Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ST7
Core Size	8-Bit
Speed	16MHz
Connectivity	SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	22
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	3.2V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SO
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72c104g1m6-tr

Table of Contents

1 INTRODUCTION

The ST72104G, ST72215G, ST72216G and ST72254G devices are members of the ST7 microcontroller family. They can be grouped as follows:

– ST72254G devices are designed for mid-range applications with ADC and I²C interface capabilities.

– ST72215/6G devices target the same range of applications but without I²C interface.

– ST72104G devices are for applications that do not need ADC and I²C peripherals.

All devices are based on a common industry-standard 8-bit core, featuring an enhanced instruction set.

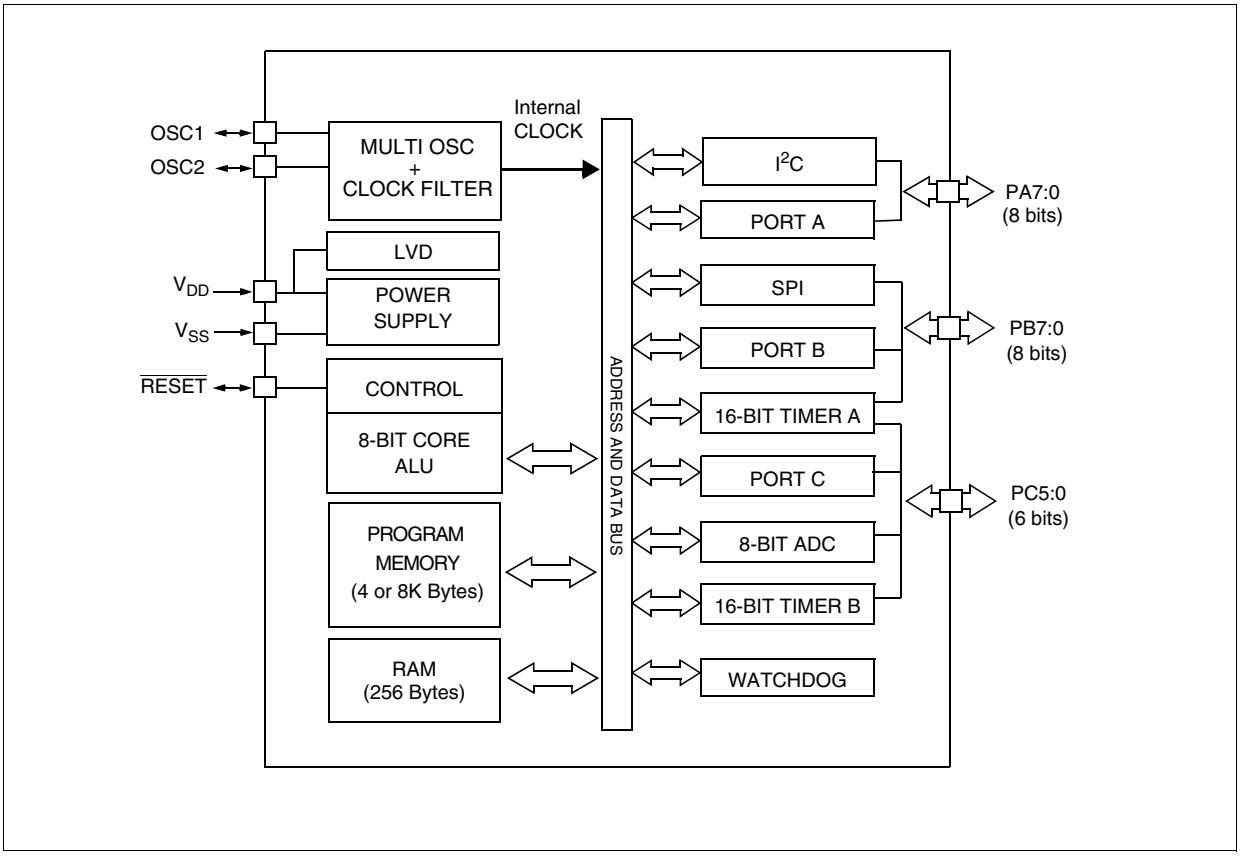
The ST72C104G, ST72C215G, ST72C216G and ST72C254G versions feature single-voltage FLASH memory with byte-by-byte In-Situ Programming (ISP) capability.

Under software control, all devices can be placed in WAIT, SLOW, or HALT mode, reducing power consumption when the application is in idle or stand-by state.

The enhanced instruction set and addressing modes of the ST7 offer both power and flexibility to software developers, enabling the design of highly efficient and compact application code. In addition to standard 8-bit data management, all ST7 microcontrollers feature true bit manipulation, 8x8 unsigned multiplication and indirect addressing modes.

For easy reference, all parametric data are located in Section 13 on page 96.

Figure 1. General Block Diagram



4 FLASH PROGRAM MEMORY

4.1 INTRODUCTION

FLASH devices have a single voltage non-volatile FLASH memory that may be programmed in-situ (or plugged in a programming tool) on a byte-by-byte basis.

4.2 MAIN FEATURES

- Remote In-Situ Programming (ISP) mode
- Up to 16 bytes programmed in the same cycle
- MTP memory (Multiple Time Programmable)
- Read-out memory protection against piracy

4.3 STRUCTURAL ORGANISATION

The FLASH program memory is organised in a single 8-bit wide memory block which can be used for storing both code and data constants.

The FLASH program memory is mapped in the upper part of the ST7 addressing space and includes the reset and interrupt user vector area .

4.4 IN-SITU PROGRAMMING (ISP) MODE

The FLASH program memory can be programmed using Remote ISP mode. This ISP mode allows the contents of the ST7 program memory to be updated using a standard ST7 programming tools after the device is mounted on the application board. This feature can be implemented with a minimum number of added components and board area impact.

An example Remote ISP hardware interface to the standard ST7 programming tool is described below. For more details on ISP programming, refer to the ST7 Programming Specification.

Remote ISP Overview

The Remote ISP mode is initiated by a specific sequence on the dedicated ISPSEL pin.

The Remote ISP is performed in three steps:

– Selection of the RAM execution mode
– Download of Remote ISP code in RAM
– Execution of Remote ISP code in RAM to program the user program into the FLASH

Remote ISP hardware configuration

In Remote ISP mode, the ST7 has to be supplied with power (V_{DD} and V_{SS}) and a clock signal (oscillator and application crystal circuit for example).

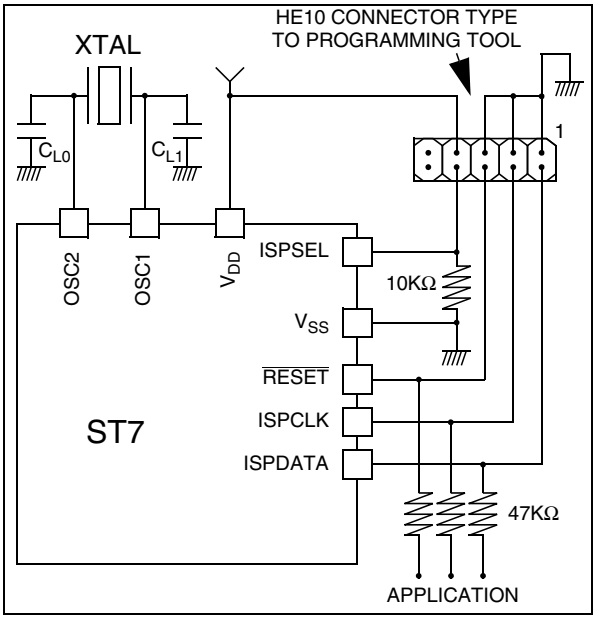
This mode needs five signals (plus the V_{DD} signal if necessary) to be connected to the programming tool. This signals are:

– $\overline{\text{RESET}}$: device reset
– V_{SS}: device ground power supply
– ISPCLK: ISP output serial clock pin
– ISPDATA: ISP input serial data pin
– ISPSEL: Remote ISP mode selection. This pin must be connected to V_{SS} on the application board through a pull-down resistor.

If any of these pins are used for other purposes on the application, a serial resistor has to be implemented to avoid a conflict if the other device forces the signal level.

Figure 5 shows a typical hardware interface to a standard ST7 programming tool. For more details on the pin locations, refer to the device pinout description.

Figure 5. Typical Remote ISP Interface



4.5 MEMORY READ-OUT PROTECTION

The read-out protection is enabled through an option bit.

For FLASH devices, when this option is selected, the program and data stored in the FLASH memory are protected against read-out piracy (including a re-write protection). When this protection option is removed the entire FLASH program memory is first automatically erased. However, the E^2PROM data memory (when available) can be protected only with ROM devices.

6 SUPPLY, RESET AND CLOCK MANAGEMENT

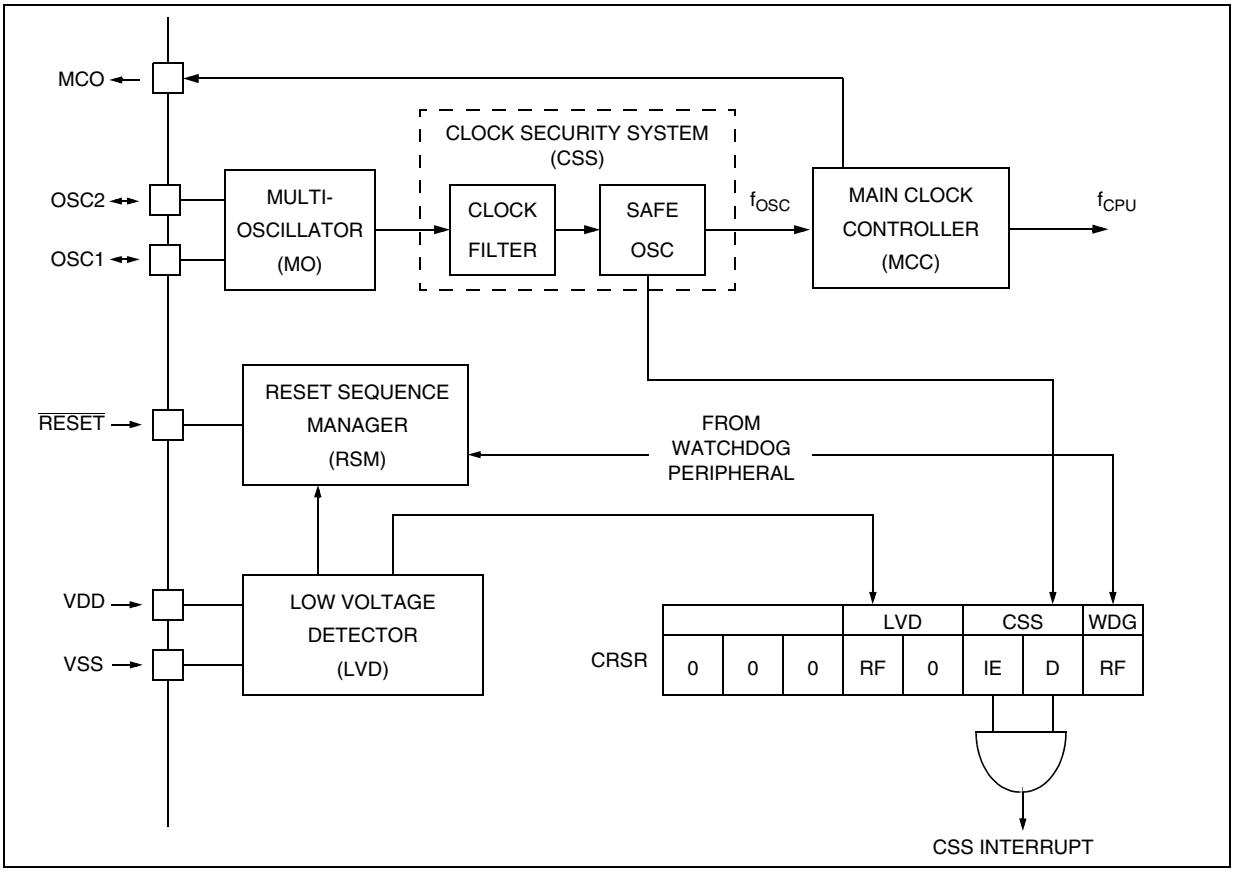
The ST72104G, ST72215G, ST72216G and ST72254G microcontrollers include a range of utility features for securing the application in critical situations (for example in case of a power brown-out), and reducing the number of external components. An overview is shown in Figure 8.

See Section 13 "ELECTRICAL CHARACTERISTICS" on page 96 for more details.

Main Features

- Supply Manager with main supply low voltage detection (LVD)
- Reset Sequence Manager (RSM)
- Multi-Oscillator (MO)
  - 4 Crystal/Ceramic resonator oscillators
  - 1 External RC oscillator
  - 1 Internal RC oscillator
- Clock Security System (CSS)
  - Clock Filter
  - Backup Safe Oscillator

Figure 8. Clock, Reset and Supply Block Diagram



RESET SEQUENCE MANAGER (Cont'd)

6.2.2 Asynchronous External $\overline{\text{RESET}}$ pin

The $\overline{\text{RESET}}$ pin is both an input and an open-drain output with integrated R_{ON} weak pull-up resistor. This pull-up has no fixed value but varies in accordance with the input voltage. It can be pulled low by external circuitry to reset the device. See electrical characteristics section for more details.

A RESET signal originating from an external source must have a duration of at least $t_{h(RSTL)in}$ in order to be recognized. This detection is asynchronous and therefore the MCU can enter reset state even in HALT mode.

The $\overline{\text{RESET}}$ pin is an asynchronous signal which plays a major role in EMS performance. In a noisy environment, it is recommended to follow the guidelines mentioned in the electrical characteristics section.

Two RESET sequences can be associated with this RESET source: short or long external reset pulse (see Figure 12).

Starting from the external RESET pulse recognition, the device $\overline{\text{RESET}}$ pin acts as an output that is pulled low during at least $t_{w(RSTL)out}$.

6.2.3 Internal Low Voltage Detection RESET

Two different RESET sequences caused by the internal LVD circuitry can be distinguished:

- Power-On RESET
- Voltage Drop RESET

The device $\overline{\text{RESET}}$ pin acts as an output that is pulled low when $V_{DD}<V_{IT+}$ (rising edge) or $V_{DD}<V_{IT-}$ (falling edge) as shown in Figure 12.

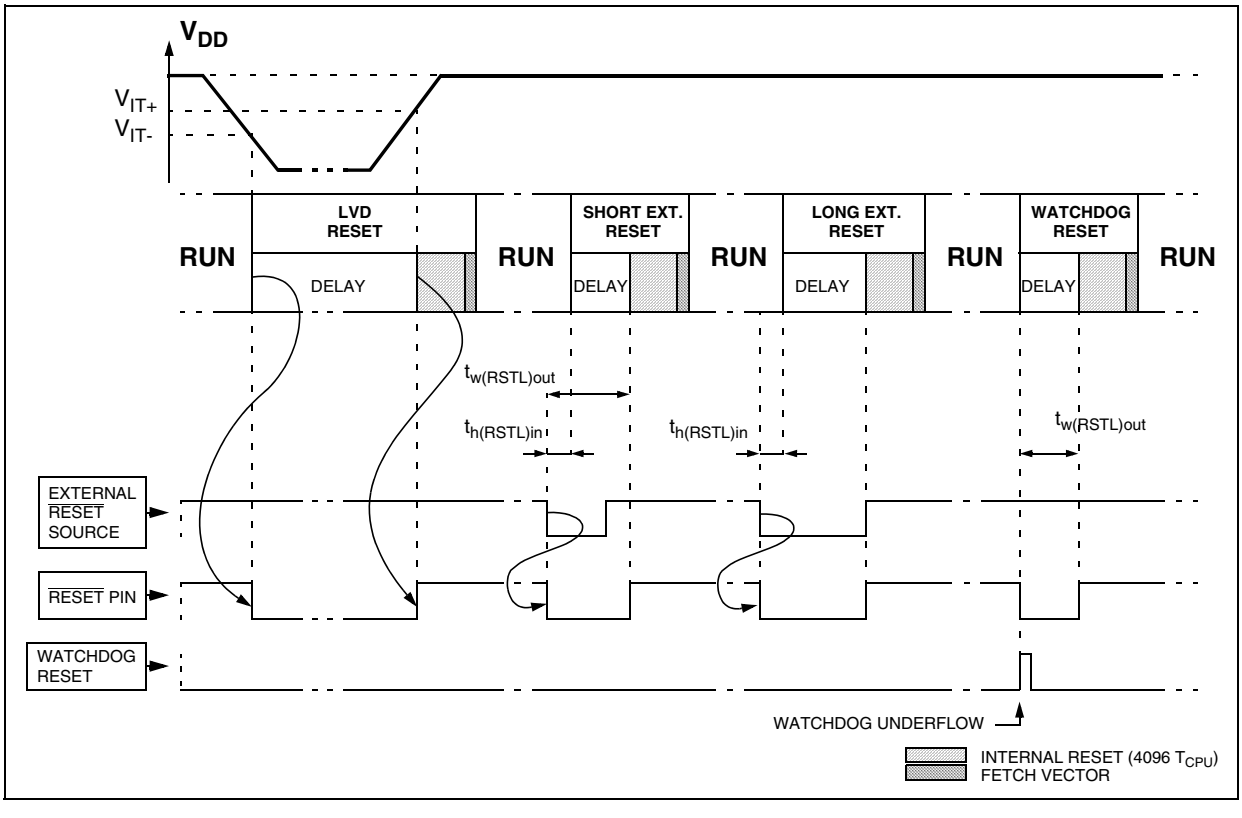
The LVD filters spikes on V_{DD} larger than $t_{g(VDD)}$ to avoid parasitic resets.

6.2.4 Internal Watchdog RESET

The RESET sequence generated by a internal Watchdog counter overflow is shown in Figure 12.

Starting from the Watchdog counter underflow, the device $\overline{\text{RESET}}$ pin acts as an output that is pulled low during at least $t_{w(RSTL)out}$.

Figure 12. RESET Sequences



6.6 MAIN CLOCK CONTROLLER (MCC)

The Main Clock Controller (MCC) supplies the clock for the ST7 CPU and its internal peripherals. It allows SLOW power saving mode to be managed by the application.

All functions are managed by the Miscellaneous register 1 (MISCR1).

The MCC block consists of:

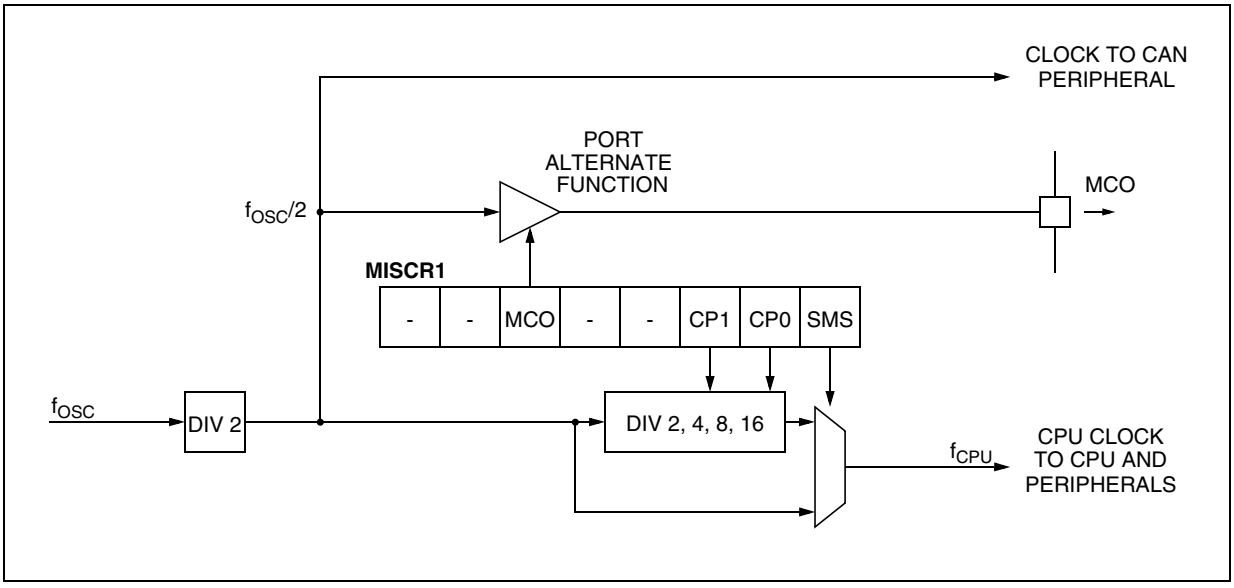
- A programmable CPU clock prescaler
- A clock-out signal to supply external devices

The prescaler allows the selection of the main clock frequency and is controlled by three bits of the MISCR1: CP1, CP0 and SMS.

The clock-out capability consists of a dedicated I/O port pin configurable as an f_{CPU} clock output to drive external devices. It is controlled by the MCO bit in the MISCR1 register.

See Section 10 "MISCELLANEOUS REGISTERS" on page 36 for more details.

Figure 14. Main Clock Controller (MCC) Block Diagram



POWER SAVING MODES (Cont'd)

8.4 HALT MODE

The HALT mode is the lowest power consumption mode of the MCU. It is entered by executing the ST7 HALT instruction (see Figure 20).

The MCU can exit HALT mode on reception of either a specific interrupt (see Table 5, "Interrupt Mapping," on page 26) or a RESET. When exiting HALT mode by means of a RESET or an interrupt, the oscillator is immediately turned on and the 4096 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 19).

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 immediately.

In the 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 "OPTION BYTES" on page 133 for more details).

Figure 19. HALT Mode Timing Overview

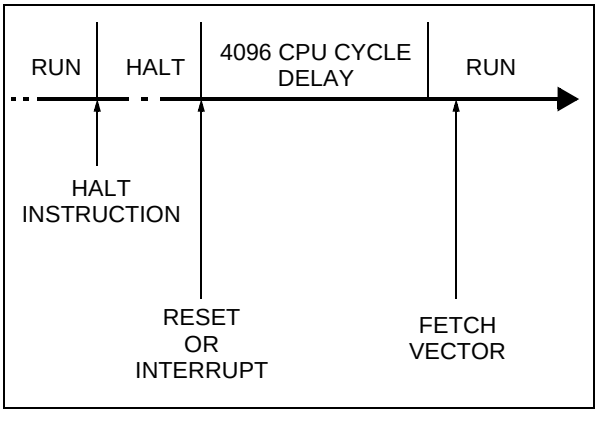
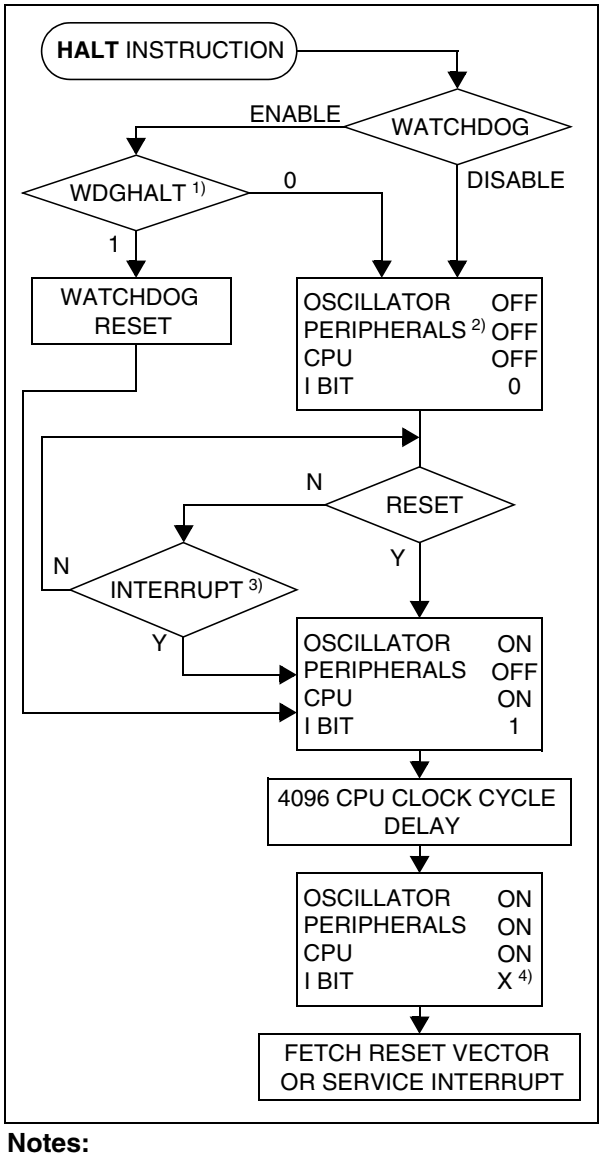


Figure 20. HALT Mode Flow-chart



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 5, "Interrupt Mapping," on page 26 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.

MISCELLANEOUS REGISTERS (Cont'd)

10.3 MISCELLANEOUS REGISTER DESCRIPTION

MISCELLANEOUS REGISTER 1 (MISCR1)

Read/Write

Reset Value: 0000 0000 (00h)

7							0
IS11	IS10	MCO	IS01	IS00	CP1	CP0	SMS

Bit 7:6 = **IS1[1:0]** *ei1 sensitivity*
The interrupt sensitivity, defined using the IS1[1:0] bits, is applied to the ei1 external interrupts. These two bits can be written only when the I bit of the CC register is set to 1 (interrupt masked).

ei1: Port B (C optional)

External Interrupt Sensitivity	IS11	IS10
Falling edge & low level	0	0
Rising edge only	0	1
Falling edge only	1	0
Rising and falling edge	1	1

Bit 5 = **MCO** *Main clock out selection*
This bit enables the MCO alternate function on the PC2 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)

Bit 4:3 = **IS0[1:0]** *ei0 sensitivity*
The interrupt sensitivity, defined using the IS0[1:0] bits, is applied to the ei0 external interrupts. These two bits can be written only when the I bit of the CC register is set to 1 (interrupt masked).

ei0: Port A (C optional)

External Interrupt Sensitivity	IS01	IS00
Falling edge & low level	0	0
Rising edge only	0	1
Falling edge only	1	0
Rising and falling edge	1	1

Bit 2:1 = **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_{OSC} / 4	0	0
f_{OSC} / 8	1	0
f_{OSC} / 16	0	1
f_{OSC} / 32	1	1

Bit 0 = **SMS** *Slow mode select*
This bit is set and cleared by software.
0: Normal mode. $f_{CPU} = f_{OSC} / 2$
1: Slow mode. f_{CPU} is given by CP1, CP0
See low power consumption mode and MCC chapters for more details.

16-BIT TIMER (Cont'd)

11.2.3.3 Input Capture

In this section, the index, *i*, may be 1 or 2 because there are 2 input capture functions in the 16-bit timer.

The two input capture 16-bit registers (IC1R and IC2R) are used to latch the value of the free running counter after a transition is detected by the ICAP*i* pin (see Figure 5).

	MS Byte	LS Byte
ICiR	IC*i*HR	IC*i*LR

The IC*i*R register is a read-only register.

The active transition is software programmable through the IEDG*i* bit of Control Registers (CR*i*).

Timing resolution is one count of the free running counter: (f_{CPU}/CC[1:0]).

Procedure:

To use the input capture function, select the following in the CR2 register:

– Select the timer clock (CC[1:0]) (see Table 1).

– Select the edge of the active transition on the ICAP2 pin with the IEDG2 bit (the ICAP2 pin must be configured as a floating input or input with pull-up without interrupt if this configuration is available).

And select the following in the CR1 register:

– Set the ICIE bit to generate an interrupt after an input capture coming from either the ICAP1 pin or the ICAP2 pin

– Select the edge of the active transition on the ICAP1 pin with the IEDG1 bit (the ICAP1 pin must be configured as a floating input or input with pull-up without interrupt if this configuration is available).

When an input capture occurs:

– The ICF*i* bit is set.

– The IC*i*R register contains the value of the free running counter on the active transition on the ICAP*i* pin (see Figure 6).

– A timer interrupt is generated if the ICIE bit is set and the I bit is cleared in the CC register. Otherwise, the interrupt remains pending until both conditions become true.

Clearing the Input Capture interrupt request (i.e. clearing the ICF*i* bit) is done in two steps:

1. Reading the SR register while the ICF*i* bit is set.
2. An access (read or write) to the IC*i*LR register.

Notes:

1. After reading the IC*i*HR register, the transfer of input capture data is inhibited and ICF*i* will never be set until the IC*i*LR register is also read.
2. The IC*i*R register contains the free running counter value which corresponds to the most recent input capture.
3. The two input capture functions can be used together even if the timer also uses the two output compare functions.
4. In One Pulse mode and PWM mode only the input capture 2 function can be used.
5. The alternate inputs (ICAP1 and ICAP2) are always directly connected to the timer. So any transitions on these pins activate the input capture function.
   Moreover if one of the ICAP*i* pin is configured as an input and the second one as an output, an interrupt can be generated if the user toggles the output pin and if the ICIE bit is set.
   This can be avoided if the input capture function *i* is disabled by reading the IC*i*HR (see note 1).
6. The TOF bit can be used with an interrupt in order to measure events that exceed the timer range (FFFFh).

16-BIT TIMER (Cont'd)

STATUS REGISTER (SR)

Read Only

Reset Value: 0000 0000 (00h)

The three least significant bits are not used.

7							0
ICF1	OCF1	TOF	ICF2	OCF2	0	0	0

Bit 7 = **ICF1** *Input Capture Flag 1.*
0: No input capture (reset value).
1: An input capture has occurred on the ICAP1 pin or the counter has reached the OC2R value in PWM mode. To clear this bit, first read the SR register, then read or write the low byte of the IC1R (IC1LR) register.

Bit 6 = **OCF1** *Output Compare Flag 1.*
0: No match (reset value).
1: The content of the free running counter matches the content of the OC1R register. To clear this bit, first read the SR register, then read or write the low byte of the OC1R (OC1LR) register.

Bit 5 = **TOF** *Timer Overflow Flag.*
0: No timer overflow (reset value).
1: The free running counter has rolled over from FFFFh to 0000h. To clear this bit, first read the SR register, then read or write the low byte of the CR (CLR) register.

Note: Reading or writing the ACLR register does not clear TOF.

Bit 4 = **ICF2** *Input Capture Flag 2.*
0: No input capture (reset value).
1: An input capture has occurred on the ICAP2 pin. To clear this bit, first read the SR register, then read or write the low byte of the IC2R (IC2LR) register.

Bit 3 = **OCF2** *Output Compare Flag 2.*
0: No match (reset value).
1: The content of the free running counter matches the content of the OC2R register. To clear this bit, first read the SR register, then read or write the low byte of the OC2R (OC2LR) register.

Bit 2-0 = Reserved, forced by hardware to 0.

INPUT CAPTURE 1 HIGH REGISTER (IC1HR)

Read Only
Reset Value: Undefined

This is an 8-bit read only register that contains the high part of the counter value (transferred by the input capture 1 event).

7							0
MSB							LSB

INPUT CAPTURE 1 LOW REGISTER (IC1LR)

Read Only
Reset Value: Undefined

This is an 8-bit read only register that contains the low part of the counter value (transferred by the input capture 1 event).

7							0
MSB							LSB

OUTPUT COMPARE 1 HIGH REGISTER (OC1HR)

Read/Write
Reset Value: 1000 0000 (80h)

This is an 8-bit register that contains the high part of the value to be compared to the CHR register.

7							0
MSB							LSB

OUTPUT COMPARE 1 LOW REGISTER (OC1LR)

Read/Write
Reset Value: 0000 0000 (00h)

This is an 8-bit register that contains the low part of the value to be compared to the CLR register.

7							0
MSB							LSB

SERIAL PERIPHERAL INTERFACE (Cont'd)

11.3.4.7 Single Master and Multimaster Configurations

There are two types of SPI systems:

– Single Master System

– Multimaster System

Single Master System

A typical single master system may be configured, using an MCU as the master and four MCUs as slaves (see Figure 6).

The master device selects the individual slave devices by using four pins of a parallel port to control the four $\overline{SS}$ pins of the slave devices.

The $\overline{SS}$ pins are pulled high during reset since the master device ports will be forced to be inputs at that time, thus disabling the slave devices.

Note: To prevent a bus conflict on the MISO line the master allows only one active slave device during a transmission.

For more security, the slave device may respond to the master with the received data byte. Then the master will receive the previous byte back from the slave device if all MISO and MOSI pins are connected and the slave has not written its DR register.

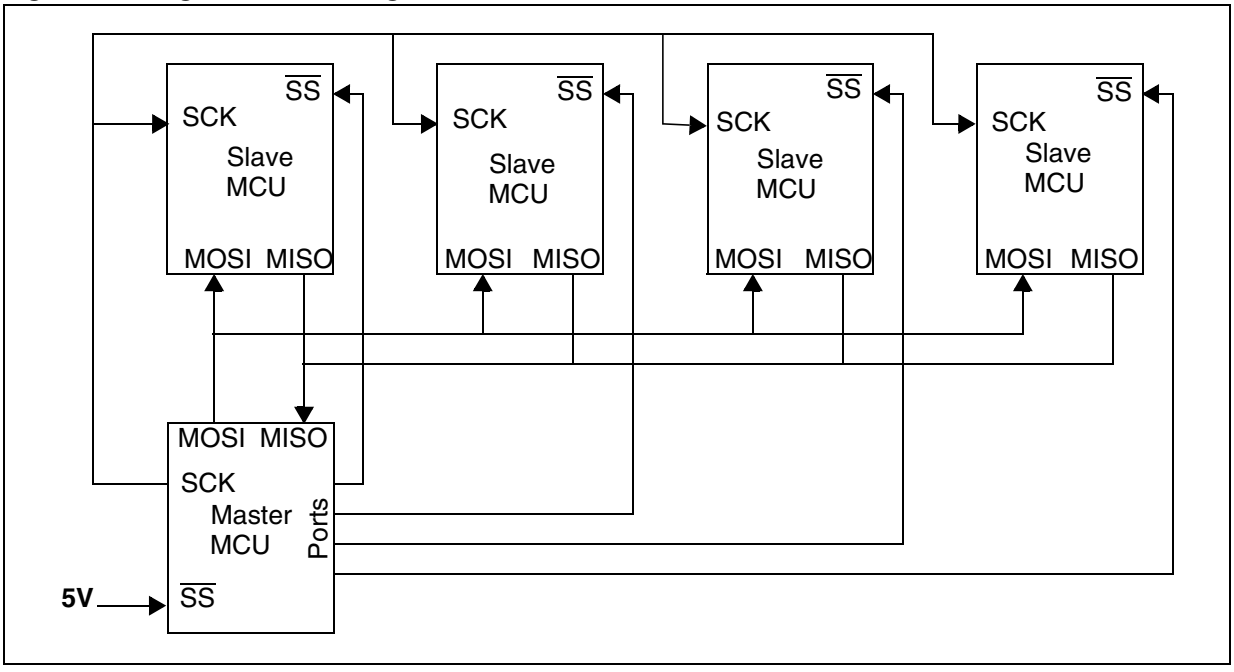
Other transmission security methods can use ports for handshake lines or data bytes with command fields.

Multi-master System

A multi-master system may also be configured by the user. Transfer of master control could be implemented using a handshake method through the I/O ports or by an exchange of code messages through the serial peripheral interface system.

The multi-master system is principally handled by the MSTR bit in the CR register and the MODF bit in the SR register.

Figure 42. Single Master Configuration



I²C BUS INTERFACE (Cont'd)

11.4.7 Register Description

I²C CONTROL REGISTER (CR)

Read / Write
Reset Value: 0000 0000 (00h)

7							0
0	0	PE	ENGC	START	ACK	STOP	ITE

Bit 7:6 = Reserved. Forced to 0 by hardware.

Bit 5 = **PE** *Peripheral enable.*
This bit is set and cleared by software.
0: Peripheral disabled
1: Master/Slave capability
Notes:
– When PE=0, all the bits of the CR register and the SR register except the Stop bit are reset. All outputs are released while PE=0
– When PE=1, the corresponding I/O pins are selected by hardware as alternate functions.
– To enable the I²C interface, write the CR register **TWICE** with PE=1 as the first write only activates the interface (only PE is set).

Bit 4 = **ENGC** *Enable General Call.*
This bit is set and cleared by software. It is also cleared by hardware when the interface is disabled (PE=0). The 00h General Call address is acknowledged (01h ignored).
0: General Call disabled
1: General Call enabled

Note: In accordance with the I2C standard, when GCAL addressing is enabled, an I2C slave can only receive data. It will not transmit data to the master.

Bit 3 = **START** *Generation of a Start condition*.
This bit is set and cleared by software. It is also cleared by hardware when the interface is disabled (PE=0) or when the Start condition is sent (with interrupt generation if ITE=1).

– In master mode:
  0: No start generation
  1: Repeated start generation
– In slave mode:
  0: No start generation
  1: Start generation when the bus is free

Bit 2 = **ACK** *Acknowledge enable.*
This bit is set and cleared by software. It is also cleared by hardware when the interface is disabled (PE=0).
0: No acknowledge returned
1: Acknowledge returned after an address byte or a data byte is received

Bit 1 = **STOP** *Generation of a Stop condition*.
This bit is set and cleared by software. It is also cleared by hardware in master mode. Note: This bit is not cleared when the interface is disabled (PE=0).

– In master mode:
  0: No stop generation
  1: Stop generation after the current byte transfer or after the current Start condition is sent. The STOP bit is cleared by hardware when the Stop condition is sent.
– In slave mode:
  0: No stop generation
  1: Release the SCL and SDA lines after the current byte transfer (BTF=1). In this mode the STOP bit has to be cleared by software.

Bit 0 = **ITE** *Interrupt enable.*
This bit is set and cleared by software and cleared by hardware when the interface is disabled (PE=0).
0: Interrupts disabled
1: Interrupts enabled
Refer to Figure 4 for the relationship between the events and the interrupt.
SCL is held low when the ADD10, SB, BTF or ADSL flags or an EV6 event (See Figure 3) is detected.

11.5 8-BIT A/D CONVERTER (ADC)

11.5.1 Introduction

The on-chip Analog to Digital Converter (ADC) peripheral is a 8-bit, successive approximation converter with internal sample and hold circuitry. This peripheral has up to 16 multiplexed analog input channels (refer to device pin out description) that allow the peripheral to convert the analog voltage levels from up to 16 different sources.

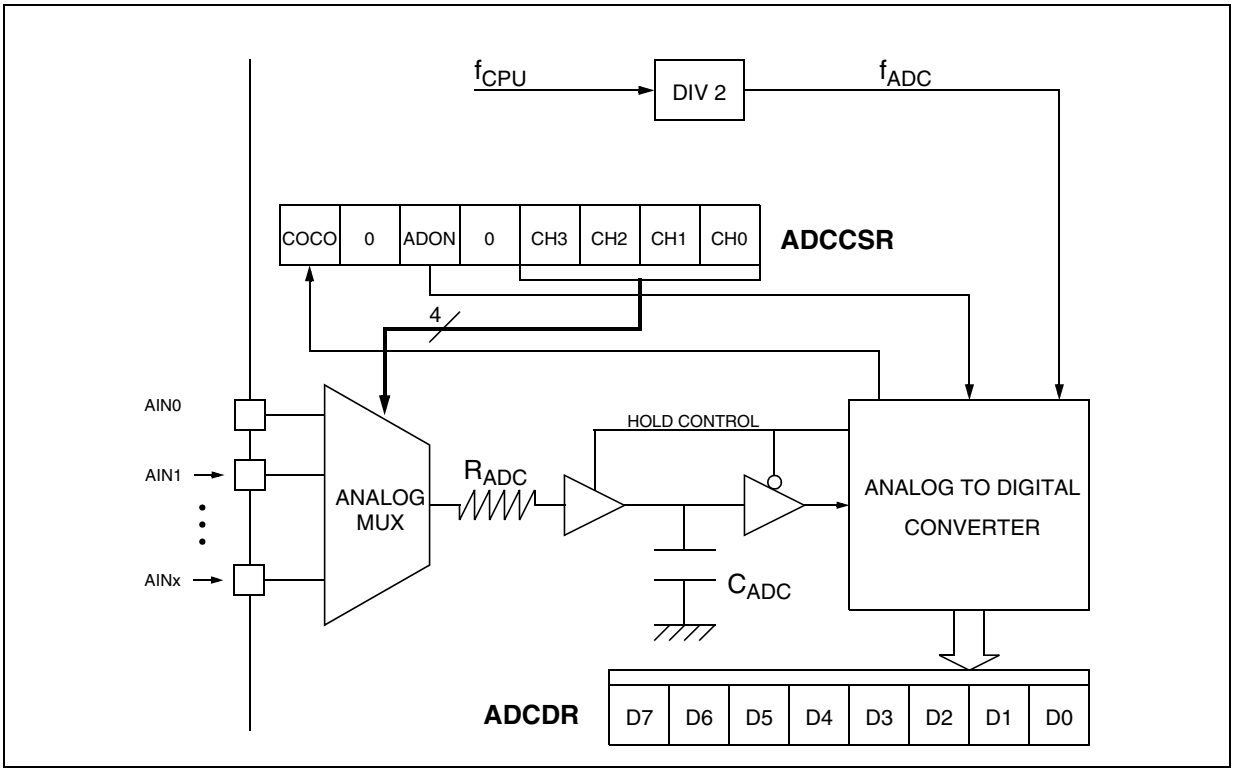
The result of the conversion is stored in a 8-bit Data Register. The A/D converter is controlled through a Control/Status Register.

11.5.2 Main Features

- 8-bit conversion
- Up to 16 channels with multiplexed input
- Linear successive approximation
- Data register (DR) which contains the results
- Conversion complete status flag
- On/off bit (to reduce consumption)

The block diagram is shown in Figure 1.

Figure 47. ADC Block Diagram



11.5.3 Functional Description

11.5.3.1 Analog Power Supply

V_{DDA} and V_{SSA} are the high and low level reference voltage pins. In some devices (refer to device pin out description) they are internally connected to the V_{DD} and V_{SS} pins.

Conversion accuracy may therefore be impacted by voltage drops and noise in the event of heavily loaded or badly decoupled power supply lines.

See electrical characteristics section for more details.

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

11.5.6 Register Description

CONTROL/STATUS REGISTER (CSR)

Read/Write

Reset Value: 0000 0000 (00h)

7							0
COCO	0	ADON	0	CH3	CH2	CH1	CH0

Bit 7 = **COCO** *Conversion Complete*
This bit is set by hardware. It is cleared by software reading the result in the DR register or writing to the CSR register.
0: Conversion is not complete
1: Conversion can be read from the DR register

Bit 6 = **Reserved.** *must always be cleared.*

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

Bit 4 = **Reserved.** *must always be cleared.*

Bits 3:0 = **CH[3:0]** *Channel Selection*
These bits are set and cleared by software. They select the analog input to convert.

Channel Pin*	CH3	CH2	CH1	CH0
AIN0	0	0	0	0
AIN1	0	0	0	1
AIN2	0	0	1	0
AIN3	0	0	1	1
AIN4	0	1	0	0
AIN5	0	1	0	1
AIN6	0	1	1	0
AIN7	0	1	1	1
AIN8	1	0	0	0
AIN9	1	0	0	1
AIN10	1	0	1	0
AIN11	1	0	1	1
AIN12	1	1	0	0
AIN13	1	1	0	1
AIN14	1	1	1	0
AIN15	1	1	1	1

***Note**: The number of pins AND the channel selection varies according to the device. Refer to the device pinout.

DATA REGISTER (DR)

Read Only

Reset Value: 0000 0000 (00h)

7							0
D7	D6	D5	D4	D3	D2	D1	D0

Bits 7:0 = **D[7:0]** *Analog Converted Value*
This register contains the converted analog value in the range 00h to FFh.

Note: Reading this register reset the COCO flag.

12.2 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	CP	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 bytes.

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.

13.5 CLOCK AND TIMING CHARACTERISTICS

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

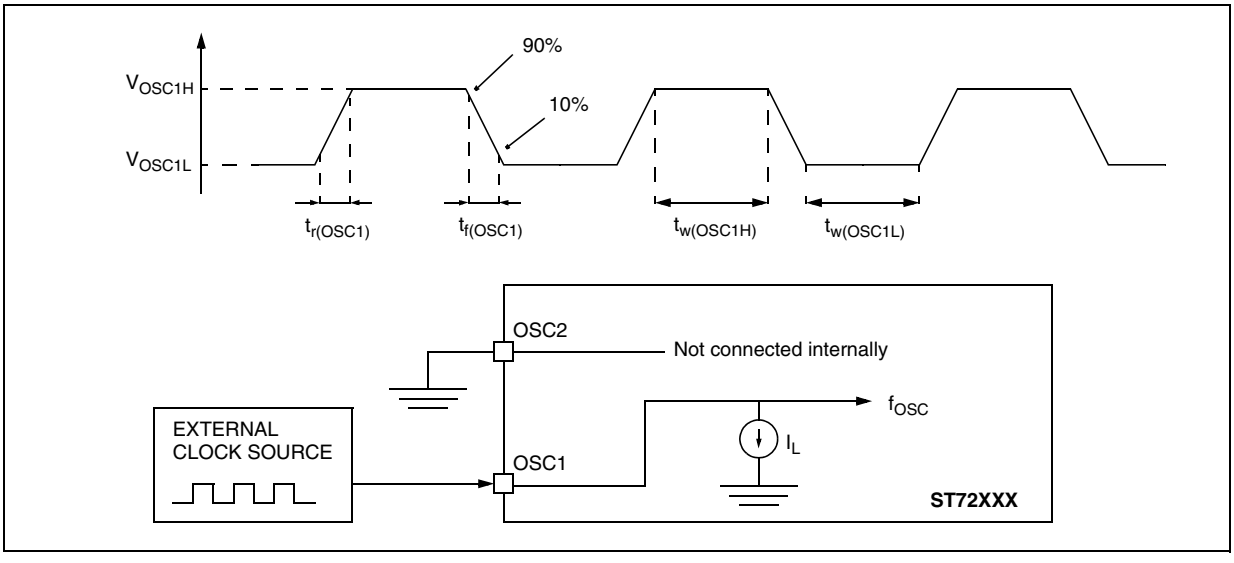
13.5.1 General Timings

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

13.5.2 External Clock Source

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{OSC1H}	OSC1 input pin high level voltage	see Figure 63	$0.7xV_{DD}$		V_{DD}	V
V_{OSC1L}	OSC1 input pin low level voltage		V_{SS}		$0.3xV_{DD}$	
$t_{w(OSC1H)}$ $t_{w(OSC1L)}$	OSC1 high or low time [3]		15			ns
$t_{r(OSC1)}$ $t_{f(OSC1)}$	OSC1 rise or fall time [3]				15	
I_L	OSCx Input leakage current	$V_{SS} \le V_{IN} \le V_{DD}$			±1	µA

Figure 63. Typical Application with an External Clock Source



Notes:

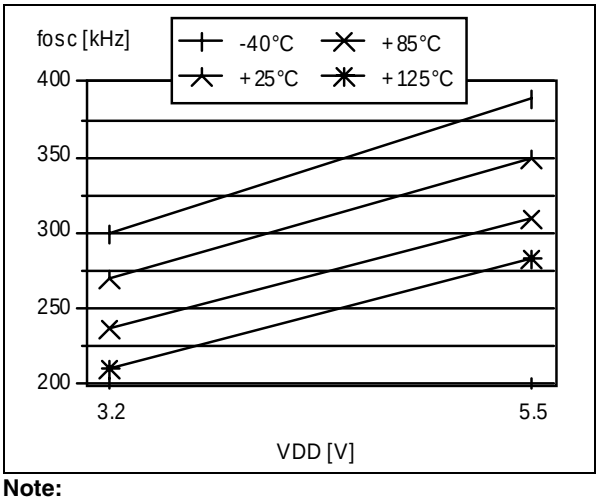
1. Data based on typical application software.
2. Time measured between interrupt event and interrupt vector fetch. $\Delta t_{c(INST)}$ is the number of t_{CPU} cycles needed to finish the current instruction execution.
3. Data based on design simulation and/or technology characteristics, not tested in production.

CLOCK CHARACTERISTICS (Cont'd)

13.5.5 Clock Security System (CSS)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{SFOSC}	Safe Oscillator Frequency [1]	T_A=25°C, V_{DD}=5.0V	250	340	550	kHz
		T_A=25°C, V_{DD}=3.4V	190	260	450	
f_{GFOSC}	Glitch Filtered Frequency [2]			30		MHz

Figure 69. Typical Safe Oscillator Frequencies



Note:

1. Data based on characterization results, tested in production between 90KHz and 600KHz.
2. Filtered glitch on the f_{OSC} signal. See functional description in Section 6.5 on page 23 for more details.

14.2 THERMAL CHARACTERISTICS

Symbol	Ratings	Value	Unit
R_{thJA}	Package thermal resistance (junction to ambient) SDIP32 SO28	 60 75	°C/W
P_D	Power dissipation [1]	500	mW
T_{Jmax}	Maximum junction temperature [2]	150	°C

Notes:

1. The power dissipation is obtained from the formula $P_D=P_{INT}+P_{PORT}$ where P_{INT} is the chip internal power ($I_{DD}xV_{DD}$) and P_{PORT} is the port power dissipation determined by the user.
2. The average chip-junction temperature can be obtained from the formula $T_J = T_A + P_D$ x RthJA.

TRANSFER OF CUSTOMER CODE (Cont'd)

MICROCONTROLLER OPTION LIST

Customer . . . . . .
Address . . . . . .
. . . . . .
Contact . . . . . .
Phone No . . . . . .
Reference . . . . . .

Device: [ ] ST72104G1 (4KB) [ ] ST72215G2 (4KB) [ ] ST72254G1 (4KB)
[ ] ST72104G2 (8KB) [ ] ST72216G1 (8KB) [ ] ST72254G2 (8KB)

Package: [ ] SO28 [ ] Tape & Reel [ ] Tube
[ ] SDIP32

Marking: [ ] Standard Marking
[ ] Special Marking SO28 (max. 13 Chars.): _ _ _ _ _ _ _ _ _ _ _ _ _
SDIP32 (max. 15 Chars.): _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
Authorized characters are letters, digits, '.', '-', '/' and spaces only. Please consult your local STMicroelectronics sales office for other marking details if required.

External Interrupt: [ ] IT0 interrupt vector Port A, IT1 interrupt vector Port B & C
[ ] IT0 interrupt vector Port A & C, IT1 interrupt vector Port B

Temperature Range: [ ] 0°C to + 70°C [ ] - 10°C to + 85°C
[ ] - 40°C to + 85°C [ ] - 40°C to + 105°C [ ] - 40°C to + 125°C

Clock Source Selection: [ ] Resonator: [ ] 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)
[ ] RC Network: [ ] Internal
[ ] External
[ ] External Clock

Clock Security System: [ ] Disabled [ ] Enabled

Watchdog Selection: [ ] Software Activation [ ] Hardware Activation
Halt when Watchdog on: [ ] Reset [ ] No reset

Readout Protection: [ ] Disabled [ ] Enabled

LVD Reset [ ] Disabled [ ] Enabled: [ ] Highest threshold
[ ] Medium threshold
[ ] Lowest threshold

Comments : . . . . . .

Supply Operating Range in the application: . . . . . .

Notes: . . . . . .

Signature: . . . . . .

15.3 DEVELOPMENT TOOLS

STmicroelectronics offers a range of hardware and software development tools for the ST7 microcontroller family. Full details of tools available for the ST7 from third party manufacturers can be obtain from the STMicroelectronics Internet site:
➟ http//mcu.st.com.

Third Party Tools

- ACTUM
- BP
- COSMIC
- CMX
- DATA I/O
- HITEX
- HIWARE
- ISYSTEM
- KANDA
- LEAP

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

STMicroelectronics Tools

Three types of development tool are offered by ST, all of them connect to a PC via a parallel (LPT) port: see Table 26 and Table 27 for more details.

Table 26. STMicroelectronics Tool Features

	In-Circuit Emulation	Programming Capability[1)]	Software Included
ST7 Development Kit	Yes. (Same features as HDS2 emulator but without logic analyzer)	Yes (DIP packages only)	ST7 CD ROM with: – ST7 Assembly toolchain – STVD7 and WGDB7 powerful Source Level Debugger for Win 3.1, Win 95 and NT – C compiler demo versions – ST Realizer for Win 3.1 and Win 95. – Windows Programming Tools for Win 3.1, Win 95 and NT
ST7 HDS2 Emulator	Yes, powerful emulation features including trace/ logic analyzer	No	
ST7 Programming Board	No	Yes (All packages)	

Table 27. Dedicated STMicroelectronics Development Tools

Supported Products	ST7 Development Kit	ST7 HDS2 Emulator	ST7 Programming Board
ST72254G1, ST72C254G1 ST72254G2, ST72C254G2 ST72215G2, ST72C215G2 ST72216G1, ST72C216G1 ST72104G1, ST72C104G1, ST72104G2, ST72C104G2	ST7MDT1-DVP2	ST7MDT1-EMU2B	ST7MDT1-EPB2/EU ST7MDT1-EPB2/US ST7MDT1-EPB2/UK

Note:
1. In-Situ Programming (ISP) interface for FLASH devices.