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

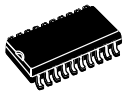
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

ST7DALIF2

8-bit MCU family with single voltage Flash memory, data EEPROM, ADC, timers, SPI, DALI

Features

- **Memories**
  - 8 Kbytes single voltage Flash Program memory with readout protection, In-Circuit Programming and In-Application programming (ICP and IAP). 10K write/erase cycles guaranteed, data retention: 20 years at 55°C.
  - 384 bytes RAM
  - 256 bytes data EEPROM with readout protection. 300K write/erase cycles guaranteed, data retention: 20 yrs at 55°C.
- **Clock, reset and supply management**
  - Enhanced reset system
  - Enhanced low voltage supervisor (LVD) for main supply and an auxiliary voltage detector (AVD) with interrupt capability for implementing safe power-down procedures
  - Clock sources: Internal 1% RC oscillator, crystal/ceramic resonator or external clock
  - Internal 32 MHz input clock for Auto-reload timer
  - Optional x4 or x8 PLL for 4 or 8 MHz internal clock
  - 5 power saving modes: Halt, Active-halt, Wait and Slow, Auto Wake Up From Halt
- **I/O ports**
  - Up to 15 multifunctional bidirectional I/Os
  - 7 high sink outputs
- **4 timers**
  - Configurable watchdog timer
  - Two 8-bit Lite timers with prescaler, watchdog, 1 real-time base and 1 input capture
  - 12-bit auto-reload timer with 4 PWM outputs, input capture and output compare functions



SO20
300”

- **2 communication interfaces**
  - SPI synchronous serial interface
  - DALI communication interface
- **Interrupt management**
  - 10 interrupt vectors plus TRAP and RESET
  - 15 external interrupt lines (on 4 vectors)
- **A/D converter**
  - 7 input channels
  - Fixed gain op-amp
  - 13-bit resolution for 0 to 430 mV (@ 5 V V_{DD})
  - 10-bit resolution for 430 mV to 5 V (@ 5 V V_{DD})
- **Instruction set**
  - 8-bit data manipulation
  - 63 basic instructions with illegal opcode detection
  - 17 main addressing modes
  - 8 x 8 unsigned multiply instructions
- **Development tools**
  - Full hardware/software development package
  - DM (Debug module)

Contents

1 Description

The ST7DALIF2 device is a member of the ST7 microcontroller family designed for DALI applications running from 2.4 to 5.5 V. Different package options offer up to 15 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 a DALI communication interface and an SPI. For power economy, the microcontroller can switch dynamically into, Slow, Wait, Active-halt, Auto Wakeup from Halt or Halt mode when the application is in idle or stand-by state.

Typical applications include consumer, home, office, lighting and industrial products.

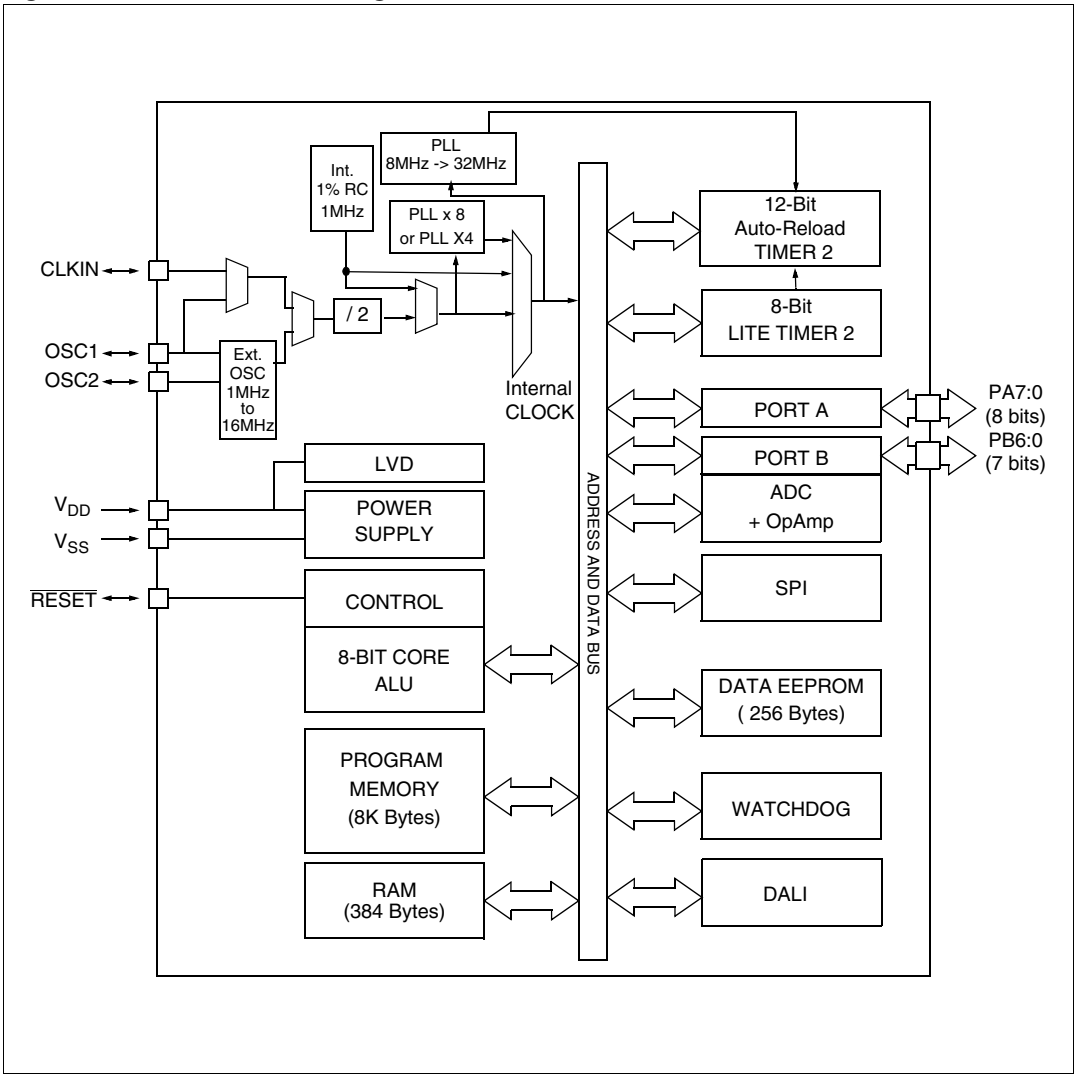
2 Device summary

Table 1. Device summary

Features	ST7DALIF2
Program memory	8 Kbytes
RAM (stack)	384 (128) bytes
Data EEPROM	256 bytes
Peripherals	Lite Timer with Watchdog, Autoreload Timer with 32 MHz input clock, SPI, 10-bit ADC with Op-Amp, DALI
Operating supply	2.4V to 5.5V
CPU frequency	Up to 8 MHz (with external OSC up to 16 MHz and internal 1 MHz RC 1% PLLx8/4 MHz)
Operating temperature	-40°C to +85°C
Packages	SO20 300”

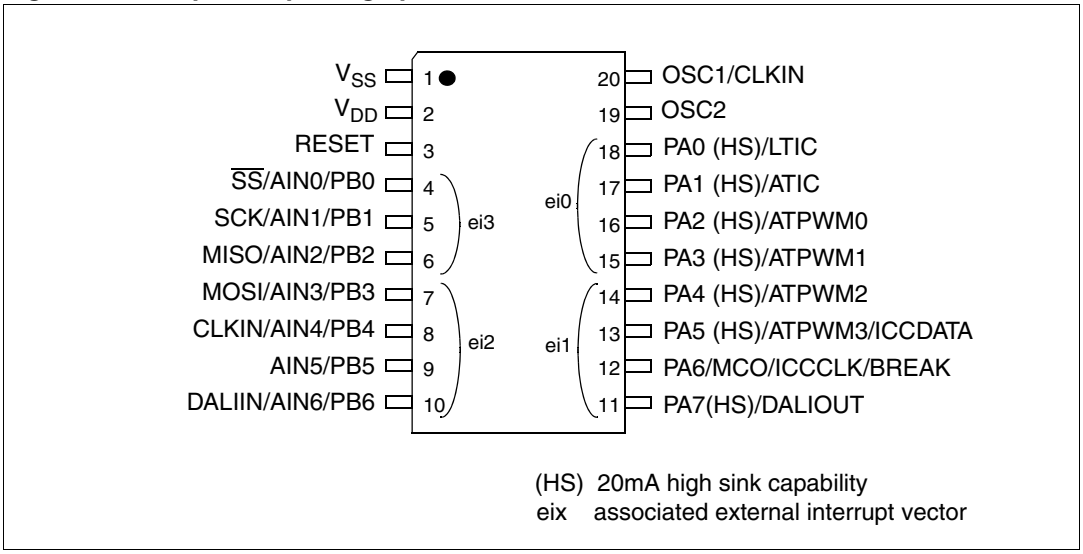
3 Block diagram

Figure 1. General block diagram



4 Pin description

Figure 2. 20-pin SO package pinout



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 input trigger

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

Port and control configuration:

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

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 no.	Pin name	Type	Level		Port / control						Main function (after reset)	Alternate function
					Input				Output			
			Input	Output	float	wpu	int	ana	OD	PP		
1	V_{SS}	S									Ground	
2	V_{DD}	S									Main power supply	
3	$\overline{\text{RESET}}$	I/O	C_T			X			X		Top priority non maskable interrupt (active low)	
4	PB0/AIN0/$\overline{\text{SS}}$	I/O	C_T		X	ei3		X	X	X	**Port B0**	ADC Analog Input 0 or SPI Slave Select (active low) **Caution:** No negative current injection allowed on this pin. For details, refer to *Section 20.2 on page 128*
5	PB1/AIN1/SCK	I/O	C_T		X			X	X	X	**Port B1**	ADC Analog Input 1 or SPI Serial Clock **Caution:** No negative current injection allowed on this pin. For details, refer to *Section 20.2 on page 128*
6	PB2/AIN2/MISO	I/O	C_T		X			X	X	X	**Port B2**	ADC Analog Input 2 or SPI Master In/ Slave Out Data
7	PB3/AIN3/MOSI	I/O	C_T		X	ei2		X	X	X	**Port B3**	ADC Analog Input 3 or SPI Master Out / Slave In Data
8	PB4/AIN4/CLKIN	I/O	C_T		X			X	X	X	**Port B4**	ADC Analog Input 4 or External clock input
9	PB5/AIN5	I/O	C_T		X			X	X	X	**Port B5**	ADC Analog Input 5
10	PB6/AIN6/DALIIN	I/O	C_T		X			X	X	X	**Port B6**	ADC Analog Input 6 or DALI Input

Table 2. Device pin description (continued)

Pin no.	Pin name	Type	Level		Port / control						Main function (after reset)	Alternate function
			Input	Output	Input				Output			
					float	wpu	int	ana	OD	PP		
11	PA7/DALIOUT	I/O	C_T	HS	X	ei1			X	X	**Port A7**	DALI Output
12	PA6 /MCO/ ICCCLK/BREAK	I/O	C_T		X				X	X	**Port A6**	Main Clock Output or In Circuit Communication Clock or External BREAK **Caution:** During normal operation this pin must be pulled- up, internally or externally (external pull-up of 10k mandatory in noisy environment). This is to avoid entering ICC mode unexpectedly during a reset. In the application, even if the pin is configured as output, any reset will put it back in input pull-up.
13	PA5 /ATPWM3/ ICCDATA	I/O	C_T	HS	X				X	X	**Port A5**	Auto-Reload Timer PWM3 or In Circuit Communication Data
14	PA4/ATPWM2	I/O	C_T	HS	X				X	X	**Port A4**	Auto-Reload Timer PWM2
15	PA3/ATPWM1	I/O	C_T	HS	X	ei0			X	X	**Port A3**	Auto-Reload Timer PWM1
16	PA2/ATPWM0	I/O	C_T	HS	X				X	X	**Port A2**	Auto-Reload Timer PWM0
17	PA1/ATIC	I/O	C_T	HS	X				X	X	**Port A1**	Auto-Reload Timer Input Capture
18	PA0 /LTIC	I/O	C_T	HS	X				X	X	**Port A0**	Lite Timer Input Capture
19	OSC2	O									Resonator oscillator inverter output	
20	OSC1/CLKIN	I									Resonator oscillator inverter input or External clock input	

5 Register and memory map

As shown in *Figure 3*, 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, 384 bytes of RAM, 256 bytes of data EEPROM and 8 Kbytes of user program memory. The RAM space includes up to 128 bytes for the stack from 180h to 1FFh.

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

The Flash memory contains two sectors (see *Figure 3*) 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 22.1 on page 161*).

Note: ***IMPORTANT:*** *memory locations marked as "Reserved" must never be accessed. Accessing a reserved area can have unpredictable effects on the device.*

Figure 3. Memory map

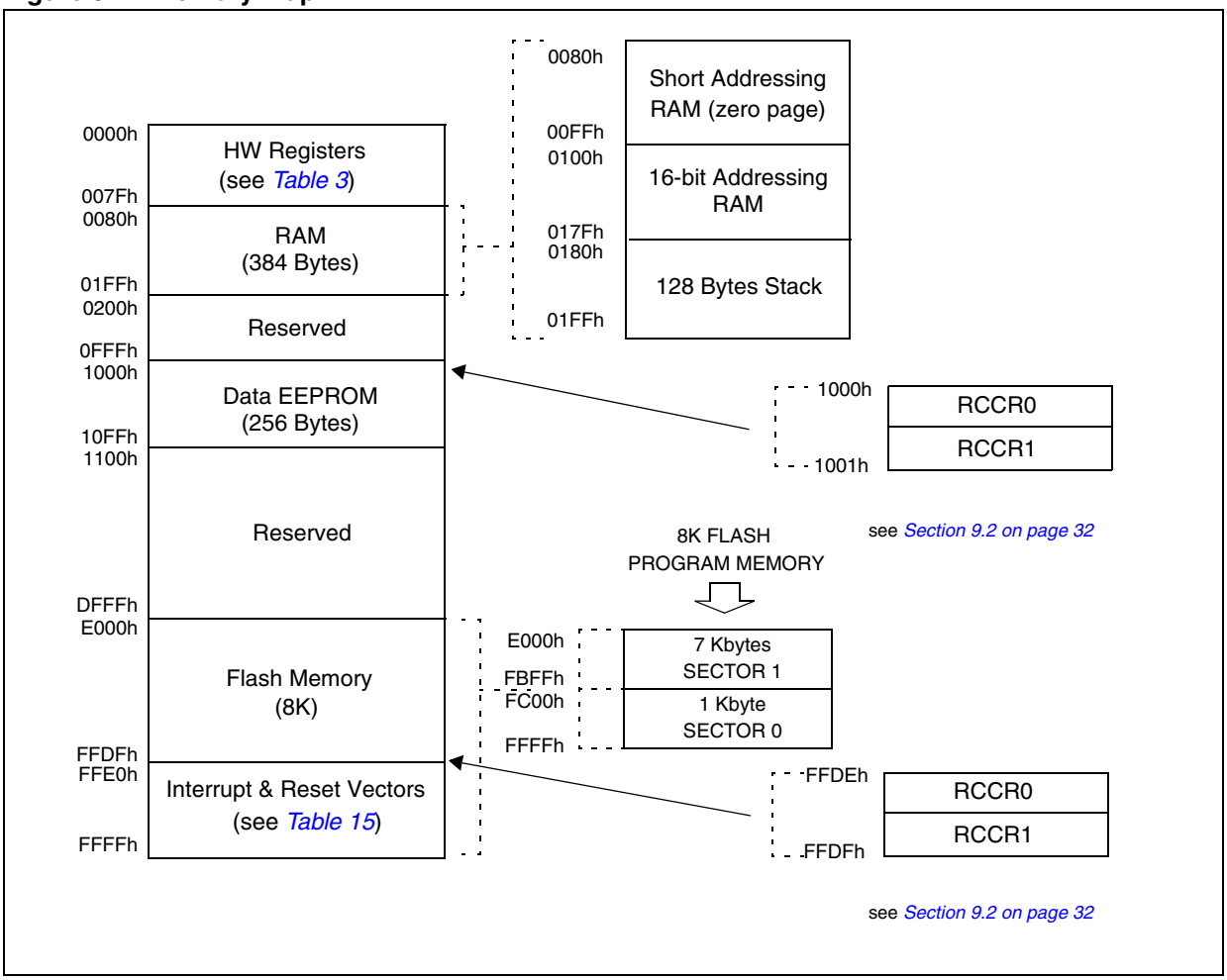


Table 3. Hardware register map

Address	Block	Register label	Register name	Reset status	Remarks
0000h	Port A	PADR	Port A Data Register	FFh(1)	R/W
0001h		PADDR	Port A Data Direction Register	00h	R/W
0002h		PAOR	Port A Option Register	40h	R/W
0003h	Port B	PBDR	Port B Data Register	FFh 1)	R/W
0004h		PBDDR	Port B Data Direction Register	00h	R/W
0005h		PBOR	Port B Option Register	00h	R/W(2)
0006h 0007h	Reserved Area (2 bytes)				
0008h	LITE TIMER 2	LTCSR2	Lite Timer Control/Status Register 2	00h	R/W
0009h		LTARR	Lite Timer Auto-reload Register	00h	R/W
000Ah		LTCNTR	Lite Timer Counter Register	00h	Read Only
000Bh		LTCSR1	Lite Timer Control/Status Register 1	0x00 0000b	R/W
000Ch		LTICR	Lite Timer Input Capture Register	00h	Read Only
000Dh	AUTO-RELOAD TIMER 2	ATCSR	Timer Control/Status Register	0x00 0000b	R/W
000Eh		CNTRH	Counter Register High	00h	Read Only
000Fh		CNTRL	Counter Register Low	00h	Read Only
0010h		ATRH	Auto-Reload Register High	00h	R/W
0011h		ATRL	Auto-Reload Register Low	00h	R/W
0012h		PWMCR	PWM Output Control Register	00h	R/W
0013h		PWM0CSR	PWM 0 Control/Status Register	00h	R/W
0014h		PWM1CSR	PWM 1 Control/Status Register	00h	R/W
0015h		PWM2CSR	PWM 2 Control/Status Register	00h	R/W
0016h		PWM3CSR	PWM 3 Control/Status Register	00h	R/W
0017h		DCR0H	PWM 0 Duty Cycle Register High	00h	R/W
0018h		DCR0L	PWM 0 Duty Cycle Register Low	00h	R/W
0019h		DCR1H	PWM 1 Duty Cycle Register High	00h	R/W
001Ah		DCR1L	PWM 1 Duty Cycle Register Low	00h	R/W
001Bh		DCR2H	PWM 2 Duty Cycle Register High	00h	R/W
001Ch		DCR2L	PWM 2 Duty Cycle Register Low	00h	R/W
001Dh		DCR3H	PWM 3 Duty Cycle Register High	00h	R/W
001Eh		DCR3L	PWM 3 Duty Cycle Register Low	00h	R/W
001Fh		ATICRH	Input Capture Register High	00h	Read Only
0020h		ATICRL	Input Capture Register Low	00h	Read Only
0021h		TRANCR	Transfer Control Register	01h	R/W
0022h		BREAKCR	Break Control Register	00h	R/W
0023h to 002Dh	Reserved area (11 bytes)				
002Eh	WDG	WDGCR	Watchdog Control Register	7Fh	R/W
0002Fh	FLASH	FCSR	Flash Control/Status Register	00h	R/W
00030h	EEPROM	EECSR	Data EEPROM Control/Status Register	00h	R/W
0031h	SPI	SPIDR	SPI Data I/O Register	xxh	R/W
0032h		SPICR	SPI Control Register	0xh	R/W
0033h		SPICSR	SPI Control Status Register	00h	R/W

Table 3. Hardware register map (continued)

Address	Block	Register label	Register name	Reset status	Remarks
0034h 0035h 0036h	ADC	ADCCSR ADCDRH ADCDRL	A/D Control Status Register A/D Data Register High A/D Amplifier Control/Data Low Register	00h xxh 0xh	R/W Read Only R/W
0037h	ITC	EICR	External Interrupt Control Register	00h	R/W
0038h	MCC	MCCSR	Main Clock Control/Status Register	00h	R/W
0039h 003Ah	Clock and Reset	RCCR SICSR	RC oscillator Control Register System Integrity Control/Status Register	FFh 0000 0xx0b	R/W R/W
003Bh	Reserved area (1 byte)				
003Ch	ITC	EISR	External Interrupt Selection Register	0Ch	R/W
003Dh to 003Fh	Reserved area (3 bytes)				
0040h 0041h 0042h 0043h 0044h 0045h	DALI	DCMCLK DCMFA DCMFD DCMBD DCMCR DCMCSR	DALI Clock Register DALI Forward Address Register DALI Forward Data Register DALI Backward Data Register DALI Control Register DALI Control/Status Register	00h 00h 00h 00h 00h 00h	R/W R/W R/W R/W R/W R/W
0046h to 0048h	Reserved area (3 bytes)				
0049h 004Ah	AWU	AWUPR AWUCSR	AWU Prescaler Register AWU Control/Status Register	FFh 00h	R/W R/W
004Bh 004Ch 004Dh 004Eh 004Fh 0050h	DM(3)	DMCR DMSR DMBK1H DMBK1L DMBK2H DMBK2L	DM Control Register DM Status Register DM Breakpoint Register 1 High DM Breakpoint Register 1 Low DM Breakpoint Register 2 High DM Breakpoint Register 2 Low	00h 00h 00h 00h 00h 00h	R/W R/W R/W R/W R/W R/W
0051h to 007Fh	Reserved area (47 bytes)				

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 a description of the Debug Module registers, see ST7 ICC Protocol Reference Manual.

Legend: x=undefined, R/W=read/write

6 Flash program memory

6.1 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 organization allows each sector to be erased and reprogrammed without affecting other sectors.

6.2 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
- Readout and write protection

6.3 Programming modes

The ST7 can be programmed in three different ways:

- Insertion in a programming tool. In this mode, FLASH sectors 0 and 1, option byte row and data EEPROM (if present) can be programmed or erased.
- In-Circuit Programming. In this mode, FLASH sectors 0 and 1, option byte row and data EEPROM (if present) can be programmed or erased without removing the device from the application board.
- In-Application Programming. In this mode, sector 1 and data EEPROM (if present) can be programmed or erased without removing the device from the application board and while the application is running.

6.3.1 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).

6.3.2 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.

6.4 ICC interface

ICP needs a minimum of 4 and up to 6 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
- CLKIN/PB4: main clock input for external source
- V_{DD}: application board power supply (optional, see Note 3)

Note: 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.

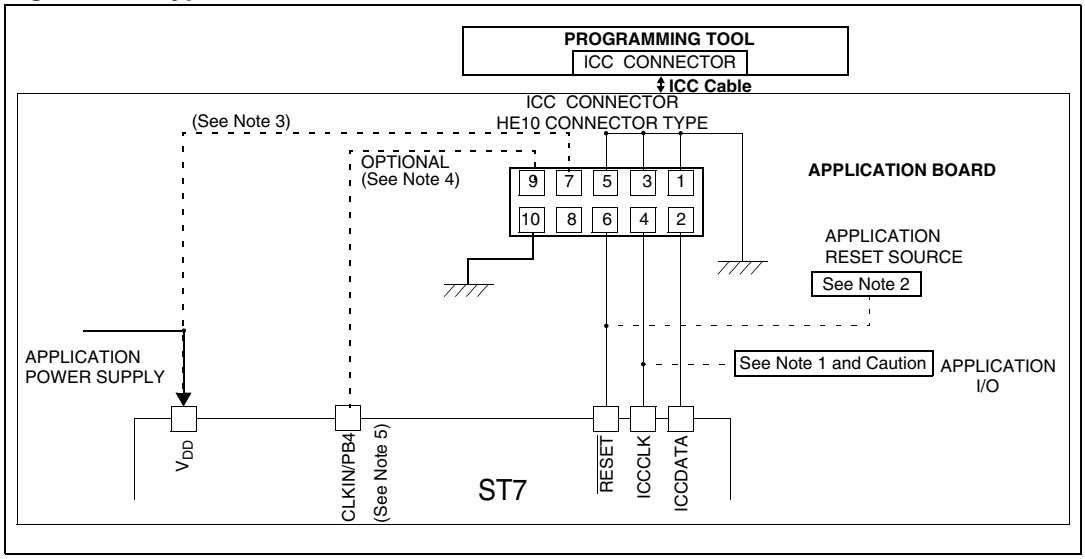
4 Pin 9 has to be connected to the CLKIN/PB4 pin of the ST7 when the clock is not available in the application or if the selected clock option is not programmed in the option byte. ST7 devices with multi-oscillator capability need to have OSC1 and OSC2 grounded in this case.

5 With any programming tool, while the ICP option is disabled, the external clock has to be provided on PB4.

Caution: During normal operation the ICCCLK pin must be pulled- up, internally or externally (external pull-up of 10k mandatory in noisy environment). This is to avoid entering ICC

mode unexpectedly during a reset. In the application, even if the pin is configured as output, any reset will put it back in input pull-up.

Figure 4. Typical ICC interface



6.5 Memory protection

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

6.5.1 Readout protection

Readout 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. Both program and data E^2 memory are protected.

In flash devices, this protection is removed by reprogramming the option. In this case, both program and data E^2 memory are automatically erased and the device can be reprogrammed.

Readout 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.

6.5.2 Flash write/erase protection

Write/erase protection, when set, makes it impossible to both overwrite and erase program memory. It does not apply to E^2 data. Its purpose is to provide advanced security to applications and prevent any change being made to the memory content.

Caution: 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.

6.6 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.

6.7 Register description

6.7.1 Flash control/status register (FCSR)

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.

1st RASS Key: 0101 0110 (56h)
2nd RASS Key: 1010 1110 (AEh)

FCSR Reset value:0000 0000 (00h)

7	6	5	4	3	2	1	0
0	0	0	0	0	OPT	LAT	PGM
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 4. Flash control/status register address and reset value

Address (Hex)	Register label	7	6	5	4	3	2	1	0
002Fh	FCSR reset value	0	0	0	0	0	0	0	0

7 Data EEPROM

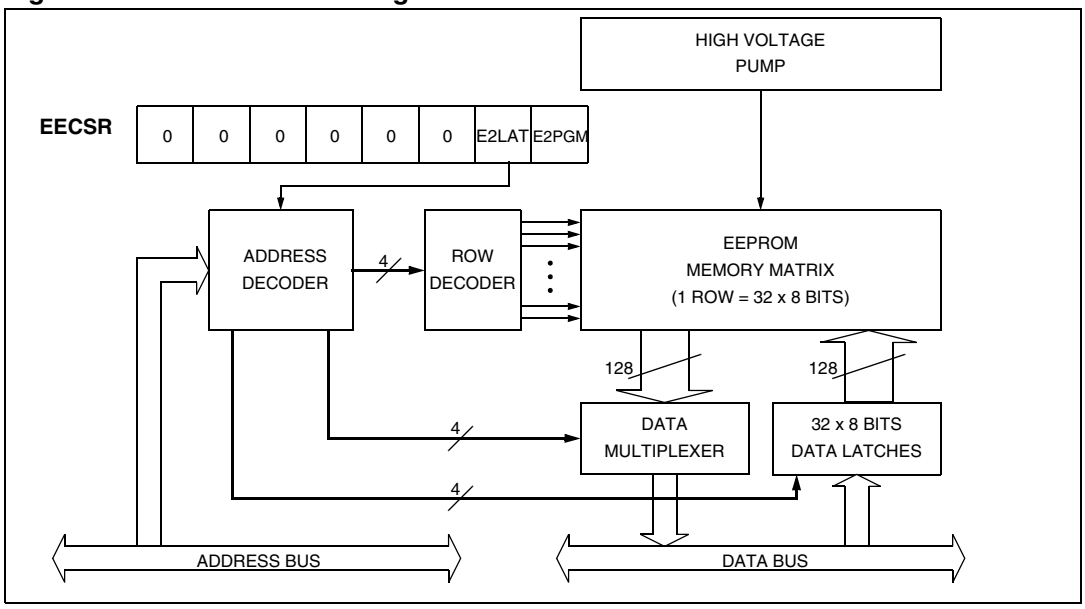
7.1 Introduction

The Electrically Erasable Programmable Read Only Memory can be used as a non volatile back-up for storing data. Using the EEPROM requires a basic access protocol described in this chapter.

7.2 Main features

- Up to 32 Bytes programmed in the same cycle
- EEPROM mono-voltage (charge pump)
- Chained erase and programming cycles
- Internal control of the global programming cycle duration
- Wait mode management
- Readout protection

Figure 5. EEPROM block diagram



7.3 Memory access

The Data EEPROM memory read/write access modes are controlled by the E2LAT bit of the EEPROM Control/Status register (EECSR). The flowchart in *Figure 6* describes these different memory access modes.

Read operation (E2LAT=0)

The EEPROM can be read as a normal ROM location when the E2LAT bit of the EECSR register is cleared.

On this device, Data EEPROM can also be used to execute machine code. Take care not to write to the Data EEPROM while executing from it. This would result in an unexpected code being executed.

Write operation (E2LAT=1)

To access the write mode, the E2LAT bit has to be set by software (the E2PGM bit remains cleared). When a write access to the EEPROM area occurs, the value is latched inside the 32 data latches according to its address.

When PGM bit is set by the software, all the previous bytes written in the data latches (up to 32) are programmed in the EEPROM cells. The effective high address (row) is determined by the last EEPROM write sequence. To avoid wrong programming, the user must take care that all the bytes written between two programming sequences have the same high address: only the five Least Significant Bits of the address can change.

At the end of the programming cycle, the PGM and LAT bits are cleared simultaneously.

Note: Care should be taken during the programming cycle. Writing to the same memory location will over-program the memory (logical AND between the two write access data result) because the data latches are only cleared at the end of the programming cycle and by the falling edge of the E2LAT bit.
It is not possible to read the latched data.
This note is illustrated by Figure 8.

Figure 6. Data EEPROM programming flowchart

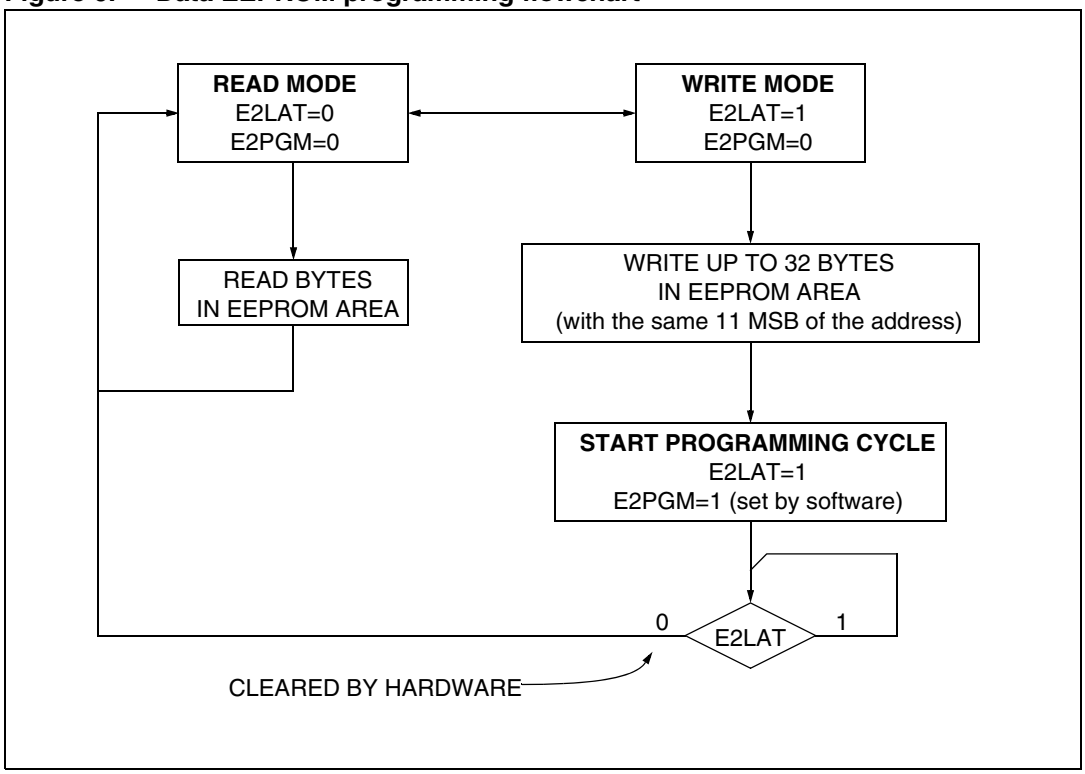
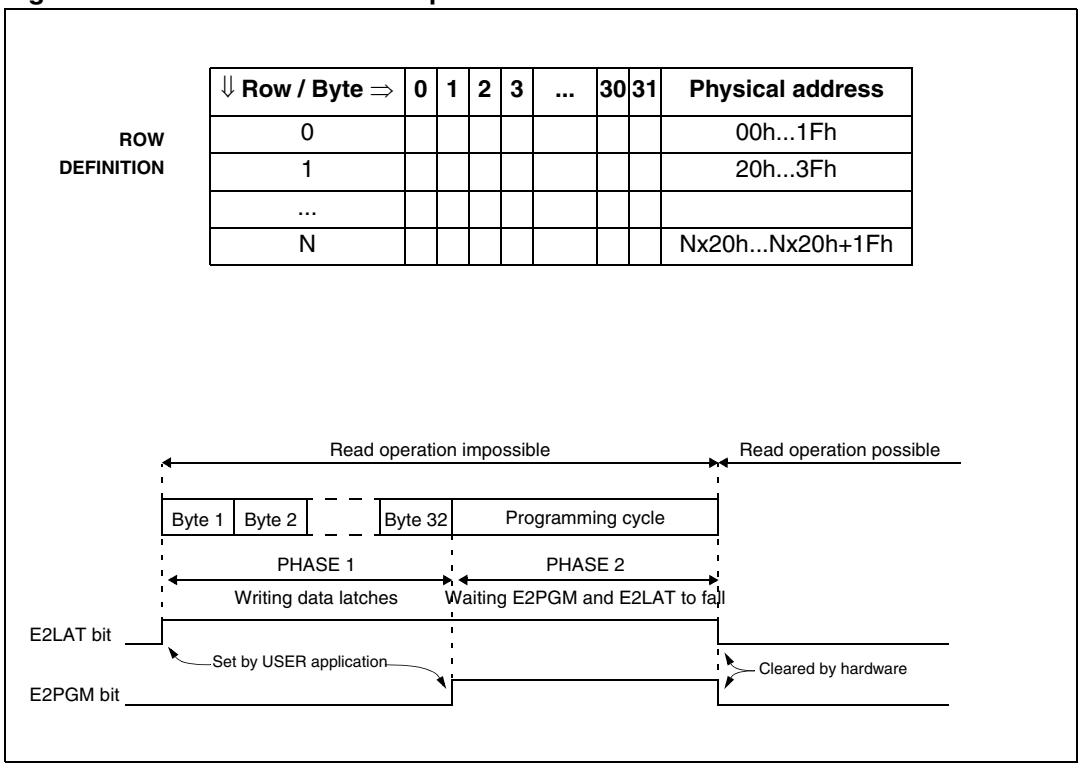


Figure 7. Data EEPROM write operation



Note: If a programming cycle is interrupted (by a reset action), the integrity of the data in memory is not guaranteed.

7.4 Power saving modes

Wait mode

The data EEPROM can enter Wait mode on execution of the WFI instruction of the microcontroller or when the microcontroller enters Active-Halt mode.The data EEPROM will immediately enter this mode if there is no programming in progress, otherwise the data EEPROM will finish the cycle and then enter Wait mode.

Active-halt mode

Refer to Wait mode.

Halt mode

The data EEPROM immediately enters Halt mode if the microcontroller executes the HALT instruction. Therefore the EEPROM will stop the function in progress, and data may be corrupted.

7.5 Access error handling

If a read access occurs while E2LAT=1, then the data bus will not be driven.

If a write access occurs while E2LAT=0, then the data on the bus will not be latched.