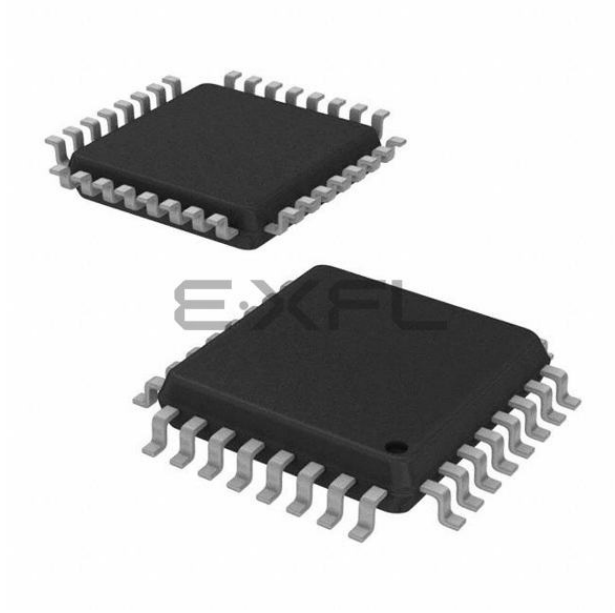


**Welcome to E-XFL.COM**

## 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 | 24 |
| Program Memory Size | 60KB (60K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 2K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3.8V ~ 5.5V |
| Data Converters | A/D 6x10b |
| Oscillator Type | External |
| 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/st72f361k9tc |

# 3 REGISTER AND MEMORY MAP

As shown in Figure 5, the MCU is capable of addressing 64 Kbytes of memories and I/O registers.

The available memory locations consist of 128 bytes of register locations, up to 2 Kbytes of RAM and up to 60 Kbytes of user program memory.

The RAM space includes up to 256 bytes for the stack from 0100h to 01FFh.The highest address bytes contain the user reset and interrupt vectors.

**IMPORTANT:** Memory locations marked as "Reserved" must never be accessed. Accessing a reseved area can have unpredictable effects on the device.

**Figure 5. Memory Map**

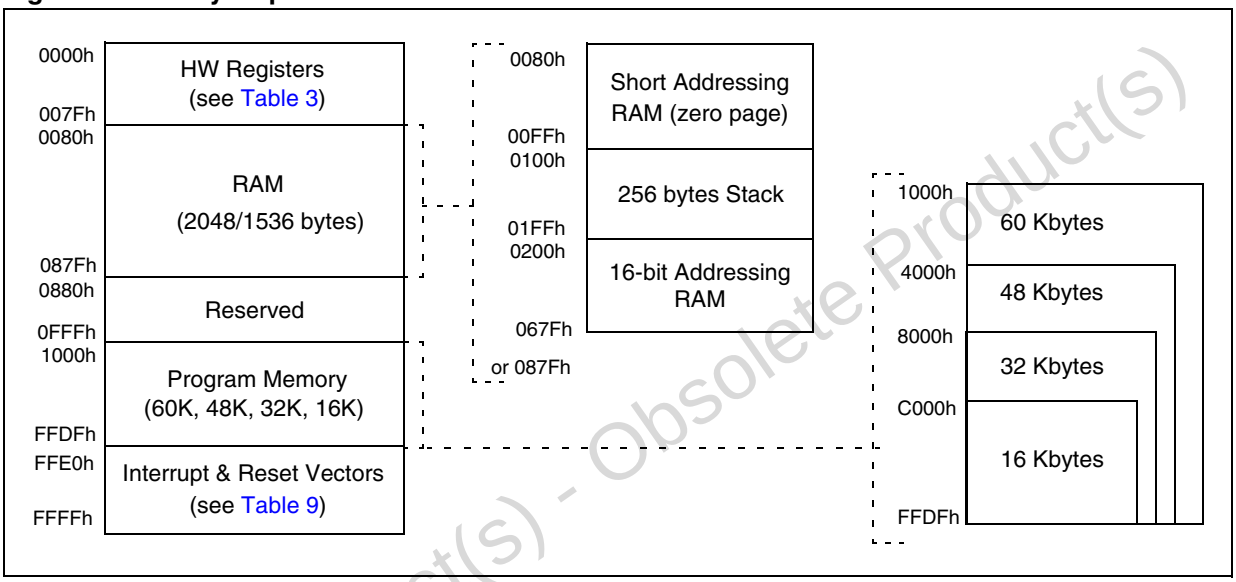





**Table 3. Hardware Register Map**

| Address | Block | Register Label | Register Name | Reset Status | Remarks |
|---|---|---|---|---|---|
| 0000h | Port A | PADR | Port A Data Register | 00h[1] | R/W[2] |
| 0001h | | PADDR | Port A Data Direction Register | 00h | R/W[2] |
| 0002h | | PAOR | Port A Option Register | 00h | R/W[2] |
| 0003h | Port B | PBDR | Port B Data Register | 00h[1] | R/W[2] |
| 0004h | | PBDDR | Port B Data Direction Register | 00h | R/W[2] |
| 0005h | | PBOR | Port B Option Register | 00h | R/W[2] |
| 0006h | Port C | PCDR | Port C Data Register | 00h[1] | R/W[2] |
| 0007h | | PCDDR | Port C Data Direction Register | 00h | R/W[2] |
| 0008h | | PCOR | Port C Option Register | 00h | R/W[2] |
| 0009h | Port D | PDDR | Port D Data Register | 00h[1] | R/W[2] |
| 000Ah | | PDDDR | Port D Data Direction Register | 00h | R/W[2] |
| 000Bh | | PDOR | Port D Option Register | 00h | R/W[2] |
| 000Ch | Port E | PEDR | Port E Data Register | 00h[1] | R/W[2] |
| 000Dh | | PEDDR | Port E Data Direction Register | 00h | R/W[2] |
| 000Eh | | PEOR | Port E Option Register | 00h | R/W[2] |

| Address | Block | Register Label | Register Name | Reset Status | Remarks |
|---|---|---|---|---|---|
| 0048h | LINSCI1 (LIN Master/ Slave) | SCI1ISR | SCI1 Status Register | C0h | Read Only |
| 0049h | | SCI1DR | SCI1 Data Register | xxh | R/W |
| 004Ah | | SCI1BRR | SCI1 Baud Rate Register | 00h | R/W |
| 004Bh | | SCI1CR1 | SCI1 Control Register 1 | xxh | R/W |
| 004Ch | | SCI1CR2 | SCI1 Control Register 2 | 00h | R/W |
| 004Dh | | SCI1CR3 | SCI1Control Register 3 | 00h | R/W |
| 004Eh | | SCI1ERPR | SCI1 Extended Receive Prescaler Register | 00h | R/W |
| 004Fh | | SCI1ETPR | SCI1 Extended Transmit Prescaler Register | 00h | R/W |
| 0050h | Reserved Area (1 byte) | | | | |
| 0051h | 16-BIT TIMER | T16CR2 | Timer Control Register 2 | 00h | R/W |
| 0052h | | T16CR1 | Timer Control Register 1 | 00h | R/W |
| 0053h | | T16CSR | Timer Control/Status Register | 00h | R/W |
| 0054h | | T16IC1HR | Timer Input Capture 1 High Register | xxh | Read Only |
| 0055h | | T16IC1LR | Timer Input Capture 1 Low Register | xxh | Read Only |
| 0056h | | T16OC1HR | Timer Output Compare 1 High Register | 80h | R/W |
| 0057h | | T16OC1LR | Timer Output Compare 1 Low Register | 00h | R/W |
| 0058h | | T16CHR | Timer Counter High Register | FFh | Read Only |
| 0059h | | T16CLR | Timer Counter Low Register | FCh | Read Only |
| 005Ah | | T16ACHR | Timer Alternate Counter High Register | FFh | Read Only |
| 005Bh | | T16ACLR | Timer Alternate Counter Low Register | FCh | Read Only |
| 005Ch | | T16IC2HR | Timer Input Capture 2 High Register | xxh | Read Only |
| 005Dh | | T16IC2LR | Timer Input Capture 2 Low Register | xxh | Read Only |
| 005Eh | | T16OC2HR | Timer Output Compare 2 High Register | 80h | R/W |
| 005Fh | | T16OC2LR | Timer Output Compare 2 Low Register | 00h | R/W |
| 0060h | LINSCI2 (LIN Master) | SCI2SR | SCI2 Status Register | C0h | Read Only |
| 0061h | | SCI2DR | SCI2 Data Register | xxh | R/W |
| 0062h | | SCI2BRR | SCI2 Baud Rate Register | 00h | R/W |
| 0063h | | SCI2CR1 | SCI2 Control Register 1 | xxh | R/W |
| 0064h | | SCI2CR2 | SCI2 Control Register 2 | 00h | R/W |
| 0065h | | SCI2CR3 | SCI2 Control Register 3 | 00h | R/W |
| 0066h | | SCI2ERPR | SCI2 Extended Receive Prescaler Register | 00h | R/W |
| 0067h | | SCI2ETPR | SCI2 Extended Transmit Prescaler Register | 00h | R/W |
| 0068h to 007Fh | Reserved area (24 bytes) | | | | |



**Legend**: x = undefined, R/W = read/write

**Notes**:

1. The contents of the I/O port DR registers are readable only in output configuration. In input configuration, the values of the I/O pins are returned instead of the DR register contents.
2. The bits associated with unavailable pins must always keep their reset value.

# 6 SUPPLY, RESET AND CLOCK MANAGEMENT

The device includes a range of utility features for securing the application in critical situations (for example, in case of a power brown-out), and reducing the number of external components. An overview is shown in Figure 11.

For more details, refer to dedicated parametric section.

### Main features

- Optional PLL for multiplying the frequency by 2
- Reset Sequence Manager (RSM)
- Multi-Oscillator Clock Management (MO)
  - 4 Crystal/Ceramic resonator oscillators
- System Integrity Management (SI)
  - Main supply Low voltage detection (LVD)
  - Auxiliary Voltage detector (AVD) with interrupt capability for monitoring the main supply

## 6.1 PHASE LOCKED LOOP

If the clock frequency input to the PLL is in the range 2 to 4 MHz, the PLL can be used to multiply the frequency by two to obtain an $f_{OSC2}$ of 4 to 8 MHz. The PLL is enabled by option byte. If the PLL is disabled, then $f_{OSC2} = f_{OSC}/2$.

**Caution:** The PLL is not recommended for applications where timing accuracy is required. See "PLL Characteristics" on page 187.

**Figure 10. PLL Block Diagram**

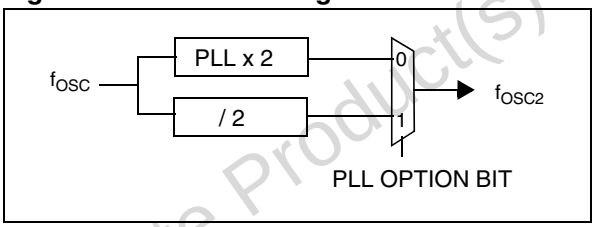
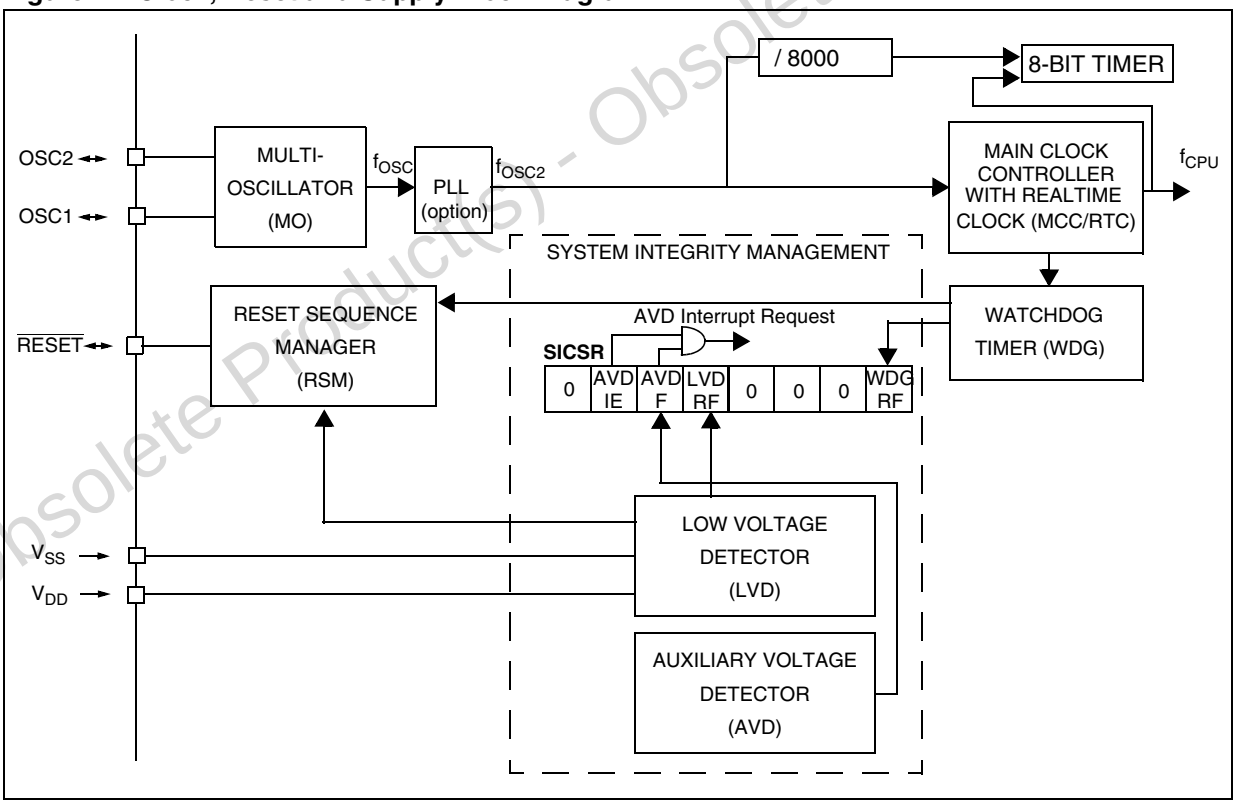



**Figure 11. Clock, Reset and Supply Block Diagram**



**POWER SAVING MODES** (Cont'd)

### 8.6.1 Register Description

**AWUFH CONTROL/STATUS REGISTER (AWUCSR)**
Read/Write (except bit 2 read only)
Reset Value: 0000 0000 (00h)

| 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | AWUF | AWUM | AWUEN |

Bits 7:3 = Reserved.

Bit 2 = **AWUF** *Auto Wake-Up Flag*
This bit is set by hardware when the AWU module generates an interrupt and cleared by software on reading AWUCSR.
0: No AWU interrupt occurred
1: AWU interrupt occurred

Bit 1 = **AWUM** *Auto Wake-Up Measurement*
This bit enables the AWU RC oscillator and connects its output to the ICAP1 input of the 16-bit timer. This allows the timer to be used to measure the AWU RC oscillator dispersion and then compensate this dispersion by providing the right value in the AWUPR register.
0: Measurement disabled
1: Measurement enabled

Bit 0 = **AWUEN** *Auto Wake-Up From Halt Enabled*
This bit enables the Auto Wake-Up From Halt feature: once HALT mode is entered, the AWUFH wakes up the microcontroller after a time delay defined by the AWU prescaler value. It is set and cleared by software.
0: AWUFH (Auto Wake-Up From Halt) mode disabled
1: AWUFH (Auto Wake-Up From Halt) mode enabled

**AWUFH PRESCALER REGISTER (AWUPR)**
Read/Write
Reset Value: 1111 1111 (FFh)

| 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|
| AWUPR7 | AWUPR6 | AWUPR5 | AWUPR4 | AWUPR3 | AWUPR2 | AWUPR1 | AWUPR0 |

Bits 7:0 = **AWUPR[7:0]** *Auto Wake-Up Prescaler*
These 8 bits define the AWUPR Dividing factor (as explained below:

| AWUPR[7:0] | Dividing factor |
|---|---|
| 00h | Forbidden (See note) |
| 01h | 1 |
| ... | ... |
| FEh | 254 |
| FFh | 255 |

In AWU mode, the period that the MCU stays in Halt Mode ($t_{AWU}$ in Figure 30) is defined by

$$t_{AWU} = 64 \times AWUPR \times \frac{1}{f_{AWURC}} + t_{RCSTRT}$$

This prescaler register can be programmed to modify the time that the MCU stays in Halt mode before waking up automatically.

**Note:** If 00h is written to AWUPR, depending on the product, an interrupt is generated immediately after a HALT instruction or the AWUPR remains unchanged.

**Table 11. AWU Register Map and Reset Values**

| Address (Hex.) | Register Label | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| 002Bh | **AWUCSR** Reset Value | 0 | 0 | 0 | 0 | 0 | AWUF 0 | AWUM 0 | AWUEN 0 |
| 002Ch | **AWUPR** Reset Value | AWUPR7 1 | AWUPR6 1 | AWUPR5 1 | AWUPR4 1 | AWUPR3 1 | AWUPR2 1 | AWUPR1 1 | AWUPR0 1 |

**I/O PORTS** (Cont'd)

**Table 14. Port Configuration**

| Port | Pin name | Input | | Output | |
|---|---|---|---|---|---|
| | | OR = 0 | OR = 1 | OR = 0 | OR = 1 |
| Port A | PA0 | floating | pull-up interrupt (ei0) | open drain | push-pull |
| | PA1 | | floating interrupt (ei0) | | |
| | PA2 | | pull-up interrupt (ei0) | | |
| | PA3 | | floating interrupt (ei0) | | |
| | PA4 | | pull-up interrupt (ei0) | | |
| | PA5 | | floating interrupt (ei0) | | |
| | PA6 | | pull-up interrupt (ei0) | | |
| | PA7 | | floating interrupt (ei0) | | |
| Port B | PB0 | floating | pull-up interrupt (ei1) | open drain | push-pull |
| | PB1 | | floating interrupt (ei1) | | |
| | PB2 | | pull-up interrupt (ei1) | | |
| | PB3 | | floating interrupt (ei1) | | |
| | PB4 | | pull-up interrupt (ei1) | | |
| | PB5 | | floating interrupt (ei1) | | |
| Port C | PC0 | floating | pull-up | open drain | push-pull |
| | PC1 | | pull-up interrupt (ei2) | | |
| | PC2 | | floating interrupt (ei2) | | |
| | PC3 | | pull-up | | |
| | PC4 | pull-up | | N/A | |
| | PC7:5 | floating | pull-up | open drain | push-pull |
| Port D | PD0 | floating | pull-up interrupt (ei3) | open drain | push-pull |
| | PD1 | | floating interrupt (ei3) | | |
| | PD3:2 | | pull-up | | |
| | PD4 | | floating interrupt (ei3) | | |
| | PD5 | | pull-up | | |
| | PD6 | | pull-up interrupt (ei3) | | |
| | PD7 | | floating interrupt (ei3) | | |
| Port E | PE7:0 | floating (TTL) | pull-up (TTL) | open drain | push-pull |
| Port F | PF7:0 | floating (TTL) | pull-up (TTL) | open drain | push-pull |

**PWM AUTO-RELOAD TIMER** (Cont'd)

### Output compare and Time base interrupt

On overflow, the OVF flag of the ARTCSR register is set and an overflow interrupt request is generated if the overflow interrupt enable bit, OIE, in the ARTCSR register, is set. The OVF flag must be reset by the user software. This interrupt can be used as a time base in the application.

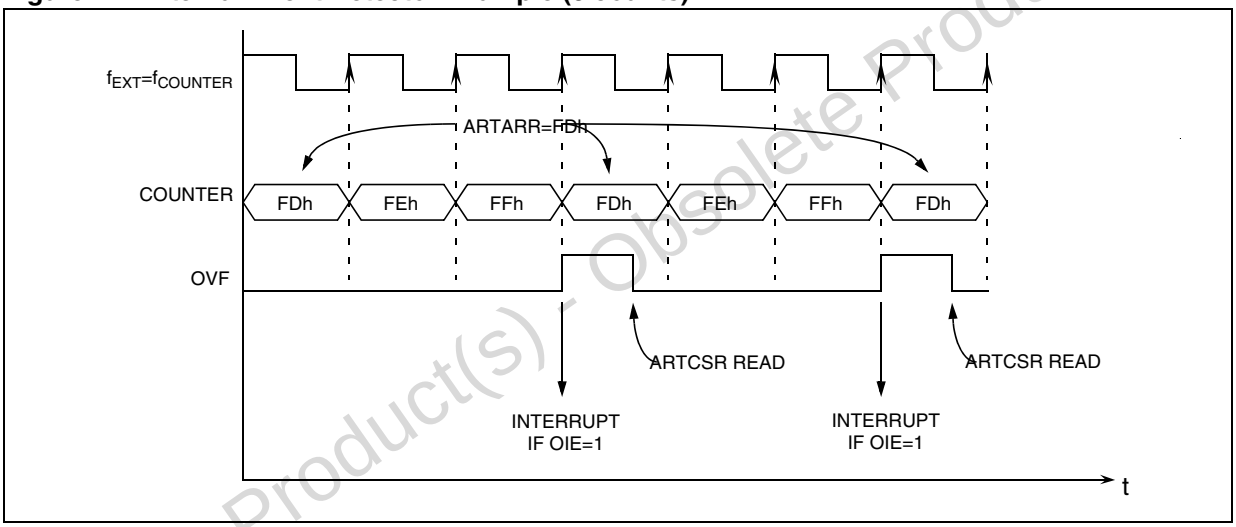
### External clock and event detector mode

Using the $f_{EXT}$ external prescaler input clock, the auto-reload timer can be used as an external clock event detector. In this mode, the ARTARR register is used to select the $n_{EVENT}$ number of events to be counted before setting the OVF flag.

$n_{EVENT}$ = 256 - ARTARR

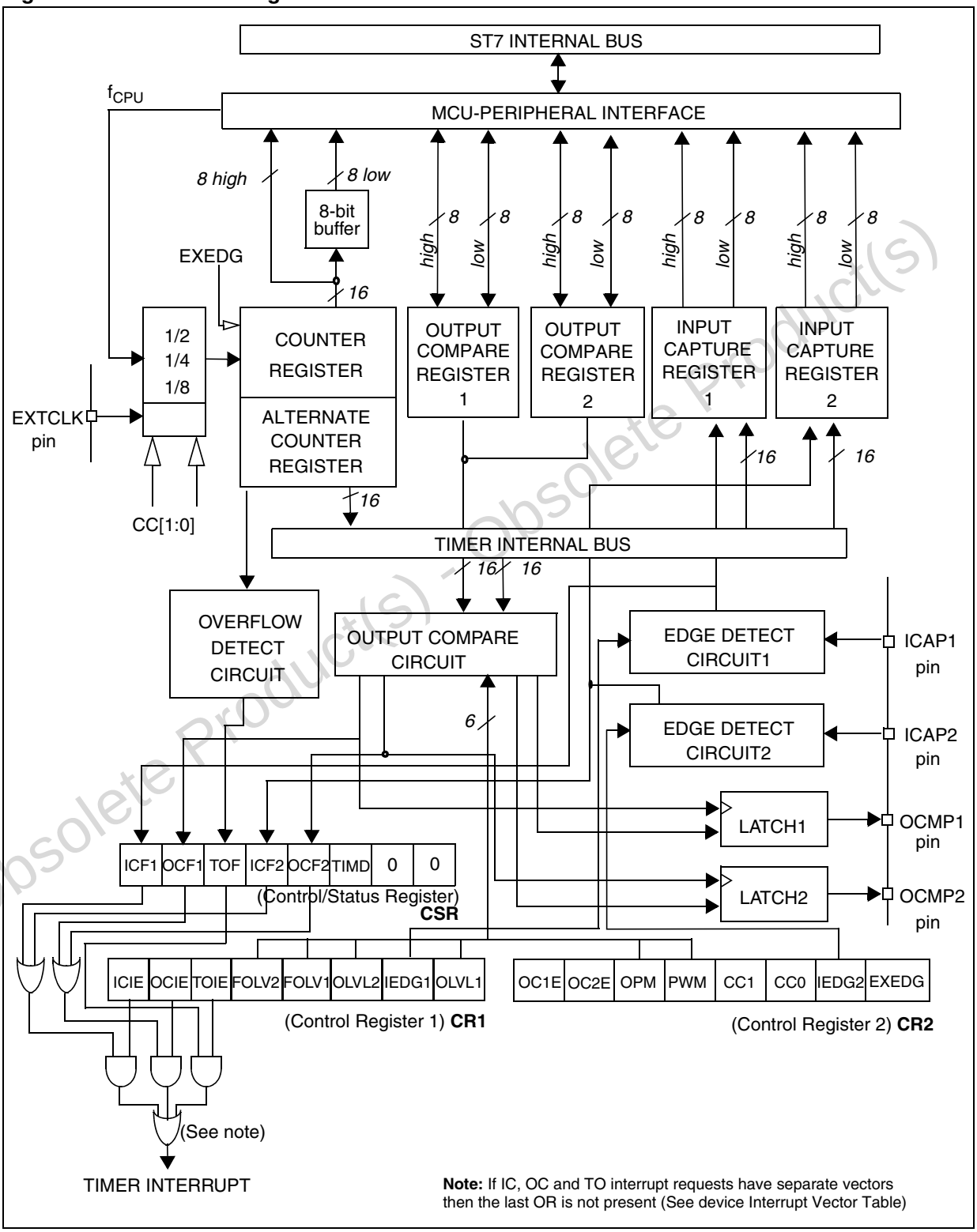
**Caution:** The external clock function is not available in HALT mode. If HALT mode is used in the application, prior to executing the HALT instruction, the counter must be disabled by clearing the TCE bit in the ARTCSR register to avoid spurious counter increments.

**Figure 44. External Event Detector Example (3 counts)**



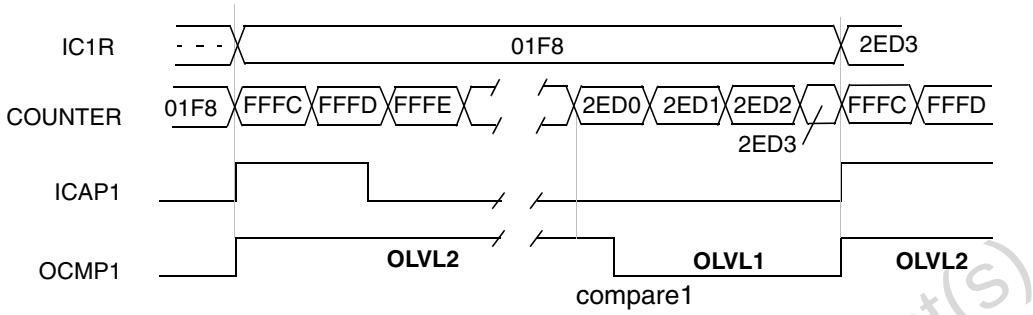
**16-BIT TIMER** (Cont'd)

**Figure 48. Timer Block Diagram**



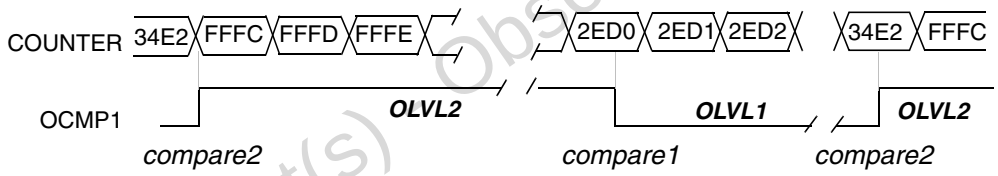
**16-BIT TIMER** (Cont'd)

**Figure 57. One Pulse Mode Timing Example**




**Note:** IEDG1 = 1, OC1R = 2ED0h, OLVL1 = 0, OLVL2 = 1

**Figure 58. Pulse Width Modulation Mode Timing Example with 2 Output Compare Functions**




**Note:** OC1R = 2ED0h, OC2R = 34E2, OLVL1 = 0, OLVL2 = 1

**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.

**8-BIT TIMER** (Cont'd)

Whatever the timer mode used (input capture, output compare, one pulse mode or PWM mode) an overflow occurs when the counter rolls over from FFh to 00h then:

– The TOF bit of the SR register is set.

– A timer interrupt is generated if:

  – TOIE bit of the CR1 register is set and

  – I bit of the CC register is cleared.

If one of these conditions is false, the interrupt remains pending to be issued as soon as they are both true.

Clearing the overflow interrupt request is done in two steps:

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

**Notes:** The TOF bit is not cleared by accesses to ACTR register. The advantage of accessing the ACTR register rather than the CTR register is that it allows simultaneous use of the overflow function and reading the free running counter at random times (for example, to measure elapsed time) without the risk of clearing the TOF bit erroneously.

The timer is not affected by WAIT mode.

In HALT mode, the counter stops counting until the mode is exited. Counting then resumes from the previous count (MCU awakened by an interrupt) or from the reset count (MCU awakened by a Reset).

### SERIAL PERIPHERAL INTERFACE (Cont'd)

**Table 22. SPI Register Map and Reset Values**

| Address (Hex.) | Register Label | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| 21 | **SPIDR** Reset Value | MSB x | x | x | x | x | x | x | LSB x |
| 22 | **SPICR** Reset Value | SPIE 0 | SPE 0 | SPR2 0 | MSTR 0 | CPOL x | CPHA x | SPR1 x | SPR0 x |
| 23 | **SPICSR** Reset Value | SPIF 0 | WCOL 0 | OVR 0 | MODF 0 | 0 | SOD 0 | SSM 0 | SSI 0 |

## 10.7 LINSCI SERIAL COMMUNICATION INTERFACE (LIN MASTER/SLAVE)

### 10.7.1 Introduction

The Serial Communications Interface (SCI) offers a flexible means of full-duplex data exchange with external equipment requiring an industry standard NRZ asynchronous serial data format. The SCI offers a very wide range of baud rates using two baud rate generator systems.

The LIN-dedicated features support the LIN (Local Interconnect Network) protocol for both master and slave nodes.

This chapter is divided into SCI Mode and LIN mode sections. For information on general SCI communications, refer to the SCI mode section. For LIN applications, refer to both the SCI mode and LIN mode sections.

### 10.7.2 SCI Features

- Full duplex, asynchronous communications
- NRZ standard format (Mark/Space)
- Independently programmable transmit and receive baud rates up to 500K baud
- Programmable data word length (8 or 9 bits)
- Receive buffer full, Transmit buffer empty and End of Transmission flags
- 2 receiver wake-up modes:
  - Address bit (MSB)
  - Idle line
- Muting function for multiprocessor configurations
- Separate enable bits for Transmitter and Receiver
- Overrun, Noise and Frame error detection
- 6 interrupt sources
  - Transmit data register empty
  - Transmission complete
  - Receive data register full
  - Idle line received
  - Overrun error
  - Parity interrupt
- Parity control:
  - Transmits parity bit
  - Checks parity of received data byte
- Reduced power consumption mode

### 10.7.3 LIN Features

– LIN Master
  – 13-bit LIN Synch Break generation

– LIN Slave
  – Automatic Header Handling
  – Automatic baud rate resynchronization based on recognition and measurement of the LIN Synch Field (for LIN slave nodes)
  – Automatic baud rate adjustment (at CPU frequency precision)
  – 11-bit LIN Synch Break detection capability
  – LIN Parity check on the LIN Identifier Field (only in reception)
  – LIN Error management
  – LIN Header Timeout
  – Hot plugging support

**LINSCI™ SERIAL COMMUNICATION INTERFACE (SCI Mode)** (cont'd)

#### 10.7.5.2 Transmitter

The transmitter can send data words of either 8 or 9 bits depending on the M bit status. When the M bit is set, word length is 9 bits and the 9th bit (the MSB) has to be stored in the T8 bit in the SCICR1 register.

**Character Transmission**

During an SCI transmission, data shifts out least significant bit first on the TDO pin. In this mode, the SCIDR register consists of a buffer (TDR) between the internal bus and the transmit 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 SCIETPR registers.

– Set the TE bit to send a preamble of 10 (M = 0) or 11 (M = 1) consecutive ones (Idle Line) as first transmission.

– Access the SCISR register and write the data to send in the SCIDR register (this sequence clears the TDRE bit). Repeat this sequence for each data to be transmitted.

Clearing the TDRE bit is always performed by the following software sequence:
1. An access to the SCISR register
2. A write to the SCIDR register

The TDRE bit is set by hardware and it indicates:

– The TDR register is empty.

– The data transfer is beginning.

– The next data can be written in the SCIDR register without overwriting the previous data.

This flag generates an interrupt if the TIE bit is set and the I[|1:0] bits are cleared in the CCR register.

When a transmission is taking place, a write instruction to the SCIDR register stores the data in the TDR register and which is copied in the shift register at the end of the current transmission.

When no transmission is taking place, a write instruction to the SCIDR register places the data directly in the shift register, the data transmission starts, and the TDRE bit is immediately set.

When a character transmission is complete (after the stop bit) the TC bit is set and an interrupt is generated if the TCIE is set and the I[1:0] bits are cleared in the CCR register.

Clearing the TC bit is performed by the following software sequence:
1. An access to the SCISR register
2. A write to the SCIDR register

**Note:** The TDRE and TC bits are cleared by the same software sequence.

**Break Characters**

Setting the SBK bit loads the shift register with a break character. The break character length depends on the M bit (see Figure 2).

As long as the SBK bit is set, the SCI sends break characters to the TDO pin. After clearing this bit by software, the SCI inserts a logic 1 bit at the end of the last break character to guarantee the recognition of the start bit of the next character.

**Idle Line**

Setting the TE bit drives the SCI to send a preamble of 10 (M = 0) or 11 (M = 1) consecutive '1's (idle line) before the first character.

In this case, clearing and then setting the TE bit during a transmission sends a preamble (idle line) after the current word. Note that the preamble duration (10 or 11 consecutive '1's depending on the M bit) does not take into account the stop bit of the previous character.

**Note:** Resetting and setting the TE bit causes the data in the TDR register to be lost. Therefore the best time to toggle the TE bit is when the TDRE bit is set, that is, before writing the next byte in the SCIDR.

## LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Mode)

### 10.7.9 LIN Mode - Functional Description.

The block diagram of the Serial Control Interface, in LIN slave mode is shown in Figure 5.

It uses six registers:

– 3 control registers: SCICR1, SCICR2 and SCICR3

– 2 status registers: the SCISR register and the LHLR register mapped at the SCIERPR address

– A baud rate register: LPR mapped at the SCIBRR address and an associated fraction register LPFR mapped at the SCIETPR address

The bits dedicated to LIN are located in the SCICR3. Refer to the register descriptions in Section 0.1.10 for the definitions of each bit.

#### 10.7.9.1 Entering LIN Mode

To use the LINSCI in LIN mode the following configuration must be set in SCICR3 register:

– Clear the M bit to configure 8-bit word length.

– Set the LINE bit.

**Master**

To enter master mode the LSLV bit must be reset In this case, setting the SBK bit will send 13 low bits.

Then the baud rate can programmed using the SCIBRR, SCIERPR and SCIETPR registers.

In LIN master mode, the Conventional and / or Extended Prescaler define the baud rate (as in standard SCI mode)

**Slave**

Set the LSLV bit in the SCICR3 register to enter LIN slave mode. In this case, setting the SBK bit will have no effect.

In LIN Slave mode the LIN baud rate generator is selected instead of the Conventional or Extended Prescaler. The LIN baud rate generator is common to the transmitter and the receiver.

Then the baud rate can be programmed using LPR and LPRF registers.

**Note:** It is mandatory to set the LIN configuration first before programming LPR and LPRF, because the LIN configuration uses a different baud rate generator from the standard one.

#### 10.7.9.2 LIN Transmission

In LIN mode the same procedure as in SCI mode has to be applied for a LIN transmission.

To transmit the LIN Header the proceed as follows:

– First set the SBK bit in the SCICR2 register to start transmitting a 13-bit LIN Synch Break

– reset the SBK bit

– Load the LIN Synch Field (0x55) in the SCIDR register to request Synch Field transmission

– Wait until the SCIDR is empty (TDRE bit set in the SCISR register)

– Load the LIN message Identifier in the SCIDR register to request Identifier transmission.

**LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Mode)** (cont'd)

SCICR2 register is set, the LHDM bit selects the Wake-Up method (replacing the WAKE bit).
0: LIN Synch Break Detection Method
1: LIN Identifier Field Detection Method

Bit 2 = **LHIE** *LIN Header Interrupt Enable*
This bit is set and cleared by software. It is only usable in LIN Slave mode.
0: LIN Header Interrupt is inhibited.
1: An SCI interrupt is generated whenever LHDF = 1.

Bit 1 = **LHDF** *LIN Header Detection Flag*
This bit is set by hardware when a LIN Header is detected and cleared by a software sequence (an access to the SCISR register followed by a read of the SCICR3 register). It is only usable in LIN Slave mode.
0: No LIN Header detected.
1: LIN Header detected.

**Notes:** The header detection method depends on the LHDM bit:

– If LHDM = 0, a header is detected as a LIN Synch Break.

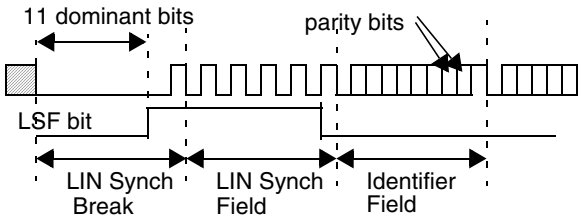
– If LHDM = 1, a header is detected as a LIN Identifier, meaning that a LIN Synch Break Field + a LIN Synch Field + a LIN Identifier Field have been consecutively received.

Bit 0 = **LSF** *LIN Synch Field State*

This bit indicates that the LIN Synch Field is being analyzed. It is only used in LIN Slave mode. In Auto Synchronization Mode (LASE bit = 1), when the SCI is in the LIN Synch Field State it waits or counts the falling edges on the RDI line.

It is set by hardware as soon as a LIN Synch Break is detected and cleared by hardware when the LIN Synch Field analysis is finished (see Figure 11). This bit can also be cleared by software to exit LIN Synch State and return to idle mode.
0: The current character is not the LIN Synch Field
1: LIN Synch Field State (LIN Synch Field undergoing analysis)

**Figure 87. LSF Bit Set and Clear**




## LIN DIVIDER REGISTERS

LDIV is coded using the two registers LPR and LPFR. In LIN Slave mode, the LPR register is accessible at the address of the SCIBRR register and the LPFR register is accessible at the address of the SCIETPR register.

### LIN PRESCALER REGISTER (LPR)

**Read/Write**
Reset Value: 0000 0000 (00h)

| 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|
| LPR7 | LPR6 | LPR5 | LPR4 | LPR3 | LPR2 | LPR1 | LPR0 |

**LPR[7:0]** *LIN Prescaler (mantissa of LDIV)*

These 8 bits define the value of the mantissa of the LIN Divider (LDIV):

| LPR[7:0] | Rounded Mantissa (LDIV) |
|---|---|
| 00h | SCI clock disabled |
| 01h | 1 |
| ... | ... |
| FEh | 254 |
| FFh | 255 |

**Caution:** LPR and LPFR registers have different meanings when reading or writing to them. Consequently bit manipulation instructions (BRES or BSET) should never be used to modify the LPR[7:0] bits, or the LPFR[3:0] bits.

**LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Master Only)** (Cont'd)

### 10.8.4.4 Conventional Baud Rate Generation

The baud rates for the receiver and transmitter (Rx and Tx) are set independently and calculated as follows

:

$$Tx = \frac{f_{CPU}}{(16*PR)*TR} \qquad Rx = \frac{f_{CPU}}{(16*PR)*RR}$$

with:

PR = 1, 3, 4 or 13 (see SCP[1:0] bits)

TR = 1, 2, 4, 8, 16, 32, 64, 128

(see SCT[2:0] bits)

RR = 1, 2, 4, 8, 16, 32, 64, 128

(see SCR[2:0] bits)

All these bits are in the SCIBRR register.

**Example:** If $f_{CPU}$ is 8 MHz (normal mode) and if PR = 13 and TR = RR = 1, the transmit and receive baud rates are 38400 baud.

**Note:** The baud rate registers MUST NOT be changed while the transmitter or the receiver is enabled.

### 10.8.4.5 Extended Baud Rate Generation

The extended prescaler option gives a very fine tuning on the baud rate, using a 255 value prescaler, whereas the conventional Baud Rate Generator retains industry standard software compatibility.

The extended baud rate generator block diagram is described in the Figure 90.

The output clock rate sent to the transmitter or to the receiver is the output from the 16 divider divided by a factor ranging from 1 to 255 set in the SCIERPR or the SCIETPR register.

**Note:** The extended prescaler is activated by setting the SCIETPR or SCIERPR register to a value other than zero. The baud rates are calculated as follows:

$$Tx = \frac{f_{CPU}}{16*ETPR*(PR*TR)} \qquad Rx = \frac{f_{CPU}}{16*ERPR*(PR*RR)}$$

with:

ETPR = 1, ..., 255 (see SCIETPR register)

ERPR = 1, ..., 255 (see SCIERPR register)

### 10.8.4.6 Receiver Muting and Wake-up Feature

In multiprocessor configurations it is often desirable that only the intended message recipient should actively receive the full message contents, thus reducing redundant SCI service overhead for all non addressed receivers.

The non addressed devices may be placed in sleep mode by means of the muting function.

Setting the RWU bit by software puts the SCI in sleep mode:

All the reception status bits cannot be set.

All the receive interrupts are inhibited.

A muted receiver may be awakened by one of the following two ways:

– by Idle Line detection if the WAKE bit is reset,

– by Address Mark detection if the WAKE bit is set.

Receiver wakes-up by Idle Line detection when the Receive line has recognized an Idle Frame. Then the RWU bit is reset by hardware but the IDLE bit is not set.

Receiver wakes-up by Address Mark detection when it received a "1" as the most significant bit of a word, thus indicating that the message is an address. The reception of this particular word wakes up the receiver, resets the RWU bit and sets the RDRF bit, which allows the receiver to receive this word normally and to use it as an address word.

## 12.2 ABSOLUTE MAXIMUM RATINGS

Stresses above those listed as “absolute maximum ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

### 12.2.1 Voltage Characteristics

| Symbol | Ratings | Maximum value | Unit |
|---|---|---|---|
| $V_{DD}$ - $V_{SS}$ | Supply voltage | 6.5 | V |
| $V_{PP}$ - $V_{SS}$ | Programming Voltage | 13 | |
| $V_{IN}$ | Input voltage on any pin[1)2)] | $V_{SS}$ - 0.3 to $V_{DD}$ + 0.3 | |
| $\lvert\Delta V_{DDx}\rvert$ and $\lvert\Delta V_{SSx}\rvert$ | Variations between different digital power pins | 50 | mV |
| $\lvert V_{SSA} - V_{SSx}\rvert$ | Variations between digital and analog ground pins | 50 | |
| $V_{ESD(HBM)}$ | Electro-static discharge voltage (Human Body Model) | see Section 12.8.3 on page 192 | |
| $V_{ESD(MM)}$ | Electro-static discharge voltage (Machine Model) | | |

### 12.2.2 Current Characteristics

| Symbol | Ratings | Maximum value | Unit |
|---|---|---|---|
| $I_{VDD}$ | Total current into $V_{DD}$ power lines (source)[3)] | 150 | mA |
| $I_{VSS}$ | Total current out of $V_{SS}$ ground lines (sink)[3)] | | |
| $I_{IO}$ | Output current sunk by any standard I/O and control pin | 25 | |
| | Output current sunk by any high sink I/O pin | 50 | |
| | Output current source by any I/Os and control pin | - 25 | |
| $I_{INJ(PIN)}$[2)4)] | Injected current on $V_{PP}$ pin | ± 5 | |
| | Injected current on $\overline{\text{RESET}}$ pin | | |
| | Injected current on OSC1 and OSC2 pins | | |
| | Injected current on PB3 (on Flash devices) | +5 | |
| | Injected current on any other pin[5)] | ± 5 | |
| $\Sigma I_{INJ(PIN)}$[2)] | Total injected current (sum of all I/O and control pins)[5)] | ± 25 | |

### 12.2.3 Thermal Characteristics

| Symbol | Ratings | Value | Unit |
|---|---|---|---|
| $T_{STG}$ | Storage temperature range | -65 to +150 | °C |
| $T_J$ | Maximum junction temperature (see Section 13.2 "THERMAL CHARACTERISTICS") | | |

**Notes:**

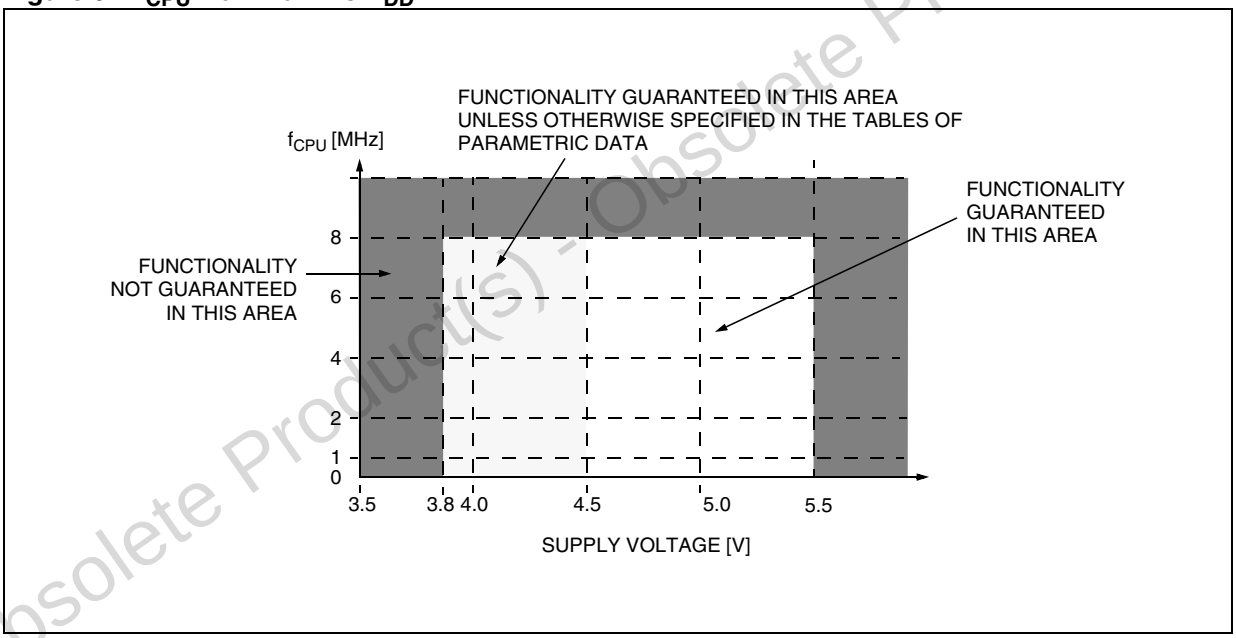
1. Directly connecting the $\overline{\text{RESET}}$ and I/O pins to $V_{DD}$ or $V_{SS}$ could damage the device if an unintentional internal reset is generated or an unexpected change of the I/O configuration occurs (for example, due to a corrupted program counter). To guarantee safe operation, this connection has to be done through a pull-up or pull-down resistor (typical: 4.7kΩ for $\overline{\text{RESET}}$, 10kΩ for I/Os). Unused I/O pins must be tied in the same way to $V_{DD}$ or $V_{SS}$ according to their reset configuration.
2. $I_{INJ(PIN)}$ must never be exceeded. This is implicitly insured if $V_{IN}$ maximum is respected. If $V_{IN}$ maximum cannot be respected, the injection current must be limited externally to the $I_{INJ(PIN)}$ value. A positive injection is induced by $V_{IN}>V_{DD}$ while a negative injection is induced by $V_{IN} < V_{SS}$. For true open-drain pads, there is no positive injection current, and the corresponding $V_{IN}$ maximum must always be respected
3. All power ($V_{DD}$) and ground ($V_{SS}$) lines must always be connected to the external supply.
4. Negative injection disturbs the analog performance of the device. See note in “10-BIT ADC CHARACTERISTICS” on page 204.
5. When several inputs are submitted to a current injection, the maximum $\Sigma I_{INJ(PIN)}$ is the absolute sum of the positive and negative injected currents (instantaneous values). These results are based on characterization with $\Sigma I_{INJ(PIN)}$ maximum current injection on four I/O port pins of the device.

## 12.3 OPERATING CONDITIONS

### 12.3.1 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 | 4.5 | 5.5 | |
| $T_A$ | Ambient temperature range | 1 Suffix Version | 0 | 70 | °C |
| | | 5 Suffix Version | -10 | 85 | |
| | | 6 Suffix Version | -40 | 85 | |
| | | 7 Suffix Version | | 105 | |
| | | 3 Suffix Version | | 125 | |

**Figure 97. $f_{CPU}$ Maximum vs $V_{DD}$**




**Note:** It is mandatory to connect all available $V_{DD}$ and $V_{DDA}$ pins to the supply voltage and all $V_{SS}$ and $V_{SSA}$ pins to ground.

**SUPPLY CURRENT CHARACTERISTICS** (Cont'd)

### 12.4.1 Supply and Clock Managers

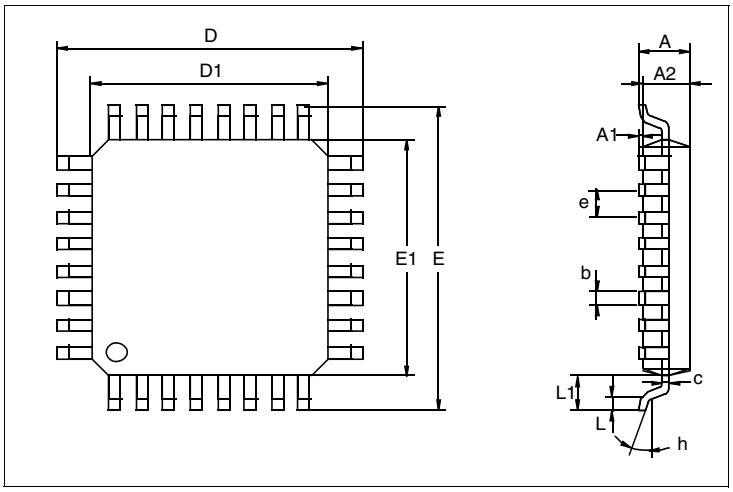
The previous current consumption specified for the ST7 functional operating modes over temperature range does not take into account the clock source current consumption. To obtain the total device consumption, the two current values must be added (except for HALT mode).

| Symbol | Parameter | Conditions | Typ | Max[1)] | Unit |
|---|---|---|---|---|---|
| $I_{DD(RES)}$ | Supply current of resonator oscillator[2)3)] | | See Section 12.5.3 on page 186 | | µA |
| $I_{DD(PLL)}$ | PLL supply current | $V_{DD}$ = 5V | 360 | | |
| $I_{DD(LVD)}$ | LVD supply current | HALT mode, $V_{DD}$ = 5V | 150 | 300 | |

**Notes:**

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

**Figure 126. 32-Pin Low Profile Quad Flat Package (7x7)**




| Dim. | mm | | | inches[1] | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| A | | | 1.60 | | | 0.0630 |
| A1 | 0.05 | | 0.15 | 0.0020 | | 0.0059 |
| A2 | 1.35 | 1.40 | 1.45 | 0.0531 | 0.0551 | 0.0571 |
| b | 0.30 | 0.37 | 0.45 | 0.0118 | 0.0146 | 0.0177 |
| C | 0.09 | | 0.20 | 0.0035 | | 0.0079 |
| D | | 9.00 | | | 0.3543 | |
| D1 | | 7.00 | | | 0.2756 | |
| E | | 9.00 | | | 0.3543 | |
| E1 | | 7.00 | | | 0.2756 | |
| e | | 0.80 | | | 0.0315 | |
| θ | 0° | 3.5° | 7° | 0° | 3.5° | 7° |
| L | 0.45 | 0.60 | 0.75 | 0.0177 | 0.0236 | 0.0295 |
| L1 | | 1.00 | | | 0.0394 | |
| | Number of Pins | | | | | |
| N | 32 | | | | | |

Note 1. Values in inches are converted from mm and rounded to 4 decimal digits.



## 13.2 THERMAL CHARACTERISTICS

| Symbol | Ratings | Value | Unit |
|---|---|---|---|
| $R_{thJA}$ | Package thermal resistance (junction to ambient)<br>LQFP64<br>LQFP44<br>LQFP32 | <br>60<br>52<br>70 | °C/W |
| $P_D$ | Power dissipation[1] | 500 | mW |
| $T_{Jmax}$ | Maximum junction temperature[2] | 150 | °C |

**Notes:**

1. The maximum power dissipation is obtained from the formula $P_D = (T_J - T_A) / R_{thJA}$. The power dissipation of an application can be defined by the user with the formula: $P_D = P_{INT} + P_{PORT}$ where $P_{INT}$ is the chip internal power ($I_{DD}$ x $V_{DD}$) and $P_{PORT}$ is the port power dissipation depending on the ports used in the application.
2. The maximum chip-junction temperature is based on technology characteristics.

## 13.3 SOLDERING AND GLUEABILITY INFORMATION

In order to meet environmental requirements, ST offers these devices in ECOPACK® packages. These packages have a lead-free second level interconnect. The category of second level interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label.

ECOPACK is an ST trademark. ECOPACK® specifications are available at www.st.com.

**Figure 129. FASTROM Factory Coded Device Types**

| Part Number | Package | Memory (Kbytes) | RAM (Kbytes) | Temp Range |
|---|---|---|---|---|
| ST72P361AR6T6 | LQFP64 10x10 | 32 | 1.5 | -40 to +85°C |
| ST72P361AR7T6 | | 48 | 2 | |
| ST72P361AR9T6 | | 60 | 2 | |
| ST72P361J6T6 | LQFP44 10x10 | 32 | 1.5 | |
| ST72P361J7T6 | | 48 | 2 | |
| ST72P361J9T6 | | 60 | 2 | |
| ST72P361K6T6 | LQFP32 7x7 | 32 | 1.5 | |
| ST72P361K7T6 | | 48 | 2 | |
| ST72P361K9T6 | | 60 | 2 | |
| ST72P361AR6T3 | LQFP64 10x10 | 32 | 1.5 | -40 to +125°C |
| ST72P361AR7T3 | | 48 | 2 | |
| ST72P361AR9T3 | | 60 | 2 | |
| ST72P361J6T3 | LQFP44 10x10 | 32 | 1.5 | |
| ST72P361J7T3 | | 48 | 2 | |
| ST72P361J9T3 | | 60 | 2 | |
| ST72P361K6T3 | LQFP32 7x7 | 32 | 1.5 | |
| ST72P361K7T3 | | 48 | 2 | |
| ST72P361K9T3 | | 60 | 2 | |