



# The Pi4J Project

Java I/O library for the Raspberry Pi



Raspberry Pi is a trademark of the Raspberry Pi Foundation  
Java is a trademark of the Oracle Corporation

Last Published: 2021-01-29 | Version: 1.3

[Home](#) | [Downloads](#) | [Github](#) | [Issues](#) | [Forum](#) | [@Pi4J \(Twitter\)](#) | [Blog](#)

PI4J

Welcome

[The Team](#)

[Contact](#)

[FAQ](#)

[License](#)

[Dependencies](#)

[Release Notes](#)

[Installation](#)

[Download](#)

[Usage](#)

[Pin Numbering Scheme](#)

[JavaDoc](#)

PIN NUMBERING

[Raspberry Pi - 1A](#)

[Raspberry Pi - 1A+](#)

[Raspberry Pi - 1B \(Rev 1\)](#)

[Raspberry Pi - 1B \(Rev 2\)](#)

[Raspberry Pi - 1B+](#)

[Raspberry Pi - 2B](#)

[Raspberry Pi - 3B](#)

[Raspberry Pi - 3B+](#)

[Raspberry Pi - 3A+](#)

[Raspberry Pi - 4B](#)

[Raspberry Pi - 400](#)

[Raspberry Pi - Zero](#)

[Raspberry Pi - Zero W](#)

[Raspberry Pi - CM1 \(Compute Module\)](#)

[Raspberry Pi - CM3 \(Compute Module\)](#)

[Raspberry Pi - CM3+ \(Compute Module\)](#)

[Raspberry Pi - CM4 \(Compute Module\)](#)

[Lemaker - BananaPi](#)

[Lemaker - BananaPro](#)

[Odroid - C1/C1+](#)

[Odroid - XU4](#)

GPIO EXAMPLES

[Control GPIO](#)

[Listen for GPIO Events](#)

[Shutdown GPIO](#)

[Trigger GPIO on Events](#)

OTHER EXAMPLES

[Serial Communication](#)

[System/Network Information](#)

[More Pi4J Examples](#)

## Welcome to Pi4J!

This project is intended to provide a friendly object-oriented I/O API and implementation libraries for Java Programmers to access the full I/O capabilities of the Raspberry Pi platform. This project abstracts the low-level native integration and interrupt monitoring to enable Java programmers to focus on implementing their application business logic.

## News / Announcements

**2021-01-13**

Version 1.3 has been released and is now available for download. ([Downloads](#) | [Installation](#))

Latest Development Updates:

- Added support for [Raspberry Pi 4B](#)
- Added support for [Raspberry Pi 400](#)
- Added support for [Raspberry Pi Compute Module 4](#)
- Various improvements and bug fixes listed here: [GitHub Issues \(v1.3\)](#)

Additional details can be found in the project's [README](#).

## Project Status

Please note that the Version 1.x codebase for this project is no longer being actively developed. The Version 1.x codebase will only be maintained and updated for major bug fixes. This codebase has been largely stable for several years and is compatible across a wide variety of Raspberry Pi SoCs and you are welcome to continue using it. However, for new projects, it is highly recommended to migrate to the [Pi4J Version 2.x](#) codebase.

Bug fixes are maintained in the 'master' branch in the Pi4J GitHub repository and SNAPSHOT builds are published in the [Sonatype OSS repository](#).

Visit the [Downloads](#) page to download the latest Pi4J v1.x library builds.

## Prerequisites

- [Java 8 JRE/JDK](#) - The **Java 8** runtime environment must be installed and configured on your SoC platform.

### ATTENTION:

If you require a newer Java Virtual Machine for your project, please review the following Pi4J projects: [Pi4J v1.4](#) or [Pi4J v2.x](#)

- [WiringPi](#) - The latest WiringPi library must be installed on your RaspberryPi.

### ATTENTION:

The original WiringPi library has been **\*\*DEPRECATED\*\*** and is no longer maintained.

To support RaspberryPi 4B/400/CM4 and newer systems you must install the latest \*unofficial\* WiringPi version which is maintained here: <https://github.com/WiringPi/WiringPi>.

(As of 2021-01-12, this is version 2.60.)

See [Dependencies](#) for additional information.

## Supported Hardware

- [Raspberry Pi - 1A](#)
- [Raspberry Pi - 1B \(Revision 1\)](#)
- [Raspberry Pi - 1B \(Revision 2\)](#)
- [Raspberry Pi - 1A+](#)
- [Raspberry Pi - 1B+](#)
- [Raspberry Pi - 2B](#)
- [Raspberry Pi - 3B](#)
- [Raspberry Pi - 3B+](#)
- [Raspberry Pi - 3A+](#)

#### UTILITIES

[Pi4J Utility Script](#)

#### PRESENTATIONS/ARTICLES

[Devovx Presentations \(2014\)](#)

[Java Magazine Article \(2014\)](#)

[JavaOne Presentation \(2013\)](#)

[Sunrise/Sunset Timer Project](#)

#### PROJECT DEVELOPMENT

[Project Information](#)

[Project Reports](#)

#### PREVIOUS RELEASES

[Pi4J Version 1.2](#)

[Pi4J Version 1.1](#)

[Pi4J Version 1.0](#)

[Pi4J Version 0.0.5](#)

[Pi4J Version 0.0.4](#)



- [Raspberry Pi - 4B](#)
- [Raspberry Pi - 400](#)
- [Raspberry Pi - Zero](#)
- [Raspberry Pi - Zero W](#)
- [Raspberry Pi - CM1 \(Compute Module\)](#)
- [Raspberry Pi - CM3 \(Compute Module\)](#)
- [Raspberry Pi - CM3+ \(Compute Module\)](#)
- [Raspberry Pi - CM4 \(Compute Module\)](#)
- [LeMaker - BananaPi](#)
- [LeMaker - BananaPro](#)
- [Odroid - C1/C1+](#)
- [Odroid - XU4](#)

## Features

- Export & unexport GPIO pins
- Configure GPIO pin direction
- Configure GPIO pin edge detection
- Control/write GPIO pin states
- Pulse GPIO pin state
- Read GPIO pin states
- Listen for GPIO pin state changes (interrupt-based; not polling)
- Automatically set GPIO states on program termination (GPIO shutdown)
- Triggers for automation based on pin state changes
- Send & receive data via RS232 serial communication
- I2C Communication
- SPI Communication
- Extensible GPIO Provider interface to add GPIO capacity via expansion boards
- Access system information and network information from the Raspberry Pi
- Wrapper classes for direct access to WiringPi Library from Java

## Getting Started

The simplest method to install Pi4J on your RaspberryPi is to execute the following command directly on your RaspberryPi.

```
curl -sSL https://pi4j.com/install | sudo bash
```

To get started using the Pi4J library, please see the following pages:

- [Installation](#)
- [Usage](#)

## Examples

Review each of the following examples to explore the functionality provided by the Pi4j library.

- [Control GPIO](#)
- [Listen for GPIO Events](#)
- [Shutdown GPIO](#)
- [Trigger GPIO on Events](#)
- [Serial Communication](#)
- [System/Network Information](#)