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	Active
Core Processor	STM8
Core Size	8-Bit
Speed	24MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	68
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	6K x 8
Voltage - Supply (Vcc/Vdd)	2.95V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	80-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/stm8s207m8t3b

4 Product overview

The following section intends to give an overview of the basic features of the STM8S20xxx functional modules and peripherals.

For more detailed information please refer to the corresponding family reference manual (RM0016).

4.1 Central processing unit STM8

The 8-bit STM8 core is designed for code efficiency and performance.

It contains 6 internal registers which are directly addressable in each execution context, 20 addressing modes including indexed indirect and relative addressing and 80 instructions.

Architecture and registers

- Harvard architecture
- 3-stage pipeline
- 32-bit wide program memory bus - single cycle fetching for most instructions
- X and Y 16-bit index registers - enabling indexed addressing modes with or without offset and read-modify-write type data manipulations
- 8-bit accumulator
- 24-bit program counter - 16-Mbyte linear memory space
- 16-bit stack pointer - access to a 64 K-level stack
- 8-bit condition code register - 7 condition flags for the result of the last instruction

Addressing

- 20 addressing modes
- Indexed indirect addressing mode for look-up tables located anywhere in the address space
- Stack pointer relative addressing mode for local variables and parameter passing

Instruction set

- 80 instructions with 2-byte average instruction size
- Standard data movement and logic/arithmetic functions
- 8-bit by 8-bit multiplication
- 16-bit by 8-bit and 16-bit by 16-bit division
- Bit manipulation
- Data transfer between stack and accumulator (push/pop) with direct stack access
- Data transfer using the X and Y registers or direct memory-to-memory transfers

Asynchronous communication (UART mode)

- Full duplex communication - NRZ standard format (mark/space)
- Programmable transmit and receive baud rates up to 1 Mbit/s (f_{CPU}/16) and capable of following any standard baud rate regardless of the input frequency
- Separate enable bits for transmitter and receiver
- Two receiver wakeup modes:
  - Address bit (MSB)
  - Idle line (interrupt)
- Transmission error detection with interrupt generation
- Parity control

LIN master capability

- Emission: Generates 13-bit sync break frame
- Reception: Detects 11-bit break frame

LIN slave mode

- Autonomous header handling - one single interrupt per valid message header
- Automatic baud rate synchronization - maximum tolerated initial clock deviation ±15%
- Sync delimiter checking
- 11-bit LIN sync break detection - break detection always active
- Parity check on the LIN identifier field
- LIN error management
- Hot plugging support

4.14.3 SPI

- Maximum speed: 10 Mbit/s (f_{MASTER}/2) both for master and slave
- Full duplex synchronous transfers
- Simplex synchronous transfers on two lines with a possible bidirectional data line
- Master or slave operation - selectable by hardware or software
- CRC calculation
- 1 byte Tx and Rx buffer
- Slave/master selection input pin

Table 6. Pin description (continued)

Pin number					Pin name	Type	Input			Output				Main function (after reset)	Default alternate function	Alternate function after remap [option bit]
LQFP80	LQFP64	LQFP48	LQFP44	LQFP32			floating	wpu	Ext. interrupt	High sink	Speed	OD	PP			
46	37	29	-	21	PC4/TIM1_CH4	I/O	**X**	X	X	HS	O3	X	X	**Port C4**	Timer 1 - channel 4	
47	38	30	27	22	PC5/SPI_SCK	I/O	**X**	X	X	HS	O3	X	X	**Port C5**	SPI clock	
48	39	31	28	-	V_{SSIO_2}	S								I/O ground		
49	40	32	29	-	V_{DDIO_2}	S								I/O power supply		
50	41	33	30	23	PC6/SPI_MOSI	I/O	**X**	X	X	HS	O3	X	X	**Port C6**	SPI master out/ slave in	
51	42	34	31	24	PC7/SPI_MISO	I/O	**X**	X	X	HS	O3	X	X	**Port C7**	SPI master in/ slave out	
52	43	35	32	-	PG0/CAN_TX(2)	I/O	**X**	X			O1	X	X	**Port G0**	beCAN transmit	
53	44	36	33	-	PG1/CAN_RX(2)	I/O	**X**	X			O1	X	X	**Port G1**	beCAN receive	
54	45	-	-	-	PG2	I/O	**X**	X			O1	X	X	**Port G2**		
55	46	-	-	-	PG3	I/O	**X**	X			O1	X	X	**Port G3**		
56	47	-	-	-	PG4	I/O	**X**	X			O1	X	X	**Port G4**		
57	48	-	-	-	PI0	I/O	**X**	X			O1	X	X	**Port I0**		
58	-	-	-	-	PI1	I/O	**X**	X			O1	X	X	**Port I1**		
59	-	-	-	-	PI2	I/O	**X**	X			O1	X	X	**Port I2**		
60	-	-	-	-	PI3	I/O	**X**	X			O1	X	X	**Port I3**		
61	-	-	-	-	PI4	I/O	**X**	X			O1	X	X	**Port I4**		
62	-	-	-	-	PI5	I/O	**X**	X			O1	X	X	**Port I5**		
63	49	-	-	-	PG5	I/O	**X**	X			O1	X	X	**Port G5**		
64	50	-	-	-	PG6	I/O	**X**	X			O1	X	X	**Port G6**		
65	51	-	-	-	PG7	I/O	**X**	X			O1	X	X	**Port G7**		
66	52	-	-	-	PE4	I/O	**X**	X	X		O1	X	X	**Port E4**		
67	53	37	-	-	PE3/TIM1_BKIN	I/O	**X**	X	X		O1	X	X	**Port E3**	Timer 1 - break input	
68	54	38	34	-	PE2/I^2C_SDA	I/O	**X**		X		O1	T(3)		**Port E2**	I^2C data	

Table 7 lists the boundary addresses for each memory size. The top of the stack is at the RAM end address in each case.

Table 7. Flash, Data EEPROM and RAM boundary addresses

Memory area	Size (bytes)	Start address	End address
Flash program memory	128 K	0x00 8000	0x02 7FFF
	64 K	0x00 8000	0x01 7FFF
	32 K	0x00 8000	0x00 FFFF
RAM	6 K	**0x00 0000**	0x00 17FF
	4 K	**0x00 0000**	0x00 1000
	2 K	**0x00 0000**	0x00 07FF
Data EEPROM	2048	0x00 4000	**0x00 47FF**
	1536	0x00 4000	**0x00 45FF**
	1024	0x00 4000	**0x00 43FF**

6.2 Register map

Table 8. I/O port hardware register map

Address	Block	Register label	Register name	Reset status
0x00 5000	Port A	PA_ODR	Port A data output latch register	0x00
0x00 5001		PA_IDR	Port A input pin value register	0x00
0x00 5002		PA_DDR	Port A data direction register	0x00
0x00 5003		PA_CR1	Port A control register 1	0x00
0x00 5004		PA_CR2	Port A control register 2	0x00
0x00 5005	Port B	PB_ODR	Port B data output latch register	0x00
0x00 5006		PB_IDR	Port B input pin value register	0x00
0x00 5007		PB_DDR	Port B data direction register	0x00
0x00 5008		PB_CR1	Port B control register 1	0x00
0x00 5009		PB_CR2	Port B control register 2	0x00
0x00 500A	Port C	PC_ODR	Port C data output latch register	0x00
0x00 500B		PB_IDR	Port C input pin value register	0x00
0x00 500C		PC_DDR	Port C data direction register	0x00
0x00 500D		PC_CR1	Port C control register 1	0x00
0x00 500E		PC_CR2	Port C control register 2	0x00

Table 9. General hardware register map (continued)

Address	Block	Register label	Register name	Reset status
0x00 50CC	CLK	CLK_HSITRIMR	HSI clock calibration trimming register	0x00
0x00 50CD		CLK_SWIMCCR	SWIM clock control register	0bXXXX XXX0
0x00 50CE to 0x00 50D0	Reserved area (3 bytes)			
0x00 50D1	WWDG	WWDG_CR	WWDG control register	0x7F
0x00 50D2		WWDG_WR	WWDR window register	0x7F
0x00 50D3 to 0x00 50DF	Reserved area (13 bytes)			
0x00 50E0	IWDG	IWDG_KR	IWDG key register	0xXX(2)
0x00 50E1		IWDG_PR	IWDG prescaler register	0x00
0x00 50E2		IWDG_RLR	IWDG reload register	0xFF
0x00 50E3 to 0x00 50EF	Reserved area (13 bytes)			
0x00 50F0	AWU	AWU_CSR1	AWU control/status register 1	0x00
0x00 50F1		AWU_APR	AWU asynchronous prescaler buffer register	0x3F
0x00 50F2		AWU_TBR	AWU timebase selection register	0x00
0x00 50F3	BEEP	BEEP_CSR	BEEP control/status register	0x1F
0x00 50F4 to 0x00 50FF	Reserved area (12 bytes)			
0x00 5200	SPI	SPI_CR1	SPI control register 1	0x00
0x00 5201		SPI_CR2	SPI control register 2	0x00
0x00 5202		SPI_ICR	SPI interrupt control register	0x00
0x00 5203		SPI_SR	SPI status register	0x02
0x00 5204		SPI_DR	SPI data register	0x00
0x00 5205		SPI_CRCPR	SPI CRC polynomial register	0x07
0x00 5206		SPI_RXCRCR	SPI Rx CRC register	0xFF
0x00 5207		SPI_TXCRCR	SPI Tx CRC register	0xFF
0x00 5208 to 0x00 520F	Reserved area (8 bytes)			
0x00 5210	I^2C	I2C_CR1	I^2C control register 1	0x00
0x00 5211		I2C_CR2	I^2C control register 2	0x00
0x00 5212		I2C_FREQR	I^2C frequency register	0x00
0x00 5213		I2C_OARL	I^2C own address register low	0x00
0x00 5214		I2C_OARH	I^2C own address register high	0x00
0x00 5215		Reserved		

Table 9. General hardware register map (continued)

Address	Block	Register label	Register name	Reset status
0x00 5300	TIM2	TIM2_CR1	TIM2 control register 1	0x00
0x00 5301		TIM2_IER	TIM2 interrupt enable register	0x00
0x00 5302		TIM2_SR1	TIM2 status register 1	0x00
0x00 5303		TIM2_SR2	TIM2 status register 2	0x00
0x00 5304		TIM2_EGR	TIM2 event generation register	0x00
0x00 5305		TIM2_CCMR1	TIM2 capture/compare mode register 1	0x00
0x00 5306		TIM2_CCMR2	TIM2 capture/compare mode register 2	0x00
0x00 5307		TIM2_CCMR3	TIM2 capture/compare mode register 3	0x00
0x00 5308		TIM2_CCER1	TIM2 capture/compare enable register 1	0x00
0x00 5309		TIM2_CCER2	TIM2 capture/compare enable register 2	0x00
0x00 530A		TIM2_CNTRH	TIM2 counter high	0x00
0x00 530B		TIM2_CNTRL	TIM2 counter low	0x00
00 530C0x		TIM2_PSCR	TIM2 prescaler register	0x00
0x00 530D		TIM2_ARRH	TIM2 auto-reload register high	0xFF
0x00 530E		TIM2_ARRL	TIM2 auto-reload register low	0xFF
0x00 530F		TIM2_CCR1H	TIM2 capture/compare register 1 high	0x00
0x00 5310		TIM2_CCR1L	TIM2 capture/compare register 1 low	0x00
0x00 5311		TIM2_CCR2H	TIM2 capture/compare reg. 2 high	0x00
0x00 5312		TIM2_CCR2L	TIM2 capture/compare register 2 low	0x00
0x00 5313		TIM2_CCR3H	TIM2 capture/compare register 3 high	0x00
0x00 5314		TIM2_CCR3L	TIM2 capture/compare register 3 low	0x00
0x00 5315 to 0x00 531F	Reserved area (11 bytes)			
0x00 5320	TIM3	TIM3_CR1	TIM3 control register 1	0x00
0x00 5321		TIM3_IER	TIM3 interrupt enable register	0x00
0x00 5322		TIM3_SR1	TIM3 status register 1	0x00
0x00 5323		TIM3_SR2	TIM3 status register 2	0x00
0x00 5324		TIM3_EGR	TIM3 event generation register	0x00
0x00 5325		TIM3_CCMR1	TIM3 capture/compare mode register 1	0x00
0x00 5326		TIM3_CCMR2	TIM3 capture/compare mode register 2	0x00
0x00 5327		TIM3_CCER1	TIM3 capture/compare enable register 1	0x00
0x00 5328		TIM3_CNTRH	TIM3 counter high	0x00
0x00 5329		TIM3_CNTRL	TIM3 counter low	0x00
0x00 532A		TIM3_PSCR	TIM3 prescaler register	0x00

Table 9. General hardware register map (continued)

Address	Block	Register label	Register name	Reset status
0x00 532B	TIM3	TIM3_ARRH	TIM3 auto-reload register high	0xFF
0x00 532C		TIM3_ARRL	TIM3 auto-reload register low	0xFF
0x00 532D		TIM3_CCR1H	TIM3 capture/compare register 1 high	0x00
0x00 532E		TIM3_CCR1L	TIM3 capture/compare register 1 low	0x00
0x00 532F		TIM3_CCR2H	TIM3 capture/compare register 2 high	0x00
0x00 5330		TIM3_CCR2L	TIM3 capture/compare register 2 low	0x00
0x00 5331 to 0x00 533F	Reserved area (15 bytes)			
0x00 5340	TIM4	TIM4_CR1	TIM4 control register 1	0x00
0x00 5341		TIM4_IER	TIM4 interrupt enable register	0x00
0x00 5342		TIM4_SR	TIM4 status register	0x00
0x00 5343		TIM4_EGR	TIM4 event generation register	0x00
0x00 5344		TIM4_CNTR	TIM4 counter	0x00
0x00 5345		TIM4_PSCR	TIM4 prescaler register	0x00
0x00 5346		TIM4_ARR	TIM4 auto-reload register	0xFF
0x00 5347 to 0x00 53FF	Reserved area (185 bytes)			
0x00 5400	ADC2	ADC _CSR	ADC control/status register	0x00
0x00 5401		ADC_CR1	ADC configuration register 1	0x00
0x00 5402		ADC_CR2	ADC configuration register 2	0x00
0x00 5403		ADC_CR3	ADC configuration register 3	0x00
0x00 5404		ADC_DRH	ADC data register high	0xXX
0x00 5405		ADC_DRL	ADC data register low	0xXX
0x00 5406		ADC_TDRH	ADC Schmitt trigger disable register high	0x00
0x00 5407		ADC_TDRL	ADC Schmitt trigger disable register low	0x00
0x00 5408 to 0x00 541F	Reserved area (24 bytes)			
0x00 5420	beCAN	CAN_MCR	CAN master control register	0x02
0x00 5421		CAN_MSR	CAN master status register	0x02
0x00 5422		CAN_TSR	CAN transmit status register	0x00
0x00 5423		CAN_TPR	CAN transmit priority register	0x0C
0x00 5424		CAN_RFR	CAN receive FIFO register	0x00
0x00 5425		CAN_IER	CAN interrupt enable register	0x00
0x00 5426		CAN_DGR	CAN diagnosis register	0x0C
0x00 5427		CAN_FPSR	CAN page selection register	0x00

8 Option bytes

Option bytes contain configurations for device hardware features as well as the memory protection of the device. They are stored in a dedicated block of the memory. Except for the ROP (read-out protection) byte, each option byte has to be stored twice, in a regular form (OPTx) and a complemented one (NOPTx) for redundancy.

Option bytes can be modified in ICP mode (via SWIM) by accessing the EEPROM address shown in *Table 12: Option bytes* below. Option bytes can also be modified 'on the fly' by the application in IAP mode, except the ROP option that can only be modified in ICP mode (via SWIM).

Refer to the STM8S Flash programming manual (PM0051) and STM8 SWIM communication protocol and debug module user manual (UM0470) for information on SWIM programming procedures.

Table 12. Option bytes

Addr.	Option name	Option byte no.	Option bits								Factory default setting
			7	6	5	4	3	2	1	0	
4800h	Read-out protection (ROP)	OPT0	ROP[7:0]								00h
4801h	User boot code (UBC)	OPT1	UBC[7:0]								00h
4802h		NOPT1	NUBC[7:0]								FFh
4803h	Alternate function remapping (AFR)	OPT2	AFR7	AFR6	AFR5	AFR4	AFR3	AFR2	AFR1	AFR0	00h
4804h		NOPT2	NAFR7	NAFR6	NAFR5	NAFR4	NAFR3	NAFR2	NAFR1	NAFR0	FFh
4805h	Watchdog option	OPT3	Reserved				LSI_EN	IWDG_HW	WWDG_HW	WWDG_HALT	00h
4806h		NOPT3	Reserved				NLSI_EN	NIWDG_HW	NWWDG_HW	NWWDG_HALT	FFh
4807h	Clock option	OPT4	Reserved				EXTCLK	CKAWUSEL	PRSC1	PRSC0	00h
4808h		NOPT4	Reserved				NEXTCLK	NCKAWUSEL	NPRSC1	NPRSC0	FFh
4809h	HSE clock startup	OPT5	HSECNT[7:0]								00h
480Ah		NOPT5	NHSECNT[7:0]								FFh
480Bh	Reserved	OPT6	Reserved								00h
480Ch		NOPT6	Reserved								FFh
480Dh	Flash wait states	OPT7	Reserved							Wait state	00h
480Eh		NOPT7	Reserved							Nwait state	FFh
487Eh	Bootloader	OPTBL	BL[7:0]								00h
487Fh		NOPTBL	NBL[7:0]								FFh

Table 13. Option byte description (continued)

Option byte no.	Description
OPTBL	**BL[7:0]** *Bootloader option byte* For STM8S products, this option is checked by the boot ROM code after reset. Depending on the content of addresses 0x487E, 0x487F, and 0x8000 (reset vector), the CPU jumps to the bootloader or to the reset vector. Refer to the UM0560 (STM8L/S bootloader manual) for more details. For STM8L products, the bootloader option bytes are on addresses 0xXXXX and 0xXXXX+1 (2 bytes). These option bytes control whether the bootloader is active or not. For more details, refer to the UM0560 (STM8L/S bootloader manual) for more details.

9 Unique ID

The devices feature a 96-bit unique device identifier which provides a reference number that is unique for any device and in any context. The 96 bits of the identifier can never be altered by the user.

The unique device identifier can be read in single bytes and may then be concatenated using a custom algorithm.

The unique device identifier is ideally suited:

- For use as serial numbers
- For use as security keys to increase the code security in the program memory while using and combining this unique ID with software cryptographic primitives and protocols before programming the internal memory.
- To activate secure boot processes

Table 14. Unique ID registers (96 bits)

Address	Content description	Unique ID bits							
		7	6	5	4	3	2	1	0
0x48CD	X co-ordinate on the wafer	U_ID[7:0]							
0x48CE		U_ID[15:8]							
0x48CF	Y co-ordinate on the wafer	U_ID[23:16]							
0x48D0		U_ID[31:24]							
0x48D1	Wafer number	U_ID[39:32]							
0x48D2	Lot number	U_ID[47:40]							
0x48D3		U_ID[55:48]							
0x48D4		U_ID[63:56]							
0x48D5		U_ID[71:64]							
0x48D6		U_ID[79:72]							
0x48D7		U_ID[87:80]							
0x48D8		U_ID[95:88]							

Table 21. Total current consumption with code execution in run mode at V_{DD} = 3.3 V

Symbol	Parameter	Conditions		Typ	Max(1)	Unit
$I_{DD(RUN)}$	Supply current in run mode, code executed from RAM	$f_{CPU} = f_{MASTER}$ = 24 MHz, $T_A \leq$ 105 °C	HSE crystal osc. (24 MHz)	4.0		mA
			HSE user ext. clock (24 MHz)	3.7	7.3	
		$f_{CPU} = f_{MASTER}$ = 16 MHz	HSE crystal osc. (16 MHz)	2.9		
			HSE user ext. clock (16 MHz)	2.7	5.8	
			HSI RC osc. (16 MHz)	2.5	3.4	
		$f_{CPU} = f_{MASTER}/128$ = 125 kHz	HSE user ext. clock (16 MHz)	1.2	4.1	
			HSI RC osc. (16 MHz)	1.0	1.3	
		$f_{CPU} = f_{MASTER}/128$ = 15.625 kHz	HSI RC osc. (16MHz/8)	0.55		
		$f_{CPU} = f_{MASTER}$ = 128 kHz	LSI RC osc. (128 kHz)	0.45		
	Supply current in run mode, code executed from Flash	$f_{CPU} = f_{MASTER}$ = 24 MHz, $T_A \leq$ 105 °C	HSE crystal osc. (24 MHz)	11.0		
			HSE user ext. clock (24 MHz)	10.8	18.0	
		$f_{CPU} = f_{MASTER}$ = 16 MHz	HSE crystal osc. (16 MHz)	8.4		
			HSE user ext. clock (16 MHz)	8.2	15.2	
			HSI RC osc. (16 MHz)	8.1	13.2	
		$f_{CPU} = f_{MASTER}$ = 2 MHz.	HSI RC osc. (16 MHz/8)(2)	1.5		
		$f_{CPU} = f_{MASTER}/128$ = 125 kHz	HSI RC osc. (16 MHz)	1.1		
		$f_{CPU} = f_{MASTER}/128$ = 15.625 kHz	HSI RC osc. (16 MHz/8)	0.6		
		$f_{CPU} = f_{MASTER}$ = 128 kHz	LSI RC osc. (128 kHz)	0.55		

1. Data based on characterization results, not tested in production.
2. Default clock configuration.

Total current consumption in halt mode

Table 26. Total current consumption in halt mode at V_{DD} = 5 V

Symbol	Parameter	Conditions	Typ	Max at 85 °C	Max at 125 °C	Unit
$I_{DD(H)}$	Supply current in halt mode	Flash in operating mode, HSI clock after wakeup	63.5			µA
		Flash in power-down mode, HSI clock after wakeup	6.5	35	100	

Table 27. Total current consumption in halt mode at V_{DD} = 3.3 V

Symbol	Parameter	Conditions	Typ	Unit
$I_{DD(H)}$	Supply current in halt mode	Flash in operating mode, HSI clock after wakeup	61.5	µA
		Flash in power-down mode, HSI clock after wakeup	4.5	

Low power mode wakeup times

Table 28. Wakeup times

Symbol	Parameter	Conditions			Typ	Max(1)	Unit
$t_{WU(WFI)}$	Wakeup time from wait mode to run mode(3)					See note(2)	µs
		$f_{CPU} = f_{MASTER}$ = 16 MHz.			0.56		
$t_{WU(AH)}$	Wakeup time active halt mode to run mode.(3)	MVR voltage regulator on(4)	Flash in operating mode(5)	HSI (after wakeup)	1(6)	2(6)	
			Flash in power-down mode(5)		3(6)		
		MVR voltage regulator off(4)	Flash in operating mode(5)		48(6)		
			Flash in power-down mode(5)		50(6)		
$t_{WU(H)}$	Wakeup time from halt mode to run mode(3)	Flash in operating mode(5)			52		
		Flash in power-down mode(5)			54		

1. Data guaranteed by design, not tested in production.
2. $t_{WU(WFI)} = 2 \times 1/f_{master} + 7 \times 1/f_{CPU}$
3. Measured from interrupt event to interrupt vector fetch.
4. Configured by the REGAH bit in the CLK_ICKR register.
5. Configured by the AHALT bit in the FLASH_CR1 register.
6. Plus 1 LSI clock depending on synchronization.

Total current consumption and timing in forced reset state

Table 29. Total current consumption and timing in forced reset state

Symbol	Parameter	Conditions	Typ	Max(1)	Unit
$I_{DD(R)}$	Supply current in reset state	V_{DD} = 5 V	1.6		mA
		V_{DD} = 3.3 V	0.8		
$t_{RESETBL}$	Reset release to bootloader vector fetch			150	µs

1. Data guaranteed by design, not tested in production.

Current consumption of on-chip peripherals

Subject to general operating conditions for V_{DD} and T_A.

HSI internal RC/f_{CPU} = f_{MASTER} = 16 MHz.

Table 30. Peripheral current consumption

Symbol	Parameter	Typ.	Unit
$I_{DD(TIM1)}$	TIM1 supply current (1)	220	µA
$I_{DD(TIM2)}$	TIM2 supply current (1)	120	
$I_{DD(TIM3)}$	TIM3 timer supply current (1)	100	
$I_{DD(TIM4)}$	TIM4 timer supply current (1)	25	
$I_{DD(UART1)}$	UART1 supply current (2)	90	
$I_{DD(UART3)}$	UART3 supply current (2)	110	
$I_{DD(SPI)}$	SPI supply current (2)	40	
$I_{DD(I^2C)}$	I^2C supply current (2)	50	
$I_{DD(CAN)}$	beCAN supply current (2)	210	
$I_{DD(ADC2)}$	ADC2 supply current when converting (3)	1000	

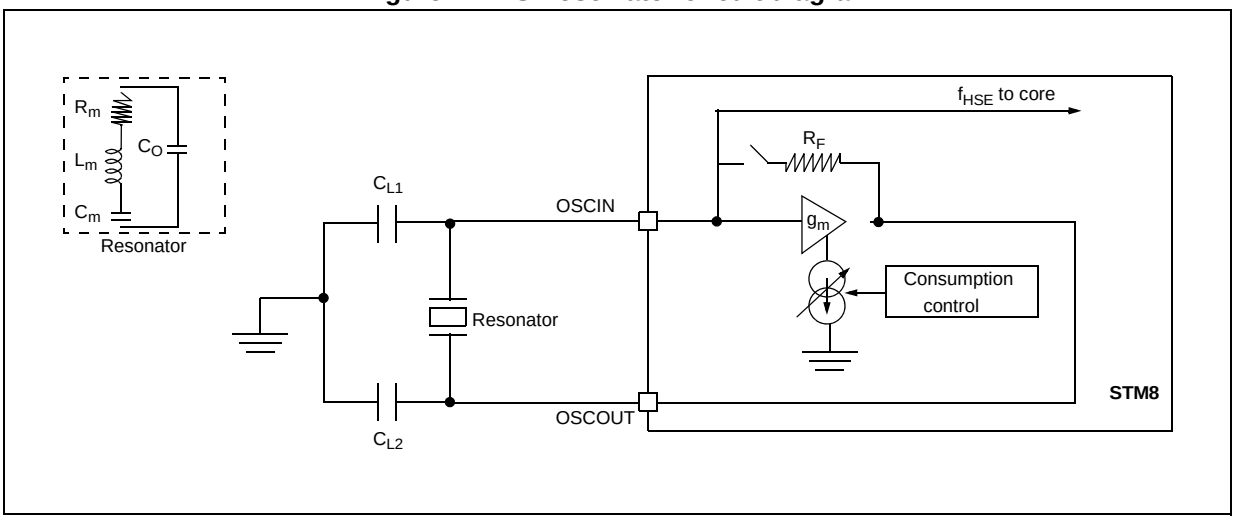
1. Data based on a differential I_{DD} measurement between reset configuration and timer counter running at 16 MHz. No IC/OC programmed (no I/O pads toggling). Not tested in production.
2. Data based on a differential I_{DD} measurement between the on-chip peripheral when kept under reset and not clocked and the on-chip peripheral when clocked and not kept under reset. No I/O pads toggling. Not tested in production.
3. Data based on a differential I_{DD} measurement between reset configuration and continuous A/D conversions. Not tested in production.

Table 32. HSE oscillator characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{HSE}	External high speed oscillator frequency		1		24	MHz
R_F	Feedback resistor			220		kΩ
$C^{(1)}$	Recommended load capacitance (2)				20	pF
$I_{DD(HSE)}$	HSE oscillator power consumption	C = 20 pF, f_{OSC} = 24 MHz			6 (startup) 2 (stabilized)(3)	mA
		C = 10 pF, f_{OSC} = 24 MHz			6 (startup) 1.5 (stabilized)(3)	
g_m	Oscillator transconductance		5			mA/V
$t_{SU(HSE)}^{(4)}$	Startup time	V_{DD} is stabilized		1		ms

1. C is approximately equivalent to 2 x crystal Cload.
2. The oscillator selection can be optimized in terms of supply current using a high quality resonator with small R_m value. Refer to crystal manufacturer for more details
3. Data based on characterization results, not tested in production.
4. $t_{SU(HSE)}$ is the start-up time measured from the moment it is enabled (by software) to a stabilized 24 MHz oscillation is reached. This value is measured for a standard crystal resonator and it can vary significantly with the crystal manufacturer.

Figure 17. HSE oscillator circuit diagram



HSE oscillator critical g_m formula

$$g_{mcrit} = (2 \times \Pi \times f_{HSE})^2 \times R_m(2Co + C)^2$$

R_m: Notional resistance (see crystal specification)
L_m: Notional inductance (see crystal specification)
C_m: Notional capacitance (see crystal specification)
Co: Shunt capacitance (see crystal specification)
C_{L1}=C_{L2}=C: Grounded external capacitance
$g_m >> g_{mcrit}$

Figure 29. Typ. V_{DD} - V_{OH} @ V_{DD} = 5 V (standard ports)

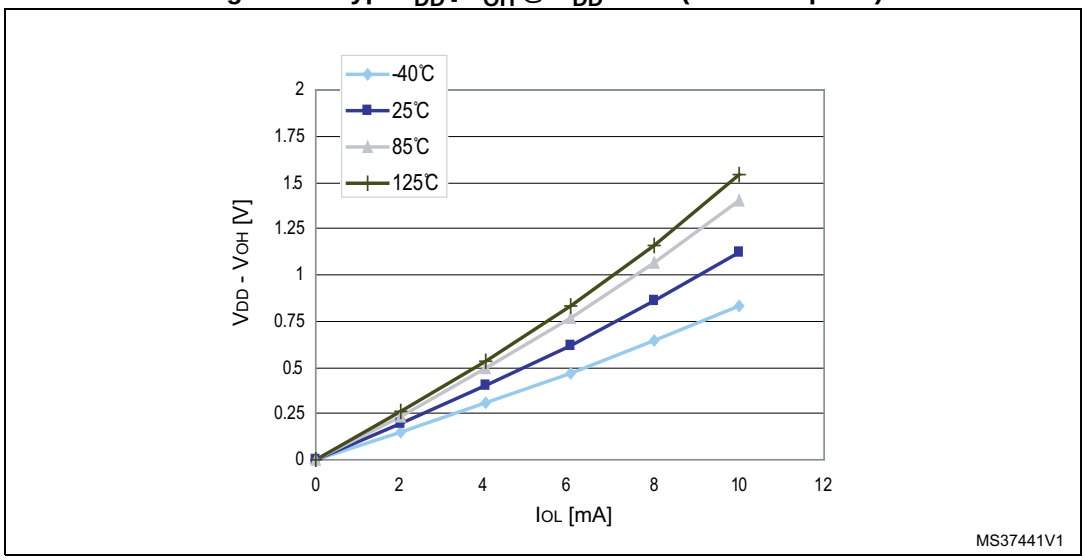


Figure 30. Typ. V_{DD} - V_{OH} @ V_{DD} = 3.3 V (standard ports)

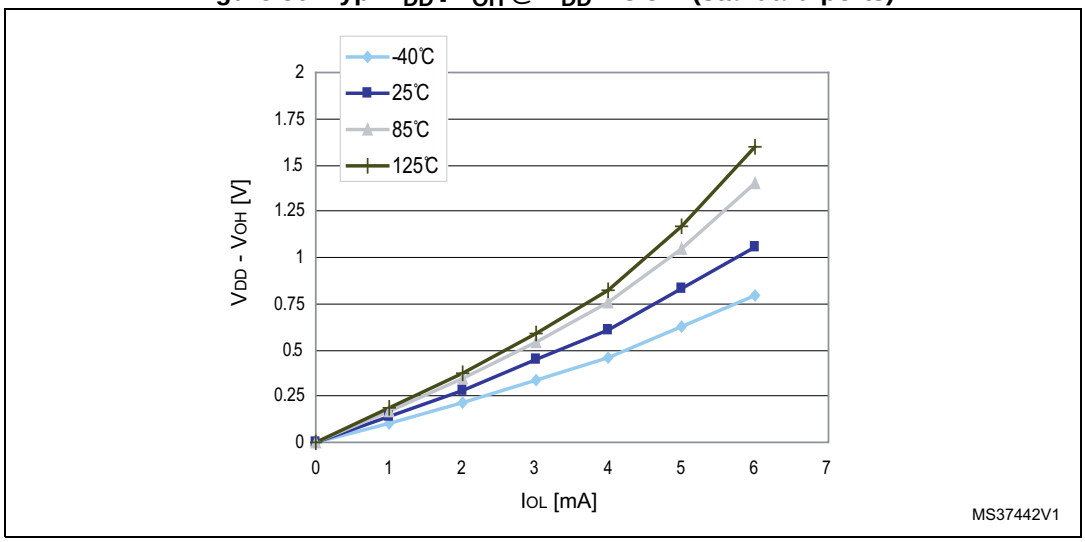
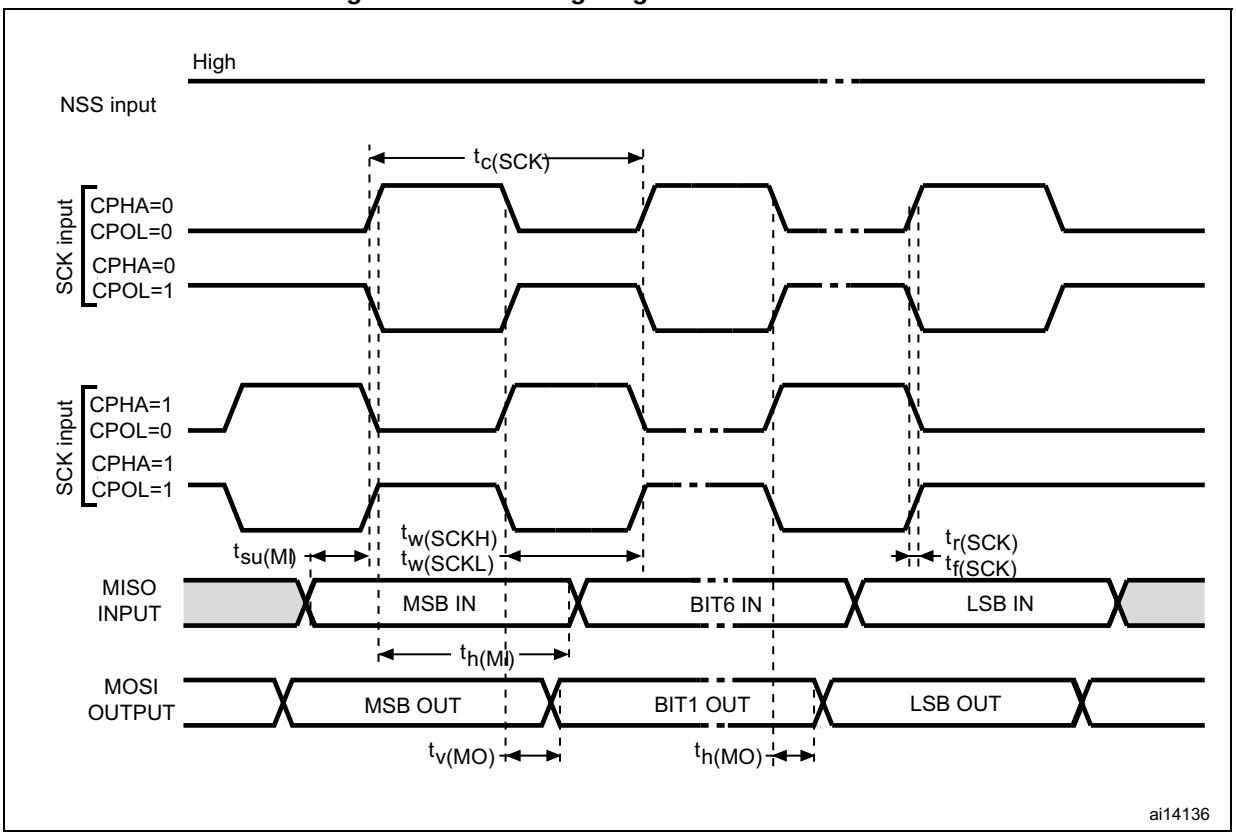


Figure 39. SPI timing diagram - master mode(1)



1. Measurement points are done at CMOS levels: 0.3 V_{DD} and 0.7 V_{DD}.

10.3.10 10-bit ADC characteristics

Subject to general operating conditions for V_{DDA}, f_{MASTER}, and T_A unless otherwise specified.

Table 44. ADC characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{ADC}	ADC clock frequency	V_{DDA} = 3 to 5.5 V	1		4	MHz
		V_{DDA} = 4.5 to 5.5 V	1		6	
V_{DDA}	Analog supply		3		5.5	V
V_{REF+}	Positive reference voltage		2.75(1)		V_{DDA}	V
V_{REF-}	Negative reference voltage		V_{SSA}		0.5(1)	V
V_{AIN}	Conversion voltage range(2)		V_{SSA}		V_{DDA}	V
		Devices with external V_{REF+}/V_{REF-} pins	V_{REF-}		V_{REF+}	V
C_{ADC}	Internal sample and hold capacitor			3		pF
t_S(2)	Sampling time	f_{ADC} = 4 MHz	0.75			µs
		f_{ADC} = 6 MHz	0.5			
t_{STAB}	Wakeup time from standby			7		µs
t_{CONV}	Total conversion time (including sampling time, 10-bit resolution)	f_{ADC} = 4 MHz	3.5			µs
		f_{ADC} = 6 MHz	2.33			µs
			14			$1/f_{ADC}$

1. Data guaranteed by design, not tested in production.
2. During the sample time the input capacitance C_{AIN} (3 pF max) can be charged/discharged by the external source. The internal resistance of the analog source must allow the capacitance to reach its final voltage level within t_S. After the end of the sample time t_S, changes of the analog input voltage have no effect on the conversion result. Values for the sample clock t_S depend on programming.

11.1.2 LQFP64 package information

Figure 46. LQFP64 - 64-pin 14 mm x 14 mm low-profile quad flat package outline

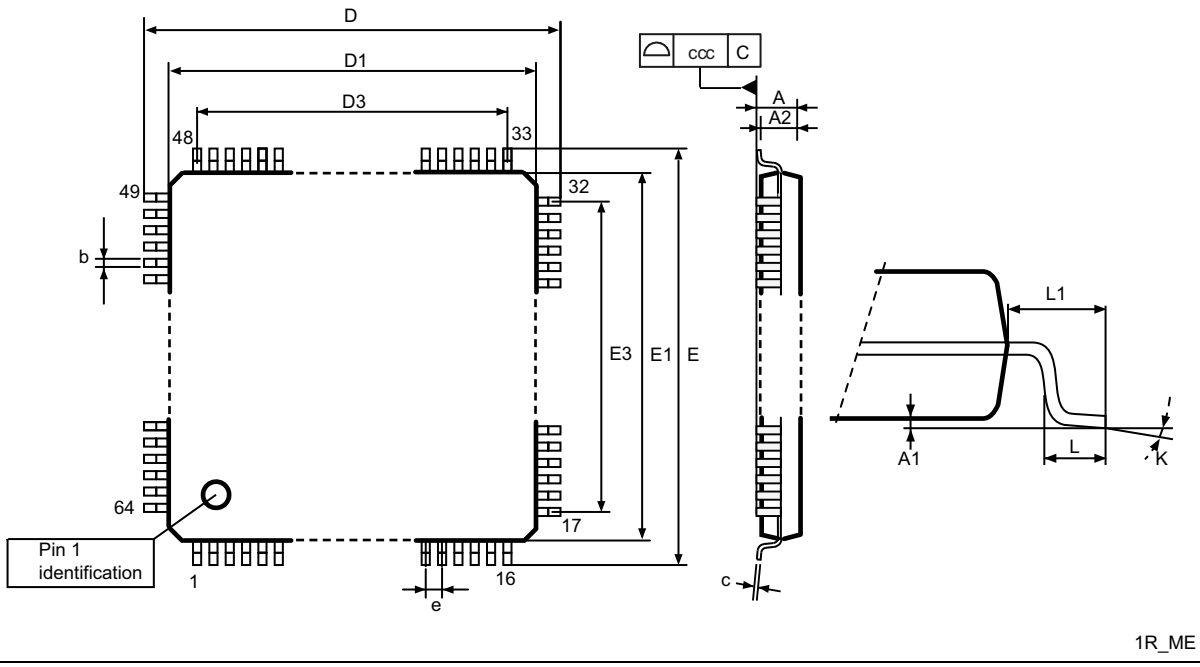


Table 52. LQFP64 - 64-pin, 14 x 14 mm low-profile quad flat package mechanical data

Symbol	mm			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A			1.600			0.0630
A1	0.050		0.150	0.0020		0.0059
A2	1.350	1.400	1.450	0.0531	0.0551	0.0571
b	0.300	0.370	0.450	0.0118	0.0146	0.0177
C	0.090		0.200	0.0035		0.0079
D	15.800	16.000	16.200	0.6220	0.6299	0.6378
D1	13.800	14.000	14.200	0.5433	0.5512	0.5591
D3		12.000			0.4724	
E	15.800	16.000	16.200	0.6220	0.6299	0.6378
E1	13.800	14.000	14.200	0.5433	0.5512	0.5591
E3		12.000			0.4724	
e		0.800			0.0315	
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1		1.000			0.0394	

11.2.2 Selecting the product temperature range

When ordering the microcontroller, the temperature range is specified in the order code (see *Figure 59: STM8S207xx/208xx performance line ordering information scheme(1) on page 112*).

The following example shows how to calculate the temperature range needed for a given application.

Assuming the following application conditions:

- Maximum ambient temperature T_{Amax}= 82 °C (measured according to JESD51-2)
- I_{DDmax} = 15 mA, V_{DD} = 5.5 V
- Maximum eight standard I/Os used at the same time in output at low level with I_{OL} = 10 mA, V_{OL}= 2 V
- Maximum four high sink I/Os used at the same time in output at low level with I_{OL} = 20 mA, V_{OL}= 1.5 V
- Maximum two true open drain I/Os used at the same time in output at low level with I_{OL} = 20 mA, V_{OL}= 2 V

  P_{INTmax} = 15 mA x 5.5 V = 82.5 mW

  P_{IOmax} = (10 mA x 2 V x 8) + (20 mA x 2 V x 2) + (20 mA x 1.5 V x 4) = 360 mW

  This gives: P_{INTmax} = 82.5 mW and P_{IOmax} 360 mW:

  P_{Dmax} = 82.5 mW + 360 mW

  Thus: P_{Dmax} = 443 mW

Using the values obtained in *Table 57: Thermal characteristics on page 108* T_{Jmax} is calculated as follows for LQFP64 10 x 10 mm = 46 °C/W:

T_{Jmax} = 82 °C + (46 °C/W x 443 mW) = 82 °C + 20 °C = 102 °C

This is within the range of the suffix 6 version parts (-40 < T_J < 105 °C).

In this case, parts must be ordered at least with the temperature range suffix 6.

12 STM8 development tools

Development tools for the STM8 microcontrollers include the full-featured STice emulation system supported by a complete software tool package including C compiler, assembler and integrated development environment with high-level language debugger. In addition, the STM8 is to be supported by a complete range of tools including starter kits, evaluation boards and a low-cost in-circuit debugger/programmer.

12.1 Emulation and in-circuit debugging tools

The STice emulation system offers a complete range of emulation and in-circuit debugging features on a platform that is designed for versatility and cost-effectiveness. In addition, STM8 application development is supported by a low-cost in-circuit debugger/programmer.

The STice is the fourth generation of full featured emulators from STMicroelectronics. It offers new advanced debugging capabilities including profiling and coverage to help detect and eliminate bottlenecks in application execution and dead code when fine tuning an application.

In addition, STice offers in-circuit debugging and programming of STM8 microcontrollers via the STM8 single wire interface module (SWIM), which allows non-intrusive debugging of an application while it runs on the target microcontroller.

For improved cost effectiveness, STice is based on a modular design that allows you to order exactly what you need to meet the development requirements and to adapt the emulation system to support existing and future ST microcontrollers.

STice key features

- Occurrence and time profiling and code coverage (new features)
- Advanced breakpoints with up to 4 levels of conditions
- Data breakpoints
- Program and data trace recording up to 128 KB records
- Read/write on the fly of memory during emulation
- In-circuit debugging/programming via SWIM protocol
- 8-bit probe analyzer
- 1 input and 2 output triggers
- Power supply follower managing application voltages between 1.62 to 5.5 V
- Modularity that allows you to specify the components you need to meet the development requirements and adapt to future requirements
- Supported by free software tools that include integrated development environment (IDE), programming software interface and assembler for STM8.