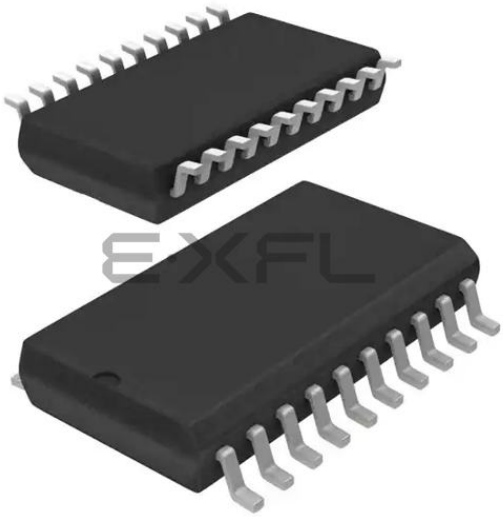


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ST7
Core Size	8-Bit
Speed	8MHz
Connectivity	LINbusSCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	15
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 7x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SO
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st7fl39f2mae

3 Register and memory map

As shown in *Figure 4*, the MCU can address 64 Kbytes 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 up to 8 Kbytes of user program memory. The RAM space includes up to 128 bytes for the stack from 180h to 1FFh.

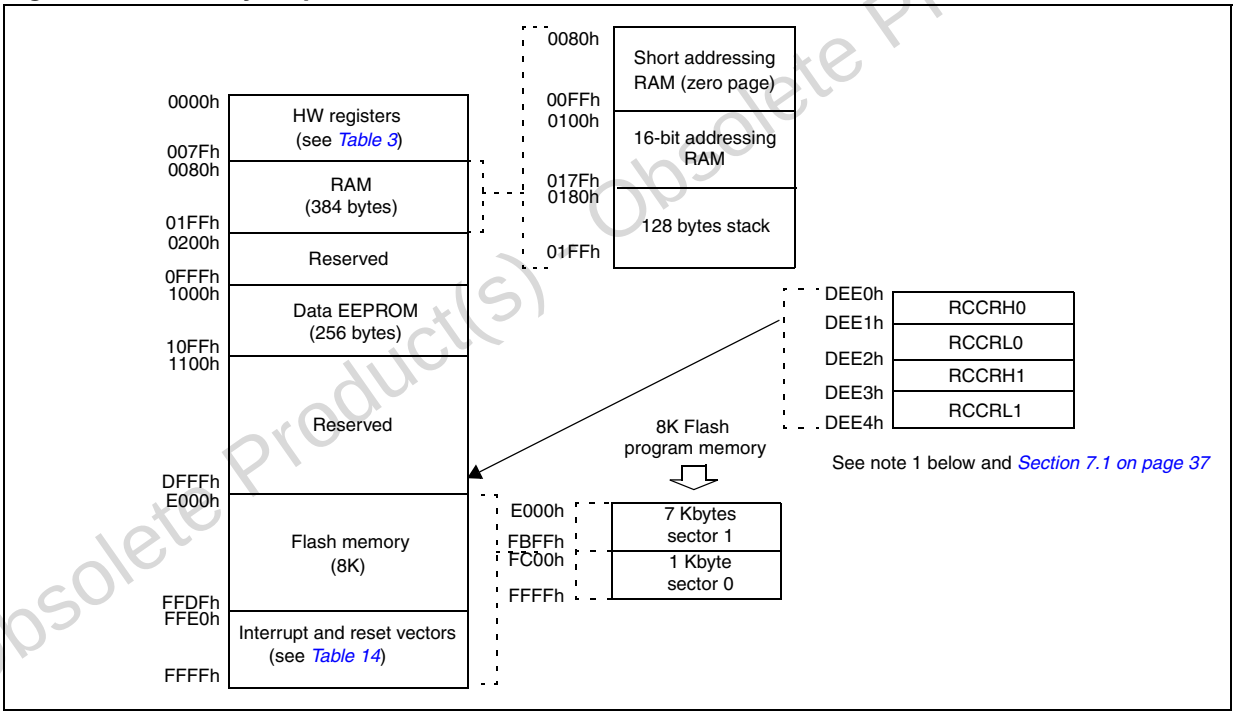
The highest address bytes contain the user reset and interrupt vectors.

The Flash memory contains two sectors (see *Figure 4*) mapped in the upper part of the ST7 addressing space so the reset and interrupt vectors are located in Sector 0 (F000h-FFFFh).

The size of Flash sector 0 and other device options are configurable by option byte (refer to *Section 15.2: Option bytes on page 215*).

Note: Memory locations marked as 'Reserved' must never be accessed. Accessing a reserved area can have unpredictable effects on the device.

Figure 4. Memory map



1. *DEE0h, DEE1h, DEE2h and DEE3h addresses are located in a reserved area but are special bytes containing also the RC calibration values which are read-accessible only in user mode. If all the EEPROM data or Flash space (including the RC calibration values locations) has been erased (after the readout protection removal), then the RC calibration values can still be obtained through these four addresses.*

5.4 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 immediately enters this mode if there is no programming in progress, otherwise the data EEPROM finishes the cycle and then enters 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 stops the function in progress, and data may be corrupted.

5.5 Access error handling

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

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

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

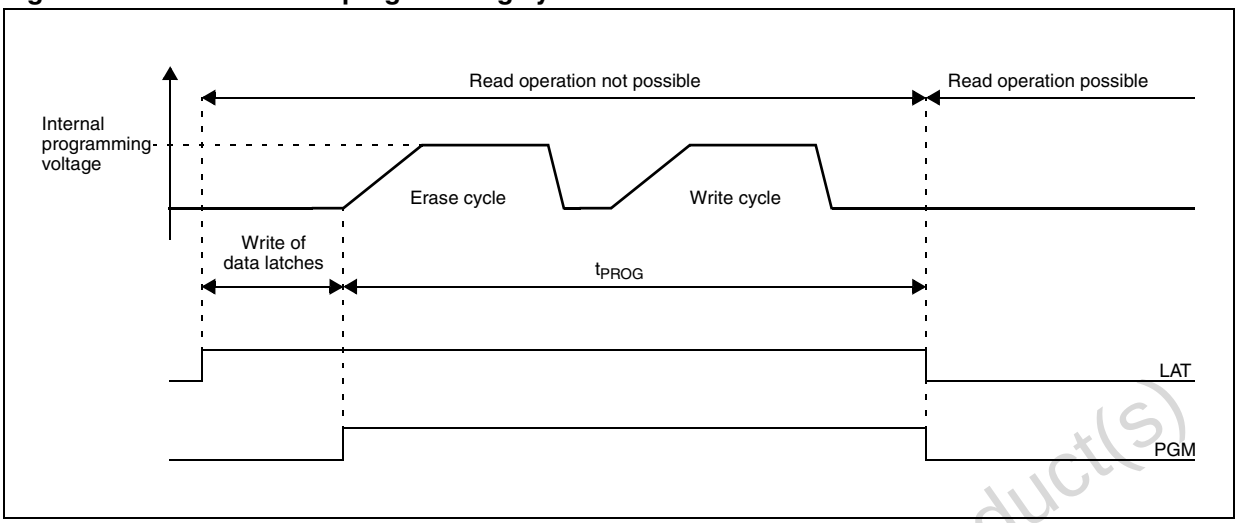
5.6 Data EEPROM readout protection

The readout protection is enabled through an option bit (see *Section 15.2: Option bytes on page 215*).

When this option is selected, the programs and data stored in the EEPROM memory are protected against readout (including a rewrite protection). In Flash devices, when this protection is removed by reprogramming the option byte, the entire program memory and EEPROM is first automatically erased.

Note: Both program memory and data EEPROM are protected using the same option bit.

Figure 9. Data EEPROM programming cycle



5.7 Register description

EEPROM control/status register (EECSR)

EECSR Reset value: 0000 0000 (00h)

7	6	5	4	3	2	1	0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	E2LAT	E2PGM
-	-	-	-	-	-	R/W	R/W

Table 4. EECSR register description

Bit	Bit name	Function
7:2	-	Reserved, forced by hardware to 0
1	E2LAT	Latch access transfer This bit is set by software. It is cleared by hardware at the end of the programming cycle. It can only be cleared by software if the E2PGM bit is cleared. 0: Read mode 1: Write mode
0	E2PGM	Programming control and status This bit is set by software to begin the programming cycle. At the end of the programming cycle, this bit is cleared by hardware. 0: Programming finished or not yet started 1: Programming cycle is in progress *Note: If the E2PGM bit is cleared during the programming cycle, the memory data is not guaranteed*

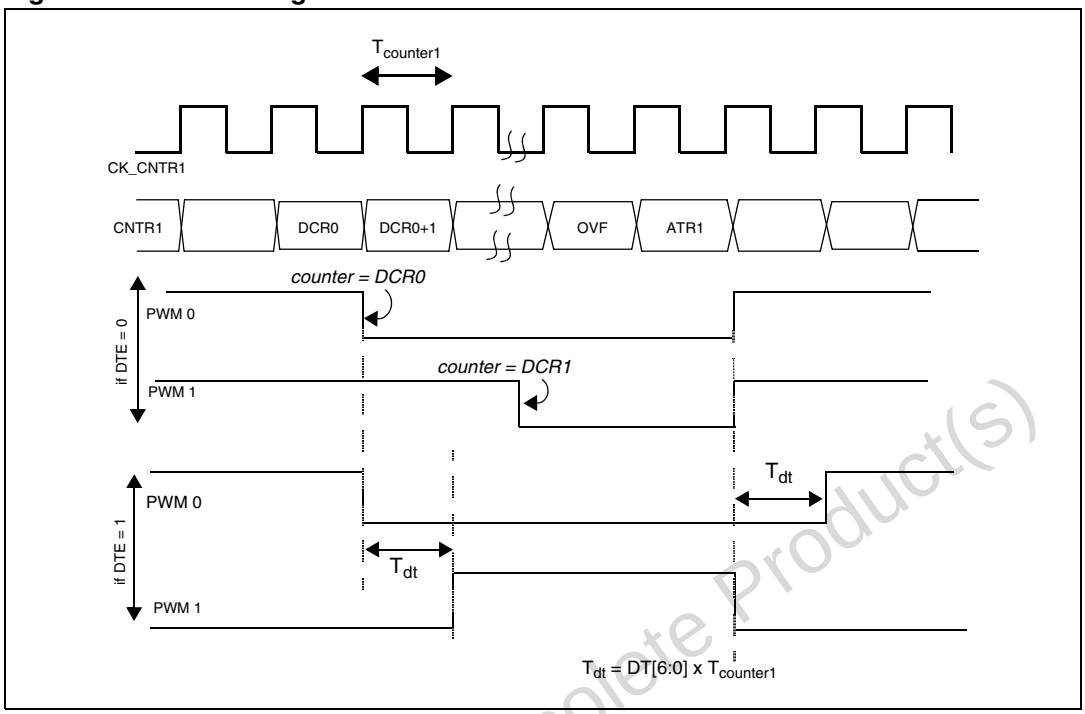
Table 5. Data EEPROM register map and reset values

Address (Hex.)	Register label	7	6	5	4	3	2	1	0
0030h	EECSR Reset value	0	0	0	0	0	0	E2LAT 0	E2PGM 0

Table 13. SICSR register description (continued)

Bit	Bit name	Function
4	WDGRF	Watchdog reset flag This bit indicates that the last reset was generated by the watchdog peripheral. It is set by hardware (watchdog reset) and cleared by software (by reading SICSR register) or an LVD reset (to ensure a stable cleared state of the WDGRF flag when CPU starts). Combined with the LVDRF flag information, the flag description is given as follows: 00 (LVDRF, WDGRF): Reset sources = External $\overline{\text{RESET}}$ pin 01 (LVDRF, WDGRF): Reset sources = Watchdog 1X (LVDRF, WDGRF): Reset sources = LVD
3	LOCKED	PLL locked flag This bit is set and cleared by hardware. It is set automatically when the PLL reaches its operating frequency. 0: PLL not locked 1: PLL locked
2	LVDRF	LVD reset flag This bit indicates that the last reset was generated by the LVD block. It is set by hardware (LVD reset) and cleared by software (by reading). When the LVD is disabled by option byte, the LVDRF bit value is undefined. *Note: The LVDRF flag is not cleared when another reset type occurs (external or watchdog), the LVDRF flag remains set to keep trace of the original failure. In this case, a watchdog reset can be detected by software while an external reset can not.*
1	AVDF	Voltage detector flag This read-only bit is set and cleared by hardware. If the AVDIE bit is set, an interrupt request is generated when the AVDF bit is set. Refer to *Figure 20* and to *Monitoring the VDD main supply on page 48* for additional details. 0: V_{DD} over AVD threshold 1: V_{DD} under AVD threshold
0	AVDIE	Voltage detector interrupt enable This bit is set and cleared by software. It enables an interrupt to be generated when the AVDF flag is set. The pending interrupt information is automatically cleared when software enters the AVD interrupt routine. 0: AVD interrupt disabled 1: AVD interrupt enabled

Figure 40. Dead time generation



In the above example, when the DTE bit is set:

- PWM goes low at DCR0 match and goes high at ATR1 + T_{dt}
- PWM1 goes high at DCR0 + T_{dt} and goes low at ATR match.

With this programmable delay (T_{dt}), the PWM0 and PWM1 signals which are generated are not overlapped.

Break function

The break function can be used to perform an emergency shutdown of the application being driven by the PWM signals.

The break function is activated by the external BREAK pin (active low). In order to use the break pin it must be previously enabled by software setting the BPEN bit in the BREAKCR register.

When a low level is detected on the break pin, the BA bit is set and the break function is activated. In this case, the four PWM signals are stopped.

Software can set the BA bit to activate the break function without using the break pin.

When a break function is activated (BA bit = 1 and BREN1/BREN2 = 1):

- The break pattern (PWM[3:0] bits in the BREAKCR is forced directly on the PWMx output pins (after the inverter)
- The 12-bit PWM counter CNTR1 is put to its reset value, that is, 00h
- The 12-bit PWM counter CNTR2 is put to its reset value, that is 00h
- ATR1, ATR2, preload and active DCRx are put to their reset values
- The PWMCR register is reset
- Counters stop counting

Table 47. Register map and reset values (continued)

Add (Hex.)	Register label	7	6	5	4	3	2	1	0
22	BREAKCR Reset value	0	0	BA 0	BPEN 0	PWM3 0	PWM2 0	PWM1 0	PWM0 0
23	ATR2H Reset value	0	0	0	0	ATR11 0	ATR10 0	ATR9 0	ATR8 0
24	ATR2L Reset value	ATR7 0	ATR6 0	ATR5 0	ATR4 0	ATR3 0	ATR2 0	ATR1 0	ATR0 0
25	DTGR Reset value	DTE 0	DT6 0	DT5 0	DT4 0	DT3 0	DT2 0	DT1 0	DT0 0

11.3 Lite timer 2 (LT2)

11.3.1 Introduction

The lite timer is used for general-purpose timing functions. It is based on two free-running 8-bit upcounters and an 8-bit input capture register.

11.3.2 Main features

- Real-time clock (RTC)
  - One 8-bit upcounter 1 ms or 2 ms timebase period (@ 8 MHz f_{OSC})
  - One 8-bit upcounter with autoreload and programmable timebase period from 4µs to 1.024ms in 4µs increments (@ 8 MHz f_{OSC})
  - 2 maskable timebase interrupts
- Input capture
  - 8-bit input capture register (LTICR)
  - Maskable interrupt with wakeup from halt mode capability

11.3.3 Functional description

Timebase counter 1

The 8-bit value of counter 1 cannot be read or written by software. After an MCU reset, it starts incrementing from 0 at a frequency of $f_{OSC}/32$. An overflow event occurs when the counter rolls over from F9h to 00h. If f_{OSC} = 8 MHz, then the time period between two counter overflow events is 1 ms. This period can be doubled by setting the TB bit in the LTCSR1 register.

When counter 1 overflows, the TB1F bit is set by hardware and an interrupt request is generated if the TB1IE bit is set. The TB1F bit is cleared by software reading the LTCSR1 register.

Timebase counter 2

Counter 2 is an 8-bit autoreload upcounter. It can be read by accessing the LTCNTR register. After an MCU reset, it increments at a frequency of $f_{OSC}/32$ starting from the value stored in the LTARR register. A counter overflow event occurs when the counter rolls over from FFh to the LTARR reload value. Software can write a new value at anytime in the LTARR register, this value will be automatically loaded in the counter when the next overflow occurs.

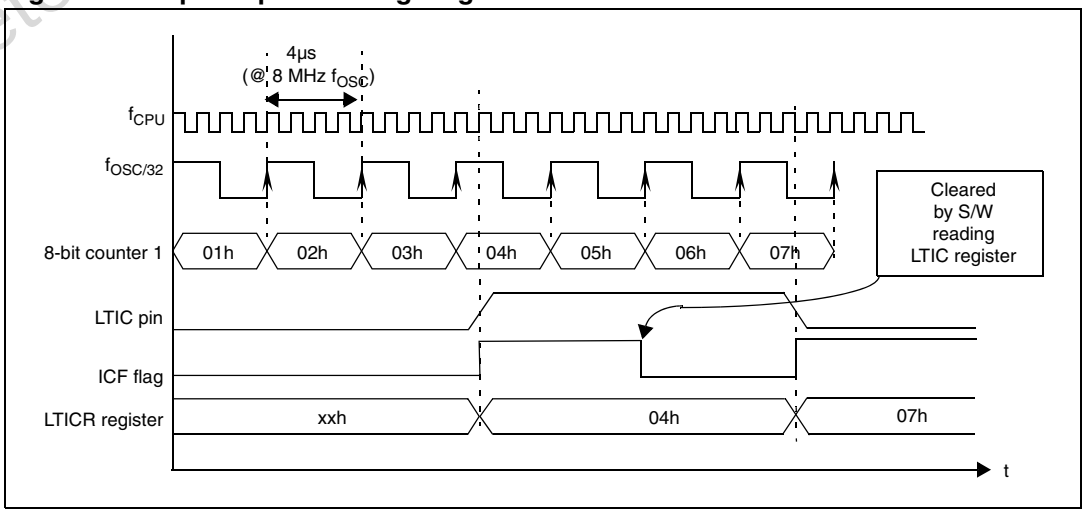
When counter 2 overflows, the TB2F bit in the LTCSR2 register is set by hardware and an interrupt request is generated if the TB2IE bit is set. The TB2F bit is cleared by software reading the LTCSR2 register.

Input capture

The 8-bit input capture register is used to latch the free-running upcounter (counter 1) 1 after a rising or falling edge is detected on the LTIC pin. When an input capture occurs, the ICF bit is set and the LTICR register contains the value of counter 1. An interrupt is generated if the ICIE bit is set. The ICF bit is cleared by reading the LTICR register.

The LTICR is a read-only register and always contains the data from the last input capture. Input capture is inhibited if the ICF bit is set.

Figure 48. Input capture timing diagram



Note: 1 While the SPIF bit is set, all writes to the SPIDR register are inhibited until the SPICSR register is read.

2 The SPIF bit can be cleared during a second transmission; however, it must be cleared before the second SPIF bit in order to prevent an overrun condition (see Overrun condition (OVR) on page 115).

11.4.4 Clock phase and clock polarity

Four possible timing relationships may be chosen by software, using the CPOL and CPHA bits (see *Figure 53: Data clock timing diagram on page 114*).

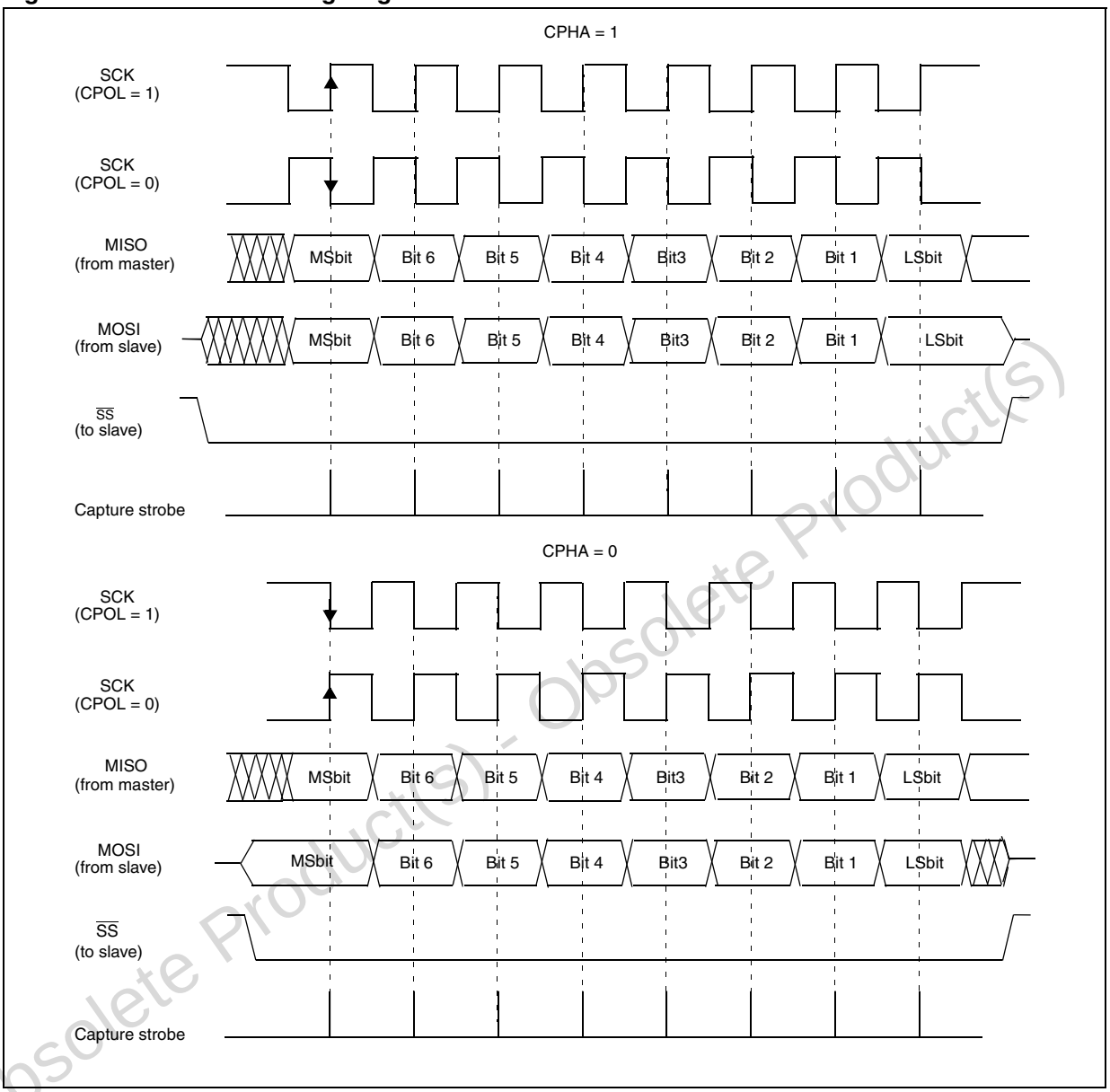
Note: The idle state of SCK must correspond to the polarity selected in the SPICSR register (by pulling up SCK if CPOL = 1 or pulling down SCK if CPOL = 0).

The combination of the CPOL clock polarity and CPHA (clock phase) bits selects the data capture clock edge.

Figure 53: Data clock timing diagram on page 114 shows an SPI transfer with the four combinations of the CPHA and CPOL bits. The diagram may be interpreted as a master or slave timing diagram where the SCK pin, the MISO pin and the MOSI pin are directly connected between the master and the slave device.

Note: If CPOL is changed at the communication byte boundaries, the SPI must be disabled by resetting the SPE bit.

Figure 53. Data clock timing diagram



1. This figure should not be used as a replacement for parametric information. Refer to *Section 13: Electrical characteristics*.

Table 60. SPI register map and reset values

Address (Hex.)	Register label	7	6	5	4	3	2	1	0
0031h	SPIDR Reset Value	MSB x	x	x	x	x	x	x	LSB x
0032h	SPICR Reset Value	SPIE 0	SPE 0	SPR2 0	MSTR 0	CPOL x	CPHA x	SPR1 x	SPR0 x
0033h	SPICSR Reset Value	SPIF 0	WCOL 0	OVR 0	MODF 0	0	SOD 0	SSM 0	SSI 0

11.5 LINSCI serial communication interface (LIN master/slave)

11.5.1 Introduction

The serial communications interface (SCI) offers a flexible means of full-duplex data exchange with external equipment requiring an industry standard NRZ asynchronous serial data format. The SCI offers a very wide range of baud rates using two baud rate generator systems.

The LIN-dedicated features support the LIN (local interconnect network) protocol for both master and slave nodes.

This chapter is divided into SCI Mode and LIN mode sections. For information on general SCI communications, refer to the SCI mode section. For LIN applications, refer to both the SCI mode and LIN mode sections.

Setting the RWU bit by software puts the SCI in sleep mode:

All the reception status bits cannot be set.

All the receive interrupts are inhibited.

A muted receiver may be woken up in one of the following ways:

- by idle line detection if the WAKE bit is reset,
- by address mark detection if the WAKE bit is set.

Idle line detection

The receiver wakes up by idle line detection when the receive line has recognized an idle line. Then the RWU bit is reset by hardware but the IDLE bit is not set.

This feature is useful in a multiprocessor system when the first characters of the message determine the address and when each message ends by an idle line: As soon as the line becomes idle, every receiver is awakened and the first characters of the message are analysed which indicates the addressed receiver. The receivers which are not addressed set the RWU bit to enter in mute mode. Consequently, they will not read the next characters constituting the next part of the message. At the end of the message, an idle line is sent by the transmitter: this wakes up every receiver which are ready to analyse the addressing characters of the new message.

In such a system, the inter-characters space must be smaller than the idle time.

Address mark detection

The receiver wakes up by address mark detection when it receives a ‘1’ as the most significant bit of a word, thus indicating that the message is an address. The reception of this particular word wakes up the receiver, resets the RWU bit and sets the RDRF bit, which allows the receiver to receive this word normally and to use it as an address word.

This feature is useful in a multiprocessor system when the most significant bit of each character (except for the break character) is reserved for address detection. As soon as the receivers receive an address character (most significant bit = ’1’), the receivers are woken up. The receivers which are not addressed set RWU bit to enter in mute mode. Consequently, they will not treat the next characters constituting the next part of the message.

Parity control

Hardware byte parity control (generation of parity bit in transmission and parity checking in reception) can be enabled by setting the PCE bit in the SCICR1 register. Depending on the character format defined by the M bit, the possible SCI character formats are as listed in *Table 61*.

Note: In case of wake-up by an address mark, the MSB bit of the data is taken into account and not the parity bit

Table 61. Character formats(1)

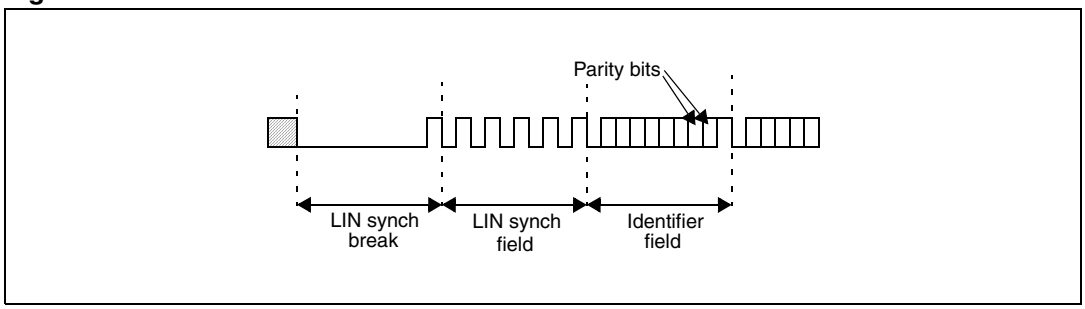
M bit	PCE bit	Character format
0	0	I SB I 8 bit data I STB I
	1	I SB I 7-bit data I PB I STB I
1	0	I SB I 9-bit data I STB I
	1	I SB I 8-bit data I PB I STB I

1. Legend: SB = start bit, STB = stop bit, PB = parity bit

Table 64. SCISR register description (continued)

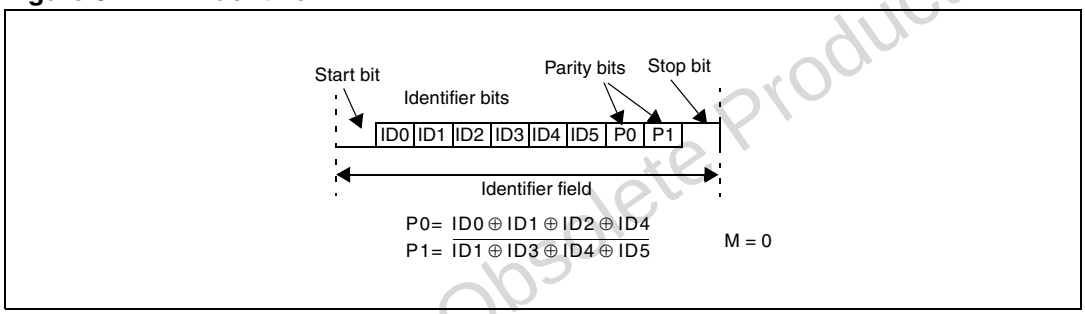
Bit	Bit name	Function
4	IDLE	Idle line detect This bit is set by hardware when an idle line is detected. An interrupt is generated if the ILIE = 1 in the SCICR2 register. It is cleared by a software sequence (an access to the SCISR register followed by a read to the SCIDR register). 0: No idle line is detected 1: Idle line is detected *Note: The IDLE bit is not set again until the RDRF bit is set (i.e. a new idle line occurs).*
3	OR	Overrun error This bit is set by hardware when the word currently being received in the shift register is ready to be transferred into the RDR register while RDRF = 1. An interrupt is generated if RIE = 1 in the SCICR2 register. It is cleared by a software sequence (an access to the SCISR register followed by a read to the SCIDR register). 0: No overrun error 1: Overrun error is detected *Note: When the IDLE bit is set the RDR register content is not lost but the shift register is overwritten.*
2	NF	Noise flag This bit is set by hardware when noise is detected on a received character. It is cleared by a software sequence (an access to the SCISR register followed by a read to the SCIDR register). 0: No noise is detected 1: Noise is detected *Note: The NF bit does not generate an interrupt as it appears at the same time as the RDRF bit which itself generates an interrupt.*
1	FE	Framing error This bit is set by hardware when a de-synchronization, excessive noise or a break character is detected. It is cleared by a software sequence (an access to the SCISR register followed by a read to the SCIDR register). 0: No framing error is detected 1: Framing error or break character is detected *Note: The FE bit does not generate an interrupt as it appears at the same time as the RDRF bit which itself generates an interrupt. If the word currently being transferred causes both frame error and overrun error, it is transferred and only the OR bit is set.*
0	PE	Parity error This bit is set by hardware when a parity error occurs (if the PCE bit is set) in receiver mode. It is cleared by a software sequence (a read to the status register followed by an access to the SCIDR data register). An interrupt is generated if PIE = 1 in the SCICR1 register. 0: No parity error 1: Parity error

Figure 61. LIN header



The bits involved are the two MSB positions (7th and 8th bits if M = 0; 8th and 9th bits if M = 0) of the identifier character. The check is performed as specified in *Figure 62* by the LIN specification.

Figure 62. LIN identifier



LIN error detection

LIN header error flag

The LIN header error flag indicates that an invalid LIN header has been detected.

When a LIN header error occurs:

- The LHE flag is set
- An interrupt is generated if the RIE bit is set and the I[1:0] bits are cleared in the CCR register.

If autosynchronization is enabled (LASE bit = 1), this can mean that the LIN synch field is corrupted, and that the SCI is in a blocked state (LSF bit is set). The only way to recover is to reset the LSF bit and then to clear the LHE bit.

- The LHE bit is reset by an access to the SCISR register followed by a read of the SCIDR register.

LHE/OVR error conditions

When auto resynchronization is disabled (LASE bit = 0), the LHE flag detects:

- That the received LIN synch field is not equal to 55h.
- That an overrun occurred (as in standard SCI mode)
- Furthermore, if LHDM is set it also detects that a LIN header reception timeout occurred (only if LHDM is set).

Table 70. SCISR register description(1)

Bit	Name	Function
7	TDRE	Transmit data register empty This bit is set by hardware when the content of the TDR register has been transferred into the shift register. An interrupt is generated if the TIE = 1 in the SCICR2 register. It is cleared by a software sequence (an access to the SCISR register followed by a write to the SCIDR register). 0: Data is not transferred to the shift register 1: Data is transferred to the shift register
6	TC	Transmission complete This bit is set by hardware when transmission of a character containing data is complete. An interrupt is generated if TCIE = 1 in the SCICR2 register. It is cleared by a software sequence (an access to the SCISR register followed by a write to the SCIDR register). 0: Transmission is not complete 1: Transmission is complete *Note: TC is not set after the transmission of a preamble or a break.*
5	RDRF	Received data ready flag This bit is set by hardware when the content of the RDR register has been transferred to the SCIDR register. An interrupt is generated if RIE = 1 in the SCICR2 register. It is cleared by a software sequence (an access to the SCISR register followed by a read to the SCIDR register). 0: Data is not received 1: Received data are ready to be read
4	IDLE	Idle line detected This bit is set by hardware when an idle line is detected. An interrupt is generated if the ILIE = 1 in the SCICR2 register. It is cleared by a software sequence (an access to the SCISR register followed by a read to the SCIDR register). 0: No idle line is detected 1: idle line is detected *Note: The IDLE bit is not set again until the RDRF bit has been set itself (that is, a new idle line occurs).*
3	LHE	LIN header error During LIN header this bit signals three error types: The LIN synch field is corrupted and the SCI is blocked in LIN synch state (LSF bit = 1). A timeout occurred during LIN header reception. An overrun error was detected on one of the header field (see OR bit description in *Status register (SCISR) on page 134*). An interrupt is generated if RIE = 1 in the SCICR2 register. If blocked in the LIN synch state, the LSF bit must first be reset (to exit LIN synch field state and then to be able to clear LHE flag). Then it is cleared by the following software sequence: An access to the SCISR register followed by a read to the SCIDR register. 0: No LIN header error 1: LIN header error detected *Note: Apart from the LIN header this bit signals an overrun error as in SCI mode, (see description in Status register (SCISR) on page 134).*

Table 103. Operating conditions (tested for T_A = -40 to +125 °C) @ V_{DD} = 3.0 to 3.6 V

Symbol	Parameter	Conditions	Flash			ROM			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	
ACC_{RC}	RC resolution	V_{DD} = 3.3 V	-1		+1	TBD		TBD	%
	Accuracy of internal RC oscillator with RCCR = RCCR0(2)(3)	T_A = -40 to +125 °C, V_{DD} = 3.3 V	-3		+5	TBD		TBD	
		T_A = -40 to +125 °C, V_{DD} = 3.0 V to 3.6 V(4)	-4		+6	TBD		TBD	

1. If the RC oscillator clock is selected, to improve clock stability and frequency accuracy, it is recommended to place a decoupling capacitor, typically 100nF, between the V_{DD} and V_{SS} pins as close as possible to the ST7 device.
2. See *Section 7.1: Internal RC oscillator adjustment on page 37*.
3. Minimum value is obtained for hot temperature and max. value is obtained for cold temperature.
4. Data based on characterization results, not tested in production.

Table 104. Operating conditions (tested for T_A = -40 to +125 °C) @ V_{DD} = 3.0 to 3.6 V(1)

	Parameter(1)	Conditions	Flash and ROM			
			Min.	Typ.	Max.	
$I_{DD(RC)}$	RC oscillator current consumption	T_A = 25 °C, V_{DD} = 3.3 V		500(2)		µA
$t_{su(RC)}$	RC oscillator setup time				10(2)	µs

1. Data based on characterization results, not tested in production.
2. Measurement made with RC calibrated at 1 MHz.

Figure 75. Typical accuracy with RCCR = RCCR1 vs. V_{DD} = 3 to 3.6 V and temperature

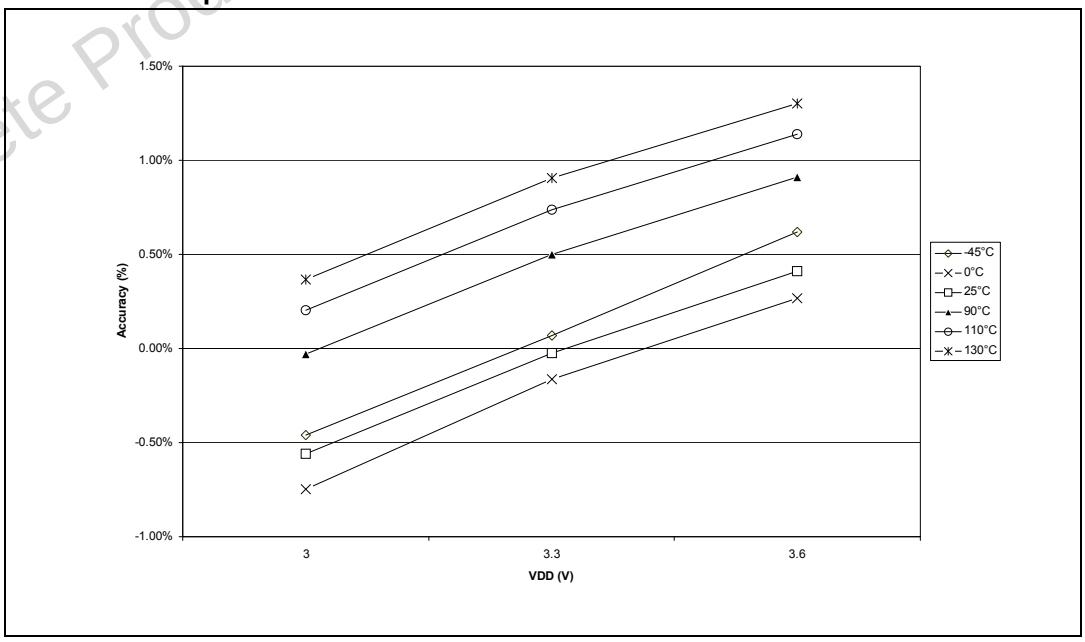
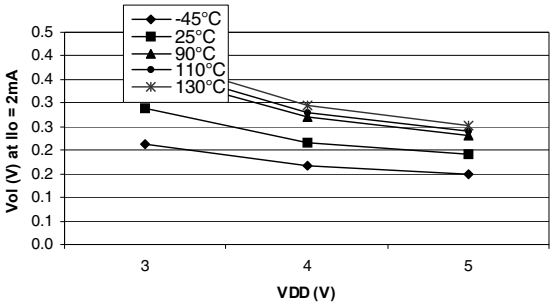


Figure 95. Typical V_{OL} vs. V_{DD} (standard I/Os)



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

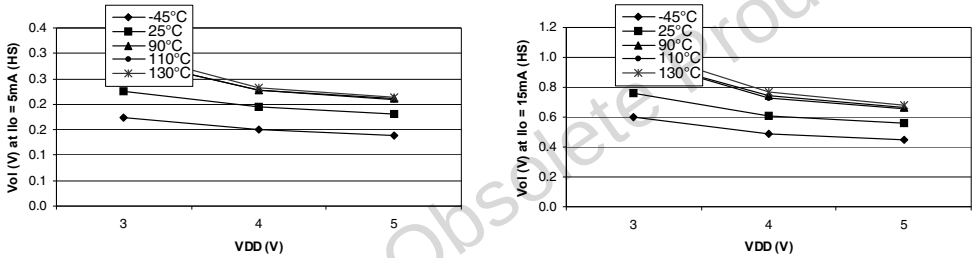


Figure 96. Typical V_{DD} - V_{OH} vs. V_{DD}

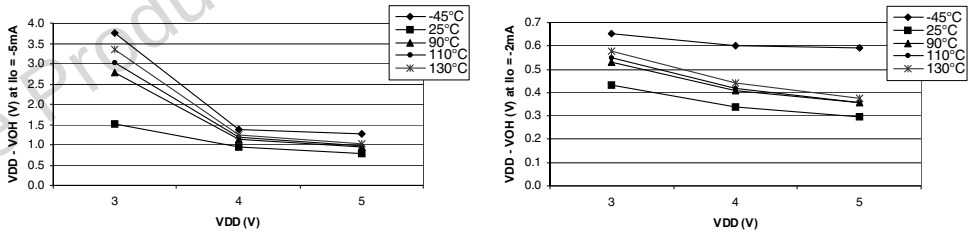
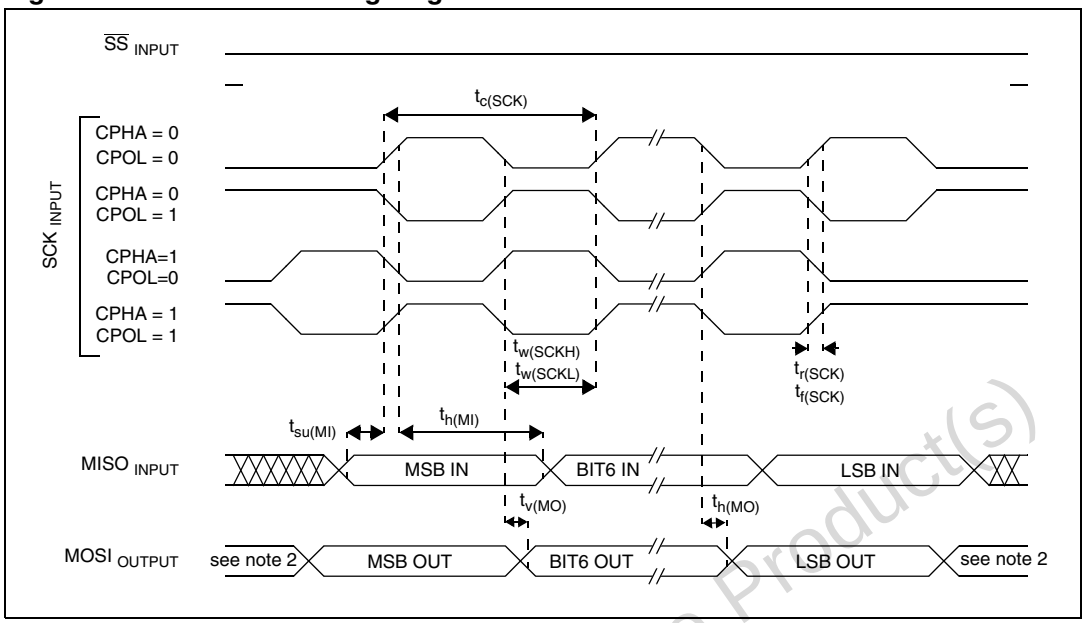


Figure 101. SPI master timing diagram



1. Measurement points are done at CMOS levels: 0.3 x V_{DD} and 0.7 x V_{DD}.
2. When no communication is on-going the data output line of the SPI (MOSI in master mode, MISO in slave mode) has its alternate function capability released. In this case, the pin status depends of the I/O port configuration.

14 Package characteristics

In order to meet environmental requirements, ST offers these devices in ECOPACK® packages. These packages have a lead-free second level interconnect. The category of second level interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label.

ECOPACK is an ST trademark. ECOPACK specifications are available at www.st.com.

14.1 Package mechanical data

Figure 104. 20-pin plastic small outline package, 300-mil width

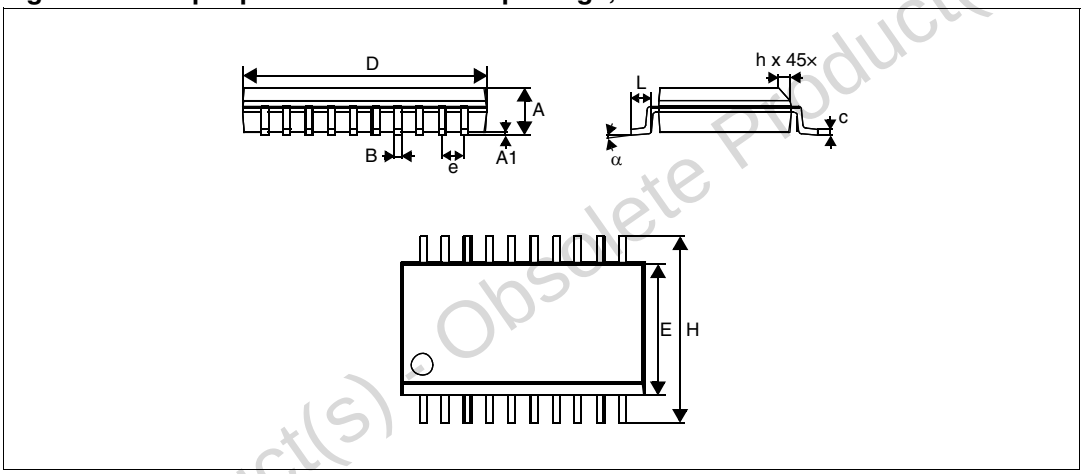
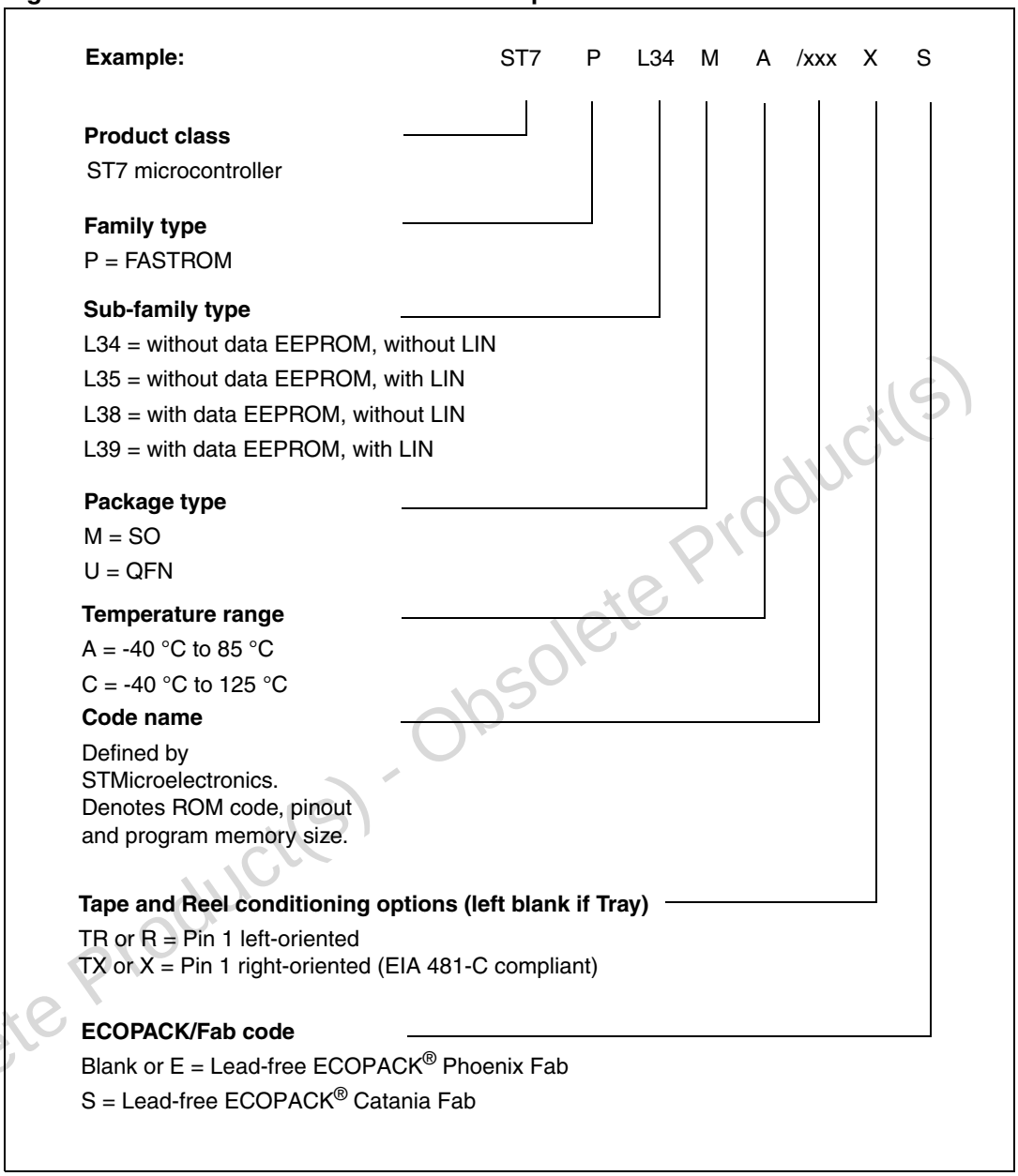


Table 127. 20-pin plastic small outline package, 300-mil width, mechanical data

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

Figure 108. ST7FL3x FASTROM commercial product structure



15.4 Development tools

15.4.1 Starter Kits

ST offers complete, affordable **starter kits**. Starter kits are complete, affordable hardware/software tool packages that include features and samples to help you quickly start developing your application.

15.4.2 Development and debugging tools

Application development for ST7 is supported by fully optimizing C compilers and the ST7 assembler-linker toolchain, which are all seamlessly integrated in the ST7 integrated development environments in order to facilitate the debugging and fine-tuning of your application. The cosmic C compiler is available in a free version that outputs up to 16 Kbytes of code.

The range of hardware tools includes full featured ST7-EMU3 series emulators, cost effective ST7-DVP3 series emulators and the low-cost RLink in-circuit debugger/programmer. These tools are supported by the ST7 Toolset from STMicroelectronics, which includes the STVD7 integrated development environment (IDE) with high-level language debugger, editor, project manager and integrated programming interface.

15.4.3 Programming tools

During the development cycle, the ST7-DVP3 and ST7-EMU3 series emulators and the RLink provide in-circuit programming capability for programming the Flash microcontroller on your application board.

ST also provides a low-cost dedicated in-circuit programmer, the ST7-STICK, as well as ST7 socket boards which provide all the sockets required for programming any of the devices in a specific ST7 subfamily on a platform that can be used with any tool with in-circuit programming capability for ST7.

For production programming of ST7 devices, ST's third-party tool partners also provide a complete range of gang and automated programming solutions, which are ready to integrate into your production environment.