Lime Microsystems Limited

Surrey Technology Centre
Occam Road
The Surrey Research Park
Guildford, Surrey GU2 7YG
United Kingdom

Tel: +44 (0) 1483 685 063
e-mail: enquiries@limemicro.com

LMS8001 Companion Board
Quick Starter Manual

Chip version:	LMS8001A
Chip revision:	2
Document version:	2.0
Document revision:	0

REVISION HISTORY

The following table shows the revision history of this document:

Date	Version	Description of Revisions
05/10/2017	1.0.0	Initial version created.
15/06/2018	1.0.1	Minor corrections
01/07/2025	2.0.0	Minor updates: 1. A new USB connector (USB type C) replaced the old one 2. New band (6.5-7.5 GHz) replaced the old one (5-6 GHz) 3. New IF frequency (2.1 GHz) replaced the old one (1.2 GHz)

Contents

1

Overview

This document describes how to make a quick start with the LMS8001A using the LMS8001 Companion Board (CMP) module.

2

Installing the LMS8001 Control Software

2.1 Introduction to installing the software

To operate the CMP board, the following items are needed:

- The main LMS8001 control software, *lms8suite*, which provides a GUI (Graphic User Interface) to control the chip.
- The Virtual COM Port driver, which provides an interface between the PC and the CMP SPI microcontroller.
- The CMP microcontroller firmware, which is preinstalled on the board prior to shipping.

Section 2.2 describes the how to download the *lms8suite* software.

Section 2.3 describes the Virtual COM Port driver installation procedure on the Windows Operating System. Section 2.4 describes how to identify which USB port is being used.

Section 2.5 describes installing the USB driver in a typical Linux distribution.

Section 2.6 describes how to start the “LMS8001 Control Software”. Section 2.7 describes how to connect the CMP with the “LMS8001 Control Software” via the USB interface.

A simple demonstration of the “LMS8001 Control Software” is given in Chapter 3.

Detailed description of the “LMS8001 Control Software” is given in Chapter 5.

2.2 Obtaining the Software

CMP control software can be obtained either by downloading the binary (Windows only), or by compiling the source.

2.2.1 Download the binary

Control software binary for Windows can be downloaded from http://downloads.myriadrf.org/builds/lms8suite/. Please note that in the case of Windows a hardware driver must also be installed as a separate part of the process.

Binaries for Linux are not available for download.

2.2.2 Compile from source

The source code is available at https://github.com/myriadrf/lms8suite.

Please follow the compilation instruction for the Lime Suite (document 'Lime Suite Software Compilation Guide') available at:

https://github.com/myriadrf/LimeSuite/blob/master/docs/Lime_Suite_Compilation_Guide.pdf

Make minor adjustments so that the lms8suite source code is compiled.

For compilation under Linux, follow the Chapter 3 of the 'Lime Suite Software Compilation Guide' document, but with the following changes to the section 3.3:

1. If git is not already installed on your PC, then install it (otherwise skip this step):

   ```
   sudo apt-get install git
   ```

2. Obtain source code from git repository:

   ```
   git clone https://github.com/myriadrf/lms8suite
   ```

3. Go to 'LMS8Suite/src' directory:

   ```
   cd LMS8Suite/src
   ```

4. Create 'build' directory and descend into it:

   ```
   mkdir build
   cd build
   ```

5. Inside the 'build' directory execute command:

   ```
   cmake ..
   ```

6. Inside the 'build' directory execute command:

   ```
   make
   ```

7. Wait for the compilation process to complete
8. Binary lms8suite can be found in "LMS8Suite/src/build/bin' directory.

2.3 Virtual COM port driver installation

This section describes the driver installation under Windows operating system. Linux users can skip to section 2.5.

In the following the repository https://github.com/myriadrf/LMS8001-Companion will be denoted as simply a “repository”.

LMS8001-CMP communicates with PC via USB Virtual COM port. To install the required virtual com port driver on your system the following steps need to be performed:

1. Go to drivers\STM32_VCP\ folder of the repository;
2. Depending on the version of windows that your system is running go to \Win7 or \Win8 directory; For Windows 10 use the \Win8 version.
3. If you are running 32-bit windows launch dpinst_x86.exe, or if you are running 64-bit windows launch dpinst_amd64.exe.
4. Click ‘Next’ when driver installation wizard appears (Figure 2.1) and wait.

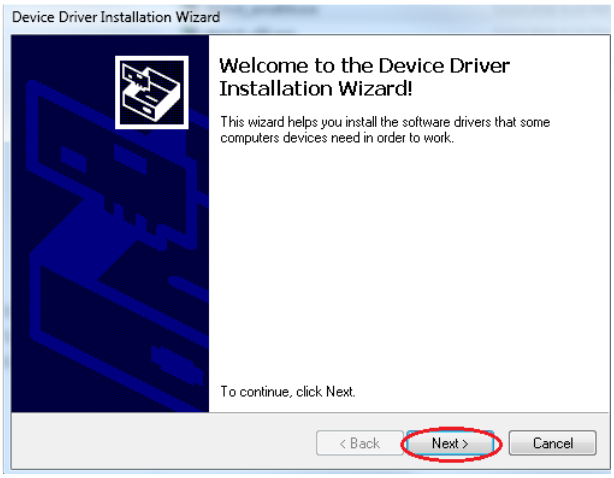


Figure 2.1: Virtual COM Port Driver Installation

5. Click ‘Finish’ when the window shown in Figure 2.2 appears to complete installation.

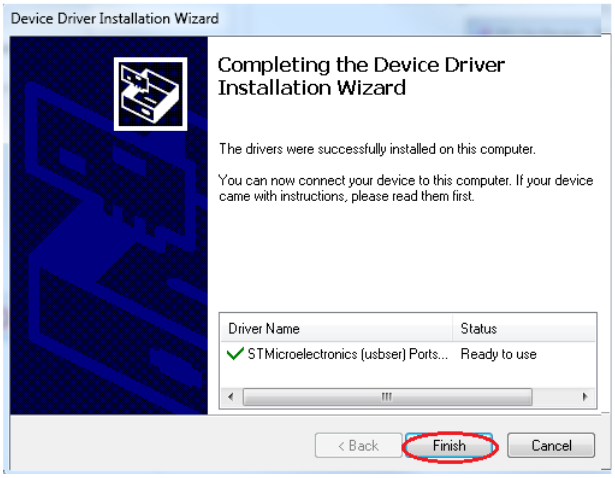


Figure 2.2: Virtual COM Port Driver Installation Finished

2.4 Determining Serial Port

After the driver installation, Windows will assign a serial port to your CMP board. To check your board's serial port number, please follow these steps:

1. Go to Control Panel > System > Device Manager
2. Locate USB Virtual Serial Port under Ports (COM & LPT)

Note the example presented in Figure 2.3 the board is assigned port COM3 .

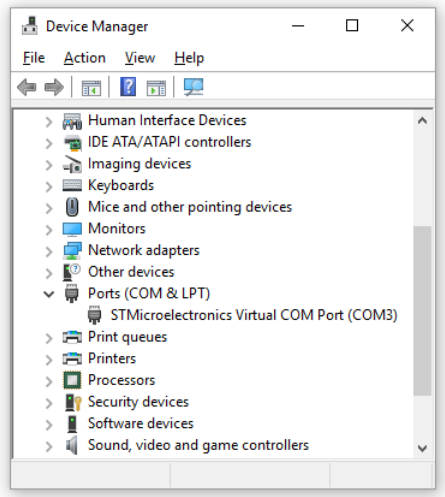


Figure 2.3: Determine the COM Port

2.5 Linux Setup

For Linux users, there is no need to install USB drivers, as the system will assign drivers automatically once the CMP board is connected to PC.

To determine port number the easiest is via the command line and type command:

```
$ setserial -g /dev/ttyS[0123]
```

2.6 Starting the LMS8001 Control Software

Apply the supply to the CMP board, as described in Section 4.1.

Run the "lms8suite.exe". The "LMS8001 GUI" application will be launched.

2.7 Connecting

Once the Windows driver is installed and the control software has been lunched, click on Options>Connection Settings. The Connection Setting windows will pop-up (Figure 2.4).

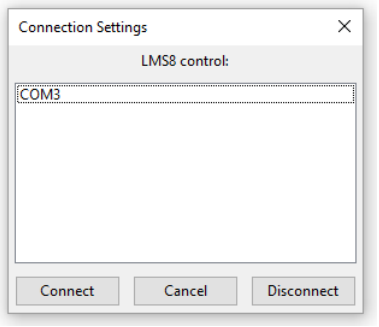


Figure 2.4: LMS8001 GUI Connection Settings

Select the dedicated USB port number of the CMP board and press “Connect”.

The GUI device name and firmware version will appear in the bottom of the application window (Figure 2.5) once the connection with the board is established.

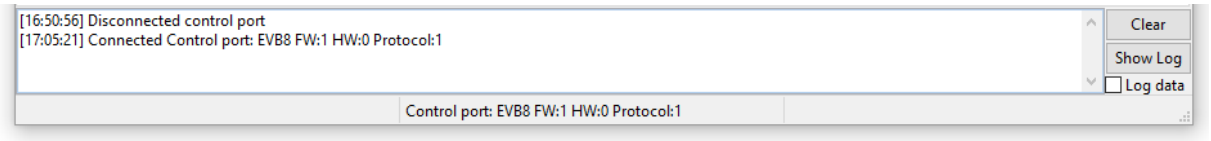


Figure 2.5: Details of the connected control port

3

Getting Started with the LMS8001 Control Software

In this chapter the very basic information regarding the CMP board and control software will be presented. The purpose is to get the CMP up and running with as little effort as possible, emphasizing the essential information. Detailed information on the board and the control software will be provided in the following chapters.

3.1 Introduction

The tabs that are relevant to basic operation are:

- PLL Configuration – Configures the PLL
- PLL Profiles – PLL settings can be stored in 8 profiles. By choosing the profile all the settings of the chosen profile are applied.
- LMS8001A – Channel – Chip LMS8001A has a Low Noise Amplifier, Mixer A and B and Power Amplifier in each of the four channels. Channels are denoted as A, B, C and D.

The CMP board contains LMS8001A version of the LMS8001 chip. Therefore, the following tab is of no interest, and should not be used:

- LMS8001B – High-Linearity Mixer – This tab is of importance for LMS8001B chip

Additionally, the following tab is also not to be used:

- LDOs Configuration – The CMP board provides external supplies to the chip, so the internal LDOs should be disabled, as they are by default.

Evaluation board is designed such that it converts f_{RF} = 6.5 – 7.5 GHz (which will be referred to as 7 GHz band) or f_{RF} = 9 – 10 GHz (which will be referred to as 10 GHz band) to f_{IF} = 1.7 – 2.7 GHz in down-conversion, and vice-versa in up-conversion.

Measurement results are presented in Chapter 6.

Details about the LMS8001 chip are available in the datasheet.

Performance optimisation in other frequency bands is possible by changing the matching components on the board.

Channels on the evaluation board are set as follows:

- Channel A – Up-conversion – 10 GHz band
- Channel B – Up-conversion – 7 GHz band
- Channel C – Down-conversion – 10 GHz band
- Channel D – Down-conversion – 7 GHz band

For example, one scenario for up-conversion would require that the input signal with the f_{IF} is connected to the channel B input (CHBI), and the up-converted signal with f_{RF} is observed at the channel B output (CHBO).

For example, one scenario for down-conversion would require that the input signal with the f_{RF} is connected to the channel D input (CHDI), and the down-converted signal with f_{IF} is observed at the channel D output (CHDO).

3.2 Basic Setup

Two profiles that can be opened by the GUI are provided:

- CMP_DOWNCONVERSION_CHD.ini

  (Channel D is enabled)

- CMP_UPCONVERSION_CHB.ini

  (Channel B is enabled)

Both profiles have 8 PLL profiles setup with the following frequencies [GHz]:

6.5, 6.625, 6.75, 6.875, 7.0, 7.125, 7.25, and 7.375.

In the following the basic procedure for loading and using the predefined profiles will be presented.

After the application (lms8suite.exe) is started it should be connected to the Companion board. This is accomplished by Options->Connection Settings and then by choosing the appropriate COM port listed and clicking on Connect.

Profile is loaded by clicking the *Open* button in the upper left part of the window, and choosing the appropriate file.

The *PLL Profile* tab should look like Figure 3.1.

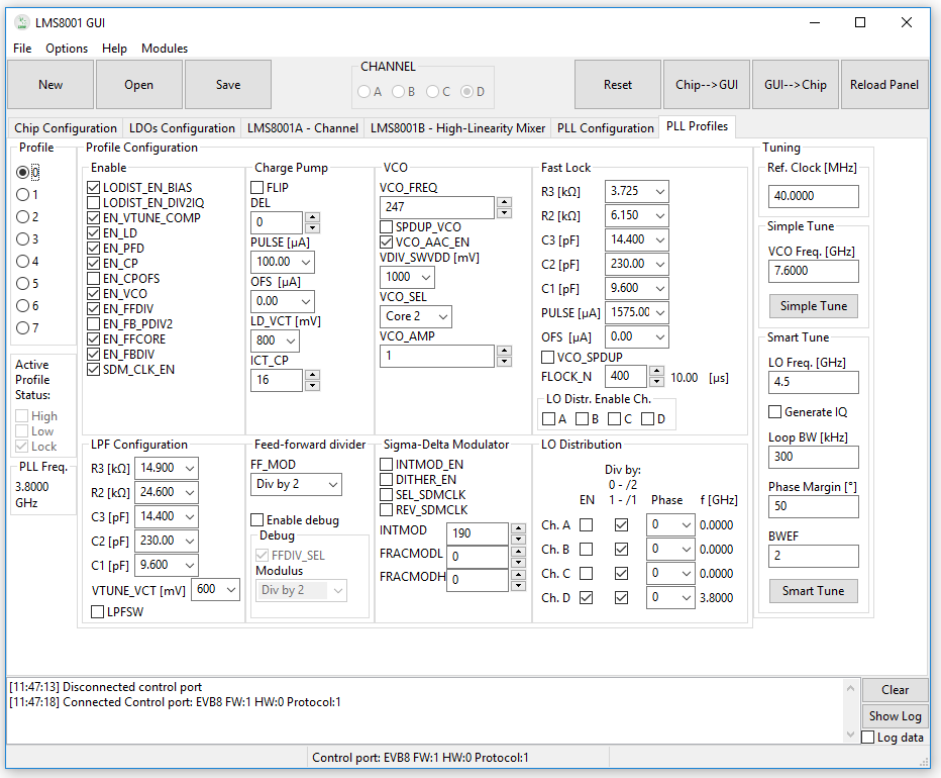


Figure 3.1: PLL Profiles tab

For each of the PLL profiles it is required to click *Simple Tune* button, to activate the automatic calibration, which will update the capacitor bank values (field VCO_FREQ), which might have small variation between the different chips and environment conditions, and lock the PLL.

Such an updated profile can be saved and used later.

Figure 3.1 shows the down-conversion case, since the LO signal is provided to the channel D. This can be observed in the *LO Distribution* section, where the *EN* is checked for the channel D.

For the up-conversion profile provided, LO will be provided to the channel B.

To select which PLL profile will be active, please switch to the PLL Configuration tab.

In the section *PLL Profile Select* click on the *Select Profile* check box, and in the combo box choose the PLL profile which will be activated, as shown in Figure 3.2.

Please note that the PLL Profiles tab does not show the active profile. It is to be used only for editing the profiles, and it shows the profile that can be edited, not the active one. As explained, the active profiles are set in the PLL Configuration tab.

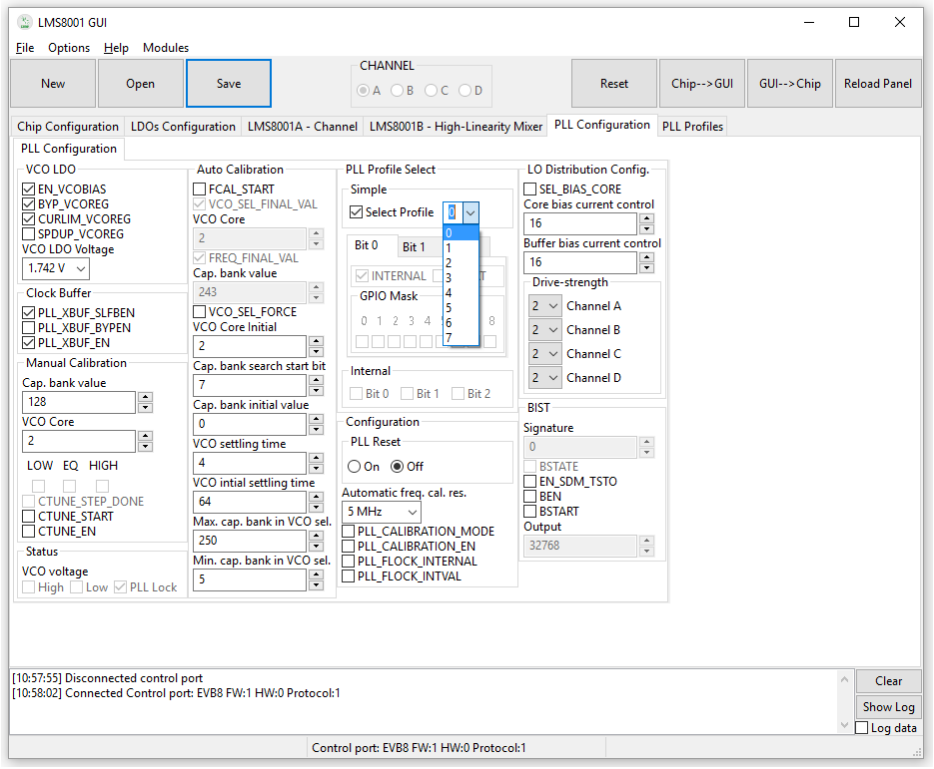


Figure 3.2: PLL Configuration tab – Choosing the active PLL profile

Following tables give examples of the combination of the PLL profile, f_{RF} and f_{IF} that can be used for measurement, both for up- and down-conversion.

Table 1: Example of the frequencies to be used for measurement

f_{RF} [GHz]	6.500	6.625	6.750	6.875	7.000	7.125	7.250	7.375
PLL Profile	0	1	2	3	4	5	6	7
f_{LO} [GHz]	4.400	4.525	4.650	4.775	4.900	5.025	5.150	5.275
f_{IF} [GHz]	2.100	2.100	2.100	2.100	2.100	2.100	2.100	2.100

3.3 Basic Manual Step-by-Step Setup

In the following, the basic setup procedure will be explained.

After starting the GUI, and connecting the evaluation board, it is recommended to reset the chip by clicking the *Reset* button in the upper right part of the window.

First, the PLL frequency will be setup.

It is suggested to enable the self-biasing at the input of the reference clock, which can be done by checking the PLL_XBUF_SLFBEN checkbox in the Clock Buffer in the PLL Configuration tab (see for example Figure 3.2).

In the *PLL Profiles* tab, the desired LO frequency should be entered in the *LO Freq. [GHz]* text input control.

After that, *Smart Tune* should be pressed.

The required PLL parameters will be recalculated, and the automatic PLL calibration procedure will be performed.

For example, for the LO frequency of 4.3 GHz, after clicking the *Smart Tune*, the window is presented in Figure 3.3.

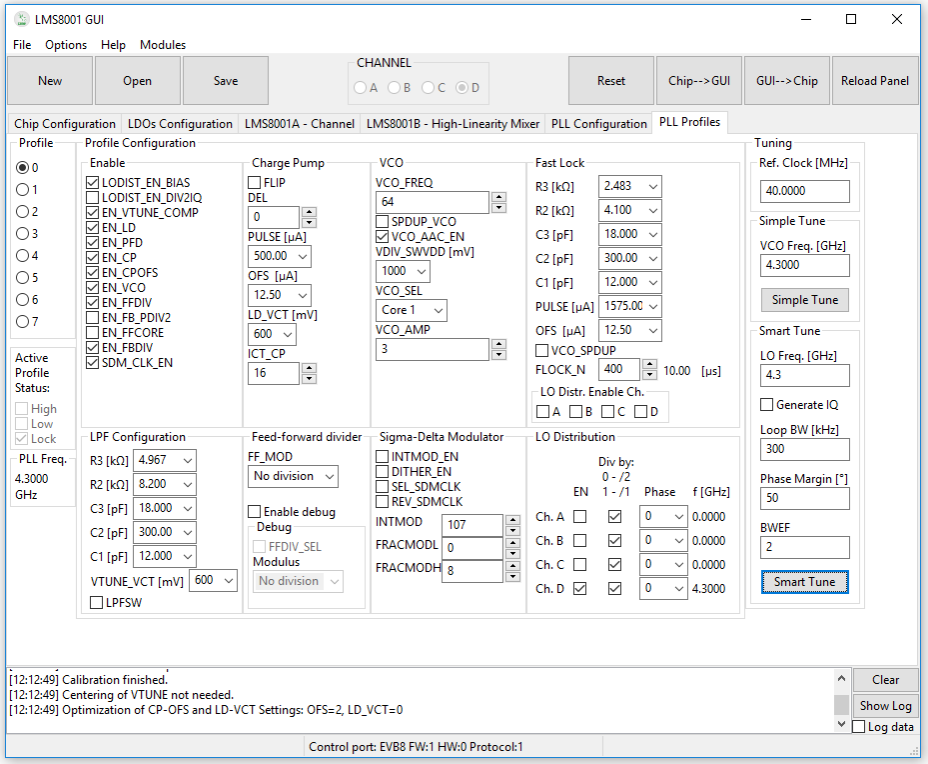


Figure 3.3: Setting the LO frequency to 4.3 GHz

In order to provide the PLL signal to the channel, click on the appropriate *EN* check box in the *LO Distribution* section.

As an example, Figure 3.3 shows the settings of the LO frequency of 4.3 GHz which are provided to the channel D.

Make sure that the appropriate PLL profile is activated in the *PLL Configuration* tab as explained in the previous section (Figure 3.2).

In order to enable channel, please go to the *LMS8001A – Channel* tab.

Channel is selected in the radio group *CHANNEL* in the upper part of the window.

Channel block diagram is shown in Figure 3.4.

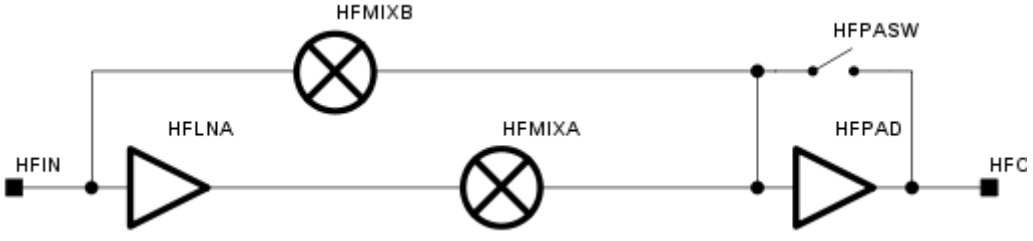


Figure 3.4: Channel block diagram.

To enable LNA uncheck the *LNA_PD* check box.

To enable MIXA uncheck the *MIXA_LOBUFF_PD* check box.

To enable MIXB uncheck the *MIXB_LOBUFF_PD* check box.

To enable PA first uncheck *PA_BYPASS* check box and then uncheck the *PA_PD* check box.

To bypass PA first check *PA_PD* check box and then check *PA_BYPASS* check box.

To disconnect the 50 Ω resistor at the PA input uncheck the *PA_R50* check box. This will usually increase the gain to some extent.

For example, one scenario for up-conversion would require Mixer B and PA to be enabled and Mixer A and LNA disabled. This example is shown in Figure 3.5 and appropriate GUI for channel B is shown in Figure 3.7 (Left).

Figure 3.5: Example of up-conversion channel configuration

Another example, scenario for down-conversion would require LNA, Mixer A and PA to be enabled and Mixer B disabled. This example is shown in Figure 3.6 and appropriate GUI for channel D is shown in Figure 3.7 (Right).

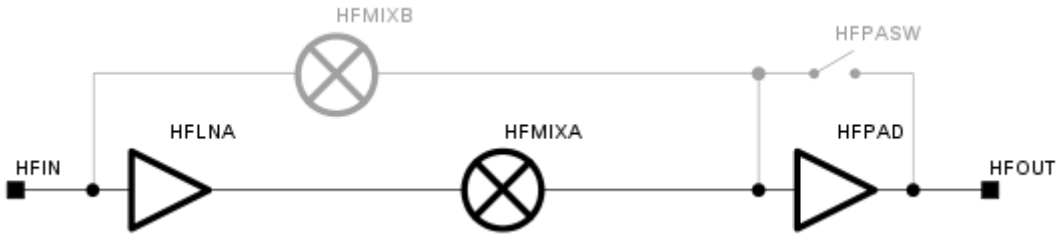


Figure 3.6: Example of down-conversion channel configuration

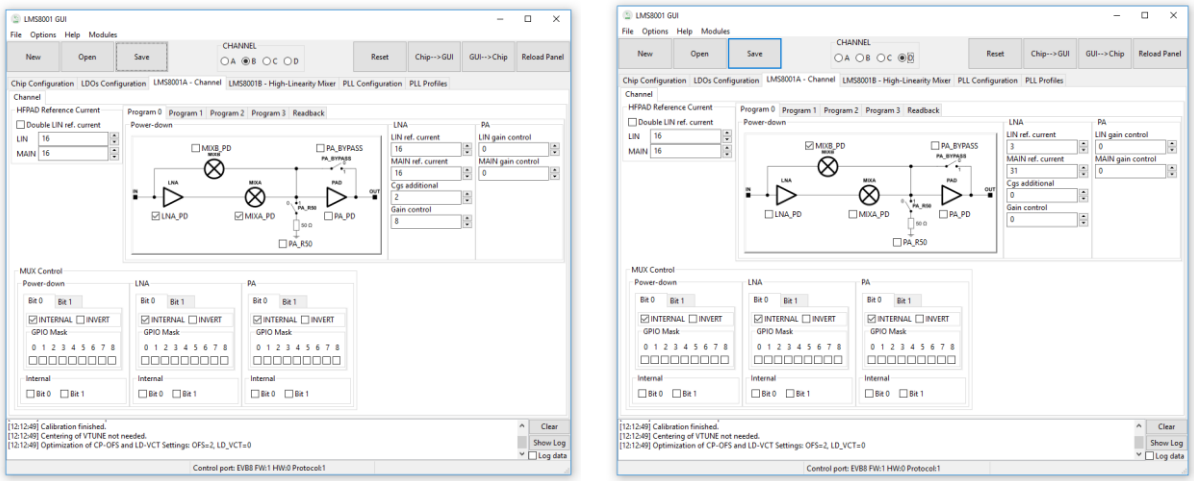


Figure 3.7: Setting GUI for up-conversion example (Left) and for down-conversion example (Right).

3.4 PLL Quadrature

LMS8001 PLL can provide the output signal in 90 degree phase steps.

In order to enable the 90 degree steps, the PLL signal needs to be scaled down in frequency by 2. This is accomplished by enabling the divide by 2 in the LO distribution.

For example, the Figure 3.8 shows the PLL setting which provides the 3.8 GHz in quadrature to the channels A and B.

Please note that the PLL frequency in this case is 7.6 GHz (FF_MOD is set to “No division”).

However, the frequency is divided by 2 in the LO distribution, provided to the channels A and B, and the phase of 90 degrees is chosen for the channel B.

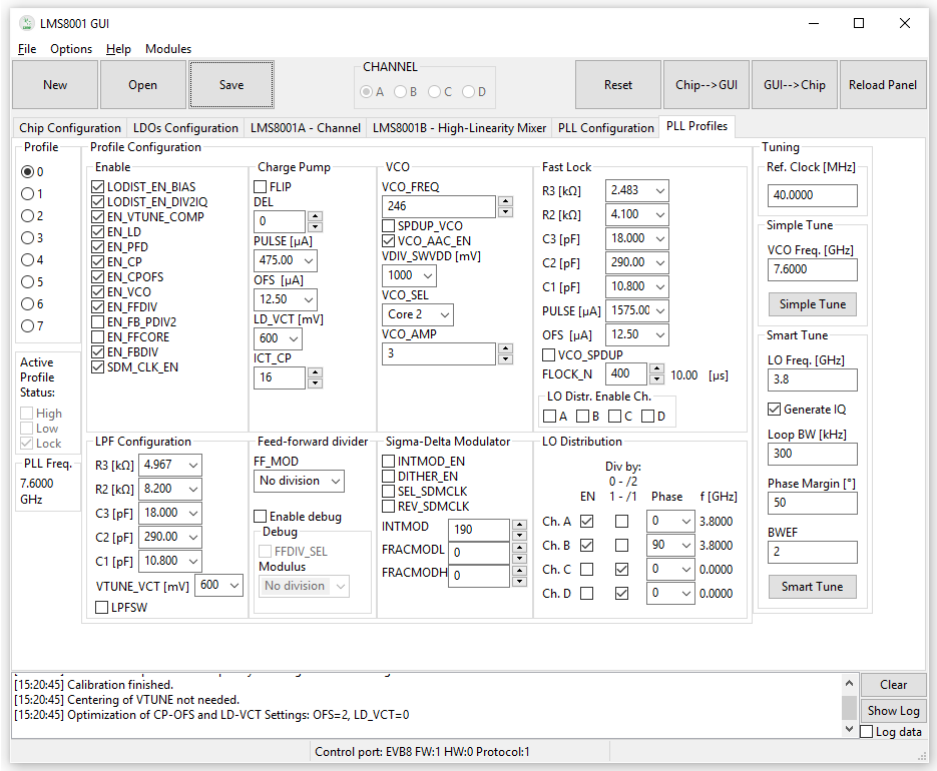


Figure 3.8: Frequency of 3.8 GHz is provided to the channels A & B in quadrature

3.5 PLL Profile Selection with GPIO

In the following it will be explained how to set up the GUI in order to be able to switch between the PLL profiles by GPIO signals.

After the setup, the GPIO0 pin will control which PLL profile will be active:

- GPIO0 = 0 then PLL profile 0 will be active
- GPIO0 = 1 then PLL profile 1 will be active

Also, the GPIO4 pin will be configured to the PLL lock detect signal, and GPIO5 pin will be configured to PLL fast lock signal.

First, the PLL profiles 0 and 1 should be setup, as previously described in Section 0.

In short, the LO frequency should be set, *Smart Tune* should be pressed, and LO distribution should be set as desired. To edit profile 1, in the radio group *Profile* value 1 should selected, and the procedure should be repeated for the profile 1.

As an example, Figure 3.9 shows one possible setup for the PLL profiles 0 and 1.

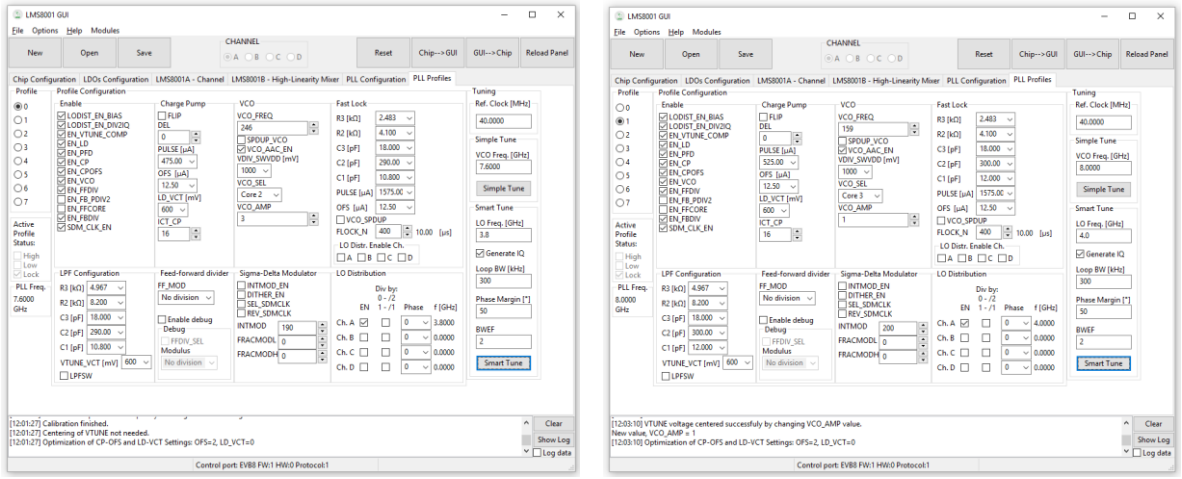


Figure 3.9: Example of the PLL Profiles 0 and 1 setup

To setup GPIO0 to toggle the active PLL profile between the profiles 0 and 1, the *PLL Configuration* tab should be set as illustrated in Figure 3.10. Namely, in the section *PLL Profile Select*, in the *Bit 0* panel, checkbox *INTERNAL* should be unchecked, and the checkbox *0* in the *GPIO Mask* should be checked.

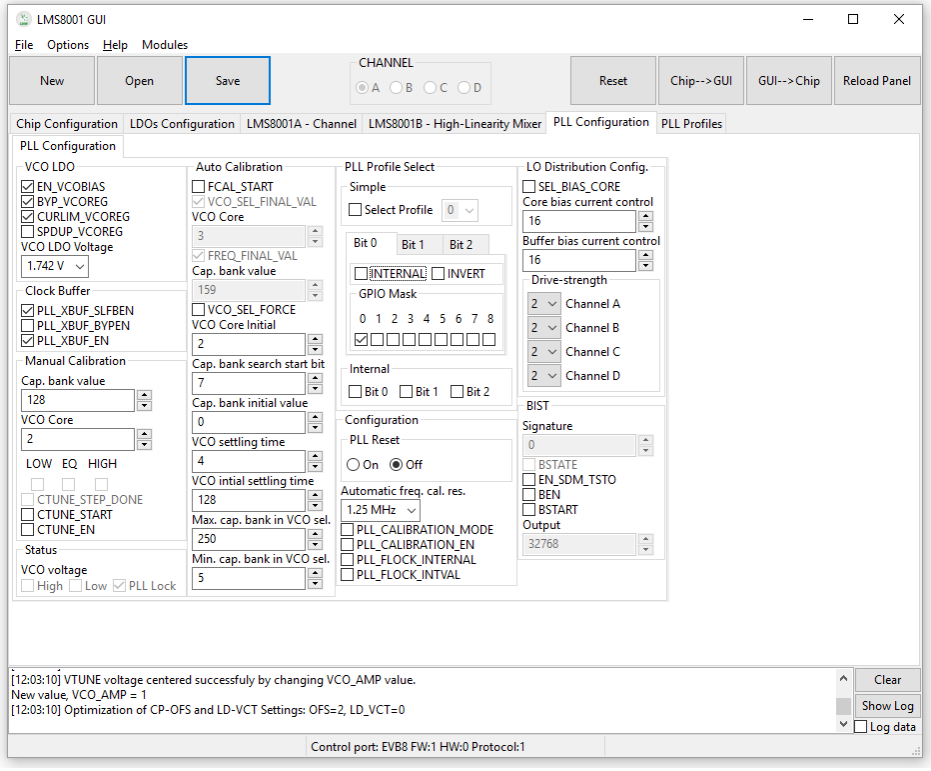


Figure 3.10: GPIO0 pin toggles the active PLL profile between profile 0 and profile 1

To setup that GPIO4 pin provides PLL lock detection, and GPIO5 pin provides PLL fast lock active signal, the *Chip Configuration* tab should be set as illustrated in Figure 3.11.

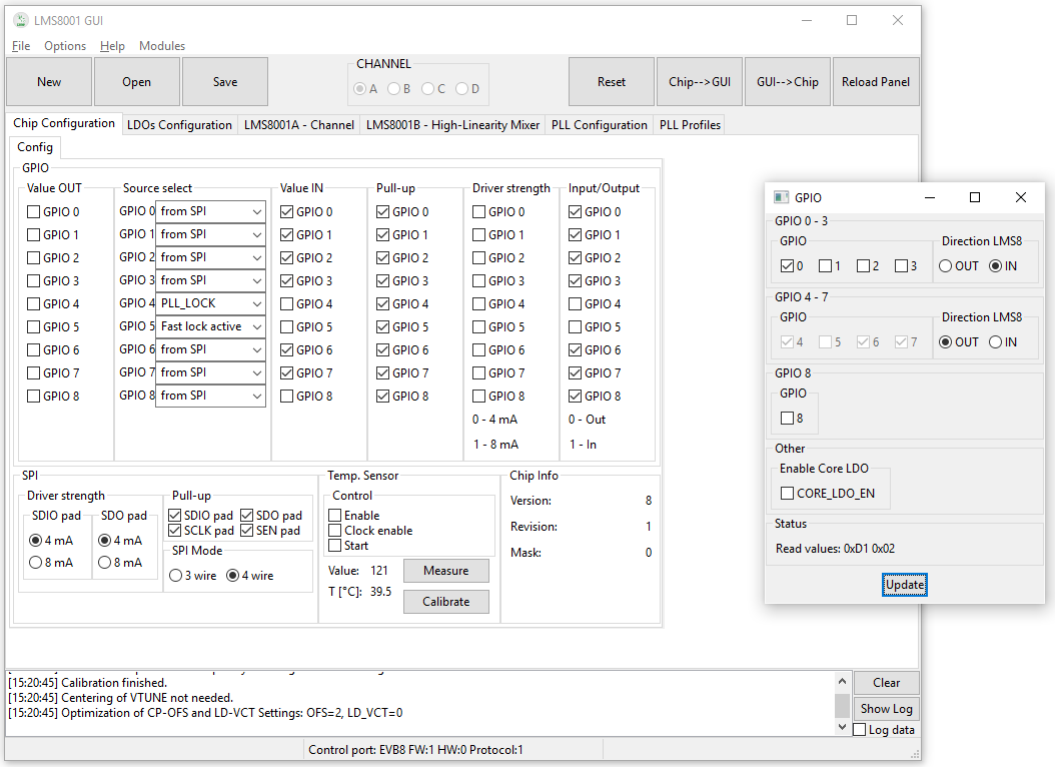


Figure 3.11: GPIO4 pin provides PLL lock detection, GPIO5 pin provides fast lock active signal

Pins GPIO4 and GPIO5 should be configured as output pins, and the source of GPIO4 should be selected as *PLL_LOCK*, while the GPIOI5 source should be selected as *Fast Lock Active*.

Finally, GPIO signals should be set and read by the microcontroller on the CMP. This can be accomplished by using the GPIO module from the GUI. To start GPIO module select the *GPIO* from the *Modules* menu in the GUI window. GPIO module is also shown in Figure 3.11.

For the GPIO group 0 – 3 select *IN*. The value for GPIO0 that will be provided by the microcontroller to the LMS8001 can be set by setting the checkbox *0*.

All the changes will be set as active by clicking the *Update* button.

GPIO signals can be accessed on the P2 connector.

4

CMP Connectors and Options

4.1 Supply Options

Supply for the LMS8001 CMP can be selected by the jumper JP1.

Microcontroller part of the CMP is always supplied through the USB connector, but the rest of the board can be supplied either from the external source, or through the USB.

To provide the supply from external source, the jumper position should be as shown in Figure 4.1. External supply should be connected to the connector J2, with middle pin connected to 5 V, while the outer pins should be connected to ground.

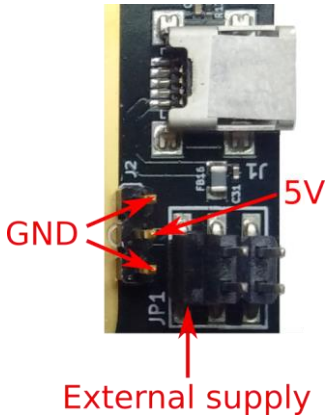


Figure 4.1: Setup for external supply for LMS8001 CMP

LMS8001 CMP can be supplied via USB. However, this is not recommended, since it may deteriorate the noise performance. In addition, limited amount of current available from the USB may not be sufficient for proper LMS8001 CMP operation. In order to supply the

LMS8001 CMP through USB, the jumper JP1 should be in the middle position, as illustrated in Figure 4.2.

Figure 4.2: Setup for USB supply for LMS8001 CMP

4.2 Introduction to the CMP Connectors and Options

Section 4.3 describes the various connectors available on the CMP.

Section 4.4 describes the hardware options available on the CMP.

The top view of the CMP board are shown in Figure 4.3.

4.3 Board Connections

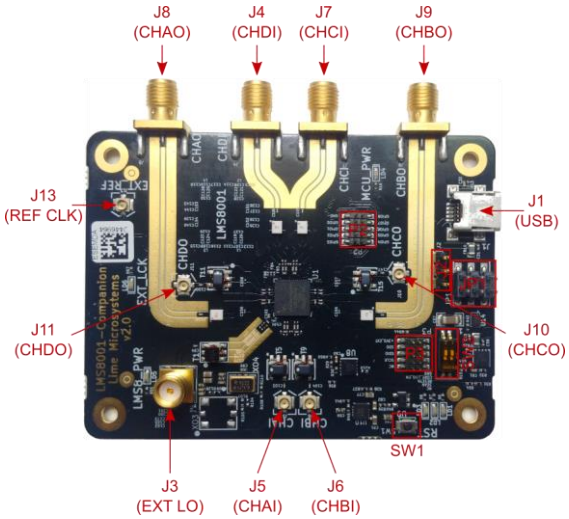


Figure 4.3: LMS8001 CMP Connectors – Top View

Table 1 describes the high-level pin assignment for each connector on the design kit.

Table 1: Design kit connectors and switches

Connector	Schematic name	Description
J1	USB	USB connector to PC (USB type C)
J2	+5V Power Supply	+5 V supply connector
J3	EXT LO	External LO input
J4	CHDI	Input – Channel D
J5	CHAI	Input – Channel A
J6	CHBI	Input – Channel B
J7	CHCI	Input – Channel C
J8	CHBO	Output – Channel A
J9	CHBO	Output – Channel B
J10	CHCO	Output – Channel C
J11	CHDO	Output – Channel D
J13	REF CLK	Reference clock input used to synchronize test equipment with CMP board to calibrate frequency error.
JP1	Supply jumper	Refer to Section 4.1.
P2	LMS8001 GPIO	LMS8001 GPIO signal pins
P3	MCU SWD	MCU SWD interface pins
SW1	MCU Reset	Reset of the microcontroller U3.
SW2	MCU BOOT	MCU boot switch

4.4 Hardware Options

4.4.1 CMP Synchronization

The LMS8001 CMP board provides options to synchronize the on-board TCXO with the base band or test equipment systems. To do that, connect a 10 MHz reference clock generated by the test equipment to CMP board J13 connector. Program the on-board PLL via the GUI ADF4002 page. When the board is synchronized the LED (LD6) will be lit.

4.4.2 CMP Matching Networks

The matching networks that are fitted to CMP at manufacture are listed in Table 2.

Table 2: Default bands matched to CMP

Connector	Schematic name	Matching network
J5	CHAI	1 – 3 GHz broadband, using TC1-1-13M+ balun
J6	CHBI	1 – 3 GHz broadband, using TC1-1-13M+ balun
J7	CHCI	10 GHz band, using NCR2-113+ balun
J4	CHDI	7 GHz band, using NCR2-113+ balun
J8	CHAO	10 GHz band, using NCR2-113+ balun
J9	CHBO	7 GHz band, using NCR2-113+ balun
J10	CHCO	1 – 3 GHz broadband, using TC1-1-13M+ balun
J11	CHDO	1 – 3 GHz broadband, using TC1-1-13M+ balun

5

Detailed Guide to LMS8001 Control Software

5.1 Control LMS8001 – Software Description

This section describes the LMS8001 control software GUI and each of the menus, buttons and embedded controls. Most of the pages in the tool corresponds to the top-level sections of the SPI programming map, which is detailed in the "LMS8001 Reference Manual" document.

The "LMS8001 Control" GUI is composed of three main areas: GUI control panel, LMS8001 register and CMP board configuration panel, and LOG panel, as illustrated in Figure 5.1.

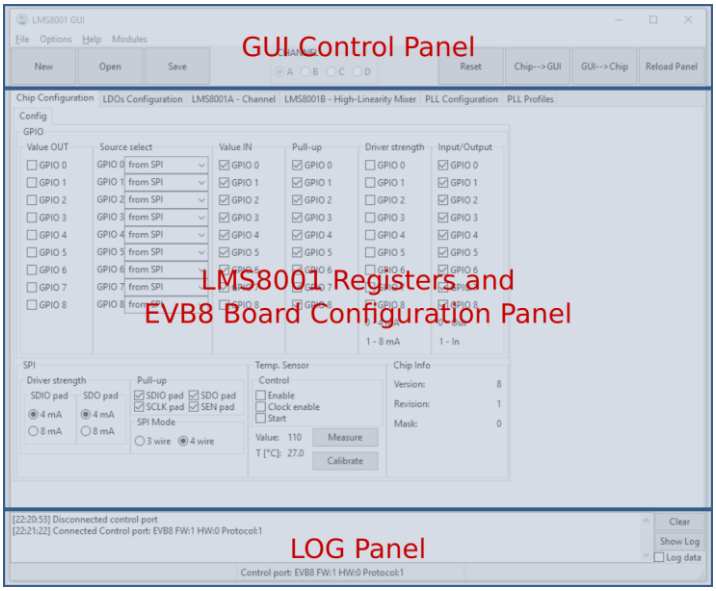


Figure 5.1: Main window organization

5.2 GUI Control Panel

This main window area consists of the main menu bar, control buttons and the channel radio group. Each of these elements will be detailed in the following.

5.2.1 The Menu Bar

The File Menu

The only item under menu is *Quit* which closes the application.

The Options Menu

Under this menu there is the item *Connection Settings* which enables connection of the software with the CMP using appropriate COM port.

The Help Menu

The item *About* in this menu provides the basic software information such as version and build date.

The Modules Menu

This menu contains the following items:

ADF4002

The ADF4002 is a PLL that locks the on-board TXCO (usually 40.00MHz) to the external reference (usually 10 MHz connected to J13). On-board reference is supplied to the LMS8001 synthesizers. This is normally used to synchronize the CMP with the measurement equipment.

Figure 5.2 presents the ADF4002 control window. All control registers of the ADF4002 are accessible from this window.

However, in vast majority of situations the following procedure is sufficient for successful synchronization:

- In the field *Fref [MHz]* input the frequency of the external reference in MHz
- In the field *Fvco [MHz]* input the frequency of the CMP on-board reference in MHz
- Press the *Calculate R, N & Upload* button

Diode LD6 indicates successful synchronization.

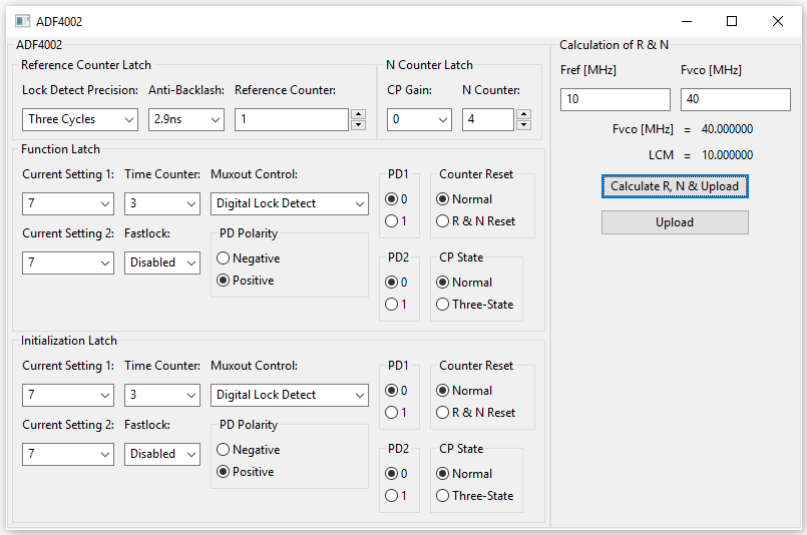


Figure 5.2: ADF4002 control window

Si5351C

This circuit is present on the CMP board.

Device Info

This menu item provides basic information on the LMS8001 device and the CMP firmware.

SPI

This module enables the user to directly provide SPI commands. In the *LMS Write* section, write SPI commands can be defined by entering the address and the value and sent to LMS8001 by pressing the *CMD Write* button.

Similarly, read SPI commands can be defined in the *LMS Read* section by providing the address. Such an SPI command can be sent to LMS8001 by pressing the *CMD Read* button.

In the *SPI Commands* section multiple SPI commands can be defined in the edit area. By pressing *SPI Step* button, these commands will be executed consecutively, line by line. Commands should start with RD or WR, which defines whether the read or write command is desired. This should be followed by either the register or the register field name as defined in “LMS8001 Reference Manual”, or with the desired address. Finally, for the write command, the desired register value should be provided.

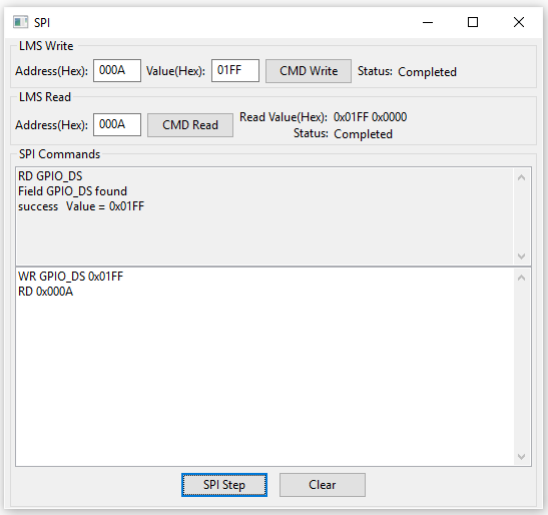


Figure 5.3: SPI window

GPIO

GPIO signals from the LMS8001 are connected to the dedicated microcontroller pins via bi-directional level shifters U6 and U7 (SN74AVC4T774). Such a connection was required since the microcontroller operates on 3.3V, while the LMS8001 IO operates by default on 1.8V.

GPIO item provides the control of the GPIO signal level shifters and microcontroller GPIO signals.

LMS8001 GPIO parameters and values are not controlled from this module, but are set from the main window (please refer to *Chip Configuration* tab).

Figure 5.4 illustrates on possible setup of the GPIO signals.

After each change it is necessary to click *Update* button to update the window values.

In the section *GPIO 0 – 3*, radio group *Direction LMS8* defines the direction of the U7 level shifter. Value *OUT* sets the direction such that the signals are fed from the LMS8001 to the MCU. In this case the *GPIO* check boxes take the values read from the dedicated MCU pins. Value *IN* presumes that the LMS8001 GPIO pins are configured as input, hence the level shifter direction is from the MCU to LMS8001. In this case *GPIO* check boxes can be edited, and define the output values of the MCU dedicated pins. Similar controls are available for the GPIO pins 4 – 7, which are connected between the LMS8001 and MCU via level shifter U6.

GPIO 8 direction is always from the MCU to LMS8001, and cannot be changed.

Check box *CORE_LDO_EN* is obsolete.

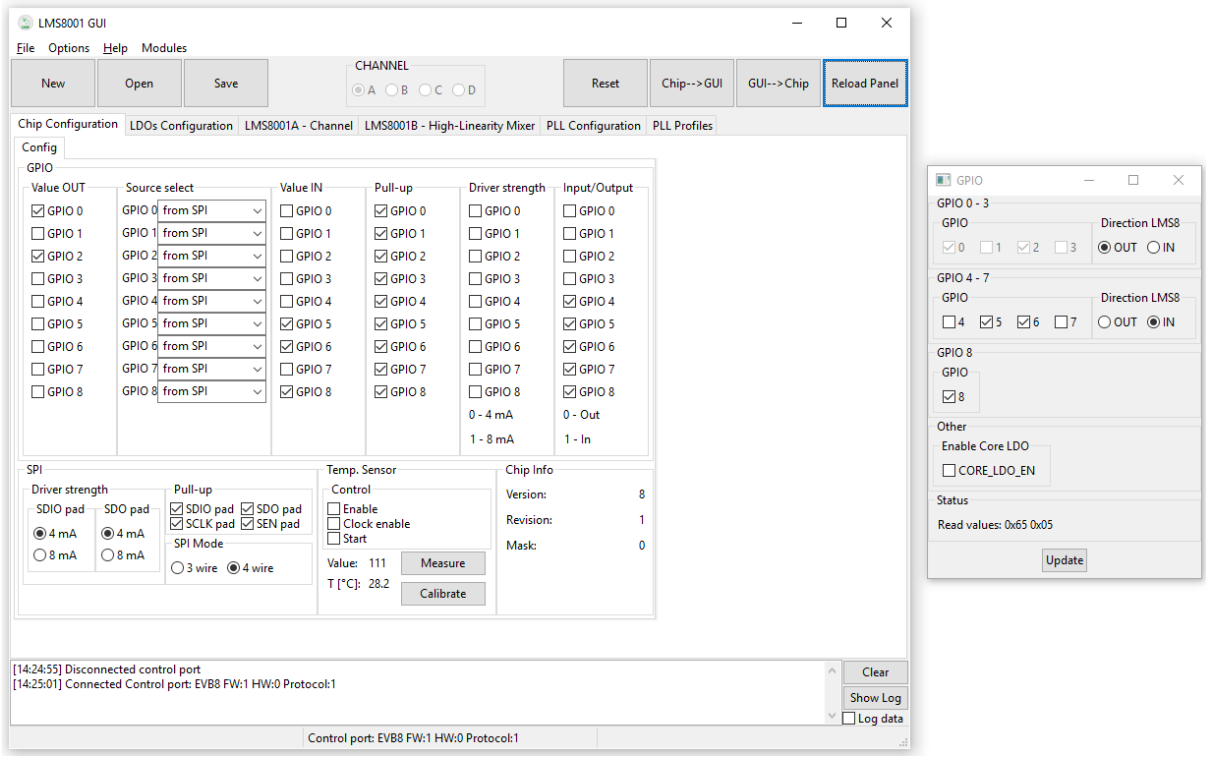


Figure 5.4: GPIO control window

Measurements

This module is used internally at Lime Microsystems with the purpose of measurement automation. It provides the means of automating the measurements that include EVB7, CMP and Agilent E4440A spectrum analyzer.

5.2.2 Control Buttons

The function of the control buttons are detailed in the following.

New

This button resets the LMS8001, read registers from the LMS8001 to the GUI and refreshes the displayed values.

Open

The complete LMS8001 register values, as well as some additional information such as the CMP reference clock frequency can be saved in a single .ini project file. This is a simple human readable initialization file.

This button loads such .ini project file.

Save

This button saves the .ini project file.

Reset

This button resets the LMS8001.

Chip →GUI

This button forces the complete LMS8001 register bank to be read from the LMS8001 chip, and the values of the GUI to be updated accordingly.

This control software communicates with the LMS8001 on each change. Hence, the values of the registers in the LMS8001 chip and in the GUI should always be the same. Thus, normally there is no need for using this button.

GUI → Chip

This button forces the complete register bank values from GUI to be written to the LMS8001 chip.

As mentioned previously, normally there is no need to use this function, since the LMS8001 register values should always be up-to-date with the GUI values.

Reload Panel

This button forces the refresh of the GUI window.

5.2.3 Channel Radio Group

Some of the main window tabs refer to the LMS8001 registers that are specific to a certain channel. Such tabs are *LMS8001A – Channel*, and *LMS8001B – High-Linearity Mixer*. If one of these tab is active, radio group *Channel* is enabled and defines the channel that is displayed.

5.3 LMS8001 Registers and CMP Configuration Panel

This main window area enables the control of the LMS8001 registers, as well as some CMP parameters.

By positioning the mouse over a certain field, the hint message is shown, as exampled on Figure 5.5. Hint message provides the field description, as well as the address of the field in the register map. Detailed information on register fields is provided in the "LMS8001 Reference Manual" document.

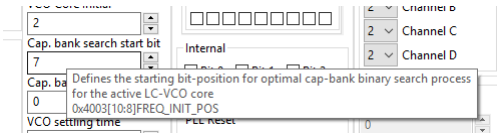


Figure 5.5: Field hint example

5.3.1 Chip Configuration

This tab enables the control of the *ChipConfig* register bank of the LMS8001. Detailed description of this register bank is available in “LMS8001 Reference Manual” document.

Additional functionality is provided for the temperature sensor. By clicking the *Measure* button the procedure of temperature measurement is performed and the return value is displayed. This value is also recalculated in Celsius degrees and displayed. Details are available in the Section 4 of the “LMS8001 Reference Manual” document.

Single point calibration of the temperature sensor is enabled by the button *Calibrate*. By entering the corresponding value/temperature pair the constant coefficient T0 is recalculated and used in subsequent temperature measurements. If .ini project file is saved afterwards, this coefficient value is saved within this file for subsequent use.

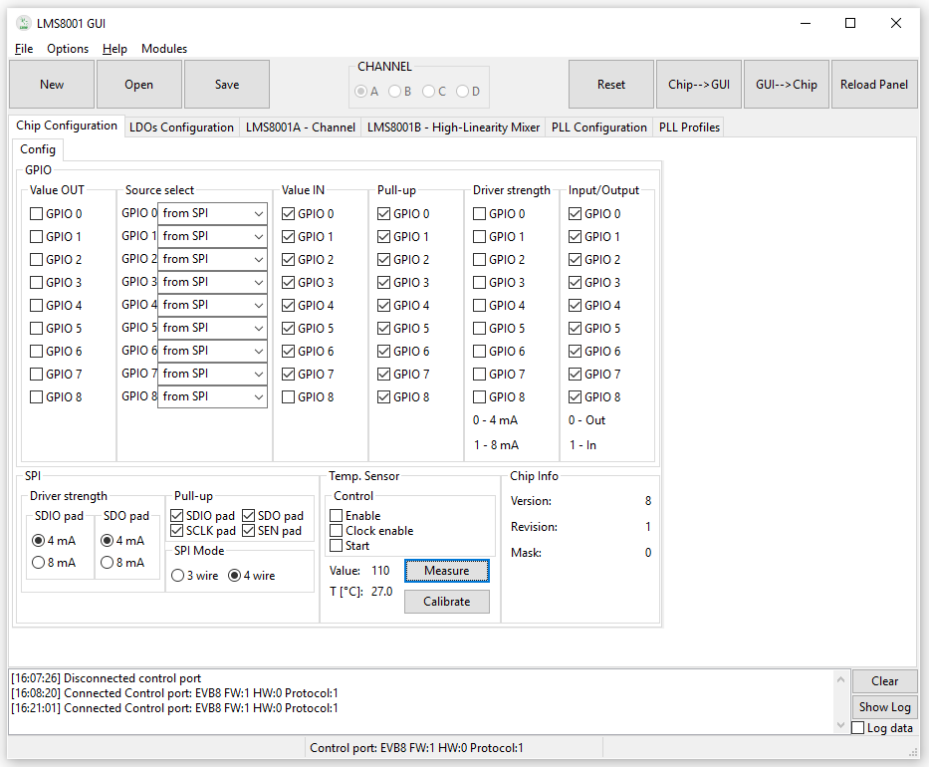


Figure 5.6: Chip Configuration tab

5.3.2 LDOs Configuration

This tab provides the control of the LMS8001 Bias and LDOs. These controls are grouped in the *BiasLDOConfig* register bank of the LMS8001. Detailed description of this register bank is available in “LMS8001 Reference Manual” document.

Since the CMP is configured such that all the LMS8001 supplies are provided from the outside, it is suggested that the default settings are used, in which the internal LDOs are disabled.

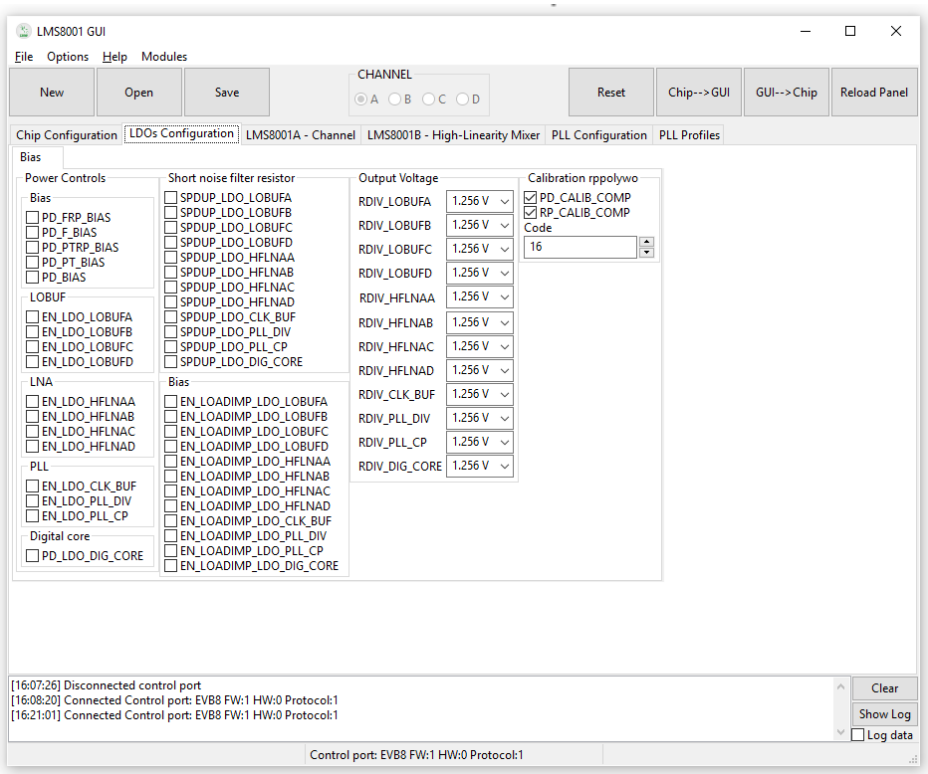


Figure 5.7: LDOs Configuration tab

5.3.3 LMS8001A – Channel

This tab is relevant only for the LMS8001A chip version.

It provides the control of the *Channel_x* register banks, where x = A, B, C, D. Detailed description of this register banks is available in "LMS8001 Reference Manual" document.

The channel that is currently displayed is defined by the *CHANNEL* radio group value.

It is possible to store up to 4 different sets of settings, which are denoted *Program X* where X = 0, 1, 2, 3. Each of the programs stores settings for power-down controls, LNA, and PA parameters. Which of the programs will be active is defined in the *MUX Control* section.

As an illustration, *MUX Control* settings in Figure 5.8 allow that LNA settings defined in *Program 0* and *Program 1* are toggled by toggling the GPIO0 signal. This allows "real-time" switching between the predefined parameter sets which may be a powerful feature in certain applications.

Detailed information on programs' multiplexer control settings is provided in "LMS8001 Reference Manual".

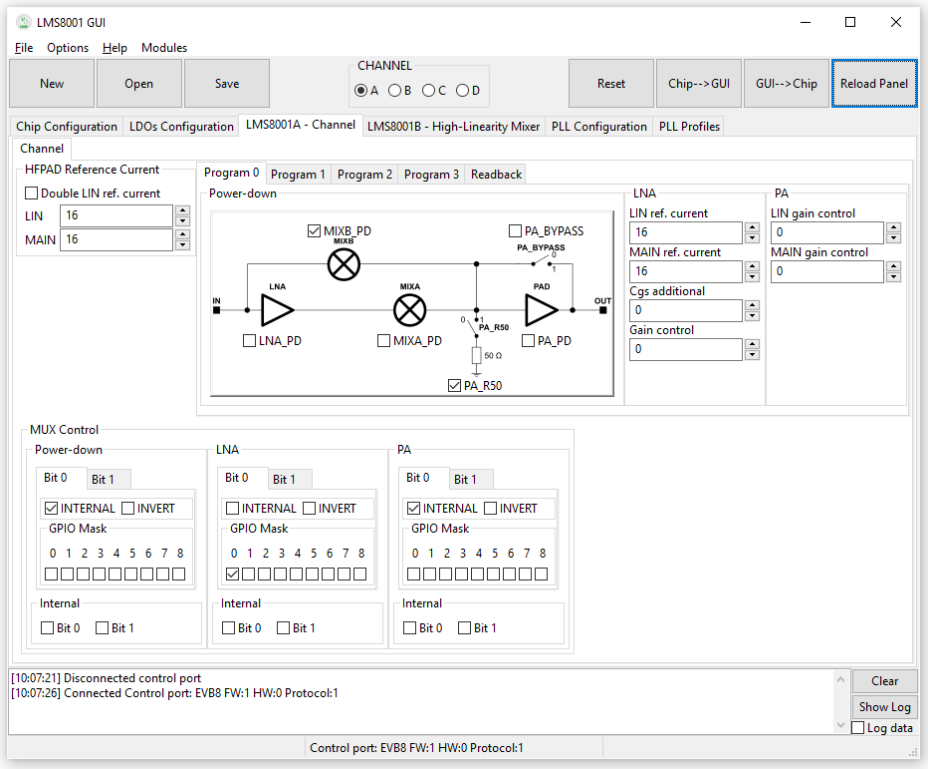


Figure 5.8: LMS8001A – Channel tab

5.3.4 LMS8001B – High-Linearity Mixer

This tab is relevant only for the LMS8001B chip version. Since the CMP board uses LMS8001A chip, it is of no interest here.

5.3.5 PLL Configuration

This tab provides the control of the PLL configuration. These controls are grouped in the *PLL_CONFIGURATION* register bank of the LMS8001. Detailed description of this register bank is available in “LMS8001 Reference Manual” document.

This tab is shown in Figure 5.9.

LMS8001 is capable of storing up to 8 PLL profiles, which are the sets of PLL settings. More about PLL profile configuration will be given in the next section.

It is possible to switch between the stored PLL profiles. *PLL Profile Select* section in the *PLL Configuration* tab defines how the active PLL profile is selected. Basically, all PLL profiles are connected to a multiplexer that provides the active profile at the output. This multiplexer can be either controlled by the internal register values, of by the GPIO signals, or by their combination. Control by the GPIO signals provides powerful feature of switching between the PLL profiles in “real-time”. Details on the multiplexer control are provided in “LMS8001 Reference Manual” document.

Status section provides the information whether the VCO voltage is high or low, and the PLL lock status. In order to refresh this values it is needed to click on the *Reload Panel* button.

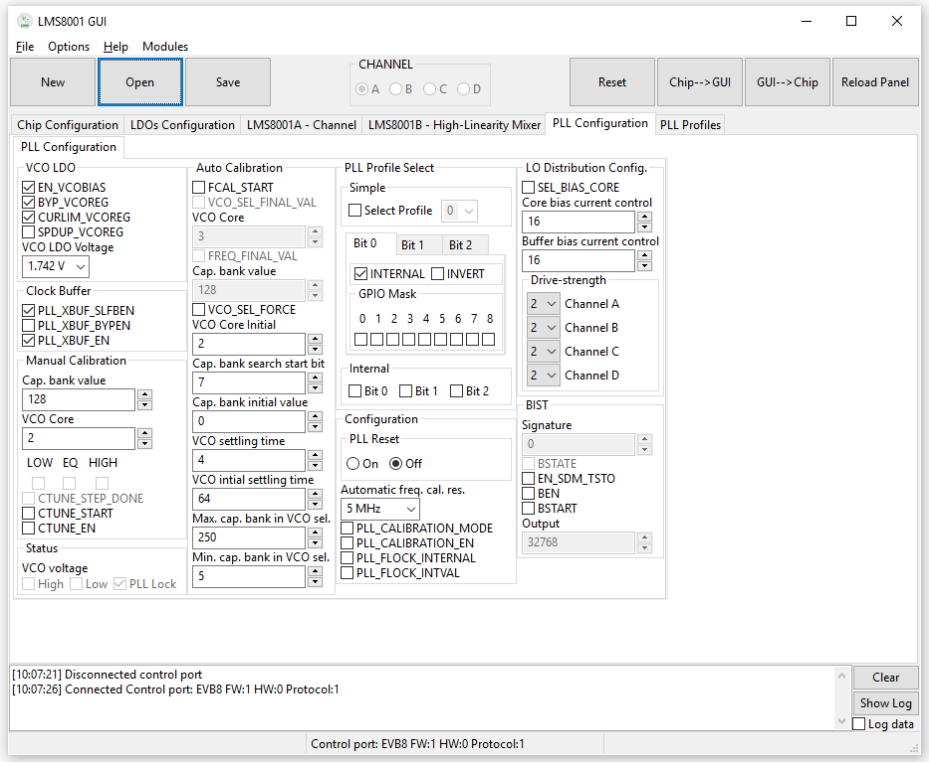


Figure 5.9: PLL Configuration tab

5.3.6 PLL Profiles

This tab provides the control of the PLL profiles. These controls are grouped in the *PLL_PROFILE_x* register banks, where $x = 0, 1, \ldots, 7$ of the LMS8001. Detailed description of this register bank is available in "LMS8001 Reference Manual" document.

This tab is shown in Figure 5.10.

The PLL profile that is currently shown is selected by the *Profile* radio group value.

It is important to understand that the profile that is currently shown is not necessarily the profile that is currently active. Defining currently active profile is described in the previous section.

GUI provides the means for easy tuning of the PLL to a desired value. These controls are grouped in sections *Tuning* and *Smart Tune*.

By entering the desired VCO oscillation frequency in the *VCO Freq. [GHz]* field and clicking the *Simple Tune* button, the currently shown profile is set as active, which is necessary for the tuning procedure, and the automatic tuning procedure of the LMS8001 is invoked. After this procedure is finished, profile which was originally set as active is activated again.

Distinction should be made between the VCO frequency, and the LO frequency. LO signal is the final output of the PLL. VCO frequency can be multiple of the LO frequency, which is defined by the feed-forward divider value (FF_MOD).

LO frequency can be directly set by entering the desired value in the *LO Freq. [GHz]* and clicking the *Smart Tune* button. This button calls a method which performs full algorithm

which includes optimizing PLL-Core settings and defining loop dynamics in Fast-Lock mode. Additionally it is possible to define if the quadrature is desired by checking the *Generate IQ* checkbox. Also, the desired PLL loop bandwidth and phase margin can be defined. According to these values appropriate loop filter elements will be automatically calculated. Finally, the *BWEF* field defines the so called *Bandwidth Extension Factor*, which, in short, determines how many times the loop filter bandwidth will be extended during the fast lock period (wider loop provides faster lock). Based on BWEF value, fast lock parameters are automatically calculated.

Details on the *Smart Tune* algorithm are given in "LMS8001 PLL-Sim Handbook" document, section 5.4 *FastSetup of PLL Profile*.

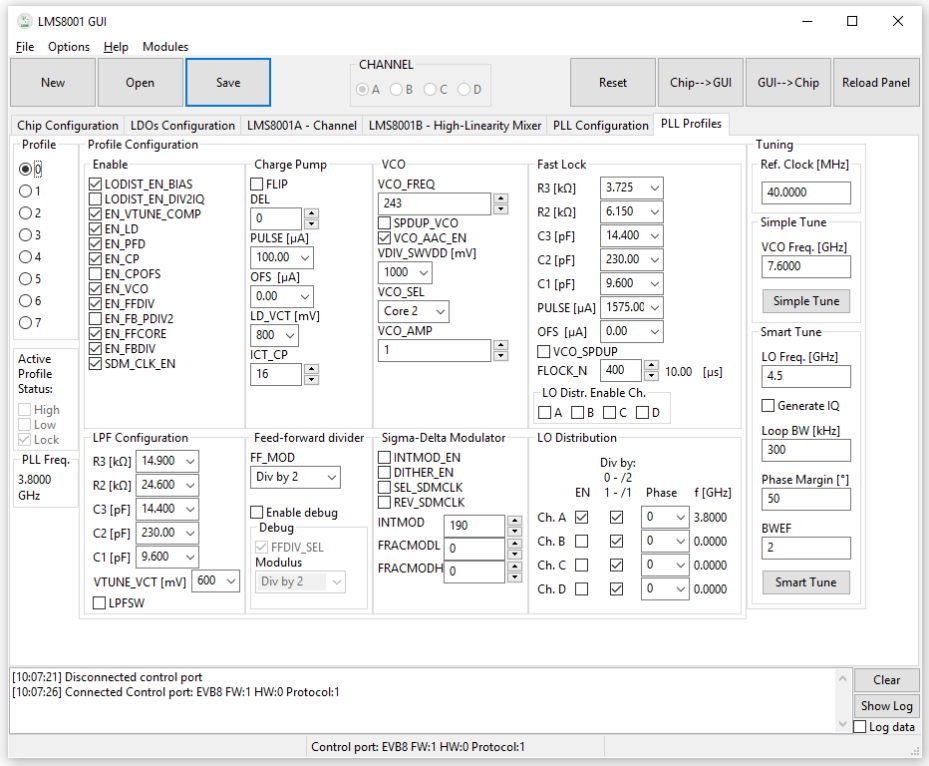


Figure 5.10: PLL Profiles tab

5.4 LOG Panel

Log panel section logs all activity executed with the GUI.

The *Clear* button deletes previously registered activity.

When *Show Log* button is pressed, the *Message log* sub-window pops-up. It contains all logged messages in the current GUI session.

If the *Log data* checkbox is checked, details of the data transfer between the PC and the CMP are logged.

The evaluation board and firmware version is displayed in the bottom status part of the log panel.

6

Basic Measurement Results

6.1 Down-Conversion

In this section the measurement results for down-conversion will be presented.

Channels were configured in the LNA-MIX-PA configuration, as illustrated in Figure 6.1.

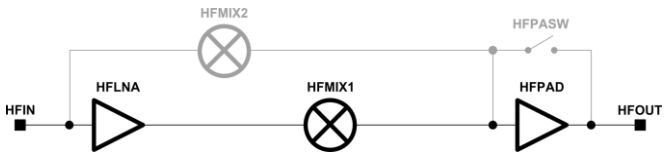


Figure 6.1: LNA-MIX-PA Configuration

Parameter values other than default that were used in the following measurements are given in Table 6.1.

Table 6.1: Parameter values (different than default) used in the following measurements

Parameter	Value
LNA_ICT_MAIN	3
LNA_ICT_LIN	31
LNA_CGSCTRL	0
LNA_GCTRL	0
PA_R50	0

6.1.1 Channel C

In the following measurements the input frequency was swept between 9 and 10 GHz, and output frequency was set to 2.1 GHz, unless otherwise stated.

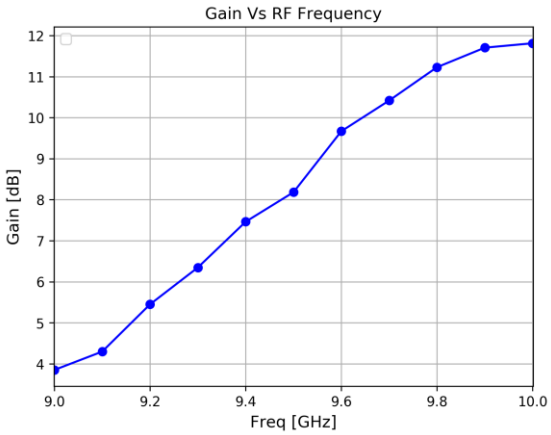


Figure 6.2: Conversion Gain

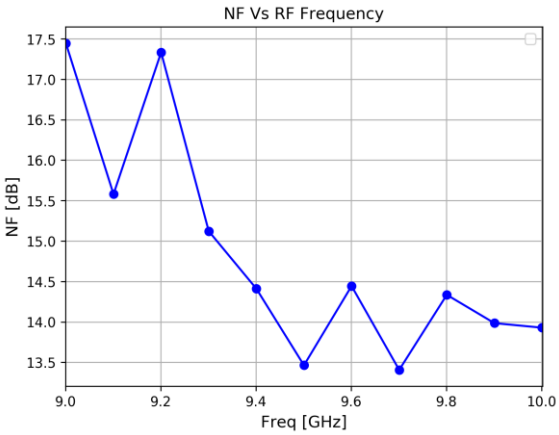


Figure 6.3: Noise Figure

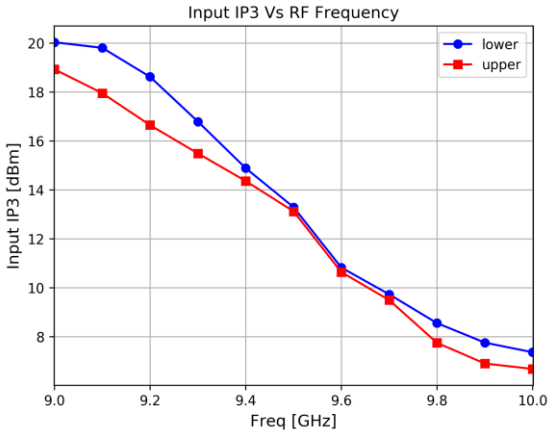


Figure 6.4: IIP3

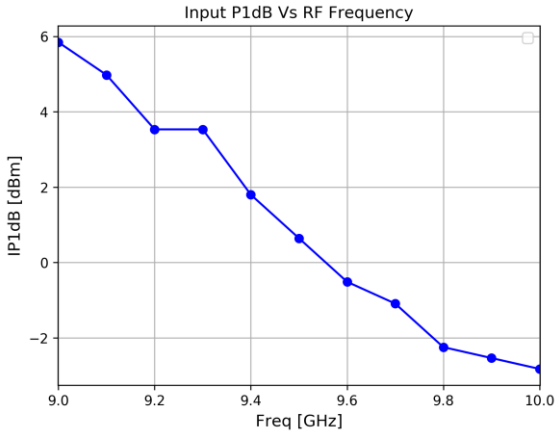


Figure 6.5: Input P1dB

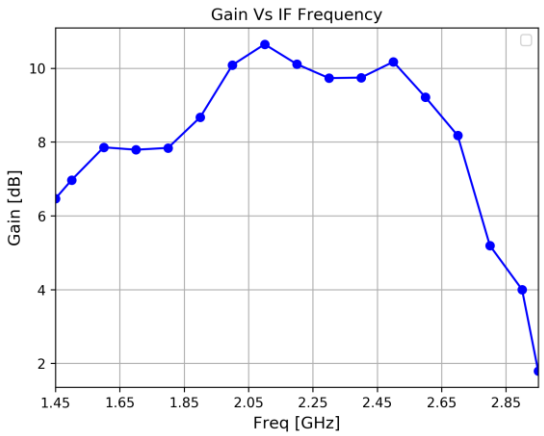
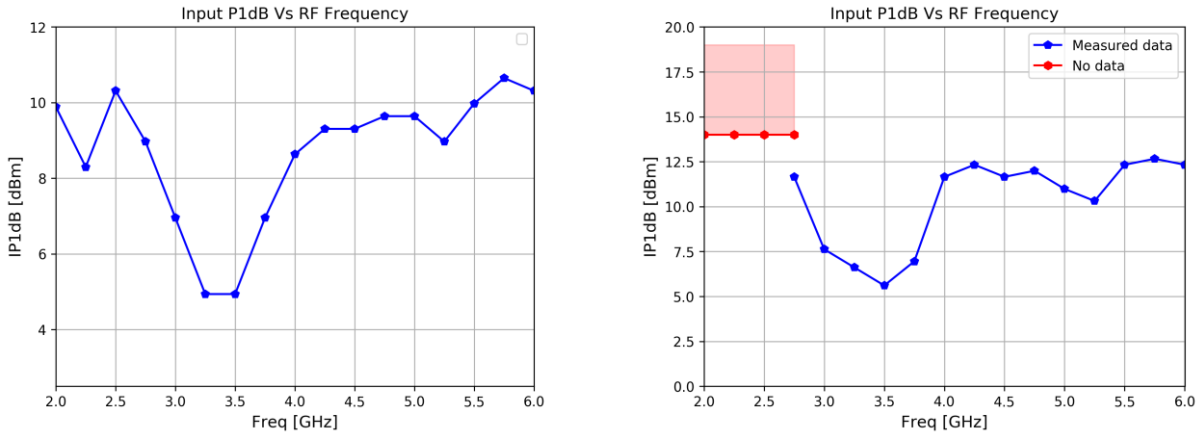


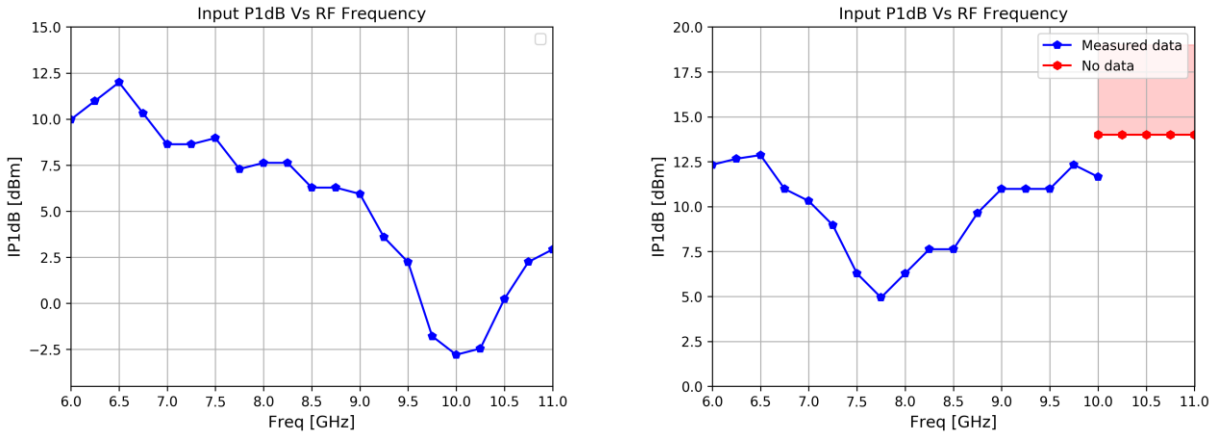
Figure 6.6: Gain vs. IF Frequency, RF Frequency = 10.0 GHz

Below are the measurement results for the 2–11 GHz frequency range, obtained using two different board configurations (LNA_ON and LNA_OFF). The following measurements were conducted with R50 set to OFF (unchecked). The lower sideband (LSB) was used for the 2–6 GHz frequency range, and the upper sideband (USB) for the 6–11 GHz range.



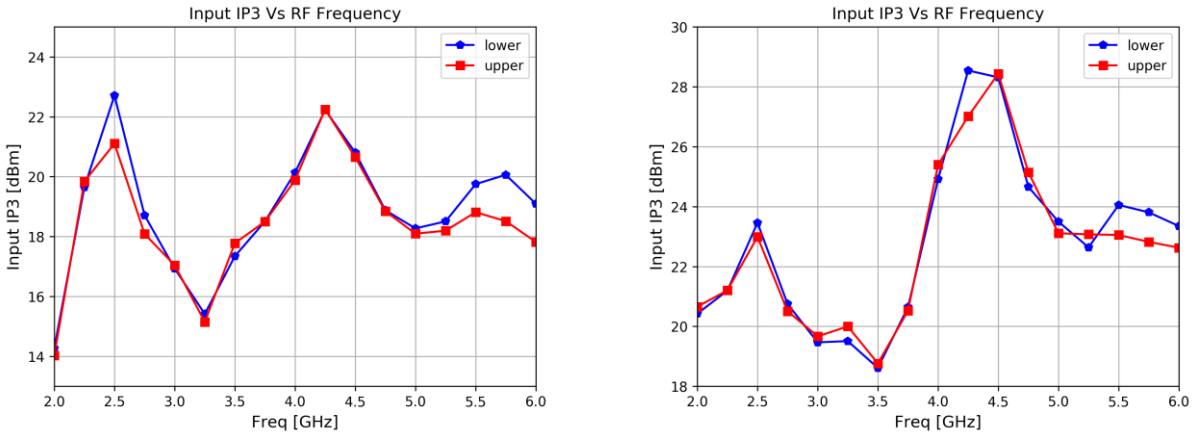
LNA_ON LNA_OFF

Figure 6.7: Input P1dB (2.0 – 6.0 GHz)



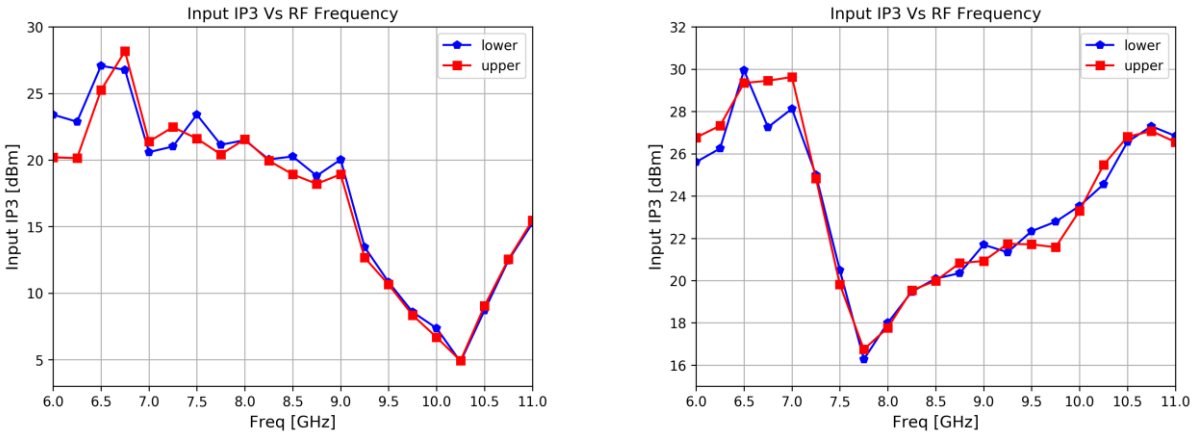
LNA_ON LNA_OFF

Figure 6.8: Input P1dB (6.0 – 11.0 GHz)



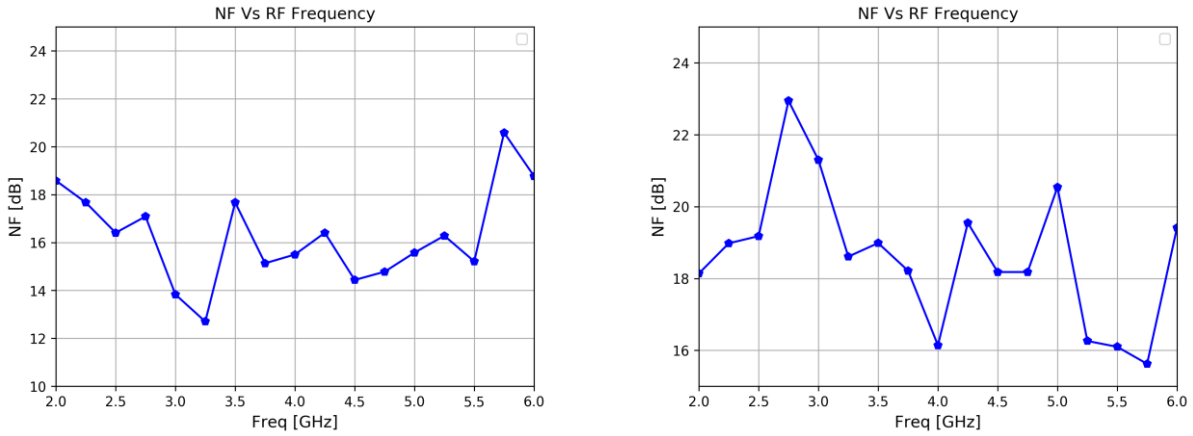
LNA_ON LNA_OFF

Figure 6.9: IIP3 (2.0 – 6.0 GHz)



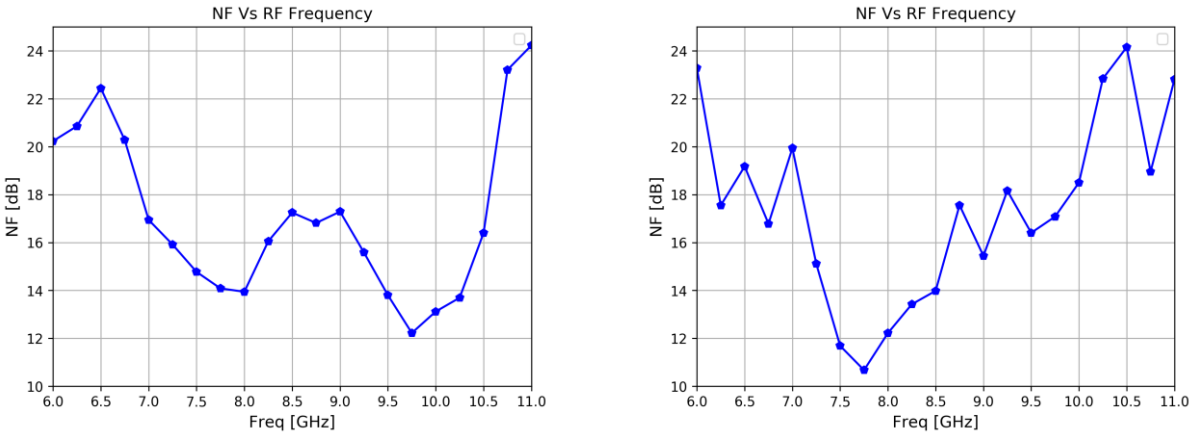
LNA_ON LNA_OFF

Figure 6.10: IIP3 (6.0 – 11.0 GHz)



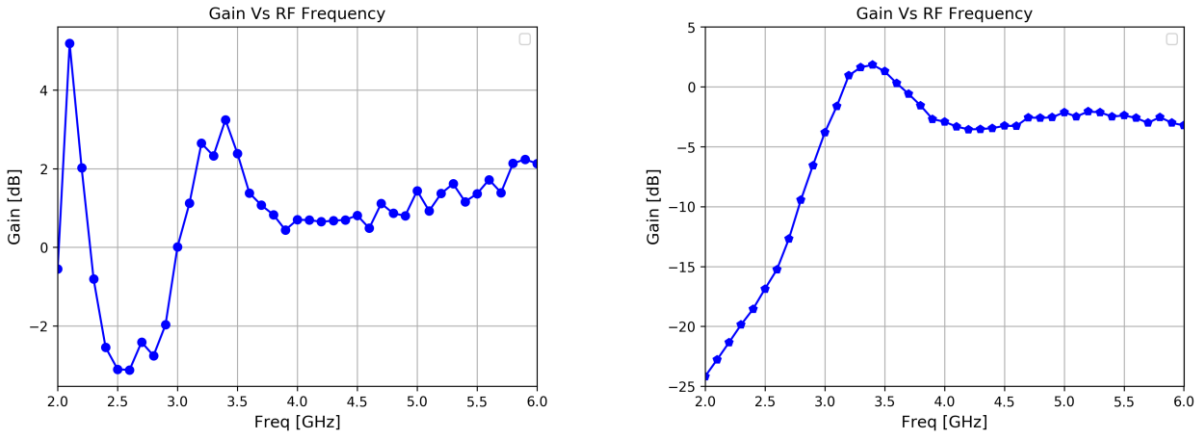
LNA_ON LNA_OFF

Figure 6.11: Noise Figure (2.0 – 6.0 GHz)



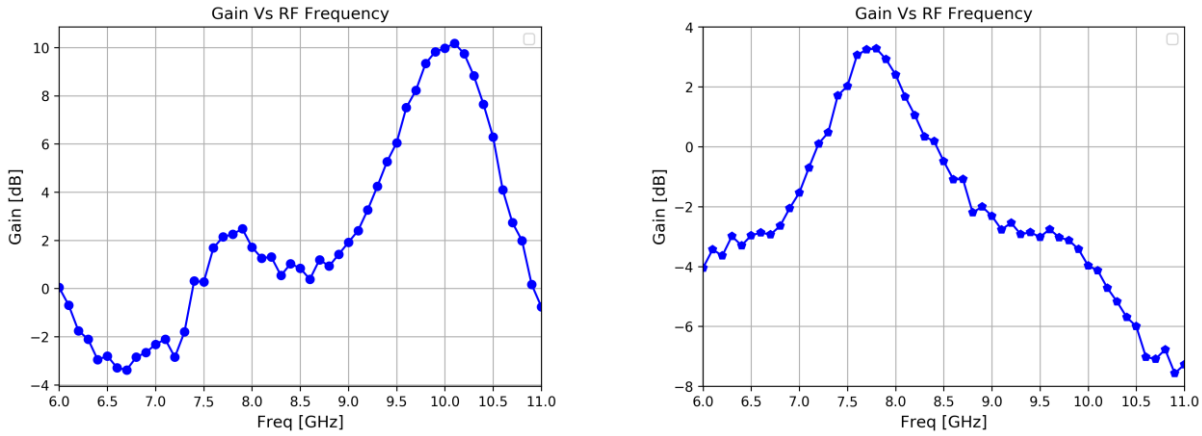
LNA_ON LNA_OFF

Figure 6.12: Noise Figure (6.0 – 11.0 GHz)



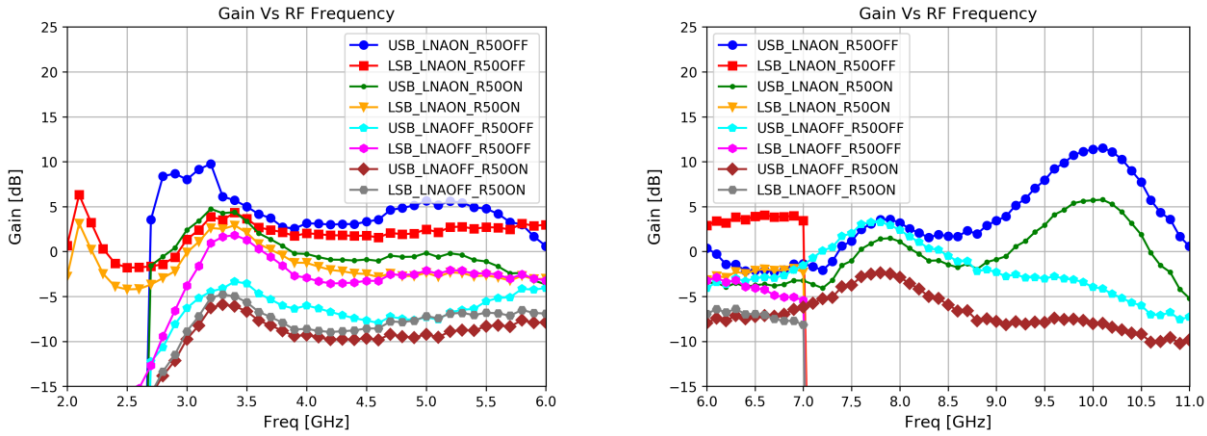
LNA_ON LNA_OFF

Figure 6.13: Gain (2.0 – 6.0 GHz)



LNA_ON LNA_OFF

Figure 6.14: Gain (6.0 – 11.0 GHz)



2.0 – 6.0 GHz 6.0 – 11.0 GHz

Figure 6.15: Conversion Gain for different board configurations

6.1.2 Channel D

In the following measurements the input frequency was swept between 6.5 and 7.5 GHz, and output frequency was set to 2.1 GHz, unless otherwise stated.

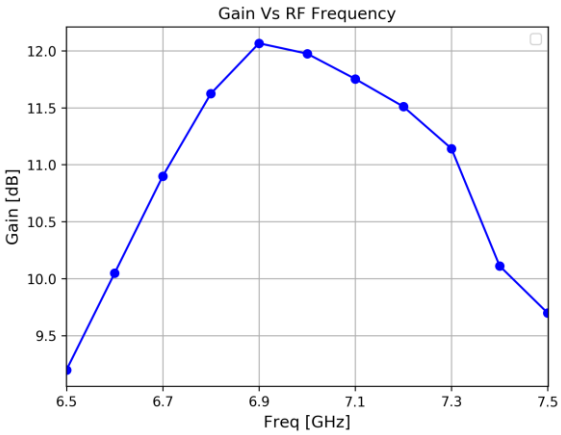


Figure 6.16: Conversion Gain

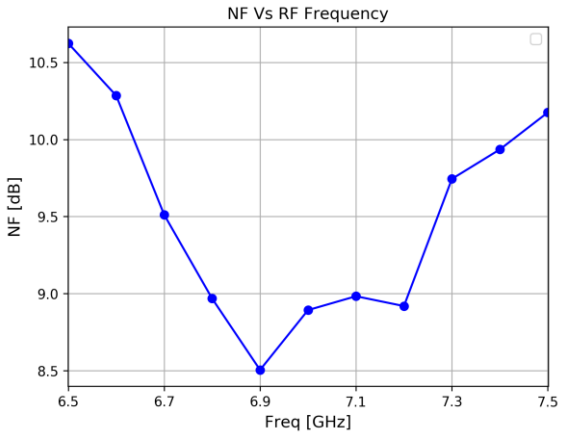


Figure 6.17: Noise Figure

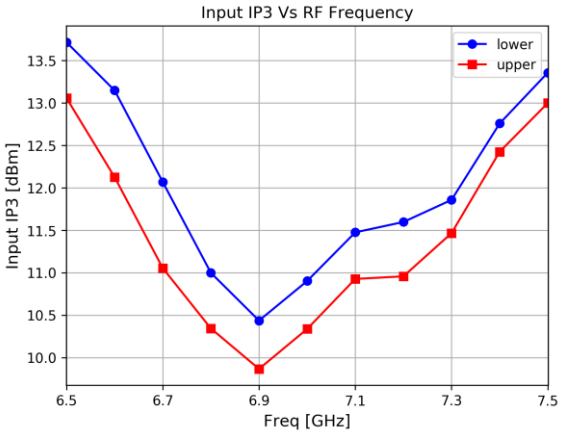


Figure 6.18: IIP3

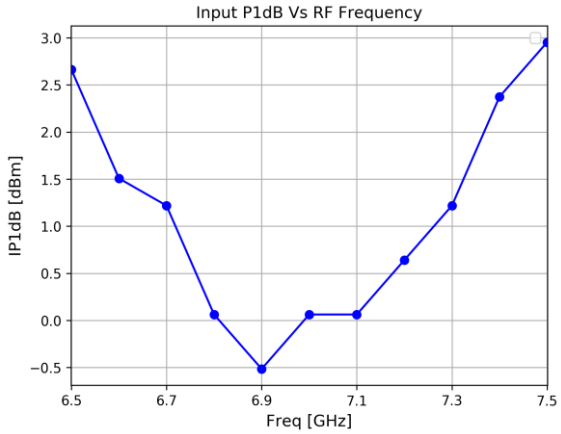


Figure 6.19: Input P1dB

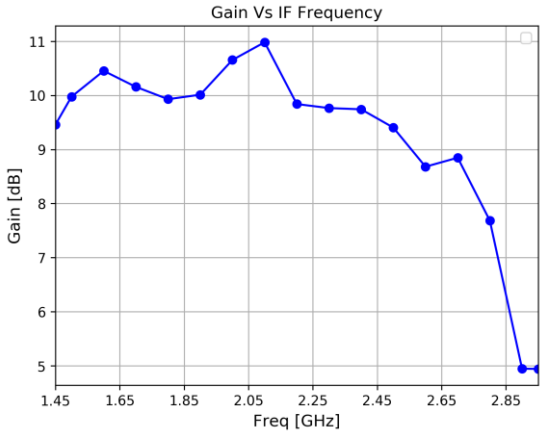
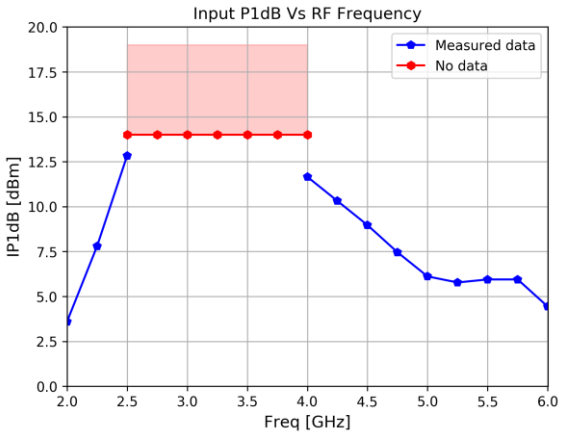
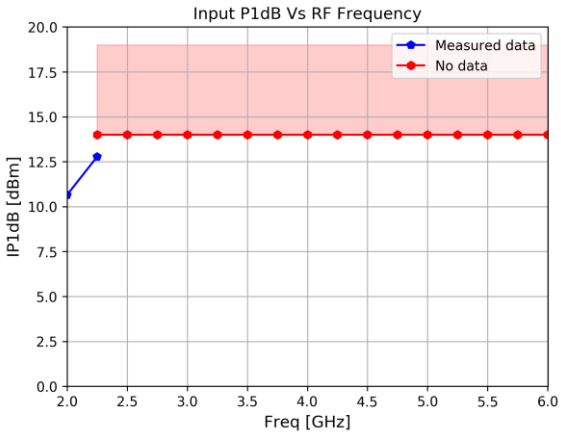


Figure 6.20: Gain vs. IF Frequency, RF Frequency = 7.0 GHz

Below are the measurement results for the 2–11 GHz frequency range, obtained using two different board configurations (LNA_ON and LNA_OFF). The following measurements were conducted with R50 set to OFF (unchecked). The lower sideband (LSB) was used for the 2–6 GHz frequency range, and the upper sideband (USB) for the 6–11 GHz range.

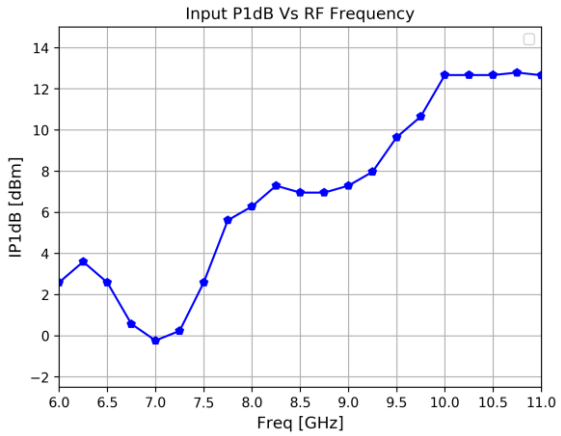


LNA_ON

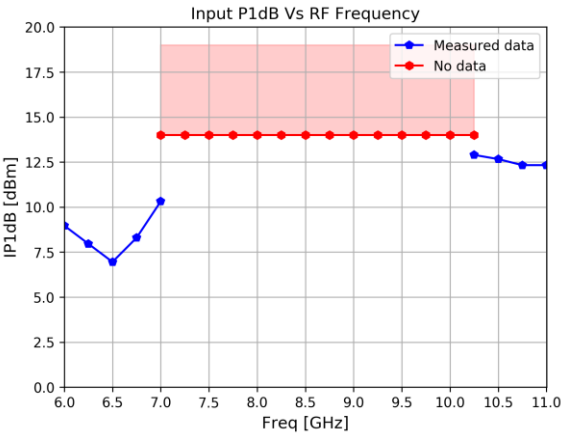


LNA_OFF

Figure 6.21: Input P1dB (2.0 – 6.0 GHz)

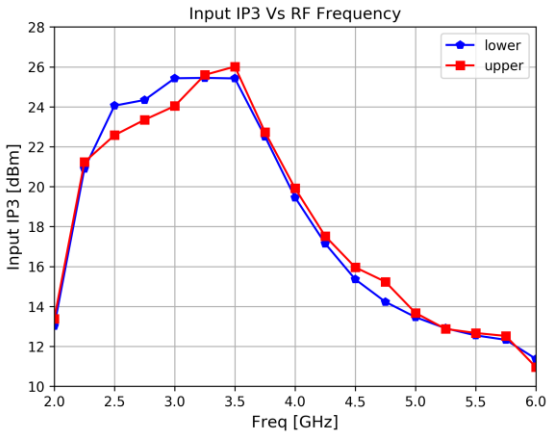


LNA_ON

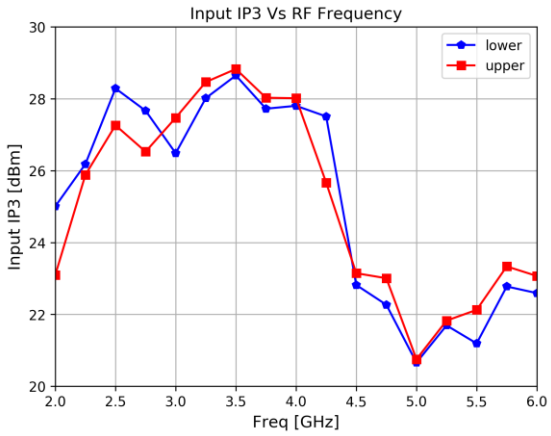


LNA_OFF

Figure 6.22: Input P1dB (6.0 – 11.0 GHz)

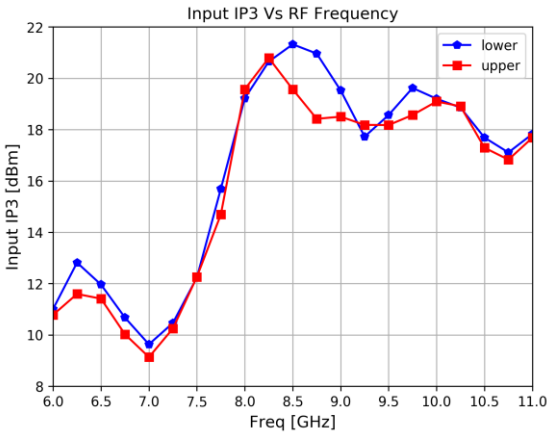


LNA_ON

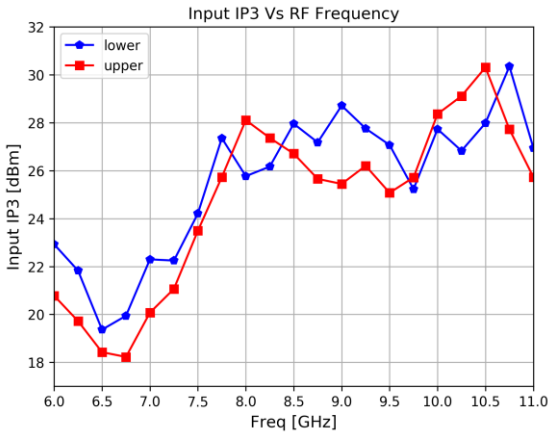


LNA_OFF

Figure 6.23: IIP3 (2.0 – 6.0 GHz)

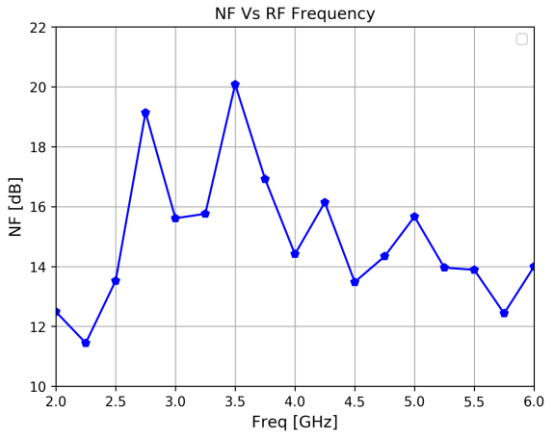


LNA_ON

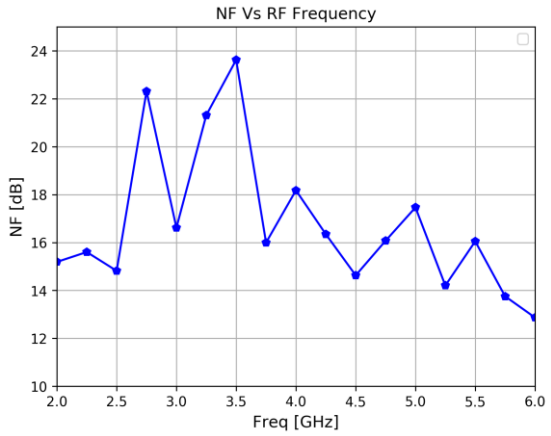


LNA_OFF

Figure 6.24: IIP3 (6.0 – 11.0 GHz)

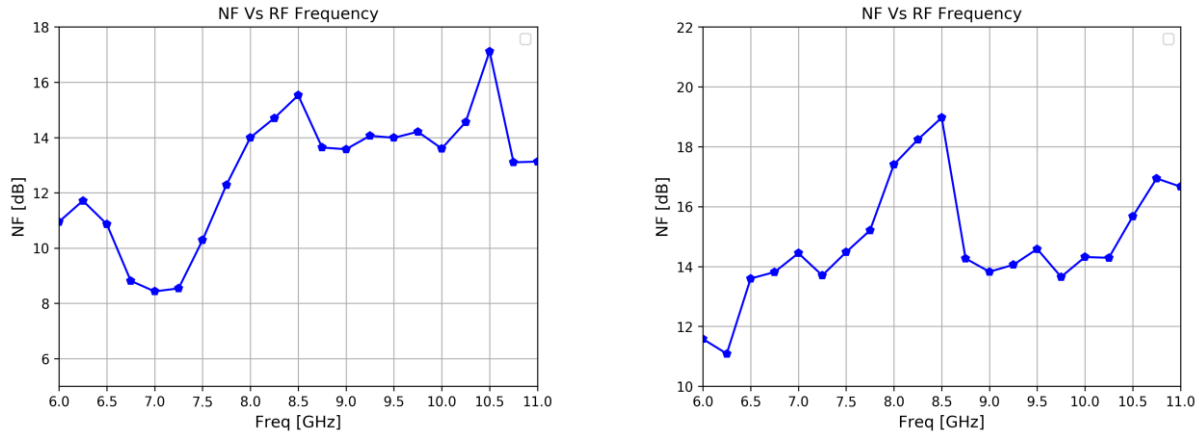


LNA_ON



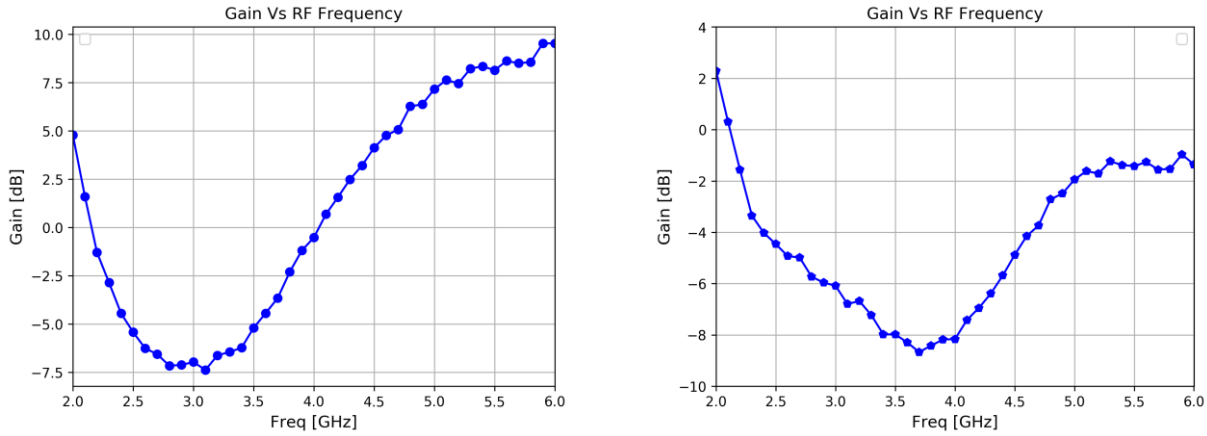
LNA_OFF

Figure 6.25: Noise Figure (2.0 – 6.0 GHz)



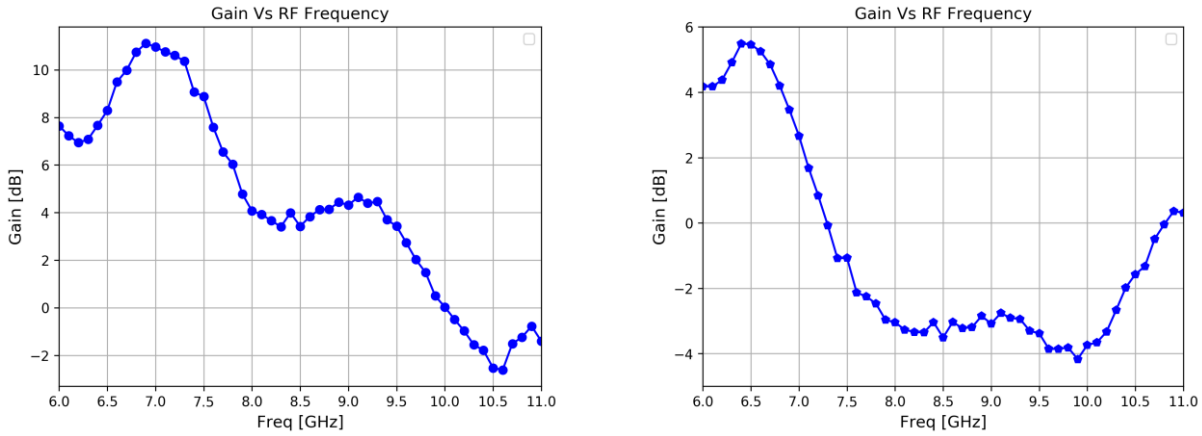
LNA_ON LNA_OFF

Figure 6.26: Noise Figure (6.0 – 11.0 GHz)



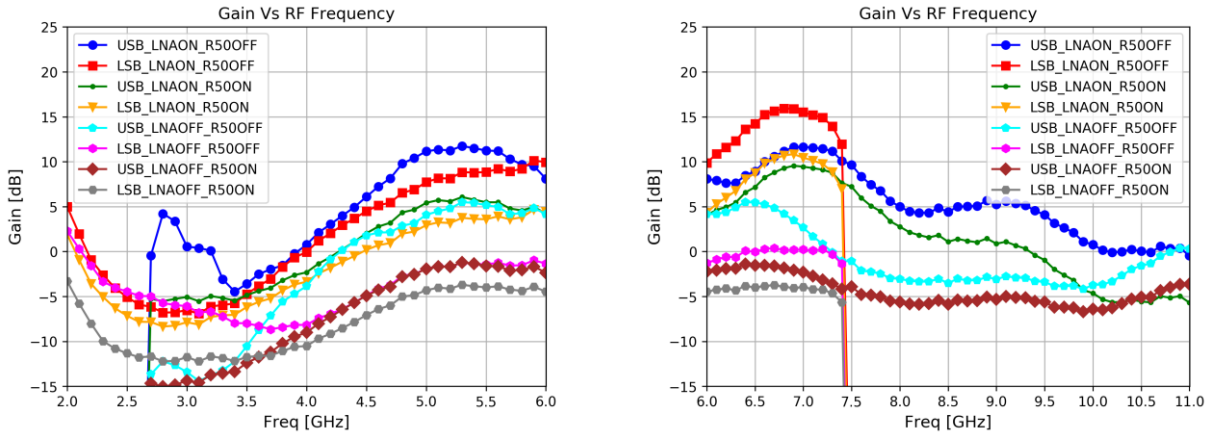
LNA_ON LNA_OFF

Figure 6.27: Gain (2.0 – 6.0 GHz)



LNA_ON LNA_OFF

Figure 6.28: Gain (6.0 – 11.0 GHz)



2.0 – 6.0 GHz

6.0 – 11.0 GHz

Figure 6.29: Conversion Gain for different board configurations

6.2 Up-Conversion

In this section the measurement results for up-conversion will be presented.

Channels were configured in the MIX-PA configuration, as illustrated in Figure 6.30: .

Figure 6.30: MIX-PA Configuration

Parameter values used in the following measurements all have default values.

6.2.1 Channel A

In the following measurements the output frequency was swept between 9 and 10 GHz, and the input frequency was kept at 2.1 GHz, unless otherwise stated.

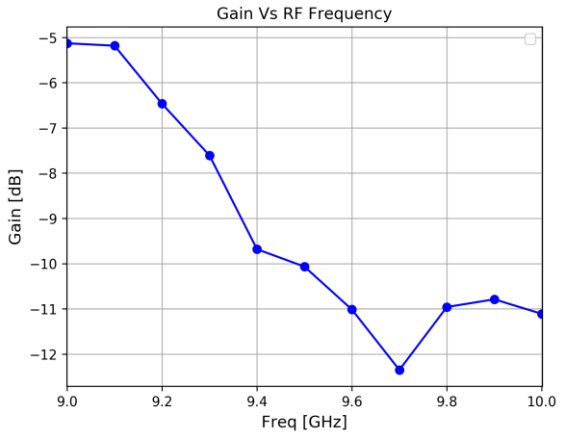


Figure 6.31: Conversion Gain

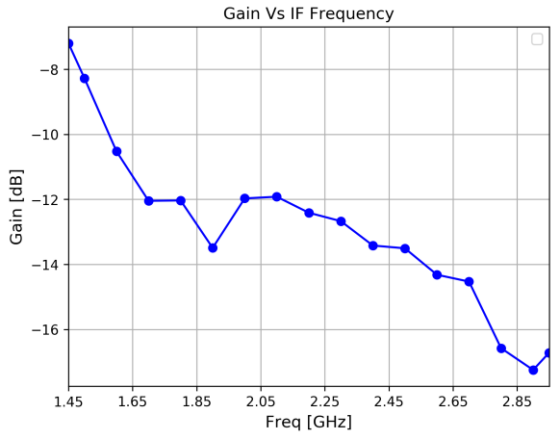


Figure 6.32: Gain vs. IF Frequency, RF Frequency = 10.0 GHz

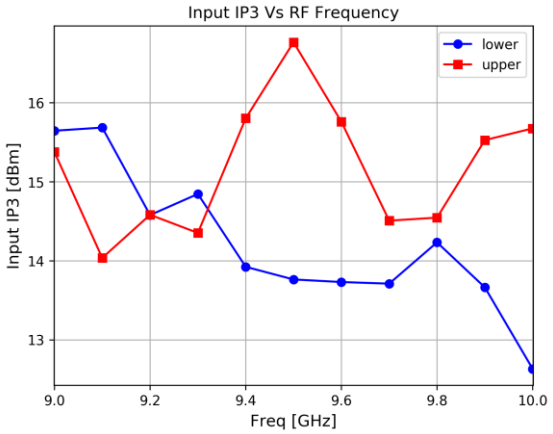


Figure 6.33: IIP3

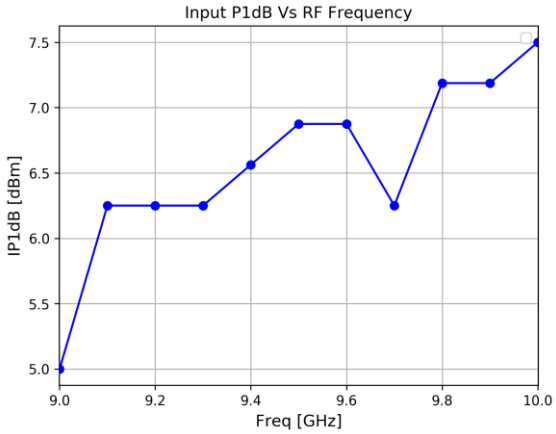
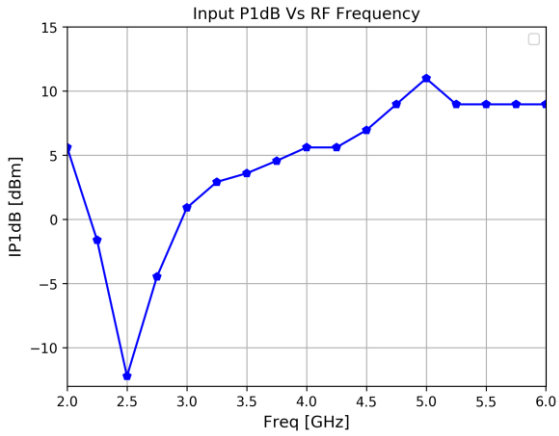
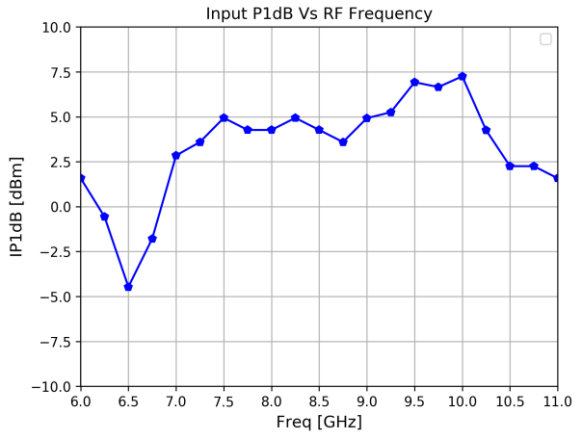


Figure 6.34: Input P1dB

Below are the measurement results for the 2–11 GHz frequency range. For Up-Conversion, the configuration with LNA_ON is always used. The following measurements were conducted with R50 set to OFF (unchecked). The lower sideband (LSB) was used for the 2–6 GHz frequency range, and the upper sideband (USB) for the 6–11 GHz range.

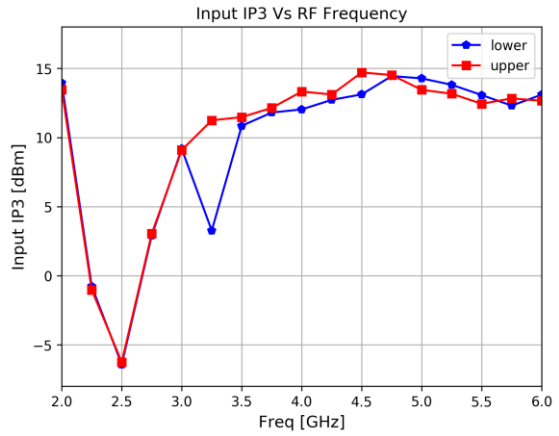


2.0 – 6.0 GHz

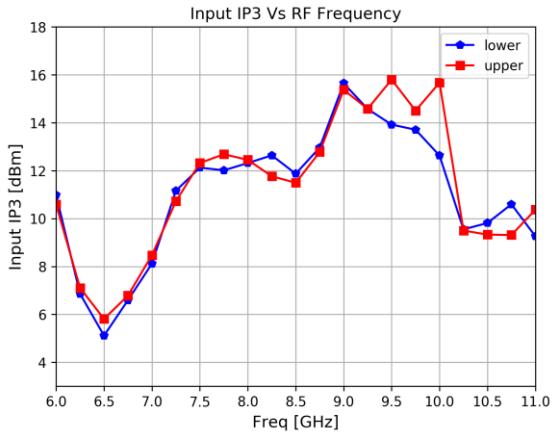


6.0 – 11.0 GHz

Figure 6.35: Input P1dB

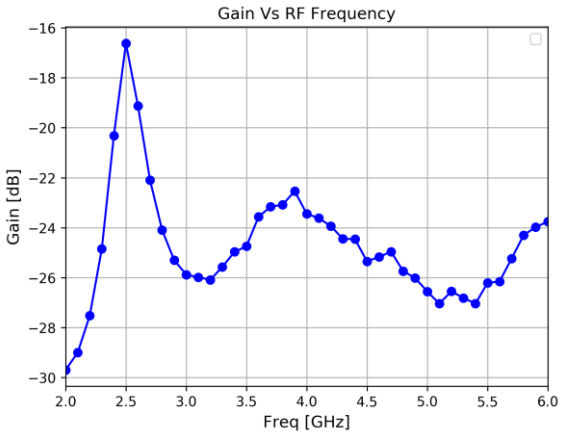


2.0 – 6.0 GHz

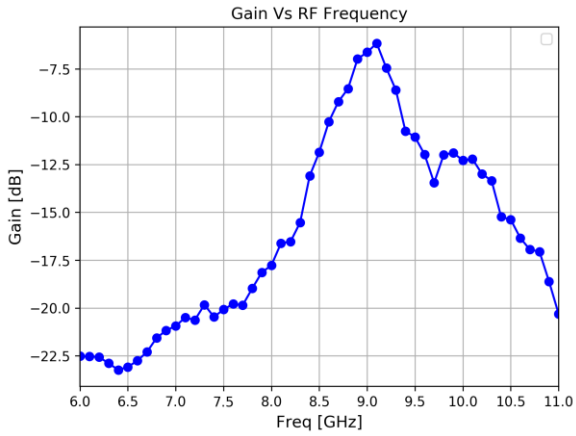


6.0 – 11.0 GHz

Figure 6.36: IIP3

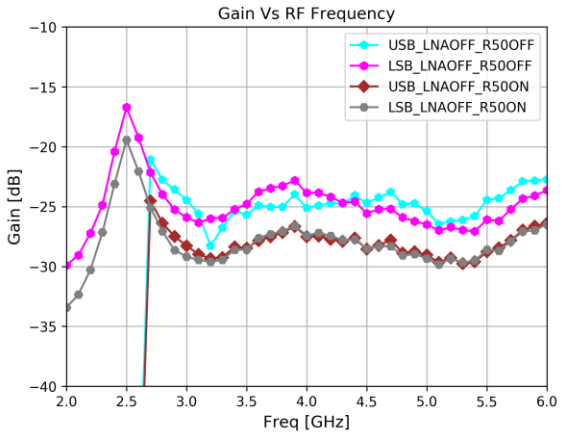


2.0 – 6.0 GHz

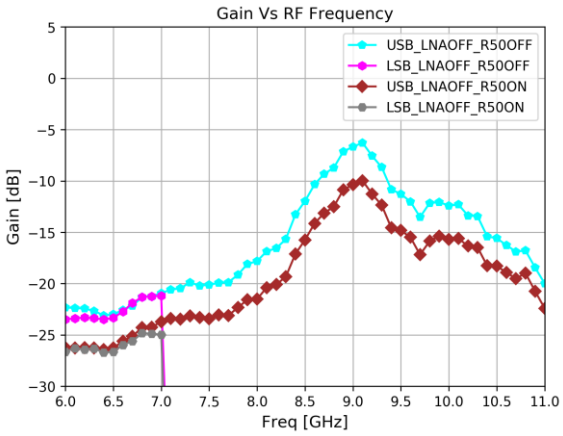


6.0 – 11.0 GHz

Figure 6.37: Gain



2.0 – 6.0 GHz



6.0 – 11.0 GHz

Figure 6.38: Conversion Gain for different board configurations

6.2.2 Channel B

In the following measurements the output frequency was swept between 6.5 and 7.5 GHz, and the input frequency was kept at 2.1 GHz, unless otherwise stated.

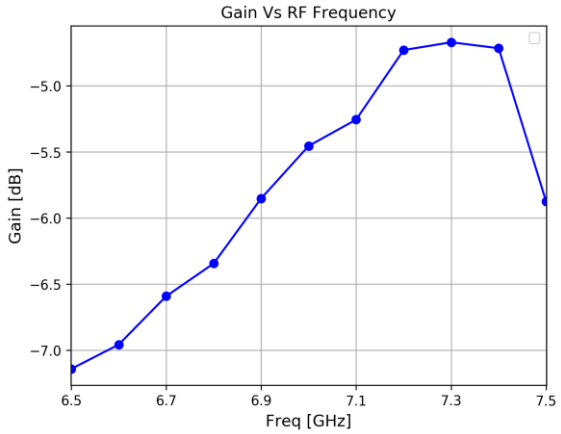


Figure 6.39: Conversion Gain

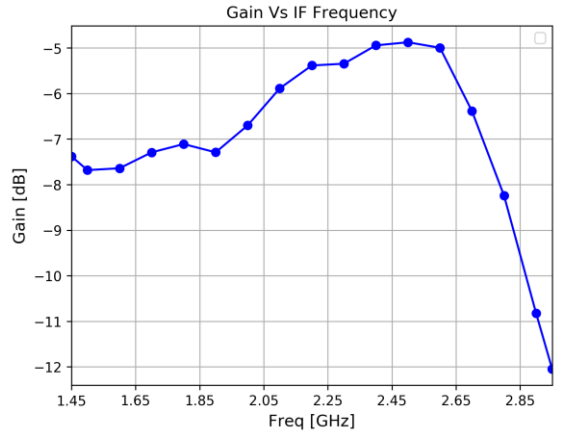


Figure 6.40: Gain vs. IF Frequency, RF Frequency = 7.0 GHz

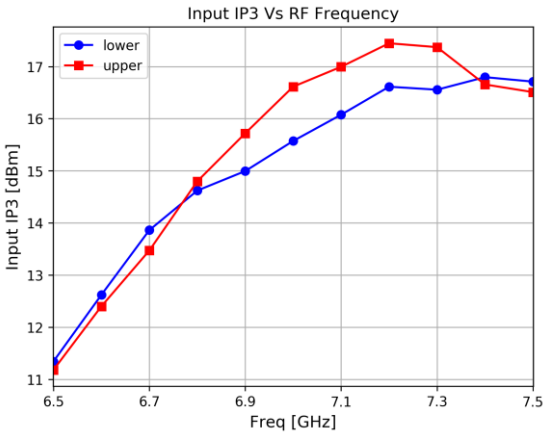


Figure 6.41: IIP3

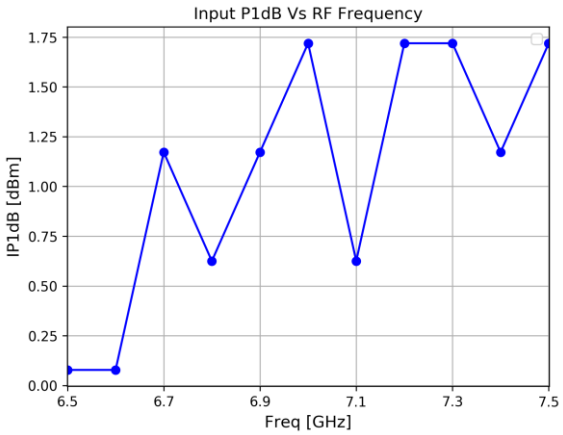
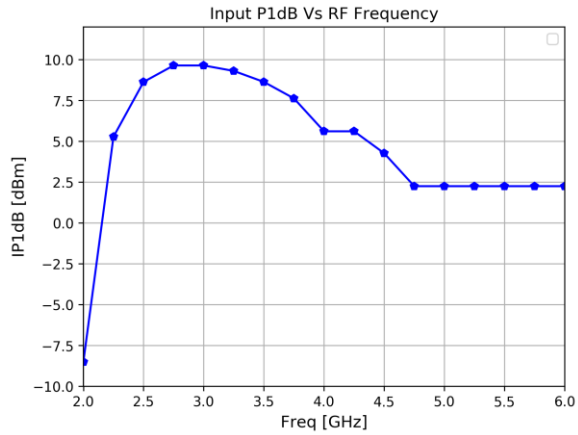
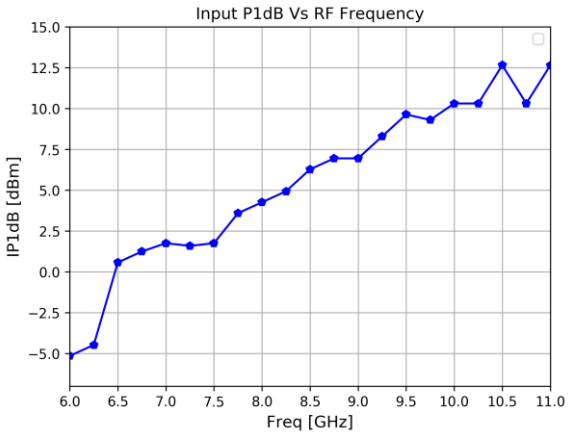


Figure 6.42: Input P1dB

Below are the measurement results for the 2–11 GHz frequency range. For Up-Conversion, the configuration with LNA_ON is always used. The following measurements were conducted with R50 set to OFF (unchecked). The lower sideband was used for the 2–6 GHz frequency range, and the upper sideband for the 6–11 GHz range.

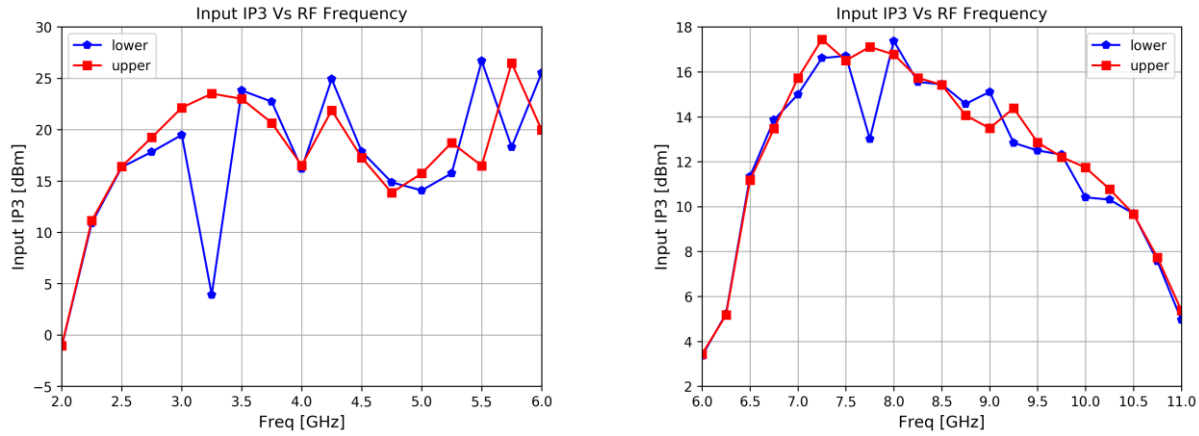


2.0 – 6.0 GHz



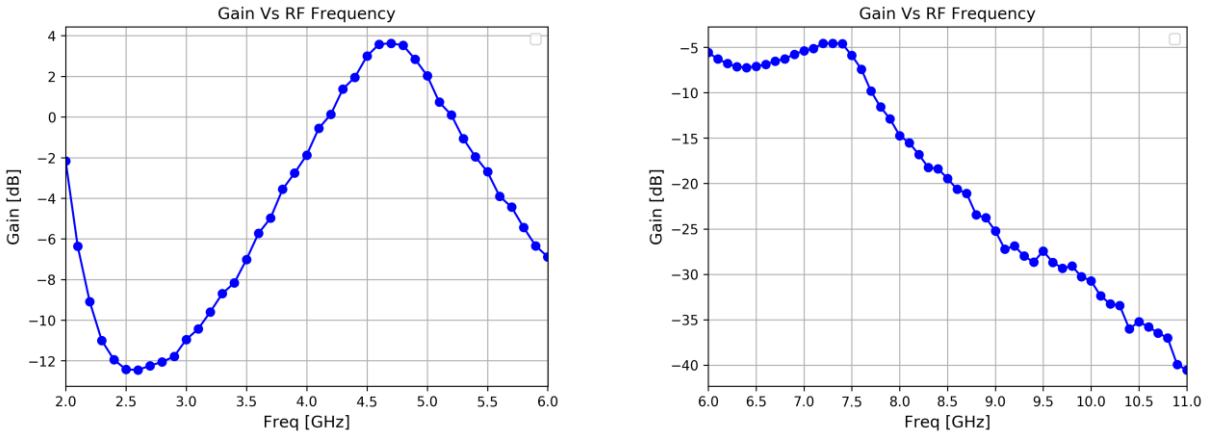
6.0 – 11.0 GHz

Figure 6.43: Input P1dB



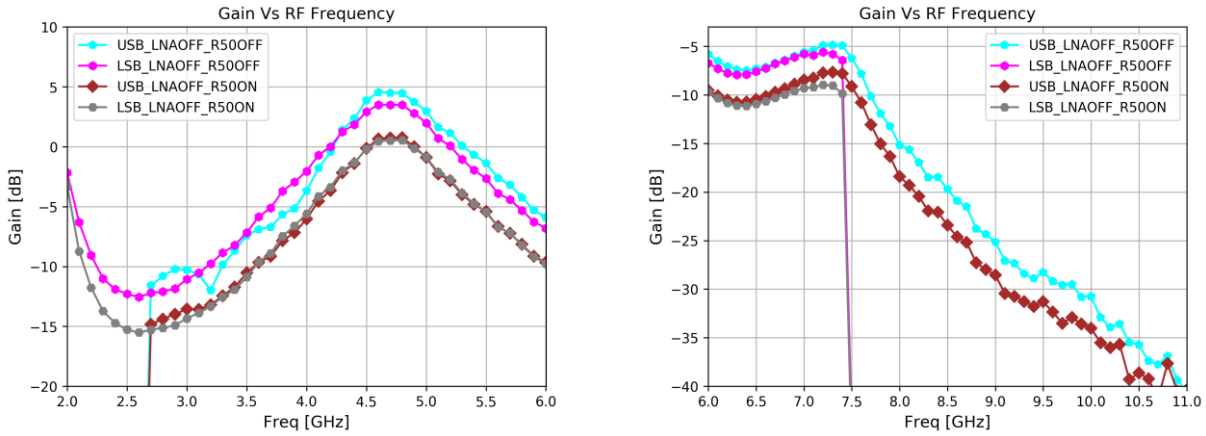
2.0 – 6.0 GHz 6.0 – 11.0 GHz

Figure 6.44: IIP3



2.0 – 6.0 GHz 6.0 – 11.0 GHz

Figure 6.45: Conversion Gain (2.0 – 11.0 GHz)



2.0 – 6.0 GHz 6.0 – 11.0 GHz

Figure 6.46: Conversion Gain for different board configurations