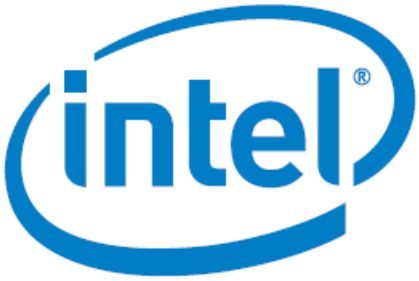


Intel® Agilex® 7 FPGA - Custom Instruction Design on Nios® V/g processor

Date: 6/30/2023

Revision: 1

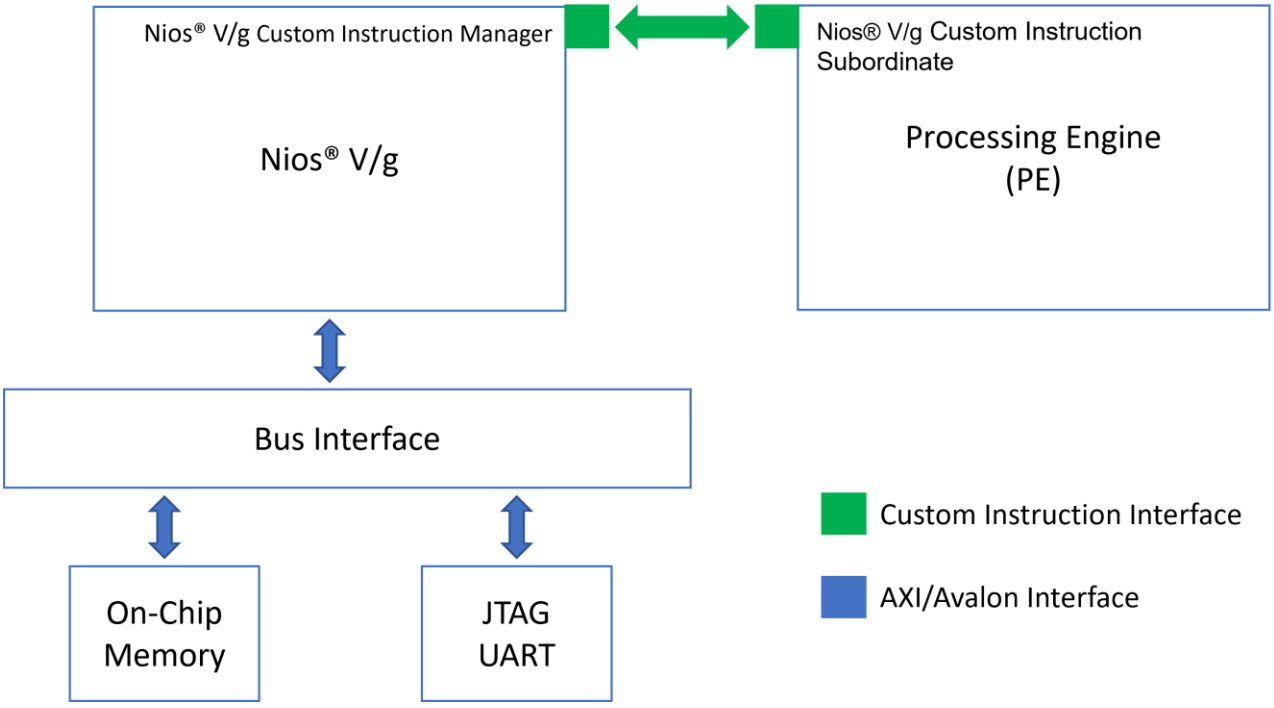
Contents

1. Theory of Operation

This design demonstrates the custom instruction feature of the Nios® V/g processor using Intel Agilex® 7 F-Series FPGA Development Kit.

a. Block Diagram

A Processing Engine (PE) that performs basic arithmetic and logical computations is connected to the Nios® V/g processor using the custom instruction interface. The current version of the Nios® V/g processor custom instruction interface supports operations up-to 32-Bit.



b. Operations performed by Processing Engine (PE)

A Processing Engine (PE) is connected to the Niosv/g processor using the custom instruction interface which performs the following 32-Bit arithmetic and logical operations:

- 1's complement
- 2's complement
- Multiplication
- Bit reversal (Reversing the order of Bits)
- Byte reversal (Reversing the order of Bytes)
- Word reversal (Reversing the order of words)
- Merge lower words (Combine/Merge the lower words of two inputs)
- Merge higher words (Combine/Merge the higher words of two inputs)

c. IP Cores used

The following IPs are used in this design.

- NIOSV/g soft processor core
- On Chip RAM
- JTAG UART
- Custom PE 1
- Custom PE 2

2. Executing the Design on Devkit

a. Creating the Design

Note: Please refer to the readme.txt file in the package for the steps to create the design, application and generate the programing files.

- Unpackage/extract the design in your working directory
- Locate the “ready_to_test” folder within the package
- The folder contains the necessary files for executing the application on the board. Refer to the readme file for the steps to program the application files on the board.
- Validate the design by observing the prints on the terminal

b. Expected Results

The following is the output as observed on the JTAG UART terminal. The output is analogous to the logic from the application code. Users should be able to observe same output on their terminal/setup.

```
Now running custom instruction basic test...
Hello world
********************************************************************************
PE1 operations
********************************************************************************
1's complement of DATA1:0xff
Expected Output:ffffff00
Actual Output:ffffff00
********************************************************************************
2's complement of DATA1:0xff
Expected Output:ffffff01
Actual Output:ffffff01
********************************************************************************
Multiplication of DATA1:0x1f and DATA2:0x80
Expected Output:f80
Actual Output:f80
********************************************************************************
Bit Reversal of DATA1:0x1f
Expected Output:f8000000
Actual Output:f8000000
********************************************************************************
Byte Reversal of DATA1:0x1f
Expected Output:1f000000
Actual Output:1f000000
********************************************************************************
Word Reversal of DATA1:0x1f
Expected Output:1f0000
Actual Output:1f0000
********************************************************************************
Merge Lower-dword DATA1:0x74009078 and DATA2:0x82007083
Expected Output:90787083
Actual Output:90787083
********************************************************************************
Merge Higher-dword of DATA1:0x74009078 and DATA2:0x82007083
Expected Output:74008200
Actual Output:74008200
End of PE1 operations
********************************************************************************
```

```
**********************************************************************
PE2 operations
**********************************************************************
1's complement of DATA1:0xff
Expected Output:ffffff00
Actual Output:ffffff00
**********************************************************************
2's complement of DATA1:0xff
Expected Output:ffffff01
Actual Output:ffffff01
**********************************************************************
Multiplication of DATA1:0x1f and DATA2:0x80
Expected Output:f80
Actual Output:f80
**********************************************************************
Bit Reversal of DATA1:0x111fff
Expected Output:fff88800
Actual Output:fff88800
**********************************************************************
Byte Reversal of DATA1:0x111fff
Expected Output:ff1f1100
Actual Output:ff1f1100
**********************************************************************
Word Reversal of DATA1:0x111fff
Expected Output:1fff0011
Actual Output:1fff0011
**********************************************************************
Merge Lower-dword of DATA1:0x74009078 and DATA2:0x82007083
Expected Output:90787083
Actual Output:90787083
**********************************************************************
Merge Higher-dword of DATA1:0x74009078 and DATA2:0x82007083
Expected Output:74008200
Actual Output:74008200
End of PE2 operations
**********************************************************************
Bye world!
```