# Install Python on Windows

<https://www.python.org/downloads/>

Download latest version for windows (3.8.0 as of Nov 4 2019)

Run the installer

* Select “Add Python 3.8 to PATH”
* Click Install Now

A little more detail is here: <https://opentechschool.github.io/python-beginners/en/getting_started.html>

# Start a command prompt (CMD.EXE)

Start -> type “CMD” in the search box and run CMD.EXE

You should see a prompt, run “Python –V” and ensure you are running the expected version

Microsoft Windows [Version 10.0.18362.449]

(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Lawrence>Python -V

Python 3.8.0

C:\Users\Lawrence>

# Install a Code editor

There are many choices, pick one. My favourite is vim ( <https://www.vim.org/download.php> ). Other suggestions are:

* [Atom](https://atom.io/): A new code editor available for Windows, Mac and Linux. It’s an open-source project developed by GitHub and is very easy to add functionality for, with its packages system.
* [Sublime Text](http://www.sublimetext.com/): A great all around editor that’s easy to use. It’s Ctl+B shortcut lets you run the python file you’re working on straight away. Runs on Windows, Mac and Linux.
* [Geany](http://www.geany.org/): A simple editor that doesn’t aim to be too complicated. Available on Windows and Linux (you can probably find it in your package manager).
* [TextMate](http://macromates.com/): One of the most famous code editors for Mac, it used to be a paid product but has since been open-sourced.
* [Gedit](https://projects.gnome.org/gedit/) and [Kate](http://kate-editor.org/): if you run Linux using Gnome or KDE respectively, you might already have one of these two installed!
* [Komodo Edit](http://www.activestate.com/komodo-edit): a sleak, free editor for Mac, Windows and Linux, based on the more powerful Komodo IDE.

# Install Python Packages

Information here: <https://packaging.python.org/tutorials/installing-packages/>

C:\Users\Lawrence>pip --version

pip 19.2.3 from c:\users\lawrence\appdata\local\programs\python\python38-32\lib\site-packages\pip (python 3.8)

C:\Users\Lawrence>python -m pip install --upgrade pip setuptools wheel

Collecting pip

Downloading https://files.pythonhosted.org/packages/00/b6/9cfa56b4081ad13874b0c6f96af8ce16cfbc1cb06bedf8e9164ce5551ec1/pip-19.3.1-py2.py3-none-any.whl (1.4MB)

|████████████████████████████████| 1.4MB 1.6MB/s

Collecting setuptools

Downloading https://files.pythonhosted.org/packages/d9/de/554b6310ac87c5b921bc45634b07b11394fe63bc4cb5176f5240addf18ab/setuptools-41.6.0-py2.py3-none-any.whl (582kB)

|████████████████████████████████| 583kB 6.4MB/s

Collecting wheel

Downloading https://files.pythonhosted.org/packages/00/83/b4a77d044e78ad1a45610eb88f745be2fd2c6d658f9798a15e384b7d57c9/wheel-0.33.6-py2.py3-none-any.whl

Installing collected packages: pip, setuptools, wheel

Found existing installation: pip 19.2.3

Uninstalling pip-19.2.3:

Successfully uninstalled pip-19.2.3

Found existing installation: setuptools 41.2.0

Uninstalling setuptools-41.2.0:

Successfully uninstalled setuptools-41.2.0

Successfully installed pip-19.3.1 setuptools-41.6.0 wheel-0.33.6

# Install “West”

C:\Users\Lawrence>pip3 install -U west

Collecting west

Downloading https://files.pythonhosted.org/packages/b5/9e/e8132ee37bff2dd7172a1e6e6a613ff60060336923411c37d571dc9bc1f2/west-0.6.3-py3-none-any.whl (53kB)

|████████████████████████████████| 61kB 787kB/s

Collecting PyYAML

. . . .

Successfully built PyYAML configobj docopt

Installing collected packages: PyYAML, colorama, six, python-dateutil, docopt, pykwalify, configobj, west

Successfully installed PyYAML-5.1.2 colorama-0.4.1 configobj-5.0.6 docopt-0.6.2 pykwalify-1.7.0 python-dateutil-2.8.1 six-1.12.0 west-0.6.3

# Clone the Zephyr Repositories

C:\Users\Lawrence>west init zephyrproject

=== Initializing in c:\users\lawrence\zephyrproject

--- Cloning manifest repository from https://github.com/zephyrproject-rtos/zephyr, rev. master

Initialized empty Git repository in c:/Users/Lawrence/zephyrproject/.west/manifest-tmp/.git/

remote: Enumerating objects: 390198, done.

remote: Total 390198 (delta 0), reused 0 (delta 0), pack-reused 390198

Receiving objects: 100% (390198/390198), 282.21 MiB | 3.79 MiB/s, done.

Resolving deltas: 100% (291629/291629), done.

From https://github.com/zephyrproject-rtos/zephyr

\* branch master -> FETCH\_HEAD

\* [new branch] arm -> origin/arm

\* [new branch] backport-17058-to-v1.14-branch -> origin/backport-17058-to-v1.14-branch

\* [new branch] backport-17066-to-v1.14-branch -> origin/backport-17066-to-v1.14-branch

. . . . . .

39f2281c89a428ff26797ef46e56e845afa1525f refs/remotes/origin/master

Checking out files: 100% (10268/10268), done.

Already on 'master'

Branch 'master' set up to track remote branch 'master' from 'origin'.

=== Initialized. Now run "west update" inside c:\users\lawrence\zephyrproject.

C:\Users\Lawrence>cd zephyrproject

C:\Users\Lawrence\zephyrproject>west update

=== updating hal\_atmel (modules\hal\atmel):

--- hal\_atmel: cloning and initializing

Initialized empty Git repository in C:/Users/Lawrence/zephyrproject/modules/hal/atmel/.git/

--- hal\_atmel: fetching, need revision 04ff67a0826a51041e51034faf8fc4d3eeacd846

remote: Enumerating objects: 1509, done.

remote: Counting objects: 100% (1509/1509), done.

remote: Compressing objects: 100% (484/484), done.

remote: Total 1509 (delta 1117), reused 1392 (delta 1001), pack-reused 0

Receiving objects: 100% (1509/1509), 2.43 MiB | 6.37 MiB/s, done.

. . . .

HEAD is now at e01f3bc check\_compliance.py: Preserve 'info\_msg' better in XML

--- ci-tools: checked out e01f3bce2a94847253369efb9a081f5c0e9ec882 as detached HEAD

=== updating civetweb (modules\lib\civetweb):

--- civetweb: cloning and initializing

Initialized empty Git repository in C:/Users/Lawrence/zephyrproject/modules/lib/civetweb/.git/

--- civetweb: fetching, need revision 99129c5efc907ea613c4b73ccff07581feb58a7a

remote: Enumerating objects: 38, done.

remote: Counting objects: 100% (38/38), done.

remote: Compressing objects: 100% (31/31), done.

. . . . .

# Install Python Dependencies

C:\Users\Lawrence\zephyrproject>pip3 install -r zephyr/scripts/requirements.txt

Collecting Pillow

Downloading https://files.pythonhosted.org/packages/dc/f3/c4244b8bb4175889a12e483d9d9ab51137dc9d7f1cbdfcf37939d14ba7f9/Pillow-6.2.1-cp38-cp38-win32.whl (1.8MB)

|████████████████████████████████| 1.8MB 1.7MB/s

Requirement already satisfied: PyYAML>=3.13 in c:\users\lawrence\appdata\local\programs\python\python38-32\lib\site-packages (from -r zephyr/scripts/requirements.txt (line 2)) (5.1.2)

Collecting breathe>=4.9.1

. . . . . . . . . .

# Install Windows Host Dependencies

Windows instructions here: <https://docs.zephyrproject.org/latest/getting_started/installation_win.html>

Start PowerShell as an administrator and run the following:

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\WINDOWS\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))

Getting latest version of the Chocolatey package for download.

Getting Chocolatey from https://chocolatey.org/api/v2/package/chocolatey/0.10.15.

Downloading 7-Zip commandline tool prior to extraction.

Extracting C:\Users\Lawrence\AppData\Local\Temp\chocolatey\chocInstall\chocolatey.zip to C:\Users\Lawrence\AppData\Local\Temp\chocolatey\chocInstall...

Installing chocolatey on this machine

Creating ChocolateyInstall as an environment variable (targeting 'Machine')

Setting ChocolateyInstall to 'C:\ProgramData\chocolatey'

WARNING: It's very likely you will need to close and reopen your shell

before you can use choco.

Restricting write permissions to Administrators

We are setting up the Chocolatey package repository.

The packages themselves go to 'C:\ProgramData\chocolatey\lib'

(i.e. C:\ProgramData\chocolatey\lib\yourPackageName).

A shim file for the command line goes to 'C:\ProgramData\chocolatey\bin'

and points to an executable in 'C:\ProgramData\chocolatey\lib\yourPackageName'.

Creating Chocolatey folders if they do not already exist.

WARNING: You can safely ignore errors related to missing log files when

upgrading from a version of Chocolatey less than 0.9.9.

'Batch file could not be found' is also safe to ignore.

'The system cannot find the file specified' - also safe.

chocolatey.nupkg file not installed in lib.

Attempting to locate it from bootstrapper.

PATH environment variable does not have C:\ProgramData\chocolatey\bin in it. Adding...

WARNING: Not setting tab completion: Profile file does not exist at

'C:\Users\Lawrence\Documents\WindowsPowerShell\Microsoft.PowerShell\_profile.ps1'.

Chocolatey (choco.exe) is now ready.

You can call choco from anywhere, command line or powershell by typing choco.

Run choco /? for a list of functions.

You may need to shut down and restart powershell and/or consoles

first prior to using choco.

Ensuring chocolatey commands are on the path

Ensuring chocolatey.nupkg is in the lib folder

PS C:\WINDOWS\system32>

Now open a CMD.EXE as Administrator

Microsoft Windows [Version 10.0.18362.449]

(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>choco feature enable -n allowGlobalConfirmation

Chocolatey v0.10.15

Enabled allowGlobalConfirmation

C:\WINDOWS\system32>choco install cmake --installargs 'ADD\_CMAKE\_TO\_PATH=System'

Chocolatey v0.10.15

Installing the following packages:

cmake

By installing you accept licenses for the packages.

Progress: Downloading cmake.install 3.15.5... 100%

Progress: Downloading cmake 3.15.5... 100%

cmake.install v3.15.5 [Approved]

cmake.install package files install completed. Performing other installation steps.

Installing 64-bit cmake.install...

cmake.install has been installed.

cmake.install may be able to be automatically uninstalled.

Environment Vars (like PATH) have changed. Close/reopen your shell to

see the changes (or in powershell/cmd.exe just type `refreshenv`).

The install of cmake.install was successful.

Software installed to 'C:\Program Files\CMake\'

cmake v3.15.5 [Approved]

cmake package files install completed. Performing other installation steps.

The install of cmake was successful.

Software install location not explicitly set, could be in package or

default install location if installer.

Chocolatey installed 2/2 packages.

See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Enjoy using Chocolatey? Explore more amazing features to take your

experience to the next level at

<https://chocolatey.org/compare>

C:\WINDOWS\system32>choco install git python ninja dtc-msys2 gperf

Chocolatey v0.10.15

Installing the following packages:

git;python;ninja;dtc-msys2;gperf

By installing you accept licenses for the packages.

Progress: Downloading git.install 2.24.0... 100%

Progress: Downloading chocolatey-core.extension 1.3.4... 100%

Progress: Downloading git 2.24.0... 100%

. . . . . . . .

Chocolatey installed 8/8 packages.

See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Installed:

- python v3.8.0

- chocolatey-core.extension v1.3.4

- python3 v3.8.0

- gperf v3.1

- dtc-msys2 v1.4.7

- git.install v2.24.0

- ninja v1.9.0.20190208

- git v2.24.0

Start a new CMD.EXE

# Install a ToolChain

Download the installer for gcc-arm-none-eabi-7-2018-q2-update-win32-sha2.exe from <https://developer.arm.com/tools-and-software/open-source-software/developer-tools/gnu-toolchain/gnu-rm/downloads> do not use the latest version….

Run the installer and ensure

* The destination directory is c:\gnu\_arm\_embedded
* “Add path to environmental variable” is selected

In your CMD.exe run the following commands

C:\Users\Lawrence> Setx ZEPHYR\_TOOLCHAIN\_VARIANT gnuarmemb

C:\Users\Lawrence> Setx GNUARMEMB\_TOOLCHAIN\_PATH c:\gnu\_arm\_embedded

Start a new CMD.EXE and check you have the environment variables set correctly

**#** Windows

**C:\Users\Lawrence>** echo %ZEPHYR\_TOOLCHAIN\_VARIANT%

gnuarmemb

**C:\Users\Lawrence>** echo %GNUARMEMB\_TOOLCHAIN\_PATH%

C:\gnu\_arm\_embedded

# Install some more dependencies

Start a new CMD.EXE (not as administrator)

C:\Users\Lawrence>cd zephyrproject

C:\Users\Lawrence\zephyrproject>cd zephyr

C:\Users\Lawrence\zephyrproject>pip3 install -r zephyr/scripts/requirements.txt

Collecting Pillow

Downloading https://files.pythonhosted.org/packages/b6/5a/ee223b0503ebba3d6adcf24516cbae85d581058f2bac635fc82e4bd6c2bf/Pillow-6.2.1-cp38-cp38-win\_amd64.whl (2.0MB)

|████████████████████████████████| 2.0MB 1.7MB/s

Requirement already satisfied: PyYAML>=3.13 in c:\python38\lib\site-packages (from -r zephyr/scripts/requirements.txt (line 2)) (5.1.2)

Collecting breathe>=4.9.1

Using cached https://files.pythonhosted.org/packages/c6/79/11b2b6ef0e398f453c87d48330914cc44c214c23bf61306fa688ddeedb29/breathe-4.13.1-py3-none-any.whl

Collecting colorama

. . . . .

C:\Users\Lawrence\zephyrproject\zephyr>python -m pip install --upgrade pip

Collecting pip

Using cached https://files.pythonhosted.org/packages/00/b6/9cfa56b4081ad13874b0c6f96af8ce16cfbc1cb06bedf8e9164ce5551ec1/pip-19.3.1-py2.py3-none-any.whl

Installing collected packages: pip

Found existing installation: pip 19.2.3

Uninstalling pip-19.2.3:

Successfully uninstalled pip-19.2.3

Successfully installed pip-19.3.1

C:\Users\Lawrence\zephyrproject\zephyr>pip3 install -U PyYAML

Collecting PyYAML

Using cached https://files.pythonhosted.org/packages/e3/e8/b3212641ee2718d556df0f23f78de8303f068fe29cdaa7a91018849582fe/PyYAML-5.1.2.tar.gz

Installing collected packages: PyYAML

Running setup.py install for PyYAML ... done

Successfully installed PyYAML-5.1.2

WARNING: You are using pip version 19.2.3, however version 19.3.1 is available.

You should consider upgrading via the 'python -m pip install --upgrade pip' command.

Next we need node.js to install xpm, to install OpenOCD the debugger.

Install node.js from <https://nodejs.org/en/download/current/> (Note: this installs chocolatey which we did above, so it is probably better to install node.js first) then

C:\Users\Lawrence>node --version

v13.0.1

C:\Users\Lawrence>npm install --global xpm

C:\Users\Lawrence\AppData\Roaming\npm\xpm -> C:\Users\Lawrence\AppData\Roaming\npm\node\_modules\xpm\bin\xpm.js

+ xpm@0.5.0

added 260 packages from 147 contributors in 16.948s

C:\Users\Lawrence>xpm --version

0.5.0

C:\Users\Lawrence>xpm install --global @xpack-dev-tools/openocd@latest

xPack manager - install package(s)

Processing @xpack-dev-tools/openocd@0.10.0-13.1...

Installing globally in 'C:\Users\Lawrence\AppData\Roaming\xPacks\@xpack-dev-tools\openocd\0.10.0-13.1'...

Downloading https://github.com/xpack-dev-tools/openocd-xpack/releases/download/v0.10.0-13/xpack-openocd-0.10.0-13-win32-x64.zip...

Extracting 'xpack-openocd-0.10.0-13-win32-x64.zip'...

847 files extracted.

'xpm install' completed in 3.895 sec.

# Build the Hello World Program

Finally lets build the Hello World program to prove the development system is set up correctlyL

C:\Users\Lawrence\zephyrproject\zephyr>west boards

em\_starterkit

em\_starterkit\_em11d

em\_starterkit\_em7d

em\_starterkit\_em7d\_normal

em\_starterkit\_em7d\_v22

. . . .

C:\Users\Lawrence\zephyrproject\zephyr>west build -b frdm\_kl25z samples/hello\_world

-- west build: build configuration:

source directory: C:\Users\Lawrence\zephyrproject\zephyr\samples\hello\_world

build directory: C:\Users\Lawrence\zephyrproject\zephyr\build

BOARD: frdm\_kl25z (origin: command line)

-- west build: generating a build system

Zephyr version: 2.0.99

-- Found PythonInterp: C:/Python38/python.exe (found suitable version "3.8.0", minimum required is "3.4")

-- Selected BOARD frdm\_kl25z

-- Found west: C:/Python38/Scripts/west.exe (found suitable version "0.6.3", minimum required is "0.6.0")

-- Loading C:/Users/Lawrence/zephyrproject/zephyr/boards/arm/frdm\_kl25z/frdm\_kl25z.dts as base

Devicetree configuration written to C:/Users/Lawrence/zephyrproject/zephyr/build/zephyr/include/generated/generated\_dts\_board.conf

Parsing Kconfig tree in C:/Users/Lawrence/zephyrproject/zephyr/Kconfig

Loaded configuration 'C:/Users/Lawrence/zephyrproject/zephyr/boards/arm/frdm\_kl25z/frdm\_kl25z\_defconfig'

Merged configuration 'C:/Users/Lawrence/zephyrproject/zephyr/samples/hello\_world/prj.conf'

Configuration saved to 'C:/Users/Lawrence/zephyrproject/zephyr/build/zephyr/.config'

-- The C compiler identification is GNU 7.3.1

-- The CXX compiler identification is GNU 7.3.1

-- The ASM compiler identification is GNU

-- Found assembler: C:/gnu\_arm\_embedded/bin/arm-none-eabi-gcc.exe

-- Cache files will be written to: C:\Users\Lawrence\AppData\Local/.cache/zephyr

-- Configuring done

-- Generating done

-- Build files have been written to: C:/Users/Lawrence/zephyrproject/zephyr/build

-- west build: building application

[108/113] Linking C executable zephyr\zephyr\_prebuilt.elf

Memory region Used Size Region Size %age Used

FLASH: 11044 B 128 KB 8.43%

SRAM: 3864 B 16 KB 23.58%

IDT\_LIST: 40 B 2 KB 1.95%

[113/113] Linking C executable zephyr\zephyr.elf

# Install a debugger

At this time pyocd should be installed, this works for the frdm\_kl25z board, however it needs libusb for support. Libusb is here <https://github.com/libusb/libusb/releases/tag/v1.0.21> you need version “libusb v1.0.21”

<https://www.nxp.com/design/microcontrollers-developer-resources/ides-for-kinetis-mcus/opensda-serial-and-debug-adapter:OPENSDA?&tid=vanOpenSDA#FRDM-KL25Z>

# Build and Flash Blinky

Now build and flash the blinky program onto your board. We have several boards available:

stm32l476g\_disco

frdm\_kl25z

nrf52840\_mdk

Substitute the correct name for your board

First clear out the build director (when changing board types, or source projects)

C:\Users\Lawrence\zephyrproject\zephyr>del build

C:\Users\Lawrence\zephyrproject\zephyr\build\\*, Are you sure (Y/N)? y

C:\Users\Lawrence\zephyrproject\zephyr>west build -b frdm\_kl25z samples/basic/blinky

And finally flash the blinky program onto the board

C:\Users\Lawrence\zephyrproject\zephyr>west flash

-- west flash: rebuilding

ninja: no work to do.

-- west flash: using runner pyocd

-- runners.pyocd: Flashing file: C:/Users/Lawrence/zephyrproject/zephyr/build/zephyr/zephyr.hex

[====================] 100%

0001396:INFO:loader:Erased 1024 bytes (1 sector), programmed 1024 bytes (1 page), skipped 13312 bytes (13 pages) at 20.27 kB/s