



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	8MHz
Connectivity	SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	24
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	3.8V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	32-SDIP (0.400", 10.16mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f32ak2b6

Email: info@E-XFL.COM

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

Table of Contents

10.5.2 Main Features	87
	87
10.5.3 General Description	87
10.5.4 Functional Description	89
10.5.5 Low Power Modes	96
10.5.6 Interrupts	96
10.5.7 Register Description	97
10.6 10-BIT A/D CONVERTER (ADC)	. 103
10.6.1 Introduction	. 103
10.6.2 Main Features	. 103
10.6.3 Functional Description	. 104
10.6.4 Low Power Modes	. 104
10.6.5 Interrupts	. 104
	105
	107
	. 107
11.1.1 Inherent	. 108
11.1.2 Immediate	108
11.1.3 DIFECT	100
11 1 5 Indirect (Short Long)	108
11 1.6 Indirect Indexed (Short Long)	109
11.1.7 Relative mode (Direct, Indirect)	. 109
11.2 INSTRUCTION GROUPS	. 110
12 ELECTRICAL CHARACTERISTICS	. 113
12.1 PARAMETER CONDITIONS	. 113
12.1.1 Minimum and Maximum values	. 113
12.1.2 Typical values	. 113
12.1.3 Typical curves	. 113
	•
12.1.4 Loading capacitor	. 113
12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.1.5 Pin input voltage 12.1.5 Pin input voltage	. 113 . 113
12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS	. 113 . 113 . 114
12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics	. 113 . 113 . 114 . 114
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 	. 113 . 113 . 114 . 114 . 114 . 114
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 	. 113 . 113 . 114 . 114 . 114 . 114 . 115
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 	. 113 . 113 . 114 . 114 . 114 . 114 . 115 . 115
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 	. 113 . 113 . 114 . 114 . 114 . 115 . 115 . 115
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 	 113 113 114 114 114 115 115 115 115 116
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 	 113 113 114 114 114 115 115 115 115 116 116
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 	 113 113 114 114 114 115 115 115 115 116 116 116 118
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 12.4.3 On-Chip Peripherals 	 113 113 114 114 114 115 115 115 115 116 116 118 119
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 12.4.3 On-Chip Peripherals 12.5 CLOCK AND TIMING CHARACTERISTICS 	 113 113 114 114 114 115 115 115 115 116 116 116 118 119 120
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 12.4.3 On-Chip Peripherals 12.5 CLOCK AND TIMING CHARACTERISTICS 12.5.1 General Timings 	 113 113 114 114 114 114 115 115 115 115 116 116 116 116 118 119 120 120
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 12.4.3 On-Chip Peripherals 12.5 CLOCK AND TIMING CHARACTERISTICS 12.5.1 General Timings 12.5.2 External Clock Source 12.5 Current Consumption 	 113 113 114 114 114 115 115 115 115 116 116 118 119 120 120 120
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 12.4.3 On-Chip Peripherals 12.5 CLOCK AND TIMING CHARACTERISTICS 12.5.1 General Timings 12.5.2 External Clock Source 12.5.3 Crystal and Ceramic Resonator Oscillators 12.5 A PLL Characteristics 	 113 113 114 114 114 115 115 115 115 116 116 116 116 118 119 120 120 120 121 122
 12.1.4 Loading capacitor 12.1.5 Pin input voltage 12.2 ABSOLUTE MAXIMUM RATINGS 12.2.1 Voltage Characteristics 12.2.2 Current Characteristics 12.2.3 Thermal Characteristics 12.3 OPERATING CONDITIONS 12.3.1 Operating Conditions 12.4 SUPPLY CURRENT CHARACTERISTICS 12.4.1 CURRENT CONSUMPTION 12.4.2 Supply and Clock Managers 12.4.3 On-Chip Peripherals 12.5 CLOCK AND TIMING CHARACTERISTICS 12.5.1 General Timings 12.5.2 External Clock Source 12.5.3 Crystal and Ceramic Resonator Oscillators 12.6 MEMORY CHABACTERISTICS 	 113 113 114 114 114 115 115 115 115 115 116 116 116 118 119 120 120 120 121 123 124

57

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.

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 from 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 16.

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 17 and Figure 18 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 18. 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.

47/



Figure 17. Concurrent Interrupt Management

Figure 18. Nested Interrupt Management



I/O PORTS (Cont'd)

CAUTION: The alternate function must not be activated as long as the pin is configured as input with interrupt, in order to avoid generating spurious interrupts.

Analog alternate function

When the pin is used as an ADC input, the I/O must be configured as floating input. The analog multiplexer (controlled by the ADC registers) switches the analog voltage present on the selected pin to the common analog rail which is connected to the ADC input.

It is recommended not to change the voltage level or loading on any port pin while conversion is in progress. Furthermore it is recommended not to 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 feature of the I/O port such as ADC Input or true 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 are likely to present unwanted side-effects such as spurious interrupt generation.

Figure 28. Interrupt I/O Port State Transitions



9.4 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.

9.5 INTERRUPTS

The external interrupt event generates an interrupt if the corresponding configuration is selected with DDR and OR registers 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
External interrupt on selected external event	-	DDRx ORx	Yes	Yes

10 ON-CHIP PERIPHERALS

10.1 WATCHDOG TIMER (WDG)

10.1.1 Introduction

The Watchdog timer is used to detect the occurrence of a software fault, usually generated by external interference or by unforeseen logical conditions, which causes the application program to abandon its normal sequence. The Watchdog circuit generates an MCU reset on expiry of a programmed time period, unless the program refreshes the counter's contents before the T6 bit becomes cleared.

10.1.2 Main Features

- Programmable free-running downcounter
- Programmable reset
- Reset (if watchdog activated) when the T6 bit reaches zero
- HALT Optional reset on instruction (configurable by option byte)
- Hardware Watchdog selectable by option byte

10.1.3 Functional Description

The counter value stored in the Watchdog Control register (WDGCR bits T[6:0]), is decremented every 16384 f_{OSC2} cycles (approx.), and the length of the timeout period can be programmed by the user in 64 increments.

If the watchdog is activated (the WDGA bit is set) and when the 7-bit timer (bits T[6:0]) rolls over from 40h to 3Fh (T6 becomes cleared), it initiates a reset cycle pulling low the reset pin for typically 30µs.

The application program must write in the WDGCR register at regular intervals during normal operation to prevent an MCU reset. This downcounter is free-running: it counts down even if the watchdog is disabled. The value to be stored in the WDGCR register must be between FFh and C0h:

- The WDGA bit is set (watchdog enabled)
- The T6 bit is set to prevent generating an immediate reset
- The T[5:0] bits contain the number of increments which represents the time delay before the watchdog produces a reset (see Figure 30. Approximate Timeout Duration). The timing varies between a minimum and a maximum value due to the unknown status of the prescaler when writing to the WDGCR register (see Figure 31).

Following a reset, the watchdog is disabled. Once activated it cannot be disabled, except by a reset.

The T6 bit can be used to generate a software reset (the WDGA bit is set and the T6 bit is cleared).

If the watchdog is activated, the HALT instruction will generate a Reset.



Figure 29. Watchdog Block Diagram

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

10.2.5 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.

10.2.6 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.

10.2.7 Register Description MCC CONTROL/STATUS REGISTER (MCCSR) Read/Write

Reset Value: 0000 0000 (00h)

7		

0

мсо	CP1	CP0	SMS	TB1	TB0	OIE	OIF
Dit 7 MCO Main alack out calaction							

Bit 7 = **MCO** Main clock out selection

This bit enables the MCO alternate function on the PF0 I/O port. It is set and cleared by software.

- 0: MCO alternate function disabled (I/O pin free for general-purpose I/O)
- 1: MCO alternate function enabled (f_{CPU} on I/O port)

Note: To reduce power consumption, the MCO function is not active in ACTIVE-HALT mode.

Bit 6:5 = **CP[1:0]** CPU clock prescaler

These bits select the CPU clock prescaler which is applied in the different slow modes. Their action is conditioned by the setting of the SMS bit. These two bits are set and cleared by software

f _{CPU} in SLOW mode	CP1	CP0
f _{OSC2} / 2	0	0
f _{OSC2} / 4	0	1
f _{OSC2} / 8	1	0
f _{OSC2} / 16	1	1

Bit 4 = **SMS** *Slow mode select*

This bit is set and cleared by software. 0: Normal mode. $f_{CPU} = f_{OSC2}$ 1: Slow mode. f_{CPU} is given by CP1, CP0 See Section 8.2 SLOW MODE and Section 10.2 MAIN CLOCK CONTROLLER WITH REAL TIME CLOCK AND BEEPER (MCC/RTC) for more details.

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	Time	TB1	TRO	
Prescaler	f _{OSC2} =4MHz	f _{OSC2} =8MHz	101	100
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.



16-BIT TIMER (Cont'd)

57

Figure 33. Timer Block Diagram



16-BIT TIMER (Cont'd)

16-bit read sequence: (from either the Counter Register or the Alternate Counter Register).

Beginning of the sequence



Sequence completed

The user must read the MS Byte first, then the LS Byte value is buffered automatically.

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

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

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

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

If one of these conditions is false, the interrupt remains pending to be issued as soon as they are both true. Clearing the overflow interrupt request is done in two steps:

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

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

The timer is not affected by WAIT mode.

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

10.3.3.2 External Clock

The external clock (where available) is selected if CC0=1 and CC1=1 in the CR2 register.

The status of the EXEDG bit in the CR2 register determines the type of level transition on the external clock pin EXTCLK that will trigger the free running counter.

The counter is synchronized with the falling edge of the internal CPU clock.

A minimum of four falling edges of the CPU clock must occur between two consecutive active edges of the external clock; thus the external clock frequency must be less than a quarter of the CPU clock frequency.

16-BIT TIMER (Cont'd)

Figure 40. Output Compare Timing Diagram, f_{TIMER} =f_{CPU}/2

INTERNAL CPU CLOCK TIMER CLOCK COUNTER REGISTER OUTPUT COMPARE REGISTER <i>i</i> (OCR <i>i</i>)	2ECF 2ED0 2ED1 2ED2 2ED3 2ED4 2ECF 2ED0 2ED1 2ED2 2ED3 2ED4
OUTPUT COMPARE FLAG i (OCFi)	
OCMP <i>i</i> PIN (OLVL <i>i</i> =1)	

Figure 41. Output Compare Timing Diagram, $f_{TIMER} = f_{CPU}/4$

INTERNAL CPU CLOCK TIMER CLOCK COUNTER REGISTER OUTPUT COMPARE REGISTER <i>i</i> (OCR <i>i</i>) COMPARE REGISTER <i>i</i> LATCH OUTPUT COMPARE FLAG <i>i</i> (OCF <i>i</i>)	
OCMP <i>i</i> PIN (OLVL <i>i</i> =1)	



SERIAL PERIPHERAL INTERFACE (Cont'd)

10.4.3.3 Master Mode Operation

In master mode, the serial clock is output on the SCK pin. The clock frequency, polarity and phase are configured by software (refer to the description of the SPICSR register).

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).

To operate the SPI in master mode, perform the following steps in order (if the SPICSR register is not written first, the SPICR register setting (MSTR bit) may be not taken into account):

1. Write to the SPICR register:

- Select the clock frequency by configuring the SPR[2:0] bits.
- Select the clock polarity and clock phase by configuring the CPOL and CPHA bits. Figure 48 shows the four possible configurations.
 Note: The slave must have the same CPOL and CPHA settings as the master.
- 2. Write to the SPICSR register:
 - Either set the SSM bit and set the SSI bit or clear the SSM bit and tie the SS pin high for the complete byte transmit sequence.
- 3. Write to the SPICR register:
 - Set the MSTR and SPE bits
 <u>Note</u>: MSTR and SPE bits remain set only if SS is high).

The transmit sequence begins when software writes a byte in the SPIDR register.

10.4.3.4 Master Mode Transmit Sequence

When software writes to the SPIDR register, the data byte is loaded into the 8-bit shift register and then shifted out serially to the MOSI pin most significant bit first.

When data transfer is complete:

- The SPIF bit is set by hardware
- An interrupt request is generated if the SPIE bit is set and the interrupt mask in the CCR register is cleared.

Clearing the SPIF bit is performed by the following software sequence:

- 1. An access to the SPICSR register while the SPIF bit is set
- 2. A read to the SPIDR register.

57

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

10.4.3.5 Slave Mode Operation

In slave mode, the serial clock is received on the SCK pin from the master device.

To operate the SPI in slave mode:

- 1. Write to the SPICSR register to perform the following actions:
 - Select the clock polarity and clock phase by configuring the CPOL and CPHA bits (see Figure 48).
 Note: The slave must have the same CPOL and CPHA settings as the master.
 - Manage the SS pin as described in Section 10.4.3.2 and Figure 46. If CPHA=1 SS must be held low continuously. If CPHA=0 SS must be held low during byte transmission and pulled up between each byte to let the slave write in the shift register.
- 2. Write to the SPICR register to clear the MSTR bit and set the SPE bit to enable the SPI I/O functions.

10.4.3.6 Slave Mode Transmit Sequence

When software writes to the SPIDR register, the data byte is loaded into the 8-bit shift register and then shifted out serially to the MISO pin most significant bit first.

The transmit sequence begins when the slave device receives the clock signal and the most significant bit of the data on its MOSI pin.

When data transfer is complete:

- The SPIF bit is set by hardware
- An interrupt request is generated if SPIE bit is set and interrupt mask in the CCR register is cleared.

Clearing the SPIF bit is performed by the following software sequence:

1. An access to the SPICSR register while the SPIF bit is set.

2. A write or a read to the SPIDR register.

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

The SPIF bit can be cleared during a second transmission; however, it must be cleared before the second SPIF bit in order to prevent an Overrun condition (see Section 10.4.5.2).

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 49).

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 10.4.5.2). An interrupt is generated if SPIE = 1 in SPICR register. The OVR bit is cleared by software reading the SPICSR register. 0: No overrun error

1: Overrun error detected

Bit 4 = MODF Mode Fault flag (Read only).

This bit is set by hardware when the SS pin is pulled low in master mode (see Section 10.4.5.1 Master Mode Fault (MODF)). An SPI interrupt can be generated if SPIE=1 in the SPICSR register. This bit is cleared by a software sequence (An access to the SPICR 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 10.4.3.2 Slave Select Management.

- 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 44).

57

SERIAL COMMUNICATIONS INTERFACE (Cont'd)

10.5.4.2 Transmitter

The transmitter can send data words of either 8 or 9 bits depending on the M bit status. When the M bit is set, word length is 9 bits and the 9th bit (the MSB) has to be stored in the T8 bit in the SCICR1 register.

Character Transmission

During an SCI transmission, data shifts out least significant bit first on the TDO pin. In this mode, the SCIDR register consists of a buffer (TDR) between the internal bus and the transmit shift register (see Figure 51).

Procedure

- Select the M bit to define the word length.
- Select the desired baud rate using the SCIBRR and the SCIETPR registers.
- Set the TE bit to assign the TDO pin to the alternate function and to send a idle frame as first transmission.
- Access the SCISR register and write the data to send in the SCIDR register (this sequence clears the TDRE bit). Repeat this sequence for each data to be transmitted.

Clearing the TDRE bit is always performed by the following software sequence:

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

The TDRE bit is set by hardware and it indicates:

- The TDR register is empty.
- The data transfer is beginning.
- The next data can be written in the SCIDR register without overwriting the previous data.

This flag generates an interrupt if the TIE bit is set and the I bit is cleared in the CCR register.

When a transmission is taking place, a write instruction to the SCIDR register stores the data in the TDR register and which is copied in the shift register at the end of the current transmission.

When no transmission is taking place, a write instruction to the SCIDR register places the data directly in the shift register, the data transmission starts, and the TDRE bit is immediately set. When a frame transmission is complete (after the stop bit) the TC bit is set and an interrupt is generated if the TCIE is set and the I bit is cleared in the CCR register.

Clearing the TC bit is performed by the following software sequence:

1. An access to the SCISR register

A write to the SCIDR register

Note: The TDRE and TC bits are cleared by the same software sequence.

Break Characters

Setting the SBK bit loads the shift register with a break character. The break frame length depends on the M bit (see Figure 52).

As long as the SBK bit is set, the SCI send break frames to the TDO pin. After clearing this bit by software the SCI insert a logic 1 bit at the end of the last break frame to guarantee the recognition of the start bit of the next frame.

Idle Characters

Setting the TE bit drives the SCI to send an idle frame before the first data frame.

Clearing and then setting the TE bit during a transmission sends an idle frame after the current word.

Note: Resetting and setting the TE bit causes the data in the TDR register to be lost. Therefore the best time to toggle the TE bit is when the TDRE bit is set i.e. before writing the next byte in the SCIDR.



SERIAL COMMUNICATIONS INTERFACE (Cont'd)

10.5.4.3 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 51).

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 SPI 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

5/

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 a overrun error occurs:

- The OR bit is set.
- The RDR content will not be lost.
- The shift register will be 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 10.5.4.10.

INSTRUCTION SET OVERVIEW (Cont'd)

11.2 INSTRUCTION GROUPS

The ST7 family devices use an Instruction Set consisting of 63 instructions. The instructions may

be subdivided into 13 main groups as illustrated in the following table:

Load and Transfer	LD	CLR						
Stack operation	PUSH	POP	RSP					
Increment/Decrement	INC	DEC						
Compare and Tests	СР	TNZ	BCP					
Logical operations	AND	OR	XOR	CPL	NEG			
Bit Operation	BSET	BRES						
Conditional Bit Test and Branch	BTJT	BTJF						
Arithmetic operations	ADC	ADD	SUB	SBC	MUL			
Shift and Rotates	SLL	SRL	SRA	RLC	RRC	SWAP	SLA	
Unconditional Jump or Call	JRA	JRT	JRF	JP	CALL	CALLR	NOP	RET
Conditional Branch	JRxx							
Interruption management	TRAP	WFI	HALT	IRET				
Condition Code Flag modification	SIM	RIM	SCF	RCF				

Using a pre-byte

The instructions are described with one to four opcodes.

In order to extend the number of available opcodes for an 8-bit CPU (256 opcodes), three different prebyte opcodes are defined. These prebytes modify the meaning of the instruction they precede.

The whole instruction becomes:

- PC-2 End of previous instruction
- PC-1 Prebyte
- PC opcode

PC+1 Additional word (0 to 2) according to the number of bytes required to compute the effective address

These prebytes enable instruction in Y as well as indirect addressing modes to be implemented. They precede the opcode of the instruction in X or the instruction using direct addressing mode. The prebytes are:

PDY 90 Replace an X based instruction using immediate, direct, indexed, or inherent addressing mode by a Y one.

PIX 92 Replace an instruction using direct, direct bit, or direct relative addressing mode to an instruction using the corresponding indirect addressing mode.

It also changes an instruction using X indexed addressing mode to an instruction using indirect X indexed addressing mode.

PIY 91 Replace an instruction using X indirect indexed addressing mode by a Y one.



12.4 SUPPLY CURRENT CHARACTERISTICS

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

12.4.1 CURRENT CONSUMPTION

Cumhal	Devemeter	Conditions	Flash I	Devices	ROM Devices		Unit
Symbol Parameter		Conditions	Тур	Max ¹⁾	Тур	Max ¹⁾	Unit
	Supply current in RUN mode ²⁾	$\begin{array}{l} f_{OSC}{=}2MHz, f_{CPU}{=}1MHz\\ f_{OSC}{=}4MHz, f_{CPU}{=}2MHz\\ f_{OSC}{=}8MHz, f_{CPU}{=}4MHz\\ f_{OSC}{=}16MHz, f_{CPU}{=}8MHz \end{array}$	1 1.4 2.4 4.4	2.3 3.5 5.3 7.0	1.3 2.0 3.6 7.1	2.0 3.0 5 10	mA
	Supply current in SLOW mode ²⁾	$\begin{array}{l} f_{OSC}=2MHz, f_{CPU}=62.5kHz\\ f_{OSC}=4MHz, f_{CPU}=125kHz\\ f_{OSC}=8MHz, f_{CPU}=250kHz\\ f_{OSC}=16MHz, f_{CPU}=500kHz \end{array}$	0.48 0.53 0.63 0.80	1 1.1 1.2 1.4	0.6 0.7 0.8 1.1	1.8 2.1 2.4 3.0	mA
Supply current in V I _{DD} Supply current in S mode ²⁾	Supply current in WAIT mode ²⁾	$\begin{array}{l} f_{OSC}=2MHz, \ f_{CPU}=1MHz\\ f_{OSC}=4MHz, \ f_{CPU}=2MHz\\ f_{OSC}=8MHz, \ f_{CPU}=4MHz\\ f_{OSC}=16MHz, \ f_{CPU}=8MHz \end{array}$	0.6 0.9 1.3 2.3	1.8 2.2 2.6 3.6	1 1.5 2.5 4.5	1.3 2.0 3.3 6	mA
	Supply current in SLOW WAIT mode ²⁾	$\begin{array}{l} f_{OSC}=2MHz, f_{CPU}=62.5kHz\\ f_{OSC}=4MHz, f_{CPU}=125kHz\\ f_{OSC}=8MHz, f_{CPU}=250kHz\\ f_{OSC}=16MHz, f_{CPU}=500kHz \end{array}$	430 470 530 660	950 1000 1050 1200	70 100 200 350	200 300 600 1200	μA
	Supply current in HALT mode ³⁾	-40°C≤T _A ≤+85°C	<1	10	<1	10	μA
		-40°C≤T _A ≤+125°C	5	50	<1	50	•
	Supply current in ACTIVE-HALT mode ⁴⁾	$f_{OSC}=2MHz$ $f_{OSC}=4MHz$ $f_{OSC}=8MHz$ $f_{OSC}=16MHz$	60 100 180 340	160 200 300 500	22 44 86 170	30 60 120 300	μA

Notes:

1. Data based on characterization results, tested in production at V_{DD} max. and f_{CPU} max.

2. Measurements are done in the following conditions:

- Progam executed from RAM, CPU running with RAM access. The increase in consumption when executing from Flash is 50%.

- All I/O pins in input mode with a static value at V_{DD} or V_{SS} (no load)
- All peripherals in reset state.
- Clock input (OSC1) driven by external square wave.

In SLOW and SLOW WAIT mode, f_{CPU} is based on f_{OSC} divided by 32.
 To obtain the total current consumption of the device, add the clock source (Section 12.5.3) and the peripheral power consumption (Section 12.4.3).

3. All I/O pins in push-pull 0 mode (when applicable) with a static value at V_{DD} or V_{SS} (no load). Data based on characterization results, tested in production at V_{DD} max. and f_{CPU} max.

4. Data based on characterisation results, not tested in production. All I/O pins in push-pull 0 mode (when applicable) with a static value at V_{DD} or V_{SS} (no load); clock input (OSC1) driven by external square wave. To obtain the total current consumption of the device, add the clock source consumption (Section 12.5.3).



I/O PORT PIN CHARACTERISTICS (Cont'd)

Figure 71. Typical V_{OL} vs. V_{DD} (std. ports)



Figure 72. Typical V_{OL} vs. V_{DD} (high-sink ports)







CONTROL PIN CHARACTERISTICS (Cont'd) 12.9.2 ICCSEL/V_{PP} Pin

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

Symbol	Parameter	Conditions	Min	Max	Unit	
V.	Input low level voltage 1)	FLASH versions	V _{SS}	0.2		
۷IL		ROM versions	V _{SS}	0.3xV _{DD}	V	
V	Input high level voltage ¹⁾	FLASH versions	V _{DD} -0.1	12.6	v	
VIН		ROM versions	0.7xV _{DD}	V _{DD}		
۱ _L	Input leakage current	V _{IN} =V _{SS}		±1	μA	

Figure 75. Two typical Applications with ICCSEL/V_{PP} Pin $^{\rm 2)}$



Notes:

<u>ل</u>حک

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

2. When ICC mode is not required by the application ICCSEL/V_{PP} pin must be tied to V_{SS} .

13.3 SOLDERING INFORMATION

In accordance with the RoHS European directive, all STMicroelectronics packages will be converted in 2005 to lead-free technology, named ECO-PACKTM (for a detailed roadmap, please refer to PCN CRP/04/744 "Lead-free Conversion Program - Compliance with RoHS", issued November 18th, 2004).

- ECOPACKTM packages are qualified according to the JEDEC STD-020B compliant soldering profile.
- Detailed information on the STMicroelectronic ECOPACKTM transition program is available on www.st.com/stonline/leadfree/, with specific technical Application notes covering the main technical aspects related to lead-free conversion (AN2033, AN2034, AN2035, AN2036).

Backward and forward compatibility:

The main difference between Pb and Pb-free soldering process is the temperature range.

- ECOPACKTM TQFP, SDIP and SO packages are fully compatible with Lead (Pb) containing soldering process (see application note AN2034)
- TQFP, SDIP and SO Pb-packages are compatible with Lead-free soldering process, nevertheless it's the customer's duty to verify that the Pb-packages maximum temperature (mentioned on the Inner box label) is compatible with their Lead-free soldering temperature.

Table 26. Soldering Compatibility (wave and reflow soldering process)

Package	Plating material devices	Pb solder paste	Pb-free solder paste
SDIP & PDIP	Sn (pure Tin)	Yes	Yes *
TQFP and SO	NiPdAu (Nickel-palladium-Gold)	Yes	Yes *

* Assemblers must verify that the Pb-package maximum temperature (mentioned on the Inner box label) is compatible with their Lead-free soldering process.

14.5 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 obtain from the STMicroelectronics Internet site: → http://:mcu.st.com.

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

Emulators

Two types of emulators are available from ST for the ST7232A family:

- ST7 DVP3 entry-level emulator offers a flexible and modular debugging and programming solution. SDIP42 & SDIP32 probes/adapters are included, other packages need a specific connection kit (refer to Table 28)
- ST7 EMU3 high-end emulator is delivered with everything (probes, TEB, adapters etc.) needed to start emulating the ST7232x family. 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 28.

Table 28. STMicroelectronics Development Tools

In-circuit Debugging Kit

Two configurations are available from ST:

- STXF521-IND/USB: Low-cost In-Circuit Debugging kit from Softec Microsystems. Includes STX-InDART/USB board (USB port) and a specific demo board for ST72521 (TQFP64)
- STxF-INDART

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 board

 ST7232x-EVAL with ICC connector for programming capability. Provides direct connection to ST7-DVP3 emulator. Supplied with daughter boards (core module) for ST72F32A chips.

		Programming			
Supported	ST7 DVP3 Series		ST7 EM		
Products	Emulator	Connection kit	Emulator	Active Probe & T.E.B.	ICC Socket Board
ST72F32AJ, ST72F324AJ	ST7MDT20-DVP3	ST7MDT20-T44/ DVP	ST7MDT20J-		ST7SP00 1/m ¹
ST7232AK, ST72F32AK	ST7MDT20-DVP3	ST7MDT20-T32/ DVP	EMU3	ST/WDT20J-TEB	31/3D20J/XX

Note 1: Add suffix /EU, /UK, /US for the power supply of your region.



14.6 ST7 APPLICATION NOTES

Table 30. ST7 Application Notes

IDENTIFICATION	DESCRIPTION
APPLICATION EX	AMPLES
AN1658	SERIAL NUMBERING IMPLEMENTATION
AN1720	MANAGING THE READ-OUT PROTECTION IN FLASH MICROCONTROLLERS
AN1755	A HIGH RESOLUTION/PRECISION THERMOMETER USING ST7 AND NE555
EXAMPLE DRIVER	R
AN 969	SCI COMMUNICATION BETWEEN ST7 AND PC
AN 970	SPI COMMUNICATION BETWEEN ST7 AND EEPROM
AN 972	ST7 SOFTWARE SPI MASTER COMMUNICATION
AN 973	SCI SOFTWARE COMMUNICATION WITH A PC USING ST72251 16-BIT TIMER
AN 974	REAL TIME CLOCK WITH ST7 TIMER OUTPUT COMPARE
AN 976	DRIVING A BUZZER THROUGH ST7 TIMER PWM FUNCTION
AN 979	DRIVING AN ANALOG KEYBOARD WITH THE ST7 ADC
AN 980	ST7 KEYPAD DECODING TECHNIQUES, IMPLEMENTING WAKE-UP ON KEYSTROKE
AN1041	USING ST7 PWM SIGNAL TO GENERATE ANALOG OUTPUT (SINUSOÏD)
AN1044	MULTIPLE INTERRUPT SOURCES MANAGEMENT FOR ST7 MCUS
AN1046	UART EMULATION SOFTWARE
AN1047	MANAGING RECEPTION ERRORS WITH THE ST7 SCI PERIPHERALS
AN1048	ST7 SOFTWARE LCD DRIVER
AN1078	PWM DUTY CYCLE SWITCH IMPLEMENTING TRUE 0% & 100% DUTY CYCLE
AN1445	EMULATED 16 BIT SLAVE SPI
AN1504	STARTING A PWM SIGNAL DIRECTLY AT HIGH LEVEL USING THE ST7 16-BIT TIMER
GENERAL PURPC	DSE
AN1476	LOW COST POWER SUPPLY FOR HOME APPLIANCES
AN1709	EMC DESIGN FOR ST MICROCONTROLLERS
AN1752	ST72324 QUICK REFERENCE NOTE
PRODUCT EVALU	ATION
AN 910	PERFORMANCE BENCHMARKING
AN 990	ST7 BENEFITS VERSUS INDUSTRY STANDARD
AN1150	BENCHMARK ST72 VS PC16
AN1151	PERFORMANCE COMPARISON BETWEEN ST72254 & PC16F876
AN1278	LIN (LOCAL INTERCONNECT NETWORK) SOLUTIONS
PRODUCT MIGRA	TION
AN1131	MIGRATING APPLICATIONS FROM ST72511/311/214/124 TO ST72521/321/324
PRODUCT OPTIM	IZATION
AN 982	USING ST7 WITH CERAMIC RESONATOR
AN1014	HOW TO MINIMIZE THE ST7 POWER CONSUMPTION
AN1015	SOFTWARE TECHNIQUES FOR IMPROVING MICROCONTROLLER EMC PERFORMANCE
AN1070	ST7 CHECKSUM SELF-CHECKING CAPABILITY
AN1181	ELECTROSTATIC DISCHARGE SENSITIVE MEASUREMENT
AN1502	EMULATED DATA EEPROM WITH ST7 HDFLASH MEMORY
AN1530	ACCURATE TIMEBASE FOR LOW-COST ST7 APPLICATIONS WITH INTERNAL RC OSCILLA- TOR
AN1636	UNDERSTANDING AND MINIMIZING ADC CONVERSION ERRORS
PROGRAMMING A	AND TOOLS
AN 978	ST7 VISUAL DEVELOP SOFTWARE KEY DEBUGGING FEATURES

