THE DATASHEET OF

TMS320F206PZ

www.win-source.net 0086-755-83957316

- **High-Performance Static CMOS Technology**
- **Includes the T320C2xLP Core CPU**
- **TMS320F206 is a Member of the TMS320C20x Generation, Which Also Includes the TMS320C203, and TMS320C209 Devices**
- **Instruction-Cycle Time 50 ns @ 5 V**
- **Source Code Compatible With TMS320C25**
- **Upwardly Code-Compatible With TMS320C5x Devices**
- **Three External Interrupts**
- **TMS320F206 Integrated Memory:**
  - **544 × 16 Words of On-Chip Dual-Access Data RAM**
  - **32K × 16 Words of On-Chip Flash Memory (EEPROM)**
  - **4K × 16 Words of On-Chip Single-Access Program/Data RAM**
- **224K × 16-Bit Maximum Addressable External Memory Space**
  - **64K Program**
  - **64K Data**
  - **64K Input/Output (I/O)**
  - **32K Global**
- **32-Bit ALU/Accumulator**
- **16 × 16-Bit Multiplier With a 32-Bit Product**
- **Block Moves from Data and Program Space**
- **TMS320F206 Peripherals:**
  - **On-Chip 16-Bit Timer**
  - **On-Chip Software-Programmable Wait-State (0 to 7) Generator**
  - **On-Chip Oscillator**
  - **On-Chip Phase-Locked Loop (PLL)**
  - **Six General-Purpose I/O Pins**
  - **Full-Duplex Asynchronous Serial Port (UART)**
  - **Enhanced Synchronous Serial Port (ESSP) With Four-Level-Deep FIFOs**
- **Input Clock Options**
  - **Options – Multiply-by-One, -Two, or -Four and Divide-by-Two**
- **Support of Hardware Wait States**
- **Power Down IDLE Mode**
- **IEEE 1149.1†-Compatible Scan-Based Emulation**
- **100-Pin Thin Quad Flat Package (TQFP) (PZ Suffix)**

description

The TMS320F206 Texas Instruments (TI™) digital signal processor (DSP) is fabricated with static CMOS integrated-circuit technology, and the architectural design is based upon that of the TMS320C20x series, optimized for low-power operation. The combination of advanced Harvard architecture, on-chip peripherals, on-chip memory, and a highly specialized instruction set is the basis of the operational flexibility and speed of the ’F206.

The ’F206 offers these advantages:

- 32K × 16 words on-chip flash EEPROM reduces system cost and facilitates prototyping
- Enhanced TMS320 architectural design for increased performance and versatility
- Advanced integrated-circuit processing technology for increased performance
- ’F206 devices are pin- and code-compatible with ’C203 devices.
- Source code for the ’F206 DSP is software-compatible with the ’C1x and ’C2x DSPs and is upwardly compatible with fifth-generation DSPs (’C5x)
- New static-design techniques for minimizing power consumption and increasing radiation tolerance



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.

TI is a trademark of Texas Instruments Incorporated.
† IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1998, Texas Instruments Incorporated

PZ PACKAGE
(TOP VIEW)

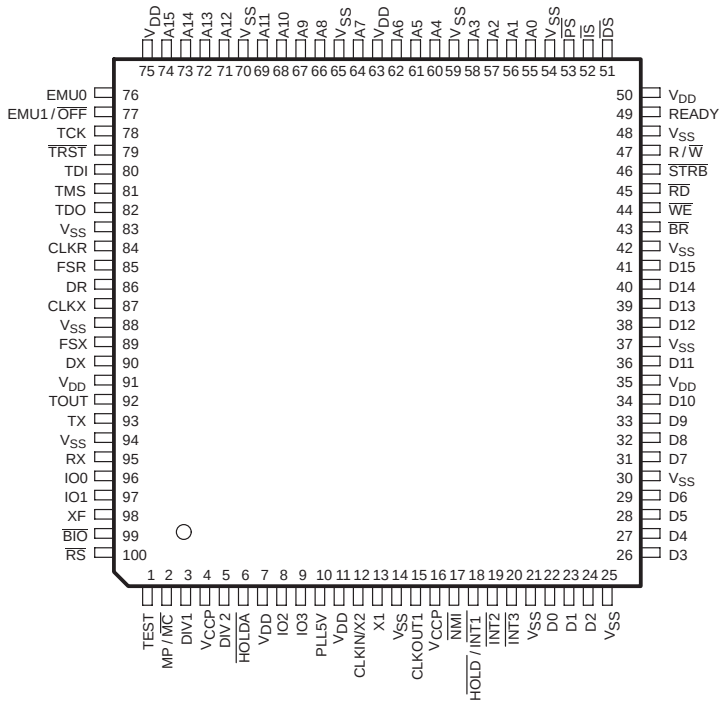


Table 1 shows the capacity of on-chip RAM and ROM, the number of serial and parallel I/O ports, the execution time of one machine cycle, and the type of package with total pin count of the TMS320F206 device.

Table 1. Characteristics of the TMS320F206 Processor

DEVICE	ON-CHIP MEMORY				I/O PORTS		POWER SUPPLY (V)	CYCLE TIME (ns)	PACKAGE TYPE WITH PIN COUNT
	RAM		ROM	FLASH EEPROM					
	DATA	DATA/ PROG	PROG	PROG	SERIAL	PARALLEL			
TMS320F206	288	4K + 256	–	32K	2	64K	5	50	100-pin TQFP

TMS320F206 Terminal Functions

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
DATA AND ADDRESS BUSES			
D15	41	I/O/Z	Parallel data bus D15 [most significant bit (MSB)] through D0 [least significant bit (LSB)]. D15–D0 are used to transfer data between the TMS320F206 and external data/program memory or I/O devices. Placed in the high-impedance state when not outputting (R/$\overline{W}$ high) or $\overline{RS}$ when asserted. They go into the high-impedance state when $\overline{OFF}$ is active low.
D14	40		
D13	39		
D12	38		
D11	36		
D10	34		
D9	33		
D8	32		
D7	31		
D6	29		
D5	28		
D4	27		
D3	26		
D2	24		
D1	23		
D0	22		
A15	74	O/Z	Parallel address bus A15 (MSB) through A0 (LSB). A15–A0 are used to address external data/program memory or I/O devices. These signals go into the high-impedance state when $\overline{OFF}$ is active low.
A14	73		
A13	72		
A12	71		
A11	69		
A10	68		
A9	67		
A8	66		
A7	64		
A6	62		
A5	61		
A4	60		
A3	58		
A2	57		
A1	56		
A0	55		
MEMORY CONTROL SIGNALS			
$\overline{PS}$	53	O/Z	Program-select signal. $\overline{PS}$ is always high unless low-level asserted for communicating to off-chip program space. $\overline{PS}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
$\overline{DS}$	51	O/Z	Data-select signal. $\overline{DS}$ is always high unless low-level asserted for communicating to off-chip program space. $\overline{DS}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
$\overline{IS}$	52	O/Z	I/O space-select signal. $\overline{IS}$ is always high unless low-level asserted for communicating to I/O ports. $\overline{IS}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
READY	49	I	Data-ready input. READY indicates that an external device is prepared for the bus transaction to be completed. If the external device is not ready (READY low), the TMS320F206 waits one cycle and checks READY again. If READY is not used, it should be pulled high.
R/$\overline{W}$	47	O/Z	Read/write signal. R/$\overline{W}$ indicates transfer direction when communicating with an external device. R/$\overline{W}$ is normally in read mode (high), unless low level is asserted for performing a write operation. R/$\overline{W}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
$\overline{RD}$	45	O/Z	Read-select indicates an active, external read cycle. $\overline{RD}$ is active on all external program, data, and I/O reads. $\overline{RD}$ goes into the high-impedance state when $\overline{OFF}$ is active low. The function of the $\overline{RD}$ pin can be programmed to provide an inverted R/$\overline{W}$ signal instead of $\overline{RD}$. The FRDN bit (bit 15) in the PMST register controls this selection.

† I = input, O = output, Z = high impedance, PWR = power, GND = ground

TMS320F206 Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
MEMORY CONTROL SIGNALS (CONTINUED)			
$\overline{WE}$	44	O/Z	Write enable. The falling edge of $\overline{WE}$ indicates that the device is driving the external data bus (D15–D0). Data can be latched by an external device on the rising edge of $\overline{WE}$. $\overline{WE}$ is active on all external program, data, and I/O writes. $\overline{WE}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
$\overline{STRB}$	46	O/Z	Strobe signal. $\overline{STRB}$ is always high unless asserted low to indicate an external bus cycle. $\overline{STRB}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
MULTI-PROCESSING SIGNALS			
$\overline{BR}$	43	O/Z	Bus-request signal. $\overline{BR}$ is asserted when a global data-memory access is initiated. $\overline{BR}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
$\overline{HOLDA}$	6	O/Z	Hold-acknowledge signal. $\overline{HOLDA}$ indicates to the external circuitry that the processor is in a hold state and that the address, data, and memory control lines are in the high-impedance state so that they are available to the external circuitry for access of local memory. $\overline{HOLDA}$ goes into the high-impedance state when $\overline{OFF}$ is active low.
XF	98	O/Z	External flag output (latched software-programmable signal). XF is used for signalling other processors in multiprocessing configurations or as a general-purpose output pin. XF goes into the high-impedance state when $\overline{OFF}$ is active low.
$\overline{BIO}$	99	I	Branch control input. When polled by the BIOZ instruction, if $\overline{BIO}$ is low, the TMS320F206 executes a branch.
IO0 IO1 IO2 IO3	96 97 8 9	I/O/Z	Software-controlled input/output pins by way of the asynchronous serial-port control register (ASPCR). At reset, IO0–IO3 are configured as inputs. These pins can be used as general-purpose input/output pins or as handshake control for the UART. IO0–IO3 go into the high-impedance state when $\overline{OFF}$ is active low. IO0 also functions as a frame-sync output when the synchronous serial port (SSP) is used in multichannel mode.
INITIALIZATION, INTERRUPTS, AND RESET OPERATIONS			
$\overline{RS}$	100	I	Reset input. $\overline{RS}$ causes the TMS320F206 to terminate execution and forces the program counter to zero. When $\overline{RS}$ is brought high, execution begins at location 0 of program memory after 16 cycles. $\overline{RS}$ affects various registers and status bits.
TEST	1	I	Reserved input pin. TEST is connected to V_{SS} for normal operation.
MP/$\overline{MC}$	2	I	Microprocessor/microcomputer-mode-select pin. If MP/$\overline{MC}$ is low, the on-chip flash memory is mapped into program space. When MP/$\overline{MC}$ is high, the device accesses off-chip memory. This pin is only sampled at reset, and its value is latched into bit 0 of the PMST register.
$\overline{NMI}$	17	I	Nonmaskable interrupt. $\overline{NMI}$ is an external interrupt that cannot be masked by way of the interrupt-mode bit (INTM) or the interrupt mask register (IMR). When $\overline{NMI}$ is activated, the processor traps to the appropriate vector location. If $\overline{NMI}$ is not used, it should be pulled high.
$\overline{HOLD}$/$\overline{INT1}$	18	I	$\overline{HOLD}$ and $\overline{INT1}$ share the same pin. Both are treated as interrupt signals. If the MODE bit is 0 in the interrupt-control register (ICR), hold logic can be implemented in combination with the IDLE instruction in software. At reset, the MODE bit in ICR is zero, enabling the HOLD mode for the pin.
$\overline{INT2}$ $\overline{INT3}$	19 20	I	External user interrupts. $\overline{INT2}$ and $\overline{INT3}$ are prioritized and maskable by the IMR and the INTM. $\overline{INT2}$ and $\overline{INT3}$ can be polled and reset by way of the interrupt flag register (IFR).
OSCILLATOR, PLL, AND TIMER SIGNALS			
TOUT	92	O/Z	Timer output. TOUT signals a pulse when the on-chip timer counts down past zero. The pulse is one CLKOUT1-cycle wide. TOUT goes into the high-impedance state when $\overline{OFF}$ is active low.
CLKOUT1	15	O/Z	Master clock output signal. The CLKOUT1 signal cycles at the machine-cycle rate of the CPU. The internal machine cycle is bounded by the rising edges of CLKOUT1. CLKOUT1 goes into the high-impedance state when $\overline{OFF}$ is active low.
CLKIN/X2 X1	12 13	I O	Input clock. CLKIN/X2 is the input clock to the device. As CLKIN, the pin operates as the external oscillator clock input, and as X2, the pin operates as the internal oscillator input with X1 being the internal oscillator output.

† I = input, O = output, Z = high impedance, PWR = power, GND = ground

TMS320F206 Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
OSCILLATOR, PLL, AND TIMER SIGNALS (CONTINUED)			
DIV1 DIV2	3 5	I	DIV1 and DIV2 provide clock-mode inputs. DIV1–DIV2 should not be changed unless the $\overline{RS}$ signal is active.
PLL5V	10	I	The TMS320F206 is strictly a 5-V device. For this reason, the PLL5V pin should always be pulled high.
SERIAL PORT AND UART SIGNALS			
CLKX	87	I/O/Z	Transmit clock. CLKX is a clock signal for clocking data from the serial-port transmit shift register (XSR) to the DX data-transmit pin. The CLKX can be an input if the MCM bit in the synchronous serial-port control register (SSPCR) is set to 0. CLKX can also be driven by the device at one-half of the CLKOUT1 frequency when MCM = 1. If the serial port is not being used, CLKX goes into the high-impedance state when $\overline{OFF}$ is active low. Value at reset is as an input.
CLKR	84	I/O/Z	Receive-clock input. External clock signal for clocking data from the DR (data-receive) pin into the serial-port receive shift register (RSR). CLKR must be present during serial-port transfers. If the serial port is not being used, CLKR can be sampled as an input by the IN0 bit of the SSPCR. This pin also functions as a frame-sync output when the SSP is used in multichannel mode.
FSR	85	I/O/Z	Frame synchronization pulse for receive input. The falling edge of the FSR pulse initiates the data-receive process, beginning the clocking of the RSR. FSR goes into the high-impedance state when $\overline{OFF}$ is active low. This pin also functions as a frame-sync output when the SSP is used in multichannel mode.
FSX	89	I/O/Z	Frame synchronization pulse for transmit input/ouput. The falling edge of the FSX pulse initiates the data-transmit process, beginning the clocking of the serial-port transmit shift register (XSR). Following reset, FSX is an input. FSX can be selected by software to be an output when the TXM bit in the SSPCR is set to 1. FSX goes into the high-impedance state when $\overline{OFF}$ is active low.
DR	86	I	Serial-data receive input. Serial data is received in the receive shift register (RSR) through the DR pin.
DX	90	O/Z	Serial-port transmit output. Serial data is transmitted from the transmit shift register (XSR) through the DX pin. DX is in the high-impedance state when $\overline{OFF}$ is active low.
TX	93	O/Z	Asynchronous transmit data pin. TX is in the high-impedance state when $\overline{OFF}$ is active low.
RX	95	I	Asynchronous receive data pin
TEST SIGNALS			
$\overline{TRST}$	79	I	IEEE Standard 1149.1 (JTAG) test reset. $\overline{TRST}$, when driven high, gives the scan system control of the operations of the device. If $\overline{TRST}$ is driven low, the device operates in its functional mode, and the test signals are ignored. **If the $\overline{TRST}$ pin is not driven, an external pulldown resistor must be used.**
TCK	78	I	JTAG test clock. TCK is normally a free-running clock signal with a 50% duty cycle. The changes on the test-access port (TAP) 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.
TMS	81	I	JTAG test-mode select. TMS is clocked into the TAP controller on the rising edge of TCK.
TDI	80	I	JTAG test-data input. TDI is clocked into the selected register (instruction or data) on a rising edge of TCK.
TDO	82	O/Z	JTAG 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.
EMU0	76	I/O/Z	Emulator pin 0. When $\overline{TRST}$ is driven low, EMU0 must be high for activation of the $\overline{OFF}$ condition. When $\overline{TRST}$ is driven high, EMU0 is used as an interrupt to or from the emulator system and is defined as an input/output through the JTAG scan.
EMU1/$\overline{OFF}$	77	I/O/Z	Emulator pin 1. Emulator pin 1 disables all outputs. When $\overline{TRST}$ is driven high, EMU1/$\overline{OFF}$ is used as an interrupt to or from the emulator system and is defined as an input/output through the JTAG scan. When $\overline{TRST}$ is driven low, this pin is configured as $\overline{OFF}$. EMU1/$\overline{OFF}$, when active low, puts all output drivers in the high-impedance state. Note that $\overline{OFF}$ is used exclusively for testing and emulation purposes (not for multiprocessing applications). Therefore, for the $\overline{OFF}$ condition, the following apply: $\overline{TRST}$ = 0 EMU0 = 1 EMU1/$\overline{OFF}$ = 0

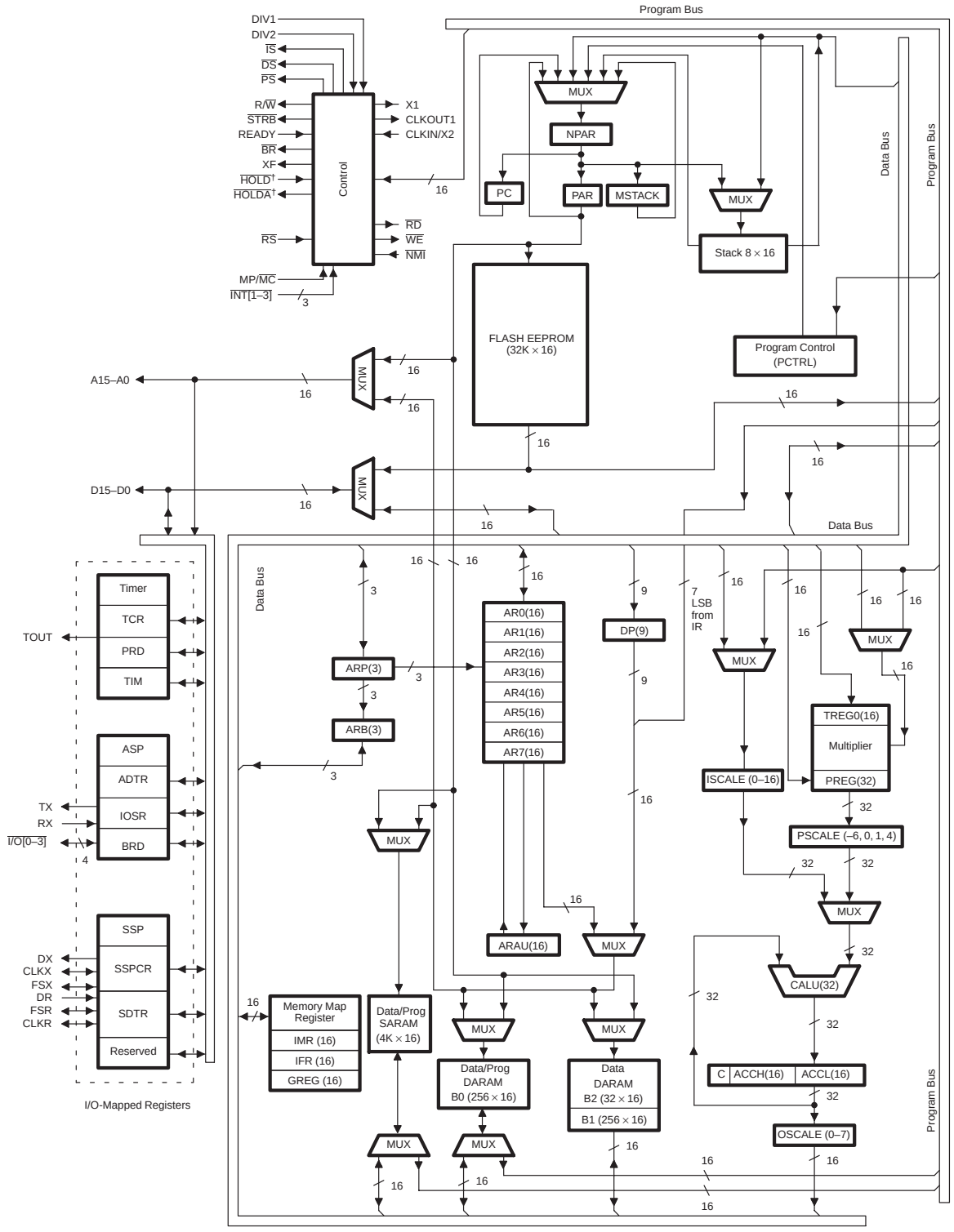
† I = input, O = output, Z = high impedance, PWR = power, GND = ground

TMS320F206 Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
SUPPLY PINS			
V_{CCP}	4 16	PWR	V_{CCP} must be connected directly to V_{DD}.
V_{DD}	7 11 35 50 63 75 91	PWR	Power
V_{SS}	14 21 25 30 37 42 48 54 59 65 70 83 88 94	GND	Ground

† I = input, O = output, Z = high impedance, PWR = power, GND = ground

functional block diagram of the ’F206 internal hardware



NOTES: A. Symbol descriptions appear in Table 3.
B. For clarity the data and program buses are shown as single buses although they include address and data bits.

Table 2. Legend for the ’F206 Internal Hardware Functional Block Diagram

SYMBOL	NAME	DESCRIPTION
ACC	Accumulator	32-bit register that stores the results and provides input for subsequent CALU operations. Also includes shift and rotate capabilities
ARAU	Auxiliary Register Arithmetic Unit	An unsigned, 16-bit arithmetic unit used to calculate indirect addresses using the auxiliary registers as inputs and outputs
AUX REGS	Auxiliary Registers 0–7	These 16-bit registers are used as pointers to anywhere within the data space address range. They are operated upon by the ARAU and are selected by the auxiliary register pointer (ARP). AR0 can also be used as an index value for AR updates of more than one and as a compare value to AR.
$\overline{BR}$	Bus Request Signal	$\overline{BR}$ is asserted during access of the external global data memory space. READY is asserted to the device when the global data memory is available for the bus transaction. $\overline{BR}$ can be used to extend the data memory address space by up to 32K words.
C	Carry	Register carry output from CALU. C is fed back into the CALU for extended arithmetic operation. The C bit resides in status register 1 (ST1), and can be tested in conditional instructions. C is also used in accumulator shifts and rotates.
CALU	Central Arithmetic Logic Unit	32-bit-wide main arithmetic logic unit for the TMS320C2xx core. The CALU executes 32-bit operations in a single machine cycle. CALU operates on data coming from ISCALE or PSCALE with data from ACC, and provides status results to PCTRL.
CNF	On-Chip RAM Configuration Control Bit	If set to 0, the reconfigurable data dual-access RAM (DARAM) block B0 is mapped to data space; otherwise, B0 is mapped to program space.
GREG	Global Memory Allocation Register	GREG specifies the size of the global data memory space.
IMR	Interrupt Mask Register	IMR individually masks or enables the seven interrupts.
IFR	Interrupt Flag Register	The 7-bit IFR indicates that the TMS320F206 has latched an interrupt from one of the seven maskable interrupts.
INTM	Interrupt-Mode Bit	When INTM is set to 0, all unmasked interrupts are enabled. When INTM is set to 1, all maskable interrupts are disabled.
INT#	Interrupt Traps	A total of 32 interrupts by way of hardware and/or software are available.
ISCALE	Input Data-Scaling Shifter	16 to 32-bit barrel left-shifter. ISCALE shifts incoming 16-bit data 0 to16 positions left, relative to the 32-bit output within the fetch cycle; therefore, no cycle overhead is required for input scaling operations.
MPY	Multiplier	16 × 16-bit multiplier to a 32-bit product. MPY executes multiplication in a single cycle. MPY operates either signed or unsigned 2s-complement arithmetic multiply.
MSTACK	Micro Stack	MSTACK provides temporary storage for the address of the next instruction to be fetched when program address-generation logic is used to generate sequential addresses in data space.
MUX	Multiplexer	Multiplexes buses to a common input
NPAR	Next Program Address Register	NPAR holds the program address to be driven out on the PAB on the next cycle.
OSCALE	Output Data-Scaling Shifter	16 to 32-bit barrel left-shifter. OSCALE shifts the 32-bit accumulator output 0 to 7 bits left for quantization management and outputs either the 16-bit high- or low-half of the shifted 32-bit data to the Data-Write Data Bus (DWEB).
PAR	Program Address Register	PAR holds the address currently being driven on PAB for as many cycles as it takes to complete all memory operations scheduled for the current bus cycle.
PC	Program Counter	PC increments the value from NPAR to provide sequential addresses for instruction-fetching and sequential data-transfer operations.
PCTRL	Program Controller	PCTRL decodes instruction, manages the pipeline, stores status, and decodes conditional operations.

Table 2. Legend for the ’F206 Internal Hardware Functional Block Diagram (Continued)

SYMBOL	NAME	DESCRIPTION
PM	Product Shift-Mode Register Bits	These two bits identify which of the four product-shift modes (–6, 0, 1, 4) are used by PSCALE. PM resides in ST1. See Table 6.
PREG	Product Register	32-bit register holds results of 16 × 16 multiply.
PSCALE	Product-Scaling Shifter	0-, 1- or 4-bit left shift, or 6-bit right shift of multiplier product. The left-shift options are used to manage the additional sign bits resulting from the 2s-complement multiply. The right-shift option is used to scale down the number to manage overflow of product accumulation in the CALU. PSCALE resides in the path from the 32-bit product shifter and from either the CALU or the Data-Write Data Bus (DWEB), and requires no cycle overhead.
TREG	Temporary Register	16-bit register holds one of the operands for the multiply operations. TREG holds the dynamic shift count for the LACT, ADDT, and SUBT instructions. TREG holds the dynamic bit position for the BITT instruction.
SSPCR	Synchronous Serial-Port Control Register	SSPCR is the control register for selecting the serial port’s mode of operation.
SDTR	Synchronous Serial-Port Transmit and Receive Register	SDTR is the data-transmit and data-receive register.
TCR	Timer-Control Register	TCR contains the control bits that define the divide-down ratio, start/stop the timer, and reload the period. Also contained in TCR is the current count in the prescaler. Reset initializes the timer-divide-down ratio to 0 and starts the timer.
PRD	Timer-Period Register	PRD contains the 16-bit period that is loaded into the timer counter when the counter borrows or when the reload bit is activated. Reset initializes the PRD to 0xFFFF.
TIM	Timer-Counter Register	TIM contains the current 16-bit count of the timer. Reset initializes the TIM to 0xFFFF.
UART	Universal Asynchronous Receive/Transmit	UART is the asynchronous serial port.
ASPCR	Asynchronous Serial-Port Control Register	ASPCR controls the asynchronous serial-port operation.
ADTR	Asynchronous Data Register	Asynchronous data-transmit and data-receive register
IOSR	I/O Status Register	IOSR detects current levels (and changes with inputs) on pins IO0–IO3 and the status of UART.
BRD	Baud-Rate Divisor	Used to set the baud rate of the UART
ST0 ST1	Status Register	ST0 and ST1 contain the status of various conditions and modes. These registers can be stored in and loaded from data memory, thereby allowing the status of the machine to be saved and restored.
IMR	Interrupt Mask Registers	IMR individually masks or enables the seven interrupts.
IFR	Interrupt Flag Register	IFR indicates that the CPU has latched an interrupt pulse from one of the maskable interrupts.
STACK	Stack	STACK is a block of memory used for storing return addresses for subroutines and interrupt-service routines, or for storing data. The ’C20x stack is 16-bit wide and eight-level deep.

architectural overview

The ’F206 advanced Harvard-type architecture maximizes the processing power by maintaining two separate memory bus structures — program and data — for full-speed execution. The multiple buses allow data and instructions to be read simultaneously. Instructions support data transfers between the two spaces. This architecture permits coefficients stored in program memory to be read in RAM, eliminating the need for a separate coefficient ROM. This, coupled with a four-deep pipeline, allows the TMS320F206 to execute most instructions in a single cycle.

status and control registers

Two status registers, ST0 and ST1, contain the status of various conditions and modes. These registers can be stored into data memory and loaded from data memory, thereby allowing the status of the machine to be saved and restored for subroutines.

The load-status-register (LST) instruction is used to write to ST0 and ST1. The store-status-register (SST) instruction is used to read from ST0 and ST1 (except the INTM bit, which is not affected by the LST instruction). The individual bits of these registers can be set or cleared when using the SETC and CLRC instructions. Table 3 and Table 4 show the organization of status registers ST0 and ST1, indicating all status and control bits contained in each. Several bits in the status registers are reserved and read as logic 1s. Refer to Table 5 for status-register field definitions.

Table 3. Status and Control Register Zero

	15 13	12	11	10	9	8 7 6 5 4 3 2 1 0
ST0	ARP	OV	OVM	1	INTM	DP

Table 4. Status and Control Register One

	15 13	12	11	10	9	8	7	6	5	4	3	2	1 0
ST1	ARB	CNF	TC	SXM	C	1	1	1	1	XF	1	1	PM

status and control registers (continued)

Table 5. Status Register Field Definitions

FIELD	FUNCTION
ARB	Auxiliary register pointer buffer. Whenever the ARP is loaded, the old ARP value is copied to the ARB except during an LST instruction. When the ARB is loaded by an LST #1 instruction, the same value is also copied to the ARP.
ARP	Auxiliary register pointer. ARP selects the AR to be used in indirect addressing. When the ARP is loaded, the old ARP value is copied to the ARB register. ARP can be modified by memory-reference instructions when using indirect addressing, and by the LARP, MAR, and LST instructions. The ARP is also loaded with the same value as ARB when an LST #1 instruction is executed.
C	Carry Bit. C is set to 1 if the result of an addition generates a carry, or reset to 0 if the result of a subtraction generates a borrow. Otherwise, C is reset after an addition or set after a subtraction, except if the instruction is ADD or SUB with a 16-bit shift. In these cases, the ADD can only set and the SUB only reset the carry bit, but cannot affect it otherwise. The single-bit shift and rotate instructions also affect C, as well as the SETC, CLRC, and LST #1 instructions. Branch instructions have been provided to branch on the status of C. C is set to 1 on a reset.
CNF	On-chip RAM configuration-control bit. If CNF is set to 0, the reconfigurable data DARAM blocks are mapped to data space; otherwise, they are mapped to program space. The CNF can be modified by the SETC CNF, CLRC CNF, and LST #1 instructions. $\overline{RS}$ sets the CNF to 0.
DP	Data memory page pointer. The 9-bit DP register is concatenated with the seven LSBs of an instruction word to form a direct memory address of 16 bits. DP can be modified by the LST and LDP instructions.
INTM	Interrupt-mode bit. When INTM is set to 0, all unmasked interrupts are enabled. When set to 1, all maskable interrupts are disabled. INTM is set and reset by the SETC INTM and CLRC INTM instructions. $\overline{RS}$ also sets INTM. INTM has no effect on the unmaskable $\overline{RS}$ and $\overline{NMI}$ interrupts. Note that INTM is unaffected by the LST instruction. This bit is set to 1 by reset. It is also set to 1 when a maskable interrupt trap is taken.
OV	Overflow-flag bit. As a latched overflow signal, OV is set to 1 when overflow occurs in the ALU. Once an overflow occurs, the OV remains set until a reset, BCND/D on OV/NOV, or LST instructions clear OV.
OVM	Overflow-mode bit. When OVM is set to 0, overflowed results overflow normally in the accumulator. When set to 1, the accumulator is set to either its most positive or negative value upon encountering an overflow. The SETC and CLRC instructions set and reset this bit, respectively. LST can also be used to modify the OVM.
PM	Product-shift mode. If these two bits are 00, the multiplier's 32-bit product is loaded into the ALU with no shift. If PM = 01, the PREG output is left-shifted one place and loaded into the ALU, with the LSB zero-filled. If PM = 10, PREG output is left-shifted by four bits and loaded into the ALU, with the LSBs zero-filled. PM = 11 produces a right shift of six bits, sign-extended. Note that the PREG contents remain unchanged. The shift takes place when transferring the contents of the PREG to the ALU. PM is loaded by the SPM and LST #1 instructions. PM is cleared by $\overline{RS}$.
SXM	Sign-extension mode bit. SXM = 1 produces sign extension on data as it is passed into the accumulator through the scaling shifter. SXM = 0 suppresses sign extension. SXM does not affect the definitions of certain instructions; for example, the ADDS instruction suppresses sign extension regardless of SXM. SXM is set by the SETC SXM and reset by the CLRC SXM instructions, and can be loaded by the LST #1. SXM is set to 1 by reset.
TC	Test/control flag bit. TC is affected by the BIT, BITT, CMPR, LST #1, and NORM instructions. TC is set to a 1 if a bit tested by BIT or BITT is a 1, if a compare condition tested by CMPR exists between AR (ARP) and AR0, if the exclusive-OR function of the two MSBs of the accumulator is true when tested by a NORM instruction. The conditional branch, call, and return instructions can execute, based on the condition of TC.
XF	XF pin status bit. XF indicates the state of the XF pin, a general-purpose output pin. XF is set by the SETC XF and reset by the CLRC XF instructions. XF is set to 1 by reset.

central processing unit

The TMS320F206 central processing unit (CPU) contains a 16-bit scaling shifter, a 16x16-bit parallel multiplier, a 32-bit central arithmetic logic unit (CALU), a 32-bit accumulator, and additional shifters at the outputs of both the accumulator and the multiplier. This section describes the CPU components and their functions. The functional block diagram shows the components of the CPU.

input scaling shifter

The TMS320F206 provides a scaling shifter with a 16-bit input connected to the data bus and a 32-bit output connected to the CALU. This shifter operates as part of the path of data coming from program or data space to the CALU and requires no cycle overhead. It is used to align the 16-bit data coming from memory to the 32-bit CALU. This is necessary for scaling arithmetic as well as aligning masks for logical operations.

The scaling shifter produces a left shift of 0 to 16 on the input data. The LSBs of the output are filled with zeros; the MSBs can either be filled with zeros or sign-extended, depending upon the value of the SXM bit (sign-extension mode) of status register ST1. The shift count is specified by a constant embedded in the instruction word or by a value in TREG. The shift count in the instruction allows for specific scaling or alignment operations specific to that point in the code. The TREG base shift allows the scaling factor to adapt to the performance of the system.

multiplier

The TMS320F206 uses a 16x16-bit hardware multiplier that is capable of computing a signed or an unsigned 32-bit product in a single machine cycle. All multiply instructions, except the MPYU (multiply unsigned) instruction, perform a signed-multiply operation. That is, two numbers being multiplied are treated as 2s-complement numbers, and the result is a 32-bit 2s-complement number. There are two registers associated with the multiplier:

- 16-bit temporary register (TREG) that holds one of the operands for the multiplier, and
- 32-bit product register (PREG) that holds the product.

Four product-shift modes (PM) are available at the PREG output (PSCALE). These shift modes are useful for performing multiply/accumulate operations, performing fractional arithmetic, or justifying fractional products. The PM field of status register ST1 specifies the PM shift mode, as shown in Table 6.

Table 6. PSCALE Product Shift Modes

PM	SHIFT	DESCRIPTION
00	no shift	Product fed to CALU or data bus with no shift
01	left 1	Removes the extra sign bit generated in a 2s-complement multiply to produce a Q31 product
10	left 4	Removes the extra four sign bits generated in a 16x13 2s-complement multiply to a produce a Q31 product when using the multiply by a 13-bit constant
11	right 6	Scales the product to allow up to 128 product accumulation without the possibility of accumulator overflow

The product can be shifted one bit to compensate for the extra sign bit gained in multiplying two 16-bit 2s-complement numbers (MPY). A four-bit shift is used in conjunction with the MPY instruction with a short immediate value (13 bits or less) to eliminate the four extra sign bits gained in multiplying a 16-bit number by a 13-bit number. Finally, the output of PREG can be right-shifted 6 bits to enable the execution of up to 128 consecutive multiply/accumulates without the possibility of overflow.

The LT (load TREG) instruction normally loads TREG to provide one operand (from the data bus), and the MPY (multiply) instruction provides the second operand (also from the data bus). A multiplication can also be performed with a 13-bit immediate operand when using the MPY instruction. A product is then obtained every two cycles. For efficient implementation of multiple products, or multiple sums of products, the CPU provides pipelining of the TREG load operation with certain CALU operations which use the PREG. These operations include: load ACC with PREG (LTP); add PREG to ACC (LTA); add PREG to ACC and shift TREG input data to next address in data memory (LTD); and subtract PREG from ACC (LTS).

multiplier (continued)

Two multiply/accumulate instructions (MAC and MACD) fully utilize the computational bandwidth of the multiplier, allowing both operands to be processed simultaneously. The data for these operations can be transferred to the multiplier each cycle by way of the program and data buses. This facilitates single-cycle multiply/accumulates when used with the repeat (RPT) instruction. In these instructions, the coefficient addresses are generated by program address generation (PAGEN), while the data addresses are generated by data address generation (DAGEN). This allows the repeated instruction to access the values sequentially from the coefficient table and step through the data in any of the indirect addressing modes.

The MACD instruction, when repeated, supports filter constructs (weighted running averages) so that as the sum-of-products is executed, the sample data is shifted in memory to make room for the next sample and to discard the oldest sample.

The MPYU instruction performs an unsigned multiplication, which greatly facilitates extended-precision arithmetic operations. The unsigned contents of TREG are multiplied by the unsigned contents of the addressed data memory location, with the result placed in PREG. This allows the operands of greater than 16 bits to be broken down into 16-bit words and processed separately to generate products of greater than 32 bits. The SQRA (square/add) and SQRS (square/subtract) instructions pass the same value to both inputs of the multiplier for squaring a data memory value.

After the multiplication of two 16-bit numbers, the 32-bit product is loaded into the 32-bit product register (PREG). The product from PREG can be transferred to the CALU or to data memory through the SPH (store product high) and SPL (store product low) instructions. Note: the transfer of PREG to either the CALU or data memory passes through the PSCALE shifter and is therefore, affected by the product-shift mode value defined by the PM bits in the ST1 register. This is important when saving PREG in an interrupt-service routine context save as the PSCALE shift effects cannot be modeled in the restore operation. PREG can be cleared by executing the MPY #0 instruction. The product register can be restored by loading the saved low half into TREG and executing a MPY #1 instruction. The high half is then loaded using the LPH instruction.

central arithmetic logic unit

The TMS320F206 CALU implements a wide range of arithmetic and logical functions, the majority of which execute in a single clock cycle. This ALU is referred to as “central” to differentiate it from a second ALU used for indirect address generation (called the ARAU). Once an operation is performed in the CALU, the result is transferred to the accumulator (ACC) where additional operations, such as shifting, can occur. Data that is input to the CALU can be scaled by ISCALE when coming from one of the data buses (DRDB or PRDB) or scaled by PSCALE when coming from the multiplier.

The CALU is a general-purpose arithmetic/logic unit that operates on 16-bit words taken from data memory or derived from immediate instructions. In addition to arithmetic operations, the CALU can perform Boolean operations, facilitating the bit manipulation ability required for a high-speed controller. One input to the CALU is always provided from the accumulator, and the other input can be provided from the product register (PREG) of the multiplier or the output of the scaling shifter (that has been read from data memory or from the ACC). After the CALU has performed the arithmetic or logical operation, the result is stored in the accumulator.

The TMS320F206 supports floating-point operations for applications requiring a large dynamic range. The NORM (normalization) instruction is used to normalize fixed-point numbers contained in the accumulator by performing left shifts. The four bits of the TREG define a variable shift through the scaling shifter for the LACT/ADDT/SUBT (load/add to/subtract from accumulator with shift specified by TREG) instructions. These instructions are useful in floating-point arithmetic where denormalization of a number is required; that is, floating-point to fixed-point conversion. They are also useful in the implementation of automatic-gain control (AGC) at the input of a filter. The BITT (bit test) instruction provides testing of a single bit of a word in data memory based on the value contained in the four LSBs of TREG.

central arithmetic logic unit (continued)

The CALU overflow saturation mode can be enabled/disabled by setting/resetting the OVM bit of ST0. Setting the OVM status register bit selects the overflow saturation mode. When the CALU is in the overflow saturation mode and an overflow occurs, the overflow flag is set and the accumulator is loaded with either the most positive or the most negative value representable in the accumulator, depending upon the direction of the overflow. The value of the accumulator upon saturation is 07FFFFFFh (positive) or 080000000h (negative). If the OVM (overflow mode) status register bit is reset and an overflow occurs, the overflowed results are loaded into the accumulator without modification. (Note that logical operations cannot result in overflow.)

The CALU can execute a variety of branch instructions that depend on the status of the CALU and accumulator. These instructions can be executed conditionally, based on various combinations of the associated status bits. For overflow management, these conditions include the OV (branch on overflow) and EQ (branch on accumulator equal to zero). In addition, the BACC (branch-to-address in accumulator) instruction provides the ability to branch to an address specified by the accumulator (computed goto). Bit test instructions (BIT and BITT), which do not affect the accumulator, allow the testing of a specified bit of a word in data memory.

The CALU also has a carry bit (bit 9 of status register ST1) that facilitates efficient computation of extended-precision products and additions or subtractions. The carry bit is also useful in overflow management. The carry bit is affected by the following operations:

- Additions to and subtractions from the accumulator:

  C = 0: When the result of a subtraction generates a borrow.

  When the result of an addition does not generate a carry. (Exception: When the ADD instruction is used with a shift of 16 and no carry is generated, the ADD instruction has no effect on C.)

  C = 1: When the result of an addition generates a carry.

  When the result of a subtraction does not generate a borrow. (Exception: When the SUB instruction is used with a shift of 16 and no borrow is generated, the SUB instruction has no effect on C.)

- Single-bit shifts and rotations of the accumulator value. During a left shift or rotation, the most significant bit of the accumulator is passed to C; during a right shift or rotation, the least significant bit is passed to C.

Note: the carry bit is set to “1” on a hardware reset.

The ADDC (add to accumulator with carry) and SUBB (subtract from accumulator with borrow) instructions provided, use the previous value of carry in their addition/subtraction operation.

accumulator

The 32-bit accumulator is the registered output of the CALU. It can be split into two 16-bit segments for storage in data memory. Shifters at the output of the accumulator provide a left shift of 0 to 7 places. This shift is performed while the data is being transferred to the data bus for storage. The contents of the accumulator remain unchanged. When the post-scaling shifter is used on the high word of the accumulator (bits 16–31), the MSBs are lost and the LSBs are filled with bits shifted in from the low word (bits 0–15). When the post-scaling shifter is used on the low word, the LSBs are zero-filled.

The SFL and SFR (in-place one-bit shift to the left/right) instructions and the ROL and ROR (rotate to the left/right) instructions implement shifting or rotating of the accumulator contents through the carry bit. The SXM bit affects the definition of the SFR (shift accumulator right) instruction. When SXM=1, SFR performs an arithmetic right shift, maintaining the sign of the accumulator data. When SXM=0, SFR performs a logical shift, shifting out the LSBs and shifting in a zero for the MSB. The SFL (shift accumulator left) instruction is not affected by the SXM bit and behaves the same in both cases, shifting out the MSB and shifting in a zero. Repeat (RPT) instructions can be used with the shift and rotate instructions for multiple-bit shifts.

auxiliary registers and auxiliary-register arithmetic unit (ARAU)

The ’F206 provides a register file containing eight auxiliary registers (AR0–AR7). The auxiliary registers are used for indirect addressing of the data memory or for temporary data storage. For indirect data memory addressing, the address of the desired memory location is placed into the selected auxiliary register. These registers are referenced with a 3-bit auxiliary register pointer (ARP) that is loaded with a value from 0 through 7, designating AR0 through AR7, respectively. The auxiliary registers and the ARP can be loaded from data memory, the ACC, the product register, or by an immediate operand defined in the instruction. The contents of these registers can also be stored in data memory or used as inputs to the CALU.

The auxiliary register file (AR0–AR7) is connected to the auxiliary register arithmetic unit (ARAU). The ARAU can autoindex the current auxiliary register while the data memory location is being addressed. Indexing either by ±1 or by the contents of the AR0 register can be performed. As a result, accessing tables of information does not require the CALU for address manipulation; therefore, the CALU is free for other operations in parallel.

memory

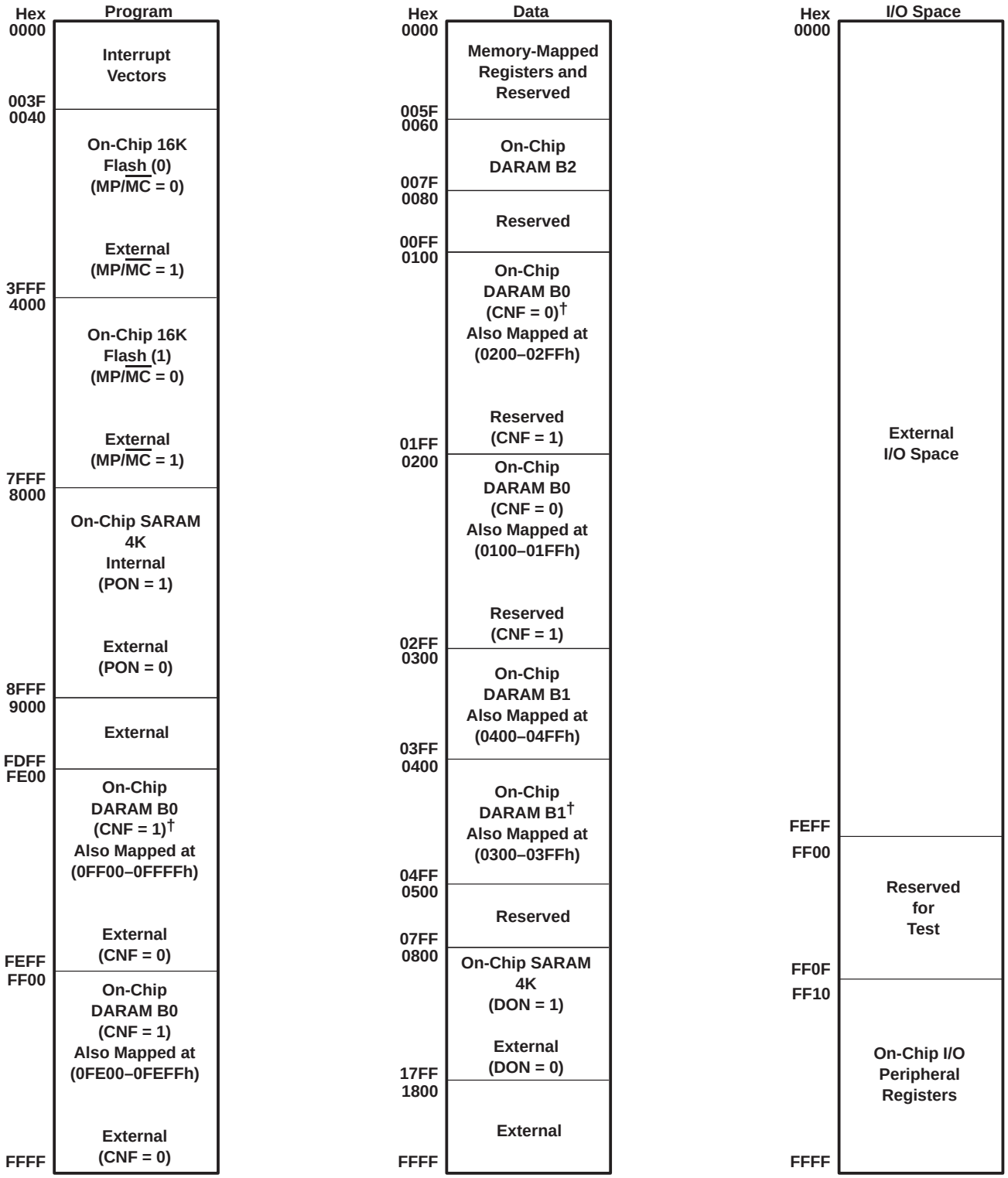
The ’F206 implements three separate address spaces for program memory, data memory, and I/O. Each space accommodates a total of 64K 16-bit words. Within the 64K words of data space, the 256 to 32K words at the top of the address range can be defined to be external global memory in increments of powers of two, as specified by the contents of the global memory allocation register (GREG). Access to global memory is arbitrated using the global memory bus request ($\overline{BR}$) signal.

On the ’F206, the first 96 (0–5Fh) data memory locations are allocated for memory-mapped registers or reserved. This memory-mapped register space contains various control and status registers including those for the CPU.

The TMS320F206 device includes 544 x 16 words of dual-access RAM (DARAM), 4K x 16 single-access RAM (SARAM), and 32K x 16 program flash memory. Table 7 shows the mapping of these memory blocks and the appropriate control bits and pins. Figure 1 shows the effects of the memory control pin MP/$\overline{MC}$ and the control bit CNF on the mapping of the respective memory spaces to on-chip or off-chip. The PON and DON bits select the SARAM (4K) mapping in program, data, or both. See Table 8 for details of the PMST register, and PON and DON bits. At reset, these bits are 11, which selects the SARAM in program and data space. The SARAM addresses are 0x800h in data and 0x8000h in program memory.

At reset, if the MP/$\overline{MC}$ pin is held high, the device is in microprocessor mode and the program address branches to 0x0000h (external program space). The MP/$\overline{MC}$ pin status is latched in the PMST register (bit 0). As long as this bit remains high, the device is in microprocessor mode. PMST register bits can be read and modified in software. If bit 0 is cleared to 0, the device enters microcontroller mode and transfers control to the on-chip flash memory at 0x0000.

The on-chip data memory blocks B0 and B1 are 256 $\times$ 16 words each, and these blocks are mapped to dual address ranges within the ’F206 memory map. For example, when CNF = 0, B0 is mapped in data space at addresses 0100–01FFh, and also at addresses 0200–02FFh. Corresponding addresses of the two ranges (0100h and 0200h, 0101h and 0201h, ...) access the same memory locations within B0. Similarly, when CNF = 1, B0 is mapped in program space at addresses 0FE00–0FEFFh, and also at addresses 0FF00–0FFFFh. The B1 block is always mapped in data space at addresses 0300–03FFh, and also at 0400–04FFh.



† DARAM blocks B0 and B1 are 256 × 16 words each; however, these memory blocks are mapped to dual address ranges within the 'F206 memory map. For more details, see the last paragraph in the memory section.

Figure 1. TMS320F206 Memory Map

memory (continued)

Table 7. TMS320F206 Memory Map

DESCRIPTION OF MEMORY BLOCK	DATA MEMORY ADDRESS	PROG MEMORY ADDRESS	MP/$\overline{MC}$†	DON†	PON†	CNF BIT†
256 x 16 word dual-access RAM (DARAM) (B0)	0x100 – 0x1FFh 0x200 – 0x2FFh‡		x	x	x	0
256 x 16 word DARAM (B0)		0xFE00 – 0xFEFFh 0xFF00 – 0xFFFFh‡	x	x	x	1
256 x 16 word DARAM (B1)	0x300 – 0x3FFh 0x400 – 0x4FFh‡		x	x	x	x
32 x 16 word DARAM (B2)	0x60 – 0x7Fh		x	x	x	x
32K x 16 word program flash memory§		0x0000 – 0x7FFFh	0	x	x	x
32K x 16 word external program memory		0x0000 – 0x7FFFh	1	x	x	x
32K x 16 word external program memory		0x8000h – 0xFFFFh	x	x	0	0
External		0x8000h – 0xFDFFh	x	x	0	1
4K x 16 word data single-access RAM (SARAM)	0x800 – 0x17FFh		x	1	x	x
4K x 16 word program SARAM		0x8000 – 0x8FFFh	x	x	1	x
4K x 16 word program and data SARAM¶	0x800 – 0x17FFh	0x8000 – 0x8FFFh	x	1	1	x
4K x 16 word SARAM	not available	not available	x	0	0	x

† Denotes don't care condition
‡ The DARAM blocks B0 and B1 are mapped to dual address ranges as shown in the table. For more details on this mapping, see the last paragraph in the memory section.
§ The 32K x 16 flash memory consists of two 16K x 16 flash modules designated by FLASH0 and FLASH1.
¶ The single SARAM (4K) block is accessible from both data and program memory space.

flash memory (EEPROM)

Flash EEPROM provides an attractive alternative to masked program ROM. Like ROM, flash is a nonvolatile memory type; however, it has the advantage of "in-target" reprogrammability. The TMS320F206 incorporates two 16K $\times$ 16-bit flash EEPROM modules which provide a contiguous 32K $\times$ 16-bit array in program space. This type of memory expands the capabilities of the TMS320F206 in the areas of prototyping, early field-testing, and single-chip applications.

Unlike most discrete flash memory, the 'F206 flash does not require a dedicated state machine, because the algorithms for programming and erasing the flash are executed by the DSP core. This enables several advantages, including: reduced chip size and sophisticated, adaptive algorithms. For production programming, the IEEE Standard 1149.1 (JTAG) scan port provides easy access to the on-chip RAM for downloading the algorithms and flash code. Other key features of the flash include zero-wait-state access rate and single 5-V power supply.

An erased bit in the TMS320F206 flash is read as a logic 1, and a programmed bit is read as a logic 0. The flash requires a block-erase of each of the two 16K modules; however, any combination of bits can be programmed. The following four algorithms are required for flash operations: clear, erase, flash-write, and program. For an explanation of these algorithms and a complete description of the flash EEPROM, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during the 2nd quarter of 1998.

flash serial loader

The on-chip flash is shipped with a serial bootloader programmed at the following addresses: 0x0000–0x00FFh. All other flash addresses are in an erased state. The serial bootloader can be used to load flash-programming algorithms or code to any destination RAM (SARAM or B0 RAM) through the on-chip UART or enhanced synchronous serial port (ESSP). Refer to the serial loader documentation to understand on-chip flash programming using the serial bootloader.

on-chip registers

The TMS320F206 includes three registers mapped to internal data space and eighteen (18) registers mapped to internal I/O space. Table 8 describes these registers and shows their respective addresses. In the table, DS refers to data space and IS refers to input/output ports.

on-chip registers (continued)

Table 8. On-Chip Memory and I/O Mapped Registers

NAME	ADDRESS	VALUE AT RESET†	DESCRIPTION
IMR	DS@0004	0000h	Interrupt mask register. This 7-bit register individually masks or enables the seven interrupts. Bit 0 shares external interrupt INT1 and HOLD. INT2 and INT3 share bit 1. Bit 2 ties to the timer interrupt, TINT. Bits 3 and 4, RINT and XINT, respectively, are for the synchronous serial port, SSP. Bit 5, TXRXINT, shares the transmit and receive interrupts for the asynchronous serial port, ASP. Bit 6 is reserved for monitor mode emulation operations and must always be set to 0 except in conjunction with emulation monitor operations. Bits 7–15 are not used in the TMS320F206.
GREG	DS@0005	0000h	Global memory allocation register. This 8-bit register specifies the size of the global memory space. GREG is set to 0 at reset.
IFR	DS@0006	0000h	Interrupt flag register. The 7-bit IFR indicates that the TMS320F206 has latched an interrupt from one of the seven maskable interrupts. Bit 0 shares external interrupt INT1 and HOLD. INT2 and INT3 share bit 1. Bit 2 ties to the timer interrupt, TINT. Bits 3 and 4, RINT and XINT, respectively, are for the synchronous serial port, SSP. Bit 5, TXRXINT shares the transmit and receive interrupts for the asynchronous serial port, ASP. Bit 6 is reserved for monitor-mode emulation operations and must always be set to 0 except in conjunction with emulation monitor operations. Writing a 1 to the respective interrupt bit clears an active flag and the respective pending interrupt. Writing a 1 to an inactive flag has no effect. Bits 7–15 are not used in the TMS320F206.
F_ACCESS0	IS@FFE0	0001h	FLASH 0 access-control register. Bit 0 selects one of two possible access modes for FLASH 0. All other bits are reserved. If bit 0 is cleared to 0, register-access mode is selected. For a detailed description of register-access mode, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998. If bit 0 is set to a 1, array-access mode is selected. In array-access mode, FLASH 0 memory array is mapped to the address range of FLASH 0. F_ACCESS0 is set to 0x0001h at reset.
F_ACCESS1	IS@FFE1	0001h	FLASH 1 access-control register. Bit 0 selects one of two possible access modes for FLASH 1. All other bits are reserved. If bit 0 is cleared to 0, register-access mode is selected. For a detailed description of register-access mode, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998. If bit 0 is set to a 1, array-access mode is selected. In array-access mode, FLASH 1 memory array is mapped to the address range of FLASH 1. F_ACCESS1 is set to 0x0001h at reset.
PMST	IS@FFE4	0006h	Bit 0 latches in the MP/$\overline{MC}$ pin at reset. This bit can be written to configure Microprocessor (1) or Microcontroller mode (0). Bits 1 and 2 configure the SARAM mapping either in program memory, data memory, or both. At reset, these bits are 11, the SARAM is mapped in both program and data space. DON (bit 2) PON (bit 1) 0 0 - SARAM not mapped, address in external memory 0 1 - SARAM in on-chip program memory at 0x8000h 1 0 SARAM in on-chip data memory at 0x800h 1 1 SARAM in on-chip program and data memory (reset value) Bit 15 – Fast $\overline{RD}$, FRDN. This bit provides software control to select an inverted R/$\overline{W}$ signal in place of the $\overline{RD}$ signal (pin 45). This is intended to help achieve zero wait-state memory interface with slow memory devices. At reset, this bit is 0 and selects $\overline{RD}$ as the signal at pin 45. If the FRDN bit is written with a 1, pin 45 is replaced with the inverted R/$\overline{W}$ signal.
CLK	IS@FFE8	0000h	CLKOUT1 on or off. At reset, bit 0 is configured as a zero for the CLKOUT1 pin to be active. If bit 0 is a 1, CLKOUT1 pin is turned off.

† 'x' indicates undefined or value based on the pin levels at reset.

on-chip registers (continued)

Table 8. On-Chip Memory and I/O Mapped Registers (Continued)

NAME	ADDRESS	VALUE AT RESET†	DESCRIPTION
ICR	IS@FFEC	0000h	Interrupt control register. This register is used to determine which interrupt is active since INT1 and HOLD share an interrupt vector as do INT2 and INT3. A portion of this register is for mask/unmask (similar to IFR). At reset, all bits are zeroed, thereby allowing the HOLD mode to be enabled. The MODE bit is used by the hold-generating circuit to determine if a HOLD or INT1 is active.
SDTR	IS@FFF0	xxxxh	Synchronous serial port (SSP) transmit and receive register
SSPCR	IS@FFF1	0030h	Synchronous serial-port control register. This register controls serial-port operation as defined by the register bits.
SSPST	IS@FFF2	0000h	Synchronous serial-port status register
SSPMC	IS@FFF3	0000h	Synchronous serial-port multichannel register
ADTR	IS@FFF4	xxxxh	Asynchronous serial port (ASP) transmit and receive register
ASPCR	IS@FFF5	0000h	Asynchronous serial-port control register (ASPCR). This register controls the asynchronous serial-port operation.
IOSR	IS@FFF6	18xxh	I/O status register. IOSR is used for detecting current levels (and changes when inputs) on pins IO0–IO3 and status of UART.
BRD	IS@FFF7	0001h	Baud-rate divisor register (baud-rate generator). 16-bit register used to determine baud rate of UART. No data is transmitted/received if BRD is zero.
TCR	IS@FFF8	0000h	Timer-control register. This 10-bit register contains the control bits that define the divide-down ratio, start/stop the timer, and reload the period. Also contained in this register is the current count in the prescaler. Reset initializes the timer divide-down ratio to 0 and starts the timer.
PRD	IS@FFF9	FFFFh	Timer-period register. This 16-bit register contains the 16-bit period that is loaded into the timer counter when the counter borrows or when the reload bit is activated. Reset initializes the PRD to 0xFFFF.
TIM	IS@FFFA	FFFFh	Timer-counter register. This 16-bit register contains the current 16-bit count of the timer. Reset initializes the TIM to 0xFFFF.
SSPCT	IS@FFFB	0000h	Synchronous serial-port counter register
WSGR	IS@FFFC	0FFFh	Wait-state generator register. This register contains 12 control bits to enable 0 to 7 wait states to program, data, and I/O space. Reset initializes WSGR to 0x0FFFh.

† 'x' indicates undefined or value based on the pin levels at reset.

external interface

The TMS320F206 can address up to 64K × 16 words of memory or registers in each of the program, data, and I/O spaces. On-chip memory, when enabled, occupies some of this off-chip range. In data space, the high 32K words can be mapped dynamically either locally or globally using the GREG register as described in the *TMS320C2xx User's Guide* (literature number SPRU127). A data-memory access that is mapped as global asserts $\overline{BR}$ low (with timing similar to the address bus).

The CPU of the TMS320F206 schedules a program fetch, data read, and data write on the same machine cycle. This is because from on-chip memory, the CPU can execute all three of these operations in the same cycle. However, the external interface multiplexes the internal buses to one address and one data bus. The external interface sequences these operations to complete first the data write, then the data read, and finally the program read.

The 'F206 supports a wide range of system interfacing requirements. Program, data, and I/O address spaces provide interface to memory and I/O, thereby maximizing system throughput. The full 16-bit address and data bus, along with the $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ space-select signals, allow addressing of 64K 16-bit words in each of the three spaces.

external interface (continued)

I/O design is simplified by having I/O treated the same way as memory. I/O devices are mapped into the I/O address space using the processor's external address and data buses in the same manner as memory-mapped devices.

The 'F206 external parallel interface provides various control signals to facilitate interfacing to the device. The R/$\overline{W}$ output signal is provided to indicate whether the current cycle is a read or a write. The $\overline{STRB}$ output signal provides a timing reference for all external cycles. For convenience, the device also provides the $\overline{RD}$ and the $\overline{WE}$ output signals, which indicate a read and a write cycle, respectively, along with timing information for those cycles. The $\overline{RD}$ pin provides additional flexibility through software control. The $\overline{RD}$ pin can be configured to provide an inverted R/$\overline{W}$ signal instead of the standard $\overline{RD}$ signal. The FRDN bit (bit 15) of the PMST register controls the $\overline{RD}$ pin signal selection. For more details on the FRDN bit control selection, see the PMST register description in Table 8. The availability of these signals minimizes external gating necessary for interfacing external devices to the 'F206.

The bus request ($\overline{BR}$) signal is used in conjunction with the other 'F206 interface signals to arbitrate external global memory accesses. Global memory is external data memory space in which the $\overline{BR}$ signal is asserted at the beginning of the access. When an external global memory device receives the bus request, it responds by asserting the READY signal after the global memory access is arbitrated and the global access is completed.

The TMS320F206 supports zero-wait-state reads on the external interface. However, to avoid bus conflicts, writes take two cycles. This allows the TMS320F206 to buffer the transition of the data bus from input to output (or output to input) by a half cycle. In most systems, TMS320F206 ratio of reads to writes is significantly large to minimize the overhead of the extra cycle on writes.

Wait states can be generated when accessing slower external resources. The wait states operate on machine-cycle boundaries and are initiated either by using the READY pin or using the software wait-state generator. The READY pin can be used to generate any number of wait states. When using the READY pin to communicate with slower devices, the 'F206 processor waits until the slower device completes its function and signals the processor by way of the READY line. Once a ready indication is provided back to the 'F206 from the external device, execution continues. For external wait states using the READY pin, the on-chip wait-state generator should be programmed to generate at least one wait state.

interrupts and subroutines

The 'F206 implements three general-purpose interrupts, $\overline{INT3}$–$\overline{INT1}$, along with reset ($\overline{RS}$) and the nonmaskable interrupt ($\overline{NMI}$) which are available for external devices to request the attention of the processor. Internal interrupts are generated by: the serial port (RINT and XINT), the timer (TINT), the UART, the TXRXINT bit in the IMR, and by the software-interrupt instructions (TRAP, INTR and NMI). Interrupts are prioritized with $\overline{RS}$ having the highest priority, followed by $\overline{NMI}$, and timer or UART having the lowest priority. Additionally, any interrupt except $\overline{RS}$ and $\overline{NMI}$ can be individually masked with a dedicated bit in the interrupt mask register (IMR) and can be cleared, set, or tested using its own dedicated bit in the interrupt flag register (IFR). The reset and NMI functions are not maskable.

All interrupt vector locations are on two-word boundaries so that branch instructions can be accommodated in those locations if desired.

A built-in mechanism protects multicycle instructions from interrupts. If an interrupt occurs during a multicycle instruction, the interrupt is not processed until the instruction completes execution. This mechanism applies to instructions that are repeated (using the RPT instruction) and to instructions that become multicycle because of wait states.

Each time an interrupt is serviced or a subroutine is entered, the PC is pushed onto an internal hardware stack, providing a mechanism for returning to the previous context. The stack contains eight locations, allowing interrupts or subroutines to be nested up to eight levels deep.

reset

The TMS320F206 utilizes an active-low reset ($\overline{RS}$) input.

A minimum pulse duration of six cycles ensures that an asynchronous reset signal resets the device properly. The TMS320F206 fetches its first instruction approximately sixteen cycles after the rising edge of $\overline{RS}$.

The reset action halts all operations whether they are completed or not; therefore, the state of the system and its data cannot be maintained through the reset operation. For example, if the device is writing to an external resource when the reset is initiated, the write is aborted; this may corrupt the contents or configuration of system resources. Therefore, it is necessary to reinitialize the system after a reset.

power-down modes

The ’F206 implements a power-down mode in which the ’F206 core enters a dormant state and dissipates less power. The power-down mode is invoked by executing an IDLE instruction. While the device is in power-down mode, the on-chip peripherals continue to operate.

While the ’F206 is in a power-down mode, all of its internal contents are maintained; this allows operation to continue unaltered when the power-down mode is terminated. All CPU activities are halted when the IDLE instruction is executed, but the CLKOUT1 pin remains active depending on the status of ICR register. The peripheral circuits continue to operate, allowing peripherals such as serial ports and timers to take the CPU out of its powered-down state. The power-down mode, when initiated by an IDLE instruction, is terminated upon receipt of an interrupt.

software-controlled wait-state generator

Due to the fast cycle time of the TMS320F206 devices, it is often necessary to operate with wait states to interface with external logic or memory. For many systems, one wait state is adequate.

The software wait-state generator can be programmed to generate between 0 and 7 wait states for a given space. Software wait states are configured by way of the wait-state generator register (WSGR). The WSGR includes four 3-bit fields to configure wait states for the following external memory spaces: data space (DSWS), upper program space (PSUWS), lower program space (PSLWS), and I/O space (ISWS). The wait-state generator enables wait states for a given memory space based on the value of the corresponding three bits, regardless of the condition of the READY signal. The READY signal can be used to generate additional wait states. All bits of the WSGR are set to 1 at reset so that the device can operate from slow memory from reset. The WSGR register (shown in Table 9, Table 10 and Table 11) resides at I/O port 0xFFFCh.

Table 9. Wait-State Generator Control Register (WSGR)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ISWS			DSWS			PSUWS			PSLWS		
0				R/W–111			R/W–111			R/W–111			R/W–111		

LEGEND:
0 = Always read as zeros, R = Read Access, W= Write Access, – n = Value after reset

software-controlled wait-state generator (continued)

Table 10. Wait-State(s) Programming

ISWS, DSWS, PSUWS, OR PSLWS BITS	WAIT STATES FOR PROGRAM, DATA, OR I/O
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Table 11. Wait-State Generator Control Register (WSGR)

BITS	NAME	DESCRIPTION
2–0	PSLWS	External program space wait states (lower). PSLWS determines that between 0 to 7 wait states are applied to all reads and writes to off-chip lower program space address (0h–7FFFh). The memory cycle can be further extended by using the READY signal. The READY signal does not override the wait states generated by PSWS. These bits are set to 1 (active) by reset ($\overline{RS}$).
5–3	PSUWS	External program space wait states (upper). PSUWS determines that between 0 to 7 wait states are applied to all reads and writes to off-chip upper program space address (8000h–0FFFFh). The memory cycle can be further extended by using the READY signal. The READY signal does not override the wait states generated by PSWS. These bits are set to 1 (active) by reset ($\overline{RS}$).
8–6	DSWS	External data space wait states. DSWS determines that between 0 to 7 wait states are applied to all reads and writes to off-chip data space. The memory cycle can be further extended by using the READY signal. The READY signal does not override the wait states generated by DSWS. These bits are set to 1 (active) by reset ($\overline{RS}$).
11–9	ISWS	External input / output space wait state. DSWS determines that between 0 to 7 wait states are applied to all reads and writes to off-chip I/O space. The memory cycle can be further extended by using the READY signal. The READY signal does not override the wait states generated by ISWS. These bits are set to 1 (active) by reset ($\overline{RS}$).
15–12	X	Don't care

timer

The TMS320F206 includes a 20-bit timer, implemented with a 16-bit main counter (TIM), and a 4-bit prescaler counter (PSC). The count values are written into the 16-bit period register (PRD), and the 4-bit timer divide-down register (TDDR). The TIM and the PRD are 16-bit registers mapped to I/O space, while the PSC and the TDDR are 4-bit fields of the timer control register (TCR). The TCR is an I/O mapped register which also includes other control bits for the timer (see Table 8).

When the timer is started, the TIM is loaded with the contents of PRD, and the PSC is loaded with the contents of the TDDR. The PSC is decremented by one at each CLKOUT1 cycle. On the CLKOUT1 cycle after the PSC decrements to zero, the PSC is reloaded with the contents of TDDR, and the TIM is decremented by one. That is, every (TDDR+1) CLKOUT1 cycles, the TIM is decremented by one. When the TIM decrements to zero, it is reloaded with the contents of the PRD on the following CLKOUT1 cycle, and a new timer interval begins. Therefore, the timer interrupt rate is defined as follows: CLKOUT1 frequency/[(TDDR+1) (PRD+1)].

The timer can be used to generate periodic CPU interrupts based on CLKOUT1. Each time the TIM decrements to zero, a timer interrupt (TINT) is generated, and a pulse equal to the duration of a CLKOUT1 cycle is generated on the TOUT pin. The timer provides a convenient means of performing periodic I/O, context switching , or other functions.

input clock options

The TMS320F206 provides multiple clock modes of divide-by-two and multiply-by-one, -two, or -four. The clock mode configuration cannot be dynamically changed without executing another reset.

synchronous serial port

A full-duplex (bidirectional) 16-bit on-chip synchronous serial port provides direct communication with serial devices such as codecs, serial A/D (analog-to-digital) converters, and other serial systems. The interface signals are compatible with codecs and many other serial devices. The serial port can also be used for intercommunication between processors in multiprocessing applications.

For data transmission, three signals are necessary to connect the transmit pins of the transmitting device with the receive pins of the receiving device. The transmitted serial data signal (DX) sends the actual data. The transmit frame synchronization signal (FSX) initiates the transfer (at the beginning of the packet), and the transmit clock signal (CLKX) clocks the bit transfer. The corresponding pins on the receive interface are DR, FSR and CLKR, respectively. When the serial port is not used, the device can be configured to shut off the serial port internal clocks, allowing the device to run in a lower power mode of operation.

The continuous mode of the synchronous serial port (SSP) provides operation that, once initiated, requires no further frame synchronization pulses when transmitting at maximum packet frequency. Both receive and transmit operations have a four-deep first-in-first-out (FIFO) buffer. The advantage of having a FIFO is to alleviate the CPU from being loaded with the task of servicing a receive- or transmit-data operation after each word, allowing a continuous communications stream of 16-bit data packets. The maximum transfer rate for both transmit and receive operations is CLKOUT1(frequency)/2. Therefore, the maximum rate at 20 million instructions per second (MIPS) is 10 megabits per second (Mbps). The serial port is fully static and functions at arbitrarily low clocking frequencies.

enhanced synchronous serial port features

The synchronous serial port of the TMS320F206 device is an enhanced synchronous serial port (ESSP). The ESSP features facilitate a glueless interface with multiple codecs and other peripherals. The SSP registers are complemented with three additional registers—ESSP status register (SSPST), ESSP multichannel register (SSPMC), and ESSP counter register (SSPCT)—to define the ESSP features. The SSPST includes control and status bits for some of the new ESSP features. Additional control bits are provided in the SSPMC to control the multichannel and prescaled clocks/frames features. The SSPCT register contains the two 8-bit prescalers to provide variable synchronous shift clock (CLKX) and frame syncs (FSX).

asynchronous serial port

The asynchronous serial port is full-duplexed and transmits and receives 8-bit data. For transmit and receive data there is one start bit and one or two configurable stop bits by way of the asynchronous serial port control register (ASPCR). Double-buffering of transmit/receive data is used in all modes. Baud-rate generation is accomplished by way of the baud-rate divisor register (BRD). This port also features auto-baud-detection logic.

scan-based emulation

TMS320F206 devices incorporate scan-based emulation logic for code- and hardware-development support. Scan-based emulation allows the emulator to control the processor in the system without the use of intrusive cables to the full pinout of the device. The scan-based emulator communicates with the ’F206 by way of the IEEE 1149.1 (JTAG) interface.

multiprocessing

The flexibility of the ’C20x allows configurations to satisfy a wide range of system requirements; the device can be used in a variety of system configurations, including but not limited to the following:

- A standalone processor
- A multiprocessor with devices in parallel
- A slave/host multiprocessor with global memory space
- A peripheral processor interfaced to another device via the processor-controlled signals

For multiprocessing applications, the ’F206 has the capability of allocating global memory space and communicating with that space by way of the $\overline{BR}$ and READY control signals. Global memory is data memory shared by more than one device. Global memory accesses must be arbitrated. The 8-bit memory-mapped global memory allocation register (GREG) specifies part of the ’C20x’s data memory as global external memory. The contents of the register determine the size of the global memory space. If the current instruction addresses an operand within that space, $\overline{BR}$ is asserted to request control of the bus. The length of the memory cycle is controlled by the READY line.

The TMS320F206 supports direct memory access (DMA) to its local (off-chip) program, data, and I/O spaces. Two signals, $\overline{HOLD}$/$\overline{INT1}$, an input to the device, and $\overline{HOLDA}$, an output, control this mechanism. The Hold feature is enabled by clearing the mode bit in the interrupt control register (ICR IS@FFECh). When the Hold feature is enabled, and $\overline{HOLD}$/$\overline{INT1}$ is asserted, executing an IDLE instruction puts the address, data, and memory control signals ($\overline{PS}$, $\overline{DS}$, $\overline{IS}$, $\overline{STRB}$, R/$\overline{W}$, and $\overline{WE}$) in a high-impedance state. When this occurs, the $\overline{HOLDA}$ signal is asserted, acknowledging that the processor has relinquished control of the external bus. It is important to note that when the mode bit is set to one, the Hold feature is disabled, and the $\overline{HOLD}$/$\overline{INT1}$ pin functions as a general-purpose interrupt ($\overline{INT1}$). That is, when the Hold feature is disabled, and $\overline{HOLD}$/$\overline{INT1}$ is asserted, the IDLE instruction does not cause the memory interface signals to enter the high-impedance mode, and it does not cause the assertion of $\overline{HOLDA}$. At reset, the mode bit is cleared to zero, and the Hold feature is enabled.

instruction set

The ’C20x microprocessor implements a comprehensive instruction set that supports both numeric-intensive signal-processing operations and general-purpose applications, such as multiprocessing and high-speed control. Source code for the ’C1x and ’C2x DSPs is upwardly compatible with the ’C20x.

For maximum throughput, the next instruction is prefetched while the current one is being executed. Because the same data lines are used to communicate to external data, program, or I/O space, the number of cycles an instruction requires to execute varies depending upon whether the next data operand fetch is from internal or external memory. Highest throughput is achieved by maintaining data memory on chip and using either internal or fast external program memory.

addressing modes

The ’C20x instruction set provides four basic memory-addressing modes: direct, indirect, immediate, and register.

In direct addressing, the instruction word contains the lower seven bits of the data memory address. This field is concatenated with the nine bits of the data memory page pointer (DP) to form the 16-bit data memory address. Thus, in the direct-addressing mode, data memory is effectively paged with a total of 512 pages, each page containing 128 words.

Indirect addressing accesses data memory through the auxiliary registers. In this addressing mode, the address of the instruction operand is contained in the currently selected auxiliary register. Eight auxiliary registers (AR0–AR7) provide flexible and powerful indirect addressing. To select a specific auxiliary register, the auxiliary register pointer (ARP) is loaded with a value from 0 to 7 for AR0 through AR7, respectively.

addressing modes (continued)

There are seven types of indirect addressing: autoincrement or autodecrement, postindexing by either adding or subtracting the contents of AR0, single indirect addressing with no increment or decrement, and bit-reversed addressing [used in Fast Fourier Transforms (FFTs)] with increment or decrement. All operations are performed on the current auxiliary register in the same cycle as the original instruction, following which the current auxiliary register and ARP can be modified.

In immediate addressing, the actual operand data is provided in a portion of the instruction word or words. There are two types of immediate addressing: long and short. In short-immediate addressing, the data is contained in a portion of the bits in a single-word instruction. In long-immediate addressing, the data is contained in the second word of a two-word instruction. The immediate-addressing mode is useful for data that does not need to be stored or used more than once during the course of program execution, such as initialization values, constants, etc.

The register-addressing mode uses operands in CPU registers either explicitly, such as with a direct reference to a specific register, or implicitly with instructions that intrinsically reference certain registers. In either case, operand reference is simplified because 16-bit values can be used without specifying a full 16-bit operand address or immediate value.

repeat feature

The repeat function can be used with instructions (as defined in Table 13) such as multiply/accumulates (MAC and MACD), block moves (BLDD and BLPD), I/O transfers (IN/OUT), and table read/writes (TBLR/TBLW). These instructions, although normally multicycle, are pipelined when the Repeat feature is used, and they effectively become single-cycle instructions. For example, the table-read instruction may take three or more cycles to execute, but when the instruction is repeated, a table location can be read every cycle.

When using the repeat feature, the repeat counter (RPTC) is loaded with the addressed data memory location if direct or indirect addressing is used, or an 8-bit immediate value if short-immediate addressing is used. The RPTC register is loaded by the RPT instruction. This results in a maximum of N + 1 executions of a given instruction, when RPTC is loaded with N. RPTC is cleared by reset. Once a repeat instruction (RPT) is decoded, all interrupts, including NMI (except reset), are masked until the completion of the repeat loop.

instruction set summary

This section summarizes the opcodes of the instruction set for the ’F206 digital signal processor (DSP). This instruction set is a superset of the ’C1x and ’C2x instruction sets. The instructions are arranged according to function and are alphabetized by mnemonic within each category. The symbols in Table 12 are used in the instruction set summary table (Table 13). The Texas Instruments ’C20x assembler accepts ’C2x instructions.

The number of words that an instruction occupies in program memory is specified in column 3 of Table 13. Several instructions specify two values separated by a slash mark (/) for the number of words. In these cases, different forms of the instruction occupy a different number of words. For example, the ADD instruction occupies one word when the operand is a short-immediate value or two words if the operand is a long-immediate value.

The number of cycles that an instruction requires to execute is in column 3 of Table 13. All instructions are assumed to be executed from internal program memory (RAM) and internal data dual-access memory. The cycle timings are for single-instruction execution, not for repeat mode.

instruction set summary (continued)

Table 12. Opcode Symbols

SYMBOL	DESCRIPTION
A	Address
ACC	Accumulator
ACCB	Accumulator buffer
ARx	Auxiliary register value (0–7)
BITx	4-bit field that specifies which bit to test for the BIT instruction
BMAR	Block-move address register
DBMR	Dynamic bit-manipulation register
I	Addressing-mode bit
II...II	Immediate operand value
INTM	Interrupt-mode flag bit
INTR#	Interrupt vector number
K	Constant
PREG	Product register
PROG	Program memory
RPTC	Repeat counter
SHF, SHFT	3/4-bit shift value
TC	Test-control bit
T P	Two bits used by the conditional execution instructions to represent the conditions TC, NTC, and BIO. T P Meaning 0 0 $\overline{\text{BIO}}$ low 0 1 TC = 1 1 0 TC = 0 1 1 None of the above conditions
TREGn	Temporary register n (n = 0, 1, or 2)
Z L V C	4-bit field representing the following conditions: Z: ACC = 0 L: ACC < 0 V: Overflow C: Carry A conditional instruction contains two of these 4-bit fields. The 4-LSB field of the instruction is a 4-bit mask field. A 1 in the corresponding mask bit indicates that the condition is being tested. The second 4-bit field (bits 4–7) indicates the state of the conditions designated by the mask bits as being tested. For example, to test for ACC ≥ 0, the Z and L fields are set while the V and C fields are not set. The next 4-bit field contains the state of the conditions to test. The Z field is set to indicate testing of the condition ACC = 0, and the L field is reset to indicate testing of the condition ACC ≥ 0. The conditions possible with these 8 bits are shown in the BCND and CC instructions. To determine if the conditions are met, the 4-LSB bit mask is ANDed with the conditions. If any bits are set, the conditions are met.

instruction set summary (continued)

Table 13. TMS320F206 Instruction Set Summary

’x20x MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE MSB			LSB
ABS	Absolute value of accumulator	1/1	1011	1110	0000	0000
ADD	Add to accumulator with shift	1/1	0010	SHFT	IADD	RESS
	Add to high accumulator	1/1	0110	0001	IADD	RESS
	Add to accumulator short immediate	1/1	1011	1000	KKKK	KKKK
	Add to accumulator long immediate with shift	2/2	1011	1111	1001	SHFT
ADDC	Add to accumulator with carry	1/1	0110	0000	IADD	RESS
ADDS	Add to low accumulator with sign extension suppressed	1/1	0110	0010	IADD	RESS
ADDT	Add to accumulator with shift specified by T register	1/1	0110	0011	IADD	RESS
ADRK	Add to auxiliary register short immediate	1/1	0111	1000	KKKK	KKKK
AND	AND with accumulator	1/1	0110	1110	IADD	RESS
	AND immediate with accumulator with shift	2/2	1011	1111	1011	SHFT 16-Bit Constant
	AND immediate with accumulator with shift of 16	2/2	1011	1110	1000	0001 16-Bit Constant
APAC	Add P register to accumulator	1/1	1011	1110	0000	0100
B	Branch unconditionally	2/4	0111	1001	IADD	RESS Branch Address
BACC	Branch to address specified by accumulator	1/4	1011	1110	0010	0000
BANZ	Branch on auxiliary register not zero	2/4/2	0111	1011	IADD	RESS Branch Address
BCND	Branch if TC bit ≠ 0	2/4/2	1110	0001	0000	0000 Branch Address
	Branch if TC bit = 0	2/4/2	1110	0010	0000	0000 Branch Address
	Branch on carry	2/4/2	1110	0011	0001	0001 Branch Address
	Branch if accumulator ≥ 0	2/4/2	1110	0011	1000	1100 Branch Address
	Branch if accumulator > 0	2/4/2	1110	0011	0000	0100 Branch Address
	Branch on I/O status low	2/4/3	1110	0000	0000	0000 Branch Address
	Branch if accumulator ≤ 0	2/4/2	1110	0011	1100	1100 Branch Address
	Branch if accumulator < 0	2/4/2	1110	0011	0100	0100 Branch Address
	Branch on no carry	2/4/2	1110	0011	0000	0001 Branch Address
	Branch if no overflow	2/4/2	1110	0011	0000	0010 Branch Address

instruction set summary (continued)

Table 13. TMS320F206 Instruction Set Summary (Continued)

’x20x MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
BCND	Branch if accumulator ≠ 0	2/4/2	1110	0011	0000	1000
			Branch Address			
	Branch on overflow	2/4/2	1110	0011	0010	0010
			Branch Address			
	Branch if accumulator = 0	2/4/2	1110	0011	1000	1000
			Branch Address			
BIT	Test bit	1/1	0100	BITx	IADD	RESS
BITT	Test bit specified by TREG	1/1	0110	1111	IADD	RESS
BLDD†	Block move from data memory to data memory source immediate	2/3	1010	1000	IADD	RESS
			Branch Address			
	Block move from data memory to data memory destination immediate	2/3	1010	1001	IADD	RESS
			Branch Address			
BLPD	Block move from program memory to data memory	2/3	1010	0101	IADD	RESS
			Branch Address			
CALA	Call subroutine indirect	1/4	1011	1110	0011	0000
CALL	Call subroutine	2/4	0111	1010	IADD	RESS
			Routine Address			
CC	Conditional call subroutine	2/4/2	1110	10TP	ZLVC	ZLVC
			Routine Address			
CLRC	Configure block as data memory	1/1	1011	1110	0100	0100
	Enable interrupt	1/1	1011	1110	0100	0000
	Reset carry bit	1/1	1011	1110	0100	1110
	Reset overflow mode	1/1	1011	1110	0100	0010
	Reset sign-extension mode	1/1	1011	1110	0100	0110
	Reset test/control flag	1/1	1011	1110	0100	1010
	Reset external flag	1/1	1011	1110	0100	1100
CMPL	Complement accumulator	1/1	1011	1110	0000	0001
CMPR	Compare auxiliary register with auxiliary register AR0	1/1	1011	1111	0100	01CM
DMOV	Data move in data memory	1/1	0111	0111	IADD	RESS
IDLE	Idle until interrupt	1/1	1011	1110	0010	0010
IN	Input data from port	2/2	1010	1111	IADD	RESS
			16BIT	I/O	PORT	ADRS
INTR	Software-interrupt	1/4	1011	1110	011K	KKKK
LACC	Load accumulator with shift	1/1	0001	SHFT	IADD	RESS
	Load accumulator long immediate with shift	2/2	1011	1111	1000	SHFT
			16-Bit Constant			
	Zero low accumulator and load high accumulator	1/1	0110	1010	IADD	RESS

† In ’C20x devices, the BLDD instruction cannot be used with memory-mapped registers IMR, IFR, and GREG.

instruction set summary (continued)

Table 13. TMS320F206 Instruction Set Summary (Continued)

'x20x MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
LACL	Load accumulator immediate short	1/1	1011	1001	KKKK	KKKK
	Zero accumulator	1/1	1011	1001	0000	0000
	Zero low accumulator and load high accumulator	1/1	0110	1010	IADD	RESS
	Zero low accumulator and load low accumulator with no sign extension	1/1	0110	1001	IADD	RESS
LACT	Load accumulator with shift specified by T register	1/1	0110	1011	IADD	RESS
LAR	Load auxiliary register	1/2	0000	0ARx	IADD	RESS
	Load auxiliary register short immediate	1/2	1011	0ARx	KKKK	KKKK
	Load auxiliary register long immediate	2/2	1011	1111	0000	1ARx
			16-Bit Constant			
LDP	Load data-memory page pointer	1/2	0000	1101	IADD	RESS
	Load data-memory page pointer immediate	1/2	1011	110P	AGEP	OINT
LPH	Load high-P register	1/1	0111	0101	IADD	RESS
LST	Load status register ST0	1/2	0000	1110	IADD	RESS
	Load status register ST1	1/2	0000	1111	IADD	RESS
LT	Load TREG	1/1	0111	0011	IADD	RESS
LTA	Load TREG and accumulate previous product	1/1	0111	0000	IADD	RESS
LTD	Load TREG, accumulate previous product, and move data	1/1	0111	0010	IADD	RESS
LTP	Load TREG and store P register in accumulator	1/1	0111	0001	IADD	RESS
LTS	Load TREG and subtract previous product	1/1	0111	0100	IADD	RESS
MAC	Multiply and accumulate	2/3	1010	0010	IADD	RESS
			16-Bit Constant			
MACD	Multiply and accumulate with data move	2/3	1010	0011	IADD	RESS
			16-Bit Constant			
MAR	Load auxiliary register pointer	1/1	1000	1011	1000	1ARx
	Modify auxiliary register	1/1	1000	1011	IADD	RESS
MPY	Multiply (with TREG, store product in P register)	1/1	0101	0100	IADD	RESS
	Multiply immediate	1/1	110C	KKKK	KKKK	KKKK
MPYA	Multiply and accumulate previous product	1/1	0101	0000	IADD	RESS
MPYS	Multiply and subtract previous product	1/1	0101	0001	IADD	RESS
MPYU	Multiply unsigned	1/1	0101	0101	IADD	RESS
NEG	Negate accumulator	1/1	1011	1110	0000	0010
NMI	Nonmaskable interrupt	1/4	1011	1110	0101	0010
NOP	No operation	1/1	1000	1011	0000	0000
NORM	Normalize contents of accumulator	1/1	1010	0000	IADD	RESS
OR	OR with accumulator	1/1	0110	1101	IADD	RESS
	OR immediate with accumulator with shift	2/2	1011	1111	1100	SHFT
			16-Bit Constant			
	OR immediate with accumulator with shift of 16	2/2	1011	1110	1000	0010
			16-Bit Constant			
OUT	Output data to port	2/3	0000 16BIT	1100 I/O	IADD PORT	RESS ADRS
PAC	Load accumulator with P register	1/1	1011	1110	0000	0011

instruction set summary (continued)

Table 13. TMS320F206 Instruction Set Summary (Continued)

’x20x MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
POP	Pop top of stack to low accumulator	1/1	1011	1110	0011	0010
POPD	Pop top of stack to data memory	1/1	1000	1010	IADD	RESS
PSHD	Push data-memory value onto stack	1/1	0111	0110	IADD	RESS
PUSH	Push low accumulator onto stack	1/1	1011	1110	0011	1100
RET	Return from subroutine	1/4	1110	1111	0000	0000
RETC	Conditional return from subroutine	1/4/2	1110	11TP	ZLVC	ZLVC
ROL	Rotate accumulator left	1/1	1011	1110	0000	1100
ROR	Rotate accumulator right	1/1	1011	1110	0000	1101
RPT	Repeat instruction as specified by data-memory value	1/1	0000	1011	IADD	RESS
	Repeat instruction as specified by immediate value	1/1	1011	1011	KKKK	KKKK
SACH	Store high accumulator with shift	1/1	1001	1SHF	IADD	RESS
SACL	Store low accumulator with shift	1/1	1001	0SHF	IADD	RESS
SAR	Store auxiliary register	1/1	1000	0ARx	IADD	RESS
SBRK	Subtract from auxiliary register short immediate	1/1	0111	1100	KKKK	KKKK
SETC	Set carry bit	1/1	1011	1110	0100	1111
	Configure block as program memory	1/1	1011	1110	0100	0101
	Disable interrupt	1/1	1011	1110	0100	0001
	Set overflow mode	1/1	1011	1110	0100	0011
	Set test/control flag	1/1	1011	1110	0100	1011
	Set external flag XF	1/1	1011	1110	0100	1101
	Set sign-extension mode	1/1	1011	1110	0100	0111
SFL	Shift accumulator left	1/1	1011	1110	0000	1001
SFR	Shift accumulator right	1/1	1011	1110	0000	1010
SPAC	Subtract P register from accumulator	1/1	1011	1110	0000	0101
SPH	Store high-P register	1/1	1000	1101	IADD	RESS
SPL	Store low-P register	1/1	1000	1100	IADD	RESS
SPM	Set P register output shift mode	1/1	1011	1111	IADD	RESS
SQRA	Square and accumulate	1/1	0101	0010	IADD	RESS
SQRS	Square and subtract previous product from accumulator	1/1	0101	0011	IADD	RESS
SST	Store status register ST0	1/1	1000	1110	IADD	RESS
	Store status register ST1	1/1	1000	1111	IADD	RESS
SPLK	Store long immediate to data memory	2/2	1010	1110	IADD	RESS
			16-Bit Constant			
SUB	Subtract from accumulator long immediate with shift	2/2	1011	1111	1010	SHFT
			16-Bit Constant			
	Subtract from accumulator with shift	1/1	0011	SHFT	IADD	RESS
	Subtract from high accumulator	1/1	0110	0101	IADD	RESS
	Subtract from accumulator short immediate	1/1	1011	1010	KKKK	KKKK

instruction set summary (continued)

Table 13. TMS320F206 Instruction Set Summary (Continued)

'x20x MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
SUBB	Subtract from accumulator with borrow	1/1	0110	0100	IADD	RESS
SUBC	Conditional subtract	1/1	0000	1010	IADD	RESS
SUBS	Subtract from low accumulator with sign extension suppressed	1/1	0110	0110	IADD	RESS
SUBT	Subtract from accumulator with shift specified by TREG	1/1	0110	0111	IADD	RESS
TBLR	Table read	1/3	1010	0110	IADD	RESS
TBLW	Table write	1/3	1010	0111	IADD	RESS
TRAP	Software interrupt	1/4	1011	1110	0101	0001
XOR	Exclusive-OR with accumulator	1/1	0110	1100	IADD	RESS
	Exclusive-OR immediate with accumulator with shift	2/2	1011	1111	1101	SHFT
			16-Bit Constant			
	Exclusive-OR immediate with accumulator with shift of 16	2/2	1011	1110	1000	0011
			16-Bit Constant			
ZALR	Zero low accumulator and load high accumulator with rounding	1/1	0110	1000	IADD	RESS

development support

Texas Instruments offers an extensive line of development tools for the 'x20x generation of DSPs, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules.

The following products support development of 'x20x-based applications:

Software Development Tools:
Assembler/Linker
Simulator
Optimizing ANSI C Compiler
Application Algorithms
C/Assembly Debugger and Code Profiler

Hardware Development Tools:
Emulator XDS510™ (supports 'x20x multiprocessor system debug)

The *TMS320 Family Development Support Reference Guide* (literature number SPRU011) contains information about development support products for all TMS320 family member devices, including documentation. Refer to this document for further information about TMS320 documentation or any other TMS320 support products from Texas Instruments. There is also an additional document, the *TMS320 Third-Party Support Reference Guide* (literature number SPRU052), which contains information about TMS320-related products from other companies in the industry. To receive copies of TMS320 literature, contact the Literature Response Center at 800/477-8924.

See Table 14 for complete listings of development support tools for the 'C20x. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

XDS510 is a trademark of Texas Instruments Incorporated.

development support (continued)

Table 14. TMS320C20x Development Support Tools

DEVELOPMENT TOOL	PLATFORM	PART NUMBER
SOFTWARE		
Compiler/Assembler/Linker	SPARC™, HP™	TMDS3242555-08
Compiler/Assembler/Linker	PC-DOS™, OS/2™	TMDS3242855-02
Assembler/Linker	PC-DOS, OS/2	TMDS3242850-02
Simulator	PC-DOS, WIN™	TMDS3245851-02
Simulator	SPARC	TMDS3245551-09
Digital Filter Design Package	PC-DOS	DFDP
Debugger/Emulation Software	PC-DOS, OS/2, WIN	TMDS3240120
Debugger/Emulation Software	SPARC	TMDS3240620
Code Composer™ Debugger	Windows™	CCMSP5XWIN
HARDWARE		
C2xx Evaluation Module	PC-DOS	TMDS32600XX
XDS510XL™ Emulator	PC-DOS, OS/2	TMDS00510
XDS510WS™ Emulator	SPARC	TMDS00510WS

WIN and Windows are trademarks of Microsoft Corporation.
Code Composer is a trademark of Go DSP Inc.
SPARC is a trademark of SPARC International, Inc.
PC-DOS and OS/2 are trademarks of International Business Machines Corp.
HP is a trademark of Hewlett-Packard Company.
XDS510XL and XDS510WS are trademarks of Texas Instruments Incorporated.

device and development support tool nomenclature

To designate the stages in the product development cycle, Texas Instruments assigns prefixes to the part numbers of all TMS320 devices and support tools. Each TMS320 member has one of three prefixes: TMX, TMP, and TMS. Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS). This development flow is defined below.

Device Development Evolutionary Flow:

TMX Experimental device that is not necessarily representative of the final device's electrical specifications

TMP Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification

TMS Fully-qualified production device

Support Tool Development Evolutionary Flow:

TMDX Development support product that has not yet completed Texas Instruments internal qualification testing

TMDS Fully qualified development support product

TMX and TMP devices and TMDX development support tools are shipped against the following disclaimer:

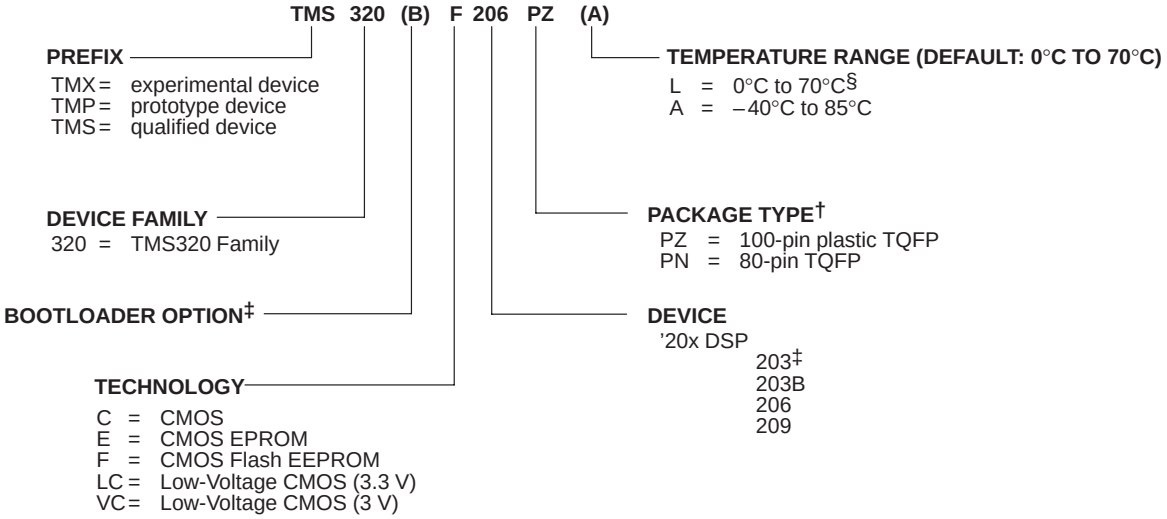
"Developmental product is intended for internal evaluation purposes."

TMS devices and TMDS development support tools have been fully characterized, and the quality and reliability of the device have been fully demonstrated. Texas Instruments standard warranty applies.

Predictions show that prototype devices (TMX or TMP) will have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate is still undefined. Only qualified production devices are to be used.

device and development support tool nomenclature (continued)

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, PZ) and temperature range (for example, A). The following figures provide a legend for reading the complete device name for any TMS320 family member.



† TQFP = Thin Quad Flat Package
‡ The TMS320C203 is a bootloader device without the B option.
§ For TMS320F206PZ devices with this temperature range, L is not printed on package.

Figure 2. TMS320x20x Device Nomenclature

documentation support

Extensive documentation supports all of the TMS320 family generations of devices from product announcement through applications development. The types of documentation available include: data sheets, such as this document, with design specifications; complete user's guides for all devices and development support tools; and hardware and software applications.

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). Also available is the *Calculation of TMS320C2xx Power Dissipation* application report (literature number SPRA088).

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. The TMS320 DSP bulletin board service (BBS) provides access to information pertaining to the TMS320 family, including documentation, source code, and object code for many DSP algorithms and utilities. The BBS can be reached at 281/274-2323.

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 operating free-air temperature range (unless otherwise noted) †

Supply voltage range, V_{DD} (see Note 1) . . . . . – 0.3 V to 7 V
Input voltage range . . . . . – 0.3 V to 7 V
Output voltage range . . . . . – 0.3 V to 7 V
Operating free-air temperature range, T_A (TMS320F206PZ) . . . . . 0°C to 70°C
(TMS320F206PZA) . . . . . – 40°C to 85°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.

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

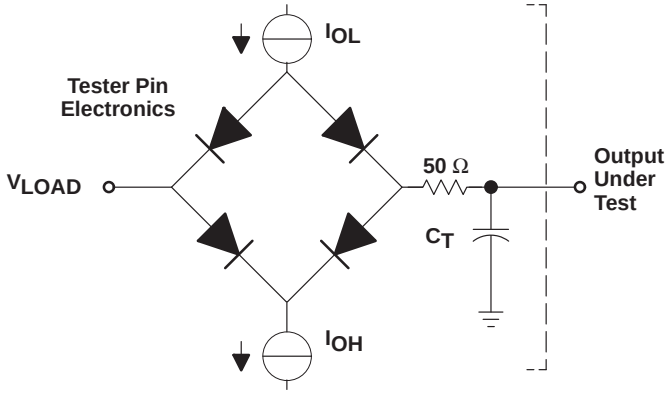
recommended operating conditions

			MIN	NOM	MAX	UNIT
V_{DD}	Supply voltage	5-V operation	4.75	5	5.25	V
V_{SS}	Supply voltage			0		V
V_{IH}	High-level input voltage	CLKIN/X2	3.2		V_{DD} + 0.3	V
		$\overline{RS}$, CLKR, CLKX, RX	2.3		V_{DD} + 0.3	
		$\overline{TRST}$, TCK	2.5		V_{DD} + 0.3	
		All other inputs	2.0		V_{DD} + 0.3	
V_{IL}	Low-level input voltage	CLKIN/X2	– 0.3		0.7	V
		$\overline{RS}$, CLKR, CLKX, RX	– 0.3		0.6	
		All other inputs	– 0.3		0.8	
I_{OH}	High-level output current				– 300	µA
I_{OL}	Low-level output current				2	mA
T_A	Operating free-air temperature	TMS320F206PZ	0		70	°C
		TMS320F206PZA	– 40		85	
Θ_{JA}	Thermal resistance, junction-to-ambient				58	°C/W
Θ_{JC}	Thermal resistance, junction-to-case				10	°C/W

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage	5-V operation, I_{OH} = MAX		2.4			V
V_{OL}	Low-level output voltage	5-V operation, I_{OL} = MAX				0.6	V
I_I	Input current	$V_I = V_{DD}$ or 0 V	CLKIN/X2	– 400		400	µA
			All other inputs	– 10		10	
I_{OZ}	Output current, high-impedance state (off-state)	$V_O = V_{DD}$ or 0 V	EMU0, EMU1 (with internal pullup)	– 60		20	µA
			TEST, FSX, FSR, CLKR, CLKX (with internal pulldown)	– 20		250	
			All other 3-state outputs	– 20		20	
I_{DD}	Supply current, core CPU	5-V operation, f_{CLKOUT} = 20.48 MHz			76		mA
C_i	Input capacitance				15		pF
C_o	Output capacitance				15		pF

PARAMETER MEASUREMENT INFORMATION



Where: I_{OL} = 2 mA (all outputs)
I_{OH} = 300 µA (all outputs)
V_{LOAD} = 1.5 V
C_T = 60-pF typical load-circuit capacitance

Figure 3. Test Load Circuit

signal-transition levels

The data in this section is shown for the 5-V version (’x20x). Note that some of the signals use different reference voltages, see the recommended operating conditions table. TTL-output levels are driven to a minimum logic-high level of 2.4 V and to a maximum logic-low level of 0.6 V.

Figure 4 shows the TTL-level outputs.

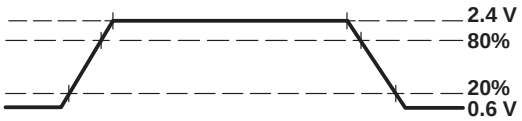


Figure 4. TTL-Level Outputs

TTL-output transition times are specified as follows:

- For a *high-to-low transition*, the level at which the output is said to be no longer high is below 80% of the total voltage range and lower and the level at which the output is said to be low is 20% of the total voltage range and lower.
- For a *low-to-high transition*, the level at which the output is said to be no longer low is 20% of the total voltage range and higher and the level at which the output is said to be high is 80% of the total voltage range and higher.

PARAMETER MEASUREMENT INFORMATION

Figure 5 shows the TTL-level inputs.

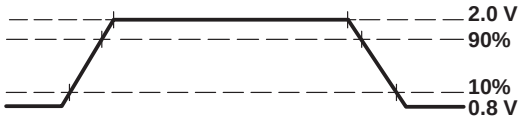


Figure 5. TTL-Level Inputs

TTL-compatible input transition times are specified as follows:

- For a *high-to-low transition* on an input signal, the level at which the input is said to be no longer high is 90% of the total voltage range and lower and the level at which the input is said to be low is 10% of the total voltage range and lower.
- For a *low-to-high transition* on an input signal, the level at which the input is said to be no longer low is 10% of the total voltage range and higher and the level at which the input is said to be high is 90% of the total voltage range and higher.

PARAMETER MEASUREMENT INFORMATION

timing parameter symbology

Timing parameter symbols used are created in accordance with JEDEC Standard 100-A. To shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

A	Address or A[15:0]
CI	CLKIN/X2
CLKR	Serial-port receive clock
CLKX	Serial-port transmit clock
CO	CLKOUT1
D	Data or D[15:0]
FR	FSR
FX	FSX
H	$\overline{\text{HOLD}}$
HA	$\overline{\text{HOLDA}}$
IN	INTN; $\overline{\text{BIO}}$, $\overline{\text{INT1}}$–$\overline{\text{INT3}}$, $\overline{\text{NMI}}$
IO	IOx : IO0, IO1, IO2, or IO3
M	Address, data, and control signals: (A, D, MS, S, $\overline{\text{BR}}$, RD, W, and R/$\overline{\text{W}}$)
MS	Memory strobe pins $\overline{\text{IS}}$, $\overline{\text{DS}}$, or $\overline{\text{PS}}$
R	READY
RD	Read cycle or $\overline{\text{RD}}$
RS	RESET pins RS or $\overline{\text{RS}}$
S	$\overline{\text{STRB}}$ or Synchronous
TP	Transitory phase
W	Write cycle or $\overline{\text{WE}}$

Lowercase subscripts and their meanings are:

a	access time
c	cycle time (period)
d	delay time
dis	disable time
en	enable time
f	fall time
h	hold time
r	rise time
su	setup time
t	transition time
v	valid time
w	pulse duration (width)

The following letters and symbols and their meanings are:

H	High
L	Low
IV	Invalid
HZ	High impedance
X	Unknown, changing, or don't care level

general notes on timing parameters

All output signals from the TMS320x20x devices (including CLKOUT1) are specified from an internal clock such that all output transitions for a given half cycle occur with a minimum of skewing relative to each other.

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

CLOCK CHARACTERISTICS AND TIMING

clock options

PARAMETER	DIV2	DIV1
Internal divide-by-two with external crystal or external oscillator	0	0
PLL multiply-by-one	0	1
PLL multiply-by-two	1	0
PLL multiply-by-four	1	1

internal divide-by-two clock option with external crystal

The internal oscillator is enabled by connecting a crystal across X1 and CLKIN/X2. The crystal should be in either fundamental or overtone operation and parallel resonant, with an effective series resistance of 30 Ω and a power dissipation of 1 mW; it should be specified at a load capacitance of 20 pF. Note that overtone crystals require an additional tuned LC circuit. Figure 6 shows an external crystal (fundamental frequency) connected to the on-chip oscillator.

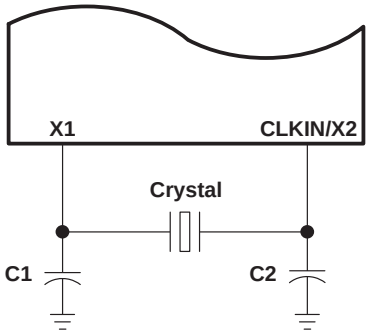


Figure 6. Internal Clock Option

timing at V_{DD} = 5 V with the PLL circuit disabled, divide-by-two mode†

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
f_X	Input clock frequency	T_A = –40°C to 85°C, 5 V	0†	40.96	MHz

† This device is implemented in static logic and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz, but is tested at f_X = 6.7 MHz to meet device test time requirements.

switching characteristics over recommended operating conditions (see Figure 7) [H = 0.5$t_{c(CO)}$]

PARAMETER		’320F206-40			UNIT
		MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	48.8	$2t_{c(CI)}$	‡	ns
$t_{d(CIH-CO)}$	Delay time, CLKIN high to CLKOUT1 high/low	1	11	20	ns
$t_{f(CO)}$	Fall time, CLKOUT1		5§		ns
$t_{r(CO)}$	Rise time, CLKOUT1		5§		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 4		H + 1	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3		H + 3	ns

‡ This device is implemented in static logic and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz, but is tested at $t_{c(CI)}$ = 300 ns to meet device test time requirements.
§ Values specified from characterization data and not tested

timing requirements over recommended operating conditions (see Figure 7)

		’320F206-40		UNIT
		MIN	MAX	
$t_{c(CI)}$	Cycle time, CLKIN	24.4	¶	ns
$t_{f(CI)}$	Fall time, CLKIN§		5	ns
$t_{r(CI)}$	Rise time, CLKIN§		5	ns
$t_{w(CIL)}$	Pulse duration, CLKIN low	11	¶	ns
$t_{w(CIH)}$	Pulse duration, CLKIN high	11	¶	ns

§ Values specified from characterization data and not tested
¶ This device is implemented in static logic and therefore can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz, but is tested at a minimum $t_{c(CI)}$ = 150 ns to meet device test time requirements.

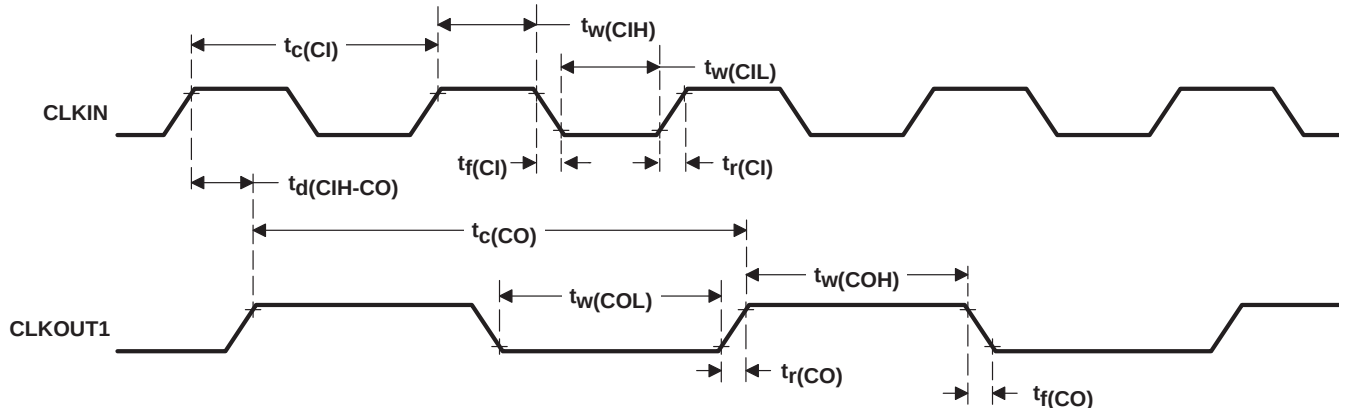


Figure 7. CLKIN-to-CLKOUT1 Timing Without PLL (using ÷2 clock option)

timing @ V_{DD} = 5 V with the PLL circuit enabled

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
f_x	Input clock frequency, multiply-by-one mode	T_A = –40°C to 85°C, 5 V	4†	20.48	MHz
	Input clock frequency, multiply-by-two mode		4†	10.24	
	Input clock frequency, multiply-by-four mode		4†	5.12	

† Values specified from characterization data and not tested

switching characteristics over recommended operating conditions (see Figure 8) [H = 0.5$t_{c(CO)}$]

PARAMETER		’320F206-40			UNIT
		MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	48.8		250	ns
$t_{f(CO)}$	Fall time, CLKOUT1†		5		ns
$t_{r(CO)}$	Rise time, CLKOUT1†		5		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low‡	H – 3	H	H + 1	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high‡	H – 1	H	H + 3	ns
$t_{d(TP)}$	Delay time, transitory phase—PLL synchronized after CLKIN supplied			2500	cycles

† Values specified from characterization data and not tested
‡ Values specified from design data and not tested

timing requirements over recommended operating conditions (see Figure 8)

		’320F206-40		UNIT
		MIN	MAX	
$t_{c(CI)}$	Cycle time, CLKIN multiply-by-one mode	48.8		ns
	Cycle time, CLKIN multiply-by-two mode	97.7		
	Cycle time, CLKIN multiply-by-four mode	195.3		
$t_{f(CI)}$	Fall time, CLKIN†		4	ns
$t_{r(CI)}$	Rise time, CLKIN†		4	ns
$t_{w(CIL)}$	Pulse duration, CLKIN low	21	125	ns
$t_{w(CIH)}$	Pulse duration, CLKIN high	21	125	ns

† Values specified from characterization data and not tested

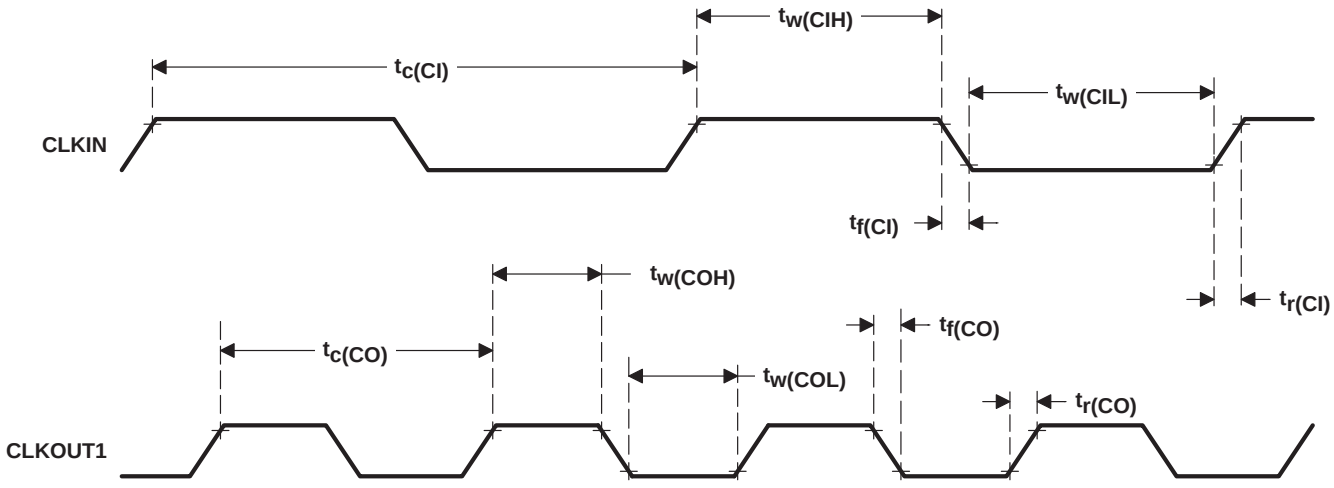


Figure 8. CLKIN-to-CLKOUT1 Timing With PLL (Enabled)

MEMORY AND PERIPHERAL INTERFACE TIMING

memory and parallel I/O interface *read* timing

A15–A0, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, R/$\overline{W}$, and $\overline{BR}$ timings are all included in the timings referenced to A15–A0 except when in transition between a read operation following a write operation or a write operation following a read operation, where $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ pulse high [see $t_{w(MS)}$].

switching characteristics over recommended operating conditions (see Figure 9) [H = 0.5$t_{c(CO)}$]

PARAMETER		'320F206-40		UNIT
		MIN	MAX	
$t_{su(A-RD)}$	Setup time, address valid before $\overline{RD}$ low	H – 4		ns
$t_{h(RD-A)}$	Hold time, address valid after $\overline{RD}$ high	– 6		ns
$t_{d(COL-A)}$	Delay time, CLKOUT1 low to read address valid		5	ns
$t_{h(COL-A)RD}$	Hold time, read address valid after CLKOUT1 low	– 5		ns
$t_{d(CO-RD)}$	Delay time, CLKOUT1 high/low to $\overline{RD}$ low/high	– 2	4	ns
$t_{d(COL-S)}$	Delay time, CLKOUT1 low to $\overline{STRB}$ low/high	4†	11	ns
$t_{w(RDL)}$	Pulse duration, $\overline{RD}$ low (no wait states)	H – 3	H + 2	ns
$t_{w(RDH)}$	Pulse duration, $\overline{RD}$ high	H – 4	H	ns

† Values specified from characterization data and not tested

timing requirements over recommended operating conditions (see Figure 9) [H = 0.5$t_{c(CO)}$]

		'320F206-40		UNIT
		MIN	MAX	
$t_{a(A)}$	Access time, from address valid to read data		2H – 18	ns
$t_{su(D-RD)}$	Setup time, read data before $\overline{RD}$ high	15		ns
$t_{h(RD-D)}$	Hold time, read data after $\overline{RD}$ high	– 2		ns
$t_{h(AIV-D)}$	Hold time, read data after address invalid	0		ns
$t_{su(D–COL)RD}$	Setup time, read data before CLKOUT1 low	15		ns
$t_{h(COL-D)RD}$	Hold time, read data after CLKOUT1 low	– 2		ns
$t_{a(RD)}$	Access time, from $\overline{RD}$ low to read data		H – 15	ns
$t_{a(S)}$	Access time, from $\overline{STRB}$ low to read data†		2H – 21	ns

† Values specified from characterization data and not tested

MEMORY AND PERIPHERAL INTERFACE TIMING (CONTINUED)

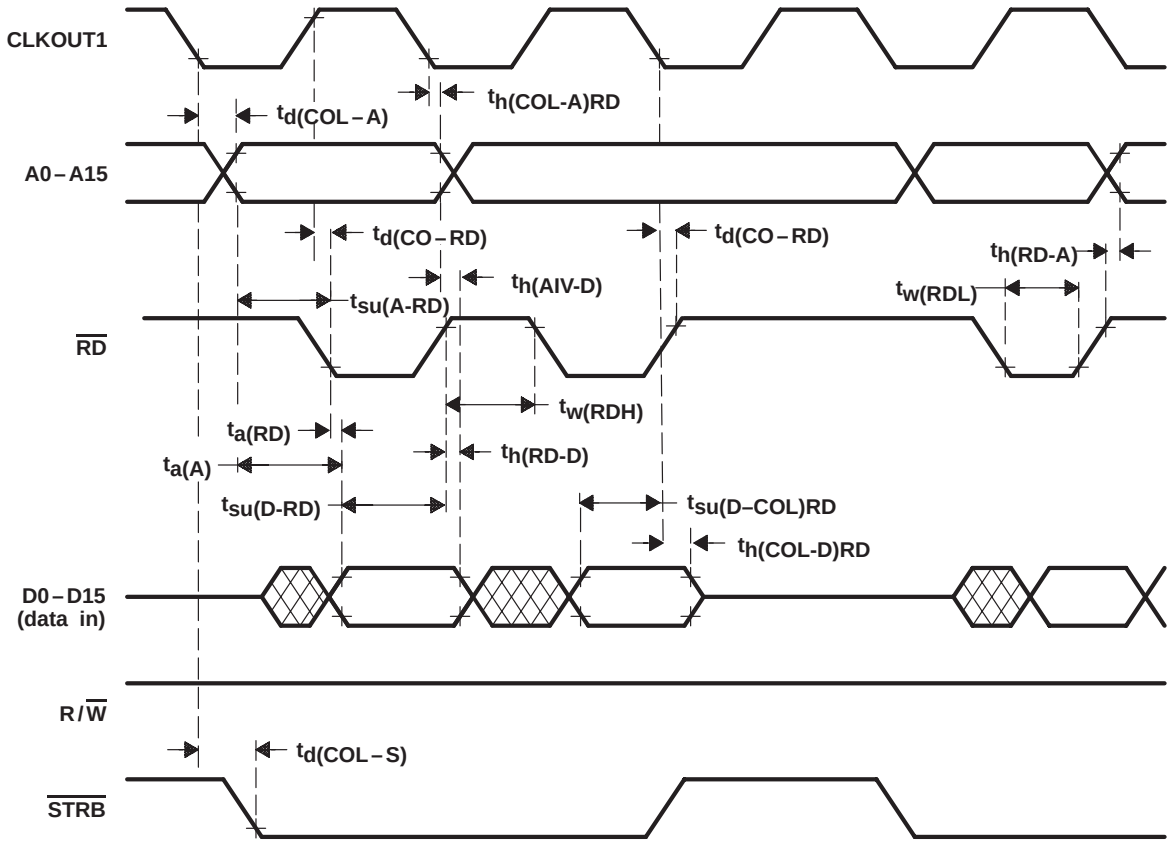


Figure 9. Memory Interface Read Timing

MEMORY AND PERIPHERAL INTERFACE TIMING (CONTINUED)

memory and parallel I/O interface *write* timing

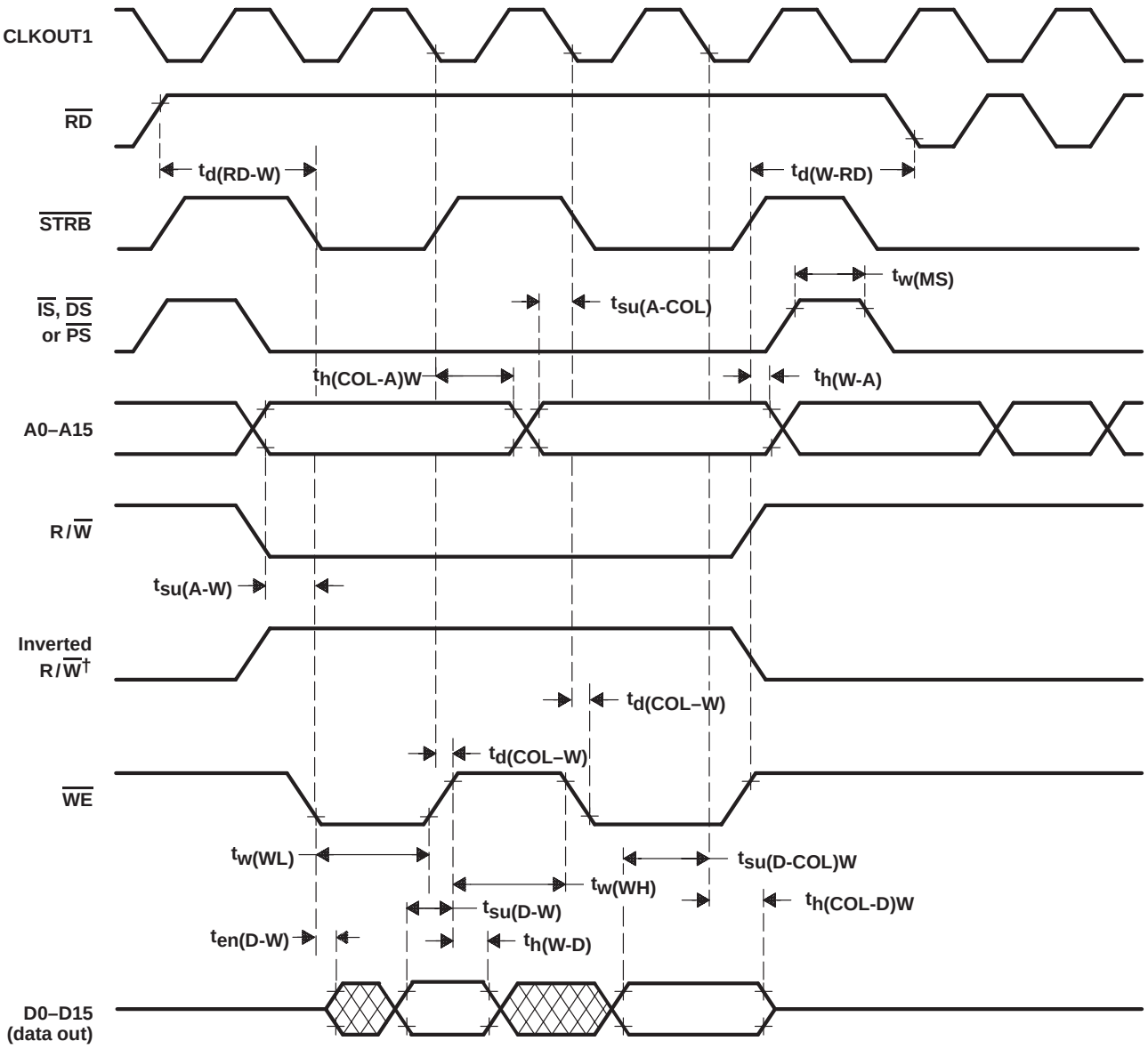
A15–A0, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, R/$\overline{W}$, and $\overline{BR}$ timings are all included in the timings referenced to A15–A0 except when in transition between a read operation following a write operation or a write operation following a read operation, where $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ pulse high [see $t_{w(MS)}$].

switching characteristics over recommended operating conditions (see Figure 10) [H = 0.5$t_{c(CO)}$]

PARAMETER		'320F206-40		UNIT
		MIN	MAX	
$t_{su(A-W)}$	Setup time, address valid before $\overline{WE}$ low	H – 7		ns
$t_{h(W-A)}$	Hold time, address valid after $\overline{WE}$ high	H – 10		ns
$t_{su(A-COL)}$	Setup time, write address valid before CLKOUT1 low	H – 9		ns
$t_{h(COL-A)W}$	Hold time, write address valid after CLKOUT1 low	H – 5		ns
$t_{w(MS)}$	Pulse duration, $\overline{IS}$, $\overline{DS}$, $\overline{PS}$ inactive high†	H – 9		ns
$t_{w(WL)}$	Pulse duration, $\overline{WE}$ low (no wait states)	2H – 5	2H	ns
$t_{w(WH)}$	Pulse duration, $\overline{WE}$ high	2H – 4		ns
$t_{d(COL-W)}$	Delay time, CLKOUT1 low to $\overline{WE}$ low/high	– 2	4	ns
$t_{d(RD-W)}$	Delay time, $\overline{RD}$ high to $\overline{WE}$ low	2H – 8		ns
$t_{d(W-RD)}$	Delay time, $\overline{WE}$ high to $\overline{RD}$ low	3H – 8		ns
$t_{su(D-W)}$	Setup time, write data valid before $\overline{WE}$ high	2H – 16	2H†	ns
$t_{h(W-D)}$	Hold time, write data valid after $\overline{WE}$ high	3	14‡	ns
$t_{su(D-COL)W}$	Setup time, write data valid before CLKOUT1 low	2H – 17	2H†	ns
$t_{h(COL-D)W}$	Hold time, write data valid after CLKOUT1 low	2	14‡	ns
$t_{en(D-W)}$	Enable time, data bus driven from $\overline{WE}$†	3		ns

† Values specified from characterization data and not tested
‡ Values specified from design data and not tested

MEMORY AND PERIPHERAL INTERFACE TIMING (CONTINUED)



† If the FRDN bit in the PMST register (FFE4h) is a 1, then the signal issued from the RD pin (pin 45) is an inverted R/W signal (or fast RD) replacing the RD signal.

Figure 10. Memory Interface Write Timing

MEMORY AND PERIPHERAL INTERFACE TIMING (CONTINUED)

READY timing

timing requirements over recommended operating conditions (see Figure 11) [H = 0.5$t_{c(CO)}$]

		’320F206-40		UNIT
		MIN	MAX	
$t_{su(R-CO)}$	Setup time, READY before CLKOUT1 rising edge	16		ns
$t_{h(CO-R)}$	Hold time, READY after CLKOUT1 rising edge	0		ns
$t_{su(R-RD)}$	Setup time, READY before $\overline{RD}$ falling edge	16		ns
$t_{h(RD-R)}$	Hold time, READY after $\overline{RD}$ falling edge	0		ns
$t_{v(R-W)}$	Valid time, READY after $\overline{WE}$ falling edge	H – 17		ns
$t_{h(W-R)}$	Hold time, READY after $\overline{WE}$ falling edge	H + 4		ns
$t_{v(R-A)RD}$	Valid time, READY after address valid on read		H – 19	ns
$t_{v(R-A)W}$	Valid time, READY after address valid on write		2H – 22	ns

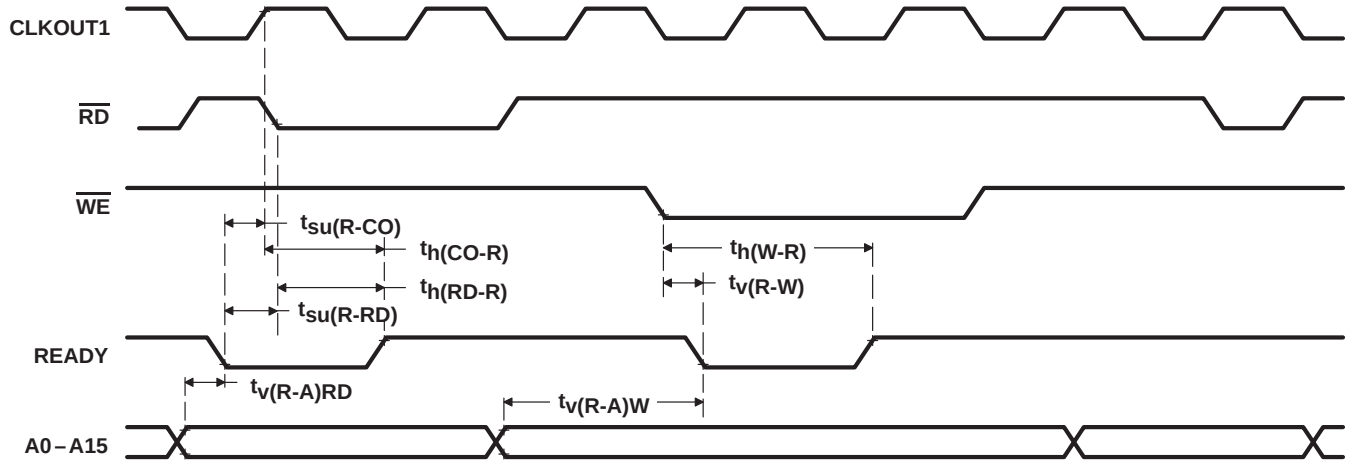


Figure 11. READY Timing

XF, TOUT, $\overline{RS}$, $\overline{INT1}$ – $\overline{INT3}$, $\overline{NMI}$, and $\overline{BIO}$ timing

switching characteristics over recommended operating conditions (see Figure 12) [H = 0.5$t_{c(CO)}$]

PARAMETER		'320F206-40		UNIT
		MIN	MAX	
$t_{d(COH-XF)}$	Delay time, CLKOUT1 high to XF valid	– 1†	13	ns
$t_{d(COL-TOUT)}$	Delay time, CLKOUT1 low to TOUT high/low	0†	17	ns
$t_{w(TOUT)}$	Pulse duration, TOUT high	2H – 8†		ns

† Values specified from characterization data and not tested

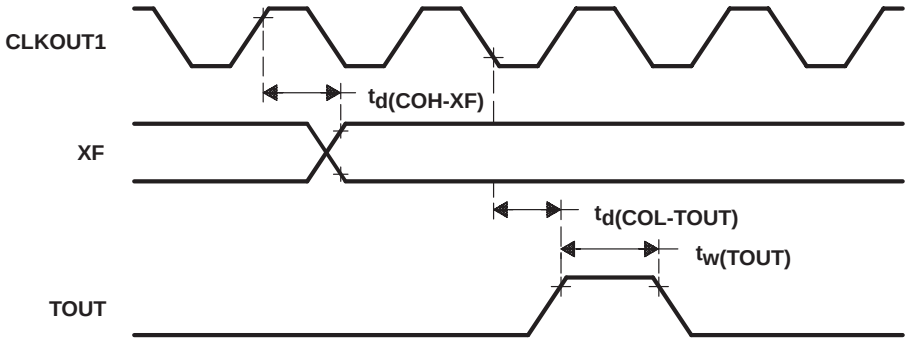


Figure 12. XF and TOUT Timing

XF, TOUT, $\overline{RS}$, $\overline{INT1}$ – $\overline{INT3}$, $\overline{NMI}$, and $\overline{BIO}$ timing (continued)

timing requirements over recommended operating conditions† (see Figure 13 and Figure 14) [H = 0.5$t_{c(CO)}$]

		’320F206-40		UNIT
		MIN	MAX	
$t_{su(RS-CIL)}$	Setup time, $\overline{RS}$ before CLKIN low	11		ns
$t_{su(RS-COL)}$	Setup time, $\overline{RS}$ before CLKOUT1 low	16		ns
$t_{w(RSL)}$	Pulse duration, $\overline{RS}$ low‡	12H		ns
$t_{d(RS-RST)}$	Delay time, $\overline{RS}$ high to reset-vector fetch	34H		ns
$t_{su(IN-COLS)}$	Setup time, INTN before CLKOUT1 low (synchronous)	10		ns
$t_{h(COLS-IN)}$	Hold time, INTN after CLKOUT1 low (synchronous)	0		ns
$t_{w(IN)}$	Pulse duration, INTN low	2H + 18		ns
$t_{d(IN-INT)}$	Delay time, INTN low to interrupt-vector fetch	12H		ns

† INTN: $\overline{BIO}$, $\overline{INT1}$ – $\overline{INT3}$, $\overline{NMI}$

‡ This parameter assumes the CLKIN to be stable before $\overline{RS}$ goes active.

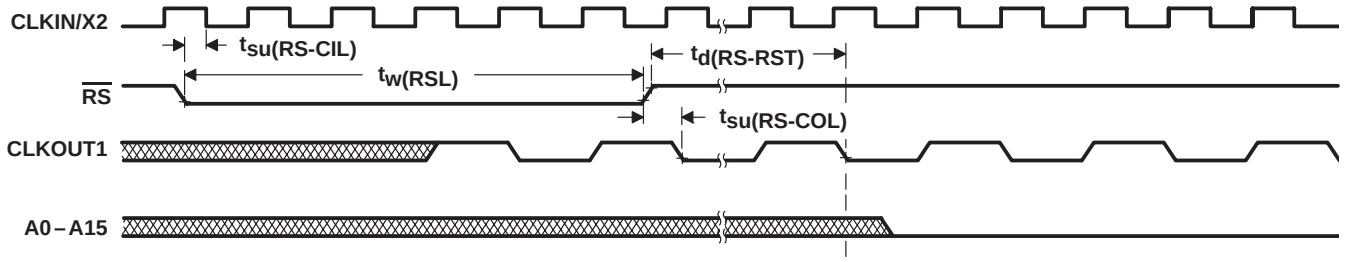
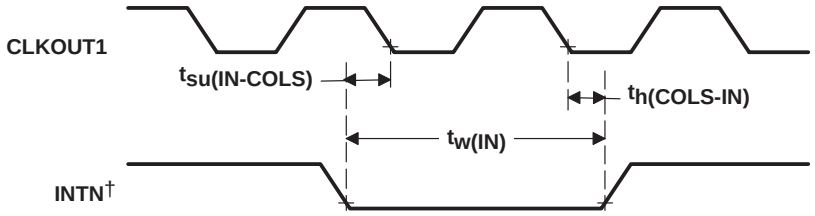


Figure 13. Reset Timing



† INTN: $\overline{BIO}$, $\overline{INT1}$ – $\overline{INT3}$, $\overline{NMI}$

Figure 14. Interrupts and $\overline{BIO}$ Timing

external DMA timing

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

PARAMETER		'320F206-40		UNIT
		MIN	MAX	
$t_{d(CO-HA)}$	Delay time, CLKOUT1 rising to $\overline{\text{HOLDA}}$		9	ns
$t_{d(HL-HAL)}$	Delay time, $\overline{\text{HOLD}}$ low to $\overline{\text{HOLDA}}$ low†	4H		ns
$t_{d(HH-HAH)}$	Delay time, $\overline{\text{HOLD}}$ high to $\overline{\text{HOLDA}}$ high	2H		ns
$t_{hz(M-HAL)}$	Address high impedance before $\overline{\text{HOLDA}}$ low‡§	H – 5		ns
$t_{en(HAH-M)}$	Enable time, address driven from $\overline{\text{HOLDA}}$ high§	H – 5		ns

† The delay values will change based on the software logic (IDLE instruction) that activates $\overline{\text{HOLDA}}$. See the *TMS320C2xx User's Guide* (literature number SPRU127) for functional description of HOLD logic.
‡ This parameter includes all memory control lines.
§ Values specified from characterization data and not tested

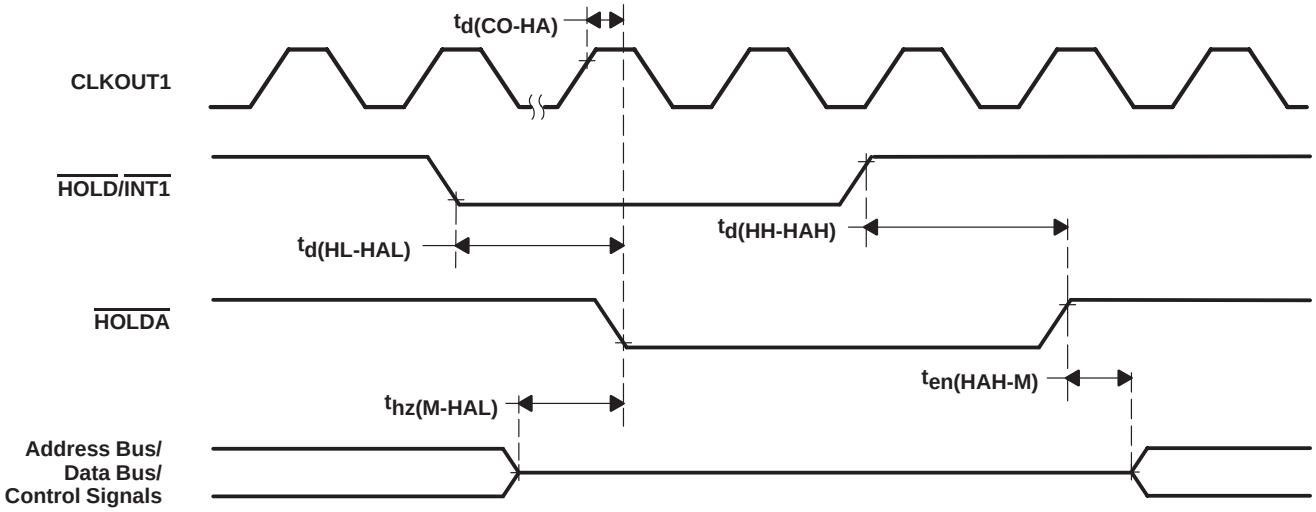


Figure 15. External DMA Timing

serial-port receive timing

timing requirements over recommended ranges of supply voltage and operating free-air temperature (see Figure 16) [H = 0.5$t_{c(CO)}$]

		’320F206-40		UNIT
		MIN	MAX	
$t_{c(CLKR)}$	Cycle time, serial-port clock (CLKR)	4H		ns
$t_{f(CLKR)}$	Fall time, serial-port clock (CLKR)†		8	ns
$t_{r(CLKR)}$	Rise time, serial-port clock (CLKR)†		8	ns
$t_{w(CLKR)}$	Pulse duration, serial-port clock (CLKR) low/high	2H		ns
$t_{su(FR-CLKR)}$	Setup time, FSR before CLKR falling edge	10		ns
$t_{su(DR-CLKR)}$	Setup time, DR before CLKR falling edge	10		ns
$t_{h(CLKR-FR)}$	Hold time, FSR after CLKR falling edge	10		ns
$t_{h(CLKR-DR)}$	Hold time, DR after CLKR falling edge	10		ns

† Values specified from characterization data and not tested

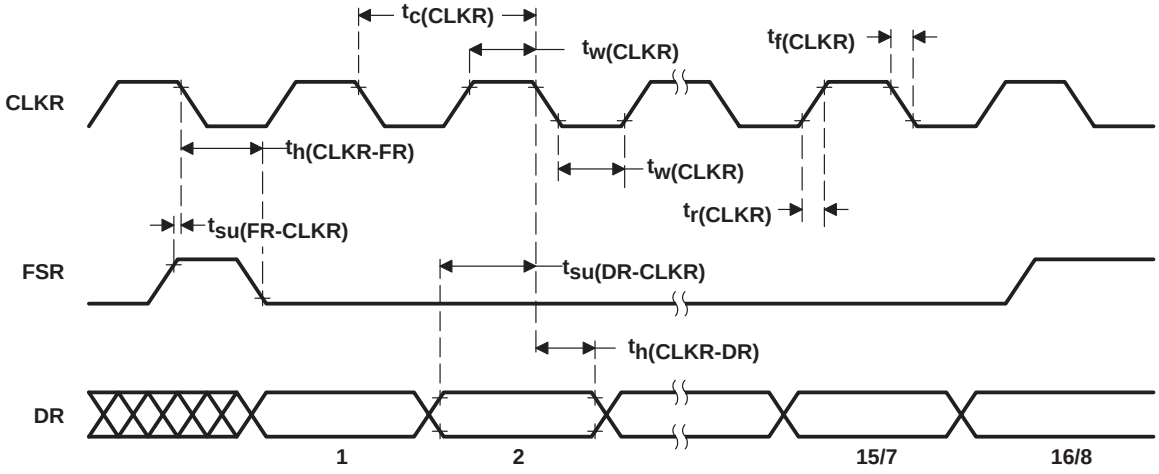


Figure 16. Serial-Port Receive Timing

serial-port transmit timings (note: timings are for all SSP modes unless otherwise specified)

switching characteristics over recommended operating conditions (see Figure 17) [H = 0.5$t_{c(CO)}$]

PARAMETER		TEST CONDITIONS	'320F206-40			UNIT
			MIN	TYP	MAX	
$t_{d(CLKX-DX)}$	Delay time, CLKX high to DX valid	Internal CLKX†	– 5		22	ns
		External CLKX†	0		20	
		Multichannel mode	– 5†		27	
		SPI mode†	– 5		4	
$t_{dis(DX-CLKX)}$	Disable time, DX valid from CLKX high†				40	ns
$t_{h(CLKX-DX)}$	Hold time, DX valid after CLKX high†		– 6			ns
$t_{c(CLKX)}$	Cycle time, serial-port clock (CLKX)	Internal CLKX		4H		ns
$t_{f(CLKX)}$	Fall time, serial-port clock (CLKX)†	Internal CLKX		5		ns
$t_{r(CLKX)}$	Rise time, serial-port clock (CLKX)†	Internal CLKX		5		ns
$t_{w(CLKX)}$	Pulse duration, serial-port clock (CLKX) low/high	Internal CLKX	2H – 10			ns
$t_{d(CLKX-FX)}$	Delay time, CLKX rising edge to FSX	Internal FSX†	5		14	ns
		Multichannel mode‡	5†		25	
		SPI mode†	– 5		2	
$t_{h(CLKXH-FX)}$	Hold time, FSX after CLKX rising edge	Internal FSX†	– 5			ns

† Values specified from characterization data and not tested
‡ These timings also apply to the following pins in multichannel mode: CLKR, FSR, IO0.

timing requirements over recommended ranges of supply voltage and operating free-air temperature (see Figure 17) [H = 0.5$t_{c(CO)}$]

			'320F206-40		UNIT
			MIN	MAX	
$t_{c(CLKX)}$	Cycle time, serial-port clock (CLKX)	External CLKX	4H		ns
$t_{f(CLKX)}$	Fall time, serial-port clock (CLKX)†	External CLKX		8	ns
$t_{r(CLKX)}$	Rise time, serial-port clock (CLKX)†	External CLKX		8	ns
$t_{w(CLKX)}$	Pulse duration, serial-port clock (CLKX) low/high	External CLKX	2H		ns
$t_{d(CLKX-FX)}$	Delay time, CLKX rising edge to FSX	External FSX		2H – 10	ns
$t_{h(CLKX-FX)}$	Hold time, FSX after CLKX falling edge	External FSX	10		ns
$t_{h(CLKXH-FX)}$	Hold time, FSX after CLKX rising edge	External FSX†		2H – 8	ns

† Values specified from characterization data and not tested

serial-port transmit timings (note: timings are for all SSP modes unless otherwise specified) (continued)

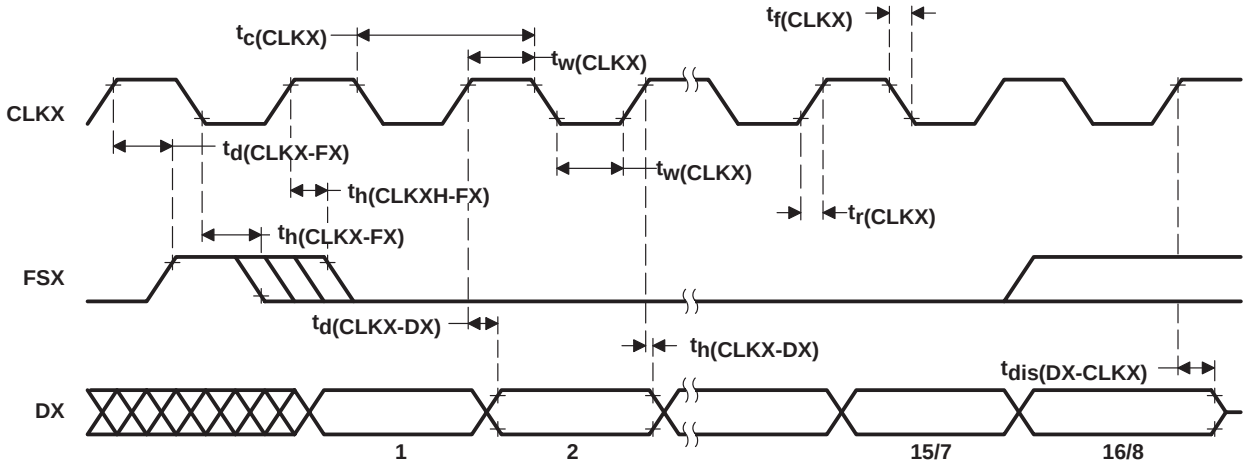


Figure 17. Serial-Port Transmit Timings

general-purpose input/output (I/O) pin timings

switching characteristics over recommended operating conditions† (see Figure 18)

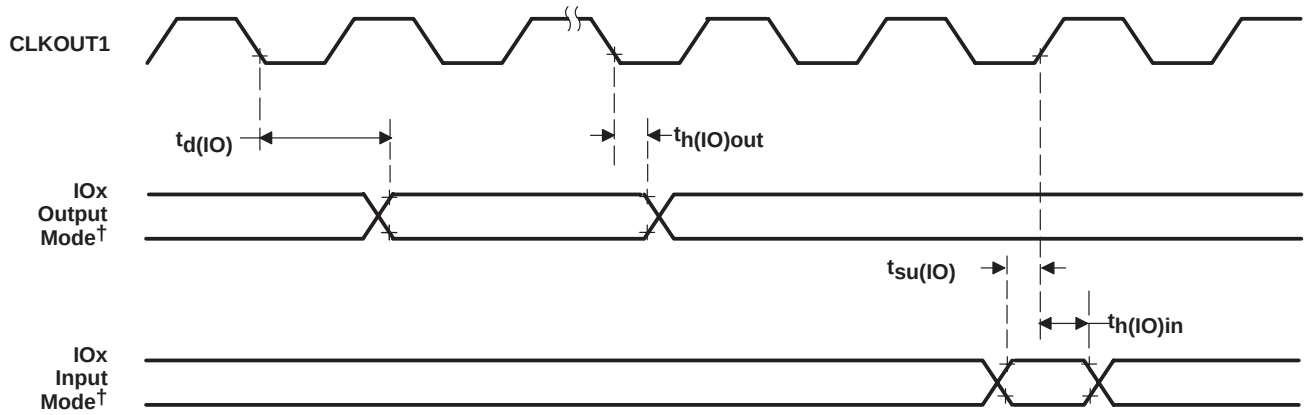
PARAMETER		’320F206-40		UNIT
		MIN	MAX	
$t_{d(IO)}$	Delay time, CLKOUT1 falling edge to IOx output valid		13	ns
$t_{h(IO)out}$	Hold time, IOx output valid after CLKOUT1 falling edge	0		ns

† Values specified from characterization data and not tested.

timing requirements over recommended operating conditions† (see Figure 18)

		’320F206-40		UNIT
		MIN	MAX	
$t_{su(IO)}$	Setup time, IOx input valid before CLKOUT1 rising edge	6		ns
$t_{h(IO)in}$	Hold time, IOx input valid after CLKOUT1 rising edge	0		ns

† Values specified from characterization data and not tested.



† IOx represents IO0, IO1, IO2, or IO3 input/output pins.

Figure 18. General-Purpose I/O Timings

flash EEPROM

switching characteristics over recommended operating conditions

PARAMETER	'320F206-40			UNIT
	MIN	TYP	MAX	
Program-erase endurance	10K			Cycles
Data retention	10			Years
Program pulses per word†	1	10	150	Pulses
Erase pulses per array†	1	20	1000	Pulses
Flash-write pulses per array†	1	20	6000	Pulses

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998.

timing requirements over recommended operating conditions

		'320F206-40		UNIT
		MIN	MAX	
$t_{d(BUSY)}$	Delay time, after mode deselect to stabilization†	10		µs
$t_{d(RD\text{-}VERIFY)}$	Delay time, verify read mode select to stabilization†	10		µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998.

programming operation

PARAMETER		'320F206-40			UNIT
		MIN	NOM	MAX	
$t_{w(PGM)}$	Pulse duration, programming algorithm†	95‡	100	105‡	µs
$t_{d(PGM\text{-}MODE)}$	Delay time, program mode select to stabilization†	10			µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998.
‡ Values specified from characterization data and not tested.

erase operation

PARAMETER		'320F206-40			UNIT
		MIN	NOM	MAX	
$t_{w(ERASE)}$	Pulse duration, erase algorithm†	6.65‡	7	7.35‡	ms
$t_{d(ERASE\text{-}MODE)}$	Delay time, erase mode select to stabilization†	10			µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998.
‡ Values specified from characterization data and not tested.

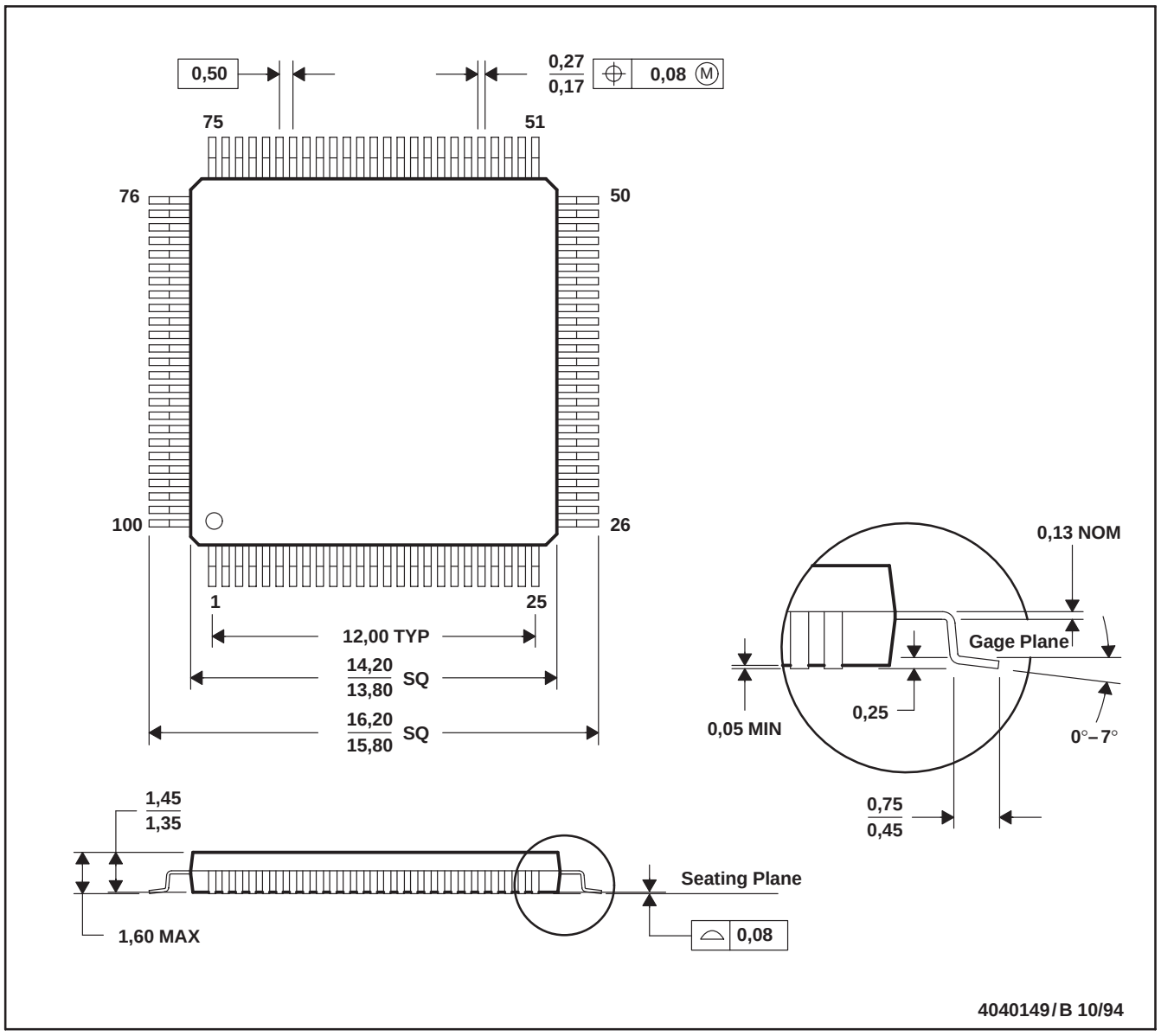
flash-write operation

PARAMETER		'320F206-40			UNIT
		MIN	NOM	MAX	
$t_{w(FLW)}$	Pulse duration, flash-write algorithm†	13.3‡	14	14.7‡	ms
$t_{d(FLW\text{-}MODE)}$	Delay time, flash-write mode select to stabilization†	10			µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, refer to the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) available during 2nd quarter of 1998.
‡ Values specified from characterization data and not tested.

MECHANICAL DATA

PZ (S-PQFP-G100) **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MO-136

Thermal Resistance Characteristics

PARAMETER	°C/W
Θ_{JA}	58
Θ_{JC}	10

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current and complete.

TI warrants performance of its semiconductor products and related software 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.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

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

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI 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.

Copyright © 1998, Texas Instruments Incorporated

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View TMS320F206PZ on WIN SOURCE
- Texas Instruments Information

Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management