

1. Installation

1.1. Software Requirements

This AFM controller software is a LabVIEW project and running it requires having LabVIEW installed on the computer. In order to install LabVIEW go to: <https://www.ni.com/en-us/support/downloads/software-products/download.labview.html> and install the 32-bit release of 2016. For details on the required packages refer to appendix A (List of Required LabVIEW Products).

Make sure that you copy the folder containing the LabVIEW project on your computer.

1.2. Hardware Requirements

2. Getting Started

Open the folder containing the project file and open the project file (with LabVIEW 2016 32-bits).

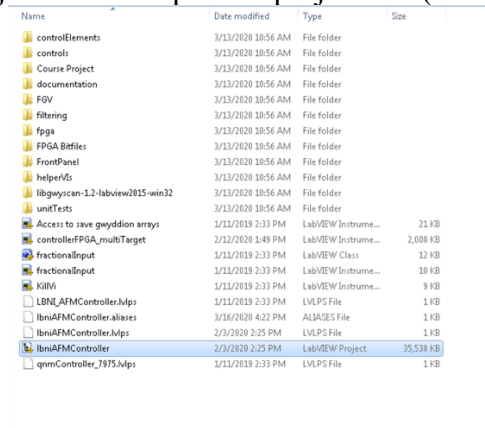


Figure 1 Project file in the project folder

In the project window, expand the **FrontPanel** folder and open **Menu.vi**.

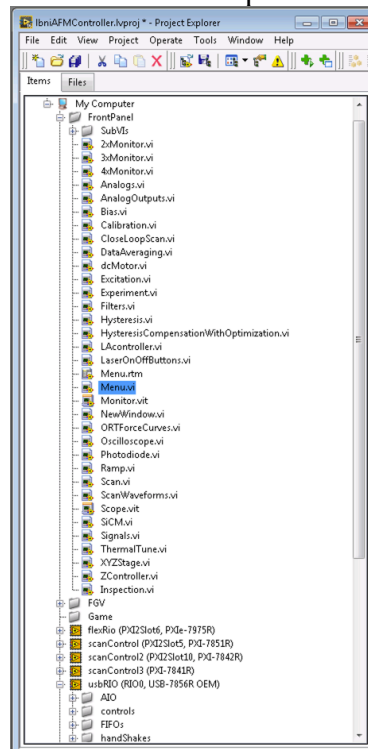


Figure 2 Project Explorer

This **Menu.vi** is the main window from which you can open the other parts of the project. Make sure that the controller is turned on and connected to the computer. If this is the first time that you have copied the project files on your computer, there are some steps you have to take in order to define the location of the bit-files for

User Manual: LBNI AFM Controller Software

the project. You can find these steps in appendix B (Configuring the address of the bit-files for the FPGA module) in the section for setting the address of bit-files.

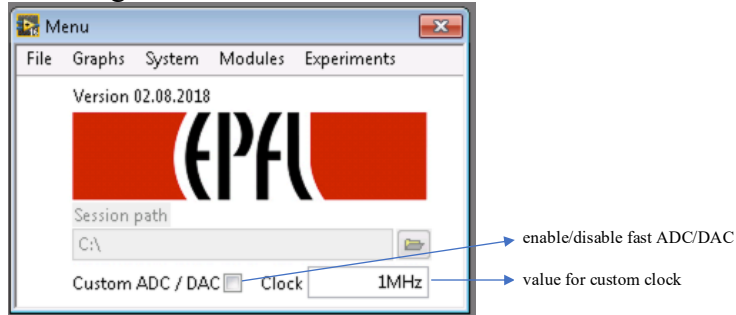


Figure 3 Menu.vi

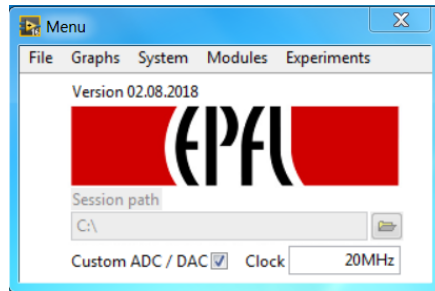


Figure 4 Menu VI with custom ADC/DAC

The fast ADC/DAC option allows you to set a custom clock (in the range of 1-80 MHz) on analog input 0 (vertical deflection) and analog output 3 (excitation signal in PORT mode).

You can access the following windows from Menu:

User Manual: LBNI AFM Controller Software

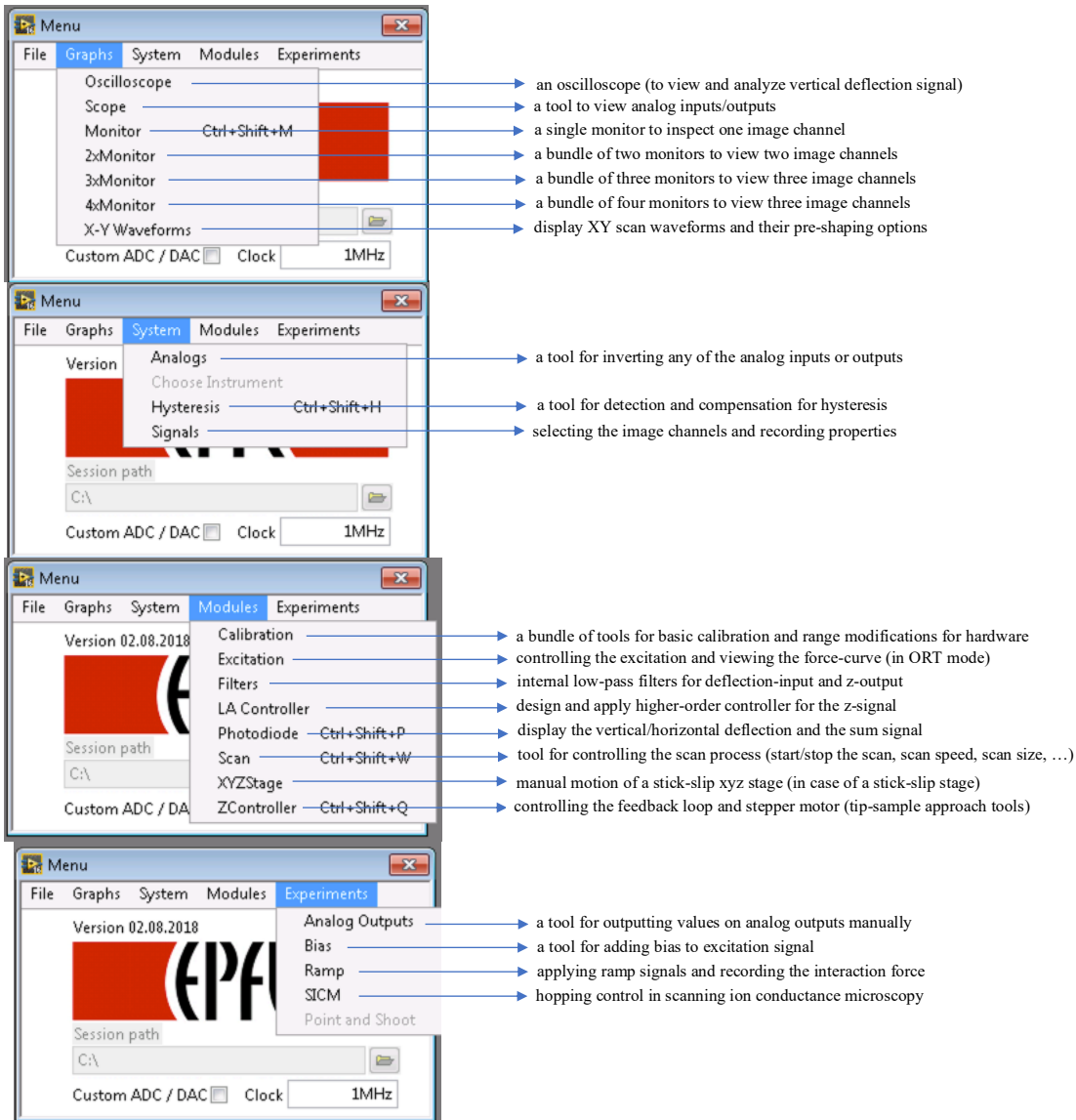


Figure 5 Different windows accessible from Menu VI

2.1. Imaging in Contact Mode

In order to familiarize yourself with the use of this controller, you can follow these steps to start imaging in contact mode.

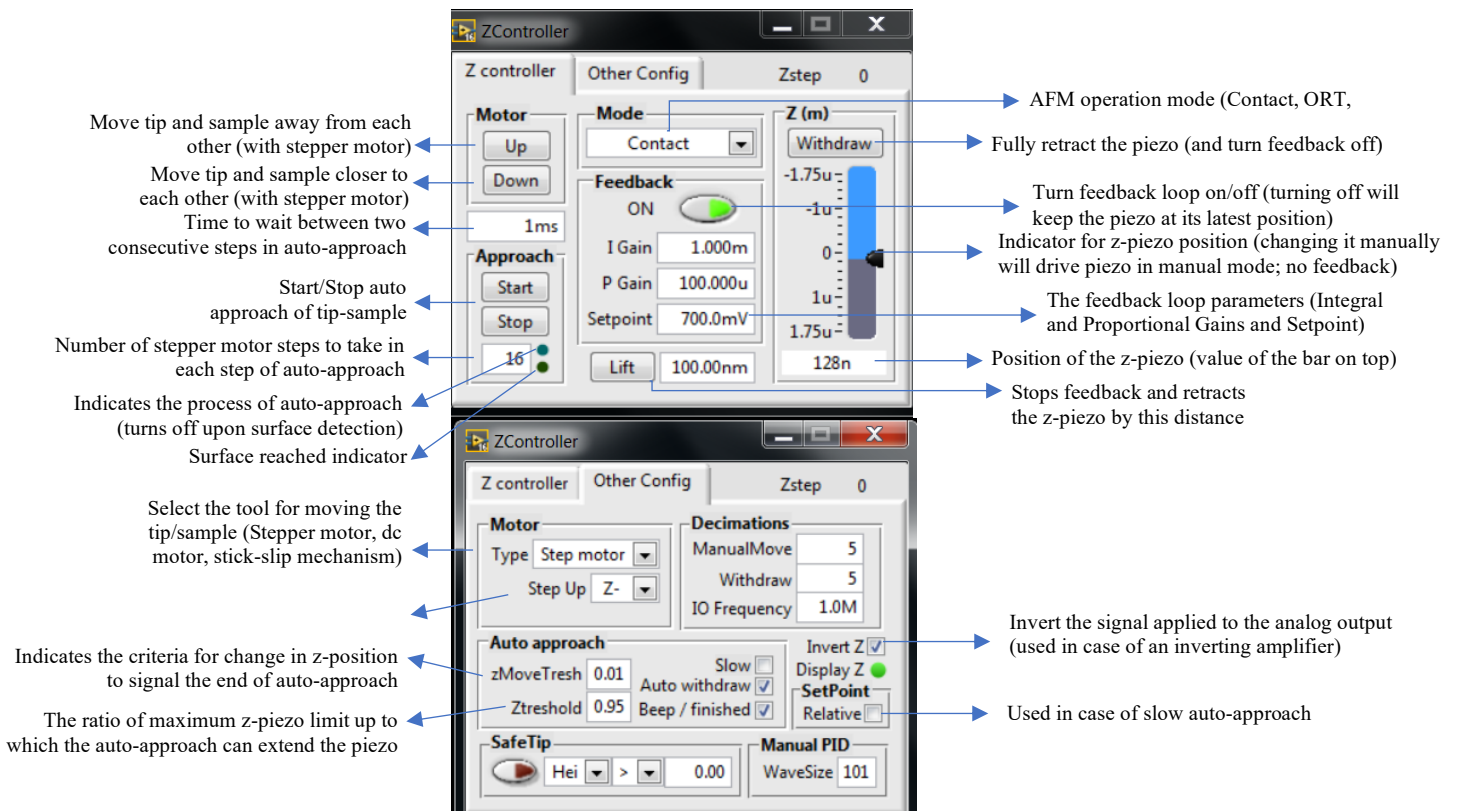
3. Documentation on Main VIs

In this section, each VI and its properties are described in details.

3.1. ZController (ZController.vi)

This is the VI responsible for controlling the position of the z-piezo and the tip-sample approach tool (stepper motor or dc motor). The main functionalities are:

- Changing the feedback loop parameters (setpoint and PI gains)
- Manual motion of the z piezo actuator
- Manual motion of the stepper motor (or dc motor)
- Auto approach of tip and sample



The **Motor** pane has buttons for activating the approach mechanism (which can be stepper motor, dc motor, or stick-slip mechanism) to move the tip and sample away from (**Up**) or toward each other (**Down**).

The operation mode of the AFM should be selected from the **Mode** pane. (either **Contact**, **OR Tapping**, or **AUX**)

The **Feedback** pane is used for manipulating the feedback loop of the AFM. To turn the feedback loop on/off, set the values for integral (**I Gain**) and proportional gain (**P Gain**), and the **Setpoint**.

The **Approach** pane is used to start or stop the auto-approach of tip-sample. In different cases of approach mechanisms (stepper motor, dc motor, or stick-slip) a parameter that regulates the speed of the auto-approach appears at the bottom of the pane. In case of the stepper motor, it is the number of steps that the motor takes in each stage of the approach. In case of a dc motor, it is the voltage applied to the motor.

The **Z(m)** pane is used in case of manual motion of the piezo actuator. Changing anything in this pane, turns off the feedback loop automatically and takes the piezo to the state regulated by this pane. **Withdraw** is used to retract the piezo to its shortest length (the top most position on the scale bar). If **Withdraw** is held down for more than 2 seconds, the motor will automatically move the tip and sample further away from each other. The scale bar represents the status of the piezo extension (the values depend on the piezo actuator that is used on the AFM setup, and they correspond to the full range of piezo displacement). The corresponding value is also represented at the bottom part of the pane. To change it manually, one should either move the scale bar with mouse or type the value on the bottom field.

User Manual: LBNI AFM Controller Software

The **Lift** pane is used to retract the piezo with the value entered in its corresponding field.

The **Motor** pane is used to select the approach instrument. The approach instrument can be a stepper motor, a dc motor or a stick-slip mechanism. The **Step Up** is used to change the direction of the approach by the motor relative to the piezo.

The **Decimations** pane is used to regulate the speed of value change of the z-piezo in the manual mode of motion. The values for **ManualMove** and **Withdraw**, are inversely proportional to the speed by which the output value for the z-piezo is updated in manual motion. The **IO Frequency** regulates the frequency by which the PID (feedback loop) is executed. In ORT mode, the feedback loop frequency is equal to the excitation frequency.

Invert Z button is used to invert the value applied to the output for the z-piezo. In case of an inverting amplifier, this option should be used. When it is ticked, the voltage applied to the analog output (AO2) corresponds to the value represented in this VI (the same sign), and if it is not ticked, the value sent to analog output has the inverse sign of what is represented here.

3.2. Scope (Scope #.vi)

This VI is used to view the value sent to z-piezo or any of the analog outputs. Each scope can be used to view one signal and to view more than one signal, you should open several instances of this VI.

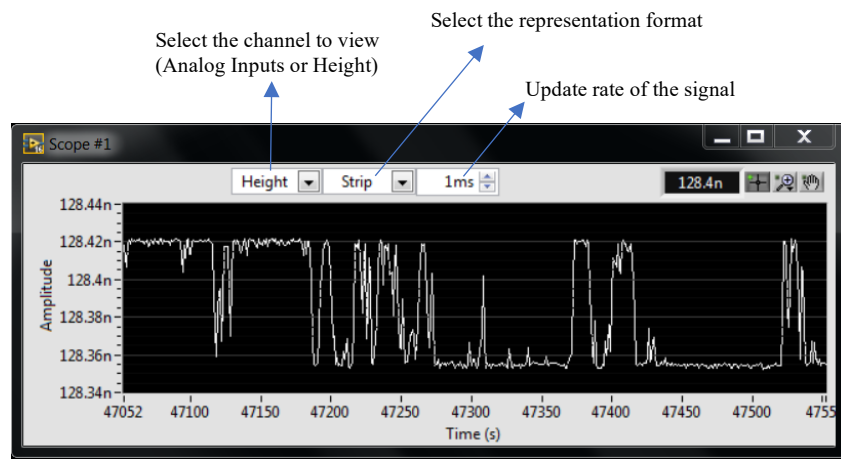


Figure 7 Details of the Scope VI

The signal that you can view is either one of the analog inputs or the signal sent to the z-piezo (Height).

You can also choose to view the signal in **Strip**, **Scope**, or **Sweep** modes. In case of the **Strip** mode, the x-axis moves as more samples are collected. In **Scope** and **Sweep** modes, however, the x-axis is stationary and the signal updates sample by sample.

Note that the sampling rate is not as high as the actual controller, and the period of the sampling can be tuned from the corresponding control on the VI on top (with a minimum value of 1ms).

The amplitude of this graph is in Volts for the analog inputs and in micrometers for the **Height** signal.

3.3. Photodiode (Photodiode.vi)

This VI is used to view the state of the laser reflection on the photodiodes. It uses data from AI0 (Vertical Deflection), AI1(Horizontal Deflection), and AI2(Sum) to show the laser spot on this virtual photodiode.

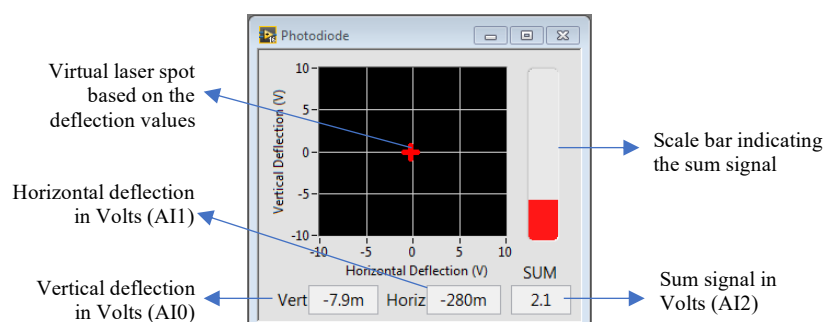


Figure 8 Details of the Photodiode VI

3.4. Scan (Scan.vi)

This VI is used for controlling the scan process. To set the parameters, command the start, pause, and stop of the scan process, and save the image.

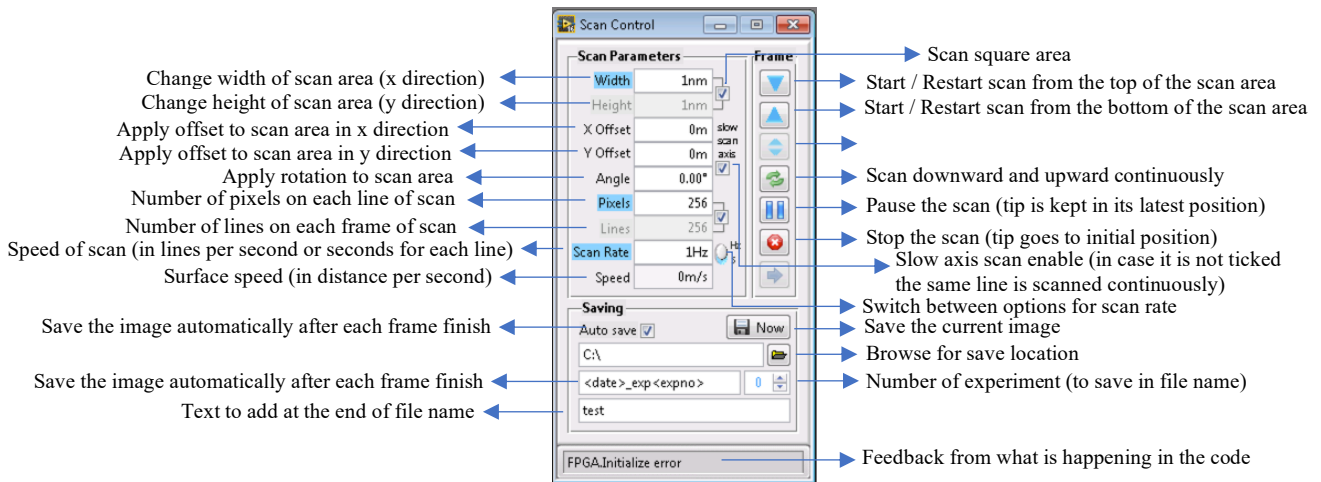


Figure 9 Details of the Scan VI

The **Scan Parameters** pane has parameters for changing the scan area, its size, offset, and rotation angle as well as pixels and speed. The values for **Pixels** determine how many points will be saved on each image. Since the number of points that are recorded are usually higher than the number of pixels, several data points are averaged to create data for each pixel point.

The **Frame** pane has buttons to start, pause, and stop the scan. It can also be used to control the direction of the scan. The down-arrow button (first button in the pane) starts the scan from the top of the scan area and continues downward line by line. The up-arrow button (the second button in the pane) starts the scan from the bottom of the scan area and continues line by line upward. Whenever the stop button is pressed, the tip goes to its initial position (center of the scan area), but the pause button pauses the scan and keeps the tip at its latest position.

In the **Saving** pane, one can select the options for naming the file that is saved, including the date and a number for each experiment (image).

3.5. Excitation (Excitation.vi)

When using the AFM in ORT mode (including PORT), this block is used to control the excitation signal and to read the recorded background curve.

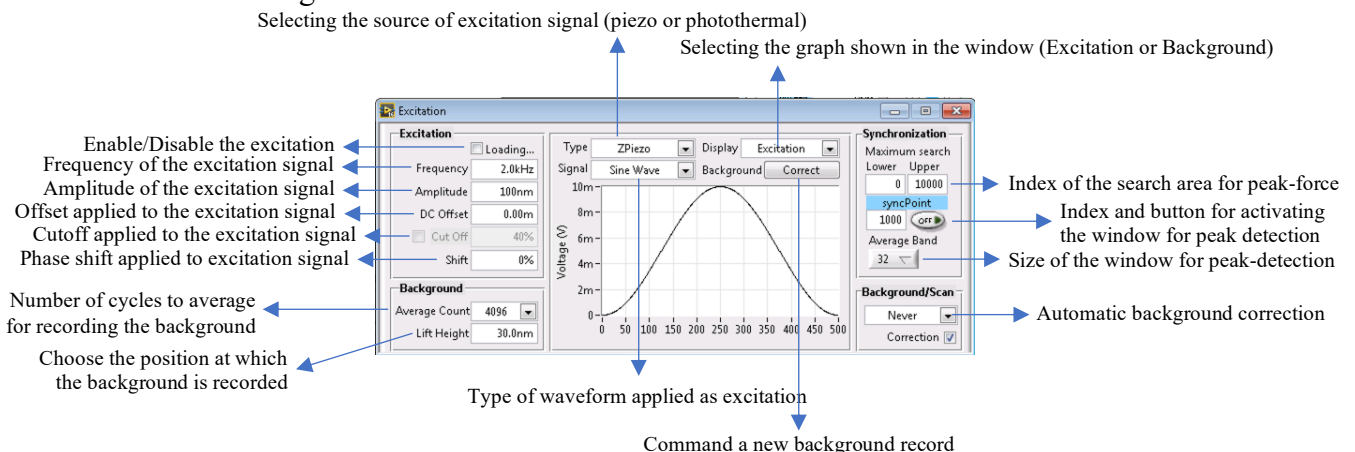


Figure 10 Details of the Excitation VI

At the top of the window there are options for selecting the type of the ORT (with **ZPiezo** or **Photothermal**) and the type of the excitation signal that is applied to the outputs (**Sine Wave**, **Triangle Wave**, **Square Wave**, **Sawtooth Wave**, **Heartbeat Signal**). The graph that is displayed is either the **Excitation** signal itself, or the latest recorded **Background**. Recording the background is also commanded with the background **Correct** button. When pressed, the z-piezo will be retracted equivalent to the value entered in **Lift Height** and several

cycles (equal to **Average Count**) of the force-curve are recorded. The average of these curves is then saved as the background.

The **Excitation** pane is used for tuning the parameters of the excitation signal. The first button is for enabling/disabling the signal. Note that when one of the signal parameters is changed, this enable button goes into the **Loading...** state until the new values of the signal are calculated and downloaded to the controller. The **Amplitude** can be selected in two modes; In case of using ORT with piezo-actuator, the amplitude is selected based on the range that the piezo is going to move (in nm or um) and in case of using photothermal excitation, the amplitude is selected based on the voltage applied at the output (mV or V), this is the peak-to-peak value of the voltage applied to the laser. Note that the signal is shifted such that the minimum value of the signal is always zero in this case, in other words, the signal always has a dc offset equal to half of the amplitude. **DC Offset** is selected similar to the **Amplitude** based on the excitation tool; It is the offset added to the excitation signal. The **Cutoff** is usually used in photothermal excitation end it keeps the high amplitude parts of the signal in order to prevent heating the cantilever. In other words, the cutoff turns the laser off when the cantilever is not trying to actively push on the surface. The **Shift** that is input as percentage is equivalent to the phase of the excitation signal. This phase shift doesn't change the signal at all, since the excitation signal is applied repeatedly. But it changes the way it is displayed. Corresponding to this change, the recorded force-curve is also shifted and it will appear as a shift on both excitation and background in the representation pane. The **Background** pane has options for the number of the force curves that are used to average and save as the background (**Average Count**). **Lift Height** represents the position of the tip where the background is recorded, it is equivalent to the height that the tip is retracted from the surface (latest feedback position) to record the background.

The **Synchronization** pane can be turned on or off with the corresponding button. In case it is turned off, the peak value is selected by searching over the whole recorded waveform. In the case of z-piezo it is turned off by default. In case it is turned on, the peak is selected by looking for the maximum value in the region defined by the **syncPoint** and the **Average Band**. **syncPoint** is the index of the synchronization position from the beginning of the excitation cycle, the index values are represented on the x-axis of the excitation/background graph. **Average Band** represents the number of points around the **syncPoint**. For more details about these two parameters, refer to **ORT Force Curves VI**.

The **Background/Scan** pane enables correcting the background automatically after each line or each frame.

3.6. ORT Force Curves (ORTForceCurves.vi)

This VI corresponds to the **Excitation** VI and it is mainly used to view the force curves (not all) and choose a sync point on the curve.

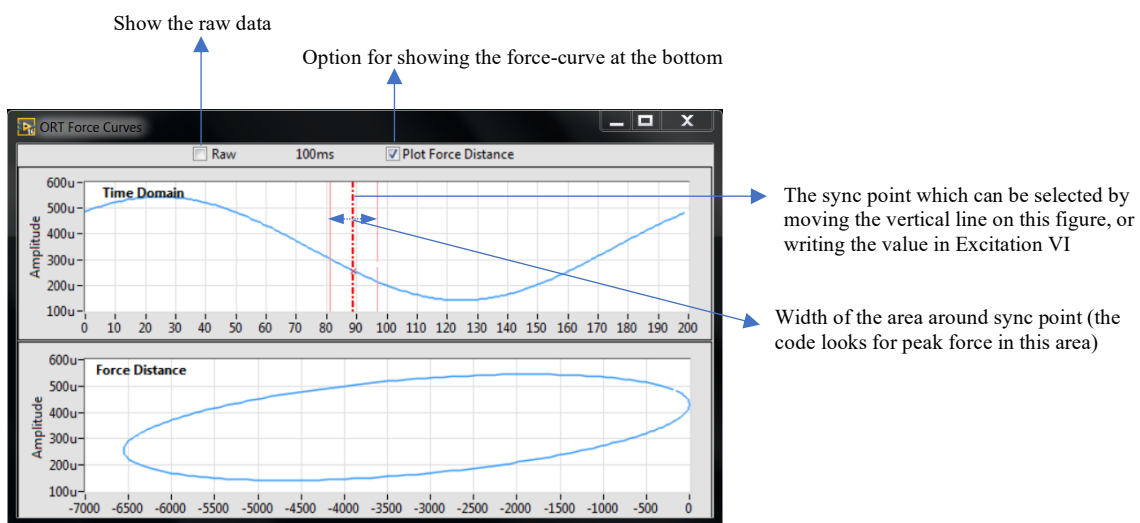


Figure 11 Details of the ORT Force Curves VI

The **syncPoint** and **Average Band** that were introduced in **Excitation** VI are represented here. They are two parameters that define the proximity of the peak-force. By moving the **syncPoint** and selecting an **Average Band** corresponding to its changes and width, the user tells the controller the approximate area of the maximum deflection.

3.7. Ramp (Ramp.vi)

This block is used to apply one or several ramp signals on the scanner (z-piezo) to record the force curve. Based on the recorded signals, one can calculate the deflection sensitivity.

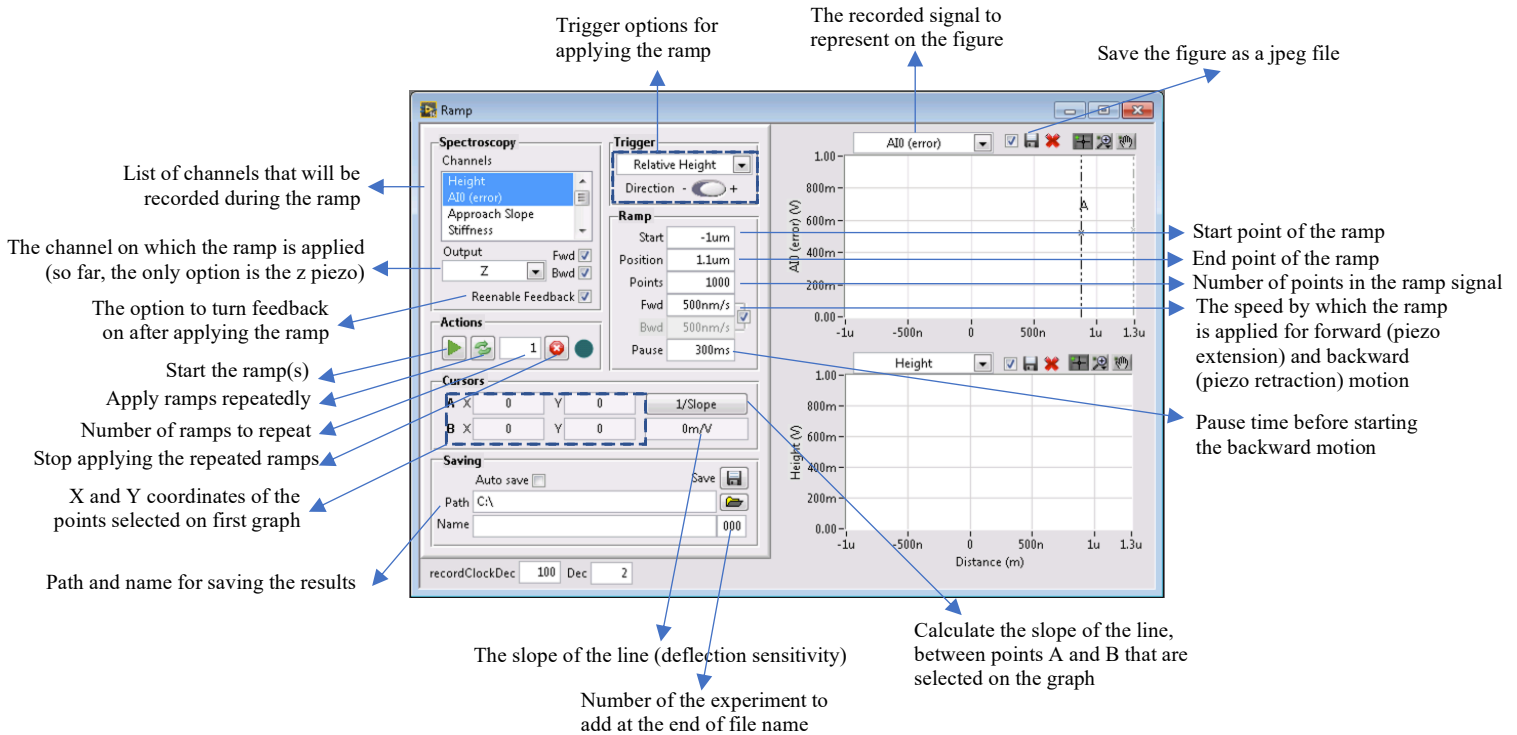


Figure 12 Details of the Ramp VI

The windows to the left show the options of the ramp signal.

Spectroscopy pane shows the channels that are recorded during the application of the ramp signal. By default, only height and deflection (vertical deflection) are recorded. In this pane, one can also select which parts of the signal are recorded, the forward motion (extension of the z-piezo) and/or the backward motion (retraction of the z-piezo).

The **Actions** pane includes buttons for commanding the start of the ramp. In case of several repeated ramps, their number can be input. There is also an option for starting a continuous repetition of ramps which has a corresponding stop button.

The **Trigger** pane shows options for the signal that is applied to the z-piezo. The options are: **Absolute Height**, **Relative Height**, **Process Variable**, **External Source**. The direction of the signal (inversion) can also be changed here.

The **Ramp** pane has values for the exact shape of the ramp function. **Start** and **Position** represent the start point and end point of the extension of the piezo. **Points** represents the number of points that are collected during ramp. **Fwd** and **Bwd**, define the speed by which the piezo is extended or retracted. **Pause** is an option for waiting before starting the backward motion.

The two figures to the right of the window, show the recorded signal. Blue graph shows the forward path and red graph shows the backward path. There are two vertical lines on the figure that represents the deflection versus z-piezo position. These two lines can be shifted by the user. The coordinates of the points where they intersect the forward and backward graphs, are shown in the **Cursors** pane.

The **Cursors** pane shows the coordinates of the points selected on deflection graph. Using these points, one can press the **1/Slope** button and get the deflection sensitivity.

Saving pane is used for saving the recorded force-curves' data.

3.8. Monitor

3.9. Analogs (Analog.vi)

This VI is used to invert any of the analog inputs/outputs. Some of the analog signals can be inverted from other parts of the software as well.

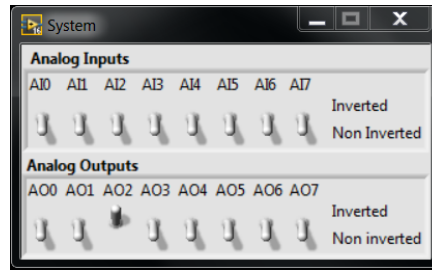


Figure 13 The Analogs VI

3.10. LA Controller

This VI is used to identify the dynamics of the components that are in the feedback loop and designs a high-order controller to achieve higher bandwidths. In the first step, a PRBS signal is applied to the piezo and the cantilever deflection is recorded. The frequency response of the system is then calculated based on the input-output mapping. The frequency response is then used to design a high-order controller with the desired order and bandwidth.

Annotations for the LA Automatic Feedback Filter Tuning window:

- Identification and filter design for axial or lateral direction of scan (points to Filtering mode)
- Output decimation (points to Output Decimation)
- Repetitions of one period of PRBS (points to Repetitions)
- Regulates the length of PRBS (points to PRBS Order)
- Lower limit of signal (points to low Value)
- Upper limit of signal (points to high Value)
- Computed based on PRBS order (points to PRBS length)
- Set based on ORT or sampling frequency in contact mode (points to Sampling Frequency)
- Save identification data in project folder (points to Save identification)
- Start identification in open-loop (points to Open-loop ID)
- Start identification in closed-loop (points to Closed-loop ID)
- The desired bandwidth for the final closed-loop system (points to Desired bandwidth)
- Order of high-order controller (points to Controller Order)
- Start controller design (points to Design Controller)
- Reset the filters that were set on feedback loop (points to Reset filters)
- Bypass the designed high-order filters (points to disable Filters)
- Equivalent gains of a PI extracted from filter (points to Kp and Ki)

In the **System Settings** pane, the user can select the direction of scan for which the filter is designed. It is either **Axial** (the z-direction) or **Lateral** (xy directions).

In **System ID Parameters** pane, the details of the identification signal are entered. The PRBs signal has a period equal to $2^{(\text{PRBS Order})} - 1$ and the **Repetitions** indicates how many times this signal is repeated. Therefore, the identification signal has a length of $\text{Repetitions} * [2^{(\text{PRBS Order})} - 1]$. This signal is binary and the upper and lower bounds are chosen with **high Value** and **low Value**. **PRBS length** is an indicator of the length of each period of the signal. The **Sampling Frequency** is also an indicator and specifies the frequency by which identification is performed. Its value comes from the ORT rate (in case of ORT/PORT) or the sampling frequency that is specified in **ZController VI** (in case of contact mode). If **Save identification** is ticked, the input-output data and frequency response will be saved in the project folder. **Open-loop ID** starts the identification process in which the input signal is applied to the piezo (while feedback is off) and the deflection is recorded.

The **Control Design** pane contains controls for design of the high-order filter. **Desired bandwidth** indicates the bandwidth of the final closed-loop system and **Controller Order** indicates its order. **Design Controller** initiates the design process and **Reset filters** removes the latest high-order filter that were applied on the feedback loop (in case there were any design before). Once the design is finished, the designed filters will be applied on the feedback loop, but of **disable Filters** is ticked, they will be bypassed. After the design, a PI (proportional + integral) controller is extracted from the high-order filter; the values corresponding to this PI controller are

User Manual: LBNI AFM Controller Software

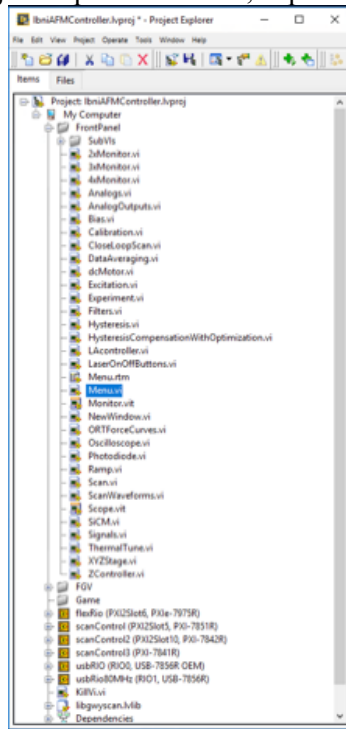
displayed in the fields [Kp](#) and [Ki](#). These values should then be copied to [ZController](#) VI in their corresponding fields.

Appendix A: List of Required LabVIEW Products:

- LabVIEW 32 bit
- LabVIEW 64 bit
- NI Signal Express
- VI Package Manager
- LabVIEW Control Design and Simulation Module
- LabVIEW Control Design and Simulation Module (64 bit)
- LabVIEW MathScript RT Module
- LabVIEW MathScript RT Module (64 bit)
- LabVIEW Digital Filter Design Toolkit
- LabVIEW Digital Filter Design Toolkit (64 bit)
- LabVIEW FPGA Module (32 bit)
- LabVIEW FPGA Module (64 bit)
- Xilinx Compilation Tools - ISE 14.7 for LabVIEW
- LabVIEW Advanced Signal Processing Toolkit
- LabVIEW Advanced Signal Processing Toolkit (64 bit)
- LabVIEW Unit Test Framework Toolkit
- LabVIEW Unit Test Framework Toolkit (64 bit)
- LabVIEW FPGA Compile Fram Server
- NI System Configuration
- NI -488
- NI DAQmx
- NI VISA
- NI PXI Platform Services
- NI R Series Multifunction RIO Device Drivers
- FlexRIO
- NI Compact RIO Device Drivers

Appendix B: Configuring the address of the bit-files for the FPGA module

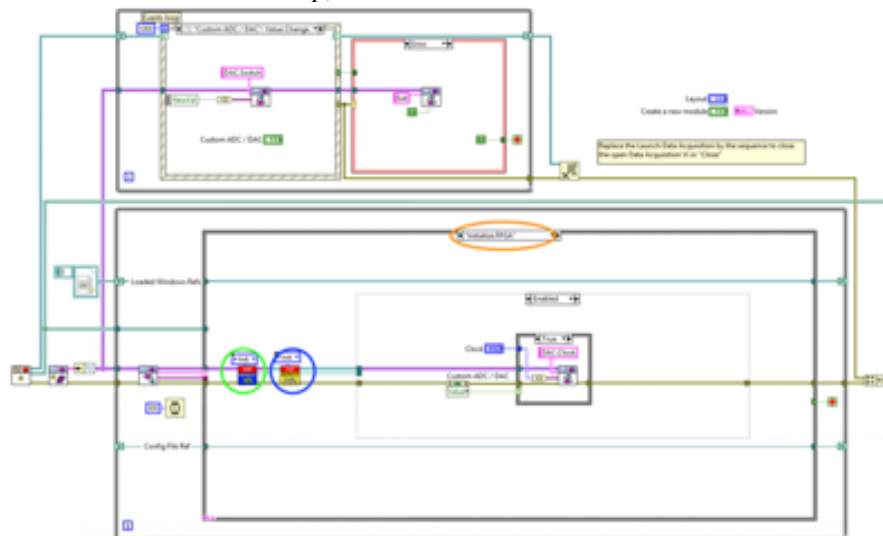
Open the LabVIEW project file. On the project explorer window, expand Front Panel and open Menu.vi



Open the Block Diagram of this VI (Menu.vi).

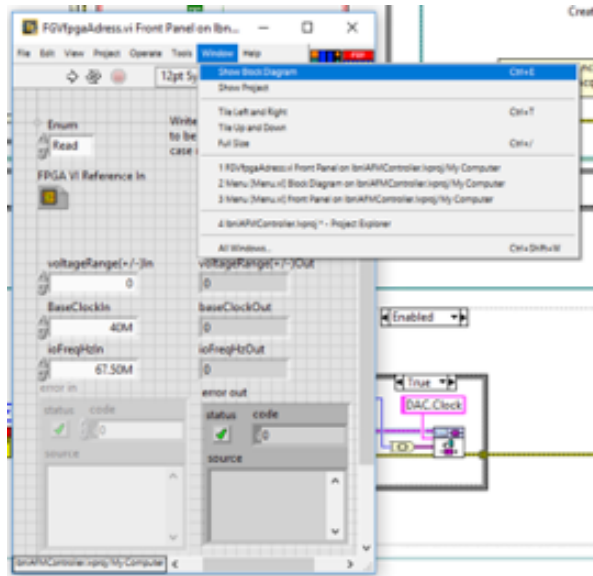


On the block diagram, in the “Initialize.FPGA” loop, there are two blocks which address the bit-files.

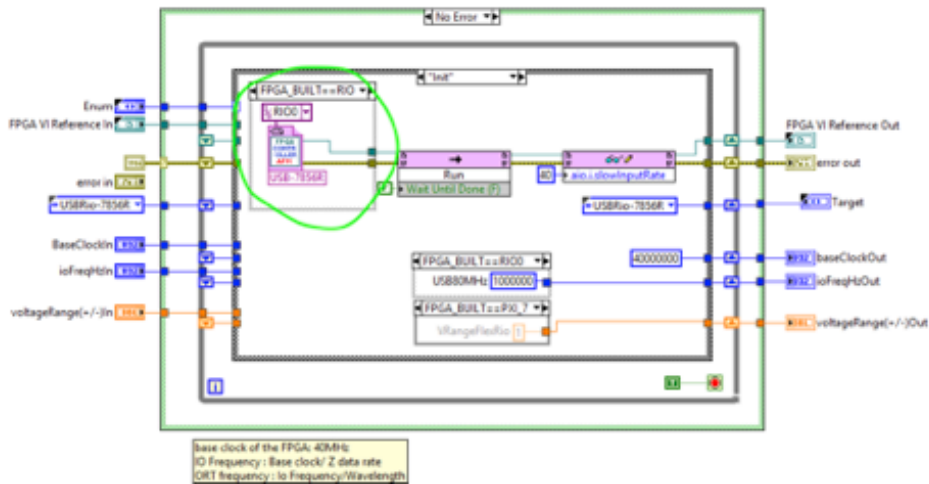


Open the first VI shown in the figure (green circle) and go to its Block Diagram.

User Manual: LBNI AFM Controller Software



Double-click on the block shown in the next image:

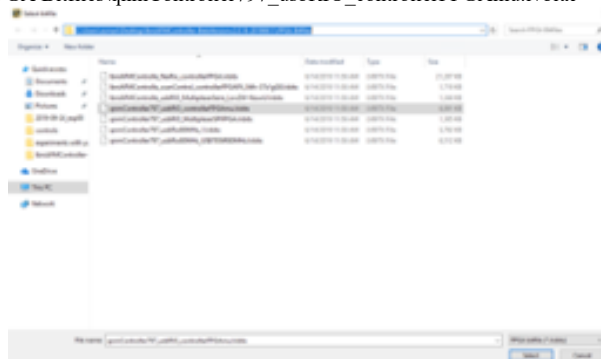


In the opened window, you have to address the bit-file correctly.



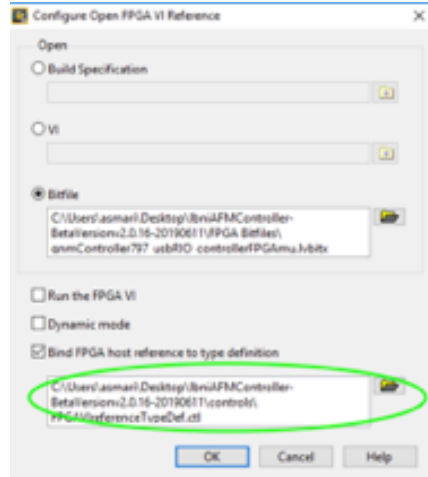
Browse for the bit-file and address it to:

[The Directory of your LabVIEW Project]\FPGA Bitfiles\qnmController797_usbRIO_controllerFPGAu.lvbitx

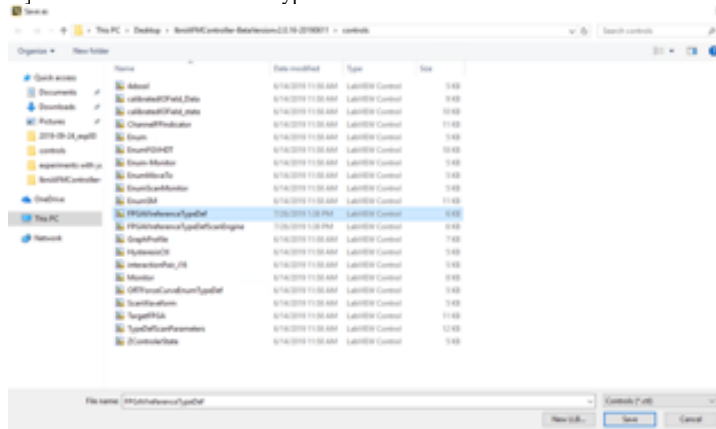


Now, address the type definition from the same window:

User Manual: LBNI AFM Controller Software

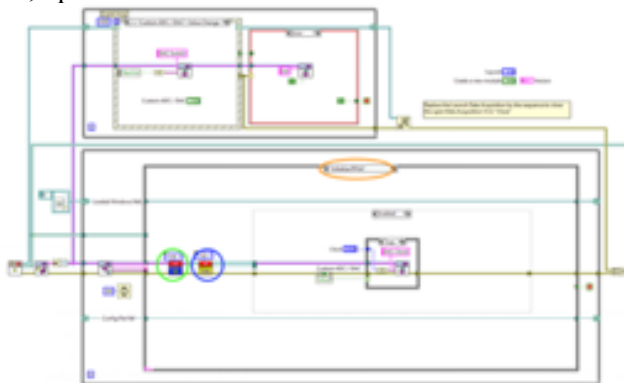


Again, browse for the Type Definition and select it from:
[The Directory of your LabVIEW Project]\controls\FPGA\IreferenceTypeDef.cti

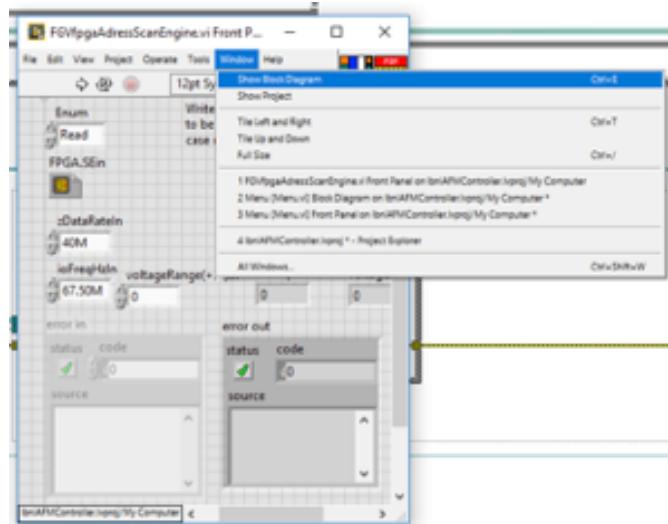


Now close the windows and save the changes. Don't close the Menu.vi because you have to apply these changes on another VI as well.

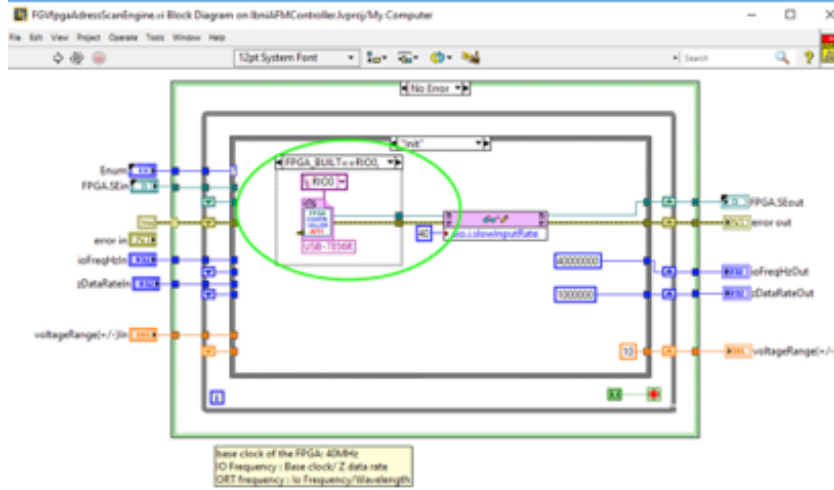
In the Block Diagram of the Menu.vi, open the VI annotated with blue circle in the next image:



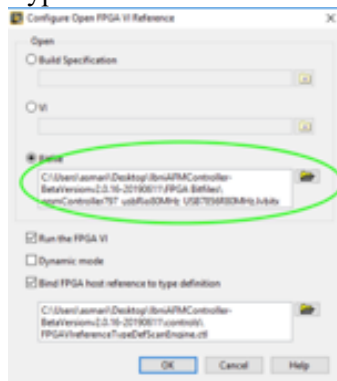
Open its Block Diagram:



User Manual: LBNI AFM Controller Software
 And again, double click on bit-file addressing block:

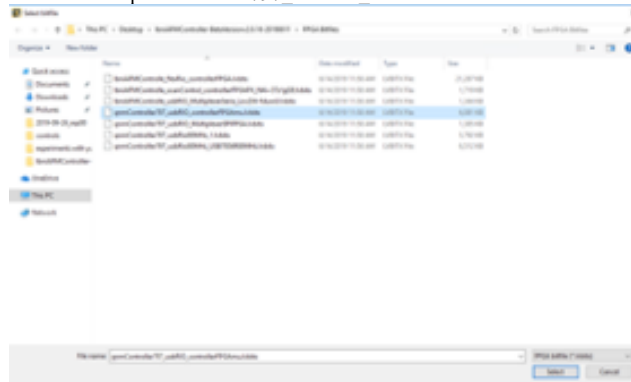


You will have the window for the bit-file and type-def address.

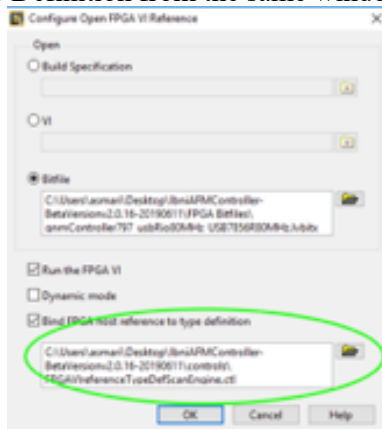


Browse for the bit-file and address it to:

[The Directory of your LabVIEW Project]\FPGA Bitfiles\qnmController797_usbRIO_controllerFPGAmu.lvbitx



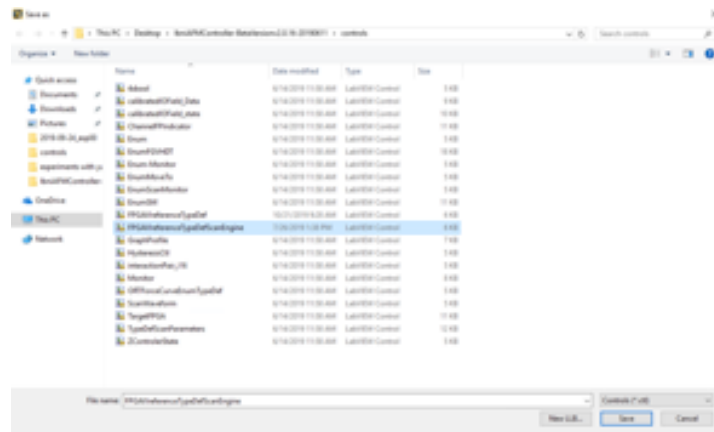
Select and choose the address for the Type Definition from the same window:



Address it to:

[The Directory of your LabVIEW Project]\controls\FPGA\IreferenceTypeDefScanEngine.cti

User Manual: LBNI AFM Controller Software



Close the windows and save the changes if you were prompted to save.