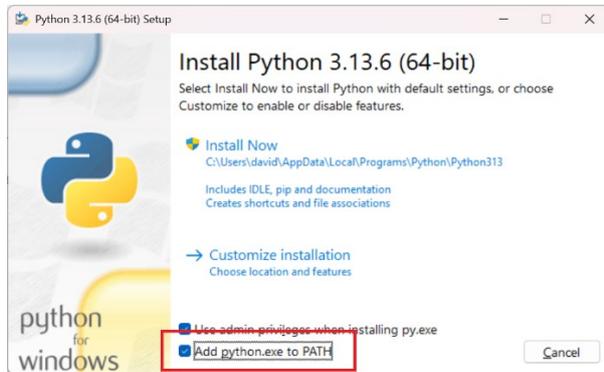


To use, please install latest Python from the official website

(<https://www.python.org/downloads/windows/>)

Please make sure the Python install folder is on the system path(*). The imslib.pyd library will need to be able to find python3.dll (and python3x.dll, currently python313.dll).

* Check box as shown



It is always a good idea to create a virtual environment.

In a Command Prompt navigate to the folder where you extracted the zipfile contents and type the following:

```
> python -m venv env
> .\env\Scripts\activate.bat
```

e.g.

```
C:\Users\name>E:
```

```
E:\> cd \Dropbox\iMS-Python\imslib_py_20250807
E:\Dropbox\iMS-Python\imslib_py_20250807>python -m venv env
E:\Dropbox\iMS-Python\imslib_py_20250807>.\env\Scripts\activate.bat
(env) E:\Dropbox\iMS-Python\imslib_py_20250807> pip install matplotlib
```

You should see the prompt now starts with (env).

Some of the examples require an external library called matplotlib which can be installed with **pip install matplotlib** (as shown above)

Now you can try running the examples e.g. Test 04 (does not iMS4 connected), with the following command format-

```
>python .\test04_types.py
```

Tip: hit tab after entering *test04* to auto complete. (applies to all tests).

```
(env) E:\Dropbox\iMS-Python\imslib_py_20250807>python .\test04_types.py
```

If it fails with the following error:

```
ImportError: DLL load failed while importing _imslib: The specified module could not be found.
```

it cannot find python3.dll. Copy in python3.dll and python313.dll (dependent on version number) from your Python install folder. Note that the python313.dll must match the version of python3.exe that you are invoking when you run the script - watch out for this if there are multiple Python versions installed.

If it runs, the output is

Test 04: iMS Defined Types

Using iMS Library version 1.9.1

Let's test some useful basic types

f is 1500.0Hz

f is now 2500.0Hz

a is 75%

a is now 60%

But it cannot be 125: 100%

Or negative: 0%

p is 90deg

p is now 30deg

You can go around more than once: 420deg

Or go backwards: -90deg

Now let's try some types derived from iMS::Frequency

f is also 2.5kHz

and 0.0025MHz

I can also see that k is 0.0025MHz

If I want to go really fast, I can try 5.0MHz

But beware. In python, assignment just rebinds the object so this is not the same in Hertz: 5.0MHz

Instead, I must use the assign operator: 5000000.0Hz

I can iterate using mixed types, so let's start at 75.0kHz

iteration 1: 75100.0Hz / 75.1kHz / 0.0751MHz

iteration 2: 75200.0Hz / 75.2kHz / 0.0752MHz

iteration 3: 75300.0Hz / 75.3kHz / 0.0753MHz

iteration 4: 75400.0Hz / 75.4kHz / 0.0754MHz

iteration 5: 75500.0Hz / 75.5kHz / 0.0755MHz

iteration 6: 75600.0Hz / 75.6kHz / 0.0756MHz

iteration 7: 75700.0Hz / 75.7kHz / 0.0757MHz

iteration 8: 75800.0Hz / 75.8kHz / 0.0758MHz

iteration 9: 75900.0Hz / 75.9kHz / 0.0759MHz

iteration 10: 76000.0Hz / 76.0kHz / 0.076MHz

Or 1.5MHz

iteration 1: 1.45MHz

iteration 2: 1.4MHz

iteration 3: 1.35MHz

iteration 4: 1.3MHz

iteration 5: 1.25MHz

(+ more...)

For subsequent use, it is only necessary to open a CMD window and Activate virtual environment e.g.

E:\>cd \Dropbox\iMS-Python\imslib_py_20250807

E:\Dropbox\iMS-Python\imslib_py_20250807>.\env\Scripts\activate.bat

(env) E:\Dropbox\iMS-Python\imslib_py_20250807>

iMS Python Library Release Notes:

With iMS4 connected USB.....
Test01 checks for connection

```
(env) E:\Dropbox\iMS-Python\imslib_py_20250807>python .\test01_scan.py
```

```
Test 01: Scan for iMS Systems  
Using iMS Library version 1.9.1  
Scanning for iMS Systems . . .  
iMS System 1: iMSP2404:USB
```

```
Found Synthesiser: iMS4d : iMS-4d 10.0-225MHz 4-Channel RF Synthesiser  
-> FW Version: 4.1.154 Mon Dec 18 2024 10:07 GMT Standard Time  
-> # Channels = 4 Freq Range = 10.17-228.92 MHz
```

```
Found Controller: iMSP : iMS-Pro Controller  
-> FW Version: 2.5.111 Mon Jan 17 2025 10:34 GMT Standard Time  
-> Max Image Size = 18000000 Max Image Rate = 5000000.0
```

GIT HUB

Update from Git hub:

Go to: <https://github.com/Isomet-Corporation>

Isomet Corporation
Founded in 1956 in Manassas, VA, Isomet Corporation develops acousto-optic devices, RF electronics, and optical subsystems for research and OEM a
<https://isomet.com/>

Overview Repositories 2 Projects Packages People

Popular repositories

- ims-lib** (Public) C
- imslib-python** (Public) C++
Builds a Python API for accessing Isomet IMS devices

People
This organization must be a member organization.

Top languages
C++ C

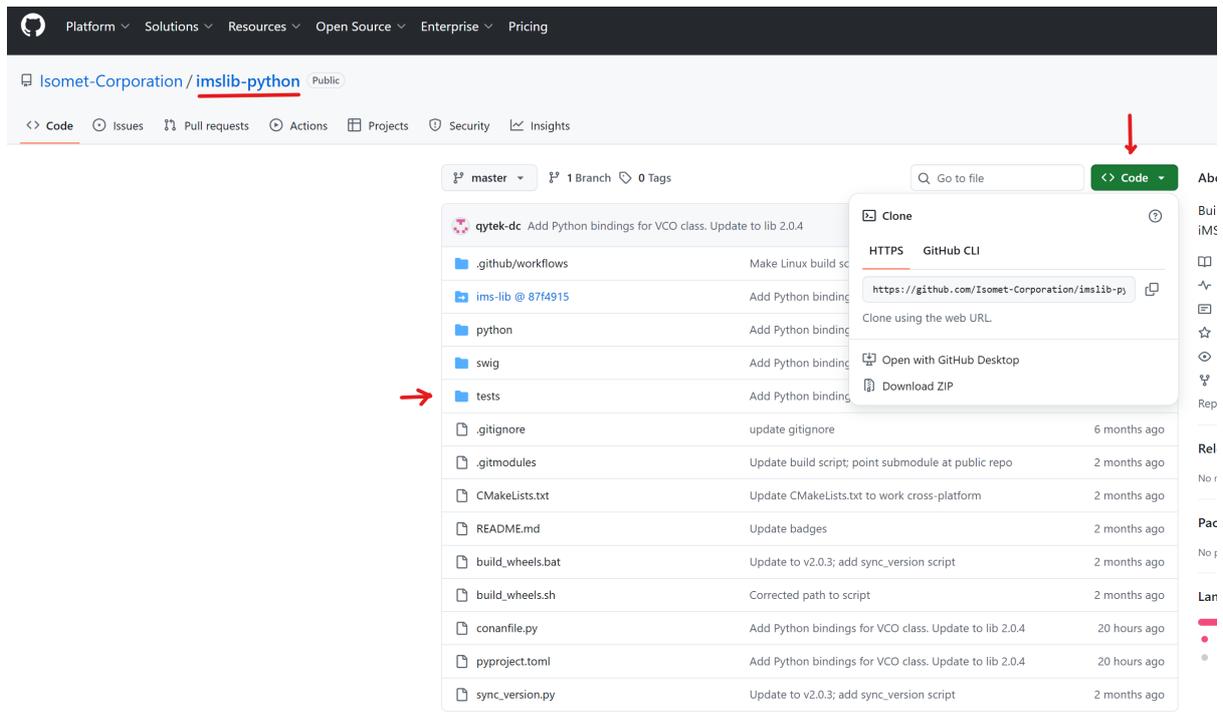
Repositories
Find a repository... Type Language Sort

Select imslib-python

Download options:

- For a complete download of repository (Select imslib-python)

Hit [`<>` Code] button, to reveal download options



The screenshot shows the GitHub repository page for `Isomet-Corporation/imslib-python`. The `Code` button is highlighted with a red arrow. A dropdown menu is open, showing options to clone the repository using HTTPS or GitHub CLI, or to download a ZIP file. The `tests` folder is also highlighted with a red arrow.

File/Folder	Description	Last Updated
<code>.github/workflows</code>	Make Linux build script	
<code>ims-lib @ 87f4915</code>	Add Python bindings for VCO class. Update to lib 2.0.4	
<code>python</code>	Add Python bindings for VCO class. Update to lib 2.0.4	
<code>swig</code>	Add Python bindings for VCO class. Update to lib 2.0.4	
<code>tests</code>	Add Python bindings for VCO class. Update to lib 2.0.4	
<code>.gitignore</code>	update gitignore	6 months ago
<code>.gitmodules</code>	Update build script; point submodule at public repo	2 months ago
<code>CMakeLists.txt</code>	Update CMakeLists.txt to work cross-platform	2 months ago
<code>README.md</code>	Update badges	2 months ago
<code>build_wheels.bat</code>	Update to v2.0.3; add sync_version script	2 months ago
<code>build_wheels.sh</code>	Corrected path to script	2 months ago
<code>conanfile.py</code>	Add Python bindings for VCO class. Update to lib 2.0.4	20 hours ago
<code>pyproject.toml</code>	Add Python bindings for VCO class. Update to lib 2.0.4	20 hours ago
<code>sync_version.py</code>	Update to v2.0.3; add sync_version script	2 months ago

For single files e.g. in 'test' folder

Click on 'test' to open folder

Double click on the required file to see code and download

