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	Not For New Designs
Core Processor	ST7
Core Size	8-Bit
Speed	16MHz
Connectivity	LINbusSCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	15
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 7x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SO
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st7flite35f2m6tr

Email: info@E-XFL.COM Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

DATA EEPROM (Cont'd)

5.4 POWER SAVING MODES

Wait mode

The DATA EEPROM can enter WAIT mode on execution of the WFI instruction of the microcontroller or when the microcontroller enters Active-HALT mode.The DATA EEPROM will immediately enter this mode if there is no programming in progress, otherwise the DATA EEPROM will finish the cycle and then enter WAIT mode.

Active-Halt mode

Refer to Wait mode.

Halt mode

The DATA EEPROM immediately enters HALT mode if the microcontroller executes the HALT instruction. Therefore the EEPROM will stop the function in progress, and data may be corrupted.

5.5 ACCESS ERROR HANDLING

If a read access occurs while E2LAT=1, then the data bus will not be driven.

If a write access occurs while E2LAT=0, then the data on the bus will not be latched.

If a programming cycle is interrupted (by RESET action), the integrity of the data in memory is not guaranteed.

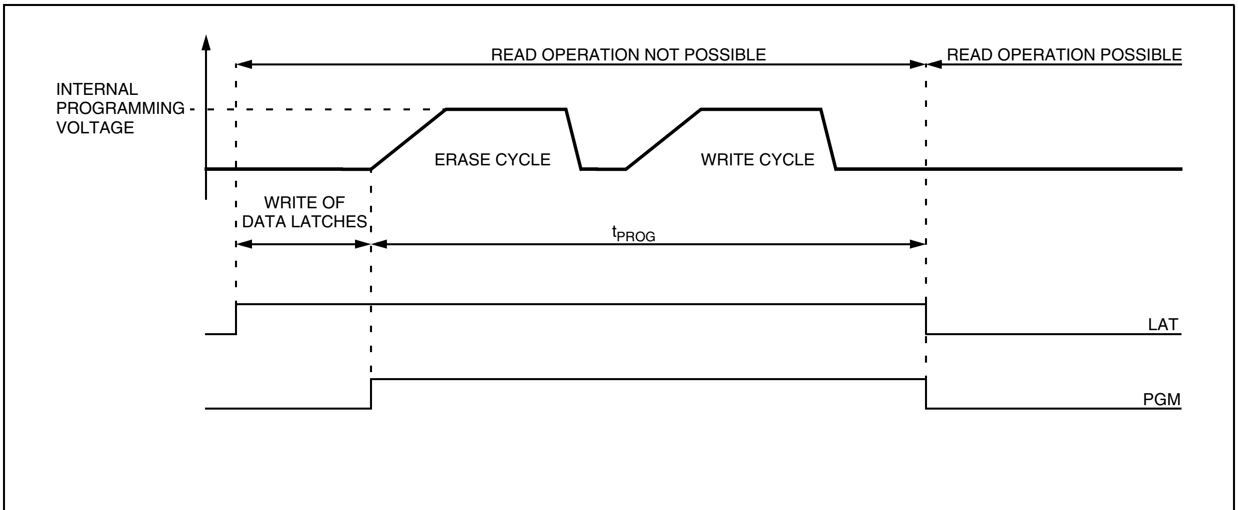
5.6 Data EEPROM Read-out Protection

The read-out protection is enabled through an option bit (see section 15.1 on page 161).

When this option is selected, the programs and data stored in the EEPROM memory are protected against read-out (including a re-write protection). In Flash devices, when this protection is removed by reprogramming the Option Byte, the entire Program memory and EEPROM is first automatically erased.

Note: Both Program Memory and data EEPROM are protected using the same option bit.

Figure 9. Data EEPROM Programming Cycle



SYSTEM INTEGRITY MANAGEMENT (Cont'd)

7.6.3 Low Power Modes

Mode	Description
WAIT	No effect on SI. AVD interrupts cause the device to exit from Wait mode.
HALT	The SICSR register is frozen. The AVD becomes inactive and the AVD interrupt cannot be used to exit from Halt mode.

7.6.3.1 Interrupts

The AVD interrupt event generates an interrupt if the corresponding Enable Control Bit (AVDIE) is set and the interrupt mask in the CC register is reset (RIM instruction).

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
AVD event	AVDF	AVDIE	Yes	No

8 INTERRUPTS

The ST7 core may be interrupted by one of two different methods: Maskable hardware interrupts as listed in the “interrupt mapping” table and a non-maskable software interrupt (TRAP). The Interrupt processing flowchart is shown in Figure 20.

The maskable interrupts must be enabled by clearing the I bit in order to be serviced. However, disabled interrupts may be latched and processed when they are enabled (see external interrupts subsection).

Note: After reset, all interrupts are disabled.

When an interrupt has to be serviced:

– Normal processing is suspended at the end of the current instruction execution.

– The PC, X, A and CC registers are saved onto the stack.

– The I bit of the CC register is set to prevent additional interrupts.

– The PC is then loaded with the interrupt vector of the interrupt to service and the first instruction of the interrupt service routine is fetched (refer to the Interrupt Mapping table for vector addresses).

The interrupt service routine should finish with the IRET instruction which causes the contents of the saved registers to be recovered from the stack.

Note: As a consequence of the IRET instruction, the I bit is cleared and the main program resumes.

Priority Management

By default, a servicing interrupt cannot be interrupted because the I bit is set by hardware entering in interrupt routine.

In the case when several interrupts are simultaneously pending, an hardware priority defines which one will be serviced first (see the Interrupt Mapping table).

Interrupts and Low Power Mode

All interrupts allow the processor to leave the WAIT low power mode. Only external and specifically mentioned interrupts allow the processor to leave the HALT low power mode (refer to the “Exit from HALT” column in the Interrupt Mapping table).

8.1 NON MASKABLE SOFTWARE INTERRUPT

This interrupt is entered when the TRAP instruction is executed regardless of the state of the I bit. It is serviced according to the flowchart in Figure 20.

8.2 EXTERNAL INTERRUPTS

External interrupt vectors can be loaded into the PC register if the corresponding external interrupt occurred and if the I bit is cleared. These interrupts allow the processor to leave the HALT low power mode.

The external interrupt polarity is selected through the miscellaneous register or interrupt register (if available).

An external interrupt triggered on edge will be latched and the interrupt request automatically cleared upon entering the interrupt service routine.

Caution: The type of sensitivity defined in the Miscellaneous or Interrupt register (if available) applies to the ei source. In case of a NANDed source (as described in the I/O ports section), a low level on an I/O pin, configured as input with interrupt, masks the interrupt request even in case of rising-edge sensitivity.

8.3 PERIPHERAL INTERRUPTS

Different peripheral interrupt flags in the status register are able to cause an interrupt when they are active if both:

– The I bit of the CC register is cleared.

– The corresponding enable bit is set in the control register.

If any of these two conditions is false, the interrupt is latched and thus remains pending.

Clearing an interrupt request is done by:

– Writing “0” to the corresponding bit in the status register or

– Access to the status register while the flag is set followed by a read or write of an associated register.

Note: The clearing sequence resets the internal latch. A pending interrupt (that is, waiting for being enabled) will therefore be lost if the clear sequence is executed.

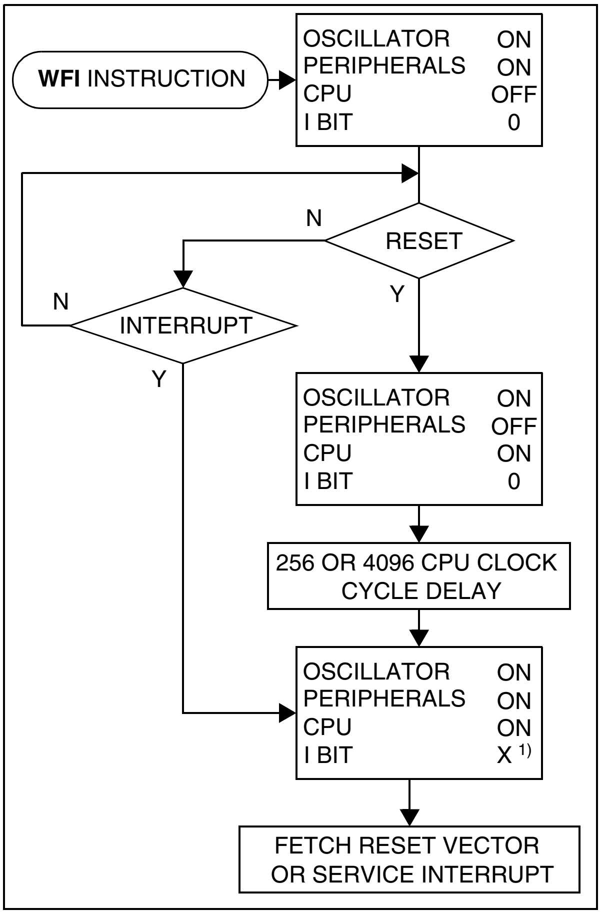
POWER SAVING MODES (Cont'd)

9.3 WAIT MODE

WAIT mode places the MCU in a low power consumption mode by stopping the CPU.
This power saving mode is selected by calling the 'WFI' instruction.
All peripherals remain active. During WAIT mode, the I bit of the CC register is cleared, to enable all interrupts. All other registers and memory remain unchanged. The MCU remains in WAIT mode until an interrupt or RESET occurs, whereupon the Program Counter branches to the starting address of the interrupt or Reset service routine.
The MCU will remain in WAIT mode until a Reset or an Interrupt occurs, causing it to wake up.

Refer to Figure 23.

Figure 23. WAIT Mode Flow-chart



Note:

1. Before servicing an interrupt, the CC register is pushed on the stack. The I bit of the CC register is set during the interrupt routine and cleared when the CC register is popped.

POWER SAVING MODES (Cont'd)

9.4 HALT MODE

The HALT mode is the lowest power consumption mode of the MCU. It is entered by executing the 'HALT' instruction when ACTIVE-HALT is disabled (see section 9.5 on page 42 for more details) and when the AWUEN bit in the AWUCSR register is cleared.

The MCU can exit HALT mode on reception of either a specific interrupt (see Table 6, "Interrupt Mapping," on page 36) or a RESET. When exiting HALT mode by means of a RESET or an interrupt, the oscillator is immediately turned on and the 256 CPU cycle delay is used to stabilize the oscillator. After the start up delay, the CPU resumes operation by servicing the interrupt or by fetching the reset vector which woke it up (see Figure 25).

When entering HALT mode, the I bit in the CC register is forced to 0 to enable interrupts. Therefore, if an interrupt is pending, the MCU wakes up immediately.

In HALT mode, the main oscillator is turned off causing all internal processing to be stopped, including the operation of the on-chip peripherals. All peripherals are not clocked except the ones which get their clock supply from another clock generator (such as an external or auxiliary oscillator).

The compatibility of Watchdog operation with HALT mode is configured by the "WDGHALT" option bit of the option byte. The HALT instruction when executed while the Watchdog system is enabled, can generate a Watchdog RESET (see section 15.1 on page 161 for more details).

Figure 24. HALT Timing Overview

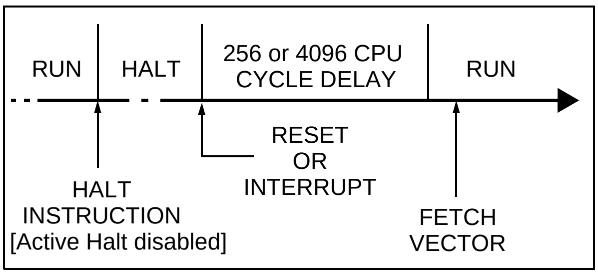
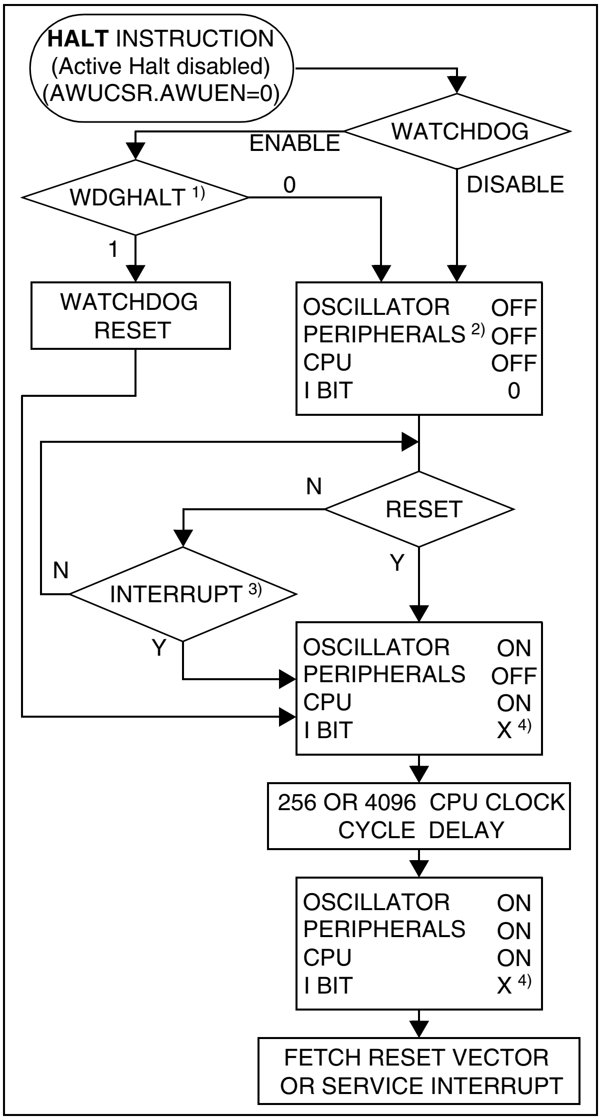


Figure 25. HALT Mode Flow-chart



Notes:

1. WDGHALT is an option bit. See option byte section for more details.

2. Peripheral clocked with an external clock source can still be active.

3. Only some specific interrupts can exit the MCU from HALT mode (such as external interrupt). Refer to Table 6, "Interrupt Mapping," on page 36 for more details.

4. Before servicing an interrupt, the CC register is pushed on the stack. The I bit of the CC register is set during the interrupt routine and cleared when the CC register is popped.

POWER SAVING MODES (Cont'd)

9.4.0.1 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" or "floating 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 memory. For example, avoid defining a constant in program memory 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).

9.5 ACTIVE-HALT MODE

ACTIVE-HALT mode is the lowest power consumption mode of the MCU with a real time clock (RTC) available. It is entered by executing the 'HALT' instruction. The decision to enter either in ACTIVE-HALT or HALT mode is given by the LTCSR/ATCSR register status as shown in the following table:.

LTCSR1 TB1IE bit	ATCSR OVFIE1 bit	ATCSR CK1 bit	ATCSR CK0 bit	Meaning
0	x	x	0	ACTIVE-HALT mode disabled
0	0	x	x	ACTIVE-HALT mode disabled
1	x	x	x	ACTIVE-HALT mode enabled
x	1	0	1	ACTIVE-HALT mode enabled

The MCU can exit ACTIVE-HALT mode on reception of a specific interrupt (see Table 6, "Interrupt Mapping," on page 36) or a RESET.

– When exiting ACTIVE-HALT mode by means of a RESET, a 256 CPU cycle delay occurs. After the start up delay, the CPU resumes operation by fetching the reset vector which woke it up (see Figure 27).

– When exiting ACTIVE-HALT mode by means of an interrupt, the CPU immediately resumes operation by servicing the interrupt vector which woke it up (see Figure 27).

When entering ACTIVE-HALT mode, the I bit in the CC register is cleared to enable interrupts. Therefore, if an interrupt is pending, the MCU wakes up immediately (see Note 3).

In ACTIVE-HALT mode, only the main oscillator and the selected timer counter (LT/AT) are running to keep a wake-up time base. All other peripherals are not clocked except those which get their clock supply from another clock generator (such as external or auxiliary oscillator).

Note: As soon as ACTIVE-HALT is enabled, executing a HALT instruction while the Watchdog is active does not generate a RESET.
This means that the device cannot spend more than a defined delay in this power saving mode.

I/O PORTS (Cont'd)

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

Standard Ports

PA7:0, PB6:0

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

Interrupt Ports

Ports where the external interrupt capability is selected using the EISR register

MODE	DDR	OR
floating input	0	0
pull-up interrupt input	0	1

Table 11. Port Configuration (Standard ports)

Port	Pin name	Input (DDR=0)		Output (DDR=1)	
		OR = 0	OR = 1	OR = 0	OR = 1
Port A	PA7:0	floating	pull-up	open drain	push-pull
Port B	PB6:0	floating	pull-up	open drain	push-pull

Note: On ports where the external interrupt capability is selected using the EISR register, the configuration will be as follows:

Table 12. Port Configuration (external interrupts)

Port	Pin name	Input with interrupt (DDR=0 ; EISR≠00)	
		OR = 0	OR = 1
Port A	PA6:1	floating	pull-up
Port B	PB5:0	floating	pull-up

Table 13. I/O Port Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0000h	**PADR** Reset Value	MSB 1	1	1	1	1	1	1	LSB 1
0001h	**PADDR** Reset Value	MSB 0	0	0	0	0	0	0	LSB 0
0002h	**PAOR** Reset Value	MSB 0	1	0	0	0	0	0	LSB 0
0003h	**PBDR** Reset Value	MSB 1	1	1	1	1	1	1	LSB 1
0004h	**PBDDR** Reset Value	MSB 0	0	0	0	0	0	0	LSB 0
0005h	**PBOR** Reset Value	MSB 0	0	0	0	0	0	0	LSB 0

DUAL 12-BIT AUTORELOAD TIMER 3 (Cont'd)

Dead Time Generation

A dead time can be inserted between PWM0 and PWM1 using the DTGR register. This is required for half-bridge driving where PWM signals must not be overlapped. The non-overlapping PWM0/ PWM1 signals are generated through a programmable dead time by setting the DTE bit.

Dead time value = DT[6:0] x Tcounter1

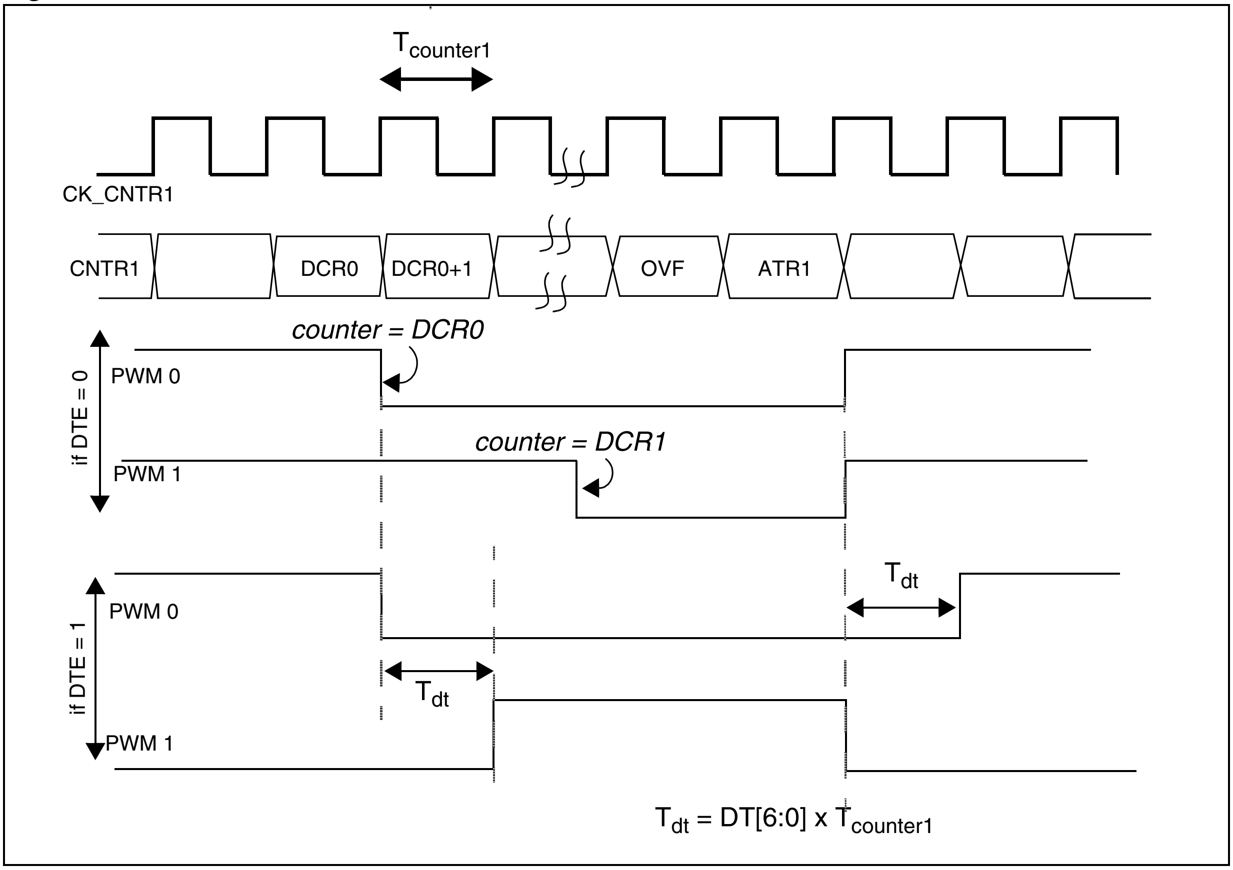
DTGR[7:0] is buffered inside so as to avoid deforming the current PWM cycle. The DTGR effect will take place only after an overflow.

Notes:

1. Dead time is generated only when DTE=1 and DT[6:0] ≠ 0. If DTE is set and DT[6:0]=0, PWM output signals will be at their reset state.

2. Half Bridge driving is possible only if polarities of PWM0 and PWM1 are not inverted, i.e. if OP0 and OP1 are not set. If polarity is inverted, overlapping PWM0/PWM1 signals will be generated.

Figure 39. Dead Time Generation



In the above example, when the DTE bit is set:

– PWM goes low at DCR0 match and goes high at ATR1+Tdt

– PWM1 goes high at DCR0+Tdt and goes low at ATR match.

With this programmable delay (Tdt), the PWM0 and PWM1 signals which are generated are not overlapped.

DUAL 12-BIT AUTORELOAD TIMER 3 (Cont'd)

Break Function

The break function can be used to perform an emergency shutdown of the application being driven by the PWM signals.

The break function is activated by the external BREAK pin (active low). In order to use the BREAK pin it must be previously enabled by software setting the BPEN bit in the BREAKCR register.

When a low level is detected on the BREAK pin, the BA bit is set and the break function is activated. In this case, the 4 PWM signals are stopped.

Software can set the BA bit to activate the break function without using the BREAK pin.

When the break function is activated (BA bit =1):

– The break pattern (PWM[3:0] bits in the BREAKCR) is forced directly on the PWMx output pins (after the inverter).

– The 12-bit PWM counter CNTR1 is put to its reset value, i.e. 00h.

– The 12-bit PWM counter CNTR2 is put to its reset value,i.e. 00h.

– ATR1, ATR2, Preload and Active DCRx are put to their reset values.

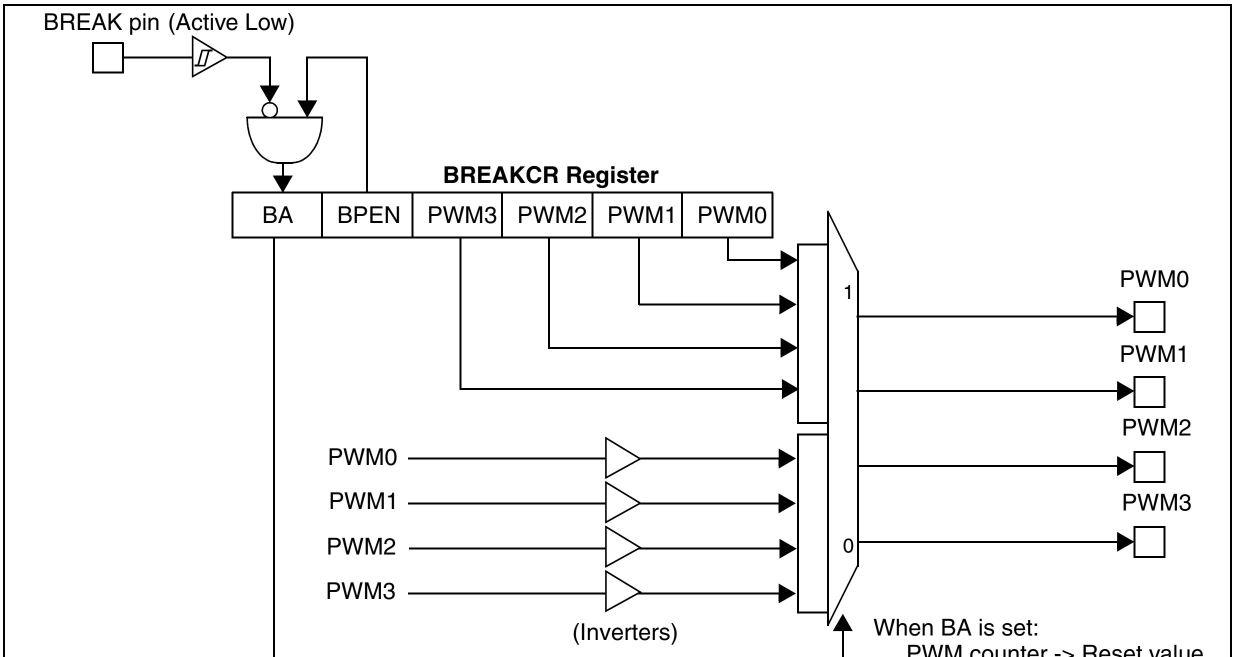
– The PWMCR register is reset.

– Counters stop counting.

When the break function is deactivated after applying the break (BA bit goes from 1 to 0 by software):

– The control of the 4 PWM outputs is transferred to the port registers.

Figure 40. Block Diagram of Break Function



Note:
The BREAK pin value is latched by the BA bit.

DUAL 12-BIT AUTORELOAD TIMER 3 (Cont'd)

11.2.3.2 Output Compare Mode

To use this function, load a 12-bit value in the Preload DCRxH and DCRxL registers.

When the 12-bit upcounter (CNTR1) reaches the value stored in the Active DCRxH and DCRxL registers, the CMPFx bit in the PWMxCSR register is set and an interrupt request is generated if the CMPIE bit is set.

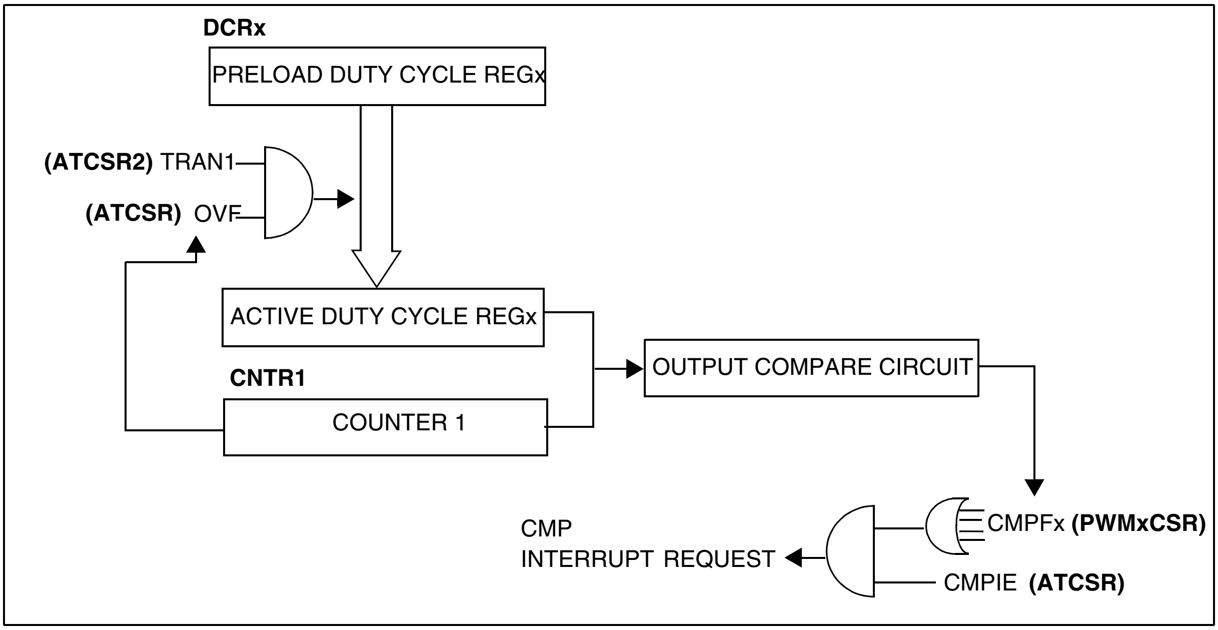
The output compare function is always performed on CNTR1 in both Single Timer mode and Dual Timer mode, and never on CNTR2. The difference is that in Single Timer mode the counter 1 can be compared with any of the four DCR registers, and in Dual Timer mode, counter 1 is compared with DCR0 or DCR1.

Notes:

1. The output compare function is only available for DCRx values other than 0 (reset value).

2. Duty cycle registers are buffered internally. The CPU writes in Preload Duty Cycle Registers and these values are transferred in Active Duty Cycle Registers after an overflow event if the corresponding transfer bit (TRAN1 bit) is set. Output compare is done by comparing these active DCRx values with the counter.

Figure 41. Block Diagram of Output Compare Mode (single timer)



Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
21	**ATCSR2** Reset Value	0	0	ICS 0	OVFIE2 0	OVF2 0	ENCNTR2 0	TRAN2 1	TRAN1 1
22	**BREAKCR** Reset Value	0	0	BA 0	BPEN 0	PWM3 0	PWM2 0	PWM1 0	PWM0 0
23	**ATR2H** Reset Value	0	0	0	0	ATR11 0	ATR10 0	ATR9 0	ATR8 0
24	**ATR2L** Reset Value	ATR7 0	ATR6 0	ATR5 0	ATR4 0	ATR3 0	ATR2 0	ATR1 0	ATR0 0
25	**DTGR** Reset Value	DTE 0	DT6 0	DT5 0	DT4 0	DT3 0	DT2 0	DT1 0	DT0 0

11.3 LITE TIMER 2 (LT2)

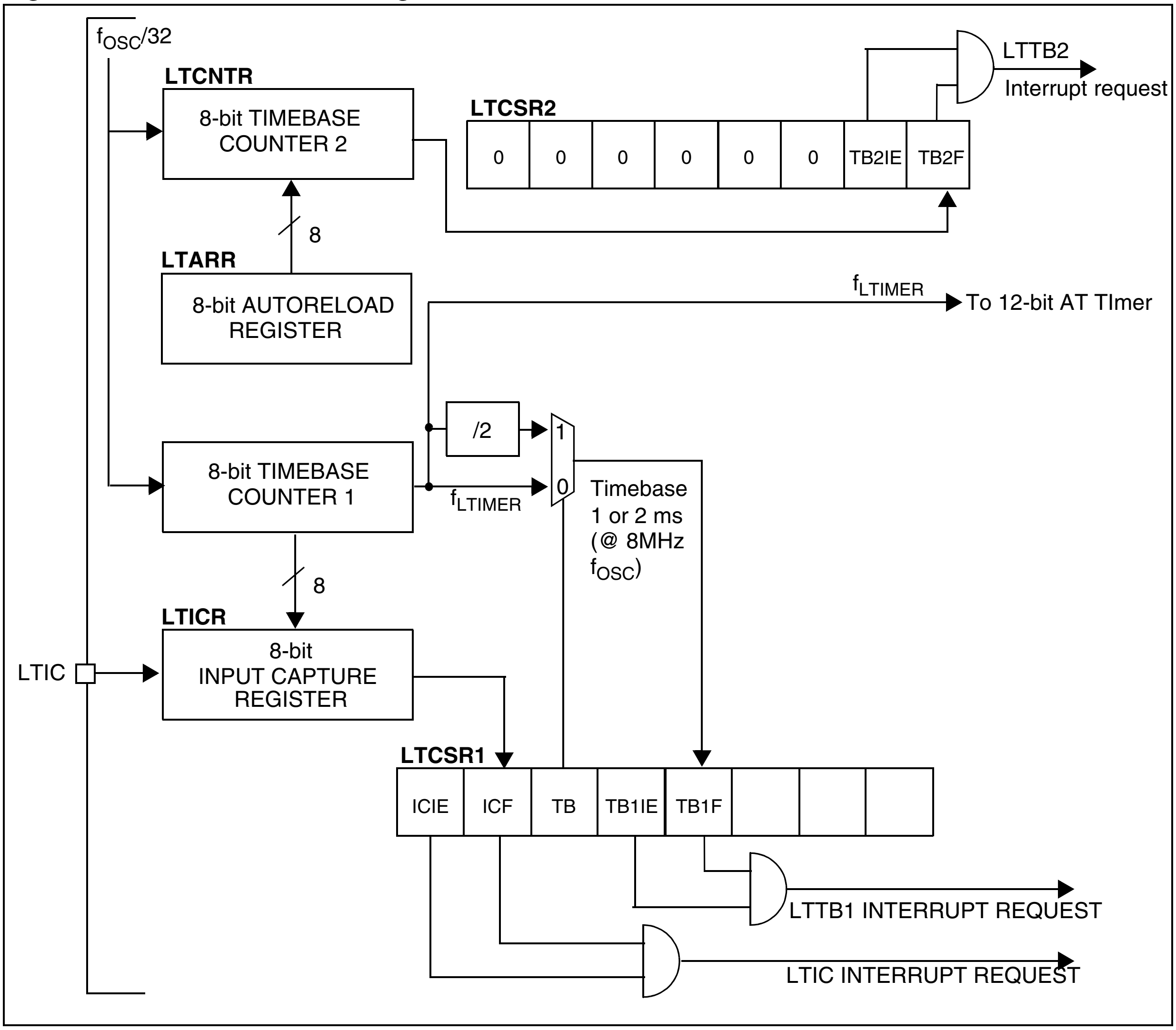
11.3.1 Introduction

The Lite Timer can be used for general-purpose timing functions. It is based on two free-running 8-bit upcounters and an 8-bit input capture register.

11.3.2 Main Features

- Realtime Clock (RTC)
  – One 8-bit upcounter 1 ms or 2 ms timebase period (@ 8 MHz f_{OSC})
  – One 8-bit upcounter with autoreload and programmable timebase period from 4µs to 1.024ms in 4µs increments (@ 8 MHz f_{OSC})
  – 2 Maskable timebase interrupts
- Input Capture
  – 8-bit input capture register (LTICR)
  – Maskable interrupt with wakeup from Halt Mode capability

Figure 46. Lite Timer 2 Block Diagram



11.4.8 Register Description

SPI CONTROL REGISTER (SPICR)

Read/Write

Reset Value: 0000 xxxx (0xh)

7							0
SPIE	SPE	SPR2	MSTR	CPOL	CPHA	SPR1	SPR0

Bit 7 = **SPIE** *Serial Peripheral Interrupt Enable*
This bit is set and cleared by software.
0: Interrupt is inhibited
1: An SPI interrupt is generated whenever an End of Transfer event, Master Mode Fault or Overrun error occurs (SPIF = 1, MODF = 1 or OVR = 1 in the SPICSR register)

Bit 6 = **SPE** *Serial Peripheral Output Enable*
This bit is set and cleared by software. It is also cleared by hardware when, in master mode, $\overline{SS}$ = 0 (see Section 11.4.5.1 Master Mode Fault (MODF)). The SPE bit is cleared by reset, so the SPI peripheral is not initially connected to the external pins.
0: I/O pins free for general purpose I/O
1: SPI I/O pin alternate functions enabled

Bit 5 = **SPR2** *Divider Enable*
This bit is set and cleared by software and is cleared by reset. It is used with the SPR[1:0] bits to set the baud rate. Refer to Table 18 SPI Master Mode SCK Frequency.
0: Divider by 2 enabled
1: Divider by 2 disabled

Note: This bit has no effect in slave mode.

Bit 4 = **MSTR** *Master Mode*
This bit is set and cleared by software. It is also cleared by hardware when, in master mode, $\overline{SS}$ = 0 (see Section 11.4.5.1 Master Mode Fault (MODF)).
0: Slave mode
1: Master mode. The function of the SCK pin changes from an input to an output and the functions of the MISO and MOSI pins are reversed.

Bit 3 = **CPOL** *Clock Polarity*
This bit is set and cleared by software. This bit determines the idle state of the serial Clock. The CPOL bit affects both the master and slave modes.
0: SCK pin has a low level idle state
1: SCK pin has a high level idle state

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

Bit 2 = **CPHA** *Clock Phase*
This bit is set and cleared by software.
0: The first clock transition is the first data capture edge.
1: The second clock transition is the first capture edge.

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

Bits 1:0 = **SPR[1:0]** *Serial Clock Frequency*
These bits are set and cleared by software. Used with the SPR2 bit, they select the baud rate of the SPI serial clock SCK output by the SPI in master mode.

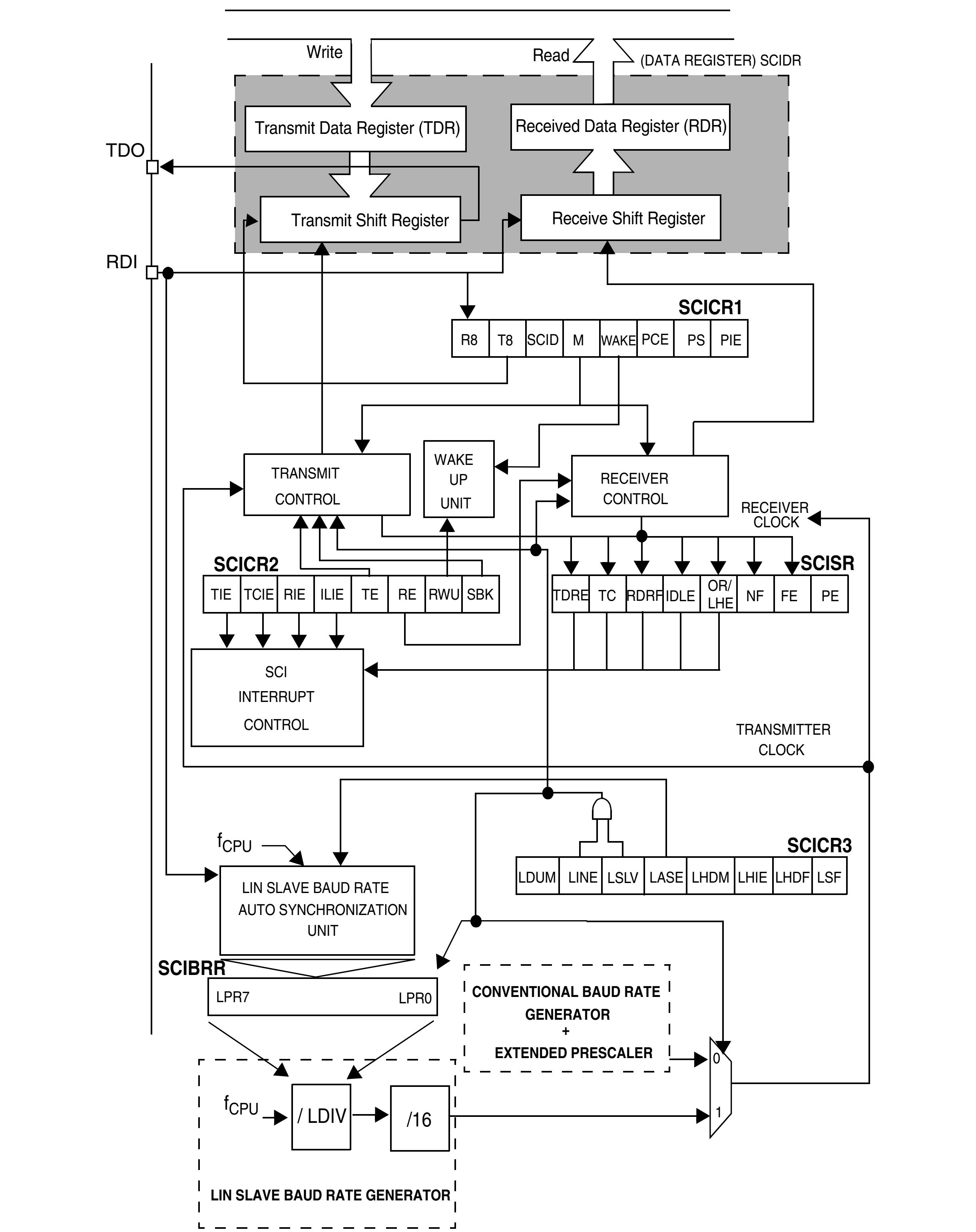
Note: These 2 bits have no effect in slave mode.

Table 18. SPI Master Mode SCK Frequency

<table>
<tr><th>Serial Clock</th><th>SPR2</th><th>SPR1</th><th>SPR0</th></tr>
<tr><td>$f_{CPU}/4$</td><td>1</td><td rowspan="3">0</td><td rowspan="2">0</td></tr>
<tr><td>$f_{CPU}/8$</td><td rowspan="2">0</td></tr>
<tr><td>$f_{CPU}/16$</td><td>1</td></tr>
<tr><td>$f_{CPU}/32$</td><td>1</td><td rowspan="3">1</td><td rowspan="2">0</td></tr>
<tr><td>$f_{CPU}/64$</td><td rowspan="2">0</td></tr>
<tr><td>$f_{CPU}/128$</td><td>1</td></tr>
</table>

LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Mode) (cont'd)

Figure 59. SCI Block Diagram in LIN Slave Mode



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

11.6.3.2 Digital A/D Conversion Result

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

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

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

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

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

11.6.3.3 A/D Conversion

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

In the ADCCSR register:

– Select the CS[2:0] bits to assign the analog channel to convert.

ADC Conversion mode

In the ADCCSR register:

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

When a conversion is complete:

– The EOC bit is set by hardware.

– The result is in the ADCDR registers.

A read to the ADCDRH or a write to any bit of the ADCCSR register resets the EOC bit.

To read the 10 bits, perform the following steps:

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

To read only 8 bits, perform the following steps:

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

11.6.3.4 Changing the conversion channel

The application can change channels during conversion.

When software modifies the CH[2:0] bits in the ADCCSR register, the current conversion is stopped, the EOC bit is cleared, and the A/D converter starts converting the newly selected channel.

11.6.4 Low Power Modes

The A/D converter may be disabled by resetting the ADON bit. This feature allows reduced power consumption when no conversion is needed and between single shot conversions.

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

11.6.5 Interrupts

None.

Table 22. ADC Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0034h	**ADCCSR** Reset Value	EOC 0	SPEED 0	ADON 0	0 0	0 0	CH2 0	CH1 0	CH0 0
0035h	**ADCDRH** Reset Value	D9 0	D8 0	D7 0	D6 0	D5 0	D4 0	D3 0	D2 0
0036h	**ADCDRL** Reset Value	0 0	0 0	0 0	0 0	SLOW 0	0 0	D1 0	D0 0

ST7 ADDRESSING MODES (cont'd)

12.1.6 Indirect Indexed (Short, Long)

This is a combination of indirect and short indexed addressing modes. The operand is referenced by its memory address, which is defined by the unsigned addition of an index register value (X or Y) with a pointer value located in memory. The pointer address follows the opcode.

The indirect indexed addressing mode consists of two submodes:

Indirect Indexed (Short)

The pointer address is a byte, the pointer size is a byte, thus allowing 00 - 1FE addressing space, and requires 1 byte after the opcode.

Indirect Indexed (Long)

The pointer address is a byte, the pointer size is a word, thus allowing 64 Kbyte addressing space, and requires 1 byte after the opcode.

Table 24. Instructions Supporting Direct, Indexed, Indirect and Indirect Indexed Addressing Modes

Long and Short Instructions	Function
LD	Load
CP	Compare
AND, OR, XOR	Logical Operations
ADC, ADD, SUB, SBC	Arithmetic Addition/subtraction operations
BCP	Bit Compare

Short Instructions Only	Function
CLR	Clear
INC, DEC	Increment/Decrement
TNZ	Test Negative or Zero
CPL, NEG	1 or 2 Complement
BSET, BRES	Bit Operations
BTJT, BTJF	Bit Test and Jump Operations
SLL, SRL, SRA, RLC, RRC	Shift and Rotate Operations
SWAP	Swap Nibbles
CALL, JP	Call or Jump subroutine

12.1.7 Relative Mode (Direct, Indirect)

This addressing mode is used to modify the PC register value by adding an 8-bit signed offset to it.

Available Relative Direct/ Indirect Instructions	Function
JRxx	Conditional Jump
CALLR	Call Relative

The relative addressing mode consists of two submodes:

Relative (Direct)

The offset follows the opcode.

Relative (Indirect)

The offset is defined in memory, of which the address follows the opcode.

13.7 EMC (ELECTROMAGNETIC COMPATIBILITY) CHARACTERISTICS

Susceptibility tests are performed on a sample basis during product characterization.

13.7.1 Functional EMS (Electro Magnetic Susceptibility)

Based on a simple running application on the product (toggling two LEDs through I/O ports), the product is stressed by two electro magnetic events until a failure occurs (indicated by the LEDs).

- **ESD**: Electrostatic Discharge (positive and negative) is applied on all pins of the device until a functional disturbance occurs. This test conforms with the IEC 1000-4-2 standard.
- **FTB**: A Burst of Fast Transient voltage (positive and negative) is applied to V_{DD} and V_{SS} through a 100pF capacitor, until a functional disturbance occurs. This test conforms with the IEC 1000-4-4 standard.

A device reset allows normal operations to be resumed. The test results are given in the table below based on the EMS levels and classes defined in application note AN1709.

13.7.1.1 Designing hardened software to avoid noise problems

EMC characterization and optimization are performed at component level with a typical application environment and simplified MCU software. It should be noted that good EMC performance is highly dependent on the user application and the software in particular.

Therefore it is recommended that the user applies EMC software optimization and prequalification tests in relation with the EMC level requested for his application.

Software recommendations:

The software flowchart must include the management of runaway conditions such as:

– Corrupted program counter

– Unexpected reset

– Critical Data corruption (control registers...)

Prequalification trials:

Most of the common failures (unexpected reset and program counter corruption) can be reproduced by manually forcing a low state on the RESET pin or the Oscillator pins for 1 second.

To complete these trials, ESD stress can be applied directly on the device, over the range of specification values. When unexpected behaviour is detected, the software can be hardened to prevent unrecoverable errors occurring (see application note AN1015).

Table 25: EMS test results

Symbol	Parameter	Conditions	Level/ Class
V_{FESD}	Voltage limits to be applied on any I/O pin to induce a functional disturbance	V_{DD}=5V, T_A=+25°C, f_{OSC}=8MHz conforms to IEC 1000-4-2	3B
V_{FFTB}	Fast transient voltage burst limits to be applied through 100pF on V_{DD} and V_{DD} pins to induce a functional disturbance	V_{DD}=5V, T_A=+25°C, f_{OSC}=8MHz conforms to IEC 1000-4-4	3B

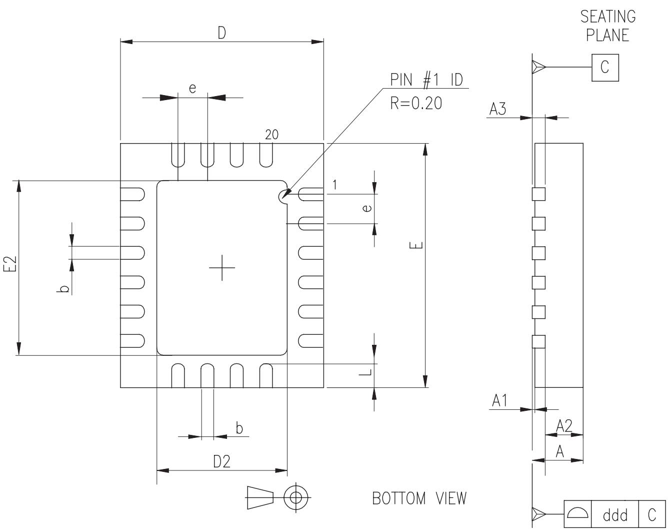
14 PACKAGE CHARACTERISTICS

14.1 PACKAGE MECHANICAL DATA

In order to meet environmental requirements, ST offers these devices in ECOPACK® packages. These packages have a Lead-free second level interconnect. The category of second Level Interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label.

ECOPACK is an ST trademark. ECOPACK specifications are available at: www.st.com.

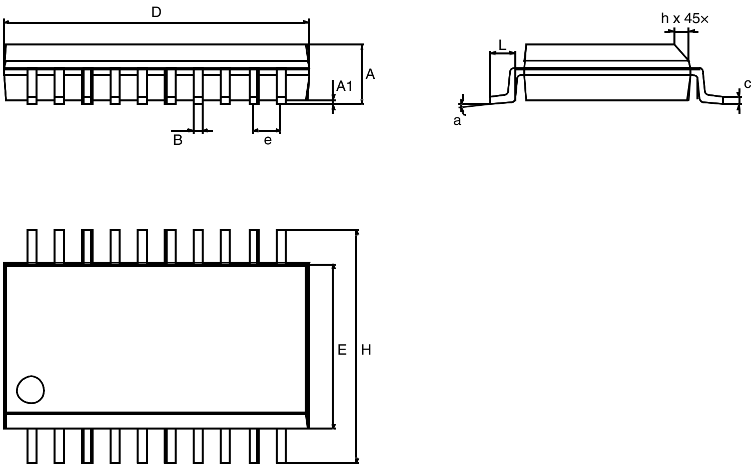
Figure 105. 20-Lead Very thin Fine pitch Quad Flat No-Lead Package



Dim.	mm			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	0.80	0.85	0.90	0.0315	0.0335	0.0354
A1	0.00	0.02	0.05		0.0008	0.0020
A3		0.02			0.0008	
b	0.25	0.30	0.35	0.0098	0.0118	0.0138
D		5.00			0.1969	
D2	3.10	3.25	3.35	0.1220	0.1280	0.1319
E		6.00			0.2362	
E2	4.10	4.25	4.35	0.1614	0.1673	0.1713
e		0.80			0.0315	
L	0.45	0.50	0.55	0.0177	0.0197	0.0217
ddd		0.08			0.0031	
	Number of Pins					
N	20					

Note 1. Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 106. 20-Pin Plastic Small Outline Package, 300-mil Width



Dim.	mm			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	2.35		2.65	0.0925		0.1043
A1	0.10		0.30	0.0039		0.0118
B	0.33		0.51	0.0130		0.0201
C	0.23		0.32	0.0091		0.0126
D	12.60		13.00	0.4961		0.5118
E	7.40		7.60	0.2913		0.2992
e		1.27			0.0500	
H	10.00		10.65	0.3937		0.4193
h	0.25		0.75	0.0098		0.0295
α	0°		8°	0°		8°
L	0.40		1.27	0.0157		0.0500
	Number of Pins					
N	20					

Note 1. Values in inches are converted from mm and rounded to 4 decimal digits.

ST7LITE3xF2 FASTROM MICROCONTROLLER OPTION LIST
(Last update: November 2007)

Customer . . . .
Address . . . .
. . . .
Contact . . . .
Phone No . . . .
Reference FASTROM Code*: . . . .
*FASTROM code name is assigned by STMicroelectronics.
FASTROM code must be sent in .S19 format. .Hex extension cannot be processed.

Device type: [ ] ST7PLITE30F2 [ ] ST7PLITE35F2 [ ] ST7PLITE39F2

Conditioning (check only one option):

PDIP20:	[ ] Tube	
SO20:	[ ] Tape & Reel	[ ] Tube
QFN20:	[ ] Tape & Reel	[ ] Tray

Special Marking: [ ] No [ ] Yes "_ _ _ _ _ _ _ _ "
Authorized characters are letters, digits, '.', '-', '/' and spaces only.
Maximum character count: 8 char. max

Temperature range [ ] - 40°C to + 85°C [ ] - 40°C to + 125°C
AWUCK Selection [ ] 32-kHz Oscillator [ ] AWU RC Oscillator
Clock Source Selection: [ ] Resonator:
- [ ] VLP: Very Low power resonator (32 to 100 kHz)
- [ ] LP: Low power resonator (1 to 2 MHz)
- [ ] MP: Medium power resonator (2 to 4 MHz)
- [ ] MS: Medium speed resonator (4 to 8 MHz)
- [ ] HS: High speed resonator (8 to 16 MHz)

[ ] External Clock
- [ ] on PB4
- [ ] on OSC1

[ ] Internal RC Oscillator

Sector 0 size:	[ ] 0.5K	[ ] 1K	[ ] 2K	[ ] 4K
Readout Protection:	[ ] Disabled	[ ] Enabled		
FLASH Write Protection	[ ] Disabled	[ ] Enabled		
PLL	[ ] Disabled	[ ] PLLx4	[ ] PLLx8	
LVD Reset	[ ] Disabled	[ ] Highest threshold [ ] Medium threshold [ ] Lowest threshold		

Watchdog Selection: [ ] Software Activation [ ] Hardware Activation

Watchdog Reset on Halt: [ ] Disabled [ ] Enabled

Comments : . . . .
Supply Operating Range in the application: . . . .
Notes . . . .
Date: . . . .
Signature: . . . .

Important note: Not all configurations are available.
Refer to Figure 108.Ordering information scheme.
Please contact the ST Sales Office nearest to you for any further information.

Please download the latest version of this option list from:
www.st.com