OpenCPI
fmcomms_2_3_tx.rcc Test App Guide

Version 1.4

Revision History

Revision	Description of Change	Date
v1.4	Initial release.	9/2018
v1.5	Version bump only.	4/2019

1 Description

This application is intended to perform a hardware-in-the-loop test of the fmcomms_2_3_tx.rcc worker. It tests to ensure that the default values and the expected possible min/max values are applied successfully for the various RF frontend parameters. It also tests that fmcomms_2_3_tx.rcc property writes do not override the fmcomms_2_3_rx.rcc worker property writes when both exist in the OAS. Note that TX data fidelity is not verified within this test.

2 Hardware Portability

This application is intended to test fmcomms_2_3_tx.rcc, which is by design specific to the FMCOMMS2/3 cards. There is nothing about the application that precludes use of any particular HDL platform, assuming that platform includes an FMC slot on which an FMCOMMS2/3 card may be used.

3 Execution

3.1 Prerequisites

The following must be true before application execution:

- An OpenCPI platform is available w/ an FMCOMMS2 or FMCOMMS3 card plugged into its FMC slot.

- The following assets are built for the HDL/RCC platform which correspond to the intended HDL/RCC runtime containers, and their build artifacts (FPGA bitstream file/shared object files) are contained within the directory list of the OCPI_LIBRARY_PATH environment variable.

  - `empty` assembly with one of the *`fmcomms_2_3`* containers for the desired HDL platform
  - `fmcomms_2_3_tx.rcc` for all `TYPE_p` configurations (both fmcomms2 and fmcomms3)
  - `fmcomms_2_3_rx.rcc` for all `TYPE_p` configurations (both fmcomms2 and fmcomms3)
  - `ad9361_config_proxy.rcc`

- The application itself (`fmcomms_2_3_tx_test`) must be built.

- The current directory is the applications/fmcomms_2_3_tx_test directory.

3.2   Command(s)

The full test is run with the following command:

```
./<target-dir>/fmcomms_2_3_tx_test
```

A software-only test can be run which performs testing on the fmcomms_2_3_tx.rcc software calculation routines (no hardware actuation).

```
./<target-dir>/fmcomms_2_3_tx_test swonly
```

4   Verification

An application exit status of 0 indicates success, and non-zero indicates failure. Either PASSED or FAILED will also be printed to the screen.

5   Troubleshooting

The application will occasionally fail with the following printed to screen:

```
      variable: actual_ad9361_config_proxy_val, expected value: 2083340, actual value: 2083340 \
        EXPECTED
Calibration TIMEOUT (0x16, 0x10)
Exception thrown: Worker ad9361_config_proxy produced error during execution: \
    ad9361_set_tx_sampling_freq() returned: -110
FAILED
```

The fmcomms_2_3_tx.rcc endpoint proxy controls the ad9361_config_proxy.rcc device proxy which wraps the ADI No-OS library for SPI command/control of the AD9361. A known defect of the AD9361 hardware/No-OS library is that the AD9361 hardware will occasionally fail to calibrate when No-OS sets low sample rate values (such as 2083340 sps in example above), resulting in a calibration timeout. When this occurs, No-OS prints to the screen:

```
Calibration TIMEOUT (0x16, 0x10)
```

Neither of the ad9361_config_proxy.rcc/fmcomms_2_3_tx.rcc workers yet implement a mechanism for overcoming this AD9361/No-OS shortcoming.