Programmable Interval Timer (PIT) Specification

Author: Robert Hayes

rehayes@opencores.org

Rev. 0.1

May 22, 2009

This page has been intentionally left blank.

Revision History

Rev.	Date	Author	Description
0.1	14/04/09	Robert Hayes	First draft release
0.2	22/05/09	Robert Hayes	Changed the recommended connection of ext_sync_i for a PIT module that is only to be used in Master mode from "tie low" to "tie to cnt_sync_o". Added wb_sel_i to the signals lists. Correct minor typo's.

Contents

1

Introduction

The Programmable Interval Timer Module, PIT, is a simple timer to generate a periodic signal for a microcontroller system. This signal may be used for a variety of purposes such as triggering the start of an Analog to Digital or Digital to Analog conversion, as a periodic system interrupt, or to synchronize the start of various other hardware processes.

The PIT RTL code is parameterized so that every module instantiation can be customized to meet system requirements without wasting unneeded resources.

FEATURES

- **16 bit Main Counter with programmable modulo**
- **15 bit Prescale Counter with programmable modulo selections**
- **Slave mode for synchronizing multiple PIT modules**
- **Interrupt or bit-polling**
- **Static synchronous design**
- **Fully synthesizable**
- **Parameterized so each instance can be optimized for size**

2

Architecture

The PIT core is built around four primary blocks; the WISHBONE Interface, the Control Registers, the Prescale Counter and the Main Counter.
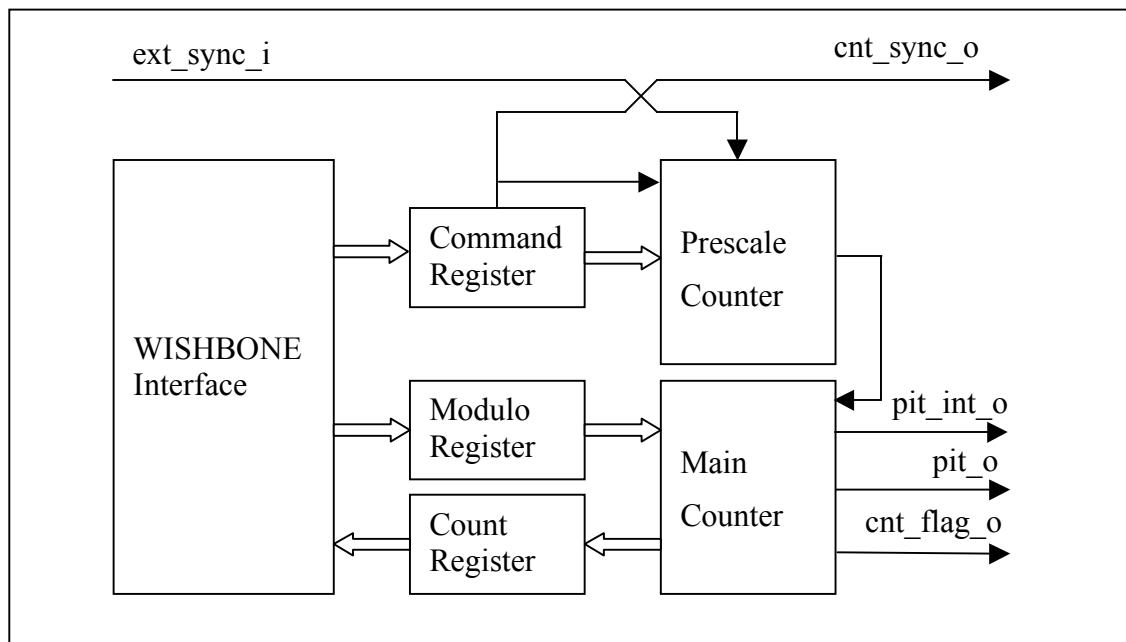
Fig. 2.1 Internal structure PIT Core

2.1 WISHBONE Interface

The WISHBONE Interface isolates the PIT functionality from the WISHBONE bus. This interface takes the bus specific signals and generates a generic set of control signals to drive the PIT control registers. Isolating the WISHBONE bus should help promote PIT module reusability by localizing the scope of changes needed to retarget the PIT module to another bus environment.

2.2 Control Registers

The Command Register, Modulo Register and Count Register are combined into a single module that controls the programmable functions of the PIT. Various bits in the Command Register define the basic operating mode and function enables of the PIT.

2.3 Prescale Counter

The Prescale counter extends the range of the main counter. Using the Prescale counter reduces the timing granularity in terms of master clocks over which the period of the output interrupts can be controlled. As an example if the Prescale Counter modulo is set for a value of 10 then counts of 90 or 100 of the master clock could be decoded but not a master clock count of 95.

2.4 Main Counter

The Main Counter increments at a rate determined by the setting of the Prescale modulo value and the master clock frequency. Each time the Main Counter reaches the modulo value the PIT output signals are activated and the PIT FLAG status bit is set.

3

Operation

The PIT module is a very simple counter that can be controlled by commands from the WISHBONE bus. The output from the module is a pulse that is one wb_clk_i period wide at a frequency of:

$$pit\_o = wb\_clk\_i / (MOD\_VAL * PRE\_SCALR)$$

Where MOD_VAL is a 16-bit register accessed from the WISHBONE bus and PRE_SCLAR is a four-bit value accessed from the WISHBONE bus that is decoded into 16 possible multiplier values.

Figure 3.1 is an illustration for the case of a PIT module operating with MOD_VAL=5 and PRE_SCLR=0. The start of the counters incrementing is determined by the setting of the CNT_EN bit in the control register. If the PIT module is operating in slave mode then the CNT_EN signal shown in the figure is actually the [ext_sync_i] signal that is controlled by the CNT_EN bit of the master PIT module. Note that the fall time of the signal [cnt_flag_o] is not explicitly defined because it is dependent on receiving a command from the WISHBONE bus to reset the FLAG register.
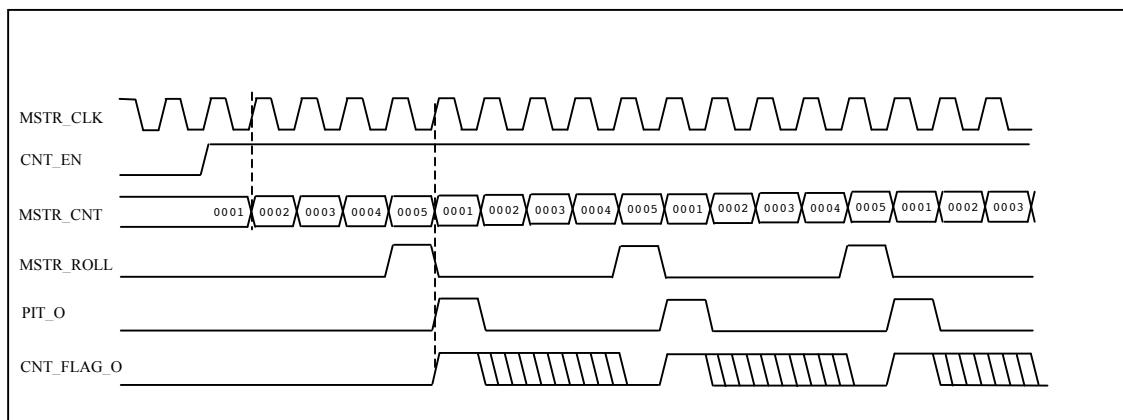
Fig. 3.1 PIT Signals**, MOD_VAL=5**

Figure 3.2 is an illustration of the timing relationships of a PIT module operating with MOD_VAL=4 and PRE_SCLR=1(Divide by 2).
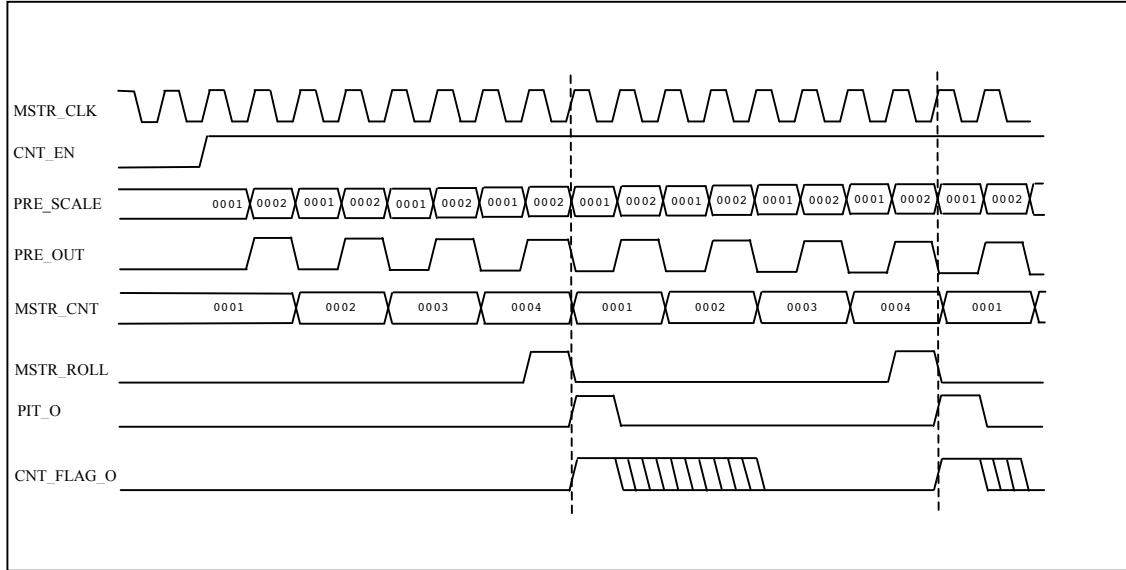
Fig. 3.2 PIT Signals, MOD_VAL=4, PRE_SCLR=1

The recommended software procedure for using the PIT Module:

1. Initialize PIT Slave modules

    a) Set MOD_VAL

    b) Set SLAVE, PRE_SCALR, and ENA_INT

2. Initialize PIT Master module

    a) Set MOD_VAL

    b) Set SLAVE, PRE_SCALR, and ENA_INT

    c) Set CNT_EN

4

Registers

The PIT module can be configured through the use of the DWIDTH parameter to have a WISHBONE bus interface with either an 8-bit bus with 8-bit granularity or to use a 16-bit bus with 16-bit granularity. This document shows the resultant address and bit field for both configurations although in an actual instance of the PIT module only one of the pairs of tables will be valid. For an end user or programmer reference it may be best to simplify this document by removing the tables that reference the unused bus configuration.

List of Registers

Name	Address	Width	Access	Description
CNTRL	0x00	16	RW	PIT Control Register
MOD	0x01	16	RW	PIT Modulo Register
CNT	0x02	16	R	PIT Counter Value

Table 1: List of registers, 16 bit data (default DWIDTH=16)

Name	Address	Width	Access	Description
CNTRL_0	0x00	8	RW	PIT Control Register Low
CNTRL_1	0x01	8	RW	PIT Control Register High
MOD_0	0x02	8	RW	PIT Modulo Register Low Byte
MOD_1	0x03	8	RW	PIT Modulo Register High Byte
CNT_0	0x04	8	R	PIT Counter Value Low Byte
CNT_1	0x05	8	R	PIT Counter Value High Byte

Table 1a: List of registers, 8 bit data (DWIDTH=8)

4.1 CNTRL Register

Bit #	Access	Description
15	RW	SLAVE, Slave mode enable bit When set to '1' the Prescaler Counter and the Main Counter are enabled by an external master mode PIT. This allows the slave mode PIT counters to be synchronized to the CNT_EN bit of the master mode PIT module. When set to '0' the PIT counters are enabled by the CNT_EN bit.
14	R	DECADE_CNTR, Returns the value of the DECADE_CNTR parameter
13	R	NO_PRESCALE, Returns the value of the NO_PRESCALE parameter.
12	R	*Reserved*
11:8	RW	PRE_SCALR, Prescale Counter modulo value. Sets the period of the Prescale Counter in bus clocks. See Table 3 If the NO_PRESCALE parameter is set then this field will always return all zeros.
7:3	R	*Reserved*
2	W	FLAG, Main counter rollover status flag. Read - Returns the current status of the rollover status flag and the interrupt output state. Write – When set to '1' clears the FLAG status bit and the interrupt output. When set to '0' there is no effect.
1	W	ENA_INT, Interrupt enable. When set to '1' the interrupt output is enabled. When set to '0' the interrupt output is disabled. Clearing this bit only disables the interrupt output, it does not clear the source of the interrupt which is the FLAG status bit.
0	W	CNT_EN, Prescale and Main Counter enable. Clearing this bit resets the Prescale and Main counter to their default state. This bit has no effect when the SLAVE bit is set.

Reset Value:

       CNTRL: 0000h

Table 2: CNTRL Register Bits

Bit #	Access	Description
7	RW	SLAVE, Slave mode enable bit
		When set to '1' the Prescaler Counter and the Main Counter are enabled by an external master mode PIT. This allows the slave mode PIT counters to be synchronized to the CNT_EN bit of the master mode PIT module.
		When set to '0' the PIT counters are enabled by the CNT_EN bit.
6	R	DECADE_CNTR, Returns the value of the DECADE_CNTR parameter
5	R	NO_PRESCALE, Returns the value of the NO_PRESCALE parameter.
4	R	*Reserved*
3:0	RW	PRE_SCALR, Prescale Counter modulo value.
		Sets the period of the Prescale Counter in bus clocks. See Table 3
		If the NO_PRESCALE parameter is set then this field will always return all zeros.

Reset Value:
CNTRL_1: 40h

Table 2a: CNTRL Register Bits

Bit #	Access	Description
7:3	R	*Reserved*
2	W	FLAG, Main counter rollover status flag.
		Read - Returns the current status of the rollover status flag and the interrupt output state.
		Write – When set to '1' clears the FLAG status bit and the interrupt output. When set to '0' there is no effect.
1	W	ENA_INT, Interrupt enable.
		When set to '1' the interrupt output is enabled.
		When set to '0' the interrupt output is disabled. Clearing this bit only disables the interrupt output, it does not clear the source of the interrupt which is the FLAG status bit.

Bit #	Access	Description
0	W	CNT_EN, Prescale and Main Counter enable. Clearing this bit resets the Prescale and Main counter to their default state. This bit has no effect when the SLAVE bit is set.

Reset Value:
     CNTRL_0: 00h

Table 2b: CNTRL Register Bits

PRE_SCALR	DECADE_ENA = 0	DECADE_ENA = 1
0	1	1
1	2	2
2	4	4
3	8	8
4	16	10
5	32	100
6	64	1000
7	128	10000
8	256	20000
9	512	20000
10	1024	20000
11	2048	20000
12	4096	20000
13	8192	20000
14	16384	20000
15	32768	20000

Table 3: PRE_SCLAR Decode Values

4.1 MOD Register

16 Bit Data Bus

Bit #	Access	Description
15:0	RW	MOD_VAL, Main Counter modulo value Sets the rollover value of the Main Counter.

Reset Value:
        MOD_VAL: 0000h

Table 4: MOD Register Bits

8 Bit Data Bus

Bit #	Access	Description
7:0	RW	MOD_VAL_0, MOD_VAL[7:0] Main Counter modulo value Sets the rollover value of the Main Counter.

Reset Value:
        MOD_VAL_0: 00h

Table 4a: MOD_VAL_0 Register Bits

Bit #	Access	Description
7:0	RW	MOD_VAL_1, MOD_VAL[15:8] Main Counter modulo value Sets the rollover value of the Main Counter.

Reset Value:
        MOD_VAL_1: 00h

Table 4b: MOD_VAL_1 Register Bits

4.2 CNT Register

16 Bit Data Bus

Bit #	Access	Description
15:0	RW	COUNT_VAL, Current state of the Main Counter.

Reset Value:
> CNT: 0001h

Table 5: CNT Register Bits

8 Bit Data Bus

Bit #	Access	Description
7:0	RW	COUNT_VAL_0, COUNT_VAL[7:0] Current state of the Main Counter. Note: To minimize the gates there is no register to capture the full value of COUNT_VAL when only a byte is read. This means that the value that the processor sees is only the approximate value of COUNT_VAL because one of the bytes will have changed between the read of the first byte of COUNT_VAL and the second byte of COUNT_VAL.

Reset Value:
> CNT_0: 01h

Table 5a: CNT_0 Register Bits

Bit #	Access	Description
7:0	RW	COUNT_VAL_1, COUNT_VAL[15:8] Current state of the Main Counter.

Reset Value:
> CNT_1: 00h

Table 5b: CNT_1 Register Bits

5

Clocks

Name	Source	Rates (MHz)			Remarks	Description
		Max	Min	Resolution		
wb_clk_i	System	200	-	-	Master clock for all PIT registers. Positive edge active.	System clock.

Table 3: List of clocks

The wb_clk_i has no timing constraints based on the RTL implementation although there may be constrains applied for synthesis results to be compatible with the target physical implementation. If the PIT is targeted for an ASIC implementation then [wb_clk_i] should be used as the scan clock, any clock multiplexing required to make [wb_clk_i] the scan clock should be done at the system level external to the PIT module.

6

IO Ports

Port	Width	Direction	Description
wb_clk_i	1	Input	WISHBONE Bus Clock Input, Master Clock
wb_rst_i	1	Input	WISHBONE Synchronous Reset
wb_adr_i	3	Input	WISHBONE Lower address bits
wb_dat_i	8/16	Input	WISHBONE Bus Data
wb_dat_o	8/16	Output	WISHBONE Bus Data
wb_we_i	1	Input	WISHBONE Write enable
wb_stb_i	1	Input	WISHBONE Strobe signal/Core select
wb_cyc_i	1	Input	WISHBONE Valid bus cycle
wb_sel_i	2	Input	WISHBONE Data Bus Byte Select
wb_ack_o	1	Output	WISHBONE Bus cycle acknowledge
pit_int_o	1	Output	PIT Interrupt signal
pit_o	1	Output	PIT output pulse
cnt_flag_o	1	Output	Counter rollover flag
cnt_sync_o	1	Output	Sync signal to slave mode PIT
ext_sync_i	1	Input	Sync signal from master mode PIT

Table 4: List of IO ports

6.1 WISHBONE Interface

The core features a WISHBONE RevB.3 compliant WISHBONE Classic interface that operates in SLAVE mode. All output signals are registered. Each access takes 2 clock cycles.

WISHBONE DATASHEET	
Description	Specification
General description:	8-bit SLAVE
Supported Cycles:	SLAVE, READ/WRITE
Data port, size: Data port, granularity: Data port, maximum operand size: Data transfer ordering: Data transfer sequencing:	Default: 16, option 8 bit Default: 16, option to match 8 bit port size

	Signal Name	WISHBONE Equiv.
Supported signal list and cross reference to equivalent WISHBONE signals:	wb_clk_i	CLK_I
	wb_rst_i	RST_I
	wb_adr_i	ADR_I()
	wb_dat_i	DAT_I()
	wb_dat_o	DAT_O()
	wb_we_i	WE_I
	wb_stb_i	STB_I
	wb_cyc_i	CYC_I
	wb_sel_i	SEL_I
	wb_ack_o	ACK_O

6.1.1 wb_rst_i

The synchronous reset signal has a minimum pulse width requirement of one [wb_clk_i] clock period. It will take two [wb_clk_i] clock cycles for all registers in the PIT module to initialize. Also see information on pin [arst_i].

6.1.2 wb_adr_i

Connections to the WISHBONE address pin will depend on the size of the WISHBONE data bus that is set by the DWIDTH parameter. If DWIDTH=8 the all address pins should be connected, if DWIDTH=16 then [wb_adr_i(2)] should be tied low.

6.1.3 wb_sel_i

The [wb_sel_i] is the WISHBONE byte lane select signal. It is currently unimplemented in the PIT module and should be tied hi.

6.2 PIT signals

6.2.1 pit_o

The PIT output signal. This is a one master clock, [wb_clk_i], wide pulse that is output when the Master Counter rolls over after reaching the value stored in the Modulo Register.

6.2.2 cnt_flag_o

This signal is an optional output that tracks the value in the FLAG register. The positive edge becomes active at the same time as the [pit_o] signal and the negative edge is dependent on the PIT interrupt service routine, or polling loop, to clear the FLAG register. This signal may be occasionally useful when an output signal with a longer pulse width is needed to be resynchronized to a slower clock domain than the PIT master clock.

6.2.3 pit_int_o

This signal is the interrupt output from the PIT. The timing and function of this signal is the same as [cnt_flag_o] with an additional output inhibit provided by the ENA_INT control bit.

6.2.4 cnt_sync_o

A PIT that is to be used in master mode will use this signal to synchronize the start of counting in PIT instances that are operating in slave mode. All slave mode PIT's that will synchronize to a common master will connect their [ext_sync_i] inputs to this output. The CNT_EN control register drives this signal.

6.2.5 ext_sync_i

A PIT module operating in slave mode will use this signal to synchronize the start of its counters to the setting of the CNT_EN register in the associated master mode PIT. If the PIT module is never to be used in SLAVE mode then this input signal should be tied to the cnt_sync_o signal.

6.2.6 arst_i

The signal [arst_i] is an asynchronous reset signal that goes to all flops in the PIT. It is provided for FPGA implementations and test methodologies that require this function for initialization. Using [arst_i] instead of [wb_rst_i] can result in lower cell-usage and higher performance for a FPGAs implementation because the standard FPGA cell already provides a dedicated asynchronous reset path. Using [wb_rst_i] for an ASIC implementation might synthesize to a smaller module because smaller non_reset flops can be used. Use either [arst_i] or [wb_rst_i], tie the other to a negated state. The active level of [arst_i] is determined by the parameter ARST_LVL which defaults to active low.

6.3 PIT Core Parameters

Parameter	Type	Default	Description
ARST_LVL	Bit	1'b0	Asynchronous reset level
COUNT_SIZE	Int	16	Number of bits in Main Counter and modulo
DECADE_CNTR	Bit	1'b1	Prescaler decode, 0 = Binary counter, 1 = Decade counter
NO_PRESCALE	Bit	1'b0	Prescale counter disabled, 0 = Prescale counter enabled, 1 = Prescale counter disabled
DWIDTH	Int	16	Data Bus size

6.3.1 ARST_LVL

The asynchronous reset level can be set to either active high (1'b1) or active low (1'b0).

Allowed values: 1'b0, 1'b1

6.3.2 COUNT_SIZE

The number of bits in the Main Counter. This feature is not fully implemented in the current PIT RTL.

Allowed values: 16-1

6.3.3 DECADE_CNTR

The Prescale Counter can be set to count as a binary decoded output or a decade decoded output. Setting this value may offer a more intuitive programmer interface. Clearing this bit will allow the maximum prescale value and may also slightly reduce the amount of combinational logic required to decode the Prescale Counter output.

Allowed values: 1'b0, 1'b1

6.3.4 NO_PRESCALE

Setting this bit should remove the Prescale Counter from the synthesis results to save size and power if this function is not needed to meet system requirements. Synthesis scripts should be coded to allow for the removal of flops with unconnected outputs. Synthesis logfiles should be reviewed to ensure the Prescale counter and it's control registers are removed.

Allowed values: 1'b0, 1'b1

6.3.5 DWIDTH

The width of the microcontroller data buses connected to PIT module. The PIT module can support either an 8-bit data bus with 8-bit resolution or a 16-bit data bus with 16-bit resolution.

Allowed values: 8, 16

Appendix A

Name

[This section may be added to outline different specifications.]

Index

[This section contains an alphabetical list of helpful document entries with their corresponding page numbers.]