

Reference [Language](http://www.arduino.cc/en/Reference/HomePage) (<http://www.arduino.cc/en/Reference/HomePage>) | [Libraries](http://www.arduino.cc/en/Reference/Libraries) (<http://www.arduino.cc/en/Reference/Libraries>) | [Comparison](http://www.arduino.cc/en/Reference/Comparison) (<http://www.arduino.cc/en/Reference/Comparison>) | [Changes](http://www.arduino.cc/en/Reference/Changes) (<http://www.arduino.cc/en/Reference/Changes>)

Robot library

The Robot library is included with Arduino IDE 1.0.5 and later (<http://www.arduino.cc/en/Main/Software>).

The Robot has a number of built in sensors and actuators. The library is designed to easily access the robot's functionality.

The robot has two boards, a motor board and control board. Each board has a separate programmable processor.

The library allows you to interface with the various sensors and peripherals on the control board :

- potentiometer
- 5 momentary switches
- 160x120 pixel color screen
- 512Kbit EEPROM
- speaker
- compass
- 3 I2C connectors
- 8 TinkerKit input connectors

The library also enables you to do a number of things with the motor board :

- control motor speed and direction
- sense the current used by each motor
- read the state of the 5 floor sensors (also known as line detection sensors)
- access I/O pins on the board
- control an I2C port
- read the state of 4 TinkerKit inputs

For more information about the Robot, visit the getting started guide (<http://www.arduino.cc/en/Guide/Robot>) and the hardware page (<http://www.arduino.cc/en/Main/Robot>).

Library structure

This library enables easy access to the functionality of the Arduino Robot (<http://www.arduino.cc/en/Main/Robot>). It relies on a number of 3rd party libraries including Fat16 (<https://code.google.com/p/fat16lib/>), EasyTransfer (<https://github.com/madsci1016/Arduino-EasyTransfer>), Squawk (<https://github.com/stg/Squawk>), and IRRemote

RobotControl class

This class commands the Control Board as well as all the I/Os and motors on the Motor Board when it has the stock firmware installed.

- RobotControl
(<http://www.arduino.cc/en/Reference/RobotConstructor>) Constructor
- begin (<http://www.arduino.cc/en/Reference/RobotBegin>())
- setMode (<http://www.arduino.cc/en/Reference/RobotSetMode>())
- pauseMode (<http://www.arduino.cc/en/Reference/RobotPauseMode>())
- isActionDone
(<http://www.arduino.cc/en/Reference/RobotIsActionDone>())
- lineFollowConfig
(<http://www.arduino.cc/en/Reference/RobotLineFollowConfig>())
- digitalRead() (<http://www.arduino.cc/en/Reference/RobotDigitalRead>)
- digitalWrite()
(<http://www.arduino.cc/en/Reference/RobotDigitalWrite>)
- analogRead()
(<http://www.arduino.cc/en/Reference/RobotAnalogRead>)
- analogWrite()
(<http://www.arduino.cc/en/Reference/RobotAnalogWrite>)
- updateIR() (<http://www.arduino.cc/en/Reference/RobotUpdateIR>)
- knobRead (<http://www.arduino.cc/en/Reference/RobotKnobRead>())
- compassRead
(<http://www.arduino.cc/en/Reference/RobotCompassRead>())
- keyboardRead
(<http://www.arduino.cc/en/Reference/RobotKeyboardRead>())
- waitContinue
(<http://www.arduino.cc/en/Reference/RobotWaitContinue>())
- motorsWrite
(<http://www.arduino.cc/en/Reference/RobotMotorsWrite>())
- motorsStop (<http://www.arduino.cc/en/Reference/RobotMotorsStop>)()
- turn (<http://www.arduino.cc/en/Reference/RobotTurn>())

(<https://github.com/shirriff/Arduino-IRremote>). It also relies on a number of Arduino libraries like TFT (<http://www.arduino.cc/en/Reference/TFTLibrary>), SPI (<http://www.arduino.cc/en/Reference/SPI>), and Wire (<http://www.arduino.cc/en/Reference/Wire>). Their functionality has been replicated inside the robot's library to optimize the code's size.

It is possible to program both the Control and the Motor boards. However, it is recommended that novice programmers begin with programming the control board, leaving the motor board for later. The library exposes the sensors on both boards through a single object.

There are two main classes that command the robot:

- RobotControl: commands the Control Board as well as the I/Os and motors on the Motor Board, when running the default firmware.
- RobotMotor: commands the Motor Board. Use it to make your own custom version of the Motor Board firmware

Examples

These illustrate the possibilities of the robot to help you get up and running quickly with the basic functionality.

- Logo (<http://www.arduino.cc/en/Tutorial/RobotLogo>) - tell your robot where to go through the on-board keyboard
- Line Following (<http://www.arduino.cc/en/Tutorial/RobotLineFollowing>)
 - draw a racing track and get your robot to run on it
- Disco Bot (<http://www.arduino.cc/en/Tutorial/RobotDiscoBot>) - turn your robot into an 8-bit jukebox and dance to the beat
- Compass (<http://www.arduino.cc/en/Tutorial/RobotCompass>) - plan a treasure hunt with this digital compass
- Inputs (<http://www.arduino.cc/en/Tutorial/RobotInputs>) - learn how to control the knob and the keyboard
- Wheel Calibration (<http://www.arduino.cc/en/Tutorial/RobotWheelCalibration>) - tune the wheels to perform even better
- Runaway Robot (<http://www.arduino.cc/en/Tutorial/RobotRunawayRobot>) - play tag with your robot using a distance sensor
- Remote control (<http://www.arduino.cc/en/Tutorial/RobotRemoteControl>) - reuse that old tv-remote to command the bot on distance
- Picture browser (<http://www.arduino.cc/en/Tutorial/RobotPictureBrowser>) - want to use your own images? This is how
- Rescue (<http://www.arduino.cc/en/Tutorial/RobotRescue>) - train your robot to look for hidden pearls in a maze
- Hello User (<http://www.arduino.cc/en/Tutorial/RobotHelloUser>) - hack the robot's welcome demo and make your own

- pointTo (<http://www.arduino.cc/en/Reference/RobotPointTo>())
- beginSpeaker (<http://www.arduino.cc/en/Reference/RobotBeginSpeaker>())
- playMelody (<http://www.arduino.cc/en/Reference/RobotPlayMelody>())
- beep (<http://www.arduino.cc/en/Reference/RobotBeep>())
- playFile (<http://www.arduino.cc/en/Reference/RobotPlayFile>())
- tuneWrite (<http://www.arduino.cc/en/Reference/RobotTuneWrite>())
- tempoWrite (<http://www.arduino.cc/en/Reference/RobotTempoWrite>)()
- beginTFT (<http://www.arduino.cc/en/Reference/RobotBeginLCD>())
- text (<http://www.arduino.cc/en/Reference/RobotText>())
- drawBMP (<http://www.arduino.cc/en/Reference/RobotDrawBMP>())
- debugPrint (<http://www.arduino.cc/en/Reference/RobotDebugPrint>())
- clearScreen (<http://www.arduino.cc/en/Reference/RobotClearScreen>())
- displayLogos (<http://www.arduino.cc/en/Reference/RobotDisplayLogos>())
- drawCompass (<http://www.arduino.cc/en/Reference/RobotDrawCompass>())
- beginSD (<http://www.arduino.cc/en/Reference/RobotBeginSD>())
- userNameRead (<http://www.arduino.cc/en/Reference/RobotUserNameRead>())
- userNameWrite (<http://www.arduino.cc/en/Reference/RobotUserNameWrite>())
- robotNameRead (<http://www.arduino.cc/en/Reference/RobotRobotNameRead>())
- robotNameWrite (<http://www.arduino.cc/en/Reference/RobotRobotNameWrite>())
- cityNameRead (<http://www.arduino.cc/en/Reference/RobotCityNameRead>())
- cityNameWrite (<http://www.arduino.cc/en/Reference/RobotCityNameWrite>())
- countryNameRead (<http://www.arduino.cc/en/Reference/RobotCountryNameRead>())
- countryNameWrite (<http://www.arduino.cc/en/Reference/RobotCountryNameWrite>())

RobotMotor class

Use this to make your own firmware for the Motor Board.

- RobotMotor (<http://www.arduino.cc/en/Reference/RobotMotorConstructor>) Constructor
- begin (<http://www.arduino.cc/en/Reference/RobotMotorBegin>())
- process (<http://www.arduino.cc/en/Reference/RobotMotorProcess>())
- parseCommand (<http://www.arduino.cc/en/Reference/RobotMotorParseCommand>())
- motorsWrite (<http://www.arduino.cc/en/Reference/RobotMotorMotorsWrite>())
- IRread (<http://www.arduino.cc/en/Reference/RobotMotorIRread>())

Read this document

(<http://www.arduino.cc/en/Reference/RobotMotorWriteYourOwnFirmware>) to learn about making your own firmware for the motor board using the RobotMotor class.

Corrections, suggestions, and new documentation should be posted to the Forum
(<http://arduino.cc/forum/index.php/board,23.0.html>).

The text of the Arduino reference is licensed under a Creative Commons Attribution-ShareAlike 3.0 License
(<http://creativecommons.org/licenses/by-sa/3.0/>). Code samples in the reference are released into the public domain.

Share



NEWSLETTER



©2016 Arduino [Copyright Notice](http://www.arduino.cc/en/Main/CopyrightNotice) [Contact us](http://www.arduino.cc/en/Main/ContactUs) [About us](http://www.arduino.cc/en/Main/AboutUs) [Careers](http://www.arduino.cc/Careers)

<https://twitter.com/arduino> <https://www.facebook.com/official.arduino> <https://plus.google.com/+Arduino>
https://www.flickr.com/photos/arduino_cc <https://youtube.com/arduinoteam>