HUNT ENGINEERING
Chestnut Court, Burton Row,
Brent Knoll, Somerset, TA9 4BP, UK
Tel: (+44) (0)1278 760188,
Fax: (+44) (0)1278 760199,
Email: sales@hunteng.co.uk
www.hunteng.co.uk
www.hunt-dsp.com

Using the HEGD9 with HERON FPGA modules

Rev 1.0 R.Williams 15-09-04

The HERON-FPGA and HERON-IO families are ranges of HERON modules that have FPGAs. The FPGAs provided on these modules enable high performance digital signal processing to be performed in a HERON system.

The GDIO family is a range of modules that provide simple I/O interfaces including Analogue to Digital conversion and Digital to Analogue.

By using FPGA and GDIO modules together, A-to-D and D-to-A conversion can be combined with the FPGA processing of data.

This document discusses how to use FPGA based HERON modules to receive data from A-to-D GDIO modules and to send data to D-to-A GDIO modules.

A VHDL example template is provided. The VHDL in the template is provided as a starting point for GDIO communication in your own FPGA design. The VHDL contained in the example templates will need to be cut and pasted to your project source as required.

History

Rev 1.0	First written

Receiving Data from A-to-D GDIO Modules

The GDIO A-to-D modules output a continuous stream of sample data from one or more channels over a single HERON FIFO connection.

This document assumes that you have already correctly configured your system such that the A-to-D data is arriving on HERON Input FIFO 0 of your FPGA module. If you are using a different FIFO number in your system you will need to change the VHDL accordingly.

With A-to-D data being presented on HERON Input FIFO 0 the FPGA must read 32-bit words of FIFO data and separate that data into distinct A-to-D streams. For each A-to-D channel of the GDIO module one unique data stream must be created within the FPGA.

In the example template there is VHDL that unpacks and strips each channel of data from HERON Input FIFO 0. Data busses are created in either signed or unsigned formats, and unique clock enable signals are created for each channel of A-to-D data.

Using the HEGD9

The HEGD9 samples data on up to four channels. These channels are numbered from 0 to 3. Each channel is 12-bits in size, and the data is unsigned.

The example VHDL receives data from the HEGD9 on HERON Input FIFO 0 and automatically creates four unique data-valid/clock enable signals, one for each channel.

Two separate output buses are created. One for unsigned data from the HEGD9, and one that changes the top-bit to create signed sample data. In your own design either data bus may be used according to your own FPGA data format requirements.

The following table shows the output signals created by the example VHDL.

VHDL Signal	Signal Function
GD9_CH0_EN	Data Valid For Channel 0
GD9_CH1_EN	Data Valid For Channel 1
GD9_CH2_EN	Data Valid For Channel 2
GD9_CH3_EN	Data Valid For Channel 3
GD9_UNS_DATA(11 downto 0)	12-bit Unsigned A-to-D Data
GD9_SGN_DATA(11 downto 0)	12-bit Signed A-to-D Data