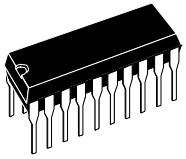
ST6208C/ST6209C
ST6210C/ST6220C

8-BIT MCUs WITH A/D CONVERTER, TWO TIMERS, OSCILLATOR SAFEGUARD & SAFE RESET

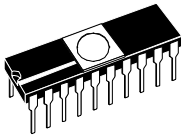
- **Memories**
  - 1K, 2K or 4K bytes Program memory (OTP, EPROM, FASTROM or ROM) with read-out protection
  - 64 bytes RAM
- **Clock, Reset and Supply Management**
  - Enhanced reset system
  - Low Voltage Detector (LVD) for Safe Reset
  - Clock sources: crystal/ceramic resonator or RC network, external clock, backup oscillator (LFAO)
  - Oscillator Safeguard (OSG)
  - 2 Power Saving Modes: Wait and Stop
- **Interrupt Management**
  - 4 interrupt vectors plus NMI and RESET
  - 12 external interrupt lines (on 2 vectors)
- **12 I/O Ports**
  - 12 multifunctional bidirectional I/O lines
  - 8 alternate function lines
  - 4 high sink outputs (20mA)
- **2 Timers**
  - Configurable watchdog timer
  - 8-bit timer/counter with a 7-bit prescaler
- **Analog Peripheral**
  - 8-bit ADC with 4 or 8 input channels (except on ST6208C)



PDIP20

SO20

SSOP20



CDIP20W

(See Section 11.5 for Ordering Information)

- **Instruction Set**
  - 8-bit data manipulation
  - 40 basic instructions
  - 9 addressing modes
  - Bit manipulation
- **Development Tools**
  - Full hardware/software development package

Device Summary

Features	ST62T08C(OTP)/ ST6208C(ROM) ST62P08C(FASTROM)	ST62T09C(OTP)/ ST6209C (ROM) ST62P09C(FASTROM)	ST62T10C(OTP)/ ST6210C (ROM) ST62P10C(FASTROM)	ST62T20C(OTP) ST6220C(ROM) ST62P20C(FASTROM)	ST62E20C(EPROM)
Program memory - bytes	1K		2K	4K	
RAM - bytes	64				
Operating Supply	3.0V to 6V				
Analog Inputs	-	4	8		
Clock Frequency	8MHz Max				
Operating Temperature	-40°C to +125°C				
Packages	PDIP20/SO20/SSOP20				CDIP20W

Rev. 3.2

Table of Contents

1 INTRODUCTION 6

2 PIN DESCRIPTION 7

3 MEMORY MAPS, PROGRAMMING MODES AND OPTION BYTES 9

3.1 MEMORY AND REGISTER MAPS 9

3.1.1 Introduction 9

3.1.2 Program Space 11

3.1.3 Readout Protection 11

3.1.4 Data Space 11

3.1.5 Stack Space 11

3.1.6 Data ROM Window 13

3.2 PROGRAMMING MODES 15

3.2.1 Program Memory 15

3.2.2 EPROM Erasing 15

3.3 OPTION BYTES 16

4 CENTRAL PROCESSING UNIT 17

4.1 INTRODUCTION 17

4.2 MAIN FEATURES 17

4.3 CPU REGISTERS 17

5 CLOCKS, SUPPLY AND RESET 19

5.1 CLOCK SYSTEM 19

5.1.1 Main Oscillator 20

5.1.2 Oscillator Safeguard (OSG) 21

5.1.3 Low Frequency Auxiliary Oscillator (LFAO) 22

5.1.4 Register Description 22

5.2 LOW VOLTAGE DETECTOR (LVD) 23

5.3 RESET 24

5.3.1 Introduction 24

5.3.2 RESET Sequence 24

5.3.3 $\overline{\text{RESET}}$ Pin 25

5.3.4 Watchdog Reset 26

5.3.5 LVD Reset 26

5.4 INTERRUPTS 27

5.5 INTERRUPT RULES AND PRIORITY MANAGEMENT 28

5.6 INTERRUPTS AND LOW POWER MODES 28

5.7 NON MASKABLE INTERRUPT 28

5.8 PERIPHERAL INTERRUPTS 28

5.9 EXTERNAL INTERRUPTS (I/O PORTS) 29

5.9.1 Notes on using External Interrupts 29

5.10 INTERRUPT HANDLING PROCEDURE 30

5.10.1 Interrupt Response Time 30

5.11 REGISTER DESCRIPTION 31

Table of Contents

6 POWER SAVING MODES . . . 32
6.1 INTRODUCTION . . . 32
6.2 WAIT MODE . . . 33
6.3 STOP MODE . . . 34
6.4 NOTES RELATED TO WAIT AND STOP MODES . . . 36
6.4.1 Exit from Wait and Stop Modes . . . 36
6.4.2 Recommended MCU Configuration . . . 36
7 I/O PORTS . . . 37
7.1 INTRODUCTION . . . 37
7.2 FUNCTIONAL DESCRIPTION . . . 37
7.2.1 Digital Input Modes . . . 37
7.2.2 Analog Inputs . . . 37
7.2.3 Output Modes . . . 37
7.2.4 Alternate Functions . . . 37
7.2.5 Instructions NOT to be used to access Port Data registers (SET, RES, INC and DEC) . . . 39
7.2.6 Recommendations . . . 39
7.3 LOW POWER MODES . . . 39
7.4 INTERRUPTS . . . 39
7.5 REGISTER DESCRIPTION . . . 41
8 ON-CHIP PERIPHERALS . . . 42
8.1 WATCHDOG TIMER (WDG) . . . 42
8.1.1 Introduction . . . 42
8.1.2 Main Features . . . 42
8.1.3 Functional Description . . . 43
8.1.4 Recommendations . . . 43
8.1.5 Low Power Modes . . . 44
8.1.6 Interrupts . . . 44
8.1.7 Register Description . . . 45
8.2 8-BIT TIMER . . . 46
8.2.1 Introduction . . . 46
8.2.2 Main Features . . . 46
8.2.3 Counter/Prescaler Description . . . 47
8.2.4 Functional Description . . . 48
8.2.5 Low Power Modes . . . 50
8.2.6 Interrupts . . . 50
8.2.7 Register Description . . . 51
8.3 A/D CONVERTER (ADC) . . . 52
8.3.1 Introduction . . . 52
8.3.2 Main Features . . . 52
8.3.3 Functional Description . . . 53
8.3.4 Recommendations . . . 54
8.3.5 Low Power Modes . . . 55
8.3.6 Interrupts . . . 55
8.3.7 Register Description . . . 55

Table of Contents

9 INSTRUCTION SET . . . 56
9.1 ST6 ARCHITECTURE . . . 56
9.2 ADDRESSING MODES . . . 56
9.3 INSTRUCTION SET . . . 57

10 ELECTRICAL CHARACTERISTICS . . . 62
10.1 PARAMETER CONDITIONS . . . 62
10.1.1 Minimum and Maximum Values . . . 62
10.1.2 Typical Values . . . 62
10.1.3 Typical Curves . . . 62
10.1.4 Loading Capacitor . . . 62
10.1.5 Pin Input Voltage . . . 62
10.2 ABSOLUTE MAXIMUM RATINGS . . . 63
10.2.1 Voltage Characteristics . . . 63
10.2.2 Current Characteristics . . . 63
10.2.3 Thermal Characteristics . . . 63
10.3 OPERATING CONDITIONS . . . 64
10.3.1 General Operating Conditions . . . 64
10.3.2 Operating Conditions with Low Voltage Detector (LVD) . . . 65
10.4 SUPPLY CURRENT CHARACTERISTICS . . . 66
10.4.1 RUN Modes . . . 66
10.4.2 WAIT Modes . . . 67
10.4.3 STOP Mode . . . 70
10.4.4 Supply and Clock System . . . 71
10.4.5 On-Chip Peripherals . . . 71
10.5 CLOCK AND TIMING CHARACTERISTICS . . . 72
10.5.1 General Timings . . . 72
10.5.2 External Clock Source . . . 72
10.5.3 Crystal and Ceramic Resonator Oscillators . . . 73
10.5.4 RC Oscillator . . . 74
10.5.5 Oscillator Safeguard (OSG) and Low Frequency Auxiliary Oscillator (LFAO) . . . 75
10.6 MEMORY CHARACTERISTICS . . . 76
10.6.1 RAM and Hardware Registers . . . 76
10.6.2 EPROM Program Memory . . . 76
10.7 EMC CHARACTERISTICS . . . 77
10.7.1 Functional EMS . . . 77
10.7.2 Absolute Electrical Sensitivity . . . 78
10.7.3 ESD Pin Protection Strategy . . . 80
10.8 I/O PORT PIN CHARACTERISTICS . . . 81
10.8.1 General Characteristics . . . 81
10.8.2 Output Driving Current . . . 82
10.9 CONTROL PIN CHARACTERISTICS . . . 85
10.9.1 Asynchronous RESET Pin . . . 85
10.9.2 NMI Pin . . . 86
10.10 TIMER PERIPHERAL CHARACTERISTICS . . . 87
10.10.1 Watchdog Timer . . . 87
10.10.2 8-Bit Timer . . . 87

Table of Contents

10.11 8-BIT ADC CHARACTERISTICS 88

11 GENERAL INFORMATION 90

11.1 PACKAGE MECHANICAL DATA 90

11.2 THERMAL CHARACTERISTICS 92

11.3 SOLDERING AND GLUEABILITY INFORMATION 93

11.4 PACKAGE/SOCKET FOOTPRINT PROPOSAL 94

11.5 ORDERING INFORMATION 95

11.6 TRANSFER OF CUSTOMER CODE 96

11.6.1FASTROM version 96

11.6.2ROM VERSION 98

12 DEVELOPMENT TOOLS 99

13 ST6 APPLICATION NOTES 101

14 SUMMARY OF CHANGES 103

15 TO GET MORE INFORMATION 103

1 INTRODUCTION

The ST6208C, 09C, 10C and 20C devices are low cost members of the ST62xx 8-bit HCMOS family of microcontrollers, which is targeted at low to medium complexity applications. All ST62xx devices are based on a building block approach: a common core is surrounded by a number of on-chip peripherals.

The ST62E20C is the erasable EPROM version of the ST62T08C, T09C, T10C and T20C devices, which may be used during the development phase for the ST62T08C, T09C, T10C and T20C target devices, as well as the respective ST6208C, 09C, 10C and 20C ROM devices.

OTP and EPROM devices are functionally identical. OTP devices offer all the advantages of user programmability at low cost, which make them the ideal choice in a wide range of applications where frequent code changes, multiple code versions or last minute programmability are required.

The ROM based versions offer the same functionality, selecting the options defined in the programmable option bytes of the OTP/EPROM versions in the ROM option list (See Section 11.6 on page 96).

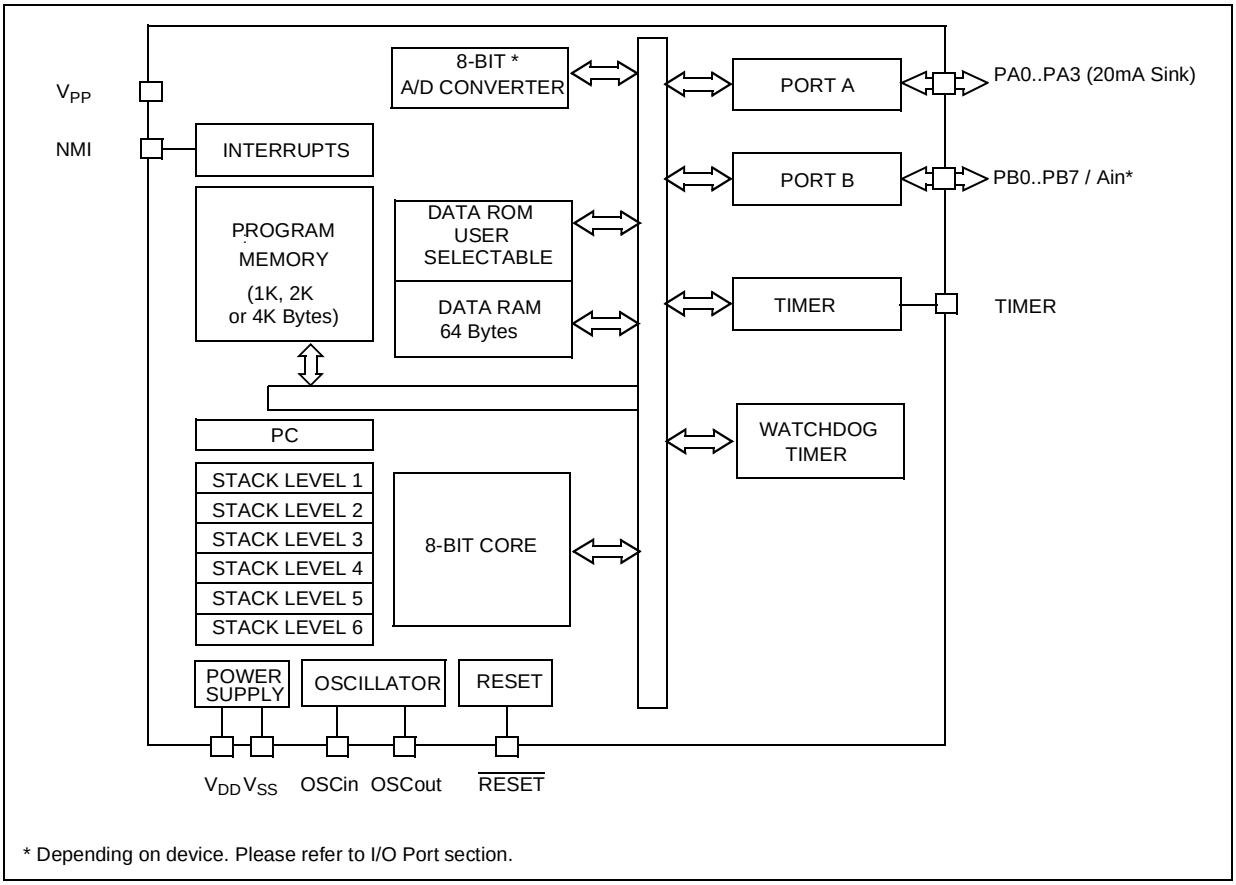
The ST62P08C/P09C/P10C/P20C are the **F**actory **A**dvanced **S**ervice **T**echnique ROM (FASTROM) versions of ST62T08C, T09C, T10C and T20C OTP devices.

They offer the same functionality as OTP devices, but they do not have to be programmed by the customer (See Section 11 on page 90).

These compact low-cost devices feature a Timer comprising an 8-bit counter with a 7-bit programmable prescaler, an 8-bit A/D Converter with up to 8 analog inputs (depending on device) and a Digital Watchdog timer, making them well suited for a wide range of automotive, appliance and industrial applications.

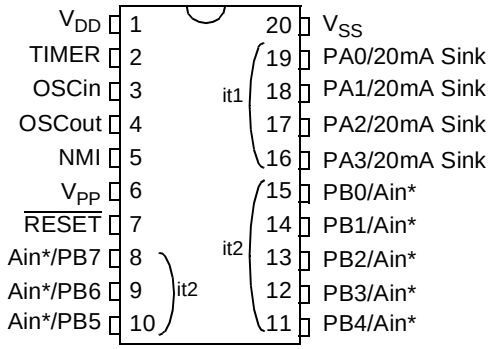
For easy reference, all parametric data are located in Section 11 on page 90.

Figure 1. Block Diagram



2 PIN DESCRIPTION

Figure 2. 20-Pin Package Pinout



itX associated interrupt vector
* Depending on device. Please refer to I/O Port section.

Table 1. Device Pin Description

Pin n°	Pin Name	Type	Main Function (after Reset)	Alternate Function
1	V_{DD}	S	Main power supply	
2	TIMER	I/O	Timer input or output	
3	OSCin	I	External clock input or resonator oscillator inverter input	
4	OSCout	O	Resonator oscillator inverter output or resistor input for RC oscillator	
5	NMI	I	Non maskable interrupt (falling edge sensitive)	
6	V_{PP}		Must be held at Vss for normal operation, if a 12.5V level is applied to the pin during the reset phase, the device enters EPROM programming mode.	
7	$\overline{RESET}$	I/O	Top priority non maskable interrupt (active low)	
8	PB7/Ain*	I/O	Pin B7 (IPU)	Analog input
9	PB6/Ain*	I/O	Pin B6 (IPU)	Analog input
10	PB5/Ain*	I/O	Pin B5 (IPU)	Analog input
11	PB4/Ain*	I/O	Pin B4 (IPU)	Analog input
12	PB3/Ain*	I/O	Pin B3 (IPU)	Analog input
13	PB2/Ain*	I/O	Pin B2 (IPU)	Analog input
14	PB1/Ain*	I/O	Pin B1 (IPU)	Analog input
15	PB0/Ain*	I/O	Pin B0 (IPU)	Analog input
16	PA3/ 20mA Sink	I/O	Pin A3 (IPU)	
17	PA2/ 20mA Sink	I/O	Pin A2 (IPU)	
18	PA1/ 20mA Sink	I/O	Pin A1 (IPU)	

Pin n°	Pin Name	Type	Main Function (after Reset)	Alternate Function
19	PA0/ 20mA Sink	I/O	Pin A0 (IPU)	
20	V_{SS}	S	Ground	

Legend / Abbreviations for Table 1**:**

* Depending on device. Please refer to Section 7 "I/O PORTS" on page 37.

I = input, O = output, S = supply, IPU = input with pull-up

The input with pull-up configuration (reset state) is valid as long as the user software does not change it. Refer to Section 7 "I/O PORTS" on page 37 for more details on the software configuration of the I/O ports.

3 MEMORY MAPS, PROGRAMMING MODES AND OPTION BYTES

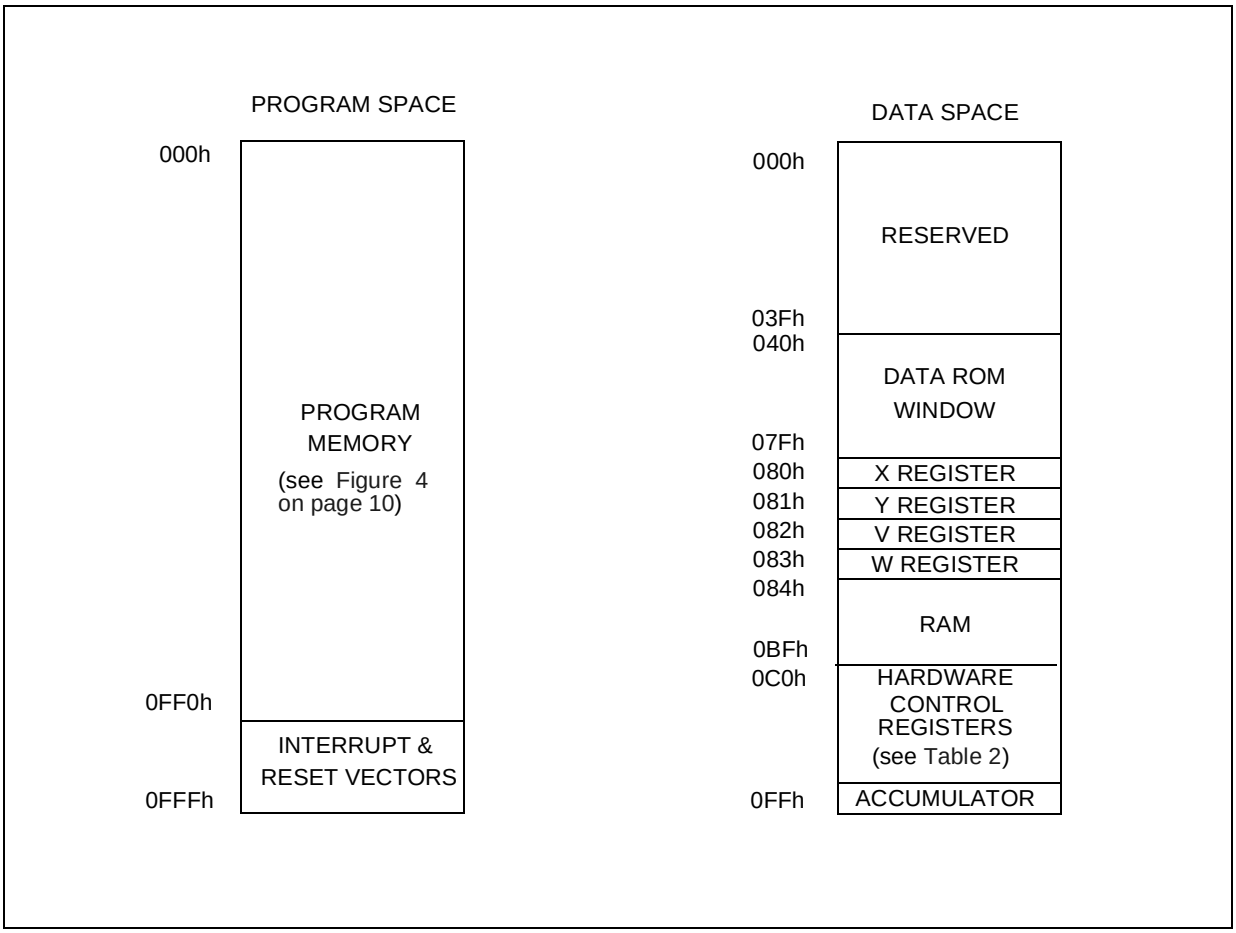
3.1 MEMORY AND REGISTER MAPS

3.1.1 Introduction

The MCU operates in three separate memory spaces: Program space, Data space, and Stack space. Operation in these three memory spaces is described in the following paragraphs.

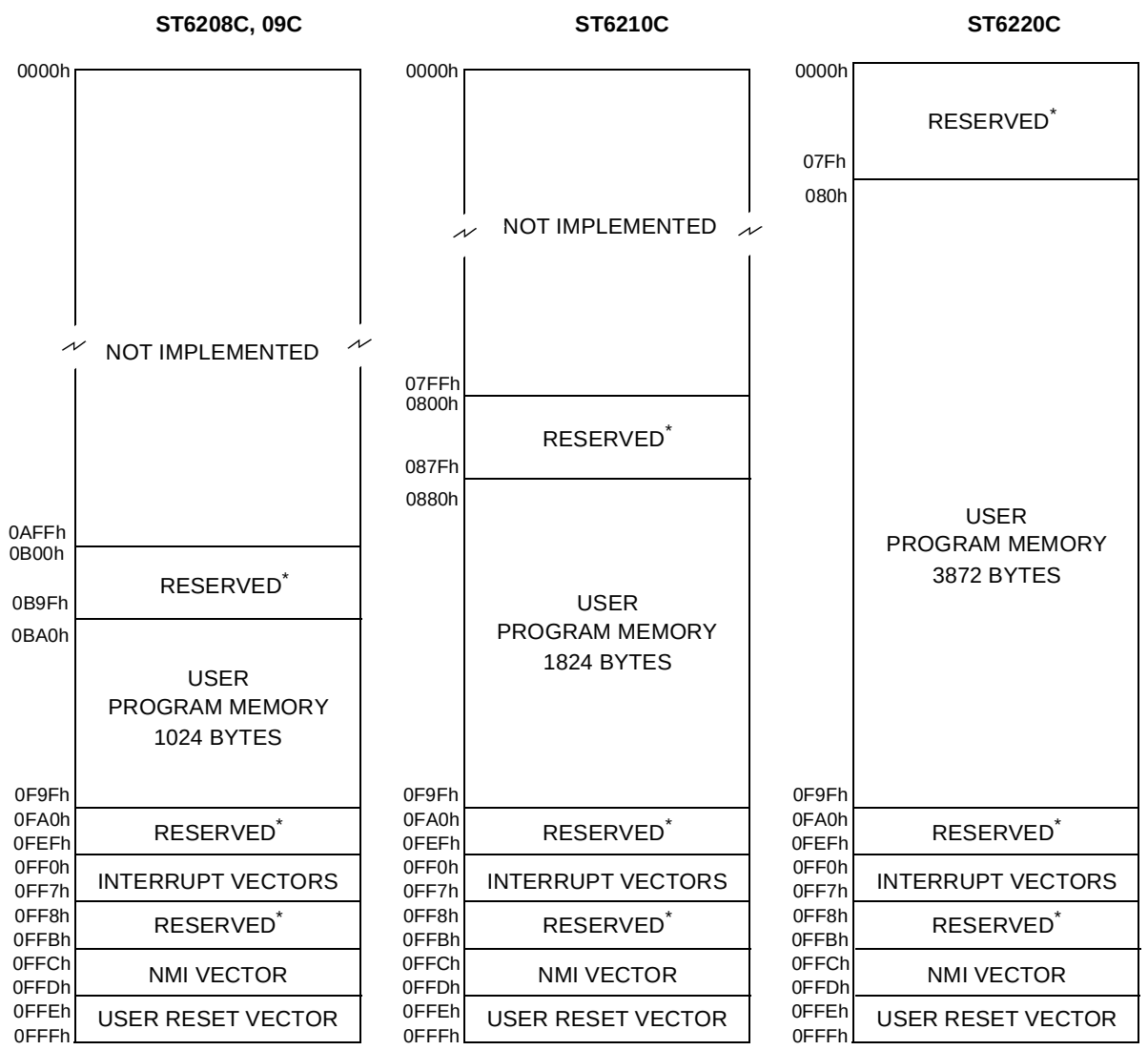
Briefly, Program space contains user program code in OTP and user vectors; Data space contains user data in RAM and in OTP, and Stack space accommodates six levels of stack for subroutine and interrupt service routine nesting.

Figure 3. Memory Addressing Diagram



MEMORY MAP (Cont'd)

Figure 4. Program Memory Map



(*) Reserved areas should be filled with 0FFh

MEMORY MAP (Cont'd)

3.1.2 Program Space

Program Space comprises the instructions to be executed, the data required for immediate addressing mode instructions, the reserved factory test area and the user vectors. Program Space is addressed via the 12-bit Program Counter register (PC register). Thus, the MCU is capable of addressing 4K bytes of memory directly.

3.1.3 Readout Protection

The Program Memory in OTP, EPROM or ROM devices can be protected against external readout of memory by setting the Readout Protection bit in the option bytes (Section 3.3 on page 16).

In the EPROM parts, Readout Protection option can be desactivated only by U.V. erasure that also results in the whole EPROM context being erased.

Note: Once the Readout Protection is activated, it is no longer possible, even for STMicroelectronics, to gain access to the OTP or ROM contents. Returned parts can therefore not be accepted if the Readout Protection bit is set.

3.1.4 Data Space

Data Space accommodates all the data necessary for processing the user program. This space comprises the RAM resource, the processor core and peripheral registers, as well as read-only data such as constants and look-up tables in OTP/EPROM.

3.1.4.1 Data ROM

All read-only data is physically stored in program memory, which also accommodates the Program Space. The program memory consequently contains the program code to be executed, as well as the constants and look-up tables required by the application.

The Data Space locations in which the different constants and look-up tables are addressed by the processor core may be thought of as a 64-byte window through which it is possible to access the read-only data stored in OTP/EPROM.

3.1.4.2 Data RAM

The data space includes the user RAM area, the accumulator (A), the indirect registers (X), (Y), the short direct registers (V), (W), the I/O port registers, the peripheral data and control registers, the interrupt option register and the Data ROM Window register (DRWR register).

3.1.5 Stack Space

Stack space consists of six 12-bit registers which are used to stack subroutine and interrupt return addresses, as well as the current program counter contents.

MEMORY MAP (Cont'd)

Table 2. Hardware Register Map

Address	Block	Register Label	Register Name	Reset Status	Remarks
080h to 083h	CPU	X,Y,V,W	X,Y index registers V,W short direct registers	xxh	R/W
0C0h 0C1h	I/O Ports	DRA [1) 2) 3)] DRB [1) 2) 3)]	Port A Data Register Port B Data Register	00h 00h	R/W R/W
0C2h 0C3h	Reserved (2 Bytes)				
0C4h 0C5h	I/O Ports	DDRA [2)] DDRB [2)]	Port A Direction Register Port B Direction Register	00h 00h	R/W R/W
0C6h 0C7h	Reserved (2 Bytes)				
0C8h	CPU	IOR	Interrupt Option Register	xxh	Write-only
0C9h	ROM	DRWR	Data ROM Window register	xxh	Write-only
0CAh 0CBh	Reserved (2 Bytes)				
0CCh 0CDh	I/O Ports	ORA [2)] ORB [2)]	Port A Option Register Port B Option Register	00h 00h	R/W R/W
0CEh 0CFh	Reserved (2 bytes)				
0D0h 0D1h	ADC [4)]	ADR ADCR	A/D Converter Data Register A/D Converter Control Register	xxh 40h	Read-only Ro/Wo
0D2h 0D3h 0D4h	Timer1	PSCR TCR TSCR	Timer 1 Prescaler Register Timer 1 Downcounter Register Timer 1 Status Control Register	7Fh 0FFh 00h	R/W R/W R/W
0D5h to 0D7h	Reserved (3 Bytes)				
0D8h	Watchdog Timer	WDGR	Watchdog Register	0FEh	R/W
0D9h to 0FEh	Reserved (38 Bytes)				
0FFh	CPU	A	Accumulator	xxh	R/W

Legend:
x = undefined, R/W = Read/Write, Ro = Read-only Bit(s) in the register, Wo = Write-only Bit(s) in the register.

Notes:
1. The contents of the I/O port DR registers are readable only in output configuration. In input configuration, the values of the I/O pins are returned instead of the DR register contents.
2. The bits associated with unavailable pins must always be kept at their reset value.
3. Do not use single-bit instructions (SET, RES...) on Port Data Registers if any pin of the port is configured in input mode (refer to Section 7 "I/O PORTS" on page 37 for more details)
4. Depending on device. See device summary on page 1.

MEMORY MAP (Cont'd)

3.1.6 Data ROM Window

The Data read-only memory window is located from address 0040h to address 007Fh in Data space. It allows direct reading of 64 consecutive bytes located anywhere in program memory, between address 0000h and 0FFFh.

There are 64 blocks of 64 bytes in a 4K device:

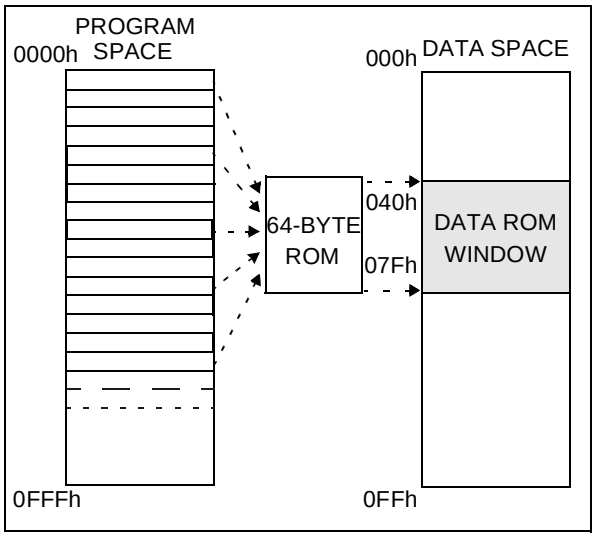
– Block 0 is related to the address range 0000h to 003Fh.

– Block 1 is related to the address range 0040h to 007Fh.

and so on...

All the program memory can therefore be used to store either instructions or read-only data. The Data ROM window can be moved in steps of 64 bytes along the program memory by writing the appropriate code in the Data ROM Window Register (DRWR).

Figure 5. Data ROM Window



3.1.6.1 Data ROM Window Register (DRWR)

The DRWR can be addressed like any RAM location in the Data Space.

This register is used to select the 64-byte block of program memory to be read in the Data ROM window (from address 40h to address 7Fh in Data space). The DRWR register is not cleared on reset, therefore it must be written to before accessing the Data read-only memory window area for the first time.

Address: 0C9h — Write Only

Reset Value = xxh (undefined)

7							0
-	-	DRWR5	DRWR4	DRWR3	DRWR2	DRWR1	DRWR0

Bits 7:6 = **Reserved**, must be cleared.

Bit 5:0 = **DRWR[5:0]** *Data read-only memory Window Register Bits.* These are the Data read-only memory Window bits that correspond to the upper bits of the data read-only memory space.

Caution: This register is undefined on reset, it is write-only, therefore do not read it nor access it using Read-Modify-Write instructions (SET, RES, INC and DEC).

MEMORY MAP (Cont'd)

3.1.6.2 Data ROM Window memory addressing

In cases where some data (look-up tables for example) are stored in program memory, reading these data requires the use of the Data ROM window mechanism. To do this:

1. The DRWR register has to be loaded with the 64-byte block number where the data are located (in program memory). This number also gives the start address of the block.

2. Then, the offset address of the byte in the Data ROM Window (corresponding to the offset in the 64-byte block in program memory) has to be loaded in a register (A, X,...).

When the above two steps are completed, the data can be read.

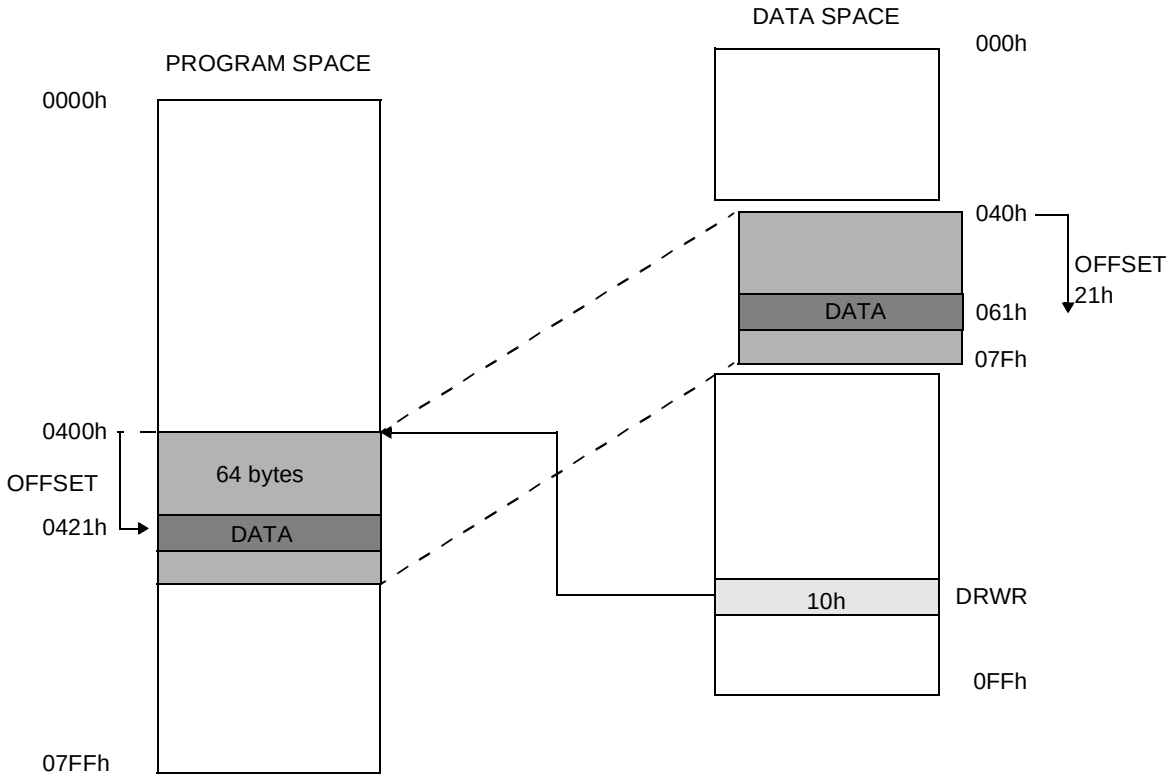
To understand how to determine the DRWR and the content of the register, please refer to the example shown in Figure 6. In any case the calculation is automatically handled by the ST6 development tools.

Please refer to the user manual of the correspoding tool.

3.1.6.3 Recommendations

Care is required when handling the DRWR register as it is write only. For this reason, the DRWR contents should not be changed while executing an interrupt service routine, as the service routine cannot save and then restore the register's previous contents. If it is impossible to avoid writing to the DRWR during the interrupt service routine, an image of the register must be saved in a RAM location, and each time the program writes to the DRWR, it must also write to the image register. The image register must be written first so that, if an interrupt occurs between the two instructions, the DRWR is not affected.

Figure 6. Data ROM Window Memory Addressing



DATA address in Program memory : 421h
DRWR content : 421h / 3Fh (64) = 10H data is located in 64-bytes window number 10h
64-byte window start address : 10h x 3Fh = 400h
Register (A, X,...)content : Offset = (421h - 400h) + 40h (Data ROM Window start address in data space) = 61h

3.2 PROGRAMMING MODES

3.2.1 Program Memory

EPROM/OTP programming mode is set by a +12.5V voltage applied to the TEST/V_{PP} pin. The programming flow of the ST62T08C,T09C,T10C, T20C and E20C is described in the User Manual of the EPROM Programming Board.

Table 3. ST6208C/09C Program Memory Map

Device Address	Description
0000h-0B9Fh	Reserved
0BA0h-0F9Fh	User ROM
0FA0h-0FEFh	Reserved
0FF0h-0FF7h	Interrupt Vectors
0FF8h-0FFBh	Reserved
0FFCh-0FFDh	NMI Interrupt Vector
0FFEh-0FFFh	Reset Vector

Table 4. ST6210C Program Memory Map

Device Address	Description
0000h-087Fh	Reserved
0880h-0F9Fh	User ROM
0FA0h-0FEFh	Reserved
0FF0h-0FF7h	Interrupt Vectors
0FF8h-0FFBh	Reserved
0FFCh-0FFDh	NMI Interrupt Vector
0FFEh-0FFFh	Reset Vector

Table 5. ST6220C Program Memory Map

Device Address	Description
0000h-007Fh	Reserved
0080h-0F9Fh	User ROM
0FA0h-0FEFh	Reserved
0FF0h-0FF7h	Interrupt Vectors
0FF8h-0FFBh	Reserved
0FFCh-0FFDh	NMI Interrupt Vector
0FFEh-0FFFh	Reset Vector

Note: OTP/EPROM devices can be programmed with the development tools available from STMicroelectronics (please refer to Section 12 on page 99).

3.2.2 EPROM Erasing

The EPROM devices can be erased by exposure to Ultra Violet light. The characteristics of the MCU are such that erasure begins when the memory is exposed to light with a wave lengths shorter than approximately 4000Å. It should be noted that sunlight and some types of fluorescent lamps have wavelengths in the range 3000-4000Å.

It is thus recommended that the window of the MCU packages be covered by an opaque label to prevent unintentional erasure problems when testing the application in such an environment.

The recommended erasure procedure is exposure to short wave ultraviolet light which have a wavelength 2537Å. The integrated dose (i.e. U.V. intensity x exposure time) for erasure should be a minimum of 30W-sec/cm^2. The erasure time with this dosage is approximately 30 to 40 minutes using an ultraviolet lamp with 12000µW/cm^2 power rating. The EPROM device should be placed within 2.5cm (1inch) of the lamp tubes during erasure.

3.3 OPTION BYTES

Each device is available for production in user programmable versions (OTP) as well as in factory coded versions (ROM). OTP devices are shipped to customers with a default content (00h), while ROM factory coded parts contain the code supplied by the customer. This implies that OTP devices have to be configured by the customer using the Option Bytes while the ROM devices are factory-configured.

The two option bytes allow the hardware configuration of the microcontroller to be selected.
The option bytes have no address in the memory map and can be accessed only in programming mode (for example using a standard ST6 programming tool).
In masked ROM devices, the option bytes are fixed in hardware by the ROM code (see Section 11.6.2 "ROM VERSION" on page 98). It is therefore impossible to read the option bytes.

The option bytes can be only programmed once. It is not possible to change the selected options after they have been programmed.

In order to reach the power consumption value indicated in Section 10.4, the option byte must be programmed to its default value. Otherwise, an over-consumption will occur.

MSB OPTION BYTE

Bits 15:10 = **Reserved**, must be always cleared.

Bit 9 = **EXTCNTL** *External STOP MODE control.*
0: EXTCNTL mode not available. STOP mode is not available with the watchdog active.
1: EXTCNTL mode available. STOP mode is available with the watchdog active by setting NMI pin to one.

Bit 8 = **LVD** *Low Voltage Detector* on/off.
This option bit enable or disable the Low Voltage Detector (LVD) feature.
0: Low Voltage Detector disabled
1: Low Voltage Detector enabled

LSB OPTION BYTE

Bit 7 = **PROTECT** *Readout Protection.*
This option bit enables or disables external access to the internal program memory.
0: Program memory not read-out protected
1: Program memory read-out protected

Bit 6 = **OSC** *Oscillator selection*.
This option bit selects the main oscillator type.
0: Quartz crystal, ceramic resonator or external clock
1: RC network

Bit 5 = **Reserved**, must be always cleared.

Bit 4 = **Reserved**, must be always set.

Bit 3 = **NMI PULL** *NMI Pull-Up* on/off.
This option bit enables or disables the internal pull-up on the NMI pin.
0: Pull-up disabled
1: Pull-up enabled

Bit 2 = **TIM PULL** *TIMER Pull-Up* on/off.
This option bit enables or disables the internal pull-up on the TIMER pin.
0: Pull-up disabled
1: Pull-up enabled

Bit 1 = **WDACT** *Hardware or software watchdog.*
This option bit selects the watchdog type.
0: Software (watchdog to be enabled by software)
1: Hardware (watchdog always enabled)

Bit 0 = **OSGEN** *Oscillator Safeguard* on/off.
This option bit enables or disables the oscillator Safeguard (OSG) feature.
0: Oscillator Safeguard disabled
1: Oscillator Safeguard enabled

	MSB OPTION BYTE								LSB OPTION BYTE							
	15							8	7							0
	Reserved						EXT CTL	LVD	PRO-TECT	OSC	Res.	Res.	NMI PULL	TIM PULL	WD ACT	OSG EN
Default Value	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

4 CENTRAL PROCESSING UNIT

4.1 INTRODUCTION

The CPU Core of ST6 devices is independent of the I/O or Memory configuration. As such, it may be thought of as an independent central processor communicating with on-chip I/O, Memory and Peripherals via internal address, data, and control buses.

4.2 MAIN FEATURES

- 40 basic instructions
- 9 main addressing modes
- Two 8-bit index registers
- Two 8-bit short direct registers
- Low power modes
- Maskable hardware interrupts
- 6-level hardware stack

4.3 CPU REGISTERS

The ST6 Family CPU core features six registers and three pairs of flags available to the programmer. These are described in the following paragraphs.

Accumulator (A). The accumulator is an 8-bit general purpose register used in all arithmetic calculations, logical operations, and data manipulations. The accumulator can be addressed in Data Space as a RAM location at address FFh. Thus the ST6 can manipulate the accumulator just like any other register in Data Space.

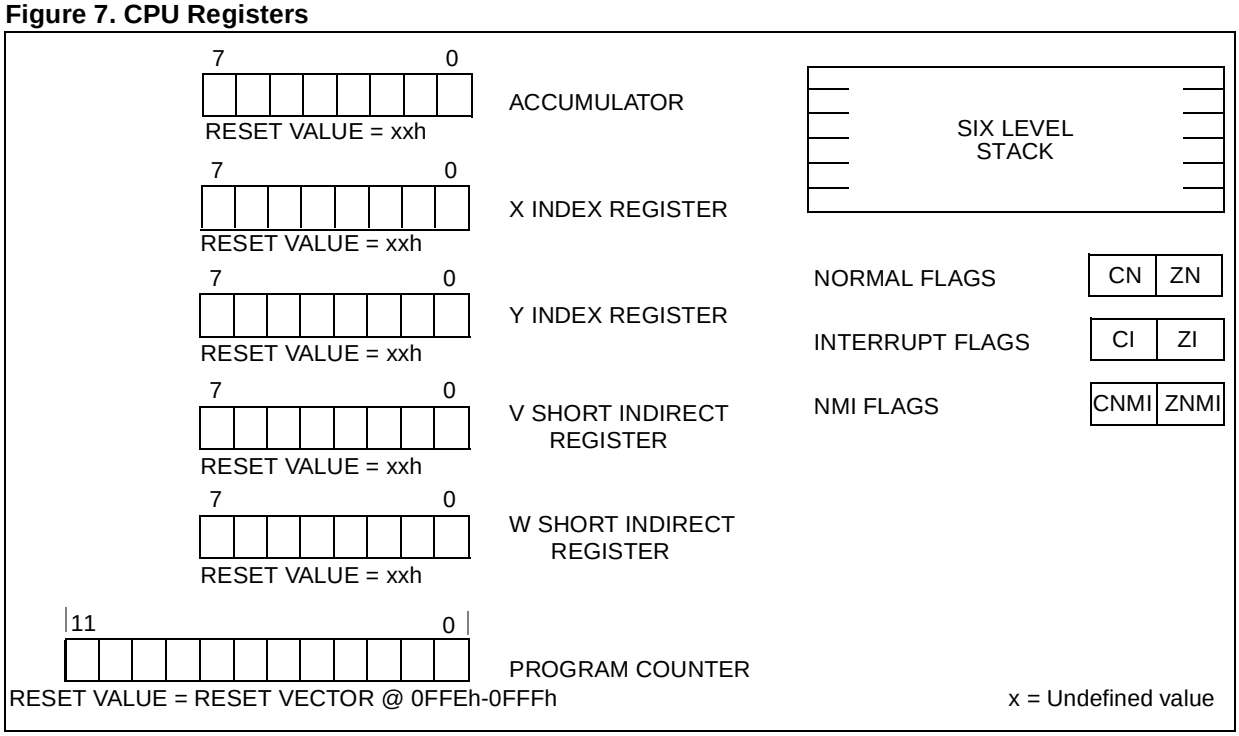
Index Registers (X, Y). These two registers are used in Indirect addressing mode as pointers to memory locations in Data Space. They can also be accessed in Direct, Short Direct, or Bit Direct addressing modes. They are mapped in Data Space at addresses 80h (X) and 81h (Y) and can be accessed like any other memory location.

Short Direct Registers (V, W). These two registers are used in Short Direct addressing mode. This means that the data stored in V or W can be accessed with a one-byte instruction (four CPU cycles). V and W can also be accessed using Direct and Bit Direct addressing modes. They are mapped in Data Space at addresses 82h (V) and 83h (W) and can be accessed like any other memory location.

Note: The X and Y registers can also be used as Short Direct registers in the same way as V and W.

Program Counter (PC). The program counter is a 12-bit register which contains the address of the next instruction to be executed by the core. This ROM location may be an opcode, an operand, or the address of an operand.

Figure 7. CPU Registers



CPU REGISTERS (Cont'd)

The 12-bit length allows the direct addressing of 4096 bytes in Program Space.

However, if the program space contains more than 4096 bytes, the additional memory in program space can be addressed by using the Program ROM Page register.

The PC value is incremented after reading the address of the current instruction. To execute relative jumps, the PC and the offset are shifted through the ALU, where they are added; the result is then shifted back into the PC. The program counter can be changed in the following ways:

– JP (Jump) instruction	PC = Jump address
– CALL instruction	PC = Call address
– Relative Branch Instruction	PC = PC +/- offset
– Interrupt	PC = Interrupt vector
– Reset	PC = Reset vector
– RET & RETI instructions	PC = Pop (stack)
– Normal instruction	PC = PC + 1

Flags (C, Z). The ST6 CPU includes three pairs of flags (Carry and Zero), each pair being associated with one of the three normal modes of operation: Normal mode, Interrupt mode and Non Maskable Interrupt mode. Each pair consists of a CARRY flag and a ZERO flag. One pair (CN, ZN) is used during Normal operation, another pair is used during Interrupt mode (CI, ZI), and a third pair is used in the Non Maskable Interrupt mode (CNMI, ZNMI).

The ST6 CPU uses the pair of flags associated with the current mode: as soon as an interrupt (or a Non Maskable Interrupt) is generated, the ST6 CPU uses the Interrupt flags (or the NMI flags) instead of the Normal flags. When the RETI instruction is executed, the previously used set of flags is restored. It should be noted that each flag set can only be addressed in its own context (Non Maskable Interrupt, Normal Interrupt or Main routine). The flags are not cleared during context switching and thus retain their status.

C : Carry flag.

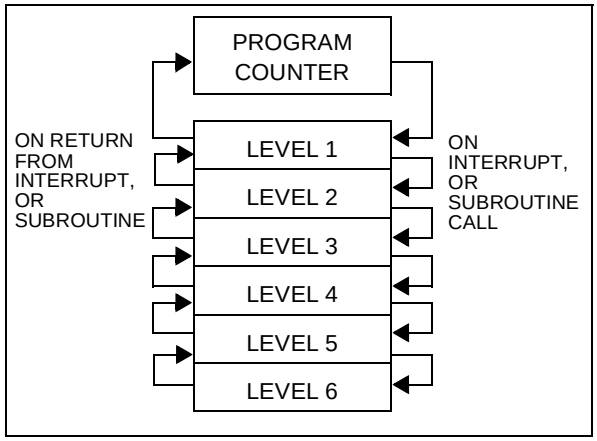
This bit is set when a carry or a borrow occurs during arithmetic operations; otherwise it is cleared. The Carry flag is also set to the value of the bit tested in a bit test instruction; it also participates in the rotate left instruction.
0: No carry has occured
1: A carry has occured

Z : Zero flag
This flag is set if the result of the last arithmetic or logical operation was equal to zero; otherwise it is cleared.
0: The result of the last operation is different from zero
1: The result of the last operation is zero

Switching between the three sets of flags is performed automatically when an NMI, an interrupt or a RETI instruction occurs. As NMI mode is automatically selected after the reset of the MCU, the ST6 core uses the NMI flags first.

Stack. The ST6 CPU includes a true LIFO (Last In First Out) hardware stack which eliminates the need for a stack pointer. The stack consists of six separate 12-bit RAM locations that do not belong to the data space RAM area. When a subroutine call (or interrupt request) occurs, the contents of each level are shifted into the next level down, while the content of the PC is shifted into the first level (the original contents of the sixth stack level are lost). When a subroutine or interrupt return occurs (RET or RETI instructions), the first level register is shifted back into the PC and the value of each level is popped back into the previous level.

Figure 8. Stack manipulation



Since the accumulator, in common with all other data space registers, is not stored in this stack, management of these registers should be performed within the subroutine.

Caution: The stack will remain in its "deepest" position if more than 6 nested calls or interrupts are executed, and consequently the last return address will be lost.

It will also remain in its highest position if the stack is empty and a RET or RETI is executed. In this case the next instruction will be executed.

5 CLOCKS, SUPPLY AND RESET

5.1 CLOCK SYSTEM

The main oscillator of the MCU can be driven by any of these clock sources:

– external clock signal

– external AT-cut parallel-resonant crystal

– external ceramic resonator

– external RC network (R_{NET}).

In addition, an on-chip Low Frequency Auxiliary Oscillator (LFAO) is available as a back-up clock system or to reduce power consumption.

An optional Oscillator Safeguard (OSG) filters spikes from the oscillator lines, and switches to the LFAO backup oscillator in the event of main oscillator failure. It also automatically limits the internal clock frequency (f_{INT}) as a function of V_{DD}, in order to guarantee correct operation. These functions are illustrated in Figure 10, and Figure 11.

Table 6 illustrates various possible oscillator configurations using an external crystal or ceramic resonator, an external clock input, an external resistor (R_{NET}), or the lowest cost solution using only the LFAO.

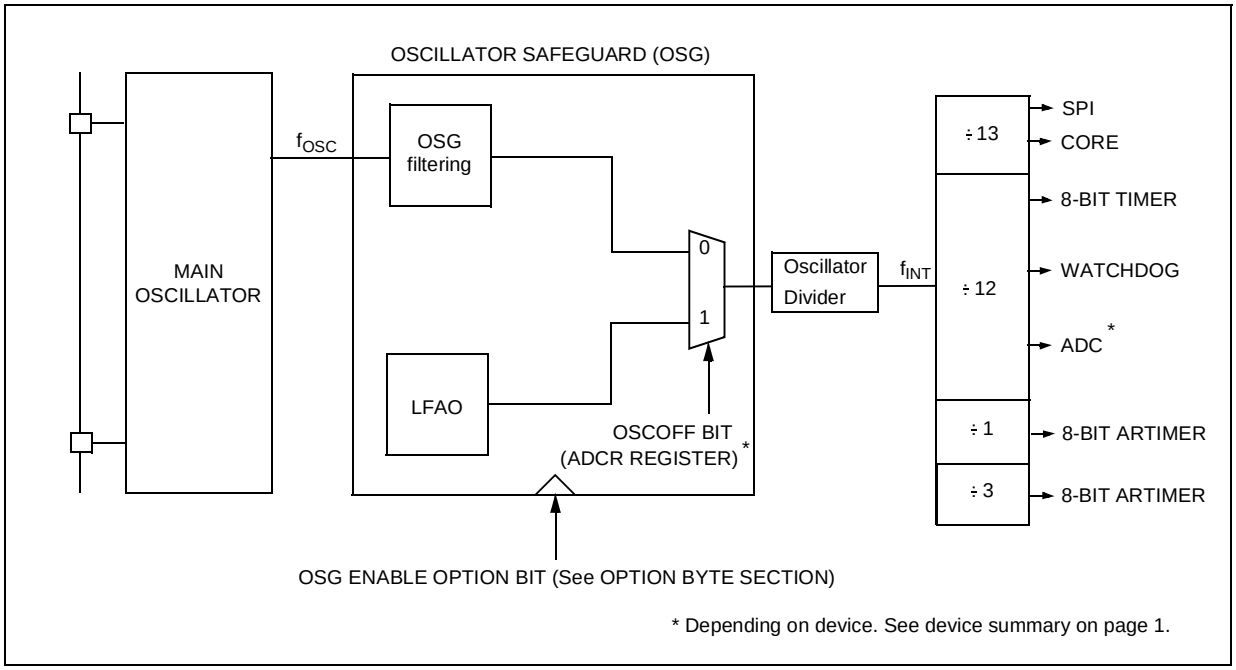
For more details on configuring the clock options, refer to the Option Bytes section of this document.

The internal MCU clock frequency (f_{INT}) is divided by 12 to drive the Timer, the Watchdog timer and the A/D converter, by 13 to drive the CPU core and the SPI and by 1 or 3 to drive the ARTIMER, as shown in Figure 9.

With an 8 MHz oscillator, the fastest CPU cycle is therefore 1.625µs.

A CPU cycle is the smallest unit of time needed to execute any operation (for instance, to increment the Program Counter). An instruction may require two, four, or five CPU cycles for execution.

Figure 9. Clock Circuit Block Diagram



CLOCK SYSTEM (Cont'd)

5.1.1 Main Oscillator

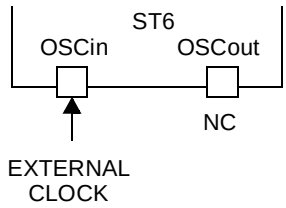
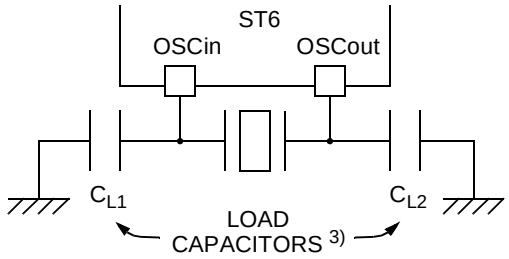
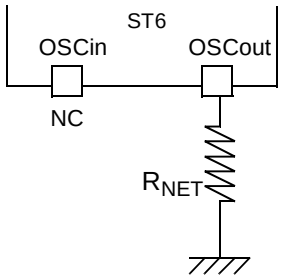
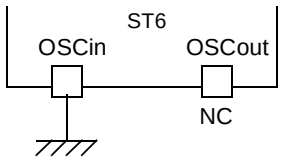
The oscillator configuration is specified by selecting the appropriate option in the option bytes (refer to the Option Bytes section of this document). When the CRYSTAL/RESONATOR option is selected, it must be used with a quartz crystal, a ceramic resonator or an external signal provided on the OSCin pin. When the RC NETWORK option is selected, the system clock is generated by an external resistor (the capacitor is implemented internally).

The main oscillator can be turned off (when the OSG ENABLED option is selected) by setting the OSCOFF bit of the ADC Control Register (not available on some devices). This will automatically start the Low Frequency Auxiliary Oscillator (LFAO).

The main oscillator can be turned off by resetting the OSCOFF bit of the A/D Converter Control Register or by resetting the MCU. When the main oscillator starts there is a delay made up of the oscillator start-up delay period plus the duration of the software instruction at a clock frequency f_{LFAO}.

Caution: It should be noted that when the RC network option is selected, the accuracy of the frequency is about 20% so it may not be suitable for some applications (For more details, please refer to the Electrical Characteristics Section).

Table 6. Oscillator Configurations

	Hardware Configuration
Crystal/Resonator Option[1]	**External Clock** 
Crystal/Resonator Option[1]	**Crystal/Resonator Clock [2]** 
RC Network Option[1]	**RC Network** 
OSG Enabled Option[1]	**LFAO** 

Notes:
1. To select the options shown in column 1 of the above table, refer to the Option Byte section.
2. This schematic are given for guidance only and are subject to the schematics given by the crystal or ceramic resonator manufacturer.
3. For more details, please refer to the Electrical Characteristics Section.

CLOCK SYSTEM (Cont'd)

5.1.2 Oscillator Safeguard (OSG)

The Oscillator Safeguard (OSG) feature is a means of dramatically improving the operational integrity of the MCU. It is available when the OSG ENABLED option is selected in the option byte (refer to the Option Bytes section of this document).

The OSG acts as a filter whose cross-over frequency is device dependent and provides three basic functions:

– Filtering spikes on the oscillator lines which would result in driving the CPU at excessive frequencies

– Management of the Low Frequency Auxiliary Oscillator (LFAO), (useable as low cost internal clock source, backup clock in case of main oscillator failure or for low power consumption)

– Automatically limiting the f_{INT} clock frequency as a function of supply voltage, to ensure correct operation even if the power supply drops.

5.1.2.1 Spike Filtering

Spikes on the oscillator lines result in an effectively increased internal clock frequency. In the absence of an OSG circuit, this may lead to an over frequency for a given power supply voltage. The OSG filters out such spikes (as illustrated in Figure 10). In all cases, when the OSG is active, the maximum internal clock frequency, f_{INT}, is limited to f_{OSG}, which is supply voltage dependent.

5.1.2.2 Management of Supply Voltage Variations

Over-frequency, at a given power supply level, is seen by the OSG as spikes; it therefore filters out some cycles in order that the internal clock frequency of the device is kept within the range the particular device can stand (depending on V_{DD}), and below f_{OSG}: the maximum authorised frequency with OSG enabled.

5.1.2.3 LFAO Management

When the OSG is enabled, the Low Frequency Auxiliary Oscillator can be used (see Section 5.1.3).

Note: The OSG should be used wherever possible as it provides maximum security for the application. It should be noted however, that it can increase power consumption and reduce the maximum operating frequency to f_{OSG} (see Electrical Characteristics section).

Caution: Care has to be taken when using the OSG, as the internal frequency is defined between a minimum and a maximum value and may vary depending on both V_{DD} and temperature. For precise timing measurements, it is not recommended to use the OSG.

Figure 10. OSG Filtering Function

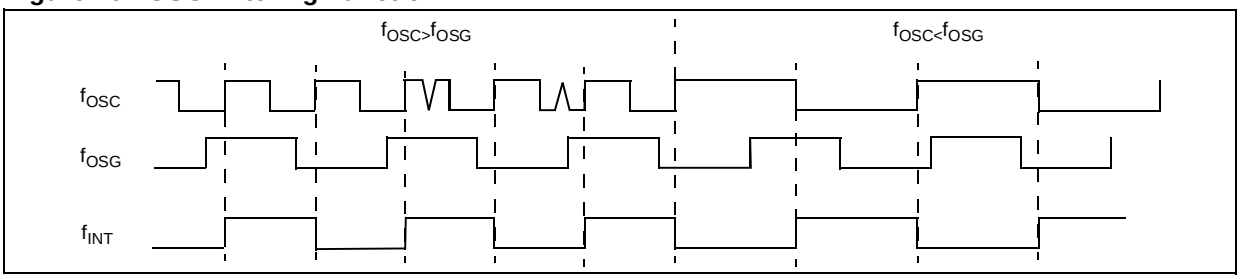
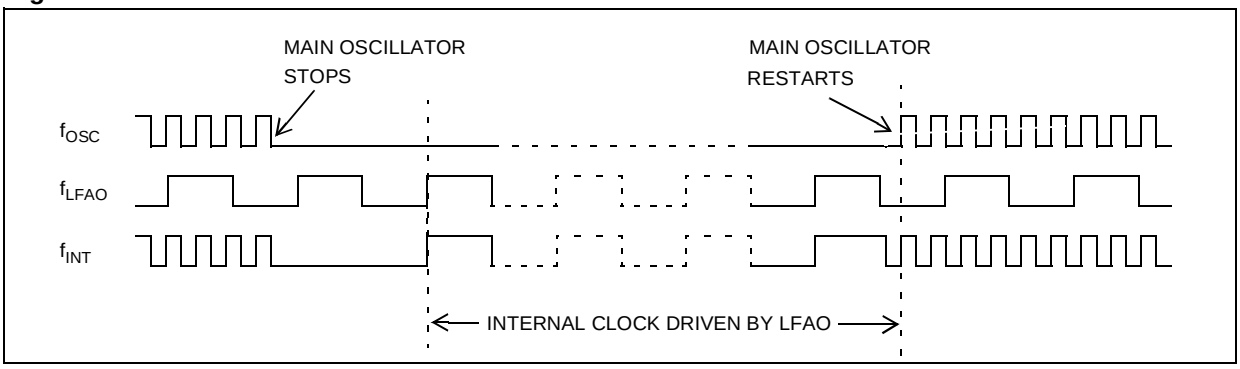


Figure 11. LFAO Oscillator Function



CLOCK SYSTEM (Cont'd)

5.1.3 Low Frequency Auxiliary Oscillator (LFAO)

The Low Frequency Auxiliary Oscillator has three main purposes. Firstly, it can be used to reduce power consumption in non timing critical routines. Secondly, it offers a fully integrated system clock, without any external components. Lastly, it acts as a backup oscillator in case of main oscillator failure.

This oscillator is available when the OSG ENABLED option is selected in the option byte (refer to the Option Bytes section of this document). In this case, it automatically starts one of its periods after the first missing edge of the main oscillator, whatever the reason for the failure (main oscillator defective, no clock circuitry provided, main oscillator switched off...). See Figure 11.

User code, normal interrupts, WAIT and STOP instructions, are processed as normal, at the reduced f_{LFAO} frequency. The A/D converter accuracy is decreased, since the internal frequency is below 1.2 MHz.

At power on, until the main oscillator starts, the reset delay counter is driven by the LFAO. If the main oscillator starts before the 2048 or 32768 cycle delay has elapsed, it takes over.

The Low Frequency Auxiliary Oscillator is automatically switched off as soon as the main oscillator starts.

5.1.4 Register Description

ADC CONTROL REGISTER (ADCR)
Address: 0D1h — Read/Write
Reset value: 0100 0000 (40h)

7							0
ADCR 7	ADCR 6	ADCR 5	ADCR 4	ADCR 3	OSC OFF	ADCR 1	ADCR 0

Bit 7:3, 1:0 = **ADCR[7:3], ADCR[1:0]** *ADC Control Register***.**
These bits are used to control the A/D converter (if available on the device) otherwise they are not used.

Bit 2 = **OSCOFF** *Main Oscillator Off.*
0: Main oscillator enabled
1: Main oscillator disabled

Note: The OSG must be enabled using the OSGEN option in the Option Byte, otherwise the OSCOFF setting has no effect.

5.2 LOW VOLTAGE DETECTOR (LVD)

The on-chip Low Voltage Detector is enabled by setting a bit in the option bytes (refer to the Option Bytes section of this document).

The LVD allows the device to be used without any external RESET circuitry. In this case, the $\overline{RESET}$ pin should be left unconnected.

If the LVD is not used, an external circuit is mandatory to ensure correct Power On Reset operation, see figure in the Reset section. For more details, please refer to the application note AN669.

The LVD generates a static Reset when the supply voltage is below a reference value. This means that it secures the power-up as well as the power-down keeping the ST6 in reset.

The V_{IT-} reference value for a voltage drop is lower than the V_{IT+} 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+} when V_{DD} is rising

– V_{IT-} when V_{DD} is falling

The LVD function is illustrated in Figure 12.

If the LVD is enabled, the MCU can be in only one of two states:

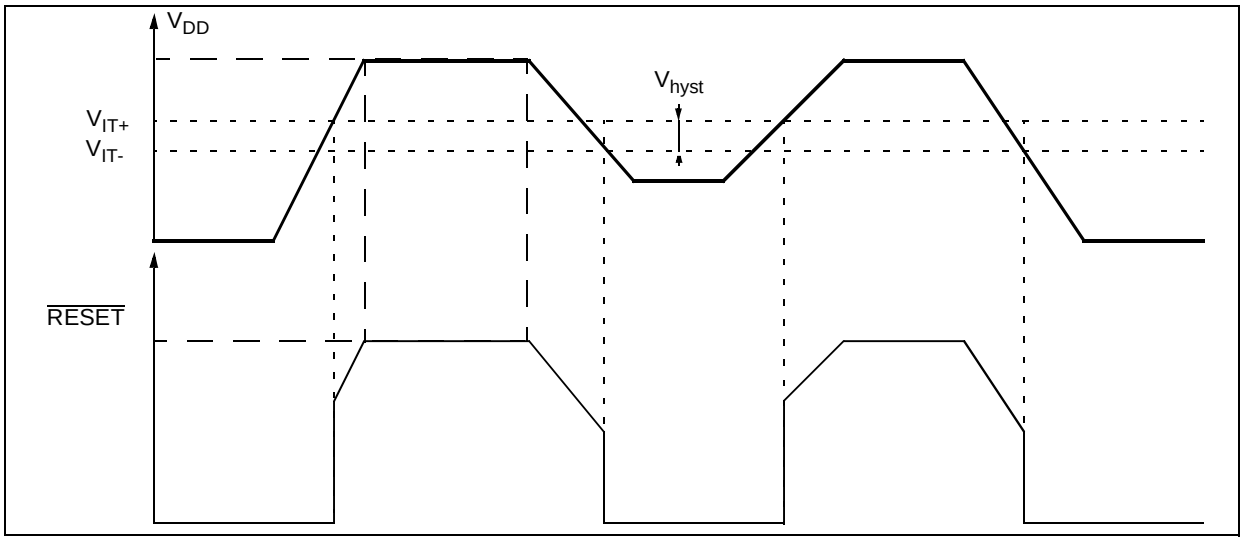
– Over the input threshold voltage, it is running under full software control

– Below the input threshold voltage, it is in static safe reset

In these conditions, secure operation is guaranteed without the need for external reset hardware.

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

Figure 12. Low Voltage Detector Reset



5.3 RESET

5.3.1 Introduction

The MCU can be reset in three ways:

- A low pulse input on the $\overline{\text{RESET}}$ pin
- Internal Watchdog reset
- Internal Low Voltage Detector (LVD) reset

5.3.2 RESET Sequence

The basic RESET sequence consists of 3 main phases:

- Internal (watchdog or LVD) or external Reset event
- A delay of 2048 or 32768 clock (f_{INT}) cycles (selected through the option bytes)
- RESET vector fetch

The reset delay allows the oscillator to stabilise and ensures that recovery has taken place from the Reset state.

The RESET vector fetch phase duration is 2 clock cycles.

When a reset occurs:

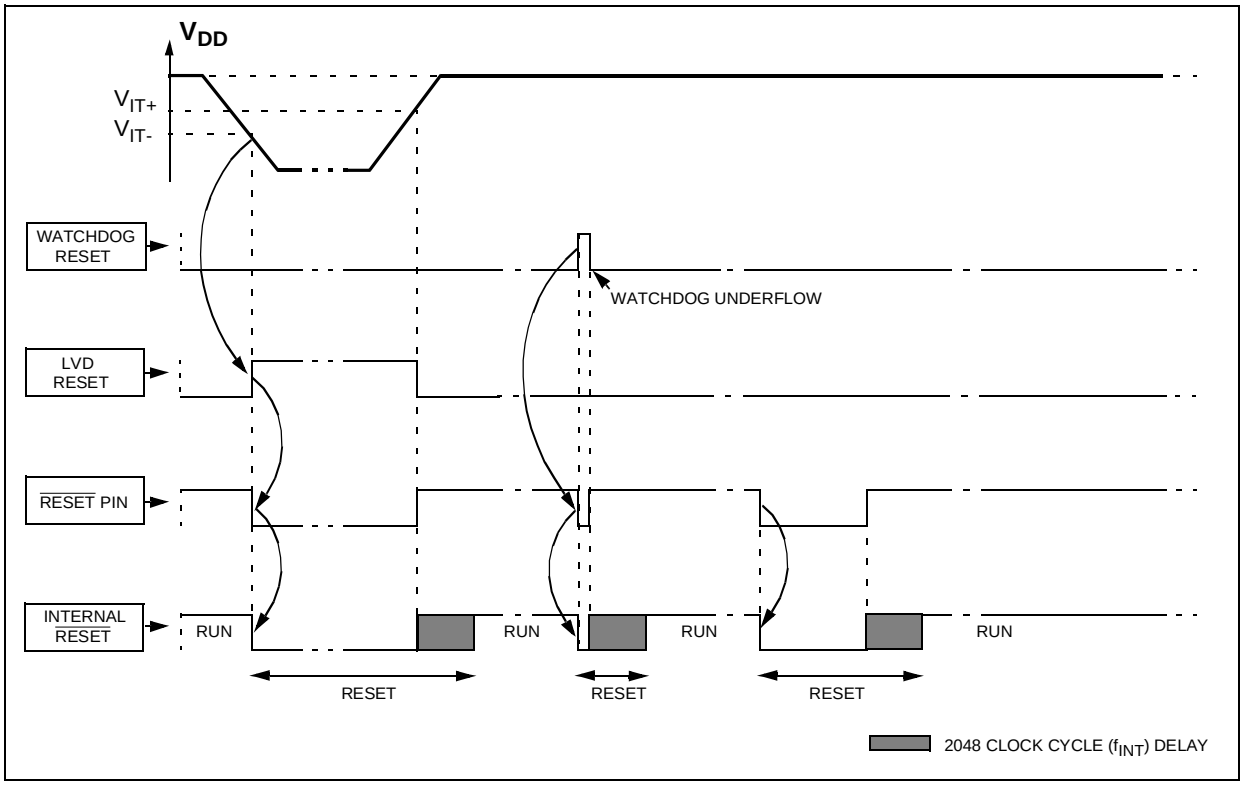
– The stack is cleared

– The PC is loaded with the address of the Reset vector. It is located in program ROM starting at address 0FFEh.

A jump to the beginning of the user program must be coded at this address.

– The interrupt flag is automatically set, so that the CPU is in Non Maskable Interrupt mode. This prevents the initialization routine from being interrupted. The initialization routine should therefore be terminated by a RETI instruction, in order to go back to normal mode.

Figure 13. RESET Sequence



RESET (Cont'd)

5.3.3 $\overline{\text{RESET}}$ Pin

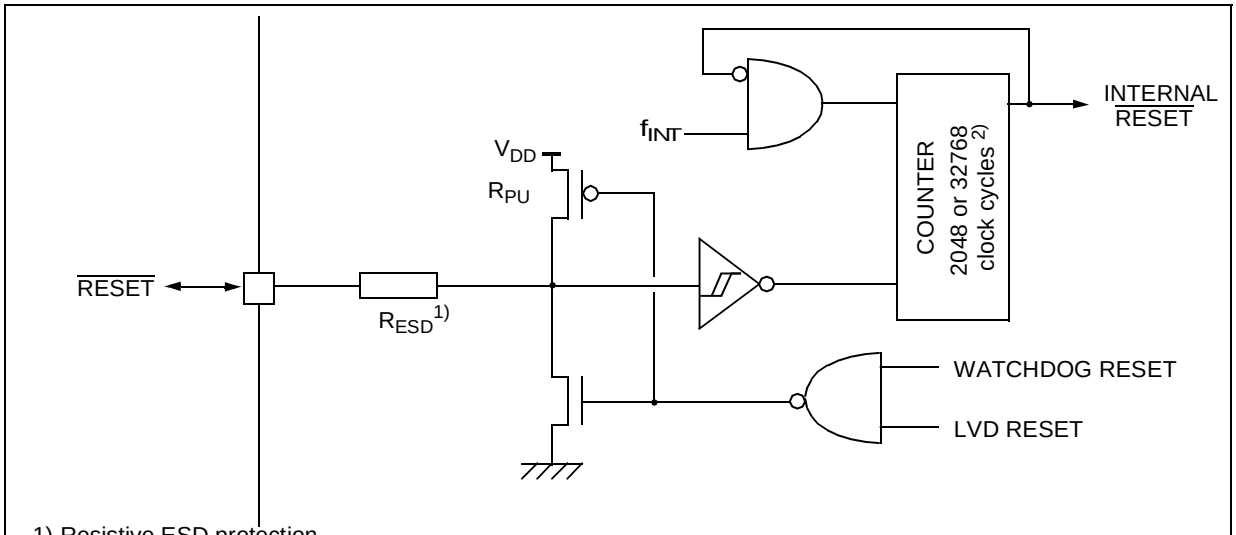
The $\overline{\text{RESET}}$ pin may be connected to a device on the application board in order to reset the MCU if required. The $\overline{\text{RESET}}$ pin may be pulled low in RUN, WAIT or STOP mode. This input can be used to reset the internal state of the MCU and ensure it starts-up correctly. The pin, which is connected to an internal pull-up, is active low and features a Schmitt trigger input. A delay (2048 clock cycles) added to the external signal ensures that even short pulses on the $\overline{\text{RESET}}$ pin are accepted as valid, provided V_{DD} has completed its rising phase and that the oscillator is running correctly (normal RUN or WAIT modes). The MCU is kept in the Reset state as long as the $\overline{\text{RESET}}$ pin is held low.

If the $\overline{\text{RESET}}$ pin is grounded while the MCU is in RUN or WAIT modes, processing of the user program is stopped (RUN mode only), the I/O ports are configured as inputs with pull-up resistors and the main oscillator is restarted. When the level on the $\overline{\text{RESET}}$ pin then goes high, the initialization sequence is executed at the end of the internal delay period.

If the $\overline{\text{RESET}}$ pin is grounded while the MCU is in STOP mode, the oscillator starts up and all the I/O ports are configured as inputs with pull-up resistors. When the $\overline{\text{RESET}}$ pin level then goes high, the initialization sequence is executed at the end of the internal delay period.

A simple external RESET circuitry is shown in Figure 15. For more details, please refer to the application note AN669.

Figure 14. Reset Block Diagram



1) Resistive ESD protection.
2) The reset delay value is selected through the option bytes.

RESET (Cont'd)

5.3.4 Watchdog Reset

The MCU provides a Watchdog timer function in order to be able to recover from software hang-ups. If the Watchdog register is not refreshed before an end-of-count condition is reached, a Watchdog reset is generated.

After a Watchdog reset, the MCU restarts in the same way as if a Reset was generated by the $\overline{\text{RESET}}$ pin.

Note: When a watchdog reset occurs, the $\overline{\text{RESET}}$ pin is tied low for very short time period, to flag the reset phase. This time is not long enough to reset external circuits.

For more details refer to the Watchdog Timer chapter.

5.3.5 LVD Reset

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

- Power-On RESET
- Voltage Drop RESET

During an LVD reset, the $\overline{\text{RESET}}$ pin is pulled low when $V_{DD}<V_{IT+}$ (rising edge) or $V_{DD}<V_{IT-}$ (falling edge).

For more details, refer to the LVD chapter.

Caution: Do not externally connect directly the $\overline{\text{RESET}}$ pin to V_{DD}, this may cause damage to the component in case of internal RESET (Watchdog or LVD).

Figure 15. Simple External Reset Circuitry

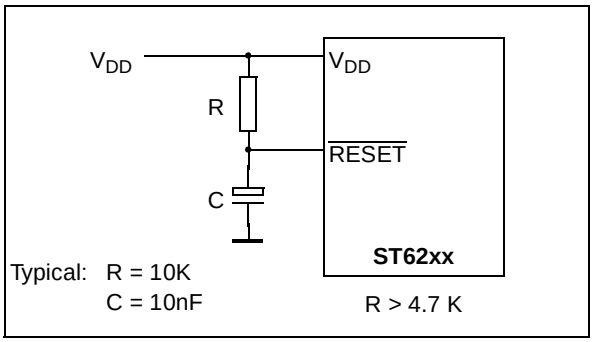
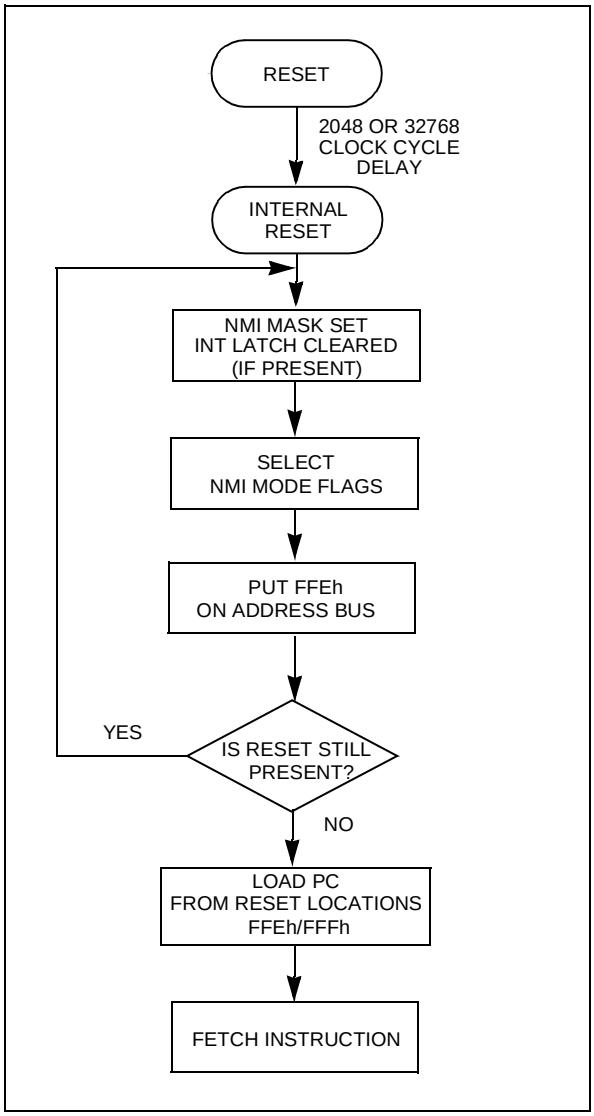


Figure 16. Reset Processing



5.4 INTERRUPTS

The ST6 core may be interrupted by four maskable interrupt sources, in addition to a Non Maskable Interrupt (NMI) source. The interrupt processing flowchart is shown in Figure 18.

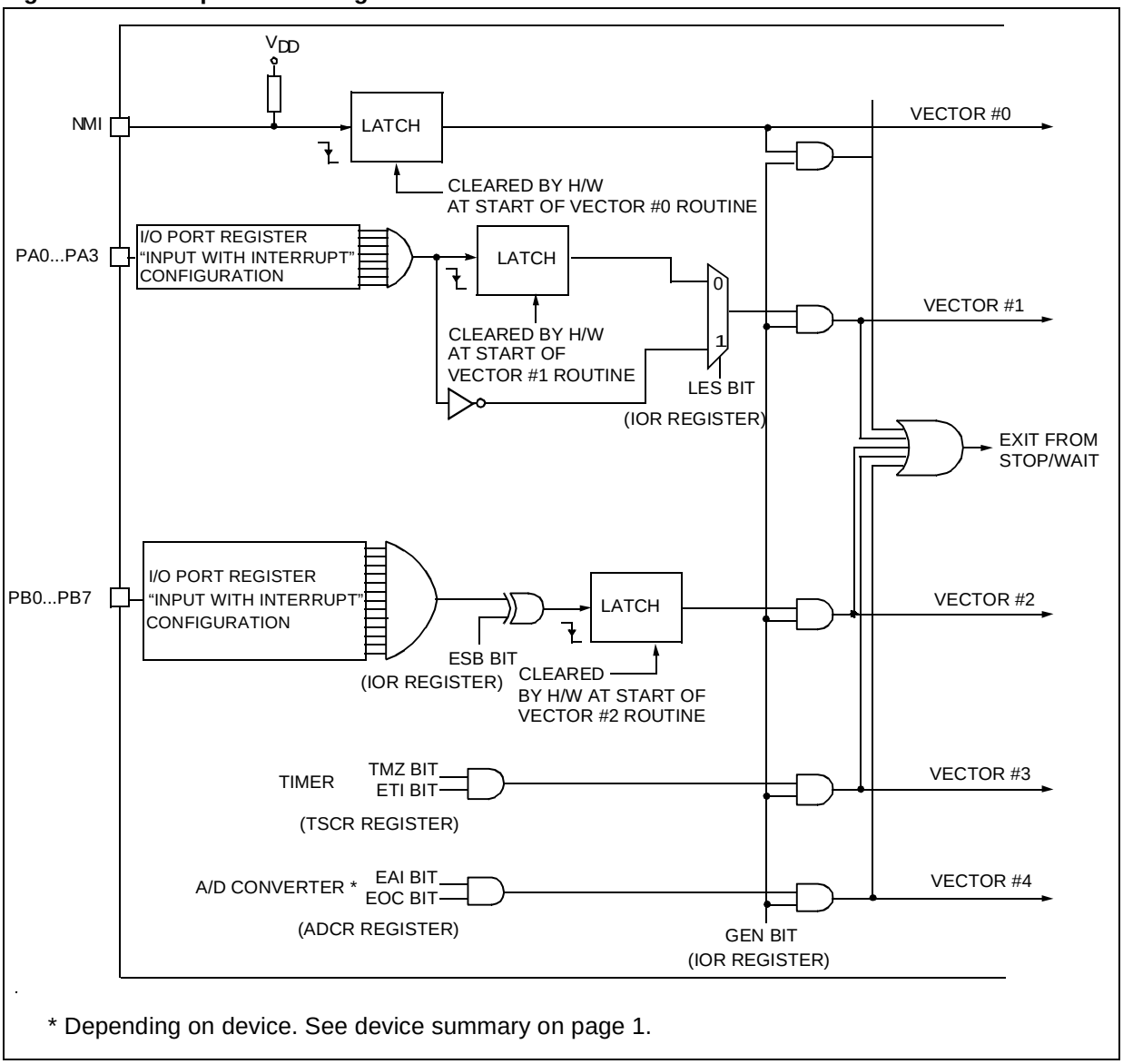
Maskable interrupts must be enabled by setting the GEN bit in the IOR register. However, even if they are disabled (GEN bit = 0), interrupt events are latched and may be processed as soon as the GEN bit is set.

Each source is associated with a specific Interrupt Vector, located in Program space (see Table 8). In the vector location, the user must write a Jump instruction to the associated interrupt service routine.

When an interrupt source generates an interrupt request, the PC register is loaded with the address of the interrupt vector, which then causes a Jump to the relevant interrupt service routine, thus servicing the interrupt.

Interrupt are triggered by events either on external pins, or from the on-chip peripherals. Several events can be ORed on the same interrupt vector. On-chip peripherals have flag registers to determine which event triggered the interrupt.

Figure 17. Interrupts Block Diagram



* Depending on device. See device summary on page 1.

5.5 INTERRUPT RULES AND PRIORITY MANAGEMENT

- A Reset can interrupt the NMI and peripheral interrupt routines
- The Non Maskable Interrupt request has the highest priority and can interrupt any peripheral interrupt routine at any time but cannot interrupt another NMI interrupt.
- No peripheral interrupt can interrupt another. If more than one interrupt request is pending, these are processed by the processor core according to their priority level: vector #1 has the highest priority while vector #4 the lowest. The priority of each interrupt source is fixed by hardware (see Interrupt Mapping table).

5.6 INTERRUPTS AND LOW POWER MODES

All interrupts cause the processor to exit from WAIT mode. Only the external and some specific interrupts from the on-chip peripherals cause the processor to exit from STOP mode (refer to the “Exit from STOP“ column in the Interrupt Mapping Table).

5.7 NON MASKABLE INTERRUPT

This interrupt is triggered when a falling edge occurs on the NMI pin regardless of the state of the GEN bit in the IOR register. An interrupt request on NMI vector #0 is latched by a flip flop which is automatically reset by the core at the beginning of the NMI service routine.

5.8 PERIPHERAL INTERRUPTS

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

– The GEN bit of the IOR register is set

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

Peripheral interrupts are linked to vectors #3 and #4. Interrupt requests are flagged by a bit in their corresponding control register. This means that a request cannot be lost, because the flag bit must be cleared by user software.

5.9 EXTERNAL INTERRUPTS (I/O Ports)

External interrupt vectors can be loaded into the PC register if the corresponding external interrupt occurred and if the GEN bit is set. These interrupts allow the processor to exit from STOP mode.

The external interrupt polarity is selected through the IOR register.

External interrupts are linked to vectors #1 and # 2.

Interrupt requests on vector #1 can be configured either as edge or level-sensitive using the LES bit in the IOR Register.

Interrupt requests from vector #2 are always edge sensitive. The edge polarity can be configured using the ESB bit in the IOR Register.

In edge-sensitive mode, a latch is set when a edge occurs on the interrupt source line and is cleared when the associated interrupt routine is started. So, an interrupt request can be stored until completion of the currently executing interrupt routine, before being processed. If several interrupt requests occurs before completion of the current interrupt routine, only the first request is stored.

Storing of interrupt requests is not possible in level sensitive mode. To be taken into account, the low level must be present on the interrupt pin when the MCU samples the line after instruction execution.

5.9.1 Notes on using External Interrupts

ESB bit Spurious Interrupt on Vector #2

If a pin associated with interrupt vector #2 is configured as interrupt with pull-up, whenever vector #2 is configured to be rising edge sensitive (by setting the ESB bit in the IOR register), an interrupt is latched although a rising edge may not have occured on the associated pin.

This is due to the vector #2 circuitry.The workaround is to discard this first interrupt request in the routine (using a flag for example).

Masking of One Interrupt by Another on Vector #2.

When two or more port pins (associated with interrupt vector #2) are configured together as input with interrupt (falling edge sensitive), as long as one pin is stuck at ’0’, the other pin can never generate an interrupt even if an active edge occurs at this pin. The same thing occurs when one pin is stuck at ’1’ and interrupt vector #2 is configured as rising edge sensitive.

To avoid this the first pin must input a signal that goes back up to ’1’ right after the falling edge. Otherwise, in the interrupt routine for the first pin, deactivate the “input with interrupt” mode using the port control registers (DDR, OR, DR). An active edge on another pin can then be latched.

I/O port Configuration Spurious Interrupt on Vector #2

If a pin associated with interrupt vector #2 is in ‘input with pull-up’ state, a ‘0’ level is present on the pin and the ESB bit = 0, when the I/O pin is configured as interrupt with pull-up by writing to the DDRx, ORx and DRx register bits, an interrupt is latched although a falling edge may not have occurred on the associated pin.

In the opposite case, if the pin is in interrupt with pull-up state , a 0 level is present on the pin and the ESB bit =1, when the I/O port is configured as input with pull-up by writing to the DDRx, ORx and DRx bits, an interrupt is latched although a rising edge may not have occurred on the associated pin.

5.10 INTERRUPT HANDLING PROCEDURE

The interrupt procedure is very similar to a call procedure, in fact the user can consider the interrupt as an asynchronous call procedure. As this is an asynchronous event, the user cannot know the context and the time at which it occurred. As a result, the user should save all Data space registers which may be used within the interrupt routines. The following list summarizes the interrupt procedure:

When an interrupt request occurs, the following actions are performed by the MCU automatically:

– The core switches from the normal flags to the interrupt flags (or the NMI flags).

– The PC contents are stored in the top level of the stack.

– The normal interrupt lines are inhibited (NMI still active).

– The internal latch (if any) is cleared.

– The associated interrupt vector is loaded in the PC.

When an interrupt request occurs, the following actions must be performed by the user software:

– User selected registers have to be saved within the interrupt service routine (normally on a software stack).

– The source of the interrupt must be determined by polling the interrupt flags (if more than one source is associated with the same vector).

– The RETI (RETurn from Interrupt) instruction must end the interrupt service routine.

After the RETI instruction is executed, the MCU returns to the main routine.

Caution: When a maskable interrupt occurs while the ST6 core is in NORMAL mode and during the execution of an “ldi IOR, 00h” instruction (disabling all maskable interrupts): if the interrupt request occurs during the first 3 cycles of the “ldi” instruction (which is a 4-cycle instruction) the core will switch to interrupt mode BUT the flags CN and ZN will NOT switch to the interrupt pair CI and ZI.

5.10.1 Interrupt Response Time

This is defined as the time between the moment when the Program Counter is loaded with the interrupt vector and when the program has jump to the interrupt subroutine and is ready to execute the code. It depends on when the interrupt occurs while the core is processing an instruction.

Figure 18. Interrupt Processing Flow Chart

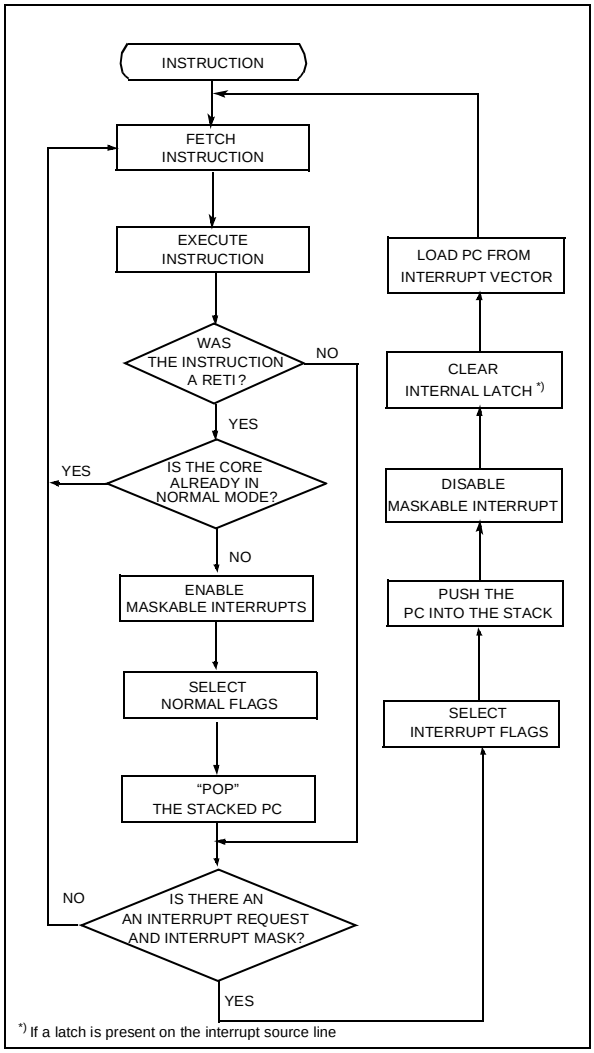


Table 7. Interrupt Response Time

Minimum	6 CPU cycles
Maximum	11 CPU cycles

One CPU cycle is 13 external clock cycles thus 11 CPU cycles = 11 x (13 /8M) = 17.875 µs with an 8 MHz external quartz.

5.11 REGISTER DESCRIPTION

INTERRUPT OPTION REGISTER (IOR)

Address: 0C8h — Write Only

Reset status: 00h

7							0
-	LES	ESB	GEN	-	-	-	-

Caution: This register is write-only and cannot be accessed by single-bit operations (SET, RES, DEC,...).

Bit 7 =Reserved, must be cleared.

Bit 6 = **LES** *Level/Edge Selection bit.*
0: Falling edge sensitive mode is selected for interrupt vector #1
1: Low level sensitive mode is selected for interrupt vector #1

Bit 5 = **ESB** *Edge Selection bit.*
0: Falling edge mode on interrupt vector #2
1: Rising edge mode on interrupt vector #2

Bit 4 = **GEN** *Global Enable Interrupt.*
0: Disable all maskable interrupts
1: Enable all maskable interrupts

Note: When the GEN bit is cleared, the NMI interrupt is active but cannot be used to exit from STOP or WAIT modes.

Bits 3:0 = Reserved, must be cleared.

Table 8. Interrupt Mapping

Vector number	Source Block	Description	Register Label	Flag	Exit from STOP	Vector Address	Priority Order
	RESET	Reset	N/A	N/A	yes	FFEh-FFFh	Highest Priority
Vector #0	NMI	Non Maskable Interrupt	N/A	N/A	yes	FFCh-FFDh	
NOT USED						FFAh-FFBh	
						FF8h-FF9h	
Vector #1	Port A	Ext. Interrupt Port A	N/A	N/A	yes	FF6h-FF7h	
Vector #2	Port B	Ext. Interrupt Port B	N/A	N/A	yes	FF4h-FF5h	
Vector #3	TIMER	Timer underflow	TSCR	TMZ	yes	FF2h-FF3h	Lowest Priority
Vector #4	ADC*	End Of Conversion	ADCR	EOC	no	FF0h-FF1h	

* Depending on device. See device summary on page 1.

6 POWER SAVING MODES

6.1 INTRODUCTION

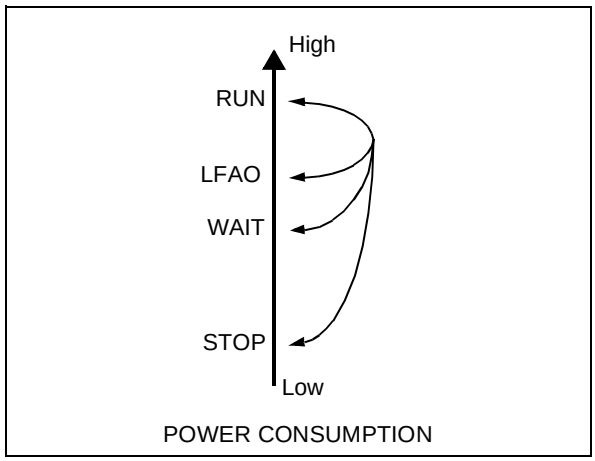
To give a large measure of flexibility to the application in terms of power consumption, two main power saving modes are implemented in the ST6 (see Figure 19).

In addition, the Low Frequency Auxiliary Oscillator (LFAO) can be used instead of the main oscillator to reduce power consumption in RUN and WAIT modes.

After a RESET the normal operating mode is selected by default (RUN mode). This mode drives the device (CPU and embedded peripherals) by means of a master clock which is based on the main oscillator frequency.

From Run mode, the different power saving modes may be selected by calling the specific ST6 software instruction or for the LFAO by setting the relevant register bit. For more information on the LFAO, please refer to the Clock chapter.

Figure 19. Power Saving Mode Transitions



6.2 WAIT MODE

The MCU goes into WAIT mode as soon as the WAIT instruction is executed. This has the following effects:

– Program execution is stopped, the microcontroller software can be considered as being in a “frozen” state.

– RAM contents and peripheral registers are preserved as long as the power supply voltage is higher than the RAM retention voltage.

– The oscillator is kept running to provide a clock to the peripherals; they are still active.

WAIT mode can be used when the user wants to reduce the MCU power consumption during idle periods, while not losing track of time or the ability to monitor external events. WAIT mode places the MCU in a low power consumption mode by stopping the CPU. The active oscillator (main oscillator or LFAO) is kept running in order to provide a clock signal to the peripherals.

If the power consumption has to be further reduced, the Low Frequency Auxiliary Oscillator (LFAO) can be used in place of the main oscillator, if its operating frequency is lower. If required, the LFAO must be switched on before entering WAIT mode.

Exit from Wait mode

The MCU remains in WAIT mode until one of the following events occurs:

– RESET (Watchdog, LVD or $\overline{\text{RESET}}$ pin)

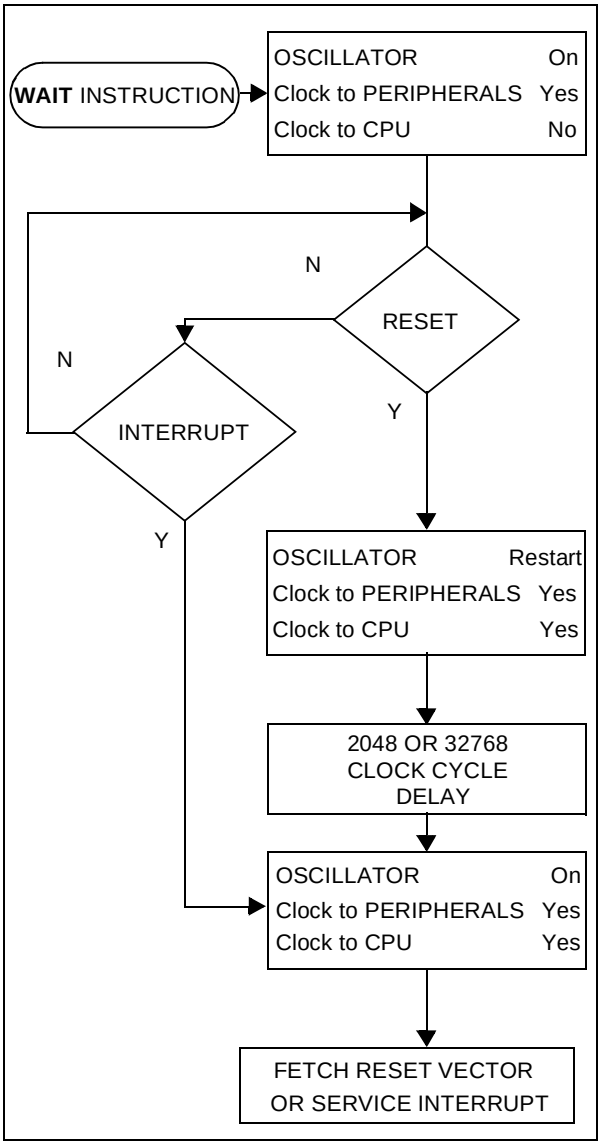
– A peripheral interrupt (timer, ADC,...),

– An external interrupt (I/O port, NMI)

The Program Counter then branches to the starting address of the interrupt or RESET service routine. Refer to Figure 20.

See also Section 6.4.1.

Figure 20. WAIT Mode Flowchart



6.3 STOP MODE

STOP mode is the lowest power consumption mode of the MCU (see Figure 22).

The MCU goes into STOP mode as soon as the STOP instruction is executed. This has the following effects:

– Program execution is stopped, the microcontroller can be considered as being “frozen”.

– The contents of RAM and the peripheral registers are kept safely as long as the power supply voltage is higher than the RAM retention voltage.

– The oscillator is stopped, so peripherals cannot work except the those that can be driven by an external clock.

Exit from STOP Mode

The MCU remains in STOP mode until one of the following events occurs:

– RESET (Watchdog, LVD or $\overline{\text{RESET}}$ pin)

– A peripheral interrupt (assuming this peripheral can be driven by an external clock)

– An external interrupt (I/O port, NMI)

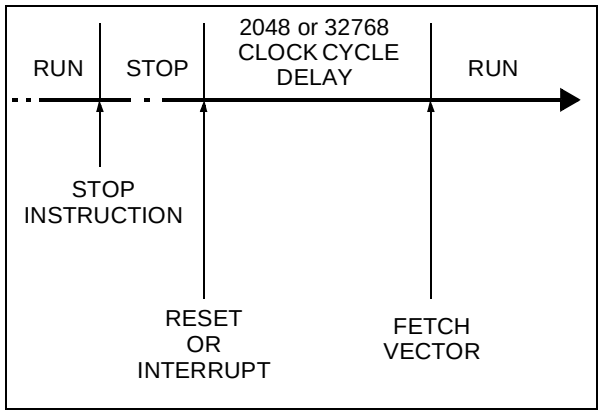
In all cases a delay of 2048 or 32768 clock cycles (f_{INT}) is generated to make sure the oscillator has started properly.

The Program Counter then points to the starting address of the interrupt or RESET service routine (see Figure 21).

STOP Mode and Watchdog

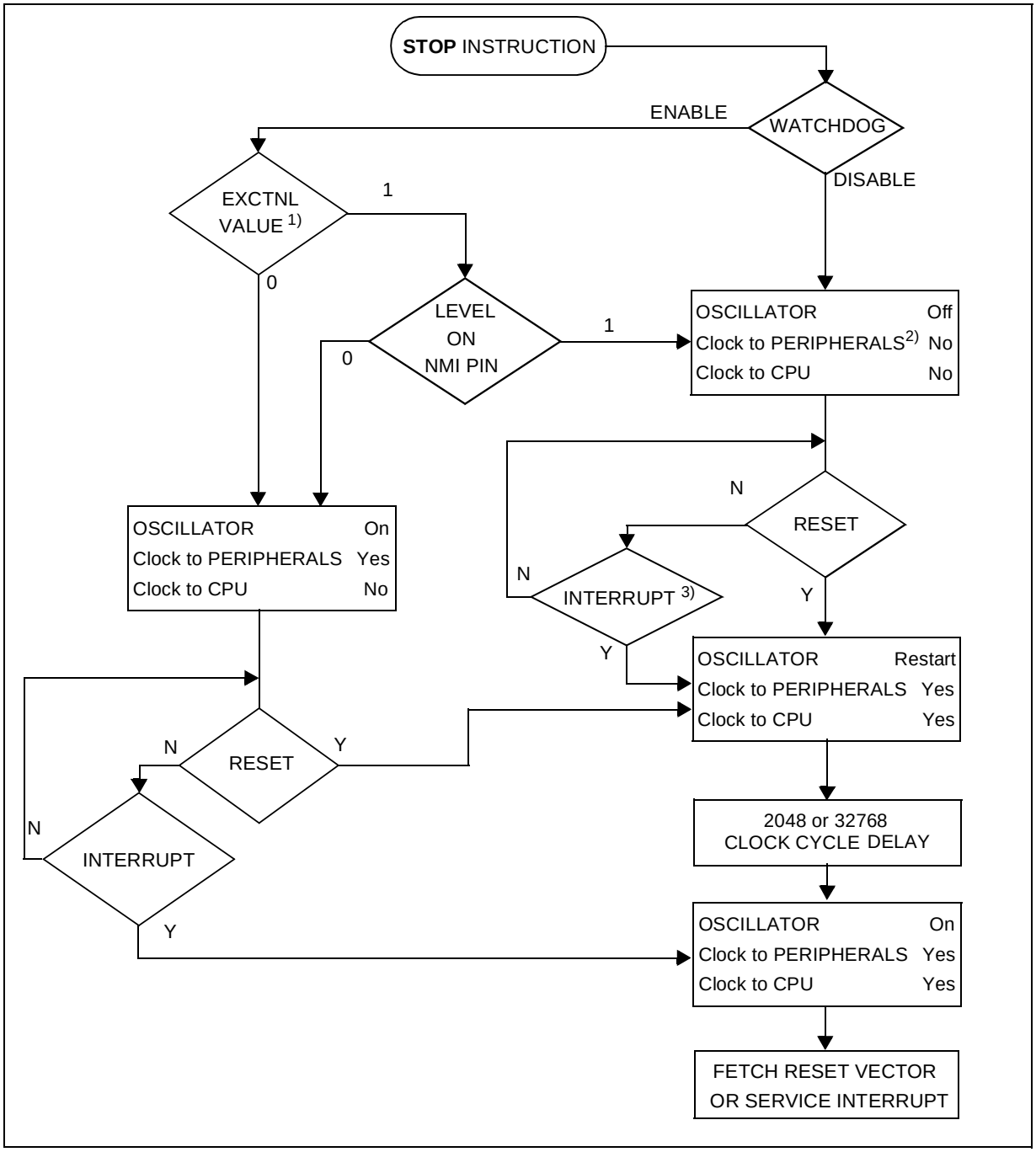
When the Watchdog is active (hardware or software activation), the STOP instruction is disabled and a WAIT instruction will be executed in its place unless the EXCTNL option bit is set to 1 in the option bytes and a a high level is present on the NMI pin. In this case, the STOP instruction will be executed and the Watchdog will be frozen.

Figure 21. STOP Mode Timing Overview



STOP MODE (Cont'd)

Figure 22. STOP Mode Flowchart



Notes:

1. EXCTNL 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 STOP mode (such as external interrupt). Refer to the Interrupt Mapping table for more details.

6.4 NOTES RELATED TO WAIT AND STOP MODES

6.4.1 Exit from Wait and Stop Modes

6.4.1.1 NMI Interrupt

It should be noted that when the GEN bit in the IOR register is low (interrupts disabled), the NMI interrupt is active but cannot cause a wake up from STOP/WAIT modes.

6.4.1.2 Restart Sequence

When the MCU exits from WAIT or STOP mode, it should be noted that the restart sequence depends on the original state of the MCU (normal, interrupt or non-maskable interrupt mode) prior to entering WAIT or STOP mode, as well as on the interrupt type.

Normal Mode. If the MCU was in the main routine when the WAIT or STOP instruction was executed, exit from Stop or Wait mode will occur as soon as an interrupt occurs; the related interrupt routine is executed and, on completion, the instruction which follows the STOP or WAIT instruction is then executed, providing no other interrupts are pending.

Non Maskable Interrupt Mode. If the STOP or WAIT instruction has been executed during execution of the non-maskable interrupt routine, the MCU exits from Stop or Wait mode as soon as an interrupt occurs: the instruction which follows the STOP or WAIT instruction is executed, and the MCU remains in non-maskable interrupt mode, even if another interrupt has been generated.

Normal Interrupt Mode. If the MCU was in interrupt mode before the STOP or WAIT instruction was executed, it exits from STOP or WAIT mode as soon as an interrupt occurs. Nevertheless, two cases must be considered:

– If the interrupt is a normal one, the interrupt routine in which the WAIT or STOP mode was entered will be completed, starting with the execution of the instruction which follows the STOP or the WAIT instruction, and the MCU is still in interrupt mode. At the end of this routine pending interrupts will be serviced according to their priority.

– In the event of a non-maskable interrupt, the non-maskable interrupt service routine is processed first, then the routine in which the WAIT or STOP mode was entered will be completed by executing the instruction following the STOP or WAIT instruction. The MCU remains in normal interrupt mode.

6.4.2 Recommended MCU Configuration

For lowest power consumption during RUN or WAIT modes, the user software must configure the MCU as follows:

– Configure unused I/Os as output push-pull low mode

– Place all peripherals in their power down modes before entering STOP mode

– Select the Low Frequency Auxiliary Oscillator (provided this runs at a lower frequency than the main oscillator).

The WAIT and STOP instructions are not executed if an enabled interrupt request is pending.

7 I/O PORTS

7.1 INTRODUCTION

Each I/O port contains up to 8 pins. Each pin can be programmed independently as digital input (with or without pull-up and interrupt generation), digital output (open drain, push-pull) or analog input (when available).

The I/O pins can be used in either standard or alternate function mode.

Standard I/O mode is used for:

– Transfer of data through digital inputs and outputs (on specific pins):

– External interrupt generation

Alternate function mode is used for:

– Alternate signal input/output for the on-chip peripherals

The generic I/O block diagram is shown in Figure 23.

7.2 FUNCTIONAL DESCRIPTION

Each port is associated with 3 registers located in Data space:

– Data Register (DR)

– Data Direction Register (DDR)

– Option Register (OR)

Each I/O pin may be programmed using the corresponding register bits in the DDR, DR and OR registers: bit x corresponding to pin x of the port. Table 9 illustrates the various port configurations which can be selected by user software.

During MCU initialization, all I/O registers are cleared and the input mode with pull-up and no interrupt generation is selected for all the pins, thus avoiding pin conflicts.

7.2.1 Digital Input Modes

The input configuration is selected by clearing the corresponding DDR register bit.

In this case, reading the DR register returns the digital value applied to the external I/O pin.

Different input modes can be selected by software through the DR and OR registers, see Table 9.

External Interrupt Function

All input lines can be individually connected by software to the interrupt system by programming the OR and DR registers accordingly. The interrupt trigger modes (falling edge, rising edge and low level) can be configured by software for each port as described in the Interrupt section.

7.2.2 Analog Inputs

Some pins can be configured as analog inputs by programming the OR and DR registers accordingly, see Table 9. These analog inputs are connected to the on-chip 8-bit Analog to Digital Converter.

Caution: ONLY ONE pin should be programmed as an analog input at any time, since by selecting more than one input simultaneously their pins will be effectively shorted.

7.2.3 Output Modes

The output configuration is selected by setting the corresponding DDR register bit. In this case, writing to the DR register applies this digital value to the I/O pin through the latch. Then, reading the DR register returns the previously stored value.

Two different output modes can be selected by software through the OR register: push-pull and open-drain.

DR register value and output pin status:

DR	Push-pull	Open-drain
0	V_{SS}	V_{SS}
1	V_{DD}	Floating

Note: The open drain setting is not a true open drain. This means it has the same structure as the push-pull setting but the P-buffer is deactivated. To avoid damaging the device, please respect the V_{OUT} absolute maximum rating described in the Electrical Characteristics section.

7.2.4 Alternate Functions

When an on-chip peripheral is configured to use a pin, the alternate function (timer input/output...) is not systematically selected but has to be configured through the DDR, OR and DR registers. Refer to the chapter describing the peripheral for more details.

I/O PORTS (Cont'd)

Figure 23. I/O Port Block Diagram

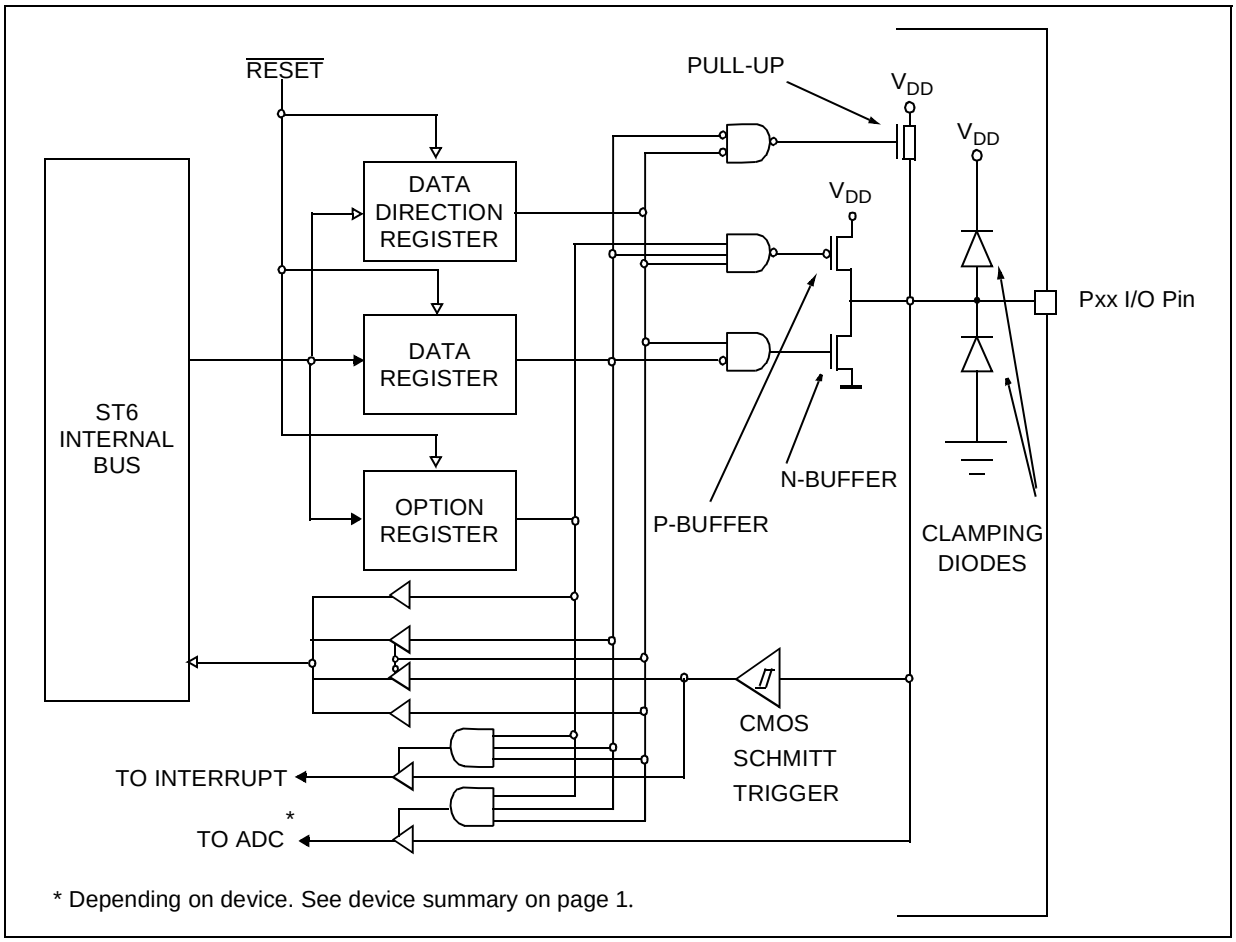


Table 9. I/O Port Configurations

DDR	OR	DR	Mode	Option
0	0	0	Input	With pull-up, no interrupt
0	0	1	Input	No pull-up, no interrupt
0	1	0	Input	With pull-up and with interrupt
0	1	1	Input	Analog input (when available)
1	0	x	Output	Open-drain output (20mA sink when available)
1	1	x	Output	Push-pull output (20mA sink when available)

Note: x = Don't care

I/O PORTS (Cont'd)

7.2.5 Instructions NOT to be used to access Port Data registers (SET, RES, INC and DEC)

DO NOT USE READ-MODIFY-WRITE INSTRUCTIONS (SET, RES, INC and DEC) ON PORT DATA REGISTERS IF ANY PIN OF THE PORT IS CONFIGURED IN INPUT MODE.

These instructions make an implicit read and write back of the entire register. In port input mode, however, the data register reads from the input pins directly, and not from the data register latches. Since data register information in input mode is used to set the characteristics of the input pin (interrupt, pull-up, analog input), these may be unintentionally reprogrammed depending on the state of the input pins.

As a general rule, it is better to only use single bit instructions on data registers when the whole (8-bit) port is in output mode. In the case of inputs or of mixed inputs and outputs, it is advisable to keep a copy of the data register in RAM. Single bit instructions may then be used on the RAM copy, after which the whole copy register can be written to the port data register:

```
SET bit, datacopy
LD a, datacopy
LD DRA, a
```

7.2.6 Recommendations

1. Safe I/O State Switching Sequence

Switching the I/O ports from one state to another should be done in a sequence which ensures that no unwanted side effects can occur. The recommended safe transitions are illustrated in Figure 24 The Interrupt Pull-up to Input Analog transition (and vice-vesra) is potentially risky and should be avoided when changing the I/O operating mode.

2. Handling Unused Port Bits

On ports that have less than 8 external pins connected:

– Leave the unbonded pins in reset state and do not change their configuration.

– Do not use instructions that act on a whole port register (INC, DEC, or read operations). Unavailable bits must be masked by software (AND instruction). Thus, when a read operation performed on an incomplete port is followed by a comparison, use a mask.

3. High Impedance Input

On any CMOS device, it is not recommended to connect high impedance on input pins. The choice of these impedance has to be done with respect to the maximum leakage current defined in the datasheet. The risk is to be close or out of specification on the input levels applied to the device.

7.3 LOW POWER MODES

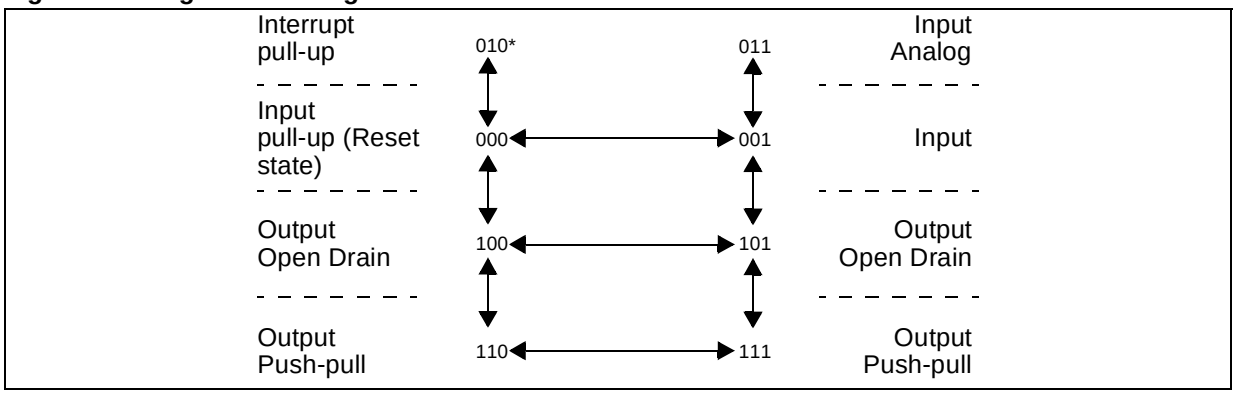
The WAIT and STOP instructions allow the ST62xx to be used in situations where low power consumption is needed. The lowest power consumption is achieved by configuring I/Os in output push-pull low mode.

Mode	Description
WAIT	No effect on I/O ports. External interrupts cause the device to exit from WAIT mode.
STOP	No effect on I/O ports. External interrupts cause the device to exit from STOP mode.

7.4 INTERRUPTS

The external interrupt event generates an interrupt if the corresponding configuration is selected with DDR, DR and OR registers (see Table 9) and the GEN-bit in the IOR register is set.

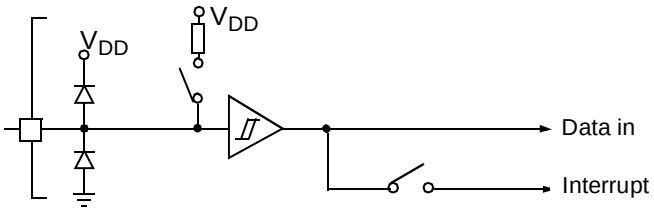
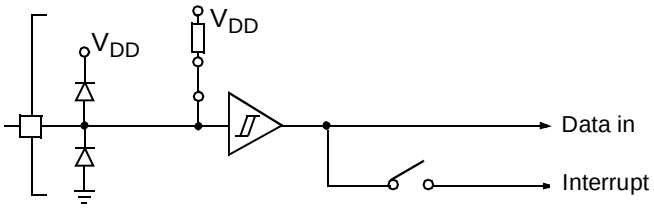
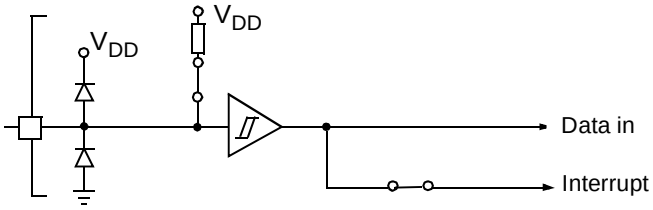
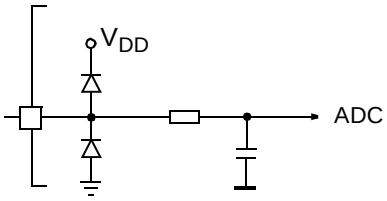
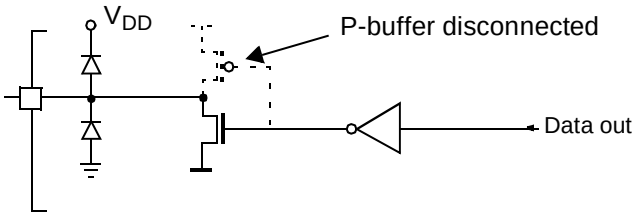
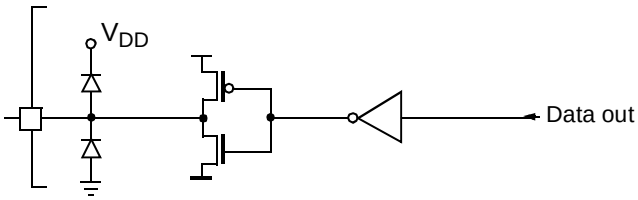
Figure 24. Diagram showing Safe I/O State Transitions



Note *. xxx = DDR, OR, DR Bits respectively

I/O PORTS (Cont'd)

Table 10. I/O Port Option Selections

	MODE			AVAILABLE ON[1]	SCHEMATIC
Digital Input	Input			PA0-PA3 PB0-PB7	
	DDRx 0	ORx 0	DRx 1		
	Reset state Input with pull up			PA0-PA3 PB0-PB7	
	DDRx 0	ORx 0	DRx 0		
	Input with pull up with interrupt			PA0-PA3 PB0-PB7	
	DDRx 0	ORx 1	DRx 0		
Analog Input	Analog Input			PB0-PB3 (ST6210C/20C only) PB4-PB7 (All devices, except ST6208C)	
	DDRx 0	ORx 1	DRx 1		
Digital output	Open drain output (5mA)			PB0-PB7	
	Open drain output (20 mA)			PA0-PA3	
	DDRx 1	ORx 0	DRx 0/1		
	Push-pull output (5mA)			PB0-PB7	
	Push-pull output (20 mA)			PA0-PA3	
	DDRx 1	ORx 1	DRx 0/1		

Note 1. Provided the correct configuration has been selected (see Table 9).

I/O PORTS (Cont'd)

7.5 REGISTER DESCRIPTION

DATA REGISTER (DR)

Port x Data Register
DRx with x = A or B.

Address DRA: 0C0h - Read/Write
Address DRB: 0C1h - Read/Write

Reset Value: 0000 0000 (00h)

7							0
D7	D6	D5	D4	D3	D2	D1	D0

Bits 7:0 = **D[7:0]** *Data register bits.*

Reading the DR register returns either the DR register latch content (pin configured as output) or the digital value applied to the I/O pin (pin configured as input).

Caution: In input mode, modifying this register will modify the I/O port configuration (see Table 9).

Do not use the Single bit instructions on I/O port data registers. See (Section 7.2.5).

DATA DIRECTION REGISTER (DDR)

Port x Data Direction Register
DDRx with x = A or B.

Address DDRA: 0C4h - Read/Write
Address DDRB: 0C5h - Read/Write

Reset Value: 0000 0000 (00h)

7							0
DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0

Bits 7:0 = **DD[7:0]** *Data direction register bits.*

The DDR register gives the input/output direction configuration of the pins. Each bit is set and cleared by software.
0: Input mode
1: Output mode

OPTION REGISTER (OR)

Port x Option Register
ORx with x = A or B.

Address ORA: 0CCh - Read/Write
Address ORB: 0CDh - Read/Write

Reset Value: 0000 0000 (00h)

7							0
O7	O6	O5	O4	O3	O2	O1	O0

Bits 7:0 = **O[7:0]** *Option register bits.*

The OR register allows to distinguish in output mode if the push-pull or open drain configuration is selected.

Output mode:
0: Open drain output(with P-Buffer deactivated)
1: Push-pull Output

Input mode: See Table 9.

Each bit is set and cleared by software.

Caution: Modifying this register, will also modify the I/O port configuration in input mode. (see Table 9).

Table 11. I/O Port Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
Reset Value of all I/O port registers		0	0	0	0	0	0	0	0
0C0h	**DRA**	MSB							LSB
0C1h	**DRB**								
0C4h	**DDRA**	MSB							LSB
0C5h	**DDRB**								
0CCh	**ORA**	MSB							LSB
0CDh	**ORB**								

8 ON-CHIP PERIPHERALS

8.1 WATCHDOG TIMER (WDG)

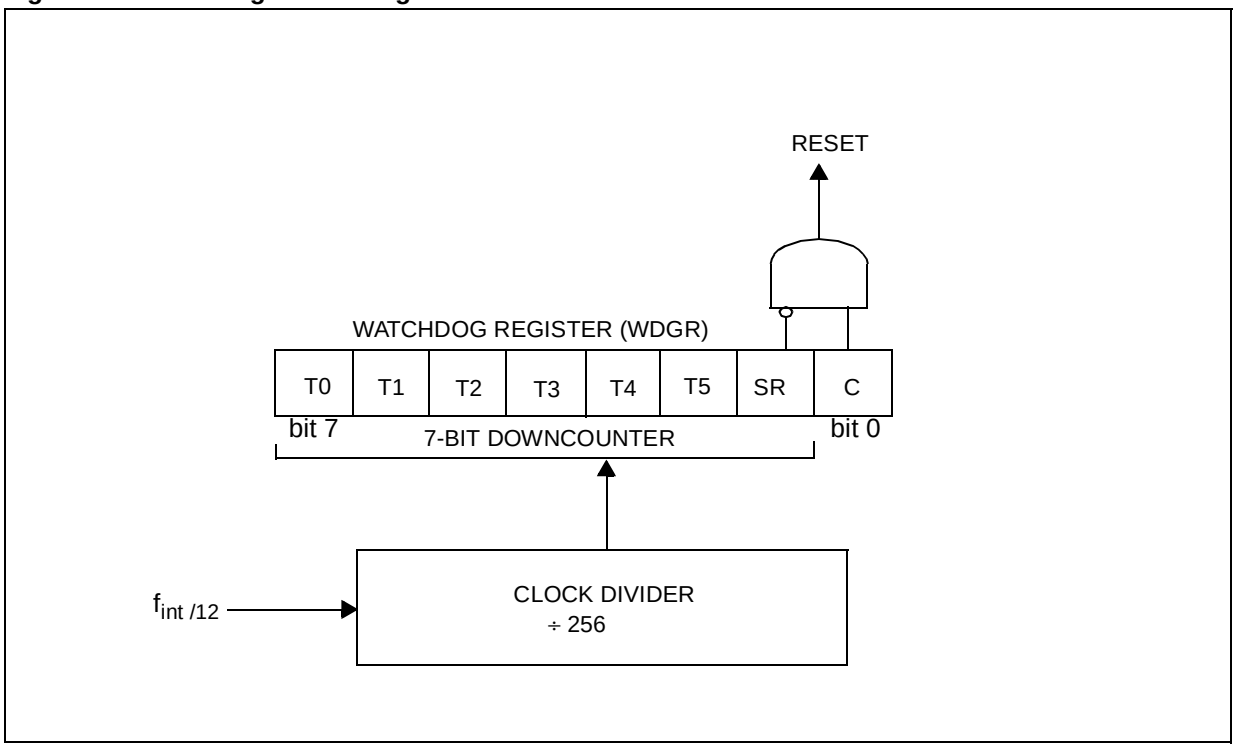
8.1.1 Introduction

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

8.1.2 Main Features

- Programmable timer (64 steps of 3072 clock cycles)
- Software reset
- Reset (if watchdog activated) when the SR bit reaches zero
- Hardware or software watchdog activation selectable by option bit (Refer to the option bytes section)

Figure 25. Watchdog Block Diagram



WATCHDOG TIMER (Cont'd)

8.1.3 Functional Description

The watchdog activation is selected through an option in the option bytes:

– **HARDWARE** Watchdog option

After reset, the watchdog is permanently active, the C bit in the WDGR is forced high and the user can not change it. However, this bit can be read equally as 0 or 1.

– **SOFTWARE** Watchdog option

After reset, the watchdog is deactivated. The function is activated by setting C bit in the WDGR register. Once activated, it cannot be deactivated. The counter value stored in the WDGR register (bits SR:T0), is decremented every 3072 clock cycles. The length of the timeout period can be programmed by the user in 64 steps of 3072 clock cycles.

If the watchdog is activated (by setting the C bit) and when the SR bit is cleared, the watchdog initiates a reset cycle pulling the reset pin low for typically 500ns.

The application program must write in the WDGR register at regular intervals during normal operation to prevent an MCU reset. The value to be stored in the WDGR register must be between FEh and 02h (see Table 12). To run the watchdog function the following conditions must be true:

– The C bit is set (watchdog activated)

– The SR bit is set to prevent generating an immediate reset

– The T[5:0] bits contain the number of decrements which represent the time delay before the watchdog produces a reset.

Table 12. Watchdog Timing (f_{OSC} = 8 MHz)

	WDGR Register initial value	WDG timeout period (ms)
Max.	FEh	24.576
Min.	02h	0.384

8.1.3.1 Software Reset

The SR bit can be used to generate a software reset by clearing the SR bit while the C bit is set.

8.1.4 Recommendations

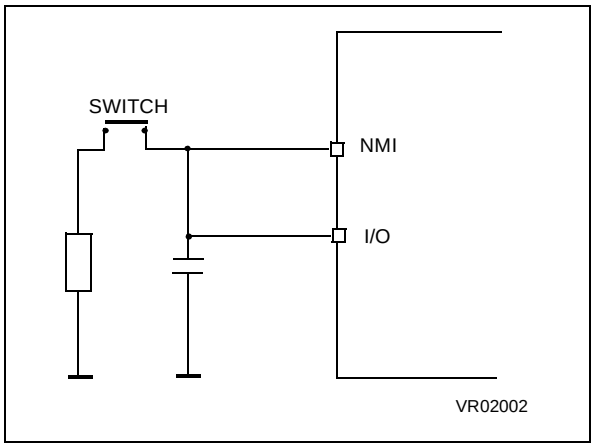
1. The Watchdog plays an important supporting role in the high noise immunity of ST62xx devices, and should be used wherever possible. Watchdog related options should be selected on the basis of a trade-off between application security and STOP mode availability (refer to the description of the WDACT and EXTCNTL bits on the Option Bytes).

When STOP mode is not required, hardware activation without EXTERNAL STOP MODE CONTROL should be preferred, as it provides maximum security, especially during power-on.

When STOP mode is required, hardware activation and EXTERNAL STOP MODE CONTROL should be chosen. NMI should be high by default, to allow STOP mode to be entered when the MCU is idle.

The NMI pin can be connected to an I/O line (see Figure 26) to allow its state to be controlled by software. The I/O line can then be used to keep NMI low while Watchdog protection is required, or to avoid noise or key bounce. When no more processing is required, the I/O line is released and the device placed in STOP mode for lowest power consumption.

Figure 26. A typical circuit making use of the EXERNAL STOP MODE CONTROL feature



2. When software activation is selected (WDACT bit in Option byte) and the Watchdog is not activated, the downcounter may be used as a simple 7-bit timer (remember that the bits are in reverse order).

The software activation option should be chosen only when the Watchdog counter is to be used as a timer. To ensure the Watchdog has not been unexpectedly activated, the following instructions should be executed:

```
jrr 0, WDGR, #+3 ; If C=0,jump to next
ldi WDGR, 0FDH   ; SR=0 -> reset

next :
```

WATCHDOG TIMER (Cont'd)

These instructions test the C bit and reset the MCU (i.e. disable the Watchdog) if the bit is set (i.e. if the Watchdog is active), thus disabling the Watchdog.

For more information on the use of the watchdog, please read application note AN1015.

Note: This note applies only when the watchdog is used as a standard timer. It is recommended to read the counter twice, as it may sometimes return an invalid value if the read is performed while the counter is decremented (counter bits in transient state). To validate the return value, both values read must be equal. The counter decrements every 384 µs at 8 MHz f_{OSC}.

8.1.5 Low Power Modes

Mode	Description
WAIT	No effect on Watchdog.
STOP	Behaviour depends on the EXTCNTL option in the Option bytes: **1. Watchdog disabled:** The MCU will enter Stop mode if a STOP instruction is executed. **2. Watchdog enabled and EXTCNTL option disabled:** If a STOP instruction is encountered, it is interpreted as a WAIT. **3. Watchdog and EXTCNTL option enabled:** If a STOP instruction is encountered when the NMI pin is low, it is interpreted as a WAIT. If, however, the STOP instruction is encountered when the NMI pin is high, the Watchdog counter is frozen and the CPU enters STOP mode. When the MCU exits STOP mode (i.e. when an interrupt is generated), the Watchdog resumes its activity.

8.1.6 Interrupts

None.

WATCHDOG TIMER (Cont'd)

8.1.7 Register Description

WATCHDOG REGISTER (WDGR)

Address: 0D8h - Read/Write

Reset Value: 1111 1110 (FEh)

7							0
T0	T1	T2	T3	T4	T5	SR	C

Bits 7:2 = **T[5:0]** *Downcounter bits*
Caution: These bits are reversed and shifted with respect to the physical counter: bit-7 (T0) is the LSB of the Watchdog downcounter and bit-2 (T5) is the MSB.

Bit 1 = **SR**: *Software Reset bit*
Software can generate a reset by clearing this bit while the C bit is set. When C = 0 (Watchdog deactivated) the SR bit is the MSB of the 7-bit timer.
0: Generate (write)

1: No software reset generated, MSB of 7-bit timer

Bit 0 = **C** *Watchdog Control bit*.
If the hardware option is selected (WDACT bit in Option byte), this bit is forced high and cannot be changed by the user (the Watchdog is always active). When the software option is selected (WDACT bit in Option byte), the Watchdog function is activated by setting the C bit, and cannot then be deactivated (except by resetting the MCU).

When C is kept cleared the counter can be used as a 7-bit timer.
0: Watchdog deactivated
1: Watchdog activated

8.2 8-BIT TIMER

8.2.1 Introduction

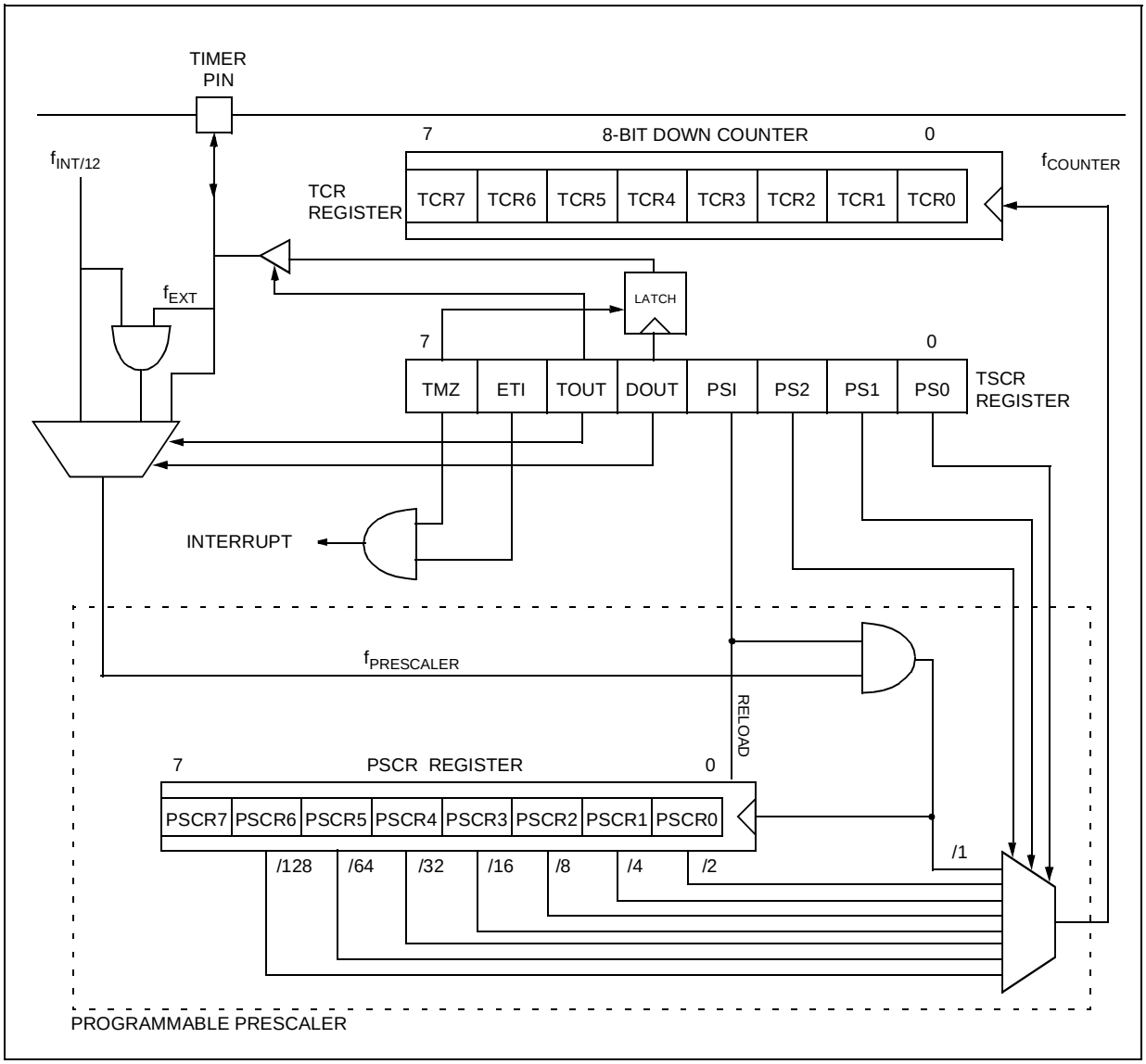
The 8-Bit Timer on-chip peripheral is a free running downcounter based on an 8-bit downcounter with a 7-bit programmable prescaler, giving a maximum count of 2^{15}. The peripheral may be configured in three different operating modes.

8.2.2 Main Features

- Time-out downcounting mode with up to 15-bit accuracy
- External counter clock source (valid also in STOP mode)
- Interrupt capability on counter underflow
- Output signal generation
- External pulse length measurement
- Event counter

The timer can be used in WAIT and STOP modes to wake up the MCU.

Figure 27. Timer Block Diagram



8-BIT TIMER (Cont'd)

8.2.3 Counter/Prescaler Description

Prescaler

The prescaler input can be the internal frequency f_{INT} divided by 12 or an external clock applied to the TIMER pin. The prescaler decrements on the rising edge, depending on the division factor programmed by the PS[2:0] bits in the TSCR register.

The state of the 7-bit prescaler can be read in the PSCR register.

When the prescaler reaches 0, it is automatically reloaded with 7Fh.

Counter

The free running 8-bit downcounter is fed by the output of the programmable prescaler, and is decremented on every rising edge of the $f_{COUNTER}$ clock signal coming from the prescaler.

It is possible to read or write the contents of the counter on the fly, by reading or writing the timer counter register (TCR).

When the downcounter reaches 0, it is automatically reloaded with the value 0FFh.

Counter Clock and Prescaler

The counter clock frequency is given by:

$$f_{COUNTER} = f_{PRESCALER} / 2^{PS[2:0]}$$

where $f_{PRESCALER}$ can be:

- $f_{INT}/12$
- f_{EXT} (input on TIMER pin)
- $f_{INT}/12$ gated by TIMER pin

The timer input clock feeds the 7-bit programmable prescaler. The prescaler output can be programmed by selecting one of the 8 available prescaler taps using the PS[2:0] bits in the Status/Control Register (TSCR). Thus the division factor of the prescaler can be set to 2^n (where n equals 0, to 7). See Figure 27.

The clock input is enabled by the PSI (Prescaler Initialize) bit in the TSCR register. When PSI is reset, the counter is frozen and the prescaler is loaded with the value 7Fh. When PSI is set, the prescaler and the counter run at the rate of the selected clock source.

Counter and Prescaler Initialization

After RESET, the counter and the prescaler are initialized to 0FFh and 7Fh respectively.

The 7-bit prescaler can be initialized to 7Fh by clearing the PSI bit. Direct write access to the prescaler is also possible when PSI =1. Then, any value between 0 and 7Fh can be loaded into it.

The 8-bit counter can be initialized separately by writing to the TCR register.

8.2.3.1 8-bit Counting and Interrupt Capability on Counter Underflow

Whatever the division factor defined for the prescaler, the Timer Counter works as an 8-bit downcounter. The input clock frequency is user selectable using the PS[2:0] bits.

When the downcounter decrements to zero, the TMZ (Timer Zero) bit in the TSCR is set. If the ETI (Enable Timer Interrupt) bit in the TSCR is also set, an interrupt request is generated.

The Timer interrupt can be used to exit the MCU from WAIT or STOP mode.

The TCR can be written at any time by software to define a time period ending with an underflow event, and therefore manage delay or timer functions.

TMZ is set when the downcounter reaches zero; however, it may also be set by writing 00h in the TCR register or by setting bit 7 of the TSCR register.

The TMZ bit must be cleared by user software when servicing the timer interrupt to avoid undesired interrupts when leaving the interrupt service routine.

Note: A write to the TCR register will predominate over the 8-bit counter decrement to 00h function, i.e. if a write and a TCR register decrement to 00h occur simultaneously, the write will take precedence, and the TMZ bit is not set until the 8-bit counter underflows again.

8-BIT TIMER (Cont'd)

8.2.4 Functional Description

There are three operating modes, which are selected by the TOUT and DOUT bits (see TSCR register). These three modes correspond to the two clocks which can be connected to the 7-bit prescaler (f_{INT} ÷ 12 or TIMER pin signal), and to the output mode.

The settings for the different operating modes are summarized Table 13.

Table 13. Timer Operating Modes

TOUT	DOUT	Timer Function	Application
0	0	Event Counter (input)	External counter clock source
0	1	Gated input (input)	External Pulse length measurement
1	0	Output "0" (output)	Output signal generation
1	1	Output "1" (output)	

8.2.4.1 Gated Mode

(TOUT = "0", DOUT = "1")

In this mode, the prescaler is decremented by the Timer clock input, but only when the signal on the TIMER pin is held high (f_{INT}/12 gated by TIMER pin). See Figure 28 and Figure 29.

This mode is selected by clearing the TOUT bit in the TSCR register (i.e. as input) and setting the DOUT bit.

Note: In this mode, if the TIMER pin is multiplexed, the corresponding port control bits have to be set in input with pull-up configuration through the DDR, OR and DR registers. For more details, please refer to the I/O Ports section.

Figure 28. f_{TIMER} Clock in Gated Mode

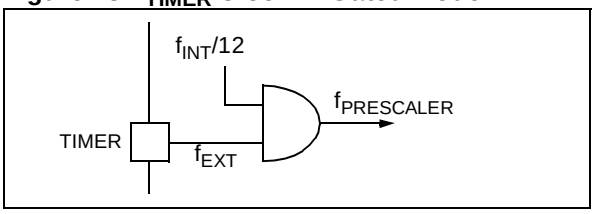
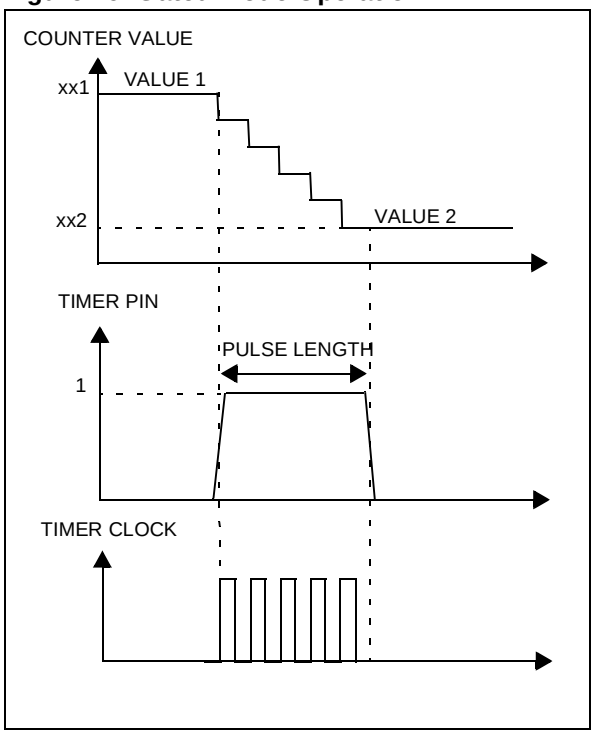


Figure 29. Gated Mode Operation



8-BIT TIMER (Cont'd)

8.2.4.2 Event Counter Mode

(TOUT = "0", DOUT = "0")

In this mode, the TIMER pin is the input clock of the Timer prescaler which is decremented on every rising edge of the input clock (allowing event count). See Figure 30 and Figure 31.

This mode is selected by clearing the TOUT bit in the TSCR register (i.e. as input) and clearing the DOUT bit.

Note: In this mode, if the TIMER pin is multiplexed, the corresponding port control bits have to be set in input with pull-up configuration.

Figure 30. f_{TIMER} Clock in Event Counter Mode

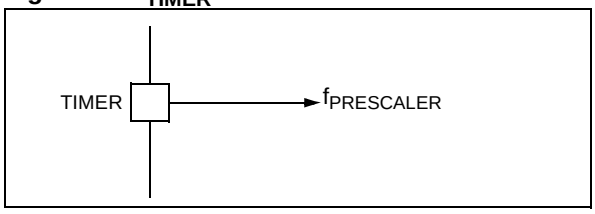
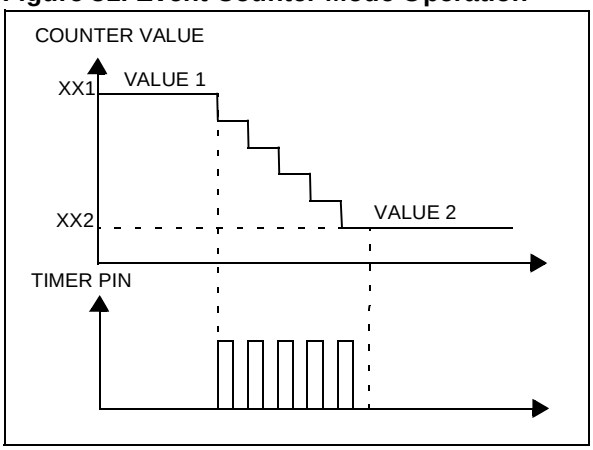


Figure 31. Event Counter Mode Operation



8.2.4.3 Output Mode

(TOUT = "1", DOUT = "data out")

In Output mode, the TIMER pin is connected to the DOUT latch, hence the Timer prescaler is clocked by the prescaler clock input (f_{INT}/12). See Figure 32.

The user can select the prescaler division ratio using the PS[2:0] bits in the TSCR register. When TCR decrements to zero, it sets the TMZ bit in the TSCR. The TMZ bit can be tested under program control to perform a timer function whenever it goes high and has to be cleared by the user. The low-to-high TMZ bit transition is used to latch the DOUT bit in the TSCR and, if the TOUT bit is set, DOUT is transferred to the TIMER pin. This operating mode allows external signal generation on the TIMER pin. See Figure 33.

This mode is selected by setting the TOUT bit in the TSCR register (i.e. as output) and setting the DOUT bit to output a high level or clearing the DOUT bit to output a low level.

Note: As soon as the TOUT bit is set, The timer pin is configured as output push-pull regardless of the corresponding I/O port control registers setting (if the TIMER pin is multiplexed).

Figure 32. Output Mode Control

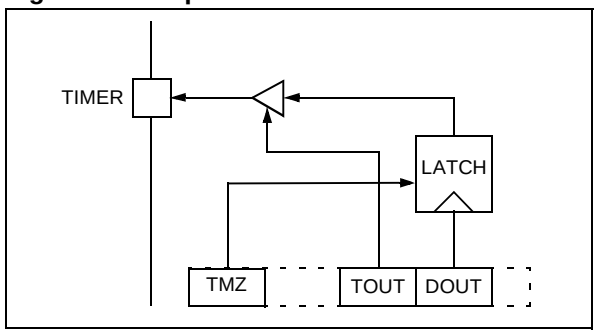
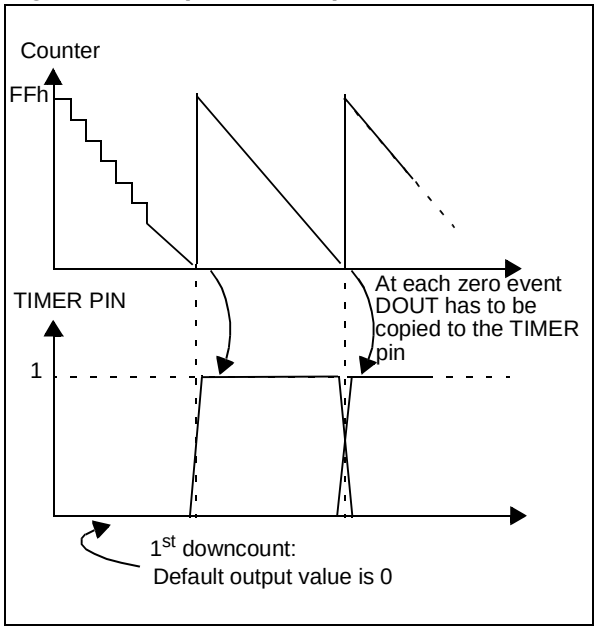


Figure 33. Output Mode Operation



8-BIT TIMER (Cont'd)

8.2.5 Low Power Modes

Mode	Description
WAIT	No effect on timer. Timer interrupt events cause the device to exit from WAIT mode.
STOP	Timer registers are frozen except in Event Counter mode (with external clock on TIMER pin).

8.2.6 Interrupts

Interrupt Event	Event Flag	Enable Bit	Exit from Wait	Exit from Stop
Timer Zero Event	TMZ	ETI	Yes	Yes

8-BIT TIMER (Cont'd)

8.2.7 Register Description

PRESCALER COUNTER REGISTER (PSCR)
Address: 0D2h - Read/Write
Reset Value: 0111 1111 (7Fh)

7							0
PSCR 7	PSCR 6	PSCR 5	PSCR 4	PSCR 3	PSCR 2	PSCR 1	PSCR 0

Bit 7 = **PSCR7:** Not used, always read as "0".

Bits 6:0 = **PSCR[6:0]** *Prescaler LSB.*

TIMER COUNTER REGISTER (TCR)
Address: 0D3h - Read / Write
Reset Value: 1111 1111 (FFh)

7							0
TCR7	TCR6	TCR5	TCR4	TCR3	TCR2	TCR1	TCR0

Bits 7:0 = **TCR[7:0]** *Timer counter bits.*

TIMER STATUS CONTROL REGISTER (TSCR)
Address: 0D4h - Read/Write
Reset Value: 0000 0000 (00h)

7							0
TMZ	ETI	TOUT	DOUT	PSI	PS2	PS1	PS0

Bit 7 = **TMZ** *Timer Zero bit.*
A low-to-high transition indicates that the timer count register has underflowed. It means that the TCR value has changed from 00h to FFh.
This bit must be cleared by user software.
0: Counter has not underflowed
1: Counter underflow occurred

Bit 6 = **ETI** *Enable Timer Interrupt.*
When set, enables the timer interrupt request. If ETI=0 the timer interrupt is disabled. If ETI=1 and TMZ=1 an interrupt request is generated.
0: Interrupt disabled (reset state)
1: Interrupt enabled

Bit 5 = **TOUT** Timer Output Control.
When low, this bit selects the input mode for the TIMER pin. When high the output mode is selected.
0: Input mode (reset state)
1: Output mode, the TIMER pin is configured as push-pull output

Bit 4 = **DOUT** *Data Output.*
Data sent to the timer output when TMZ is set high (output mode only). Input mode selection (input mode only).

Bit 3 = **PSI**: *Prescaler Initialize bit.*
Used to initialize the prescaler and inhibit its counting. When PSI="0" the prescaler is set to 7Fh and the counter is inhibited. When PSI="1" the prescaler is enabled to count downwards. As long as PSE="1" both counter and prescaler are not running
0: Counting disabled
1: Counting enabled

Bits 1:0 = **PS[2:0]** *Prescaler Mux. Select.*
These bits select the division ratio of the prescaler register.

Table 14. Prescaler Division Factors

PS2	PS1	PS0	Divided by
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Table 15. 8-Bit Timer Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0D2h	**PSCR** Reset Value	PSCR7 0	PSCR6 1	PSCR5 1	PSCR4 1	PSCR3 1	PSCR2 1	PSCR1 1	PSCR0 1
0D3h	**TCR** Reset Value	TCR7 1	TCR6 1	TCR5 1	TCR4 1	TCR3 1	TCR2 1	TCR1 1	TCR0 1
0D4h	**TSCR** Reset Value	TMZ 0	ETI 0	TOUT 0	DOUT 0	PSI 0	PS2 0	PS1 0	PS0 0

8.3 A/D CONVERTER (ADC)

8.3.1 Introduction

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

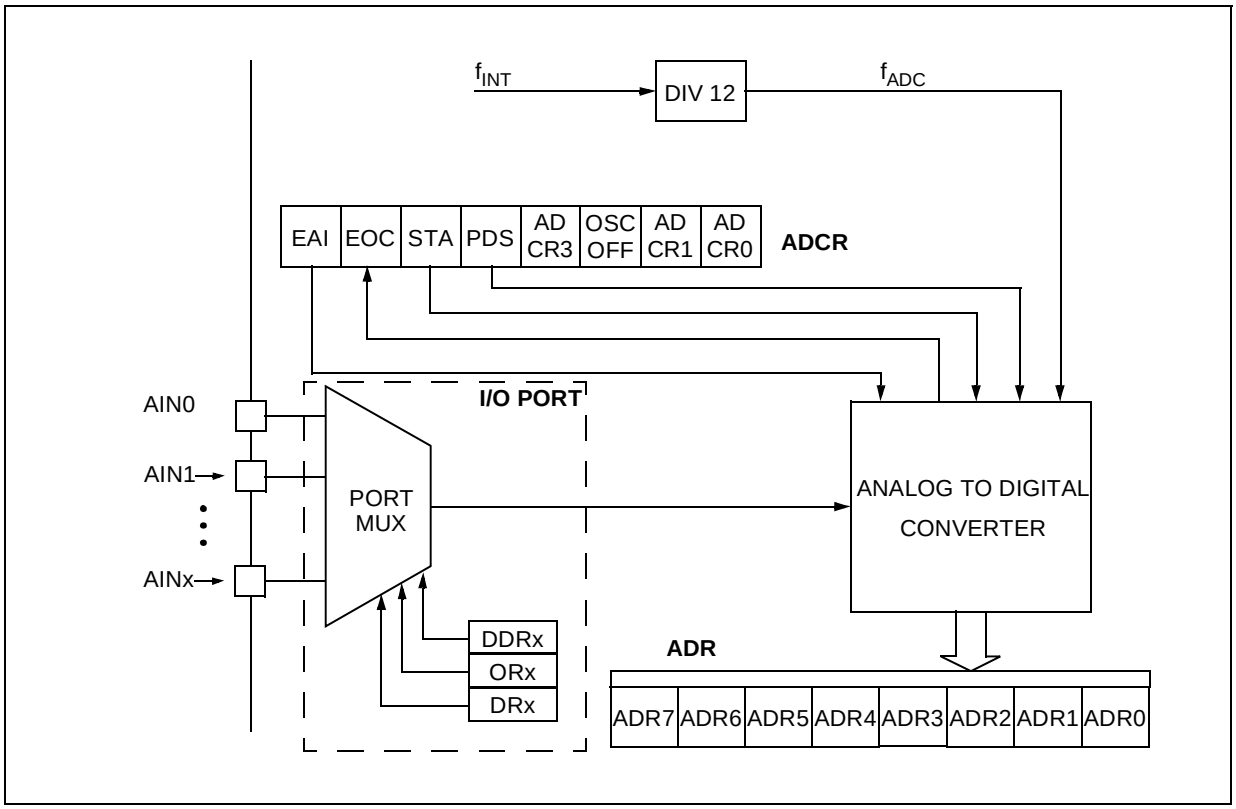
The result of the conversion is stored in a 8-bit Data Register. The A/D converter is controlled through a Control Register.

8.3.2 Main Features

- 8-bit conversion
- Multiplexed analog input channels
- Linear successive approximation
- Data register (DR) which contains the results
- End of Conversion flag
- On/Off bit (to reduce consumption)
- Typical conversion time 70 µs (with an 8 MHz crystal)

The block diagram is shown in Figure 34.

Figure 34. ADC Block Diagram



Note: ADC not present on some devices. See device summary on page 1.

A/D CONVERTER (Cont'd)

8.3.3 Functional Description

8.3.3.1 Analog Power Supply

The high and low level reference voltage pins 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.

8.3.3.2 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 or equal to V_{DDA} (high-level voltage reference) then the conversion result in the DR register is FFh (full scale) without overflow indication.

If input voltage (V_{AIN}) is lower than or equal to V_{SSA} (low-level voltage reference) then the conversion result in the DR register is 00h.

The A/D converter is linear and the digital result of the conversion is stored in the ADR register. The accuracy of the conversion is described in the parametric 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 allocated time. Refer to the electrical characteristics chapter for more details.

With an oscillator clock frequency less than 1.2MHz, conversion accuracy is decreased.

8.3.3.3 Analog Input Selection

Selection of the input pin is done by configuring the related I/O line as an analog input via the Data Direction, Option and Data registers (refer to I/O ports description for additional information).

Caution: Only one I/O line must be configured as an analog input at any time. The user must avoid any situation in which more than one I/O pin is selected as an analog input simultaneously, because they will be shorted internally.

8.3.3.4 Software Procedure

Refer to the Control register (ADCR) and Data register (ADR) in Section 8.3.7 for the bit definitions.

Analog Input Configuration

The analog input must be configured through the Port Control registers (DDRx, ORx and DRx). Refer to the I/O port chapter.

ADC Configuration

In the ADCR register:

– Reset the PDS bit to power on the ADC. This bit must be set at least one instruction before the beginning of the conversion to allow stabilisation of the A/D converter.

– Set the EAI bit to enable the ADC interrupt if needed.

ADC Conversion

In the ADCR register:

– Set the STA bit to start a conversion. This automatically clears (resets to "0") the End Of Conversion Bit (EOC).

When a conversion is complete

– The EOC bit is set by hardware to flag that conversion is complete and that the data in the ADC data conversion register is valid.

– An interrupt is generated if the EAI bit was set

Setting the STA bit will start a new count and will clear the EOC bit (thus clearing the interrupt condition)

Note:

Setting the STA bit must be done by a different instruction from the instruction that powers-on the ADC (setting the PDS bit) in order to make sure the voltage to be converted is present on the pin.

Each conversion has to be separately initiated by writing to the STA bit.

The STA bit is continuously scanned so that, if the user sets it to "1" while a previous conversion is in progress, a new conversion is started before completing the previous one. The start bit (STA) is a write only bit, any attempt to read it will show a logical "0".

A/D CONVERTER (Cont'd)

8.3.4 Recommendations

The following six notes provide additional information on using the A/D converter.

1.The A/D converter does not feature a sample and hold circuit. The analog voltage to be measured should therefore be stable during the entire conversion cycle. Voltage variation should not exceed ±1/2 LSB for optimum conversion accuracy. A low pass filter may be used at the analog input pins to reduce input voltage variation during conversion.

2. When selected as an analog channel, the input pin is internally connected to a capacitor C_{ad} of typically 9pF. For maximum accuracy, this capacitor must be fully charged at the beginning of conversion. In the worst case, conversion starts one instruction (6.5 µs) after the channel has been selected. The impedance of the analog voltage source (ASI) in worst case conditions, is calculated using the following formula:

$$6.5\mu s = 9 \times C_{ad} \times ASI$$

(capacitor charged to over 99.9%), i.e. 30 kΩ including a 50% guardband.
The ASI can be higher if C_{ad} has been charged for a longer period by adding instructions before the start of conversion (adding more than 26 CPU cycles is pointless).

3. Since the ADC is on the same chip as the microprocessor, the user should not switch heavily loaded output signals during conversion, if high precision is required. Such switching will affect the supply voltages used as analog references.

4. Conversion accuracy depends on the quality of the power supplies (V_{DD} and V_{SS}). The user must take special care to ensure a well regulated reference voltage is present on the V_{DD} and V_{SS} pins (power supply voltage variations must be less than 0.1V/ms). This implies, in particular, that a suitable decoupling capacitor is used at the V_{DD} pin.
The converter resolution is given by:

$$\frac{V_{DD} - V_{SS}}{256}$$

The Input voltage (Ain) which is to be converted must be constant for 1µs before conversion and remain constant during conversion.

5. Conversion resolution can be improved if the power supply voltage (V_{DD}) to the microcontroller is lowered.

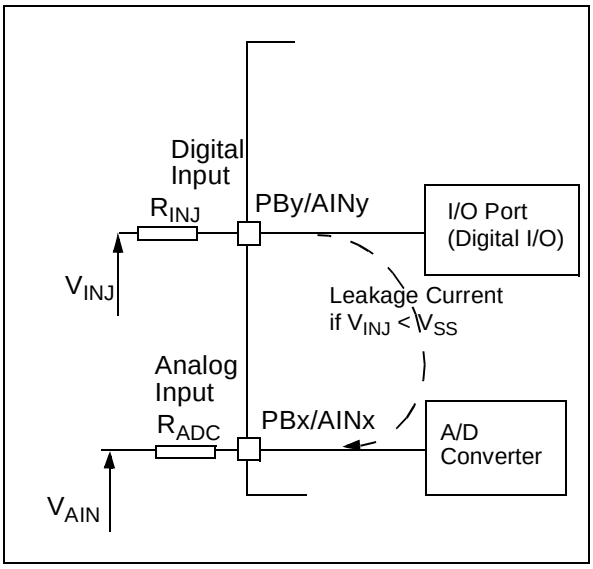
6. In order to optimize the conversion resolution, the user can configure the microcontroller in WAIT mode, because this mode minimises noise disturbances and power supply variations due to output switching. Nevertheless, the WAIT instruction should be executed as soon as possible after the beginning of the conversion, because execution of the WAIT instruction may cause a small variation of the V_{DD} voltage. The negative effect of this variation is minimized at the beginning of the conversion when the converter is less sensitive, rather than at the end of conversion, when the least significant bits are determined.
The best configuration, from an accuracy standpoint, is WAIT mode with the Timer stopped. In this case only the ADC peripheral and the oscillator are then still working. The MCU must be woken up from WAIT mode by the ADC interrupt at the end of the conversion. The microcontroller can also be woken up by the Timer interrupt, but this means the Timer must be running and the resulting noise could affect conversion accuracy.

Caution: When an I/O pin is used as an analog input, A/D conversion accuracy will be impaired if negative current injections ($V_{INJ} < V_{SS}$) occur from adjacent I/O pins with analog input capability. Refer to Figure 35. To avoid this:

– Use another I/O port located further away from the analog pin, preferably not multiplexed on the A/D converter

– Increase the input resistance $R_{IN\ J}$ (to reduce the current injections) and reduce R_{ADC} (to preserve conversion accuracy).

Figure 35. Leakage from Digital Inputs



A/D CONVERTER (Cont'd)

8.3.5 Low Power Modes

Mode	Description
WAIT	No effect on A/D Converter. ADC interrupts cause the device to exit from Wait mode.
STOP	A/D Converter disabled.

Note: The A/D converter may be disabled by clearing the PDS bit. This feature allows reduced power consumption when no conversion is needed.

8.3.6 Interrupts

Interrupt Event	Event Flag	Enable Bit	Exit from Wait	Exit from Stop
End of Conversion	EOC	EAI	Yes	No

Note: The EOC bit is cleared only when a new conversion is started (it cannot be cleared by writing 0). To avoid generating further EOC interrupt, the EAI bit has to be cleared within the ADC interrupt subroutine.

8.3.7 Register Description

A/D CONVERTER CONTROL REGISTER (ADCR)

Address: 0D1h - Read/Write (Bit 6 Read Only, Bit 5 Write Only)

Reset value: 0100 0000 (40h)

7							0
EAI	EOC	STA	PDS	ADCR 3	OSC OFF	ADCR 1	ADCR 0

Bit 7 = **EAI** *Enable A/D Interrupt.*
0: ADC interrupt disabled
1: ADC interrupt enabled

Bit 6 = **EOC** *End of conversion. Read Only*
When a conversion has been completed, this bit is set by hardware and an interrupt request is generated if the EAI bit is set. The EOC bit is automatically cleared when the STA bit is set. Data in the data conversion register are valid only when this bit is set to "1".
0: Conversion is not complete
1: Conversion can be read from the ADR register

Bit 5 = **STA***: Start of Conversion. Write Only.*
0: No effect
1: Start conversion

Note: Setting this bit automatically clears the EOC bit. If the bit is set again when a conversion is in progress, the present conversion is stopped and a new one will take place. This bit is write only, any attempt to read it will show a logical zero.

Bit 4 = **PDS** *Power Down Selection.*

0: A/D converter is switched off
1: A/D converter is switched on

Bit 3 = **ADCR3 Reserved**, must be cleared.

Bit 2 = **OSCOFF** *Main Oscillator off.*

0: Main Oscillator enabled
1: Main Oscillator disabled

Note: This bit does not apply to the ADC peripheral but to the main clock system. Refer to the Clock System section.

Bits 1:0 = **ADCR[1:0] Reserved**, must be cleared.

A/D CONVERTER DATA REGISTER (ADR)

Address: 0D0h - Read only

Reset value: xxxx xxxx (xxh)

7							0
ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0

Bits 7:0 = **ADR[7:0]***: 8 Bit A/D Conversion Result.*

Table 16. ADC Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0D0h	**ADR** Reset Value	ADR7 0	ADR6 0	ADR5 0	ADR4 0	ADR3 0	ADR2 0	ADR1 0	ADR0 0
0D1h	**ADCR** Reset Value	EAI 0	EOC 1	STA 0	PDS 0	ADCR3 0	OSCOFF 0	ADCR1 0	ADCR0 0

9 INSTRUCTION SET

9.1 ST6 ARCHITECTURE

The ST6 architecture has been designed for maximum efficiency while keeping byte usage to a minimum; in short, to provide byte-efficient programming. The ST6 core has the ability to set or clear any register or RAM location bit in Data space using a single instruction. Furthermore, programs can branch to a selected address depending on the status of any bit in Data space.

9.2 ADDRESSING MODES

The ST6 has nine addressing modes, which are described in the following paragraphs. Three different address spaces are available: Program space, Data space, and Stack space. Program space contains the instructions which are to be executed, plus the data for immediate mode instructions. Data space contains the Accumulator, the X, Y, V and W registers, peripheral and Input/Output registers, the RAM locations and Data ROM locations (for storage of tables and constants). Stack space contains six 12-bit RAM cells used to stack the return addresses for subroutines and interrupts.

Immediate. In immediate addressing mode, the operand of the instruction follows the opcode location. As the operand is a ROM byte, the immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

Direct. In direct addressing mode, the address of the byte which is processed by the instruction is stored in the location which follows the opcode. Direct addressing allows the user to directly address the 256 bytes in Data Space memory with a single two-byte instruction.

Short Direct. The core can address the four RAM registers X, Y, V, W (locations 80h, 81h, 82h, 83h) in short-direct addressing mode. In this case, the instruction is only one byte and the selection of the location to be processed is contained in the opcode. Short direct addressing is a subset of direct addressing mode. (Note that 80h and 81h are also indirect registers).

Extended. In extended addressing mode, the 12-bit address needed to define the instruction is obtained by concatenating the four least significant bits of the opcode with the byte following the opcode. The instructions (JP, CALL) which use extended addressing mode are able to branch to any address in the 4 Kbyte Program space.

Extended addressing mode instructions are two bytes long.

Program Counter Relative. Relative addressing mode is only used in conditional branch instructions. The instruction is used to perform a test and, if the condition is true, a branch with a span of -15 to +16 locations next to the address of the relative instruction. If the condition is not true, the instruction which follows the relative instruction is executed. Relative addressing mode instructions are one byte long. The opcode is obtained by adding the three most significant bits which characterize the test condition, one bit which determines whether it is a forward branch (when it is 0) or backward branch (when it is 1) and the four least significant bits which give the span of the branch (0h to Fh) which must be added or subtracted from the address of the relative instruction to obtain the branch destination address.

Bit Direct. In bit direct addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode points to the address of the byte in which the specified bit must be set or cleared. Thus, any bit in the 256 locations of Data space memory can be set or cleared.

Bit Test & Branch. Bit test and branch addressing mode is a combination of direct addressing and relative addressing. Bit test and branch instructions are three bytes long. The bit identification and the test condition are included in the opcode byte. The address of the byte to be tested is given in the next byte. The third byte is the jump displacement, which is in the range of -127 to +128. This displacement can be determined using a label, which is converted by the assembler.

Indirect. In indirect addressing mode, the byte processed by the register-indirect instruction is at the address pointed to by the content of one of the indirect registers, X or Y (80h, 81h). The indirect register is selected by bit 4 of the opcode. Register indirect instructions are one byte long.

Inherent. In inherent addressing mode, all the information necessary for executing the instruction is contained in the opcode. These instructions are one byte long.

9.3 INSTRUCTION SET

The ST6 offers a set of 40 basic instructions which, when combined with nine addressing modes, yield 244 usable opcodes. They can be divided into six different types: load/store, arithmetic/logic, conditional branch, control instructions, jump/call, and bit manipulation. The following paragraphs describe the different types.

All the instructions belonging to a given type are presented in individual tables.

Load & Store. These instructions use one, two or three bytes depending on the addressing mode. For LOAD, one operand is the Accumulator and the other operand is obtained from data memory using one of the addressing modes.

For Load Immediate, one operand can be any of the 256 data space bytes while the other is always immediate data.

Table 17. Load & Store Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
LD A, X	Short Direct	1	4	Δ	*
LD A, Y	Short Direct	1	4	Δ	*
LD A, V	Short Direct	1	4	Δ	*
LD A, W	Short Direct	1	4	Δ	*
LD X, A	Short Direct	1	4	Δ	*
LD Y, A	Short Direct	1	4	Δ	*
LD V, A	Short Direct	1	4	Δ	*
LD W, A	Short Direct	1	4	Δ	*
LD A, rr	Direct	2	4	Δ	*
LD rr, A	Direct	2	4	Δ	*
LD A, (X)	Indirect	1	4	Δ	*
LD A, (Y)	Indirect	1	4	Δ	*
LD (X), A	Indirect	1	4	Δ	*
LD (Y), A	Indirect	1	4	Δ	*
LDI A, #N	Immediate	2	4	Δ	*
LDI rr, #N	Immediate	3	4	*	*

Legend:

X, Y Index Registers,
V, W Short Direct Registers
Immediate data (stored in ROM memory)
rr Data space register
Δ Affected
* Not Affected

INSTRUCTION SET (Cont'd)

Arithmetic and Logic. These instructions are used to perform arithmetic calculations and logic operations. In AND, ADD, CP, SUB instructions one operand is always the accumulator while, depending on the addressing mode, the other can be either a data space memory location or an immediate value. In CLR, DEC, INC instructions the operand can be any of the 256 data space addresses. In COM, RLC, SLA the operand is always the accumulator.

Table 18. Arithmetic & Logic Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
ADD A, (X)	Indirect	1	4	Δ	Δ
ADD A, (Y)	Indirect	1	4	Δ	Δ
ADD A, rr	Direct	2	4	Δ	Δ
ADDI A, #N	Immediate	2	4	Δ	Δ
AND A, (X)	Indirect	1	4	Δ	Δ
AND A, (Y)	Indirect	1	4	Δ	Δ
AND A, rr	Direct	2	4	Δ	Δ
ANDI A, #N	Immediate	2	4	Δ	Δ
CLR A	Short Direct	2	4	Δ	Δ
CLR r	Direct	3	4	*	*
COM A	Inherent	1	4	Δ	Δ
CP A, (X)	Indirect	1	4	Δ	Δ
CP A, (Y)	Indirect	1	4	Δ	Δ
CP A, rr	Direct	2	4	Δ	Δ
CPI A, #N	Immediate	2	4	Δ	Δ
DEC X	Short Direct	1	4	Δ	*
DEC Y	Short Direct	1	4	Δ	*
DEC V	Short Direct	1	4	Δ	*
DEC W	Short Direct	1	4	Δ	*
DEC A	Direct	2	4	Δ	*
DEC rr	Direct	2	4	Δ	*
DEC (X)	Indirect	1	4	Δ	*
DEC (Y)	Indirect	1	4	Δ	*
INC X	Short Direct	1	4	Δ	*
INC Y	Short Direct	1	4	Δ	*
INC V	Short Direct	1	4	Δ	*
INC W	Short Direct	1	4	Δ	*
INC A	Direct	2	4	Δ	*
INC rr	Direct	2	4	Δ	*
INC (X)	Indirect	1	4	Δ	*
INC (Y)	Indirect	1	4	Δ	*
RLC A	Inherent	1	4	Δ	Δ
SLA A	Inherent	2	4	Δ	Δ
SUB A, (X)	Indirect	1	4	Δ	Δ
SUB A, (Y)	Indirect	1	4	Δ	Δ
SUB A, rr	Direct	2	4	Δ	Δ
SUBI A, #N	Immediate	2	4	Δ	Δ

Notes:
X,Y Index Registers
V, W Short Direct Registers
Δ Affected
Immediate data (stored in ROM memory)
* Not Affected
rr Data space register

INSTRUCTION SET (Cont'd)

Conditional Branch. Branch instructions perform a branch in the program when the selected condition is met.

Bit Manipulation Instructions. These instructions can handle any bit in Data space memory. One group either sets or clears. The other group (see Conditional Branch) performs the bit test branch operations.

Control Instructions. Control instructions control microcontroller operations during program execution.

Jump and Call. These two instructions are used to perform long (12-bit) jumps or subroutine calls to any location in the whole program space.

Table 19. Conditional Branch Instructions

Instruction	Branch If	Bytes	Cycles	Flags	
				Z	C
JRC e	C = 1	1	2	*	*
JRNC e	C = 0	1	2	*	*
JRZ e	Z = 1	1	2	*	*
JRNZ e	Z = 0	1	2	*	*
JRR b, rr, ee	Bit = 0	3	5	*	Δ
JRS b, rr, ee	Bit = 1	3	5	*	Δ

Notes:
b 3-bit address
e 5 bit signed displacement in the range -15 to +16
ee 8 bit signed displacement in the range -126 to +129
rr Data space register
Δ Affected. The tested bit is shifted into carry.
* Not Affected

Table 20. Bit Manipulation Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
SET b,rr	Bit Direct	2	4	*	*
RES b,rr	Bit Direct	2	4	*	*

Notes:
b 3-bit address
rr Data space register
* Not Affected
Bit Manipulation Instructions should not be used on Port Data Registers and any registers with read only and/or write only bits (see I/O port chapter)

Table 21. Control Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
NOP	Inherent	1	2	*	*
RET	Inherent	1	2	*	*
RETI	Inherent	1	2	Δ	Δ
STOP (1)	Inherent	1	2	*	*
WAIT	Inherent	1	2	*	*

Notes:
1. This instruction is deactivated and a WAIT is automatically executed instead of a STOP if the watchdog function is selected.
Δ Affected
*Not Affected

Table 22. Jump & Call Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
CALL abc	Extended	2	4	*	*
JP abc	Extended	2	4	*	*

Notes:
abc 12-bit address
* Not Affected

Opcode Map Summary. The following table contains an opcode map for the instructions used by the ST6

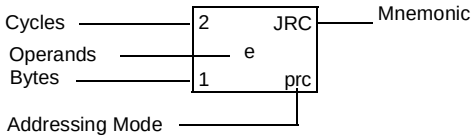
LOW / HI	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	LOW / HI
0 0000	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b0,rr,ee 3 bt	2 JRZ e NOP 1 pcr	# 	2 JRC e 1 prc	4 LD a,(x) 1 ind	**0 0000**
1 0001	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b0,rr,ee 3 bt	2 JRZ e 1 pcr	4 INC x 1 sd	2 JRC e 1 prc	4 LDI a,nn 2 imm	**1 0001**
2 0010	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b4,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 CP a,(x) 1 ind	**2 0010**
3 0011	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b4,rr,ee 3 bt	2 JRZ e 1 pcr	4 LD a,x 1 sd	2 JRC e 1 prc	4 CPI a,nn 2 imm	**3 0011**
4 0100	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b2,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 ADD a,(x) 1 ind	**4 0100**
5 0101	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b2,rr,ee 3 bt	2 JRZ e 1 pcr	4 INC y 1 sd	2 JRC e 1 prc	4 ADDI a,nn 2 imm	**5 0101**
6 0110	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b6,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 INC (x) 1 ind	**6 0110**
7 0111	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b6,rr,ee 3 bt	2 JRZ e 1 pcr	4 LD a,y 1 sd	2 JRC e 1 prc	# 	**7 0111**
8 1000	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b1,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 LD (x),a 1 ind	**8 1000**
9 1001	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b1,rr,ee 3 bt	2 JRZ e 1 pcr	4 INC v 1 sd	2 JRC e 1 prc	# 	**9 1001**
A 1010	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b5,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 AND a,(x) 1 ind	**A 1010**
B 1011	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b5,rr,ee 3 bt	2 JRZ e 1 pcr	4 LD a,v 1 sd	2 JRC e 1 prc	4 ANDI a,nn 2 imm	**B 1011**
C 1100	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b3,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 SUB a,(x) 1 ind	**C 1100**
D 1101	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b3,rr,ee 3 bt	2 JRZ e 1 pcr	4 INC w 1 sd	2 JRC e 1 prc	4 SUBI a,nn 2 imm	**D 1101**
E 1110	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRR b7,rr,ee 3 bt	2 JRZ e 1 pcr	# 	2 JRC e 1 prc	4 DEC (x) 1 ind	**E 1110**
F 1111	2 JRNZ e 1 pcr	4 CALL abc 2 ext	2 JRNC e 1 pcr	5 JRS b7,rr,ee 3 bt	2 JRZ e 1 pcr	4 LD a,w 1 sd	2 JRC e 1 prc	# 	**F 1111**

Abbreviations for Addressing Modes:

dir	Direct
sd	Short Direct
imm	Immediate
inh	Inherent
ext	Extended
b.d	Bit Direct
bt	Bit Test
pcr	Program Counter Relative
ind	Indirect

Legend:

#	Indicates Illegal Instructions
e	5-bit Displacement
b	3-bit Address
rr	1-byte Data space address
nn	1-byte immediate data
abc	12-bit address
ee	8-bit displacement



Opcode Map Summary (Continued)

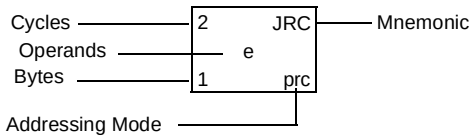
LOW / HI	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	LOW / HI
0 **0000**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b0,rr 2 b.d	2 JRZ e 1 pcr	4 LDI rr,nn 3 imm	2 JRC e 1 prc	4 LD a,(y) 1 ind	**0** **0000**
1 **0001**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b0,rr 2 b.d	2 JRZ e 1 pcr	4 DEC x 1 sd	2 JRC e 1 prc	4 LD a,rr 2 dir	**1** **0001**
2 **0010**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b4,rr 2 b.d	2 JRZ e 1 pcr	4 COM a	2 JRC e 1 prc	4 CP a,(y) 1 ind	**2** **0010**
3 **0011**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b4,rr 2 b.d	2 JRZ e 1 pcr	4 LD x,a 1 sd	2 JRC e 1 prc	4 CP a,rr 2 dir	**3** **0011**
4 **0100**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b2,rr 2 b.d	2 JRZ e 1 pcr	2 RETI 1 inh	2 JRC e 1 prc	4 ADD a,(y) 1 ind	**4** **0100**
5 **0101**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b2,rr 2 b.d	2 JRZ e 1 pcr	4 DEC y 1 sd	2 JRC e 1 prc	4 ADD a,rr 2 dir	**5** **0101**
6 **0110**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b6,rr 2 b.d	2 JRZ e 1 pcr	2 STOP 1 inh	2 JRC e 1 prc	4 INC (y) 1 ind	**6** **0110**
7 **0111**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b6,rr 2 b.d	2 JRZ e 1 pcr	4 LD y,a 1 sd	2 JRC e 1 prc	4 INC rr 2 dir	**7** **0111**
8 **1000**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b1,rr 2 b.d	2 JRZ e 1 pcr	#	2 JRC e 1 prc	4 LD (y),a 1 ind	**8** **1000**
9 **1001**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b1,rr 2 b.d	2 JRZ e 1 pcr	4 DEC v 1 sd	2 JRC e 1 prc	4 LD rr,a 2 dir	**9** **1001**
A **1010**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b5,rr 2 b.d	2 JRZ e 1 pcr	4 RCL a 1 inh	2 JRC e 1 prc	4 AND a,(y) 1 ind	**A** **1010**
B **1011**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b5,rr 2 b.d	2 JRZ e 1 pcr	4 LD v,a 1 sd	2 JRC e 1 prc	4 AND a,rr 2 dir	**B** **1011**
C **1100**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b3,rr 2 b.d	2 JRZ e 1 pcr	2 RET 1 inh	2 JRC e 1 prc	4 SUB a,(y) 1 ind	**C** **1100**
D **1101**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b3,rr 2 b.d	2 JRZ e 1 pcr	4 DEC w 1 sd	2 JRC e 1 prc	4 SUB a,rr 2 dir	**D** **1101**
E **1110**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b7,rr 2 b.d	2 JRZ e 1 pcr	2 WAIT 1 inh	2 JRC e 1 prc	4 DEC (y) 1 ind	**E** **1110**
F **1111**	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b7,rr 2 b.d	2 JRZ e 1 pcr	4 LD w,a 1 sd	2 JRC e 1 prc	4 DEC rr 2 dir	**F** **1111**

Abbreviations for Addressing Modes:

dir	Direct
sd	Short Direct
imm	Immediate
inh	Inherent
ext	Extended
b.d	Bit Direct
bt	Bit Test
pcr	Program Counter Relative
ind	Indirect

Legend:

#	Indicates Illegal Instructions
e	5-bit Displacement
b	3-bit Address
rr	1-byte Data space address
nn	1-byte immediate data
abc	12-bit address
ee	8-bit Displacement



10 ELECTRICAL CHARACTERISTICS

10.1 PARAMETER CONDITIONS

Unless otherwise specified, all voltages are referred to V_{SS}.

10.1.1 Minimum and Maximum Values

Unless otherwise specified the minimum and maximum values are guaranteed in the worst conditions of ambient temperature, supply voltage and frequencies by tests in production on 100% of the devices with an ambient temperature at T_A=25°C and T_A=$T_A$max (given by the selected temperature range).

Data based on characterization results, design simulation and/or technology characteristics are indicated in the table footnotes and are not tested in production. Based on characterization, the minimum and maximum values refer to sample tests and represent the mean value plus or minus three times the standard deviation (mean±3Σ).

10.1.2 Typical Values

Unless otherwise specified, typical data are based on T_A=25°C, V_{DD}=5V (for the 4.5V≤V_{DD}≤6.0V voltage range) and V_{DD}=3.3V (for the 3V≤V_{DD}≤3.6V voltage range). They are given only as design guidelines and are not tested.

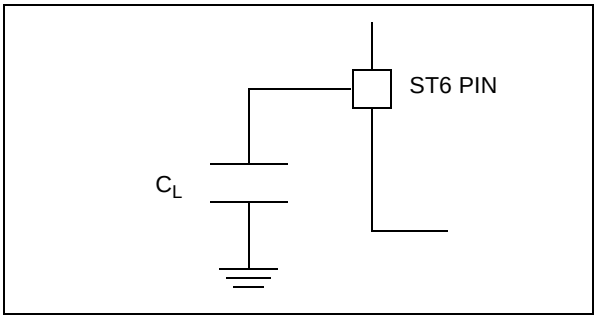
10.1.3 Typical Curves

Unless otherwise specified, all typical curves are given only as design guidelines and are not tested.

10.1.4 Loading Capacitor

The loading conditions used for pin parameter measurement is shown in Figure 36.

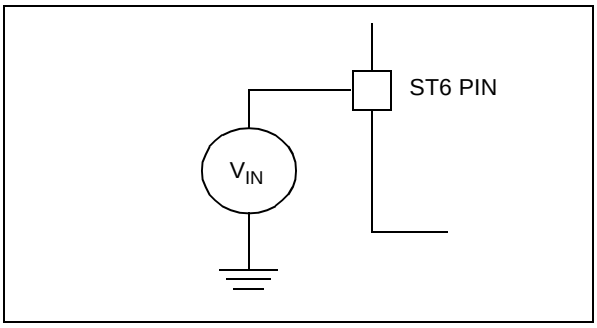
Figure 36. Pin Loading Conditions



10.1.5 Pin Input Voltage

The input voltage measurement on a pin of the device is described in Figure 37.

Figure 37. Pin Input Voltage



10.2 ABSOLUTE MAXIMUM RATINGS

Stresses above those listed as “absolute maximum ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

10.2.1 Voltage Characteristics

Symbol	Ratings	Maximum value	Unit
V_{DD} - V_{SS}	Supply voltage	7	V
V_{IN}	Input voltage on any pin 1) & 2)	V_{SS}-0.3 to V_{DD}+0.3	
V_{OUT}	Output voltage on any pin 1) & 2)	V_{SS}-0.3 to V_{DD}+0.3	
$V_{ESD(HBM)}$	Electro-static discharge voltage (Human Body Model)	3500	

10.2.2 Current Characteristics

Symbol	Ratings	Maximum value	Unit
I_{VDD}	Total current into V_{DD} power lines (source) 3)	80	mA
I_{VSS}	Total current out of V_{SS} ground lines (sink) 3)	100	
I_{IO}	Output current sunk by any standard I/O and control pin	20	
	Output current sunk by any high sink I/O pin	40	
	Output current source by any I/Os and control pin	15	
$I_{INJ(PIN)}$ 2) & 4)	Injected current on $\overline{RESET}$ pin	±5	
	Injected current on any other pin	±5	

10.2.3 Thermal Characteristics

Symbol	Ratings	Value	Unit
T_{STG}	Storage temperature range	-60 to +150	°C
T_J	Maximum junction temperature (see THERMAL CHARACTERISTICS section)		

Notes:

1. Directly connecting the $\overline{RESET}$ and I/O pins to V_{DD} or V_{SS} could damage the device if an unintentional internal reset is generated or an unexpected change of the I/O configuration occurs (for example, due to a corrupted program counter). To guarantee safe operation, this connection has to be done through a pull-up or pull-down resistor (typical: 4.7kΩ for $\overline{RESET}$, 10kΩ for I/Os). Unused I/O pins must be tied in the same way to V_{DD} or V_{SS} according to their reset configuration.
2. When the current limitation is not possible, the V_{IN} absolute maximum rating must be respected, otherwise refer to $I_{INJ(PIN)}$ specification. A positive injection is induced by $V_{IN}>V_{DD}$ while a negative injection is induced by $V_{IN}<V_{SS}$.
3. Power (V_{DD}) and ground (V_{SS}) lines must always be connected to the external supply.
4. Negative injection disturbs the analog performance of the device. In particular, it induces leakage currents throughout the device including the analog inputs. To avoid undesirable effects on the analog functions, care must be taken:
   - Analog input pins must have a negative injection less than 1mA (assuming that the impedance of the analog voltage is lower than the specified limits).
   - Pure digital pins must have a negative injection less than 1mA. In addition, it is recommended to inject the current as far as possible from the analog input pins.

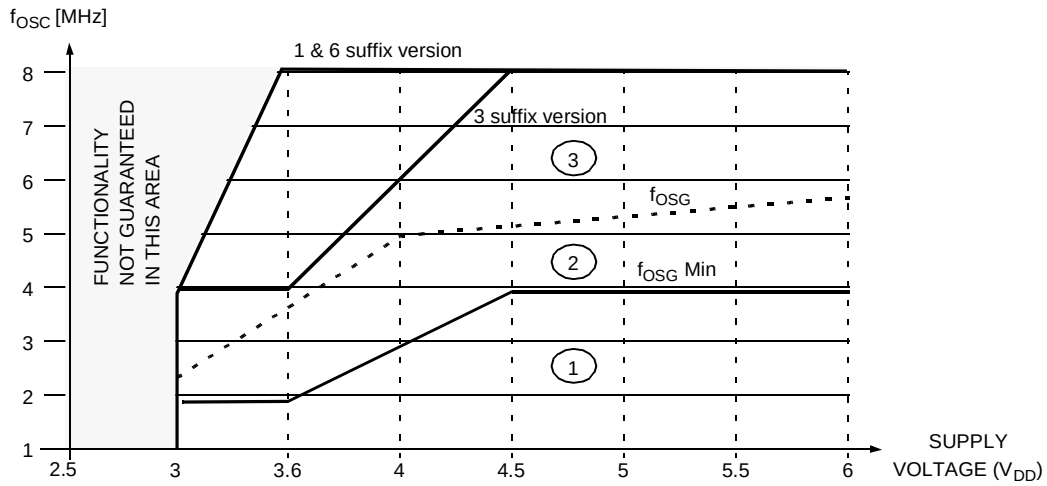
10.3 OPERATING CONDITIONS

10.3.1 General Operating Conditions

Symbol	Parameter	Conditions	Min	Max	Unit
V_{DD}	Supply voltage	see Figure 38	3.0	6	V
f_{OSC}	Oscillator frequency	V_{DD}=3.0V, 1 & 6 Suffix	0 [1)]	4	MHz
		V_{DD}=3.0V, 3 Suffix	0 [1)]	4	
		V_{DD}=3.6V, 1 & 6Suffix	0 [1)]	8	
		V_{DD}=3.6V, 3 Suffix	0 [1)]	4	
V_{DD}	Operating Supply Voltage	f_{OSC}=4MHz, 1 & 6 Suffix	3.0	6.0	V
		f_{OSC}=4MHz, 3 Suffix	3.0	6.0	
		f_{OSC}=8MHz, 1 & 6 Suffix	3.6	6.0	
		f_{OSC}=8MHz, 3 Suffix	4.5	6.0	
T_A	Ambient temperature range	1 Suffix Version	0	70	°C
		6 Suffix Version	-40	85	
		3 Suffix Version	-40	125	

Notes:
1. An oscillator frequency above 1.2MHz is recommended for reliable A/D results.
2. Operating conditions with T_A=-40 to +125°C.

Figure 38. f_{OSC} Maximum Operating Frequency Versus V_{DD} Supply Voltage for OTP & ROM devices



1. In this area, operation is guaranteed at the quartz crystal frequency.
2. When the OSG is disabled, operation in this area is guaranteed at the crystal frequency. When the OSG is enabled, operation in this area is guaranteed at a frequency of at least f_{OSG} Min.
3. When the OSG is disabled, operation in this area is guaranteed at the quartz crystal frequency. When the OSG is enabled, access to this area is prevented. The internal frequency is kept at f_{OSG}.

OPERATING CONDITIONS (Cont'd)

10.3.2 Operating Conditions with Low Voltage Detector (LVD)

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

Symbol	Parameter	Conditions	Min	Typ [1]	Max	Unit
V_{IT+}	Reset release threshold (V_{DD} rise)		3.9	4.1	4.3	V
V_{IT-}	Reset generation threshold (V_{DD} fall)		3.6	3.8	4	
V_{hys}	LVD voltage threshold hysteresis	V_{IT+}-V_{IT-}	50	300	700	mV
Vt_{POR}	V_{DD} rise time rate [2]					mV/s
$t_{g(VDD)}$	Filtered glitch delay on V_{DD} [3]	Not detected by the LVD		30		ns

Notes:

1. LVD typical data are based on T_A=25°C. They are given only as design guidelines and are not tested.
2. The minimum V_{DD} rise time rate is needed to insure a correct device power-on and LVD reset. Not tested in production.
3. Data based on characterization results, not tested in production.

Figure 39. LVD Threshold Versus V_{DD} and f_{OSC} [3]

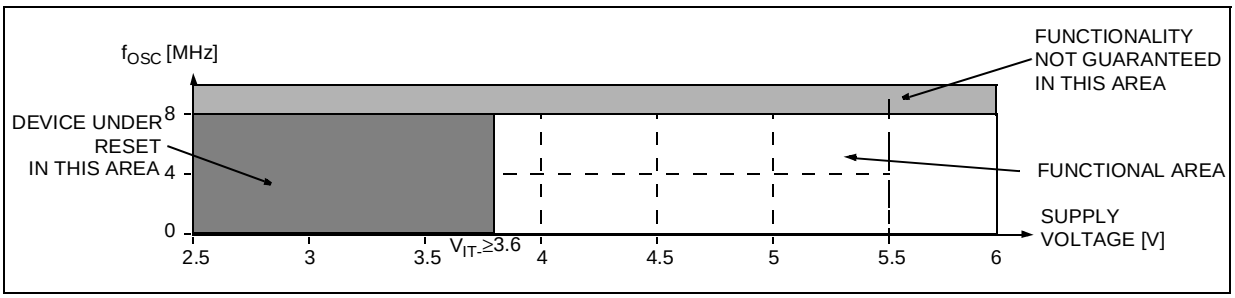


Figure 40. Typical LVD Thresholds Versus Temperature for OTP devices

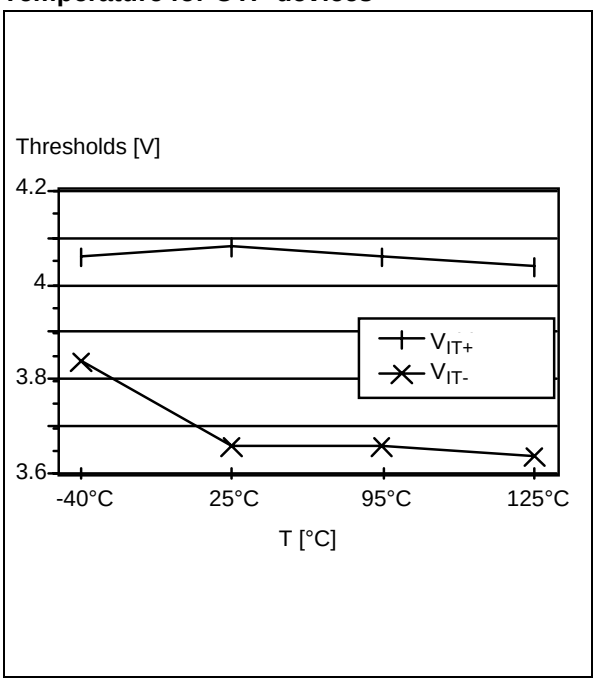
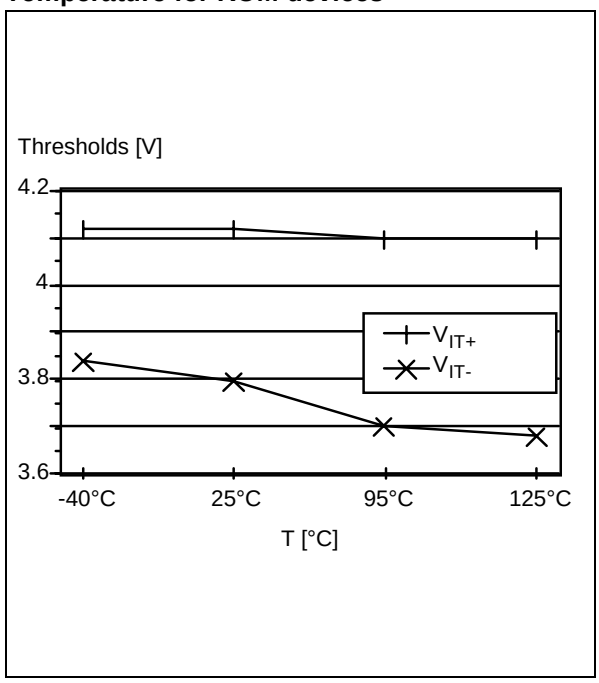


Figure 41. Typical LVD thresholds vs. Temperature for ROM devices



10.4 SUPPLY CURRENT CHARACTERISTICS

The following current consumption specified for the ST6 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 STOP mode for which the clock is stopped).

10.4.1 RUN Modes

Symbol	Parameter	Conditions		Typ [1]	Max [2]	Unit
I_{DD}	Supply current in RUN mode [3] (see Figure 42 & Figure 43)	$4.5V \le V_{DD} \le 6.0V$	f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	0.5 1.3 1.6 2.2 3.3	0.7 1.7 2.4 3.3 4.8	mA
	Supply current in RUN mode [3] (see Figure 42 & Figure 43)	$3V \le V_{DD} \le 3.6V$	f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	0.3 0.6 0.9 1.0 1.8	0.4 0.8 1.2 1.5 2.3	

Notes:

1. Typical data are based on T_A=25°C, V_{DD}=5V (4.5V≤V_{DD}≤6.0V range) and V_{DD}=3.3V (3V≤V_{DD}≤3.6V range).
2. Data based on characterization results, tested in production at V_{DD} max. and f_{OSC} max.
3. CPU running with memory access, all I/O pins in input with pull-up mode (no load), all peripherals in reset state; clock input (OSC_{IN}) driven by external square wave, OSG and LVD disabled, option bytes not programmed.

Figure 42. Typical I_{DD} in RUN vs. f_{CPU}

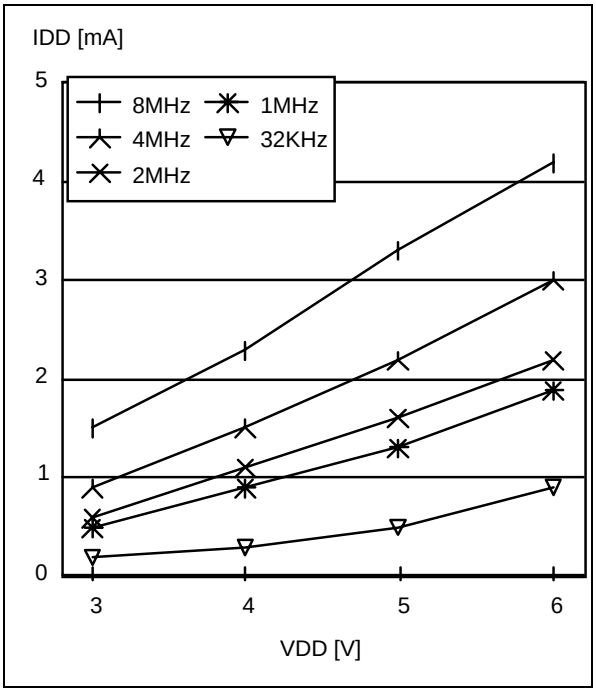
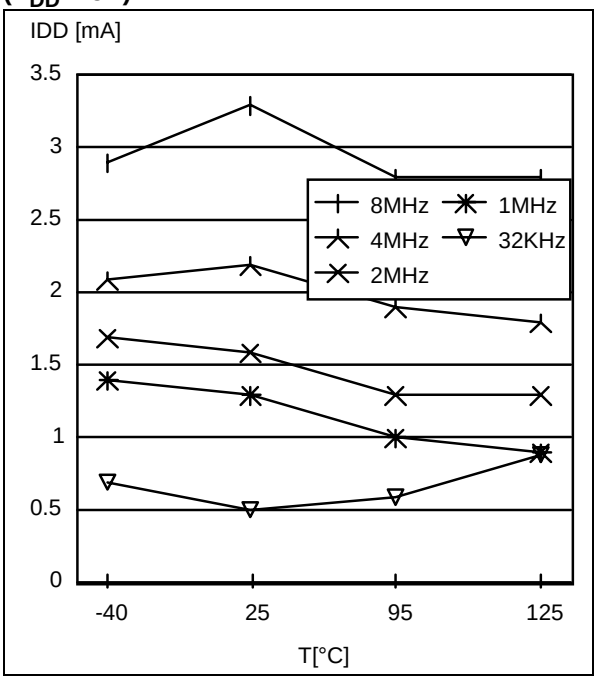


Figure 43. Typical I_{DD} in RUN vs. Temperature (V_{DD} = 5V)



SUPPLY CURRENT CHARACTERISTICS (Cont'd)

10.4.2 WAIT Modes

Symbol	Parameter	Conditions			Typ [1]	Max [2]	Unit
I_{DD}	Supply current in WAIT mode [3] Option bytes not programmed (see Figure 44)	$4.5V \le V_{DD} \le 6.0V$	OTP devices	f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	330 350 370 410 480	550 600 650 700 800	µA
	Supply current in WAIT mode [3] Option bytes programmed to 00H (see Figure 45)			f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	18 26 41 57 70	60 80 120 180 200	
	Supply current in WAIT mode[3] (see Figure 46)		ROM devices	f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	190 210 240 280 350	300 350 400 500 600	
	Supply current in WAIT mode [3] Option bytes not programmed (see Figure 44)	$3V \le V_{DD} \le 3.6V$	OTP devices	f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	80 90 100 120 150	120 140 150 200 250	
	Supply current in WAIT mode [3] Option bytes programmed to 00H (see Figure 45)			f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	5 8 16 18 20	30 40 50 60 100	
	Supply current in WAIT mode [3] Option bytes not programmed (see Figure 46)		ROM devices	f_{OSC}=32kHz f_{OSC}=1MHz f_{OSC}=2MHz f_{OSC}=4MHz f_{OSC}=8MHz	60 65 80 100 130	100 110 120 150 210	

Notes:

1. Typical data are based on T_A=25°C, V_{DD}=5V ($4.5V \le V_{DD} \le 6.0V$ range) and V_{DD}=3.3V ($3V \le V_{DD} \le 3.6V$ range).
2. Data based on characterization results, tested in production at V_{DD} max. and f_{OSC} max.
3. All I/O pins in input with pull-up mode (no load), all peripherals in reset state; clock input (OSC_{IN}) driven by external square wave, OSG and LVD disabled.

SUPPLY CURRENT CHARACTERISTICS (Cont'd)

Figure 44. Typical I_{DD} in WAIT vs f_{CPU} and Temperature for OTP devices with option bytes not programmed

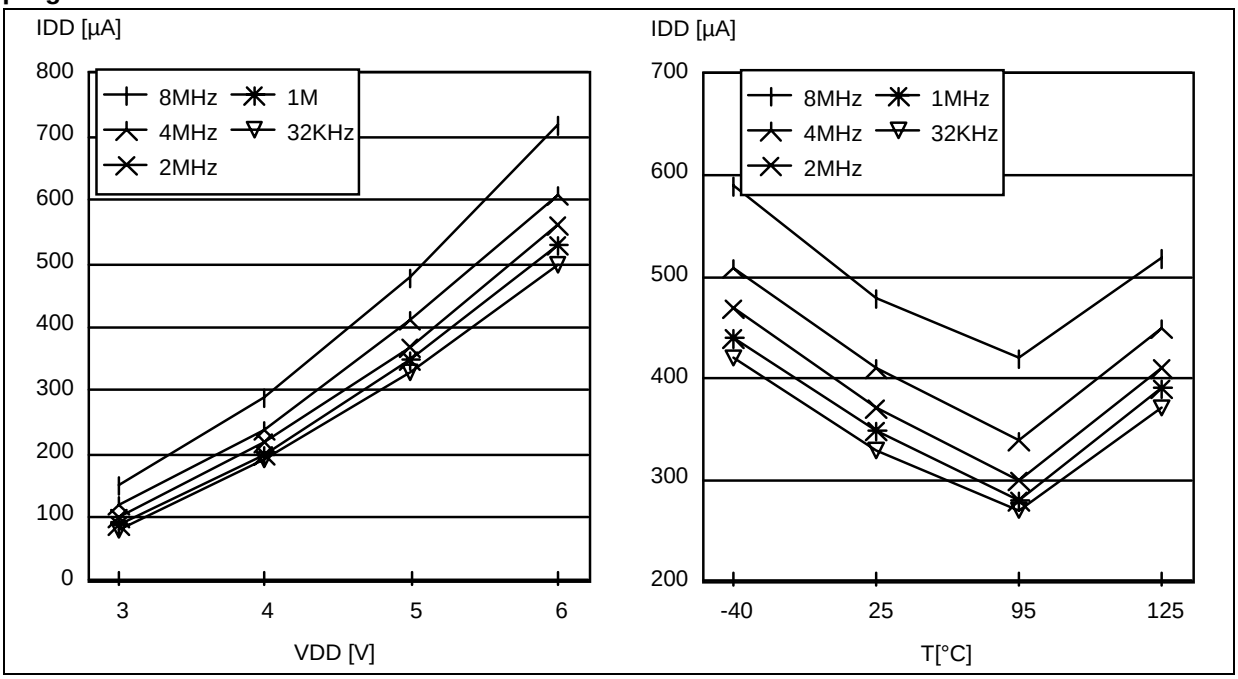
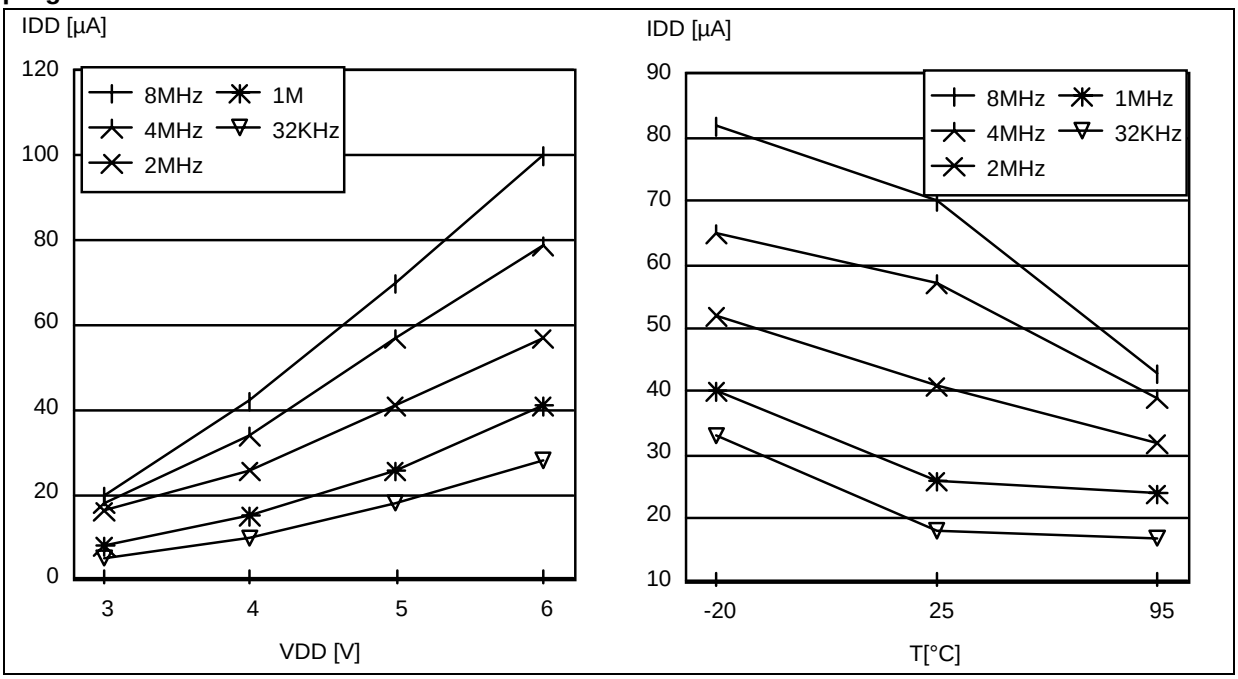
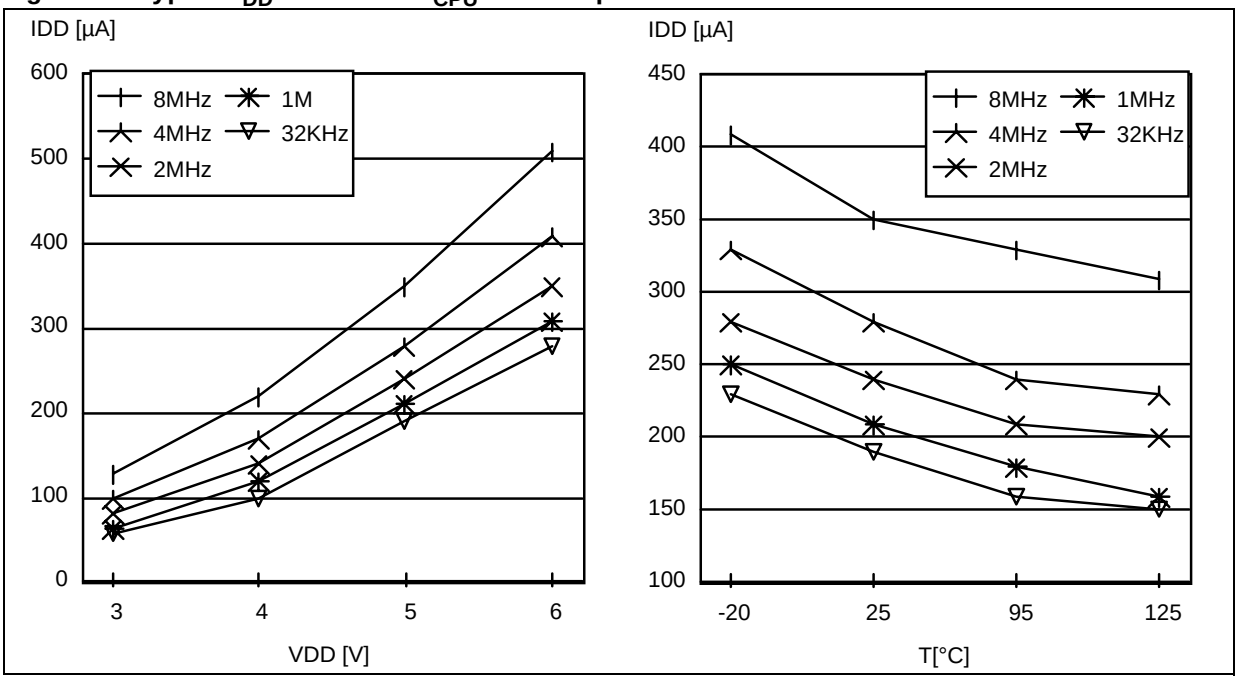


Figure 45. Typical I_{DD} in WAIT vs f_{CPU} and Temperature for OTP devices with option bytes programmed to 00H



SUPPLY CURRENT CHARACTERISTICS (Cont'd)

Figure 46. Typical I_{DD} in WAIT vs f_{CPU} and Temperature for ROM devices



SUPPLY CURRENT CHARACTERISTICS (Cont'd)

10.4.3 STOP Mode

Symbol	Parameter	Conditions	Typ [1]	Max	Unit
I_{DD}	Supply current in STOP mode [2] (see Figure 47 & Figure 48)	OTP devices	0.3	10 [3] 20 [4]	µA
		ROM devices	0.1	2 [3] 20 [4]	

Notes:

1. Typical data are based on V_{DD}=5.0V at T_A=25°C.
2. All I/O pins in input with pull-up mode (no load), all peripherals in reset state, OSG and LVD disabled, option bytes programmed to 00H. Data based on characterization results, tested in production at V_{DD} max. and f_{CPU} max.
3. Maximum STOP consumption for -40°C<Ta<90°C
4. Maximum STOP consumption for -40°C<Ta<125°C

Figure 47. Typical I_{DD} in STOP vs Temperature for OTP devices

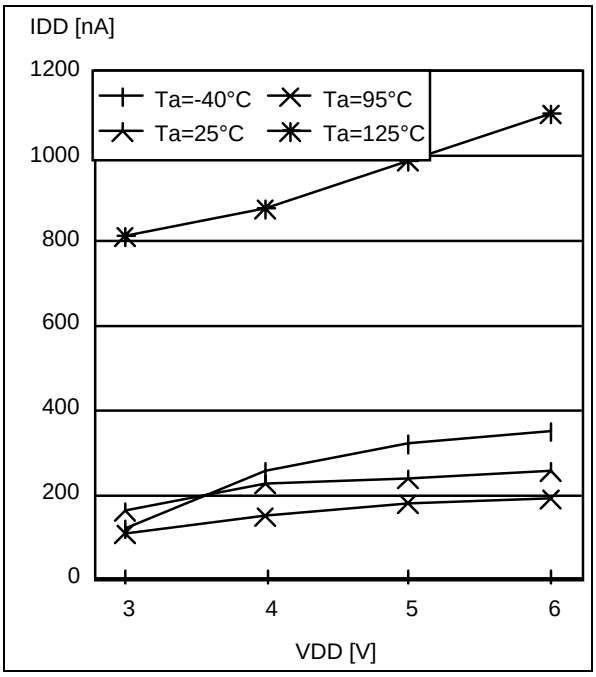
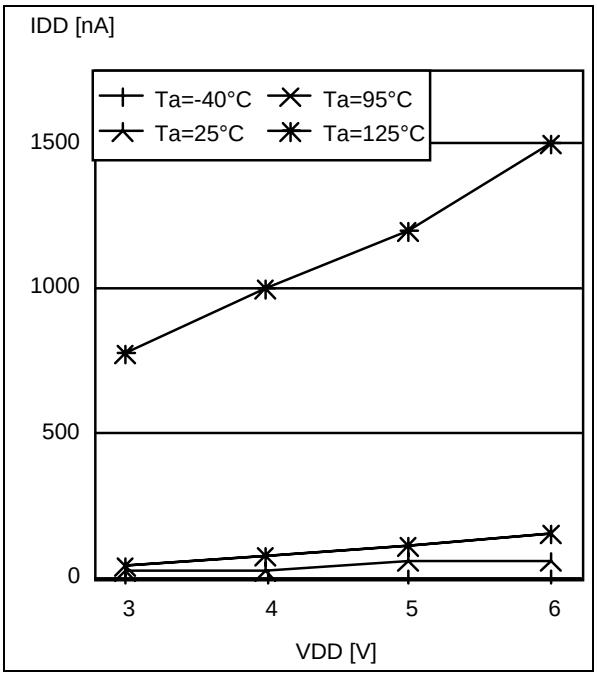


Figure 48. Typical I_{DD} in STOP vs Temperature for ROM devices



SUPPLY CURRENT CHARACTERISTICS (Cont'd)

10.4.4 Supply and Clock System

The previous current consumption specified for the ST6 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 STOP mode).

Symbol	Parameter	Conditions		Typ [1]	Max [2]	Unit
$I_{DD(CK)}$	Supply current of RC oscillator	f_{OSC}=32 kHz, f_{OSC}=1 MHz f_{OSC}=2 MHz f_{OSC}=4 MHz f_{OSC}=8 MHz	V_{DD}=5.0 V	 230 260 340 480		µA
		f_{OSC}=32 kHz, f_{OSC}=1 MHz f_{OSC}=2 MHz f_{OSC}=4 MHz f_{OSC}=8 MHz	V_{DD}=3.3 V	 80 110 180 320		
	Supply current of resonator oscillator	f_{OSC}=32 kHz, f_{OSC}=1 MHz f_{OSC}=2 MHz f_{OSC}=4 MHz f_{OSC}=8MHz	V_{DD}=5.0 V	900 280 240 140 40		
		f_{OSC}=32 kHz, f_{OSC}=1 MHz f_{OSC}=2 MHz f_{OSC}=4 MHz f_{OSC}=8 MHz	V_{DD}=3.3 V	120 70 50 20 10		
$I_{DD(LFAO)}$	LFAO supply current [3]	V_{DD}=5.0 V		102		
$I_{DD(OSG)}$	OSG supply current [4]	V_{DD}=5.0 V		40		
$I_{DD(LVD)}$	LVD supply current [5]	V_{DD}=5.0 V		170		

10.4.5 On-Chip Peripherals

Symbol	Parameter	Conditions		Typ [1]	Unit
$I_{DD(TIM)}$	8-bit Timer supply current [6]	f_{OSC}=8 MHz	V_{DD}=5.0 V	170	µA
			V_{DD}=3.3 V	100	
$I_{DD(ADC)}$	ADC supply current when converting [7]	f_{OSC}=8 MHz	V_{DD}=5.0 V	80	
			V_{DD}=3.3 V	50	

Notes:

1. Typical data are based on T_A=25°C.
2. Data based on characterization results, not tested in production.
3. Data based on a differential I_{DD} measurement between reset configuration (OSG and LFAO disabled) and LFAO running (also includes the OSG stand alone consumption).
4. Data based on a differential I_{DD} measurement between reset configuration with OSG disabled and OSG enabled.
5. Data based on a differential I_{DD} measurement between reset configuration with LVD disabled and LVD enabled.
6. Data based on a differential I_{DD} measurement between reset configuration (timer disabled) and timer running.
7. Data based on a differential I_{DD} measurement between reset configuration and continuous A/D conversions.

10.5 CLOCK AND TIMING CHARACTERISTICS

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

10.5.1 General Timings

Symbol	Parameter	Conditions	Min	Typ [1]	Max	Unit
$t_{c(INST)}$	Instruction cycle time		2	4	5	t_{CPU}
		f_{CPU}=8 MHz	3.25	6.5	8.125	µs
$t_{v(IT)}$	Interrupt reaction time [2] $t_{v(IT)} = \Delta t_{c(INST)} + 6$		6		11	t_{CPU}
		f_{CPU}=8 MHz	9.75		17.875	µs

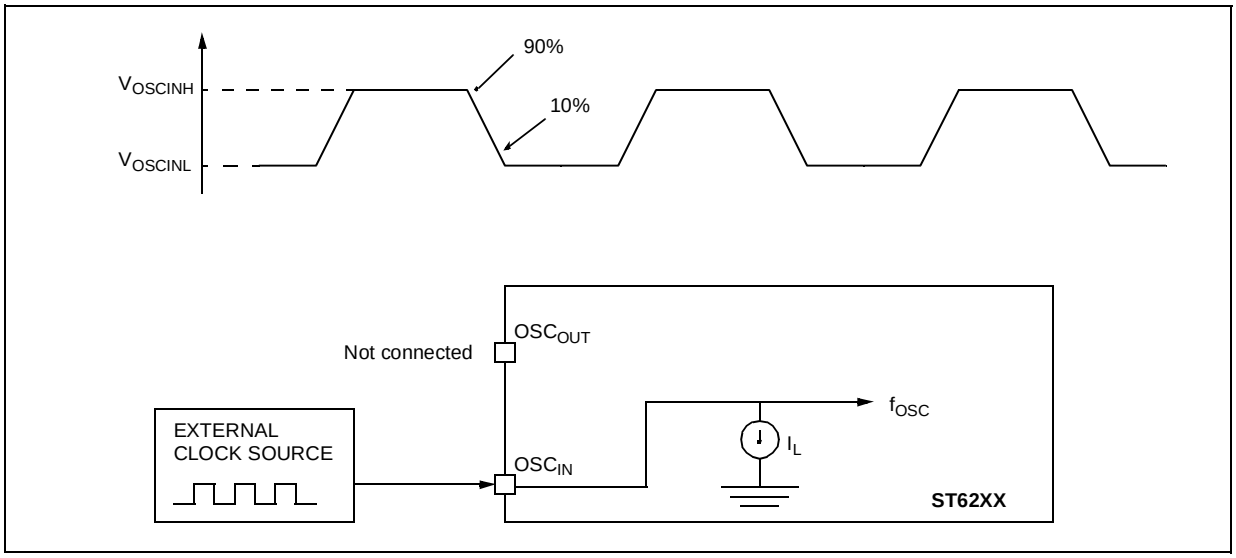
10.5.2 External Clock Source

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{OSCINH}	OSC_{IN} input pin high level voltage	See Figure 49	$0.7xV_{DD}$		V_{DD}	V
V_{OSCINL}	OSC_{IN} input pin low level voltage		V_{SS}		$0.3xV_{DD}$	
I_L	OSCx Input leakage current	$V_{SS} \leq V_{IN} \leq V_{DD}$			± 2	µA

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.

Figure 49. Typical Application with an External Clock Source



CLOCK AND TIMING CHARACTERISTICS (Cont'd)

10.5.3 Crystal and Ceramic Resonator Oscillators

The ST6 internal clock can be supplied with several different Crystal/Ceramic resonator oscillators. Only parallel resonant crystals can be used. All the information given in this paragraph are based on characterization results with specified typical external components. Refer to the crystal/ceramic resonator manufacturer for more details (frequency, package, accuracy...).

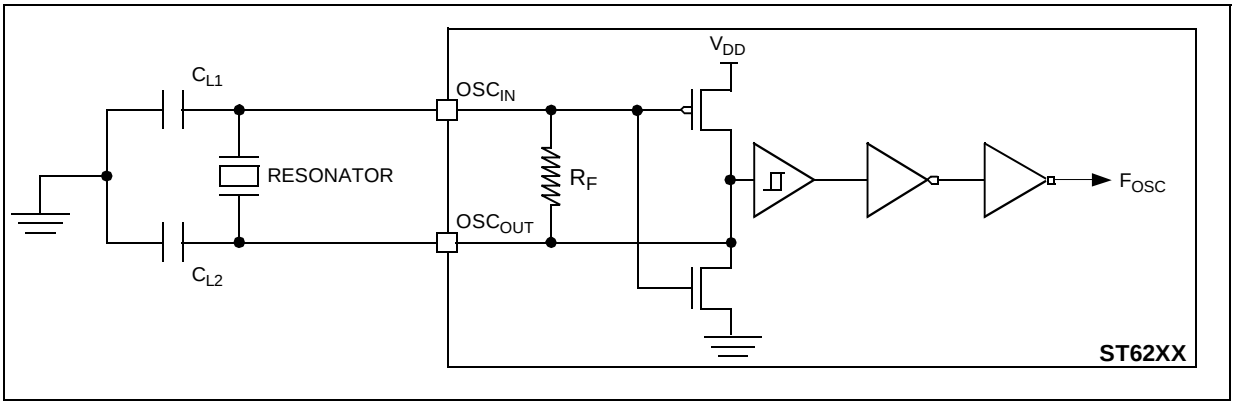
Symbol	Parameter	Conditions	Typ	Unit
R_F	Feedback resistor		3	MΩ
C_{L1} C_{L2}	Recommended load capacitances versus equivalent crystal or ceramic resonator frequency	f_{OSC}=32 kHz, f_{OSC}=1 MHz f_{OSC}=2 MHz f_{OSC}=4 MHz f_{OSC}=8 MHz	120 47 33 33 22	pF

Oscillator	Typical Crystal or Ceramic Resonators				C_{L1} [pF]	C_{L2} [pF]	$t_{SU(osc)}$ [ms] 1)
		Reference	Freq.	Characteristic 1)			
Ceramic	MURATA	CSB455E	455KHz	Δf_{OSC}=[±0.5KHz$_{tolerance}$,±0.3%$_{\Delta Ta}$,±0.5%$_{aging}$]	220	220	
		CSB1000J	1MHz	Δf_{OSC}=[±0.5KHz$_{tolerance}$,±0.3%$_{\Delta Ta}$,±0.5%$_{aging}$]	100	100	
		CSTCC2.00MG0H6	2MHz	Δf_{OSC}=[±0.5%$_{tolerance}$,±0.5%$_{\Delta Ta}$,±0.3%$_{aging}$]	47	47	
		CSTCC4.00MG0H6	4MHz	Δf_{OSC}=[±0.5%$_{tolerance}$,±0.3%$_{\Delta Ta}$,±0.3%$_{aging}$]	47	47	
		CSTCC8.00MG	8MHz	Δf_{OSC}=[±0.5%$_{tolerance}$,±0.3%$_{\Delta Ta}$,±0.3%$_{aging}$]	15	15	

Notes:

1. Resonator characteristics given by the crystal/ceramic resonator manufacturer.
2. $t_{SU(OSC)}$ is the typical oscillator start-up time measured between V_{DD}=2.8V and the fetch of the first instruction (with a quick V_{DD} ramp-up from 0 to 5V (<50µs).
3. The oscillator selection can be optimized in terms of supply current using an high quality resonator with small R_S value. Refer to crystal/ceramic resonator manufacturer for more details.

Figure 50. Typical Application with a Crystal or Ceramic Resonator



CLOCK AND TIMING CHARACTERISTICS (Cont'd)

10.5.4 RC Oscillator

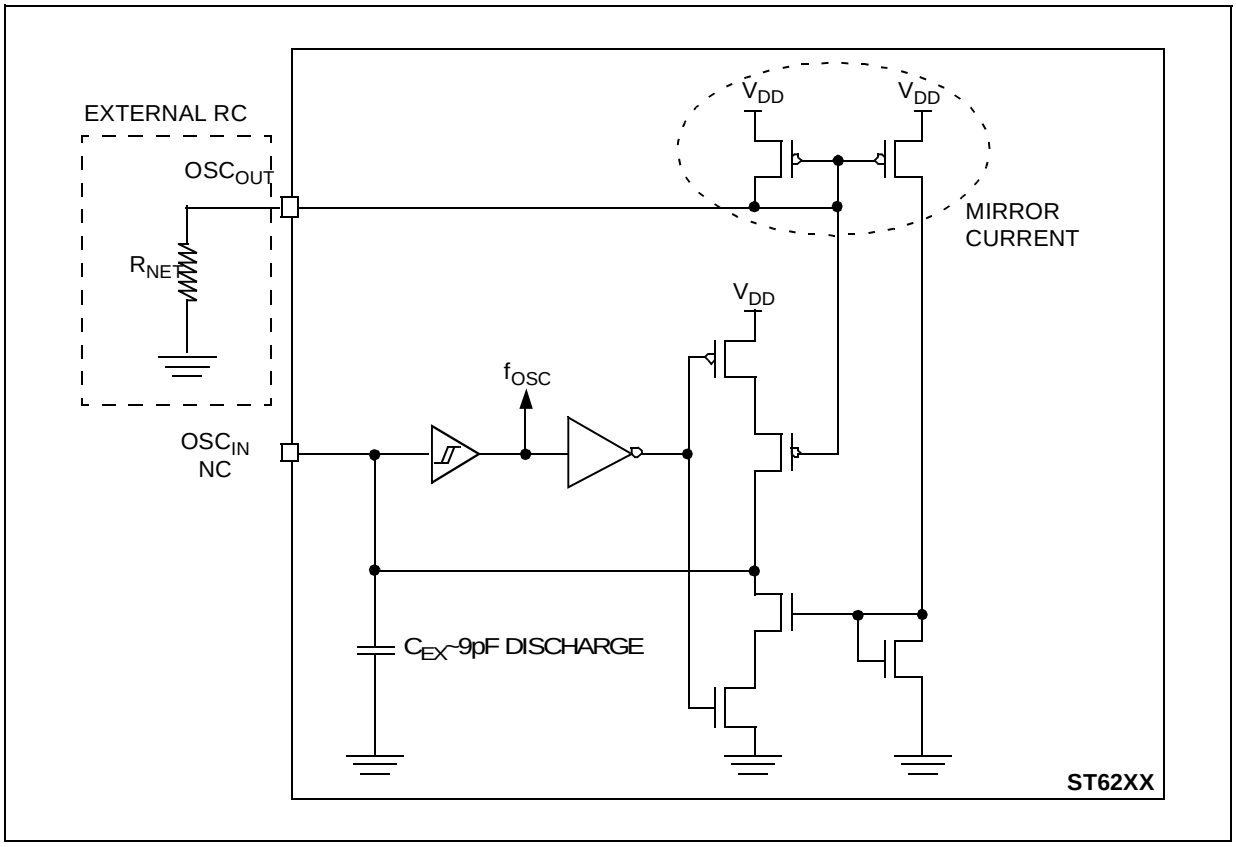
The ST6 internal clock can be supplied with an external RC oscillator. Depending on the R_{NET} value, the accuracy of the frequency is about 20%, so it may not be suitable for some applications.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
f_{OSC}	RC oscillator frequency [1]	$4.5V \le V_{DD} \le 6.0V$	R_{NET}=22 kΩ	7.2	8.6	10	MHz
			R_{NET}=47 kΩ	5.1	5.7	6.5	
			R_{NET}=100 kΩ	3.2	3.4	3.8	
			R_{NET}=220 kΩ	1.8	1.9	2	
			R_{NET}=470 kΩ	0.9	0.95	1.1	
		$3V \le V_{DD} \le 3.6V$	R_{NET}=22 kΩ	3.7	4.3	4.9	
			R_{NET}=47 kΩ	2.8	3	3.3	
			R_{NET}=100 kΩ	1.8	1.9	2	
			R_{NET}=220 kΩ	1	1.1	1.2	
			R_{NET}=470 kΩ	0.5	0.55	0.6	
R_{NET}	RC Oscillator external resistor [2]		see Figure 52 & Figure 53	22		870	kΩ

Notes:

1. Data based on characterization results, not tested in production. These measurements were done with the OSCin pin unconnected (only soldered on the PCB).
2. R_{NET} must have a positive temperature coefficient (ppm/°C), carbon resistors should therefore not be used.

Figure 51. Typical Application with RC Oscillator



CLOCK AND TIMING CHARACTERISTICS (Cont'd)

Figure 52. Typical RC Oscillator frequency vs. V_{DD}

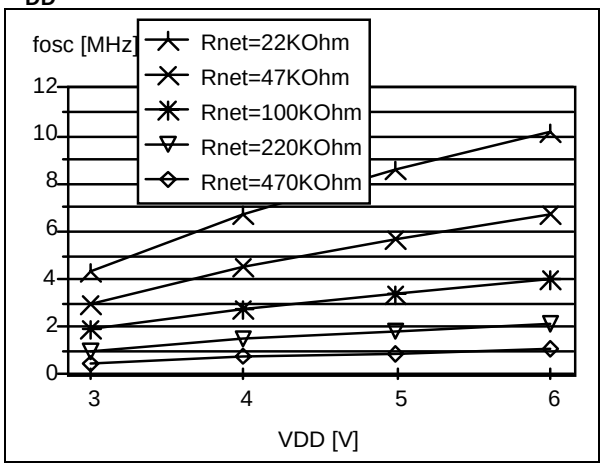
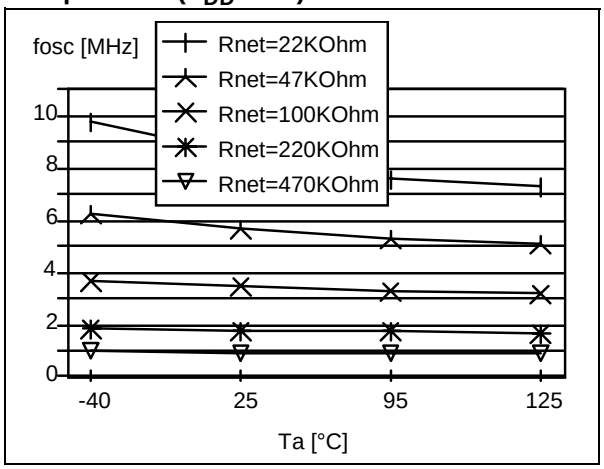


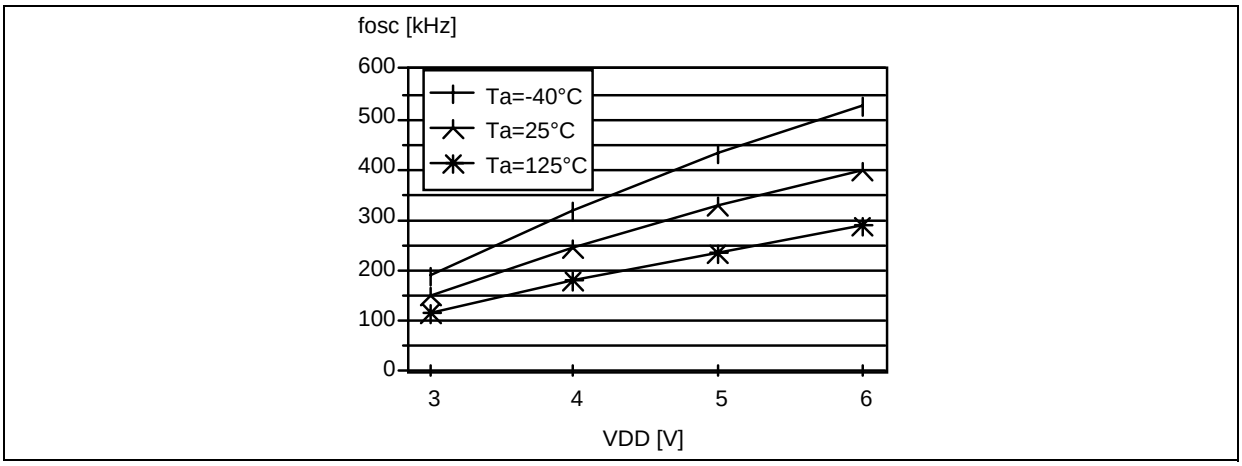
Figure 53. Typical RC Oscillator frequency vs. Temperature (V_{DD} = 5V)



10.5.5 Oscillator Safeguard (OSG) and Low Frequency Auxiliary Oscillator (LFAO)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{LFAO}	Low Frequency Auxiliary Oscillator Frequency [1)]	T_A=25° C, V_{DD}=5.0 V	200	350	800	kHz
		T_A=25° C, V_{DD}=3.3 V	86	150	340	
f_{OSG}	Internal Frequency with OSG enabled	T_A=25° C, V_{DD}=4.5 V	4			MHz
		T_A=25° C, V_{DD}=3.3 V	2			

Figure 54. Typical LFAO Frequencies



Note:
1. Data based on characterization results.

10.6 MEMORY CHARACTERISTICS

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

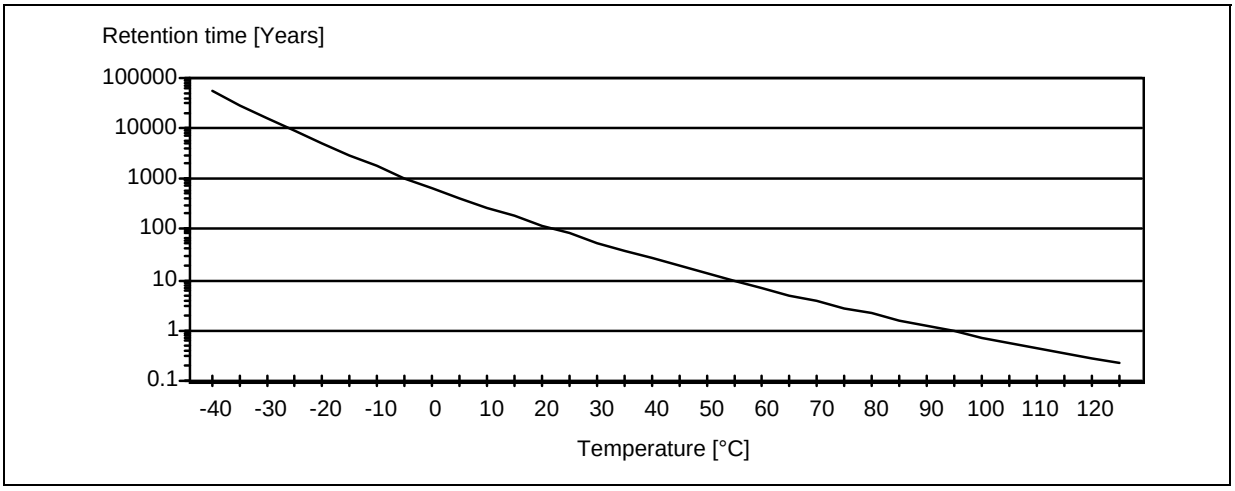
10.6.1 RAM and Hardware Registers

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{RM}	Data retention[1)]		0.7			V

10.6.2 EPROM Program Memory

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
t_{ret}	Data retention [2)]	T_A=+55°C [3)]	10			years

Figure 55. EPROM Retention Time vs. Temperature



Notes:

1. Minimum V_{DD} supply voltage without losing data stored in RAM (in STOP mode or under RESET) or in hardware registers (only in STOP mode). Guaranteed by construction, not tested in production.
2. Data based on reliability test results and monitored in production.
3. The data retention time increases when the T_A decreases, see Figure 55.

10.7 EMC CHARACTERISTICS

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

10.7.1 Functional EMS

(Electro Magnetic Susceptibility)

Based on a simple running application on the product (toggling 2 LEDs through I/O ports), the product is stressed by two electro magnetic events until a failure occurs (indicated by the LEDs).

- **ESD**: Electro-Static 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.

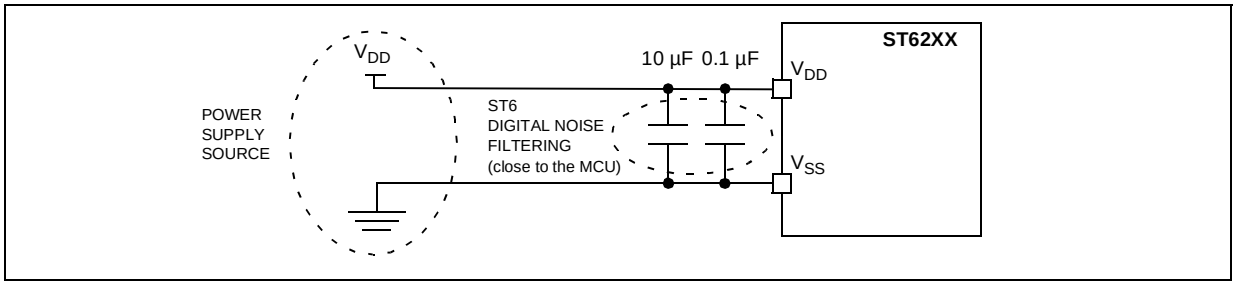
A device reset allows normal operations to be resumed.

Symbol	Parameter	Conditions	Neg [1]	Pos [1]	Unit
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	-2	2	kV
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	-2.5	3	

Notes:

1. Data based on characterization results, not tested in production.
2. The suggested 10 µF and 0.1 µF decoupling capacitors on the power supply lines are proposed as a good price vs. EMC performance tradeoff. They have to be put as close as possible to the device power supply pins. Other EMC recommendations are given in other sections (I/Os, RESET, OSCx pin characteristics).

Figure 56. EMC Recommended Star Network Power Supply Connection [2]



EMC CHARACTERISTICS (Cont'd)

10.7.2 Absolute Electrical Sensitivity

Based on three different tests (ESD, LU and DLU) using specific measurement methods, the product is stressed in order to determine its performance in terms of electrical sensitivity. For more details, refer to the AN1181 application note.

10.7.2.1 Electro-Static Discharge (ESD)

Electro-Static Discharges (3 positive then 3 negative pulses separated by 1 second) are applied to the pins of each sample according to each pin combination. The sample size depends of the number of supply pins of the device (3 parts*(n+1) supply pin). Two models are usually simulated: Human Body Model and Machine Model. This test conforms to the JESD22-A114A/A115A standard. See Figure 57 and the following test sequences.

Human Body Model Test Sequence

– C_L is loaded through S1 by the HV pulse generator.

– S1 switches position from generator to R.

– A discharge from C_L through R (body resistance) to the ST6 occurs.

– S2 must be closed 10 to 100ms after the pulse delivery period to ensure the ST6 is not left in charge state. S2 must be opened at least 10ms prior to the delivery of the next pulse.

Machine Model Test Sequence

– C_L is loaded through S1 by the HV pulse generator.

– S1 switches position from generator to ST6.

– A discharge from C_L to the ST6 occurs.

– S2 must be closed 10 to 100ms after the pulse delivery period to ensure the ST6 is not left in charge state. S2 must be opened at least 10ms prior to the delivery of the next pulse.

– R (machine resistance), in series with S2, ensures a slow discharge of the ST6.

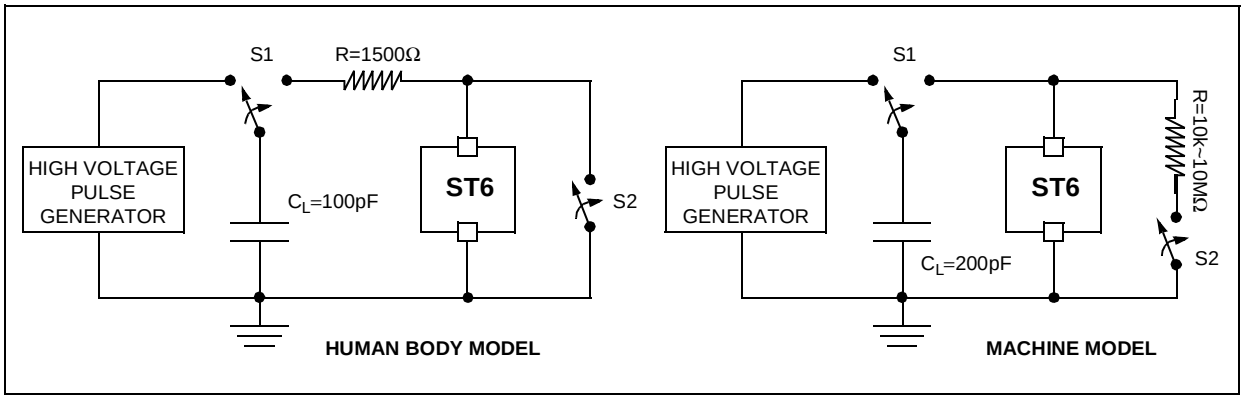
Absolute Maximum Ratings

Symbol	Ratings	Conditions	Maximum value [1]	Unit
$V_{ESD(HBM)}$	Electro-static discharge voltage (Human Body Model)	T_A=+25°C	2000	V
$V_{ESD(MM)}$	Electro-static discharge voltage (Machine Model)	T_A=+25°C	200	

Notes:

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

Figure 57. Typical Equivalent ESD Circuits



EMC CHARACTERISTICS (Cont'd)

10.7.2.2 Static and Dynamic Latch-Up

- **LU**: 3 complementary static tests are required on 10 parts to assess the latch-up performance. A supply overvoltage (applied to each power supply pin), a current injection (applied to each input, output and configurable I/O pin) and a power supply switch sequence are performed on each sample. This test conforms to the EIA/JESD 78 IC latch-up standard. For more details, refer to the AN1181 application note.
- **DLU**: Electro-Static Discharges (one positive then one negative test) are applied to each pin of 3 samples when the micro is running to assess the latch-up performance in dynamic mode. Power supplies are set to the typical values, the oscillator is connected as near as possible to the pins of the micro and the component is put in reset mode. This test conforms to the IEC1000-4-2 and SAEJ1752/3 standards and is described in Figure 58. For more details, refer to the AN1181 application note.

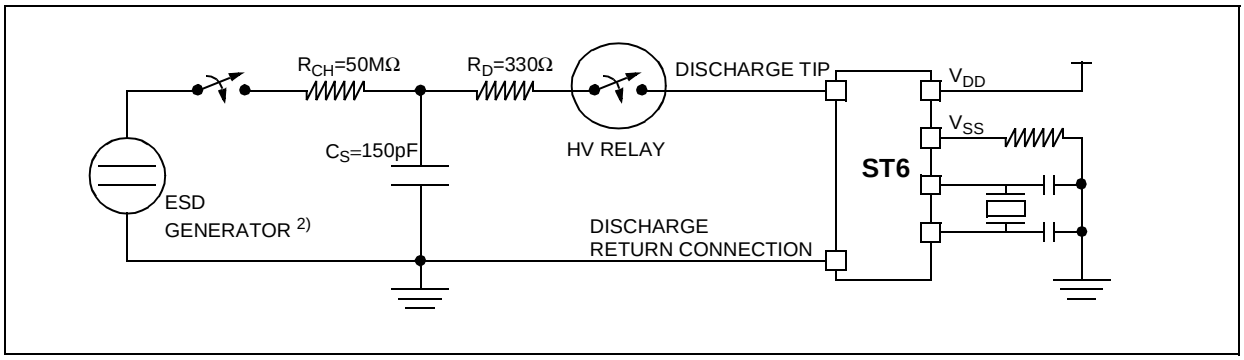
Electrical Sensitivities

Symbol	Parameter	Conditions	Class [1]
LU	Static latch-up class	T_A=+25°C T_A=+85°C	A A
DLU	Dynamic latch-up class	V_{DD}=5V, f_{OSC}=4MHz, T_A=+25°C	A

Notes:

1. Class description: A Class is an STMicroelectronics internal specification. All its limits are higher than the JEDEC specifications, that means when a device belongs to Class A it exceeds the JEDEC standard. B Class strictly covers all the JEDEC criteria (international standard).
2. Schaffner NSG435 with a pointed test finger.

Figure 58. Simplified Diagram of the ESD Generator for DLU



EMC CHARACTERISTICS (Cont'd)

10.7.3 ESD Pin Protection Strategy

To protect an integrated circuit against Electro-Static Discharge the stress must be controlled to prevent degradation or destruction of the circuit elements. The stress generally affects the circuit elements which are connected to the pads but can also affect the internal devices when the supply pads receive the stress. The elements to be protected must not receive excessive current, voltage or heating within their structure.

An ESD network combines the different input and output ESD protections. This network works, by allowing safe discharge paths for the pins subjected to ESD stress. Two critical ESD stress cases are presented in Figure 59 and Figure 60 for standard pins.

Standard Pin Protection

To protect the output structure the following elements are added:

– A diode to V_{DD} (3a) and a diode from V_{SS} (3b)

– A protection device between V_{DD} and V_{SS} (4)

To protect the input structure the following elements are added:

– A resistor in series with the pad (1)

– A diode to V_{DD} (2a) and a diode from V_{SS} (2b)

– A protection device between V_{DD} and V_{SS} (4)

Figure 59. Positive Stress on a Standard Pad vs. V_{SS}

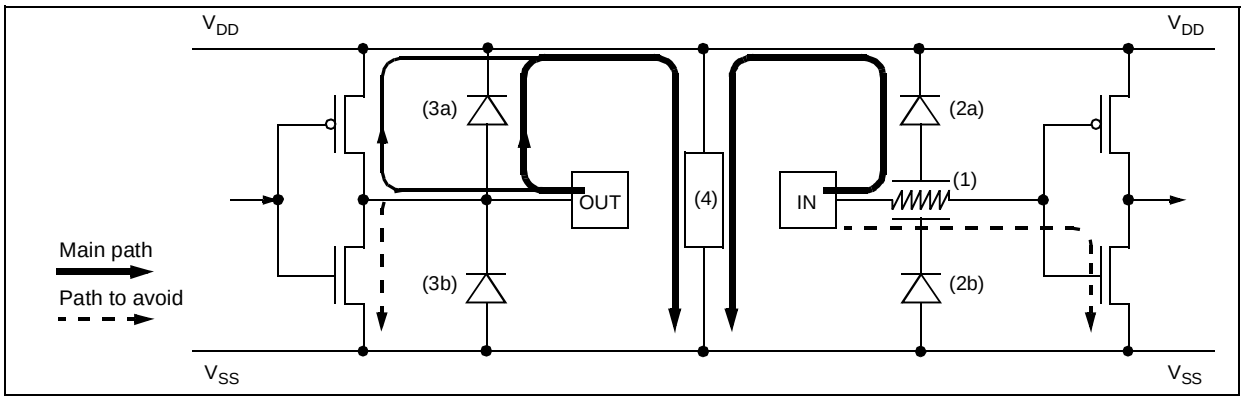
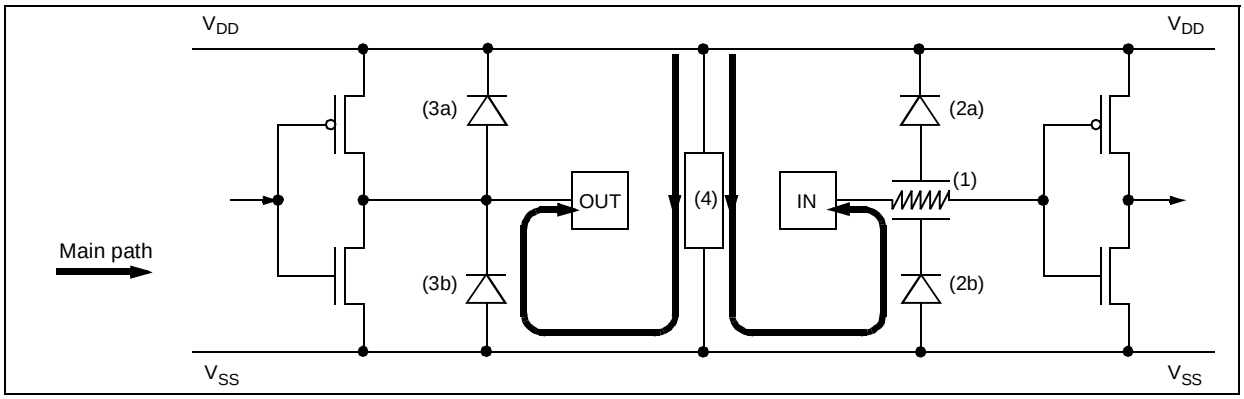


Figure 60. Negative Stress on a Standard Pad vs. V_{DD}



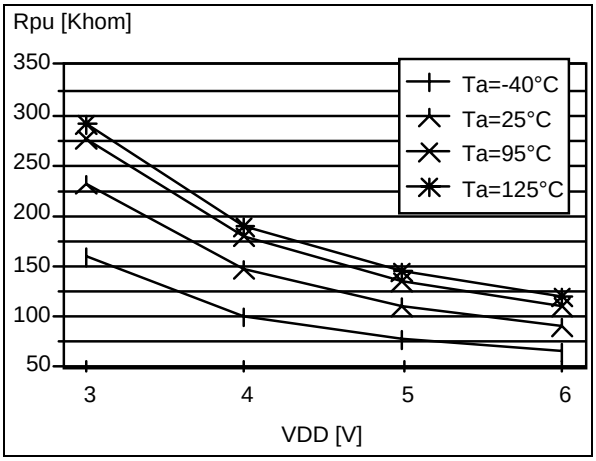
10.8 I/O PORT PIN CHARACTERISTICS

10.8.1 General Characteristics

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

Symbol	Parameter	Conditions		Min	Typ [1]	Max	Unit
V_{IL}	Input low level voltage [2]					$0.3xV_{DD}$	V
V_{IH}	Input high level voltage [2]			$0.7xV_{DD}$			
V_{hys}	Schmitt trigger voltage hysteresis [3]	V_{DD}=5V		200	400		mV
		V_{DD}=3.3V		200	400		
I_L	Input leakage current	$V_{SS} \leq V_{IN} \leq V_{DD}$ (no pull-up configured)			0.1	1	µA
R_{PU}	Weak pull-up equivalent resistor [4]	$V_{IN}=V_{SS}$	V_{DD}=5V	40	110	350	kΩ
			V_{DD}=3.3V	80	230	700	
C_{IN}	I/O input pin capacitance				5	10	pF
C_{OUT}	I/O output pin capacitance				5	10	pF
$t_{f(IO)out}$	Output high to low level fall time [5]	C_L=50pF Between 10% and 90%			30		ns
$t_{r(IO)out}$	Output low to high level rise time [5]				35		
$t_{w(IT)in}$	External interrupt pulse time [6]			1			t_{CPU}

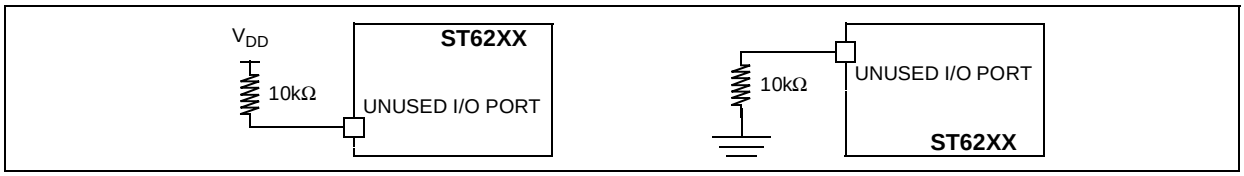
Figure 61. Typical R_{PU} vs. V_{DD} with V_{IN} = V_{SS}



Notes:

1. Unless otherwise specified, typical data are based on T_A=25°C and V_{DD}=5V.
2. Data based on characterization results, not tested in production.
3. Hysteresis voltage between Schmitt trigger switching levels. Based on characterization results, not tested.
4. The R_{PU} pull-up equivalent resistor is based on a resistive transistor. This data is based on characterization results, not tested in production.
5. Data based on characterization results, not tested in production.
6. To generate an external interrupt, a minimum pulse width has to be applied on an I/O port pin configured as an external interrupt source.

Figure 62. Two typical Applications with unused I/O Pin



I/O PORT PIN CHARACTERISTICS (Cont'd)

10.8.2 Output Driving Current

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

Symbol	Parameter	Conditions		Min	Max	Unit
V_{OL} 1)	Output low level voltage for a standard I/O pin (see Figure 63 and Figure 66)	V_{DD}=5V	I_{IO}=+10µA, T_A≤125°C		0.1	V
			I_{IO}=+3mA, T_A≤125°C		0.8	
			I_{IO}=+5mA, T_A≤85°C		0.8	
			I_{IO}=+10mA, T_A≤85°C		1.2	
	Output low level voltage for a high sink I/O pin (see Figure 64 and Figure 67)		I_{IO}=+10µA, T_A≤125°C		0.1	
			I_{IO}=+7mA, T_A≤125°C		0.8	
			I_{IO}=+10mA, T_A≤85°C		0.8	
			I_{IO}=+15mA, T_A≤125°C		1.3	
			I_{IO}=+20mA, T_A≤85°C		1.3	
			I_{IO}=+30mA, T_A≤85°C		2	
V_{OH} 2)	Output high level voltage for an I/O pin (see Figure 65 and Figure 68)		I_{IO}=-10µA, T_A≤125°C	V_{DD}-0.1		
			I_{IO}=-3mA, T_A≤125°C	V_{DD}-1.5		
			I_{IO}=-5mA, T_A≤85°C	V_{DD}-1.5		

Notes:

1. The I_{IO} current sunk must always respect the absolute maximum rating specified in Section 10.2.2 and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VSS}.
2. The I_{IO} current source must always respect the absolute maximum rating specified in Section 10.2.2 and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VDD}. True open drain I/O pins does not have V_{OH}.

Figure 63. Typical V_{OL} at V_{DD} = 5V (standard)

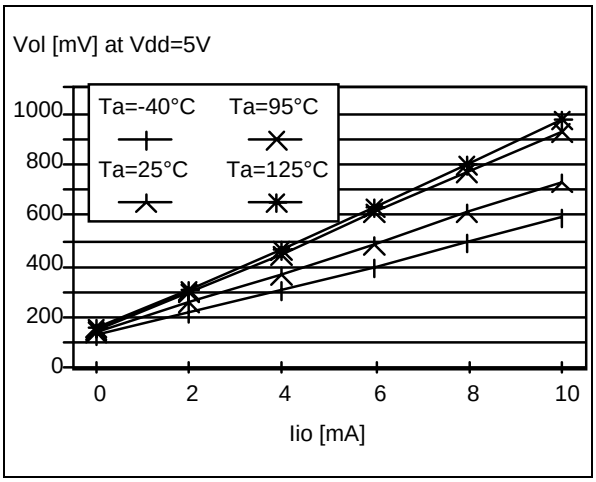
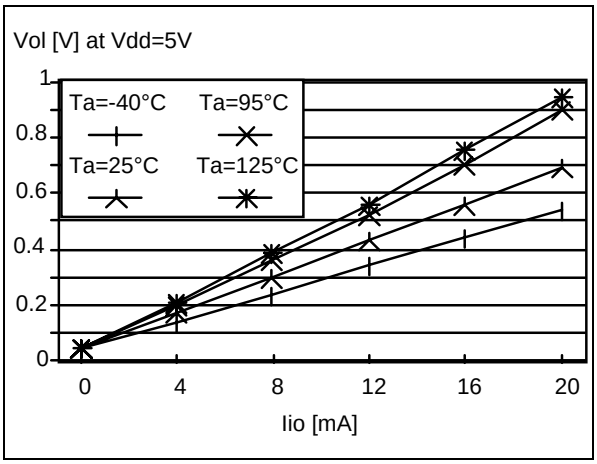


Figure 64. Typical V_{OL} at V_{DD} = 5V (high-sink)



I/O PORT PIN CHARACTERISTICS (Cont'd)

Figure 65. Typical V_{OH} at V_{DD} = 5V

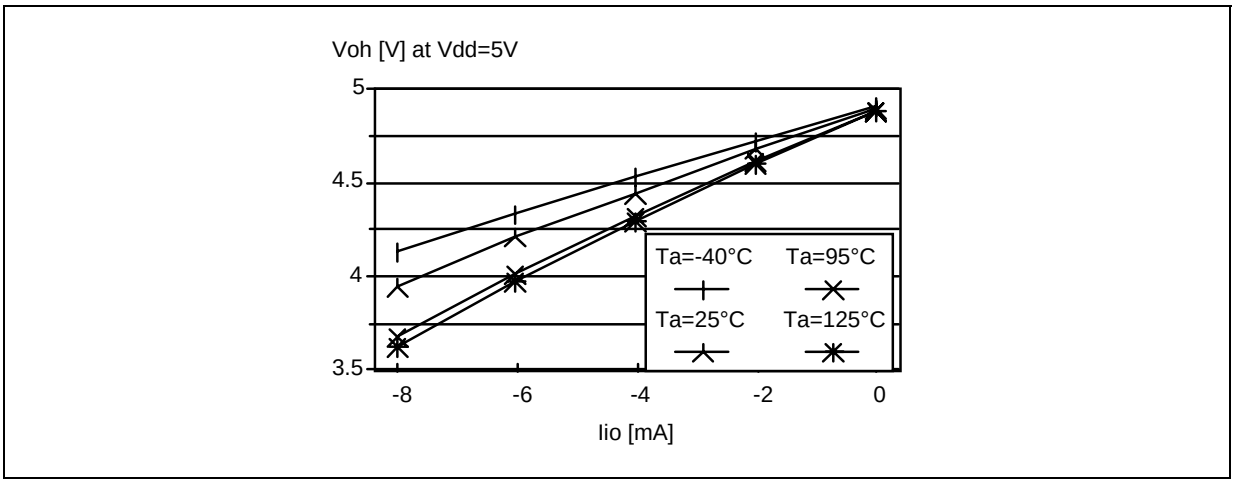


Figure 66. Typical V_{OL} vs V_{DD} (standard I/Os)

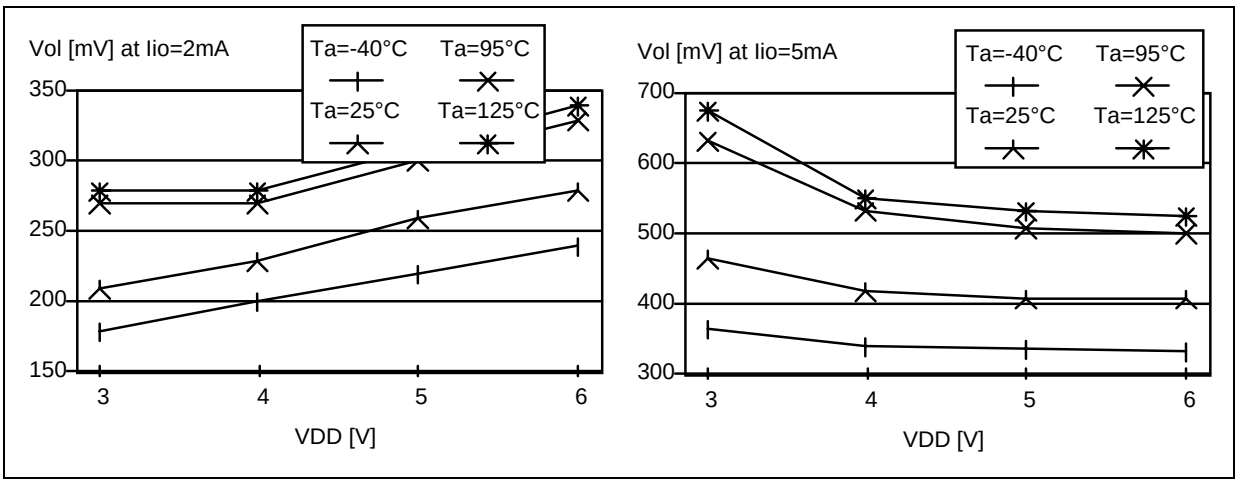
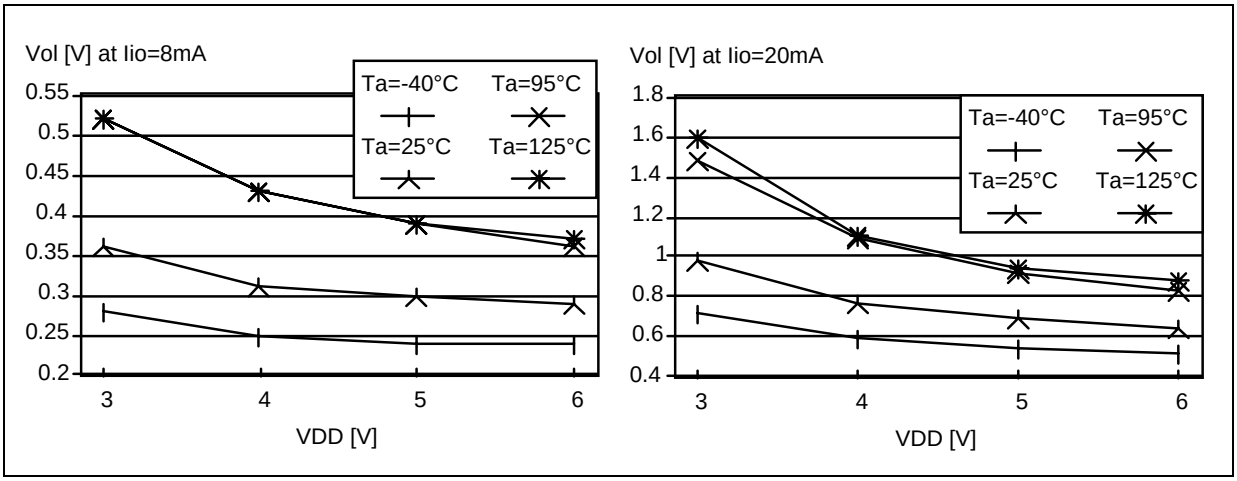
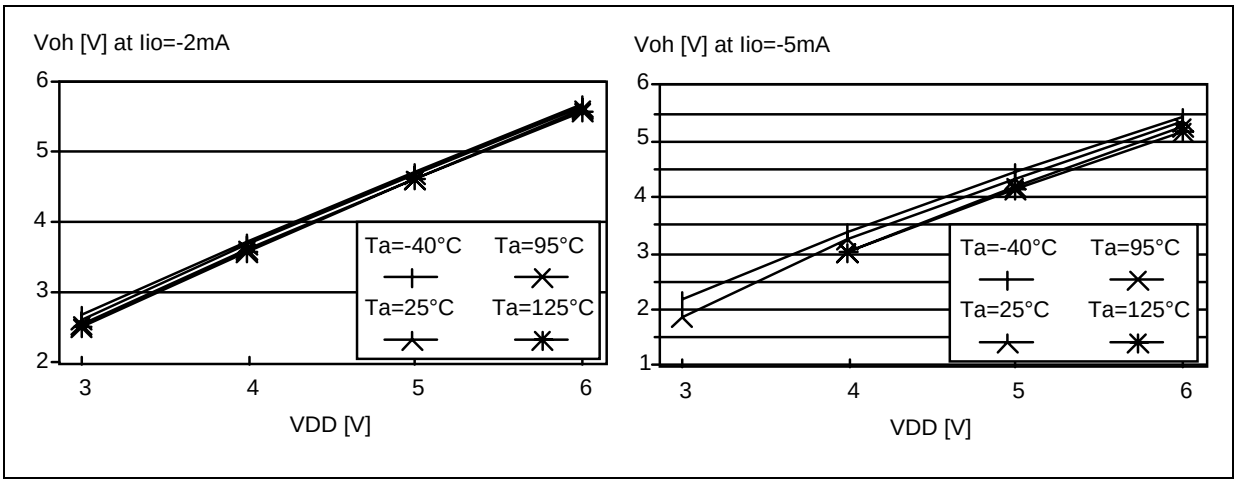


Figure 67. Typical V_{OL} vs V_{DD} (high-sink I/Os)



I/O PORT PIN CHARACTERISTICS (Cont'd)

Figure 68. Typical V_{OH} vs V_{DD}



10.9 CONTROL PIN CHARACTERISTICS

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

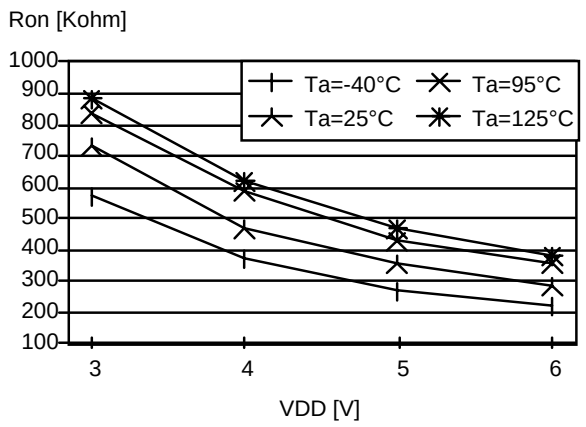
Subject to general operating conditions for V_{DD}, f_{OSC}, and T_A unless otherwise specified.

Symbol	Parameter	Conditions		Min	Typ [1]	Max	Unit
V_{IL}	Input low level voltage [2]					$0.3xV_{DD}$	V
V_{IH}	Input high level voltage [2]			$0.7xV_{DD}$			V
V_{hys}	Schmitt trigger voltage hysteresis [3]			200	400		mV
R_{ON}	Weak pull-up equivalent resistor [4]	$V_{IN}=V_{SS}$	V_{DD}=5V	150	350	900	kΩ
			V_{DD}=3.3V	300	730	1900	
R_{ESD}	ESD resistor protection	$V_{IN}=V_{SS}$	V_{DD}=5V		2.8		kΩ
			V_{DD}=3.3V				
$t_{w(RSTL)out}$	Generated reset pulse duration	External pin or internal reset sources					t_{CPU} µs
$t_{h(RSTL)in}$	External reset pulse hold time [5]						µs
$t_{g(RSTL)in}$	Filtered glitch duration [6]						ns

Notes:

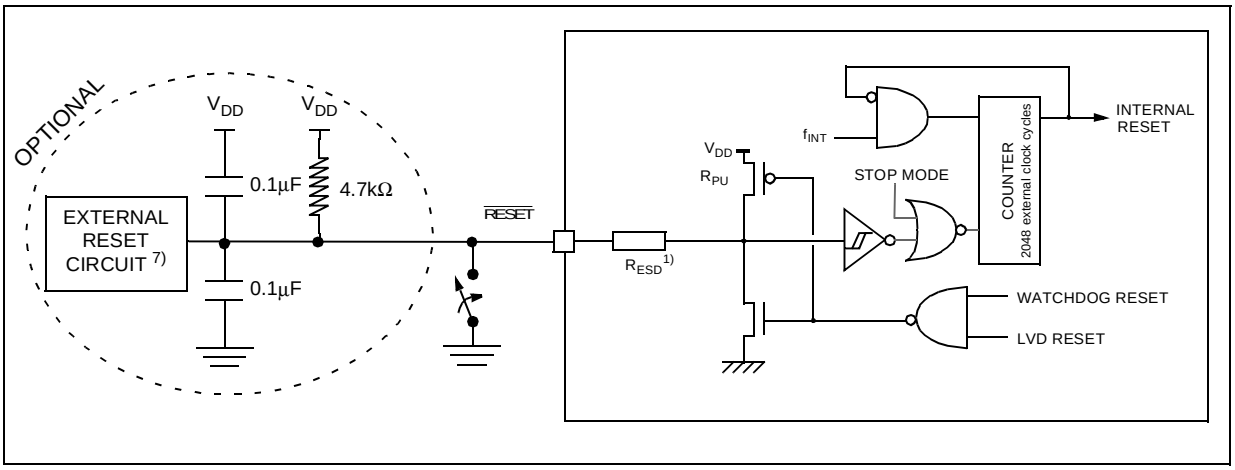
1. Unless otherwise specified, typical data are based on T_A=25°C and V_{DD}=5V.
2. Data based on characterization results, not tested in production.
3. Hysteresis voltage between Schmitt trigger switching levels. Based on characterization results, not tested.
4. The R_{ON} pull-up equivalent resistor is based on a resistive transistor. This data is based on characterization results, not tested in production.
5. All short pulse applied on $\overline{\text{RESET}}$ pin with a duration below $t_{h(RSTL)in}$ can be ignored.
6. The reset network protects the device against parasitic resets, especially in a noisy environment.
7. The output of the external reset circuit must have an open-drain output to drive the ST6 reset pad. Otherwise the device can be damaged when the ST6 generates an internal reset (LVD or watchdog).

Figure 69. Typical R_{ON} vs V_{DD} with $V_{IN}=V_{SS}$



CONTROL PIN CHARACTERISTICS (Cont'd)

Figure 70. Typical Application with $\overline{\text{RESET}}$ pin [8)]



10.9.2 NMI Pin

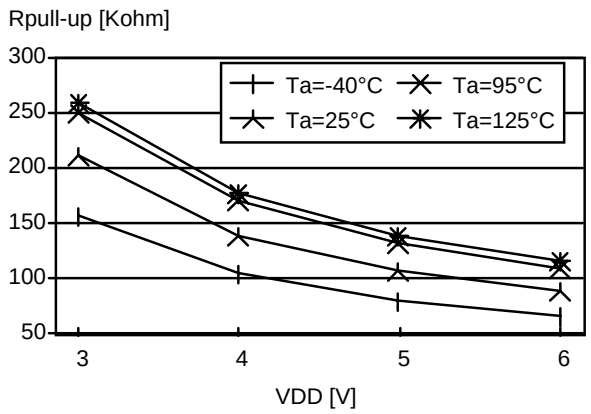
Subject to general operating conditions for V_{DD}, f_{OSC}, and T_A unless otherwise specified.

Symbol	Parameter	Conditions		Min	Typ [1)]	Max	Unit
V_{IL}	Input low level voltage [2)]					$0.3xV_{DD}$	V
V_{IH}	Input high level voltage [2)]			$0.7xV_{DD}$			
V_{hys}	Schmitt trigger voltage hysteresis [3)]			200	400		mV
$R_{pull-up}$	Weak pull-up equivalent resistor [4)]	$V_{IN}=V_{SS}$	V_{DD}=5V	40	100	350	kΩ
			V_{DD}=3.3V	80	200	700	

Notes:

1. Unless otherwise specified, typical data are based on T_A=25°C and V_{DD}=5V.
2. Data based on characterization results, not tested in production.
3. Hysteresis voltage between Schmitt trigger switching levels. Based on characterization results, not tested.
4. The $R_{pull-up}$ equivalent resistor is based on a resistive transistor. This data is based on characterization results, not tested in production.

Figure 71. Typical $R_{pull-up}$ vs. V_{DD} with $V_{IN}=V_{SS}$



CONTROL PIN CHARACTERISTICS (Cont'd)

10.10 TIMER PERIPHERAL CHARACTERISTICS

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

Refer to I/O port characteristics for more details on the input/output alternate function characteristics (TIMER).

10.10.1 Watchdog Timer

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$t_{w(WDG)}$	Watchdog time-out duration		3,072		196,608	t_{INT}
		f_{CPU}=4MHz	0.768		49.152	ms
		f_{CPU}=8MHz	0.384		24.576	ms

10.10.2 8-Bit Timer

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{EXT}	Timer external clock frequency		0		$f_{INT}/4$	MHz
t_w	Pulse width at TIMER pin	VDD>4.5V	125			ns
		VDD=3V	1			µs

10.11 8-BIT ADC CHARACTERISTICS

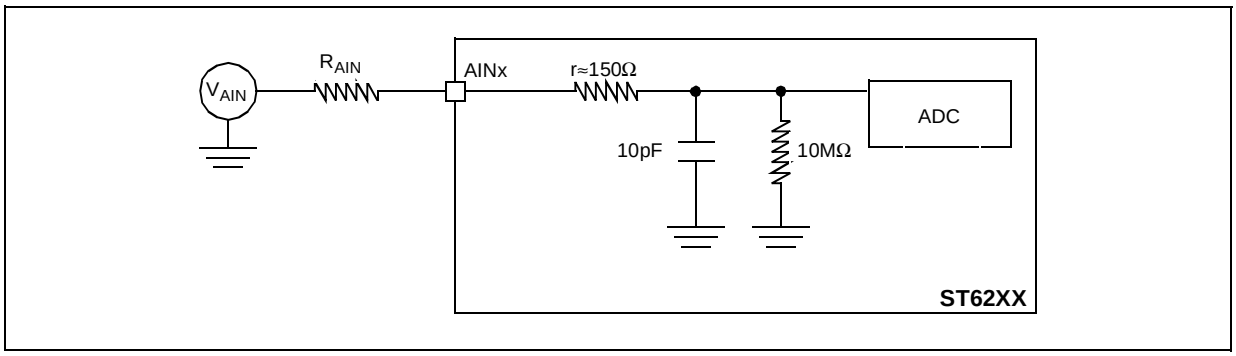
Subject to general operating conditions for V_{DD}, f_{OSC}, and T_A unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ [1]	Max	Unit
f_{OSC}	Clock frequency		1.2		f_{OSC}	MHz
V_{AIN}	Conversion range voltage [2]		V_{SS}		V_{DD}	V
R_{AIN}	External input resistor				10 [3]	kΩ
t_{ADC}	Total convertion time	f_{OSC}=8MHz f_{OSC}=4MHz		70 140		µs
t_{STAB}	Stabilization time [4]			2	4	t_{CPU}
		f_{OSC}=8MHz		3.25	6.5	µs
AD_I	Analog input current during conversion				1.0	µA
AC_{IN}	Analog input capacitance			2	5	pF

Notes:

1. Unless otherwise specified, typical data are based on T_A=25°C and V_{DD}=5V.
2. The ADC refers to V_{DD} and V_{SS}.
3. Any added external serial resistor will downgrade the ADC accuracy (especially for resistance greater than 10kΩ). Data based on characterization results, not tested in production.
4. As a stabilization time for the AD converter is required, the first conversion after the enable can be wrong.

Figure 72. Typical Application with ADC



Note: ADC not present on some devices. See device summary on page 1.

8-BIT ADC CHARACTERISTICS (Cont'd)

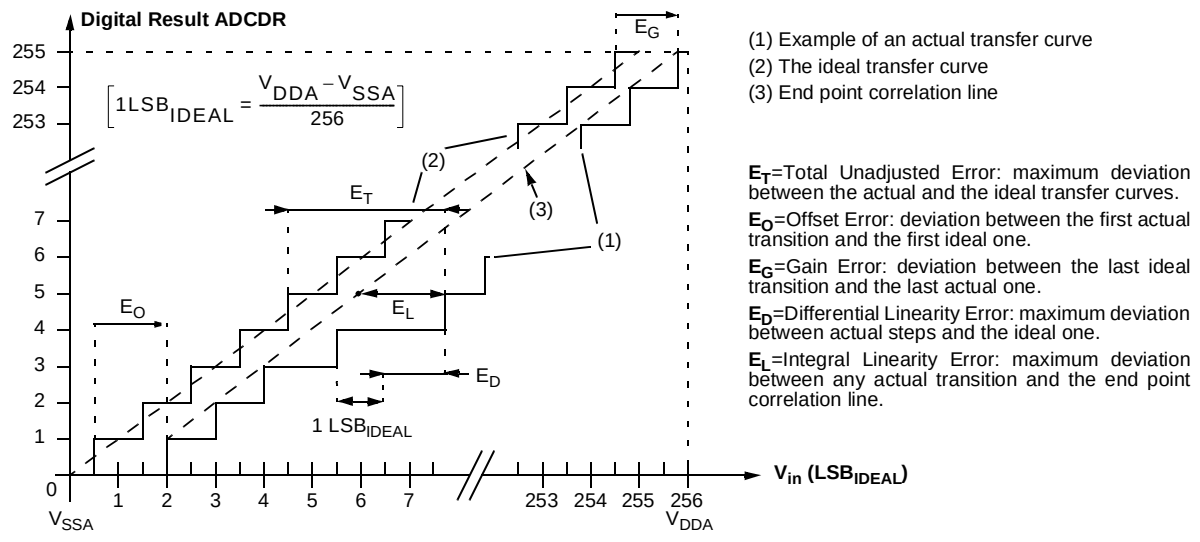
ADC Accuracy

Symbol	Parameter	Conditions	Min	Typ.	Max	Unit
$\|E_T\|$	Total unadjusted error [1]	V_{DD}=5V [2] f_{OSC}=8MHz		1.2	±2, fosc>1.2MHz ±4, fosc>32KHz	LSB
E_O	Offset error [1]			0.72		
E_G	Gain Error [1]			-0.31		
$\|E_D\|$	Differential linearity error [1]			0.54		
$\|E_L\|$	Integral linearity error [1]					

Notes:

1. Negative injection disturbs the analog performance of the device. In particular, it induces leakage currents throughout the device including the analog inputs. To avoid undesirable effects on the analog functions, care must be taken:
   - Analog input pins must have a negative injection less than 1mA (assuming that the impedance of the analog voltage is lower than the specified limits).
   - Pure digital pins must have a negative injection less than 1mA. In addition, it is recommended to inject the current as far as possible from the analog input pins.
2. Data based on characterization results over the whole temperature range, monitored in production.

Figure 73. ADC Accuracy Characteristics

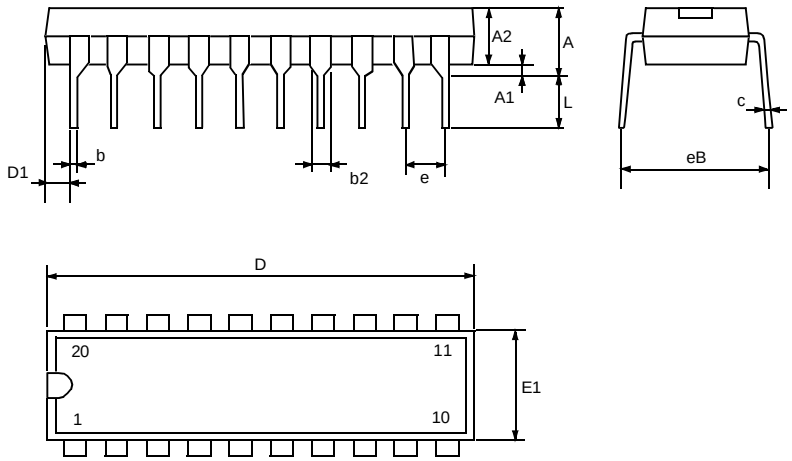


Note: ADC not present on some devices. See device summary on page 1.

11 GENERAL INFORMATION

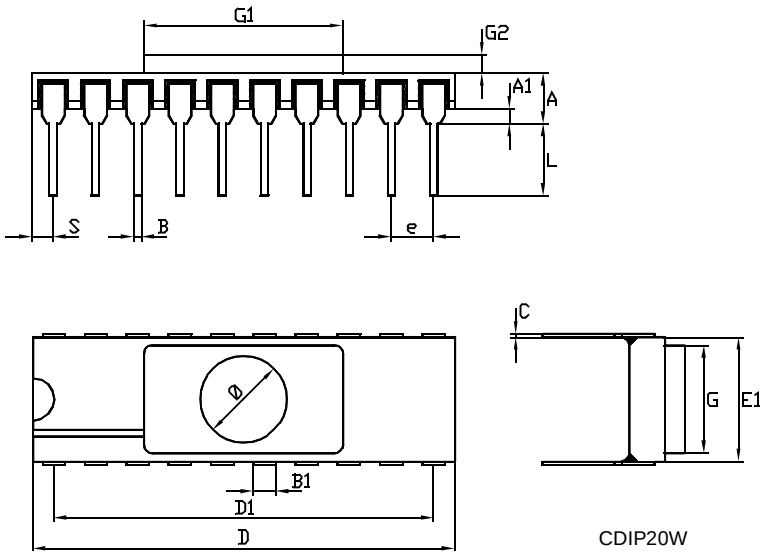
11.1 PACKAGE MECHANICAL DATA

Figure 74. 20-Pin Plastic Dual In-Line Package, 300-mil Width



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			5.33			0.210
A1	0.38			0.015		
A2	2.92	3.30	4.95	0.115	0.130	0.195
b	0.36	0.46	0.56	0.014	0.018	0.022
b2	1.14	1.52	1.78	0.045	0.060	0.070
c	0.20	0.25	0.36	0.008	0.010	0.014
D	24.89	26.16	26.92	0.980	1.030	1.060
D1	0.13			0.005		
e		2.54			0.100	
eB			10.92			0.430
E1	6.10	6.35	7.11	0.240	0.250	0.280
L	2.92	3.30	3.81	0.115	0.130	0.150
	Number of Pins					
N	20					

Figure 75. 20-Pin Ceramic Side-Brazed Dual In-Line Package

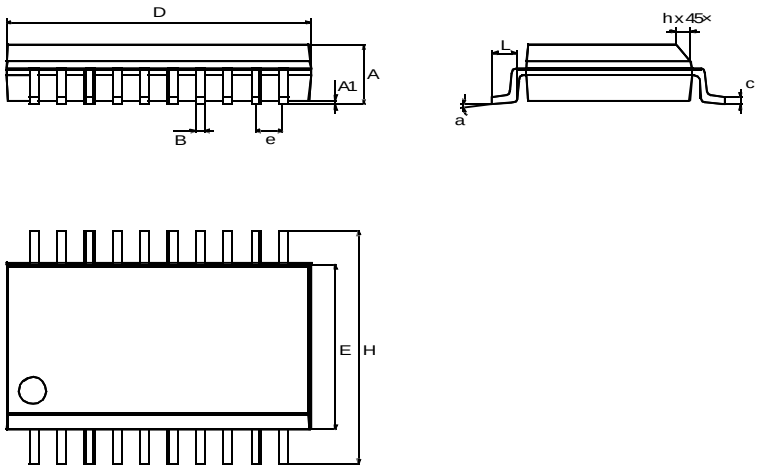


CDIP20W

Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			3.63			0.143
A1	0.38			0.015		
B	3.56	0.46	0.56	0.140	0.018	0.022
B1	1.14	12.70	1.78	0.045	0.500	0.070
C	0.20	0.25	0.36	0.008	0.010	0.014
D	24.89	25.40	25.91	0.980	1.000	1.020
D1		22.86			0.900	
E1	6.99	7.49	8.00	0.275	0.295	0.315
e		2.54			0.100	
G	6.35	6.60	6.86	0.250	0.260	0.270
G1	9.47	9.73	9.98	0.373	0.383	0.393
G2		1.14			0.045	
L	2.92	3.30	3.81	0.115	0.130	0.150
S		12.70			0.500	
Ø		4.22			0.166	
	Number of Pins					
N	20					

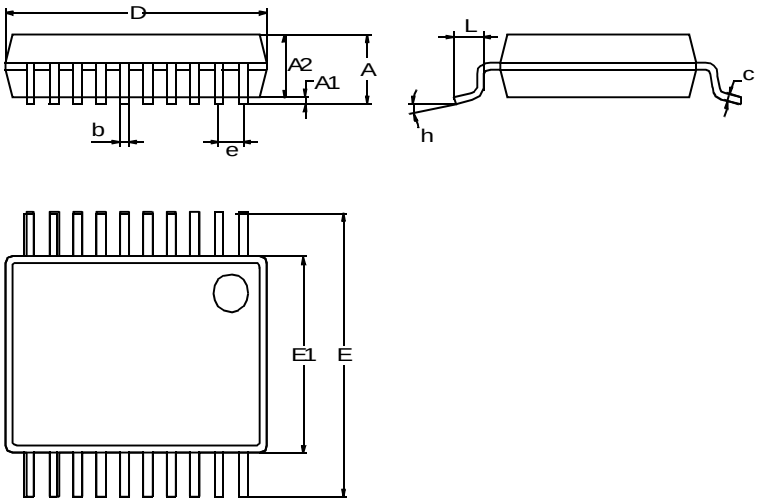
PACKAGE MECHANICAL DATA (Cont'd)

Figure 76. 20-Pin Plastic Small Outline Package, 300-mil Width



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A	2.35		2.65	0.093		0.104
A1	0.10		0.30	0.004		0.012
B	0.33		0.51	0.013		0.020
C	0.23		0.32	0.009		0.013
D	12.60		13.00	0.496		0.512
E	7.40		7.60	0.291		0.299
e		1.27			0.050	
H	10.00		10.65	0.394		0.419
h	0.25		0.75	0.010		0.030
α	0°		8°	0°		8°
L	0.40		1.27	0.016		0.050
	Number of Pins					
N	20					

Figure 77. 20-Pin Plastic Shrink Small Outline Package



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			2.00			0.079
A1	0.05			0.002		
A2	1.65	1.75	1.85	0.065	0.069	0.073
b	0.22		0.38	0.009		0.015
c	0.09		0.25	0.004		0.010
D	6.90	7.20	7.50	0.272	0.283	0.295
E	7.40	7.80	8.20	0.291	0.307	0.323
E1	5.00	5.30	5.60	0.197	0.209	0.220
e		0.65			0.026	
θ	0°	4°	8°	0°	4°	8°
L	0.55	0.75	0.95	0.022	0.030	0.037
	Number of Pins					
N	20					

11.2 THERMAL CHARACTERISTICS

Symbol	Ratings	Value	Unit
R_{thJA}	Package thermal resistance (junction to ambient) DIP20 SO20 SSOP20	 60 80 115	°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.

11.3 SOLDERING AND GLUEABILITY INFORMATION

Recommended soldering information given only as design guidelines in Figure 78 and Figure 79.

Recommended glue for SMD plastic packages:

- Heraeus: PD945, PD955
- Loctite: 3615, 3298

Figure 78. Recommended Wave Soldering Profile (with 37% Sn and 63% Pb)

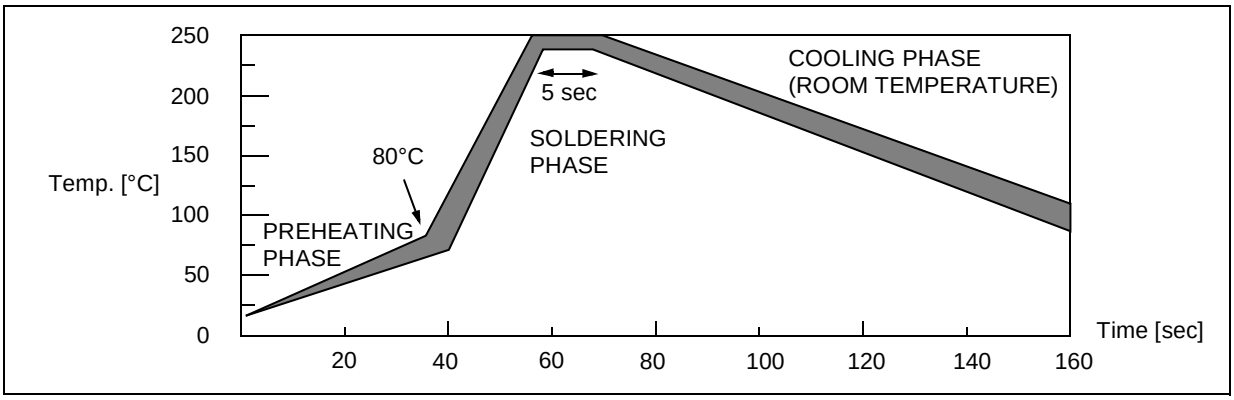
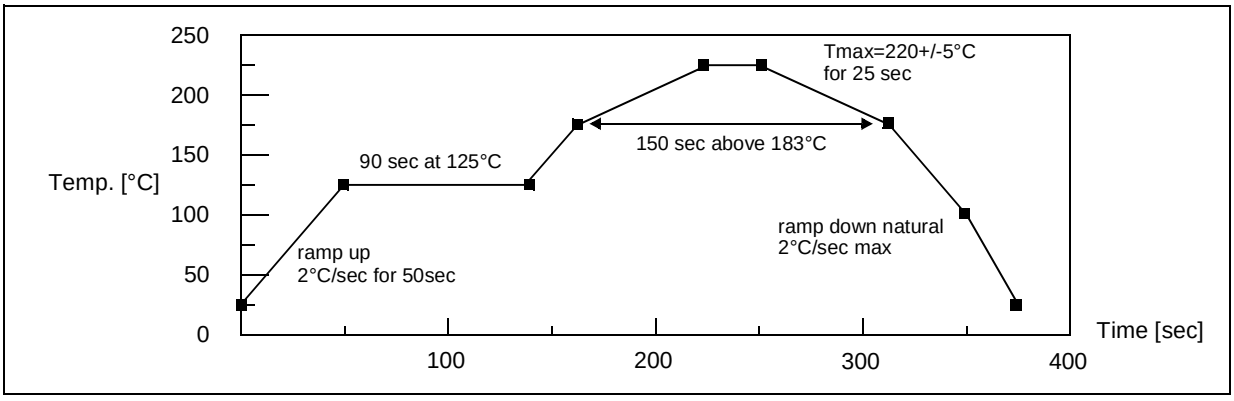


Figure 79. Recommended Reflow Soldering Oven Profile (MID JEDEC)



11.4 PACKAGE/SOCKET FOOTPRINT PROPOSAL

Table 23. Suggested List of DIP20 Socket Types

Package / Probe	Adaptor / Socket Reference		Same Footprint	Socket Type
DIP20	TEXTOOL	220-33-42	X	Textool

Table 24. Suggested List of SO20 Socket Types

Package / Probe	Adaptor / Socket Reference		Same Footprint	Socket Type
SO20	ENPLAS	OTS-20-1.27-04		Open Top
	YAMAICHI	IC51-0202-714		Clamshell
EMU PROBE	Adapter from SO20 to DIP20 footprint (delivered with emulator)		X	SMD to DIP
Programming Adapter	Logical Systems	PA20SO1-08H-6	X	Open Top

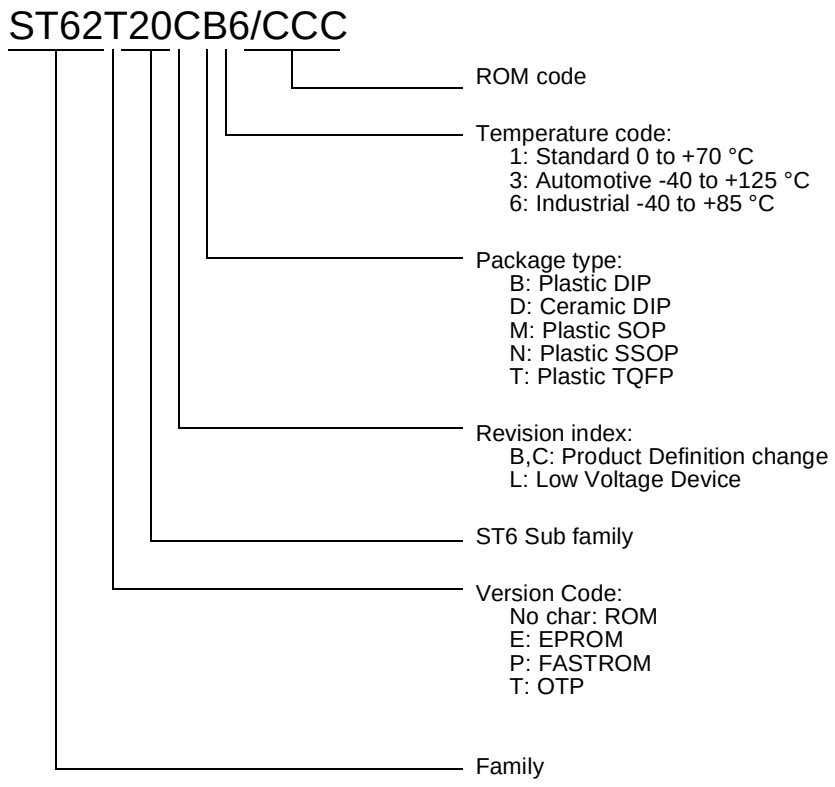
Table 25. Suggested List of SSOP20 Socket Types

Package / Probe	Adaptor / Socket Reference		Same Footprint	Socket Type
SSOP20	ENPLAS	OTS-20-0.65-01	X	Open Top
Programming Adapter	Logical Systems	PA20SS-OT-6	X	Open Top

11.5 ORDERING INFORMATION

The following section deals with the procedure for transfer of customer codes to STMicroelectronics and also details the ST6 factory coded device type.

Figure 80. ST6 Factory Coded Device Types



11.6 TRANSFER OF CUSTOMER CODE

Customer code is made up of the ROM contents and the list of the selected FASTROM options. The ROM contents are to be sent on diskette, or by electronic means, with the hexadecimal file generated by the development tool. All unused bytes must be set to FFh.

The selected options are communicated to STMicroelectronics using the correctly filled OPTION LIST appended. See page 97.

The STMicroelectronics Sales Organization will be pleased to provide detailed information on contractual points.

Listing Generation and Verification. When STMicroelectronics receives the user's ROM contents, a computer listing is generated from it. This listing refers exactly to the ROM contents and options which will be used to produce the specified MCU. The listing is then returned to the customer who must thoroughly check, complete, sign and return it to STMicroelectronics. The signed listing forms a part of the contractual agreement for the production of the specific customer MCU.

11.6.1 FASTROM version

The ST62P08C/P09C/P10C and P20C are the **F**actory **A**dvanced **S**ervice **T**echnique ROM (FASTROM) versions of ST62T08C, T09C, T10C and T20C OTP devices.

They offer the same functionality as OTP devices, but they do not have to be programmed by the customer. The customer code must be sent to STMicroelectronics in the same way as for ROM devices. The FASTROM option list has the same options as defined in the programmable option byte of the OTP version. It also offers an identifier option. If this option is enabled, each FASTROM device is programmed with a unique 5-byte number which is mapped at addresses 0F9Bh-0F9Fh. The user must therefore leave these bytes blanked.

The identification number is structured as follows:

0F9Bh	T0
0F9Ch	T1
0F9Dh	T2
0F9Eh	T3
0F9Fh	Test ID

with T0, T1, T2, T3 = time in seconds since 01/01/1970 and Test ID = Tester Identifier.

TRANSFER OF CUSTOMER CODE (Cont'd)

ST6208C/09C/10C/20C/P08C/P09C/P10C/P20C MICROCONTROLLER OPTION LIST

Customer: . . . . . .
Address: . . . . . .
. . . . . .
Contact: . . . . . .
Phone: . . . . . .
Reference: . . . . . .

STMicroelectronics references:

Device: [ ] ST6208C (1 KB) [ ] ST6209C (1 KB)
[ ] ST6210C (2 KB) [ ] ST6220C (4 KB)
[ ] ST62P08C (1 KB) [ ] ST62P09C (1 KB)
[ ] ST62P10C (2 KB) [ ] ST62P20C (4 KB)

Package: [ ] Dual in Line Plastic
[ ] Small Outline Plastic with conditioning
[ ] Shrink Small Outline Plastic with conditioning

Conditioning option: [ ] Standard (Tube) [ ] Tape & Reel

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

Marking: [ ] Standard marking
[ ] Special marking (ROM only):
PDIP20 (10 char. max): _ _ _ _ _ _ _ _ _ _
SO20 (8 char. max): _ _ _ _ _ _ _ _
SSOP20 (11 char. max): _ _ _ _ _ _ _ _ _ _ _
Authorized characters are letters, digits, '.', '-', '/' and spaces only.

Oscillator Safeguard: [ ] Enabled [ ] Disabled
Watchdog Selection: [ ] Software Activation [ ] Hardware Activation
Timer pull-up: [ ] Enabled [ ] Disabled
NMI pull-up: [ ] Enabled [ ] Disabled
Oscillator Selection: [ ] Quartz crystal / Ceramic resonator
[ ] RC network

Readout Protection: FASTROM:
[ ] Enabled [ ] Disabled
ROM:
[ ] Enabled:
[ ] Fuse is blown by STMicroelectronics
[ ] Fuse can be blown by the customer
[ ] Disabled

Low Voltage Detector: [ ] Enabled [ ] Disabled
External STOP Mode Control: [ ] Enabled [ ] Disabled
Identifier (FASTROM only): [ ] Enabled [ ] Disabled

Comments:

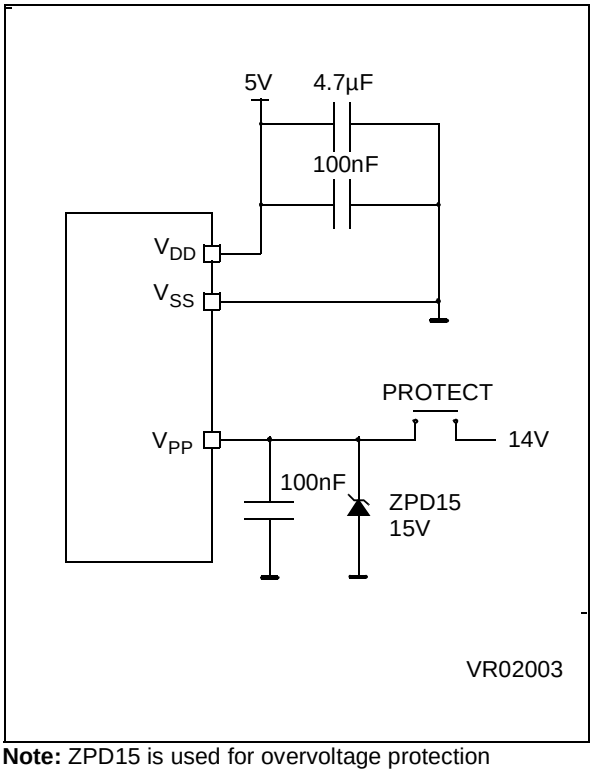
Oscillator Frequency in the application: . . . . . .
Supply Operating Range in the application: . . . . . .
Notes: . . . . . .
Date: . . . . . .
Signature: . . . . . .

11.6.2 ROM VERSION

The ST6208C, 09C, 10C and 20C are mask programmed ROM version of ST62T08C, T09C, T10C and T20C OTP devices.

They offer the same functionality as OTP devices, selecting as ROM options the options defined in the programmable option byte of the OTP version.

Figure 81. Programming Circuit

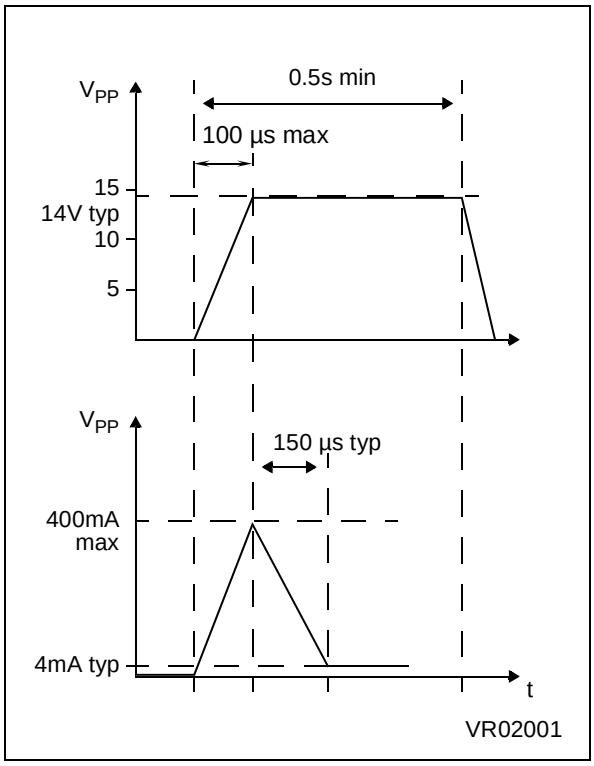


Note: ZPD15 is used for overvoltage protection

ROM Readout Protection. If the ROM READOUT PROTECTION option is selected, a protection fuse can be blown to prevent any access to the program memory content.

In case the user wants to blow this fuse, high voltage must be applied on the V_{PP} pin.

Figure 82. Programming wave form



12 DEVELOPMENT TOOLS

STMicroelectronics offers a range of hardware and software development tools for the ST6 microcontroller family. Full details of tools available for the ST6 from third party manufacturers can be obtain from the STMicroelectronics Internet site: ➟ http://mcu.st.com.

Table 26. Dedicated Third Parties Development Tools

Third Party [1)]	Designation	ST Sales Type	Web site address
ACTUM	ST-REALIZER II: Graphical Schematic based Development available from STMicroelectronics.	STREALIZER-II	http://www.actum.com/
CEIBO	Low cost emulator available from CEIBO.		http://www.ceibo.com/
RAISONANCE	This tool includes in the same environment: an assembler, linker, C compiler, debugger and simulator. The assembler package (plus limited C compiler) is free and can be downloaded from raisonance web site. The full version is available both from STMicroelectronics and Raisonance.	ST6RAIS-SWC/PC	http://www.raisonance.com/
SOFTEC	High end emulator available from SOFTEC.		http://www.softecmicro.com/
	Gang programmer available from SOFTEC.		
ADVANCED EQUIPMENT	Single and gang programmers		http://www.aec.com.tw/
ADVANCED TRANSDATA			http://www.adv-transdata.com/
BP MICROSYSTEMS			http://www.bpmicro.com/
DATA I/O			http://www.data-io.com/
DATAMAN			http://www.dataman.com/
EE TOOLS			http://www.eetools.com/
ELNEC			http://www.elnec.com/
HI-LO SYSTEMS			http://www.hilosystems.com.tw/
ICE TECHNOLOGY			http://www.icetech.com/
LEAP			http://www.leap.com.tw/
LLOYD RESEARCH			http://www.lloyd-research.com/
LOGICAL DEVICES			http://www.chipprogrammers.com/
MQP ELECTRONICS			http://www.mqp.com/
NEEDHAMS ELECTRONICS			http://www.needhams.com/
STAG PROGRAMMERS			http://www.stag.co.uk/
SYSTEM GENERAL CORP			http://www.sg.com.tw
TRIBAL MICROSYSTEMS			http://www.tribalmicro.com/
XELTEK			http://www.xeltek.com/

Note 1: For latest information on third party tools, please visit our Internet site: ➟ http://mcu.st.com.

DEVELOPMENT TOOLS (Cont'd)

STMicroelectronics Tools

Four types of development tool are offered by ST, all of them connect to a PC via a parallel or serial port: see Table 27 and Table 28 for more details.

Table 27. STMicroelectronics Tool Features

	Emulation Type	Programming Capability	Software Included
ST6 Starter Kit	Device simulation (limited emulation as interrupts are not supported)	Yes (DIP packages only)	MCU CD ROM with: – Rkit-ST6 from Raisonance – ST6 Assembly toolchain – WGDB6 powerful Source Level Debugger for Win 3.1, Win 95 and NT – Various software demo versions. – Windows Programming Tools for Win 3.1, Win 95 and NT
ST6 HDS2 Emulator	In-circuit powerful emulation features including trace/ logic analyzer	No	
ST6 EPROM Programmer Board	No	Yes	

Table 28. Dedicated STMicroelectronics Development Tools

Supported Products	ST6 Starter Kit	ST6 HDS2 Emulator	ST6 Programming Board
ST6208C, ST6209C, ST6210C and ST6220C	ST622XC-KIT	Complete: ST62GP-EMU2 Dedication board: ST62GP-DBE	ST62E2XC-EPB

13 ST6 APPLICATION NOTES

IDENTIFICATION	DESCRIPTION
MOTOR CONTROL	
AN392	MICROCONTROLLER AND TRIACS ON THE 110/240V MAINS
AN414	CONTROLLING A BRUSH DC MOTOR WITH AN ST6265 MCU
AN416	SENSORLESS MOTOR DRIVE WITH THE ST62 MCU + TRIAC
AN422	IMPROVES UNIVERSAL MOTOR DRIVE
AN863	IMPROVED SENSORLESS CONTROL WITH THE ST62 MCU FOR UNIVERSAL MOTOR
BATTERY MANAGEMENT	
AN417	FROM NICD TO NIMH FAST BATTERY CHARGING
AN433	ULTRA FAST BATTERY CHARGER USING ST6210 MICROCONTROLLER
AN859	AN INTELLIGENT ONE HOUR MULTICHARGER FOR Li-Ion, NiMH and NiCd BATTERIES
HOME APPLIANCE	
AN674	MICROCONTROLLERS IN HOME APPLIANCES: A SOFT REVOLUTION
AN885	ST62 MICROCONTROLLERS DRIVE HOME APPLIANCE MOTOR TECHNOLOGY
GRAPHICAL DESIGN	
AN676	BATTERY CHARGER USING THE ST6-REALIZER
AN677	PAINLESS MICROCONTROLLER CODE BY GRAPHICAL APPLICATION DESCRIPTION
AN839	ANALOG MULTIPLE KEY DECODING USING THE ST6-REALIZER
AN840	CODED LOCK USING THE ST6-REALIZER
AN841	A CLOCK DESIGN USING THE ST6-REALIZER
AN842	7 SEGMENT DISPLAY DRIVE USING THE ST6-REALIZER
COST REDUCTION	
AN431	USING ST6 ANALOG INPUTS FOR MULTIPLE KEY DECODING
AN594	DIRECT SOFTWARE LCD DRIVE WITH ST621X AND ST626X
AN672	OPTIMIZING THE ST6 A/D CONVERTER ACCURACY
AN673	REDUCING CURRENT CONSUMPTION AT 32KHZ WITH ST62
DESIGN IMPROVEMENTS	
AN420	EXPANDING A/D RESOLUTION OF THE ST6 A/D CONVERTER
AN432	USING ST62XX I/O PORTS SAFELY
AN434	MOVEMENT DETECTOR CONCEPTS FOR NOISY ENVIRONMENTS
AN435	DESIGNING WITH MICROCONTROLLERS IN NOISY ENVIRONMENTS
AN669	SIMPLE RESET CIRCUITS FOR THE ST6
AN670	OSCILLATOR SELECTION FOR ST62
AN671	PREVENTION OF DATA CORRUPTION IN ST6 ON-CHIP EEPROM
AN911	ST6 MICRO IS EMC CHAMPION
AN975	UPGRADING FROM ST625X/6XB TO ST625X/6XC
AN1015	SOFTWARE TECHNIQUES FOR IMPROVING ST6 EMC PERFORMANCE
PERIPHERAL OPERATIONS	
AN590	PWM GENERATION WITH ST62 AUTO-RELOAD TIMER
AN591	INPUT CAPTURE WITH ST62 AUTO-RELOAD TIMER
AN592	PLL GENERATION USING THE ST62 AUTO-RELOAD TIMER
AN593	ST62 IN-CIRCUIT PROGRAMMING
AN678	LCD DRIVING WITH ST6240

IDENTIFICATION	DESCRIPTION
AN913	PWM GENERATION WITH ST62 16-BIT AUTO-RELOAD TIMER
AN914	USING ST626X SPI AS UART
AN1016	ST6 USING THE ST623XB/ST628XB UART
AN1050	ST6 INPUT CAPTURE WITH ST62 16-BIT AUTO-RELOAD TIMER
AN1127	USING THE ST62T6XC/5XC SPI IN MASTER MODE
GENERAL	
AN683	MCUS - 8/16-BIT MICROCONTROLLERS (MCUS) APPLICATION NOTES ABSTRACTS BY TOPICS
AN886	SELECTING BETWEEN ROM AND OTP FOR A MICROCONTROLLER
AN887	MAKING IT EASY WITH MICROCONTROLLERS
AN898	EMC GENERAL INFORMATION
AN899	SOLDERING RECOMMENDATIONS AND PACKAGING INFORMATION
AN900	INTRODUCTION TO SEMICONDUCTOR TECHNOLOGY
AN901	EMC GUIDE-LINES FOR MICROCONTROLLER - BASED APPLICATIONS
AN902	QUALITY AND RELIABILITY INFORMATION
AN912	A SIMPLE GUIDE TO DEVELOPMENT TOOLS
AN1181	ELECTROSTATIC DISHARGE SENSITIVITY MEASUREMENT

14 SUMMARY OF CHANGES

Description of the changes between the current release of the specification and the previous one.

Revision	Main Changes	Date
3.1	Pinout: Pin 10, TEST function is changed to non-user info. See Figure 2. Modification of Caution (Read-Modify-Write instructions) in Section 3.1.6.1. Addition of power consumption information in Section 3.3. Addition of Note in Section 7.2.3. Modification of Table 10. Modification of text (READ-MODIFY-WRITE instructions) in Section 7.2.5. Modification of bit names in ADCR and ADR registers in Section 8.3. Modification of Electrical Characteristics in Section 10.2.1 (Add. of V_{OUT}) and in Section 10.5.2 (Del. of tw and tr).	15 Jan 2001
3.2	Removed “TEST” in Figure 1. Updated Table 2 on page 12: CPU added for 0C8h and ROM added for 0C9h. Changed bits 5:0 to bits 7:2 for T[5:0] bits (WDGR register on Section 8.1.7 on page 45) Changed $4.5V \leq V_{DD} \leq 5.5V$ to $4.5V \leq V_{DD} \leq 6.0V$ (Section 10 on page 62). Added one note on RC oscillator (Section 10.5.4 on page 74). Changed Figure 52 title on page 75 (vs V_{DD} instead of vs R_{NET}). Updated Electrical Sensitivities table on page 79. Changed Figure 74 on page 90 and Figure 76 and Figure 77 on page 91 Changed Section 11.6.1 on page 96. Changed option list (page 97). Updated Section 12 on page 99 and Section 13 on page 101.	July 2001

15 TO GET MORE INFORMATION

To get the latest information on this product please use the STMicroelectronics web server.

➟ http://mcu.st.com/

Notes:

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics

©2001 STMicroelectronics - All Rights Reserved.

Purchase of I^2C Components by STMicroelectronics conveys a license under the Philips I^2C Patent. Rights to use these components in an I^2C system is granted provided that the system conforms to the I^2C Standard Specification as defined by Philips.

STMicroelectronics Group of Companies
Australia - Brazil - China - Finland - France - Germany - Hong Kong - India - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain
Sweden - Switzerland - United Kingdom - U.S.A.

http://www.st.com