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	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	EBI/EMI, I²C, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	91
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 2.75V
Data Converters	A/D 16x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s12e128mpve

0x0200 – 0x023F PMF (Pulse width Modulator with Fault protection) (Sheet 4 of 4)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0239	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x023A	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x023B	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x023C	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x023D	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x023E	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x023F	Reserved	R	0	0	0	0	0	0	0	0
		W								

0x0240 – 0x027F PIM (Port Interface Module) (Sheet 1 of 4)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0240	PTT	R	PTT7	PTT6	PTT5	PTT4	PTT3	PTT2	PTT1	PTT0
		W								
0x0241	PTIT	R	PTIT7	PTIT6	PTIT5	PTIT4	PTIT3	PTIT2	PTIT1	PTIT0
		W								
0x0242	DDRT	R	DDRT7	DDRT7	DDRT5	DDRT4	DDRT3	DDRT2	DDRT1	DDRT0
		W								
0x0243	RDRT	R	RDRT7	RDRT6	RDRT5	RDRT4	RDRT3	RDRT2	RDRT1	RDRT0
		W								
0x0244	PERT	R	PERT7	PERT6	PERT5	PERT4	PERT3	PERT2	PERT1	PERT0
		W								
0x0245	PPST	R	PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0
		W								
0x0246	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0247	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0248	PTS	R	PTS7	PTS6	PTS5	PTS4	PTS3	PTS2	PTS1	PTS0
		W								
0x0249	PTIS	R	PTIS7	PTIS6	PTIS5	PTIS4	PTIS3	PTIS2	PTIS1	PTIS0
		W								

2.3.2 Register Descriptions

The Flash module contains a set of 16 control and status registers located between module base + 0x0000 and 0x000F. A summary of the Flash module registers is given in Figure 2-3. Detailed descriptions of each register bit are provided.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 FCLKDIV	R	FDIVLD	PRDIV8	FDIV5	FDIV4	FDIV3	FDIV2	FDIV1	FDIV0
	W								
0x0001 FSEC	R	KEYEN1	KEYEN0	NV5	NV4	NV3	NV2	SEC1	SEC0
	W								
0x0002 RESERVED1[1]	R	0	0	0	0	0	0	0	0
	W								
0x0003 FCNFG	R	CBEIE	CCIE	KEYACC	0	0	0	0	0
	W								
0x0004 FPROT	R	FPOPEN	NV6	FPHDIS	FPHS1	FPHS0	FPLDIS	FPLS1	FPLS0
	W								
0x0005 FSTAT	R	CBEIF	CCIF	PVIOL	ACCERR	0	BLANK	FAIL	DONE
	W								
0x0006 FCMD	R	0	CMDB6	CMDB5	0	0	CMDB2	0	CMDB0
	W								
0x0007 RESERVED2[1]	R	0	0	0	0	0	0	0	0
	W								
0x0008 FADDRHI[1]	R	FABHI							
	W								
0x0009 FADDRLO[1]	R	FABLO							
	W								
0x000A FDATAHI[1]	R	FDHI							
	W								
0x000B FDATALO[1]	R	FDLO							
	W								
0x000C RESERVED3[1]	R	0	0	0	0	0	0	0	0
	W								
0x000D RESERVED4[1]	R	0	0	0	0	0	0	0	0
	W								
0x000E RESERVED5[1]	R	0	0	0	0	0	0	0	0
	W								
0x000F RESERVED6[1]	R	0	0	0	0	0	0	0	0
	W								

= Unimplemented or Reserved

Figure 2-3. Flash Register Summary

1 Intended for factory test purposes only.

3.1.2 Block Diagram

Figure 3-1 is a block diagram of the PIM9E128V1.

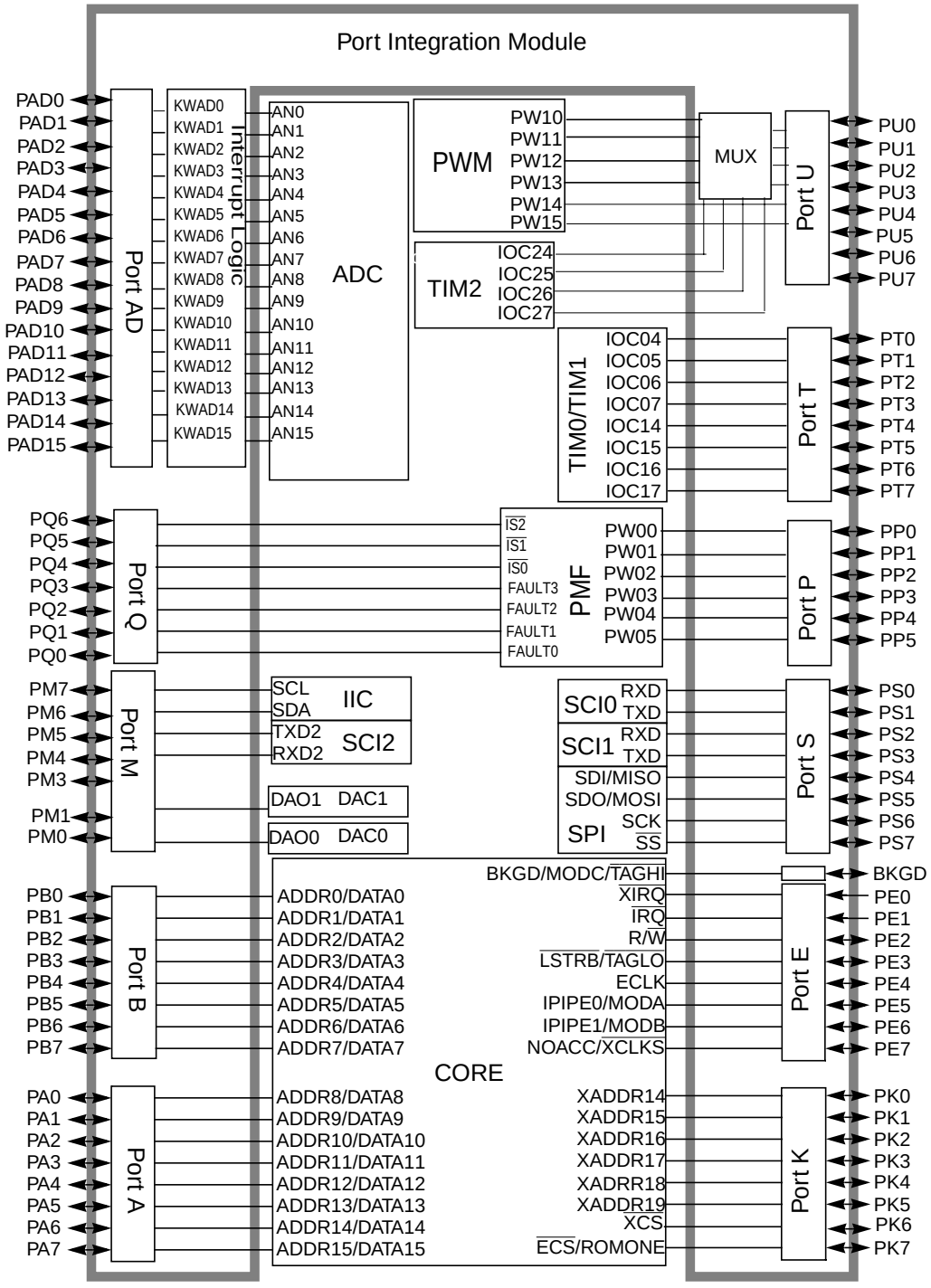


Figure 3-1. PIM9E128V1 Block Diagram

Table 4-4. CLKSEL Field Descriptions (continued)

Field	Description
1 RTIWAI	**RTI Stops in Wait Mode Bit** — Write: anytime 0 RTI keeps running in wait mode. 1 RTI stops and initializes the RTI dividers whenever the part goes into wait mode.
0 COPWAI	**COP Stops in Wait Mode Bit** — Normal modes: Write once —Special modes: Write anytime 0 COP keeps running in wait mode. 1 COP stops and initializes the COP dividers whenever the part goes into wait mode.

4.3.2.7 CRG PLL Control Register (PLLCTL)

This register controls the PLL functionality.

	7	6	5	4	3	2	1	0
R	CME	PLLON	AUTO	ACQ	0	PRE	PCE	SCME
W								
Reset	1	1	1	1	0	0	0	1

= Unimplemented or Reserved

Figure 4-10. CRG PLL Control Register (PLLCTL)

Read: anytime

Write: refer to each bit for individual write conditions

Table 4-5. PLLCTL Field Descriptions

Field	Description
7 CME	**Clock Monitor Enable Bit** — CME enables the clock monitor. Write anytime except when SCM = 1. 0 Clock monitor is disabled. 1 Clock monitor is enabled. Slow or stopped clocks will cause a clock monitor reset sequence or self-clock mode. **Note:** Operating with CME = 0 will not detect any loss of clock. In case of poor clock quality this could cause unpredictable operation of the MCU. **Note:** In Stop Mode (PSTP = 0) the clock monitor is disabled independently of the CME bit setting and any loss of clock will not be detected.
6 PLLON	**Phase Lock Loop On Bit** — PLLON turns on the PLL circuitry. In self-clock mode, the PLL is turned on, but the PLLON bit reads the last latched value. Write anytime except when PLLSEL = 1. 0 PLL is turned off. 1 PLL is turned on. If AUTO bit is set, the PLL will lock automatically.
5 AUTO	**Automatic Bandwidth Control Bit** — AUTO selects either the high bandwidth (acquisition) mode or the low bandwidth (tracking) mode depending on how close to the desired frequency the VCO is running. Write anytime except when PLLWAI=1, because PLLWAI sets the AUTO bit to 1. 0 Automatic mode control is disabled and the PLL is under software control, using ACQ bit. 1 Automatic mode control is enabled and ACQ bit has no effect.
4 ACQ	**Acquisition Bit** — Write anytime. If AUTO=1 this bit has no effect. 0 Low bandwidth filter is selected. 1 High bandwidth filter is selected.

6.3.2.4 ATD Control Register 3 (ATDCTL3)

This register controls the conversion sequence length, FIFO for results registers and behavior in Freeze Mode. Writes to this register will abort current conversion sequence but will not start a new sequence.

	7	6	5	4	3	2	1	0
R	0	S8C	S4C	S2C	S1C	FIFO	FRZ1	FRZ0
W								
Reset	0	0	1	0	0	0	0	0

= Unimplemented or Reserved

Figure 6-6. ATD Control Register 3 (ATDCTL3)

Read: Anytime

Write: Anytime

Table 6-4. ATDCTL3 Field Descriptions

Field	Description
6 S8C	**Conversion Sequence Length** — This bit controls the number of conversions per sequence. Table 6-5 shows all combinations. At reset, S4C is set to 1 (sequence length is 4). This is to maintain software continuity to HC12 Family.
5 S4C	**Conversion Sequence Length** — This bit controls the number of conversions per sequence. Table 6-5 shows all combinations. At reset, S4C is set to 1 (sequence length is 4). This is to maintain software continuity to HC12 Family.
4 S2C	**Conversion Sequence Length** — This bit controls the number of conversions per sequence. Table 6-5 shows all combinations. At reset, S4C is set to 1 (sequence length is 4). This is to maintain software continuity to HC12 Family.
3 S1C	**Conversion Sequence Length** — This bit controls the number of conversions per sequence. Table 6-5 shows all combinations. At reset, S4C is set to 1 (sequence length is 4). This is to maintain software continuity to HC12 Family.

Chapter 8
Serial Communication Interface (SCIV3)

8.1 Introduction

This block description chapter provides an overview of serial communication interface (SCI) module.

The SCI allows full duplex, asynchronous, serial communication between the CPU and remote devices, including other CPUs. The SCI transmitter and receiver operate independently, although they use the same baud rate generator. The CPU monitors the status of the SCI, writes the data to be transmitted, and processes received data.

8.1.1 Glossary

IR: infrared

IrDA: Infrared Design Association

IRQ: interrupt request

LSB: least significant bit

MSB: most significant bit

NRZ: non-return-to-Zero

RZI: return-to-zero-inverted

RXD: receive pin

SCI: serial communication interface

TXD: transmit pin

8.1.2 Features

The SCI includes these distinctive features:

- Full-duplex or single-wire operation
- Standard mark/space non-return-to-zero (NRZ) format
- Selectable IrDA 1.4 return-to-zero-inverted (RZI) format with programmable pulse widths
- 13-bit baud rate selection
- Programmable 8-bit or 9-bit data format
- Separately enabled transmitter and receiver
- Programmable transmitter output parity

8.4.5.6.1 Idle Input Line Wakeup (WAKE = 0)

In this wakeup method, an idle condition on the RXD pin clears the RWU bit and wakes up the SCI. The initial frame or frames of every message contain addressing information. All receivers evaluate the addressing information, and receivers for which the message is addressed process the frames that follow. Any receiver for which a message is not addressed can set its RWU bit and return to the standby state. The RWU bit remains set and the receiver remains on standby until another idle character appears on the RXD pin.

Idle line wakeup requires that messages be separated by at least one idle character and that no message contains idle characters.

The idle character that wakes a receiver does not set the receiver idle bit, IDLE, or the receive data register full flag, RDRF.

The idle line type bit, ILT, determines whether the receiver begins counting logic 1s as idle character bits after the start bit or after the stop bit. ILT is in SCI control register 1 (SCICR1).

8.4.5.6.2 Address Mark Wakeup (WAKE = 1)

In this wakeup method, a logic 1 in the most significant bit (MSB) position of a frame clears the RWU bit and wakes up the SCI. The logic 1 in the MSB position marks a frame as an address frame that contains addressing information. All receivers evaluate the addressing information, and the receivers for which the message is addressed process the frames that follow. Any receiver for which a message is not addressed can set its RWU bit and return to the standby state. The RWU bit remains set and the receiver remains on standby until another address frame appears on the RXD pin.

The logic 1 MSB of an address frame clears the receiver's RWU bit before the stop bit is received and sets the RDRF flag.

Address mark wakeup allows messages to contain idle characters but requires that the MSB be reserved for use in address frames.

NOTE

With the WAKE bit clear, setting the RWU bit after the RXD pin has been idle can cause the receiver to wake up immediately.

8.4.6 Single-Wire Operation

Normally, the SCI uses two pins for transmitting and receiving. In single-wire operation, the RXD pin is disconnected from the SCI. The SCI uses the TXD pin for both receiving and transmitting.

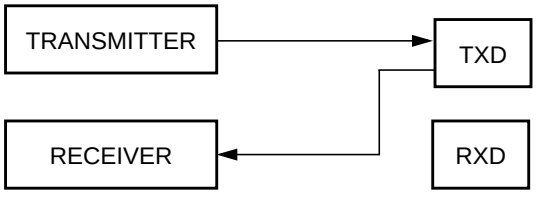


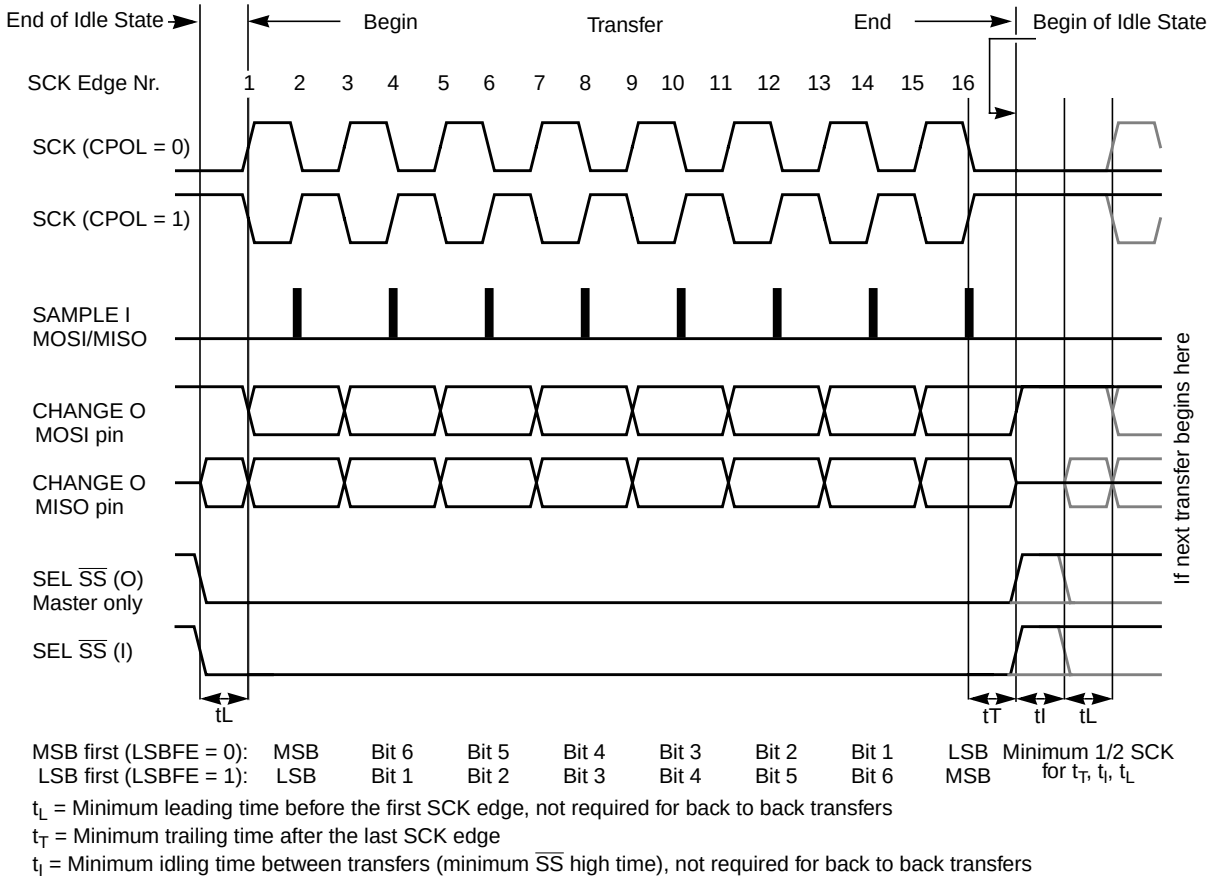
Figure 8-24. Single-Wire Operation (LOOPS = 1, RSRC = 1)

Table 9-7. Example SPI Baud Rate Selection (25 MHz Bus Clock) (continued)

SPPR2	SPPR1	SPPR0	SPR2	SPR1	SPR0	Baud Rate Divisor	Baud Rate
1	0	0	1	1	1	1280	19.53 kHz
1	0	1	0	0	0	12	2.08333 MHz
1	0	1	0	0	1	24	1.04167 MHz
1	0	1	0	1	0	48	520.83 kHz
1	0	1	0	1	1	96	260.42 kHz
1	0	1	1	0	0	192	130.21 kHz
1	0	1	1	0	1	384	65.10 kHz
1	0	1	1	1	0	768	32.55 kHz
1	0	1	1	1	1	1536	16.28 kHz
1	1	0	0	0	0	14	1.78571 MHz
1	1	0	0	0	1	28	892.86 kHz
1	1	0	0	1	0	56	446.43 kHz
1	1	0	0	1	1	112	223.21 kHz
1	1	0	1	0	0	224	111.61 kHz
1	1	0	1	0	1	448	55.80 kHz
1	1	0	1	1	0	896	27.90 kHz
1	1	0	1	1	1	1792	13.95 kHz
1	1	1	0	0	0	16	1.5625 MHz
1	1	1	0	0	1	32	781.25 kHz
1	1	1	0	1	0	64	390.63 kHz
1	1	1	0	1	1	128	195.31 kHz
1	1	1	1	0	0	256	97.66 kHz
1	1	1	1	0	1	512	48.83 kHz
1	1	1	1	1	0	1024	24.41 kHz
1	1	1	1	1	1	2048	12.21 kHz

NOTE

In slave mode of SPI S-clock speed DIV2 is not supported.



t_L = Minimum leading time before the first SCK edge, not required for back to back transfers
t_T = Minimum trailing time after the last SCK edge
t_I = Minimum idling time between transfers (minimum $\overline{SS}$ high time), not required for back to back transfers

Figure 9-10. SPI Clock Format 1 (CPHA = 1)

9.4.4 SPI Baud Rate Generation

Baud rate generation consists of a series of divider stages. Six bits in the SPI Baud Rate register (SPPR2, SPPR1, SPPR0, SPR2, SPR1, and SPR0) determine the divisor to the SPI module clock which results in the SPI baud rate.

The SPI clock rate is determined by the product of the value in the baud rate preselection bits (SPPR2–SPPR0) and the value in the baud rate selection bits (SPR2–SPR0). The module clock divisor equation is shown in Figure 9-11

When all bits are clear (the default condition), the SPI module clock is divided by 2. When the selection bits (SPR2–SPR0) are 001 and the preselection bits (SPPR2–SPPR0) are 000, the module clock divisor becomes 4. When the selection bits are 010, the module clock divisor becomes 8 etc.

When the preselection bits are 001, the divisor determined by the selection bits is multiplied by 2. When the preselection bits are 010, the divisor is multiplied by 3, etc. See Table 9-7 for baud rate calculations for all bit conditions, based on a 25-MHz bus clock. The two sets of selects allows the clock to be divided by a non-power of two to achieve other baud rates such as divide by 6, divide by 10, etc.

11.1.2 Modes of Operation

Care must be exercised when using this module in the modes listed in Table 11-1. PWM outputs are placed in their inactive states in stop mode, and optionally under WAIT and freeze modes. PWM outputs will be reactivated (assuming they were active to begin with) when these modes are exited.

Table 11-1. Modes When PWM Operation is Restricted

Mode	Description
Stop	PWM outputs are disabled
Wait	PWM outputs are disabled as a function of the PMFWAI bit.
Freeze	PWM outputs are disabled as a function of the PMFFRZ bit.

11.1.3 Block Diagrams

Figure 11-1 provides an overview of the PMF module.

The Mux/Swap/Current Sense block is tightly integrated with the dead time insertion block . This detail is shown in Figure 11-2.

NOTE

It is possible to have both channels of a complementary pair to be high. For example, if the TOPNEGA (negative polarity for PWM0), BOTNEGA (negative polarity for PWM1), MASK0, and MASK1 bits are set, both the PWM complementary outputs of generator A will be high. See Section 11.3.2.2, “PMF Configure 1 Register (PMFCFG1)” for the description of TOPNEG and BOTNEG bits, and Section 11.3.2.3, “PMF Configure 2 Register (PMFCFG2)” for the description of the MSK0 and MSK1 bits.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0012	PMFVAL1	R W	PMFVAL1							
0x0013	PMFVAL1	R W	PMFVAL1							
0x0014	PMFVAL2	R W	PMFVAL2							
0x0015	PMFVAL2	R W	PMFVAL2							
0x0016	PMFVAL3	R W	PMFVAL3							
0x0017	PMFVAL3	R W	PMFVAL3							
0x0018	PMFVAL4	R W	PMFVAL4							
0x0019	PMFVAL4	R W	PMFVAL4							
0x001A	PMFVAL5	R W	PMFVAL5							
0x001B	PMFVAL5	R W	PMFVAL5							
0x001C ↓ 0X001F	Reserved	R W								
0x0020	PMFENCA	R W	PWMENA	0	0	0	0	0	LDOKA	PWMRIEA
0x0021	PMFFQCA	R W	LDFQA				HALFA	PRSCA		PWMRFA
0x0022	PMFCNTA	R W	0	PMFCNTA						
0x0023	PMFCNTA	R W	PMFCNTA							
0x0024	PMFMODA	R W	0	PMFMODA						
0x0025	PMFMODA	R W	PMFMODA							
0x0026	PMFDTMA	R W	0	0	0	0	PMFDTMA			
0x0027	PMFDTMA	R W	PMFDTMA							

= Unimplemented or Reserved

Figure 11-3. PMF15B6C Register Summary (Sheet 2 of 3)

11.3.2.20 PMF Enable Control A Register (PMFENCA)

Module Base + 0x0020

	7	6	5	4	3	2	1	0
R	PWMENA	0	0	0	0	0	LDOKA	PWMRIEA
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 11-26. PMF Enable Control A Register (PMFENCA)

Read and write anytime.

Table 11-25. PMFENCA Field Descriptions

Field	Description
7 PWMENA	**PWM Generator A Enable** — When MTG is clear, this bit when set enables the PWM generators A, B and C and the PWM0–5 pins. When PWMENA is clear, PWM generators A, B and C are disabled, and the PWM0–5 pins are in their inactive states unless the corresponding OUTCTLx bits are set. When MTG is set, this bit when set enables the PWM generator A and the PWM0 and PWM1 pins. When PWMENA is clear, the PWM generator A is disabled and PWM0 and PWM1 pins are in their inactive states unless the OUTCTL0 and OUTCTL1 bits are set. 0 PWM generator A and PWM0–1 (2–5 if MTG=0) pins disabled unless the respective OUTCTL bit is set. 1 PWM generator A and PWM0–1 (2–5 if MTG=0) pins enabled.
1 LDOKA	**Load Okay A** — When MTG is clear, this bit allows loads of the PRSCA bits, the PMFMODA register and the PWMVAL0–5 registers into a set of buffers. The buffered prescaler A divisor, PWM counter modulus A value, and all PWM pulse widths take effect at the next PWM reload. When MTG is set, this bit allows loads of the PRSCA bits, the PMFMODA register and the PWMVAL0–1 registers into a set of buffers. The buffered prescaler divisor A, PWM counter modulus A value, PWM0–1 pulse widths take effect at the next PWM reload. Set LDOKA by reading it when it is logic zero and then writing a logic one to it. LDOKA is automatically cleared after the new values are loaded, or can be manually cleared before a reload by writing a logic zero to it. Reset clears LDOKA. 0 Do not load new modulus A, prescaler A, and PWM0–1 (2–5 if MTG=0) values 1 Load prescaler A, modulus A, and PWM0–1 (2–5 if MTG=0) values **Note:** Do not set PWMENA bit before setting the LDOKA bit and do not clear the LDOKA bit at the same time as setting the PWMENA bit.
0 PWMRIEA	**PWM Reload Interrupt Enable A** — This bit enables the PWMRFA flag to generate CPU interrupt requests. 0 PWMRFA CPU interrupt requests disabled 1 PWMRFA CPU interrupt requests enabled

11.3.2.31 PMF Frequency Control C Register (PMFFQCC)

Module Base + 0x0031

	7	6	5	4	3	2	1	0
R W	LDFQC				HALFC	PRSCC		PWMRFC
Reset	0	0	0	0	0	0	0	0

Figure 11-37. PMF Frequency Control C Register (PMFFQCC)

Read anytime and write only if MTG is set.

Table 11-40. PMFFQCC Field Descriptions

Field	Description
7–4 LDFQC	**Load Frequency C** — This field selects the PWM load frequency according to Table 11-41. See Section 11.4.7.2, “Load Frequency” for more details. **Note:** The LDFQC field takes effect when the current load cycle is complete, regardless of the state of the load okay bit, LDOKC. Reading the LDFQC field reads the buffered value and not necessarily the value currently in effect.
3 HALFC	**Half Cycle Reload C** — This bit enables half-cycle reloads in center-aligned PWM mode. This bit has no effect on edge-aligned PWMs. 0 Half-cycle reloads disabled 1 Half-cycle reloads enabled
2 PRSCC	**Prescaler C** — This buffered field selects the PWM clock frequency illustrated in Table 11-42. **Note:** Reading the PRSCC field reads the buffered value and not necessarily the value currently in effect. The PRSCC field takes effect at the beginning of the next PWM cycle and only when the load okay bit, LDOKC, is set.
0 PWMRFC	**PWM Reload Flag C** — This flag is set at the beginning of every reload cycle regardless of the state of the LDOKC bit. Clear PWMRFC by reading PMFFQCC with PWMRFC set and then writing a logic one to the PWMRFC bit. If another reload occurs before the clearing sequence is complete, writing logic one to PWMRFC has no effect. 0 No new reload cycle since last PWMRFC clearing 1 New reload cycle since last PWMRFC clearing **Note:** Clearing PWMRFC satisfies pending PWMRFC CPU interrupt requests.

Table 11-41. PWM Reload Frequency C

LDFQC	PWM Reload Frequency	LDFQ[3:0]	PWM Reload Frequency
0000	Every PWM opportunity	1000	Every 9 PWM opportunities
0001	Every 2 PWM opportunities	1001	Every 10 PWM opportunities
0010	Every 3 PWM opportunities	1010	Every 11 PWM opportunities
0011	Every 4 PWM opportunities	1011	Every 12 PWM opportunities
0100	Every 5 PWM opportunities	1100	Every 13 PWM opportunities
0101	Every 6 PWM opportunities	1101	Every 14 PWM opportunities
0110	Every 7 PWM opportunities	1110	Every 15 PWM opportunities
0111	Every 8 PWM opportunities	1111	Every 16 PWM opportunities

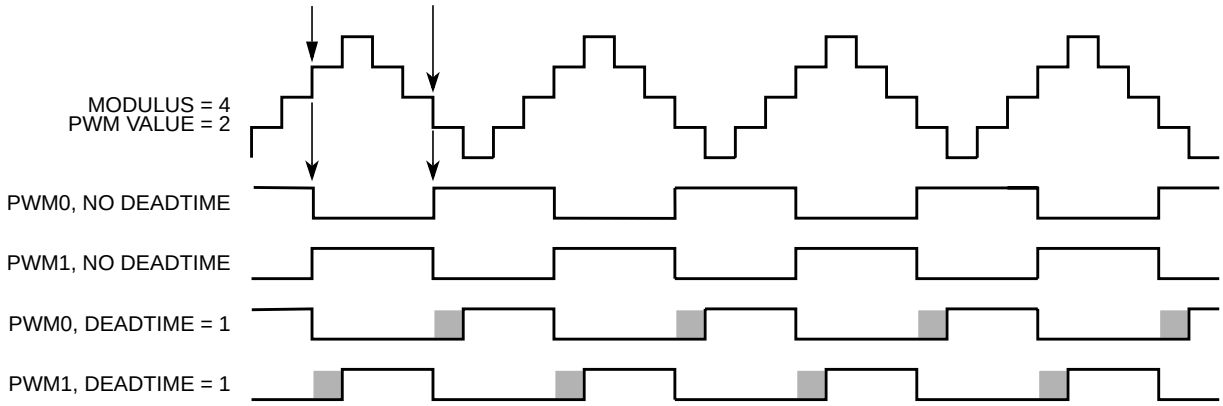


Figure 11-50. Deadtime Insertion, Center Alignment

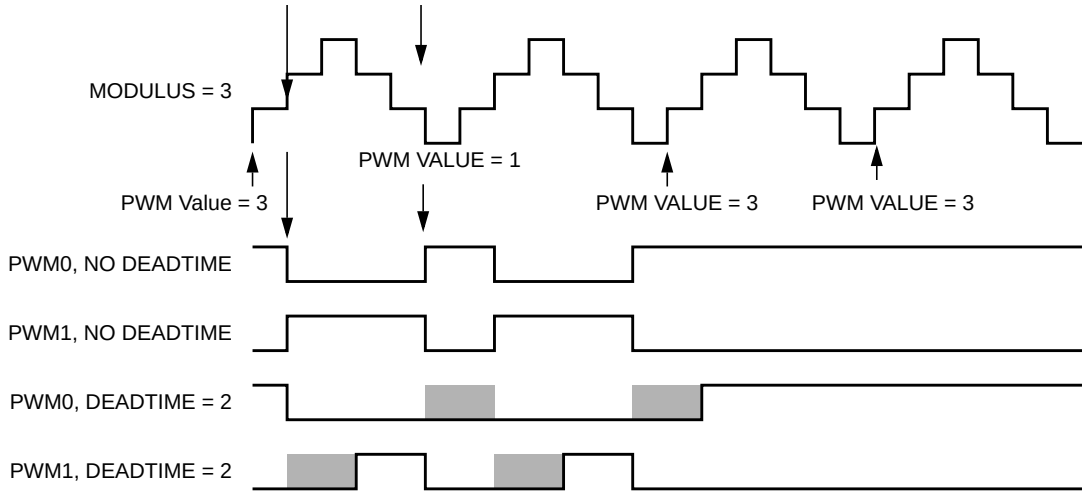


Figure 11-51. Deadtime at Duty Cycle Boundaries

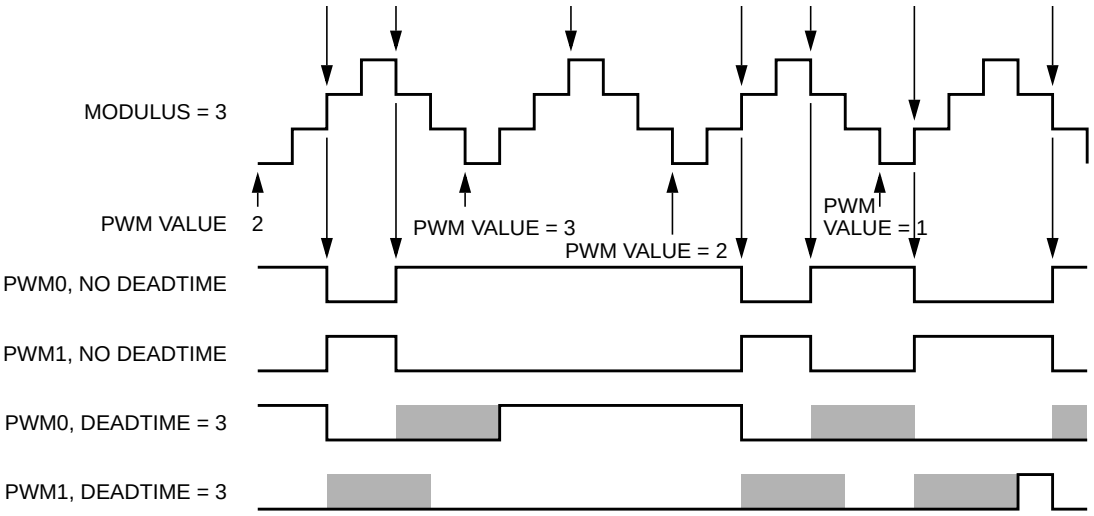


Figure 11-52. Deadtime and Small Pulse Widths

NOTE

The waveform at the pad is delayed by two bus clock cycles for deadtime insertion.

11.4.7.3 Reload Flag

With a reload opportunity, regardless an actual reload occurs as determined by LDOK bit, the PWMF reload flag is set. If the PWM reload interrupt enable bit, PWMRIE is set, the PWMF flag generates CPU interrupt requests allowing software to calculate new PWM parameters in real time. When PWMRIE is not set, reloads still occur at the selected reload rate without generating CPU interrupt requests.

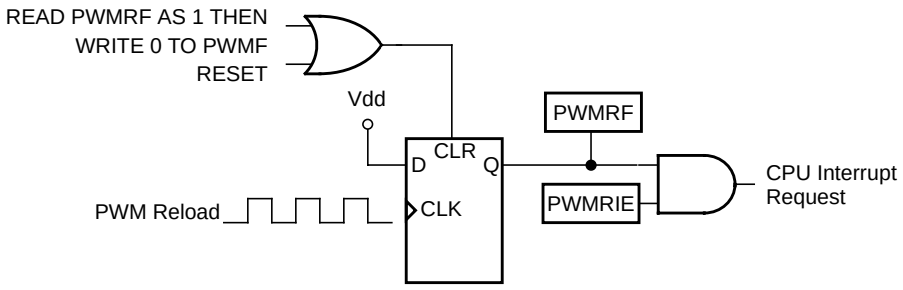


Figure 11-67. PWMRF Reload Interrupt Request

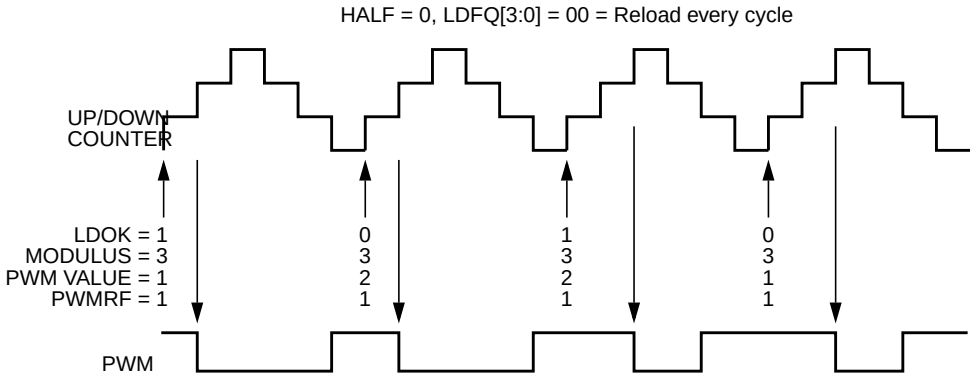


Figure 11-68. Full-Cycle Center-Aligned PWM Value Loading

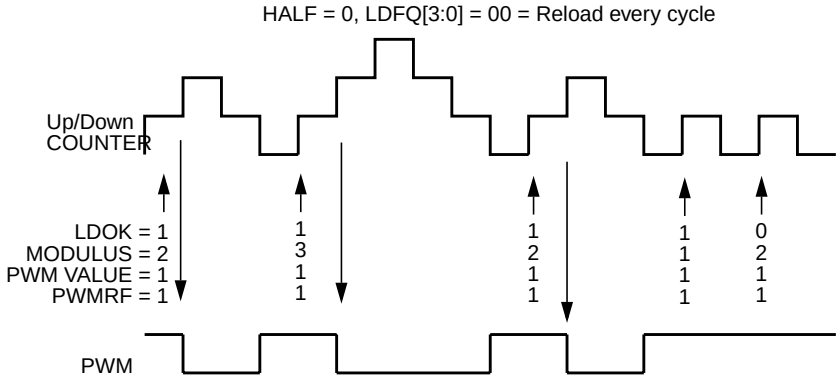


Figure 11-69. Full-Cycle Center-Aligned Modulus Loading

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

Figure 12-18. PWM Channel Counter Registers (PWMCNT3)

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

Figure 12-19. PWM Channel Counter Registers (PWMCNT4)

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

Figure 12-20. PWM Channel Counter Registers (PWMCNT5)

Read: anytime

Write: anytime (any value written causes PWM counter to be reset to 0x0000).

12.3.2.13 PWM Channel Period Registers (PWMPERx)

There is a dedicated period register for each channel. The value in this register determines the period of the associated PWM channel.

The period registers for each channel are double buffered so that if they change while the channel is enabled, the change will NOT take effect until one of the following occurs:

- The effective period ends
- The counter is written (counter resets to 0x0000)
- The channel is disabled

In this way, the output of the PWM will always be either the old waveform or the new waveform, not some variation in between. If the channel is not enabled, then writes to the period 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 period due to the double buffering scheme.

Reference Section 12.4.2.3, “PWM Period and Duty,” for more information.

12.4.2.1 PWM Enable

Each PWM channel has an enable bit (PWMEx) to start its waveform output. When any of the PWMEx bits are set (PWMEx = 1), the associated PWM output signal is enabled immediately. However, the actual PWM waveform is not available on the associated PWM output until its clock source begins its next cycle due to the synchronization of PWMEx and the clock source. An exception to this is when channels are concatenated. Refer to Section 12.4.2.7, “PWM 16-Bit Functions,” for more detail.

NOTE

The first PWM cycle after enabling the channel can be irregular.

On the front end of the PWM timer, the clock is enabled to the PWM circuit by the PWMEx bit being high. There is an edge-synchronizing circuit to guarantee that the clock will only be enabled or disabled at an edge. When the channel is disabled (PWMEx = 0), the counter for the channel does not count.

12.4.2.2 PWM Polarity

Each channel has a polarity bit to allow starting a waveform cycle with a high or low signal. This is shown on the block diagram as a mux select of either the Q output or the $\overline{Q}$ output of the PWM output flip-flop. When one of the bits in the PWMPOL register is set, the associated PWM channel output is high at the beginning of the waveform, then goes low when the duty count is reached. Conversely, if the polarity bit is 0, the output starts low and then goes high when the duty count is reached.

12.4.2.3 PWM Period and Duty

Dedicated period and duty registers exist for each channel and are double buffered so that if they change while the channel is enabled, the change will NOT take effect until one of the following occurs:

- The effective period ends
- The counter is written (counter resets to 0x0000)
- The channel is disabled

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

A change in duty or period can be forced into effect “immediately” by writing the new value to the duty and/or period registers and then writing to the counter. This forces the counter to reset and the new duty and/or period values to be latched. In addition, because the counter is readable it is possible to know where the count is with respect to the duty value and software can be used to make adjustments.

NOTE

When forcing a new period or duty into effect immediately, an irregular PWM cycle can occur.

Depending on the polarity bit, the duty registers will contain the count of either the high time or the low time.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000B Reserved	R	0	0	0	0	0	0	0	0
	W								
0x000C TIE	R	C7I	C6I	C5I	C4I	C3I	C2I	C1I	C0I
	W								
0x000D TSCR2	R	TOI	0	0	0	TCRE	PR2	PR1	PR0
	W								
0x000E TFLG1	R	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
	W								
0x000F TFLG2	R	TOF	0	0	0	0	0	0	0
	W								
0x0010–0x0017 Reserved	R	0	0	0	0	0	0	0	0
	W								
0x0018–0x001F TCxH–TCxL	R	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
	W								
	R	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	W								
0x0020 PACTL	R	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
	W								
0x0021 PAFLG	R	0	0	0	0	0	0	PAOVF	PAIF
	W								
0x0022 PACNTH	R	PACNT15	PACNT14	PACNT13	PACNT12	PACNT11	PACNT10	PACNT9	PACNT8
	W								
0x0023 PACNTL	R	PACNT7	PACNT6	PACNT5	PACNT4	PACNT3	PACNT2	PACNT1	PACNT0
	W								
0x0024–0x002F Reserved	R								
	W								

= Unimplemented or Reserved

Figure 13-5. TIM16B4CV1 Register Summary (continued)

NOTE

BDM should not be entered from a breakpoint unless the ENABLE bit is set in the BDM. Even if the ENABLE bit in the BDM is cleared, the CPU actually executes the BDM firmware code. It checks the ENABLE and returns if ENABLE is not set. If the BDM is not serviced by the monitor then the breakpoint would be re-asserted when the BDM returns to normal CPU flow.

There is no hardware to enforce restriction of breakpoint operation if the BDM is not enabled.

When program control returns from a tagged breakpoint through an RTI or a BDM GO command, it will return to the instruction whose tag generated the breakpoint. Unless breakpoints are disabled or modified in the service routine or active BDM session, the instruction will be tagged again and the breakpoint will be repeated. In the case of BDM breakpoints, this situation can also be avoided by executing a TRACE1 command before the GO to increment the program flow past the tagged instruction.

16.4.1.4 Using Comparator C in BKP Mode

The original BKP_ST12_A module supports two breakpoints. The DBG_ST12_A module can be used in BKP mode and allow a third breakpoint using comparator C. Four additional bits, BKCEN, TAGC, RWCEN, and RWC in DBGC2 in conjunction with additional comparator C address registers, DBGCCX, DBGCCH, and DBGCCL allow the user to set up a third breakpoint. Using PAGSEL in DBGCCX for expanded memory will work differently than the way paged memory is done using comparator A and B in BKP mode. See Section 16.3.2.5, “Debug Comparator C Extended Register (DBGCCX),” for more information on using comparator C.

16.4.2 DBG Operating in DBG Mode

Enabling the DBG module in DBG mode, allows the arming, triggering, and storing of data in the trace buffer and can be used to cause CPU breakpoints. The DBG module is made up of three main blocks, the comparators, trace buffer control logic, and the trace buffer.

NOTE

In general, there is a latency between the triggering event appearing on the bus and being detected by the DBG circuitry. In general, tagged triggers will be more predictable than forced triggers.

16.4.2.1 Comparators

The DBG contains three comparators, A, B, and C. Comparator A compares the core address bus with the address stored in DBGCAH and DBGCAL. Comparator B compares the core address bus with the address stored in DBGCBH and DBGCBL except in full mode, where it compares the data buses to the data stored in DBGCBH and DBGCBL. Comparator C can be used as a breakpoint generator or as the address comparison unit in the loop1 mode. Matches on comparator A, B, and C are signaled to the trace buffer

Table A-26. Expanded Bus Timing Characteristics (5V Range)

Conditions are 4.75V < VDDX < 5.25V, Junction Temperature -40˚C to +140˚C, C_{LOAD} = 50pF							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	P	Frequency of operation (E-clock)	f_o	0	—	25.0	MHz
2	P	Cycle time	t_{cyc}	40	—	—	ns
3	D	Pulse width, E low	PW_{EL}	19	—	—	ns
4	D	Pulse width, E high[1]	PW_{EH}	19	—	—	ns
5	D	Address delay time	t_{AD}	—	—	8	ns
6	D	Address valid time to E rise (PW_{EL}–t_{AD})	t_{AV}	11	—	—	ns
7	D	Muxed address hold time	t_{MAH}	2	—	—	ns
8	D	Address hold to data valid	t_{AHDS}	7	—	—	ns
9	D	Data hold to address	t_{DHA}	2	—	—	ns
10	D	Read data setup time	t_{DSR}	13	—	—	ns
11	D	Read data hold time	t_{DHR}	0	—	—	ns
12	D	Write data delay time	t_{DDW}	—	—	7	ns
13	D	Write data hold time	t_{DHW}	2	—	—	ns
14	D	Write data setup time[1] (PW_{EH}–t_{DDW})	t_{DSW}	12	—	—	ns
15	D	Address access time[1] (t_{cyc}–t_{AD}–t_{DSR})	t_{ACCA}	19	—	—	ns
16	D	E high access time[1] (PW_{EH}–t_{DSR})	t_{ACCE}	6	—	—	ns
17	D	Non-multiplexed address delay time	t_{NAD}	—	—	6	ns
18	D	Non-muxed address valid to E rise (PW_{EL}–t_{NAD})	t_{NAV}	14	—	—	ns
19	D	Non-multiplexed address hold time	t_{NAH}	2	—	—	ns
20	D	Chip select delay time	t_{CSD}	—	—	16	ns
21	D	Chip select access time[1] (t_{cyc}–t_{CSD}–t_{DSR})	t_{ACCS}	11	—	—	ns
22	D	Chip select hold time	t_{CSH}	2	—	—	ns
23	D	Chip select negated time	t_{CSN}	8	—	—	ns
24	D	Read/write delay time	t_{RWD}	—	—	7	ns
25	D	Read/write valid time to E rise (PW_{EL}–t_{RWD})	t_{RWV}	14	—	—	ns
26	D	Read/write hold time	t_{RWH}	2	—	—	ns
27	D	Low strobe delay time	t_{LSD}	—	—	7	ns
28	D	Low strobe valid time to E rise (PW_{EL}–t_{LSD})	t_{LSV}	14	—	—	ns
29	D	Low strobe hold time	t_{LSH}	2	—	—	ns
30	D	NOACC strobe delay time	t_{NOD}	—	—	7	ns
31	D	NOACC valid time to E rise (PW_{EL}–t_{NOD})	t_{NOV}	14	—	—	ns
32	D	NOACC hold time	t_{NOH}	2	—	—	ns
33	D	IPIPO[1:0] delay time	t_{P0D}	2	—	7	ns
34	D	IPIPO[1:0] valid time to E rise (PW_{EL}–t_{P0D})	t_{P0V}	11	—	—	ns
35	D	IPIPO[1:0] delay time[1] (PW_{EH}-t_{P1V})	t_{P1D}	2	—	25	ns
36	D	IPIPO[1:0] valid time to E fall	t_{P1V}	11	—	—	ns

[1] Affected by clock stretch: add N x t_{cyc} where N=0,1,2 or 3, depending on the number of clock stretches.