

OpenCPI
FSK App Getting Started Guide
(E310 Supplement)

Version 1.4

Revision History

Revision	Description of Change	Date
v1.1	Initial Release	3/2017
v1.2	Updated for OpenCPI Release 1.2	8/2017
v1.3	Updated for OpenCPI Release 1.3	1/2018
v1.3.1-E3XX	Updated for E310 support	3/2018
v1.4	Updated with simplifications and references to assets' document	9/2018

Table of Contents

1	References	4
2	Overview	5
3	Prerequisites	5
4	Build the Core Project	5
5	Build the Assets Project	5
6	Build the E310 BSP Project	5
7	Build the Assets for E310	5
8	Build the FSK Application Executable	5
9	Running the Application	6
9.1	View the Results	6

1 References

This document assumes a basic understanding of the Linux command line (or “shell”) environment. The reference(s) in Table 1 can be used as an overview of OpenCPI and may prove useful.

Table 1: References

Title	Published By	Link
Getting Started	ANGRYVIPER Team	Getting_Started.pdf
Installation Guide	ANGRYVIPER Team	RPM_Installation_Guide.pdf
Acronyms and Definitions	ANGRYVIPER Team	Acronyms_and_Definitions.pdf
Overview	OpenCPI	http://opencpi.github.io/Overview.pdf
FSK App ¹	OpenCPI	FSK_app.pdf

¹Provides details of the “FSK App” reference application

2 Overview

The purpose of this document is to provide a compact set of instructions to build, run, and verify the OpenCPI FSK App reference application.

3 Prerequisites

This document assumes that the OpenCPI framework has been installed. The application is supported on the Ettus E310 platform.

4 Build the Core Project

If the Core Project has not been created yet, follow the instructions in the OpenCPI Getting Started Guide. Once the Core project has been created, the following `ocpidev` command can be used to build the primitives and workers required by the FSK app. Navigate to the Core project directory and run the command:

```
ocpidev build --rcc --rcc-platform xilinx13_4 --hdl --hdl-platform e3xx --no-assemblies
```

Note: The `--no-assemblies` argument excludes the creation of executable bitstreams.
This step takes approximately 20 minutes to complete.

5 Build the Assets Project

If the Assets Project has not been created yet, follow the instructions in the OpenCPI Getting Started Guide. Once the Assets project has been created, the following `ocpidev` command can be used to build the primitives and workers required by the FSK app. Navigate to the Assets project directory and run the command:

```
ocpidev build --rcc --rcc-platform xilinx13_4 --hdl --hdl-platform e3xx --no-assemblies
```

Note: The `--no-assemblies` is necessary here because we cannot build assemblies for the e3xx platform until that platform has itself been built.
This step takes approximately 50 minutes to complete.

6 Build the E310 BSP Project

For the Ettus E310, there is a BSP project that contains a copy of the FSK assemblies, so you can build it *after* the assets project. Navigate to the E310 BSP project directory and run the following command:

```
ocpidev build --rcc --rcc-platform xilinx13_4 --hdl --hdl-platform e3xx
```

This step takes approximately 45 minutes to complete.

7 Build the Assets for E310

Now that the E310 platform has been built, we can return to the assets project and build the `fsk_filerw` assembly for the E310 platform:

```
ocpidev build hdl assembly fsk_filerw --hdl-platform e3xx
```

This step takes approximately 10 minutes to complete.

8 Build the FSK Application Executable

Next, the executable for the FSK Application must be built. Navigate to the E310 BSP project's `applications/FSK` directory of the Assets project and run the command:

```
ocpidev build --rcc-platform xilinx13_4
```

If successful, a new directory named `target-xilinx13_4` will contain the executable.

9 Running the Application

For more information, see the full *FSK_App_Getting_Started_Guide* in the Assets project. You can also reference the E310 *FSK_app* document, or run “**make show**” (on the host) in the E310 BSP’s **applications/FSK** directory for more E310 specific information (especially **OCPI_LIBRARY_PATH** settings, etc...).

In short, once the radio is set up, run the application *on the embedded radio* by running the executable and passing a “mode” such as **filerw**, **tx**, **rx** or **txrx**. For example:

Connect port TRXA to RX2A using an SMA connector, enter the E310 BSP project’s **applications/FSK** directory, and run the following:

```
./target-xilinx13_4/FSK txrx
```

At the runtime, choose TX port: TRXA and RX port: RX2A. The default values should suffice for the remaining options.

9.1 View the Results

After the application completes, the output can be found in the **applications/FSK/odata** directory. To view the results on the Development Host, navigate to the E310 BSP project’s **applications/FSK** directory and execute the command:

```
eog odata/out_app_fsk_txrx.bin
```



Figure 1: FSK input file

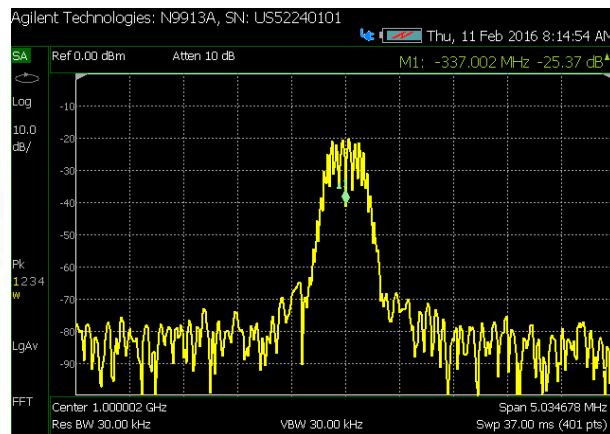


Figure 2: Output of FSK App RF transmit