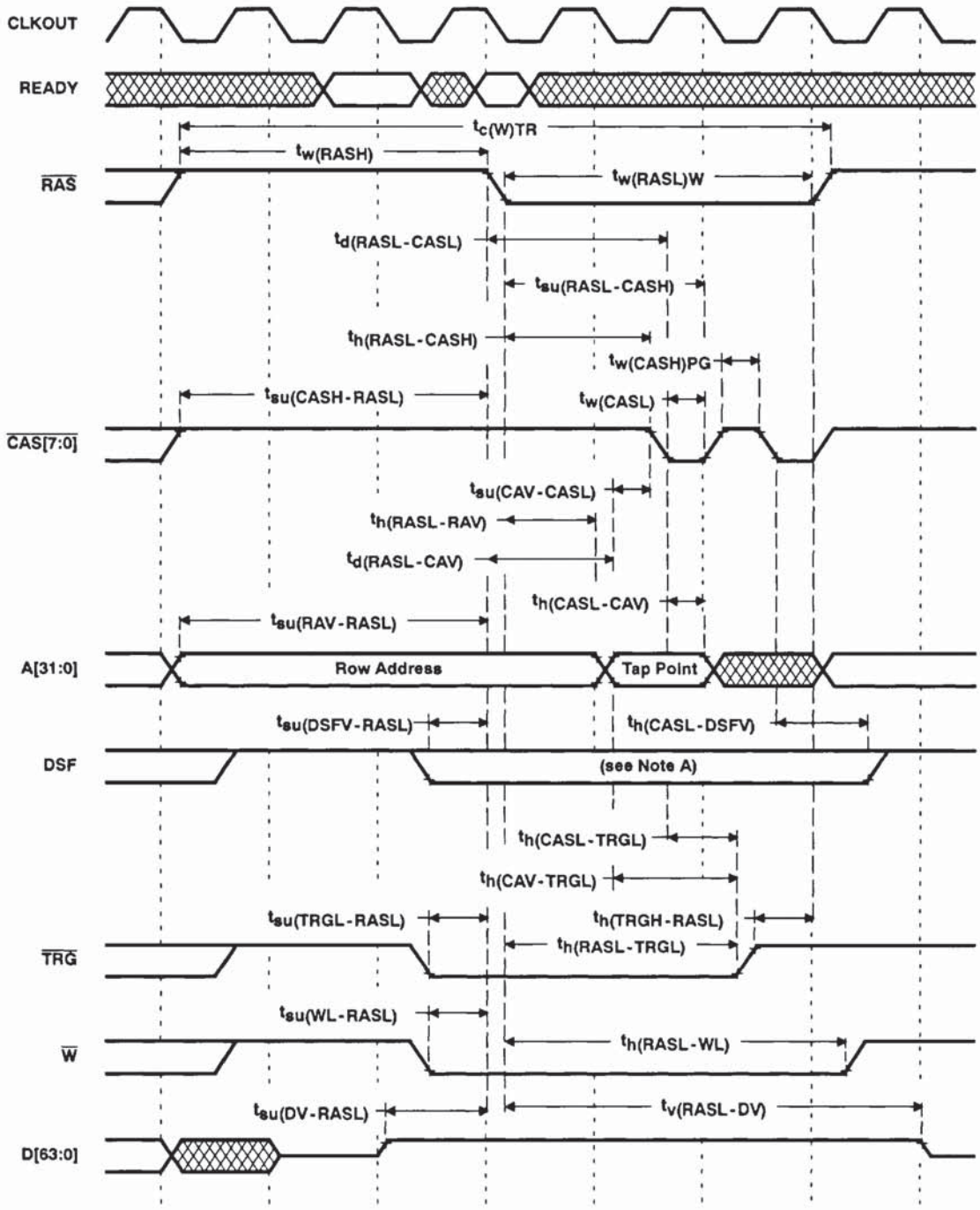
Figure 60. Pipelined 1-Cycle/Column Write-, Block-Write- and Load-Color-Register-Cycle Timing



NOTE A: DSF = 0 for a read-transfer cycle. DSF = 1 for a split-register read-transfer cycle.

Figure 61. Pipelined 1-Cycle/Column Read-Transfer and Split-Register Read-Transfer-Cycle Timing

ADVANCE INFORMATION



NOTE A: DSF = 0 for a write-transfer cycle. DSF = 1 for a split-register write-transfer cycle.

Figure 62. Pipelined 1-Cycle/Column Write-Transfer and Split-Register Write-Transfer-Cycle Timing

host-interface timing

		MIN	MAX	UNIT
$t_{a(REQV-HRQL)}$	Access time, REQ1–REQ0 valid to $\overline{HREQ}$ low (see Note 8)		$3t_H - 15$	ns
$t_{h(CKOH-REQV)}$	Hold time, REQ1–REQ0 valid after CLKOUT high	$t_H - 7$		ns
$t_{h(HRQL-HAKL)}$	Hold time for $\overline{HACK}$ high after $\overline{HREQ}$ goes low	$4t_H - 12$		ns
$t_{d(HRQH-HAKH)}$	Delay time, $\overline{HREQ}$ high to $\overline{HACK}$ no longer low		15	ns
$t_{d(HRQH-MCTD)}$	Delay time, $\overline{HREQ}$ high to memory control signals driven	20		ns
$t_{su(REQV-CKOH)}$	Setup time, REQ1–REQ0 valid to CLKOUT no longer low	$t_H - 7$		ns
$t_{su(HRQL-CKOH)}$	Setup time, $\overline{HREQ}$ low to CLKOUT no longer low (see Note 8)	8		ns
$t_{su(MCTZ-HAKL)}$	Setup time, memory-control signals hi-Z before $\overline{HACK}$ no longer high	0		ns

NOTE 8: Parameter must be met only to ensure $\overline{HREQ}$ recognition during the indicated clock cycle.

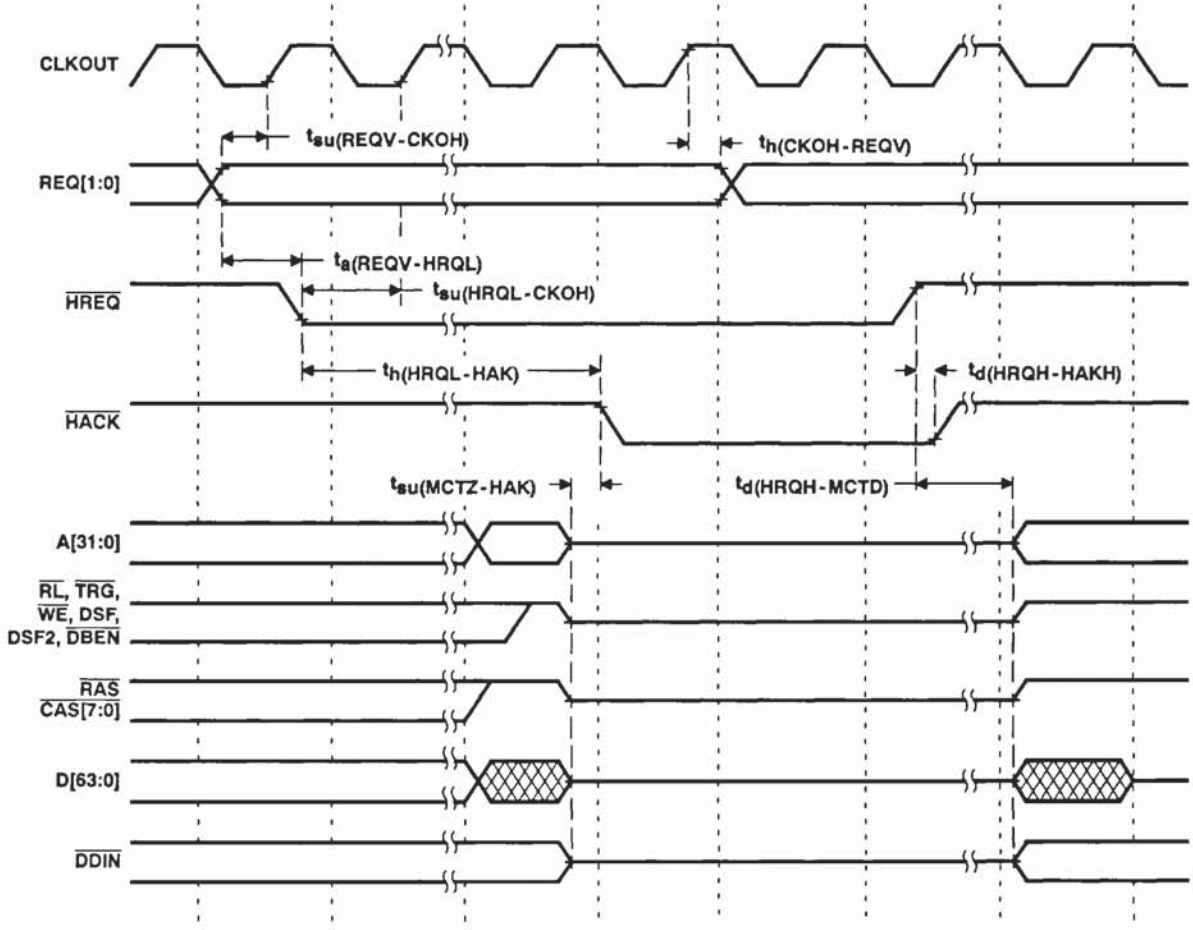


Figure 63. Host-Interface Timing

ADVANCE INFORMATION

video-interface timing: SCLK timing

		MIN	MAX	UNIT
$t_{c(SCK)}$	SCLK period	16		ns
$t_{w(SCKH)}$	Pulse duration, SCLK high	5		ns
$t_{w(SCKL)}$	Pulse duration, SCLK low	5		ns
$t_{t(SCK)}$	Transition time, SCLK (rise and fall)	3		ns

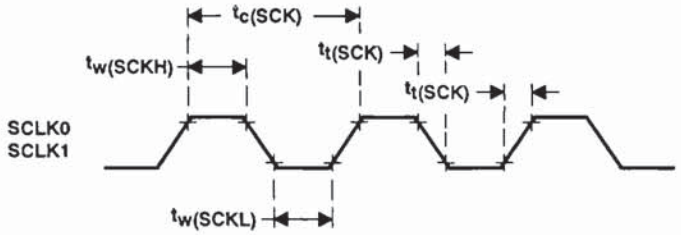


Figure 64. Video-Interface Timing: SCLK Timing

video-interface timing: FCLK input and video outputs

		MIN	MAX	UNIT
$t_{c(FCK)}$	FCLK period	25		ns
$t_{w(FCKH)}$	Pulse duration, FCLK high	8		ns
$t_{w(FCKL)}$	Pulse duration, FCLK low	8		ns
$t_{t(FCK)}$	Transition time, FCLK (rise and fall)	3		ns
$t_{d(FCKL-SYL)}$	Delay time, FCLK no longer high to $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$/$\overline{HBLNK}$, $\overline{CBLNK}$/$\overline{VBLNK}$, or CAREA low		20	ns
$t_{d(FCKL-SYH)}$	Delay time, FCLK no longer high to $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$/$\overline{HBLNK}$, $\overline{CBLNK}$/$\overline{VBLNK}$, or CAREA high		20	ns
$t_{h(FCKL-SYH)}$	Hold time, $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$/$\overline{HBLNK}$, $\overline{CBLNK}$/$\overline{VBLNK}$, or CAREA high after FCLK low	0		ns
$t_{h(FCKL-SYL)}$	Hold time, $\overline{HSYNC}$, $\overline{VSYNC}$, $\overline{CSYNC}$/$\overline{HBLNK}$, $\overline{CBLNK}$/$\overline{VBLNK}$, or CAREA low after FCLK low	0		ns

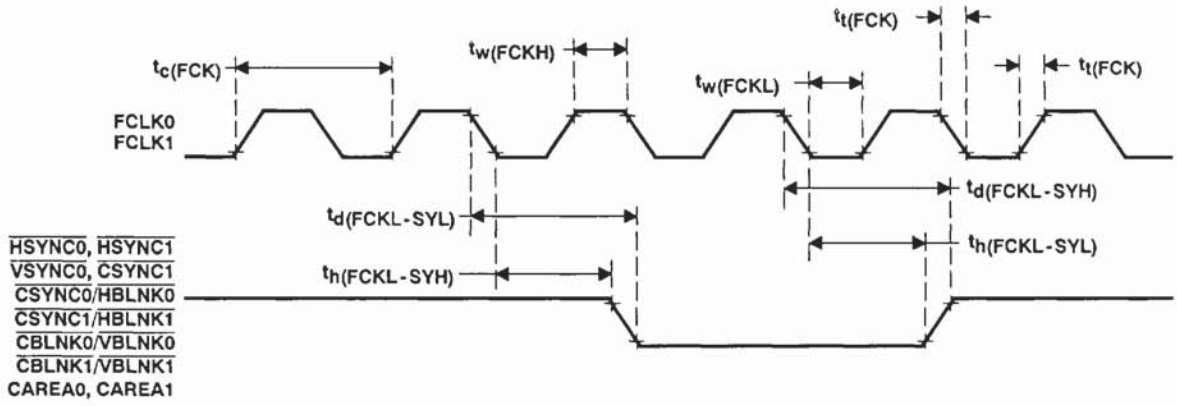


Figure 65. Video-Interface Timing: FCLK Input and Video Outputs

video-interface timing: external sync inputs

		MIN	MAX	UNIT
$t_{su(SIV-FCKH)}$	Setup time, $\overline{HSYNC}$, $\overline{VSYNC}$, or $\overline{CSYNC}$ valid to FCLK no longer low	10		ns
$t_{h(SIV-FCKH)}$	Hold time, $\overline{HSYNC}$, $\overline{VSYNC}$, or $\overline{CSYNC}$ valid after FCLK high	2		ns

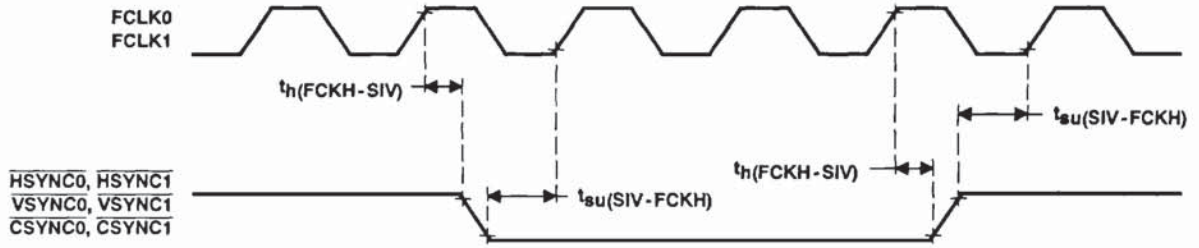
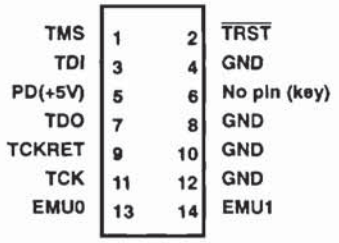


Figure 66. Video-Interface Timing: External Sync Inputs

ADVANCE INFORMATION

emulator interface connection

The 'C80 supports emulation through a dedicated emulation port that is a superset of the IEEE-1149.1 (JTAG) standard. To support the 'C80 emulator, a target system must include a 14-pin header (2 rows of 7 pins) with the connections shown below.



XDS510 SIGNAL	XDS510 STATE	TARGET STATE	DESCRIPTION
TMS	O	I	IEEE-1149.1 test mode select
TDI	O	I	IEEE-1149.1 test data input
TDO	I	O	IEEE-1149.1 test data output
TCK	O	I	IEEE-1149.1 test clock – 10 MHz clock source from emulator. Can be used to drive system test clock.
$\overline{\text{TRST}}$	O	I	IEEE-1149.1 test reset
EMU0	I	I/O	Emulation pin 0
EMU1	I	I/O	Emulation pin 1
PD (5 V)	I	O	Presence detect. Indicates that the target is connected and powered-up. Should be tied to + 5 V on target system
TCKRET	I	O	IEEE-1149.1 test clock return. Test clock input to the XDS510 emulator. Can be buffered or unbuffered version of TCK.

For best results, the emulation header should be located as close as possible to the 'C80. If the distance exceeds 6 inches, the emulation signals should be buffered.

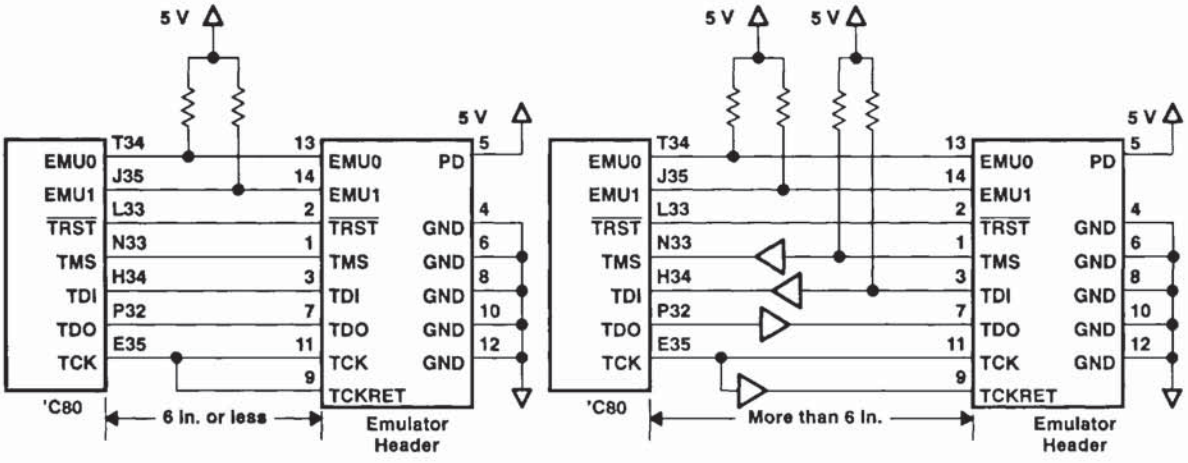


Figure 67. Suggested Emulation Interface Configurations

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

emulator interface connection (continued)

The target system can also generate the test clock. This allows the user to:

- Set the test clock frequency to match his system requirements. (The emulator provides only a 10-MHz test clock.)
- Have other devices in the system that require a test clock when the emulator is not connected

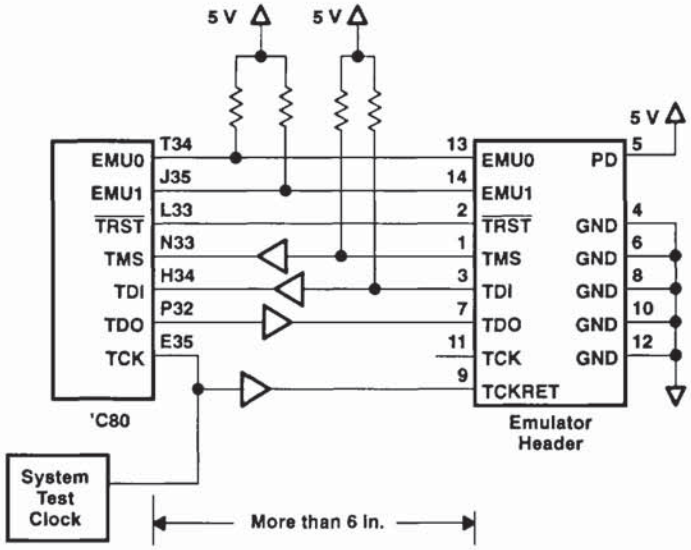


Figure 68. Emulation Interface Configuration With Test Clock Generated by the Target System

For multiprocessor applications, the following are recommended:

- TMS, TDI, TDO, and TCK should be buffered through the same physical package to reduce timing skew.
- Input buffers for TMS, TDI and TCK should be pulled high (5 V). A pullup resistor of 4.7 kΩ or greater is suggested.
- Buffering EMU0 and EMU1 is highly recommended to provide isolation. The buffers need not be in the same physical package as TMS, TCK, TDI, or TDO. Pullups to 5 V are required and should provide a signal rise time of less than 10 μs. A 4.7-kΩ resistor is suggested for most applications.
- To ensure high quality signals, special PWB routing and use of termination resistors may be required. The emulator provides fixed series termination (33 Ω) of TMS and TDI and optional parallel terminators (180 Ω pullup and 270 Ω pulldown) on TCKRET and TDO.

ADVANCE INFORMATION

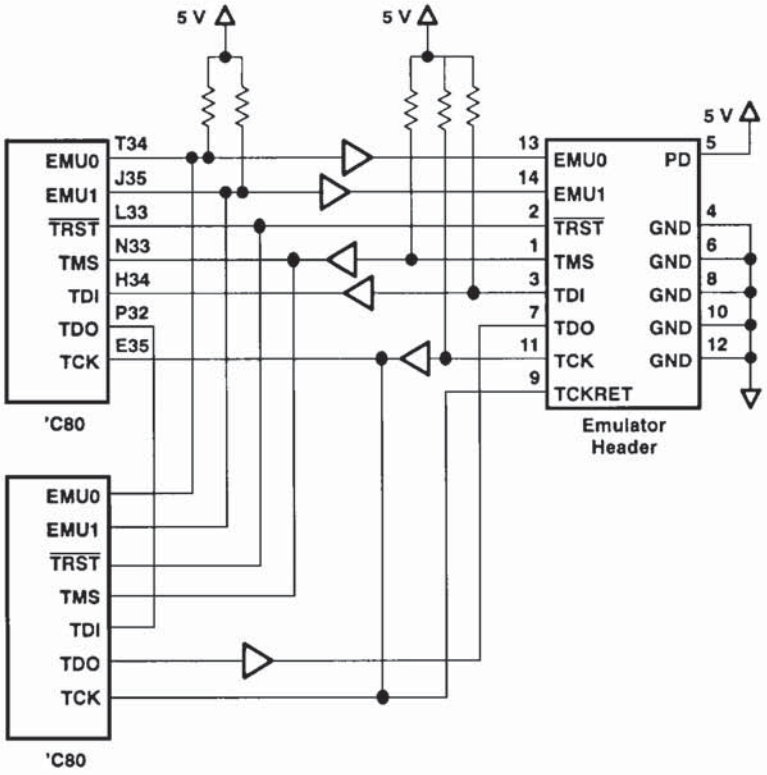
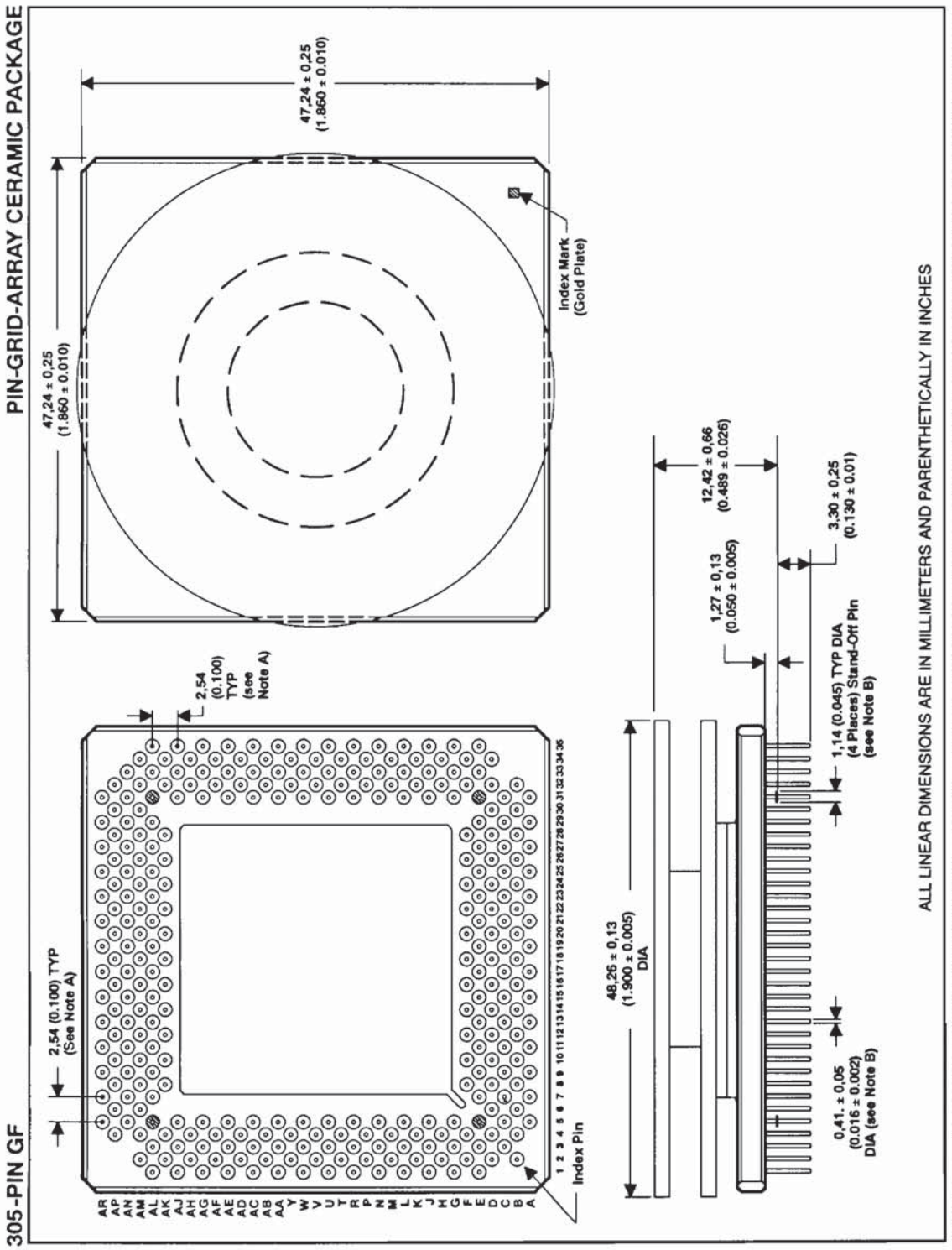


Figure 69. Emulator Interface With Multiprocessor Applications

ADVANCE INFORMATION

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

MECHANICAL DATA



NOTES: A. Pins are located within 0,13 (0.005) radius of the true position relative to each other at maximum material condition and within 0,457 (0.018) radius of the center of the ceramic.

B. Dimensions do not include solder finish.

ADVANCE INFORMATION