Guide to the LabVIEW code

# Interface

# Manual setup

A screenshot of a computer program

Description automatically generated

This is where you can control the stage and do a soft homing of the stage.

It is also the place where you can look at the voice coil in a live mode.

# Acquisition

A screenshot of a computer

Description automatically generated

There are four while loops running

The first makes sure to establish connection to the different devices (currently only the stage)

Then it has an event listener. The event listener should not execute long codes but instead send messages to the other loops to execute a given action. This way the code is still listening to you cancelling the operation even though it is not finished executing an operation.

Event listener loop.

Acquisition handler

Live mode signal generation loop (DAQ que)

Command sequence handler loop

Function generation

Synchronization of camera trigger, voice coil output, AOTF lines, Blanking and

* Global flags.

Stage position and position log

Stage calibration - Homing

Laser line output selector

Update boundaries – using the script or manually

# Live mode for calibrating the voice coil and aligning

Sawtooth mode

Manual voltage output

From file

This is synchronized with a camera trigger

# The structure of the acquisition code

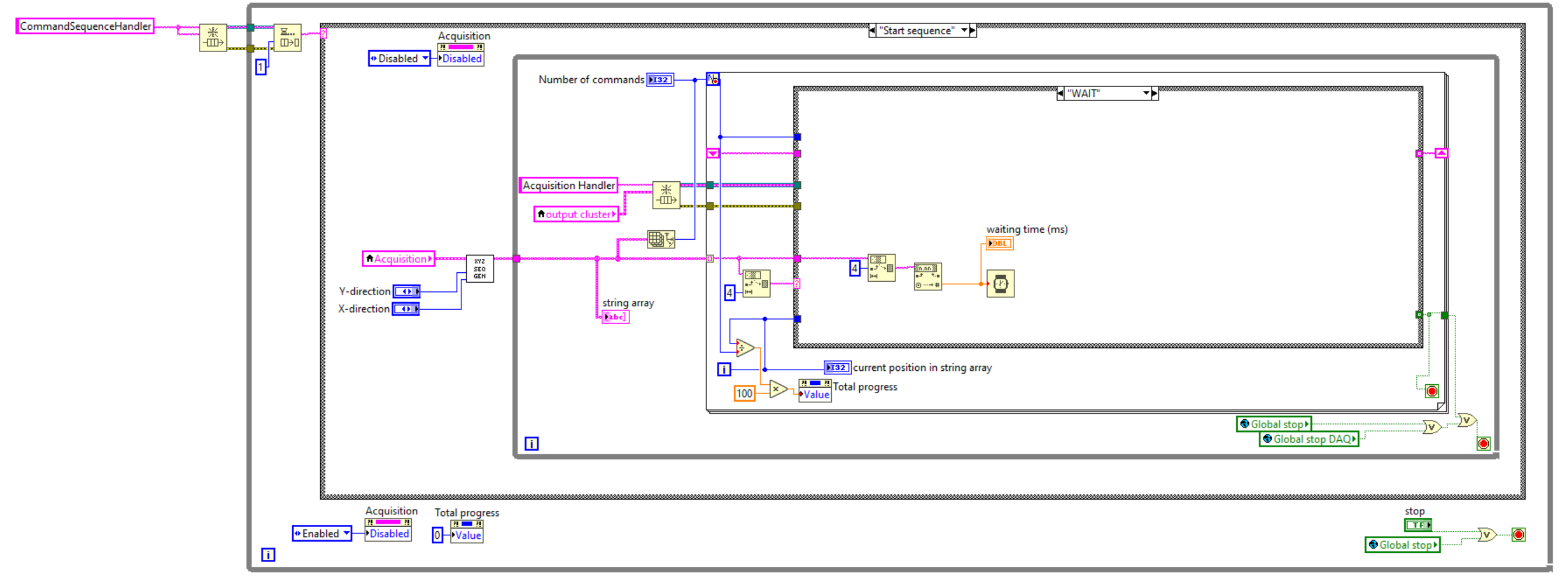
When the acquire button is clicked. The event listener loop sends a message to the command sequence handler to make an acquisition.

A screenshot of a computer

Description automatically generated

This is done by adding a string with “Start sequence”.

The command sequence loop continuously reads the queue and if it contains a message saying: “Start sequence” then it will read the values of the acquisition panel like number of XYZ tiles and how long the time is between XY-tiles. This information is used to generate a list of commands using the function XYZ\_sequence\_generator with flags to generate a list of strings to execute.



A diagram of a diagram

Description automatically generated

The sequence generator will create a list of string like this when the XYZ parameters are 2by2by10

A screenshot of a computer

Description automatically generated

The command sequence handler loop then goes through the list of strings.

The acquire Zstack message sends a message to the acquisition loop to acquire a Zstack.

A diagram of a computer

Description automatically generated

The acquisition loop now resets some global flags and reads some of the acquisition parameters and activate the signal generation function. The DAC OUTPUT CAM TRIG function synchronizes the voice coil signal generation with the camera trigger, the AOTF lines and the blanking.

A screenshot of a computer

Description automatically generated

The camera trigger will also trigger a global variable called “Global Counter count”. This global value is used to control when the stage is moved. If three lasers are on there should be three image acquisitions before the stage should be moved. The stage will move in the flyback time (while not imaging). The position is read and stored in a position log for each image acquisition. Once the Z-stack acquisition is finished it will change the global flag “zstack acquisition finished flag” to high.

The command sequence handler loop listens for the “zstack acquisition finished flag” to turn on before it will proceed to the next element in the string array list. In the list it will have a command saying MOVEZ Back which will send the stage to the initial Z-stack position and raise the flag “Global destination achieved” so the command sequence handler can proceed to the next element in the list. The next command string will be MOVEX+1 which will move to stage one step size in x. Plus or minus would indicate the direction of the movement, while X or Y would indicate the axis of movement. The next command would be to wait the time specified between tiles. The waiting time is there to make sure that it has settled. Now the setup would acquire a new Z-stack and repeat the process until the desired volume has been acquired.

Once the command sequence handler has reached the end element, it will be listening for new acquisitions again in the queue.

An acquisition can be cancelled by pressing abort acquisition and confirming the cancelation of the acquisition. This will set the Global stop DAQ flag high which will stop the acquisition and make the command sequence handler loop ready for a new acquisition.

# Laser line selection

The laser lines can be selected from the panel

A screenshot of a computer program

Description automatically generated

Before acquisition one laser has to be number 1. The panel offers flexibility in the order of the acquisition and will keep the order throughout the volume acquisition. If you are using two lasers they should follow one should be number 1 and the other should be number 2. You cannot let one be number 1 and the other be number 3 and they can also not both be 1.

# DAQ signal generation

A computer screen shot of a computer

Description automatically generated

The DAQ signal generation loop is used to operate the live mode of the microscope. This is the part that can be used the send a fixed voltage level to the voice coil or generate a sawtooth function with a specific frequency or generate a signal that is read from a text file.

This part is activated when these controls are changed:

A screenshot of a computer

Description automatically generated

# Signal generation

A diagram of a computer

Description automatically generated

7 different signals are generated in this part:

* Camera trigger – digital clock signal used to trigger the camera and synchronize the other signals
* Voice coil signal – Analog Output generated from a text file
* AOTF laser lines – Digital Output each laser line (total of 4) has a line that is high when the laser should be on
* Blanking channel – Digital clock used to globally turn off the AOTF output during flyback to avoid unnecessary bleaching.

The signal outputs are updated with a 10kHz rate. The camera trigger is set to run for a finite sample that is equal to the number of active laser times the number of z-positions.

The DAQ outputs are specified using the DAQ channel setup vi and selecting microscope or test setup depending on the DAQ card used. If you want to update the Output pins you have to change it in the DAQ channel setup vi for the setup you want to use and then press Make current value default.

A screenshot of a computer

Description automatically generated

# Voice coil calibration optimization scheme

A diagram of a computer

Description automatically generated

The optimization algorithm has a few layers.

The overall optimization control keeps track of the parameters to put into the error calculating function. In other words it organizes the optimization walk in parameter space based on the error from each set of parameters.

The overall optimization algorithm calls the error calculating function. The error calculating function calls a signal generation function. The error calculating function will generate a signal corresponding to the parameters send from the overall optimization algorithm.

The plan is to make the error calculating function trigger the LabVIEW camera acquisition program to acquire the two frames. The first frame should be discarded and the second one should be processed to evaluate the uniformity of the beam profile. The error should then be returned to the error calculating function and the error will be passed on to the overall optimization algorithm, such that each set of parameters have an error value. The overall optimization updates the parameters to lower the error value.

A computer screen shot of a computer

Description automatically generated

Instead of the current random number generation, the error should be calculated from the image data.

Pitfalls: ensure to set the camera to be ready to acquire before sending the voltage signals to the Voice coil. One has to consider the timing of setting up the camera. I imagine it being solved using a global variable.

It could be implemented somewhat along the lines of In the error calculation function the global flag is pulled high to signal the camera to be ready to listen for camera triggers with the current acquisition parameters. The camera script should listen for the global flag being pulled HIGH. When it is HIGH it should make the camera ready to acquire and then pull the flag low. Meanwhile the error calculation function listens for the flag being LOW and when it is pulled low it should start the DAQ task. Now the images should be acquired and evaluated, and the error should be sent back to the error calculation function to update the error for this set of parameters.