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

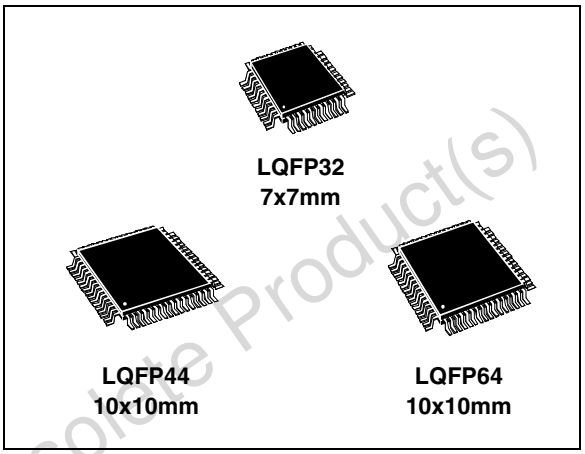
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

ST72361

8-bit MCU with Flash or ROM, 10-bit ADC, 5 timers, SPI, 2x LINSCI™

Features

- **Memories**
  - 16K to 60K High Density Flash (HDFlash) or ROM with read-out protection capability. In-Application Programming and In-Circuit Programming for HDFlash devices
  - 1.5 to 2K RAM
  - HDFlash endurance: 100 cycles, data retention 40 years at 85°C
- **Clock, Reset and Supply Management**
  - Low power crystal/ceramic resonator oscillators and bypass for external clock
  - PLL for 2x frequency multiplication
  - 5 power saving modes: Halt, Auto Wake Up From Halt, Active Halt, Wait and Slow
- **Interrupt Management**
  - Nested interrupt controller
  - 14 interrupt vectors plus TRAP and RESET
  - TLI top level interrupt (on 64-pin devices)
  - Up to 21 external interrupt lines (on 4 vectors)
- **Up to 48 I/O Ports**
  - Up to 48 multifunctional bidirectional I/O lines
  - Up to 36 alternate function lines
  - Up to 6 high sink outputs
- **5 Timers**
  - 16-bit timer with 2 input captures, 2 output compares, external clock input, PWM and pulse generator modes
  - 8-bit timer with 1 or 2 input captures, 1 or 2 output compares, PWM and pulse generator modes
  - 8-bit PWM auto-reload timer with 1 or 2 input captures, 2 or 4 independent PWM output channels, output compare and time base interrupt, external clock with event detector



  - Main clock controller with real-time base and clock output
  - Window watchdog timer
- **Up to 3 Communications Interfaces**
  - SPI synchronous serial interface
  - Master/slave LINSCI™ asynchronous serial interface
  - Master-only LINSCI™ asynchronous serial interface
- **Analog Peripheral (Low Current Coupling)**
  - 10-bit A/D converter with up to 16 inputs
  - Up to 9 robust ports (low current coupling)
- **Instruction Set**
  - 8-bit data manipulation
  - 63 basic instructions
  - 17 main addressing modes
  - 8 x 8 unsigned multiply instruction
- **Development Tools**
  - Full hardware/software development package

Table 1. Device Summary

Features	ST72361AR9/ST72361J9/ ST72361K9	ST72361AR7/ST72361J7/ ST72361K7	ST72361AR6/ST72361J6/ ST72361K6
Program memory - bytes	60K	48K	32K
RAM (stack) - bytes	2K (256)	2K (256)	1.5K (256)
Operating Supply	4.5V to 5.5V		
CPU Frequency	External Resonator Osc. w/ PLLx2/8 MHz		
Max. Temp. Range	-40°C to +125°C		
Packages	LQFP64 10x10mm (AR), LQFP44 10x10mm (J), LQFP32 7x7mm (K)		

Rev. 4

Table of Contents

Table of Contents

1 DESCRIPTION

The ST72361 devices are members of the ST7 microcontroller family designed for mid-range applications with LIN (Local Interconnect Network) interface.

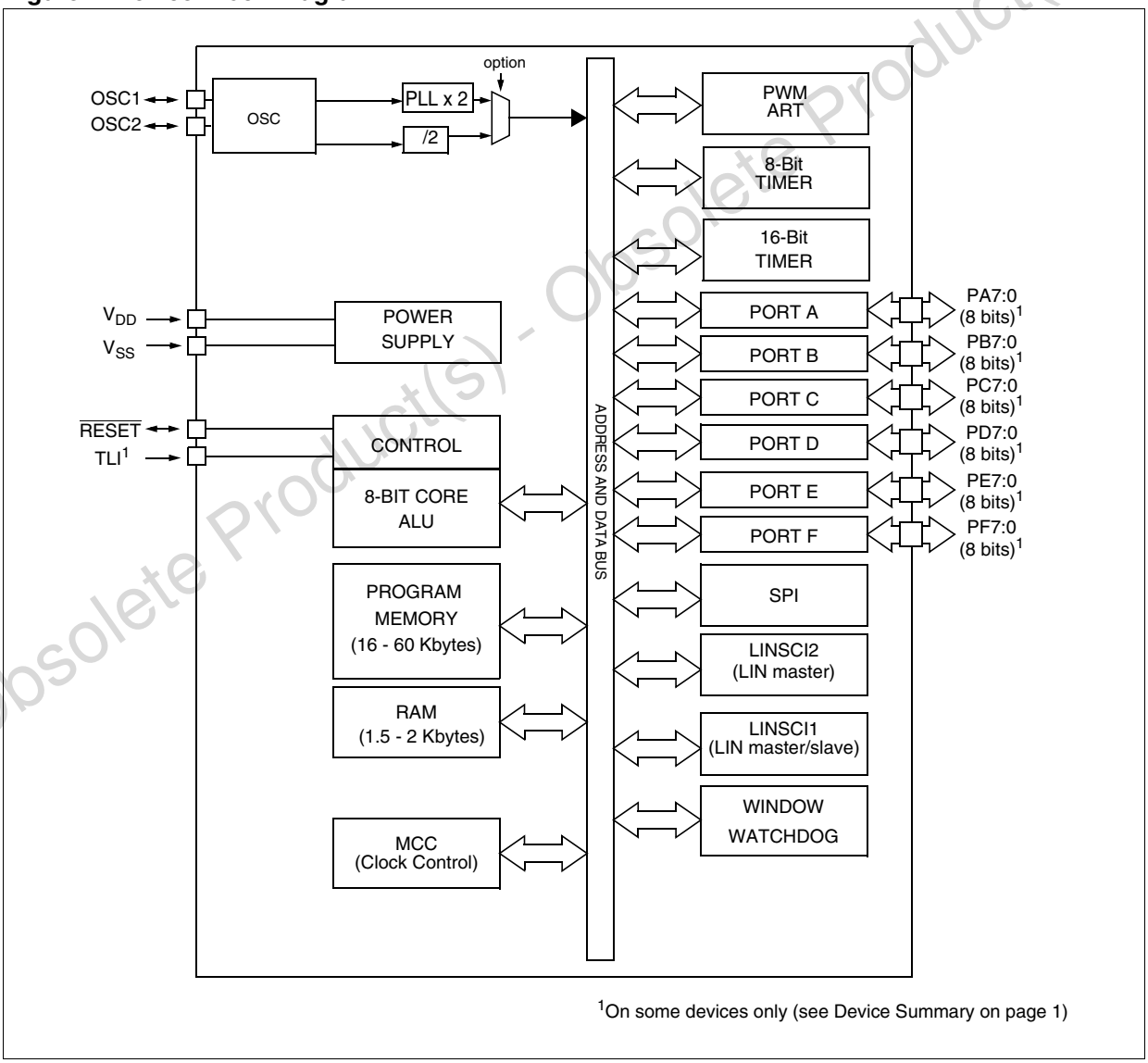
All devices are based on a common industry-standard 8-bit core, featuring an enhanced instruction set and are available with Flash or ROM program memory. The ST7 family architecture offers both power and flexibility to software developers, enabling the design of highly efficient and compact application code.

The on-chip peripherals include an A/D converter, a PWM Autoreload timer, 2 general purpose timers, 2 asynchronous serial interfaces, and an SPI interface.

For power economy, microcontroller can switch dynamically into WAIT, SLOW, Active-Halt, Auto Wake-up from HALT (AWU) or HALT mode when the application is in idle or stand-by state.

Typical applications are consumer, home, office and industrial products.

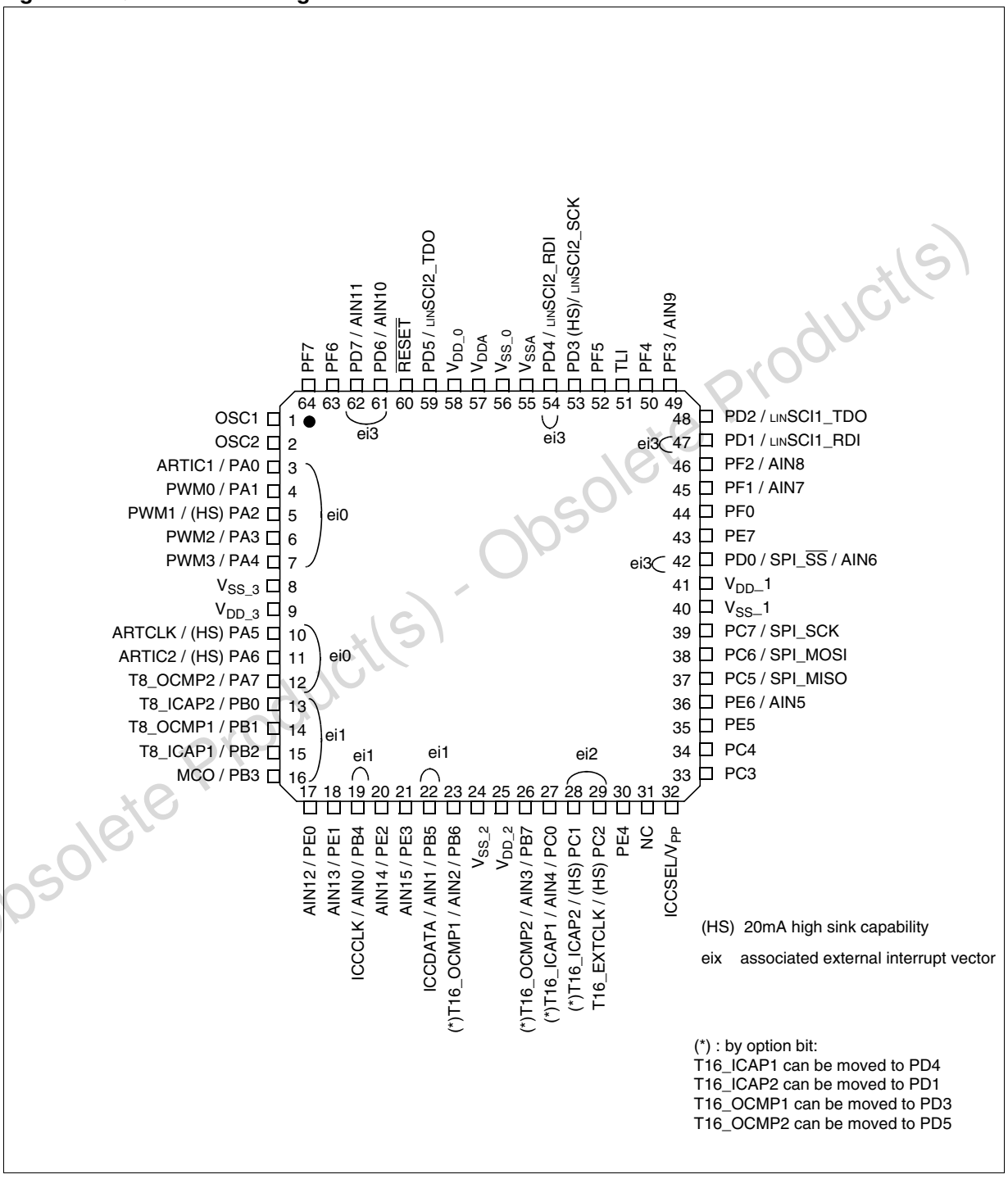
Figure 1. Device Block Diagram



[1]On some devices only (see Device Summary on page 1)

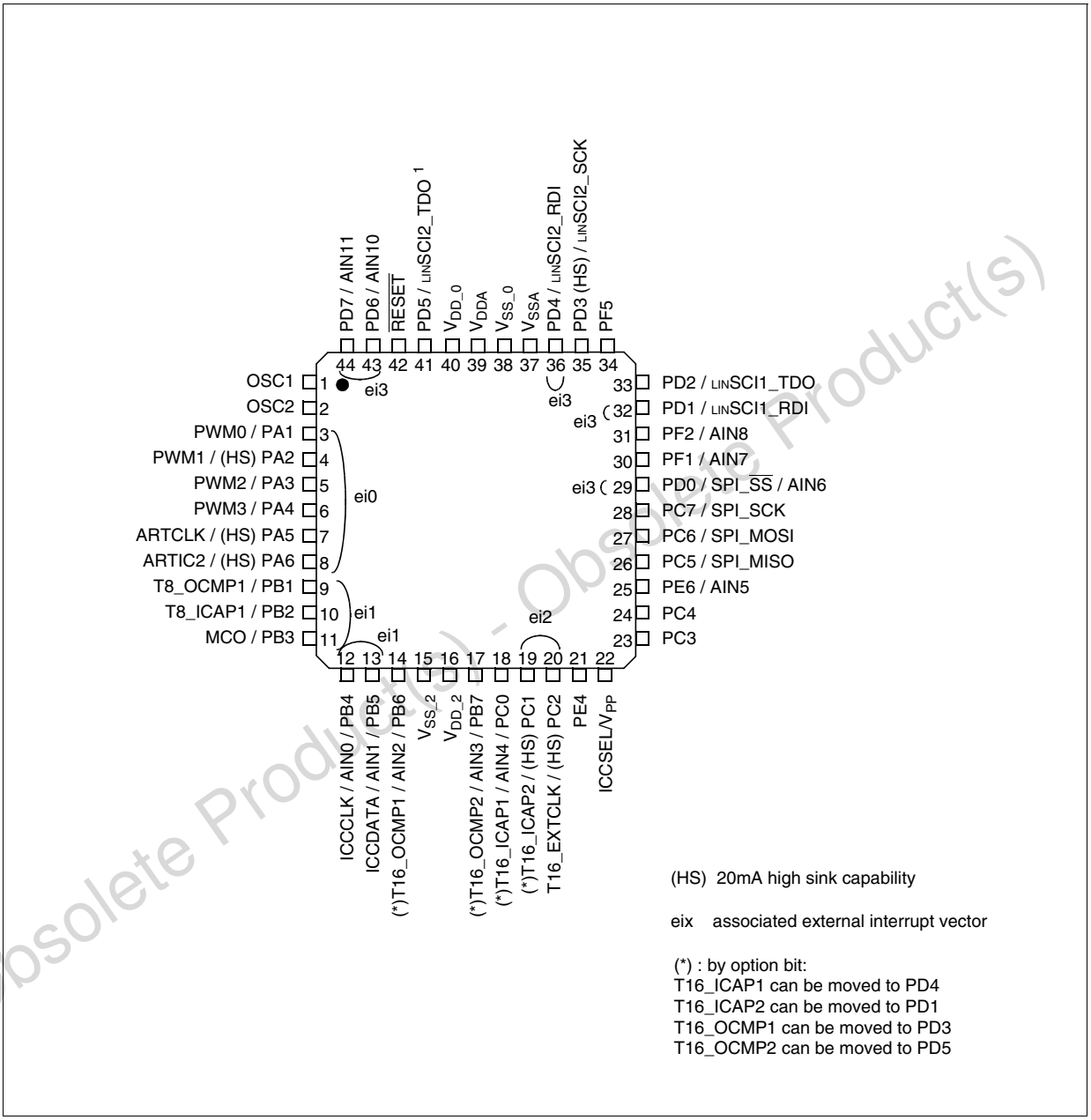
2 PIN DESCRIPTION

Figure 2. LQFP 64-Pin Package Pinout



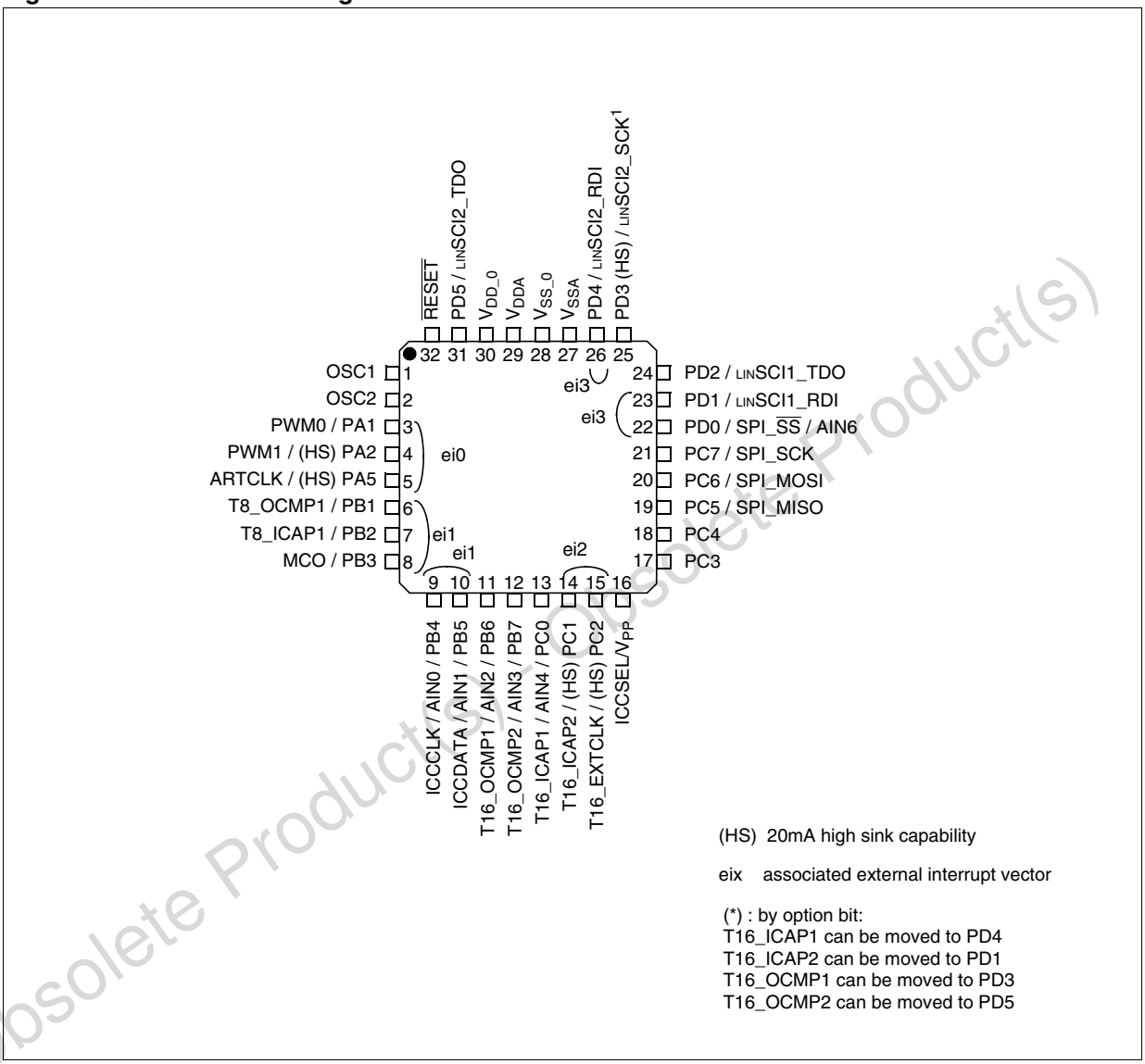
PIN DESCRIPTION (Cont'd)

Figure 3. LQFP 44-Pin Package Pinout



PIN DESCRIPTION (Cont'd)

Figure 4. LQFP 32-Pin Package Pinout



For external pin connection guidelines, refer to "ELECTRICAL CHARACTERISTICS" on page 178.

PIN DESCRIPTION (Cont'd)

For external pin connection guidelines, refer to "ELECTRICAL CHARACTERISTICS" on page 178.

Legend / Abbreviations for Table 2:

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

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

T_T= TTL 0.8V / 2V with Schmitt trigger

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

Port and control configuration:

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

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

Refer to "I/O PORTS" on page 45 for more details on the software configuration of the I/O ports.

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

Table 2. Device Pin Description

Pin n°			Pin Name	Type	Level		Port						Main function (after reset)	Alternate function	
							Input				Output				
LQFP64	LQFP44	LQFP32			Input	Output	float	wpu	int	ana	OD	PP			
1	1	1	OSC1[3)]	I									External clock input or Resonator oscillator inverter input		
2	2	2	OSC2[3)]	I/O									Resonator oscillator inverter output		
3	-	-	PA0 / ARTIC1	I/O	C_T		**X**		ei0		X	X	Port A0	ART Input Capture 1	
4	3	3	PA1 / PWM0	I/O	C_T		**X**			ei0	X	X	Port A1	ART PWM Output 0	
5	4	4	PA2 (HS) / PWM1	I/O	C_T	HS	**X**		ei0		X	X	Port A2	ART PWM Output 1	
6	5	-	PA3 / PWM2	I/O	C_T		**X**			ei0	X	X	Port A3	ART PWM Output 2	
7	6	-	PA4 / PWM3	I/O	C_T		**X**		ei0		X	X	Port A4	ART PWM Output 3	
8	-	-	V_{SS_3}	S									Digital Ground Voltage		
9	-	-	V_{DD_3}	S									Digital Main Supply Voltage		
10	7	5	PA5 (HS) / ARTCLK	I/O	C_T	HS	**X**			ei0	X	X	Port A5	ART External Clock	
11	8	-	PA6 (HS) / ARTIC2	I/O	C_T	HS	**X**		ei0		X	X	Port A6	ART Input Capture 2	
12	-	-	PA7 / T8_OCMP2	I/O	C_T		**X**			ei0	X	X	Port A7	TIM8 Output Compare 2	
13	-	-	PB0 /T8_ICAP2	I/O	C_T		**X**		ei1		X	X	Port B0	TIM8 Input Capture 2	
14	9	6	PB1 /T8_OCMP1	I/O	C_T		**X**			ei1	X	X	Port B1	TIM8 Output Compare 1	
15	10	7	PB2 / T8_ICAP1	I/O	C_T		**X**		ei1		X	X	Port B2	TIM8 Input Capture 1	
16	11	8	PB3 / MCO	I/O	C_T		**X**			ei1	X	X	Port B3	Main clock out (f_{OSC2})	
17	-	-	PE0 / AIN12	I/O	T_T		**X**	X		RB	X	X	Port E0	ADC Analog Input 12	
18	-	-	PE1 / AIN13	I/O	T_T		**X**	X		RB	X	X	Port E1	ADC Analog Input 13	
19	12	9	PB4 / AIN0 / ICCCLK	I/O	C_T		**X**		ei1	RB	X	X	Port B4	ICC Clock input	ADC Analog Input 0
20	-	-	PE2 / AIN14	I/O	T_T		**X**	X		RB	X	X	Port E2	ADC Analog Input 14	
21	-	-	PE3 / AIN15	I/O	T_T		**X**	X		RB	X	X	Port E3	ADC Analog Input 15	
22	13	10	PB5 / AIN1 / ICCDATA	I/O	C_T		**X**			ei1 RB	X	X	Port B5	ICC Data input	ADC Analog Input 1

Pin n°			Pin Name	Type	Level		Port						Main function (after reset)	Alternate function	
							Input				Output				
LQFP64	LQFP44	LQFP32			Input	Output	float	wpu	int	ana	OD	PP			
23	14	11	PB6 / AIN2 / T16_OCMP1	I/O	C_T		**X**	X		RB	X	X	Port B6	TIM16 Output Compare 1	ADC Analog Input 2
24	15	-	V_{SS_2}	S									Digital Ground Voltage		
25	16	-	V_{DD_2}	S									Digital Main Supply Voltage		
26	17	12	PB7 /AIN3 / T16_OCMP2	I/O	C_T		**X**	X		RB	X	X	Port B7	TIM16 Output Compare 2	ADC Analog Input 3
27	18	13	PC0 / AIN4 / T16_ICAP1	I/O	C_T		**X**	X		RB	X	X	Port C0	TIM16 Input Capture 1	ADC Analog Input 4
28	19	14	PC1 (HS) / T16_ICAP2	I/O	C_T	HS	**X**	ei2			X	X	Port C1	TIM16 Input Capture 2	
29	20	15	PC2 (HS) / T16_EXTCLK	I/O	C_T	HS	**X**		ei2		X	X	Port C2	TIM16 External Clock input	
30	21	-	PE4	I/O	T_T		**X**	X			X	X	Port E4		
31	-	-	NC	Not Connected											
32	22	16	V_{PP}	I									Flash programming voltage. Must be tied low in user mode.		
33	23	17	PC3	I/O	C_T		**X**	X			X	X	Port C3		
34	24	18	PC4	I/O	C_T		**X**					X[2]	Port C4		
35	-	-	PE5	I/O	T_T		**X**	X			X	X	Port E5		
36	25	-	PE6 / AIN5	I/O	T_T		**X**	X		X	X	X	Port E6	ADC Analog Input 5	
37	26	19	PC5 /MISO	I/O	C_T		**X**	X			X	X	Port C5	SPI Master In/Slave Out	
38	27	20	PC6 / MOSI	I/O	C_T		**X**	X			X	X	Port C6	SPI Master Out/Slave In	
39	28	21	PC7 /SCK	I/O	C_T		**X**	X			X	X	Port C7	SPI Serial Clock	
40	-	-	V_{SS_1}	S									Digital Ground Voltage		
41	-	-	V_{DD_1}	S									Digital Main Supply Voltage		
42	29	22	PD0 / $\overline{SS}$/ AIN6	I/O	C_T		**X**	ei3		X	X	X	Port D0	SPI Slave Select	ADC Analog Input 6
43	-	-	PE7	I/O	T_T		**X**	X			X	X	Port E7		
44	-	-	PF0	I/O	T_T		**X**	X			X	X	Port F0		
45	30	-	PF1 / AIN7	I/O	T_T		**X**	X		X	X	X	Port F1	ADC Analog Input 7	
46	31	-	PF2 / AIN8	I/O	T_T		**X**	X		X	X	X	Port F2	ADC Analog Input 8	
47	32	23	PD1 / SCI1_RDI	I/O	C_T		**X**		ei3		X	X	Port D1	LINSCI1 Receive Data input	
48	33	24	PD2 / SCI1_TDO	I/O	C_T		**X**	X			X	X	Port D2	LINSCI1 Transmit Data output	
49	-	-	PF3 / AIN9	I/O	T_T		**X**	X		X	X	X	Port F3	ADC Analog Input 9	
50	-	-	PF4	I/O	T_T		**X**	X			X	X	Port F4		
51	-	-	TLI	I	C_T		**X**		X				Top level interrupt input pin		
52	34	-	PF5	I/O	T_T		**X**	X			X	X	Port F5		
53	35	25	PD3 (HS) / SCI2_SCK	I/O	C_T	HS	**X**	X			X	X	Port D3	LINSCI2 Serial Clock Output	

Pin n°			Pin Name	Type	Level		Port						Main function (after reset)	Alternate function
							Input				Output			
LQFP64	LQFP44	LQFP32			Input	Output	float	wpu	int	ana	OD	PP		
54	36	26	PD4 / SCI2_RDI	I/O	C_T		**X**		ei3		X	X	Port D4	LINSCI2 Receive Data input
55	37	27	V_{SSA}	S									Analog Ground Voltage	
56	38	28	V_{SS_0}	S									Digital Ground Voltage	
57	39	29	V_{DDA}	I									Analog Reference Voltage for ADC	
58	40	30	V_{DD_0}	S									Digital Main Supply Voltage	
59	41	31	PD5 / SCI2_TDO	I/O	C_T		**X**	X			X	X	Port D5	LINSCI2 Transmit Data output
60	42	32	$\overline{\text{RESET}}$	I/O	C_T								Top priority non maskable interrupt.	
61	43	-	PD6 / AIN10	I/O	C_T		**X**	ei3		X	X	X	Port D6	ADC Analog Input 10
62	44	-	PD7 / AIN11	I/O	C_T		**X**		ei3	X	X	X	Port D7	ADC Analog Input 11
63	-	-	PF6	I/O	T_T		**X**	X			X	X	Port F6	
64	-	-	PF7	I/O	T_T		**X**	X			X	X	Port F7	

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 pull-up interrupt input, else the configuration is floating interrupt input.
2. Input mode can be used for general purpose I/O, output mode cannot be used.
3. OSC1 and OSC2 pins connect a crystal/ceramic resonator, or an external source to the on-chip oscillator; see Section 6 and Section 12.5 "CLOCK AND TIMING CHARACTERISTICS" for more details.
4. On the chip, each I/O port has eight pads. Pads that are not bonded to external pins are in input pull-up configuration after reset. The configuration of these pads must be kept at reset state to avoid added current consumption.

3 REGISTER AND MEMORY MAP

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

The available memory locations consist of 128 bytes of register locations, up to 2 Kbytes of RAM and up to 60 Kbytes of user program memory.

The RAM space includes up to 256 bytes for the stack from 0100h to 01FFh.The highest address bytes contain the user reset and interrupt vectors.

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

Figure 5. Memory Map

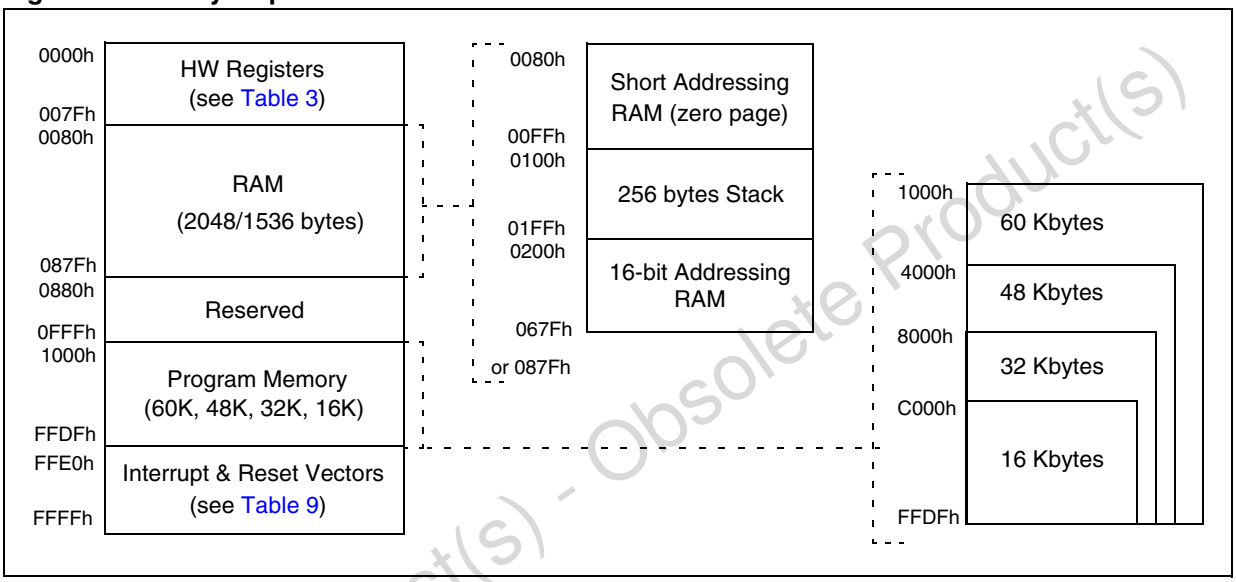


Table 3. Hardware Register Map

Address	Block	Register Label	Register Name	Reset Status	Remarks
0000h 0001h 0002h	Port A	PADR PADDR PAOR	Port A Data Register Port A Data Direction Register Port A Option Register	00h[1] 00h 00h	R/W[2] R/W[2] R/W[2]
0003h 0004h 0005h	Port B	PBDR PBDDR PBOR	Port B Data Register Port B Data Direction Register Port B Option Register	00h[1] 00h 00h	R/W[2] R/W[2] R/W[2]
0006h 0007h 0008h	Port C	PCDR PCDDR PCOR	Port C Data Register Port C Data Direction Register Port C Option Register	00h[1] 00h 00h	R/W[2] R/W[2] R/W[2]
0009h 000Ah 000Bh	Port D	PDDR PDDDR PDOR	Port D Data Register Port D Data Direction Register Port D Option Register	00h[1] 00h 00h	R/W[2] R/W[2] R/W[2]
000Ch 000Dh 000Eh	Port E	PEDR PEDDR PEOR	Port E Data Register Port E Data Direction Register Port E Option Register	00h[1] 00h 00h	R/W[2] R/W[2] R/W[2]

Address	Block	Register Label	Register Name	Reset Status	Remarks
000Fh 0010h 0011h	Port F	PFDR PFDDR PFOR	Port F Data Register Port F Data Direction Register Port F Option Register	00h[1] 00h 00h	R/W[2] R/W[2] R/W[2]
0012h to 0020h	Reserved Area (15 bytes)				
0021h 0022h 0023h	SPI	SPIDR SPICR SPICSR	SPI Data I/O Register SPI Control Register SPI Control/Status Register	xxh 0xh 00h	R/W R/W R/W
0024h	FLASH	FCSR	Flash Control/Status Register	00h	R/W
0025h 0026h 0027h 0028h 0029h 002Ah	ITC	ISPR0 ISPR1 ISPR2 ISPR3 EICR0 EICR1	Interrupt Software Priority Register 0 Interrupt Software Priority Register 1 Interrupt Software Priority Register 2 Interrupt Software Priority Register 3 External Interrupt Control Register 0 External Interrupt Control Register 1	FFh FFh FFh FFh 00h 00h	R/W R/W R/W R/W R/W R/W
002Bh 002Ch	AWU	AWUCSR AWUPR	Auto Wake up f. Halt Control/Status Register Auto Wake Up From Halt Prescaler	00h FFh	R/W R/W
002Dh 002Eh	CKCTRL	SICSR MCCSR	System Integrity Control / Status Register Main Clock Control / Status Register	0xh 00h	R/W R/W
002Fh 0030h	WWDG	WDGCR WDGWR	Watchdog Control Register Watchdog Window Register	7Fh 7Fh	R/W R/W
0031h 0032h 0033h 0034h 0035h 0036h 0037h 0038h 0039h 003Ah 003Bh	PWM ART	PWMDCR3 PWMDCR2 PWMDCR1 PWMDCR0 PWMCR ARTCSR ARTCAR ARTARR ARTICCSR ARTICR1 ARTICR2	Pulse Width Modulator Duty Cycle Register 3 PWM Duty Cycle Register 2 PWM Duty Cycle Register 1 PWM Duty Cycle Register 0 PWM Control register Auto-Reload Timer Control/Status Register Auto-Reload Timer Counter Access Register Auto-Reload Timer Auto-Reload Register ART Input Capture Control/Status Register ART Input Capture Register 1 ART Input Capture register 2	00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h	R/W R/W R/W R/W R/W R/W R/W R/W R/W Read Only Read Only
003Ch 003Dh 003Eh 003Fh 0040h 0041h 0042h 0043h 0044h	8-BIT TIMER	T8CR2 T8CR1 T8CSR T8IC1R T8OC1R T8CTR T8ACTR T8IC2R T8OC2R	Timer Control Register 2 Timer Control Register 1 Timer Control/Status Register Timer Input Capture 1 Register Timer Output Compare 1 Register Timer Counter Register Timer Alternate Counter Register Timer Input Capture 2 Register Timer Output Compare 2 Register	00h 00h 00h xxh 00h FCh FCh xxh 00h	R/W R/W Read Only Read Only R/W Read Only Read Only Read Only R/W
0045h 0046h 0047h	ADC	ADCCSR ADCDRH ADCDRL	Control/Status Register Data High Register Data Low Register	00h 00h 00h	R/W Read Only Read Only

Address	Block	Register Label	Register Name	Reset Status	Remarks
0048h	LINSCI1 (LIN Master/ Slave)	SCI1ISR	SCI1 Status Register	C0h	Read Only
0049h		SCI1DR	SCI1 Data Register	xxh	R/W
004Ah		SCI1BRR	SCI1 Baud Rate Register	00h	R/W
004Bh		SCI1CR1	SCI1 Control Register 1	xxh	R/W
004Ch		SCI1CR2	SCI1 Control Register 2	00h	R/W
004Dh		SCI1CR3	SCI1Control Register 3	00h	R/W
004Eh		SCI1ERPR	SCI1 Extended Receive Prescaler Register	00h	R/W
004Fh		SCI1ETPR	SCI1 Extended Transmit Prescaler Register	00h	R/W
0050h	Reserved Area (1 byte)				
0051h	16-BIT TIMER	T16CR2	Timer Control Register 2	00h	R/W
0052h		T16CR1	Timer Control Register 1	00h	R/W
0053h		T16CSR	Timer Control/Status Register	00h	R/W
0054h		T16IC1HR	Timer Input Capture 1 High Register	xxh	Read Only
0055h		T16IC1LR	Timer Input Capture 1 Low Register	xxh	Read Only
0056h		T16OC1HR	Timer Output Compare 1 High Register	80h	R/W
0057h		T16OC1LR	Timer Output Compare 1 Low Register	00h	R/W
0058h		T16CHR	Timer Counter High Register	FFh	Read Only
0059h		T16CLR	Timer Counter Low Register	FCh	Read Only
005Ah		T16ACHR	Timer Alternate Counter High Register	FFh	Read Only
005Bh		T16ACLR	Timer Alternate Counter Low Register	FCh	Read Only
005Ch		T16IC2HR	Timer Input Capture 2 High Register	xxh	Read Only
005Dh		T16IC2LR	Timer Input Capture 2 Low Register	xxh	Read Only
005Eh		T16OC2HR	Timer Output Compare 2 High Register	80h	R/W
005Fh		T16OC2LR	Timer Output Compare 2 Low Register	00h	R/W
0060h	LINSCI2 (LIN Master)	SCI2SR	SCI2 Status Register	C0h	Read Only
0061h		SCI2DR	SCI2 Data Register	xxh	R/W
0062h		SCI2BRR	SCI2 Baud Rate Register	00h	R/W
0063h		SCI2CR1	SCI2 Control Register 1	xxh	R/W
0064h		SCI2CR2	SCI2 Control Register 2	00h	R/W
0065h		SCI2CR3	SCI2 Control Register 3	00h	R/W
0066h		SCI2ERPR	SCI2 Extended Receive Prescaler Register	00h	R/W
0067h		SCI2ETPR	SCI2 Extended Transmit Prescaler Register	00h	R/W
0068h to 007Fh	Reserved area (24 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.

4 FLASH PROGRAM MEMORY

4.1 INTRODUCTION

The ST7 dual voltage High Density Flash (HDFlash) is a non-volatile memory that can be electrically erased as a single block or by individual sectors and programmed on a Byte-by-Byte basis using an external V_{PP} supply.

The HDFlash devices can be programmed and erased off-board (plugged in a programming tool) or on-board using ICP (In-Circuit Programming) or IAP (In-Application Programming).

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

4.2 MAIN FEATURES

- 3 Flash programming modes:
  - Insertion in a programming tool. In this mode, all sectors including option bytes can be programmed or erased.
  - ICP (In-Circuit Programming). In this mode, all sectors including option bytes can be programmed or erased without removing the device from the application board.
  - IAP (In-Application Programming) In this mode, all sectors except Sector 0, can be programmed or erased without removing the device from the application board and while the application is running.
- ICT (In-Circuit Testing) for downloading and executing user application test patterns in RAM
- Read-out protection
- Register Access Security System (RASS) to prevent accidental programming or erasing

4.3 STRUCTURE

The Flash memory is organised in sectors and can be used for both code and data storage.

Depending on the overall Flash memory size in the microcontroller device, there are up to three user sectors (see Table 3). Each of these sectors can be erased independently to avoid unnecessary erasing of the whole Flash memory when only a partial erasing is required.

The first two sectors have a fixed size of 4 Kbytes (see Figure 6). They are mapped in the upper part of the ST7 addressing space so the reset and interrupt vectors are located in Sector 0 (F000h-FFFFh).

Table 4. Sectors available in Flash devices

Flash Size (bytes)	Available Sectors
4K	Sector 0
8K	Sectors 0,1
> 8K	Sectors 0,1, 2

4.3.1 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.

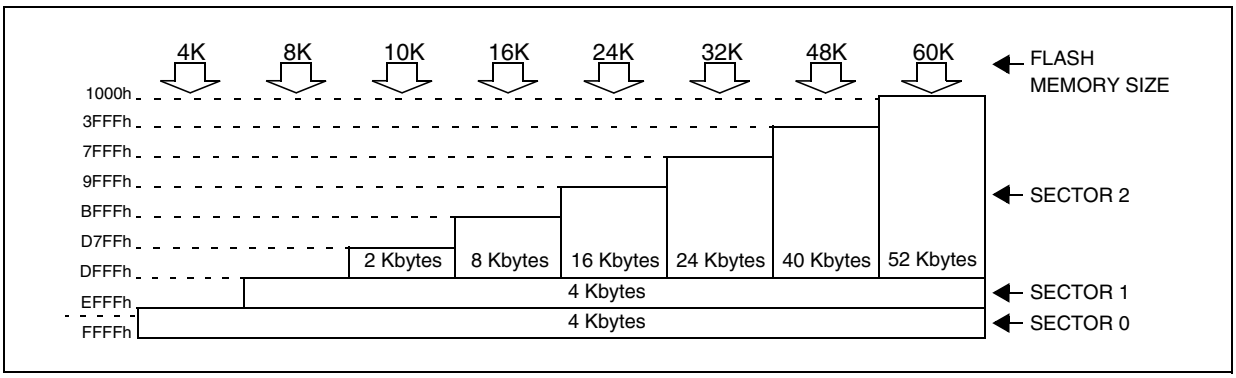
In Flash devices, this protection is removed by reprogramming the option. In this case, the entire program memory is first automatically erased and the device can be reprogrammed.

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.

Figure 6. Memory Map and Sector Address



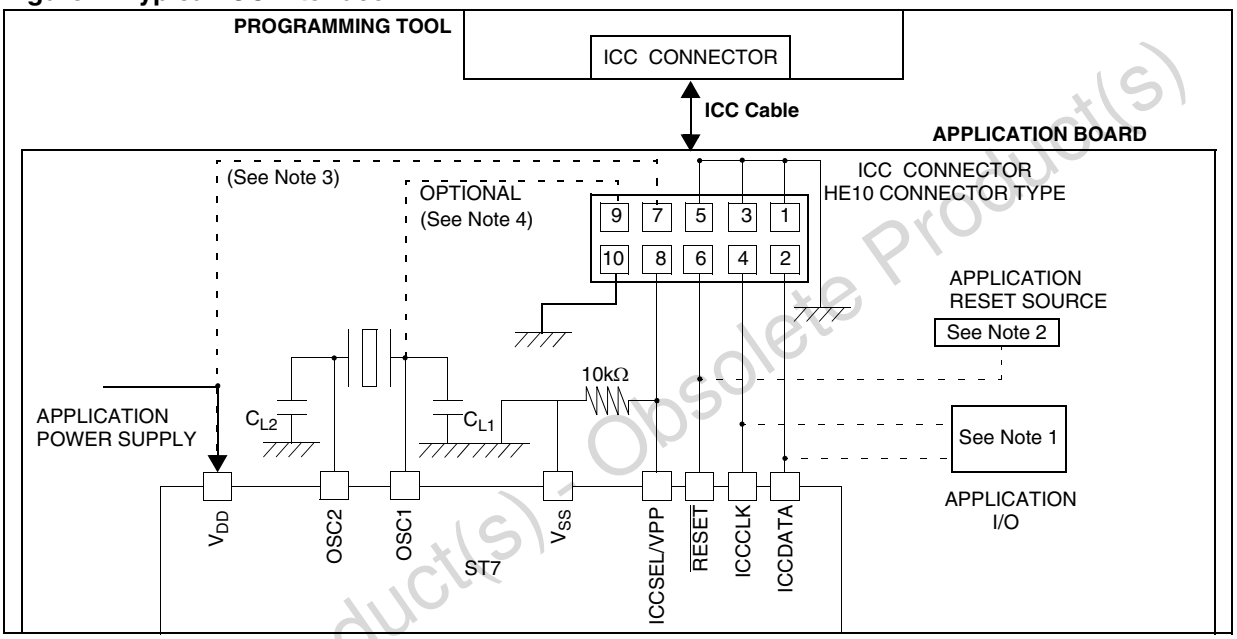
FLASH PROGRAM MEMORY (Cont'd)

4.4 ICC INTERFACE

ICC (In-Circuit Communication) needs a minimum of four and up to six pins to be connected to the programming tool (see Figure 7). These pins are:

- $\overline{\text{RESET}}$: device reset
- V_{SS}: device power supply ground
- ICCCLK: ICC output serial clock pin
- ICCDATA: ICC input/output serial data pin
- ICCSEL/V_{PP}: programming voltage
- OSC1(or OSCIN): main clock input for external source (optional)
- V_{DD}: application board power supply (see Figure 7, Note 3)

Figure 7. Typical ICC Interface



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 implemented in case another device forces the signal. Refer to the Programming Tool documentation for recommended resistor values.

2. During the ICC 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 $\overline{\text{RESET}}$ 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 or OSCIN 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.

FLASH PROGRAM MEMORY (Cont'd)

4.5 ICP (IN-CIRCUIT PROGRAMMING)

To perform ICP the microcontroller must be switched to ICC (In-Circuit Communication) mode by an external controller or programming tool.

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

When using an STMicroelectronics or third-party programming tool that supports ICP and the specific microcontroller device, the user needs only to implement the ICP hardware interface on the application board (see Figure 7). For more details on the pin locations, refer to the device pinout description.

4.6 IAP (IN-APPLICATION PROGRAMMING)

This mode uses a BootLoader program previously stored in Sector 0 by the user (in ICP mode or by plugging the device in a programming tool).

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.). For example, it is possible to download code from the SPI, SCI or other type of serial interface and program it in the Flash. IAP mode can be used to program any of the Flash sectors except Sector 0, which is write/erase protected to allow recovery in case errors occur during the programming operation.

4.7 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*.

4.8 REGISTER DESCRIPTION

FLASH CONTROL/STATUS REGISTER (FCSR)

Read/Write

Reset Value: 0000 0000 (00h)

7							0
0	0	0	0	0	0	0	0

This register is reserved for use by Programming Tool software. It controls the Flash programming and erasing operations.

Table 5. Flash Control/Status Register Address and Reset Value

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0024h	**FCSR** Reset Value	0	0	0	0	0	0	0	0

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 six CPU registers shown in Figure 8 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)

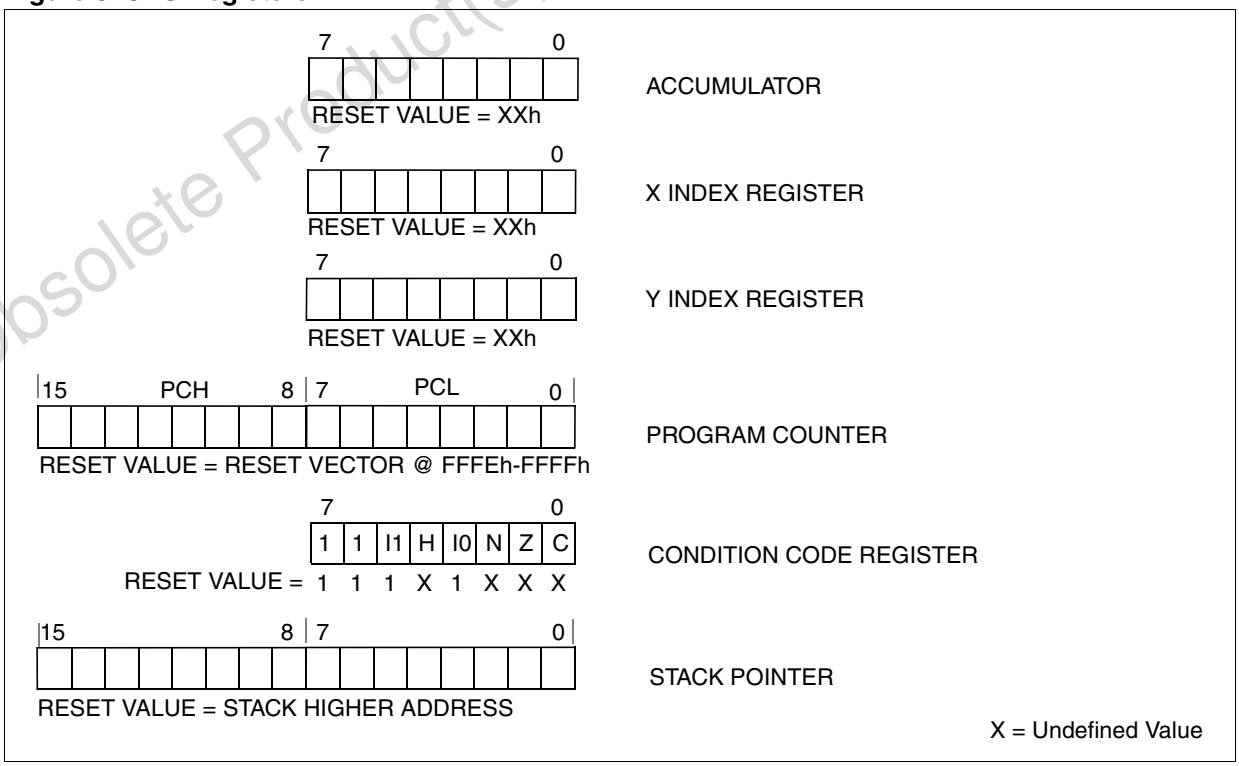
These 8-bit registers are used to create effective addresses or as temporary storage areas for data manipulation. (The Cross-Assembler generates a precede instruction (PRE) to indicate that the following instruction refers to the Y register.)

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 8. 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 (that is, 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 FFh

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 9).

Since the stack is 256 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.

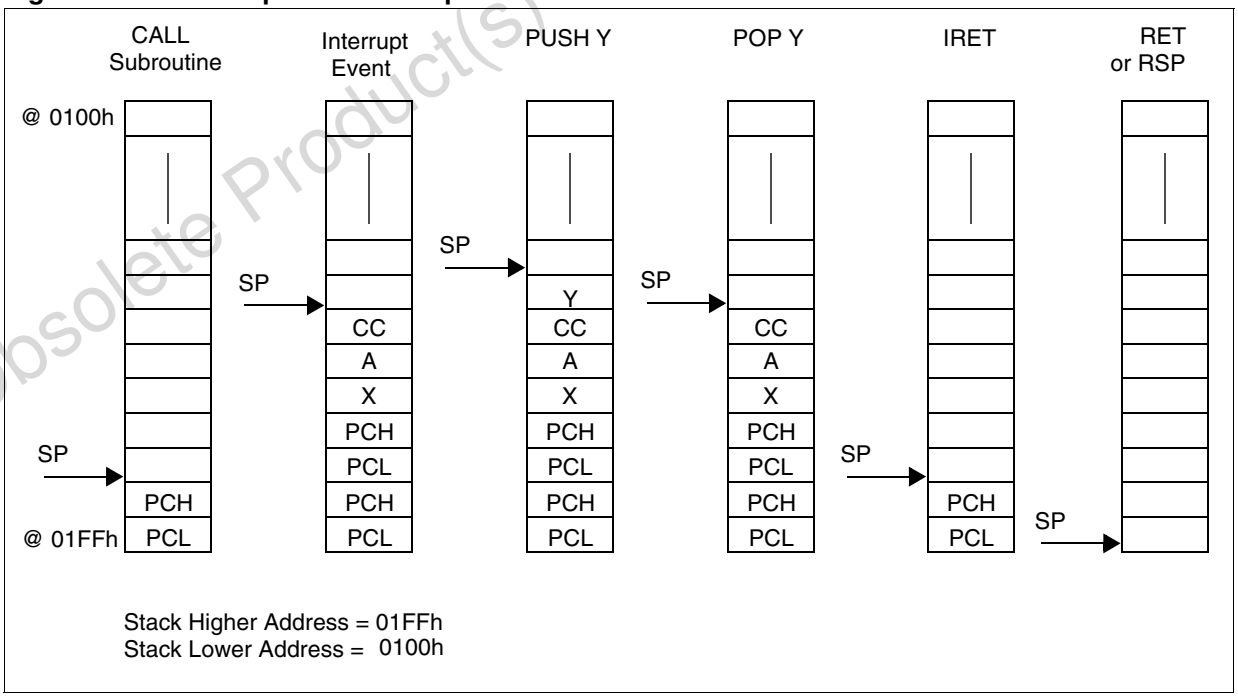
The stack is used to save the return address during a subroutine call and the CPU context during an interrupt. The user may also directly manipulate the stack by means of the PUSH and POP instructions. In the case of an interrupt, the PCL is stored at the first location pointed to by the SP. Then the other registers are stored in the next locations as shown in Figure 9.

– 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 9. 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 11.

For more details, refer to dedicated parametric section.

Main features

- Optional PLL for multiplying the frequency by 2
- Reset Sequence Manager (RSM)
- Multi-Oscillator Clock Management (MO)
  - 4 Crystal/Ceramic resonator oscillators
- System Integrity Management (SI)
  - Main supply Low voltage detection (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 range 2 to 4 MHz, 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 187.

Figure 10. PLL Block Diagram

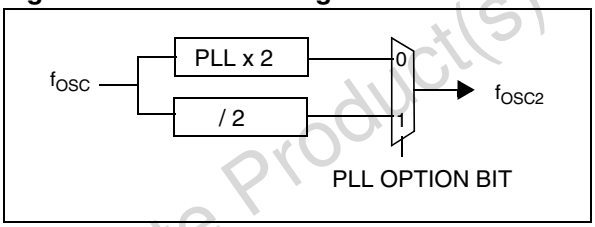
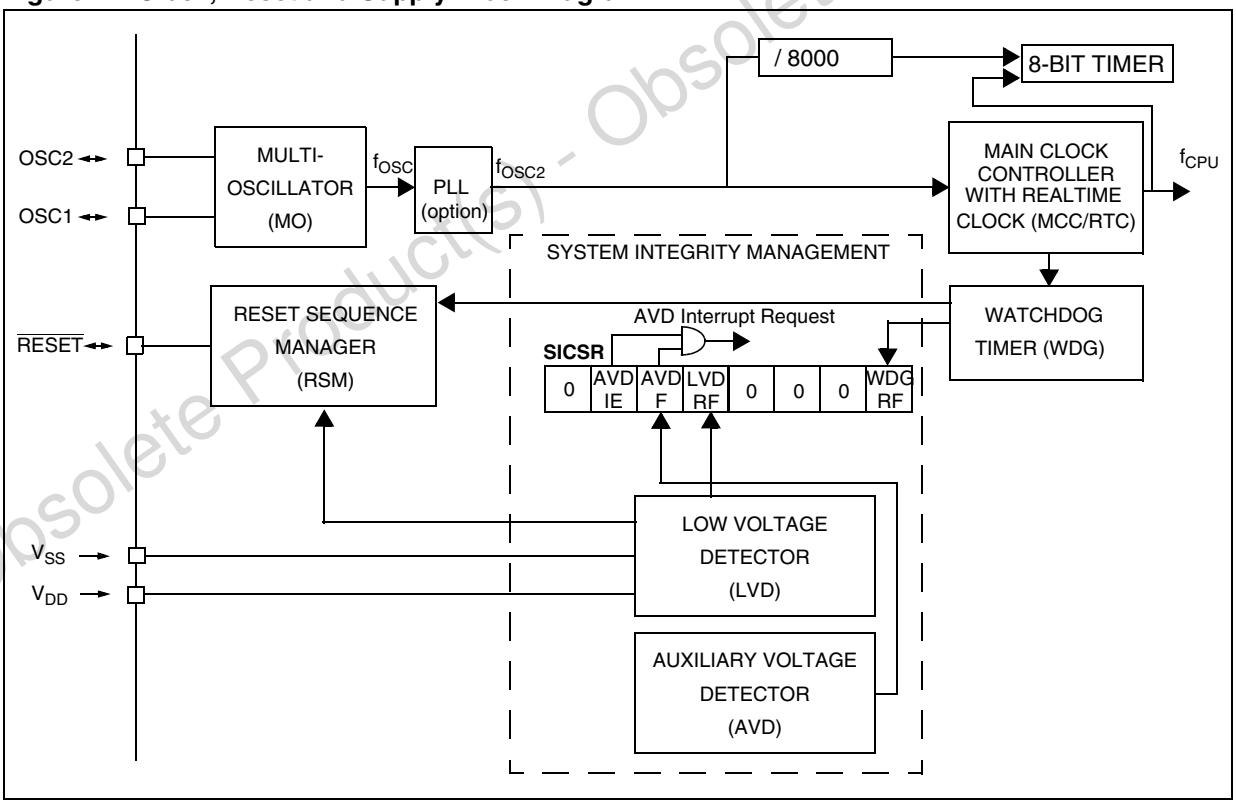


Figure 11. Clock, Reset and Supply Block Diagram



6.2 MULTI-OSCILLATOR (MO)

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

- an external source
- a crystal or ceramic resonator oscillator

Each oscillator is optimized for a given frequency range in terms of consumption and is selectable through the option byte. The associated hardware configuration are shown in Table 6. 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 Effect 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 (> 16 MHz), putting the ST7 in an unsafe/undefined state. The product behavior must therefore be considered undefined when the OSC pins are left unconnected.

External Clock Source

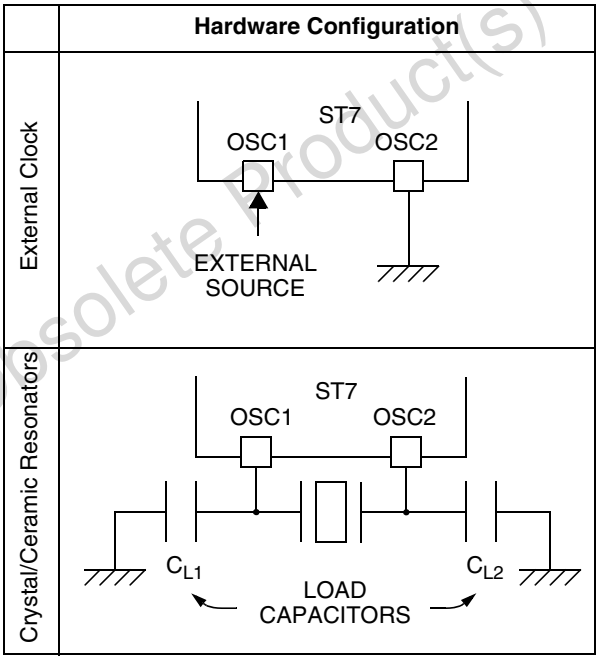
In external clock mode, a clock signal (square, sinus or triangle) with ~50% duty cycle has to drive the OSC1 pin while the OSC2 pin is tied to ground.

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 five oscillators with different frequency ranges must be done by option byte in order to reduce consumption (refer to Section 14.1 on page 210 for more details on the frequency ranges). The resonator and the load capacitors must 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.

Table 6. ST7 Clock Sources



6.3 RESET SEQUENCE MANAGER (RSM)

6.3.1 Introduction

The reset sequence manager includes three RESET sources as shown in Figure 2:

- 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.

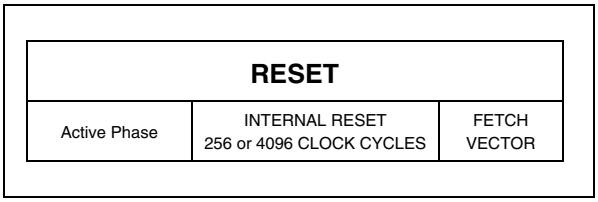
The basic RESET sequence consists of three phases as shown in Figure 1:

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

The 256 or 4096 CPU clock cycle delay allows the oscillator to stabilize 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 two clock cycles.

Figure 12. RESET Sequence Phases



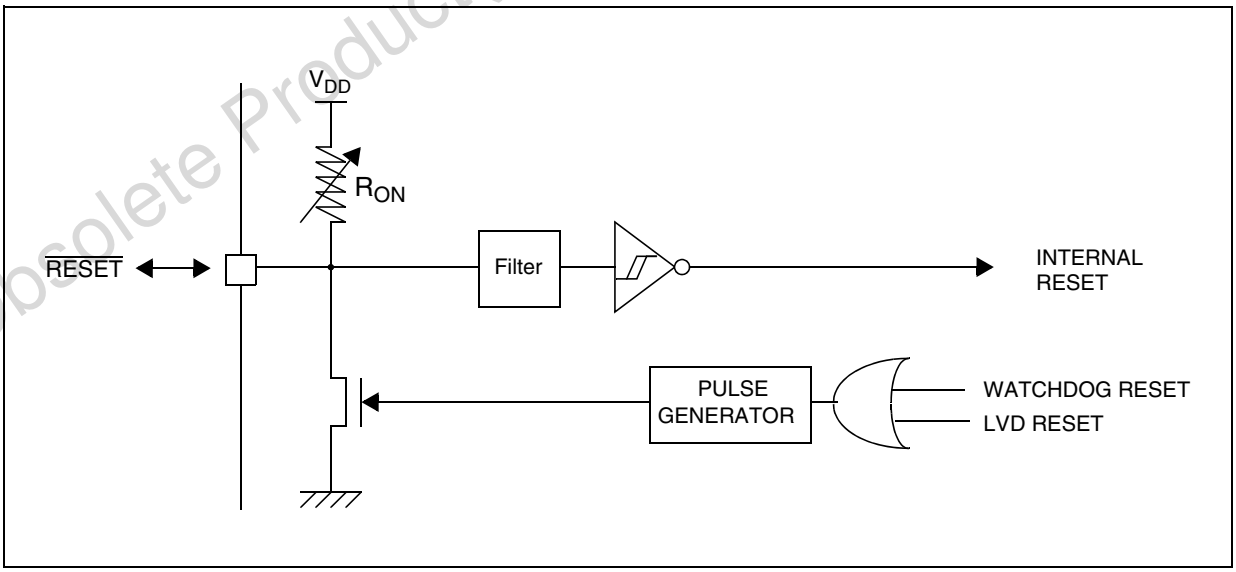
Caution: When the ST7 is unprogrammed or fully erased, the Flash is blank and the RESET vector is not programmed. For this reason, it is recommended to keep the RESET pin in low state until programming mode is entered, in order to avoid unwanted behavior.

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 3). This detection is asynchronous and therefore the MCU can enter reset state even in HALT mode.

Figure 13. 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 3.

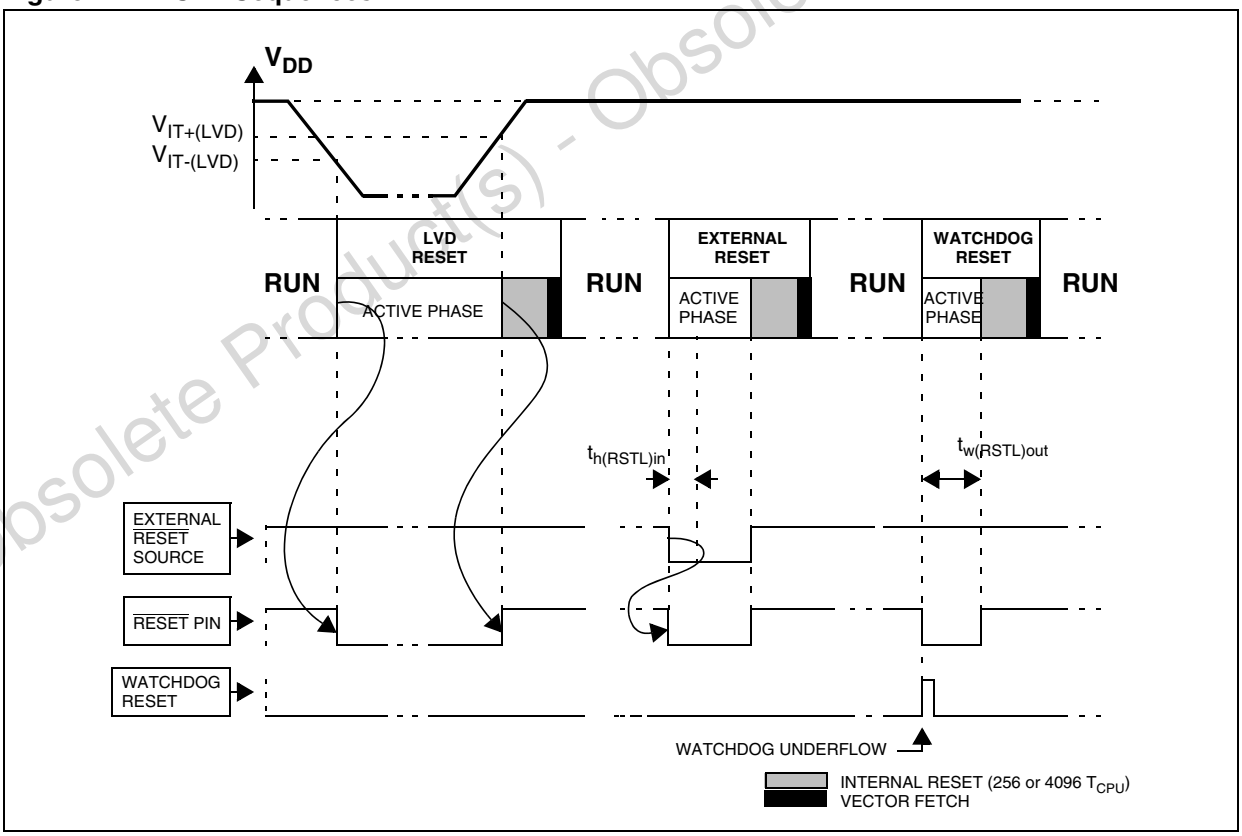
The LVD filters spikes on V_{DD} larger than $t_{g(VDD)}$ to avoid parasitic resets.

6.3.5 Internal Watchdog RESET

The RESET sequence generated by a internal Watchdog counter overflow is shown in Figure 3.

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 14. RESET Sequences



6.4 SYSTEM INTEGRITY MANAGEMENT (SI)

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

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 $V_{IT-(LVD)}$ reference value. This means that it secures the power-up as well as the power-down keeping the ST7 in reset.

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

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

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

The LVD function is illustrated in Figure 15.

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

– under full software control
– in static safe reset

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

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

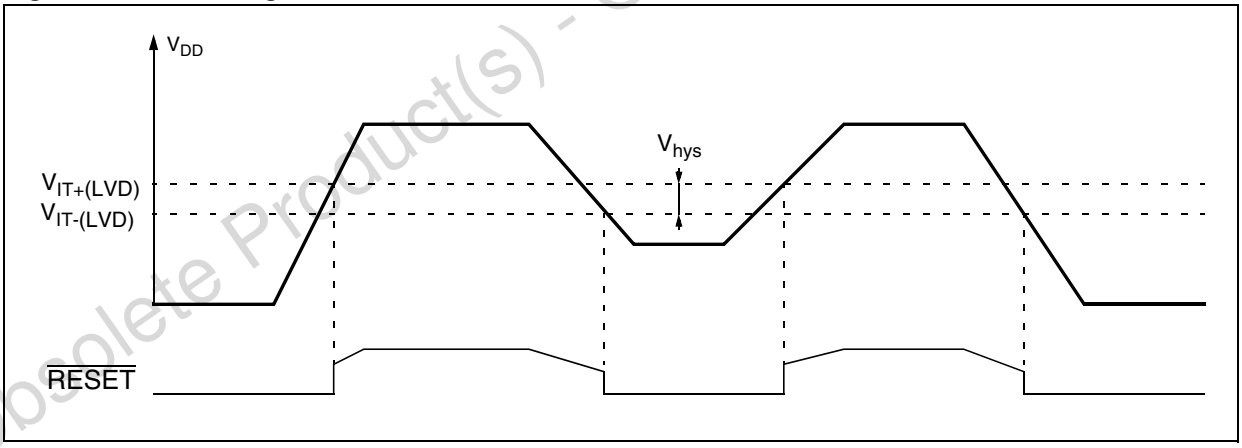
Notes:

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

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

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 15. 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 $V_{IT-(AVD)}$ and $V_{IT+(AVD)}$ reference value and the V_{DD} main supply. The $V_{IT-(AVD)}$ reference value for falling voltage is lower than the $V_{IT+(AVD)}$ 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 (AVDF) in the SICSR register. This bit is read only.

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

6.4.2.1 Monitoring the V_{DD} Main Supply

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 16.

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 two 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 occurs.

Figure 16. Using the AVD to Monitor V_{DD}