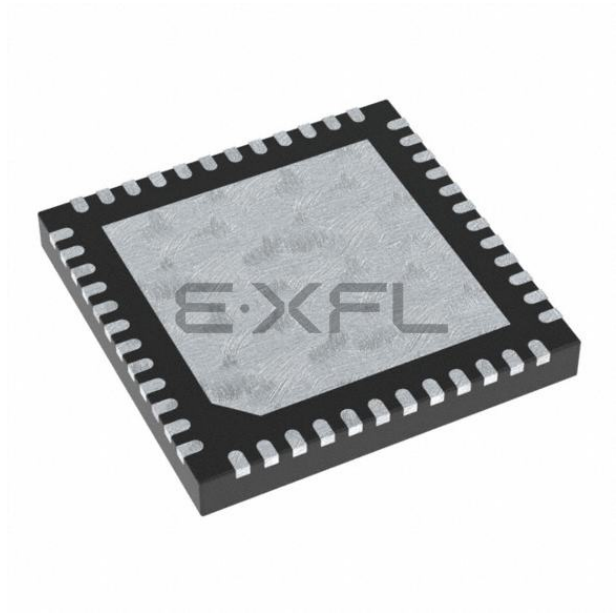


Welcome to E-XFL.COM

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	ARM® Cortex®-M4
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	32
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	1.68V ~ 3.6V
Data Converters	A/D 3x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsam4ls4aa-mu

Figure 3-4. ATSAM4LC TQFP64/QFN64 Pinout

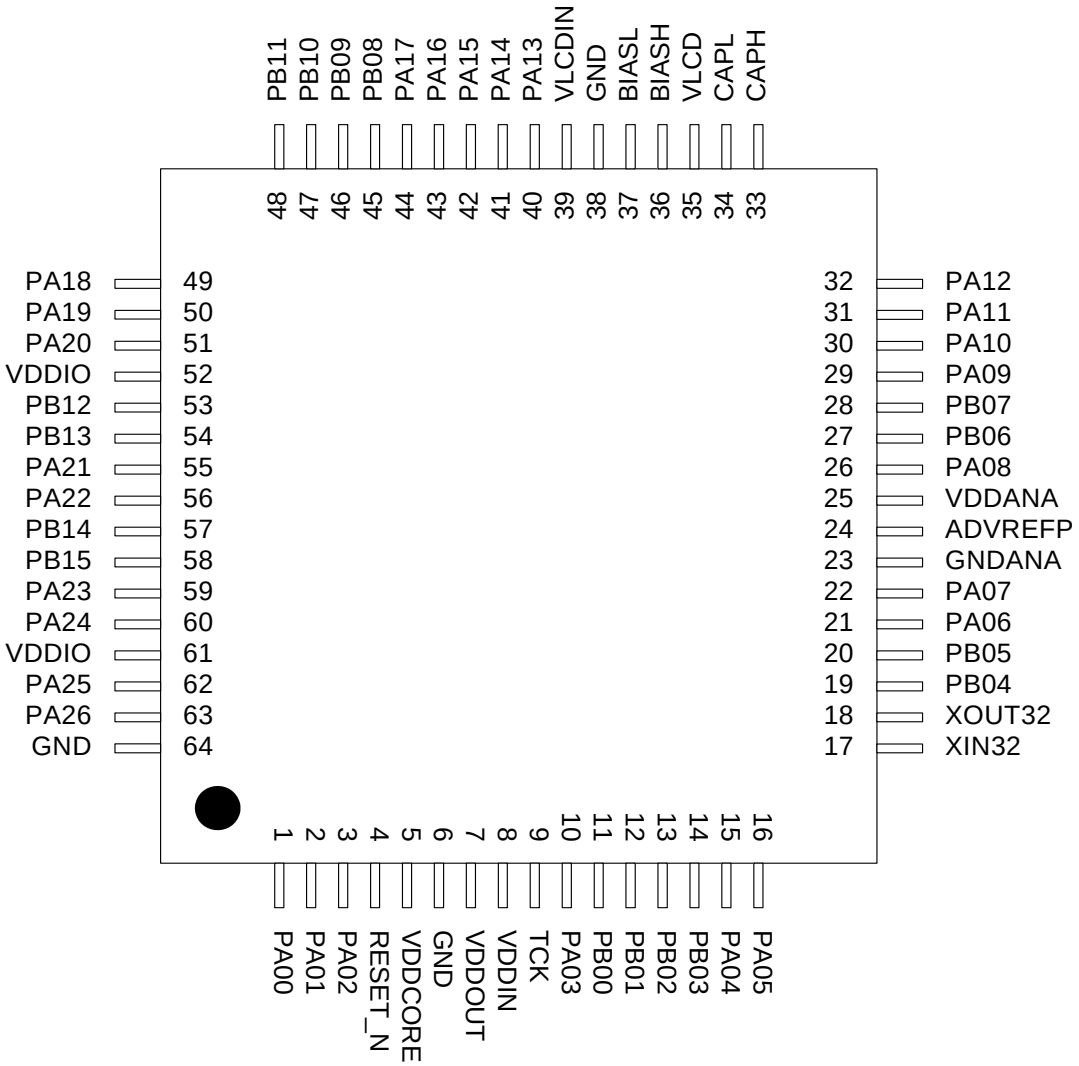


Table 4-2. Interrupt Request Signal Map (Sheet 3 of 3)

Line	Module	Signal
48	External Interrupt Controller	EIC 4
49	External Interrupt Controller	EIC 5
50	External Interrupt Controller	EIC 6
51	External Interrupt Controller	EIC 7
52	External Interrupt Controller	EIC 8
53	Inter-IC Sound (I2S) Controller	IISC
54	Serial Peripheral Interface	SPI
55	Timer/Counter	TC00
56	Timer/Counter	TC01
57	Timer/Counter	TC02
58	Timer/Counter	TC10
59	Timer/Counter	TC11
60	Timer/Counter	TC12
61	Two-wire Master Interface	TWIM0
62	Two-wire Slave Interface	TWIS0
63	Two-wire Master Interface	TWIM1
64	Two-wire Slave Interface	TWIS1
65	Universal Synchronous Asynchronous Receiver Transmitter	USART0
66	Universal Synchronous Asynchronous Receiver Transmitter	USART1
67	Universal Synchronous Asynchronous Receiver Transmitter	USART2
68	Universal Synchronous Asynchronous Receiver Transmitter	USART3
69	ADC controller interface	ADCIFE
70	DAC Controller	DACC
71	Analog Comparator Interface	ACIFC
72	Audio Bitstream DAC	ABDACB
73	True Random Number Generator	TRNG
74	Parallel Capture	PARC
75	Capacitive Touch Module B	CATB
77	Two-wire Master Interface	TWIM2
78	Two-wire Master Interface	TWIM3
79	LCD Controller A	LCDCA

8.7 Enhanced Debug Port (EDP)

Rev.: 1.0.0.0

8.7.1 Features

- **IEEE1149.1 compliant JTAG debug port**
- **Serial Wire Debug Port**
- **Boundary-Scan chain on all digital pins for board-level testing**
- **Debugger Hot-Plugging**
- **SMAP core reset request source**

8.7.2 Overview

The enhanced debug port embeds a standard ARM debug port plus some specific hardware intended for testability and activation of the debug port features. All the information related to the ARM Debug Interface implementation can be found in the ARM Debug Interface v5.1 Architecture Specification document.

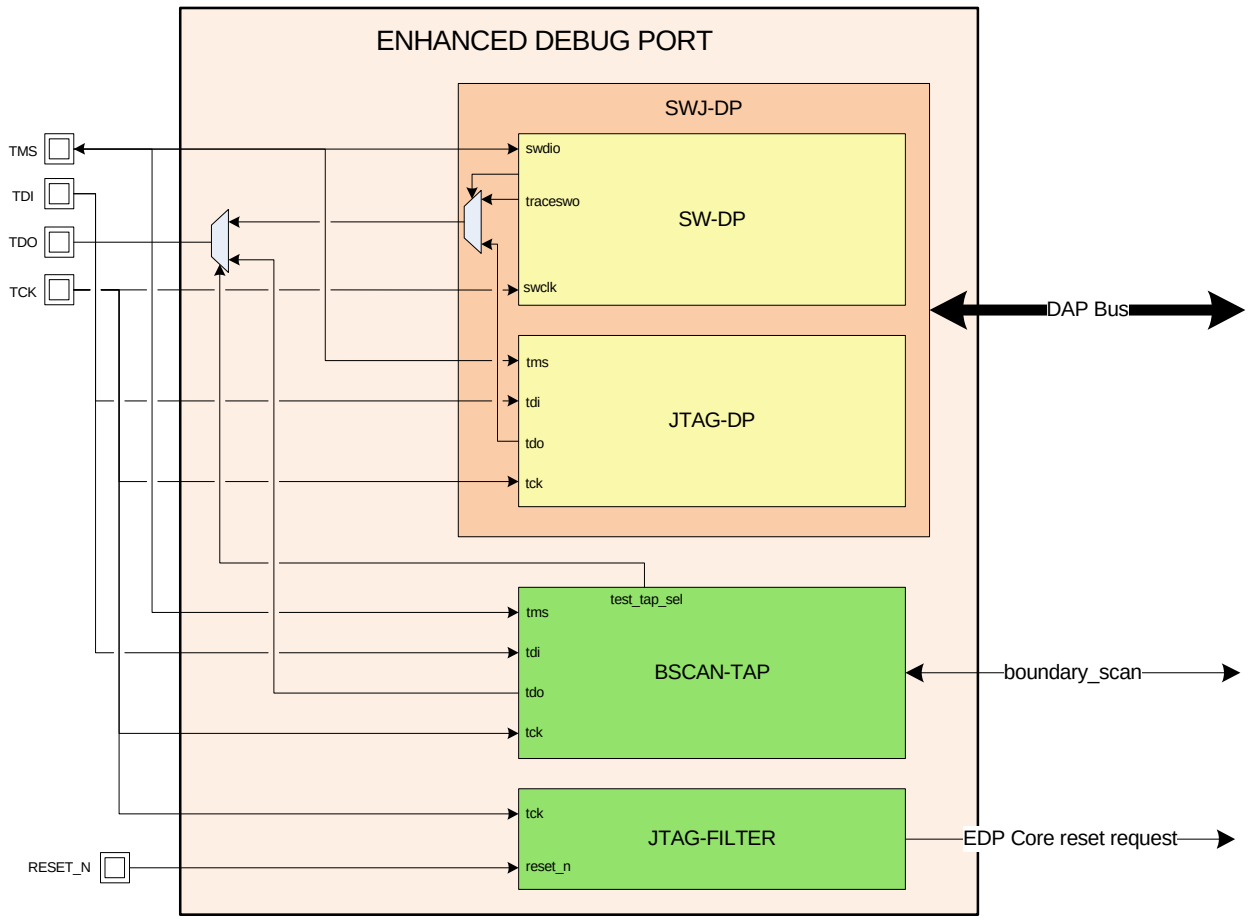
It features:

- A single Debug Port (SWJ-DP), that provides the external physical connection to the interface and supports two DP implementations:
  - the JTAG Debug Port (JTAG-DP)
  - the Serial Wire Debug Port (SW-DP)
- A supplementary JTAG TAP (BSCAN-TAP) connected in parallel with the JTAG-DP that implements the boundary scan instructions detailed in
- A JTAG-FILTER module that monitors TCK and RESET_N pins to handle specific features like the detection of a debugger hot-plugging and the request of reset of the Cortex-M4 at startup.

The JTAG-FILTER module detects the presence of a debugger. When present, JTAG pins are automatically assigned to the Enhanced Debug Port(EDP). If the SWJ-DP is switched to the SW mode, then TDI and TDO alternate functions are released. The JTAG-FILTER also implements a CPU halt mechanism. When triggered, the Cortex-M4 is maintained under reset after the external reset is released to prevent any system corruption during later programmation operations.

8.7.3 Block Diagram

Figure 8-3. Enhanced Debug Port Block Diagram



8.7.4 I/O Lines Description

Table 8-1. I/O Lines Description

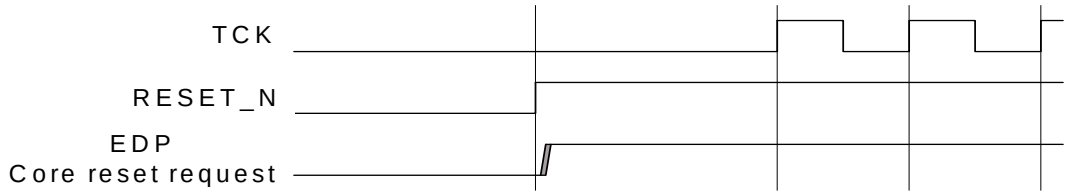
Name	JTAG Debug Port		SWD Debug Port	
	Type	Description	Type	Description
TCK/SWCLK	I	Debug Clock	I	Serial Wire Clock
TDI	I	Debug Data in	-	NA
TDO/TRACESWO	O	Debug Data Out	O	Trace asynchronous Data Out
TMS/SWDIO	I	Debug Mode Select	I/O	Serial Wire Input/Output
RESET_N	I	Reset	I	Reset

The Debug Port pins assignation is then forced to the EDP function even if they were already assigned to another module. This allows to connect a debugger at any time without reseting the device. The connection is non-intrusive meaning that the chip will continue its execution without being disturbed. The CPU can of course be halted later on by issuing Cortex-M4 OCD features.

8.7.8 SMAP Core Reset Request Source

The EDP has the ability to send a request to the SMAP for a Cortex-M4 Core reset. The procedure to do so is to hold TCK low until RESET_N is released. This mechanism aims at halting the CPU to prevent it from changing the system configuration while the SMAP is operating.

Figure 8-5. SMAP Core Reset Request Timings Diagram



The SMAP can de-assert the core reset request for this operation, refer to Section 2.8.8 ”Cortex-M4 Core Reset Source” on page 57.

8.7.9 SWJ-DP

The Cortex-M4 embeds a SWJ-DP Debug port which is the standard CoreSight™ debug port. It combines Serial Wire Debug Port (SW-DP), from 2 to 3 pins and JTAG debug Port(JTAG-DP), 5 pins.

By default, the JTAG Debug Port is active. If the host debugger wants to switch to the Serial Wire Debug Port, it must provide a dedicated JTAG sequence on TMS/SWDIO and TCK/SWCLK which disables JTAG-DP and enables SW-DP.

When the EDP has been switched to Serial Wire mode, TDO/TRACESWO can be used for trace (for more information refer to the section below). The asynchronous TRACE output (TRACESWO) is multiplexed with TDO. So the asynchronous trace can only be used with SW-DP, not JTAG-DP.

The SWJ-DP provides access to the AHB-AP and SMAP access ports which have the following APSEL value:

Figure 8-6. Access Ports APSEL

Acces Port (AP)	APSEL
AHB-AP	0
SMAP	1

Refer to the ARM Debug Interface v5.1 Architecture Specification for more details on SWJ-DP.

0: No bus error has been detected sincle last clear of this bit

- **HCR: Hold Core reset**
  1: The Cortex-M4 core is held under reset
  0: The Cortex-M4 core is not held under reset
- **DONE: Operation done**
  1: At least one operation has terminated since last clear of this field
  0: No operation has terminated since last clear of this field

8.9.11.4 Address Register

Name: ADDR

Access Type: Read/Write

Offset: 0x0C

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
ADDR							

23	22	21	20	19	18	17	16
ADDR							

15	14	13	12	11	10	9	8
ADDR							

7	6	5	4	3	2	1	0
ADDR						-	-

- **ADDR: Address Value**
  Addess values are always world aligned

8.9.11.5 Length Register

Name: LENGTH

Access Type: Read/Write

Offset: 0x10

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
LENGTH							

23	22	21	20	19	18	17	16
LENGTH							

15	14	13	12	11	10	9	8
LENGTH							

7	6	5	4	3	2	1	0
LENGTH						-	-

- **LENGTH: Length Value, Bits 1-0 are always zero**

8.9.11.8 Chip Identification Register

Name: CIDR

Access Type: Read-Only

Offset: 0xF0

Reset Value: -

31	30	29	28	27	26	25	24
EXT	NVPTYP			ARCH			

23	22	21	20	19	18	17	16
ARCH				SRAMSIZ			

15	14	13	12	11	10	9	8
NVPSIZ2				NVPSIZ			

7	6	5	4	3	2	1	0
EPROC				VERSION			

Note: Refer to section CHIPID for more information on this register.

8.11.8 Chip erase typical procedure

The chip erase operation is triggered by writing a one in the CE bit in the Control Register (CR.CE). This clears first all volatile memories in the system and second the whole flash array. Note that the User page is not erased in this process. To ensure that the chip erase operation is completed, check the DONE bit in the Status Register (SR.DONE). Also note that the chip erase operation depends on clocks and power management features that can be altered by the CPU. It is important to ensure that it is stopped. The recommended sequence is shown below:

1. At power up, RESET_N is driven low by a debugger. The on-chip regulator holds the system in a POR state until the input supply is above the POR threshold. The system continues to be held in this static state until the internally regulated supplies have reached a safe operating.
2. PM starts, clocks are switched to the slow clock (Core Clock, System Clock, Flash Clock, and any Bus Clocks that do not have clock gate control). Internal resets are maintained due to the external reset.
   – The debug port and access ports receives a clock and leave the reset state
3. The debugger maintains a low level on TCK and release RESET_N.
   – The SMAP asserts the core_hold_reset signal
4. The Cortex-M4 core remains in reset state, meanwhile the rest of the system is released.
5. The Chip erase operation can be performed by issuing the SMAP Chip Erase command. In this case:
   – volatile memories are cleared first
   – followed by the clearing of the flash array
   – followed by the clearing of the protected state
6. After operation is completed, the chip must be restarted by either controling RESET_N or switching power off/on. Make sure that the TCK pin is high when releasing RESET_N not to halt the core.

8.11.9 Setting the protected state

This is done by issuing a specific flash controller command, for more information, refer to the Flash Controller chapter and to section 8.11.7Flash Programming typical procedure97. The protected state is defined by a highly secure Flash builtin mechanism. Note that for this programmation to propagate, it is required to reset the chip.

9.4 Maximum Clock Frequencies

Table 9-4. Maximum Clock Frequencies in Power Scaling Mode 0/2 and RUN Mode

Symbol	Parameter	Description	Max	Units
f_{CPU}	CPU clock frequency		48	MHz
f_{PBA}	PBA clock frequency		48	
f_{PBB}	PBB clock frequency		48	
f_{PBC}	PBC clock frequency		48	
f_{PBD}	PBD clock frequency		48	
f_{GCLK0}	GCLK0 clock frequency	DFLLIF main reference, GCLK0 pin	50	
f_{GCLK1}	GCLK1 clock frequency	DFLLIF dithering and SSG reference, GCLK1 pin	50	
f_{GCLK2}	GCLK2 clock frequency	AST, GCLK2 pin	20	
f_{GCLK3}	GCLK3 clock frequency	CATB, GCLK3 pin	50	
f_{GCLK4}	GCLK4 clock frequency	FLO and AESA	50	
f_{GCLK5}	GCLK5 clock frequency	GLOC, TC0 and RC32KIFB_REF	80	
f_{GCLK6}	GCLK6 clock frequency	ABDACB and IISC	50	
f_{GCLK7}	GCLK7 clock frequency	USBC	50	
f_{GCLK8}	GCLK8 clock frequency	TC1 and PEVC[0]	50	
f_{GCLK9}	GCLK9 clock frequency	PLL0 and PEVC[1]	50	
f_{GCLK10}	GCLK10 clock frequency	ADCIFE	50	
f_{GCLK11}	GCLK11 clock frequency	Master generic clock. Can be used as source for other generic clocks	150	
f_{OSC0}	OSC0 output frequency	Oscillator 0 in crystal mode	30	
		Oscillator 0 in digital clock mode	50	
f_{PLL}	PLL output frequency	Phase Locked Loop	240	
f_{DFLL}	DFLL output frequency	Digital Frequency Locked Loop	220	
f_{RC80M}	RC80M output frequency	Internal 80MHz RC Oscillator	80	

Table 9-9. ATSAM4L8 Current consumption and Wakeup time for power scaling mode 1

Mode	Conditions	T_A	Typical Wakeup Time	Typ	Max (1)	Unit
RUN	CPU running a Fibonacci algorithm Linear mode	25°C	N/A	222	240	µA/MHz
		85°C		233	276	
	CPU running a CoreMark algorithm Linear mode	25°C	N/A	233	276	
		85°C		230	270	
	CPU running a Fibonacci algorithm Switching mode	25°C	N/A	100	112	
		85°C		100	119	
	CPU running a CoreMark algorithm Switching mode	25°C	N/A	104	128	
		85°C		107	138	
SLEEP0	Switching mode	25°C	9 * Main clock cycles	527	627	µA
		85°C		579	739	
SLEEP1	Switching mode	25°C	9 * Main clock cycles + 500ns	369	445	
		85°C		404	564	
SLEEP2	Switching mode	25°C	9 * Main clock cycles + 500ns	305	381	
		85°C		334	442	
SLEEP3	Linear mode	25°C		46	55	
WAIT	OSC32K and AST running Fast wake-up enable		1.5µs	5.5		
	OSC32K and AST stopped Fast wake-up enable			4.3		
RETENTION	OSC32K running AST running at 1kHz		1.5µs	3.4		
	AST and OSC32K stopped			2.3		
BACKUP	OSC32K running AST running at 1kHz			1.5	3.1	
	AST and OSC32K stopped			0.9	1.7	

1. These values are based on characterization. These values are not covered by test limits in production.

Table 9-10. Typical Power Consumption running CoreMark on CPU clock sources (1)

Clock Source	Conditions	Regulator	Frequency (MHz)	Typ	Unit

Table 9-10. Typical Power Consumption running CoreMark on CPU clock sources [1]

RCSYS (MCSEL = 0)	Power scaling mode 1	Switching Mode	0.115	978	µA/MHz
OSC0 (MCSEL = 1)	Power scaling mode 1		0.5	354	
			12	114	
	Power scaling mode 0		12	228	
			30	219	
OSC0 (MCSEL = 1) External Clock (MODE=0)	Power scaling mode 1		0.6	292	
			12	111	
	Power scaling mode 0		12	193	
	Power scaling mode 2		50	194	
PLL (MCSEL = 2)	Power scaling mode 2 Input Freq = 4MHz from OSC0		40	188	
			50	185	
DFLL (MCSEL = 3)	Power scaling mode 0 Input Freq = 32kHz from OSC32K		20	214	
	Power scaling mode 2 Input Freq = 32kHz from OSC32K		50	195	
RC1M (MCSEL = 4)	Power scaling mode 1		1	267	
RCFAST (MCSEL = 5)	Power scaling mode 1 RCFAST frequency is configurable from 4 to 12MHz		4	153	
			12	114	
RC80M (MCSEL = 6)	Power scaling mode 2 f_{CPU} = RC80M / 2 = 40MHz		40	211	

1. These values are based on characterization. These values are not covered by test limits in production.

9.6.3 USB I/O Pin : PA25, PA26

Table 9-15. USB I/O Pin Characteristics in GPIO configuration (1)

Symbol	Parameter	Conditions		Min	Typ	Max	Units
R_{PULLUP}	Pull-up resistance (2)				40		kΩ
$R_{PULLDOWN}$	Pull-down resistance(2)				40		kΩ
V_{IL}	Input low-level voltage			-0.3		0.2 * V_{VDD}	V
V_{IH}	Input high-level voltage			0.8 * V_{VDD}		V_{VDD} + 0.3	
V_{OL}	Output low-level voltage					0.4	
V_{OH}	Output high-level voltage			V_{VDD} - 0.4			
I_{OL}	Output low-level current (3)	ODCR0=0	1.68V<V_{VDD}<2.7V		20		mA
			2.7V<V_{VDD}<3.6V		30		
I_{OH}	Output high-level current(3)	ODCR0=0	1.68V<V_{VDD}<2.7V		20		mA
			2.7V<V_{VDD}<3.6V		30		
F_{PINMAX}	Maximum frequency(2)	ODCR0=0 OSRR0=0	load = 25pF			20	MHz
I_{LEAK}	Input leakage current(3)	Pull-up resistors disabled			0.01	1	µA
C_{IN}	Input capacitance(2)				5		pF

1. V_{VDD} corresponds to either V_{VDDIN} or V_{VDDIO}, depending on the supply for the pin. Refer to Section 3-5 on page 13 for details
2. These values are based on simulation. These values are not covered by test limits in production or characterization
3. These values are based on characterization. These values are not covered by test limits in production

9.6.4 TWI Pin : PA21, PA22, PA23, PA24, PB14, PB15

Table 9-16. TWI Pin Characteristics in TWI configuration (1)

Symbol	Parameter	Conditions		Min	Typ	Max	Units
R_{PULLUP}	Pull-up resistance (2)				40		kΩ
$R_{PULLDOWN}$	Pull-down resistance(2)				40		kΩ
V_{IL}	Input low-level voltage			-0.3		0.3 * V_{VDD}	V
V_{IH}	Input high-level voltage			0.7 * V_{VDD}		V_{VDD} + 0.3	V
V_{OL}	Output low-level voltage					0.4	V
I_{OL}	Output low-level current (3)	DRIVEL=0				0.5	mA
		DRIVEL=1				1.0	
		DRIVEL=2				1.6	
		DRIVEL=3				3.1	
		DRIVEL=4				6.2	
		DRIVEL=5				9.3	
		DRIVEL=6				15.5	
		DRIVEL=7				21.8	

Table 9-20. High Drive TWI Pin Characteristics in GPIO configuration (1)

Symbol	Parameter	Conditions		Min	Typ	Max	Units
R_{PULLUP}	Pull-up resistance (2)				40		kΩ
$R_{PULLDOWN}$	Pull-up resistance(2)				40		kΩ
V_{IL}	Input low-level voltage			-0.3		0.2 * V_{VDD}	V
V_{IH}	Input high-level voltage			0.8 * V_{VDD}		V_{VDD} + 0.3	
V_{OL}	Output low-level voltage					0.4	
V_{OH}	Output high-level voltage			V_{VDD} - 0.4			
I_{OL}	Output low-level current (3)	ODCR0=0	1.68V<V_{VDD}<2.7V			3.4	mA
			2.7V<V_{VDD}<3.6V			6	
		ODCR0=1	1.68V<V_{VDD}<2.7V			5.2	mA
			2.7V<V_{VDD}<3.6V			8	
I_{OH}	Output high-level current(3)	ODCR0=0	1.68V<V_{VDD}<2.7V			3.4	mA
			2.7V<V_{VDD}<3.6V			6	
		ODCR0=1	1.68V<V_{VDD}<2.7V			5.2	mA
			2.7V<V_{VDD}<3.6V			8	
t_{RISE}	Rise time(2)	OSRR0=0	ODCR0=0 1.68V<V_{VDD}<2.7V, Cload = 25pF		18		ns
		OSRR0=1			110		
		OSRR0=0	ODCR0=0 2.7V<V_{VDD}<3.6V, Cload = 25pF		10		ns
		OSRR0=1			50		
t_{FALL}	Fall time(2)	OSRR0=0	ODCR0=0 1.68V<V_{VDD}<2.7V, Cload = 25pF		19		ns
		OSRR0=1			140		
		OSRR0=0	ODCR0=0 2.7V<V_{VDD}<3.6V, Cload = 25pF		12		ns
		OSRR0=1			63		

1. V_{VDD} corresponds to either V_{VDDIN} or V_{VDDIO}, depending on the supply for the pin. Refer to Section 3-5 on page 13 for details
2. These values are based on simulation. These values are not covered by test limits in production or characterization
3. These values are based on characterization. These values are not covered by test limits in production

Table 9-21. Common High Drive TWI Pin Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LEAK}	Input leakage current (1)	Pull-up resistors disabled		0.01	2	µA
C_{IN}	Input capacitance(1)			10		pF

1. These values are based on simulation. These values are not covered by test limits in production or characterization

9.9.5 Digital to Analog Converter Characteristics

Table 9-49. Operating conditions

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	Analog Supply Voltage (1)	on VDDANA	2.4	3	3.6	V
	Digital Supply Voltage(1)	on VDDCORE	1.62	1.8	1.98	V
	Resolution (2)			10		bits
	Clock frequency(1)	Cload = 50pF ; Rload = 5kΩ			500	kHz
	Load(1)	CLoad			50	pF
		RLoad	5			kΩ
INL	Integral Non Linearity (1)	Best fit-line method			±2	LSBs
DNL	Differential Non Linearity (1)	Best fit-line method	-0.9		+1	LSBs
	Zero Error (offset) (1)	CDR[9:0] = 0		1	5	mV
	Gain Error (1)	CDR[9:0] = 1023		5	10	mV
	Total Harmonic Distortion(1)	80% of VDDANA @ fin = 70kHz	-56		7	dB
	Delay to vout (1)	CDR[9:0] = 512/ Cload = 50 pF / Rload = 5 kΩ	2			µs
	Startup time(1)	CDR[9:0] = 512	5		9	µs
	Output Voltage Range	(ADVREFP < VDDANA – 100mV) is mandatory	0		ADVREFP	V
	ADVREFP Voltage Range(1)	(ADVREFP < VDDANA – 100mV) is mandatory	2.3		3.5	V
	ADVREFN Voltage Range(1)	ADVREFP = GND		0		V
	Standby Current(1)	On VDDANA			500	nA
		On VDDCORE			100	
	DC Current consumption(1)	On VDDANA (no Rload)		485	660	µA
		On ADVREFP (CDR[9:0] = 512)		250	295	

1. These values are based on simulation. These values are not covered by test limits in production or characterization
2. These values are based on characterization. These values are not covered by test limits in production

9.9.6 Analog Comparator Characteristics

Table 9-50. Analog Comparator Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	Positive input voltage range		0.1		VDDIO-0.1	V
	Negative input voltage range		0.1		VDDIO-0.1	
	Offset (1)	V_{ACREFN} =0.1V to VDDIO-0.1V, hysteresis = 0 (2) Fast mode	-12		13	mV
		V_{ACREFN} =0.1V to VDDIO-0.1V, hysteresis = 0(2) Low power mode	-11		12	mV

Figure 9-10. USART in SPI Slave Mode with (CPOL= CPHA= 0) or (CPOL= CPHA= 1)

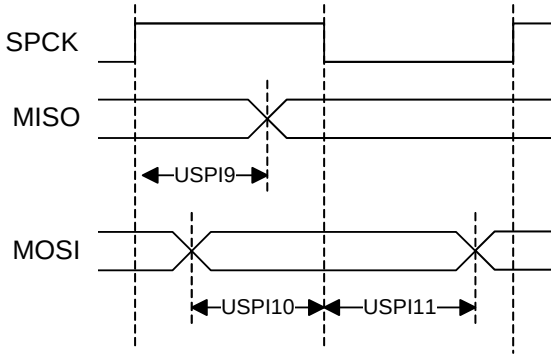


Figure 9-11. USART in SPI Slave Mode, NPCS Timing

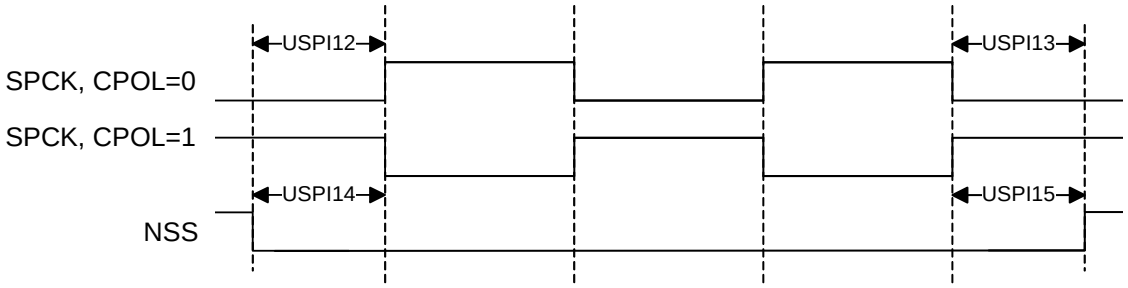


Table 9-58. USART0 in SPI mode Timing, Slave Mode[1]

Symbol	Parameter	Conditions	Min	Max	Units
USPI6	SPCK falling to MISO delay	V_{VDDIO} from 3.0V to 3.6V, maximum external capacitor = 40pF		740.67	ns
USPI7	MOSI setup time before SPCK rises		56.73 + t_{SAMPLE}[2] + t_{CLK_USART}		
USPI8	MOSI hold time after SPCK rises		45.18 -(t_{SAMPLE}[2] + t_{CLK_USART})		
USPI9	SPCK rising to MISO delay			670.18	
USPI10	MOSI setup time before SPCK falls		56.73 +(t_{SAMPLE}[2] + t_{CLK_USART})		
USPI11	MOSI hold time after SPCK falls		45.18 -(t_{SAMPLE}[2] + t_{CLK_USART})		
USPI12	NSS setup time before SPCK rises		688.71		
USPI13	NSS hold time after SPCK falls		-2.25		
USPI14	NSS setup time before SPCK falls		688.71		
USPI15	NSS hold time after SPCK rises		-2.25		

Figure 9-16. SPI Slave Mode, NPCS Timing

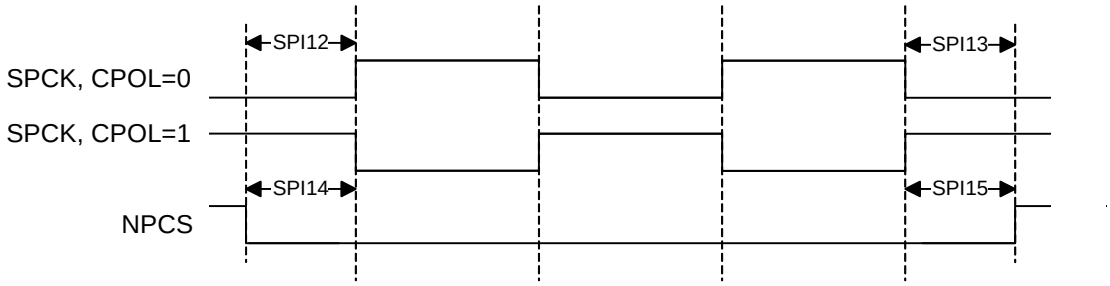


Table 9-63. SPI Timing, Slave Mode[1]

Symbol	Parameter	Conditions	Min	Max	Units
SPI6	SPCK falling to MISO delay	V_{VDDIO} from 2.85V to 3.6V, maximum external capacitor = 40pF	19	47	ns
SPI7	MOSI setup time before SPCK rises		0		
SPI8	MOSI hold time after SPCK rises		5.4		
SPI9	SPCK rising to MISO delay		19	46	
SPI10	MOSI setup time before SPCK falls		0		
SPI11	MOSI hold time after SPCK falls		5.3		
SPI12	NPCS setup time before SPCK rises		4		
SPI13	NPCS hold time after SPCK falls		2.5		
SPI14	NPCS setup time before SPCK falls		6		
SPI15	NPCS hold time after SPCK rises		1.1		

Note: 1. These values are based on simulation. These values are not covered by test limits in production.

Maximum SPI Frequency, Slave Input Mode

The maximum SPI slave input frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(f_{CLKSPI}, \frac{1}{SPIn})$$

Where $SPIn$ is the MOSI setup and hold time, SPI7 + SPI8 or SPI10 + SPI11 depending on CPOL and NCPHA. f_{CLKSPI} is the maximum frequency of the CLK_SPI. Refer to the SPI chapter for a description of this clock.

Maximum SPI Frequency, Slave Output Mode

The maximum SPI slave output frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(f_{PINMAX}, \frac{1}{SPIn + t_{SETUP}})$$

Fix/Workaround
Read the last received data, then perform a software reset by writing a one to the Software Reset bit in the Control Register (CR.SWRST).

Disabling SPI has no effect on the SR.TDRE bit
Disabling SPI has no effect on the SR.TDRE bit whereas the write data command is filtered when SPI is disabled. Writing to TDR when SPI is disabled will not clear SR.TDRE. If SPI is disabled during a PDCA transfer, the PDCA will continue to write data to TDR until its buffer is empty, and this data will be lost.
Fix/Workaround
Disable the PDCA, add two NOPs, and disable the SPI. To continue the transfer, enable the SPI and PDCA.

SPI bad serial clock generation on 2nd chip_select when SCBR=1, CPOL=1, and NCPHA=0
When multiple chip selects (CS) are in use, if one of the baudrates equal 1 while one (CSRn.SCBR=1) of the others do not equal 1, and CSRn.CPOL=1 and CSRn.NCPHA=0, then an additional pulse will be generated on SCK.
Fix/Workaround
When multiple CS are in use, if one of the baudrates equals 1, the others must also equal 1 if CSRn.CPOL=1 and CSRn.NCPHA=0.

12.1.5 TC

Channel chaining skips first pulse for upper channel
When chaining two channels using the Block Mode Register, the first pulse of the clock between the channels is skipped.
Fix/Workaround
Configure the lower channel with RA = 0x1 and RC = 0x2 to produce a dummy clock cycle for the upper channel. After the dummy cycle has been generated, indicated by the SR.CPCS bit, reconfigure the RA and RC registers for the lower channel with the real values.

12.1.6 USBC

In USB host mode, entering suspend mode for low speed device can fail when the USB freeze (USBCON.FRZCLK=1) is done just after UHCON.SOFE=0.
Fix/Workaround
When entering suspend mode (UHCON.SOFE is cleared), check that USBFSM.DRDSTATE is not equal to three before freezing the clock (USBCON.FRZCLK=1).

In USB host mode, the asynchronous attach detection (UDINT.HWUPI) can fail when the USB clock freeze (USBCON.FRZCLK=1) is done just after setting the USB-STA.VBUSRQ bit.
Fix/Workaround
After setting USBSTA.VBUSRQ bit, wait until the USBFSM register value is ‘A_WAIT_BCON’ before setting the USBCON.FRZCLK bit.

12.1.7 FLASHCALW

Corrupted data in flash may happen after flash page write operations.
After a flash page write operation, reading (data read or code fetch) in flash may fail. This may lead to an expecption or to others errors derived from this corrupted read access.
Fix/Workaround
Before any flash page write operation, each 64-bit doublewords write in the page buffer must preceded by a 64-bit doublewords write in the page buffer with 0xFFFFFFFF_FFFFFFFF content at any address in the page. Note that special care is required when loading page buffer, refer to Section 2.5.9 ”Page Buffer Operations” on page 11.