Software User manual for the PLL Controller software tool

Benoit Ladrange

06/07/2016

Contents

List of Figures

1 Introduction

1.1 Intended readership

This document is intended to any person wishing to drive a PLL frequency synthesiser component (which has the same characteristics as the PLL Frequency Synthesiser ADF4108 from *Analog Devices*) using an USB device acting as a "USB-to-SPI translator". However, the members of the ALTIUS project of BIRA-IASB are directly concerned by this document.

1.2 Applicability

This Sofware User Manual applies to the PLL Controller software, version 0.1.

1.3 Purpose

The User Manual's purpose is to assist the user in installing and using the PLL Controller software.

1.4 How to use this document

- **Part 2** gives an overview of the PLL Controller software.
- **Part 3** contains the instructions to get the application working.
- **Part 4** guides the user through the different features offered by the application. This part gives a step by step guide for the different operations including normal, save settings, load settings, auto sweep and frequency sweep operations.
- **Part 5** explains how to quit the application.

To use this manual effectively, you should be familiar with *Microsoft Windows* and the *Python* programming language, preferably by using *PyQt*.

1.5 Software characteristics

The PLL Controller application has been programmed using the Python language. More precisely, PyQt has been used which is the Python binding for the multi platform graphical framework Qt. This type of binding is mostly used to implement practical and graphical user interfaces (GUI applications).

1.6 Related documents

As mentioned previously, the PLL component tested with the application is an ADF4108 whose datasheet can be downloaded at `http://www.analog.com/en/products/rf-microwave/pll-synth/adf4108.html`. It is recommended to go

through this datasheet before using the PLL Controller software in order to address the correct settings to the PLL according to your purpose.

Furthermore, this document goes also along with a pdf document entitled "Python_Code_PLL_comments.pdf" which is the application's commented source code. The source code file will also be in the user's possession ("Int_Graph_PLL.py" file).

1.7 Problem reporting

If any problem occurs with the PLL Controller software application, please contact Benoit Ladrange by e-mail at benoit.ladrange@gmail.com. The source code will also be in open source so the user may directly fix prospective issues.

2 Overview

This document will present the software tool entitled "PLL Controller" used to drive a PLL frequency synthesiser : the ADF4108 from *Analog Devices*. This component is integrated onto a home-made electronic card designed by BIRA-IASB and meant to synthesise different frequencies according to specific channels : VIS, NIR and UV.
This card implements a complete Phase Locked Loop (PLL) with a loop filter and a voltage controlled oscillator (VCO).

The purpose of the application is to translate commands entered by the user on the app into understandable instructions for the PLL. The commands will be sent to an USB device which will act as a intermediary between the computer and the effective PLL component. Thanks to this device, the settings entered by the user on the app will be translated into suitable SPI wefts and presented at the input of the PLL.

3 How to get the application running

Several files will be attached to this document in order to be able to launch the application in different ways in case the basic method doesn't work.

The basic method is to simply double-click on the "PLL_Controller_version-0.1.exe" file and a window will automatically be displayed (see figure 1).

In case this method doesn't work, another file can be used : "Int_Graph_PLL.py". This file contains the source code but in contrast to the previous process, some considerations must be taken into account. These considerations will be explained below.

As mentioned before, the software has been programmed using Python so it is necessary to have the Python package installed on your computer (see Python download page : https://www.python.org/downloads).

The application has been successfully tested with Python version 2.7.12 package which includes PyQt4. If using Python version 3 or another version of PyQt, please be aware of the portability issues and therefore we do not guarantee the proper functioning of the application.

In order to have access to the source code, the user must perform a right click on the "Int_Graph_PLL.py" file and select "Edit with IDLE" which is Python's Integrated Development and Learning Environment. The user can then add or change any application's setting[1]. Once the modifications have been made, the user can check and run the program in the "Run" tab or by pressing F5.

However, if the user doesn't want to access the source code, the application can be directly launched by double clicking on the Python file.

The following figure shows the window that should be opened after running the app :

[1] The user must be aware that if any modification is made in the source file then the ".exe" file will be out-of-date and will not take into account the modifications. If the user wishes to generate a new executable file, we suggest this page : https://pyinstaller.readthedocs.io/en/stable/

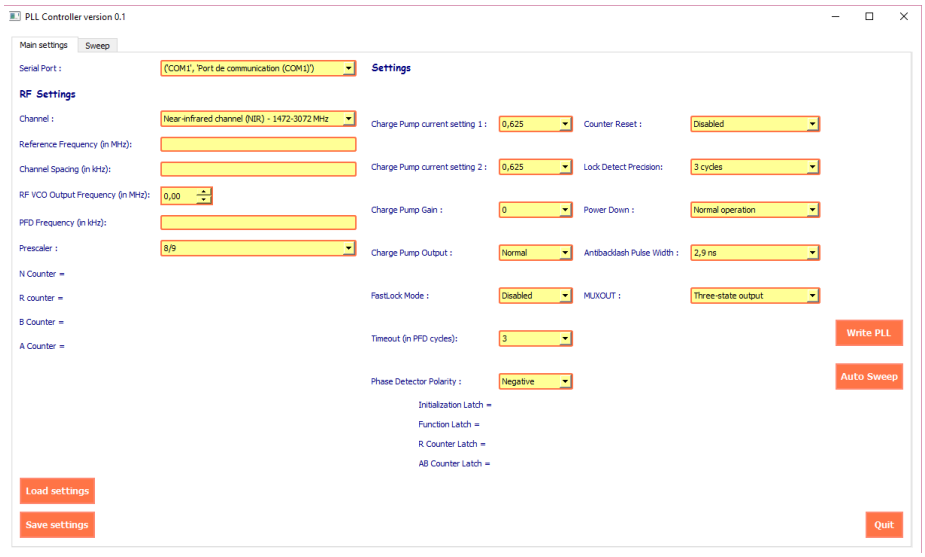

Figure 1: Screen shot of the opening window of the PLL Controller application

The next part will describe in details the different functionalities of the software and how to use them efficiently.

4 How to use the PLL Controller Software

4.1 Prerequisites

Before starting to use the application, the USB device must be connected to the computer on any available port. It is then possible for the user to run the application and he will be able to select the port in a combo box in the top left corner of the window :

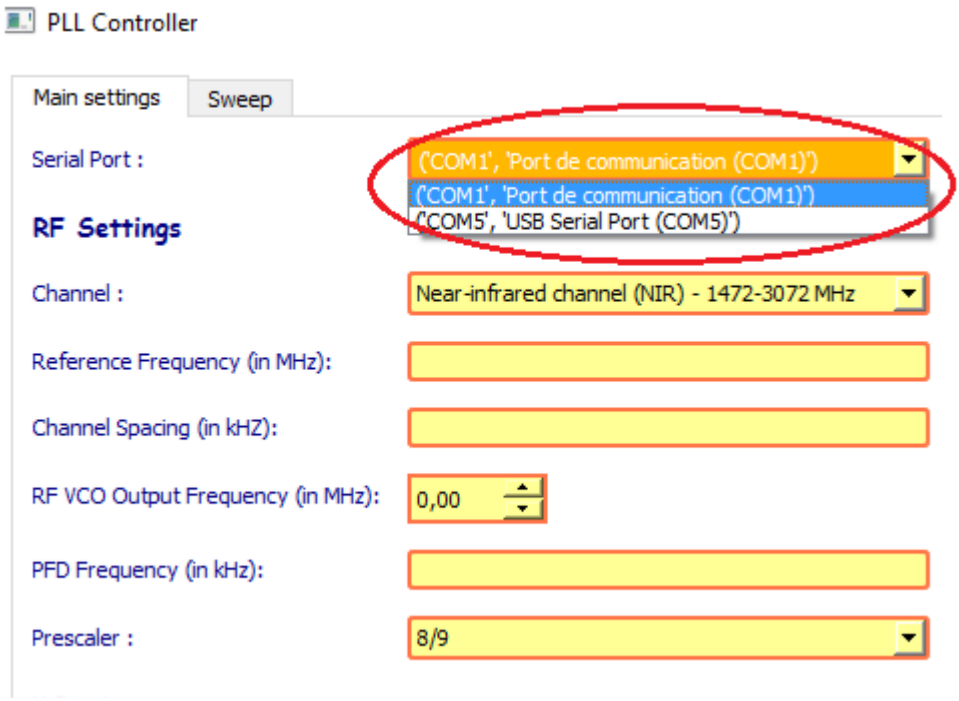

Figure 2: Selecting the serial port

As one can notice in figure 2, there are two used serial ports but COM1 is only a communication port unlike COM5 which is a USB Serial Port. Consequently, my USB device is connected to that port[2].

Once the USB device connected and the correct port selected in the combo box, the user can start changing the settings. This will be detailed in the next section.

[2]If the port to which the device is connected doesn't appear, close the app and relaunch it. If your port is still not displayed, try to connect your device to another port. An error will be displayed if you attempt to write to the PLL without selecting a serial port

4.2 "Main settings" tab's operations

4.2.1 Write PLL operation

A "Write PLL" operation consists in a basic use of the application using the eponymous push button. During such use, the user has to enter the different settings displayed in the window's first tab ("Main settings" tab). The following list explains briefly what each settings displayed in the tab accounts for :

- **Serial Port** : allows the user to select the port to which the USB device is connected. Choices depend on the devices connected to the computer as the combo box will display every serial port which has a device connected to it.
- **Channel** : allows the user to select which frequency channel he wants to work with. Choices are : NIR (Near InraRed), UV and VIS (VISible light).
- **Reference Frequency** : the user must enter which frequency is used as a reference frequency for the PLL component. The range of the value is 20 to 250 MHz.
- **Channel Spacing** : the user must select a channel spacing which will also be the value of the PFD Frequency. The range is from 40 to 100 000 kHz and the input value must be a round number.
- **RF VCO Output Frequency** : it is the frequency that will be synthesised by the VCO. The user selects the frequency he wishes to obtain at the output of the VCO by incrementing or decrementing by one channel spacing using the arrows[3]. The range of the spin box is fixed according to the channel selected previously (NIR, UV or VIS).
- **PFD Frequency** : stands for "Phase/Frequency Detector frequency". Therefore, it is the reference frequency for the phase detector of the PLL, which is the same value as the Channel Spacing[4].
- **Prescaler** : value of the PLL's prescaler (see the ADF4108 datasheet). Choices are : 8/9, 16/17, 32/33 and 64/65.
- **Charge Pump current setting 1** : allows the user to choose a current value which corresponds to the maximum value of the current coming out of the charge pump (see datasheet). Values may be 0.625, 1.25, 1.875, 2.5, 3.125, 3.75, 4.375, 5.0 mA.
- **Charge Pump current setting 2** : the ADF4108 offers the possibility to select a second charge pump current. For more information, please refer

[3] If no value has been entered in the Channel Spacing line edit, the default step of the spin box is 1 MHz

[4] If a value has already been entered in the Channel Spacing line edit, the same value has been written in the PFD Frequency line.

to the component's datasheet. The values are the same as current setting 1.

- **Charge Pump Gain** : allows, with the FastLock Mode, to select a specific PLL operation (see datasheet) using the charge pump's current settings. Values are 0 or 1.
- **Charge Pump Output** : the charge pump output may be in three-state mode (negative, positive or no output current) or in normal mode (positive or negative current). Choices are Normal or Three-state.
- **FastLock Mode** : as mentioned before, allows to select a specific PLL operation according to the Charge Pump Gain value. Choices are Disabled, Mode 1 or Mode 2.
- **Timeout** : this setting is also related to the FastLock mode and Charge Pump gain settings. Values are within the range 3 - 63 PFD cycles with a step of 4 (3, 7, 11 ...).
- **Phase Detector Polarity** : used to set the phase detector's polarity. Choices are Negative or Positive.
- **Counter Reset** : resets the PLL's internal AB and R counters if enabled. Choices are Disabled or Enabled.
- **Lock Detect Precision** : it is the number of PFD cycles (during which the PLL is locked) necessary before a lock can be detected. Choices are 3 or 5 cycles.
- **Power Down** : allows to select a programmable power-down among several modes that are explained in detail in the PLL's datasheet. Choices are Normal operation, Asynchronous power-down or Synchronous power-down.
- **Antibacklash Pulse Width** : this pulse ensures that there is no dead zone in the PFD transfer function. Choices are 2.5, 1.3 (for test mode only) or 6.0 ns.
- **MUXOUT** : the PLL component has a pin with a selectable output which can have different purposes. One can choose which signal to drive out of this pin amongst : Three-state output, Digital Lock Detect, N Divider Output, DVDD, R Divider Output, Analog Lock Detect, Serial Data Output or DGND.

After all the necessary tunings have been made, the user will be expecting to send these settings to the PLL. This is done by pressing the "Write PLL" button, at the right of the window. After pressing the button, error messages will be displayed if some settings are incorrect. Indeed, the default choices of the combo boxes may be left untouched but the line edits must be filled in, otherwise an error message will be displayed when pressing the "Write PLL"

push button.
The content of the error message is detailed and specific so that the user can situate and correct the mistake swiftly. For example, the following message box appears after pressing the "Write PLL" button without entering any value in the "Reference Frequency" line edit :

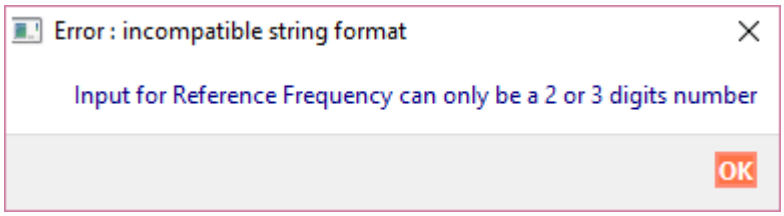

Figure 3: Example of an error message box

As one can notice in the figure above, the message explicitly states which setting is incorrect so there is no need to search everywhere to correct the mistake. Hence, this message will appear if the user enters anything else than a 2 or 3 digits number (character, string of characters, symbols ...).

However, if the user has entered a number with the correct format but out-of-range, the following message will pop-up :

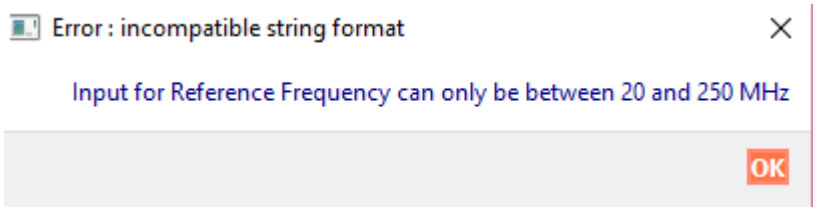

Figure 4: Another example of an error message box

Once again, the content of the message is quite explicit to correct the error rapidly.
The same kind of error messages will also be generated for the Reference Frequency line edit.

If no error messages are shown after pressing the "Write PLL" button, one can notice that other values appear on the window (see figure 5).

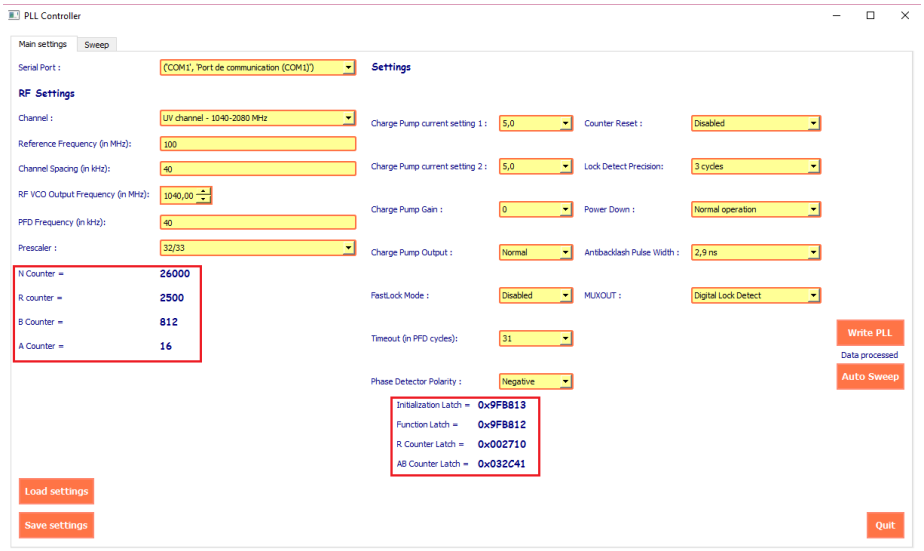

Figure 5: Values appearing after pressing the "Write PLL" button

The values appearing on the left side represent the values of the PLL's counters (N, R, A and B) that are used to perform the frequency synthesis according to the following equations (as explained in the PLL's datasheet) :

$$RFout = N \times \frac{RFref}{R}$$

with

$$N = BP + A$$

P being the prescaler's value.

These values are directly calculated by the program according to the user's choices for the output and reference frequencies and the prescaler's value.

The values appearing in the middle of the window are the hexadecimal values of the registers that will be sent to the PLL according to the latch maps described in the ADF4108 datasheet. As for the counter values, the content of the registers are automatically builded by the program.

If all the correct value have been entered and the counter and register values have been displayed, that means that all the data have been processed and are being sent through the serial port selected. To indicate this "processing state" to the user, a small message will appear right under the "Write PLL" button, on the far right side of the window :

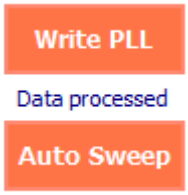

Figure 6: Label indicating that data is being processed

The counter and register values as well as the "Data processed" message will not be displayed if one of the line edit has an incorrect value, in other words when an error message has appeared.

Provided that the serial port selected actually corresponds to the right USB device, once the counter values, register values and "Data processed" message appear, then the settings have been sent to the PLL.

The next section will deal with the "Save settings" option.

4.2.2 Save settings option

If the user is happy with his settings, he has the opportunity to save them in a text file by pressing the "Save settings" push button at the bottom left corner of the window, as shown in the following figure :

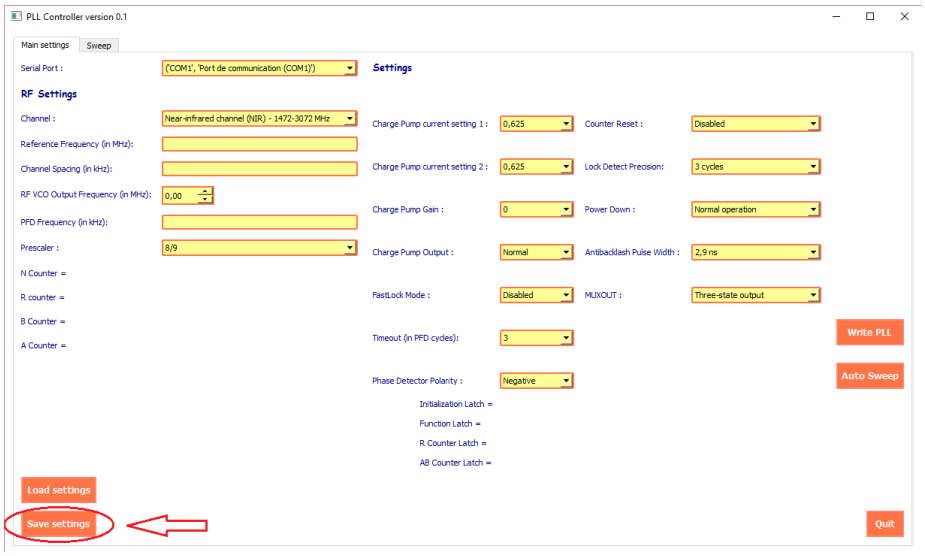

Figure 7: Position of the "Save settings" button

Once this button is pressed, a small dialog box opens in order to ask the user to enter a file name (see following figure).

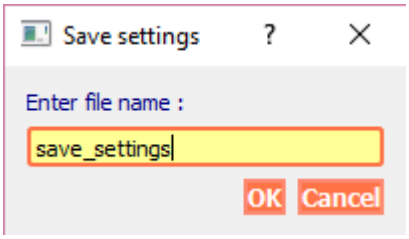

Figure 8: Dialog box opening on a "Save settings" button pressed event

As soon as the user enters a suitable file name, he may press the "OK" button and the settings will be saved in the specified file. This file will be created in the same folder as the PLL Controller's application. The user must be aware that if he enters a name that has already been used for a file in the folder, this file will be overwritten.
If everything goes well, the application will display a message box to inform of the operation's success (see following figure).

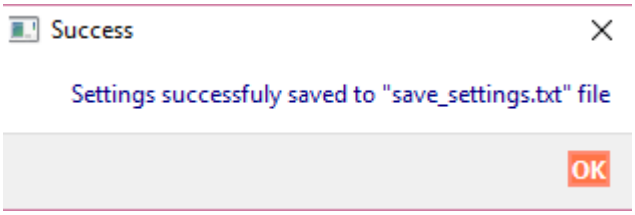

Figure 9: Message box opening after a successful save operation

However, if the user presses the "Cancel" button or enters nothing in the line edit and presses the "OK" button, an error message will be displayed informing the user that the operation was unsuccessful due to a wrong input for the file name (see figure below).

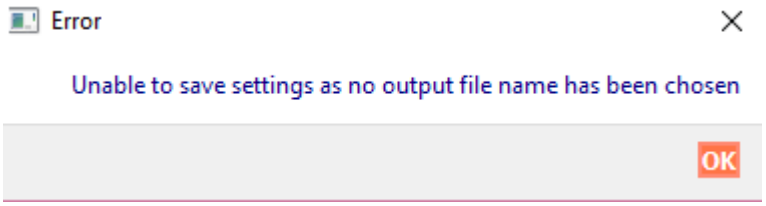

Figure 10: Error message displayed when an unsuccessful "Save settings" operation occurs

This problem is solved by pressing the "Save settings" button again and

entering a suitable name.

The following section will deal with the "Load settings" operation.

4.2.3 Load settings option

This specific option can only be used if a "Save settings" operation has already been conducted (see section 4.2.2 page 13).

As mentioned in the previous section, once the user has successfully saved his settings, he can load them by pressing the "Load settings" button situated at the bottom left corner of the window, right above the "Save settings" button (see figure 7 in section 4.2.2 page 13).

Once this button is pressed, a small dialog box opens in order to ask the user to enter the name of the file containing the settings he wishes to load (see following figure).

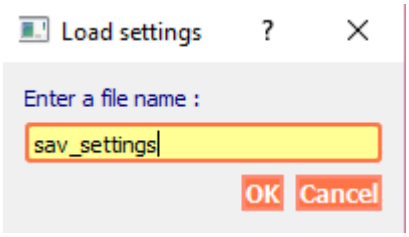

Figure 11: Dialog box opening on a "Load settings" button pressed event

As soon as the user enters a suitable file name, he may press the "OK" button and the settings will be loaded. Indeed, the user will witness the automatic and immediate update of the different settings according to the settings saved in the chosen file.

However, if the user presses the "Cancel" button or enters nothing in the line edit and presses the "OK" button, an error message will be displayed informing the user that the operation was unsuccessful due to a wrong input for the file name (see figure below).

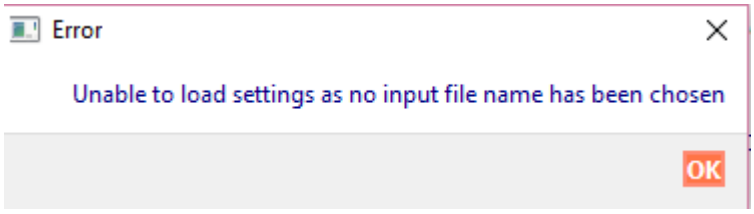

Figure 12: Error message displayed when an unsuccessful "Load settings" operation occurs

This problem is solved by pressing the "Load settings" button again and entering a suitable name.

If the user enters a file name that does not match any file present in the folder, another kind of error message will be opened stating explicitly that the file doesn't exist, as shown in the figure below :

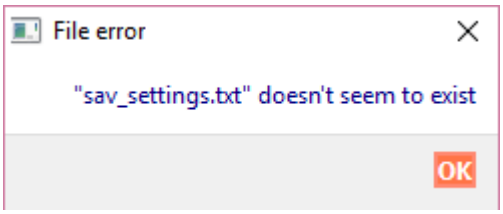

Figure 13: Error message displayed if the file name doesn't match any existing file of the folder

In order to resolve this problem, the user must open the folder containing the PLL's Controller application's files and check if the file he is looking for is actually there.
It is likely that a typing mistake in the dialog box was the reason of the operation's failure so the user may press the "Load settings" button again and try to enter the file's name again.

The next section will deal with the "Auto sweep" operation.

4.2.4 Auto sweep operation

The PLL Controller application has an "Auto sweep" feature that allows to generate successive output frequencies automatically every 2 seconds. Indeed, the application proposes three frequency channels to work with (NIR, UV and VIS) each having a particular frequency range. For the NIR channel, the frequency range is 1472 - 3072 MHz, for the UV channel it is 1040 - 2080 MHz and finally VIS channel range is 1152 - 2496 MHz. Setting a specific channel induces a change of the range of the RF VCO Output Frequency spin box.

An auto sweep is an operation that consists in performing a frequency sweep through the range of that spin box starting with the minimum frequency value until the maximum with an increment equal to the value entered in the Channel Spacing line edit.
For example, performing an auto sweep for the UV channel will generate a frequency of 1040 MHz, then 1040 + channel spacing and so on.

For the user to employ this functionality, a simple press on the "Auto Sweep" button will do the trick. The following figure shows the location of the push button on the application's window :

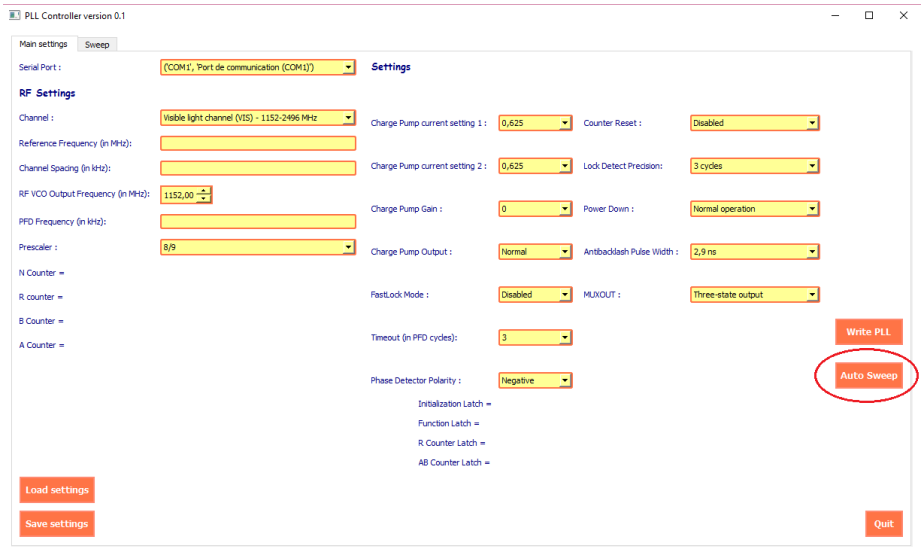

Figure 14: Location of the "Auto Sweep" button in the application

After pressing the button, the values of the PLL's counters and registers will appear on the left side of the window and in the middle respectively (see figure 5 in section 4.2.1 page 12). The contents of the N, B and A counters as well as the AB Counter Latch register will dynamically change over time as they are linked to the varying output frequency. In addition, the value of the RF VCO Output Frequency spin box will also change dynamically, incremented by one channel spacing every 2 seconds.

The user will be able to assist directly to the evolution of the auto sweep thanks to another element which is a progress bar, displayed beneath the "Auto Sweep" button as soon as this button is pressed (see figure below).

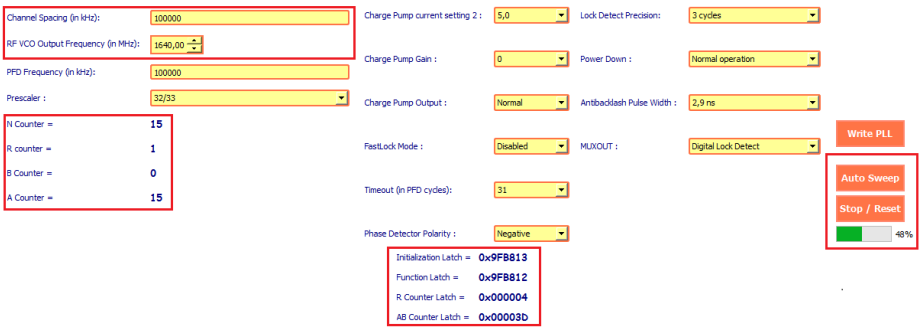

Figure 15: Screen shot of the application's window while performing an auto sweep operation (not the complete window)

If the user has entered a very small channel spacing value, the auto sweep may take a certain time to be completed and the progress bar will also take time before displaying the progress as it is quite imprecise.

It is still possible to interrupt the sweep by pressing the "Stop / Reset" button located above the progress bar (see figure above). A certain number of actions occurs after this button is pressed :

1. The RF VCO Output Frequency spin box is reset to its minimal value.
2. The progress bar is cleared and the percentage indicating the progress is removed.
3. The values of the counters and registers are also cleared.
4. The timer counting the 2 seconds between two increments is stopped (although it won't be noticeable by the user).

Figure 16: Actions performed on a "Stop / Reset" button pressed event (not the complete window)

Even if the user waits until the end of the automatic frequency sweep, he will have to press the "Stop / Reset" button in order to be ready for another auto sweep operation.

The "Stop / Reset" button and the progress bar will be removed as soon as the user presses the "Load settings" or the "Write PLL" button.

The following section will discuss the frequency sweep feature.

4.3 "Sweep" tab's operation

4.3.1 Prerequisite

In order to successfully perform an operation in the "Sweep" tab, the user must previously change the desired settings in the "Main settings" tab. However, the

choice for the Channel combo box will not be relevant and will not be used for a "Sweep" process.

If the Reference Frequency and/or Channel Spacing line edits have not been filled in by the user, the kind of error message presented in section 4.2.1 page 11 will be displayed.

It is important that the "Main settings" tab's parameters are seriously dealt with because a "Sweep" operation is also an action leading to the writing of the PLL component's registers. Therefor, this device must be correctly tuned according to the user's needs.

Once the "Main settings" tab filled, the user may change the window's current tab. The next steps will be detailed in the following subsection.

4.3.2 Sweep operation

If the prerequisites have been assimilated, a "Sweep" operation may be considered.

This functionality resembles the Auto sweep option but there are nevertheless noticeable differences.

First of all, in order for the user to have access to this operation, he will have to go to the second tab of the application's window by performing a left click on the corresponding tab ("Sweep" tab) :

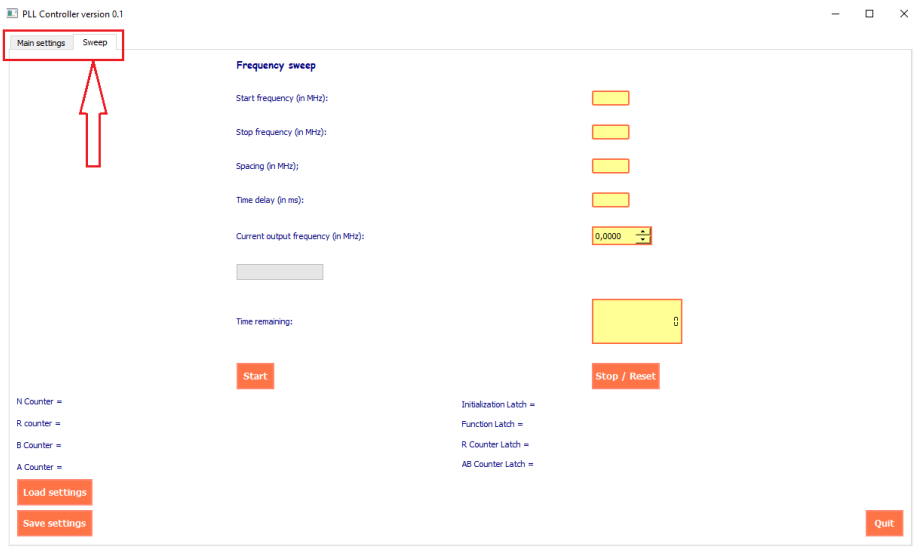

Figure 17: How to change the window's tab

The window shown on the figure above will then appear.

A "Sweep" operation consists in generating a series of consecutive frequencies within a frequency range given by the user. To do so, he will have to enter a starting and finishing point, a step and a time delay.

As one can notice, the main difference between an "Auto sweep" and a "Sweep" process is that, in a "Sweep" action, the user can enter his own frequency range while this range is fixed by the Channel combo box in an "Auto sweep" operation.

The following list explains briefly what each settings displayed in the tab accounts for :

- **Start Frequency** : this frequency value will be the starting point of the frequency sweep. The line edit accepts values within the range 1000 to 2400 MHz.
- **Stop Frequency** : this value is the stop point of the frequency sweep. The value must be within the range 1000 to 2400 MHz and must be greater than the start frequency input.
- **Spacing** : in order to go from the start to the stop frequency, the user must enter a step which corresponds to the frequency increment. This value must be lesser than Start - Stop frequencies and may be a floating number with 4 decimal points maximum.
- **Time delay** : it is possible for the user to set a delay between each increment of the output frequency. The value must be in the range 0 - 10 000 ms.
- **Current output frequency** : this spin box displays the frequency that must be obtained at the output of the VCO. The range of the spin box is given by start and stop frequencies and this value will be incremented by one spacing each time period given by the "Time delay" line.
- **Time remaining** : this object displays the time remaining before reaching the stop frequency. The displayed format is "00 DAYS 00:00:00" to indicate days, hours, minutes and seconds.

Every line edit must be filled in and once it is done, the user may press the "Start" button (right beneath the "Time remaining" label) to launch the frequency sweep. If an incorrect input is detected in a line edit, a specific error message will pop-up, the same kind as shown in figures 3 and 4 of section 4.2.1 page 11.

If no error message is displayed, then the sweep will begin and several elements will indicate the progress of the operation to the user :

- The Current output frequency spin box is dynamically incremented by one spacing every time delay.

- A progress bar displays the evolution of the operation just beneath the "Current output frequency" label.
- The time remaining before the end of the sweep is displayed next to the "Time remaining" label.
- The values of the PLL device's internal counters are displayed (counters N, R, A and B displayed on the bottom left side of the window) along with the PLL's registers (on the bottom right side of the window). The contents of the N, B and A counters as well as the AB Counter Latch register will dynamically change over time as they are linked to the varying output frequency.

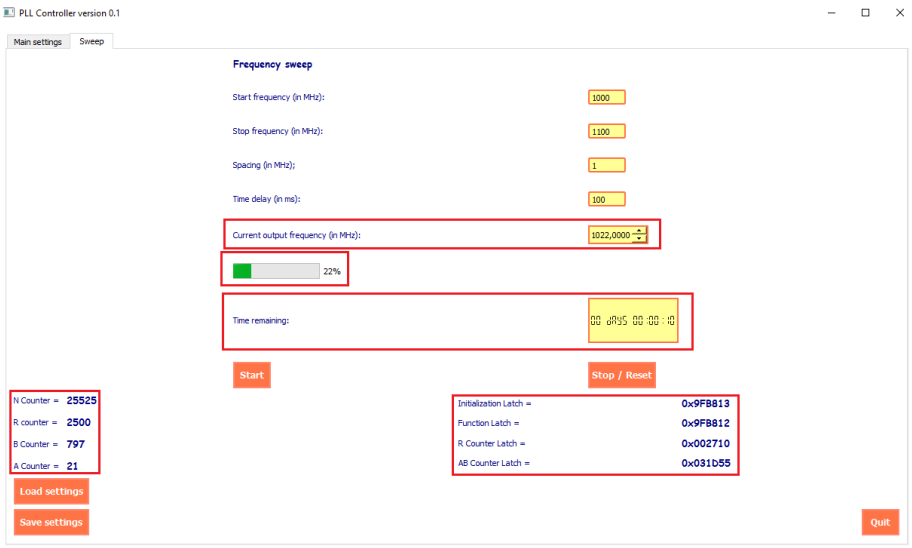

Figure 18: Actions performed on a "Start" button pressed event

If the user has entered a very small spacing value or a high time delay, the sweep may take a certain time to be completed and the progress bar will also take time before displaying a progress as it is quite imprecise.

In addition, the remaining time displayed will be decremented by one time delay at every output frequency increment. So the user mustn't worry if the remaining time doesn't seem to change as it is likely that the time delay he has entered is too high to expect a 1 second precision.

The end of the sweep will be indicated by a small label displayed at the right of the progress bar where the user can read "Sweep completed".

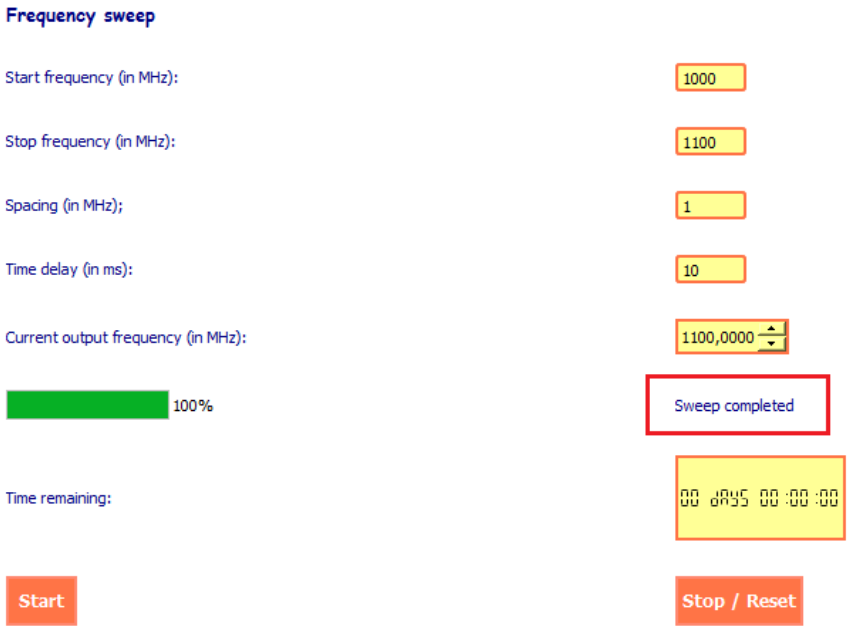

Figure 19: Screen shot of the "Sweep" tab at the end of a frequency sweep operation (not the complete window)

It is also possible to interrupt the frequency sweep by pressing the "Stop / Reset" button located under the time remaining (see figure above). A certain number of actions occur after pressing this button :

- The Current output frequency spin box is reset to its minimal value (which is equal to the Start frequency value).
- The progress bar is cleared and the percentage indicating the progress is removed.
- The time remaining is also reset to its nominal value (which is "00 DAYS 00:00:00").
- The label indicating the end of the frequency sweep is removed.
- The values of the counters and registers are also cleared.
- The timer counting the time period equal to the value of the "Time delay" line edit is stopped (although this won't be noticeable by the user).

Figure 20: Actions performed on a "Stop / Reset" button pressed event

However, if the user has waited until the end of the frequency sweep (progress bar displaying a 100% progress and time remaining is "00 DAYS 00:00:00" with a relative error of 1 second), he still has to press the "Stop / Reset" button so that the application is ready to perform another "Sweep" operation.

The next subsection will deal with the "Load settings" and "Save settings" operations.

4.3.3 Load and save settings operations

As one can notice, two push buttons are displayed at the bottom left corner of the "Sweep" tab :

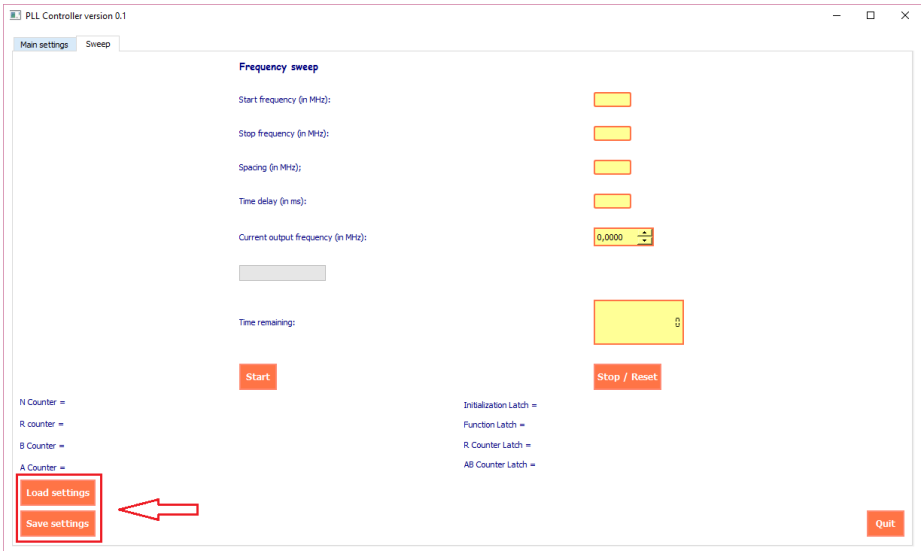

Figure 21: "Load settings" and "Save settings" buttons of the "Sweep" tab

These two buttons have exactly the same purposes as the two same buttons of the "Main settings" tab.

For more information on the "Save settings" and "Load settings" operations, please refer to sections 4.2.2 page 13 and section 4.2.3 page 15 respectively.

5 How to quit the application

In order to close the PLL Controller software, a "Quit" button has been implemented and displayed on both of the window's tabs. Therefor, the user may exit the application from any tab only by pressing this button :

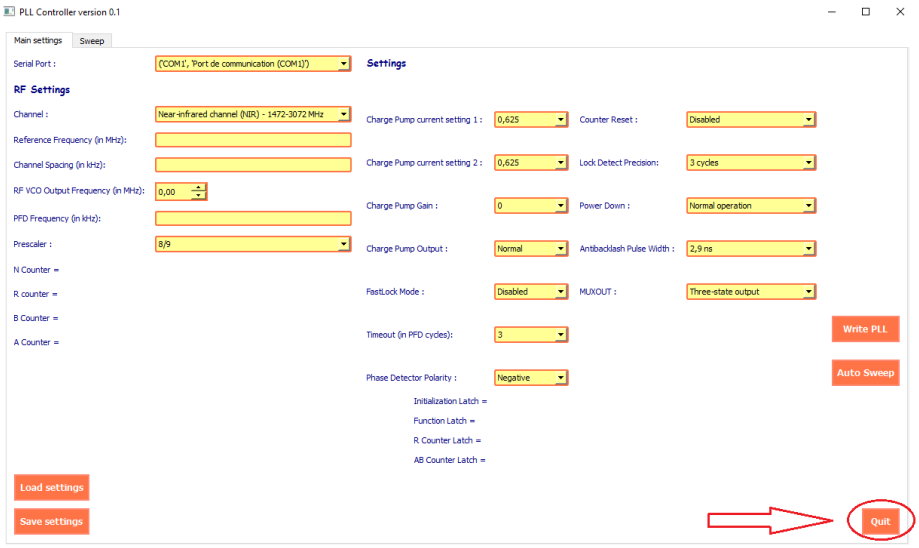

Figure 22: "Quit" button of the "Main settings" tab

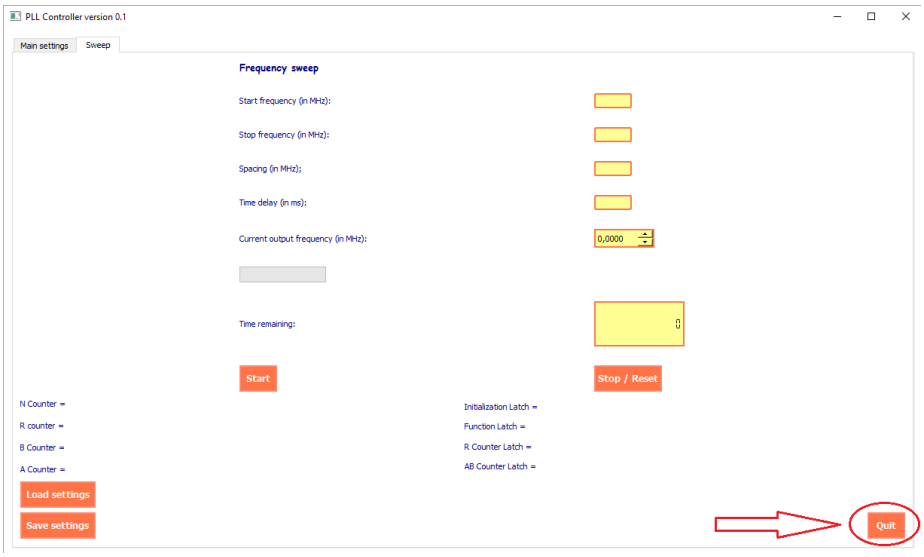

Figure 23: "Quit" button of the "Sweep" tab

Two actions occur when pressing this button :

1. The application is stopped.
2. The window is closed.

The basic way to close a window in *Microsoft Windows* is still available to close the application by clicking the "X" button in the upper right corner.

A Error messages and recovery procedures

Examples of error messages can be found in section 4.2.1 page 11, figures 3 and 4. These types of error occur when the user enters a wrong value in a line edit.

The error detection mechanism starts by analysing the format of an input string (for example checking if the data entered is a 4 digits integer). If this format is correct, then the program tests if the value is within a certain range, in order to detect any out-of-range number.
Therefor two kinds of error messages can be displayed per line edit present in the app : the first one concerns the format of the input string and the second informs the user that the value entered is out-of-range.

Each error message states explicitly which line edit is concerned and reminds the format or the range to which the value must comply.

This appendix will show all the error messages that can be addressed to the user by the software during use[5].

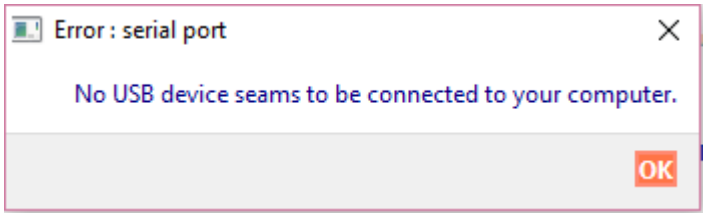

Figure 24: Error message : no serial port has been selected

If the error message above appears, that means that no choice has been displayed in the "Serial Port" combo box. If such message is displayed, the user must check if the USB device is well connected to the computer. After doing so, the application must be relaunched and if the combo box is still empty, you may try to connect the device to another port and relaunch the application.

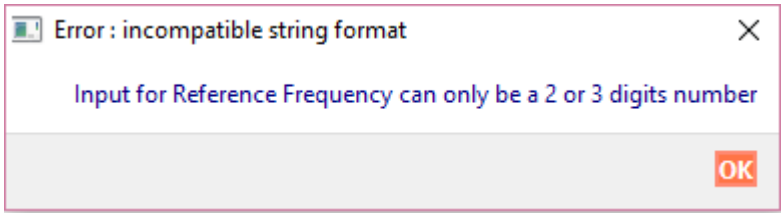

Figure 25: Error message : wrong format for the Reference Frequency

[5] As the PFD Frequency and Channel Spacing line edits are linked, only the error messages bound to the PFD Frequency are displayed.

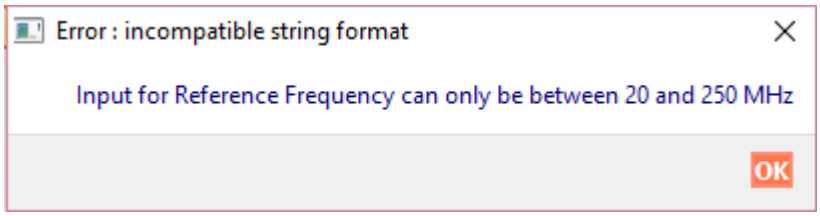

Figure 26: Error message : out-of-range value for the Reference Frequency

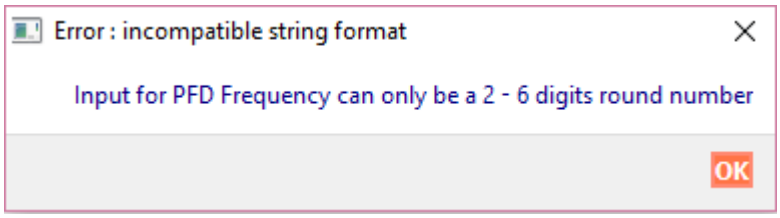

Figure 27: Error message : wrong format for the PFD Frequency

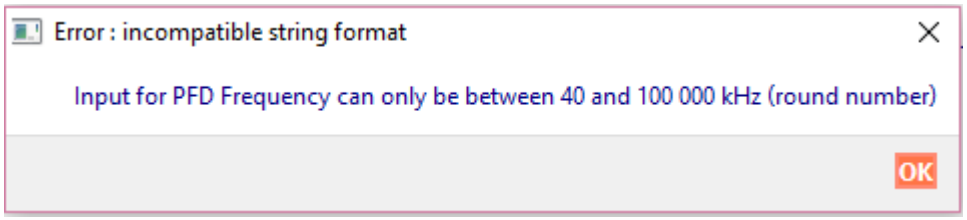

Figure 28: Error message : out-of-range value for the PFD Frequency

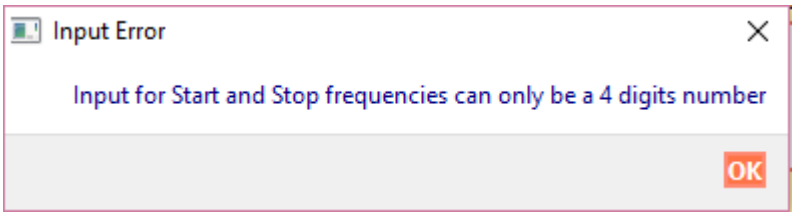

Figure 29: Error message : wrong format for the Start and Stop frequencies ("Sweep" tab)

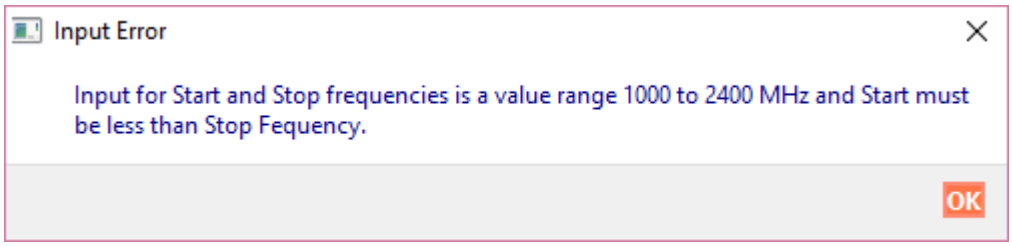

Figure 30: Error message : out-of-range value for the Start and Stop frequencies

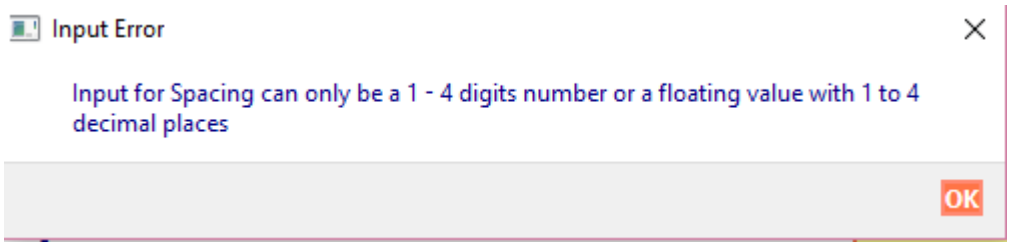

Figure 31: Error message : wrong format for the Spacing ("Sweep" tab)

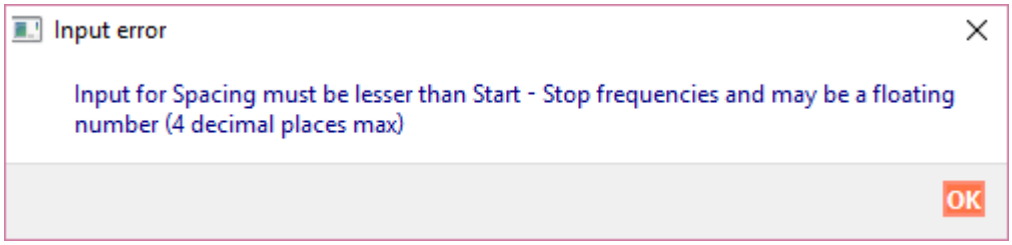

Figure 32: Error message : out-of-range value for the Spacing

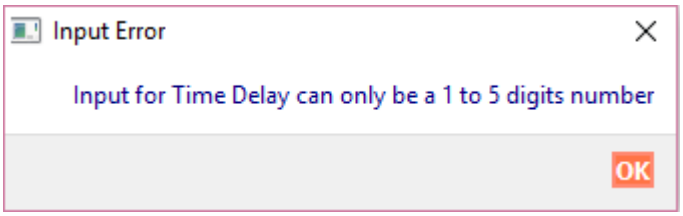

Figure 33: Error message : wrong format for the Time Delay ("Sweep" tab)

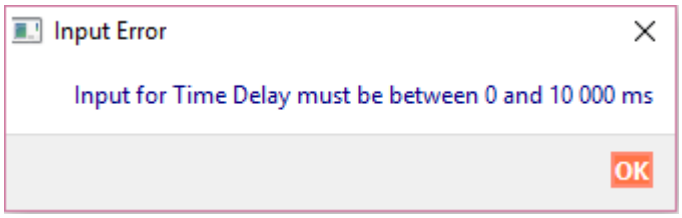

Figure 34: Error message : out-of-range value for the Time Delay

The content of the error messages is sufficiently explicit to allow the user to correct them immediately by reentering a correct value in the relevant line edit.

Error messages can also pop-up while dealing with text files for the "Save settings" and "Load settings" operations. This messages are shown and explained in sections 4.2.2 page 13 and 4.2.3 page 15.

B Glossary and abbreviations

Word	Meaning
Auto sweep	It is one of the software's possible operation. It consists in browsing through a specific frequency range (NIR, VIS or UV) automatically to generate consecutive frequencies thanks to a fixed step (channel spacing).
Channel spacing	Frequency step allowing to go from the beginning to the end of a frequency range.
Charge Pump	Electronic circuit that allows to create higher or lower voltage power sources.
Combo box	Graphical user interface widget also known as "drop down list".
Frequency sweep	It is one of the software's possible operation. It allows to browse through a frequency channel given by the user. He will indicate a starting and stopping point, a step and a time delay to perform the sweep. The purpose is to generate consecutive frequencies within a particular range chosen by the user.
GUI	Graphic User Interface : interface that allows the user to interact with electronic devices through graphical icons.
NIR	Near InfraRed frequency channel.
Phase detector	Electronic comparator that is the centrepiece of a PLL. It allows to compare the phases of the input and output signals.
PLL	Phase Locked Loop : control system that generates an output signal whose phase is directly linked to the phase of an input signal.
Port	Serial port of the computer (commonly a USB port).
Prescaler	Electronic counting circuit used to reduce a high frequency signal.
Python	Object-oriented programming language.
PyQt	Python binding in order to use the Qt library used for graphical framework.
SPI	Serial Peripheral Interface : synchronous serial communication protocol.
Spin box	Graphical user interface widget with arrowed buttons to select a value.

Three-state output	Output that can be either a logic 1, a logic 0 or a in high impedance mode.
UV	UltraViolet frequency channel.
VCO	Voltage Controlled Oscillator : electronic oscillator whose oscillation frequency is controlled by a input voltage.
VIS	VISible light frequency channel.

C References

[1] Qt library : http://www.qtcentre.org/content/

[2] PyQt binding : https://wiki.python.org/moin/PyQt

[3] Radio-Electronics : Phase Locked Loop Tutorial : http://www.radio-electronics.com/info/rf-technology-design/pll-synthesizers/phase-locked-loop-tutorial.php

[4] IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) : *Charge Pump, Loop Filter and VCO for Phase Lock Loop*. Kashyap K. Patel, Nilesh D. Patel, Kruti P. Thakore. Volume 2, issue 4 (May - June 2013). http://iosrjournals.org/iosr-jvlsi/papers/vol2-issue4/D0242125.pdf?id=2039

[5] *Rapid GUI Programming with Python and Qt*. Mark Summerfield, 18 October 2007.