1300 Henley Court
Pullman, WA 99163
509.334.6306
www.digilentinc.com

Zmod Scope Configuration 1.0 IP Core User Guide

Revised March 18, 2022; Author Eduard Niță

1 Introduction

This user guide describes the Digilent **Zmod Scope Configuration** Intellectual Property. It is used in conjunction with the **Zmod Scope Controller IP** to configure the **calibration coefficients, gain, coupling** and inspect the Zmod Scope's **status flags** using an **AXI-Lite interface**.

IP quick facts	
Supported device families	Zynq®-7000, 7 series
Supported user interfaces	Xilinx®: AXI4-Lite
Provided with core	
Design files	C++ VHDL/Verilog (generated)
Simulation model	HLS Cosimulation
Constraints file	XDC
Software driver	HLS Generated
Tested design flows	
Design entry	Vitis™ HLS 2021.1
Synthesis	Vivado Synthesis 2021.1

2 Features

- Allows setting the Zmod Scope's **calibration coefficients, High/Low gain, and AC/DC coupling** of each channel
- Allows monitoring of the Zmod Scope's **status flags**
- Xilinx interfaces used: AXI4-Lite

3 Designing with the core

The IP has been initially designed for a xc7z020clg400-1 target device with a target clock frequency of 125MHz (8.00 ns).

The IP is **compatible with all Zmod Scope variations**.

3.1 Customization

Changes to the target device and target clock frequency can be done from the project GUI after the project was generated or by modifying the **SOLUTION_PART/SOLUTION_CLKP** variables found inside the *run_hls_standalone.tcl* file and then generating the project, according to the steps found in Generating the HLS Project.

4 Register map

Offset	Register Name	Description
0x00	Control signals	bit 0 - ap_start (Read/Write/COH) bit 1 - ap_done (Read/COR) bit 2 - ap_idle (Read) bit 3 - ap_ready (Read) bit 7 - auto_restart (Read/Write) others - reserved
0x04	Global Interrupt Enable Register	bit 0 - Global Interrupt Enable (Read/Write)
0x08	IP Interrupt Enable Register (Read/Write)	bit 0 - Channel 0 (ap_done) bit 1 - Channel 1 (ap_ready)
0x0C	IP Interrupt Status Register (Read/TOW)	bit 0 - Channel 0 (ap_done) bit 1 - Channel 1 (ap_ready)
0x10	Channel 1 High Gain Multiplicative Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch1HgMultCoef data (Read/Write)
0x18	Channel 1 Low Gain Multiplicative Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch1LgMultCoef data (Read/Write)
0x20	Channel 1 High Gain Additive Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch1HgAddCoef data (Read/Write)
0x28	Channel 1 Low Gain Additive Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch1LgAddCoef data (Read/Write)
0x30	Channel 2 High Gain Multiplicative Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch2HgMultCoef data (Read/Write)
0x38	Channel 2 Low Gain Multiplicative Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch2LgMultCoef data (Read/Write)
0x40	Channel 2 High Gain Additive Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch2HgAddCoef data (Read/Write)
0x48	Channel 2 Low Gain Additive Coefficient Register	bit 31~18 – Unused bit 17~0 – Ch2LgAddCoef data (Read/Write)

Offset	Register Name	Description
0x50	Relay Configuration Register	bit 31~4 – Unused bit 0 – Channel 1 Gain (Read/Write) bit 1 – Channel 2 Gain (Read/Write) bit 2 – Channel 1 Coupling (Read/Write) bit 3 – Channel 2 Coupling (Read/Write)
0x58	Zmod Scope Status Register	bit 31~5 – Unused bit 0 – RstBusy (Read) bit 1 – InitDoneADC (Read) bit 2 – ConfigError (Read) bit 3 – InitDoneRelay (Read) bit 4 – DataOverflow (Read)

// (SC = Self Clear, COR = Clear on Read, TOW = Toggle on Write, COH = Clear on Handshake)

Bit 0 of the **control register**, **ap_start**, kicks off the core from software. Writing 1 to this bit **read any inputs grouped into the AXI4-Lite** slave interface.

To set the core in free running mode, bit 7 of this register, auto_restart, must be set to 1.

Details on the **0x00-0x0C registers** can be found in Vitis High-Level Synthesis User Guide (UG1399)[1].

Details on the Zmod Scope **calibration coefficients, gain, coupling and status bits** can be found in Zmod Scope Controller IP Core User Guide[2]..

5 Generating the HLS Project

Opening the IP in HLS is possible by executing the following command in the Vitis HLS Command Prompt:

```
cd <path_to_IP>/hls_proj

vitis_hls -f run_hls_standalone.tcl
```

Besides creating the project, the script will also **synthesize** the design and **export** the IP as an archive.

The **source files** of the project can be found in the **src** directory.

The **generated project** will be found inside the **ws** directory.

6 References

6.1 https://www.xilinx.com/support/documentation/sw_manuals/xilinx2021_1/ug1399-vitis-hls.pdf
6.2 https://github.com/Digilent/vivado-library/blob/master/ip/Zmods/ZmodScopeController/docs/ZmodScopeController.pdf