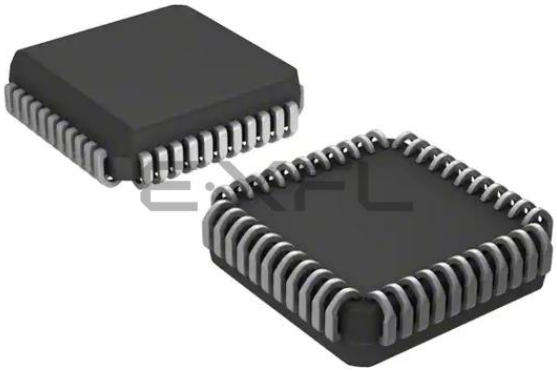


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	36
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p89v662fa-512

3.1 Ordering options

Table 2. Ordering options

Type number	Flash memory	Temperature range	Frequency
P89V662FA	32 kB	–40 °C to +85 °C	0 MHz to 40 MHz
P89V662FBC	32 kB		
P89V664FA	64 kB		
P89V664FBC	64 kB		

4. Block diagram

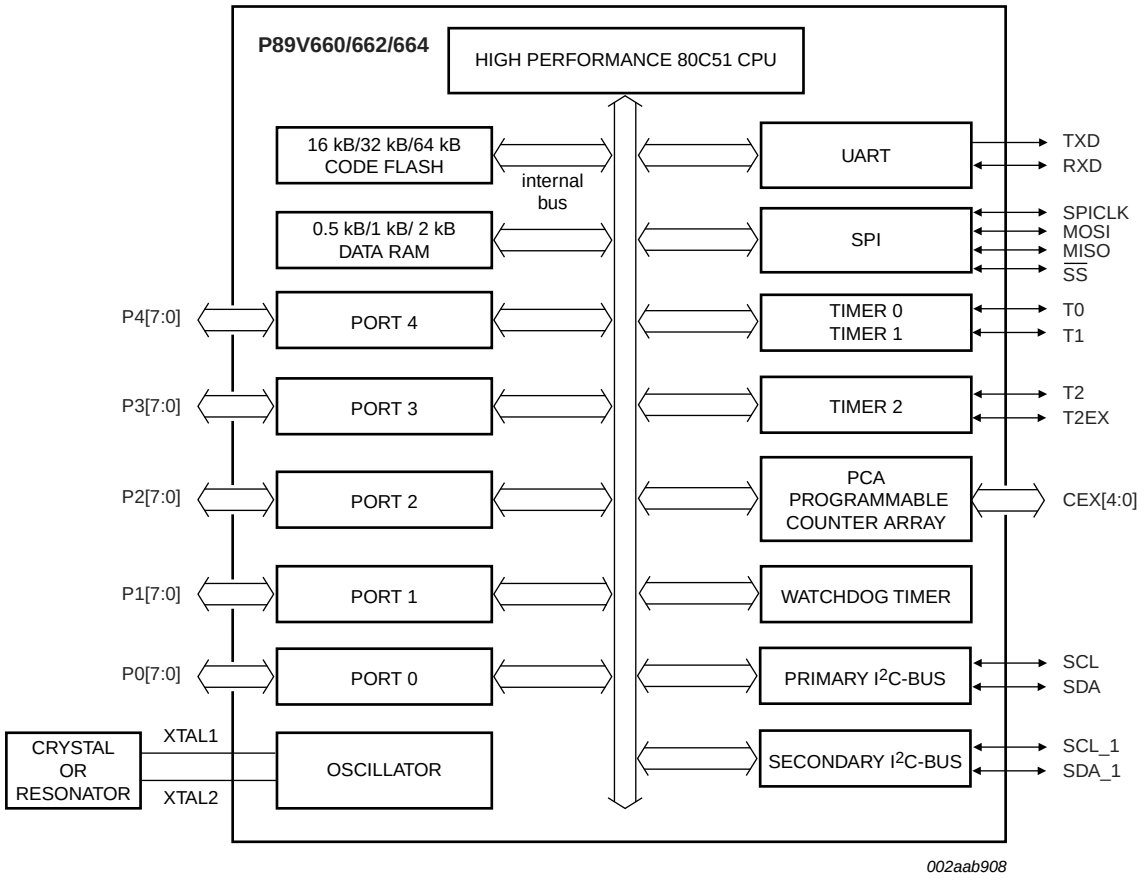


Fig 1. Block diagram

5. Pinning information

5.1 Pinning

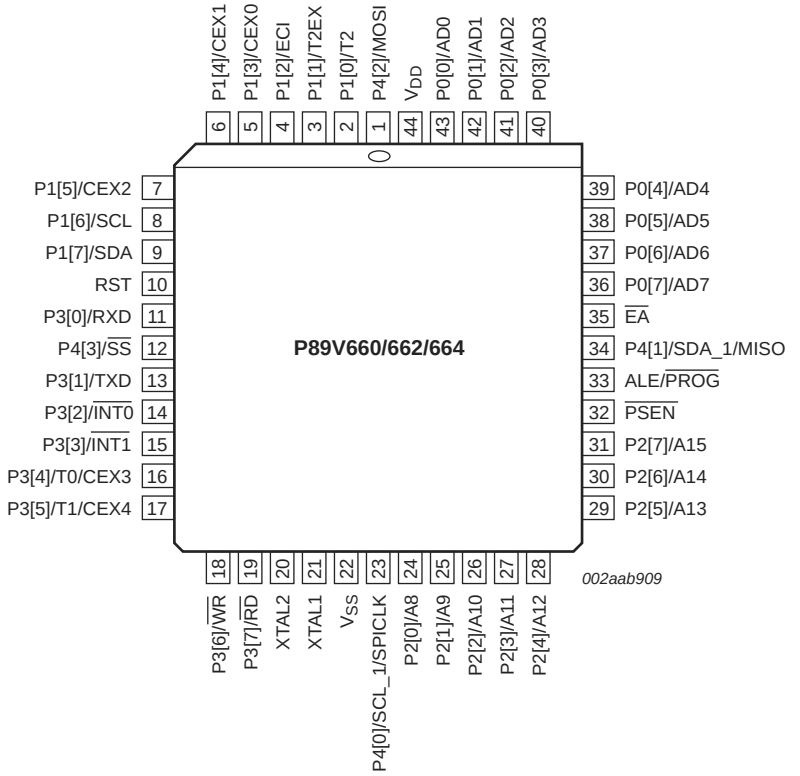


Fig 2. PLCC44 pin configuration

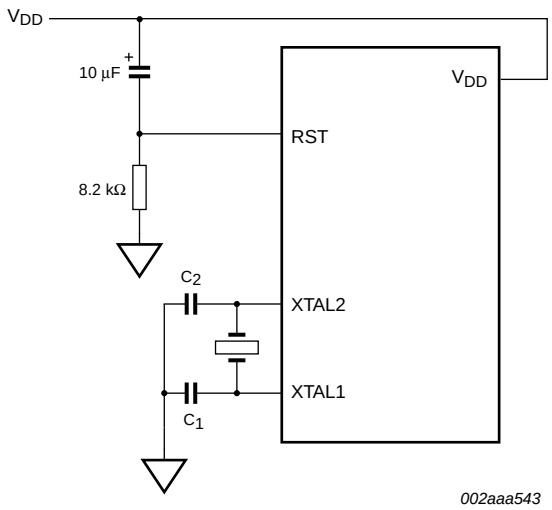


Fig 6. Power-on reset circuit

6.3 Flash memory

6.3.1 Flash organization

The P89V660/662/664 program memory consists of a 16/32/64 kB block for user code. The flash can be read or written in bytes and can be erased in 128 pages. A chip erase function will erase the entire user code memory and its associated security bits. There are three methods of erasing or programming the flash memory that may be used. First, the flash may be programmed or erased in the end-user application by calling LOW-state routines through a common IAP entry point. Second, the on-chip ISP bootloader may be invoked. This ISP bootloader will, in turn, call LOW-state routines through the same common entry point that can be used by the end-user application. Third, the flash may be programmed or erased using the parallel method by using a commercially available EPROM programmer which supports this device.

6.3.2 Features

- Flash internal program memory with 128-byte page erase.
- Internal Boot block, containing LOW-state IAP routines available to user code.
- Boot vector allows user-provided flash loader code to reside anywhere in the flash memory space, providing flexibility to the user.
- Default loader providing ISP via the serial port, located in upper end of program memory.
- Programming and erase over the full operating voltage range.
- Read/Programming/Erase using ISP/IAP.
- Programming with industry-standard commercial programmers.
- 10000 typical erase/program cycles for each byte.
- 100 year minimum data retention.

Table 11. ISP hex record formats *…continued*

Record type	Command/data function
04	Display Device Data or Blank Check :05xxxx04sssseeeeffcc Where 05 = number of bytes in the record xxxx = required field but value is a 'don't care' 04 = function code for display or blank check ssss = starting address, MSB first eeee = ending address, MSB first ff = subfunction 00 = display data 01 = blank check cc = checksum Subfunction codes: Example: :0500000400001FFF00D9 (display from 0000H to 1FFFH)
05	Miscellaneous Read Functions :02xxxx05ffsscc Where: 02 = number of bytes in the record xxxx = required field but value is a 'don't care' 05 = function code for misc read ffss = subfunction and selection code 0000 = read manufacturer id 0001 = read device id 1 0002 = read device id 2 0003 = read 6x/12x bit (bit 7 = 1 is 6x, bit 7 = 0 is 12x) 0080 = read boot code version 0700 = read security bits 0701 = read Status bit 0702 = read Boot vector cc = checksum Example: :020000050000F9 (display manufacturer id)
06	Direct Load of Baud Rate :02xxxx06HHLLcc Where: 02 = number of bytes in the record xxxx = required field but value is a 'don't care' HH = high byte of timer T2 LL = low byte of timer T2 cc = checksum Example: :02000006FFFFcc (load T2 = FFFF)

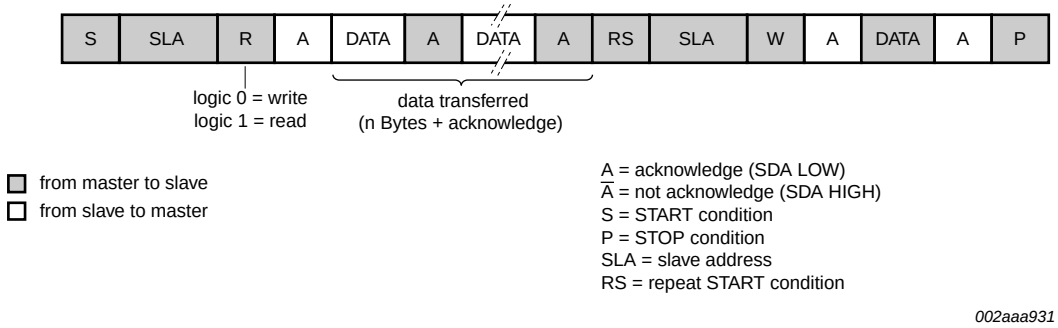


Fig 10. A Master Receiver switches to Master Transmitter after sending Repeated Start

6.4.5.3 Slave receiver mode

In the Slave Receiver mode, data bytes are received from a master transmitter. To initialize the Slave Receiver mode, the user should write the slave address to the Slave Address Register (S1ADR) and the I²C-bus Control Register (S1CON) should be configured as follows:

Table 21. I²C-bus control register (S1CON - address D8H)

Bit	7	6	5	4	3	2	1	0
Symbol	CR2	ENS1	STA	STO	SI	AA	CR1	CR0
Value	-	1	0	0	0	1	-	-

CR2:0 are not used for slave mode. ENS1 must be set = 1 to enable I²C-bus function. AA bit must be set = 1 to acknowledge its own slave address or the general call address. STA, STO and SI are cleared to 0.

After S1ADR and S1CON are initialized, the interface waits until it is addressed by its own address or general address followed by the data direction bit which is 0(W). If the direction bit is 1(R), it will enter Slave Transmitter mode. After the address and the direction bit have been received, the SI bit is set and a valid status code can be read from the Status Register(S1STA). Refer to Table 25 for the status codes and actions.

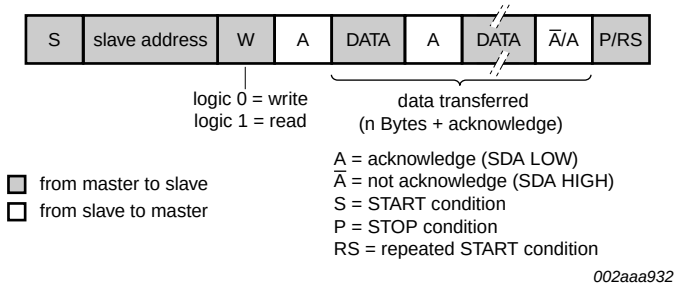


Fig 11. Format of slave receiver mode

6.4.5.4 Slave transmitter mode

The first byte is received and handled as in the Slave Receiver mode. However, in this mode, the direction bit will indicate that the transfer direction is reversed. Serial data is transmitted via P1[7]/SDA while the serial clock is input through P1[6]/SCL. START and

Table 24. Slave Receiver mode *…continued*

Status code (S1STA)	Status of the I²C-bus hardware	Application software response					Next action taken by I²C-bus hardware
		to/from S1DAT	to S1CON				
			STA	STO	SI	AA	
A0H	A STOP condition or repeated START condition has been received while still addressed as SLA/REC or SLA/TRX.	No S1DAT action	0	0	0	0	Switched to not addressed SLA mode; no recognition of own SLA or General call address.
		no S1DAT action	0	0	0	1	Switched to not addressed SLA mode; Own slave address will be recognized; General call address will be recognized if S1ADR.0 = 1.
		no S1DAT action	1	0	0	0	Switched to not addressed SLA mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free.
		no S1DAT action	1	0	0	1	Switched to not addressed SLA mode; Own slave address will be recognized; General call address will be recognized if S1ADR.0 = 1. A START condition will be transmitted when the bus becomes free.

Table 25. Slave transmitter mode

Status code (S1STA)	Status of the I²C-bus hardware	Application software response					Next action taken by I²C-bus hardware
		to/from S1DAT	to S1CON				
			STA	STO	SI	AA	
A8H	Own SLA+R has been received; ACK has been returned.	Load data byte or	x	0	0	0	Last data byte will be transmitted and ACK bit will be received.
		load data byte	x	0	0	1	Data byte will be transmitted; ACK will be received.
B0H	Arbitration lost in SLA+R/W as master; Own SLA+R has been received, ACK has been returned.	Load data byte or	x	0	0	0	Last data byte will be transmitted and ACK bit will be received.
		load data byte	x	0	0	1	Data byte will be transmitted; ACK bit will be received.
B8H	Data byte in S1DAT has been transmitted; ACK has been received.	Load data byte or	x	0	0	0	Last data byte will be transmitted and ACK bit will be received
		load data byte	x	0	0	1	Data byte will be transmitted; ACK will be received.

Table 25. Slave transmitter mode *…continued*

Status code (S1STA)	Status of the I²C-bus hardware	Application software response					Next action taken by I²C-bus hardware
		to/from S1DAT	to S1CON				
			STA	STO	SI	AA	
C0H	Data byte in S1DAT has been transmitted; NOT ACK has been received.	No S1DAT action or	0	0	0	0	Switched to not addressed SLA mode; no recognition of own SLA or General call address.
		no S1DAT action or	0	0	0	1	Switched to not addressed SLA mode; Own slave address will be recognized; General call address will be recognized if S1ADR.0 = 1.
		no S1DAT action or	1	0	0	0	Switched to not addressed SLA mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free.
		no S1DAT action	1	0	0	1	Switched to not addressed SLA mode; Own slave address will be recognized; General call address will be recognized if S1ADR.0 = 1. A START condition will be transmitted when the bus becomes free.
C8H	Last data byte in S1DAT has been transmitted (AA = 0); ACK has been received.	No S1DAT action or	0	0	0	0	Switched to not addressed SLA mode; no recognition of own SLA or General call address.
		no S1DAT action or	0	0	0	1	Switched to not addressed SLA mode; Own slave address will be recognized; General call address will be recognized if S1ADR.0 = 1.
		no S1DAT action or	1	0	0	0	Switched to not addressed SLA mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free.
		no S1DAT action	1	0	0	1	Switched to not addressed SLA mode; Own slave address will be recognized; General call address will be recognized if S1ADR.0 = 1. A START condition will be transmitted when the bus becomes free.

6.5 Timers/counters 0 and 1

The two 16-bit Timer/Counter registers: Timer 0 and Timer 1 can be configured to operate either as timers or event counters (see Table 26 and Table 27).

In the ‘Timer’ function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of six oscillator periods, the count rate is $\frac{1}{6}$ of the oscillator frequency.

In the ‘Counter’ function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled once every machine cycle.

When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register in the machine cycle following the one in which the transition was detected. Since it takes two machine cycles (12 oscillator periods) for 1-to-0 transition to be recognized, the maximum count rate is $^1/_{12}$ of the oscillator frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle. In addition to the ‘Timer’ or ‘Counter’ selection, Timer 0 and Timer 1 have four operating modes from which to select.

The ‘Timer’ or ‘Counter’ function is selected by control bits C/T in the Special Function Register TMOD. These two Timer/Counters have four operating modes, which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timers/Counters. Mode 3 is different. The four operating modes are described in the following text.

Table 26. TMOD - Timer/Counter mode control register (address 89H) bit allocation
Not bit addressable; Reset value: 0000 0000B; Reset source(s): any source

Bit	7	6	5	4	3	2	1	0
Symbol	T1GATE	T1C/$\overline{T}$	T1M1	T1M0	T0GATE	T0C/$\overline{T}$	T0M1	T0M0

Table 27. TMOD - Timer/Counter mode control register (address 89H) bit description

Bit	Symbol	Description
7	T1GATE	Gating control for Timer 1. When set, Timer/Counter is enabled only while the $\overline{INT1}$ pin is high and the TR1 control pin is set. When cleared, Timer 1 is enabled when the TR1 control bit is set.
6	T1C/$\overline{T}$	Timer or Counter select for Timer 1. Cleared for Timer operation (input from CCLK). Set for Counter operation (input from T1 input pin).
5	T1M1	Mode select for Timer 1.
4	T1M0	
3	T0GATE	Gating control for Timer 0. When set, Timer/Counter is enabled only while the $\overline{INT0}$ pin is high and the TR0 control pin is set. When cleared, Timer 0 is enabled when the TR0 control bit is set.
2	T0C/$\overline{T}$	Timer or Counter select for Timer 0. Cleared for Timer operation (input from CCLK). Set for Counter operation (input from T0 input pin).
1	T0M1	Mode Select for Timer 0.
0	T0M0	

Table 28. TMOD - Timer/Counter mode control register (address 89H) M1/M0 operating mode

M1	M0	Operating mode	
0	0	0	8048 timer ‘TLx’ serves as 5-bit prescaler.
0	1	1	16-bit Timer/Counter ‘THx’ and ‘TLx' are cascaded; there is no prescaler.

The Timer 2 as a baud rate generator mode is valid only if RCLK and/or TCLK = 1 in T2CON register. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Thus, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Also if the EXEN2 (T2 external enable flag) is set, a 1-to-0 transition in T2EX (Timer/counter 2 trigger input) will set EXF2 (T2 external flag) but will not cause a reload from (RCAP2H, RCAP2L) to (TH2,TL2). Therefore when Timer 2 is in use as a baud rate generator, T2EX can be used as an additional external interrupt, if needed.

When Timer 2 is in the baud rate generator mode, one should not try to read or write TH2 and TL2. Under these conditions, a read or write of TH2 or TL2 may not be accurate. The RCAP2 registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers. Table 36 shows commonly used baud rates and how they can be obtained from Timer 2.

6.6.5 Summary of baud rate equations

Timer 2 is in baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1[0]) the baud rate is:

Baud rate = Timer 2 overflow rate / 16

If Timer 2 is being clocked internally, the baud rate is:

Baud rate = f_{osc} / (16 × (65536 − (RCAP2H, RCAP2L)))

Where f_{osc} = oscillator frequency

To obtain the reload value for RCAP2H and RCAP2L, the above equation can be rewritten as:

RCAP2H, RCAP2L = 65536 − f_{osc} / (16 × baud rate)

Table 36. Timer 2 generated commonly used baud rates

Rate	Oscillator frequency	Timer 2	
		RCAP2H	RCAP2L
750 kBd	12 MHz	FF	FF
19.2 kBd	12 MHz	FF	D9
9.6 kBd	12 MHz	FF	B2
4.8 kBd	12 MHz	FF	64
2.4 kBd	12 MHz	FE	C8
600 Bd	12 MHz	FB	1E
220 Bd	12 MHz	F2	AF
600 Bd	6 MHz	FD	8F
220 Bd	6 MHz	F9	57

6.7 UARTs

The UART operates in all standard modes. Enhancements over the standard 80C51 UART include Framing Error detection, and automatic address recognition.

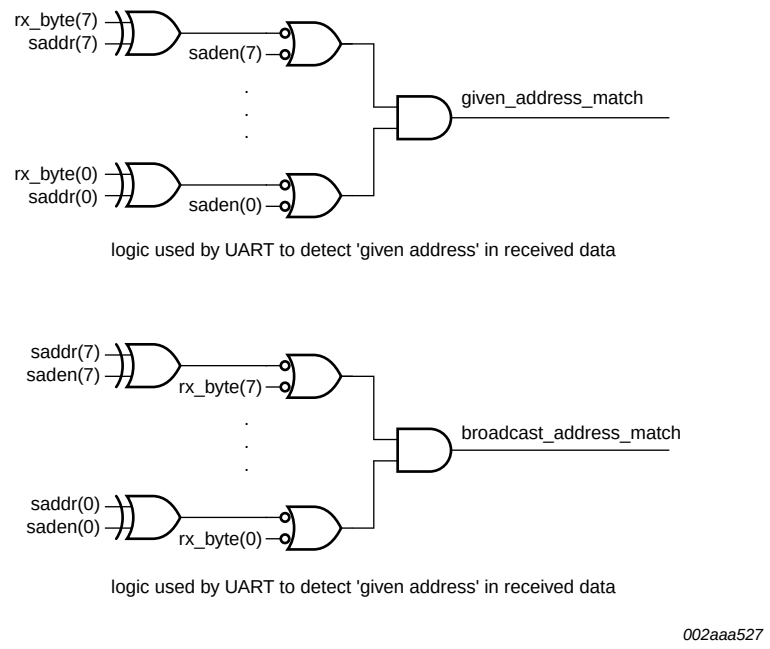


Fig 22. Schemes used by the UART to detect 'given' and 'broadcast' addresses when multiprocessor communications is enabled

The following examples will help to show the versatility of this scheme.

Example 1, slave 0:

$$\begin{array}{l} \text{SADDR} = 1100\ 0000 \\ \underline{\text{SADEN} = 1111\ 1101} \\ \text{Given} = 1100\ 00\text{X}0 \end{array} \quad (4)$$

Example 2, slave 1:

$$\begin{array}{l} \text{SADDR} = 1100\ 0000 \\ \underline{\text{SADEN} = 1111\ 1110} \\ \text{Given} = 1100\ 000\text{X} \end{array} \quad (5)$$

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a '0' in bit 0 and it ignores bit 1. Slave 1 requires a '0' in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a '0' in bit 1. A unique address for slave 1 would be 1100 0001 since a '1' in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Example 1, slave 0:

SADDR = 1100 0000
SADEN = 1111 1001
Given = 1100 0XX0 (6)

Example 2, slave 1:

SADDR = 1110 0000
SADEN = 1111 1010
Given = 1110 0X0X (7)

Example 2, slave 2:

SADDR = 1100 0000
SADEN = 1111 1100
Given = 1100 00XX (8)

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2. The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal. Upon reset SADDR and SADEN are loaded with 0s. This produces a given address of all 'don't cares' as well as a Broadcast address of all 'don't cares'. This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard UART drivers which do not make use of this feature.

6.8 Serial Peripheral Interface (SPI)

6.8.1 SPI features

- Master or slave operation
- 10 MHz bit frequency (max)
- LSB first or MSB first data transfer
- Four programmable bit rates
- End of transmission (SPIF)
- Write collision flag protection (WCOL)
- Wake-up from Idle mode (slave mode only)

6.8.2 SPI description

The serial peripheral interface allows high-speed synchronous data transfer between the P89V660/662/664 and peripheral devices or between several P89V660/662/664 devices. Figure 23 shows the correspondence between master and slave SPI devices. The

6.11 Security bits

The security bits protects against software piracy and prevents the contents of the flash from being read by unauthorized parties in Parallel Programmer mode and ISP mode. Since the end application might need to erase pages and read from the code memory, the security bits have no effect in IAP mode. However, the security bits' programmed/erased state may be read using IAP function calls allowing the end user code to limit access, if desired. The security bits and their effects are shown in Table 53.

Note: On this device MOVC instructions executed from external code memory are disabled from fetching code bytes from internal code memory.

Table 53. Security bit functions

Security bit	Description
1	Write protect. When programmed, prohibits further erasing or programming, except to program other security bits or a chip erase.
2	Read protect. When programmed inhibits reading of user code memory.
3	External execution inhibit. When programmed prevents any execution of instructions from external code memory.

6.12 Interrupt priority and polling sequence

The device supports eight interrupt sources under a four level priority scheme. Table 54 summarizes the polling sequence of the supported interrupts. Note that the SPI serial interface and the UART share the same interrupt vector. (See Figure 32).

Table 54. Interrupt polling sequence

Description	Interrupt flag	Vector address	Interrupt enable	Interrupt priority	Service priority	Wake-up Power-down
External Interrupt 0	IE0	0003H	EX0	PX0/H	1 (highest)	yes
T0	TF0	000BH	ET0	PT0/H	3	no
External Interrupt 1	IE1	0013H	EX1	PX1/H	4	yes
T1	TF1	001BH	ET1	PT1/H	5	no
UART	TI/RI	0023H	ES0	PS0/H	6	no
I²C-bus (primary)	-	002BH	ES1	PS1/H	2	no
PCA	CF/CCFn	0033H	EC	PPCH	8	no
T2	TF2, EXF2	003BH	ET2	PT2/H	7	no
I²C-bus (secondary)	-	0043H	ES2	PS2/H	10	no
SPI	SPIF	004BH	ES3	PS3/H	9	no

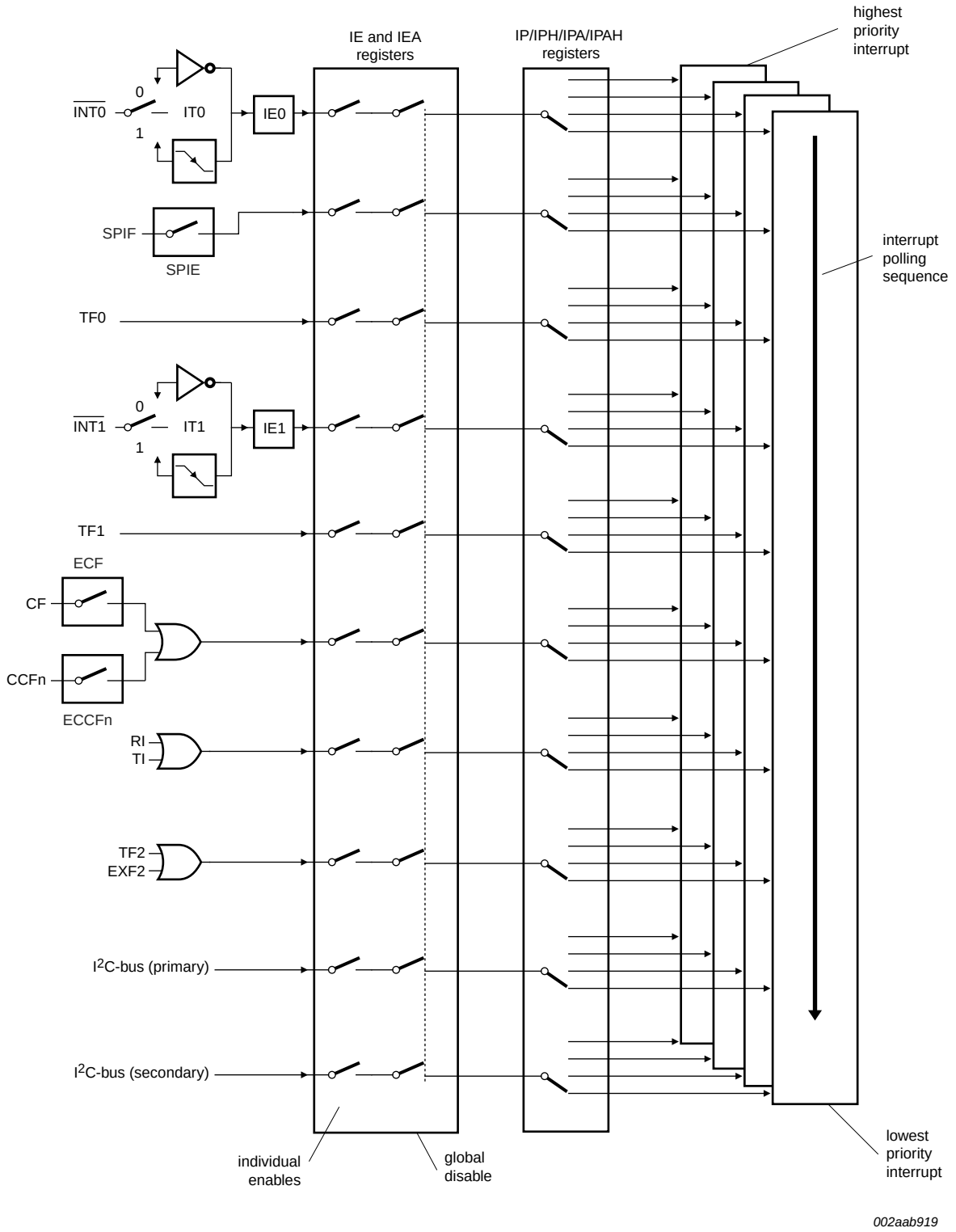


Fig 32. Interrupt structure

6.14 System clock and clock options

6.14.1 Clock input options and recommended capacitor values for the oscillator

Shown in Figure 33 and Figure 34 are the input and output of an internal inverting amplifier (XTAL1, XTAL2), which can be configured for use as an on-chip oscillator.

When driving the device from an external clock source, XTAL2 should be left disconnected and XTAL1 should be driven.

At start-up, the external oscillator may encounter a higher capacitive load at XTAL1 due to interaction between the amplifier and its feedback capacitance. However, the capacitance will not exceed 15 pF once the external signal meets the V_{IL} and V_{IH} specifications.

Resonator manufacturer, supply voltage, and other factors may cause circuit performance to differ from one application to another. C_1 and C_2 should be adjusted appropriately for each design. Table 68 shows the typical values for C_1 and C_2 vs. resonator type for various frequencies

Table 68. Recommended values for C_1 and C_2 by crystal type

Resonator	$C_1 = C_2$
Quartz	20 pF to 30 pF
Ceramic	40 pF to 50 pF

6.14.2 Clock doubling option

By default, the device runs at six clocks per machine cycle. The device may be run in 12 clocks per machine cycle mode by flash programming of the 6x/12x bit.

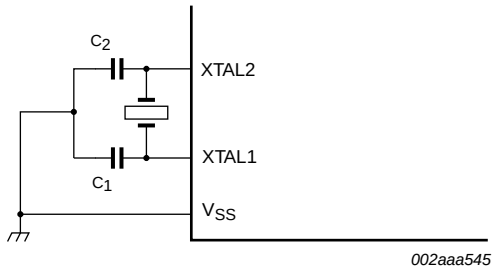


Fig 33. Oscillator characteristics (using the on-chip oscillator)

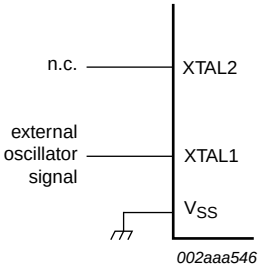


Fig 34. Oscillator characteristics (external clock drive)

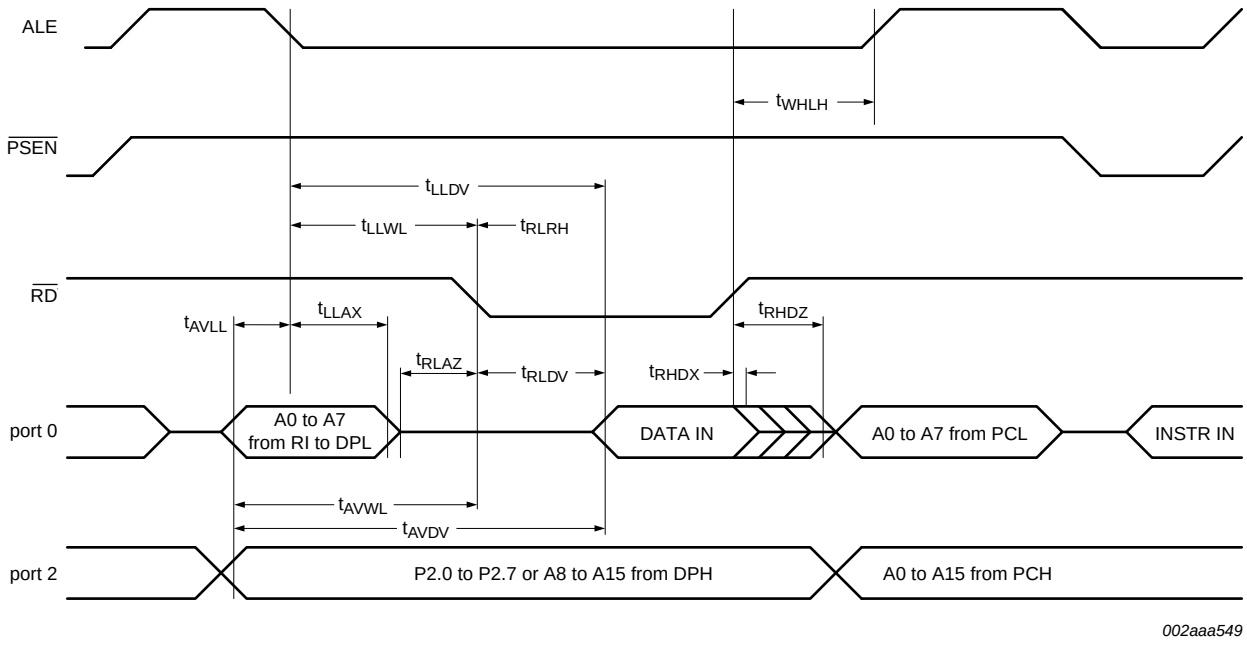


Fig 37. External data memory read cycle

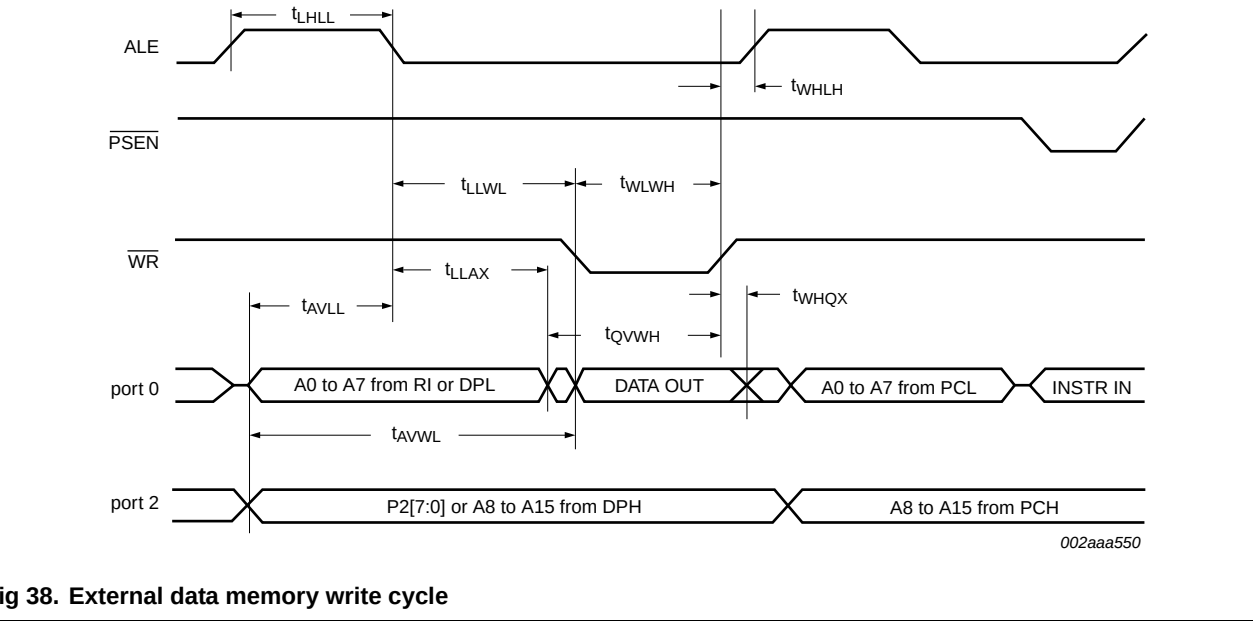


Fig 38. External data memory write cycle

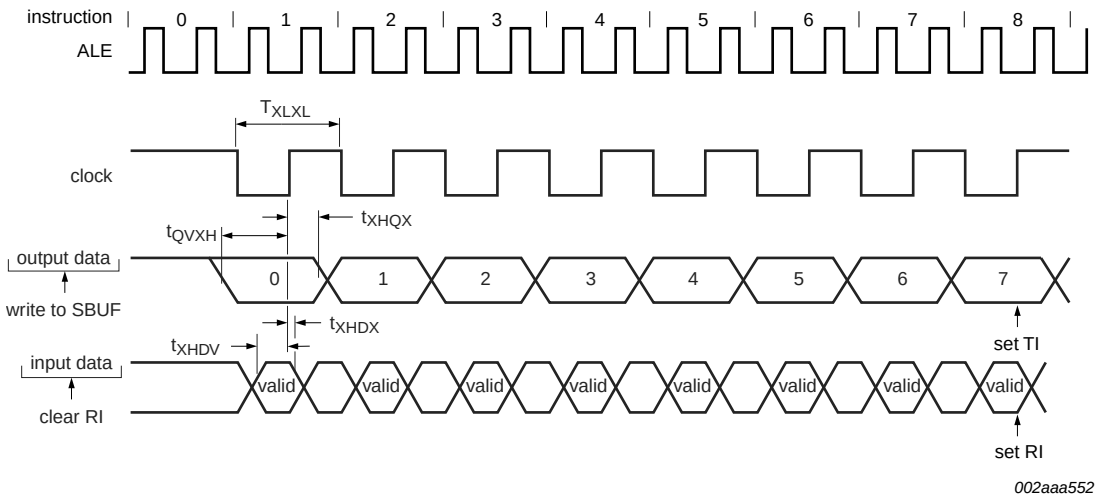


Fig 40. Shift register mode timing waveforms

Table 74. I²C-bus interface timing (12-clock mode)

Symbol	Parameter	Conditions	Input	Output	Unit
$t_{HD;STA}$	hold time (repeated) START condition		$\geq 14T_{cy(clk)}$	> 4.0[1]	µs
t_{LOW}	LOW period of the SCL clock		$\geq 16T_{cy(clk)}$	> 4.7[1]	µs
t_{HIGH}	HIGH period of the USCL clock		$\geq 14T_{cy(clk)}$	> 4.0[1]	µs
$t_{r(SCL)}$	SCL rise time		≤ 1	-[2]	µs
$t_{f(SCL)}$	SCL fall time		≤ 0.3	≤ 0.3[3]	µs
$t_{SU;DAT}$	data set-up time		≥ 250	$20T_{cy(clk)} - t_{r(SDA)}$	ns
t_{suDAT1}	data set-up time 1	before repeated START	≥ 250	> 1000[1]	ns
t_{suDAT2}	data set-up time 2	before STOP condition	≥ 250	$> 8T_{cy(clk)}$	ns
$t_{HD;DAT}$	data hold time		≥ 0	$> 8T_{cy(clk)} - t_{f(SCL)}$	ns
$t_{SU;STA}$	set-up time for a repeated START condition		$\geq 14T_{cy(clk)}$[1]	> 4.7[1]	µs
$t_{SU;STO}$	set-up time for STOP condition		$\geq 14T_{cy(clk)}$[1]	> 4.0[1]	µs
t_{BUF}	bus free time between a STOP and START condition		$\geq 14T_{cy(clk)}$[1]	> 4.7[1]	µs
$t_{r(SDA)}$	SDA rise time		≤ 0.3	≤ 0.3	µs
$t_{f(SDA)}$	SDA fall time		≤ 0.3	≤ 0.3[3]	µs

[1] At 100 kb/s. All other bit rates, this value is inversely proportional to the bit rate of 100 kb/s.

[2] Determined by the external bus capacitance and pull-up resistor. This must be < 1 µs.

[3] Spikes on SDA and SCL with a duration less than $3T_{cy(clk)}$ will be filtered out. Max capacitance on SDA and SCL = 400 pF.

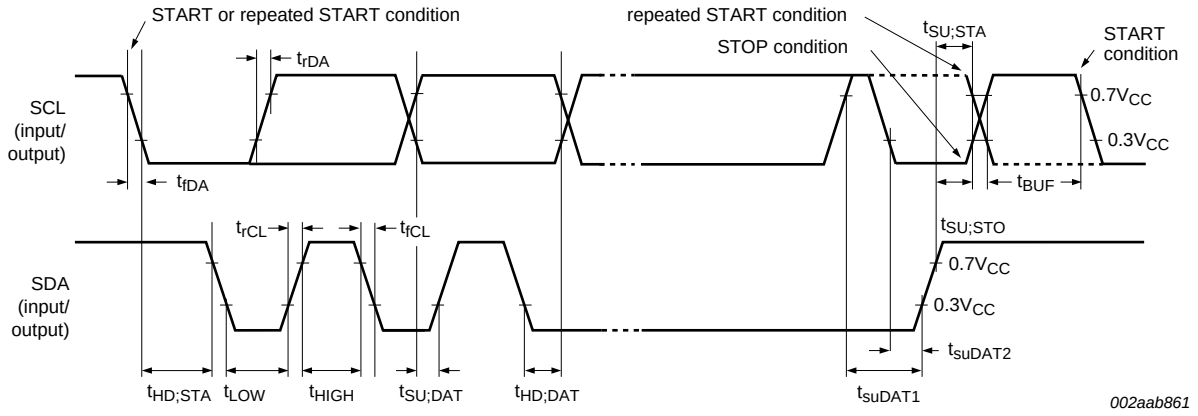


Fig 41. I²C-bus interface timing

Table 75. SPI interface timing

Symbol	Parameter	Conditions	Variable clock		f_{osc} = 18 MHz		Unit
			Min	Max	Min	Max	
f_{SPI}	SPI operating frequency		0	$T_{cy(clk)}$ / 4	0	10	MHz
T_{SPICYC}	SPI cycle time	see Figure 42, 43, 44, 45	$4T_{cy(clk)}$	-	222	-	ns
$t_{SPILEAD}$	SPI enable lead time	see Figure 44, 45	250	-	250	-	ns
t_{SPILAG}	SPI enable lag time	see Figure 44, 45	250	-	250	-	ns
$t_{SPICLKH}$	SPICLK HIGH time	see Figure 42, 43, 44, 45	$2T_{cy(clk)}$	-	111	-	ns
$t_{SPICLKL}$	SPICLK LOW time	see Figure 42, 43, 44, 45	$2T_{cy(clk)}$	-	111	-	ns
t_{SPIDSU}	SPI data set-up time	master or slave; see Figure 42, 43, 44, 45	100	-	100	-	ns
t_{SPIDH}	SPI data hold time	master or slave; see Figure 42, 43, 44, 45	100	-	100	-	ns
t_{SPIA}	SPI access time	see Figure 44, 45	0	80	0	80	ns
t_{SPIDIS}	SPI disable time	see Figure 44, 45	0	160	-	160	ns
t_{SPIDV}	SPI enable to output data valid time	see Figure 42, 43, 44, 45	-	111	-	111	ns
t_{SPIOH}	SPI output data hold time	see Figure 42, 43, 44, 45	0	-	0	-	ns
t_{SPIR}	SPI rise time	see Figure 42, 43, 44, 45					
	SPI outputs (SPICLK, MOSI, MISO)		-	100	-	100	ns
	SPI inputs (SPICLK, MOSI, MISO, $\overline{SS}$)		-	2000	-	2000	ns
t_{SPIF}	SPI fall time	see Figure 42, 43, 44, 45					
	SPI outputs (SPICLK, MOSI, MISO)		-	100	-	100	ns
	SPI inputs (SPICLK, MOSI, MISO, $\overline{SS}$)		-	2000	-	2000	ns

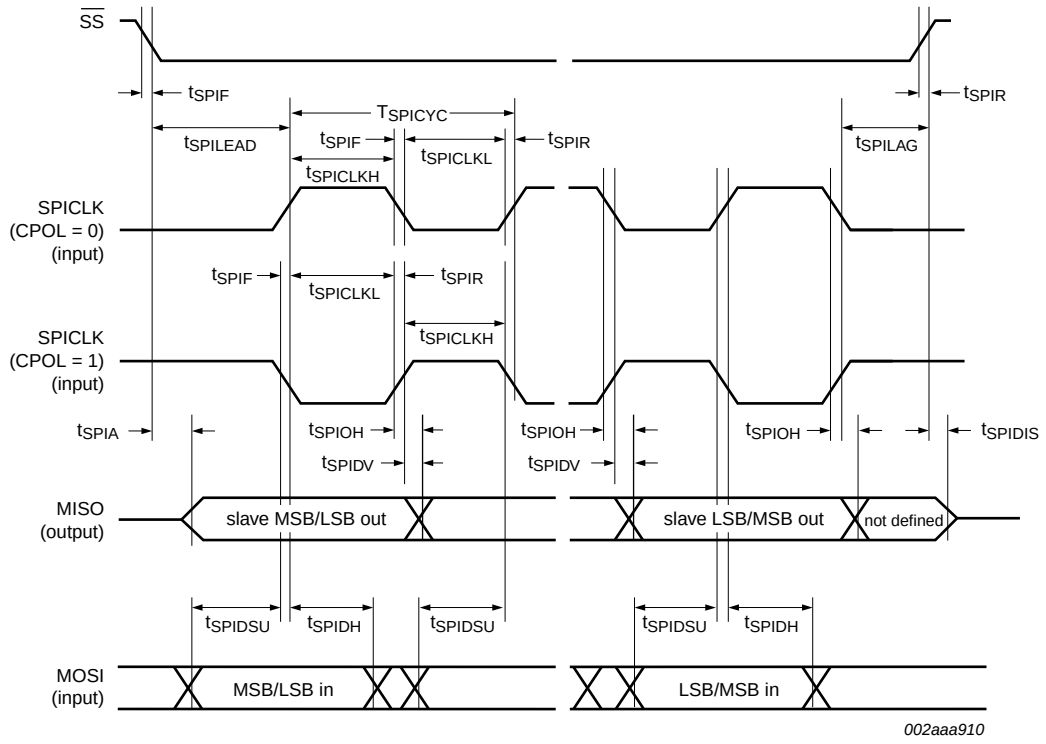


Fig 44. SPI slave timing (CPHA = 0)

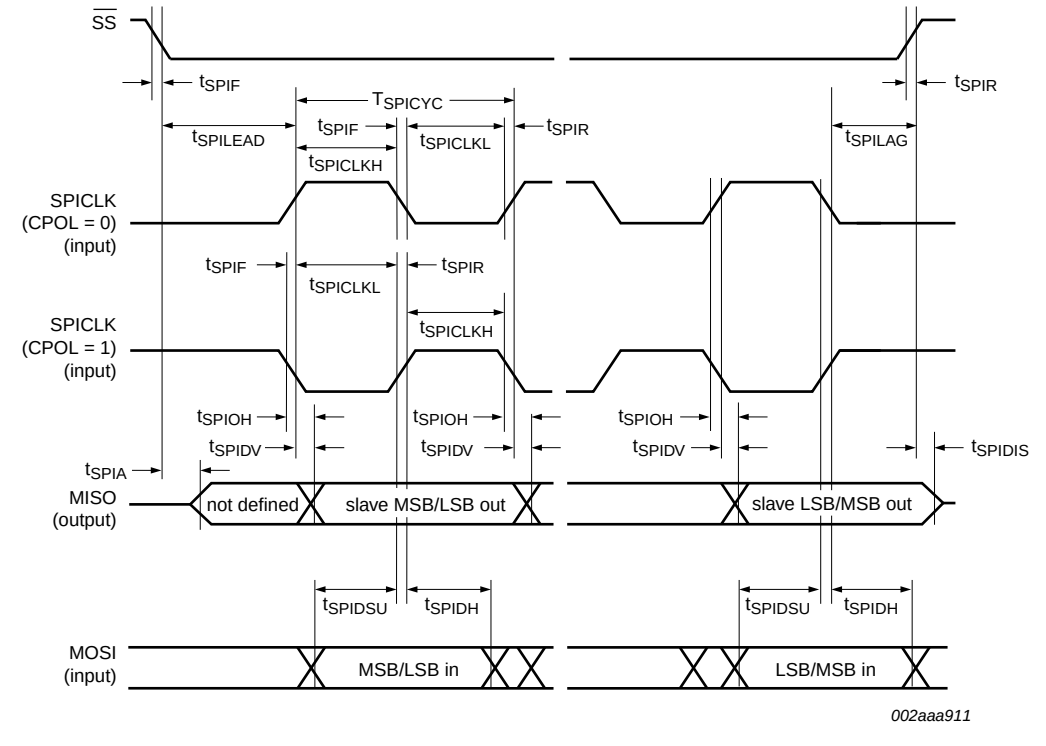
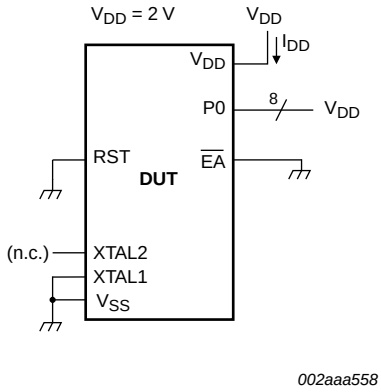


Fig 45. SPI slave timing (CPHA = 1)



All other pins disconnected

Fig 49. I_{DD} test condition, Power-down mode

11. Abbreviations

Table 76. Acronym list

Acronym	Description
ALE	Address Latch Enabled
CPU	Central Processing Unit
DUT	Device Under Test
EPROM	Erasable Programmable Read-Only Memory
EMI	Electro-Magnetic Interference
MCU	Microcontroller Unit
PWM	Pulse Width Modulator
RAM	Random Access Memory
RC	Resistance-Capacitance
SFR	Special Function Register
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter