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	LINbusSCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	48
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	3.8V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f361ar6tae

List of figures

Table 4. Hardware register map (continued)

Address	Block	Register label	Register name	Reset status	Remarks[1]
000Fh 0010h 0011h	Port F	PFDR PFDDR PFOR	Port F Data Register Port F Data Direction Register Port F Option Register	00h[2] 00h 00h	R/W[3] R/W[3] R/W[3]
0012h to 0020h	Reserved Area (15 bytes)				
0021h 0022h 0023h	SPI	SPIDR SPICR SPICSR	SPI Data I/O Register SPI Control Register SPI Control/Status Register	xxh 0xh 00h	R/W R/W R/W
0024h	FLASH	FCSR	Flash Control/Status Register	00h	R/W
0025h 0026h 0027h 0028h 0029h 002Ah	ITC	ISPR0 ISPR1 ISPR2 ISPR3 EICR0 EICR1	Interrupt Software Priority Register 0 Interrupt Software Priority Register 1 Interrupt Software Priority Register 2 Interrupt Software Priority Register 3 External Interrupt Control Register 0 External Interrupt Control Register 1	FFh FFh FFh FFh 00h 00h	R/W R/W R/W R/W R/W R/W
002Bh 002Ch	AWU	AWUCSR AWUPR	Auto Wake up f. Halt Control/Status Register Auto Wake Up From Halt Prescaler	00h FFh	R/W R/W
002Dh 002Eh	CKCTRL	SICSR MCCSR	System Integrity Control / Status Register Main Clock Control / Status Register	0xh 00h	R/W R/W
002Fh 0030h	WWDG	WDGCR WDGWR	Watchdog Control Register Watchdog Window Register	7Fh 7Fh	R/W R/W
0031h 0032h 0033h 0034h 0035h 0036h 0037h 0038h 0039h 003Ah 003Bh	PWM ART	PWMDCR3 PWMDCR2 PWMDCR1 PWMDCR0 PWMCR ARTCSR ARTCAR ARTARR ARTICCSR ARTICR1 ARTICR2	Pulse Width Modulator Duty Cycle Register 3 PWM Duty Cycle Register 2 PWM Duty Cycle Register 1 PWM Duty Cycle Register 0 PWM Control register Auto-Reload Timer Control/Status Register Auto-Reload Timer Counter Access Register Auto-Reload Timer Auto-Reload Register ART Input Capture Control/Status Register ART Input Capture Register 1 ART Input Capture register 2	00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h	R/W R/W R/W R/W R/W R/W R/W R/W R/W Read Only Read Only
003Ch 003Dh 003Eh 003Fh 0040h 0041h 0042h 0043h 0044h	8-BIT TIMER	T8CR2 T8CR1 T8CSR T8IC1R T8OC1R T8CTR T8ACTR T8IC2R T8OC2R	Timer Control Register 2 Timer Control Register 1 Timer Control/Status Register Timer Input Capture 1 Register Timer Output Compare 1 Register Timer Counter Register Timer Alternate Counter Register Timer Input Capture 2 Register Timer Output Compare 2 Register	00h 00h 00h xxh 00h FCh FCh xxh 00h	R/W R/W Read Only Read Only R/W Read Only Read Only Read Only R/W
0045h 0046h 0047h	ADC	ADCCSR ADCDRH ADCDRL	Control/Status Register Data High Register Data Low Register	00h 00h 00h	R/W Read Only Read Only

4 Central processing unit

4.1 Introduction

This CPU has a full 8-bit architecture and contains six internal registers allowing efficient 8-bit data manipulation.

4.2 Main features

- Enable executing 63 basic instructions
- Fast 8-bit by 8-bit multiply
- 17 main addressing modes (with indirect addressing mode)
- Two 8-bit index registers
- 16-bit stack pointer
- Low power HALT and WAIT modes
- Priority maskable hardware interrupts
- Non-maskable software/hardware interrupts

4.3 CPU registers

The six CPU registers shown in *Figure 8* are not present in the memory mapping and are accessed by specific instructions.

4.3.1 Accumulator (A)

The Accumulator is an 8-bit general purpose register used to hold operands and the results of the arithmetic and logic calculations and to manipulate data.

4.3.2 Index registers (X and Y)

These 8-bit registers are used to create effective addresses or as temporary storage areas for data manipulation. (The Cross-Assembler generates a precede instruction (PRE) to indicate that the following instruction refers to the Y register.)

The Y register is not affected by the interrupt automatic procedures.

4.3.3 Program counter (PC)

The program counter is a 16-bit register containing the address of the next instruction to be executed by the CPU. It is made of two 8-bit registers PCL (Program Counter Low which is the LSB) and PCH (Program Counter High which is the MSB).

This bit is accessed by the JRMI and JRPL instructions.

Bit 1 = **Z** *Zero*.

This bit is set and cleared by hardware. This bit indicates that the result of the last arithmetic, logical or data manipulation is zero.

0: The result of the last operation is different from zero.
1: The result of the last operation is zero.

This bit is accessed by the JREQ and JRNE test instructions.

Bit 0 = **C** *Carry/borrow.*

This bit is set and cleared by hardware and software. It indicates an overflow or an underflow has occurred during the last arithmetic operation.

0: No overflow or underflow has occurred.
1: An overflow or underflow has occurred.

This bit is driven by the SCF and RCF instructions and tested by the JRC and JRNC instructions. It is also affected by the “bit test and branch”, shift and rotate instructions.

Interrupt management bits

Bit 5,3 = **I1, I0** *Interrupt*

The combination of the I1 and I0 bits gives the current interrupt software priority.

Table 7. Interrupt software priority selection

Interrupt software priority	I1	I0
Level 0 (main)	1	0
Level 1	0	1
Level 2	0	0
Level 3 (= interrupt disable)	1	1

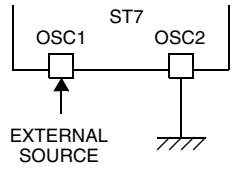
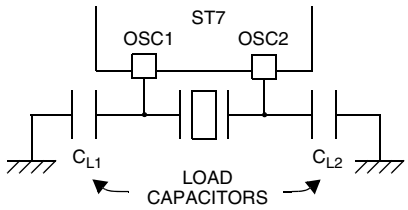
These two bits are set/cleared by hardware when entering in interrupt. The loaded value is given by the corresponding bits in the interrupt software priority registers (IxSPR). They can be also set/cleared by software with the RIM, SIM, IRET, HALT, WFI and PUSH/POP instructions.

See the interrupt management chapter for more details.

output distortion and start-up stabilization time. The loading capacitance values must be adjusted according to the selected oscillator.

These oscillators are not stopped during the RESET phase to avoid losing time in the oscillator start-up phase.

Table 8. ST7 clock sources

	Hardware configuration
External clock	
Crystal/Ceramic resonators	

5.5 Reset sequence manager (RSM)

5.5.1 Introduction

The reset sequence manager includes three RESET sources as shown in *Figure 13*:

- External $\overline{\text{RESET}}$ source pulse
- Internal LVD reset (Low Voltage Detection)
- Internal watchdog reset

These sources act on the $\overline{\text{RESET}}$ pin and it is always kept low during the delay phase.

The reset service routine vector is fixed at addresses FFFEh-FFFFh in the ST7 memory map.

The basic RESET sequence consists of three phases as shown in *Figure 12*:

- Active phase depending on the reset source
- 256 or 4096 CPU clock cycle delay (selected by option byte)
- RESET vector fetch

The 256 or 4096 CPU clock cycle delay allows the oscillator to stabilize and ensures that recovery has taken place from the Reset state. The shorter or longer clock cycle delay should be selected by option byte to correspond to the stabilization time of the external oscillator used in the application.

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 (that is, waiting for being serviced) will therefore be lost if the clear sequence is executed.

6.3 Interrupts and low power modes

All interrupts allow the processor to exit the WAIT low power mode. On the contrary, only external and other specified interrupts allow the processor to exit from the HALT modes (see column “Exit from HALT” in “Interrupt Mapping” table). When several pending interrupts are present while exiting HALT mode, the first one serviced can only be an interrupt with exit from HALT mode capability and it is selected through the same decision process shown in *Figure 18*.

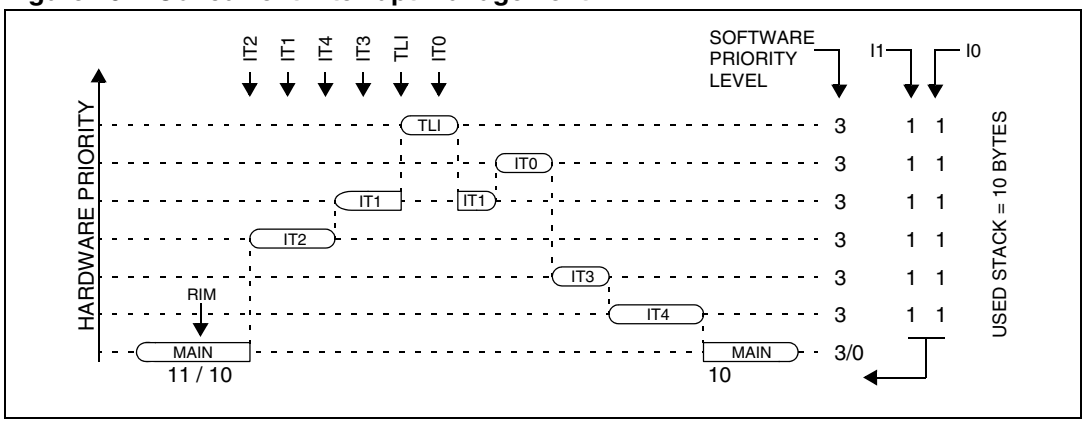
Note: If an interrupt, that is not able to Exit from HALT mode, is pending with the highest priority when exiting HALT mode, this interrupt is serviced after the first one serviced.

6.4 Concurrent & nested management

The following *Figure 19* and *Figure 20* show two different interrupt management modes. The first is called concurrent mode and does not allow an interrupt to be interrupted, unlike the nested mode in *Figure 20*. The interrupt hardware priority is given in this order from the lowest to the highest: MAIN, IT4, IT3, IT2, IT1, IT0, TLI. The software priority is given for each interrupt.

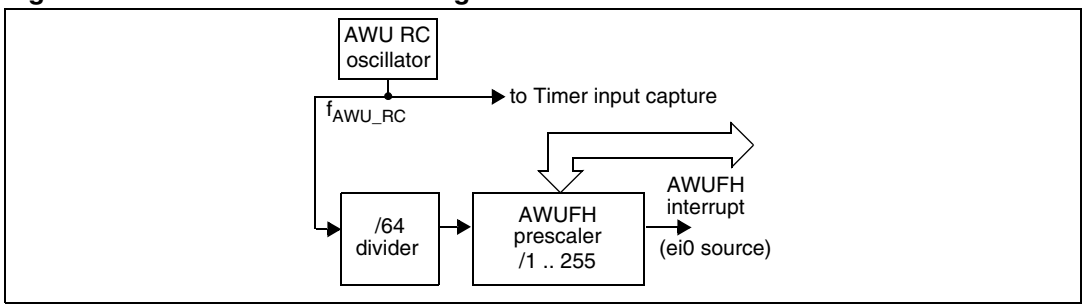
Warning: A stack overflow may occur without notifying the software of the failure.

Figure 19. Concurrent interrupt management



It is entered by executing the HALT instruction when the AWUEN bit in the AWUCSR register has been set and the OIE bit in the MCCSR register is cleared (see *Section 10: Main clock controller with real time clock MCC/RTC* for more details).

Figure 29. AWUFH mode block diagram



As soon as HALT mode is entered, and if the AWUEN bit has been set in the AWUCSR register, the AWU RC oscillator provides a clock signal (f_{AWU_RC}). Its frequency is divided by a fixed divider and a programmable prescaler controlled by the AWUPR register. The output of this prescaler provides the delay time. When the delay has elapsed the AWUF flag is set by hardware and an interrupt wakes up the MCU from Halt mode. At the same time the main oscillator is immediately turned on and a 256 or 4096 cycle delay is used to stabilize it. After this start-up delay, the CPU resumes operation by servicing the AWUFH interrupt. The AWU flag and its associated interrupt are cleared by software reading the AWUCSR register.

To compensate for any frequency dispersion of the AWU RC oscillator, it can be calibrated by measuring the clock frequency f_{AWU_RC} and then calculating the right prescaler value. Measurement mode is enabled by setting the AWUM bit in the AWUCSR register in Run mode. This connects f_{AWU_RC} to the ICAP1 input of the 16-bit timer, allowing the f_{AWU_RC} to be measured using the main oscillator clock as a reference time base.

Similarities with halt mode

The following AWUFH mode behavior is the same as normal Halt mode:

- The MCU can exit AWUFH mode by means of any interrupt with exit from Halt capability or a reset (see *Section 7.4: Halt mode*).
- When entering AWUFH mode, the I[1:0] bits in the CC register are forced to 10b to enable interrupts. Therefore, if an interrupt is pending, the MCU wakes up immediately.
- In AWUFH mode, the main oscillator is turned off causing all internal processing to be stopped, including the operation of the on-chip peripherals. None of the peripherals are clocked except those which get their clock supply from another clock generator (such as an external or auxiliary oscillator like the AWU oscillator).
- The compatibility of Watchdog operation with AWUFH mode is configured by the WDGHALT option bit in the option byte. Depending on this setting, the HALT instruction when executed while the Watchdog system is enabled, can generate a Watchdog RESET.

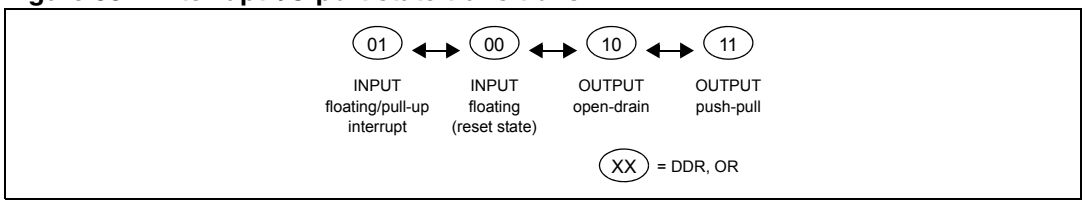
Warning: **The analog input voltage level must be within the limits stated in the absolute maximum ratings.**

8.3 I/O port implementation

The hardware implementation on each I/O port depends on the settings in the DDR and OR registers and specific feature of the I/O port such as ADC Input or true open drain.

Switching these I/O ports from one state to another should be done in a sequence that prevents unwanted side effects. Recommended safe transitions are illustrated in *Figure 33*. Other transitions are potentially risky and should be avoided, since they are likely to present unwanted side-effects such as spurious interrupt generation.

Figure 33. Interrupt I/O port state transitions



8.4 I/O port register configurations

The I/O port register configurations are summarized as follows.

8.4.1 Standard ports

Table 28. Configuration of PB7:6, PC0, PC3, PC7:5, PD3:2, PD5, PE7:0, PF7:0

Mode	DDR	OR
Floating input	0	0
Pull-up input		1
Open drain output	1	0
Push-pull output		1

Bit 5 = **TOIE** *Timer Overflow Interrupt Enable.*

0: Interrupt is inhibited.
1: A timer interrupt is enabled whenever the TOF bit of the SR register is set.

Bit 4 = **FOLV2** *Forced Output Compare 2.*
This bit is set and cleared by software.

0: No effect on the OCMP2 pin.
1:Forces the OLVL2 bit to be copied to the OCMP2 pin, if the OC2E bit is set and even if there is no successful comparison.

Bit 3 = **FOLV1** *Forced Output Compare 1.*
This bit is set and cleared by software.

0: No effect on the OCMP1 pin.
1: Forces OLVL1 to be copied to the OCMP1 pin, if the OC1E bit is set and even if there is no successful comparison.

Bit 2 = **OLVL2** *Output Level 2.*
This bit is copied to the OCMP2 pin whenever a successful comparison occurs with the OC2R register and OCxE is set in the CR2 register. This value is copied to the OCMP1 pin in One Pulse mode and Pulse Width Modulation mode.

Bit 1 = **IEDG1** *Input Edge 1.*
This bit determines which type of level transition on the ICAP1 pin will trigger the capture.

0: A falling edge triggers the capture.
1: A rising edge triggers the capture.

Bit 0 = **OLVL1** *Output Level 1.*
The OLVL1 bit is copied to the OCMP1 pin whenever a successful comparison occurs with the OC1R register and the OC1E bit is set in the CR2 register.

12.7.2 Control register 2 (CR2)

Read/ write

Reset value: 0000 0000 (00h)

7							0
OC1E	OC2E	OPM	PWM	CC1	CC0	IEDG2	EXEDG

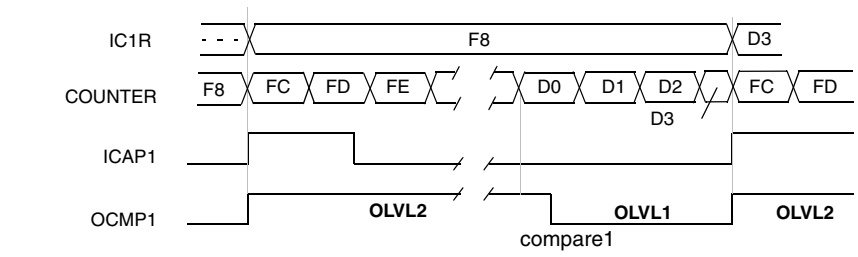
Bit 7 = **OC1E** *Output Compare 1 Pin Enable.*
This bit is used only to output the signal from the timer on the OCMP1 pin (OLV1 in Output Compare mode, both OLV1 and OLV2 in PWM and one-pulse mode). Whatever the value of the OC1E bit, the Output Compare 1 function of the timer remains active.

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.

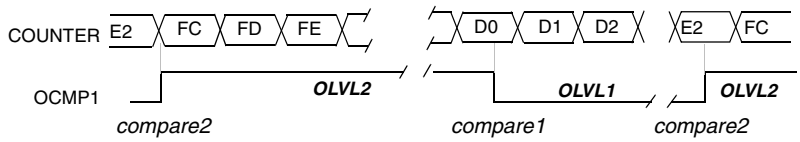
0: OCMP2 pin alternate function disabled (I/O pin free for general-purpose I/O).
1: OCMP2 pin alternate function enabled.

Figure 68. One pulse mode timing example



Note: IEDG1 = 1, OC1R = D0h, OLVL1 = 0, OLVL2 = 1

Figure 69. Pulse width modulation mode timing example



Note: OC1R = D0h, OC2R = E2, OLVL1 = 0, OLVL2 = 1

13.3.6 Pulse width modulation mode

Pulse Width Modulation (PWM) mode enables the generation of a signal with a frequency and pulse length determined by the value of the OC1R and OC2R registers.

Pulse Width Modulation mode uses the complete Output Compare 1 function plus the OC2R register, and so this functionality can not be used when PWM mode is activated.

In PWM mode, double buffering is implemented on the output compare registers. Any new values written in the OC1R and OC2R registers are taken into account only at the end of the PWM period (OC2) to avoid spikes on the PWM output pin (OCMP1).

Procedure

To use pulse width modulation mode:

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

Note: If the LINE bit is set, the WAKE bit is deactivated and replaced by the LHDM bit.

Bit 2 = **PCE** *Parity control enable.*
This bit is set and cleared by software. It selects the hardware parity control (generation and detection for byte parity, detection only for LIN parity).

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 will be 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). The parity error involved can be a byte parity error (if bit PCE is set and bit LPE is reset) or a LIN parity error (if bit PCE is set and bit LPE is set).

0: parity error interrupt disabled
1: parity error interrupt enabled

15.8.3 Control register 2 (SCICR2)

Read/ write

Reset value: 0000 0000 (00h)

7							0
TIE	TCIE	RIE	ILIE	TE	RE	RWU(1)	SBK(1)

1. This bit has a different function in LIN mode, please refer to the LIN mode register description.

Bit 7 = TIE *Transmitter interrupt enable*.
This bit is set and cleared by software.

0: interrupt is inhibited
1: in SCI interrupt is generated whenever TDRE = 1 in the SCISR register

16.8.3 Control register 2 (SCICR2)

Read/ write

Reset value: 0000 0000 (00h)

7							0
TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK

Bit 7 = TIE *Transmitter interrupt enable*.
This bit is set and cleared by software.

0: interrupt is inhibited
1: an SCI interrupt is generated whenever TDRE = 1 in the SCISR register

Bit 6 = **TCIE** *Transmission complete interrupt enable*
This bit is set and cleared by software.

0: interrupt is inhibited
1: an SCI interrupt is generated whenever TC = 1 in the SCISR register

Bit 5 = **RIE** *Receiver interrupt enable*.
This bit is set and cleared by software.

0: interrupt is inhibited
1: an SCI interrupt is generated whenever OR = 1 or RDRF = 1 in the SCISR register

Bit 4 = ILIE *Idle line interrupt enable.*
This bit is set and cleared by software.

0: interrupt is inhibited
1: an SCI interrupt is generated whenever IDLE = 1 in the SCISR register.

Bit 3 = TE *Transmitter enable.*
This bit enables the transmitter. It is set and cleared by software.

0: transmitter is disabled
1: transmitter is enabled

Note: During transmission, a "0" pulse on the TE bit ("0" followed by "1") sends a preamble (idle line) after the current word.

When TE is set there is a 1 bit-time delay before the transmission starts.

Bit 2 = RE *Receiver enable.*
This bit enables the receiver. It is set and cleared by software.

0: receiver is disabled
1: receiver is enabled and begins searching for a start bit

Bit 1 = RWU *Receiver wake-up.*
This bit determines if the SCI is in mute mode or not. It is set and cleared by software and can be cleared by hardware when a wake-up sequence is recognized.

0: receiver in active mode
1: receiver in mute mode

Note: Before selecting Mute mode (by setting the RWU bit) the SCI must first receive a data byte, otherwise it cannot function in Mute mode with wakeup by Idle line detection.

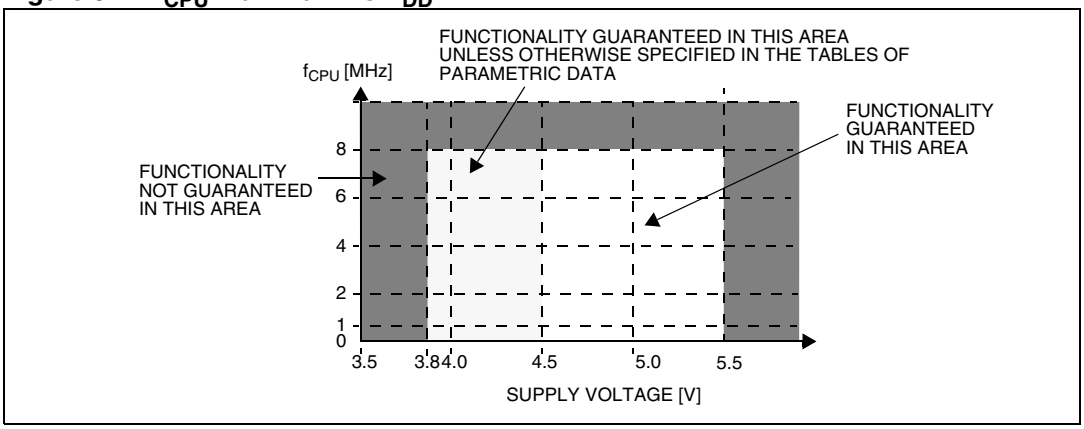
In Address Mark Detection Wake-Up configuration (WAKE bit = 1) the RWU bit cannot be modified by software while the RDRF bit is set.

19.3 Operating conditions

19.3.1 General operating conditions

Symbol	Parameter	Conditions	Min	Max	Unit
f_{CPU}	Internal clock frequency		0	8	MHz
V_{DD}	Extended operating voltage	No Flash write/ erase. Analog parameters not guaranteed.	3.8	4.5	V
	Standard operating voltage		4.5	5.5	
	Operating voltage for flash write/ erase	V_{PP} = 11.4 to 12.6V			
T_A	Ambient temperature range	A Suffix version	-40	85	°C
		C Suffix version		125	

Figure 97. f_{CPU} maximum vs V_{DD}



19.3.2 Operating conditions with low voltage detector (LVD)

Subject to general operating conditions for T_A.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{IT+(LVD)}$	Reset release threshold (V_{DD} rise)		4.0(1)	4.2	4.5	V
$V_{IT-(LVD)}$	Reset generation threshold (V_{DD} fall)		3.8	4.0	4.25(1)	
$V_{hys(LVD)}$	LVD voltage threshold hysteresis(1)	$V_{IT+(LVD)}$-$V_{IT-(LVD)}$	150	200	250	mV
Vt_{POR}	V_{DD} rise time rate(1)		6			µs/V
					100	ms/V
$t_{g(VDD)}$	V_{DD} glitches filtered (not detected) by LVD(1)	Measured at $V_{IT-(LVD)}$			40	ns

1. Data based on characterization results, not tested in production.

Table 91. Supply current consumption

Symbol	Parameter	Conditions		Flash devices		ROM devices		Unit
				Typ(1)	Max(2)	Typ(1)	Max(2)	
I_{DD}	Supply current in RUN mode(3)	f_{OSC} = 2 MHz, f_{CPU} = 1 MHz f_{OSC} = 4 MHz, f_{CPU} = 2 MHz f_{OSC} = 8 MHz, f_{CPU} = 4 MHz f_{OSC} = 16 MHz, f_{CPU} = 8 MHz		1.8 3.2 6 10	3 5 8 15	1.1 2.2 4.4 8.9	2 3.5 6 12	mA
	Supply current in SLOW mode(3)	f_{OSC} = 2 MHz, f_{CPU} = 62.5kHz f_{OSC} = 4 MHz, f_{CPU} = 125 kHz f_{OSC} = 8 MHz, f_{CPU} = 250 kHz f_{OSC} = 16 MHz, f_{CPU} = 500 kHz		0.5 0.6 0.85 1.25	2.7 3 3.6 4	0.1 0.2 0.4 0.8	0.2 0.4 0.8 1.5	
	Supply current in WAIT mode(3)	f_{OSC} = 2 MHz, f_{CPU} = 1 MHz f_{OSC} = 4 MHz, f_{CPU} = 2 MHz f_{OSC} = 8 MHz, f_{CPU} = 4 MHz f_{OSC} = 16 MHz, f_{CPU} = 8 MHz		1 1.8 3.4 6.4	3 4 5 7	0.7 1.4 2.9 5.7	3 4 5 7	
	Supply current in SLOW WAIT mode(2)	f_{OSC} = 2 MHz, f_{CPU} = 62.5 kHz f_{OSC} = 4 MHz, f_{CPU} = 125 kHz f_{OSC} = 8 MHz, f_{CPU} = 250 kHz f_{OSC} = 16 MHz, f_{CPU} = 500 kHz		0.4 0.5 0.6 0.8	1.2 1.3 1.8 2	0.07 0.14 0.28 0.56	0.12 0.2 0.5 1	
	Supply current in HALT mode(4)	V_{DD} = 5.5V	-40°C ≤ T_A ≤ +85°C	<1	10	<1	10	µA
			-40°C ≤ T_A ≤ +125°C		50		50	
	Supply current in ACTIVE HALT mode(4)(5)			0.5	1.2	0.18	0.25	mA
	Supply current in AWUFH mode(4)(5)	V_{DD} = 5.5V	-40°C ≤ T_A ≤ +85°C	25	30	25	30	µA
			-40°C ≤ T_A ≤ +125°C		70		70	

1. Typical data are based on T_A = 25°C, V_{DD} = 5V (4.5V ≤ V_{DD} ≤ 5.5V range).
2. Data based on characterization results, tested in production at V_{DD} max., f_{CPU} max. and T_A max.
3. Measurements are done in the following conditions:
   - Program executed from Flash, CPU running with Flash (for flash devices).
   - All I/O pins in input mode with a static value at V_{DD} or V_{SS} (no load)
   - All peripherals in reset state.
   - Clock input (OSC1) driven by external square wave.
   - In SLOW and SLOW WAIT mode, f_{CPU} is based on f_{OSC} divided by 32.

   To obtain the total current consumption of the device, add the clock source (*Section 20.5.1: Crystal and ceramic resonator oscillators*) and the peripheral power consumption (*Section 20.4.2: On-chip peripherals*).
4. All I/O pins in input mode with a static value at V_{DD} or V_{SS} (no load). Data based on characterization results, tested in production at V_{DD} max., f_{CPU} max. and T_A max.
5. This consumption refers to the Halt period only and not the associated run period which is software dependent.

19.4.1 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 obtain the total device consumption, the two current values must be added (except for HALT mode).

19.5.1 Crystal and ceramic resonator oscillators

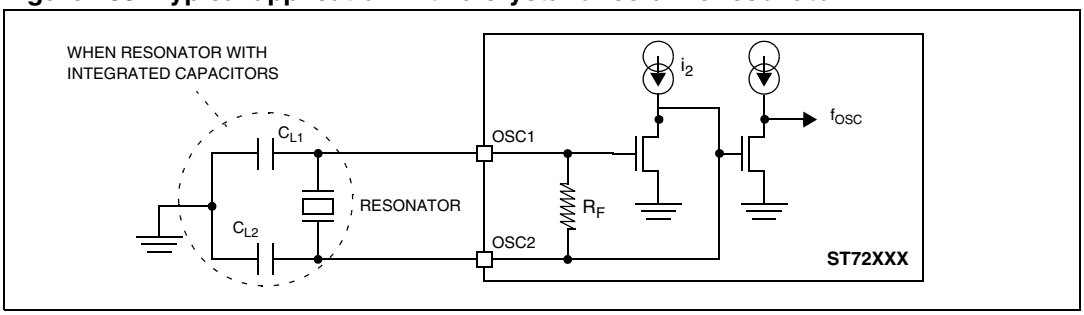
The ST7 internal clock can be supplied with four different crystal/ ceramic resonator oscillators. All the information given in this paragraph is based on characterization results with specified typical external components. In the application, the resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and start-up stabilization time. Refer to the crystal/ceramic resonator manufacturer for more details (frequency, package, accuracy...)[(a)(b)].

Table 96. Oscillator characteristics

Symbol	Parameter	Conditions	Min	Max	Unit
f_{OSC}	Oscillator Frequency[(1)]	LP: Low power oscillator MP: Medium power oscillator MS: Medium speed oscillator HS: High speed oscillator	1 >2 >4 >8	2 4 8 16	MHz
R_F	Feedback resistor		20	40	kΩ
C_{L1} C_{L2}	Recommended load capacitance versus equivalent serial resistance of the crystal or ceramic resonator (R_S)	R_S = 200Ω LP oscillator R_S = 200Ω MP oscillator R_S = 200Ω MS oscillator R_S = 100Ω HS oscillator	22 22 18 15	56 46 33 33	pF
i_2	OSC2 driving current	V_{DD} = 5V LP oscillator V_{IN} = V_{SS} MP oscillator MS oscillator HS oscillator	80 160 310 610	150 250 460 910	µA

1. The oscillator selection can be optimized in terms of supply current using an high quality resonator with small R_S value. Refer to crystal/ceramic resonator manufacturer for more details.

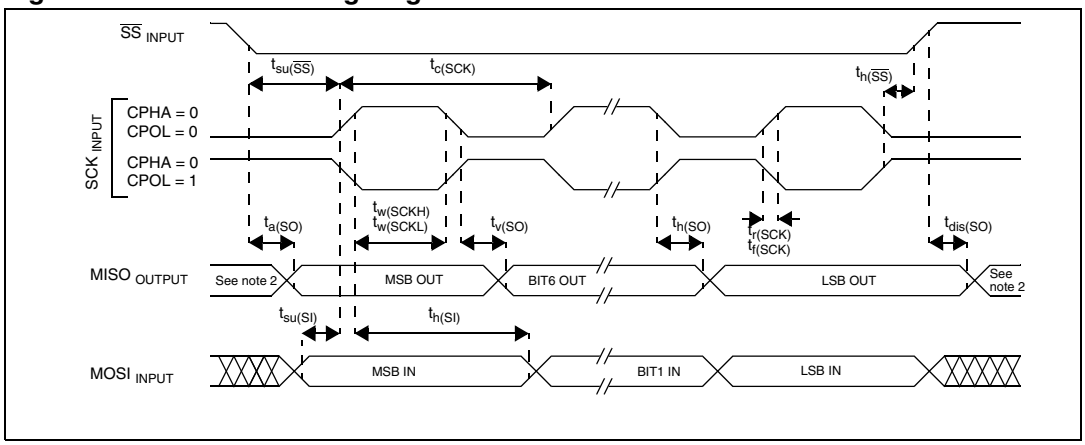
Figure 100. Typical application with a crystal or ceramic resonator



a. Resonator characteristics given by the crystal/ceramic resonator manufacturer.

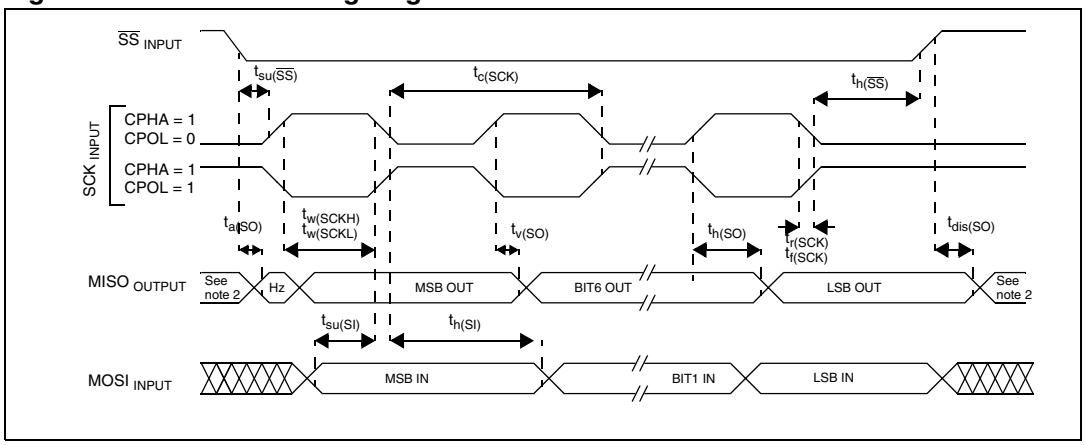
b. $t_{SU(OSC)}$ is the typical oscillator start-up time measured between V_{DD} = 2.8V and the fetch of the first instruction (with a quick V_{DD} ramp-up from 0 to 5V (< 50μs).

Figure 116. SPI slave timing diagram with CPHA = 0



1. Measurement points are done at CMOS levels: 0.3 x V_{DD} and 0.7 x V_{DD}
2. When no communication is on-going the data output line of the SPI (MOSI in master mode, MISO in slave mode) has its alternate function capability released. In this case, the pin status depends on the I/O port configuration.

Figure 117. SPI slave timing diagram with CPHA = 1



1. Measurement points are done at CMOS levels: 0.3 x V_{DD} and 0.7 x V_{DD}.
2. When no communication is on-going the data output line of the SPI (MOSI in master mode, MISO in slave mode) has its alternate function capability released. In this case, the pin status depends on the I/O port configuration.

```
PUSH CC
.ext1_rt; entry to interrupt routine
LD A, #$00
LD sema, A
IRET
```

23.1.4 Unexpected reset fetch

If an interrupt request occurs while a "POP CC" instruction is executed, the interrupt controller does not recognize the source of the interrupt and, by default, passes the RESET vector address to the CPU.

Workaround

To solve this issue, a "POP CC" instruction must always be preceded by a "SIM" instruction.

23.1.5 Header time-out does not prevent wake-up from mute mode

Normally, when LINSCI is configured in LIN slave mode, if a header time-out occurs during a LIN header reception (that is, header length > 57 bits), the LIN Header Error bit (LHE) is set, an interrupt occurs to inform the application but the LINSCI should stay in mute mode, waiting for the next header reception.

Problem description

The LINSCI sampling period is Tbit / 16. If a LIN Header time-out occurs between the 9th and the 15th sample of the Identifier Field Stop Bit (refer to *Figure 155*), the LINSCI wakes up from mute mode. Nevertheless, LHE is set and LIN Header Detection Flag (LHDF) is kept cleared.

In addition, if LHE is reset by software before this 15th sample (by accessing the SCISR register and reading the SCIDR register in the LINSCI interrupt routine), the LINSCI will generate another LINSCI interrupt (due to the RDRF flag setting).

Impact on application

Software may execute the interrupt routine twice after header reception.

Moreover, in reception mode, as the receiver is no longer in mute mode, an interrupt will be generated on each data byte reception.

Workaround

The problem can be detected in the LINSCI interrupt routine. In case of timeout error (LHE is set and LHLR is loaded with 00h), the software can check the RWU bit in the SCICR2 register. If RWU is cleared, it can be set by software. Refer to *Figure 156*. Workaround is shown in bold characters.

Table 120. Document revision history (continued)

Date	Revision	Changes
19-Sep-2006	1 (continued)	Updated “ST72361-Auto MICROCONTROLLER OPTION LIST” on page 215 as follows: - added 16K devices to FASTROM options - specified “7” as maximum number of special marking characters for TQFP32 package - replaced Standard and Automotive temperature versions with temperature ranges A and B - removed footnote “16K ROM in automotive version only” Changed Section 15 on page 216 and added Table 37 on page 216 Deleted Section 15.1.5 “Clearing active interrupts outside interrupt routine” (text already exists in Section 16.1.2 on page 217) Added Section 16.1.5 on page 219 Updated disclaimer (last page) to include a mention about the use of ST products in automotive applications
07-Mar-2008	2	Removed ‘mcu’ from the URL reference in “PROGRAMMING TOOLS” on page 1 Removed section on “SOLDERING AND GLUEABILITY INFORMATION” and replaced with “Soldering information”
02-Aug-2010	3	Updated following figures: *Figure 128: ST72F361xx-Auto Flash commercial product structure on page 264* *Figure 129: ST72P361xxx-Auto FastROM commercial product structure on page 265* *Figure 130: ST72361xx-Auto ROM commercial product structure on page 266* Added *Section 20.4: Packaging for automatic handling on page 259*