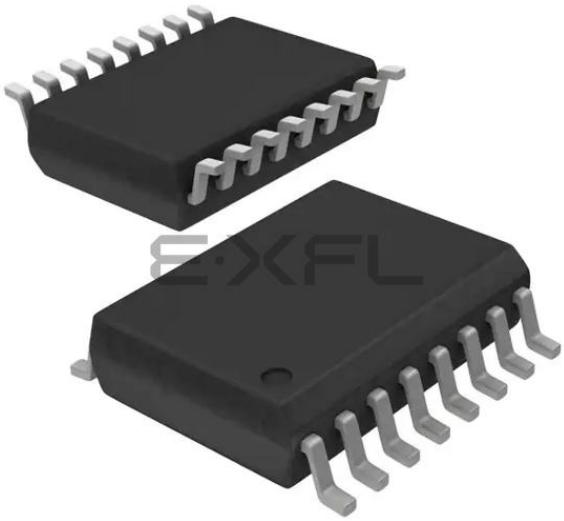


Welcome to E-XFL.COM

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	8MHz
Connectivity	SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	11
Program Memory Size	4KB (4K 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 5x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st7flit10by1m6

Table of Contents

7.6 SYSTEM INTEGRITY MANAGEMENT (SI)

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

Note: A reset can also be triggered following the detection of an illegal opcode or prebyte code. Refer to section 12.2.1 on page 107 for further details.

7.6.1 Low Voltage Detector (LVD)

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

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

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

– $V_{IT+(LVD)}$when V_{DD} is rising

– $V_{IT-(LVD)}$ when V_{DD} is falling

The LVD function is illustrated in Figure 18.

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

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

– under full software control

– in static safe reset

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

During a Low Voltage Detector Reset, the $\overline{\text{RESET}}$ pin is held low, thus permitting the MCU to reset other devices.

Notes:

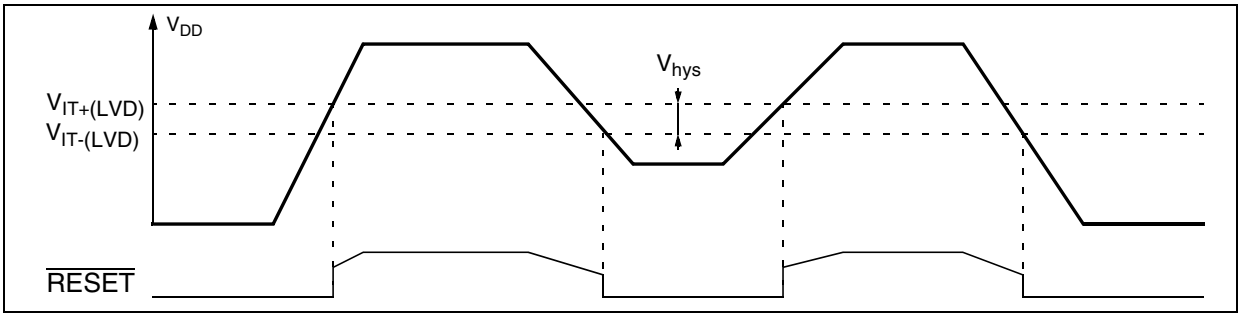
The LVD allows the device to be used without any external RESET circuitry.

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

Use of LVD with capacitive power supply: with this type of power supply, if power cuts occur in the application, it is recommended to pull V_{DD} down to 0V to ensure optimum restart conditions. Refer to circuit example in Figure 106 on page 136 and note 4.

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

Figure 18. Low Voltage Detector vs Reset



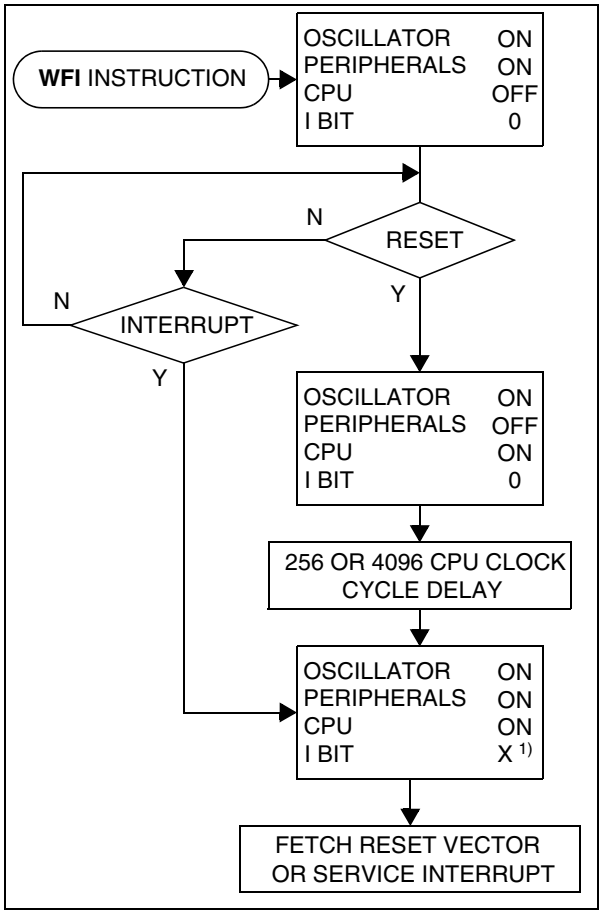
POWER SAVING MODES (Cont'd)

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 24.

Figure 24. 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)

9.4.1 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, re-initialize the corresponding I/O as "Input Pull-up with Interrupt" before executing the HALT instruction. The main reason for this is that the I/O may be wrongly configured due to external interference or by an unforeseen logical condition.

– For the same reason, re-initialize 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 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 OVFIE bit	ATCSR CK1 bit	ATCSR CK0 bit	Meaning
0	x	x	0	ACTIVE-HALT mode disabled
0	0	x	x	
1	x	x	x	ACTIVE-HALT mode enabled
x	1	0	1	

The MCU can exit ACTIVE-HALT mode on reception of a specific interrupt (see Table 5, "Interrupt Mapping," on page 37) or a RESET.

– When exiting ACTIVE-HALT mode by means of a RESET, a 256 or 4096 CPU cycle delay occurs. After the start up delay, the CPU resumes operation by fetching the reset vector which woke it up (see Figure 28).

– 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 28).

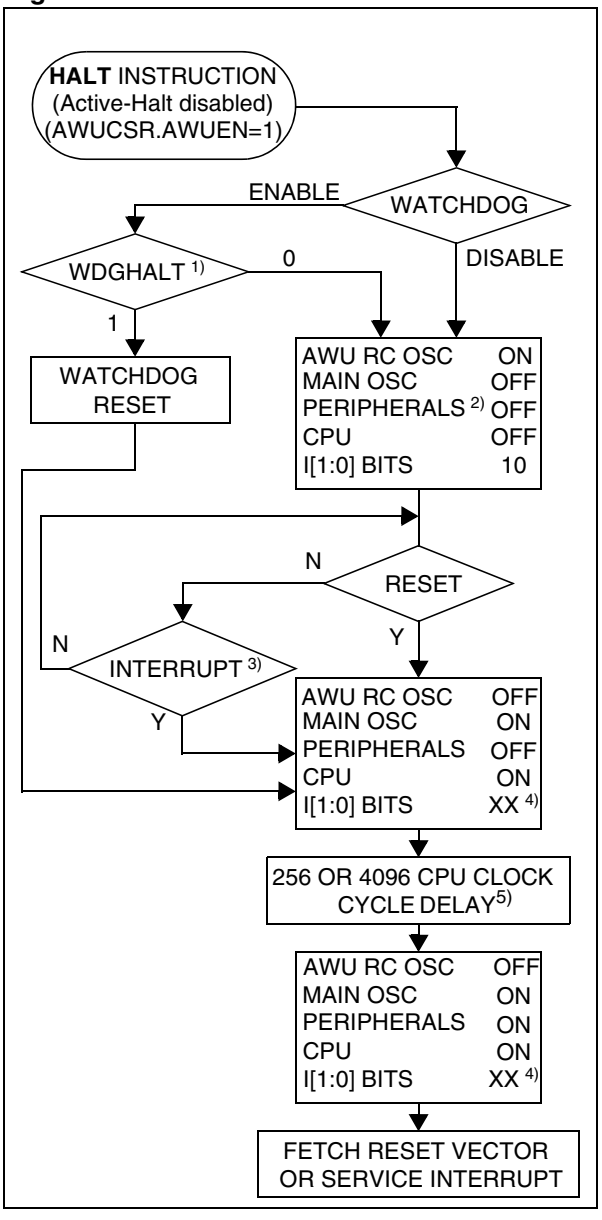
When entering ACTIVE-HALT mode, the I bit in the CC register is cleared to enable interrupts. Therefore, if an interrupt is pending, the MCU wakes up immediately (see Note 3).

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.

POWER SAVING MODES (Cont'd)

Figure 31. AWUFH 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 an AWUFH interrupt and some specific interrupts can exit the MCU from HALT mode (such as external interrupt). Refer to Table 5, "Interrupt Mapping," on page 37 for more details.

4. Before servicing an interrupt, the CC register is pushed on the stack. The I[1:0] bits of the CC register are set to the current software priority level of the interrupt routine and recovered when the CC register is popped.

5. If the PLL is enabled by option byte, it outputs the clock after an additional delay of $t_{STARTUP}$ (see Figure 13).

I/O PORTS (Cont'd)

Figure 32. I/O Port General Block Diagram

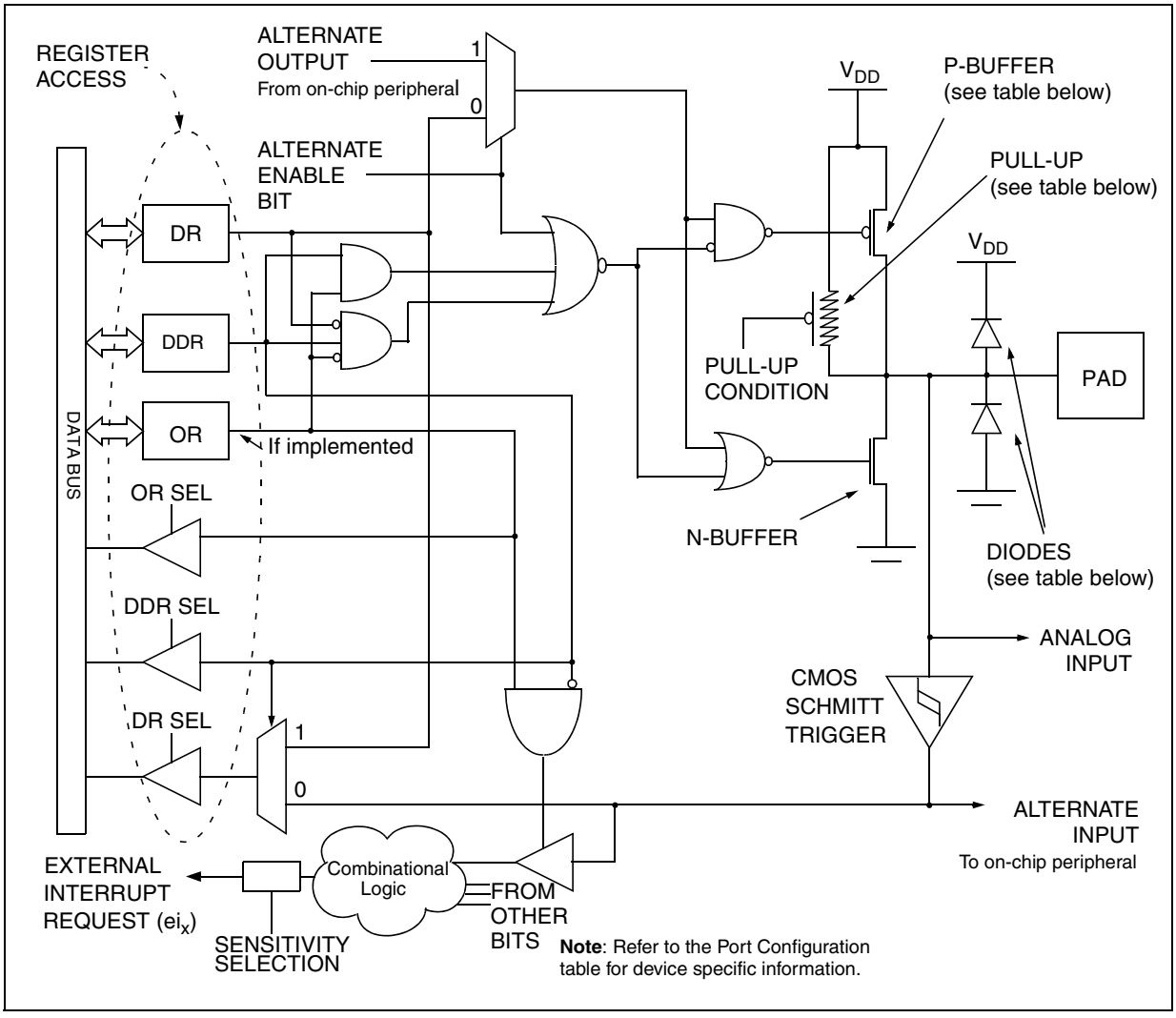


Table 8. I/O Port Mode Options

Configuration Mode		Pull-Up	P-Buffer	Diodes	
				to V_{DD}	to V_{SS}
Input	Floating with/without Interrupt	Off	Off	On	On
	Pull-up with/without Interrupt	On			
Output	Push-pull	Off	On		
	Open Drain (logic level)		Off		

Legend: Off - implemented not activated
On - implemented and activated

11.2 DUAL 12-BIT AUTORELOAD TIMER 4 (AT4)

11.2.1 Introduction

The 12-bit Autoreload Timer can be used for general-purpose timing functions. It is based on one or two free-running 12-bit upcounters with an input capture register and four PWM output channels. There are 7 external pins:

– Four PWM outputs

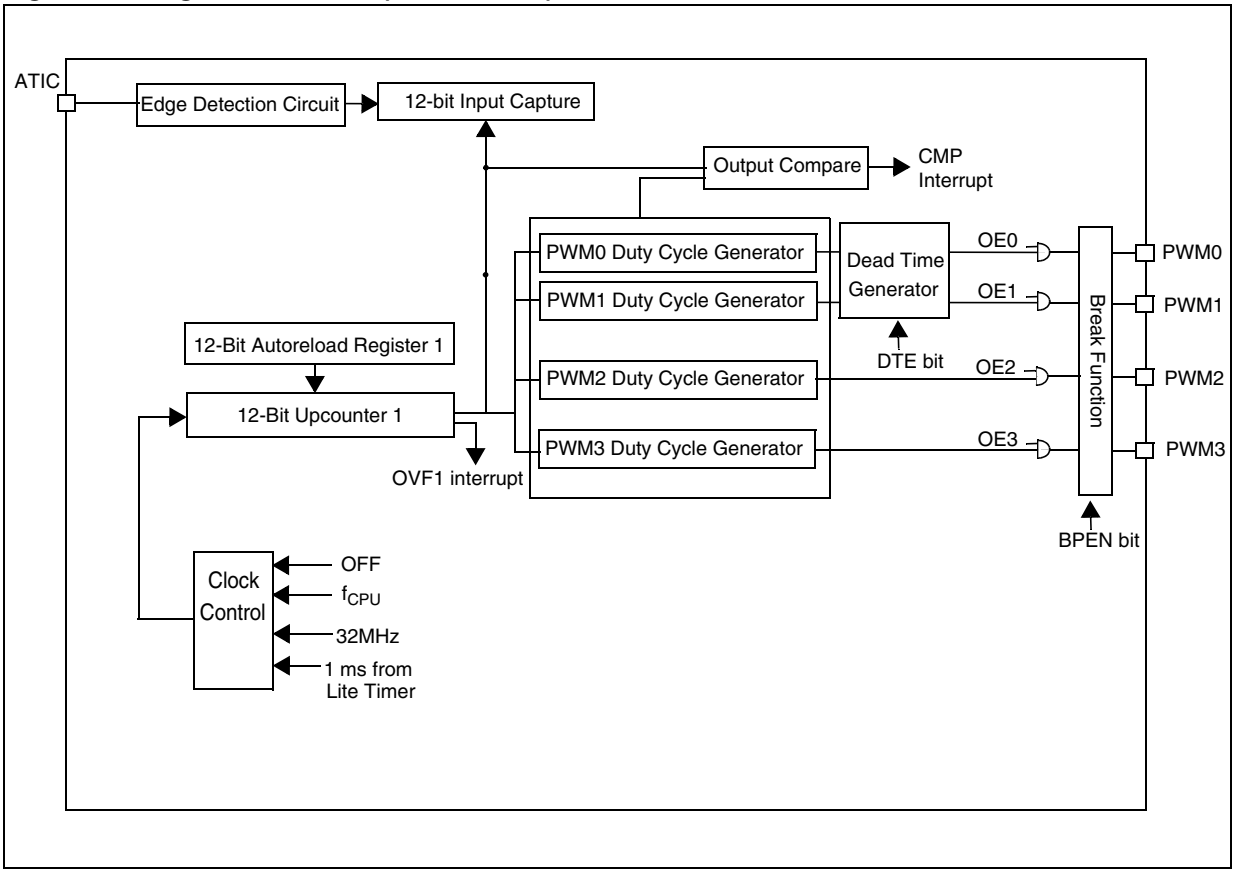
– ATIC/LTIC pins for the Input Capture function

– BREAK pin for forcing a break condition on the PWM outputs

11.2.2 Main Features

- Single Timer or Dual Timer mode with two 12-bit upcounters (CNTR1/CNTR2) and two 12-bit autoreload registers (ATR1/ATR2)
- Maskable overflow interrupts
- PWM mode
  - Generation of four independent PWMx signals
  - Dead time generation for Half bridge driving mode with programmable dead time
  - Frequency 2 kHz - 4 MHz (@ 8 MHz f_{CPU})
  - Programmable duty-cycles
  - Polarity control
  - Programmable output modes
- Output Compare Mode
- Input Capture Mode
  - 12-bit input capture register (ATICR)
  - Triggered by rising and falling edges
  - Maskable IC interrupt
  - Long range input capture
- Internal/External Break control
- Flexible Clock control
- One Pulse mode on PWM2/3
- Force Update

Figure 35. Single Timer Mode (ENCNTR2=0)



DUAL 12-BIT AUTORELOAD TIMER 4 (Cont'd)

11.2.3 Functional Description

11.2.3.1 PWM Mode

This mode allows up to four Pulse Width Modulated signals to be generated on the PWMx output pins.

PWM Frequency

The four PWM signals can have the same frequency (f_{PWM}) or can have two different frequencies. This is selected by the ENCNTR2 bit which enables single timer or dual timer mode (see Figure 1 and Figure 2).

The frequency is controlled by the counter period and the ATR register value. In dual timer mode, PWM2 and PWM3 can be generated with a different frequency controlled by CNTR2 and ATR2.

$$f_{PWM} = f_{COUNTER} / (4096 - ATR)$$

Following the above formula,

– If $f_{COUNTER}$ is 4 MHz, the maximum value of f_{PWM} is 2 MHz (ATR register value = 4094), the minimum value is 1 kHz (ATR register value = 0).

– If $f_{COUNTER}$ is 32 MHz, the maximum value of f_{PWM} is 8 MHz (ATR register value = 4092), the minimum value is 8 kHz (ATR register value = 0).

Notes:

1. The maximum value of ATR is 4094 because it must be lower than the DC4R value which must be 4095 in this case.

2. To update the DCRx registers at 32 MHz, the following precautions must be taken:

– if the PWM frequency is < 1 MHz and the TRANx bit is set asynchronously, it should be set twice after a write to the DCRx registers.

– if the PWM frequency is > 1 MHz, the TRANx bit should be set along with FORCEx bit with the same instruction (use a load instruction and not 2 bset instructions).

Duty Cycle

The duty cycle is selected by programming the DCRx registers. These are preload registers. The DCRx values are transferred in Active duty cycle registers after an overflow event if the corresponding transfer bit (TRANx bit) is set.

The TRAN1 bit controls the PWMx outputs driven by counter 1 and the TRAN2 bit controls the PWMx outputs driven by counter 2.

PWM generation and output compare are done by comparing these active DCRx values with the counter.

The maximum available resolution for the PWMx duty cycle is:

$$Resolution = 1 / (4096 - ATR)$$

where ATR is equal to 0. With this maximum resolution, 0% and 100% duty cycle can be obtained by changing the polarity.

At reset, the counter starts counting from 0.

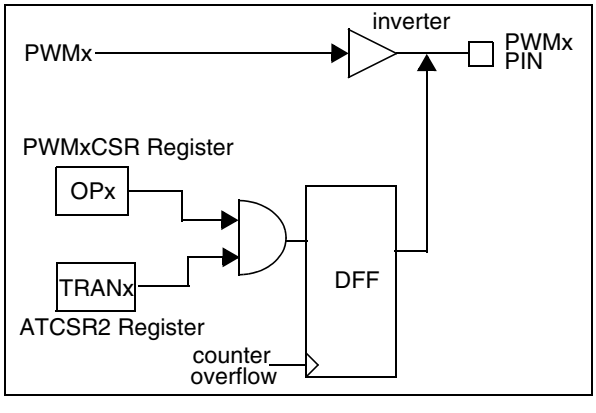
When a upcounter overflow occurs (OVF event), the preloaded Duty cycle values are transferred to the active Duty Cycle registers and the PWMx signals are set to a high level. When the upcounter matches the active DCRx value the PWMx signals are set to a low level. To obtain a signal on a PWMx pin, the contents of the corresponding active DCRx register must be greater than the contents of the ATR register.

The maximum value of ATR is 4094 because it must be lower than the DCR value which must be 4095 in this case.

Polarity Inversion

The polarity bits can be used to invert any of the four output signals. The inversion is synchronized with the counter overflow if the corresponding transfer bit in the ATCSR2 register is set (reset value). See Figure 3.

Figure 37. PWM Polarity Inversion



The Data Flip Flop (DFF) applies the polarity inversion when triggered by the counter overflow input.

Output Control

The PWMx output signals can be enabled or disabled using the OEx bits in the PWMCR register.

DUAL 12-BIT AUTORELOAD TIMER 4 (Cont'd)

8. Set the OP_EN bit in the PWM3CSR register to enable one-pulse mode.

9. Enable the PWM3 output by setting the OE3 bit in the PWMCR register.

The "Wait for Overflow event" in step 6 can be replaced by forced update (writing the FORCE2 bit).

Follow the same procedure for PWM2 with the bits corresponding to PWM2.

Note: When break is applied in one-pulse mode, the CNTR2, DCR2/3 & ATR2 registers are reset. Consequently, these registers have to be initialized again when break is removed.

Figure 47. Block Diagram of One Pulse Mode

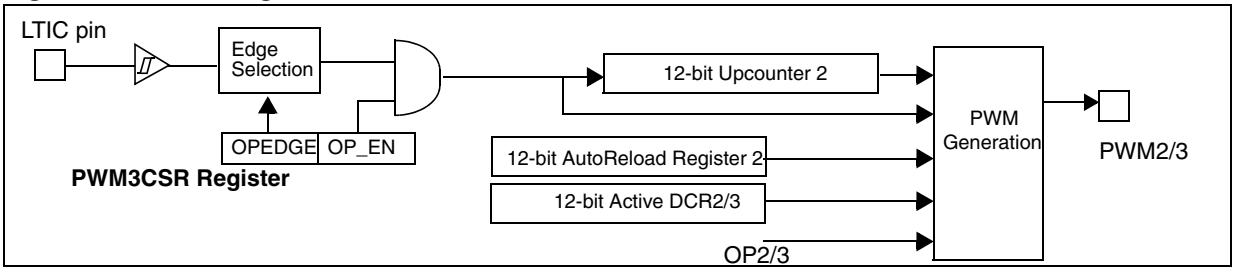
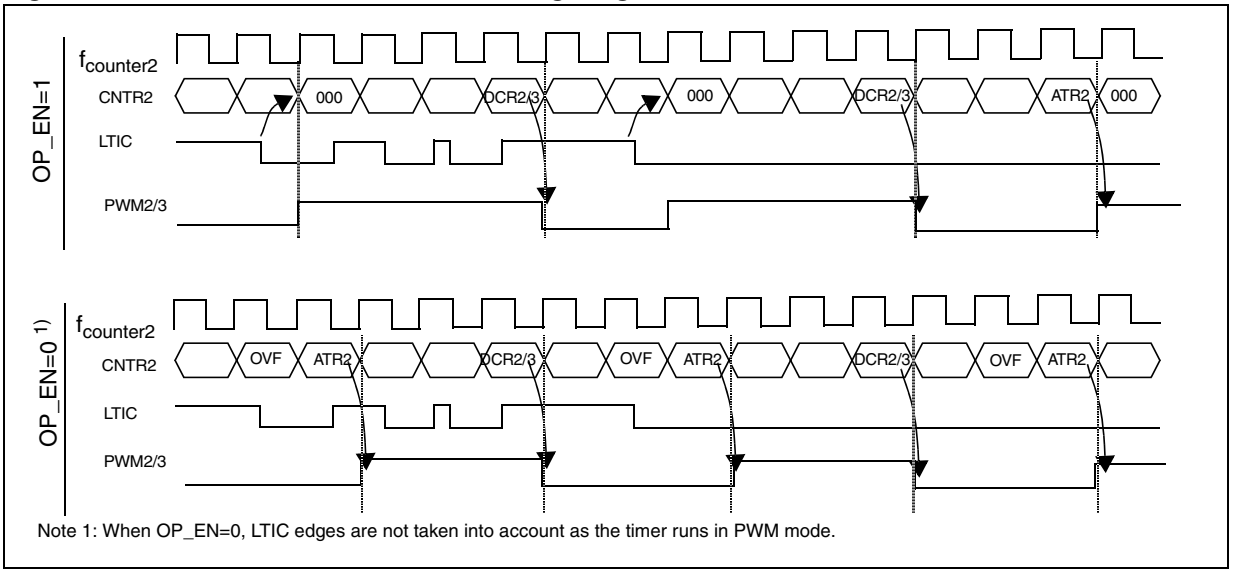
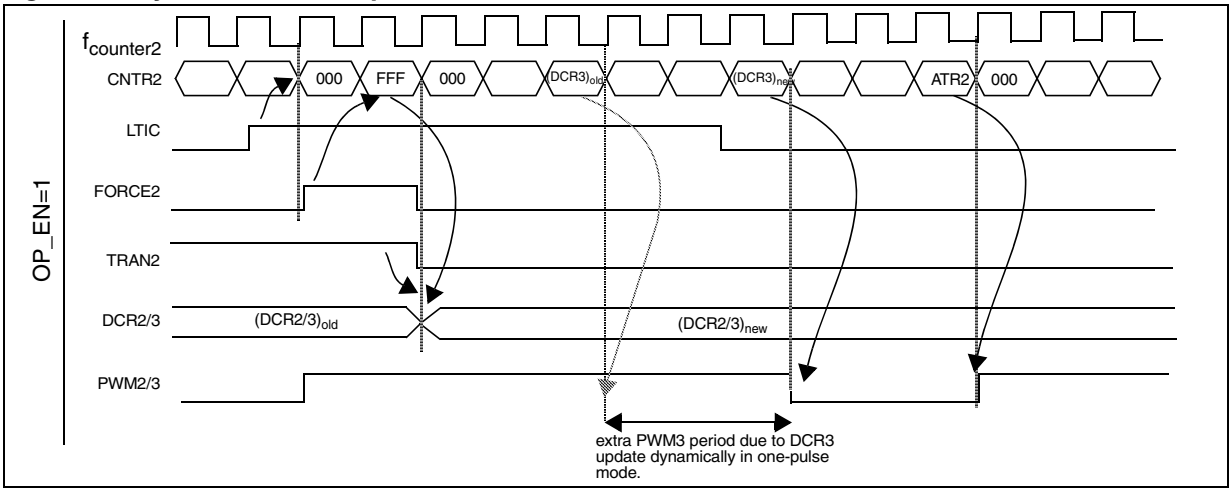


Figure 48. One Pulse Mode and PWM Timing Diagram



Note 1: When OP_EN=0, LTIC edges are not taken into account as the timer runs in PWM mode.

Figure 49. Dynamic DCR2/3 update in One Pulse Mode



DUAL 12-BIT AUTORELOAD TIMER 4 (Cont'd)

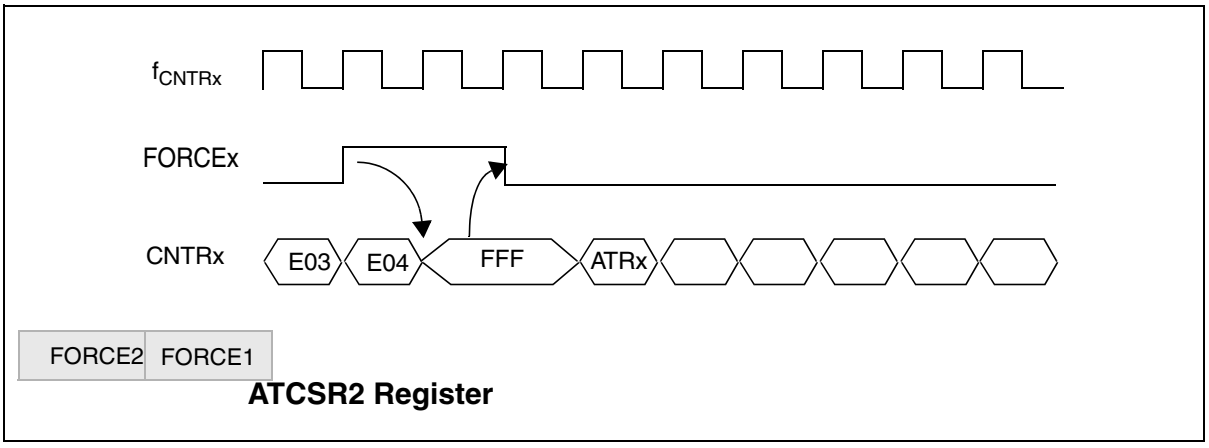
11.2.3.7 Force Update

In order not to wait for the counter$_x$ overflow to load the value into active DCRx registers, a programmable counter$_x$ overflow is provided. For both counters, a separate bit is provided which when set, make the counters start with the overflow value, i.e. FFFh. After overflow, the counters start counting from their respective auto reload register values.

These bits are FORCE1 and FORCE2 in the ATCSR2 register. FORCE1 is used to force an overflow on Counter 1 and, FORCE2 is used for Counter 2. These bits are set by software and reset by hardware after the respective counter overflow event has occurred.

This feature can be used at any time. All related features such as PWM generation, Output Compare, Input Capture, One-pulse (refer to Figure 15. Dynamic DCR2/3 update in One Pulse Mode) can be used this way.

Figure 50. Force Overflow Timing 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 2.

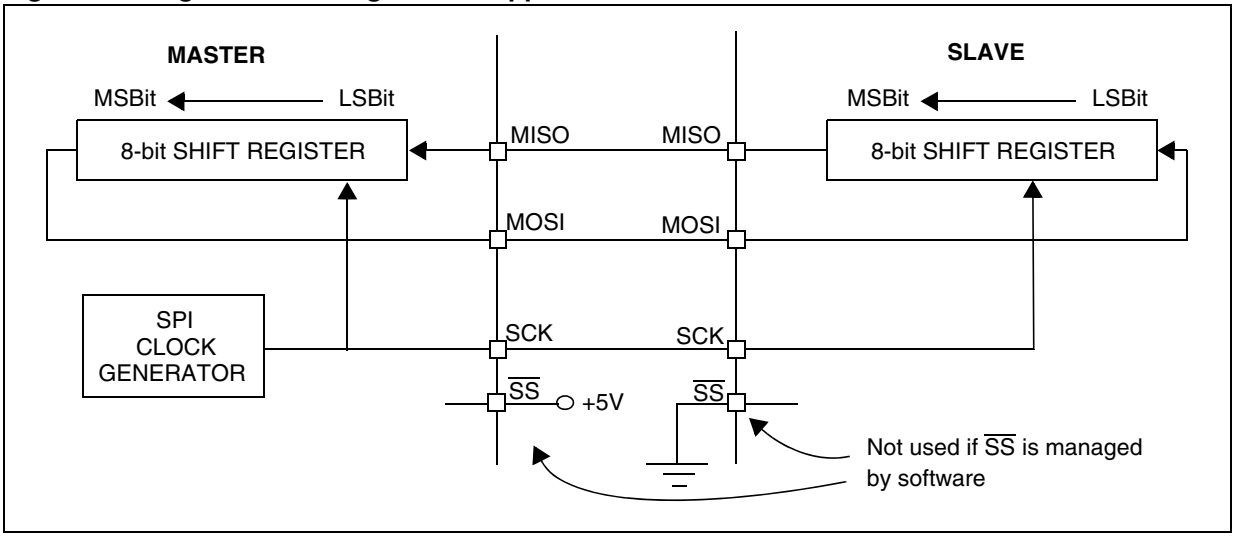
The MOSI pins are connected together and the MISO pins are connected together. In this way data is transferred serially between master and slave (most significant bit first).

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 5 on page 7) but master and slave must be programmed with the same timing mode.

Figure 54. Single Master/ Single Slave Application



SERIAL PERIPHERAL INTERFACE (cont'd)

11.4.4 Clock Phase and Clock Polarity

Four possible timing relationships may be chosen by software, using the CPOL and CPHA bits (See Figure 5).

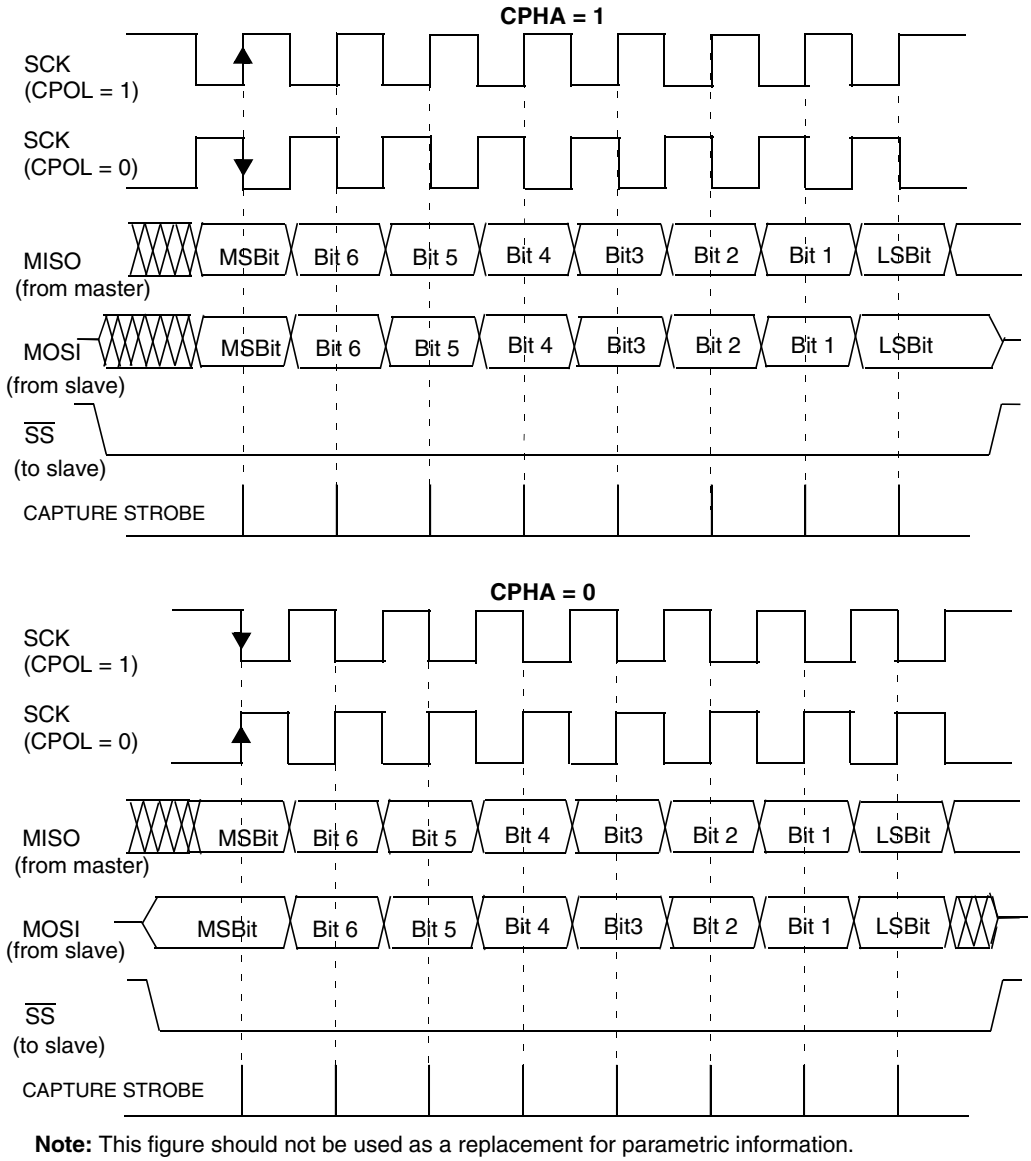
Note: The idle state of SCK must correspond to the polarity selected in the SPICSR register (by pulling up SCK if CPOL = 1 or pulling down SCK if CPOL = 0).

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

Figure 5 shows an SPI transfer with the four combinations of the CPHA and CPOL bits. The diagram may be interpreted as a master or slave timing diagram where the SCK pin, the MISO pin and the MOSI pin are directly connected between the master and the slave device.

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

Figure 57. Data Clock Timing Diagram



Note: This figure should not be used as a replacement for parametric information. Refer to the Electrical Characteristics chapter.

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

11.5.6 Register Description

CONTROL/STATUS REGISTER (ADCCSR)

Read/Write (Except bit 7 read only)

Reset Value: 0000 0000 (00h)

7							0
EOC	SPEED	ADON	0	0	CH2	CH1	CH0

Bit 7 = **EOC** *End of Conversion*
This bit is set by hardware. It is cleared by hardware when software reads the ADCDRH register or writes to any bit of the ADCCSR register.
0: Conversion is not complete
1: Conversion complete

Bit 6 = **SPEED** *ADC clock selection*
This bit is set and cleared by software. It is used together with the SLOW bit to configure the ADC clock speed. Refer to the table in the SLOW bit description (ADCDRL register).

Bit 5 = **ADON** *A/D Converter on*
This bit is set and cleared by software.
0: A/D converter and amplifier are switched off
1: A/D converter and amplifier are switched on

Bits 4:3 = **Reserved.** Must be kept cleared.

Bits 2:0 = **CH[2:0]** *Channel Selection*
These bits are set and cleared by software. They select the analog input to convert.

Channel Pin*	CH2	CH1	CH0
AIN0	0	0	0
AIN1	0	0	1
AIN2	0	1	0
AIN3	0	1	1
AIN4	1	0	0
AIN5	1	0	1
AIN6	1	1	0

*The number of channels is device dependent. Refer to the device pinout description.

DATA REGISTER HIGH (ADCDRH)

Read Only

Reset Value: xxxx xxxx (xxh)

7							0
D9	D8	D7	D6	D5	D4	D3	D2

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

AMP CONTROL/DATA REGISTER LOW (ADCDRL)

Read/Write

Reset Value: 0000 00xx (0xh)

7							0
0	0	0	AMP CAL	SLOW	AMP-SEL	D1	D0

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

Bit 4 = **AMPCAL** *Amplifier Calibration Bit*
This bit is set and cleared by software. It is advised to use this bit to calibrate the ADC when amplifier is ON. Setting this bit internally connects amplifier input to 0V. Hence, corresponding ADC output can be used in software to eliminate amplifier-offset error.
0: Calibration off
1: Calibration on. (The input voltage of the amplifier is set to 0V)

Bit 3 = **SLOW** *Slow mode*
This bit is set and cleared by software. It is used together with the SPEED bit in the ADCCSR register to configure the ADC clock speed as shown on the table below.

f_{ADC}	SLOW	SPEED
$f_{CPU}/2$	0	0
f_{CPU}	0	1
$f_{CPU}/4$	1	x

Note: max f_{ADC} allowed = 4MHz (see section 13.11 on page 139)

Figure 66. Typical accuracy with RCCR=RCCR0 vs V_{DD}= 4.5-5.5V and Temperature

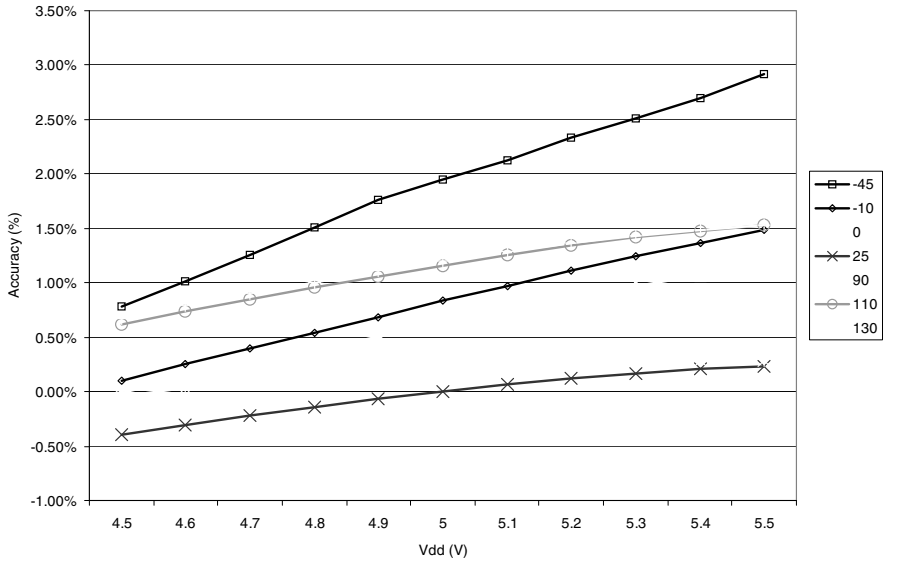
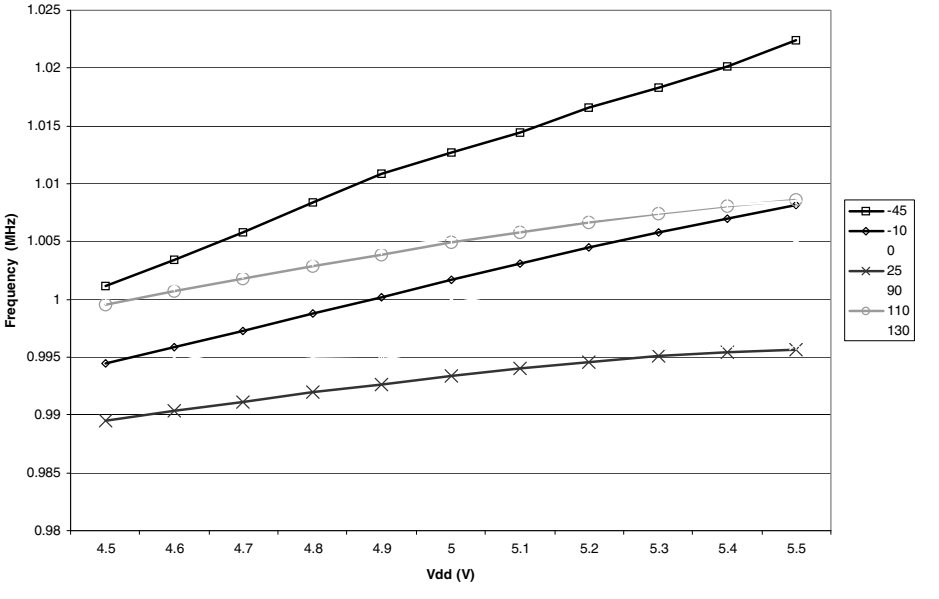


Figure 67. Typical RCCR0 vs V_{DD} and Temperature



13.4 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).

13.4.1 Supply Current

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

Symbol	Parameter	Conditions		Typ	Max	Unit
I_{DD}	Supply current in RUN mode	V_{DD}=5.5V	f_{CPU}=8MHz 1)	7	9	mA
	Supply current in WAIT mode		f_{CPU}=8MHz 2)	3	3.6	
	Supply current in SLOW mode		f_{CPU}=250kHz 3)	0.7	0.9	
	Supply current in SLOW WAIT mode7		f_{CPU}=250kHz 4)	0.5	0.8	
	Supply current in HALT mode5)		-40°C≤T_A≤+125°C	<1	6	µA
	Supply current in AWUFH mode 6)7)		-40°C≤T_A≤+125°C	20		

Notes:

1. CPU running with memory access, 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 (CLKIN) driven by external square wave, LVD disabled.

2. 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 (CLKIN) driven by external square wave, LVD disabled.

3. SLOW mode selected with f_{CPU} based on f_{OSC} divided by 32. 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 (CLKIN) driven by external square wave, LVD disabled.

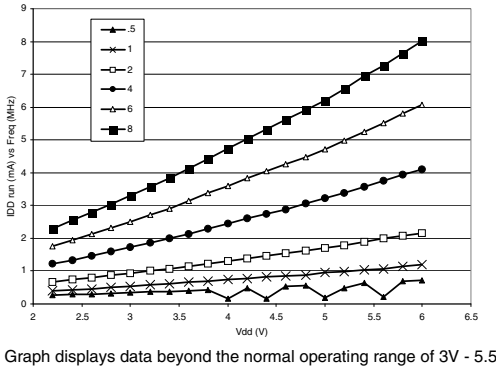
4. SLOW-WAIT mode selected with f_{CPU} based on f_{OSC} divided by 32. All I/O pins in input mode with a static value at V_{DD} or V_{SS} (no load), all peripherals in reset state; clock input (CLKIN) driven by external square wave, LVD disabled.

5. All I/O pins in output mode with a static value at V_{SS} (no load), LVD disabled. Data based on characterization results, tested in production at V_{DD} max and f_{CPU} max.

6. All I/O pins in input mode with a static value at V_{DD} or V_{SS} (no load). Data tested in production at V_{DD} max. and f_{CPU} max.

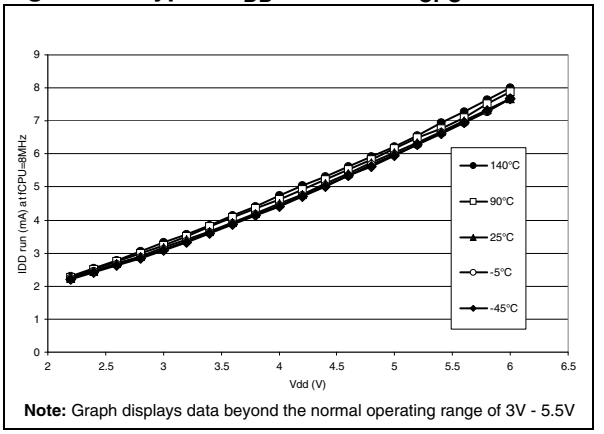
7. This consumption refers to the Halt period only and not the associated run period which is software dependent.

Figure 71. Typical I_{DD} in RUN vs. f_{CPU}



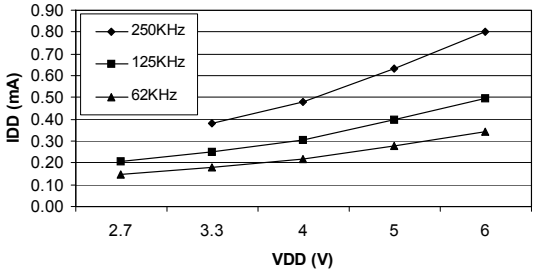
Note: Graph displays data beyond the normal operating range of 3V - 5.5V

Figure 72. Typical I_{DD} in RUN at f_{CPU} = 8MHz



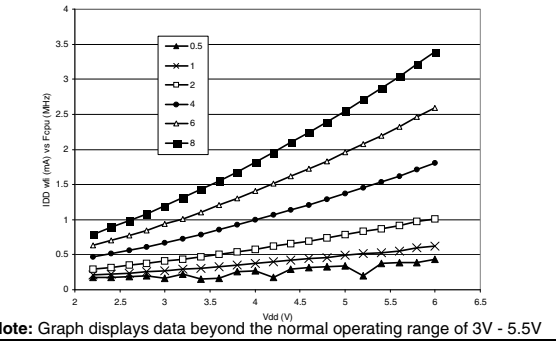
Note: Graph displays data beyond the normal operating range of 3V - 5.5V

Figure 73. Typical I_{DD} in SLOW vs. f_{CPU}



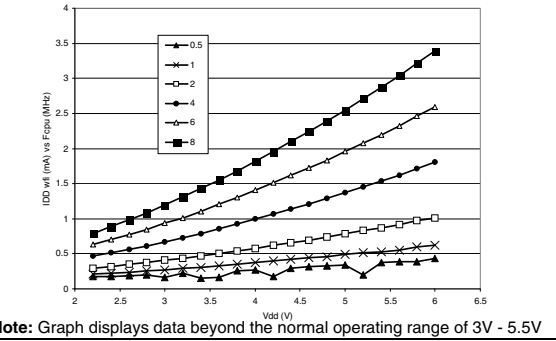
Note: Graph displays data beyond the normal operating range of 3V - 5.5V

Figure 74. Typical I_{DD} in WAIT vs. f_{CPU}



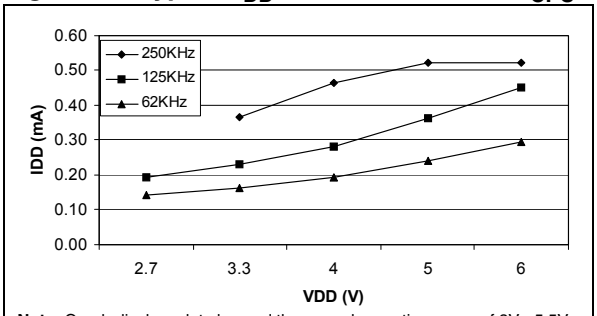
Note: Graph displays data beyond the normal operating range of 3V - 5.5V

Figure 75. Typical I_{DD} in WAIT at f_{CPU}= 8MHz



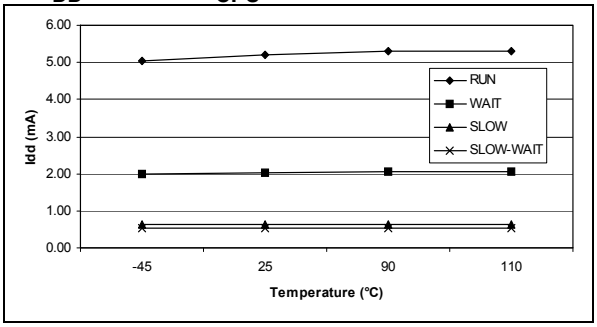
Note: Graph displays data beyond the normal operating range of 3V - 5.5V

Figure 76. Typical I_{DD} in SLOW-WAIT vs. f_{CPU}



Note: Graph displays data beyond the normal operating range of 3V - 5.5V

Figure 77. Typical I_{DD} vs. Temperature at V_{DD} = 5V and f_{CPU} = 8MHz



CLOCK AND TIMING CHARACTERISTICS (Cont'd)

13.5.4 Crystal and Ceramic Resonator Oscillators

The ST7 internal clock can be supplied with ten different Crystal/Ceramic resonator oscillators. All the information given in this paragraph are based on characterization results with specified typical external components. In the application, the resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and start-up stabilization time. Refer to the crystal/ceramic resonator manufacturer for more details (frequency, package, accuracy...).

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{CrOSC}	Crystal Oscillator Frequency		2		16	MHz
C_{L1} C_{L2}	Recommended load capacitance versus equivalent serial resistance of the crystal or ceramic resonator (R_S)		See table below			pF

Supplier	f_{CrOSC} (MHz)	Typical Ceramic Resonators[1]		CL1 [3] [pF]	CL2 [3] [pF]	Rd [Ω]	Supply Voltage Range [V]	Temperature Range [°C]
		Type[2]	Reference					
Murata	1	SMD	CSBFB1M00J58-R0	220	*220*	2.2k	3.3V to 5.5V	-40 to 85
		LEAD	CSBLA1M00J58-B0	220	220	2.2k		
	2	SMD	CSTCC2M00G56Z-R0	(47)	(47)	0	3.0V to 5.5V	
	4	SMD	CSTCR4M00G53Z-R0	(15)	(15)	0		
		LEAD	CSTLS4M00G53Z-B0	(15)	(15)	0		
	8	SMD	CSTCE8M00G52Z-R0	(10)	(10)	0		
		LEAD	CSTLS8M00G53Z-B0	(15)	(15)	0		
	12	SMD	CSTCE12M0G52Z-R0	(10)	(10)	0	3.3V to 5.5V	
	16	SMD	CSTCE16M0V51Z-R0	(5)	(5)	0		
		LEAD	CSTLS16M0X51Z-B0	(5)	(5)	0		

Notes:

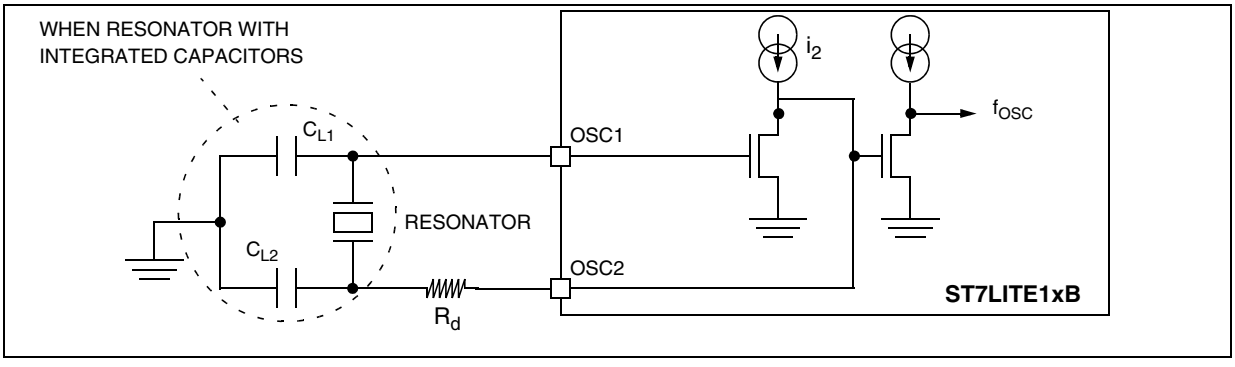
1. Resonator characteristics given by the ceramic resonator manufacturer. For more information on these resonators, please consult www.murata.com

2. SMD = [-R0: Plastic tape package (∅ =180mm)]

LEAD = [-B0: Bulk]

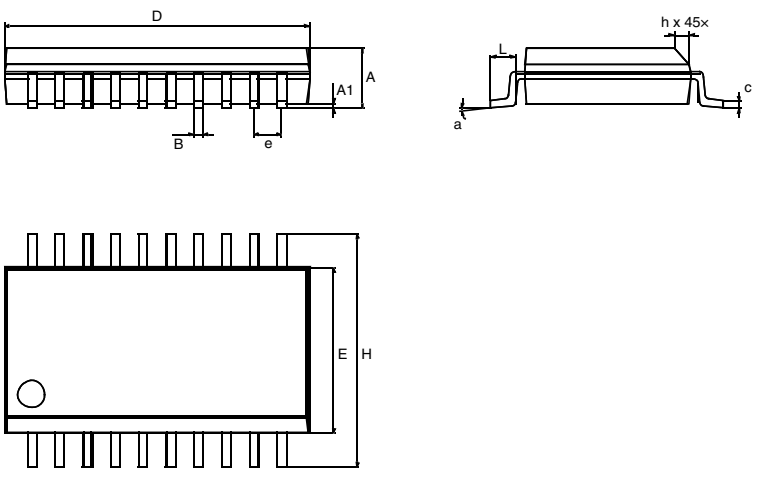
3. () means load capacitor built in resonator

Figure 79. Typical Application with a Crystal or Ceramic Resonator



PACKAGE CHARACTERISTICS (Cont'd)

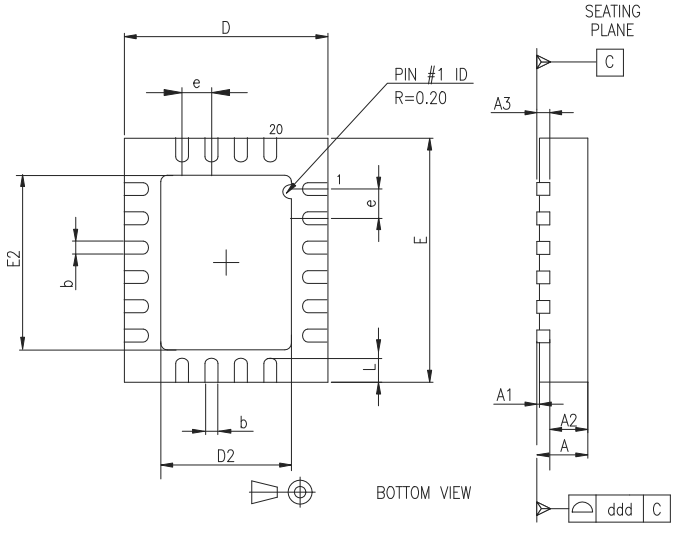
Figure 116. 20-Pin Plastic Small Outline Package, 300-mil Width



Dim.	mm			inches)		
	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 . Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 117. 20-Lead Very thin Fine pitch Quad Flat No-Lead Package



Dim.	mm			inches)		
	Min	Typ	Max	Min	Typ	Ma
A	0.80	0.85	0.90	0.0315	0.0335	0.03
A1	0.00	0.02	0.05		0.0008	0.00
A3		0.02			0.0008	
b	0.25	0.30	0.35	0.0098	0.0118	0.01
D		5.00			0.1969	
D2	3.10	3.25	3.35	0.1220	0.1280	0.13
E		6.00			0.2362	
E2	4.10	4.25	4.35	0.1614	0.1673	0.17
e		0.80			0.0315	
L	0.45	0.50	0.55	0.0177	0.0197	0.02
ddd		0.08			0.0031	
	Number of Pins					
N	20					

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

14.2 SOLDERING INFORMATION

In accordance with the RoHS European directive, all STMicroelectronics packages have been converted to lead-free technology, named ECOPACK™.

- ECOPACK™ packages are qualified according to the JEDEC STD-020C compliant soldering profile.
- Detailed information on the STMicroelectronics ECOPACK™ 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.

– ECOPACK™ TQFP, SDIP, SO and QFN20 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 25. 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 *
QFN	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.