

Juha Arvio

Funbase Arria II GX demo

MANUAL

Table of Contents

[1. Identification 4](#_Toc307412519)

[2. Version history 5](#_Toc307412520)

[2.1 Documentation 5](#_Toc307412521)

[3. IP-block files 6](#_Toc307412522)

[3.1 Documentation 6](#_Toc307412523)

[3.2 HDL files 6](#_Toc307412524)

[3.3 Software drivers 7](#_Toc307412525)

[3.4 HDL testbench 7](#_Toc307412526)

[3.5 Test systems 7](#_Toc307412527)

[4. Dependencies 8](#_Toc307412528)

[5. General description 9](#_Toc307412529)

[5.1.1 Ports and signals 10](#_Toc307412530)

List of Abbreviations

fifo first in, first out

FPGA Field Programmable Gate Array

Gbps Giga bits per second

H2P HIBI to PC

HDL Hardware Design Language

HIBI Heterogeneous IP Block Interconnect

HW Hardware

ID Identification

I/O Input / Output

IP Intellectual Property

IRQ Interrupt Request

P2H PC to HIBI

PC Personal Computer

SoC System-on-Chip

SW Software

tbd to be decided

TLP Transaction Layer Protocol

TUT Tampere University of Technology

VHDL VHSIC Hardware Description Language

VHSIC Very High Speed Integrated Circuit

# Identification

Vendor: TUT

Library: soc

Name: arria\_ii\_gx\_demo\_soc

Version: 1.0

Author(s): Lauri Matilainen

Date created: 22.09.2011

Date last modified: 08.11.2011

# Version history

## Documentation

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Change description | Author |
| 31.10.2011 | 0.1 | SoC documentation started | Juha Arvio |

# Arria II GX demo

The Arria II GX demo demonstrates how a picture manipulator can be created from pre-made components with little effort.



Figure 3-1 Arria II GX demo block diagram

# Creating the Arria II GX demo SoC

To create the Arria II GX demo SoC you can take example from the DE2 demo SoC tutorial in the Kactus 2 manual. The result should resemble block diagram shown in figure .



Figure 4- Arria II GX demo block diagram in Kactus 2

### Ports and signals

The interface signals can be divided into the PCIe controller side and to the HIBI side. The PCIe controller side has four different signal groups which include the Avalon ST RX, Avalon ST TX, PCIe IRQ and the LM interface. The HIBI side has also four signal groups which include the HIBI RX, HIBI TX, HIBI message RX and HIBI message TX.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Signal group | Signal | Width [bits] | Dir. | Meaning |
| Reset | rst\_n | 1 | i | Active low reset |
| Clocks | clk | 1 | i | Clock, active on rising edge. |
| HIBI RX | hibi\_addr\_in | 32 | i | HIBI read address |
| hibi\_data\_in | 32 | i | HIBI read data |
| hibi\_comm\_in | 3 | i | HIBI read comm type (write, read req…) |
| hibi\_empty\_in | 1 | i | HIBI read empty |
| hibi\_re\_out | 1 | o | HIBI read enable |
| HIBI TX | hibi\_addr\_out | 32 | o | HIBI write address |
| hibi\_data\_out | 32 | o | HIBI write data |
| hibi\_comm\_out | 3 | o | HIBI write comm type |
| hibi\_full\_in | 1 | i | HIBI write full |
| hibi\_we\_out | 1 | o | HIBI write out |
| HIBI message RX | hibi\_msg\_addr\_in | 32 | i | HIBI message read address |
| hibi\_msg\_data\_in | 32 | i | HIBI message read data |
| hibi\_msg\_comm\_in | 3 | i | HIBI message read comm type |
| hibi\_msg\_empty\_in | 1 | i | HIBI message read empty |
| hibi\_msg\_re\_out | 1 | o | HIBI message read enabe |
| HIBI message TX | hibi\_msg\_addr\_out | 32 | o | HIBI message write address |
| hibi\_msg\_data\_out | 32 | o | HIBI message write data |
| hibi\_msg\_comm\_out | 3 | o | HIBI message write comm type |
| hibi\_msg\_full\_in | 1 | i | HIBI message write full |
| hibi\_msg\_we\_out | 1 | o | HIBI message write out |
| Avalon ST RX | Rx\_St\_Data\_i | 128 | i | Avalon ST receive data |
| Rx\_St\_Valid\_i | 1 | i | Avalon ST receive data valid |
| Rx\_St\_Sop\_i | 1 | i | Avalon ST receive start of packet |
| Rx\_St\_Eop\_i | 1 | i | Avalon ST receive end of packet |
| Rx\_St\_Bardec\_i | 8 | i | Avalon ST receive bar decoded |
| Rx\_St\_Be\_i | 16 | i | Avalon ST receive byte enable |
| Rx\_St\_Ready\_o | 1 | o | Avalon ST receive ready |
| Rx\_St\_Mask\_o | 1 | o | Avalon ST receive mask |
| Avalon ST TX | Tx\_St\_Sop\_o | 1 | o | Avalon ST transmit start of packet |
| Tx\_St\_Eop\_o | 1 | o | Avalon ST transmit end of packet |
| Tx\_St\_Valid\_o | 1 | o | Avalon ST transmit data valid |
| Tx\_St\_Data\_o | 128 | o | Avalon ST transmit data |
| Tx\_St\_Ready\_i | 1 | i | Avalon ST transmit ready |
| TxCred\_i | 36 | i | Avalon ST transmit credit |
| PCIe IRQ | app\_msi\_req | 1 | o | Interrupt request |
| app\_msi\_ack | 1 | i | Interrupt acknowledge |
| app\_msi\_tc | 3 | o |  |
| app\_msi\_num | 5 | o | Interrupt number |
| pex\_msi\_num | 5 | o |  |
| app\_int\_sts | 1 | o |  |
| app\_int\_ack | 1 | i |  |
| LMI | lmi\_data\_in | 32 | i | LMI read data |
| lmi\_re\_out | 1 | o | LMI read enable |
| lmi\_we\_out | 1 | o | LMI write enable |
| lmi\_ack\_in | 1 | i | LMI acknowledge |
| lmi\_addr\_out | 12 | o | LMI address |
| lmi\_data\_out | 32 | o | LMI write data |

Table 4- PCIe to HIBI ports