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	CANbus, Ethernet, I²C, IrDA, LINbus, SPI, SSC, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	44
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 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-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsame70j20a-ant

7.8 Fast Startup

The SAM E70/S70/V70/V71 allows the processor to restart in a few microseconds while the processor is in Wait mode or in Sleep mode. A fast startup can occur upon detection of a low level on any of the following wakeup sources:

- WKUP0 to WKUP13 pins
- Supply Monitor
- RTC alarm
- RTT alarm
- USBHS interrupt line (WAKEUP)
- Processor debug request (CDBGPWRUPREQ)
- GMAC wake on LAN event

Note: CAN wakeup requires the use of any WKUP0–13 pin.

The fast restart circuitry is fully asynchronous and provides a fast startup signal to the Power Management Controller. As soon as the fast startup signal is asserted, the PMC automatically restarts the Main RC oscillator, switches the Master clock on this clock and re-enables the processor clock.

19.4.2 Bus Matrix Slave Configuration Registers

Name: MATRIX_SCFGx
Offset: 0x40 + x*0x04 [x=0..8]
Reset: 0x000001FF
Property: Read/Write

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			FIXED_DEFMSTR[3:0]				DEFMSTR_TYPE[1:0]	
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
							SLOT_CYCLE[8:7]	
Access							R/W	R/W
Reset							0	1

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

Bits 21:18 – FIXED_DEFMSTR[3:0] Fixed Default Master
Number of the Default Master for this slave. Only used if DEFMSTR_TYPE is 2. Specifying the number of a master which is not connected to the selected slave is equivalent to setting DEFMSTR_TYPE to 0.

Bits 17:16 – DEFMSTR_TYPE[1:0] Default Master Type

Value	Name	Description
0	NONE	No Default Master—At the end of the current slave access, if no other master request is pending, the slave is disconnected from all masters. This results in a one clock cycle latency for the first access of a burst transfer or for a single access.
1	LAST	Last Default Master—At the end of the current slave access, if no other master request is pending, the slave stays connected to the last master having accessed it. This results in not having one clock cycle latency when the last master tries to access the slave again.
2	FIXED	Fixed Default Master—At the end of the current slave access, if no other master request is pending, the slave connects to the fixed master the number that has been written in the FIXED_DEFMSTR field.

35.13.3 Ready Mode

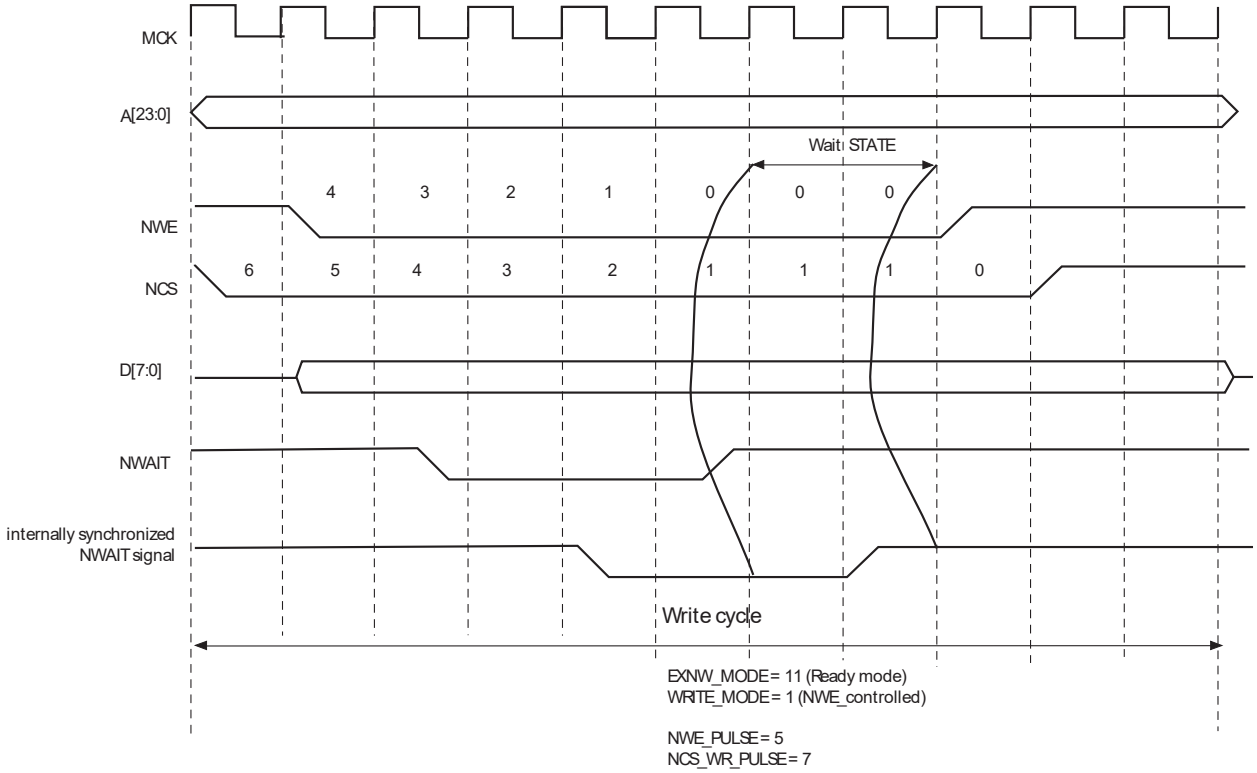
In Ready mode (SMC_MODE.EXNW_MODE = 11), the SMC behaves differently. Normally, the SMC begins the access by down counting the setup and pulse counters of the read/write controlling signal. In the last cycle of the pulse phase, the resynchronized NWAIT signal is examined.

If asserted, the SMC suspends the access as shown in Figure 35-29 and Figure 35-30. After deassertion, the access is completed: the hold step of the access is performed.

This mode must be selected when the external device uses deassertion of the NWAIT signal to indicate its ability to complete the read or write operation.

If the NWAIT signal is deasserted before the end of the pulse, or asserted after the end of the pulse of the controlling read/write signal, it has no impact on the access length as shown in Figure 35-30.

Figure 35-29. NWAIT Assertion in Write Access: Ready Mode (SMC_MODE.EXNW_MODE = 11)



38.8.42 GMAC Multicast Frames Transmitted Register

Name: GMAC_MFT
Offset: 0x110
Reset: 0x00000000
Property: -

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

Bit	23	22	21	20	19	18	17	16
	MFTX[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
	MFTX[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
	MFTX[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – MFTX[31:0] Multicast Frames Transmitted without Error
This register counts the number of multicast frames successfully transmitted without error, i.e., no underrun and not too many retries. Excludes pause frames.

38.8.89 GMAC PTP Event Frame Transmitted Seconds Low Register

Name: GMAC_EFTSL
Offset: 0x1E0
Reset: 0x00000000
Property: Read-only

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

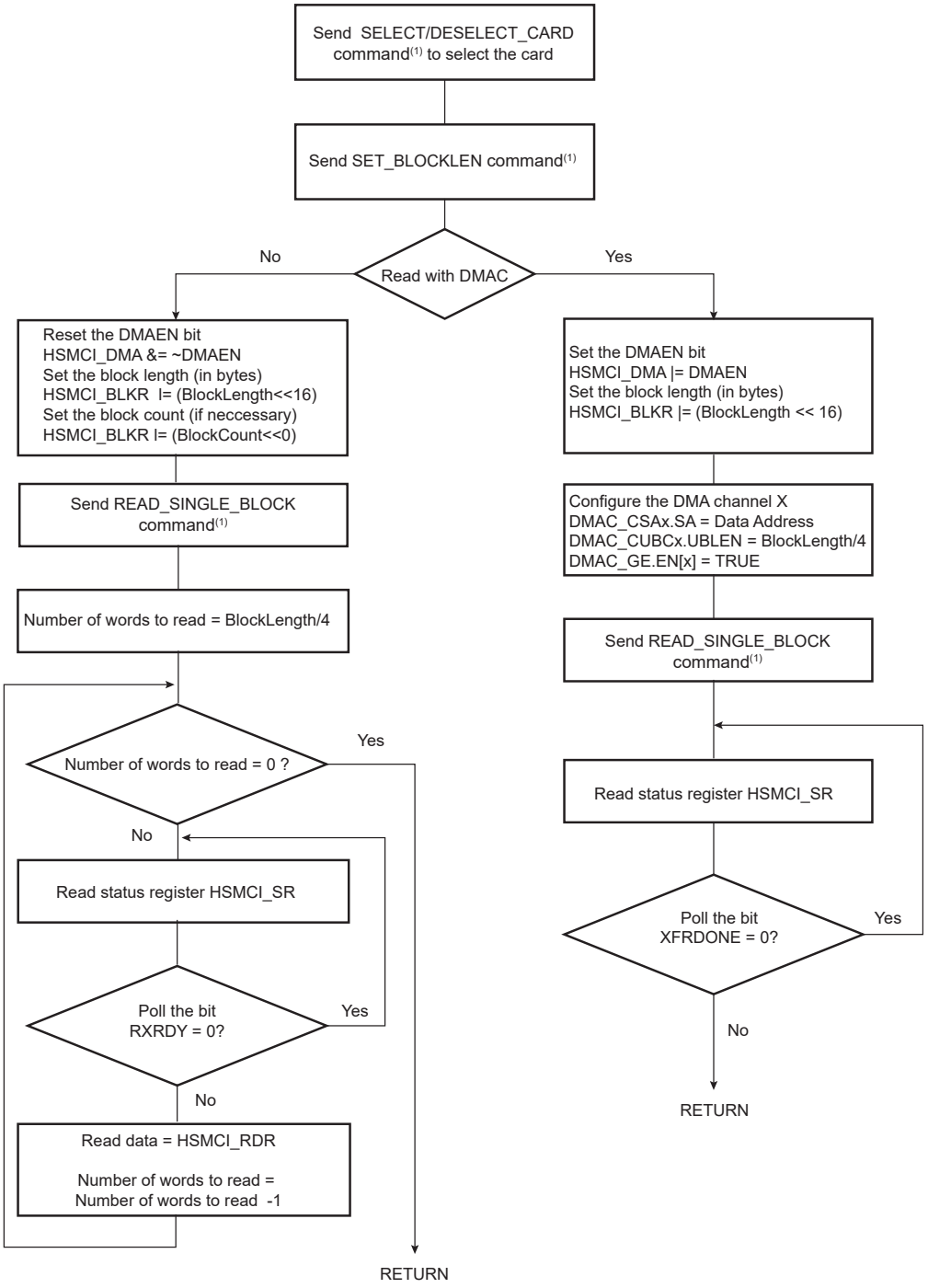
Bit	23	22	21	20	19	18	17	16
	RUD[23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

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 31:0 – RUD[31:0] Register Update
The register is updated with the value that the IEEE 1588 Timer Seconds Register holds when the SFD of a PTP transmit primary event crosses the MII interface. An interrupt is issued when the register is updated.

Figure 40-8. Read Functional Flow Diagram



Note 1: It is assumed that this command has been correctly sent (see the Command/Response Functional Flow Diagram).

40.8.4 Write Operation

In write operation, the HSMCI Mode Register (HSMCI_MR) is used to define the padding value when writing non-multiple block size. If the bit PADV is 0, then 0x00 value is used when padding data, otherwise 0xFF is used.

43.6.5.8 Asynchronous Partial Wakeup (SleepWalking)

The TWIHS includes an asynchronous start condition detector. It is capable of waking the device up from a Sleep mode upon an address match (and optionally an additional data match), including Sleep modes where the TWIHS peripheral clock is stopped.

After detecting the START condition on the bus, the TWIHS stretches TWCK until the TWIHS peripheral clock has started. The time required for starting the TWIHS depends on which Sleep mode the device is in. After the TWIHS peripheral clock has started, the TWIHS releases its TWCK stretching and receives one byte of data (slave address) on the bus. At this time, only a limited part of the device, including the TWIHS module, receives a clock, thus saving power. If the address phase causes a TWIHS address match (and, optionally, if the first data byte causes data match as well), the entire device is woken up and normal TWIHS address matching actions are performed. Normal TWIHS transfer then follows. If the TWIHS is not addressed (or if the optional data match fails), the TWIHS peripheral clock is automatically stopped and the device returns to its original Sleep mode.

The TWIHS has the capability to match on more than one address. The SADR1EN, SADR2EN and SADR3EN bits in TWIHS_SMR enable address matching on additional addresses which can be configured through SADR1, SADR2 and SADR3 fields in the TWIHS_SWMR. The SleepWalking matching process can be extended to the first received data byte if TWIHS_SMR.DATAMEN is set and, in this case, a complete matching includes address matching and first received data matching. TWIHS_SWMR.DATAM configures the data to match on the first received byte.

When the system is in Active mode and the TWIHS enters Asynchronous Partial Wakeup mode, the flag SVACC must be programmed as the unique source of the TWIHS interrupt and the data match comparison must be disabled.

When the system exits Wait mode as the result of a matching condition, the SVACC flag is used to determine if the TWIHS is the source of exit.

Figure 43-39. Address Match Only (Data Matching Disabled)

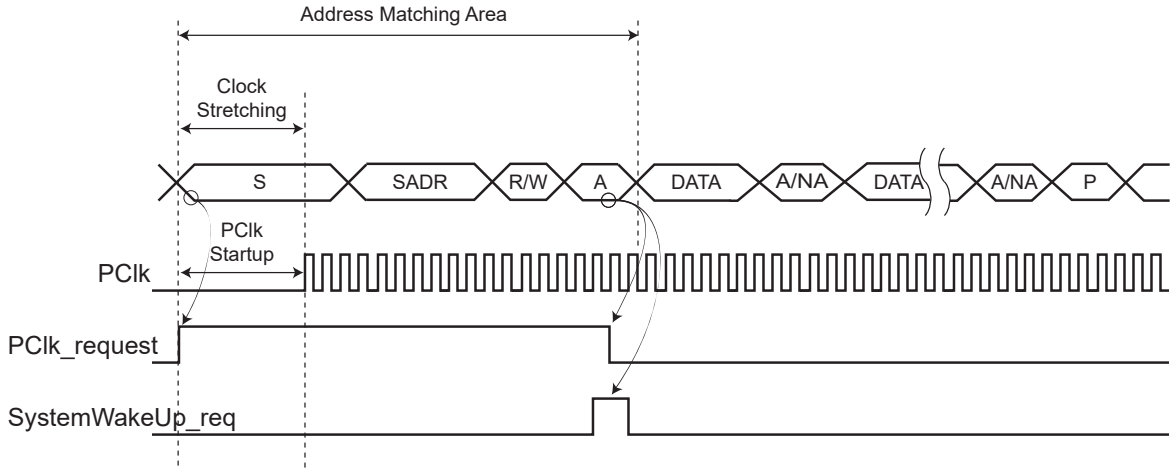
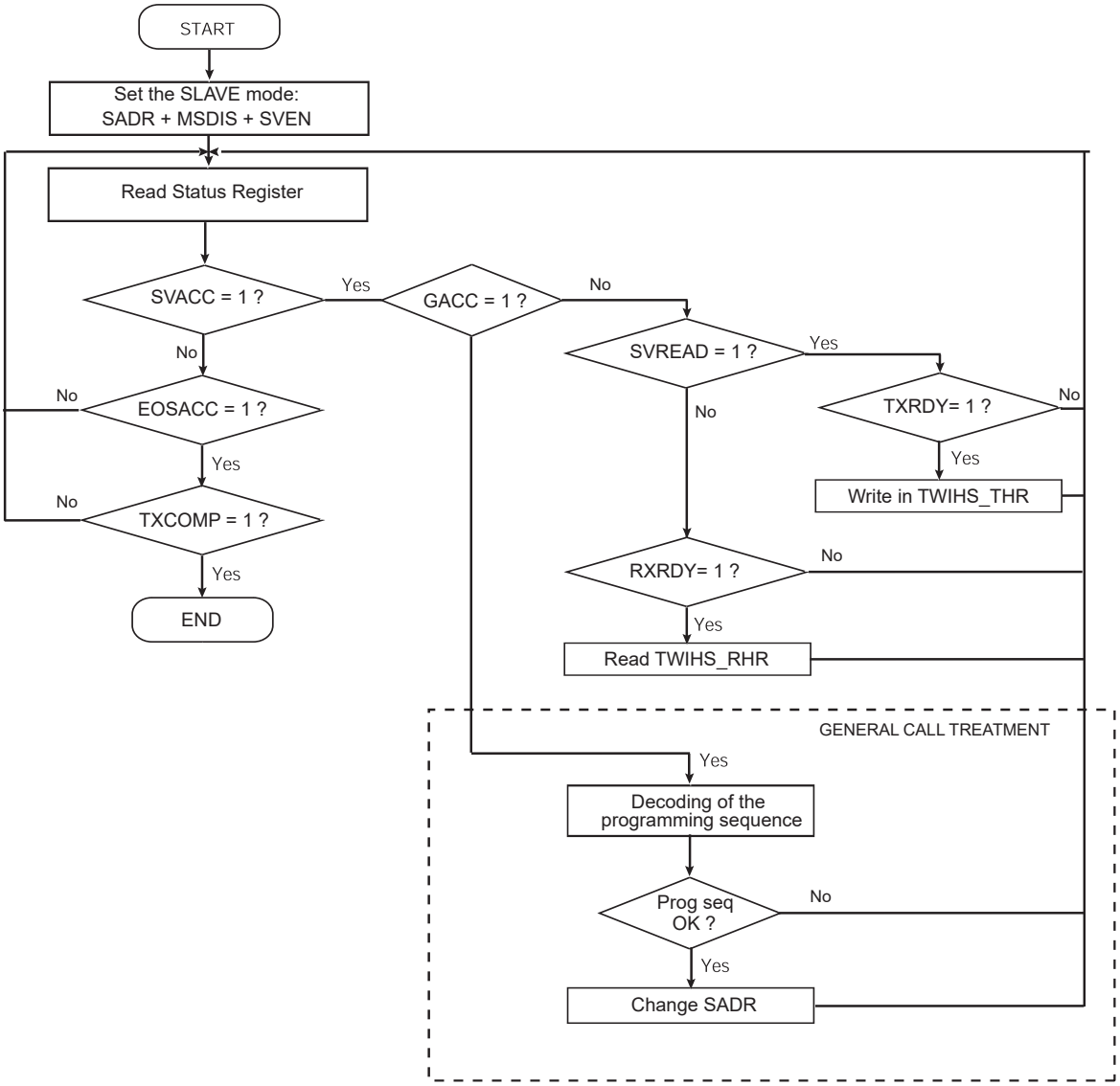


Figure 43-43. Read Write Flowchart in Slave Mode



45.8.3 I2SC Status Register

Name: I2SC_SR
Offset: 0x08
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			TXURCH[1:0]					
Access			R	R				
Reset			0	0				

Bit	15	14	13	12	11	10	9	8
							RXORCH[1:0]	
Access							R	R
Reset							0	0

Bit	7	6	5	4	3	2	1	0
		TXUR	TXRDY	TXEN		RXOR	RXRDY	RXEN
Access		R	R	R		R	R	R
Reset		0	0	0		0	0	0

Bits 21:20 – TXURCH[1:0] Transmit Underrun Channel

Value	Description
0	This field is cleared when I2SC_SCR.TXUR is written to ’1’.
1	Bit i of this field is set when a transmit underrun error occurred in channel i (i = 0 for first channel of the frame).

Bits 9:8 – RXORCH[1:0] Receive Overrun Channel
This field is cleared when I2SC_SCR.RXOR is written to ’1’.

Bit i of this field is set when a receive overrun error occurred in channel i (i = 0 for first channel of the frame).

Bit 6 – TXUR Transmit Underrun

Value	Description
0	This bit is cleared when the corresponding bit in I2SC_SCR is written to ’1’.
1	This bit is set when an underrun error occurs on I2SC_THR or when the corresponding bit in I2SC_SSR is written to ’1’.

Bit 5 – TXRDY Transmit Ready

Value (see Note)	RxStatus	Description
		channel format or was out of sequence. Only allowed on control and asynchronous channels.
74h... 7Eh	rsvd	Reserved
System Responses (Rx Device response in System Channel):		
00h	DeviceNotPresent	
80h	DevicePresent	
82h	DeviceServiceRequest	Device response to DeviceAddress scan (MLBScan), where the scanned Device needs some or all its ChannelAddresses configured.
84h...FEh	rsvd	Reserved

Note: All odd values (LSB set) are reserved.

48.6.1.5 System Commands

The Controller sends out System commands in the physical channel associated with the FRAMESYNC MediaLB frame alignment ChannelAddress (PC0). The NoData command indicates no command exists on the System Channel for this frame. All System commands are optional and may or may not be implemented on the MediaLB Controller. Additionally, System responses (including dynamic configuration) are optional and may or may not be implemented on a specific MediaLB Device.

The MOSTLock and MOSTUnlock commands indicate the status of the Controller relative to the MOST Network. When the Controller is not locked to the MOST Network (MOSTUnlock), all MediaLB data being transferred to or from the MOST Network must also stop. Buffers in the Controller could delay the stopping point to beyond when MOSTUnlock shows up on MediaLB.

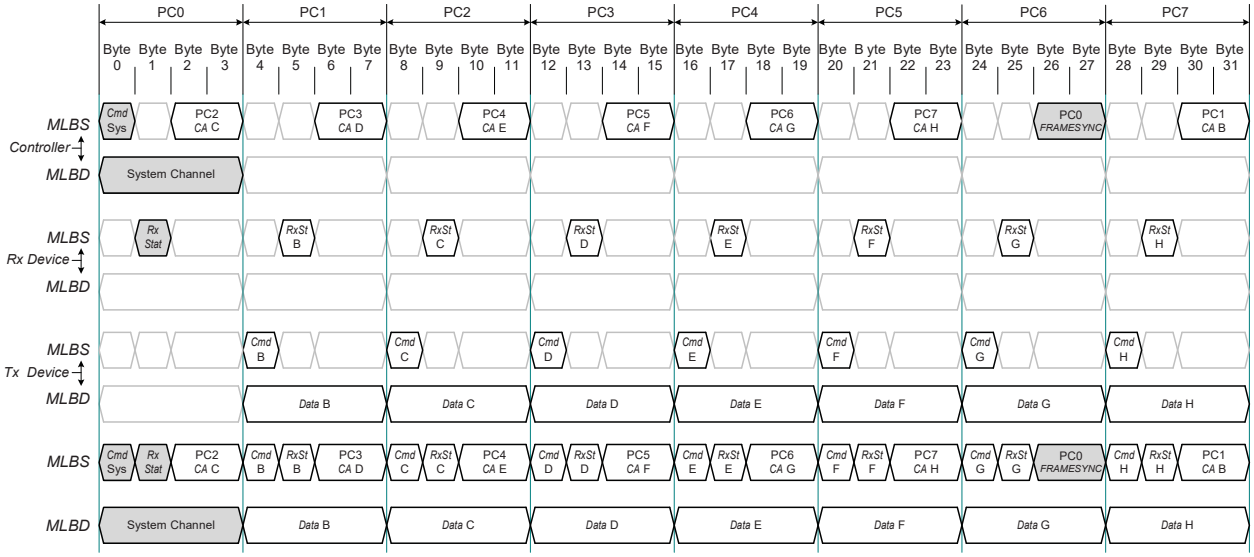
The MLBReset command is designed to place the MediaLB interface in one or all Devices in a known state. When a MediaLB Device receives the MLBReset command, it will look at the corresponding first two received (most significant) data bytes on the MLBD line:

- If the first two bytes are zero, then all MediaLB Devices must reset their MediaLB interface to an initialized known state (broadcast reset to all Devices).
- If the first two bytes match the local DeviceAddress, then only the Device with the matching DeviceAddress will reset its MediaLB interface to an initialized known state (reset targeted to only one Device).

The MLBSubCmd command is used for configuration and status information from the Controller to Devices. A sub-command is contained in the first byte of the MLBD quadlet. When MediaLB Device interfaces receive the MLBSubCmd command, they will store the command and corresponding data quadlet (sub-command). Currently, only one sub-command is defined (scSetCA) and is used in dynamic configuration.

MediaLB Devices and ChannelAddresses can be configured using two methods: static or dynamic. When the EHC MediaLB Device uses the dynamic method, it instructs the Controller to scan for other MediaLB

Figure 48-5. 3-pin MediaLB 256Fs Interface Example



48.6.1.7 Initialization

At power up, the MediaLB Controller might output a MLBReset command in the System Channel (all System commands are optional). Upon reception of the MLBReset command, all MediaLB Devices will cancel any current transmissions or receptions and clear their buffers.

Two scenarios are supported to configure MediaLB Devices and ChannelAddresses:

- Static pre-configured before startup. The system implementor decides which ChannelAddresses are to be used for every communication path on MediaLB. This static MediaLB configuration can be communicated by the EHC to the Controller through pre-defined power-up logical channels or through a secondary port.
- Dynamically at run-time. Dynamic configuration allows the board designer to support multiple build options where the EHC can query to find out if a particular Device is present or not on a particular board. The EHC instructs the Controller to scan for a particular DeviceAddress in the System Channel. The Controller uses the MLBScan command to look for a Device. The Controller then notifies the EHC whether the Device is present or not. If the Device is present, then the EHC can instruct the Controller to set the ChannelAddresses for the Device found. The EHC sends messages to the Controller to set each Indices/Logical channel, and waits the appropriate amount of time between each message as specified in the Devices documentation. When that particular Device is configured, the EHC can instruct the Controller to scan for the next Device.

Since the MediaLB Controller is the interface between the MediaLB Devices and the MOST Network, the Controller provides the MLBC signal and will also continue to operate even when the MOST Network is unlocked. When no activity exists on MediaLB, the Controller can shut off the MLBC placing MediaLB in a low-power state. The ChannelAddress assignments are not affected in low-power state; therefore, the same communication paths exists once MLBC is restarted.

MediaLB Devices are synchronously slaved to the MediaLB Controller through the MLBC signal. Since the Controller is synchronized to the MOST Network, the MLBC signal provides Network synchronization to all MediaLB Devices. Once the Controller starts up MLBC, all MediaLB Devices must synchronize to the MediaLB frame before communication can commence. When not frame-locked, Devices must search for the FRAMESYNC pattern, which defines a byte and physical channel boundary. Additionally, the start of the MediaLB frame (PC0) occurs one quadlet after FRAMESYNC is present on the bus. Even when a

49.6.38 MCAN Transmit Buffer Request Pending

Name: MCAN_TXBRP
Offset: 0xCC
Reset: 0x00000000
Property: Read-only

MCAN_TXBRP bits which are set while a Tx scan is in progress are not considered during this particular Tx scan. In case a cancellation is requested for such a Tx Buffer, this Add Request is cancelled immediately, the corresponding MCAN_TXBRP bit is reset.

Bit	31	30	29	28	27	26	25	24
	TRP31	TRP30	TRP29	TRP28	TRP27	TRP26	TRP25	TRP24
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	TRP23	TRP22	TRP21	TRP20	TRP19	TRP18	TRP17	TRP16
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
	TRP15	TRP14	TRP13	TRP12	TRP11	TRP10	TRP9	TRP8
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
	TRP7	TRP6	TRP5	TRP4	TRP3	TRP2	TRP1	TRP0
Access	R	R	R	R	R	R	R	R
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 – TRPx Transmission Request Pending for Buffer x
Each Tx Buffer has its own Transmission Request Pending bit. The bits are set via register MCAN_TXBAR. The bits are reset after a requested transmission has completed or has been cancelled via register MCAN_TXBCR.

TXBRP bits are set only for those Tx Buffers configured via MCAN_TXBC. After a MCAN_TXBRP bit has been set, a Tx scan (see Tx Handling) is started to check for the pending Tx request with the highest priority (Tx Buffer with lowest Message ID).

A cancellation request resets the corresponding transmission request pending bit of register MCAN_TXBRP. In case a transmission has already been started when a cancellation is requested, this is done at the end of the transmission, regardless whether the transmission was successful or not. The cancellation request bits are reset directly after the corresponding TXBRP bit has been reset.

After a cancellation has been requested, a finished cancellation is signalled via MCAN_TXBCF.

- after successful transmission together with the corresponding MCAN_TXBTO bit.
- when the transmission has not yet been started at the point of cancellation.

50.7.3 TC Channel Mode Register: Waveform Mode

Name: TC_CMRx
Offset: 0x04 + x*0x40 [x=0..2]
Reset: 0x00000000
Property: Read/Write

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

Bit	31	30	29	28	27	26	25	24
	BSWTRG[1:0]		BEEVT[1:0]		BCPC[1:0]		BCPB[1:0]	
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	23	22	21	20	19	18	17	16
	ASWTRG[1:0]		AEEVT[1:0]		ACPC[1:0]		ACPA[1:0]	
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	15	14	13	12	11	10	9	8
	WAVE	WAVSEL[1:0]		ENETRG	EEVT[1:0]		EEVTEDG[1:0]	
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
	CPCDIS	CPCSTOP	BURST[1:0]		CLKI	TCCLKS[2:0]		
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 31:30 – BSWTRG[1:0] Software Trigger Effect on TIOBx

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

Bits 29:28 – BEEVT[1:0] External Event Effect on TIOBx

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

Bits 27:26 – BCPC[1:0] RC Compare Effect on TIOBx

Value	Name	Description
0	NONE	None
1	SET	Set

- An output override block that can force the two complementary outputs to a programmed value (OOOHx/OOOLx).
- An asynchronous fault protection mechanism that has the highest priority to override the two complementary outputs (PWMHx/PWMLx) in case of fault detection (outputs forced to ‘0’, ‘1’ or Hi-Z).

51.6.2.2 Comparator

The comparator continuously compares its counter value with the channel period defined by CPRD in the PWM Channel Period Register (PWM_CPRDx) and the duty-cycle defined by CDTY in the PWM Channel Duty Cycle Register (PWM_CDTYx) to generate an output signal OCx accordingly.

The different properties of the waveform of the output OCx are:

- the clock selection. The channel counter is clocked by one of the clocks provided by the clock generator described in the previous section. This channel parameter is defined in the CPRE field of the PWM Channel Mode Register (PWM_CMRx). This field is reset at ‘0’.
- the waveform period. This channel parameter is defined in the CPRD field of the PWM_CPRDx register.
  If the waveform is left-aligned, then the output waveform period depends on the counter source clock and can be calculated:
  By using the PWM peripheral clock divided by a given prescaler value “X” (where X = 2^{PREA} is 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, or 1024). The resulting period formula is:

  $$\frac{(X \times \mathrm{CPRD})}{f_{\mathrm{peripheral\ clock}}}$$

  By using the PWM peripheral clock divided by a given prescaler value “X” (see above) and by either the DIVA or the DIVB divider. The formula becomes, respectively:

  $$\frac{(X \times \mathrm{CPRD} \times \mathrm{DIVA})}{f_{\mathrm{peripheral\ clock}}} \text{ or } \frac{(X \times \mathrm{CPRD} \times \mathrm{DIVB})}{f_{\mathrm{peripheral\ clock}}}$$

  If the waveform is center-aligned, then the output waveform period depends on the counter source clock and can be calculated:
  By using the PWM peripheral clock divided by a given prescaler value “X” (where X = 2^{PREA} is 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, or 1024). The resulting period formula is:

  $$\frac{(2 \times X \times \mathrm{CPRD})}{f_{\mathrm{peripheral\ clock}}}$$

  By using the PWM peripheral clock divided by a given prescaler value “X” (see above) and by either the DIVA or the DIVB divider. The formula becomes, respectively:

  $$\frac{(2 \times X \times \mathrm{CPRD} \times \mathrm{DIVA})}{f_{\mathrm{peripheral\ clock}}} \text{ or}$$

  $$\frac{(2 \times X \times \mathrm{CPRD} \times \mathrm{DIVB})}{f_{\mathrm{peripheral\ clock}}}$$
- the waveform duty-cycle. This channel parameter is defined in the CDTY field of the PWM_CDTYx register.

51.7.5 PWM Interrupt Enable Register 1

Name: PWM_IER1
Offset: 0x10
Reset: –
Property: Write-only

This register can only be written if bits WPSWS1 and WPHWS1 are cleared in the PWM Write Protection Status Register.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
					FCHID3	FCHID2	FCHID1	FCHID0
Access					W	W	W	W
Reset					0	0	0	–

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
					CHID3	CHID2	CHID1	CHID0
Access					W	W	W	W
Reset					0	0	0	–

Bits 16, 17, 18, 19 – FCHIDx Fault Protection Trigger on Channel x Interrupt Enable

Bits 0, 1, 2, 3 – CHIDx Counter Event on Channel x Interrupt Enable

54. Analog Comparator Controller (ACC)

54.1 Description

The Analog Comparator Controller (ACC) configures the analog comparator and generates an interrupt depending on user settings. The analog comparator embeds two 8-to-1 multiplexers that generate two internal inputs. These inputs are compared, resulting in a compare output. The hysteresis level, edge detection and polarity are configurable.

The ACC also generates a compare event which can be used by the Pulse Width Modulator (PWM).

54.2 Embedded Characteristics

- ANA_INPUTS User Analog Inputs Selectable for Comparison
- VOLT_REF Voltage References Selectable for Comparison: External Voltage Reference, DAC0, DAC1, Temperature Sensor (TS)
- Interrupt Generation
- Compare Event Fault Generation for PWM

54.3 Block Diagram

Figure 54-1. Analog Comparator Controller Block Diagram

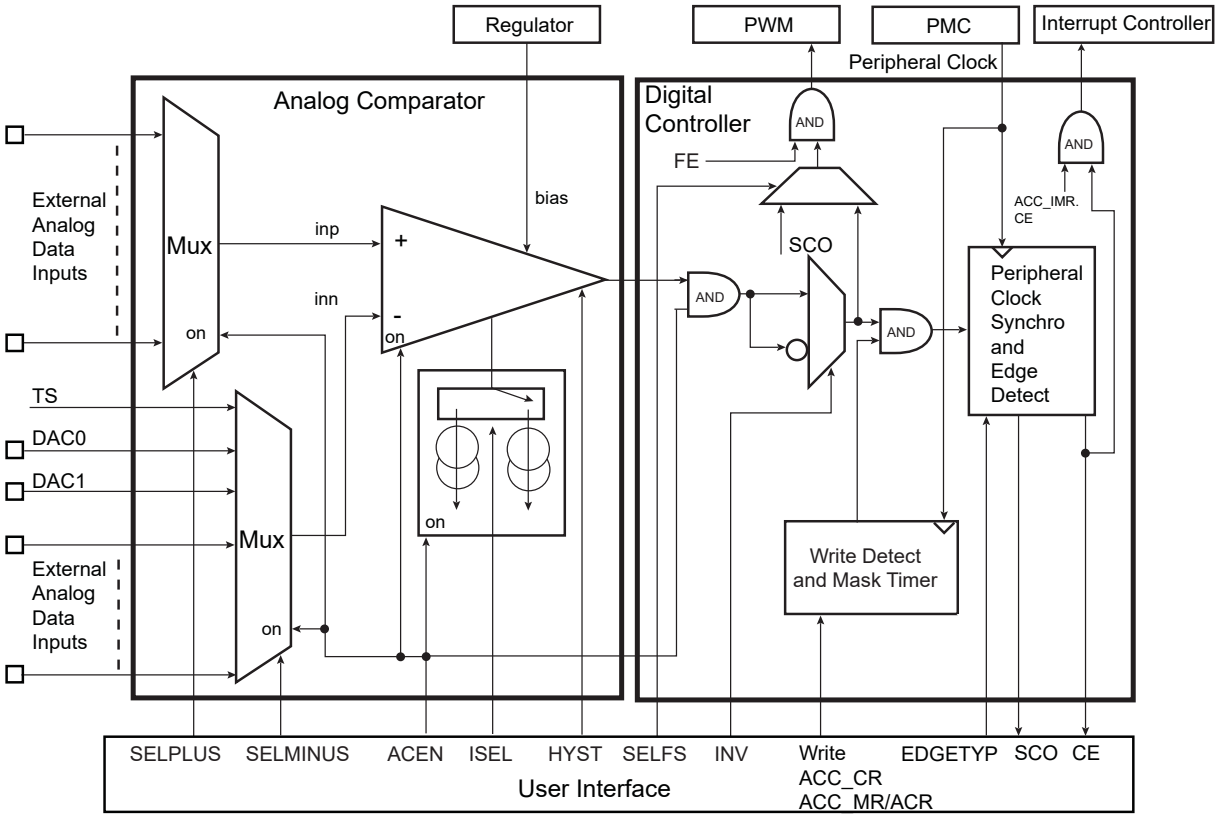


Table 59-55. QSPI Timings

Symbol	Parameter	Conditions	Min	Max	Unit
$QSPI_0$	QIOx data in to QSCK rising edge (input setup time)	3.3V domain	2.5	–	ns
		1.8V domain	2.9	–	ns
$QSPI_1$	QIOx data in to QSCK rising edge (input hold time)	3.3V domain	0	–	ns
		1.8V domain	0	–	ns
$QSPI_2$	QSCK rising edge to QIOx data out valid	3.3V domain	-1.3	1.9	ns
		1.8V domain	-2.5	3.0	ns
$QSPI_3$	QIOx data in to QSCK falling edge (input setup time)	3.3V domain	2.9	–	ns
		1.8V domain	3.2	–	ns
$QSPI_4$	QIOx data in to QSCK falling edge(input hold time)	3.3V domain	0	–	ns
		1.8V domain	0	–	ns
$QSPI_5$	QSCK falling edge to QIOx data out valid	3.3V domain	-1.6	1.8	ns
		1.8V domain	-2.7	3.1	ns

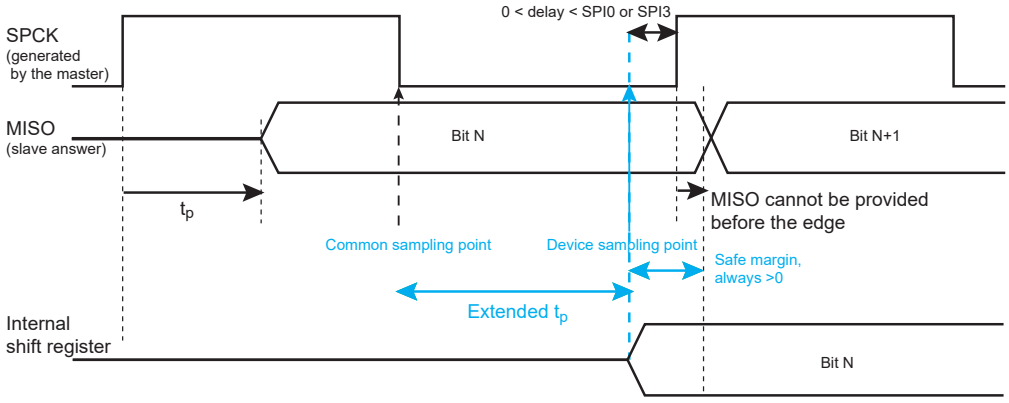
Timings are given for the 3.3V domain, with VDDIO from 2.85V to 3.6V, maximum external capacitor = 40 pF.

59.13.1.6 SPI Characteristics

In the figures below, the MOSI line shifting edge is represented with a hold time equal to 0. However, it is important to note that for this device, the MISO line is sampled prior to the MOSI line shifting edge. As shown further below, the device sampling point extends the propagation delay (t_p) for slave and routing delays to more than half the SPI clock period, whereas the common sampling point allows only less than half the SPI clock period.

As an example, an SPI Slave working in Mode 0 can be safely driven if the SPI Master is configured in Mode 0.

Figure 59-19. MISO Capture in Master Mode



Symbol	Parameter	Condition	Min	Max	Unit
SSC_4	TK edge to TF/TD (TK output, TF input)	–	-3.9[1]	3.0[1]	ns
		STTDLY = 0 START = 4, 5 or 7	-3.9 + (2× t_{CPMCK})[1]	3.0 + (2 × t_{CPMCK})[1]	
SSC_5	TF setup time before TK edge (TK input)	–	0	–	ns
SSC_6	TF hold time after TK edge (TK input)	–	t_{CPMCK}	–	ns
SSC_7	TK edge to TF/TD (TK input, TF input)	–	3.1[1]	11.8[1]	ns
		STTDLY = 0 START = 4, 5 or 7	3.1 + (3 × t_{CPMCK})[1]	11.8 + (3 × t_{CPMCK})[1]	
Receiver					
SSC_8	RF/RD setup time before RK edge (RK input)	–	0	–	ns
SSC_9	RF/RD hold time after RK edge (RK input)	–	t_{CPMCK}	–	ns
SSC_{10}	RK edge to RF (RK input)	–	2.9[1]	9.2[1]	ns
SSC_{11}	RF/RD setup time before RK edge (RK output)	–	10.1 - t_{CPMCK}	–	ns
SSC_{12}	RF/RD hold time after RK edge (RK output)	–	t_{CPMCK} - 2.8	–	ns
SSC_{13}	RK edge to RF (RK output)	–	-2.1[1]	1.9[1]	ns

Note: For output signals (TF, TD, RF), min and max access times are defined. The min access time is the time between the TK (or RK) edge and the signal change. The max access timing is the time between the TK edge and the signal stabilization. The information below illustrates min and max accesses for SSC0. The same applies for SSC1, SSC4, and SSC7, SSC10 and SSC13.

Table 59-75. SSC Timings with 1.8V Peripheral Supply

Symbol	Parameter	Condition	Min	Max	Unit
Transmitter					
SSC_0	TK edge to TF/TD (TK output, TF output)	–	-5.7[1]	5.3[1]	ns
SSC_1	TK edge to TF/TD (TK input, TF output)	–	3.6[1]	16.8[1]	ns
SSC_2	TF setup time before TK edge (TK output)	–	17.3	–	ns

Date	Changes
	Section 46. ”Universal Synchronous Asynchronous Receiver Transceiver (USART)” Section 46.4 ”I/O Lines Description”: removed mention of USART3 as fully equipped with modem signals. Updated Figure 46-27 “RTS Line Software Control when US_MR.USART_MODE = 2” Section 46.7.17 ”USART Channel Status Register”: updated RTSDIS description.
	Section 49. ”Controller Area Network (MCAN)” Section 49.1 ”Description”: updated information on compliance. Updated Table 49-2 “Peripheral IDs” .
	Section 50. ”Timer Counter (TC)” Section 50.6.16.2 ”Input Preprocessing”: removed unit following equation in 3rd paragraph. Added limitation on maximum pulse duration. Section 50.6.16.4 ”Position and Rotation Measurement”: in 3rd paragraph, added “The process must be started by configuring TC_CCR.CLKEN and TC_CCR.SWTRG.” “Detecting a Missing Index Pulse” now Section 50.6.16.6 (was Section 50.6.17). Corrected value of TC_RC0.RC in example in 2nd paragraph. Added Section 50.6.16.7 ”Detecting Contamination/Dust at Rotary Encoder Low Speed”. Section 50.7.16 ”TC Block Mode Register”: added AUTOC at index 18 and bit description. Added MAXCMP at index [29:26] and field description. Updated MAXFILT field description. Section 50.7.17 ”TC QDEC Interrupt Enable Register”, Section 50.7.17 ”TC QDEC Interrupt Enable Register”, Section 50.7.17 ”TC QDEC Interrupt Enable Register” and Section 50.7.17 ”TC QDEC Interrupt Enable Register”: added bit MPE at index 3 and bit description
	Section 51. ”Pulse Width Modulation Controller (PWM)” Throughout, “PWMTRG” and “EXTTRG” renamed to “PWMEXTRG”. Updated Figure 51-1 “Pulse Width Modulation Controller Block Diagram”. Updated section “Recoverable Fault”. Updated Figure 51-16 “Fault Protection”. Section 51.6.7 ”Register Write Protection”: added PWM_IER1, PWM_IDR1, PWM_IER2 and PWM_IDR2 to list of write-protected registers in Register group 1. Section 51-8 ”Register Mapping”: modified offsets for “PWM External Trigger Register 1”, “PWM Leading-Edge Blanking Register 1”, “PWM External Trigger Register 2” and “PWM Leading-Edge Blanking Register 2”. Section 51.7.5 ”PWM Interrupt Enable Register 1”, Section 51.7.6 ”PWM Interrupt Disable Register 1”, Section 51.7.14 ”PWM Interrupt Enable Register 2”, Section 51.7.15 ”PWM Interrupt Disable Register 2”: below each register table, added “This register can only be written if bits WPSWS1 and WPHWS1 are cleared in the PWM Write Protection Status Register.”
12-Oct-16	Section 52. ”Analog Front-End Controller (AFEC)” Section 52.5.7 ”Fault Output”: updated section with details on AFEC_TEMPMR and

Date	Comments
	Table 50.6.3 “Clock Selection”: updated notes (1) and (2). Updated Section 50.6.16.4 “Position and Rotation Measurement”. Added Section 50.6.17 “Detecting a Missing Index Pulse”.
cont’d	
01-June-16	Section 52. “Analog Front-End Controller (AFEC)” Section 52.2 “Embedded Characteristics”: deleted bullet on conversion rate (redundant with Electrical Characteristics) Section 52.6.1 “Analog Front-End Conversion”: updated and changed clock frequency range. Updated formula to calculate AFE conversion time. Section 52.6.3 “Conversion Resolution”: added Note. Section 52.6.6 “Conversion Triggers”: added detail on effects of delay variation. Section 52.6.11 “Input Gain and Offset”: updated information on AOFF field. Section 52.6.12 “AFE Timings”: updated Warning and deleted paragraph on settling time. Section 52.6.15 “Automatic Error Correction”: - modified the description of Gs and value (now 15, was 11). - modified the formula given to obtain the final conversion result after error correction. - added details on OFFSETCORR and GAINCORR fields. - deleted definitions of unused terms ‘ConvValue’ and ‘Resolution’ - added Figure 7-14 “AFE Digital Signal Processing”. Section 52.7.2 “AFEC Mode Register”: updated descriptions of fieldsTRACKTIM and TRANSFER. Section 52.7.18 “AFEC Channel Selection Register”: updated CSEL bit description. Section 52.7.20 “AFEC Channel Offset Compensation Register”: added note on configuration of AOFF.
	Section 58. “Electrical Characteristics” Added Table 58-2 “Recommended Thermal Operating Conditions”. Updated Table 58-3 “DC Characteristics”. Updated Table 58-31 “AFE Timing Characteristics”. Modified AFEC_ACR.IBCTL value in Note (1) of Table 58-31 “AFE Timing Characteristics” and in Section 58.8.1.2 “ADC Bias Current”. Table 58-38 “Number of Tau:n”: deleted bullets on calculated tracking time. Updated Table 58-41 “Temperature Sensor Characteristics”. Table 58-52 “I/O Characteristics”: updated VDDIO for FreqMax1. Section 58.13.1.5 “QSPI Characteristics”: updated comments in “Master Read Mode”