Electrosense converter board

Firmware Design

|  |  |  |
| --- | --- | --- |
| **Author** | **Description** | **Date** |
| Franco Minucci | First draft | 12/01/2020 |
|  |  |  |

# 

Table of Contents

[Table of contents 1](#__RefHeading___Toc106_212709630)

# Introduction

The purpose of the converter board is to extend the frequency range of the RTL-SDR used as sensing element for each node.

The converter board consists of 3 paths:

* Up converter path: Brings low frequencies from 300kHz to 27MHz up to 30 MHz
* Mid path: Straight connection between the antenna to the RTL-SDR
* Down converter path: Brigns the frequencies above 1700MHz to 1576MHz within the range of the RTL-SDR

This document describes the firmware running on the microcontroller of the extension board. The firmware runs on an STM32F103C8T6. It may vary in future versions of the extension board.

This document is divided into three sections:

* Feature list
* Implementation
* Test strategy

Section 1 describes the expected feature list in order of importance. Section 2 describes the implementation details and caveats. Section 3 describes the test strategy for each feature.

# Feature list

This section goes over the list of expected features and functionality implemented in the firmware.

## User interface

* The firmware must present a command line interface (CLI) towards the user.
* The CLI is exposed to every computer connected to the USB interface of the board.
* The USB is configured to appear to the connected computer as a serial port device running at 115200 kbps.
* The CLI must be bit rate independent.
* The CLI exposes the following prompt to the user: “esense>”
* The CLI accepts the following list of commands:
  + help: shows a list of all available commands
  + reboot: restarts the microcontroller
  + syslog: display the events log of the device
  + tasks: displays a list of running tasks
  + sanity: performs a sanity check
  + license: shows the firmware license
  + mco: enables or disables the input clock of the upconverter
  + Sof: ?
  + convert: configure the input frequency and returns the converter frequency (to which the RTL-SDR must be tuned)
  + temp: Shows the last reading from the temperature sensor
  + gpio: drives a gpio
  + i2c: ?
  + Max: ?
* The CLI runs in its own task

## RF Switch configuration

* The firmware must be able to configure the RF switch U11 (SKY13322-375LF) in the correct position.
* The switch is controlled by 3 signals:
  + SW\_SW: Connects the downconverter to the RTL-SDR
  + SW\_BYPASS: Connects the straight path to the RTL-SDR
  + SW\_MIX: Connects the upconverter to the RTL-SDR
* The 3 signals cannot be all HIGH at the same time

## Band Selection

* The firmware must control the switch U7 (SKY13286-359LF) to select the upper of lower band of the diplexer
* The selection operation is transparent for the user of the CLI
* The “low band” indicates frequencies below 3500MHz
* The “high band” indicates frequencies above 3500MHz
* The band is selected through the signal BANDSEL:
  + BANDSEL = 1 => LOW band
  + BANDSEL = 0 => HIGH band

## Temperature sensor

The temperature sensor is connected to the I²C bus with address: 0x18.