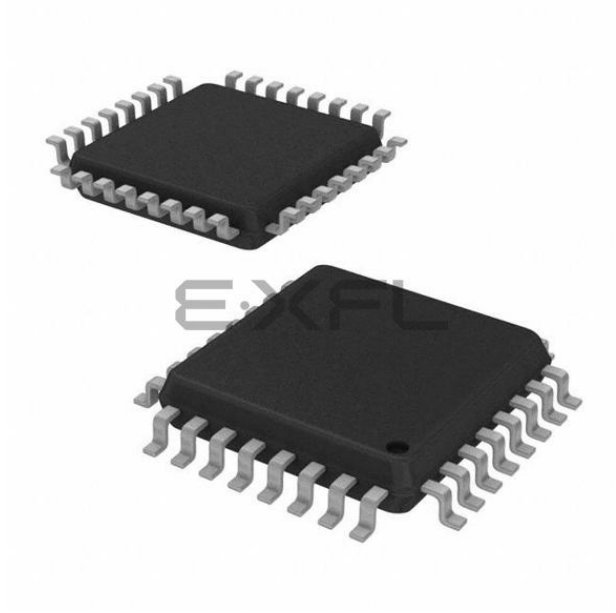


Welcome to E-XFL.COM

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	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/st72f324bk2t6

Contents

List of tables

Refer to *Section 9: I/O ports on page 58* for more details on the software configuration of the I/O ports.

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

Table 2. Device pin description

Pin			Type	Level		Port						Main function (after reset)	Alternate function	
No.		Name		Input	Output	Input				Output				
LQFP44	LQFP32					float	wpu	int	ana	OD	PP			
6	30	PB4 (HS)	I/O	C_T	HS	**X**	ei3			X	X	Port B4		
7	31	PD0/AIN0	I/O	C_T		**X**	X		X	X	X	Port D0	ADC analog input 0	
8	32	PD1/AIN1	I/O	C_T		**X**	X		X	X	X	Port D1	ADC analog input 1	
9	-	PD2/AIN2	I/O	C_T		**X**	X		X	X	X	Port D2	ADC analog input 2	
10	-	PD3/AIN3	I/O	C_T		**X**	X		X	X	X	Port D3	ADC analog input 3	
11	-	PD4/AIN4	I/O	C_T		**X**	X		X	X	X	Port D4	ADC analog input 4	
12	-	PD5/AIN5	I/O	C_T		**X**	X		X	X	X	Port D5	ADC analog input 5	
13	1	V_{AREF}(1)	S									Analog reference voltage for ADC		
14	2	V_{SSA}(1)	S									Analog ground voltage		
15	3	PF0/MCO/AIN8	I/O	C_T		**X**	ei1		X	X	X	Port F0	Main clock out (f_{CPU})	ADC analog input 8
16	4	PF1 (HS)/BEEP	I/O	C_T	HS	**X**	ei1			X	X	Port F1	Beep signal output	
17	-	PF2 (HS)	I/O	C_T	HS	**X**	ei1			X	X	Port F2		
18	5	PF4/OCMP1_A /AIN10	I/O	C_T		**X**	X		X	X	X	Port F4	Timer A output compare 1	ADC analog Input 10
19	6	PF6 (HS)/ICAP1_A	I/O	C_T	HS	**X**	X			X	X	Port F6	Timer A input capture 1	
20	7	PF7 (HS)/EXTCLK_A	I/O	C_T	HS	**X**	X			X	X	Port F7	Timer A external clock source	
21	-	V_{DD_0}(1)	S									Digital main supply voltage		
22	-	V_{SS_0}(1)	S									Digital ground voltage		
23	8	PC0/OCMP2_B /AIN12	I/O	C_T		**X**	X		X	X	X	Port C0	Timer B output compare 2	ADC analog input 12
24	9	PC1/OCMP1_B /AIN13	I/O	C_T		**X**	X		X	X	X	Port C1	Timer B output compare 1	ADC analog input 13
25	10	PC2 (HS)/ICAP2_B	I/O	C_T	HS	**X**	X			X	X	Port C2	Timer B input capture 2	

They can be also set/cleared by software with the RIM, SIM, HALT, WFI, IRET and PUSH/POP instructions (see *Table 18: Dedicated interrupt instruction set*).

7.5.2 Interrupt software priority registers (ISPRx)

ISPRx Reset value: 1111 1111 (FFh)

	7	6	5	4	3	2	1	0
ISPR0	I1_3	I0_3	I1_2	I0_2	I1_1	I0_1	I1_0	I0_0
ISPR1	I1_7	I0_7	I1_6	I0_6	I1_5	I0_5	I1_4	I0_4
ISPR2	I1_11	I0_11	I1_10	I0_10	I1_9	I0_9	I1_8	I0_8
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ISPR3	1	1	1	1	I1_13	I0_13	I1_12	I0_12
	RO	RO	RO	RO	R/W	R/W	R/W	R/W

These four registers contain the interrupt software priority of each interrupt vector.

- Each interrupt vector (except reset and TRAP) has corresponding bits in these registers where its own software priority is stored. This correspondence is shown in the following *Table 17*.

Table 17. ISPRx interrupt vector correspondence

Vector address	ISPRx bits
FFFBh-FFFAh	I1_0 and I0_0 bits
FFF9h-FFF8h	I1_1 and I0_1 bits
...	...
FFE1h-FFE0h	I1_13 and I0_13 bits

- Each I1_x and I0_x bit value in the ISPRx registers has the same meaning as the I1 and I0 bits in the CC register.
- Level 0 cannot be written (I1_x = 1, I0_x = 0). In this case, the previously stored value is kept (for example, previous value = CFh, write = 64h, result = 44h).

The reset, and TRAP vectors have no software priorities. When one is serviced, the I1 and I0 bits of the CC register are both set.

Caution: If the I1_x and I0_x bits are modified while the interrupt x is executed the following behavior has to be considered: If the interrupt x is still pending (new interrupt or flag not cleared) and the new software priority is higher than the previous one, the interrupt x is re-entered. Otherwise, the software priority stays unchanged up to the next interrupt request (after the IRET of the interrupt x).

Table 18. Dedicated interrupt instruction set(1)

Instruction	New description	Function/example	I1	H	I0	N	Z	C
HALT	Entering HALT mode		1		0			

Table 25. Interrupt mapping

No.	Source block	Description	Register label	Priority order	Exit from Halt/Active Halt	Address vector
	Reset	Reset	N/A		yes	FFFEh-FFFFh
	TRAP	Software interrupt			no	FFFCh-FFFDh
0	Not used					FFFAh-FFFBh
1	MCC/RTC	Main clock controller time base interrupt	MCCSR	Higher priority	yes	FFF8h-FFF9h
2	ei0	External interrupt port A3..0	N/A		yes	FFF6h-FFF7h
3	ei1	External interrupt port F2..0			yes	FFF4h-FFF5h
4	ei2	External interrupt port B3..0			yes	FFF2h-FFF3h
5	ei3	External interrupt port B7..4			yes	FFF0h-FFF1h
6	Not used					FFEEh-FFEFh
7	SPI	SPI peripheral interrupts	SPICSR		yes	FFECh-FFEDh
8	Timer A	Timer A peripheral interrupts	TASR		no	FFEAh-FFEBh
9	Timer B	Timer B peripheral interrupts	TBSR		no	FFE8h-FFE9h
10	SCI	SCI peripheral interrupts	SCISR	Lower priority	no	FFE6h-FFE7h
11	AVD	Auxiliary voltage detector interrupt	SICSR		no	FFE4h-FFE5h

8.4 Active Halt and Halt modes

Active Halt and Halt modes are the two lowest power consumption modes of the MCU. They are both entered by executing the 'HALT' instruction. The decision to enter either in Active Halt or Halt mode is given by the MCC/RTC interrupt enable flag (OIE bit in the MCCSR register).

Table 26. MCC/RTC low power mode selection

MCCSR OIE bit	Power saving mode entered when HALT instruction is executed
0	Halt mode
1	Active Halt mode

8.4.1 Active Halt mode

Active Halt mode is the lowest power consumption mode of the MCU with a real-time clock available. It is entered by executing the 'HALT' instruction when the OIE bit of the Main Clock Controller Status register (MCCSR) is set (see *Section 10.2: Main clock controller with real-time clock and beeper (MCC/RTC) on page 69* for more details on the MCCSR register).

The MCU can exit Active Halt mode on reception of either an MCC/RTC interrupt, a specific interrupt (see *Table 25: Interrupt mapping*) or a reset. When exiting Active Halt mode by means of an interrupt, no 256 or 4096 CPU cycle delay occurs. The CPU resumes operation by servicing the interrupt or by fetching the reset vector which woke it up (see *Figure 25*).

When entering Active Halt mode, the I[1:0] bits in the CC register are forced to '10b' to enable interrupts. Therefore, if an interrupt is pending, the MCU wakes up immediately.

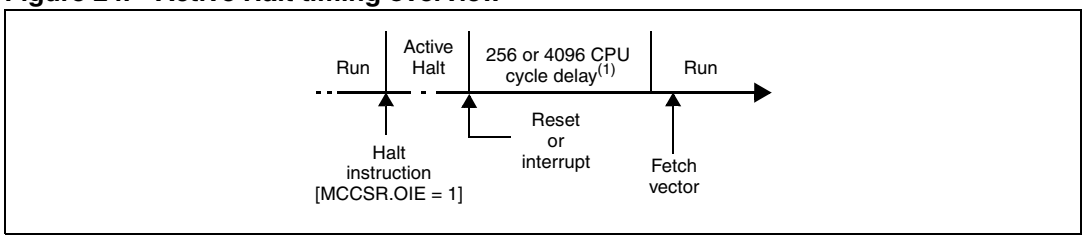
In Active Halt mode, only the main oscillator and its associated counter (MCC/RTC) are running to keep a wake-up time base. All other peripherals are not clocked except those which get their clock supply from another clock generator (such as external or auxiliary oscillator).

The safeguard against staying locked in Active Halt mode is provided by the oscillator interrupt.

Note: As soon as the interrupt capability of one of the oscillators is selected (MCCSR.OIE bit set), entering Active Halt mode while the Watchdog is active does not generate a reset. This means that the device cannot spend more than a defined delay in this power saving mode.

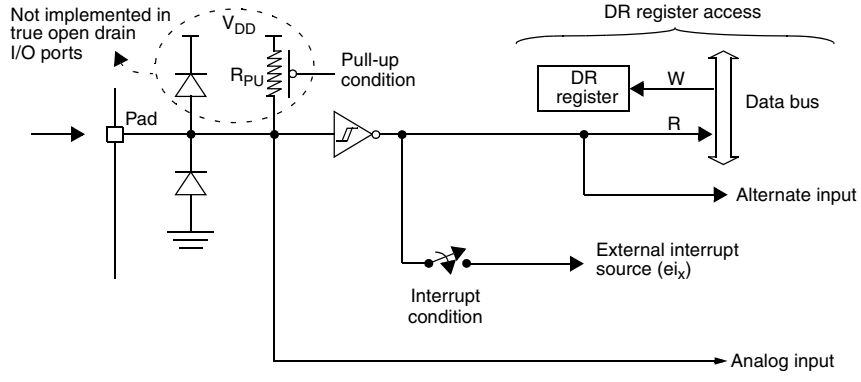
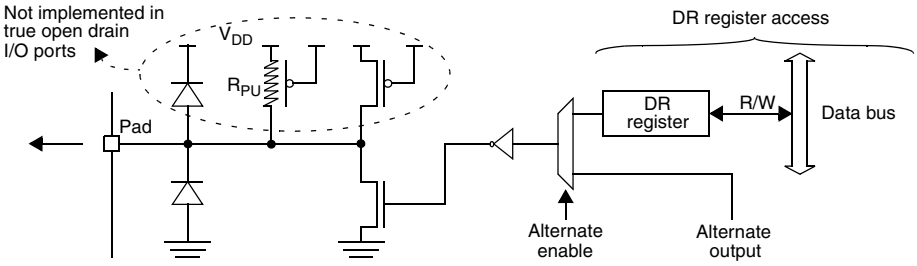
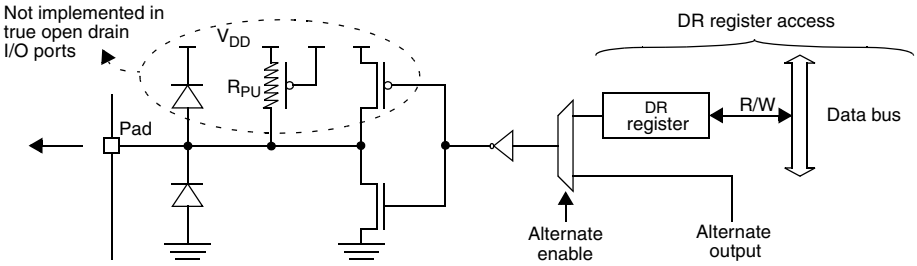
Caution: When exiting Active Halt mode following an interrupt, OIE bit of MCCSR register must not be cleared before t_{DELAY} after the interrupt occurs (t_{DELAY} = 256 or 4096 t_{CPU} delay depending on option byte). Otherwise, the ST7 enters Halt mode for the remaining t_{DELAY} period.

Figure 24. Active Halt timing overview



1. This delay occurs only if the MCU exits Active Halt mode by means of a reset.

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.

Analog alternate function

When the pin is used as an ADC input, the I/O must be configured as floating input. The analog multiplexer (controlled by the ADC registers) switches the analog voltage present on the selected pin to the common analog rail which is connected to the ADC input.

It is recommended not to change the voltage level or loading on any port pin while conversion is in progress. Furthermore it is recommended not to have clocking pins located close to a selected analog pin.

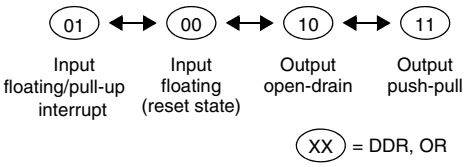
Warning: The analog input voltage level must be within the limits stated in the absolute maximum ratings.

9.3 I/O port implementation

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

Switching these I/O ports from one state to another should be done in a sequence that prevents unwanted side effects. Recommended safe transitions are illustrated in *Figure 29.* Other transitions are potentially risky and should be avoided, since they are likely to present unwanted side-effects such as spurious interrupt generation.

Figure 29. Interrupt I/O port state transitions



9.4 Low power modes

Table 30. Effect of low power modes on I/O ports

Mode	Description
Wait	No effect on I/O ports. External interrupts cause the device to exit from Wait mode.
Halt	No effect on I/O ports. External interrupts cause the device to exit from Halt mode.

9.5 Interrupts

The external interrupt event generates an interrupt if the corresponding configuration is selected with DDR and OR registers and the interrupt mask in the CC register is not active (RIM instruction).

10.3.3 Functional description

Counter

The main block of the programmable timer is a 16-bit free running upcounter and its associated 16-bit registers. The 16-bit registers are made up of two 8-bit registers called high and low.

- Counter Register (CR)
  - Counter High Register (CHR) is the most significant byte (MSB)
  - Counter Low Register (CLR) is the least significant byte (LSB)
- Alternate Counter Register (ACR)
  - Alternate Counter High Register (ACHR) is the most significant byte (MSB)
  - Alternate Counter Low Register (ACLR) is the least significant byte (LSB)

These two read-only 16-bit registers contain the same value but with the difference that reading the ACLR register does not clear the TOF bit (timer overflow flag), located in the Status register (SR) (see note at the end of paragraph entitled *16-bit read sequence)*.

Writing in the CLR register or ACLR register resets the free running counter to the FFFCh value. Both counters have a reset value of FFFCh (this is the only value which is reloaded in the 16-bit timer). The reset value of both counters is also FFFCh in one pulse mode and PWM mode.

The timer clock depends on the clock control bits of the CR2 register, as illustrated in *Table 50*. The value in the counter register repeats every 131072, 262144 or 524288 CPU clock cycles depending on the CC[1:0] bits. The timer frequency can be $f_{CPU}/2$, $f_{CPU}/4$, $f_{CPU}/8$ or an external frequency.

Table 52. 16-bit timer register map and reset values (continued)

Address (Hex.)	Register label	7	6	5	4	3	2	1	0
Timer A: 35 Timer B: 45	IC1LR Reset value	MSB x	x	x	x	x	x	x	LSB x
Timer A: 36 Timer B: 46	OC1HR Reset value	MSB 1	0	0	0	0	0	0	LSB 0
Timer A: 37 Timer B: 47	OC1LR Reset value	MSB 0	0	0	0	0	0	0	LSB 0
Timer A: 3E Timer B: 4E	OC2HR Reset value	MSB 1	0	0	0	0	0	0	LSB 0
Timer A: 3F Timer B: 4F	OC2LR Reset value	MSB 0	0	0	0	0	0	0	LSB 0
Timer A: 38 Timer B: 48	CHR Reset value	MSB 1	1	1	1	1	1	1	LSB 1
Timer A: 39 Timer B: 49	CLR Reset value	MSB 1	1	1	1	1	1	0	LSB 0
Timer A: 3A Timer B: 4A	ACHR Reset value	MSB 1	1	1	1	1	1	1	LSB 1
Timer A: 3B Timer B: 4B	ACLR Reset value	MSB 1	1	1	1	1	1	0	LSB 0
Timer A: 3C Timer B: 4C	IC2HR Reset value	MSB x	x	x	x	x	x	x	LSB x
Timer A: 3D Timer B: 4D	IC2LR Reset value	MSB x	x	x	x	x	x	x	LSB x

10.4 Serial peripheral interface (SPI)

10.4.1 Introduction

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

10.4.2 Main features

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

Collision error will occur when the slave writes to the shift register (see *Write collision error (WCOL) on page 103*).

Figure 50. Generic $\overline{SS}$ timing diagram

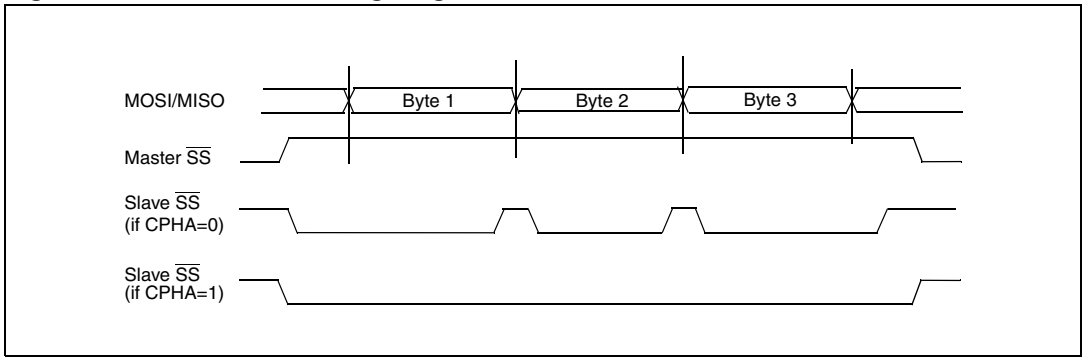
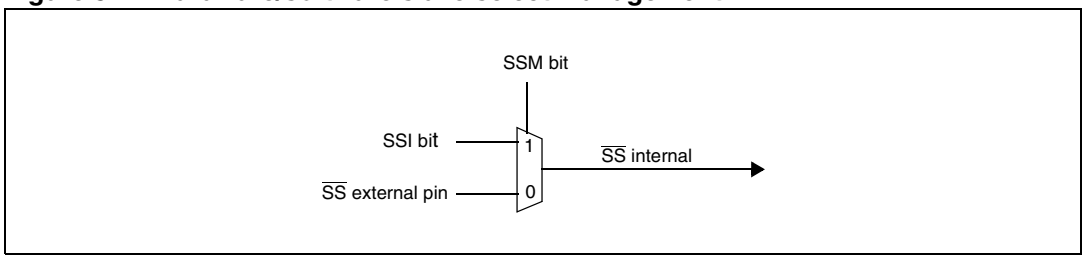


Figure 51. Hardware/software slave select management



Master mode operation

In master mode, the serial clock is output on the SCK pin. The clock frequency, polarity and phase are configured by software (refer to the description of the SPICSR register).

Note: The idle state of SCK must correspond to the polarity selected in the SPICSR register (by pulling up SCK if CPOL = 1 or pulling down SCK if CPOL = 0).

How to operate the SPI in master mode

To operate the SPI in master mode, perform the following steps in order:

1. Write to the SPICR register:
   - Select the clock frequency by configuring the SPR[2:0] bits.
   - Select the clock polarity and clock phase by configuring the CPOL and CPHA bits. *Figure 52* shows the four possible configurations.
     Note: The slave must have the same CPOL and CPHA settings as the master.
2. Write to the SPICSR register:
   - Either set the SSM bit and set the SSI bit or clear the SSM bit and tie the $\overline{SS}$ pin high for the complete byte transmit sequence.
3. Write to the SPICR register:
   - Set the MSTR and SPE bits.
     Note: MSTR and SPE bits remain set only if $\overline{SS}$ is high.

Caution: If the SPICSR register is not written first, the SPICR register setting (MSTR bit) might not be taken into account.

The transmit sequence begins when software writes a byte in the SPIDR register.

10.4.8 SPI registers

SPI Control Register (SPICR)

SPICR Reset value: 0000 xxxx (0xh)

7	6	5	4	3	2	1	0
SPIE	SPE	SPR2	MSTR	CPOL	CPHA	SPR[1:0]	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Table 55. SPICR register description

Bit	Name	Function
7	SPIE	Serial Peripheral Interrupt Enable This bit is set and cleared by software. 0: Interrupt is inhibited. 1: An SPI interrupt is generated whenever SPIF = 1, MODF = 1 or OVR = 1 in the SPICSR register.
6	SPE	Serial Peripheral Output Enable This bit is set and cleared by software. It is also cleared by hardware when, in master mode, $\overline{SS}$ = 0 (see *Master mode fault (MODF) on page 103*). The SPE bit is cleared by reset, so the SPI peripheral is not initially connected to the external pins. 0: I/O pins free for general purpose I/O 1: SPI I/O pin alternate functions enabled
5	SPR2	Divider Enable This bit is set and cleared by software and is cleared by reset. It is used with the SPR[1:0] bits to set the baud rate. Refer to *Table 56: SPI master mode SCK frequency*. 0: Divider by 2 enabled 1: Divider by 2 disabled *Note: This bit has no effect in slave mode.*
4	MSTR	Master mode This bit is set and cleared by software. It is also cleared by hardware when, in master mode, $\overline{SS}$ = 0 (see *Master mode fault (MODF) on page 103*). 0: Slave mode 1: Master mode. The function of the SCK pin changes from an input to an output and the functions of the MISO and MOSI pins are reversed.
3	CPOL	Clock Polarity This bit is set and cleared by software. This bit determines the idle state of the serial Clock. The CPOL bit affects both the master and slave modes. 0: SCK pin has a low level idle state 1: SCK pin has a high level idle state *Note: If CPOL is changed at the communication byte boundaries, the SPI must be disabled by resetting the SPE bit.*

10.5.4 Functional description

The block diagram of the serial control interface is shown in *Figure 55*. It contains six dedicated registers:

- 2 control registers (SCICR1 and SCICR2)
- a status register (SCISR)
- a baud rate register (SCIBRR)
- an extended prescaler receiver register (SCIERPR)
- an extended prescaler transmitter register (SCIETPR)

Refer to the register descriptions in *Section 10.5.7* for the definitions of each bit.

Serial data format

Word length may be selected as being either 8 or 9 bits by programming the M bit in the SCICR1 register (see *Figure 55*).

The TDO pin is in low state during the start bit.

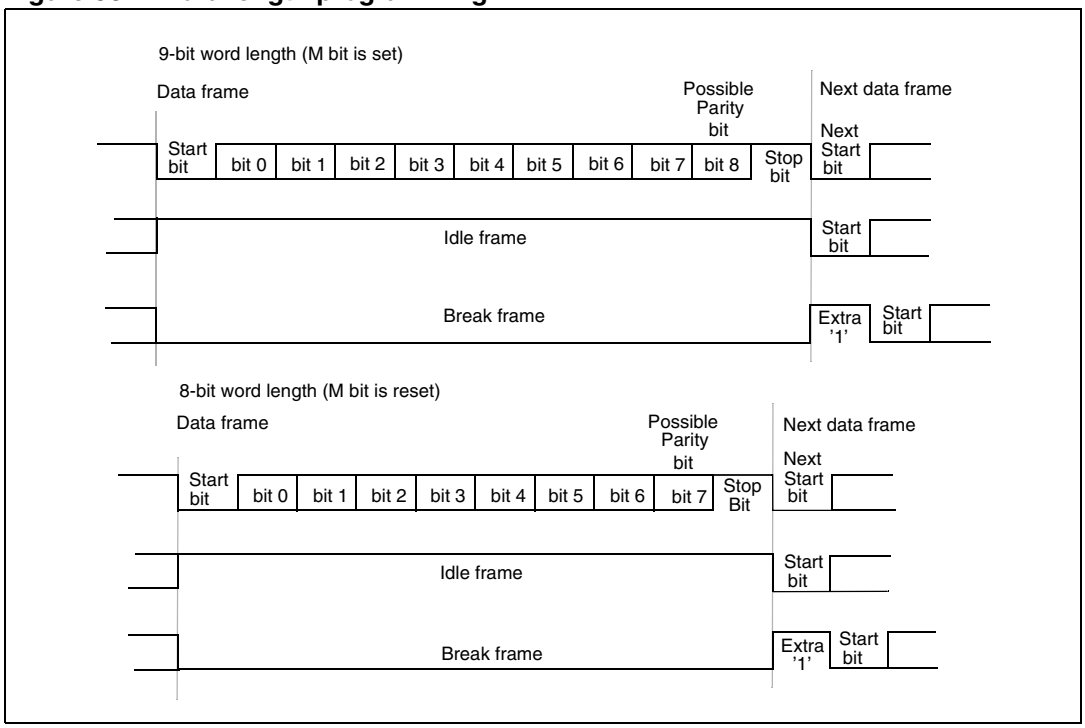
The TDO pin is in high state during the stop bit.

An Idle character is interpreted as an entire frame of ‘1’s followed by the start bit of the next frame which contains data.

A Break character is interpreted on receiving ‘0’s for some multiple of the frame period. At the end of the last break frame the transmitter inserts an extra ‘1’ bit to acknowledge the start bit.

Transmission and reception are driven by their own baud rate generator.

Figure 56. Word length programming



11.1.5 Indirect (short, long)

The required data byte to do the operation is found by its memory address, located in memory (pointer).

The pointer address follows the opcode. The indirect addressing mode consists of two submodes:

Indirect (short)

The pointer address is a byte, the pointer size is a byte, thus allowing 00 - FF addressing space, and requires 1 byte after the opcode.

Indirect (long)

The pointer address is a byte, the pointer size is a word, thus allowing 64 Kbyte addressing space, and requires 1 byte after the opcode.

11.1.6 Indirect indexed (short, long)

This is a combination of indirect and short indexed addressing modes. The operand is referenced by its memory address, which is defined by the unsigned addition of an index register value (X or Y) with a pointer value located in memory. The pointer address follows the opcode.

The indirect indexed addressing mode consists of two submodes:

Indirect indexed (short)

The pointer address is a byte, the pointer size is a byte, thus allowing 00 - 1FE addressing space, and requires 1 byte after the opcode.

Indirect indexed (long)

The pointer address is a byte, the pointer size is a word, thus allowing 64 Kbyte addressing space, and requires 1 byte after the opcode.

Table 79. Instructions supporting direct, indexed, indirect and indirect indexed addressing modes

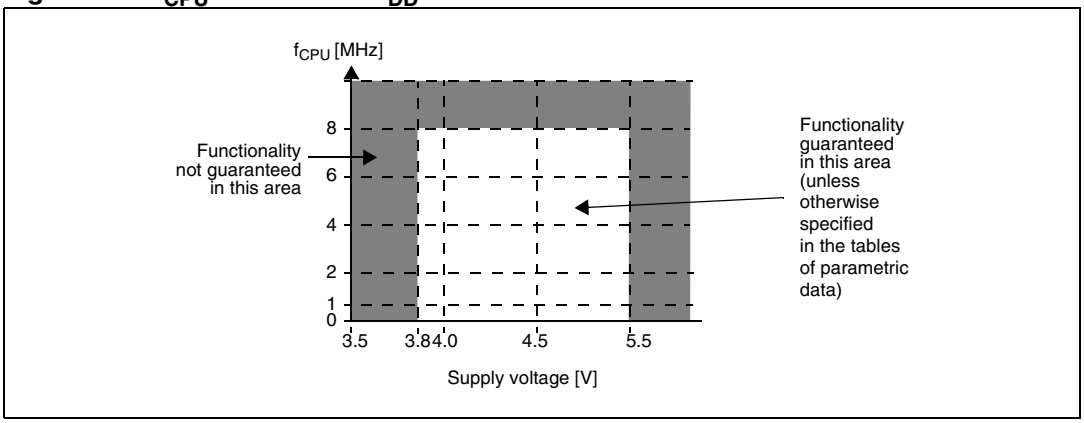
Instructions		Function
Long and short	LD	Load
	CP	Compare
	AND, OR, XOR	Logical operations
	ADC, ADD, SUB, SBC	Arithmetic Additions/Subtractions operations
	BCP	Bit Compare

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.

Figure 73. Typical V_{OL} vs. V_{DD} (standard ports)

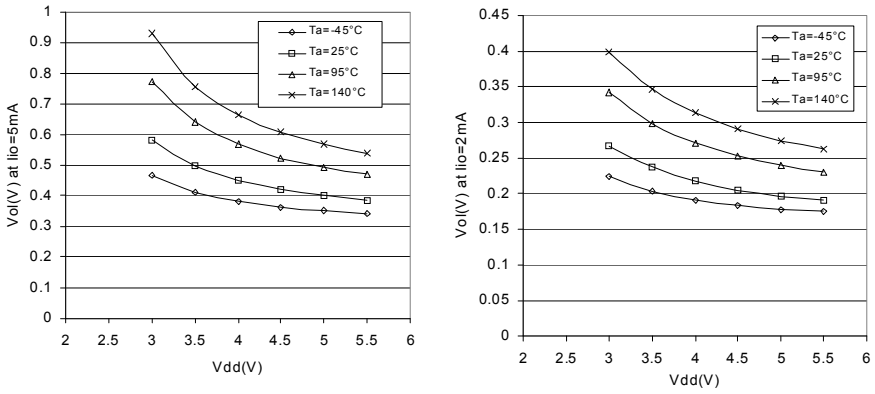


Figure 74. Typical V_{OL} vs. V_{DD} (high-sink ports)

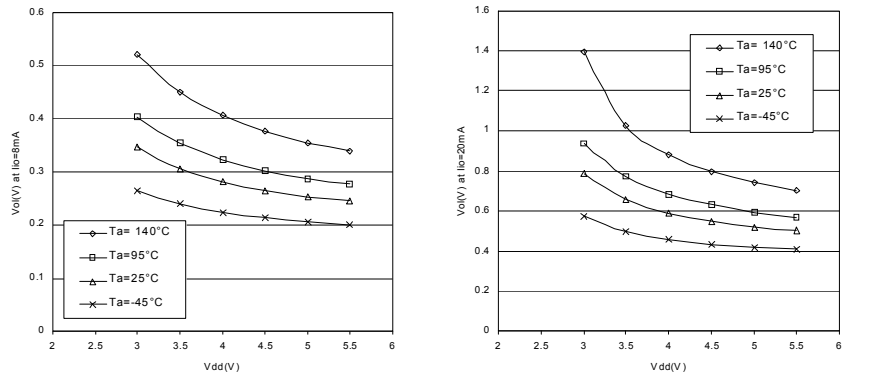


Figure 75. Typical V_{OH} vs. V_{DD}

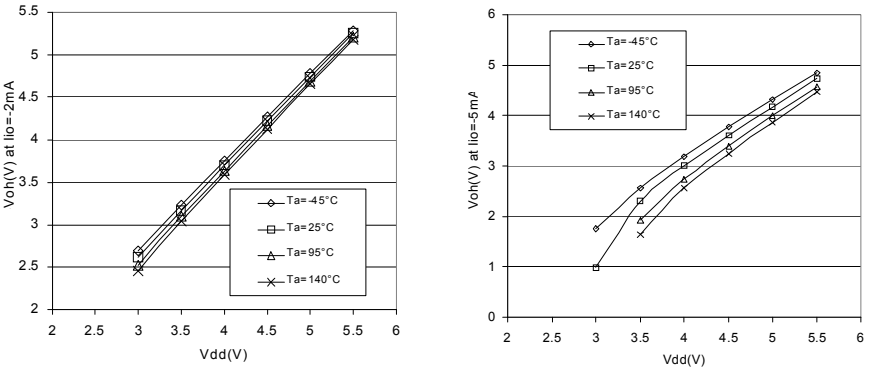
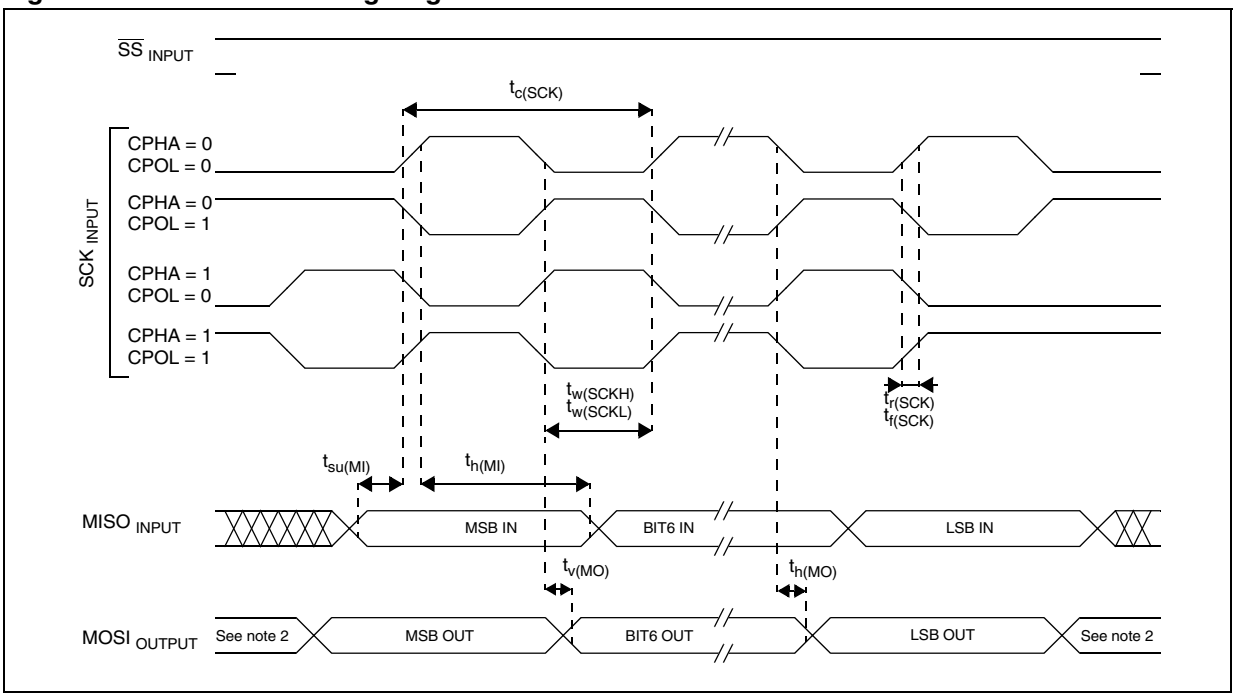


Figure 81. SPI master timing diagram(1)



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

ST72324B-Auto Microcontroller FASTROM/ROM Option List

(Last update: July 2007)

Customer: . . . . . .
Address: . . . . . .
. . . . . .
Contact: . . . . . .
Phone No: . . . . . .
Reference: . . . . . .

The FASTROM/ROM code name is assigned by STMicroelectronics.
FASTROM/ROM code must be sent in .S19 format. .Hex extension cannot be processed.

Device Type/Memory Size/Package (check only one option):

FASTROM DEVICE:	32K	16K	8K
LQFP44 10x10:	[ ] ST72P324B(J6)T	[ ] ST72P324B(J4)T	[ ] ST72P324B(J2)T
LQFP32 7x17:	[ ] ST72P324B(K6)T	[ ] ST72P324B(K4)T	[ ] ST72P324B(K2)T

ROM DEVICE:	32K	16K	8K
LQFP44 10x10:	[ ] ST72324B(J6)T	[ ] ST72324B(J4)T	[ ] ST72324B(J2)T
LQFP32 7x7:	[ ] ST72324B(K6)T	[ ] ST72324B(K4)T	[ ] ST72324B(K2)T

Conditioning for LQFP package (check only one option):

[ ] Tape and Reel [ ] Tray

Temperature range :
[ ] A (-40°C to +85°C)
[ ] B (-40°C to +105°C)
[ ] C (-40°C to +125°C)
[ ] D (-40°C to +150°C)

Special Marking: [ ] No [ ] Yes "_ _ _ _ _ _ _ _ _ _"
LQFP32: 7 characters max. Other packages: 10 characters max.
Authorized characters are letters, digits, '.', '-', '/' and spaces only.

Clock Source Selection: [ ] Resonator:
[ ] LP: Low power resonator (1 to 2 MHz)
[ ] MP: Medium power resonator (2 to 4 MHz)
[ ] MS: Medium speed resonator (4 to 8 MHz)
[ ] HS: High speed resonator (8 to 16 MHz)
[ ] Internal RC
[ ] External clock (sets MP medium power resonator in option byte)

PLL (1)(2) [ ] Disabled [ ] Enabled

LVD reset [ ] Disabled [ ] High threshold
[ ] Medium threshold [ ] Low threshold

Reset delay [ ] 256 cycles [ ] 4096 cycles

Watchdog selection [ ] Software activation [ ] Hardware activation

Halt when Watchdog on [ ] Reset [ ] No reset

Readout protection [ ] Disabled [ ] Enabled

Date . . . . . Signature . . . . . .

1. PLL must be disabled if internal RC network is selected.
2. The PLL can be enabled only if the resonator is configured to “Medium power: 2~4 MHz”.

CAUTION: The readout protection binary value is inverted between ROM and Flash products. The option byte checksum will differ between ROM and Flash.

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. *1 analog peripheral (low current coupling) on page 1*: Replaced '12 robust input ports' with '12 input ports' *Table 1: Device summary on page 1*: Corrected order of listed packages Added *Section 1.2: Differences between ST72324B-Auto and ST72324B datasheets on page 16* *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) *Table 2: Device pin description on page 18*: - replaced V_{DDA} with V_{REF} in *Note 1* - modified *Note 2* *Section 5.3.4: Condition Code register (CC) on page 29*: Replaced IxSPR with ISPRx *Section 9.5.1: I/O port implementation on page 65*: Removed following tables: - Standard ports PA5:4, PC7:0, PD5:0, PE1:0, PF7:6, 4 - Interrupt ports PB4, PB2:0, PF2:0 (with pull-up) - Interrupt ports PA3, PB3 (without pull-up) - *True open drain ports PA7:6* (configurations in these four tables already exist in *Table 32: Port configuration*) *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* *Table 96: OSCRANGE selection for typical resonators on page 157*: Deleted footnote detailing SMD and LEAD which was linked to 'Reference' column header *Table 102: EMI emissions on page 161*: - added LQFP32 package to all listed devices - changed values for 32 Kbyte ROM devices *Table 105: General characteristics on page 163*: - modified *Note 5* - modified *Note 6* *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 *Table 111: 10-bit ADC characteristics on page 172*: Modified input current leakage parameter and added *Note 2* *Table 112: ADC accuracy on page 175*: - added conditions to total unadjusted error, to offset error and to gain error - modified *Note 2*