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	SPI
Peripherals	PWM, WDT
Number of I/O	13
Program Memory Size	1.5KB (1.5K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 5x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SOIC (0.154", 3.90mm Width)
Supplier Device Package	16-SO
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st7fl05y0mae

5 DATA EEPROM

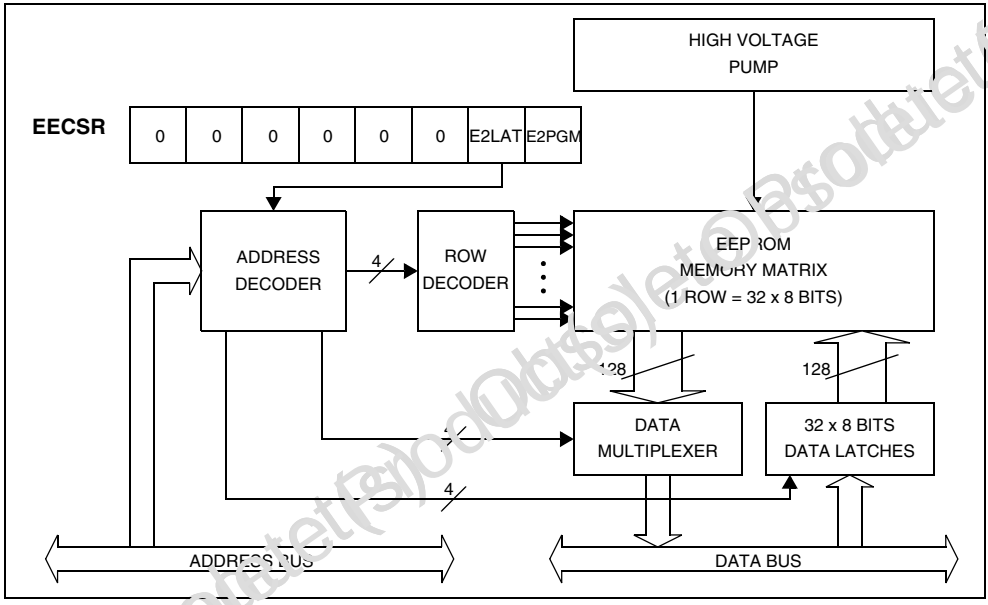
5.1 INTRODUCTION

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.

5.2 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
- Read-out protection

Figure 5. EEPROM Block Diagram



8 INTERRUPTS

The ST7 core may be interrupted by one of two different methods: Maskable hardware interrupts as listed in Table 7, “Interrupt Mapping,” on page 29 and a non-maskable software interrupt (TRAP). The Interrupt processing flowchart is shown in Figure 15.

The maskable interrupts must be enabled by clearing the I bit in order to be serviced. However, disabled interrupts may be latched and processed when they are enabled (see external interrupts subsection).

Note: After reset, all interrupts are disabled.

When an interrupt has to be serviced:

– Normal processing is suspended at the end of the current instruction execution.

– The PC, X, A and CC registers are saved onto the stack.

– The I bit of the CC register is set to prevent additional interrupts.

– The PC is then loaded with the interrupt vector of the interrupt to service and the first instruction of the interrupt service routine is fetched (refer to the Interrupt Mapping table for vector addresses).

The interrupt service routine should finish with the IRET instruction which causes the contents of the saved registers to be recovered from the stack.

Note: As a consequence of the IRET instruction, the I bit is cleared and the main program resumes.

Priority Management

By default, a servicing interrupt cannot be interrupted because the I bit is set by hardware entering in interrupt routine.

In the case when several interrupts are simultaneously pending, an hardware priority defines which one will be serviced first (see the Interrupt Mapping table).

Interrupts and Low Power Mode

All interrupts allow the processor to leave the WAIT low power mode. Only external and specifically mentioned interrupts allow the processor to leave the HALT low power mode (refer to the “Exit from HALT” column in the Interrupt Mapping table).

8.1 NON-MASKABLE SOFTWARE INTERRUPT

This interrupt is entered when the TRAP instruction is executed regardless of the state of the I bit. It is serviced according to the flowchart in Figure 15.

8.2 EXTERNAL INTERRUPTS

External interrupt vectors can be loaded into the PC register if the corresponding external interrupt occurred and if the I bit is cleared. These interrupts allow the processor to leave the HALT low power mode.

The external interrupt polarity is selected through the miscellaneous register or interrupt register (if available).

An external interrupt triggered on edge will be latched and the interrupt request automatically cleared upon entering the interrupt service routine.

Caution: The type of sensitivity defined in the Miscellaneous or Interrupt register (if available) applies to the ei source. In case of a NANDed source (as described in the I/O ports section), a low level on an I/O pin, configured as input with interrupt, masks the interrupt request even in case of rising-edge sensitivity.

8.3 PERIPHERAL INTERRUPTS

Different peripheral interrupt flags in the status register are able to cause an interrupt when they are active if both:

– The I bit of the CC register is cleared.

– The corresponding enable bit is set in the control register.

If any of these two conditions is false, the interrupt is latched and thus remains pending.

Clearing an interrupt request is done by:

– Writing “0” to the corresponding bit in the status register or

– Access to the status register while the flag is set followed by a read or write of an associated register.

Note: The clearing sequence resets the internal latch. A pending interrupt (that is, waiting for being enabled) will therefore be lost if the clear sequence is executed.

POWER SAVING MODES (Cont'd)

9.4 ACTIVE HALT AND HALT MODES

ACTIVE HALT and HALT modes are the two lowest power consumption modes of the MCU. They are both entered by executing the 'HALT' instruction. The decision to enter either in ACTIVE HALT or HALT mode is given by the LTCSR/ATCSR register status as shown in the following table:

LTCSR TBIE bit	ATCSR OVFIE bit	ATCSR CK1 bit	ATCSR CK0 bit	Meaning
0	x	x	0	ACTIVE HALT mode disabled
0	0	x	x	
0	1	1	1	
1	x	x	x	ACTIVE HALT mode enabled
x	1	0	1	

9.4.1 ACTIVE HALT Mode

ACTIVE HALT mode is the lowest power consumption mode of the MCU with a real time clock available. It is entered by executing the 'HALT' instruction when active halt mode is enabled.

The MCU can exit ACTIVE HALT mode on reception of a Lite Timer / AT Timer interrupt or a RESET.

– When exiting ACTIVE HALT mode by means of a RESET, a 256 CPU cycle delay occurs. After the start up delay, the CPU resumes operation by fetching the reset vector which woke it up (see Figure 20).

– When exiting ACTIVE HALT mode by means of an interrupt, the CPU immediately resumes operation by servicing the interrupt vector which woke it up (see Figure 20).

When entering ACTIVE HALT mode, the I bit in the CC register is cleared to enable interrupts. Therefore, if an interrupt is pending, the MCU wakes up immediately.

In ACTIVE HALT mode, only the main oscillator and the selected timer counter (LT/AT) are running to keep a wake-up time base. All other peripherals are not clocked except those which get their clock supply from another clock generator (such as external or auxiliary oscillator).

Caution: As soon as ACTIVE HALT is enabled, executing a HALT instruction while the Watchdog is active does not generate a RESET if the WDGHALT bit is reset.
This means that the device cannot spend more than a defined delay in this power saving mode.

Figure 19. ACTIVE HALT Timing Overview

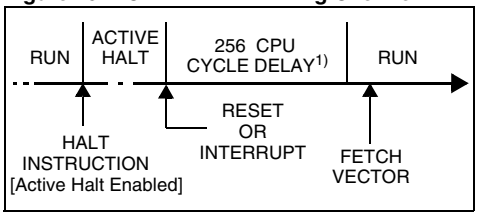
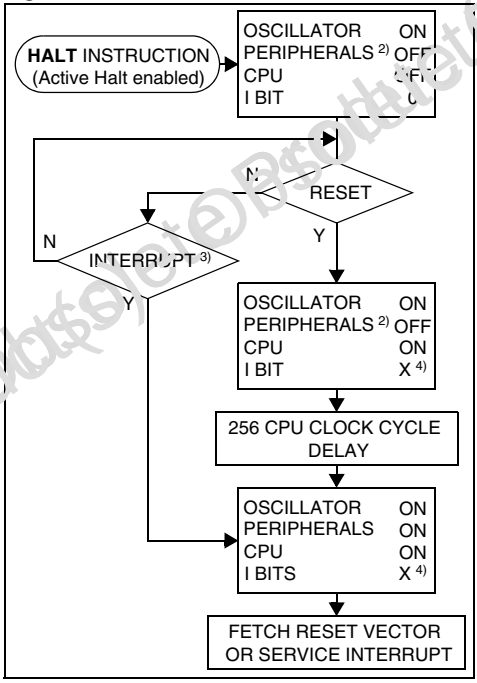


Figure 20. ACTIVE HALT Mode Flowchart



Notes:

1. This delay occurs only if the MCU exits ACTIVE HALT mode by means of a RESET.
2. Peripherals clocked with an external clock source can still be active.
3. Only the Lite Timer RTC and AT Timer interrupts can exit the MCU from ACTIVE HALT mode.
4. Before servicing an interrupt, the CC register is pushed on the stack. The I bit of the CC register is set during the interrupt routine and cleared when the CC register is popped.

11 ON-CHIP PERIPHERALS

11.1 LITE TIMER (LT)

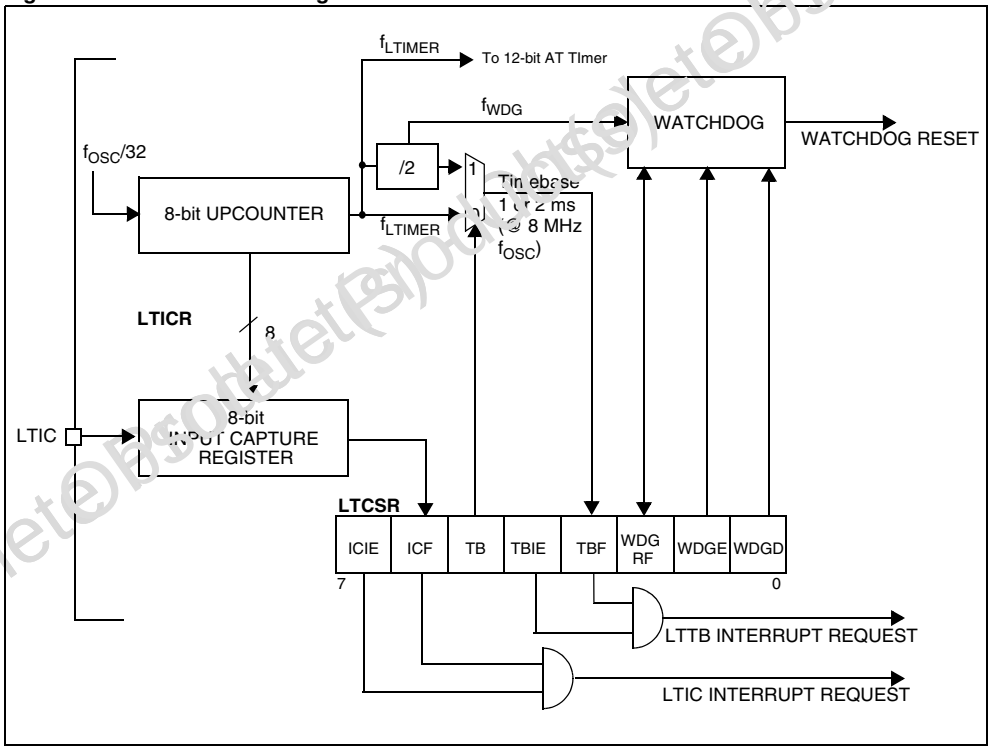
11.1.1 Introduction

The Lite Timer can be used for general-purpose timing functions. It is based on a free-running 8-bit upcounter with two software-selectable timebase periods, an 8-bit input capture register and watchdog function.

11.1.2 Main Features

- Realtime Clock
  - 8-bit upcounter
  - 1 ms or 2 ms timebase period (@ 8 MHz f_{OSC})
  - Maskable timebase interrupt
- Input Capture
  - 8-bit input capture register (LTICR)
  - Maskable interrupt with wakeup from Halt Mode capability
- Watchdog
  - Enabled by hardware or software (configurable by option byte)
  - Optional reset on HALT instruction (configurable by option byte)
  - Automatically resets the device unless disable bit is refreshed
  - Software reset (Forced Watchdog reset)
  - Watchdog reset status flag

Figure 25. Lite Timer Block Diagram



LITE TIMER (Cont'd)

11.1.3 Functional Description

The value of the 8-bit counter cannot be read or written by software. After an MCU reset, it starts incrementing from 0 at a frequency of $f_{OSC}/32$. A counter 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 LTCSR register.

When the timer overflows, the TBF bit is set by hardware and an interrupt request is generated if the TBIE is set. The TBF bit is cleared by software reading the LTCSR register.

11.1.3.1 Watchdog

The watchdog is enabled using the WDGE bit. The normal Watchdog timeout is 2ms (@ = 8 MHz f_{OSC}), after which it then generates a reset.

To prevent this watchdog reset occuring, software must set the WDGD bit. The WDGD bit is cleared by hardware after t_{WDG}. This means that software must write to the WDGD bit at regular intervals to prevent a watchdog reset occurring. Refer to Figure 26.

If the watchdog is not enabled immediately after reset, the first watchdog timeout will be shorter than 2ms, because this period is counted starting from reset. Moreover, if a 2ms period has already elapsed after the last MCU reset, the watchdog reset will take place as soon as the WDGE bit is set. For these reasons, it is recommended to enable the Watchdog immediately after reset or else to set the WDGD bit before the WDGE bit so a watchdog reset will not occur for at least 2ms.

Note: Software can use the timebase feature to set the WDGD bit at 1 or 2ms intervals.

A Watchdog reset can be forced at any time by setting the WDGRF bit. To generate a forced watchdog reset, first watchdog has to be activated by setting the WDGE bit and then the WDGRF bit has to be set.

The WDGRF bit also acts as a flag, indicating that the Watchdog was the source of the reset. It is automatically cleared after it has been read.

Caution: When the WDGRF bit is set, software must clear it, otherwise the next time the watchdog is enabled (by hardware or software), the microcontroller will be immediately reset.

Hardware Watchdog Option

If Hardware Watchdog is selected by option byte the watchdog is always active and the WDGE bit in the LTCSR is not used.

Refer to the Option Byte description in the device configuration and ordering information section.

Using Halt Mode with the Watchdog (option)

If the Watchdog reset on HALT option is not selected by option byte, the Halt mode can be used when the watchdog is enabled.

In this case, the HALT instruction stops the oscillator. When the oscillator is stopped, the Lite Timer stops counting and is no longer able to generate a Watchdog reset until the microcontroller receives an external interrupt or a reset.

When external interrupt is received, the WDG restarts counting after 256 CPU clocks. If a reset is generated, the Watchdog is disabled (reset state).

If Halt mode with Watchdog is enabled by option byte (No watchdog reset on HALT instruction), it is recommended before executing the HALT instruction to refresh the WDG counter, to avoid an unexpected WDG reset immediately after waking up the microcontroller.

11.2 12-BIT AUTORELOAD TIMER (AT)

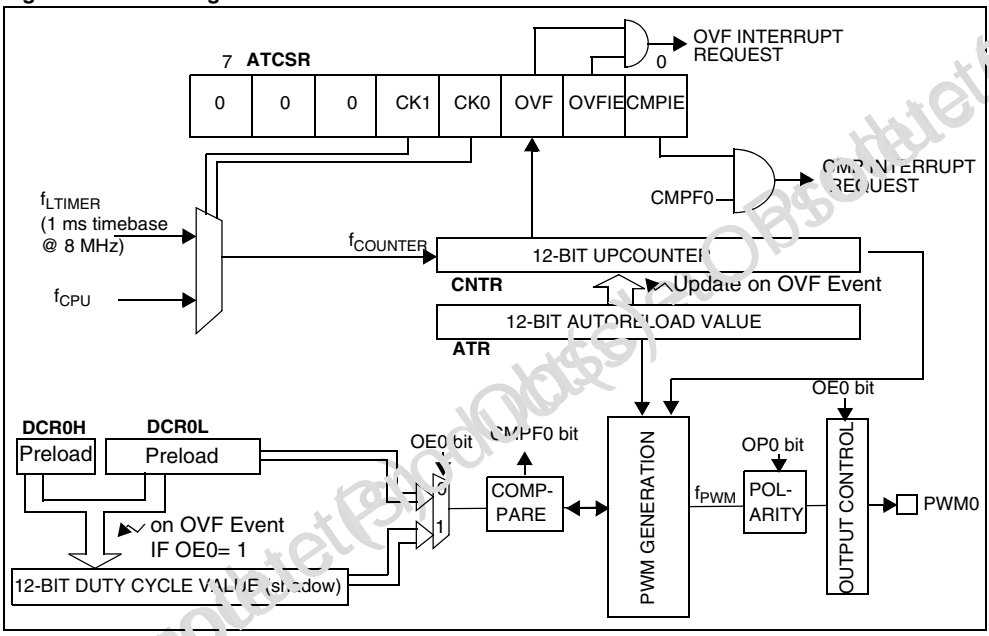
11.2.1 Introduction

The 12-bit Autoreload Timer can be used for general-purpose timing functions. It is based on a free-running 12-bit upcounter with a PWM output channel.

11.2.2 Main Features

- 12-bit upcounter with 12-bit autoreload register (ATR)
- Maskable overflow interrupt
- PWM signal generator
- Frequency range 2 kHz to 4 MHz (@ 8 MHz f_{CPU})
  - Programmable duty-cycle
  - Polarity control
  - Maskable Compare interrupt
- Output Compare Function

Figure 28. Block Diagram



12-BIT AUTORELOAD TIMER (Cont'd)

11.2.6 Register Description

TIMER CONTROL STATUS REGISTER (ATCSR)
Read / Write
Reset Value: 0000 0000 (00h)

7							0
0	0	0	CK1	CK0	OVF	OVFIE	CMPIE

Bit 7:5 = Reserved, must be kept cleared.

Bit 4:3 = **CK[1:0]** *Counter Clock Selection*
These bits are set and cleared by software and cleared by hardware after a reset. They select the clock frequency of the counter.

Counter Clock Selection	CK1	CK0
OFF	0	0
f_{LTIMER} (1 ms timebase @ 8 MHz)	0	1
f_{CPU}	1	0
Reserved	1	1

Bit 2 = **OVF** *Overflow Flag*
This bit is set by hardware and cleared by software by reading the ATCSR register. It indicates the transition of the counter from FFFh to ATR value.
0: No counter overflow occurred
1: Counter overflow occurred

Caution: When set, the OVF bit stays high for 1 $f_{COUNTER}$ cycle, (up to 1ms depending on the clock selection).

Bit 1 = **OVFIE** *Overflow Interrupt Enable*
This bit is read/write by software and cleared by hardware after a reset.
0: OVF interrupt disabled
1: OVF interrupt enabled

Bit 0 = **CMPIE** *Compare Interrupt Enable*
This bit is read/write by software and clear by hardware after a reset. It allows to mask the interrupt generation when CMPF bit is set.
0: CMPF interrupt disabled
1: CMPF interrupt enabled

COUNTER REGISTER HIGH (CNTRH)
Read only
Reset Value: 0000 0000 (00h)

15							8
0	0	0	0	CN11	CN10	CN9	CN8

COUNTER REGISTER LOW (CNTRL)
Read only
Reset Value: 0000 0000 (00h)

7							0
CN7	CN6	CN5	CN4	CN3	CN2	CN1	CN0

Bits 15:12 = Reserved, must be kept cleared.

Bits 11:0 = **CNTR[11:0]** *Counter Value*
This 12-bit register is read by software and cleared by hardware after a reset. The counter is incremented continuously as soon as a counter clock is selected. To obtain the 12-bit value, software should read the counter value in two consecutive read operations. The CNTRH register can be incremented between the two reads, and in order to be accurate when $f_{TIMER} = f_{CPU}$, the software should take this into account when CNTRL and CNTRH are read. If CNTRL is close to its highest value, CNTRH could be incremented before it is read.

When a counter overflow occurs, the counter restarts from the value specified in the ATR register.

SERIAL PERIPHERAL INTERFACE (Cont'd)

11.3.3.2 Slave Select Management

As an alternative to using the $\overline{SS}$ pin to control the Slave Select signal, the application can choose to manage the Slave Select signal by software. This is configured by the SSM bit in the SPICSR register (see Figure 34)

In software management, the external $\overline{SS}$ pin is free for other application uses and the internal $\overline{SS}$ signal level is driven by writing to the SSI bit in the SPICSR register.

In Master mode:

– $\overline{SS}$ internal must be held high continuously

In Slave mode:

Two cases depend on the data/clock timing relationship (see Figure 33):

If CPHA = 1 (data latched on 2nd clock edge):

– $\overline{SS}$ internal must be held low during the entire transmission. This implies that in single slave applications the $\overline{SS}$ pin either can be tied to V_{SS}, or made free for standard I/O by managing the $\overline{SS}$ function by software (SSM = 1 and SSI = 0 in the in the SPICSR register)

If CPHA = 0 (data latched on 1st clock edge):

– $\overline{SS}$ internal must be held low during byte transmission and pulled high between each byte to allow the slave to write to the shift register. If $\overline{SS}$ is not pulled high, a Write Collision error will occur when the slave writes to the shift register (see Section 11.3.5.3).

Figure 33. Generic $\overline{SS}$ Timing Diagram

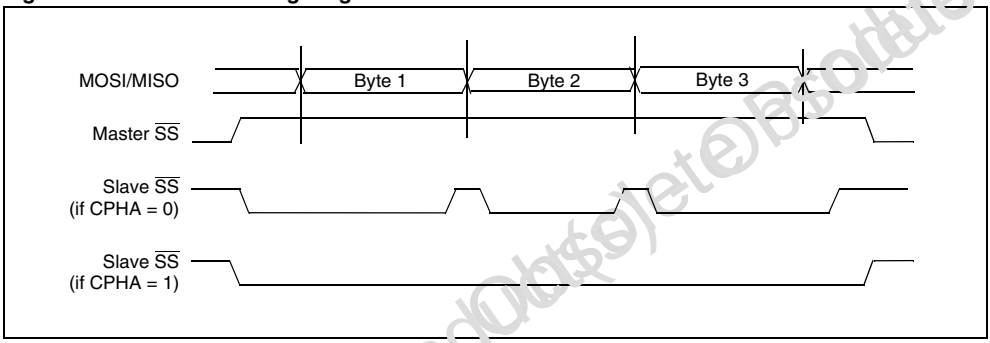
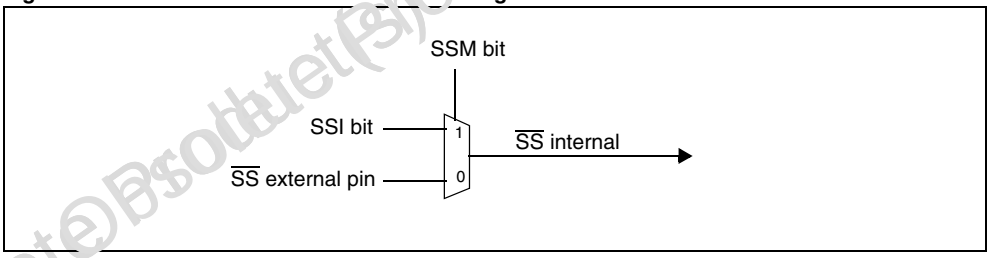


Figure 34. Hardware/Software Slave Select Management



SERIAL PERIPHERAL INTERFACE (Cont'd)

11.3.8 Register Description

CONTROL REGISTER (SPICR)

Read/Write

Reset Value: 0000 xxxx (0xh)

7							0
SPIE	SPE	SPR2	MSTR	CPOL	CPHA	SPR1	SPR0

Bit 7 = **SPIE** *Serial Peripheral Interrupt Enable*
This bit is set and cleared by software.
0: Interrupt is inhibited
1: An SPI interrupt is generated whenever SPIF = 1, MODF = 1 or OVR = 1 in the SPICSR register

Bit 6 = **SPE** *Serial Peripheral Output Enable*
This bit is set and cleared by software. It is also cleared by hardware when, in master mode, $\overline{SS}$ = 0 (see Section 11.3.5.1 Master Mode Fault (MODF)). The SPE bit is cleared by reset, so the SPI peripheral is not initially connected to the external pins.
0: I/O pins free for general purpose I/O
1: SPI I/O pin alternate functions enabled

Bit 5 = **SPR2** *Divider Enable*
This bit is set and cleared by software and is cleared by reset. It is used with the SPR[1:0] bits to set the baud rate. Refer to Table 15 SPI Master mode SCK Frequency.
0: Divider by 2 enabled
1: Divider by 2 disabled

Note: This bit has no effect in slave mode.

Bit 4 = **MSTR** *Master Mode*
This bit is set and cleared by software. It is also cleared by hardware when, in master mode, $\overline{SS}$ = 0 (see Section 11.3.5.1 Master Mode Fault (MODF)).
0: Slave mode
1: Master mode. The function of the SCK pin changes from an input to an output and the functions of the MISO and MOSI pins are reversed.

Bit 3 = **CPOL** *Clock Polarity*
This bit is set and cleared by software. This bit determines the idle state of the serial Clock. The CPOL bit affects both the master and slave modes.
0: SCK pin has a low level idle state
1: SCK pin has a high level idle state

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

Bit 2 = **CPHA** *Clock Phase*
This bit is set and cleared by software.
0: The first clock transition is the first data capture edge.
1: The second clock transition is the first capture edge.

Note: The slave must have the same CPOL and CPHA settings as the master.

Bits 1:0 = **SPR[1:0]** *Serial Clock Frequency*
These bits are set and cleared by software. Used with the SPR2 bit, they select the baud rate of the SPI serial clock SCK output by the SPI in master mode.

Note: These 2 bits have no effect in slave mode.

Table 15. SPI Master mode SCK Frequency

Serial Clock	SPR2	SPR1	SPR0
$f_{CPU}/4$	1	0	0
$f_{CPU}/8$	0	0	0
$f_{CPU}/16$	0	0	1
$f_{CPU}/32$	1	1	0
$f_{CPU}/64$	0	1	0
$f_{CPU}/128$	0	1	1

8-BIT A/D CONVERTER (ADC) (Cont'd)

11.4.3.2 Digital A/D Conversion Result

The conversion is monotonic, meaning that the result never decreases if the analog input does not and never increases if the analog input does not.

If the input voltage (V_{AIN}) is greater than or equal to V_{DDA} (high-level voltage reference) then the conversion result in the DR register is FFh (full scale) without overflow indication.

If input voltage (V_{AIN}) is lower than or equal to V_{SSA} (low-level voltage reference) then the conversion result in the DR register is 00h.

The A/D converter is linear and the digital result of the conversion is stored in the ADCDR register. The accuracy of the conversion is described in the parametric section.

R_{AIN} is the maximum recommended impedance for an analog input signal. If the impedance is too high, this will result in a loss of accuracy due to leakage and sampling not being completed in the allotted time.

11.4.3.3 A/D Conversion Phases

The A/D conversion is based on two conversion phases as shown in Figure 39:

- Sample capacitor loading [duration: t_{SAMPLE}]
  During this phase, the V_{AIN} input voltage to be measured is loaded into the C_{ADC} sample capacitor.
- A/D conversion [duration: t_{HOLD}]
  During this phase, the A/D conversion is computed (8 successive approximations cycles) and the C_{ADC} sample capacitor is disconnected from the analog input pin to get the optimum analog to digital conversion accuracy.
- The total conversion time:
  $t_{CONV} = t_{SAMPLE} + t_{HOLD}$

While the ADC is on, these two phases are continuously repeated.

At the end of each conversion, the sample capacitor is kept loaded with the previous measurement load. The advantage of this behavior is that it minimizes the current consumption on the analog pin in case of single input channel measurement.

11.4.3.4 Software Procedure

Refer to the control/status register (CSR) and data register (DR) in Section 11.4.6 for the bit definitions and to Figure 39 for the timings.

ADC Configuration

The analog input ports must be configured as input, no pull-up, no interrupt. Refer to the «I/O ports» chapter. Using these pins as analog inputs does not affect the ability of the port to be read as a logic input.

In the CSR register:

– Select the CH[2:0] bits to assign the analog channel to be converted.

ADC Conversion

In the CSR register:

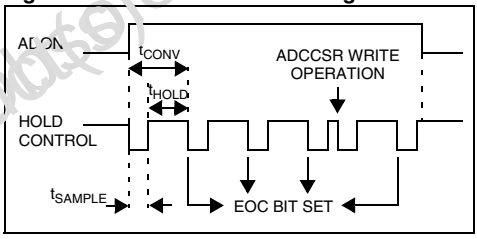
– Set the ADON bit to enable the A/D converter and to start the first conversion. From this time on, the ADC performs a continuous conversion of the selected channel.

When a conversion is complete

– The EOC bit is set by hardware.
– No interrupt is generated.
– The result is in the DR register and remains valid until the next conversion has ended.

A write to the ADCCSR register (with ADON set) aborts the current conversion, resets the EOC bit and starts a new conversion.

Figure 39. ADC Conversion Timings



11.4.4 Low Power Modes

Mode	Description
WAIT	No effect on A/D Converter
HALT	A/D Converter disabled. After wakeup from Halt mode, the A/D Converter requires a stabilization time before accurate conversions can be performed.

Note: The A/D converter may be disabled by resetting the ADON bit. This feature allows reduced power consumption when no conversion is needed and between single shot conversions.

11.4.5 Interrupts

None

ST7 ADDRESSING MODES (Cont'd)

12.1.1 Inherent

All Inherent instructions consist of a single byte. The opcode fully specifies all the required information for the CPU to process the operation.

Inherent Instruction	Function
NOP	No operation
TRAP	S/W Interrupt
WFI	Wait For Interrupt (Low Power Mode)
HALT	Halt Oscillator (Lowest Power Mode)
RET	Subroutine Return
IRET	Interrupt Subroutine Return
SIM	Set Interrupt Mask
RIM	Reset Interrupt Mask
SCF	Set Carry Flag
RCF	Reset Carry Flag
RSP	Reset Stack Pointer
LD	Load
CLR	Clear
PUSH/POP	Push/Pop to/from the stack
INC/DEC	Increment/Decrement
TNZ	Test Negative or Zero
CPL, NEG	1 or 2 Complement
MUL	Byte Multiplication
SLL, SRL, SRA, RLC, RRC	Shift and Rotate Operations
SWAP	Swap Nibbles

12.1.2 Immediate

Immediate instructions have 2 bytes, the first byte contains the opcode, the second byte contains the operand value.

Immediate Instruction	Function
LD	Load
CP	Compare
BCP	Bit Compare
AND, OR, XOR	Logical Operations
ADC, ADD, SUB, SBC	Arithmetic Operations

12.1.3 Direct

In Direct instructions, the operands are referenced by their memory address.

The direct addressing mode consists of two submodes:

Direct (Short)

The address is a byte, thus requires only 1 byte after the opcode, but only allows 00 - FF addressing space.

Direct (Long)

The address is a word, thus allowing 64 Kbyte addressing space, but requires 2 bytes after the opcode.

12.1.4 Indexed (No Offset, Short, Long)

In this mode, the operand is referenced by its memory address, which is defined by the unsigned addition of an index register (X or Y) with an offset.

The indirect addressing mode consists of three submodes:

Indexed (No Offset)

There is no offset (no extra byte after the opcode), and allows 00 - FF addressing space.

Indexed (Short)

The offset is a byte, thus requires only 1 byte after the opcode and allows 00 - 1FE addressing space.

Indexed (Long)

The offset is a word, thus allowing 64 Kbyte addressing space and requires 2 bytes after the opcode.

12.1.5 Indirect (Short, Long)

The required data byte to do the operation is found by its memory address, located in memory (pointer).

The pointer address follows the opcode. The indirect addressing mode consists of two submodes:

Indirect (Short)

The pointer address is a byte, the pointer size is a byte, thus allowing 00 - FF addressing space, and requires 1 byte after the opcode.

Indirect (Long)

The pointer address is a byte, the pointer size is a word, thus allowing 64 Kbyte addressing space, and requires 1 byte after the opcode.

INSTRUCTION GROUPS (Cont'd)

Mnemo	Description	Function/Example	Dst	Src	H	I	N	Z	C
ADC	Add with Carry	A = A + M + C	A	M	H		N	Z	C
ADD	Addition	A = A + M	A	M	H		N	Z	C
AND	Logical And	A = A . M	A	M			N	Z	
BCP	Bit compare A, Memory	tst (A . M)	A	M			N	Z	
BRES	Bit Reset	bres Byte, #3	M						
BSET	Bit Set	bset Byte, #3	M						
BTJF	Jump if bit is false (0)	btjf Byte, #3, Jmp1	M						C
BTJT	Jump if bit is true (1)	btjt Byte, #3, Jmp1	M						C
CALL	Call subroutine								
CALLR	Call subroutine relative								
CLR	Clear		reg, M				0	1	
CP	Arithmetic Compare	tst(Reg - M)	reg	M			N	Z	C
CPL	One Complement	A = FFH-A	reg, M				N	Z	1
DEC	Decrement	dec Y	reg, M				N	Z	
HALT	Halt					0			
IRET	Interrupt routine return	Pop CC, A, X, PC			H	I	N	Z	C
INC	Increment	inc X	reg, M				N	Z	
JP	Absolute Jump	jp [TBL.w]							
JRA	Jump relative always								
JRT	Jump relative								
JRF	Never jump	jrf *							
JRIH	Jump if ext. interrupt = 1								
JRIL	Jump if ext. interrupt = 0								
JRH	Jump if H = 1	H = 1 ?							
JRNH	Jump if H = 0	H = 0 ?							
JRM	Jump if I = 1	I = 1 ?							
JRNM	Jump if I = 0	I = 0 ?							
JRMI	Jump if N = 1 (minus)	N = 1 ?							
JRPL	Jump if N = 0 (plus)	N = 0 ?							
JREQ	Jump if Z = 1 (equal)	Z = 1 ?							
JRNE	Jump if Z = 0 (not equal)	Z = 0 ?							
JRC	Jump if C = 1	C = 1 ?							
JRNC	Jump if C = 0	C = 0 ?							
JRULT	Jump if C = 1	Unsigned <							
JRUGE	Jump if C = 0	Jmp if unsigned >=							
JRUGT	Jump if (C + Z = 0)	Unsigned >							

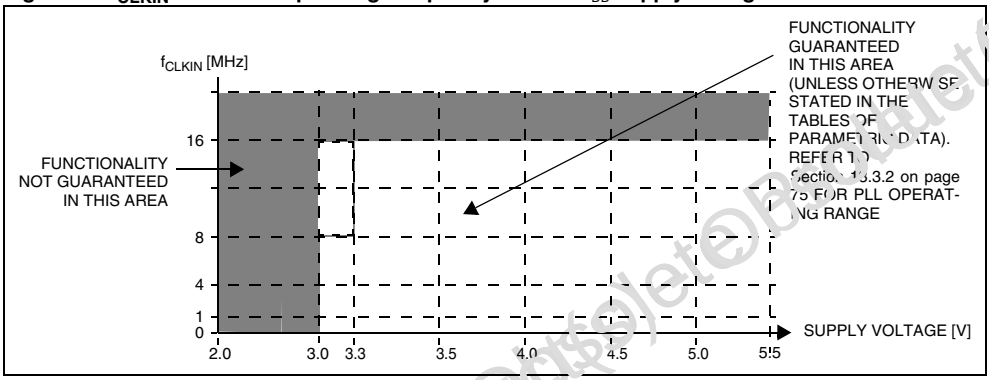
13.3 OPERATING CONDITIONS

13.3.1 General Operating Conditions

T_A = -40 to +105°C, unless otherwise specified.

Symbol	Parameter	Conditions	Min	Max	Unit
V_{DD}	Supply voltage	f_{OSC} = 16 MHz max T_A = -40°C to T_A max	3.0	5.5	V
f_{CLKIN}	External clock frequency on CLKIN pin	$V_{DD} \geq 3V$	0	16	MHz
T_A	Ambient temperature range	A Suffix version	-40	+85	°C
		B Suffix version		+105	

Figure 42. f_{CLKIN} Maximum Operating Frequency Versus V_{DD} Supply Voltage



Note: For further information on clock management and f_{CLKIN} description, refer to Figure 12 in Section 7 on page 23.

13.3.2 Internal RC Oscillator and PLL

The ST7 internal clock can be supplied by an internal RC oscillator and PLL (selectable by option byte).

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{DD(RC)}$	Internal RC oscillator operating voltage		3.0		5.5	V
$V_{DD(x8PLL)}$	x8 PLL operating voltage		3.6			
$t_{STARTUP}$	PLL start up time			60		PLL input clock (f_{PLL}) cycles

OPERATING CONDITIONS (Cont'd)

Figure 43. RC Osc Freq vs V_{DD} (Calibrated with RCCR0: 5V@ 25°C)

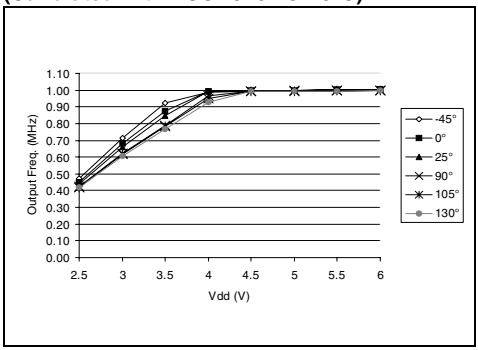


Figure 44. Typical RC oscillator Accuracy vs temperature @ V_{DD} = 5V (Calibrated with RCCR0: 5V @ 25°C

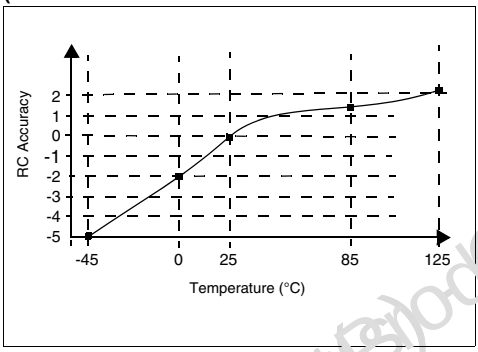


Figure 45. RC Osc Freq vs V_{DD} and RCCR Value

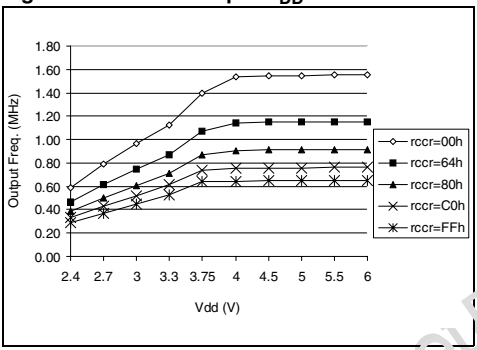
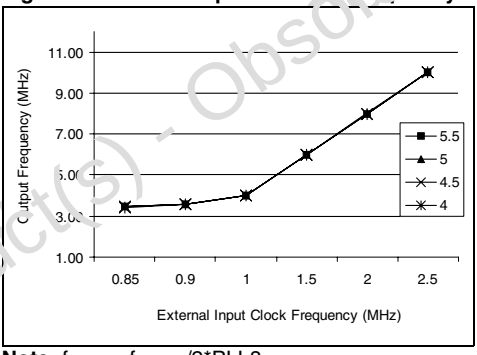


Figure 46. PLLx8 Output vs CLKIN frequency



Note: $f_{OSC} = f_{CLKIN}/2*PLL8$

EMC CHARACTERISTICS (Cont'd)

13.7.3 Absolute Maximum Ratings (Electrical Sensitivity)

Based on three different tests (ESD, LU and DLU) using specific measurement methods, the product is stressed in order to determine its performance in terms of electrical sensitivity. For more details, refer to the application note AN1181.

13.7.3.1 Electro-Static Discharge (ESD)

Electro-Static Discharges (a positive then a negative pulse separated by 1 second) are applied to the pins of each sample according to each pin combination. The sample size depends on the number of supply pins in the device (3 parts*(n+1) supply pin). Three models can be simulated: Human Body Model, Machine Model and Charge Device Model. This test conforms to the JESD22-A114A/A115A standard.

Absolute Maximum Ratings

Symbol	Ratings	Conditions	Maximum value[1]	Unit
$V_{ESD(HBM)}$	Electro-static discharge voltage (Human Body Model)	$T_A = 25°C$	2000	V
$V_{ESD(MM)}$	Electro-static discharge voltage (Machine Model)		200	
$V_{ESD(CDM)}$	Electro-static discharge voltage (Charge Device Model)	Pins 1, 8, 9 and 16 ($T_A = 25°C$)	750	
		All other pins ($T_A = 25°C$)	500	

Notes:

1. Data based on characterization results, not tested in production.

13.7.3.2 Static and Dynamic Latch-Up

- **LU**: Three complementary static tests are required on 10 parts to assess the latch-up performance. A supply overvoltage (applied to each power supply pin) and a current injection (applied to each input, output and configurable I/O pin) are performed on each sample. This test conforms to the EIA/JESD 78 IC latch-up standard. For more details, refer to the application note AN1181.
- **DLU**: Electro-Static Discharges (one positive then one negative test) are applied to each pin of three samples when the micro is running to assess the latch-up performance in dynamic mode. Power supplies are set to the typical values, the oscillator is connected as near as possible to the pins of the micro and the component is put in reset mode. This test conforms to the IEC1000-4-2 and SAEJ1752/3 standards. For more details, refer to the application note AN1181.

Electrical Sensitivities

Symbol	Parameter	Conditions	Class[1]
LU	Static latch-up class	$T_A = 25°C$, $T_A = 105°C$	A
DLU	Dynamic latch-up class	$V_{DD} = 5.5V$, $f_{OSC} = 4$ MHz, $T_A = 25°C$	

Notes:

1. Class description: A Class is an STMicroelectronics internal specification. All its limits are higher than the JEDEC specifications, that means when a device belongs to Class A it exceeds the JEDEC standard. B Class strictly covers all the JEDEC criteria (international standard).

13.8 I/O PORT PIN CHARACTERISTICS

13.8.1 General Characteristics

Subject to general operating conditions for V_{DD}, f_{OSC} and T_A (-40 to +105°C), unless otherwise specified.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
V_{IL}	Input low level voltage			V_{SS} - 0.3		0.3 x V_{DD}	V
V_{IH}	Input high level voltage			0.7xV_{DD}		V_{DD} + 0.3	
V_{hys}	Schmitt trigger voltage hysteresis[1]				400		mV
I_L	Input leakage current	$V_{SS} \le V_{IN} \le V_{DD}$				± 1	µA
I_S	Static current consumption induced by each floating input pin[2]	Floating input mode			400		
R_{PU}	Weak pull-up equivalent resistor[3]	$V_{IN} = V_{SS}$	V_{DD} = 5V	50	120	250	kΩ
C_{IO}	I/O pin capacitance				5		pF
$t_{f(IO)out}$	Output high to low level fall time[1]	C_L = 50pF Between 10% and 90%			25		ns
$t_{r(IO)out}$	Output low to high level rise time[1]						
$t_{w(IT)in}$	External interrupt pulse time[4]			1			t_{CPU}

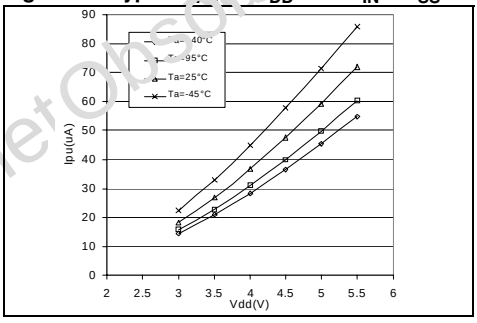
Notes:

1. Data based on characterization results, not tested in production.
2. Configuration not recommended, all unused pins must be kept at a fixed voltage: Using the output mode of the I/O for example or an external pull-up or pull-down resistor (see Figure 57). Static peak current value taken at a fixed V_{IN} value, based on design simulation and technology characteristics, not tested in production. This value depends on V_{DD} and temperature values.
3. The R_{PU} pull-up equivalent resistor is based on a resistive transistor (corresponding I_{PU} current characteristics described in Figure 53).
4. To generate an external interrupt, a minimum pulse width must be applied on an I/O port pin configured as an external interrupt source.

Figure 52. Two Typical Applications with Unused I/O Pin Configured as Input

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.
Note: I/O can be left unconnected if it is configured as output (0 or 1) by the software. This has the advantage of greater EMC robustness and lower cost.

Figure 53. Typical I_{PU} vs V_{DD} with $V_{IN} = V_{SS}$



PACKAGE CHARACTERISTICS (Cont'd)

14.3 SOLDERING INFORMATION

In accordance with the RoHS European directive, all STMicroelectronics packages have been converted to lead-free technology named ECOPACK™.

- ECOPACK™ packages are qualified according to the JEDEC STD-020C compliant soldering profile.
- Detailed information on the STMicroelectronics ECOPACK™ transition program is available on www.st.com/stonline/leadfree/, with specific technical application notes covering the main technical aspects related to lead-free conversion (AN2033, AN2034, AN2035, and AN2036).

Forward compatibility

ECOPACK™ SO packages are fully compatible with a lead (Pb) containing soldering process (see application note AN2034).

Table 22. Soldering Compatibility (Wave and Reflow Soldering Process)

Package	Plating material	Pb solder paste	Pb-free solder paste
SO	NiPdAu (Nickel-Palladium-Gold)	Yes	Yes

DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)

15.3 FLASH DEVICE ORDERING INFORMATiON

Figure 73. Flash Device Types

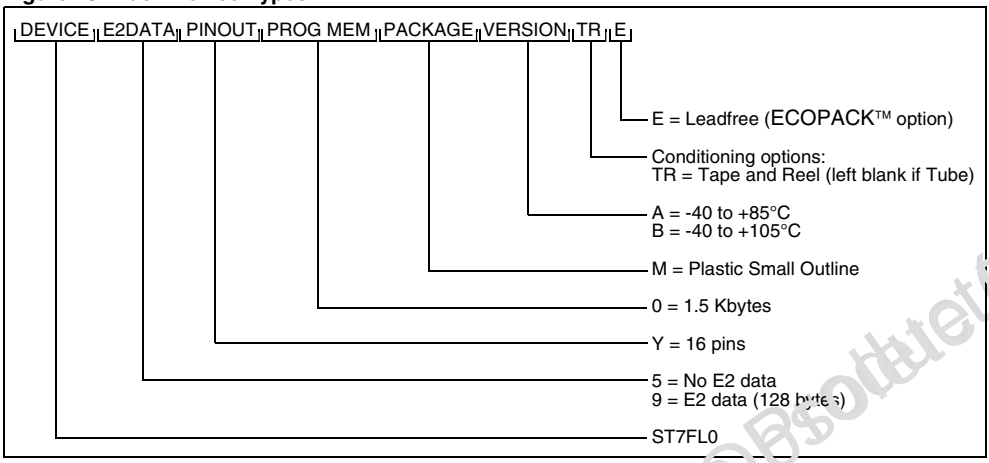


Table 24. Flash User Programmable Device Types

Part Number	Program Memory (Bytes)	Data EEPROM (Bytes)	RAM (Bytes)	Temp. Range	Package
ST7FL05Y0MA	1.5K Flash	-	128	-40 to +85°C	SO16
ST7FL09Y0MA		128			
ST7FL05Y0MB				-40 to +105°C	
ST7FL09Y0MB		128			

15.4 TRANSFER OF CUSTOMER CODE

Customer code is made up of the FASTROM contents and the list of the selected options (if any). The FASTROM contents are to be sent on diskette, or by electronic means, with the S19 hexadecimal file generated by the development tool. All unused bytes must be set to FFh.

The selected options are communicated to STMicroelectronics using the correctly completed OPTION LIST appended.

Refer to application note AN1635 for information on the counter listing returned by ST after code has been transferred.

The STMicroelectronics Sales Organization will be pleased to provide detailed information on contractual points.

DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)

Figure 74. FASTROM Commercial Product Code Structure

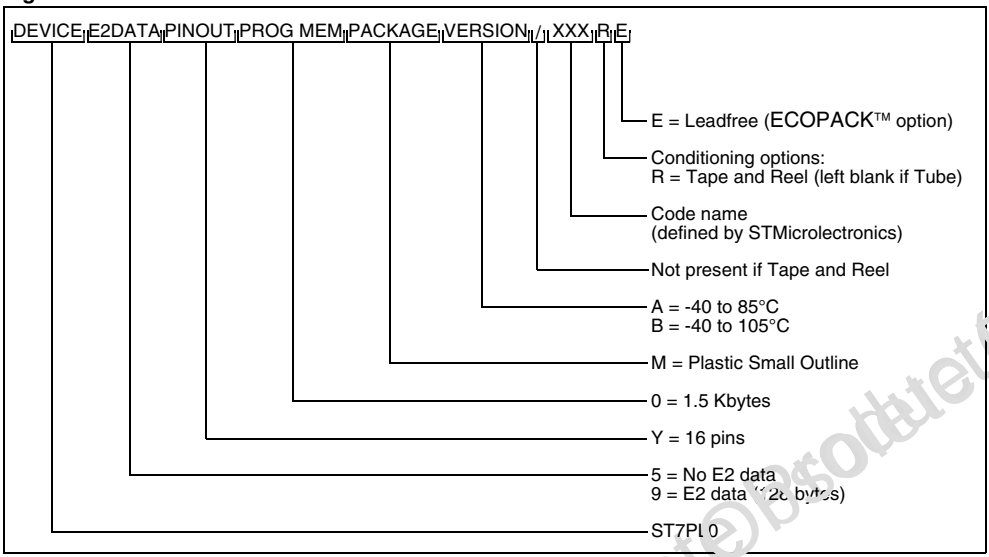


Table 25. FASTROM Factory Coded Device Types

Part Number	Program Memory (Bytes)	Data EEPROM (Bytes)	RAM (Bytes)	Temp. Range	Package
ST7PL05Y0MA	1.5K FASTROM	-	128	-40 to +85°C	SO16
ST7PL09Y0MA		128			
ST7PL05Y0MB		-		-40 to +105°C	
ST7PL09Y0MB		128			

18 REVISION HISTORY

Table 27. Revision History

Date	Revision	Description of changes
27-Mar-06	1	Initial Release
09-Oct-06	2	Changed “Memories” on page 1 Changed last paragraph of Section 5.5 on page 18 Changed description of bits 11:0 of CNTR register in Section 11.2.6 on page 50 Changed I_{VSS}, I_{VDD} and I_{IO} maximum values in Section 13.2.2 on page 74 Changed ACC_{RC} parameter for RC oscillator operating conditions in Section 13.3.2 on page 75 Removed note “tested in production” from Figure 44 on page 77 Changed values for $tv_{(MO)}$ and $th_{(MO)}$ in Section 13.10.1 on page 89 Replaced “CPHA = 0” with “CPHA = 1” in Figure 66 on page 90 Repositioned $tv_{(MO)}$ and $th_{(MO)}$ in Figure 67 on page 90 Changed values and note 3 in Table 20 on page 93 and Table 21 on page 93 Removed figure 72 “16-Pin Plastic Dual In-Line Package, 300-mil Width” from Section 14.1 on page 94 Changed Section 14.3 on page 95 Changed Table 22 on page 95 Changed Section 15.3 on page 98 by adding Figure 73. Flash Device Types and Table 24, “Flash User Programmable Device Types,” on page 98 Changed Section 15.4 on page 98 by adding Figure 74. FASTROM Commercial Product Code Structure and Table 25, “FASTROM Factory Coded Device Types,” on page 99 Updated “ST7L0x FASTROM MICROCONTROLLER OPTION LIST (Last update: October 2006)” on page 100 Changed Section 16 DEVELOPMENT TOOLS Removed section STARTER KITS from Section 16 DEVELOPMENT TOOLS Removed Starter Kits from Table 26, “Development Tool Order Codes for the ST7L0 Family,” on page 101 Added a statement to indicate that application notes can be found on the ST website and removed Table 26, ST7 Application Notes from Section 16.5 on page 101 Updated disclaimer (last page) to include a mention about the use of ST products in automotive applications
04-Dec-06	3	Replaced “ST7L0” with “ST7L05, ST7L09” in document name on page 1 Added “Features” heading above list of features on page 1 Added table number to “Device Summary” on page 1 Changed title of Section 1 on page 5 Table 2 on page 7**:** Removed caution about PB0 negative current injection restriction Section 7.4.1 on page 26: Added caution about avoiding unwanted behavior during Reset sequence Section 11.2.6 on page 50: Changed description of bits 11:0 of CNTR register (last sentence which had been inadvertantly removed in Rev. 2 has now been restored) Section 13.2.2 on page 74**:** - Replaced “Injected current on PB0 and PB1 pins” with “Injected current on PB1 pin” in ratings - Changed note 5 to remove PB0 negative current restriction Section 13.3.2 on page 75: Changed ACC_{RC} parameter for RC oscillator operating conditions Section 14.3 on page 95: Removed text concerning Pb-containing packages Table 22 on page 95: - Changed title of Plating Material column - Added Pb solder paste - Removed note 1 Removed link to st.com from bottom of “ST7L0x FASTROM MICROCONTROLLER OPTION LIST” on page 100