

## Overview

This is an example using Impulse C to create a hardware module that:

- 1) Parses Ethernet packets looking for specific IPv4 UDP packet. On match sends payload on for pattern match.
- 2) Look for payload to match a specific pattern triggering a message on match.

This is effectively the same functionality as the original ANTS demo's `sf_ds_header_match_msg_extract` (left intact with trigger ignored) and `sf_ds_sample_algo` (removed) Verilog blocks.

## Installation

Prerequisite step: The following steps are only what are required to modify the ANTS demo to use an equivalent Impulse C module. Before proceeding, please be sure that you have a functional FDK v3.1.1 ANTS demo that has had hardware and software built and runs successfully.

## Impulse C

After installing CoDeveloper, a patch is required to be installed to provide lower latency connections between processes. To install the patch:

- 1) Extract the `CoDeveloper_Patch.zip` into the CoDeveloper installation directory typically "`C:\Impulse\CoDeveloper3`"

## Files

The installed patch files overwrite the existing files:

- `Architectures\VHDL\Altera\lib\impack.vhd`
- `Architectures\VHDL\Altera\lib\stream.vhd`

## Hardware

Hardware file installation steps:

- 1) Install files using the overlay `sorrento_fdk_release_pkg.tar` file:
  - a. `cd <full path>/sorrento_fdk_rtl_release_area`
  - b. `tar -xvf <path to demo files>/sorrento_fdk_release_pkg.tar`
- 2) Edit the `sorrento_fdk_release_pkg/quartus/file_list/ants_file_list` file and add the line at the end of the file:
  - a. `set_global_assignment -name QSYS_FILE ../../rtl_source/impulse/impc_qsys.qsys`

## Files

The installed hardware files are:

- `sorrento_fdk_release_pkg/rtl_source/reference_design/sf_fdk_ds_ref_design_core.sv` – modified core design to instantiate `impc_qsys.qsys` system
- `sorrento_fdk_release_pkg/rtl_source/impulse/impc_qsys.qsys` - Top Qsys design
- `sorrento_fdk_release_pkg/rtl_source/impulse/ip` - exported Impulse C module from CoDeveloper

## Software

Software file installation steps:

- 1) Install files using the overlay sorrento\_fdk\_release\_pkg.tar file:
  - a. `cd <top software directory above sfaoesw-3.1.1.1003.dev>`
  - b. `tar -xvf <path to demo files>/sfaoesw-3.1.1.1003.dev.tar`

## Files

Modified software files to add communication from host software with Impulse C module in hardware:

- ants\_ds\_sample\_app/ants\_ds\_sample\_app/hal/src/ants\_hal\_api.c
- ants\_ds\_sample\_app/ants\_ds\_sample\_app/hal/include/ants\_reg\_access.h
- ants\_ds\_sample\_app/ants\_ds\_sample\_app/hal/src/ants\_reg\_access.c

## Build

### AOE2 Hardware

After files are installed, build normally from within the quartus/build directory using the command:

```
../scripts/qsfl_rel.tcl ANTS
```

### Host Software

After files are installed, build normally from within the sfaoesw-3.1.1.1003.dev directory using the make command.

## Impulse C Module

To re-export the Impulse C module used from CoDeveloper:

- 1) Open the Impulse\_C\ants\_trigger.icProj project (Windows)
- 2) Use the GUI: Project->"Export Generated Hardware (HDL)".
  - a. The exported module will appear in the subdirectory quartus/ip
- 3) Copy the entire "ip" directory to the "sorrento\_fdk\_release\_pkg/rtl\_source/impulse" being used

## Run

### AOE2 Hardware

Program the AOE2 normally starting from the newly generated .rpd file.

### Host Software

Run the host ANTS demo normally running the ants\_delegated\_client and ants\_delegated\_server executables on separate machines.