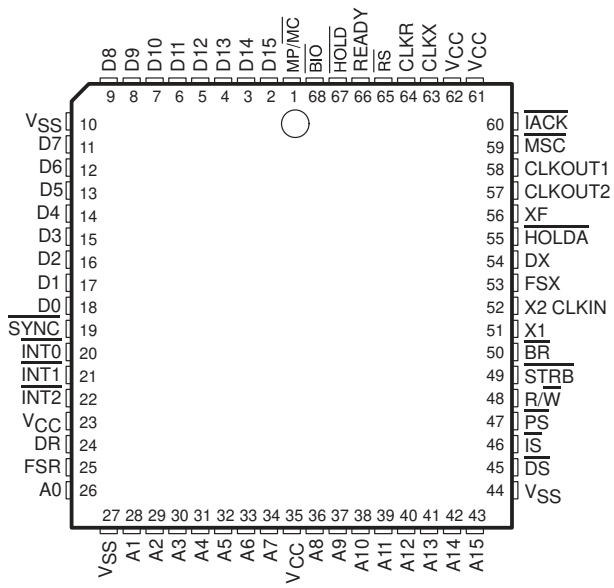
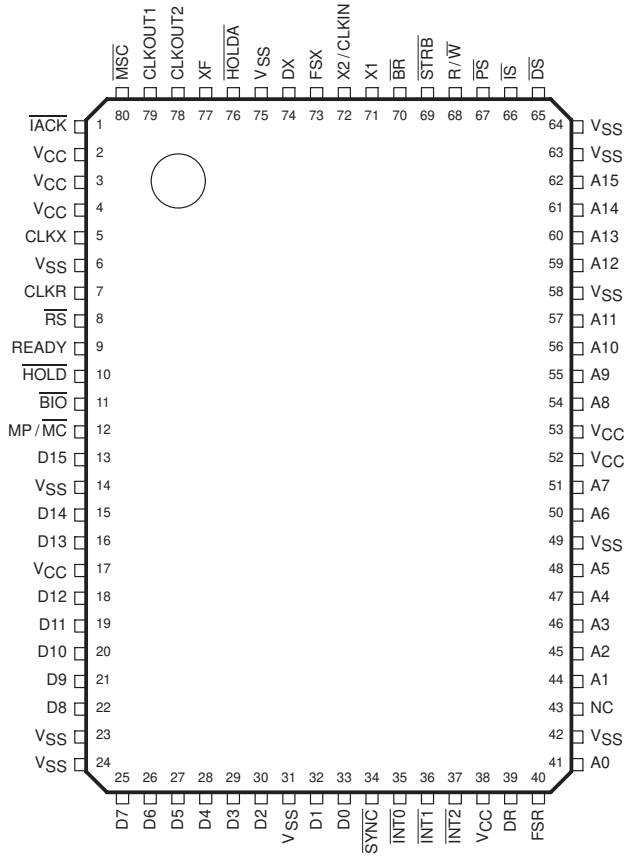
ADVANCE INFORMATION

- **Instruction Cycle Time of 100 ns (40 MHz)**
- **4K Words of On-Chip Secure Program EPROM**
- **544 Words of On-Chip Data RAM**
- **128K Words of Data/Program Space**
- **16 Parallel I/O Ports**
- **32-Bit ALU/Accumulator**
- **16 × 16-Bit Multiplier With a 32-Bit Product**
- **Block Moves for Data/Program Management**
- **Repeat Instructions for Efficient Use of Program Space**
- **Serial Port for Direct Codec Interface**
- **Synchronization Input for Synchronous Multiprocessor Configurations**
- **Wait States for Communication to Slow Off-Chip Memories/Peripherals**
- **On-Chip Timer for Control Operations**
- **Single 5-V Supply**
- **Packaging:**
  - **68-Lead Plastic J-Leaded Chip Carrier (FN Suffix)**
  - **80-Lead Plastic Quad Flatpack (PH Suffix)**
- **68-to-28-Lead Conversion Adapter Socket for EPROM Programming**

FN PACKAGE
(TOP VIEW)



PH PACKAGE
(TOP VIEW)



description

The TMS320P25 digital signal processor is a member of the TMS320 family of VLSI digital signal processors and peripherals. The TMS320 family supports a wide range of digital signal processing applications, such as telecommunications, modems, image processing, speech processing, spectrum analysis, audio processing, digital filtering, high-speed control, graphics, and other computation intensive applications.

With a 100-ns instruction cycle time and an innovative memory configuration, the ’320P25 performs operations necessary for many real-time digital signal processing algorithms. Since most instructions require only one cycle, the TMS320P25 is capable of executing ten million instructions per second. On-chip programmable data/program RAM of 544 words of 16 bits, on-chip program EPROM of 4K words (one-time programmable memory), direct addressing of up

description (continued)

to 64K words of external program and 64K words of data memory space, and multiprocessor interface features for sharing global memory minimize unnecessary data transfers to take full advantage of the capabilities of the processor.

Terminal Functions

TERMINAL NAME	TYPE†	DESCRIPTION
V_{CC}	I	5-V supply
V_{SS}	I	Ground
X1	O	Output from internal oscillator for crystal
X2/CLKIN	I	Input to internal oscillator from crystal or external clock
CLKOUT1	O	Master clock-output (crystal or CLKIN frequency/4)
CLKOUT2	O	A second clock-output signal
D15-D0	I/O/Z	16-bit data bus D15 (MSB) through D0 (LSB). Multiplexed between program, data, and I/O spaces.
A15-A0	O/Z	16-bit address bus A15 (MSB) through A0 (LSB)
$\overline{PS}$, $\overline{DS}$, $\overline{IS}$	O/Z	Program, data, and I/O space select signals
R/$\overline{W}$	O/Z	Read/write signal
$\overline{STRB}$	O/Z	Strobe signal
$\overline{RS}$	I	Reset input
$\overline{INT2}$-$\overline{INT0}$	I	External user interrupt inputs
MP/$\overline{MC}$	I	Microprocessor/microcomputer mode select
$\overline{MSC}$	O	Microstate complete signal
$\overline{IACK}$	O	Interrupt acknowledge signal
READY	OI	Data ready input. Asserted by external logic when using slower devices to indicate that the current bus transaction is complete.
$\overline{BR}$	O	Bus request. Asserted when the TMS320P25 requires access to an external global data memory space.
XF	O	External flag output (latched software-programmable signal)
$\overline{HOLD}$	I	Hold input. When asserted, TMS320P25 goes into an idle mode and places the data, address, and control lines in the high-impedance state.
$\overline{HOLDA}$	O	Hold acknowledge
$\overline{SYNC}$	I	Synchronization input
$\overline{BIO}$	I	Branch control input. Polled by BIOZ instruction.
DR	I	Serial data receive input
CLKR	I	Clock for receive input for serial port
FSR	I	Frame synchronization pulse for receive input
DX	O/Z	Serial-data-transmit output
CLKX	I	Clock for transmit output for serial port
FSX	I/O/Z	Frame synchronization pulse for transmit. Configuration as either an input or an output.

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

ADVANCE INFORMATION

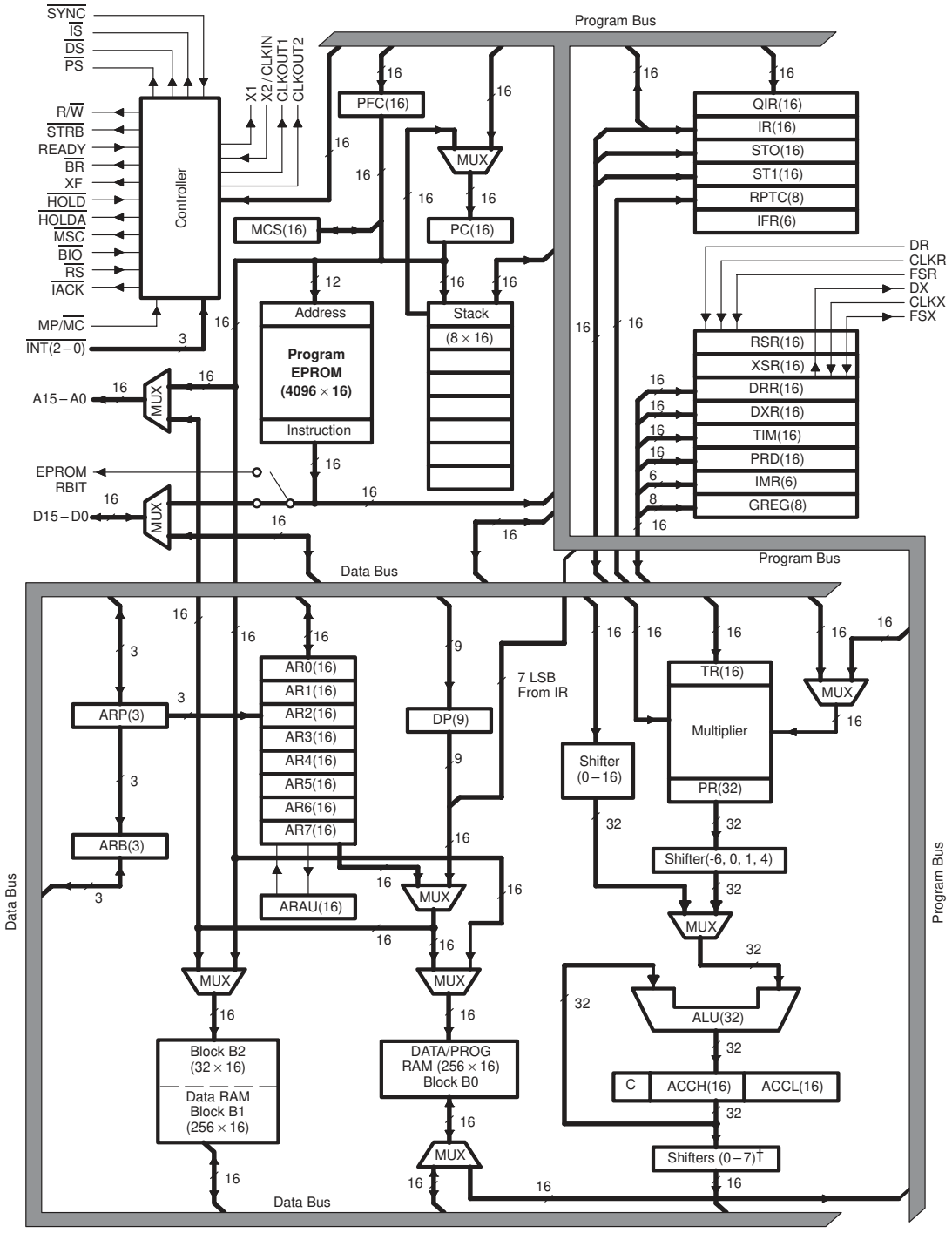
architecture

The TMS320 family utilizes a modified Harvard architecture for speed and flexibility. In a strict Harvard architecture, program and data memory reside in two separate spaces permitting a full overlap of instruction fetch and execution. The TMS320 family's modification of the Harvard architecture allows transfers between program and data spaces, thereby increasing the flexibility of the device. This modification permits coefficients stored in program memory to be read into the RAM, eliminating the need for a separate coefficient ROM. It also makes available immediate instructions and subroutines based on computed values.

Increased throughput on the TMS320P25 devices for many DSP applications is accomplished by means of single-cycle multiply/accumulate instructions with a data move option, up to eight auxiliary registers with a dedicated arithmetic unit, and faster I/O necessary for data-intensive signal processing.

The architectural design of the TMS320P25 emphasizes overall speed, communication, and flexibility in processor configuration. Control signals and instructions provide floating-point support, block-memory transfers, communication to slower off-chip devices, and multiprocessing implementations.

functional block diagram



Legend:

ACCH	=	Accumulator high	IFR	=	Interrupt-flag register	PC	=	Program counter
ACCL	=	Accumulator low	IMR	=	Interrupt-mask register	PFC	=	Prefetch counter
ALU	=	Arithmetic logic unit	IR	=	Instruction register	RPTC	=	Repeat-instruction counter
ARAU	=	Auxiliary-register-arithmetic unit	MCS	=	Microcall stack	GREG	=	Global-memory-allocation register
ARB	=	Auxiliary-register-pointer buffer	QIR	=	Queue-instruction register	RSR	=	Serial-port receive-shift register
ARP	=	Auxiliary register pointer	PR	=	Product register	XSR	=	Serial-port transmit-shift register
DP	=	Data memory page pointer	PRD	=	Period register for timer	AR0–AR7	=	Auxiliary registers
DRR	=	Serial-port data-receive register	TIM	=	Timer	ST0, ST1	=	Status registers
DXR	=	Serial-port data-transmit register	TR	=	Temporary register	C	=	Carry bit

ADVANCE INFORMATION

32-bit ALU/accumulator

The 32-bit arithmetic logic unit (ALU) and accumulator perform a wide range of arithmetic and logical instructions, the majority of which execute in a single clock cycle. The ALU executes a variety of branch instructions dependent on the status of the ALU or a single bit in a word. These instructions provide the following capabilities:

- Branch to an address specified by the accumulator
- Normalize fixed-point numbers contained in the accumulator
- Test a specified bit of a word in data memory

One input to the ALU is always provided from the accumulator, and the other input can be provided from the product register (PR) of the multiplier or the input scaling shifter that has fetched data from the RAM on the data bus. After the ALU has performed the arithmetic or logical operations, the result is stored in the accumulator.

The 32-bit accumulator is split into two 16-bit segments for storage in data memory. Additional shifters at the output of the accumulator perform shifts while the data is being transferred to the data bus for storage. The contents of the accumulator remain unchanged.

scaling shifter

The TMS320P25 scaling shifter has a 16-bit input connected to the data bus and a 32-bit output connected to the ALU. The scaling shifter produces a left shift of 0 to 16 bits on the input data, as programmed in the instruction. The LSBs of the output are filled with zeros, and the MSBs can be either filled with zeros or sign extended, depending upon the status programmed into the SXM (sign-extension mode) bit of status register ST1.

16 × 16-bit parallel multiplier

The 16 × 16-bit hardware multiplier is capable of computing a signed or unsigned 32-bit product in a single machine cycle. The multiplier has the following two associated registers:

- A 16-bit temporary register (TR) that holds one of the operands for the multiplier
- A 32-bit product register (PR) that holds the product

Incorporated into the instruction set are single-cycle multiply/accumulate instructions that allow both operands to be processed simultaneously. The data for these operations can reside anywhere in internal or external memory and can be transferred to the multiplier each cycle via the program and data buses.

Four product shift modes are available at the product register (PR) output that are useful when performing multiply/accumulate operations, fractional arithmetic, or justifying fractional products.

timer

The TMS320P25 provides a memory-mapped 16-bit timer for control operations. The on-chip timer (TIM) register is a down counter that is continuously clocked by CLKOUT1 on the TMS320P25. A timer interrupt (TINT) is generated every time the timer decrements to zero. The timer is reloaded with the value contained in the period (PRD) register within the next cycle after it reaches zero so that interrupts can be programmed to occur at regular intervals of PRD + 1 cycles of CLKOUT1 on the TMS320P25.

memory control

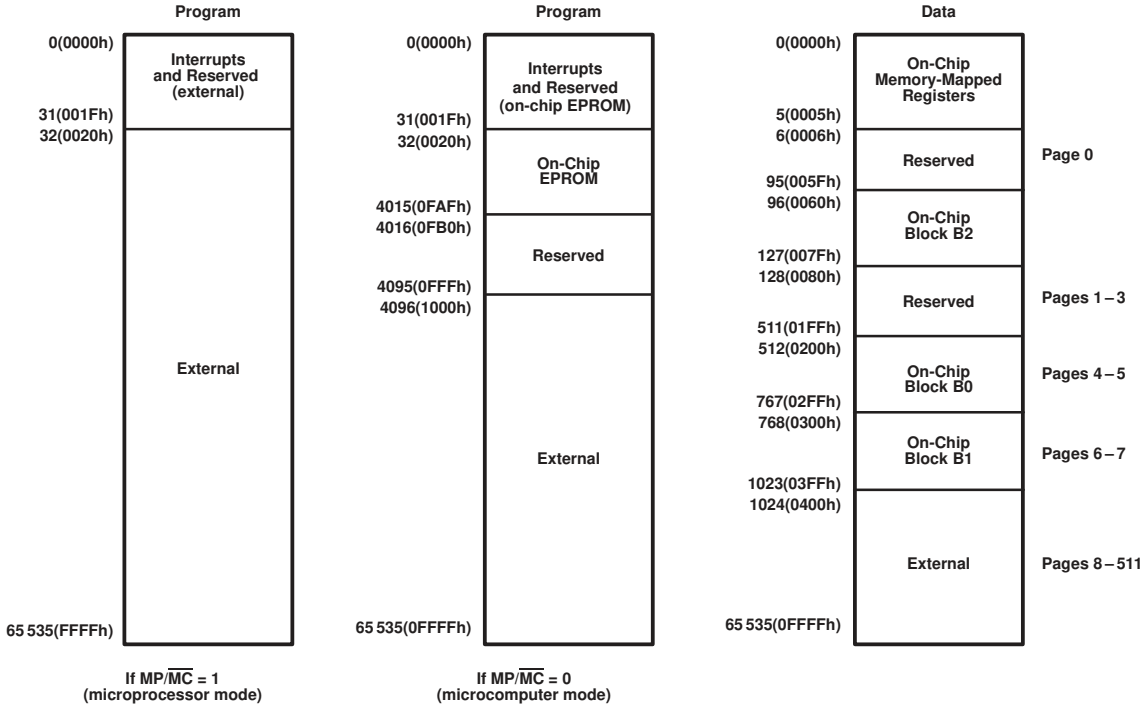
The TMS320P25 provides a total of 544 16-bit words of on-chip data RAM, divided into three separate blocks (B0, B1, and B2). Of the 544 words, 288 words (blocks B1 and B2) are always data memory and 256 words (block B0) are programmable as either data or program memory. A data memory size of 544 words allows the TMS320P25 to handle a data array of 512 words (256 words if on-chip RAM is used for program memory), while still leaving 32 locations for intermediate storage. When using block B0 as program memory, instructions can be downloaded from external program memory into on-chip RAM and then executed.

memory control (continued)

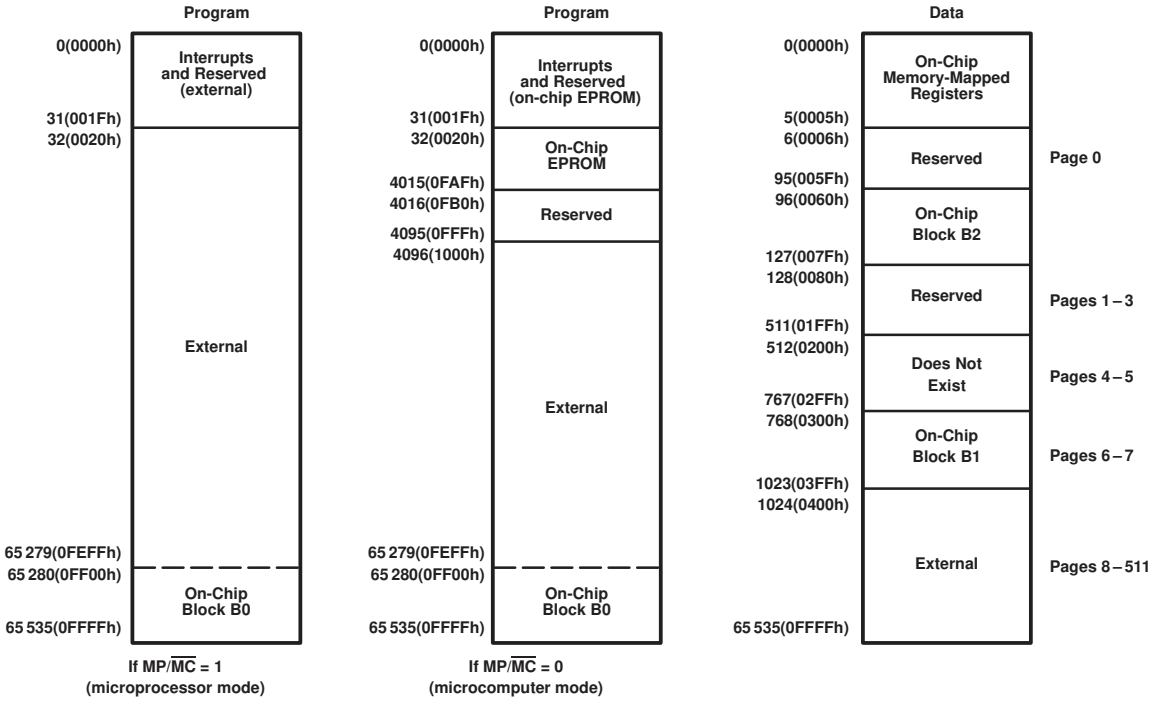
When using on-chip program RAM, EPROM, or high-speed external program memory, the TMS320P25 runs at full speed without wait states. However, the READY line can be used to interface the TMS320P25 to slower, less-expensive external memory. Downloading programs from slow off-chip memory to on-chip program RAM speeds processing while cutting system costs.

The TMS320P25 provides three separate address spaces for program memory, data memory, and I/O. The on-chip memory is mapped into either the 64K-word data memory or program memory space, depending upon the memory configuration (see Figure 1). The CNFD (configure block B0 as data memory) and CNFP (configure block B0 as program memory) instructions allow dynamic configuration of the memory maps through software. Regardless of the configuration, the user can still execute from external program memory.

The TMS320P25 has six registers that are mapped into the data memory space: a serial-port data-receive register, serial-port data-transmit register, timer register, period register, interrupt-mask register, and global-memory-allocation register.



(a) MEMORY MAPS AFTER A CNFD INSTRUCTION



(b) MEMORY MAPS AFTER A CNFP INSTRUCTION

Figure 1. Memory Maps

ADVANCE INFORMATION

external interface

The TMS320P25 supports a wide range of system-interfacing requirements. Program, data, and I/O address spaces provide interface to memory and I/O, thus maximizing system throughput. 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. Interface to memory and I/O devices of varying speeds is accomplished by using the READY line. When transactions are made with slower devices, the TMS320P25 processor waits until the other device completes its function and signals the processor via the READY line. Then, the TMS320P25 continues execution.

A full-duplex serial port provides communication with serial devices such as codecs, serial A/D converters, and other serial systems. The interface signals are compatible with codecs and many other serial devices with a minimum of external hardware. The serial port can also be used for communication between processors in multiprocessing applications.

The serial port has two memory-mapped registers: the data-transmit register (DXR) and the data-receive register (DRR). Both registers operate in either the byte mode or 16-bit word mode and can be accessed in the same manner as any other data memory location. Each register has an external clock, a framing synchronization pulse, and associated shift registers. One method of multiprocessing can be implemented by programming one device to transmit while the others are in the receive mode. The serial port on the TMS320P25 is double buffered and fully static.

interrupts and subroutines

The TMS320P25 has three external maskable user interrupts $\overline{\text{INT2}}$–$\overline{\text{INT0}}$, available for external devices that interrupt the processor. Internal interrupts are generated by the serial port (RINT and XINT), by the timer (TINT), and by the software interrupt (TRAP) instruction. Interrupts are prioritized with reset ($\overline{\text{RS}}$) having the highest priority and the serial-port transmit interrupt (XINT) having the lowest priority. All interrupt 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 is completed. This mechanism applies to instructions that are repeated and to instructions that become multicycle because of the READY signal.

multiprocessing

The flexibility of the TMS320P25 allows configurations to satisfy a wide range of system requirements and can be used as follows:

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

For multiprocessing applications, the TMS320P25 has the capability of allocating global data-memory space and communicating with that space via the bus request ($\overline{\text{BR}}$) and READY control signals. Global data-memory is data memory shared by more than one processor. Global data-memory access must be arbitrated. The 8-bit memory-mapped global memory-allocation register (GREG) specifies part of the TMS320P25 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, $\overline{\text{BR}}$ is asserted to request control of the bus. The length of the memory cycle is controlled by the READY line.

The TMS320P25 supports direct memory access (DMA) to its external program/data memory using the $\overline{\text{HOLD}}$ and $\overline{\text{HOLDA}}$ signals. Another processor can take complete control of the TMS320P25's external memory by asserting $\overline{\text{HOLD}}$ low. This causes the TMS320P25 to place its address data and control lines in the high-impedance state and assert $\overline{\text{HOLDA}}$. On the TMS320P25, program execution from on-chip EPROM can proceed concurrently when the device is in the hold mode.

instruction set

The TMS320P25 microprocessor implements a comprehensive instruction set that supports both numeric-intensive signal processing operations as well as general-purpose applications, such as multiprocessing and high-speed control.

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 can vary 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 TMS320P25 instruction set provides three memory addressing modes: direct, indirect, and immediate addressing. Both direct and indirect addressing can be used to access data memory. In direct memory 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 to form the full 16-bit address. Memory is paged in the direct addressing mode with a total of 512 pages, each page containing 128 words. Indirect addressing accesses data memory through the auxiliary registers. In immediate addressing, the data is based on a portion of the instruction word(s).

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.

There are seven types of indirect addressing: auto-increment or auto-decrement, post-indexing by either adding or subtracting the contents of AR0, single indirect addressing with no increment or decrement, and bit-reversal addressing (used in 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.

repeat feature

A repeat feature, used with instructions such as multiply/accumulates, block moves, I/O transfers, and table read/writes, allows a single instruction to be performed up to 256 times. The repeat counter (RPTC) is loaded with either a data memory value (RPT instruction) or an immediate value (RPTK instruction). The value of this operand is one less than the number of times that the next instruction is executed. Those instructions that are normally multicycle are pipelined when using the repeat feature, and effectively become single-cycle instructions.

instruction set summary

Table 1 lists the symbols and abbreviations used in Table 2. Table 2 consists primarily of single-cycle, single-word instructions. Infrequently used branch, I/O, and CALL instructions are multicycle. The instruction set summary is arranged according to function and alphabetized within each functional grouping. The symbol † indicates those instructions that are not included in the TMS320C1x instruction set.

Table 1. Instruction Symbols

SYMBOL	DEFINITION
B	4-bit field specifying a bit code
CM	2-bit field specifying compare mode
D	Data memory address field
FO	Format status bit
I	Addressing mode bit
K	Immediate operand field
PA	Port address (PA0 through PA15 are predefined assembler symbols equal to 0 through 15, respectively.)
PM	2-bit field specifying P register output shift code
AR	3-bit operand field specifying auxiliary register
S	4-bit left-shift code
X	3-bit accumulator left-shift field

Table 2. TMS320P25 Instruction Set Summary

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS																	
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE														
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABS	Absolute value of accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	1	0	1	1
ADD	Add to accumulator with shift	1	0	0	0	0	←	S	→		I	←			D			→
ADDC	Add to accumulator with carry	1	0	1	0	0	0	0	1	1	I	←			D			→
ADDH	Add to high accumulator	1	0	1	0	0	1	0	0	0	I	←			D			→
ADDK	Add to accumulator short immediate	1	1	1	0	0	1	1	0	0	←				K			→
ADDS	Add to low accumulator with sign extension suppressed	1	0	1	0	0	1	0	0	1	I	←			D			→
ADDT	Add to accumulator with shift specified by T register	1	0	1	0	0	1	0	1	0	I	←			D			→
ADLK†	Add to accumulator long immediate with shift	2	1	1	0	1	←	S	→		0	0	0	0	0	0	1	0
AND	AND with accumulator	1	0	1	0	0	1	1	1	0	I	←			D			→
ANDK†	AND immediate with accumulator with shift	2	1	1	0	1	←	S	→		0	0	0	0	0	1	0	0
CMPL†	Complement accumulator	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	1	1
LAC	Load accumulator with shift	1	0	0	1	0	←	S	→		I	←			D			→
LACK	Load accumulator immediate short	1	1	1	0	0	1	0	1	0	←				K			→
LACT†	Load accumulator with shift specified by T register	1	0	1	0	0	0	0	1	0	I	←			D			→
LALK†	Load accumulator long immediate with shift	2	1	1	0	1	←	S	→		0	0	0	0	0	0	0	1
NEG†	Negate accumulator	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	1	1
NORM†	Normalize contents of accumulator	1	1	1	0	0	1	1	1	0	1	X	X	X	0	0	1	0
OR	OR with accumulator	1	0	1	0	0	1	1	0	1	I	←			D			→
ORK†	OR immediate with accumulator with shift	2	1	1	0	1	←	S	→		0	0	0	0	0	1	0	1
ROL	Rotate accumulator left	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	0	0
ROR	Rotate accumulator right	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	0	1
SACH	Store high accumulator with shift	1	0	1	1	0	1	←	X	→	I	←			D			→
SACL	Store low-order accumulator with shift	1	0	1	1	0	0	←	X	→	I	←			D			→
SBLK†	Subtract from accumulator long immediate with shift	2	1	1	0	1	←	S	→		0	0	0	0	0	0	1	1
SFL†	Shift accumulator left	1	1	1	0	0	1	1	1	0	0	0	0	1	1	0	0	0
SFR†	Shift accumulator right	1	1	1	0	0	1	1	1	0	0	0	0	1	1	0	0	1
SUB	Subtract from accumulator with shift	1	0	0	0	1	←	S	→		I	←			D			→
SUBB	Subtract from accumulator with borrow	1	0	1	0	0	1	1	1	1	I	←			D			→
SUBC	Conditional subtract	1	0	1	0	0	0	1	1	1	I	←			D			→
SUBH	Subtract from high accumulator	1	0	1	0	0	0	1	0	0	I	←			D			→
SUBK	Subtract from accumulator short immediate	1	1	1	0	0	1	1	0	1	←				K			→
SUBS	Subtract from low accumulator with sign extension suppressed	1	0	1	0	0	0	1	0	1	I	←			D			→

† These instructions are not included in the TMS320C1x instruction set.

ADVANCE INFORMATION

Table 2. TMS320P25 Instruction Set Summary (Continued)

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS																	
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE														
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SUBT†	Subtract from accumulator with shift specified by T register	1	0	1	0	0	0	1	1	0	I	←			D			→
XOR	Exclusive-OR with accumulator	1	0	1	0	0	1	1	0	0	I	←			D			→
XORK†	Exclusive-OR immediate with accumulator with shift	2	1	1	0	1	←	S	→		0	0	0	0	0	1	1	0
ZAC	Zero accumulator	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0
ZALH	Zero low accumulator and load high accumulator	1	0	1	0	0	0	0	0	0	I	←			D			→
ZALR	Zero low accumulator and load high accumulator with rounding	1	0	1	1	1	1	0	1	1	I	←			D			→
ZALS	Zero accumulator and load low accumulator with sign extension suppressed	1	0	1	0	0	0	0	0	1	I	←			D			→

AUXILIARY REGISTERS AND DATA PAGE POINTER INSTRUCTIONS																	
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE														
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADRK	Add to auxiliary register short immediate	1	0	1	1	1	1	1	1	0	←				K			→
CMPR†	Compare auxiliary register with auxiliary register AR0	1	1	1	0	0	1	1	1	0	0	1	0	1	0	0	←	CM →
LAR	Load auxiliary register	1	0	0	1	1	0	←	R	→	I	←			D			→
LARK	Load auxilliary register short immediate	1	1	1	0	0	0	←	R	→	←				K			→
LARP	Load auxilliary register pointer	1	0	1	0	1	0	1	0	1	1	0	0	0	1	←	R	→
LDP	Load data memory page pointer	1	0	1	0	1	0	0	1	0	I	←			D			→
LDPK	Load data memory page pointer immediate	1	1	1	0	0	1	0	0	←				DP				→
LRLK†	Load auxiliary register long immediate	2	1	1	0	1	0	←	R	→	0	0	0	0	0	0	0	0
MAR	Modify auxiliary register	1	0	1	0	1	0	1	0	1	I	←			D			→
SAR	Store auxiliary register	1	0	1	1	1	0	←	R	→	I	←			D			→
SBRK	Subtract from auxiliary register short immediate	1	0	1	1	1	1	1	1	1	←				K			→

† These instructions are not included in the TMS320C1x instruction set.

ADVANCE INFORMATION

Table 2. TMS320P25 Instruction Set Summary (Continued)

T REGISTER, P REGISTER, AND MULTIPLY INSTRUCTIONS																		
MNEMONIC	**DESCRIPTION**	**NO. WORDS**	**INSTRUCTION BIT CODE**															
			15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
APAC	Add P register to accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	0	1	0	1
LPH†	Load high P register	1	0	1	0	1	0	0	1	1	I	◄			D			►
LT	Load T register	1	0	0	1	1	1	1	0	0	I	◄			D			►
LTA	Load T register and accumulate previous product	1	0	0	1	1	1	1	0	1	I	◄			D			►
LTD	Load T register, accumulate previous product, and move data	1	0	0	1	1	1	1	1	1	I	◄			D			►
LTP†	Load T register and store P register in accumulator	1	0	0	1	1	1	1	1	0	I	◄			D			►
LTS†	Load T register and subtract previous product	1	0	1	0	1	1	0	1	1	I	◄			D			►
MAC†	Multiply and accumulate	2	0	1	0	1	1	1	0	1	I	◄			D			►
MACD†	Multiply and accumulate with data move	2	0	1	0	1	1	1	0	0	I	◄			D			►
MPY	Multiply (with T register, store product in P register)	1	0	0	1	1	1	0	0	0	I	◄			D			►
MPYA	Multiply and accumulate previous product	1	0	0	1	1	1	0	1	0	I	◄			D			►
MPYK	Multiply immediate	1	1	0	1	◄						K						►
MPYS	Multiply and subtract previous product	1	0	0	1	1	1	0	1	1	I	◄			D			►
MPYU	Multiply unsigned	1	1	1	0	0	1	1	1	1	I	◄			D			►
PAC	Load accumulator with P register	1	1	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0
SPAC	Subtract P register from accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	0	1	1	0
SPH	Store high P register	1	0	1	1	1	1	1	0	1	I	◄			D			►
SPL	Store low P register	1	0	1	1	1	1	1	0	0	I	◄			D			►
SPM†	Set P register output shift mode	1	1	1	0	0	1	1	1	0	0	0	0	0	1	0	◄	PM ►
SQRA†	Square and accumulate	1	0	0	1	1	1	0	0	1	I	◄			D			►
SQRS†	Square and subtract previous product	1	0	1	0	1	1	0	1	0	I	◄			D			►

† These instructions are not included in the TMS320C1x instruction set.

ADVANCE INFORMATION

Table 2. TMS320P25 Instruction Set Summary (Continued)

BRANCH/CALL INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B	Branch unconditionally	2	1	1	1	1	1	1	1	1	1	←			D			→
BACC†	Branch to address specified by accumulator	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	0	1
BANZ	Branch on auxiliary register not zero	2	1	1	1	1	1	0	1	1	1	←			D			→
BBNZ†	Branch if TC bit ≠ 0	2	1	1	1	1	1	0	0	1	1	←			D			→
BBZ†	Branch if TC bit = 0	2	1	1	1	1	1	0	0	0	1	←			D			→
BC	Branch on carry	2	0	1	0	1	1	1	1	0	1	←			D			→
BGEZ	Branch if accumulator ≥ 0	2	1	1	1	1	0	1	0	0	1	←			D			→
BGZ	Branch if accumulator > 0	2	1	1	1	1	0	0	0	1	1	←			D			→
BIOZ	Branch on I/O status = 0	2	1	1	1	1	1	0	1	0	1	←			D			→
BLEZ	Branch if accumulator ≤ 0	2	1	1	1	1	0	0	1	0	1	←			D			→
BLZ	Branch if accumulator < 0	2	1	1	1	1	0	0	1	1	1	←			D			→
BNC	Branch on no carry	2	0	1	0	1	1	1	1	1	1	←			D			→
BNV†	Branch if no overflow	2	1	1	1	1	0	1	1	1	1	←			D			→
BNZ	Branch if accumulator ≠ 0	2	1	1	1	1	0	1	0	1	1	←			D			→
BV	Branch on overflow	2	1	1	1	1	0	0	0	0	1	←			D			→
BZ	Branch if accumulator = 0	2	1	1	1	1	0	1	1	0	1	←			D			→
CALA	Call subroutine indirect	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	0	0
CALL	Call subroutine	2	1	1	1	1	1	1	1	0	1	←			D			→
RET	Return from subroutine	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	1	0
I/O AND DATA MEMORY OPERATIONS																		
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLKD†	Block move from data memory to data memory	2	1	1	1	0	1	1	0	1	I	←			D			→
BLKP†	Block move from program memory to data memory	2	1	1	1	1	1	1	0	0	I	←			D			→
DMOV	Data move in data memory	1	0	1	0	1	0	1	1	0	I	←			D			→
FORT†	Format serial port registers	1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	1	FO
IN	Input data from port	1	1	0	0	0	←	PA	→	I		←			D			→
OUT	Output data to port	1	1	1	1	0	←	PA	→	I		←			D			→
RFSM	Reset serial port frame synchronization mode	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	1	0
RTXM†	Reset serial port transmit mode	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	0	0
RXF†	Reset external flag	1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	0	0
SFSM	Set serial port frame synchronization mode	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	1	1
STXM	Set serial port transmit mode	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	0	1
SXF†	Set external flag	1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1
TBLR	Table read	1	0	1	0	1	1	0	0	0	I	←			D			→
TBLW	Table write	1	0	1	0	1	1	0	0	1	I	←			D			→

† These instructions are not included in the TMS320C1x instruction set.

ADVANCE INFORMATION

Table 2. TMS320P25 Instruction Set Summary (Continued)

CONTROL INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BIT†	Test bit	1	1	0	0	1	←	B	→		I	←			D			→
BITT†	Test bit specified by T register	1	0	1	0	1	0	1	1	1	I	←			D			→
CNFD†	Configure block as data memory	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	0	0
CNFP†	Configure block as program memory	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	0	1
DINT	Disable interrupt	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0	1
EINT	Enable interrupt	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0
IDLE†	Idle until interrupt	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1	1
LST	Load status register STO	1	0	1	0	1	0	0	0	0	I	←			D			→
LST1†	Load status register ST1	1	0	1	0	1	0	0	0	1	I	←			D			→
NOP	No operation	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0
POP	Pop top of stack to low accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	0	1
POPD†	Pop top of stack to data memory	1	0	1	1	1	1	0	1	0	I	←			D			→
PSHD†	Push data memory value onto stack	1	0	1	0	1	0	1	0	0	I	←			D			→
PUSH	Push low accumulator onto stack	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	0	0
RC	Reset carry bit	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	0	0
RHM	Reset hold mode	1	1	1	0	0	1	1	1	0	0	0	1	1	1	0	0	0
ROVM	Reset overflow mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	1	0
RPT†	Repeat instruction as specified by data memory value	1	0	1	0	0	1	0	1	1	I	←			D			→
RPTK†	Repeat instruction as specified by immediate value	1	1	1	0	0	1	0	1	1	←				K			→
RSXM†	Reset sign-extension mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	1	0
RTC	Reset test/control flag	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	1	0
SC	Set carry bit	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	0	1
SHM	Set hold mode	1	1	1	0	0	1	1	1	0	0	0	1	1	1	0	0	1
SOVM	Set overflow mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	1	1
SST	Store status register ST0	1	0	1	1	1	1	0	0	0	I	←			D			→
SST1†	Store status register ST1	1	0	1	1	1	1	0	0	1	I	←			D			→
SSXM†	Set sign-extension mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	1	1
STC	Set test/control flag	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	1	1
TRAP†	Software interrupt	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1	0

† These instructions are not included in the TMS320C1x instruction set.

ADVANCE INFORMATION

ADVANCE INFORMATION

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC} (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
Continuous power dissipation . . . . . 2 W
Operating free-air temperature range, T_A . . . . . 0°C to 70°C
Storage temperature range . . . . . – 55°C to 150°C

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

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

recommended operating conditions

			MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage		4.75	5	5.25	V
V_{SS}	Supply voltage			0		V
V_{IH}	High-level input voltage	All inputs except CLKIN/CLKX/CLKR/$\overline{INT0}$–$\overline{INT2}$	2.35		V_{CC} + 0.3	V
		CLKIN/CLKX/CLKR	3.65		V_{CC} + 0.3	
		$\overline{INT0}$–$\overline{INT2}$	2.5		V_{CC} + 0.3	
V_{IL}	Low-level input voltage	All inputs except MP/$\overline{MC}$	– 0.3		0.8	V
		MP/$\overline{MC}$	– 0.3		0.8	V
I_{OH}	High-level output current				300	μA
I_{OL}	Low-level output current				2	mA
T_A	Operating free-air temperature		0		70	°C

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

PARAMETER			TEST CONDITIONS‡	MIN	TYP§	MAX	UNIT
V_{OH}	High-level output voltage		V_{CC} = MIN, I_{OH} = MAX	2.4	3		V
V_{OL}	Low-level output voltage		V_{CC} = MIN, I_{OL} = MAX		0.3	0.6	V
I_Z	Three-state current		V_{CC} = MAX	–20		20	μA
I_I	Input current		V_I = V_{SS} to V_{CC}	–10		10	μA
I_{CC}	Supply current	Normal	T_A = 0°C, V_{CC} = MAX, f_x = MAX		110	185	mA
		IDLE, $\overline{HOLD}$			50	100	mA
C_i	Input capacitance				15		pF
C_o	Output capacitance				15		pF

‡ For test conditions shown as MIN/ MAX, use the appropriate value listed in the recommended operating conditions or the internal clock option table.

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

This device contains circuits to protect its inputs and outputs against damage due to high static voltages or electrostatic fields. These circuits have been qualified to protect this device against electrostatic discharges (ESD) of up to 2 kV according to MIL-STD-883C, Method 3015; however, it is advised that precautions should be taken to avoid application of any voltage higher than maximum-rated voltages to these high-impedance circuits. During storage or handling, the device leads should be shorted together or the device should be placed in conductive foam. In a circuit, unused inputs should always be connected to an appropriated logic voltage level, preferably either V_{CC} or ground. Specific guidelines for handling devices of this type are contained in the publication *Guidelines for Handling Electrostatic-Discharge-Sensitive (ESDS) Devices and Assemblies* available from Texas Instruments.

CLOCK CHARACTERISTICS AND TIMING

The TMS320P25 can use either its internal oscillator or an external frequency source for a clock.

internal clock option

The internal oscillator is enabled by connecting a crystal across X1 and X2/CLKIN (see Figure 2). The frequency of CLKOUT1 is one-fourth the crystal fundamental frequency. The crystal should be fundamental mode and parallel resonant with an effective series resistance of 30 Ω, a power dissipation of 1 mW, and be specified at a load capacitance of 20 pF.

recommended operating conditions for internal clock option

		TEST CONDITIONS	MIN	NOM†	MAX	UNIT
f_X	Input clock frequency	T_A = 0°C to 70°C	6.7		40.96	MHz
f_{XS}	Serial port frequency	T_A = 0°C to 70°C	0†		5.12	MHz
C1, C2	Load capacitance	T_A = 0°C to 70°C		10		pF

† The serial port is tested at a minimum frequency of 1.25 MHz. However, the serial port is fully static and properly functions down to a clock rate of f_{SX} = 0 Hz.

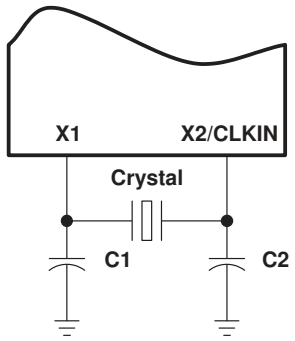


Figure 2. Internal Clock Option

external clock option

An external frequency source can be used by injecting the frequency directly into X2/CLKIN with X1 left unconnected. The external frequency injected must conform to the specifications listed in the following table.

switching characteristics for external clock option (see Note 2)

PARAMETER		MIN	TYP	MAX	UNIT
$t_{c(C)}$	Cycle time, CLKOUT1/CLKOUT2	97.7		597	ns
$t_{d(CIH-C)}$	Delay time, CLKIN high to CLKOUT1/CLKOUT2/$\overline{STRB}$ high/low	5		30	ns
$t_{f(C)}$	Fall time, CLKOUT1/CLKOUT2/$\overline{STRB}$			5	ns
$t_{r(C)}$	Rise time, CLKOUT1/CLKOUT2/$\overline{STRB}$			5	ns
$t_{w(CL)}$	Pulse duration, CLKOUT1/CLKOUT2 low	2Q – 8	2Q	2Q + 8	ns
$t_{w(CH)}$	Pulse duration, CLKOUT1/CLKOUT2 high	2Q – 8	2Q	2Q + 8	ns
$t_{d(C1-C2)}$	Delay time, CLKOUT1 high to CLKOUT2 low, CLKOUT2 high to CLKOUT1 high, etc.	Q – 8	Q	Q + 2	ns

NOTE 2: Q = 1/4$t_{c(C)}$

ADVANCE INFORMATION

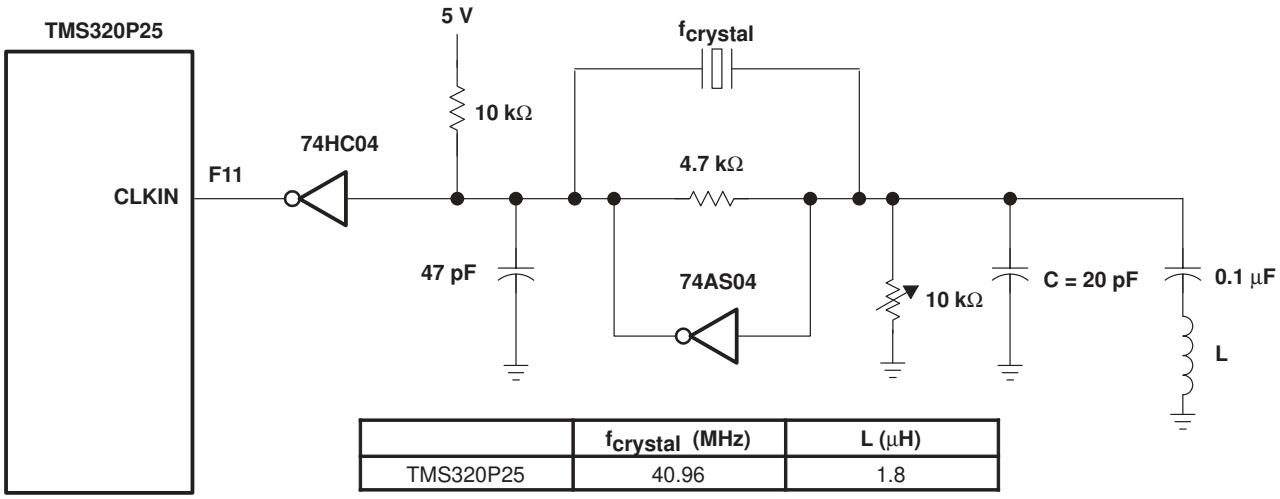
timing requirements for external clock option (see Note 2)

		MIN	MAX	UNIT
$t_{c(CI)}$	Cycle time, CLKIN	24.4	150	ns
$t_{f(CI)}$	Fall time, CLKIN		5†	ns
$t_{r(CI)}$	Rise time, CLKIN		5†	ns
$t_{w(CIL)}$	Pulse duration, CLKIN low, $t_{c(CI)}$ = 50 ns (see Note 3)	20		ns
$t_{w(CIH)}$	Pulse duration, CLKIN high, $t_{c(CI)}$ = 50 ns (see Note 3)	20		ns
$t_{su(S)}$	Setup time, $\overline{\text{SYNC}}$ before CLKIN low	5	Q – 8	ns
$t_{h(S)}$	Hold time, $\overline{\text{SYNC}}$ from CLKIN low	8		ns

† Value is derived from characterization data and not tested.

NOTES: 2. $Q = 1/4t_{c(C)}$

3. CLKIN duty cycle $[t_{r(CI)} + t_{w(CIH)}] / t_{c(CI)}$ must be within 40-60%.



	fcrystal (MHz)	L (µH)
TMS320P25	40.96	1.8

Figure 3. External Clock Option

ADVANCE INFORMATION

MEMORY AND PERIPHERAL INTERFACE TIMING

switching characteristics over recommended operating conditions (see Note 2)

PARAMETER			MIN	TYP	MAX	UNIT
$t_{d(C1-S)}$	Delay time, $\overline{STRB}$ from CLKOUT1 (if $\overline{STRB}$ is present)		Q – 6	Q	Q + 6	ns
$t_{d(C2-S)}$	Delay time, CLKOUT2 to $\overline{STRB}$ (if $\overline{STRB}$ is present)		– 6		6	ns
$t_{d(MSC)}$	Delay time, $\overline{MSC}$ valid from CLKOUT1		–12	0	12	ns
$t_{dis(D)}$	Disable time, data bus in the high-impedance state after $\overline{STRB}$ high (write cycle)			Q	Q + 15†	ns
$t_{en(D)}$	Enable time, data bus starts being driven after $\overline{STRB}$ low (write cycle)		0†			ns
$t_{h(A)}$	Hold time, address after $\overline{STRB}$ high (see Note 4)		Q – 8			ns
$t_{h(D)W}$	Hold time, data write from $\overline{STRB}$ high		Q – 10	Q		ns
$t_{w(SL)}$	Pulse duration, $\overline{STRB}$ low (no wait states, see Note 5)		2Q – 5		2Q + 5†	ns
$t_{w(SH)}$	Pulse duration, $\overline{STRB}$ high (between consecutive cycles, see Note 5)		2Q – 5		2Q + 5	ns
$t_{su(A)}$	Setup time, address before $\overline{STRB}$ low (see Note 4)		Q – 12			ns
$t_{su(D)W}$	Setup time, data write before $\overline{STRB}$ high (no wait states)	TMS320P25FN	2Q – 20			ns
		TMS320P25PH	2Q – 23			

† Value is derived from characterization data and not tested.

NOTES: 2. $Q = 1/4t_{c(C)}$
4. A15-A0, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, R/$\overline{W}$, and $\overline{BR}$ timings are all included in timings referenced as address.
5. Delays between CLKOUT1/CLKOUT2 edges and $\overline{STRB}$ edges track each other, resulting in $t_{w(SL)}$ and $t_{w(SH)}$ being 2Q with no wait states.

timing requirements over recommended operating conditions (see Note 2)

			MIN	MAX	UNIT
$t_{a(A)}$	Access time, read data from address time (read cycle, see Notes 4 and 6)			3Q – 35	ns
$t_{d(SL-R)}$	Delay time, READY valid after $\overline{STRB}$ low (no wait states)	TMS320P25FN		Q – 20	ns
		TMS320P25PH		Q – 22	
$t_{d(C2H-R)}$	Delay time, READY valid after CLKOUT2 high			Q – 20	ns
$t_{d(M-R)}$	Delay time, READY valid after $\overline{MSC}$ valid			2Q – 25	ns
$t_{h(D)R}$	Hold time, data read from $\overline{STRB}$ high		0		ns
$t_{h(SL-R)}$	Hold time, READY after $\overline{STRB}$ low (no wait states)		Q + 3		ns
$t_{h(C2H-R)}$	Hold time, READY after CLKOUT2 high		Q + 3		ns
$t_{h(M-R)}$	Hold time, READY after $\overline{MSC}$ valid		0		ns
$t_{su(D)R}$	Setup time, data read before $\overline{STRB}$ high		23		ns

† Value is derived from characterization data and not tested.

NOTES: 2. $Q = 1/4t_{c(C)}$
4. A15-A0, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, R/$\overline{W}$, and $\overline{BR}$ timings are all included in timings referenced as address.
6. Read data access time is defined as $t_{a(A)} = t_{su(A)} + t_{w(SL)} - t_{su(D)R}$.

ADVANCE INFORMATION

$\overline{RS}$, $\overline{INT}$, $\overline{BIO}$, AND XF TIMING

switching characteristics over recommended operating conditions (see Notes 2 and 7)

PARAMETER		MIN	TYP	MAX	UNIT
$t_{d(RS)}$	Delay time, CLKOUT1 low to reset state entered			22†	ns
$t_{d(IACK)}$	Delay time, CLKOUT1 to $\overline{IACK}$ valid	– 6	0	12	ns
$t_{d(XF)}$	Delay time, XF valid before falling edge of $\overline{STRB}$	Q – 15			ns

NOTES: 2. $Q = 1/4t_{c(C)}$
7. $\overline{RS}$, $\overline{INT}$, and $\overline{BIO}$ are asynchronous inputs and can occur at any time during a clock cycle. However, if the specified setup time is met, the exact sequence shown in the timing diagrams occurs.

timing requirements over recommended operating conditions (see Note 7)

		MIN	MAX	UNIT
$t_{su(IN)}$	Setup time, $\overline{INT}$/$\overline{BIO}$/$\overline{RS}$ before CLKOUT1 high	32		ns
$t_{h(IN)}$	Hold time, $\overline{INT}$/$\overline{BIO}$/$\overline{RS}$ after CLKOUT1 high	0		ns
$t_{f(IN)}$	Fall time, $\overline{INT}$/$\overline{BIO}$		8†	ns
$t_{w(IN)}$	Pulse duration, $\overline{INT}$/$\overline{BIO}$ low	$t_{c(C)}$		ns
$t_{w(RS)}$	Pulse duration, $\overline{RS}$ low	$3t_{c(C)}$		ns

† Value is derived from characterization data and not tested.
NOTE 7: $\overline{RS}$, $\overline{INT}$, and $\overline{BIO}$ are asynchronous inputs and can occur at any time during a clock cycle. However, if the specified setup time is met, the exact sequence shown in the timing diagrams occurs.

$\overline{HOLD}$ TIMING

switching characteristics over recommended operating conditions

PARAMETER		MIN	TYP	MAX	UNIT
$t_{d(C1L-AL)}$	Delay time, $\overline{HOLDA}$ low after CLKOUT1 low	0		10	ns
$t_{d(HH-AH)}$	Delay time, $\overline{HOLD}$ high to $\overline{HOLDA}$ high			25	ns
$t_{dis(AL-A)}$	Disable time, $\overline{HOLDA}$ low to address 3-state		0†		ns
$t_{dis(C1L-A)}$	Disable time, address in the high-impedance state after CLKOUT1 low ($\overline{HOLD}$ mode, see Note 8)			20†	ns
$t_{en(A-C1L)}$	Enable time, address driven before CLKOUT1 low ($\overline{HOLD}$ mode, see Note 8)			8†	ns

† Value is derived from characterization data and not tested.
NOTE 8: A15-A0, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, $\overline{STRB}$, and R/$\overline{W}$ timings are all included in timings referenced as address.

timing requirements over recommended operating conditions (see Note 2)

		MIN	MAX	UNIT
$t_{d(C2H-H)}$	Delay time, $\overline{HOLD}$ valid after CLKOUT2 high		Q – 24	ns

NOTE 2: $Q = 1/4t_{c(C)}$

ADVANCE INFORMATION

SERIAL PORT TIMING

switching characteristics over recommended operating conditions

PARAMETER		MIN	MAX	UNIT
$t_{d(CH-DX)}$	Delay time, DX valid after CLKX rising edge (see Note 9)		75	ns
$t_{d(FL-DX)}$	Delay time, DX valid after FSX falling edge (TXM = 0, see Note 9)		40	ns
$t_{d(CH-FS)}$	Delay time, FSX valid after CLKX rising edge (TXM = 1)		40	ns

NOTE 9: The last occurrence of FSX falling and CLKX rising

timing requirements over recommended operating conditions

		MIN	MAX	UNIT
$t_{c(SCK)}$	Cycle time, serial port clock (CLKX/CLKR)	200		ns
$t_{f(SCK)}$	Fall time, serial port clock (CLKX/CLKR)		25†	ns
$t_{r(SCK)}$	Rise time, serial port clock (CLKX/CLKR)		25†	ns
$t_{w(SCK)}$	Pulse duration, serial port clock (CLKX/CLKR) low (see Note 10)	80		ns
$t_{w(SCK)}$	Pulse duration, serial port clock (CLKX/CLKR) high (see Note 10)	80		ns
$t_{su(FS)}$	Setup time, FSX/FSR before CLKX/CLKR falling edge (TXM = 0)	18		ns
$t_{h(FS)}$	Hold time, FSX/FSR after CLKX/CLKR falling edge (TXM = 0)	20		ns
$t_{su(DR)}$	Setup time, DR before CLKR falling edge	10		ns
$t_{h(DR)}$	Hold time, DR after CLKR falling edge	20		ns

† Value is derived from characterization data and not tested.
NOTE 10: The duty cycle of the serial port clock must be within 40–60%.

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

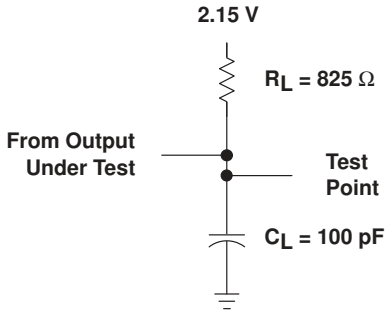
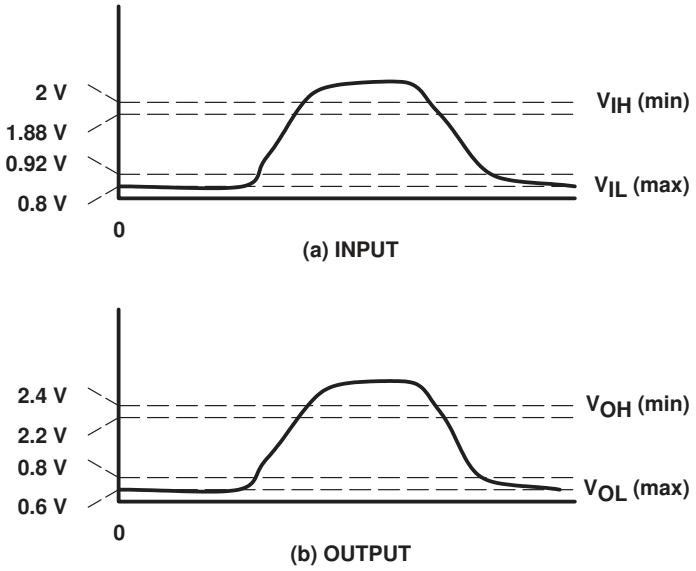


Figure 4. Test Load Circuit



NOTE A. Timing measurements are referenced to and from a low voltage of 0.8 V and a high voltage of 2 V, unless otherwise noted.

Figure 5. Voltage Reference Levels

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

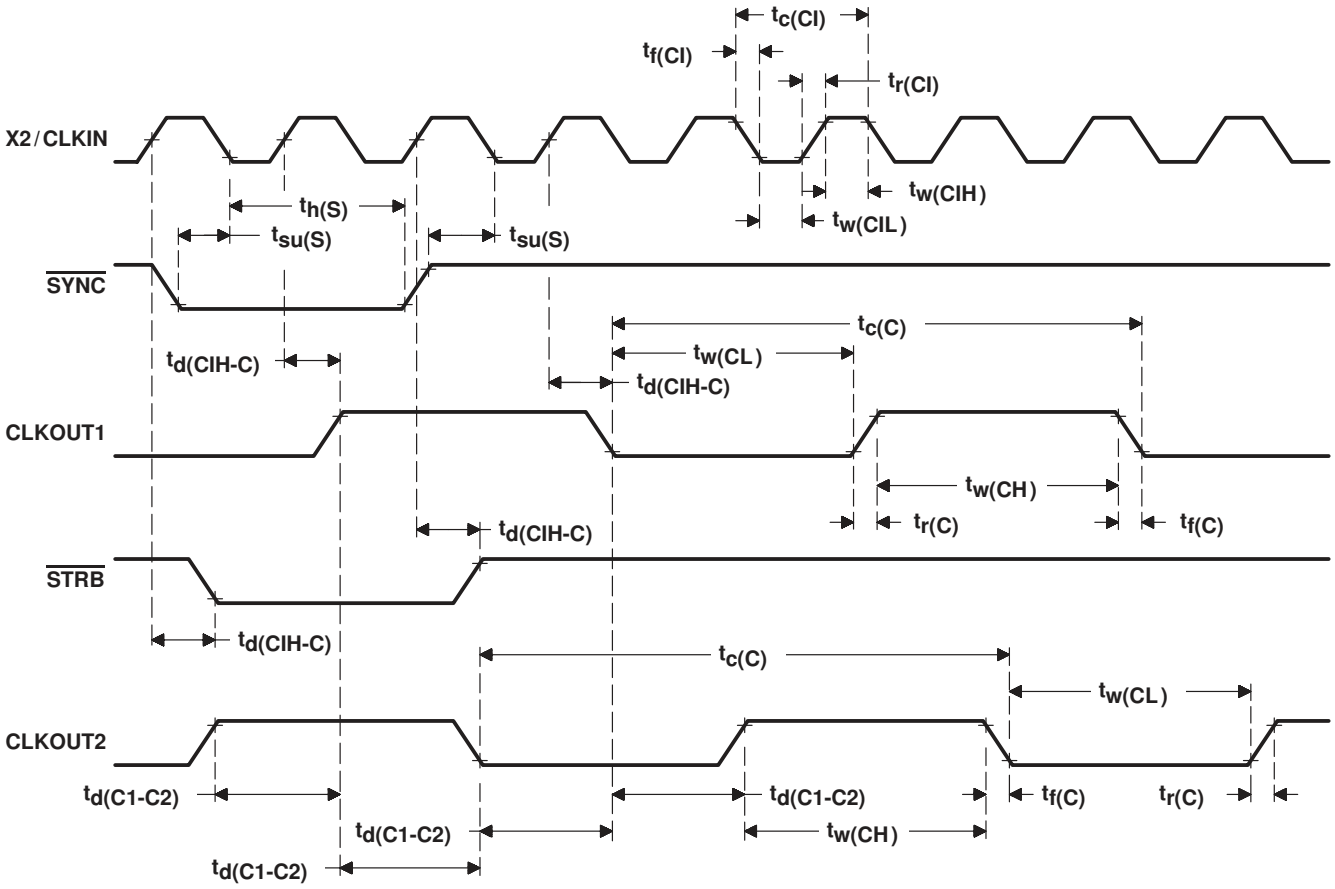


Figure 6. Clock Timing

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

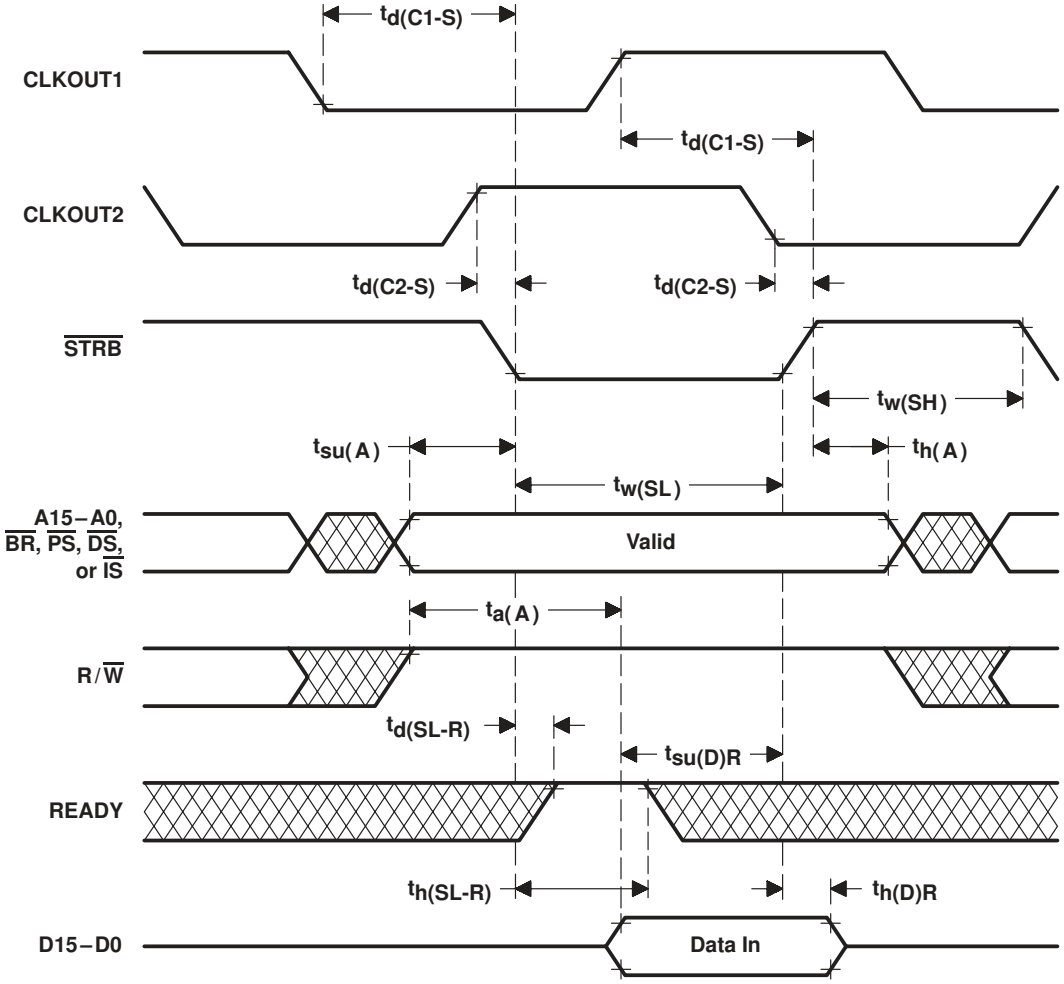


Figure 7. Memory-Read-Cycle Timing

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

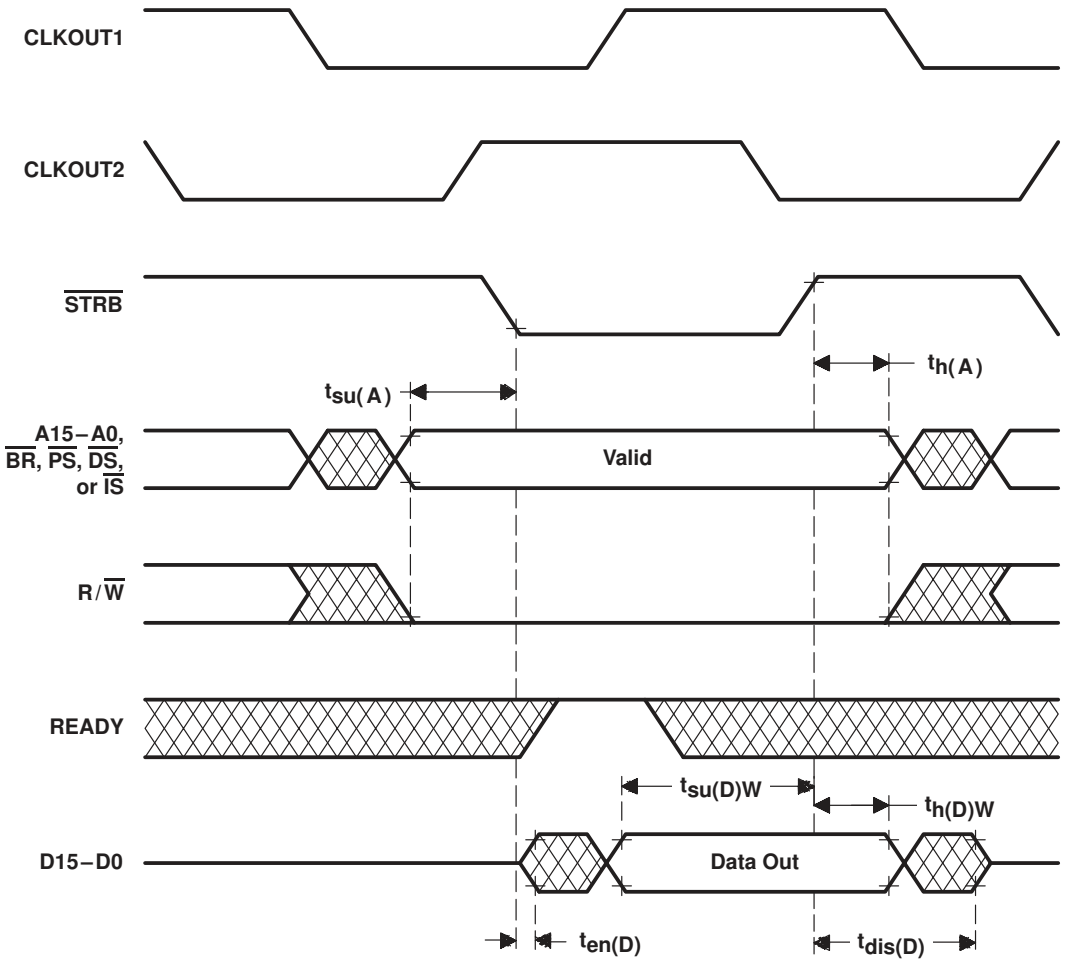


Figure 8. Memory-Write-Cycle Timing

PARAMETER MEASUREMENT INFORMATION

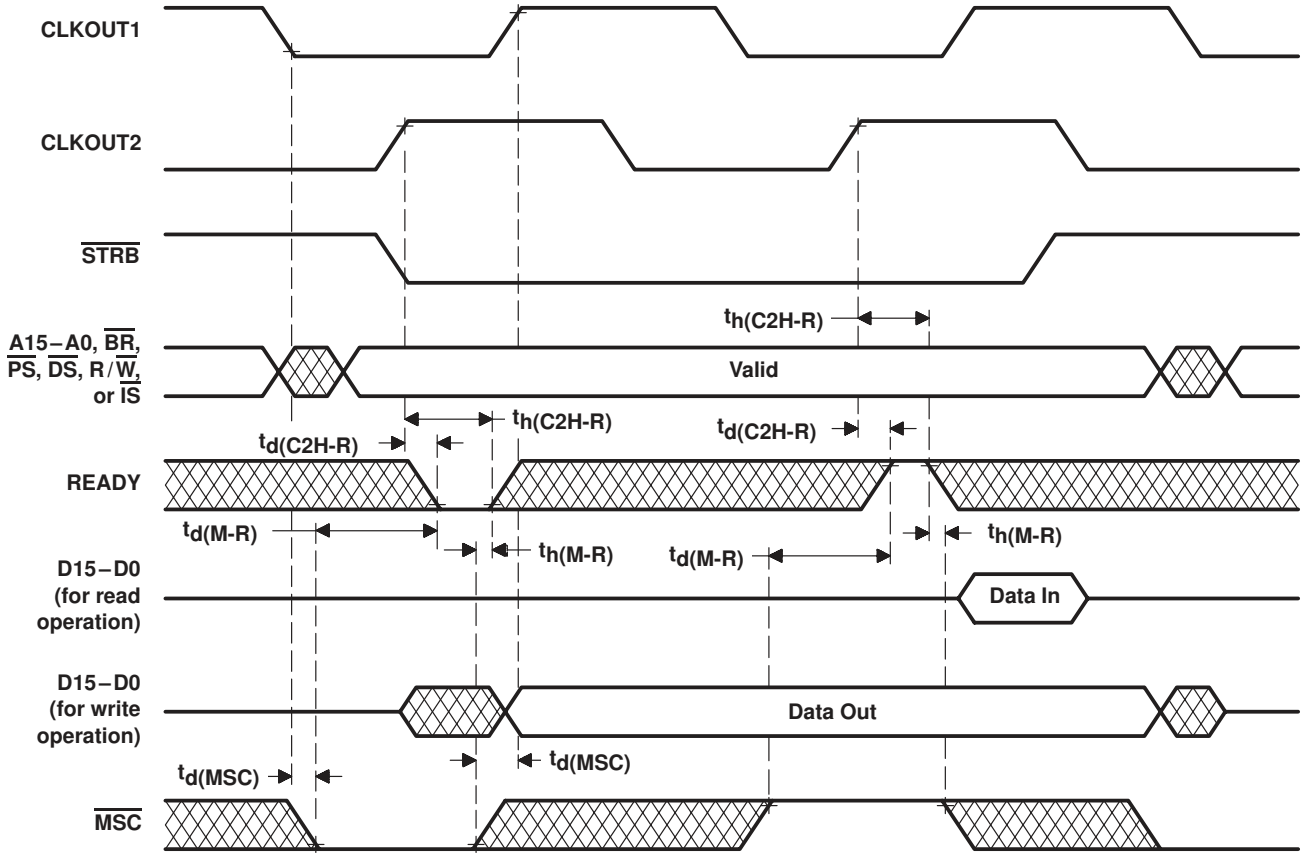
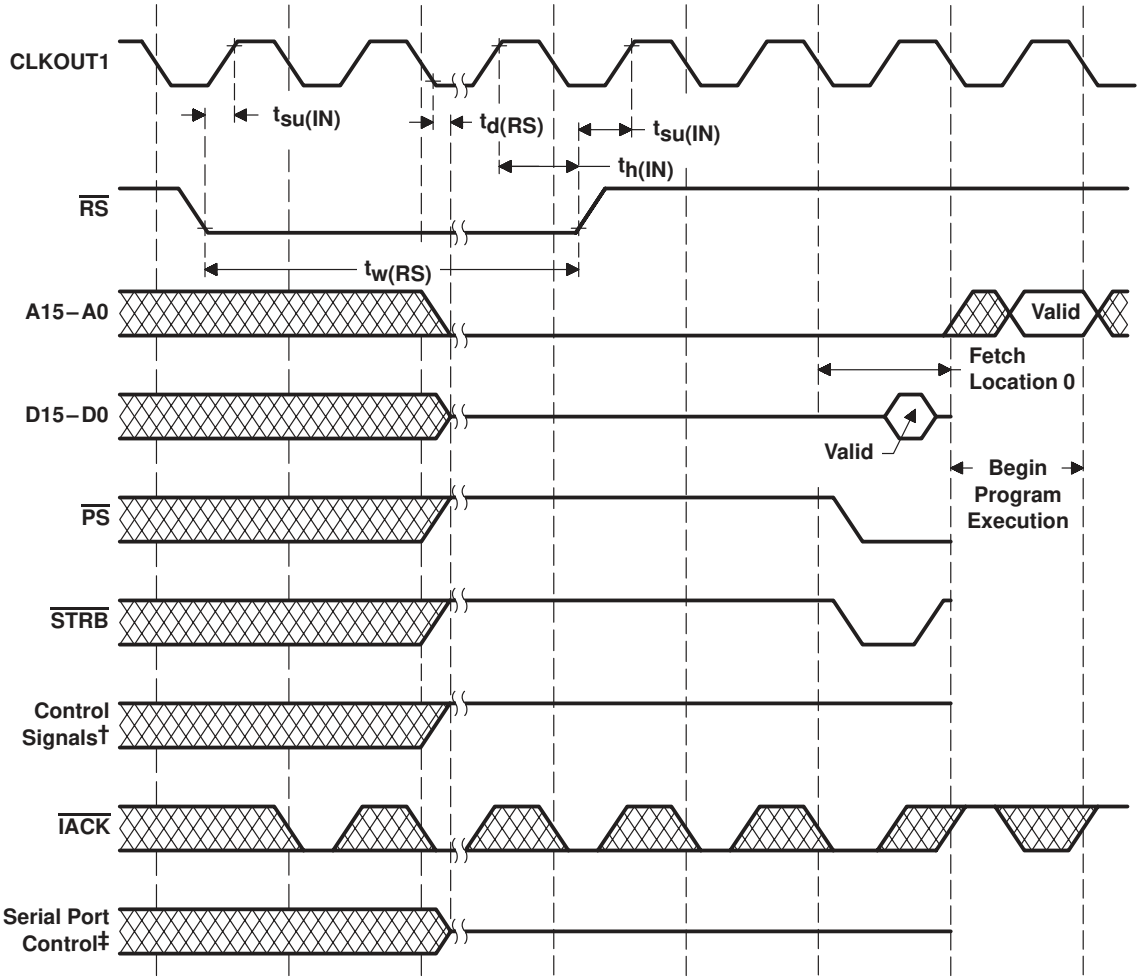


Figure 9. One-Wait-State Memory-Access-Cycle Timing

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION



† Control signals are DS, IS, R/W, and XF.
‡ Serial port controls are DX and FSX.

Figure 10. Reset Timing

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

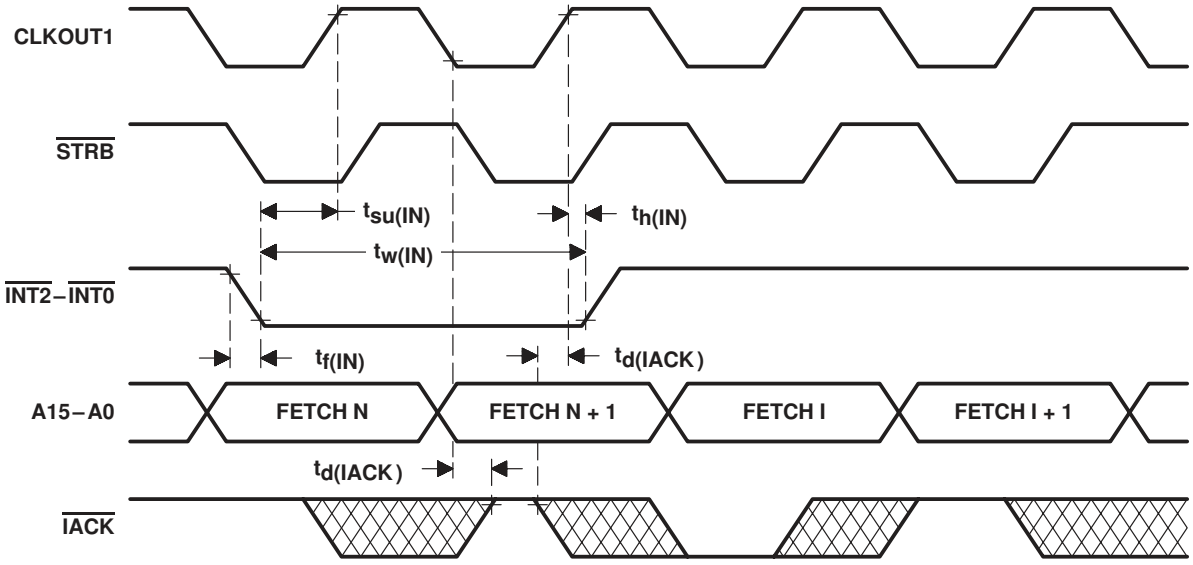


Figure 11. Interrupt Timing

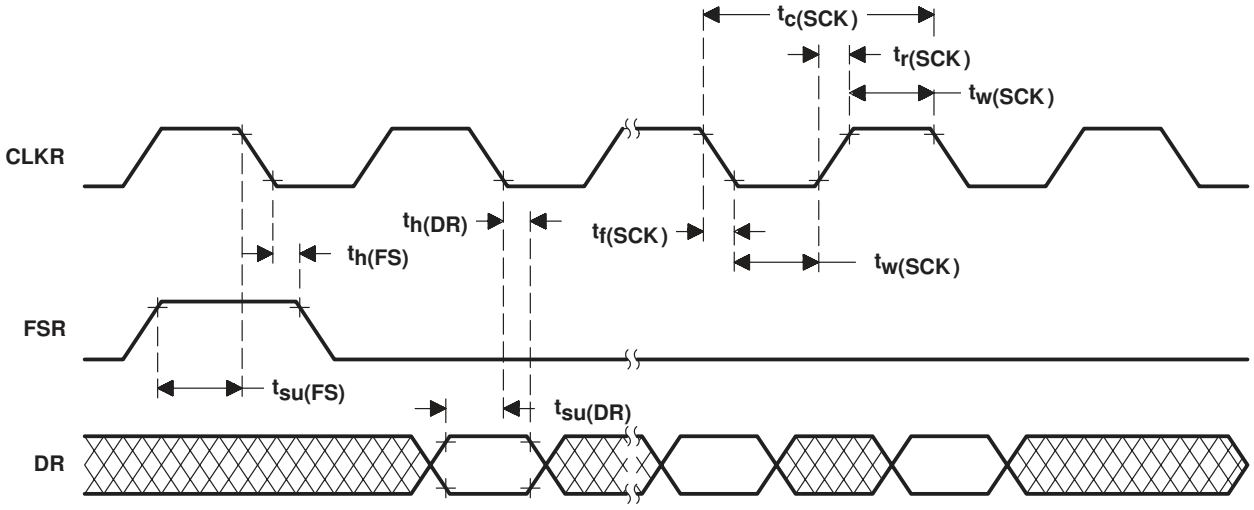


Figure 12. Serial-Port Receive Timing

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

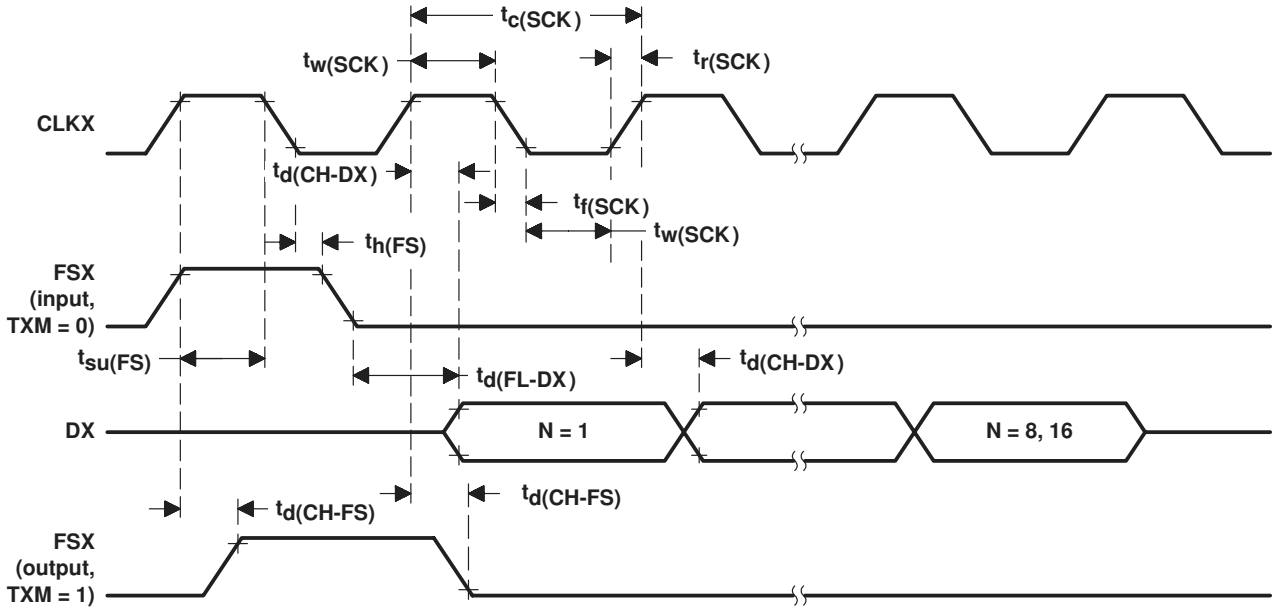


Figure 13. Serial-Port Transmit Timing

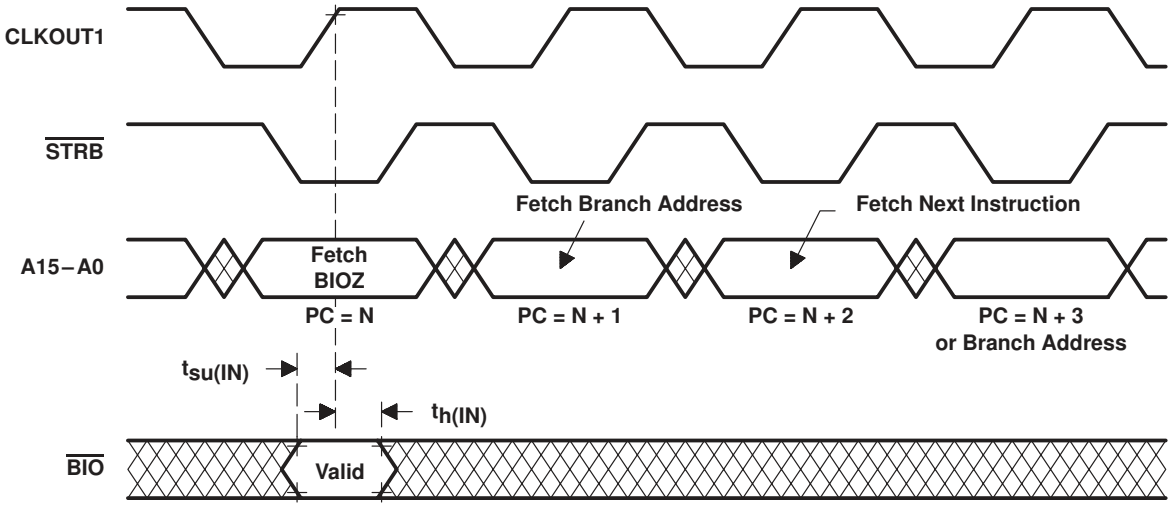


Figure 14. $\overline{\text{BIO}}$ Timing

PARAMETER MEASUREMENT INFORMATION

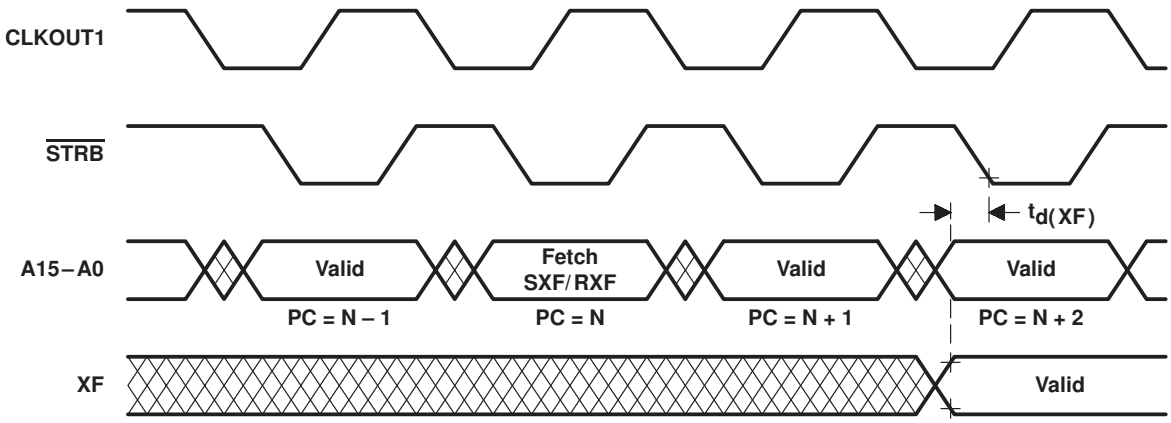
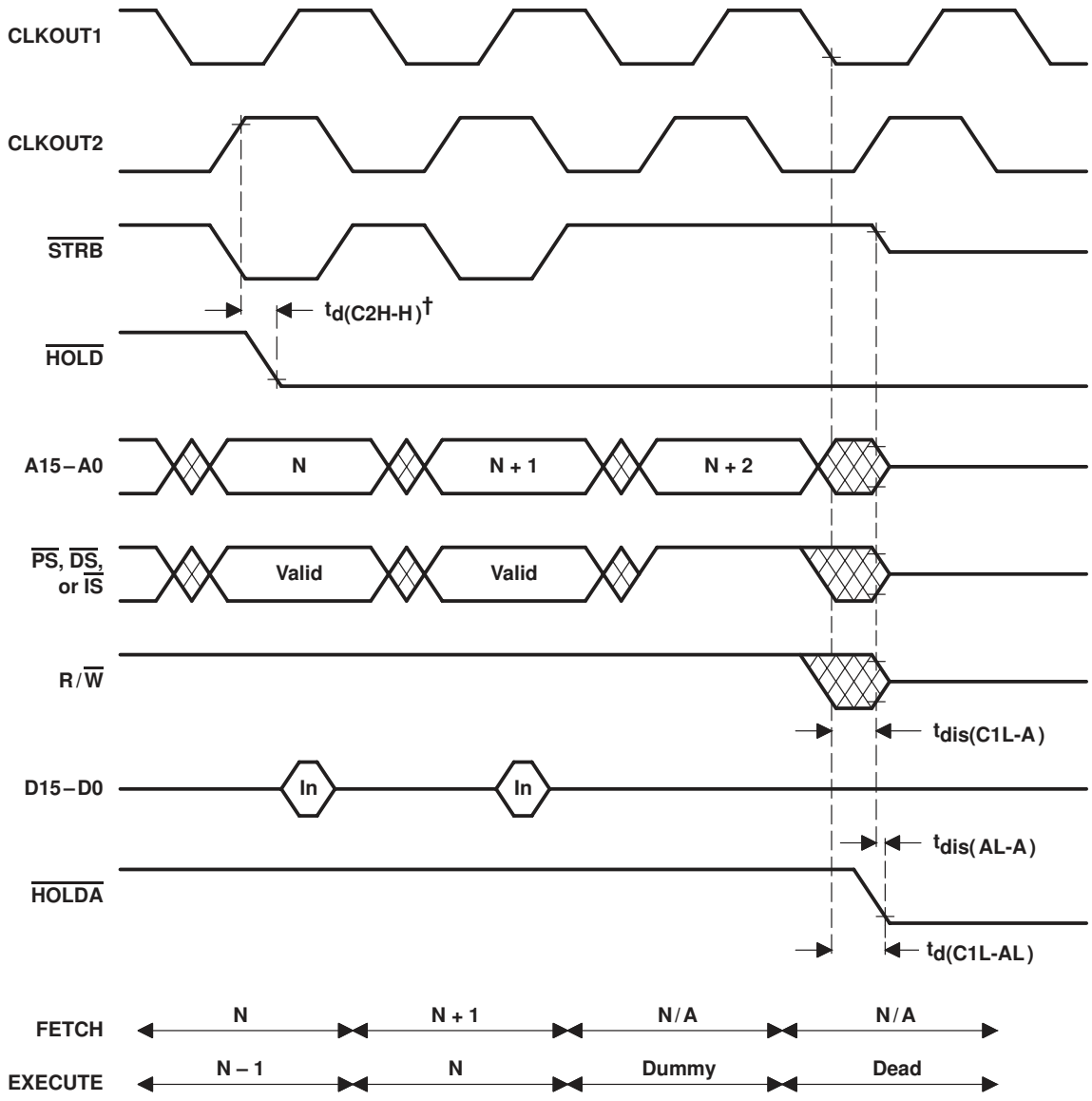


Figure 15. External Flag Timing

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION

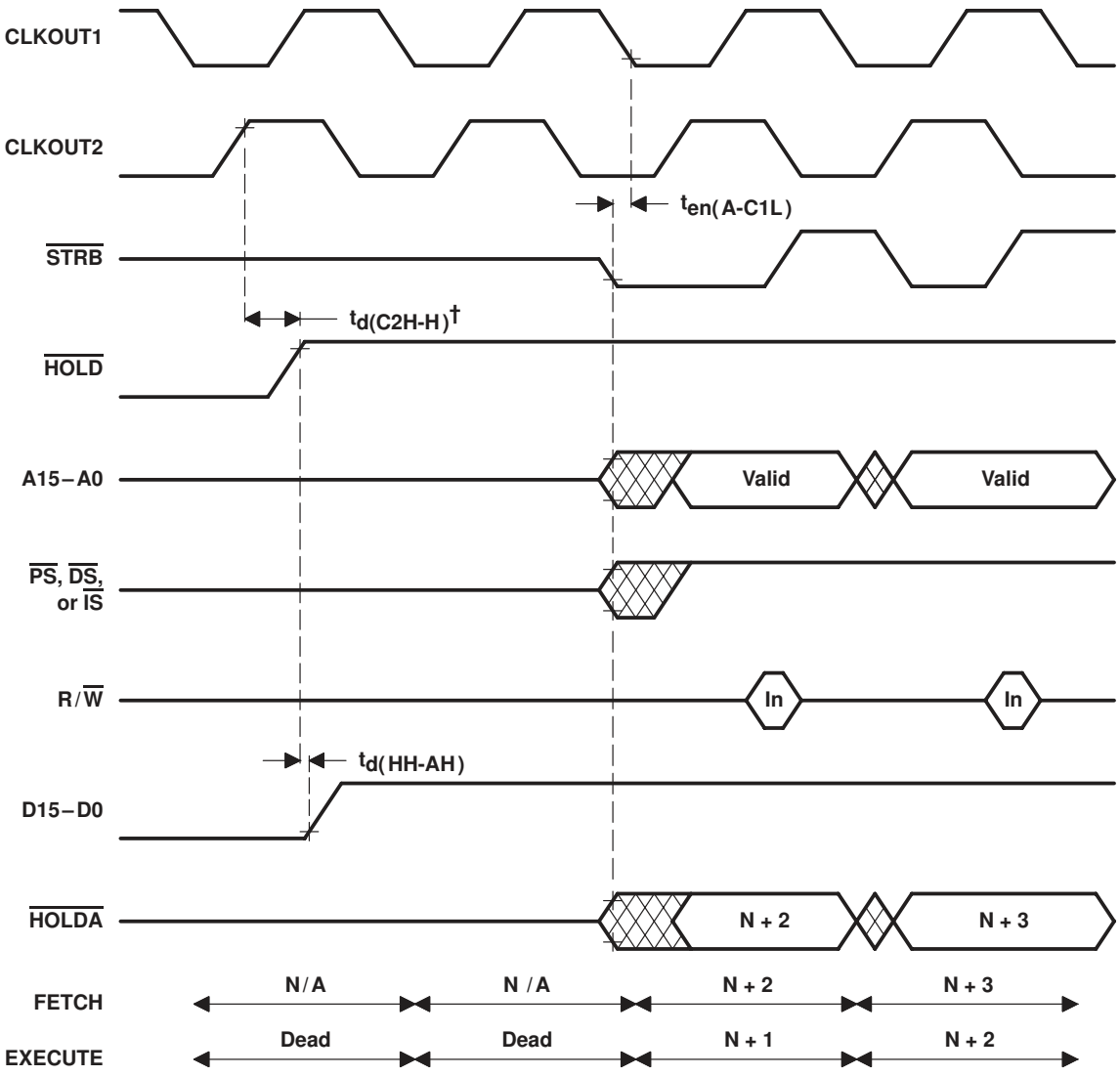


† $\overline{\text{HOLD}}$ is an asynchronous input and can occur at any time during a clock cycle. If the specified timing is met, the exact sequence shown occurs; otherwise, a delay of one CLKOUT2 cycle occurs.

Figure 16. $\overline{\text{HOLD}}$ Timing (Part A)

ADVANCE INFORMATION

PARAMETER MEASUREMENT INFORMATION



† HOLD is an asynchronous input and can occur at any time during a clock cycle. If the specified timing is met, the exact sequence shown occurs; otherwise, a delay of one CLKOUT2 cycle occurs.

Figure 17. HOLD Timing (Part B)

programming the TMS320P25 EPROM cell

The TMS320P25 incorporates a one-time programmable (OTP) 4K × 16-bit EPROM, which is implemented from a standard TMS27C128 EPROM cell. This expands the capabilities of the TMS320P25 in the areas of prototyping, early field testing, and production.

A key feature of the EPROM cell is its use of standard programming techniques with verification capability of all bits. The EPROM cell has an internal security mechanism that prevents all proprietary data from being read and thereby protects privileged information against possible copyright violations. An adapter socket (e.g., part number TMDX3270120 for FN package) provides the 68-lead to 28-lead or 80-lead to 28-lead conversion that is necessary when programming the TMS320P25.

using the EPROM-programmer-adapter socket

Most EPROM programmers have a 28-lead DIP-type socket for use with EPROM devices such as the TMS27C128. In order to use this type of programmer to program the EPROM of a TMS320 device, you must use a special adapter that converts the programmer socket into a socket that can accept a TMS320 device.

Figure 18 shows an example of a PLCC-type adapter socket for the device and the portion that plugs into the EPROM programmer.

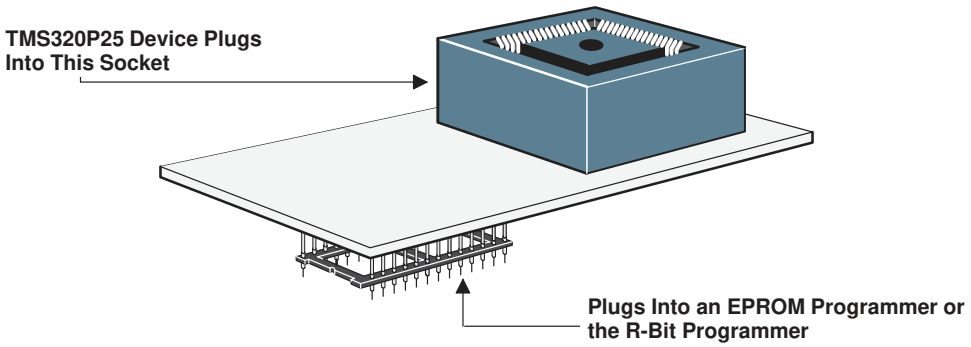


Figure 18. An Example of an EPROM-Programmer Adapter Socket

supplying external power

The adapter socket has two sets of jumpers that indicate whether the power supply is internal (from the EPROM programmer) or external. The adapter socket is shipped from the factory with the jumpers at the internal power setting. In some cases, the EPROM programmer cannot supply the V_{CC} power needs of the TMS320P25 device, so it becomes necessary to supply external V_{CC}. The following conditions determine whether external power is needed:

- The TMS320P25's clock must be disabled during programming. Because the device uses dynamic logic for much of its internal circuitry, the I_{CC} requirements for V_{CC} are significantly greater than a typical '27C128-type EPROM. As a result, many EPROM programmers sense this condition and erroneously indicate that the chip is plugged in backwards. To prevent this from occurring, a jumper connection and test point are available for an external 5-V logic supply. This effectively bypasses the EPROM programmer's I_{CC} test and allows the device to be programmed.
- Additionally, a jumper and test point are available for the V_{PP} supply. The V_{PP} signal is a pulsed signal and fully complies with the standards for a '27C128 EPROM device. This option is never needed, and the jumpers should be left in the internal position at all times.

supplying external power (continued)

To supply external V_{CC}:

- Find the jumper nearest the V_{CC} terminal on the adapter socket and move the jumper so that it is over the EXT and center terminals.
- Connect the external V_{CC} to the terminal labeled V_{CC}.

Figure 19 shows the jumper-setting placement for internal and external power. The V_{CC} and V_{PP} terminals are also shown.

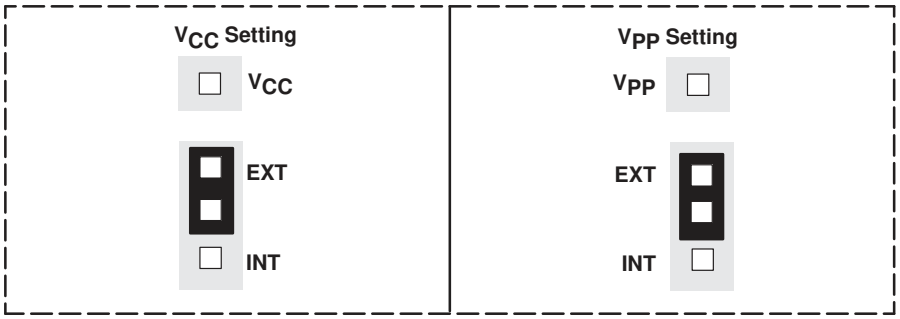


Figure 19. V_{CC} and V_{PP} Jumper Settings for External Power

CAUTION:
Whenever supplying an external V_{CC}, you *must* connect a common ground lead between the power supply and the programmer adapter.

programming and verification

The TMS320P25, like the TMS27C128, requires a 5-V supply for reading and a 12.5-V supply for programming. All programming signals are TTL-level signals. Locations can be systematically or randomly programmed as a single or blocked address. Unlike some EPROM cells that can require the high byte before the low byte, each byte of data must be loaded into the TMS320P25 EPROM cell with the low byte preceding the high byte (see Figure 20). To avoid memorization of the proper order, an inverter is placed in the circuit of Figure 21 and performs the necessary byte reversal for the TMS320P25.

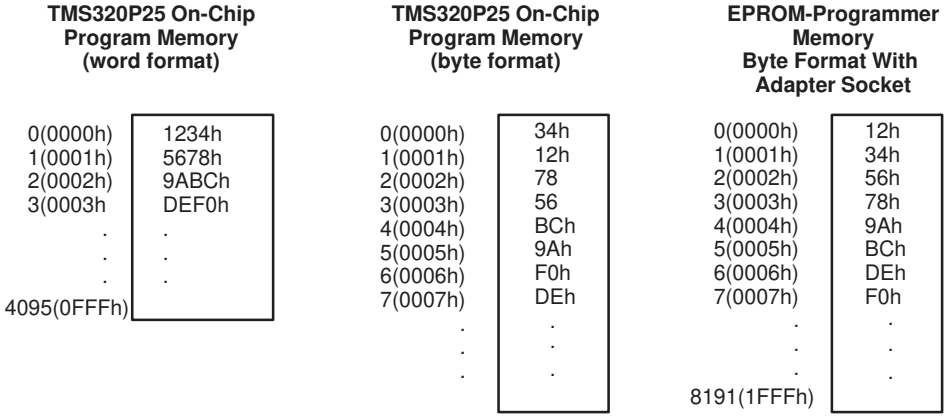


Figure 20. EPROM-Programming Data Format

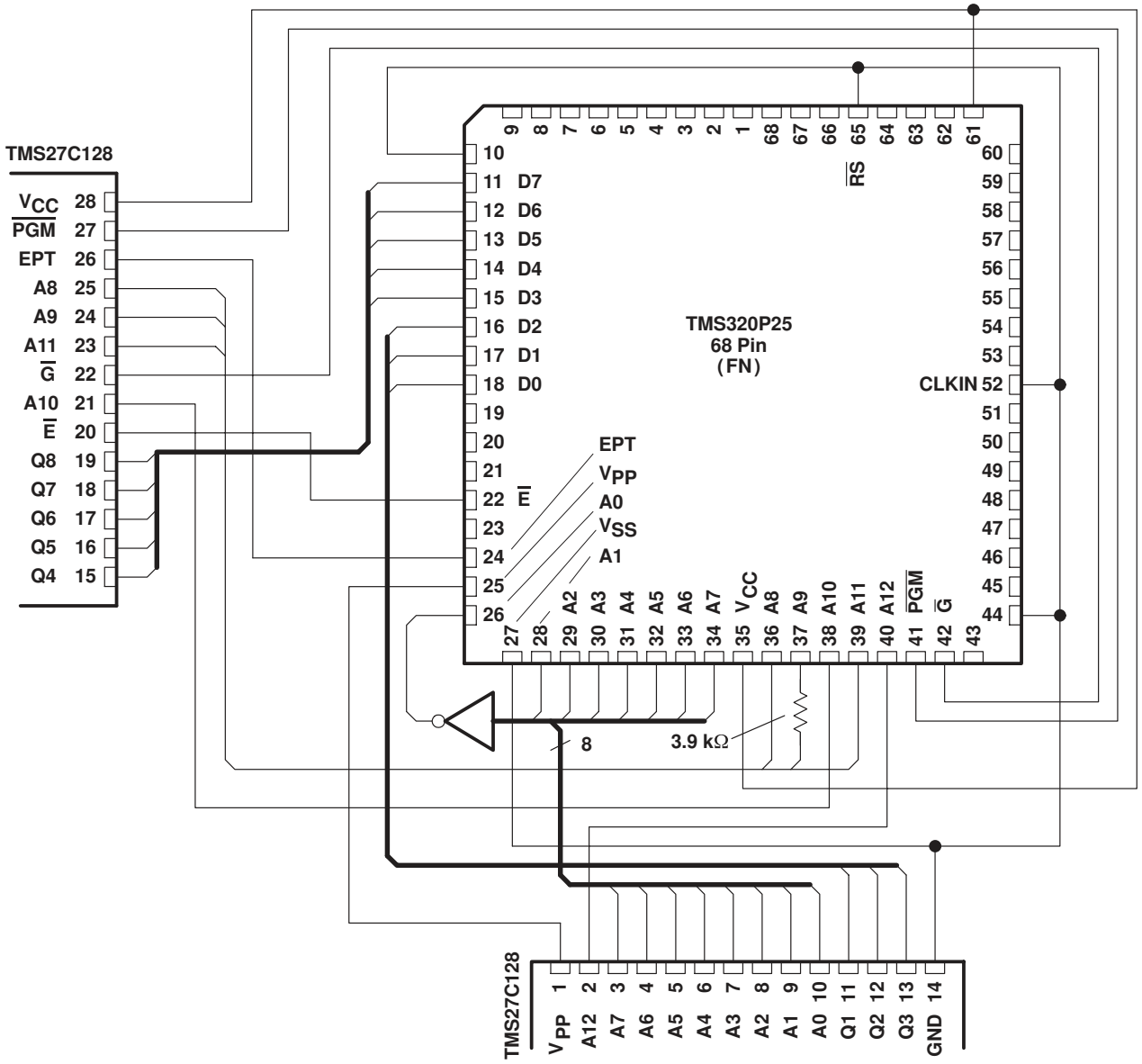
Figure 21 shows the wiring diagram when the TMS320P25 is programmed with the TMS27C128 in its 28-lead output form for the FN package. The illustration furnishes a table for each pin nomenclature on the TMS27C128 with a description of that terminal. Programming the code into the device should be done in the serial mode.

CAUTION:

Although acceptable by some EPROM programmers, the signature mode *cannot* be used on the TMS320P25 device. The signature mode will input a high-level voltage (12.5 V_{DC}) onto pin A9. Because the TMS320P25 EPROM cell is not designed for high voltage, the cell will be damaged in this mode. To prevent an accidental application of voltage, TI has inserted a 3.9-kΩ resistor between A9 of the TI programmer socket and the programmer itself.

ADVANCE INFORMATION

programming and verification (continued)



SIGNALS	I/O	DEFINITION
A12(MSB)–A0 (LSB)	I	On-chip EPROM programming address lines
CLKIN	I	Clock oscillator input
$\overline{E}$	I	EPROM chip select
EPT	I	EPROM test mode select
$\overline{G}$	I	EPROM read/verify select
GND	I	Ground
$\overline{PGM}$	I	EPROM write/program select
Q8(MSB)–Q1(LSB)	I/O	Data lines for byte-wide programming of on-chip 8K bytes of EPROM
$\overline{RS}$	I	Reset for initializing the device
V_{CC}	I	5-V power supply
V_{PP}	I	12.5-V power supply

Figure 21. TMS320P25 EPROM Conversion to TMS27C128 EPROM Pinout Example Using FN Package

ADVANCE INFORMATION

programming and verification (continued)

Table 3 shows the programming levels that are required when programming, verifying, and reading the EPROM cell. Following Table 3 are individual descriptions of each programming level.

Table 3. TMS320P25 Programming-Mode Levels

PROGRAMMING OPERATION SIGNAL	NORMAL OPERATION SIGNAL	TMS320P25 TERMINAL		TMS27C128 TERMINAL	PROGRAM	PROGRAM VERIFY	READ	EPROM PROTECT	PROTECT VERIFY
		FN PKG	PH PKG						
$\overline{E}$	$\overline{INT2}$	22	37	20					
$\overline{G}$	A14	42	61	22	V_{IL}	V_{IL}	V_{IL}	V_{IH}	V_{IL}
$\overline{PGM}$	A13	41	60	27	V_{IH}	$\overline{PULSE}$	$\overline{PULSE}$	V_{IH}	V_{IL}
V_{PP}	FSR	25	40	1	$\overline{PULSE}$	V_{IH}	V_{IH}	V_{IH}	V_{IH}
V_{CC}	V_{CC}	61,35	38, 52	28	V_{PP}	V_{PP}	V_{CC}	V_{PP}	V_{CC} + 1
V_{SS}	V_{SS}	27,44, 10	24, 42, 64	14	V_{CC} + 1	V_{CC} + 1	V_{CC}	V_{CC} + 1	V_{CC} + 1
CLKIN	X2/CLKIN	52	72	14	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}
$\overline{RS}$	$\overline{RS}$	65	8	14	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}
EPT	DR	24	39	26	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}
Q8–Q1	D7–D0	11–18	25–30, 32, 33	19–15, 13–11	V_{SS}	V_{SS}	V_{SS}	V_{PP}	V_{PP}
A12–A7	A12 – A10	40–36, 34	59, 57–54, 51	2, 23, 21, 24, 25, 3	D_{IN}	Q_{OUT}	Q_{OUT}	$Q_8 = \overline{PULSE}$	Q_8 = RBIT
A6	A6	33	50	4	ADDR	ADDR	ADDR	X	V_{IL}
A5	A5	32	48	5	ADDR	ADDR	ADDR	X	X
A4	A4	31	47	6	ADDR	ADDR	ADDR	V_{IH}	X
A3–A0	A3 – A0	30–28, 26	46–44, 41	7–10	ADDR	ADDR	ADDR	X	X

Legend:
ADDR = Byte-address bit
D_{IN} = Byte to be programmed at ADDR
$\overline{PULSE}$ = Low-going TTL pulse
Q_{OUT} = Byte stored at ADDR
RBIT = ROM-protect bit
V_{PP} = 12.5 V ± 0.25 V (Fast) or 13 V ± 0.25 V (SNAP! Pulse)
V_{CC} + 1 = 6 V ± 0.25 V (Fast) or 6.5 V ± 0.25 V (SNAP! Pulse)
X = Don't care

erasure

The TMS320P25 comes with 4K words of EPROM cells set to a logic 1 and ready to be programmed. Because of the type of packaging offered for the TMS320P25, the EPROM is only one-time programmable.

Fast programming

Logic 0s must be programmed into their locations. The Fast programming algorithm, illustrated in Figure 22, is normally used to program the entire EPROM contents, although individual locations can be programmed separately. Data is presented in parallel (eight bits) from terminals D7–D0 of the TMS320P25 to terminals Q8–Q1 of the TMS27C128. Once addresses and data are stable, $\overline{PGM}$ is pulsed. The programming mode is achieved when V_{PP} = 12.5 V, $\overline{PGM}$ = V_{IL}, V_{CC} = 6.0 V, $\overline{G}$ = V_{IH}, and $\overline{E}$ = V_{IL}. More than one TMS320P25 can be programmed simultaneously by connecting the devices in parallel with each other. Locations can be programmed in any order.

ADVANCE INFORMATION

Fast programming (continued)

Fast programming uses two types of programming pulses: prime and final. The length of the prime pulse is 1-ms; this pulse is applied X times. After each prime pulse, the byte being programmed is verified. If correct data is read, the final programming pulse is applied; if correct data is not read, an additional 1-ms prime pulse is applied up to a maximum of 25 times. The final programming pulse is 3X long. This sequence of programming and verifying is performed at V_{CC} = 6 V, and V_{PP} = 12.5 V. When the full Fast programming routine has been completed, all bits are verified with $V_{CC} = V_{PP}$ = 5 V.

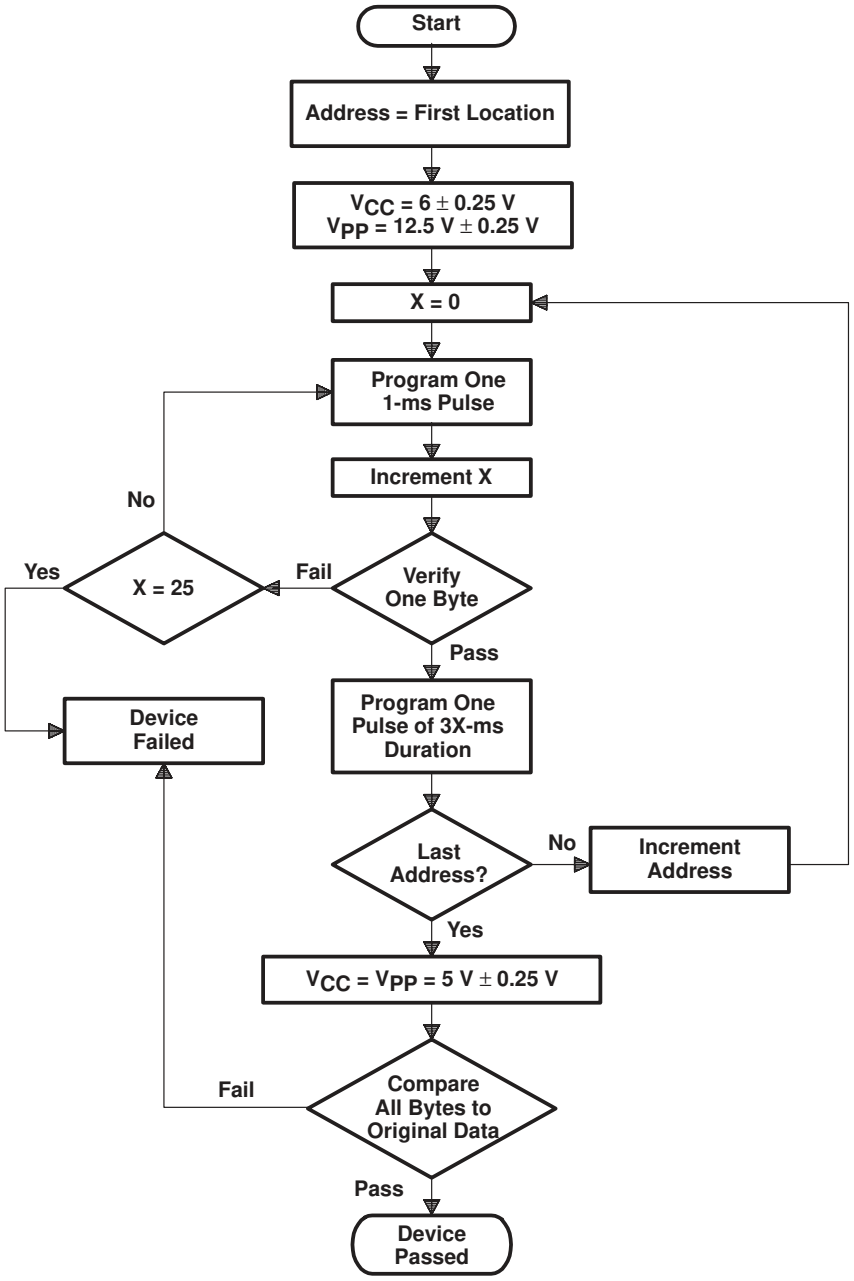


Figure 22. Fast Programming Flowchart

ADVANCE INFORMATION

SNAP! Pulse programming

The EPROM can be programmed by using the TI SNAP! Pulse programming algorithm illustrated in Figure 23. Programming time is greatly reduced to a nominal duration of one second. Actual programming time varies as a function of the programmer that is being used. Data is presented in parallel (eight bits) on terminals Q8–Q1. Once addresses and data are stable, $\overline{PGM}$ is pulsed.

The SNAP! Pulse programming algorithm uses pulses of 100 μs followed by a byte verification to determine if the addressed byte has been successfully programmed. Up to ten 100 μs pulses per byte are verified before a failure is recognized.

The programming mode is achieved when V_{PP} = 13 V, V_{CC} = 6.5 V, and $\overline{G} = V_{IH}$, and $\overline{E} = V_{IL}$. More than one TMS320P25 can be programmed simultaneously by connecting the devices in parallel with each other. Locations can be programmed in any order. When the SNAP! Pulse programming routine has been completed, all bits are verified with $V_{CC} = V_{PP}$ = 5 V.

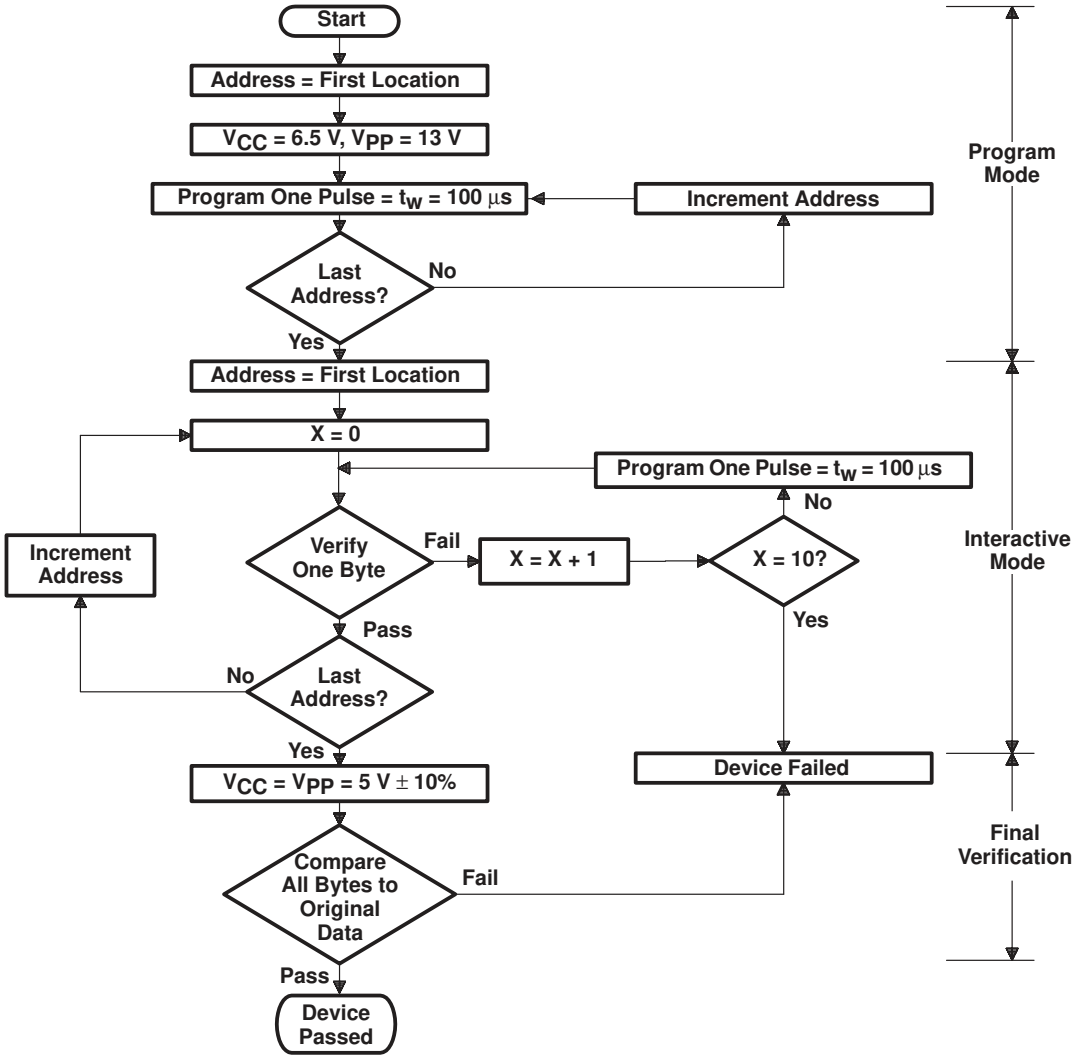


Figure 23. SNAP! Pulse Programming Flowchart

ADVANCE INFORMATION

program verify

Programmed bits can be verified with V_{PP} = 12.5 V when $\overline{G} = V_{IL}$, $\overline{E} = V_{IL}$, and $\overline{PGM} = V_{IH}$. Figure 24 shows the timing of the program and verification operations for both Fast and SNAP! Pulse programming.

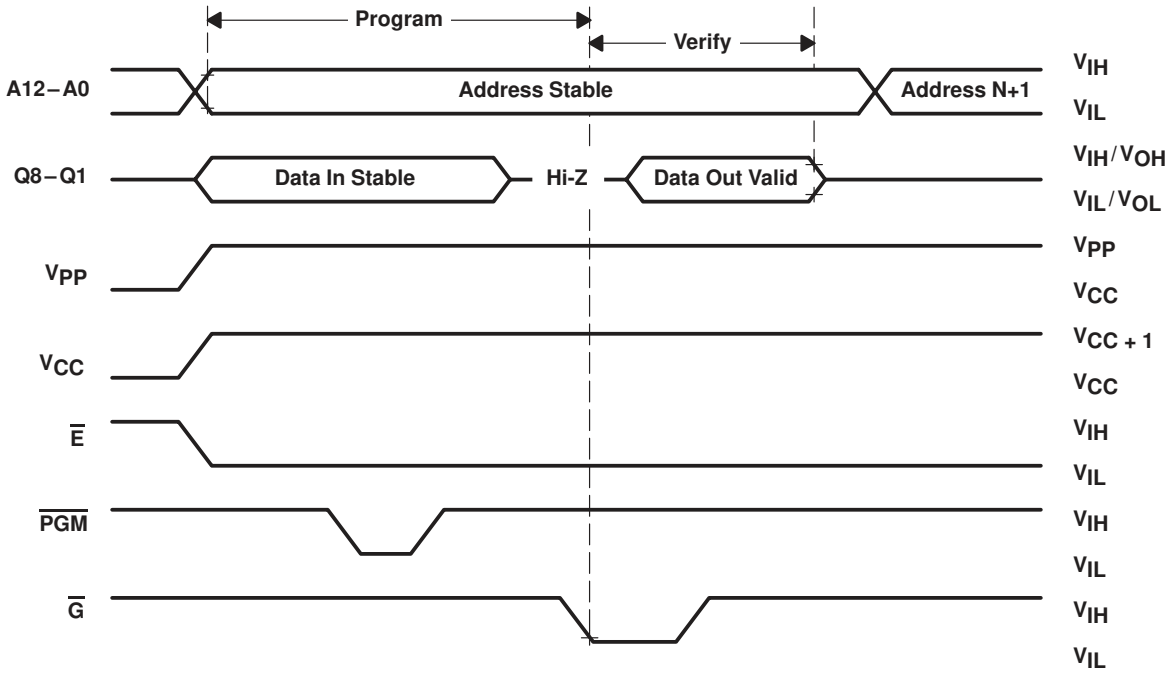


Figure 24. Programming Timing

program inhibit

Programming can be inhibited by maintaining a high-level input on $\overline{E}$ or $\overline{PGM}$.

read

The EPROM contents can be read outside of the programming cycle if the RBIT (ROM-protect bit) has not been programmed. The read mode is accomplished by setting $\overline{E}$ to zero and pulsing G low. The contents of the EPROM location, selected by the value on the address inputs, appear on D7–D0.

output disable

During the EPROM programming process, the EPROM data outputs can be disabled, if desired, by setting the output-disable mode. Depending upon the application, the output-disable mode can be selected by setting either $\overline{G}$ or $\overline{E}$ on the TMS320P25 high. The selection of the terminal determines the duration for which the outputs Q8–Q1 of the TMS27C128 are in the high-impedance state. During this mode, D7–D0 on the TMS320P25 are in the high-impedance state.

EPROM protection and verification

An internal mechanism protects the customer's code from being illegally copied by competitors. Table 4 shows the programming levels required for protecting the EPROM contents and verifying that protection. Following the table, individual functions of the protect and verify modes are described.

Table 4. TMS320P25 EPROM-Protect and Protect-Verify Mode Levels

PROGRAMMING OPERATION SIGNAL	NORMAL OPERATION SIGNAL	TMS320P25 TERMINAL		TMS27C128 TERMINAL	EPROM PROTECT	PROTECT VERIFY
		FN PKG	PH PKG			
$\overline{E}$	$\overline{INT2}$	22	37	20	V_{IH}	V_{IL}
$\overline{G}$	A14	42	61	22	V_{IH}	V_{IL}
$\overline{PGM}$	A13	41	60	27	V_{IH}	V_{IH}
V_{PP}	FSR	25	40	1	V_{PP}	V_{CC} + 1
V_{CC}	V_{CC}	61,35	38, 52	28	V_{CC} + 1	V_{CC} + 1
V_{SS}	V_{SS}	27,44,10	24, 42, 64	14	V_{SS}	V_{SS}
CLKIN	X2/CLKIN	52	72	14	V_{SS}	V_{SS}
$\overline{RS}$	$\overline{RS}$	65	8	14	V_{SS}	V_{SS}
EPT	DR	24	39	26	V_{PP}	V_{PP}
Q8–Q1	D7–D0	11–18	25 – 30, 32, 33	19–15,13–11	$Q_8 = \overline{PULSE}$	Q_8 = RBIT
A12–A10	A12–A10	38	59, 57, 56	2, 23, 21	X	X
A9–A7	A9–A7	37, 36, 34	55, 54, 51	24, 25. 3	X	X
A6	A6	33	50	4	X	V_{IL}
A5	A5	32	48	5	X	X
A4	A4	31	47	6	V_{IH}	X
A3–A0	A3–A0	30–28,26	46 – 44, 41	7–10	X	X

Legend:
$\overline{PULSE}$ = Low-going TTL level pulse
RBIT = ROM-protect bit
V_{CC} + 1 = 6 V ± 0.25 V (Fast) or 6.5 V ± 0.25 V (SNAP! Pulse)
V_{PP} = 12.5 V ± 0.25 V (Fast) or 13 V ± 0.25 V (SNAP! Pulse)
X = Don't care

ADVANCE INFORMATION

EPROM protection

The EPROM-protection mechanism is provided to prevent an intentional or accidental reading of the memory contents, assuring security of all proprietary algorithms. This special feature is implemented by a unique EPROM cell called the RBIT (ROM-protect bit) cell. Once the contents are programmed into the EPROM, the RBIT can be programmed, preventing access to the EPROM contents and disabling the microprocessor mode. Once programmed, the RBIT can be disabled only by erasing the entire EPROM array with ultraviolet light, thereby maintaining security of all proprietary algorithms. Programming of the RBIT is accomplished by the EPROM-protection cycle, which consists of setting $\overline{E}$, $\overline{G}$, $\overline{PGM}$, and A4 to a high level, applying 12.5 V ± 0.25 V to both V_{PP} and EPT, and pulsing Q8 to a low level. The complete sequence of operations is shown in Figure 25.

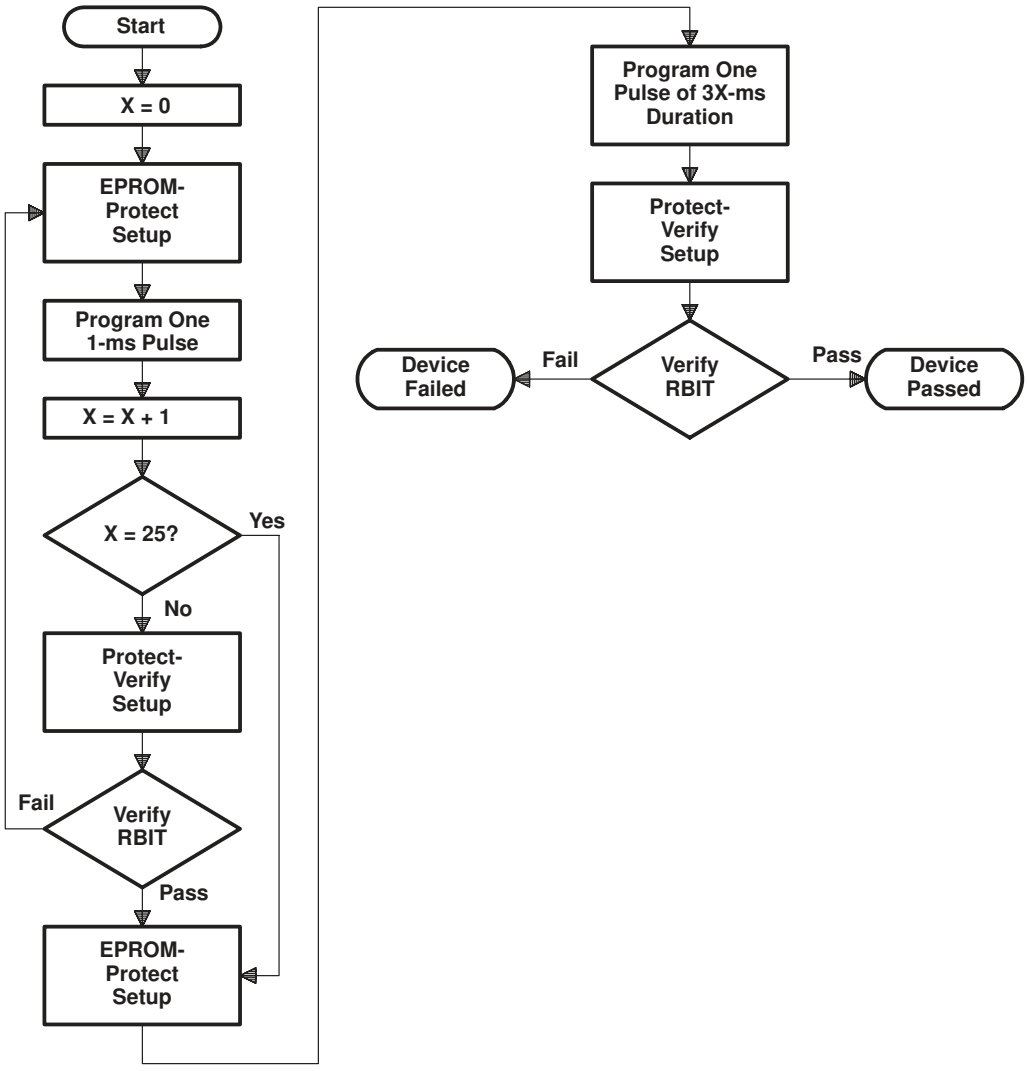


Figure 25. EPROM-Protection Flowchart

how the RBIT works

When enabled, the RBIT disconnects the internal program-memory bus (PBUS) from the multiplexer that combines it with the internal data bus (DBUS) to create the external program/data bus. For the TMS320P25, the internal nodes are left floating. Figure 26 shows a portion of the TMS320P25 block diagram and includes the RBIT to show how it disconnects the external and internal program spaces.

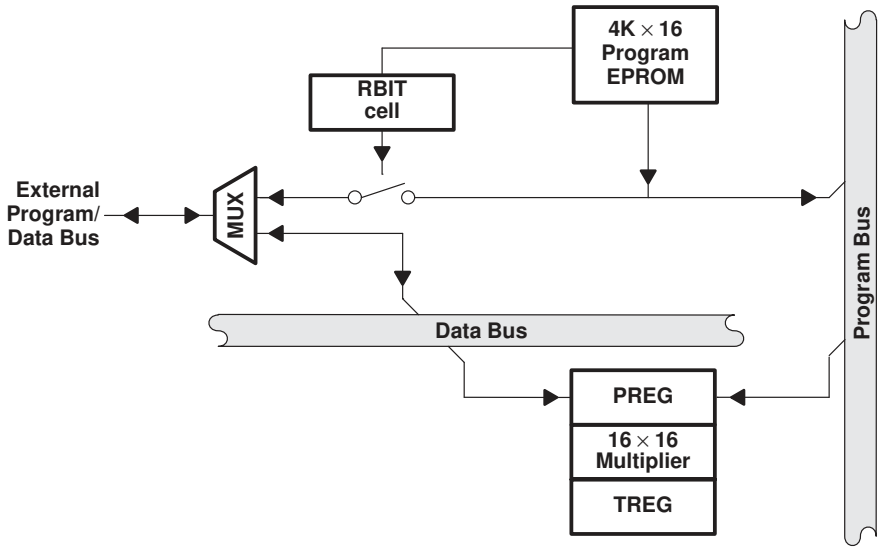


Figure 26. How the RBIT Fits into the TMS320P25 Block Diagram

Programming the RBIT has some side effects that can, at first, give the appearance that the device is not operating properly. However, because enabling the RBIT protects the EPROM space, this is normal operation. These side effects include:

- Instructions. Some instructions that use the external program space for storage will not operate in the same manner when the RBIT is set.

  For example, TBLW, BLKP, and similar commands may seem to work when used to transfer external program memory to the internal data space connected to DBUS. However, a transfer from the internal program space to the external bus will not work. This happens because the RBIT feature is protecting this memory space.

  Similarly, the MAC instruction cannot read tables stored in external program space. In this case, the data and program must be swapped, sacrificing one cycle per repeated instruction.

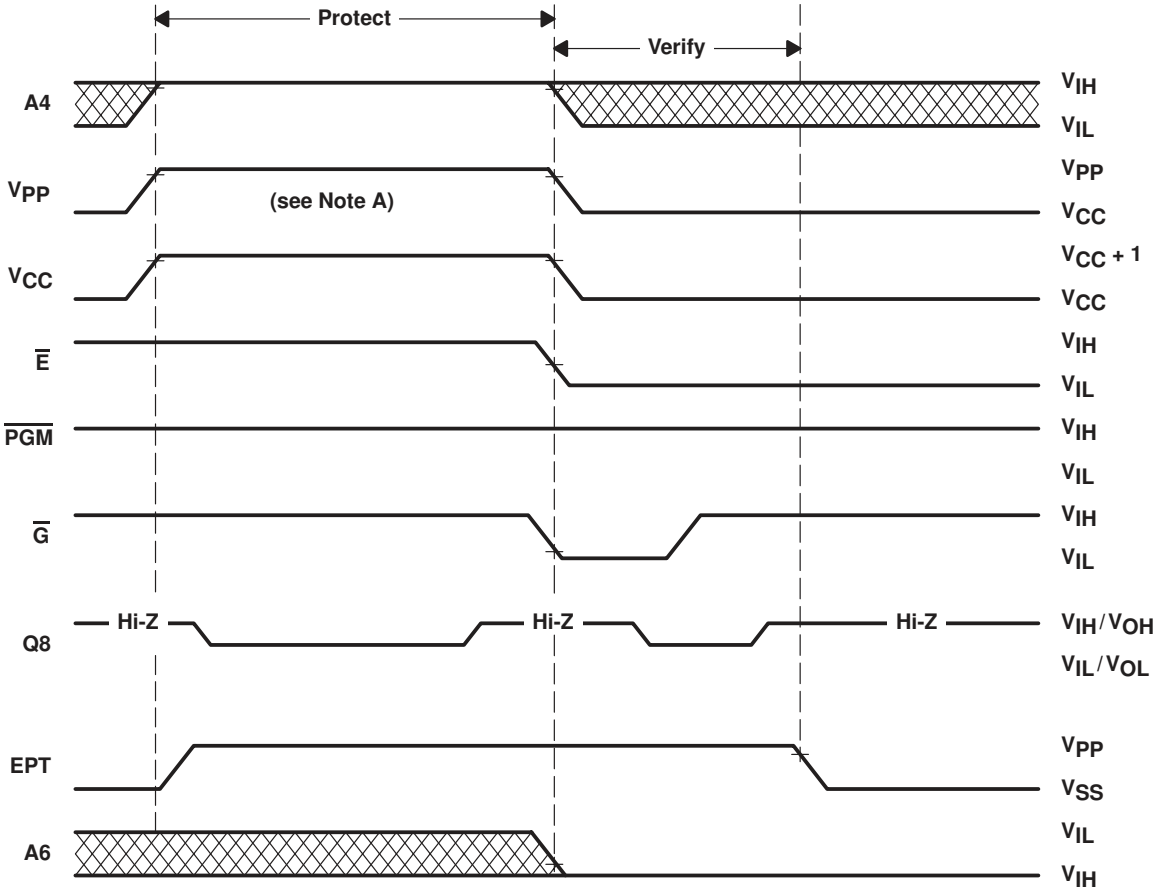
- Invalid microprocessor mode. Microprocessor mode cannot be used after enabling the RBIT because the PBUS is disconnected from the external program space.

protect verify

Following the EPROM-protect mode, the protect-verify mode reviews and verifies the programming of the RBIT for accuracy. When using this mode, D7 outputs the state of the RBIT. When RBIT = 1, the EPROM is unprotected; when RBIT = 0, the EPROM is protected. The EPROM protection and verification timings are shown in Figure 27.

ADVANCE INFORMATION

protect verify (continued)



NOTE B. V_{PP} = 12.5 V and V_{CC} = 6 V for Fast programming; for SNAP! Pulse programming, V_{PP} = 13 V and V_{CC} = 6.5 V.

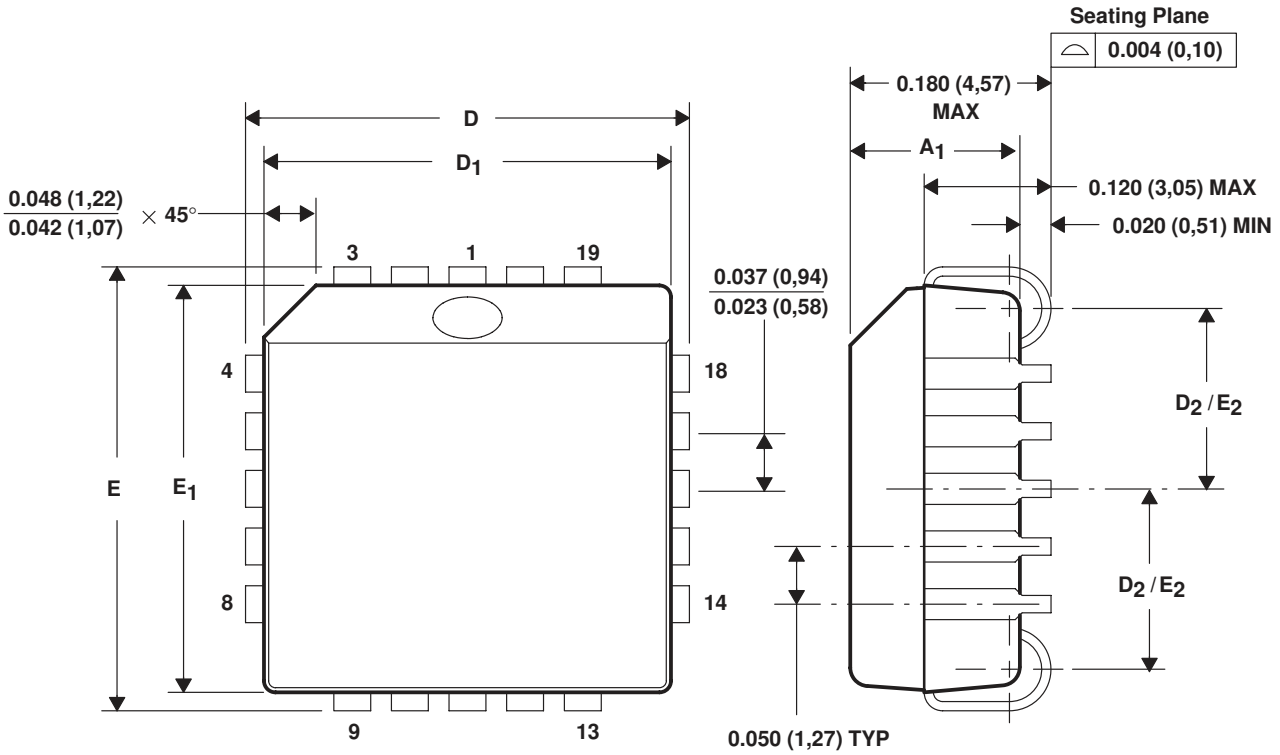
Figure 27. EPROM-Protection Timing

ADVANCE INFORMATION

MECHANICAL DATA

FN/S-PQCC-J** **PLASTIC J-LEADED CHIP CARRIER**

20-PIN SHOWN



JEDEC OUTLINE	NO. OF PINS**	A1		D/E		D1/E1		D2/E2	
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
MS-018AA	20	0.062 (0,16)	0.083 (0,21)	0.385 (9,78)	0.395 (10,03)	0.350 (8,89)	0.356 (9,04)	0.141 (3,58)	0.169 (4,29)
MS-018AB	28	0.062 (0,16)	0.083 (0,21)	0.485 (12,32)	0.495 (12,57)	0.450 (11,43)	0.456 (11,58)	0.191 (4,85)	0.219 (5,56)
MS-018AC	44	0.062 (0,16)	0.083 (0,21)	0.685 (17,40)	0.695 (17,65)	0.650 (16,51)	0.656 (16,66)	0.291 (7,39)	0.319 (8,10)
MS-018AD	52	0.062 (0,16)	0.083 (0,21)	0.785 (19,94)	0.795 (20,19)	0.750 (19,05)	0.756 (19,20)	0.341 (8,66)	0.369 (9,37)
MS-018AE	68	0.062 (0,16)	0.083 (0,21)	0.985 (25,02)	0.995 (25,27)	0.950 (24,13)	0.956 (24,28)	0.441 (11,20)	0.469 (11,91)
MS-018AF	84	0.059 (0,15)	0.080 (0,20)	1.185 (30,10)	1.195 (30,35)	1.150 (29,21)	1.158 (29,41)	0.541 (13,74)	0.569 (14,45)

4040005/A–10/93

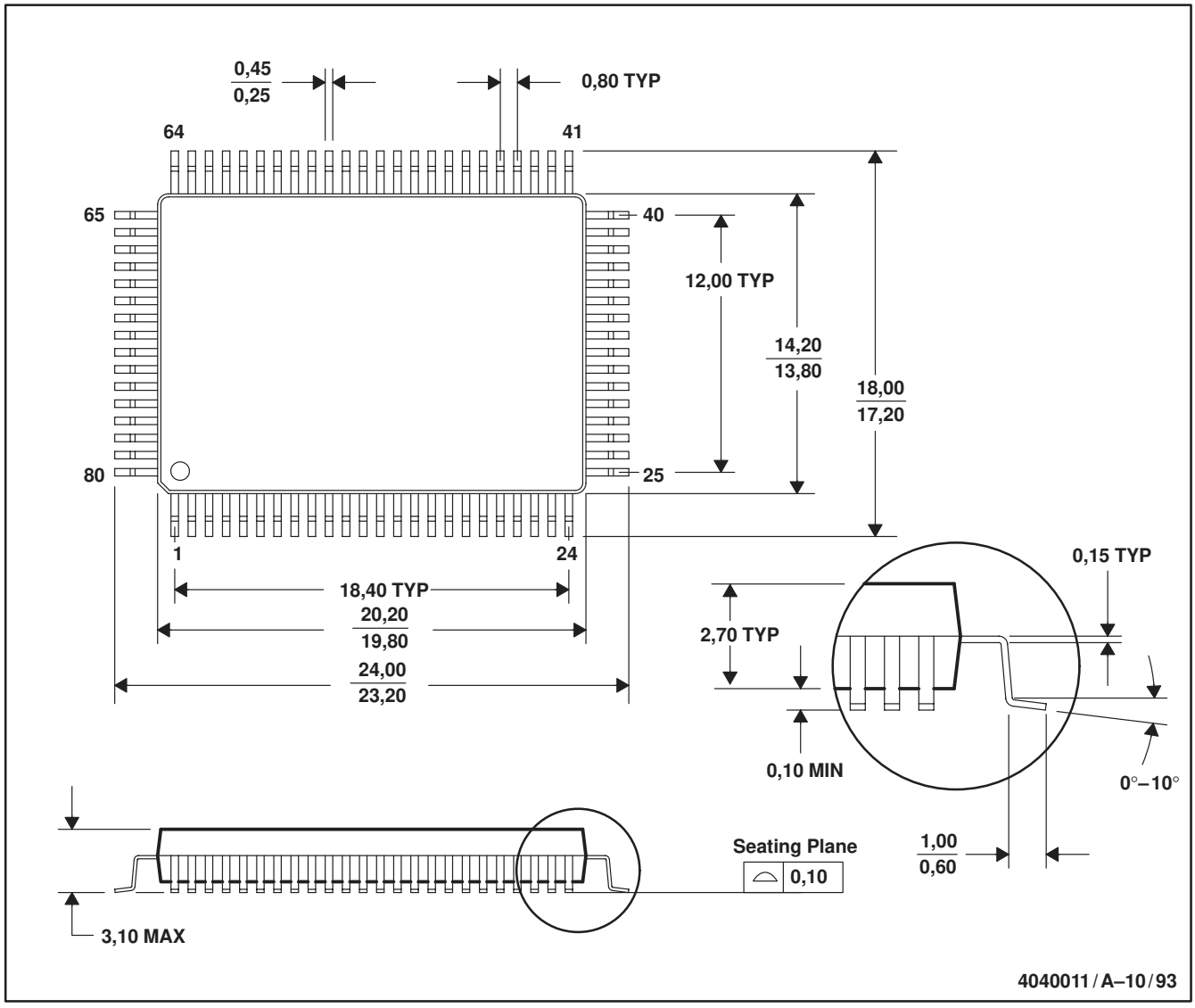
NOTES: C. All linear dimensions are in inches (millimeters).
D. This drawing is subject to change without notice.
E. Falls within JEDEC MS-018

ADVANCE INFORMATION

ADVANCE INFORMATION

MECHANICAL DATA

PH/R-PQFP-G80 **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.

PACKAGING INFORMATION

Orderable Device	Status [1]	Package Type	Package Drawing	Pins	Package Qty	Eco Plan [2]	Lead/Ball Finish	MSL Peak Temp [3]
TMS320P25FNA	OBSOLETE	PLCC	FN	68		TBD	Call TI	Call TI
TMS320P25FNL	OBSOLETE	PLCC	FN	68		TBD	Call TI	Call TI
TMS320P25PH	OBSOLETE	QFP	PH	80		TBD	Call TI	Call TI

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check http://www.ti.com/productcontent for the latest availability information and additional product content details.
TBD: The Pb-Free/Green conversion plan has not been defined.
Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.
Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.
Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

(3) MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

IMPORTANT NOTICE

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

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

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

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

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

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

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

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

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