



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Obsolete
Core Processor	ST7
Core Size	8-Bit
Speed	8MHz
Connectivity	CANbus, LINbusSCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3.8V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f561k9tatr

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

2 PIN DESCRIPTION

57

Figure 2. LQFP 64-Pin Package Pinout



CENTRAL PROCESSING UNIT (Cont'd)

Condition Code Register (CC)

Read/Write

Reset Value: 111x1xxx

7							0
1	1	11	Н	10	Ν	Z	С

The 8-bit Condition Code register contains the interrupt masks and four flags representative of the result of the instruction just executed. This register can also be handled by the PUSH and POP instructions.

These bits can be individually tested and/or controlled by specific instructions.

Arithmetic Management Bits

Bit $4 = \mathbf{H}$ Half carry.

This bit is set by hardware when a carry occurs between bits 3 and 4 of the ALU during an ADD or ADC instructions. It is reset by hardware during the same instructions.

0: No half carry has occurred. 1: A half carry has occurred.

This bit is tested using the JRH or JRNH instruction. The H bit is useful in BCD arithmetic subroutines.

Bit 2 = N Negative.

This bit is set and cleared by hardware. It is representative of the result sign of the last arithmetic, logical or data manipulation. It's a copy of the result 7^{th} bit.

0: The result of the last operation is positive or null.

1: The result of the last operation is negative (that is, the most significant bit is a logic 1).

This bit is accessed by the JRMI and JRPL instructions.

Bit 1 = **Z** Zero.

This bit is set and cleared by hardware. This bit indicates that the result of the last arithmetic, logical or data manipulation is zero.

- 0: The result of the last operation is different from zero.
- 1: The result of the last operation is zero.

This bit is accessed by the JREQ and JRNE test instructions.

Bit 0 = C Carry/borrow.

This bit is set and cleared by hardware and software. It indicates an overflow or an underflow has occurred during the last arithmetic operation.

0: No overflow or underflow has occurred.

1: An overflow or underflow has occurred.

This bit is driven by the SCF and RCF instructions and tested by the JRC and JRNC instructions. It is also affected by the "bit test and branch", shift and rotate instructions.

Interrupt Management Bits

Bit 5,3 = 11, 10 Interrupt

The combination of the I1 and I0 bits gives the current interrupt software priority.

Interrupt Software Priority	11	10
Level 0 (main)	1	0
Level 1	0	1
Level 2	0	0
Level 3 (= interrupt disable)	1	1

These two bits are set/cleared by hardware when entering in interrupt. The loaded value is given by the corresponding bits in the interrupt software priority registers (IxSPR). They can be also set/ cleared by software with the RIM, SIM, IRET, HALT, WFI and PUSH/POP instructions.

See the interrupt management chapter for more details.

57

POWER SAVING MODES (Cont'd)

8.6.1 Register Description

AWUFH CONTROL/STATUS REGISTER (AWUCSR)

Read/Write (except bit 2 read only) Reset Value: 0000 0000 (00h)

7							0
0	0	0	0	0	AWU F	AWU M	AWU EN

Bits 7:3 = Reserved.

Bit 2 = AWUF Auto Wake-Up Flag

This bit is set by hardware when the AWU module generates an interrupt and cleared by software on reading AWUCSR.

0: No AWU interrupt occurred

1: AWU interrupt occurred

Bit 1 = **AWUM** Auto Wake-Up Measurement This bit enables the AWU RC oscillator and connects its output to the ICAP1 input of the 16-bit timer. This allows the timer to be used to measure the AWU RC oscillator dispersion and then compensate this dispersion by providing the right value in the AWUPR register.

0: Measurement disabled

1: Measurement enabled

Bit 0 = **AWUEN** Auto Wake-Up From Halt Enabled This bit enables the Auto Wake-Up From Halt feature: once HALT mode is entered, the AWUFH wakes up the microcontroller after a time delay defined by the AWU prescaler value. It is set and cleared by software.

Table 11. AWU Register Map and Reset Values

0: AWUFH	(Auto	Wake-Up	From	Halt)	mode	disa-
bled		-				

1: AWUFH (Auto Wake-Up From Halt) mode enabled

AWUFH PRESCALER REGISTER (AWUPR) Read/Write

Reset Value: 1111 1111 (FFh)

7							0
AWU							
PR7	PR6	PR5	PR4	PR3	PR2	PR1	PR0

Bits 7:0 = **AWUPR[7:0]** Auto Wake-Up Prescaler These 8 bits define the AWUPR Dividing factor (as explained below:

AWUPR[7:0]	Dividing factor
00h	Forbidden (See note)
01h	1
C FEh	254
FFh	255

In AWU mode, the period that the MCU stays in Halt Mode (t_{AWU} in Figure 30) is defined by

$$^{t}AWU = 64 \times AWUPR \times \frac{1}{f_{AWURC}} + ^{t}RCSTRT$$

This prescaler register can be programmed to modify the time that the MCU stays in Halt mode before waking up automatically.

Note: If 00h is written to AWUPR, depending on the product, an interrupt is generated immediately after a HALT instruction or the AWUPR remains unchanged.

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
002Bh	AWUCSR Reset Value	0	0	0	0	0	AWUF 0	AWUM 0	AWUEN 0
002Ch	AWUPR Reset Value	AWUPR7 1	AWUPR6 1	AWUPR5 1	AWUPR4 1	AWUPR3 1	AWUPR2 1	AWUPR1 1	AWUPR0 1

I/O PORTS (Cont'd)

CAUTION: The alternate function must not be activated as long as the pin is configured as input with interrupt, in order to avoid generating spurious interrupts.

Analog alternate function

When the pin is used as an ADC input, the I/O must be configured as floating input. The analog multiplexer (controlled by the ADC registers) switches the analog voltage present on the selected pin to the common analog rail which is connected to the ADC input.

It is recommended not to change the voltage level or loading on any port pin while conversion is in progress. Furthermore it is recommended not to have clocking pins located close to a selected analog pin.

WARNING: The analog input voltage level must be within the limits stated in the absolute maximum ratings.

9.3 I/O PORT IMPLEMENTATION

<u>لرک</u>

The hardware implementation on each I/O port depends on the settings in the DDR and OR registers and specific feature of the I/O port such as ADC Input or true open drain.

Switching these I/O ports from one state to another should be done in a sequence that prevents unwanted side effects. Recommended safe transitions are illustrated in Figure 33 on page 49. Other transitions are potentially risky and should be avoided, since they are likely to present unwanted side-effects such as spurious interrupt generation.

Figure 33. Interrupt I/O Port State Transitions



9.4 LOW POWER MODES

Mode	Description
WAIT	No effect on I/O ports. External interrupts cause the device to exit from WAIT mode.
HALT	No effect on I/O ports. External interrupts cause the device to exit from HALT mode.

9.5 INTERRUPTS

The external interrupt event generates an interrupt if the corresponding configuration is selected with DDR and OR registers and the interrupt mask in the CC register is not active (RIM instruction).

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
External interrupt on selected external event	-	DDRx ORx	Yes	

I/O PORTS (Cont'd)

Table 15. I/O Port Register Map and Reset Values

(Hex.)	Register Label	7	6	5	4	3	2	1	0
Rese	t Value	0	0	0	0	0	0	0	0
0000h	PADR								
0001h	PADDR	MSB							LSB
0002h	PAOR	-							_
0003h	PBDR								
0004h	PBDDR	MSB						1	LSB
0005h	PBOR							× \	51
0006h	PCDR								
0007h	PCDDR	MSB							LSB
0008h	PCOR						.0		
0009h	PDDR								
000Ah	PDDDR	MSB							LSB
000Bh	PDOR					×0			
000Ch	PEDR					0,			
000Dh	PEDDR	MSB							LSB
000Eh	PEOR				5				
000Fh	PFDR				Y				
0010h	PFDDR	MSB							LSB
0011h	PFOR								
Obsolf	stePr	odu	cils						

WINDOW WATCHDOG (Cont'd)

57

10.1.5 How to Program the Watchdog Timeout

Figure 2 shows the linear relationship between the 6-bit value to be loaded in the Watchdog Counter (CNT) and the resulting timeout duration in milliseconds. This can be used for a quick calculation without taking the timing variations into account. If

more precision is needed, use the formulae in Figure 3.

Caution: When writing to the WDGCR register, always write 1 in the T6 bit to avoid generating an immediate reset.



Figure 35. Approximate Timeout Duration

WINDOW WATCHDOG (Cont'd)

10.1.9 Interrupts

None.

10.1.10 Register Description CONTROL REGISTER (WDGCR)

Read/Write

Reset Value: 0111 1111 (7Fh)

7							0
WDGA	Т6	T5	T4	Т3	T2	T1	то

Bit 7 = WDGA Activation bit.

This bit is set by software and only cleared by hardware after a reset. When WDGA = 1, the watchdog can generate a reset.

0: Watchdog disabled

1: Watchdog enabled

Note: This bit is not used if the hardware watchdog option is enabled by option byte.

Bits 6:0 = **T[6:0]** 7-bit counter (MSB to LSB). These bits contain the value of the watchdog counter. It is decremented every 16384 f_{OSC2} cycles (approx.). A reset is produced when it rolls over from 40h to 3Fh (T6 becomes cleared).

WINDOW REGISTER (WDGWR)

Read/Write

Reset Value: 0111 1111 (7Fh)

7							0
-	W6	W5	W4	WЗ	W2	W1	WO

Bit 7 = Reserved

Bits 6:0 = **W[6:0]** *7-bit window value* These bits contain the window value to be compared to the downcounter.

Figure 38. Watchdog Timer Register Map and Reset Values

	Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
	C2E	WDGCR	WDGA	Т6	T5	T4	Т3	T2	T1	TO
	0-21	Reset Value	0	1	1	1	1	1	1	1
\bigcirc	20	WDGWR	-	W6	W5	W4	W3	W2	W1	W0
	30	Reset Value	0	1	1	1	1	1	1	1

16-BIT TIMER (Cont'd)

10.4.3.4 Output Compare

In this section, the index, *i*, may be 1 or 2 because there are two output compare functions in the 16-bit timer.

This function can be used to control an output waveform or indicate when a period of time has elapsed.

When a match is found between the Output Compare register and the free running counter, the output compare function:

- Assigns pins with a programmable value if the OC*i*E bit is set
- Sets a flag in the status register
- Generates an interrupt if enabled

Two 16-bit registers Output Compare Register 1 (OC1R) and Output Compare Register 2 (OC2R) contain the value to be compared to the counter register each timer clock cycle.

	MS Byte	LS Byte
OC <i>i</i> R	OC <i>i</i> HR	OC <i>i</i> LR

These registers are readable and writable and are not affected by the timer hardware. A reset event changes the OC*i*R value to 8000h.

Timing resolution is one count of the free running counter: $(f_{CPU/CC[1:0]})$.

Procedure:

To use the output compare function, select the following in the CR2 register:

- Set the OC*i*E bit if an output is needed then the OCMP*i* pin is dedicated to the output compare *i* signal.
- Select the timer clock (CC[1:0]) (see Table 17 Clock Control Bits).

And select the following in the CR1 register:

- Select the OLVL*i* bit to applied to the OCMP*i* pins after the match occurs.
- Set the OCIE bit to generate an interrupt if it is needed.

When a match is found between OC*i*R register and CR register:

OCF*i* bit is set.

- The OCMP*i* pin takes OLVL*i* bit value (OCMP*i* pin latch is forced low during reset).
- A timer interrupt is generated if the OCIE bit is set in the CR1 register and the I bit is cleared in the CC register (CC).

The OC*i*R register value required for a specific timing application can be calculated using the following formula:

$$\Delta \text{ OC} i R = \frac{\Delta t * f_{CPU}}{\text{PRESC}}$$

Where:

- Δt = Output compare period (in seconds)
- f_{CPU} = CPU clock frequency (in hertz)
- PRESC = Timer prescaler factor (2, 4 or 8 depending on CC[1:0] bits, see Table 17 Clock Control Bits)

If the timer clock is an external clock, the formula is:

$$\Delta \text{ OC}_{i} R = \Delta t * f_{EXT}$$

Where:

 Δt = Output compare period (in seconds)

f_{EXT} = External timer clock frequency (in hertz)

Clearing the output compare interrupt request (that is, clearing the OCF*i* bit) is done by:

- 1. Reading the SR register while the OCF*i* bit is set.
- 2. An access (read or write) to the OCiLR register.

The following procedure is recommended to prevent the OCF*i* bit from being set between the time it is read and the write to the OC*i*R register:

- Write to the OC*i*HR register (further compares are inhibited).
- Read the SR register (first step of the clearance of the OCF*i* bit, which may be already set).
- Write to the OCiLR register (enables the output compare function and clears the OCF*i* bit).

<u>(۲</u>/

8-BIT TIMER (Cont'd)

10.5.3.5 Pulse Width Modulation Mode

Pulse Width Modulation (PWM) mode enables the generation of a signal with a frequency and pulse length determined by the value of the OC1R and OC2R registers.

Pulse Width Modulation mode uses the complete Output Compare 1 function plus the OC2R register, and so this functionality can not be used when PWM mode is activated.

In PWM mode, double buffering is implemented on the output compare registers. Any new values written in the OC1R and OC2R registers are taken into account only at the end of the PWM period (OC2) to avoid spikes on the PWM output pin (OCMP1).

Procedure

To use pulse width modulation mode:

- 1. Load the OC2R register with the value corresponding to the period of the signal using the formula in the opposite column.
- 2. Load the OC1R register with the value corresponding to the period of the pulse if (OLVL1 = 0 and OLVL2 = 1) using the formula in the opposite column.
- 3. Select the following in the CR1 register:
 - Using the OLVL1 bit, select the level to be applied to the OCMP1 pin after a successful comparison with the OC1R register.
 - Using the OLVL2 bit, select the level to be applied to the OCMP1 pin after a successful comparison with the OC2R register.
- 4. Select the following in the CR2 register:
 - Set OC1E bit: the OCMP1 pin is then dedicated to the output compare 1 function.
 - Set the PWM bit.
 - Select the timer clock (CC[1:0]) (see Table 19 Clock Control Bits).



If OLVL1 = 1 and OLVL2 = 0 the length of the positive pulse is the difference between the OC2R and OC1R registers.

If OLVL1 = OLVL2 a continuous signal will be seen on the OCMP1 pin.

The OC*i*R register value required for a specific timing application can be calculated using the following formula:

$$OC_{i}R Value = \frac{t \cdot f_{CPU}}{PBESC} - 5$$

Where:

t = Signal or pulse period (in seconds)

f_{CPU} = PLL output x2 clock frequency in hertz (or f_{OSC}/2 if PLL is not enabled)

PRESC = Timer prescaler factor (2, 4, 8 or 8000 depending on CC[1:0] bits, see Table 19 Clock Control Bits)

The Output Compare 2 event causes the counter to be initialized to FCh (See Figure 69)

Notes:

- 1. The OCF1 and OCF2 bits cannot be set by hardware in PWM mode therefore the Output Compare interrupt is inhibited.
- 2. The ICF1 bit is set by hardware when the counter reaches the OC2R value and can produce a timer interrupt if the ICIE bit is set and the I bit is cleared.
- 3. In PWM mode the ICAP1 pin can not be used to perform input capture because it is disconnected to the timer. The ICAP2 pin can be used to perform input capture (ICF2 can be set and IC2R can be loaded) but the user must take care that the counter is reset each period and ICF1 can also generates interrupt if ICIE is set.
- 4. When the Pulse Width Modulation (PWM) and One Pulse Mode (OPM) bits are both set, the PWM mode is the only active one.

<u>ل</u>رک

SERIAL PERIPHERAL INTERFACE (SPI) (cont'd)





SERIAL PERIPHERAL INTERFACE (cont'd)

10.6.3.1 Functional Description

A basic example of interconnections between a single master and a single slave is illustrated in Figure 71.

The MOSI pins are connected together and the MISO pins are connected together. In this way data is transferred serially between master and slave (most significant bit first).

The communication is always initiated by the master. When the master device transmits data to a slave device via MOSI pin, the slave device responds by sending data to the master device via the MISO pin. This implies full duplex communication with both data out and data in synchronized with the same clock signal (which is provided by the master device via the SCK pin).

To use a single data line, the MISO and MOSI pins must be connected at each node (in this case only simplex communication is possible).

Four possible data/clock timing relationships may be chosen (see Figure 74 on page 114) but master and slave must be programmed with the same timing mode.



Figure 71. Single Master/ Single Slave Application



LINSCI™ SERIAL COMMUNICATION INTERFACE (SCI Mode) (cont'd)

CONTROL REGISTER 2 (SCICR2)

Read/Write Reset Value: 0000 0000 (00h)

7							0
TIE	TCIE	RIE	ILIE	TE	RE	RWU ¹⁾	SBK ¹⁾

¹⁾This bit has a different function in LIN mode, please refer to the LIN mode register description.

Bit 7 = **TIE** *Transmitter interrupt enable*

This bit is set and cleared by software.

0: Interrupt is inhibited

1: In SCI interrupt is generated whenever TDRE = 1 in the SCISR register

Bit 6 = **TCIE** *Transmission complete interrupt enable*

This bit is set and cleared by software.

0: Interrupt is inhibited

1: An SCI interrupt is generated whenever TC = 1 in the SCISR register

Bit 5 = **RIE** *Receiver interrupt enable*

This bit is set and cleared by software.

- 0: Interrupt is inhibited
- 1: An SCI interrupt is generated whenever OR = 1 or RDRF = 1 in the SCISR register

Bit 4 = ILIE Idle line interrupt enable

This bit is set and cleared by software.

0: Interrupt is inhibited

1: An SCI interrupt is generated whenever IDLE = 1 in the SCISR register.

Bit 3 = **TE** Transmitter enable

This bit enables the transmitter. It is set and cleared by software.

0: Transmitter is disabled

1: Transmitter is enabled

Notes:

- During transmission, a "0" pulse on the TE bit ("0" followed by "1") sends a preamble (idle line) after the current word.
- When TE is set there is a 1 bit-time delay before the transmission starts.

Bit 2 = **RE** *Receiver enable*

This bit enables the receiver. It is set and cleared by software.

0: Receiver is disabled in the SCISR register

1: Receiver is enabled and begins searching for a start bit

Bit 1 = RWU Receiver wake-up

This bit determines if the SCI is in mute mode or not. It is set and cleared by software and can be cleared by hardware when a wake-up sequence is recognized.

- 0: Receiver in active mode
- 1: Receiver in mute mode

Notes:

- Before selecting Mute mode (by setting the RWU bit) the SCI must first receive a data byte, otherwise it cannot function in Mute mode with wakeup by Idle line detection.
- In Address Mark Detection Wake-Up configuration (WAKE bit = 1) the RWU bit cannot be modified by software while the RDRF bit is set.

Bit 0 = SBK Send break

This bit set is used to send break characters. It is set and cleared by software.

0: No break character is transmitted

1: Break characters are transmitted

Note: If the SBK bit is set to "1" and then to "0", the transmitter will send a BREAK word at the end of the current word.

DATA REGISTER (SCIDR)

Read/Write

7

Reset Value: Undefined

Contains the Received or Transmitted data character, depending on whether it is read from or written to.

DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0

The Data register performs a double function (read and write) since it is composed of two registers, one for transmission (TDR) and one for reception (RDR).

The TDR register provides the parallel interface between the internal bus and the output shift register (see Figure 1).

The RDR register provides the parallel interface between the input shift register and the internal bus (see Figure 1).



0

LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Mode) (cont'd)

10.7.9.4 LIN Error Detection

LIN Header Error Flag

The LIN Header Error Flag indicates that an invalid LIN Header has been detected.

When a LIN Header Error occurs:

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

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

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

LHE/OVR Error Conditions

When Auto Resynchronization is disabled (LASE bit = 0), the LHE flag detects:

- That the received LIN Synch Field is not equal to 55h.
- That an overrun occurred (as in standard SCI mode)
- Furthermore, if LHDM is set it also detects that a LIN Header Reception Timeout occurred (only if LHDM is set).

When the LIN auto-resynchronization is enabled (LASE bit = 1), the LHE flag detects:

- That the deviation error on the Synch Field is outside the LIN specification which allows up to +/-15.5% of period deviation between the slave and master oscillators.
- A LIN Header Reception Timeout occurred.
 If T_{HEADER} > T_{HEADER_MAX} then the LHE flag is set. Refer to Figure 6. (only if LHDM is set to 1)
- An overflow during the Synch Field Measurement, which leads to an overflow of the divider registers. If LHE is set due to this error then the SCI goes into a blocked state (LSF bit is set).
- That an overrun occurred on Fields other than the Synch Field (as in standard SCI mode)

Deviation Error on the Synch Field

The deviation error is checking by comparing the current baud rate (relative to the slave oscillator) with the received LIN Synch Field (relative to the master oscillator). Two checks are performed in parallel:

 The first check is based on a measurement between the first falling edge and the last falling edge of the Synch Field. Let us refer to this period deviation as D:

If the LHE flag is set, it means that:

D > 15.625%

If LHE flag is not set, it means that:

D < 16.40625%

If $15.625\% \le D < 16.40625\%$, then the flag can be either set or reset depending on the dephasing between the signal on the RDI line and the CPU clock.

 The second check is based on the measurement of each bit time between both edges of the Synch Field: this checks that each of these bit times is large enough compared to the bit time of the current baud rate.

When LHE is set due to this error then the SCI goes into a blocked state (LSF bit is set).

LIN Header Time-out Error

When the LIN Identifier Field Detection Method is used (by configuring LHDM to 1) or when LIN auto-resynchronization is enabled (LASE bit = 1), the LINSCI automatically monitors the T_{HEADER_MAX} condition given by the LIN protocol.

If the entire Header (up to and including the STOP bit of the LIN Identifier Field) is not received within the maximum time limit of 57 bit times then a LIN Header Error is signalled and the LHE bit is set in the SCISR register.

Figure 82. LIN Header Reception Timeout



The time-out counter is enabled at each break detection. It is stopped in the following conditions:

- A LIN Identifier Field has been received

- An LHE error occurred (other than a timeout error).

- A software reset of LSF bit (transition from high to low) occurred during the analysis of the LIN Synch Field or

If LHE bit is set due to this error during the LIN Synchr Field (if LASE bit = 1) then the SCI goes into a blocked state (LSF bit is set).

LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Mode) (cont'd)

10.7.9.9 Error due to LIN Synch measurement

The LIN Synch Field is measured over eight bit times.

This measurement is performed using a counter clocked by the CPU clock. The edge detections are performed using the CPU clock cycle.

This leads to a precision of 2 CPU clock cycles for the measurement which lasts 16*8*LDIV clock cycles.

Consequently, this error (D_{MEAS}) is equal to:

2 / (128*LDIV_{MIN}).

 $LDIV_{MIN}$ corresponds to the minimum LIN prescaler content, leading to the maximum baud rate, taking into account the maximum deviation of +/-15%.

10.7.9.10 Error due to Baud Rate Quantization

The baud rate can be adjusted in steps of 1 / (16 * LDIV). The worst case occurs when the "real" baud rate is in the middle of the step.

This leads to a quantization error (D_{QUANT}) equal to 1 / (2*16*LDIV_{MIN}).

10.7.9.11 Impact of Clock Deviation on Maximum Baud Rate

The choice of the nominal baud rate (LDIV_{NOM}) will influence both the quantization error (D_{QUANT}) and the measurement error (D_{MEAS}). The worst case occurs for LDIV_{MIN}.

Consequently, at a given CPU frequency, the maximum possible nominal baud rate (LPR_{MIN}) should be chosen with respect to the maximum tolerated deviation given by the equation:

D_{TRA} + 2 / (128*LDIV_{MIN}) + 1 / (2*16*LDIV_{MIN})

 $+ D_{REC} + D_{TCL} < 3.75\%$

Example:

A nominal baud rate of 20Kbits/s at T_{CPU} = 125ns (8 MHz) leads to LDIV_{NOM} = 25d.

LDIV_{MIN} = 25 - 0.15*25 = 21.25

D_{MEAS} = 2 / (128*LDIV_{MIN}) * 100 = 0.00073%

D_{QUANT} = 1 / (2*16*LDIV_{MIN}) * 100 = 0.0015%

LIN Slave systems

For LIN Slave systems (the LINE and LSLV bits are set), receivers wake up by LIN Synch Break or LIN Identifier detection (depending on the LHDM bit).

Hot Plugging Feature for LIN Slave Nodes

In LIN Slave Mute Mode (the LINE, LSLV and RWU bits are set) it is possible to hot plug to a network during an ongoing communication flow. In this case the SCI monitors the bus on the RDI line until 11 consecutive dominant bits have been detected and discards all the other bits received.



beCAN CONTROLLER (Cont'd)

10.9.8.2 Mailbox Registers

This chapter describes the registers of the transmit and receive mailboxes. Refer to Section 0.1.4.4 Message Storage for detailed register mapping.

Transmit and receive mailboxes have the same registers except:

- MCSR register in a transmit mailbox is replaced by MFMI register in a receive mailbox.
- A receive mailbox is always write protected.
- A transmit mailbox is write enable only while empty, corresponding TME bit in the CTPR register set.

MAILBOX CONTROL STATUS REGISTER (MCSR)

Read / Write

Reset Value: 0000 0000 (00h)

7							0
0	0	TERR	ALST	тхок	RQCP	ABRQ	TXRQ

Bits 7:6 = Reserved. Forced to 0 by hardware.

Bit 5 = **TERR** *Transmission Error* - Read

This bit is updated by hardware after each transmission attempt.

0: The previous transmission was successful

1: The previous transmission failed due to an error

Bit 4 = **ALST** Arbitration Lost - Read

This bit is updated by hardware after each transmission attempt.

0: The previous transmission was successful

1: The previous transmission failed due to an arbitration lost

Bit 3 = **TXOK** *Transmission OK*

- Read

The hardware updates this bit after each transmission attempt.

0: The previous transmission failed

1: The previous transmission was successful

Note: This bit has the same value as the corresponding TXOKx bit in the CTSR register.

Bit 2 = RQCP Request Completed

- Read/Clear

Set by hardware when the last request (transmit or abort) has been performed.

Cleared by software writing a "1" or by hardware on transmission request.

Note: This bit has the same value as the corresponding RQCPx bit of the CTSR register.

Clearing this bit clears all the status bits (TX-OK, ALST and TERR) in the MCSR register and the RQCP and TXOK bits in the CTSR register.

Bit 1 = **ABRQ** Abort Request for Mailbox - Read/Set

Set by software to abort the transmission request for the corresponding mailbox.

Cleared by hardware when the mailbox becomes empty.

Setting this bit has no effect when the mailbox is not pending for transmission.

Bit 0 = **TXRQ** *Transmit Mailbox Request*

- Read/Set

Set by software to request the transmission for the corresponding mailbox.

Cleared by hardware when the mailbox becomes empty.

Note: This register is implemented only in transmit mailboxes. In receive mailboxes, the MFMI register is mapped at this location.

57

Address (Hex.)	Register Name	7	6	5	4	3	2	1	0
7Eh	MTSLR	TIME7	TIME6	TIME5	TIME4	TIME3	TIME2	TIME1	TIME0
	Reset Value	x	x	x	x	x	x	x	x
7Fh	MTSHR	TIME15	TIME14	TIME13	TIME12	TIME11	TIME10	TIME9	TIME8
	Reset Value	x	x	x	x	x	x	x	x

 Table 34. beCAN Filter Configuration Page - Register Map and Reset Values

Address (Hex.)	Register Name	7	6	5	4	3	2	1	50
	CESR		LEC2	LEC1	LEC0		BOFF	EPVF	EWGF
70h	Reset Value	0	0	0	0	0	0	0	0
	CEIER	ERRIE			LECIE		BOFIE	EPVIE	EWGIE
/1n	Reset Value	0	0	0	0	0	0	0	0
701	TECR	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0
72n	Reset Value	0	0	0	0	0	0	0	0
706	RECR	REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0
73N	Reset Value	0	0	0	0	0	0	0	0
746	CBTR0	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
74N	Reset Value	0	0	0	0	0	0	0	0
754	CBTR1		BS22	BS21	BS20	BS13	BS12	BS11	BS10
75N	Reset Value	0	0	1	0	0	0	1	1
76h	Reserved	x OX	×	x	x	x	x	x	x
77h	Reserved	x	x	x	x	x	x	x	x
701	CFMR0	FMH3	FML3	FMH2	FML2	FMH1	FML1	FMH0	FML0
780	Reset Value	0	0	0	0	0	0	0	0
7 9h	CFMR1 Reset Value	0	0	0	0	FMH5 0	FML5 0	FMH4 0	FML4 0
	CFCB0	FFA1	ESC11	ESC10	FACT1	FFA0	FSC01	FSC00	FACT0
7Ah	Reset Value	0	0	0	0	0	0	0	0
	CFCR1	FFA3	FSC31	FSC30	FACT3	FFA2	FSC21	FSC20	FACT2
7Bh	Reset Value	0	0	0	0	0	0	0	0
	CFCR2	FFA5	FSC51	FSC50	FACT5	FFA4	FSC41	FSC40	FACT4
7Ch	Reset Value	0	0	0	0	0	0	0	0



10.10 10-BIT A/D CONVERTER (ADC)

10.10.1 Introduction

The on-chip Analog to Digital Converter (ADC) peripheral is a 10-bit, successive approximation converter with internal sample and hold circuitry. This peripheral has up to 16 multiplexed analog input channels (refer to device pin out description) that allow the peripheral to convert the analog voltage levels from up to 16 different sources.

The result of the conversion is stored in a 10-bit Data Register. The A/D converter is controlled through a Control/Status Register.

10.10.2 Main Features

- 10-bit conversion
- Up to 16 channels with multiplexed input
- Linear successive approximation
- Data register (DR) which contains the results
- Conversion complete status flag
- On/off bit (to reduce consumption)

The block diagram is shown in Figure 116.

Figure 116. ADC Block Diagram

10.10.3 Functional Description

10.10.3.1 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 V_{DDA} (high-level voltage reference) then the conversion result is FFh in the ADCDRH register and 03h in the ADCDRL register (without overflow indication).

If the input voltage (V_{AIN}) is lower than V_{SSA} (low-level voltage reference) then the conversion result in the ADCDRH and ADCDRL registers is 00 00h.

The A/D converter is linear and the digital result of the conversion is stored in the ADCDRH and AD-CDRL registers. The accuracy of the conversion is described in the Electrical Characteristics 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.



EMC CHARACTERISTICS (Cont'd)

12.8.2 Electro Magnetic Interference (EMI)

Based on a simple application running on the product (toggling two LEDs through the I/O ports), the product is monitored in terms of emission. This emission test is in line with the norm SAE J 1752/ 3 which specifies the board and the loading of each pin.

Cumbal	Doromotor	Conditions	Monitored	Max vs. [t	losc/fcpu]	Unit
Symbol	Farameter	Conditions	Frequency Band	8/4 MHz	16/8 MHz	Unit
			0.1 MHz to 30 MHz	10	13	
		Flash devices: $V_{DD} = 5V$, $I_A = +25^{\circ}C$,	30 MHz to 130 MHz	12	19	dBμV
		conforming to SAE J 1752/3	130 MHz to 1 GHz	8	14	÷.
		3	SAE EMI Level	2.5	3	
			0.1 MHz to 30 MHz	31	32	5
S	Poak lovel1)	Flash devices: $V_{DD} = 5V$, $I_A = +25^{\circ}C$,	30 MHz to 130 MHz	32	37	dBμV
J EWI	i eakievei	conforming to SAE J 1752/3	130 MHz to 1 GHz	11	16	
			SAE EMI Level	3.0	3.5	-
			0.1 MHz to 30 MHz	10	18	
		ROM devices: $V_{DD} = 5V$, $T_A = +25^{\circ}C$,	30 MHz to 130 MHz	15	25	dBμV
		∟QFP64 package conforming to SAE J 1752/3	130 MHz to 1 GHz	-3	1	
			SAE EMI Level	2.0	2.5	-
Notes: I. Not tes	ted in product	ion.	,50°			
Notes: I. Not tes	ted in product	ion. Otucils)				



ADC CHARACTERISTICS (Cont'd)

12.13.0.1 Analog Power Supply and Reference Pins

Depending on the MCU pin count, the package may feature separate V_{DDA} and V_{SSA} analog power supply pins. These pins supply power to the A/D converter cell and function as the high and low reference voltages for the conversion. In smaller packages V_{DDA} and V_{SSA} pins are not available and the analog supply and reference pads are internally bonded to the V_{DD} and V_{SS} pins.

Separation of the digital and analog power pins allow board designers to improve A/D performance. Conversion accuracy can be impacted by voltage drops and noise in the event of heavily loaded or badly decoupled power supply lines (see Section 12.13.0.2 "General PCB Design Guidelines").

12.13.0.2 General PCB Design Guidelines

To obtain best results, some general design and layout rules should be followed when designing the application PCB to shield the noise-sensitive, analog physical interface from noise-generating CMOS logic signals.

 Use separate digital and analog planes. The analog ground plane should be connected to the digital ground plane via a single point on the PCB.

لركم

- Filter power to the analog power planes. It is recommended to connect capacitors, with good high frequency characteristics, between the power and ground lines, placing 0.1µF and optionally, if needed 10pF capacitors as close as possible to the ST7 power supply pins and a 1 to 10µF capacitor close to the power source (see Figure 144).

- The analog and digital power supplies should be connected in a star network. Do not use a resistor, as V_{DDA} is used as a reference voltage by the A/D converter and any resistance would cause a voltage drop and a loss of accuracy.

Properly place components and route the signal traces on the PCB to shield the analog inputs. Analog signals paths should run over the analog ground plane and be as short as possible. Isolate analog signals from digital signals that may switch while the analog inputs are being sampled by the A/D converter. Do not toggle digital outputs on the same I/O port as the A/D input being converted.

12.13.0.3 Software Filtering of Spurious Conversion Results

For EMC performance reasons, it is recommended to filter A/D conversion outliers using software filtering techniques.



15 DEVELOPMENT TOOLS

Full details of tools available for the ST7 from third party manufacturers can be obtained from the STMicroelectronics Internet site: → www.st.com/mcu

Tools from iSystem and Hitex include C compliers, emulators and gang programmers.

Note: Before designing the board layout, it is recobsolete Product(s). Obsolete Product(s) ommended to check the overall dimensions of the socket as they may be greater than the dimen-

For footprint and other mechanical information about these sockets and adapters, refer to the manufacturer's datasheet.

ST Programming Tools

- ST7MDT25-EPB: For in-socket or ICC programming
- ST7-STICK: For ICC programming