CorePWM Datasheet

DirectCore

Product Summary

Intended Use

- General Purpose Pulse Width Modulation (PWM) Module for Motor Control, Tone Generation, Battery Charging, Heating Elements, and Digital-to-Analog Conversions

Key Features

- Low Cost PWM Solution with up to 8 Separate PWM Digital Outputs, Configurable via a Register Interface
- All PWM Outputs Are Double-Edge Controlled
- Edge Control Based on a Configurable 8-Bit PWM Period with 8-Bit Prescaler Value and 0 to 100% Duty Cycle Capability
- Set High, Set Low, and Toggle Edge-Control Modes
- PWM Configuration Updates Are Synchronized to the Beginning of the PWM Period, Preventing Erroneous Pulse Generation
- Interrupts Generated at Each Edge of the PWM Outputs
- Can Be Programmed on the Fly from a Microcontroller, such as Core8051, CoreMP7, or the Fusion Backbone
- When Combined with Fusion and a Microcontroller, Can Perform a Closed-Loop Control Function

Supported Families

- Fusion
- ProASIC®3/E
- ProASIC$^{\text{PLUS}}$®
- Axcelerator®
- RTAX-S

Core Deliverables

- Evaluation Version
  - Compiled RTL Simulation Model Fully Supported in Actel Libero® Integrated Design Environment (IDE)
- Netlist Version
  - Structural Verilog and VHDL Netlists (with and without I/O pads) Compatible with Actel Designer Software Place-and-Route Tool
  - Compiled RTL Simulation Model Fully Supported in Actel Libero IDE
- RTL Version
  - Verilog and VHDL Core Source Code
  - Core Synthesis Scripts
- Testbench (Verilog and VHDL)

Synthesis and Simulation Support

- Directly Supported within Actel Libero IDE
- Synthesis: Synplicity®, Synopsys®, Mentor Graphics®
- Simulation: OVI-compliant Verilog Simulators and Vital-Compliant VHDL Simulators

Core Verification

- Comprehensive Actel-Developed VHDL and Verilog Testbenches
- User Can Easily Modify Testbench, Using Existing Format to Add Custom Tests

Table of Contents

General Description

The CorePWM (Pulse Width Modulation) macro generates up to eight general purpose PWM signals, as shown in Figure 1. CorePWM includes the following blocks: Timebase Generation, PWM Generation, and Register Interface.

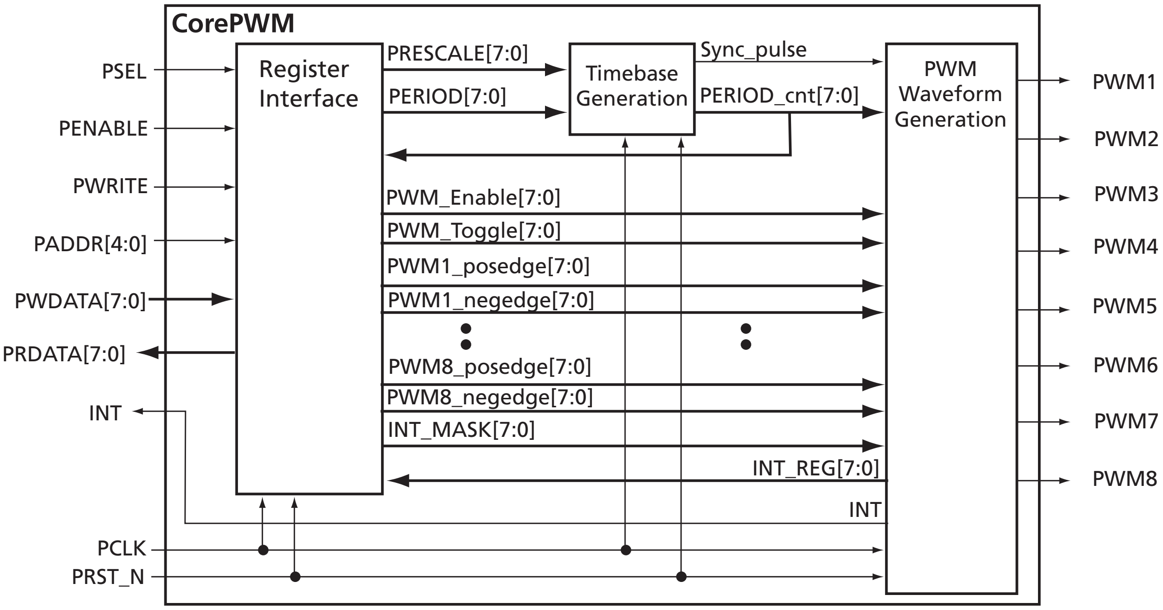


Figure 1 • **CorePWM Block Diagram**

The Register Interface block connects to a standard 8-bit microcontroller for PWM register configuration and updating. Descriptions for all registers are given in Table 6 on page 8. The core uses a shadow register so that PWM waveform updates occur only at the beginning of a PWM period. Interrupts are generated for the microcontroller at each PWM edge and are stored in an Interrupt Register (an ORed interrupt output signal is also available to the microcontroller).

The Timebase Generation block accepts PRESCALE and PERIOD register values and produces a PERIOD count from 0 to 255. The number of system clocks between PERIOD counts is equal to the PRESCALE value.

The PWM Waveform Generation block takes the input period count value and compares it with the positive and negative edge register values. When the count value is equal to any of these edge registers, the respective PWM output waveform is set to the correct high/low/toggle value, and the interrupt register is updated.

An example PWM waveform configuration is shown in Figure 5 on page 9. The example explains the relationship between the PRESCALE and PERIOD register values, and how to configure the PWM waveforms with a given PRESCALE/PERIOD timebase.

A typical temperature monitor application using CorePWM is shown in Figure 2. In this example, fan speed is controlled by fluctuations in the negative temperature coefficient (NTC) thermistor's resistance. As shown, changes in the input voltage to the voltage monitor port will be converted to a digital value via the analog to digital converter (ADC) and forwarded to an on-chip microcontroller, such as the Core8051 or CoreMP7. The microcontroller algorithm will periodically configure/reconfigure CorePWM registers based on the thermistor value.

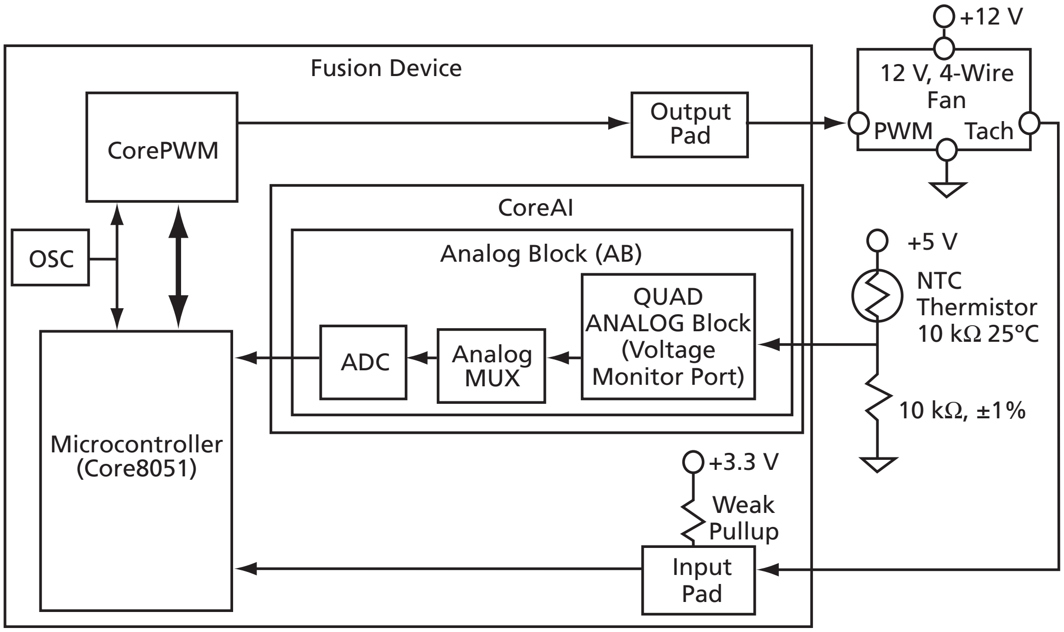


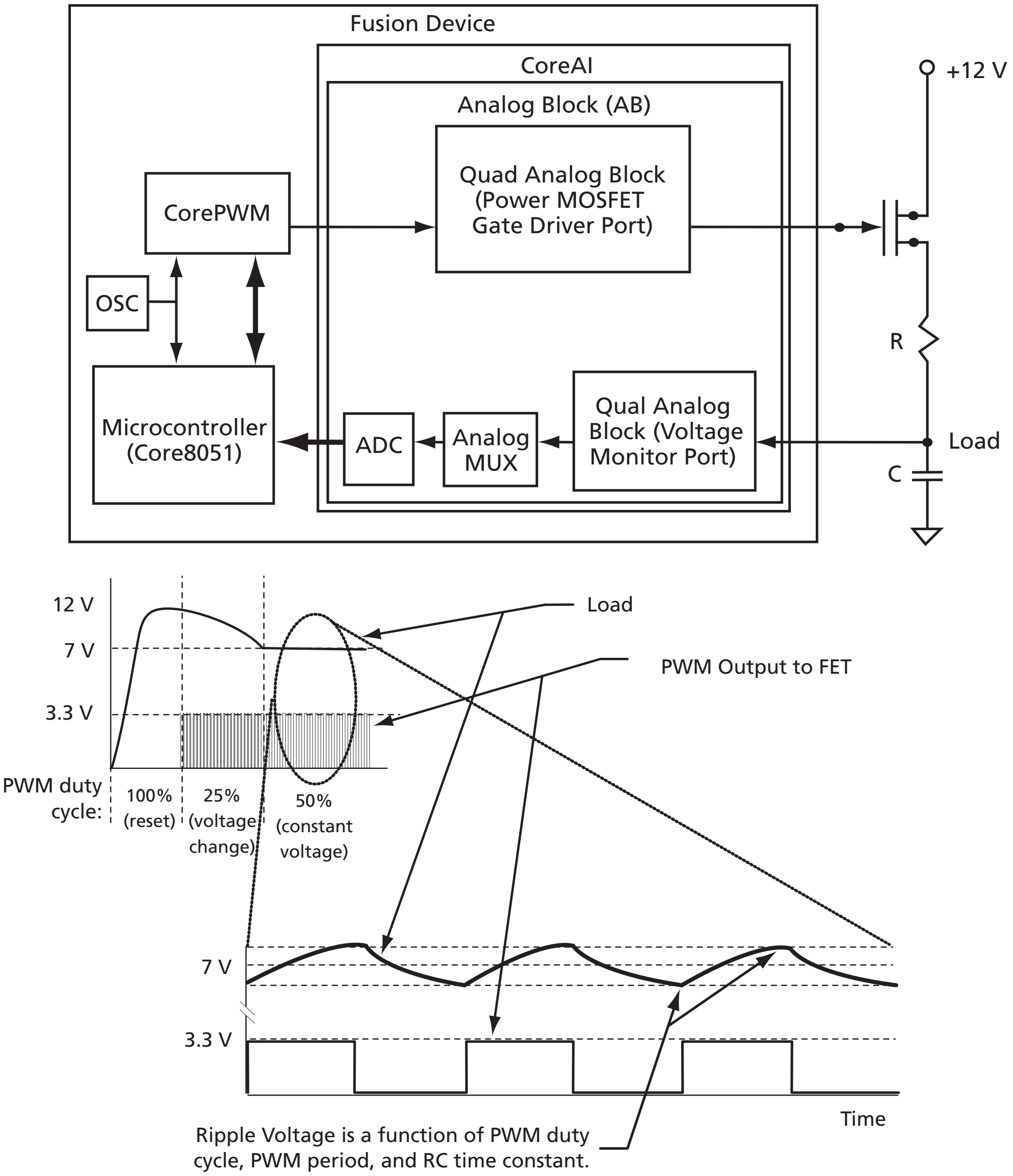
Figure 2 • **Temperature/Voltage Monitor Application Using CorePWM in a Fusion Device**

Alternatively, the CorePWM registers could be controlled with a custom RTL block rather than a microcontroller; the thermistor value coming from the ADC could be used to directly update the duty cycle of the PWM output via a simple transformation and update cycle in the custom RTL block.

A typical DAC application using CorePWM is shown in Figure 3. In this example, the PWM output is averaged to a varying DC voltage. At reset, the PWM duty cycle is 100% and the voltage increases to the rail of 12 volts. A PWM duty cycle less than 50% reduces the voltage level, whereas a duty cycle greater than 50% increases the voltage level. A 50% duty cycle maintains the current voltage level. The generated ripple voltage is a function of the RC circuit values, the PWM period, and the PWM duty cycle.

The FET is used to increase and decouple the output voltage/current from the Fusion device. The load is monitored and changes to the PWM output are processed via a microcontroller.

As in the previous example, the CorePWM registers could be controlled with a custom RTL block rather than a microcontroller.



Fusion Device
CoreAI
Analog Block (AB)
Quad Analog Block (Power MOSFET Gate Driver Port)
CorePWM
OSC
Microcontroller (Core8051)
ADC
Analog MUX
Qual Analog Block (Voltage Monitor Port)
+12 V
R
Load
C
12 V
7 V
3.3 V
PWM duty cycle:
100% (reset)
25% (voltage change)
50% (constant voltage)
Load
PWM Output to FET
7 V
3.3 V
Time
Ripple Voltage is a function of PWM duty cycle, PWM period, and RC time constant.

Figure 3 • **DAC Application Using CorePWM in a Fusion Device**

CorePWM Device Requirements

CorePWM has been implemented in several Actel device families. A summary of the implementation data is listed in Table 1, Table 2, and Table 3.

Table 1 • **CorePWM Device Utilization and Performance (Minimum Configuration)**

Family	Tiles			Utilization		Performance
	Sequential	Combinatorial	Total	Device	Total	
Fusion	20	67	87	AFS600	1%	120 MHz
ProASIC3/E	20	68	88	M7A3P250	1%	120 MHz
ProASIC^{PLUS}	20	79	99	APA075	3%	90 MHz
Axcelerator	16	28	44	AX250	1%	266 MHz
RTAX-S	16	28	44	RTAX250S	1%	192 MHz

Note: Data in this table were achieved using typical synthesis and layout settings. Top-level parameters/generics were set as follows: PWM_NUM = 1, PWM_FIXED_REG_SEL = 1.

Table 2 • **CorePWM Device Utilization and Performance (Typical Configuration)**

Family	Tiles			Utilization		Performance
	Sequential	Combinatorial	Total	Device	Total	
Fusion	216	434	650	AFS600	5%	108 MHz
ProASIC3/E	216	431	647	M7A3P250	11%	108 MHz
ProASIC^{PLUS}	216	662	878	APA075	29%	88 MHz
Axcelerator	217	281	498	AX250	10%	131 MHz
RTAX-S	217	281	498	RTAX250S	12%	92 MHz

Note: Data in this table were achieved using typical synthesis and layout settings. Top-level parameters/generics were set as follows: PWM_NUM = 4, PWM_FIXED_REG_SEL = 0.

Table 3 • **CorePWM Device Utilization and Performance (Maximum Configuration)**

Family	Tiles			Utilization		Performance
	Sequential	Combinatorial	Total	Device	Total	
Fusion	370	687	1057	AFS600	8%	100 MHz
ProASIC3/E	370	691	1061	M7A3P250	17%	100 MHz
ProASIC^{PLUS}	370	1115	1485	APA075	48%	76 MHz
Axcelerator	374	427	801	AX250	19%	123 MHz
RTAX-S	374	427	801	RTAX250S	19%	90 MHz

Notes:

1. Data in this table were achieved using typical synthesis and layout settings. Top-level parameters/generics were set as follows: PWM_NUM = 8, PWM_FIXED_REG_SEL = 0.
2. Minimum I/O = 9 (PWM_NUM = 1, PWM_FIXED_REG_SEL = 1)
   Maximum I/O = 35 (PWM_NUM = 8, PWM_FIXED_REG_SEL = 0)

I/O Signal Descriptions

The port signals for the CorePWM macro are defined in Table 4 and illustrated in Figure 4. All signals are either "Input" (input-only) or "Output" (output-only).

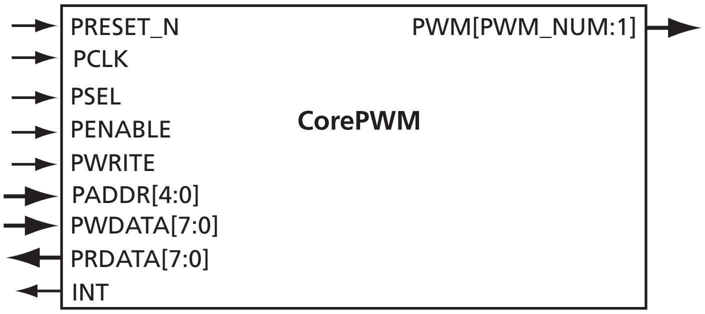


Figure 4 • **CorePWM I/O Signal Diagram**

Table 4 • **CorePWM I/O Signal Descriptions**

Name	Type	Description
System Signals		
PRESET_N	Input	Active low asynchronous reset
PCLK	Input	System clock; all operations and status are synchronous to the rising edge of this signal.
Microcontroller Signals		
PSEL	Input	Select line for CorePWM
PENABLE	Input	Read output enable
PWRITE	Input	Write enable
PADDR[4:0]	Input	Register address; refer to Figure 6 on page 8 and Figure 7 on page 10 for read/ write procedures.
PWDATA[7:0]	Input	Write address/data input
PRDATA[7:0]	Output	Read data output
INT	Output	ORed interrupt signal; set to '1' at each PWM output transition.
PWM Signals		
PWM[PWM_NUM:1]	Output	Pulse width modulation output(s); up to 8 PWM outputs, see PWM_NUM parameter/generic description in Table 5 on page 7.

Note: All signals are active high (logic 1) unless otherwise noted by an "_N" at the end of the signal name.

Parameter/Generic Descriptions

CorePWM has parameters (Verilog) and generics (VHDL) for configuring the RTL code, as described in Figure 5. All parameters and generics are integer types.

Table 5 • **CorePWM Parameters/Generic Descriptions**

Name	Description
PWM_NUM	Number of PWM outputs: 1 = 1 PWM output 2 = 2 PWM outputs 3 = 3 PWM outputs 4 = 4 PWM outputs 5 = 5 PWM outputs 6 = 6 PWM outputs 7 = 7 PWM outputs 8 = 8 PWM outputs
PWM_FIXED_REG_SEL	Fixed register select: 0 = Normal, APB configurable register operation. 1 = Hardwired register operation; values based on the following FIXED_* parameters.
FIXED_PRESCALE	Hardwired PRESCALE register value, from 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PERIOD	Hardwired PERIOD register value, from 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_ENABLE	Hardwired PWM_ENABLE register value, from 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_INT_MASK	Hardwired PWM_INT_MASK register value, from 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM1_POSEDGE	Hardwired POSEDGE1 register value, from 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM1_NEGEDGE	Hardwired POSEDGE1 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM2_POSEDGE	Hardwired POSEDGE2 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM2_NEGEDGE	Hardwired POSEDGE2 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM3_POSEDGE	Hardwired POSEDGE3 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM3_NEGEDGE	Hardwired POSEDGE3 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM4_POSEDGE	Hardwired POSEDGE4 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM4_NEGEDGE	Hardwired POSEDGE4 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM5_POSEDGE	Hardwired POSEDGE5 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM5_NEGEDGE	Hardwired POSEDGE5 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM6_POSEDGE	Hardwired POSEDGE6 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM6_NEGEDGE	Hardwired POSEDGE6 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM7_POSEDGE	Hardwired POSEDGE7 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM7_NEGEDGE	Hardwired POSEDGE7 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM8_POSEDGE	Hardwired POSEDGE8 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**
FIXED_PWM8_NEGEDGE	Hardwired POSEDGE8 register value, 0–255. **Used only if PWM_FIXED_REG_SEL = 1.**

Register Descriptions

All registers are eight bits wide. Full register descriptions are given in Table 6.

Table 6 • **CorePWM Register Definition**

Register Name	PADDR[4:0]	Description	Type	Default
PRESCALE	0h00	The system clock cycle is multiplied by the PRESCALE value, resulting in the minimum PERIOD count timebase.	R/W	0h08
PERIOD	0h01	The PRESCALE value is multiplied by the PERIOD value, yielding the PWM waveform cycle. Example: system clock = 40 ns, PRESCALE register = 255, PERIOD register = 127. The PWM waveforms will repeat every 40 ns × 256 × 128 = 1.31 ms. The resolution of the PWM waveforms will be 1.31ms ÷128 = 10.23 µs.	R/W	0h08
PWM_ENABLE	0h02	'1' enables each PWM output.	R/W	0h00
INT_MASK	0h03	'1' masks each respective bit in the INTERRUPT register.	R/W	0h00
INTERRUPT	0h04	Each interrupt bit is set to '1' at either edge of a PWM output.	Clear on Read	0h00
PWM1_POSEDGE	0h05	Sets positive edge of each PWM1 output with respect to the PERIOD resolution	R/W	0h00
PWM1_NEGEDGE	0h06	Sets negative edge of each PWM1 output with respect to the PERIOD resolution	R/W	0h00
PWM2_ POSEDGE	0h07	Sets positive edge of each PWM2 output with respect to the PERIOD resolution	R/W	0h00
PWM2_ NEGEDGE	0h08	Sets negative edge of each PWM2 output with respect to the PERIOD resolution	R/W	0h00
PWM3_ POSEDGE	0h09	Sets positive edge of each PWM3 output with respect to the PERIOD resolution	R/W	0h00
PWM3_ NEGEDGE	0h0A	Sets negative edge of each PWM3 output with respect to the PERIOD resolution	R/W	0h00
PWM4_ POSEDGE	0h0B	Sets positive edge of each PWM4 output with respect to the PERIOD resolution	R/W	0h00
PWM4_ NEGEDGE	0h0C	Sets negative edge of each PWM4 output with respect to the PERIOD resolution	R/W	0h00
PWM5_ POSEDGE	0h0D	Sets positive edge of each PWM5 output with respect to the PERIOD resolution	R/W	0h00
PWM5_ NEGEDGE	0h0E	Sets negative edge of each PWM5 output with respect to the PERIOD resolution	R/W	0h00
PWM6_ POSEDGE	0h0F	Sets positive edge of each PWM6 output with respect to the PERIOD resolution	R/W	0h00
PWM6_ NEGEDGE	0h10	Sets negative edge of each PWM6 output with respect to the PERIOD resolution	R/W	0h00
PWM7_ POSEDGE	0h11	Sets positive edge of each PWM7 output with respect to the PERIOD resolution	R/W	0h00
PWM7_ NEGEDGE	0h12	Sets negative edge of each PWM7 output with respect to the PERIOD resolution	R/W	0h00
PWM8_ POSEDGE	0h13	Sets positive edge of each PWM8 output with respect to the PERIOD resolution	R/W	0h00
PWM8_ NEGEDGE	0h14	Sets negative edge of each PWM8 output with respect to the PERIOD resolution	R/W	0h00

Note: 0d = decimal; 0h = hexadecimal; 0b = binary

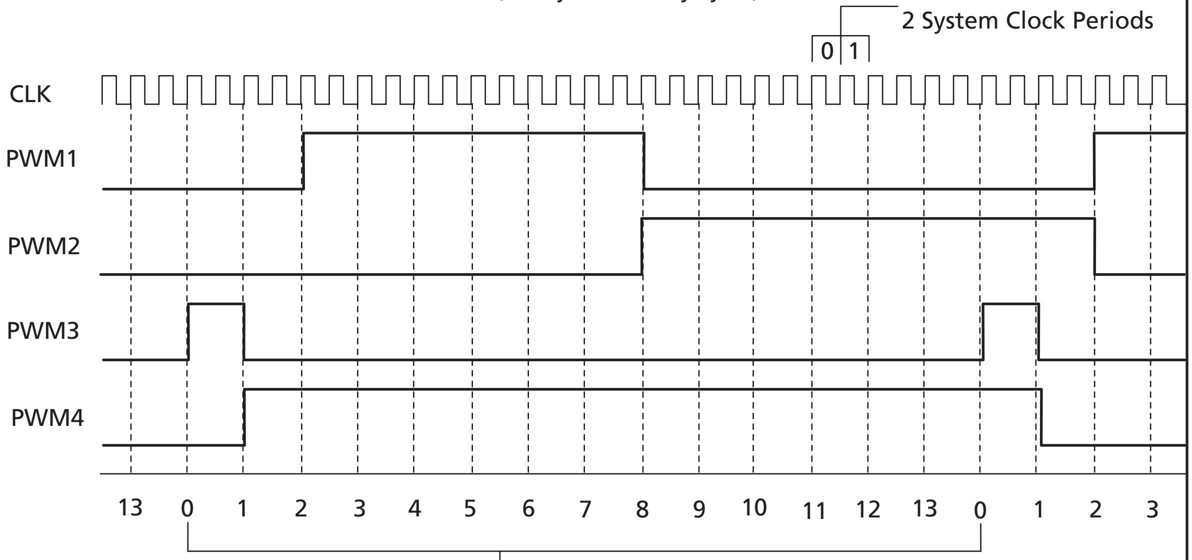
Example Configuration

Figure 5 demonstrates how several register configurations affect PWM output waveform generation.

Configuring the registers as follows will yield the PWM waveforms below, based on a 25 MHz system clock (40 ns system clock period).

Note: 0d = decimal; 0h = hexidecimal; 0b = binary

PRESCALE = 0d1	PWM period granularity [PWM_PG] = 40 ns × 2 = 80 ns
PERIOD = 0d13	PWM period = PWM_PG × 14 = 80 ns × 14 = 1.12 µs
PWM_EN = 0h0F	Enable PWM signals 1, 2, 3, and 4.
INT_MASK = 0h73E	Mask all but the bit0 interrupt for PWM 1
PWM1_POSEDGE = 0d2	PWM1 duty cycle = 43%
PWM1_NEGEDGE = 0d7	
PWM2_POSEDGE = 0d8	PWM2 duty cycle = 57%
PWM2_NEGEDGE = 0d2	
PWM3_POSEDGE = 0d0	PWM3 duty cycle = 7.1%
PWM3_POSEDGE = 0d1	
PWM4_POSEDGE = 0d1	Toggle PWM4 output
PWM4_POSEDGE = 0d1	(always 50% duty cycle)



PRESCALE Value of 1 =
2 System Clock Periods

Period Value of 13 with Prescale Value of 1 =
14 × 2 System Clock Periods per PWM Period

Figure 5 • **CorePWM Waveform Generation Example**

Register Access

Figure 6 and Figure 7 depict typical write cycle and read cycle timing relationships relative to the system clock.

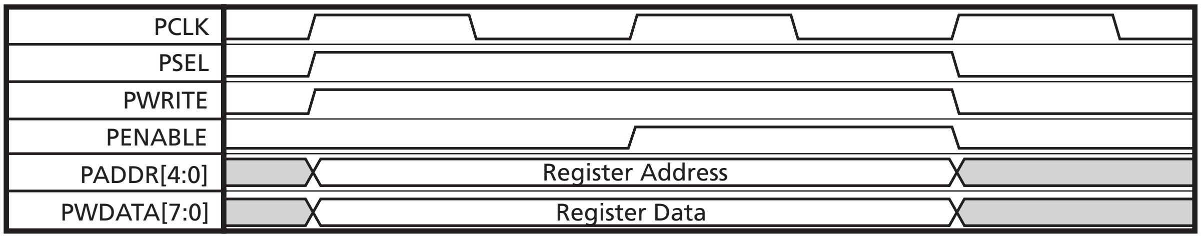


Figure 6 • **Data Write Cycles**

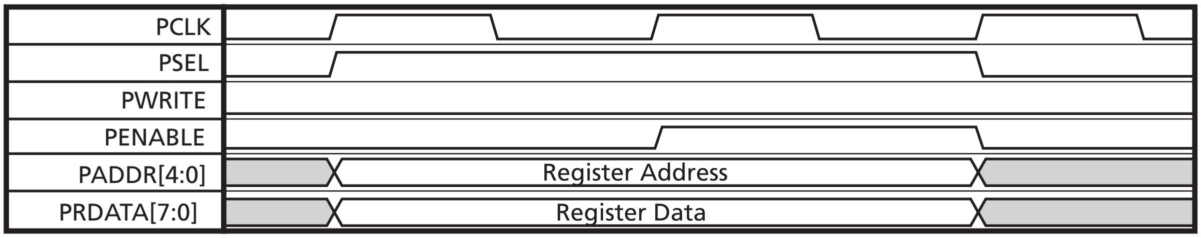


Figure 7 • **Data Read Cycles**

Duty Cycle Calculator

A Duty Cycle Calculator assists in calculating the PWM POSEDGE and NEGEDGE register values, given a requested duty cycle. This is provided on the Actel website as a downloadable Excel spreadsheet:

http://www.actel.com/documents/duty_cycle_calc.zip.

Ordering Information

Order CorePWM through your local Actel sales representative. Use the following naming convention when ordering: CorePWM-XX, where XX is listed in Table 7.

Table 7 • **Ordering Codes**

XX	**Description**
EV	Evaluation version
SN	Single-use Netlist for use on Actel devices
AN	Netlist for unlimited use on Actel devices
AR	RTL for unlimited use on Actel devices
UR	RTL for unlimited use and not restricted to Actel devices

Datasheet Categories

In order to provide the latest information to designers, some datasheets are published before data has been fully characterized. Datasheets are designated as "Product Brief," "Advanced," and "Production." The definitions of these categories are as follows:

Product Brief

The product brief is a summarized version of an advanced or production datasheet containing general product information. This brief summarizes specific device and family information for unreleased products.

Advanced

This datasheet version contains initial estimated information based on simulation, other products, devices, or speed grades. This information can be used as estimates, but not for production.

Unmarked (production)

This datasheet version contains information that is considered to be final.

www.actel.com

Actel Corporation

2061 Stierlin Court
Mountain View, CA
94043-4655 USA
Phone 650.318.4200
Fax 650.318.4600

Actel Europe Ltd.

Dunlop House, Riverside Way
Camberley, Surrey GU15 3YL
United Kingdom
Phone +44 (0) 1276 401 450
Fax +44 (0) 1276 401 490

Actel Japan
www.jp.actel.com
EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan
Phone +81.03.3445.7671
Fax +81.03.3445.7668

Actel Hong Kong
www.actel.com.cn
Suite 2114, Two Pacific Place
88 Queensway, Admiralty
Hong Kong
Phone +852 2185 6460
Fax +852 2185 6488