Quad PWM

v1.0

QuadPWM_1

Quad PWM

clock   pwm1
         pwm2
         pwm3
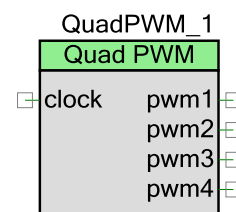   pwm4

Features

- Generates 4x 8-bit PWM outputs from a single UDB

- Easy APIs to initialize and set duty cycles

- Perfect for Low-end PSoC 4 devices

General Description

The Quad PWM component provides four 8-bit PWM outputs using a single Universal Digital Block (UDB). The current PWM component available in PSoC Creator only provides 2 PWM output signals per UDB.

All four PWM signals share the same period, which depends on the input clock frequency. The duty cycles are set directly in the customizer or by the SetDutyCycle() API.

When to Use this Component

The Quad PWM component is recommended on low-end PSoC 4 devices that do not have many UDB resources; or designs that lack of hardware resources and require many PWM signals.

Input/Output Connections

This section describes the various input and output connections for the Quad PWM component.

Note All signals are active high unless otherwise specified.

Input	May Be Hidden	Description
clock	N	The clock input defines the PWM output frequency.

Output	May Be Hidden	Description
pwm1	N	The pwm1 is the first pulse-width modulated output. Its duty cycle is set in the customizer DutyCycle1 parameter or by SetDutyCyle().
pwm2	N	The pwm2 is the second pulse-width modulated output. Its duty cycle is set in the customizer DutyCycle2 parameter or by SetDutyCyle().
pwm3	N	The pwm3 is the third pulse-width modulated output. Its duty cycle is set in the customizer DutyCycle3 parameter or by SetDutyCyle().

Output	May Be Hidden	Description
pwm4	N	The pwm4 is the forth pulse-width modulated output. Its duty cycle is set in the customizer DutyCycle4 parameter or by SetDutyCyle().

Component Parameters

Drag a PWM component onto your design and double click it to open the **Configure** dialog.

DutyCycleX

Define the duty cycle of pwmX output. The value range is from 0 (0% duty cycle) to 255 (100% duty cycle).

Clock Selection

There is no internal clock in this component. You must attach a clock source. The PWM frequency equation is calculated as:

$$\text{PWM Freq (Hz)} = \text{Clock Freq (Hz)} / 1024$$

Application Programming Interface

Application Programming Interface (API) routines allow you to configure the component using software. The following table lists and describes the interface to each function. The subsequent sections cover each function in more detail.

By default, PSoC Creator assigns the instance name "QuadPWM_1" to the first instance of a component in a given design. You can rename it to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol. For readability, the instance name used in the following table is "QuadPWM".

Functions

Function	Description
QuadPWM_Init()	Initializes the PWM based on the duty cycles provided in the customizer.
QuadPWM_SetDutyCycle()	Set the duty cycle for the given PWM output.

void QuadPWM_Init(void)

Description:	Initializes the PWM based on the duty cycles provided in the customizer.
Parameters:	None
Return Value:	None
Side Effects:	None

void QuadPWM_SetDutyCycle(uint32 pwm, uint32 dutyCycle)

Description:	Set the duty cycle for the given PWM output
Parameters:	pwm: PWM output to be set (from 1 to 4) dutyCycle: new duty cycle value (from 0 to 255)
Return Value:	None
Side Effects:	None

API Constants

Function	Description
QuadPWM_MAX_DUTY_CYCLE	Maximum duty cycle supported
QuadPWM_MIN_DUTY_CYCLE	Minimum duty cycle supported
QuadPWM_PWM_1	Index to PWM 1 output
QuadPWM_PWM_2	Index to PWM 2 output
QuadPWM_PWM_3	Index to PWM 3 output
QuadPWM_PWM_4	Index to PWM 4 output
QuadPWM_DP_INIT	Initialization value for the datapath

Sample Firmware Source Code

This section shows an example of the main loop using this component.

```c
#include <project.h>

int main()
{
    uint8 counter = 0;

    QuadPWM_Init();

    for(;;)
    {
        counter++;
        QuadPWM_SetDutyCycle(QuadPWM_PWM_1, counter);
        CyDelay(1000);
    }
}
```

Resources

The UDB Implementation utilizes the following resources.

Configuration	Resource Type					
	Datapath Cells	Macrocells	Status Cells	Control Cells	DMA Channels	Interrupts
4x 8-bit PWM	1	6	0	0	–	–

DC and AC Electrical Characteristics

The following values indicate expected performance and are based on initial characterization data.

SRAM Scrubber DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
	TBD					

SRAM Scrubber AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
PWMFreq	Maximum PWM frequency				46.875	KHz

Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
**	Initial release	Initial release