



Welcome to [E-XFL.COM](#)

What is "[Embedded - Microcontrollers](#)"?

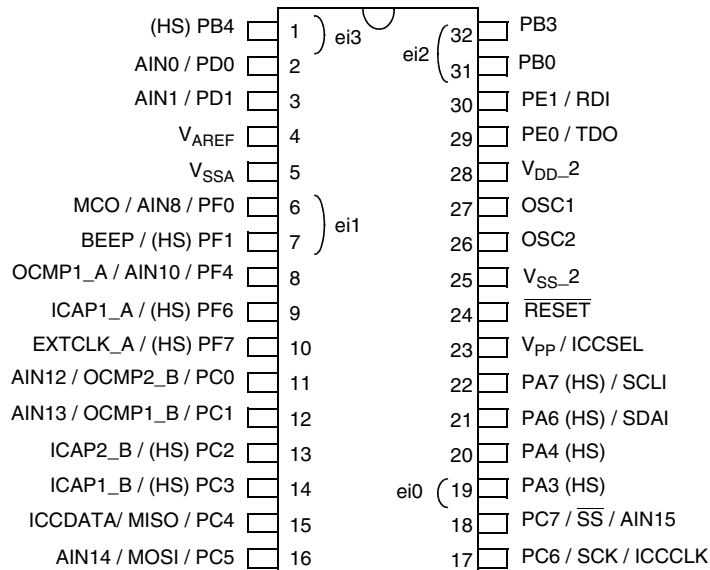
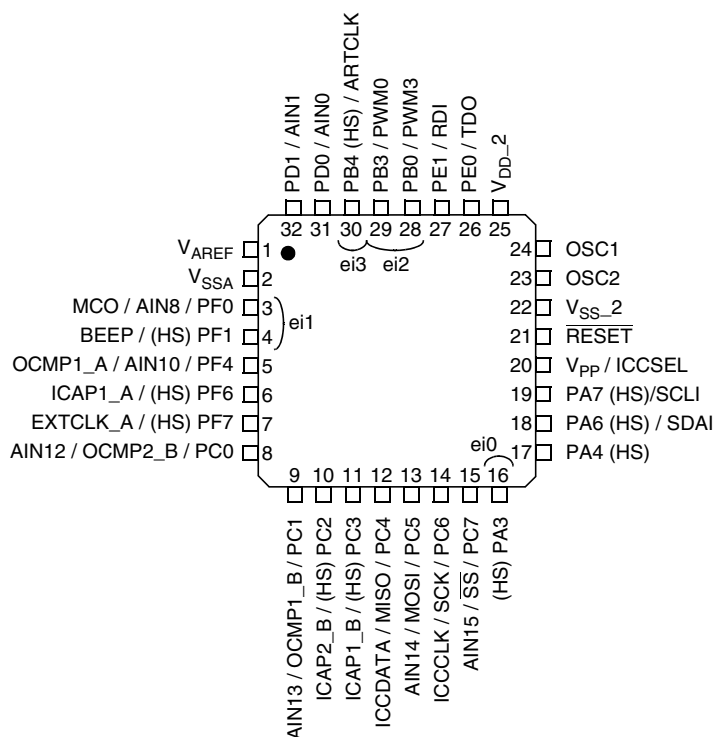
"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	ST7
Core Size	8-Bit
Speed	8MHz
Connectivity	I ² C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	3.8V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f325k6tc

Figure 5. 32-Pin LQFP/DIP Package Pinouts



(HS) 20mA high sink capability
 eiX associated external interrupt vector

PIN DESCRIPTION (Cont'd)

For external pin connection guidelines, refer to See "ELECTRICAL CHARACTERISTICS" on page 142.

Legend / Abbreviations for Table 2 and Table 3:

Type: I = input, O = output, S = supply

Input level: A = Dedicated analog input

In/Output level: C = CMOS $0.3V_{DD}/0.7V_{DD}$

C_T = CMOS $0.3V_{DD}/0.7V_{DD}$ with input trigger

Output level: HS = 20mA high sink (on N-buffer only)

Port and control configuration:

- Input: float = floating, wpu = weak pull-up, int = interrupt ¹⁾, ana = analog
- Output: OD = open drain ²⁾, PP = push-pull

Refer to "I/O PORTS" on page 50 for more details on the software configuration of the I/O ports.

The RESET configuration of each pin is shown in bold. This configuration is valid as long as the device is in reset state.

= Pin not connected in ST72325S devices

Table 2. LQFP64/48/44 and SDIP42 Device Pin Descriptions

Pin n°					Pin Name	Type	Level		Port						Main function (after reset)	Alternate function
LQFP64	LQFP48C	LQFP48S	LQFP44	SDIP42			Input	Output	Input				Output			
									float	wpu	int	ana	OD	PP		
1	2	-	-	-	PE4 (HS)	I/O	C _T	HS	X	X			X	X	Port E4	
2	⁻⁴⁾	-	-	-	PE5 (HS)	I/O	C _T	HS	X	X			X	X	Port E5	
3	⁻⁴⁾	-	-	-	PE6 (HS)	I/O	C _T	HS	X	X			X	X	Port E6	
4	⁻⁴⁾	-	-	-	PE7 (HS)	I/O	C _T	HS	X	X			X	X	Port E7	
5	3	3	2	39	PB0/PWM3	I/O	C _T		X	ei2			X	X	Port B0	PWM Output 3
																Caution: Negative current injection not allowed on this pin
6	4	4	3	40	PB1/PWM2	I/O	C _T		X	ei2			X	X	Port B1	PWM Output 2
7	5	5	4	41	PB2/PWM1	I/O	C _T		X	ei2			X	X	Port B2	PWM Output 1
8	6	6	5	42	PB3/PWM0	I/O	C _T		X		ei2		X	X	Port B3	PWM Output 0
9	7	7	6	1	PB4 (HS)/ARTCLK	I/O	C _T	HS	X	ei3			X	X	Port B4	PWM-ART External Clock
10	8	-	-	-	PB5 / ARTIC1	I/O	C _T		X	ei3			X	X	Port B5	PWM-ART Input Capture 1
11	⁻⁴⁾	-	-	-	PB6 / ARTIC2	I/O	C _T		X	ei3			X	X	Port B6	PWM-ART Input Capture 2
12	⁻⁴⁾	-	-	-	PB7	I/O	C _T		X		ei3		X	X	Port B7	
13	9	9	7	2	PD0/AIN0	I/O	C _T		X	X		X	X	X	Port D0	ADC Analog Input 0
14	19	10	8	3	PD1/AIN1	I/O	C _T		X	X		X	X	X	Port D1	ADC Analog Input 1
15	11	11	9	4	PD2/AIN2	I/O	C _T		X	X		X	X	X	Port D2	ADC Analog Input 2
16	12	12	10	5	PD3/AIN3	I/O	C _T		X	X		X	X	X	Port D3	ADC Analog Input 3

INTERRUPTS (Cont'd)

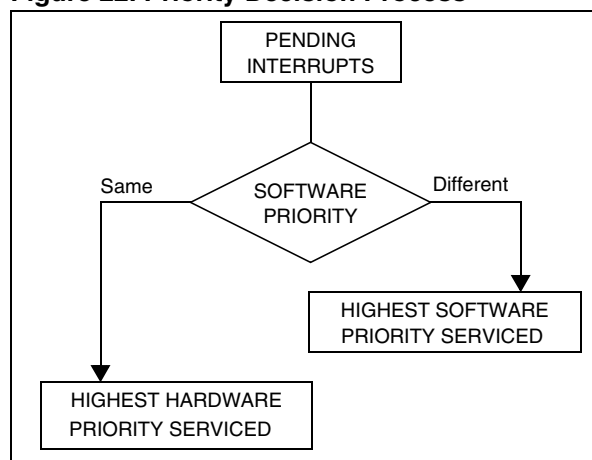
Servicing Pending Interrupts

As several interrupts can be pending at the same time, the interrupt to be taken into account is determined by the following two-step process:

- the highest software priority interrupt is serviced,
- if several interrupts have the same software priority then the interrupt with the highest hardware priority is serviced first.

Figure 22 describes this decision process.

Figure 22. Priority Decision Process



When an interrupt request is not serviced immediately, it is latched and then processed when its software priority combined with the hardware priority becomes the highest one.

Note 1: The hardware priority is exclusive while the software one is not. This allows the previous process to succeed with only one interrupt.

Note 2: TLI, RESET and TRAP can be considered as having the highest software priority in the decision process.

Different Interrupt Vector Sources

Two interrupt source types are managed by the ST7 interrupt controller: the non-maskable type (RESET, TRAP) and the maskable type (external or from internal peripherals).

Non-Maskable Sources

These sources are processed regardless of the state of the I1 and I0 bits of the CC register (see Figure 21). After stacking the PC, X, A and CC registers (except for RESET), the corresponding vector is loaded in the PC register and the I1 and I0 bits of the CC are set to disable interrupts (level 3). These sources allow the processor to exit HALT mode.

■ TRAP (Non Maskable Software Interrupt)

This software interrupt is serviced when the TRAP instruction is executed. It will be serviced according to the flowchart in Figure 21.

Caution: TRAP can be interrupted by a TLI.

■ RESET

The RESET source has the highest priority in the ST7. This means that the first current routine has the highest software priority (level 3) and the highest hardware priority.

See the RESET chapter for more details.

Maskable Sources

Maskable interrupt vector sources can be serviced if the corresponding interrupt is enabled and if its own interrupt software priority (in ISPRx registers) is higher than the one currently being serviced (I1 and I0 in CC register). If any of these two conditions is false, the interrupt is latched and thus remains pending.

■ TLI (Top Level Hardware Interrupt)

This hardware interrupt occurs when a specific edge is detected on the dedicated TLI pin. It will be serviced according to the flowchart in Figure 21 as a trap.

Caution: A TRAP instruction must not be used in a TLI service routine.

■ External Interrupts

External interrupts allow the processor to exit from HALT low power mode. External interrupt sensitivity is software selectable through the External Interrupt Control register (EICR).

External interrupt triggered on edge will be latched and the interrupt request automatically cleared upon entering the interrupt service routine.

If several input pins of a group connected to the same interrupt line are selected simultaneously, these will be logically ORed.

■ Peripheral Interrupts

Usually the peripheral interrupts cause the MCU to exit from HALT mode except those mentioned in the "Interrupt Mapping" table. A peripheral interrupt occurs when a specific flag is set in the peripheral status registers and if the corresponding enable bit is set in the peripheral control register. The general sequence for clearing an interrupt is based on an access to the status register followed by a read or write to an associated register.

Note: The clearing sequence resets the internal latch. A pending interrupt (i.e. waiting for being serviced) will therefore be lost if the clear sequence is executed.

16-BIT TIMER (Cont'd)

10.4.3.4 Output Compare

In this section, the index, i , may be 1 or 2 because there are two output compare functions in the 16-bit timer.

This function can be used to control an output waveform or indicate when a period of time has elapsed.

When a match is found between the Output Compare register and the free running counter, the output compare function:

- Assigns pins with a programmable value if the OC/E bit is set
- Sets a flag in the status register
- Generates an interrupt if enabled

Two 16-bit registers Output Compare Register 1 (OC1R) and Output Compare Register 2 (OC2R) contain the value to be compared to the counter register each timer clock cycle.

	MS Byte	LS Byte
OC/R	OC/HR	OC/LR

These registers are readable and writable and are not affected by the timer hardware. A reset event changes the OC/R value to 8000h.

Timing resolution is one count of the free running counter: $(f_{\text{CPU}}/CC[1:0])$.

Procedure:

To use the output compare function, select the following in the CR2 register:

- Set the OC/E bit if an output is needed then the OCMP/ i pin is dedicated to the output compare i signal.
- Select the timer clock (CC[1:0]) (see Table 1).

And select the following in the CR1 register:

- Select the OLVL/ i bit to applied to the OCMP/ i pins after the match occurs.
- Set the OCIE bit to generate an interrupt if it is needed.

When a match is found between OC/R register and CR register:

- OCF/ i bit is set.

- The OCMP/ i pin takes OLVL/ i bit value (OCMP/ i pin latch is forced low during reset).
- A timer interrupt is generated if the OCIE bit is set in the CR1 register and the I bit is cleared in the CC register (CC).

The OC/R register value required for a specific timing application can be calculated using the following formula:

$$\Delta \text{OC/R} = \frac{\Delta t * f_{\text{CPU}}}{\text{PRESC}}$$

Where:

Δt = Output compare period (in seconds)

f_{CPU} = CPU clock frequency (in hertz)

PRESC = Timer prescaler factor (2, 4 or 8 depending on CC[1:0] bits, see Table 1)

If the timer clock is an external clock, the formula is:

$$\Delta \text{OC/R} = \Delta t * f_{\text{EXT}}$$

Where:

Δt = Output compare period (in seconds)

f_{EXT} = External timer clock frequency (in hertz)

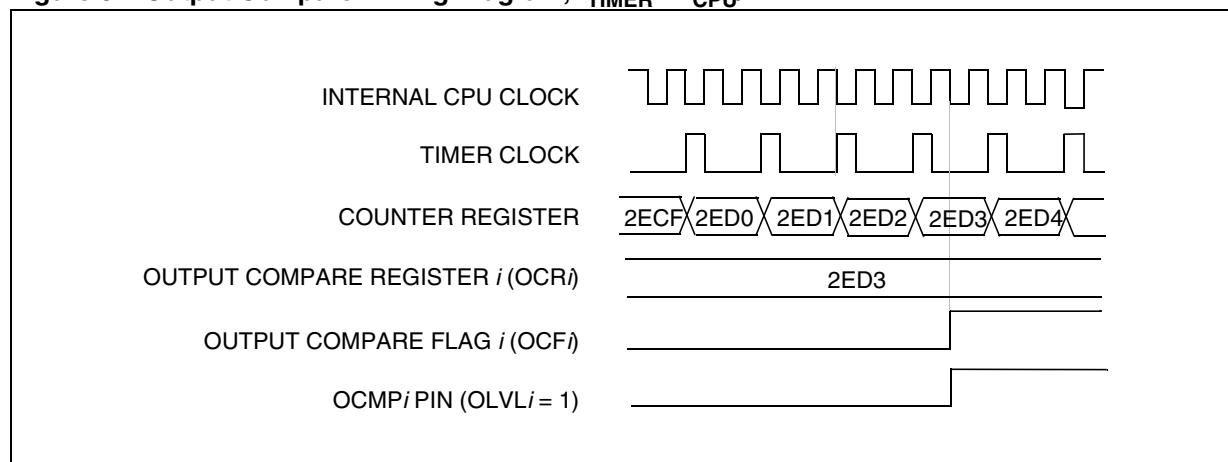
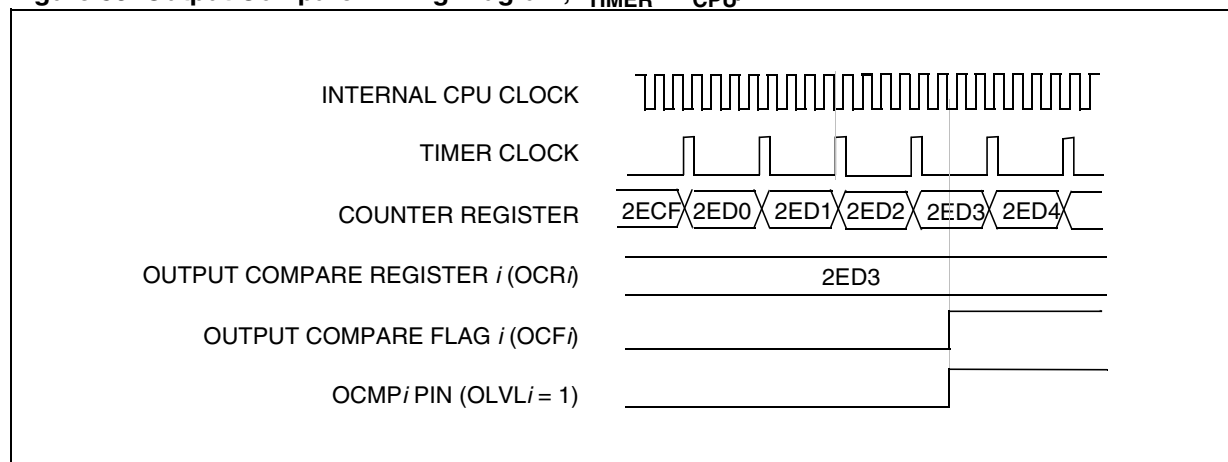
Clearing the output compare interrupt request (that is, clearing the OCF/ i bit) is done by:

1. Reading the SR register while the OCF/ i bit is set.
2. An access (read or write) to the OC/LR register.

The following procedure is recommended to prevent the OCF/ i bit from being set between the time it is read and the write to the OC/R register:

- Write to the OC/HR register (further compares are inhibited).
- Read the SR register (first step of the clearance of the OCF/ i bit, which may be already set).
- Write to the OC/LR register (enables the output compare function and clears the OCF/ i bit).

16-BIT TIMER (Cont'd)

Figure 52. Output Compare Timing Diagram, $f_{\text{TIMER}} = f_{\text{CPU}}/2$ Figure 53. Output Compare Timing Diagram, $f_{\text{TIMER}} = f_{\text{CPU}}/4$ 

16-BIT TIMER (Cont'd)**10.4.3.5 One Pulse Mode**

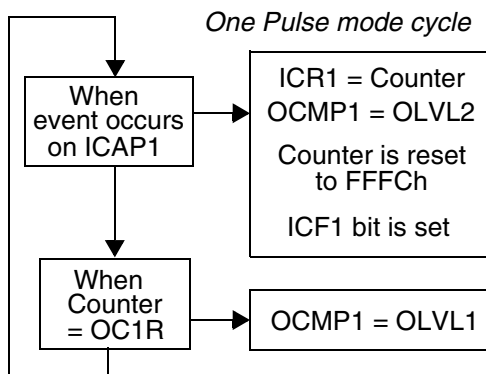
One Pulse mode enables the generation of a pulse when an external event occurs. This mode is selected via the OPM bit in the CR2 register.

The One Pulse mode uses the Input Capture1 function and the Output Compare1 function.

Procedure:

To use One Pulse mode:

1. Load the OC1R register with the value corresponding to the length of the pulse (see the formula in the opposite column).
2. Select the following in the CR1 register:
 - Using the OLVL1 bit, select the level to be applied to the OCMP1 pin after the pulse.
 - Using the OLVL2 bit, select the level to be applied to the OCMP1 pin during the pulse.
 - Select the edge of the active transition on the ICAP1 pin with the IEDG1 bit (the ICAP1 pin must be configured as floating input).
3. Select the following in the CR2 register:
 - Set the OC1E bit, the OCMP1 pin is then dedicated to the Output Compare 1 function.
 - Set the OPM bit.
 - Select the timer clock CC[1:0] (see Table 1).



Then, on a valid event on the ICAP1 pin, the counter is initialized to FFFCh and OLVL2 bit is loaded on the OCMP1 pin, the ICF1 bit is set and the value FFFDh is loaded in the IC1R register.

Because the ICF1 bit is set when an active edge occurs, an interrupt can be generated if the ICIE bit is set.

Clearing the Input Capture interrupt request (that is, clearing the ICF1 bit) is done in two steps:

1. Reading the SR register while the ICF1 bit is set.
2. An access (read or write) to the IC1LR register.

The OC1R register value required for a specific timing application can be calculated using the following formula:

$$\text{OC1R Value} = \frac{t \cdot f_{\text{CPU}}}{\text{PRESC}} - 5$$

Where:

t = Pulse period (in seconds)

f_{CPU} = CPU clock frequency (in hertz)

PRESC = Timer prescaler factor (2, 4 or 8 depending on the CC[1:0] bits, see Table 1)

If the timer clock is an external clock the formula is:

$$\text{OC1R} = t \cdot f_{\text{EXT}} - 5$$

Where:

t = Pulse period (in seconds)

f_{EXT} = External timer clock frequency (in hertz)

When the value of the counter is equal to the value of the contents of the OC1R register, the OLVL1 bit is output on the OCMP1 pin, (See Figure 10).

Notes:

1. The OCF1 bit cannot be set by hardware in One Pulse mode but the OCF2 bit can generate an Output Compare interrupt.
2. When the Pulse Width Modulation (PWM) and One Pulse mode (OPM) bits are both set, the PWM mode is the only active one.
3. If OLVL1 = OLVL2 a continuous signal will be seen on the OCMP1 pin.
4. The ICAP1 pin can not be used to perform input capture. The ICAP2 pin can be used to perform input capture (ICF2 can be set and IC2R can be loaded) but the user must take care that the counter is reset each time a valid edge occurs on the ICAP1 pin and ICF1 can also generates interrupt if ICIE is set.
5. When One Pulse mode is used OC1R is dedicated to this mode. Nevertheless OC2R and OCF2 can be used to indicate a period of time has been elapsed but cannot generate an output waveform because the level OLVL2 is dedicated to the One Pulse mode.

16-BIT TIMER (Cont'd)

Figure 54. One Pulse Mode Timing Example

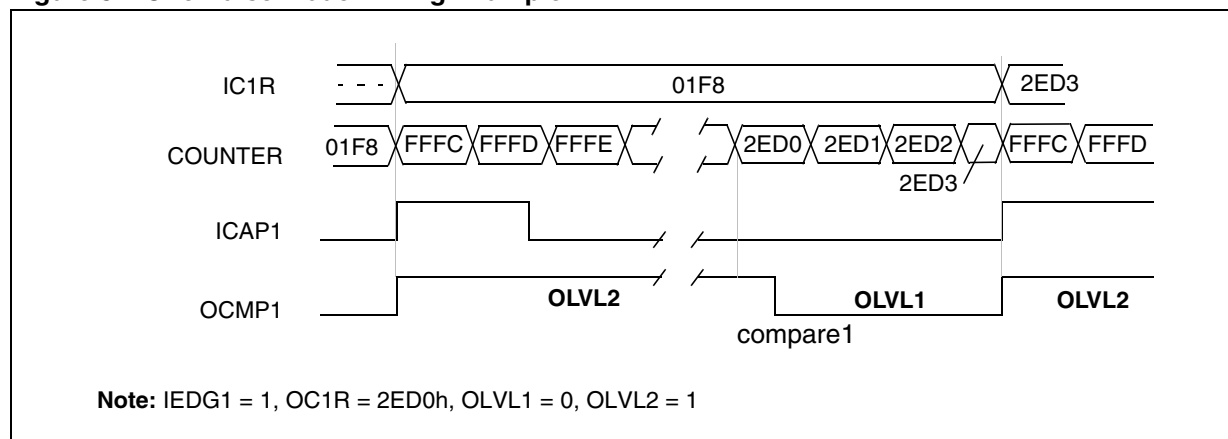
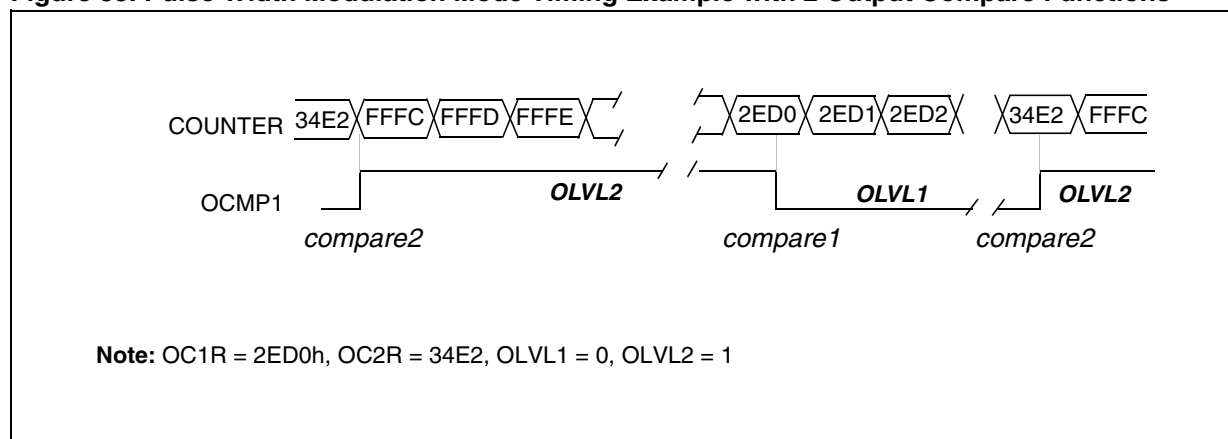


Figure 55. Pulse Width Modulation Mode Timing Example with 2 Output Compare Functions



Note: On timers with only one Output Compare register, a fixed frequency PWM signal can be generated using the output compare and the counter overflow to define the pulse length.

16-BIT TIMER (Cont'd)**CONTROL REGISTER 2 (CR2)**

Read/Write

Reset Value: 0000 0000 (00h)

7							0
OC1E	OC2E	OPM	PWM	CC1	CC0	IEDG2	EXEDG

Bit 7 = OC1E Output Compare 1 Pin Enable.

This bit is used only to output the signal from the timer on the OCMP1 pin (OLV1 in Output Compare mode, both OLV1 and OLV2 in PWM and one-pulse mode). Whatever the value of the OC1E bit, the Output Compare 1 function of the timer remains active.

0: OCMP1 pin alternate function disabled (I/O pin free for general-purpose I/O).

1: OCMP1 pin alternate function enabled.

Bit 6 = OC2E Output Compare 2 Pin Enable.

This bit is used only to output the signal from the timer on the OCMP2 pin (OLV2 in Output Compare mode). Whatever the value of the OC2E bit, the Output Compare 2 function of the timer remains active.

0: OCMP2 pin alternate function disabled (I/O pin free for general-purpose I/O).

1: OCMP2 pin alternate function enabled.

Bit 5 = OPM One Pulse Mode.

0: One Pulse mode is not active.

1: One Pulse mode is active, the ICAP1 pin can be used to trigger one pulse on the OCMP1 pin; the active transition is given by the IEDG1 bit. The length of the generated pulse depends on the contents of the OC1R register.

Bit 4 = PWM Pulse Width Modulation.

0: PWM mode is not active.

1: PWM mode is active, the OCMP1 pin outputs a programmable cyclic signal; the length of the pulse depends on the value of OC1R register; the period depends on the value of OC2R register.

Bit 3, 2 = CC[1:0] Clock Control.

The timer clock mode depends on these bits:

Table 18. Clock Control Bits

Timer Clock	CC1	CC0
$f_{CPU} / 4$	0	0
$f_{CPU} / 2$		1
$f_{CPU} / 8$	1	0
External Clock (where available)		1

Note: If the external clock pin is not available, programming the external clock configuration stops the counter.

Bit 1 = IEDG2 Input Edge 2.

This bit determines which type of level transition on the ICAP2 pin will trigger the capture.

0: A falling edge triggers the capture.

1: A rising edge triggers the capture.

Bit 0 = EXEDG External Clock Edge.

This bit determines which type of level transition on the external clock pin EXTCLK will trigger the counter register.

0: A falling edge triggers the counter register.

1: A rising edge triggers the counter register.

10.5 SERIAL PERIPHERAL INTERFACE (SPI)

10.5.1 Introduction

The Serial Peripheral Interface (SPI) allows full-duplex, synchronous, serial communication with external devices. An SPI system may consist of a master and one or more slaves however the SPI interface can not be a master in a multi-master system.

10.5.2 Main Features

- Full duplex synchronous transfers (on 3 lines)
- Simplex synchronous transfers (on 2 lines)
- Master or slave operation
- Six master mode frequencies ($f_{CPU}/4$ max.)
- $f_{CPU}/2$ max. slave mode frequency (see note)
- \overline{SS} Management by software or hardware
- Programmable clock polarity and phase
- End of transfer interrupt flag
- Write collision, Master Mode Fault and Overrun flags

Note: In slave mode, continuous transmission is not possible at maximum frequency due to the software overhead for clearing status flags and to initiate the next transmission sequence.

10.5.3 General Description

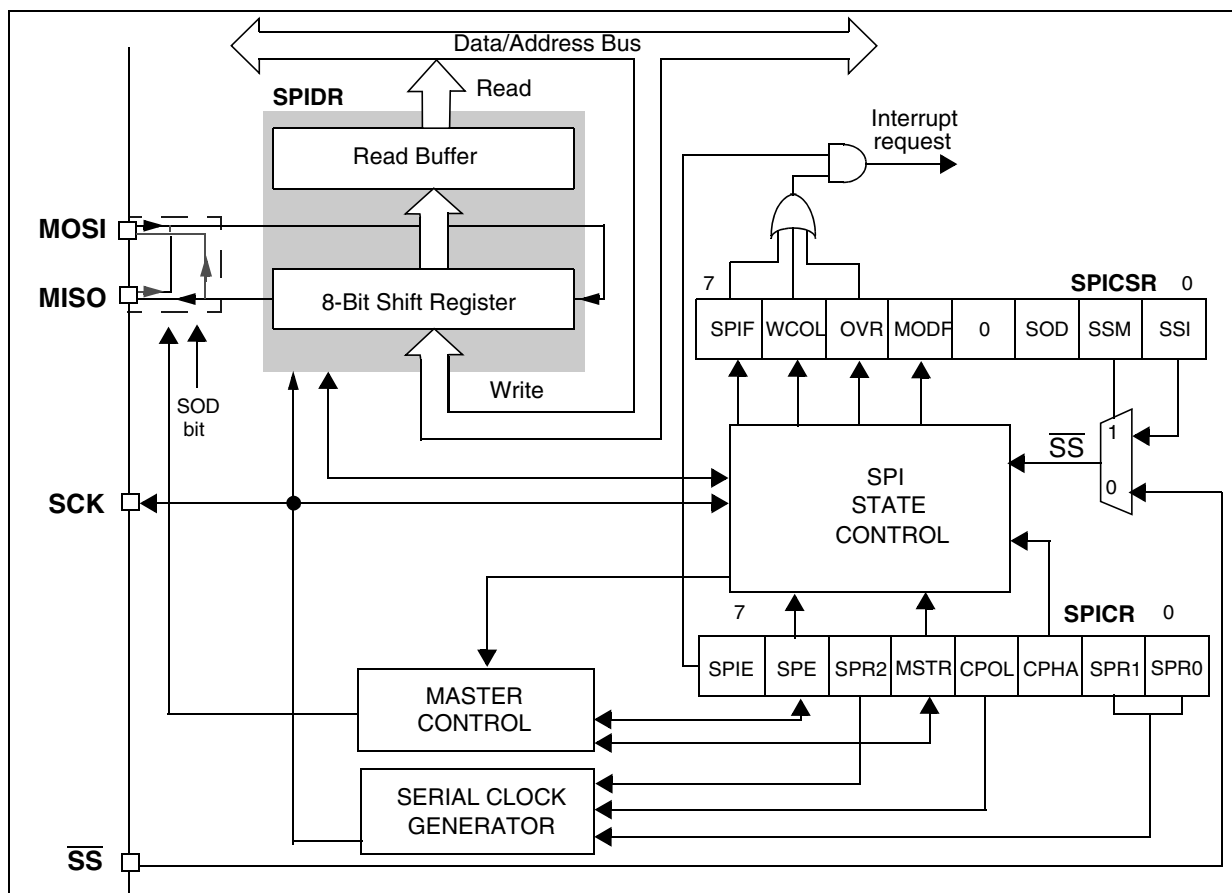
Figure 56 shows the serial peripheral interface (SPI) block diagram. There are 3 registers:

- SPI Control Register (SPICR)
- SPI Control/Status Register (SPICSR)
- SPI Data Register (SPIDR)

The SPI is connected to external devices through 4 pins:

- MISO: Master In / Slave Out data
- MOSI: Master Out / Slave In data
- SCK: Serial Clock out by SPI masters and input by SPI slaves

Figure 56. Serial Peripheral Interface Block Diagram



SERIAL PERIPHERAL INTERFACE (Cont'd)– \overline{SS} : Slave select:

This input signal acts as a 'chip select' to let the SPI master communicate with slaves individually and to avoid contention on the data lines. Slave \overline{SS} inputs can be driven by standard I/O ports on the master MCU.

10.5.3.1 Functional Description

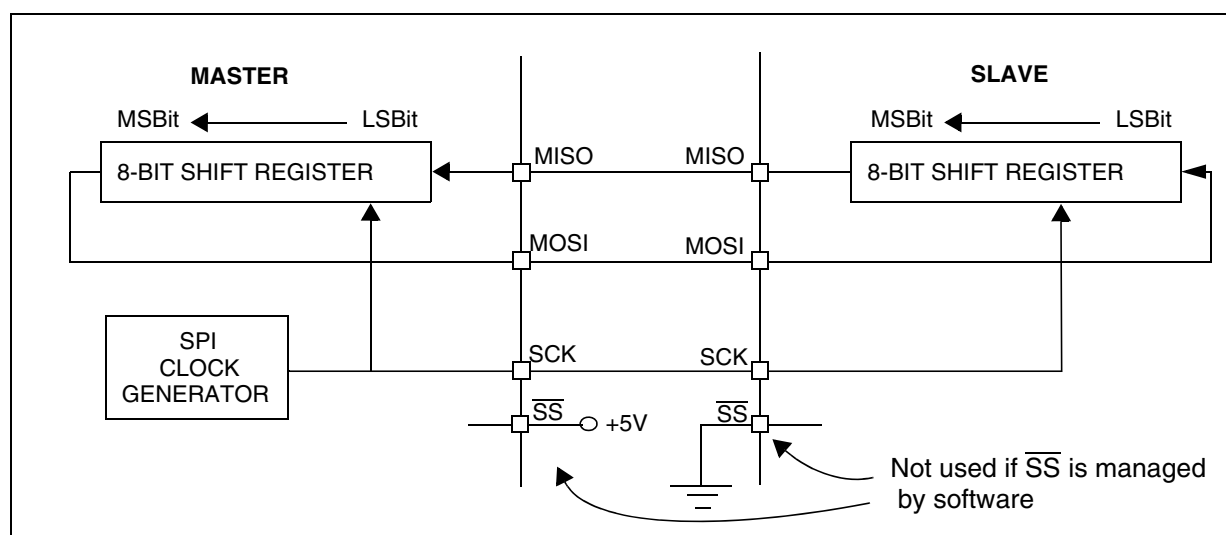
A basic example of interconnections between a single master and a single slave is illustrated in Figure 57.

The MOSI pins are connected together and the MISO pins are connected together. In this way data is transferred serially between master and slave (most significant bit first).

The communication is always initiated by the master. When the master device transmits data to a slave device via MOSI pin, the slave device responds by sending data to the master device via the MISO pin. This implies full duplex communication with both data out and data in synchronized with the same clock signal (which is provided by the master device via the SCK pin).

To use a single data line, the MISO and MOSI pins must be connected at each node (in this case only simplex communication is possible).

Four possible data/clock timing relationships may be chosen (see Figure 60) but master and slave must be programmed with the same timing mode.

Figure 57. Single Master/ Single Slave Application

SERIAL PERIPHERAL INTERFACE (Cont'd)**10.5.8 Register Description****CONTROL REGISTER (SPICR)**

Read/Write

Reset Value: 0000 xxxx (0xh)

7							0
SPIE	SPE	SPR2	MSTR	CPOL	CPHA	SPR1	SPR0

Bit 7 = **SPIE** *Serial Peripheral Interrupt Enable*.
This bit is set and cleared by software.

0: Interrupt is inhibited

1: An SPI interrupt is generated whenever
SPIF=1, MODF=1 or OVR=1 in the SPICSR
register

Bit 6 = **SPE** *Serial Peripheral Output Enable*.
This bit is set and cleared by software. It is also
cleared by hardware when, in master mode, $\overline{SS}=0$
(see Section 10.5.5.1 Master Mode Fault
(MODF)). The SPE bit is cleared by reset, so the
SPI peripheral is not initially connected to the ex-
ternal pins.

0: I/O pins free for general purpose I/O

1: SPI I/O pin alternate functions enabled

Bit 5 = **SPR2** *Divider Enable*.

This bit is set and cleared by software and is
cleared by reset. It is used with the SPR[1:0] bits to
set the baud rate. Refer to Table 20 SPI Master
mode SCK Frequency.

0: Divider by 2 enabled

1: Divider by 2 disabled

Note: This bit has no effect in slave mode.

Bit 4 = **MSTR** *Master Mode*.

This bit is set and cleared by software. It is also
cleared by hardware when, in master mode, $\overline{SS}=0$
(see Section 10.5.5.1 Master Mode Fault
(MODF)).

0: Slave mode

1: Master mode. The function of the SCK pin
changes from an input to an output and the func-
tions of the MISO and MOSI pins are reversed.

Bit 3 = **CPOL** *Clock Polarity*.

This bit is set and cleared by software. This bit de-
termines the idle state of the serial Clock. The
CPOL bit affects both the master and slave
modes.

0: SCK pin has a low level idle state

1: SCK pin has a high level idle state

Note: If CPOL is changed at the communication
byte boundaries, the SPI must be disabled by re-
setting the SPE bit.

Bit 2 = **CPHA** *Clock Phase*.

This bit is set and cleared by software.

0: The first clock transition is the first data capture
edge.1: The second clock transition is the first capture
edge.

Note: The slave must have the same CPOL and
CPHA settings as the master.

Bits 1:0 = **SPR[1:0]** *Serial Clock Frequency*.

These bits are set and cleared by software. Used
with the SPR2 bit, they select the baud rate of the
SPI serial clock SCK output by the SPI in master
mode.

Note: These 2 bits have no effect in slave mode.**Table 20. SPI Master mode SCK Frequency**

Serial Clock	SPR2	SPR1	SPR0
$f_{CPU}/4$	1	0	0
$f_{CPU}/8$	0	0	0
$f_{CPU}/16$	0	0	1
$f_{CPU}/32$	1	1	0
$f_{CPU}/64$	0	1	0
$f_{CPU}/128$	0	1	1

SERIAL PERIPHERAL INTERFACE (Cont'd)**Table 21. SPI Register Map and Reset Values**

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0021h	SPIDR Reset Value	MSB x	x	x	x	x	x	x	LSB x
0022h	SPICR Reset Value	SPIE 0	SPE 0	SPR2 0	MSTR 0	CPOL x	CPHA x	SPR1 x	SPR0 x
0023h	SPICSR Reset Value	SPIF 0	WCOL 0	OVR 0	MODF 0	0	SOD 0	SSM 0	SSI 0

SERIAL COMMUNICATIONS INTERFACE (Cont'd)

10.6.4.3 Receiver

The SCI can receive data words of either 8 or 9 bits. When the M bit is set, word length is 9 bits and the MSB is stored in the R8 bit in the SCICR1 register.

Character reception

During a SCI reception, data shifts in least significant bit first through the RDI pin. In this mode, the SCIDR register consists of a buffer (RDR) between the internal bus and the received shift register (see Figure 1.).

Procedure

- Select the M bit to define the word length.
- Select the desired baud rate using the SCIBRR and the SCIERPR registers.
- Set the RE bit, this enables the receiver which begins searching for a start bit.

When a character is received:

- The RDRF bit is set. It indicates that the content of the shift register is transferred to the RDR.
- An interrupt is generated if the RIE bit is set and the I bit is cleared in the CCR register.
- The error flags can be set if a frame error, noise or an overrun error has been detected during reception.

Clearing the RDRF bit is performed by the following software sequence done by:

1. An access to the SCISR register
2. A read to the SCIDR register.

The RDRF bit must be cleared before the end of the reception of the next character to avoid an overrun error.

Break Character

When a break character is received, the SCI handles it as a framing error.

Idle Character

When an idle frame is detected, there is the same procedure as a data received character plus an interrupt if the ILIE bit is set and the I bit is cleared in the CCR register.

Overrun Error

An overrun error occurs when a character is received when RDRF has not been reset. Data can not be transferred from the shift register to the

RDR register as long as the RDRF bit is not cleared.

When an overrun error occurs:

- The OR bit is set.
- The RDR content is not lost.
- The shift register is overwritten.
- An interrupt is generated if the RIE bit is set and the I bit is cleared in the CCR register.

The OR bit is reset by an access to the SCISR register followed by a SCIDR register read operation.

Noise Error

Oversampling techniques are used for data recovery by discriminating between valid incoming data and noise. Normal data bits are considered valid if three consecutive samples (8th, 9th, 10th) have the same bit value, otherwise the NF flag is set. In the case of start bit detection, the NF flag is set on the basis of an algorithm combining both valid edge detection and three samples (8th, 9th, 10th). Therefore, to prevent the NF flag getting set during start bit reception, there should be a valid edge detection as well as three valid samples.

When noise is detected in a frame:

- The NF flag is set at the rising edge of the RDRF bit.
- Data is transferred from the Shift register to the SCIDR register.
- No interrupt is generated. However this bit rises at the same time as the RDRF bit which itself generates an interrupt.

The NF flag is reset by a SCISR register read operation followed by a SCIDR register read operation.

During reception, if a false start bit is detected (e.g. 8th, 9th, 10th samples are 011,101,110), the frame is discarded and the receiving sequence is not started for this frame. There is no RDRF bit set for this frame and the NF flag is set internally (not accessible to the user). This NF flag is accessible along with the RDRF bit when a next valid frame is received.

Note: If the application Start Bit is not long enough to match the above requirements, then the NF Flag may get set due to the short Start Bit. In this case, the NF flag may be ignored by the application software when the first valid byte is received.

See also Section 0.1.4.10 .

SERIAL COMMUNICATIONS INTERFACE (Cont'd)**CONTROL REGISTER 1 (SCICR1)**

Read/Write

Reset Value: x000 0000 (x0h)

7							0
R8	T8	SCID	M	WAKE	PCE	PS	PIE

Bit 7 = R8 *Receive data bit 8.*

This bit is used to store the 9th bit of the received word when M = 1.

Bit 6 = T8 *Transmit data bit 8.*

This bit is used to store the 9th bit of the transmitted word when M = 1.

Bit 5 = SCID *Disabled for low power consumption*
 When this bit is set the SCI prescalers and outputs are stopped and the end of the current byte transfer in order to reduce power consumption. This bit is set and cleared by software.

0: SCI enabled

1: SCI prescaler and outputs disabled

Bit 4 = M *Word length.*

This bit determines the word length. It is set or cleared by software.

0: 1 Start bit, 8 Data bits, 1 Stop bit

1: 1 Start bit, 9 Data bits, 1 Stop bit

Note: The M bit must not be modified during a data transfer (both transmission and reception).

Bit 3 = WAKE *Wake-Up method.*

This bit determines the SCI Wake-Up method, it is set or cleared by software.

0: Idle Line

1: Address Mark

Bit 2 = PCE *Parity control enable.*

This bit selects the hardware parity control (generation and detection). When the parity control is enabled, the computed parity is inserted at the MSB position (9th bit if M = 1; 8th bit if M = 0) and parity is checked on the received data. This bit is set and cleared by software. Once it is set, PCE is active after the current byte (in reception and in transmission).

0: Parity control disabled

1: Parity control enabled

Bit 1 = PS *Parity selection.*

This bit selects the odd or even parity when the parity generation/detection is enabled (PCE bit set). It is set and cleared by software. The parity is selected after the current byte.

0: Even parity

1: Odd parity

Bit 0 = PIE *Parity interrupt enable.*

This bit enables the interrupt capability of the hardware parity control when a parity error is detected (PE bit set). It is set and cleared by software.

0: Parity error interrupt disabled

1: Parity error interrupt enabled.

10-BIT A/D CONVERTER (ADC) (Cont'd)**10.8.6 Register Description****CONTROL/STATUS REGISTER (ADCCSR)**

Read/Write (Except bit 7 read only)

Reset Value: 0000 0000 (00h)

7				0			
EOC	SPEED	ADON	0	CH3	CH2	CH1	CH0

Bit 7 = **EOC** *End of Conversion*

This bit is set by hardware. It is cleared by hardware when software reads the ADCDRH register or writes to any bit of the ADCCSR register.

0: Conversion is not complete

1: Conversion complete

Bit 6 = **SPEED** *ADC clock selection*

This bit is set and cleared by software.

0: $f_{ADC} = f_{CPU}/4$ 1: $f_{ADC} = f_{CPU}/2$ Bit 5 = **ADON** *A/D Converter on*

This bit is set and cleared by software.

0: Disable ADC and stop conversion

1: Enable ADC and start conversion

Bit 4 = **Reserved**. Must be kept cleared.Bit 3:0 = **CH[3:0]** *Channel Selection*

These bits are set and cleared by software. They select the analog input to convert.

Channel Pin*	CH3	CH2	CH1	CH0
AIN0	0	0	0	0
AIN1	0	0	0	1
AIN2	0	0	1	0
AIN3	0	0	1	1
AIN4	0	1	0	0
AIN5	0	1	0	1
AIN6	0	1	1	0
AIN7	0	1	1	1
AIN8	1	0	0	0
AIN9	1	0	0	1
AIN10	1	0	1	0
AIN11	1	0	1	1
AIN12	1	1	0	0
AIN13	1	1	0	1
AIN14	1	1	1	0
AIN15	1	1	1	1

*The number of channels is device dependent. Refer to the device pinout description.

DATA REGISTER (ADCDRH)

Read Only

Reset Value: 0000 0000 (00h)

7				0			
D9	D8	D7	D6	D5	D4	D3	D2

Bit 7:0 = **D[9:2]** *MSB of Converted Analog Value***DATA REGISTER (ADCDRL)**

Read Only

Reset Value: 0000 0000 (00h)

7				0			
0	0	0	0	0	0	D1	D0

Bit 7:2 = Reserved. Forced by hardware to 0.

Bit 1:0 = **D[1:0]** *LSB of Converted Analog Value*

SUPPLY CURRENT CHARACTERISTICS (Cont'd)**12.4.2 Supply and Clock Managers**

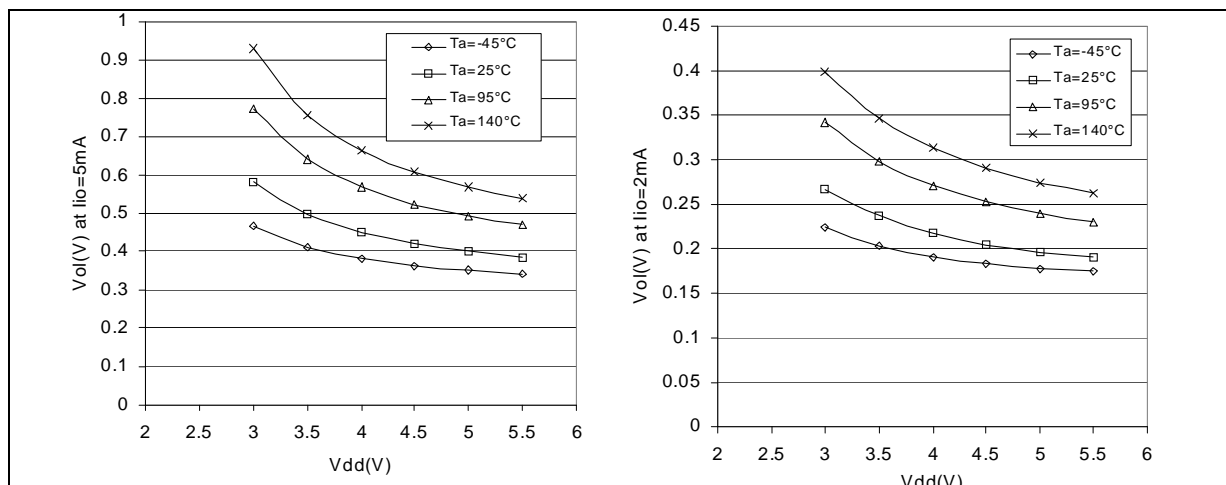
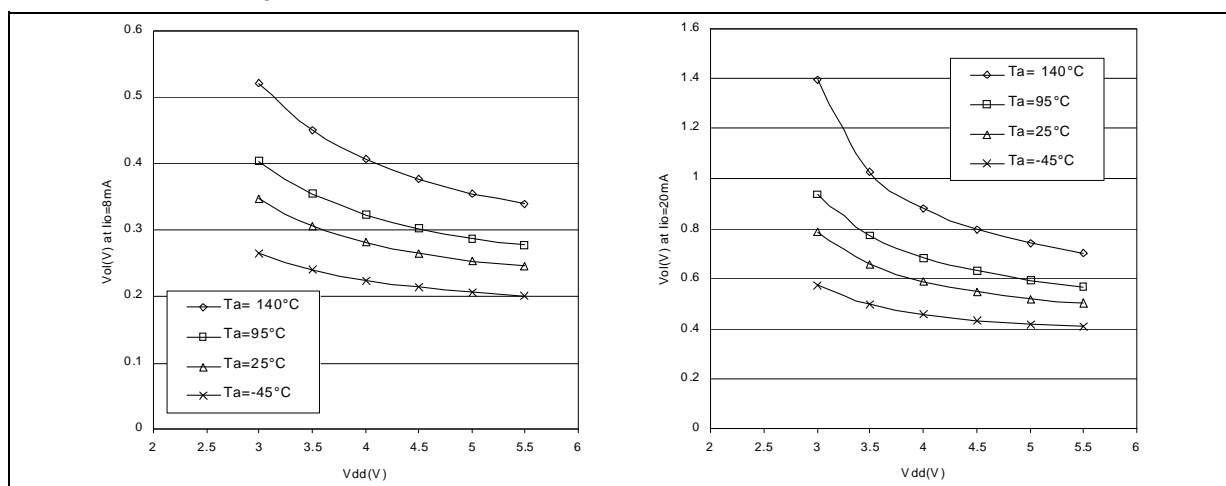
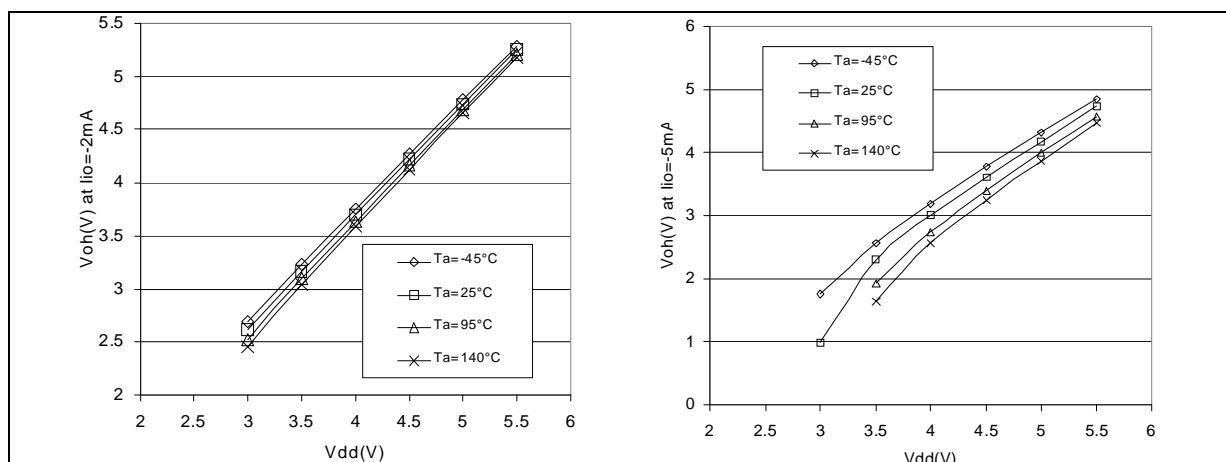
The previous current consumption specified for the ST7 functional operating modes over temperature range does not take into account the clock source current consumption. To get the total device consumption, the two current values must be added (except for HALT mode).

Symbol	Parameter	Conditions	Typ	Max	Unit
I _{DD(RCINT)}	Supply current of internal RC oscillator		625		μA
I _{DD(RES)}	Supply current of resonator oscillator ¹⁾ & ²⁾		see section 12.5.3 on page 150		
I _{DD(PLL)}	PLL supply current	V _{DD} = 5V	360		
I _{DD(CSS)}	Clock security system supply current	V _{DD} = 5V	250		
I _{DD(LVD)}	LVD supply current	V _{DD} = 5V	150	300	

Notes:

- 1.. Data based on characterization results done with the external components specified in Section 12.5.3, not tested in production.
2. As the oscillator is based on a current source, the consumption does not depend on the voltage.

I/O PORT PIN CHARACTERISTICS (Cont'd)

Figure 84. Typical V_{OL} vs. V_{DD} (standard)Figure 85. Typical V_{OL} vs. V_{DD} (high-sink)Figure 86. Typical $V_{DD}-V_{OH}$ vs. V_{DD} 

ADC CHARACTERISTICS (Cont'd)

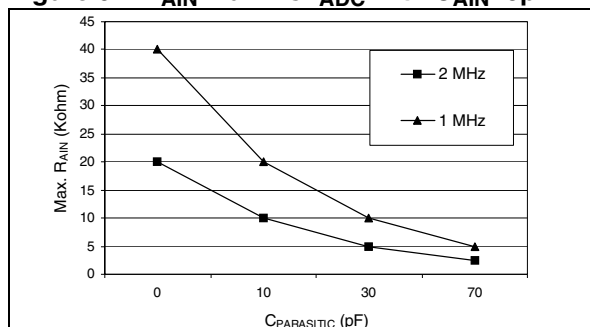
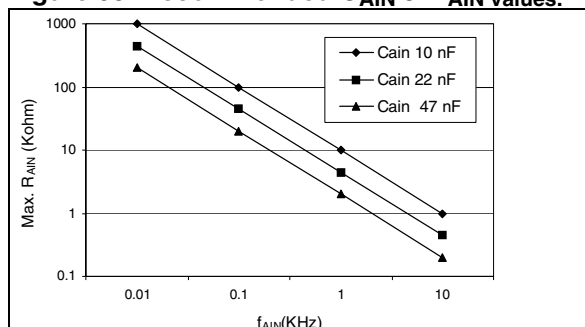
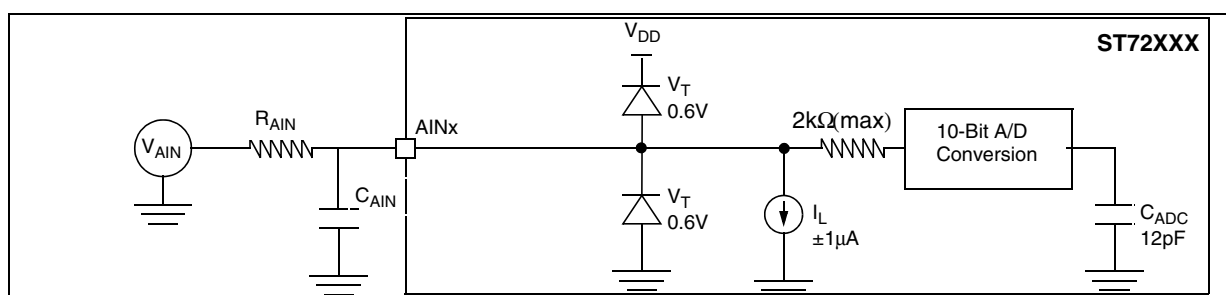
Figure 94. R_{AIN} max. vs f_{ADC} with $C_{AIN}=0pF$ ¹⁾Figure 95. Recommended C_{AIN} & R_{AIN} values.²⁾

Figure 96. Typical A/D Converter Application



Notes:

1. $C_{PARASITIC}$ represents the capacitance of the PCB (dependent on soldering and PCB layout quality) plus the pad capacitance (3pF). A high $C_{PARASITIC}$ value will downgrade conversion accuracy. To remedy this, f_{ADC} should be reduced.
2. This graph shows that depending on the input signal variation (f_{AIN}), C_{AIN} can be increased for stabilization time and decreased to allow the use of a larger serial resistor (R_{AIN}).

10-BIT ADC CHARACTERISTICS (Cont'd)

12.12.3 ADC Accuracy

Conditions: $V_{DD}=5V$ ¹⁾

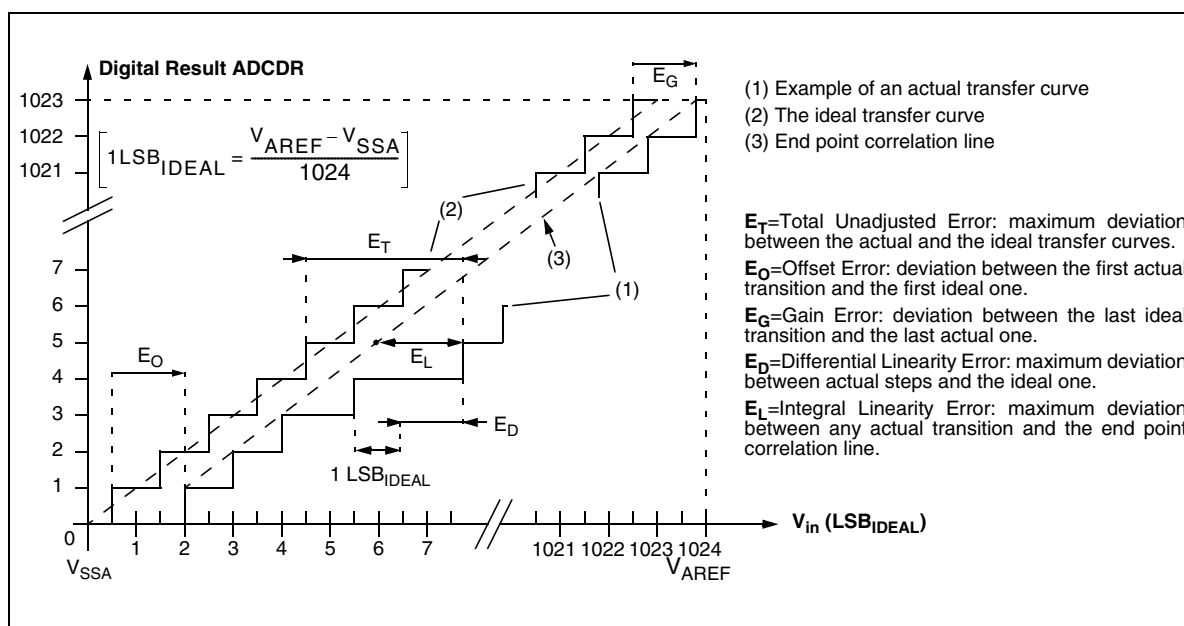
Symbol	Parameter	Conditions	Typ	Max ²⁾	Unit
$ E_T $	Total unadjusted error ¹⁾		3	4	LSB
$ E_O $	Offset error ¹⁾		2	3	
$ E_G $	Gain Error ¹⁾		0.5	3	
$ E_D $	Differential linearity error ¹⁾	CPU in run mode @ f_{ADC} 2 MHz.	1	2	
$ E_L $	Integral linearity error ¹⁾	CPU in run mode @ f_{ADC} 2 MHz.	1	2	

Notes:

1. ADC Accuracy vs. Negative Injection Current: Injecting negative current may reduce the accuracy of the conversion being performed on another analog input.

Any positive injection current within the limits specified for $I_{INJ(PIN)}$ and $\Sigma I_{INJ(PIN)}$ in Section 12.8 does not affect the ADC accuracy.

2. Data based on characterization results, monitored in production to guarantee 99.73% within \pm max value from -40°C to 125°C ($\pm 3\sigma$ distribution limits).

Figure 98. ADC Accuracy Characteristics

DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)

Table 31. Suggested List of Socket Types

Device	Socket	Emulator Adapter
LQFP64 14 x14	CAB 3303262	CAB 3303351
LQFP64 10 x10	YAMAICHI IC149-064-*75-*5	YAMAICHI ICP-064-6
LQFP48 7 X7	CAB 3303238	CAB 3303333
LQFP44 10 X10	YAMAICHI IC149-044-*52-*5	YAMAICHI ICP-044-5
LQFP32 7 X 7	IRONWOOD SF-QFE32SA-L-01	IRONWOOD SK-UGA06/32A-01

14.3.4 Socket and Emulator Adapter Information

For information on the type of socket that is supplied with the emulator, refer to the suggested list of sockets in Table 31.

Note: Before designing the board layout, it is recommended to check the overall dimensions of the socket as they may be greater than the dimensions of the device.

For footprint and other mechanical information about these sockets and adapters, refer to the manufacturer's datasheet.

Related Documentation

AN 978: ST7 Visual Develop Software Key Debugging Features

AN 1938: ST7 Visual Develop for ST7 Cosmic C toolset users

AN 1940: ST7 Visual Develop for ST7 Assembler Linker toolset users