

# Processing

[Processing](#) is an open source programming language and environment for people who want to create images, animations, and interactions. Compared to Java, Processing greatly simplifies and enhances 2- and 3-D image manipulation, audio, and video. Processing is built on top of Java, which makes integration with the Finch's existing Java libraries fairly easy. We have created a contributed Processing library based on the Finch Java library.

## Documentation

Our Processing library uses the Finch Java library; you can use the methods described in our [Javadocs](#). The docs cover the methods used to control the Finch, read Finch sensor data, read data from RSS feeds, grab image data from webcams, and control computer speakers.

### Installing the Environment and Running an Example

1. If you haven't, download and install [Processing](#).
2. Download the Finch Processing package for your OS.
3. Launch and then close Processing once - this is necessary for Processing to set up a folder in your documents folder.
4. The Finch download is a Processing library, so follow the instructions for [installing a Processing library](#).
5. Launch Processing. If you've correctly installed the library, you can find three Finch example programs by going to File->Sketchbook->libraries->finch->examples.

Tips for creating your own programs:

- All Finch programs require the following import statement: `import edu.cmu.ri.createlab.terk.robot.finch.*`; Do not use Processing's "import library" menu option.
- You should instantiate a Finch object globally (not in any of the methods), so that you can use it in the different methods.

- In addition to Processing's standard setup and draw methods, you should add a stop method. This is a standard method Processing calls. The code in stop is executed if you close the window or hit the stop button. In this method, call the Finch's quit() method (all three examples have a stop method).

## Structure of the Library Package

The Finch Processing library conforms to [Processing's library specification](#):

library - Location of the finch.jar library as well as two libraries for Finch webcam support.

reference - Location of the Javadocs as well as of licensing details.

examples - Contains three example Processing programs written for the Finch

src - Contains a link to the finch.jar's source.

## Example files

The Processing package includes three example files demonstrating how to read sensors and set the LED color and motor speeds in Processing. To see the example files, navigate to File->Sketchbook->libraries->finch->examples

## Known Issues

There are two ways to import libraries in Processing - one is by manually typing an import statement like "import edu.cmu.ri.createlab.terk.robot.finch.\*;". The other is to navigate to Sketch->Import Library. If you use the second option, Processing will import every package in the library. There are several packages in the Finch library that are called "enum". This is a keyword in Processing and causes an error. Due to this, we recommend that you DO NOT use the Sketch->Import Library option. Instead, place the following import statement at the top of your program:

```
import edu.cmu.ri.createlab.terk.robot.finch.*;
```

If you wish to play around with RSS feeds, you can also add this one:

```
import edu.cmu.ri.createlab.rss.readers.*
```

LTI-Civil, the third-party library used to grab image data from webcams, does not work in a 64-bit OS. We recommend you use [Processing libraries for video recording and analysis](#).

Our current download does not appear to work on some Mac OS 10.5 computers. We believe this is because Mac OS 10.5 does not come with a 32-bit JVM of Java 6.

On a Mac, you must search for the libhidapi64.dylib and libhidapi32.dylib libraries and copy them into the Mac's /usr/local/lib directory. To find this directory, select Go/Go to Folder in the Finder. If the directory does not exist, you can create it in Terminal with the following command: `sudo mkdir /usr/local/lib`

## Version Information

Version 1.0 is now available and has been successfully tested on Windows XP 32-bit, Windows 7 64-bit, and Mac OS 10.6 with Processing 1.2.1.