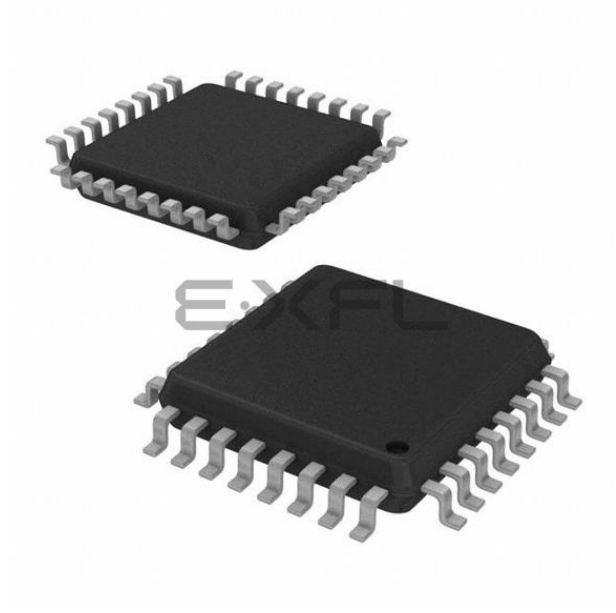


**Welcome to E-XFL.COM**

## 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 | 32KB (32K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 1K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3.8V ~ 5.5V |
| Data Converters | A/D 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/st72f324bk6t6tr |

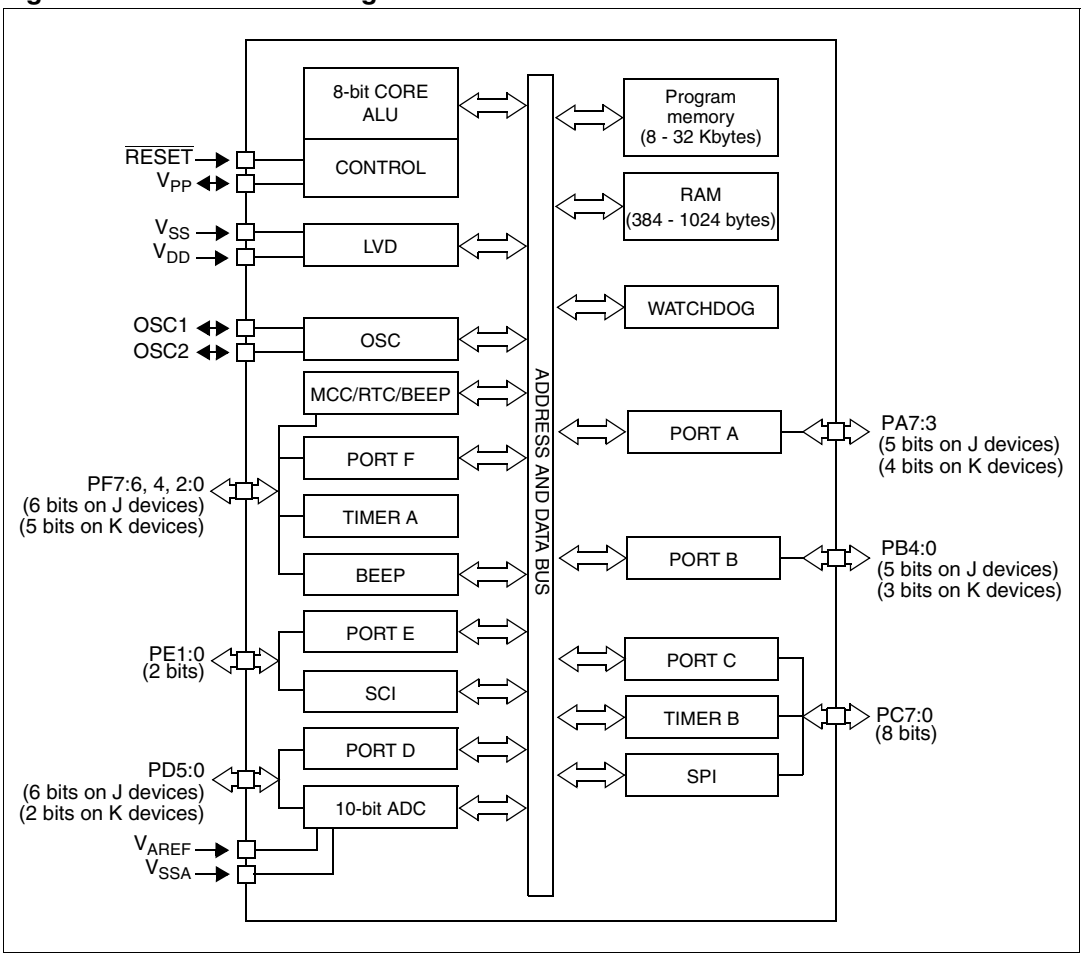
# 1 Description

The ST72324B-Auto devices are members of the ST7 microcontroller family designed for mid-range automotive applications running from 3.8 to 5.5V. Different package options offer up to 32 I/O pins.

All devices are based on a common industry-standard 8-bit core, featuring an enhanced instruction set and are available with Flash or ROM program memory. The ST7 family architecture offers both power and flexibility to software developers, enabling the design of highly efficient and compact application code.

The on-chip peripherals include an A/D converter, two general purpose timers, an SPI interface and an SCI interface. For power economy, the microcontroller can switch dynamically into, Slow, Wait, Active Halt or Halt mode when the application is in idle or stand-by state.

**Figure 1. Device block diagram**




Typical applications include

- all types of car body applications such as window lift, DC motor control, rain sensors
- safety microcontroller in airbag and engine management applications
- auxiliary functions in car radios

### 5.3.1 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.

### 5.3.2 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.

### 5.3.3 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).

### 5.3.4 Condition Code register (CC)

The 8-bit Condition Code register contains the interrupt masks and four flags representative of the result of the instruction just executed. This register can also be handled by the PUSH and POP instructions. These bits can be individually tested and/or controlled by specific instructions.

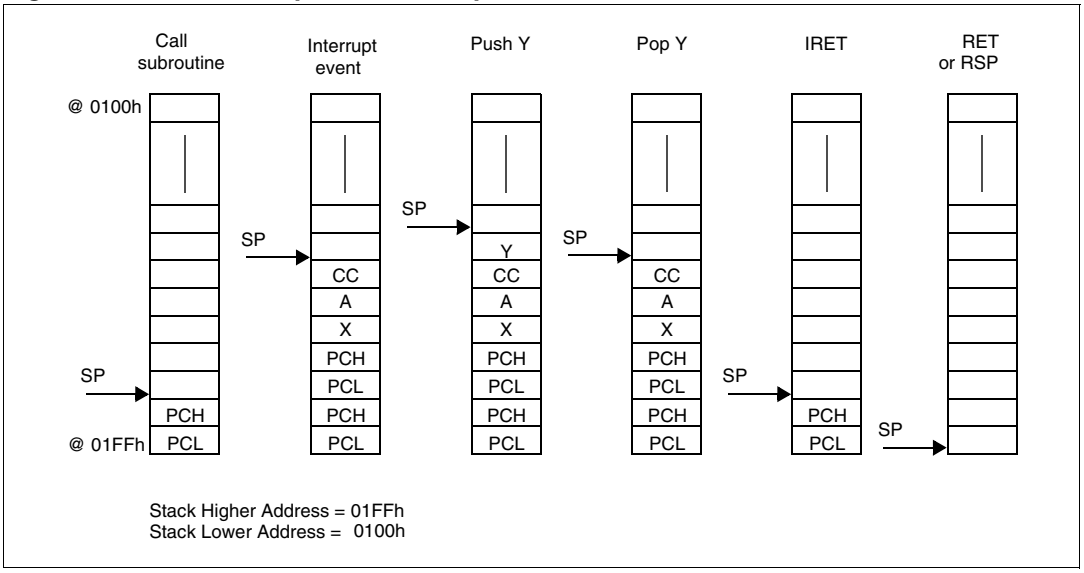
CC Reset value: 111x1xxx

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 1 | 1 | I1 | H | I0 | N | Z | C |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 6. Arithmetic management bits**

| Bit | Name | Function |
|---|---|---|
| 4 | H | Half carry<br>This bit is set by hardware when a carry occurs between bits 3 and 4 of the ALU during an ADD or ADC instructions. It is reset by hardware during the same instructions.<br>0: No half carry has occurred.<br>1: A half carry has occurred.<br>This bit is tested using the JRH or JRNH instruction. The H bit is useful in BCD arithmetic subroutines. |
| 2 | N | Negative<br>This bit is set and cleared by hardware. It is representative of the result sign of the last arithmetic, logical or data manipulation. It is a copy of the result 7th bit.<br>0: The result of the last operation is positive or null.<br>1: The result of the last operation is negative (that is, the most significant bit is a logic 1.<br>This bit is accessed by the JRMI and JRPL instructions. |

**Figure 8. Stack manipulation example**



## 6.4 Reset sequence manager (RSM)

The reset sequence manager includes three reset sources as shown in *Figure 12*:

- External reset source pulse
- Internal LVD reset
- 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 11*:

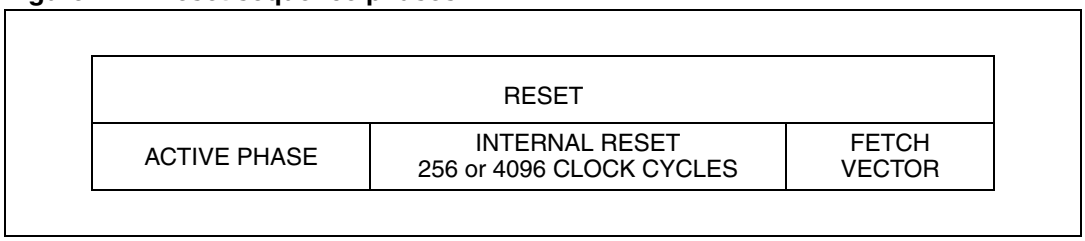
- Active Phase depending on the reset source
- 256 or 4096 CPU clock cycle delay (selected by option byte)
- Reset vector fetch

**Caution:** When the ST7 is unprogrammed or fully erased, the Flash is blank and the RESET vector is not programmed. For this reason, it is recommended to keep the RESET pin in low state until programming mode is entered, in order to avoid unwanted behavior.

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.

The reset vector fetch phase duration is two clock cycles.

**Figure 11. Reset sequence phases**




### 6.4.1 Asynchronous external $\overline{\text{RESET}}$ pin

The $\overline{\text{RESET}}$ pin is both an input and an open-drain output with integrated $R_{ON}$ weak pull-up resistor. This pull-up has no fixed value but varies in accordance with the input voltage. It can be pulled low by external circuitry to reset the device. See the *Electrical characteristics* section for more details.

A reset signal originating from an external source must have a duration of at least $t_{h(RSTL)in}$ in order to be recognized (see *Figure 13*). This detection is asynchronous and therefore the MCU can enter reset state even in Halt mode.

Provided the minimum $V_{DD}$ value (guaranteed for the oscillator frequency) is above $V_{IT-}$, the MCU can only be in two modes:

- under full software control
- in static safe reset

In these conditions, secure operation is always ensured for the application without the need for external reset hardware.

During an LVD reset, the $\overline{\text{RESET}}$ pin is held low, thus permitting the MCU to reset other devices.

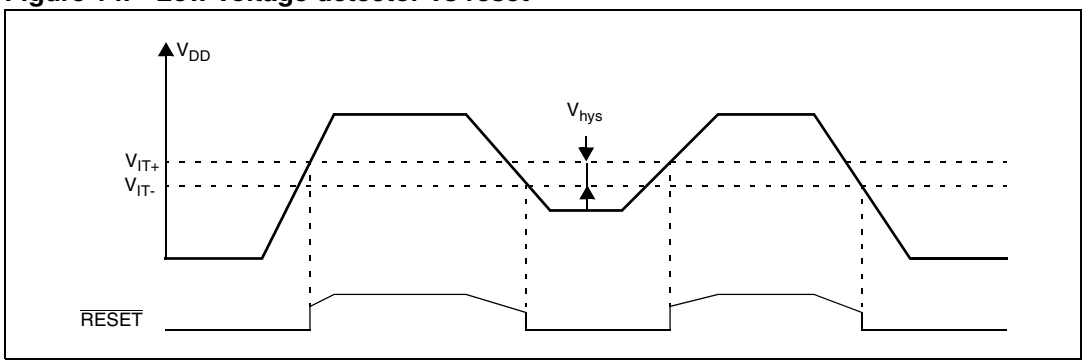
*Note: 1 The LVD allows the device to be used without any external reset circuitry.*

*2 If the medium or low thresholds are selected, the detection may occur outside the specified operating voltage range. Below 3.8V, device operation is not guaranteed.*

*3 The LVD is an optional function which can be selected by option byte.*

*4 It is recommended to make sure that the $V_{DD}$ supply voltage rises monotonously when the device is exiting from reset, to ensure the application functions properly.*

**Figure 14. Low voltage detector vs reset**




## 6.5.2 AVD (auxiliary voltage detector)

The AVD is based on an analog comparison between a $V_{IT-(AVD)}$ and $V_{IT+(AVD)}$ reference value and the $V_{DD}$ main supply. The $V_{IT-}$ reference value for falling voltage is lower than the $V_{IT+}$ reference value for rising voltage in order to avoid parasitic detection (hysteresis).

The output of the AVD comparator is directly readable by the application software through a real-time status bit (AVDF) in the SICSR register. This bit is read only.

**Caution:** The AVD function is active only if the LVD is enabled through the option byte (see *Section 14.1 on page 179*).

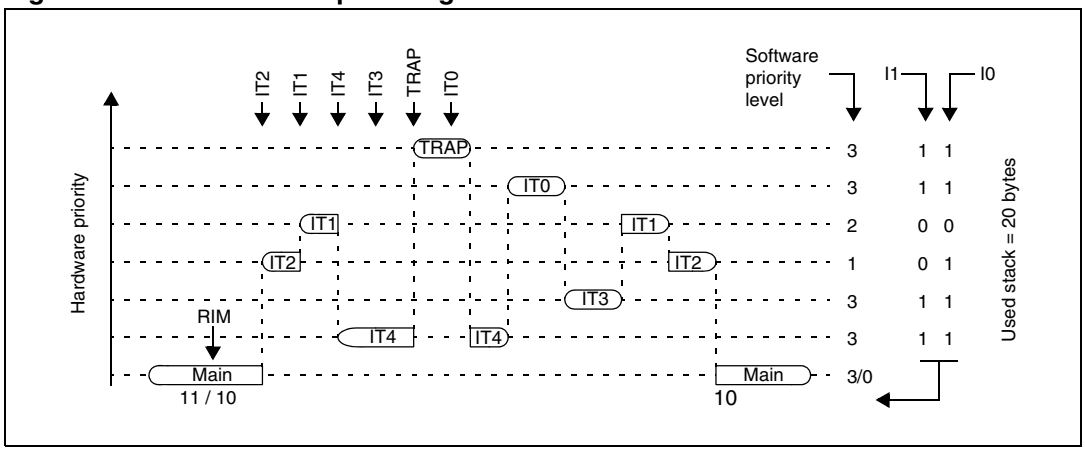
### Monitoring the $V_{DD}$ main supply

The AVD voltage threshold value is relative to the selected LVD threshold configured by option byte (see *Section 14.1 on page 179*).

If the AVD interrupt is enabled, an interrupt is generated when the voltage crosses the $V_{IT+(AVD)}$ or $V_{IT-(AVD)}$ threshold (AVDF bit toggles).

In the case of a drop in voltage, the AVD interrupt acts as an early warning, allowing software to shut down safely before the LVD resets the microcontroller. See *Figure 15*.

**Figure 19. Nested interrupt management**




## 7.5 Interrupt registers

### 7.5.1 CPU CC register interrupt bits

CPU CC Reset value: 111x 1010(xAh)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 1 | 1 | **I1** | H | **I0** | N | Z | C |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 15. CPU CC register interrupt bits description**

| Bit | Name | Function |
|---|---|---|
| 5 | I1 | Software Interrupt Priority 1 |
| 3 | I0 | Software Interrupt Priority 0 |

**Table 16. Interrupt software priority levels**

| Interrupt software priority | Level | I1 | I0 |
|---|---|---|---|
| Level 0 (main) | Low | 1 | 0 |
| Level 1 | ↓ | 0 | 1 |
| Level 2 | ↓ | 0 | 0 |
| Level 3 (= interrupt disable)(1) | High | 1 | 1 |

1. TRAP and RESET events can interrupt a level 3 program.

These two bits indicate the current interrupt software priority (see *Table 16*) and are set/cleared by hardware when entering in interrupt. The loaded value is given by the corresponding bits in the interrupt software priority registers (ISPRx).

### Halt mode recommendations

- Make sure that an external event is available to wake up the microcontroller from Halt mode.
- When using an external interrupt to wake up the microcontroller, reinitialize the corresponding I/O as “Input Pull-up with Interrupt” before executing the HALT instruction. The main reason for this is that the I/O may be wrongly configured due to external interference or by an unforeseen logical condition.
- For the same reason, reinitialize the sensitivity level of each external interrupt as a precautionary measure.
- The opcode for the HALT instruction is 0x8E. To avoid an unexpected HALT instruction due to a program counter failure, it is advised to clear all occurrences of the data value 0x8E from memory. For example, avoid defining a constant in ROM with the value 0x8E.
- As the HALT instruction clears the interrupt mask in the CC register to allow interrupts, the user may choose to clear all pending interrupt bits before executing the HALT instruction. This avoids entering other peripheral interrupt routines after executing the external interrupt routine corresponding to the wake-up event (reset or external interrupt).

### External interrupt function

When an I/O is configured as 'Input with Interrupt', an event on this I/O can generate an external interrupt request to the CPU.

Each pin can independently generate an interrupt request. The interrupt sensitivity is independently programmable using the sensitivity bits in the EICR register.

Each external interrupt vector is linked to a dedicated group of I/O port pins (see pinout description and interrupt section). If several input pins are selected simultaneously as interrupt sources, these are first detected according to the sensitivity bits in the EICR register and then logically ORed.

The external interrupts are hardware interrupts, which means that the request latch (not accessible directly by the application) is automatically cleared when the corresponding interrupt vector is fetched. To clear an unwanted pending interrupt by software, the sensitivity bits in the EICR register must be modified.

## 9.2.2 Output modes

The output configuration is selected by setting the corresponding DDR register bit. In this case, writing the DR register applies this digital value to the I/O pin through the latch. Then reading the DR register returns the previously stored value.

Two different output modes can be selected by software through the OR register: Output push-pull and open-drain.

**Table 27. DR register value and output pin status**

| DR | Push-pull | Open-drain |
|---|---|---|
| 0 | $V_{SS}$ | $V_{SS}$ |
| 1 | $V_{DD}$ | Floating |

## 9.2.3 Alternate functions

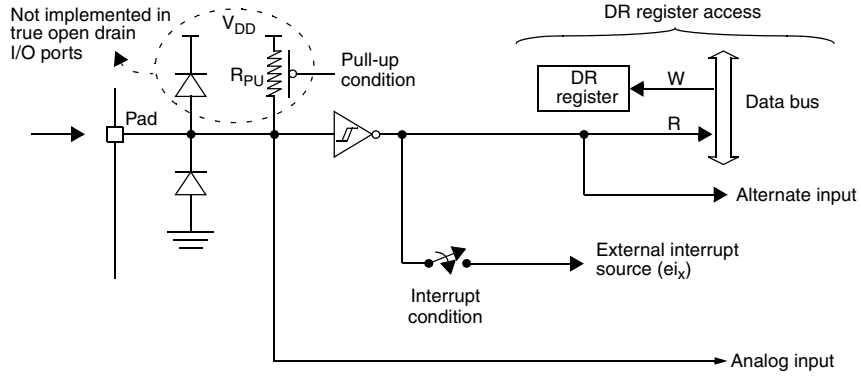
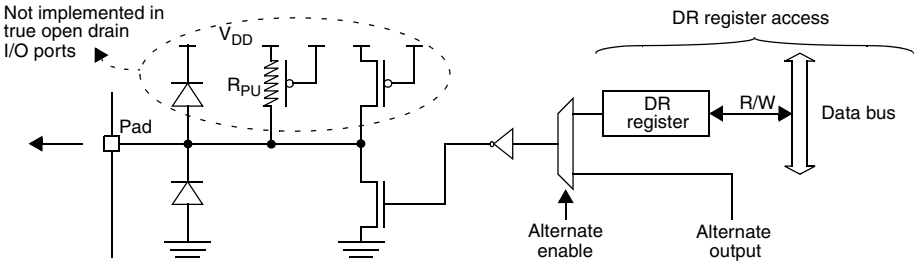
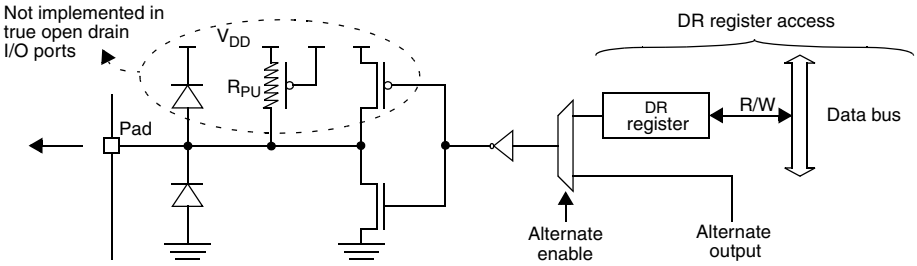
When an on-chip peripheral is configured to use a pin, the alternate function is automatically selected. This alternate function takes priority over the standard I/O programming.

When the signal is coming from an on-chip peripheral, the I/O pin is automatically configured in output mode (push-pull or open drain according to the peripheral).

When the signal is going to an on-chip peripheral, the I/O pin must be configured in input mode. In this case, the pin state is also digitally readable by addressing the DR register.

*Note: Input pull-up configuration can cause unexpected value at the input of the alternate peripheral input. When an on-chip peripheral use a pin as input and output, this pin has to be configured in input floating mode.*

**Table 29. I/O port configurations**

| | Hardware configuration |
|---|---|
| Input(1) |  |
| Open-drain output(2) |  |
| PUSH-pull output(2) |  |

1. When the I/O port is in input configuration and the associated alternate function is enabled as an output, reading the DR register will read the alternate function output status.
2. When the I/O port is in output configuration and the associated alternate function is enabled as an input, the alternate function reads the pin status given by the DR register content.

**Caution:** The alternate function must not be activated as long as the pin is configured as input with interrupt, in order to avoid generating spurious interrupts.

**Table 49. CR1 register description (continued)**

| Bit | Name | Function |
|---|---|---|
| 4 | FOLV2 | Forced Output compare 2<br>This bit is set and cleared by software.<br>0: No effect on the OCMP2 pin.<br>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. |
| 3 | FOLV1 | Forced Output compare 1<br>This bit is set and cleared by software.<br>0: No effect on the OCMP1 pin.<br>1: Forces OLVL1 to be copied to the OCMP1 pin, if the OC1E bit is set and even if there is no successful comparison. |
| 2 | OLVL2 | Output Level 2<br>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. |
| 1 | IEDG1 | Input Edge 1<br>This bit determines which type of level transition on the ICAP1 pin will trigger the capture.<br>0: A falling edge triggers the capture.<br>1: A rising edge triggers the capture. |
| 0 | OLVL1 | Output Level 1<br>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. |

### Control Register 2 (CR2)

CR2 Reset value: 0000 0000 (00h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| OC1E | OC2E | OPM | PWM | CC[1:0] | | IEDG2 | EXEDG |
| R/W | R/W | R/W | R/W | R/W | | R/W | R/W |

### Input Capture 1 High Register (IC1HR)

This is an 8-bit register that contains the high part of the counter value (transferred by the input capture 1 event).

IC1HR Reset value: undefined

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| MSB | | | | | | | LSB |
| RO | RO | RO | RO | RO | RO | RO | RO |

### 10.4.5 Error flags

#### Master mode fault (MODF)

Master mode fault occurs when the master device has its $\overline{SS}$ pin pulled low.

When a Master mode fault occurs:

- The MODF bit is set and an SPI interrupt request is generated if the SPIE bit is set.
- The SPE bit is reset. This blocks all output from the device and disables the SPI peripheral.
- The MSTR bit is reset, thus forcing the device into slave mode.

Clearing the MODF bit is done through a software sequence:

1. A read access to the SPICSR register while the MODF bit is set.
2. A write to the SPICR register.

*Note: To avoid any conflicts in an application with multiple slaves, the $\overline{SS}$ pin must be pulled high during the MODF bit clearing sequence. The SPE and MSTR bits may be restored to their original state during or after this clearing sequence.*

Hardware does not allow the user to set the SPE and MSTR bits while the MODF bit is set except in the MODF bit clearing sequence.

#### Overrun condition (OVR)

An overrun condition occurs, when the master device has sent a data byte and the slave device has not cleared the SPIF bit issued from the previously transmitted byte.

When an Overrun occurs the OVR bit is set and an interrupt request is generated if the SPIE bit is set.

In this case, the receiver buffer contains the byte sent after the SPIF bit was last cleared. A read to the SPIDR register returns this byte. All other bytes are lost.

The OVR bit is cleared by reading the SPICSR register.

#### Write collision error (WCOL)

A write collision occurs when the software tries to write to the SPIDR register while a data transfer is taking place with an external device. When this happens, the transfer continues uninterrupted and the software write is unsuccessful.

Write collisions can occur both in master and slave mode. See also *Slave Select management on page 99*.

*Note: A read collision will never occur since the received data byte is placed in a buffer in which access is always synchronous with the MCU operation.*

The WCOL bit in the SPICSR register is set if a write collision occurs.

No SPI interrupt is generated when the WCOL bit is set (the WCOL bit is a status flag only).

A software sequence clears the WCOL bit (see *Figure 53*).

### 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.

A 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.

A 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.

**Caution:** In Mute mode, do not write to the SCICR2 register. If the SCI is in Mute mode during the read operation (RWU = 1) and an address mark wake-up event occurs (RWU is reset) before the write operation, the RWU bit will be set again by this write operation. Consequently the address byte is lost and the SCI is not woken up from Mute mode.

### Parity control

Parity control (generation of parity bit in transmission and parity checking in reception) can be enabled by setting the PCE bit in the SCICR1 register. Depending on the frame length defined by the M bit, the possible SCI frame formats are as listed in *Table 59*.

**Table 59. Frame formats(1)(2)**

| M bit | PCE bit | SCI frame |
|---|---|---|
| 0 | 0 | \| SB \| 8 bit data \| STB \| |
| 0 | 1 | \| SB \| 7-bit data \| PB \| STB \| |
| 1 | 0 | \| SB \| 9-bit data \| STB \| |
| 1 | 1 | \| SB \| 8-bit data PB \| STB \| |

1. SB = Start bit, STB = Stop bit, and PB = Parity bit.
2. In case of wake-up by an address mark, the MSB bit of the data is taken into account and not the Parity bit.

### 10.5.5 Low power modes

**Table 60. Effect of low power modes on SCI**

| Mode | Description |
|---|---|
| Wait | No effect on SCI.<br>SCI interrupts cause the device to exit from Wait mode. |
| Halt | SCI registers are frozen.<br>In Halt mode, the SCI stops transmitting/receiving until Halt mode is exited. |

### 10.5.6 Interrupts

The SCI interrupt events are connected to the same interrupt vector.

These events generate an interrupt if the corresponding Enable Control bit is set and the interrupt mask in the CC register is reset (RIM instruction).

**Table 61. SCI interrupt control/wake-up capability**

| Interrupt event | Event flag | Enable control bit | Exit from WAIT | Exit from HALT |
|---|---|---|---|---|
| Transmit data register empty | TDRE | TIE | Yes | No |
| Transmission complete | TC | TCIE | Yes | No |
| Received data ready to be read | RDRF | RIE | Yes | No |
| Overrun error detected | OR | | Yes | No |
| Idle line detected | IDLE | ILIE | Yes | No |
| Parity error | PE | PIE | Yes | No |

### 10.5.7 SCI registers

#### SCI Status Register (SCISR)

SCISR Reset value: 1100 0000 (C0h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TDRE | TC | RDRF | IDLE | OR | NF | FE | PE |
| RO | RO | RO | RO | RO | RO | RO | RO |

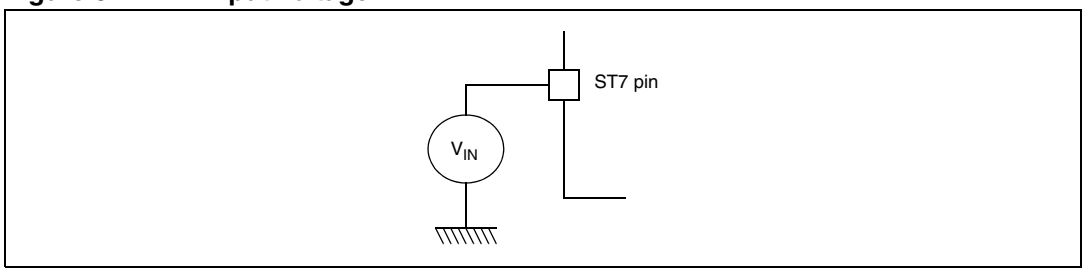
**Table 64. SCICR2 register description (continued)**

| Bit | Name | Function |
|---|---|---|
| 5 | RIE | Receiver interrupt Enable<br>This bit is set and cleared by software.<br>0: Interrupt is inhibited<br>1: An SCI interrupt is generated whenever OR = 1 or RDRF = 1 in the SCISR register. |
| 4 | ILIE | Idle Line Interrupt Enable<br>This bit is set and cleared by software.<br>0: Interrupt is inhibited<br>1: An SCI interrupt is generated whenever IDLE = 1 in the SCISR register. |
| 3 | TE | Transmitter Enable<br>This bit enables the transmitter. It is set and cleared by software.<br>0: Transmitter is disabled<br>1: Transmitter is enabled<br>*Notes:*<br>*- During transmission, a '0' pulse on the TE bit ('0' followed by '1') sends a preamble (Idle line) after the current word.*<br>*- When TE is set there is a 1 bit-time delay before the transmission starts.*<br>**Caution:** The TDO pin is free for general purpose I/O only when the TE and RE bits are both cleared (or if TE is never set). |
| 2 | RE | Receiver Enable<br>This bit enables the receiver. It is set and cleared by software.<br>0: Receiver is disabled<br>1: Receiver is enabled and begins searching for a start bit<br>*Note: Before selecting Mute mode (setting the RWU bit), the SCI must first receive some data, otherwise it cannot function in Mute mode with Wake-Up by Idle line detection.* |
| 1 | RWU | Receiver Wake-Up<br>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.<br>0: Receiver in Active mode<br>1: Receiver in Mute mode |
| 0 | SBK | Send Break<br>This bit set is used to send break characters. It is set and cleared by software.<br>0: No break character is transmitted.<br>1: Break characters are transmitted.<br>*Note: If the SBK bit is set to '1' and then to '0', the transmitter will send a Break word at the end of the current word.* |

### 12.1.5 Pin input voltage

The input voltage measurement on a pin of the device is described in *Figure 61*.

**Figure 61. Pin input voltage**




## 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

**Table 83. Voltage characteristics**

| Symbol | Ratings | Maximum value | Unit |
|---|---|---|---|
| $V_{DD}$ - $V_{SS}$ | Supply voltage | 6.5 | V |
| $V_{PP}$ - $V_{SS}$ | Programming voltage | 13 | |
| $V_{IN}$ (1)(2) | Input voltage on true open drain pin | $V_{SS}$ - 0.3 to 6.5 | |
| | Input voltage on any other pin | $V_{SS}$ - 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 | |
| $V_{ESD(HBM)}$ | Electrostatic discharge voltage (human body model) | see *Section 12.8.3 on page 160* | |
| $V_{ESD(MM)}$ | Electrostatic discharge voltage (machine model) | | |

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). For the same reason, unused I/O pins must not be directly tied to $V_{DD}$ or $V_{SS}$.
2. $I_{INJ(PIN)}$ must never be exceeded. This is implicitly ensured 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.

2. Not tested in production.

### 12.8.3 Absolute maximum ratings (electrical sensitivity)

Based on three different tests (ESD, LU and DLU) using specific measurement methods, the product is stressed in order to determine its performance in terms of electrical sensitivity. For more details, refer to the application note AN1181.

#### Electrostatic discharge (ESD)

Electrostatic discharges (a positive then a negative pulse separated by 1 second) are applied to the pins of each sample according to each pin combination. The sample size depends on the number of supply pins in the device (3 parts*(n+1) supply pin). Two models can be simulated for standard microcontrollers: Human Body Model and Charged Device Model. These tests conform to standards JESD22-A114 and JESD22-C101. There is an additional model for automotive microcontrollers: Machine Model, JESD22-A115.

**Table 103. Absolute maximum ratings**

| Symbol | Ratings | Conditions | Maximum value(1) | Unit |
|---|---|---|---|---|
| $V_{ESD(HBM)}$ | Electrostatic discharge voltage (human body model) | $T_A$ = +25°C | 2000 | V |
| $V_{ESD(MM)}$ | Electrostatic discharge voltage (machine model) | | 200 | |
| $V_{ESD(CDM)}$ | Electrostatic discharge voltage (charged device model) | | 750 | |

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

#### Static and dynamic Latch-Up

- **LU**: 3 complementary static tests are required on 10 parts to assess the latch-up performance. A supply overvoltage (applied to each power supply pin) and a current injection (applied to each input, output and configurable I/O pin) are performed on each sample. This test conforms to the EIA/JESD 78 IC latch-up standard. For more details, refer to the application note AN1181.
- **DLU**: Electrostatic discharges (one positive then one negative test) are applied to each pin of three samples when the micro is running to assess the latch-up performance in dynamic mode. Power supplies are set to the typical values, the oscillator is connected as near as possible to the pins of the micro and the component is put in reset mode. This test conforms to the IEC1000-4-2 and SAEJ1752/3 standards. For more details, refer to the application note AN1181.

**Figure 70. Typical $V_{OL}$ at $V_{DD}$ = 5V (standard ports)**

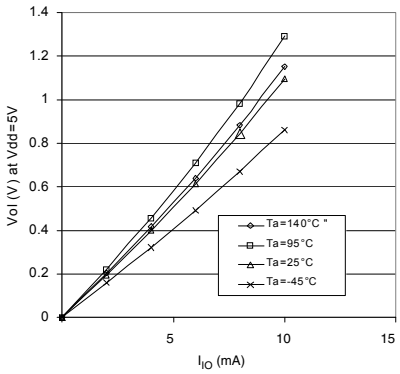



**Figure 71. Typical $V_{OL}$ at $V_{DD}$ = 5V (high-sink ports)**

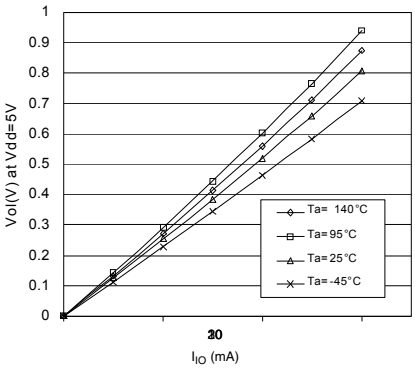
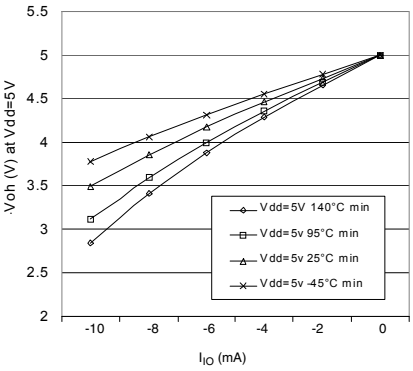



**Figure 72. Typical $V_{OH}$ at $V_{DD}$ = 5V**



# 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* |