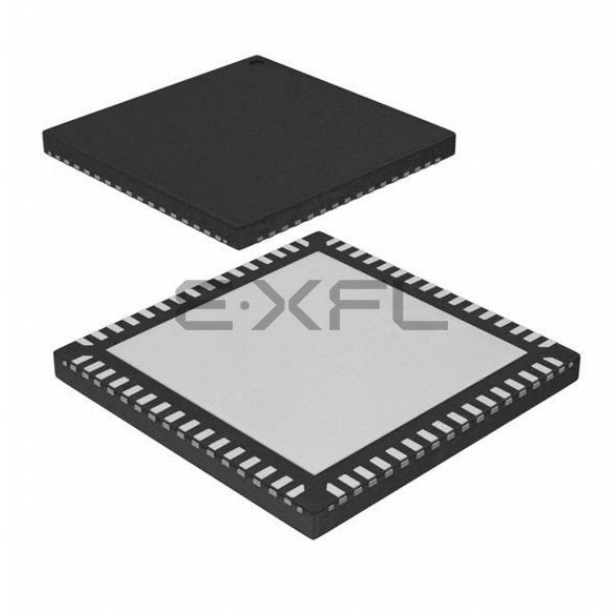


Welcome to E-XFL.COM

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, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	44
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256K x 8
Voltage - Supply (Vcc/Vdd)	1.08V ~ 3.6V
Data Converters	A/D 5x12b; D/A 1x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70j19a-mnt

31. Power Management Controller (PMC)

31.1 Description

The Power Management Controller (PMC) optimizes power consumption by controlling all system and user peripheral clocks. The PMC enables/disables the clock inputs to many of the peripherals and the Cortex-M7 processor.

The Supply Controller selects either the Slow RC oscillator or the 32.768 kHz crystal oscillator as the source of SLCK. The unused oscillator is disabled automatically so that power consumption is optimized.

By default, at startup, the chip runs out of MCK using the Main RC oscillator running at 12 MHz.

31.2 Embedded Characteristics

The Power Management Controller provides the following clocks:

- Master Clock (MCK), programmable from a few hundred Hz to the maximum operating frequency of the device. It is available to the modules running permanently, such as the Enhanced Embedded Flash Controller
- Processor Clock (HCLK), automatically switched off when entering the processor in Sleep mode
- Free-running processor Clock (FCLK)
- The Cortex-M7 SysTick external clock
- USB Clock (USB_48M), required by the USB peripheral
- Peripheral Clocks with independent ON/OFF control, provided to the peripherals
- Programmable Clock Outputs (PCKx), selected from the clock generator outputs to drive the device PCK pins
- Clock sources independent of MCK and HCLK, provided by internal PCKx for USART, UART, TC, Embedded Trace Macrocell (ETM) and CAN Clocks
- Generic Clock (GCLK) with controllable division and ON/OFF control, independent of MCK and HCLK. Provided to selected peripherals.

The Power Management Controller also provides the following features on clocks:

- A Main crystal oscillator failure detector
- A 32.768 kHz crystal oscillator frequency monitor
- A frequency counter on Main crystal oscillator or Main RC oscillator
- An on-the-fly adjustable Main RC oscillator frequency

31.15 Main Crystal Oscillator Failure Detection

The Main crystal oscillator failure detector monitors the Main crystal oscillator against the Slow RC oscillator and provides an automatic switchover of the MAINCK source to the Main RC oscillator in case of failure detection.

The failure detector can be enabled or disabled by configuring the CKGR_MOR.CFDEN, and it can also be disabled in either of the following cases:

- After a VDDCORE reset
- When the Main crystal oscillator is disabled (MOSCXTEN = 0)

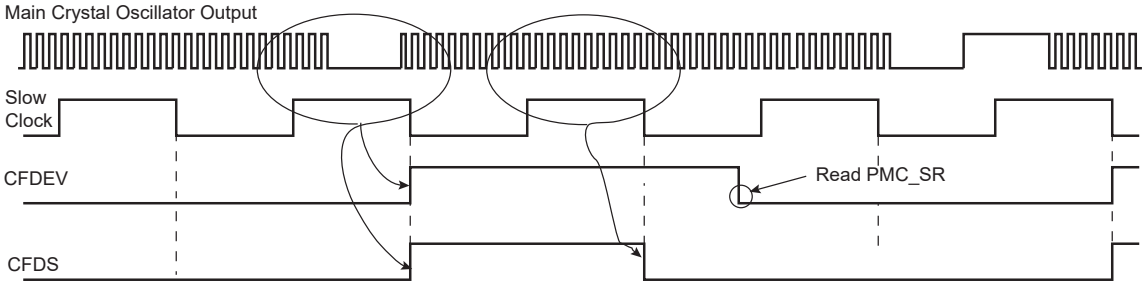
A failure is detected by means of a counter incrementing on the Main crystal oscillator output and detection logic is triggered by the Slow RC oscillator which is automatically enabled when CFDEN = 1.

The counter is cleared when the Slow RC oscillator clock signal is low and enabled when the signal is high. Thus, the failure detection time is one Slow RC oscillator period. If, during the high level period of the Slow RC oscillator clock signal, less than eight Main crystal oscillator clock periods have been counted, then a failure is reported. Note that when enabling the failure detector, up to two cycles of the Slow RC oscillator are needed to detect a failure of the Main crystal oscillator.

If a failure of Main crystal oscillator is detected, PMC_SR.CFDEV and PMC_SR.FOS both indicate a failure event. PMC_SR.CFDEV is cleared on read of PMC_SR, and PMC_SR.FOS is cleared by writing a ‘1’ to the FOCLR bit in the PMC Fault Output Clear Register (PMC_FOCR).

Only PMC_SR.CFDEV can generate an interrupt if the corresponding interrupt source is enabled in PMC_IER. The current status of the clock failure detection can be read at any time from PMC_SR.CFDS.

Figure 31-5. Clock Failure Detection Example



Note: Ratio of clock periods is for illustration purposes only.

If the Main crystal oscillator is selected as the source clock of MAINCK (CKGR_MOR.MOSCSEL = 1), and if the MCK source is PLLACK or UPLLCKDIV (CSS = 2 or 3), a clock failure detection automatically forces MAINCK to be the source clock for MCK. Then, regardless of the PMC configuration, a clock failure detection automatically forces the Main RC oscillator to be the source clock for MAINCK. If the Main RC oscillator is disabled when a clock failure detection occurs, it is automatically re-enabled by the clock failure detection mechanism.

Two Slow RC oscillator clock cycles are necessary to detect and switch from the Main crystal oscillator to the Main RC oscillator if the source of MCK is MAINCK, or three Slow RC oscillator clock cycles if the source of MCK is PLLACK or UPLLCKDIV.

A clock failure detection activates a fault output that is connected to the Pulse Width Modulator (PWM) Controller. With this connection, the PWM controller is able to force its outputs and to protect the driven device, if a clock failure is detected.

32.6.1.44 PIO Fall/Rise - Low/High Status Register

Name: PIO_FRLHSR
Offset: 0x00D8
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
	P31	P30	P29	P28	P27	P26	P25	P24
Access								
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	P23	P22	P21	P20	P19	P18	P17	P16
Access								
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	P15	P14	P13	P12	P11	P10	P9	P8
Access								
Reset	0	0	0	0	0	0	0	0

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

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 Edge/Level Interrupt Source Selection

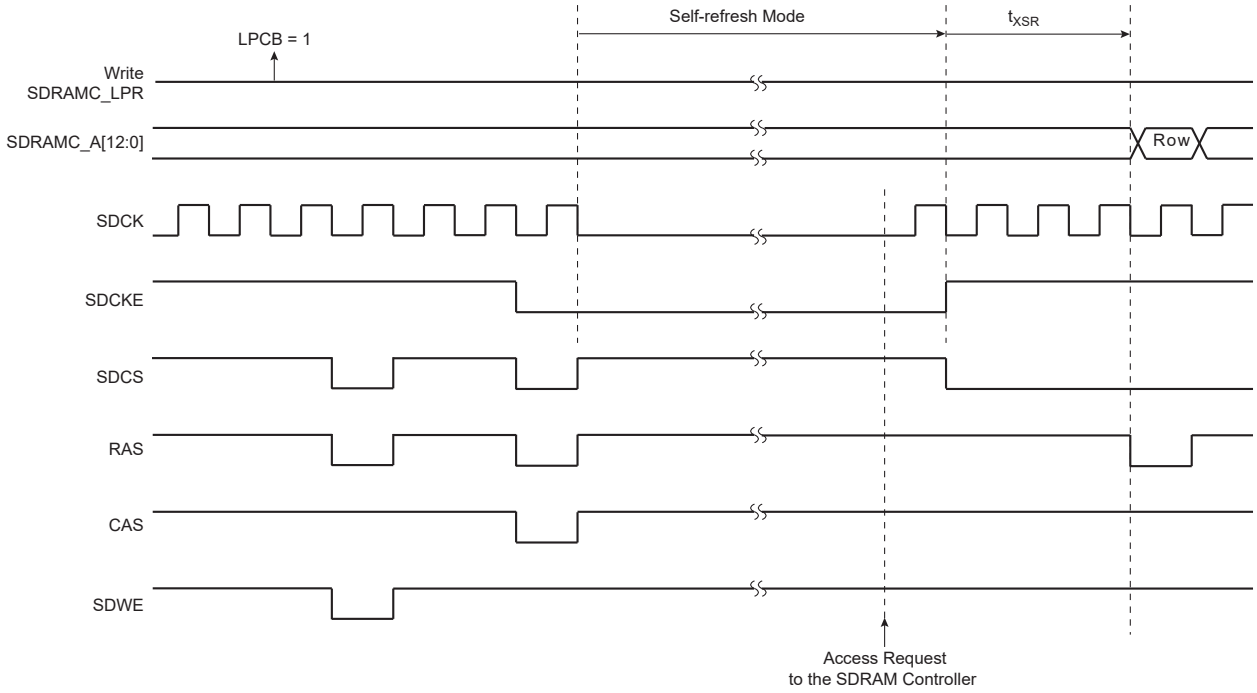
Value	Description
0	The interrupt source is a falling edge detection (if PIO_ELSR = 0) or low-level detection event (if PIO_ELSR = 1).
1	The interrupt source is a rising edge detection (if PIO_ELSR = 0) or high-level detection event (if PIO_ELSR = 1).

After initialization, as soon as the PASR/DS/TCSR fields are modified and Self-refresh mode is activated, the Extended Mode register is accessed automatically and the PASR/DS/TCSR bits are updated before entry into Self-refresh mode. This feature is not supported when SDRAMC shares an external bus with another controller.

The SDRAM device must remain in Self-refresh mode for a minimum period of t_{RAS} and may remain in Self-refresh mode for an indefinite period. Refer to the following figure.

Note: Some SDRAM providers impose some cycles of burst autorefresh immediately before self-refresh entry and immediately after self-refresh exit. For example, a SDRAM with 4096 rows will impose 4096 cycles of burst autorefresh. This constraint is not supported.

Figure 34-6. Self-refresh Mode Behavior

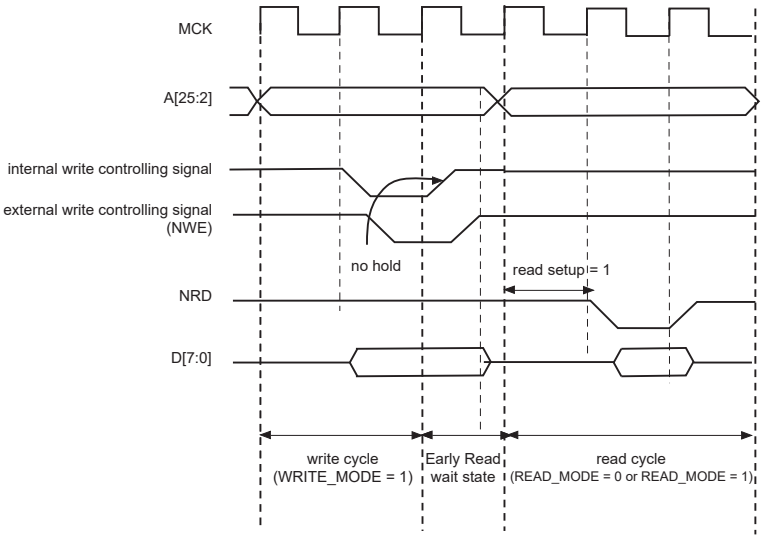


34.6.5.2 Low-power Mode

This mode is selected by configuring SDRAMC_LPR.LPCB to 2. Power consumption is greater than in Self-refresh mode. All the input and output buffers of the SDRAM device are deactivated except SDCKE, which remains low. In contrast to Self-refresh mode, the SDRAM device cannot remain in Low-power mode longer than the refresh period (64 ms for a whole device refresh operation). As no autorefresh operations are performed by the SDRAM itself, the SDRAMC carries out the refresh operation. The exit procedure is faster than in Self-refresh mode.

Refer to the following figure.

Figure 35-20. Early Read Wait State: NWE-controlled write with no hold followed by a read with one set-up cycle



35.11.3 Reload User Configuration Wait State

The user may change any of the configuration parameters by writing the SMC user interface.

When detecting that a new user configuration has been written in the user interface, the SMC inserts a wait state before starting the next access. This “reload user configuration wait state” is used by the SMC to load the new set of parameters to apply to next accesses.

The reload configuration wait state is not applied in addition to the chip select wait state. If accesses before and after re-programming the user interface are made to different devices (chip selects), then one single chip select wait state is applied.

On the other hand, if accesses before and after writing the user interface are made to the same device, a reload configuration wait state is inserted, even if the change does not concern the current chip select.

35.11.3.1 User Procedure

To insert a reload configuration wait state, the SMC detects a write access to any SMC_MODE register of the user interface. If the user only modifies timing registers (SMC_SETUP, SMC_PULSE, SMC_CYCLE registers) in the user interface, he must validate the modification by writing the SMC_MODE, even if no change was made on the mode parameters.

The user must not change the configuration parameters of an SMC chip select (Setup, Pulse, Cycle, Mode) if accesses are performed on this CS during the modification. Any change of the chip select parameters, while fetching the code from a memory connected on this CS, may lead to unpredictable behavior. The instructions used to modify the parameters of an SMC chip select can be executed from the internal RAM or from a memory connected to another CS.

35.11.3.2 Slow Clock Mode Transition

A reload configuration wait state is also inserted when the Slow Clock mode is entered or exited, after the end of the current transfer (see ”Slow Clock Mode”).

35.11.4 Read to Write Wait State

Due to an internal mechanism, a wait cycle is always inserted between consecutive read and write SMC accesses.

36.9.30 XDMAC Channel x Source Microblock Stride Register [x = 0..23]

Name: XDMAC_CSUS
Offset: 0x80 + n*0x40 [n=0..23]
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
	SUBS[23:16]							
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	15	14	13	12	11	10	9	8
	SUBS[15:8]							
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
	SUBS[7:0]							
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 23:0 – SUBS[23:0] Channel x Source Microblock Stride
Two’s complement microblock stride for channel x.

Offset	Name	Bit Pos.								
0xB8	GMAC_WOL	7:0	IP[7:0]							
		15:8	IP[15:8]							
		23:16					MTI	SA1	ARP	MAG
		31:24								
0xBC	GMAC_IPGS	7:0	FL[7:0]							
		15:8	FL[15:8]							
		23:16								
		31:24								
0xC0	GMAC_SVLAN	7:0	VLAN_TYPE[7:0]							
		15:8	VLAN_TYPE[15:8]							
		23:16								
		31:24	ESVLAN							
0xC4	GMAC_TPFCP	7:0	PEV[7:0]							
		15:8	PQ[7:0]							
		23:16								
		31:24								
0xC8	GMAC_SAMB1	7:0	ADDR[7:0]							
		15:8	ADDR[15:8]							
		23:16	ADDR[23:16]							
		31:24	ADDR[31:24]							
0xCC	GMAC_SAMT1	7:0	ADDR[7:0]							
		15:8	ADDR[15:8]							
		23:16								
		31:24								
0xD0 ... 0xDB	Reserved									
0xDC	GMAC_NSC	7:0	NANOSEC[7:0]							
		15:8	NANOSEC[15:8]							
		23:16			NANOSEC[21:16]					
		31:24								
0xE0	GMAC_SCL	7:0	SEC[7:0]							
		15:8	SEC[15:8]							
		23:16	SEC[23:16]							
		31:24	SEC[31:24]							
0xE4	GMAC_SCH	7:0	SEC[7:0]							
		15:8	SEC[15:8]							
		23:16								
		31:24								
0xE8	GMAC_EFTSH	7:0	RUD[7:0]							
		15:8	RUD[15:8]							
		23:16								
		31:24								
0xEC	GMAC_EFRSH	7:0	RUD[7:0]							
		15:8	RUD[15:8]							
		23:16								

38.8.28 GMAC Transmit PFC Pause Register

Name: GMAC_TPFCP
Offset: 0x0C4
Reset: 0x00000000
Property: -

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
	PQ[7:0]							
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
	PEV[7:0]							
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 15:8 – PQ[7:0] Pause Quantum
When the Remove FCS bit in the GMAC Network Configuration register (GMAC_NCFGR.RFCS) is written to '1', and one or more bits in this bit field are written to '0', the associated PFC pause frame's pause quantum field value is taken from the Transmit Pause Quantum register (GMAC_TPQ).

For each entry equal to '1' in this bit field, the pause quantum associated with that entry will be zero.

Bits 7:0 – PEV[7:0] Priority Enable Vector
When the Remove FCS bit in the GMAC Network Configuration register (GMAC_NCFGR.RFCS) is written to '1', the priority enable vector of the PFC priority-based pause frame is set to the value stored in this bit field.

38.8.90 GMAC PTP Event Frame Transmitted Nanoseconds Register

Name: GMAC_EFTN
Offset: 0x1E4
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
			RUD[29:24]					
Access			R	R	R	R	R	R
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	RUD[23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	RUD[15:8]							
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
	RUD[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 29:0 – RUD[29:0] Register Update
The register is updated with the value that the IEEE 1588 Timer Nanoseconds Register holds when the SFD of a PTP transmit primary event crosses the MII interface. An interrupt is issued when the bit field is updated.

Offset	Name	Bit Pos.								
		31:24								
0x0590	USBHS_HSTPIPIFR0 (INTPIPES)	7:0	SHORTPACKETIS	RXSTALLDIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0590	USBHS_HSTPIPIFR0 (ISOPIPES)	7:0	SHORTPACKETIS	CRCERRIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0594	USBHS_HSTPIPIFR1 (INTPIPES)	7:0	SHORTPACKETIS	RXSTALLDIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0594	USBHS_HSTPIPIFR1 (ISOPIPES)	7:0	SHORTPACKETIS	CRCERRIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0598	USBHS_HSTPIPIFR2 (INTPIPES)	7:0	SHORTPACKETIS	RXSTALLDIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0598	USBHS_HSTPIPIFR2 (ISOPIPES)	7:0	SHORTPACKETIS	CRCERRIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x059C	USBHS_HSTPIPIFR3 (INTPIPES)	7:0	SHORTPACKETIS	RXSTALLDIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x059C	USBHS_HSTPIPIFR3 (ISOPIPES)	7:0	SHORTPACKETIS	CRCERRIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x05A0	USBHS_HSTPIPIFR4 (INTPIPES)	7:0	SHORTPACKETIS	RXSTALLDIS	OVERFIS	NAKEDIS	PERRIS	UNDERFIS	TXOUTIS	RXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								

1: Set when the current bank is ready to accept a new IN packet. This triggers a PEP_x interrupt if TXINE = 1.

For IN endpoints:

0: Cleared when TXINIC = 1. This acknowledges the interrupt, which has no effect on the endpoint FIFO. USBHS_DEVEPTISRx.TXINI shall always be cleared before clearing USBHS_DEVEPTIMRx.FIFOCON.

1: Set at the same time as USBHS_DEVEPTIMRx.FIFOCON when the current bank is free. This triggers a PEP_x interrupt if TXINE = 1.

The user writes into the FIFO and clears the USBHS_DEVEPTIMRx.FIFOCON bit to allow the USBHS to send the data. If the IN endpoint is composed of multiple banks, this also switches to the next bank. The USBHS_DEVEPTISRx.TXINI and USBHS_DEVEPTIMRx.FIFOCON bits are set/cleared in accordance with the status of the next bank.

This bit is inactive (cleared) for OUT endpoints.

Bit 0 – RXINIC Received IN Data Interrupt Clear

Bit 19 – PECERR PEC Error (cleared on read)

Value	Description
0	No SMBus PEC error occurred since the last read of TWIHS_SR.
1	An SMBus PEC error occurred since the last read of TWIHS_SR.

Bit 18 – TOUT Timeout Error (cleared on read)

Value	Description
0	No SMBus timeout occurred since the last read of TWIHS_SR.
1	An SMBus timeout occurred since the last read of TWIHS_SR.

Bit 16 – MCACK Master Code Acknowledge (cleared on read)
MACK used in Slave mode:

Value	Description
0	No Master Code has been received since the last read of TWIHS_SR.
1	A Master Code has been received since the last read of TWIHS_SR.

Bit 11 – EOSACC End Of Slave Access (cleared on read)
This bit is used in Slave mode only.

EOSACC behavior can be seen in Repeated Start and Reversal from Read Mode to Write Mode and Repeated Start and Reversal from Write Mode to Read Mode.

Value	Description
0	A slave access is being performing.
1	The Slave Access is finished. End Of Slave Access is automatically set as soon as SVACC is reset.

Bit 10 – SCLWS Clock Wait State
This bit is used in Slave mode only.

SCLWS behavior can be seen in the figures, Clock Stretching in Read Mode and Clock Stretching in Write Mode.

Value	Description
0	The clock is not stretched.
1	The clock is stretched. TWIHS_THR / TWIHS_RHR buffer is not filled / emptied before the transmission / reception of a new character.

Bit 9 – ARBLST Arbitration Lost (cleared on read)
This bit is used in Master mode only.

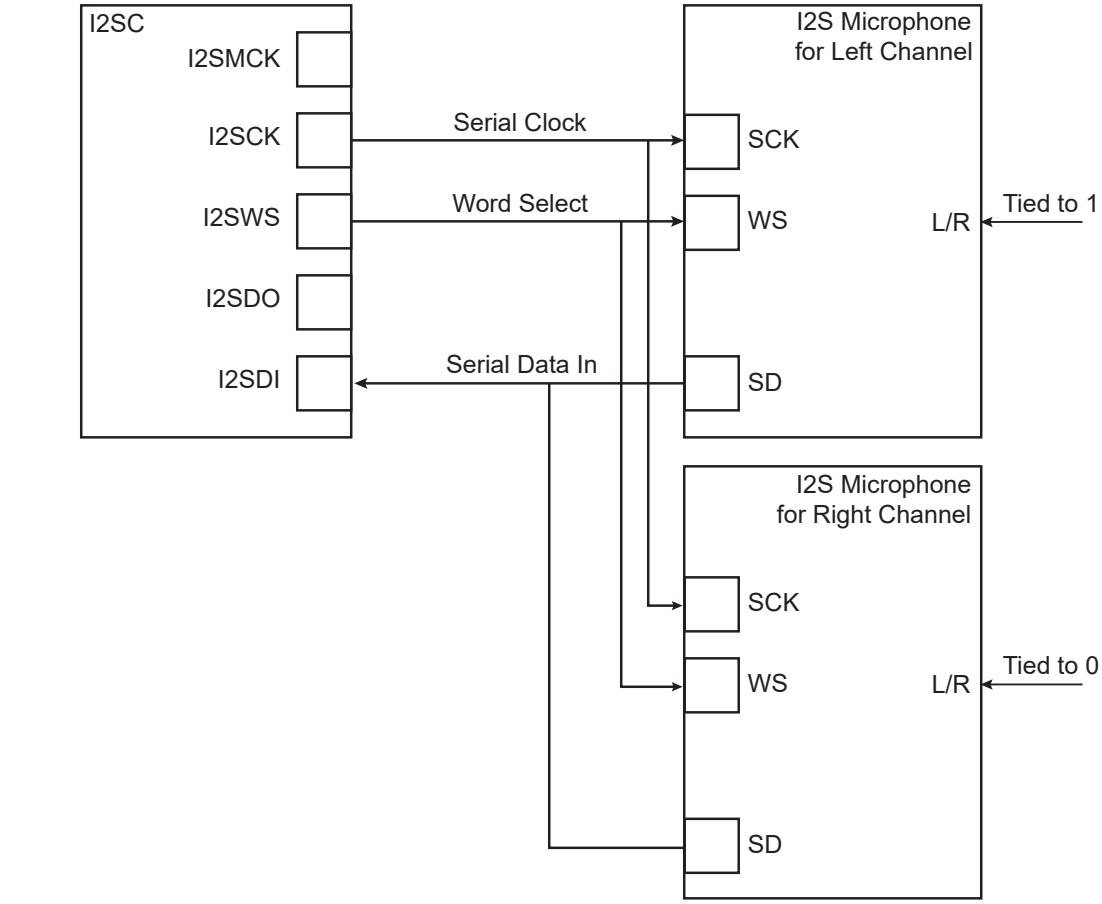
Value	Description
0	Arbitration won.
1	Arbitration lost. Another master of the TWIHS bus has won the multimaster arbitration. TXCOMP is set at the same time.

Bit 8 – NACK Not Acknowledged (cleared on read)

- NACK used in Master mode:

0: Each data byte has been correctly received by the far-end side TWIHS slave component.

Figure 45-6. Dual Microphone Application Block Diagram



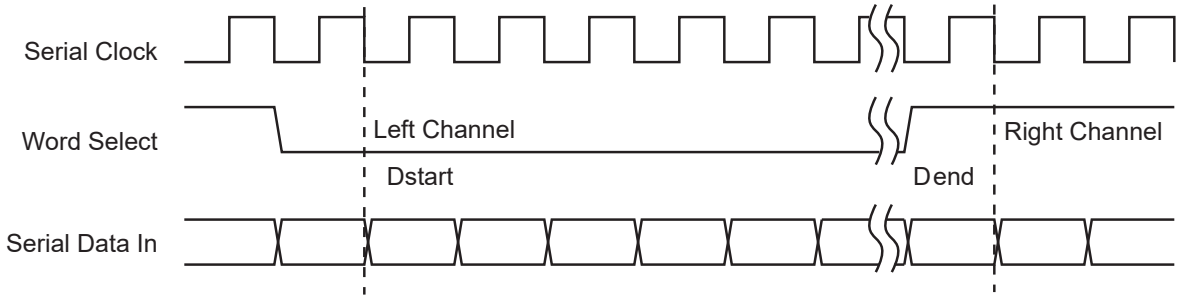
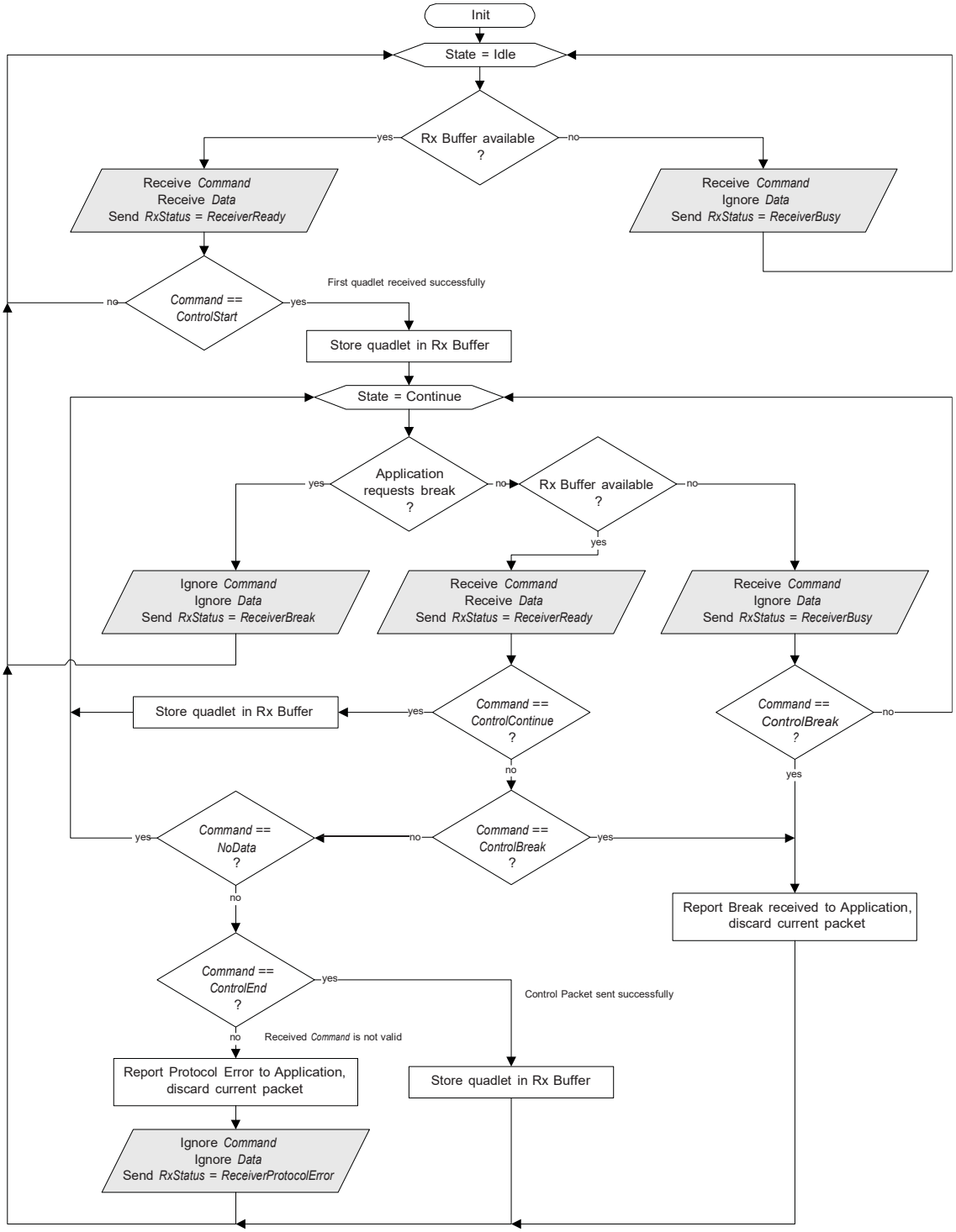


Figure 48-9. Control Packet Rx Device Protocol



Synchronous

Synchronous stream data is sent in a continuous and broadcast fashion, without block information. Therefore, receiving Devices must not respond to the synchronous command; thereby leaving RxStatus in the ReceiverReady state (logic low). For 3-pin MediaLB, the required pull-down on MLBS leaves this signal in the ReceiverReady command when no synchronous data is transmitted on the MLBD line.

Value	Name	Description
2	CLEAR	Clear
3	TOGGLE	Toggle

Bits 25:24 – BCPB[1:0] RB Compare Effect on TIOBx

Value	Name	Description
0	NONE	None
1	SET	Set
2	CLEAR	Clear
3	TOGGLE	Toggle

Bits 23:22 – ASWTRG[1:0] Software Trigger Effect on TIOAx

Value	Name	Description
0	NONE	None
1	SET	Set
2	CLEAR	Clear
3	TOGGLE	Toggle

Bits 21:20 – AEEVT[1:0] External Event Effect on TIOAx

Value	Name	Description
0	NONE	None
1	SET	Set
2	CLEAR	Clear
3	TOGGLE	Toggle

Bits 19:18 – ACPC[1:0] RC Compare Effect on TIOAx

Value	Name	Description
0	NONE	None
1	SET	Set
2	CLEAR	Clear
3	TOGGLE	Toggle

Bits 17:16 – ACPA[1:0] RA Compare Effect on TIOAx

Value	Name	Description
0	NONE	None
1	SET	Set
2	CLEAR	Clear
3	TOGGLE	Toggle

Bit 15 – WAVE Waveform Mode

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

Bits 14:13 – WAVSEL[1:0] Waveform Selection

  - PWM Sync Channels Mode Register
  - PWM Channel Mode Register
  - PWM Stepper Motor Mode Register
  - PWM Fault Protection Value Register 2
  - PWM Leading-Edge Blanking Register
  - PWM Channel Mode Update Register
- Register group 3:
  - PWM Spread Spectrum Register
  - PWM Spread Spectrum Update Register
  - PWM Channel Period Register
  - PWM Channel Period Update Register
- Register group 4:
  - PWM Channel Dead Time Register
  - PWM Channel Dead Time Update Register
- Register group 5:
  - PWM Fault Mode Register
  - PWM Fault Protection Value Register 1

There are two types of write protection:

- SW write protection—can be enabled or disabled by software
- HW write protection—can be enabled by software but only disabled by a hardware reset of the PWM controller

Both types of write protection can be applied independently to a particular register group by means of the WPCMD and WPRGx fields in PWM_WPCR. If at least one type of write protection is active, the register group is write-protected. The value of field WPCMD defines the action to be performed:

- 0: Disables SW write protection of the register groups of which the bit WPRGx is at ‘1’
- 1: Enables SW write protection of the register groups of which the bit WPRGx is at ‘1’
- 2: Enables HW write protection of the register groups of which the bit WPRGx is at ‘1’

At any time, the user can determine whether SW or HW write protection is active in a particular register group by the fields WPSWS and WPHWS in the PWM Write Protection Status Register (PWM_WPSR).

If a write access to a write-protected register is detected, the WPVS flag in PWM_WPSR is set and the field WPVSRC indicates the register in which the write access has been attempted.

The WPVS and WPVSRC fields are automatically cleared after reading PWM_WPSR.

Offset	Name	Bit Pos.								
		23:16	DTL[7:0]							
		31:24	DTL[15:8]							
0x021C	PWM_DTUPD0	7:0	DTHUPD[7:0]							
		15:8	DTHUPD[15:8]							
		23:16	DTLUPD[7:0]							
		31:24	DTLUPD[15:8]							
0x0220	PWM_CMR1	7:0					CPRE[3:0]			
		15:8			TCTS	DPOLI	UPDS	CES	CPOL	CALG
		23:16					PPM	DTLI	DTHI	DTE
		31:24								
0x0224	PWM_CDTY1	7:0	CDTY[7:0]							
		15:8	CDTY[15:8]							
		23:16	CDTY[23:16]							
		31:24								
0x0228	PWM_CDTYUPD1	7:0	CDTYUPD[7:0]							
		15:8	CDTYUPD[15:8]							
		23:16	CDTYUPD[23:16]							
		31:24								
0x022C	PWM_CPRD1	7:0	CPRD[7:0]							
		15:8	CPRD[15:8]							
		23:16	CPRD[23:16]							
		31:24								
0x0230	PWM_CPRDUPD1	7:0	CPRDUPD[7:0]							
		15:8	CPRDUPD[15:8]							
		23:16	CPRDUPD[23:16]							
		31:24								
0x0234	PWM_CCNT1	7:0	CNT[7:0]							
		15:8	CNT[15:8]							
		23:16	CNT[23:16]							
		31:24								
0x0238	PWM_DT1	7:0	DTH[7:0]							
		15:8	DTH[15:8]							
		23:16	DTL[7:0]							
		31:24	DTL[15:8]							
0x023C	PWM_DTUPD1	7:0	DTHUPD[7:0]							
		15:8	DTHUPD[15:8]							
		23:16	DTLUPD[7:0]							
		31:24	DTLUPD[15:8]							
0x0240	PWM_CMR2	7:0					CPRE[3:0]			
		15:8			TCTS	DPOLI	UPDS	CES	CPOL	CALG
		23:16					PPM	DTLI	DTHI	DTE
		31:24								
0x0244	PWM_CDTY2	7:0	CDTY[7:0]							
		15:8	CDTY[15:8]							
		23:16	CDTY[23:16]							
		31:24								

59.4.9.2 Printed Circuit Board (PCB)

To minimize inductive and capacitive parasitics associated with XIN, XOUT, XIN32 and XOUT32 nets, it is recommended to route them as short as possible. Additionally, it is of prime importance to keep these nets away from noisy switching signals (clock, data, PWM, etc.). A good practice is to shield them with a quiet ground net to avoid coupling to neighboring signals.

59.5 PLLA Characteristics

Table 59-26. PLLA Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{IN}	Input Frequency	–	8	–	32	MHz
f_{OUT}	Output Frequency	–	160	–	500	MHz
I_{PLL}	Current Consumption	Active mode at 160 MHz at 1.2V	–	2.2	3	mA
		Active mode at 500 MHz at 1.2V	–	8	12	
t_{START}	Startup Time	–	–	–	300	µs

59.6 PLLUSB Characteristics

Table 59-27. PLLUSB Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{IN}	Input Frequency	–	–	12 or 16	–	MHz
f_{OUT}	Output Frequency	–	–	480	–	MHz
I_{PLLUSB}	Current Consumption	In Active mode, on VDDPLLUSB	–	4.9	6.4	mA
		In Active mode, on VDDCORE	–	0.4	1	mA
t_{START}	Startup Time	–	–	–	50	µs

59.7 USB Transceiver Characteristics

The device conforms to all voltage, power, and timing characteristics and specifications as set forth in the USB 2.0 Specification. Please refer to the USB 2.0 Specification for more information.

Table 59-28. USB Transceiver Dynamic Power Consumption

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{BIAS}	Bias Current Consumption on VBG	–	–	–	12	mA
$I_{VDDUTMII}$	HS Transceiver Current Consumption	HS transmission	–	–	44	mA
	HS Transceiver Current Consumption	HS reception	–	–	24	mA
	LS / FS Transceiver Current Consumption	FS transmission 0m cable (see **Note 1**)	–	–	5	mA

Signal Name	Recommended Pin Connection	Description
TD	Application dependent.	SSC Transmit Data Pulled-up input (100 kOhm) to VDDIO at reset.
RD	Application dependent.	SSC Receive Data Pulled-up input (100 kOhm) to VDDIO at reset.
TK	Application dependent.	SSC Transmit Clock Pulled-up input (100 kOhm) to VDDIO at reset.
RK	Application dependent.	SSC Receive Clock I Pulled-up input (100 kOhm) to VDDIO at reset.
TF	Application dependent.	SSC Transmit Frame Sync Pulled-up input (100 kOhm) to VDDIO at reset.
RF	Application dependent.	SSC Receive Frame Sync Pulled-up input (100 kOhm) to VDDIO at reset.
Image Sensor Interface		
ISI_D0–ISI_D11	Application dependent. (Signal can be level-shifted depending on the image sensor characteristics)	Image Sensor Data Pulled-up inputs (100 kOhm) to VDDIO at reset.
ISI_MCK	Application dependent. (Signal can be level-shifted depending on the image sensor characteristics)	Image sensor reference clock. No dedicated signal, PCK1 can be used. Pulled-up input (100 kOhm) to VDDIO at reset.
ISI_HSYNC	Application dependent. (Signal can be level-shifted depending on the image sensor characteristics)	Image sensor horizontal synchro Pulled-up input (100 kOhm) to VDDIO at reset.
ISI_VSYNC	Application dependent. (Signal can be level-shifted depending on the image sensor characteristics)	Image sensor vertical synchro Pulled-up input (100 kOhm) to VDDIO at reset.
ISI_PCK	Application dependent. (Signal can be level-shifted depending on the image sensor characteristics)	Image sensor data clock Pulled-up input (100 kOhm) to VDDIO at reset.
Timer/Counter		
TCLKx	Application dependent.	TC Channel x External Clock Input Pulled-up inputs (100 kOhm) to VDDIO at reset.
TIOAx	Application dependent.	TC Channel x I/O Line A