

# Kobuki User Guide



i c l e b o  
**Kobuki**

Version 1.0.6  
2016/01/15

# Contents

<a href="#">About</a>
<a href="#">ChangeLog</a>
<a href="#">Safety Guideline</a>
<a href="#">Precautions related to power</a>
<a href="#">Kobuki use restrictions</a>
<a href="#">Get Started</a>
<a href="#">First Run</a>
<a href="#">Level Up</a>
<a href="#">Batteries</a>
<a href="#">Specification</a>
<a href="#">Charging</a>
<a href="#">Replacing The Battery</a>
<a href="#">Anatomy</a>
<a href="#">Top View</a>
<a href="#">Bottom View</a>
<a href="#">Control Panel</a>
<a href="#">Serial Port</a>
<a href="#">Specifications</a>
<a href="#">Functional</a>
<a href="#">Hardware</a>
<a href="#">Software</a>
<a href="#">Firmware</a>
<a href="#">Communication</a>
<a href="#">Protocol Specification</a>
<a href="#">Version Checking</a>
<a href="#">Updating</a>
<a href="#">Special Firmware Modes</a>
<a href="#">Activating</a>
<a href="#">Random Walker Mode</a>
<a href="#">Arduino / Embedded Board Support Mode</a>
<a href="#">Troubleshooting</a>
<a href="#">After Sales Service</a>
<a href="#">Links</a>
<a href="#">Appendix</a>
<a href="#">Accessories</a>
<a href="#">Reference Platforms</a>
<a href="#">License</a>

# About

Introducing Korea's first robotic turtle.

*kobuki* [            ] *n. turtle*

Kobuki is robotically engineered to be long-lived, tough and fast. With high performance batteries, Kobuki will tirelessly work alongside you through those long coffee-powered nights. He'll also happily burden himself with your modded array of sensors, actuators, laptops, embedded boards, portside cannons and do it all at a speed that makes his real world cousins seem like ... well, turtles.

Use him for serving 차|맥 (chi-mek), chasing your neighbour's kids or simply, to make your own robot ideas become reality.

Kobuki is still young, don't expect him to remain as he is . Kobuki's development has already been significantly influenced by the community and as he marches towards old age, we will continue to work with the community and you to ensure he becomes better with time.

Sincerely,  
Kobuki Team.

## ChangeLog

- 1.0.0 : official release
- 1.0.1 : upgrading firmware link fix
- 1.0.2 : documentation link fix
- 1.0.4 : updating the spec. of serial communication
- 1.0.5 : fixing battey & connector link
- 1.0.6 : updating link after Kobuki web site revision

## Safety Guideline

### Precautions related to power

- Do not force to bend the power cord or pressure it with heavy weight.
- Keep the pin and interface of the power plug clean from dust or water.
- Do not pull the power cord or touch the power plug with wet hands.
- Do not use damaged power plug or power cord, loose outlet.
- Do not put or pull the power plug consecutively.
- Do not touch the power cord of the charger with wet hands.

### Kobuki use restrictions

- Kobuki is for indoor use only.
- Do not pour or spray water onto Kobuki.
- Do not use Kobuki to pick up anything that is burning or smoking.
- Always remove the battery before long-term storage or transportation.
- Do not sit or stand on this device.

# Get Started

## First Run

You just unpacked your Kobuki and want to see it move? Sure, we added a special random walker mode, which you can activate on start-up:

1. Turn on Kobuki.
2. Within in the first 3 seconds press and hold button B0 for 2 seconds.
3. LED2 should\* start blinking and Kobuki wandering around.

**Note(\*):** This mode has been introduced to the firmware with version 1.1.0. In case your Kobuki is not running this or a later version, please refer to the section about [updating the firmware](#).

## Level Up

In order to gain access to all of Kobuki's features, you need to provided an external computing unit, which communicates with Kobuki. This could be a laptop, netbook, tablet, an embedded board or other devices.

To let you quickly start using Kobuki, we added tutorials for each supported environment to the documentation section on the Kobuki website:

- [Getting Started - Linux \(ROS\)](#)
- [Getting Started - Windows](#)
- [Getting Started - Embedded](#)

# Batteries

## Specification

Kobuki by the default ships with a small Lithium-Ion battery pack (4S1P, 2200mAh, 14.8V). For extra long operation, a big battery pack (4S2P, 4400mAh, 14,8V) can be ordered as well.

## Charging

In order to charge the battery connect the power adapter to Kobuki or dock Kobuki in the docking station. If Kobuki is turned on, you will hear a short sound when charging starts. The battery is also charged if it is off, but you won't hear any sound, nor will you see any LED signal.

For checking the battery status, turn Kobuki on and have a look at the status LED. Jump to the [Anatomy - Control Panel](#) section for more information about the different statuses.

More details about the batteries can be found on Kobuki's website:

- <http://kobuki.yujinrobot.com/wiki/battery-details/>

## Replacing The Battery

- Turn off Kobuki.
- Flip it and open the battery bay in the centre.
- Unplug and remove the battery.
- Plug in the new battery.
- Close the battery bay and flip it again.
- Turn on Kobuki.

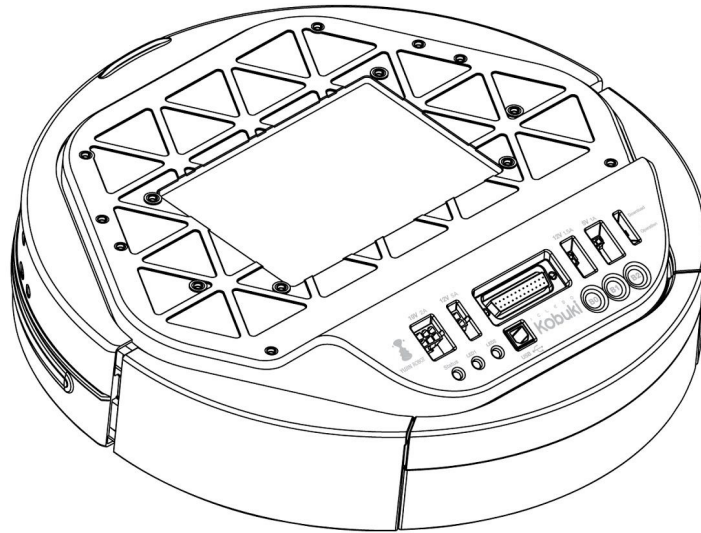
**Note:** It is not possible to use multiple battery packs at the same time.

A more detailed tutorial including pictures is located on Kobuki's website:

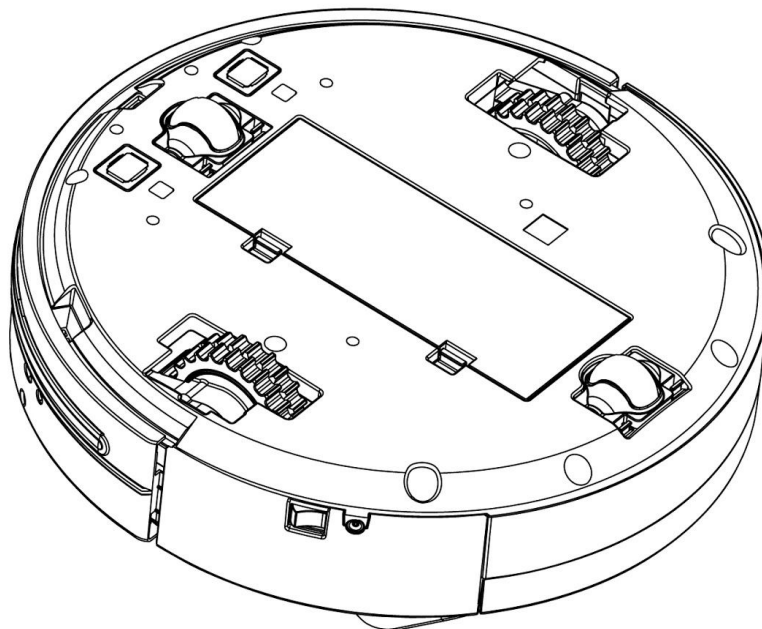
- <http://kobuki.yujinrobot.com/wiki/swapping-batteries/>

# Anatomy

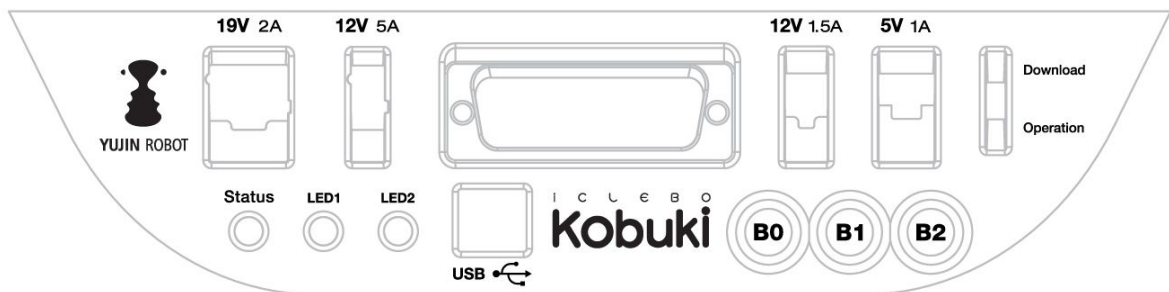
## Top View



## Bottom View



## Control Panel



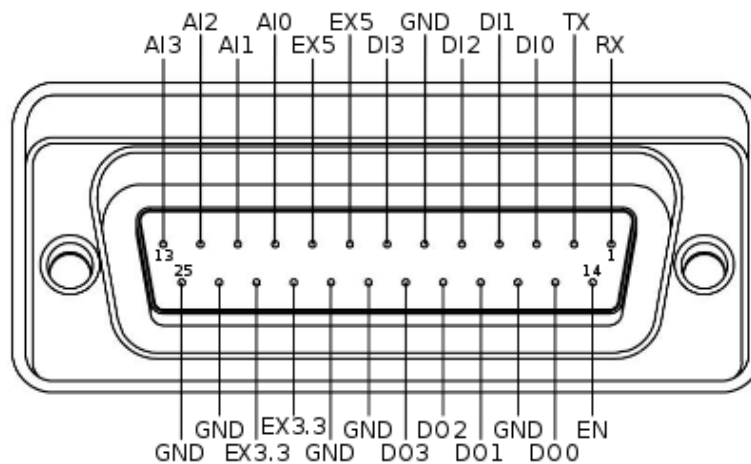
- 19V/2A: Laptop power supply
- 12V/5A: Arm power supply
- 12V/1.5A: Microsoft Kinect power supply
- 5V/1A: General power supply
- Status LED: Indicates Kobuki's status
  - Green: Kobuki is turned on and battery at high voltage level
  - Orange: On - Low battery voltage level (please charge soon)
  - Green blinking: On - Battery charging
  - Off: Kobuki is turned off.
- LED1/2: Programmable LEDs
- USB: Data connection
- B0/1/2: Buttons
- Firmware switch: Enable/disables the firmware update mode

Additional information (e.g. connector types) is located online:

- <http://kobuki.yujinrobot.com/wiki/connectors/>



## Serial Port



- RX / TX: Serial data connection (RS232; used voltage level is 3.3V!)
- EX3.3 / EX5: 3.3V/1A and 5V/1A power supply
- DI0 - 3: 4 x Digital input (high: 3.3 - 5V, low: 0V)
- DO0 - 3: 4 x Digital output (open-drain, pull-up resistor required)
- AI0-3: 4 x Analog input (12bit ADC: 0 - 4095, 0 - 3.3V)
- GND: Ground
- EN: Used for detecting an external board (connect to external ground)

## Specifications

### Functional

- Maximum translational velocity: 70 cm/s
- Maximum rotational velocity: 180 deg/s (>110 deg/s gyro performance will degrade)
- Payload: 5 kg (hard floor), 4 kg (carpet)
- Cliff: will not drive off a cliff with a depth greater than 5cm
- Threshold Climbing: climbs thresholds of 12 mm or lower
- Rug Climbing: climbs rugs of 12 mm or lower
- Expected Operating Time: 3/7 hours (small/large battery)
- Expected Charging Time: 1.5/2.6 hours (small/large battery)
- Docking: within a 2mx5m area in front of the docking station

### Hardware

- PC Connection: USB or via RX/TX pins on the parallel port
- Motor Overload Detection: disables power on detecting high current (>3A)
- Odometry: 52 ticks/enc rev, 2578.33 ticks/wheel rev, 11.7 ticks/mm
- Gyro: factory calibrated, 1 axis (110 deg/s)
- Bumpers: left, center, right

- Cliff sensors: left, center, right
- Wheel drop sensor: left, right
- Power connectors: 5V/1A, 12V/1.5A, 12V/5A
- Expansion pins: 3.3V/1A, 5V/1A, 4 x analog in, 4 x digital in, 4 x digital out
- Audio : several programmable beep sequences
- Programmable LED: 2 x two-coloured LED
- State LED: 1 x two coloured LED [Green - high, Orange - low, Green & Blinking - charging]
- Buttons: 3 x touch buttons
- Battery: Lithium-Ion, 14.8V, 2200 mAh (4S1P - small), 4400 mAh (4S2P - large)
- Firmware upgradeable: via usb
- Sensor Data Rate: 50Hz
- Recharging Adapter: Input: 100-240V AC, 50/60Hz, 1.5A max; Output: 19V DC, 3.16A
- Netbook recharging connector (only enabled when robot is recharging): 19V/2.1A DC
- Docking IR Receiver: left, centre, right

## Software

- C++ drivers for Linux and Windows
- ROS driver
- Gazebo simulation

# Firmware

## Communication

- Baud rate: 115200 BPS, Data bit: 8 bit, Stop bit: 1 bit, No Parity
- Electronic protocol: USB / RS232 (@ DB25 connector)

## Protocol Specification

The driver communicates with the robot by using predefined protocol. In general, the driver sends the commands to the robot and the robot sends some feedback data or sensor readings. This command and feedback data are converted into bytestreams for communication via serial interface. The protocol specify that rules and forms of bytestream.

For more detailed information, visit:

- <http://yujinrobot.github.com/kobuki/doxygen/enAppendixProtocolSpecification.html>

## Version Checking

Every time the robot comes alive (because it is connected or powered on), the driver checks the compatibility between software (i.e. driver) and firmware. Firmware versions are of the form M.m.p, where:

- M (ayor) represents a deep rebuild of the code that almost surely breaks protocol compatibility. In consequence, if mayor version doesn't match, the driver will show an error, suggest the required update and shutdown.
- m (inor) represents a new feature that could not work if the driver is outdated, but the protocol itself is spared. In consequence, if minor version doesn't match, the driver will show a warning suggesting the required update and continue working.
- p (atch) represents a fix on the code and is not checked at all.

You can check which firmware version your robot is running on the log (on of the first lines at driver startup), by running the version\_info program included on the kobuki driver, or if you are using ROS, just echoing /mobile\_base/version\_info topic. Generally speaking, it's recommended to upgrade to the latest stable version. Next section explains how to do it.

## Updating

The firmware update process is a bit different depending on which operating system used for flashing. The documentation section on the Kobuki websites contains detailed instructions on how to update the firmware using Linux and Windows:

- Linux tutorial
  - <http://kobuki.yujinrobot.com/wiki/upgrading-firmware-linux/>
- Windows tutorial

- <http://kobuki.yujinrobot.com/wiki/upgrading-firmware-windows/>

## Special Firmware Modes

### Activating

Kobuki has some special firmware modes, which can be activated on startup.

Currently implemented are:

- I. Random walker mode
- II. Arduino/Embedded board support mode

To activate one of them, follow these instructions:

1. Turn on Kobuki.
2. Within in the first 3 seconds press and hold button BO(I) / B1 (II) for 2 seconds.
3. If\* you see LED2(I) / LED1(II) switching between red and green, your chosen mode is activated.

**Note(\*):** These modes have been introduced to the firmware with version 1.1.0. In case your Kobuki is not running this or a later version, please refer to the section about [updating the firmware](#).

### Random Walker Mode

In the random walker mode Kobuki is driving around until it hits an object with the bumper or a cliff is detected. In both cases, Kobuki will stop, turn by a random amount of degrees and continue driving .

**Note:** In this mode Kobuki's wheel drop sensors are not activated. So, be careful when lifting up Kobuki!

### Arduino / Embedded Board Support Mode

In this mode the serial port (DB25 connector) gives access to basic controls of Kobuki. You can hook up the digital/analog inputs/outputs of your Arduino or other embedded boards and start writing simple control programs.

Below is the special pin setting listed. Please refer to the [serial port description](#) for the name to pin mapping.

- DI0: Not used
- DI1: Not used
- DI2: Not used
- DI3: Not used
- DO0: Bumper left (pressed/released)
- DO1: Bumper centre (pressed/released)
- DO2: Bumper right (pressed/released)
- DO3: Wheel drop sensors (at least one wheel is dropped / none is dropped)

- AI0: Wheel speed right (0V - full speed backward, 3.3V - full speed forward)
- AI1: Wheel speed left (0V - full speed backward, 3.3V - full speed forward)
- AI2: Not used
- AI3: Not used

All other pins (GND, RX, TX etc.) remain unchanged.

**Note:** To enable the motors you need to press button B0.

## **Troubleshooting**

Will be filled as problems appear.

## **After Sales Service**

Kobuki comes with a 1 year warranty. In case you bought Kobuki from one of Yujin Robot's distributors, please contact them if you have questions regarding warranty and after sales service. Otherwise please contact us at [kobuki@yujinrobot.com](mailto:kobuki@yujinrobot.com).

## Links

Website	<a href="http://kobuki.yujinrobot.com">http://kobuki.yujinrobot.com</a>
Blog	<a href="http://blog.yujinrobot.com">http://blog.yujinrobot.com</a>
Mailing List	<a href="mailto:kobuki-users@yujinrobot.com">kobuki-users@yujinrobot.com</a>
Documentation	<a href="http://kobuki.yujinrobot.com/documentation/">http://kobuki.yujinrobot.com/documentation/</a>
Contact	<a href="mailto:kobuki@yujinrobot.com">kobuki@yujinrobot.com</a>



# Appendix

## Accessories

- Small (4S1P) and big (4S2P) battery packs
- Docking Station
- Kinect Modification Kit
- Netbook Modification Kit
- Turtlebot Accessories

## Reference Platforms

- Turtlebot 2

# License

The Kobuki user guide is licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/).

Based on a work at <http://kobuki.yujinrobot.com/home-en/documentation/online-user-guide/>.

