HPSDR – Ethernet DFC Protocol

Revisions

Rev	Date	Changes	By
1.0	4 Aug 14	First release for Angelia (ANAN-100D) board	VK6PH
1.1	24 Nov 14	Added Discovery option and Hermes version	VK6PH
1.2	17 Jan 15	Reversed order of Bytes so not required at PC	VK6PH
1.3	16 Jul 17	Changed payload from 1,500 to 8,192 bytes Reformatted command data Corrected Discovery reply packed ID to 0xEFFF from 0xE000 Corrected Discovery reply packed zero padding Advised 31dB attenuator is set to 0dB Replaced counter with 21.073 MHz sinewave signal Add Transmit Drive Level control Added Transmit data format.	VK6PH

Protocol Overview (V1.3):

This protocol is for experimental purposes. It enables an openHPSDR board e.g. Hermes or ANAN-10/100/200 to send raw ADC samples at 61.44 Msps via raw Ethernet frames to an associated PC or Single Board Computer.

The ADC(s) on the board are clocked at 122.88 MHz. 16 bit samples are fed to the input of a FIFO that is clocked at 61.44 MHz in order to decimate by two. The FIFO output is clocked at 125 MHz and provides 8 bit data for the Ethernet controller.

Hence the Ethernet bit rate is 61.44 x 16 = 983.04 Mbps.

Data may also be sent from the associated PC to a FIFO the output of which is connected to the DAC on the attached hardware. Raw DAC data is sent to the DAC which is clocked at 61.44 Msps.

Data is sent to/from the hardware using raw Ethernet frames. This minimizes the packet overhead in order to provide the fastest data rate.

The 31dB onboard receive attenuator is set to 0dB.

If an Alex board is connected to the hardware then the 30MHz LPF is engaged and no HPFs are selected.

Protocol – From PC to board

Discovery

In order to determine the MAC address of the board the PC sends the following raw Ethernet packet:

<FF:FF:FF:FF:FF:FF> <source MAC> <0xE000> <0x00> <60 bytes of zero>

Where <source MAC> is the MAC address of the associated PC.

The board will respond with

<PC MAC> <Board MAC> <0xEFFF> <0x00> <64 bytes of zero>

The ‘Board MAC’ can then be used with subsequent commands.

Commands

The following commands are then available:

<board MAC> <source MAC> <0xE000> <n[7:0]> <Drive level[7:0]> < 58 bytes of zero>

Where:

n[0] = 0 – do not send data, 1 – send data
n[1] = 0 – send ADC data, 1 – send sinewave data
n[2] = 0 – send ADC0 data, 1 – send ADC1 data (ADC1 data only relevant for dual ADC hardware)

Alternatively, the data stream can be started immediately as follows:

<FF:FF:FF:FF:FF:FF> <source MAC> <0xE000> <0x01> <Drive Level> <59 bytes of zero> for ADC0

<FF:FF:FF:FF:FF:FF> <source MAC> <0xE000> <0x05> <Drive Level> <59 bytes of zero> for ADC1

<FF:FF:FF:FF:FF:FF> <source MAC> <0xE000> <0x03> <Drive Level> <59 bytes of zero> for sinewave data

When transmitting, Drive Level can be set from 0x00 (lowest RF output) to 0xFF (highest RF output). Drive level only needs to be sent once initially and then whenever it changes.

Transmit data

Transmitter data may be sent from the PC to the DAC in the hardware as follows:

<Board MAC> <PC MAC> <0xE001> <payload = 8,192 bytes>

The payload consists of consecutive DAC samples in the form <DAC[7:0], DAC[15:8]>

The hardware DAC is clocked at 61.44 Msps. In which case it is not recommended that, due to the close spacing of the first alias signal, the transmitter is operated above 25 MHz.

NOTE: In order to ensure that the Tx FIFO in the FPGA is not over/under run then Tx packets should be sequenced such that for each ADC packet that is received a DAC packet is sent.

Protocol – From Board to PC

When the send bit is set the board will send raw Ethernet frames to the MAC address of the associated PC as follows:

<source MAC> <board MAC> <0xEFFF> <payload = 8,192 bytes>

When the source is set to ADC the payload consists of consecutive ADC samples in the form <ADC[7:0], ADC[15:8]>.

When the source is set to sinewave the payload consists of consecutive samples from a full scale, 16 bit, 21.073 MHz sinewave that is sampled at 61.44Msps and has the form <sine[7:0], sine[15:8]>. This facility enables the user to check frequency accuracy, full scale amplitude and DSP functions.

NOTE: The FPGA code does not presently provide a facility to load new versions of code. In which case to load new version of code, or revert to a previous version, Bootloader mode will be required.