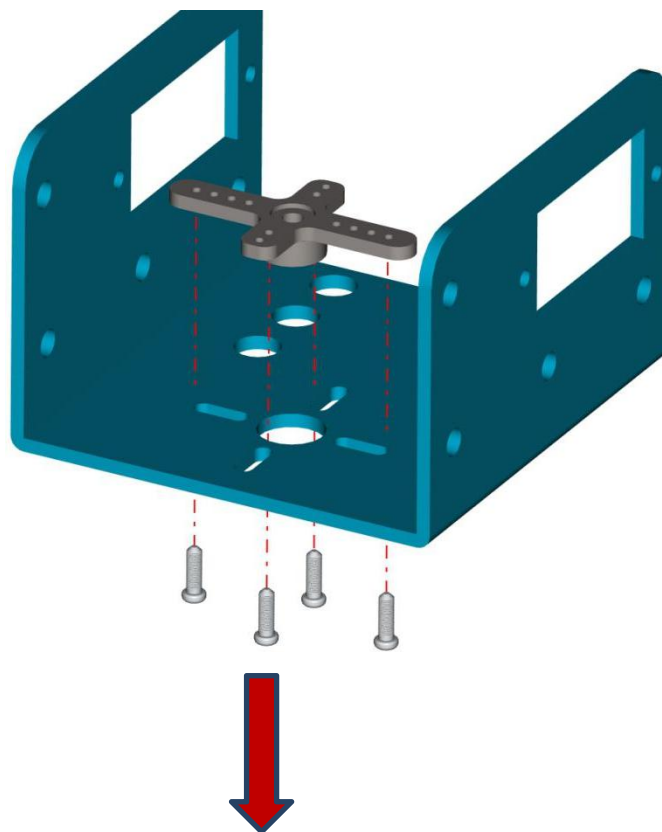


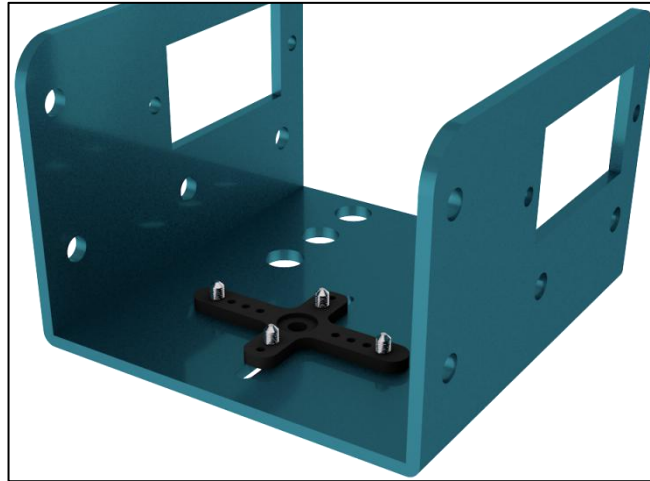


Install the claw of the robot arm:

- 4pcs M1.2*4MM self-tapping screws
- A black servo horn(belong to the servo)
- 1pcs M2*5 screw (belong to the servo)

Mount a black servo horn onto the corresponding metal sheet with 4pcs M1.2*4mm self-taping screws, then we get the accessory 6





accessory 6

- 2 pcs M2*8mm round-head screws
- 2 pcs M2 nuts
- 180° metal gear servo
- Accessory 6

A. Need to set the angle of the servo to 60°

When setting the angle of the servo, connect the servo to A0 end of the shield(the silk print at the back), then stack the shield onto the control board, plug in power and press the reset button. Then the servo will rotate to 60°



Set a servo to 60°:

```
int servopin=A0;//Define digital interface A0 to connect servo steering
gear signal line

int myangle;      //Define angle variables

int pulsewidth; //Define pulse width variables

void setup()
{
  pinMode(servopin,OUTPUT); //Set steering gear interface as Output
}

void servopulse(int servopin,int myangle) //Define Function
{
  pulsewidth=(myangle*11)+500; //Converts the Angle to a pulse width
value of 500 - 2480

  digitalWrite(servopin,HIGH); //The steering gear interface level is high
  delayMicroseconds(pulsewidth); //The microsecond number of the delay
pulse width value

  digitalWrite(servopin,LOW); //Turn the steering gear interface level to low
  delay(20-pulsewidth/1000);
}

void loop()
```



```
{  
servopulse(servopin,60); //Set steering gear Angle  
}
```

//0 Degree Code:

```
// servopulse(servopin,0);
```

//90 Degree Code:

```
// servopulse(servopin,90);
```

//180 Degree Code:

```
// servopulse(servopin,180);
```



```
2 | Arduino 1.8.9
File Edit Sketch Tools Help

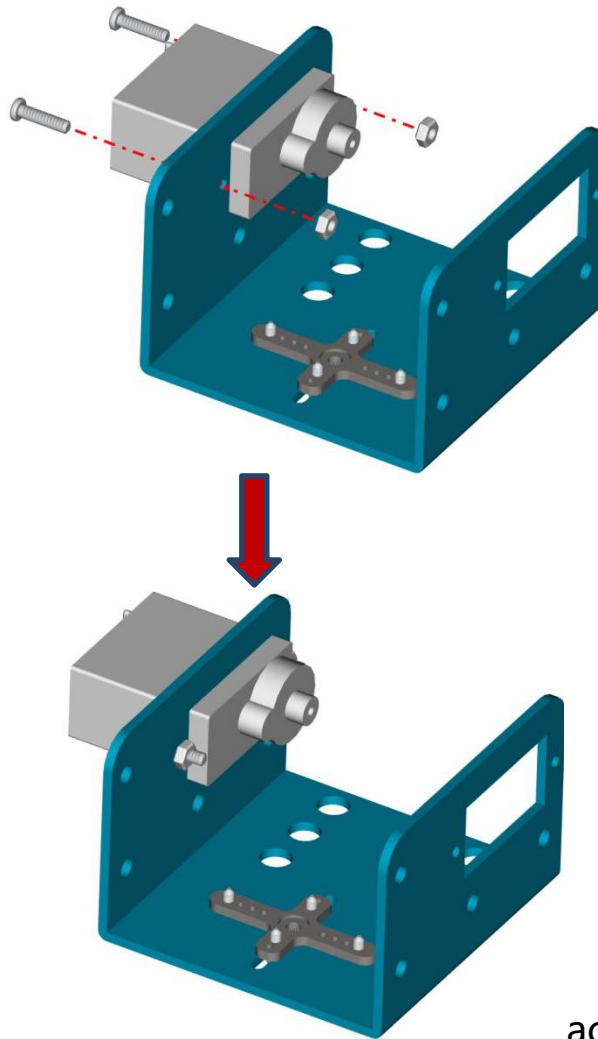
2 §

int servopin=A0; //Define digital interface A0 to connect servo steering gear signal line
int myangle; //Define angle variables
int pulsewidth; //Define pulse width variables
void setup()
{
  pinMode(servopin,OUTPUT); //Set steering gear interface as Output
}
void servopulse(int servopin,int myangle) //Define Function
{
  pulsewidth=(myangle*11)+500; //Converts the Angle to a pulse width value of 500 - 2480
  digitalWrite(servopin,HIGH); //The steering gear interface level is high
  delayMicroseconds(pulsewidth); //The microsecond number of the delay pulse width value
  digitalWrite(servopin,LOW); //Turn the steering gear interface level to low
  delay(20-pulsewidth/1000);
}
void loop()
{
  servopulse(servopin,60); //Set steering gear Angle
}

Done uploading.
Sketch uses 926 bytes (2%) of program storage space. Maximum is 32256 bytes.
Global variables use 11 bytes (0%) of dynamic memory, leaving 2037 bytes for local variables.

6 Arduino/Genuino Uno on COM18
```

Remove this servo, use 2pcs M2*8mm round-head screws and 2 pcs hex nuts to install the 180° metal servo onto the accessory 6. Then we get the accessory 7.



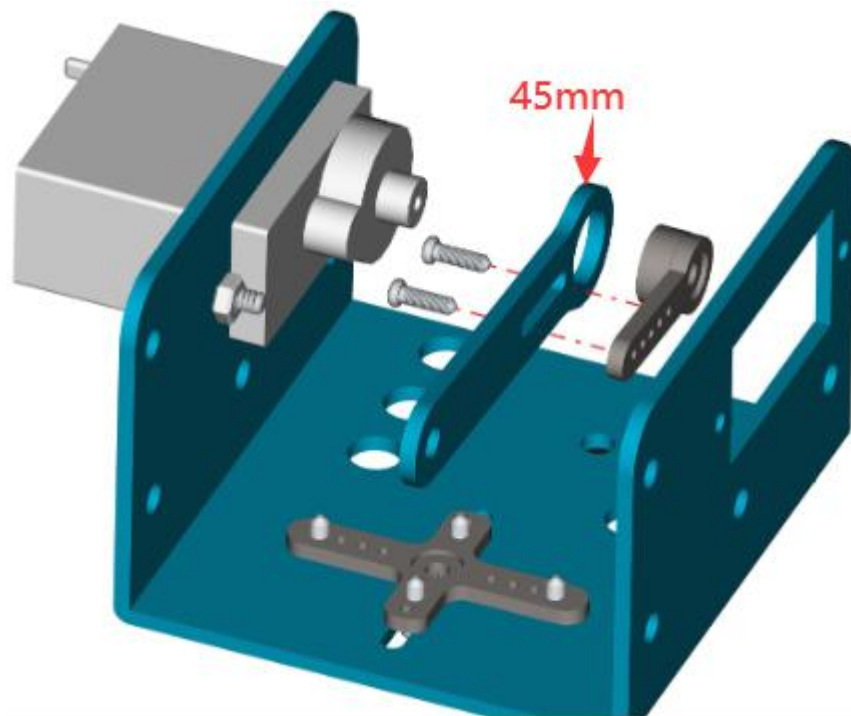
accessory7

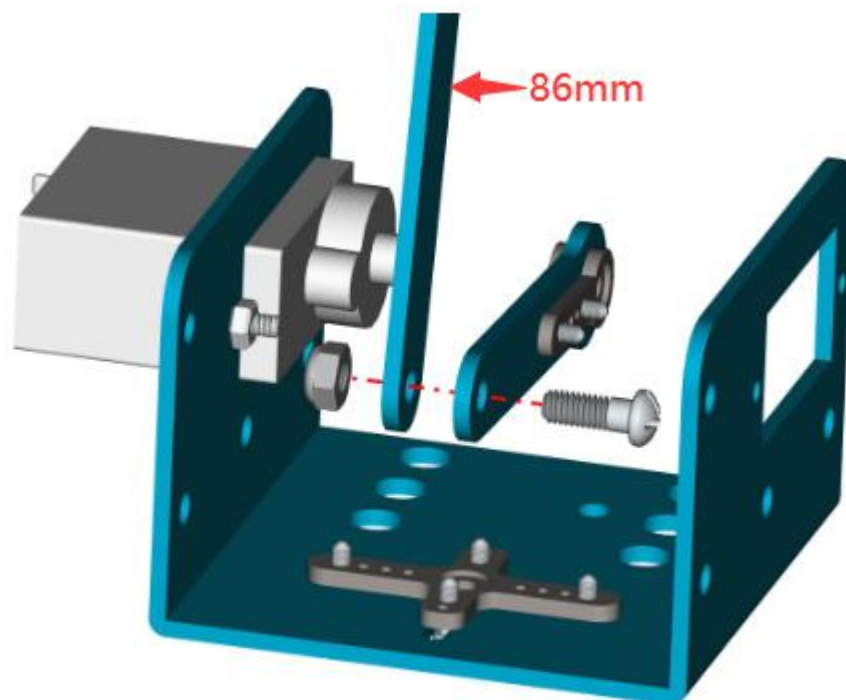
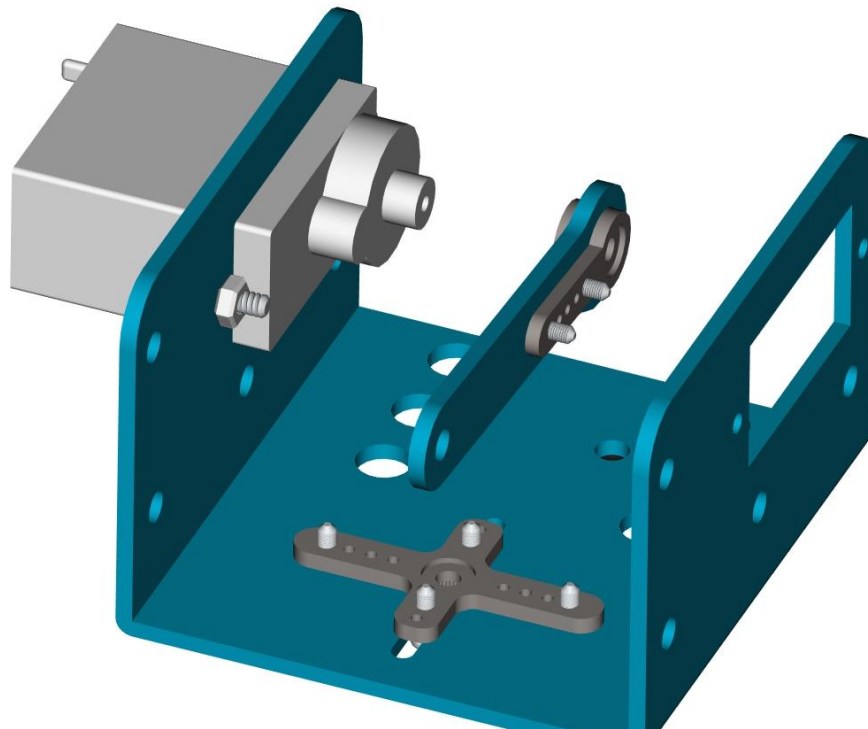
- 2pcs M1.2*4MM self-tapping screws
- 1pcs M3*8MM round-head screw
- 1pcs M3 nut
- 1pcs 45mm metal sheet
- 1pcs 86mm metal sheet
- 1 pcs M2*5 screw(belong to the servo)
- A black horn (belong to the servo)

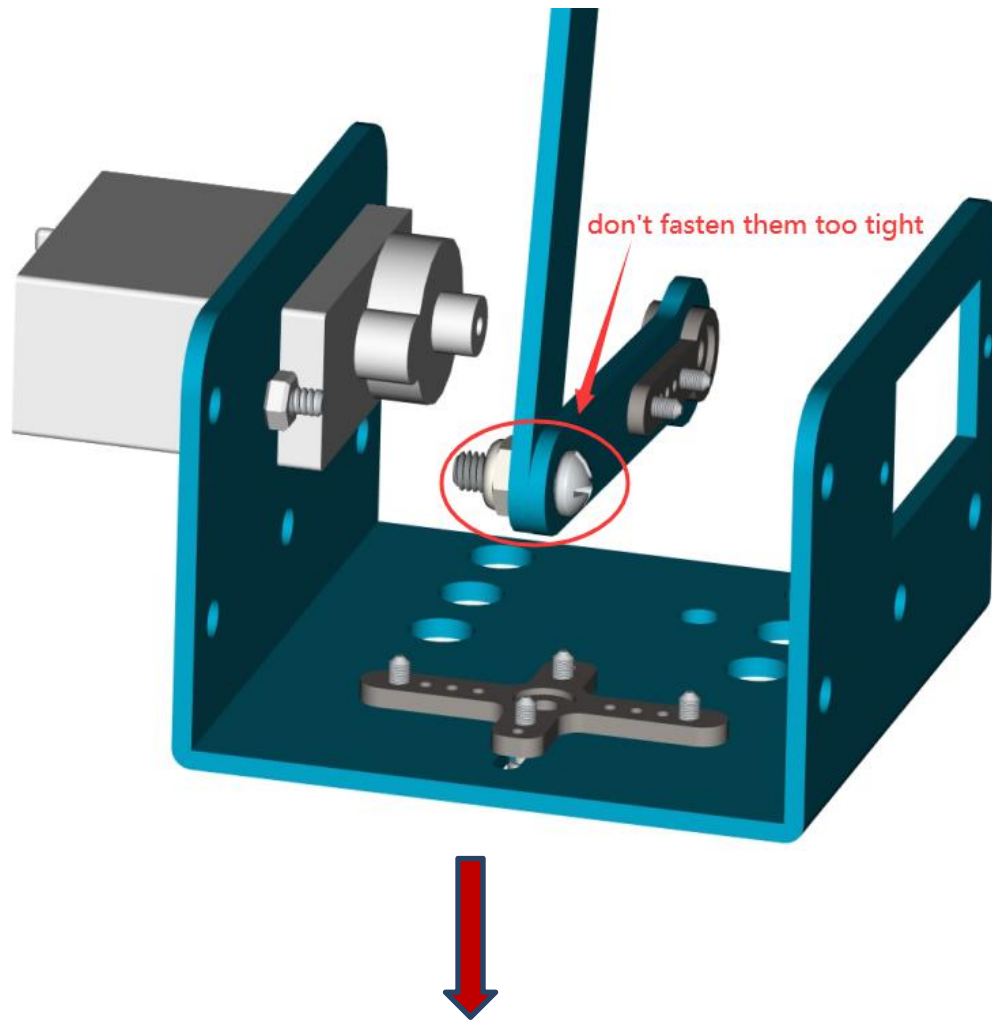


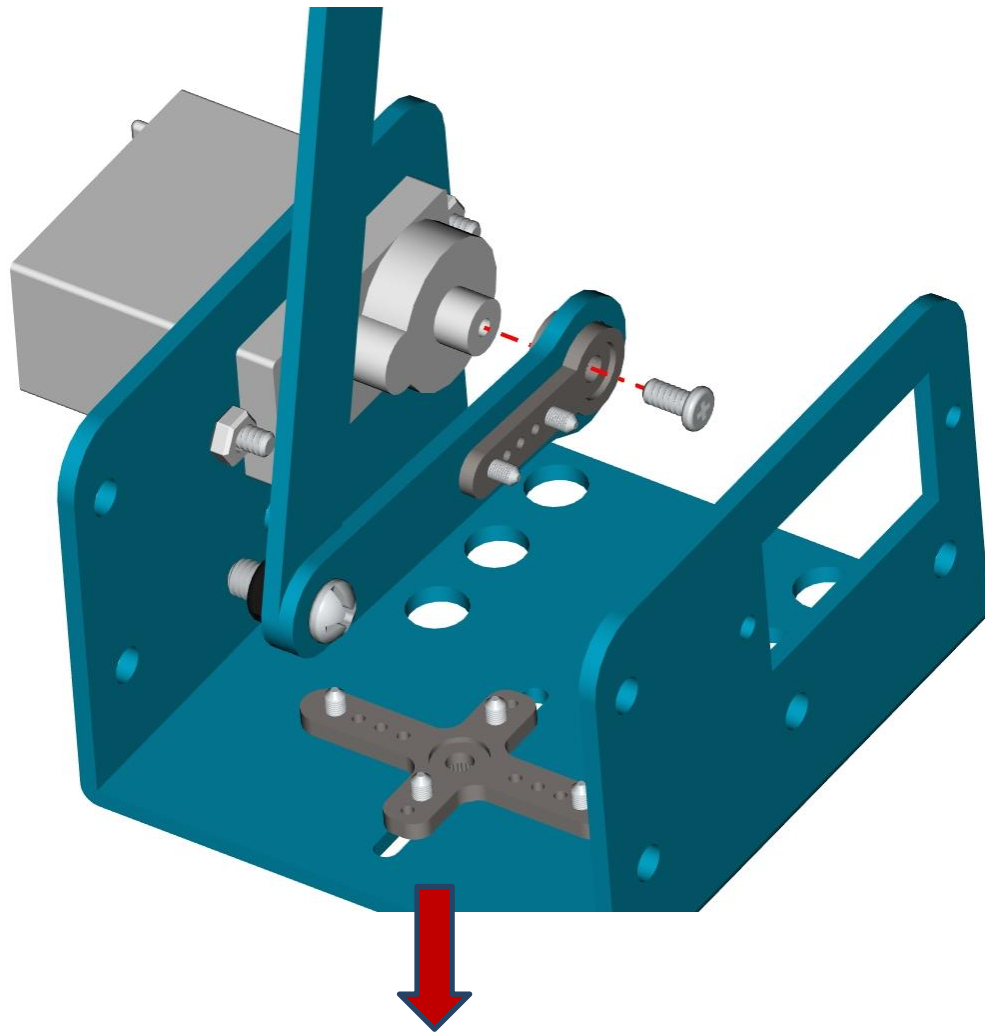
- Accessory7

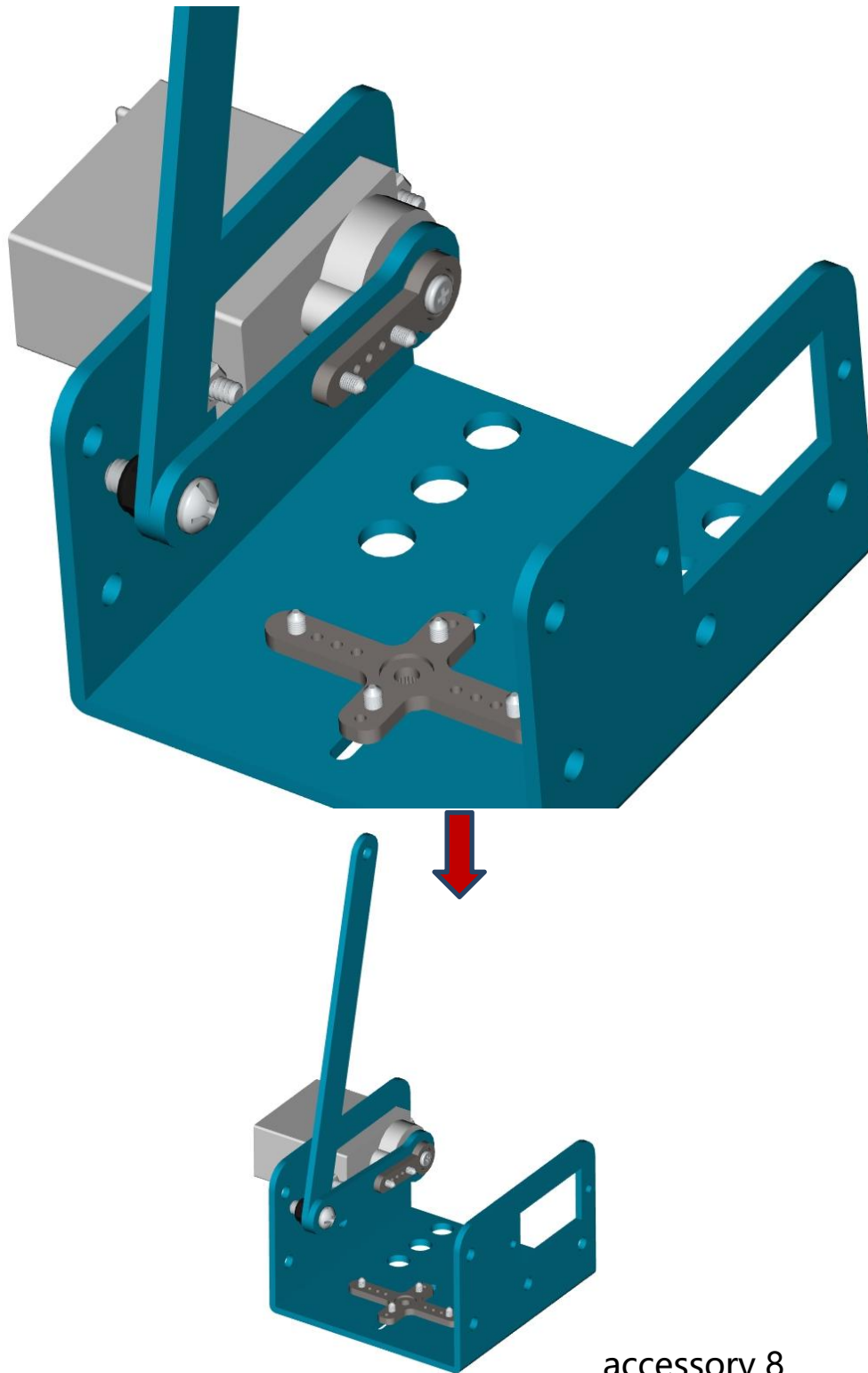
Use the above screws , nuts and a black horn to fix the two metal sheets with the accessory 7, then we get the accessory 8.







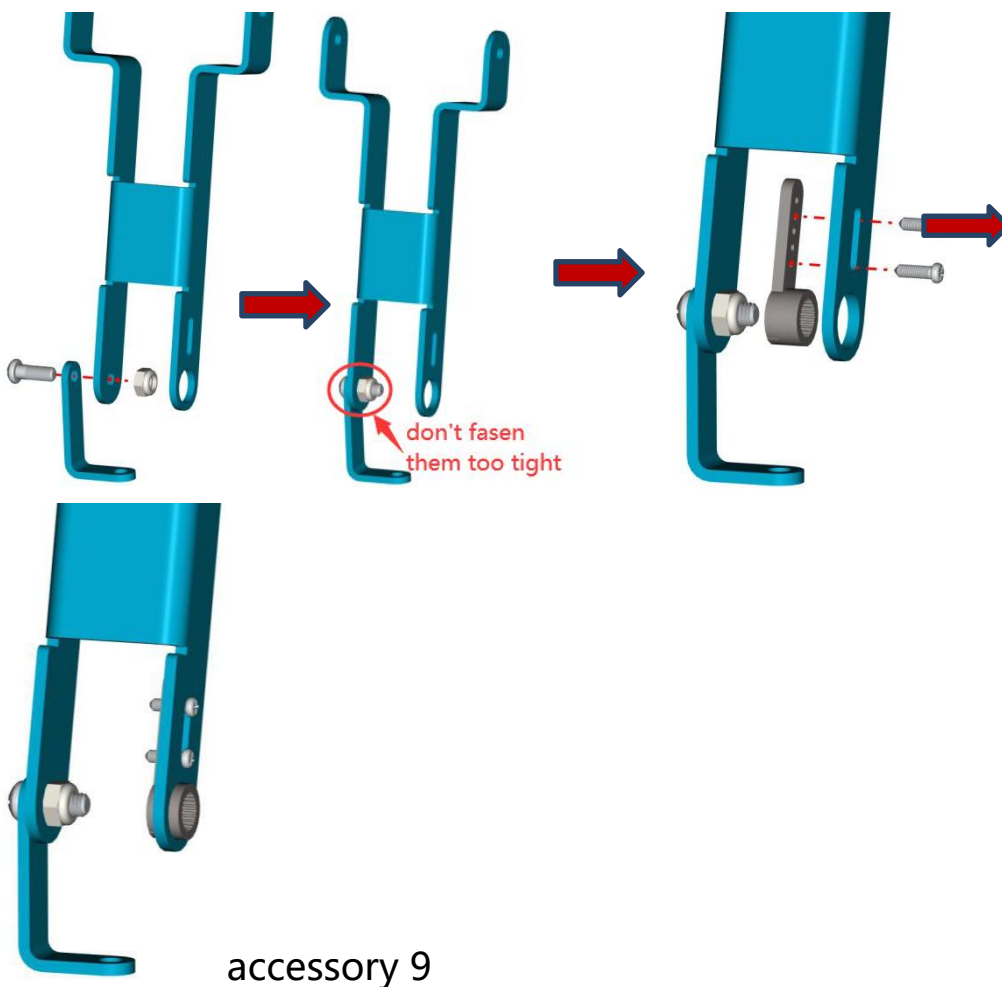






- 1pcs M3*8 MM round-head screws
- 1pcs M3 hex self-locking nuts
- 1pcs "L" type metal
- 1 pcs "Y" type metal
- 2pcs M1.2*4MM self-tapping screws
- A black horn (belong to servo)

Assemble the above parts together to get the accessory 9





- 1pcs M3*8MM round-head screw
- 1pcs M3 hex self-locking nut
- 2 pcs M2*8MM round-head screws
- 2 pcs M2 hex nuts
- 1pcs 180° metal servo
- 1 pcs M2*5 screw(belong to servo)
- Accessory 8
- Accessory 9

Need to set the angle of the servo to 130°.

The method of setting angles is as same as the step 5. You just need to replace `servopulse(servopin,90)` with `servopulse(servopin,130)` in the code.



The screenshot shows the Arduino IDE 1.8.9 window. The title bar reads "2 | Arduino 1.8.9". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar contains icons for opening, saving, and uploading. The sketch editor shows a file named "2" with the following code:

```
int servopin=A0; //Define digital interface A0 to connect servo steering gear signal line
int myangle;      //Define angle variables
int pulsewidth;  //Define pulse width variables
void setup()
{
  pinMode(servopin,OUTPUT); //Set steering gear interface as Output
}
void servopulse(int servopin,int myangle) //Define Function
{
  pulsewidth=(myangle*11)+500; //Converts the Angle to a pulse width value of 500 - 2480
  digitalWrite(servopin,HIGH); //The steering gear interface level is high
  delayMicroseconds(pulsewidth); //The microsecond number of the delay pulse width value
  digitalWrite(servopin,LOW); //Turn the steering gear interface level to low
  delay(20-pulsewidth/1000);
}
void loop()
{
  servopulse(servopin,130); //Set steering gear Angle
}
```

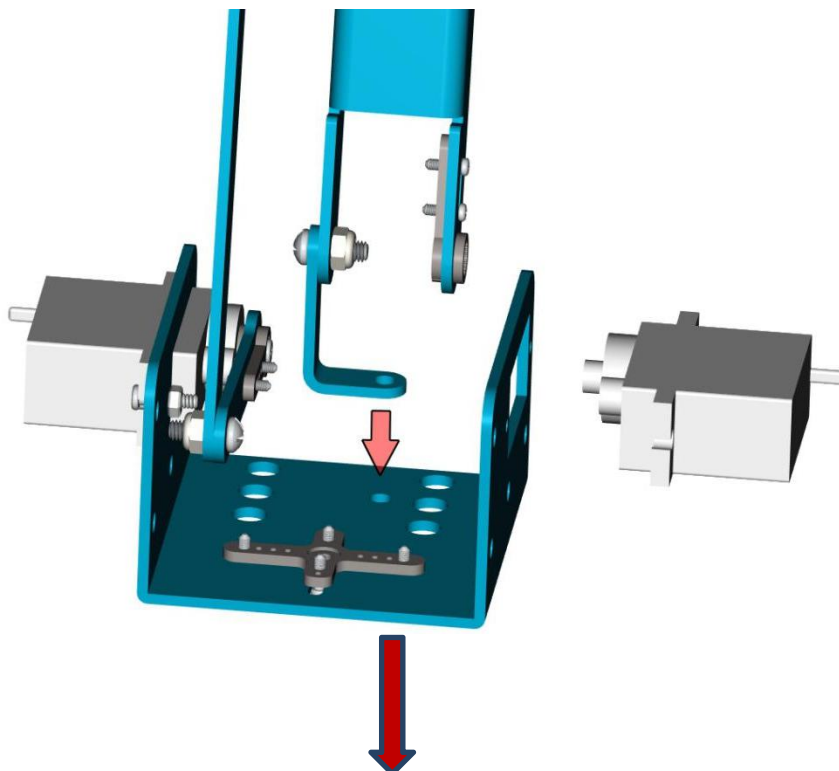
Below the editor, a status bar indicates "Done uploading." and provides memory usage details: "Sketch uses 926 bytes (2%) of program storage space. Maximum is 32256 bytes. Global variables use 11 bytes (0%) of dynamic memory, leaving 2037 bytes for local variables." The bottom status bar shows "18" and "Arduino/Genuino Uno on COM18".

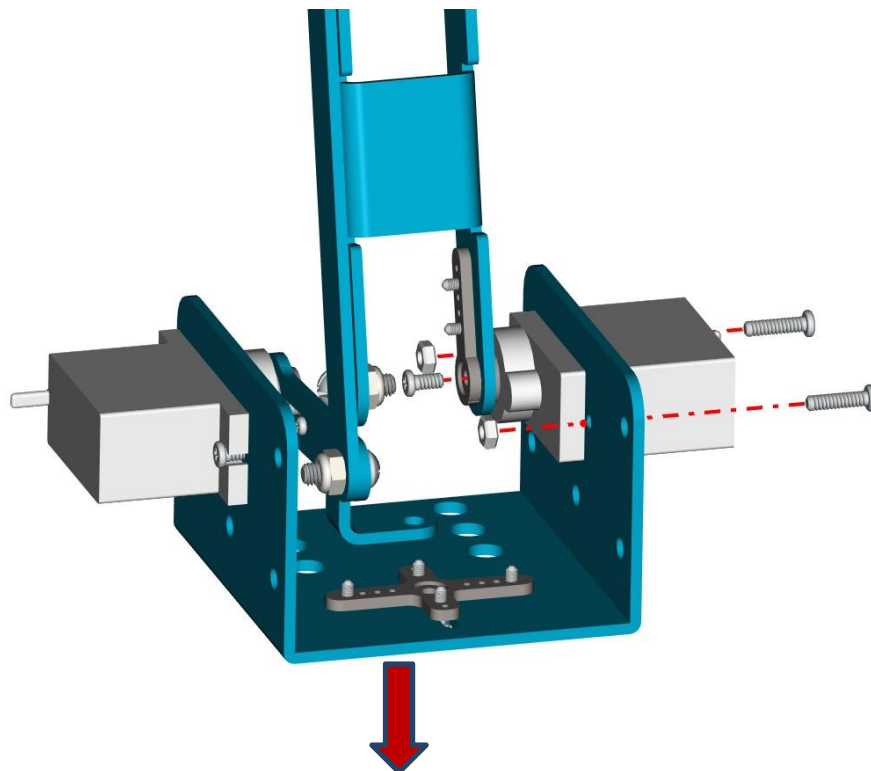
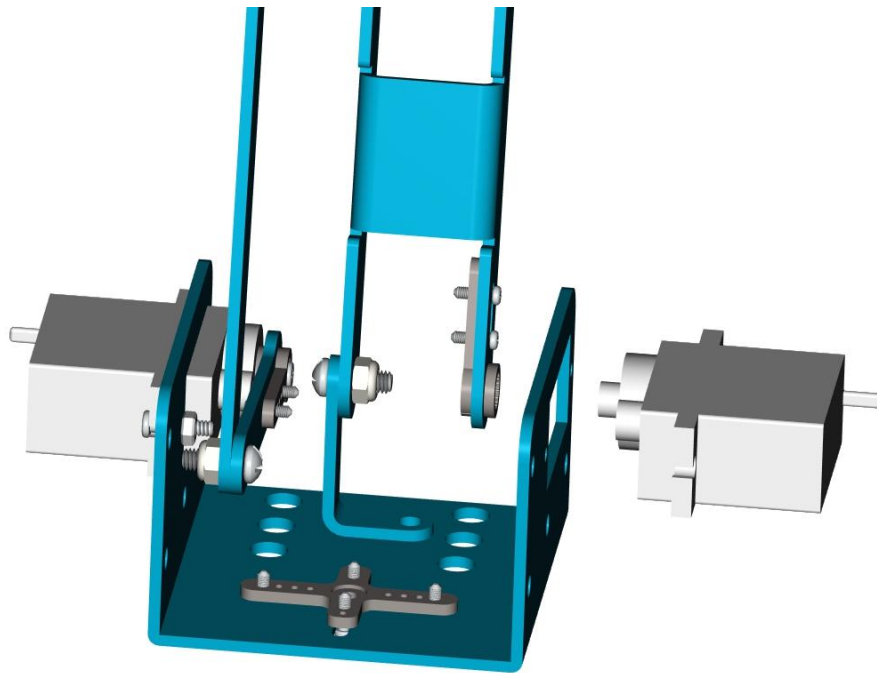


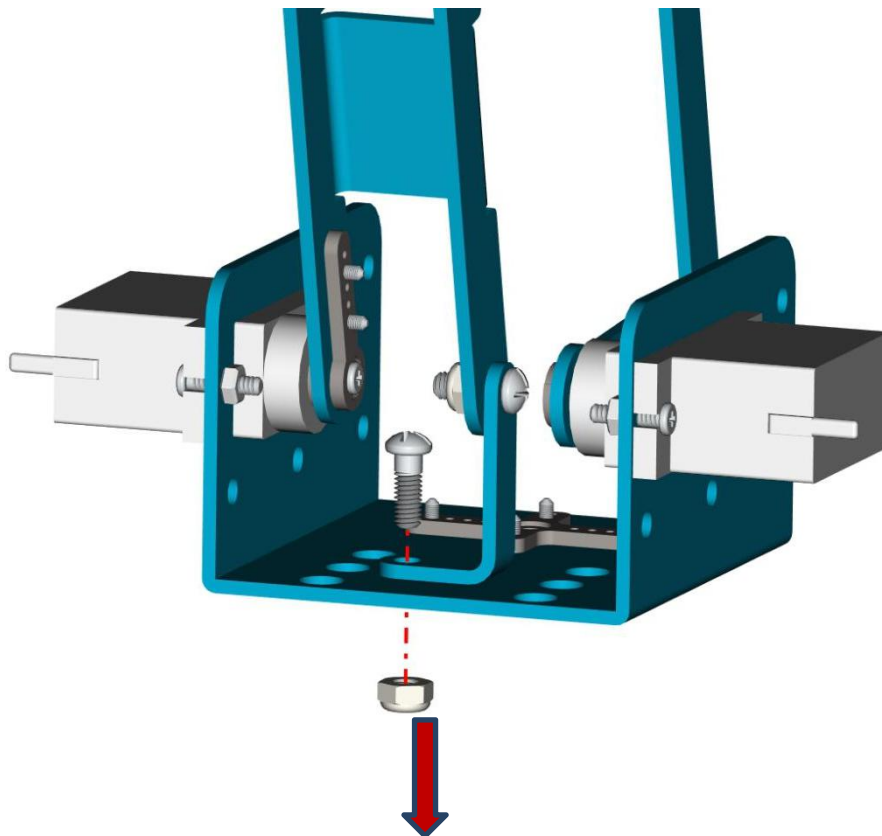
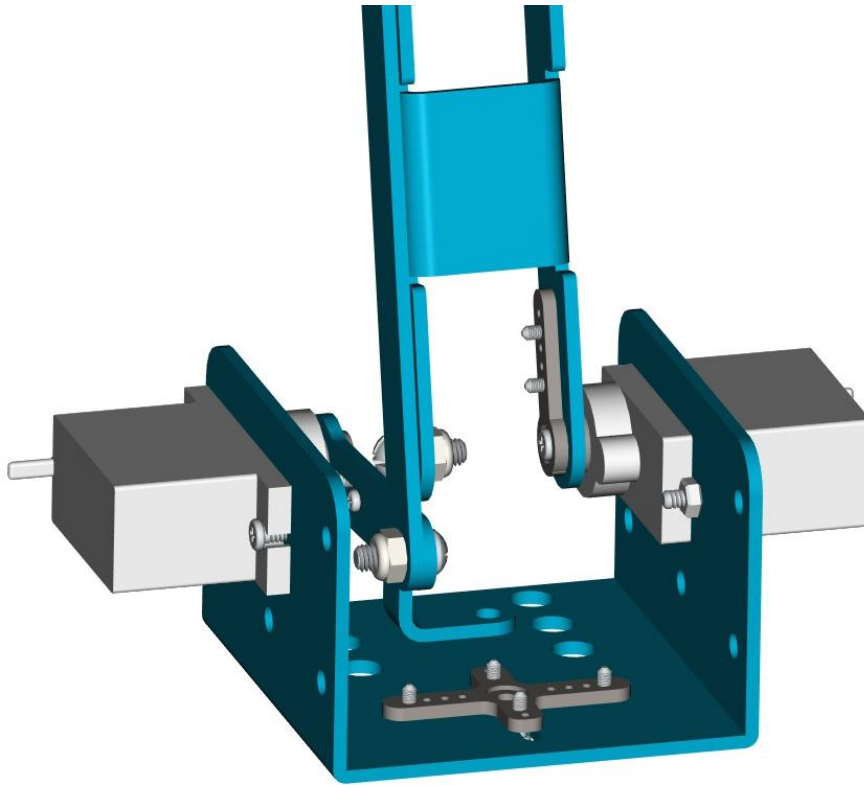
Remove the servo.

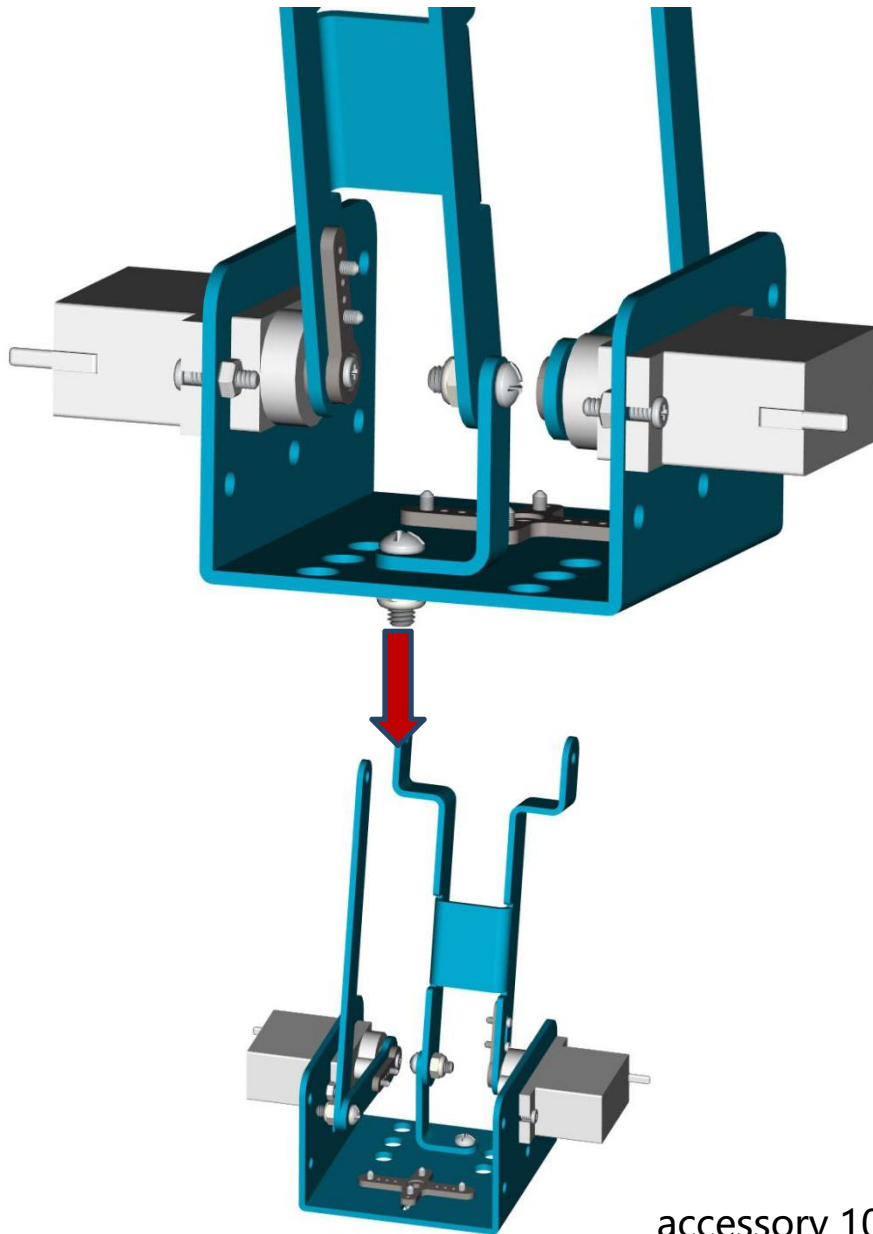
- 1pcs M3*8MMround-head screw
- 1pcs M3 hex self-locking nut
- 2 pcs M2*8MM round-head screws
- 2 pcs M2 hex nuts
- 1 pcs 180° servo
- 1 pcs M2*5 screw
- Accessory8
- Accessory9

Assemble the above parts together to get the accessory 10







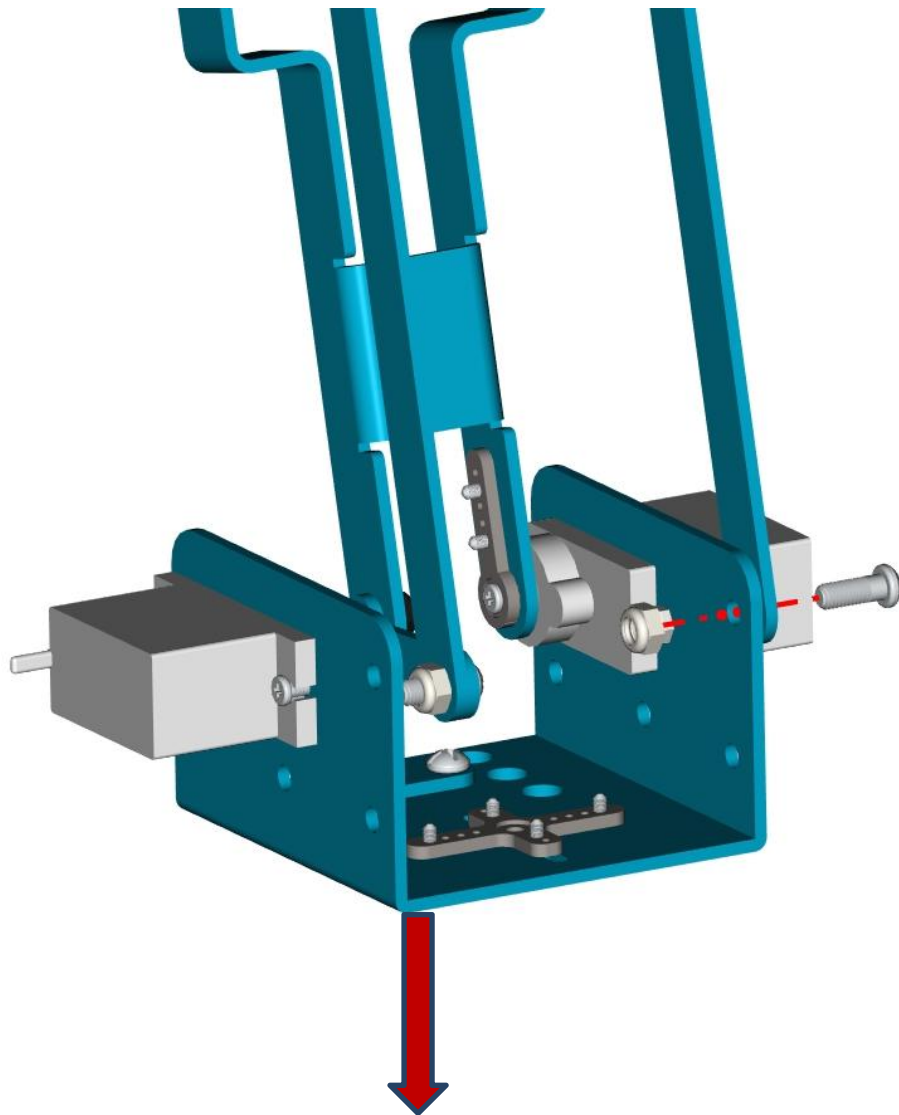


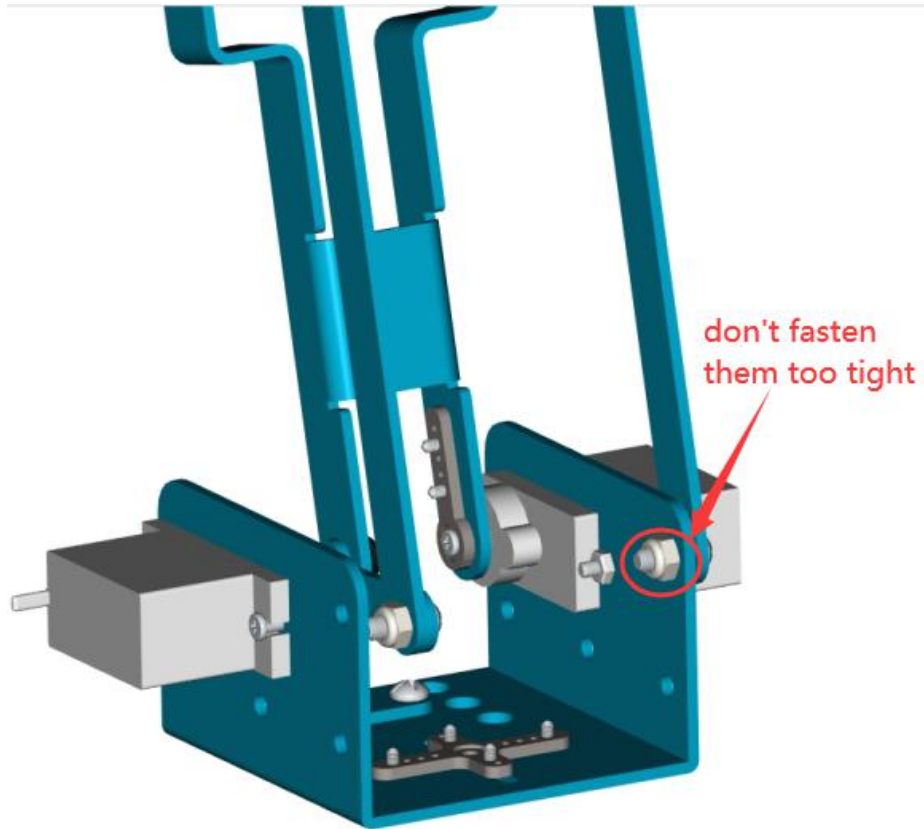
accessory 10

- 1pcs M3*8MM round-head screw
- 1pcs M3hex self-locking nut
- 1pcs 86mm metal sheet
- Accessory10



Mount the above parts together, then we get the accessory 11.

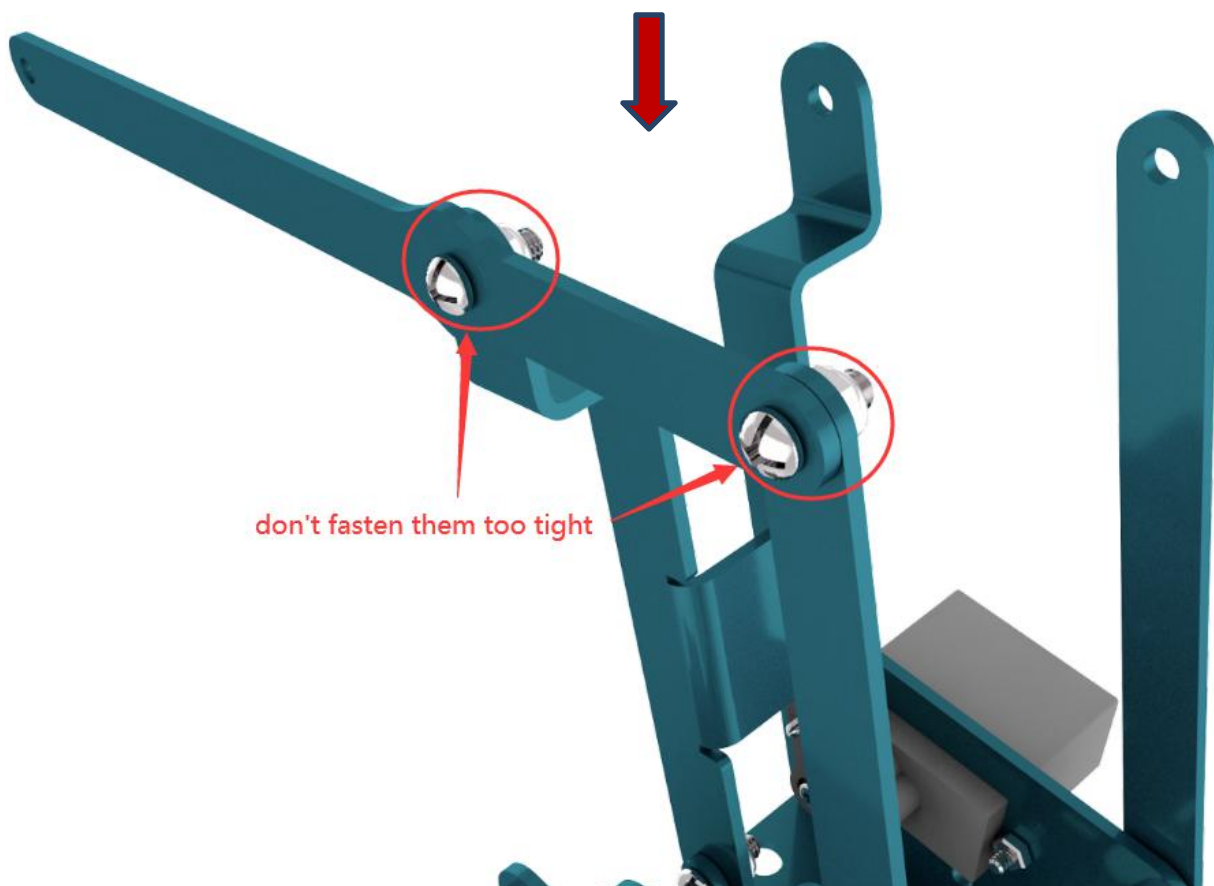




accessory11

- 2pcs M3*8MM round-head screws
- 2pcs M3 hex self-locking nuts
- 1pcs 125mm metal sheet
- Accessory11

Fix the above parts to get the accessory 12

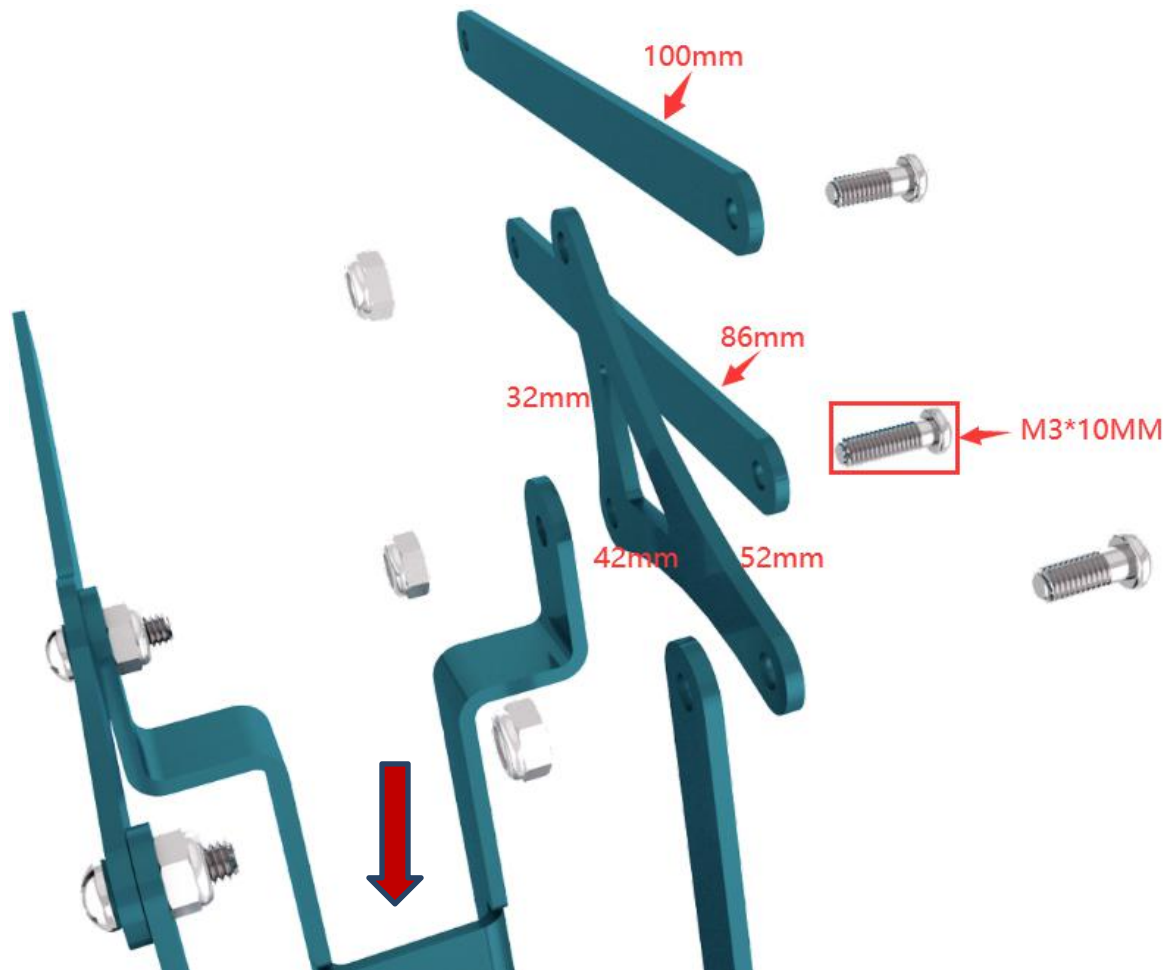


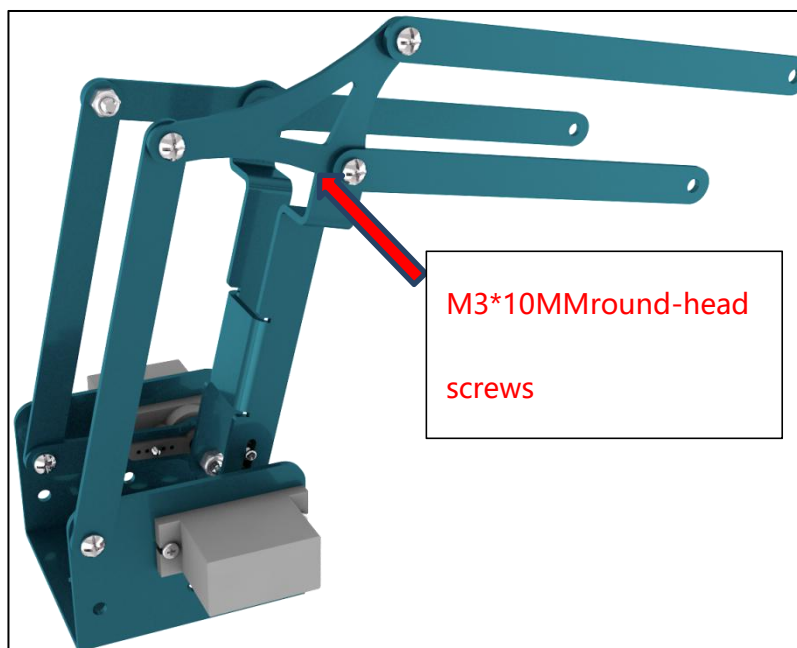
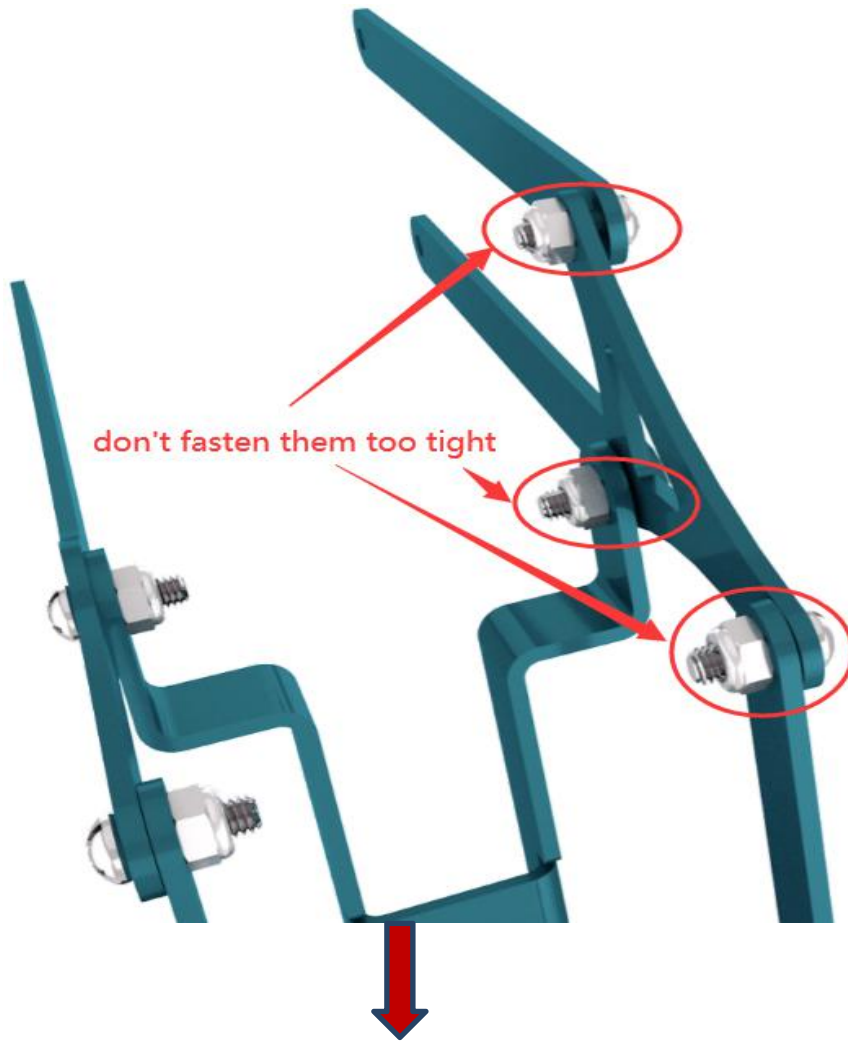
accessory12



- 2pcs M3*8MM round-head screws
- 1pcs M3*10MM round-head screw
- 3pcs M3 hex self-locking nuts
- 1pcs triangle metal sheet
- 1pcs 86mm metal sheet
- 1pcs 100mm metal sheet
- Accessory12

Install the above parts together to get the accessory 13



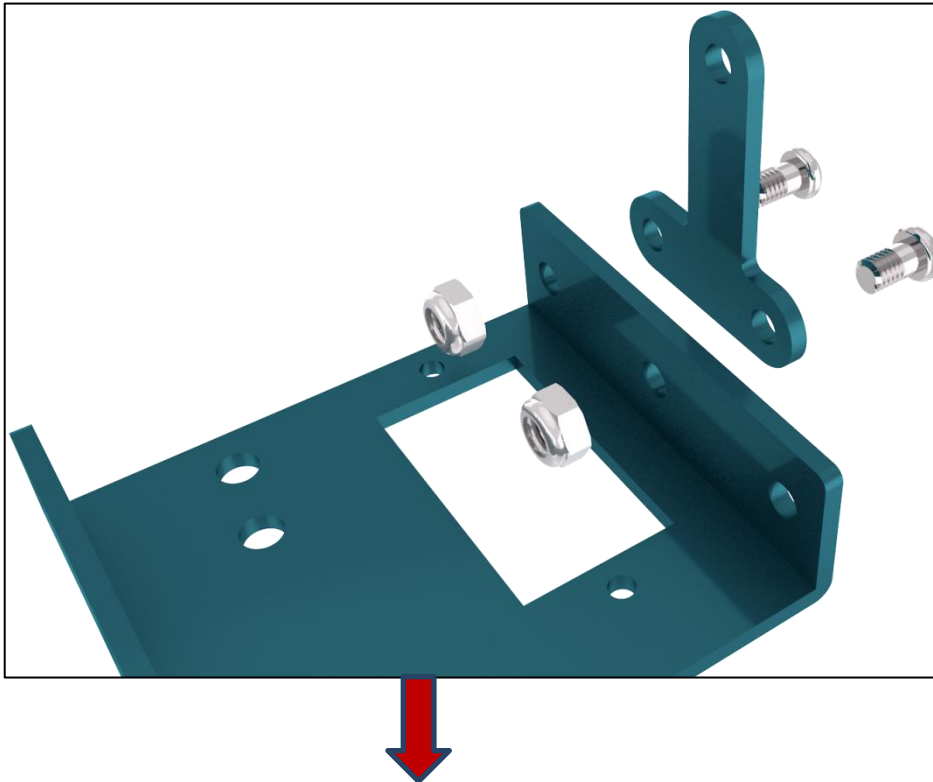


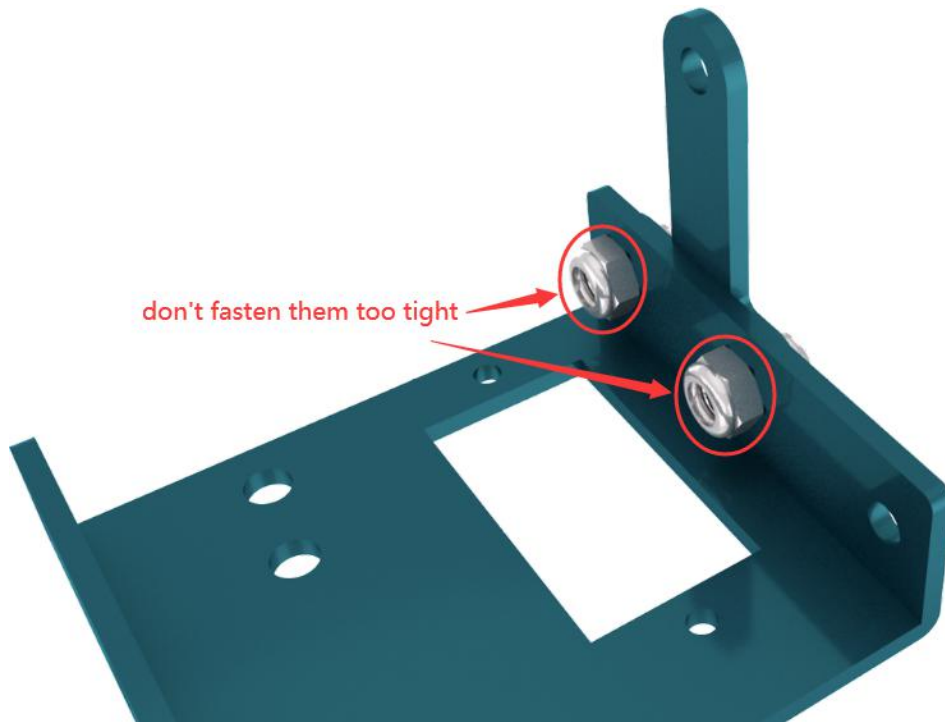
accessory 13



- 2pcs M3*6MM round-head screws
- 2pcs M3 hex self-locking nuts
- 1pcs "T" type metal sheet
- 1pcs corresponding metal sheet

Mount the above parts to get the accessory 14

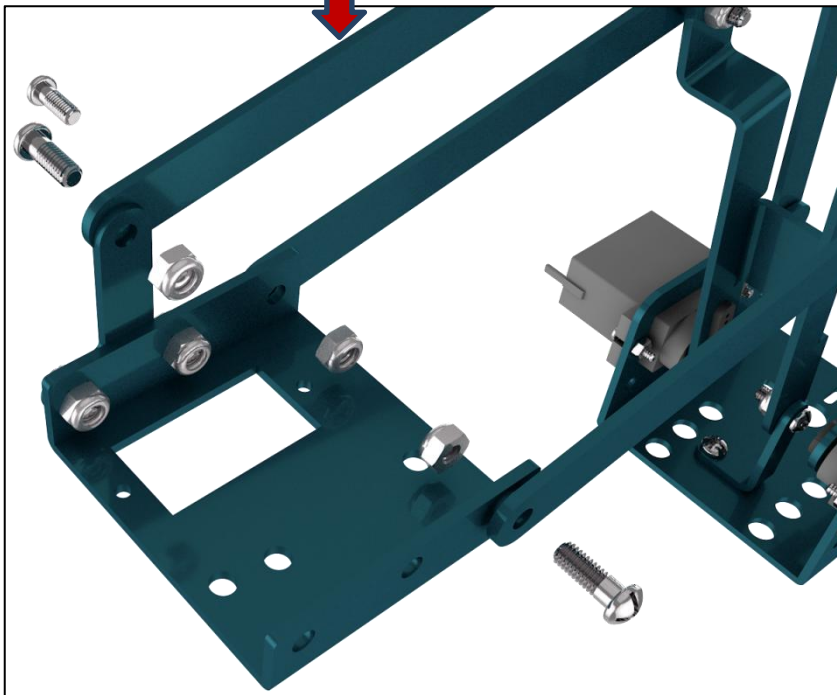
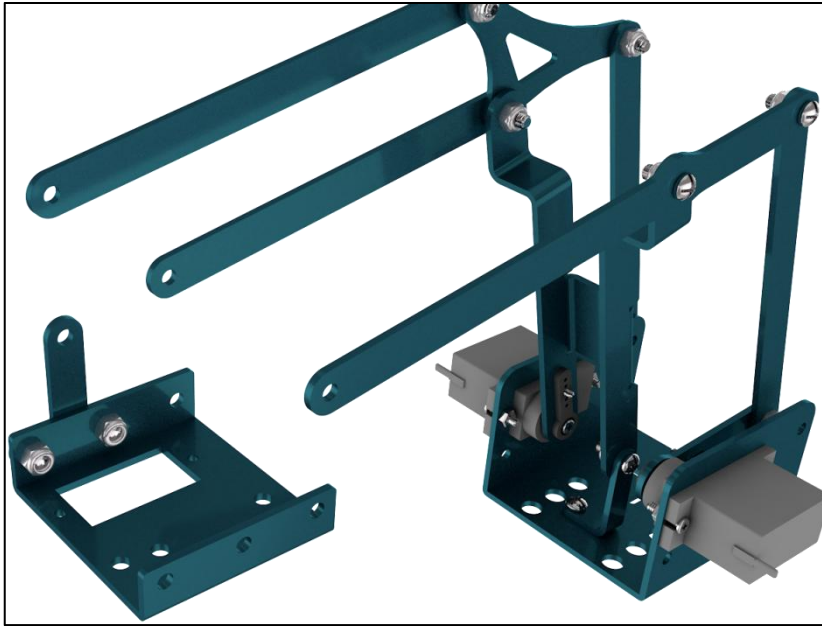


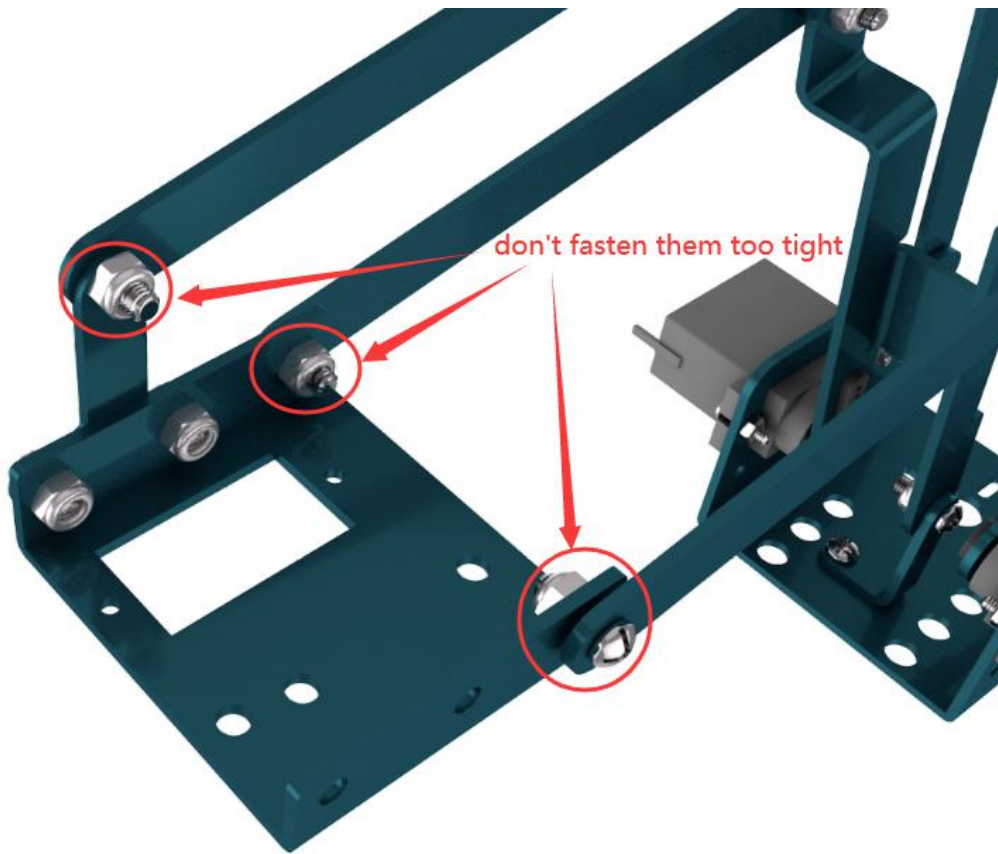


accessory 14

- 3pcs M3*8MM round-head screws
- 3pcs M3 hex self-locking nuts
- Accessory 13
- Accessory 14

Mount the above parts to get the accessory 15





accessory 15

- 2 pcs M2*12MM round-head screws
- 2 pcs M2 nuts
- 2 pcs insulation column posts
- 1pcs 180° servo
- Accessory 15



Need to set the angle of the servo to 0° firstly.

You only need to change `servopulse(servopin,90)` into `servopulse(servopin,0)`.

```
2 | Arduino 1.8.9
File Edit Sketch Tools Help
[Icons]
25
int servopin=A0; //Define digital interface A0 to connect servo steering gear signal line
int myangle;      //Define angle variables
int pulsewidth;  //Define pulse width variables
void setup()
{
  pinMode(servopin,OUTPUT); //Set steering gear interface as Output
}
void servopulse(int servopin,int myangle) //Define Function
{
  pulsewidth=(myangle*11)+500; //Converts the Angle to a pulse width value of 500 - 2480
  digitalWrite(servopin,HIGH); //The steering gear interface level is high
  delayMicroseconds(pulsewidth); //The microsecond number of the delay pulse width value
  digitalWrite(servopin,LOW); //Turn the steering gear interface level to low
  delay(20-pulsewidth/1000);
}
void loop()
{
  servopulse(servopin,0); //Set steering gear Angle
}
```

Done uploading.

Sketch uses 926 bytes (2%) of program storage space. Maximum is 32256 bytes.
Global variables use 11 bytes (0%) of dynamic memory, leaving 2037 bytes for local variables.

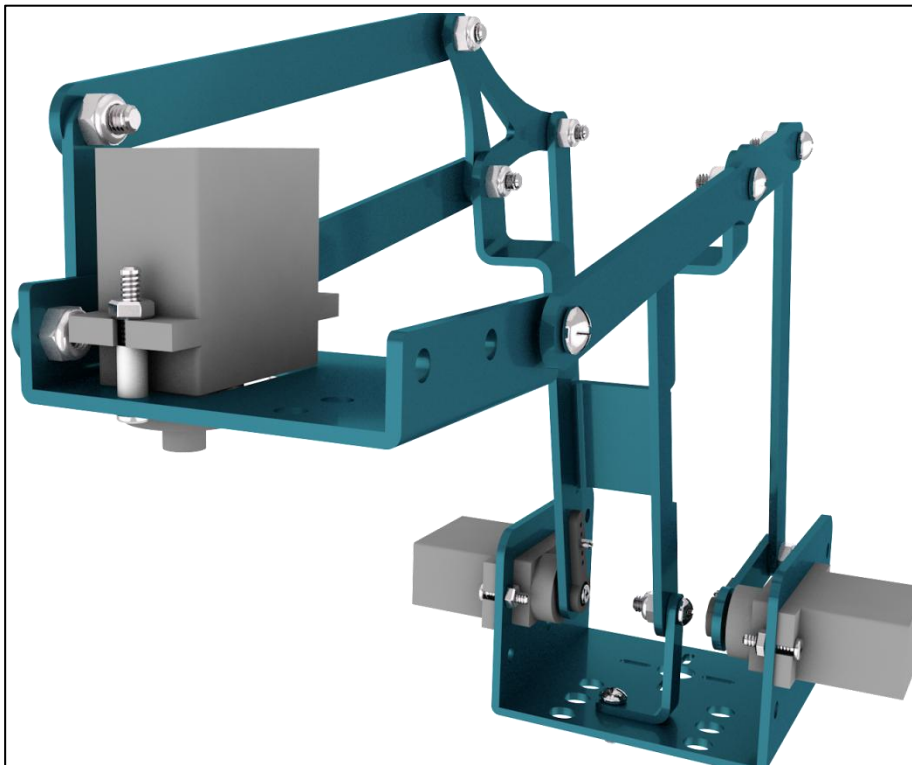
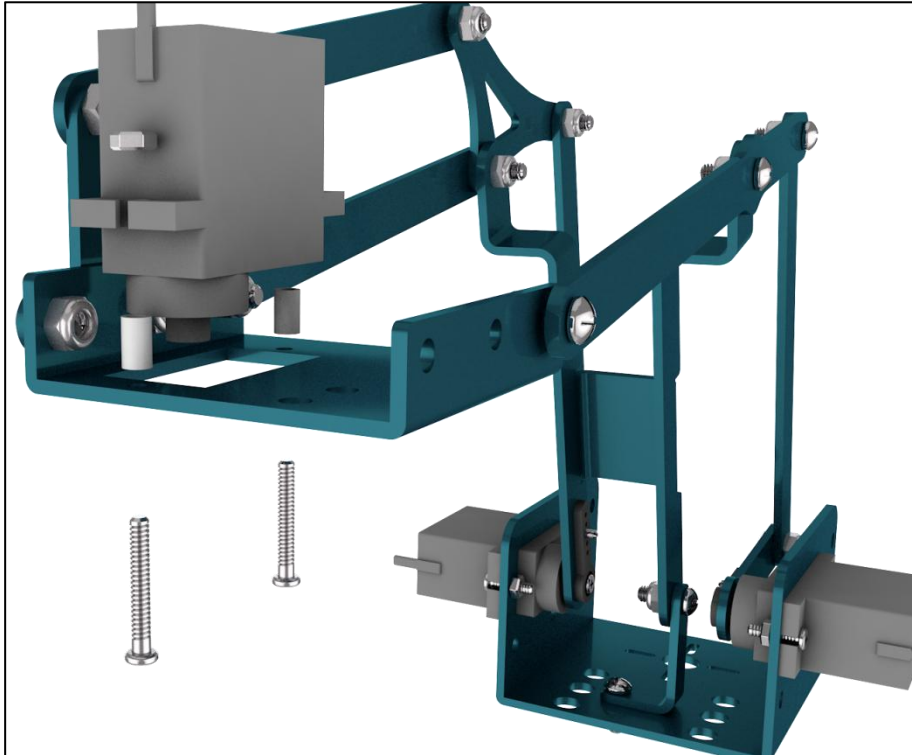
18 Arduino/Genuino Uno on COM18



Remove the servo.

- 2 pcs M2*12MM round-head screws
- 2 pcs M2 hex nuts
- 2 pcs insulation column posts
- 1 pcs 180° metal servo
- Accessory15

Install the above parts together, then we get the accessory 16

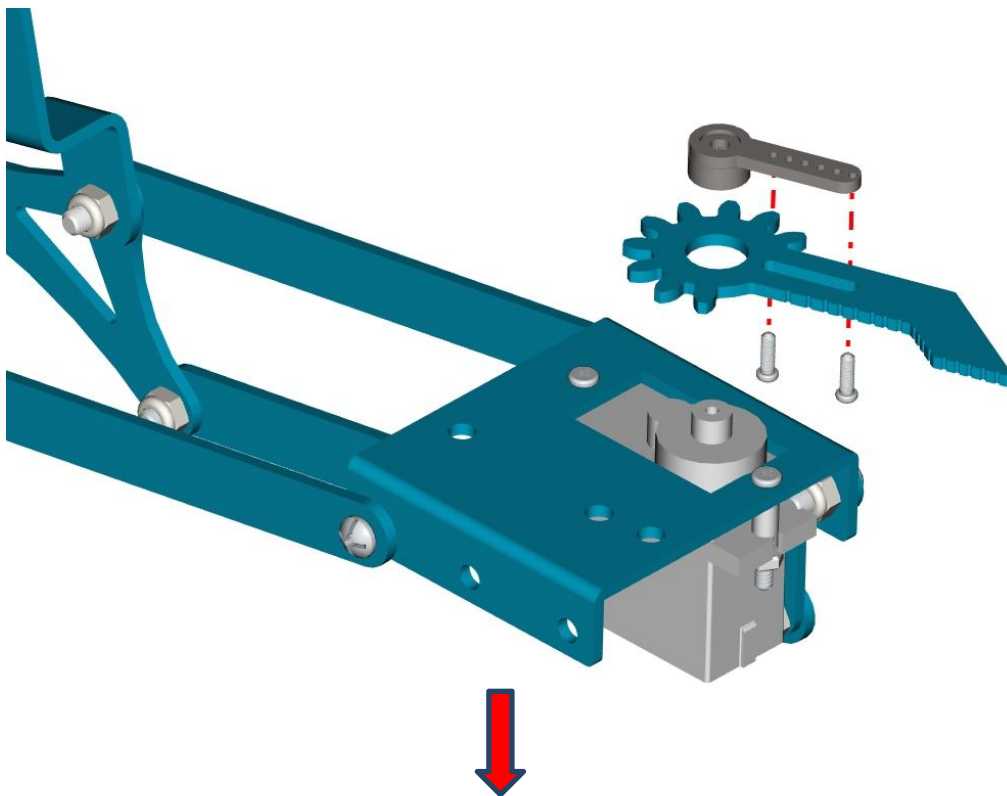


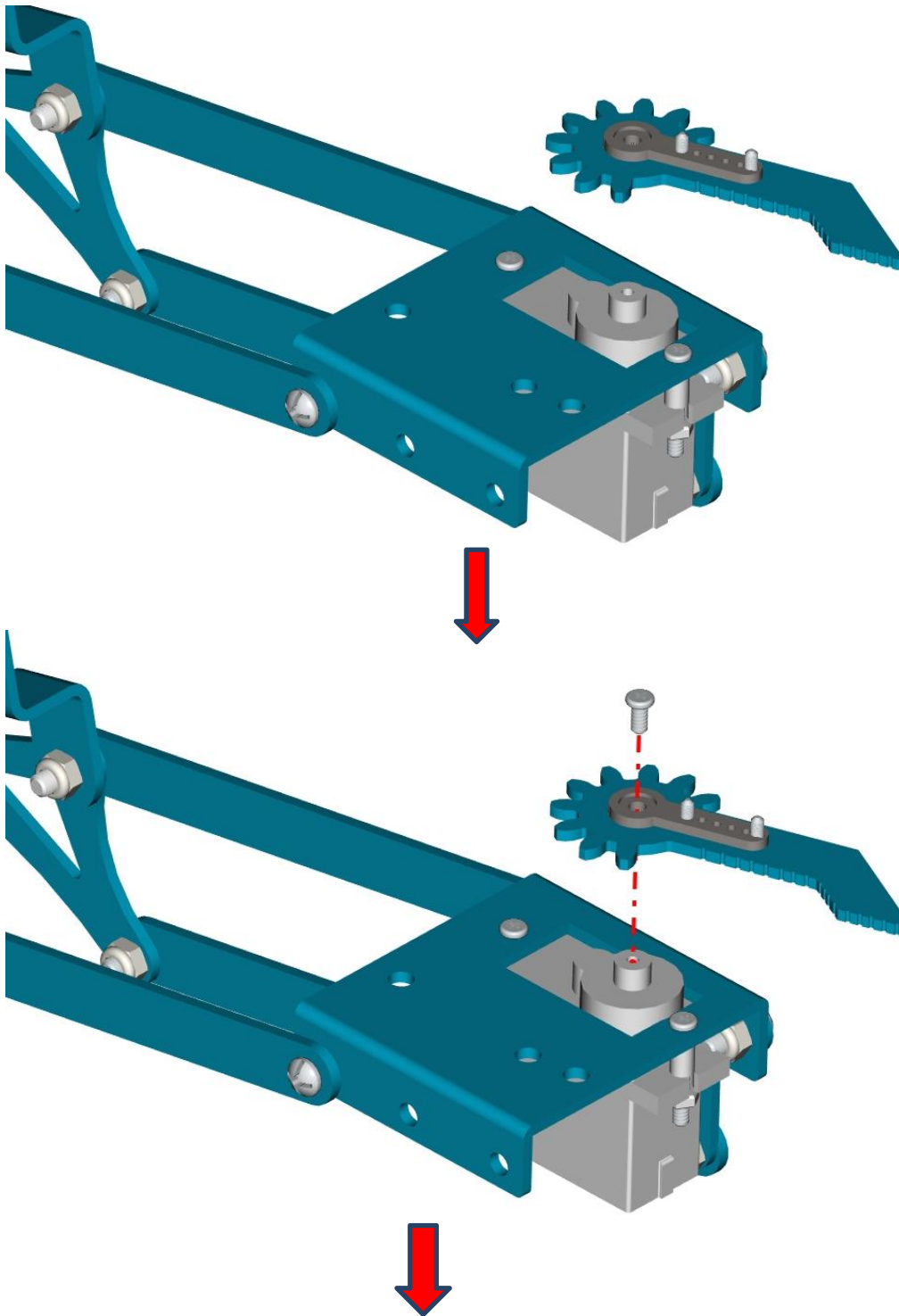
accessory 16

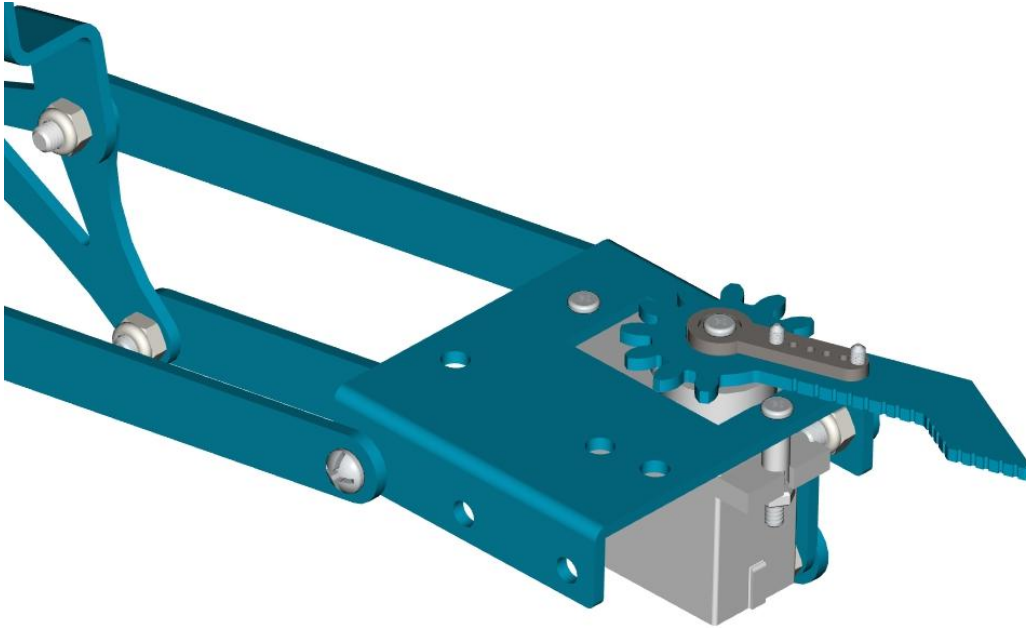


- 2pcs M1.2*4MMself-tapping screws
- A black horn(belong to servo)
- 1 pcs metal sheet with a gear
- 1 pcs M2*5 screw(belong to servo)
- Accessory 16

Install the above parts to get the accessory 17



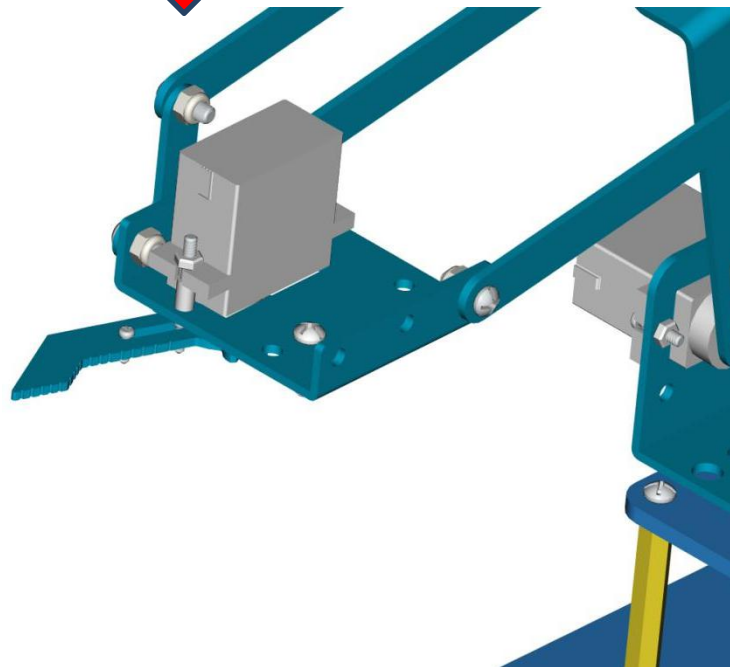
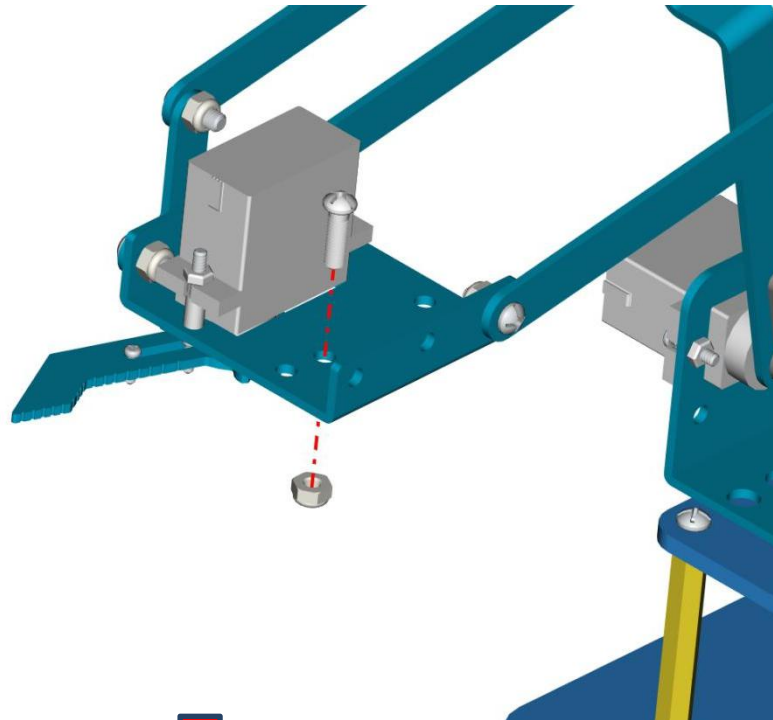


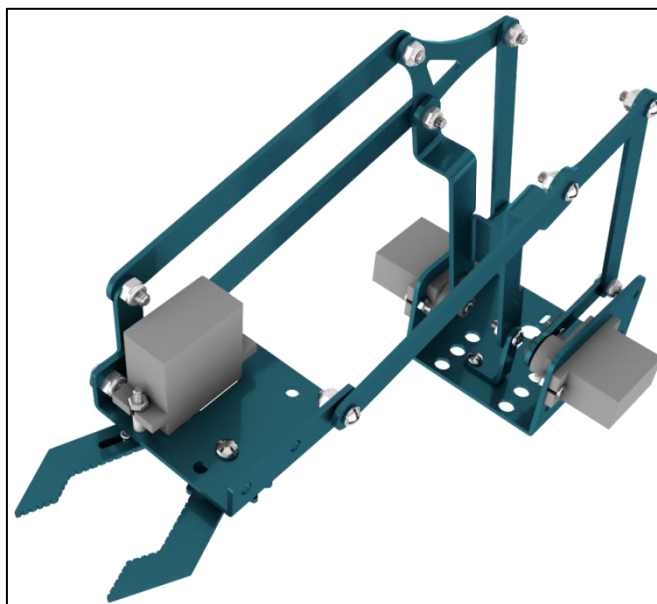
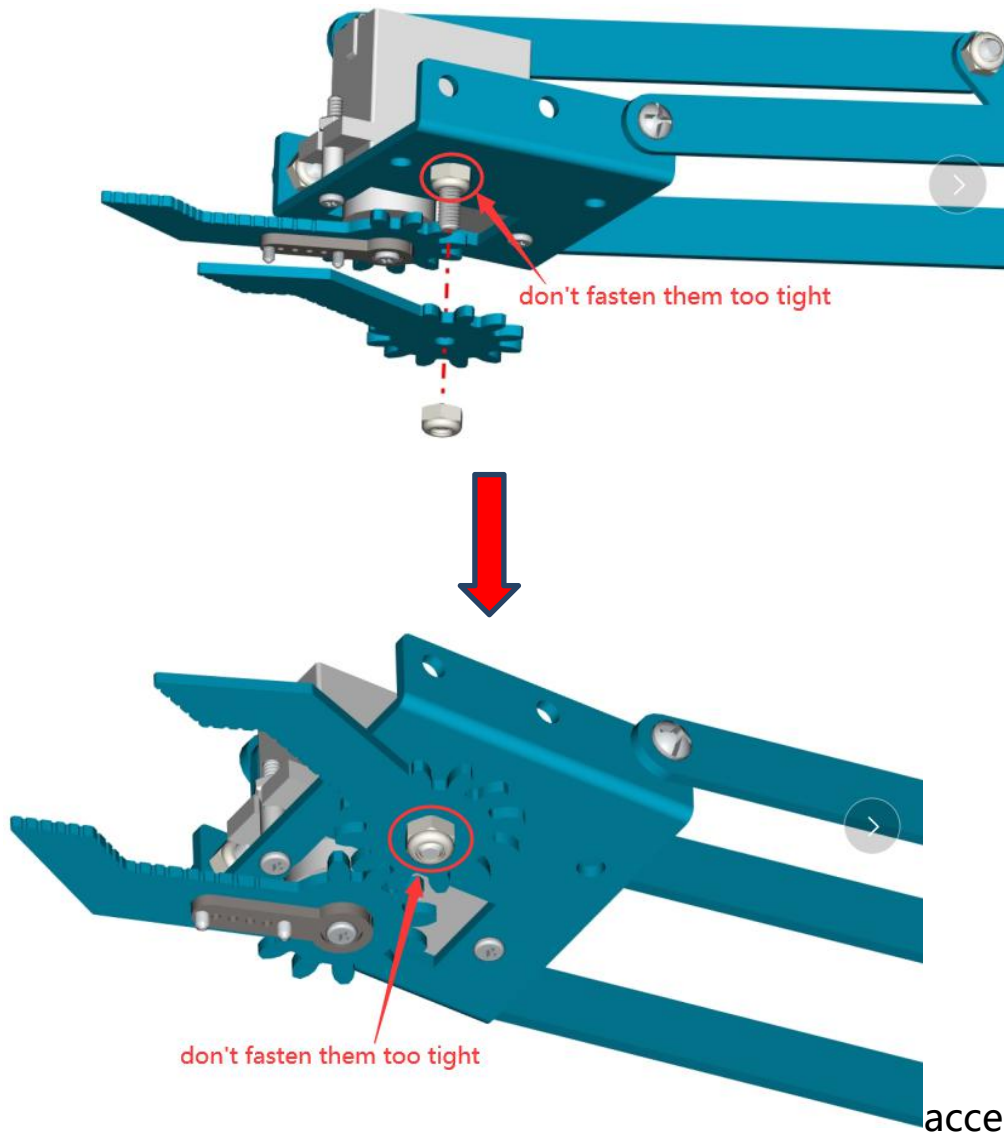


accessory 17

- 1pcs M3*10MM round-head screw
- 2 pcs M3 hex self-locking nuts
- 1 pcs metal sheet with a gear
- Accessory17

Install the above parts to get the accessory 18





accessory 19



The claw of the robot arm has been finished.