Lidarbot
Odos
Contents

Overview
&Product features

Shipping List
&Application

Protocol
for car bottom board

Parameter
&interfaces

User manual
Match methods

YDLIDAR X2
paramater&interfaces

https://github.com/YDLIDAR/lidarCar (code)
Overview

Odos is a powerful development kit for Automated Guided Vehicles (AGVs). Equipped with YDLIDAR X2, 4 Mecanum wheels, M5 Core, RGB Bars and a remote controller with Joystick panel and more. With 4 Mecanum wheels, you can make it move to any direction like forward, backward, left and right. The Lipo Batteries empower the Robot to run long-hours. You can display the map data, that obtained from the lidar sensor, on the screen or upload somewhere else thru Wi-Fi and program it into any format.

Real-Time communication has implemented via ESP-NOW between robot and remote, Maze-running, self-tracing and more. If you are interested in AGV development, it's encouraged to modify the open-source code we offered on GitHub and enhance it yourself.

Product Features

- Mecanum wheel
- 360 degree Lidar sensor
- Compatible with LEGO structure
- Waterproof metal self-locking switch
- RGB LED strip * 8
Shipping List

- LidarBoT x 1
- remote control handle x 1
- battery (1300mAh @ 11.1V) x 2
- battery charger x 1
- Type-C USB x 1

Application

- Indoor navigation
- Walk the maze autonomously
- route plan
- Autopilot
## Protocol for CarBottomBoard

Protocol Format: Data Header (command type) + Data Packet + Data Tail

<table>
<thead>
<tr>
<th>Control Target</th>
<th>Protocol Format</th>
<th>Example</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wheels</td>
<td>0xAA, SpeedX(-7 ~ 7), SpeedY, SpeedZ, SpeedA, 0x55</td>
<td>0xAA, 5, 5, 5, 5, 0x55 (Go ahead, speed: 5)</td>
<td>ControlWheel(5, 5, 5)</td>
</tr>
<tr>
<td>One RGB</td>
<td>0xAB, LedIndex, R(0 ~ 254), G, B, 0x55</td>
<td>0xAB, 3, 20, 50, 100, 0x55 (3th RGB displays specific color)</td>
<td>setLedColor(3, 20, 50, 100)</td>
</tr>
<tr>
<td>Front RGB Bar</td>
<td>0xAC, R(0 ~ 254), G, B, 0x55</td>
<td>0xAC, 20, 50, 100, 0x55 (Front LED Bar displays specific color)</td>
<td>setFrontLedBar(20, 50, 100)</td>
</tr>
<tr>
<td>Back RGB Bar</td>
<td>0xAD, R(0 ~ 254), G, B, 0x55</td>
<td>0xAD, 20, 50, 100, 0x55 (Back LED Bar displays specific color)</td>
<td>setBackLedBar(20, 50, 100)</td>
</tr>
<tr>
<td>All RGB</td>
<td>0xAE, R(0 ~ 254), G, B, 0x55</td>
<td>0xAE, 20, 50, 100, 0x55 (All LED display specific color)</td>
<td>setLedAll(20, 50, 100)</td>
</tr>
<tr>
<td>ServoMotor0</td>
<td>0xAF, Angle(0 ~ 180), 0x55</td>
<td>0xAF, 100, 0x55 (Servo 0 turns angle 100 degree)</td>
<td>setServo0Angle(100)</td>
</tr>
<tr>
<td>ServoMotor1</td>
<td>0xB0, Angle(0 ~ 180), 0x55</td>
<td>0xB0, 100, 0x55 (Servo 1 turns angle 100 degree)</td>
<td>setServo1Angle(100)</td>
</tr>
</tbody>
</table>
**Parameters and interfaces**

**Communication Parameter:**
M5Core <-> Lidar (U1RXD(GPIO16) <-> Lidar sensor)
Serial Configuration: "230400bps, 8, n, 1"(8 bits data, no parity, 1 stop bit)

M5Core <-> Bottom Board (U2TXD(GPIO17) <-> Bottom Board) Serial Configuration: "115200bps, 8, n, 1"(8 bits data, no parity, 1 stop bit)

**PIN Map:**
ServoMotor0 <-> A0(MEGA328)
ServoMotor1 <-> A1(MEGA328)
RGB LED <-> 11(MEGA328)
Press A and the handle can control Odos left and right walking, front and backward is opposite as before.

① Lidarbot- Long press C button and then press reset button (the red one) ----------------------------------------

② Control handle- Long press A and then reset

③ Lidarbot- press B (ok) (BLUE/RED light on)------------------------

Visit 192.168.4.1/map directly can also view the point cloud map when Odos is ready.
Detail instructions

- **## X2lidarBot**

- **### Control and display**

  Odos and the handle achieve communication through EspNow. Point cloud(map) would be displayed on the screen, and the handle can control Odos.

  - **Normal control mode**: Move the joystick, Odos will realize forward, backward and steering.

  - **Omnidirectional control mode**: Hold down the A button of the handle, (the leftmost one) and then move the joystick to achieve the left and right horizontal movement. Notice: The front and back directions are reversed.

- **### Connect and match**

  In the unconnected state or when one party is not connected to the other, the display or control may have problems. If so, please re-connect.

  - **STEP 1**: Keep holding the C button and press the power button one time. Wait for the screen to restart and then release the C button to enter the broadcast mode.

    All slaves will receive the signal from the master.

    - When Odos enters the broadcast mode, repeat **STEP 1** to see the current broadcast host on the screen. Button A/C means up and down, and button B means to confirm. The Mac address of the host can be viewed from the mobile phone or computer near Wi-Fi. The host starts with lidar and is followed by the host Mac address.

    - After confirming the master, the screen of the master and Odos will receive the confirmation signal of the slave, and the address of the slave and the handle will also be selected and determined by the ABC key. After pressing the B key to confirm, Odos and the handlebar have completed the communication configuration, and the two parties can send messages to each other to realize the Lidar chart display and handlebar control.

- **### Web page shows the Lidar image**

  If you don’t want to match the handle and Odos, please visit 192.168.4.1/map to check the Lidar chart information.
YDLIDAR X2

Dimensions
### Product Parameter

<table>
<thead>
<tr>
<th>Item</th>
<th>Min</th>
<th>Typical</th>
<th>Max</th>
<th>Unit</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ranging frequency</td>
<td>-</td>
<td>3000</td>
<td>-</td>
<td>Hz</td>
<td>3000 times per second</td>
</tr>
<tr>
<td>Motor frequency</td>
<td>-</td>
<td>7</td>
<td>-</td>
<td>Hz</td>
<td>PWM or Voltage Regulation</td>
</tr>
<tr>
<td>Ranging distance</td>
<td>0.10</td>
<td>-</td>
<td>&gt;8</td>
<td>m</td>
<td>Indoor</td>
</tr>
<tr>
<td>Scanning angle</td>
<td>-</td>
<td>0~360</td>
<td>-</td>
<td>Deg</td>
<td></td>
</tr>
<tr>
<td>Absolute error</td>
<td>-</td>
<td>2</td>
<td>-</td>
<td>cm</td>
<td>Distance≤0.5m</td>
</tr>
<tr>
<td>Relative error</td>
<td>-</td>
<td>1.5%</td>
<td>-</td>
<td>-</td>
<td>0.5m&lt;Distance≤6m</td>
</tr>
<tr>
<td>Angle resolution</td>
<td>0.82</td>
<td>0.84</td>
<td>0.86</td>
<td>Deg</td>
<td>Scanning frequency=7</td>
</tr>
</tbody>
</table>

### Electrical Parameter

<table>
<thead>
<tr>
<th>Item</th>
<th>Min</th>
<th>Typical</th>
<th>Max</th>
<th>Unit</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply voltage</td>
<td>4.8</td>
<td>5</td>
<td>5.2</td>
<td>V</td>
<td>Excessive voltage might damage the Lidar while low affect normal performance</td>
</tr>
<tr>
<td>Voltage ripple</td>
<td>0</td>
<td>50</td>
<td>100</td>
<td>mV</td>
<td>Excessive ripple affect normal performance</td>
</tr>
<tr>
<td>Starting current</td>
<td>300</td>
<td>400</td>
<td>500</td>
<td>mA</td>
<td>Higher current required at start-up</td>
</tr>
<tr>
<td>Working current</td>
<td>200</td>
<td>350</td>
<td>380</td>
<td>mA</td>
<td>Normal working</td>
</tr>
</tbody>
</table>

### Interface

- **Pin**
  - **VCC**: Power Supply, Positive, Defults: 5V, Range: 4.8V-5.2V, Remarks: -
  - **Tx**: Output, System serial output, Defults: - , Range: - , Remarks: Data stream, Lidar ➔ Peripherals
  - **GND**: Power Supply, negative, Defults: 0V, Range: 0V, Remarks: -
  - **M_SCTR**: Input, Motor speed control terminal, Defults: 1.8V, Range: 0V-3.3V, Voltage or PWM speed regulation