

OpenCPI  
FSK App Getting Started Guide  
(E310 Supplement)

Version 1.5

*Revision History*

Revision	Description of Change	Date
v1.4	Updated with simplifications and references to assets' document	9/2018
v1.5	Version bump only	4/2019

## Table of Contents

<b>1</b>	<b>References</b>	<b>4</b>
<b>2</b>	<b>Overview</b>	<b>5</b>
<b>3</b>	<b>Prerequisites</b>	<b>5</b>
<b>4</b>	<b>Build the OpenCPI Core Project</b>	<b>5</b>
<b>5</b>	<b>Build the OpenCPI Assets Project (Excluding Assemblies)</b>	<b>5</b>
<b>6</b>	<b>Build the OpenCPI E310 BSP Project</b>	<b>5</b>
<b>7</b>	<b>Build the Assets Project's fsk_filerw Assembly</b>	<b>5</b>
<b>8</b>	<b>Build the E310 BSP Project's FSK Application Executable</b>	<b>6</b>
<b>9</b>	<b>Running the Application</b>	<b>6</b>
9.1	Example txrx Mode Usage with SMB loopback . . . . .	6
9.2	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

<b>Title</b>	<b>Published By</b>	<b>Link</b>
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	<a href="http://opencpi.github.io/Overview.pdf">http://opencpi.github.io/Overview.pdf</a>
FSK App <sup>1</sup>	OpenCPI	FSK_app.pdf

<sup>1</sup>Provides 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 OpenCPI 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. Run the following:

```
ocpidev -d <core-project-dir> build --rcc-platform xilinx13_4 --hdl-platform e3xx
```

This step takes approximately 20 minutes to complete.

## 5 Build the OpenCPI Assets Project (Excluding Assemblies)

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. Run the following:

```
ocpidev build -d <assets-project-dir> --rcc-platform xilinx13_4 --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 OpenCPI E310 BSP Project

For the Ettus E310, there is a BSP project that contains a copy of the FSK assemblies. This project must be built *after* the assets project. Run the following:

```
ocpidev build -d <e310-project-dir> --rcc-platform xilinx13_4 --hdl-platform e3xx
```

Note that this step builds the assemblies necessary for the application's `tx`, `rx`, and `txrx` modes on the E310 (and not the `filerw` mode, which is built via the instructions in 7). This step takes approximately 45 minutes to complete.

## 7 Build the Assets Project's `fsk_filerw` Assembly

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. Run the following:

```
ocpidev build -d <assets-project-dir> hdl assembly fsk_filerw --hdl-platform e3xx
```

Note that this step builds the assembly necessary for the application's `filerw` mode on E310 (and not the `tx`, `rx`, and `txrx` modes, which are built via the instructions in 6). This step takes approximately 10 minutes to complete.

## 8 Build the E310 BSP Project's FSK Application Executable

Next, the executable for the FSK Application must be built. Run the following:

```
ocpidev build -d <e310-project-dir/applications/FSK> --rcc-platform xilinx13_4
```

If successful, a new directory named `target-xilinx13_4` will be created in `<e310-project-dir/applications/FSK>` that contains 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`.

### 9.1 Example txrx Mode Usage with SMB loopback

Connect port TRXA to RX2A using an SMB cable and run the following:

```
cd <e310-project-dir/applications/FSK>
./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.

### 9.2 View the Results

After the application completes, the results can be viewed on the Development Host by running the following:

```
cd <e310-project-dir/applications/FSK>
eog odata/out_app_fsk_txrx.bin
```



Figure 1: Output file produced by successful application execution

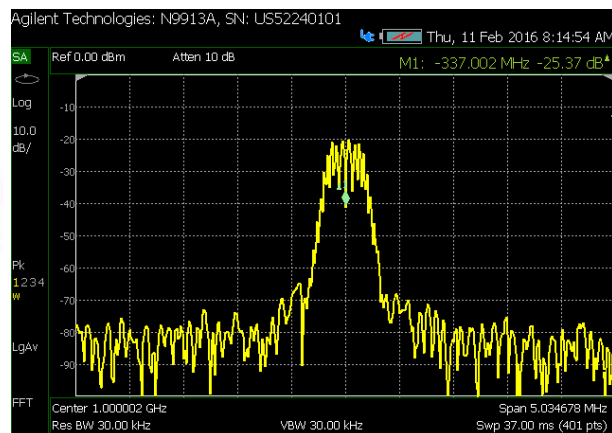


Figure 2: Output of FSK App RF transmit