



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	Not For New Designs
Core Processor	ST7
Core Size	8-Bit
Speed	8MHz
Connectivity	I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	24
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 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f325k4t3tr

Email: info@E-XFL.COM

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

POWER SAVING MODES (Cont'd)

8.4.2.1 Halt Mode Recommendations

- Make sure that an external event is available to wake up the microcontroller from Halt mode.
- When using an external interrupt to wake up the microcontroller, reinitialize the corresponding I/O as "Input Pull-up with Interrupt" before executing the HALT instruction. The main reason for this is that the I/O may be wrongly configured due to external interference or by an unforeseen logical condition.
- For the same reason, reinitialize the level sensitiveness of each external interrupt as a precautionary measure.
- The opcode for the HALT instruction is 0x8E. To avoid an unexpected HALT instruction due to a program counter failure, it is advised to clear all occurrences of the data value 0x8E from memo-

ry. For example, avoid defining a constant in ROM with the value 0x8E.

 As the HALT instruction clears the interrupt mask in the CC register to allow interrupts, the user may choose to clear all pending interrupt bits before executing the HALT instruction. This avoids entering other peripheral interrupt routines after executing the external interrupt routine corresponding to the wake-up event (reset or external interrupt).

Related Documentation

AN 980: ST7 Keypad Decoding Techniques, Implementing Wake-Up on Keystroke

AN1014: How to Minimize the ST7 Power Consumption

AN1605: Using an active RC to wakeup the ST7LITE0 from power saving mode

16-BIT TIMER (Cont'd)

Figure 45. Timer Block Diagram



16-BIT TIMER (Cont'd)

<u>ل</u>حک







SERIAL PERIPHERAL INTERFACE (Cont'd)

10.5.4 Clock Phase and Clock Polarity

Four possible timing relationships may be chosen by software, using the CPOL and CPHA bits (See Figure 60).

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

The combination of the CPOL clock polarity and CPHA (clock phase) bits selects the data capture clock edge

Figure 60, shows an SPI transfer with the four combinations of the CPHA and CPOL bits. The diagram may be interpreted as a master or slave timing diagram where the SCK pin, the MISO pin, the MOSI pin are directly connected between the master and the slave device.

Note: If CPOL is changed at the communication byte boundaries, the SPI must be disabled by resetting the SPE bit.

LSBit

L\$Bit

<u>/</u>۲۸



Figure 60. Data Clock Timing Diagram



CPHA =1

SERIAL PERIPHERAL INTERFACE (Cont'd)

10.5.6 Low Power Modes

<u>ل</u>رک

Mode	Description
WAIT	No effect on SPI. SPI interrupt events cause the device to exit from WAIT mode.
HALT	SPI registers are frozen. In HALT mode, the SPI is inactive. SPI oper- ation resumes when the MCU is woken up by an interrupt with "exit from HALT mode" ca- pability. The data received is subsequently read from the SPIDR register when the soft- ware is running (interrupt vector fetching). If several data are received before the wake- up event, then an overrun error is generated. This error can be detected after the fetch of the interrupt routine that woke up the device.

10.5.6.1 Using the SPI to wakeup the MCU from Halt mode

In slave configuration, the SPI is able to wakeup the ST7 device from HALT mode through a SPIF interrupt. The data received is subsequently read from the SPIDR register when the software is running (interrupt vector fetch). If multiple data transfers have been performed before software clears the SPIF bit, then the OVR bit is set by hardware. **Note:** When waking up from Halt mode, if the SPI remains in Slave mode, it is recommended to perform an extra communications cycle to bring the SPI from Halt mode state to normal state. If the SPI exits from Slave mode, it returns to normal state immediately.

Caution: The SPI can wake up the ST7 from Halt mode only if the Slave Select signal (external SS pin or the SSI bit in the SPICSR register) is low when the ST7 enters Halt mode. So if Slave selection is configured as external (see Section 10.5.3.2), make sure the master drives a low level on the SS pin when the slave enters Halt mode.

10.5.7 Interrupts

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
SPI End of Transfer Event	SPIF		Yes	Yes
Master Mode Fault Event	MODF	SPIE	Yes	No
Overrun Error	OVR		Yes	No

Note: The SPI interrupt events are connected to the same interrupt vector (see Interrupts chapter). They generate an interrupt if the corresponding Enable Control Bit is set and the interrupt mask in

SERIAL COMMUNICATIONS INTERFACE (Cont'd)

10.6.4.3 Receiver

The SCI can receive data words of either 8 or 9 bits. When the M bit is set, word length is 9 bits and the MSB is stored in the R8 bit in the SCICR1 register.

Character reception

During a SCI reception, data shifts in least significant bit first through the RDI pin. In this mode, the SCIDR register consists or a buffer (RDR) between the internal bus and the received shift register (see Figure 1.).

Procedure

- Select the M bit to define the word length.
- Select the desired baud rate using the SCIBRR and the SCIERPR registers.
- Set the RE bit, this enables the receiver which begins searching for a start bit.

When a character is received:

- The RDRF bit is set. It indicates that the content of the shift register is transferred to the RDR.
- An interrupt is generated if the RIE bit is set and the I bit is cleared in the CCR register.
- The error flags can be set if a frame error, noise or an overrun error has been detected during reception.

Clearing the RDRF bit is performed by the following software sequence done by:

- 1. An access to the SCISR register
- 2. A read to the SCIDR register.

The RDRF bit must be cleared before the end of the reception of the next character to avoid an overrun error.

Break Character

When a break character is received, the SCI handles it as a framing error.

Idle Character

When a idle frame is detected, there is the same procedure as a data received character plus an interrupt if the ILIE bit is set and the I bit is cleared in the CCR register.

Overrun Error

An overrun error occurs when a character is received when RDRF has not been reset. Data can not be transferred from the shift register to the RDR register as long as the RDRF bit is not cleared.

When an overrun error occurs:

- The OR bit is set.
- The RDR content is not lost.
- The shift register is overwritten.
- An interrupt is generated if the RIE bit is set and the I bit is cleared in the CCR register.

The OR bit is reset by an access to the SCISR register followed by a SCIDR register read operation.

Noise Error

Oversampling techniques are used for data recovery by discriminating between valid incoming data and noise. Normal data bits are considered valid if three consecutive samples (8th, 9th, 10th) have the same bit value, otherwise the NF flag is set. In the case of start bit detection, the NF flag is set on the basis of an algorithm combining both valid edge detection and three samples (8th, 9th, 10th). Therefore, to prevent the NF flag getting set during start bit reception, there should be a valid edge detection as well as three valid samples.

When noise is detected in a frame:

- The NF flag is set at the rising edge of the RDRF bit.
- Data is transferred from the Shift register to the SCIDR register.
- No interrupt is generated. However this bit rises at the same time as the RDRF bit which itself generates an interrupt.

The NF flag is reset by a SCISR register read operation followed by a SCIDR register read operation.

During reception, if a false start bit is detected (e.g. 8th, 9th, 10th samples are 011,101,110), the frame is discarded and the receiving sequence is not started for this frame. There is no RDRF bit set for this frame and the NF flag is set internally (not accessible to the user). This NF flag is accessible along with the RDRF bit when a next valid frame is received.

Note: If the application Start Bit is not long enough to match the above requirements, then the NF Flag may get set due to the short Start Bit. In this case, the NF flag may be ignored by the application software when the first valid byte is received.

See also Section 0.1.4.10.



SERIAL COMMUNICATION INTERFACE (Cont'd)

Table 24. SCI Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0050h	SCISR	TDRE	TC	RDRF	IDLE	OVR	NF	FE	PE
005011	Reset Value	1	1	0	0	0	0	0	0
0051h	SCIDR	MSB							LSB
003111	Reset Value	х	х	х	х	х	х	х	х
0052h	SCIBRR	SCP1	SCP0	SCT2	SCT1	SCT0	SCR2	SCR1	SCR0
005211	Reset Value	0	0	0	0	0	0	0	0
0052h	SCICR1	R8	T8	SCID	М	WAKE	PCE	PS	PIE
005511	Reset Value	х	0	0	0	0	0	0	0
0054h	SCICR2	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
003411	Reset Value	0	0	0	0	0	0	0	0
0055b	SCIERPR	MSB							LSB
005511	Reset Value	0	0	0	0	0	0	0	0
0057h	SCIPETPR	MSB							LSB
005711	Reset Value	0	0	0	0	0	0	0	0



Acknowledge may be enabled and disabled by software.

The I²C interface address and/or general call address can be selected by software.

The speed of the I^2C interface may be selected between Standard (up to 100KHz) and Fast I^2C (up to 400KHz).

SDA/SCL Line Control

Transmitter mode: the interface holds the clock line low before transmission to wait for the microcontroller to write the byte in the Data Register.

Receiver mode: the interface holds the clock line low after reception to wait for the microcontroller to read the byte in the Data Register. The SCL frequency ($\rm F_{scl}$) is controlled by a programmable clock divider which depends on the $\rm l^2C$ bus mode.

When the I^2C cell is enabled, the SDA and SCL ports must be configured as floating inputs. In this case, the value of the external pull-up resistor used depends on the application.

When the I²C cell is disabled, the SDA and SCL ports revert to being standard I/O port pins.

لرکا





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.

10.7.4.2 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 69 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 69 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 69 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 69 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.



10.7.5 Low Power Modes

Mode	Description
WAIT	No effect on I ² C interface. I ² C interrupts cause the device to exit from WAIT mode.
HALT	I^2C registers are frozen. In HALT mode, the I^2C interface is inactive and does not acknowledge data on the bus. The I^2C interface resumes operation when the MCU is woken up by an interrupt with "exit from HALT mode" capability.

10.7.6 Interrupts

57

Figure 70. Event Flags and Interrupt Generation



Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
10-bit Address Sent Event (Master mode)	ADD10		Yes	No
End of Byte Transfer Event	BTF		Yes	No
Address Matched Event (Slave mode)	ADSEL		Yes	No
Start Bit Generation Event (Master mode)	SB		Yes	No
Acknowledge Failure Event	AF		Yes	No
Stop Detection Event (Slave mode)	STOPF		Yes	No
Arbitration Lost Event (Multimaster configuration)	ARLO		Yes	No
Bus Error Event	BERR		Yes	No

Note: The I^2C interrupt events are connected to the same interrupt vector (see Interrupts chapter). They generate an interrupt if the corresponding Enable Control Bit is set and the I-bit in the CC register is reset (RIM instruction).

125/197

Bit 1 = **M/SL** *Master/Slave*.

This bit is set by hardware as soon as the interface is in Master mode (writing START=1). It is cleared by hardware after detecting a Stop condition on the bus or a loss of arbitration (ARLO=1). It is also cleared when the interface is disabled (PE=0). 0: Slave mode

1: Master mode

Bit 0 = SB Start bit (Master mode).

This bit is set by hardware as soon as the Start condition is generated (following a write START=1). An interrupt is generated if ITE=1. It is cleared by software reading SR1 register followed by writing the address byte in DR register. It is also cleared by hardware when the interface is disabled (PE=0).

0: No Start condition

1: Start condition generated

I²C STATUS REGISTER 2 (SR2)

Read Only

Reset Value: 0000 0000 (00h)

7							0
0	0	0	AF	STOPF	ARLO	BERR	GCAL

Bit 7:5 = Reserved. Forced to 0 by hardware.

Bit 4 = **AF** Acknowledge failure.

This bit is set by hardware when no acknowledge is returned. An interrupt is generated if ITE=1. It is cleared by software reading SR2 register or by hardware when the interface is disabled (PE=0).

The SCL line is not held low while AF=1 but by other flags (SB or BTF) that are set at the same time.

0: No acknowledge failure

1: Acknowledge failure

Note:

– When an AF event occurs, the SCL line is not held low; however, the SDA line can remain low if the last bits transmitted are all 0. It is then necessary to release both lines by software.

Bit 3 =**STOPF** *Stop detection (Slave mode).* This bit is set by hardware when a Stop condition is detected on the bus after an acknowledge (if ACK=1). An interrupt is generated if ITE=1. It is cleared by software reading SR2 register or by hardware when the interface is disabled (PE=0). The SCL line is not held low while STOPF=1.

0: No Stop condition detected

1: Stop condition detected

Bit 2 = ARLO Arbitration lost.

This bit is set by hardware when the interface loses the arbitration of the bus to another master. An interrupt is generated if ITE=1. It is cleared by software reading SR2 register or by hardware when the interface is disabled (PE=0).

After an ARLO event the interface switches back automatically to Slave mode (M/SL=0).

The SCL line is not held low while ARLO=1.

0: No arbitration lost detected

1: Arbitration lost detected

Note:

- In a Multimaster environment, when the interface is configured in Master Receive mode it does not perform arbitration during the reception of the Acknowledge Bit. Mishandling of the ARLO bit from the I2CSR2 register may occur when a second master simultaneously requests the same data from the same slave and the I²C master does not acknowledge the data. The ARLO bit is then left at 0 instead of being set.

Bit 1 = **BERR** Bus error.

This bit is set by hardware when the interface detects a misplaced Start or Stop condition. An interrupt is generated if ITE=1. It is cleared by software reading SR2 register or by hardware when the interface is disabled (PE=0).

The SCL line is not held low while BERR=1.

0: No misplaced Start or Stop condition

1: Misplaced Start or Stop condition

Note:

 If a Bus Error occurs, a Stop or a repeated Start condition should be generated by the Master to re-synchronize communication, get the transmission acknowledged and the bus released for further communication

Bit 0 = GCAL General Call (Slave mode).

This bit is set by hardware when a general call address is detected on the bus while ENGC=1. It is cleared by hardware detecting a Stop condition (STOPF=1) or when the interface is disabled (PE=0).

- 0: No general call address detected on bus
- 1: general call address detected on bus

Table 25. I²C Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0018h	I2CCR Reset Value	0	0	PE 0	ENGC 0	START 0	ACK 0	STOP 0	ITE 0
0019h	I2CSR1 Reset Value	EVF 0	ADD10 0	TRA 0	BUSY 0	BTF 0	ADSL 0	M/SL 0	SB 0
001Ah	I2CSR2 Reset Value	0	0	0	AF 0	STOPF 0	ARLO 0	BERR 0	GCAL 0
001Bh	I2CCCR Reset Value	FM/SM 0	CC6 0	CC5 0	CC4 0	CC3 0	CC2 0	CC1 0	CC0 0
001Ch	I2COAR1 Reset Value	ADD7 0	ADD6 0	ADD5 0	ADD4 0	ADD3 0	ADD2 0	ADD1 0	ADD0 0
001Dh	I2COAR2 Reset Value	FR1 0	FR0 1	0	0	0	ADD9 0	ADD8 0	0
001Eh	I2CDR Reset Value	MSB 0	0	0	0	0	0	0	LSB 0

12.5 CLOCK AND TIMING CHARACTERISTICS

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

12.5.1 General Timings

Symbol	Parameter	Conditions	Min	Typ ¹⁾	Max	Unit
t _{c(INST)}	Instruction evolution		2	3	12	t _{CPU}
		f _{CPU} =8MHz	250	375	1500	ns
t _{v(IT)}	Interrupt reaction time ²⁾		10		22	t _{CPU}
	$t_{v(IT)} = \Delta t_{c(INST)} + 10$	f _{CPU} =8MHz	1.25		2.75	μs

12.5.2 External Clock Source

Symbol	Parameter	Conditions	Min	Тур	Max	Unit
V _{OSC1H}	OSC1 input pin high level voltage		$0.7 \mathrm{xV}_{\mathrm{DD}}$		V _{DD}	V
V _{OSC1L}	OSC1 input pin low level voltage	V _{SS}		$0.3 \mathrm{xV}_{\mathrm{DD}}$	v	
t _{w(OSC1H)} t _{w(OSC1L)}	OSC1 high or low time ³⁾	see Figure 75	5			ne
t _{r(OSC1)} t _{f(OSC1)}	OSC1 rise or fall time ³⁾				15	113
١L	OSC1 Input leakage current	V _{SS} ≤V _{IN} ≤V _{DD}			±1	μÂ

Figure 75. Typical Application with an External Clock Source



Notes:

1. Data based on typical application software.

2. Time measured between interrupt event and interrupt vector fetch. $\Delta t_{c(INST)}$ is the number of t_{CPU} cycles needed to finish the current instruction execution.

3. Data based on design simulation and/or technology characteristics, not tested in production.



CLOCK AND TIMING CHARACTERISTICS (Cont'd)

Supplier	f _{osc} (MHz)	Typical Ceramic Resonators ²⁾	
	2	CSTCC2M00G56Z-R0	
	Λ	SMD CSTCR4M00G53Z-R0	
b	4	4	Lead CSTLS4M00G53Z-R0
urat	8	0	SMD CSTCE8M00G52Z-R0
Ē		Lead CSTLS4M0052Z-R0	
	16	SMD CSTCE16M0V51Z-R0	
	10	Lead CSTLS16M0X51Z-R0	

Notes:

1. Resonator characteristics given by the ceramic resonator manufacturer.

2. SMD = [-R0: Plastic tape package (∅ =180mm), -B0: Bulk] LEAD = [-A0: Flat pack package (Radial taping Ho= 18mm), -B0: Bulk]

For more information on these resonators, please consult www.murata.com



EMC CHARACTERISTICS (Cont'd)

12.7.3 Absolute Maximum Ratings (Electrical Sensitivity)

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

12.7.3.1 Electro-Static Discharge (ESD)

Electro-Static Discharges (a positive then a negative pulse separated by 1 second) are applied to the pins of each sample according to each pin combination. The sample size depends on the number of supply pins in the device (3 parts*(n+1) supply pin). Two models can be simulated: Human Body Model and Machine Model. This test conforms to the JESD22-A114A/A115A standard.

Absolute Maximum Ratings

Symbol	Ratings	Conditions	Maximum value ¹⁾	Unit
V _{ESD(HBM)}	Electro-static discharge voltage (Human Body Model)	T _A =+25°C	2000	V
V _{ESD(MM)}	Electro-static discharge voltage (Machine Model)	T _A =+25°C	200	v

Notes:

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

12.7.3.2 Static Latch-Up

 LU: 2 complementary static tests are required on 6 parts to assess the latch-up performance. A supply overvoltage (applied to each power supply pin) and a current injection (applied to each input, output and configurable I/O pin) are performed on each sample. This test conforms to the EIA/JESD 78 IC latch-up standard.

Electrical Sensitivities

Symbol	Parameter	Conditions	Class ¹⁾
LU	Static latch-up class	T _A =+125°C conforming to JESD 78	II level A



12.8 I/O PORT PIN CHARACTERISTICS

12.8.1 General Characteristics

Subject to general operating conditions for $V_{\text{DD}},\,f_{\text{OSC}},$ and T_{A} unless otherwise specified.

Symbol	Parameter	Cond	litions	Min	Тур	Мах	Unit
V _{IL}	Input low level voltage 1)	CMOS ports				$0.3 \mathrm{xV}_{\mathrm{DD}}$	
V _{IH}	Input high level voltage 1)			$0.7 \mathrm{xV}_{\mathrm{DD}}$			V
V _{hys}	Schmitt trigger voltage hysteresis 2)				0.7		
I _{INJ(PIN)} ³⁾	Injected Current on PB0 (Flash de- vices only)	V _{DD} =5V		0		+4	
- ()	Injected Current on an I/O pin					± 4	mA
$\Sigma I_{\rm INJ(PIN)}^{3)}$	Total injected current (sum of all I/O and control pins)					± 25	
١ _L	Input leakage current	V _{SS} ≤V _{IN} ≤V _{DD}				±1	
۱ _S	Static current consumption	Floating input mode 4)			400		μΛ
R _{PU}	Weak pull-up equivalent resistor 5)	V _{IN} =V _{SS}	V _{DD} =5V	50	120	250	kΩ
C _{IO}	I/O pin capacitance				5		pF
t _{f(IO)out}	Output high to low level fall time 1)	C _L =50pF Between 10% and 90%			25		ne
t _{r(IO)out}	Output low to high level rise time ¹⁾				25		115
t _{w(IT)in}	External interrupt pulse time ⁶⁾			1			t _{CPU}

Figure 79. Unused I/Os configured as input



Figure 80. Typical I_{PU} vs. V_{DD} with V_{IN}=V_{SS}



Notes:

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

2. Hysteresis voltage between Schmitt trigger switching levels. Based on characterization results, not tested.

3. When the current limitation is not possible, the V_{IN} maximum must be respected, otherwise refer to I_{INJ(PIN)} specification. A positive injection is induced by V_{IN}>V_{DD} while a negative injection is induced by V_{IN}<V_{SS}. Refer to section 12.2.2 on page 143 for more details.

4. Configuration not recommended, all unused pins must be kept at a fixed voltage: using the output mode of the I/O for example and leaving the I/O unconnected on the board or an external pull-up or pull-down resistor (see Figure 79). Static peak current value taken at a fixed V_{IN} value, based on design simulation and technology characteristics, not tested in production. This value depends on V_{DD} and temperature values.

5. The R_{PU} pull-up equivalent resistor is based on a resistive transistor (corresponding I_{PU} current characteristics described in Figure 80).

6. To generate an external interrupt, a minimum pulse width has to be applied on an I/O port pin configured as an external interrupt source.

13.2 THERMAL CHARACTERISTICS

Symbol	Ratings	Value	Unit
	Package thermal resistance (junction to ambient)		
R _{thJA}	LQFP64 10x10	50	
	LQFP48 7x7	80	
	LQFP44 10x10	52	°C/W
	SDIP42	55	
	LQFP32 7x7	70	
	SDIP32	50	
PD	Power dissipation ¹⁾	500	mW
T _{Jmax}	Maximum junction temperature ²⁾	150	°C

Notes:

1. The maximum chip-junction temperature is based on technology characteristics.

2. The maximum power dissipation is obtained from the formula PD = (TJ - TA) / RthJA.

The power dissipation of an application can be defined by the user with the formula: PD=PINT+PPORT where PINT is the chip internal power (IDDxVDD) and PPORT is the port power dissipation depending on the ports used in the application.



ST72325 DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)

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 4.3.1 and the ST7 Flash Programming Reference Manual for more details. 0: Read-out protection enabled

1: Read-out protection disabled

OPTION BYTE 1

OPT7= **PKG1** *Package selection bit 1* This option bit selects the package.

Version	Selected Package	Flash size	PKG 1
R/AR	LQFP64	32/48/60K	1
С	LQFP48(C)	32/48/60K	1
J	LQFP44/SDIP42	48/60K	0
S/J	LQFP48(S)/LQFP44/ SDIP42	16/32K	1
К	LQFP32/SDIP32	16/32K	0

Note: On the chip, each I/O port has up to 8 pads. Pads that are not bonded to external pins are forced in input pull-up configuration after reset. The configuration of these pads must be kept at reset state to avoid added current consumption.

In LQFP48(C) devices (PA0, PA1, PB6, PB7, PD6, PD7, PE3, PE5, PE6, PE7, PF3, PF5) are in input floating configuration after reset. Refer to Note 4 on page 16.

OPT6 = **RSTC** *RESET* clock cycle selection This option bit selects the number of CPU cycles applied during the RESET phase and when exiting HALT mode. For resonator oscillators, it is advised to select 4096 due to the long crystal stabilization time.

0: Reset phase with 4096 CPU cycles

1: Reset phase with 256 CPU cycles

OPT5:4 = OSCTYPE[1:0] Oscillator Type

These option bits select the ST7 main clock source type.

OPT3:1 = **OSCRANGE[2:0]** Oscillator range When the resonator oscillator type is selected, these option bits select the resonator oscillator

Clock Source	OSCTYPE		
CIOCK Source	1	0	
Resonator Oscillator	0	0	
Reserved	0	1	
Internal RC Oscillator	1	0	
External Source	1	1	

current source corresponding to the frequency range of the used resonator. Otherwise, these bits are used to select the normal operating frequency range.

Tun Frog Pongo	OSCRANGE			
Typ. Fleq. hallye	2	1	0	
1~2MHz	0	0	0	
2~4MHz	0	0	1	
4~8MHz	0	1	0	
8~16MHz	0	1	1	

OPT0 = PLLOFF PLL activation

This option bit activates the PLL which allows multiplication by two of the main input clock frequency. The PLL is guaranteed only with an input frequency between 2 and 4MHz, for this reason the PLL must not be used with the internal RC oscillator. 0: PLL x2 enabled

1: PLL x2 disabled

CAUTION: the PLL can be enabled only if the "OSC RANGE" (OPT3:1) bits are configured to " 2~4MHz". Otherwise, the device functionality is not guaranteed.



T72325 DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)					
ST72325 ROM MICROCONTROLLER OPTION LIST (Last update: October 2008)					
Customer:	Customer: Address:				
Contact: Phone No: Reference/ROM Code: The ROM code name is as ROM code must be sent ir	ssigned by STMicroelectro	onics. sion cannot be processed.			
Device Type/Memory Size	/Package (check only one	e option):			
ROM device:	32K	16K			
LQFP32 7x7: SDIP32: LQFP48 7x7: LQFP44 10x10: SDIP42:	[] [] [] []	[] [] [] []			
Die form:	32K	16K			
 	[]	[]			
Conditioning (check only c	ne option):				
Packag	ed Product	Die Product (dice tested at 25°C only)			
[] Tape & Reel []	Tray	I [] Tape & Reel I [] Inked wafer I [] Sawn wafer on sticky foil			
Temp. Range (do not cheo Temp. Range [] -40°C to +85°C [] -40°C to +125°C	:k for die product). 				
Special Marking: Authorized characters are	[] No letters, digits, '.', '-', '/' and	[] Yes "" (10 char. max) d spaces only.			
Clock Source Selection:	[] Resonator: [] Internal RC	[] 1 to 2 MHz [] 2 to 4 MHz [] 4 to 8 MHz [] 8 to 16 MHz			
PLL:	[] External Clock [] Disabled	[] Enabled			
CSS:	[] Disabled	[] Enabled			
LVD Reset: [] Disabled Reset Delay: Watchdog Selection: Watchdog Reset on Halt: Readout Protection:	[] High threshold []256 Cycles [] Software Activa [] Reset [] Disabled	[] Med. threshold [] Low threshold [] 4096 Cycles tion [] Hardware Activation [] No Reset [] Enabled			
Date		Signature			
Please download the latest version of this option list from: http://www.st.com/mcu > downloads > ST7 microcontrollers > Option list					

Table 32. ST7 Application Notes

IDENTIFICATION	DESCRIPTION
AN1071	HALF DUPLEX USB-TO-SERIAL BRIDGE USING THE ST72611 USB MICROCONTROLLER
AN1106	TRANSLATING ASSEMBLY CODE FROM HC05 TO ST7
AN1179	PROGRAMMING ST7 FLASH MICROCONTROLLERS IN REMOTE ISP MODE (IN-SITU PRO- GRAMMING)
AN1446	USING THE ST72521 EMULATOR TO DEBUG AN ST72324 TARGET APPLICATION
AN1477	EMULATED DATA EEPROM WITH XFLASH MEMORY
AN1527	DEVELOPING A USB SMARTCARD READER WITH ST7SCR
AN1575	ON-BOARD PROGRAMMING METHODS FOR XFLASH AND HDFLASH ST7 MCUS
AN1576	IN-APPLICATION PROGRAMMING (IAP) DRIVERS FOR ST7 HDFLASH OR XFLASH MCUS
AN1577	DEVICE FIRMWARE UPGRADE (DFU) IMPLEMENTATION FOR ST7 USB APPLICATIONS
AN1601	SOFTWARE IMPLEMENTATION FOR ST7DALI-EVAL
AN1603	USING THE ST7 USB DEVICE FIRMWARE UPGRADE DEVELOPMENT KIT (DFU-DK)
AN1635	ST7 CUSTOMER ROM CODE RELEASE INFORMATION
AN1754	DATA LOGGING PROGRAM FOR TESTING ST7 APPLICATIONS VIA ICC
AN1796	FIELD UPDATES FOR FLASH BASED ST7 APPLICATIONS USING A PC COMM PORT
AN1900	HARDWARE IMPLEMENTATION FOR ST7DALI-EVAL
AN1904	ST7MC THREE-PHASE AC INDUCTION MOTOR CONTROL SOFTWARE LIBRARY
AN1905	ST7MC THREE-PHASE BLDC MOTOR CONTROL SOFTWARE LIBRARY
SYSTEM OPTIMIZ	ATION
AN1711	SOFTWARE TECHNIQUES FOR COMPENSATING ST7 ADC ERRORS
AN1827	IMPLEMENTATION OF SIGMA-DELTA ADC WITH ST7FLITE05/09
AN2009	PWM MANAGEMENT FOR 3-PHASE BLDC MOTOR DRIVES USING THE ST7FMC
AN2030	BACK EMF DETECTION DURING PWM ON TIME BY ST7MC