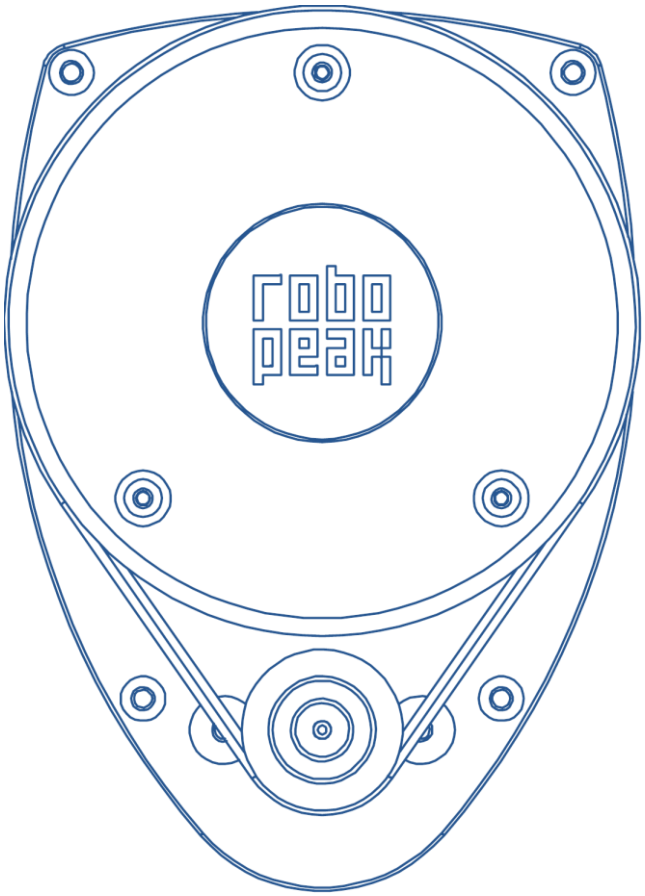
RPLIDAR Application Note

Arduino Driver Support & Demo

2014-5
Rev.1



Contents:

1. Overview

RoboPeak provides RPLIDAR driver library and related demo sketches to help users fast integrate RPLIDAR into their Arduino based systems.

The Arduino driver library can be used on standard Atmel AVR based Arduino boards like Arduino UNO and any other Arduino-compatible boards like Intel Galileo, Leaf Lab maple and etc.

The library provides similar APIs with those appeared in the standard RPLIDAR SDK.

In this document, a simple demo based on the RPLIDAR Arduino driver is introduced.

Performance Issue

Users are required to pay attention to the code efficiency when using this RPLIDAR Arduino driver Library.

As RPLIDAR continuous outputs data to a host system with 2000 samples per second, the host system must **perform data processing task within 500us** for each samples (measurement data) it receives.

For standard Atmel AVR based Arduino board like Arduino UNO or Arduino Mega, due to the slow speed of the MCU, application code must be efficient enough to finish the processing task in time. Otherwise, sample data lose may occur.

For sophisticated applications like SLAM, RoboPeak recommend users to use a more powerful system other than the standard Arduino.

2. RPLIDAR Arduino Driver Library

About the Library

The RPLIDAR's Arduino Driver Library is available on RPLIDAR's product download page:

http://rplidar.robopeak.com

The source code can be found on github:

https://github.com/robopeak/rplidar_arduino

System Requirement

- Arduino IDE:
  1.0.5 or higher

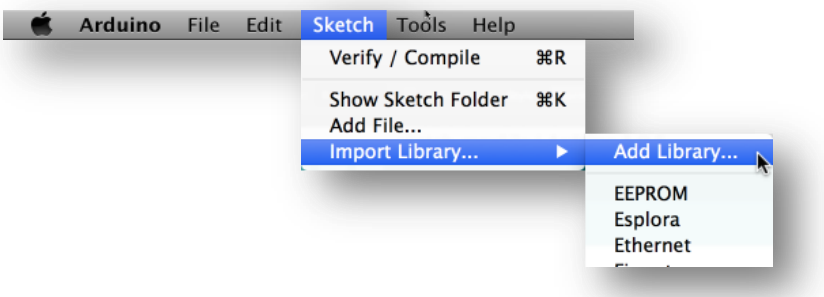
- Hardware:
  Arduino board or compatible versions with AT LEAST one hardware serial port
  RPLIDAR Model A1M1 Development Kit

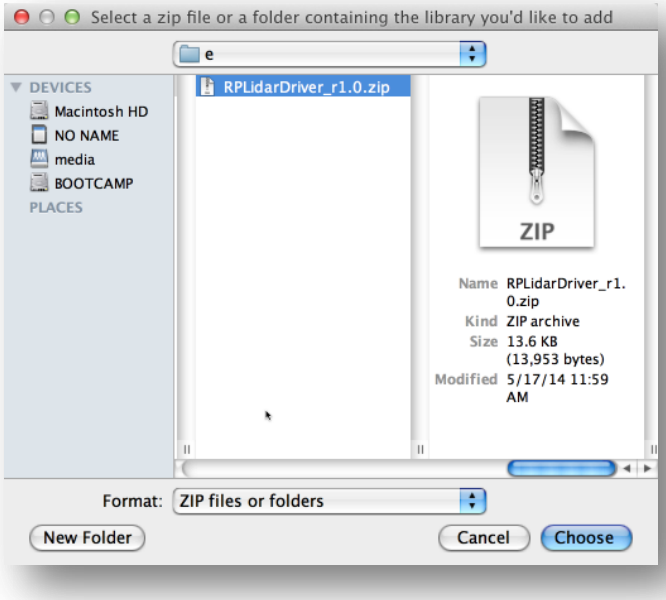
Library Installation

The driver library is provided as a standard Arduino Library, i.e. a ZIP package which can be installed into the Arduino IDE using the Import Library command provided by the Arduino IDE.

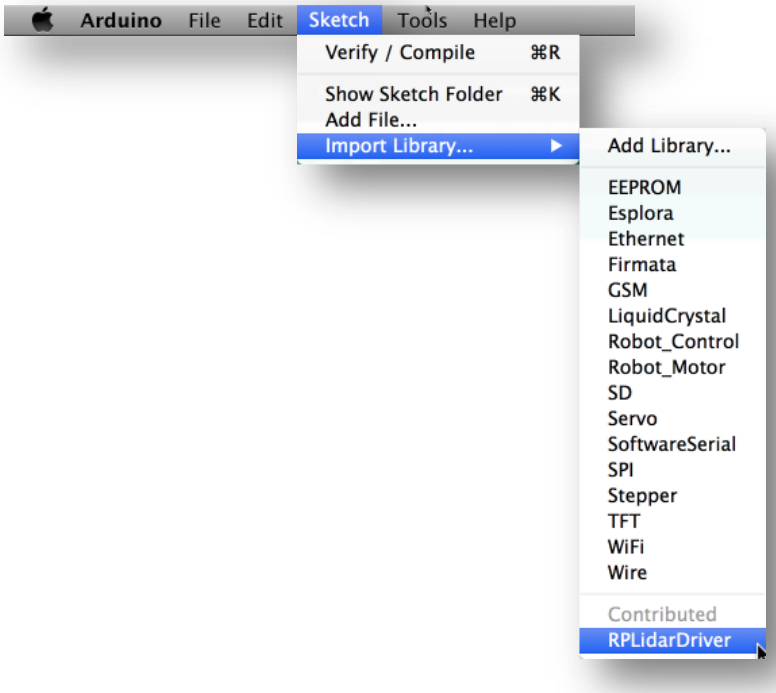
Please follow the following steps to install it, or refer to the instructions on the Arduino website: http://arduino.cc/en/Guide/Libraries

- STEP#1
  Download the Arduino driver library ZIP package, and import it using the Add Library command in the Arduino IDE:





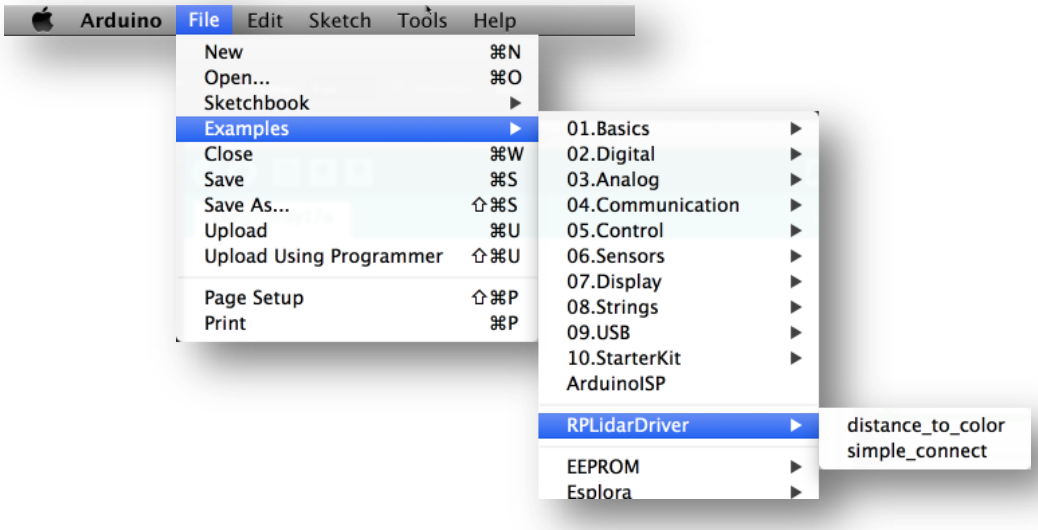
- STEP#2
  Once the driver library gets imported, you will see the related item in the available library list menu.

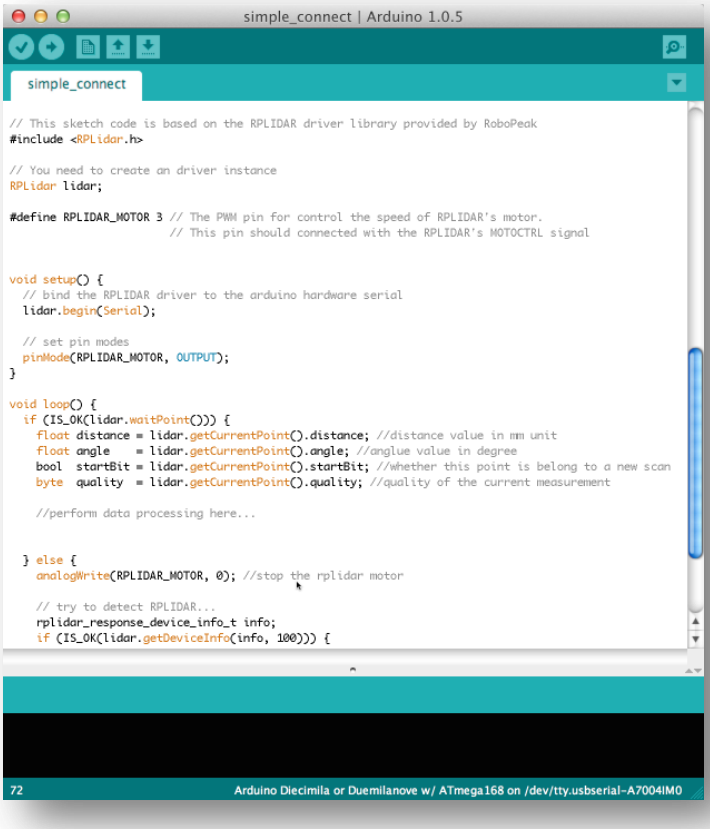


Examples

You may start using the driver library by first referring to the included examples provided by the library package.

Once the library gets installed, you can find the related examples in the example list menu:



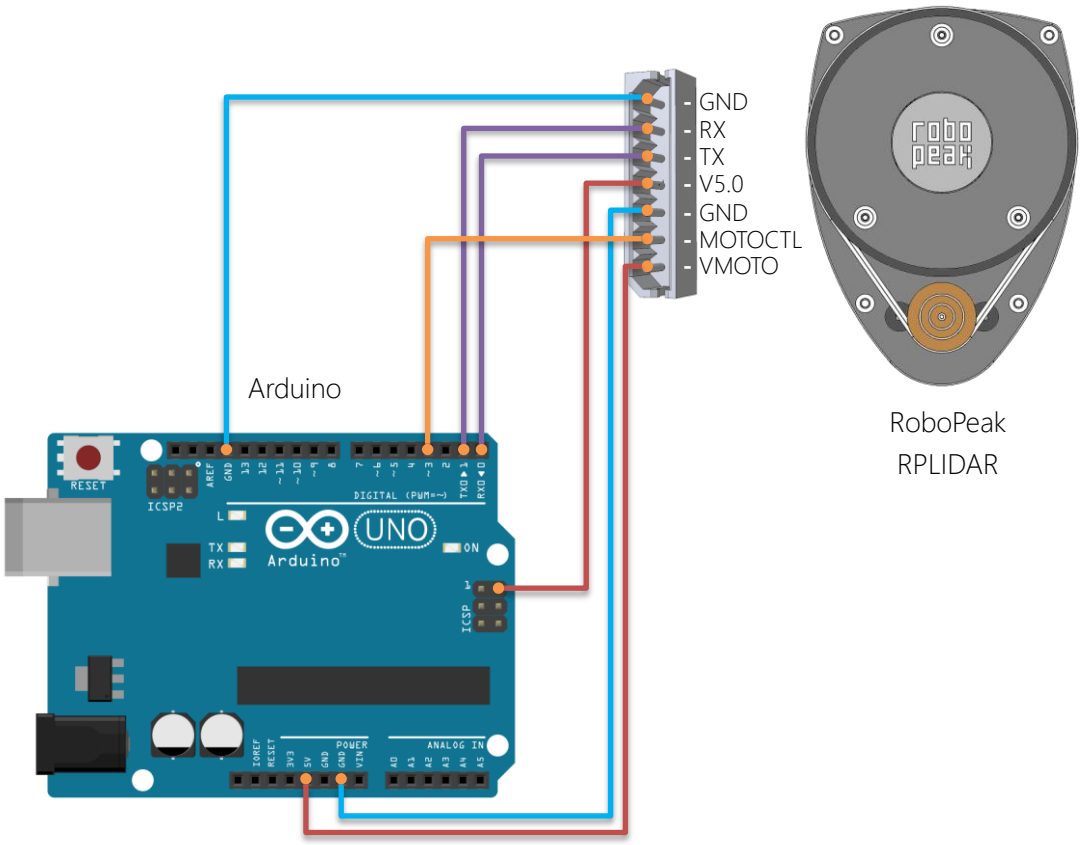


3. Example Demonstrations

Basics

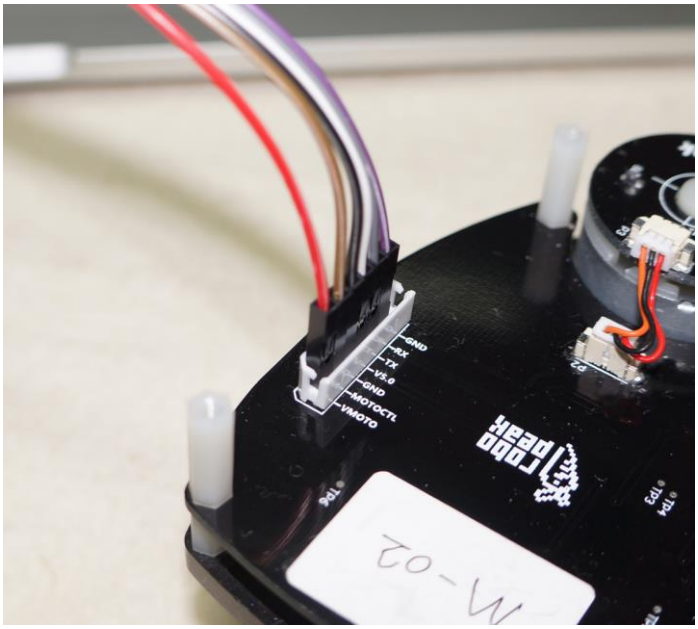
- System Connection:

Connect RPLIDAR to your Arduino board as the following figure illustrates. The 2.5mm dupont wires can be used to make the connection.



RPLIDAR Pin	Arduino Pin	Description
GND	GND	Power Ground
V5.0	5V	5V Power Supply for the RPLIDAR Core
VMOTO	5V/ICSP Pin2	5V Power Supply for the RPLIDAR motor
RX	Pin1 (TXD)	Serial port, RX<-TX
TX	Pin0 (RXD)	Serial port, TX->RX
MOTOCTL	Pin3 (PWM)	RPLIDAR Motor speed control

If you just want the RPLIDAR rotation at maximum speed, the MOTOCTL pin can be connected to the 3V3 pin on the Arduino board.



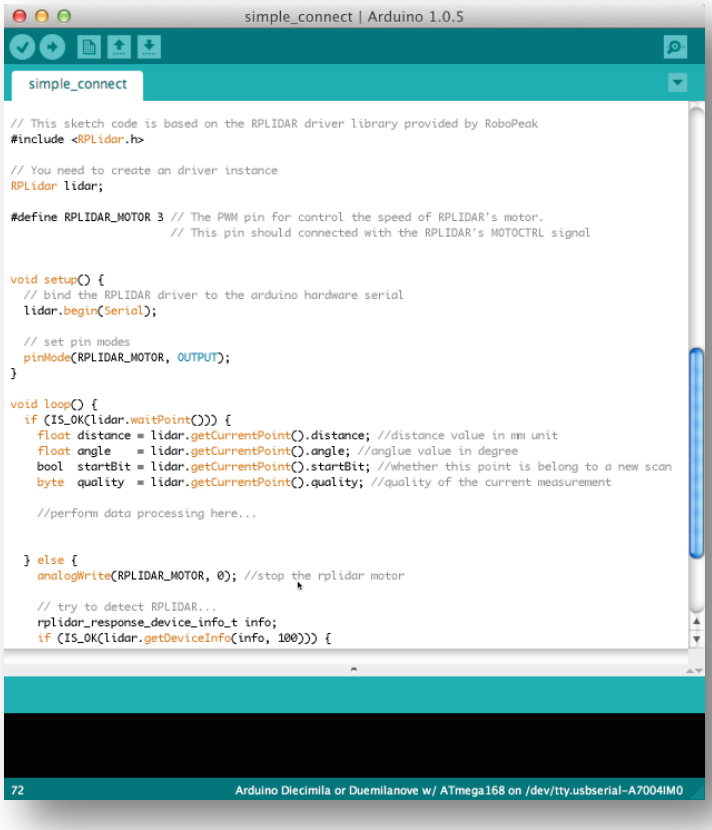
- Caution in Sketch Downloading

As the RXD, TXD pin on Arduino are occupied during sketch downloading operation, you must leave these pins disconnected with the RPLIDAR pins. Otherwise, the download operation will fail.

Example: simple_connect

This example shows the common procedure of connecting to an RPLIDAR, start scanning operation and fetching the measurement data. You may use this example as a template for your application.

The waitPoint() operation returns RESULT_OK when there is a measurement data node has been received by Arduino. Users can add their application code inside this code block.



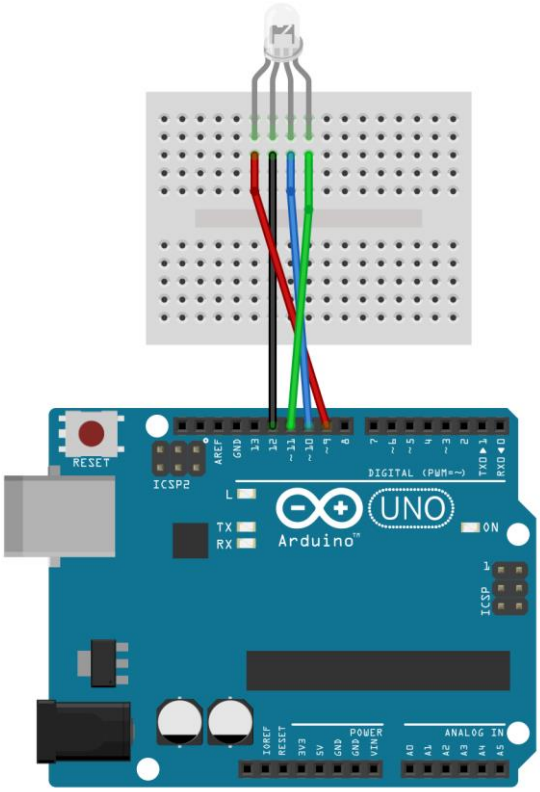
Example: distance_to_color

This example uses an RGB color LED to show the direction and distance of the closest object sensed by the RPLIDAR.

The direction angle of the object is represented by the hue (color) of the LED. The distance is represented by the light intense of the LED.

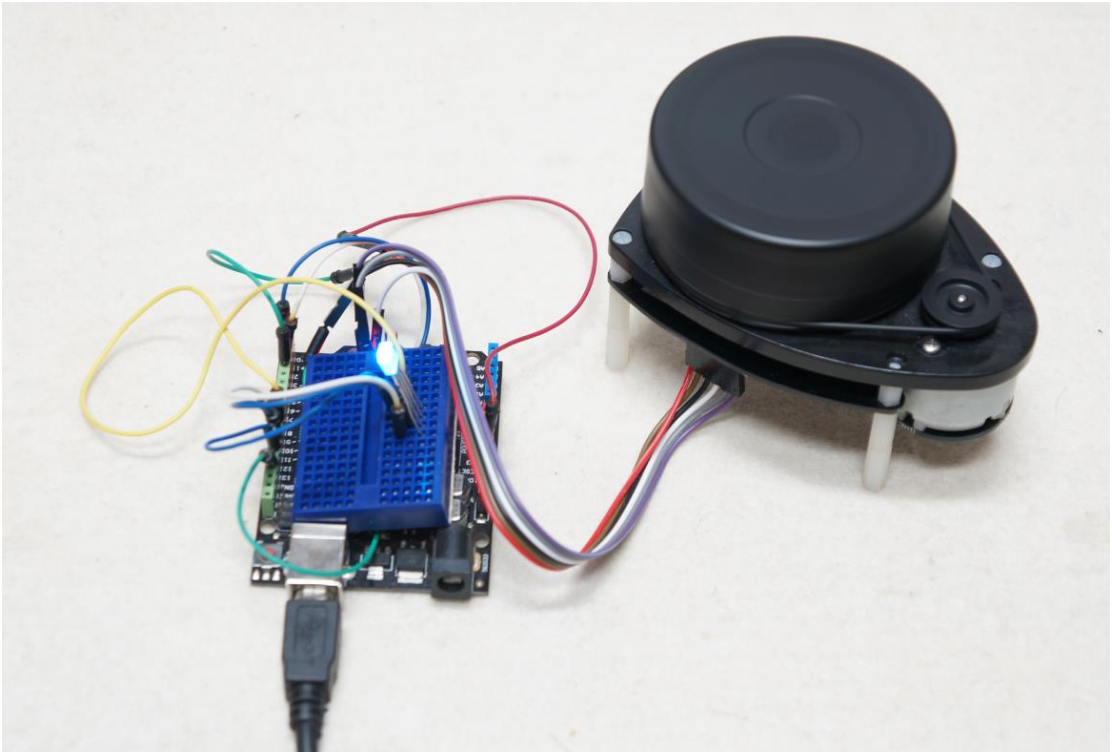
- System Connection:

Please follow the instructions as showed in the above Basics section first. Then make the extra connection to an RGB LED:



LED Pin	Arduino Pin
Common Anode	12
Red	9
Green	11
Blue	10

The whole system looks like the following:



You may refer to the following video for details:

https://www.youtube.com/watch?v=9zgn-GuVt3A

4. Revision History

Date	Content
2013-5-17	Initial draft