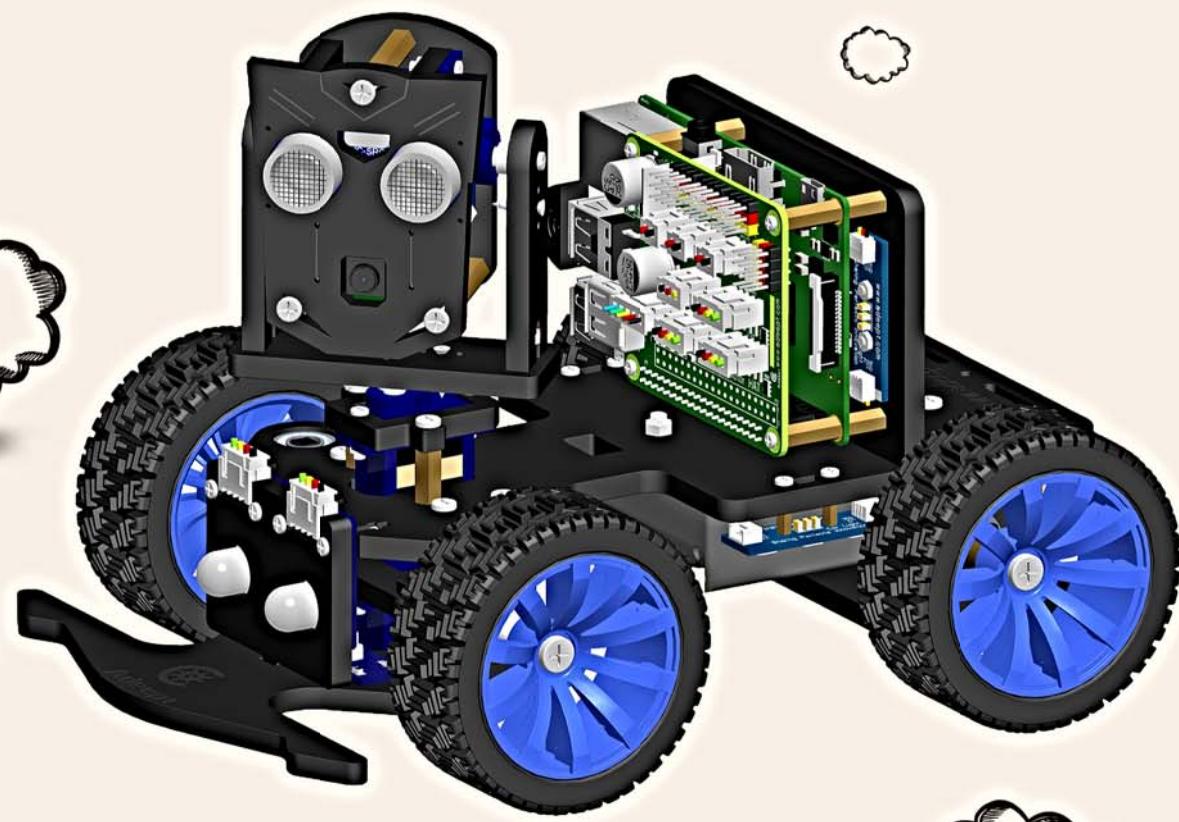
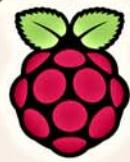




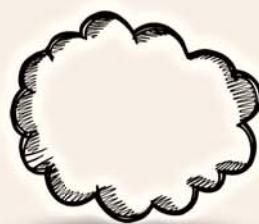
Adeept

# Adeept Smart Car Robot Kit for Raspberry Pi PiCar-B



SETS

SETS



## Warning

Please pay attention to the following issues when purchasing or using the product:

- ★ There are small components included in this kit. Swallowing mistakenly or misoperation can cause serious infection and be even fatal. When an accident occurs, please seek medical assistance immediately.
- ★ Please place the product in a safe place where an under-6-year-old cannot touch, who should not use or approach the product.
- ★ Juveniles should use the product with their parents.
- ★ Do not place the product or the components near any AC socket or other circuits to avoid electric shock.
- ★ Do not use the product near any liquid or flame.
- ★ Do not use or store the product in an extreme environment such as in extremely low or high temperature and heavy humidity.
- ★ Please remember to power off when the product is not in use.
- ★ Do not touch the moving or rotating part of the product.
- ★ The product may get heat at some part, which is just normal. But misoperation may cause overheat.
- ★ Misoperation may cause damage to the product. Please take care.
- ★ Do not connect the positive and negative poles of the power inversely, or the devices in the circuit may be damaged.
- ★ Please place and put the product gently. Do not smash or shock it.

## About

Adeept is a technical service team of open source software and hardware. Dedicated to applying the Internet and the latest industrial technology in open source area, we strive to provide the best hardware support and software service for general makers and electronic enthusiasts around the world. We aim to create infinite possibilities with sharing. No matter what field you are in, we can lead you into the electronic world and bring your ideas into reality.

The code and manual of our product are open source. You can check on our website:

<http://www.adeept.com/>

If you have any problems, feel free to send an email for technical support and assistance:

[support@adeept.com](mailto:support@adeept.com)

On weekdays, we usually will reply within 24 hours. Also welcome to post in our official forum:

<http://www.adeept.com/forum/>

## Copyright

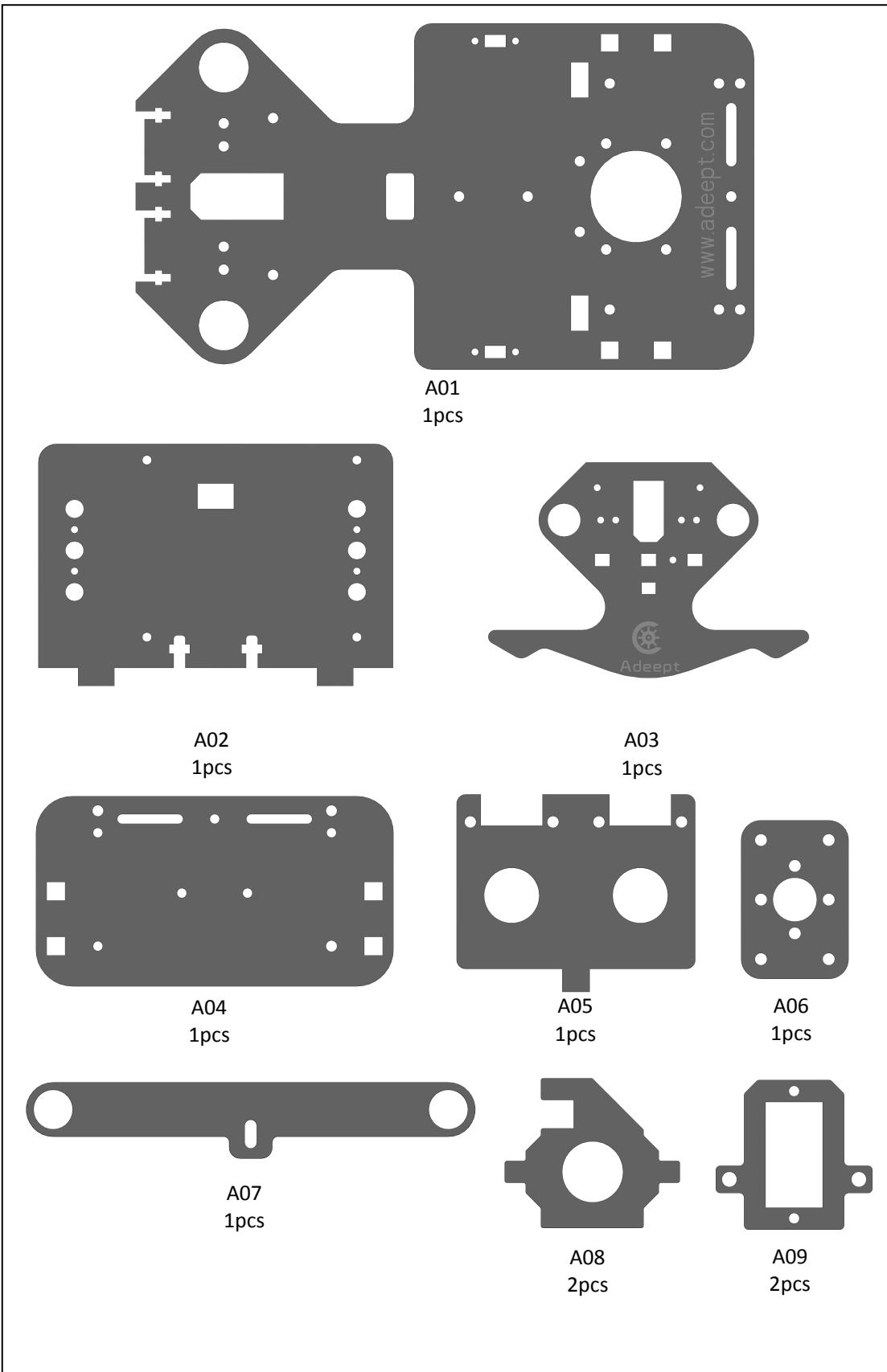
This user manual and code can be used for learning, DIY, refitting, etc., except for commercial purpose. The Adeept Company owns all rights of contents in the manual, including but not limited to texts, images, data, etc. Any distribution or printing should be implemented with the permission of the Company, or it will be deemed illegal.

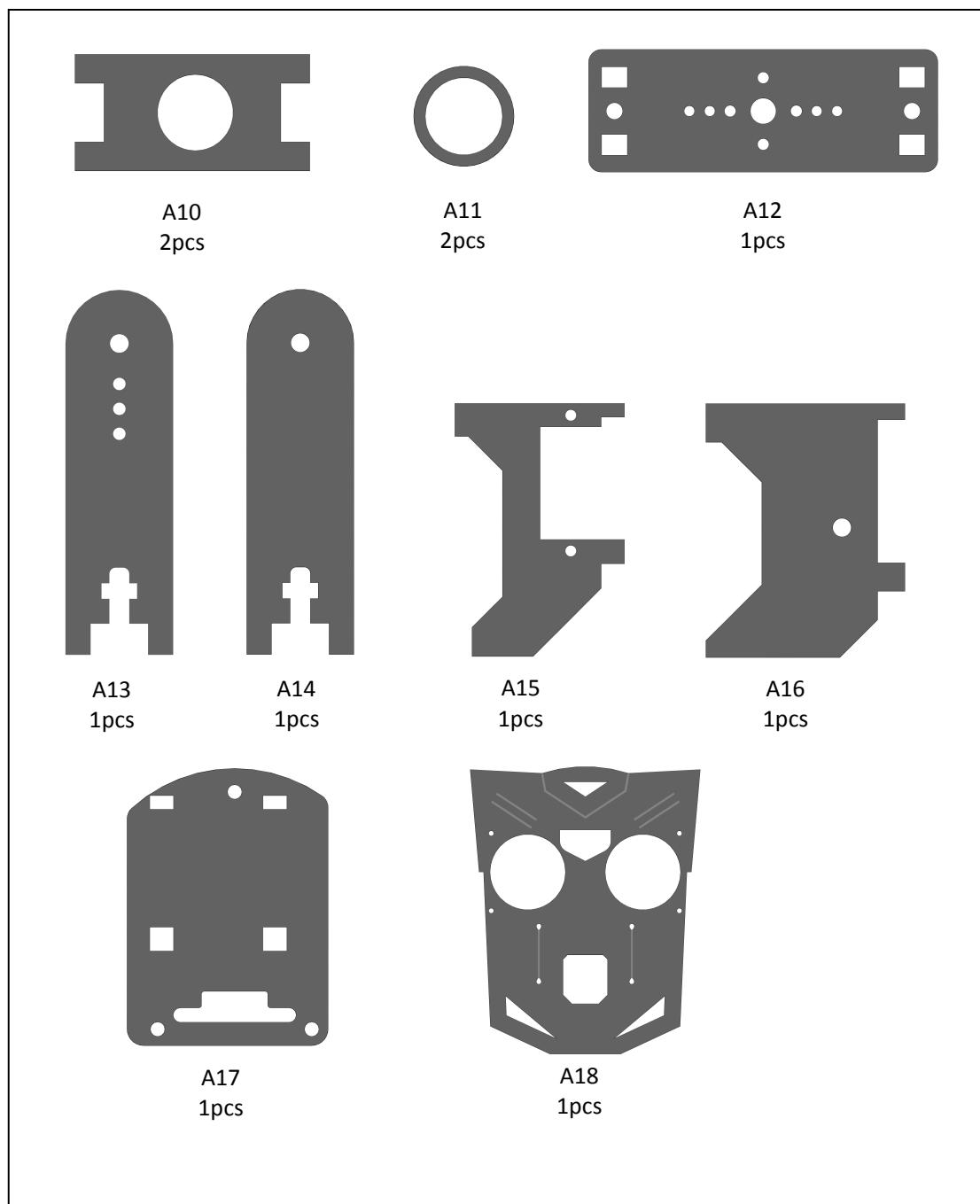
# contents

1. Components List.....	1
1.1. Acrylic Plates.....	1
1.2. Machinery Parts.....	3
1.3. Transmission Parts.....	4
1.4. Electronic Parts.....	5
1.5. Tools.....	6
1.6. Self-prepared Parts.....	6
2. Assembly.....	7
2.1. Preparations before Assembly.....	7
2.2. Car Light and Battery Holder Assembly.....	9
2.3. Rear wheel assembly.....	18
2.4. Front wheel assembly.....	26
2.5. Front part assembly.....	37
2.6. Raspberry Pi assembly.....	46
2.7. Install and Remove Batteries.....	50
2.8. Circuit Connection.....	51
3. Software & Hardware.....	52
3.1. Software Installation.....	52
3.2. Download Program.....	57
3.3. Set Up a Raspberry Pi.....	58
3.4. Install Python3.7 in the PC.....	66
3.5. Run the PiCar-B.....	71
4. Afterword.....	76

# 1. Components List

## 1.1. Acrylic Plates





The acrylic plates are fragile, so please be careful when assembling them in case of breaking.

The acrylic plate is covered with a layer of protective film. You need to remove it first.

Some holes in the acrylic may have residues, so you need to clean them before the use.

## 1.2. Machinery Parts

M2 Nut  X26 <a href="http://www.adeept.com">www.adeept.com</a>	M3 Nut  X12 <a href="http://www.adeept.com">www.adeept.com</a>	M4 Nut  X2 <a href="http://www.adeept.com">www.adeept.com</a>	M2*10 Screw  X11 <a href="http://www.adeept.com">www.adeept.com</a>	M2*14 Screw  X8 <a href="http://www.adeept.com">www.adeept.com</a>
M2.5*8 Screw  X8 <a href="http://www.adeept.com">www.adeept.com</a>	M3*8 Screw  X32 <a href="http://www.adeept.com">www.adeept.com</a>	M3*12 Screw  X9 <a href="http://www.adeept.com">www.adeept.com</a>	M3*18 Screw  X1 <a href="http://www.adeept.com">www.adeept.com</a>	M4*6 Screw  X2 <a href="http://www.adeept.com">www.adeept.com</a>
M4*40 Screw  X2 <a href="http://www.adeept.com">www.adeept.com</a>	M3*10 Countersunk Head Screw  X2 <a href="http://www.adeept.com">www.adeept.com</a>	M1.4*6 Self-tapping Screw  X8 <a href="http://www.adeept.com">www.adeept.com</a>	M4 Spring Washer  X6 <a href="http://www.adeept.com">www.adeept.com</a>	F624ZZ Bearing  X6 <a href="http://www.adeept.com">www.adeept.com</a>
F687ZZ Bearing  X4 <a href="http://www.adeept.com">www.adeept.com</a>	M2*6 Copper Standoff  X4 <a href="http://www.adeept.com">www.adeept.com</a>	M2.5*10+6 Copper Standoff  X4 <a href="http://www.adeept.com">www.adeept.com</a>	M2.5*14 Copper Standoff  X4 <a href="http://www.adeept.com">www.adeept.com</a>	M3*12 Copper Standoff  X4 <a href="http://www.adeept.com">www.adeept.com</a>
M3*30 Copper Standoff  X11 <a href="http://www.adeept.com">www.adeept.com</a>	Nylon Isolation column  X2 <a href="http://www.adeept.com">www.adeept.com</a>			

## 1.3. Transmission Parts

Bevel gear unit



Bevel gear x2



M3\*3 Locking Screw x2

S12D4 Coupling Set



S12D4 Coupling x2

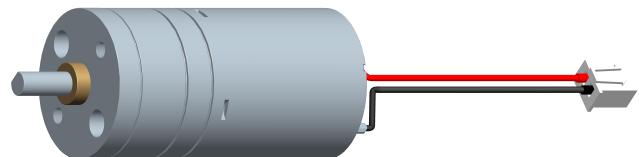


M4\*4 Locking Screw x4

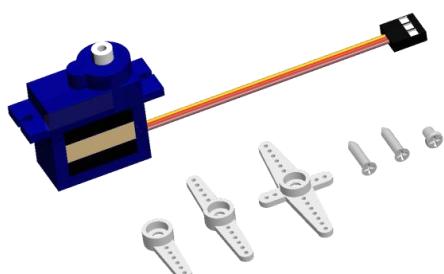
D3.9L120 Axle x1



Motor x1



Servo x3

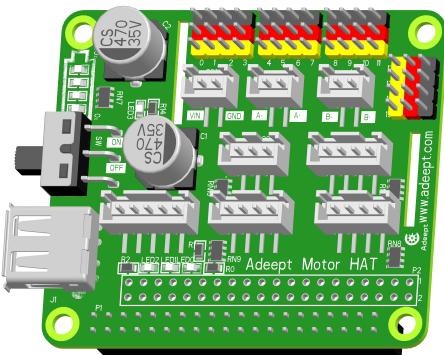


Wheel x4

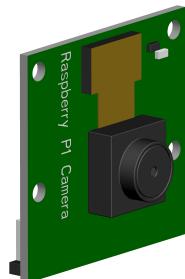


## 1.4. Electronic Parts

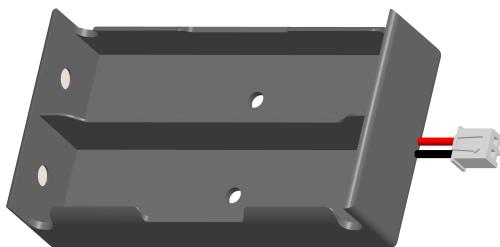
Adeept Motor HAT X1



Raspberry Pi Camera X1



18650x2 Battery Holder X1



Adeept Ultrasonic Module X1



Adeept RGB LED Module X2



3-CH WS2812 RGB LED X4



3-CH Line Tracking Module X1



Voice Module X1



Raspberry P1 Camera Ribbon X1



3-Pin Wires -A X3



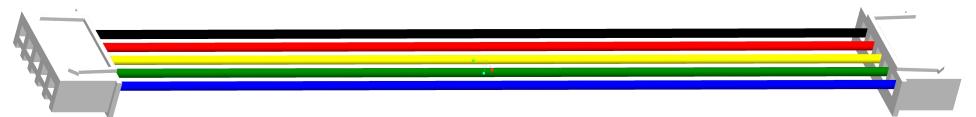
3-Pin Wires -B X1



4-Pin Wires X3



5-Pin Wires X1



## 1.5. Tools

Hex Wrench-1.5mm X1



Hex Wrench-2.0mm X1



Cross Screwdriver X1



Cross Socket Wrench X1



Ribbon X1



Large Cross-head Screwdriver X1

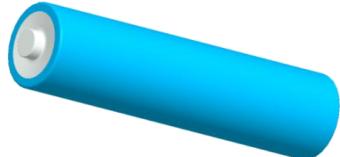


Winding Pipe X1

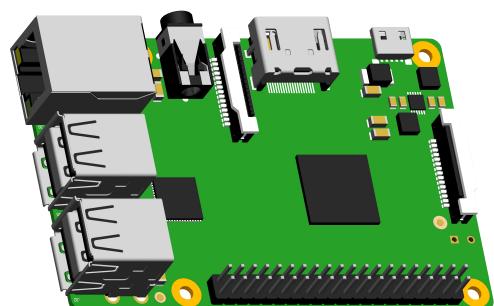


## 1.6. Self-prepared Parts

18650 Battery X2



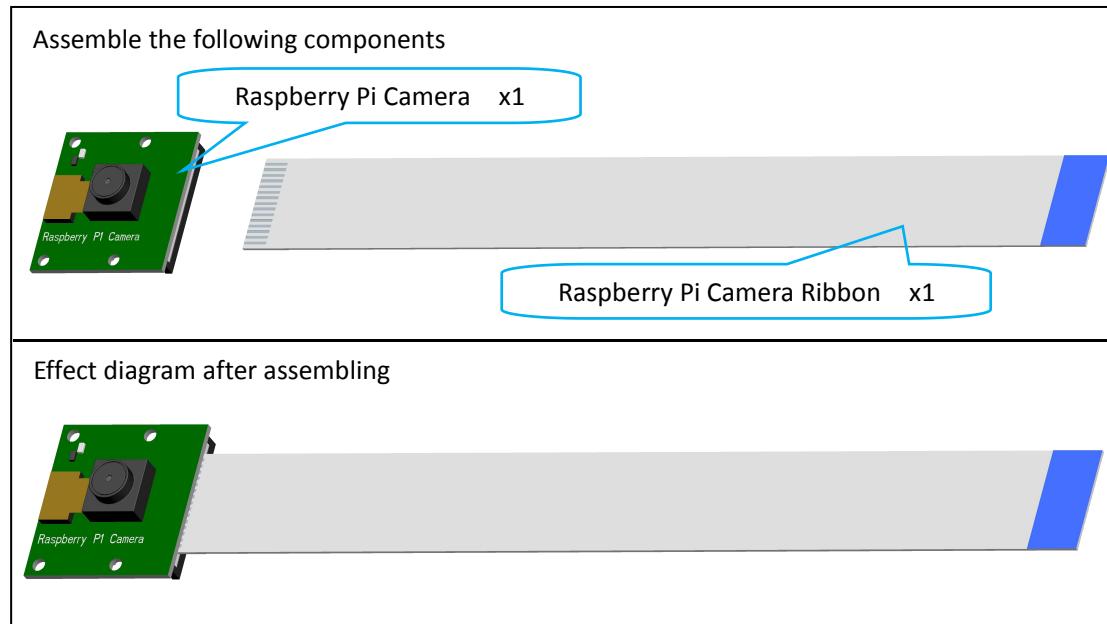
Raspberry Pi X1



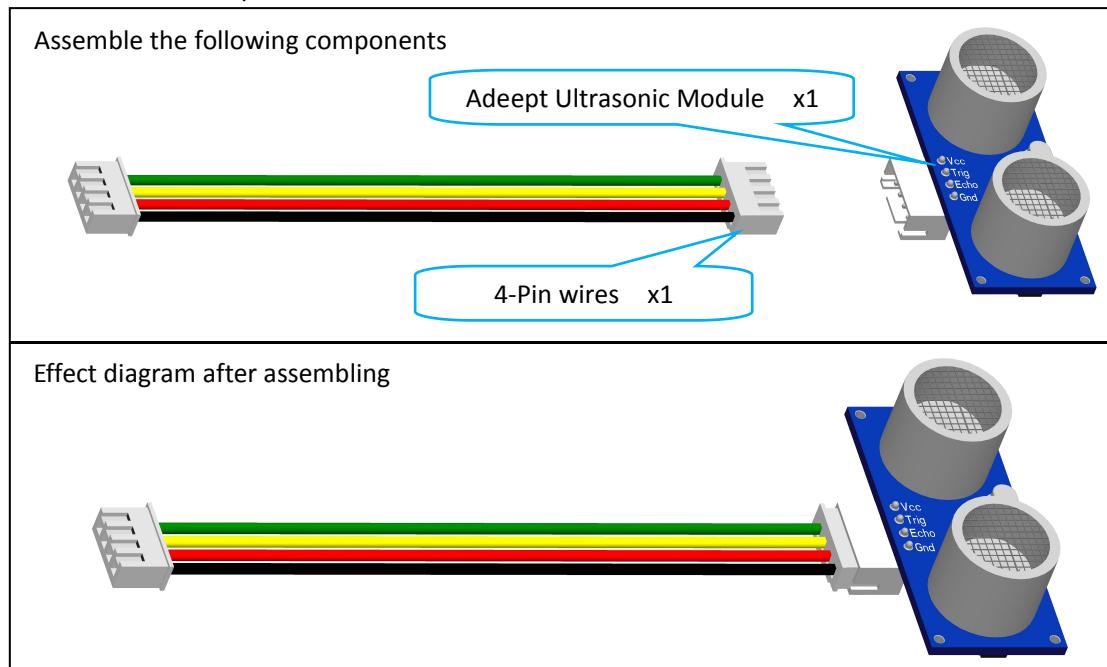
## 2. Assembly

### 2.1. Preparations before Assembly

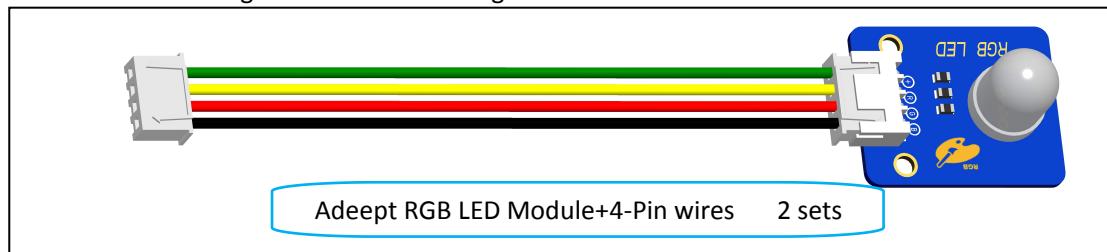
1. Connect the Raspberry Pi Camera and the ribbon.

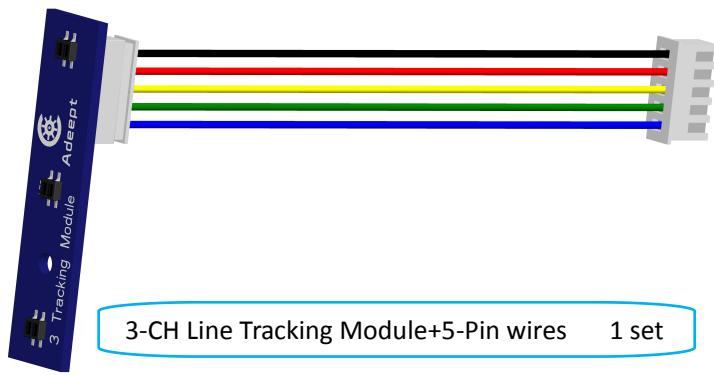


2. Connect the Adeept Ultrasonic Module with 4-Pin wires.



The connection diagrams of the remaining modules and wires are as follows:





The two plugs of 3-Pin Wires-A are small plugs, 3-Pin Wires-B has a small plug at one end and a large plug at the other end. To prevent confusion between the following two components when reading the manual, we define the following two components as 3-CH WS2812 RGB LED-A and 3-CH WS2812 RGB LED-B.



3-CH WS2812 RGB LED+3-Pin Wires -A 3 sets  
Defined 3-CH WS2812 RGB LED-A

Wires are connected to the input of 3-CH WS2812 RGB LED (the end marked with a white strip pattern)

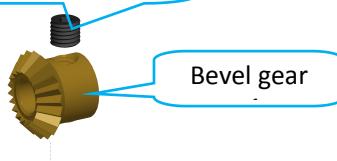


3-CH WS2812 RGB LED+3-Pin Wires -B 1 set  
Defined as 3-CH WS2812 RGB LED-B

### 3. Screw the M3\*3 Locking Screw into the bevel gear (2 sets).

Assemble the following components

M3\*3 Locking Screw x1



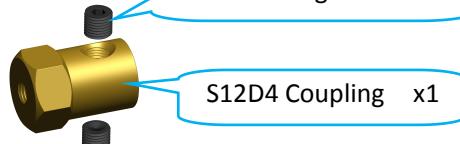
Effect diagram after assembling



### 4. Screw the M4\*4 Locking Screw into the S12D4 Coupling (2 sets).

Assemble the following components

M4\*4 Locking Screw x2



Effect diagram after assembling

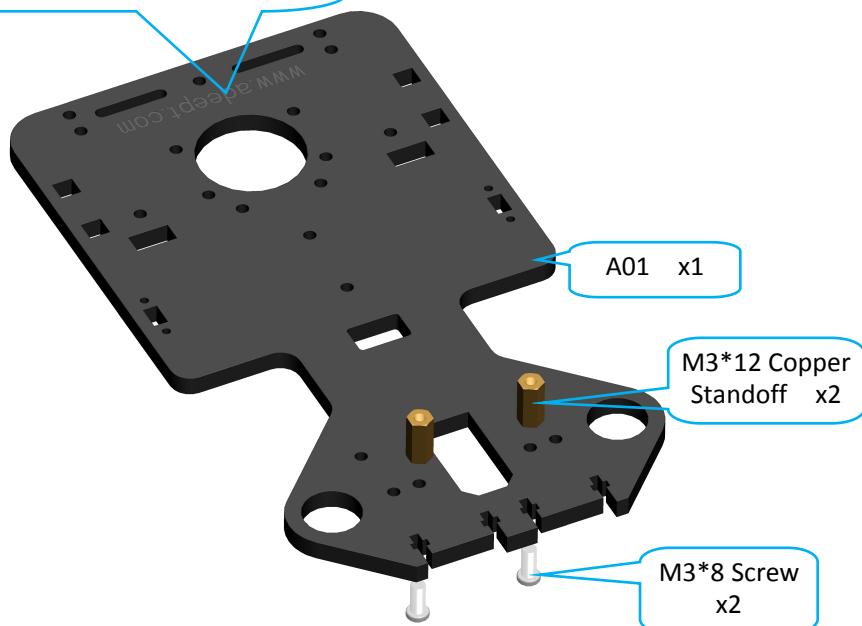


## 2.2. Car Light and Battery Holder Assembly

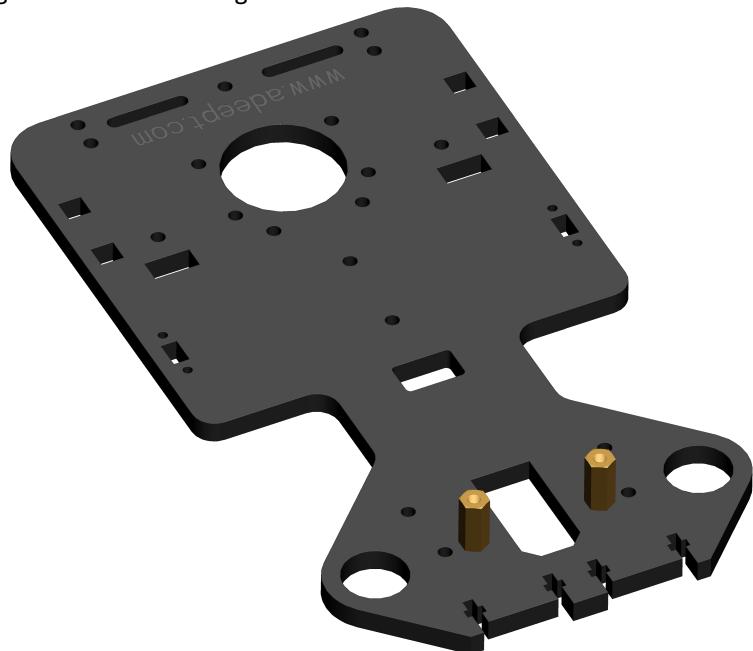
1. Fix two M3\*12 Copper Standoff on A01.

Assemble the following components

M3\*12 Copper Standoff is fixed on the marked side of A01

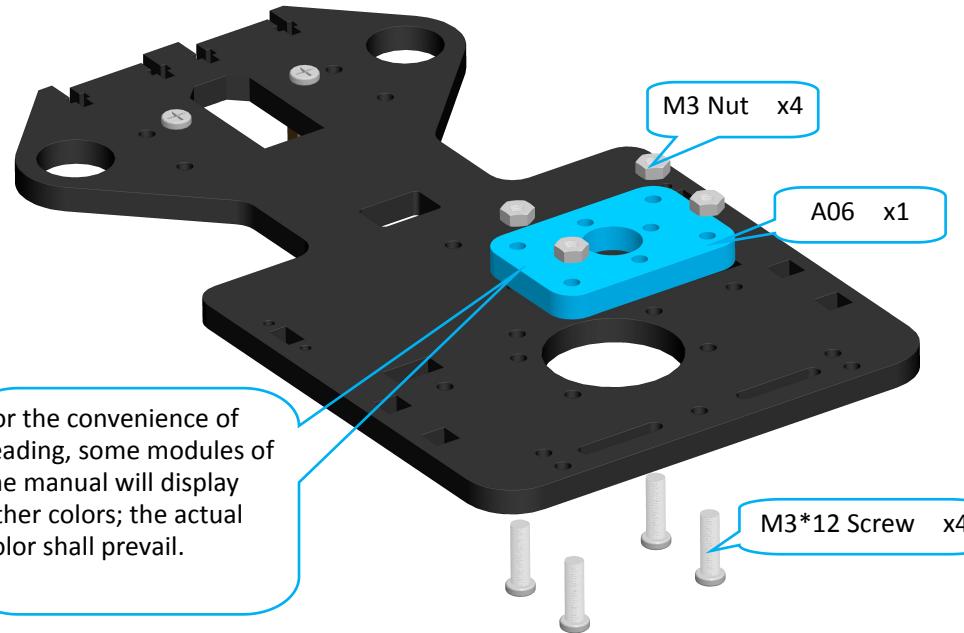


Effect diagram after assembling

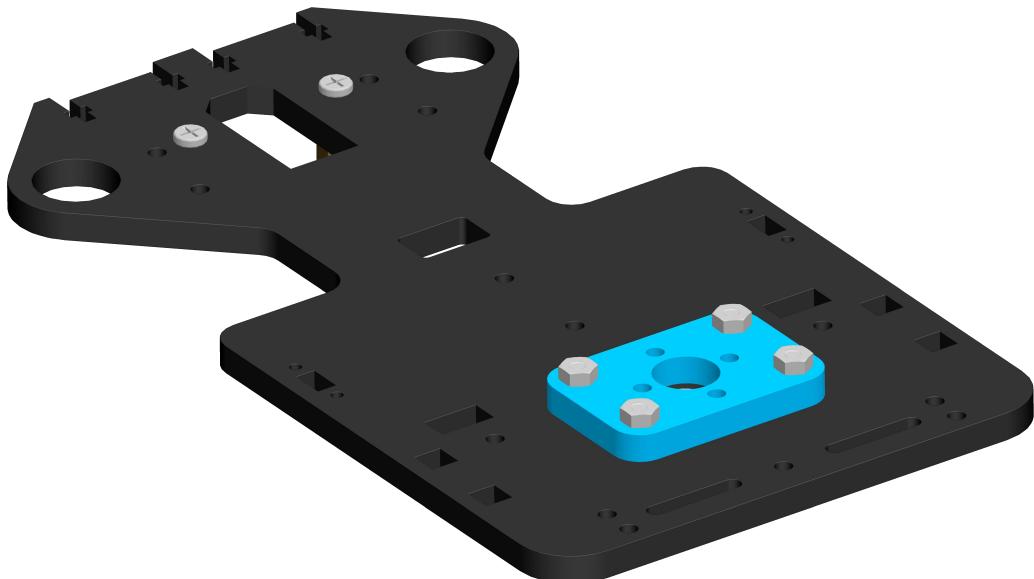


## 2. Fix A06 on A01.

Assemble the following components

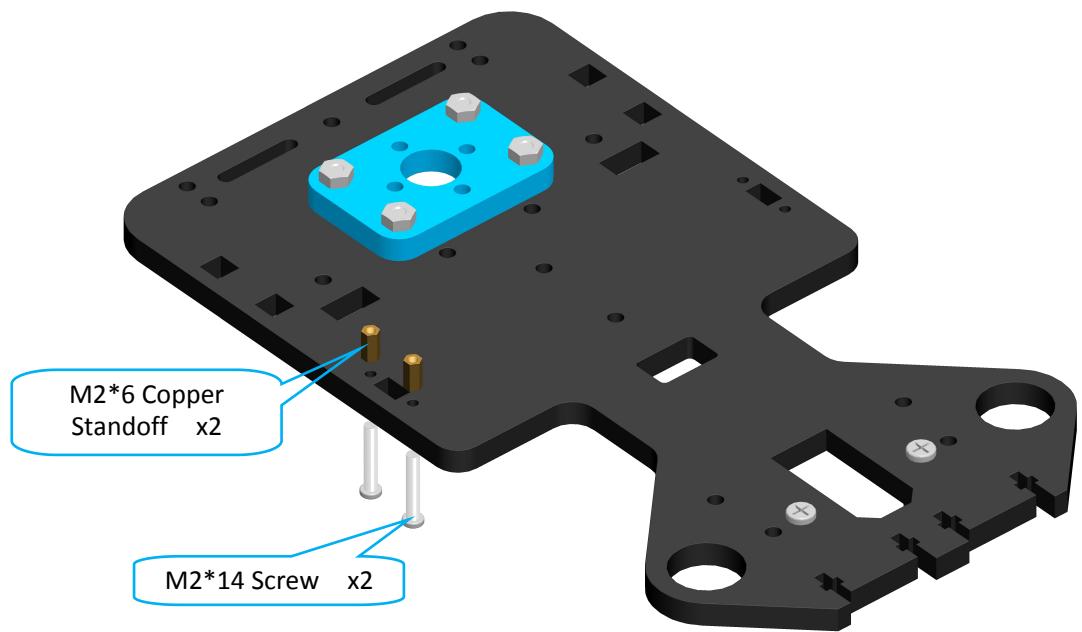


Effect diagram after assembling

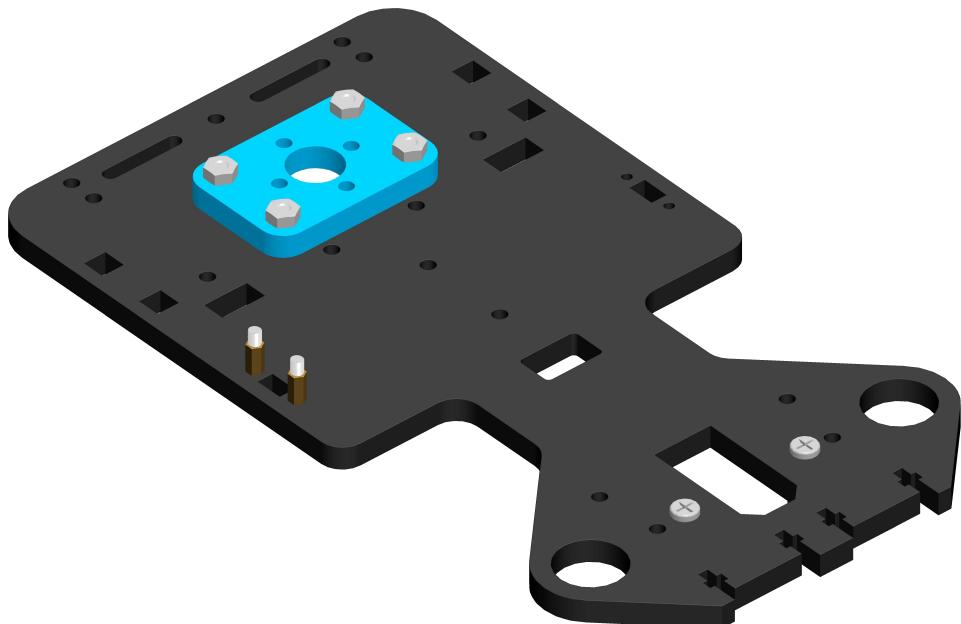


3. Fix M2\*6 Copper Standoff on A01.

Assemble the following components

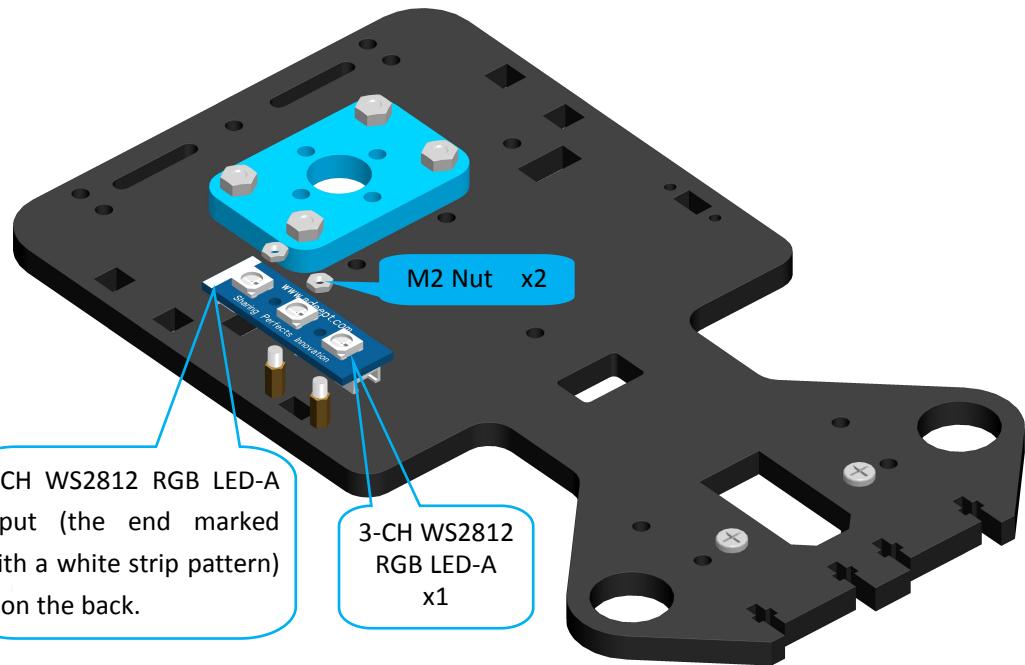


Effect diagram after assembling

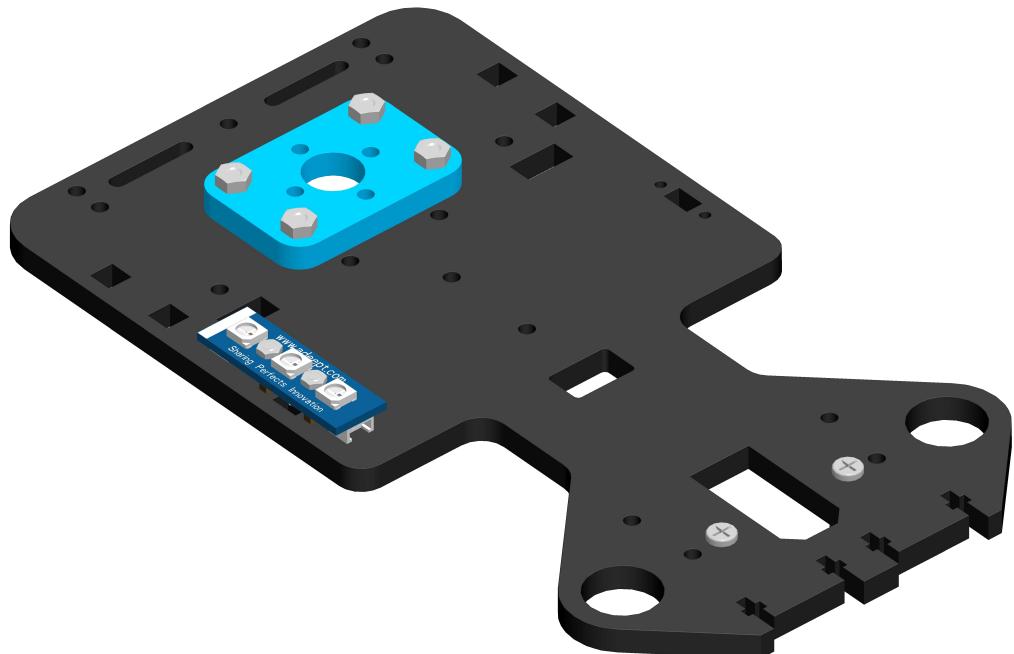


4. Fix one 3-CH WS2812 RGB LED-A on M2\*6 Copper Standoff.

Assemble the following components



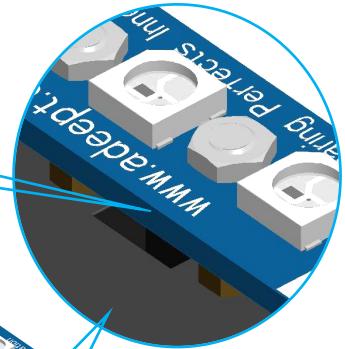
Effect diagram after assembling



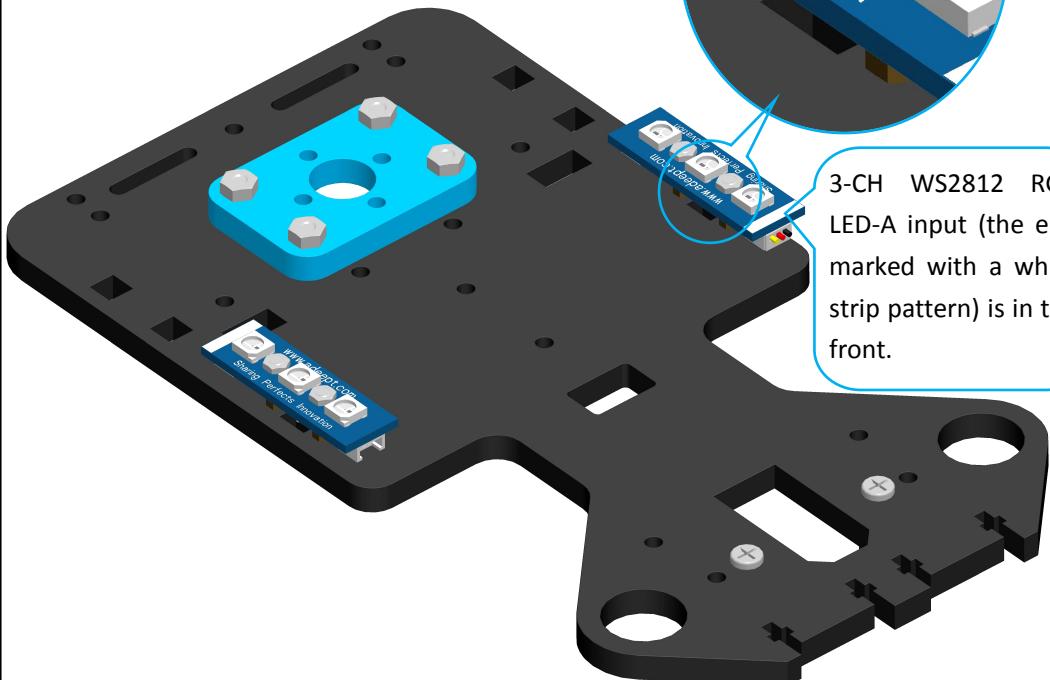
5. Follow step 3 and 4, fix another 3-CH WS2812 RGB LED-A on A01.

Effect diagram after assembling

Before install the 3-CH WS2812 RGB LED-A,  
pass the 3-Pin Wires-A on the 3-CH WS2812  
RGB LED-A through A01 via the hole here.

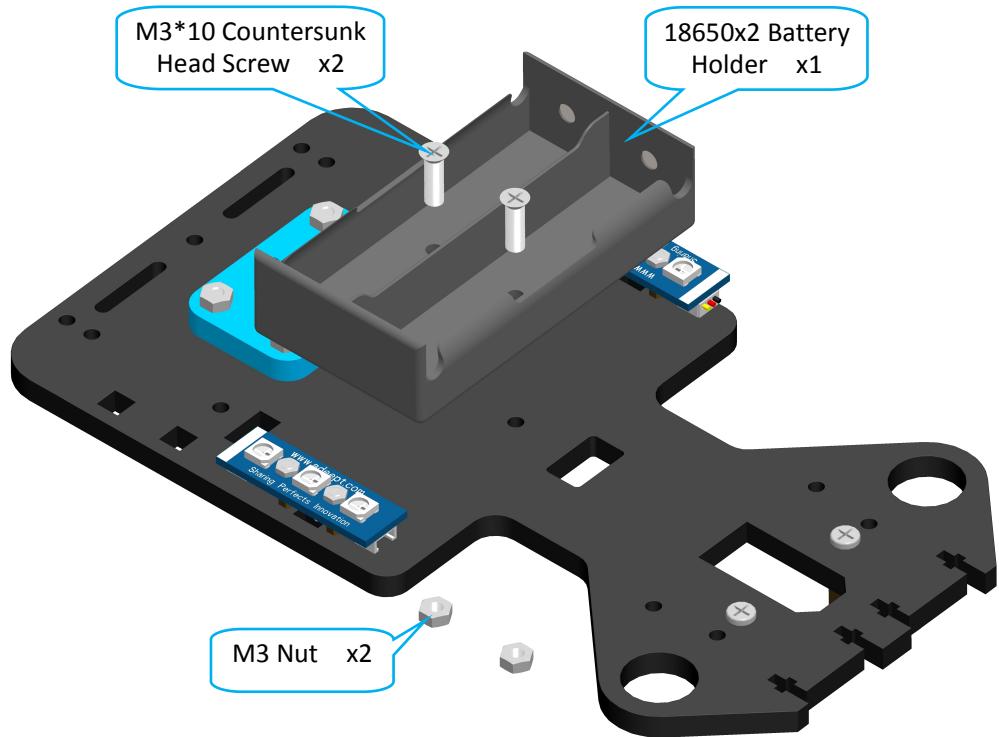


3-CH WS2812 RGB  
LED-A input (the end  
marked with a white  
strip pattern) is in the  
front.



6. Fix 18650x2 Battery Holder on A01.

Assemble the following components



Effect diagram after assembling

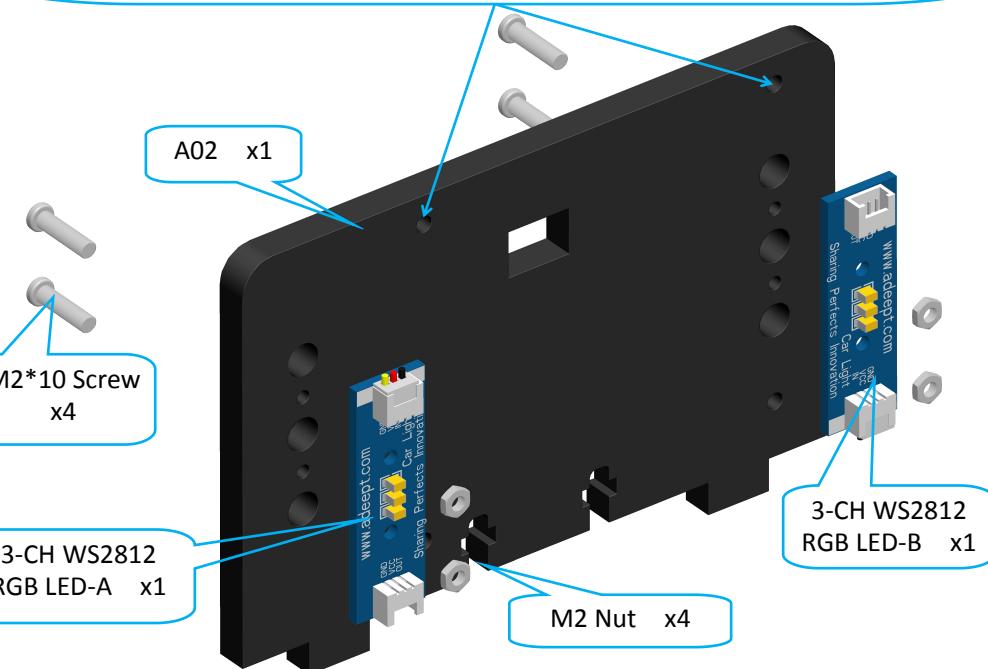
After the 18650x2 Battery Holder is fixed, connect the 3-Pin Wires -A of 3-CH WS2812 RGB LED-A on the left to the 3-CH WS2812 RGB LED-A on the right.



7. Fix one 3-CH WS2812 RGB LED-A and 3-CH WS2812 RGB LED-B respectively on both sides of the A02 .

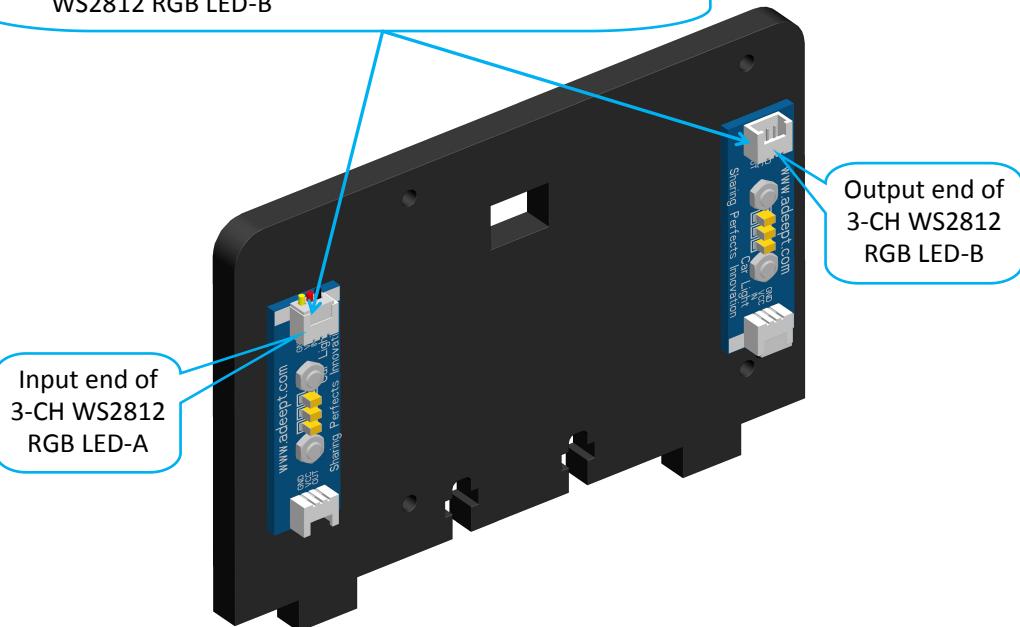
Assemble the following components

With reference to the position of the two holes here (to the right), 3-CH WS2812 RGB LED-A and 3-CH WS2812 RGB LED-B cannot be fixed to the other side of A02. And 3-CH WS2812 RGB LED-A is on the left side, the input end (the end marked with a white strip pattern) faces up, 3-CH WS2812 RGB LED-B is on the right side, and the input end (the end marked with a white strip pattern) faces down.



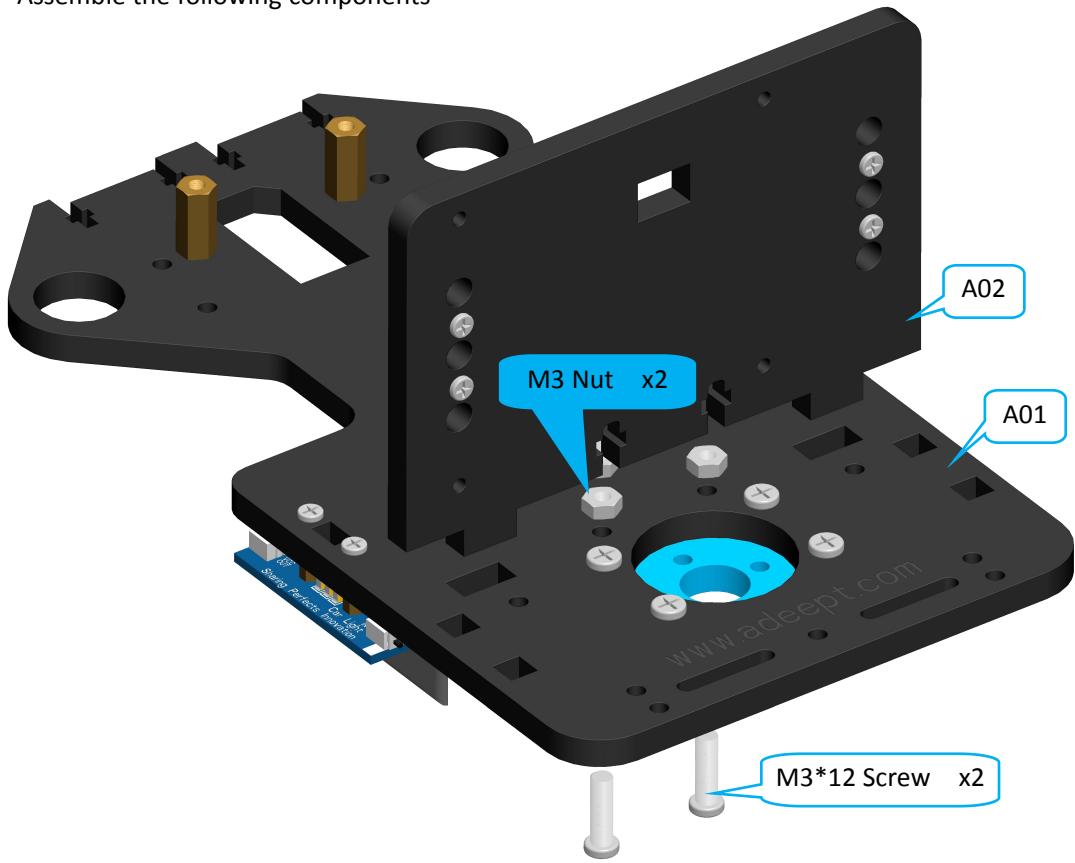
Effect diagram after assembling

After assembly, connect the wires on the input end of 3-CH WS2812 RGB LED-A to the output end of 3-CH WS2812 RGB LED-B

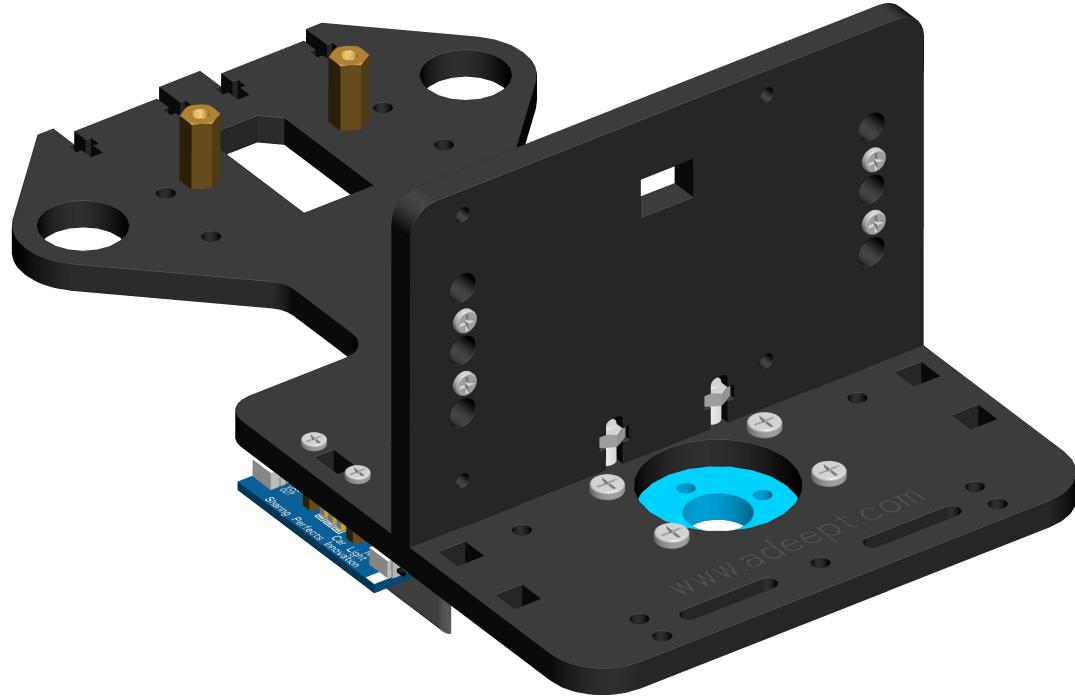


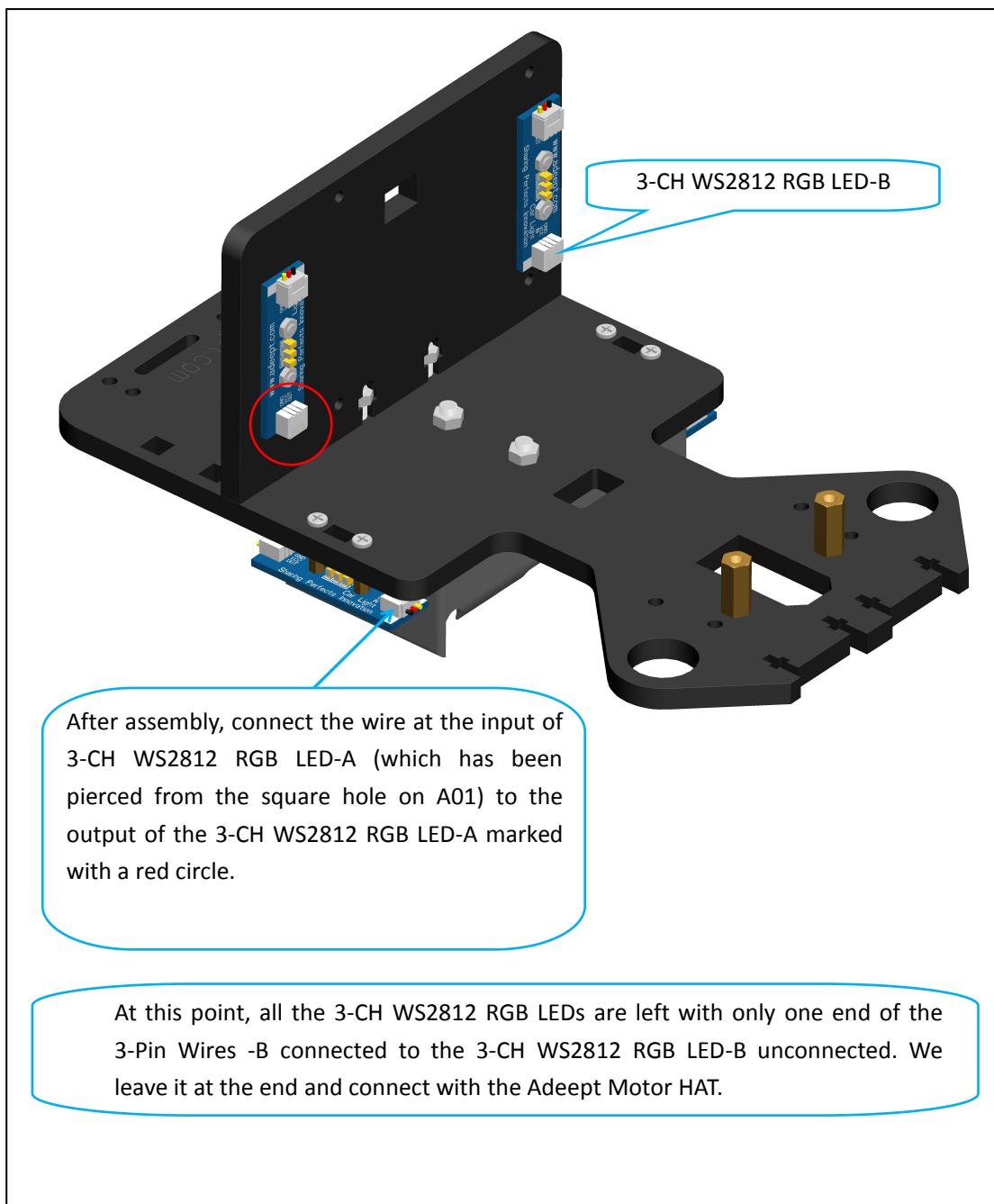
8. Fix A02 on A01.

Assemble the following components



Effect diagram after assembling

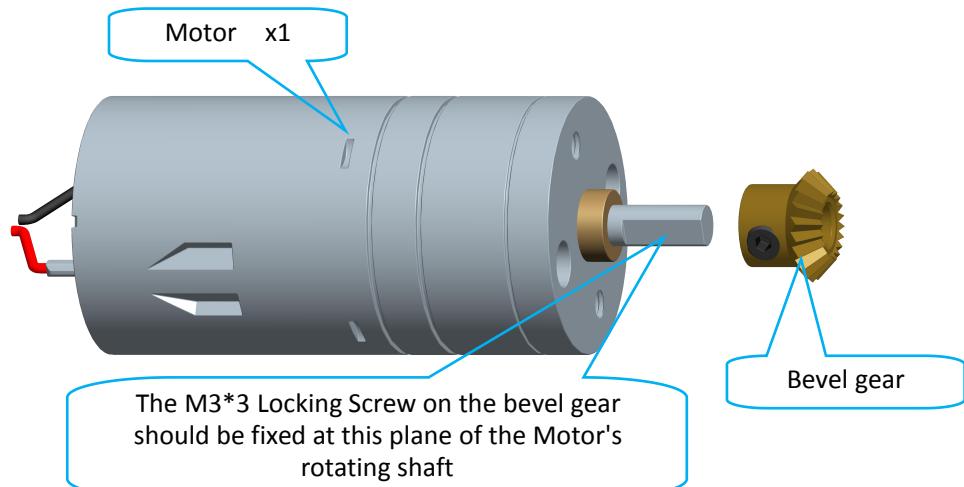




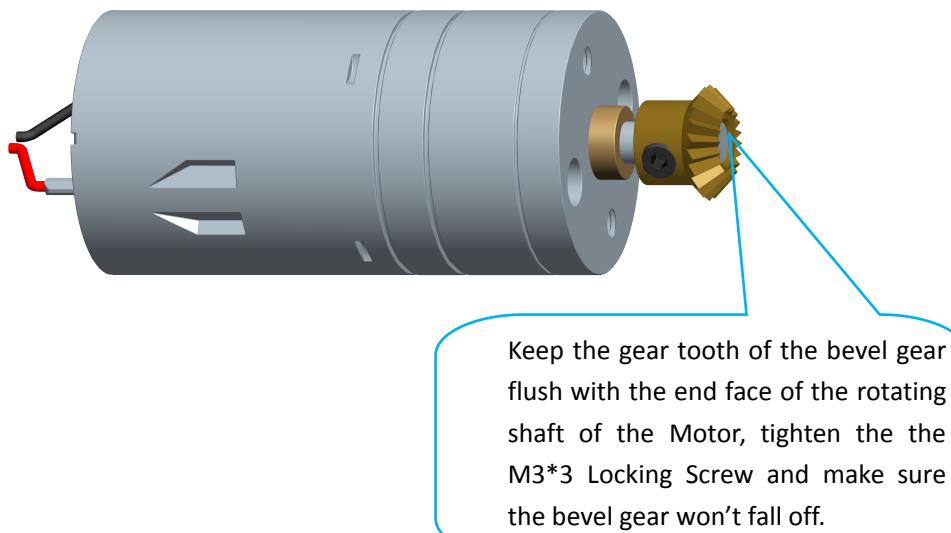
## 2.3. Rear wheel assembly

1. Fix the bevel gear on the Motor.

Assemble the following components

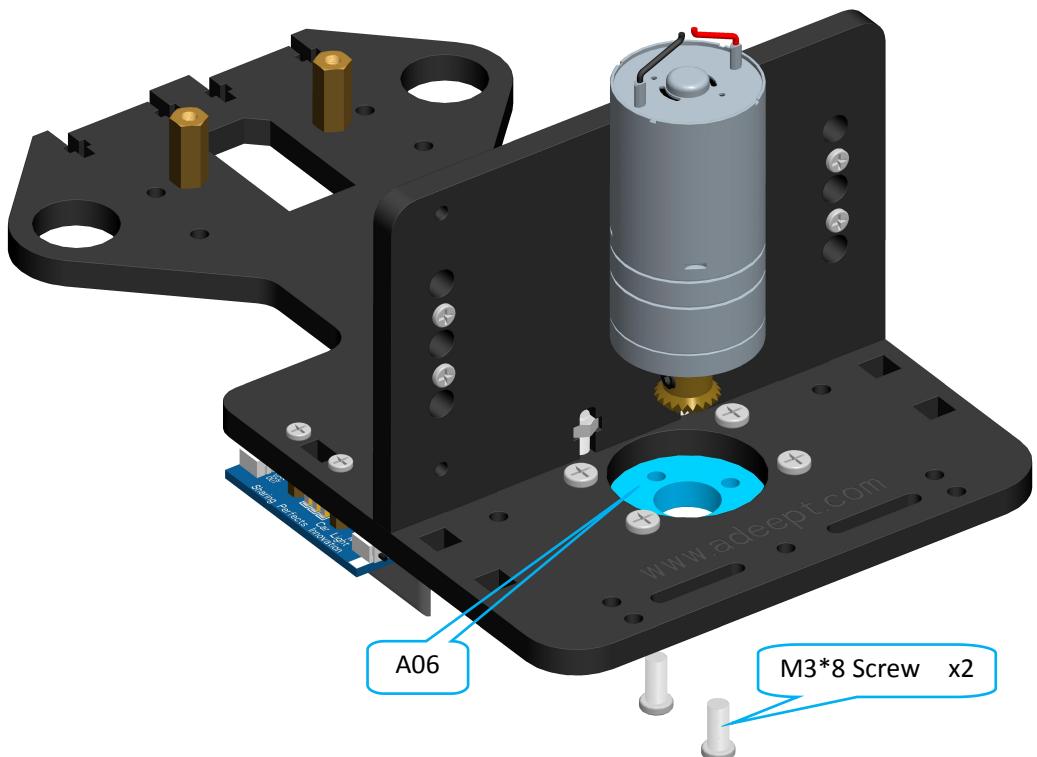


Effect diagram after assembling



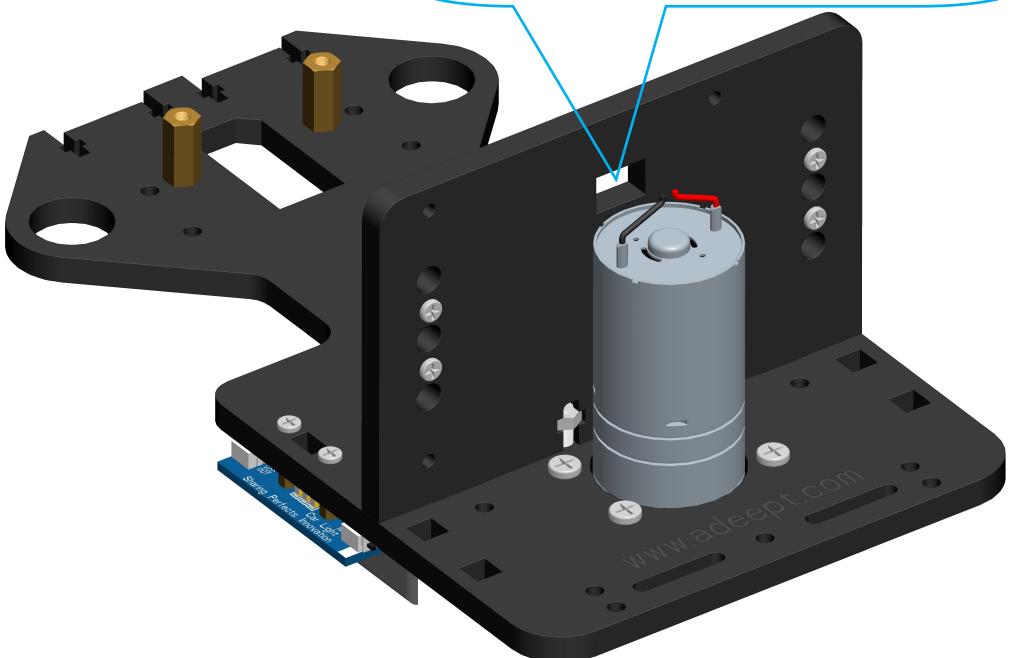
## 2. Fix Motor on A06.

Assemble the following components



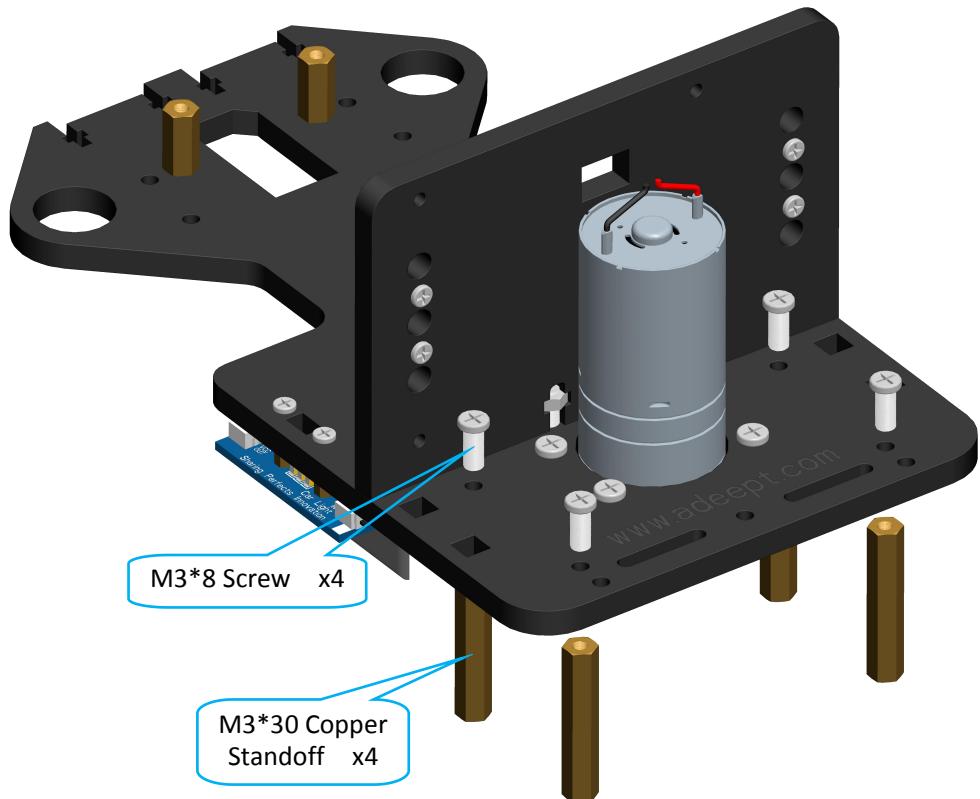
Effect diagram after assembling

Once assembled, the wires on the Motor can be passed through this hole in the A02.

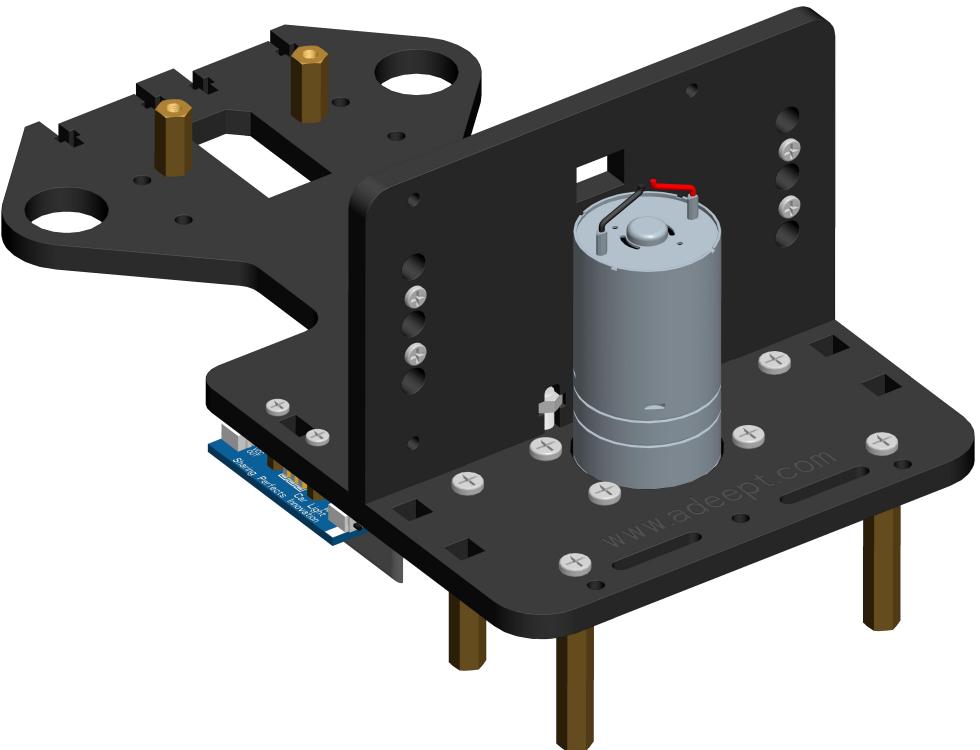


3. Fix four M3\*30 Copper Standoff on A01.

Assemble the following components

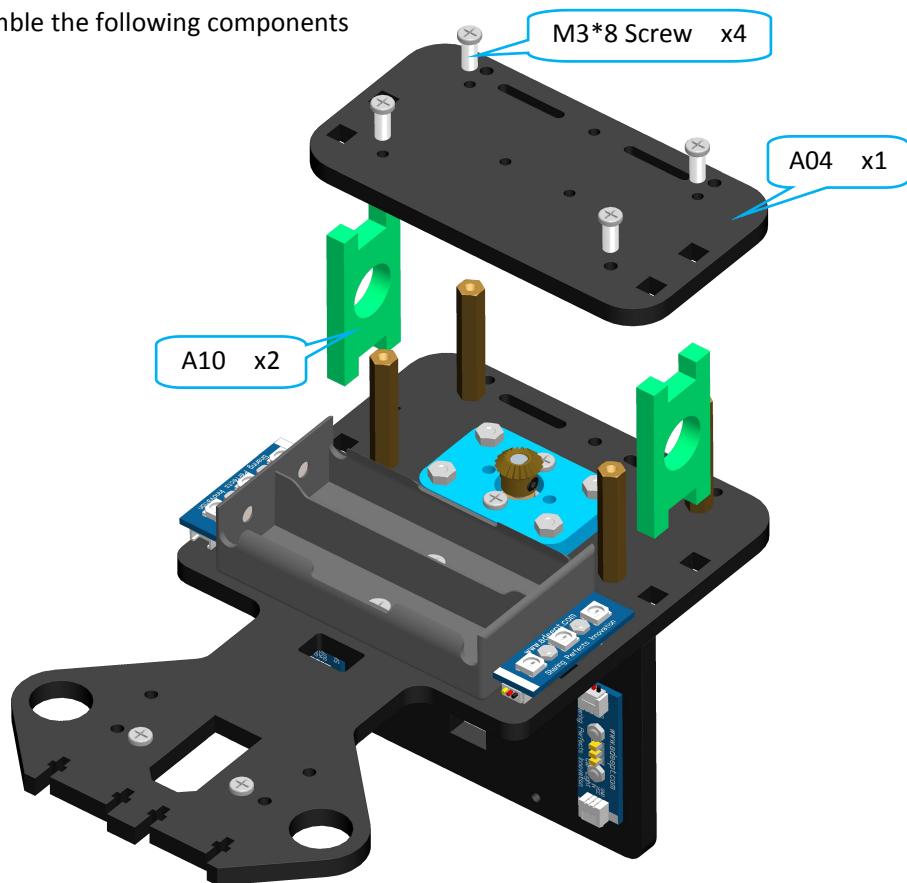


Effect diagram after assembling

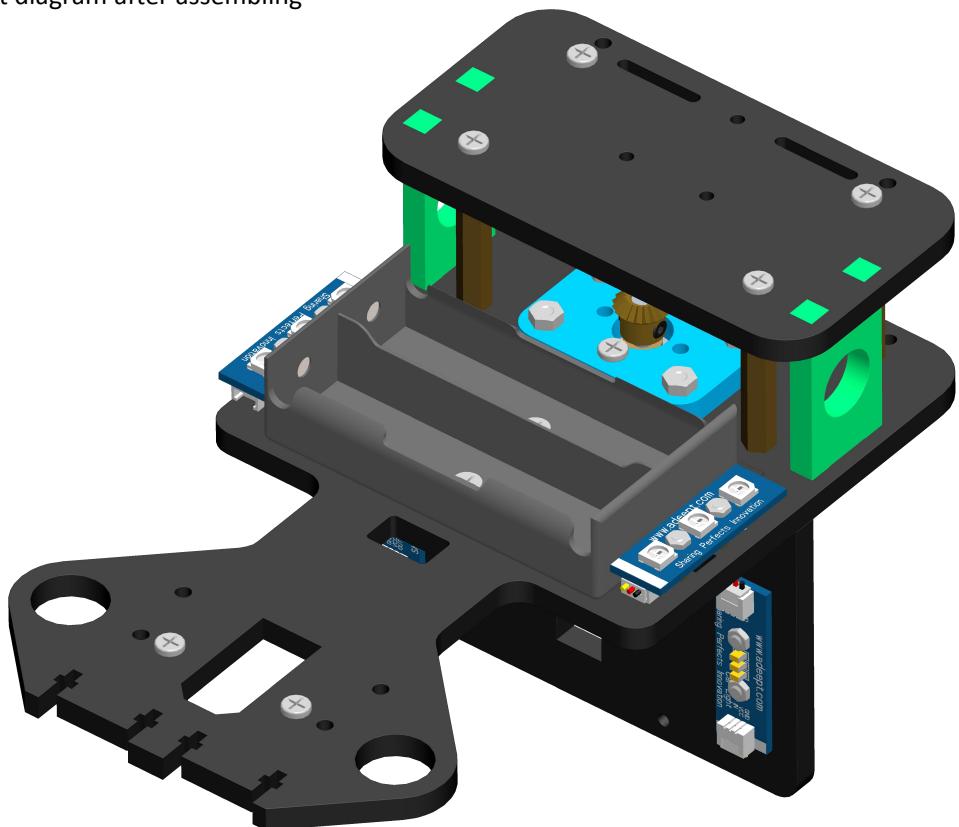


4. Then fix the A04 on the M3\*30 Copper Standoff.

Assemble the following components

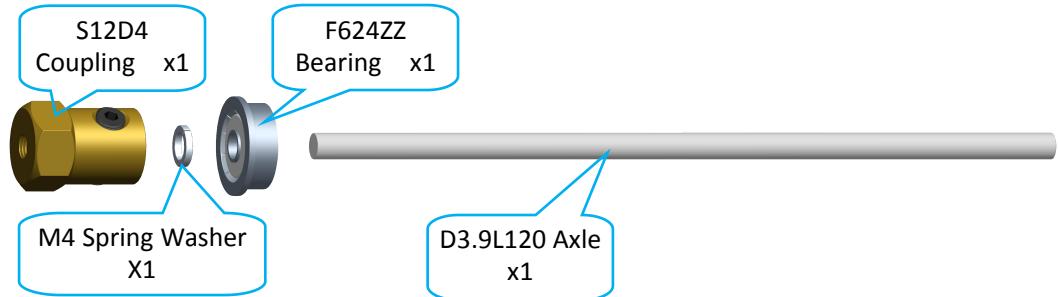


Effect diagram after assembling



5. Take one S12D4 Coupling and fix it on the D3.9L120 Axle.

Assemble the following components



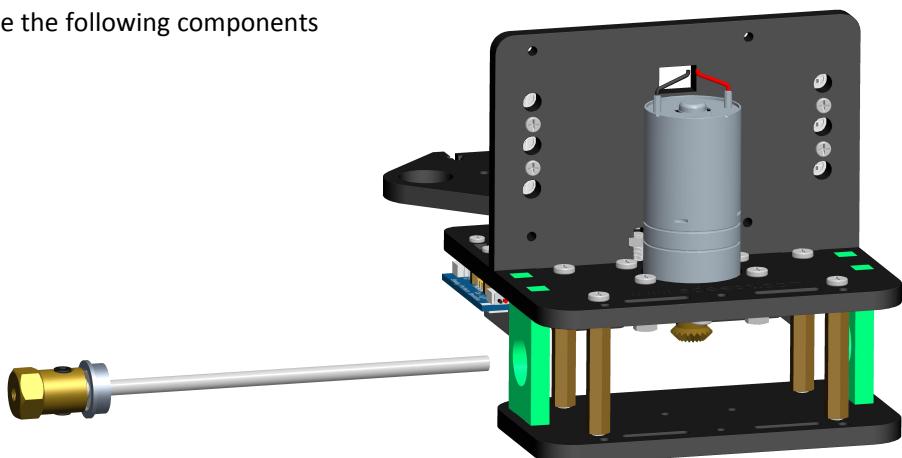
Effect diagram after assembling

Insert D3.9L120 Axle to the bottom end of the S12D4 Coupling shaft hole and tighten the M4\*4 Locking Screw on the S12D4 Coupling

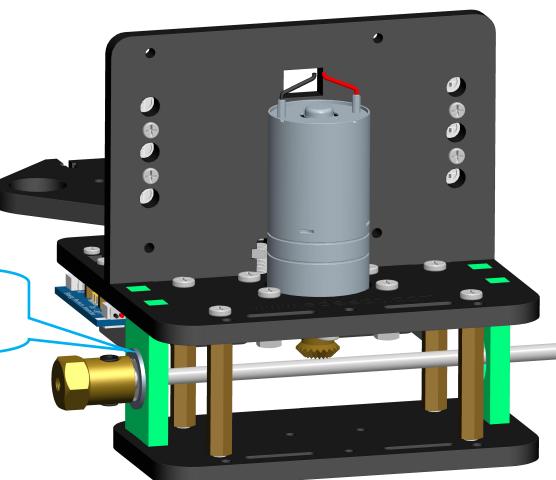


6. Then pass the D3.9L120 Axle through the A10.

Assemble the following components

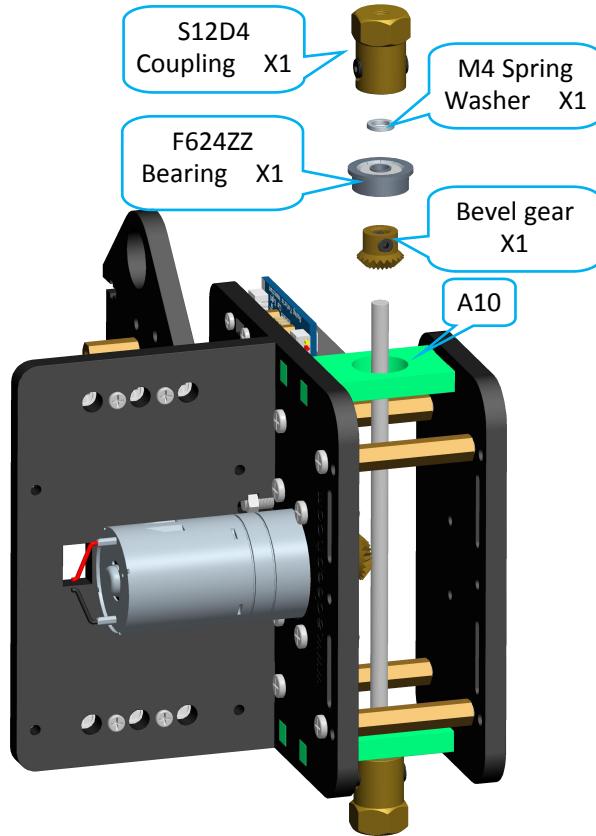


Effect diagram after assembling



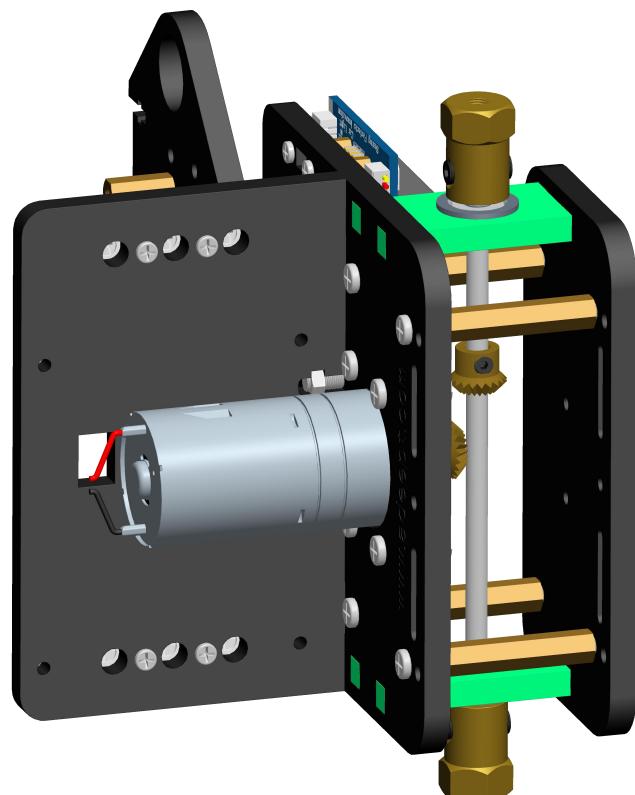
7. Take another S12D4 Coupling and fix it on the D3.9L120 Axle.

Assemble the following components



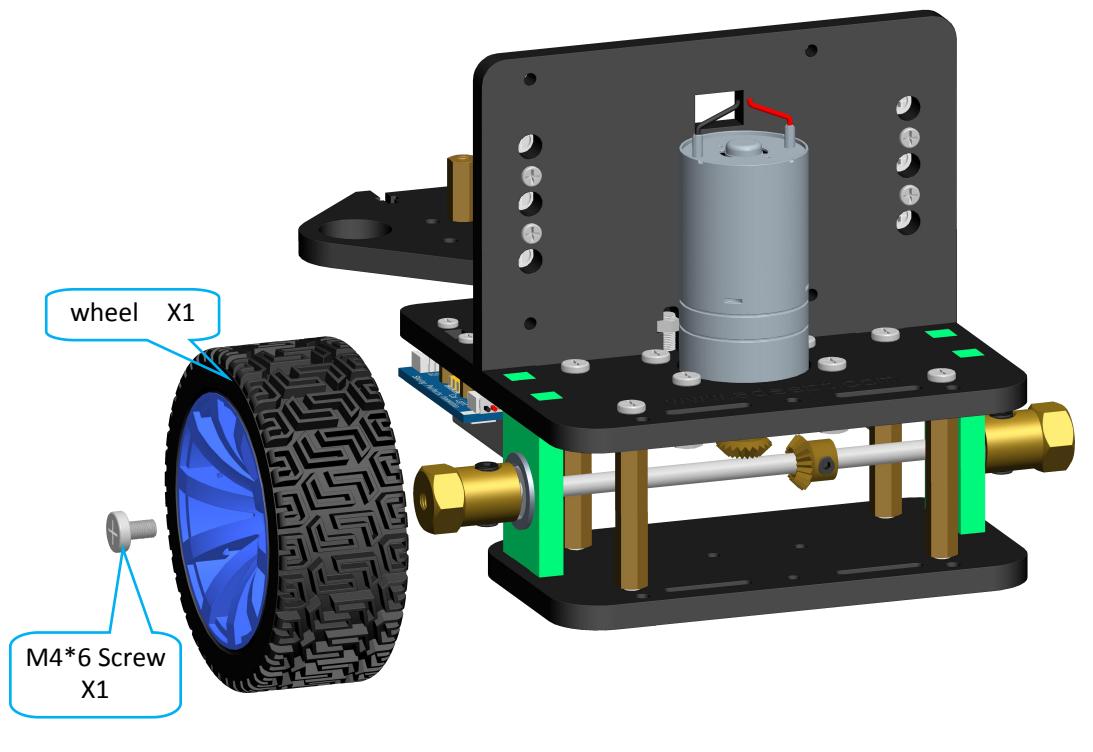
When installing, pay attention to that the tooth surface of the bevel gear should be facing down, and the bevel gear is above the motor shaft. At this time, no need to fix the bevel gear. The F624ZZ Bearing should be embedded in the A10. When fixing the S12D4 Coupling, put the lower S12D4 Coupling against the table top. Then, press the upper S12D4 Coupling down by hand, and then tighten the M4\*4 Locking Screw on the S12D4 Coupling.

Effect diagram after assembling

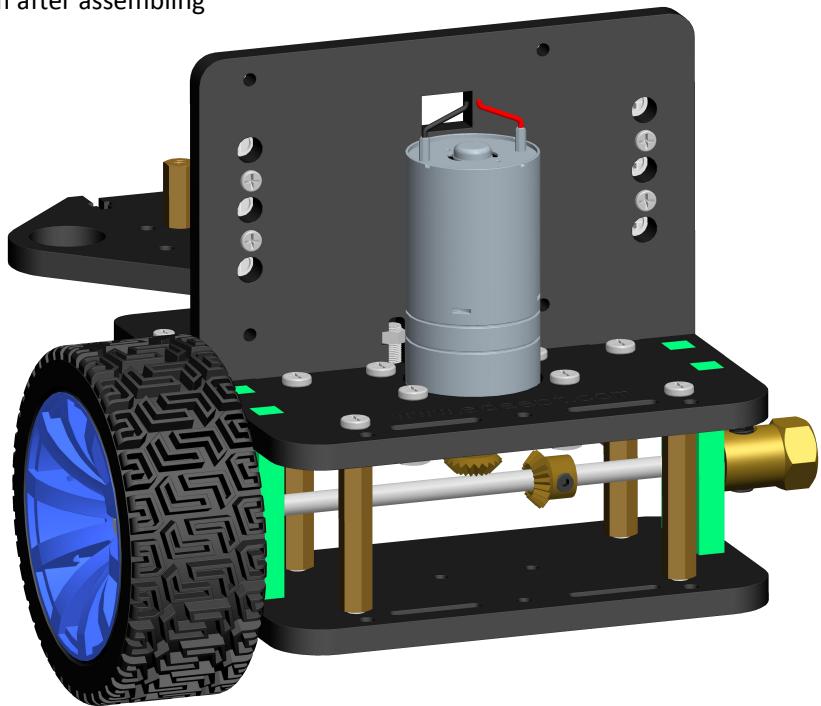


8. Fix the tire on the S12D4 Coupling.

Assemble the following components

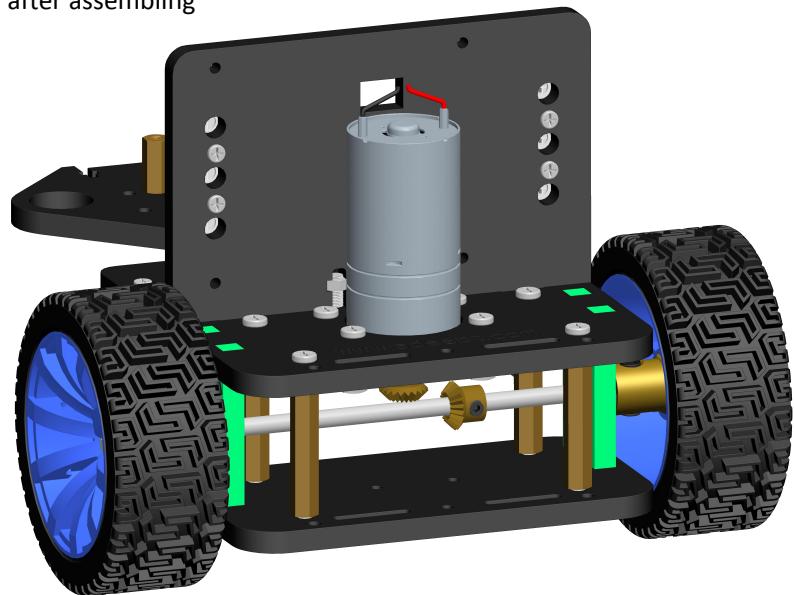


Effect diagram after assembling



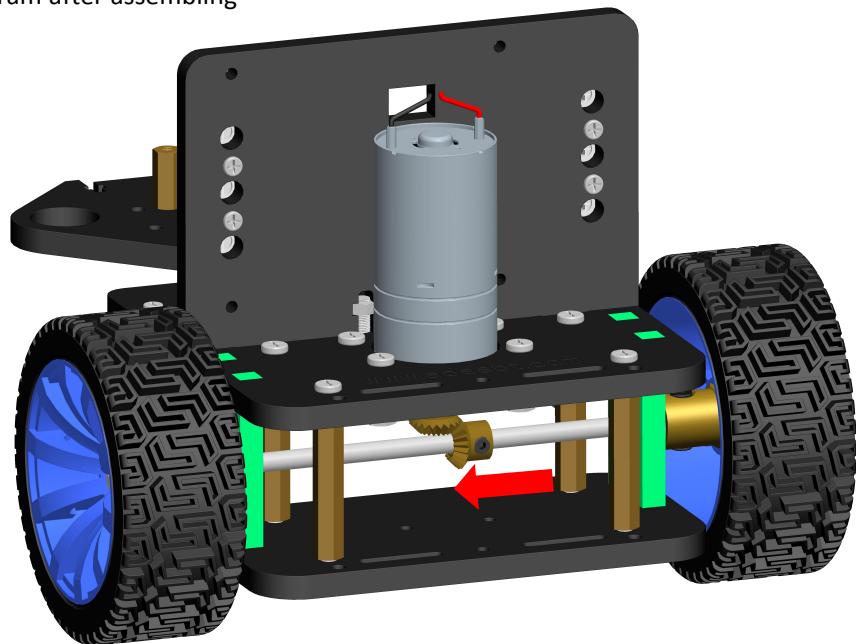
9. In the same way, fix another S12D4 Coupling to a tire..

Effect diagram after assembling



10. Fix the bevel gear on the D3.9L120 Axle.

Effect diagram after assembling

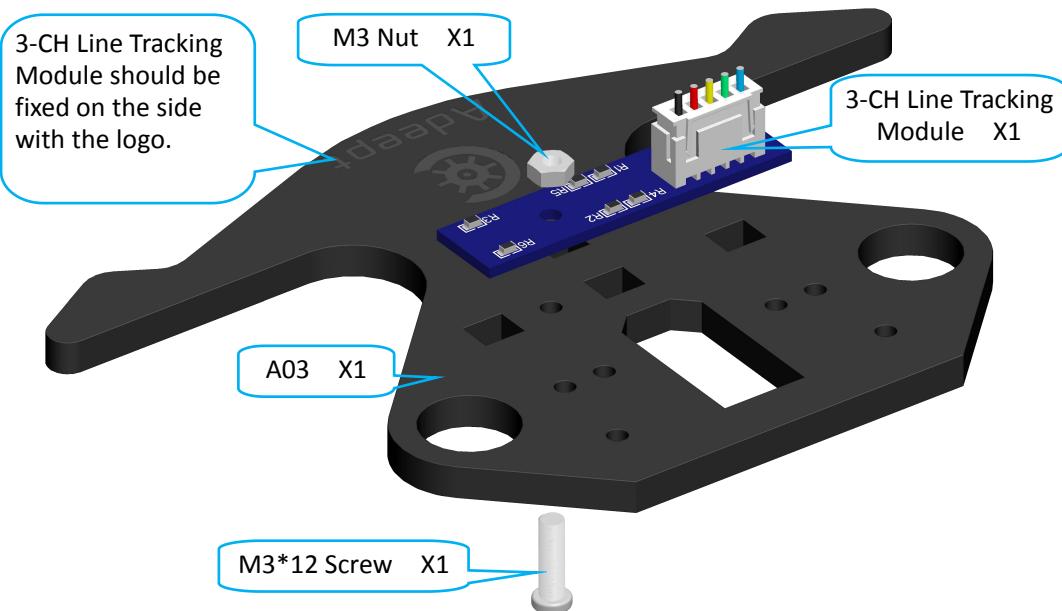


Push the bevel gear on the D3.9L120 Axle from right to left with one hand until it engages with the bevel gear on the Motor, and then tighten the M3\*3 Locking Screw of the bevel gear on the D3.9L120 Axle with the other hand to fix the bevel gear.

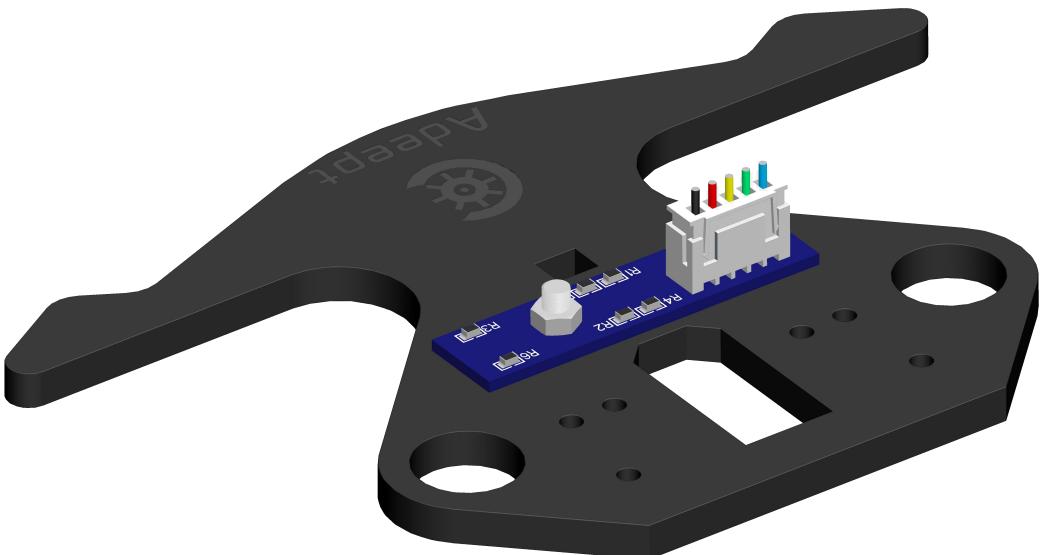
## 2.4. Front wheel assembly

1. Fix 3-CH Line Tracking Module on A03.

Assemble the following components

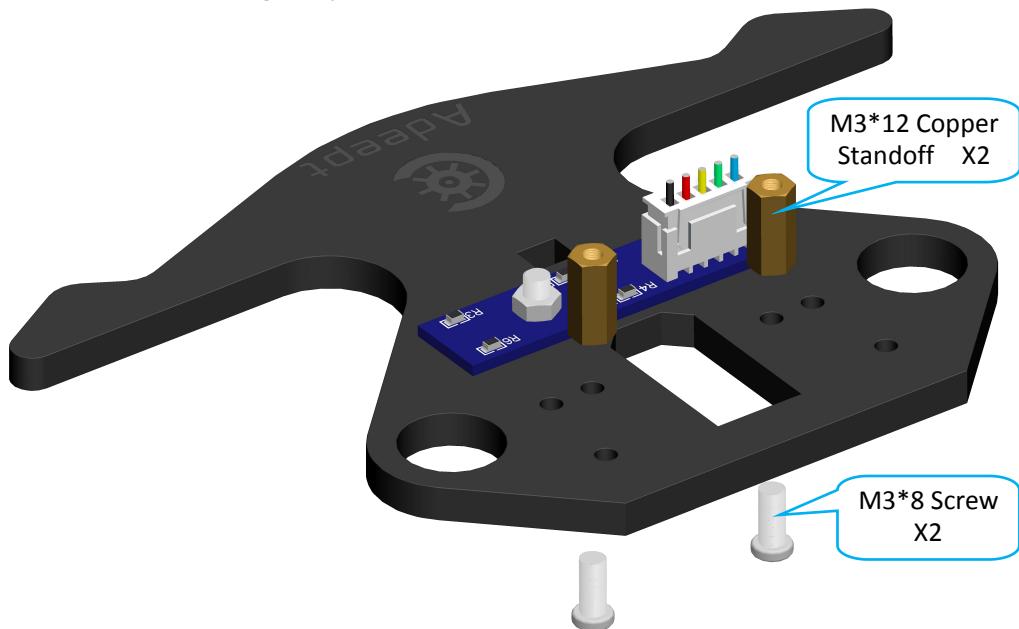


Effect diagram after assembling

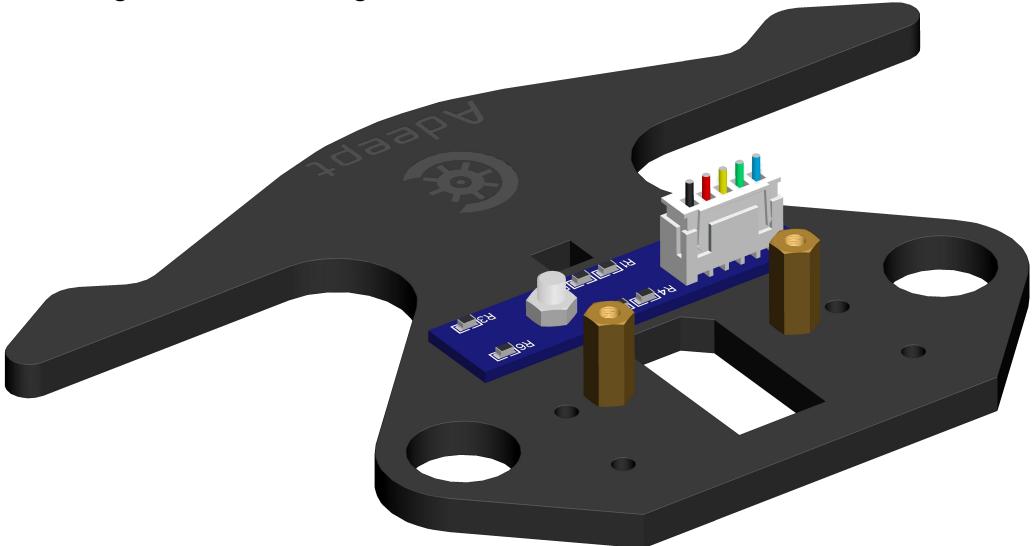


2. Fix two 3\*12 Copper Standoff on A03.

Assemble the following components

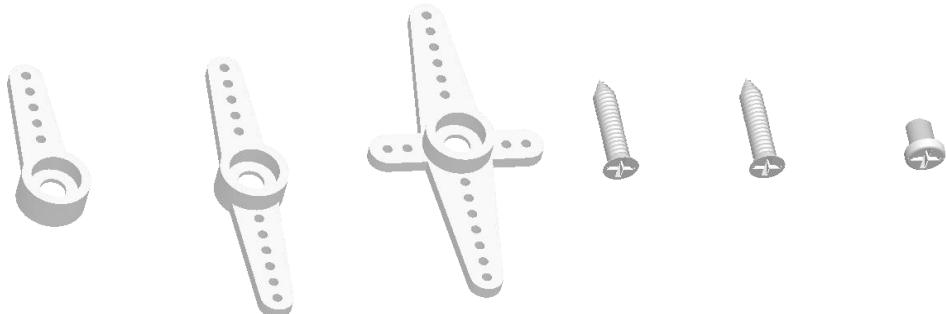


Effect diagram after assembling



### 3. Calibrate the servos.

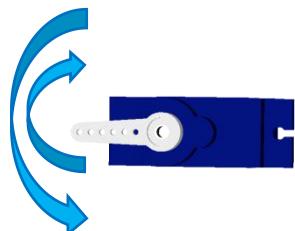
First, learn the structure. The servo can connect the rocker arm and spin to drive components bound with the arm. There are 3 types of rocker arms and 3 screws in the package. The smallest one is to fix the arm onto the servo.



Mount and remove the rocker arm.



Rotate the rocker arm between 0 and 180 degrees.

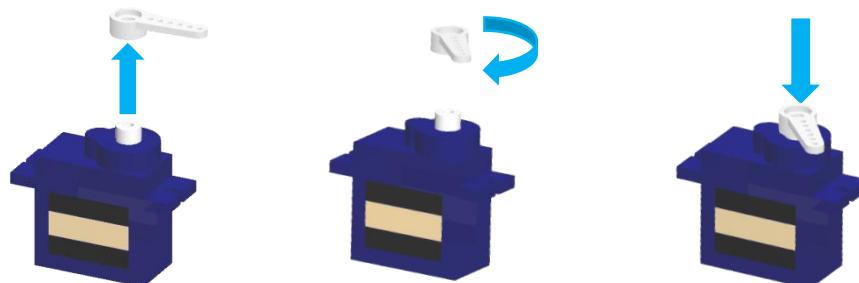


Now adjust the servo. This step is to make the servo shaft in the middle, so the component connected to the servo can be driven to move in a certain scope as needed.

Adjust the rocker arm to make it rotate to an almost equal angle towards left and right.



If the angle is not nearly the same, please remove the arm and install it again. Repeat the step until nearly the SAME degree.



After the adjusting, the rocker arm should be in the middle axis. Remove the arm.



Make sure all servos have been adjusted and DO NOT spin the servo shaft before the whole assembly is done for the car. If you move it accidentally, readjust before the assembly.

4. Take two servos and fix them with A09 respectively.

Assemble the following components

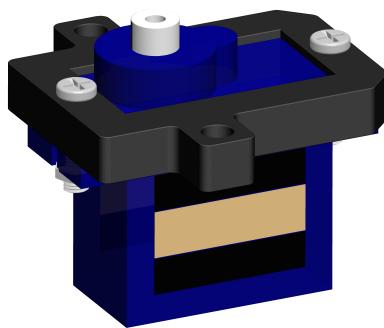
M2\*10  
Screw X2

A09 X1

Servo X1

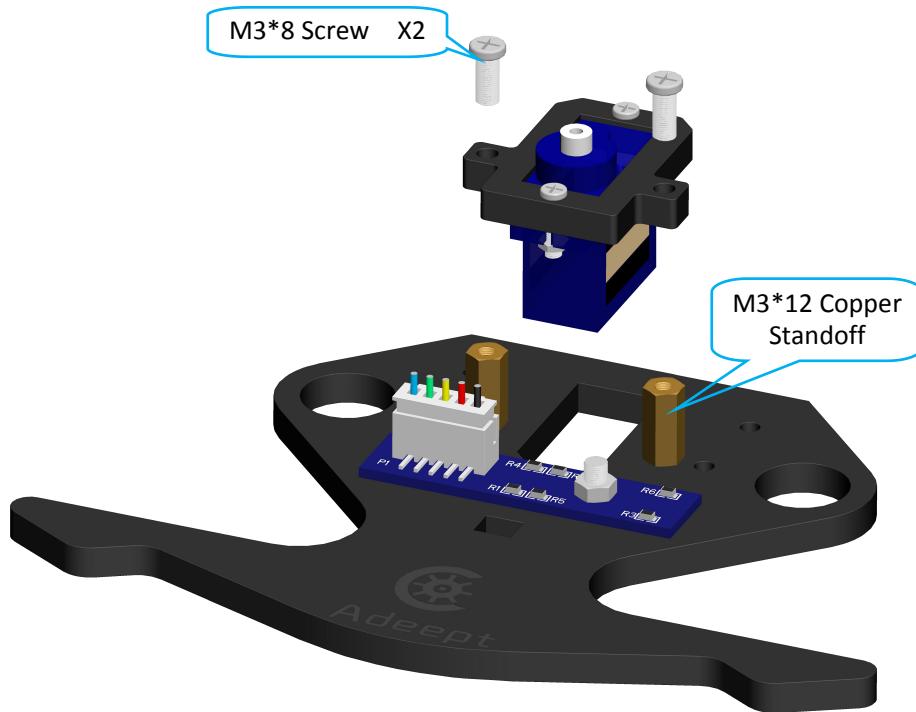
M2 Nut X2

Effect diagram after assembling

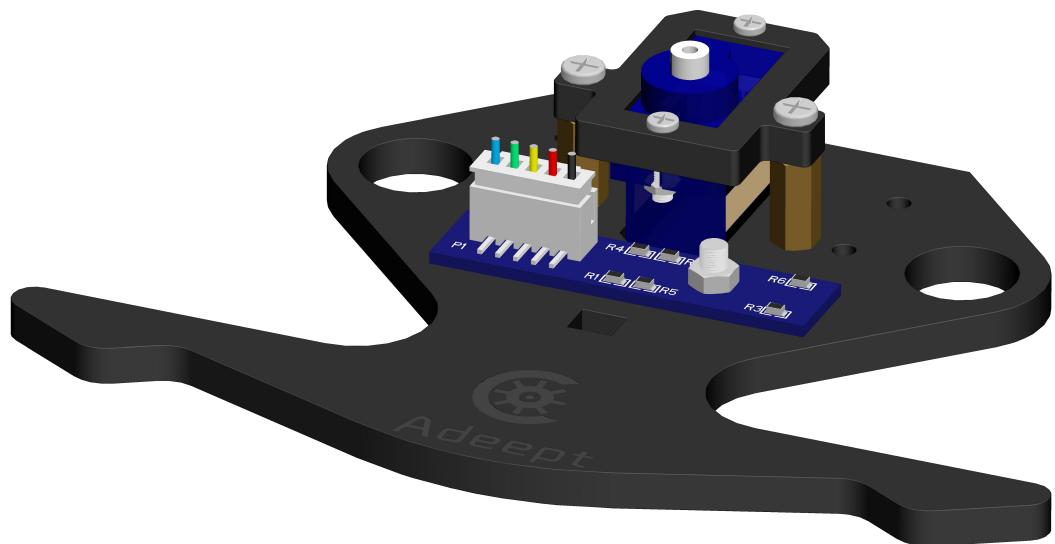


5. Fix the assembled servo to the M3\*12 Copper Standoff on the A03.

Assemble the following components

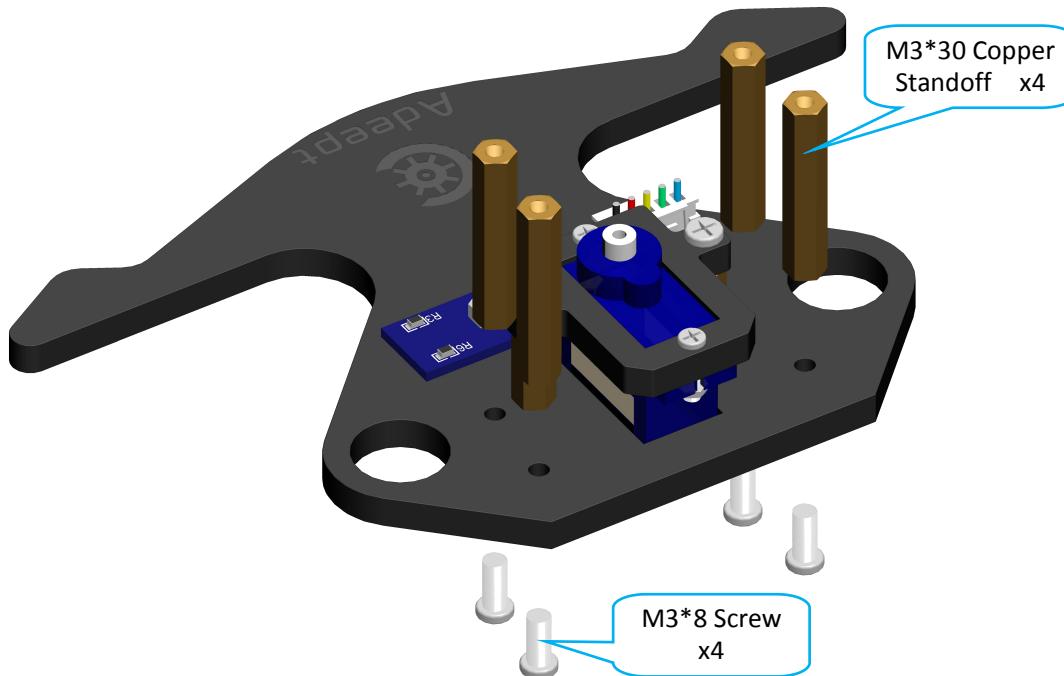


Effect diagram after assembling



6. Fix four M3\*30 Copper Standoff on A03.

Assemble the following components

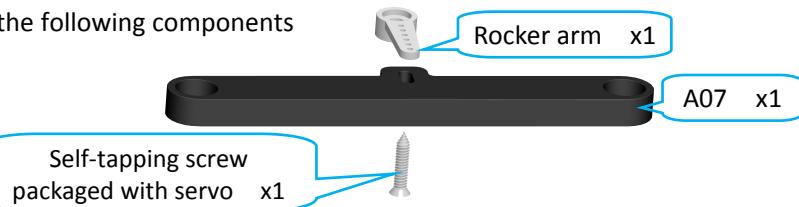


Effect diagram after assembling



7. Connect the rocker arm of a servo with A07.

Assemble the following components



Effect diagram after assembling

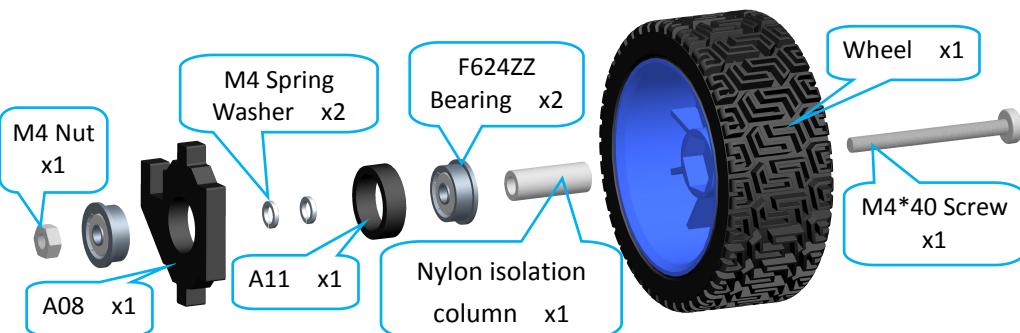
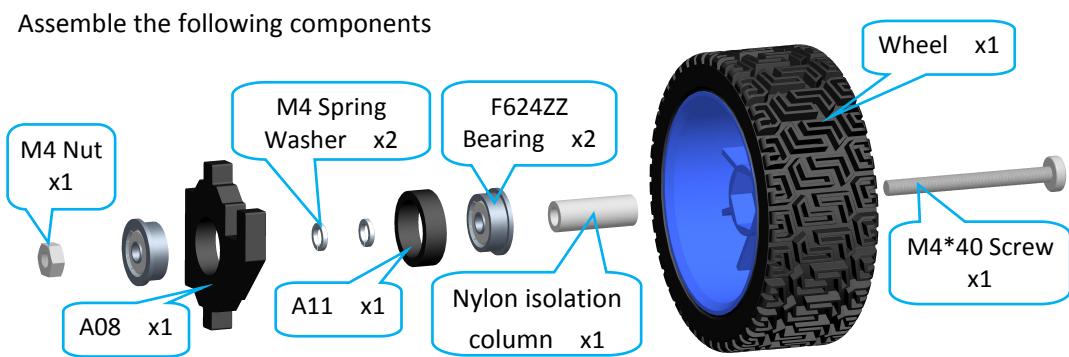


Pay attention not to tighten the screw, otherwise the A07 can't move freely.

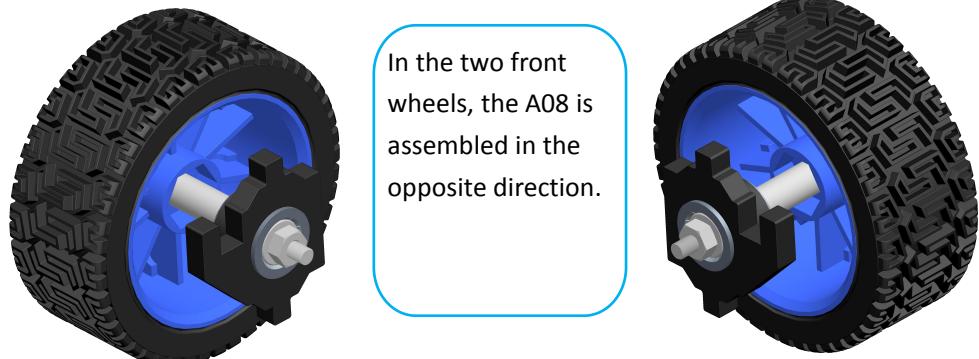
Keep a gap between the Screw and A07

8. Two front wheels assembly.

Assemble the following components

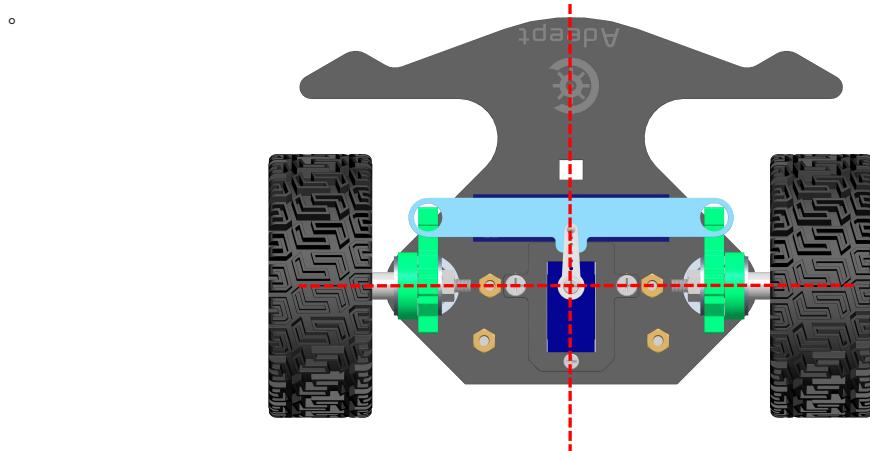


Effect diagram after assembling

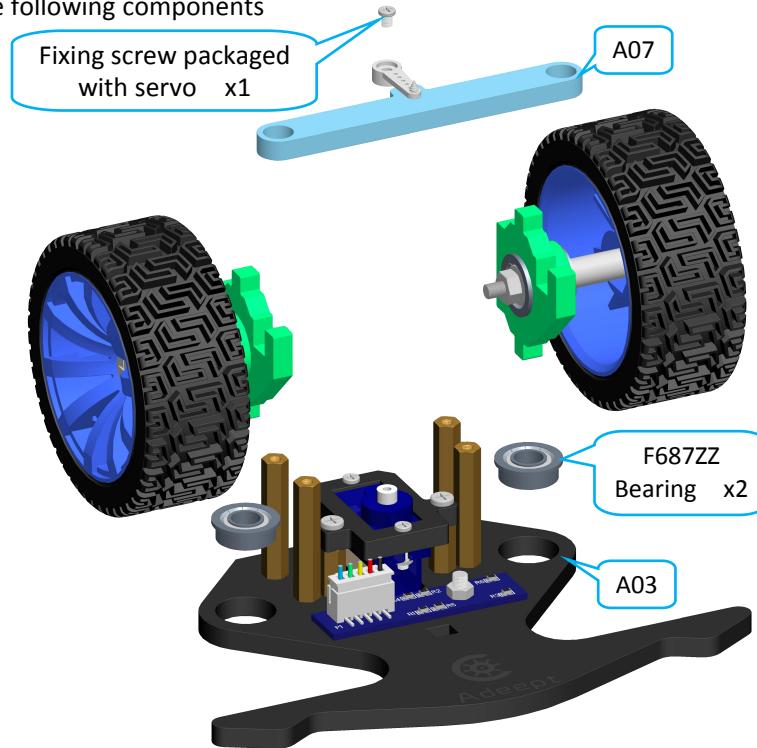


9. Fix the rocker arm on the A07 to the servo on the A03.

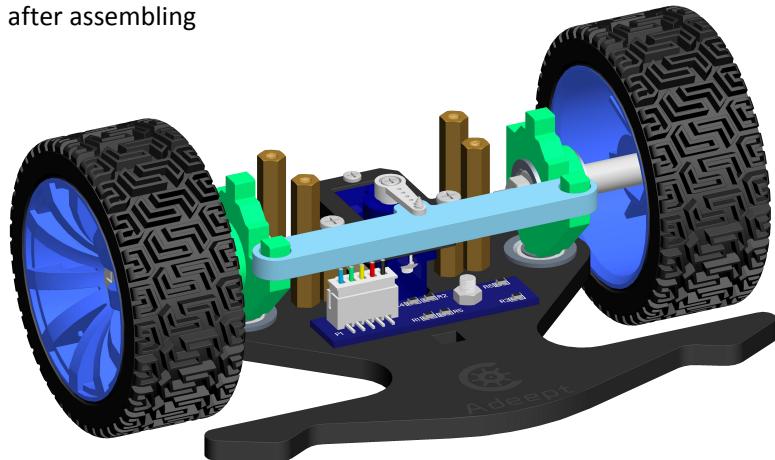
Insert the rocker arm into the servo in the direction as the following picture and then fix it.



Assemble the following components

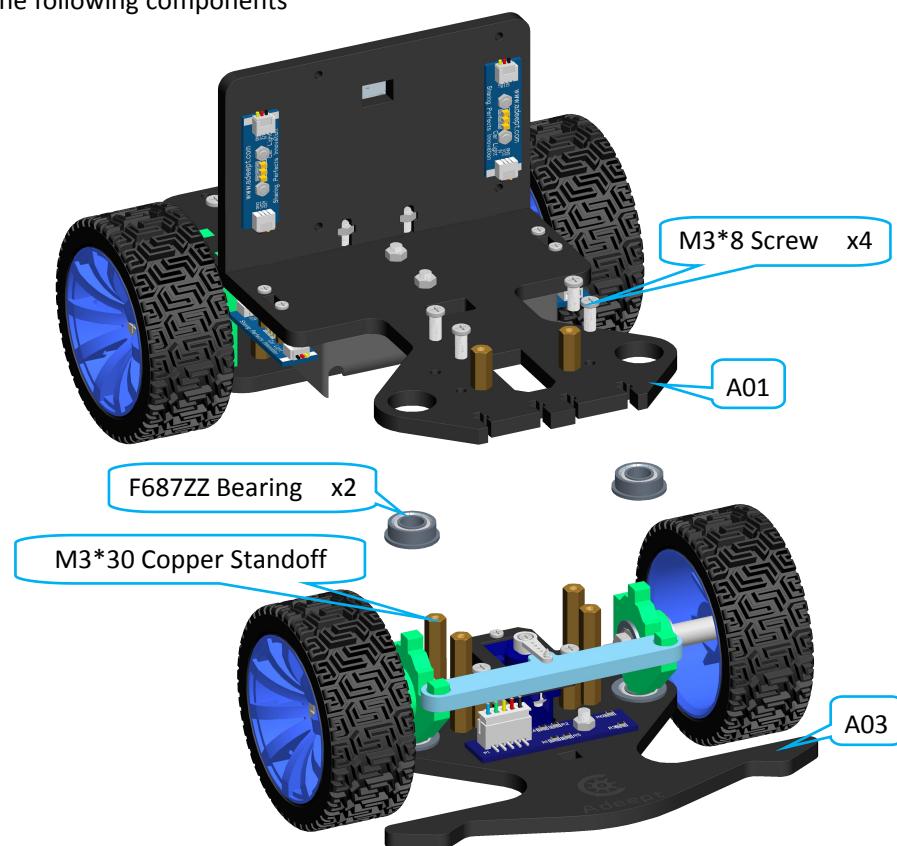


Effect diagram after assembling

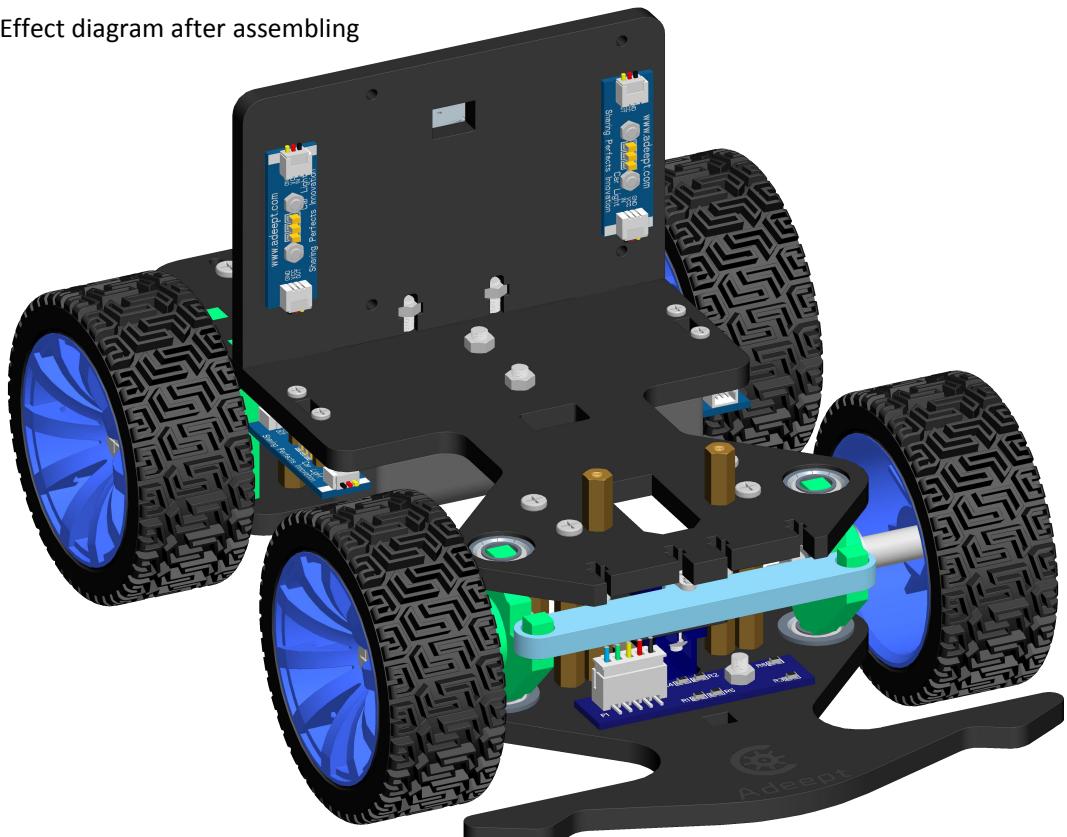


10. Fix the M3\*30 Copper Standoff on A01 and A03.

Assemble the following components

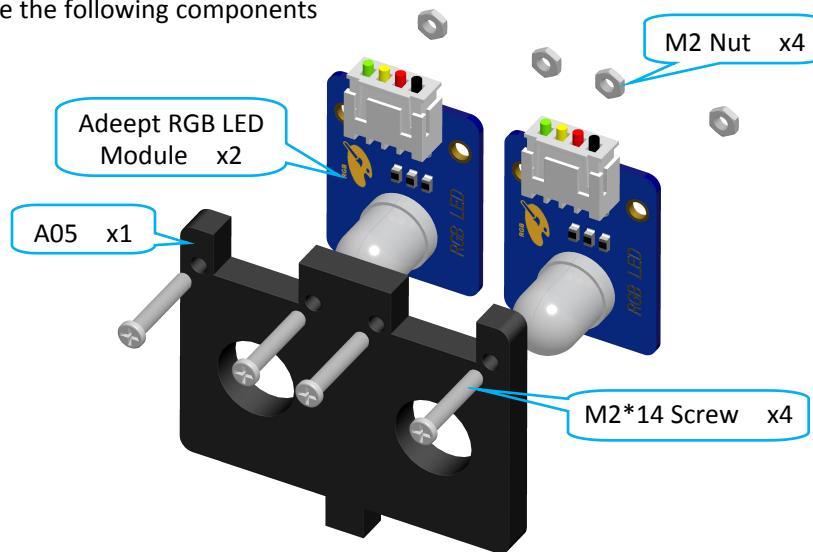


Effect diagram after assembling

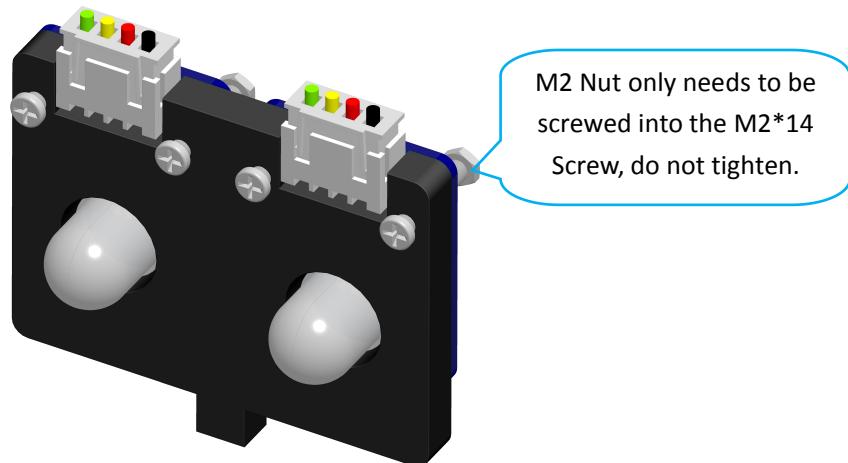


11. Fix Adeept RGB LED Module on A05.

Assemble the following components

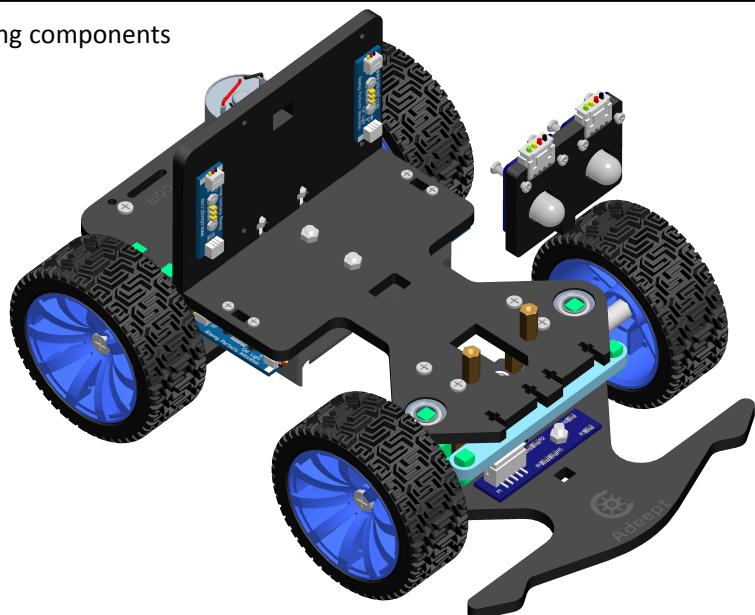


Effect diagram after assembling

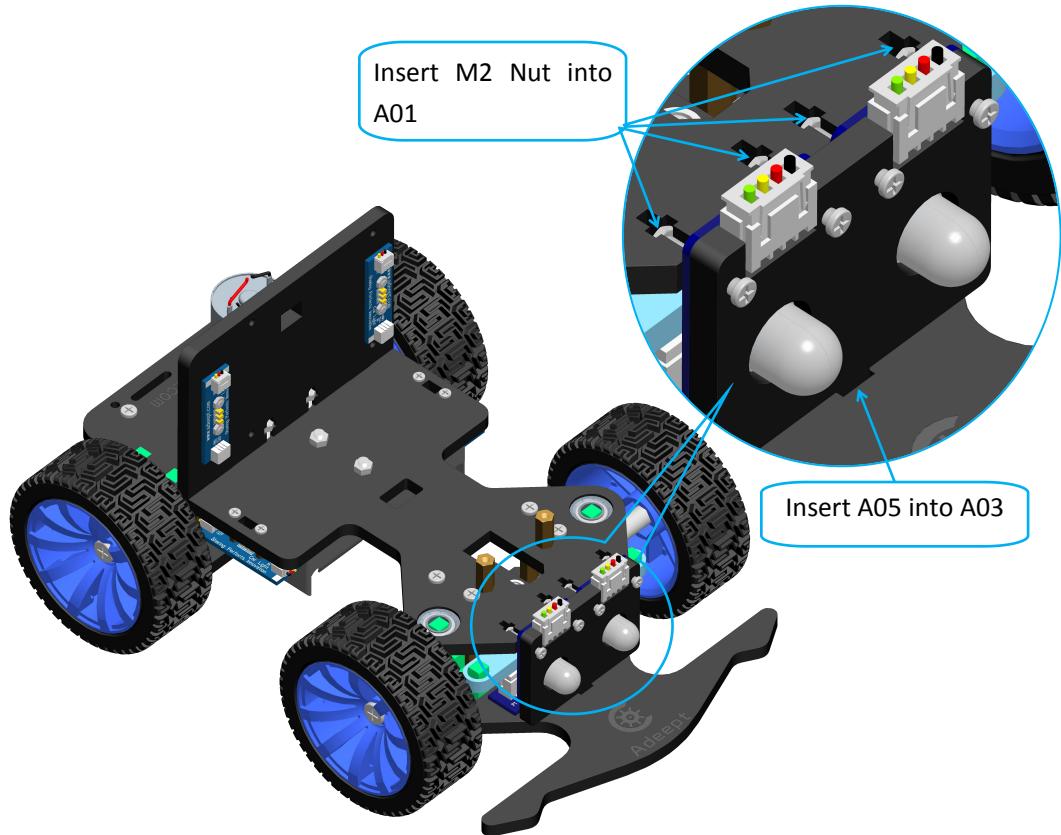


12. Fix Adeept RGB LED Module and A05 on A01.

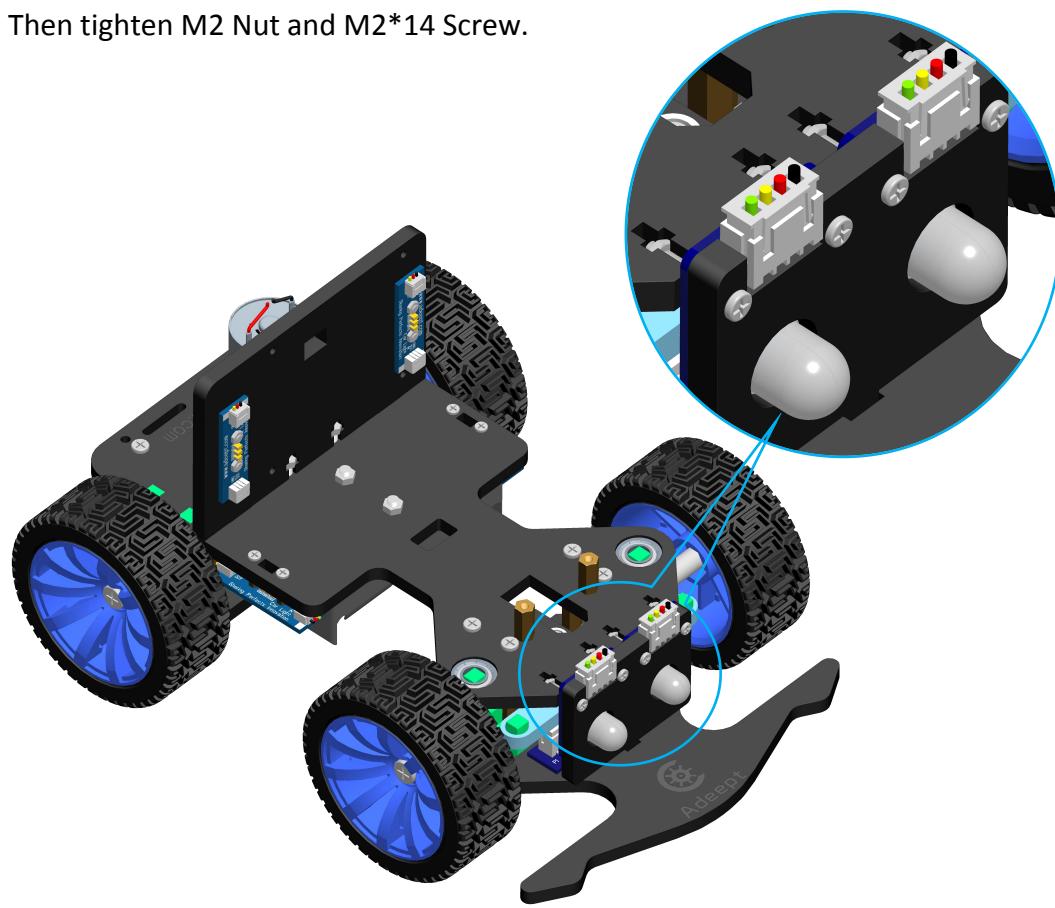
Assemble the following components



First insert M2 Nut into A01, A05 into A03. Then tighten M2 Nut and M2\*14 Screw.



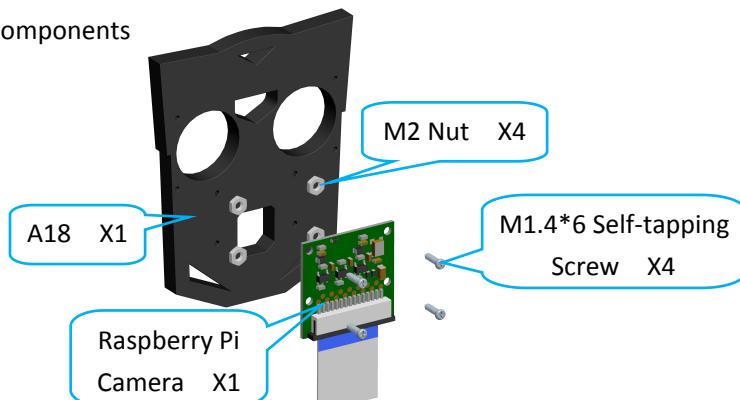
Then tighten M2 Nut and M2\*14 Screw.



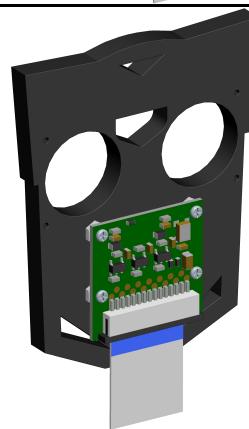
## 2.5. Front part assembly

1. Fix Raspberry Pi Camera to A18.

Assemble the following components

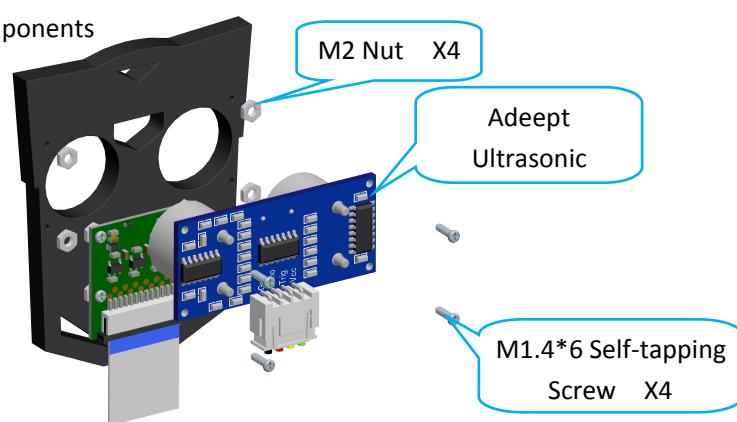


Effect diagram after assembling

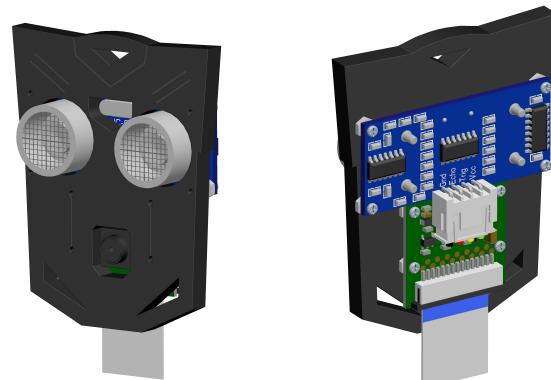


2. Fix the Adeept Ultrasonic Module to the A18.

Assemble the following components

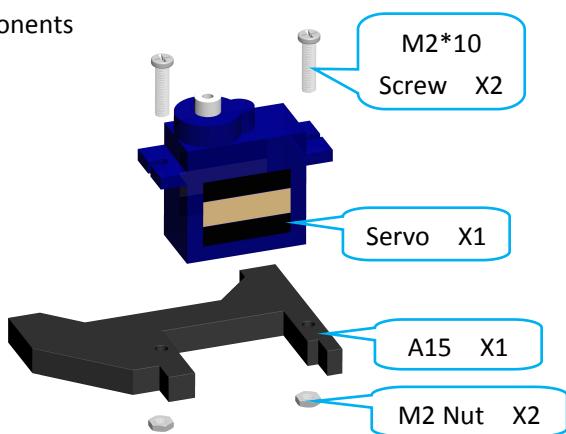


Effect diagram after assembling



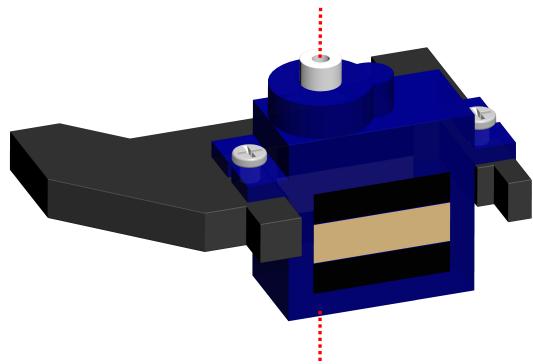
3. Take a servo and fix it with A15.

Assemble the following components



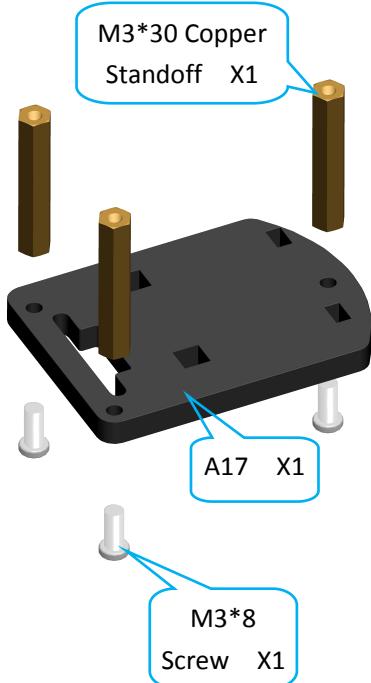
Effect diagram after assembling

Install strictly according to the position in the picture, A15 is below the Servo, output shaft of the Servo is to the left

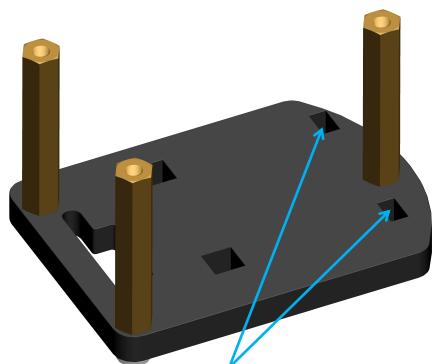


4. Fix three M3\*30 Copper Standoff to A17.

Assemble the following components



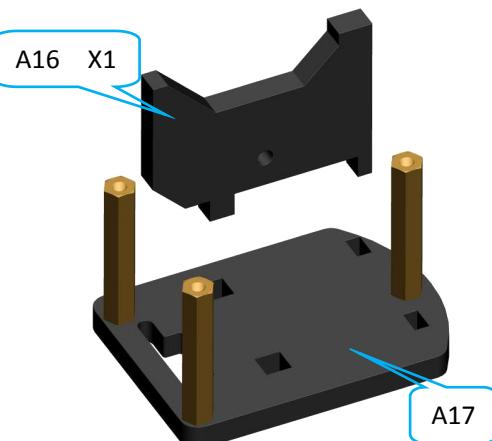
Effect diagram after assembling



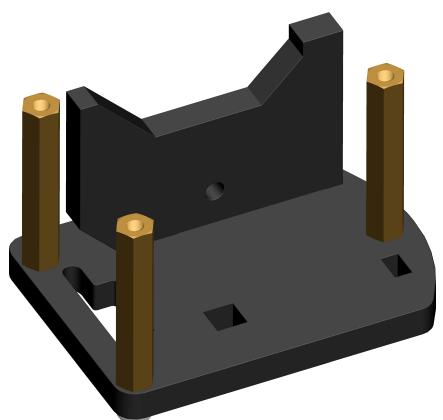
Install strictly according to the position of this angle of view. Do not reverse the A17. It can be judged according to the position of the two square holes indicated by the arrows.

5. Insert A16 into A17.

Assemble the following components

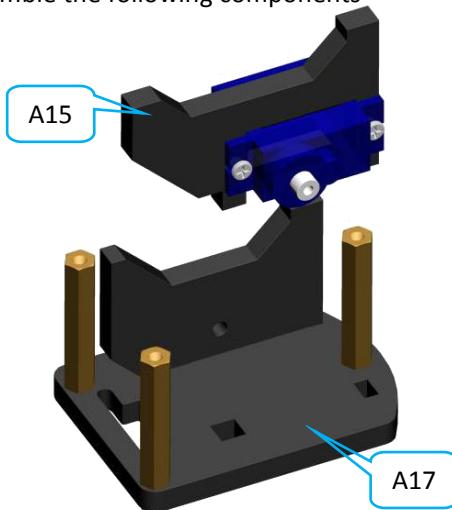


Effect diagram after assembling

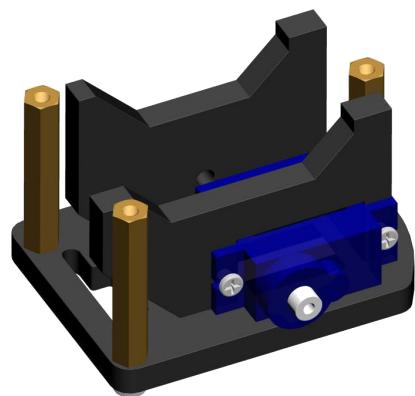


6. Insert A15 into A17.

Assemble the following components

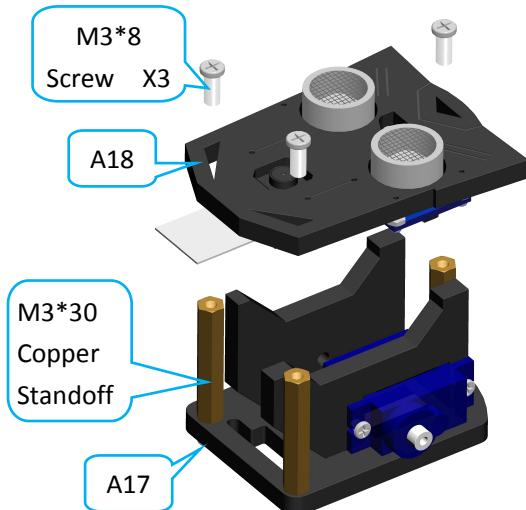


Effect diagram after assembling

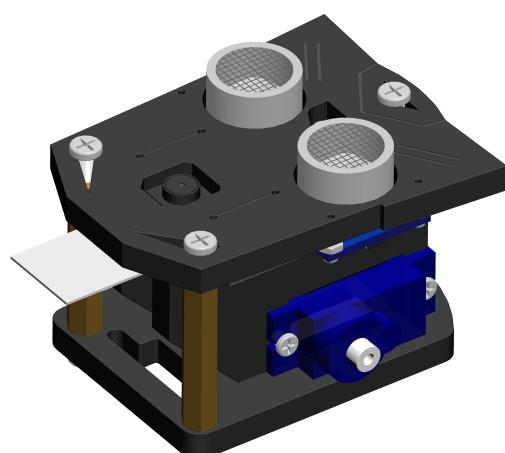


7. Fix the A18 to the M3\*30 Copper Standoff on the A17.

Assemble the following components

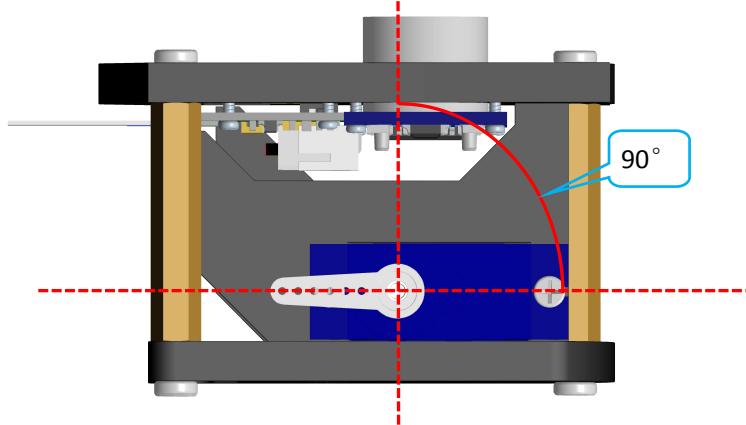


Effect diagram after assembling

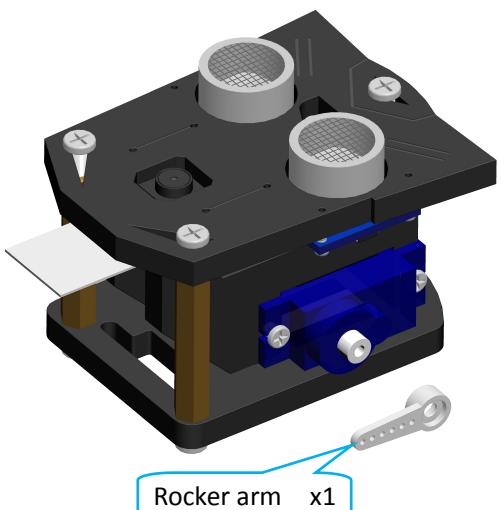


8. Install a rocker arm on the servo.

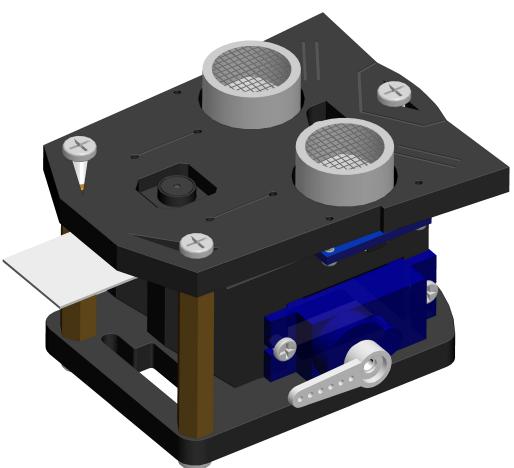
Install the rocker arm on the servo at the angle as shown in the picture.



Assemble the following components

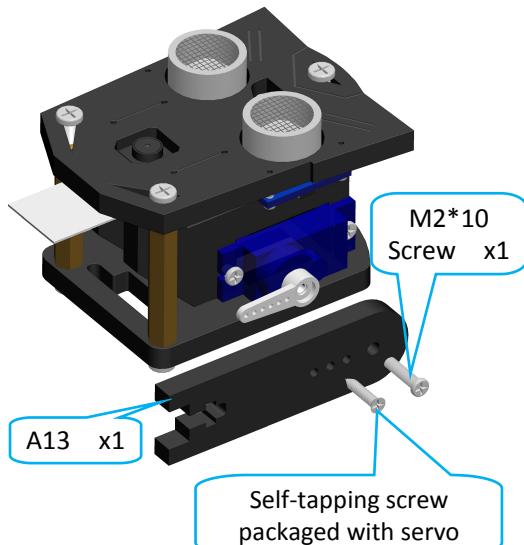


Effect diagram after assembling

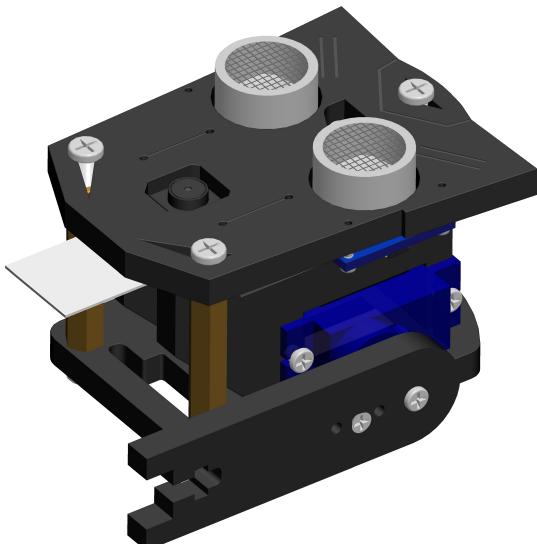


9. Fix A13 with servo.

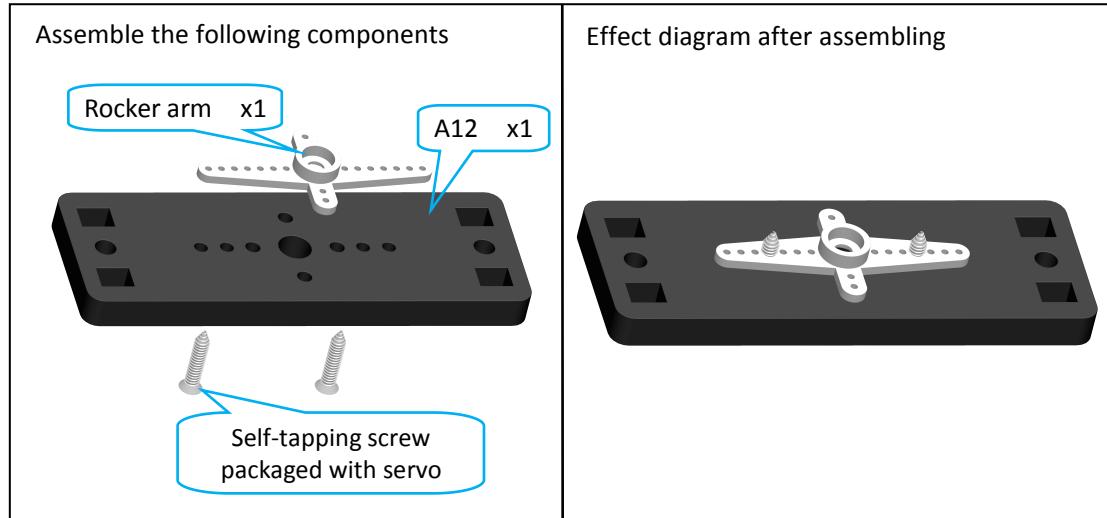
Assemble the following components



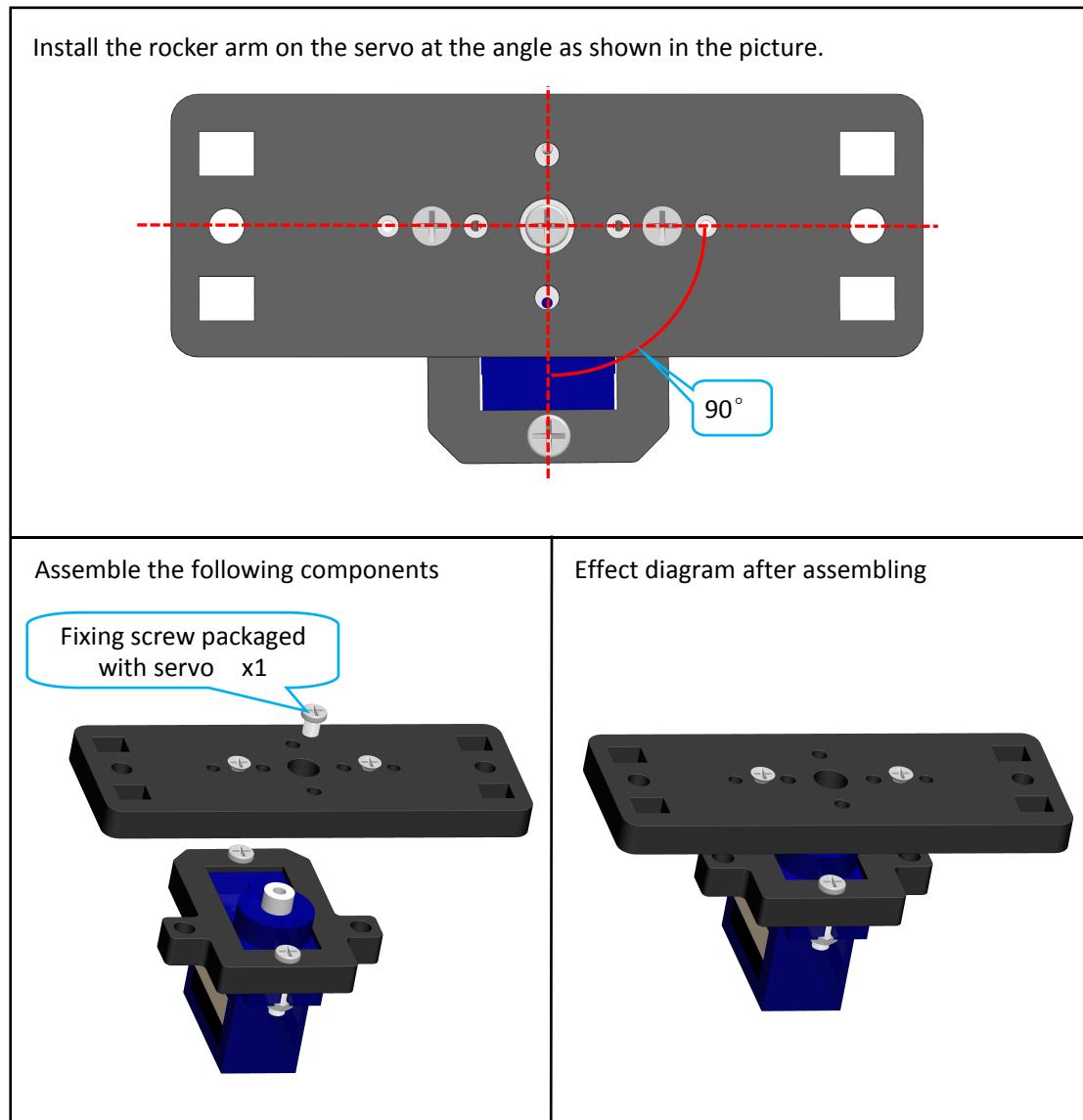
Effect diagram after assembling



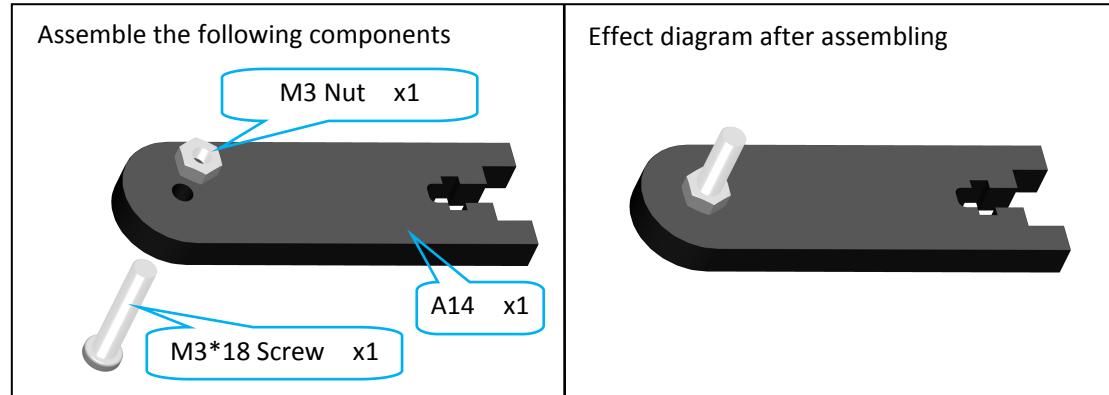
10. Select a 'cross' rocker arm and fix to the A12.



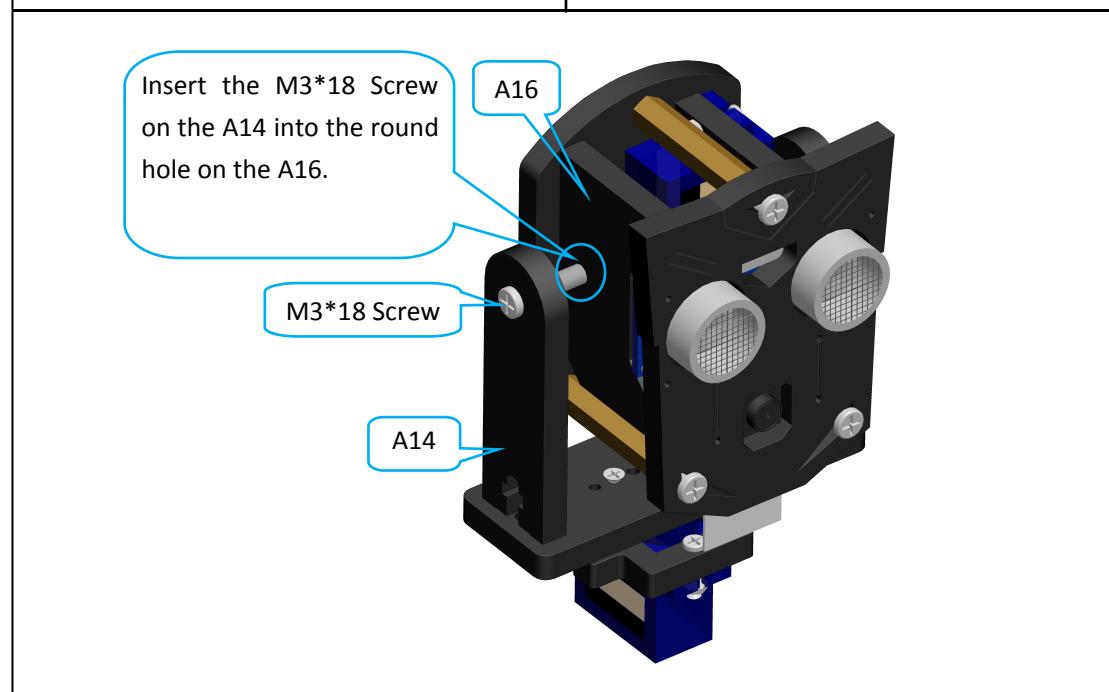
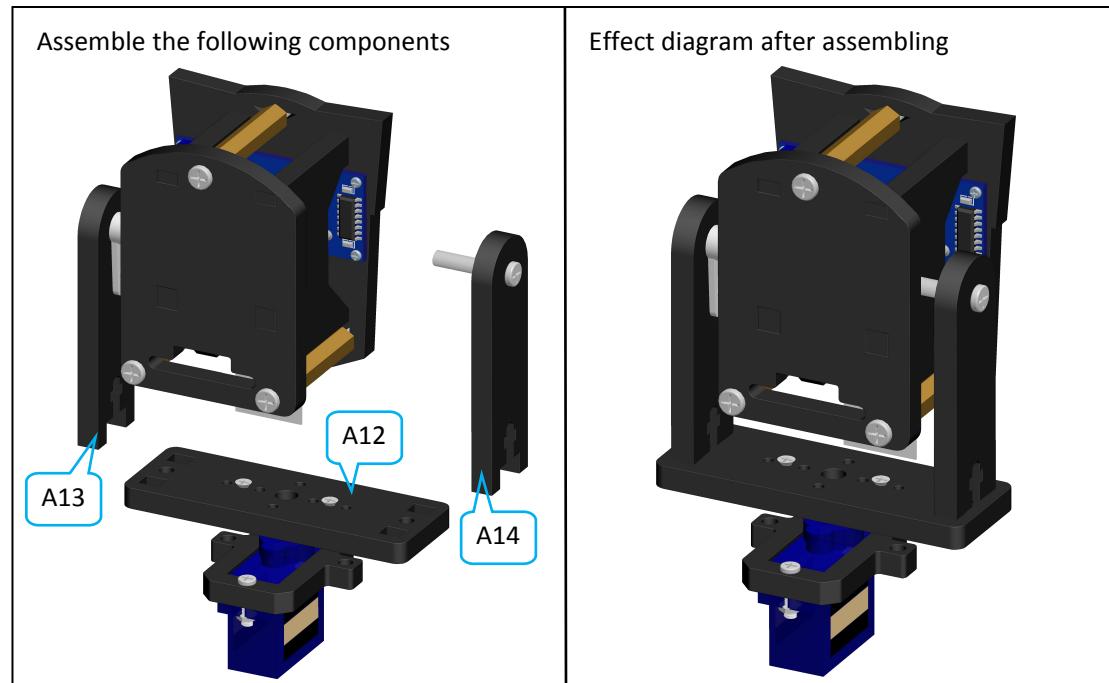
11. Then fix the rocker arm to the servo assembled with A09.



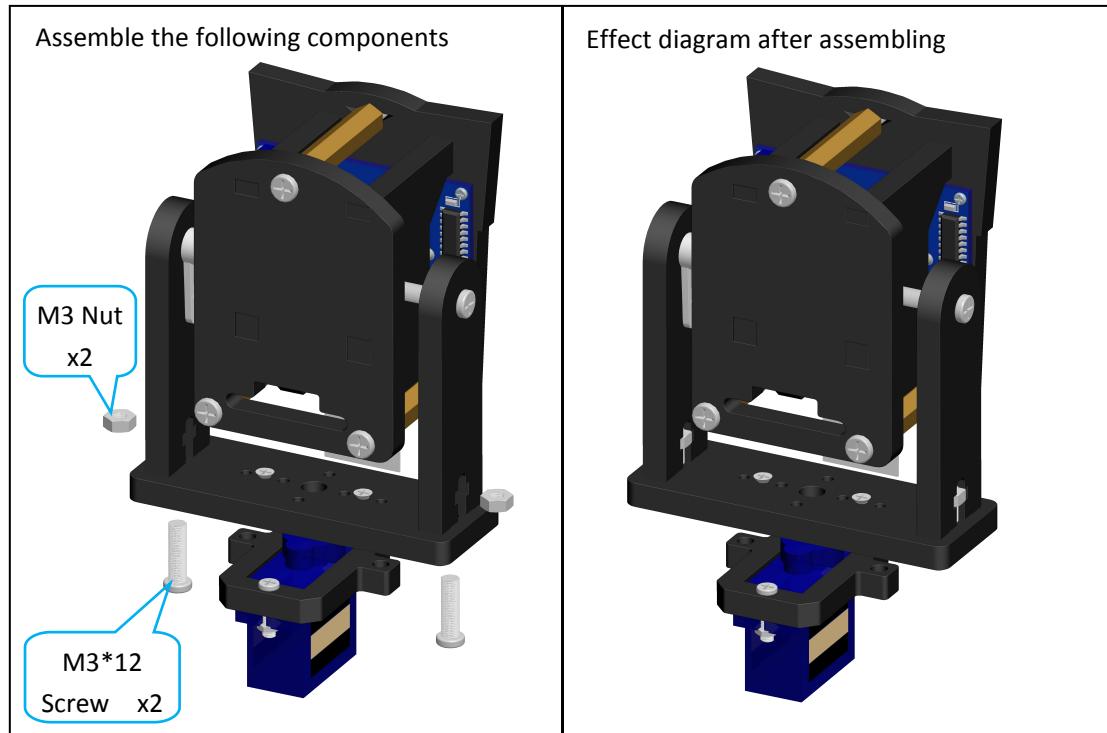
12. Fix M3\*18 Screw on A14.



13. Insert A13, A14 into A12.

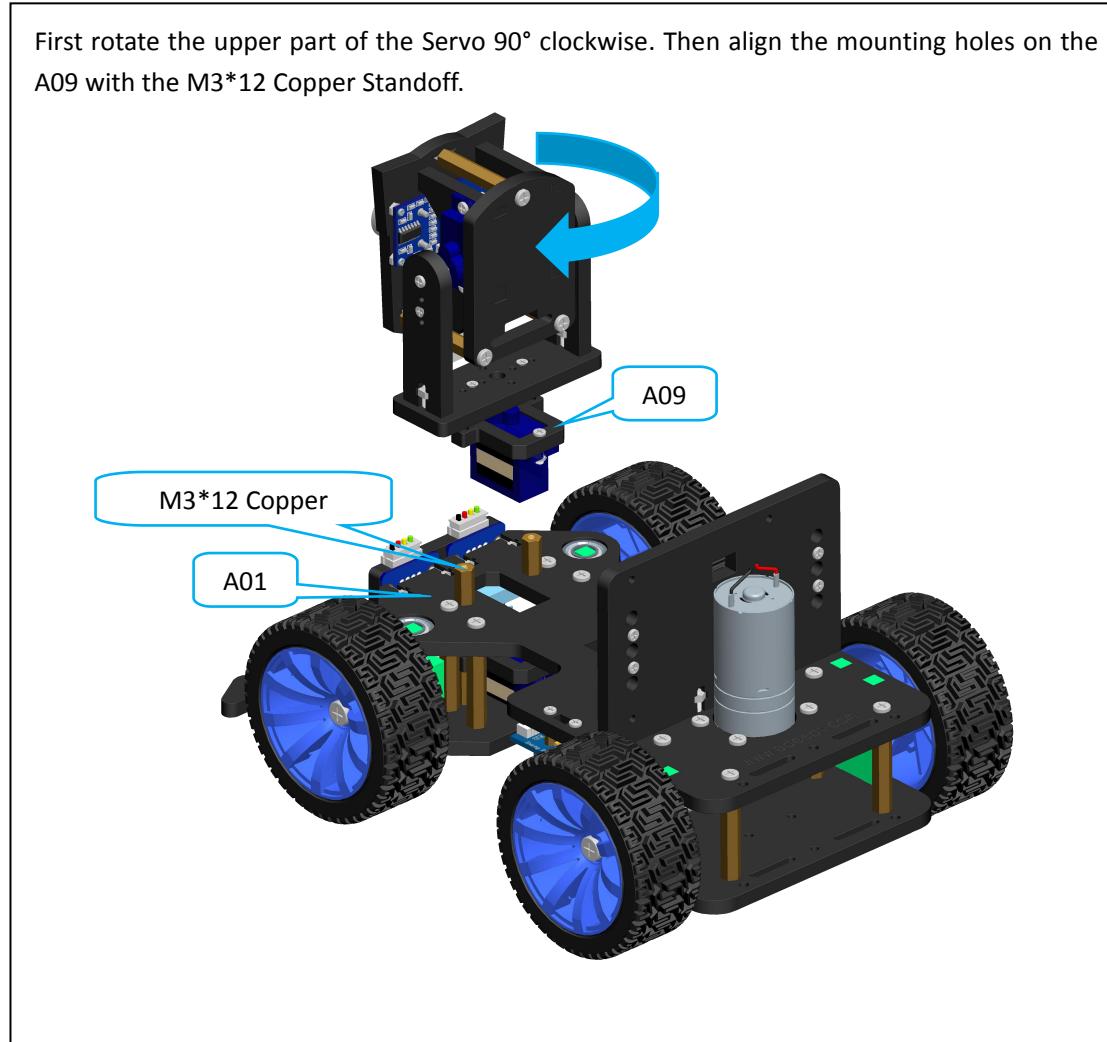


14. Then use A3\*12 Screw to fix A13 and A14 to A12.

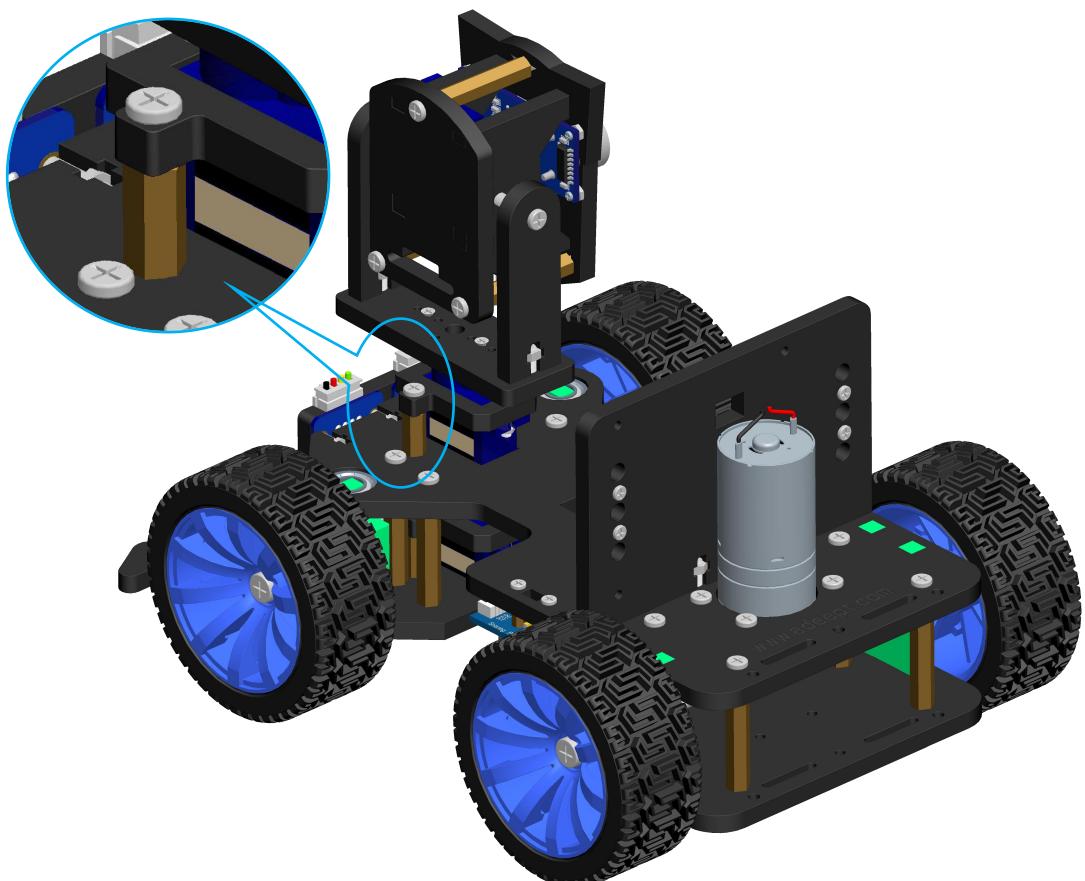
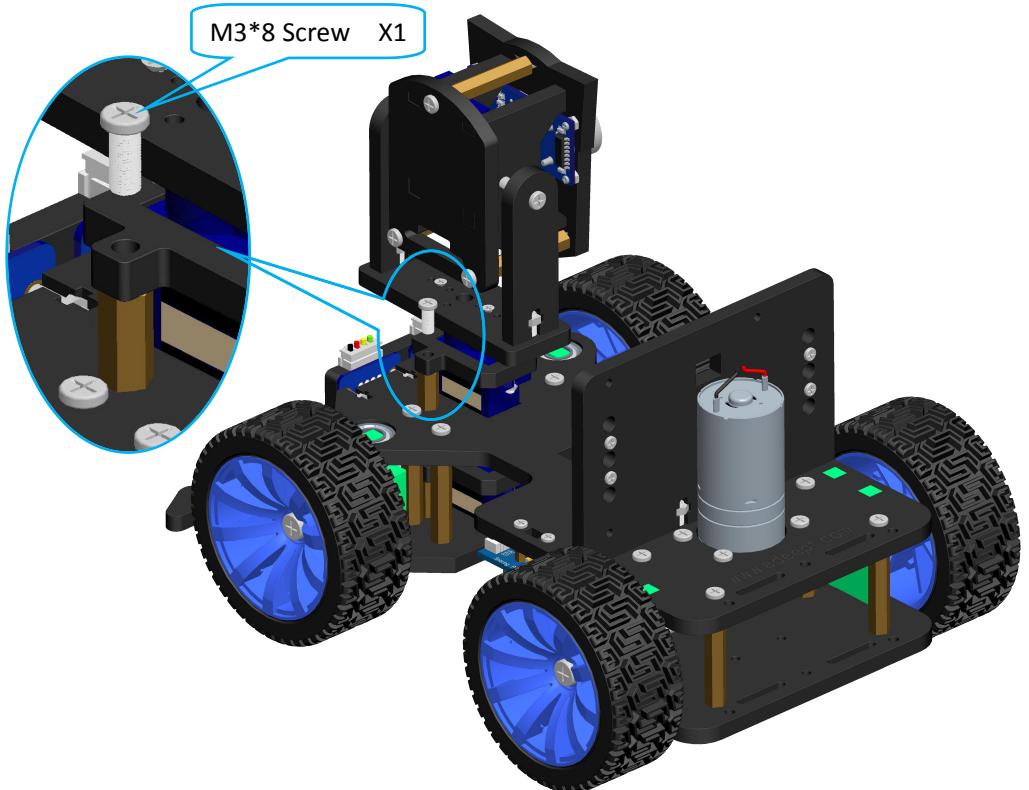


15. Fix the assembly completed in the previous step on the M3\*12 Copper Standoff on the A1.

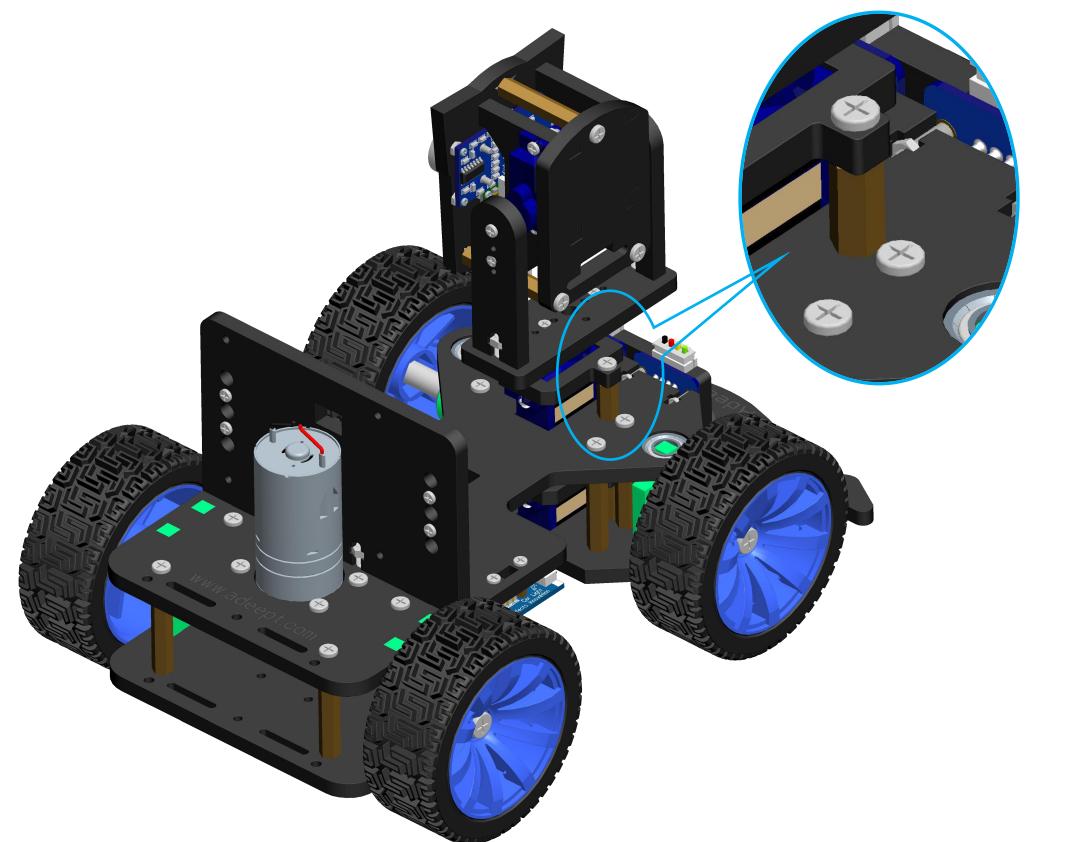
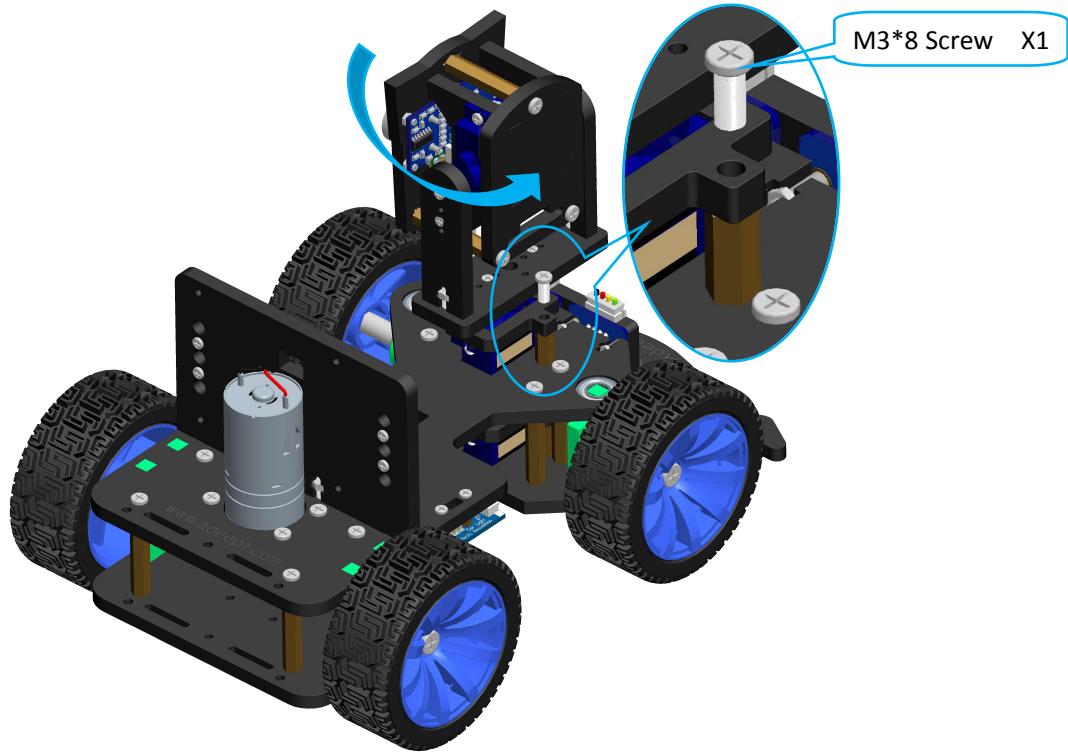
First rotate the upper part of the Servo 90° clockwise. Then align the mounting holes on the A09 with the M3\*12 Copper Standoff.



Then fix A09 to M3\*12 Copper Standoff with M3\*8 Screw.

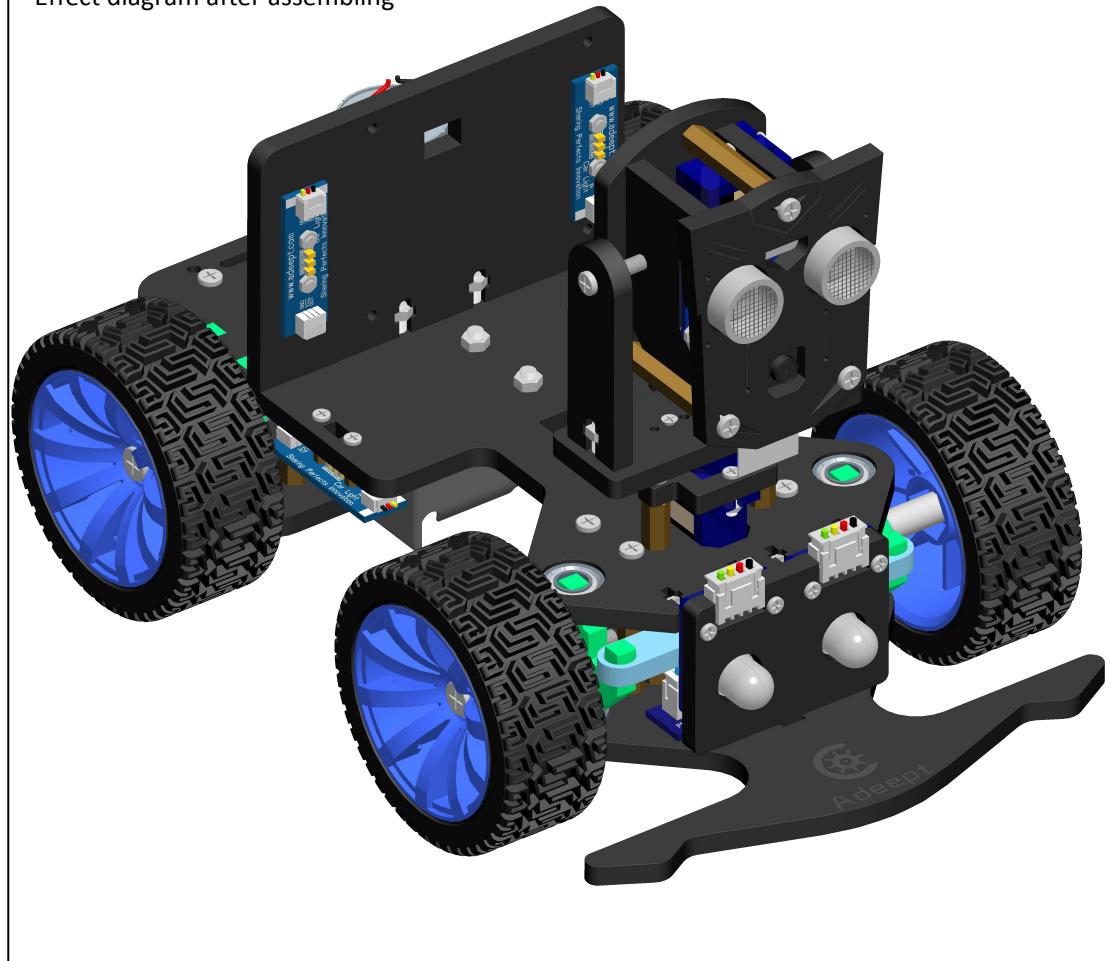


Rotate the upper part of the Servo 180° counterclockwise, then fix the A09 and M3\*12 Copper Standoff with the M3\*8 Screw.



Finally turn the upper part of the servo back to original position.

Effect diagram after assembling

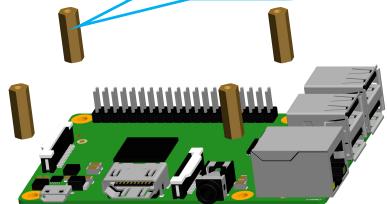


## 2.6. Raspberry Pi assembly

1. Fix M2.5\*10+6 Copper Standoff and M2.5\*14 Copper Standoff to Raspberry Pi.

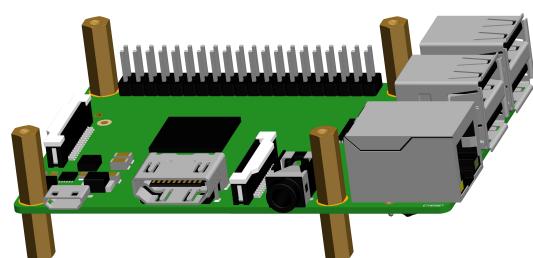
Assemble the following components

M2.5\*14 Copper  
Standoff X1

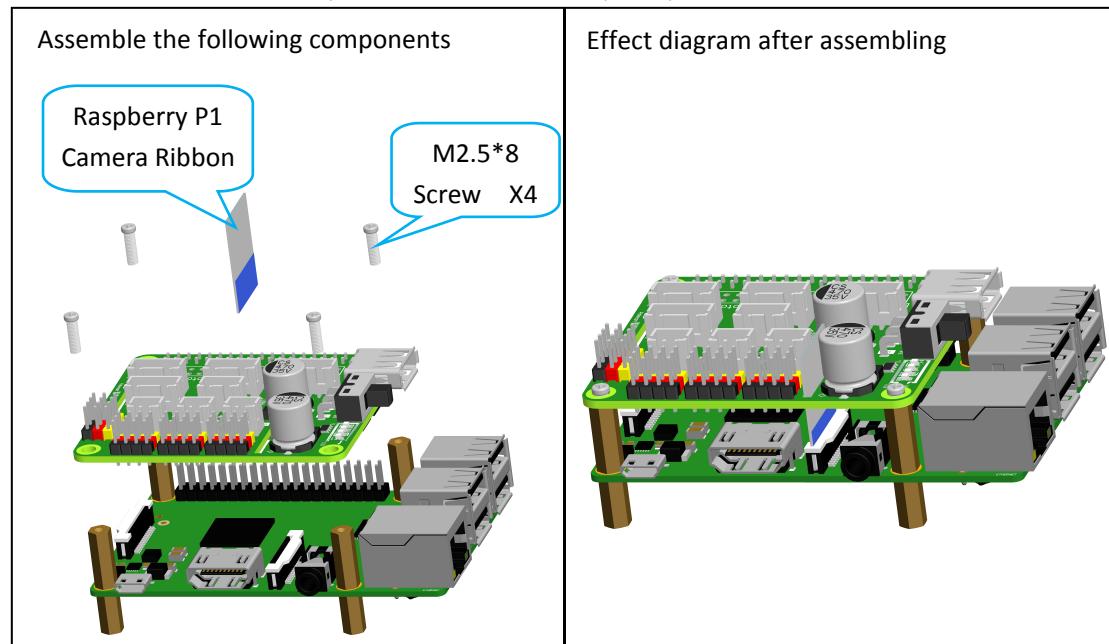


M2.5\*10+6 Copper  
Standoff X1

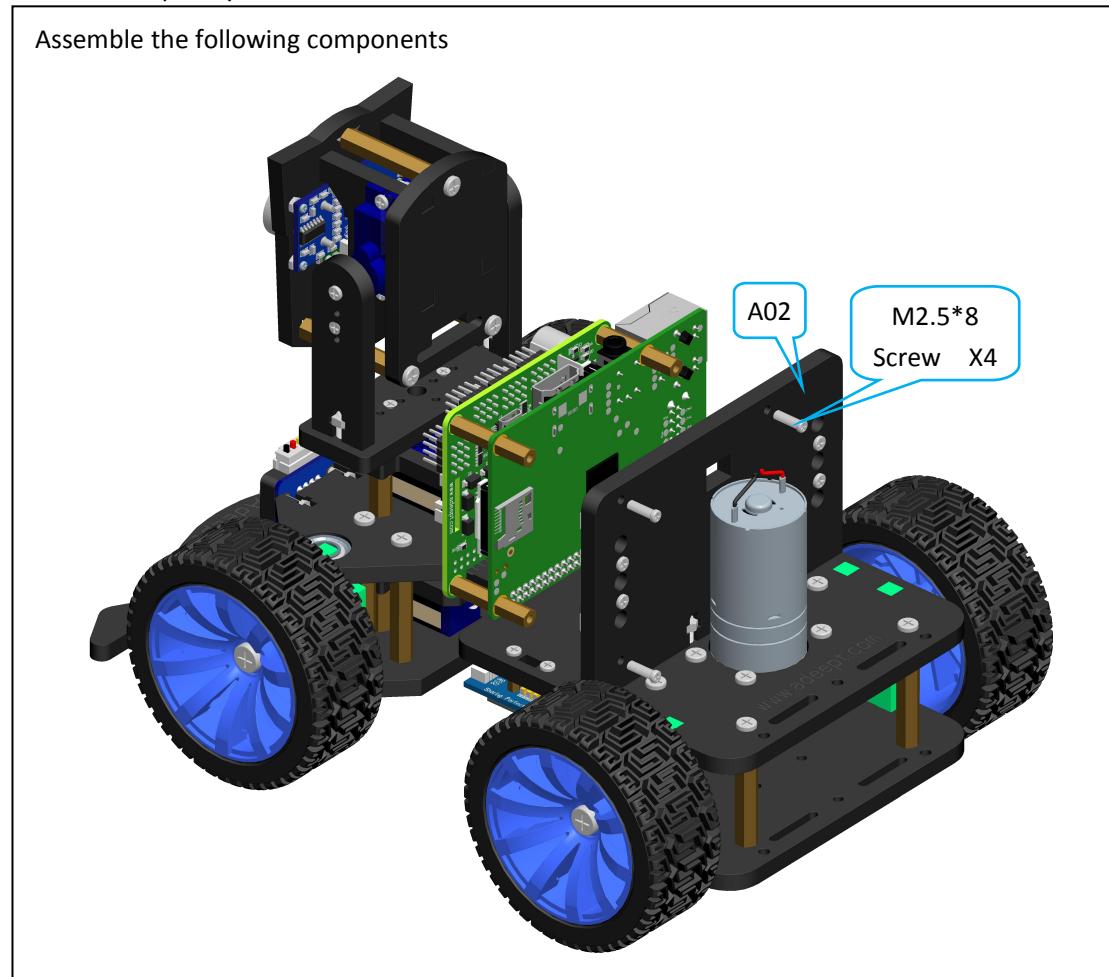
Effect diagram after assembling



2. Connect the other end of the Raspberry P1 Camera Ribbon to the Raspberry Pi via Adeept Motor HAT and fix the Adeept Motor HAT to the Raspberry Pi.

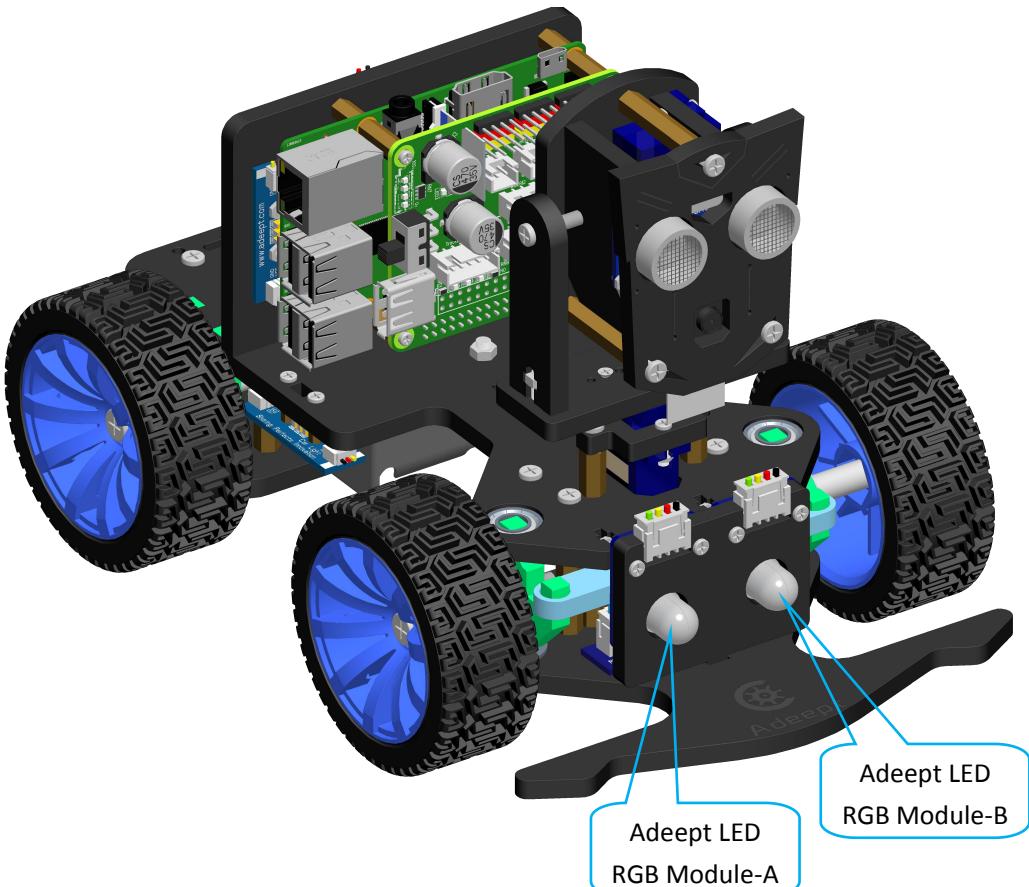
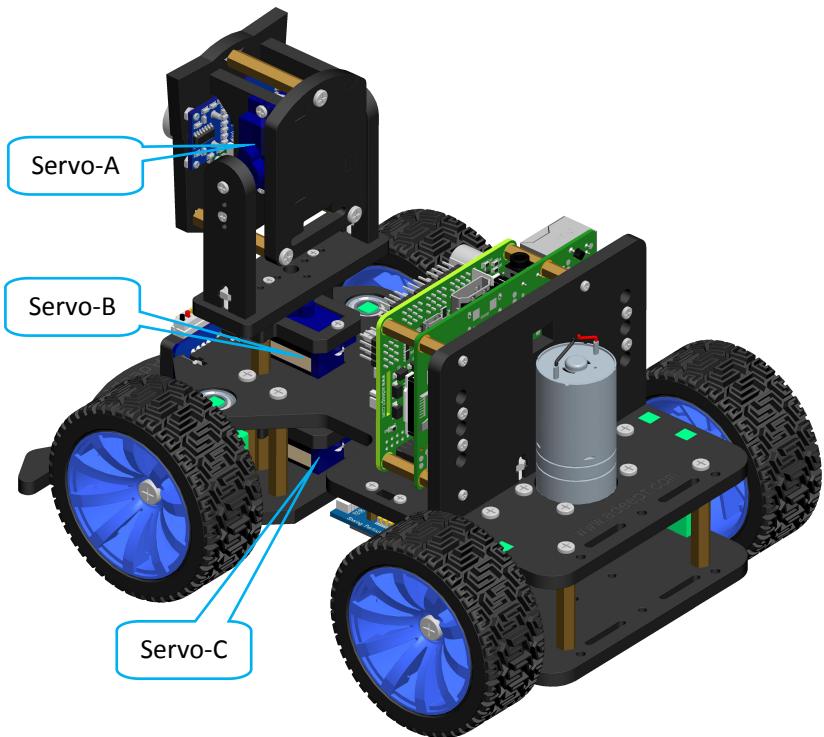


3. Fix the Raspberry Pi on the A02.



Effect diagram after assembling

Number the three servos and two Adeept RGB LED Modules to facilitate the next circuit connection.

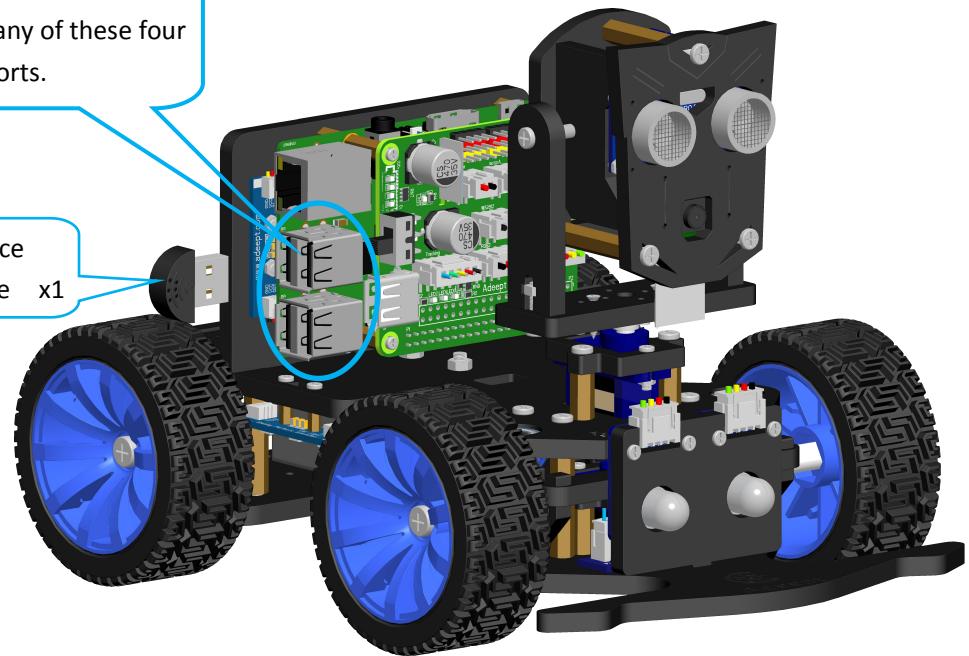


4. If you have to use the Voice Module, you need to insert it into the Raspberry Pi.

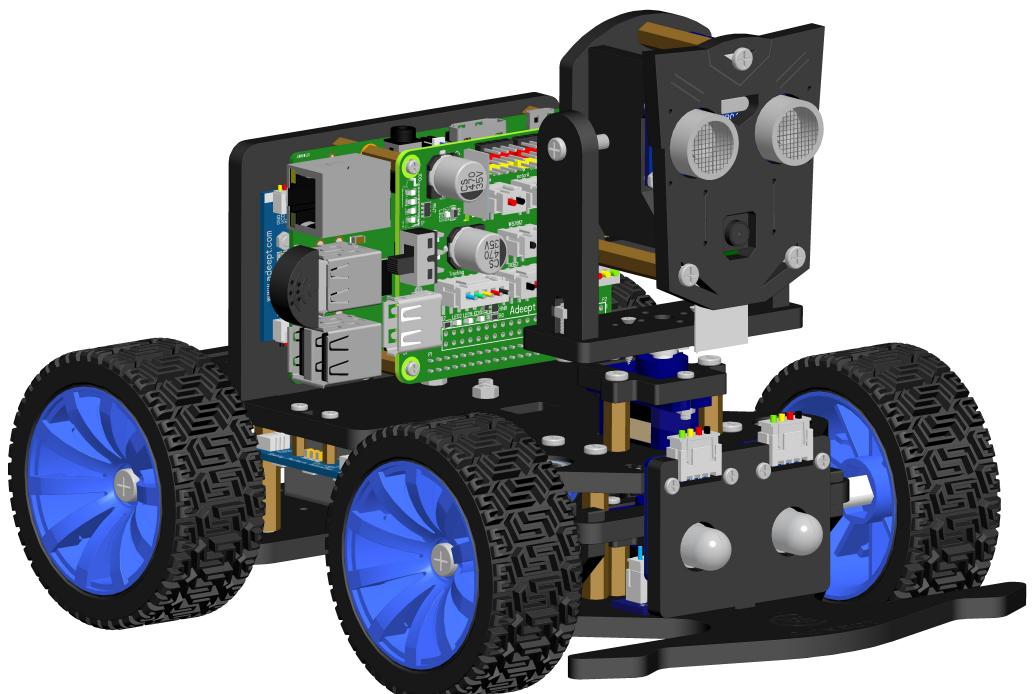
Assemble the following components

Insert the Voice Module into any of these four USB ports.

Voice Module x1

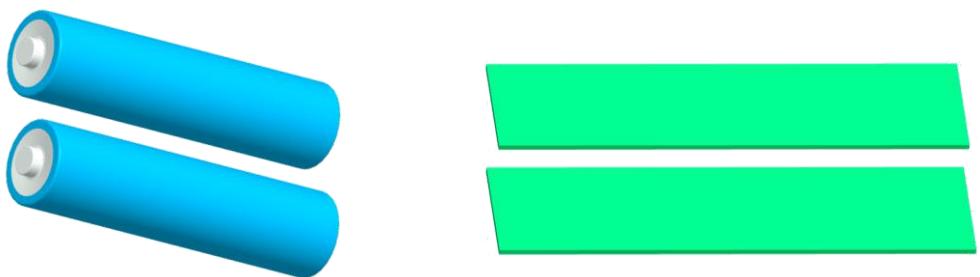


Effect diagram after assembling



## 2.7. Install and Remove Batteries

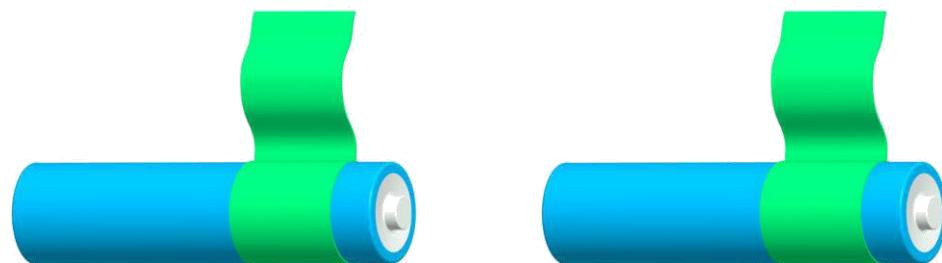
Take out 2 ribbons and 2 batteries.



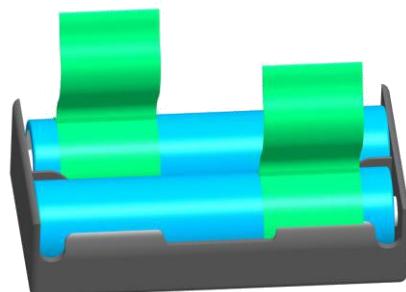
Roll one end of the ribbon to let through a battery and fix.



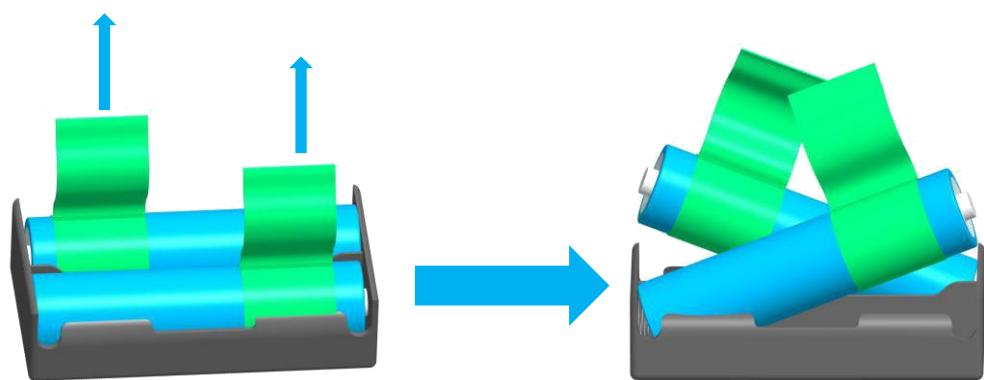
Insert the batteries into the rings - ribbon closer to the anode.



Install the batteries into the holder based on the pole.



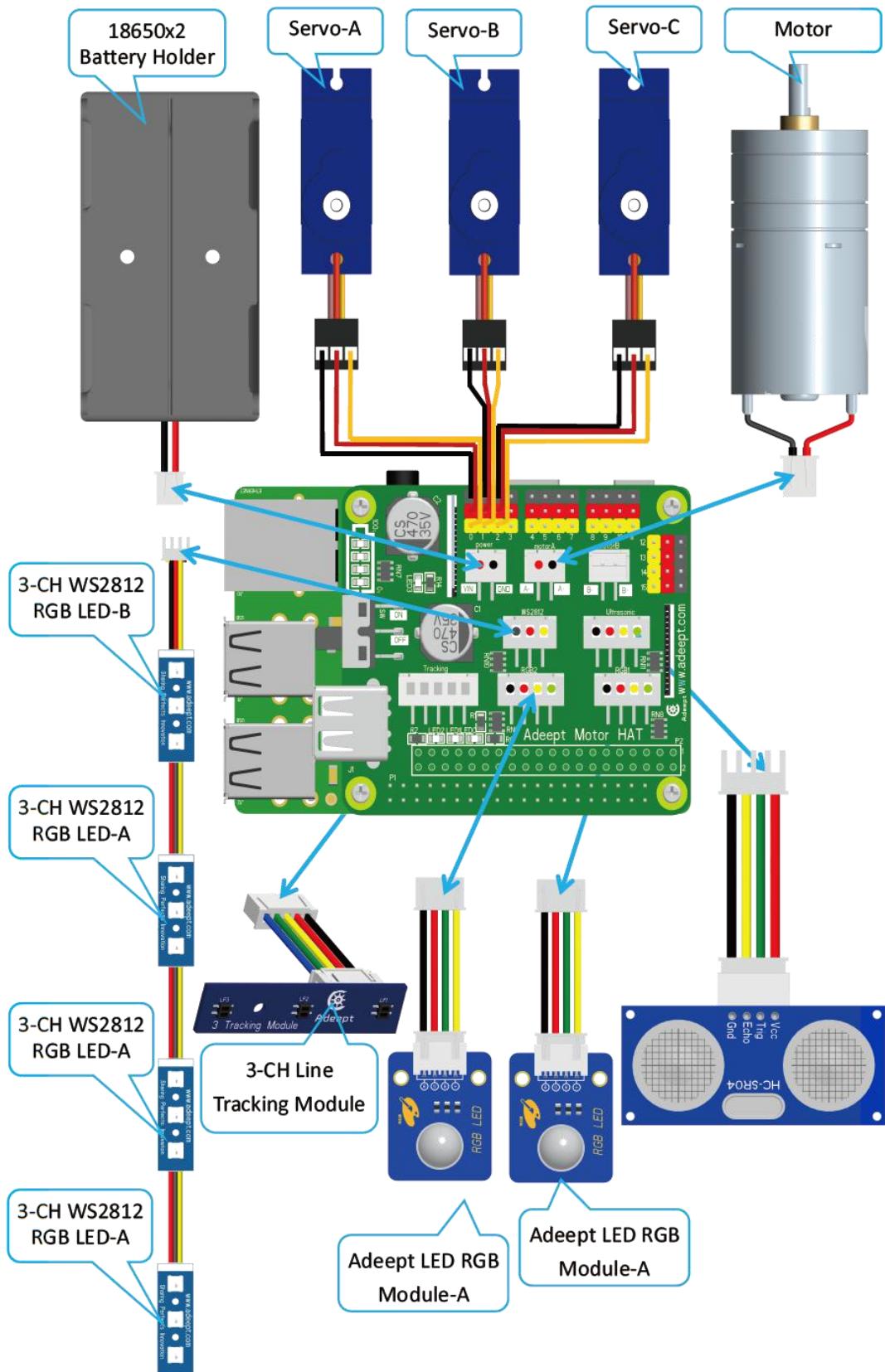
To remove the batteries, just pull the ribbon and take them out.



## 2.8. Circuit Connection

Connect components based on the figure.

Pay attention to match the wire and port and not connect inversely.



## 3. Software & Hardware

### 3.1. Software Installation

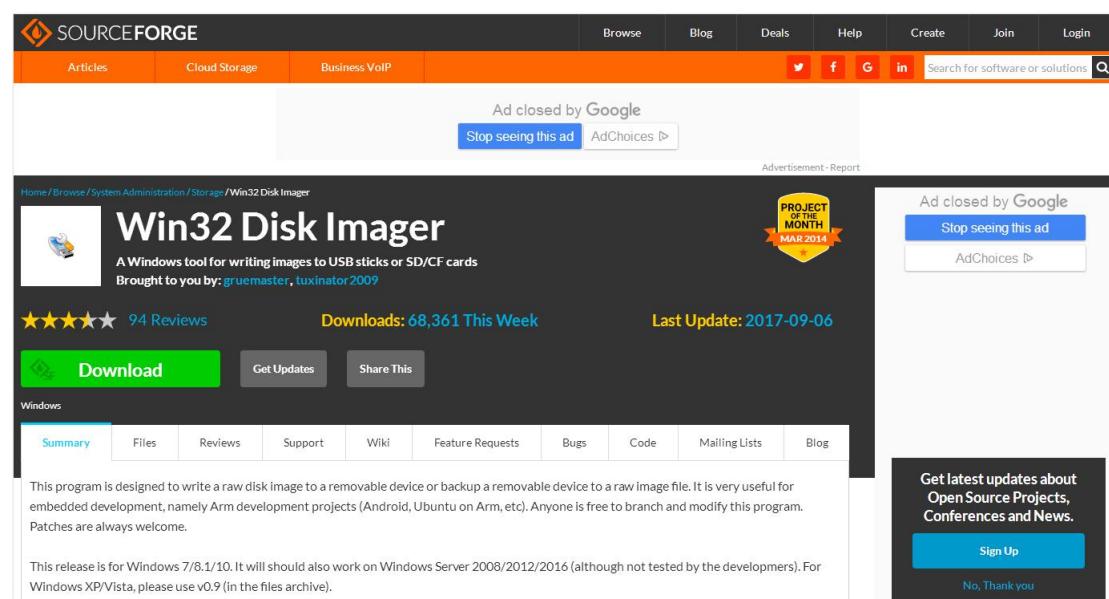
#### Install the Raspbian Operating System

First, install the operating system for the Raspberry Pi. The official system, Raspbian, is recommended. If you've finished the installation and the system works well, you may skip this step.

You need to download the Win32 Disk Imager and burn the operating system to the SD card.

Download the Win32 Disk Imager at:

<https://sourceforge.net/projects/win32diskimager/>

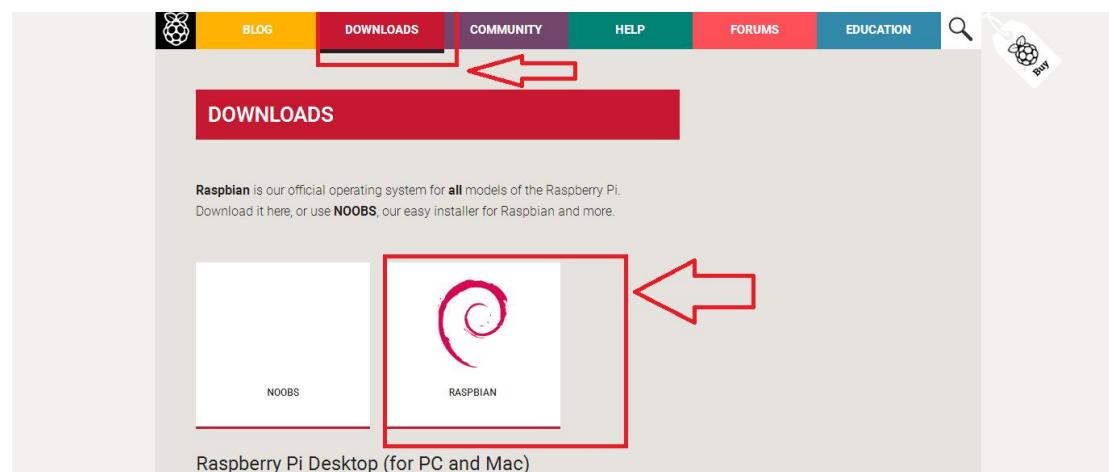


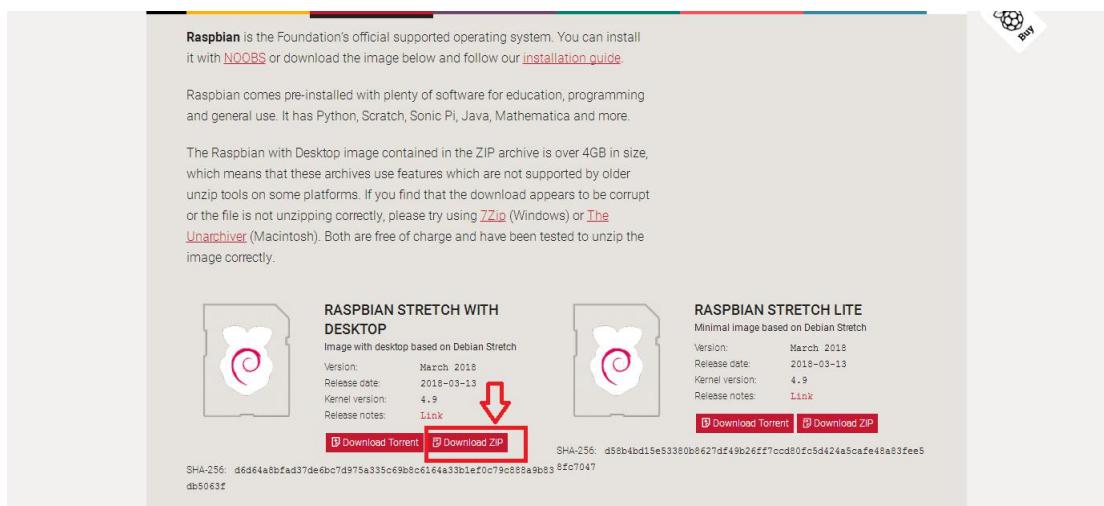
#### Download the Image for Raspbian

Go to Raspberry Pi official website:

<https://www.raspberrypi.org/>

click through **Download->Raspbian**. Raspbian is suitable for novice since it's supported by Raspberry Pi and based on Linux.



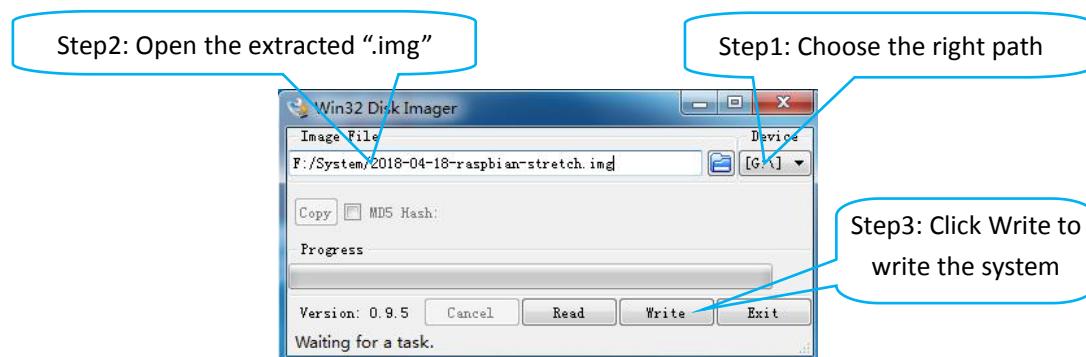


After it's downloaded, unzip it for later SD card system creation.

### Write Raspberry Pi Operating System to SD Card

First, insert the SD card into the card reader and connect it to the USB port of the computer.

Click open the **Win32 Disk Imager** and choose the path of the SD card (here it's Disk G). Click open the **.img** file extracted previously, and click **Write**.



### Display the Filename Extension

For some operations, you may need to change the filename extension (suffix). In some Windows systems, they are hidden by default and you need to make the setting. You may search on the Internet by yourself for how to display the filename extension (suffix) in your own system.

For example, in Windows 7, you may go to **My Computer ->Organization ->Folder and Search->View**, and uncheck the **Hide extensions for known file types**.

### Enable SSH and Setup WiFi

Keep the SD card connected with the computer. Open the root directory of the card and create a file named **ssh** without any suffixes.

Under the root directory of the SD card, create a file **wpa\_supplicant.txt** and write the following contents into the file:

```
country=US
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
```

```
network={  
    ssid="WIFI"  
    psk="PASSWORD"  
    key_mgmt=WPA-PSK  
    priority=1  
}  
country=US  
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev  
update_config=1  
  
network={  
    ssid="WIFI"  
    psk="PASSWORD"  
    key_mgmt=WPA-PSK  
    priority=1  
}
```

In the code above, replace **WIFI** with your own WiFi SSID name and **PASSWORD** with your password for the WiFi network. Save the file and change the name of the file *wpa\_supplicant.txt* into *wpa\_supplicant.conf*.

- \* Make sure MAC filtering has been turned off for the router.
- \* The *WPA-PSK* behind *key\_mgmt=* is the common encryption method for most routers. If the network connection fails, you may log in and check on the router management page.
- \* For more about the network connection for Raspberry Pi, please visit the related page via this link:

<https://www.raspberrypi.org/forums/viewtopic.php?t=203716>

The two files newly created are as shown below:

<input type="checkbox"/> start_x.elf	2018/6/19 12:06	ELF 文件	3,831 KB
<input type="checkbox"/> wpa_supplicant.conf	2018/8/21 16:40	CONF 文件	1 KB
<input type="checkbox"/> ssh	2018/8/21 16:41	文件	0 KB

### Download and Install PuTTy

PuTTy is a software that connects with the Raspberry Pi via ssh. With the tool, you may control the Raspberry Pi by the computer.

Download:

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

The installer packages above will provide all of these (except PuTTYtel), but you can download them one by one.  
(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

**putty.exe (the SSH and Telnet client itself)**

32-bit:	<a href="#">putty.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">putty.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

**pscp.exe (an SCP client, i.e. command-line secure file copy)**

32-bit:	<a href="#">pscp.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">pscp.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

**psftp.exe (an SFTP client, i.e. general file transfer sessions much like FTP)**

32-bit:	<a href="#">psftp.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">psftp.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

**puttytel.exe (a Telnet-only client)**

32-bit:	<a href="#">puttytel.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">puttytel.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

**plink.exe (a command-line interface to the PuTTY back ends)**

32-bit:	<a href="#">plink.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">plink.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

**pageant.exe (an SSH authentication agent for PuTTY, PSCP, PSFTP, and Plink)**

32-bit:	<a href="#">pageant.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">pageant.exe</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

[http://blog.csdn.net/github\\_38111866](http://blog.csdn.net/github_38111866)

## Acquire Raspberry Pi's IP Address

Install the 18650 batteries and switch on the car.

**Method A:** Log in to the router management page on the computer to check the address of the Raspberry Pi.

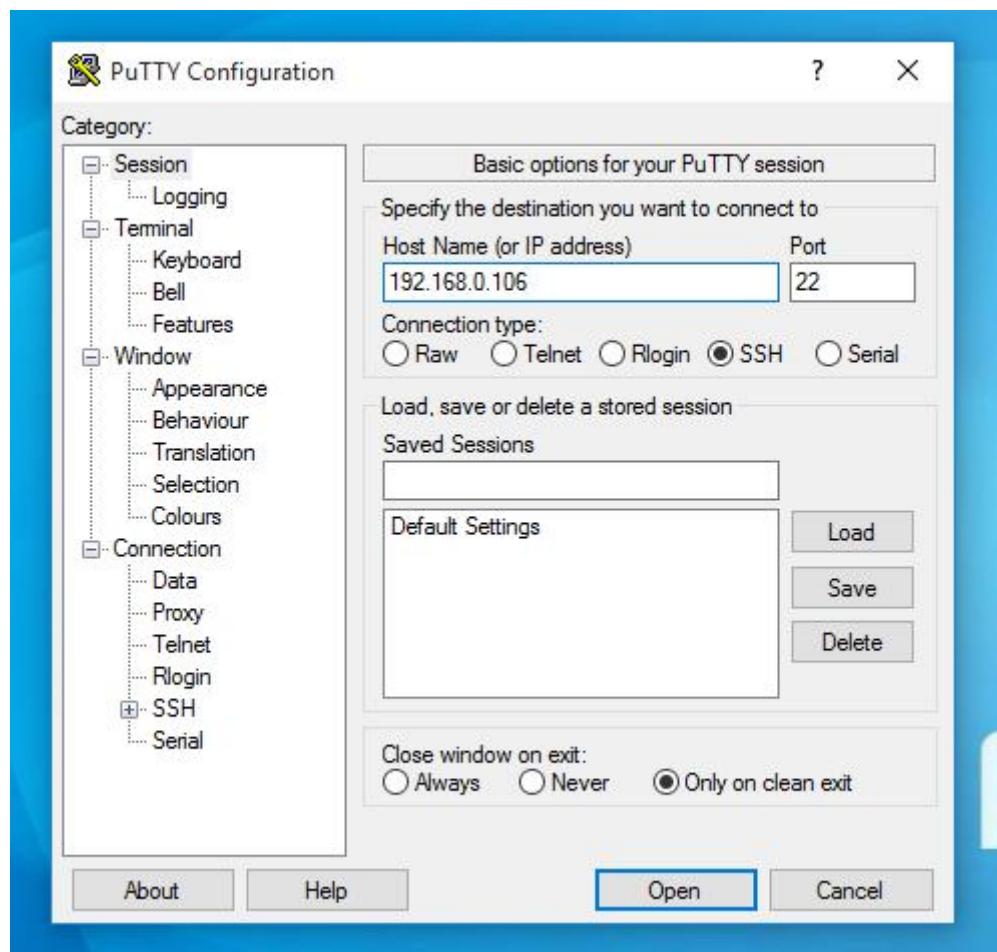
**Method B:** Download the **Network Scanner App** to check the address.

The address of the Raspberry Pi is the one with "Raspberry".

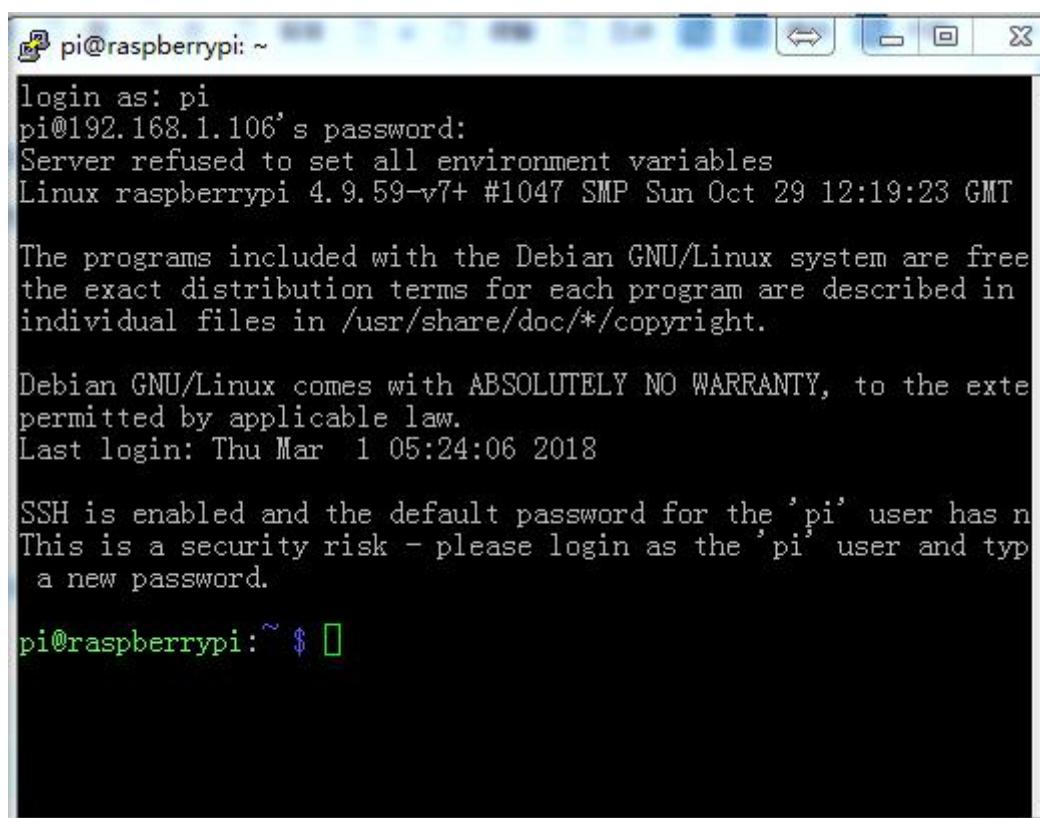
\* The name of the router that the computer or mobile connects should be consistent with the one of the WiFi in the file *wpa\_supplicant.conf* written to the root directory of the SD card in the Raspberry Pi.

## Connect the Raspberry Pi and Computer

Open PuTTY, enter the IP address of the Raspberry Pi in **Host Name (or IP address)** and click **Open**.



If a warning window prompts, click **Yes**.



```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.106's password:
Server refused to set all environment variables
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT
The programs included with the Debian GNU/Linux system are free
the exact distribution terms for each program are described in
individual files in /usr/share/doc/*/*copyright.

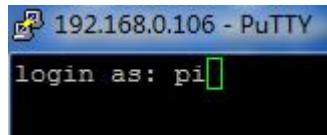
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the exte
permitted by applicable law.
Last login: Thu Mar  1 05:24:06 2018

SSH is enabled and the default password for the 'pi' user has n
This is a security risk - please login as the 'pi' user and typ
a new password.

pi@raspberrypi: ~ $
```

---

Then a terminal will pop up. The default account is pi.



The password for login is *raspberry* by default.

\* When you typing in the password, nothing will appear on the screen but it does not mean no input. Type in the password carefully and press **Enter** after it's done.

Log in successfully.

## 3.2. Download Program

Setting up in a Raspberry Pi may take you a lot of time, and there are too many libraries needed, so we write a python program to do the most of works for you.

**NOTE:** If you want to know the exact details of the program, you can skip this chapter.

**NOTE:** This `setup.py` program is needed to download and install a lot of applications and libraries, and sometimes the server or internet may break down, which may lead to some problems that the `setup.py` could not fix. Then you must set up a raspberry pi yourself, by following the instructions of next chapter named [Set Up a Raspberry Pi](#).

Download the program of the PiCar-B.

Input the code below to download:

```
git clone https://github.com/adeept/adeept_picar-b.git
```

Then setup :

```
sudo python adeept_picar-b/server/setup.py
```

It may take some time to finish.

Now you can skip the next chapter and [Install Python3.7 in the PC](#) directly.

**NOTE:**

The installation process of the Raspberry Pi can take a long time, so we have prepared a test version of the program which only needs to enable i2c and install adafruit\_pca9685.

The test program runs as follows:

```
sudo pip3 install adafruit_pca9685
```

Enable i2c (you can refer to the documentation on 3.3.5 to know how to enable i2c)

```
sudo cp -f //home/pi/adeept_picar-b/server/set.txt //home/pi/set.txt
```

```
sudo python3 adeept_picar-b/server/serverTest.py
```

### 3.3. Set Up a Raspberry Pi

**NOTE:** You should skip this chapter if you finished the last one, and Install Python3.7 in the PC directly.

#### Unrestricted Access to Raspberry Pi

The Linux operating system is a multi-user operating system which allows multiple users to log in and use the computer. Normal users are not normally allowed to edit files in other users' folders or any of the system files. You can run commands as the **root** user by using the **sudo** command before the program you want to run.

More information in: <https://www.raspberrypi.org/documentation/linux/usage/root.md>

#### Set Up A Root User

To change ordinary user into root user on the terminal, you need to setup up a password for the user by entering the following command in the terminal of the RPi:

**sudo passwd root**

Enter the **passwd** twice, as shown below:

```
pi@raspberrypi:~ $ sudo passwd root
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
pi@raspberrypi:~ $
```

#### Log in as Root User

Type in the following command in the terminal of the Raspberry Pi:

**SU –**

Press **Enter**, and type in the password to confirm.

#### Update System

The system you downloaded may not be the latest version and it may cause inconvenience in the subsequent operations. Here we first upgrade the system of the Raspberry Pi. Type in the following command:

**sudo apt-get update**

The **apt-get update** command is to acquire the up-to-date software lists and update all the software, to provide the latest software for next system upgrade.

**sudo apt-get upgrade**

```
scratch2 shared-mime-info systemd systemd-sysv tzdata udev wolfram-e
wolframscript
68 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
Need to get 504 MB of archives.
After this operation, 124 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

The window will prompts "**Do you want to continue?**", type in **Y** and press **Enter**. It may take some time for the system upgrade. During this period, do not carry out any operations with the Raspberry Pi until it's done.

In this process, you may continue reading the following contents to download the Python 3.7and OpenCV under the Windows system.

## 5.Enable I2C and Camera

After the Raspberry Pi is updated, you may continue operations on it.

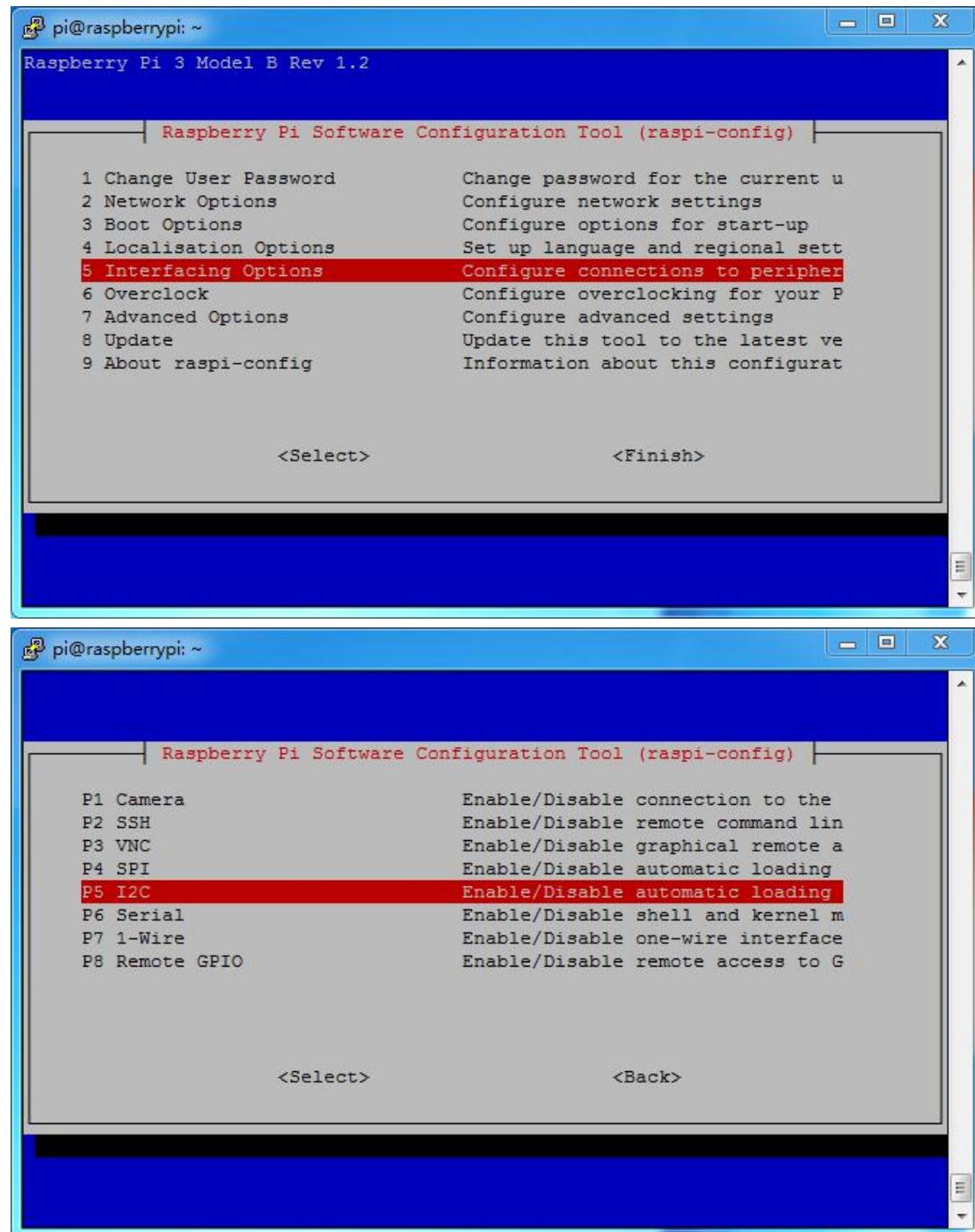
The Adept Motor HAT V1.0 communicates with the Raspberry Pi via the I2C port but the I2C port is disabled by default. You need to enable it:

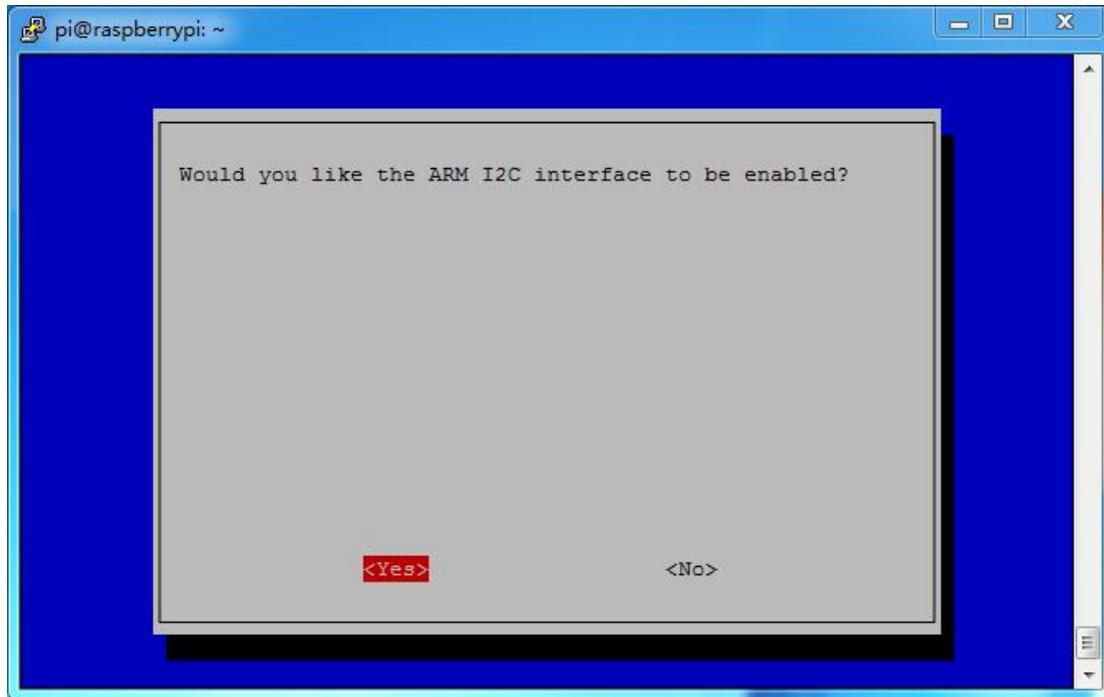
`sudo raspi-config`

A window will pop up.

Click through

**5 Interfacing Options->P5 I2C->Yes->OK->Finish**





Now you've already enabled I2C, move on to enable Camera.

`sudo raspi-config`

Click through:

**5 Interfacing Options ->P1 Camera ->Yes ->OK ->Finish ->Yes**

Reboot the Raspberry Pi. If it does not reboot automatically, type in the following command:

`reboot`

Then the I2C and Camera module are both enabled.

After the Raspberry Pi is rebooted, type in the following command to check that the module has been enabled:

`lsmod | grep i2c`

If the following contents are shown, it indicates the module is enabled successfully.

```
root@raspberrypi:~# sudo raspi-config
root@raspberrypi:~# lsmod | grep i2c
i2c_bcm2835           16384  0
i2c_dev                16384  0
root@raspberrypi:~#
```

## 6.Install I2C-Tools

Install I2C-Tools to check whether the external devices are connected successfully as well as the address of the devices. Type in the command to install:

`sudo apt-get install i2c-tools`

Install the Python drive program for PCA9685:

`sudo pip3 install adafruit-pca9685`

To install the library from source, so that you can use the PCA9685 PWM servo/LED controller with a Raspberry Pi.

Look for more information and example at its homepage:

[https://github.com/adafruit/Adafruit\\_Python\\_PCA9685/](https://github.com/adafruit/Adafruit_Python_PCA9685/)

---

\*Now you may wonder about the difference between apt-get install and pip install, here is the answer:

`pip3` is used to download and install packages directly from PyPI (Python Package Index), hosted by Python Software Foundation. It is a specialized package manager that only deals with python packages.

**Note:** There are two versions of Python in Raspberry Pi by default, for our Python programs are written with Python 3.x(3.7), you need to use `pip3` instead of `pip`, which is used to install software in Python 2.x.

`apt-get` is used to download and install packages from Ubuntu repositories which are hosted by Canonical.

## 7. Install Software for Speech Recognition

First, you need to update the tools for installing:

`sudo pip3 pip setuptools wheel`

`Setuptools` is a package development process library designed to facilitate packaging Python projects by enhancing the Python standard library `distutils` (distribution utilities).

Homepage: <https://github.com/pypa/setuptools>

Python Wheel Package. `WHL` file is a Python Wheel Package. Python is a dynamicobject-oriented programming language. `Wheel` is a built-package format for Python.

A `wheel` is a ZIP-format archive with a specially formatted filename and the `.whl` extension.

Homepage: <https://pythonwheels.com/>

Then you need to install `PyAudio` because we use microphone input, otherwise it will raise an `AttributeError` when the programs attempting to instantiate a microphone object.

`sudo apt-get install portaudio19-dev python3-all-dev python3-pyaudio`

`sudo pip3 install pyaudio`

`sudo apt-get install flac`

`Portaudio19-dev` and `python-all-dev` are the packages that `pyaudio` requires.

Be sure to install the `portaudio` library development package (`portaudio19-dev`) and the `python` development package (`python3-all-dev`) before `pyaudio`.

`PyAudio` can input sound from microphone and then you need `flac` to convert the sound to a `.flac` file for `PocketSphinx` speech recognition.

Now you can install `Speech Recognition` library for performing speech recognition, which with support for several engines and APIs, online and offline.

### Install the latest version of `sphinxbase` and `pocketsphinx`

You can find the downloads here:

<http://cmusphinx.sourceforge.net/wiki/download>

### Type in the command to download:

```
sudo wget
```

<https://sourceforge.net/projects/cmusphinx/files/sphinxbase/5prealpha/sphinxbase-5prealpha.tar.gz/download> -O sphinxbase.tar.gz

```
sudo wget
```

<https://sourceforge.net/projects/cmusphinx/files/pocketsphinx/5prealpha/pocketsphinx-5prealpha.tar.gz/download> -O pocketsphinx.tar.gz

GNU Wget (or just Wget, formerly Geturl, also written as its package name, wget) is a computer program that retrieves content from web servers. It is part of the GNU Project. Its name derives from World Wide Web. It supports downloading via HTTP, HTTPS, and FTP. (<https://en.wikipedia.org/wiki/Wget>)

**NOTE:** The -O in the command above is the letter O , not 0( the number ), meaning the documents will not be written to the appropriate files, but all will be concatenated together and written to file.

### Extract the files into separate directories:

```
sudo tar -xzvf sphinxbase.tar.gz
```

```
sudo tar -xzvf pocketsphinx.tar.gz
```

Tarball( tar ) is a type of format that combines and compress multiple files.

-x: Extract a tar ball.

-z: Decompress and extract the contents of the compressed archive created by gzip program (tar.gz extension)

-v: Verbose output or show progress while extracting files.

-f: Specify an archive or a tarball filename

### Then install bison, ALSA, and swig:

```
sudo apt-get install bison libasound2-dev swig
```

```
sudo apt-get install python3 python3-dev python3-pip build-essential libpulse-dev
```

**Bison** is a program that converts the formal description of a computer language grammar into a C language program that can parse the syntax and symbols of that grammar into instructions that the computer can execute.

**ALSA**, also known as Advanced Linux Sound Architecture, is a software framework and part of the Linux kernel that provides an application programming interface (API) for sound card device drivers.

**Swig**, the Simplified Wrapper and Interface Generator, is an open-source software tool used to connect computer programs or libraries written in C or C++ with other scripting languages, in our case, Python.

More information:

<https://whatis.techtarget.com/definition/Bison>

[https://en.wikipedia.org/wiki/Advanced\\_Linux\\_Sound\\_Architecture](https://en.wikipedia.org/wiki/Advanced_Linux_Sound_Architecture)

<https://en.wikipedia.org/wiki/SWIG>

---

**Compile Sphinxbase:**

```
cd sphinxbase-5prealpha
```

```
./configure --enable-fixed
```

```
make
```

```
sudo make install
```

```
sudo pip3 install pocketsphinx
```

Sphinxbase is a package contains the basic libraries that Pocketsphinx requires.

More information:

<https://github.com/cmusphinx/sphinxbase>

The `./configure` scrip is responsible for getting ready to build the software on your specific system.(Get ready to build the software)

The `make` scrip runs a series of tasks to build the finished program from its source code.  
(Build the software, get ready to install the software)

The `sudo make install` command will copy the built program, and its libraries and documentation, to the correct location.(Install the software)

More information:

<https://robots.thoughtbot.com/the-magic-behind-configure-make-make-install>

**Compile Pocketsphinx:**

Pocketsphinx is a part of CMU Sphinx Open Source Toolkit for Speech Recognition.

More information:

<https://github.com/bambocher/pocketsphinx-python>

```
cd ../pocketsphinx-5prealpha
```

```
./configure
```

```
make
```

```
sudo make install
```

Then you can install SpeechRecognition:

```
sudo pip3 install SpeechRecognition
```

SpeechRecognition is a library for performing speech recognition, with support for several engines and APIs, online and offline.

Homepage: [https://github.com/Uberi/speech\\_recognition#readme](https://github.com/Uberi/speech_recognition#readme)

## OpenCV Installation

```
sudo apt-get install libopencv-dev
```

```
sudo apt-get install python-opencv
```

```
sudo pip3 install imutils
```

```
sudo pip3 install opencv-python
```

The libopencv-dev is a meta package. It has dependencies to many packages that do contain the necessary libraries and header files that OpenCV require.

Python-OpenCV and OpenCV-Python are libraries of Python bindings designed to solve computer vision problems.

More information:

[https://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_setup/py\\_intro/py\\_intro.html](https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_setup/py_intro/py_intro.html)

Imutils contains a series of convenient functions which make basic image processing functions such as translation, rotation, resizing, skeletonization, and displaying Matplotlib images easier with OpenCV.

More information:

<https://github.com/jrosebr1/imutils>

libatlas-base-dev libjasper-dev libqtgui4 python3-pyqt5 libqt4-test are Python packages for FPV functions and sending jpeg stream to PC.

Then you need to install zmq and pybase64 for FPV function:

```
sudo apt-get install libatlas-base-dev libjasper-dev libqtgui4 python3-pyqt5 libqt4-test
```

```
sudo pip3 install zmq pybase64
```

libatlas-base-dev libjasper-dev libqtgui4 python3-pyqt5 libqt4-test are Python packages includes the static libraries and symbolic links needed for program development.

Zmq(ZeroMQ): Carries messages across inproc, IPC, TCP, TIPC, multicast.

Homepage: <http://zeromq.org/>

pybase64: It provides a fast base64 implementation for base64 encoding/decoding.

Homepage: <https://github.com/mayeut/pybase64>

## Download Packages for WS\_2812:

```
sudo pip3 install rpi_ws281x
```

rpi\_ws281x is a Raspberry Pi library for controlling WS281X LEDs.

## Download Program for the PiCar-B

```
git clone https://github.com/adeept/adeept_picar-b.git
```

## Set AP-Hotspot

Configure the Raspberry Pi as the Wi-Fi hotspot mode to build up a direct communication between the PC and the car.

If the RPi Car starts and there is no Wi-Fi to connect with, the RPi Car will set a AP-Hotsposd itself. You can connect it with your PC:

Search Wi-Fi SSID name: AdeeptCar

Password: 12345678

Check the following steps and more information from Github:

[https://github.com/oblique/create\\_ap](https://github.com/oblique/create_ap)

Download the code from Github to a local path for installation:

```
git clone https://github.com/oblique/create_ap
```

```
cd create_ap
```

```
sudo make install
```

Install the dependent libraries:

```
sudo apt-get install util-linux procps hostapd iproute2 iw haveged dnsmasq
```

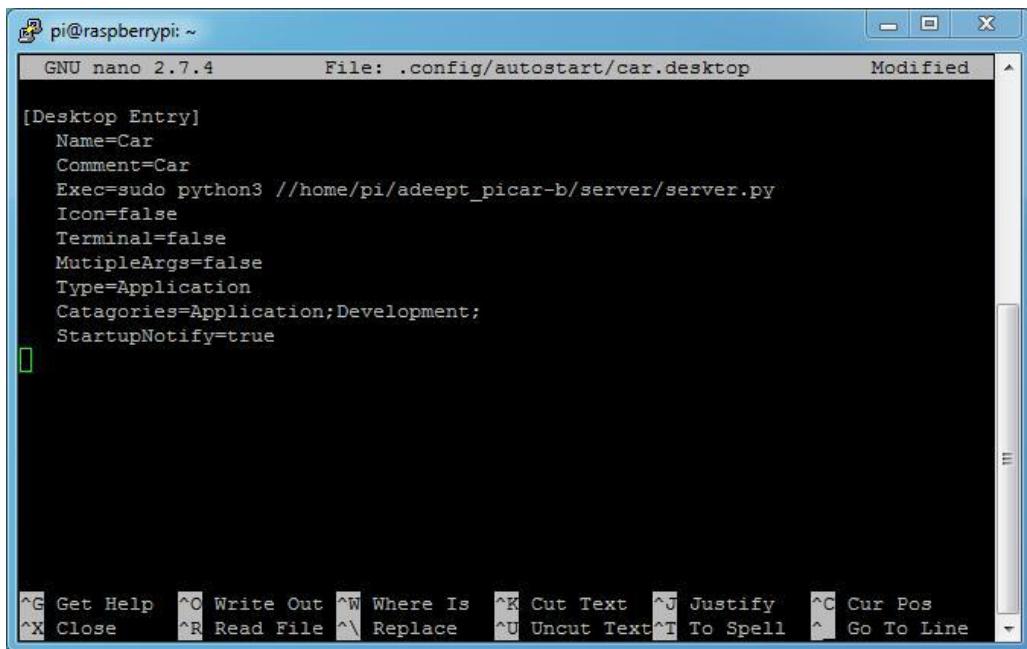
### Add the RPi Car program to auto-start

Create a car.desktop to add the RPi Car program to auto-run

```
sudo nano /home/pi/.config/autostart/car.desktop
```

Add

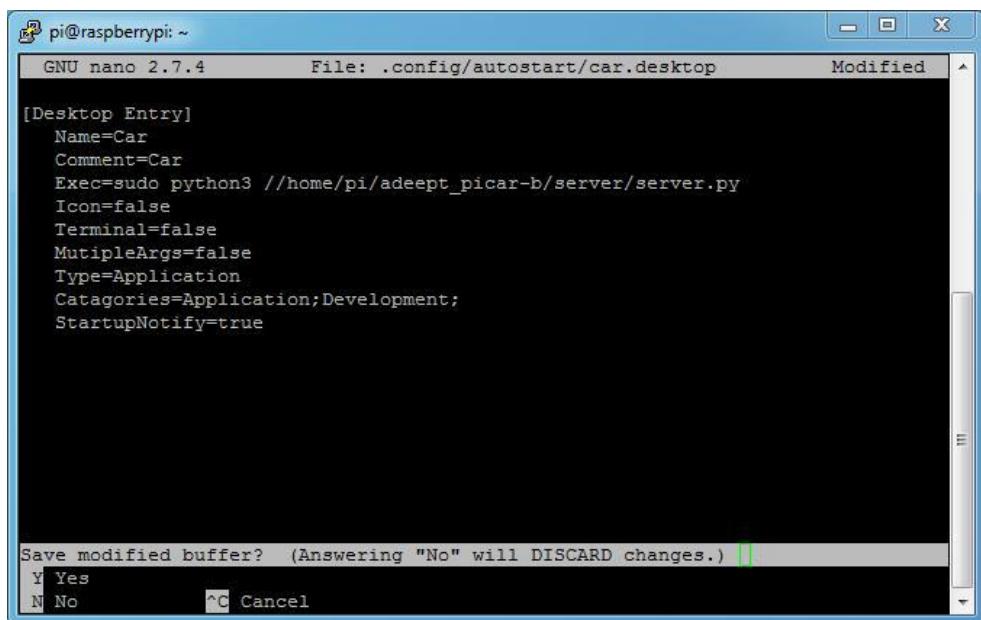
```
[Desktop Entry]
Name=Car
Comment=Car
Exec=sudo python3 /home/adept_picar-b/server/server.py
Icon=false
Terminal=false
MultipleArgs=false
Type=Application
Categories=Application;Development;
StartupNotify=true
```



```
pi@raspberrypi: ~
GNU nano 2.7.4          File: .config/autostart/car.desktop      Modified
[Desktop Entry]
Name=Car
Comment=Car
Exec=sudo python3 //home/pi/adept_picar-b/server/server.py
Icon=false
Terminal=false
MultipleArgs=false
Type=Application
Categories=Application;Development;
StartupNotify=true
```

The screenshot shows a terminal window titled "pi@raspberrypi: ~". Inside the window, the nano text editor is open, displaying the contents of the file ".config/autostart/car.desktop". The file contains the desktop entry configuration provided in the previous text block. The terminal window has a standard blue border and a title bar.

Then press **Ctrl + X** to exit editing



```
pi@raspberrypi: ~
GNU nano 2.7.4          File: .config/autostart/car.desktop      Modified
[Desktop Entry]
Name=Car
Comment=Car
Exec=sudo python3 //home/pi/adeept_picar-b/server/server.py
Icon=false
Terminal=false
MultipleArgs=false
Type=Application
Categories=Application;Development;
StartupNotify=true

Save modified buffer? (Answering "No" will DISCARD changes.) | |
Y Yes
N No           ^C Cancel
```

Input **Y** to save and press **Enter** to confirm.

Then you need to copy the `set.txt` to `/home/pi/` so that the program could find it and load settings. If you set a wrong setting and the car won't work well, you can simply copy it again to replace the old one, and the car will go back to normal.

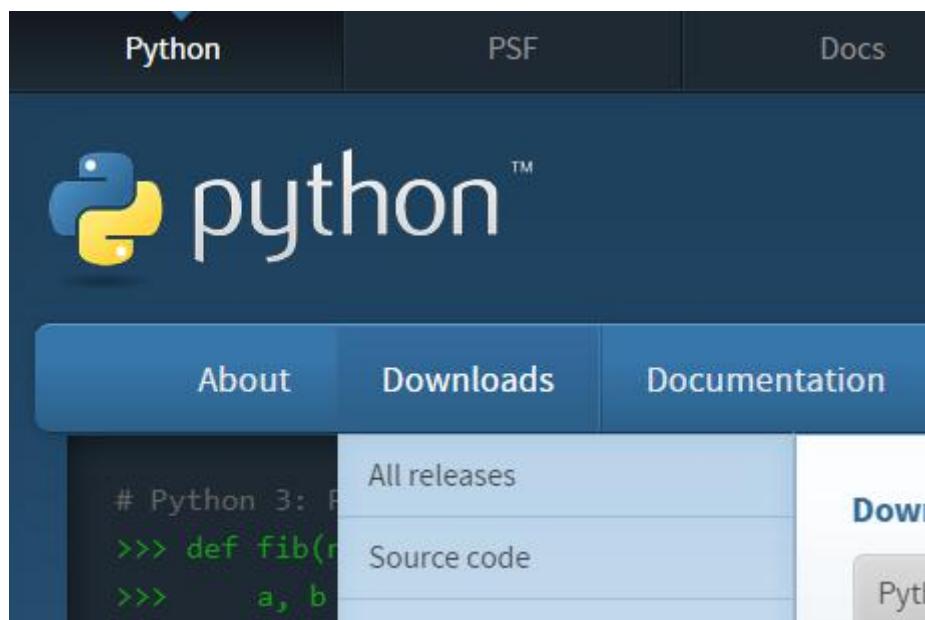
```
sudo cp -f /home/adeept_picar-b/server/set.txt /home/pi/set.txt
```

## 3.4. Install Python3.7 in the PC

### Install Python3.7

So far there are two versions of Python: 2.X and 3.X. The graphical UI of the terminal control is written in Python 3.7 and it supports multiple platforms. Here we'll focus on the installation of Python 3.7 under Windows.

Download Python 3.7: <https://www.python.org/>



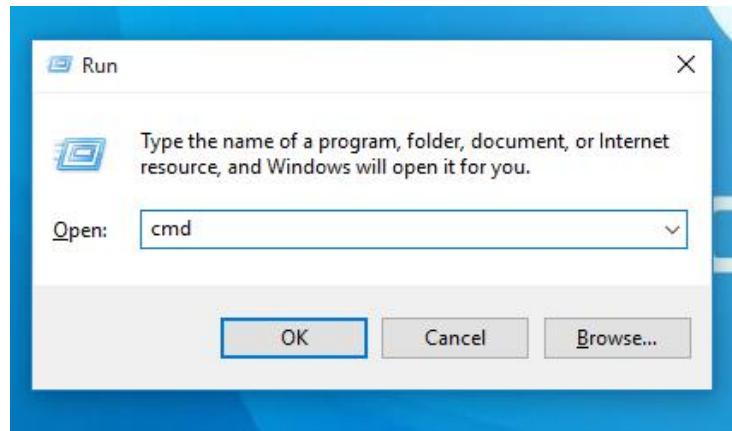
Click through **Downloads->Download Python 3.7.0.**

Install it after download is done. Python will configure the environment variables during the installation.

\*Pay attention to the 32-bit or 64-bit of your system when downloading Python - choose the corresponding Python version based on your own system.

## Install Speech Recognition on Windows

Press **Windows + R** in Window and enter CMD in the textbox



**Click OK**

Type in:

**python -m pip install --upgrade pip setuptools wheel**

to upgrade setup tools

```
C:\Users\Administrator>python -m pip install --upgrade pip setuptools wheel
Collecting pip
  Downloading https://files.pythonhosted.org/packages/5f/25/e52d3f31441505a5f3af41213346e5b6c221c9e086a166f3703d2ddaf940
/pip-18.0-py2.py3-none-any.whl (1.3MB)
  100% [██████████] | 1.3MB 2.2MB/s
Collecting setuptools
  Downloading https://files.pythonhosted.org/packages/66/e8/570bb5ca88a8bcd2a1db9c6246bb66615750663ffaaeada95b04ffe74e12
/setuptools-40.2.0-py2.py3-none-any.whl (568kB)
  100% [██████████] | 573kB 3.3MB/s
Requirement already up-to-date: wheel in c:\users\administrator\appdata\local\programs\python37\lib\site-packages
(0.31.1)
Installing collected packages: pip, setuptools
  Found existing installation: pip 10.0.1
    Uninstalling pip-10.0.1:
      Successfully uninstalled pip-10.0.1
  Found existing installation: setuptools 39.0.1
    Uninstalling setuptools-39.0.1:
      Successfully uninstalled setuptools-39.0.1
Successfully installed pip-18.0 setuptools-40.2.0
```

**pip install SpeechRecognition**

to install Speech Recognition for Python

**NOTE:** If Python3.7 is the only version in your PC, you can use both **pip** and **pip3** to install software, but when you also have Python2.x installed, you must use **pip3** to install it in your Python3.7 library.

```
C:\Users\Administrator>pip3 install SpeechRecognition
Collecting SpeechRecognition
  Downloading https://files.pythonhosted.org/packages/26/e1/7f5678cd94ec1234269d23756dbdaa4c8cfaed973412f38ae8adf7893a50
/SpeechRecognition-3.8.1-py2.py3-none-any.whl (32.8MB)
  100% [██████████] | 32.8MB 109kB/s
Installing collected packages: SpeechRecognition
Successfully installed SpeechRecognition-3.8.1
```

## Install PyAudio

Download pyaudio.whl to install PyAudio or you may install Visual C++ 14.0 to build the wheel, which takes a lot of space and time.

You don't have to install Visual C++ to build the wheel, you can download the wheel from [www.lfd.uci.edu](http://www.lfd.uci.edu), but you must choose the right version of wheel for your Python.

<https://www.lfd.uci.edu/~gohlke/pythonlibs/#pyaudio>

**PyAudio** provides bindings for the PortAudio library.

Includes ASIO, DS, WMME, WASAPI, WDMKS support.

[PyAudio-0.2.11-cp27-cp27m-win32.whl](#)

[PyAudio-0.2.11-cp27-cp27m-win\\_amd64.whl](#)

[PyAudio-0.2.11-cp34-cp34m-win32.whl](#)

[PyAudio-0.2.11-cp34-cp34m-win\\_amd64.whl](#)

[PyAudio-0.2.11-cp35-cp35m-win32.whl](#)

[PyAudio-0.2.11-cp35-cp35m-win\\_amd64.whl](#)

[PyAudio-0.2.11-cp36-cp36m-win32.whl](#)

[PyAudio-0.2.11-cp36-cp36m-win\\_amd64.whl](#)

[PyAudio-0.2.11-cp37-cp37m-win32.whl](#)

[PyAudio-0.2.11-cp37-cp37m-win\\_amd64.whl](#)

In our case, we download [PyAudio-0.2.11-cp37-cp37m-win\\_amd64.whl](#) for [Python3.7](#) on [x64 OS](#).

(If you are using a x86 system, you need to download [PyAudio-0.2.11-cp37-cp37m-win32.whl](#))

Download it and save it in [C:\Users\Administrator\](#) so you don't have to input the path when installing the wheel.

Install PyAudio using pip:

`pip3 install PyAudio-0.2.11-cp37-cp37m-win_amd64.whl`

That is a long name but you can input [pip3 install Py](#) and then press the **Tab** on the left side of the keyboard to let the computer finish it.

```
C:\Users\Administrator>pip3 install PyAudio-0.2.11-cp37-cp37m-win_amd64.whl
Processing c:\users\administ...PyAudio-0.2.11-cp37-cp37m-win_amd64.whl
Installing collected packages: PyAudio
Successfully installed PyAudio-0.2.11

C:\Users\Administrator>
```

## Download Swig:

<http://www.swig.org/download.html>

### The Latest Release

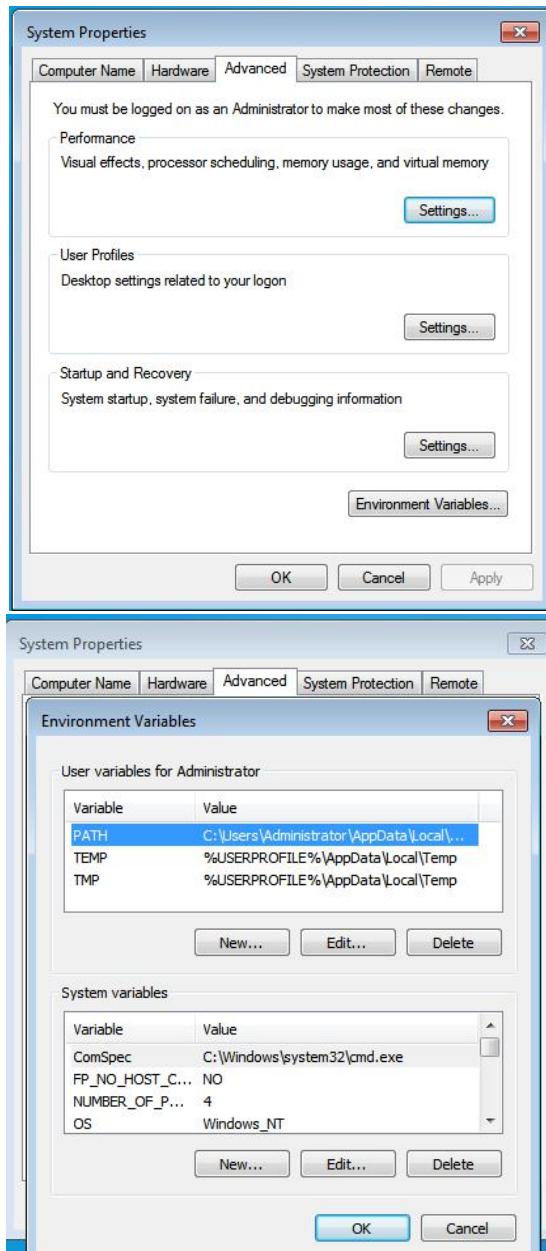
The latest release is [swig-3.0.12](#). View the [release notes](#).

Windows users should download [swigwin-3.0.12](#) which includes a prebuilt executable.

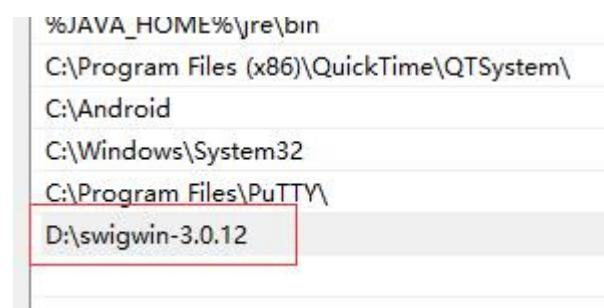
Many Unix-like operating systems also include packages of SWIG (e.g. Debian GNU/Linux, FreeBSD, Cygwin). Consult your package management application to see if your operating system does.

Set the environment variable to point to the "swig" 's executable which is under the root directory of swig.

Just hit **Shift** and **R** at the same time to get command prompt. Then type **sysdm.cpl**, go to Advanced and select Environmental Variables.



Double click **Path** and add the path of Swig.exe.



## Download PocketSphinx.whl:

<https://www.lfd.uci.edu/~gohlke/pythonlibs/#pocketsphinx>

**Pocketsphinx**, an interface to the CMU SphinxBase and PocketSphinx libraries.

[pocketsphinx-0.1.15-cp27-cp27m-win32.whl](#)  
[pocketsphinx-0.1.15-cp27-cp27m-win\\_amd64.whl](#)  
[pocketsphinx-0.1.15-cp34-cp34m-win32.whl](#)  
[pocketsphinx-0.1.15-cp34-cp34m-win\\_amd64.whl](#)  
[pocketsphinx-0.1.15-cp35-cp35m-win32.whl](#)  
[pocketsphinx-0.1.15-cp35-cp35m-win\\_amd64.whl](#)  
[pocketsphinx-0.1.15-cp36-cp36m-win32.whl](#)  
[pocketsphinx-0.1.15-cp36-cp36m-win\\_amd64.whl](#)  
[pocketsphinx-0.1.15-cp37-cp37m-win32.whl](#)  
[pocketsphinx-0.1.15-cp37-cp37m-win\\_amd64.whl](#)

In our case, we download [pocketsphinx-0.1.15-cp37-cp37m-win\\_amd64.whl](#) for Python3.7 on x64 OS.

Download it and save it in C:\Users\Administrator\ so you don't have to input the path when installing it.

Now you can install PocketSphinx:

`pip3 install pocketsphinx-0.1.15-cp37-cp37m-win_amd64.whl`

```
C:\Users\Administrator>pip3 install pocketsphinx-0.1.15-cp37-cp37m-win_amd64.whl
Processing c:\users\administrator\pocketsphinx-0.1.15-cp37-cp37m-win_amd64.whl
Installing collected packages: pocketsphinx
Successfully installed pocketsphinx-0.1.15
```

## Install OpenCV:

Install numpy:

`pip3 install numpy`

NumPy is a general-purpose array-processing package designed to efficiently manipulate large multi-dimensional arrays of arbitrary records without sacrificing too much speed for small multi-dimensional arrays.

Download OpenCV\_python.whl:

<https://www.lfd.uci.edu/~gohlke/pythonlibs/#opencv>

**OpenCV**, a real time computer vision library.

[opencv\\_python-2.4.13.5-cp27-cp27m-win32.whl](#)  
[opencv\\_python-2.4.13.5-cp27-cp27m-win\\_amd64.whl](#)  
[opencv\\_python-3.1.0-cp34-cp34m-win32.whl](#)  
[opencv\\_python-3.1.0-cp34-cp34m-win\\_amd64.whl](#)  
[opencv\\_python-3.4.3+contrib-cp35-cp35m-win32.whl](#)  
[opencv\\_python-3.4.3+contrib-cp35-cp35m-win\\_amd64.whl](#)  
[opencv\\_python-3.4.3+contrib-cp36-cp36m-win32.whl](#)  
[opencv\\_python-3.4.3+contrib-cp36-cp36m-win\\_amd64.whl](#)  
[opencv\\_python-3.4.3+contrib-cp37-cp37m-win32.whl](#)  
[opencv\\_python-3.4.3+contrib-cp37-cp37m-win\\_amd64.whl](#)  
[opencv\\_python-3.4.3-cp35-cp35m-win32.whl](#)  
[opencv\\_python-3.4.3-cp35-cp35m-win\\_amd64.whl](#)  
[opencv\\_python-3.4.3-cp36-cp36m-win32.whl](#)  
[opencv\\_python-3.4.3-cp36-cp36m-win\\_amd64.whl](#)  
[opencv\\_python-3.4.3-cp37-cp37m-win32.whl](#)  
[opencv\\_python-3.4.3-cp37-cp37m-win\\_amd64.whl](#)

In our case, we download [opencv\\_python-3.4.3-cp37-cp37m-win\\_amd64.whl](#) for Python3.7 on x64 OS.

Download it and save it in `C:\Users\Administrator\` so you don't have to input the path when installing.

Now you can install OpenCV\_python:

`pip3 install opencv_python-3.4.3-cp37-cp37m-win_amd64.whl`

And then you need to install zmq and pybase64(same reason in RPi):

`pip3 install zmq pybase64`

## 3.5. Run the PiCar-B

### Start

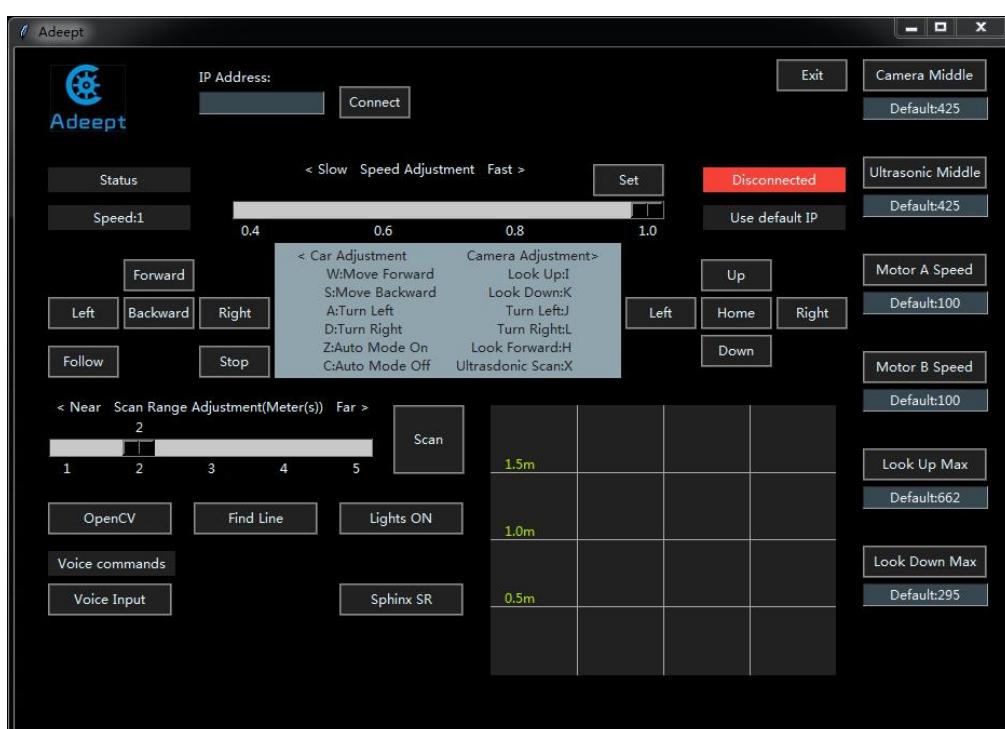
Switch on the car.

After a while, if the LEDs turn **red**, it means the car's server is connected to a Wi-Fi waiting for the PC client to join.

If there is no Wi-Fi for the car to connect with, the LEDs turn **blue**, it means the car has set up an AP-Hotspot, you can use your PC to search it, the RPi Car's AP-Hotspot's SSID name is **AdeeptCar** and password is **12345678**.

Then implement operations in Windows.

Double click to run the file **client.py** in the folder **client**.



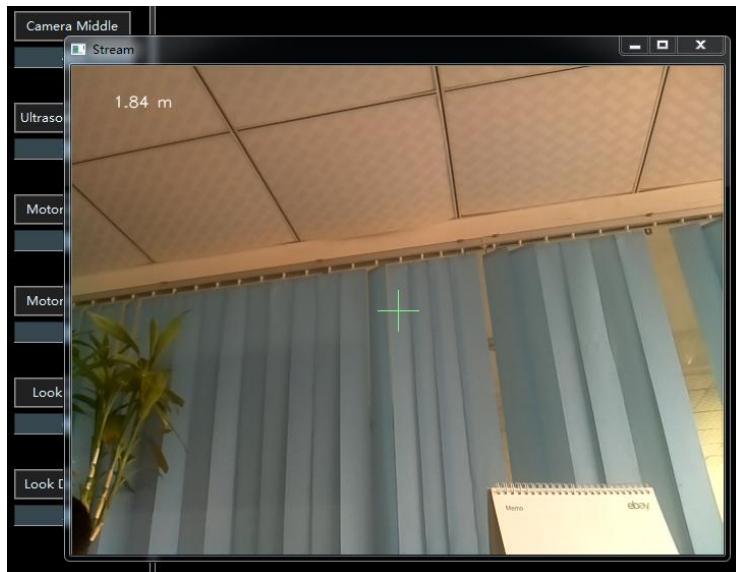
For initial running, you need to enter the IP address of the Raspberry Pi car **IP Address**, then click **Connect**, and the program will connect to the Raspberry Pi.



After connection, the program will save the IP address. For the next use, if the IP address of the Raspberry Pi has not changed, you may press **Enter** directly next time to connect.



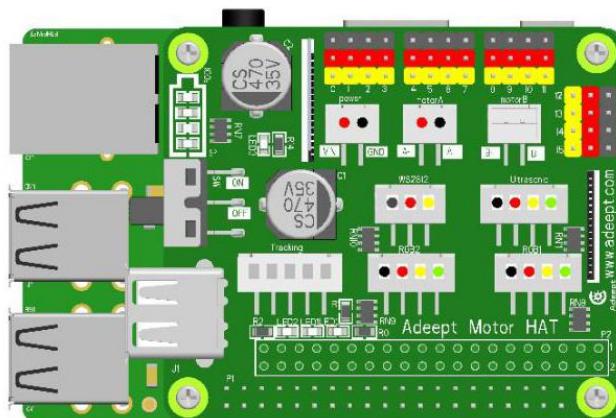
After the connection is made successfully, the Video window shows up.



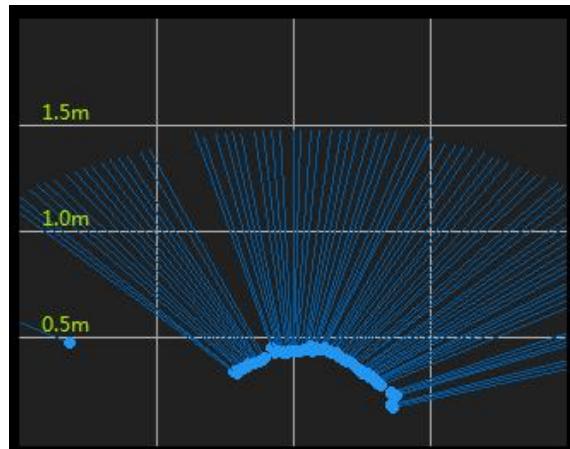
Now you may control the car by the keyboard based on the instructions on the GUI.



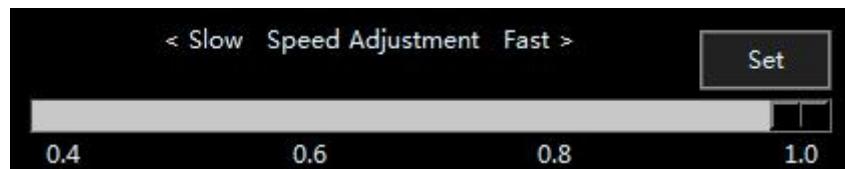
**NOTE:** If the motor isn't working at the right direction, you should unplug the motors from A+ A- and plug it into B+ B-.



Press **X** to implement ultrasonic scanning. During this process, the car is unable to execute other actions. After the scanning is done, the results will be shown on the GUI as shown below:

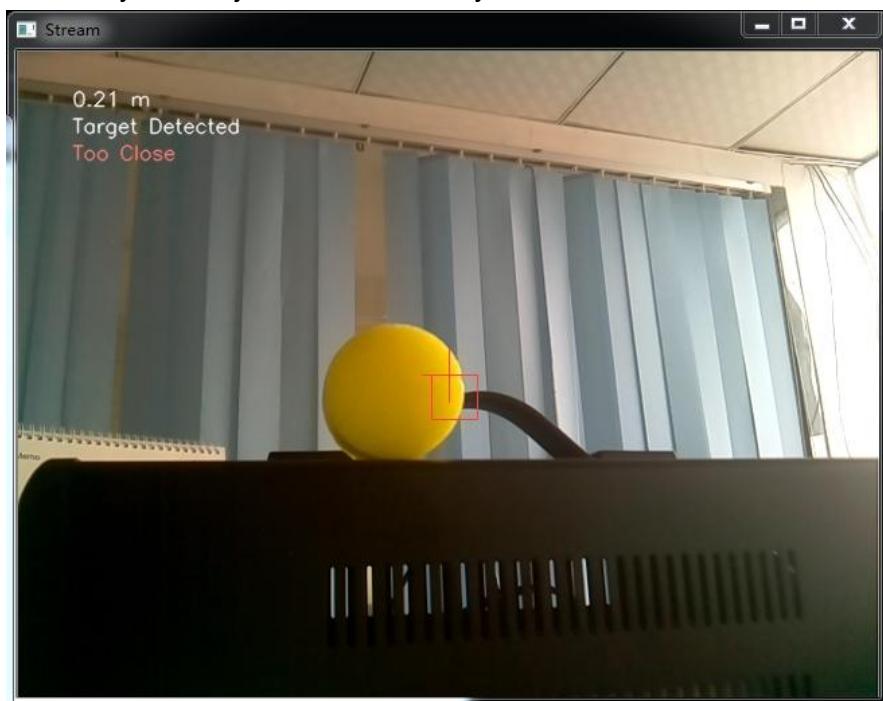


You may select the range of scanning in the upper scale and set the speed of car running in the bottom one. After all changes, you need to click **Set** button to send the new data to the car.



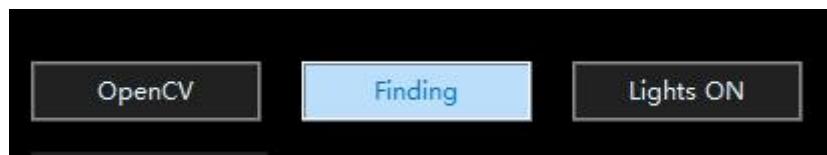
### OpenCV Vision Recognition

By default, the Rover finds the biggest yellow object in its view and follows it with its LEDs turned green. When it gets close enough, it would stop and the LEDs turn blue, and if it gets too close to the yellow object, it would go back with the LEDs turned red; when the rover couldn't find a yellow object, the LEDs turn yellow.



## Line Following Mode

The Rover can track lines and follow them, proceeding along a preset path that can be altered by moving the lines, and this part of Python program is easy to understand. You can open *findline.py* and learn to write it yourself.



## Ultrasonic Follow and Stay in a Distance

Ultrasonic keep scanning forward, if there is an object in the follow range (*distance\_stay* value default: 2m), the Rover would move toward it and stay in a certain distance (*distance\_range* value default: 0.4m), both of the values can be changed in *server.py* in the Rover.

```
82  distance_stay = 0.4  
83  distance_range = 2
```

## Modes Stop and Led Control

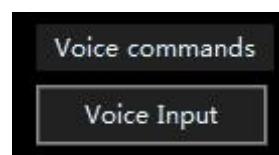
When you've start a Mode such as **OpenCV vision recognition** or **Find Line**, you can click **Stop** to switch to normal mode, or you can **press C** on keyboard to do so.

You can click **Lights ON** to light up LEDs or you can **press F** on keyboard and click or press again to turn them off.

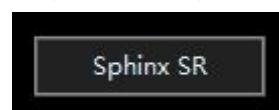


## Speech Control

There is a **Voice Input** button on the GUI, click it and the mic on **PC** would listen to your voice command and control the Rover.



The Label above the Voice Input button shows the speech recognition result so you can tell whether the computer understands you correctly.



---

The Sphinx SR button calls the Speech Recognition program in the Rover, click it and you can directly tell the Rover what to do, it can understand simple command such as ‘forward’ ‘backward’ ‘turn left’ ‘turn right’ and ‘stop’, you can also add your own command by change the code in *speech.py*.

```
if v_command == 'forward':
    motor.motor_left(status, forward, left_spd*spd_ad_2)
    motor.motor_right(status, forward, right_spd*spd_ad_2)

elif v_command == 'backward':
    motor.motor_left(status, backward, left_spd)
    motor.motor_right(status, backward, right_spd)

elif v_command == 'left':
    turn.left()

elif v_command == "right":
    turn.right()

elif v_command == 'stop':
    motor.motorStop()

else:
    pass
```

## Safe Shutdown

You may notice there's no such thing as a power button for the Raspberry Pi as for PC. Most people would directly unplug the power cable for the Raspberry Pi, which may cause damage to the Raspberry Pi and SD card, data loss, etc. To avoid such issues, you need a safe shutdown for the Raspberry Pi.

If you just use the Raspberry Pi independently, you may shut it down with the following command:

```
sudo shutdown -h now
```

When the green light stops blinking on the Raspberry Pi, you may unplug the power cable. If you're applying the Raspberry Pi smart car, you may tab the Exit button in the app of this product. When the green light stops blinking on the Raspberry Pi, switch to OFF for the Power switch on Shield and you can shut down the Raspberry Pi then.

## 4. Afterword

Thanks for purchasing our product and reading the manual! If you spot any errors or have any ideas or questions for the product and this guide, welcome to contact us! We will correct them if any as quickly as possible.

After completing all projects in the guide, you should have some knowledge of the Raspberry Pi and Robot, thus you can try to change the robot into other projects by adding more Adeept modules or changing the code for extended functions.

For more information about Arduino, Raspberry Pi, Smart car robot, or robotics, etc., please follow our website [www.adeept.com](http://www.adeept.com). We will introduce more cost-effective, innovative and intriguing products!

Thanks again for choose Adeept product and service!



Adeept

Sharing Perfects Innovation

E-mail: support@adeept.com

website: [www.adeept.com](http://www.adeept.com)