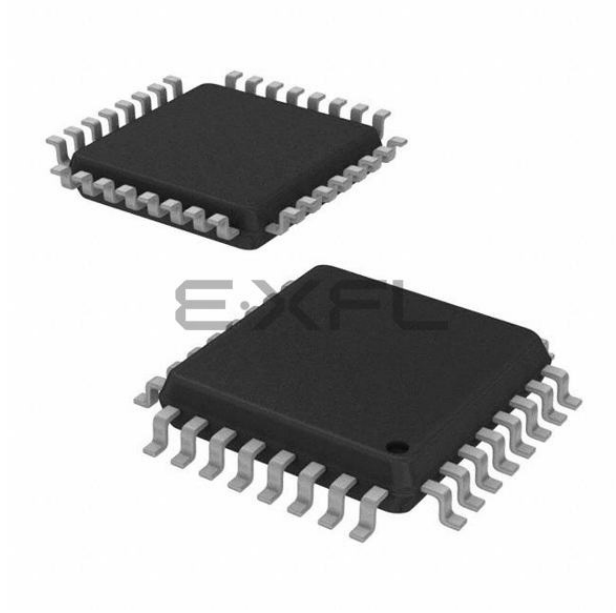


**Welcome to E-XFL.COM**

## 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 | Not For New Designs |
| Core Processor | ST7 |
| Core Size | 8-Bit |
| Speed | 8MHz |
| Connectivity | SCI, SPI |
| Peripherals | LVD, POR, PWM, WDT |
| Number of I/O | 24 |
| Program Memory Size | 8KB (8K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 384 x 8 |
| Voltage - Supply (Vcc/Vdd) | 3.8V ~ 5.5V |
| Data Converters | A/D 12x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 32-LQFP |
| Supplier Device Package | - |
| Purchase URL | https://www.e-xfl.com/product-detail/stmicroelectronics/st72f324bk2t6tr |

# List of figures

**Table 2. Device pin description (continued)**

| Pin | | | Type | Level | | Port | | | | | | Main function (after reset) | Alternate function | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| No. | | Name | | | | Input | | | | Output | | | | |
| LQFP44 | LQFP32 | | | Input | Output | float | wpu | int | ana | OD | PP | | | |
| 26 | 11 | PC3 (HS)/ICAP1_B | I/O | $C_T$ | HS | **X** | X | | | X | X | Port C3 | Timer B input capture 1 | |
| 27 | 12 | PC4/MISO/ICCDATA | I/O | $C_T$ | | **X** | X | | | X | X | Port C4 | SPI master in/slave out data | ICC data input |
| 28 | 13 | PC5/MOSI /AIN14 | I/O | $C_T$ | | **X** | X | | X | X | X | Port C5 | SPI master out/slave in data | ADC analog input 14 |
| 29 | 14 | PC6/SCK /ICCCLK | I/O | $C_T$ | | **X** | X | | | X | X | Port C6 | SPI serial clock | ICC clock output |
| 30 | 15 | PC7/$\overline{SS}$/AIN15 | I/O | $C_T$ | | **X** | X | | X | X | X | Port C7 | SPI slave select (active low) | ADC analog input 15 |
| 31 | 16 | PA3 (HS) | I/O | $C_T$ | HS | **X** | | ei0 | | X | X | Port A3 | | |
| 32 | - | $V_{DD\_1}$(1) | S | | | | | | | | | Digital main supply voltage | | |
| 33 | - | $V_{SS\_1}$(1) | S | | | | | | | | | Digital ground voltage | | |
| 34 | 17 | PA4 (HS) | I/O | $C_T$ | HS | **X** | X | | | X | X | Port A4 | | |
| 35 | - | PA5 (HS) | I/O | $C_T$ | HS | **X** | X | | | X | X | Port A5 | | |
| 36 | 18 | PA6 (HS) | I/O | $C_T$ | HS | **X** | | | | T | | Port A6(2) | | |
| 37 | 19 | PA7 (HS) | I/O | $C_T$ | HS | **X** | | | | T | | Port A7(2) | | |
| 38 | 20 | $V_{PP}$ /ICCSEL | I | | | | | | | | | Must be tied low. In the Flash programming mode, this pin acts as the programming voltage input $V_{PP}$. See *Section 12.10.2* for more details. High voltage must not be applied to ROM devices. | | |
| 39 | 21 | $\overline{RESET}$ | I/O | $C_T$ | | | | | | | | Top priority non-maskable interrupt | | |
| 40 | 22 | $V_{SS\_2}$(1) | S | | | | | | | | | Digital ground voltage | | |
| 41 | 23 | OSC2(3) | O | | | | | | | | | Resonator oscillator inverter output | | |
| 42 | 24 | OSC1(3) | I | | | | | | | | | External clock input or resonator oscillator inverter input | | |
| 43 | 25 | $V_{DD\_2}$(1) | S | | | | | | | | | Digital main supply voltage | | |
| 44 | 26 | PE0/TDO | I/O | $C_T$ | | **X** | X | | | X | X | Port E0 | SCI transmit data out | |
| 1 | 27 | PE1/RDI | I/O | $C_T$ | | **X** | X | | | X | X | Port E1 | SCI receive data in | |

## 4.6 IAP (in-application programming)

This mode uses a BootLoader program previously stored in Sector 0 by the user (in ICP mode or by plugging the device in a programming tool).

This mode is fully controlled by user software. This allows it to be adapted to the user application, (such as user-defined strategy for entering programming mode, choice of communications protocol used to fetch the data to be stored). For example, it is possible to download code from the SPI, SCI, USB or CAN interface and program it in the Flash. IAP mode can be used to program any of the Flash sectors except Sector 0, which is write/erase protected to allow recovery in case errors occur during the programming operation.

## 4.7 Related documentation

For details on Flash programming and ICC protocol, refer to the *ST7 Flash Programming Reference Manual* and to the *ST7 ICC Protocol Reference Manual*.

### 4.7.1 Flash Control/Status Register (FCSR)

This register is reserved for use by programming tool software. It controls the Flash programming and erasing operations.

FCSR Reset value:0000 0000 (00h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 5. Flash control/status register address and reset value**

| Address (Hex) | Register label | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| 0029h | FCSR reset value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### 5.3.5 Stack Pointer register (SP)

SP — Reset value: 01 FFh

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | SP7 | SP6 | SP5 | SP4 | SP3 | SP2 | SP1 | SP0 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

The Stack Pointer is a 16-bit register which is always pointing to the next free location in the stack. It is then decremented after data has been pushed onto the stack and incremented before data is popped from the stack (see *Figure 8*).

Since the stack is 256 bytes deep, the 8 most significant bits are forced by hardware. Following an MCU reset, or after a Reset Stack Pointer instruction (RSP), the Stack Pointer contains its reset value (the SP7 to SP0 bits are set) which is the stack higher address.

The least significant byte of the Stack Pointer (called S) can be directly accessed by an LD instruction.

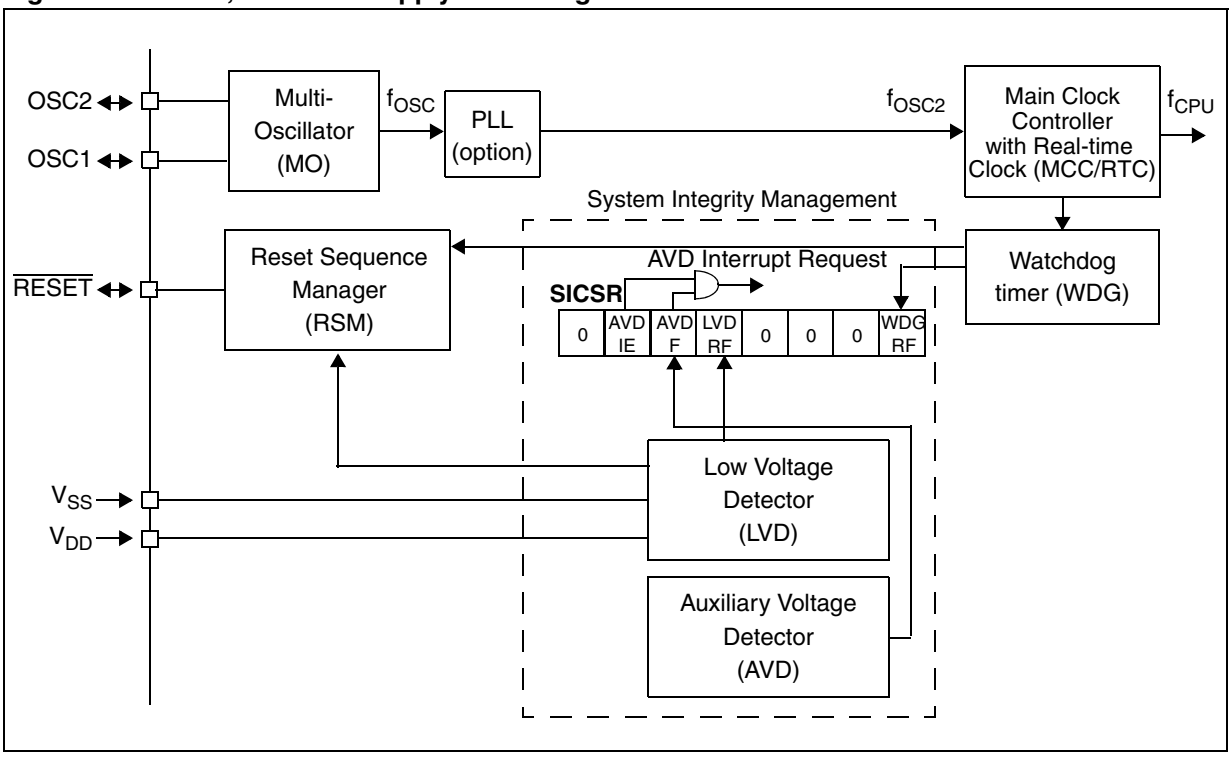
*Note: When the lower limit is exceeded, the Stack Pointer wraps around to the stack upper limit, without indicating the stack overflow. The previously stored information is then overwritten and therefore lost. The stack also wraps in case of an underflow.*

The stack is used to save the return address during a subroutine call and the CPU context during an interrupt. The user may also directly manipulate the stack by means of the PUSH and POP instructions. In the case of an interrupt, the PCL is stored at the first location pointed to by the SP. Then the other registers are stored in the next locations as shown in *Figure 8*.

- When an interrupt is received, the SP is decremented and the context is pushed on the stack.
- On return from interrupt, the SP is incremented and the context is popped from the stack.

A subroutine call occupies two locations and an interrupt five locations in the stack area.

**Figure 10. Clock, reset and supply block diagram**




## 6.3 Multi-oscillator (MO)

The main clock of the ST7 can be generated by three different source types coming from the multi-oscillator block:

- an external source
- 4 crystal or ceramic resonator oscillators
- an internal high frequency RC oscillator

Each oscillator is optimized for a given frequency range in terms of consumption and is selectable through the option byte. The associated hardware configurations are shown in *Table 9*. Refer to the electrical characteristics section for more details.

**Caution:** The OSC1 and/or OSC2 pins must not be left unconnected. For the purposes of Failure Mode and Effect Analysis, it should be noted that if the OSC1 and/or OSC2 pins are left unconnected, the ST7 main oscillator may start and, in this configuration, could generate an $f_{OSC}$ clock frequency in excess of the allowed maximum (> 16 MHz.), putting the ST7 in an unsafe/undefined state. The product behavior must therefore be considered undefined when the OSC pins are left unconnected.

### 6.3.1 External clock source

In this external clock mode, a clock signal (square, sinus or triangle) with ~50% duty cycle has to drive the OSC1 pin while the OSC2 pin is tied to ground.

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 to be serviced) is therefore lost if the clear sequence is executed.*

## 7.3 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 *Table 25: Interrupt mapping*). 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 17*.

*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.*

## 7.4 Concurrent and nested management

*Figure 18* and *Figure 19* 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 19*. The interrupt hardware priority is given in order from the lowest to the highest as follows: MAIN, IT4, IT3, IT2, IT1, IT0. Software priority is given for each interrupt.

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

**Figure 18. Concurrent interrupt management**

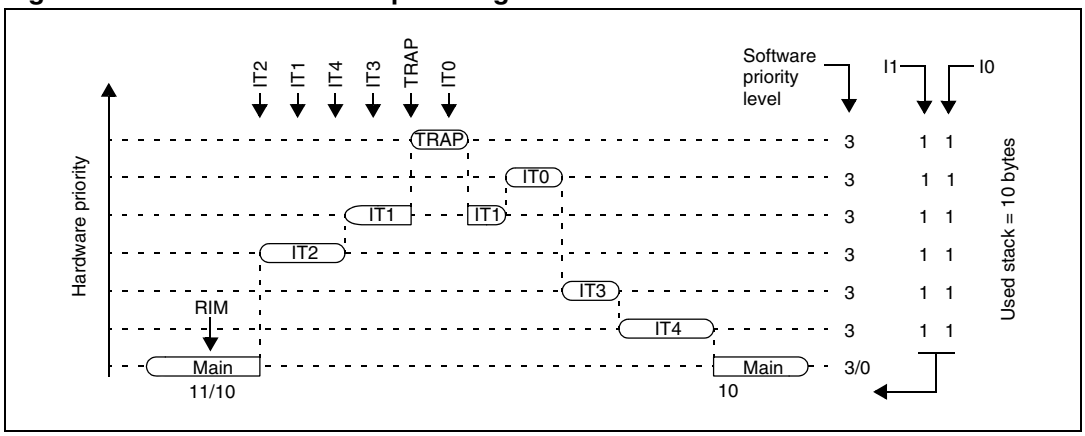


**Table 18. Dedicated interrupt instruction set(1) (continued)**

| Instruction | New description | Function/example | I1 | H | I0 | N | Z | C |
|---|---|---|---|---|---|---|---|---|
| IRET | Interrupt routine return | POP CC, A, X, PC | I1 | H | I0 | N | Z | C |
| JRM | Jump if I1:0=11 (level 3) | I1:0=11 ? | | | | | | |
| JRNM | Jump if I1:0<>11 | I1:0<>11 ? | | | | | | |
| POP CC | POP CC from the Stack | Mem => CC | I1 | H | I0 | N | Z | C |
| RIM | Enable interrupt (level 0 set) | Load 10 in I1:0 of CC | 1 | | 0 | | | |
| SIM | Disable interrupt (level 3 set) | Load 11 in I1:0 of CC | 1 | | 1 | | | |
| TRAP | Software TRAP | Software NMI | 1 | | 1 | | | |
| WFI | WAIT for interrupt | | 1 | | 0 | | | |

1. During the execution of an interrupt routine, the HALT, POP CC, RIM, SIM and WFI instructions change the current software priority up to the next IRET instruction or one of the previously mentioned instructions.

**Figure 32. Exact timeout duration ($t_{min}$ and $t_{max}$)**

**WHERE:**

$t_{min0} = (LSB + 128) \times 64 \times t_{OSC2}$

$t_{max0} = 16384 \times t_{OSC2}$

$t_{OSC2} = 125ns$ if $f_{OSC2} = 8$ MHz

CNT = value of T[5:0] bits in the WDGCR register (6 bits)

MSB and LSB are values from the table below depending on the timebase selected by the TB[1:0] bits in the MCCSR register

| TB1 bit (MCCSR reg.) | TB0 bit (MCCSR reg.) | Selected MCCSR timebase | MSB | LSB |
|---|---|---|---|---|
| 0 | 0 | 2ms | 4 | 59 |
| 0 | 1 | 4ms | 8 | 53 |
| 1 | 0 | 10ms | 20 | 35 |
| 1 | 1 | 25ms | 49 | 54 |

To calculate the minimum Watchdog timeout ($t_{min}$):

**IF** $CNT < \left\lceil \frac{MSB}{4} \right\rceil$ **THEN** $t_{min} = t_{min0} + 16384 \times CNT \times t_{osc2}$

**ELSE** $t_{min} = t_{min0} + \left[ 16384 \times \left( CNT - \left\lceil \frac{4CNT}{MSB} \right\rceil \right) + (192 + LSB) \times 64 \times \left\lceil \frac{4CNT}{MSB} \right\rceil \right] \times t_{osc2}$

To calculate the maximum Watchdog timeout ($t_{max}$):

**IF** $CNT \leq \left\lceil \frac{MSB}{4} \right\rceil$ **THEN** $t_{max} = t_{max0} + 16384 \times CNT \times t_{osc2}$

**ELSE** $t_{max} = t_{max0} + \left[ 16384 \times \left( CNT - \left\lceil \frac{4CNT}{MSB} \right\rceil \right) + (192 + LSB) \times 64 \times \left\lceil \frac{4CNT}{MSB} \right\rceil \right] \times t_{osc2}$

**NOTE:** In the above formulae, division results must be rounded down to the next integer value.

**EXAMPLE**: With 2ms timeout selected in MCCSR register

| Value of T[5:0] bits in WDGCR register (Hex.) | Min. Watchdog timeout (ms) $t_{min}$ | Max. Watchdog timeout (ms) $t_{max}$ |
|---|---|---|
| 00 | 1.496 | 2.048 |
| 3F | 128 | 128.552 |

### Control/Status Register (CSR)

CSR Reset value: xxxx x0xx (xxh)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ICF1 | OCF1 | TOF | ICF2 | OCF2 | TIMD | Reserved | |
| RO | RO | RO | RO | RO | R/W | - | |

**Table 51. CSR register description**

| Bit | Name | Function |
|---|---|---|
| 7 | ICF1 | Input Capture Flag 1<br>0: No Input Capture (reset value).<br>1: An Input Capture has occurred on the ICAP1 pin or the counter has reached the OC2R value in PWM mode. To clear this bit, first read the SR register, then read or write the low byte of the IC1R (IC1LR) register. |
| 6 | OCF1 | Output Compare Flag 1<br>0: No match (reset value).<br>1: The content of the free running counter has matched the content of the OC1R register. To clear this bit, first read the SR register, then read or write the low byte of the OC1R (OC1LR) register. |
| 5 | TOF | Timer Overflow Flag<br>0: No timer overflow (reset value).<br>1: The free running counter rolled over from FFFFh to 0000h. To clear this bit, first read the SR register, then read or write the low byte of the CR (CLR) register.<br>*Note: Reading or writing the ACLR register does not clear TOF.* |
| 4 | ICF2 | Input Capture Flag 2<br>0: No input capture (reset value).<br>1: An Input Capture has occurred on the ICAP2 pin. To clear this bit, first read the SR register, then read or write the low byte of the IC2R (IC2LR) register. |
| 3 | OCF2 | Output Compare Flag 2<br>0: No match (reset value).<br>1: The content of the free running counter has matched the content of the OC2R register. To clear this bit, first read the SR register, then read or write the low byte of the OC2R (OC2LR) register. |
| 2 | TIMD | Timer Disable<br>This bit is set and cleared by software. When set, it freezes the timer prescaler and counter and disabled the output functions (OCMP1 and OCMP2 pins) to reduce power consumption. Access to the timer registers is still available, allowing the timer configuration to be changed, or the counter reset, while it is disabled.<br>0: Timer enabled.<br>1: Timer prescaler, counter and outputs disabled. |
| 1:0 | - | Reserved, must be kept cleared. |

# 10.6 10-bit A/D converter (ADC)

## 10.6.1 Introduction

The on-chip analog-to-digital converter (ADC) peripheral is a 10-bit, successive approximation converter with internal sample and hold circuitry. This peripheral has up to 16 multiplexed analog input channels (refer to device pin out description) that allow the peripheral to convert the analog voltage levels from up to 16 different sources.

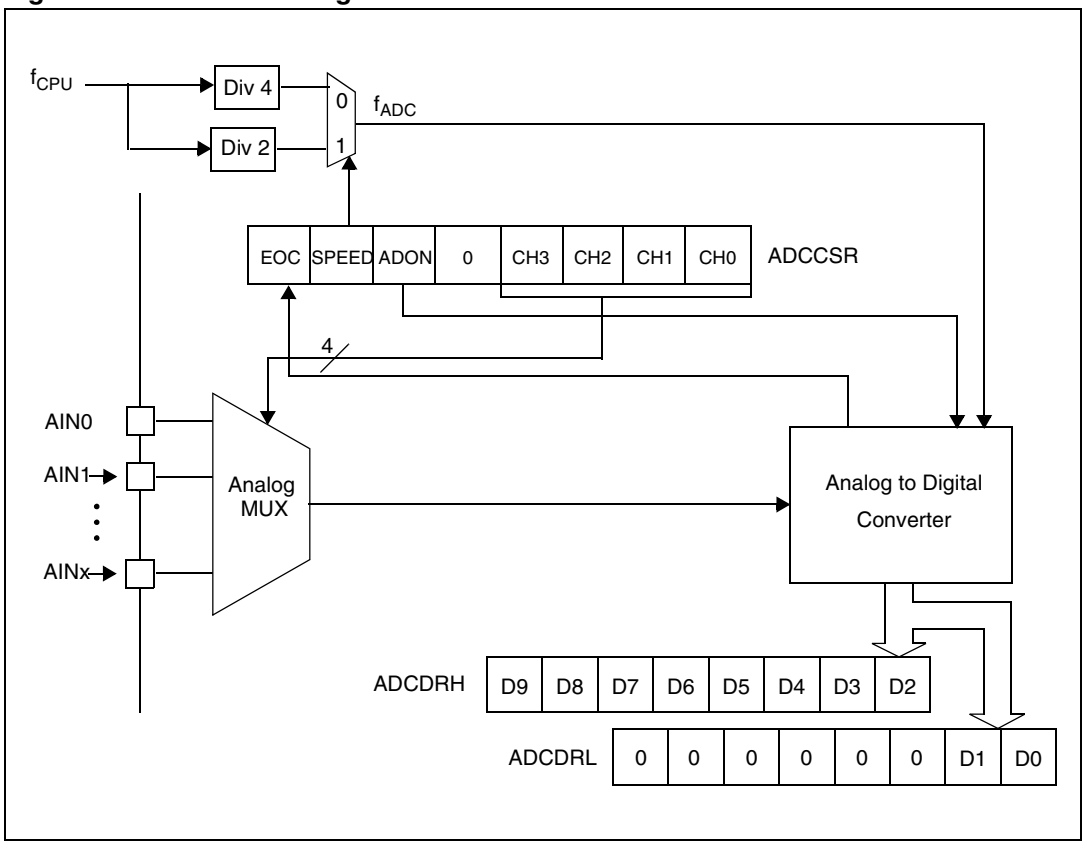
The result of the conversion is stored in a 10-bit Data Register. The A/D converter is controlled through a Control/Status Register.

## 10.6.2 Main features

- 10-bit conversion
- Up to 16 channels with multiplexed input
- Linear successive approximation
- Data register (DR) which contains the results
- Conversion complete status flag
- On/off bit (to reduce consumption)

The block diagram is shown in *Figure 59*.

**Figure 59. ADC block diagram**

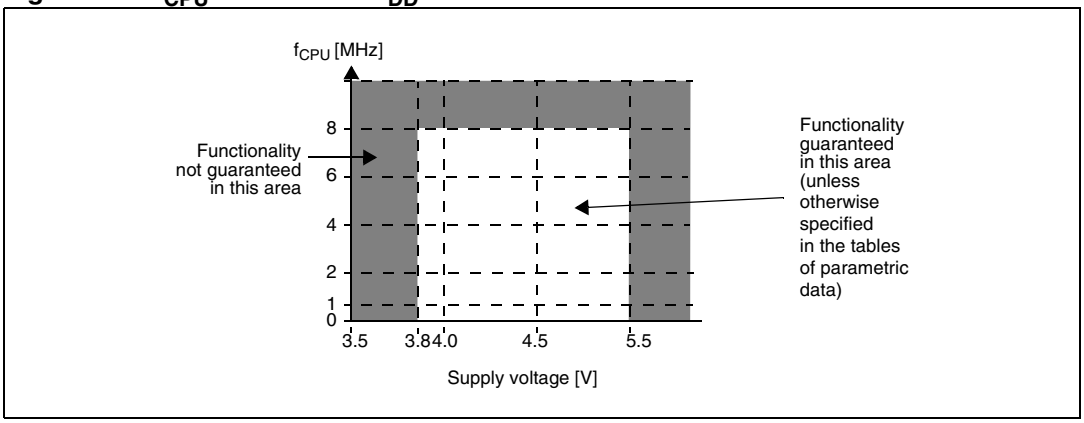


## 12.3 Operating conditions

**Table 86. Operating conditions**

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $f_{CPU}$ | Internal clock frequency | | 0 | 8 | MHz |
| $V_{DD}$ | Operating voltage (except Flash Write/Erase) | | 3.8 | 5.5 | V |
| | Operating Voltage for Flash Write/Erase | $V_{PP}$ = 11.4 to 12.6V | 4.5 | 5.5 | |
| $T_A$ | Ambient temperature range | A-suffix versions | -40 | 85 | °C |
| | | B-suffix versions | | 105 | |
| | | C-suffix version | | 125 | |
| | | D-suffix version | | 150 | |

**Figure 62. $f_{CPU}$ max versus $V_{DD}$**




*Note: Some temperature ranges are only available with a specific package and memory size. Refer to Section 14: Device configuration and ordering information.*

**Warning: Do not connect 12V to $V_{PP}$ before $V_{DD}$ is powered on, as this may damage the device.**

## 12.4 LVD/AVD characteristics

### 12.4.1 Operating conditions with LVD

Subject to general operating conditions for $T_A$.

**Table 87. Operating conditions with LVD**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IT+(LVD)}$ | Reset release threshold ($V_{DD}$ rise) | VD level = high in option byte | 4.0(1) | 4.2 | 4.5 | V |
| | | VD level = med. in option byte(2) | 3.55(1) | 3.75 | 4.0(1) | |
| | | VD level = low in option byte(2) | 2.95(1) | 3.15 | 3.35(1) | |
| $V_{IT-(LVD)}$ | Reset generation threshold ($V_{DD}$ fall) | VD level = high in option byte | 3.8 | 4.0 | 4.25(1) | |
| | | VD level = med. in option byte(2) | 3.35(1) | 3.55 | 3.75(1) | |
| | | VD level = low in option byte(2) | 2.8(1) | 3.0 | 3.15(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(1) | Flash devices | 6µs/V | | 100ms/V | |
| | | 8/16K ROM devices | | | 20ms/V | |
| | | 32K ROM devices | | | ∝ ms/V | |
| $t_{g(VDD)}$ | Filtered glitch delay on $V_{DD}$(1) | Not detected by the LVD | | | 40 | ns |

1. Data based on characterization results, tested in production for ROM devices only.
2. If the medium or low thresholds are selected, the detection may occur outside the specified operating voltage range.

### 12.4.2 Auxiliary voltage detector (AVD) thresholds

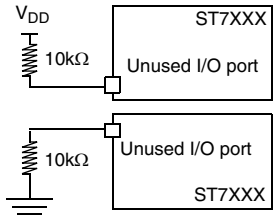
Subject to general operating conditions for $T_A$.

**Table 88. AVD thresholds**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IT+(AVD)}$ | 1 ⇒ 0 AVDF flag toggle threshold ($V_{DD}$ rise) | VD level = high in option byte | 4.4(1) | 4.6 | 4.9 | V |
| | | VD level = med. in option byte | 3.95(1) | 4.15 | 4.4(1) | |
| | | VD level = low in option byte | 3.4(1) | 3.6 | 3.8(1) | |
| $V_{IT-(AVD)}$ | 0 ⇒ 1 AVDF flag toggle threshold ($V_{DD}$ fall) | VD level = high in option byte | 4.2 | 4.4 | 4.65(1) | |
| | | VD level = med. in option byte | 3.75(1) | 4.0 | 4.2(1) | |
| | | VD level = low in option byte | 3.2(1) | 3.4 | 3.6(1) | |
| $V_{hys(AVD)}$ | AVD voltage threshold hysteresis | $V_{IT+(AVD)}$-$V_{IT-(AVD)}$ | | 200 | | mV |
| $\Delta V_{IT-}$ | Voltage drop between AVD flag set and LVD reset activated | $V_{IT-(AVD)}$-$V_{IT-(LVD)}$ | | 450 | | |

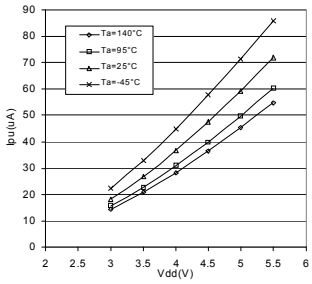
1. Data based on characterization results, tested in production for ROM devices only.

**Figure 68. Unused I/O pins configured as input(1)**




1. I/O can be left unconnected if it is configured as output (0 or 1) by the software. This has the advantage of greater EMC robustness and lower cost.

**Figure 69. Typical $I_{PU}$ vs. $V_{DD}$ with $V_{IN} = V_{SS}$**




## 12.9.2 Output driving current

Subject to general operating conditions for $V_{DD}$, $f_{CPU}$, and $T_A$ unless otherwise specified.

**Table 106. Output driving current**

| Symbol | Parameter | Conditions | | Min | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{OL}$(1) | Output low level voltage for a standard I/O pin when 8 pins are sunk at same time (see *Figure 70*) | $V_{DD}$ = 5V | $I_{IO}$ = +5mA | | 1.2 | V |
| | | | $I_{IO}$ = +2mA | | 0.5 | |
| | Output low level voltage for a high sink I/O pin when 4 pins are sunk at same time (see *Figure 71* and *Figure 73*) | | $I_{IO}$ = +20mA<br>$T_A \leq 85°$<br>$T_A > 85°C$ | | 1.3<br>1.5 | |
| | | | $I_{IO}$ = +8mA | | 0.6 | |
| $V_{OH}$(2) | Output high level voltage for an I/O pin when 4 pins are sourced at same time (see *Figure 72* and *Figure 75*) | | $I_{IO}$ = -5mA,<br>$T_A \leq 85°C$<br>$T_A > 85°C$ | <br>$V_{DD}$ - 1.4<br>$V_{DD}$ - 1.6 | | |
| | | | $I_{IO}$ = -2mA | $V_{DD}$ - 0.7 | | |

1. The $I_{IO}$ current sunk must always respect the absolute maximum rating specified in *Section 12.2.2* and the sum of $I_{IO}$ (I/O ports and control pins) must not exceed $I_{VSS}$.
2. The $I_{IO}$ current sourced must always respect the absolute maximum rating specified in *Section 12.2.2* and the sum of $I_{IO}$ (I/O ports and control pins) must not exceed $I_{VDD}$. True open drain I/O pins do not have $V_{OH}$.

## 12.10 Control pin characteristics

### 12.10.1 Asynchronous $\overline{\text{RESET}}$ pin

Subject to general operating conditions for $V_{DD}$, $f_{CPU}$, and $T_A$ unless otherwise specified.

**Table 107. Asynchronous $\overline{\text{RESET}}$ pin**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low level voltage(1) | | | | $0.3xV_{DD}$ | V |
| $V_{IH}$ | Input high level voltage(1) | | $0.7xV_{DD}$ | | | |
| $V_{hys}$ | Schmitt trigger voltage hysteresis(2) | | | 2.5 | | |
| $V_{OL}$ | Output low level voltage(3) | $V_{DD}$ = 5V, $I_{IO}$ = +2mA | | 0.2 | 0.5 | V |
| $I_{IO}$ | Driving current on $\overline{\text{RESET}}$ pin | | | 2 | | mA |
| $R_{ON}$ | Weak pull-up equivalent resistor | $V_{DD}$ = 5V | 20 | 30 | 120 | kΩ |
| $t_{w(RSTL)out}$ | Generated reset pulse duration | Internal reset sources | 20 | 30 | 42(4) | µs |
| $t_{h(RSTL)in}$ | External reset pulse hold time(5) | | 2.5 | | | µs |
| $t_{g(RSTL)in}$ | Filtered glitch duration(6) | | | 200 | | ns |

1. Data based on characterization results, not tested in production.
2. Hysteresis voltage between Schmitt trigger switching levels.
3. The $I_{IO}$ current sunk must always respect the absolute maximum rating specified in *Section 12.2.2* and the sum of $I_{IO}$ (I/O ports and control pins) must not exceed $I_{VSS}$.
4. Data guaranteed by design, not tested in production.
5. To guarantee the reset of the device, a minimum pulse has to be applied to the $\overline{\text{RESET}}$ pin. All short pulses applied on the $\overline{\text{RESET}}$ pin with a duration below $t_{h(RSTL)in}$ can be ignored.
6. The reset network (the resistor and two capacitors) protects the device against parasitic resets, especially in noisy environments.

## 12.12 Communication interface characteristics

### 12.12.1 Serial peripheral interface (SPI)

The following characteristics are ubject to general operating conditions for $V_{DD}$, $f_{CPU}$, and $T_A$ unless otherwise specified. The data is based on design simulation and/or characterization results, not tested in production.

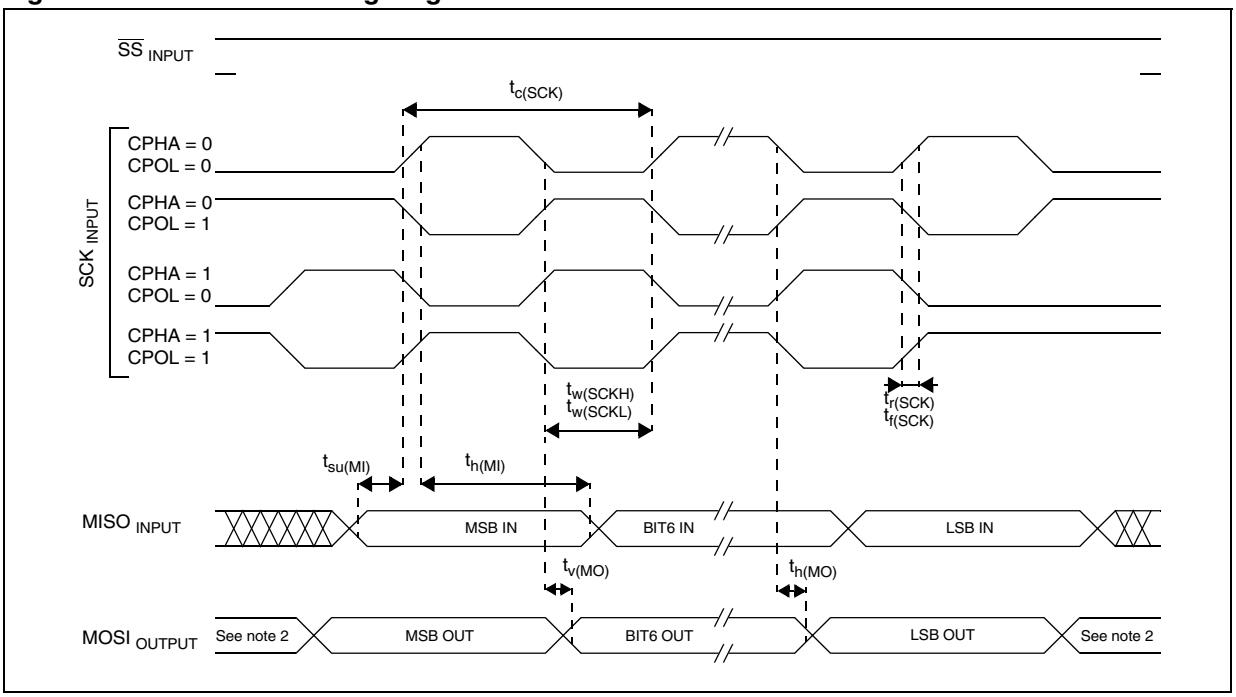
When no communication is on-going the data output line of the SPI (MOSI in master mode, MISO in slave mode) has its alternate function capability released. In this case, the pin status depends on the I/O port configuration. Refer to the I/O port characteristics for more details on the input/output alternate function characteristics ($\overline{SS}$, SCK, MOSI, MISO).

**Table 110. SPI characteristics**

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $f_{SCK}$<br>$1/t_{c(SCK)}$ | SPI clock frequency | Master $f_{CPU}$ = 8 MHz | $f_{CPU}/128 = 0.0625$ | $f_{CPU}/4 = 2$ | MHz |
| | | Slave $f_{CPU}$ = 8 MHz | 0 | $f_{CPU}/2 = 4$ | |
| $t_{r(SCK)}$<br>$t_{f(SCK)}$ | SPI clock rise and fall time | | see I/O port pin description | | |
| $t_{su(\overline{SS})}$ (1) | $\overline{SS}$ setup time(2) | Slave | $t_{CPU}$ + 50 | | ns |
| $t_{h(\overline{SS})}$ (1) | $\overline{SS}$ hold time | Slave | 120 | | |
| $t_{w(SCKH)}$ (1)<br>$t_{w(SCKL)}$ (1) | SCK high and low time | Master<br>Slave | 100<br>90 | | |
| $t_{su(MI)}$ (1)<br>$t_{su(SI)}$ (1) | Data input setup time | Master<br>Slave | 100<br>100 | | |
| $t_{h(MI)}$ (1)<br>$t_{h(SI)}$ (1) | Data input hold time | Master<br>Slave | 100<br>100 | | |
| $t_{a(SO)}$ (1) | Data output access time | Slave | 0 | 120 | |
| $t_{dis(SO)}$ (1) | Data output disable time | Slave | | 240 | |
| $t_{v(SO)}$ (1) | Data output valid time | Slave (after enable edge) | | 120 | |
| $t_{h(SO)}$ (1) | Data output hold time | | 0 | | |
| $t_{v(MO)}$ (1) | Data output valid time | Master (after enable edge) | | 120 | |
| $t_{h(MO)}$ (1) | Data output hold time | | 0 | | |

1. Data based on design simulation and/or characterization results, not tested in production.
2. Depends on $f_{CPU}$. For example, if $f_{CPU}$ = 8 MHz, then $t_{CPU} = 1 / f_{CPU}$ = 125ns and $t_{su(SS)}$ = 175ns.

**Figure 81. SPI master timing diagram(1)**




1. Measurement points are done at CMOS levels: $0.3xV_{DD}$ and $0.7xV_{DD}$.
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.

## 12.13 10-bit ADC characteristics

Subject to general operating conditions for $V_{DD}$, $f_{CPU}$, and $T_A$ unless otherwise specified.

**Table 111. 10-bit ADC characteristics**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $f_{ADC}$ | ADC clock frequency | | 0.4 | | 2 | MHz |
| $V_{AREF}$ | Analog reference voltage | $0.7*V_{DD} \le V_{AREF} \le V_{DD}$ | 3.8 | | $V_{DD}$ | V |
| $V_{AIN}$ | Conversion voltage range(1) | | $V_{SSA}$ | | $V_{AREF}$ | |
| $I_{lkg}$ | Input leakage current for analog input(2) | $-40°C \le T_A \le +85°C$ | | | ±250 | nA |
| | | Other $T_A$ ranges | | | ±1 | µA |
| $R_{AIN}$ | External input impedance | | | | See *Figure 82* and *Figure 83* | kΩ |
| $C_{AIN}$ | External capacitor on analog input | | | | | pF |
| $f_{AIN}$ | Variation freq. of analog input signal | | | | | Hz |
| $C_{ADC}$ | Internal sample and hold capacitor | | | 12 | | pF |

**Table 120. STMicroelectronics development tools**

| Supported products | Emulation | | | | Programming |
|---|---|---|---|---|---|
| | ST7 DVP3 series | | ST7 EMU3 series | | ICC socket board |
| | Emulator | Connection kit | Emulator | Active probe and TEB | |
| ST72324BJ, ST72F324BJ | ST7MDT20-DVP3 | ST7MDT20-T44/DVP | ST7MDT20J-EMU3 | ST7MDT20J-TEB | ST7SB20J/xx(1) |
| ST72324BK, ST72F324BK | | ST7MDT20-T32/DVP | | | |

1. Add suffix /EU, /UK, /US for the power supply of your region.

### 14.3.5 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 121*.

*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 (www.yamaichi.de for LQFP44 10x10 and www.ironwoodelectronics.com for LQFP32 7x7).

**Table 121. Suggested list of socket types**

| Device | Socket (supplied with ST7MDT20J-EMU3) | Emulator adapter (supplied with ST7MDT20J-EMU3) |
|---|---|---|
| LQFP32 7X7 | IRONWOOD SF-QFE32SA-L-01 | IRONWOOD SK-UGA06/32A-01 |
| LQFP44 10X10 | YAMAICHI IC149-044-*52-*5 | YAMAICHI ICP-044-5 |

## 14.4 ST7 Application notes

All relevant ST7 application notes can be found on www.st.com.

**Case 1:** Writing to PxOR or PxDDR with global interrupts enabled:

```
LD A,#01
LD sema,A; set the semaphore to '1'
LD A,PFDR
AND A,#02
LD  X,A; store the level before writing to PxOR/PxDDR
LD A,#$90
LD PFDDR,A ; Write to PFDDR
LD A,#$ff
LD  PFOR,A ; Write to PFOR
LD A,PFDR
AND A,#02
LD Y,A; store the level after writing to PxOR/PxDDR
LD A,X; check for falling edge
cp A,#02
jrne OUT
TNZ Y
jrne OUT
LD A,sema ; check the semaphore status if edge is detected
CP A,#01
jrne OUT
call call_routine ; call the interrupt routine
OUT:LD A,#00
LD sema,A
.call_routine ; entry to call_routine
PUSH A
PUSH X
PUSH CC
.ext1_rt ; entry to interrupt routine
LD A,#00
LD sema,A
IRET
```

**Case 2:** Writing to PxOR or PxDDR with global interrupts disabled:

```
SIM ; set the interrupt mask
LD A,PFDR
AND A,#$02
LD  X,A ; store the level before writing to PxOR/PxDDR
LD A,#$90
LD PFDDR,A ; Write into PFDDR
LD A,#$ff
LD PFOR,A ; Write to PFOR
LD A,PFDR
AND A,#$02
LD Y,A ; store the level after writing to PxOR/PxDDR
LD A,X ; check for falling edge
cp A,#$02
jrne OUT
TNZ Y
jrne OUT
LD A,#$01
LD sema,A ; set the semaphore to '1' if edge is detected
```

# 16 Revision history

**Table 123. Document revision history**

| Date | Revision | Changes |
|---|---|---|
| 23-May-2007 | 1 | Initial release |
| 23-Jul-2007 | 2 | Replaced ST72324B-Auto with ST72324Bxx-Auto in document title on cover page.<br>*1 analog peripheral (low current coupling) on page 1*: Replaced '12 robust input ports' with '12 input ports'<br>*Table 1: Device summary on page 1*: Corrected order of listed packages<br>Added *Section 1.2: Differences between ST72324B-Auto and ST72324B datasheets on page 16*<br>*Figure 2: 44-pin LQFP package pinout on page 17*: Displayed port numbers for pins 18 and 20 (port numbers were hidden due to formatting error)<br>*Table 2: Device pin description on page 18*:<br>- replaced $V_{DDA}$ with $V_{REF}$ in *Note 1*<br>- modified *Note 2*<br>*Section 5.3.4: Condition Code register (CC) on page 29*: Replaced IxSPR with ISPRx<br>*Section 9.5.1: I/O port implementation on page 65*: Removed following tables:<br>- Standard ports PA5:4, PC7:0, PD5:0, PE1:0, PF7:6, 4<br>- Interrupt ports PB4, PB2:0, PF2:0 (with pull-up)<br>- Interrupt ports PA3, PB3 (without pull-up)<br>- *True open drain ports PA7:6* (configurations in these four tables already exist in *Table 32: Port configuration*)<br>*Section 12.6.3: Crystal and ceramic resonator oscillators*: Replaced two tables *Crystal and ceramic resonator oscillators (8/16K Flash and ROM devices)* and *Crystal and ceramic resonator oscillators (32 Kbyte Flash and ROM devices)* with single *Table 95: Crystal and ceramic resonator oscillators on page 156*<br>*Table 96: OSCRANGE selection for typical resonators on page 157*: Deleted footnote detailing SMD and LEAD which was linked to 'Reference' column header<br>*Table 102: EMI emissions on page 161*:<br>- added LQFP32 package to all listed devices<br>- changed values for 32 Kbyte ROM devices<br>*Table 105: General characteristics on page 163*:<br>- modified *Note 5*<br>- modified *Note 6*<br>*Figure 76: RESET pin protection when LVD is enabled(1)(2)(3)(4)(5)(6) on page 168*: Replaced 'MW' with 'M ohm' in footnotes to correct formatting error<br>*Table 111: 10-bit ADC characteristics on page 172*: Modified input current leakage parameter and added *Note 2*<br>*Table 112: ADC accuracy on page 175*:<br>- added conditions to total unadjusted error, to offset error and to gain error<br>- modified *Note 2* |

**Table 123. Document revision history (continued)**

| Date | Revision | Changes |
|---|---|---|
| 23-Jul-2007 | 2 (cont'd) | *Table 121: Flash user programmable device types on page 189*:<br>- added footnote to order code column<br>- modified order codes<br>- replaced R with TR for tape and reel in order codes<br>*Figure 89: Flash commercial product code structure on page 183*:<br>- replaced R with TR for tape and reel<br>- changed presentation of temperature ranges<br>*Section 14.2: ROM device ordering information and transfer of customer code on page 184*: Added links to option list, to *Table 122* and to *Table 123*<br>*Table 122: FASTROM factory coded device types on page 191*:<br>- added footnote to order code column<br>- modified order codes<br>*Figure 90: FASTROM commercial product code structure on page 184*: Changed presentation of temperature ranges<br>*Table 123: ROM factory coded device types on page 192*:<br>- added footnote to order code column<br>- modified order codes<br>*Figure 91: ROM commercial product code structure on page 184*: s<br>- changed title<br>- changed presentation of temperature range<br>*ST72324B-Auto Microcontroller FASTROM/ROM Option List on page 185*:<br>- replaced ST72324B with ST72324B-Auto in title<br>- grouped device code characters defining pinout and memory size in parentheses<br>- modified special marking max characters allowed |
| 10-Jun-2010 | 3 | Removed section covering differences between automotive and standard devices.<br>*Table 86: Operating conditions on page 148*:<br>-added D temperature range<br>*Section 12.8.3: Absolute maximum ratings (electrical sensitivity) on page 160*<br>- standard microcontrollers: HB and CDM models specified only<br>- automotive microcontrollers: plus an additional test of MM<br>*Figure 90: ST72F324Bxx-Auto Flash commercial product structure on page 182* and *Figure 91: ST72P324Bxx-Auto FastROM commercial product structure on page 184*:<br>- modified figure to reflect leadfree package in Catania (from E to S).<br>- modified tape and reel symbol from R to X or TX.<br>-Table 121.Flash user programmable device removed.<br>-Table 122.FASTROM factory coded device removed.<br>*Figure 92: ST72324Bxx-Auto ROM commercial product structure on page 185*:<br>- modified figure to reflect leadfree package in Catania (from E to S).<br>- modified tape and reel symbol from R to X or TX.<br>- added D temperature range.<br>-Table 123.ROM factory coded device removed.<br>Option List ordering sheet: added D temperature range |