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	32
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 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	44-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f324bj4t6

List of figures

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	

Table 3. Hardware register map (continued)

Address	Block	Register label	Register name	Reset status(1)	Remarks(1)
000Fh 0010h 0011h	Port F(1)	PFDR PFDDR PFOR	Port F data register Port F data direction register Port F option register	00h(2) 00h 00h	R/W R/W R/W
0012h to 0020h	Reserved area (15 bytes)				
0021h 0022h 0023h	SPI	SPIDR SPICR SPICSR	SPI data I/O register SPI control register SPI control/status register	xxh 0xh 00h	R/W R/W R/W
0024h 0025h 0026h 0027h	ITC	ISPR0 ISPR1 ISPR2 ISPR3	Interrupt software priority register 0 Interrupt software priority register 1 Interrupt software priority register 2 Interrupt software priority register 3	FFh FFh FFh FFh	R/W R/W R/W R/W
0028h		EICR	External interrupt control register	00h	R/W
0029h	Flash	FCSR	Flash control/status register	00h	R/W
002Ah	Watchdog	WDGCR	Watchdog control register	7Fh	R/W
002Bh	SI	SICSR	System integrity control/status register	000x 000xb	R/W
002Ch 002Dh	MCC	MCCSR MCCBCR	Main clock control/status register Main clock controller: beep control register	00h 00h	R/W R/W
002Eh to 0030h	Reserved area (3 bytes)				
0031h 0032h 0033h 0034h 0035h 0036h 0037h 0038h 0039h 003Ah 003Bh 003Ch 003Dh 003Eh 003Fh	Timer A	TACR2 TACR1 TACSR TAIC1HR TAIC1LR TAOC1HR TAOC1LR TACHR TACLR TAACHR TAACLR TAIC2HR TAIC2LR TAOC2HR TAOC2LR	Timer A control register 2 Timer A control register 1 Timer A control/status register Timer A input capture 1 high register Timer A input capture 1 low register Timer A output compare 1 high register Timer A output compare 1 low register Timer A counter high register Timer A counter low register Timer A alternate counter high register Timer A alternate counter low register Timer A input capture 2 high register Timer A input capture 2 low register Timer A output compare 2 high register Timer A output compare 2 low register	00h 00h xxxx x0xxb xxh xxh 80h 00h FFh FCh FFh FCh xxh xxh 80h 00h	R/W R/W R/W Read only Read only R/W R/W Read only Read only Read only Read only Read only Read only R/W R/W
0040h	Reserved area (1 byte)				

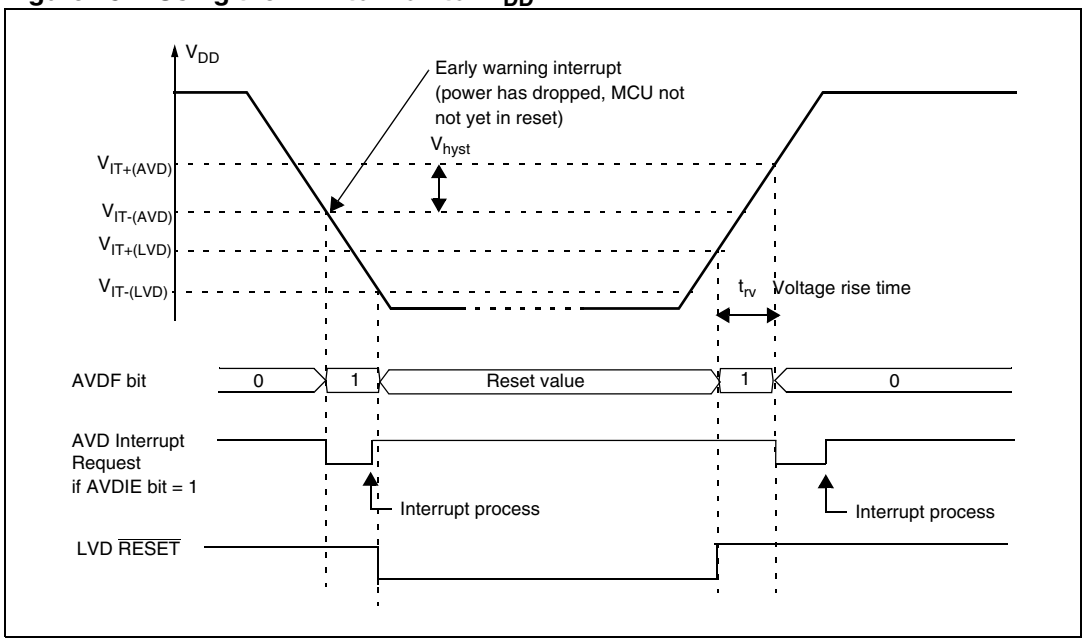
The interrupt on the rising edge is used to inform the application that the V_{DD} warning state is over.

If the voltage rise time t_{rv} is less than 256 or 4096 CPU cycles (depending on the reset delay selected by option byte), no AVD interrupt will be generated when $V_{IT+(AVD)}$ is reached.

If t_{rv} is greater than 256 or 4096 cycles then:

- If the AVD interrupt is enabled before the $V_{IT+(AVD)}$ threshold is reached, then 2 AVD interrupts will be received: the first when the AVDIE bit is set, and the second when the threshold is reached.
- If the AVD interrupt is enabled after the $V_{IT+(AVD)}$ threshold is reached then only one AVD interrupt will occur.

Figure 15. Using the AVD to monitor V_{DD}



6.5.3 Low power modes

Table 10. Effect of low power modes on SI

Mode	Description
Wait	No effect on SI. AVD interrupt causes the device to exit from Wait mode.
Halt	The CRSR register is frozen.

6.5.4 Interrupts

The AVD interrupt event generates an interrupt if the AVDIE bit is set and the interrupt mask in the CC register is reset (RIM instruction).

Table 11. AVD interrupt control/wake-up capability

Interrupt event	Event flag	Enable Control bit	Exit from WAIT	Exit from HALT
AVD event	AVDF	AVDIE	Yes	No

7 Interrupts

7.1 Introduction

The ST7 enhanced interrupt management provides the following features:

- Hardware interrupts
- Software interrupt (TRAP)
- Nested or concurrent interrupt management with flexible interrupt priority and level management:
  - up to 4 software programmable nesting levels
  - up to 16 interrupt vectors fixed by hardware
  - 2 non-maskable events: reset, TRAP

This interrupt management is based on:

- Bit 5 and bit 3 of the CPU CC register (I1:0)
- Interrupt software priority registers (ISPRx)
- Fixed interrupt vector addresses located at the high addresses of the memory map (FFE0h to FFFFh) sorted by hardware priority order

This enhanced interrupt controller guarantees full upward compatibility with the standard (not nested) ST7 interrupt controller.

7.2 Masking and processing flow

The interrupt masking is managed by the I1 and I0 bits of the CC register and the ISPRx registers which give the interrupt software priority level of each interrupt vector (see *Table 14*). The processing flow is shown in *Figure 16.*

When an interrupt request has to be serviced:

- Normal processing is suspended at the end of the current instruction execution.
- The PC, X, A and CC registers are saved onto the stack.
- I1 and I0 bits of CC register are set according to the corresponding values in the ISPRx registers of the serviced interrupt vector.
- The PC is then loaded with the interrupt vector of the interrupt to service and the first instruction of the interrupt service routine is fetched (refer to *Table 25: Interrupt mapping* for vector addresses).

The interrupt service routine should end with the IRET instruction which causes the contents of the saved registers to be recovered from the stack.

Note: As a consequence of the IRET instruction, the I1 and I0 bits will be restored from the stack and the program in the previous level will resume.

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

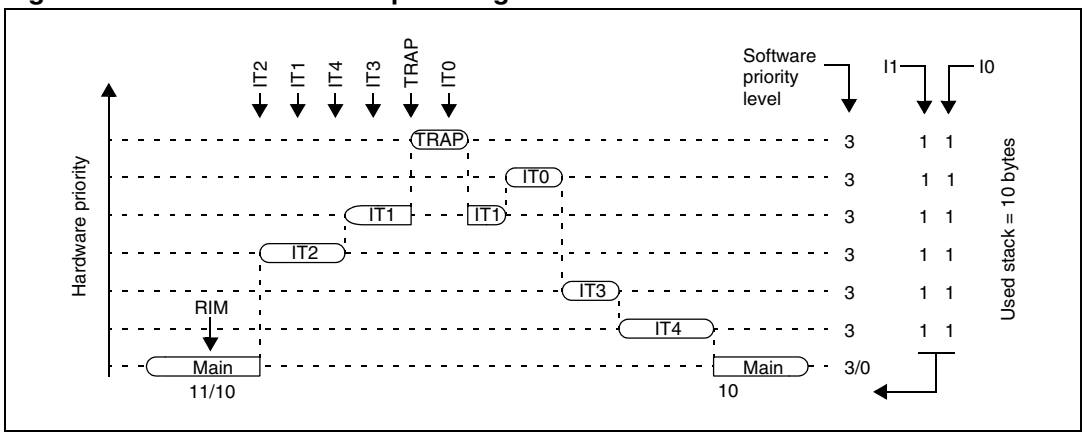
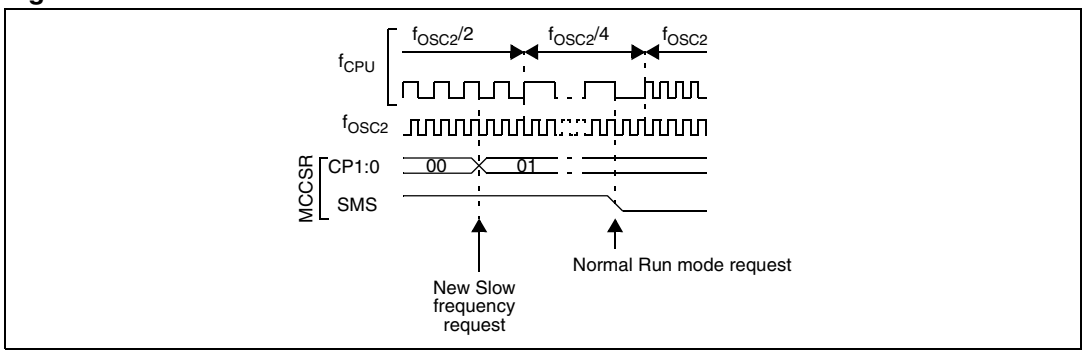


Figure 22. Slow mode clock transitions



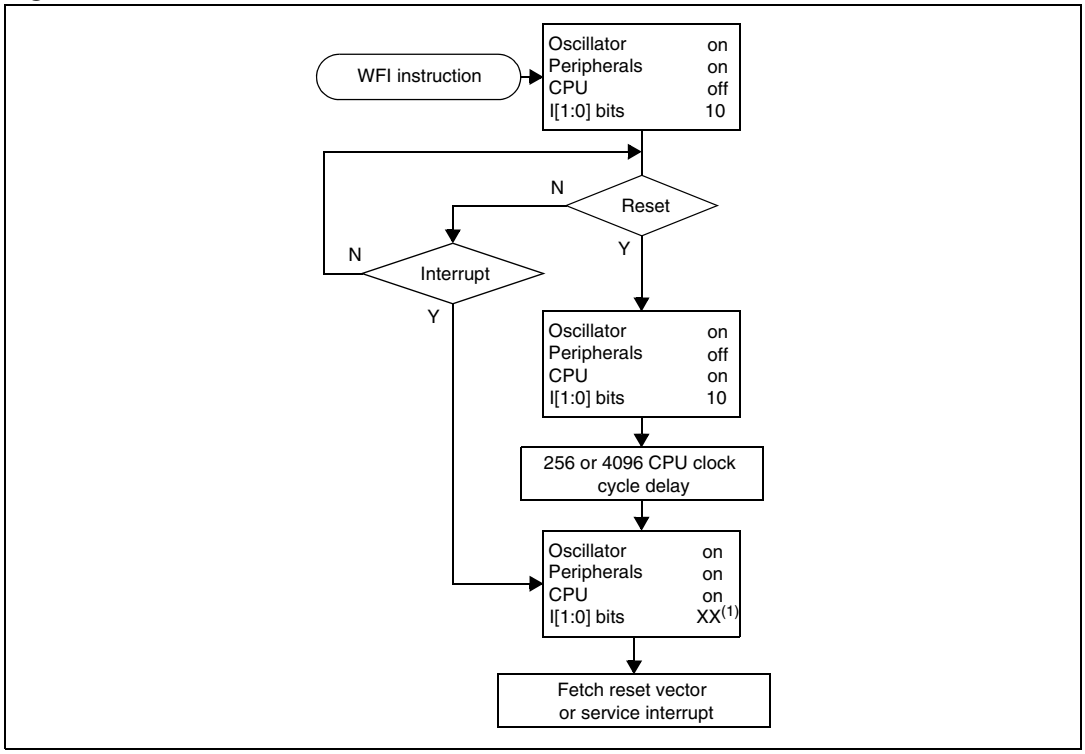
8.3 Wait mode

Wait mode places the MCU in a low power consumption mode by stopping the CPU.

This power saving mode is selected by calling the 'WFI' instruction.

All peripherals remain active. During Wait mode, the I[1:0] bits of the CC register are forced to '10', to enable all interrupts. All other registers and memory remain unchanged. The MCU remains in Wait mode until an interrupt or reset occurs, whereupon the Program Counter branches to the starting address of the interrupt or reset service routine. The MCU will remain in Wait mode until a reset or an interrupt occurs, causing it to wake up. Refer to *Figure 23*.

Figure 23. Wait mode flowchart



1. Before servicing an interrupt, the CC register is pushed on the stack. The I[1:0] bits of the CC register are set to the current software priority level of the interrupt routine and recovered when the CC register is popped.

MCC beep control register (MCCBCR)

MCCBCR Reset value: 0000 0000 (00h)

7	6	5	4	3	2	1	0
Reserved						BC[1:0]	
-						R/W	

Table 41. MCCBCR register description

Bit	Name	Function
7:2	-	Reserved, must be kept cleared
1:0	BC[1:0]	Beep Control These 2 bits select the PF1 pin beep capability (see *Table 42*). The beep output signal is available in Active Halt mode but has to be disabled to reduce the consumption.

Table 42. Beep frequency selection

BC1	BC0	Beep mode with f_{OSC2} = 8 MHz	
0	0	Off	
0	1	~2 kHz	Output Beep signal ~50% duty cycle
1	0	~1 kHz	
1	1	~500 Hz	

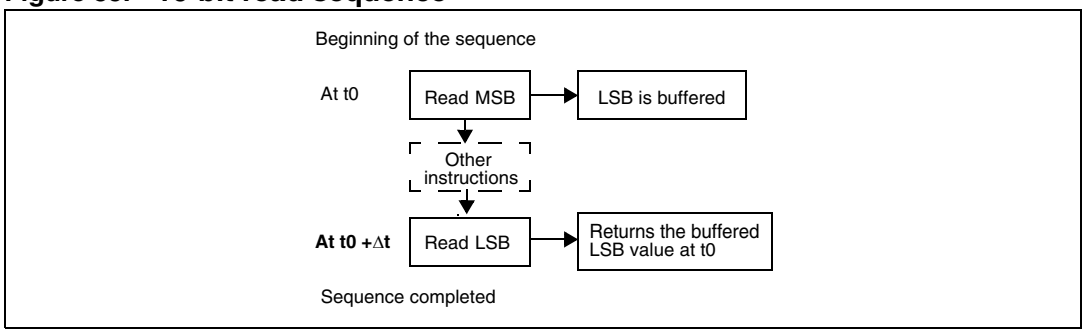
Table 43. Main clock controller register map and reset values

Address (Hex.)	Register label	7	6	5	4	3	2	1	0
002Bh	SICSR Reset value		AVDIE 0	AVDF 0	LVDRF x				WDGRF x
		0				0	0	0	
002Ch	MCCSR Reset value	MCO 0	CP1 0	CP0 0	SMS 0	TB1 0	TB0 0	OIE 0	OIF 0
002Dh	MCCBCR Reset value							BC1 0	BC0 0
		0	0	0	0	0	0		

16-bit read sequence

The 16-bit read sequence (from either the Counter register or the Alternate Counter register) is illustrated in the following*Figure 35*.

Figure 35. 16-bit read sequence



The user must first read the MSB, afterwhich the LSB value is automatically buffered.

This buffered value remains unchanged until the 16-bit read sequence is completed, even if the user reads the MSB several times.

After a complete reading sequence, if only the CLR register or ACLR register are read, they return the LSB of the count value at the time of the read.

Whatever the timer mode used (input capture, output compare, one pulse mode or PWM mode) an overflow occurs when the counter rolls over from FFFFh to 0000h then:

- The TOF bit of the SR register is set.
- A timer interrupt is generated if:
  - TOIE bit of the CR1 register is set and
  - I bit of the CC register is cleared.

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

Clearing the overflow interrupt request is done in two steps:

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

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

The timer is not affected by Wait mode.

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

Table 49. CR1 register description (continued)

Bit	Name	Function
4	FOLV2	Forced Output compare 2 This bit is set and cleared by software. 0: No effect on the OCMP2 pin. 1: Forces the OLVL2 bit to be copied to the OCMP2 pin, if the OC2E bit is set and even if there is no successful comparison.
3	FOLV1	Forced Output compare 1 This bit is set and cleared by software. 0: No effect on the OCMP1 pin. 1: Forces OLVL1 to be copied to the OCMP1 pin, if the OC1E bit is set and even if there is no successful comparison.
2	OLVL2	Output Level 2 This bit is copied to the OCMP2 pin whenever a successful comparison occurs with the OC2R register and OCxE is set in the CR2 register. This value is copied to the OCMP1 pin in One Pulse mode and Pulse Width modulation mode.
1	IEDG1	Input Edge 1 This bit determines which type of level transition on the ICAP1 pin will trigger the capture. 0: A falling edge triggers the capture. 1: A rising edge triggers the capture.
0	OLVL1	Output Level 1 The OLVL1 bit is copied to the OCMP1 pin whenever a successful comparison occurs with the OC1R register and the OC1E bit is set in the CR2 register.

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

Table 50. CR2 register description

Bit	Name	Function
7	OCIE	Output Compare 1 Pin Enable This bit is used only to output the signal from the timer on the OCMP1 pin (OLV1 in Output Compare mode, both OLV1 and OLV2 in PWM and One-Pulse mode). Whatever the value of the OC1E bit, the Output Compare 1 function of the timer remains active. 0: OCMP1 pin alternate function disabled (I/O pin free for general-purpose I/O). 1: OCMP1 pin alternate function enabled.
6	OC2E	Output Compare 2 Pin Enable This bit is used only to output the signal from the timer on the OCMP2 pin (OLV2 in Output Compare mode). Whatever the value of the OC2E bit, the Output Compare 2 function of the timer remains active. 0: OCMP2 pin alternate function disabled (I/O pin free for general-purpose I/O). 1: OCMP2 pin alternate function enabled.
5	OPM	One Pulse Mode 0: One Pulse mode is not active. 1: One Pulse mode is active, the ICAP1 pin can be used to trigger one pulse on the OCMP1 pin; the active transition is given by the IEDG1 bit. The length of the generated pulse depends on the contents of the OC1R register.
4	PWM	Pulse Width Modulation 0: PWM mode is not active. 1: PWM mode is active, the OCMP1 pin outputs a programmable cyclic signal; the length of the pulse depends on the value of OC1R register; the period depends on the value of OC2R register.
3:2	CC[1:0]	Clock Control The timer clock mode depends on these bits. 00: Timer clock = $f_{CPU}/4$ 01: Timer clock = $f_{CPU}/2$ 10: Timer clock = $f_{CPU}/8$ 11: Timer clock = external clock (where available) *Note: If the external clock pin is not available, programming the external clock configuration stops the counter.*
1	IEDG2	Input Edge 2 This bit determines which type of level transition on the ICAP2 pin will trigger the capture. 0: A falling edge triggers the capture. 1: A rising edge triggers the capture.
0	EXEDG	External Clock Edge This bit determines which type of level transition on the external clock pin EXTCLK will trigger the counter register. 0: A falling edge triggers the counter register. 1: A rising edge triggers the counter register.

SPI Data I/O Register (SPIDR)

SPIDR Reset value: undefined

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The SPIDR register is used to transmit and receive data on the serial bus. In a master device, a write to this register will initiate transmission/reception of another byte.

Note: During the last clock cycle the SPIF bit is set, a copy of the received data byte in the shift register is moved to a buffer. When the user reads the serial peripheral data I/O register, the buffer is actually being read.

While the SPIF bit is set, all writes to the SPIDR register are inhibited until the SPICSR register is read.

Warning: A write to the SPIDR register places data directly into the shift register for transmission.

A read to the SPIDR register returns the value located in the buffer and not the content of the shift register (see *Figure 48*).

Table 58. SPI register map and reset values

Address (Hex.)	Register label	7	6	5	4	3	2	1	0
0021h	SPIDR Reset value	MSB x	x	x	x	x	x	x	LSB x
0022h	SPICR Reset value	SPIE 0	SPE 0	SPR2 0	MSTR 0	CPOL x	CPHA x	SPR1 x	SPR0 x
0023h	SPICSR Reset value	SPIF 0	WCOL 0	OVR 0	MODF 0	0	SOD 0	SSM 0	SSI 0

10.5.3 General description

The interface is externally connected to another device by two pins (see *Figure 56*):

- TDO: Transmit Data Output. When the transmitter and the receiver are disabled, the output pin returns to its I/O port configuration. When the transmitter and/or the receiver are enabled and nothing is to be transmitted, the TDO pin is at high level.
- RDI: Receive Data Input is the serial data input. Oversampling techniques are used for data recovery by discriminating between valid incoming data and noise.

Through these pins, serial data is transmitted and received as frames comprising:

- an Idle Line prior to transmission or reception
- a start bit
- a data word (8 or 9 bits) least significant bit first
- a Stop bit indicating that the frame is complete

This interface uses two types of baud rate generator:

- a conventional type for commonly-used baud rates
- an extended type with a prescaler offering a very wide range of baud rates even with non-standard oscillator frequencies

11 Instruction set

11.1 CPU addressing modes

The CPU features 17 different addressing modes which can be classified in 7 main groups (see *Table 75*).

Table 75. Addressing mode groups

Addressing mode	Example
Inherent	nop
Immediate	ld A,#$55
Direct	ld A,$55
Indexed	ld A,($55,X)
Indirect	ld A,([$55],X)
Relative	jrne loop
Bit operation	bset byte,#5

The CPU Instruction Set is designed to minimize the number of bytes required per instruction: To do so, most of the addressing modes may be divided in two submodes called long and short:

- Long addressing mode is more powerful because it can use the full 64 Kbyte address space, however it uses more bytes and more CPU cycles.
- Short addressing mode is less powerful because it can generally only access page zero (0000h - 00FFh range), but the instruction size is more compact, and faster. All memory to memory instructions use short addressing modes only (CLR, CPL, NEG, BSET, BRES, BTJT, BTJF, INC, DEC, RLC, RRC, SLL, SRL, SRA, SWAP)

The ST7 Assembler optimizes the use of long and short addressing modes.

Table 76. CPU addressing mode overview

Mode			Syntax	Destination	Pointer address (Hex.)	Pointer size (Hex.)	Length (bytes)
Inherent			nop				+ 0
Immediate			ld A,#$55				+ 1
Short	Direct		ld A,$10	00..FF			+ 1
Long	Direct		ld A,$1000	0000..FFFF			+ 2
No offset	Direct	Indexed	ld A,(X)	00..FF			+ 0
Short	Direct	Indexed	ld A,($10,X)	00..1FE			+ 1
Long	Direct	Indexed	ld A,($1000,X)	0000..FFFF			+ 2
Short	Indirect		ld A,[$10]	00..FF	00..FF	byte	+ 2
Long	Indirect		ld A,[$10.w]	0000..FFFF	00..FF	word	+ 2
Short	Indirect	Indexed	ld A,([$10],X)	00..1FE	00..FF	byte	+ 2

11.1.2 Immediate

Immediate instructions have two bytes: The first byte contains the opcode and the second byte contains the operand value.

Table 78. Immediate instructions

Instruction	Function
LD	Load
CP	Compare
BCP	Bit Compare
AND, OR, XOR	Logical operations
ADC, ADD, SUB, SBC	Arithmetic operations

11.1.3 Direct

In Direct instructions, the operands are referenced by their memory address. The direct addressing mode consists of two submodes:

Direct (short)

The address is a byte, thus requiring only one byte after the opcode, but only allows 00 - FF addressing space.

Direct (long)

The address is a word, thus allowing 64 Kbyte addressing space, but requires 2 bytes after the opcode.

11.1.4 Indexed (no offset, short, long)

In this mode, the operand is referenced by its memory address, which is defined by the unsigned addition of an index register (X or Y) with an offset.

The indexed addressing mode consists of three submodes:

Indexed (no offset)

There is no offset, (no extra byte after the opcode), and it allows 00 - FF addressing space.

Indexed (short)

The offset is a byte, thus requiring only one byte after the opcode and allows 00 - 1FE addressing space.

Indexed (long)

The offset is a word, thus allowing 64 Kbyte addressing space and requires 2 bytes after the opcode.

12.5.2 Flash current consumption

Table 90. Flash current consumption

Symbol	Parameter	Conditions	32K Flash		16/8K Flash		Unit
			Typ	Max(1)	Typ	Max(1)	
I_{DD}	Supply current in Run mode(2)	f_{OSC} = 2 MHz, f_{CPU} = 1 MHz f_{OSC} = 4 MHz, f_{CPU} = 2 MHz f_{OSC} = 8 MHz, f_{CPU} = 4 MHz f_{OSC} = 16 MHz, f_{CPU} = 8 MHz	1.3 2.0 3.6 7.1	3.0 5.0 8.0 15.0	1 1.4 2.4 4.4	2.3 3.5 5.3 7.0	mA
	Supply current in Slow mode(2)	f_{OSC} = 2 MHz, f_{CPU} = 62.5 kHz f_{OSC} = 4 MHz, f_{CPU} = 125 kHz f_{OSC} = 8 MHz, f_{CPU} = 250 kHz f_{OSC} = 16 MHz, f_{CPU} = 500 kHz	0.6 0.7 0.8 1.1	2.7 3.0 3.6 4.0	0.48 0.53 0.63 0.80	1 1.1 1.2 1.4	
	Supply current in Wait mode(2)	f_{OSC} = 2 MHz, f_{CPU} = 1 MHz f_{OSC} = 4 MHz, f_{CPU} = 2 MHz f_{OSC} = 8 MHz, f_{CPU} = 4 MHz f_{OSC} = 16 MHz, f_{CPU} = 8 MHz	0.8 1.2 2.0 3.5	3.0 4.0 5.0 7.0	0.6 0.9 1.3 2.3	1.8 2.2 2.6 3.6	
	Supply current in Slow Wait mode(2)	f_{OSC} = 2 MHz, f_{CPU} = 62.5 kHz f_{OSC} = 4 MHz, f_{CPU} = 125 kHz f_{OSC} = 8 MHz, f_{CPU} = 250 kHz f_{OSC} = 16 MHz, f_{CPU} = 500 kHz	580 650 770 1050	1200 1300 1800 2000	430 470 530 660	950 1000 1050 1200	µA
	Supply current in Halt mode(3)	-40°C ≤ T_A ≤ +85°C	<1	10	<1	10	
		-40°C ≤ T_A ≤ +125°C	5	50	<1	50	
	Supply current in Active Halt mode(4)	f_{OSC} = 2 MHz f_{OSC} = 4 MHz f_{OSC} = 8 MHz f_{OSC} = 16 MHz	365 380 410 500	475 500 550 650	315 330 360 460	425 450 500 600	

1. Data based on characterization results, tested in production at V_{DD} max. and f_{CPU} max.
2. Measurements are done in the following conditions:
   - Program executed from RAM, CPU running with RAM access. The increase in consumption when executing from Flash is 50%.
   - All I/O pins in input mode with a static value at V_{DD} or V_{SS} (no load)
   - All peripherals in reset state
   - LVD disabled
   - Clock input (OSC1) driven by external square wave
   - In Slow and Slow Wait modes, f_{CPU} is based on f_{OSC} divided by 32
   - To obtain the total current consumption of the device, add the clock source (*Section 12.6.3*) and the peripheral power consumption (*Section 12.5.4*).
3. All I/O pins in push-pull 0 mode (when applicable) with a static value at V_{DD} or V_{SS} (no load), LVD disabled. Data based on characterization results, tested in production at V_{DD} max. and f_{CPU} max.
4. Data based on characterization results, not tested in production. All I/O pins in push-pull 0 mode (when applicable) with a static value at V_{DD} or V_{SS} (no load); clock input (OSC1) driven by external square wave, LVD disabled. To obtain the total current consumption of the device, add the clock source consumption (*Section 12.6.3*).

Figure 65. Typical application with a crystal or ceramic resonator (32 Kbyte Flash and ROM devices)

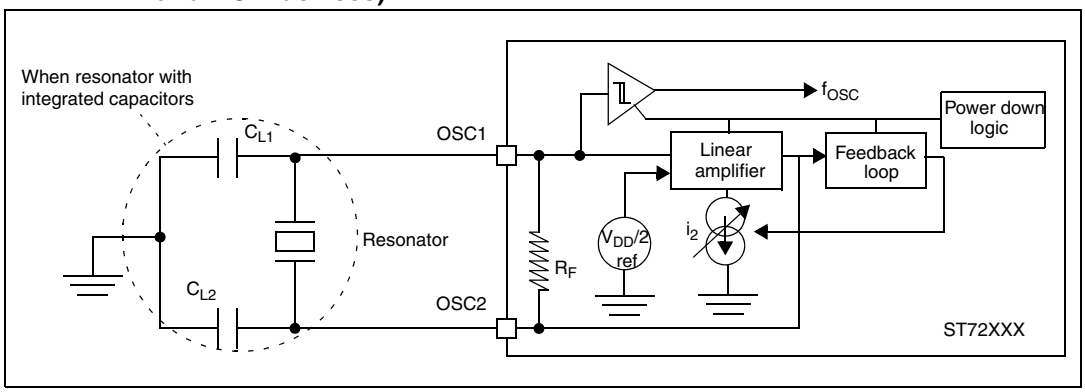


Table 96. OSCRANGE selection for typical resonators

Supplier	f_{OSC} (MHz)	Typical ceramic resonators(1)	
		Reference	Recommended OSCRANGE option bit configuration
Murata	2	CSTCC2M00G56A-R0	MP mode(2)
	4	CSTCR4M00G55B-R0	MS mode
	8	CSTCE8M00G52A-R0	HS mode
	16	CSTCE16M0V51A-R0	HS mode

1. Resonator characteristics given by the ceramic resonator manufacturer.
2. LP mode is not recommended for 2 MHz resonator because the peak to peak amplitude is too small (>0.8V). For more information on these resonators, please consult www.murata.com.

12.6.4 RC oscillators

Table 97. RC oscillators

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$f_{OSC\ (RCINT)}$	Internal RC oscillator frequency (see *Figure 66)*	T_A = 25°C, V_{DD} = 5V	2	3.5	5.6	MHz

Figure 66. Typical $f_{OSC(RCINT)}$ vs T_A

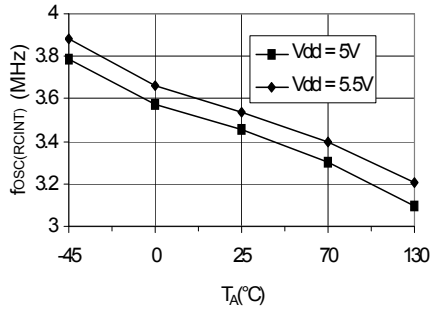


Table 101. EMS test results

Symbol	Parameter	Conditions	Level/class
V_{FESD}	Voltage limits to be applied on any I/O pin to induce a functional disturbance	32 Kbyte Flash or ROM device: V_{DD} = 5V, T_A = +25°C, f_{OSC} = 8 MHz conforms to IEC 1000-4-2	3B
		8 or 16 Kbyte ROM device: V_{DD} = 5V, T_A = +25°C, f_{OSC} = 8 MHz conforms to IEC 1000-4-2	4A
		8 or 16 Kbyte Flash device: V_{DD} = 5V, T_A = +25°C, f_{OSC} = 8 MHz conforms to IEC 1000-4-2	4B
V_{FFTB}	Fast transient voltage burst limits to be applied through 100pF on V_{DD} and V_{DD} pins to induce a functional disturbance	V_{DD} = 5V, T_A = +25°C, f_{OSC} = 8 MHz conforms to IEC 1000-4-4	4A

12.8.2 Electromagnetic interference (EMI)

Based on a simple application running on the product (toggling two LEDs through the I/O ports), the product is monitored in terms of emission. This emission test is in line with the norm SAE J 1752/3 which specifies the board and the loading of each pin.

Table 102. EMI emissions

Symbol	Parameter	Conditions	Device/package(1)	Monitored frequency band	Max vs [f_{OSC}/f_{CPU}]		Unit
					8/4 MHz	16/8 MHz	
S_{EMI}	Peak level(2)	V_{DD} = 5V T_A = +25°C conforming to SAE J 1752/3	8/16 Kbyte Flash LQFP32 and LQFP44	0.1 MHz to 30 MHz	12	18	dBµV
				30 MHz to 130 MHz	19	25	
				130 MHz to 1 GHz	15	22	
				SAE EMI Level	3	3.5	-
			32 Kbyte Flash LQFP32 and LQFP44	0.1 MHz to 30 MHz	13	14	dBµV
				30 MHz to 130 MHz	20	25	
				130 MHz to 1 GHz	16	21	
				SAE EMI Level	3.0	3.5	-
			8/16 Kbyte ROM LQFP32 and LQFP44	0.1 MHz to 30 MHz	12	15	dBµV
				30 MHz to 130 MHz	23	26	
				130 MHz to 1 GHz	15	20	
				SAE EMI Level	3.0	3.5	-
			32 Kbyte ROM LQFP32 and LQFP44	0.1 MHz to 30 MHz	17	21	dBµV
				30 MHz to 130 MHz	24	30	
				130 MHz to 1 GHz	18	23	
				SAE EMI Level	3.0	3.5	-

1. Refer to application note AN1709 for data on other package types.

13 Package characteristics

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK® packages, depending on their level of environmental compliance. ECOPACK® specifications, grade definitions and product status are available at: *www.st.com*. ECOPACK® is an ST trademark.

13.1 LQFP44 package characteristics

Figure 87. 44-pin low profile quad flat package outline

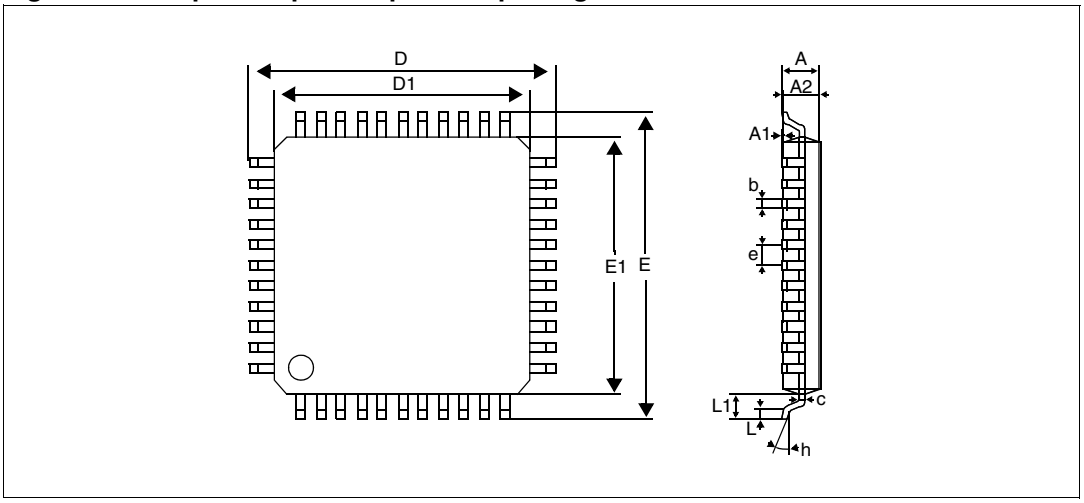


Table 113. 44-pin low profile quad flat package mechanical data

Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			1.60			0.063
A1	0.05		0.15	0.002		0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
b	0.30	0.37	0.45	0.012	0.015	0.018
C	0.09		0.20	0.004	0.000	0.008
D		12.00			0.472	
D1		10.00			0.394	
E		12.00			0.472	
E1		10.00			0.394	
e		0.80			0.031	
θ	0°	3.5°	7°	0°	3.5°	7°

Table 122. Port A and F configuration

PLL	PA3	PF4	PF1	PF0	Clock disturbance
Off	0	1	0	Toggling	Maximum 2 clock cycles lost at each rising or falling edge of PF0
On	0	1	0	1	Maximum 1 clock cycle lost out of every 16

As a consequence, for cycle-accurate operations, these configurations are prohibited in either input or output mode.

Workaround

To avoid this from occurring, it is recommended to connect one of these pins to GND (PF4 or PF0) or V_{DD} (PA3 or PF1).