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	16MHz
Connectivity	I²C, 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-SO
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f264g2m6

Pin n°			Pin Name	Type	Level		Port / Control						Main Function (after reset)	Alternate Function
							Input				Output			
SDIP32	SO28	BGA			Input	Output	float	wpu	int	ana	OD	PP		
15	13	E2	PC4/OCMP2_B/AIN4	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C4**	Timer B Output Compare 2 or ADC Analog Input 4
16	14	F3	PC3/ ICAP2_B/AIN3	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C3**	Timer B Input Capture 2 or ADC Analog Input 3
17	15	E3	PC2/MCO/AIN2	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C2**	Main clock output (f_{CPU}) or ADC Analog Input 2
18	16	F4	PC1/OCMP1_B/AIN1	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C1**	Timer B Output Compare 1 or ADC Analog Input 1
19	17	D3	PC0/ICAP1_B/AIN0	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C0**	Timer B Input Capture 1 or ADC Analog Input 0
20	18	E4	PA7/TDO	I/O	C_T	HS	**X**	ei0			X	X	**Port A7**	SCI output
21	19	F5	PA6/SDAI	I/O	C_T	HS	**X**		ei0		T		**Port A6**	I^2C DATA
22	20	F6	PA5 /RDI	I/O	C_T	HS	**X**	ei0			X	X	**Port A5**	SCI input
23	21	E6	PA4/SCLI	I/O	C_T	HS	**X**		ei0		T		**Port A4**	I^2C CLOCK
24		E5	NC	Not Connected										
25		D6	NC											
		D5	NC											
26	22	C6	PA3	I/O	C_T	HS	**X**	ei0			X	X	**Port A3**	
27	23	D4	PA2	I/O	C_T	HS	**X**	ei0			X	X	**Port A2**	
		C5	NC	Not Connected										
		B6	NC											
28	24	A6	PA1/ICCDATA	I/O	C_T	HS	**X**	ei0			X	X	**Port A1**	In Circuit Communication Data
29	25	A5	PA0/ICCCLK	I/O	C_T	HS	**X**	ei0			X	X	**Port A0**	In Circuit Communication Clock
30	26	B5	ICCSEL	I	C_T		**X**						ICC mode pin, must be tied low	
31	27	A4	V_{SS}	S									Ground	
32	28	B4	V_{DD}	S									Main power supply	

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 a pull-up interrupt input, otherwise the configuration is a floating interrupt input. Port C is mapped to ei0 or ei1 by option byte.

2. In the open drain output column, “T” defines a true open drain I/O (P-Buffer and protection diode to V_{DD} are not implemented). See Section 9 "I/O PORTS" on page 38 for more details.

3. OSC1 and OSC2 pins connect a crystal or ceramic resonator, or an external source to the on-chip oscillator see Section 2 "PIN DESCRIPTION" on page 6 and Section 6.2 "MULTI-OSCILLATOR (MO)" on page 21 for more details.

4: For details refer to Section 13.8 on page 144

FLASH PROGRAM MEMORY (Cont'd)

4.4 ICC interface

ICP needs a minimum of 4 and up to 7 pins to be connected to the programming tool. These pins are:

- $\overline{\text{RESET}}$: device reset
- V_{SS}: device power supply ground
- ICCCLK: ICC output serial clock pin
- ICCDATA: ICC input serial data pin
- ICCSEL: ICC selection (not required on devices without ICCSEL pin)
- OSC1: main clock input for external source (not required on devices without OSC1/OSC2 pins)
- V_{DD}: application board power supply (optional, see Note 3)

Notes:

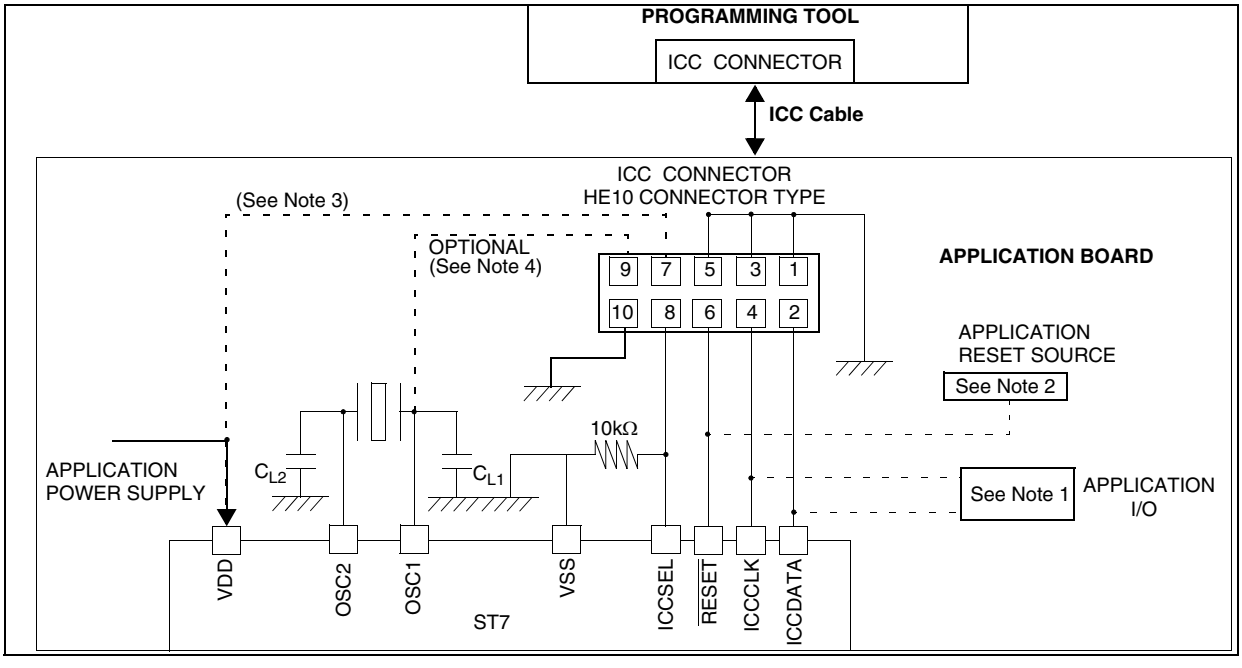
1. If the ICCCLK or ICCDATA pins are only used as outputs in the application, no signal isolation is necessary. As soon as the Programming Tool is plugged to the board, even if an ICC session is not in progress, the ICCCLK and ICCDATA pins are not available for the application. If they are used as inputs by the application, isolation such as a serial resistor has to be implemented in case another device forces the signal. Refer to the Programming Tool documentation for recommended resistor values.

2. During the ICP session, the programming tool must control the $\overline{\text{RESET}}$ pin. This can lead to conflicts between the programming tool and the application reset circuit if it drives more than 5mA at high level (push pull output or pull-up resistor<1K). A schottky diode can be used to isolate the application RESET circuit in this case. When using a classical RC network with R>1K or a reset management IC with open drain output and pull-up resistor>1K, no additional components are needed. In all cases the user must ensure that no external reset is generated by the application during the ICC session.

3. The use of Pin 7 of the ICC connector depends on the Programming Tool architecture. This pin must be connected when using most ST Programming Tools (it is used to monitor the application power supply). Please refer to the Programming Tool manual.

4. Pin 9 has to be connected to the OSC1 pin of the ST7 when the clock is not available in the application or if the selected clock option is not programmed in the option byte. ST7 devices with multi-oscillator capability need to have OSC2 grounded in this case.

Figure 6. Typical ICC Interface



6 SUPPLY, RESET AND CLOCK MANAGEMENT

The device includes a range of utility features for securing the application in critical situations (for example in case of a power brown-out), and reducing the number of external components. An overview is shown in Figure 10.

For more details, refer to dedicated parametric section.

Main Features

- Optional PLL for multiplying the frequency by 2 (not to be used with internal RC oscillator)
- Reset Sequence Manager (RSM)
- Multi-Oscillator Clock Management (MO)
  - 4 Crystal/Ceramic resonator oscillators
  - 1 Internal RC oscillator
- System Integrity Management (SI)
  - Main supply Low Voltage Detector (LVD)
  - Auxiliary Voltage Detector (AVD) with interrupt capability for monitoring the main supply

6.1 PHASE LOCKED LOOP

If the clock frequency input to the PLL is in the 2 to 4 MHz range, the PLL can be used to multiply the frequency by two to obtain an f_{OSC2} of 4 to 8 MHz.

The PLL is enabled by option byte. If the PLL is disabled, then $f_{OSC2} = f_{OSC}/2$.

Caution: The PLL is not recommended for applications where timing accuracy is required. See "PLL Characteristics" on page 139.

Figure 9. PLL Block Diagram

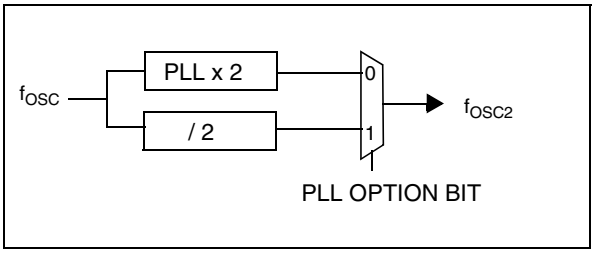
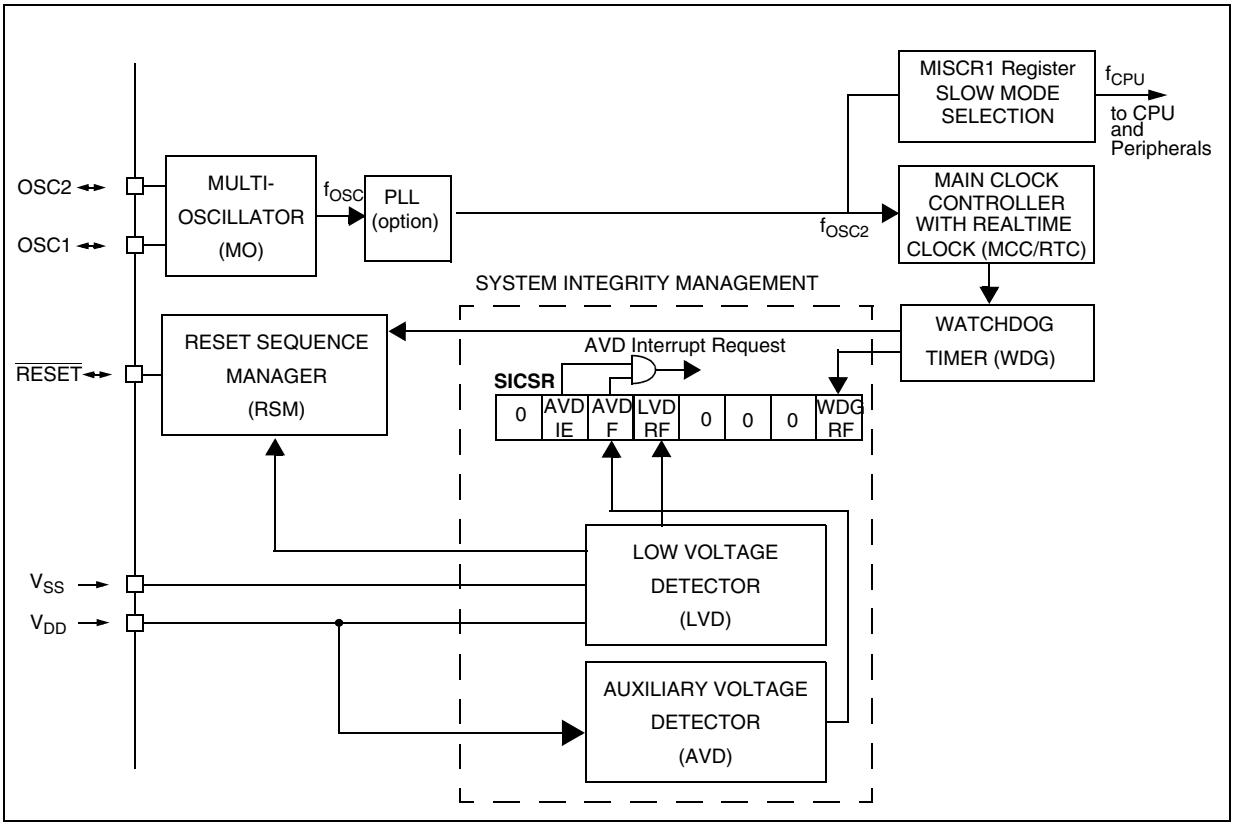


Figure 10. Clock, Reset and Supply Block Diagram



INTERRUPTS (Cont'd)

Table 5. Interrupt Mapping

N°	Source Block	Description	Register Label	Priority Order	Exit from HALT	Address Vector
	RESET	Reset	N/A	Highest Priority	yes	FFFEh-FFFFh
	TRAP	Software Interrupt			no	FFFCh-FFFDh
0	ei0	External Interrupt Port A7..0 (C5..0[1])			yes	FFFAh-FFFBh
1	ei1	External Interrupt Port B7..0 (C5..0[1])				FFF8h-FFF9h
2		Not used				FFF6h-FFF7h
3	SPI	SPI Peripheral Interrupts	SPISR		yes	FFF4h-FFF5h
4	TIMER A	TIMER A Peripheral Interrupts	TASR		no	FFF2h-FFF3h
5	MCC	Time base interrupt	MCCSR		yes	FFF0h-FFF1h
6	TIMER B	TIMER B Peripheral Interrupts	TBSR		no	FFEEh-FFEFh
7	AVD	Auxiliary Voltage Detector interrupt	SICSR			FFECh-FFEDh
8		Not used				FFEAh-FFEBh
9		Not used				FFE8h-FFE9h
10	SCI	SCI Peripheral Interrupt	SCISR		no	FFE6h-FFE7h
11	I^2C	I^2C Peripheral Interrupt	I2CSRx		no	FFE4h-FFE5h
12		Not Used		Lowest Priority		FFE2h-FFE3h
13		Not Used				FFE0h-FFE1h

Note 1. Configurable by option byte.

Table 6. Nested Interrupts Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
001Ch		SPI		Not Used		EI1		EI0	
	ISPR0 Reset Value	I1_3 1	I0_3 1	I1_2 1	I0_2 1	I1_1 1	I0_1 1	I1_0 1	I0_0 1
001Dh		AVD		TIMERB		MCC		TIMERA	
	ISPR1 Reset Value	I1_7 1	I0_7 1	I1_6 1	I0_6 1	I1_5 1	I0_5 1	I1_4 1	I0_4 1
001Eh		I^2C		SCI		Not Used		Not Used	
	ISPR2 Reset Value	I1_11 1	I0_11 1	I1_10 1	I0_10 1	I1_9 1	I0_9 1	I1_8 1	I0_8 1
001Fh						Not Used		Not Used	
	ISPR3 Reset Value	1	1	1	1	I1_13 1	I0_13 1	I1_12 1	I0_12 1

MAIN CLOCK CONTROLLER WITH REAL TIME CLOCK (Cont'd)

11.2.2 Low Power Modes

Mode	Description
WAIT	No effect on MCC/RTC peripheral. MCC/RTC interrupt cause the device to exit from WAIT mode.
ACTIVE-HALT	No effect on MCC/RTC counter (OIE bit is set), the registers are frozen. MCC/RTC interrupt cause the device to exit from ACTIVE-HALT mode.
HALT	MCC/RTC counter and registers are frozen. MCC/RTC operation resumes when the MCU is woken up by an interrupt with "exit from HALT" capability.

11.2.3 Interrupts

The MCC/RTC interrupt event generates an interrupt if the OIE bit of the MCCSR register is set and the interrupt mask in the CC register is not active (RIM instruction).

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
Time base overflow event	OIF	OIE	Yes	No [1]

Note:

The MCC/RTC interrupt wakes up the MCU from ACTIVE-HALT mode, not from HALT mode.

11.2.4 Register Description

MCC CONTROL/STATUS REGISTER (MCCSR)

Read/Write

Reset Value: 0000 0000 (00h)

7							0
0	0	0	0	TB1	TB0	OIE	OIF

Bit 7:4 = **reserved**

Bit 3:2 = **TB[1:0]** *Time base control*

These bits select the programmable divider time base. They are set and cleared by software.

Counter Prescaler	Time Base		TB1	TB0
	f_{OSC2} =4MHz	f_{OSC2}=8MHz		
16000	4ms	2ms	0	0
32000	8ms	4ms	0	1
80000	20ms	10ms	1	0
200000	50ms	25ms	1	1

A modification of the time base is taken into account at the end of the current period (previously set) to avoid an unwanted time shift. This allows to use this time base as a real time clock.

Bit 1 = **OIE** *Oscillator interrupt enable*
This bit set and cleared by software.
0: Oscillator interrupt disabled
1: Oscillator interrupt enabled
This interrupt can be used to exit from ACTIVE-HALT mode.
When this bit is set, calling the ST7 software HALT instruction enters the ACTIVE-HALT power saving mode.

Bit 0 = **OIF** *Oscillator interrupt flag*
This bit is set by hardware and cleared by software reading the CSR register. It indicates when set that the main oscillator has reached the selected elapsed time (TB1:0).
0: Timeout not reached
1: Timeout reached

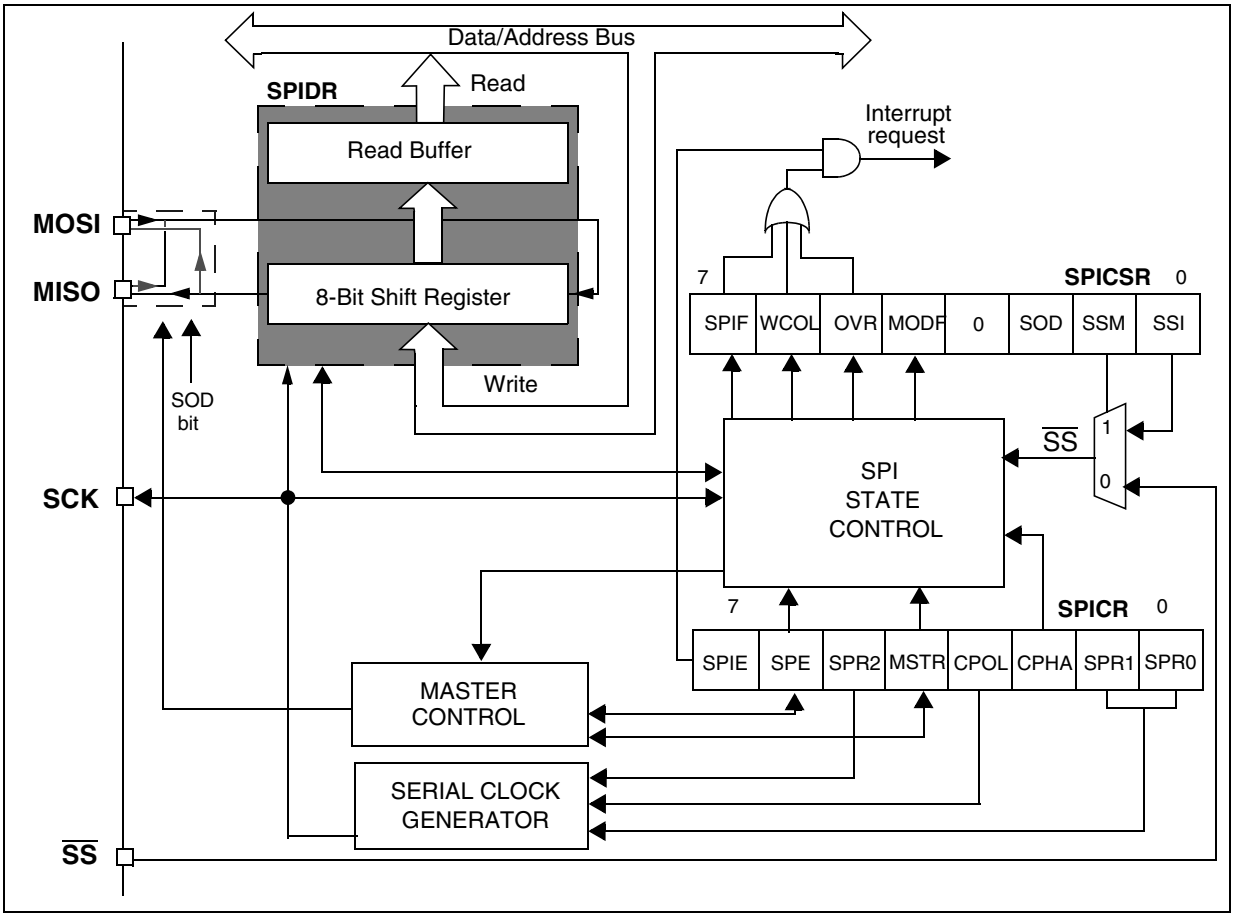
CAUTION: The BRES and BSET instructions must not be used on the MCCSR register to avoid unintentionally clearing the OIF bit.

Table 13. Main Clock Controller Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0025h	**SICSR** Reset Value	 0	AVDIE 0	AVDF 0	LVDRF x	 0	 0	 0	WDGRF x
0026h	**MCCSR** Reset Value	 0	 0	 0	 0	TB1 0	TB0 0	OIE 0	OIF 0

SERIAL PERIPHERAL INTERFACE (Cont'd)

Figure 46. Serial Peripheral Interface Block Diagram



SERIAL PERIPHERAL INTERFACE (Cont'd)

11.4.3.1 Functional Description

A basic example of interconnections between a single master and a single slave is illustrated in Figure 47.

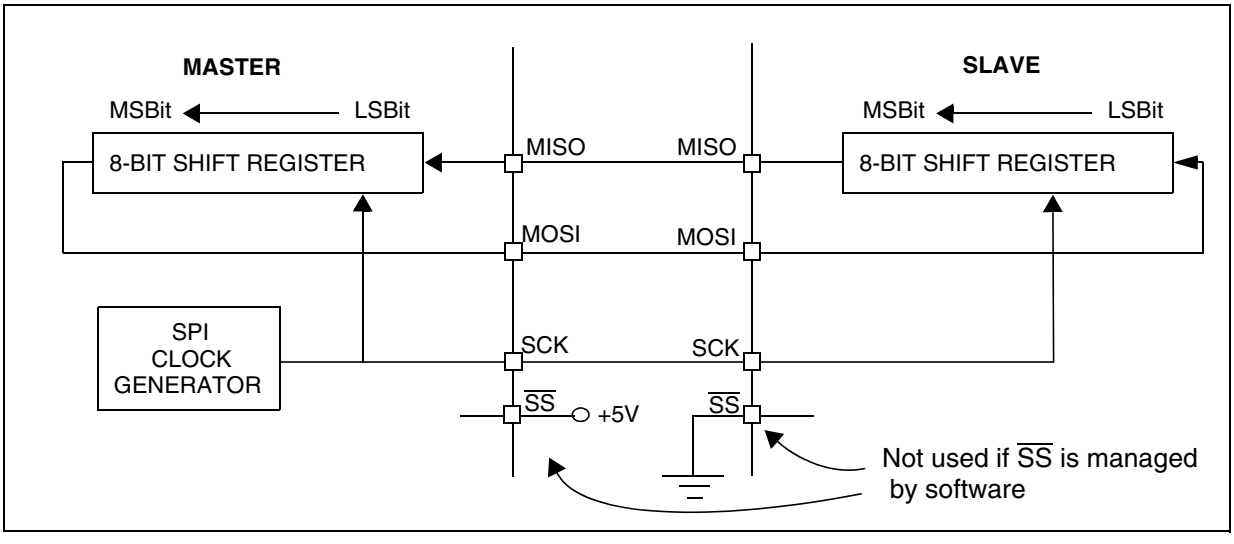
The MOSI pins are connected together and the MISO pins are connected together. In this way data is transferred serially between master and slave (most significant bit first).

The communication is always initiated by the master. When the master device transmits data to a slave device via MOSI pin, the slave device responds by sending data to the master device via the MISO pin. This implies full duplex communication with both data out and data in synchronized with the same clock signal (which is provided by the master device via the SCK pin).

To use a single data line, the MISO and MOSI pins must be connected at each node (in this case only simplex communication is possible).

Four possible data/clock timing relationships may be chosen (see Figure 50) but master and slave must be programmed with the same timing mode.

Figure 47. Single Master/ Single Slave Application



SERIAL PERIPHERAL INTERFACE (Cont'd)

11.4.6 Low Power Modes

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 operation resumes when the Device is woken up by an interrupt with "exit from HALT mode" capability. The data received is subsequently read from the SPIDR register when the software 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.

11.4.6.1 Using the SPI to wake-up the Device from Halt mode

In slave configuration, the SPI is able to wake-up the 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 Device from Halt mode only if the Slave Select signal (external $\overline{SS}$ pin or the SSI bit in the SPICSR register) is low when the Device enters Halt mode. So if Slave selection is configured as external (see Section 11.4.3.2), make sure the master drives a low level on the $\overline{SS}$ pin when the slave enters Halt mode.

11.4.7 Interrupts

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
SPI End of Transfer Event	SPIF	SPIE	Yes	Yes
Master Mode Fault Event	MODF		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 the CC register is reset (RIM instruction).

SERIAL COMMUNICATIONS INTERFACE (Cont'd)

11.5.4 Functional Description

The block diagram of the Serial Control Interface, is shown in Figure 53. It contains 6 dedicated registers:

– Two control registers (SCICR1 & SCICR2)

– A status register (SCISR)

– A baud rate register (SCIBRR)

– An extended prescaler receiver register (SCIERPR)

– An extended prescaler transmitter register (SCIETPR)

Refer to the register descriptions in Section 11.5.7for the definitions of each bit.

11.5.4.1 Serial Data Format

Word length may be selected as being either 8 or 9 bits by programming the M bit in the SCICR1 register (see Figure 53).

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

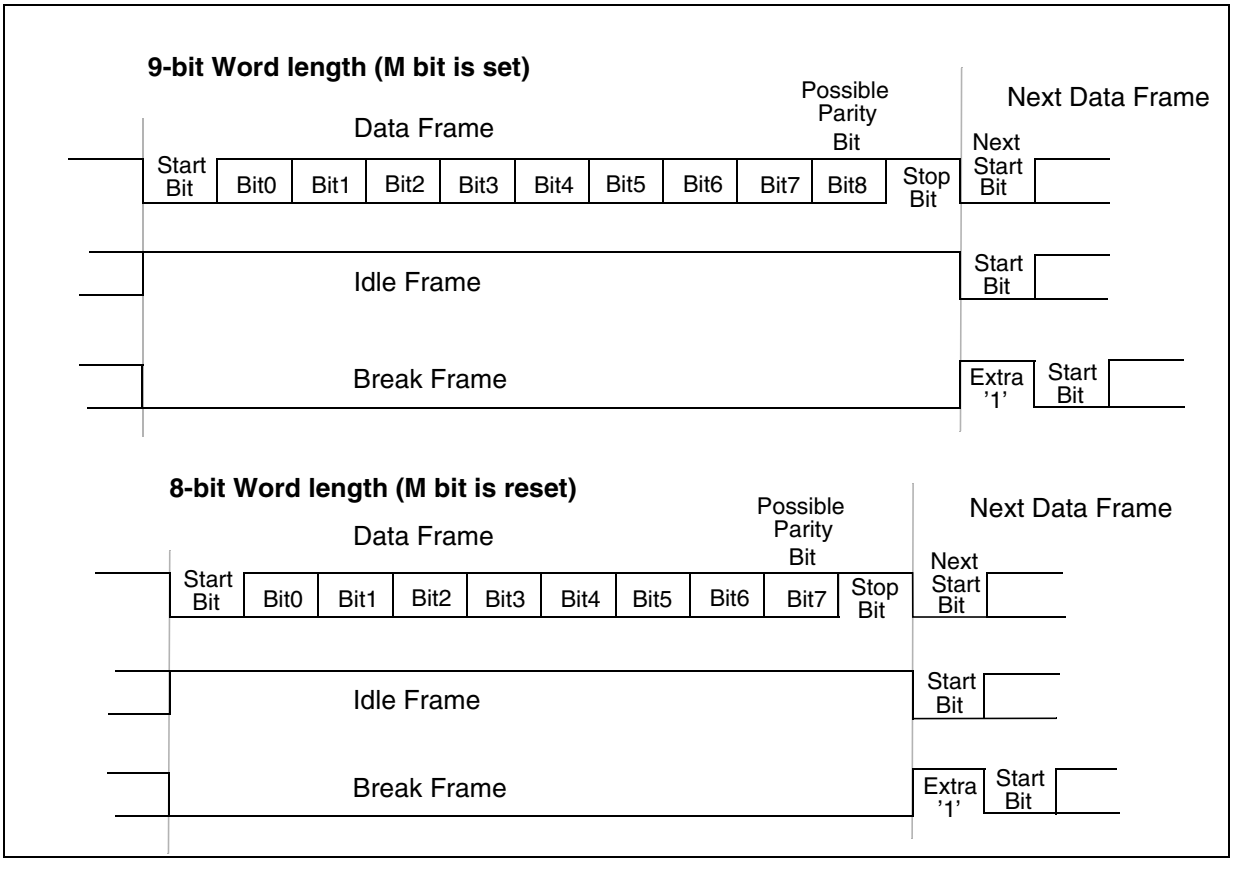
The TDO pin is in high state during the stop bit.

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

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

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

Figure 54. Word Length Programming



INSTRUCTION SET OVERVIEW (Cont'd)

Mnemo	Description	Function/Example	Dst	Src	I1	H	I0	N	Z	C
ADC	Add with Carry	A = A + M + C	A	M		H		N	Z	C
ADD	Addition	A = A + M	A	M		H		N	Z	C
AND	Logical And	A = A . M	A	M				N	Z	
BCP	Bit compare A, Memory	tst (A . M)	A	M				N	Z	
BRES	Bit Reset	bres Byte, #3	M							
BSET	Bit Set	bset Byte, #3	M							
BTJF	Jump if bit is false (0)	btjf Byte, #3, Jmp1	M							C
BTJT	Jump if bit is true (1)	btjt Byte, #3, Jmp1	M							C
CALL	Call subroutine									
CALLR	Call subroutine relative									
CLR	Clear		reg, M					0	1	
CP	Arithmetic Compare	tst(Reg - M)	reg	M				N	Z	C
CPL	One Complement	A = FFH-A	reg, M					N	Z	1
DEC	Decrement	dec Y	reg, M					N	Z	
HALT	Halt				1		0			
IRET	Interrupt routine return	Pop CC, A, X, PC			I1	H	I0	N	Z	C
INC	Increment	inc X	reg, M					N	Z	
JP	Absolute Jump	jp [TBL.w]								
JRA	Jump relative always									
JRT	Jump relative									
JRF	Never jump	jrf *								
JRIH	Jump if ext. INT pin = 1	(ext. INT pin high)								
JRIL	Jump if ext. INT pin = 0	(ext. INT pin low)								
JRH	Jump if H = 1	H = 1 ?								
JRNH	Jump if H = 0	H = 0 ?								
JRM	Jump if I1:0 = 11	I1:0 = 11 ?								
JRNM	Jump if I1:0 <> 11	I1:0 <> 11 ?								
JRMI	Jump if N = 1 (minus)	N = 1 ?								
JRPL	Jump if N = 0 (plus)	N = 0 ?								
JREQ	Jump if Z = 1 (equal)	Z = 1 ?								
JRNE	Jump if Z = 0 (not equal)	Z = 0 ?								
JRC	Jump if C = 1	C = 1 ?								
JRNC	Jump if C = 0	C = 0 ?								
JRULT	Jump if C = 1	Unsigned <								
JRUGE	Jump if C = 0	Jmp if unsigned >=								
JRUGT	Jump if (C + Z = 0)	Unsigned >								

CLOCK CHARACTERISTICS (Cont'd)

13.5.4 RC Oscillators

The ST7 internal clock can be supplied with an internal RC oscillator.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$f_{OSC\ (RCINT)}$	Internal RC oscillator frequency See Figure 73	T_A=25°C, V_{DD}=5V	2	3.5	6	MHz

Figure 72. Typical Application with RC oscillator

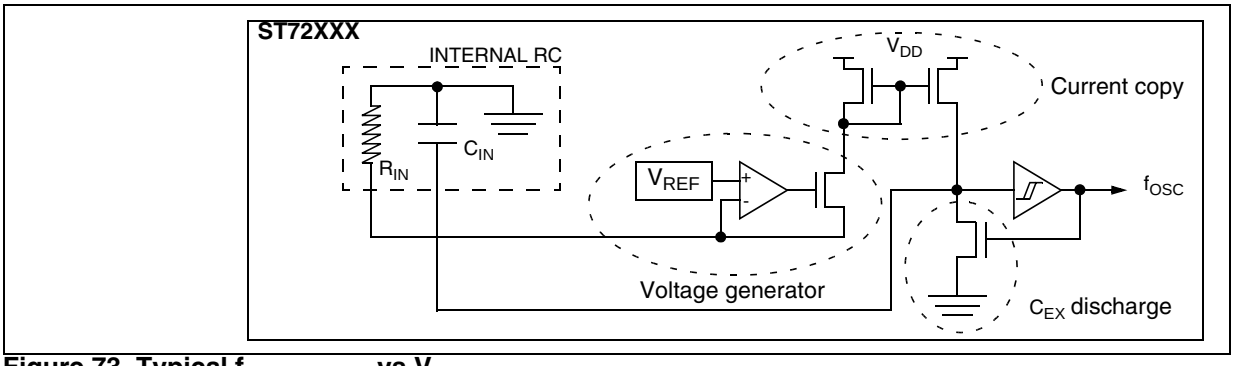
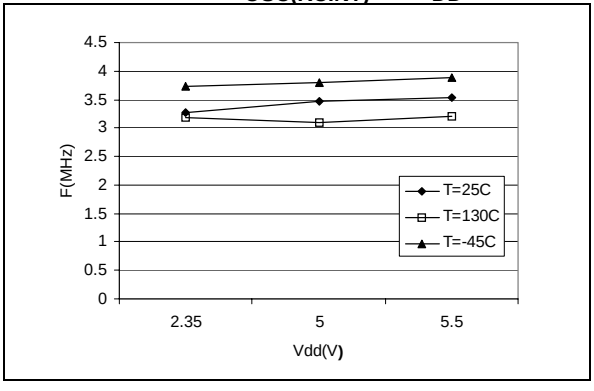


Figure 73. Typical $f_{OSC(RCINT)}$ vs V_{DD}



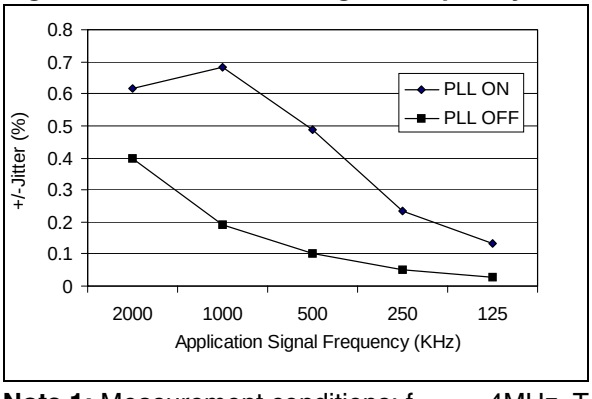
CLOCK CHARACTERISTICS (Cont'd)

13.5.5 PLL Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{DD(PLL)}$	PLL Operating Range	T_A 0 to 70°C	3.5		5.5	V
		T_A -40 to +85°C	4.5		5.5	
f_{OSC}	PLL input frequency range		2		4	MHz
$\Delta f_{CPU}/f_{CPU}$	Instantaneous PLL jitter 1)	f_{OSC} = 4 MHz.		1.0	2.5	%
		f_{OSC} = 2 MHz.		2.5	4.0	%

Note:
1. Data characterized but not tested.

Figure 74. PLL Jitter vs. Signal frequency[1]



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°C

I/O PORT PIN CHARACTERISTICS (Cont'd)

13.8.2 Output Driving Current

T_A = -40 to +85°C unless otherwise specified

Symbol	Parameter	Conditions		Min	Max	Unit
V_{OL} 1)	Output low level voltage for a standard I/O pin when 8 pins are sunk at same time	V_{DD}=5V	I_{IO}=+5mA		1.2	V
			I_{IO}=+2mA		0.5	
	Output low level voltage for a high sink I/O pin when 4 pins are sunk at same time		I_{IO}=+20mA,		1.3	
			I_{IO}=+8mA		0.75	
V_{OH} 2)	Output high level voltage for an I/O pin when 4 pins are sourced at same time		I_{IO}=-5mA,	V_{DD}-1.6		
			I_{IO}=-2mA	V_{DD}-0.8		
V_{OL} 1)3)	Output low level voltage for a standard I/O pin when 8 pins are sunk at same time	V_{DD}=3.3V	I_{IO}=+2mA		0.6	
	Output low level voltage for a high sink I/O pin when 4 pins are sunk at same time		I_{IO}=+8mA		0.6	
V_{OH} 2)3)	Output high level voltage for an I/O pin when 4 pins are sourced at same time		I_{IO}=-2mA $T_A \leq 85°C$	V_{DD}-0.8		
V_{OL} 1)3)	Output low level voltage for a standard I/O pin when 8 pins are sunk at same time	V_{DD}=2.7V	I_{IO}=+2mA		0.7	
	Output low level voltage for a high sink I/O pin when 4 pins are sunk at same time		I_{IO}=+8mA		0.7	
V_{OH} 2)3)	Output high level voltage for an I/O pin when 4 pins are sourced at same time		I_{IO}=-2mA	V_{DD}-0.9		

Notes:

1. The I_{IO} current sunk must always respect the absolute maximum rating specified in Section 13.2.2 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 13.2.2 and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VDD}. True open drain I/O pins does not have V_{OH}.
3. Not tested in production, based on characterization results.

I/O PORT PIN CHARACTERISTICS (Cont'd)

Figure 80. Typ. V_{OL} at V_{DD}=5V (standard)

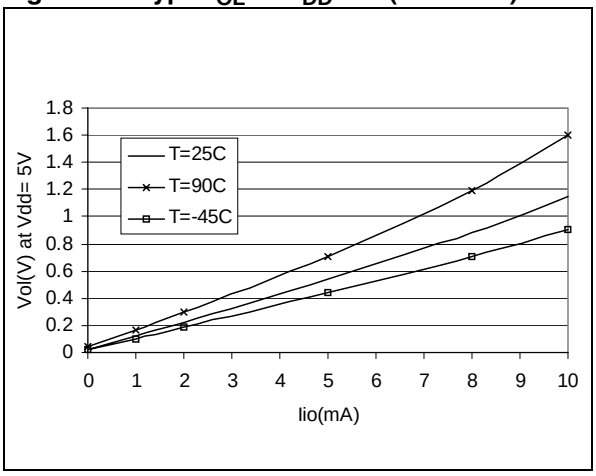


Figure 81. Typ. V_{OL} at V_{DD}=3V (high-sink)

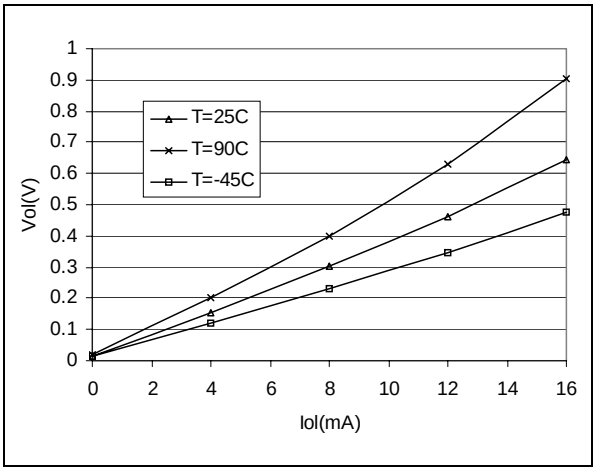


Figure 82. Typ. V_{OL} at V_{DD}=2.7V (standard)

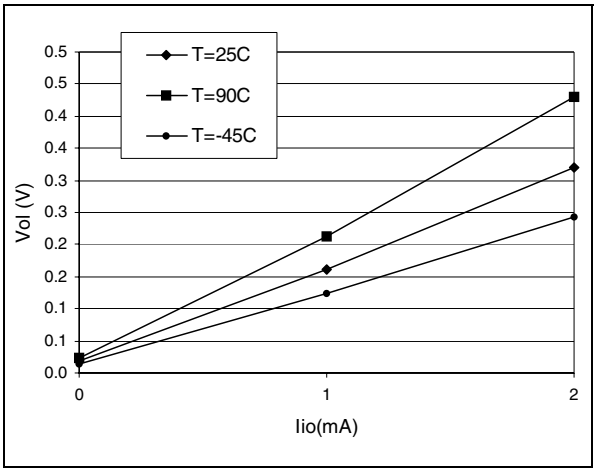
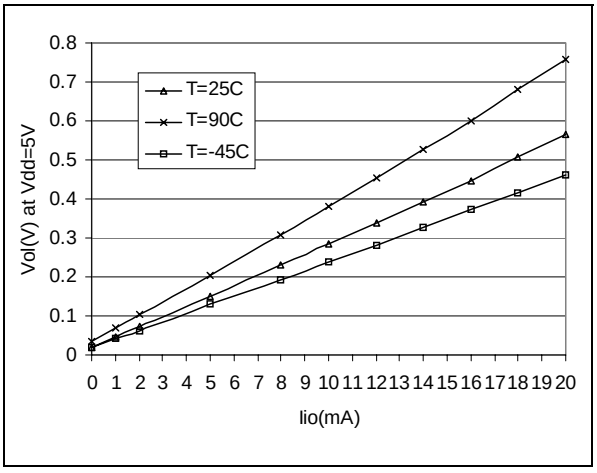


Figure 83. Typ. V_{OL} at V_{DD}=5V (high-sink)



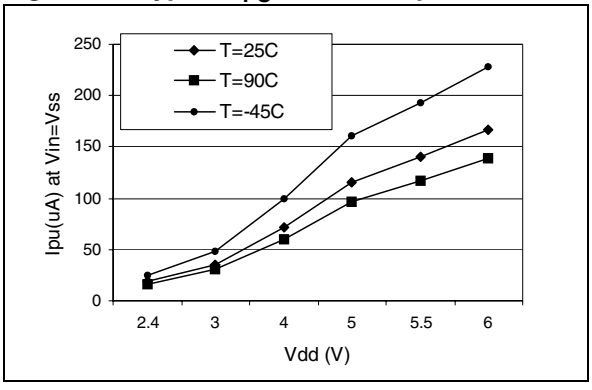
13.9 CONTROL PIN CHARACTERISTICS

13.9.1 Asynchronous $\overline{\text{RESET}}$ Pin

T_A = -40 to +85°C unless otherwise specified

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
V_{IL}	Input low level voltage			V_{SS} - 0.3		0.3xV_{DD}	V
V_{IH}	Input high level voltage			0.7xV_{DD}		V_{DD} + 0.3	
V_{hys}	Schmitt trigger voltage hysteresis[1]				2.5		V
V_{OL}	Output low level voltage [2]	V_{DD}=5V	I_{IO}=+5mA		0.68	0.95	V
			I_{IO}=+2mA		0.28	0.45	
R_{ON}	Pull-up equivalent resistor	V_{DD}=5V		20	40	80	kΩ
		V_{DD}=3V			85		
$t_{w(RSTL)out}$	Generated reset pulse duration	Internal reset sources			30		µs
$t_{h(RSTL)in}$	External reset pulse hold time [3]			20			µs
$t_{g(RSTL)in}$	Filtered glitch duration				200		ns

Figure 90. Typical I_{PU} on $\overline{\text{RESET}}$ pin



Notes:

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

2. The I_{IO} current sunk must always respect the absolute maximum rating specified in Section 13.2.2 and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VSS}.

3. To guarantee the reset of the device, a minimum pulse has to be applied to the $\overline{\text{RESET}}$ pin. All short pulses applied on $\overline{\text{RESET}}$ pin with a duration below $t_{h(RSTL)in}$ can be ignored.

ADC CHARACTERISTICS (Cont'd)

13.12.0.1 General PCB Design Guidelines

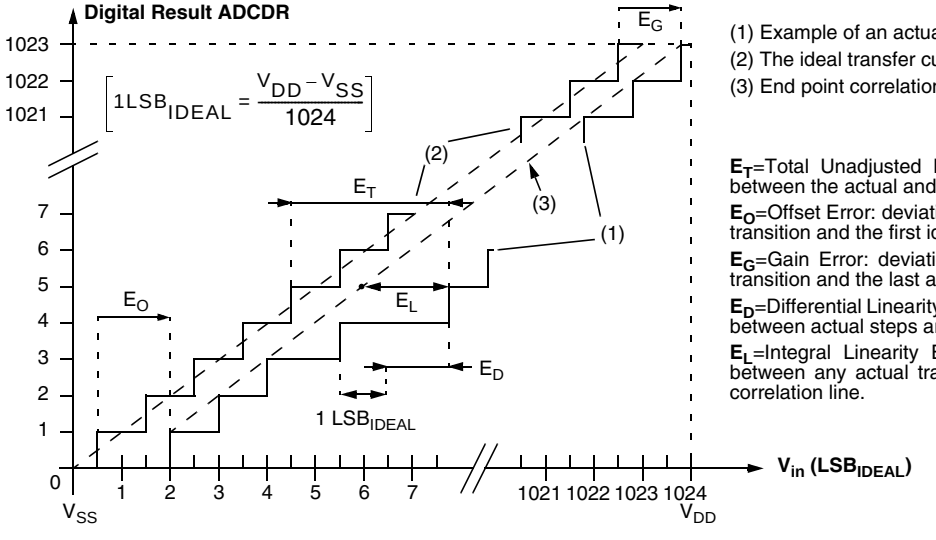
To obtain best results, some general design and layout rules should be followed when designing the application PCB to shield the noise-sensitive, analog physical interface from noise generating CMOS logic signals.

– Properly place components and route the signal traces on the PCB to shield the analog inputs. Analog signals paths should run over the analog ground plane and be as short as possible. Isolate analog signals from digital signals that may switch while the analog inputs are being sampled by the A/D converter. Do not toggle digital outputs on the same I/O port as the A/D input being converted.

ADC Accuracy with f_{CPU}=8 MHz, f_{ADC}=4 MHz R_{AIN}< 10kΩ, V_{DD}= 4.5V to 5.5V

Symbol	Parameter	Conditions	FLASH		ROM		Unit
			Typ[2]	Max	Typ[2]	Max	
$\|E_T\|$	Total unadjusted error [1]		4	6	TBD	TBD	LSB
$\|E_O\|$	Offset error [1]		1	5	TBD	TBD	
$\|E_G\|$	Gain Error [1]		1	4.5	TBD	TBD	
$\|E_D\|$	Differential linearity error[1]		1.5	4.5	TBD	TBD	
$\|E_L\|$	Integral linearity error [1]		3	4.5	TBD	TBD	

Figure 100. ADC Accuracy Characteristics



Notes:

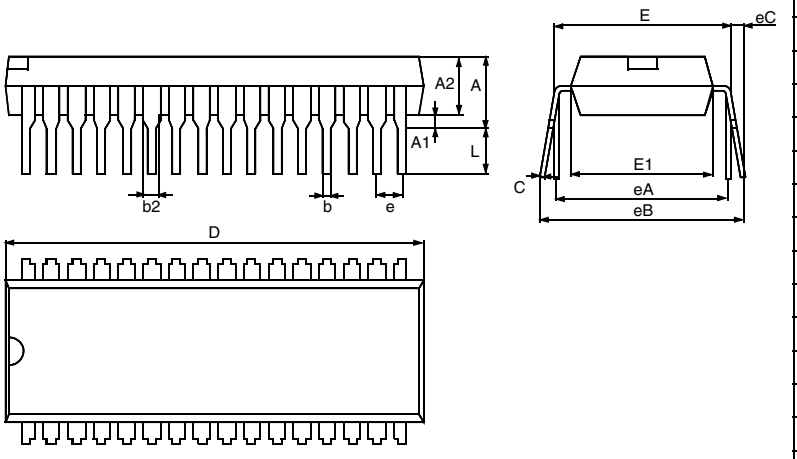
1. ADC Accuracy vs. Negative Injection Current: Injecting negative current on any of the analog input pins significantly reduces the accuracy of the conversion being performed on another analog input.
For I_{INJ-}=0.8mA, the typical leakage induced inside the die is 1.6µA and the effect on the ADC accuracy is a loss of 4 LSB for each 10KΩ increase of the external analog source impedance. It is recommended to add a Schottky diode (pin to ground) to analog pins which may potentially inject negative current. Any positive injection current within the limits specified for $I_{INJ(PIN)}$ and $\Sigma I_{INJ(PIN)}$ in Section 13.8 does not affect the ADC accuracy.

2. Refer to "Typical values" on page 126 for more information on typical ADC accuracy values.

14 PACKAGE CHARACTERISTICS

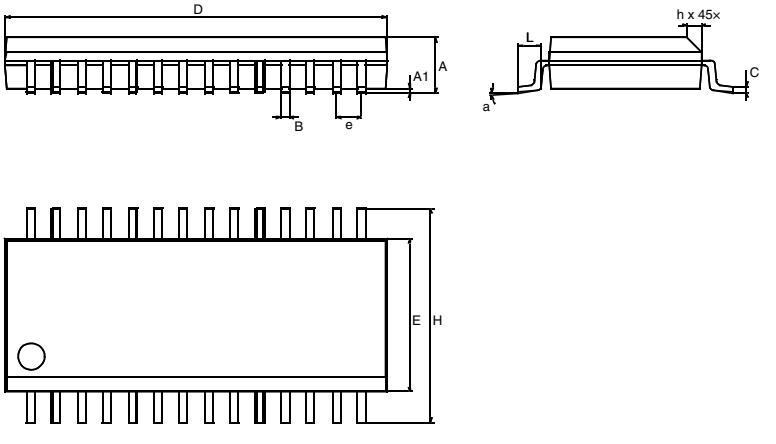
14.1 PACKAGE MECHANICAL DATA

Figure 101. 32-Pin Plastic Dual In-Line Package, Shrink 400-mil Width



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A	3.56	3.76	5.08	0.140	0.148	0.200
A1	0.51			0.020		
A2	3.05	3.56	4.57	0.120	0.140	0.180
b	0.36	0.46	0.58	0.014	0.018	0.023
b1	0.76	1.02	1.40	0.030	0.040	0.055
C	0.20	0.25	0.36	0.008	0.010	0.014
D	27.43		28.45	1.080	1.100	1.120
E	9.91	10.41	11.05	0.390	0.410	0.435
E1	7.62	8.89	9.40	0.300	0.350	0.370
e		1.78			0.070	
eA		10.16			0.400	
eB			12.70			0.500
eC			1.40			0.055
L	2.54	3.05	3.81	0.100	0.120	0.150
	Number of Pins					
N	32					

Figure 102.
Figure 103. 28-Pin Plastic Small Outline Package, 300-mil Width



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A	2.35		2.65	0.093		0.104
A1	0.10		0.30	0.004		0.012
B	0.33		0.51	0.013		0.020
C	0.23		0.32	0.009		0.013
D	17.70		18.10	0.697		0.713
E	7.40		7.60	0.291		0.299
e		1.27			0.050	
H	10.00		10.65	0.394		0.419
h	0.25		0.75	0.010		0.030
α	0°		8°	0°		8°
L	0.40		1.27	0.016		0.050
	Number of Pins					
N	28					

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. 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

Table 29. STMicroelectronics Development Tools

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

15.3.2 PACKAGE/SOCKET FOOTPRINT PROPOSAL

Table 30. Suggested List of SDIP32 Socket Types

Package / Probe	Adaptor / Socket Reference	Same Footprint	Socket Type
SDIP32 EMU PROBE	TEXTOOL 232-1291-00	X	Textool

Table 31. Suggested List of SO28 Socket Types

Package / Probe	Adaptor / Socket Reference	Same Footprint	Socket Type
SO28	YAMAICHI IC51-0282-334-1		Clamshell
EMU PROBE	Adapter from SO28 to SDIP32 footprint (delivered with emulator)	X	SMD to SDIP

Table 32. Suggested LFBGA Socket Type

Package	Socket Reference
LFBGA 6 X6	ENPLAS OTB-36(144)-0.8-04

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.