

Welcome to Pi4J!

This project is intended to provide a bridge between the native libraries and Java for full access to the Raspberry Pi.

Project Status

Please note that this project is currently under development and only SNAPSHOT builds are available at this time.

The functionality is reasonable stable but some documentation and development tasks have not yet been completed.

The API is fairly well defined; however, is subject to minor change until a release build is produced.

You are welcome to start using and integrating this library in your project. Please provide any feedback good, bad or ugly. (Well maybe keep the ugly feedback to a minimum.)

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

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

To get started using the Pi4J library, please see the [Usage](#) page and review each of the examples below 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](#)

Copyright © 2012-2013 [Pi4J](#). All Rights Reserved.