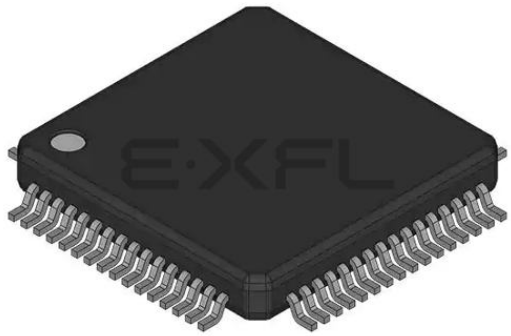


Welcome to E-XFL.COM

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	S12Z
Core Size	16-Bit
Speed	50MHz
Connectivity	CANbus, LINbus, SCI, SPI
Peripherals	DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3.5V ~ 40V
Data Converters	A/D 9x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 150°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP Exposed Pad
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=s912zvml32f1wkh

Chapter 14
Programmable Trigger Unit (PTUV3)

Chapter 20
Flash Module (S12ZFTMRZ)

  - — Water pump
  - — Oil pump
  - — A/C compressor
  - — HVAC blower
  - — Engine cooling fan
  - — Electric vehicle battery cooling fan
- Brush DC motor control requiring driving in 2 directions, along with PWM control for
  - — Reversible wiper
  - — Trunk opener

1.2 Features

This section describes the key features of the MC9S12ZVM-Family. It documents the superset of features within the family. Some module versions differ from one part to another within the family. Section 1.2.1 MC9S12ZVM-Family Member Comparison provides information to help access the correct information for a particular part within the family.

1.2.1 MC9S12ZVM-Family Member Comparison

Table 1-2 provides a summary of feature set differences within the MC9S12ZVM-Family.

Table 1-2. S12ZVM Family Feature Set Differences

Feature	ZVMC256	ZVML128	ZVMC128	ZVML64	ZVMC64	ZVML32	ZVML31	ZVML31	ZVM32	ZVM32	ZVM16	ZVM16
Flash	256 KB	128 KB	128 KB	64 KB	64 KB	32 KB	32 KB	32 KB	32 KB	32 KB	16 KB	16 KB
EEPROM	1 KB	512 Bytes	512 Bytes	512 Bytes	512 Bytes	512 Bytes	128 Bytes	128 Bytes	128 Bytes	128 Bytes	128 Bytes	128 Bytes
RAM	32 KB	8 KB	8 KB	4 KB	4 KB	4 KB	4 KB	4 KB	4 KB	4 KB	2 KB	2 KB
Package	80 pin	64 pin	64 pin	64 pin	64 pin	64 pin	64 pin	48 pin	64 pin	48 pin	64 pin	48 pin
LINPHY	–	1	–	1	–	1	1	1	–	–	–	–
HVPHY	–	–	–	–	–	–	–	–	1	1	1	1
SCI	2	2	2	2	2	2	2	2	2	2	2	2
SPI	1	1	1	1	1	1	1	0	1	0	1	0
ADC channels	8+8	4+5	4+5	4+5	4+5	4+5	4+5	1+3	4+5	1+3	4+5	1+3
PMF channels	6	6	6	6	6	6	6	6	6	6	6	6
TIM channels	4 TIM0 + 2 TIM1	4	4	4	4	4	4	3	4	3	4	3

The exposed pad on the package bottom must be connected to a grounded contact pad on the PCB.

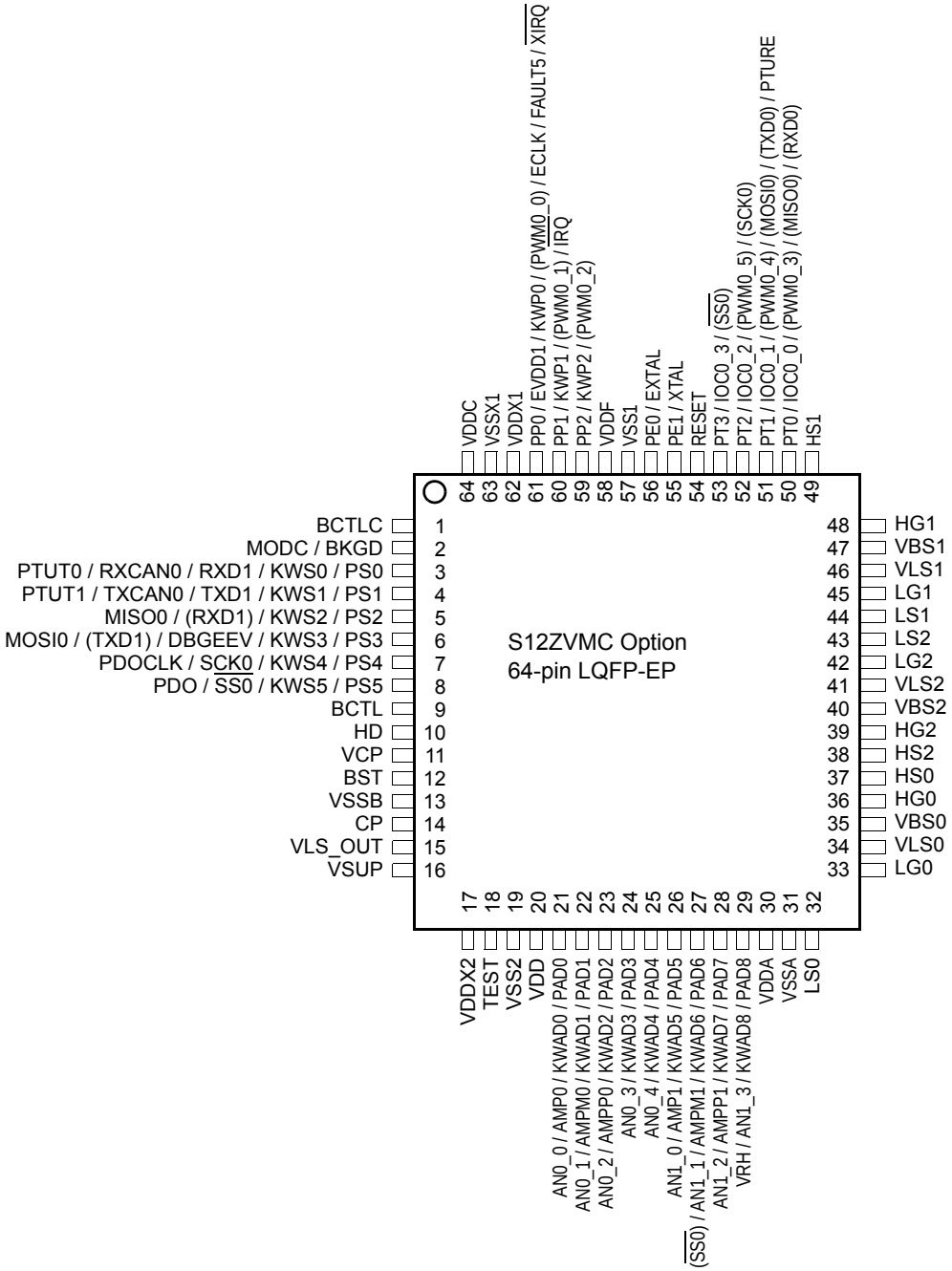
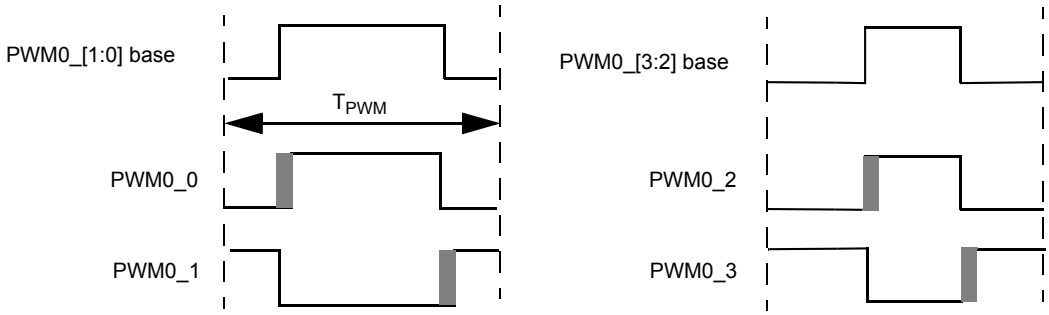


Figure 1-5. S12ZVMC Option 64-pin LQFP pin out

Table 1-9. Pin Summary For 80-Pin Package Option (ZVMC256 Only) (Sheet 4 of 5)

Pin #	Pin Name	Function (Priority and routing options defined in PIM chapter)							Supply	Internal Pull Resistor	
		1st Func.	2nd Func.	3rd Func.	4th Func.	5th Func.	6th Func.	7th Func.		CTRL	Reset State
48	HS0	—	—	—	—	—	—	—	—	—	—
49	HS2	—	—	—	—	—	—	—	—	—	—
50	HG2	—	—	—	—	—	—	—	—	—	—
51	VBS2	—	—	—	—	—	—	—	—	—	—
52	VLS2	—	—	—	—	—	—	—	—	—	—
53	LG2	—	—	—	—	—	—	—	—	—	—
54	LS2	—	—	—	—	—	—	—	—	—	—
55	LS1	—	—	—	—	—	—	—	—	—	—
56	LG1	—	—	—	—	—	—	—	—	—	—
57	VLS1	—	—	—	—	—	—	—	—	—	—
58	VBS1	—	—	—	—	—	—	—	—	—	—
59	HG1	—	—	—	—	—	—	—	—	—	—
60	HS1	—	—	—	—	—	—	—	—	—	—
61	PT0	IOC0_0	PWM1_3	MISO0	RXD0	PWM0_5	—	—	V_{DDX}	PERT/PPST	Off
62	PT1	IOC0_1	PWM1_4	MOSI0	TXD0	—	—	—	V_{DDX}	PERT/PPST	Off
63	PT2	IOC0_2	PWM1_0	SCK0	PWM0_7	—	—	—	V_{DDX}	PERT/PPST	Off
64	PT3	IOC0_3	PWM1_2	SS0	PWM0_3	—	—	—	V_{DDX}	PERT/PPST	Off
65	RESET	—	—	—	—	—	—	—	V_{DDX}	TEST pin	Up
66	PE1	XTAL	—	—	—	—	—	—	V_{DDX}	PERE/PPSE	Down
67	PE0	EXTAL	—	—	—	—	—	—	V_{DDX}	PERE/PPSE	Down
68	TEST	—	—	—	—	—	—	—	—	RESET	Down
69	PS3	KWS3	TXD1	MOSI0	CPTXD0	DBGEEV	IOC1_1	—	V_{DDX}	PERS/PPSS	Up
70	PS2	KWS2	RXD1	MISO0	CPRXD0	IOC1_0	—	—	V_{DDX}	PERS/PPSS	Up

Figure 1-13. BDCM Complementary Mode Waveform



Assuming first quadrant operation, forward accelerating operation, the applied voltage at node A must exceed the applied voltage at node B (Figure 1-11). Thus the PWM0_0 duty cycle must exceed the PWM0_2 duty cycle.

The duty cycle of PWM0_0 defines the voltage at the first power stage branch.

The duty cycle of PWM0_2 defines the voltage at the second power stage branch.

Modulating the duty cycle every period using the function F_{PWM} then the duty cycle is expressed as:

PWM0_0 duty-cycle = 0.5 + (0.5 * F_{PWM}); For -1<=F_{PWM} <= 1;

PWM0_2 duty-cycle = 0.5 - (0.5 * F_{PWM})

- Selectable drive strength for high current capable outputs

2.2 External Signal Description

This section lists and describes the signals that do connect off-chip.

Table 2-2 to Table 2-8 show all pins with the pins and functions that are controlled by the PIM. Routing options are denoted in parenthesis.

NOTE

If there is more than one function associated with a pin, the output priority is indicated by the position in the table from top (highest priority) to bottom (lowest priority).Inputs do not arbitrate priority unless noted differently in Table 2-40.

Table 2-2. BKGD Pin Functions and Priorities

Port	Pin Name	ZVMC256	ZVMC128\64	ZVML128/64/32	ZVML31	ZVM32/16	Pin Function & Priority	I/O	Description	Routing Register Bit	Pin Function after Reset
-	BKGD	✔	✔	✔	✔	✔	MODC[1]	I	MODC input during $\overline{\text{RESET}}$	—	BKGD
		✔	✔	✔	✔	✔	BKGD	I/O	S12ZBDC communication	—	

1. Function active when $\overline{\text{RESET}}$ asserted.

Table 2-3. Port E Pin Functions and Priorities

Port	Pin Name	ZVMC256	ZVMC128\64	ZVML128/64/32	ZVML31	ZVM32/16	Pin Function & Priority	I/O	Description	Routing Register Bit	Pin Function after Reset
E	PE1	✔	✔	✔	✔	✔	XTAL	-	CPMU OSC signal	—	GPIO
		✔	✔	✔	✔	✔	PTE[1]	I/O	General-purpose	—	
	PE0	✔	✔	✔	✔	✔	EXTAL	-	CPMU OSC signal	—	
		✔	✔	✔	✔	✔	PTE[0]	I/O	General-purpose	—	

2.3 Memory Map and Register Definition

This section provides a detailed description of all port integration module registers.

Subsection 2.3.1 shows all registers and bits at their related addresses within the global SoC register map. A detailed description of every register bit is given in subsection 2.3.2 to 2.3.4.

3.1.1 Glossary

Table 3-2. Glossary Of Terms

Term	Definition
MCU	Microcontroller Unit
CPU	S12Z Central Processing Unit
BDC	S12Z Background Debug Controller
ADC	Analog-to-Digital Converter
PTU	Programmable Trigger Unit
unmapped address range	Address space that is not assigned to a memory
reserved address range	Address space that is reserved for future use cases
illegal access	Memory access, that is not supported or prohibited by the S12ZMMC, e.g. a data store to NVM
access violation	Either an illegal access or an uncorrectable ECC error
byte	8-bit data
word	16-bit data

3.1.2 Overview

The S12ZMMC provides access to on-chip memories and peripherals for the S12ZCPU, the S12ZBDC, the PTU, and the ADC. It arbitrates memory accesses and determines all of the MCU memory maps. Furthermore, the S12ZMMC is responsible for selecting the MCUs functional mode.

3.1.3 Features

- S12ZMMC mode operation control
- Memory mapping for S12ZCPU and S12ZBDC, PTU and ADCs
  - — Maps peripherals and memories into a 16 MByte address space for the S12ZCPU, the S12ZBDC, the PTU, and the ADCs
  - — Handles simultaneous accesses to different on-chip resources (NVM, RAM, and peripherals)
- Access violation detection and logging
  - — Triggers S12ZCPU machine exceptions upon detection of illegal memory accesses and uncorrectable ECC errors
  - — Logs the state of the S12ZCPU and the cause of the access error

The number of core clock cycles since the last entry equals the timestamp + 1. The core clock runs at twice the frequency of the bus clock. The timestamp of the first trace buffer entry is 0x0000. With timestamps enabled trace buffer entries are initiated in the following ways:

- according to the trace mode specification, for example COF PC addresses in Normal mode
- on a timestamp counter overflow
  If the timestamp counter reaches 0xFFFF then a trace buffer entry is made, with timestamp= 0xFFFF and the timestamp overflow bit TOVF is set.
- on a match of comparator D
  If STAMP and DSTAMP are set then comparator D is used for forcing trace buffer entries with timestamps. The state control register settings determine if comparator D is also used to trigger the state sequencer. Thus if the state control register configuration does not use comparator D, then it is used solely for the timestamp function. If comparator D initiates a timestamp then the CTI bit is set in the INFO byte. This can be used in Normal/Loop1 mode to indicate when a particular data access occurs relative to the PC flow. For example when the timing of an access may be unclear due to the use of indexes.

NOTE

If comparator D is configured to match a PC address then associated timestamps trigger a trace buffer entry during execution of the previous instruction. Thus the PC stored to the trace buffer is that of the previous instruction.The comparator must contain the PC address of the instruction's first opcode byte

Timestamps are disabled in Pure PC mode.

6.4.5.4 Reading Data from Trace Buffer

The data stored in the trace buffer can be read using either the background debug controller (BDC) module or the CPU provided the DBG module is not armed and is configured for tracing by TSOURCE. When the ARM bit is set the trace buffer is locked to prevent reading. The trace buffer can only be unlocked for reading by an aligned word write to DBGTB when the module is disarmed. The trace buffer can only be read through the DBGTB register using aligned word reads. Reading the trace buffer while the DBG module is armed, or trace buffer locked returns 0xEE and no shifting of the RAM pointer occurs. Any byte or misaligned reads return 0xEE and do not cause the trace buffer pointer to increment to the next trace buffer address.

Reading the trace buffer is prevented by internal hardware whilst profiling is active because the RAM pointer is used to indicate the next row to be transmitted. Thus attempted reads of DBGTB do not return valid data when the PTACT bit is set. To initialize the pointer and read profiling data, the PTACT bit must be cleared and remain cleared.

The trace buffer data is read out first-in first-out. By reading CNT in DBGCNT the number of valid 64-bit lines can be determined. DBGCNT does not decrement as data is read.

Whilst reading, an internal pointer is used to determine the next line to be read. After a tracing session, the pointer points to the oldest data entry, thus if no overflow has occurred, the pointer points to line0. The

7.2.2.5 ECC Debug Data (ECCDDH, ECCDDL)

Module Base + 0x000C — Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	DDATA[15:8]							
Reset	0	0	0	0	0	0	0	0

Module Base + 0x000D — Access: User read/write

	7	6	5	4	3	2	1	0
R W	DDATA[7:0]							
Reset	0	0	0	0	0	0	0	0

= Unimplemented

Figure 7-6. ECC Debug Data (ECCDDH, ECCDDL)

1. Read: Anytime
   Write: Anytime

Table 7-6. ECCDD Register Field Descriptions

Field	Description
DDATA [23:0]	ECC Debug Raw Data — This register contains the raw data which will be written into the system memory during a debug write command or the read data from the debug read command.

7.2.2.6 ECC Debug ECC (ECCDE)

Module Base + 0x000E — Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	DECC[5:0]					
W								
Reset	0	0	0	0	0	0	0	0

1. Read: Anytime
   Write: Anytime

Figure 7-7. ECC Debug ECC (ECCDE)

Table 7-7. ECCDE Field Description

Field	Description
5:0 DECC[5:0]	ECC Debug ECC — This register contains the raw ECC value which will be written into the system memory during a debug write command or the ECC read value from the debug read command.

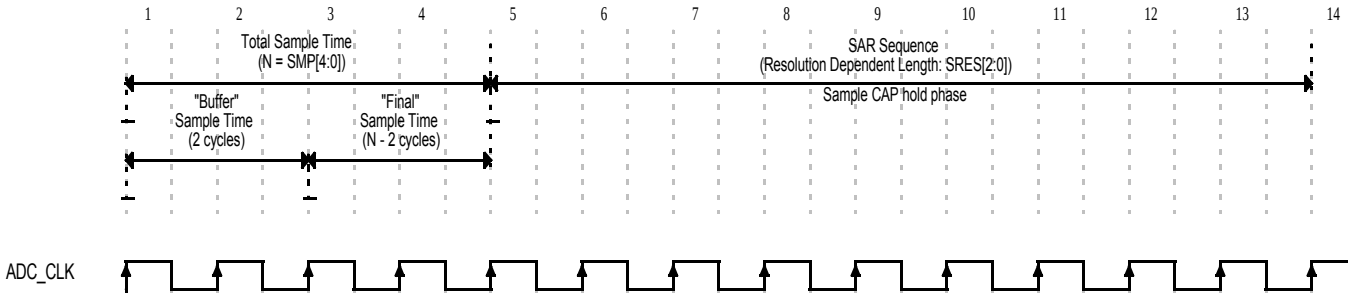


Figure 9-28. Sampling and Conversion Timing Example (8-bit Resolution, 4 Cycle Sampling)

Please note that there is always a pump phase of two ADC_CLK cycles before the sample phase begins, hence glitches during the pump phase could impact the conversion accuracy for short sample times.

9.6.3 Digital Sub-Block

The digital sub-block contains a list-based programmer's model and the control logic for the analog sub-block circuits.

9.6.3.1 Analog-to-Digital (A/D) Machine

The A/D machine performs the analog-to-digital conversion. The resolution is program selectable to be either 8- or 10- or 12 bits. The A/D machine uses a successive approximation architecture. It functions by comparing the sampled and stored analog voltage with a series of binary coded discrete voltages.

By following a binary search algorithm, the A/D machine identifies the discrete voltage that is nearest to the sampled and stored voltage.

Only analog input signals within the potential range of VRL_0/1 to VRH_0/1/3 (availability of VRL_1 and VRH_2 see Table 9-2) (A/D reference potentials) will result in a non-railed digital output code.

9.6.3.2 Introduction of the Programmer's Model

The ADC_LBA provides a programmer's model that uses a system memory list-based architecture for definition of the conversion command sequence and conversion result handling.
The Command Sequence List (CSL) and Result Value List (RVL) are implemented in double buffered manner and the buffer mode is user selectable for each list (bits CSL_BMOD, RVL_BMOD). The 32-bit wide conversion command is double buffered and the currently active command is visible in the ADC register map at ADCCMD register space.

– *When finished:*
This bit is cleared when the first conversion command of the sequence from top of active Sequence Command List is loaded

– *Mandatory Requirement:*
- In all ADC conversion flow control modes a Restart Event causes bit RSTA to be set. Bit SEQA is set simultaneously by ADC hardware if:
* ADC not idle (a conversion or conversion sequence is ongoing and current CSL not finished) and no Sequence Abort Event in progress (bit SEQA not already set or set simultaneously via internal interface or data bus)
* ADC idle but RVL done condition not reached
The RVL done condition is reached by one of the following:
* A “End Of List” command type has been executed
* A Sequence Abort Event is in progress or has been executed (bit SEQA already set or set simultaneously via internal interface or data bus)
The ADC executes the Sequence Abort Event followed by the Restart Event for the conditions described before or only a Restart Event.
- In ADC conversion flow control mode “Trigger Mode” a Restart Event causes bit TRIG being set automatically. Bit TRIG is set when no conversion or conversion sequence is ongoing (ADC idle) and the RVL done condition is reached by one of the following:
* A “End Of List” command type has been executed
* A Sequence Abort Event is in progress or has been executed
The ADC executes the Restart Event followed by the Trigger Event.
- In ADC conversion flow control mode “Trigger Mode” a Restart Event and a simultaneous Trigger Event via internal interface or data bus causes the TRIG_EIF bit being set and ADC cease operation.

- **Restart Event + CSL Exchange (Swap)**
Internal Interface Signals: Restart + LoadOK
Corresponding Bit Names: RSTA + LDOK

– *Function:*
Go to top of active CSL (clear index register for CSL) and switch to other offset register for address calculation if configured for double buffer mode (exchange the CSL list)
Requested by:
- Internal interface with the assertion of Interface Signal Restart the interface Signal LoadOK is evaluated and bit LDOK is set accordingly (bit LDOK set if Interface Signal LoadOK asserted when Interface Signal Restart asserts).
- Write Access via data bus to set control bit RSTA simultaneously with bit LDOK.

– *When finished:*
Bit LDOK can only be cleared if it was set as described before and both bits (LDOK, RSTA) are cleared when the first conversion command from top of active Sequence Command List is loaded

– *Mandatory Requirement:*
No ongoing conversion or conversion sequence
Details if using the internal interface:

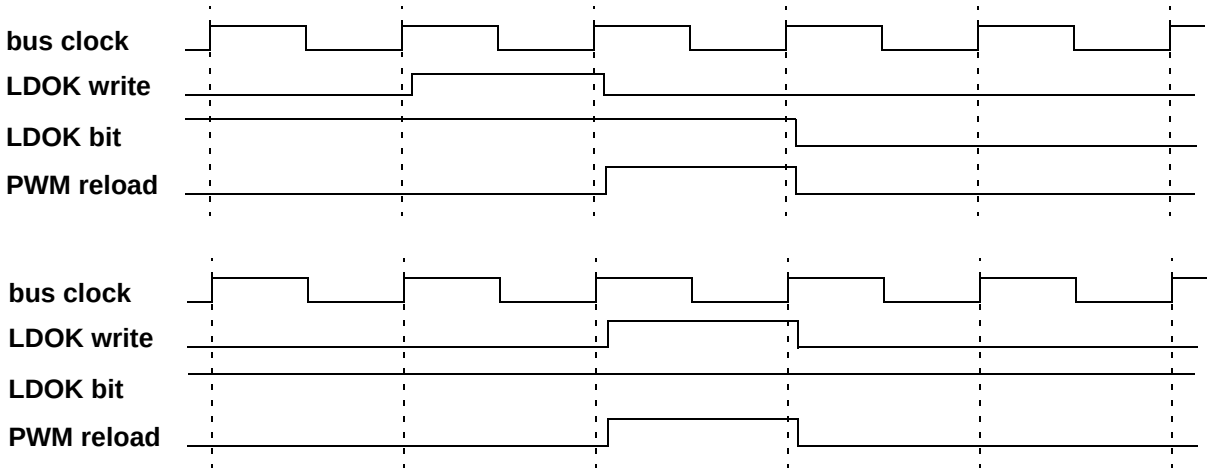


Figure 15-72. Setting asserted LDOK bit at PWM reload event

15.4.12.2 Global Load Enable

If a global load enable bit GLDOKA, B, or C is set, the global load OK bit defined on device level as input to the PMF replaces the function of the related local LDOKA, B, or C bits. The global load OK signal is typically shared between multiple IP blocks with the same double buffer scheme. Software handling must be transferred to the global load OK bit at the chip level.

15.4.12.3 Load Frequency

The LDFQ3, LDFQ2, LDFQ1, and LDFQ0 bits in the PWM control register (PMFFQCx) select an integral loading frequency of 1 to 16-PWM reload opportunities. The LDFQ bits take effect at every PWM reload opportunity, regardless the state of the related load okay bit or global load OK. The *half* bit in the PMFFQC register controls half-cycle reloads for center-aligned PWMs. If the *half* bit is set, a reload opportunity occurs at the beginning of every PWM cycle and half cycle when the count equals the modulus. If the half bit is not set, a reload opportunity occurs only at the beginning of every cycle. Reload opportunities can only occur at the beginning of a PWM cycle in edge-aligned mode.

NOTE

Setting the half bit takes effect immediately. Depending on whether the counter is incrementing or decrementing at this point in time, reloads at even-numbered reload frequencies (every 2, 4, 6,... reload opportunities) will occur only when the counter matches the modulus or only when the counter equals zero, respectively (refer to example of reloading at every two opportunities in Figure 15-74).

NOTE

Loading a new modulus on a half cycle will force the count to the new modulus value minus one on the next clock cycle. Half cycle reloads are possible only in center-aligned mode. Enabling or disabling half-cycle reloads in edge-aligned mode will have no effect on the reload rate.

Table 15-46. Effects of OUTCTL and OUT Bits on PWM Output Pair in Complementary Mode

OUTCTL (odd,even)	OUT (odd,even)	PWM (odd)	PWM (even)
00	xx	PWMgen(even)	PWMgen(even)
11	10	$\overline{\text{OUTB(even)}}$=1	OUTB(even)=0
01	x0	0	OUTB(even)=0

The recommended setup is:

```
PMFCFG0[INDEPC,INDEPB,INDEPA] = 0x0;   // Complementary mode
PMFCFG1[ENCE]                 = 1;     // Enable commutation event
PMFOUTB = 0x2A;                        // Set return path pattern, high-side off, low-side on
PMFOUTC = 0x1C;                        // Branch A->B, "mask" C // 0°
```

The commutation sequence is:

```
PMFOUTC = 0x34;                        // Branch A->C, "mask" B // 60°
PMFOUTC = 0x31;                        // Branch B->C, "mask" A // 120°
PMFOUTC = 0x13;                        // Branch B->A, "mask" C // 180°
PMFOUTC = 0x07;                        // Branch C->A, "mask" B // 240°
PMFOUTC = 0x0D;                        // Branch C->B, "mask" A // 300°
PMFOUTC = 0x1C;                        // Branch A->B, "mask" C // 360°
```

Table 15-47. Unipolar Switching Sequence

Branch	Channel	0°	60°	120°	180°	240°	300°
A	PWM0	PWMgen		0	0		0
	PWM1	PWMgen		0	1		0
B	PWM2	0	0	PWMgen		0	0
	PWM3	1	0	PWMgen		0	1
C	PWM4	0	0		0	PWMgen	
	PWM5	0	1		0	PWMgen	

15.8.2.2 Bipolar Switching Mode

Bipolar switching mode uses register bits MSK5-0 and PINVA, B, C to perform commutation.

The recommended setup is:

```
PMFCFG0[INDEPC,INDEPB,INDEPA] = 0x0;  // Complementary mode
PMFCFG1[ENCE]                 = 1;    // Enable commutation event
PMFCFG2[MSK5:MSK0]            = 0x30; // Branch A<->B, mask C // 0°
PMFCFG3[PINVC,PINVB,PINVA]    = 0x2;  // Invert B
```

The commutation sequence is:

```
PMFCFG2[MSK5:MSK0]            = 0x03; // Branch C<->B, mask A // 60°
PMFCFG3[PINVC,PINVB,PINVA]    = 0x2;  // Invert B
```

16.4.5 Transmitter

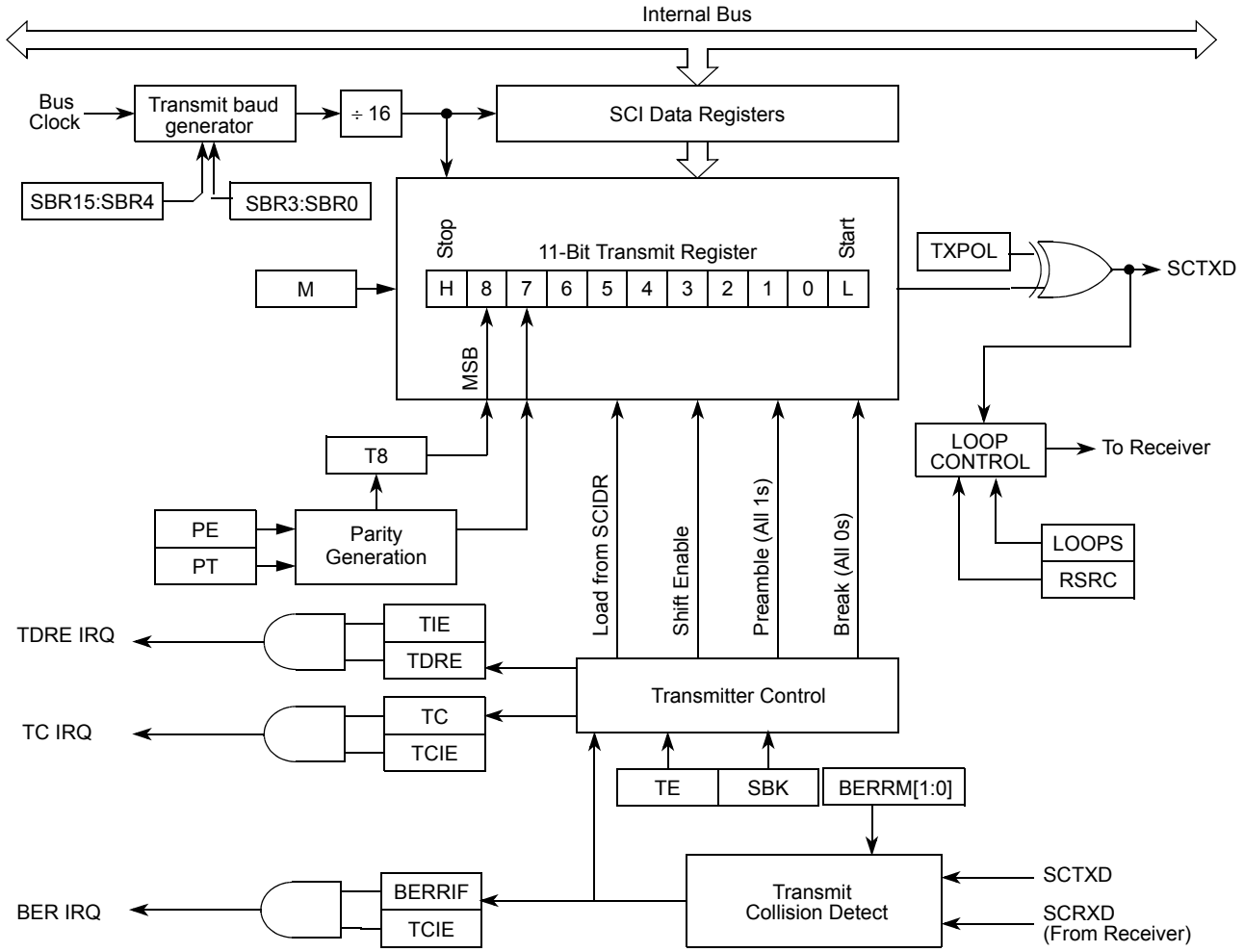


Figure 16-16. Transmitter Block Diagram

16.4.5.1 Transmitter Character Length

The SCI transmitter can accommodate either 8-bit or 9-bit data characters. The state of the M bit in SCI control register 1 (SCICR1) determines the length of data characters. When transmitting 9-bit data, bit T8 in SCI data register high (SCIDRH) is the ninth bit (bit 8).

16.4.5.2 Character Transmission

To transmit data, the MCU writes the data bits to the SCI data registers (SCIDRH/SCIDRL), which in turn are transferred to the transmitter shift register. The transmit shift register then shifts a frame out through the TXD pin, after it has prefaced them with a start bit and appended them with a stop bit. The SCI data registers (SCIDRH and SCIDRL) are the write-only buffers between the internal data bus and the transmit shift register.

19.3.2.6 LIN Status Register (LPSR)

Module Base + Address 0x0005 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	LPDT	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented

Figure 19-8. LIN Status Register (LPSR)

1. Read: Anytime

   Write: Never, writes to this register have no effect

Table 19-7. LPSR Field Description

Field	Description
7 LPDT	**LIN Transmitter TxD-dominant timeout Status Bit** — This read-only bit signals that the LPTxD pin is still dominant after a TxD-dominant timeout. As long as the LPTxD is dominant after the timeout the LIN transmitter is shut down and the LPTDIF is set again after attempting to clear it. 0 If there was a TxD-dominant timeout, LPTxD has ceased to be dominant after the timeout. 1 LPTxD is still dominant after a TxD-dominant timeout.

19.3.2.7 LIN Interrupt Enable Register (LPIE)

Module Base + Address 0x0006 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	LPDTIE	LPOCIE	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented

Figure 19-9. LIN Interrupt Enable Register (LPIE)

1. Read: Anytime

   Write: Anytime

Table 20-46. Erase All Blocks Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 000 at command launch
		Set if command not available in current mode (see Table 20-29)
	FPVIOL	Set if any area of the P-Flash or EEPROM memory is protected
	MGSTAT1	Set if any errors have been encountered during the verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation

20.4.7.7.1 Erase All Pin

The functionality of the Erase All Blocks command is also available in an uncommanded fashion from the *soc_erase_all_req* input pin on the Flash module. Refer to the Reference Manual for information on control of *soc_erase_all_req*.

The erase-all function requires the clock divider register FCLKDIV (see Section 20.3.2.1) to be loaded before invoking this function using *soc_erase_all_req* input pin. The FCLKDIV configuration for this feature is described at device level. If FCLKDIV is not properly set the erase-all operation will not execute and the ACCERR flag in FSTAT register will set. After the execution of the erase-all function the FCLKDIV register will be reset and the value of register FCLKDIV must be loaded before launching any other command afterwards.

Before invoking the erase-all function using the *soc_erase_all_req* pin, the ACCERR and FPVIOL flags in the FSTAT register must be clear. When invoked from *soc_erase_all_req* the erase-all function will erase all P-Flash memory and EEPROM memory space regardless of the protection settings. If the post-erase verify passes, the routine will then release security by setting the SEC field of the FSEC register to the unsecure state (see Section 20.3.2.2). The security byte in the Flash Configuration Field will be programmed to the unsecure state (see Table 20-9). The status of the erase-all request is reflected in the ERSAREQ bit in the FCNFG register (see Section 20.3.2.5). The ERSAREQ bit in FCNFG will be cleared once the operation has completed and the normal FSTAT error reporting will be available as described inTable 20-47.

At the end of the erase-all sequence Protection will remain configured as it was before executing the erase-all function. If the application requires programming P-Flash and/or EEPROM after the erase-all function completes, the existing protection limits must be taken into account. If protection needs to be disabled the user may need to reset the system right after completing the erase-all function.

Table 20-47. Erase All Pin Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if command not available in current mode (see Table 20-29)
	MGSTAT1	Set if any errors have been encountered during the erase verify operation, or during the program verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the erase verify operation, or during the program verify operation

- The channel is disabled

In this way, the output of the PWM will always be either the old duty waveform or the new duty waveform, not some variation in between. If the channel is not enabled, then writes to the duty register will go directly to the latches as well as the buffer.

NOTE

Reads of this register return the most recent value written. Reads do not necessarily return the value of the currently active duty due to the double buffering scheme.

See Section 22.4.2.3, “PWM Period and Duty” for more information.

NOTE

Depending on the polarity bit, the duty registers will contain the count of either the high time or the low time. If the polarity bit is one, the output starts high and then goes low when the duty count is reached, so the duty registers contain a count of the high time. If the polarity bit is zero, the output starts low and then goes high when the duty count is reached, so the duty registers contain a count of the low time.

To calculate the output duty cycle (high time as a% of period) for a particular channel:

- Polarity = 0 (PPOL x =0)

  Duty Cycle = [(PWMPERx-PWMDTYx)/PWMPERx] * 100%
- Polarity = 1 (PPOLx = 1)

  Duty Cycle = [PWMDTYx / PWMPERx] * 100%

For boundary case programming values, please refer to Section 22.4.2.8, “PWM Boundary Cases”.

Module Base + 0x001C = PWMDTY0, 0x001D = PWMDTY1, 0x001E = PWMDTY2, 0x001F = PWMDTY3
Module Base + 0x0020 = PWMDTY4, 0x0021 = PWMDTY5, 0x0022 = PWMDTY6, 0x0023 = PWMDTY7

	7	6	5	4	3	2	1	0
R W	Bit 7	6	5	4	3	2	1	Bit 0
Reset	1	1	1	1	1	1	1	1

Figure 22-14. PWM Channel Duty Registers (PWMDTYx)

[1] This register is available only when the corresponding channel exists and is reserved if that channel does not exist. Writes to a reserved register have no functional effect. Reads from a reserved register return zeroes.

Read: Anytime

Write: Anytime

NOTES:

1. DIMENSIONS ARE IN MILLIMETERS.
2. INTERPRET DIMENSIONS AND TOLERANCES PER ASME Y14.5M–1994.
3. DATUMS A, B AND D TO BE DETERMINED AT DATUM PLANE H.
4. DIMENSIONS TO BE DETERMINED AT SEATING PLANE C.
5. DIMENSION DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM DIMENSION BY MORE THAN 0.08 MM. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD OR PROTRUSION 0.07 MM.
6. DIMENSIONS DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 MM PER SIDE. DIMENSIONS ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.
7. EXACT SHAPE OF EACH CORNER IS OPTIONAL.
8. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 MM AND 0.25 MM FROM THE LEAD TIP.
9. HATCHED AREA TO BE KEEP OUT ZONE FOR PCB ROUTING.

TITLE: LQFP, 10 X 10 X 1.4 PKG, 0.5 PITCH, 64LD, 6.1 x 6.1 EXPOSED PAD	CASE NUMBER: 1899-03
	STANDARD: JEDEC MS-026 BCD
	SHEET: 4