Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

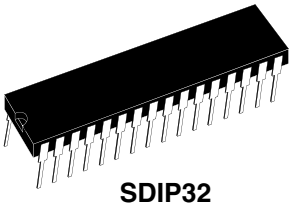
Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

ST72260Gx, ST72262Gx, ST72264Gx

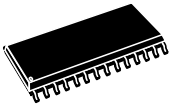
8-BIT MCU WITH FLASH OR ROM MEMORY, ADC, TWO 16-BIT TIMERS, I²C, SPI, SCI INTERFACES

- **Memories**
  - 4 K or 8 Kbytes Program memory: ROM or Single voltage extended Flash (XFlash) with read-out protection write protection and In-Circuit Programming and In-Application Programming (ICP and IAP). 10K write/erase cycles guaranteed, data retention: 20 years at 55°C.
  - 256 bytes RAM
- **Clock, Reset and Supply Management**
  - Enhanced reset system
  - Enhanced low voltage supply supervisor (LVD) with 3 programmable levels and auxiliary voltage detector (AVD) with interrupt capability for implementing safe power-down procedures
  - Clock sources: crystal/ceramic resonator oscillators, internal RC oscillator and bypass for external clock
  - PLL for 2x frequency multiplication
  - Clock-out capability
  - 4 Power Saving Modes: Halt, Active Halt,Wait and Slow
- **Interrupt Management**
  - Nested interrupt controller
  - 10 interrupt vectors plus TRAP and RESET
  - 22 external interrupt lines (on 2 vectors)
- **22 I/O Ports**
  - 22 multifunctional bidirectional I/O lines
  - 20 alternate function lines
  - 8 high sink outputs
- **4 Timers**
  - Main Clock Controller with Real time base and Clock-out capabilities
  - Configurable watchdog timer



SDIP32

LFBGA 6x6mm



SO28

  - Two 16-bit timers with: 2 input captures, 2 output compares, external clock input on one timer, PWM and Pulse generator modes
- **3 Communication Interfaces**
  - SPI synchronous serial interface
  - I²C multimaster interface (SMBus V1.1 Compliant)
  - SCI asynchronous serial interface
- **1 Analog peripheral**
  - 10-bit ADC with 6 input channels
- **Instruction Set**
  - 8-bit data manipulation
  - 63 basic instructions with illegal opcode detection
  - 17 main addressing modes
  - 8 x 8 unsigned multiply instruction
- **Development Tools**
  - Full hardware/software development package

Device Summary

Features	ST72260G1	ST72262G1	ST72262G2	ST72264G1	ST72264G2	
Program memory - bytes	4K	4K	8K	4K	8K	
RAM (stack) - bytes	256 (128)					
Peripherals	Watchdog timer, RTC, Two16-bit timers, SPI	Watchdog timer, RTC, Two 16-bit timers, SPI, ADC		Watchdog timer, RTC, Two 16-bit timers, SPI, SCI, I²C, ADC		
Operating Supply	2.7 V to 5.5 V					
CPU Frequency	Up to 8 MHz (with oscillator up to 16 MHz) PLL 4/8 MHz					
Operating Temperature	-40° C to +85° C				-40° C to +85° C	0° C to +70° C / -40° C to +85° C
Packages	SO28 / SDIP32				SO28 / SDIP32	LFBGA

Rev. 3

Table of Contents

1 INTRODUCTION 5
2 PIN DESCRIPTION 6
3 REGISTER & MEMORY MAP 10
4 FLASH PROGRAM MEMORY 14
4.1 INTRODUCTION 14
4.2 MAIN FEATURES 14
4.3 PROGRAMMING MODES 14
4.4 ICC INTERFACE 15
4.5 MEMORY PROTECTION 16
4.6 RELATED DOCUMENTATION 16
4.7 REGISTER DESCRIPTION 16
5 CENTRAL PROCESSING UNIT 17
5.1 INTRODUCTION 17
5.2 MAIN FEATURES 17
5.3 CPU REGISTERS 17
6 SUPPLY, RESET AND CLOCK MANAGEMENT 20
6.1 PHASE LOCKED LOOP 20
6.2 MULTI-OSCILLATOR (MO) 21
6.3 RESET SEQUENCE MANAGER (RSM) 22
6.4 SYSTEM INTEGRITY MANAGEMENT (SI) 24
7 INTERRUPTS 28
7.1 INTRODUCTION 28
7.2 MASKING AND PROCESSING FLOW 28
7.3 INTERRUPTS AND LOW POWER MODES 30
7.4 CONCURRENT & NESTED MANAGEMENT 30
7.5 INTERRUPT REGISTER DESCRIPTION 31
8 POWER SAVING MODES 33
8.1 INTRODUCTION 33
8.2 SLOW MODE 33
8.3 WAIT MODE 34
8.4 ACTIVE-HALT AND HALT MODES 35
8.5 HALT MODE 36
9 I/O PORTS 38
9.1 INTRODUCTION 38
9.2 FUNCTIONAL DESCRIPTION 38
9.3 I/O PORT IMPLEMENTATION 41
9.4 UNUSED I/O PINS 41
9.5 LOW POWER MODES 41
9.6 INTERRUPTS 41
9.7 DEVICE-SPECIFIC I/O PORT CONFIGURATION 42
9.8 I/O PORT REGISTER DESCRIPTION 43
10 MISCELLANEOUS REGISTERS 45

Table of Contents

10.1 I/O PORT INTERRUPT SENSITIVITY 45
10.2 I/O PORT ALTERNATE FUNCTIONS 45
10.3 MISCELLANEOUS REGISTER DESCRIPTION 46
11 ON-CHIP PERIPHERALS 48
11.1 WATCHDOG TIMER (WDG) 48
11.2 MAIN CLOCK CONTROLLER WITH REAL TIME CLOCK (MCC/RTC) 53
11.3 16-BIT TIMER 55
11.4 SERIAL PERIPHERAL INTERFACE (SPI) 75
11.5 SERIAL COMMUNICATIONS INTERFACE (SCI) 87
11.6 I2C BUS INTERFACE (I2C) 103
11.7 10-BIT A/D CONVERTER (ADC) 116
12 INSTRUCTION SET 120
12.1 CPU ADDRESSING MODES 120
12.2 INSTRUCTION GROUPS 123
13 ELECTRICAL CHARACTERISTICS 126
13.1 PARAMETER CONDITIONS 126
13.2 ABSOLUTE MAXIMUM RATINGS 127
13.3 OPERATING CONDITIONS 128
13.4 SUPPLY CURRENT CHARACTERISTICS 131
13.5 CLOCK AND TIMING CHARACTERISTICS 135
13.6 MEMORY CHARACTERISTICS 140
13.7 EMC CHARACTERISTICS 141
13.8 I/O PORT PIN CHARACTERISTICS 144
13.9 CONTROL PIN CHARACTERISTICS 150
13.10 TIMER PERIPHERAL CHARACTERISTICS 152
13.11 COMMUNICATION INTERFACE CHARACTERISTICS 153
13.12 10-BIT ADC CHARACTERISTICS 157
14 PACKAGE CHARACTERISTICS 159
14.1 PACKAGE MECHANICAL DATA 159
14.2 THERMAL CHARACTERISTICS 160
14.3 LEAD-FREE PACKAGE INFORMATION 161
15 DEVICE CONFIGURATION AND ORDERING INFORMATION 162
15.1 OPTION BYTES 162
15.2 DEVICE ORDERING INFORMATION AND TRANSFER OF CUSTOMER CODE 164
15.3 DEVELOPMENT TOOLS 166
16 KNOWN LIMITATIONS 168
16.1 ALL FLASH AND ROM DEVICES 168
16.2 FLASH DEVICES ONLY 168
17 REVISION HISTORY 171

To obtain the most recent version of this datasheet,
please check at www.st.com>products>technical literature>datasheet
Please note that the list of known limitations can be found at the end of this document on page 168.

1 INTRODUCTION

The ST72260Gx, ST72262Gx and ST72264Gx devices are members of the ST7 microcontroller family. They can be grouped as follows :

– ST72264Gx devices are designed for mid-range applications with ADC, I^2C and SCI interface capabilities.

– ST72262Gx devices target the same range of applications but without I^2C interface or SCI.

– ST72260Gx devices are for applications that do not need ADC, I^2C peripherals or SCI.

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

The ST72F260G, ST72F262G, and ST72F264G versions feature single-voltage FLASH memory with byte-by-byte In-Circuit Programming (ICP) capabilities.

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

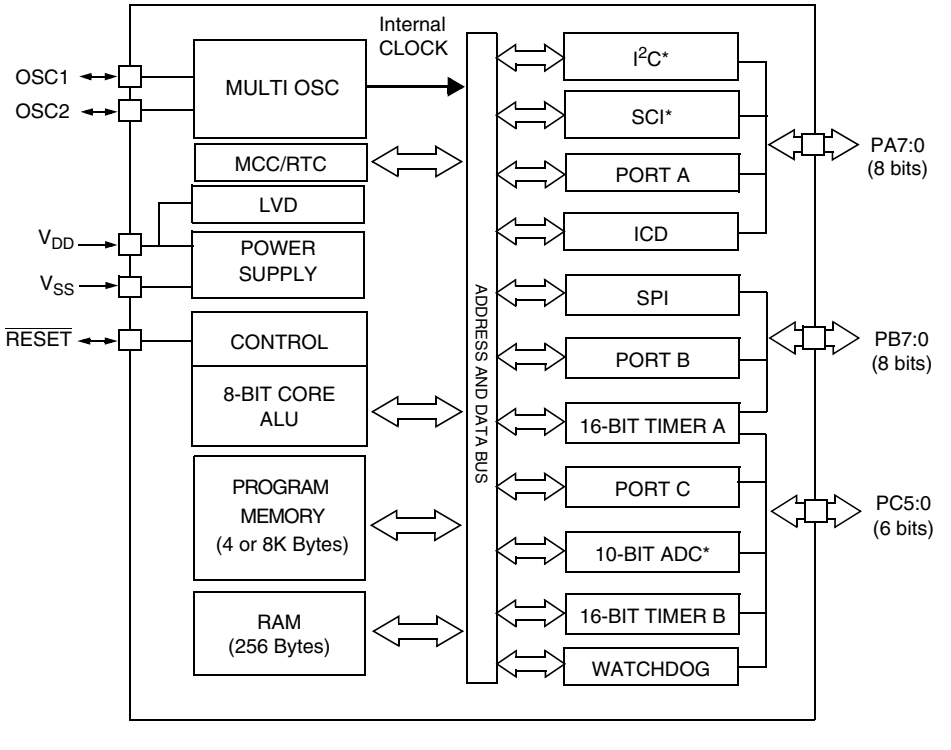
The enhanced instruction set and addressing modes of the ST7 offer both power and flexibility to software developers, enabling the design of highly efficient and compact application code. In addition to standard 8-bit data management, all ST7 microcontrollers feature true bit manipulation, 8x8 unsigned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

For easy reference, all parametric data is located in Section 13 on page 126.

Related Documentation

AN1365: Guidelines for migrating ST72C254 applications to ST72F264

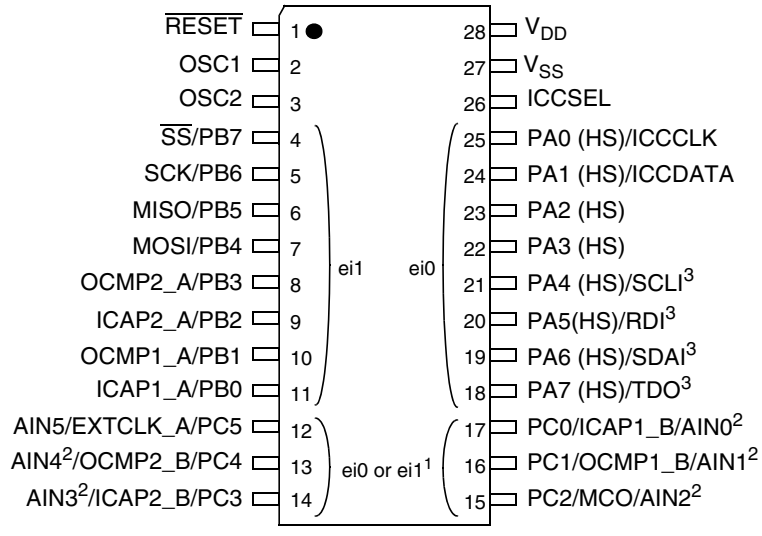
Figure 1. General Block Diagram



*Not available on some devices, see device summary on page 1.

2 PIN DESCRIPTION

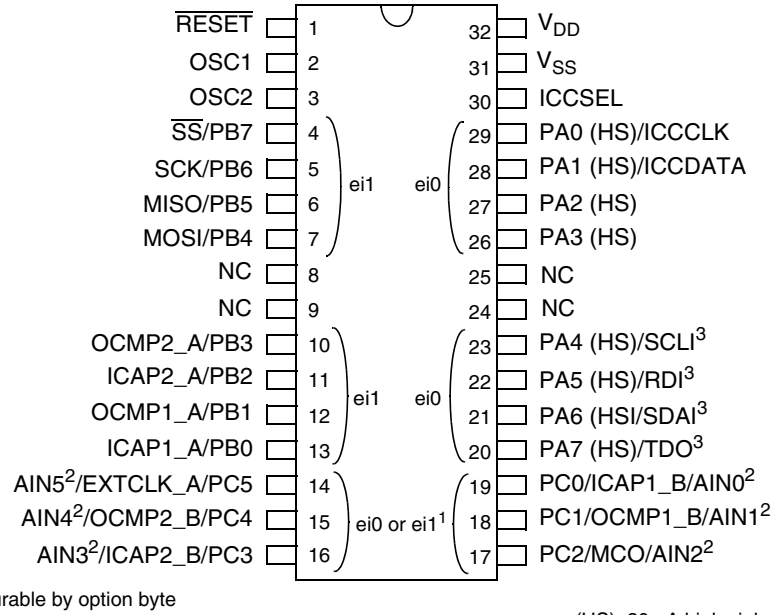
Figure 2. 28-Pin SO Package Pinout



1 Configurable by option byte
2 Alternate function not available on ST72260
3 Alternate function not available on ST72260 and ST72262

(HS) 20mA high sink capability
eiX associated external interrupt vector

Figure 3. 32-Pin SDIP Package Pinout

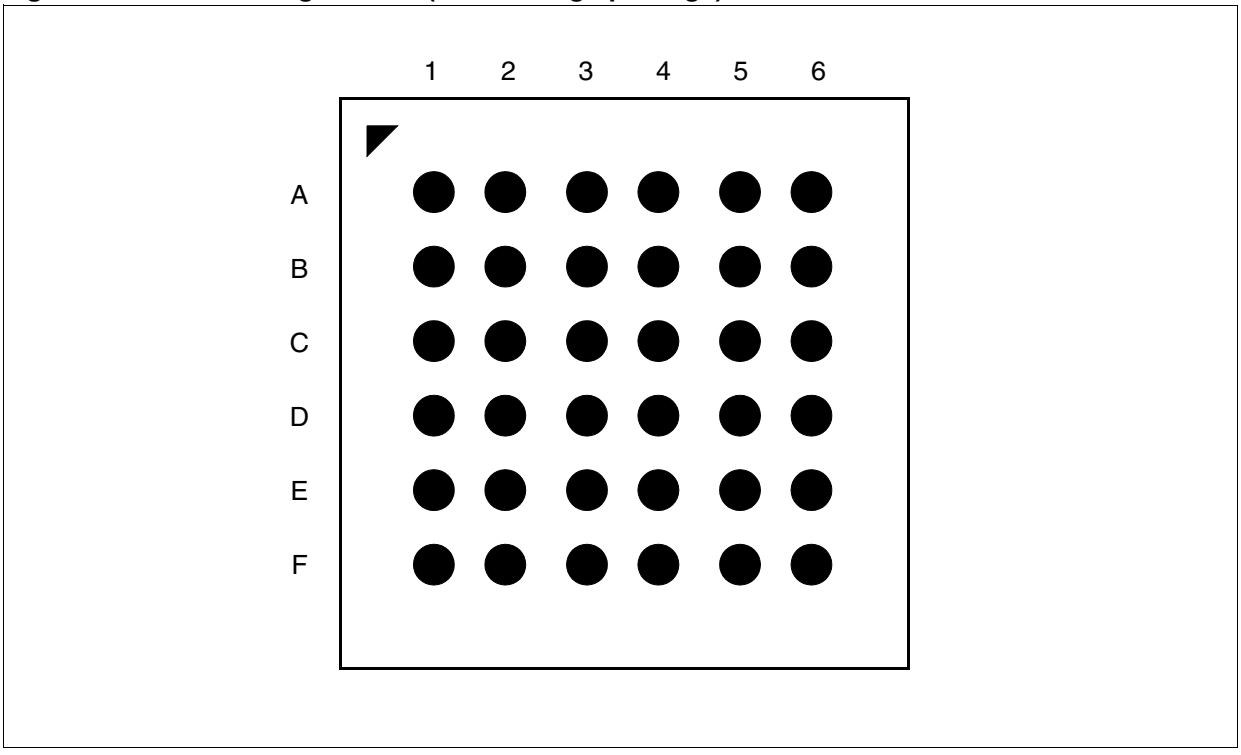


1 Configurable by option byte
2 Alternate function not available on ST72260
3 Alternate function not available on ST72260 and ST72262

(HS) 20mA high sink capability
eiX associated external interrupt vector

PIN DESCRIPTION (Cont'd)

Figure 4. TFBGA Package Pinout (view through package)



PIN DESCRIPTION (Cont'd)

For external pin connection guidelines, refer to Section 13 "ELECTRICAL CHARACTERISTICS" on page 126.

Legend / Abbreviations for Table 1:

Type: I = input, O = output, S = supply

Input level: A = Dedicated analog input

In/Output level: C_T= CMOS 0.3 V_{DD}/0.7 V_{DD} with input trigger

Output level: HS = 20 mA high sink (on N-buffer only)

Port and control configuration:

– Input: float = floating, wpu = weak pull-up, int = interrupt [1], ana = analog

– Output: OD = open drain [2], PP = push-pull

Refer to Section 9 "I/O PORTS" on page 38 for more details on the software configuration of the I/O ports.

The RESET configuration of each pin is shown in bold. This configuration is valid as long as the device is in reset state.

Table 1. Device Pin Description

Pin n°			Pin Name	Type	Level		Port / Control						Main Function (after reset)	Alternate Function
							Input				Output			
SDIP32	SO28	BGA			Input	Output	float	wpu	int	ana	OD	PP		
1	1	A3	$\overline{\text{RESET}}$	I/O	C_T			X			X		Top priority non maskable interrupt (active low)	
2	2	C4	OSC1 [3]	I									External clock input or Resonator oscillator inverter input or resistor input for RC oscillator	
3	3	B3	OSC2 [3]	O									Resonator oscillator inverter output or capacitor input for RC oscillator	
4	4	A2	PB7/$\overline{\text{SS}}$	I/O	C_T		**X**	ei1			X	X	**Port B7**	SPI Slave Select (active low)
5	5	A1	PB6/SCK	I/O	C_T		**X**	ei1			X	X	**Port B6**	SPI Serial Clock
6	6	B1	PB5/MISO	I/O	C_T		**X**	ei1			X	X	**Port B5**	SPI Master In/ Slave Out Data
7	7	B2	PB4/MOSI	I/O	C_T		**X**	ei1			X	X	**Port B4**	SPI Master Out / Slave In Data
8		C1	NC	Not Connected										
9		C2	NC											
		D1	NC											
10	8	C3	PB3/OCMP2_A	I/O	C_T		**X**	ei1			X	X	**Port B3**	Timer A Output Compare 2
11	9	D2	PB2/ICAP2_A	I/O	C_T		**X**	ei1			X	X	**Port B2**	Timer A Input Capture 2
12	10	E1	PB1 /OCMP1_A	I/O	C_T		**X**	ei1			X	X	**Port B1**	Timer A Output Compare 1 **Caution:** Negative current injection not allowed on this pin[4].
13	11	F1	PB0 /ICAP1_A	I/O	C_T		**X**	ei1			X	X	**Port B0**	Timer A Input Capture 1 **Caution:** Negative current injection not allowed on this pin[4].
14	12	F2	PC5/EXTCLK_A/AIN5	I/O	C_T		**X**	ei0/ei1	X		X	X	**Port C5**	Timer A Input Clock or ADC Analog Input 5

Pin n°			Pin Name	Type	Level		Port / Control						Main Function (after reset)	Alternate Function
							Input				Output			
SDIP32	SO28	BGA			Input	Output	float	wpu	int	ana	OD	PP		
15	13	E2	PC4/OCMP2_B/AIN4	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C4**	Timer B Output Compare 2 or ADC Analog Input 4
16	14	F3	PC3/ ICAP2_B/AIN3	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C3**	Timer B Input Capture 2 or ADC Analog Input 3
17	15	E3	PC2/MCO/AIN2	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C2**	Main clock output (f_{CPU}) or ADC Analog Input 2
18	16	F4	PC1/OCMP1_B/AIN1	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C1**	Timer B Output Compare 1 or ADC Analog Input 1
19	17	D3	PC0/ICAP1_B/AIN0	I/O	C_T		**X**	ei0/ei1		X	X	X	**Port C0**	Timer B Input Capture 1 or ADC Analog Input 0
20	18	E4	PA7/TDO	I/O	C_T	HS	**X**	ei0			X	X	**Port A7**	SCI output
21	19	F5	PA6/SDAI	I/O	C_T	HS	**X**		ei0		T		**Port A6**	I^2C DATA
22	20	F6	PA5 /RDI	I/O	C_T	HS	**X**	ei0			X	X	**Port A5**	SCI input
23	21	E6	PA4/SCLI	I/O	C_T	HS	**X**		ei0		T		**Port A4**	I^2C CLOCK
24		E5	NC	Not Connected										
25		D6	NC											
		D5	NC											
26	22	C6	PA3	I/O	C_T	HS	**X**	ei0			X	X	**Port A3**	
27	23	D4	PA2	I/O	C_T	HS	**X**	ei0			X	X	**Port A2**	
		C5	NC	Not Connected										
		B6	NC											
28	24	A6	PA1/ICCDATA	I/O	C_T	HS	**X**	ei0			X	X	**Port A1**	In Circuit Communication Data
29	25	A5	PA0/ICCCLK	I/O	C_T	HS	**X**	ei0			X	X	**Port A0**	In Circuit Communication Clock
30	26	B5	ICCSEL	I	C_T		**X**						ICC mode pin, must be tied low	
31	27	A4	V_{SS}	S									Ground	
32	28	B4	V_{DD}	S									Main power supply	

Notes:

1. In the interrupt input column, “eiX” defines the associated external interrupt vector. If the weak pull-up column (wpu) is merged with the interrupt column (int), then the I/O configuration is a pull-up interrupt input, otherwise the configuration is a floating interrupt input. Port C is mapped to ei0 or ei1 by option byte.

2. In the open drain output column, “T” defines a true open drain I/O (P-Buffer and protection diode to V_{DD} are not implemented). See Section 9 "I/O PORTS" on page 38 for more details.

3. OSC1 and OSC2 pins connect a crystal or ceramic resonator, or an external source to the on-chip oscillator see Section 2 "PIN DESCRIPTION" on page 6 and Section 6.2 "MULTI-OSCILLATOR (MO)" on page 21 for more details.

4: For details refer to Section 13.8 on page 144

3 REGISTER & MEMORY MAP

As shown in Figure 5, the MCU is capable of addressing 64K bytes of memories and I/O registers.

The available memory locations consist of 128 bytes of register location, 256 bytes of RAM and up to 8 Kbytes of user program memory. The RAM space includes up to 128 bytes for the stack from 0100h to 017Fh.

The highest address bytes contain the user reset and interrupt vectors.

The Flash memory contains two sectors (see Figure 5) mapped in the upper part of the ST7 addressing space so the reset and interrupt vectors are located in Sector 0 (F000h-FFFFh).

The size of Flash Sector 0 and other device options are configurable by Option byte (refer to Section 15.1 on page 162).

IMPORTANT: Memory locations marked as "Reserved" must never be accessed. Accessing a reseved area can have unpredictable effects on the device.

Related Documentation

AN 985: Executing Code in ST7 RAM

Figure 5. Memory Map

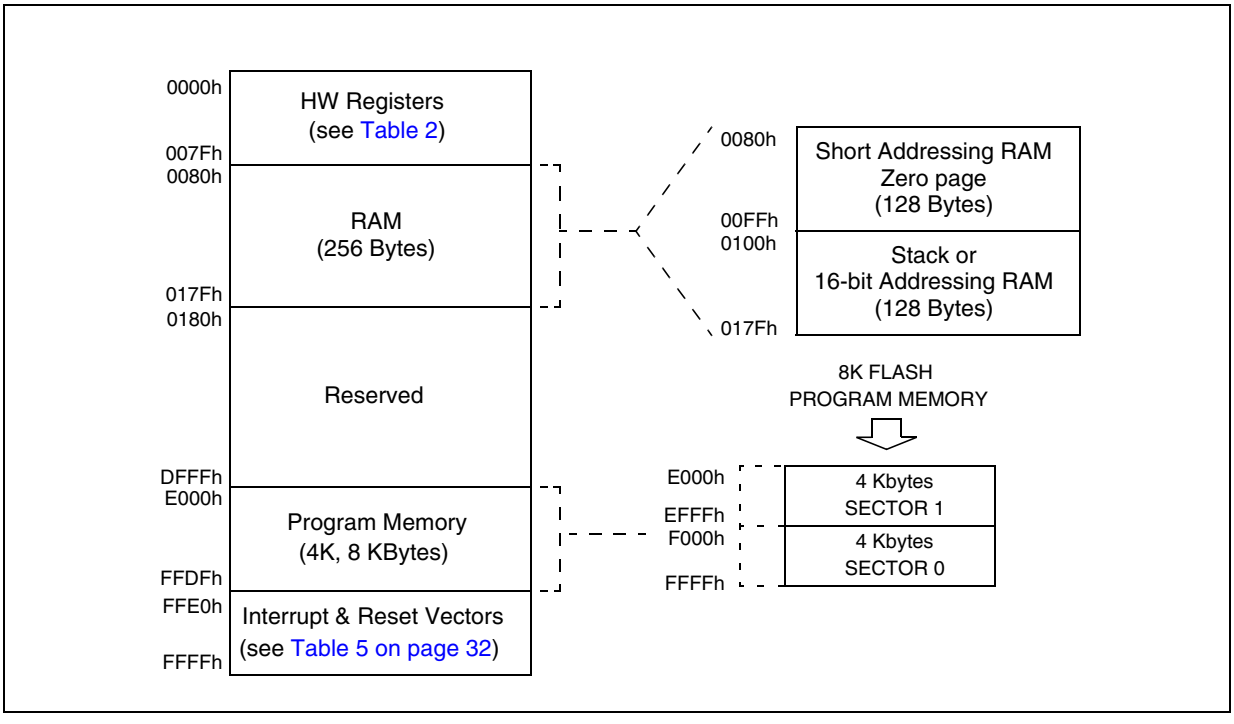


Table 2. Hardware Register Map

Address	Block	Register Label	Register Name	Reset Status	Remarks
0000h 0001h 0002h	Port C	PCDR PCDDR PCOR	Port C Data Register Port C Data Direction Register Port C Option Register	xx000000h[1] 00h 00h	R/W [2] R/W [2] R/W [2]
0003h	Reserved (1 Byte)				
0004h 0005h 0006h	Port B	PBDR PBDDR PBOR	Port B Data Register Port B Data Direction Register Port B Option Register	00h [1] 00h 00h	R/W R/W R/W.
0007h	Reserved (1 Byte)				
0008h 0009h 000Ah	Port A	PADR PADDR PAOR	Port A Data Register Port A Data Direction Register Port A Option Register	00h [1] 00h 00h	R/W R/W R/W
000Bh to 001Bh	Reserved (17 Bytes)				
001Ch 001Dh 001Eh 001Fh	ITC	ISPR0 ISPR1 ISPR2 ISPR3	Interrupt software priority register0 Interrupt software priority register1 Interrupt software priority register2 Interrupt software priority register3	FFh FFh FFh FFh	R/W R/W R/W R/W
0020h		MISCR1	Miscellanous register 1	00h	R/W
0021h 0022h 0023h	SPI	SPIDR SPICR SPICSR	SPI Data I/O Register SPI Control Register SPI Status Register	xxh 0xh 00h	R/W R/W R/W
0024h	WATCHDOG	WDGCR	Watchdog Control Register	7Fh	R/W
0025h		SICSR	System Integrity Control / Status Register	000x 000x	R/W
0026h	MCC	MCCSR	Main Clock Control / Status Register	00h	R/W
0027h	Reserved (1 Byte)				
0028h 0029h 002Ah 002Bh 002Ch 002Dh 002Eh	I^2C	I2CCR I2CSR1 I2CSR2 I2CCCR I2COAR1 I2COAR2 I2CDR	I^2C Control Register I^2C Status Register 1 I^2C Status Register 2 I^2C Clock Control Register I^2C Own Address Register 1 I^2C Own Address Register2 I^2C Data Register	00h 00h 00h 00h 00h 40h 00h	R/W Read Only Read Only R/W R/W R/W R/W
002Fh 0030h	Reserved (2 Bytes)				

Address	Block	Register Label	Register Name	Reset Status	Remarks
0031h	TIMER A	TACR2	Timer A Control Register 2	00h	R/W
0032h		TACR1	Timer A Control Register 1	00h	R/W
0033h		TASCSR	Timer A Control/Status Register	xxh	R/W
0034h		TAIC1HR	Timer A Input Capture 1 High Register	xxh	Read Only
0035h		TAIC1LR	Timer A Input Capture 1 Low Register	xxh	Read Only
0036h		TAOC1HR	Timer A Output Compare 1 High Register	80h	R/W
0037h		TAOC1LR	Timer A Output Compare 1 Low Register	00h	R/W
0038h		TACHR	Timer A Counter High Register	FFh	Read Only
0039h		TACLR	Timer A Counter Low Register	FCh	Read Only
003Ah		TAACHR	Timer A Alternate Counter High Register	FFh	Read Only
003Bh		TAACLR	Timer A Alternate Counter Low Register	FCh	Read Only
003Ch		TAIC2HR	Timer A Input Capture 2 High Register	xxh	Read Only
003Dh		TAIC2LR	Timer A Input Capture 2 Low Register	xxh	Read Only
003Eh		TAOC2HR	Timer A Output Compare 2 High Register	80h	R/W
003Fh		TAOC2LR	Timer A Output Compare 2 Low Register	00h	R/W
0040h		MISCR2	Miscellanous register 2	00h	R/W
0041h	TIMER B	TBCR2	Timer B Control Register 2	00h	R/W
0042h		TBCR1	Timer B Control Register 1	00h	R/W
0043h		TBSCSR	Timer B Control/Status Register	xxh	R/W
0044h		TBIC1HR	Timer B Input Capture 1 High Register	xxh	Read Only
0045h		TBIC1LR	Timer B Input Capture 1 Low Register	xxh	Read Only
0046h		TBOC1HR	Timer B Output Compare 1 High Register	80h	R/W
0047h		TBOC1LR	Timer B Output Compare 1 Low Register	00h	R/W
0048h		TBCHR	Timer B Counter High Register	FFh	Read Only
0049h		TBCLR	Timer B Counter Low Register	FCh	Read Only
004Ah		TBACHR	Timer B Alternate Counter High Register	FFh	Read Only
004Bh		TBACLR	Timer B Alternate Counter Low Register	FCh	Read Only
004Ch		TBIC2HR	Timer B Input Capture 2 High Register	xxh	Read Only
004Dh		TBIC2LR	Timer B Input Capture 2 Low Register	xxh	Read Only
004Eh		TBOC2HR	Timer B Output Compare 2 High Register	80h	R/W
004Fh		TBOC2LR	Timer B Output Compare 2 Low Register	00h	R/W
0050h	SCI	SCISR	SCI Status Register	C0h	Read Only
0051h		SCIDR	SCI Data Register	xxh	R/W
0052h		SCIBRR	SCI Baud Rate Register	00h	R/W
0053h		SCICR1	SCI Control Register1	x000 0000h	R/W
0054h		SCICR2	SCI Control Register2	00h	R/W
0055h		SCIERPR	SCI Extended Receive Prescaler Register	00h	R/W
0056h		SCIETPR	SCI Extended Transmit Prescaler Register	00h	R/W
0057h to 006Eh	Reserved (24 Bytes)				
006Fh	ADC	ADCDRL	Data Register Low[3)]	00h	Read Only
0070h		ADCDRH	Data Register High[3)]	00h	Read Only
0071h		ADCCSR	Control/Status Register	00h	R/W
0072h	FLASH	FCSR	Flash Control Register	00h	R/W
0073h to 007Fh	Reserved (13 Bytes)				

Legend: x=Undefined, R/W=Read/Write

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 keep their reset value.

3. For compatibility with the ST72C254, the ADCDRL and ADCDRH data registers are located with the LSB on the lower address (6Fh) and the MSB on the higher address (70h). As this scheme is not little Endian, the ADC data registers cannot be treated by C programs as an integer, but have to be treated as two char registers.

4 FLASH PROGRAM MEMORY

4.1 Introduction

The ST7 single voltage extended Flash (XFlash) is a non-volatile memory that can be electrically erased and programmed either on a byte-by-byte basis or up to 32 bytes in parallel.

The XFlash devices can be programmed off-board (plugged in a programming tool) or on-board using In-Circuit Programming or In-Application Programming.

The array matrix organisation allows each sector to be erased and reprogrammed without affecting other sectors.

4.2 Main Features

- ICP (In-Circuit Programming)
- IAP (In-Application Programming)
- ICT (In-Circuit Testing) for downloading and executing user application test patterns in RAM
- Sector 0 size configurable by option byte
- Read-out and write protection against piracy

4.3 PROGRAMMING MODES

The ST7 can be programmed in three different ways:

– Insertion in a programming tool. In this mode, FLASH sectors 0 and 1 and option byte row can be programmed or erased.

– In-Circuit Programming. In this mode, FLASH sectors 0 and 1 and option byte row can be programmed or erased without removing the device from the application board.

– In-Application Programming. In this mode, sector 1 can be programmed or erased without removing the device from the application board and while the application is running.

4.3.1 In-Circuit Programming (ICP)

ICP uses a protocol called ICC (In-Circuit Communication) which allows an ST7 plugged on a printed circuit board (PCB) to communicate with an external programming device connected via cable. ICP is performed in three steps:

Switch the ST7 to ICC mode (In-Circuit Communications). This is done by driving a specific signal sequence on the ICCCLK/DATA pins while the RESET pin is pulled low. When the ST7 enters ICC mode, it fetches a specific RESET vector which points to the ST7 System Memory containing the ICC protocol routine. This routine enables the ST7 to receive bytes from the ICC interface.

– Download ICP Driver code in RAM from the ICCDATA pin

– Execute ICP Driver code in RAM to program the FLASH memory

Depending on the ICP Driver code downloaded in RAM, FLASH memory programming can be fully customized (number of bytes to program, program locations, or selection of the serial communication interface for downloading).

4.3.2 In Application Programming (IAP)

This mode uses an IAP Driver program previously programmed in Sector 0 by the user (in ICP mode).

This mode is fully controlled by user software. This allows it to be adapted to the user application, (user-defined strategy for entering programming mode, choice of communications protocol used to fetch the data to be stored etc.)
IAP mode can be used to program any memory areas except Sector 0, which is write/erase protected to allow recovery in case errors occur during the programming operation.

FLASH PROGRAM MEMORY (Cont'd)

4.4 ICC interface

ICP needs a minimum of 4 and up to 7 pins to be connected to the programming tool. These pins are:

- $\overline{\text{RESET}}$: device reset
- V_{SS}: device power supply ground
- ICCCLK: ICC output serial clock pin
- ICCDATA: ICC input serial data pin
- ICCSEL: ICC selection (not required on devices without ICCSEL pin)
- OSC1: main clock input for external source (not required on devices without OSC1/OSC2 pins)
- V_{DD}: application board power supply (optional, see Note 3)

Notes:

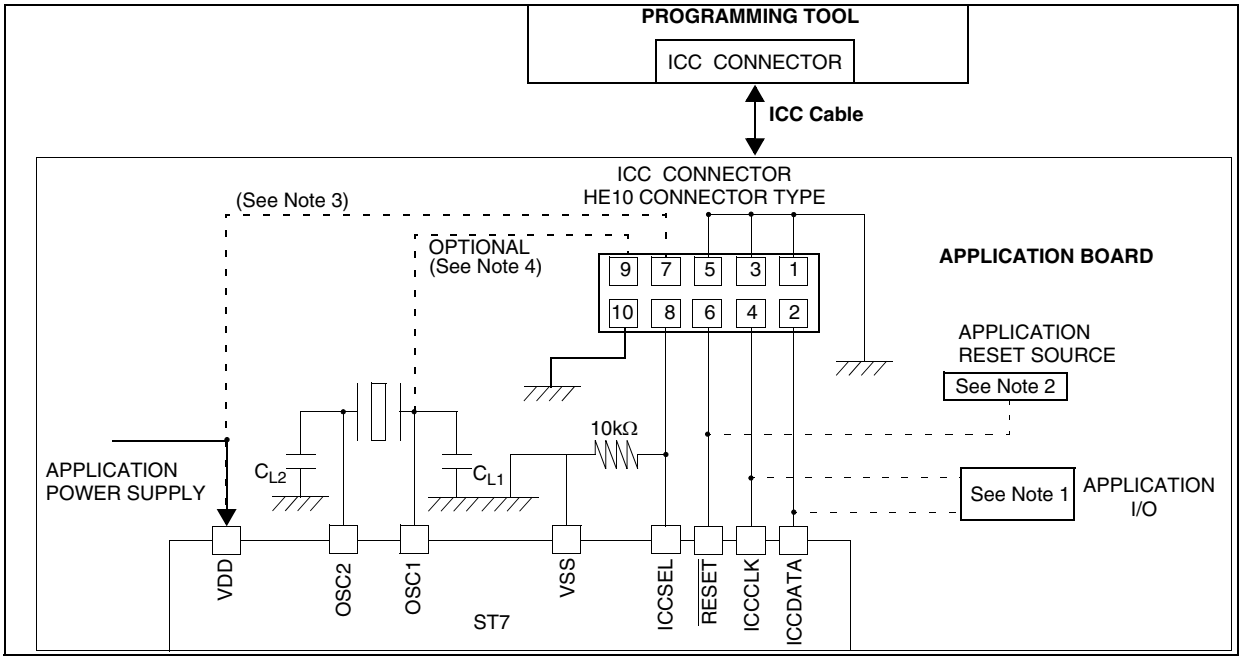
1. If the ICCCLK or ICCDATA pins are only used as outputs in the application, no signal isolation is necessary. As soon as the Programming Tool is plugged to the board, even if an ICC session is not in progress, the ICCCLK and ICCDATA pins are not available for the application. If they are used as inputs by the application, isolation such as a serial resistor has to be implemented in case another device forces the signal. Refer to the Programming Tool documentation for recommended resistor values.

2. During the ICP session, the programming tool must control the $\overline{\text{RESET}}$ pin. This can lead to conflicts between the programming tool and the application reset circuit if it drives more than 5mA at high level (push pull output or pull-up resistor<1K). A schottky diode can be used to isolate the application RESET circuit in this case. When using a classical RC network with R>1K or a reset management IC with open drain output and pull-up resistor>1K, no additional components are needed. In all cases the user must ensure that no external reset is generated by the application during the ICC session.

3. The use of Pin 7 of the ICC connector depends on the Programming Tool architecture. This pin must be connected when using most ST Programming Tools (it is used to monitor the application power supply). Please refer to the Programming Tool manual.

4. Pin 9 has to be connected to the OSC1 pin of the ST7 when the clock is not available in the application or if the selected clock option is not programmed in the option byte. ST7 devices with multi-oscillator capability need to have OSC2 grounded in this case.

Figure 6. Typical ICC Interface



FLASH PROGRAM MEMORY (Cont'd)

4.5 Memory Protection

There are two different types of memory protection: Read Out Protection and Write/Erase Protection which can be applied individually.

4.5.1 Read out Protection

Read-out protection, when selected, provides a protection against Program Memory content extraction and against write access to Flash memory. Even if no protection can be considered as totally unbreakable, the feature provides a very high level of protection for a general purpose microcontroller.

In flash devices, this protection is removed by reprogramming the option. In this case the program memory is automatically erased and the device can be reprogrammed.

Read-out protection selection depends on the device type:

– In Flash devices it is enabled and removed through the FMP_R bit in the option byte.

– In ROM devices it is enabled by mask option specified in the Option List.

4.5.2 Flash Write/Erase Protection

Write/erase protection, when set, makes it impossible to both overwrite and erase program memory. Its purpose is to provide advanced security to applications and prevent any change being made to the memory content.

Warning: Once set, Write/erase protection can never be removed. A write-protected flash device is no longer reprogrammable.

Write/erase protection is enabled through the FMP_W bit in the option byte.

4.6 Related Documentation

For details on Flash programming and ICC protocol, refer to the ST7 Flash Programming Reference Manual and to the ST7 ICC Protocol Reference Manual.

AN1477: Emulated data EEPROM with XFlash memory

AN1576: IAP drivers for ST7 HDFlash or XFlash MCUs

AN1575: On Board Programming methods for ST7 HDFlash or XFlash MCUs

AN1070: Checksum self checking capability

4.7 Register Description

FLASH CONTROL/STATUS REGISTER (FCSR)
Read/Write
Reset Value: 000 0000 (00h)
1st RASS Key: 0101 0110 (56h)
2nd RASS Key: 1010 1110 (AEh)

7							0
0	0	0	0	0	OPT	LAT	PGM

Note: This register is reserved for programming using ICP, IAP or other programming methods. It controls the XFlash programming and erasing operations.

When an EPB or another programming tool is used (in socket or ICP mode), the RASS keys are sent automatically.

5 CENTRAL PROCESSING UNIT

5.1 INTRODUCTION

This CPU has a full 8-bit architecture and contains six internal registers allowing efficient 8-bit data manipulation.

5.2 MAIN FEATURES

- Enable executing 63 basic instructions
- Fast 8-bit by 8-bit multiply
- 17 main addressing modes (with indirect addressing mode)
- Two 8-bit index registers
- 16-bit stack pointer
- Low power HALT and WAIT modes
- Priority maskable hardware interrupts
- Non-maskable software/hardware interrupts

5.3 CPU REGISTERS

The 6 CPU registers shown in Figure 7 are not present in the memory mapping and are accessed by specific instructions.

Accumulator (A)

The Accumulator is an 8-bit general purpose register used to hold operands and the results of the arithmetic and logic calculations and to manipulate data.

Index Registers (X and Y)

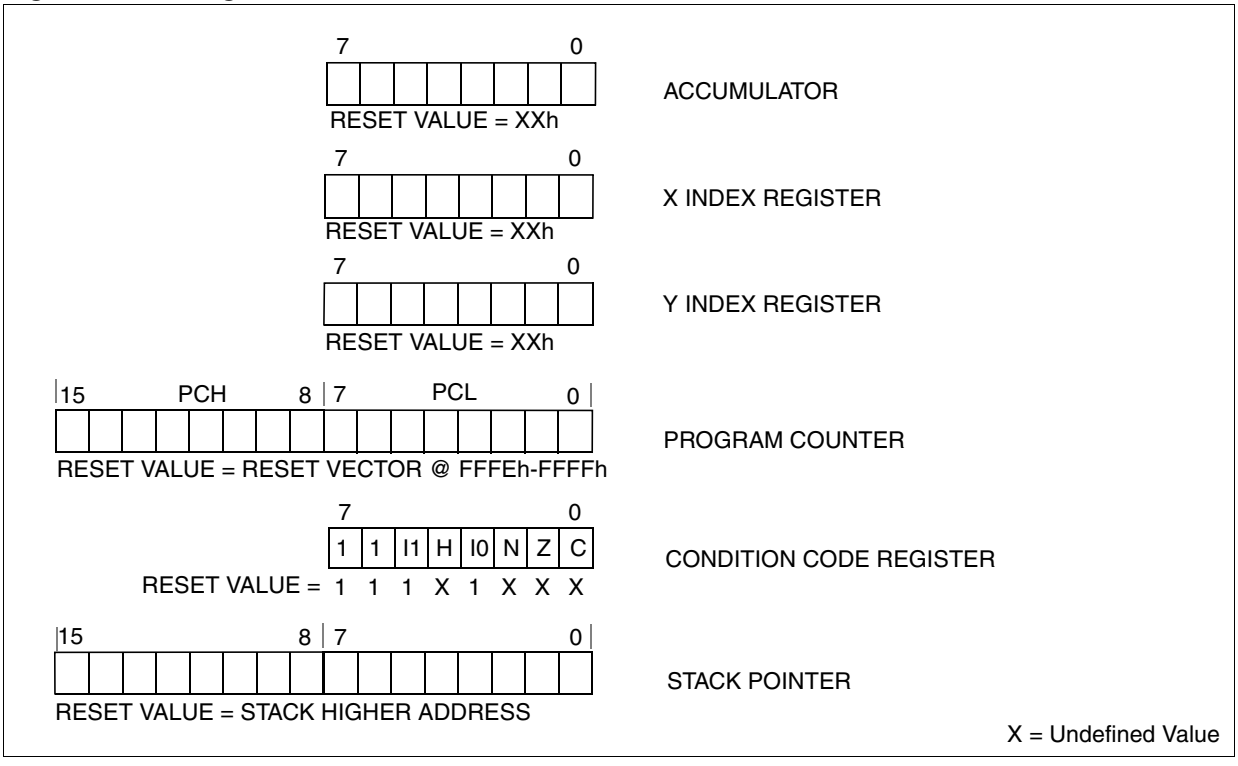
These 8-bit registers are used to create effective addresses or as temporary storage areas for data manipulation. (The Cross-Assembler generates a precede instruction (PRE) to indicate that the following instruction refers to the Y register.)

The Y register is not affected by the interrupt automatic procedures.

Program Counter (PC)

The program counter is a 16-bit register containing the address of the next instruction to be executed by the CPU. It is made of two 8-bit registers PCL (Program Counter Low which is the LSB) and PCH (Program Counter High which is the MSB).

Figure 7. CPU Registers



CENTRAL PROCESSING UNIT (Cont'd)

Condition Code Register (CC)

Read/Write

Reset Value: 111x1xxx

7							0
1	1	I1	H	I0	N	Z	C

The 8-bit Condition Code register contains the interrupt masks and four flags representative of the result of the instruction just executed. This register can also be handled by the PUSH and POP instructions.

These bits can be individually tested and/or controlled by specific instructions.

Arithmetic Management Bits

Bit 4 = **H** *Half carry*.

This bit is set by hardware when a carry occurs between bits 3 and 4 of the ALU during an ADD or ADC instructions. It is reset by hardware during the same instructions.

0: No half carry has occurred.
1: A half carry has occurred.

This bit is tested using the JRH or JRNH instruction. The H bit is useful in BCD arithmetic subroutines.

Bit 2 = **N** *Negative*.

This bit is set and cleared by hardware. It is representative of the result sign of the last arithmetic, logical or data manipulation. It's a copy of the result 7^{th} bit.
0: The result of the last operation is positive or null.
1: The result of the last operation is negative (i.e. the most significant bit is a logic 1).

This bit is accessed by the JRMI and JRPL instructions.

Bit 1 = **Z** *Zero*.

This bit is set and cleared by hardware. This bit indicates that the result of the last arithmetic, logical or data manipulation is zero.
0: The result of the last operation is different from zero.
1: The result of the last operation is zero.

This bit is accessed by the JREQ and JRNE test instructions.

Bit 0 = **C** *Carry/borrow.*

This bit is set and cleared by hardware and software. It indicates an overflow or an underflow has occurred during the last arithmetic operation.
0: No overflow or underflow has occurred.
1: An overflow or underflow has occurred.

This bit is driven by the SCF and RCF instructions and tested by the JRC and JRNC instructions. It is also affected by the "bit test and branch", shift and rotate instructions.

Interrupt Management Bits

Bit 5,3 = **I1, I0** *Interrupt*

The combination of the I1 and I0 bits gives the current interrupt software priority.

Interrupt Software Priority	I1	I0
Level 0 (main)	1	0
Level 1	0	1
Level 2	0	0
Level 3 (= interrupt disable)	1	1

These two bits are set/cleared by hardware when entering in interrupt. The loaded value is given by the corresponding bits in the interrupt software priority registers (IxSPR). They can be also set/cleared by software with the RIM, SIM, IRET, HALT, WFI and PUSH/POP instructions.

See the interrupt management chapter for more details.

CENTRAL PROCESSING UNIT (Cont'd)

Stack Pointer (SP)

Read/Write

Reset Value: 01 7Fh

15							8
0	0	0	0	0	0	0	1

7							0
SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0

The Stack Pointer is a 16-bit register which is always pointing to the next free location in the stack. It is then decremented after data has been pushed onto the stack and incremented before data is popped from the stack (see Figure 8).

Since the stack is 128 bytes deep, the 8 most significant bits are forced by hardware. Following an MCU Reset, or after a Reset Stack Pointer instruction (RSP), the Stack Pointer contains its reset value (the SP7 to SP0 bits are set) which is the stack higher address.

The least significant byte of the Stack Pointer (called S) can be directly accessed by a LD instruction.

Note: When the lower limit is exceeded, the Stack Pointer wraps around to the stack upper limit, without indicating the stack overflow. The previously stored information is then overwritten and therefore lost. The stack also wraps in case of an underflow.

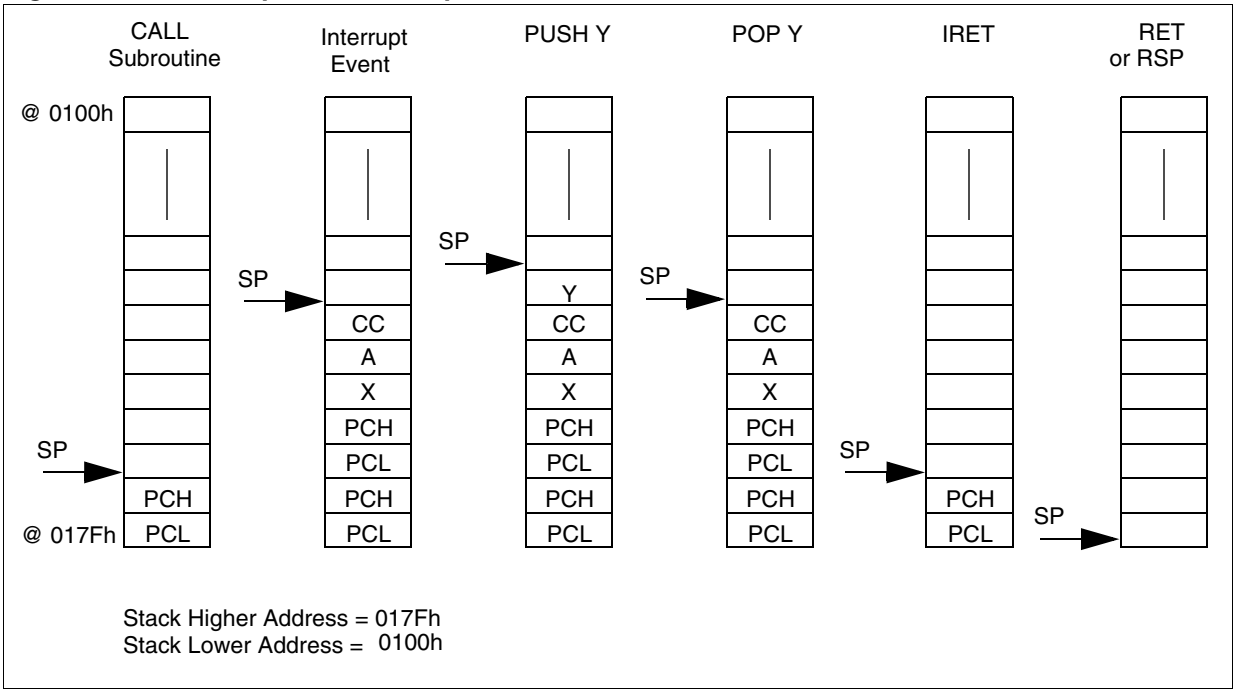
The stack is used to save the return address during a subroutine call and the CPU context during an interrupt. The user may also directly manipulate the stack by means of the PUSH and POP instructions. In the case of an interrupt, the PCL is stored at the first location pointed to by the SP. Then the other registers are stored in the next locations as shown in Figure 8

– When an interrupt is received, the SP is decremented and the context is pushed on the stack.

– On return from interrupt, the SP is incremented and the context is popped from the stack.

A subroutine call occupies two locations and an interrupt five locations in the stack area.

Figure 8. Stack Manipulation Example



6 SUPPLY, RESET AND CLOCK MANAGEMENT

The device includes a range of utility features for securing the application in critical situations (for example in case of a power brown-out), and reducing the number of external components. An overview is shown in Figure 10.

For more details, refer to dedicated parametric section.

Main Features

- Optional PLL for multiplying the frequency by 2 (not to be used with internal RC oscillator)
- Reset Sequence Manager (RSM)
- Multi-Oscillator Clock Management (MO)
  - 4 Crystal/Ceramic resonator oscillators
  - 1 Internal RC oscillator
- System Integrity Management (SI)
  - Main supply Low Voltage Detector (LVD)
  - Auxiliary Voltage Detector (AVD) with interrupt capability for monitoring the main supply

6.1 PHASE LOCKED LOOP

If the clock frequency input to the PLL is in the 2 to 4 MHz range, the PLL can be used to multiply the frequency by two to obtain an f_{OSC2} of 4 to 8 MHz.

The PLL is enabled by option byte. If the PLL is disabled, then $f_{OSC2} = f_{OSC}/2$.

Caution: The PLL is not recommended for applications where timing accuracy is required. See "PLL Characteristics" on page 139.

Figure 9. PLL Block Diagram

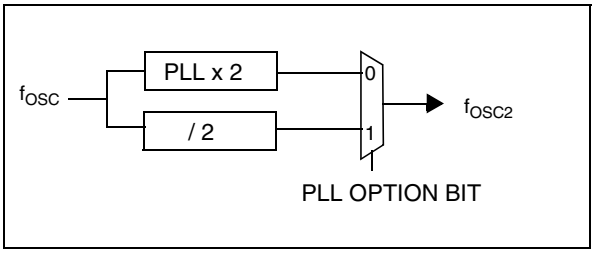
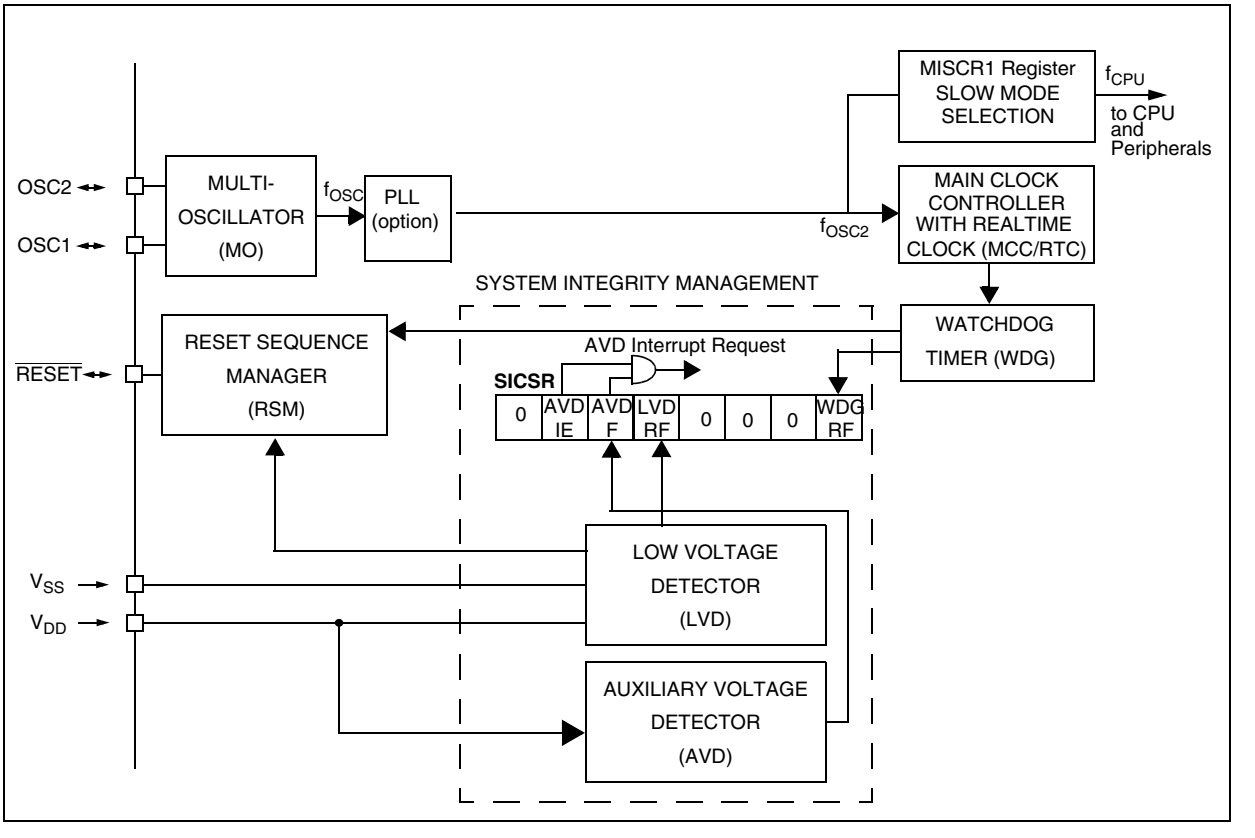


Figure 10. Clock, Reset and Supply Block Diagram



6.2 MULTI-OSCILLATOR (MO)

The main clock of the ST7 can be generated by four different source types coming from the multi-oscillator block:

- an external source
- 5 crystal or ceramic resonator oscillators
- an internal high frequency RC oscillator

Each oscillator is optimized for a given frequency range in terms of consumption and is selectable through the option byte. The associated hardware configurations are shown in Table 3. Refer to the electrical characteristics section for more details.

Caution: The OSC1 and/or OSC2 pins must not be left unconnected. For the purposes of Failure Mode and Effects Analysis, it should be noted that if the OSC1 and/or OSC2 pins are left unconnected, the ST7 main oscillator may start and, in this configuration, could generate an f_{OSC} clock frequency in excess of the allowed maximum (>16MHz.), putting the ST7 in an unsafe/undefined state. The product behaviour must therefore be considered undefined when the OSC pins are left unconnected.

External Clock Source

In this external clock mode, a clock signal (square, sinus or triangle) with ~50% duty cycle has to drive the OSC1 pin while the OSC2 pin is tied to ground.

Crystal/Ceramic Oscillators

This family of oscillators has the advantage of producing a very accurate rate on the main clock of the ST7. The selection within a list of 5 oscillators with different frequency ranges has to be done by option byte in order to reduce consumption (refer to Section 15.1 on page 162 for more details on the frequency ranges). In this mode of the multi-oscillator, the resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and start-up stabilization time. The loading capacitance values must be adjusted according to the selected oscillator.

These oscillators are not stopped during the RESET phase to avoid losing time in the oscillator start-up phase.

Internal RC Oscillator

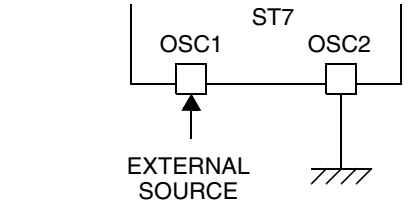
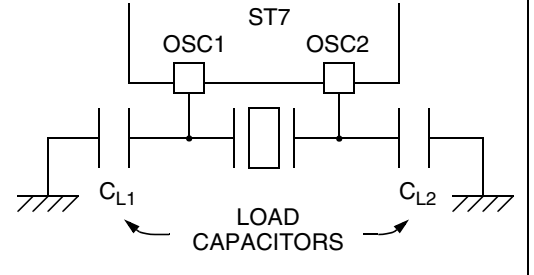
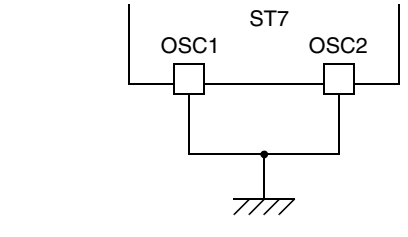
This oscillator allows a low cost solution for the main clock of the ST7 using only an internal resistor and capacitor. Internal RC oscillator mode has the drawback of a lower frequency accuracy and should not be used in applications that require accurate timing.

In this mode, the two oscillator pins have to be tied to ground.

Related documentation

AN1530: Accurate timebase for low cost ST7 applications with internal RC.

Table 3. ST7 Clock Sources

	Hardware Configuration
External Clock	
Crystal/Ceramic Resonators	
Internal RC Oscillator	

6.3 RESET SEQUENCE MANAGER (RSM)

6.3.1 Introduction

The reset sequence manager includes three RESET sources as shown in Figure 12:

- External $\overline{\text{RESET}}$ source pulse
- Internal LVD RESET (Low Voltage Detection)
- Internal WATCHDOG RESET

These sources act on the $\overline{\text{RESET}}$ pin and it is always kept low during the delay phase.

The RESET service routine vector is fixed at addresses FFFEh-FFFFh in the ST7 memory map.

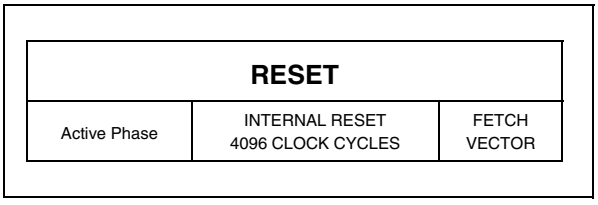
The basic RESET sequence consists of 3 phases as shown in Figure 11:

- Active Phase depending on the RESET source
- 4096 CPU clock cycle delay (selected by option byte)
- RESET vector fetch

The 4096 CPU clock cycle delay allows the oscillator to stabilise and ensures that recovery has taken place from the Reset state. The shorter or longer clock cycle delay should be selected by option byte to correspond to the stabilization time of the external oscillator used in the application.

The RESET vector fetch phase duration is 2 clock cycles.

Figure 11. RESET Sequence Phases

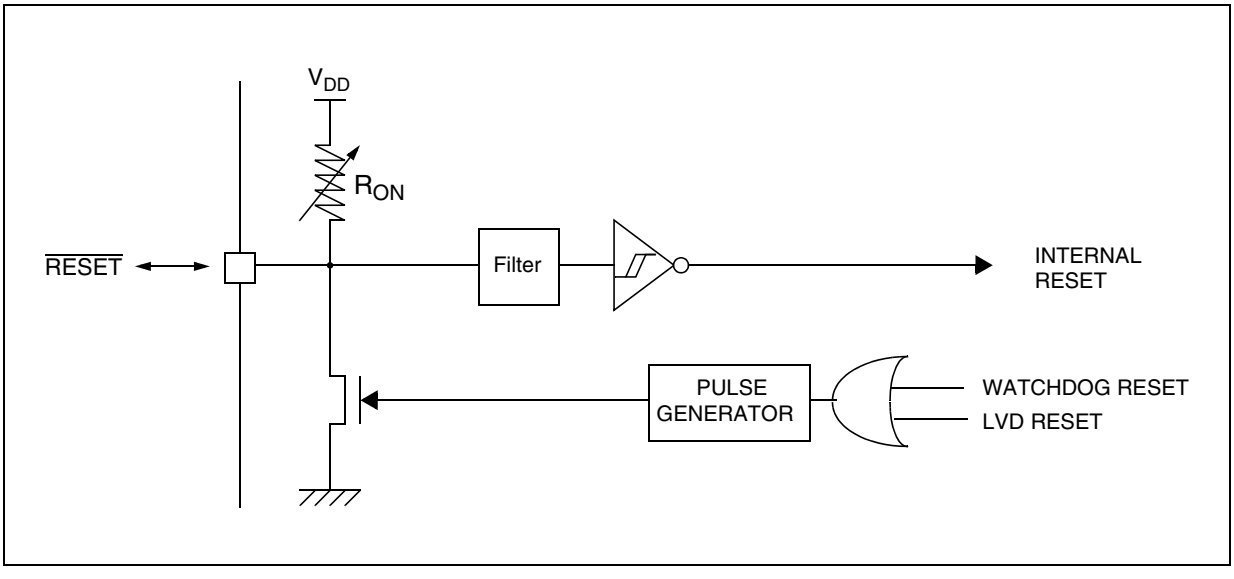


6.3.2 Asynchronous External $\overline{\text{RESET}}$ pin

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

A RESET signal originating from an external source must have a duration of at least $t_{h(RSTL)in}$ in order to be recognized (see Figure 13). This detection is asynchronous and therefore the MCU can enter reset state even in HALT mode.

Figure 12. Reset Block Diagram



RESET SEQUENCE MANAGER (Cont'd)

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

6.3.3 External Power-On RESET

If the LVD is disabled by option byte, to start up the microcontroller correctly, the user must ensure by means of an external reset circuit that the reset signal is held low until V_{DD} is over the minimum level specified for the selected f_{OSC} frequency.

A proper reset signal for a slow rising V_{DD} supply can generally be provided by an external RC network connected to the $\overline{\text{RESET}}$ pin.

6.3.4 Internal Low Voltage Detector (LVD) RESET

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

- Power-On RESET
- Voltage Drop RESET

The device $\overline{\text{RESET}}$ pin acts as an output that is pulled low when $V_{DD}<V_{IT+}$ (rising edge) or $V_{DD}<V_{IT-}$ (falling edge) as shown in Figure 13.

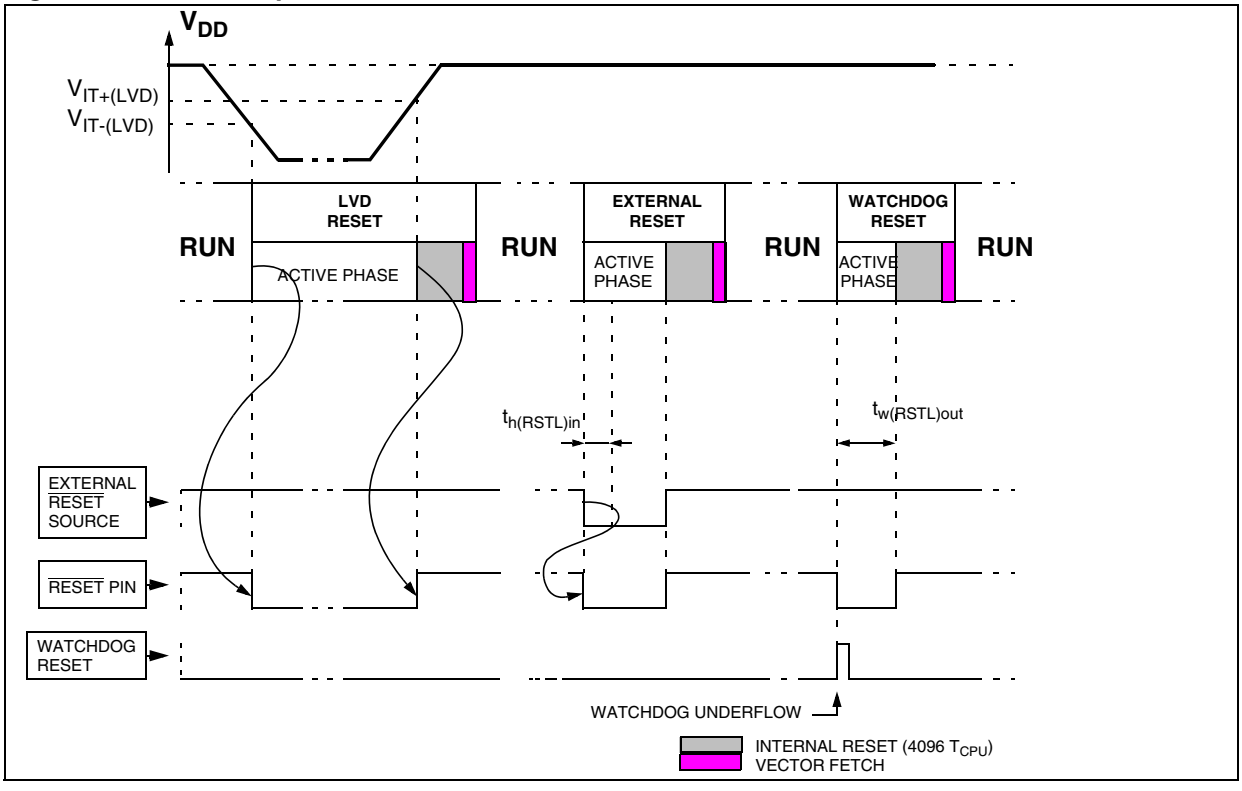
The LVD filters spikes on V_{DD} larger than $t_{g(VDD)}$ to avoid parasitic resets.

6.3.5 Internal Watchdog RESET

The RESET sequence generated by a internal Watchdog counter overflow is shown in Figure 13.

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

Figure 13. RESET Sequences



6.4 SYSTEM INTEGRITY MANAGEMENT (SI)

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

Note: A reset can also be triggered following the detection of an illegal opcode or prebyte code. Refer to Section 12.2.1 on page 123 for further details.

6.4.1 Low Voltage Detector (LVD)

The Low Voltage Detector function (LVD) generates a static reset when the V_{DD} supply voltage is below a V_{IT-} reference value. This means that it secures the power-up as well as the power-down keeping the ST7 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 14.

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

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

– under full software control
– in static safe reset

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

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

Notes:

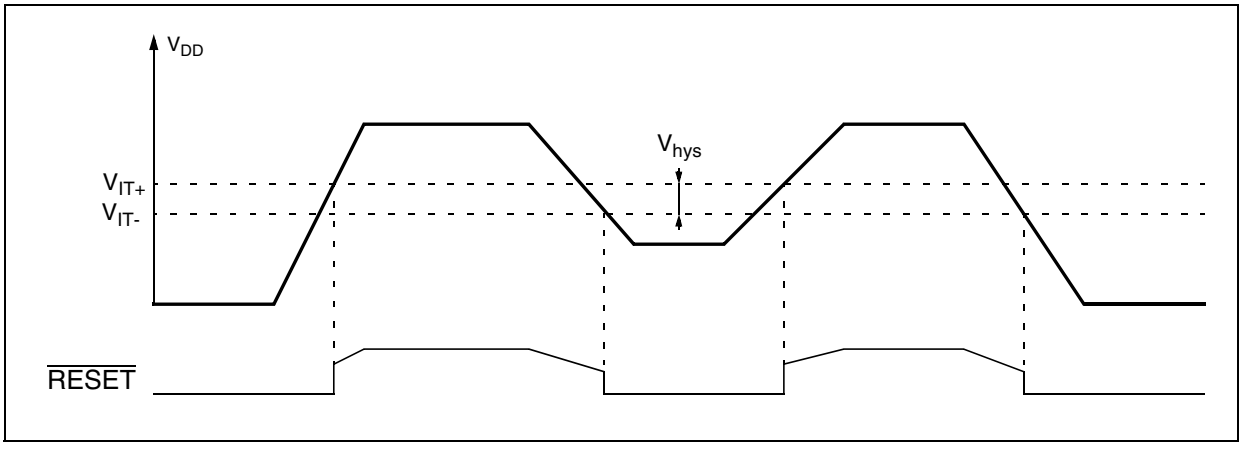
The LVD allows the device to be used without any external RESET circuitry.

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

Use of LVD with capacitive power supply: with this type of power supply, if power cuts occur in the application, it is recommended to pull V_{DD} down to 0V to ensure optimum restart conditions. Refer to circuit example in Figure 91 on page 151 and note 6.

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

Figure 14. Low Voltage Detector vs Reset



SYSTEM INTEGRITY MANAGEMENT (Cont'd)

6.4.2 Auxiliary Voltage Detector (AVD)

The Voltage Detector function (AVD) is based on an analog comparison between a V_{IT-} and V_{IT+} reference value and the V_{DD} main supply. The V_{IT-} reference value for falling voltage is lower than the V_{IT+} reference value for rising voltage in order to avoid parasitic detection (hysteresis).

The output of the AVD comparator is directly readable by the application software through a real time status bit (VDF) in the SICSR register. This bit is read only.

Caution: The AVD functions only if the LVD is enabled through the option byte.

6.4.2.1 Monitoring the V_{DD} Main Supply

The AVD voltage threshold value is relative to the selected LVD threshold configured by option byte (see Section 15.1 on page 162).

If the AVD interrupt is enabled, an interrupt is generated when the voltage crosses the $V_{IT+(AVD)}$ or $V_{IT-(AVD)}$ threshold (AVDF bit toggles).

In the case of a drop in voltage, the AVD interrupt acts as an early warning, allowing software to shut down safely before the LVD resets the microcontroller. See Figure 15.

The interrupt on the rising edge is used to inform the application that the V_{DD} warning state is over.

If the voltage rise time t_{rv} is less than 256 or 4096 CPU cycles (depending on the reset delay selected by option byte), no AVD interrupt will be generated when $V_{IT+(AVD)}$ is reached.

If t_{rv} is greater than 256 or 4096 cycles then:

– If the AVD interrupt is enabled before the $V_{IT+(AVD)}$ threshold is reached, then 2 AVD interrupts will be received: the first when the AVDIE bit is set, and the second when the threshold is reached.

– If the AVD interrupt is enabled after the $V_{IT+(AVD)}$ threshold is reached then only one AVD interrupt will occur.

Figure 15. Using the AVD to Monitor V_{DD}