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

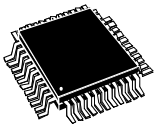
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

ST72324Lxx

3V range 8-bit MCU with 8 to 32K Flash/ROM, 10-bit ADC, 4 timers, SPI, SCI interface

Features

- **Memories**
  - 8 to 32K dual voltage High Density Flash (HDFlash) or ROM with read-out protection capability. In-Application Programming and In-Circuit Programming for HDFlash devices
  - 384 to 1K bytes RAM
  - HDFlash endurance: 100 cycles, data retention: 40 years at 85°C
- **Clock, Reset And Supply Management**
  - Clock sources: crystal/ceramic resonator oscillators, internal RC oscillator, and bypass for external clock
  - PLL for 2x frequency multiplication
  - Four power saving modes: Halt, Active-Halt, Wait and Slow
- **Interrupt Management**
  - Nested interrupt controller
  - 10 interrupt vectors plus TRAP and RESET
  - 9/6 external interrupt lines (on 4 vectors)
- **Up to 32 I/O Ports**
  - 32/24 multifunctional bidirectional I/O lines
  - 22/17 alternate function lines
  - 12/10 high sink outputs
- **4 Timers**
  - Main Clock Controller with: Real time base, Beep and Clock-out capabilities
  - Configurable watchdog timer
  - 16-bit Timer A with: 1 input capture, 1 output compare, external clock input, PWM and pulse generator modes
  - 16-bit Timer B with: 2 input captures, 2 output compares, PWM and pulse generator modes



LQFP44
10 x 10

LQFP32
7 x 7

LQFP48
7 x 7

SDIP32
400 mil

- **2 Communication Interfaces**
  - SPI synchronous serial interface
  - SCI asynchronous serial interface
- **1 Analog Peripheral**
  - 10-bit ADC with up to 12 input ports
- **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
  - In-Circuit Testing capability

Table 1. Device Summary

Features	ST72324LJ6 ST72324LK6	ST72324LJ4 ST72324LK4 ST72324LS4	ST72324LJ2 ST72324LK2 ST72324LS2
Program memory - bytes	Flash 32K	Flash/ROM 16K	Flash/ROM 8K
RAM (stack) - bytes	1024 (256)	512 (256)	384 (256)
Voltage Range	2.85 to 3.6V		
Temp. Range	up to -40°C to +85°C		
Packages	LQFP44 10x10 (J), LQFP48 7x7 (S), SDIP32, LQFP32 7x7 (K)		

Rev. 5

Table of Contents

Table of Contents

1 DESCRIPTION

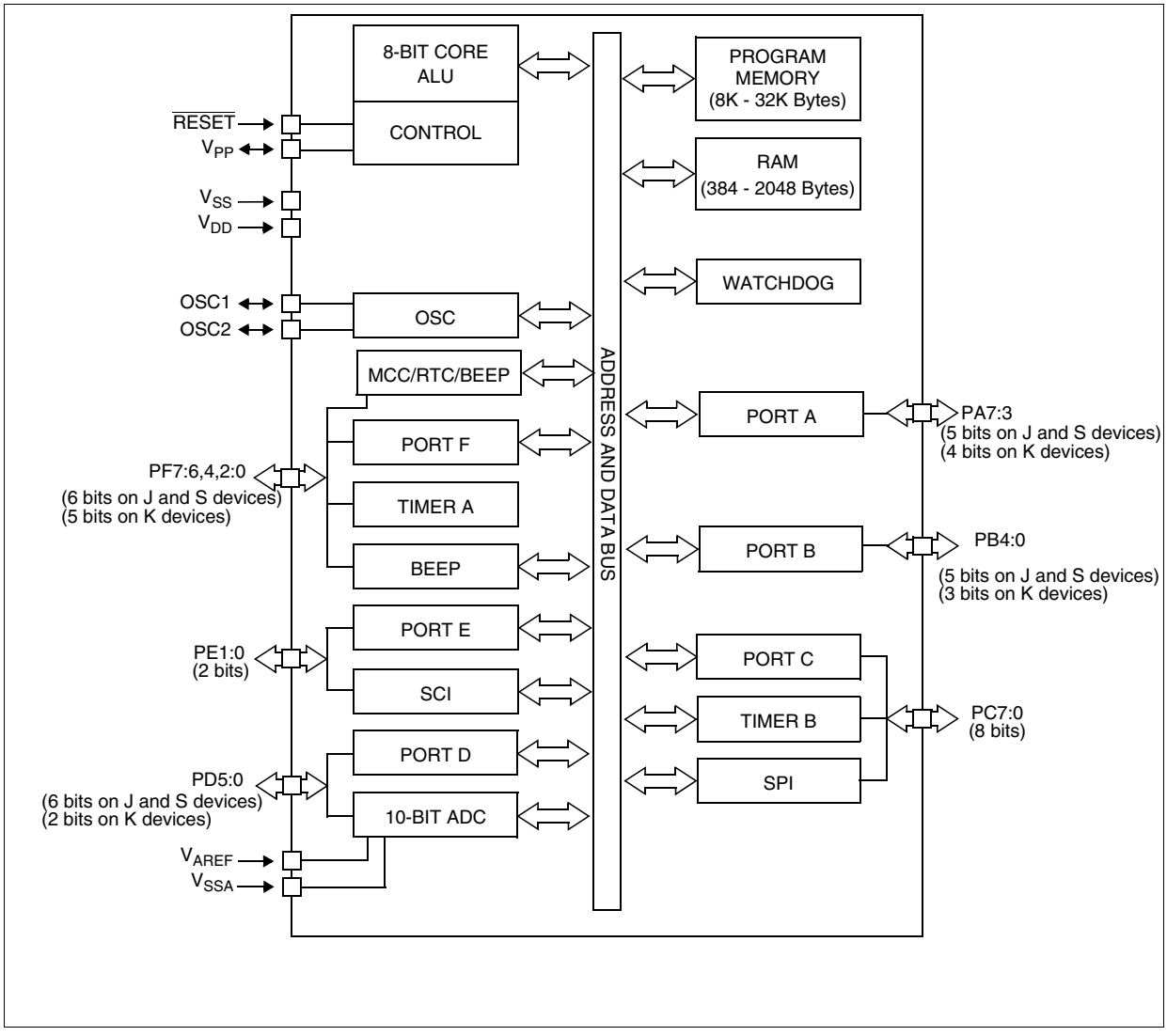
The ST72F324L and ST72324BL devices are members of the ST7 microcontroller family designed for mid-range applications running at 3.3V. Different package options offer up to 32 I/O pins. All devices are based on a common industry-standard 8-bit core, featuring an enhanced instruction set and are available 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, 2 general purpose timers, an SPI interface and an SCI interface.

For power economy, microcontroller can switch dynamically into WAIT, SLOW, ACTIVE-HALT 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



2 PIN DESCRIPTION

Figure 2. 48-Pin LQFP 7x7 Device Pinout

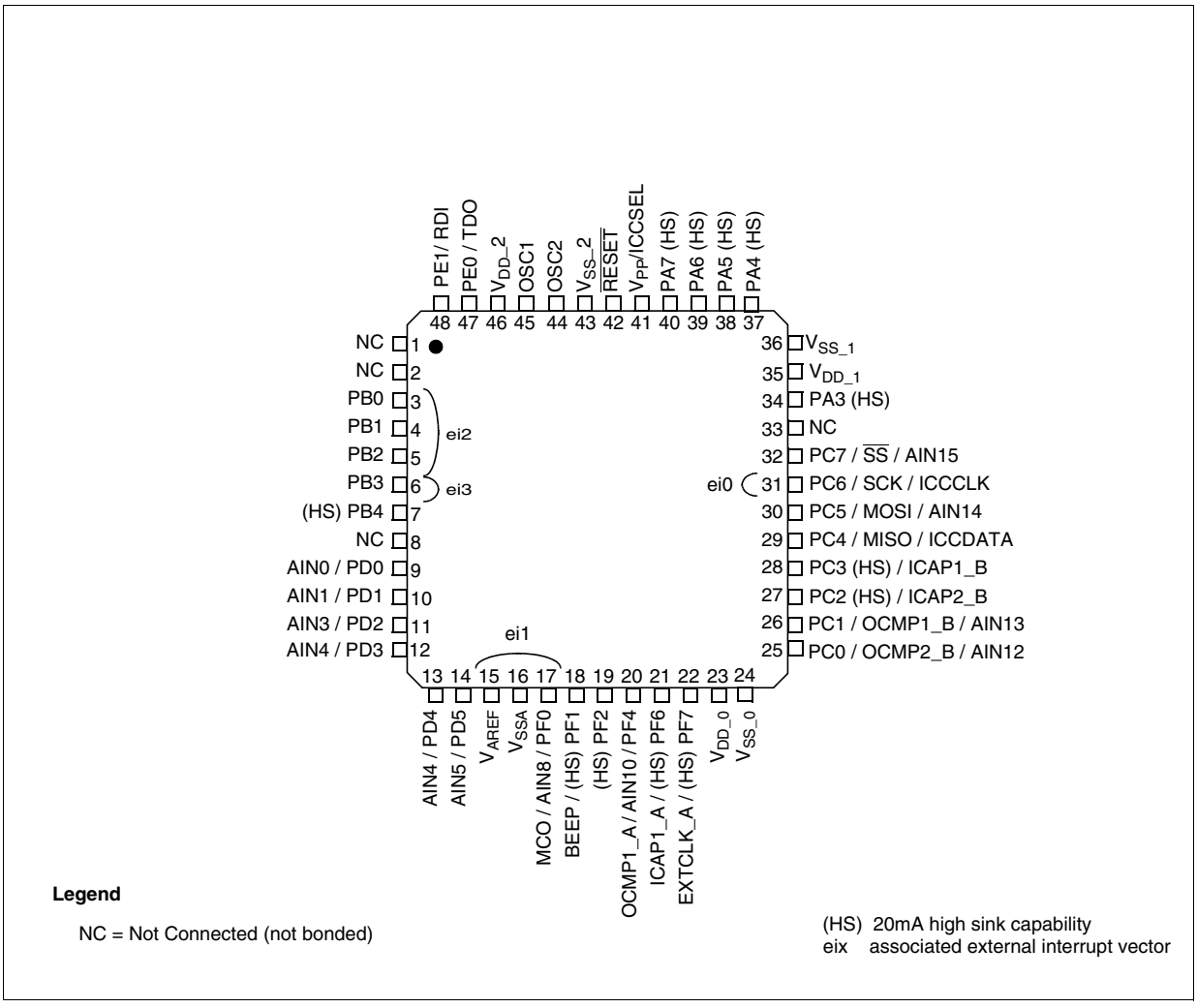
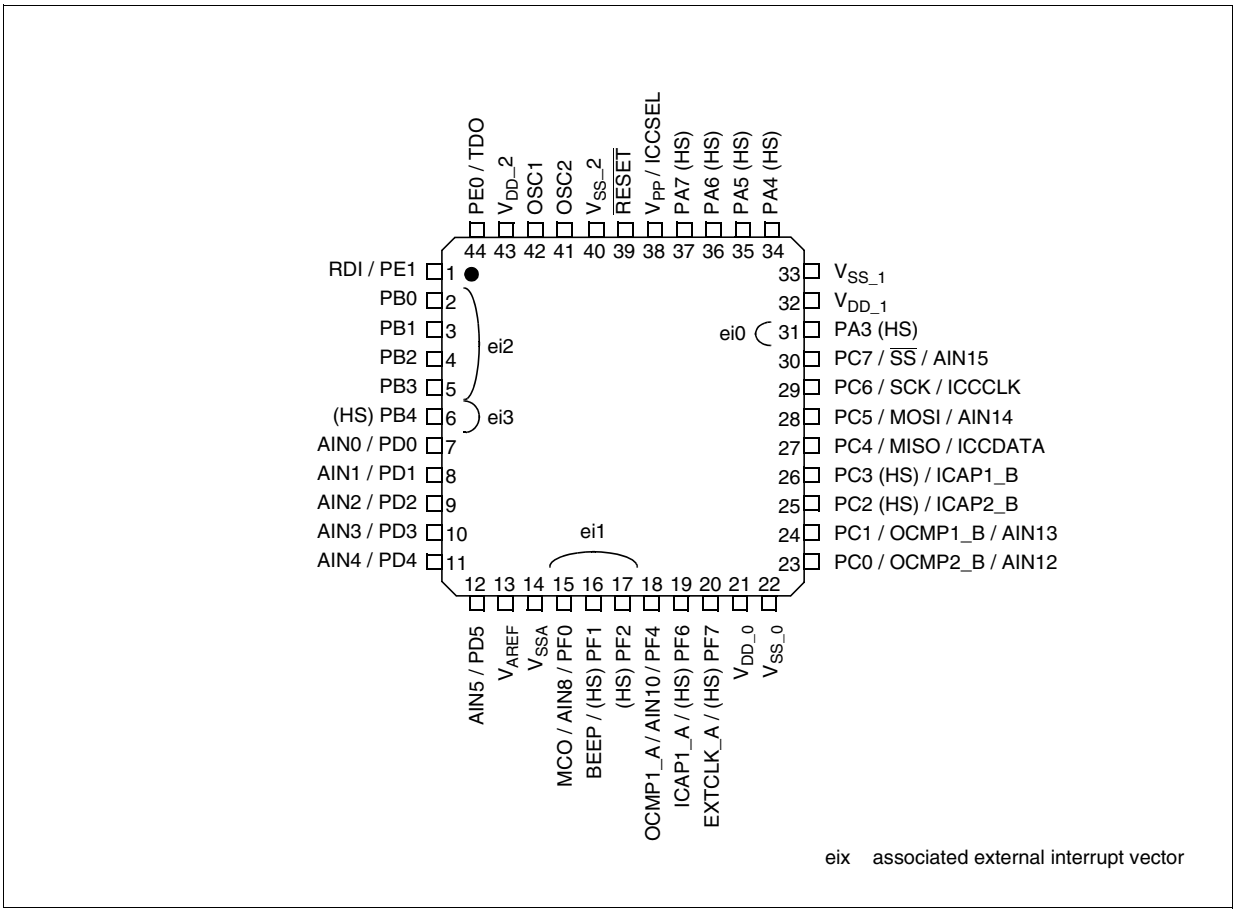


Figure 3. 44-Pin LQFP Package Pinout



PIN DESCRIPTION (Cont'd)

Figure 4. 32-Pin SDIP Package Pinout

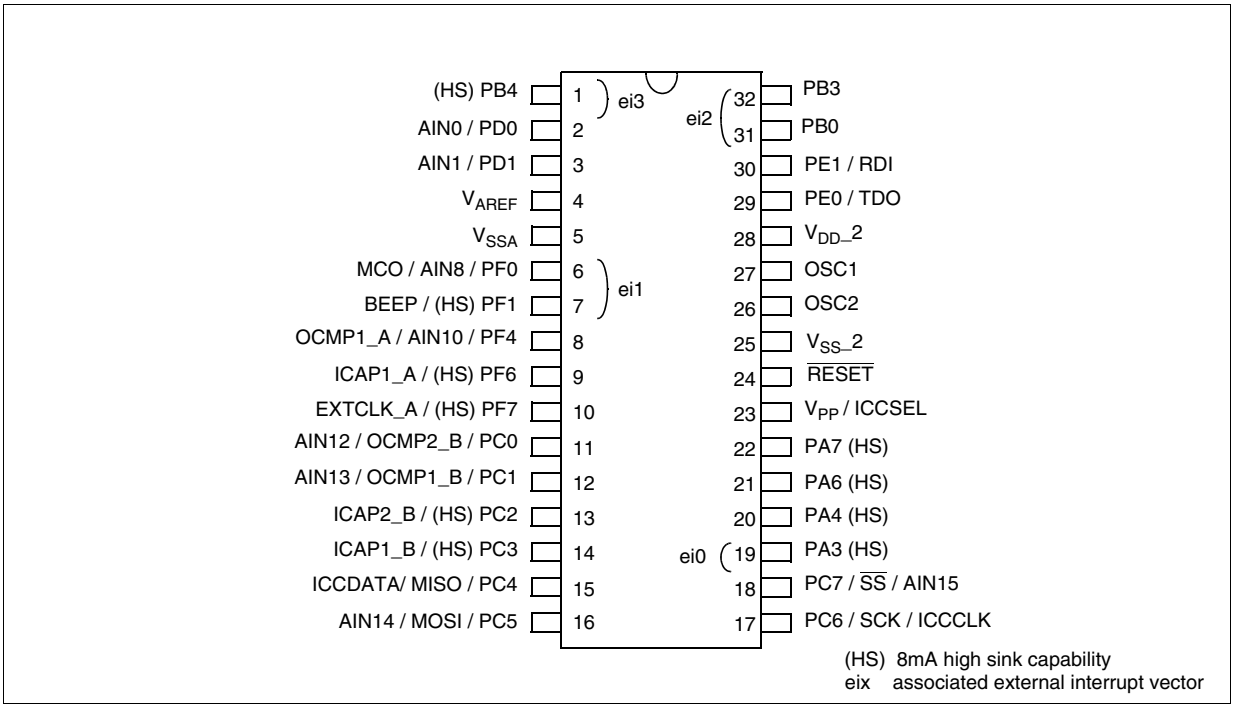
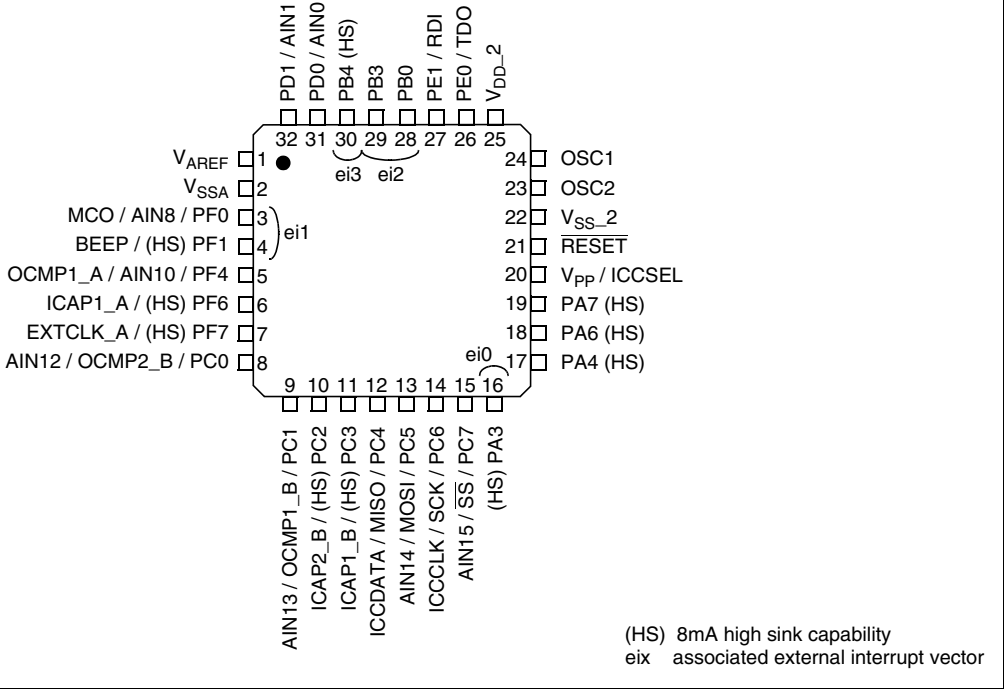


Figure 5. 32-Pin LQFP 7x7 Package Pinout



PIN DESCRIPTION (Cont'd)

For more details, refer to "ELECTRICAL CHARACTERISTICS" on page 110

Legend / Abbreviations for Table 2:

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

In/Output level: C = CMOS
C_T= CMOS with input trigger

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

Port and control configuration:

- Input: float = floating, wpu = weak pull-up, int = interrupt [1], ana = analog ports
- Output: OD = open drain [2], PP = push-pull

Refer to "I/O PORTS" on page 40 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 2. Device Pin Description

Pin n°				Pin Name	Type	Level		Port						Main function (after reset)	Alternate Function	
								Input				Output				
LQFP48	LQFP44	LQFP32	SDIP32			Input	Output	float	wpu	int	ana	OD	PP			
7	6	30	1	PB4 (HS)	I/O	C_T	HS	**X**	ei3			X	X	Port B4		
9	7	31	2	PD0/AIN0	I/O	C_T		**X**	X		X	X	X	Port D0	ADC Analog Input 0	
10	8	32	3	PD1/AIN1	I/O	C_T		**X**	X		X	X	X	Port D1	ADC Analog Input 1	
11	9			PD2/AIN2	I/O	C_T		**X**	X		X	X	X	Port D2	ADC Analog Input 2	
12	10			PD3/AIN3	I/O	C_T		**X**	X		X	X	X	Port D3	ADC Analog Input 3	
13	11			PD4/AIN4	I/O	C_T		**X**	X		X	X	X	Port D4	ADC Analog Input 4	
14	12			PD5/AIN5	I/O	C_T		**X**	X		X	X	X	Port D5	ADC Analog Input 5	
15	13	1	4	V_{AREF}	S									Analog Reference Voltage for ADC[5]		
16	14	2	5	V_{SSA}	S									Analog Ground Voltage[5]		
17	15	3	6	PF0/MCO/AIN8	I/O	C_T		**X**	ei1		X	X	X	Port F0	Main clock out (f_{OSC}/2)	ADC Analog Input 8
18	16	4	7	PF1 (HS)/BEEP	I/O	C_T	HS	**X**	ei1			X	X	Port F1	Beep signal output	
19	17			PF2 (HS)	I/O	C_T	HS	**X**		ei1		X	X	Port F2		
20	18	5	8	PF4/OCMP1_A/ AIN10	I/O	C_T		**X**	X		X	X	X	Port F4	Timer A Output Compare 1	ADC Analog Input 10
21	19	6	9	PF6 (HS)/ICAP1_A	I/O	C_T	HS	**X**	X			X	X	Port F6	Timer A Input Capture 1	
22	20	7	10	PF7 (HS)/ EXTCLK_A	I/O	C_T	HS	**X**	X			X	X	Port F7	Timer A External Clock Source	
23	21			V_{DD_0}	S									Digital Main Supply Voltage[5]		
24	22			V_{SS_0}	S									Digital Ground Voltage[5]		
25	23	8	11	PC0/OCMP2_B/ AIN12	I/O	C_T		**X**	X		X	X	X	Port C0	Timer B Output Compare 2	ADC Analog Input 12
26	24	9	12	PC1/OCMP1_B/ AIN13	I/O	C_T		**X**	X		X	X	X	Port C1	Timer B Output Compare 1	ADC Analog Input 13

Pin n°				Pin Name	Type	Level		Port						Main function (after reset)	Alternate Function	
								Input				Output				
LQFP48	LQFP44	LQFP32	SDIP32			Input	Output	float	wpu	int	ana	OD	PP			
27	25	10	13	PC2 (HS)/ICAP2_B	I/O	C_T	HS	**X**	X			X	X	Port C2	Timer B Input Capture 2	
28	26	11	14	PC3 (HS)/ICAP1_B	I/O	C_T	HS	**X**	X			X	X	Port C3	Timer B Input Capture 1	
29	27	12	15	PC4/MISO/ICCDATA	I/O	C_T		**X**	X			X	X	Port C4	SPI Master In / Slave Out Data	ICC Data Input
30	28	13	16	PC5/MOSI/AIN14	I/O	C_T		**X**	X		X	X	X	Port C5	SPI Master Out / Slave In Data	ADC Analog Input 14
31	29	14	17	PC6/SCK/ICCCLK	I/O	C_T		**X**	X			X	X	Port C6	SPI Serial Clock	ICC Clock Output
32	30	15	18	PC7/$\overline{SS}$/AIN15	I/O	C_T		**X**	X		X	X	X	Port C7	SPI Slave Select (active low)	ADC Analog Input 15
34	31	16	19	PA3 (HS)	I/O	C_T	HS	**X**		ei0		X	X	Port A3		
35	32			V_{DD_1}	S									Digital Main Supply Voltage[5]		
36	33			V_{SS_1}	S									Digital Ground Voltage[5]		
37	34	17	20	PA4 (HS)	I/O	C_T	HS	**X**	X			X	X	Port A4		
38	35			PA5 (HS)	I/O	C_T	HS	**X**	X			X	X	Port A5		
39	36	18	21	PA6 (HS)	I/O	C_T	HS	**X**				T		Port A6 [1]		
40	37	19	22	PA7 (HS)	I/O	C_T	HS	**X**				T		Port A7 [1]		
41	38	20	23	V_{PP} /ICCSEL	I									Must be tied low. In the flash programming mode, this pin acts as the programming voltage input V_{PP}. See Section 12.9.2 for more details. High voltage must not be applied to ROM devices.		
42	39	21	24	$\overline{RESET}$	I/O	C_T								Top priority non maskable interrupt.		
43	40	22	25	V_{SS_2}	S									Digital Ground Voltage[5]		
44	41	23	26	OSC2	O									Resonator oscillator inverter output		
45	42	24	27	OSC1	I									External clock input or Resonator oscillator inverter input		
46	43	25	28	V_{DD_2}	S									Digital Main Supply Voltage[5]		
47	44	26	29	PE0/TDO	I/O	C_T		**X**	X			X	X	Port E0	SCI Transmit Data Out	
48	1	27	30	PE1/RDI	I/O	C_T		**X**	X			X	X	Port E1	SCI Receive Data In	
3	2	28	31	PB0	I/O	C_T		**X**	ei2			X	X	Port B0		
4	3			PB1	I/O	C_T		**X**	ei2			X	X	Port B1		
5	4			PB2	I/O	C_T		**X**	ei2			X	X	Port B2		
6	5	29	32	PB3	I/O	C_T		**X**		ei2		X	X	Port B3		

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. 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 See “I/O PORTS” on page 40. and Section 12.8 I/O PORT PIN CHARACTERISTICS for more details.

3. OSC1 and OSC2 pins connect a crystal/ceramic resonator, or an external source to the on-chip oscillator; see Section 2 PIN DESCRIPTION and Section 12.5 CLOCK AND TIMING CHARACTERISTICS for more details.

4. On the chip, each I/O port has 8 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.

5. It is mandatory to connect all available VDD and VREF pins to the supply voltage and all VSS and VSSA pins to ground.

3 REGISTER & MEMORY MAP

As shown in Figure 6, the MCU is capable of addressing 64K bytes of memories and I/O registers.

The available memory locations consist of 128 bytes of register locations, up to 1024 bytes of RAM and up to 32 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 reserved area can have unpredictable effects on the device.

Figure 6. Memory Map

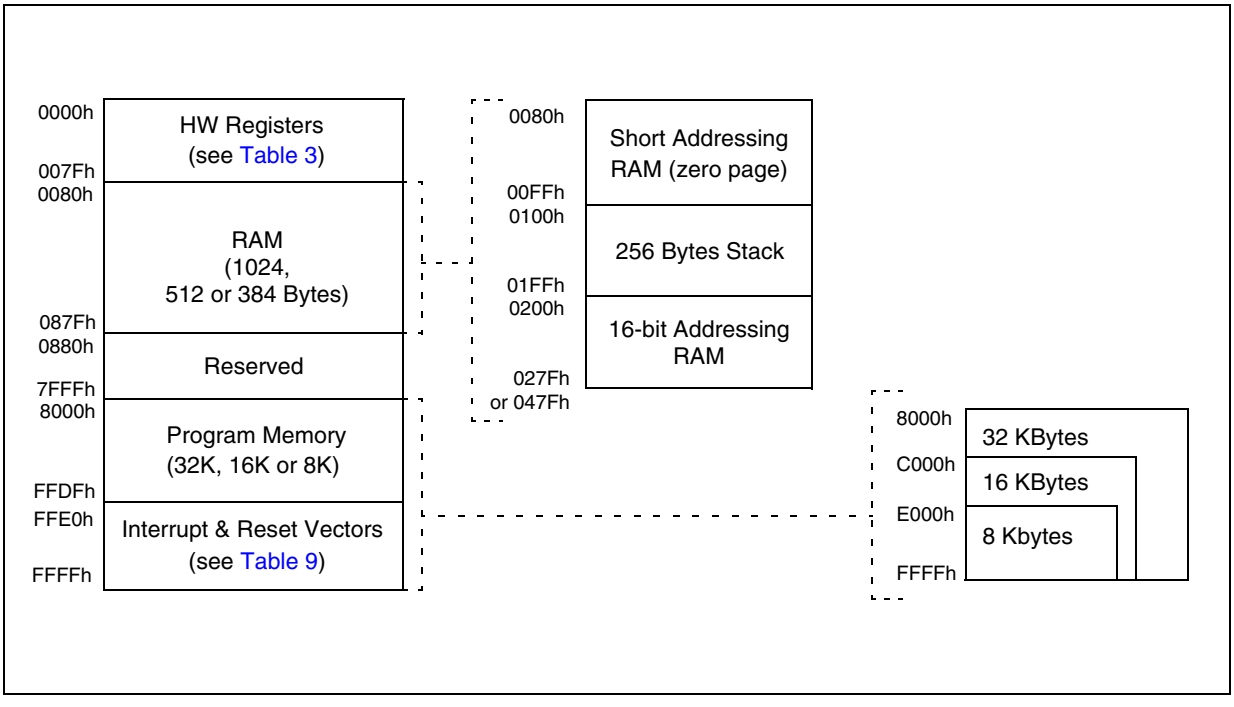


Table 3. Hardware Register Map

Address	Block	Register Label	Register Name	Reset Status	Remarks
0000h 0001h 0002h	Port A [2]	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
0003h 0004h 0005h	Port B [2]	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
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 R/W R/W
0009h 000Ah 000Bh	Port D [2]	PDADR PDDDR PDOR	Port D Data Register Port D Data Direction Register Port D Option Register	00h[1] 00h 00h	R/W R/W R/W
000Ch 000Dh 000Eh	Port E [2]	PEDR PEDDR PEOR	Port E Data Register Port E Data Direction Register Port E Option Register	00h[1] 00h 00h	R/W R/W[2] R/W[2]
000Fh 0010h 0011h	Port F [2]	PFDR PFDDR PFOR	Port F Data Register Port F Data Direction Register Port F Option Register	00h[1] 00h 00h	R/W R/W R/W
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 0025h 0026h 0027h	ITC	ISPR0 ISPR1 ISPR2 ISPR3	Interrupt Software Priority Register 0 Interrupt Software Priority Register 1 Interrupt Software Priority Register 2 Interrupt Software Priority Register 3	FFh FFh FFh FFh	R/W R/W R/W R/W
0028h		EICR	External Interrupt Control Register	00h	R/W
0029h	FLASH	FCSR	Flash Control/Status Register	00h	R/W
002Ah	WATCHDOG	WDGCR	Watchdog Control Register	7Fh	R/W
002Bh	Reserved Area (1 Byte)				
002Ch 002Dh	MCC	MCCSR MCCBCR	Main Clock Control / Status Register Main Clock Controller: Beep Control Register	00h 00h	R/W R/W
002Eh to 0030h	Reserved Area (3 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		TACSR	Timer A Control/Status Register[3)4)]	xxxx x0xxb	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[3)]	xxh	Read Only
003Dh		TAIC2LR	Timer A Input Capture 2 Low Register[3)]	xxh	Read Only
003Eh		TAOC2HR	Timer A Output Compare 2 High Register[4)]	80h	R/W
003Fh		TAOC2LR	Timer A Output Compare 2 Low Register[4)]	00h	R/W
0040h	Reserved Area (1 Byte)				
0041h	TIMER B	TBCR2	Timer B Control Register 2	00h	R/W
0042h		TBCR1	Timer B Control Register 1	00h	R/W
0043h		TBCSR	Timer B Control/Status Register	xxxx x0xxb	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 Register 1	x000 0000h	R/W
0054h		SCICR2	SCI Control Register 2	00h	R/W
0055h		SCIERPR	SCI Extended Receive Prescaler Register	00h	R/W
0056h			Reserved area	---	
0057h		SCIETPR	SCI Extended Transmit Prescaler Register	00h	R/W
0058h to 006Fh	Reserved Area (24 Bytes)				
0070h	ADC	ADCCSR	Control/Status Register	00h	R/W
0071h		ADCDRH	Data High Register	00h	Read Only
0072h		ADCDRL	Data Low Register	00h	Read Only
0073h 007Fh	Reserved Area (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. The Timer A Input Capture 2 pin is not available (not bonded).
   – In Flash devices:
   The TAIC2HR and TAIC2LR registers are not present. Bit 4 of the TACSR register (ICF2) is forced by hardware to 0. Consequently, the corresponding interrupt cannot be used.
4. The Timer A Output Compare 2 pin is not available (not bonded).
   – In ROM devices:
   The TAOC2HR and TAOC2LR Registers can be used in PWM mode or for timebase generation.
   – In Flash devices:
   The TAOC2HR and TAOC2LR Registers are write only, reading them will return undefined values. Bit 3 of the TACSR register (OCF2) is forced by hardware to 0. Consequently, the corresponding interrupt cannot be used.

Caution: The TAIC2HR and TAIC2LR registers and the ICF2 and OCF2 flags are not present in the ST72F324L but are present in the emulator. For compatibility with the emulator, it is recommended to perform a dummy access (read or write) to the TAIC2LR and TAOC2LR registers to clear the interrupt flags.

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

- Three 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 4). 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 7). 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.

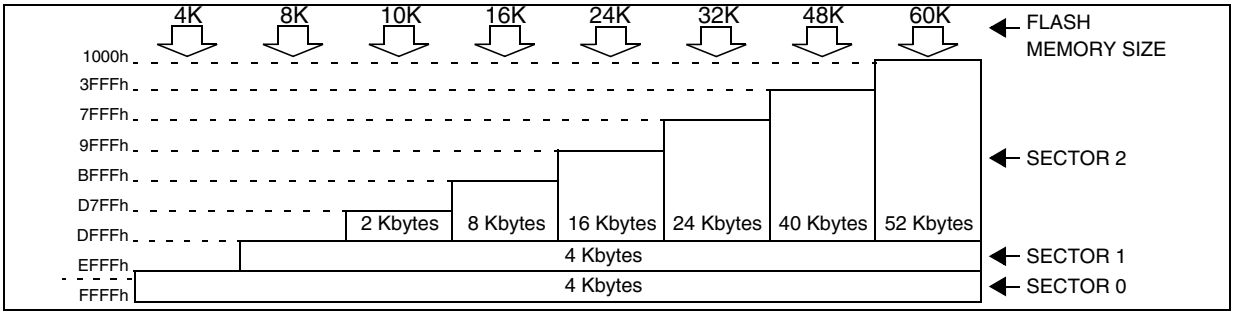
In flash devices, this protection is removed by reprogramming the option. In this case, the entire program memory is first automatically erased.

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 7. Memory Map and Sector Address



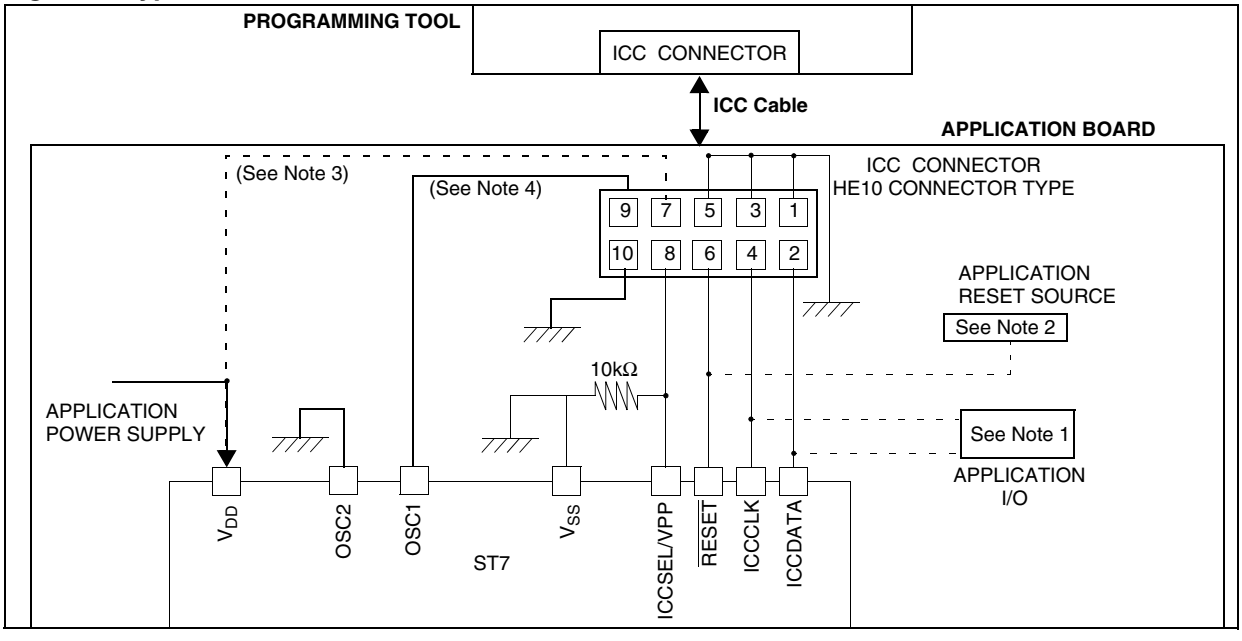
FLASH PROGRAM MEMORY (Cont'd)

4.4 ICC Interface

ICC needs a minimum of 5 and up to 6 pins to be connected to the programming tool (see Figure 8). 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
- V_{DD}: application board power supply (optional, see Figure 8, Note 3)

Figure 8. 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 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. External clock ICC entry mode is mandatory in this device. Pin 9 must be connected to the OSC1 or OSCIN pin of the ST7 and OSC2 must be grounded.

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 using 36-pulse mode.

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 8). 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, USB or CAN 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.

4.7.1 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
0029h	**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 9 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)

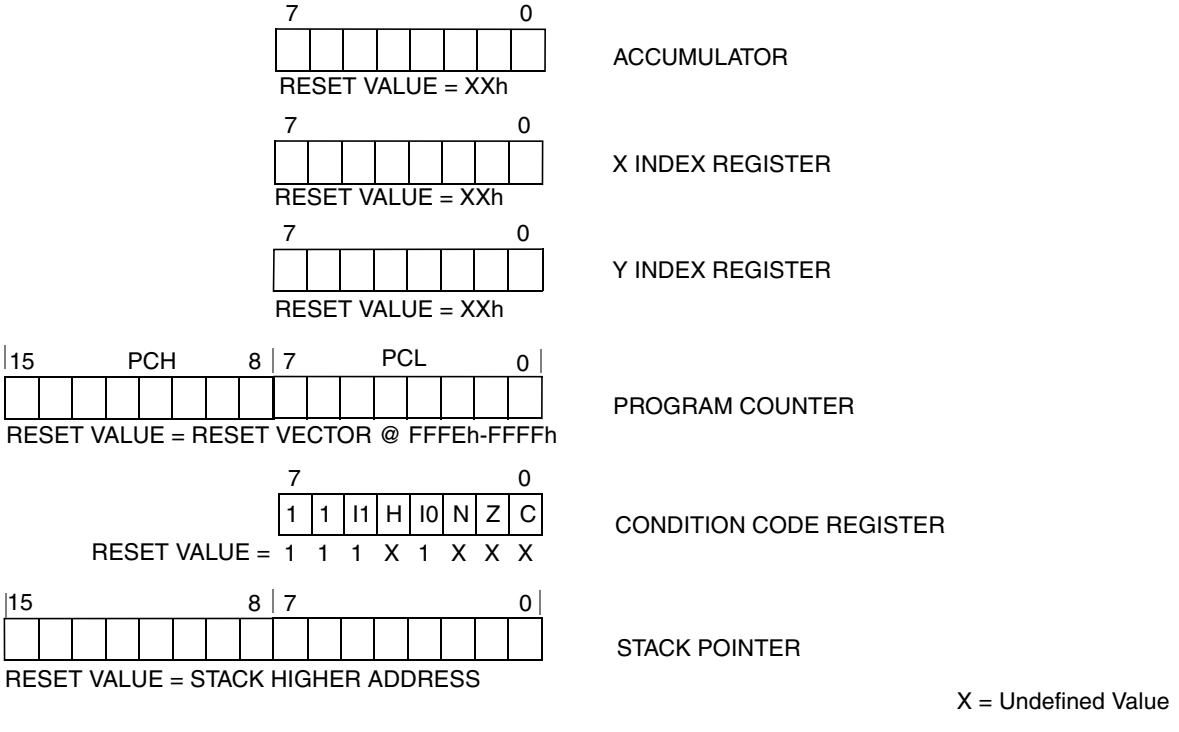
These 8-bit registers are used to create effective addresses or as temporary storage areas for data manipulation. (The Cross-Assembler generates a precede instruction (PRE) to indicate that the following instruction refers to the Y register.)

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 9. 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 10).

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.

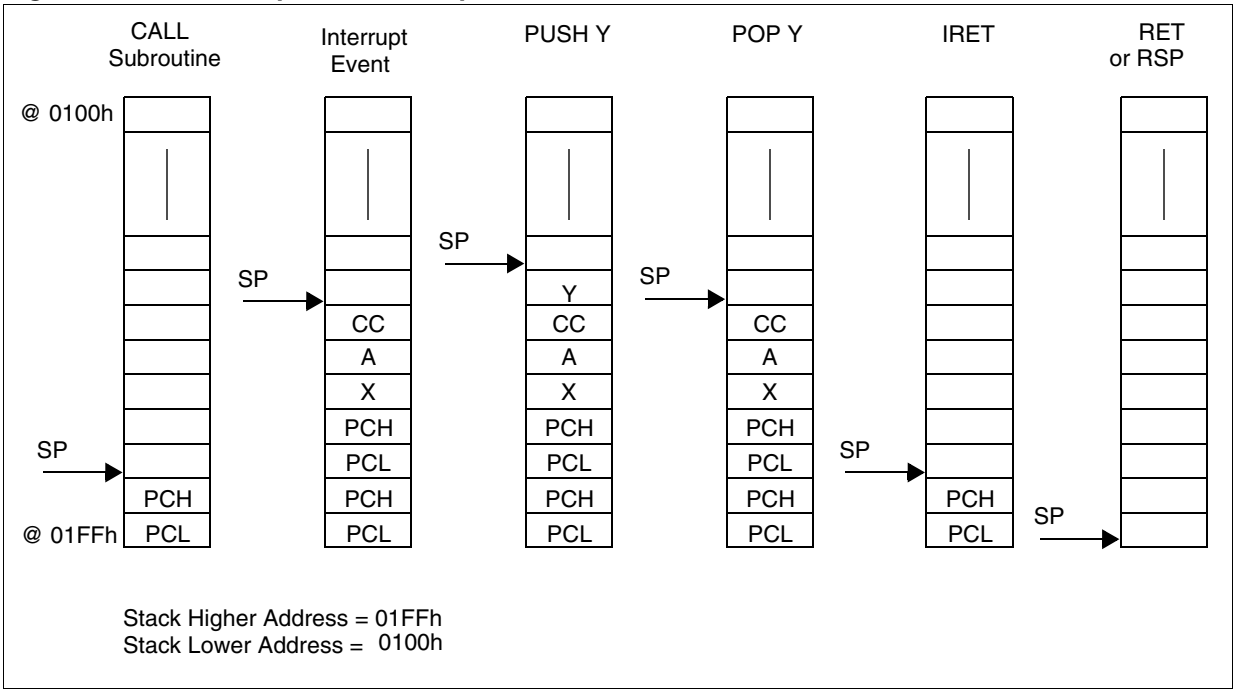
The stack is used to save the return address during a subroutine call and the CPU context during an interrupt. The user may also directly manipulate the stack by means of the PUSH and POP instructions. In the case of an interrupt, the PCL is stored at the first location pointed to by the SP. Then the other registers are stored in the next locations as shown in Figure 10.

– 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 10. 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 (not to be used with internal RC oscillator)
- Reset Sequence Manager (RSM)
- Multi-Oscillator Clock Management (MO)
  - 5 Crystal/Ceramic resonator oscillators
  - 1 Internal RC oscillator

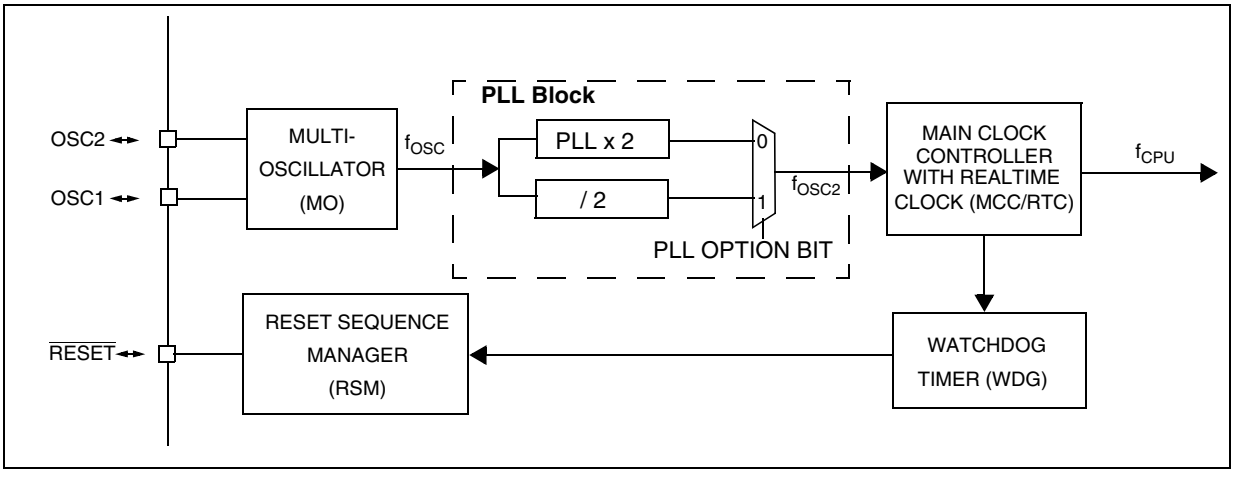
6.1 PHASE LOCKED LOOP

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 Section 6.1 on page 23.

Caution: The PLL must not be used with the internal RC oscillator.

Figure 11. Clock, Reset and Supply Block Diagram



6.2 MULTI-OSCILLATOR (MO)

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

- an external source
- 4 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 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 (>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 4 oscillators with different frequency ranges has to be done by option byte in order to reduce consumption (refer to Section 14.1 on page 140 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.

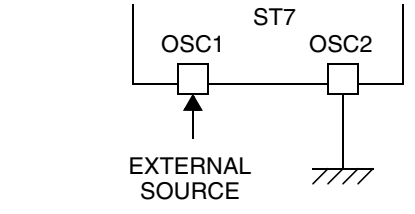
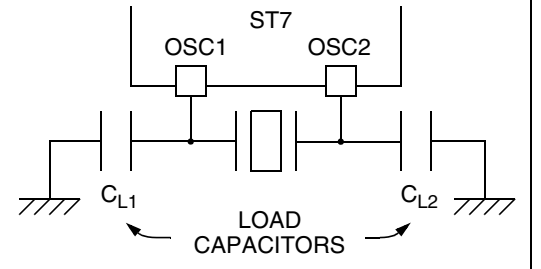
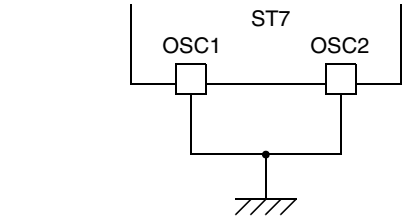
These oscillators are not stopped during the RESET phase to avoid losing time in the oscillator start-up phase.

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.

Table 6. 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 two RESET sources as shown in Figure 13:

- External $\overline{\text{RESET}}$ source pulse
- 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.

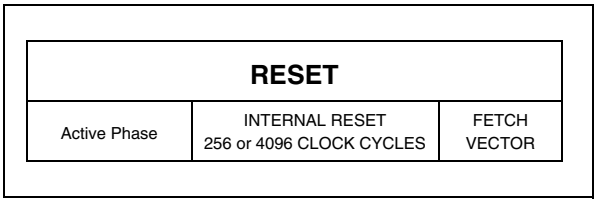
The basic RESET sequence consists of 3 phases as shown in Figure 12:

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

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

6.3.3 External Power-On RESET

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 Watchdog RESET

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 Block Diagram