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	CANbus, I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	48
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	3.8V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st72f521r6tctr

List of tables

6.5.5 Internal watchdog RESET

The RESET sequence generated by an internal Watchdog counter overflow is shown in *Figure 13*.

Starting from the Watchdog counter underflow, the device $\overline{\text{RESET}}$ pin acts as an output that is pulled low during at least $t_{w(RSTL)out}$.

Figure 13. RESET sequences

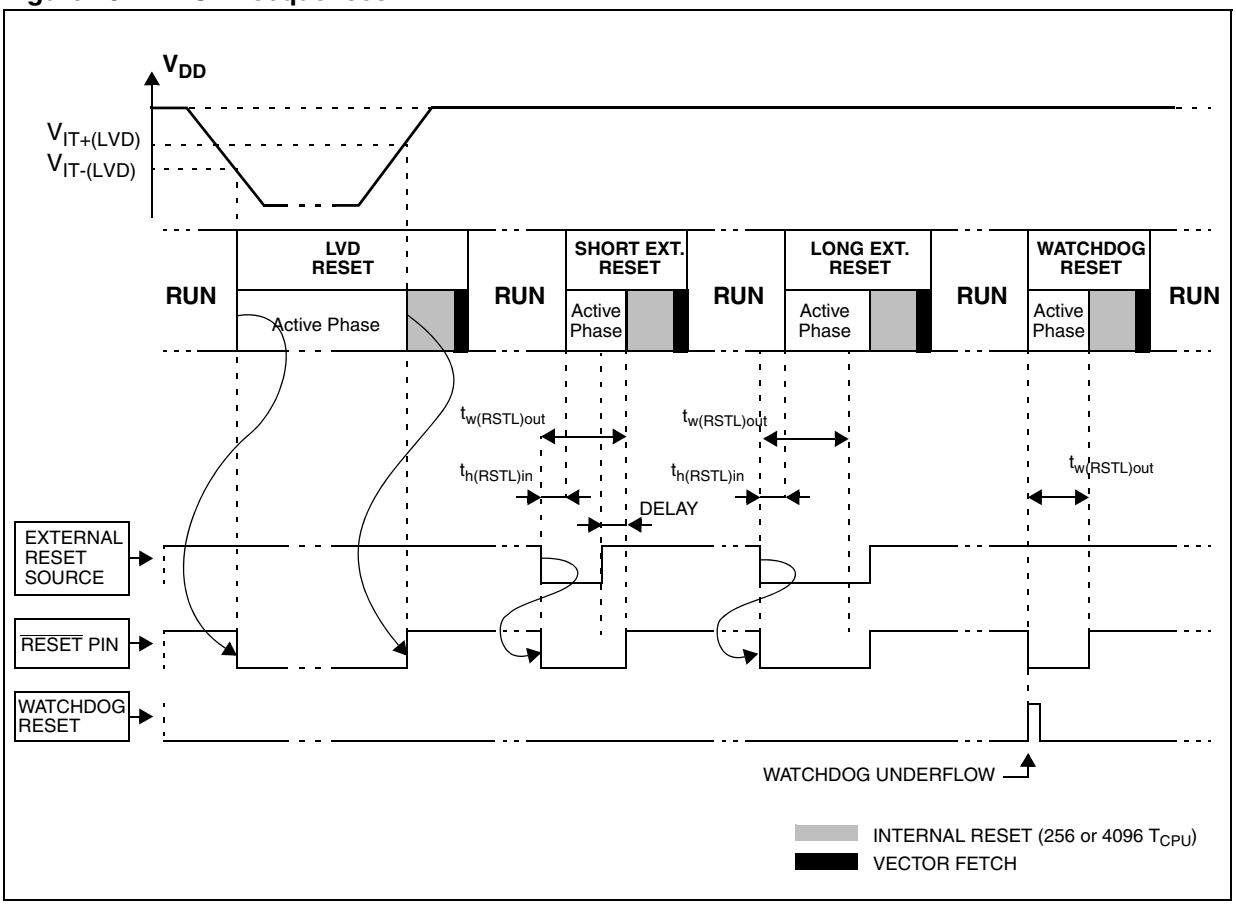
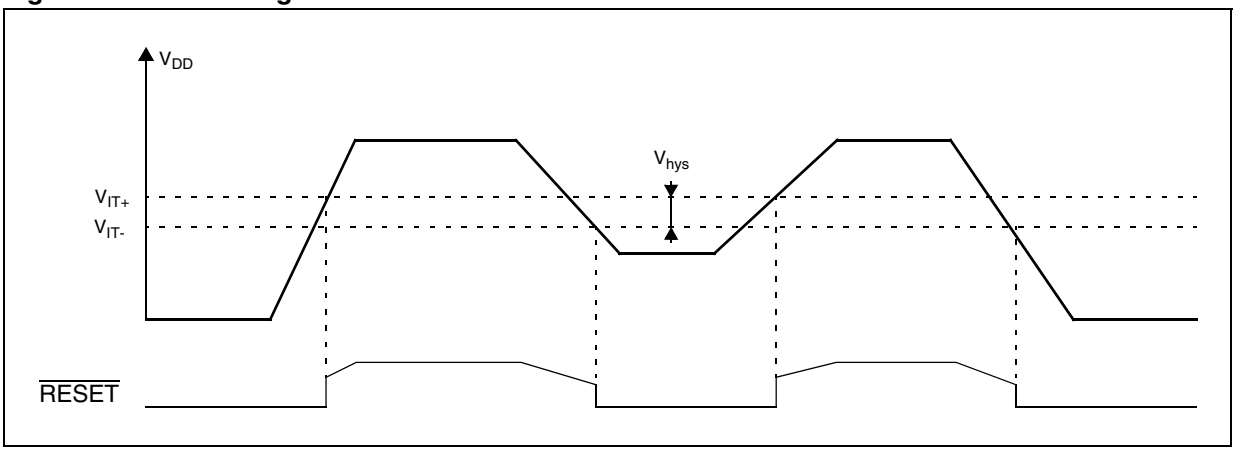


Figure 14. Low voltage detector versus reset



6.6.2 Auxiliary voltage detector (AVD)

The auxiliary voltage detector function (AVD) is based on an analog comparison between a $V_{IT-(AVD)}$ and $V_{IT+(AVD)}$ reference value and the V_{DD} main supply or the external EVD pin voltage level (V_{EVD}). The V_{IT-} reference value for falling voltage is lower than the V_{IT+} reference value for rising voltage in order to avoid parasitic detection (hysteresis).

The output of the AVD comparator can be read directly by the application software through a real-time status bit (AVDF) in the SICSR register. This bit is read only.

Caution: The AVD function is active only if the LVD is enabled through the option byte.

Monitoring the V_{DD} main supply

This mode is selected by clearing the AVDS bit in the SICSR register.

The AVD voltage threshold value is relative to the selected LVD threshold configured by option byte (see *Section 22.1.1: Flash configuration on page 257*).

If the AVD interrupt is enabled, an interrupt is generated when the voltage crosses the $V_{IT+(AVD)}$ or $V_{IT-(AVD)}$ threshold (AVDF bit toggles).

In the case of a drop in voltage, the AVD interrupt acts as an early warning, allowing software to shut down safely before the LVD resets the microcontroller. See *Figure 15*.

The interrupt on the rising edge is used to inform the application that the V_{DD} warning state is over.

If the voltage rise time t_{rv} is less than 256 or 4096 CPU cycles (depending on the reset delay selected by option byte), no AVD interrupt will be generated when $V_{IT+(AVD)}$ is reached.

If t_{rv} is greater than 256 or 4096 cycles

- two AVD interrupts will be received if the AVD interrupt is enabled **before** the $V_{IT+(AVD)}$ threshold is reached: the first when the AVDIE bit is set, and the second when the threshold is reached.
- only one AVD interrupt will occur if the AVD interrupt is enabled **after** the $V_{IT+(AVD)}$ threshold is reached.

Figure 37. PWM auto-reload timer function

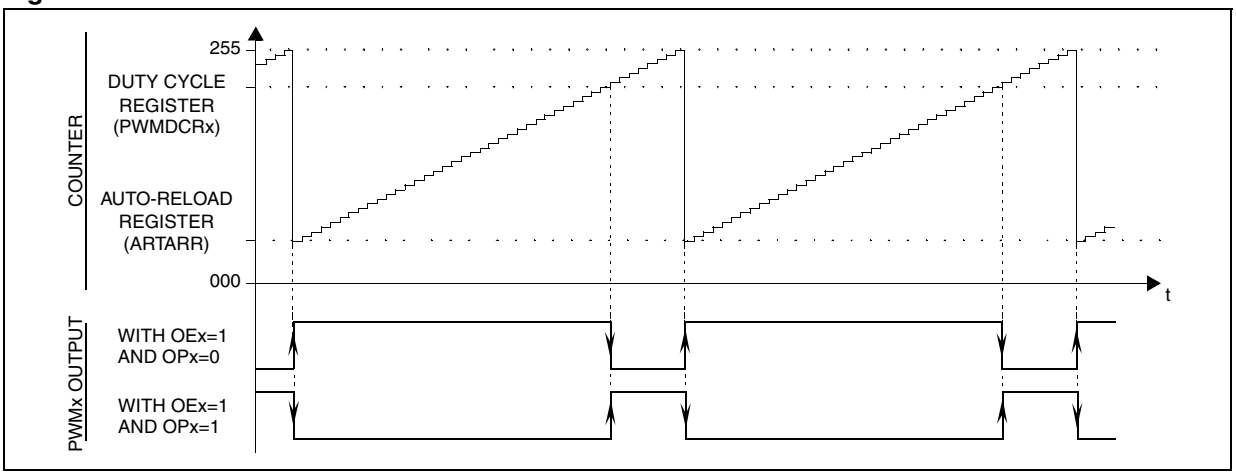
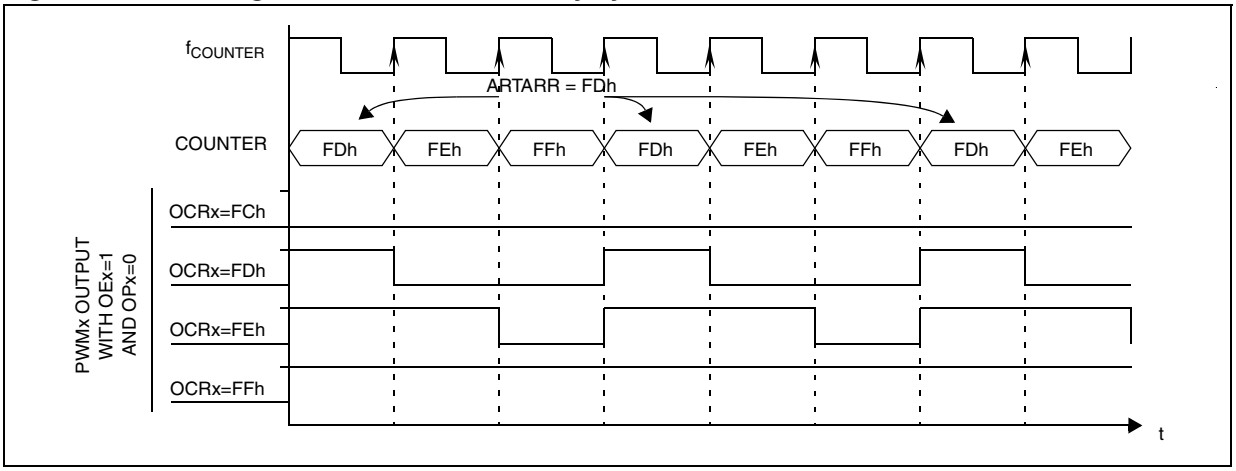


Figure 38. PWM signal from 0% to 100% duty cycle



12.2.6 Output compare and time base interrupt

On overflow, the OVF flag of the ARTCSR register is set and an overflow interrupt request is generated if the overflow interrupt enable bit, OIE, in the ARTCSR register, is set. The OVF flag must be reset by the user software. This interrupt can be used as a time base in the application.

12.2.7 External clock and event detector mode

Using the f_{EXT} external prescaler input clock, the auto-reload timer can be used as an external clock event detector. In this mode, the ARTARR register is used to select the n_{EVENT} number of events to be counted before setting the OVF flag.

$$n_{EVENT} = 256 - ARTARR$$

Caution: The external clock function is not available in Halt mode. If Halt mode is used in the application, prior to executing the HALT instruction, the counter must be disabled by clearing the TCE bit in the ARTCSR register to avoid spurious counter increments.

Table 66. SPICR register description (continued)

Bit	Name	Function
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 67. 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

14.8.2 Control/status register (SPICSR)

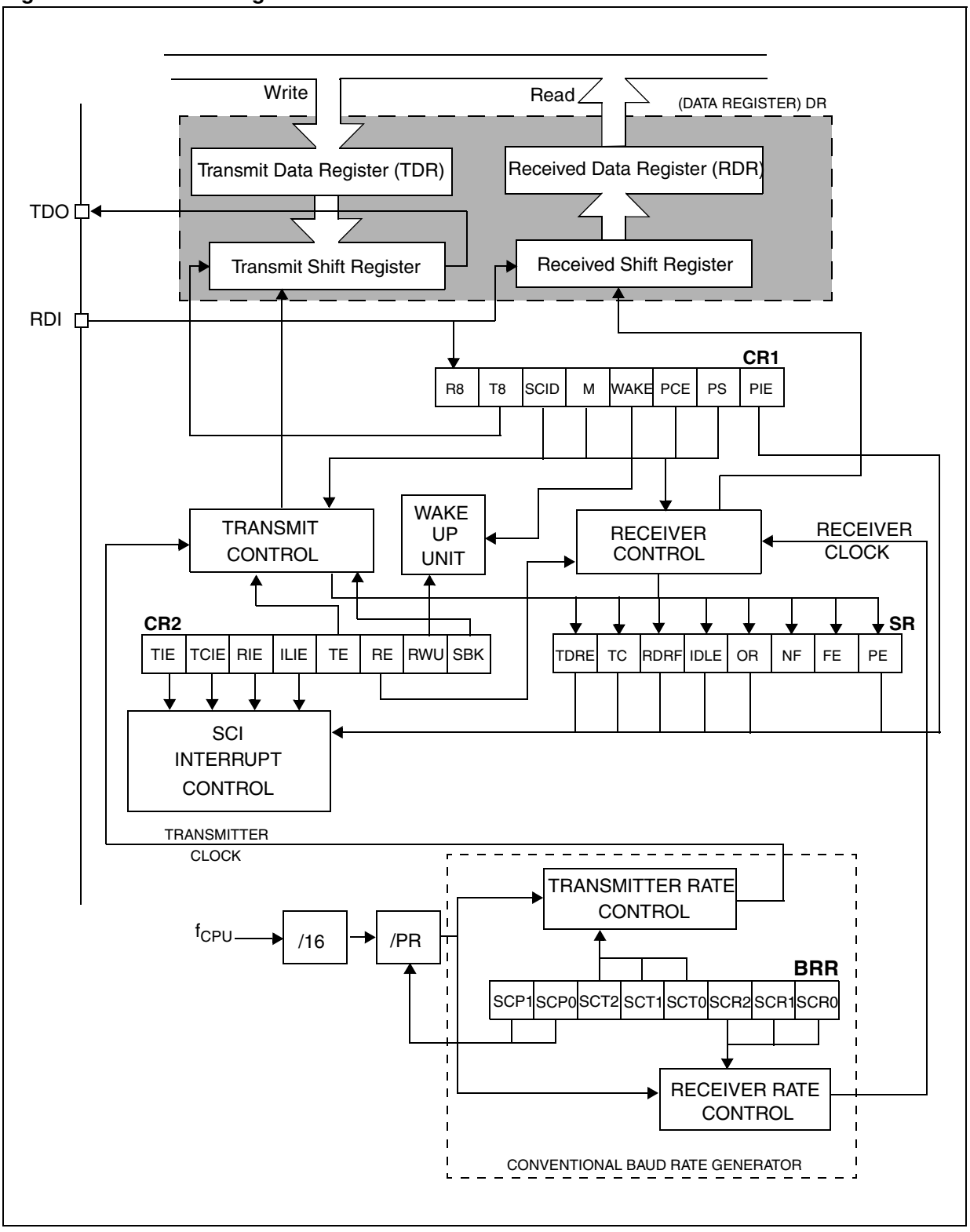
SPICSR Reset value: 0000 0000 (00h)

7	6	5	4	3	2	1	0
SPIF	WCOL	OVR	MODF	Reserved	SOD	SSM	SSI
RO	RO	RO	RO	-	RW	RW	RW

Table 68. SPICSR register description

Bit	Name	Function
7	SPIF	*Serial Peripheral Data Transfer Flag* This bit is set by hardware when a transfer has been completed. An interrupt is generated if SPIE = 1 in the SPICR register. It is cleared by a software sequence (an access to the SPICSR register followed by a write or a read to the SPIDR register). 0: Data transfer is in progress or the flag has been cleared 1: Data transfer between the device and an external device has been completed. While the SPIF bit is set, all writes to the SPIDR register are inhibited until the SPICSR register is read.
6	WCOL	*Write Collision status* This bit is set by hardware when a write to the SPIDR register is done during a transmit sequence. It is cleared by a software sequence (see *Figure 60*). 0: No write collision occurred. 1: A write collision has been detected.
5	OVR	*SPI Overrun error* This bit is set by hardware when the byte currently being received in the shift register is ready to be transferred into the SPIDR register while SPIF = 1 (see *Overrun condition (OVR) on page 131*). An interrupt is generated if SPIE = 1 in SPICR register. The OVR bit is cleared by software reading the SPICSR register. 0: No overrun error 1: Overrun error detected

Figure 62. SCI block diagram



15.7.2 Control register 1 (SCICR1)

SCICR1 Reset value: X000 0000 (x0h)

7	6	5	4	3	2	1	0
R8	T8	SCID	M	WAKE	PCE	PS	PIE
RW	RW	RW	RW	RW	RW	RW	RW

Table 74. SCICR1 register description

Bit	Name	Function
7	R8	*Receive data bit 8* This bit is used to store the 9th bit of the received word when M = 1.
6	T8	*Transmit data bit 8* This bit is used to store the 9th bit of the transmitted word when M = 1.
5	SCID	*Disabled for low power consumption* When this bit is set the SCI prescalers and outputs are stopped and the end of the current byte transfer in order to reduce power consumption.This bit is set and cleared by software. 0: SCI enabled 1: SCI prescaler and outputs disabled
4	M	*Word length* This bit determines the word length. It is set or cleared by software. 0: 1 Start bit, 8 Data bits, 1 Stop bit 1: 1 Start bit, 9 Data bits, 1 Stop bit *Note: The M bit must not be modified during a data transfer (both transmission and reception).*
3	WAKE	*Wake-up method* This bit determines the SCI wake-up method. It is set or cleared by software. 0: Idle line 1: Address mark
2	PCE	*Parity control enable* This bit selects the hardware parity control (generation and detection). When the parity control is enabled, the computed parity is inserted at the MSB position (9th bit if M = 1; 8th bit if M = 0) and parity is checked on the received data. This bit is set and cleared by software. Once it is set, PCE is active after the current byte (in reception and in transmission). 0: Parity control disabled 1: Parity control enabled
1	PS	*Parity selection* This bit selects the odd or even parity when the parity generation/detection is enabled (PCE bit set). It is set and cleared by software. The parity is selected after the current byte. 0: Even parity 1: Odd parity

16 I²C bus interface (I2C)

16.1 Introduction

The I²C bus interface serves as an interface between the microcontroller and the serial I²C bus. It provides both multimaster and slave functions, and controls all I²C bus-specific sequencing, protocol, arbitration and timing. It supports fast I²C mode (400 kHz).

16.2 Main features

- Parallel-bus/I²C protocol converter
- Multimaster capability
- 7-bit/10-bit addressing
- SMBus V1.1 compliant
- Transmitter/Receiver flag
- End-of-byte transmission flag
- Transfer problem detection

16.2.1 I²C master features

- Clock generation
- I²C bus busy flag
- Arbitration Lost flag
- End of byte transmission flag
- Transmitter/Receiver flag
- Start bit detection flag
- Start and Stop generation

16.2.2 I²C slave features

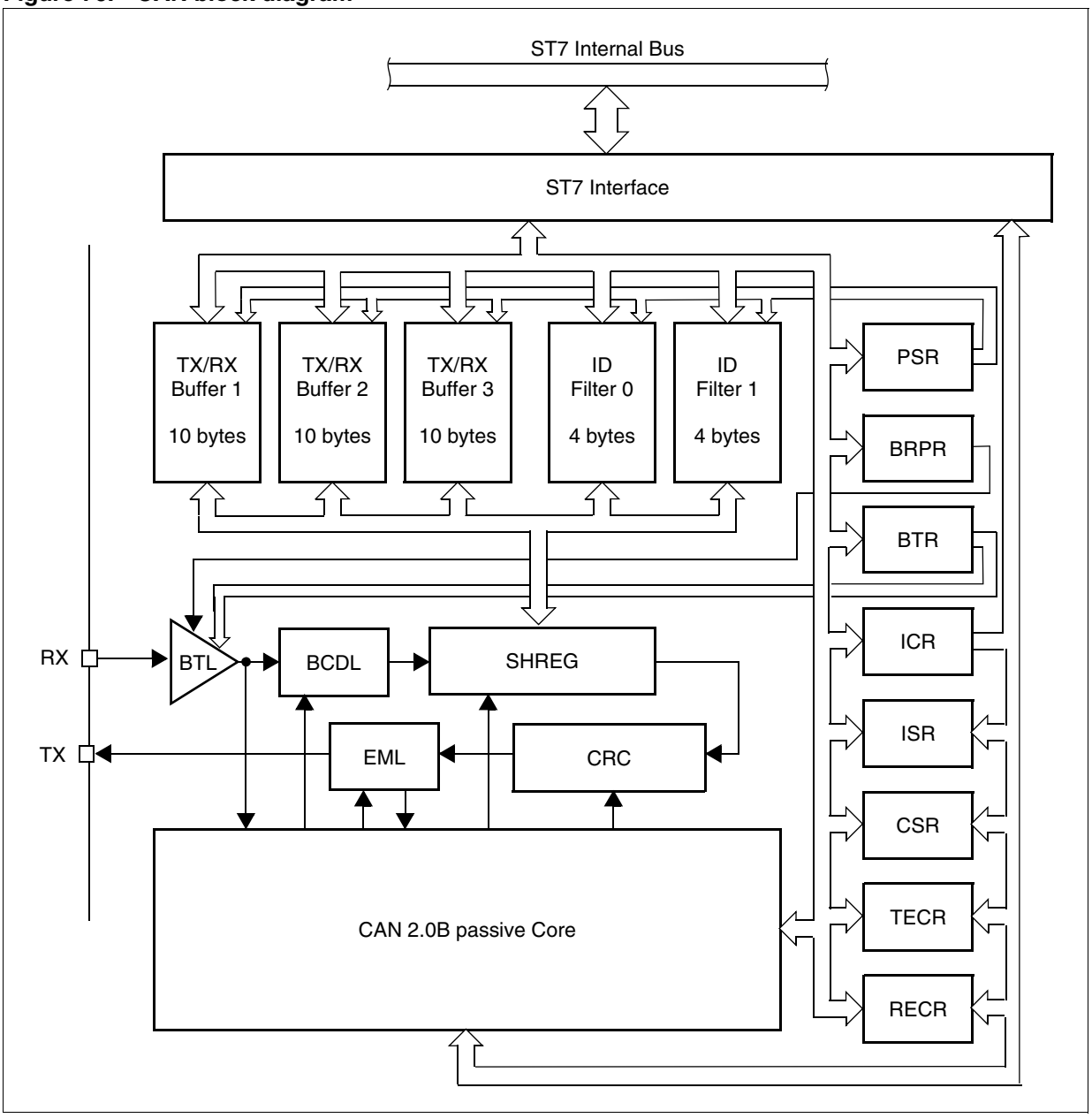
- Stop bit detection
- I²C bus busy flag
- Detection of misplaced start or stop condition
- Programmable I²C address detection
- Transfer problem detection
- End-of-byte transmission flag
- Transmitter/Receiver flag

17 Controller area network (CAN)

17.1 Introduction

This peripheral is designed to support serial data exchanges using a multimaster contention based priority scheme as described in CAN specification Rev. 2.0 part A. It can also be connected to a 2.0 B network without problems, since extended frames are checked for correctness and acknowledged accordingly although such frames cannot be transmitted nor received. The same applies to overload frames which are recognized but never initiated.

Figure 70. CAN block diagram



Filter high registers (FHRx)

FHRx — Reset value: Undefined

7	6	5	4	3	2	1	0
FIL[11:4]							
R/W							

Table 105. FHRx register description

Bit	Name	Function
7:0	FIL[11:4]	*Acceptance Filter* These are the most significant 8 bits of a 12-bit message filter. The acceptance filter is compared bit by bit with the identifier and the RTR bit of the incoming message. If there is a match for the set of bits specified by the acceptance mask then the message is stored in a receive buffer.

Filter low registers (FLRx)

FLRx — Reset value: Undefined

7	6	5	4	3	2	1	0
FIL[3:0]				Reserved			
R/W				-			

Table 106. FLRx register description

Bit	Name	Function
7:4	FIL[3:0]	*Acceptance Filter* These are the least significant 4 bits of a 12-bit message filter.
3:0	-	Reserved; must be kept at ‘0’

Mask high registers (MHRx)

MHRx — Reset value: Undefined

7	6	5	4	3	2	1	0
MSK[11:4]							
R/W							

Table 107. MHRx register description

Bit	Name	Function
7:0	MSK[11:4]	*Acceptance Mask* These are the most significant 8 bits of a 12-bit message mask. The acceptance mask defines which bits of the acceptance filter should match the identifier and the RTR bit of the incoming message. MSK[i] = 0: Don’t care MSK[i] = 1: Match required

Table 112. ADCCSR register description (continued)

Bit	Name	Function
5	ADON	*A/D Converter on* This bit is set and cleared by software. 0: Disable ADC and stop conversion 1: Enable ADC and start conversion
4	-	Reserved. Must be kept cleared
3:0	CH[3:0]	*Channel Selection* These bits are set and cleared by software. They select the analog input to convert. 0000: Channel pin = AIN0 0001: Channel pin = AIN1 0010: Channel pin = AIN2 0011: Channel pin = AIN3 0100: Channel pin = AIN4 0101: Channel pin = AIN5 0110: Channel pin = AIN6 0111: Channel pin = AIN7 1000: Channel pin = AIN8 1001: Channel pin = AIN9 1010: Channel pin = AIN10 1011: Channel pin = AIN11 1100: Channel pin = AIN12 1101: Channel pin = AIN13 1110: Channel pin = AIN14 1111: Channel pin = AIN15 *Note: The number of channels is device dependent. Refer to the device pinout description.*

18.6.2 Data register (ADCDRH)

ADCDRH Reset value: 0000 0000 (00h)

7	6	5	4	3	2	1	0
D[9:2]							
RO							

Table 113. ADCDRH register description

Bit	Name	Function
7:0	D[9:2]	*MSB of Converted Analog Value*

20.4.3 On-chip peripherals

Measured on LQFP64 generic board T_A = 25°C, f_{CPU} = 4 MHz.

Table 133. On-chip peripherals current consumption

Symbol	Parameter	Conditions	Typ	Unit
$I_{DD(TIM)}$	16-bit timer supply current(1)	V_{DD} = 5.0V	50	µA
$I_{DD(ART)}$	ART PWM supply current(2)	V_{DD} = 5.0V	75	µA
$I_{DD(SPI)}$	SPI supply current(3)	V_{DD} = 5.0V	400	µA
$I_{DD(SCI)}$	SCI supply current(4)			
$I_{DD(I2C)}$	I2C supply current(5)	V_{DD} = 5.0V	175	µA
$I_{DD(ADC)}$	ADC supply current when converting(6)	V_{DD} = 5.0V	400	µA
$I_{DD(CAN)}$	CAN supply current(7)	V_{DD} = 5.0V	400	µA

1. Data based on a differential I_{DD} measurement between reset configuration (timer counter running at $f_{CPU}/4$) and timer counter stopped (only TIMD bit set). Data valid for one timer.
2. Data based on a differential I_{DD} measurement between reset configuration (timer stopped) and timer counter enabled (only TCE bit set).
3. Data based on a differential I_{DD} measurement between reset configuration (SPI disabled) and a permanent SPI master communication at maximum speed (data sent equal to 55h). This measurement includes the pad toggling consumption.
4. Data based on a differential I_{DD} measurement between SCI low power state (SCID = 1) and a permanent SCI data transmit sequence.
5. Data based on a differential I_{DD} measurement between reset configuration (I2C disabled) and a permanent I2C master communication at 100 kHz (data sent equal to 55h). This measurement includes the pad toggling consumption (27k ohm external pull-up on clock and data lines).
6. Data based on a differential I_{DD} measurement between reset configuration and continuous A/D conversions.
7. Data based on a differential I_{DD} measurement between reset configuration (CAN disabled) and a permanent CAN data transmit sequence with RX and TX connected together. This measurement include the pad toggling consumption.

20.5 Clock and timing characteristics

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

20.5.1 General timings

Table 134. General timings

Symbol	Parameter	Conditions	Min	Typ(1)	Max	Unit
$t_{c(INST)}$	Instruction cycle time		2	3	12	t_{CPU}
		f_{CPU} = 8 MHz	250	375	1500	ns
$t_{v(IT)}$	Interrupt reaction time(2) $t_{v(IT)} = \Delta t_{c(INST)} + 10$		10		22	t_{CPU}
		f_{CPU} = 8 MHz	1.25		2.75	µs

1. Data based on typical application software.
2. Time measured between interrupt event and interrupt vector fetch. $\Delta t_{c(INST)}$ is the number of t_{CPU} cycles needed to finish the current instruction execution.

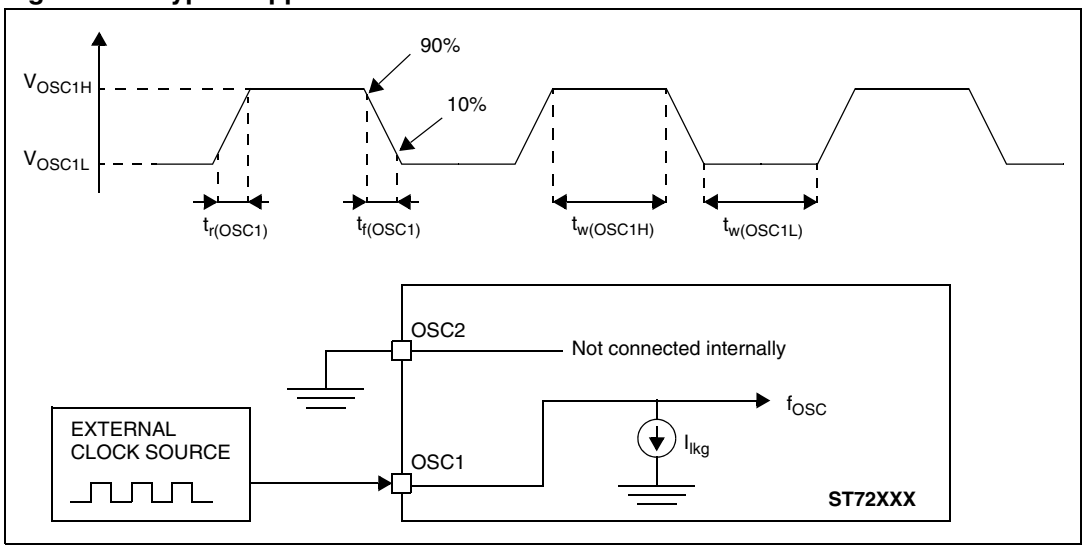
20.5.2 External clock source

Table 135. External clock source

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{OSC1H}	OSC1 input pin high level voltage	See *Figure 93*	0.7xV_{DD}		V_{DD}	V
V_{OSC1L}	OSC1 input pin low level voltage		V_{SS}		0.3xV_{DD}	
$t_{w(OSC1H)}$ $t_{w(OSC1L)}$	OSC1 high or low time(1)		5			ns
$t_{r(OSC1)}$ $t_{f(OSC1)}$	OSC1 rise or fall time(1)				15	
I_{lkg}	OSC1 input leakage current	$V_{SS} \le V_{IN} \le V_{DD}$			±1	µA

1. Data based on design simulation and/or technology characteristics, not tested in production.

Figure 93. Typical application with an external clock source



20.11.2 I²C - inter IC control interface

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

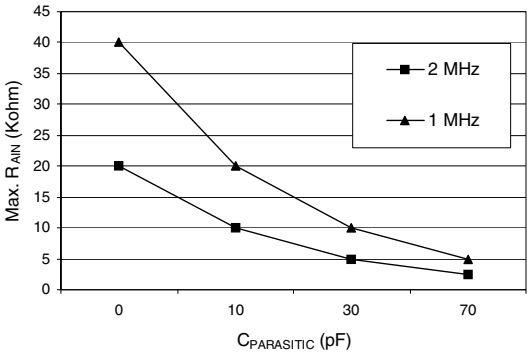
Refer to *Section 20.8: I/O port pin characteristics* for more details on the input/output alternate function characteristics (SDAI and SCLI). The ST7 I2C interface meets the requirements of the standard I2C communication protocol described in the following table.

Table 153. I²C control interface characteristics

Symbol	Parameter	Standard mode I²C		Fast mode I²C(1)		Unit
		Min(2)	Max(2)	Min(2)	Max(2)	
$t_{w(SCLL)}$	SCL clock low time	4.7		1.3		µs
$t_{w(SCLH)}$	SCL clock high time	4.0		0.6		
$t_{su(SDA)}$	SDA setup time	250		100		ns
$t_{h(SDA)}$	SDA data hold time	0(3)		0(4)	900(3)	
$t_{r(SDA)}$ $t_{r(SCL)}$	SDA and SCL rise time		1000	20+0.1C_b	300	
$t_{f(SDA)}$ $t_{f(SCL)}$	SDA and SCL fall time		300			
$t_{h(STA)}$	START condition hold time	4.0		0.6		µs
$t_{su(STA)}$	Repeated START condition setup time	4.7				
$t_{su(STO)}$	STOP condition setup time	4.0				
$t_{w(STO:STA)}$	STOP to START condition time (bus free)	4.7		1.3		
C_b	Capacitive load for each bus line		400		400	pF

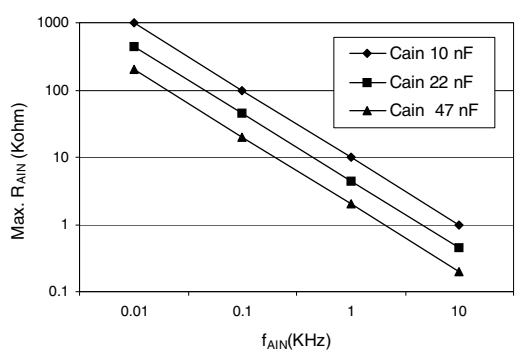
1. At 4 MHz f_{CPU}, maximum I²C speed (400 kHz) is not achievable. In this case, maximum I²C speed will be approximately 260 kHz.
2. Data based on standard I²C protocol requirement, not tested in production.
3. The maximum hold time of the START condition has only to be met if the interface does not stretch the low period of SCL signal.
4. The device must internally provide a hold time of at least 300ns for the SDA signal in order to bridge the undefined region of the falling edge of SCL.

Figure 112. R_{AIN} maximum versus f_{ADC} with C_{AIN} = 0pF(1)



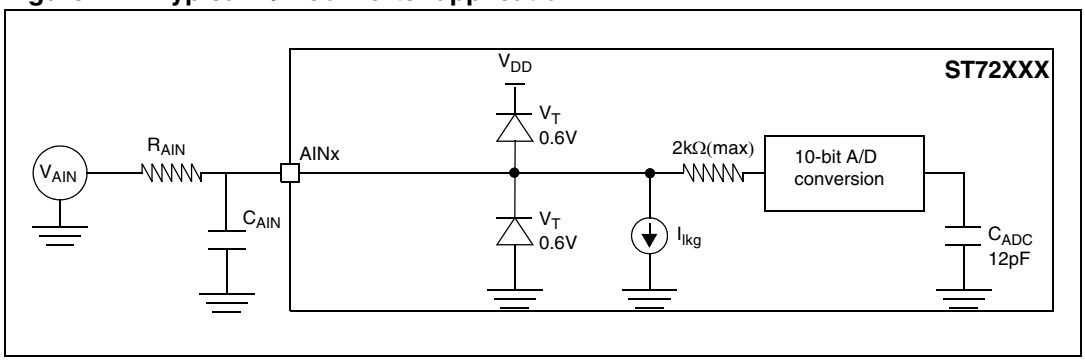
1. $C_{PARASITIC}$ represents the capacitance of the PCB (dependent on soldering and PCB layout quality) plus the pad capacitance (3pF). A high $C_{PARASITIC}$ value will downgrade conversion accuracy. To remedy this, f_{ADC} should be reduced.

Figure 113. Recommended C_{AIN} and R_{AIN} values(1)



1. This graph shows that, depending on the input signal variation (f_{AIN}), C_{AIN} can be increased for stabilization time and decreased to allow the use of a larger serial resistor (R_{AIN}).

Figure 114. Typical A/D converter application



20.12.1 Analog power supply and reference pins

Depending on the MCU pin count, the package may feature separate V_{AREF} 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.

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 20.12.2: General PCB design guidelines*).

24 Revision history

Table 170. Document revision history

Date	Revision	Changes
12-Jul-2010	1	Initial release