PWM user manual

Features

- Work as one PWM or one timer.
- 16 bits main counter.
- PWM/Timer can choose between Wishbone interface clock or external clock as working clock.
- PWM can choose between dedicated duty cycle input or internal register as source of duty cycle.
- Duty cycle and period can be changed at runtime.
- Hosted through Wishbone slave interface.
- Working clock can be down clocked to at most 1/65535 or $1/(2^{16}-1)$ of original frequency.
- Period register also serves as timer target register when module is in timer mode.

Operations

All operations of this PWM core is controlled by manipulating registers inside the core through WISHBONE slave interface. The interface is Wishbone B4 compliant.

PWM mode

When mode selection bit (ctrl bit 1) is set the core is in PWM mode. Port o_pwm will output PWM signal continuously when main counter enable bit (ctrl bit 2) and PWM signal output enable bit (ctrl bit 4) are set. It's period is stored in register period. Signals coming from register DC or port i_DC can be used as duty cycle. Both period and duty cycle are defined by number of clock cycles. When main counter enable bit is reset the main counter will stop running and o_pwm will stop changing until the main counter enable bit is set again or the counter logic is reset by external reset or main counter reset bit (ctrl bit 7).

Timer mode

When mode selection bit (ctrl bit 1) is reset the core is in timer mode. Right after main counter enable bit (ctrl bit 2) is set the main counter starts to increment. When the number stores in main counter reaches or exceeds the number stored in register period an interrupt is generated and output through o_pwm. Then the main counter will be reset. If continuous run (ctrl bit 3) is selected and main counter

enable bit is set the main counter will start to increment again no mater whether interrupt is cleared or not. If not main counter won't start again until interrupt is cleared.

Down clocking

Clock for incrementing the main counter can be down clocked before entering main counter logic. Frequency can be divided by odd or even number. Divisor is stored in 16-bit register. If divisor equals 0 or 1, no down clocking is performed. Else, the output frequency is original frequency/divisor.

Main counter reset

Main counter can be reset by system reset or by setting ctrl bit 7. When changing from PWM mode to timer mode main counter must be reset before timer starts.

Interrupt

In timer mode, when main counter exceeds period an interrupt will be generated. A low to high transition on port o_pwm and set of ctrl bit 5 indicate the interrupt request. Interrupt is cleared by writing 0 to ctrl bit 5.

Control registers

All 4 control registers are accessible through Wishbone slave interface.

Name	Address	Width	Access	Description
Ctrl	Base + 0	8	R/W	Core's operation control register
Divisor	Base + 2	16	R/W	Stores divisor for down clocking.
Period	Base + 4	16	R/W	Stores period defined by number of clock cycles. Also used as timer target when working in timer mode.
DC	Base + 6	16	R/W	Stores duty cycle defined by number of clock cycles during the pulse.

Ctrl/Operation control register

Bit	Reset	Description
Bit 0	0	When set, external clock is chosen for PWM/timer. When cleared, wb clock is used for PWM/timer.
Bit 1	0	When set, PWM is enabled. When cleared, timer is enabled.
Bit 2	0	When set, PWM/timer starts. When cleared, PWM/timer stops.
Bit 3	0	When set, timer runs continuously. When cleared, timer runs one time.
Bit 4	0	When set, o_pwm enabled.
Bit 5	0	timer interrupt bit When it is written with 0, interrupt request is cleared.
Bit 6	0	When set, a 16-bit external signal i_DC is used as duty cycle. When cleared, register DC is used.
Bit 7	0	When set, main counter resets, o_pwm and crtl bit 5 are also reset. When changing from PWM mode to timer mode main counter reset is needed before timer starts.

Interfaces

Wishbone Slave Interface

It's Wishbone B4 compliant.

Name	Name in Wishbone B4	Size	Direction	Description
i_clk	CLK_I	1	input	Clock input
i_rst	RST_I	1	input	Reset input
i_wb_cyc	CYC_I	1	input	Indicates valid bus cycle (core select)
i_wb_stb	STB_I	1	input	Indicates valid data transfer cycle
i_wb_we	WE_I	1	input	Write transaction when asserted high
i_wb_adr	ADR_I	16	input	Address input
i_wb_data	DAT_I	16	input	Data input
o_wb_ack	ACK_O	1	output	Acknowledgment output (indicates normal transaction termination)
o_wb_data	DATA_O	16	output	Data output

Other interfaces

Name	Size	Direction	Description

i_extclk	1	Input	External clock input
i_DC	16	Input	Dedicated duty cycle input
i_DC_valid	1	Input	Indicate valid signal on i_DC
o_pwm	1	output	PWM signal or interrupt output

Optional Features (Extra Points):

- Multi-Channel PWM: Up to 4 outputs with separate Period/DC.
- Error Handling: Handle issues like DC > Period or invalid Divisor.
- Multi-Clock Domains: The design Support two clock domains when using the external clock (e.g., 50 MHz + external) without glitches.

If You Get Stuck:

- **Down Clocking:** If it is tough, skip it and use 50 MHz clock. This lowers Functionality score but keeps your design running. Try a simple counter instead of prescaler.