Trion DDR Calibration and Debug IP User Guide

UG-TRIONDDR-CALDBG-v1.1
March 2024
www.efinixinc.com

Contents

Introduction

The Trion DDR Calibration and Debug IP core helps you calibrate and reset the Trion DDR controller using CA training, write leveling, read leveling, gate training, and reset sequencing (memory initialization).

Use the IP Manager to select IP, customize it, and generate files. The Trion DDR Calibration and Debug IP core has an interactive wizard to help you set parameters. The wizard also has options to create example design targeting an Efinix® development board.

Features

- Automatically performs leveling calibration for the DDR DRAM interface and external memory module
- Supports CA training, write leveling, read leveling, reset (memory initialization), and gate training calibration
- x16 and x32 DQ widths
- Verilog RTL
- Includes an example design targeting the Trion® T120F324 FPGA and T120F576 Development Board.

Device Support

Table 1: Trion DDR Calibration and Debug IP Core Device Support

FPGA Family	Supported Device
Trion	T20 (BGA324 and BGA400 packages only), T35, T55, T85, T120
Titanium	–

Resource Utilization and Performance

Note: The resources and performance values provided are based on some of the supported FPGAs. These values are just guidance and can change depending on the device resource utilization, design congestion, and user design.

Table 2: Resource Utilization and Performance

FPGA	Logic Elements (Logic, Adders, Flipflops, etc.)	Memory Block	DSP Block	f_{MAX} (MHz)	Efinity® Version[1]
T120 BGA324 C4	4,663/112,128 (4%)	24/1,056 (2%)	0	70	2023.2

Release Notes

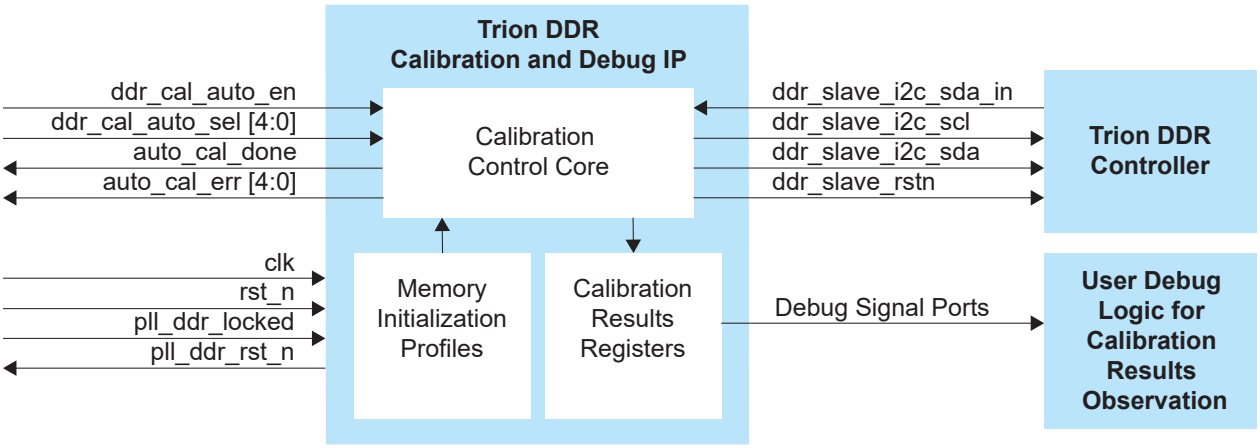
You can refer to the IP Core Release Notes for more information about the IP core changes. The IP Core Release Notes is available in the Efinity Downloads page under each Efinity software release version.

Note: You must be logged in to the Support Portal to view the IP Core Release Notes.

Functional Description

The core consists of a calibration control core, a memory initiation profiles and a calibration result register.

Figure 1: Trion DDR Calibration and Debug IP Core Block Diagram



[1] Using Verilog HDL.

Ports

Table 3: Trion DDR Calibration and Debug IP Core Ports

Port	Direction	Description
rst_n	Input	Reset. Asynchronous, active-low reset signal that initializes all internal pointers and output registers.
clk	Input	System clock. This clock drives all signals in the core. Efinix recommends using a clock in the range of 50 MHz.
ddr_cal_auto_en	Input	Enable signal to start the automation state machine.
ddr_cal_auto_sel[4:0]	Input	Select signal to configure the calibration or reset mode you want to run. Bit 0: Enable gate training Bit 1: Enable read leveling Bit 2: Enable write leveling[2] Bit 3: Enable memory initialization. Bit 4:Enable CA training You need to select the applicable option in the **Control Tab** of the DDR instance in the Interface Designer before instantiating the core. See **Using the Core**.
pll_ddr_locked	Input	PLL lock indicator from the generated system clock. Set signal to 1 (high) if the system clock is not from a PLL.
pll_ddr_rst_n	Output	PLL active-low reset to the generated DDR clock. Normally set to 1 (high).
auto_cal_done	Output	Optional. Indicates that the automation state machine successfully completed calibration.
auto_cal_err	Output	Optional. Indicates that the automation state machine started with an invalid ddr_cal_auto_sel setting.
ddr_slave_i2c_scl	Output	I^2C clock to the DDR controller.
ddr_slave_i2c_sda	Output	I^2C data to the DDR controller.
ddr_slave_i2c_sda_in	Input	I^2C enable to the DDR controller.
ddr_slave_rstn	Output	Reset signal to DDR controller.

[2] Disable the write leveling (Bit 2 = 0) if you do not use fly-by topology in your board design.

Debug Signal Ports

Table 4: Trion DDR Calibration and Debug IP Core Debug Signal Ports

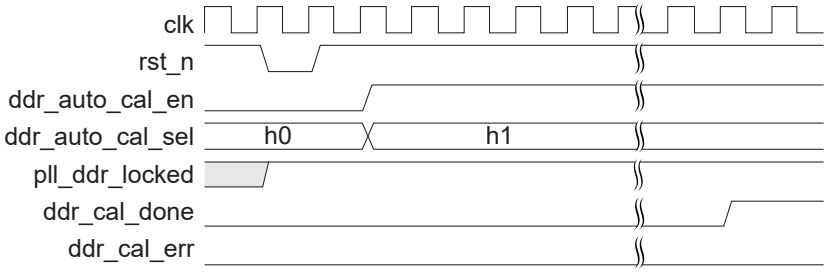
Port	Direction	Description
calvl_range[31:0]	Output	CA leveling calibration result in delay (ps). MSB starts the scanning of the Windows. LSB ends the scanning of the Windows. Each bit shows the result of the time slot. 1 = Pass; 0 = Fail.
calvl_window[31:0]	Output	CA leveling calibration valid windows.
gatlvl_pass[3:0]	Output	Gate training calibration pass. Bit [0] = 1, Byte 0 passed. Bit [1] = 1, Byte 1 passed. Bit [2] = 1, Byte 2 passed. Bit [3] = 1, Byte 3 passed.
gatlvl_delay*N*[31:0]	Output	Gate training calibration result in delay (ps). N for byte index.
wrlvl_pass[3:0]	Output	Write leveling calibration pass. Bit [0] = 1, Byte 0 passed. Bit [1] = 1, Byte 1 passed. Bit [2] = 1, Byte 2 passed. Bit [3] = 1, Byte 3 passed.
wrlvl_delay*N*[31:0]	Output	Write leveling calibration result in delay (ps). N for byte index. MSB starts the scanning of the Windows. LSB ends the scanning of the Windows. Each bit shows the result of the time slot. 1 = Pass; 0 = Fail.
rdlvl_range*N*[31:0]	Output	Read leveling calibration result in delay (ps). N for byte index.
rdlvl_window*N*[31:0]	Output	Read leveling calibration valid windows. N for byte index.
gatlvl_value*N*[31:0]	Output	Gate training calibration result. gatlvl_valueN [15:0] fine delay, gatlvl_valueN [31:16] coarse delay, N for byte index.

Calibration State Machine Flow

The core logic runs the initiation and calibration flow based on the memory initiation profiles. These profiles correspond to the parameter settings from the Interface Designer.

When you assert `ddr_cal_auto_en`, the calibration state machine starts. Each calibration mode could be enabled by the `ddr_cal_auto_sel` signal. The core prioritizes memory initialization reset, followed by CA training, write leveling, gate leveling, and finally read leveling. For example, if `ddr_cal_auto_sel` is set to `5'b00101`, the calibration state machine performs write leveling calibration and then gate leveling calibration. Refer to **Table 3: Trion DDR Calibration and Debug IP Core Ports** on page 5.

Figure 2: Trion DDR Calibration and Debug IP Operation

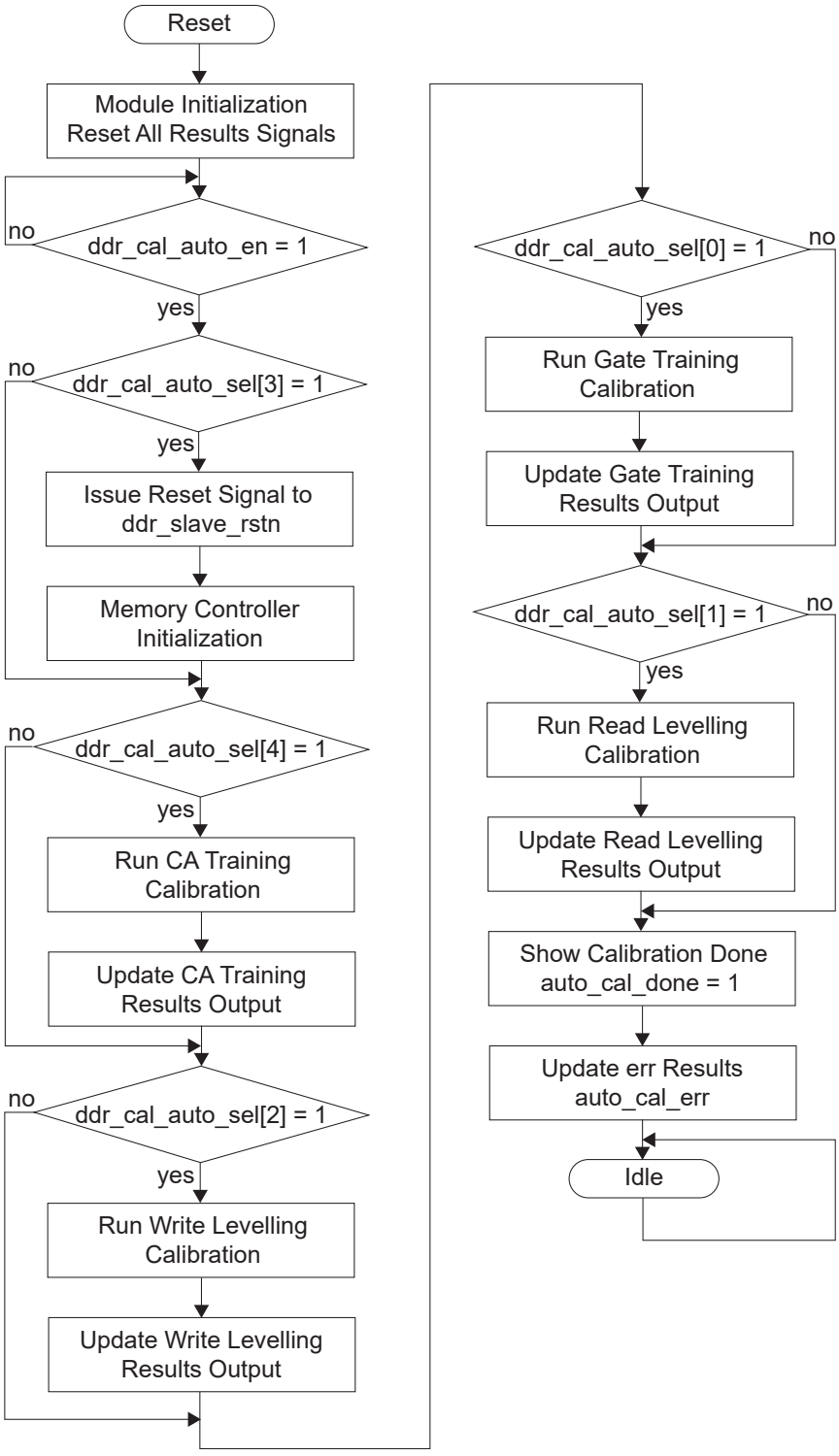


During the calibration process, the core configures the Trion DDR controller through the I^2C signals, e.g., `ddr_slave*`, etc.

Upon completion of the calibration process, the `ddr_auto_cal_done` signal is asserted, while the `ddr_cal_auto_err` and the debug signals ports are updated.

You may want to implement a user debug logic in your design to probe for the debug signal using the Efinity software debugger. For further details regarding its operation and waveform, refer to **Figure 2: Trion DDR Calibration and Debug IP Operation** on page 7 and the **Figure 3: Simplified Calibration State Machine Flow** on page 8 respectively.

Figure 3: Simplified Calibration State Machine Flow



Using the Core

Before generating the Trion DDR Calibration and Debug IP core in the Efinity IP Manager, you must create a DDR block in the Interface Designer and set the settings in the **Block Editor**. After creating the DDR block, click the **Generate Efinity Constraints Files** button. Then, instantiate the core with the `ddr_cal_auto_sel[4:0]` input driven accordingly.

Note: Only one DDR block can be added in the Interface Designer for each Trion FPGA.

Examples:

Operation	Interface Designer Block Editor (Control Tab)	ddr_cal_auto_sel[4:0]	Note
Calibration and Reset	Enable Calibration and Reset	ddr_cal_auto_sel[3]=1 ddr_cal_auto_sel[2:0] and ddr_cal_auto_sel[4] set as per required calibration	The Interface Designer **Block Editor** (Control Tab) enables the Master Reset Pin. You need to toggle from low to high, before using the core. You cannot disable the reset (memory initialization) in this operation option.

Important: For the Trion DDR Calibration and Debug IP core to properly propagate settings from the Interface Designer, the project directory name must match the project file names (**<project>.xml** and **<project>.peri.xml**).

You need to regenerate the Trion DDR Calibration and Debug IP core each time you change the settings in the Interface Designer and click the **Generate Efinity Constraints Files** button.

Note: You should only trigger reset and recalibration after any ongoing calibration is completed (`auto_cal_done` signal is asserted).

Important:

- You need to choose **Enable Reset and Calibration** in the **Control Tab** to use the Trion DDR Calibration and Debug IP core.
- After generating the IP, go to **File** > **Edit Project** > **Synthesis** > **Verilog define Macro** to add the `DEBUG_MODE` and `EXTRA_DEBUG_SIGNAL`. Set both the values to 1.

IP Manager

The Efinity® IP Manager is an interactive wizard that helps you customize and generate Efinix® IP cores. The IP Manager performs validation checks on the parameters you set to ensure that your selections are valid. When you generate the IP core, you can optionally generate an example design targeting an Efinix development board. This wizard is helpful in situations in which you use several IP cores, multiple instances of an IP core with different parameters, or the same IP core for different projects.

Note: Not all Efinix IP cores include an example design or a testbench.

Generating the Trion DDR Calibration and Debug IP Core with the IP Manager

The following steps explain how to customize an IP core with the IP Configuration wizard.

1. Open the IP Catalog.
2. Choose **Memory Controllers > Trion DDR Calibration and Debug IP** core and click **Next**. The **IP Configuration** wizard opens.
3. Enter the module name in the **Module Name** box.

   Note: You cannot generate the core without a module name.
4. Customize the IP core using the options shown in the wizard. For detailed information on the options, refer to the Customizing the Trion DDR Calibration and Debug IP on page 11.
5. (Optional) In the **Deliverables** tab, specify whether to generate an IP core example design targeting an Efinix® development board. These options are turned on by default.
6. (Optional) In the **Summary** tab, review your selections.
7. Click **Generate** to generate the IP core and other selected deliverables.
8. In the **Review configuration generation** dialog box, click **Generate**. The Console in the **Summary** tab shows the generation status.

   Note: You can disable the **Review configuration generation** dialog box by turning off the **Show Confirmation Box** option in the wizard.
9. When generation finishes, the wizard displays the **Generation Success** dialog box. Click **OK** to close the wizard.

The wizard adds the IP to your project and displays it under **IP** in the Project pane.

Generated Files

The IP Manager generates these files and directories:

- **<module name>_define.vh**—Contains the customized parameters.
- **<module name>_tmpl.v**—Verilog HDL instantiation template.
- **<module name>_tmpl.vhd**—VHDL instantiation template.
- **<module name>.v**—IP source code.
- **settings.json**—Configuration file.
- **<kit name>**—Has generated RTL, example design, and Efinity® project targeting a specific development board.

Customizing the Trion DDR Calibration and Debug IP

The core has parameters so you can customize its function. You set the parameters in the General tab of the core's IP Configuration window.

Table 5: Trion DDR Calibration and Debug IP Core Parameters

Parameters must be the same as the settings you set for the DDR block in Interface Designer. See **Using the Core** on page 9.

Parameter	Options	Description
DDR Memory Type	DDR3/DDR3L, LPDDR2, LPDDR3	Defines the DDR memory module type. Default: LPDDR3
DDR Memory Width	16-bits, 32-bits	Defines the DDR memory width. Only LPDDR3 has the option for x32. Default: 16-bits
DDR Frequency	DDR3/DDR3L: 300 – 533 MHz LPDDR2: 75 – 533 MHz LPDDR3: 75 – 533 MHz	Defines the DDR PHY frequency. Default: 533 MHz Use the DDR frequency from PLL to calculate the time (ps) of the step. **Note:** You must ensure that the DDR frequency matches the DDR clock of the DDR controller.

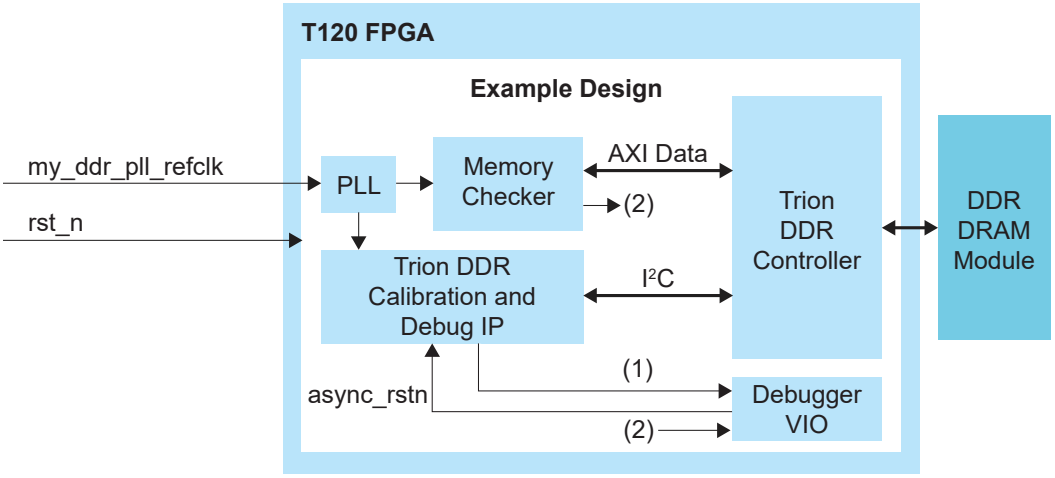
Trion DDR Calibration and Debug IP Example Design

You can choose to generate the example design when generating the core in the IP Manager Configuration window. Compile the example design project and download the **.hex** or **.bit** file to your board through the Efinity Debugger.

Important: Efinix tested the example design generated with the default parameter options only.

The example design targets the Trion® T120F324 FPGA (you may also use Trion® T120 BGA576 Development Board). It implements a memory checker that writes data into the DDR DRAM module on the board and then performs a read operation. The design compares the returned data to the sent data and displays the results, `memtest_pass` signal through the Efinity Debugger. You can turn calibration on or off to see how the DDR DRAM module works with and without calibration. The design has one clock feeding the PLL, 50 MHz (`my_ddr_pll_refclk`).

Figure 4: Trion DDR Calibration and Debug IP Core Example Design



Note: (1) - User Debug Logic for calibration result and observations.
(2) - Interconnect between Memory Checker and Debugger VIO.

Table 6: Example Design Implementation

FPGA	Logic Elements (Logic, Adders, Flipflops, etc.)	Memory Block	DSP Block	f_{MAX} (MHz)	Efinity® Version[3]
T120 BGA324 C4	7707/112128 (7%)	24/1056 (2%)	0	70	2023.2

[3] Using Verilog HDL.

Control the Calibration Mode with Efinity Debugger

The design lets you switch the calibration mode between CA training, write levelling, gate training, and read leveling calibration operations through the `ddr_cal_auto_sel` debug signal probed by the **vio0** in the Efinity Debugger. You need to toggle the `async_rstn` signal each time you change the calibration mode through the `ddr_cal_auto sel` signal. The calibration results and observations can be observed through **vio1** of the Efinity Debugger. For further details regarding its operation and waveform, refer to **Table 3: Trion DDR Calibration and Debug IP Core Ports** on page 5 and the **Figure 3: Simplified Calibration State Machine Flow** on page 8.

Using the Example Design

As the example design runs, the **vio0** and **vio1** of Efinity Debugger shows the probed debug signal and calibration results. The `ddr_cal_auto_sel` signal is set to `5'b11111` where it performs memory initialization reset, followed by CA Training, write leveling, gate leveling, and finally read leveling. You can monitor the calibration results and observations through **vio0** and **vio1**.

- The `cali_done` and `memtest_start` signals are asserted to indicate the completion of calibration and the operation of the memory checker.
- If the memory checker successfully writes and then reads the memory, the `memtest_pass` signal is asserted.

Revision History

Table 7: Revision History

Date	Version	Description
March 2024	1.1	Updated Features, Resource Utilization and Performance, Using the Core, IP Manager, Customizing The Trion DDR Calibration and Debug IP, and Trion DDR Calibration and Debug IP Example Design. (DOC-1727)
December 2023	1.0	Initial release.