# Acquisition Code

This directory contains the code required to run the SAMURAI system. This currently requires a direct path and connection to the U:/ network drive. This code has been tested lately with python 3.6 but should be backward compatable with 2.7.

## dependencies

* Anaconda with python 3.X
* pyvisa
* cannot remember the last one

### Installing Anaconda and Dependencies

* Install Anaconda
  1. Go to the anaconda download page [here](https://www.anaconda.com/distribution/#download-section)
  2. Select the correct operating system and installer
  3. Download and run the executable and follow the installer to install Anaconda 3.X
* Install pyvisa
  1. Type into the windows search ‘anaconda prompt’ and run the program
  2. In the terminal run conda install -c conda-forge pyvisa to install pyvisa

## Subdirectories

### /support

This subdirectory contains multiple classes and functions to run the SAMURAI system including control for the VNA and Meca500 Robot arm. This contains the following files

* /support/autoPNAGrabber.py
  + python control of PNAGrabber. Requires PNAGrabber executable to be run from command line with -r flag to ‘press’ the ‘meas all’ button without opening the UI
* /support/Meca500.py
  + Class for controlling Meca500 Robot
* /support/metaFileController.py
  + Class for interacting with already written metafile from a measurement
* /support/pnaController.py
  + Class to interact with PNA to get various settings and set some things on it
* /support/samurai\_metaFile.py
  + Class for creating metafile for SAMURAI measurements
* /support/samurai\_support.py
  + Some support functions to use. (mainly functions used when USC came out in Aug. 2018 for phased array calibration)

# SAMURAI Hardware Information

graph TD;  
 A-->B;  
 A-->C;  
 B-->D;  
 C-->D;

(PICTURE OF SETUP)

## Hardware

### Meca 500 6 axis positioner

Small 6 axis positioner.

* [Website](https://www.mecademic.com/products/Meca500-small-robot-arm)
* [User Guide](hardware/datasheets/Meca500-R3-User-Manual.pdf)
* [Programming Manual](hardware/datasheets/Meca500-R3-Programming-Manual.pdf)

### Keysight PNA-X (N5245A)

10MHz to 50GHz VNA. Ports are 2.4mm Male typically with 2.4mm F-F connector savers on them.

* [Datasheet](hardware/datasheets/N5245.pdf)

### Antennas

* Sage-millimeter 17dBi WR-28 Horn antenna
  + [Datasheet](hardware/datasheets/17dBi_horn_sagemillimeter.pdf)
* Sage-millimeter 23dBi WR-28 Horn antenna
  + [Datasheet](hardware/datasheets/23dBi_horn_sagemillimeter.pdf)

### Cables

* Junkosha 2.4mm (M-M) 3m Cables
  + info (link datasheet)
* Junkosha 2.4mm (M-M) 0.25m Cables
  + info (link datasheet)

### Adapters

* Sage-millimeter 2.4mm to WR-28 Adapters

## Networking

Currently, the samurai system is run over a custom local network run through a simple network switch. This connects to the VNA, Meca500 Robot arm, and eventually cameras.

### Remote PNA-X control

-Info on Remote KVM setup

### IP and VISA Addresses

* PNA-X = [10.0.0.2](http://10.0.0.2)
  + VISA Address = ‘TCPIP0::10.0.0.2::inst0::INSTR’
* Meca500 = [10.0.0.5](http://10.0.0.5)
  + VISA Address = Could not get VISA to work correctly! Connect using sockets.

# Running the SAMURAI Software

This section covers the steps required to run a SAMURAI measurement

## Running from python command line interface (CLI)

*[CLI]: Command Line Interface*  
[IDE]: Integrated Development Environment (e.g. Spyder)  
The following steps are to run a SAMURAI measurement from the python CLI. The steps using the python CLI here are valid for the integrated command line within the Spyder IDE. While these steps will be similar using a basic python setup, the importing of the SAMURAI classes and libraries may be a bit more complex.

### 1. Create a new SAMURAI measurement directory

1. Make a copy of meas\_template in the directory U:\67Internal\DivisionProjects\Channel Model Uncertainty\Measurements\Synthetic\_Aperture
2. Rename the copy to the current date in the format mm-dd-yyyy
   * From here on, this newly created directory will be referred to as <working-directory>
3. Copy and paste the correct comma separated value (CSV) file containing the positions into <working-directory>/synthetic\_aperture/raw
   * Some commonly used templates are contained in <working-directory>/synthetic\_aperture/raw/position\_templates directory.
   * Once the desired CSV file has been copied, rename it positions.csv

### 2. Perform 2 Port VNA Calibration

1. In the windows file explorer navigate to <working-directory>/cal/calibration\_pre
2. double click on ‘cal.pnagrabber’ to start PNAGrabber for the calibration.
3. Attach each of the standards to the calibration plane with the naming convention <standard-port-1>\_<standard-port-2>.s2p
   * (e.g. load\_short.s2p is load on port 1 and short on port 2)
4. When the calibration is completed, make a copy of each of the .s2p files generated and put them into the <working-directory>/cal/calibration\_pre/raw folder

### 3. Import the SAMURAI\_System Module

1. Open the python CLI (e.g. the command window in Spyder)
2. With the file opened in the Spyder IDE, click the green play button on the top toolbar OR type the code below where <dir-of-code> is the directory where SAMURAI\_System.py is located on the system.

runfile('<dir-of-code>/SAMURAI\_System.py')

### 4. Create a SAMURAI\_System Object

1. With the SAMURAI\_System module imported, create a SAMURAI\_System object by typing mysam = SAMURAI\_System() into the CLI.

### 5. Change directory to measurement directory

1. Change the directory to <working-directory>/synthetic\_aperture/raw by running the following set of commands:

* import os  
  os.chdir(<working-directory>/synthetic\_aperture/raw)
* OR in certain iPython CLIs
* cd <working-directory>/synthetic\_aperture/raw

### 6. Mount the Antennas

1. Mount the Tx Antenna (usually port 2) to the fixed holder
2. Move the SAMURAI Robot to the mountain position using the commands below
   * The mysam object must exist for this step to work
   * Keep in mind, after this code the positioner is still connected and activated after these commands

* mysam.connect\_rx\_positioner() #connect and home the positioner  
  mysam.move\_to\_mounting\_position() #move to an easy position to mount the antenna

1. Use the four m3 screws to attach the Antenna to the Meca500

### 8. Open the Robot’s Web interface (Optional)

Before running the sweep we can perform the extra step of viewing the robot’s movement and status through its web monitoring interface. To open up the web monitoring interface:

1. Open a web browser (tested in chrome)
2. type [10.0.0.5](http://10.0.0.5) into the address bar
3. In the web interface, click the ‘Connection’ button on the top toolbar.
4. In the pop-up window select ‘Monitoring’ and click ‘Connect’

### 8. Run the Synthetic Aperture Sweep

Now we can begin the sweep

1. Ensure the working directory is set to <working-directory>/synthetic\_aperture/raw (see step 5)
   * Some editors/IDE’s (e.g. spyder) show this in a top bar of the screen
   * The current directory can be found from a python CLI by typing import os; os.getcwd()
2. Type the following code and hit enter to begin the sweep
   * This step assumes the robot has previously been connected and initialized (activated and homed)
   * This also assumes the mysam object has already been created

* mysam.csv\_sweep('./','./positions.csv',template\_path='template.pnagrabber');disconnect\_rx\_positioner()
  + NOTE: If a csv file is being tested, the flag run\_vna=False can be added to the mysam.csv\_sweep() call to prevent the VNA from running
  + NOTE: The robot can also be put into simulation mode where all commands are sent and the web interface shows the robot moving, but the robot does not physically move. For more information on this reference the code documentation.

### 9. Unmount the Antennas

1. Create mysam object if it does not exist
2. Connect to positioner (refer to ‘Mount the Antennas’ section)

### 10. Collect and Save data

1. copy data from <working-directory>/synthetic\_aperture/raw to <working-directory>/synthetic\_aperture/
2. Perform post-calibration in <working-directory>/cal/calibration\_post (refer to ‘Perform 2 Port VNA Calibration’ section)

## Running from the Graphical User Interface (GUI) SAMURGUI

This code needs to be finished

# Measurement TODO List

* ☒ Line of Sight Measurements
* ☒ Cylinder Measurement
* ☒ Cylinder non-LOS measurement

# Code Editing TODO List

* ☐ allow user to move\_to\_mounting\_position from SAMURGUI
* ☐ allow user to run csv\_sweep without connect/disconnect from SAMURGUI
* ☐ add metafile editing interface to SAMURGUI
* ☐ add Meca500 status viewer in SAMURGUI