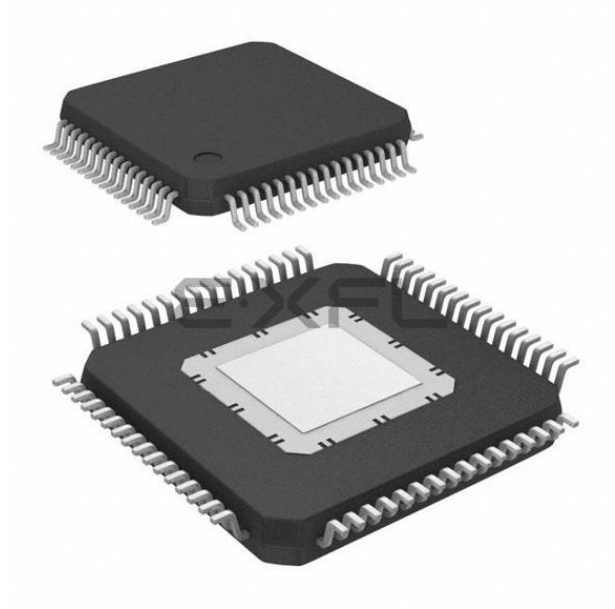

Welcome to E-XFL.COM

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	Obsolete
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 ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP Exposed Pad
Supplier Device Package	64-HLQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s912zvml32f2vkh

Chapter 15
Pulse Width Modulator with Fault Protection (PMF15B6CV4)

Chapter 16
Serial Communication Interface (S12SCIV6)

1.8.3 Device Level PMF Connectivity

Table 1-12. Mapping of PMF signals

PMF Connection	Usage
Channel0	High-Side Gate and Source Pins HG[0], HS[0]
Channel1	Low-Side Gate and Source Pins LG[0], LS[0]
Channel2	High-Side Gate and Source Pins HG[1], HS[1]
Channel3	Low-Side Gate and Source Pins LG[1], LS[1]
Channel4	High-Side Gate and Source Pins HG[2], HS[2]
Channel5	Low-Side Gate and Source Pins LG[2], LS[2]
FAULT5	External FAULT5 pin
FAULT4	HD Over voltage or GDU over current
FAULT3	VLS under voltage
FAULT2	GDU Desaturation[2] or GDU over current
FAULT1	GDU Desaturation[1] or GDU over current
FAULT0	GDU Desaturation[0] or GDU over current
IS2	GDU Phase Status[2]
IS1	GDU Phase Status[1]
IS0	GDU Phase Status[0]
async_event_edge_sel[1:0]	Tied to b11 (both edges active)

1.8.4 BDC Clock Source Connectivity

The BDC clock, BDCCLK, is mapped to the IRCCLK generated in the CPMU module.

The BDC clock, BDCFCLK is mapped to the device bus clock, generated in the CPMU module.

1.8.5 LINPHY Connectivity

The VLINSUP supply is device dependent.

On ZVML128, ZVMC128, ZVML64, ZVMC64 and ZVML32 devices with the maskset number 2N95G it is connected to VSUP

On all other devices it is connected to the device HD pin.

The LINPHY0 signals are mapped internally to SCI0. The receiver can be routed to TIM0 input capture channel3. These routing options are described in detail in the PIM section.

1.8.6 HVPHY Connectivity

The HVPHY signals (S12ZVM32 and S12ZVM16 derivatives only) are mapped internally to SCI0. The receiver can be routed to TIM0 input capture channel3.

4.4.5 Exception Priority

The priority (from highest to lowest) and address of all exception vectors issued by the INT module upon request by the CPU are shown in Table 4-8. Generally, all non-maskable interrupts have higher priorities than maskable interrupts. Please note that between the four software interrupts (Unimplemented op-code trap page1/page2 requests, SWI request, SYS request) there is no real priority defined since they cannot occur simultaneously (the S12Z CPU executes one instruction at a time).

Table 4-8. Exception Vector Map and Priority

Vector Address[1]	Source
0xFFFFFC	Pin reset, power-on reset, low-voltage reset, clock monitor reset, COP watchdog reset
(Vector base + 0x0001F8)	Unimplemented page1 op-code trap (SPARE) vector request
(Vector base + 0x0001F4)	Unimplemented page2 op-code trap (TRAP) vector request
(Vector base + 0x0001F0)	Software interrupt instruction (SWI) vector request
(Vector base + 0x0001EC)	System call interrupt instruction (SYS) vector request
(Vector base + 0x0001E8)	Machine exception vector request
(Vector base + 0x0001E4)	Reserved
(Vector base + 0x0001E0)	Reserved
(Vector base + 0x0001DC)	Spurious interrupt
(Vector base + 0x0001D8)	$\overline{\text{XIRQ}}$ interrupt request
(Vector base + 0x0001D4)	$\overline{\text{IRQ}}$ interrupt request
(Vector base + 0x000010 .. Vector base + 0x0001D0)	Device specific I-bit maskable interrupt sources (priority determined by the associated configuration registers, in descending order)

1. 24 bits vector address based

4.4.6 Interrupt Vector Table Layout

The interrupt vector table contains 128 entries, each 32 bits (4 bytes) wide. Each entry contains a 24-bit address (3 bytes) which is stored in the 3 low-significant bytes of the entry. The content of the most significant byte of a vector-table entry is ignored. Figure 4-13 illustrates the vector table entry format.

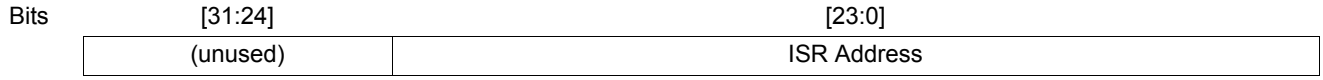

Figure 4-13. Interrupt Vector Table Entry

4.5 Initialization/Application Information

4.5.1 Initialization

After system reset, software should:

- Initialize the interrupt vector base register if the interrupt vector table is not located at the default location (0xFFFE00–0xFFFFFB).

Chapter 5
Background Debug Controller (S12ZBDCV2)

Table 5-1. Revision History

Revision Number	Revision Date	Sections Affected	Description of Changes
V2.04	03.Dec.2012	Section 5.1.3.3	Included BACKGROUND/ Stop mode dependency
V2.05	22.Jan.2013	Section 5.3.2.2	Improved NORESP description and added STEP1/ Wait mode dependency
V2.06	22.Mar.2013	Section 5.3.2.2	Improved NORESP description of STEP1/ Wait mode dependency
V2.07	11.Apr.2013	Section 5.1.3.3.1	Improved STOP and BACKGROUND interdepency description
V2.08	31.May.2013	Section 5.4.4.4 Section 5.4.7.1	Removed misleading WAIT and BACKGROUND interdepency description Added subsection dedicated to Long-ACK
V2.09	29.Aug.2013	Section 5.4.4.12	Noted that READ_DBGTB is only available for devices featuring a trace buffer.
V2.10	21.Oct.2013	Section 5.1.3.3.2	Improved description of NORESP dependence on WAIT and BACKROUND
V2.11	02.Feb.2015	Section 5.1.3.3.1 Section 5.3.2	Corrected name of clock that can stay active in Stop mode

5.1 Introduction

The background debug controller (BDC) is a single-wire, background debug system implemented in on-chip hardware for minimal CPU intervention. The device BKGD pin interfaces directly to the BDC.

The S12ZBDC maintains the standard S12 serial interface protocol but introduces an enhanced handshake protocol and enhanced BDC command set to support the linear instruction set family of S12Z devices and offer easier, more flexible internal resource access over the BDC serial interface.

5.1.1 Glossary

Table 5-2. Glossary Of Terms

Term	Definition
DBG	On chip Debug Module
BDM	Active Background Debug Mode
CPU	S12Z CPU
SSC	Special Single Chip Mode (device operating mode
NSC	Normal Single Chip Mode (device operating mode)
BDCSI	Background Debug Controller Serial Interface. This refers to the single pin BKGD serial interface.
EWAIT	Optional S12 feature which allows external devices to delay external accesses until deassertion of EWAIT

in bytes[6:4], the other payload bytes may be compressed or complete addresses as indicated by the info byte bits.

Table 6-57. Pure PC Mode Trace Buffer Format Single Source

Mode	8-Byte Wide Trace Buffer Line							
	7	6	5	4	3	2	1	0
CPU	CXINF	BASE	BASE	BASE	PLB3	PLB2	PLB1	PLB0

If the info bit for byte3 indicates a full CPU PC address, whereby bytes[5:3] are used, then the info bit mapped to byte[4] is redundant and the byte[6] is unused because a line overflow has occurred. Similarly a base address stored in bytes[4:2] causes line overflow, so bytes[6:5] are unused.

CXINF[6:4] indicate how many bytes in a line contain valid data, since tracing may terminate before a complete line has been filled.

CXINF Information Byte Source Tracing

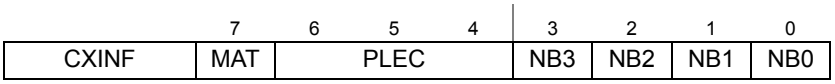

Figure 6-29. Pure PC Mode CXINF

Table 6-58. CXINF Field Descriptions

Field	Description
MAT	**Mid Aligned Trigger**— This bit indicates a mid aligned trigger position. When a mid aligned trigger occurs, the next trace buffer entry is a base address and the counter is incremented to a new line, independent of the number of bytes used on the current line. The MAT bit is set on the current line, to indicate the position of the trigger. When configured for begin or end aligned trigger, this bit has no meaning. NOTE: In the case when ARM and TRIG are simultaneously set together in the same cycle that a new PC value is registered, then this PC is stored to the same trace buffer line and MAT set. 0 Line filled without mid aligned trigger occurrence 1 Line last entry is the last PC entry before a mid aligned trigger
PLEC[2:0]	**Payload Entry Count**— This field indicates the number of valid bytes in the trace buffer line Binary encoding is used to indicate up to 7 valid bytes.
NBx	**Payload Compression Indicator**— This field indicates if the corresponding payload byte is the lowest byte of a base PC entry 0 Corresponding payload byte is a not the lowest byte of a base PC entry 1 Corresponding payload byte is the lowest byte of a base PC entry

Pure PC mode tracing does not support timestamps or external event entries.

6.4.5.3 Timestamp

When set, the STAMP bit in DBGTCRL configures the DBG to add a timestamp to trace buffer entries in Normal, Loop1 and Detail trace buffer modes. The timestamp is generated from a 16-bit counter and is stored to the trace buffer line each time a trace buffer entry is made.

8.2.16 VDD— Core Logic Supply Pin

VDD is the supply domain for the core logic.

An off-chip decoupling capacitor (220 nF(X7R ceramic)) between VDD and VSS is required and can improve the quality of this supply.

This supply domain is monitored by the Low Voltage Reset circuit and The Power On Reset circuit.

8.2.17 VDDF— NVM Logic Supply Pin

VDDF is the supply domain for the NVM logic.

An off-chip decoupling capacitor (220 nF(X7R ceramic)) between VDDF and VSS is required and can improve the quality of this supply.

This supply domain is monitored by the Low Voltage Reset circuit.

8.2.18 API_EXTCLK — API external clock output pin

This pin provides the signal selected via APIES and is enabled with APIEA bit. See the device specification if this clock output is available on this device and to which pin it might be connected.

8.2.19 TEMPSENSE — Internal Temperature Sensor Output Voltage

Depending on the VSEL setting either the voltage level generated by the temperature sensor or the VREG bandgap voltage is driven to a special channel input of the ADC Converter. See device level specification for connectivity of ADC special channels.

Table 8-13. RTI Frequency Divide Rates for RTDEC=1

RTR[3:0]	RTR[6:4] =							
	000 ($1x10^3$)	**001 ($2x10^3$)**	**010 ($5x10^3$)**	**011 ($10x10^3$)**	**100 ($20x10^3$)**	**101 ($50x10^3$)**	**110 ($100x10^3$)**	**111 ($200x10^3$)**
0000 (÷1)	$1x10^3$	$2x10^3$	$5x10^3$	$10x10^3$	$20x10^3$	$50x10^3$	$100x10^3$	$200x10^3$
0001 (÷2)	$2x10^3$	$4x10^3$	$10x10^3$	$20x10^3$	$40x10^3$	$100x10^3$	$200x10^3$	$400x10^3$
0010 (÷3)	$3x10^3$	$6x10^3$	$15x10^3$	$30x10^3$	$60x10^3$	$150x10^3$	$300x10^3$	$600x10^3$
0011 (÷4)	$4x10^3$	$8x10^3$	$20x10^3$	$40x10^3$	$80x10^3$	$200x10^3$	$400x10^3$	$800x10^3$
0100 (÷5)	$5x10^3$	$10x10^3$	$25x10^3$	$50x10^3$	$100x10^3$	$250x10^3$	$500x10^3$	$1x10^6$
0101 (÷6)	$6x10^3$	$12x10^3$	$30x10^3$	$60x10^3$	$120x10^3$	$300x10^3$	$600x10^3$	$1.2x10^6$
0110 (÷7)	$7x10^3$	$14x10^3$	$35x10^3$	$70x10^3$	$140x10^3$	$350x10^3$	$700x10^3$	$1.4x10^6$
0111 (÷8)	$8x10^3$	$16x10^3$	$40x10^3$	$80x10^3$	$160x10^3$	$400x10^3$	$800x10^3$	$1.6x10^6$
1000 (÷9)	$9x10^3$	$18x10^3$	$45x10^3$	$90x10^3$	$180x10^3$	$450x10^3$	$900x10^3$	$1.8x10^6$
1001 (÷10)	$10\ x10^3$	$20x10^3$	$50x10^3$	$100x10^3$	$200x10^3$	$500x10^3$	$1x10^6$	$2x10^6$
1010 (÷11)	$11\ x10^3$	$22x10^3$	$55x10^3$	$110x10^3$	$220x10^3$	$550x10^3$	$1.1x10^6$	$2.2x10^6$
1011 (÷12)	$12x10^3$	$24x10^3$	$60x10^3$	$120x10^3$	$240x10^3$	$600x10^3$	$1.2x10^6$	$2.4x10^6$
1100 (÷13)	$13x10^3$	$26x10^3$	$65x10^3$	$130x10^3$	$260x10^3$	$650x10^3$	$1.3x10^6$	$2.6x10^6$
1101 (÷14)	$14x10^3$	$28x10^3$	$70x10^3$	$140x10^3$	$280x10^3$	$700x10^3$	$1.4x10^6$	$2.8x10^6$
1110 (÷15)	$15x10^3$	$30x10^3$	$75x10^3$	$150x10^3$	$300x10^3$	$750x10^3$	$1.5x10^6$	$3x10^6$
1111 (÷16)	$16x10^3$	$32x10^3$	$80x10^3$	$160x10^3$	$320x10^3$	$800x10^3$	$1.6x10^6$	$3.2x10^6$

9.5.2.5 ADC Format Register (ADCFMT)

Module Base + 0x0004

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

= Unimplemented or Reserved

Figure 9-8. ADC Format Register (ADCFMT)

Read: Anytime

Write: Bits DJM and SRES[2:0] are writable if bit ADC_EN clear or bit SMOD_ACC set

Table 9-8. ADCFMT Field Descriptions

Field	Description
7 DJM	**Result Register Data Justification** — Conversion result data format is always unsigned. This bit controls justification of conversion result data in the conversion result list. 0 Left justified data in the conversion result list. 1 Right justified data in the conversion result list.
2-0 SRES[2:0]	**ADC Resolution Select** — These bits select the resolution of conversion results. See Table 9-9 for coding.

Table 9-9. Selectable Conversion Resolution

SRES[2]	SRES[1]	SRES[0]	ADC Resolution
0	0	0	8-bit data
0	0	1	Reserved[1.]
0	1	0	10-bit data
0	1	1	Reserved[1.]
1	0	0	12-bit data
1	x	x	Reserved[(1)]

1. Reserved settings cause a severe error at ADC conversion start whereby the CMD_EIF flag is set and ADC ceases operation

9.5.2.12 ADC Conversion Interrupt Flag Register (ADCCONIF)

After being set any of these bits can be cleared by writing a value of 1’b1. All bits are cleared if bit ADC_EN is clear or via ADC soft-reset (bit ADC_SR set). Writing any flag with value 1’b0 does not clear the flag. Writing any flag with value 1’b1 does not set the flag.

Module Base + 0x000C

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CON_IF[15:1]															EOL_IF
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 9-15. ADC Conversion Interrupt Flag Register (ADCCONIF)

Read: Anytime

Write: Anytime

Table 9-17. ADCCONIF Field Descriptions

Field	Description
15-1 CON_IF[15:1]	**Conversion Interrupt Flags** — These bits could be set by the binary coded interrupt select bits INTFLG_SEL[3:0] when the corresponding conversion command has been processed and related data has been stored to RAM. See also notes below.
0 EOL_IF	**End Of List Interrupt Flag** — This bit is set by the binary coded conversion command type select bits CMD_SEL[1:0] for “end of list” type of commands and after such a command has been processed and the related data has been stored RAM. See also second note below

NOTE

These bits can be used to indicate if a certain packet of conversion results is available. Clearing a flag indicates that conversion results have been retrieved by software and the flag can be used again (see also Section 9.9.6, “RVL swapping in RVL double buffer mode and related registers ADCIMDRI and ADCEOLRI.

NOTE

Overrun situation of a flag CON_IF[15:1] and EOL_IF are indicated by flag CONIF_OIF.

9.6 Functional Description

9.6.1 Overview

The ADC12B_LBA consists of an analog sub-block and a digital sub-block. It is a successive approximation analog-to-digital converter including a sample-and-hold mechanism and an internal charge scaled C-DAC (switched capacitor scaled digital-to-analog converter) with a comparator to realize the successive approximation algorithm.

9.6.2 Analog Sub-Block

The analog sub-block contains all analog circuits (sample and hold, C-DAC, analog Comparator, and so on) required to perform a single conversion. Separate power supplies VDDA and VSSA allow noise from the MCU circuitry to be isolated from the analog sub-block for improved accuracy.

9.6.2.1 Analog Input Multiplexer

The analog input multiplexers connect one of the external or internal analog input channels to the sample and hold storage node.

9.6.2.2 Sample and Hold Machine with Sample Buffer Amplifier

The Sample and Hold Machine controls the storage and charge of the storage node (sample capacitor) to the voltage level of the analog signal at the selected ADC input channel. This architecture employs the advantage of reduced crosstalk between channels.

The sample buffer amplifier is used to raise the effective input impedance of the A/D machine, so that external components (higher bandwidth or higher impedance connected as specified) are less significant to accuracy degradation.

During the sample phase, the analog input connects first via a sample buffer amplifier with the storage node always for two ADC clock cycles (“Buffer” sample time). For the remaining sample time (“Final” sample time) the storage node is directly connected to the analog input source. Please see also Figure 9-28 for illustration and the Appendix of the device reference manual for more details.
The input analog signals are unipolar and must be within the potential range of VSSA to VDDA.
During the hold process, the analog input is disconnected from the storage node.

15.3.2.17 PMF Internal Correction Control Register (PMFICCTL)

Address: Module Base + 0x001E

Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	PECC	PECB	PECA	ICCC	ICCB	ICCA
W								
Reset	0	0	0	0	0	0	0	0

Figure 15-20. PMF Internal Correction Control Register (PMFICCTL)

1. Read: Anytime
   Write: Anytime

This register is used to control PWM pulse generation for various applications, such as a power-supply phase-shifting application.

ICC*x* bits apply only in center-aligned operation during complementary mode. These control bits determine whether values set in the IPOL*x* bits control or the whether PWM count direction controls which PWM value register is used.

NOTE

The ICC*x* bits are buffered. The value written does not take effect until the next PWM load cycle begins regardless of the state of the LDOK bit or global load OK. Reading ICC*x* returns the value in a buffer and not necessarily the value the PWM generator is currently using.

The PEC*x* bits apply in edge-aligned and center-aligned operation during complementary mode. Setting the PEC*x* bits overrides the ICC*x* settings. This allows the PWM pulses generated by both the odd and even PWM value registers to be ANDed together prior to the complementary logic and deadtime insertion.

NOTE

The PEC*x* bits are buffered. The value written does not take effect until the related LDOK bit or global load OK is set and the next PWM load cycle begins. Reading PEC*n* returns the value in a buffer and not necessarily the value the PWM generator is currently using.

Figure 15-21. PMF Internal Correction Control Register (PMFICCTL) Descriptions

Field	Description
5 PECC	Pulse Edge Control — This bit controls PWM4/PWM5 pair. 0 Normal operation 1 Allow one of PMFVAL4 and PMFVAL5 to activate the PWM pulse and the other to deactivate the pulse
4 PECB	Pulse Edge Control — This bit controls PWM2/PWM3 pair. 0 Normal operation 1 Allow one of PMFVAL2 and PMFVAL3 to activate the PWM pulse and the other to deactivate the pulse
3 PECA	Pulse Edge Control — This bit controls PWM0/PWM1 pair. 0 Normal operation 1 Allow one of PMFVAL0 and PMFVAL1 to activate the PWM pulse and the other to deactivate the pulse

In Figure 16-22 the verification samples RT3 and RT5 determine that the first low detected was noise and not the beginning of a start bit. The RT clock is reset and the start bit search begins again. The noise flag is not set because the noise occurred before the start bit was found.

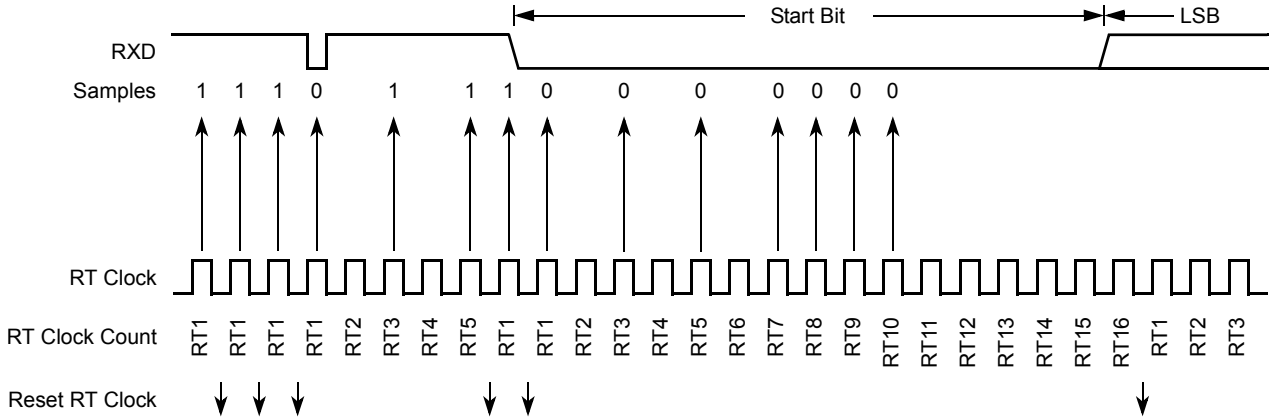

Figure 16-22. Start Bit Search Example 1

In Figure 16-23, verification sample at RT3 is high. The RT3 sample sets the noise flag. Although the perceived bit time is misaligned, the data samples RT8, RT9, and RT10 are within the bit time and data recovery is successful.

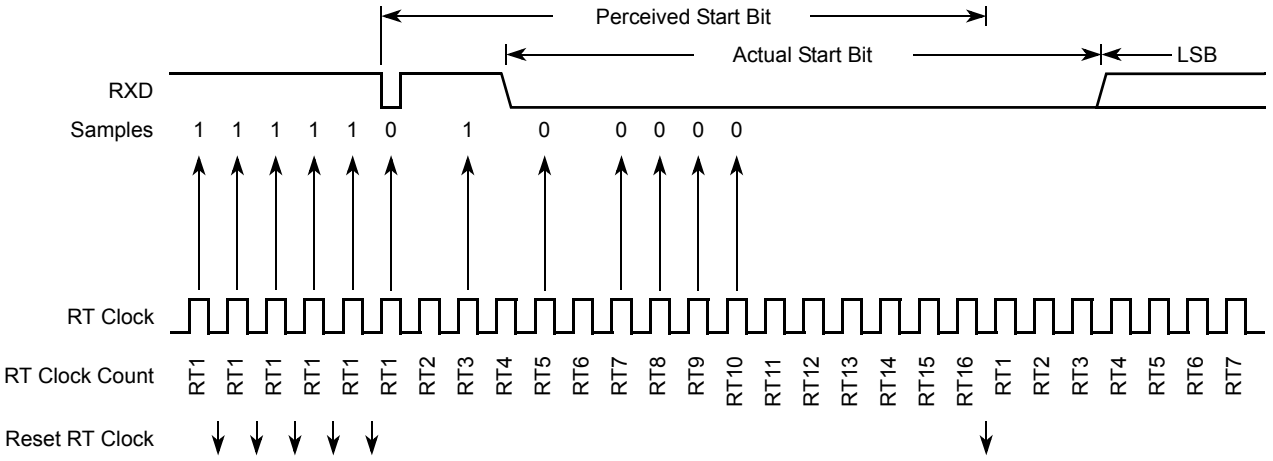

Figure 16-23. Start Bit Search Example 2

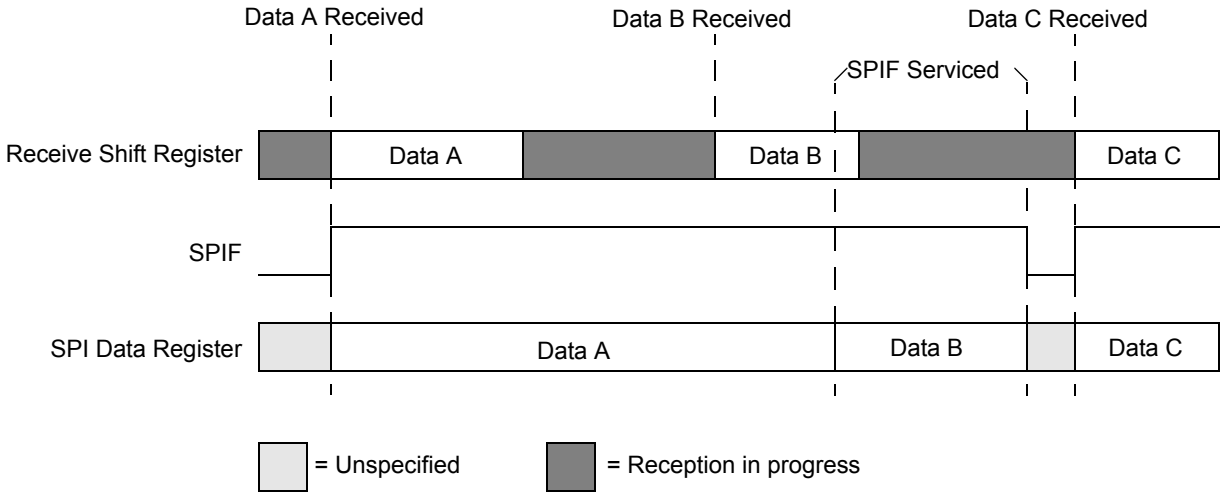

Figure 17-9. Reception with SPIF serviced in Time

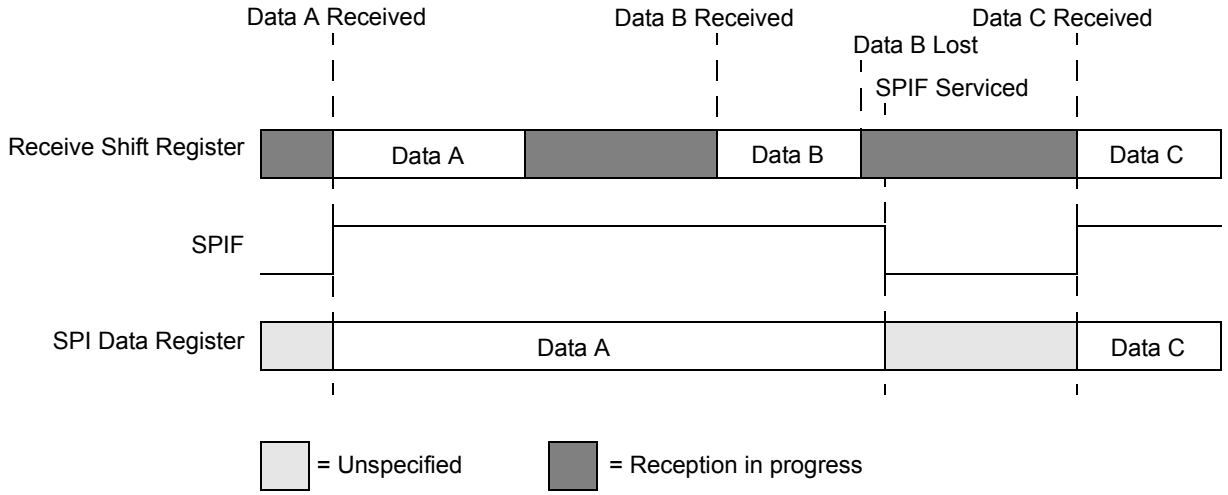

Figure 17-10. Reception with SPIF serviced too late

17.4 Functional Description

The SPI module allows a duplex, synchronous, serial communication between the MCU and peripheral devices. Software can poll the SPI status flags or SPI operation can be interrupt driven.

The SPI system is enabled by setting the SPI enable (SPE) bit in SPI control register 1. While SPE is set, the four associated SPI port pins are dedicated to the SPI function as:

- Slave select ($\overline{SS}$)
- Serial clock (SCK)
- Master out/slave in (MOSI)
- Master in/slave out (MISO)

As long as no more than one slave device drives the system slave's serial data output line, it is possible for several slaves to receive the same transmission from a master, although the master would not receive return information from all of the receiving slaves.

If the CPHA bit in SPI control register 1 is clear, odd numbered edges on the SCK input cause the data at the serial data input pin to be latched. Even numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

If the CPHA bit is set, even numbered edges on the SCK input cause the data at the serial data input pin to be latched. Odd numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

When CPHA is set, the first edge is used to get the first data bit onto the serial data output pin. When CPHA is clear and the $\overline{SS}$ input is low (slave selected), the first bit of the SPI data is driven out of the serial data output pin. After the nth[1] shift, the transfer is considered complete and the received data is transferred into the SPI data register. To indicate transfer is complete, the SPIF flag in the SPI status register is set.

NOTE

A change of the bits CPOL, CPHA, SSOE, LSBFE, MODFEN, SPC0, or BIDIROE with SPC0 set in slave mode will corrupt a transmission in progress and must be avoided.

17.4.3 Transmission Formats

During an SPI transmission, data is transmitted (shifted out serially) and received (shifted in serially) simultaneously. The serial clock (SCK) synchronizes shifting and sampling of the information on the two serial data lines. A slave select line allows selection of an individual slave SPI device; slave devices that are not selected do not interfere with SPI bus activities. Optionally, on a master SPI device, the slave select line can be used to indicate multiple-master bus contention.

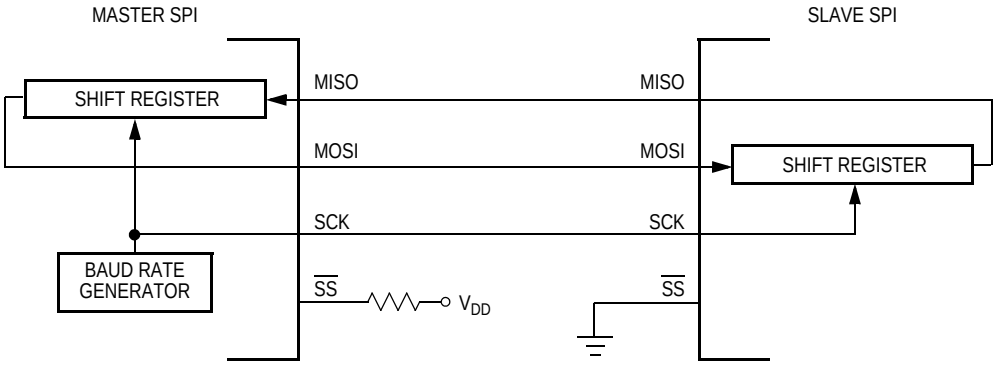

Figure 17-11. Master/Slave Transfer Block Diagram

17.4.3.1 Clock Phase and Polarity Controls

Using two bits in the SPI control register 1, software selects one of four combinations of serial clock phase and polarity.

1. n depends on the selected transfer width, please refer to Section 17.3.2.2, "SPI Control Register 2 (SPICR2)

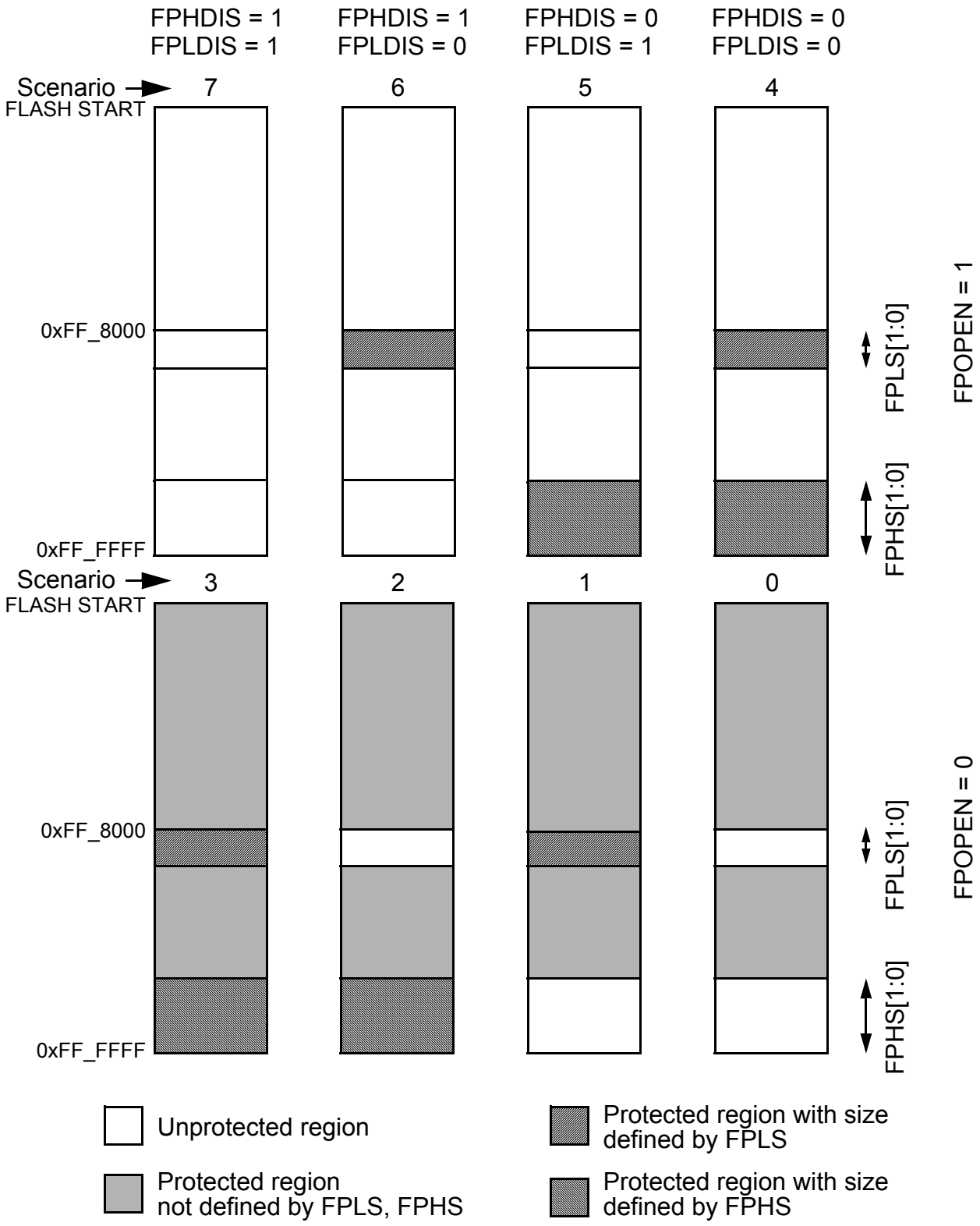

Figure 20-14. P-Flash Protection Scenarios

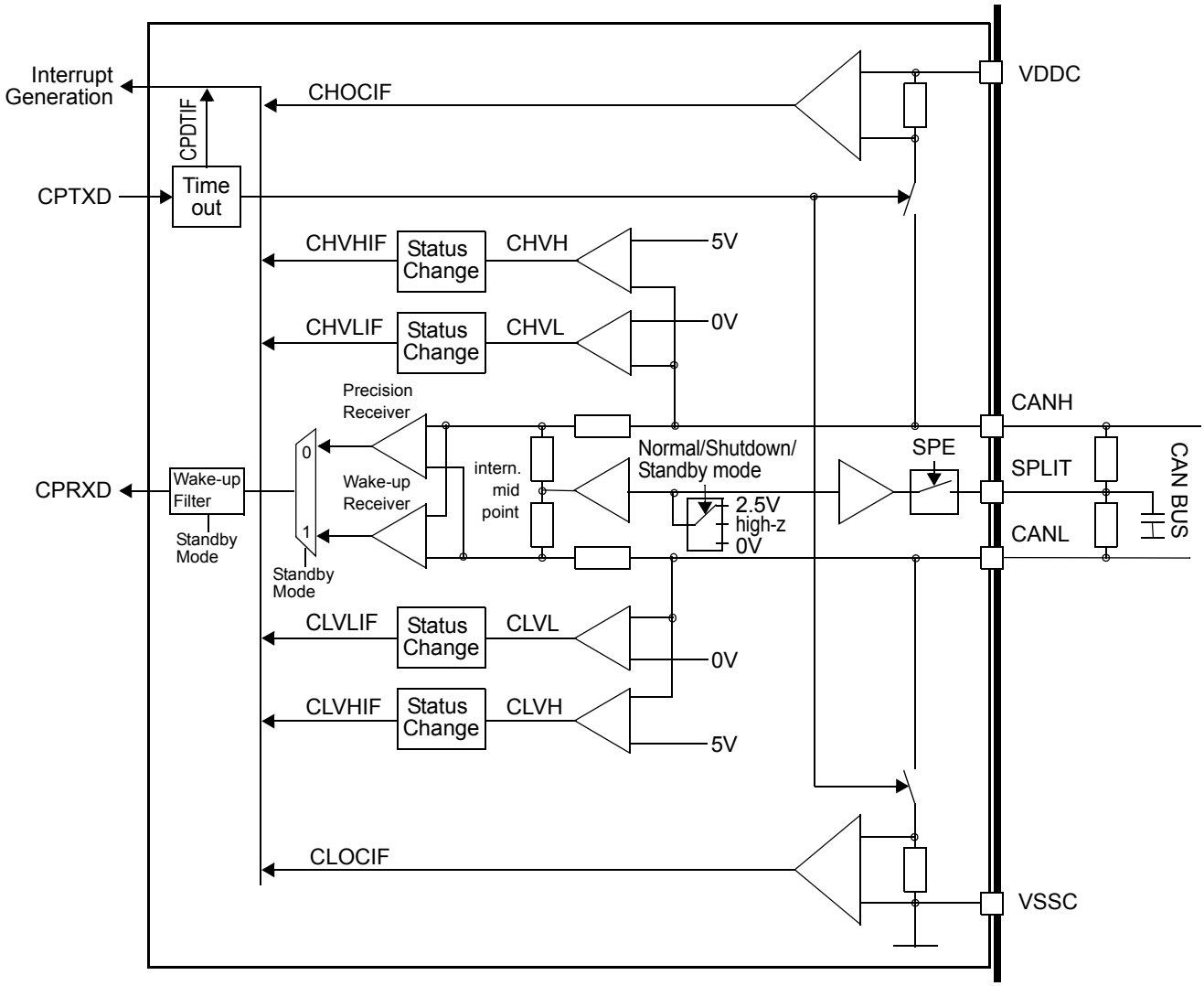

Figure 21-1. CAN Physical Layer Block Diagram

21.2 External Signal Description

Table 21-2 shows the external pins associated with the CAN Physical Layer.

Table 21-2. CAN Physical Layer Signal Properties

Name	Function
CANH	CAN Bus High Pin
SPLIT	2.5 V Termination Pin
CANL	CAN Bus Low Pin
VDDC	Supply Pin for CAN Physical Layer
VSSC	Ground Pin for CAN Physical Layer

In freeze mode there is a software programmable option to disable the input clock to the prescaler. This is useful for emulation.

Wait: The prescaler keeps on running, unless PSWAI in PWMCTL is set to 1.

Freeze: The prescaler keeps on running, unless PFRZ in PWMCTL is set to 1.

22.1.3 Block Diagram

Figure 22-1 shows the block diagram for the 8-bit up to 8-channel scalable PWM block.

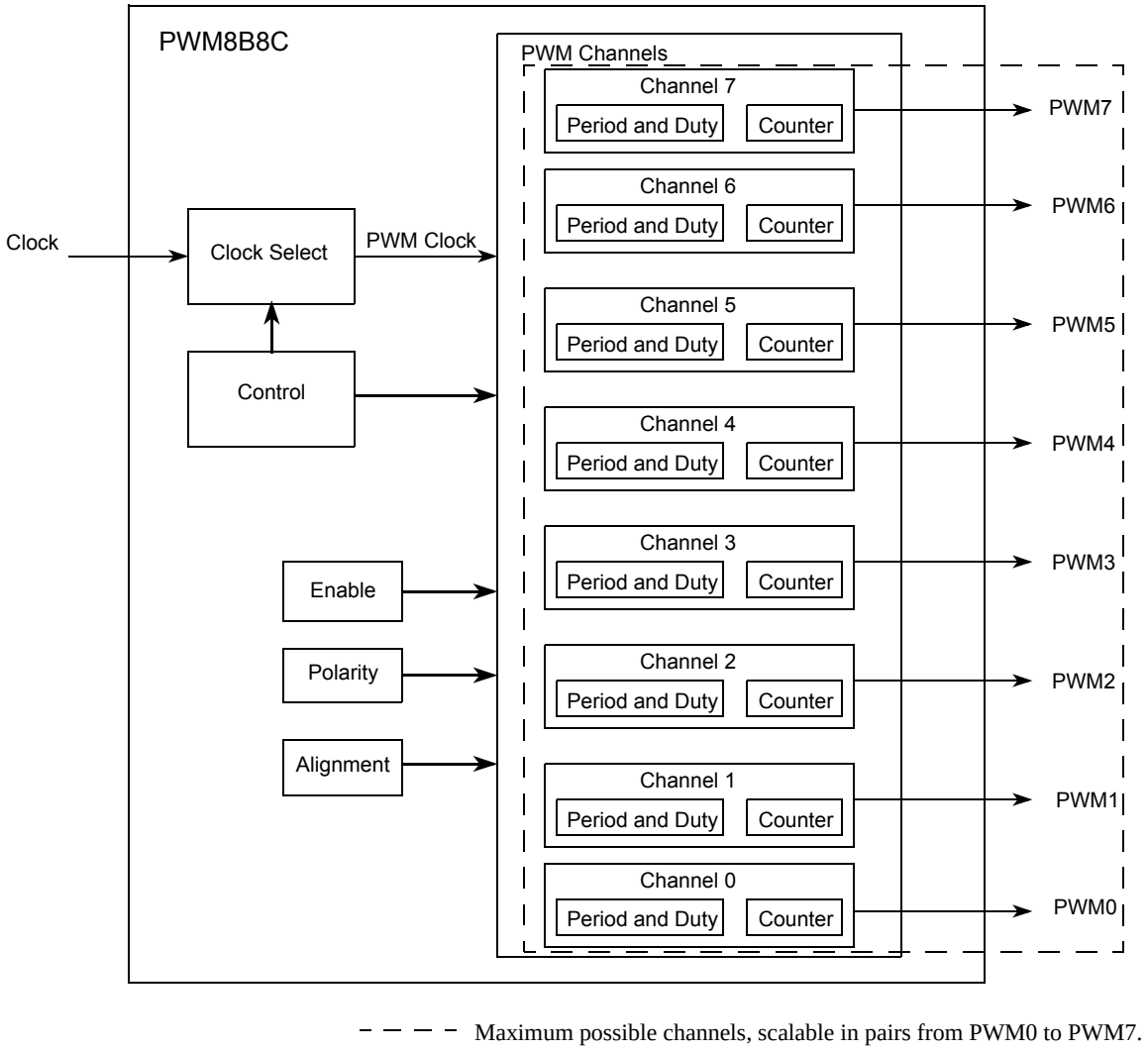

Figure 22-1. Scalable PWM Block Diagram

22.2 External Signal Description

The scalable PWM module has a selected number of external pins. Refer to device specification for exact number.

12. V(VBSx) - V(VLSx) > 9V, resp VLSx > 9V, pmos branch only
13. VLS > 6V
14. Output current range for which the effective output resistance specification applies
15. Input resistance can be calculated from the pin input leakage because the sense amp has high impedance MOS inputs
16. Av=10, no frequency compensation in feedback network, 90% output swing
17. Low side desaturation comparator range extends to LSx <= 2.35V - $V_{desatls}$

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0412	TIM1TC1H	R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
0x0413	TIM1TC1L	R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0414– 0x042B	Reserved	R W								
0x042C	TIM1OCPD	R W							OCPD1	OCPD0
0x042D	Reserved	R W								
0x042E	TIM1PTPSR	R W	PTPS7	PTPS6	PTPS5	PTPS4	PTPS3	PTPS2	PTPS1	PTPS0
0x042F	Reserved	R W								

M.9 0x0480-0x04AF PWM0

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0480	PWME	R W	PWME7	PWME6	PWME5	PWME4	PWME3	PWME2	PWME1	PWME0
0x0481	PWMPOL	R W	PPOL7	PPOL6	PPOL5	PPOL4	PPOL3	PPOL2	PPOL1	PPOL0
0x0482	PWMCLK	R W	PCLK7	PCLKL6	PCLK5	PCLK4	PCLK3	PCLK2	PCLK1	PCLK0
0x0483	PWMPRCLK	R W	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
0x0484	PWMCAE	R W	CAE7	CAE6	CAE5	CAE4	CAE3	CAE2	CAE1	CAE0
0x0485	PWMCTL	R W	CON67	CON45	CON23	CON01	PSWAI	PFRZ	0	0
0x0486	PWMCLKAB	R W	PCLKAB7	PCLKAB6	PCLKAB5	PCLKAB4	PCLKAB3	PCLKAB2	PCLKAB1	PCLKAB0
0x0487	RESERVED	R W	0	0	0	0	0	0	0	0
0x0488	PWMSCLA	R W	Bit 7	6	5	4	3	2	1	Bit 0

M.9 0x0480-0x04AF PWM0

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x049B	PWMPER7	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x049C	PWMDTY0	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x049D	PWMDTY1	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x049E	PWMDTY2	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x049F	PWMDTY32	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x04A0	PWMDTY42	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x04A1	PWMDTY52	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x04A2	PWMDTY62	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x04A3	PWMDTY72	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x04A4 - 0x04AF	RESERVED	R W	0	0	0	0	0	0	0	0

M.10 0x0500-x053F PMF15B6C

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0500	PMFCFG0	R W	WP	MTG	EDGEC	EDGEB	EDGEA	INDEPC	INDEPB	INDEPA
0x0501	PMFCFG1	R W	0	ENCE	BOTNEGC	TOPNEGC	BOTNEGB	TOPNEGB	BOTNEGA	TOPNEGA
0x0502	PMFCFG2	R W	REV1	REV0	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0
0x0503	PMFCFG3	R W	PMFWAI	PMFFRZ	0	VLMODE		PINVC	PINVB	PINVA
0x0504	PMFFEN	R W	0	FEN5	0	FEN4	FEN3	FEN2	FEN1	FEN0
0x0505	PMFFMOD	R W	0	FMOD5	0	FMOD4	FMOD3	FMOD2	FMOD1	FMOD0