E·XFL



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Obsolete
Core Processor	ST7
Core Size	8-Bit
Speed	16MHz
Connectivity	I ² C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	22
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-50
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f264g2m6-tr

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

CENTRAL PROCESSING UNIT (Cont'd)

Condition Code Register (CC)

Read/Write

Reset Value: 111x1xxx



The 8-bit Condition Code register contains the interrupt masks and four flags representative of the result of the instruction just executed. This register can also be handled by the PUSH and POP instructions.

These bits can be individually tested and/or controlled by specific instructions.

Arithmetic Management Bits

Bit $4 = \mathbf{H}$ Half carry.

This bit is set by hardware when a carry occurs between bits 3 and 4 of the ALU during an ADD or ADC instructions. It is reset by hardware during the same instructions.

0: No half carry has occurred.

1: A half carry has occurred.

This bit is tested using the JRH or JRNH instruction. The H bit is useful in BCD arithmetic subroutines.

Bit 2 = N Negative.

This bit is set and cleared by hardware. It is representative of the result sign of the last arithmetic, logical or data manipulation. It's a copy of the result 7^{th} bit.

0: The result of the last operation is positive or null.

1: The result of the last operation is negative

(i.e. the most significant bit is a logic 1).

This bit is accessed by the JRMI and JRPL instructions.

Bit 1 = **Z** Zero.

This bit is set and cleared by hardware. This bit indicates that the result of the last arithmetic, logical or data manipulation is zero.

0: The result of the last operation is different from zero.

1: The result of the last operation is zero.

This bit is accessed by the JREQ and JRNE test instructions.

Bit 0 = C Carry/borrow.

This bit is set and cleared by hardware and software. It indicates an overflow or an underflow has occurred during the last arithmetic operation.

0: No overflow or underflow has occurred.

1: An overflow or underflow has occurred.

This bit is driven by the SCF and RCF instructions and tested by the JRC and JRNC instructions. It is also affected by the "bit test and branch", shift and rotate instructions.

Interrupt Management Bits

Bit 5,3 = 11, 10 Interrupt

The combination of the I1 and I0 bits gives the current interrupt software priority.

Interrupt Software Priority	11	10
Level 0 (main)	1	0
Level 1	0	1
Level 2	0	0
Level 3 (= interrupt disable)	1	1

These two bits are set/cleared by hardware when entering in interrupt. The loaded value is given by the corresponding bits in the interrupt software priority registers (IxSPR). They can be also set/ cleared by software with the RIM, SIM, IRET, HALT, WFI and PUSH/POP instructions.

See the interrupt management chapter for more details.



6.4 SYSTEM INTEGRITY MANAGEMENT (SI)

The System Integrity Management block contains group the Low voltage Detector (LVD) and Auxiliary Voltage Detector (AVD) functions. It is managed by the SICSR register.

Note: A reset can also be triggered following the detection of an illegal opcode or prebyte code. Refer to Section 12.2.1 on page 123 for further details.

6.4.1 Low Voltage Detector (LVD)

The Low Voltage Detector function (LVD) generates a static reset when the V_{DD} supply voltage is below a V_{IT}- reference value. This means that it secures the power-up as well as the power-down keeping the ST7 in reset.

The V_{IT-} reference value for a voltage drop is lower than the V_{IT+} reference value for power-on in order to avoid a parasitic reset when the MCU starts running and sinks current on the supply (hysteresis).

The LVD Reset circuitry generates a reset when V_{DD} is below:

 $-V_{IT+}$ when V_{DD} is rising

 $-V_{IT}$ when V_{DD} is falling

The LVD function is illustrated in Figure 14.

The voltage threshold can be configured by option byte to be low, medium or high.

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

- under full software control
- in static safe reset

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

During a Low Voltage Detector Reset, the RESET pin is held low, thus permitting the MCU to reset other devices.

Notes:

The LVD allows the device to be used without any external RESET circuitry.

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

Use of LVD with capacitive power supply: with this type of power supply, if power cuts occur in the application, it is recommended to pull V_{DD} down to 0V to ensure optimum restart conditions. Refer to circuit example in Figure 91 on page 151 and note 6.

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



Figure 14. Low Voltage Detector vs Reset

INTERRUPTS (Cont'd)

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 the HALT modes (see column "Exit from HALT" in "Interrupt Mapping" table). 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 & NESTED MANAGEMENT

The following 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 this order from the lowest to the highest: MAIN, IT4, IT3, IT2, IT1, IT0. The software priority is given for each interrupt.

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

Note: TLI (Top Level Interrupt) is not available in this product.

Related Documentation

AN1044: Multiple interrupt source management for ST7 MCUs



Figure 18. Concurrent Interrupt Management



		SOFTWARE PRIORITY LEVEL		0I ES
		3	1 1	BYT
	+ + + + IT1	2	0 0	= 20
	* i +	2 1	0 1	ACK
	<u>-</u>	3	1 1	D ST
] IIT4) [.]	3	1 1	USEI
[⊥] <u>MAIN</u>		- MAIN - 3/0 10		-

I/O PORTS (Cont'd)



Figure 27. I/O Port General Block Diagram

Table 7. I/O Port Mode Options

Configuration Mode		Configuration Mode Pull-Up		Diodes		
		Full-Op	r-builei	to V _{DD}	to V _{SS}	
Input	Floating with/without Interrupt	Off	0#			
input	Pull-up with/without Interrupt	On	- Oli	On		
	Push-pull	Off	On	OII	On	
Output	Open Drain (logic level)	Oli	Off			
	True Open Drain	NI	NI	NI (see note)		

Legend:NI - not implemented

- Off implemented not activated
- On implemented and activated

Note: The diode to V_{DD} is not implemented in the true open drain pads. A local protection between the pad and V_{OL} is implemented to protect the device against positive stress.

I/O PORTS (Cont'd)

Analog alternate function

Configure the I/O as floating input to use an ADC input. The analog multiplexer (controlled by the ADC registers) switches the analog voltage present on the selected pin to the common analog rail, connected to the ADC input.

Analog Recommendations

Do not change the voltage level or loading on any I/O while conversion is in progress. Do not 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 I/O port features such as ADC input or 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 28. Other transitions are potentially risky and should be avoided, since they may present unwanted side-effects such as spurious interrupt generation.

Figure 28. Interrupt I/O Port State Transitions



9.4 UNUSED I/O PINS

Unused I/O pins must be connected to fixed voltage levels. Refer to Section 13.8.

9.5 LOW POWER MODES

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.					

9.6 INTERRUPTS

The external interrupt event generates an interrupt if the corresponding configuration is selected with DDR and OR registers and if the I bit in the CC register is cleared (RIM instruction).

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
External interrupt on selected external event	-	DDRx ORx	Yes	Yes

Related Documentation

AN 970: SPI Communication between ST7 and EEPROM

AN1045: S/W implementation of I2C bus master AN1048: Software LCD driver

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0024h	WDGCR	WDGA	T6	T5	T4	T3	T2	T1	Т0
	Reset Value	0	1	1	1	1	1	1	1

Table 12. Watchdog Timer Register Map and Reset Values



16-BIT TIMER (Cont'd)

3. In PWM mode the ICAP1 pin can not be used to perform input capture because it is disconnected to the timer. The ICAP2 pin can be used to perform input capture (ICF2 can be set and IC2R can be loaded) but the user must take care that the counter is reset each period and

11.3.4 Low Power Modes

ICF1 can also generates interrupt if ICIE is set.

4. When the Pulse Width Modulation (PWM) and One Pulse Mode (OPM) bits are both set, the PWM mode is the only active one.

Mode	Description
WAIT	No effect on 16-bit Timer.
WAII	Timer interrupts cause the Device to exit from WAIT mode.
	16-bit Timer registers are frozen.
HALT	In HALT mode, the counter stops counting until Halt mode is exited. Counting resumes from the previous count when the Device is woken up by an interrupt with "exit from HALT mode" capability or from the counter reset value when the Device is woken up by a RESET.
	If an input capture event occurs on the ICAP <i>i</i> pin, the input capture detection circuitry is armed. Consequent- ly, when the Device is woken up by an interrupt with "exit from HALT mode" capability, the ICF <i>i</i> bit is set, and the counter value present when exiting from HALT mode is captured into the IC <i>i</i> R register.

11.3.5 Interrupts

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
Input Capture 1 event/Counter reset in PWM mode	ICF1		Yes	No
Input Capture 2 event	ICF2	ICIL	Yes	No
Output Compare 1 event (not available in PWM mode)	OCF1	OCIE	Yes	No
Output Compare 2 event (not available in PWM mode)	OCF2		Yes	No
Timer Overflow event	TOF	TOIE	Yes	No

Note: The 16-bit Timer interrupt events are connected to the same interrupt vector (see Interrupts chapter). These events generate an interrupt if the corresponding Enable Control Bit is set and the interrupt mask in the CC register is reset (RIM instruction).

11.3.6 Summary of Timer modes

MODES	AVAILABLE RESOURCES							
MODES	Input Capture 1	Input Capture 2	Output Compare 1	Output Compare 2				
Input Capture (1 and/or 2)	Yes	Yes	Yes	Yes				
Output Compare (1 and/or 2)	Yes	Yes	Yes	Yes				
One Pulse Mode	No	Not Recommended ¹⁾	No	Partially ²⁾				
PWM Mode	No	Not Recommended ³⁾	No	No				

¹⁾ See note 4 in Section 11.3.3.5 "One Pulse Mode" on page 64

²⁾ See note 5 in Section 11.3.3.5 "One Pulse Mode" on page 64

³⁾ See note 4 in Section 11.3.3.6 "Pulse Width Modulation Mode" on page 66

SERIAL PERIPHERAL INTERFACE (Cont'd)

11.4.5.4 Single Master and Multimaster Configurations

There are two types of SPI systems:

- Single Master System
- Multimaster System

Single Master System

A typical single master system may be configured, using a device as the master and four devices as slaves (see Figure 52).

The master device selects the individual slave devices by using four pins of a parallel port to control the four SS pins of the slave devices.

The \overline{SS} pins are pulled high during reset since the master device ports will be forced to be inputs at that time, thus disabling the slave devices.

Note: To prevent a bus conflict on the MISO line the master allows only one active slave device during a transmission.

For more security, the slave device may respond to the master with the received data byte. Then the master will receive the previous byte back from the slave device if all MISO and MOSI pins are connected and the slave has not written to its SPIDR register.

Other transmission security methods can use ports for handshake lines or data bytes with command fields.

Multi-Master System

A multi-master system may also be configured by the user. Transfer of master control could be implemented using a handshake method through the I/O ports or by an exchange of code messages through the serial peripheral interface system.

The multi-master system is principally handled by the MSTR bit in the SPICR register and the MODF bit in the SPICSR register.





SERIAL PERIPHERAL INTERFACE (Cont'd) 11.4.8 Register Description CONTROL REGISTER (SPICR)

Read/Write

Reset Value: 0000 xxxx (0xh)

7							0
SPIE	SPE	SPR2	MSTR	CPOL	CPHA	SPR1	SPR0

Bit 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 an End of Transfer event, Master Mode Fault or Overrun error occurs (SPIF=1, MODF=1 or OVR=1 in the SPICSR register)

Bit 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 Section 11.4.5.1 "Master Mode Fault (MODF)" on page 81). 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

Bit 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 16 SPI Master mode SCK Frequency.

0: Divider by 2 enabled

1: Divider by 2 disabled

Note: This bit has no effect in slave mode.

Bit 4 = MSTR Master Mode.

This bit is set and cleared by software. It is also cleared by hardware when, in master mode, SS=0 (see Section 11.4.5.1 "Master Mode Fault (MODF)" on page 81).

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.

Bit 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.

Bit 2 = CPHA Clock Phase.

This bit is set and cleared by software.

- 0: The first clock transition is the first data capture edge.
- 1: The second clock transition is the first capture edge.

Note: The slave must have the same CPOL and CPHA settings as the master.

Bits 1:0 = SPR[1:0] Serial Clock Frequency.

These bits are set and cleared by software. Used with the SPR2 bit, they select the baud rate of the SPI serial clock SCK output by the SPI in master mode.

Note: These 2 bits have no effect in slave mode.

Table 16. SPI Master mode SCK Frequency

Serial Clock	SPR2	SPR1	SPR0
f _{CPU} /4	1	0	0
f _{CPU} /8	0	0	0
f _{CPU} /16	0	0	1
f _{CPU} /32	1	1	0
f _{CPU} /64	0	1	0
f _{CPU} /128	0	1	1

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 51).

0: No write collision occurred

1: A write collision has been detected

Bit 5 = **OVR** SPI 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 11.4.5.2). An interrupt is generated if SPIE = 1 in the 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 SS pin is pulled low in master mode (see Section 11.4.5.1 "Master Mode Fault (MODF)" on page 81). 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** *SS Management*.

This bit is set and cleared by software. When set, it disables the alternate function of the SPI SS pin and uses the SSI bit value instead. See Section 11.4.3.2 "Slave Select Management" on page 78.

- 0: Hardware management (SS managed by external pin)
- 1: Software management (internal SS signal controlled by SSI bit. External SS pin free for general-purpose I/O)

Bit 0 = SSI <u>SS</u> 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 46).

57

SERIAL COMMUNICATIONS INTERFACE (Cont'd)

11.5.4.7 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 18.

Table 18. Frame Form	nats
----------------------	------

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.

Ex: data=00110101; 4 bits set => parity bit will be 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.

Ex: data=00110101; 4 bits set => parity bit will be 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.

11.5.4.8 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 will be "1", but the Noise Flag bit is be 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 will be at 28µs, 32µs & 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 COMMUNICATIONS INTERFACE (Cont'd) CONTROL REGISTER 1 (SCICR1)

Read/Write

Reset Value: x000 0000 (x0h)

7							0
R8	Т8	SCID	М	WAKE	PCE	PS	PIE

Bit 7 = R8 Receive data bit 8.

This bit is used to store the 9th bit of the received word when M=1.

Bit 6 = T8 Transmit data bit 8.

This bit is used to store the 9th bit of the transmitted word when M=1.

Bit 5 = **SCID** *Disabled for low power consumption* When this bit is set the SCI prescalers and outputs are stopped and the end of the current byte transfer in order to reduce power consumption. This bit is set and cleared by software.

0: SCI enabled

1: SCI prescaler and outputs disabled

Bit $4 = \mathbf{M}$ Word length. This bit determines the word length. It is set or cleared by software.

0: 1 Start bit, 8 Data bits, 1 Stop bit

1: 1 Start bit, 9 Data bits, 1 Stop bit

Note: The M bit must not be modified during a data transfer (both transmission and reception).

Bit 3 = WAKE Wake-Up method.

This bit determines the SCI Wake-Up method, it is set or cleared by software. 0: Idle Line 1: Address Mark

Bit 2 = **PCE** *Parity control enable.*

This bit selects the hardware parity control (generation and detection). When the parity control is enabled, the computed parity is inserted at the MSB position (9th bit if M=1; 8th bit if M=0) and parity is checked on the received data. This bit is set and cleared by software. Once it is set, PCE is active after the current byte (in reception and in transmission).

0: Parity control disabled

1: Parity control enabled

Bit 1 = **PS** Parity selection.

This bit selects the odd or even parity when the parity generation/detection is enabled (PCE bit set). It is set and cleared by software. The parity will be selected after the current byte.

0: Even parity

1: Odd parity

Bit 0 = **PIE** Parity interrupt enable.

This bit enables the interrupt capability of the hardware parity control when a parity error is detected (PE bit set). It is set and cleared by software.

0: Parity error interrupt disabled

1: Parity error interrupt enabled.

I²C INTERFACE (Cont'd)

How to release the SDA / SCL lines

Set and subsequently clear the STOP bit while BTF is set. The SDA/SCL lines are released after the transfer of the current byte.

SMBus Compatibility

ST7 I²C is compatible with SMBus V1.1 protocol. It supports all SMBus adressing modes, SMBus bus protocols and CRC-8 packet error checking. Refer to AN1713: SMBus Slave Driver For ST7 I²C Peripheral.

11.6.4.2 Master Mode

To switch from default Slave mode to Master mode a Start condition generation is needed.

Start condition

Setting the START bit while the BUSY bit is cleared causes the interface to switch to Master mode (M/SL bit set) and generates a Start condition.

Once the Start condition is sent:

 The EVF and SB bits are set by hardware with an interrupt if the ITE bit is set.

Then the master waits for a read of the SR1 register followed by a write in the DR register with the Slave address, **holding the SCL line low** (see Figure 59 Transfer sequencing EV5).

Slave address transmission

Then the slave address is sent to the SDA line via the internal shift register.

In 7-bit addressing mode, one address byte is sent.

In 10-bit addressing mode, sending the first byte including the header sequence causes the following event:

 The EVF bit is set by hardware with interrupt generation if the ITE bit is set.

Then the master waits for a read of the SR1 register followed by a write in the DR register, **holding the SCL line low** (see Figure 59 Transfer sequencing EV9). Then the second address byte is sent by the interface.

After completion of this transfer (and acknowledge from the slave if the ACK bit is set):

 The EVF bit is set by hardware with interrupt generation if the ITE bit is set.

Then the master waits for a read of the SR1 register followed by a write in the CR register (for example set PE bit), **holding the SCL line low** (see Figure 59 Transfer sequencing EV6).

Next the master must enter Receiver or Transmitter mode.

Note: In 10-bit addressing mode, to switch the master to Receiver mode, software must generate a repeated Start condition and resend the header sequence with the least significant bit set (11110xx1).

Master Receiver

Following the address transmission and after SR1 and CR registers have been accessed, the master receives bytes from the SDA line into the DR register via the internal shift register. After each byte the interface generates in sequence:

- Acknowledge pulse if the ACK bit is set
- EVF and BTF bits are set by hardware with an interrupt if the ITE bit is set.

Then the interface waits for a read of the SR1 register followed by a read of the DR register, **holding the SCL line low** (see Figure 59 Transfer sequencing EV7).

To close the communication: before reading the last byte from the DR register, set the STOP bit to generate the Stop condition. The interface goes automatically back to slave mode (M/SL bit cleared).

Note: In order to generate the non-acknowledge pulse after the last received data byte, the ACK bit must be cleared just before reading the second last data byte.

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

11.7.3.2 Digital A/D Conversion Result

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_{DDA} (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 AD-CDRL 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.

11.7.3.3 A/D Conversion

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 CH[2:0] bits to assign the analog channel to convert.

ADC Conversion mode

In the ADCCSR register:

- Set the SPEED or the SLOW bits
- 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 or a write to any bit of the ADCCSR resets the EOC bit.

To read the 10 bits, perform the following steps:

- 1. Poll EOC bit
- 2. Read ADCDRL. This locks the ADCDRH until it is read.
- 3. Read ADCDRH. This clears EOC automatically.

To read only 8 bits, perform the following steps:

- 1. Poll EOC bit
- 2. Read ADCDRH. This clears EOC automatically.

11.7.4 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 and between single shot conversions.

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

11.7.5 Interrupts

None.

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

11.7.6 Register Description

CONTROL/STATUS REGISTER (ADCCSR)

Read/Write (Except bit 7 read only)

Reset Value: 0000 0000 (00h)

7							0
EOC	SPEED	ADON	SLOW	0	CH2	CH1	CH0

Bit 7 = **EOC** End of Conversion

This bit is set by hardware. It is cleared by software reading the ADCDRH register or writing to any bit of the ADCCSR register.

0: Conversion is not complete

1: Conversion complete

Bit 6 = **SPEED** A/D clock selection

This bit is set and cleared by software.

Table 22. A/D Clock Selection (See Note 1)

f _{ADC} Frequency	SLOW	SPEED
f _{CPU} (See Note 2)	0	1
f/2	1	1
ICPU/2	0	0
f _{CPU} /4	1	0

¹⁾The SPEED and SLOW bits must be updated before setting the ADON bit.

²⁾Use this setting only if $f_{CPU} \le 4$ MHz

Bit 5 = ADON A/D Converter on

This bit is set and cleared by software. 0: Disable ADC and stop conversion 1: Enable ADC and start conversion

Bit 4 = **SLOW** A/D Clock Selection This bit is set and cleared by software. It works to-

gether with the SPEED bit. Refer to Table 22.

Bit 2:0 = CH[2:0] Channel Selection

These bits are set and cleared by software. They select the analog input to convert.

Channel Pin	CH2	CH1	CH0
AINO	0	0	0
AIN1	0	0	1
AIN2	0	1	0
AIN3	0	1	1
AIN4	1	0	0
AIN5	1	0	1

DATA REGISTER (ADCDRH)

Read Only

Reset Value: 0000 0000 (00h)

7							0
D9	D8	D7	D6	D5	D4	D3	D2

Bit 7:0 = D[9:2] MSB of Analog Converted Value

DATA REGISTER (ADCDRL)

Read Only

Reset Value: 0000 0000 (00h)

7							0
0	0	0	0	0	0	D1	D0

Bit 7:2 = Reserved. Forced by hardware to 0.

Bit 1:0 = **D**[1:0] *LSB of Analog Converted Value*



CLOCK CHARACTERISTICS (Cont'd) 13.5.5 PLL Characteristics

Symbol	Parameter	Conditions	Min	Тур	Max	Unit
V _{DD(PLL)}	BLL Operating Banga	T _A 0 to 70°C	3.5		5.5	V
	FLL Operating Range	T _A -40 to +85°C	4.5		5.5	v
f _{OSC}	PLL input frequency range		2		4	MHz
$\Delta f_{CPU}/f_{CPU}$	Instantanceus PLL iittor ¹⁾	f _{OSC} = 4 MHz.		1.0	2.5	%
		f _{OSC} = 2 MHz.		2.5	4.0	%

Note:

5/

1. Data characterized but not tested.



Figure 74. PLL Jitter vs. Signal frequency¹

The user must take the PLL jitter into account in the application (for example in serial communication or sampling of high frequency signals). The PLL jitter is a periodic effect, which is integrated over several CPU cycles. Therefore the longer the period of the application signal, the less it will be impacted by the PLL jitter.

Figure 74 shows the PLL jitter integrated on application signals in the range 125kHz to 2MHz. At frequencies of less than 125KHz, the jitter is negligible.

Note 1: Measurement conditions: $f_{CPU} = 4MHz$, $T_A = 25^{\circ}C$

COMMUNICATION INTERFACE CHARACTERISTICS (Cont'd)



Figure 94. SPI Slave Timing Diagram with CPHA=1¹⁾

Figure 95. SPI Master Timing Diagram 1)



Notes:

1. Measurement points are done at CMOS levels: $0.3 x V_{\text{DD}}$ and $0.7 x V_{\text{DD}}.$

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 of the I/O port configuration.

لركم



Figure 104. Low Profile Fine Pitch Ball Grid Array Package

14.2 THERMAL CHARACTERISTICS

Symbol	Ratings	Value	Unit
R _{thJA}	Package thermal resistance (junction to ambient)		
	SDIP32	60	
	SO28	75	°C/W
	LFBGA 6x6 (on multilayer PCB)	56	
	LFBGA 6x6 (on single-layer PCB)	72	
PD	Power dissipation 1)	500	mW
T _{Jmax}	Maximum junction temperature ²⁾	150	°C

Notes:

1. The power dissipation is obtained from the formula $P_D = P_{INT} + P_{PORT}$ where P_{INT} is the chip internal power ($I_{DD}xV_{DD}$) and P_{PORT} is the port power dissipation determined by the user.

2. The average chip-junction temperature can be obtained from the formula $T_J = T_A + P_D x RthJA$.



15.3 DEVELOPMENT TOOLS

STMicroelectronics offers a range of hardware and software development tools for the ST7 microcontroller family. Full details of tools available for the ST7 from third party manufacturers can be obtained from the STMicroelectronics Internet site: http://www.st.com.

Tools from these manufacturers include C compliers, evaluation tools, emulators and programmers.

Emulators

Two types of emulators are available from ST for the ST72260/262/264 family:

- ST7 DVP3 entry-level emulator offers a flexible and modular debugging and programming solution.
- ST7 EMU3 high-end emulator is delivered with everything (probes, TEB, adapters etc.) needed to start emulating the ST72260/262/264. To configure it to emulate other ST7 subfamily devices, the active probe for the ST7EMU3 can be changed and the ST7EMU3 probe is designed for easy interchange of TEBs (Target Emulation Board). See Table 29.

In-circuit Debugging Kit

Three configurations are available from ST:

 ST7F264-IND/USB: Low-cost In-Circuit Debugging kit from Softec Microsystems.

Table 29. STMicroelectronics Development Tools

Includes STX-InDART/USB board (USB port) and one specific evaluation board for ST72264 (package SDIP32)

- ST7F264-INDART: Low-cost In-Circuit Debugging kit from Softec Microsystems Includes STX-InDART/USB board (parallel port) and one specific evaluation board for ST72264 (package SDIP32)
- STxF-INDART/USB

Flash Programming tools

- ST7-STICK ST7 In-circuit Communication Kit, a complete software/hardware package for programming ST7 Flash devices. It connects to a host PC parallel port and to the target board or socket board via ST7 ICC connector.
- ICC Socket Boards provide an easy to use and flexible means of programming ST7 Flash devices. They can be connected to any tool that supports the ST7 ICC interface, such as ST7 EMU3, ST7-DVP3, inDART, ST7-STICK, or many third-party development tools.

Evaluation boards

One evaluation tool is available from ST:

 ST7FOPTIONS-EVAL: ST7 Clock Security System evaluation board

Supported Products	Emulation				Programming
	ST7 DVP3 Series		ST7 EMU3 series		
	Emulator	Connection kit	Emulator	Active Probe & T.E.B.	ICC Socket Board
ST7226xGx	ST7MDT10- DVP3 ¹⁾	ST7MDT10-32/DVP	ST7MDT10- EMU3 ¹⁾	ST7MDT10-TEB	ST7SB10-26x ²⁾

Notes:

1. BGA adapter not available for ST7MDT10-DVP3 and ST7MDT10-EMU3 .

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

15.3.1 Related Documentation

AN 978: Key features of the STVD7 ST7 Visual Debug Package

AN 983: Key Features of the Cosmic ST7 C-Compiler Package

AN 988: Getting started with ST7 Assembly Tool chain

AN 989: Getting started with ST7 Hiware C Toolchain

AN1604: How to use ST7MDT1-TRAIN with ST72F264

bit) accuracy does not meet the accuracy specified in the data sheet.

Workaround

In order to have the accuracy specified in the datasheet, the first conversion after a ADC switch-on has to be ignored.

16.2.11 Negative injection impact on ADC accuracy

Injecting a negative current on an analog input pins significantly reduces the accuracy of the AD Converter. Whenever necessary, the negative injection should be prevented by the addition of a Schottky diode between the concerned I/Os and ground. Injecting a negative current on digital input pins degrades ADC accuracy especially if performed on a pin close to ADC channel in use.

16.2.12 ADC conversion spurious results

Spurious conversions occur with a rate lower than 50 per million. Such conversions happen when the measured voltage is just between 2 consecutive digital values.

Workaround

A software filter should be implemented to remove erratic conversion results whenever they may cause unwanted consequences.