Fixed Interval Timer v2.0

LogiCORE IP Product Guide

PG110 November 18, 2015

XILINX®

XILINX

Table of Contents

Introduction

The Fixed Interval Timer (FIT) core is a peripheral that generates a strobe (interrupt) signal at fixed intervals and is not attached to any bus. The FIT core generates an interrupt every C_NO_CLOCKS. The interrupt signal is held High for one clock cycle. The core begins operation immediately after device configuration unless the clock is held or a reset is connected to the FIT.

Features

- Configurable number of clock cycles between interrupts
- Configurable inaccuracy in clock intervals between interrupts
- Optional reset

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family[1]	UltraScale+™ Families, UltraScale™ Architecture, Zynq®-7000 All Programmable SoC, 7 series
Supported User Interfaces	N/A
Resources	Performance and Resource Utilization web page
Provided with Core	
Design Files	Vivado: VHDL
Example Design	Not Provided
Test Bench	Not Provided
Constraints File	N/A
Simulation Model	VHDL Behavioral
Supported S/W Driver	N/A
Tested Design Flows[2]	
Design Entry	Vivado® Design Suite
Simulation	For supported simulators, see the Xilinx Design Tools: Release Notes Guide.
Synthesis	Vivado Synthesis
Support	
Provided by Xilinx at the Xilinx Support web page	

Notes:

1. For a complete listing of supported devices, see the Vivado IP Catalog.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Overview

FIT Interrupt Descriptions

The FIT generates an interrupt every C_NO_CLOCKS. The interrupt signal is held High for one clock cycle. The core begins operation immediately after device configuration if the reset is not connected. If the reset is connected and asserted the FIT begins operation after the reset is released. To reset the SRL16 primitives the reset must be asserted for a minimum of 17 cycles. To reset the bit counter implementation the reset need only be asserted for one clock cycle.

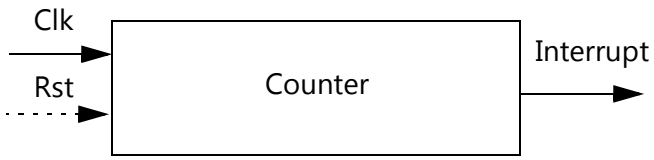

Figure 1-1: **FIT Block Diagram**

Design Tools

The FIT design is generated by the Vivado® Design Suite.

Vivado synthesis is the synthesis tool used for synthesizing the FIT. The netlist output from the synthesis tool is then input to the Vivado Design Suite for actual device implementation.

Licensing and Ordering Information

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado® Design Suite under the terms of the Xilinx End User License. Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

Product Specification

Performance

The frequency and latency of the FIT are optimized for use with MicroBlaze™. This means that the frequency targets are aligned to MicroBlaze targets.

Resource Utilization

For details about resource utilization, visit Performance and Resource Utilization.

Port Descriptions

The I/O signals for the FIT are listed in Table 2-1.

Table 2-1: **FIT I/O Signals**

Signal Name	Interface	I/O	Initial State	Description
Clk	N/A	I	N/A	Clock
Rst	N/A	I	N/A	Optional reset
Interrupt	N/A	O	0	Interrupt signal

Designing with the Core

General Design Guidelines

This chapter includes guidelines and additional information to facilitate designing with the core.

Clocking

The input clock must be connected to the `Clk` port. The frequency of this clock, together with the C_NO_CLOCKS parameter, determines the time between strobes.

The inaccuracy between strobes is affected by the parameter C_INACCURACY. The parameter should normally be set to the default value 0, but resource usage can be reduced by allowing a certain inaccuracy.

Resets

The `Rst` port can be left unconnected or tied to ground to reduce resource usage. When resetting the SRL16 the reset signal must be asserted for at least 17 cycles. When resetting the bit counter the reset need only be asserted for one cycle.

Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows and the IP integrator can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 1]
- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 3]
- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 4]

Customizing and Generating the Core

This section includes information about using Xilinx tools to customize and generate the core in the Vivado Design Suite.

If you are customizing and generating the core in the Vivado IP integrator, see the *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 1] for detailed information. IP integrator might auto-compute certain configuration values when validating or generating the design. To check whether the values do change, see the description of the parameter in this chapter. To view the parameter value, run the `validate_bd_design` command in the Tcl console.

You can customize the IP for use in your design by specifying values for the various parameters associated with the IP core using the following steps:

1. Select the IP from the Vivado IP catalog.
2. Double-click the selected IP or select the **Customize IP** command from the toolbar or right-click menu.

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2] and the *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 3].

Note: Figures in this chapter are illustrations of the Vivado Integrated Design Environment (IDE). The layout depicted here might vary from the current version.

The FIT configuration dialog is shown in Table 4-1.

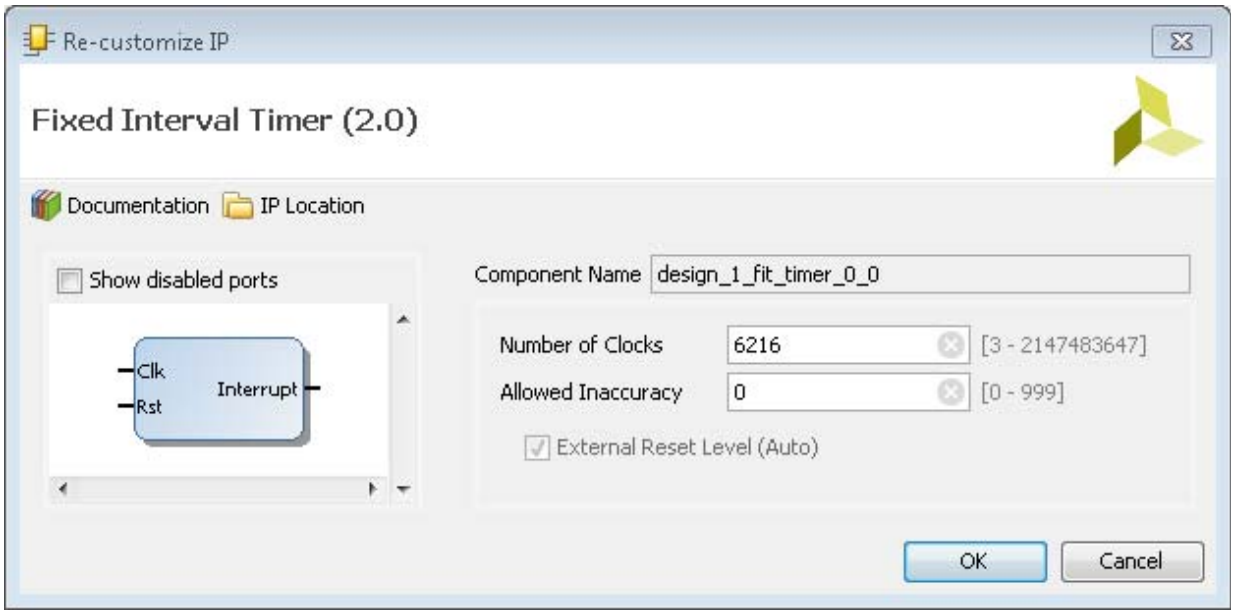

Figure 4-1: **Configuration Dialog**

- **Number of Clocks** - Sets the number of clock cycles between strobes.
- **Allowed Inaccuracy** - Defines the allowed inaccuracy between strobes. Normally this should not be changed from the default value of 0.
- **External Reset Level** - When checked indicates active-High level, and when unchecked indicates active-Low level.

Parameter Values

Table 4-1: **FIT Parameters**

Parameter Name	Feature /Description	Allowable Values	Default Value	Tool Assigned	VHDL Type
C_FAMILY	Device Family	Supported architectures	virtex7	Yes	string
C_NO_CLOCKS	The number of clock cycles between strobes	3 - 2147483647	6216	No	integer
C_INACCURACY	The allowed inaccuracy in the number of clock cycles between strobes. Expressed in per thousands.	0 - 999	0	No	integer
C_EXT_RESET_HIGH	Level of reset	0 = active-Low 1 = active-High	1	No	integer

User Parameters

Table 4-2 shows the relationship between the fields in the Vivado IDE and the User Parameters (which can be viewed in the Tcl console).

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship**

Vivado IDE Parameter	User Parameter	Default Value
Number of Clocks	C_NO_CLOCKS	6216
Allowed Inaccuracy	C_INACCURACY	0
External Reset Level	C_EXT_RESET_HIGH	1

Constraining the Core

This section contains information about constraining the core in the Vivado Design Suite.

Required Constraints

There are no required constraints for this core.

Device, Package, and Speed Grade Selections

There are no Device, Package or Speed Grade requirements for this core.

Clock Frequencies

There are no specific clock frequency requirements for this core.

Clock Management

There are no specific Clock management requirements for this core.

Clock Placement

There are no specific Clock placement requirements for this core.

Banking

There are no specific Banking rules for this core.

Transceiver Placement

There are no Transceiver Placement requirements for this core.

I/O Standard and Placement

There are no specific I/O standards and placement requirements for this core.

Simulation

For comprehensive information about Vivado simulation components, as well as information about using supported third-party tools, see the *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 4].

IMPORTANT: *For cores targeting 7 series or Zynq-7000 devices, UNIFAST libraries are not supported. Xilinx IP is tested and qualified with UNISIM libraries only.*

Synthesis and Implementation

For details about synthesis and implementation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2].

Migrating

This appendix contains information about upgrading to a more recent version of the IP core.

For information on migrating to the Vivado® Design Suite, see the *ISE to Vivado Design Suite Migration Guide* (UG911) [Ref 5].

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

Finding Help on Xilinx.com

To help in the design and debug process when using the FIT core, the Xilinx Support web page contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

Documentation

This product guide is the main document associated with the FIT core. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the Downloads page. For more information about this tool and the features available, open the online help after installation.

Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records for this core can be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as

- Product name
- Tool message(s)
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

Answer Records for the FIT Core

- AR54457

Technical Support

Xilinx provides technical support at the Xilinx Support web page for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

- Implement the solution in devices that are not defined in the documentation.
- Customize the solution beyond that allowed in the product documentation.
- Change any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support, navigate to the Xilinx Support web page.

1. Navigate to the Xilinx Support web page.
2. Open a support case by selecting the Open a Service Request link located under Additional Resources.

When opening a support case, include:

- Additional files based on the specific issue might also be required. See the relevant sections in this debug guide for guidelines about which file(s) to include with the support case.

Note: Access to the Xilinx Service Portal is not available in all cases. Log in to the portal to see your specific support options.

Debug Tools

The main tool available to address FIT design issues is the Vivado® Design Suite debug feature.

Vivado Design Suite Debug Feature

The Vivado® Design Suite debug feature inserts logic analyzer and virtual I/O cores directly into your design. The debug feature also allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature in the Vivado IDE is used for logic debugging and validation of a design running in Xilinx devices.

The Vivado logic analyzer is used to interact with the logic debug LogiCORE IP cores, including:

- ILA 2.0 (and later versions)
- VIO 2.0 (and later versions)

Reference Boards

All 7 series Xilinx development boards support the FIT core. These boards can be used to prototype designs and establish that the core can communicate with the system.

Simulation Debug

The simulation debug flow for Mentor Graphics Questa Simulator (QuestaSim) is described below. A similar approach can be used with other simulators.

- Check for the latest supported versions of QuestaSim in the Xilinx Design Tools: Release Notes Guide. Is this version being used? If not, update to this version.
- If using Verilog, do you have a mixed mode simulation license? If not, obtain a mixed-mode license.
- Ensure that the proper libraries are compiled and mapped. In the Vivado Design Suite **Flow > Simulation Settings** can be used to define the libraries.

Hardware Debug

This section provides debug steps for common issues. The Vivado Design Suite debug feature is a valuable resources to use in hardware debug.

General Checks

Ensure that all the timing constraints for the core were properly incorporated from the example design and that all constraints were met during implementation.

- Does it work in post-place and route timing simulation? If problems are seen in hardware but not in timing simulation, this could indicate a PCB issue. Ensure that all clock sources are active and clean.
- If using MMCMs in the design, ensure that all MMCMs have obtained lock by monitoring the `LOCKED` port.

Appendix C

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

References

This document provides supplemental material useful with this product guide:

1. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
2. *Vivado Design Suite User Guide: Designing with IP* (UG896)
3. *Vivado Design Suite User Guide: Getting Started* (UG910)
4. *Vivado Design Suite User Guide: Logic Simulation* (UG900)
5. *ISE to Vivado Design Suite Migration Guide* (UG911)
6. *Vivado Design Suite User Guide: Programming and Debugging* (UG908)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
11/18/2015	2.0	Added support for UltraScale+ families.
06/24/2015	2.0	Moved performance and resource utilization data to the web.
03/20/2013	1.0	This Product Guide replaces PG086. There are no documentation changes for this release.

Please Read: Important Legal Notices