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	52
Program Memory Size	128KB (128K 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 ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/stm8s207rbt6ctr

List of tables

2 Description

The STM8S20xxx performance line 8-bit microcontrollers offer from 32 to 128 Kbytes Flash program memory. They are referred to as high-density devices in the STM8S microcontroller family reference manual.

All STM8S20xxx devices provide the following benefits: reduced system cost, performance robustness, short development cycles, and product longevity.

The system cost is reduced thanks to an integrated true data EEPROM for up to 300 k write/erase cycles and a high system integration level with internal clock oscillators, watchdog, and brown-out reset.

Device performance is ensured by 20 MIPS at 24 MHz CPU clock frequency and enhanced characteristics which include robust I/O, independent watchdogs (with a separate clock source), and a clock security system.

Short development cycles are guaranteed due to application scalability across a common family product architecture with compatible pinout, memory map and modular peripherals. Full documentation is offered with a wide choice of development tools.

Product longevity is ensured in the STM8S family thanks to their advanced core which is made in a state-of-the art technology for applications with 2.95 V to 5.5 V operating supply.

4.6 Power management

For efficient power management, the application can be put in one of four different low-power modes. You can configure each mode to obtain the best compromise between lowest power consumption, fastest start-up time and available wakeup sources.

- ***Wait mode***: In this mode, the CPU is stopped, but peripherals are kept running. The wakeup is performed by an internal or external interrupt or reset.
- ***Active halt mode with regulator on***: In this mode, the CPU and peripheral clocks are stopped. An internal wakeup is generated at programmable intervals by the auto wake up unit (AWU). The main voltage regulator is kept powered on, so current consumption is higher than in active halt mode with regulator off, but the wakeup time is faster. Wakeup is triggered by the internal AWU interrupt, external interrupt or reset.
- ***Active halt mode with regulator off:*** This mode is the same as active halt with regulator on, except that the main voltage regulator is powered off, so the wake up time is slower.
- ***Halt mode***: In this mode the microcontroller uses the least power. The CPU and peripheral clocks are stopped, the main voltage regulator is powered off. Wakeup is triggered by external event or reset.

4.7 Watchdog timers

The watchdog system is based on two independent timers providing maximum security to the applications.

Activation of the watchdog timers is controlled by option bytes or by software. Once activated, the watchdogs cannot be disabled by the user program without performing a reset.

Window watchdog timer

The window watchdog is used to detect the occurrence of a software fault, usually generated by external interferences or by unexpected logical conditions, which cause the application program to abandon its normal sequence.

The window function can be used to trim the watchdog behavior to match the application perfectly.

The application software must refresh the counter before time-out and during a limited time window.

A reset is generated in two situations:

1. Timeout: At 16 MHz CPU clock the time-out period can be adjusted between 75 µs up to 64 ms.
2. Refresh out of window: The downcounter is refreshed before its value is lower than the one stored in the window register.

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 9. General hardware register map (continued)

Address	Block	Register label	Register name	Reset status
0x00 5216	I²C	I2C_DR	I²C data register	0x00
0x00 5217		I2C_SR1	I²C status register 1	0x00
0x00 5218		I2C_SR2	I²C status register 2	0x00
0x00 5219		I2C_SR3	I²C status register 3	0x00
0x00 521A		I2C_ITR	I²C interrupt control register	0x00
0x00 521B		I2C_CCRL	I²C clock control register low	0x00
0x00 521C		I2C_CCRH	I²C clock control register high	0x00
0x00 521D		I2C_TRISER	I²C TRISE register	0x02
0x00 521E to 0x00 522F	Reserved area (18 bytes)			
0x00 5230	UART1	UART1_SR	UART1 status register	0xC0
0x00 5231		UART1_DR	UART1 data register	0xXX
0x00 5232		UART1_BRR1	UART1 baud rate register 1	0x00
0x00 5233		UART1_BRR2	UART1 baud rate register 2	0x00
0x00 5234		UART1_CR1	UART1 control register 1	0x00
0x00 5235		UART1_CR2	UART1 control register 2	0x00
0x00 5236		UART1_CR3	UART1 control register 3	0x00
0x00 5237		UART1_CR4	UART1 control register 4	0x00
0x00 5238		UART1_CR5	UART1 control register 5	0x00
0x00 5239		UART1_GTR	UART1 guard time register	0x00
0x00 523A		UART1_PSCR	UART1 prescaler register	0x00
0x00 523B to 0x00 523F	Reserved area (5 bytes)			
0x00 5240	UART3	UART3_SR	UART3 status register	C0h
0x00 5241		UART3_DR	UART3 data register	0xXX
0x00 5242		UART3_BRR1	UART3 baud rate register 1	0x00
0x00 5243		UART3_BRR2	UART3 baud rate register 2	0x00
0x00 5244		UART3_CR1	UART3 control register 1	0x00
0x00 5245		UART3_CR2	UART3 control register 2	0x00
0x00 5246		UART3_CR3	UART3 control register 3	0x00
0x00 5247		UART3_CR4	UART3 control register 4	0x00
0x00 5248		Reserved		
0x00 5249		UART3_CR6	UART3 control register 6	0x00
0x00 524A to 0x00 524F	Reserved area (6 bytes)			

Table 10. CPU/SWIM/debug module/interrupt controller registers (continued)

Address	Block	Register Label	Register Name	Reset Status
0x00 7F98	DM	DM_CSR1	DM debug module control/status register 1	0x10
0x00 7F99		DM_CSR2	DM debug module control/status register 2	0x00
0x00 7F9A		DM_ENFCTR	DM enable function register	0xFF
0x00 7F9B to 0x00 7F9F	Reserved area (5 bytes)			

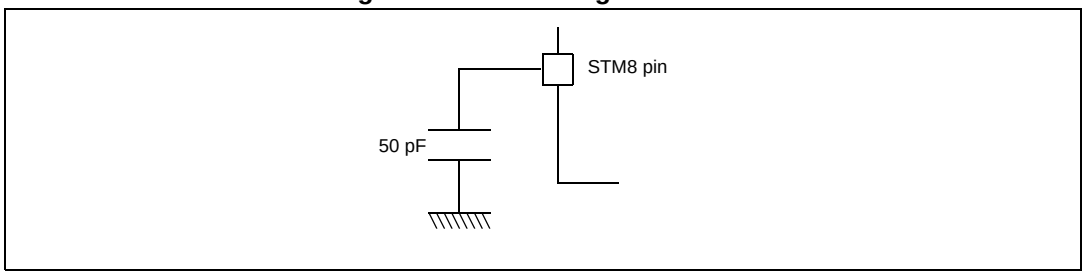
1. Accessible by debug module only
2. Product dependent value, see *Figure 8: Memory map*.

10.1.5 Pin loading conditions

10.1.6 Loading capacitor

The loading conditions used for pin parameter measurement are shown in *Figure 10*.

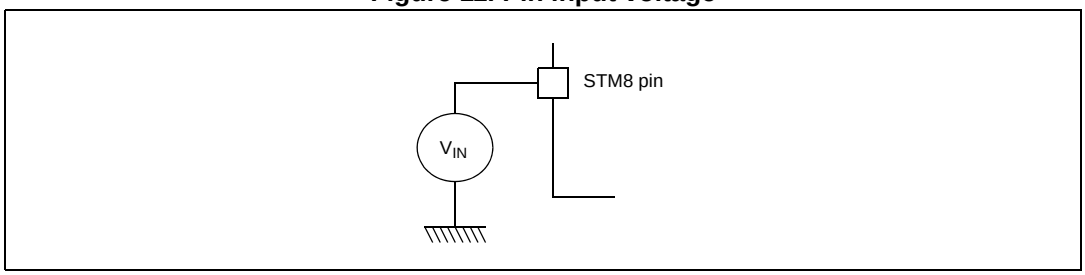
Figure 10. Pin loading conditions



10.1.7 Pin input voltage

The input voltage measurement on a pin of the device is described in *Figure 11*.

Figure 11. Pin input voltage



10.2 Absolute maximum ratings

Stresses above those listed as 'absolute maximum ratings' may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 15. Voltage characteristics

Symbol	Ratings	Min	Max	Unit
V_{DDx} - V_{SS}	Supply voltage (including V_{DDA} and V_{DDIO})(1)	-0.3	6.5	V
V_{IN}	Input voltage on true open drain pins (PE1, PE2)(2)	V_{SS} - 0.3	6.5	
	Input voltage on any other pin(2)	V_{SS} - 0.3	V_{DD} + 0.3	
$\lvert V_{DDx} - V_{DD} \rvert$	Variations between different power pins		50	mV
$\lvert V_{SSx} - V_{SS} \rvert$	Variations between all the different ground pins		50	
V_{ESD}	Electrostatic discharge voltage	see *Absolute maximum ratings (electrical sensitivity) on page 89*		

1. All power (V_{DD}, V_{DDIO}, V_{DDA}) and ground (V_{SS}, V_{SSIO}, V_{SSA}) pins must always be connected to the external power supply
2. $I_{INJ(PIN)}$ must never be exceeded. This is implicitly insured if V_{IN} maximum is respected. If V_{IN} maximum cannot be respected, the injection current must be limited externally to the $I_{INJ(PIN)}$ value. A positive injection is induced by $V_{IN}>V_{DD}$ while a negative injection is induced by $V_{IN}<V_{SS}$. For true open-drain pads, there is no positive injection current, and the corresponding V_{IN} maximum must always be respected

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.

Current consumption curves

Figure 14 and *Figure 15* show typical current consumption measured with code executing in RAM.

Figure 14. Typ. $I_{DD(RUN)}$ vs V_{DD}, HSI RC osc, f_{CPU} = 16 MHz

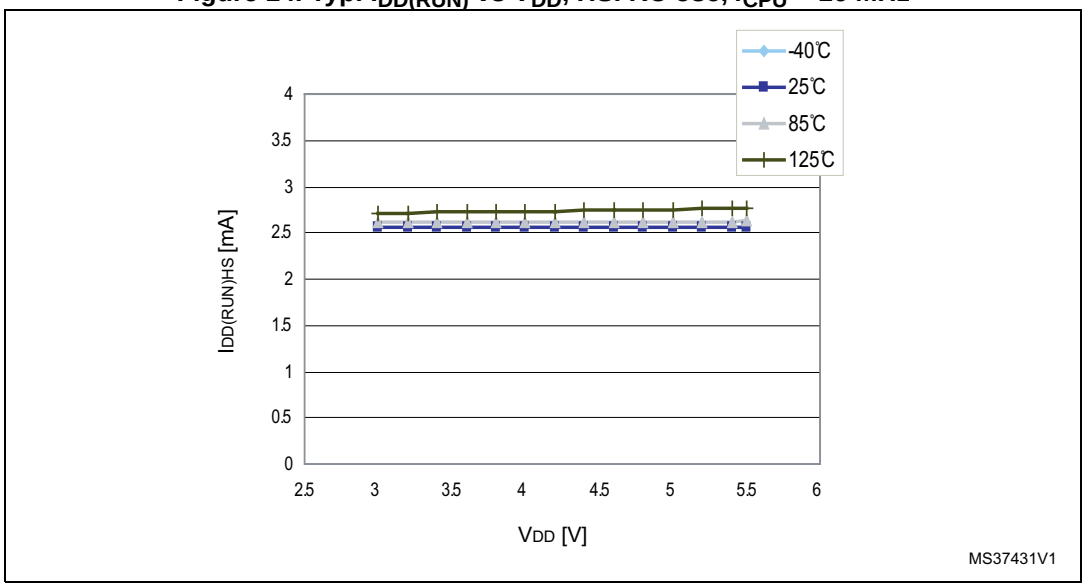
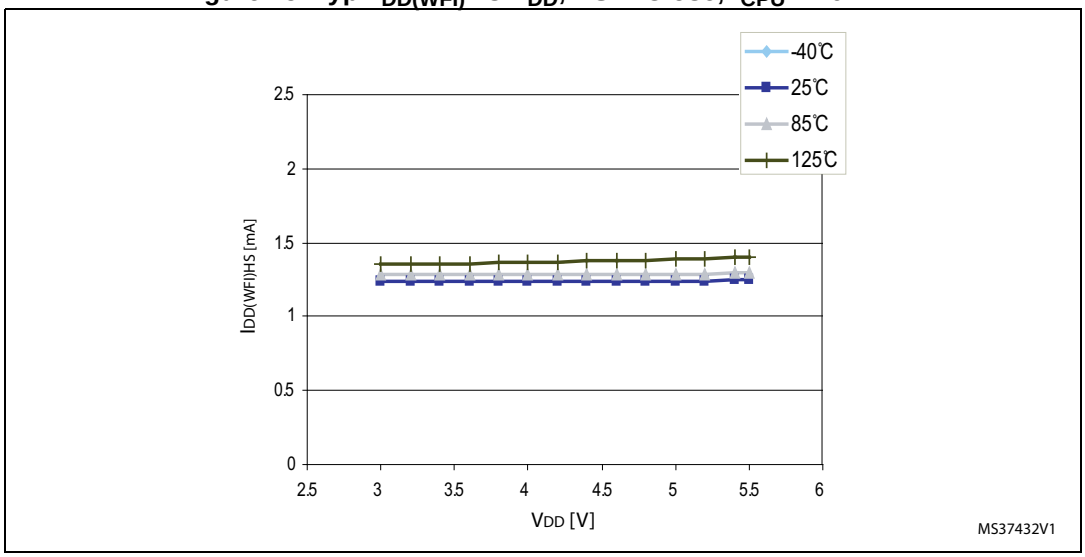


Figure 15. Typ. $I_{DD(WFI)}$ vs V_{DD}, HSI RC osc, f_{CPU} = 16 MHz



10.3.5 Memory characteristics

RAM and hardware registers

Table 35. RAM and hardware registers

Symbol	Parameter	Conditions	Min	Unit
V_{RM}	Data retention mode(1)	Halt mode (or reset)	V_{IT-max}(2)	V

1. Minimum supply voltage without losing data stored in RAM (in halt mode or under reset) or in hardware registers (only in halt mode). Guaranteed by design, not tested in production.
2. Refer to *Table 19 on page 57* for the value of V_{IT-max}.

Flash program memory/data EEPROM memory

General conditions: T_A = -40 to 125 °C.

Table 36. Flash program memory/data EEPROM memory

Symbol	Parameter	Conditions	Min(1)	Typ	Max	Unit
V_{DD}	Operating voltage (all modes, execution/write/erase)	$f_{CPU} \leq$ 24 MHz	2.95		5.5	V
t_{prog}	Standard programming time (including erase) for byte/word/block (1 byte/4 bytes/128 bytes)			6	6.6	ms
	Fast programming time for 1 block (128 bytes)			3	3.3	ms
t_{erase}	Erase time for 1 block (128 bytes)			3	3.3	ms
N_{RW}	Erase/write cycles(2) (program memory)	T_A = 85 °C	10 k			cycles
	Erase/write cycles (data memory)(2)	T_A = 125 ° C	300 k	1M		
t_{RET}	Data retention (program memory) after 10 k erase/write cycles at T_A = 85 °C	T_{RET} = 55° C	20			years
	Data retention (data memory) after 10 k erase/write cycles at T_A = 85 °C	T_{RET} = 55° C	20			
	Data retention (data memory) after 300k erase/write cycles at T_A = 125 °C	T_{RET} = 85° C	1			
I_{DD}	Supply current (Flash programming or erasing for 1 to 128 bytes)			2		mA

1. Data based on characterization results, not tested in production.
2. The physical granularity of the memory is 4 bytes, so cycling is performed on 4 bytes even when a write/erase operation addresses a single byte.

10.3.6 I/O port pin characteristics

General characteristics

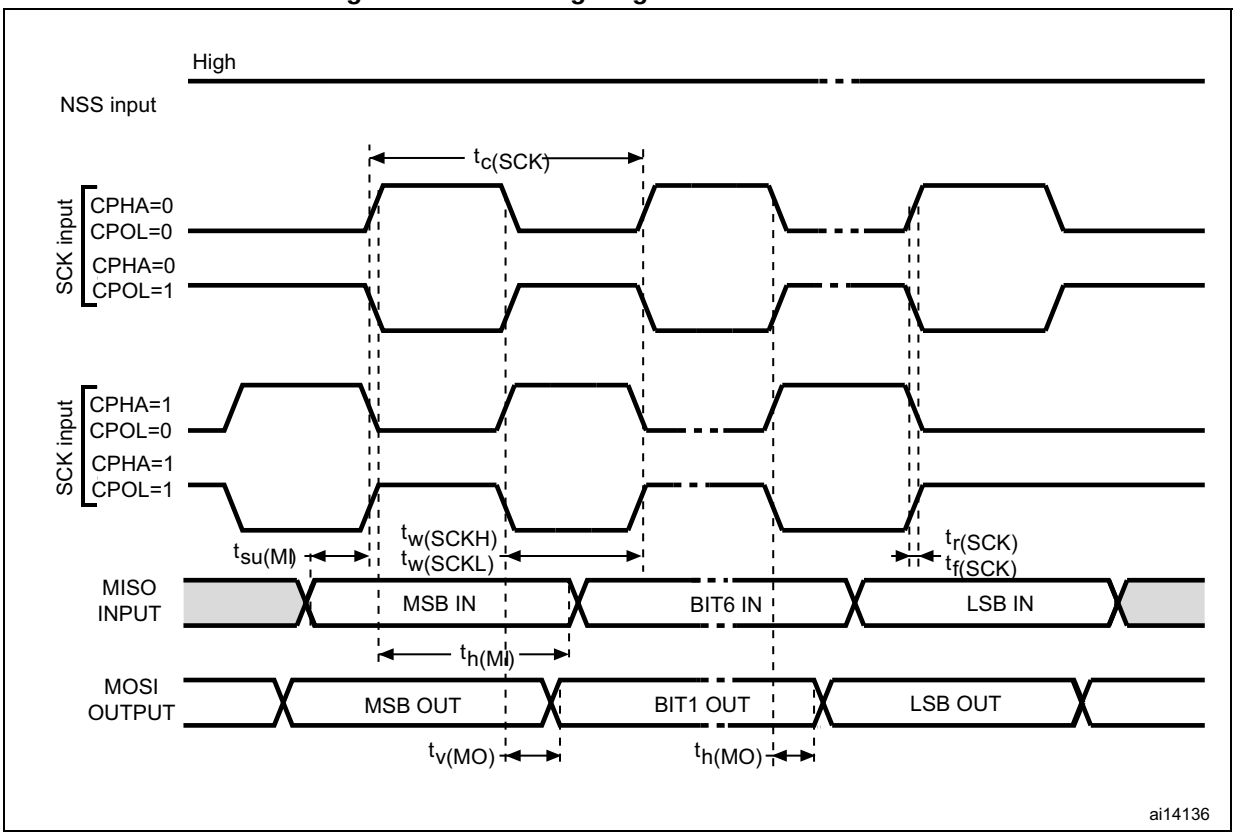
Subject to general operating conditions for V_{DD} and T_A unless otherwise specified. All unused pins must be kept at a fixed voltage: using the output mode of the I/O for example or an external pull-up or pull-down resistor.

Table 37. I/O static characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{IL}	Input low level voltage	V_{DD} = 5 V	-0.3		0.3 x V_{DD}	V
V_{IH}	Input high level voltage		0.7 x V_{DD}		V_{DD} + 0.3 V	
V_{hys}	Hysteresis(1)			700		mV
R_{pu}	Pull-up resistor	V_{DD} = 5 V, V_{IN} = V_{SS}	30	55	80	kΩ
t_R, t_F	Rise and fall time (10% - 90%)	Fast I/Os Load = 50 pF			20 (2)	ns
		Standard and high sink I/Os Load = 50 pF			125 (2)	
		Fast I/Os Load = 20 pF			35(3)	
		Standard and high sink I/Os Load = 20 pF			125(3)	
I_{lkg}	Input leakage current, analog and digital	$V_{SS} \leq V_{IN} \leq V_{DD}$			±1	µA
$I_{lkg\ ana}$	Analog input leakage current	$V_{SS} \leq V_{IN} \leq V_{DD}$			±250 (2)	nA
$I_{lkg(inj)}$	Leakage current in adjacent I/O(2)	Injection current ±4 mA			±1(2)	µA

1. Hysteresis voltage between Schmitt trigger switching levels. Based on characterization results, not tested in production.
2. Data based on characterization results, not tested in production.
3. Guaranteed by design.

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

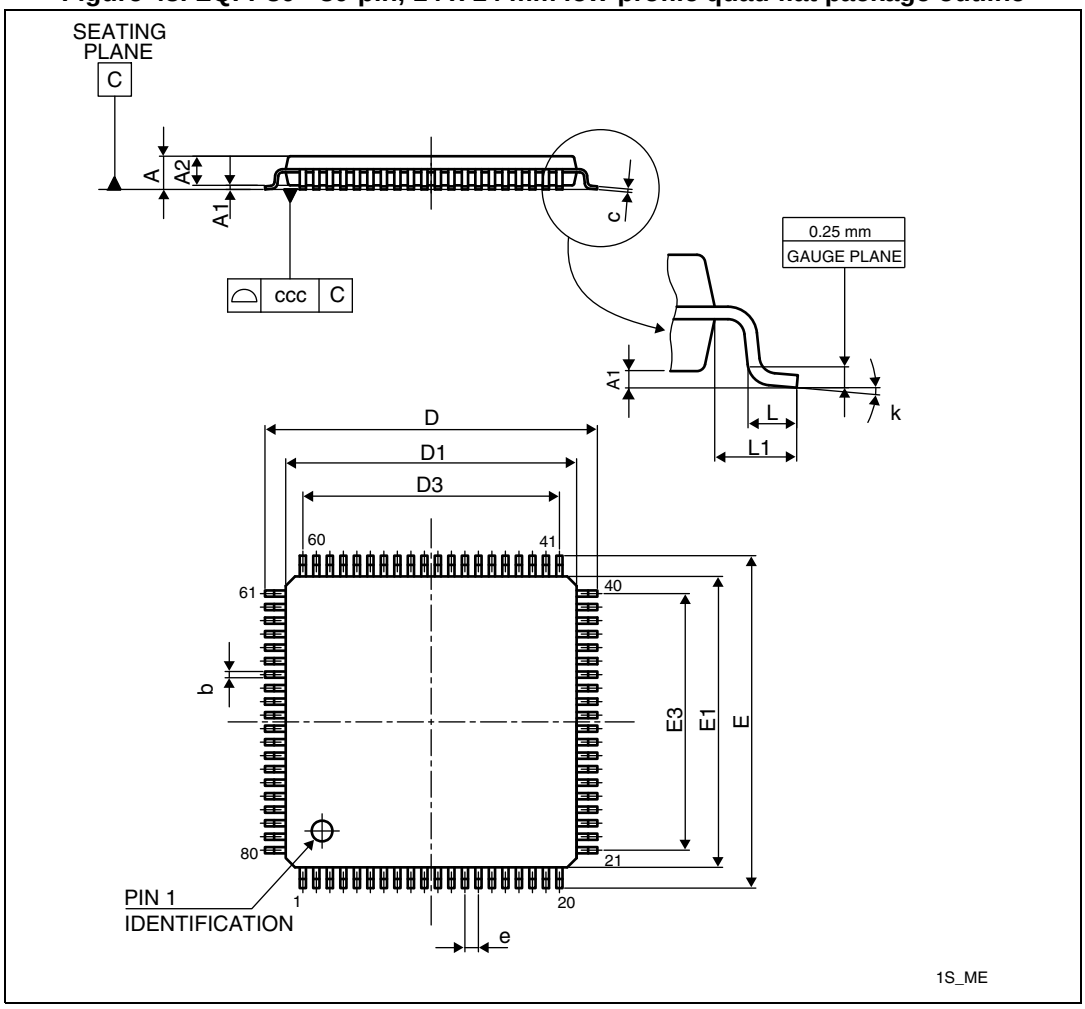


1. Measurement points are done at CMOS levels: 0.3 V_{DD} and 0.7 V_{DD}.

11.1 Package information

11.1.1 LQFP80 package information

Figure 43. LQFP80 - 80-pin, 14 x 14 mm low-profile quad flat package outline



1. Drawing is not to scale.

Table 51. LQFP80 - 80-pin, 14 x 14 mm low-profile quad flat package mechanical data(1)

Symbol	millimeters			inches		
	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	0.220	0.320	0.380	0.0087	0.0126	0.0150
c	0.090	-	0.200	0.0035	-	0.0079

11.1.2 LQFP64 package information

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

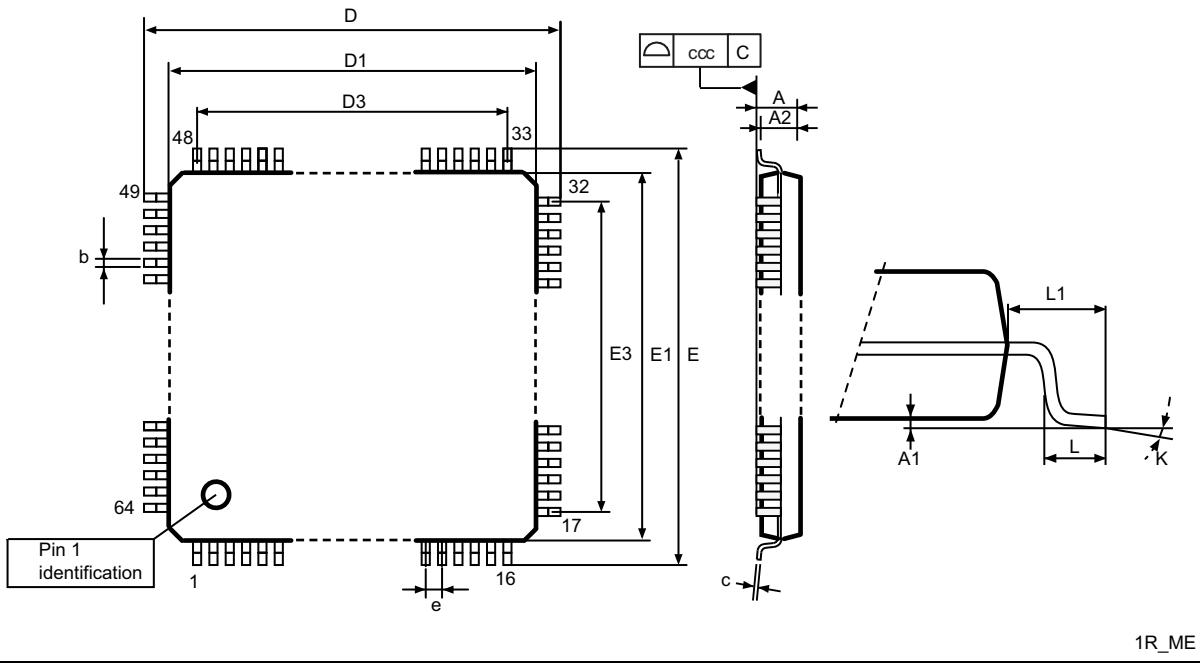


Table 52. LQFP64 - 64-pin, 14 x 14 mm low-profile quad flat package mechanical data

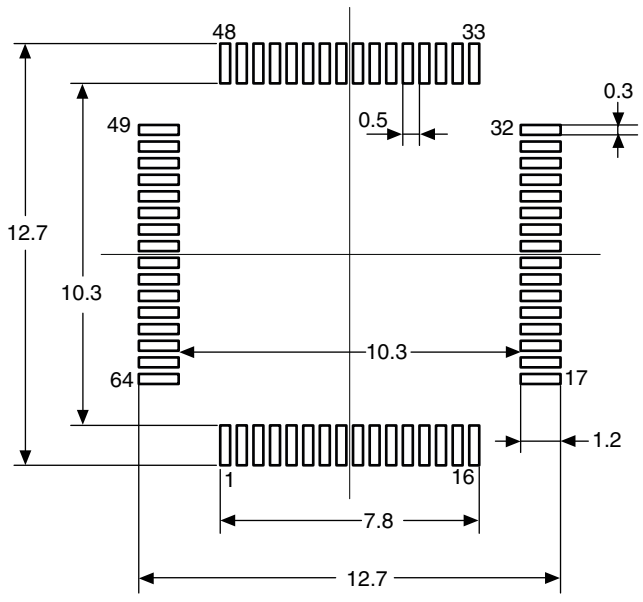
Symbol	mm			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A			1.600			0.0630
A1	0.050		0.150	0.0020		0.0059
A2	1.350	1.400	1.450	0.0531	0.0551	0.0571
b	0.300	0.370	0.450	0.0118	0.0146	0.0177
C	0.090		0.200	0.0035		0.0079
D	15.800	16.000	16.200	0.6220	0.6299	0.6378
D1	13.800	14.000	14.200	0.5433	0.5512	0.5591
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	

Table 53. LQFP64 - 64-pin, 10 x 10 mm low-profile quad flat package mechanical data (continued)

Symbol	mm			inches[1]		
	Min	Typ	Max	Min	Typ	Max
D	-	12.000	-	-	0.4724	-
D1	-	10.000	-	-	0.3937	-
D3	-	7.500	-	-	0.2953	-
E	-	12.000	-	-	0.4724	-
E1	-	10.000	-	-	0.3937	-
E3	-	7.500	-	-	0.2953	-
e	-	0.500	-	-	0.0197	-
θ	0°	3.5°	7°	0°	3.5°	7°
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1	-	1.000	-	-	0.0394	-
ccc	-	-	0.080	-	-	0.0031

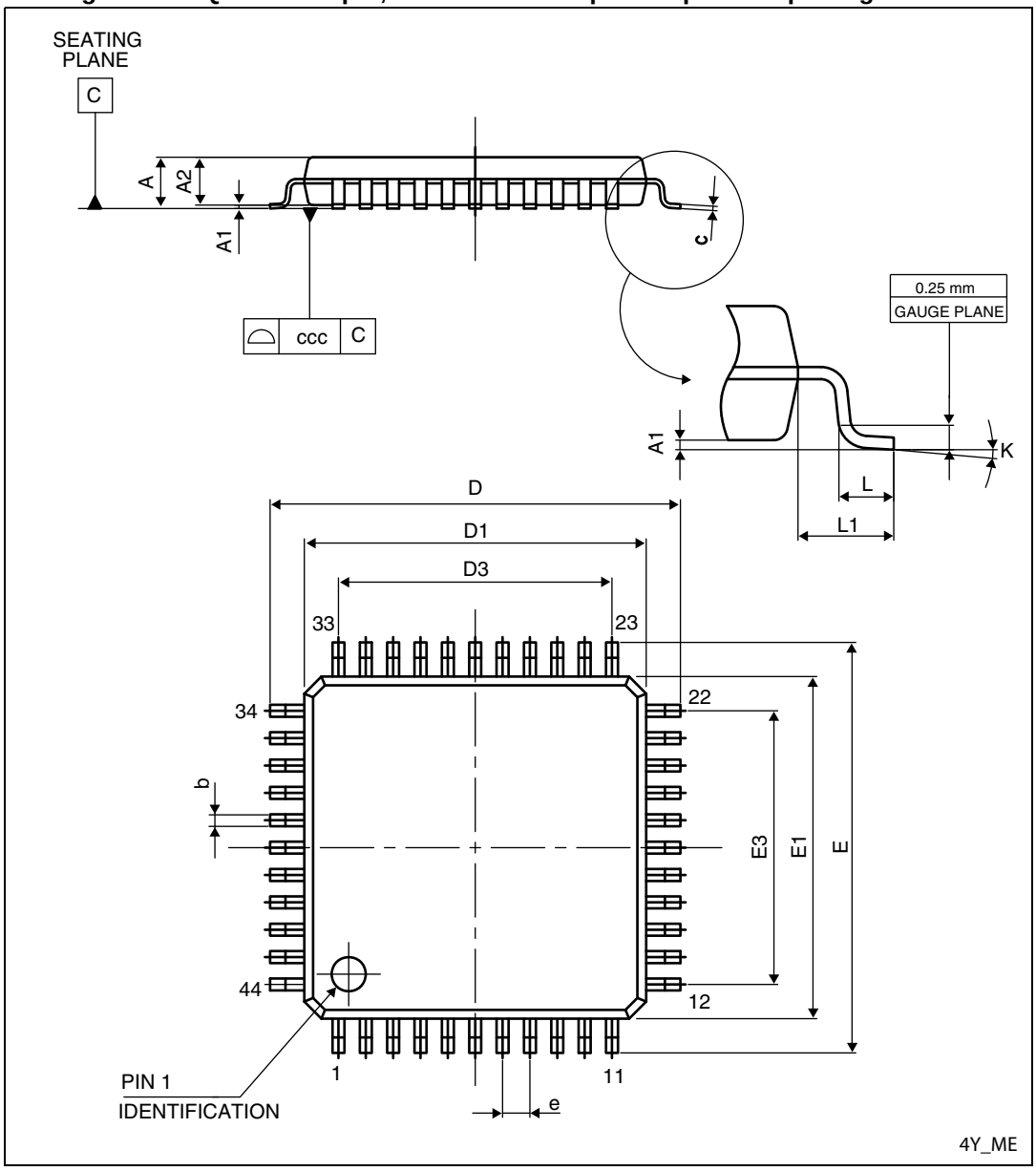
1. Values in inches are converted from mm and rounded to four decimal places.

Figure 48. LQFP64 - 64-pin, 10 x 10 mm low-profile quad flat recommended footprint



11.1.4 LQFP44 package information

Figure 53. LQFP44 - 44-pin, 10 x 10 mm low-profile quad flat package outline



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.

12.2 Software tools

STM8 development tools are supported by a complete, free software package from STMicroelectronics that includes ST Visual Develop (STVD) IDE and the ST Visual Programmer (STVP) software interface. STVD provides seamless integration of the Cosmic and Raisonance C compilers for STM8. A free version that outputs up to 32 Kbytes of code is available.

12.2.1 STM8 toolset

STM8 toolset with STVD integrated development environment and STVP programming software is available for free download at *www.st.com/mcu*. This package includes:

ST Visual Develop – Full-featured integrated development environment from ST, featuring

- Seamless integration of C and ASM toolsets
- Full-featured debugger
- Project management
- Syntax highlighting editor
- Integrated programming interface
- Support of advanced emulation features for STice such as code profiling and coverage

ST Visual Programmer (STVP) – Easy-to-use, unlimited graphical interface allowing read, write and verification of the STM8 microcontroller Flash program memory, data EEPROM and option bytes. STVP also offers project mode for saving programming configurations and automating programming sequences.

12.2.2 C and assembly toolchains

Control of C and assembly toolchains is seamlessly integrated into the STVD integrated development environment, making it possible to configure and control the building of the application directly from an easy-to-use graphical interface.

Available toolchains include:

- ***Cosmic C compiler for STM8*** – One free version that outputs up to 32 Kbytes of code is available. For more information, see www.cosmic-software.com.
- ***Raisonance C compiler for STM8*** – One free version that outputs up to 32 Kbytes of code. For more information, see www.raisonance.com.
- ***STM8 assembler linker*** – Free assembly toolchain included in the STVD toolset, which allows you to assemble and link the application source code.

12.3 Programming tools

During the development cycle, STice provides in-circuit programming of the STM8 Flash microcontroller on the application board via the SWIM protocol. Additional tools are to include a low-cost in-circuit programmer as well as ST socket boards, which provide dedicated programming platforms with sockets for programming the STM8.

For production environments, programmers will include a complete range of gang and automated programming solutions from third-party tool developers already supplying programmers for the STM8 family.