- Advanced Multibus Architecture With Three Separate 16-Bit Data Memory Buses and One Program Memory Bus
- 40-Bit Arithmetic Logic Unit (ALU) Including a 40-Bit Barrel Shifter and Two Independent 40-Bit Accumulators
- 17- × 17-Bit Parallel Multiplier Coupled to a 40-Bit Dedicated Adder for Non-Pipelined Single-Cycle Multiply/Accumulate (MAC) Operation
- Compare, Select, and Store Unit (CSSU) for the Add/Compare Selection of the Viterbi Operator
- Exponent Encoder to Compute an Exponent Value of a 40-Bit Accumulator Value in a Single Cycle
- Two Address Generators With Eight Auxiliary Registers and Two Auxiliary Register Arithmetic Units (ARAUs)
- Data Bus With a Bus Holder Feature
- Extended Addressing Mode for 8M × 16-Bit Maximum Addressable External Program Space
- 64K x 16-Bit On-Chip RAM Composed of:
  - Four Blocks of 2K × 16-Bit On-Chip Dual-Access Program/Data RAM
  - Seven Blocks of 8K × 16-Bit On-Chip Single-Access Program/Data RAM
- 16K × 16-Bit On-Chip ROM Configured to Program Memory
- Enhanced External Parallel Interface (XIO2)
- Single-Instruction-Repeat and Block-Repeat Operations for Program Code
- Block-Memory-Move Instructions for Better Program and Data Management
- Instructions With a 32-Bit Long Word Operand
- Instructions With Two- or Three-Operand Reads
- Arithmetic Instructions With Parallel Store and Parallel Load
- Conditional Store Instructions
- Fast Return From Interrupt
- On-Chip Peripherals
  - Software-Programmable Wait-State Generator and Programmable Bank-Switching
  - On-Chip Programmable Phase-Locked Loop (PLL) Clock Generator With Internal Oscillator or External Clock Source
  - One 16-Bit Timer
  - Six-Channel Direct Memory Access (DMA) Controller
  - Three Multichannel Buffered Serial Ports (McBSPs)
  - 8-Bit Enhanced Parallel Host-Port Interface (HPI8)
- Power Consumption Control With IDLE1, IDLE2, and IDLE3 Instructions With Power-Down Modes
- CLKOUT Off Control to Disable CLKOUT
- On-Chip Scan-Based Emulation Logic, IEEE Std 1149.1† (JTAG) Boundary Scan Logic
- 144-Pin Thin Quad Flatpack (TQFP) (PGE Suffix)
- 176-Pin Ball Grid Array (BGA) (GGW Suffix)
- 10-ns and 8.3-ns Single-Cycle Fixed-Point Instruction Execution Time (100 and 120 MIPS)
- 3.3-V I/O and 2.5-V Core Supply Voltages

description

The TMS320VC5410 fixed-point, digital signal processor (DSP) (hereafter referred to as the ’5410 unless otherwise specified) is based on an advanced modified Harvard architecture that has one program memory bus and three data memory buses. This processor provides an arithmetic logic unit (ALU) with a high degree of parallelism, application-specific hardware logic, on-chip memory, and additional on-chip peripherals. The basis of the operational flexibility and speed of this DSP is a highly specialized instruction set.

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

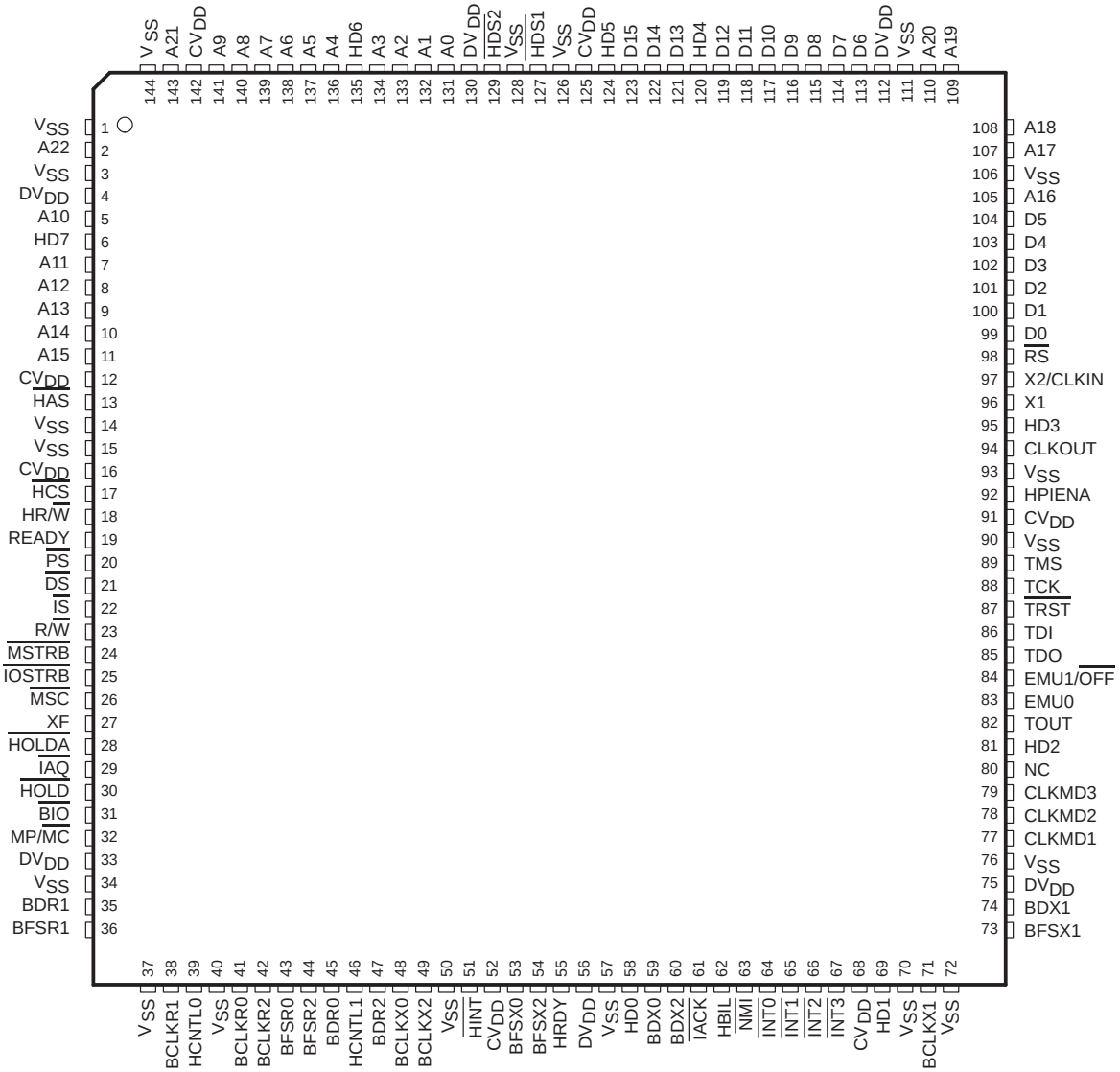
† IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

description (continued)

Separate program and data spaces allow simultaneous access to program instructions and data, providing a high degree of parallelism. Two read operations and one write operation can be performed in a single cycle. Instructions with parallel store and application-specific instructions can fully utilize this architecture. In addition, data can be transferred between data and program spaces. Such parallelism supports a powerful set of arithmetic, logic, and bit-manipulation operations that can all be performed in a single machine cycle. The ’5410 also includes the control mechanisms to manage interrupts, repeated operations, and function calls.

NOTE: This data sheet is designed to be used in conjunction with the *TMS320C5000 DSP Family Functional Overview* (literature number SPRU307).

PGE PACKAGE†‡
(TOP VIEW)

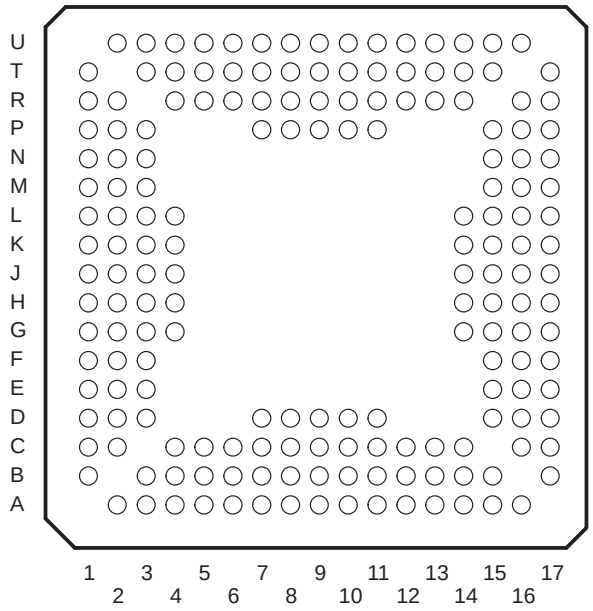


† V_{SS} and DV_{DD} are power supplies for I/O pins while V_{SS} and CV_{DD} are power supplies for core CPU.
‡ The McBSP pins BCLKS0, BCLKS1, and BCLKS2 are not available on the PGE package.

The pin assignments table lists each signal and pin number for the TMS320VC5410PGE (144-pin) package. The terminal functions table lists each terminal name, function, and operating mode for the TMS320VC5410.

description (continued)

GGW PACKAGE
(BOTTOM VIEW)



The pin assignments table lists each signal and pin number for the TMS320VC5410GGW (176-pin) package. The terminal functions table lists each terminal name, function, and operating modes for the TMS320VC5410.

Pin Assignments for the TMS320VC5410PGE (144-Pin Package) and the TMS320VC5410GGW (176-Pin Package)

PIN NAME	PGE PIN NO.	GGW PIN NO.	PIN NAME	PGE PIN NO.	GGW PIN NO.
V_{SS}	1	B1	BFSR1	36	R2
A22	2	C2	CV_{DD}		T1
V_{SS}	3	C1	V_{SS}	37	U2
DV_{DD}	4	D3	BCLKR1	38	T3
CV_{DD}		D2	HCNTL0	39	U3
A10	5	D1	V_{SS}	40	R4
HD7	6	E3	DV_{DD}		T4
V_{SS}		E2	BCLKR0	41	U4
A11	7	E1	BCLKR2	42	R5
A12	8	F3	BFSR0	43	T5
A13	9	F2	BCLKS0		U5
A14	10	F1	BFSR2	44	R6
A15	11	G4	BDR0	45	T6
DV_{DD}		G3	V_{SS}		U6
CV_{DD}	12	G2	HCNTL1	46	P7
$\overline{HAS}$	13	G1	BDR2	47	R7
V_{SS}	14	H1	CV_{DD}		T7
V_{SS}	15	H4	BCLKX0	48	U7
CV_{DD}	16	H3	BCLKX2	49	U8
$\overline{HCS}$	17	H2	BCLKS2		P8
HR/$\overline{W}$	18	J1	V_{SS}	50	R8
READY	19	J4	$\overline{HINT}$	51	T8
$\overline{PS}$	20	J3	CV_{DD}	52	U9
$\overline{DS}$	21	J2	BFSX0	53	P9
V_{SS}		K1	BFSX2	54	R9
$\overline{IS}$	22	K2	HRDY	55	T9
R/$\overline{W}$	23	K4	DV_{DD}	56	U10
DV_{DD}		K3	V_{SS}	57	T10
$\overline{MSTRB}$	24	L1	HD0	58	P10
$\overline{IOSTRB}$	25	L2	BDX0	59	R10
CV_{DD}		L3	CV_{DD}		U11
$\overline{MSC}$	26	L4	BDX2	60	T11
XF	27	M1	$\overline{IACK}$	61	R11
$\overline{HOLDA}$	28	M2	V_{SS}		P11
$\overline{IAQ}$	29	M3	HBIL	62	U12
$\overline{HOLD}$	30	N1	$\overline{NMI}$	63	T12
$\overline{BIO}$	31	N2	$\overline{INT0}$	64	R12
MP/$\overline{MC}$	32	N3	$\overline{INT1}$	65	U13
DV_{DD}	33	P1	DV_{DD}		T13
V_{SS}	34	P2	$\overline{INT2}$	66	R13
BCLKS1		P3	$\overline{INT3}$	67	U14
BDR1	35	R1	CV_{DD}	68	T14
HD1	69	R14	CV_{DD}		D17
V_{SS}	70	U15	A16	105	D16

Pin Assignments for the TMS320VC5410PGE (144-Pin Package) and the TMS320VC5410GGW (176-Pin Package) (Continued)

PIN NAME	PGE PIN NO.	GGW PIN NO.	PIN NAME	PGE PIN NO.	GGW PIN NO.
BCLKX1	71	T15	V_{SS}	106	D15
V_{SS}	72	U16	A17	107	C17
CV_{DD}		T17	A18	108	C16
BFSX1	73	R16	DV_{DD}		B17
BDX1	74	R17	CV_{DD}		A16
DV_{DD}	75	P15	A19	109	B15
V_{SS}	76	P16	A20	110	A15
CLKMD1	77	P17	V_{SS}	111	C14
CLKMD2	78	N15	DV_{DD}	112	B14
CLKMD3	79	N16	D6	113	A14
NC	80	N17	D7	114	C13
HD2	81	M15	D8	115	B13
TOUT	82	M16	D9	116	A13
EMU0	83	M17	D10	117	C12
V_{SS}		L14	D11	118	B12
EMU1/$\overline{OFF}$	84	L15	DV_{DD}		A12
TDO	85	L16	D12	119	D11
TDI	86	L17	HD4	120	C11
$\overline{TRST}$	87	K17	V_{SS}		B11
TCK	88	K14	D13	121	A11
TMS	89	K15	D14	122	A10
V_{SS}	90	K16	D15	123	D10
CV_{DD}	91	J17	HD5	124	C10
HPIENA	92	J14	CV_{DD}	125	B10
V_{SS}	93	J15	V_{SS}	126	A9
DV_{DD}		J16	$\overline{HDS1}$	127	D9
CLKOUT	94	H17	V_{SS}	128	C9
HD3	95	H16	$\overline{HDS2}$	129	B9
X1	96	H14	DV_{DD}	130	A8
X2/CLKIN	97	H15	A0	131	B8
$\overline{RS}$	98	G17	A1	132	D8
V_{SS}		G16	CV_{DD}		C8
D0	99	G15	A2	133	A7
D1	100	G14	A3	134	B7
DV_{DD}		F17	DV_{DD}		C7
D2	101	F16	HD6	135	D7
D3	102	F15	A4	136	A6
V_{SS}		E17	V_{SS}		B6
D4	103	E16	A5	137	C6
D5	104	E15	A6	138	A5
DV_{DD}		B5	DV_{DD}		C4
A7	139	C5	CV_{DD}	142	A3
A8	140	A4	A21	143	B3
A9	141	B4	V_{SS}	144	A2

terminal functions

The terminal functions table lists each signal, function, and operating mode(s) grouped by function.

Terminal Functions

TERMINAL NAME		I/O†	DESCRIPTION
DATA SIGNALS			
A22 A21 A20 A19 A18 A17 A16 A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0	(MSB) (LSB)	O/Z	Parallel address bus A22 [most significant bit (MSB)] through A0 [least significant bit (LSB)]. The sixteen LSB lines, A0 to A15, are multiplexed to address external memory (program, data) or I/O. The seven MSB lines, A16 to A22, address external program space memory. A22–A0 is placed in the high-impedance state in the hold mode. A22–A0 also goes into the high-impedance state when $\overline{OFF}$ is low. The address bus has a bus holder feature that eliminates passive components and the power dissipation associated with them. The bus holder keeps the address bus at the previous logic level when the bus goes into a high-impedance state.
D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0	(MSB) (LSB)	I/O/Z	Parallel data bus D15 (MSB) through D0 (LSB). D15–D0 is multiplexed to transfer data between the core CPU and external data/program memory or I/O devices. D15–D0 is placed in high-impedance state when not outputting data or when $\overline{RS}$ or $\overline{HOLD}$ is asserted. D15–D0 also goes into the high-impedance state when $\overline{OFF}$ is low. The data bus has a bus holder feature that eliminates passive components and the power dissipation associated with them. The bus holder keeps the data bus at the previous logic level when the bus goes into a high-impedance state. The bus holders on the data bus can be enabled/disabled under software control.

† I = Input, O = Output, Z = High-impedance, S = Supply

Terminal Functions (Continued)

TERMINAL NAME	I/O†	DESCRIPTION
INITIALIZATION, INTERRUPT AND RESET OPERATIONS		
$\overline{\text{IACK}}$	O/Z	Interrupt acknowledge signal. $\overline{\text{IACK}}$ indicates receipt of an interrupt and that the program counter is fetching the interrupt vector location designated by A15–A0. $\overline{\text{IACK}}$ also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
$\overline{\text{INT0}}$ $\overline{\text{INT1}}$ $\overline{\text{INT2}}$ $\overline{\text{INT3}}$	I	External user interrupt inputs. $\overline{\text{INT0}}$–$\overline{\text{INT3}}$ is prioritized and is maskable by the interrupt mask register (IMR) and interrupt mode bit. $\overline{\text{INT0}}$ –$\overline{\text{INT3}}$ can be polled and reset by way of the interrupt flag register (IFR).
$\overline{\text{NMI}}$	I	Nonmaskable interrupt. $\overline{\text{NMI}}$ is an external interrupt that cannot be masked by way of the INTM or the IMR. When $\overline{\text{NMI}}$ is activated, the processor traps to the appropriate vector location.
$\overline{\text{RS}}$	I	Reset. $\overline{\text{RS}}$ causes the digitial signal processor (DSP) to terminate execution and forces the program counter to 0FF80h. When $\overline{\text{RS}}$ is brought to a high level, execution begins at location 0FF80h of program memory. $\overline{\text{RS}}$ affects various registers and status bits.
MP/$\overline{\text{MC}}$	I	Microprocessor/microcomputer mode select pin. If active low at reset (microcomputer mode), MP/$\overline{\text{MC}}$ causes the internal program ROM to be mapped into the upper 16K words of program memory space. In the microprocessor mode, off-chip memory and its corresponding addresses (instead of internal program ROM) are accessed by the DSP.
MULTIPROCESSING SIGNALS		
$\overline{\text{BIO}}$	I	Branch control. A branch can be conditionally executed when $\overline{\text{BIO}}$ is active. If low, the processor executes the conditional instruction. The $\overline{\text{BIO}}$ condition is sampled during the decode phase of the pipeline for the XC instruction, and all other instructions sample $\overline{\text{BIO}}$ during the read phase of the pipeline.
XF	O/Z	External flag output (latched software-programmable signal). XF is set high by the SSBX XF instruction, set low by RSBX XF instruction or by loading ST1. XF is used for signaling other processors in multiprocessor configurations or used as a general-purpose output pin. XF goes into the high-impedance state when $\overline{\text{OFF}}$ is low, and is set high at reset.
MEMORY CONTROL SIGNALS		
$\overline{\text{DS}}$ $\overline{\text{PS}}$ $\overline{\text{IS}}$	O/Z	Data, program, and I/O space select signals. $\overline{\text{DS}}$, $\overline{\text{PS}}$, and $\overline{\text{IS}}$ are always high unless driven low for communicating to a particular external space. Active period corresponds to valid address information. $\overline{\text{DS}}$, $\overline{\text{PS}}$, and $\overline{\text{IS}}$ are placed into the high-impedance state in the hold mode; these signals also go into the high-impedance state when $\overline{\text{OFF}}$ is low.
$\overline{\text{MSTRB}}$	O/Z	Memory strobe signal. $\overline{\text{MSTRB}}$ is always high unless low-level asserted to indicate an external bus access to data or program memory. $\overline{\text{MSTRB}}$ is placed in the high-impedance state in the hold mode; it also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
READY	I	Data ready. READY indicates that an external device is prepared for a bus transaction to be completed. If the device is not ready (READY is low), the processor waits one cycle and checks READY again. Note that the processor performs ready detection if at least two software wait states are programmed. The READY signal is not sampled until the completion of the software wait states.
R/$\overline{\text{W}}$	O/Z	Read/write signal. R/$\overline{\text{W}}$ indicates transfer direction during communication to an external device. R/$\overline{\text{W}}$ is normally in the read mode (high), unless it is asserted low when the DSP performs a write operation. R/$\overline{\text{W}}$ is placed in the high-impedance state in the hold mode; and it also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
$\overline{\text{IOSTRB}}$	O/Z	I/O strobe signal. $\overline{\text{IOSTRB}}$ is always high unless low-level asserted to indicate an external bus access to an I/O device. $\overline{\text{IOSTRB}}$ is placed in the high-impedance state in the hold mode; it also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
$\overline{\text{HOLD}}$	I	Hold input. $\overline{\text{HOLD}}$ is asserted to request control of the address, data, and control lines. When acknowledged by the 'VC5410, these lines go into the high-impedance state.
$\overline{\text{HOLDA}}$	O/Z	Hold acknowledge. $\overline{\text{HOLDA}}$ indicates to the external circuitry that the processor is in a hold state and that the address, data, and control lines are in the high-impedance state, allowing them to be available to the external circuitry. $\overline{\text{HOLDA}}$ also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
$\overline{\text{MSC}}$	O/Z	Microstate complete. $\overline{\text{MSC}}$ goes low when the last wait state of two or more internal software wait states programmed is executed. If connected to the READY line, $\overline{\text{MSC}}$ forces one external wait state after the last internal wait state has been completed. $\overline{\text{MSC}}$ also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.

† I = Input, O = Output, Z = High-impedance, S = Supply

Terminal Functions (Continued)

TERMINAL NAME	I/O†	DESCRIPTION
MEMORY CONTROL SIGNALS (CONTINUED)		
$\overline{IAQ}$	O/Z	Instruction acquisition signal. $\overline{IAQ}$ is asserted (active low) when there is an instruction address on the address bus and goes into the high-impedance state when $\overline{OFF}$ is low.
OSCILLATOR/TIMER SIGNALS		
CLKOUT	O/Z	Clock output signal. CLKOUT can represent the machine-cycle rate of the CPU divided by 1, 2, 3, or 4 as configured in the bank-switching control register (BSCR). Following reset, CLKOUT represents the machine-cycle rate divided by 4.
CLKMD1 CLKMD2 CLKMD3	I	Clock mode select signals. CLKMD1 – CLKMD3 allows the selection and configuration of different clock modes such as crystal, external clock, PLL mode.
X2/CLKIN	I	Clock/oscillator input. If the internal oscillator is not being used, X2/CLKIN functions as the clock input.
X1	O	Output pin from the internal oscillator for the crystal. If the internal oscillator is not used, X1 should be left unconnected. X1 does not go into the high-impedance state when $\overline{OFF}$ is low.
TOUT	O	Timer output. TOUT signals a pulse when the on-chip timer counts down past zero. The pulse is one CLKOUT cycle wide. TOUT also goes into the high-impedance state when $\overline{OFF}$ is low.
MULTICHANNEL BUFFERED SERIAL PORT 0 (McBSP #0), MULTICHANNEL BUFFERED SERIAL PORT 1 (McBSP #1), AND MULTICHANNEL BUFFERED SERIAL PORT 2 (McBSP #2) SIGNALS		
BCLKR0 BCLKR1 BCLKR2	I/O/Z	Receive clock input. BCLKR serves as the serial shift clock for the buffered serial port receiver.
BDR0 BDR1 BDR2	I	Serial data receive input
BFSR0 BFSR1 BFSR2	I/O/Z	Frame synchronization pulse for receive input. The BFSR pulse initiates the receive data process over BDR.
BCLKX0 BCLKX1 BCLKX2	I/O/Z	Transmit clock. BCLKX serves as the serial shift clock for the McBSP transmitter. BCLKX can be configured as an input or an output, and is configured as an input following reset. BCLKX enters the high-impedance state when $\overline{OFF}$ goes low.
BDX0 BDX1 BDX2	O/Z	Serial data transmit output. BDX is placed in the high-impedance state when not transmitting, when $\overline{RS}$ is asserted, or when $\overline{OFF}$ is low.
BFSX0 BFSX1 BFSX2	I/O/Z	Frame synchronization pulse for transmit input/output. The BFSX pulse initiates the data transmit process over BDX. BFSX can be configured as an input or an output, and is configured as an input following reset. BFSX goes into the high-impedance state when $\overline{OFF}$ is low.
BCLKS0 BCLKS1 BCLKS2	I	Serial port clock reference. The McBSP can be programmed to use either BCLKS or the CPU clock as a reference for generation of internal clock and frame sync signals. Pins with internal pullup devices. NOTE: These pins are not available on the PGE package.
MISCELLANEOUS SIGNAL		
NC		No connection
HOST-PORT INTERFACE SIGNALS		
HD0–HD7	I/O/Z	Parallel bidirectional data bus. HD0–HD7 is placed in the high-impedance state when not outputting data. The signals go into the high-impedance state when $\overline{OFF}$ is low. The HPI data bus has a feature called a bus holder that eliminates passive components and the power dissipation associated with them. The bus holder keeps the data bus at the previous logic level when the bus goes into high-impedance state. The bus holder on the HPI data bus can be enabled/disabled under software control.

† I = Input, O = Output, Z = High-impedance, S = Supply

Terminal Functions (Continued)

TERMINAL NAME	I/O†	DESCRIPTION
HOST-PORT INTERFACE SIGNALS (CONTINUED)		
HCNTL0 HCNTL1	I	Control inputs
HBIL	I	Byte identification
$\overline{\text{HCS}}$	I	Chip select
$\overline{\text{HDS1}}$ $\overline{\text{HDS2}}$	I	Data strobe
$\overline{\text{HAS}}$	I	Address strobe
HR/$\overline{\text{W}}$	I	Read/write
HRDY	O/Z	Ready output. HRDY goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
$\overline{\text{HINT}}$	O/Z	Interrupt output. When the DSP is in reset, $\overline{\text{HINT}}$ is driven high. $\overline{\text{HINT}}$ goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
HPIENA	I	HPI module select. HPIENA must be tied to DV_{DD} to have HPI selected. If HPIENA is left open or connected to ground, the HPI module is not selected, internal pullup for the HPI input pins are enabled, and the HPI data bus has holders set. HPIENA is provided with an internal pulldown resistor that is active only when $\overline{\text{RS}}$ is low. HPIENA is sampled when $\overline{\text{RS}}$ goes high and is ignored until $\overline{\text{RS}}$ goes low again.
SUPPLY PNS		
V_{SS}	S	Ground. Dedicated power supply for the core CPU.
CV_{DD}	S	$+V_{DD}$. Dedicated power supply for the core CPU.
DV_{DD}	S	$+V_{DD}$. Dedicated power supply for I/O pins.
TEST PINS		
TCK	I	IEEE standard 1149.1 test clock. TCK is normally a free-running clock signal with a 50% duty cycle. The changes on test access port (TAP) of input signals TMS and TDI are clocked into the TAP controller, instruction register, or selected test data register on the rising edge of TCK. Changes at the TAP output signal (TDO) occur on the falling edge of TCK.
TDI	I	IEEE standard 1149.1 test data input. Pin with internal pullup device. TDI is clocked into the selected register (instruction or data) on a rising edge of TCK.
TDO	O/Z	IEEE standard 1149.1 test data output. The contents of the selected register (instruction or data) are shifted out of TDO on the falling edge of TCK. TDO is in the high-impedance state except when the scanning of data is in progress. TDO also goes into the high-impedance state when $\overline{\text{OFF}}$ is low.
TMS	I	IEEE standard 1149.1 test mode select. Pin with internal pullup device. This serial control input is clocked into the TAP controller on the rising edge of TCK.
$\overline{\text{TRST}}$	I	IEEE standard 1149.1 test reset. $\overline{\text{TRST}}$, when high, gives the IEEE standard 1149.1 scan system control of the operations of the device. If $\overline{\text{TRST}}$ is not connected or driven low, the device operates in its functional mode, and the IEEE standard 1149.1 signals are ignored. Pin with internal pulldown device.
EMU0	I/O/Z	Emulator 0 pin. When $\overline{\text{TRST}}$ is driven low, EMU0 must be high for activation of the $\overline{\text{OFF}}$ condition. When $\overline{\text{TRST}}$ is driven high, EMU0 is used as an interrupt to or from the emulator system and is defined as input/output by way of the IEEE standard 1149.1 scan system.
EMU1/$\overline{\text{OFF}}$	I/O/Z	Emulator 1 pin/disable all outputs. When $\overline{\text{TRST}}$ is driven high, EMU1/$\overline{\text{OFF}}$ is used as an interrupt to or from the emulator system and is defined as input/output by way of IEEE standard 1149.1 scan system. When $\overline{\text{TRST}}$ is driven low, EMU1/$\overline{\text{OFF}}$ is configured as $\overline{\text{OFF}}$. The EMU1/$\overline{\text{OFF}}$ signal, when active low, puts all output drivers into the high-impedance state. Note that $\overline{\text{OFF}}$ is used exclusively for testing and emulation purposes (not for multiprocessing applications). Therefore, for the $\overline{\text{OFF}}$ condition, the following apply: $\overline{\text{TRST}}$ = low, EMU0 = high EMU1/$\overline{\text{OFF}}$ = low

† I = Input, O = Output, Z = High-impedance, S = Supply

architecture

The ’VC5410 DSP implements the standard ’C54x CPU which uses an advanced, modified Harvard architecture that maximizes processing power by maintaining three separate bus structures for data memory and one for program memory. Separate program and data spaces allow simultaneous access to program instructions and data, providing a high degree of parallelism. For example, two read operations and one write operation can be performed in a single cycle. Instructions with parallel store and application-specific instructions fully utilize this architecture. In addition, data can be transferred between data and program spaces. Such parallelism supports a powerful set of arithmetic, logic, and bit-manipulation operations that can all be performed in a single machine cycle. In addition, the ’VC5410 includes the control mechanisms to manage interrupts, repeated operations, and function calls.

For detailed information on the architecture of the C5000 family of DSPs, refer to the *TMS320C5000 DSP Family Functional Overview* (literature number SPRU307).

memory

The ’VC5410 device provides both on-chip ROM and RAM memories to aid in system performance and integration.

on-chip ROM with bootloader

The ’VC5410 features a 16K-word $\times$ 16-bit on-chip maskable ROM that can only be mapped into program memory space.

Customers can arrange to have the ROM of the ’VC5410 programmed with contents unique to any particular application.

A bootloader is available in the standard ’VC5410 on-chip ROM. This bootloader can be used to automatically transfer user code from an external source to anywhere in the program memory at power up. If MP/$\overline{\text{MC}}$ of the device is sampled low during a hardware reset, execution begins at location FF80h of the on-chip ROM. This location contains a branch instruction to the start of the bootloader program. The standard ’VC5410 devices provide different ways to download the code to accomodate various system requirements:

- Parallel from 8-bit or 16-bit-wide EPROM
- Parallel from I/O space, 8-bit or 16-bit mode
- Serial boot from serial ports, 8-bit or 16-bit mode
- Host-port interface boot
- Warm boot

on-chip ROM with bootloader (continued)

The standard on-chip ROM layout is shown in Table 1.

Table 1. Standard On-Chip ROM Layout†

ADDRESS RANGE	DESCRIPTION
C000h–D4FFh	ROM tables for the GSM EFR speech codec
D500h–D6FFh	256-point complex radix-2 DIT FFT with looped code
D700h–DCFFh	FFT twiddle factors for a 256-point complex radix-2 FFT
DD00h–DEFFh	1024-point complex radix-2 DIT FFT with looped code
DF00h–F7FFh	FFT twiddle factors for a 1024-point complex radix-2 FFT
F800h–FBFFh	Bootloader
FC00h–FCFFh	μ-Law expansion table
FD00h–FDFFh	A-Law expansion table
FE00h–FEFFh	Sine look-up table
FF00h–FF7Fh	Reserved†
FF80h–FFFFh	Interrupt vector table

† In the ’VC5410 ROM, 128 words are reserved for factory device-testing purposes. Application code to be implemented in on-chip ROM must reserve these 128 words at addresses FF00h–FF7Fh in program space.

on-chip RAM

The ’VC5410 device contains 8K words × 16-bit on-chip dual-access RAM (DARAM) and 56K words × 16-bit of on-chip single-access RAM (SARAM).

The DARAM is composed of four blocks of 2K words each. Each block in the DARAM can support two reads in one cycle, or a read and a write in one cycle. The DARAM is located in the address range 0080h–1FFFh in data space, and can be mapped into program/data space by setting the OVLY bit to one.

The SARAM is composed of seven blocks of 8K words each. Each of these seven blocks is a single-access memory. For example, an instruction word can be fetched from one SARAM block in the same cycle as a data word is written to another SARAM block. The SARAM located in the address range 2000h–7FFFh in data space can be mapped into program space by setting the OVLY bit to one, while the SARAM located in the address range 18000h–1FFFFh in program space can be mapped into data space by setting the DROM bit to one.

on-chip memory security

The ’VC5410 device has a maskable option to protect the contents of on-chip memories. When the ROM protect bit is set, no externally originating instruction can access the on-chip memory spaces. In addition, when the ROM protect option is enabled, HPI8 read access is limited to address range 0001000h – 0001FFFh. Data located outside this range cannot be read through the HPI8. Write access to the entire HPI8 memory map is still maintained.

memory map

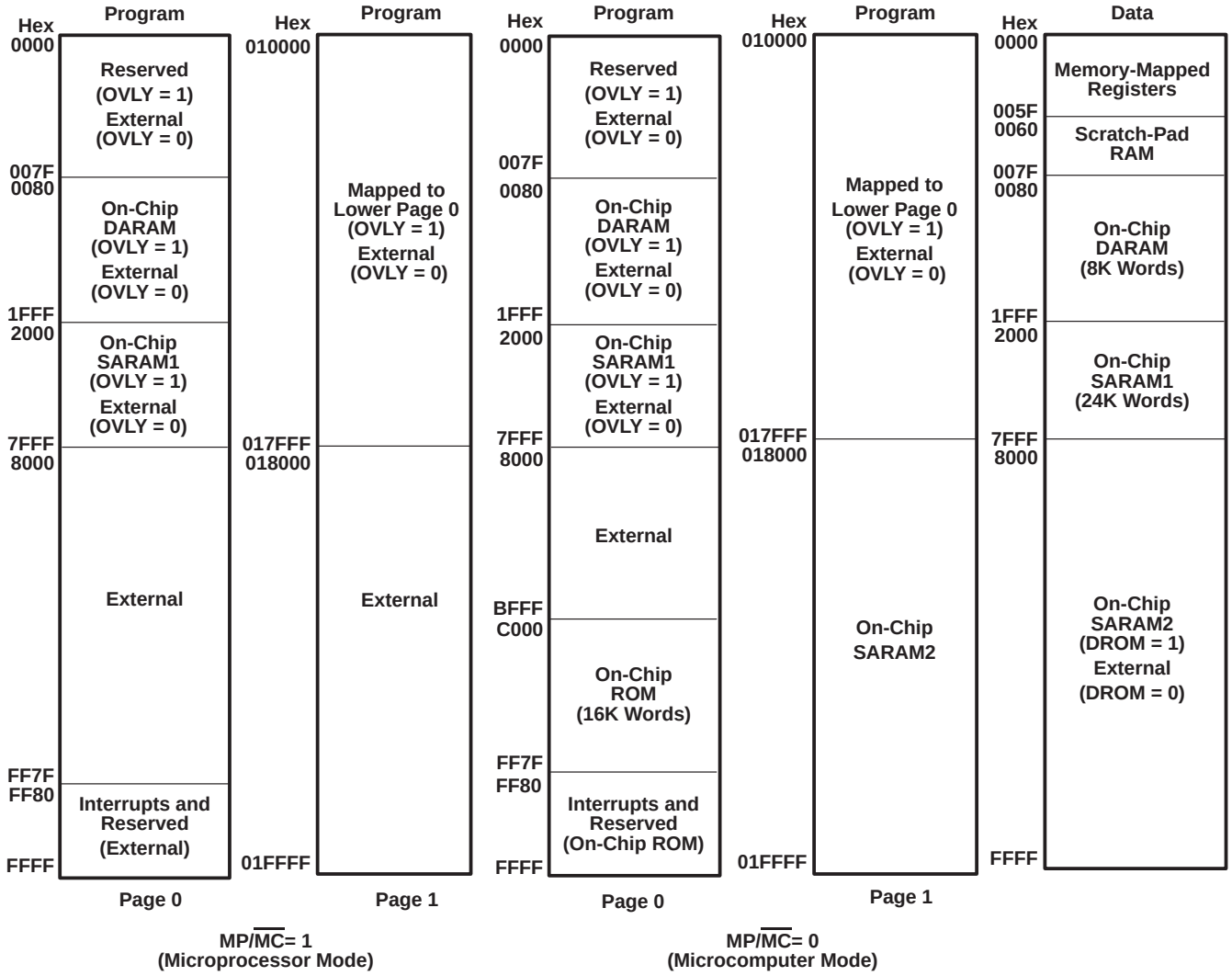


Figure 1. Memory Map

program memory

Software can configure their memory cells to reside inside or outside of the program address map. When the cells are mapped into program space, the device automatically accesses them when their addresses are within bounds. When the program-address generation (PAGEN) logic generates an address outside its bounds, the device automatically generates an external access. The advantages of operating from on-chip memory are as follows:

- Higher performance because no wait states are required
- Lower cost than external memory
- Lower power than external memory

The advantage of operating from off-chip memory is the ability to access a larger address space.

relocatable interrupt vector table

The reset, interrupt, and trap vectors are addressed in program space. These vectors are soft — meaning that the processor, when taking the trap, loads the program counter (PC) with the trap address and executes the code at the vector location. Four words, either two 1-word instructions or one 2-word instruction, are reserved at each vector location to accommodate a delayed branch instruction which allows branching to the appropriate interrupt service routine without the overhead.

At device reset, the reset, interrupt, and trap vectors are mapped to address FF80h in program space. However, these vectors can be remapped to the beginning of any 128-word page in program space after device reset. This is done by loading the interrupt vector pointer (IPTR) bits in the PMST register with the appropriate 128-word page boundary address. After loading IPTR, any user interrupt or trap vector is mapped to the new 128-word page.

NOTE: The hardware reset ($\overline{RS}$) vector cannot be remapped because the hardware reset loads the IPTR with 1s. Therefore, the reset vector is always fetched at location FF80h in program space.

extended program memory

The ’VC5410 uses a paged extended memory scheme in program space to allow access of up to 8192K of program memory. In order to implement this scheme, the ’VC5410 includes several features which are also present on ’C548/549:

- Twenty-three address lines, instead of sixteen
- An extra memory-mapped register, the XPC
- Six extra instructions for addressing extended program space

Program memory in the ’VC5410 is organized into 128 pages that are each 64K in length, as shown in Figure 2.

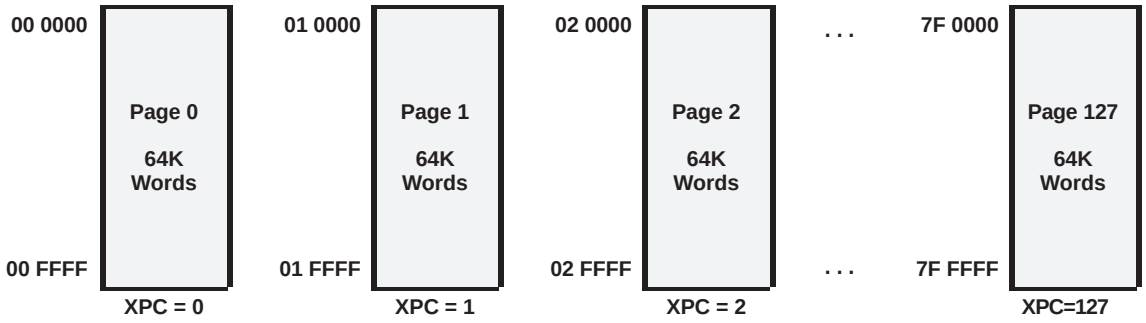
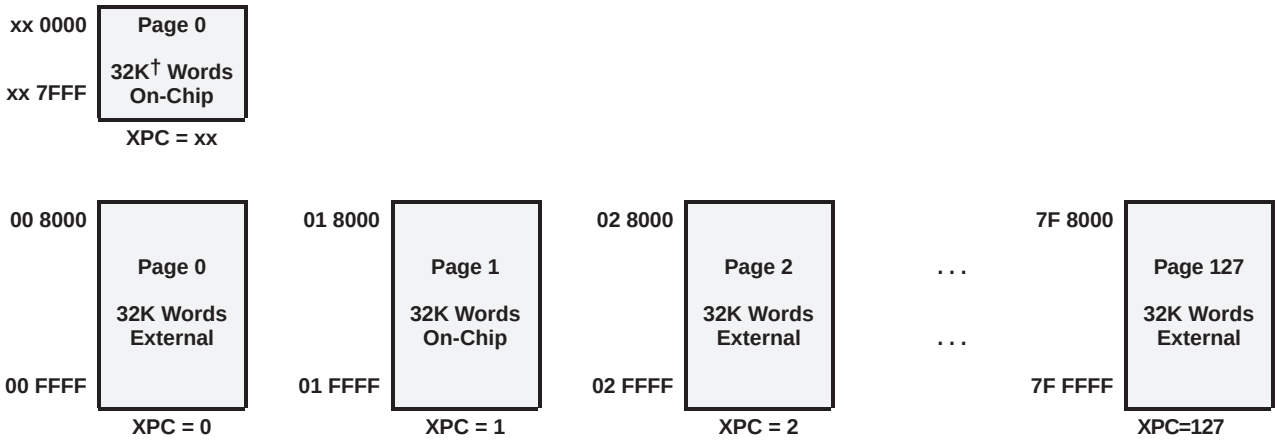


Figure 2. Extended Program Memory
(On-Chip RAM Not Mapped in Program Space and Data Space, OVLY = 0)

extended program memory (continued)

When the on-chip RAM is enabled in program space, each page of program memory is made up of two parts: a common block of 32K words and a unique block of 32K words. The common block is shared by all pages and each unique block is accessible only through its assigned page. Figure 3 shows the common and unique blocks.



† See Figure 1 for more information about this on-chip memory region.

NOTE A: When the on-chip RAM is enabled in program space, all accesses to the region xx 0000 – xx 7FFF, regardless of page number, are mapped to the on-chip RAM at 00 0000 – 00 7FFF.

Figure 3. Extended Program Memory
(On-Chip RAM Mapped in Program Space and Data Space, OVLY = 1)

If the on-chip ROM is enabled (MP/$\overline{MC}$ = 0), it is enabled only on page 0. It is not mapped to any other page in program memory.

The value of the XPC register defines the page selection. This register is memory-mapped into data space to address 001Eh. At a hardware reset, the XPC is initialized to 0.

To facilitate page-switching through software, the ’VC5410 has six special instructions that affect the XPC:

- *FB[D]* pmad (23 bits) – Far branch
- *FBACC[D]* Accu[22:0] – Far branch to the location specified by the value in accumulator A or accumulator B
- *FCALL[D]* pmad (23 bits) – Far call
- *FCALA[D]* Accu[22:0] – Far call to the location specified by the value in accumulator A or accumulator B
- *FRET[D]* – Far return
- *FRETE[D]* – Far return with interrupts enabled

In addition to these new instructions, two ’54x instructions are extended to use 23 bits in the ’VC5410:

- READA data_memory (using 23-bit accumulator address)
- WRITA data_memory (using 23-bit accumulator address)

All other instructions, software and hardware interrupts do not modify the XPC register and access only memory within the current page.

data memory

The data memory space addresses up to 64K of 16-bit words. The device automatically accesses the on-chip RAM when addressing within its bounds. When an address is generated outside the RAM bounds, the device automatically generates an external access.

The advantages of operating from on-chip memory are as follows:

- Higher performance because no wait states are required
- Higher performance because of better flow within the pipeline of the central arithmetic logic unit (CALU)
- Lower cost than external memory
- Lower power than external memory

The advantage of operating from off-chip memory is the ability to access a larger address space.

on-chip peripherals

The ’VC5410 device has the following peripherals:

- Software-programmable wait-state generator
- Programmable bank-switching
- A host-port interface (HPI8)
- Three multichannel buffered serial ports (McBSPs)
- A hardware timer
- A clock generator with a multiple phase-locked loop (PLL)
- Enhanced external parallel interface (XIO2)
- A DMA controller (DMA)

software-programmable wait-state generator

The software-programmable wait-state generator can extend external bus cycles by up to fourteen CLKOUT cycles, providing a convenient means of interfacing the ’VC5410 with slower external devices. Devices that require more than fourteen wait states can be interfaced using the hardware READY line. When all external accesses are configured for zero wait states, the internal clocks to the wait-state generator are shut off; shutting off these paths from the internal clocks allows the device to run with lower power consumption.

The software-programmable wait-state generator is controlled by the 16-bit software wait-state register (SWWSR), which is memory-mapped to address 0028h in data space.

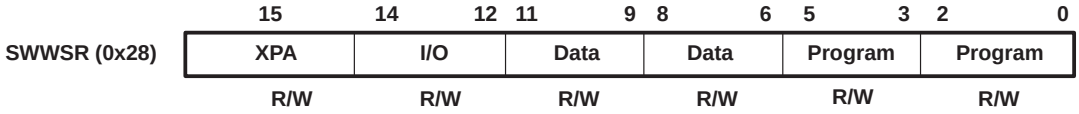
The program and data spaces each consist of two 32K-word blocks; the I/O space consists of one 64K-word block. Each of these blocks has a corresponding 3-bit field in the SWWSR. These fields are shown in Figure 4 and described in Table 2.

The value of a 3-bit field in SWWSR, in conjunction with the software wait-state multiplier (SWSM) bit in the software wait-state control register (SWCR), specifies the number of wait states to be inserted for each access in the corresponding space and address range.

- When SWSM = 0, the possible values for the number of wait states are 0, 1, 2, 3, 4, 5, 6, and 7. This is the default configuration.
- When SWSM = 1, the possible values for the number of wait states are 0, 2, 4, 6, 8, 10, 12, and 14.

At reset, the SWWSR is set to 7FFFh, and SWSM to 0, configuring seven wait states for all external accesses.

software-programmable wait-state generator (continued)



R = Read, W = Write, Reset value = 7FFFh

Figure 4. Software Wait-State Register (SWWSR)

Table 2. Software Wait-State Register Fields

BIT	NAME	RESET VALUE	FUNCTION
15	XPA	0	Extended program address control bit. XPA selects the address ranges selected by the program fields.
14–12	I/O	1	I/O space. The field value (0–14) corresponds to the number of wait states for I/O space 0000–FFFFh.
11–9	Data	1	Data space. The field value (0–14) corresponds to the number of wait states for data space 8000–FFFFh.
8–6†	Data	1	Data space. The field value (0–14) corresponds to the number of wait states for data space 0000–7FFFh.
5–3	Program	1	Program space. The field value (0–14) corresponds to the number of wait states for: XPA = 0 xx8000–xxFFFFh XPA = 1 400000h–7FFFFF
2–0	Program	1	Program space. The field value (0–14) corresponds to the number of wait states for: XPA = 0 xx0000–xx7FFFh XPA = 1 000000–3FFFFFh

† Although this field is present to maintain compatibility with previous C5000 family DSPs, there is no external data space on the 'VC5410 in this address range; therefore, the configuration of this bit field has no effect.

The SWSM bit is located in the software wait-state control register (SWCR), a memory-mapped register (MMR) at address 0x2B, bit 0 position (LSB). The bit fields of the SWCR are shown in Figure 5 and are described in Table 3.

Figure 5. Software Wait-State Control Register (SWCR)

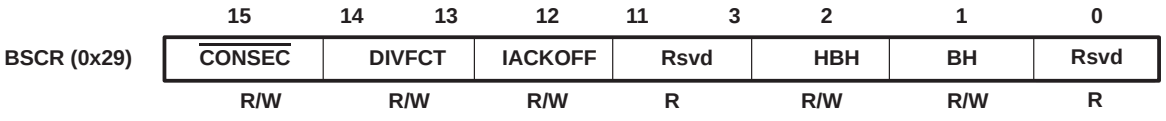
Table 3. Software Wait-State Control Register Fields

BIT	NAME	RESET VALUE	FUNCTION
15–1	Reserved	–	Reserved
0	SWSM	0	Software wait-state multiplier bit. SWSM = 0 Wait states in SWWSR are not multiplied by 2 SWSM = 1 Wait states in SWWSR are multiplied by 2

programmable bank-switching

Programmable bank-switching logic allows the ’VC5410 to switch between external memory banks without requiring external wait states for memories that need additional time to turn off. The bank-switching logic automatically inserts one cycle when accesses cross a 32K-word memory-bank boundary inside program or data space.

Bank-switching is defined by the bank-switching control register (BSCR), which is memory-mapped at address 0029h. The bit fields of the BSCR are shown in Figure 6 and are described in Table 4.



R = Read, W = Write

Figure 6. Bank-Switching Control Register (BSCR)

Table 4. Bank-Switching Control Register Fields

BIT	NAME	RESET VALUE	FUNCTION
15	$\overline{\text{CONSEC}}$†	1	Consecutive bank-switching. Specifies the bank-switching mode. $\overline{\text{CONSEC}}$ = 0 Bank-switching on 32K bank boundaries only. This bit is cleared if fast access is desired for continuous memory reads (i.e., no starting and trailing cycles between read cycles). $\overline{\text{CONSEC}}$ = 1 consecutive bank switches on external memory reads. Each read cycle consists of 3 cycles: starting cycle, read cycle, and trailing cycle.
13–14	DIVFCT	11	CLKOUT output divide factor. The CLKOUT output is driven by an on-chip source having a frequency equal to 1/(DIVFCT+1) of the DSP clock. DIVFCT = 00 CLKOUT is not divided. DIVFCT = 01 CLKOUT is divided by 2 from the DSP clock. DIVFCT = 10 CLKOUT is divided by 3 from the DSP clock. DIVFCT = 11 CLKOUT is divided by 4 from the DSP clock (default value following reset).
12	IACKOFF	1	$\overline{\text{IACK}}$ signal output off. Controls the output of the $\overline{\text{IACK}}$ signal. IACKOFF is set to 1 at reset. IACKOFF = 0 The $\overline{\text{IACK}}$ signal output off function is disabled. IACKOFF = 1 The $\overline{\text{IACK}}$ signal output off function is enabled.
11–3	Rsvd	–	Reserved
2	HBH	0	HPI bus holder. Controls the HPI bus holder. HBH is cleared to 0 at reset. HBH = 0 The bus holder is disabled. HBH = 1 The bus holder is enabled. When not driven, the HPI data bus, HD[7:0] is held in the previous logic level.
1	BH	0	Bus holder. Controls the bus holder. BH is cleared to 0 at reset. BH = 0 The bus holder is disabled. BH = 1 The bus holder is enabled. When not driven, the data bus, D[15:0] is held in the previous logic level.
0	Rsvd	–	Reserved

† For additional information, see the “enhanced external parallel interface (XIO2)” section of this document.

programmable bank-switching (continued)

The ’VC5410 has an internal register that holds the MSB of the last address used for a read or write operation in program or data space. In the non-consecutive bank switches ($\overline{\text{CONSEC}}$ = 0), if the MSB of the address used for the current read does not match that contained in this internal register, the $\overline{\text{MSTRB}}$ (memory strobe) signal is not asserted for one CLKOUT cycle. During this extra cycle, the address bus switches to the new address. The contents of the internal register are replaced with the MSB for the read of the current address. If the MSB of the address used for the current read matches the bits in the register, a normal read cycle occurs.

In non-consecutive bank switches ($\overline{\text{CONSEC}}$ = 0), if repeated reads are performed from the same memory bank, no extra cycles are inserted. When a read is performed from a different memory bank, memory conflicts are avoided by inserting an extra cycle. For more information, see the “enhanced external parallel interface (XIO2)” section of this document.

The bank-switching mechanism automatically inserts one extra cycle in the following cases:

- A memory read followed by another memory read from a different memory bank.
- A program-memory read followed by a data-memory read.
- A data-memory read followed by a program-memory read.
- A program-memory read followed by another program-memory read from a different page.

parallel I/O ports

Each device has a total of 64K I/O ports. These ports can be addressed by the PORTR instruction or the PORTW instruction. The $\overline{\text{IS}}$ signal indicates a read/write operation through an I/O port. The ’VC5410 can interface easily with external devices through the I/O ports while requiring minimal off-chip address-decoding circuits.

enhanced host-port interface (HPI8)

The enhanced host-port interface (HPI8) in the ’VC5410 is an 8-bit parallel port used to interface a host processor to the DSP. Data can be exchanged between the host processor and the DSP throughout the entire on-chip memory via the DMA controller. The extended program memory pages are also accessible by both the host and the DSP. The DSP and the host control the HPI8 activity through the HPI8 control register (HPIC). The host can address memory through the HPI8 address register (HPIA).

Data transfers of 16-bit words occur as two consecutive bytes with a dedicated pin (HBIL) indicating whether the high or low byte is being transmitted. Two pins (controlled by the host), HCNTL0 and HCNTL1, indicate whether the being exchanged is the most significant or least significant byte. Control pins (HCNTL0 and HCNTL1) determine whether the data is directed to the HPIA, the HPIC, or to memory. The DSP can interrupt the host with a dedicated $\overline{\text{HINT}}$ pin that the host can acknowledge and clear.

The ’VC5410 is the first device in the C5000 family in which the HPI8 can address all on-chip memory, including extended memory pages. Extended memory addresses are defined by a 23-bit address. The HPI8 sets the upper 6 bits of the extended memory address by writing a one to the XHPIA bit in HPIC, and then writing address bits A[22:16] into HPIA. The lower 16 bits of the extended memory address are set by writing a zero to XHPIA, followed by writing bits A[15:0] to HPIA. Similar to previous implementations of the HPI, after a write is performed to XHPIA or HPIA, a memory prefetch is initiated. The XHPIA bit is accessible only to the host. XHPIA is uninitialized following reset. The host should always initialize XHPIA prior to the first HPI8 access following a device reset.

The HPI8 interface has two data strobes ($\overline{\text{HDS1}}$ and $\overline{\text{HDS2}}$), a read/write strobe (HR/$\overline{\text{W}}$), and an address strobe ($\overline{\text{HAS}}$), to enable a glueless interface to a variety of industry-standard host devices. The HPI8 is easily interfaced to hosts with multiplexed address/data bus, separate address and data buses, one data strobe, and a read/write strobe, or two separate strobes for read and write.

enhanced host-port interface (HPI8) (continued)

All memory accesses on the ’VC5410 are in shared-access mode, meaning both the DSP and the host can access memory. Asynchronous host accesses are resynchronized internally, and in the event that the CPU and the host both request access to the same memory block, the host has access priority. The HRDY pin provides handshaking to the host during memory access.

The HPI8 also provides the capability to access memory during reset and power-down states. During reset, data or application code can be loaded via the HPI8, and the application can be initiated through the HPI option of the bootloader. During IDLE2/3 states, the HPI8 and the other six DMA channels continue to operate, and all pending DMA events complete before the DSP stops the clocks. The HPI8 has higher priority than the other six DMA channels. The HPI8 continues to have access to memory in IDLE2/3 even after the DSP has stopped the internal clocks as long as X2/CLKIN is maintained. The ’VC5410 HPI8 also remains active during emulation stop. The HPI8 can access any on-chip RAM on the device. The HPI8 memory map for the ’VC5410 is shown in Figure 7. The HPI8 determines memory location by address only (program or data space is not relevant).

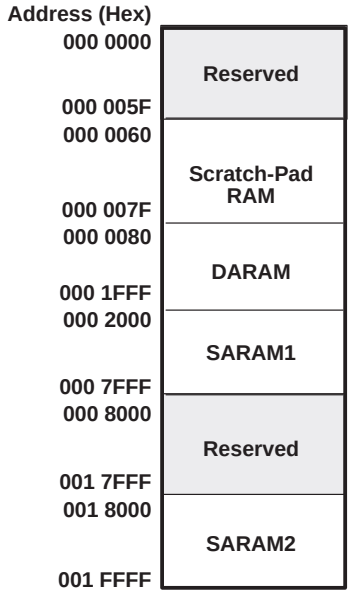


Figure 7. HPI8 Memory Map

multichannel buffered serial ports

The ’VC5410 device provides three high-speed, full-duplex, multichannel buffered serial ports that allow direct interface to other ’C54x/’LC54x devices, codecs, and other devices in a system. The McBSPs are based on the standard serial-port interface found on other ’54x devices. Like their predecessors, the McBSPs provide:

- Full-duplex communication
- Double-buffer data registers, which allow a continuous data stream
- Independent framing and clocking for receive and transmit

In addition, the McBSPs have the following capabilities:

- Direct interface to:
  - T1/E1 framers
  - MVIP switching compatible and ST-BUS compliant devices
  - IOM-2 compliant devices
  - AC97-compliant devices
  - IIS-compliant devices
  - Serial peripheral interface (SPI™)
- Multichannel transmit and receive of up to 128 channels
- A wide selection of data sizes, including 8, 12, 16, 20, 24, or 32 bits
- μ-law and A-law companding
- Programmable polarity for both frame synchronization and data clocks
- Programmable internal clock and frame generation

The McBSPs consist of separate transmit and receive channels that operate independently. The external interface of each McBSP consists of the following pins:

- BCLKX Transmit reference clock
- BDX Transmit data
- BFSX Transmit frame synchronization
- BCLKR Receive reference clock
- BDR Receive data
- BFSR Receive frame synchronization
- BCLKS External clock reference for the programmable clock generator

The first six pins listed are identical to the previous serial-port interface pins on the C5000 family of DSPs. The BCLKS pin is an additional signal to provide a clock reference to the McBSP programmable clock generator. As a compatibility option, the ’VC5410 is provided in a 144-pin TQFP package (designated PGE) that is pin-compatible with the ’C548/549 devices. BCLKS is not implemented on this package.

On the transmitter, transmit frame synchronization and clocking are indicated by the BFSX and BCLKX pins, respectively. The CPU or DMA can initiate transmission of data by writing to the data transmit register (DXR). Data written to DXR is shifted out on the BDX pin through a transmit shift register (XSR). This structure allows DXR to be loaded with the next word to be sent while the transmission of the current word is in progress.

On the receiver, receive frame synchronization and clocking are indicated by the BFSR and BCLKR pins respectively. The CPU or DMA can read received data from the data receive register (DRR). Data received on the BDR pin is shifted into a receive shift register (RSR) and then buffered in the receive buffer register (RBR). If DRR is empty, the RBR contents are copied into DRR. If not, RBR holds the data until DRR is available. This structure allows storage of the two previous words while the reception of the current word is in progress.

SPI is a trademark of Motorola Incorporated.

multichannel buffered serial ports (continued)

The CPU and DMA can move data to and from the McBSPs and can synchronize transfers based on McBSP interrupts, event signals, and status flags. The DMA is capable of handling data movement between the McBSPs and memory with no intervention from the CPU.

In addition to the standard serial-port functions, the McBSP provides programmable clock and frame synchronization generation. Among the programmable functions are:

- Frame synchronization pulse width
- Frame period
- Frame synchronization delay
- Clock reference (internal vs. external)
- Clock division
- Clock and frame synchronization polarity

The on-chip companding hardware allows compression and expansion of data in either μ-law or A-law format. When companding is used, transmit data is encoded according to specified companding law and received data is decoded to 2s complement format.

The McBSP allows the multiple channels to be independently selected for the transmitter and receiver. When the multiple channels are selected, each frame represents a time-division multiplexed (TDM) data stream. In using TDM data streams, the CPU may only need to process a few of them. Thus, to save memory and bus bandwidth, multichannel selection allows independent enabling of particular channels for transmission and reception. Up to 32 channels in a bit stream of up to 128 channels can be enabled.

The clock-stop mode (CLKSTP) in the McBSP provides compatibility with the serial peripheral interface (SPI) protocol. Clock-stop mode works with only single-phase frames and one word per frame. The word sizes supported by the McBSP are programmable for 8-, 12-, 16-, 20-, 24-, or 32-bit operation. When the McBSP is configured to operate in SPI mode, both the transmitter and the receiver operate together as a master or as a slave.

The McBSP is fully static and operates at arbitrarily low clock frequencies. The maximum frequency is CPU clock frequency divided by 2.

hardware timer

The ’VC5410 device features a 16-bit timing circuit with a 4-bit prescaler. The timer counter is decremented by one every CPU clock cycle. Each time the counter decrements to 0, a timer interrupt is generated. The timer can be stopped, restarted, reset, or disabled by specific status bits.

clock generator

The clock generator provides clocks to the ’VC5410 device, and consists of an internal oscillator and a phase-locked loop (PLL) circuit. The clock generator requires a reference clock input, which can be provided by using a crystal resonator with the internal oscillator, or from an external clock source. The reference clock input is then divided by two (DIV mode) to generate clocks for the ’VC5410 device, or the PLL circuit can be used (PLL mode) to generate the device clock by multiplying the reference clock frequency by a scale factor, allowing use of a clock source with a lower frequency than that of the CPU.The PLL is an adaptive circuit that, once synchronized, locks onto and tracks an input clock signal.

When the PLL is initially started, it enters a transitional mode during which the PLL acquires lock with the input signal. Once the PLL is locked, it continues to track and maintain synchronization with the input signal. Then, other internal clock circuitry allows the synthesis of new clock frequencies for use as master clock for the ’VC5410 device.

clock generator (continued)

This clock generator allows system designers to select the clock source. The sources that drive the clock generator are:

- A crystal resonator circuit. The crystal resonator circuit is connected across the X1 and X2/CLKIN pins of the ’VC5410 to enable the internal oscillator.
- An external clock. The external clock source is directly connected to the X2/CLKIN pin, and X1 is left unconnected.

The software-programmable PLL features a high level of flexibility, and includes a clock scaler that provides various clock multiplier ratios, capability to directly enable and disable the PLL, and a PLL lock timer that can be used to delay switching to PLL clocking mode of the device until lock is achieved.Devices that have a built-in software-programmable PLL can be configured in one of two clock modes:

- PLL mode. The input clock (X2/CLKIN) is multiplied by 1 of 31 possible ratios.
- DIV (divider) mode. The input clock is divided by 2 or 4. Note that when DIV mode is used, the PLL can be completely disabled in order to minimize power dissipation.

The software-programmable PLL is controlled using the 16-bit memory-mapped (address 0058h) clock mode register (CLKMD). The CLKMD register is used to define the clock configuration of the PLL clock module. Note that upon reset, the CLKMD register is initialized with a predetermined value dependent only upon the state of the CLKMD1 – CLKMD3 pins. The CLKMD pin configured clock options are shown in Table 5.

Table 5. CLKMD Pin Configured Clock Options

CLKMD1	CLKMD2	CLKMD3	CLKMD REGISTER RESET VALUE	CLOCK MODE
0	0	0	0000h	Divide-by-2, with external source
0	0	1	1000h	Divide-by-2, with external source
0	1	0	2000h	Divide-by-2, with external source
0	1	1	–	Stop mode
1	0	0	4000h	Divide-by-2, internal oscillator enabled
1	0	1	0007h	PLLx1 with external source
1	1	0	6000h	Divide-by-2, with external source
1	1	1	7000h	Reserved

enhanced external parallel interface (XIO2)

The ’VC5410 external interface has been redesigned to include several improvements, including: simplification of the bus sequence, more immunity to bus contention when transitioning between read and write operation, the ability for external memory access to the DMA controller, and optimization of the power-down modes.

The bus sequence on the ’VC5410 still maintains all of the same interface signals as on previous ’54x devices, but the signal sequence has been simplified. Most external accesses now require 3 cycles composed of a leading cycle, an active (read or write) cycle, and a trailing cycle. The leading and trailing cycles provide additional immunity against bus contention when switching between read operations and write operations. To maintain high-speed read access, a consecutive read mode that performs single-cycle reads as on previous ’54x devices is available.

enhanced external parallel interface (XIO2) (continued)

Figure 8 shows the bus sequence for three cases: all I/O reads, memory reads in nonconsecutive mode, or single memory reads in consecutive mode. The accesses shown in Figure 8 always require 3 CLKOUT cycles to complete.

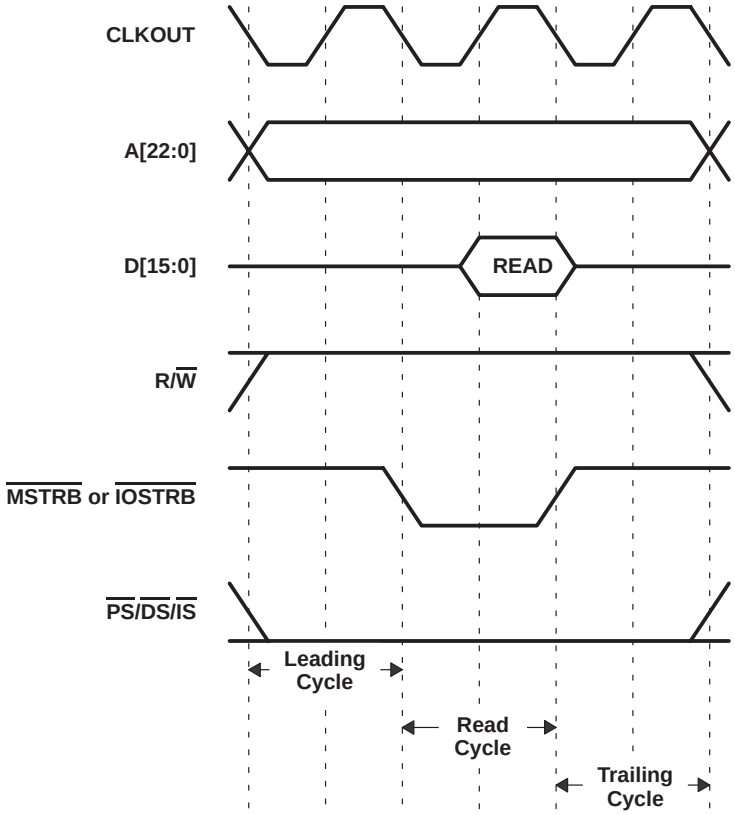


Figure 8. Nonconsecutive Memory Read and I/O Read Bus Sequence

enhanced external parallel interface (XIO2) (continued)

Figure 9 shows the bus sequence for repeated memory reads in consecutive mode. The accesses shown in Figure 9 require (2+n) CLKOUT cycles to complete, where n is the number of consecutive reads performed.

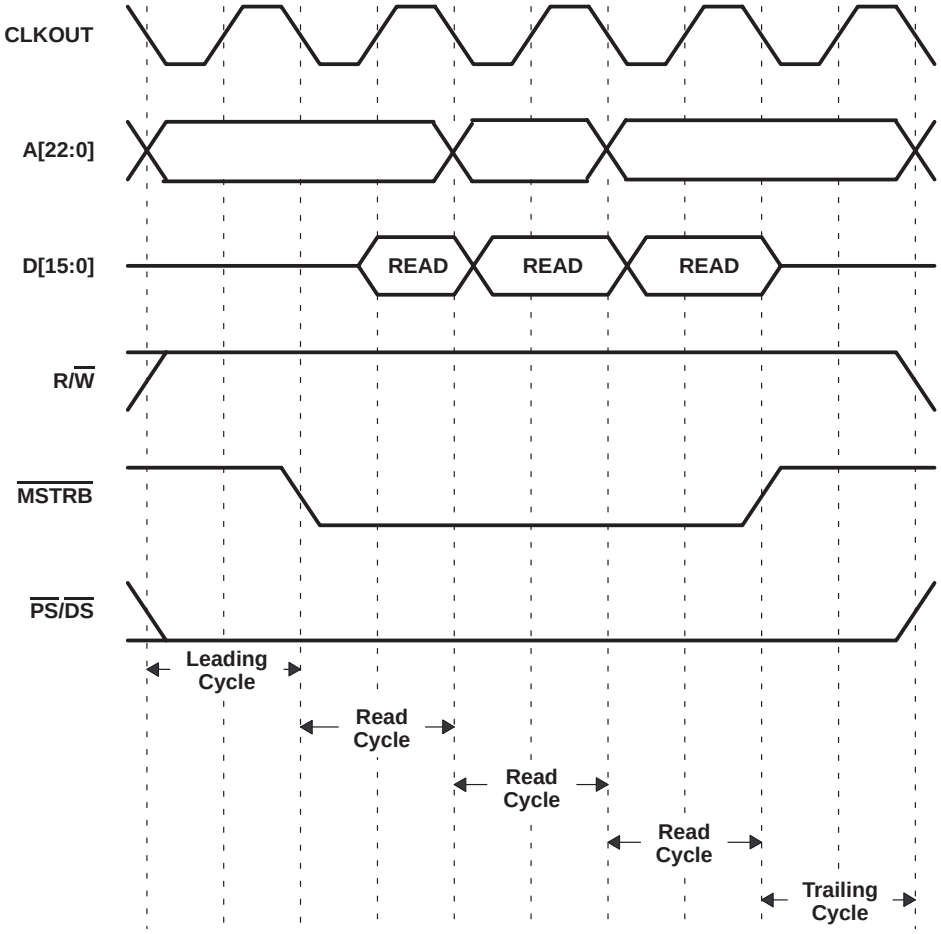


Figure 9. Consecutive Memory Read Bus Sequence (n = 3 reads)

enhanced external parallel interface (XIO2) (continued)

Figure 10 shows the bus sequence for all memory writes and I/O writes. The accesses shown in Figure 10 always require 3 CLKOUT cycles to complete.

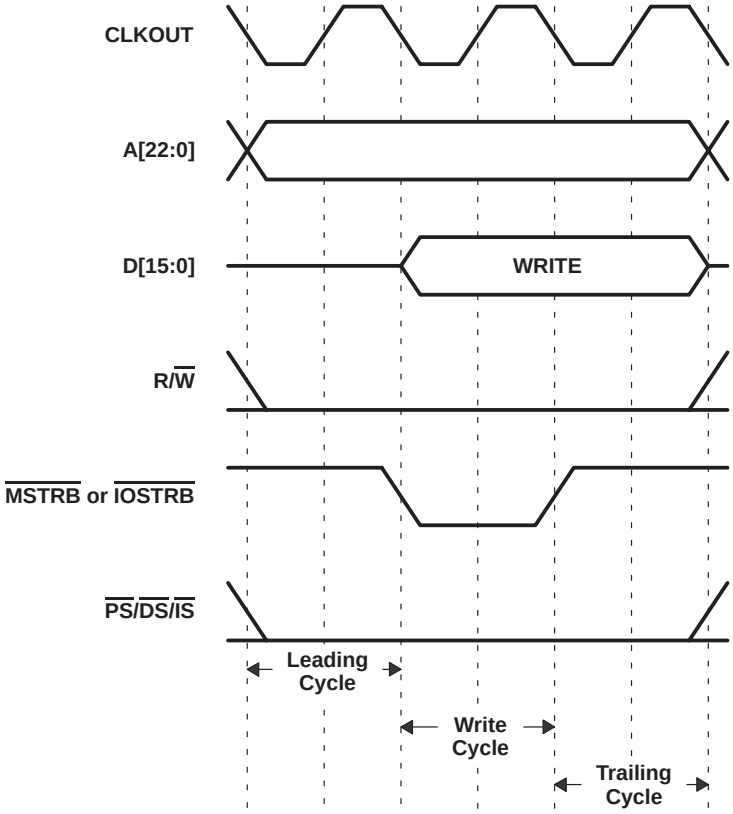


Figure 10. Memory Write and I/O Write Bus Sequence

The enhanced interface also provides the ability for DMA transfers to extend to external memory. For more information on DMA capability, see the DMA sections that follow.

The enhanced interface improves the low-power performance already present on the ’C5000 family by switching off the internal clocks to the interface when it is not being used. This power-saving feature is automatic, requires no software setup, and causes no latency in the operation of the interface.

Additional features integrated in the enhanced interface are the ability to automatically insert bank-switching cycles when crossing 32K memory boundaries (see the “programmable bank-switching” section), the ability to program up to 14 wait states through software (see the “software-programmable wait-state generator” section), and the ability to divide down CLKOUT by a factor of 1, 2, 3, or 4. Dividing down CLKOUT provides an alternative to wait states when interfacing to slower external memory or peripheral devices. While inserting wait states extends the bus sequence during read or write accesses, it does not slow down the bus signal sequences at the beginning and the end of the access. Dividing down CLKOUT provides a method of slowing the entire bus sequence when necessary. The CLKOUT divide-down factor is controlled through the DIVFCT field in the bank-switching control register (BSCR).

DMA controller

The ’VC5410 direct memory access (DMA) controller transfers data between points in the memory map without intervention by the CPU. The DMA allows movements of data to and from internal program/data memory, internal peripherals (such as the McBSPs), or external memory devices to occur in the background of CPU operation. The DMA has six independent programmable channels, allowing six different contexts for DMA operation.

features

The DMA has the following features:

- The DMA operates independently of the CPU.
- The DMA has six channels. The DMA can keep track of the contexts of six independent block transfers.
- The DMA has higher priority than the CPU for both internal and external accesses.
- Each channel has independently programmable priorities.
- Each channel’s source and destination address registers can have configurable indexes through memory on each read and write transfer, respectively. The address may remain constant, be post-incremented, be post-decremented, or be adjusted by a programmable value.
- Each read or write transfer may be initialized by selected events.
- On completion of a half- or entire-block transfer, each DMA channel may send an interrupt to the CPU.
- The DMA can perform double-word transfers (a 32-bit transfer of two 16-bit words).

DMA memory map

The DMA memory map, see Figure 11, allows the DMA transfer to be unaffected by the status of the MP/$\overline{MC}$, DROM, and OVLY bits.

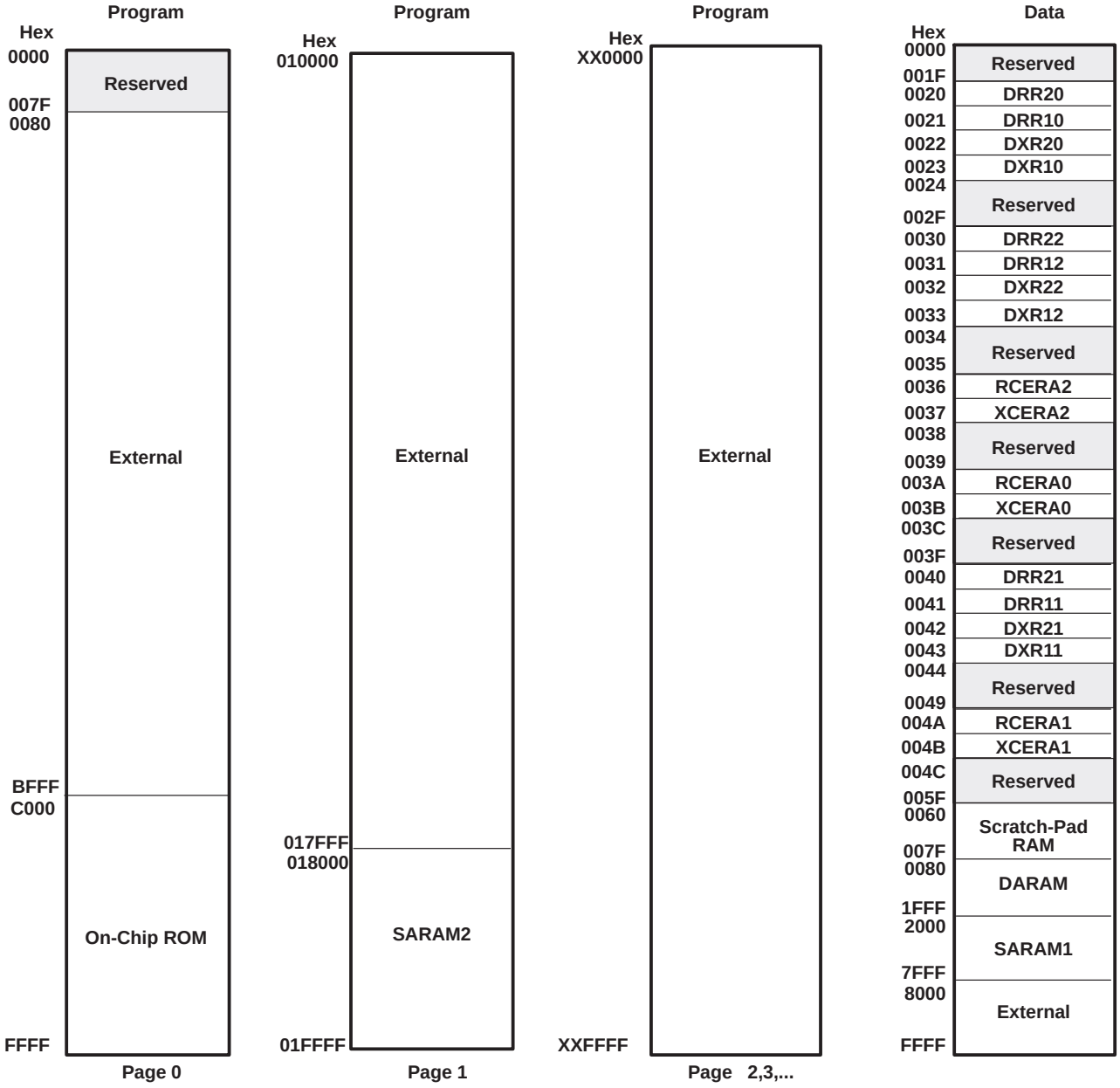


Figure 11. DMA Memory Map

DMA priority level

Each DMA channel can be independently assigned high- or low-priority relative to each other. Multiple DMA channels that are assigned to the same priority level are handled in a round-robin manner.

DMA source/destination address modification

The DMA provides flexible address-indexing modes for easy implementation of data management schemes such as autobuffering and circular buffers. Source and destination addresses can be indexed separately and can be post-incremented, post-decremented, or post-incremented with a specified index offset.

DMA in autoinitialization mode

The DMA can automatically reinitialize itself after completion of a block transfer. Some of the DMA registers can be preloaded for the next block transfer through the DMA global reload registers (DMGSA, DMGDA, and DMGCR). Autoinitialization allows:

- Continuous operation: Normally, the CPU would have to reinitialize the DMA immediately after the completion of the current block transfers, but with the global reload registers, it can reinitialize these values for the next block transfer any time after the current block transfer begins.
- Repetitive operation: The CPU does not preload the global reload register with new values for each block transfer but only loads them on the first block transfer.

DMA transfer counting

The DMA channel element count register (DMCTRx) and the frame count register (DMFRCx) contain bit fields that represent the number of frames and the number of elements per frame to be transferred.

- Frame count. This 8-bit value defines the total number of frames in the block transfer. The maximum number of frames per block transfer is 128 (FRAME COUNT= 0ffh). The counter is decremented upon the last read transfer in a frame transfer. Once the last frame is transferred, the selected 8-bit counter is reloaded with the DMA global frame reload register (DMGFR) if the AUTOINIT bit is set to 1. A frame count of 0 (default value) means the block transfer contains a single frame.
- Element count. This 16-bit value defines the number of elements per frame. This counter is decremented after the read transfer of each element. The maximum number of elements per frame is 65536 (DMCTRn = 0ffffh). In autoinitialization mode, once the last frame is transferred, the counter is reloaded with the DMA global count reload register (DMGCR).

DMA transfer in double-word mode

Double-word mode allows the DMA to transfer 32-bit words in any index mode. In double-word mode, two consecutive 16-bit transfers are initiated and the source and destination addresses are automatically updated following each transfer. In this mode, each 32-bit word is considered to be one element.

DMA channel index registers

The particular DMA channel index register is selected by way of the SIND and DIND fields in the DMA mode control register (DMMCRx). Unlike basic address adjustment, in conjunction with the frame index DMFRI0 and DMFRI1, the DMA allows different adjustment amounts depending on whether or not the element transfer is the last in the current frame. The normal adjustment value (element index) is contained in the element index registers DMIDX0 and DMIDX1. The adjustment value (frame index) for the end of the frame, is determined by the selected DMA frame index register, either DMFRI0 or DMFRI1.

The element index and the frame index affect address adjustment as follows:

- Element index: For all except the last transfer in the frame, the element index determines the amount to be added to the DMA channel for the source/destination address register (DMSRCx/DMDSTx) as selected by the SIND/DIND bits.
- Frame index: If the transfer is the last in a frame, frame index is used for address adjustment as selected by the SIND/DIND bits. This occurs in both single-frame and multi-frame transfers.

DMA interrupts

The ability of the DMA to interrupt the CPU based on the status of the data transfer is configurable and is determined by the IMOD and DINM bits in the DMA channel mode control register (DMMCRn). The available modes are shown in Table 6.

Table 6. DMA Interrupts

MODE	DINM	IMOD	INTERRUPT
ABU (non-decrement)	1	0	At full buffer only
ABU (non-decrement)	1	1	At half buffer and full buffer
Multi-Frame	1	0	At block transfer complete (DMCTRn = DMSEFCn[7:0] = 0)
Multi-Frame	1	1	At end of frame and end of block (DMCTRn = 0)
Either	0	X	No interrupt generated
Either	0	X	No interrupt generated

memory-mapped registers

The ’VC5410 has 27 memory-mapped CPU registers, which are mapped in data memory space address 0h to 1Fh. Each ’VC5410 device also has a set of memory-mapped registers associated with peripherals. Table 7 gives a list of CPU memory-mapped registers (MMRs) available on ’VC5410. Table 8 shows additional peripheral MMRs associated with the ’VC5410.

Table 7. CPU Memory-Mapped Registers

NAME	ADDRESS		DESCRIPTION
	DEC	HEX	
IMR	0	0	Interrupt mask register
IFR	1	1	Interrupt flag register
—	2–5	2–5	Reserved for testing
ST0	6	6	Status register 0
ST1	7	7	Status register 1
AL	8	8	Accumulator A low word (15–0)
AH	9	9	Accumulator A high word (31–16)
AG	10	A	Accumulator A guard bits (39–32)
BL	11	B	Accumulator B low word (15–0)
BH	12	C	Accumulator B high word (31–16)
BG	13	D	Accumulator B guard bits (39–32)
TREG	14	E	Temporary register
TRN	15	F	Transition register
AR0	16	10	Auxiliary register 0
AR1	17	11	Auxiliary register 1
AR2	18	12	Auxiliary register 2
AR3	19	13	Auxiliary register 3
AR4	20	14	Auxiliary register 4
AR5	21	15	Auxiliary register 5
AR6	22	16	Auxiliary register 6
AR7	23	17	Auxiliary register 7
SP	24	18	Stack pointer register
BK	25	19	Circular buffer size register
BRC	26	1A	Block repeat counter
RSA	27	1B	Block repeat start address
REA	28	1C	Block repeat end address
PMST	29	1D	Processor mode status (PMST) register
XPC	30	1E	Extended program page register
—	31	1F	Reserved

memory-mapped registers (continued)

Table 8. Peripheral Memory-Mapped Registers

NAME	ADDRESS	SUB-ADDRESS	DESCRIPTION	TYPE
DRR20	20h	—	McBSP0 data receive register	McBSP #0
DRR10	21h	—	McBSP0 data receive register	McBSP #0
DXR20	22h	—	McBSP0 data transmit register	McBSP #0
DXR10	23h	—	McBSP0 data transmit register	McBSP #0
TIM	24h	—	Timer register	Timer
PRD	25h	—	Timer period counter	Timer
TCR	26h	—	Timer control register	Timer
—	27h	—	Reserved	
SWWSR	28h	—	Software wait-state register	External Bus
BSCR	29h	—	Bank-switching control register	External Bus
—	2Ah	—	Reserved	
SWCR	2Bh	—	Software wait-state control register	External Bus
HPIC	2Ch	—	HPI control register	HPI
—	2Dh–2Fh	—	Reserved	
DRR22	30h	—	McBSP2 data receive register	McBSP #2
DRR12	31h	—	McBSP2 data receive register	McBSP #2
DXR22	32h	—	McBSP2 data transmit register	McBSP #2
DXR12	33h	—	McBSP2 data transmit register	McBSP #2
SPSA2	34h	—	McBSP2 sub-address register	McBSP #2
SPCR12	35h	00h	McBSP2 serial port control register 1	McBSP #2
SPCR22	35h	01h	McBSP2 serial port control register 2	McBSP #2
RCR12	35h	02h	McBSP2 receive control register 1	McBSP #2
RCR22	35h	03h	McBSP2 receive control register 2	McBSP #2
XCR12	35h	04h	McBSP2 transmit control register 1	McBSP #2
XCR22	35h	05h	McBSP2 transmit control register 2	McBSP #2
SRGR12	35h	06h	McBSP2 sample rate generator register 1	McBSP #2
SRGR22	35h	07h	McBSP2 sample rate generator register 2	McBSP #2
MCR12	35h	08h	McBSP2 multichannel register 1	McBSP #2
MCR22	35h	09h	McBSP2 multichannel register 2	McBSP #2
RCERA2	35h	0Ah	McBSP2 receive channel enable register partition A	McBSP #2
RCERB2	35h	0Bh	McBSP2 receive channel enable register partition B	McBSP #2
XCERA2	35h	0Ch	McBSP2 transmit channel enable register partition A	McBSP #2
XCERB2	35h	0Dh	McBSP2 transmit channel enable register partition B	McBSP #2
PCR2	35h	0Eh	McBSP2 pin control register	McBSP #2
—	36h–37h	—	Reserved	
SPSA0	38h	—	McBSP0 sub-address register	McBSP #0
SPCR10	39h	00h	McBSP0 serial port control register 1	McBSP #0
SPCR20	39h	01h	McBSP0 serial port control register 2	McBSP #0
RCR10	39h	02h	McBSP0 receive control register 1	McBSP #0
RCR20	39h	03h	McBSP0 receive control register 2	McBSP #0
XCR10	39h	04h	McBSP0 transmit control register 1	McBSP #0

† Accesses to address 56h update the sub-addressed register and post-increment the sub-address contained in DMSBAR. Accesses to 57h update the sub-addressed register without modifying DMSBAR.

memory-mapped registers (continued)

Table 8. Peripheral Memory-Mapped Registers (Continued)

NAME	ADDRESS	SUB-ADDRESS	DESCRIPTION	TYPE
XCR20	39h	05h	McBSP0 transmit control register 2	McBSP #0
SRGR10	39h	06h	McBSP0 sample rate generator register 1	McBSP #0
SRGR20	39h	07h	McBSP0 sample rate generator register 2	McBSP #0
MCR10	39h	08h	McBSP0 multichannel register 1	McBSP #0
MCR20	39h	09h	McBSP0 multichannel register 2	McBSP #0
RCERA0	39h	0Ah	McBSP0 receive channel enable register partition A	McBSP #0
RCERB0	39h	0Bh	McBSP0 receive channel enable register partition B	McBSP #0
XCERA0	39h	0Ch	McBSP0 transmit channel enable register partition A	McBSP #0
XCERB0	39h	0Dh	McBSP0 transmit channel enable register partition B	McBSP #0
PCR0	39h	0Eh	McBSP0 pin control register	McBSP #0
—	3Ah–3Fh	—	Reserved	
DRR21	40h	—	McBSP1 Data receive register 2	McBSP #1
DRR11	41h	—	McBSP1 Data receive register 1	McBSP #1
DXR21	42h	—	McBSP1 Data transmit register 2	McBSP #1
DXR11	43h	—	McBSP1 Data transmit register 1	McBSP #1
—	44h–47h	—	Reserved	
SPSA1	48h	—	McBSP1 sub-address register	McBSP #1
SPCR11	49h	00h	McBSP1 serial port control register 1	McBSP #1
SPCR21	49h	01h	McBSP1 serial port control register 2	McBSP #1
RCR11	49h	02h	McBSP1 receive control register 1	McBSP #1
RCR21	49h	03h	McBSP1 receive control register 2	McBSP #1
XCR11	49h	04h	McBSP1 transmit control register 1	McBSP #1
XCR21	49h	05h	McBSP1 transmit control register 2	McBSP #1
SRGR11	49h	06h	McBSP1 sample rate generator register 1	McBSP #1
SRGR21	49h	07h	McBSP1 sample rate generator register 2	McBSP #1
MCR11	49h	08h	McBSP1 multichannel register 1	McBSP #1
MCR21	49h	09h	McBSP1 multichannel register 2	McBSP #1
RCERA1	49h	0Ah	McBSP1 receive channel enable register partition A	McBSP #1
RCERB1	49h	0Bh	McBSP1 receive channel enable register partition B	McBSP #1
XCERA1	49h	0Ch	McBSP1 transmit channel enable register partition A	McBSP #1
XCERB1	49h	0Dh	McBSP1 transmit channel enable register partition B	McBSP #1
PCR1	49h	0Eh	McBSP1 pin control register	McBSP #1
—	4Ah–53h	—	Reserved	
DMPREC	54h	—	DMA channel priority and enable control register	DMA
DMSBAR	55h	—	DMA channel sub-address register	DMA
DMSRC0	56h/57h†	00h	DMA channel 0 source address register	DMA
DMDST0	56h/57h†	01h	DMA channel 0 destination address register	DMA
DMCTR0	56h/57h†	02h	DMA channel 0 element count register	DMA
DMSFC0	56h/57h†	03h	DMA channel 0 sync select and frame count register	DMA
DMMCR0	56h/57h†	04h	DMA channel 0 transfer mode control register	DMA
DMSRC1	56h/57h†	05h	DMA channel 1 source address register	DMA

† Accesses to address 56h update the sub-addressed register and post-increment the sub-address contained in DMSBAR. Accesses to 57h update the sub-addressed register without modifying DMSBAR.

memory-mapped registers (continued)

Table 8. Peripheral Memory-Mapped Registers (Continued)

NAME	ADDRESS	SUB-ADDRESS	DESCRIPTION	TYPE
DMDST1	56h/57h†	06h	DMA channel 1 destination address register	DMA
DMCTR1	56h/57h†	07h	DMA channel 1 element count register	DMA
DMSFC1	56h/57h†	08h	DMA channel 1 sync select and frame count register	DMA
DMMCR1	56h/57h†	09h	DMA channel 1 transfer mode control register	DMA
DMSRC2	56h/57h†	0Ah	DMA channel 2 source address register	DMA
DMDST2	56h/57h†	0Bh	DMA channel 2 destination address register	DMA
DMCTR2	56h/57h†	0Ch	DMA channel 2 element count register	DMA
DMSFC2	56h/57h†	0Dh	DMA channel 2 sync select and frame count register	DMA
DMMCR2	56h/57h†	0Eh	DMA channel 2 transfer mode control register	DMA
DMSRC3	56h/57h†	0Fh	DMA channel 3 source address register	DMA
DMDST3	56h/57h†	10h	DMA channel 3 destination address register	DMA
DMCTR3	56h/57h†	11h	DMA channel 3 element count register	DMA
DMSFC3	56h/57h†	12h	DMA channel 3 sync select and frame count register	DMA
DMMCR3	56h/57h†	13h	DMA channel 3 transfer mode control register	DMA
DMSRC4	56h/57h†	14h	DMA channel 4 source address register	DMA
DMDST4	56h/57h†	15h	DMA channel 4 destination address register	DMA
DMCTR4	56h/57h†	16h	DMA channel 4 element count register	DMA
DMSFC4	56h/57h†	17h	DMA channel 4 sync select and frame count register	DMA
DMMCR4	56h/57h†	18h	DMA channel 4 transfer mode control register	DMA
DMSRC5	56h/57h†	19h	DMA channel 5 source address register	DMA
DMDST5	56h/57h†	1Ah	DMA channel 5 destination address register	DMA
DMCTR5	56h/57h†	1Bh	DMA channel 5 element count register	DMA
DMSFC5	56h/57h†	1Ch	DMA channel 5 sync select and frame count register	DMA
DMMCR5	56h/57h†	1Dh	DMA channel 5 transfer mode control register	DMA
DMSRCP	56h/57h†	1Eh	DMA source program page address (common channel)	DMA
DMDSTP	56h/57h†	1Fh	DMA destination program page address (common channel)	DMA
DMIDX0	56h/57h†	20h	DMA element index address register 0	DMA
DMIDX1	56h/57h†	21h	DMA element index address register 1	DMA
DMFRI0	56h/57h†	22h	DMA frame index register 0	DMA
DMFRI1	56h/57h†	23h	DMA frame index register 1	DMA
DMGSA	56h/57h†	24h	DMA global source address reload register	DMA
DMGDA	56h/57h†	25h	DMA global destination address reload register	DMA
DMGCR	56h/57h†	26h	DMA global count reload register	DMA
DMGFR	56h/57h†	27h	DMA global frame count reload register	DMA
CLKMD	58h	—	Clock mode register	PLL
—	59h – 5Fh	—	Reserved	

† Accesses to address 56h update the sub-addressed register and post-increment the sub-address contained in DMSBAR. Accesses to 57h update the sub-addressed register without modifying DMSBAR.

interrupts

Vector-relative locations and priorities for all internal and external interrupts are shown in Table 9.

Table 9. Interrupt Locations and Priorities

NAME	LOCATION DECIMAL	LOCATION HEX	PRIORITY	FUNCTION
$\overline{RS}$, SINTR	0	00	1	Reset (hardware and software reset)
$\overline{NMI}$, SINT16	4	04	2	Nonmaskable interrupt
SINT17	8	08	—	Software interrupt #17
SINT18	12	0C	—	Software interrupt #18
SINT19	16	10	—	Software interrupt #19
SINT20	20	14	—	Software interrupt #20
SINT21	24	18	—	Software interrupt #21
SINT22	28	1C	—	Software interrupt #22
SINT23	32	20	—	Software interrupt #23
SINT24	36	24	—	Software interrupt #24
SINT25	40	28	—	Software interrupt #25
SINT26	44	2C	—	Software interrupt #26
SINT27	48	30	—	Software interrupt #27
SINT28	52	34	—	Software interrupt #28
SINT29	56	38	—	Software interrupt #29
SINT30	60	3C	—	Software interrupt #30
$\overline{INT0}$, SINT0	64	40	3	External user interrupt #0
$\overline{INT1}$, SINT1	68	44	4	External user interrupt #1
$\overline{INT2}$, SINT2	72	48	5	External user interrupt #2
TINT, SINT3	76	4C	6	Timer interrupt
RINT0, SINT4	80	50	7	McBSP #0 receive interrupt (default)
XINT0, SINT5	84	54	8	McBSP #0 transmit interrupt (default)
RINT2, SINT6	88	58	9	McBSP #2 receive interrupt (default)
XINT2, SINT7	92	5C	10	McBSP #2 transmit interrupt (default)
$\overline{INT3}$, SINT8	96	60	11	External user interrupt #3
$\overline{HINT}$, SINT9	100	64	12	HPI interrupt
RINT1, SINT10	104	68	13	McBSP #1 receive interrupt (default)
XINT1, SINT11	108	6C	14	McBSP #1 transmit interrupt (default)
DMAC4,SINT12	112	70	15	DMA channel 4 (default)
DMAC5,SINT13	116	74	16	DMA channel 5 (default)
Reserved	120–127	78–7F	—	Reserved

The bit layout of the interrupt flag register (IFR) and the interrupt mask register (IMR) is shown in Figure 12.

15–14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES	DMAC5	DMAC4	XINT1	RINT1	HINT	$\overline{INT3}$	XINT2	RINT2	XINT0	RINT0	TINT	$\overline{INT2}$	$\overline{INT1}$	$\overline{INT0}$

Figure 12. IFR and IMR

documentation support

Extensive documentation supports all TMS320 family generations of devices from product announcement through applications development. The following types of documentation are available to support the design and use of the ’C5000 family of DSPs:

- *TMS320C5000 DSP Family Functional Overview* (literature number SPRU307)
- Device-specific data sheets (such as this document)
- Complete user’s guides
- Development support tools
- Hardware and software application reports

The four-volume *TMS320C54x DSP Reference Set* (literature number SPRU210) consists of:

- *Volume 1: CPU and Peripherals* (literature number SPRU131)
- *Volume 2: Mnemonic Instruction Set* (literature number SPRU172)
- *Volume 3: Algebraic Instruction Set* (literature number SPRU179)
- *Volume 4: Applications Guide* (literature number SPRU173)

The reference set describes in detail the ’54x TMS320 products currently available and the hardware and software applications, including algorithms, for fixed-point TMS320 devices.

For general background information on DSPs and TI devices, see the three-volume publication *Digital Signal Processing Applications with the TMS320 Family* (literature numbers SPRA012, SPRA016, and SPRA017).

A series of DSP textbooks is published by Prentice-Hall and John Wiley & Sons to support digital signal processing research and education. The TMS320 newsletter, *Details on Signal Processing*, is published quarterly and distributed to update TMS320 customers on product information.

Information regarding TI DSP products is also available on the Worldwide Web at *http://www.ti.com* uniform resource locator (URL).

absolute maximum ratings over specified temperature range (unless otherwise noted)†

Supply voltage I/O range, DV_{DD}‡ . . . . . –0.3 V to 4.6 V
Supply voltage core range, CV_{DD}‡ . . . . . –0.3 V to 3.75 V
Input voltage range . . . . . –0.3 V to 4.6 V
Output voltage range . . . . . –0.3 V to 4.6 V
Operating case temperature range, T_C . . . . . –40°C to 100°C
Storage temperature range, T_{stg} . . . . . –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.
‡ All voltage values are with respect to V_{SS}.

recommended operating conditions

			MIN	NOM	MAX	UNIT
DV_{DD}	Device supply voltage, I/O§		3	3.3	3.6	V
CV_{DD}	Device supply voltage, core§		2.4	2.5	2.75	V
V_{SS}	Supply voltage, GND			0		V
V_{IH}	High-level input voltage, I/O	TCK, DV_{DD} = 3.3 ± 0.3 V	3		DV_{DD} + 0.3	V
		$\overline{RS}$, $\overline{INTn}$, $\overline{NMI}$, X2/CLKIN, BCLKR0, BCLKR1, BCLKR2, BCLKX0, BCLKX1, BCLKX2, $\overline{BCLKS0}$, $\overline{BCLKS1}$, $\overline{BCLKS2}$, $\overline{HCS}$, $\overline{HDS1}$, $\overline{HDS2}$, HAS CLKMDn, DV_{DD} = 3.3 ± 0.3 V	2.5		DV_{DD} + 0.3	V
		All other inputs	2		DV_{DD} + 0.3	
V_{IL}	Low-level input voltage		–0.3		0.8	V
I_{OH}	High-level output current				–300	µA
I_{OL}	Low-level output current				1.5	mA
T_C	Operating case temperature		–40		100	°C

§ Texas Instrument DSPs do not require specific power sequencing between the core supply and the I/O supply. However, systems should be designed to ensure that neither supply is powered up for extended periods of time if the other supply is below the proper operating voltage. Excessive exposure to these conditions can adversely affect the long term reliability of the devices. System-level concerns such as bus contention may require supply sequencing to be implemented. In this case, the core supply should be powered up at the same time as or prior to the I/O buffers and then powered down after the I/O buffers.

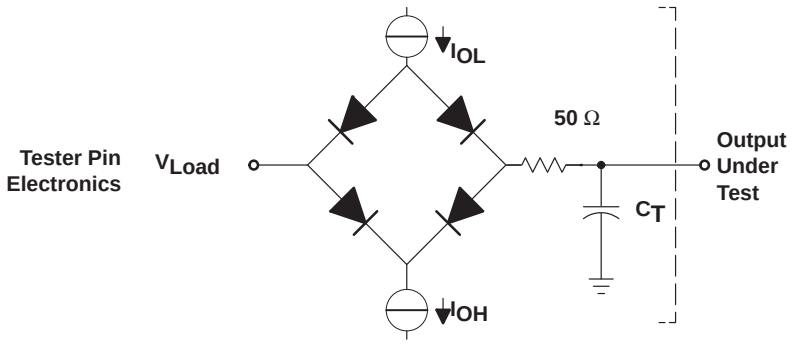
Refer to Figure 13 for 3.3-V device test load circuit values.

electrical characteristics over recommended operating case temperature range (unless otherwise noted)

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage‡		DV_{DD} = 3.3 ± 0.3 V, I_{OH} = MAX	2.4			V
V_{OL}	Low-level output voltage‡		I_{OL} = MAX			0.4	V
I_{IZ}	Input current in high impedance	A[22:0]	DV_{DD} = MAX, V_O = V_{SS} to DV_{DD}	–175		175	µA
I_I	Input current (V_I = V_{SS} to V_{DD})	$\overline{TRST}$	With internal pulldown	–10		800	µA
		HPIENA	With internal pulldown, $\overline{RS}$ = 0	–10		400	
		TMS, TCK, TDI, BCLKS0, BCLKS1, BCLKS2, HPI§	With internal pullups	–400		10	
		D[15:0], HD[7:0]	Bus holders enabled, DV_{DD} = MAX, V_I = DV_{SS} to DV_{DD}	–175		175	
		All other input-only pins		–10		10	
I_{DDC}	Supply current, core CPU		CV_{DD} = 2.5 V, 100 MHz CPU clock,¶ T_C = 25°C		47#		mA
I_{DDP}	Supply current, pins		DV_{DD} = 3.3 V, 100 MHz CPU clock, T_C = 25°C		22‖		mA
I_{DD}	Supply current, standby	IDLE2	PLL × 2 mode, 50 MHz input, T_C = 25°C		2		mA
		IDLE3	Divide-by-two mode, CLKIN stopped, T_C = 25°C		5		µA
C_i	Input capacitance				10		pF
C_o	Output capacitance				10		pF

† All values are typical unless otherwise specified.
‡ All input and output voltage levels except $\overline{RS}$, $\overline{INT0}$–$\overline{INT3}$, $\overline{NMI}$, X2/CLKIN, CLKMD0–CLKMD3 are LVTTL-compatible.
§ HPI input signals except for HPIENA.
¶ Clock mode: PLL × 2 with external source
# This value was obtained with 50% usage of MAC and 50% usage of NOP instructions. Actual operating current varies with program being executed.
‖ This value was obtained with continous external writes, CLKOFF = 0 and load = 15 pF. For more details on how this calculation is performed, refer to the *Calculation of TMS320LC54x Power Dissipation* application report (literature number SPRA164).

PARAMETER MEASUREMENT INFORMATION



Where: I_{OL} = 1.5 mA (all outputs)
I_{OH} = 300 μA (all outputs)
V_{Load} = 1.5 V
C_T = 40-pF typical load circuit capacitance

Figure 13. 3.3-V Test Load Circuit

internal divide-by-two clock option with external crystal

The internal oscillator on the ’5410 is enabled by setting the CLKMD(1,2,3) pins to (1,0,0) at reset and connecting a crystal or ceramic resonator across X1 and X2/CLKIN. The CPU clock frequency is one-half the crystal’s oscillation frequency following reset. Since the internal oscillator can be used as a clock source to the PLL, the crystal oscillation frequency can be multiplied to generate the CPU clock if desired.

The crystal should be in fundamental mode operation and parallel resonant with an effective series resistance of 30 ohms and power dissipation of 1 mW. The connection of the required circuit, consisting of the crystal and two load capacitors, is shown in Figure 14. The load capacitors, C_1 and C_2, should be chosen such that the equation below is satisfied. C_L in the equation is the load specified for the crystal.

$$C_L = \frac{C_1 C_2}{(C_1 + C_2)}$$

		’VC5410-100			’VC5410-120			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
f_X	Input clock frequency	0†		50‡	0†		50‡	MHz

† This device utilizes a fully static design and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz.

‡ It is recommended that the PLL clocking option be used for maximum frequency operation.

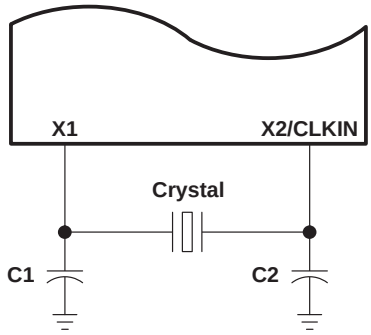


Figure 14. Internal Divide-by-Two Clock Option With External Crystal

external divide-by-two clock option

An external frequency source can be used by applying an input clock to X2/CLKIN with X1 left unconnected. Table 5 shows the configuration options for the CLKMD pins that generate the external divide-by-2 clock option. This external input clock frequency is divided by two to generate the CPU machine cycle.

The external frequency injected must conform to specifications listed in the timing requirements table.

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Figure 14, Figure 15, and the recommended operating conditions table)

PARAMETER		'VC5410-100			'VC5410-120			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT	10†	$2t_{c(CI)}$	‡	8.33†	$2t_{c(CI)}$	‡	ns
$t_{d(CIH-CO)}$	Delay time, X2/CLKIN high to CLKOUT high/low	3	6	10	3	6	10	ns
$t_{f(CO)}$	Fall time, CLKOUT		2			2		ns
$t_{r(CO)}$	Rise time, CLKOUT		2			2		ns
$t_{w(COL)}$	Pulse duration, CLKOUT low	H–2	H–1	H	H–2	H–1	H	ns
$t_{w(COH)}$	Pulse duration, CLKOUT high	H–2	H–1	H	H–2	H–1	H	ns

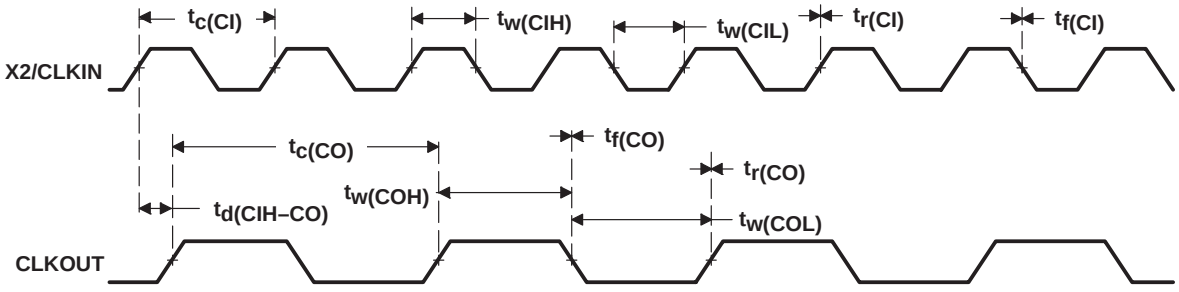
† It is recommended that the PLL clocking option be used for maximum frequency operation.
‡ This device utilizes a fully static design and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz.

external divide-by-two clock option (continued)

timing requirements (see Figure 15)

		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{c(CI)}$	Cycle time, X2/CLKIN	5	†	4.167	†	ns
$t_{f(CI)}$	Fall time, X2/CLKIN		1		1	ns
$t_{r(CI)}$	Rise time, X2/CLKIN		1		1	ns
$t_{w(CIL)}$	Pulse duration, X2/CLKIN low	2	†	2	†	ns
$t_{w(CIH)}$	Pulse duration, X2/CLKIN high	2	†	2	†	ns

† This device utilizes a fully static design and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz.



NOTE A: The CLKOUT timing in this diagram assumes the CLKOUT divide factor (DIVFCT field in the BSCR) is configured as 00 (CLKOUT not divided). DIVFCT is configured as CLKOUT divided-by-4 mode following reset.

Figure 15. External Divide-by-Two Clock Timing

external multiply-by-N clock option

An external frequency source can be used by applying an input clock to X2/CLKIN with X1 left unconnected. Figure 14 shows the configuration options for the CLKMD pins that generate the external divide-by-2 clock option. Following reset, the software PLL can be programmed for the desired multiplication factor. Refer to the *TMS320C54x DSP CPU and Peripherals Reference Set*, Volume 1 (literature number SPRU131) for detailed information on programming the PLL. The external input clock frequency is multiplied by the multiplication factor N to generate the internal CPU machine cycle.

The external frequency injected must conform to specifications listed in the timing requirements table.

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Figure 16 and the recommended operating conditions table)

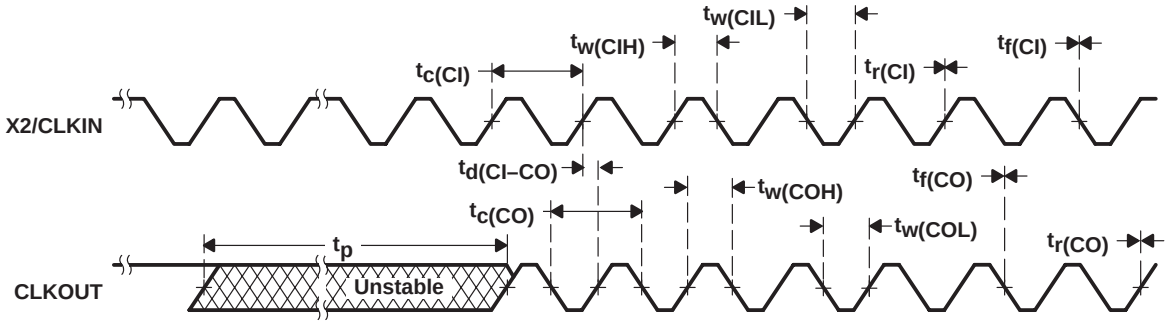
PARAMETER		’VC5410-100			’VC5410-120			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT	10	$t_{c(CI)/N}$†		8.33	$t_{c(CI)/N}$†		ns
$t_{d(CI-CO)}$	Delay time, X2/CLKIN high/low to CLKOUT high/low	3	6	10	3	6	10	ns
$t_{f(CO)}$	Fall time, CLKOUT		2			2		ns
$t_{r(CO)}$	Rise time, CLKOUT		2			2		ns
$t_{w(COL)}$	Pulse duration, CLKOUT low	H–2	H–1	H	H–2	H–1	H	ns
$t_{w(COH)}$	Pulse duration, CLKOUT high	H–2	H–1	H	H–2	H–1	H	ns
t_p	Transitory phase, PLL lock-up time			35			35	µs

† N is the multiplication factor.

timing requirements† (see Figure 16)

			’VC5410-100		’VC5410-120		UNIT
			MIN	MAX	MIN	MAX	
$t_{c(CI)}$	Cycle time, X2/CLKIN	Integer PLL multiplier N (N = 1–15)	10N	400N	8.33N	400N	ns
		PLL multiplier N = x.5	10N	200N	8.33N	200N	
		PLL multiplier N = x.25, x.75	10N	100N	8.33N	100N	
$t_{f(CI)}$	Fall time, X2/CLKIN			4		4	ns
$t_{r(CI)}$	Rise time, X2/CLKIN			4		4	ns
$t_{w(CIL)}$	Pulse duration, X2/CLKIN low		2		2		ns
$t_{w(CIH)}$	Pulse duration, X2/CLKIN high		2		2		ns

† N is the multiplication factor.



NOTE A: The CLKOUT timing in this diagram assumes the CLKOUT divide factor (DIVFCT field in the BSCR) is configured as 00 (CLKOUT not divided). DIVFCT is configured as CLKOUT divided-by-4 mode following reset.

Figure 16. External Multiply-by-One Clock Timing

memory and parallel I/O interface timing

memory read

External memory reads can be performed in consecutive or nonconsecutive mode under control of the CONSEC bit in the BSCR.

switching characteristics over recommended operating conditions ($\overline{\text{MSTRB}}$ = 0)† (see Figure 17 and Figure 18)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(CLKL-A)}$	Delay time, CLKOUT low to address valid	– 1	4	– 1	6	ns
$t_{d(CLKL-MSL)}$	Delay time, CLKOUT low to $\overline{\text{MSTRB}}$ low	– 1	4	– 1	6	ns
$t_{d(CLKL-MSH)}$	Delay time, CLKOUT low to $\overline{\text{MSTRB}}$ high	– 1	4	– 1	6	ns

† Address,R/$\overline{W}$, $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ timings are all included in timings referenced as address.

timing requirements ($\overline{\text{MSTRB}}$ = 0) [H = 0.5 $t_{c(CO)}$]† (see Figure 17 and Figure 18)

		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{a(A)M1}$	Access time, read data access from address valid, first read access		4H–10		4H–10	ns
$t_{a(A)M2}$	Access time, read data access from address valid, consecutive read accesses		2H–10		2H–10	ns
$t_{su(D)R}$	Setup time, read data valid before CLKOUT low	6		6		ns
$t_{h(D)R}$	Hold time, read data valid after CLKOUT low	0		0		ns

† Address,R/$\overline{W}$, $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ timings are all included in timings referenced as address.

memory and parallel I/O interface timing (continued)

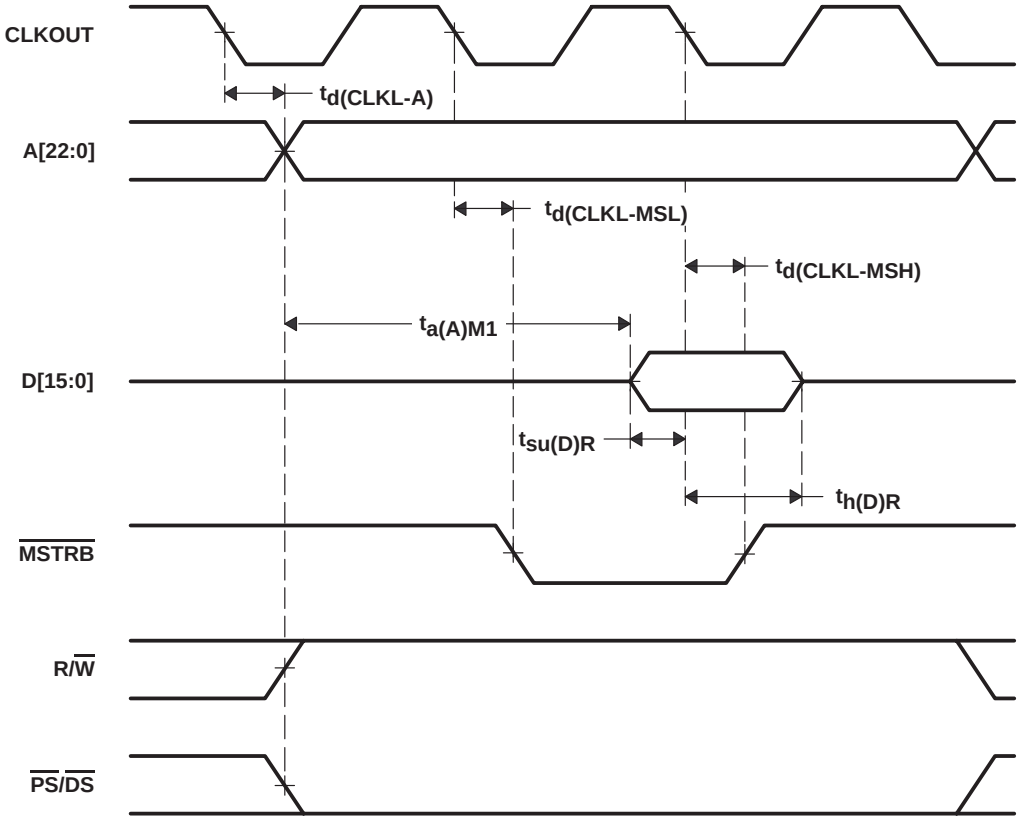


Figure 17. Nonconsecutive Mode Memory Reads

memory and parallel I/O interface timing (continued)

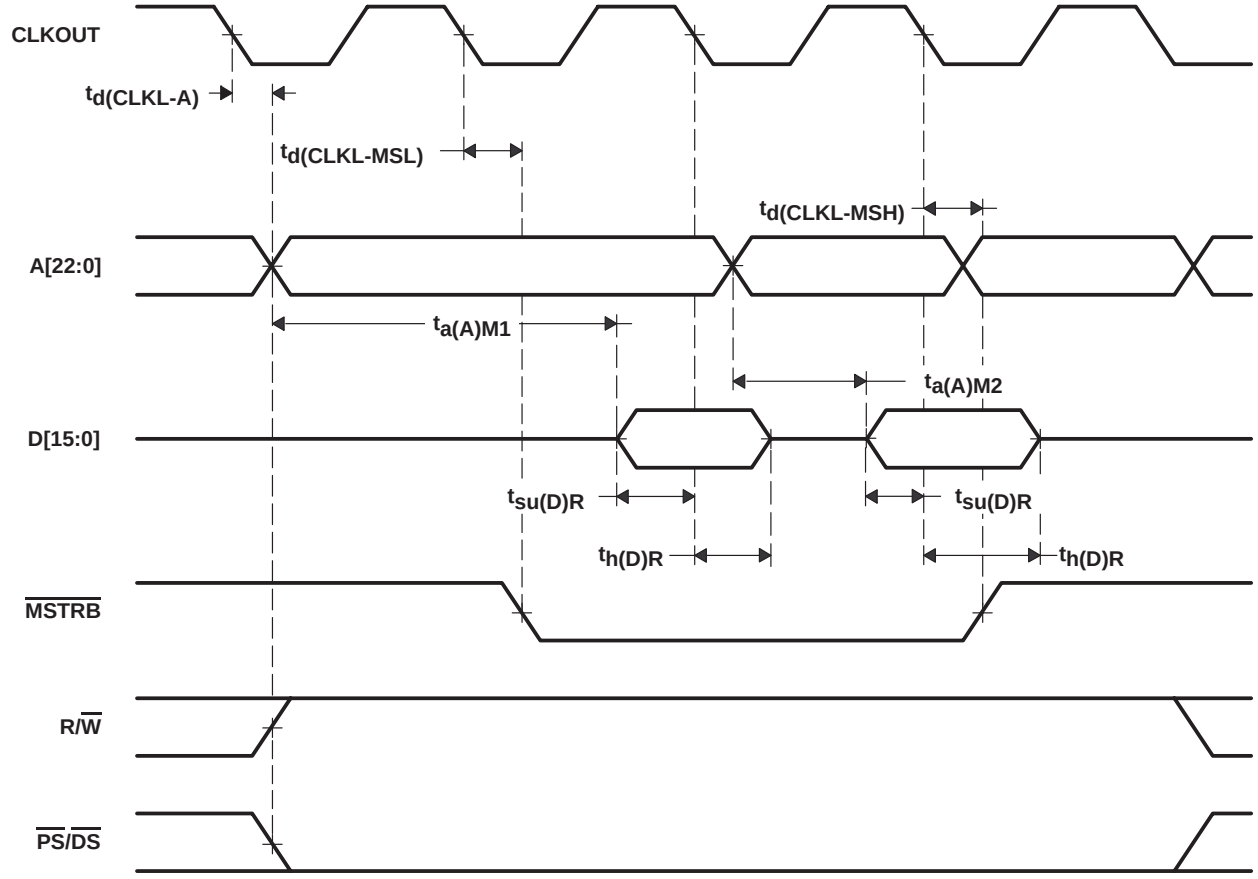


Figure 18. Consecutive Mode Memory Reads

memory and parallel I/O interface timing (continued)

memory write

switching characteristics over recommended operating conditions ($\overline{\text{MSTRB}}$ = 0) [H = 0.5 $t_{c(CO)}$]† (see Figure 19)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(CLKL-A)}$	Delay time, CLKOUT low to address valid	– 1	4	– 1	6	ns
$t_{su(A)MSL}$	Setup time, address valid before $\overline{\text{MSTRB}}$ low	2H – 5		2H – 5		ns
$t_{d(CLKL-D)W}$	Delay time, CLKOUT low to data valid	0	5	0	7	ns
$t_{su(D)MSH}$	Setup time, data valid before $\overline{\text{MSTRB}}$ high	2H – 5	2H + 5	2H – 5		ns
$t_{h(D)MSH}$	Hold time, data valid after $\overline{\text{MSTRB}}$ high	2H – 5	2H + 5	2H – 5		ns
$t_{d(CLKL-MSL)}$	Delay time, CLKOUT low to $\overline{\text{MSTRB}}$ low	– 1	4	– 1	6	ns
$t_{w(SL)MS}$	Pulse duration, $\overline{\text{MSTRB}}$ low	2H – 5		2H – 5		ns
$t_{d(CLKL-MSH)}$	Delay time, CLKOUT low to $\overline{\text{MSTRB}}$ high	– 1	4	– 1	6	ns

† Address, R/$\overline{\text{W}}$, $\overline{\text{PS}}$, $\overline{\text{DS}}$, and $\overline{\text{IS}}$ timings are all included in timings referenced as address.

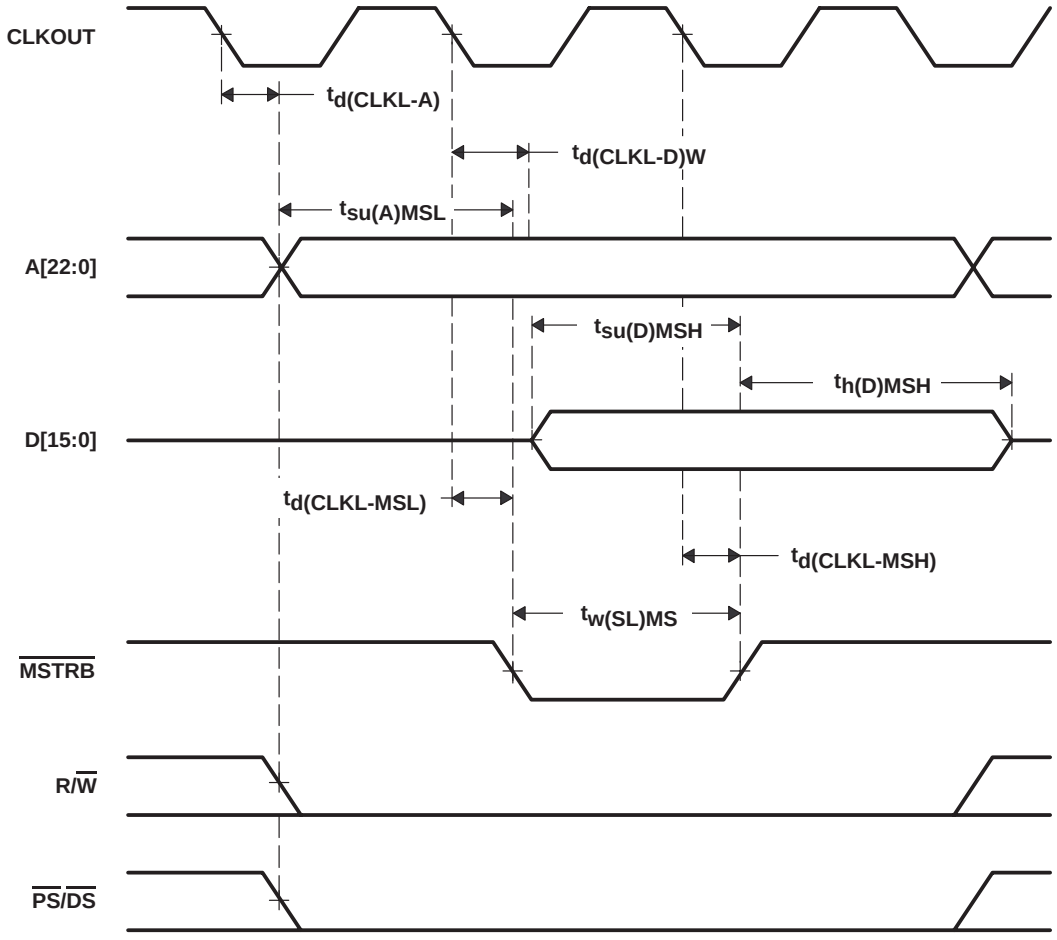


Figure 19. Memory Write ($\overline{\text{MSTRB}}$ = 0)

memory and parallel I/O interface timing (continued)

I/O read

switching characteristics over recommended operating conditions ($\overline{\text{IOSTRB}}$ = 0)† (see Figure 20)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(CLKL-A)}$	Delay time, CLKOUT low to address valid	– 1	4	– 1	6	ns
$t_{d(CLKL-IOSL)}$	Delay time, CLKOUT low to $\overline{\text{IOSTRB}}$ low	– 1	4	– 1	6	ns
$t_{d(CLKL-IOSH)}$	Delay time, CLKOUT low to $\overline{\text{IOSTRB}}$ high	– 1	4	– 1	6	ns

† Address R/$\overline{W}$, $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ timings are included in timings referenced as address.

timing requirements for a parallel I/O port read ($\overline{\text{IOSTRB}}$ = 0) [H = 0.5 $t_{c(CO)}$]† (see Figure 20)

		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{a(A)M1}$	Access time, read data access from address valid, first read access		4H–10		4H–10	ns
$t_{su(D)R}$	Setup time, read data valid before CLKOUT low	6		6		ns
$t_{h(D)R}$	Hold time, read data valid after CLKOUT low	0		0		ns

† Address R/$\overline{W}$, $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ timings are included in timings referenced as address.

memory and parallel I/O interface timing (continued)

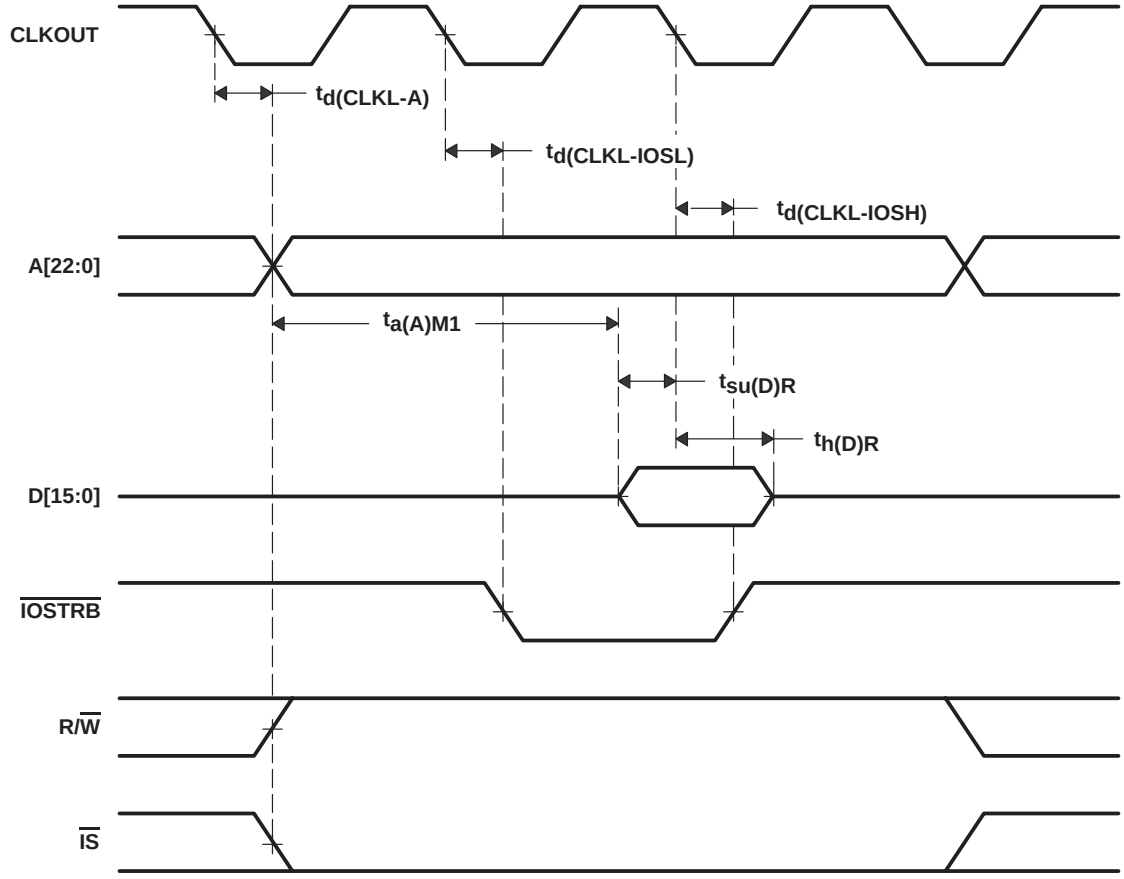


Figure 20. Parallel I/O Port Read ($\overline{\text{IOSTRB}}$ = 0)

memory and parallel I/O interface timing (continued)

I/O write

switching characteristics over recommended operating conditions ($\overline{IOSTRB}$ = 0) [H = 0.5 $t_{c(CO)}$] (see Figure 21)†

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(CLKL-A)}$	Delay time, CLKOUT low to address valid	– 1	4	– 1	6	ns
$t_{su(A)IOSL}$	Setup time, address valid before $\overline{IOSTRB}$ low	2H – 5		2H – 5		ns
$t_{d(CLKL-D)W}$	Delay time, CLKOUT low to write data valid	0	5	0	7	ns
$t_{su(D)IOSH}$	Setup time, data valid before $\overline{IOSTRB}$ high	2H – 5	2H + 5	2H – 5	2H + 5	ns
$t_{h(D)IOSH}$	Hold time, data valid after $\overline{IOSTRB}$ high	2H – 5	2H + 5	2H – 5	2H + 5	ns
$t_{d(CLKL-IOSL)}$	Delay time, CLKOUT low to $\overline{IOSTRB}$ low	– 1	4	– 1	6	ns
$t_{w(SL)IOS}$	Pulse duration, $\overline{IOSTRB}$ low	2H – 5		2H – 5		ns
$t_{d(CLKL-IOSH)}$	Delay time, CLKOUT low to $\overline{IOSTRB}$ high	– 1	4	– 1	6	ns

† Address R/$\overline{W}$, $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ timings are included in timings referenced as address.

memory and parallel I/O interface timing (continued)

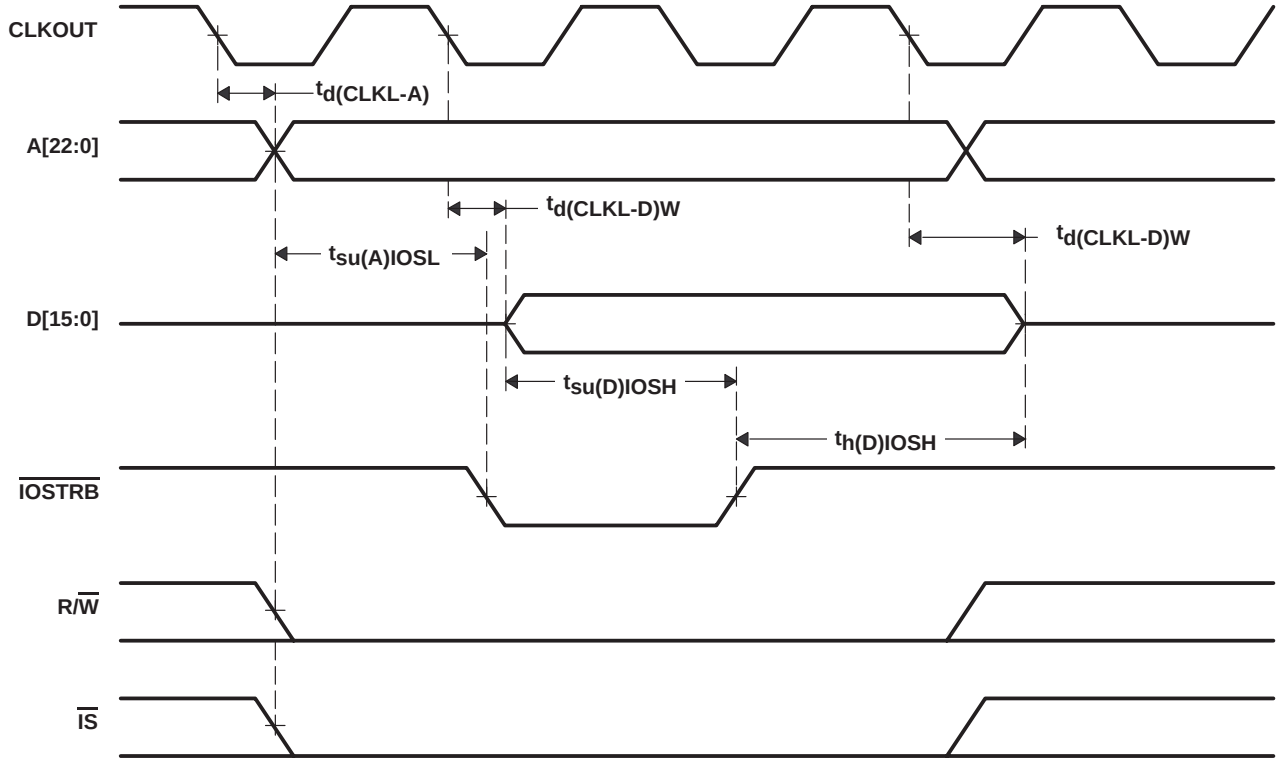


Figure 21. Parallel I/O Port Write ($\overline{\text{IOSTRB}}$ = 0)

ready timing for externally generated wait states

switching characteristics over recommended operating conditions[†‡] (see Figure 22, Figure 23, Figure 24, and Figure 25)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(MSCL)}$	Delay time, CLKOUT low to $\overline{MSC}$ low	– 1	4	– 1	6	ns
$t_{d(MSCH)}$	Delay time, CLKOUT low to $\overline{MSC}$ high	– 1	4	– 1	6	ns

† The hardware wait states can be used only in conjunction with the software wait states to extend the bus cycles. To generate wait states by READY, at least two software wait states must be programmed. READY is not sampled until the completion of the internal software wait states.
‡ These timings are included for reference only. The critical timings for READY are those referenced to CLKOUT.

timing requirements for externally generated wait states [H = 0.5 $t_{c(CO)}$][†] (see Figure 22, Figure 23, Figure 24, and Figure 25)

		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(RDY)}$	Setup time, READY before CLKOUT low	5		5		ns
$t_{h(RDY)}$	Hold time, READY after CLKOUT low	0		0		ns
$t_{v(RDY)MSTRB}$	Valid time, READY after $\overline{MSTRB}$ low[‡]		4H–8		4H–8	ns
$t_{h(RDY)MSTRB}$	Hold time, READY after $\overline{MSTRB}$ low[‡]	4H		4H		ns
$t_{v(RDY)IOSTRB}$	Valid time, READY after $\overline{IOSTRB}$ low[‡]		4H–8		4H–8	ns
$t_{h(RDY)IOSTRB}$	Hold time, READY after $\overline{IOSTRB}$ low[‡]	4H		4H		ns

† The hardware wait states can be used only in conjunction with the software wait states to extend the bus cycles. To generate wait states by READY, at least two software wait states must be programmed. READY is not sampled until the completion of the internal software wait states.
‡ These timings are included for reference only. The critical timings for READY are those referenced to CLKOUT.

ready timing for externally generated wait states (continued)

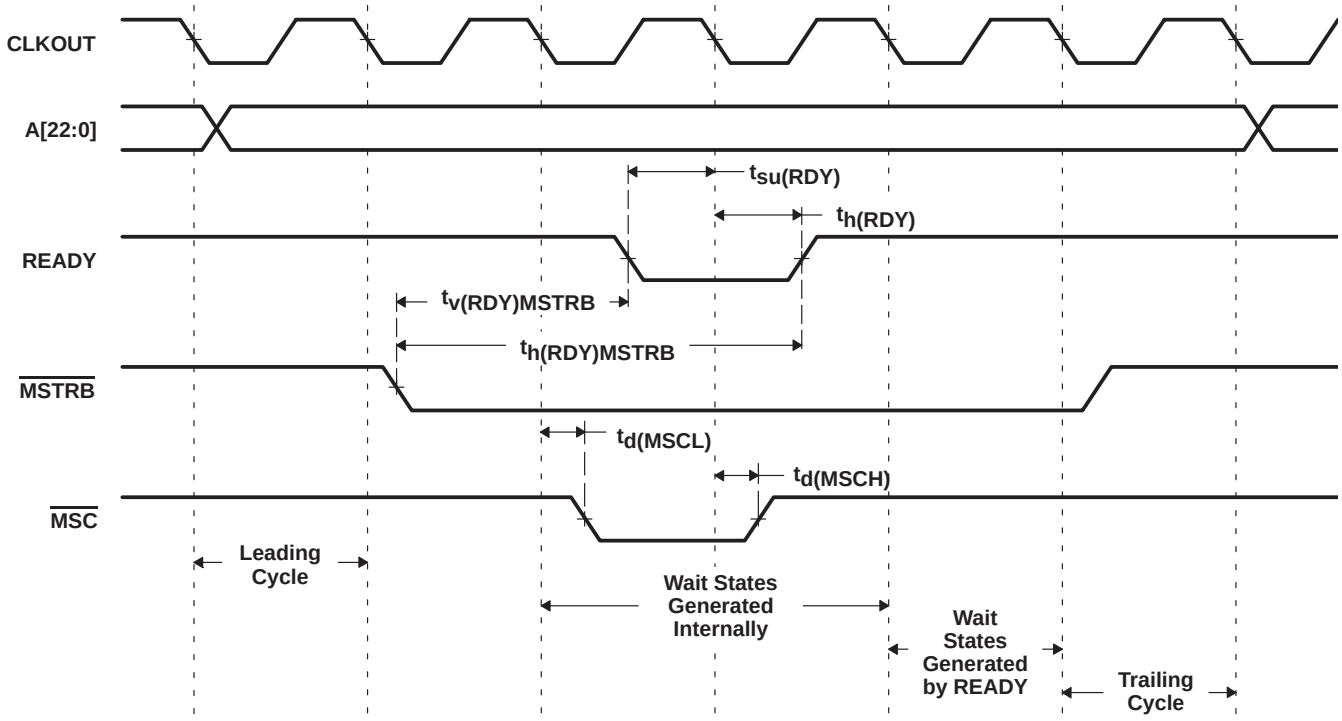


Figure 22. Memory Read With Externally Generated Wait States

ready timing for externally generated wait states (continued)

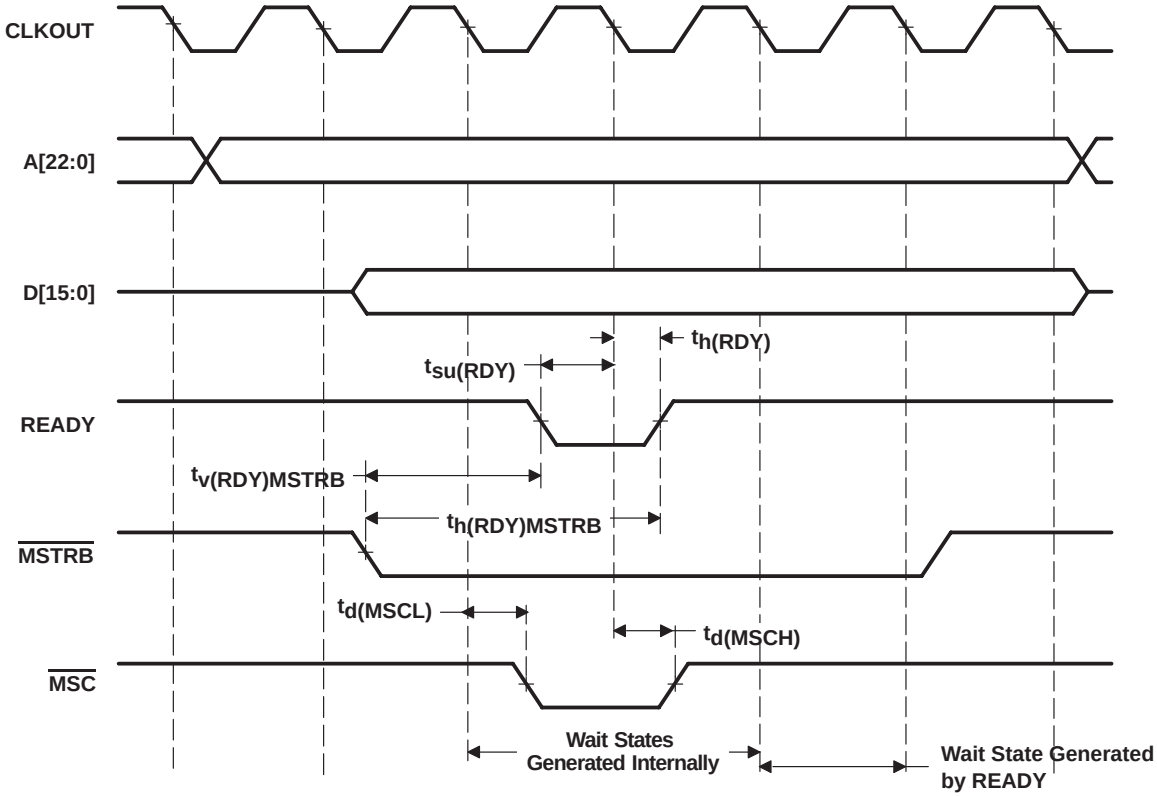


Figure 23. Memory Write With Externally Generated Wait States

ready timing for externally generated wait states (continued)

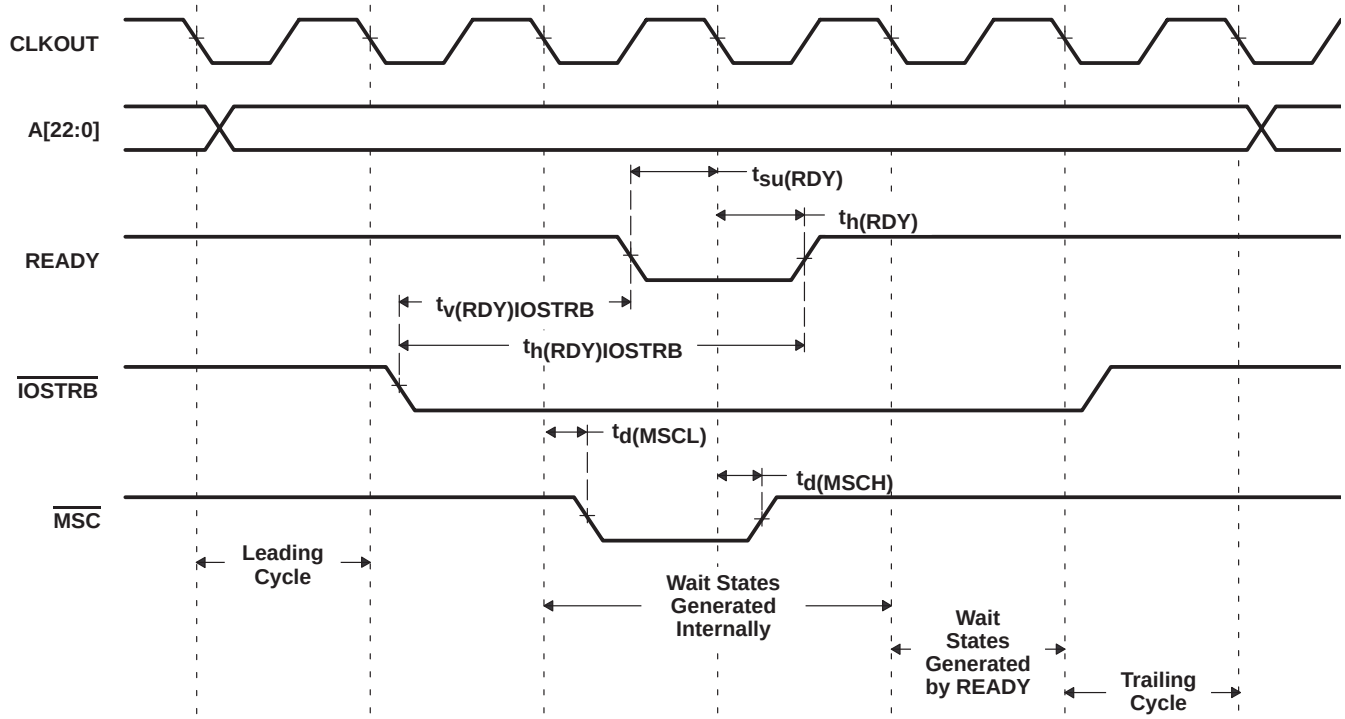


Figure 24. I/O Read With Externally Generated Wait States

ready timing for externally generated wait states (continued)

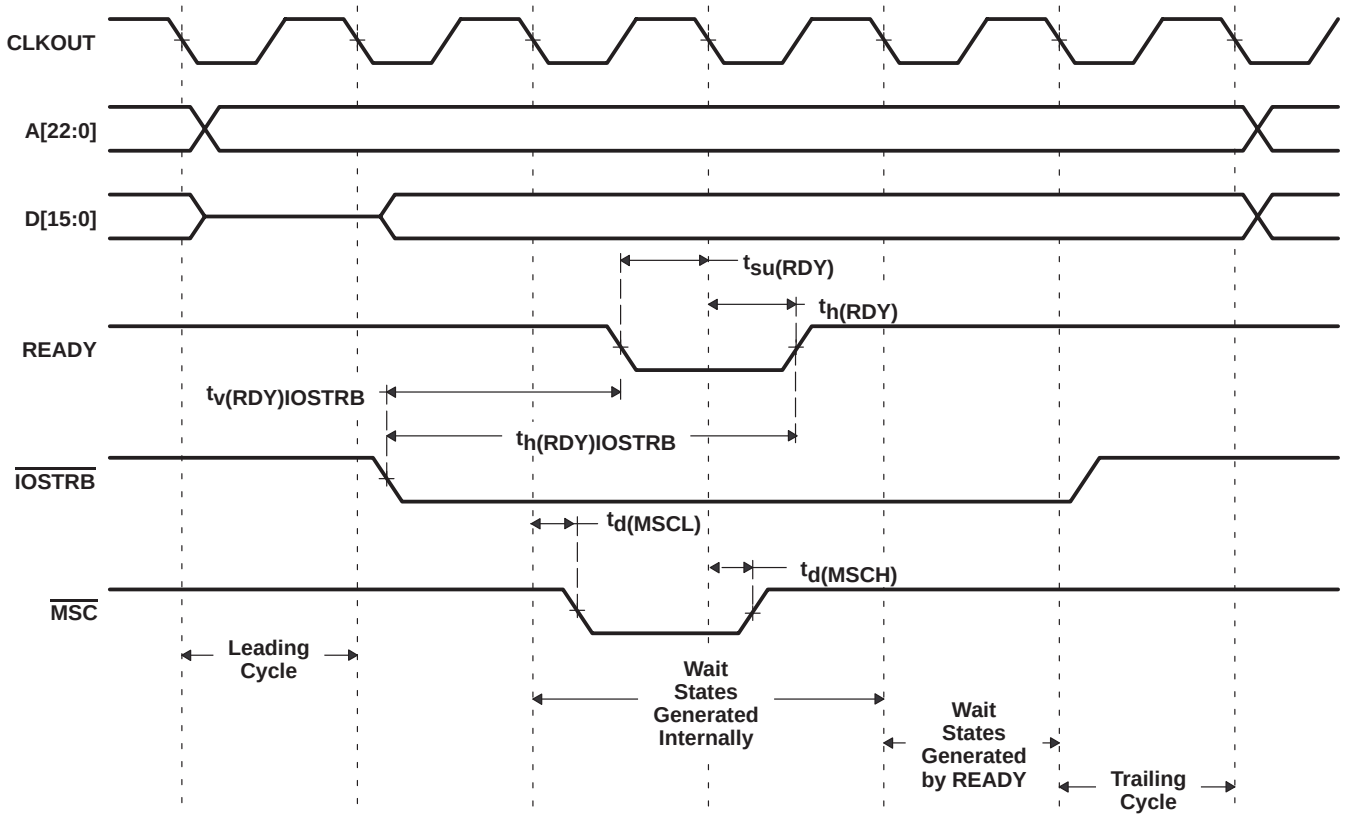


Figure 25. I/O Write With Externally Generated Wait States

$\overline{\text{HOLD}}$ and $\overline{\text{HOLDA}}$ timings

switching characteristics over recommended operating conditions for memory control signals and $\overline{\text{HOLDA}}$ [H = 0.5 $t_{c(CO)}$] (see Figure 26)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{dis(CLKL-A)}$	Disable time, Address, $\overline{\text{PS}}$, $\overline{\text{DS}}$, $\overline{\text{IS}}$ high impedance from CLKOUT low		5		5	ns
$t_{dis(CLKL-RW)}$	Disable time, R/$\overline{\text{W}}$ high impedance from CLKOUT low		5		5	ns
$t_{dis(CLKL-S)}$	Disable time, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$ high impedance from CLKOUT low		5		5	ns
$t_{en(CLKL-A)}$	Enable time, Address, $\overline{\text{PS}}$, $\overline{\text{DS}}$, $\overline{\text{IS}}$ valid from CLKOUT low		2H+5		2H+5	ns
$t_{en(CLKL-RW)}$	Enable time, R/$\overline{\text{W}}$ enabled from CLKOUT low		2H+5		2H+5	ns
$t_{en(CLKL-S)}$	Enable time, $\overline{\text{MSTRB}}$, $\overline{\text{IOSTRB}}$ enabled from CLKOUT low		2H+5		2H+5	ns
$t_{v(HOLDA)}$	Valid time, $\overline{\text{HOLDA}}$ low after CLKOUT low	– 1	4	– 1	4	ns
	Valid time, $\overline{\text{HOLDA}}$ high after CLKOUT low	– 1	4	– 1	4	ns
$t_{w(HOLDA)}$	Pulse duration, $\overline{\text{HOLDA}}$ low duration	2H–3		2H–3		ns

timing requirements for $\overline{\text{HOLD}}$ [H = 0.5 $t_{c(CO)}$] (see Figure 26)

		’VC5410-100		’VC5410-120		UNIT
		MAX	MIN	MAX	MIN	
$t_{w(HOLD)}$	Pulse duration, $\overline{\text{HOLD}}$ low duration	4H+10		4H+10		ns
$t_{su(HOLD)}$	Setup time, $\overline{\text{HOLD}}$ before CLKOUT low	9		9		ns

$\overline{\text{HOLD}}$ and $\overline{\text{HOLDA}}$ timings (continued)

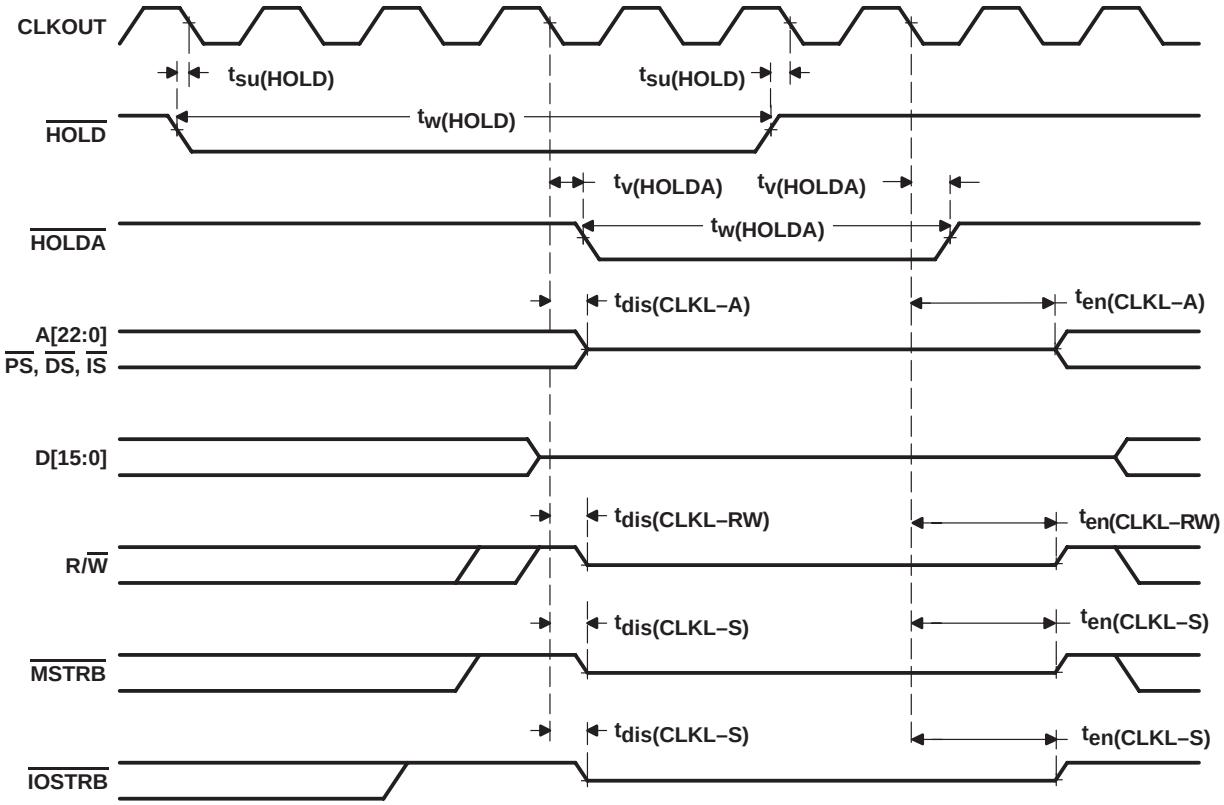


Figure 26. $\overline{\text{HOLD}}$ and $\overline{\text{HOLDA}}$ Timings (HM = 1)

reset, $\overline{\text{BIO}}$, interrupt, and MP/$\overline{\text{MC}}$ timings

timing requirements for reset, $\overline{\text{BIO}}$, interrupt, and MP/$\overline{\text{MC}}$ [H = 0.5 $t_{c(CO)}$] (see Figure 27, Figure 28, and Figure 29)

		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{h(RS)}$	Hold time, $\overline{\text{RS}}$ after CLKOUT low	0		0		ns
$t_{h(BIO)}$	Hold time, $\overline{\text{BIO}}$ after CLKOUT low	0		0		ns
$t_{h(INT)}$	Hold time, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$, after CLKOUT low†	0		0		ns
$t_{h(MPMC)}$	Hold time, MP/$\overline{\text{MC}}$ after CLKOUT low	0		0		ns
$t_{w(RSL)}$	Pulse duration, $\overline{\text{RS}}$ low‡§	4H+5		4H+5		ns
$t_{w(BIO)S}$	Pulse duration, $\overline{\text{BIO}}$ low, synchronous	2H+5		2H+5		ns
$t_{w(BIO)A}$	Pulse duration, $\overline{\text{BIO}}$ low, asynchronous	4H		4H		ns
$t_{w(INTH)S}$	Pulse duration, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$ high (synchronous)	2H+7		2H+7		ns
$t_{w(INTH)A}$	Pulse duration, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$ high (asynchronous)	4H		4H		ns
$t_{w(INTL)S}$	Pulse duration, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$ low (synchronous)	2H+7		2H+7		ns
$t_{w(INTL)A}$	Pulse duration, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$ low (asynchronous)	4H		4H		ns
$t_{w(INTL)WKP}$	Pulse duration, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$ low for IDLE2/IDLE3 wakeup	8		8		ns
$t_{su(RS)}$	Setup time, $\overline{\text{RS}}$ before X2/CLKIN low¶	5		5		ns
$t_{su(BIO)}$	Setup time, $\overline{\text{BIO}}$ before CLKOUT low	8	12	8	12	ns
$t_{su(INT)}$	Setup time, $\overline{\text{INTn}}$, $\overline{\text{NMI}}$, $\overline{\text{RS}}$ before CLKOUT low	9	13	8	12	ns
$t_{su(MPMC)}$	Setup time, MP/$\overline{\text{MC}}$ before CLKOUT low	8		8		ns

† The external interrupts ($\overline{\text{INT0}}$–$\overline{\text{INT3}}$, $\overline{\text{NMI}}$) are synchronized to the core CPU by way of a two-flip-flop synchronizer that samples these inputs with consecutive falling edges of CLKOUT. The input to the interrupt pins is required to represent a 1–0–0 sequence at the timing that is corresponding to three CLKOUTs sampling sequence.

‡ If the PLL mode is selected, then at power-on sequence, or at wakeup from IDLE3, $\overline{\text{RS}}$ must be held low for at least 50 µs to ensure synchronization and lock-in of the PLL.

§ Note that $\overline{\text{RS}}$ may cause a change in clock frequency, therefore changing the value of H.

¶ The diagram assumes clock mode is divide-by-2 and the CLKOUT divide factor is set to no-divide mode (DIVFCT=00 field in the BSCR).

reset, $\overline{\text{BIO}}$, interrupt, and MP/$\overline{\text{MC}}$ timings (continued)

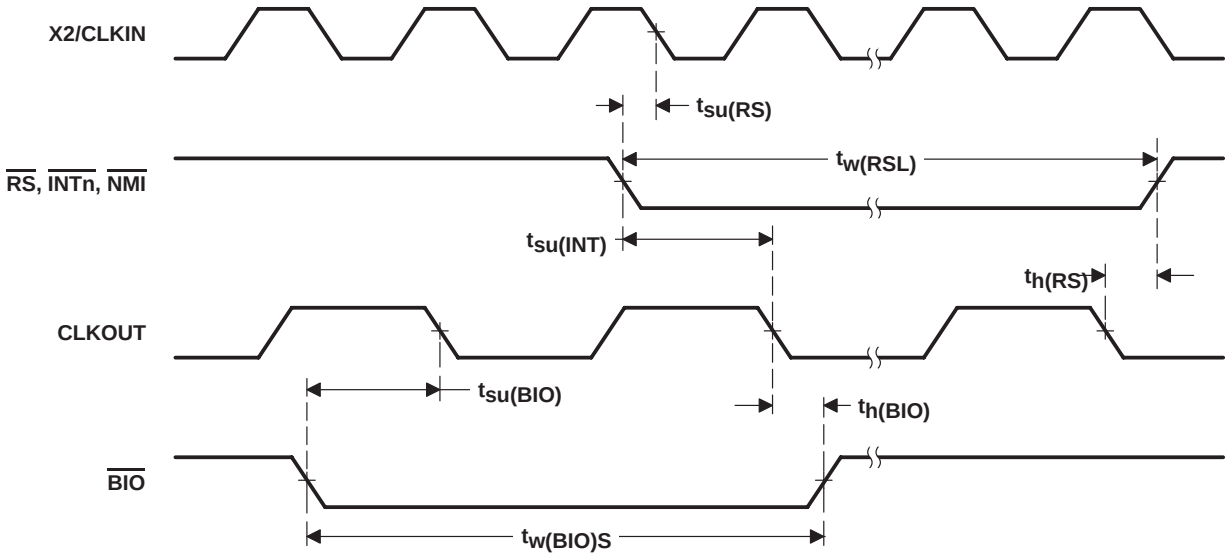


Figure 27. Reset and $\overline{\text{BIO}}$ Timings

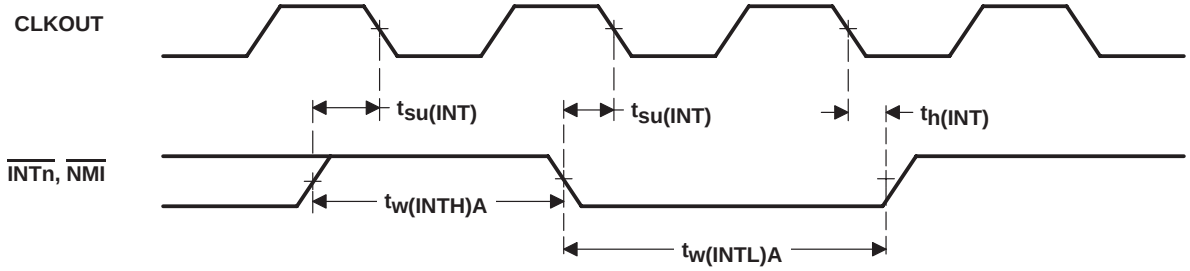


Figure 28. Interrupt Timing

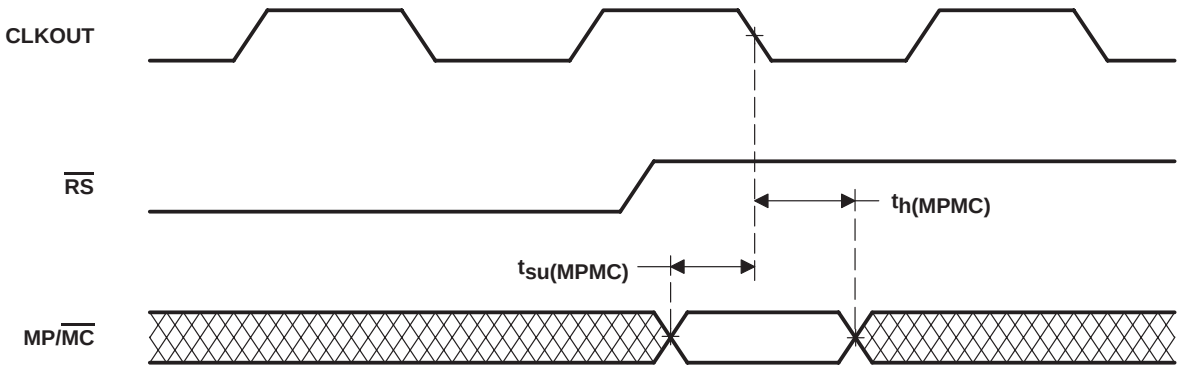


Figure 29. MP/$\overline{\text{MC}}$ Timing

instruction acquisition ($\overline{IAQ}$), interrupt acknowledge ($\overline{IACK}$), external flag (XF), and TOUT timings

switching characteristics over recommended operating conditions for $\overline{IAQ}$ and $\overline{IACK}$ [H = 0.5 $t_{c(CO)}$] (see Figure 30)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(CLKL-IAQL)}$	Delay time, CLKOUT low to $\overline{IAQ}$ low	– 1	4	– 1	7	ns
$t_{d(CLKL-IAQH)}$	Delay time, CLKOUT low to $\overline{IAQ}$ high	– 1	4	– 1	7	ns
$t_{d(A)IAQ}$	Delay time, $\overline{IAQ}$ low to address valid		3		4	ns
$t_{d(CLKL-IACKL)}$	Delay time, CLKOUT low to $\overline{IACK}$ low	– 1	4	– 1	6	ns
$t_{d(CLKL-IACKH)}$	Delay time, CLKOUT low to $\overline{IACK}$ high	– 1	4	– 1	6	ns
$t_{d(A)IACK}$	Delay time, $\overline{IACK}$ low to address valid		3		3	ns
$t_{h(A)IAQ}$	Hold time, address valid after $\overline{IAQ}$ high	– 3		– 6		ns
$t_{h(A)IACK}$	Hold time, address valid after $\overline{IACK}$ high	– 3		– 6		ns
$t_{w(IAQL)}$	Pulse duration, $\overline{IAQ}$ low	2H–3		2H–6		ns
$t_{w(IACKL)}$	Pulse duration, $\overline{IACK}$ low	2H–3		2H–6		ns

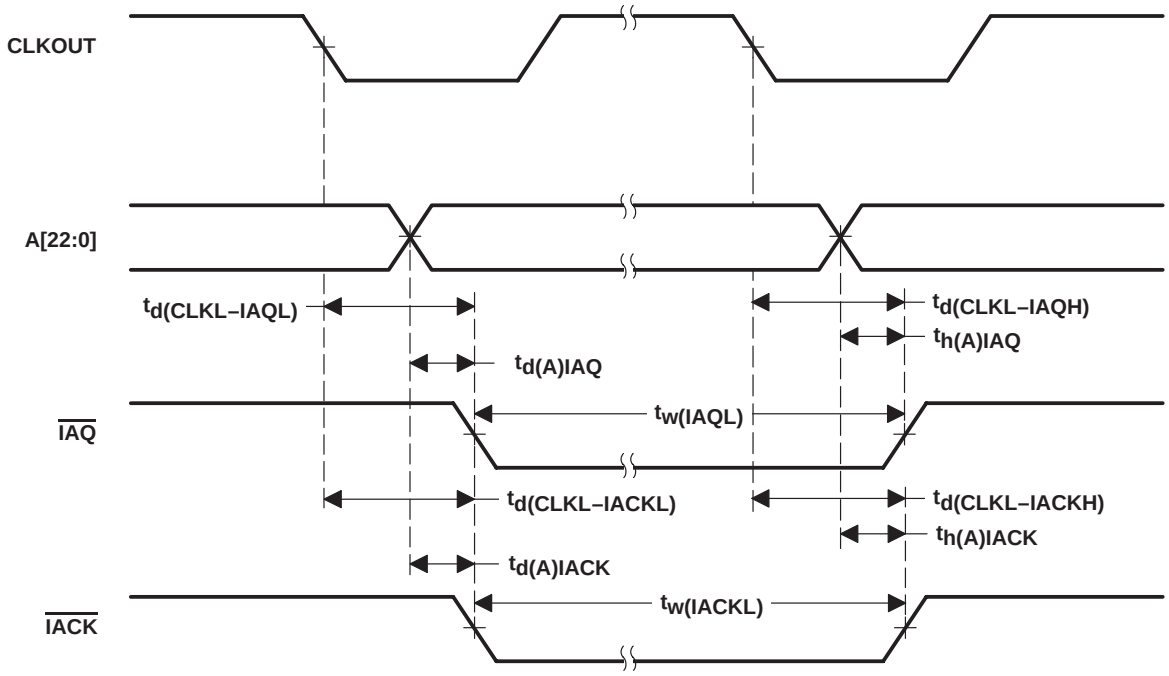


Figure 30. Instruction Acquisition ($\overline{IAQ}$) and Interrupt Acknowledge ($\overline{IACK}$) Timings

instruction acquisition ($\overline{IAQ}$), interrupt acknowledge ($\overline{IACK}$), external flag (XF), and TOUT timings (continued)

switching characteristics over recommended operating conditions for XF and TOUT [H = 0.5 $t_{c(CO)}$] (see Figure 31 and Figure 32)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(XF)}$	Delay time, CLKOUT low to XF high	–1	4	–1	6	ns
	Delay time, CLKOUT low to XF low	–1	4	–1	6	
$t_{d(TOUTH)}$	Delay time, CLKOUT low to TOUT high	–1	4	–1	6	ns
$t_{d(TOUTL)}$	Delay time, CLKOUT low to TOUT low	–1	4	–1	6	ns
$t_{w(TOUT)}$	Pulse duration, TOUT	2H–10		2H–6		ns

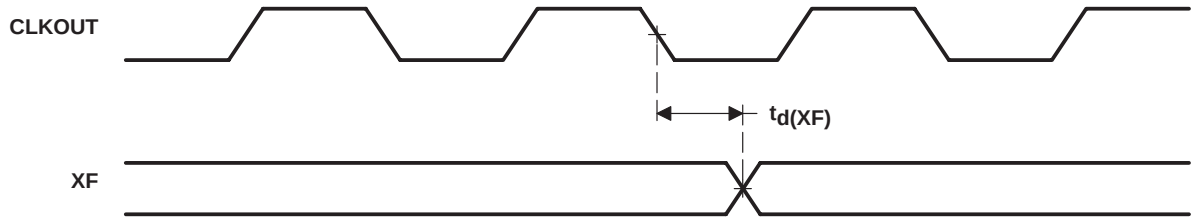


Figure 31. External Flag (XF) Timing

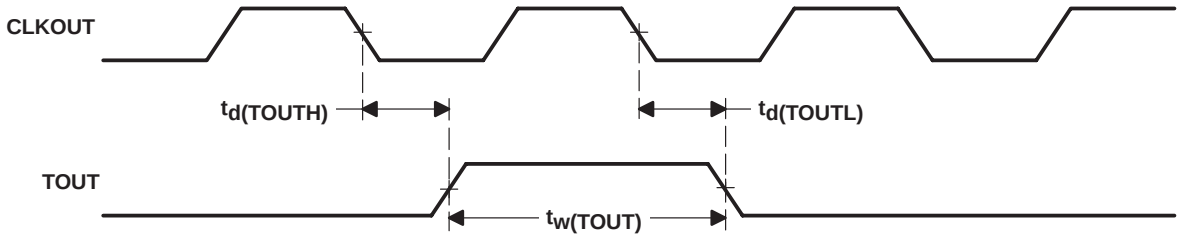


Figure 32. TOUT Timing

multichannel buffered serial port timing

timing requirements for McBSP [H=0.5$t_{c(CO)}$]† (see Figure 33 and Figure 34)

			’VC5410-100		’VC5410-120		UNIT
			MIN	MAX	MIN	MAX	
$t_{c(BCKRX)}$	Cycle time, BCLKR/X	BCLKR/X ext	4H		4H		ns
$t_{w(BCKRX)}$	Pulse duration, BCLKR/X high or BCLKR/X low	BCLKR/X ext	2H–1		2H–1		ns
$t_{su(BFRH-BCKRL)}$	Setup time, external BFSR high before BCLKR low	BCLKR int	13		13		ns
		BCLKR ext	4		4		
$t_{h(BCKRL-BFRH)}$	Hold time, external BFSR high after BCLKR low	BCLKR int	0		0		ns
		BCLKR ext	4		4		
$t_{su(BDRV-BCKRL)}$	Setup time, BDR valid before BCLKR low	BCLKR int	13		13		ns
		BCLKR ext	3		3		
$t_{h(BCKRL-BDRV)}$	Hold time, BDR valid after BCLKR low	BCLKR int	0		0		ns
		BCLKR ext	5		5		
$t_{su(BFXH-BCKXL)}$	Setup time, external BFSX high before BCLKX low	BCLKX int	13		13		ns
		BCLKX ext	5		5		
$t_{h(BCKXL-BFXH)}$	Hold time, external BFSX high after BCLKX low	BCLKX int	0		0		ns
		BCLKX ext	4		4		
$t_{r(BCKRX)}$	Rise time, BCLKR/X	BCLKR/X ext		8		8	ns
$t_{f(BCKRX)}$	Fall time, BCLKR/X	BCLKR/X ext		8		8	ns

† CLKRP = CLKXP = FSRP = FSXP = 0. If the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

switching characteristics for McBSP [H=0.5$t_{c(CO)}$]† (see Figure 33 and Figure 34)

PARAMETER			’VC5410-100		’VC5410-120		UNIT
			MIN	MAX	MIN	MAX	
$t_{c(BCKRX)}$	Cycle time, BCLKR/X	BCLKR/X int	4H		4H		ns
$t_{w(BCKRXH)}$	Pulse duration, BCLKR/X high	BCLKR/X int	D – 2‡	D‡	D – 2‡	D‡	ns
$t_{w(BCKRXL)}$	Pulse duration, BCLKR/X low	BCLKR/X int	C – 2‡	C‡	C – 2‡	C‡	ns
$t_{d(BCKRH-BFRV)}$	Delay time, BCLKR high to internal BFSR valid	BCLKR int	– 4	2	– 4	2	ns
		BCLKR ext	1	13	1	13	ns
$t_{d(BCKXH-BFXV)}$	Delay time, BCLKX high to internal BFSX valid	BCLKX int	– 4	2	– 4	2	ns
		BCLKX ext	1	13	1	13	
$t_{dis(BCKXH-BDXHZ)}$	Disable time, BCLKX high to BDX high impedance following last data bit of transfer	BCLKX int	– 3	5	– 3	5	ns
		BCLKX ext	1	19	1	19	
$t_{d(BCKXH-BDXV)}$	Delay time, BCLKX high to BDX valid	BCLKX int	0§	6	0§	6	ns
		BCLKX ext	3	15	3	15	
$t_{d(BFXH-BDXV)}$	Delay time, BFSX high to BDX valid ONLY applies when in data delay 0 (XDATDLY = 00b) mode	BFSX int	0§	8	0§	8	ns
		BFSX ext	0	10	0	10	

† CLKRP = CLKXP = FSRP = FSXP = 0. If the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.
‡ T = BCLKRX period = (1 + CLKGDV) * 2H
C = BCLKRX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * 2H when CLKGDV is even
D = BCLKRX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * 2H when CLKGDV is even
§ Minimum delay times also represent minimum output hold times.

multichannel buffered serial port timing (continued)

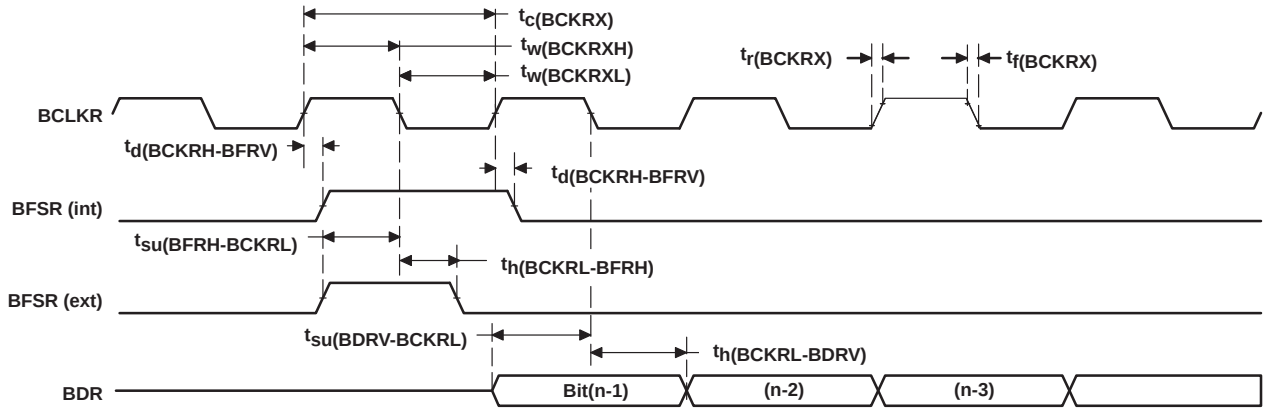


Figure 33. McBSP Receive Timings

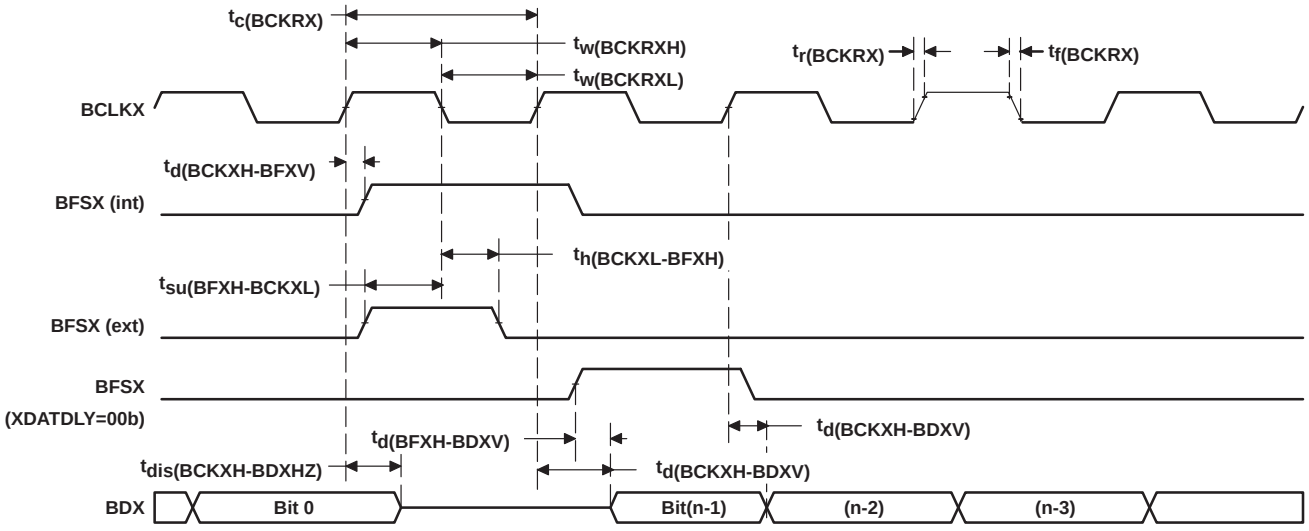


Figure 34. McBSP Transmit Timings

multichannel buffered serial port timing (continued)

timing requirements for McBSP general-purpose I/O (see Figure 35)

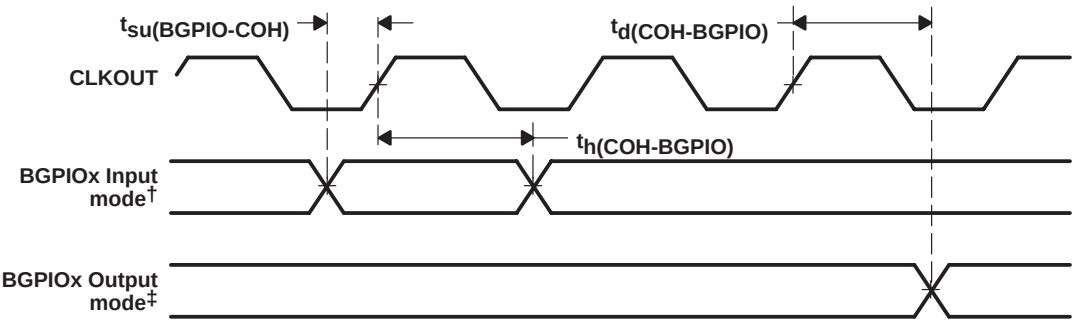
		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(BGPIO-COH)}$	Setup time, BGPIOx input mode before CLKOUT high†	9		8		ns
$t_{h(COH-BGPIO)}$	Hold time, BGPIOx input mode after CLKOUT high†	0		0		ns

† BGPIOx refers to BCLKRx, BFSRx, BDRx, BCLKXx, or BFSXx when configured as a general-purpose input.

switching characteristics for McBSP general-purpose I/O (see Figure 35)

PARAMETER		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{d(COH-BGPIO)}$	Delay time, CLKOUT high to BGPIOx output mode‡	0	5	0	5	ns

‡ BGPIOx refers to BCLKRx, BFSRx, BCLKXx, BFSXx, or BDXx when configured as a general-purpose output.



† BGPIOx refers to BCLKRx, BFSRx, BDRx, BCLKXx, or BFSXx when configured as a general-purpose input.
‡ BGPIOx refers to BCLKRx, BFSRx, BCLKXx, BFSXx, or BDXx when configured as a general-purpose output.

Figure 35. McBSP General-Purpose I/O Timings

multichannel buffered serial port timing (continued)

timing requirements for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 10b, CLKXP = 0† (see Figure 36)

		’5410-100				’5410-120				UNIT
		MASTER		SLAVE		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(BDRV-BCKXL)}$	Setup time, BDR valid before BCLKX low	12		7 – 6H		12		7 – 6H		ns
$t_{h(BCKXL-BDRV)}$	Hold time, BDR valid after BCLKX low	0		5 + 6H		0		5 + 6H		ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 10b, CLKXP = 0† (see Figure 36)

PARAMETER		’5410-100				’5410-120				UNIT
		MASTER‡		SLAVE		MASTER‡		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{h(BCKXL-BFXL)}$	Hold time, BFSX low after BCLKX low§	T – 7	T + 4			T – 7	T + 4			ns
$t_{d(BFXL-BCKXH)}$	Delay time, BFSX low to BCLKX high¶	C – 7	C + 5			C – 7	C + 5			ns
$t_{d(BCKXH-BDXV)}$	Delay time, BCLKX high to BDX valid	– 3	4	6H + 4	10H + 15	– 3	4	6H + 4	10H + 15	ns
$t_{dis(BCKXL-BDXHZ)}$	Disable time, BDX high impedance following last data bit from BCLKX low	C – 2	C + 3			C – 2	C + 3			ns
$t_{dis(BFXH-BDXHZ)}$	Disable time, BDX high impedance following last data bit from BFSX high			2H+ 3	6H + 17			2H+ 3	6H + 17	ns
$t_{d(BFXL-BDXV)}$	Delay time, BFSX low to BDX valid			4H + 2	8H + 17			4H + 2	8H + 17	ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
‡ T = BCLKX period = (1 + CLKGDV) * 2H
C = BCLKX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * 2H when CLKGDV is even
§ FSRP = FSXP = 1. As a SPI master, BFSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on BFSX and BFSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP
¶ BFSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (BCLKX).

multichannel buffered serial port timing (continued)

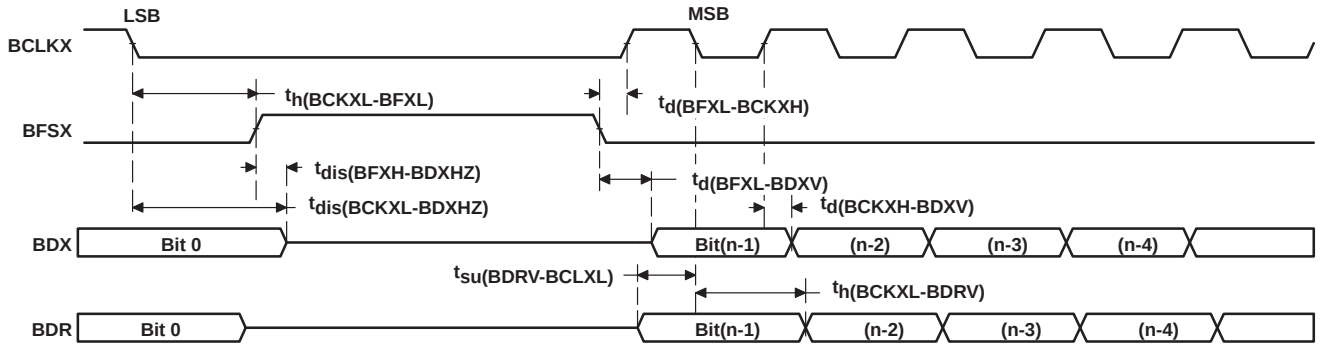


Figure 36. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

multichannel buffered serial port timing (continued)

timing requirements for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 11b, CLKXP = 0† (see Figure 37)

		'5410-100				'5410-120				UNIT
		MASTER		SLAVE		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(BDRV-BCKXL)}$	Setup time, BDR valid before BCLKX low	12		7 – 6H		12		7 – 6H		ns
$t_{h(BCKXH-BDRV)}$	Hold time, BDR valid after BCLKX high	0		5 + 6H		0		5 + 6H		ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 11b, CLKXP = 0† (see Figure 37)

PARAMETER		'5410-100				'5410-120				UNIT
		MASTER‡		SLAVE		MASTER‡		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{h(BCKXL-BFXL)}$	Hold time, BFSX low after BCLKX low§	C – 7	C + 4			C – 7	C + 4			ns
$t_{d(BFXL-BCKXH)}$	Delay time, BFSX low to BCLKX high¶	T –7	T + 5			T –7	T + 5			ns
$t_{d(BCKXL-BDXV)}$	Delay time, BCLKX low to BDX valid	– 3	4	6H + 4	10H + 15	– 3	4	6H + 4	10H + 15	ns
$t_{dis(BCKXL-BDXHZ)}$	Disable time, BDX high impedance following last data bit from BCLKX low	–2	4	6H + 3	10H + 17	–2	4	6H + 3	10H + 17	ns
$t_{d(BFXL-BDXV)}$	Delay time, BFSX low to BDX valid	D – 3	D + 5	4H + 2	8H + 17	D – 3	D + 5	4H + 2	8H + 17	ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

‡ T = BCLKX period = (1 + CLKGDV) * 2H
C = BCLKX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * 2H when CLKGDV is even
D = BCLKX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * 2H when CLKGDV is even

§ FSRP = FSXP = 1. As a SPI master, BFSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on BFSX and BFSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

¶ BFSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (BCLKX).

multichannel buffered serial port timing (continued)

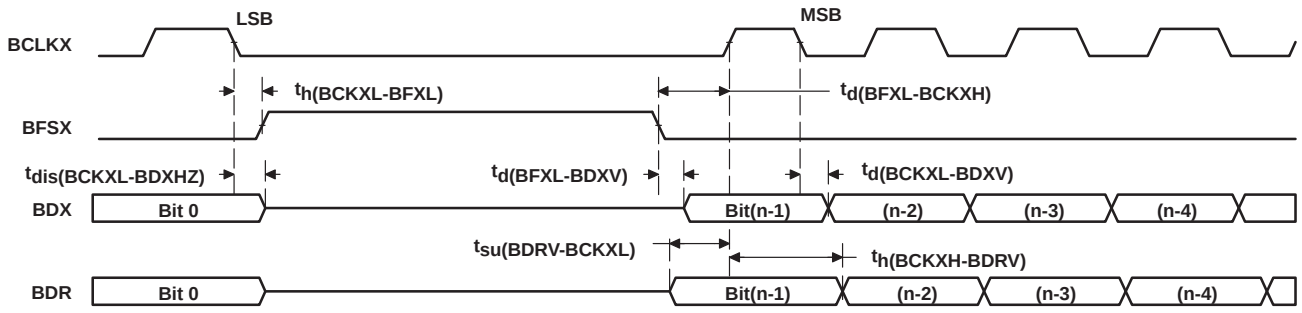


Figure 37. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0

multichannel buffered serial port timing (continued)

timing requirements for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 10b, CLKXP = 1† (see Figure 38)

		'5410-100				'5410-120				UNIT
		MASTER		SLAVE		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(BDRV-BCKXH)}$	Setup time, BDR valid before BCLKX high	12		7 – 6H		12		7 – 6H		ns
$t_{h(BCKXH-BDRV)}$	Hold time, BDR valid after BCLKX high	0		5 + 6H		0		5 + 6H		ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 10b, CLKXP = 1†‡ (see Figure 38)

PARAMETER		'5410-100				'5410-120				UNIT
		MASTER		SLAVE		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{h(BCKXH-BFXL)}$	Hold time, BFSX low after BCLKX high§	T – 7	T + 4			T – 7	T + 4			ns
$t_{d(BFXL-BCKXL)}$	Delay time, BFSX low to BCLKX low¶	D – 7	D + 5			D – 7	D + 5			ns
$t_{d(BCKXL-BDXV)}$	Delay time, BCLKX low to BDX valid	– 3	4	6H + 4	10H + 15	– 3	4	6H + 4	10H + 15	ns
$t_{dis(BCKXH-BDXHZ)}$	Disable time, BDX high impedance following last data bit from BCLKX high	D – 2	D + 3			D – 2	D + 3			ns
$t_{dis(BFXH-BDXHZ)}$	Disable time, BDX high impedance following last data bit from BFSX high			2H + 3	6H + 17			2H + 3	6H + 17	ns
$t_{d(BFXL-BDXV)}$	Delay time, BFSX low to BDX valid			4H + 2	8H + 17			4H + 2	8H + 17	ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
‡ T = BCLKX period = (1 + CLKGDV) * 2H
D = BCLKX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * 2H when CLKGDV is even
§ FSRP = FSXP = 1. As a SPI master, BFSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on BFSX and BFSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP
¶ BFSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (BCLKX).

multichannel buffered serial port timing (continued)

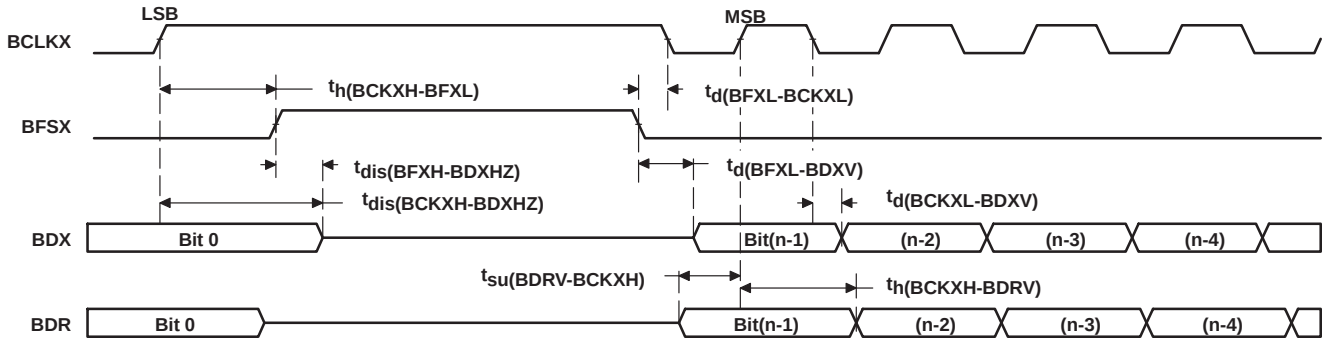


Figure 38. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1

multichannel buffered serial port timing (continued)

timing requirements for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 11b, CLKXP = 1† (see Figure 39)

		’5410-100				’5410-120				UNIT
		MASTER		SLAVE		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(BDRV-BCKXL)}$	Setup time, BDR valid before BCLKX low	12		7 – 6H		12		7 – 6H		ns
$t_{h(BCKXL-BDRV)}$	Hold time, BDR valid after BCLKX low	0		5 + 6H		0		5 + 6H		ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics for McBSP as SPI master or slave: [H=0.5$t_{c(CO)}$] CLKSTP = 11b, CLKXP = 1†‡ (see Figure 39)

PARAMETER		’5410-100				’5410-120				UNIT
		MASTER‡		SLAVE		MASTER‡		SLAVE		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{h(BCKXH-BFXL)}$	Hold time, BFSX low after BCLKX high§	D – 7	D + 4			D – 7	D + 4			ns
$t_{d(BFXL-BCKXL)}$	Delay time, BFSX low to BCLKX low¶	T – 7	T + 5			T – 7	T + 5			ns
$t_{d(BCKXH-BDXV)}$	Delay time, BCLKX high to BDX valid	– 3	4	6H + 4	10H + 15	– 3	4	6H + 4	10H + 15	ns
$t_{dis(BCKXH-BDXHZ)}$	Disable time, BDX high impedance following last data bit from BCLKX high	–2	4	6H + 3	10H + 17	–2	4	6H + 3	10H + 17	ns
$t_{d(BFXL-BDXV)}$	Delay time, BFSX low to BDX valid	C – 3	C + 5	4H + 2	8H + 17	C – 3	C + 5	4H + 2	8H + 17	ns

† For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.
‡ T = BCLKX period = (1 + CLKGDV) * 2H
C = BCLKX low pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2) * 2H when CLKGDV is even
D = BCLKX high pulse width = T/2 when CLKGDV is odd or zero and = (CLKGDV/2 + 1) * 2H when CLKGDV is even
§ FSRP = FSXP = 1. As a SPI master, BFSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on BFSX and BFSR is inverted before being used internally.
CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP
CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP
¶ BFSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (BCLKX).

multichannel buffered serial port timing (continued)

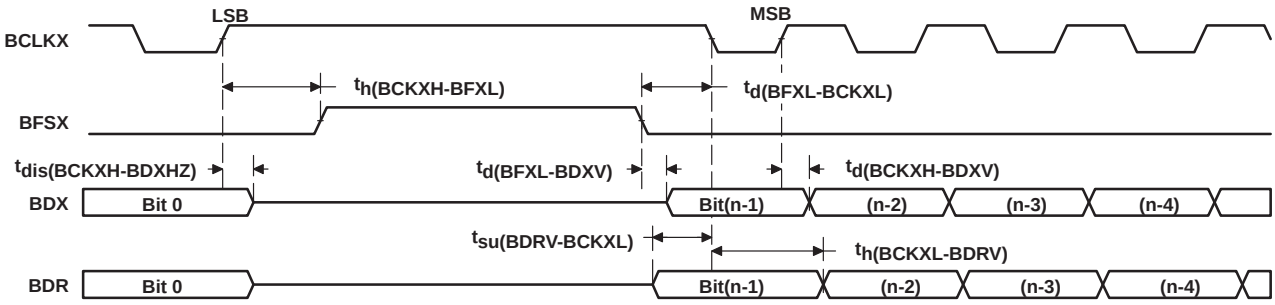


Figure 39. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1

HPI8 timing

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$][†‡§] (see Figure 40, Figure 41, and Figure 42)

PARAMETER			’5410-100		’5410-120		UNIT
			MIN	MAX	MIN	MAX	
$t_{en(DSL-HD)}$	Enable time, HD driven from DS low		2	15	2	15	ns
$t_{d(DSL-HDV1)}$	Delay time, DS low to HDx valid for first byte of an HPI read	Case 1a: Memory accesses when DMAC is active in 16-bit mode and $t_{w(DSH)} < 18H$		$18H+15 – t_{w(DSH)}$		$18H+15 – t_{w(DSH)}$	ns
		Case 1b: Memory accesses when DMAC is active in 16-bit mode and $t_{w(DSH)} \geq 18H$		15		15	
		Case 1c: Memory access when DMAC is active in 32-bit mode and $t_{w(DSH)} < 26H$		$26H+15 – t_{w(DSH)}$		$26H+15 – t_{w(DSH)}$	
		Case 1d: Memory access when DMAC is active in 32-bit mode and $t_{w(DSH)} \geq 26H$		15		15	
		Case 2a: Memory accesses when DMAC is inactive and $t_{w(DSH)} < 10H$		$10H+15 – t_{w(DSH)}$		$10H+15 – t_{w(DSH)}$	
		Case 2b: Memory accesses when DMAC is inactive and $t_{w(DSH)} \geq 10H$		15		15	
		Case 3: Register accesses		15		15	
$t_{d(DSL-HDV2)}$	Delay time, DS low to HDx valid for second byte of an HPI read			15		15	ns
$t_{h(DSH-HDV)R}$	Hold time, HDx valid after DS high, for a HPI read		1	5	1	5	ns
$t_{v(HYH-HDV)}$	Valid time, HDx valid after HRDY high			5		5	
$t_{d(DSH-HYL)}$	Delay time, DS high to HRDY low (see Note 1)			8		8	ns
$t_{d(DSH-HYH)}$	Delay time, DS high to HRDY high	Case 1a: Memory accesses when DMAC is active in 16-bit mode		18H+12		18H+12	ns
		Case 1b: Memory accesses when DMAC is active in 32-bit mode		26H+12		26H+12	ns
		Case 2: Memory accesses when DMAC is inactive		10H+12		10H+12	ns
		Case 3: Write accesses to HPIC register (see Note 2)		6H+12		6H+12	
$t_{d(HCS-HRDY)}$	Delay time, $\overline{HCS}$ low/high to HRDY low/high			5		5	ns
$t_{d(COH-HYH)}$	Delay time, CLKOUT high to HRDY high			10		10	ns
$t_{d(COH-HTX)}$	Delay time, CLKOUT high to $\overline{HINT}$ change			10		10	ns

† DS refers to the logical OR of $\overline{HCS}$, $\overline{HDS1}$, and $\overline{HDS2}$.
‡ HD refers to any of the HPI data bus pins (HD0, HD1, HD2, etc.). HAD stands for HCNTL0, HCNTL1, and HR/$\overline{W}$.
§ DMAC stands for direct memory access controller (DMAC). The HPI8 shares the internal DMA bus with the DMAC, thus HPI8 access times are affected by DMAC activity.

NOTES: 1. The HRDY output is always high when the $\overline{HCS}$ input is high, regardless of DS timings.
2. This timing applies when writing a one to the DSPINT bit or $\overline{HINT}$ bit of the HPIC register. All other writes to the HPIC occur asynchronously, and do not cause HRDY to be deasserted.

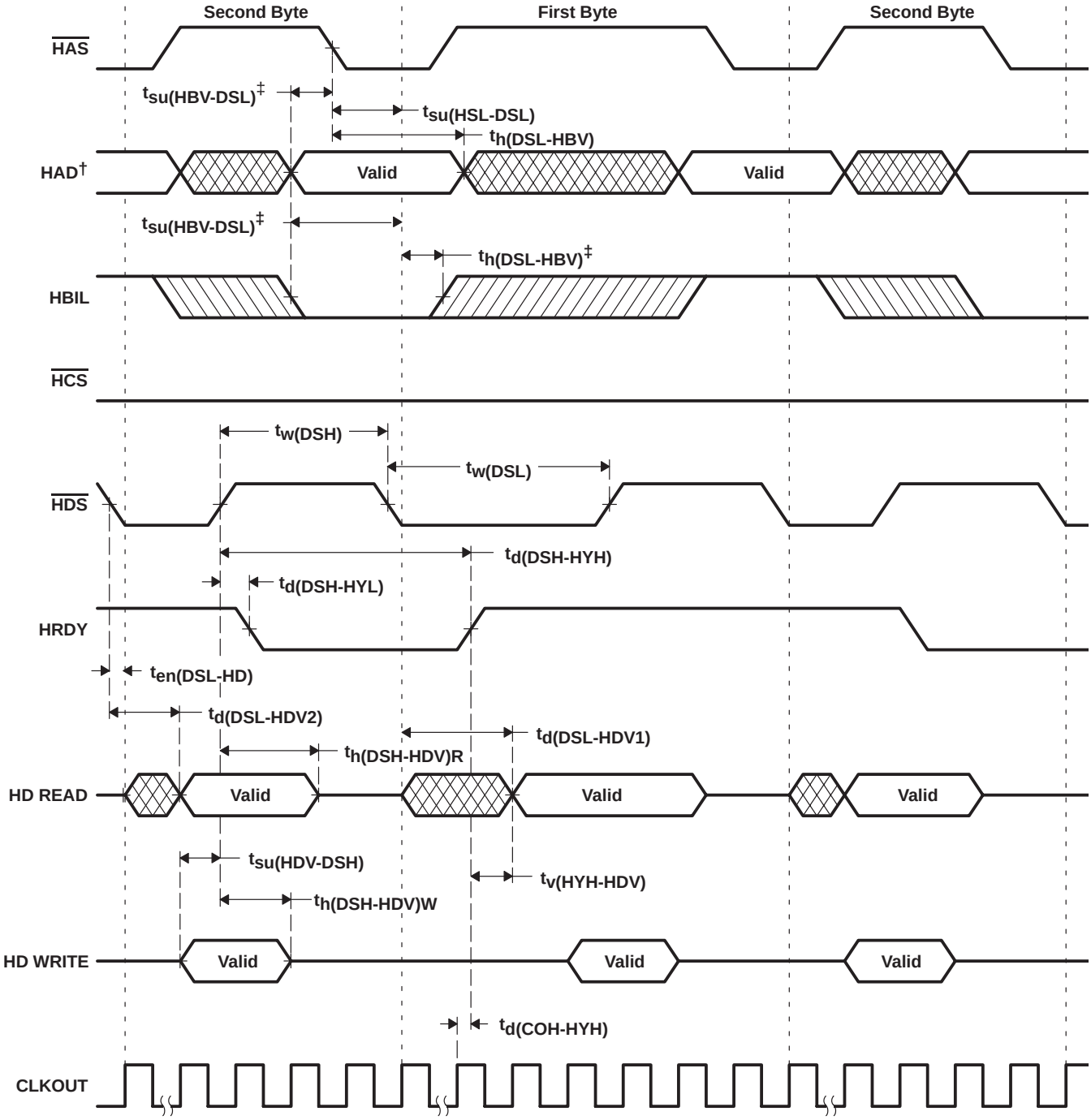
HPI8 timing (continued)

timing requirements†‡ (see Figure 40, Figure 41, and Figure 42)

		’VC5410-100		’VC5410-120		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(HBV-DSL)}$	Setup time, HBIL valid before DS low§	5		5		ns
$t_{h(DSL-HBV)}$	Hold time, HBIL valid after DS low	5		5		ns
$t_{su(HSL-DSL)}$	Setup time, $\overline{HAS}$ low before DS low	10		10		ns
$t_{w(DSL)}$	Pulse duration, DS low	20		20		ns
$t_{w(DSH)}$	Pulse duration, DS high	10		10		ns
$t_{su(HDV-DSH)}$	Setup time, HD valid before DS high, HPI write	5		5		ns
$t_{h(DSH-HDV)W}$	Hold time, HD valid after DS high, HPI write	3		3		ns

† DS refers to the logical OR of $\overline{HCS}$, $\overline{HDS1}$, and $\overline{HDS2}$.
‡ HD refers to any of the HPI data bus pins (HD0, HD1, HD2, etc.).
§ When $\overline{HAS}$ is not used ($\overline{HAS}$ always high), this timing refers to DS

HPI8 timing (continued)



† HAD refers to HCNTL0, HCNTL1, and HR/$\overline{W}$.
‡ When $\overline{HAS}$ is not used ($\overline{HAS}$ always high), this timing refers to DS

Figure 40. Using $\overline{HDS}$ to Control Accesses ($\overline{HCS}$ Always Low)

HPI8 timing (continued)

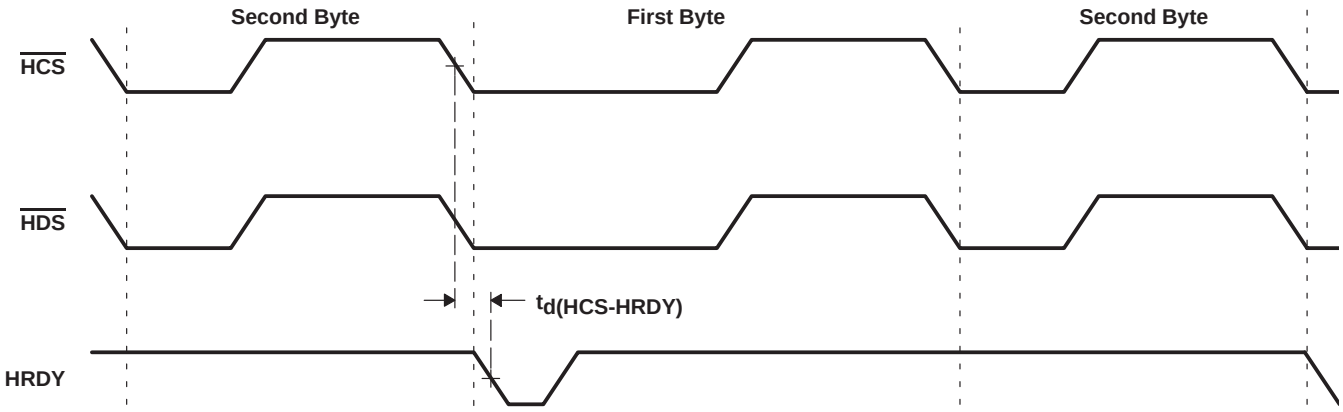


Figure 41. Using HCS to Control Accesses

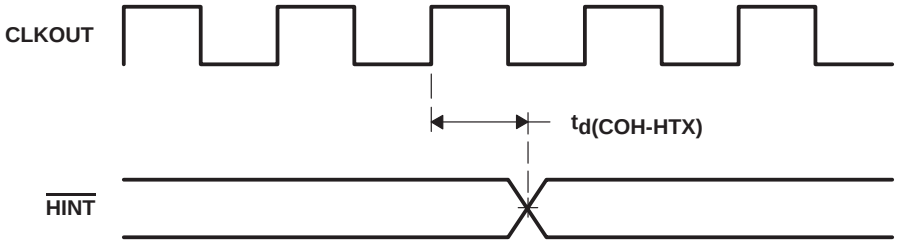
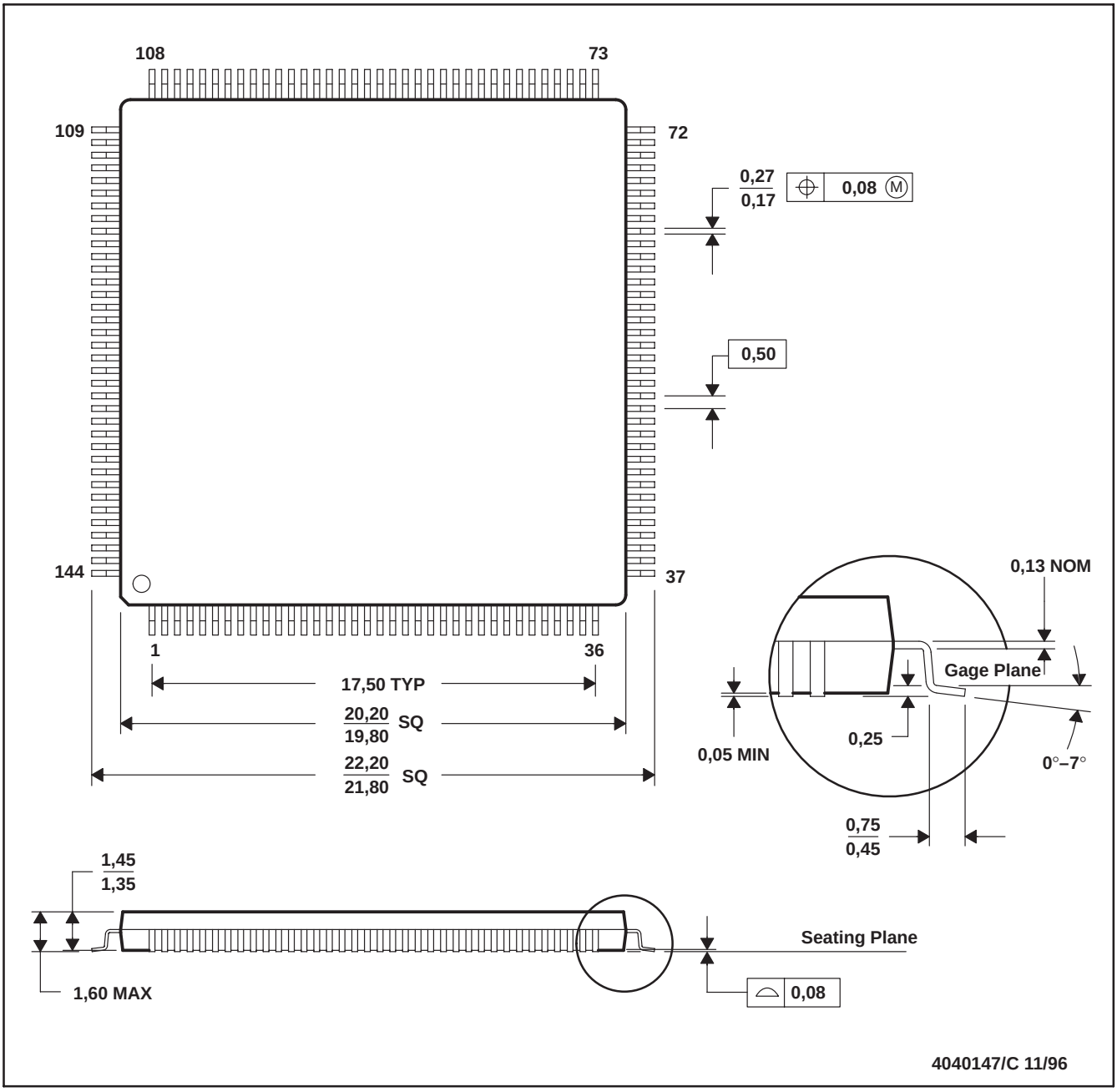


Figure 42. HINT Timing

MECHANICAL DATA

PGE (S-PQFP-G144) **PLASTIC QUAD FLATPACK**



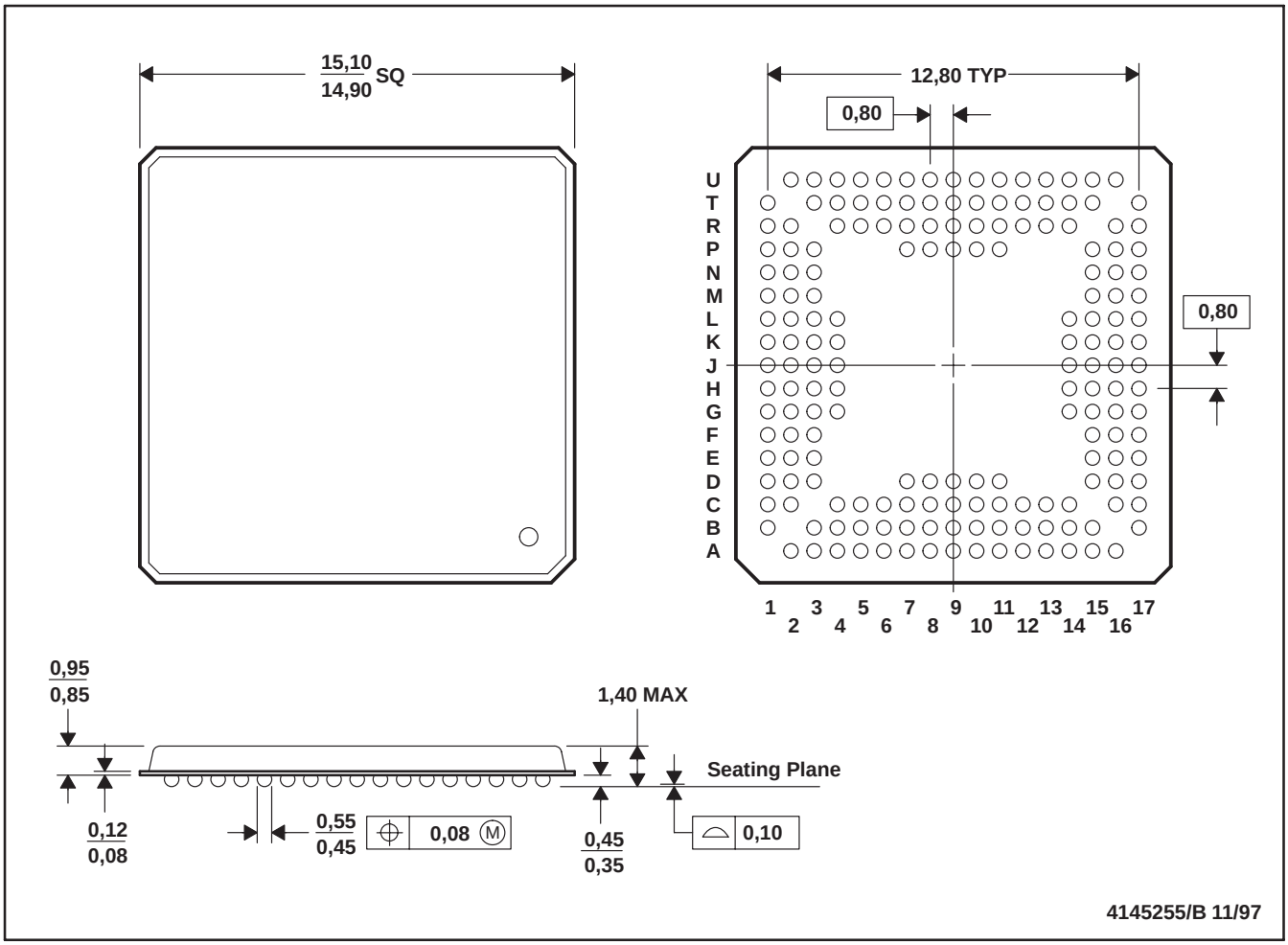
NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

Thermal Resistance Characteristics

PARAMETER	°C/W
$R_{\Theta JA}$	56
$R_{\Theta JC}$	5

MECHANICAL DATA

GGW (S-PBGA-N176) **PLASTIC BALL GRID ARRAY PACKAGE**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. MicroStar BGA™ configuration

MicroStar BGA is a trademark of Texas Instruments Incorporated.

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, patent infringement, and limitation of liability.

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

Customers are responsible for their applications using TI components.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.