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	POR, PWM, WDT
Number of I/O	32
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	2.85V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-10°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/st72f324lj2t5tr

PIN DESCRIPTION (Cont'd)

For more details, refer to "ELECTRICAL CHARACTERISTICS" on page 110

Legend / Abbreviations for Table 2:

Type: I = input, O = output, S = supply

In/Output level: C = CMOS
C_T= CMOS with input trigger

Output level: HS = high sink (on N-buffer only)

Port and control configuration:

- Input: float = floating, wpu = weak pull-up, int = interrupt [1], ana = analog ports
- Output: OD = open drain [2], PP = push-pull

Refer to "I/O PORTS" on page 40 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 n°				Pin Name	Type	Level		Port						Main function (after reset)	Alternate Function	
								Input				Output				
LQFP48	LQFP44	LQFP32	SDIP32			Input	Output	float	wpu	int	ana	OD	PP			
7	6	30	1	PB4 (HS)	I/O	C_T	HS	**X**	ei3			X	X	Port B4		
9	7	31	2	PD0/AIN0	I/O	C_T		**X**	X		X	X	X	Port D0	ADC Analog Input 0	
10	8	32	3	PD1/AIN1	I/O	C_T		**X**	X		X	X	X	Port D1	ADC Analog Input 1	
11	9			PD2/AIN2	I/O	C_T		**X**	X		X	X	X	Port D2	ADC Analog Input 2	
12	10			PD3/AIN3	I/O	C_T		**X**	X		X	X	X	Port D3	ADC Analog Input 3	
13	11			PD4/AIN4	I/O	C_T		**X**	X		X	X	X	Port D4	ADC Analog Input 4	
14	12			PD5/AIN5	I/O	C_T		**X**	X		X	X	X	Port D5	ADC Analog Input 5	
15	13	1	4	V_{AREF}	S									Analog Reference Voltage for ADC[5]		
16	14	2	5	V_{SSA}	S									Analog Ground Voltage[5]		
17	15	3	6	PF0/MCO/AIN8	I/O	C_T		**X**	ei1		X	X	X	Port F0	Main clock out (f_{OSC}/2)	ADC Analog Input 8
18	16	4	7	PF1 (HS)/BEEP	I/O	C_T	HS	**X**	ei1			X	X	Port F1	Beep signal output	
19	17			PF2 (HS)	I/O	C_T	HS	**X**		ei1		X	X	Port F2		
20	18	5	8	PF4/OCMP1_A/ AIN10	I/O	C_T		**X**	X		X	X	X	Port F4	Timer A Output Compare 1	ADC Analog Input 10
21	19	6	9	PF6 (HS)/ICAP1_A	I/O	C_T	HS	**X**	X			X	X	Port F6	Timer A Input Capture 1	
22	20	7	10	PF7 (HS)/ EXTCLK_A	I/O	C_T	HS	**X**	X			X	X	Port F7	Timer A External Clock Source	
23	21			V_{DD_0}	S									Digital Main Supply Voltage[5]		
24	22			V_{SS_0}	S									Digital Ground Voltage[5]		
25	23	8	11	PC0/OCMP2_B/ AIN12	I/O	C_T		**X**	X		X	X	X	Port C0	Timer B Output Compare 2	ADC Analog Input 12
26	24	9	12	PC1/OCMP1_B/ AIN13	I/O	C_T		**X**	X		X	X	X	Port C1	Timer B Output Compare 1	ADC Analog Input 13

Pin n°				Pin Name	Type	Level		Port						Main function (after reset)	Alternate Function	
								Input				Output				
LQFP48	LQFP44	LQFP32	SDIP32			Input	Output	float	wpu	int	ana	OD	PP			
27	25	10	13	PC2 (HS)/ICAP2_B	I/O	C_T	HS	**X**	X			X	X	Port C2	Timer B Input Capture 2	
28	26	11	14	PC3 (HS)/ICAP1_B	I/O	C_T	HS	**X**	X			X	X	Port C3	Timer B Input Capture 1	
29	27	12	15	PC4/MISO/ICCDATA	I/O	C_T		**X**	X			X	X	Port C4	SPI Master In / Slave Out Data	ICC Data Input
30	28	13	16	PC5/MOSI/AIN14	I/O	C_T		**X**	X		X	X	X	Port C5	SPI Master Out / Slave In Data	ADC Analog Input 14
31	29	14	17	PC6/SCK/ICCCLK	I/O	C_T		**X**	X			X	X	Port C6	SPI Serial Clock	ICC Clock Output
32	30	15	18	PC7/$\overline{SS}$/AIN15	I/O	C_T		**X**	X		X	X	X	Port C7	SPI Slave Select (active low)	ADC Analog Input 15
34	31	16	19	PA3 (HS)	I/O	C_T	HS	**X**		ei0		X	X	Port A3		
35	32			V_{DD_1}	S									Digital Main Supply Voltage[5]		
36	33			V_{SS_1}	S									Digital Ground Voltage[5]		
37	34	17	20	PA4 (HS)	I/O	C_T	HS	**X**	X			X	X	Port A4		
38	35			PA5 (HS)	I/O	C_T	HS	**X**	X			X	X	Port A5		
39	36	18	21	PA6 (HS)	I/O	C_T	HS	**X**				T		Port A6 [1]		
40	37	19	22	PA7 (HS)	I/O	C_T	HS	**X**				T		Port A7 [1]		
41	38	20	23	V_{PP} /ICCSEL	I									Must be tied low. In the flash programming mode, this pin acts as the programming voltage input V_{PP}. See Section 12.9.2 for more details. High voltage must not be applied to ROM devices.		
42	39	21	24	$\overline{RESET}$	I/O	C_T								Top priority non maskable interrupt.		
43	40	22	25	V_{SS_2}	S									Digital Ground Voltage[5]		
44	41	23	26	OSC2	O									Resonator oscillator inverter output		
45	42	24	27	OSC1	I									External clock input or Resonator oscillator inverter input		
46	43	25	28	V_{DD_2}	S									Digital Main Supply Voltage[5]		
47	44	26	29	PE0/TDO	I/O	C_T		**X**	X			X	X	Port E0	SCI Transmit Data Out	
48	1	27	30	PE1/RDI	I/O	C_T		**X**	X			X	X	Port E1	SCI Receive Data In	
3	2	28	31	PB0	I/O	C_T		**X**	ei2			X	X	Port B0		
4	3			PB1	I/O	C_T		**X**	ei2			X	X	Port B1		
5	4			PB2	I/O	C_T		**X**	ei2			X	X	Port B2		
6	5	29	32	PB3	I/O	C_T		**X**		ei2		X	X	Port B3		

Notes:

1. In the interrupt input column, “eiX” defines the associated external interrupt vector. If the weak pull-up column (wpu) is merged with the interrupt column (int), then the I/O configuration is pull-up interrupt input, else the configuration is floating interrupt input.

2. In the open drain output column, “T” defines a true open drain I/O (P-Buffer and protection diode to V_{DD}

INTERRUPTS (Cont'd)

Table 9. Interrupt Mapping

N°	Source Block	Description	Register Label	Priority Order	Exit from HALT	Exit from Active HALT	Address Vector
	RESET	Reset	N/A		yes	yes	FFFEh-FFFFh
	TRAP	Software interrupt			no	no	FFFCh-FFFDh
0	Not used						FFFAh-FFFBh
1	MCC/RTC	Main clock controller time base interrupt	MCCSR	Higher Priority	no	yes	FFF8h-FFF9h
2	ei0	External interrupt port A3..0	N/A		yes	yes[1)]	FFF6h-FFF7h
3	ei1	External interrupt port F2..0			yes	yes[1)]	FFF4h-FFF5h
4	ei2	External interrupt port B3..0			yes	yes[1)]	FFF2h-FFF3h
5	ei3	External interrupt port B7..4			yes	yes[1)]	FFF0h-FFF1h
6	Not used						FFEEh-FFEFh
7	SPI	SPI peripheral interrupts	SPICSR		yes	yes[1)]	FFECh-FFEDh
8	TIMER A	TIMER A peripheral interrupts	TASR		no	no	FFEAh-FFEBh
9	TIMER B	TIMER B peripheral interrupts	TBSR		no	no	FFE8h-FFE9h
10	SCI	SCI Peripheral interrupts	SCISR	Lower Priority	no	no	FFE6h-FFE7h

Notes:

1. Valid for ROM devices. For Flash devices only a RESET or MCC/RTC interrupt can be used to wake-up from Active Halt mode.

7.6 EXTERNAL INTERRUPTS

7.6.1 I/O Port Interrupt Sensitivity

The external interrupt sensitivity is controlled by the IPA, IPB and ISxx bits of the EICR register (Figure 18). This control allows to have up to 4 fully independent external interrupt source sensitivities.

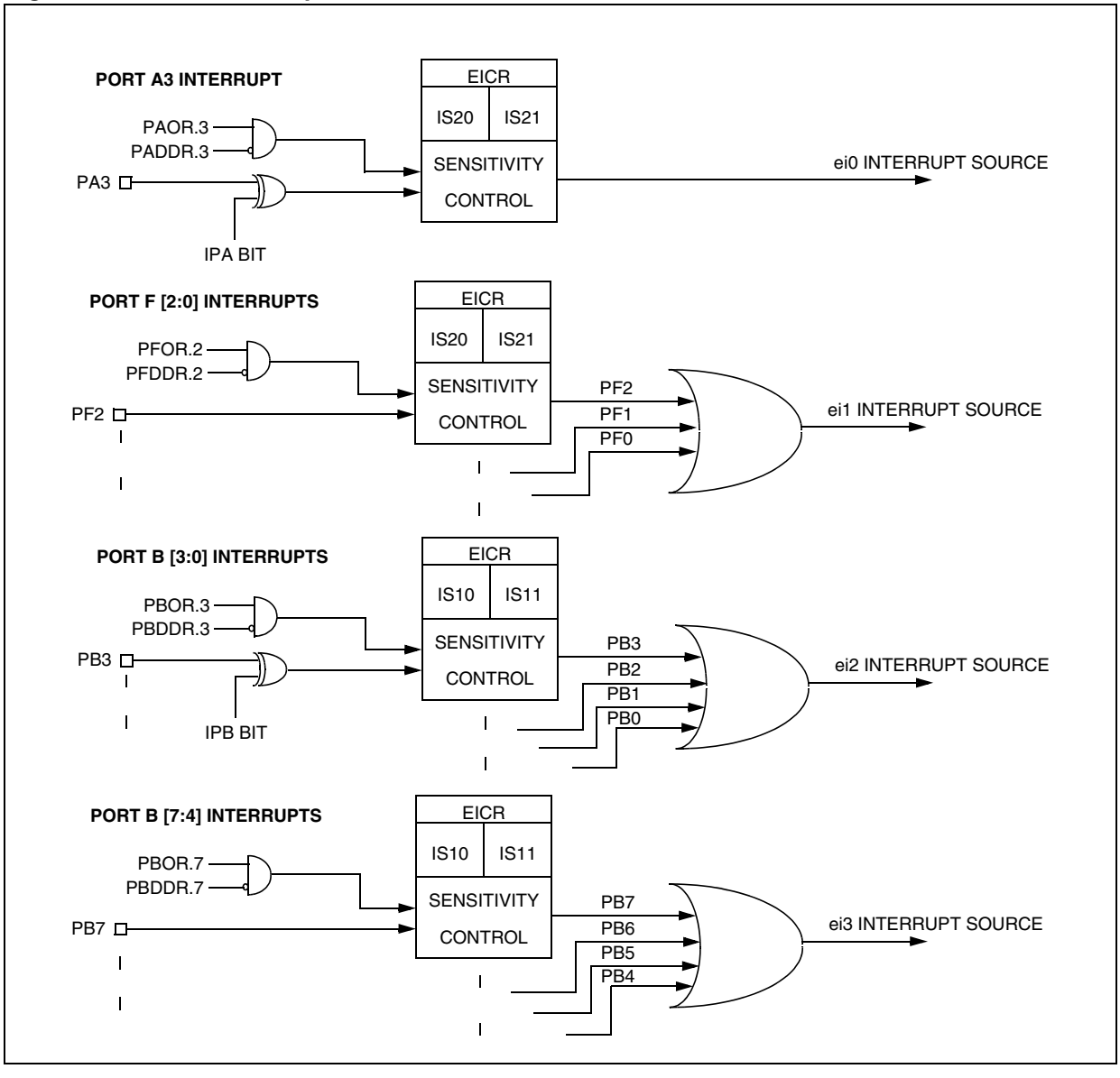
Each external interrupt source can be generated on four (or five) different events on the pin:

- Falling edge
- Rising edge
- Falling and rising edge
- Falling edge and low level
- Rising edge and high level (only for ei0 and ei2)

To guarantee correct functionality, the sensitivity bits in the EICR register can be modified only when the I1 and I0 bits of the CC register are both set to 1 (level 3). This means that interrupts must be disabled before changing sensitivity.

The pending interrupts are cleared when disabling these interrupts by setting their I0_x and I1_x in the matching ISPR

Figure 18. External Interrupt Control bits



I/O PORTS (Cont'd)

9.5.1 I/O Port Implementation

The I/O port register configurations are summarised as follows.

Standard Ports

PA5:4, PC7:0, PD5:0, PE1:0, PF7:6, 4

MODE	DDR	OR
floating input	0	0
pull-up input	0	1
open drain output	1	0
push-pull output	1	1

Interrupt Ports

PB4, PB2:0, PF1:0 (with pull-up)

MODE	DDR	OR
floating input	0	0
pull-up interrupt input	0	1
open drain output	1	0
push-pull output	1	1

PA3, PB3, PF2 (without pull-up)

MODE	DDR	OR
floating input	0	0
floating interrupt input	0	1
open drain output	1	0
push-pull output	1	1

True Open Drain Ports

PA7:6

MODE	DDR
floating input	0
open drain (high sink ports)	1

Table 13. Port Configuration

Port	Pin name	Input		Output	
		OR = 0	OR = 1	OR = 0	OR = 1
Port A	PA7:6	floating		true open-drain	
	PA5:4	floating	pull-up	open drain	push-pull
	PA3	floating	floating interrupt	open drain	push-pull
Port B	PB3	floating	floating interrupt	open drain	push-pull
	PB4, PB2:0	floating	pull-up interrupt	open drain	push-pull
Port C	PC7:0	floating	pull-up	open drain	push-pull
Port D	PD5:0	floating	pull-up	open drain	push-pull
Port E	PE1:0	floating	pull-up	open drain	push-pull
Port F	PF7:6, 4	floating	pull-up	open drain	push-pull
	PF2	floating	floating interrupt	open drain	push-pull
	PF1:0	floating	pull-up interrupt	open drain	push-pull

16-BIT TIMER (Cont'd)

10.3.3.3 Input Capture

In this section, the index, *i*, may be 1 or 2 because there are 2 input capture functions in the 16-bit timer.

The two 16-bit input capture registers (IC1R and IC2R) are used to latch the value of the free running counter after a transition is detected on the ICAP*i* pin (see figure 5).

	MS Byte	LS Byte
ICiR	IC*i*HR	IC*i*LR

IC*i*R register is a read-only register.

The active transition is software programmable through the IEDG*i* bit of Control Registers (CR*i*).

Timing resolution is one count of the free running counter: (f_{CPU}/CC[1:0]).

Procedure:

To use the input capture function select the following in the CR2 register:

– Select the timer clock (CC[1:0]) (see Table 17 Clock Control Bits).

– Select the edge of the active transition on the ICAP2 pin with the IEDG2 bit (the ICAP2 pin must be configured as floating input or input with pull-up without interrupt if this configuration is available).

And select the following in the CR1 register:

– Set the ICIE bit to generate an interrupt after an input capture coming from either the ICAP1 pin or the ICAP2 pin

– Select the edge of the active transition on the ICAP1 pin with the IEDG1 bit (the ICAP1pin must be configured as floating input or input with pull-up without interrupt if this configuration is available).

When an input capture occurs:

– ICF*i* bit is set.

– The IC*i*R register contains the value of the free running counter on the active transition on the ICAP*i* pin (see Figure 37).

– A timer interrupt is generated if the ICIE bit is set and the I bit is cleared in the CC register. Otherwise, the interrupt remains pending until both conditions become true.

Clearing the Input Capture interrupt request (i.e. clearing the ICF*i* bit) is done in two steps:

1. Reading the SR register while the ICF*i* bit is set.
2. An access (read or write) to the IC*i*LR register.

Notes:

1. After reading the IC*i*HR register, transfer of input capture data is inhibited and ICF*i* will never be set until the IC*i*LR register is also read.
2. The IC*i*R register contains the free running counter value which corresponds to the most recent input capture.
3. The 2 input capture functions can be used together even if the timer also uses the 2 output compare functions.
4. In One pulse Mode and PWM mode only Input Capture 2 can be used.
5. The alternate inputs (ICAP1 & ICAP2) are always directly connected to the timer. So any transitions on these pins activates the input capture function.
   Moreover if one of the ICAP*i* pins is configured as an input and the second one as an output, an interrupt can be generated if the user toggles the output pin and if the ICIE bit is set.
   This can be avoided if the input capture function *i* is disabled by reading the IC*i*HR (see note 1).
6. The TOF bit can be used with interrupt generation in order to measure events that go beyond the timer range (FFFFh).
7. In Flash devices, the ICAP2 registers (TAIC2HR, TAIC2LR) are not available on Timer A. The corresponding interrupts cannot be used (ICF2 is forced by hardware to 0).

16-BIT TIMER (Cont'd)

CONTROL REGISTER 2 (CR2)

Read/Write

Reset Value: 0000 0000 (00h)

7							0
OC1E	OC2E	OPM	PWM	CC1	CC0	IEDG2	EXEDG

Bit 7 = **OC1E** *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.

Bit 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.

Note: In Flash devices, this bit is not available for Timer A. It must be kept at its reset value.

Bit 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.

Bit 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.

Bit 3, 2 = **CC[1:0]** *Clock Control.*

The timer clock mode depends on these bits:

Table 17. Clock Control Bits

Timer Clock	CC1	CC0
f_{CPU} / 4	0	0
f_{CPU} / 2	0	1
f_{CPU} / 8	1	0
External Clock (where available)	1	1

Note: If the external clock pin is not available, programming the external clock configuration stops the counter.

Bit 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.

Bit 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.

ALTERNATE COUNTER HIGH REGISTER (ACHR)
Read Only
Reset Value: 1111 1111 (FFh)

This is an 8-bit register that contains the high part of the counter value.

7							0
MSB							LSB

ALTERNATE COUNTER LOW REGISTER (ACLR)
Read Only
Reset Value: 1111 1100 (FCh)

This is an 8-bit register that contains the low part of the counter value. A write to this register resets the counter. An access to this register after an access to CSR register does not clear the TOF bit in the CSR register.

7							0
MSB							LSB

INPUT CAPTURE 2 HIGH REGISTER (IC2HR)
Read Only
Reset Value: Undefined

This is an 8-bit read only register that contains the high part of the counter value (transferred by the Input Capture 2 event).

7							0
MSB							LSB

Note: In Flash devices, this register is not implemented for Timer A.

INPUT CAPTURE 2 LOW REGISTER (IC2LR)
Read Only
Reset Value: Undefined

This is an 8-bit read only register that contains the low part of the counter value (transferred by the Input Capture 2 event).

7							0
MSB							LSB

Note: In Flash devices, this register is not implemented for Timer A.

SERIAL PERIPHERAL INTERFACE (Cont'd)

CONTROL/STATUS REGISTER (SPICSR)
Read/Write (some bits Read Only)
Reset Value: 0000 0000 (00h)

7							0
SPIF	WCOL	OVR	MODF	-	SOD	SSM	SSI

Bit 7 = **SPIF** *Serial Peripheral Data Transfer Flag (Read only).*
This bit is set by hardware when a transfer has been completed. An interrupt is generated if SPIE=1 in the SPICR register. It is cleared by a software sequence (an access to the SPICSR register followed by a write or a read to the SPIDR register).
0: Data transfer is in progress or the flag has been cleared.
1: Data transfer between the device and an external device has been completed.

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

Bit 6 = **WCOL** *Write Collision status (Read only).*
This bit is set by hardware when a write to the SPIDR register is done during a transmit sequence. It is cleared by a software sequence (see Figure 48).
0: No write collision occurred
1: A write collision has been detected

Bit 5 = **OVR** S*PI Overrun error (Read only).*
This bit is set by hardware when the byte currently being received in the shift register is ready to be transferred into the SPIDR register while SPIF = 1 (See Section 10.4.5.2). An interrupt is generated if SPIE = 1 in SPICR register. The OVR bit is cleared by software reading the SPICSR register.
0: No overrun error
1: Overrun error detected

Bit 4 = **MODF** *Mode Fault flag (Read only).*
This bit is set by hardware when the $\overline{SS}$ pin is pulled low in master mode (see Section 10.4.5.1 Master Mode Fault (MODF)). An SPI interrupt can be generated if SPIE=1 in the SPICR register. This bit is cleared by a software sequence (An access to the SPICSR register while MODF=1 followed by a write to the SPICR register).
0: No master mode fault detected
1: A fault in master mode has been detected

Bit 3 = Reserved, must be kept cleared.

Bit 2 = **SOD** *SPI Output Disable.*
This bit is set and cleared by software. When set, it disables the alternate function of the SPI output (MOSI in master mode / MISO in slave mode)
0: SPI output enabled (if SPE=1)
1: SPI output disabled

Bit 1 = **SSM** $\overline{SS}$ *Management.*
This bit is set and cleared by software. When set, it disables the alternate function of the SPI $\overline{SS}$ pin and uses the SSI bit value instead. See Section 10.4.3.2 Slave Select Management.
0: Hardware management ($\overline{SS}$ managed by external pin)
1: Software management (internal $\overline{SS}$ signal controlled by SSI bit. External $\overline{SS}$ pin free for general-purpose I/O)

Bit 0 = **SSI** $\overline{SS}$ *Internal Mode.*
This bit is set and cleared by software. It acts as a 'chip select' by controlling the level of the $\overline{SS}$ slave select signal when the SSM bit is set.
0: Slave selected
1: Slave deselected

DATA I/O REGISTER (SPIDR)
Read/Write
Reset Value: Undefined

7							0
D7	D6	D5	D4	D3	D2	D1	D0

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.

Notes: 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 43).

SERIAL PERIPHERAL INTERFACE (Cont'd)

Table 20. SPI Register Map and Reset Values

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

SERIAL COMMUNICATIONS INTERFACE (Cont'd)

10.5.4.7 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 21.

Table 21. Frame Formats

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 |

Legend: SB = Start Bit, STB = Stop Bit, PB = Parity Bit

Note: In case of wake up by an address mark, the MSB bit of the data is taken into account and not the parity bit

Even parity: the parity bit is calculated to obtain an even number of "1s" inside the frame made of the 7 or 8 LSB bits (depending on whether M is equal to 0 or 1) and the parity bit.

Example: data = 00110101; 4 bits set => parity bit is 0 if even parity is selected (PS bit = 0).

Odd parity: the parity bit is calculated to obtain an odd number of "1s" inside the frame made of the 7 or 8 LSB bits (depending on whether M is equal to 0 or 1) and the parity bit.

Example: data = 00110101; 4 bits set => parity bit is 1 if odd parity is selected (PS bit = 1).

Transmission mode: If the PCE bit is set then the MSB bit of the data written in the data register is not transmitted but is changed by the parity bit.

Reception mode: If the PCE bit is set then the interface checks if the received data byte has an even number of "1s" if even parity is selected (PS = 0) or an odd number of "1s" if odd parity is selected (PS = 1). If the parity check fails, the PE flag is set in the SCISR register and an interrupt is generated if PIE is set in the SCICR1 register.

10.5.4.8 SCI Clock Tolerance

During reception, each bit is sampled 16 times. The majority of the 8th, 9th and 10th samples is considered as the bit value. For a valid bit detection, all the three samples should have the same value otherwise the noise flag (NF) is set. For example: If the 8th, 9th and 10th samples are 0, 1 and 1 respectively, then the bit value is "1", but the Noise Flag bit is set because the three samples values are not the same.

Consequently, the bit length must be long enough so that the 8th, 9th and 10th samples have the desired bit value. This means the clock frequency should not vary more than 6/16 (37.5%) within one bit. The sampling clock is resynchronized at each start bit, so that when receiving 10 bits (one start bit, 1 data byte, 1 stop bit), the clock deviation must not exceed 3.75%.

Note: The internal sampling clock of the microcontroller samples the pin value on every falling edge. Therefore, the internal sampling clock and the time the application expects the sampling to take place may be out of sync. For example: If the baud rate is 15.625 Kbaud (bit length is 64µs), then the 8th, 9th and 10th samples are at 28µs, 32µs and 36µs respectively (the first sample starting ideally at 0µs). But if the falling edge of the internal clock occurs just before the pin value changes, the samples would then be out of sync by ~4us. This means the entire bit length must be at least 40µs (36µs for the 10th sample + 4µs for synchronization with the internal sampling clock).

SERIAL COMMUNICATION INTERFACE (Cont'd)

Table 23. SCI Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0050h	**SCISR**	TDRE	TC	RDRF	IDLE	OR	NF	FE	PE
	Reset Value	1	1	0	0	0	0	0	0
0051h	**SCIDR**	MSB							LSB
	Reset Value	x	x	x	x	x	x	x	x
0052h	**SCIBRR**	SCP1	SCP0	SCT2	SCT1	SCT0	SCR2	SCR1	SCR0
	Reset Value	0	0	0	0	0	0	0	0
0053h	**SCICR1**	R8	T8	SCID	M	WAKE	PCE	PS	PIE
	Reset Value	x	0	0	0	0	0	0	0
0054h	**SCICR2**	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
	Reset Value	0	0	0	0	0	0	0	0
0055h	**SCIERPR**	MSB							LSB
	Reset Value	0	0	0	0	0	0	0	0
0057h	**SCIPETPR**	MSB							LSB
	Reset Value	0	0	0	0	0	0	0	0

10-BIT A/D CONVERTER (ADC) (Cont'd)

10.6.3 Functional Description

The conversion is monotonic, meaning that the result never decreases if the analog input does not and never increases if the analog input does not.

If the input voltage (V_{AIN}) is greater than V_{AREF} (high-level voltage reference) then the conversion result is FFh in the ADCDRH register and 03h in the ADCDRL register (without overflow indication).

If the input voltage (V_{AIN}) is lower than V_{SSA} (low-level voltage reference) then the conversion result in the ADCDRH and ADCDRL registers is 00 00h.

The A/D converter is linear and the digital result of the conversion is stored in the ADCDRH and ADCDRL registers. The accuracy of the conversion is described in the Electrical Characteristics Section.

R_{AIN} is the maximum recommended impedance for an analog input signal. If the impedance is too high, this will result in a loss of accuracy due to leakage and sampling not being completed in the alloted time.

10.6.3.1 A/D Converter Configuration

The analog input ports must be configured as input, no pull-up, no interrupt. Refer to the «I/O ports» chapter. Using these pins as analog inputs does not affect the ability of the port to be read as a logic input.

In the ADCCSR register:

– Select the CS[3:0] bits to assign the analog channel to convert.

10.6.3.2 Starting the Conversion

In the ADCCSR register:

– Set the ADON bit to enable the A/D converter and to start the conversion. From this time on, the ADC performs a continuous conversion of the selected channel.

When a conversion is complete:

– The EOC bit is set by hardware.

– The result is in the ADCDR registers.

A read to the ADCDRH resets the EOC bit.

To read the 10 bits, perform the following steps:

1. Poll the EOC bit
2. Read the ADCDRL register
3. Read the ADCDRH register. This clears EOC automatically.

Note: The data is not latched, so both the low and the high data register must be read before the next conversion is complete, so it is recommended to disable interrupts while reading the conversion result.

To read only 8 bits, perform the following steps:

1. Poll the EOC bit
2. Read the ADCDRH register. This clears EOC automatically.

10.6.3.3 Changing the conversion channel

The application can change channels during conversion. When software modifies the CH[3:0] bits in the ADCCSR register, the current conversion is stopped, the EOC bit is cleared, and the A/D converter starts converting the newly selected channel.

10.6.4 Low Power Modes

Note: The A/D converter may be disabled by resetting the ADON bit. This feature allows reduced power consumption when no conversion is needed.

Mode	Description
WAIT	No effect on A/D Converter
HALT	A/D Converter disabled. After wakeup from Halt mode, the A/D Converter requires a stabilization time t_{STAB} (see Electrical Characteristics) before accurate conversions can be performed.

10.6.5 Interrupts

None.

SUPPLY CURRENT CHARACTERISTICS (Cont'd)

12.4.2 Supply and Clock Managers

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

Symbol	Parameter	Conditions	Typ	Max [1]	Unit
$I_{DD(RCINT)}$	Supply current of internal RC oscillator		625		µA
$I_{DD(RES)}$	Supply current of resonator oscillator [2] & [3]		see Section 12.5.3 on page 117		µA
$I_{DD(PLL)}$	PLL supply current		180		µA

Notes:

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

12.4.3 On-Chip Peripherals

T_A = 25°C f_{CPU}=4MHz.

Symbol	Parameter	Conditions	Typ	Unit
$I_{DD(TIM)}$	16-bit Timer supply current [1]	V_{DD}=3.3V	20	µA
$I_{DD(SPI)}$	SPI supply current [2]		250	
$I_{DD(ADC)}$	ADC supply current when converting [3]		300	

Notes:

1. Data based on a differential I_{DD} measurement between reset configuration (timer counter running at $f_{CPU}/4$) and timer counter stopped (only TIMD bit set). Data valid for one timer.
2. Data based on a differential I_{DD} measurement between reset configuration (SPI disabled) and a permanent SPI master communication at maximum speed (data sent equal to 55h). This measurement includes the pad toggling consumption.
3. Data based on a differential I_{DD} measurement between reset configuration and continuous A/D conversions.

CLOCK CHARACTERISTICS (Cont'd)

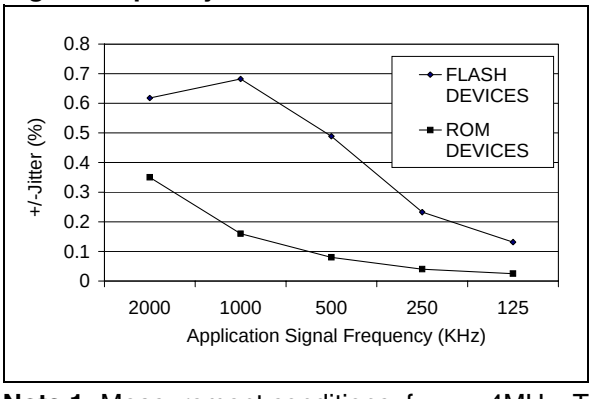
12.5.5 PLL Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{DD(PLL)}$	PLL Operating Range		2.85		3.6	V
f_{OSC}	PLL input frequency range		2		4	MHz
$\Delta f_{CPU}/f_{CPU}$	Instantaneous PLL jitter [1)]	f_{OSC} = 4 MHz. (f_{CPU}=8MHz.)		3.5	5.5	%

Note:

1. Instantaneous PLL jitter is the absolute maximum deviation on a single clock period. Data characterized, not tested in production.

Figure 64. PLL Clock Jitter vs. Application Signal frequency[1]



PLL clock jitter may cause application errors if high frequency signals are input or output by the application (e.g. high speed serial I/O or sampling of high frequency signals).

Using the PLL increases clock jitter, however this is a periodic effect which is absorbed over several CPU cycles. The lower the frequency of the application signal, the less the impact.

Figure 64 shows the effect of jitter (with and without PLL) on application signals in the range 125kHz to 2MHz. At frequencies of less than 125kHz, the jitter is negligible.

Note 1: Measurement conditions: f_{CPU} = 4MHz, T_A= 25°C

12.7.2 Electro Magnetic Interference (EMI)

Based on a simple application running on the product (toggling 2 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.

Symbol	Parameter	Conditions	Monitored Frequency Band	Max vs. [f_{OSC}/f_{CPU}]		Unit
				8/4MHz	16/8MHz	
S_{EMI}	Peak level	**Flash device:** V_{DD}=3.3V, T_A=+25°C, LQFP44 package conforming to SAE J 1752/3	0.1MHz to 30MHz	14	15	dBµV
			30MHz to 130MHz	18	23	
			130MHz to 1GHz	16	22	
			SAE EMI Level	3.0	3.5	-
S_{EMI}	Peak level	**ROM device:** V_{DD}=3.3V, T_A=+25°C, LQFP44 package conforming to SAE J 1752/3	0.1MHz to 30MHz	8	4	dBµV
			30MHz to 130MHz	16	20	
			130MHz to 1GHz	8	14	
			SAE EMI Level	2.5	3.0	-

Notes:

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

I/O PORT PIN CHARACTERISTICS (Cont'd)

12.8.2 Output Driving Current

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

Symbol	Parameter	Conditions		Min	Typ.	Max.	Unit
V_{OL} [1]	Output low level voltage for a standard I/O pin when 8 pins are sunk at same time (see Figure 67 and Figure 70)	V_{DD}=3V	I_{IO}=+2mA		0.3	0.7	V
	Output low level voltage for a high sink I/O pin when 4 pins are sunk at same time (see Figure 68 and Figure 71)		I_{IO}=+10mA		0.3	0.7	
V_{OH} [2]	Output high level voltage for an I/O pin when 4 pins are sourced at same time (see Figure 69 and Figure 72)		I_{IO}=-2mA	V_{DD}-0.9	2.6		

Figure 67. Typical V_{OL} at V_{DD}=3V (std. ports)

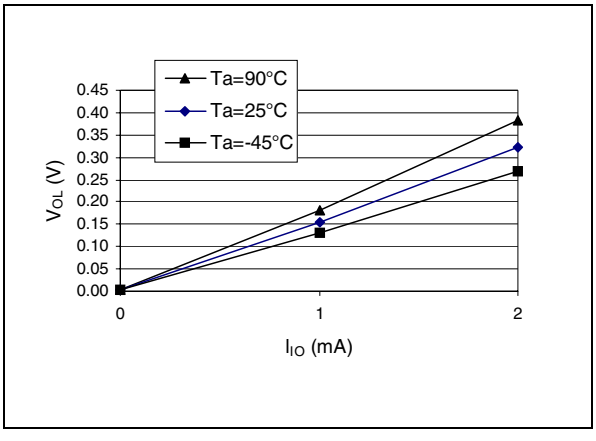


Figure 68. Typ. V_{OL} at V_{DD}=3V (high-sink ports)

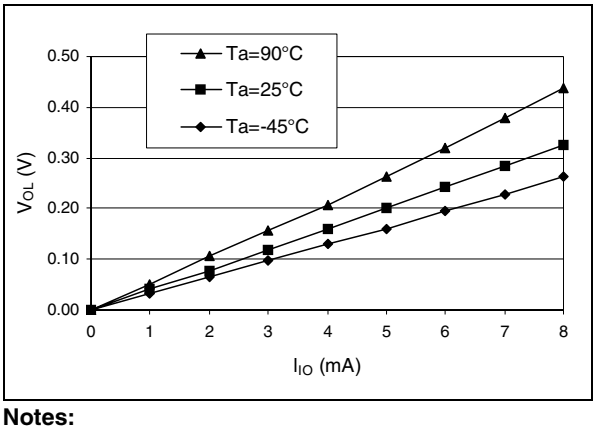
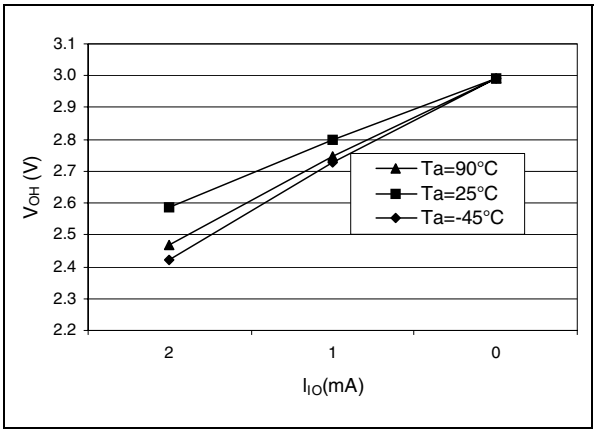


Figure 69. Typical V_{OH} at V_{DD}=3V



Notes:

1. The I_{IO} current sunk must always respect the absolute maximum rating specified in Section 12.2.2 and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VSS}.

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

I/O PORT PIN CHARACTERISTICS (Cont'd)

Figure 70. Typical V_{OL} vs. V_{DD} (std. ports)

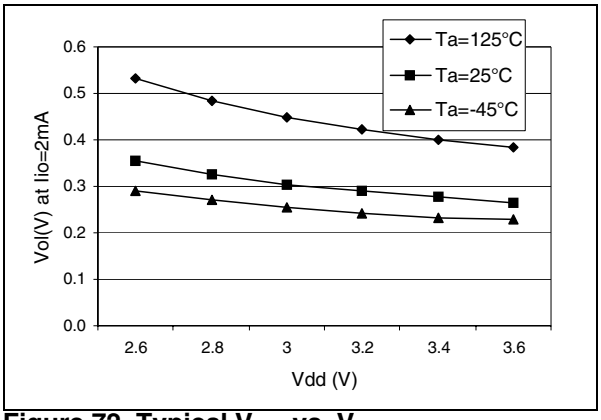


Figure 71. Typical V_{OL} vs. V_{DD} (high-sink ports)

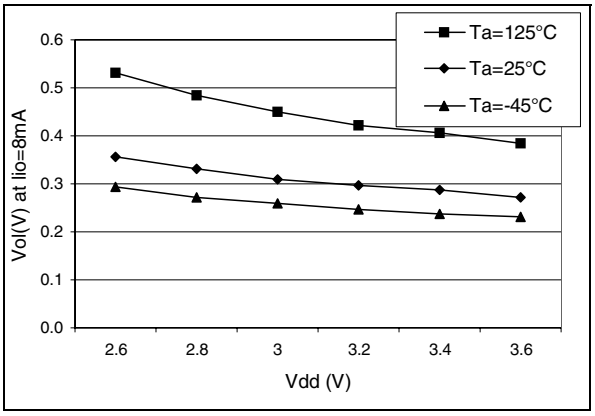
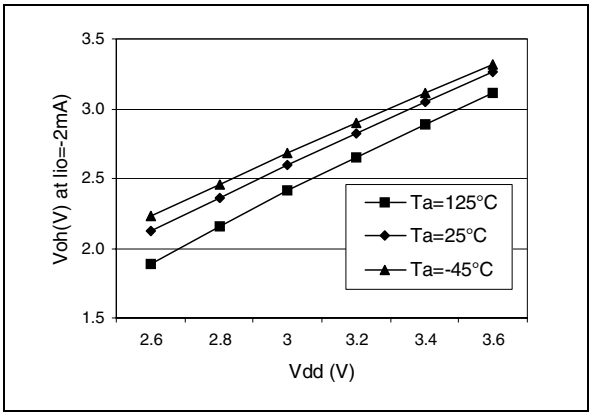


Figure 72. Typical V_{OH} vs. V_{DD}



14 DEVICE CONFIGURATION AND ORDERING INFORMATION

Each device is available for production in user programmable versions (FLASH) as well as in factory coded versions (ROM). ST72324BL devices are ROM versions. ST72P324L devices are Factory Advanced Service Technique ROM (FASTROM) versions: they are factory-programmed HDFlash devices. FLASH devices are shipped to customers with a default content (FFh), while ROM factory coded parts contain the code supplied by the customer. This implies that FLASH devices have to be configured by the customer using the Option Bytes while the ROM devices are factory-configured.

14.1 FLASH OPTION BYTES

	STATIC OPTION BYTE 0								STATIC OPTION BYTE 1							
	7							0	7							0
	WDG		CSS	VD		Reserved	Reserved	FMP_R	PKG1	RSTC	OSCTYPE		OSCRANGE			PLLOFF
	HALT	SW		1	0						1	0	2	1	0	
Default	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1

The option bytes allows the hardware configuration of the microcontroller to be selected. They have no address in the memory map and can be accessed only in programming mode (for example using a standard ST7 programming tool). The default content of the FLASH is fixed to FFh. To program directly the FLASH devices using ICP, FLASH devices are shipped to customers with the internal RC clock source. In masked ROM devices, the option bytes are fixed in hardware by the ROM code (see option list).

OPTION BYTE 0

OPT7= **WDG HALT** *Watchdog reset on HALT*
This option bit determines if a RESET is generated when entering HALT mode while the Watchdog is active.
0: No Reset generation when entering Halt mode
1: Reset generation when entering Halt mode

OPT6= **WDG SW** *Hardware or software watchdog*
This option bit selects the watchdog type.
0: Hardware (watchdog always enabled)
1: Software (watchdog to be enabled by software)

OPT5 = **CSS** *Clock security system on/off*
Reserved in current silicon revision, must be kept at default value.

OPT4:3= **VD[1:0]** *Voltage detection*
Reserved, must be kept at default value.

OPT2:1 = Reserved, must be kept at default value.

OPT0= **FMP_R** *Flash memory read-out protection*
Read-out protection, when selected, provides a protection against Program Memory content extraction and against write access to Flash memory.

Erasing the option bytes when the FMP_R option is selected causes the whole user memory to be erased first, and the device can be reprogrammed. Refer to Section 7.3.1 on page 37 and the ST7 Flash Programming Reference Manual for more details.
0: Read-out protection enabled
1: Read-out protection disabled

15.1.6 SCI Wrong Break duration

Description

A single break character is sent by setting and resetting the SBK bit in the SCICR2 register. In some cases, the break character may have a longer duration than expected:

- 20 bits instead of 10 bits if M=0
- 22 bits instead of 11 bits if M=1.

In the same way, as long as the SBK bit is set, break characters are sent to the TDO pin. This may lead to generate one break more than expected.

Occurrence

The occurrence of the problem is random and proportional to the baudrate. With a transmit frequency of 19200 baud (f_{CPU}=8 MHz and SCIBRR=0xC9), the wrong break duration occurrence is around 1%.

Workaround

If this wrong duration is not compliant with the communication protocol in the application, software can request that an Idle line be generated before the break character. In this case, the break duration is always correct assuming the application is not doing anything between the idle and the break. This can be ensured by temporarily disabling interrupts.

The exact sequence is:

- Disable interrupts
- Reset and Set TE (IDLE request)
- Set and Reset SBK (Break Request)
- Re-enable interrupts

15.1.7 External interrupt missed

To avoid any risk of generating a parasitic interrupt, the edge detector is automatically disabled for one clock cycle during an access to either DDR and OR. Any input signal edge during this period will not be detected and will not generate an interrupt.

This case can typically occur if the application refreshes the port configuration registers at intervals during runtime.

Workaround

The workaround is based on software checking the level on the interrupt pin before and after writing to the PxOR or PxDDR registers. If there is a level change (depending on the sensitivity programmed for this pin) the interrupt routine is invoked using the call instruction with three extra PUSH instructions before executing the interrupt routine (this is to make the call compatible with the IRET instruction at the end of the interrupt service routine).

But detection of the level change does not make sure that edge occurs during the critical 1 cycle duration and the interrupt has been missed. This may lead to occurrence of same interrupt twice (one hardware and another with software call).

To avoid this, a semaphore is set to '1' before checking the level change. The semaphore is changed to level '0' inside the interrupt routine. When a level change is detected, the semaphore status is checked and if it is '1' this means that the last interrupt has been missed. In this case, the interrupt routine is invoked with the call instruction.

There is another possible case that is, if writing to PxOR or PxDDR is done with global interrupts disabled (interrupt mask bit set). In this case, the semaphore is changed to '1' when the level change is detected. Detecting a missed interrupt is done after the global interrupts are enabled (interrupt mask bit reset) and by checking the status of the semaphore. If it is '1' this means that the last interrupt was missed and the interrupt routine is invoked with the call instruction.

To implement the workaround, the following software sequence is to be followed for writing into the PxOR/PxDDR registers. The example is for Port PF1 with falling edge interrupt sensitivity. The software sequence is given for both cases (global interrupt disabled/enabled).

Case 1: Writing to PxOR or PxDDR with Global Interrupts Enabled:

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