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®-M7
Core Size	32-Bit Single-Core
Speed	300MHz
Connectivity	I²C, IrDA, LINbus, MMC/SD/SDIO, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	75
Program Memory Size	2MB (2M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384K x 8
Voltage - Supply (Vcc/Vdd)	1.08V ~ 3.6V
Data Converters	A/D 10x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TFBGA
Supplier Device Package	100-TFBGA (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70n21b-cn

LQFP Pin	QFN Pin (11)	Power Rail	I/O Type	Primary		Alternate		PIO Peripheral A		PIO Peripheral B		PIO Peripheral CDir		PIO Peripheral DDir		Reset State
				Signal	Dir	Signal	Dir	Signal	Dir	Signal	Dir	Signal	Dir	Signal	Dir	Signal, Dir, PU, PD, HiZ, ST
15	15	VDDIO	CLOCK	PA7	I/O	XIN32(3)	I	–	–	PWMC0_PWMH3	–	–	–	–	–	PIO, HiZ
16	16	VDDIO	CLOCK	PA8	I/O	XOUT32(3)	O	PWMC1_PWMH3	O	AFE0_ADTRG	I	–	–	–	–	PIO, HiZ
33	33	VDDIO	GPIO_AD	PA9	I/O	WKUP6/PIODC3(2)	I	URXD0	I	ISI_D3	I	PWMC0_PWM FI0	I	–	–	PIO, I, PU, ST
28	28	VDDIO	GPIO_AD	PA10	I/O	PIODC4(1)	I	UTXD0	O	PWMC0_PWMEXTRG0	I	RD	I	–	–	PIO, I, PU, ST
27	27	VDDIO	GPIO_AD	PA11	I/O	WKUP7/PIODC5(2)	I	QCS	O	PWMC0_PWMH0	O	PWMC1_PWM L0	O	–	–	PIO, I, PU, ST
29	29	VDDIO	GPIO_AD	PA12	I/O	PIODC6(1)	I	QIO1	I/O	PWMC0_PWMH1	O	PWMC1_PWM H0	O	–	–	PIO, I, PU, ST
18	18	VDDIO	GPIO_AD	PA13	I/O	PIODC7(1)	I	QIO0	I/O	PWMC0_PWMH2	O	PWMC1_PWM L1	O	–	–	PIO, I, PU, ST
19	19	VDDIO	GPIO_CLK	PA14	I/O	WKUP8/PIODCEN1(2)	I	QSCK	O	PWMC0_PWMH3	O	PWMC1_PWM H1	O	–	–	PIO, I, PU, ST
12	12	VDDIO	GPIO_AD	PA21	I/O	AFE0_AD1/PIODCEN2(7)	I	RXD1	I	PCK1	O	PWMC1_PWM FI0	I	–	–	PIO, I, PU, ST
17	17	VDDIO	GPIO_AD	PA22	I/O	PIODCCLK(1)	I	RK	I/O	PWMC0_PWMEXTRG1	I	NCS2	O	–	–	PIO, I, PU, ST
23	23	VDDIO	GPIO_AD	PA24	I/O	–	–	RTS1	O	PWMC0_PWMH1	O	A20	O	ISI_PCK	I	PIO, I, PU, ST
30	30	VDDIO	GPIO_AD	PA27	I/O	–	–	DTR1	O	TIOB2	I/O	MCDA3	I/O	ISI_D7	I	PIO, I, PU, ST
8	8	VDDIO	GPIO	PB0	I/O	AFE0_AD10/RTCOUT0(6)	I	PWMC0_PWMH0	O	–	–	RXD0	I	TF	I/O	PIO, I, PU, ST
7	7	VDDIO	GPIO	PB1	I/O	AFE1_AD0/RTCOUT1(6)	I	PWMC0_PWMH1	O	GTSUCOMP	O	TXD0	I/O	TK	I/O	PIO, I, PU, ST
9	9	VDDIO	GPIO	PB2	I/O	AFE0_AD5(4)	I	CANTX0	O	–	–	CTS0	I	SPI0_NPCS0	I/O	PIO, I, PU, ST
11	11	VDDIO	GPIO_AD	PB3	I/O	AFE0_AD2/WKUP12(6)	I	CANRX0	I	PCK2	O	RTS0	O	ISI_D2	I	PIO, I, PU, ST
46	46	VDDIO	GPIO_MLB	PB4	I/O	TDI(8)	I	TWD1	I/O	PWMC0_PWMH2	O	MLBCLK-	I-	TXD1	I/O	PIO, I, PD, ST
47	47	VDDIO	GPIO_MLB	PB5	I/O	TDO/TRACESWO/WKUP13(8)	O	TWCK1	O	PWMC0_PWML0	O	MLBDAT-	I/O-	TD	O	O, PU
35	35	VDDIO	GPIO	PB6	I/O	SWDIO/TMS(8)	I	–	–	–	–	–	–	–	–	PIO,I,ST
39	39	VDDIO	GPIO	PB7	I/O	SWCLK/TCK(8)	I	–	–	–	–	–	–	–	–	PIO,I,ST
62	63	VDDIO	CLOCK	PB8	I/O	XOUT(9)	O	–	–	–	–	–	–	–	–	PIO, HiZ
63	64	VDDIO	CLOCK	PB9	I/O	XIN(9)	I	–	–	–	–	–	–	–	–	PIO, HiZ
38	38	VDDIO	GPIO	PB12	I/O	ERASE(8)	I	PWMC0_PWML1	O	GTSUCOMP	O	–	–	PCK0	O	PIO, I, PD, ST
1	2	VDDIO	GPIO_AD	PD0	I/O	DAC1(11)	I	GTXCK	I	PWMC1_PWML0	O	SPI1_NPCS1	I/O	DCD0	I	PIO, I, PU, ST
57	57	VDDIO	GPIO	PD1	I/O	–	–	GTXEN	O	PWMC1_PWMH0	O	SPI1_NPCS2	I/O	DTR0	O	PIO, I, PU, ST
56	56	VDDIO	GPIO	PD2	I/O	–	–	GTX0	O	PWMC1_PWML1	O	SPI1_NPCS3	I/O	DSR0	I	PIO, I, PU, ST

Function	Application	Description	Event Source	Event Destination
	Motor control	Puts the PWM outputs in Safe mode (overspeed detection through timer quadrature decoder) (see **Notes 2, 6**)	TC0	PWM0
			TC1	PWM1
	General-purpose, motor control, power factor correction (PFC)	Puts the PWM outputs in Safe mode (general-purpose fault inputs) (see **Note 2**)	PIO PA9, PD8, PD9	PWM0
			PIO PA21, PA26, PA28	PWM1
Security	General-purpose	Immediate GPBR clear (asynchronous) on tamper detection through WKUP0/1 IO pins (see **Note 5**)	PIO WKUP0/1	GPBR
Measurement trigger	Power factor correction (DC-DC, lighting, etc.)	Duty cycle output waveform correction Trigger source selection in PWM (see **Notes 7, 8**)	ACC	PWM0
			PIO PA10, PA22	PWM0
			ACC	PWM1
			PIO PA30, PA18	PWM1
	General-purpose	Trigger source selection in AFEC (see **Note 9**)	PIO AFE0_ADTRG	AFEC0
			TC0 TIOA0	AFEC0
			TC0 TIOA1	AFEC0
			TC0 TIOA2	AFEC0
			ACC	AFEC0
	Motor control	ADC-PWM synchronization (see **Notes 12, 14**) Trigger source selection in AFEC (see **Note 9**)	PWM0 Event Line 0 and 1	AFEC0
	General-purpose	Trigger source selection in AFEC (see **Note 9**)	PIO AFE1_ADTRG	AFEC1
			TC1 TIOA3	AFEC1
			TC1 TIOA4	AFEC1
			TC1 TIOA5	AFEC1
			ACC	AFEC1
	Motor control	ADC-PWM synchronization (see **Notes 12, 14**) Trigger source selection in AFEC (see **Note 9**)	PWM1 Event Line 0 and 1	AFEC1

1. Round-robin Arbitration (default)
2. Fixed Priority Arbitration

Each algorithm may be complemented by selecting a default master configuration for each slave.

When re-arbitration is required, specific conditions apply. Refer to the "Arbitration Rules" section.

19.3.3.1 Arbitration Rules

Each arbiter has the ability to arbitrate between requests from two or more masters. To avoid burst breaking and to provide maximum throughput for slave interfaces, arbitration should take place during the following cycles:

- Idle cycles: When a slave is not connected to any master or is connected to a master which is not currently accessing it
- Single cycles: When a slave is performing a single access
- End of Burst cycles: When the current cycle is the last cycle of a burst transfer. For a defined length burst, predicted end of burst matches the size of the transfer but is managed differently for undefined length burst. Refer to the "Undefined Length Burst Arbitration" section.
- Slot cycle limit: When the slot cycle counter has reached the limit value indicating that the current master access is too long and must be broken. Refer to the "Slot Cycle Limit Arbitration" section.

19.3.3.1.1 Undefined Length Burst Arbitration

In order to prevent slave handling during undefined length bursts, the user can trigger the re-arbitration before the end of the incremental bursts. The re-arbitration period can be selected from the following Undefined Length Burst Type (ULBT) possibilities:

1. Unlimited: no predetermined end of burst is generated. This value enables 1-Kbyte burst lengths.
2. 1-beat bursts: predetermined end of burst is generated at each single transfer during the INCR transfer.
3. 4-beat bursts: predetermined end of burst is generated at the end of each 4-beat boundary during INCR transfer.
4. 8-beat bursts: predetermined end of burst is generated at the end of each 8-beat boundary during INCR transfer.
5. 16-beat bursts: predetermined end of burst is generated at the end of each 16-beat boundary during INCR transfer.
6. 32-beat bursts: predetermined end of burst is generated at the end of each 32-beat boundary during INCR transfer.
7. 64-beat bursts: predetermined end of burst is generated at the end of each 64-beat boundary during INCR transfer.
8. 128-beat bursts: predetermined end of burst is generated at the end of each 128-beat boundary during INCR transfer.

The use of undefined length16-beat bursts, or less, is discouraged since this decreases the overall bus bandwidth due to arbitration and slave latencies at each first access of a burst.

If the master does not permanently and continuously request the same slave or has an intrinsically limited average throughput, the ULBT should be left at its default unlimited value, knowing that the AHB specification natively limits all word bursts to 256 beats and double-word bursts to 128 beats because of its 1-Kbyte address boundaries.

Unless duly needed, the ULBT should be left at its default value of 0 for power saving.

This selection is made through the ULBT field of the Master Configuration Registers (MATRIX_MCFG).

22.4.3.8 Unique Identifier Area

Each device is programmed with a 128-bit unique identifier area .

See “Flash Memory Areas”.

The sequence to read the unique identifier area is the following:

1. Execute the ‘Start Read Unique Identifier’ command by writing EEFC_FCR.FCMD with the STUI command. Field EEFC_FCR.FARG is meaningless.
2. Wait until the bit EEFC_FSR.FRDY falls to read the unique identifier area. The unique identifier field is located in the first 128 bits of the Flash memory mapping. The ‘Start Read Unique Identifier’ command reuses some addresses of the memory plane for code, but the unique identifier area is physically different from the memory plane for code.
3. To stop reading the unique identifier area, execute the ‘Stop Read Unique Identifier’ command by writing EEFC_FCR.FCMD with the SPUI command. Field EEFC_FCR.FARG is meaningless.
4. When the SPUI command has been executed, the bit EEFC_FSR.FRDY rises. If an interrupt was enabled by setting the bit EEFC_FMR.FRDY, the interrupt line of the interrupt controller is activated.

Note: During the sequence, the software cannot be fetched from the Flash.

22.4.3.9 User Signature Area

Each product contains a user signature area of 512 bytes. It can be used for storage. Read, write and erase of this area is allowed.

See “Flash Memory Areas”.

The sequence to read the user signature area is the following:

1. Execute the ‘Start Read User Signature’ command by writing EEFC_FCR.FCMD with the STUS command. Field EEFC_FCR.FARG is meaningless.
2. Wait until the bit EEFC_FSR.FRDY falls to read the user signature area. The user signature area is located in the first 512 bytes of the Flash memory mapping. The ‘Start Read User Signature’ command reuses some addresses of the memory plane but the user signature area is physically different from the memory plane
3. To stop reading the user signature area, execute the ‘Stop Read User Signature’ command by writing EEFC_FCR.FCMD with the SPUS command. Field EEFC_FCR.FARG is meaningless.
4. When the SPUI command has been executed, the bit EEFC_FSR.FRDY rises. If an interrupt was enabled by setting the bit EEFC_FMR.FRDY, the interrupt line of the interrupt controller is activated.

Note: During the sequence, the software cannot be fetched from the Flash or from the second plane in case of dual plane.

One error can be detected in EEFC_FSR after this sequence:

- Command Error: A bad keyword has been written in EEFC_FCR.

The sequence to write the user signature area is the following:

1. Write the full page, at any page address, within the internal memory area address space.
2. Execute the ‘Write User Signature’ command by writing EEFC_FCR.FCMD with the WUS command. Field EEFC_FCR.FARG is meaningless.

32.6.1.31 PIO Pad Pull-Down Enable Register

Name: PIO_PPDER
Offset: 0x0094
Property: Write-only

This register can only be written if the WPEN bit is cleared in the PIO Write Protection Mode Register.

Bit	31	30	29	28	27	26	25	24
	P31	P30	P29	P28	P27	P26	P25	P24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	P23	P22	P21	P20	P19	P18	P17	P16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	P15	P14	P13	P12	P11	P10	P9	P8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
	P7	P6	P5	P4	P3	P2	P1	P0
Access								
Reset								

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 – P PIO Pull-Down Enable

Value	Description
0	No effect.
1	Enables the pull-down resistor on the I/O line.

Bits 1:0 – NC[1:0] Number of Column Bits
Reset value is 8 column bits.

Value	Name	Description
0	COL8	8 bits to define the column number, up to 256 columns.
1	COL9	9 bits to define the column number, up to 512 columns.
2	COL10	10 bits to define the column number, up to 1024 columns.
3	COL11	11 bits to define the column number, up to 2048 columns.

35.16.1.1 SMC Setup Register

Name: SMC_SETUP[0..3]
Offset: 0x00
Reset: 0
Property: R/W

This register can only be written if the WPEN bit is cleared in the “SMC Write Protection Mode Register” .

Bit	31	30	29	28	27	26	25	24
			NCS_RD_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			NRD_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			NCS_WR_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			NWE_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

Bits 29:24 – NCS_RD_SETUP[5:0] NCS Setup Length in READ Access
In read access, the NCS signal setup length is defined as:

NCS setup length = (128* NCS_RD_SETUP[5] + NCS_RD_SETUP[4:0]) clock cycles

Bits 21:16 – NRD_SETUP[5:0] NRD Setup Length
The NRD signal setup length is defined in clock cycles as:

NRD setup length = (128* NRD_SETUP[5] + NRD_SETUP[4:0]) clock cycles

Bits 13:8 – NCS_WR_SETUP[5:0] NCS Setup Length in WRITE Access
In write access, the NCS signal setup length is defined as:

NCS setup length = (128* NCS_WR_SETUP[5] + NCS_WR_SETUP[4:0]) clock cycles

Bits 5:0 – NWE_SETUP[5:0] NWE Setup Length
The NWE signal setup length is defined as:

NWE setup length = (128* NWE_SETUP[5] + NWE_SETUP[4:0]) clock cycles

36.9.3 XDMAC Global Weighted Arbiter Configuration Register

Name: XDMAC_GWAC
Offset: 0x08
Reset: 0x00000000
Property: Read/Write

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	PW3[3:0]				PW2[3:0]			
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	PW1[3:0]				PW0[3:0]			
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 15:12 – PW3[3:0] Pool Weight 3
This field indicates the weight of pool 3 in the arbitration scheme of the DMA scheduler.

Bits 11:8 – PW2[3:0] Pool Weight 2
This field indicates the weight of pool 2 in the arbitration scheme of the DMA scheduler.

Bits 7:4 – PW1[3:0] Pool Weight 1
This field indicates the weight of pool 1 in the arbitration scheme of the DMA scheduler.

Bits 3:0 – PW0[3:0] Pool Weight 0
This field indicates the weight of pool 0 in the arbitration scheme of the DMA scheduler.

37.6.16 DMA Channel Disable Register

Name: ISI_DMA_CHDR
Offset: 0x3C
Reset: –
Property: Write-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
							C_CH_DIS	P_CH_DIS
Access							W	W
Reset							–	–

Bit 1 – C_CH_DIS Codec Channel Disable Request

Value	Description
0	No effect.
1	Disables the channel. Poll C_CH_S in DMA_CHSR to verify that the codec channel status has been successfully modified.

Bit 0 – P_CH_DIS Preview Channel Disable Request

Value	Description
0	No effect.
1	Disables the channel. Poll P_CH_S in DMA_CHSR to verify that the preview channel status has been successfully modified.

Offset	Name	Bit Pos.								
0x0544	USBHS_HSTPIPISR5 (INTPIPES)	7:0	SHORTPACKETI	RXSTALLDI	OVERFI	NAKEDI	PERRI	UNDERFI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x0544	USBHS_HSTPIPISR5 (ISOPIPES)	7:0	SHORTPACKETI	CRCERRI	OVERFI	NAKEDI	PERRI	UNDERFI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x0548	USBHS_HSTPIPISR6	7:0	SHORTPACKETI	RXSTALLDI	OVERFI	NAKEDI	PERRI	TXSTPI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x0548	USBHS_HSTPIPISR6 (INTPIPES)	7:0	SHORTPACKETI	RXSTALLDI	OVERFI	NAKEDI	PERRI	UNDERFI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x0548	USBHS_HSTPIPISR6 (ISOPIPES)	7:0	SHORTPACKETI	CRCERRI	OVERFI	NAKEDI	PERRI	UNDERFI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x054C	USBHS_HSTPIPISR7	7:0	SHORTPACKETI	RXSTALLDI	OVERFI	NAKEDI	PERRI	TXSTPI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x054C	USBHS_HSTPIPISR7 (INTPIPES)	7:0	SHORTPACKETI	RXSTALLDI	OVERFI	NAKEDI	PERRI	UNDERFI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x054C	USBHS_HSTPIPISR7 (ISOPIPES)	7:0	SHORTPACKETI	CRCERRI	OVERFI	NAKEDI	PERRI	UNDERFI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						
0x0550	USBHS_HSTPIPISR8	7:0	SHORTPACKETI	RXSTALLDI	OVERFI	NAKEDI	PERRI	TXSTPI	TXOUTI	RXINI
		15:8	CURRBK[1:0]		NBUSYBK[1:0]				DTSEQ[1:0]	
		23:16	PBYCT[3:0]					CFGOK		RWALL
		31:24		PBYCT[10:4]						

Value	Description
0	Cleared when USBHS_HSTPIPIDR.OVERFIEC = 1. This disables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.OVERFIE).
1	Set when USBHS_HSTPIPIER.OVERFIES = 1. This enables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.OVERFIE).

Bit 4 – NAKEDE NAKed Interrupt Enable

Value	Description
0	Cleared when USBHS_HSTPIPIDR.NAKEDEC = 1. This disables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.NAKEDE).
1	Set when USBHS_HSTPIPIER.NAKEDES = 1. This enables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.NAKEDE).

Bit 3 – PERRE Pipe Error Interrupt Enable

Value	Description
0	Cleared when USBHS_HSTPIPIDR.PERREC = 1. This disables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.PERRE).
1	Set when USBHS_HSTPIPIER.PERRES = 1. This enables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.PERRE).

Bit 2 – TXSTPE Transmitted SETUP Interrupt Enable

Value	Description
0	Cleared when USBHS_HSTPIPIDR.TXSTPEC = 1. This disables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.TXSTPE).
1	Set when USBHS_HSTPIPIER.TXSTPES = 1. This enables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.TXSTPE).

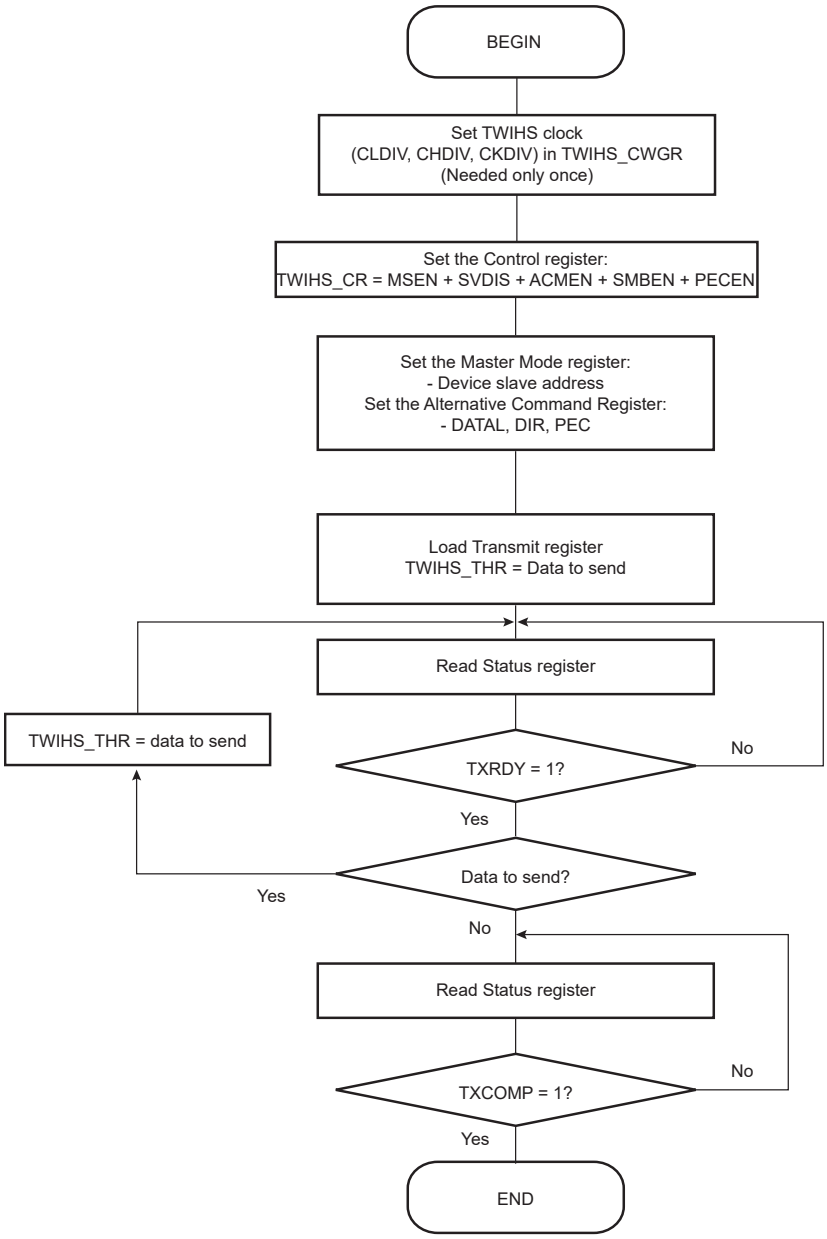
Bit 1 – TXOUTE Transmitted OUT Data Interrupt Enable

Value	Description
0	Cleared when USBHS_HSTPIPIDR.TXOUTEC = 1. This disables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.TXOUTE).
1	Set when USBHS_HSTPIPIER.TXOUTES = 1. This enables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.TXOUTE).

Bit 0 – RXINE Received IN Data Interrupt Enable

Value	Description
0	Cleared when USBHS_HSTPIPIDR.RXINEC = 1. This disables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.RXINE).
1	Set when USBHS_HSTPIPIER.RXINES = 1. This enables the Transmitted IN Data interrupt (USBHS_HSTPIPIMR.RXINE).

Figure 43-20. TWIHS Write Operation with Multiple Data Bytes + Read Operation and Alternative Command Mode + PEC



Bits 11:8 – START[3:0] Receive Start Selection

Value	Name	Description
0	CONTINUOUS	Continuous, as soon as the receiver is enabled, and immediately after the end of transfer of the previous data.
1	TRANSMIT	Transmit start
2	RF_LOW	Detection of a low level on RF signal
3	RF_HIGH	Detection of a high level on RF signal
4	RF_FALLING	Detection of a falling edge on RF signal
5	RF_RISING	Detection of a rising edge on RF signal
6	RF_LEVEL	Detection of any level change on RF signal
7	RF_EDGE	Detection of any edge on RF signal
8	CMP_0	Compare 0

Bits 7:6 – CKG[1:0] Receive Clock Gating Selection

Value	Name	Description
0	CONTINUOUS	None
1	EN_RF_LOW	Receive Clock enabled only if RF Low
2	EN_RF_HIGH	Receive Clock enabled only if RF High

Bit 5 – CKI Receive Clock Inversion
CKI affects only the Receive Clock and not the output clock signal.

Value	Description
0	The data inputs (Data and Frame Sync signals) are sampled on Receive Clock falling edge. The Frame Sync signal output is shifted out on Receive Clock rising edge.
1	The data inputs (Data and Frame Sync signals) are sampled on Receive Clock rising edge. The Frame Sync signal output is shifted out on Receive Clock falling edge.

Bits 4:2 – CKO[2:0] Receive Clock Output Mode Selection

Value	Name	Description
0	NONE	None, RK pin is an input
1	CONTINUOUS	Continuous Receive Clock, RK pin is an output
2	TRANSFER	Receive Clock only during data transfers, RK pin is an output

Bits 1:0 – CKS[1:0] Receive Clock Selection

Value	Name	Description
0	MCK	Divided Clock
1	TK	TK Clock signal
2	RK	RK pin

45.5.2 Power Management

If the CPU enters a Sleep mode that disables clocks used by the I2SC, the I2SC stops functioning and resumes operation after the system wakes up from Sleep mode.

45.5.3 Clocks

The clock for the I2SC bus interface is generated by the Power Management Controller (PMC). I2SC must be disabled before disabling the clock to avoid freezing the I2SC in an undefined state.

45.5.4 DMA Controller

The I2SC interfaces to the DMA Controller. Using the I2SC DMA functionality requires the DMA Controller to be programmed first.

45.5.5 Interrupt Sources

The I2SC interrupt line is connected to the Interrupt Controller. Using the I2SC interrupt requires the Interrupt Controller to be programmed first.

45.6 Functional Description

45.6.1 Initialization

The I2SC features a receiver, a transmitter and a clock generator for Master and Controller modes. Receiver and transmitter share the same serial clock and word select.

Before enabling the I2SC, the selected configuration must be written to the I2SC Mode Register (I2SC_MR) and to the Peripheral Clock Configuration Register (CCFG_PCCR) described in the section “Bus Matrix (MATRIX)”.

If the I2SC_MR.IMCKMODE bit is set, the I2SC_MR.IMCKFS field must be configured as described in section “Serial Clock and Word Select Generation”.

Once the I2SC_MR has been written, the I2SC clock generator, receiver, and transmitter can be enabled by writing a ’1’ to the CKEN, RXEN, and TXEN bits in the Control Register (I2SC_CR). The clock generator can be enabled alone in Controller mode to output clocks to the I2SC_MCK, I2SC_CK, and I2SC_WS pins. The clock generator must also be enabled if the receiver or the transmitter is enabled.

The clock generator, receiver, and transmitter can be disabled independently by writing a ’1’ to I2SC_CR.CXDIS, I2SC_CR.RXDIS and/or I2SC_CR.TXDIS, respectively. Once requested to stop, they stop only when the transmission of the pending frame transmission is completed.

45.6.2 Basic Operation

The receiver can be operated by reading the Receiver Holding Register (I2SC_RHR), whenever the Receive Ready (RXRDY) bit in the Status Register (I2SC_SR) is set. Successive values read from RHR correspond to the samples from the left and right audio channels for the successive frames.

The transmitter can be operated by writing to the Transmitter Holding Register (I2SC_THR), whenever the Transmit Ready (TXRDY) bit in the I2SC_SR is set. Successive values written to THR correspond to the samples from the left and right audio channels for the successive frames.

The RXRDY and TXRDY bits can be polled by reading the I2SC_SR.

The I2SC processor load can be reduced by enabling interrupt-driven operation. The RXRDY and/or TXRDY interrupt requests can be enabled by writing a ’1’ to the corresponding bit in the Interrupt Enable Register (I2SC_IER). The interrupt service routine associated to the I2SC interrupt request is executed whenever the Receive Ready or the Transmit Ready status bit is set.

49.6.24 MCAN High Priority Message Status

Name: MCAN_HPMS
Offset: 0x94
Reset: 0x00000000
Property: Read-only

This register is updated every time a Message ID filter element configured to generate a priority event matches. This can be used to monitor the status of incoming high priority messages and to enable fast access to these messages.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	FLST	FIDX[6:0]						
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	MSI[1:0]		BIDX[5:0]					
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit 15 – FLST Filter List
Indicates the filter list of the matching filter element.

Value	Description
0	Standard filter list
1	Extended filter list

Bits 14:8 – FIDX[6:0] Filter Index
Index of matching filter element. Range is 0 to MCAN_SIDFC.LSS - 1 resp. MCAN_XIDFC.LSE - 1.

Bits 7:6 – MSI[1:0] Message Storage Indicator

Value	Name	Description
0	NO_FIFO_SEL	No FIFO selected.
1	LOST	FIFO message lost.
2	FIFO_0	Message stored in FIFO 0.
3	FIFO_1	Message stored in FIFO 1.

Value	Name	Description
0	NONE	None
1	RISING	Rising edge of TIOAx
2	FALLING	Falling edge of TIOAx
3	EDGE	Each edge of TIOAx

Bit 15 – WAVE Waveform Mode

Value	Description
0	Capture mode is enabled.
1	Capture mode is disabled (Waveform mode is enabled).

Bit 14 – CPCTRG RC Compare Trigger Enable

Value	Description
0	RC Compare has no effect on the counter and its clock.
1	RC Compare resets the counter and starts the counter clock.

Bit 10 – ABETRG TIOAx or TIOBx External Trigger Selection

Value	Description
0	TIOBx is used as an external trigger.
1	TIOAx is used as an external trigger.

Bits 9:8 – ETRGEDG[1:0] External Trigger Edge Selection

Value	Name	Description
0	NONE	The clock is not gated by an external signal.
1	RISING	Rising edge
2	FALLING	Falling edge
3	EDGE	Each edge

Bit 7 – LDBDIS Counter Clock Disable with RB Loading

Value	Description
0	Counter clock is not disabled when RB loading occurs.
1	Counter clock is disabled when RB loading occurs.

Bit 6 – LDBSTOP Counter Clock Stopped with RB Loading

Value	Description
0	Counter clock is not stopped when RB loading occurs.
1	Counter clock is stopped when RB loading occurs.

Bits 5:4 – BURST[1:0] Burst Signal Selection

Value	Name	Description
0	NONE	The clock is not gated by an external signal.
1	XC0	XC0 is ANDed with the selected clock.
2	XC1	XC1 is ANDed with the selected clock.
3	XC2	XC2 is ANDed with the selected clock.

If conversion results are signed and resolution is less than 16 bits, the sign is extended up to the bit 15 (e.g., 0xF43 for 12-bit resolution will be read as 0xFF43 and 0x467 will be read as 0x0467).

52.6.6 Conversion Triggers

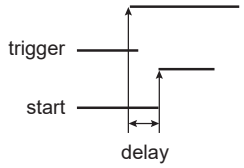
Conversions of the active analog channels are started with a software or hardware trigger. The software trigger is provided by writing a '1' to the bit START in the Control Register (AFEC_CR).

The hardware trigger can be one of the TIOA outputs of the Timer Counter channels, PWM Event line, or the external trigger input of the AFEC (ADTRG). The hardware trigger is selected with AFEC_MR.TRGSEL. The selected hardware trigger is enabled with AFEC_MR.TRGEN

The minimum time between two consecutive trigger events must be strictly greater than the duration of the longest conversion sequence according to configuration of registers AFEC_MR, AFEC_CHSR, AFEC_SEQ1R, AFEC_SEQ2R.

If a hardware trigger is selected, the start of a conversion is triggered after a delay starting at each rising edge of the selected signal. Due to asynchronous handling, the delay may vary in a range of two peripheral clock periods to one AFE clock period. This delay varies from trigger to trigger and so introduces a jitter error leading to a reduced Signal-to-Noise ratio performance.

Figure 52-6. Conversion Start with the Hardware Trigger



If one of the TIOA outputs is selected, the corresponding Timer Counter channel must be programmed in Waveform mode.

Only one start command is necessary to initiate a conversion sequence on all the channels. The AFEC hardware logic automatically performs the conversions on the active channels, then waits for a new request. The Channel Enable (AFEC_CHER) and Channel Disable (AFEC_CHDR) registers permit the analog channels to be enabled or disabled independently.

If the AFEC is used with a DMA, only the transfers of converted data from enabled channels are performed and the resulting data buffers should be interpreted accordingly.

52.6.7 Sleep Mode and Conversion Sequencer

The AFEC Sleep mode maximizes power saving by automatically deactivating the AFE when it is not being used for conversions. Sleep mode is selected by setting AFEC_MR.SLEEP.

Sleep mode is managed by a conversion sequencer, which automatically processes the conversions of all channels at lowest power consumption.

This mode can be used when the minimum period of time between two successive trigger events is greater than the startup period of the AFEC. Refer to the AFE Characteristics in the section "Electrical Characteristics".

When a start conversion request occurs, the AFE is automatically activated. As the analog cell requires a startup time, the logic waits during this lapse and starts the conversion on the enabled channels. When all conversions are complete, the AFE is deactivated until the next trigger. Triggers occurring during the sequence are not taken into account.

A fast wakeup mode is available in the AFEC_MR as a compromise between power-saving strategy and responsiveness. Setting the FWUP bit enables the Fast Wakeup mode. In Fast Wakeup mode, the AFE is

58.8.5 AFE Electrical Characteristics

Table 58-34. AFE INL and DNL, $f_{AFE\ CLOCK}$ = < 20 MHz Maximum, IBCTL = 10

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Differential Mode						
INL	Integral Non-Linearity	Gain = 1	-4	±0.7	4	LSB
		Gain = 2		±1		
		Gain = 4		±1.2		
DNL	Differential Non-Linearity	–	-2	±0.6	2	LSB
Single-Ended Mode						
INL	Integral Non-Linearity	Gain = 1	-6	±1	4	LSB
		Gain = 2		±1.3		
		Gain = 4		±1.7		
DNL	Differential Non-Linearity	–	-2	±0.6	2	LSB

Note: INL/DNL given inside the linear range of the AFE: 2% to 98% of VREFP.

Table 58-35. AFE INL and DNL, $f_{AFE\ CLOCK}$ = > 20 MHz to 40 MHz, IBCTL = 11

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Differential Mode						
INL	Integral Non-Linearity	Gain = 1	-12	±2	12	LSB
		Gain = 2		±2.1		
		Gain = 4		±2.5		
DNL	Differential Non-Linearity	–	-6	±2	6	LSB
Single-Ended Mode						
INL	Integral Non-Linearity	Gain = 1	-12	±2	12	LSB
		Gain = 2		±2.6		
		Gain = 4		±2.7		
DNL	Differential Non-Linearity	–	-6	±2	6	LSB

Note: INL/DNL given inside the linear range of the AFE: 2% to 98% of VREFP.

Table 58-36. AFE Offset and Gain Error, V_{VREFP} = 1.7V to 3.3V

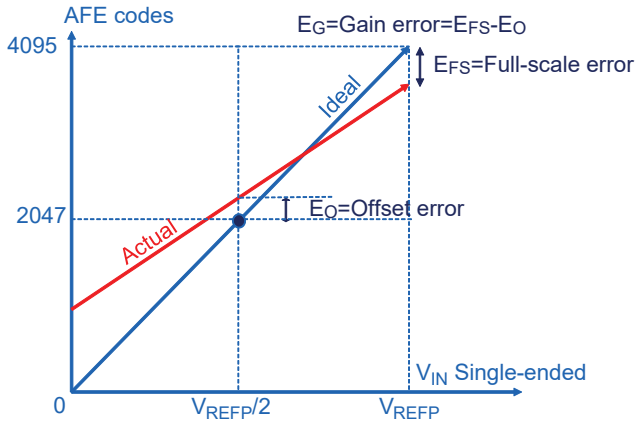
Symbol	Parameter	Conditions	Min	Typ(1)	Max	Unit
Differential Mode						
E_O	Differential Offset Error (see **Note 1**)	Gain=1	-20	–	35	LSB
E_G	Differential Gain Error	Gain=1	-0.3	0	0.7	%

Symbol	Parameter	Condition	Min	Max	Unit
C_i(1)	Capacitance for each I/O Pin	–	–	10	pF
f_{TWCK}	TWCK Clock Frequency	–	0	400	kHz
R_P	Value of Pull-up resistor	$f_{TWCK} \le 100$ kHz	(V_{DDIO} - 0.4V) ÷ 3mA	1000ns ÷ C_b	Ω
		$f_{TWCK} > 100$ kHz		300ns ÷ C_b	Ω
t_{LOW}	Low Period of the TWCK clock	$f_{TWCK} \le 100$ kHz	(3)	–	µs
		$f_{TWCK} > 100$ kHz	(3)	–	µs
t_{HIGH}	High period of the TWCK clock	$f_{TWCK} \le 100$ kHz	(4)	–	µs
		$f_{TWCK} > 100$ kHz	(4)	–	µs
$t_{HD;STA}$	Hold Time (repeated) START Condition	$f_{TWCK} \le 100$ kHz	t_{HIGH}	–	µs
		$f_{TWCK} > 100$ kHz	t_{HIGH}	–	µs
$t_{SU;STA}$	Set-up time for a repeated START condition	$f_{TWCK} \le 100$ kHz	t_{HIGH}	–	µs
		$f_{TWCK} > 100$ kHz	t_{HIGH}	–	µs
$t_{HD;DAT}$	Data hold time	$f_{TWCK} \le 100$ kHz	0	3 × t_{CPMCK}(5)	µs
		$f_{TWCK} > 100$ kHz	0	3 ×t_{CPMCK}(5)	µs
$t_{SU;DAT}$	Data setup time	$f_{TWCK} \le 100$ kHz	t_{LOW} - 3 × t_{CPMCK}(5)	–	ns
		$f_{TWCK} > 100$ kHz	t_{LOW} - 3 × t_{CPMCK}(5)	–	ns
$t_{SU;STO}$	Setup time for STOP condition	$f_{TWCK} \le 100$ kHz	t_{HIGH}	–	µs
		$f_{TWCK} > 100$ kHz	t_{HIGH}	–	µs
$t_{HD;STA}$	Hold Time (repeated) START Condition	$f_{TWCK} \le 100$ kHz	t_{HIGH}	–	µs
		$f_{TWCK} > 100$ kHz	t_{HIGH}	–	µs

Note:

1. Required only for $f_{TWCK} > 100$ kHz.
2. C_b = capacitance of one bus line in pF. Per I^2C standard, C_b max = 400pF.
3. The TWCK low period is defined as follows: $t_{LOW} = ((CLDIV \times 2^{CKDIV}) + 4) \times t_{MCK}$.
4. The TWCK high period is defined as follows: $t_{HIGH} = ((CHDIV \times 2^{CKDIV}) + 4) \times t_{MCK}$.
5. t_{CPMCK} = MCK bus period

Figure 59-14. Gain and Offset Errors in Single-ended Mode



where:

- Full-scale error E_{FS} =(E_{FS+})-(E_{FS-}), unit is LSB code
- Offset error E_O is the offset error measured for $V_{REFP/2}$= 0V
- Gain error E_G=100 x E_{FS} /4096, unit in %

The error values in the tables below include the DAC, the sample-and-hold error as well as the PGA gain error.

59.8.5 AFE Electrical Characteristics

Table 59-34. AFE INL and DNL, $f_{AFE\ CLOCK}$ = < 20 MHz Maximum, IBCTL = 10

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Differential Mode						
INL	Integral Non-Linearity	Gain = 1	-4	±0.7	4	LSB
		Gain = 2		±1		
		Gain = 4		±1.2		
DNL	Differential Non-Linearity	–	-2	±0.6	2	LSB
Single-Ended Mode						
INL	Integral Non-Linearity	Gain = 1	-6	±1	4	LSB
		Gain = 2		±1.3		
		Gain = 4		±1.7		
DNL	Differential Non-Linearity	–	-2	±0.6	2	LSB

Note: INL/DNL given inside the linear range of the AFE: 2% to 98% of VREFP.

Table 59-35. AFE INL and DNL, $f_{AFE\ CLOCK}$ = > 20 MHz to 40 MHz, IBCTL = 11

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Differential Mode						
INL	Integral Non-Linearity	Gain = 1	-12	±2	12	LSB