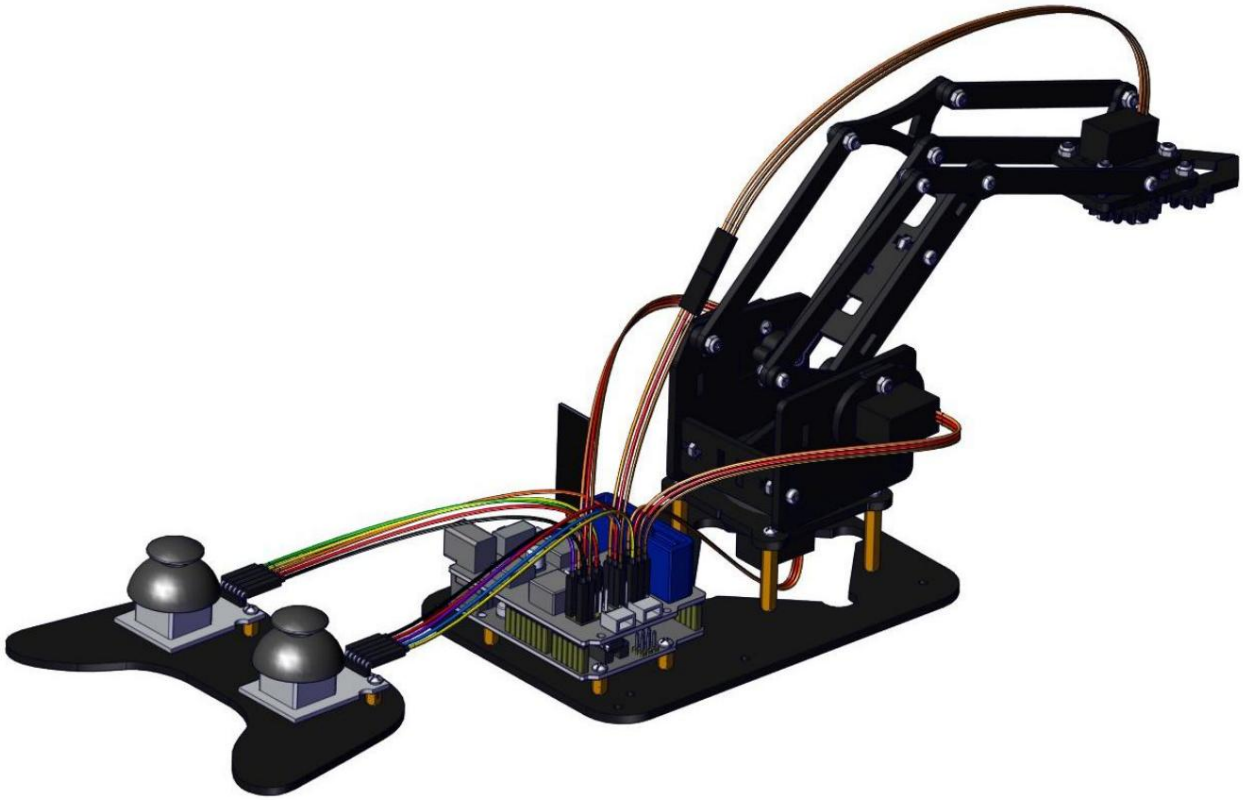
Machine Translated by Google

Keyestudio 4DOF Robot Mechanische Arm Kit voor Arduino

doe-het-zelf



Machine Translated by Google

Inhoud

1. Overzicht..... 1

2. Functies..... 2

3. Kitlijst 3

4. Montagehandleiding 8

(1) Installeer de basis van de robotarm..... 8

(2) Servo's op de basis monteren..... 13

5. Robotarmprojecten..... 65

(1) Arduino IDE installeren 72

(3) Stuurprogramma van V4.0-kaart installeren 73

(4) Arduino IDE-instelling..... 78

Project 2: 4DOF-rotatie en pincontrole..... 84

1) Gewrichtsrotatie en servohoekinstellingen..... 84

2) Pincontrole 84

Project 3: Bedien de robotarm met joysticks..... 85

3.1 Servobesturing..... 85

3.2 De joystickwaarde aflezen..... 94

3.3 Bediening met twee joysticks..... 99

3.4 Geheugenfunctie toevoegen..... 110

Project 4: BT-gestuurde robotarm..... ..... 158 ÿ

Principe van Bluetooth-besturing..... ..... 158 ÿ

Bluetooth-bedieningssleuteltest ....

160 ÿ Bluetooth bestuurt de robotarm..... 167

Project 5: PS2-gestuurde robotarm (uitbreiding)..... 176

5.1 PS2 Joypad-toetstest..... 176

Machine Translated by Google

5.2 PS2 Joypad-bediening..... 186

5.3 PS2-houdingsgeheugen regelen..... 201

5.4 PS2 bestuurt Posture Memory en Loop..... 221

6. Hulpbronnen..... 241

Machine Translated by Google

1. Overzicht

DIY is de activiteit van het zelf maken of repareren van dingen,

vooral bij jou thuis. Historisch gezien is het allemaal populair geweest

over de hele wereld sinds 1960, wat ons routinematige leven interessant maakt.

In combinatie met STEM-onderwijs kunnen doe-het-zelfproducten de fantasie en creativiteit van tieners sterk stimuleren.

Daarom rolt de R&D-groep van Keyestudio een verbazingwekkende 4DOF mechanische armkit uit, die bijdraagt aan het verbeteren van het hands-on vermogen, het logisch denken en het observatievermogen van kinderen.

Het is op zijn zachtst gezegd gemakkelijk te bouwen. In feite worden de vier servo's van deze robotarm bestuurd door een V4.0-besturingskaart en twee joystickmodules. Bovendien zijn de gedetailleerde tutorials voor u beschikbaar, zelfs als u een totale beginner bent.

Voor deze mechanische robotarm zijn er drie methodes

controle. De eerste is de controllerhandgreep die we leveren (joystickmodules), de tweede is de app; en de derde is een draadloze PS2-joystickmodule (niet inbegrepen in deze kit).

Ik geloof dat je het niet kunt laten om met deze kit aan de slag te gaan.

Laten we vervolgens aan de slag gaan.

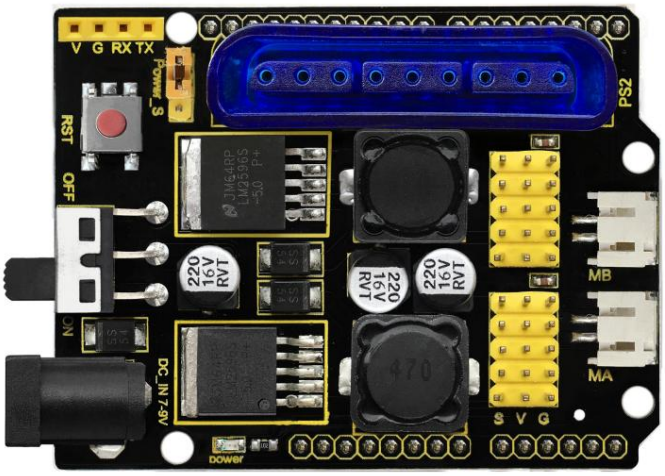
Machine Translated by Google

2. Functies

U kunt deze functies bekijken:

ÿ Gedetailleerde installatie-instructies

ÿ Gedetailleerde methoden voor foutopsporing, waarbij Arduino vanaf het begin wordt gestart. ÿ Drie besturingsmethoden: bedrade JoyStick-bediening; Telefoon Bluetooth-bediening; Draadloze PS2 JoyStick-bediening.



De parameters van keyestudio TB6612FNG motor/servo drive uitbreidingskaart zijn als volgt: ÿ VIN spanning: VIN = DC 7-15V

ÿ VIN-stroom: 5A

ÿ Tweeweg 5V-uitgang: 5V/3A ÿ TB6612FNG: VIN-ingang DC 7-15V; gemiddelde aandrijfstroom 1,2A; piekstroom 3.2A ÿ PS2-interface: compatibel met Sony PS2-ontvanger, kan rechtstreeks op de uitbreidingskaart worden aangesloten.

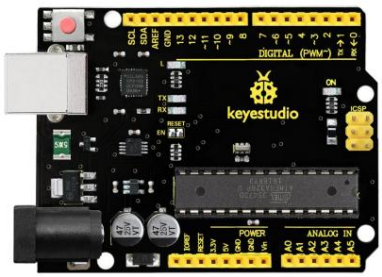
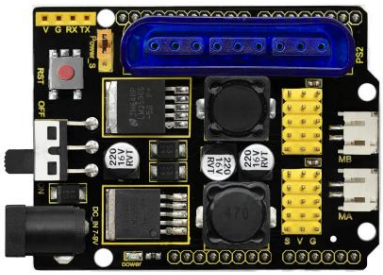
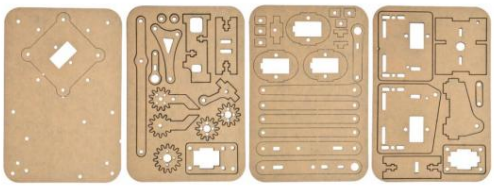
ÿ Afmetingen: 73*53.34mm

Machine Translated by Google

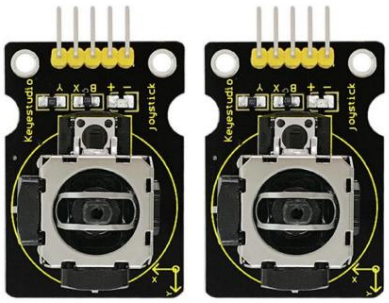
3. Kitlijst

Je ziet een vrij mooie verpakkingsdoos voor de armkit en in de verpakking vind je alle onderdelen en schroeven
hieronder opgesomd.

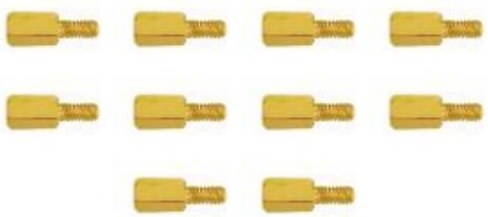
Opmerking: verwijder eerst de plastic folie van het bord wanneer u het installeert robotarm.

Nee.	Item	AANTAL	Afbeelding
1	Keyestudio V4.0 Besturingskaart	1	
2	Keyestudio TB6612FNG Motor/servo Stuurprogramma schild	1	
3	Acrylplaten 1		

Machine Translated by Google

4 Acryl handvat 1			
5	MeArm Zwart buikspieren Cilindrisch Houder	1	
6	180° Zwart Servo	4	
7	Keyestudio BT Modulair (HC-10)	1	
8	Keyestudio Joystick Modulair	2	
9	3D PS2 Joystick kap	2	

Machine Translated by Google

10	3*40 MM Schroevendraaier	1	
11	Verzinkt Moersleutel	1	
12	M3*6MM Ronde Kop Schroeven	12	
13	M3*10MM Ronde Kop Schroeven	22	
14	M3*14MM Vlak Kop Schroeven	2	
15	M3*12MM Ronde Kop Schroeven	12	
16	M3*24+6MM Koperen pijler	4	
17	M3*6mm+6m M Koperen pijler	10	

Machine Translated by Google

18	M3 roestvrij Stalen zeskant Noten	22	
19	M3 Zeshoek Noten	24	
20	M1.2x5MM Philips Zelftappend Schroeven	8	
21	M2x5MM Philips Zelftappend Schroeven	10	
22	M3 304 Roestvrij staal Platte ring	10	
23	M2x8MM Philips Zelftappend Schroeven	2	
24	M3*16MM Vlak Kop Schroeven	2	
25	Man vrouw 10CM Doorverbindingsdraad	4	

Machine Translated by Google

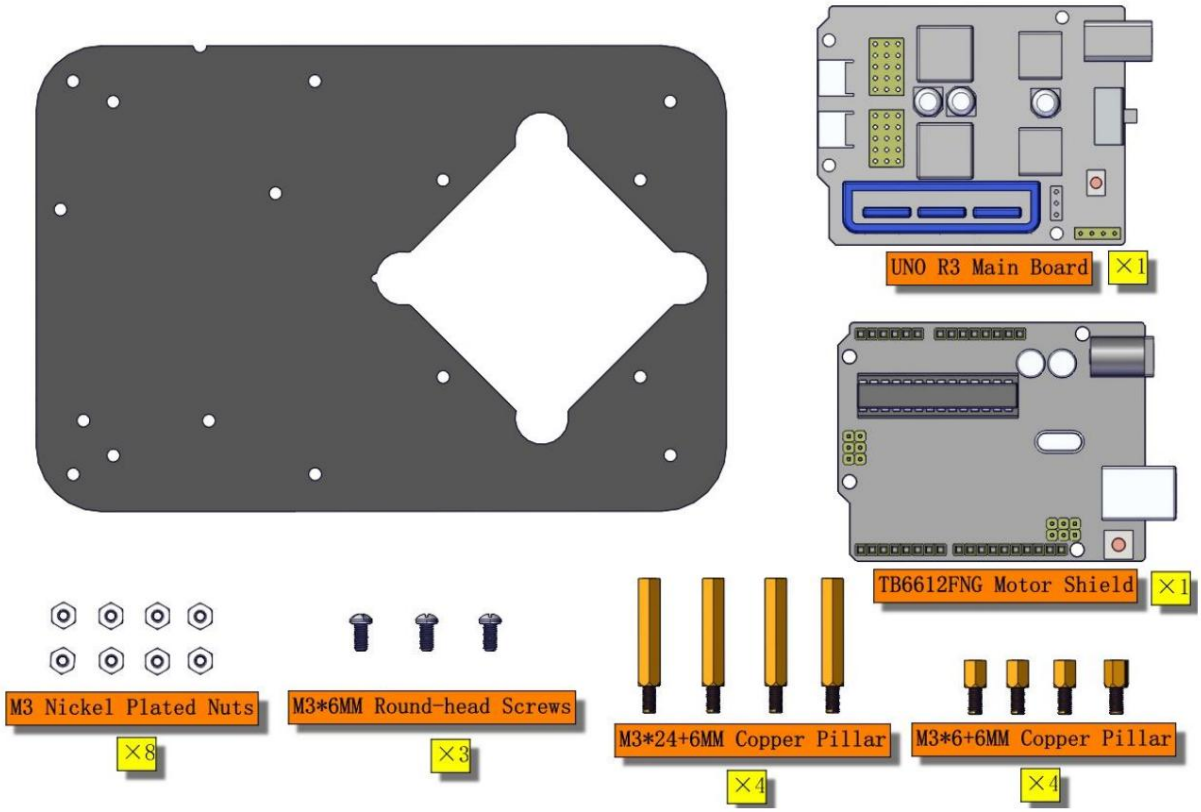
26	Vrouwelijk- Vrouw 50CM Doorverbindingsdraad	10	
27	Zwart 3*100 MM Kabelbinders	7	
28	18650 2-slots Accu Houder	1	

Machine Translated by Google

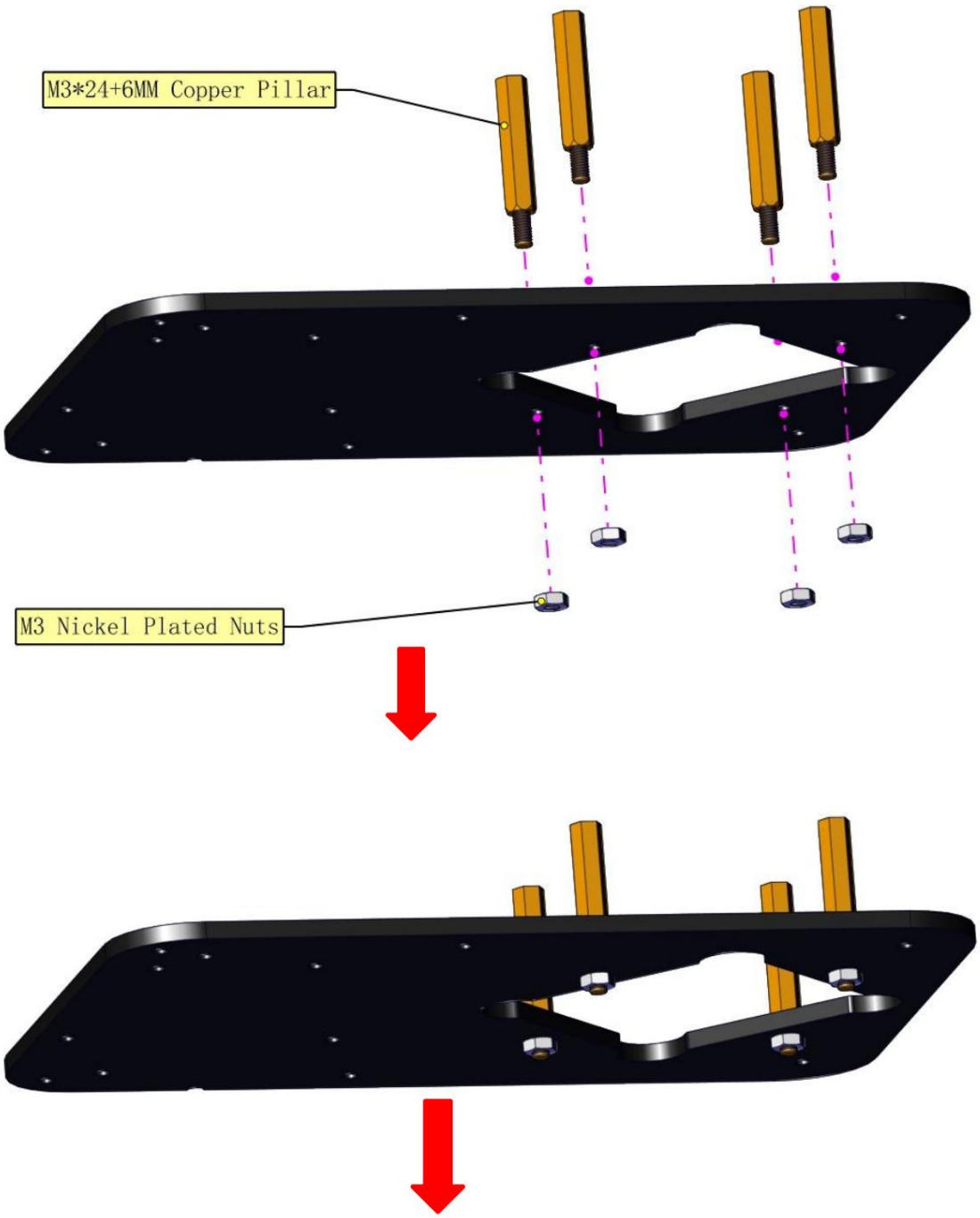
4. Montagehandleiding

(1) Installeer de basis van de robotarm

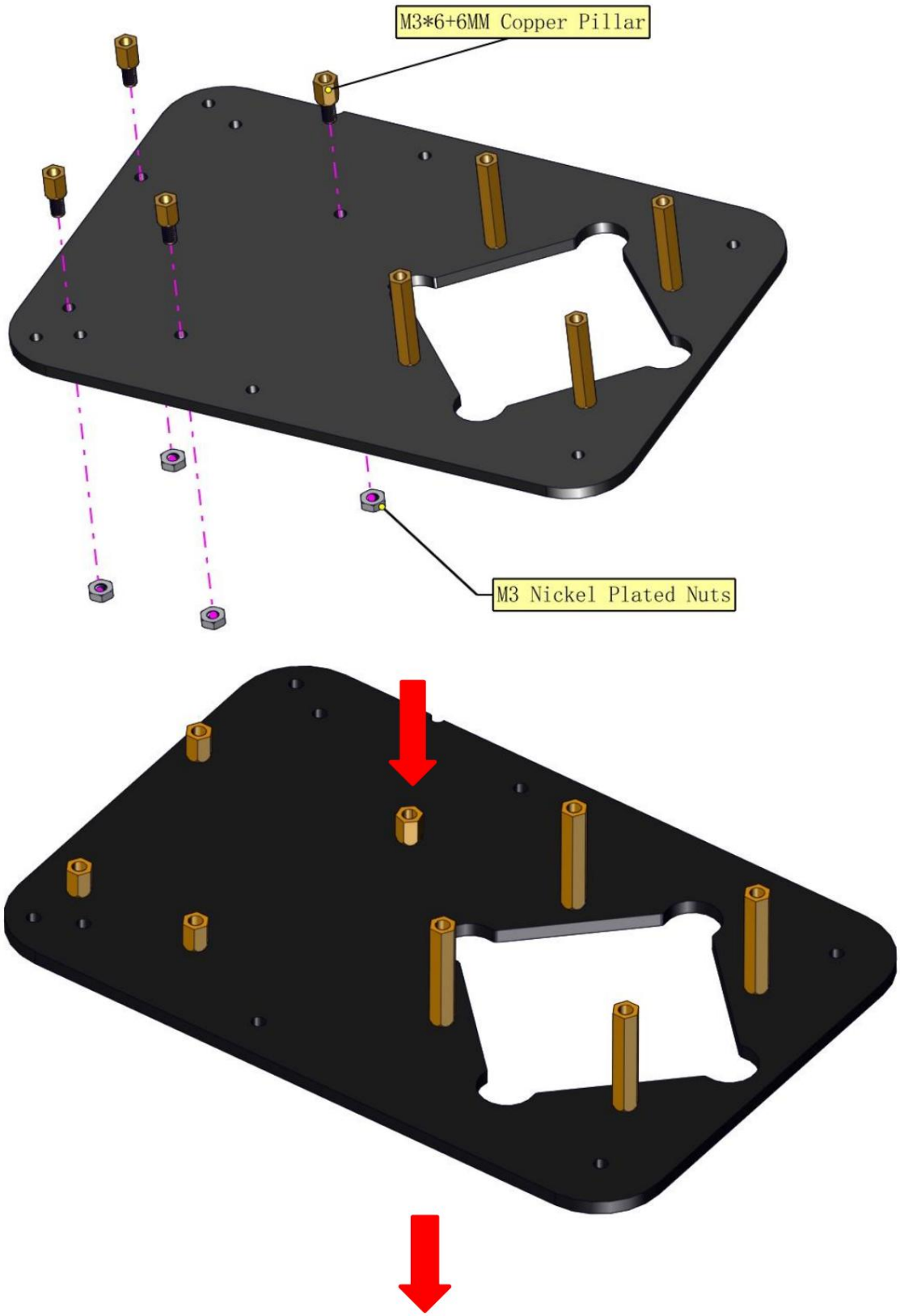
Benodigde componenten:



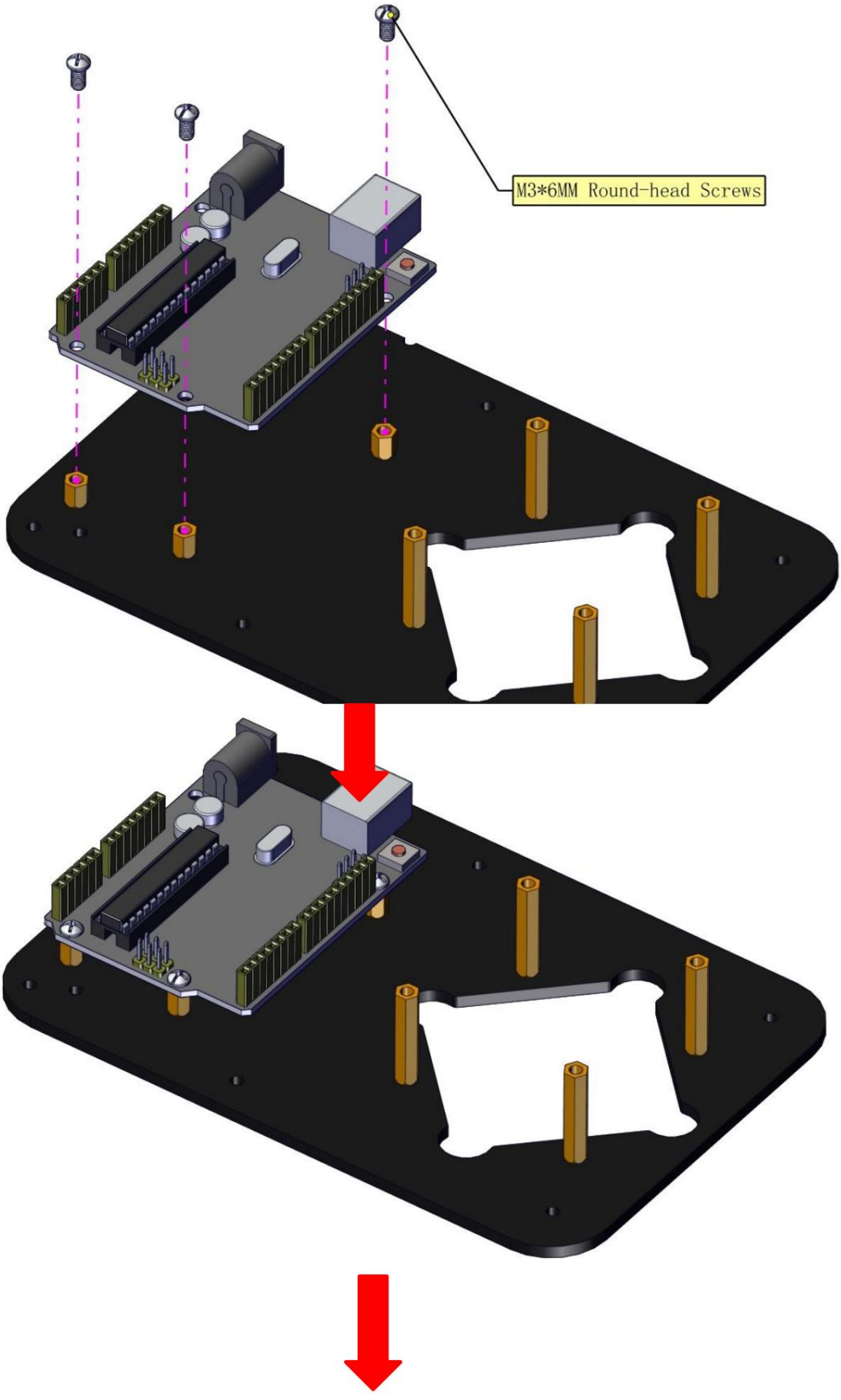
Machine Translated by Google



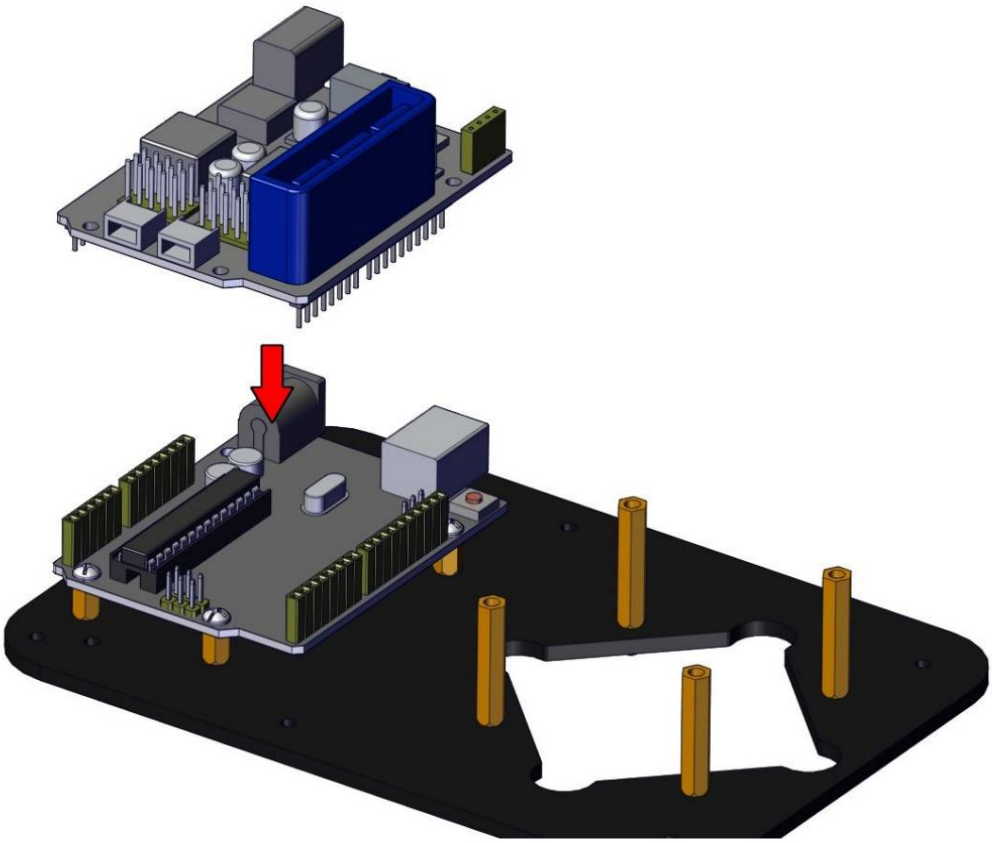
Machine Translated by Google



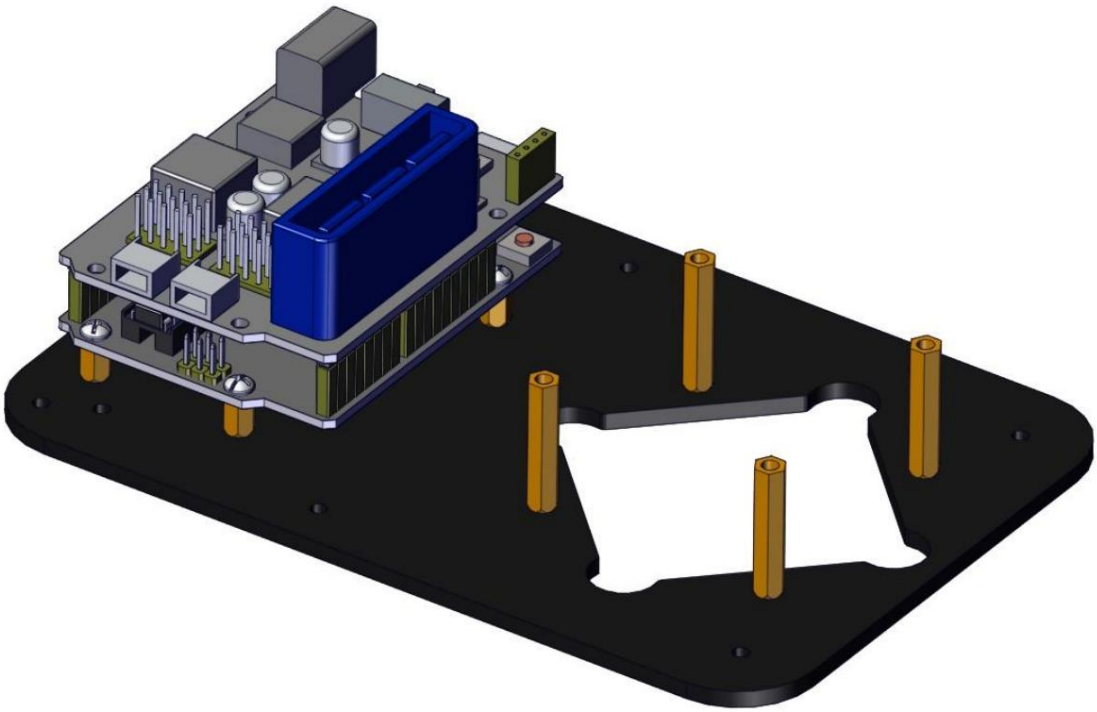
Machine Translated by Google



Machine Translated by Google



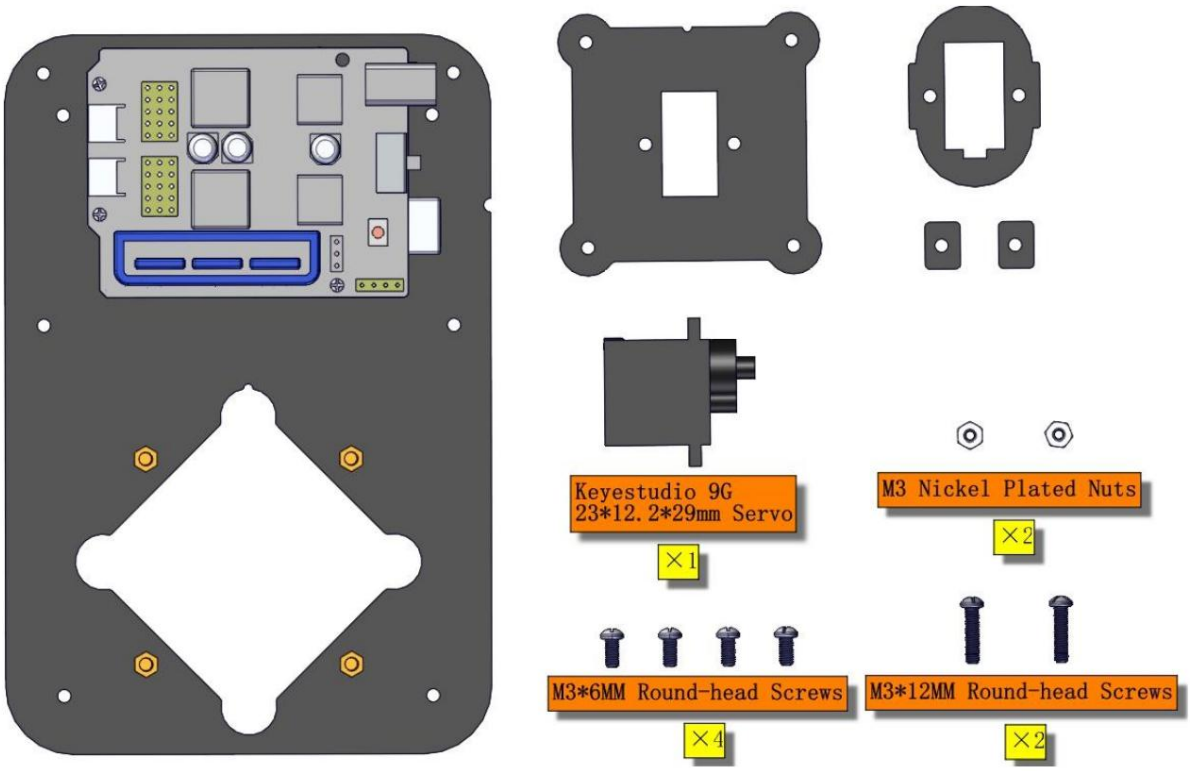
De basis is succesvol geïnstalleerd.



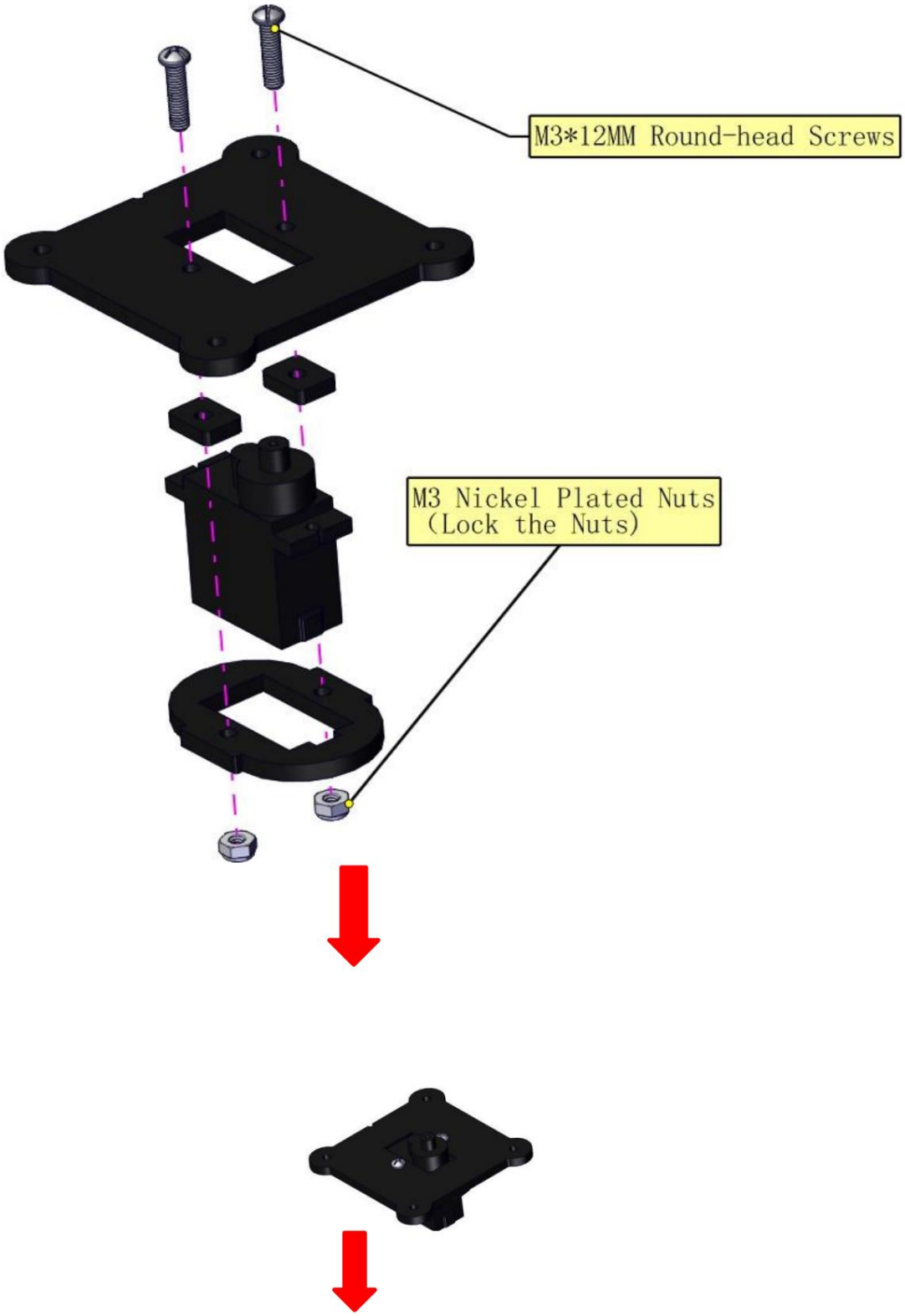
Machine Translated by Google

(2) Monteer servo's op de basis

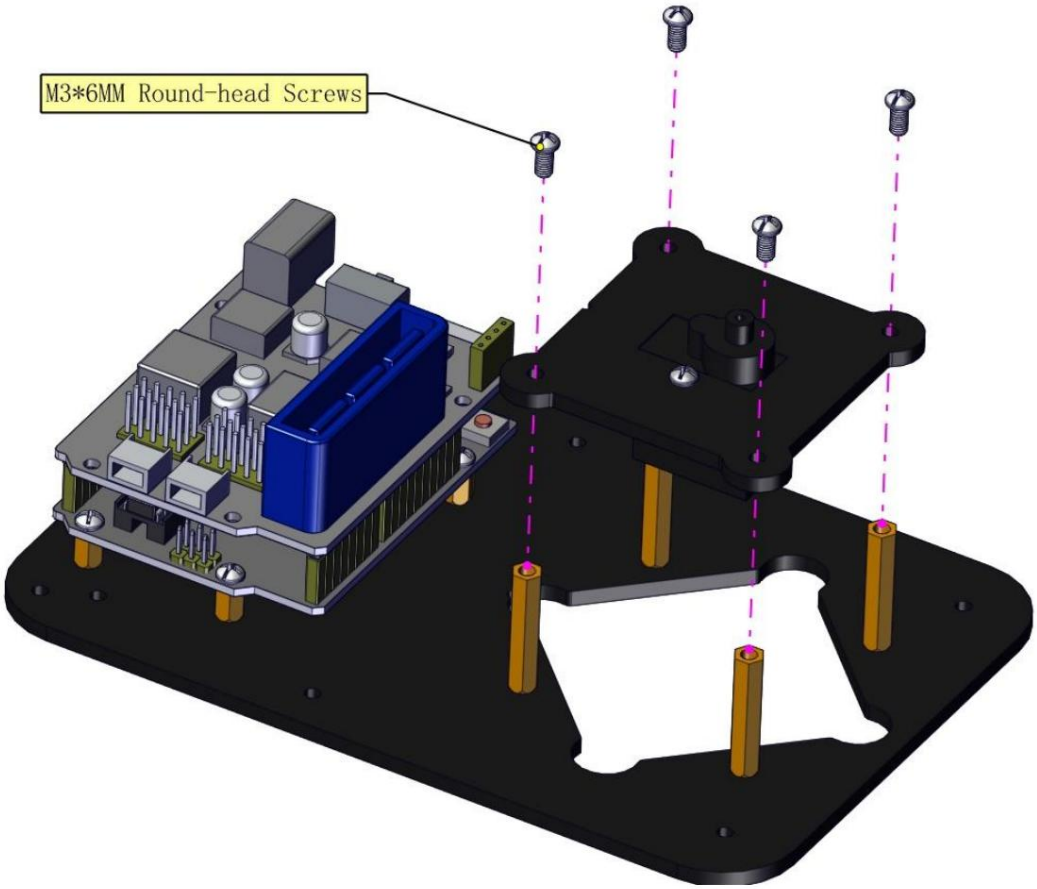
Benodigde componenten:

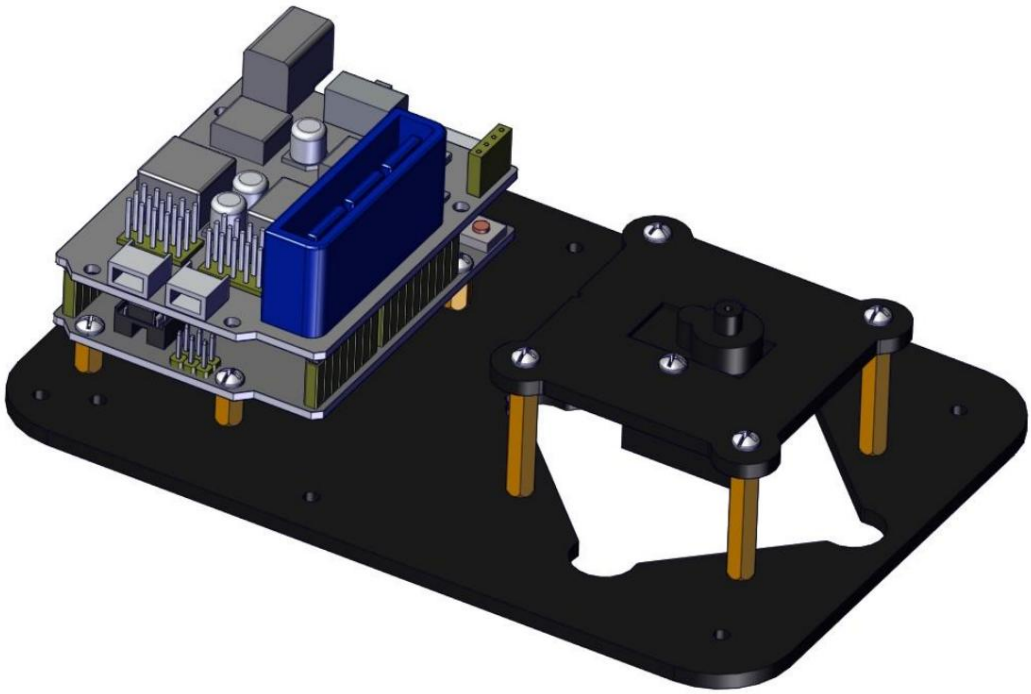


Machine Translated by Google



Machine Translated by Google

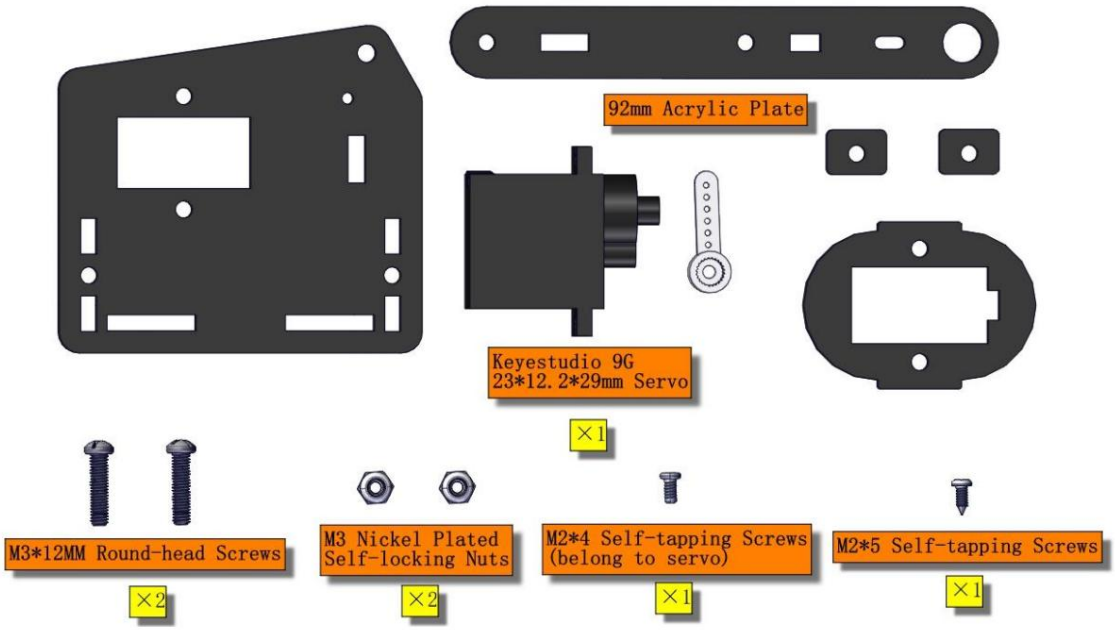


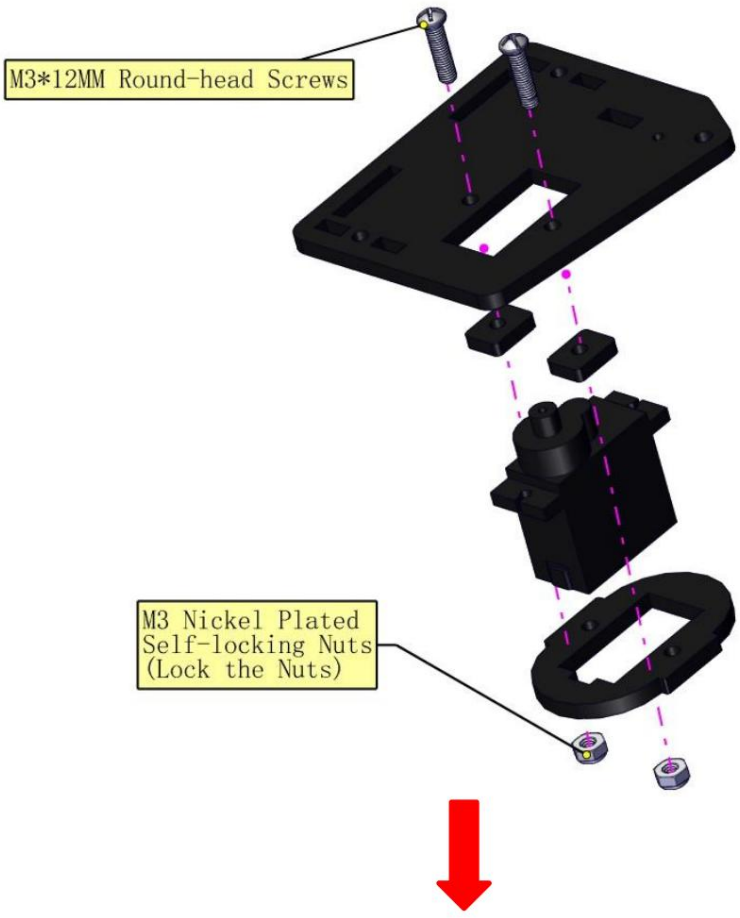


Machine Translated by Google

Monteer een servo (links) op het linkerbord

Benodigde componenten:



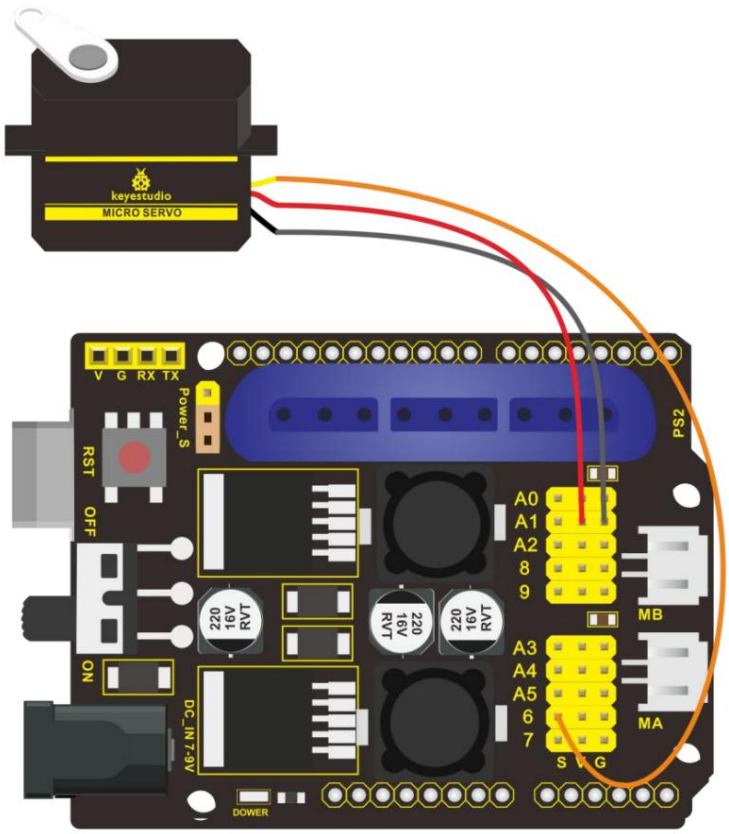


Machine Translated by Google



Initialiseer de linker servo

Bevestig deze linkerservo aan G, V en S (6) van de TB6612FNG-driver shield, upload de volgende code, steek de stekker in het stopcontact en druk op de rustknop op het V4.0-bord. Dan draait de linker servo naar 180°



Machine Translated by Google

Testcode:

********** **********

****

```
#include <Servo.h>

Servo mijnservo; // maak een servo-object om een servo te besturen


ongeldige instellingen ()

{

    Serieel.begin(9600);

    vertraging(1000);

}


ongeldige lus()

{

    mijnservo.attach(6); // Verander pin om één voor één aan te passen

    mijnservo.schrijven(180); //Hoek

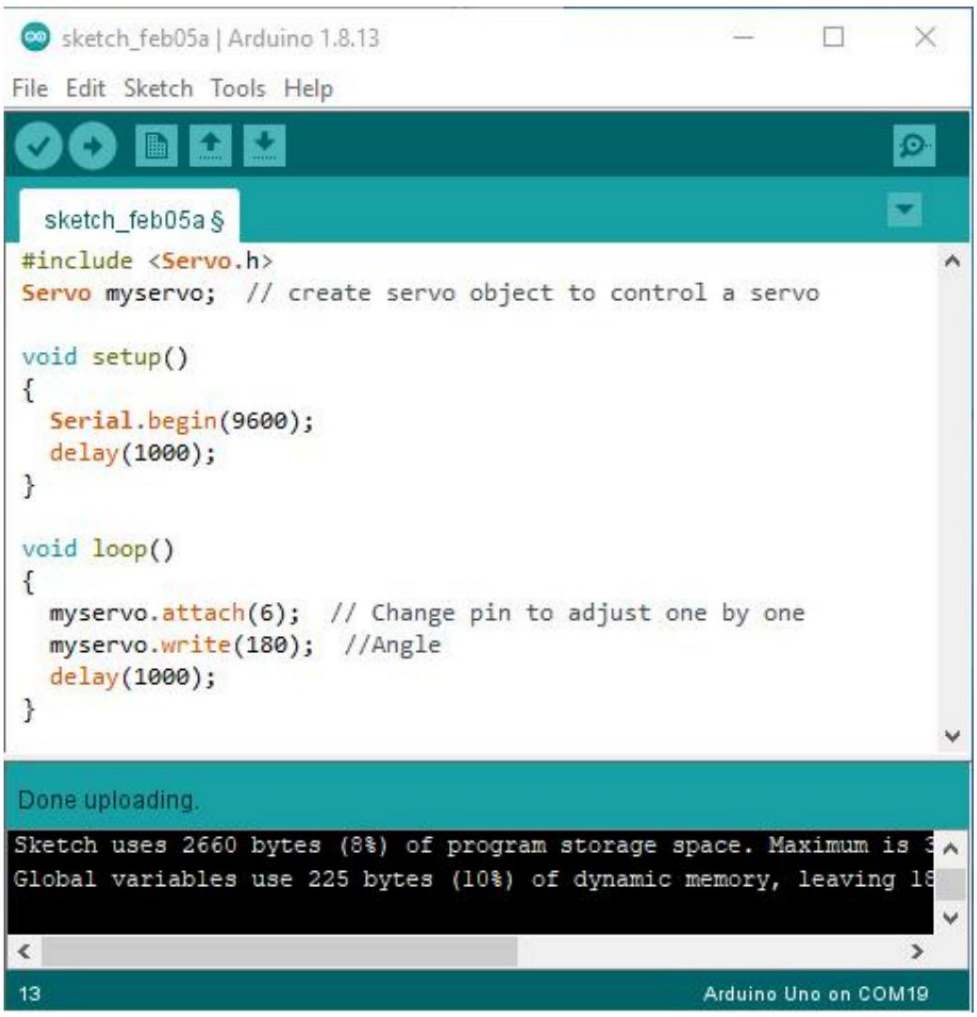
    vertraging(1000);

}
```

********** **********

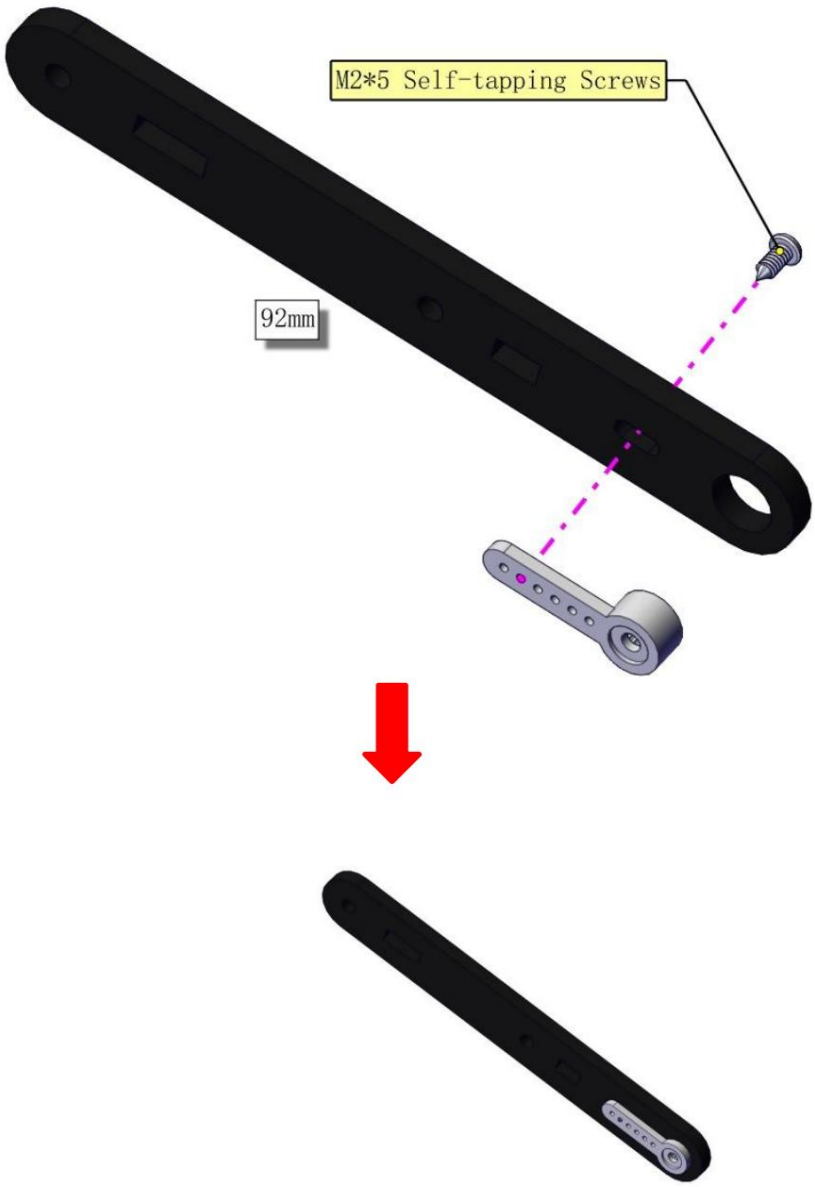
****

Machine Translated by Google

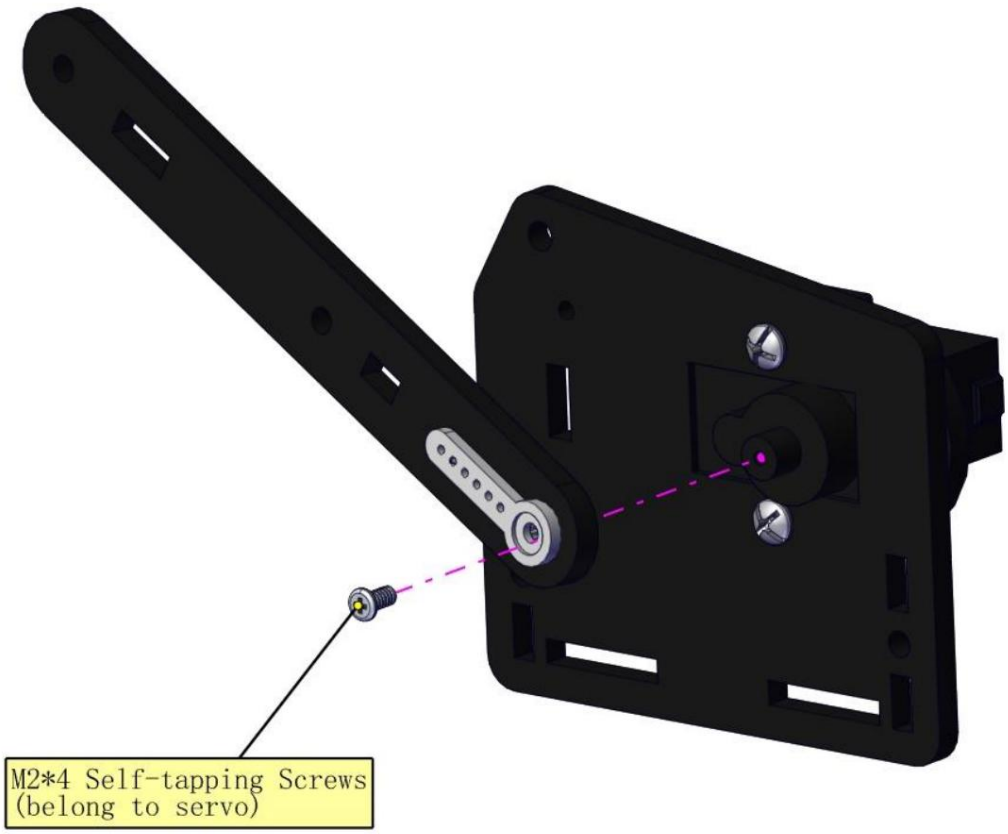


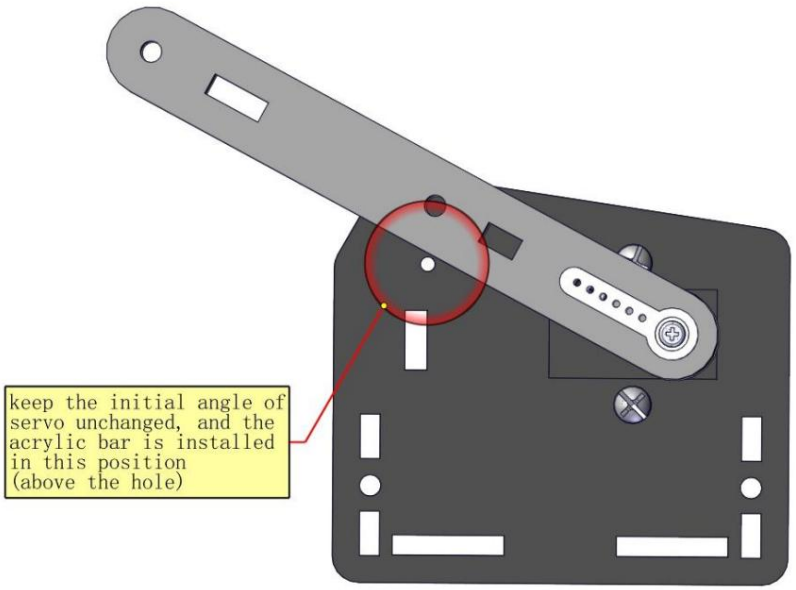
Machine Translated by Google

A. Bevestig de arm:

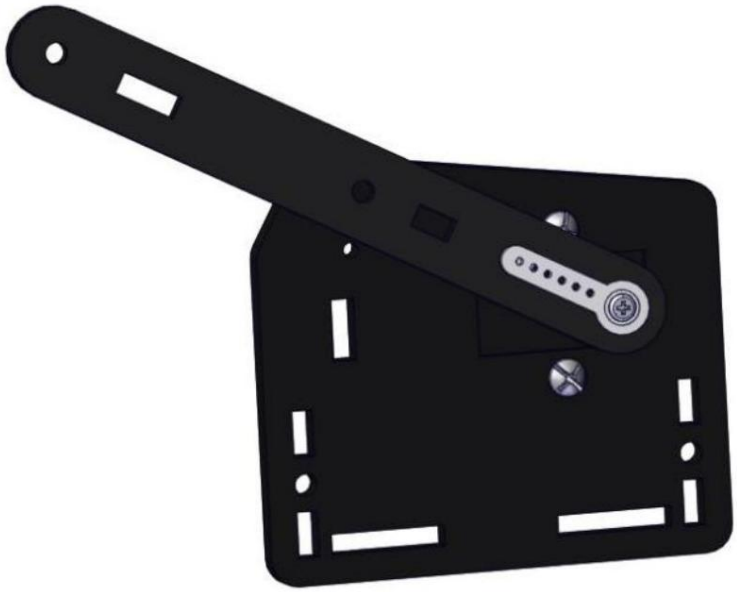


Machine Translated by Google



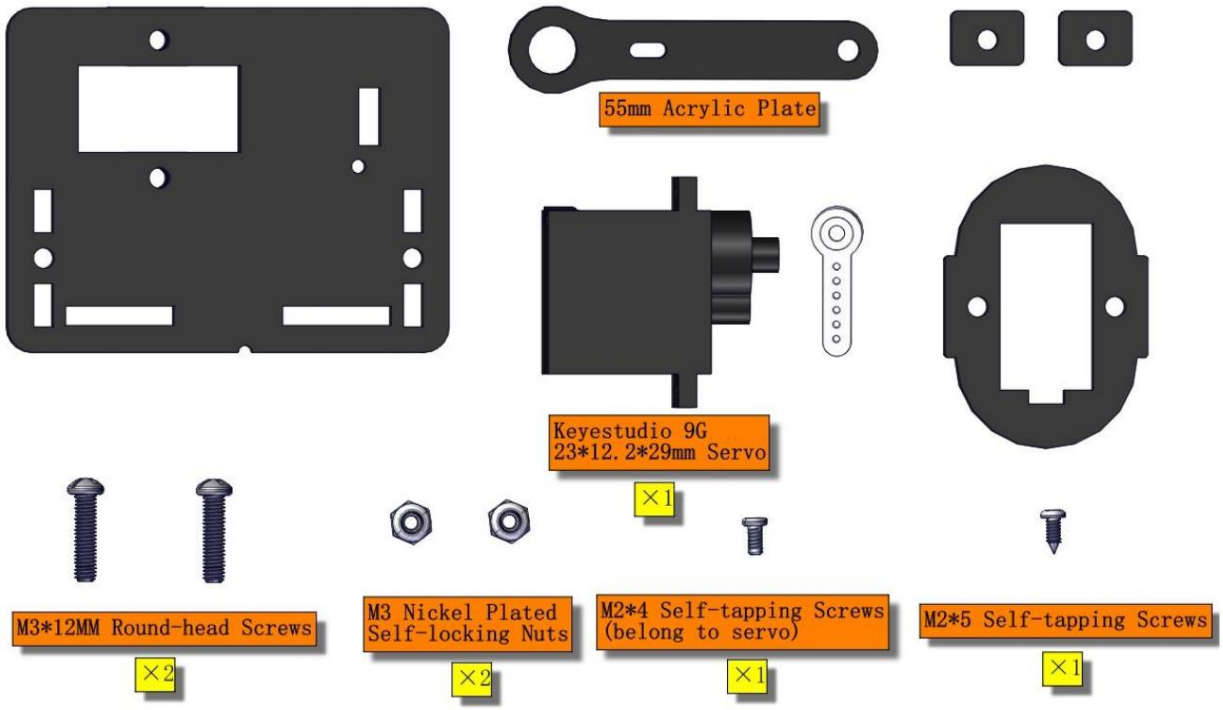


Machine Translated by Google



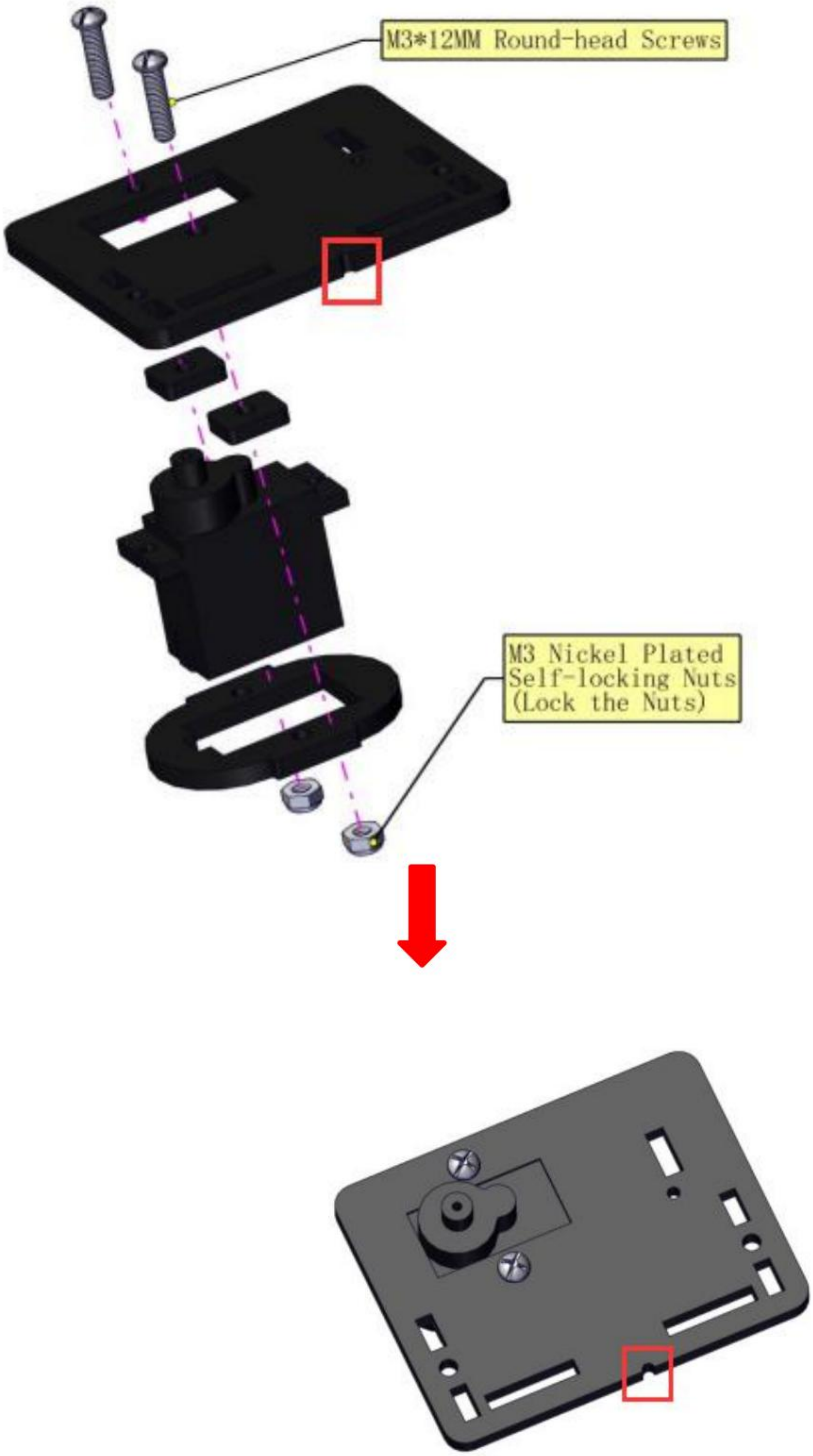
Monteer een servo (rechts) op het rechterbord

Componenten nodig



Machine Translated by Google

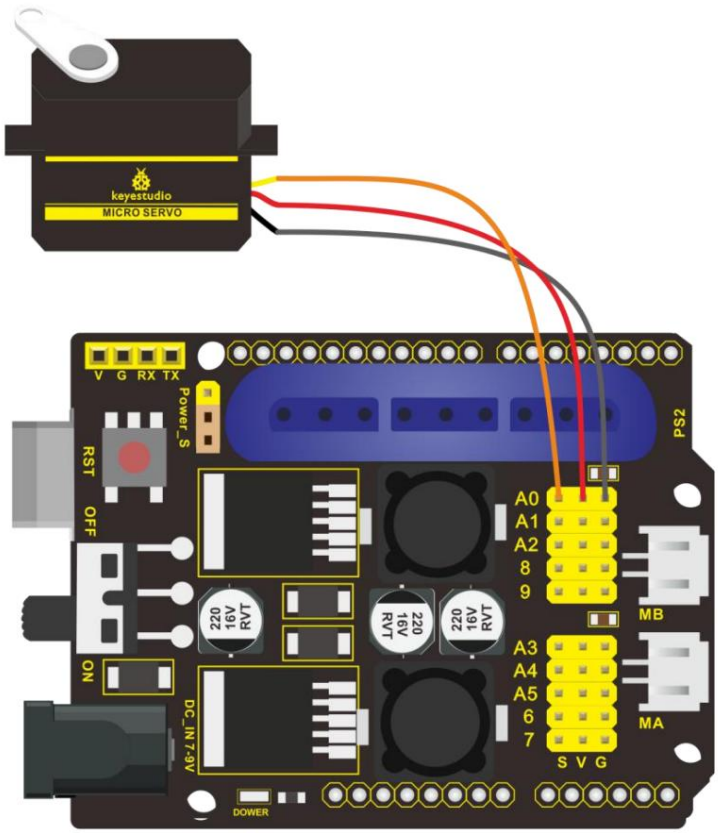
Let op de breukrichting van acrylplaat



Machine Translated by Google

B. Initialiseer de juiste servo

Bevestig deze linkerservo aan G, V en S (A0) van de TB6612FNG-driver shield, upload de volgende code, steek de stekker in het stopcontact en druk op de rustknop op het V4.0-bord. Dan draait de linker servo naar 0°



Zet de servo op 0°ÿ

********** **********

****

Machine Translated by Google


```
#include <Servo.h>

Servo mijnservo; // maak een servo-object om een servo te besturen



ongeldige instellingen ()

{

    Serieel.begin(9600);

    vertraging(1000);

}



ongeldige lus()

{

    mijnservo.attach(A0); // Verander pin om één voor één aan te passen

    mijnservo.write(0); //Hoek

    vertraging(1000);

}

********** **********

**********
```

Machine Translated by Google

sketch_feb05a | Arduino 1.8.13

File Edit Sketch Tools Help

sketch_feb05a §

```
#include <Servo.h>
Servo myservo;  // create servo object to control a servo

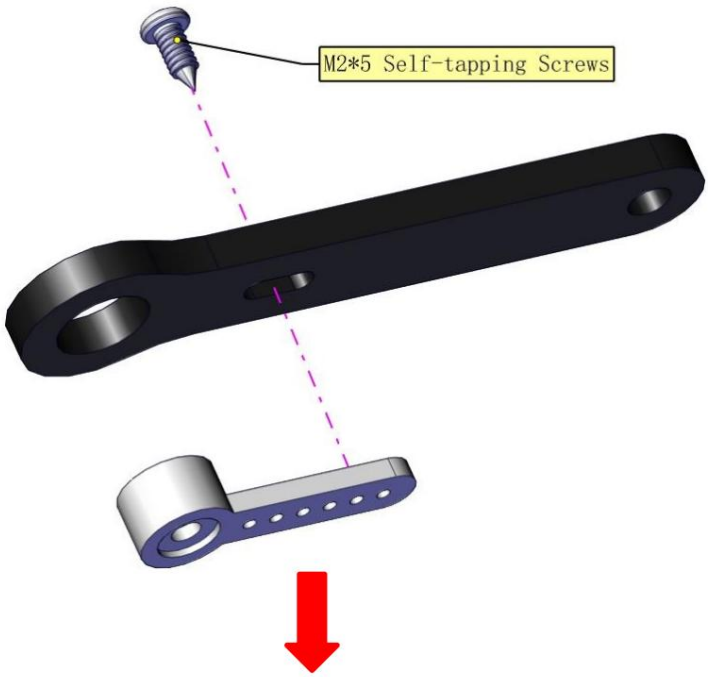
void setup()
{
  Serial.begin(9600);
  delay(1000);
}

void loop()
{
  myservo.attach(A0);  // Change pin to adjust one by one
  myservo.write(0);  //Angle
  delay(1000);
}
```

Done uploading.

```
Sketch uses 2658 bytes (8%) of program storage space. Maximum is 3
Global variables use 225 bytes (10%) of dynamic memory, leaving 18
```

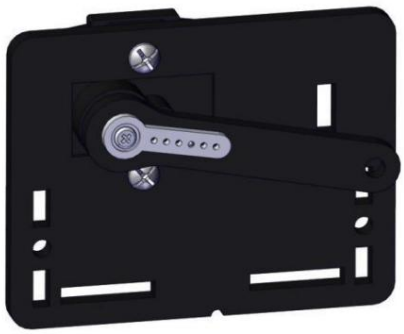
15 Arduino Uno on COM19



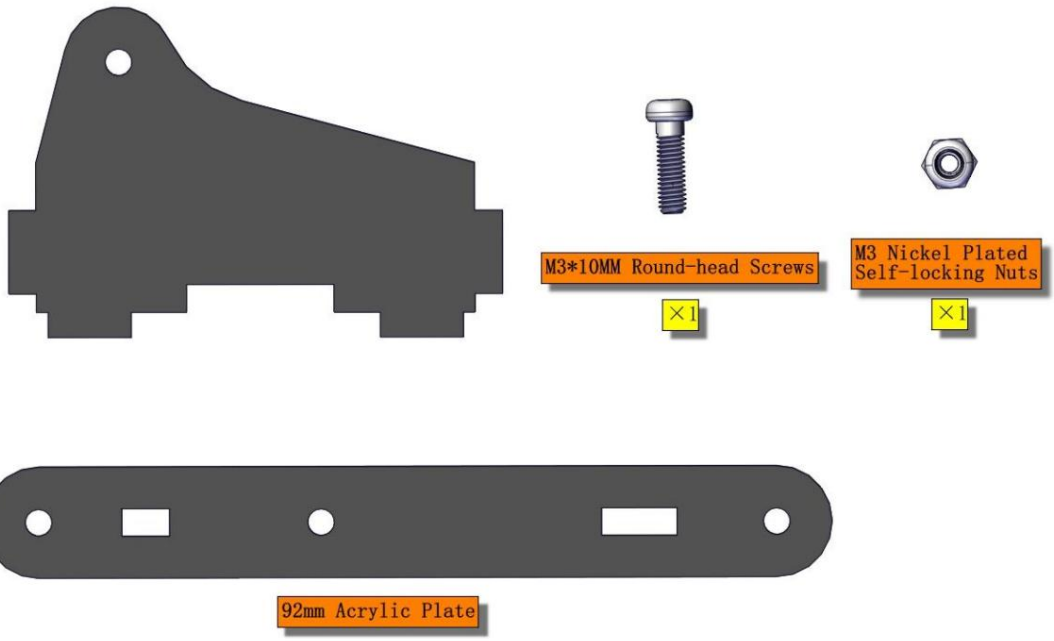
Machine Translated by Google



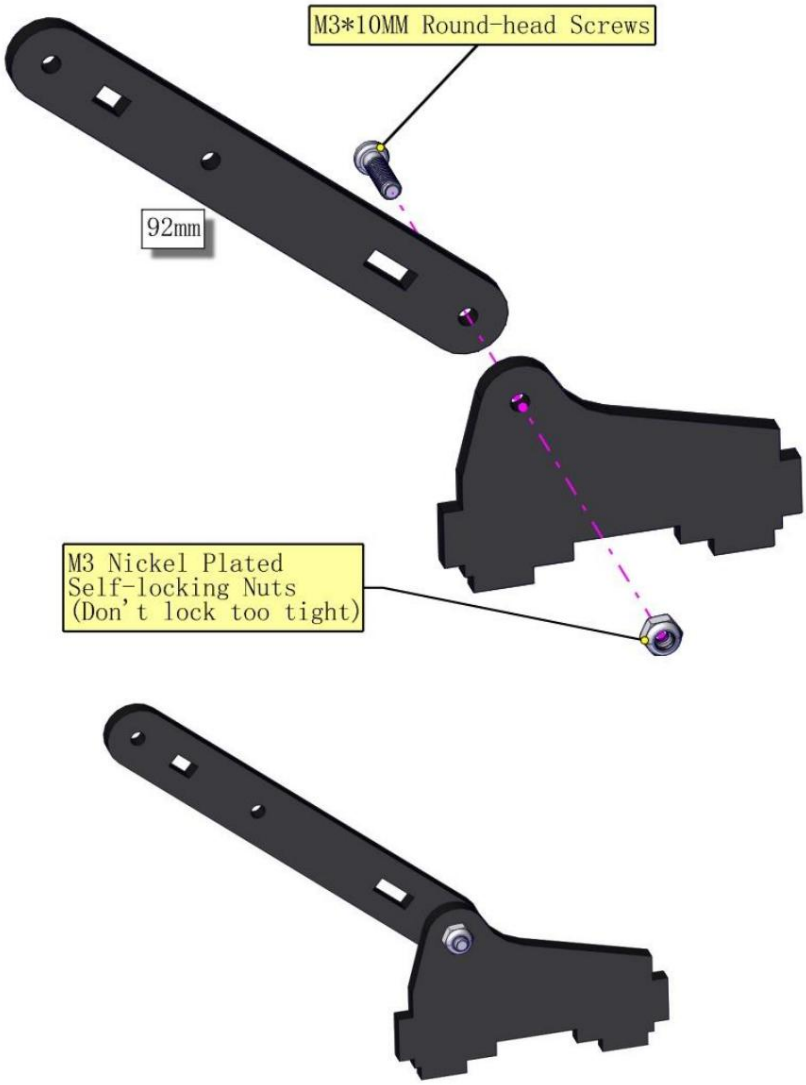
Machine Translated by Google



Installeer het houdergedeelte

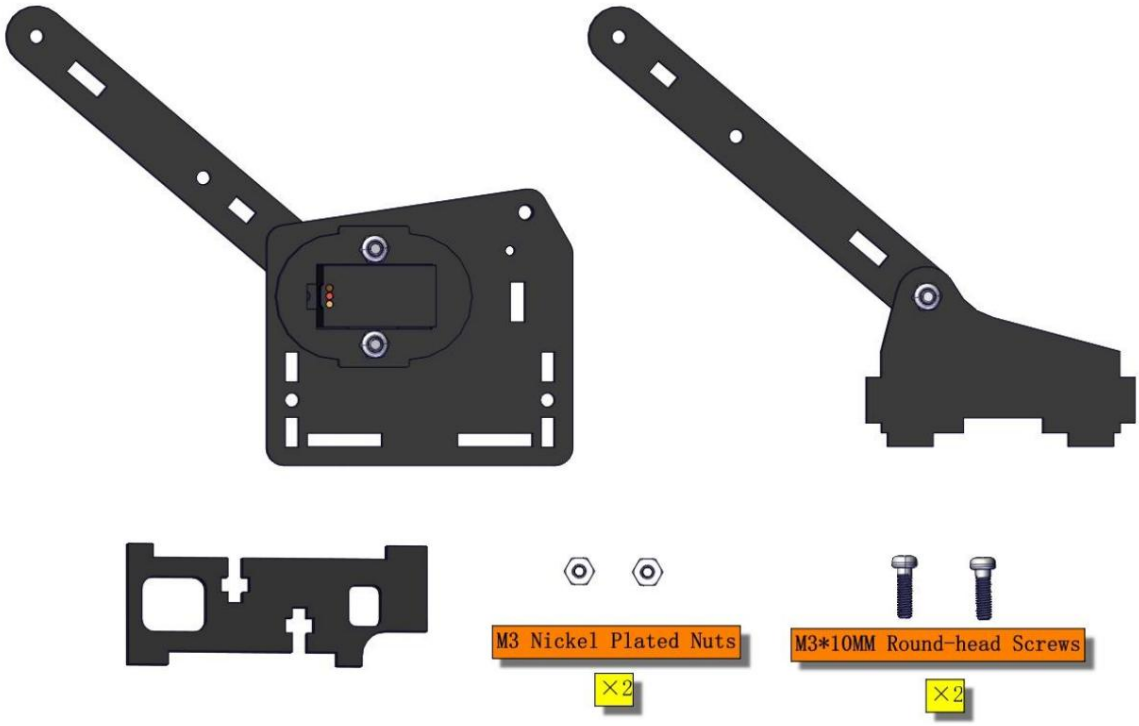


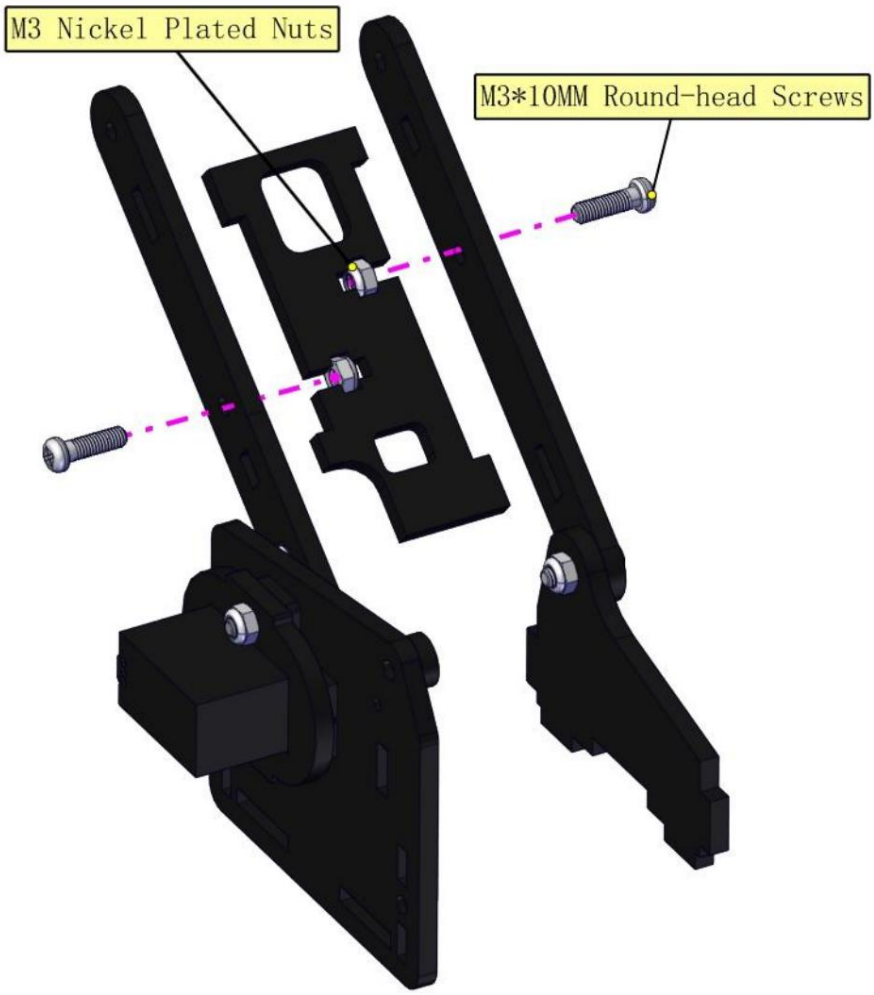
Machine Translated by Google



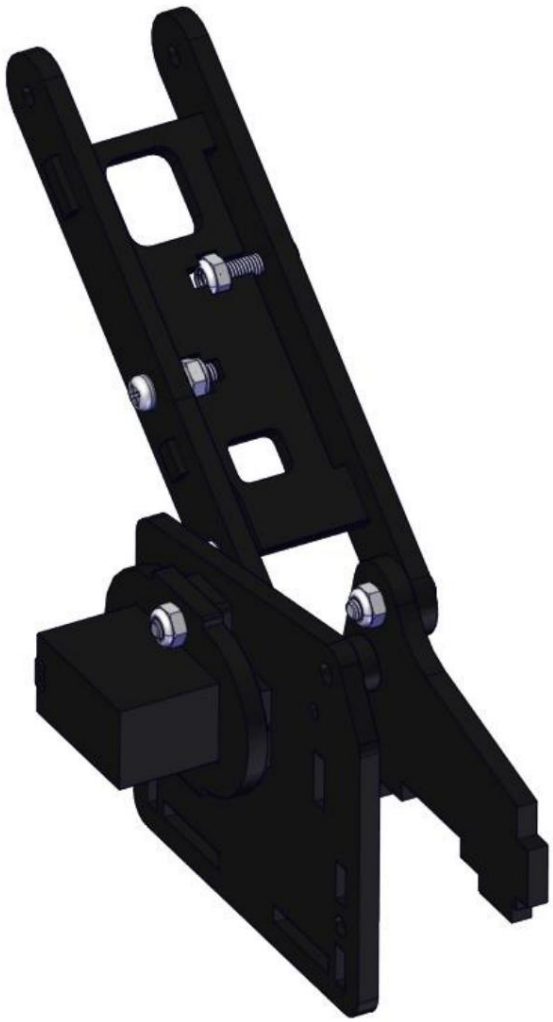
Bevestig het linkerdeel en het montagedeel aan elkaar

Machine Translated by Google



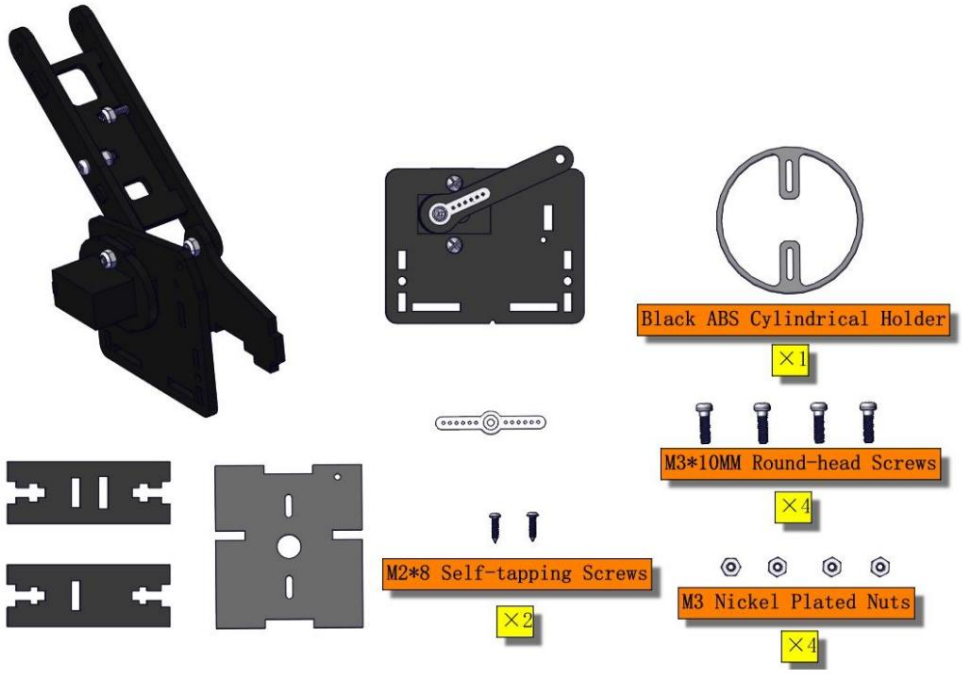


Machine Translated by Google

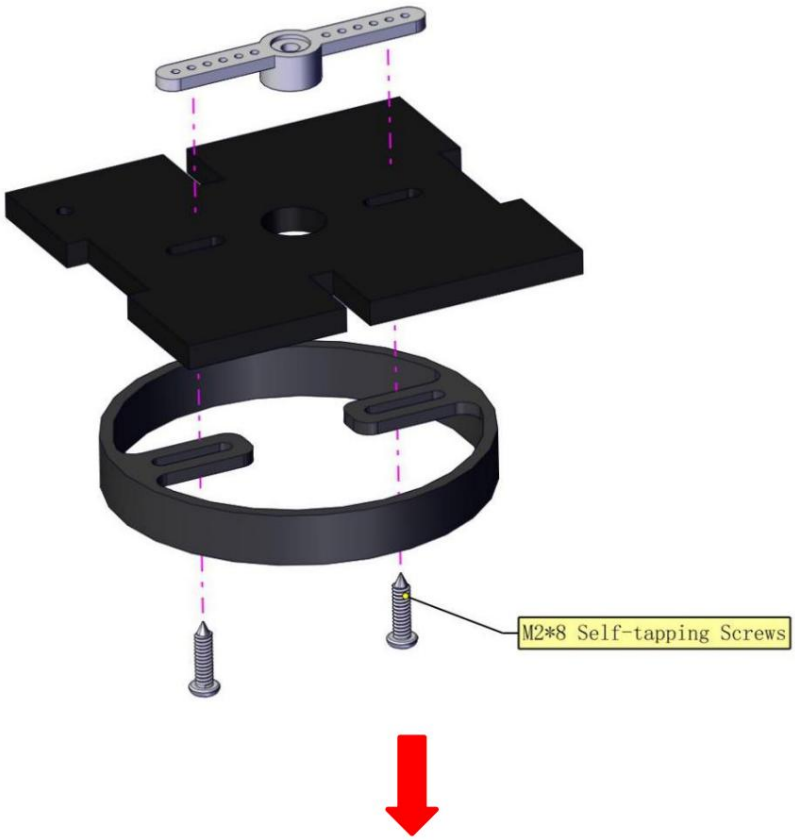


Bevestig het rechterdeel en de ABS-houder aan elkaar

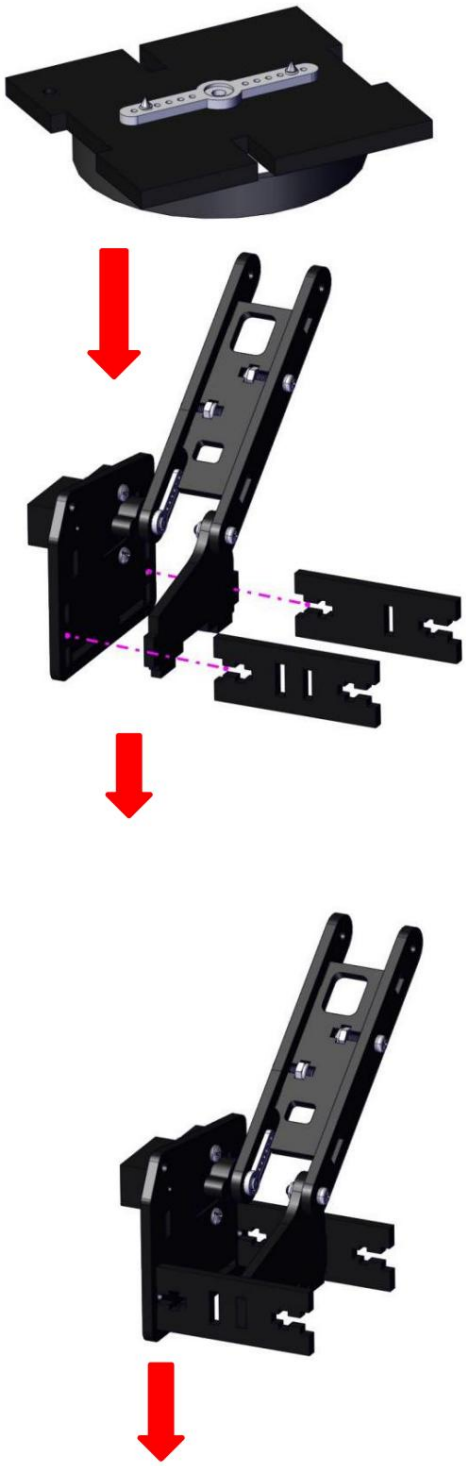
Machine Translated by Google



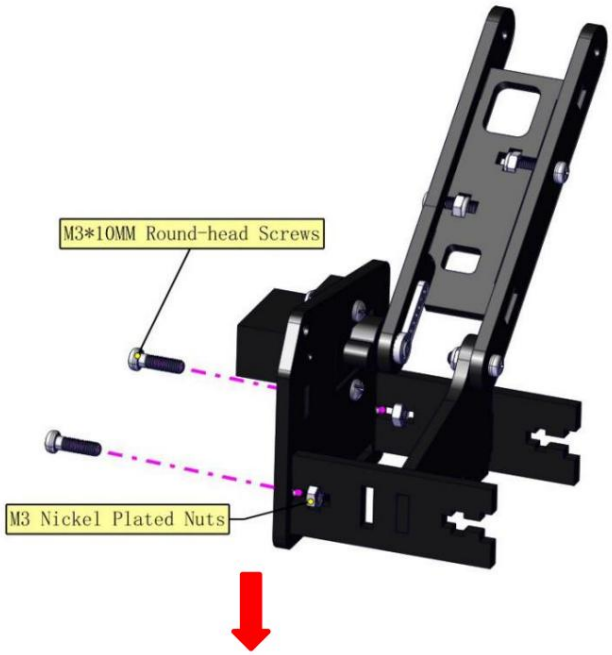
Let op de richting van de ABS-houder



Machine Translated by Google

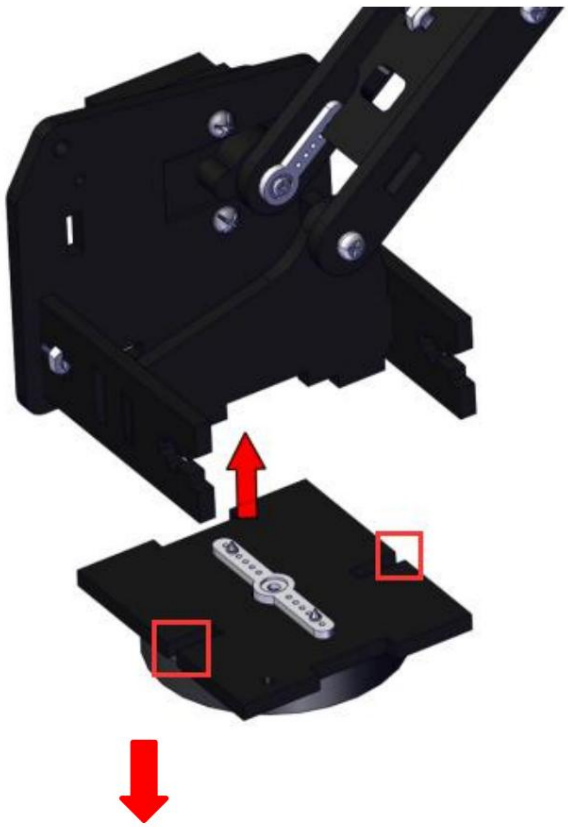


Machine Translated by Google



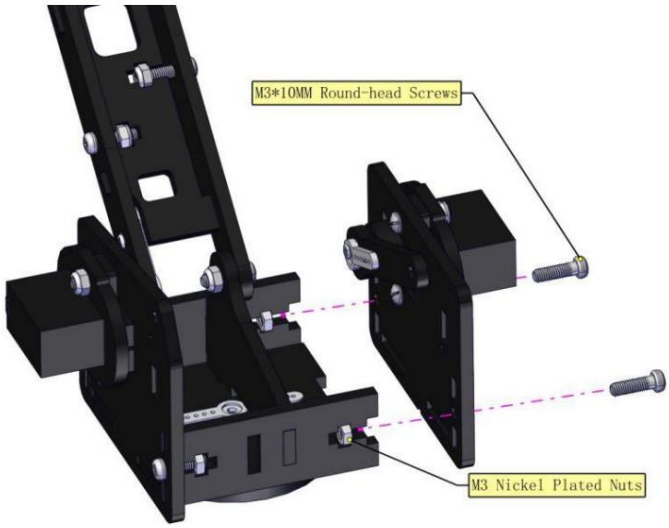


Machine Translated by Google





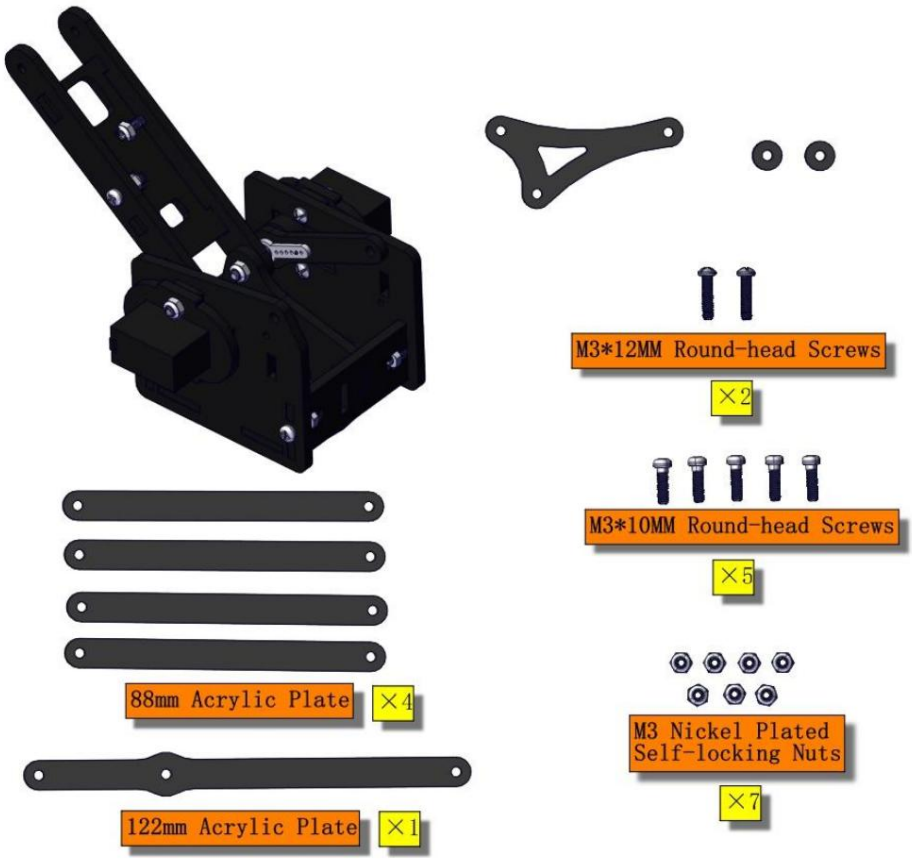
Machine Translated by Google

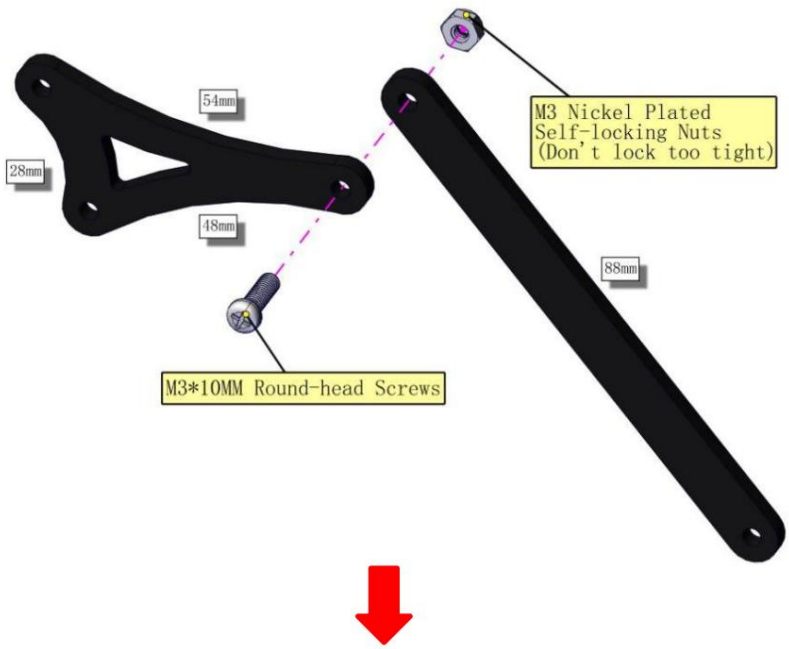




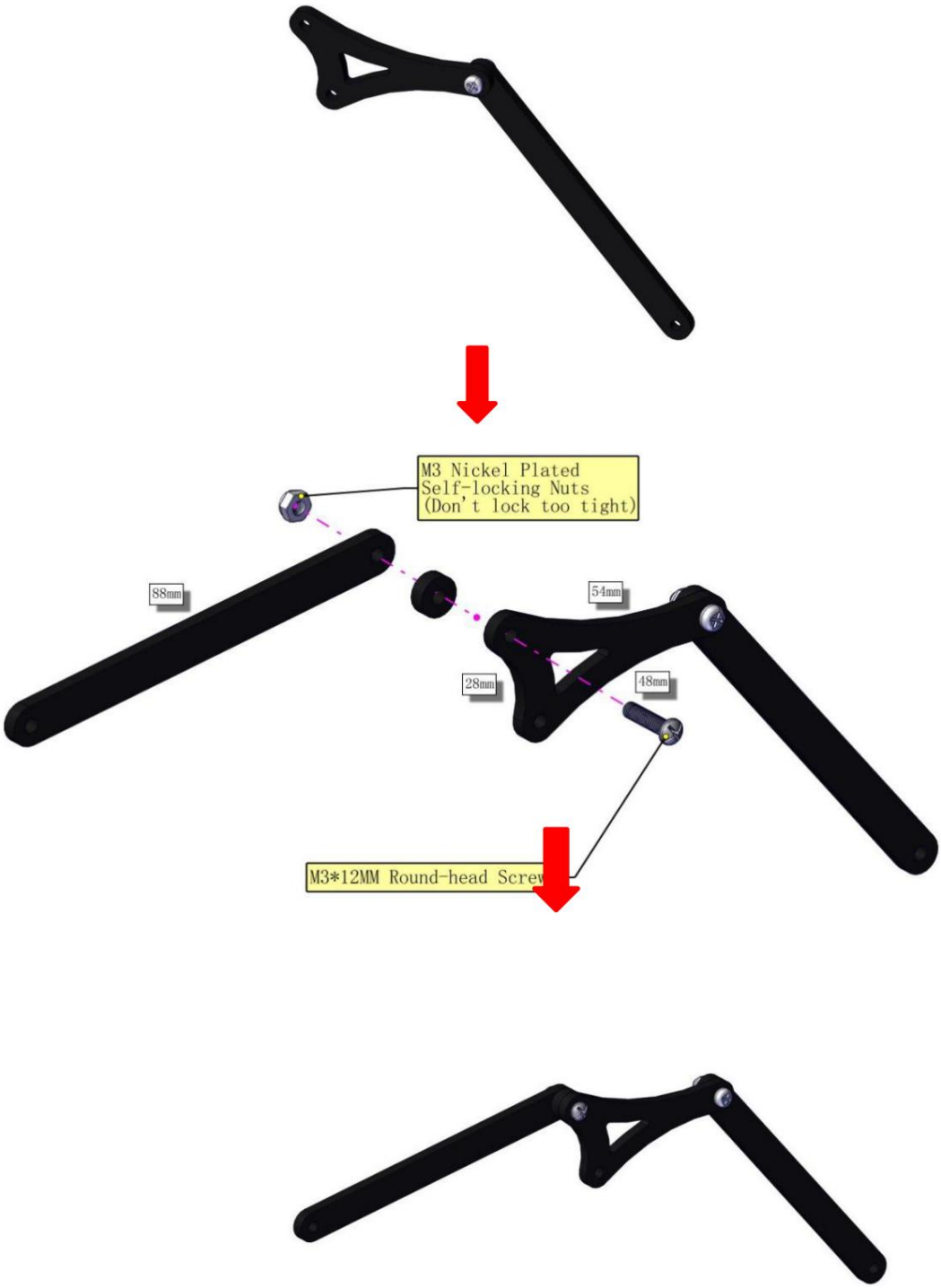
Monteer het middelste deel

Machine Translated by Google

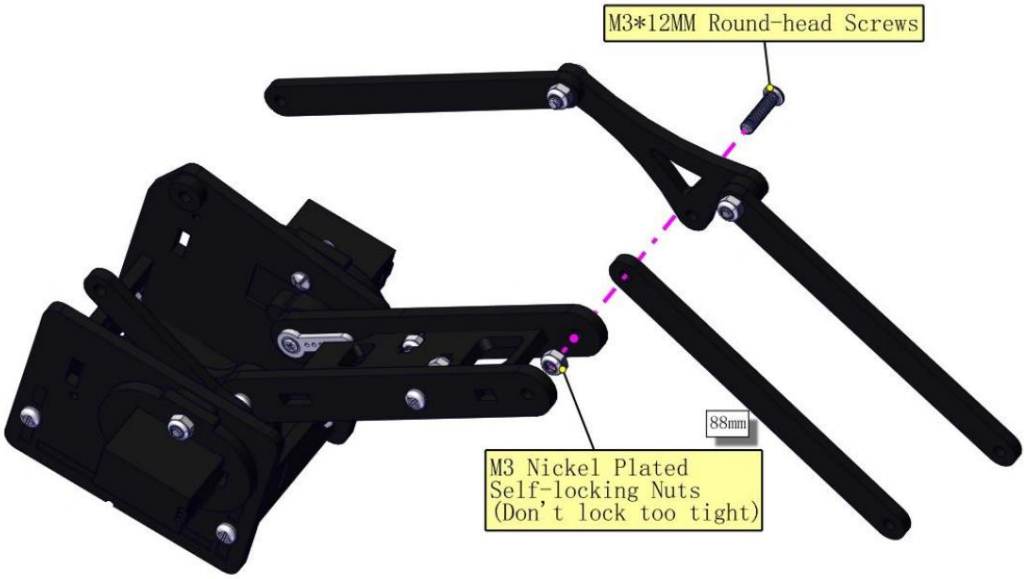


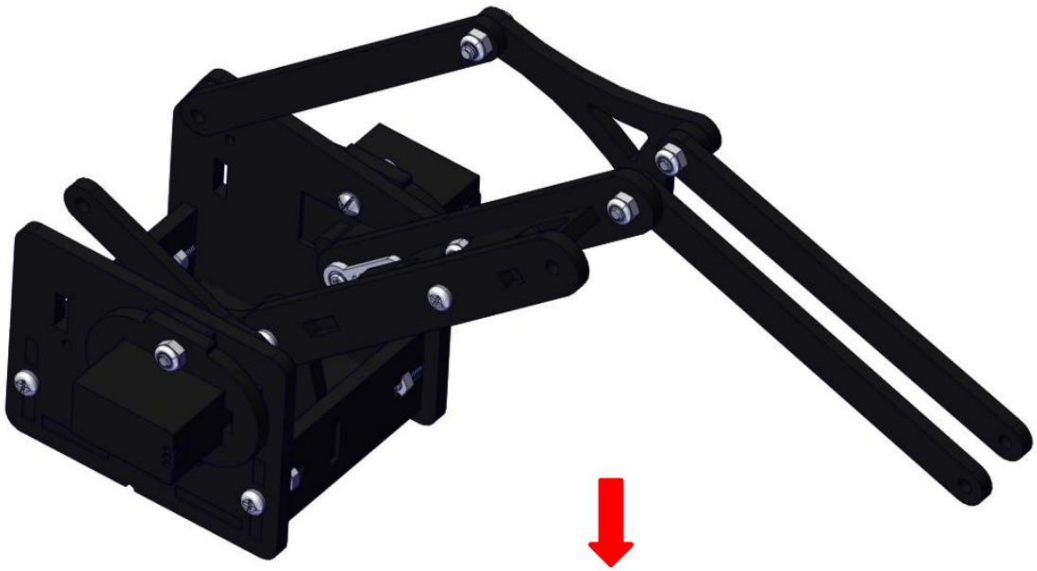


Machine Translated by Google

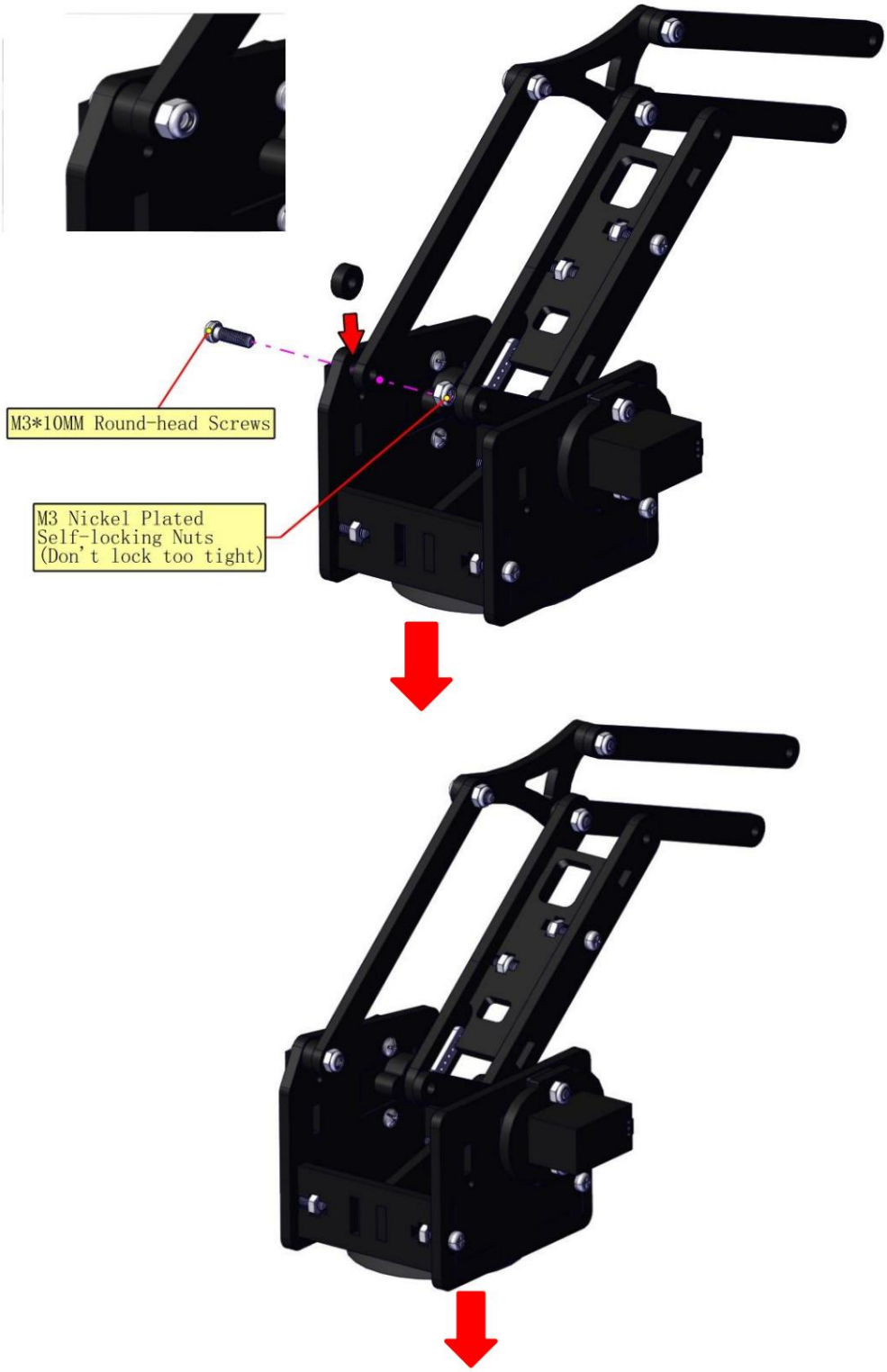


Machine Translated by Google

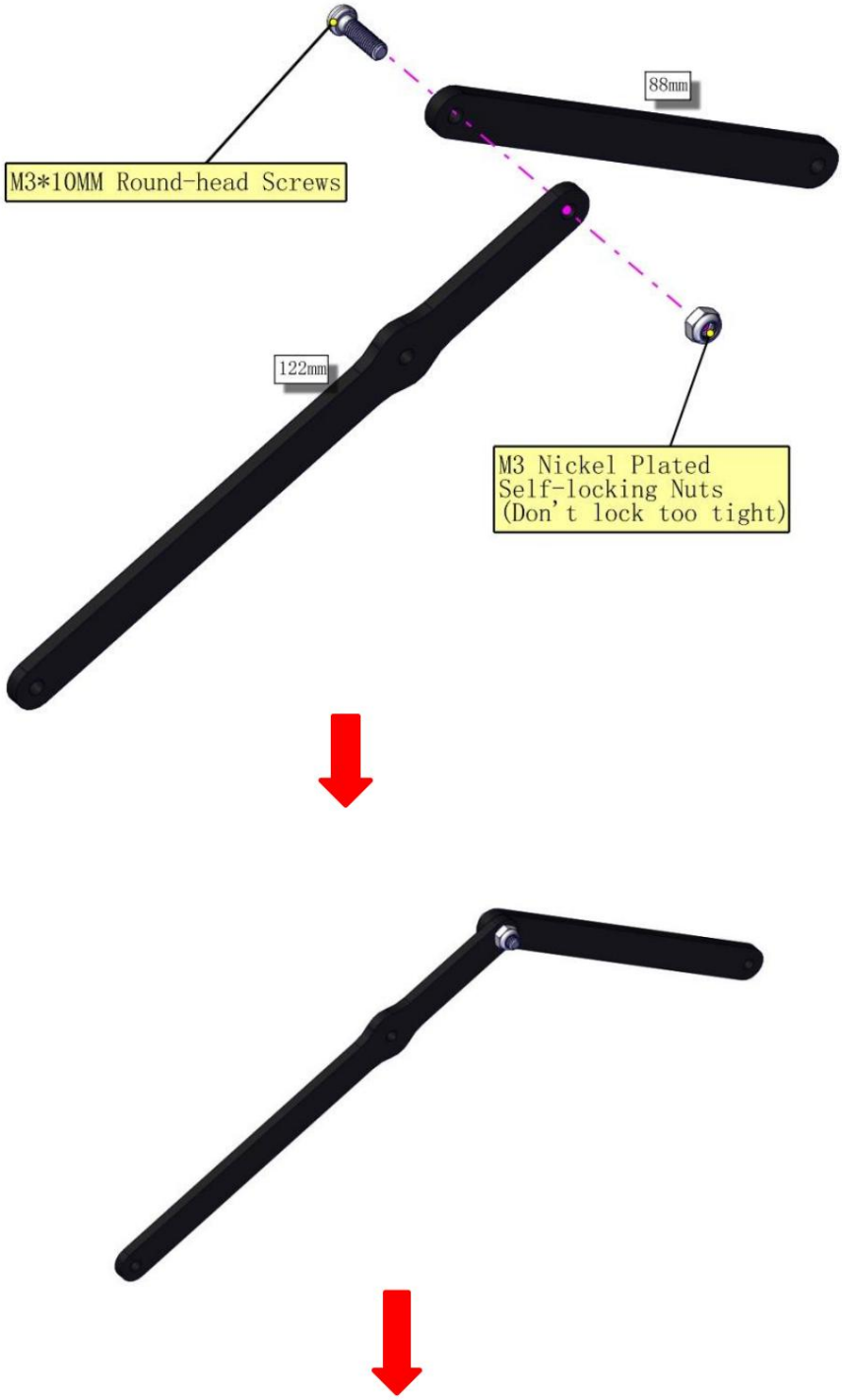




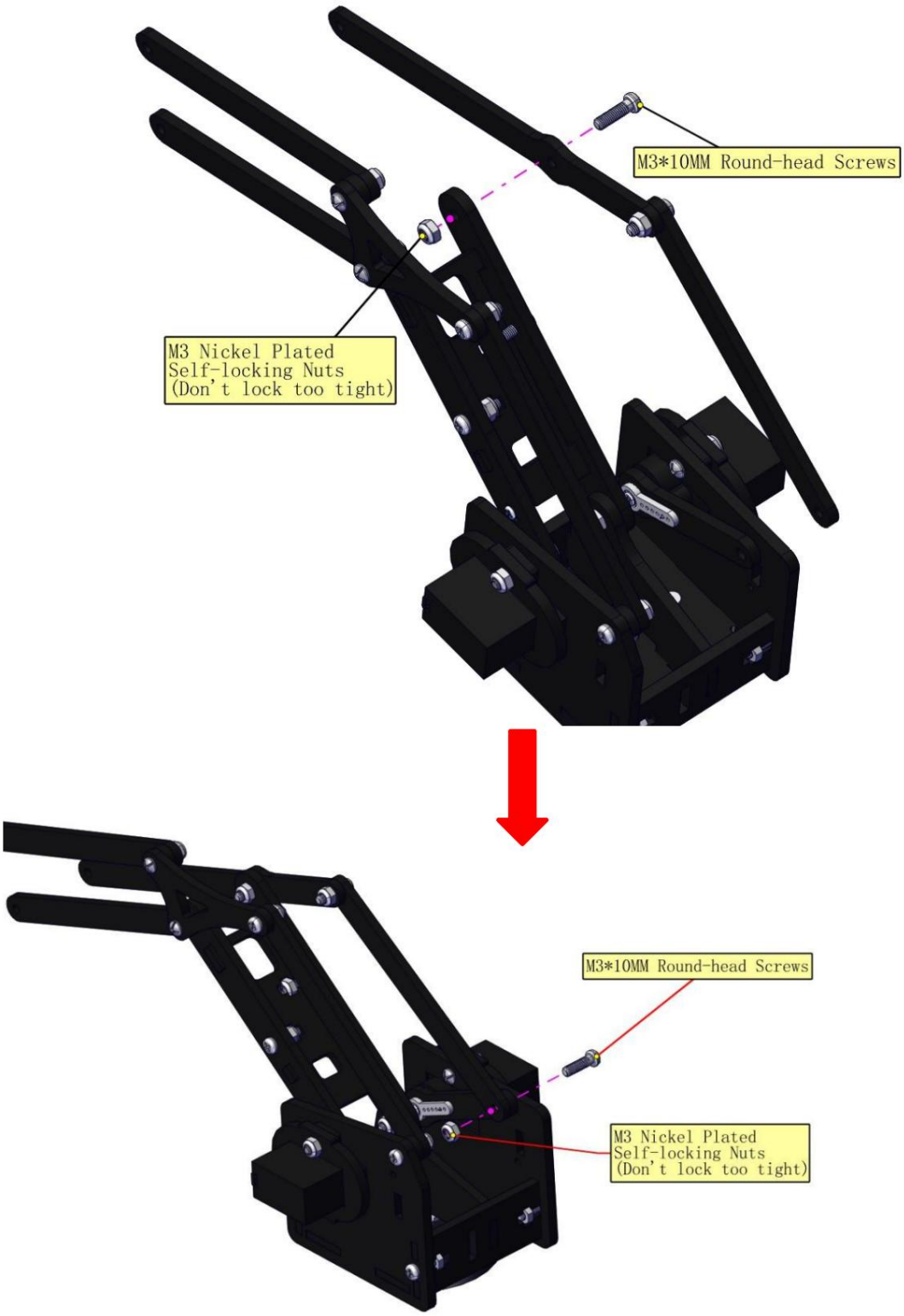
Machine Translated by Google



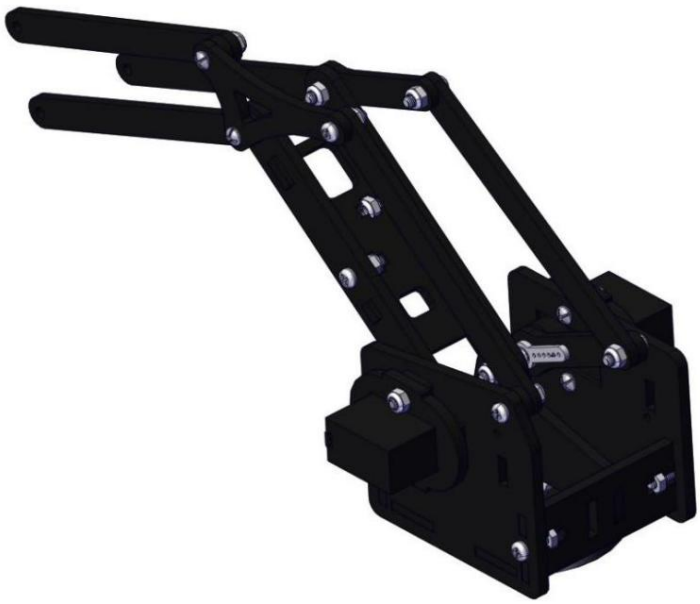
Machine Translated by Google



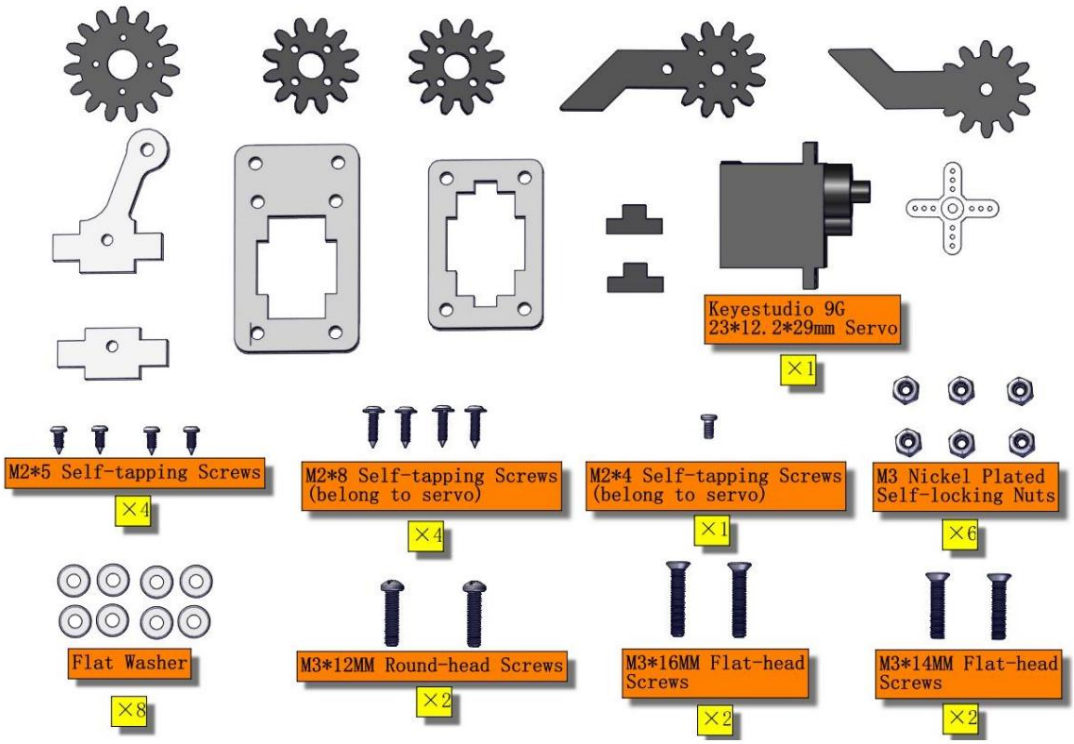
Machine Translated by Google



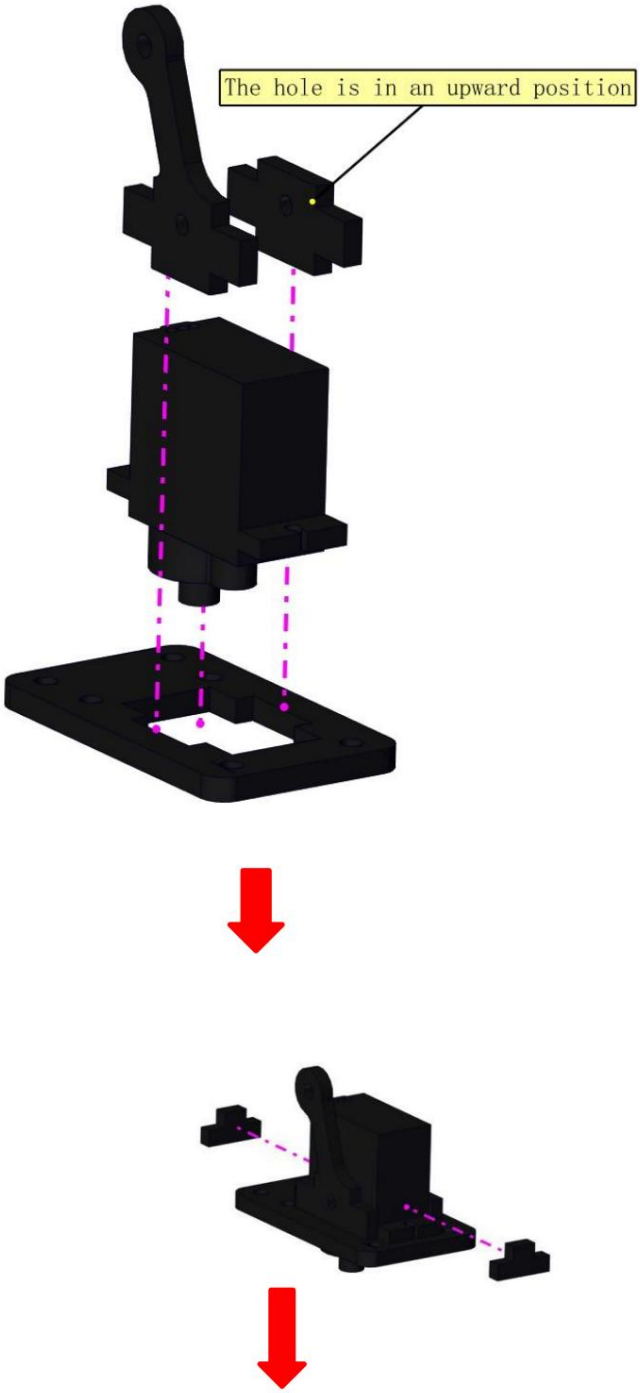
Machine Translated by Google



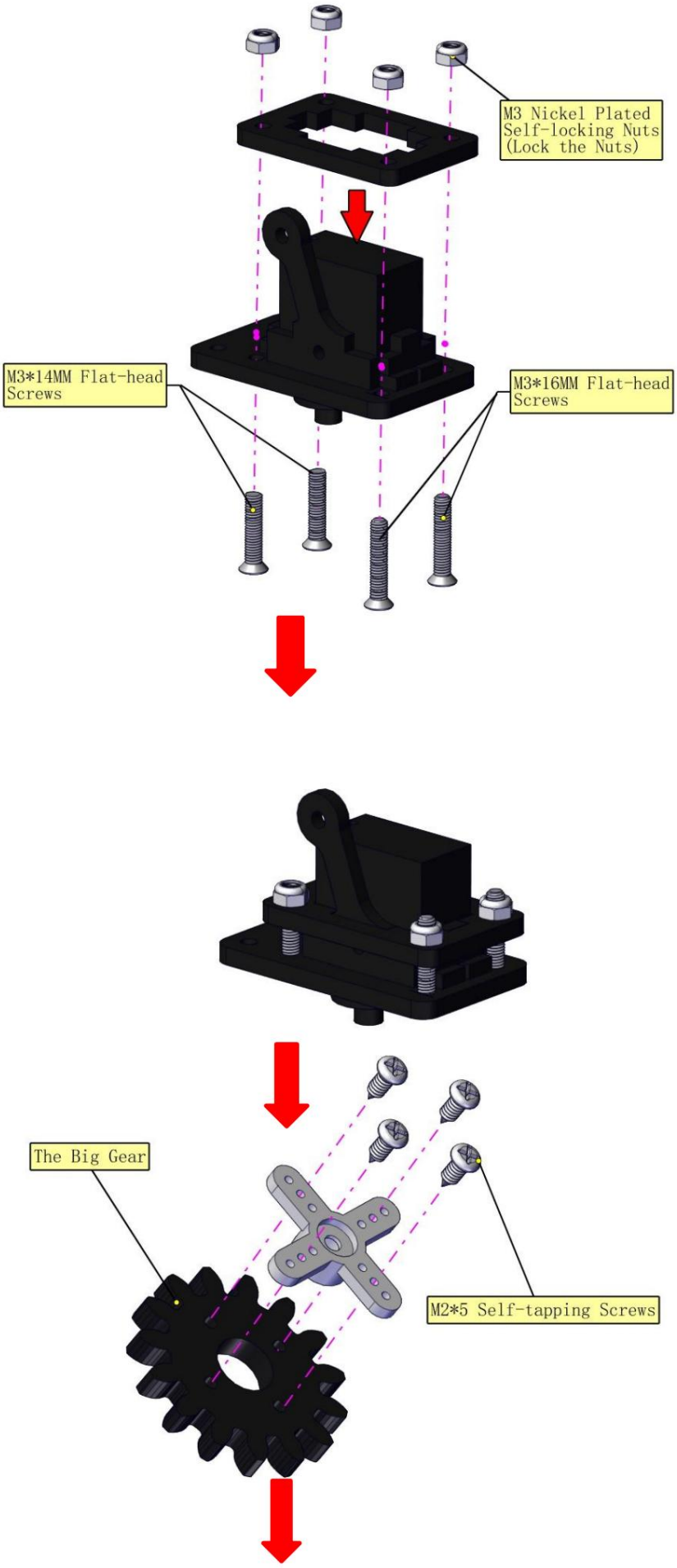
Monteer de klauw



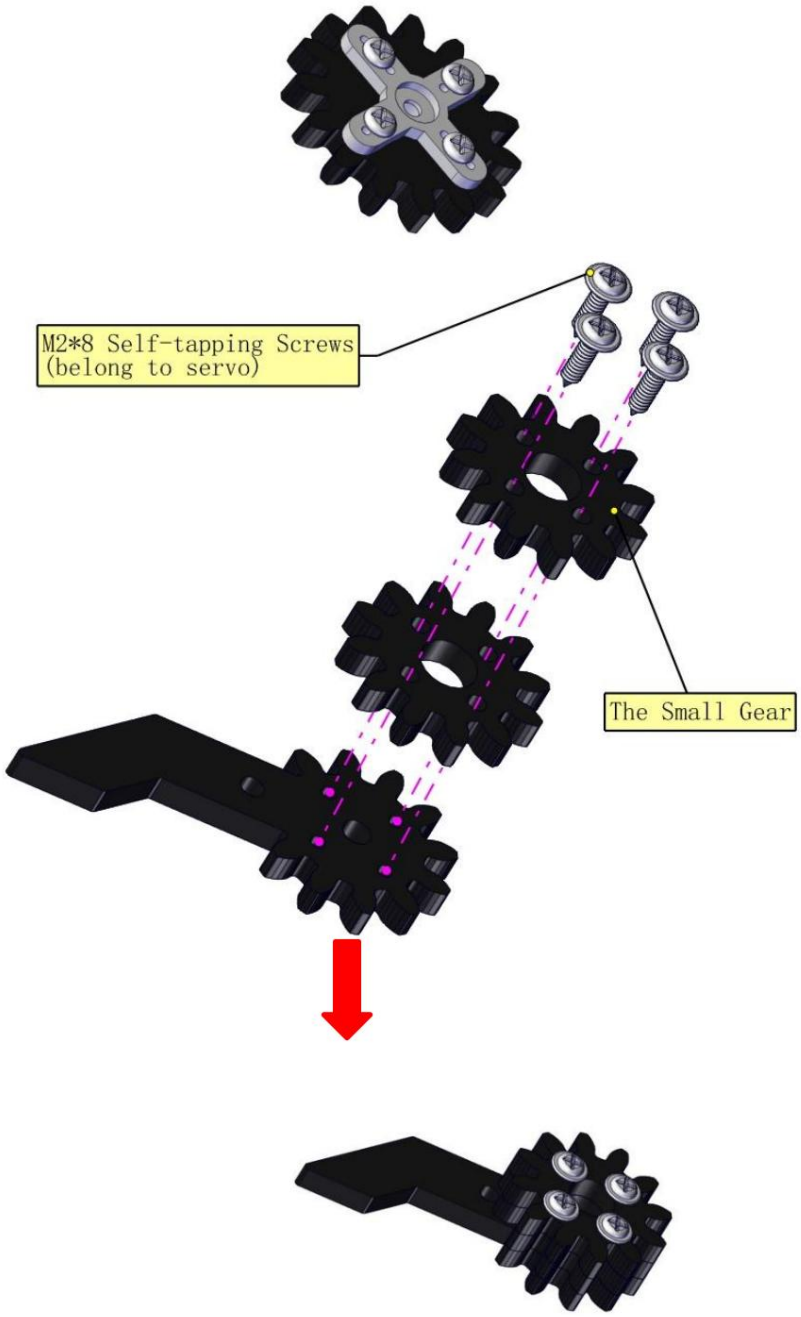
Machine Translated by Google



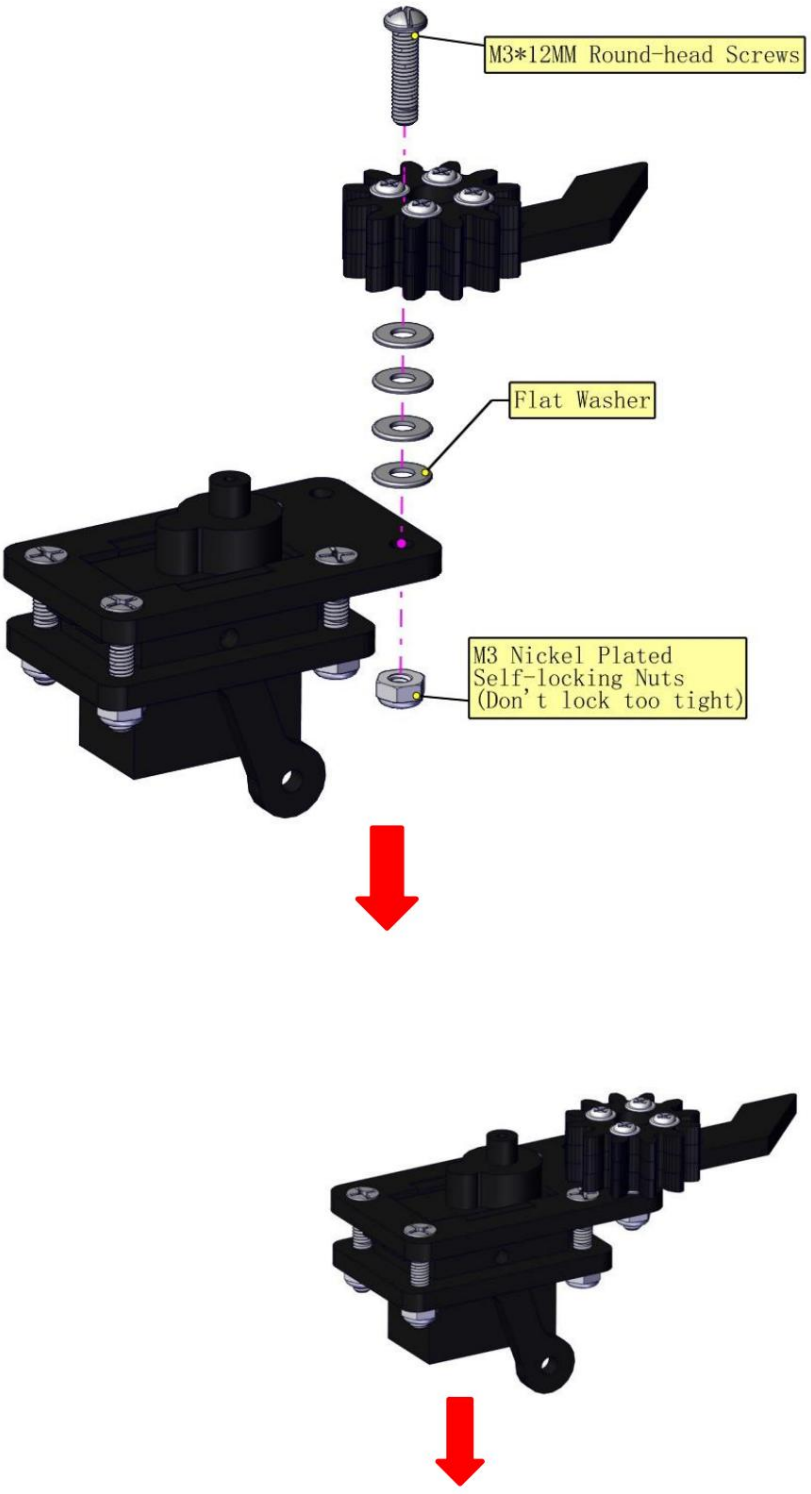
Machine Translated by Google



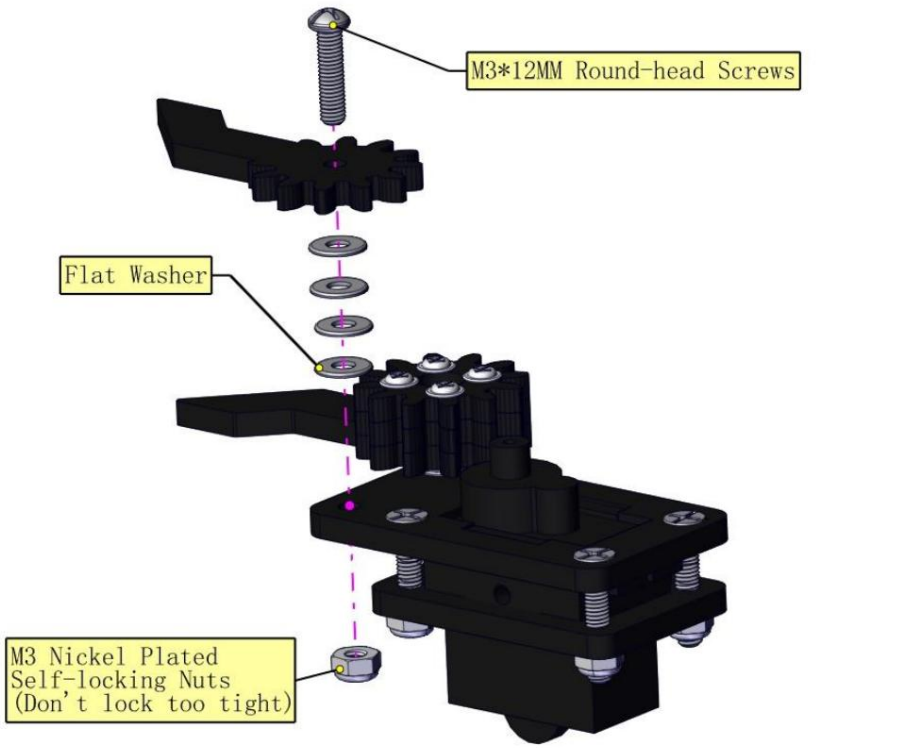
Machine Translated by Google



Machine Translated by Google



Machine Translated by Google





C. Initialiseer de servo

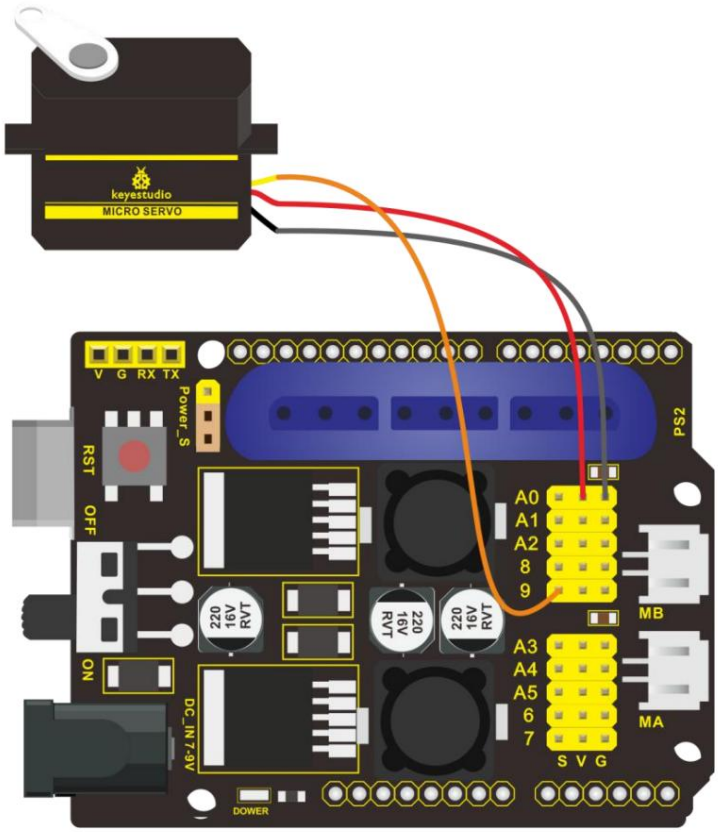
Bevestig deze linkerservo aan G, V en S (9) van de TB6612FNG-driver

shield, upload de volgende code, steek de stekker in het stopcontact en druk op

de rustknop op het V4.0-bord. Dan draait de linker servo

naar 0°

Machine Translated by Google



Zet de servo op 0°ÿ

********** **********

****

```
#include <Servo.h>

Servo mijnservo; // maak een servo-object om een servo te besturen


ongeldige instellingen ()

{

    Serieel.begin(9600);

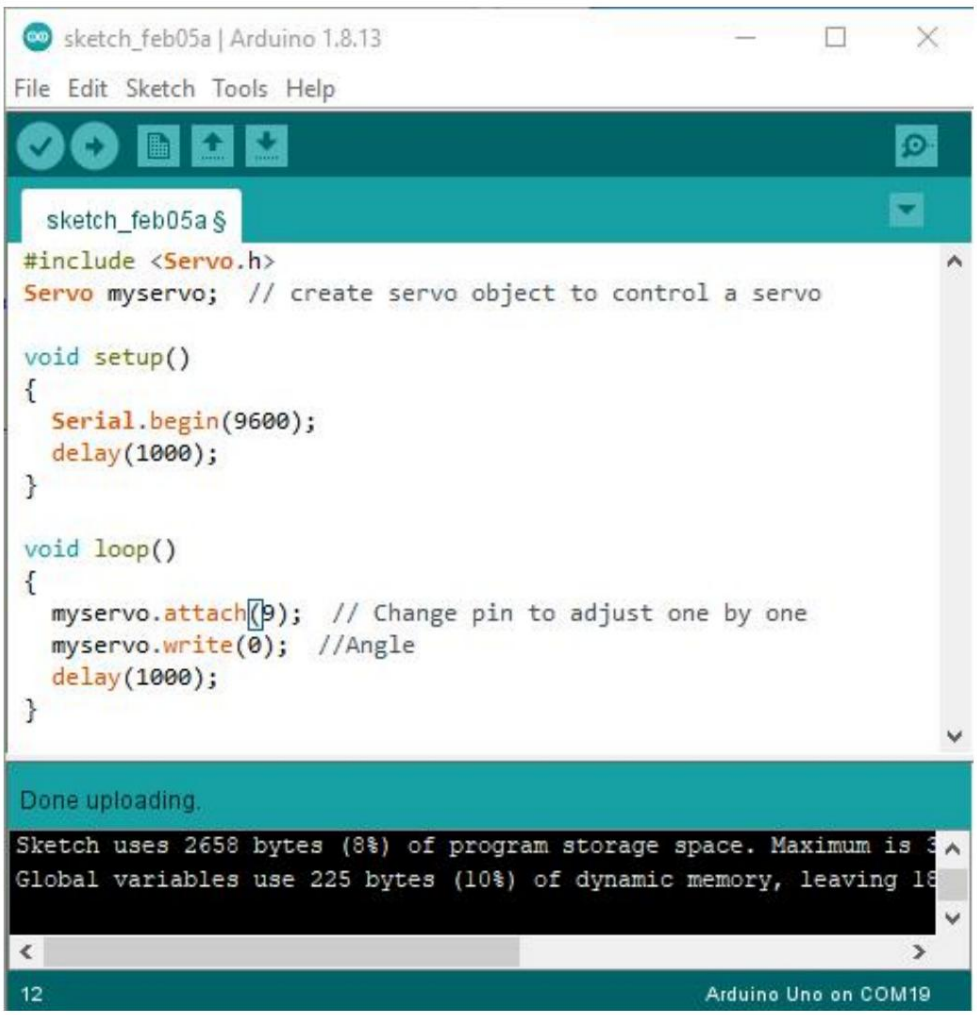
    vertraging(1000);
```

Machine Translated by Google


```
}

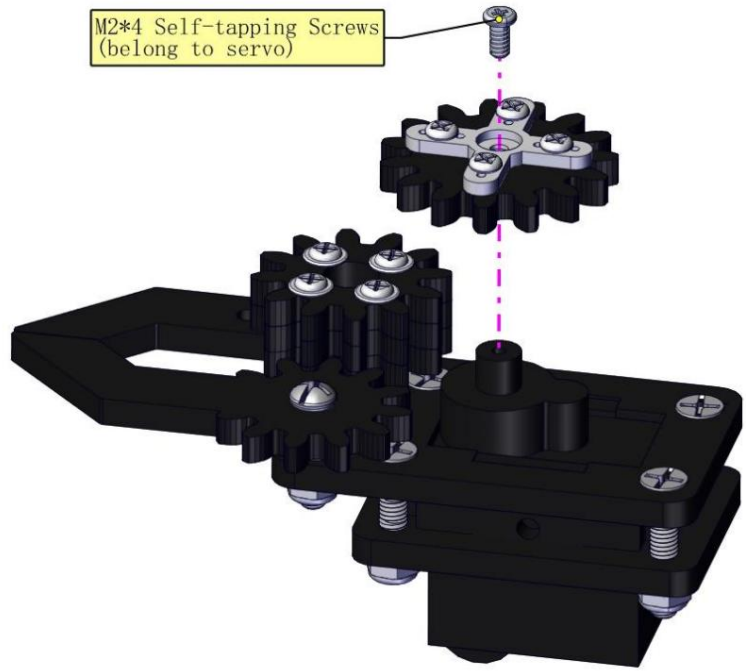

ongeldige lus()
{
    mijnservo.attach(9); // Verander pin om één voor één aan te passen
    mijnservo.write(0); //Hoek
    vertraging(1000);
}
********** **********

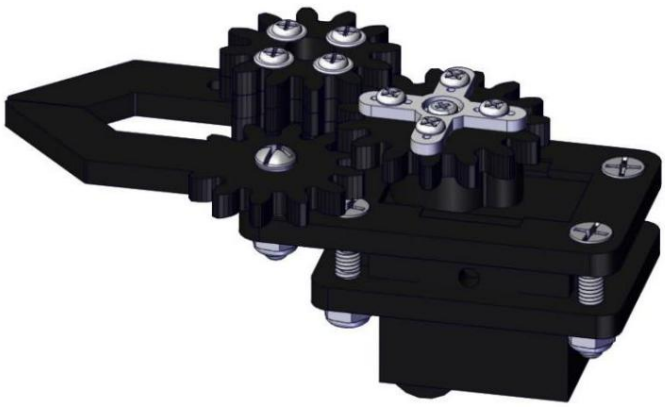
**********
```



Machine Translated by Google

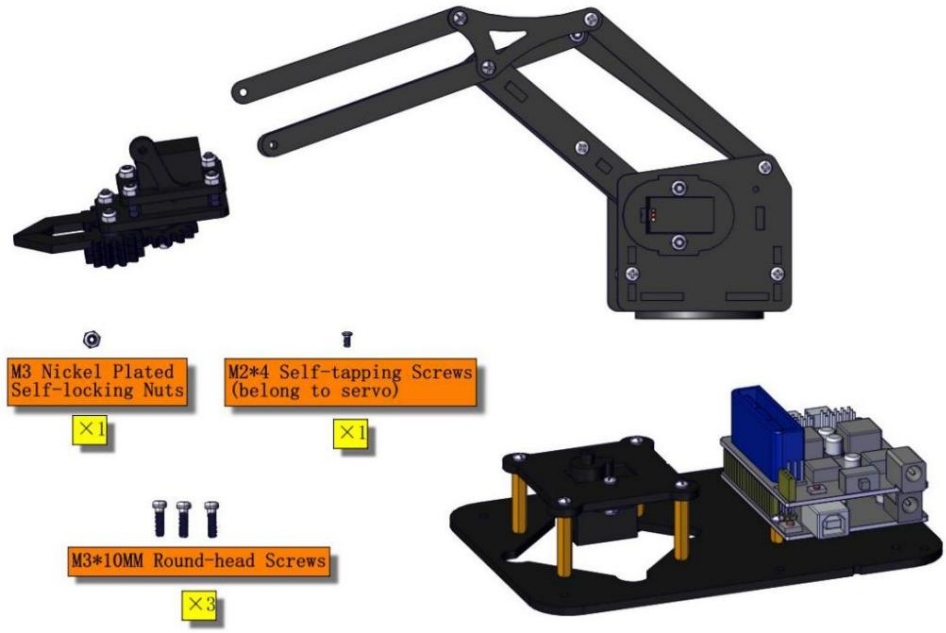
Tandwielen monteren:

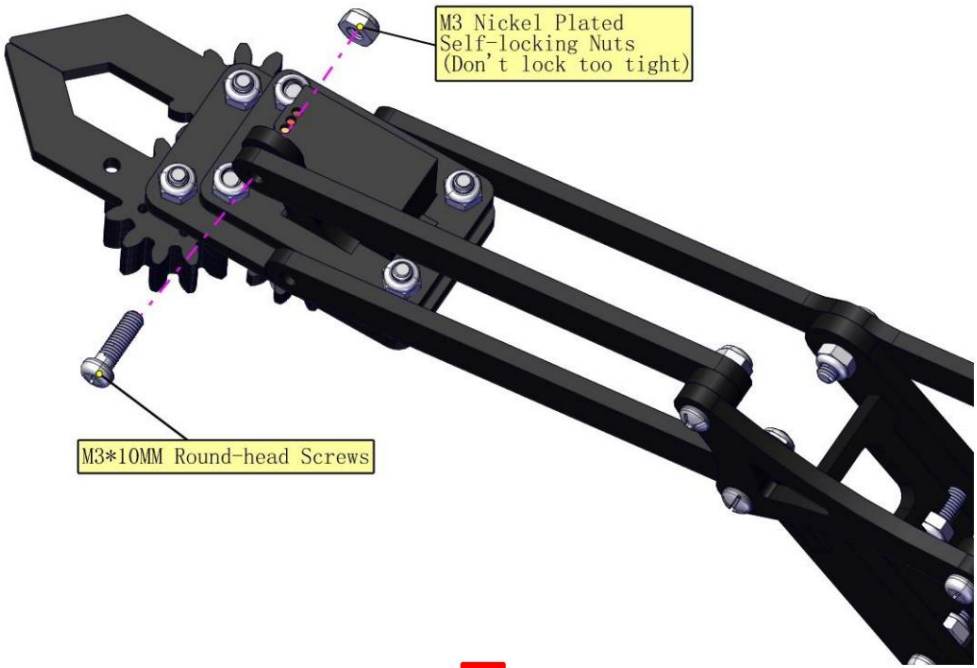




Machine Translated by Google

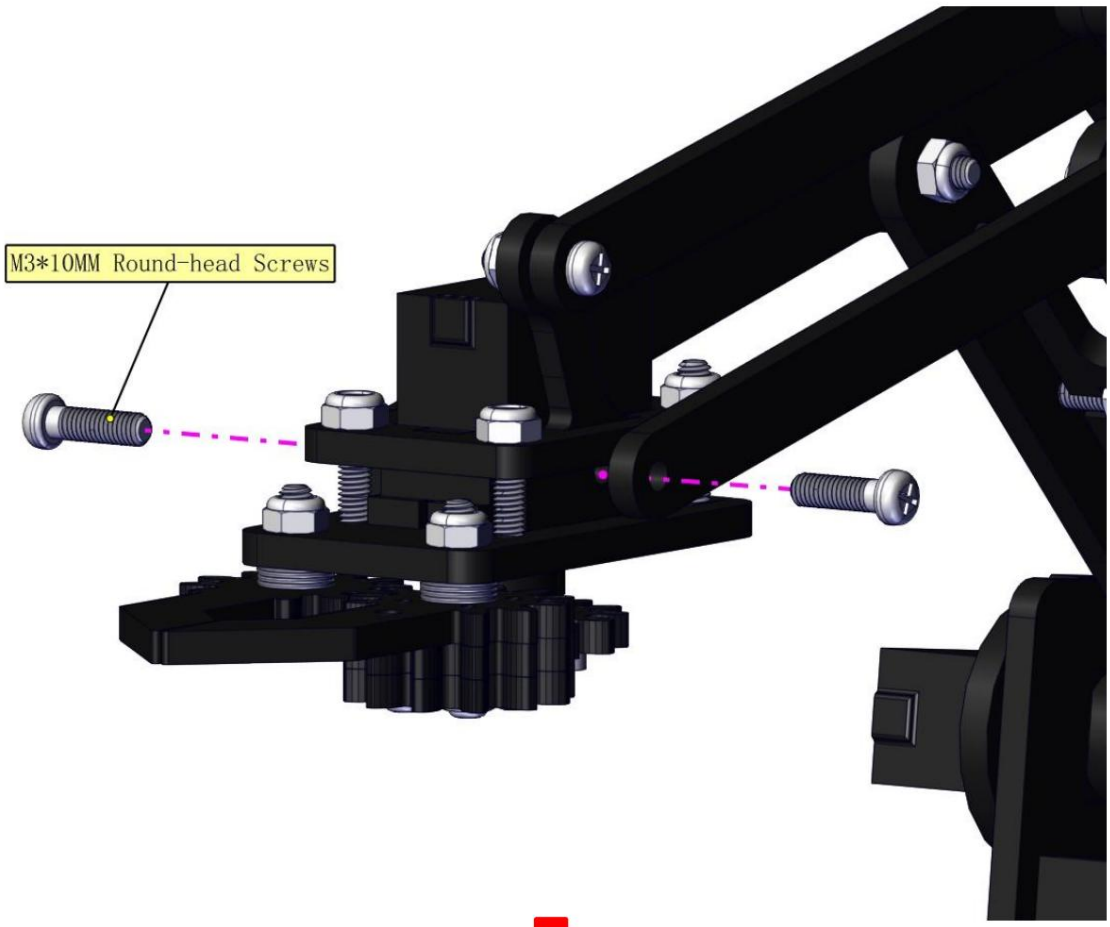
Benodigde componenten:





Machine Translated by Google





Machine Translated by Google



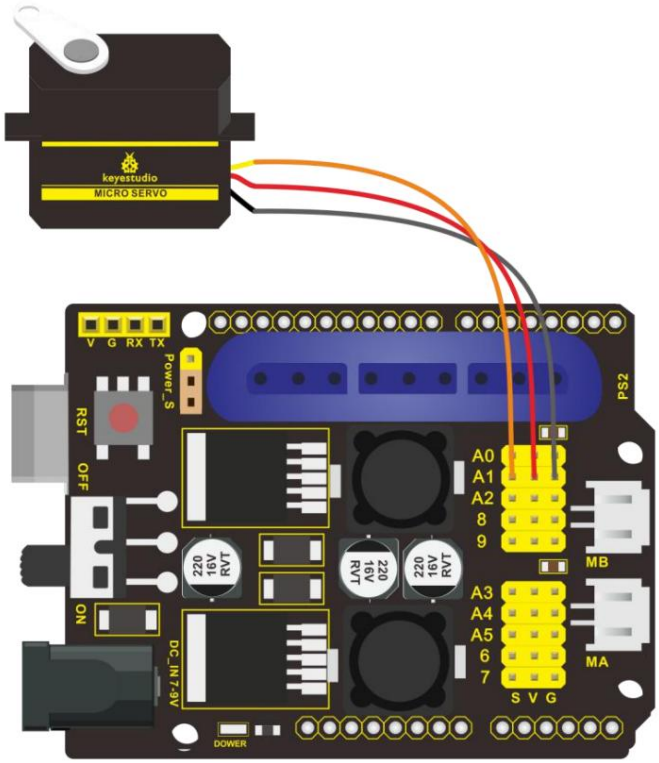
Initialiseer de servo

Bevestig deze linkerservo aan G, V en S (A1) van de TB6612FNG-driver

shield, upload de volgende code, steek de stekker in het stopcontact en druk op de rustknop op het V4.0-bord. Dan draait de linker servo naar

80°

Machine Translated by Google



Zet de servo op 80°:

********** **********

****

```
#include <Servo.h>

Servo mijnservo; // maak een servo-object om een servo te besturen



ongeldige instellingen ()

{

    Serieel.begin(9600);

    vertraging(1000);

}
```

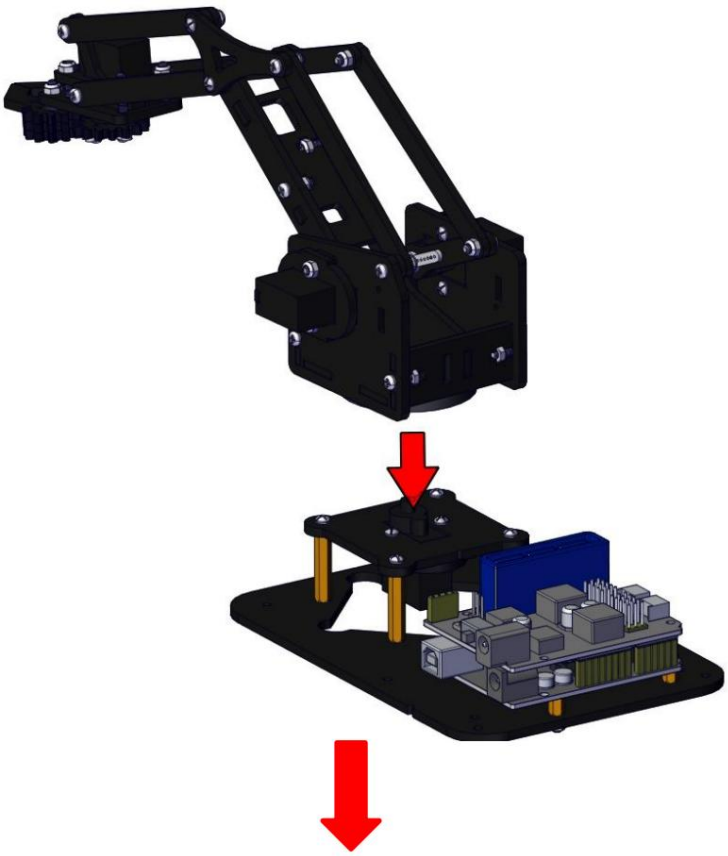
Machine Translated by Google


```
ongeldige lus()
{
    mijnservo.attach(A1); // Verander pin om één voor één aan te passen
    mijnservo.schrijven(80); //Hoek
    vertraging(1000);
}
********** **********
**********
```

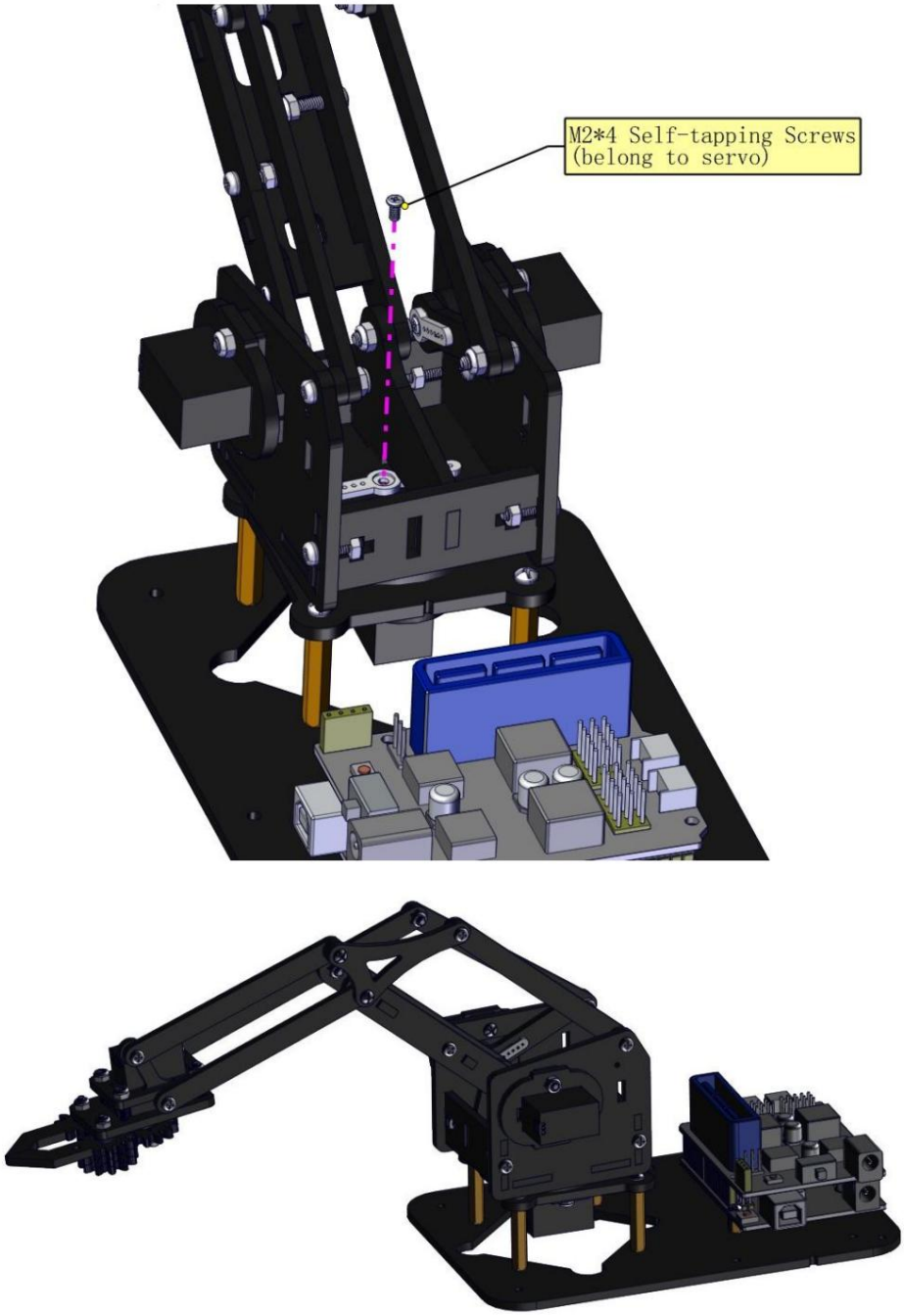


Machine Translated by Google

Installeer de robotarm:

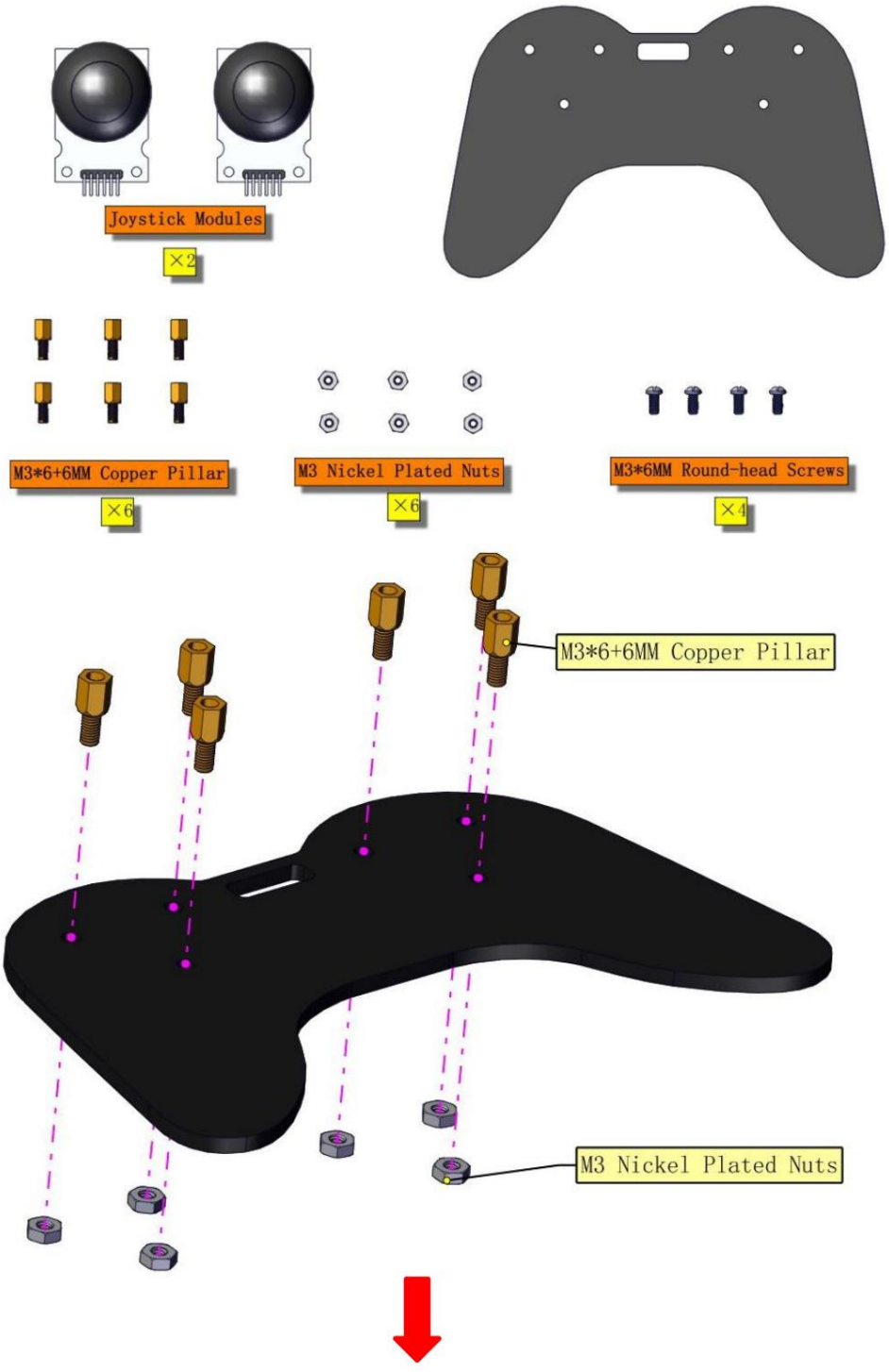


Machine Translated by Google

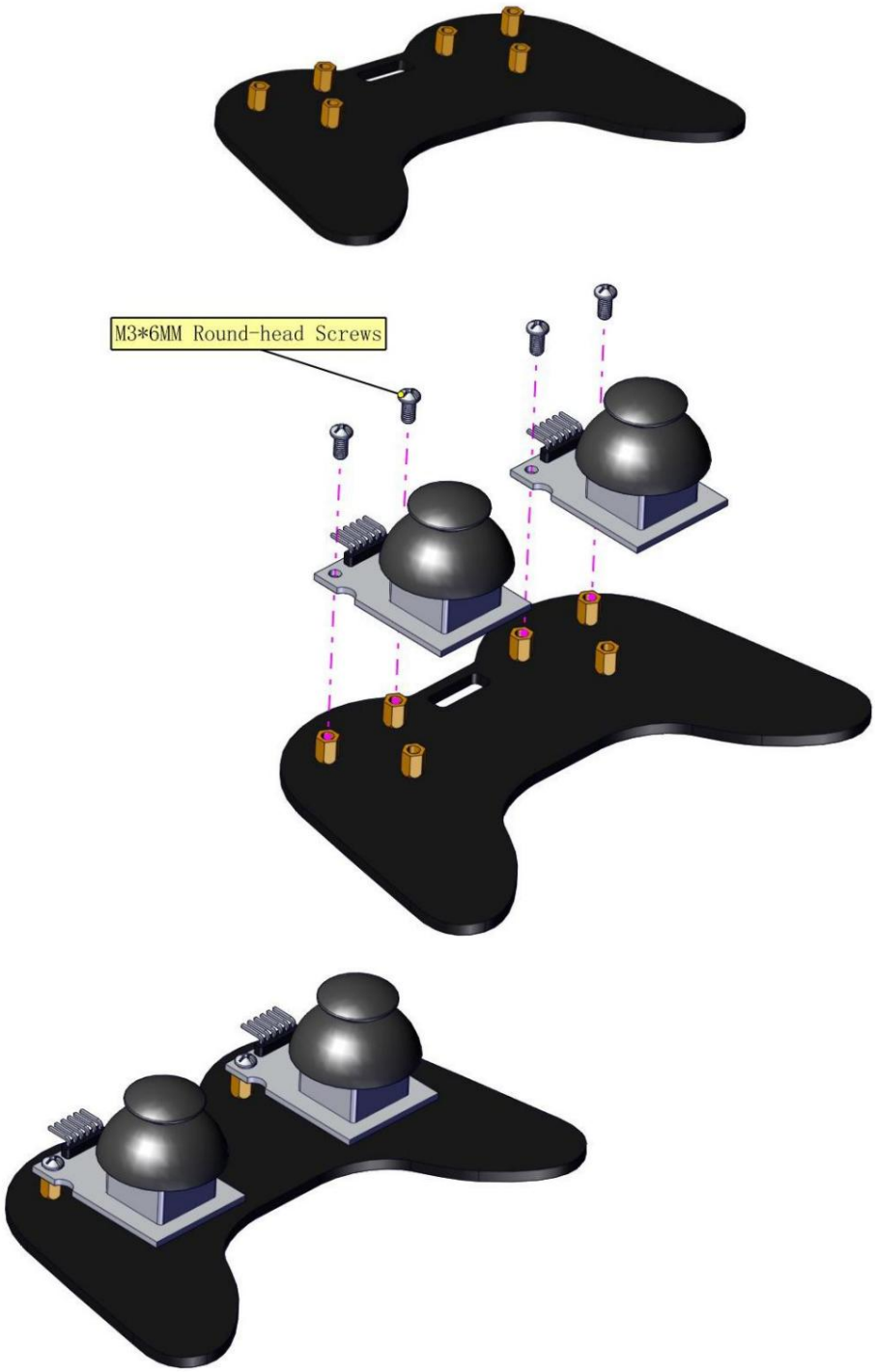


Machine Translated by Google

Monteer het bedieningsgedeelte



Machine Translated by Google

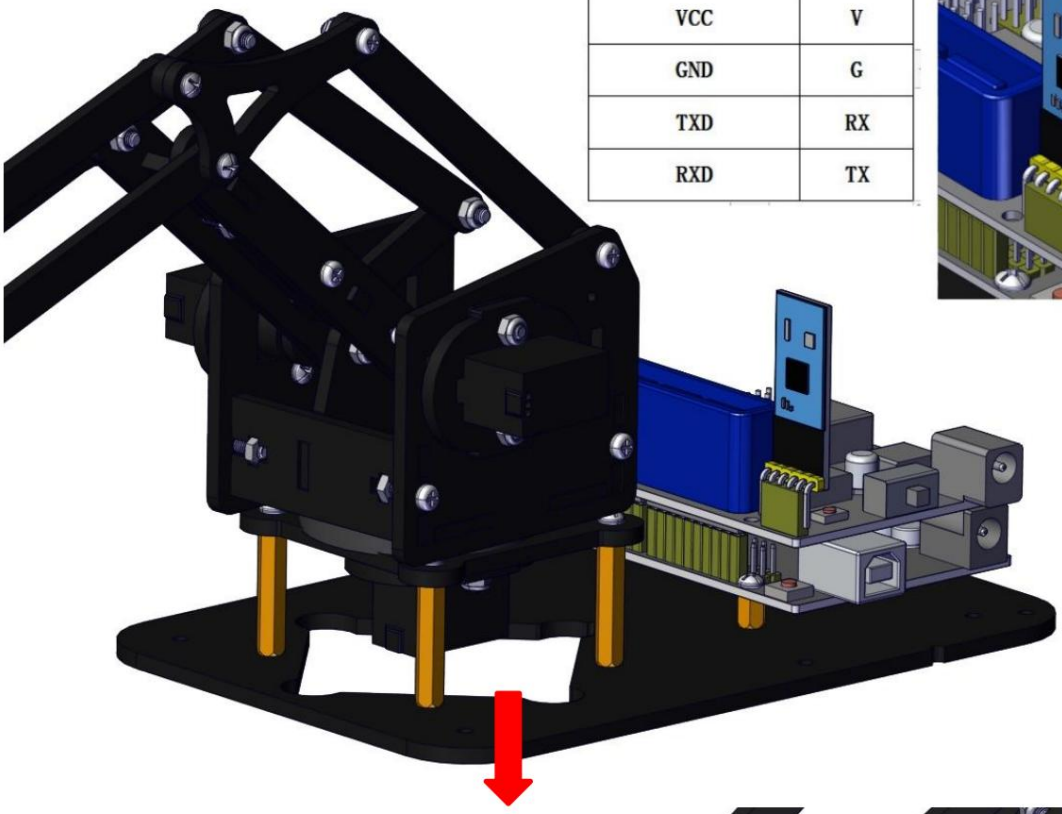


Machine Translated by Google

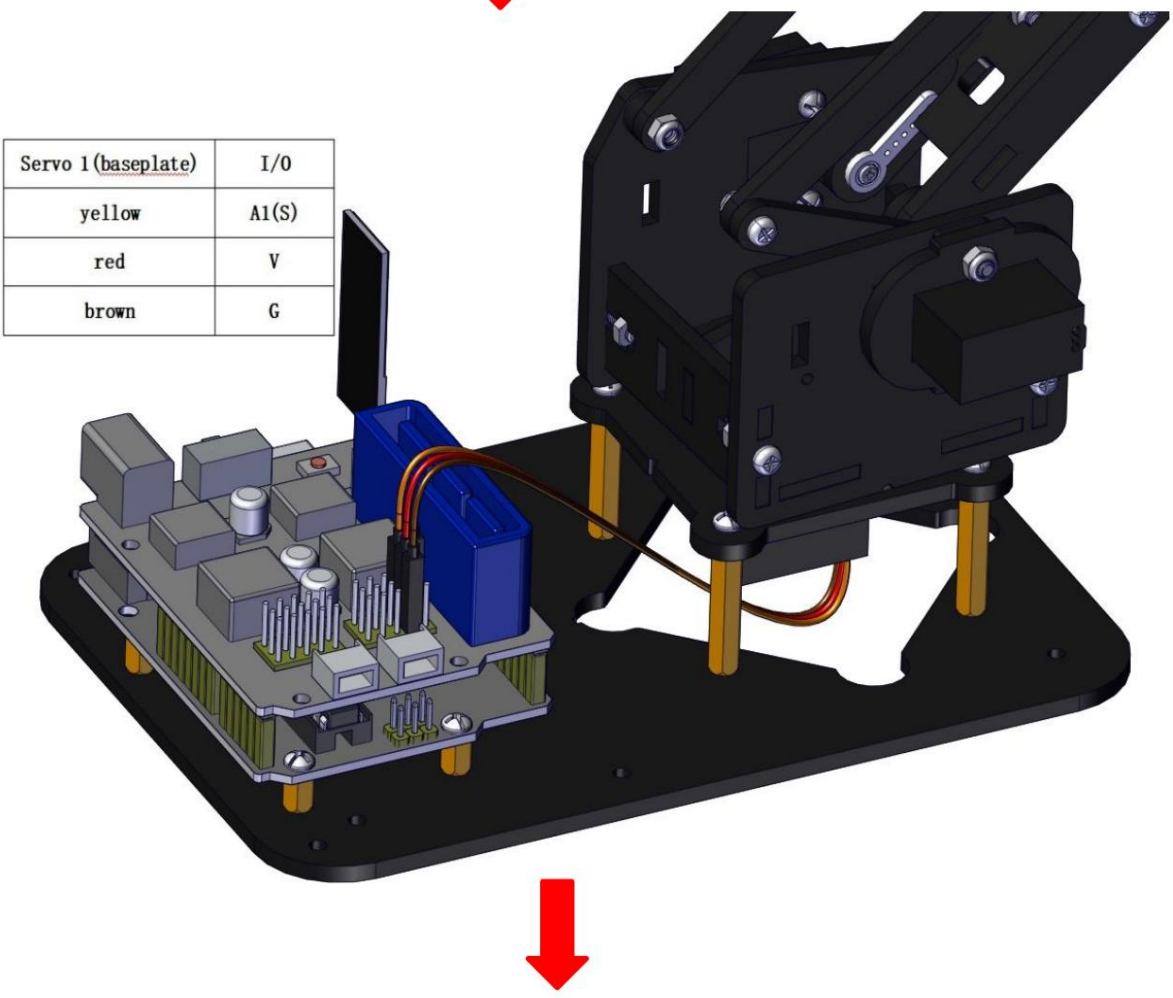
Bedrading gids

Bluetooth	I/O
VCC	V
GND	G
TXD	RX
RXD	TX



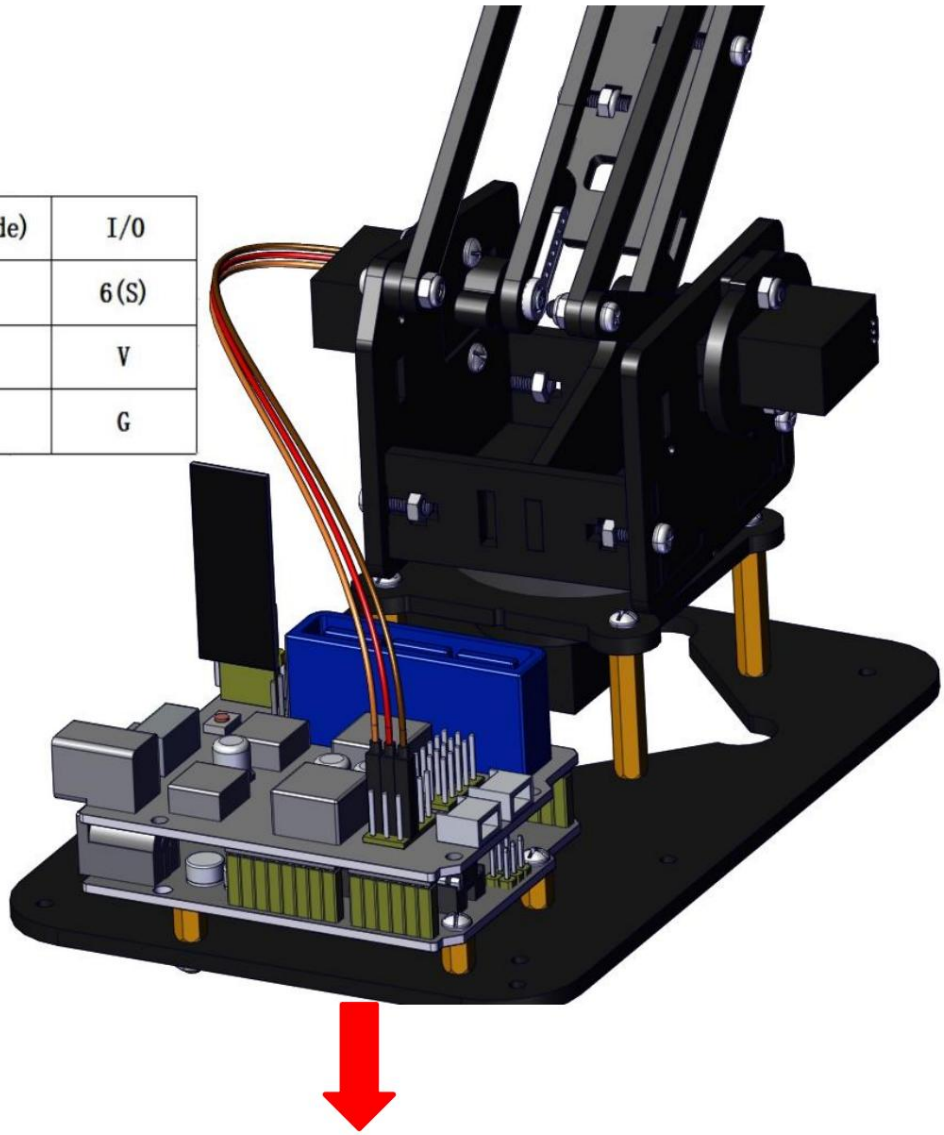


Servo 1(baseplate)	I/O
yellow	A1(S)
red	V
brown	G



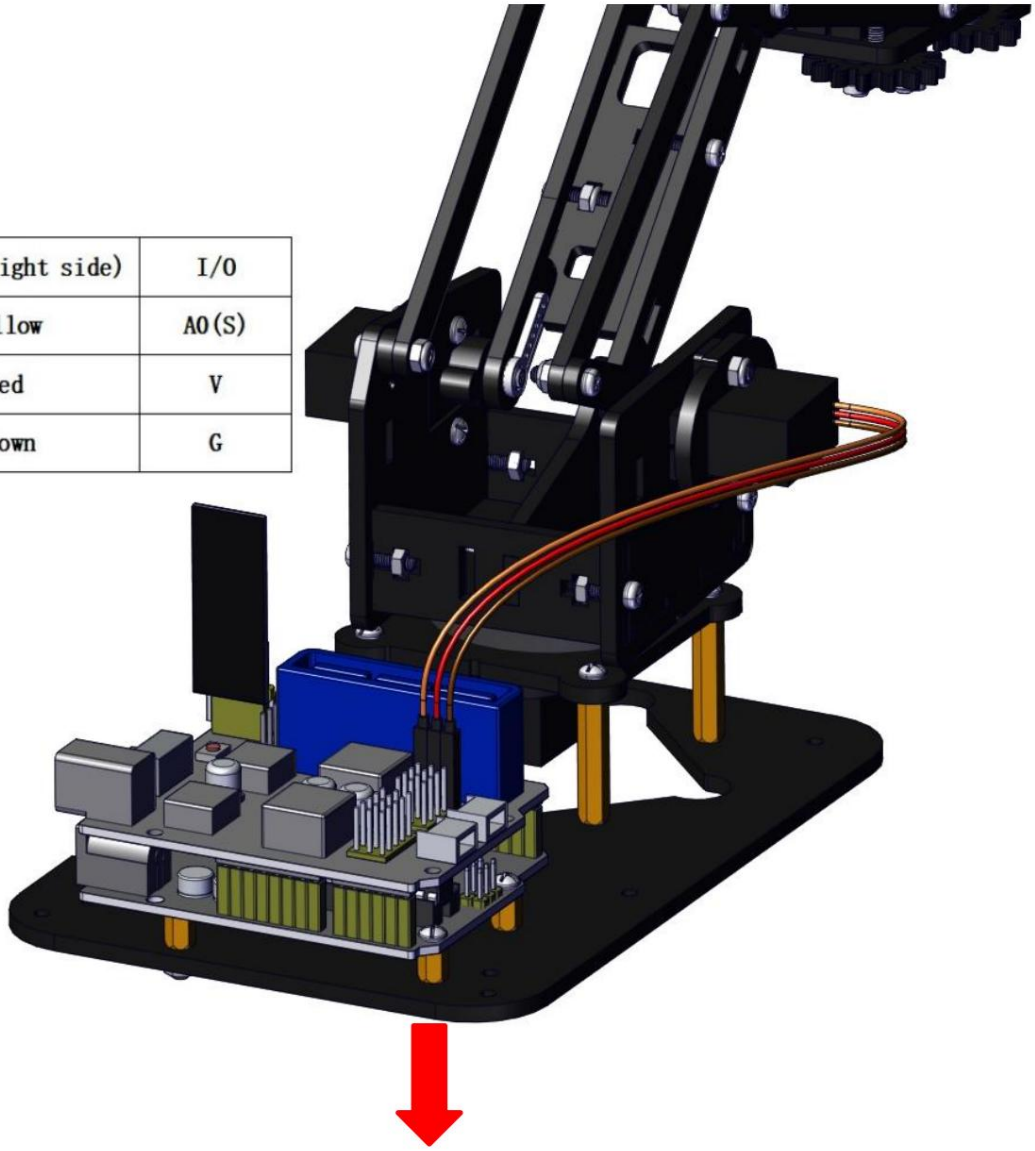
Machine Translated by Google

Servo 3(left side)	I/O
yellow	6(S)
red	V
brown	G



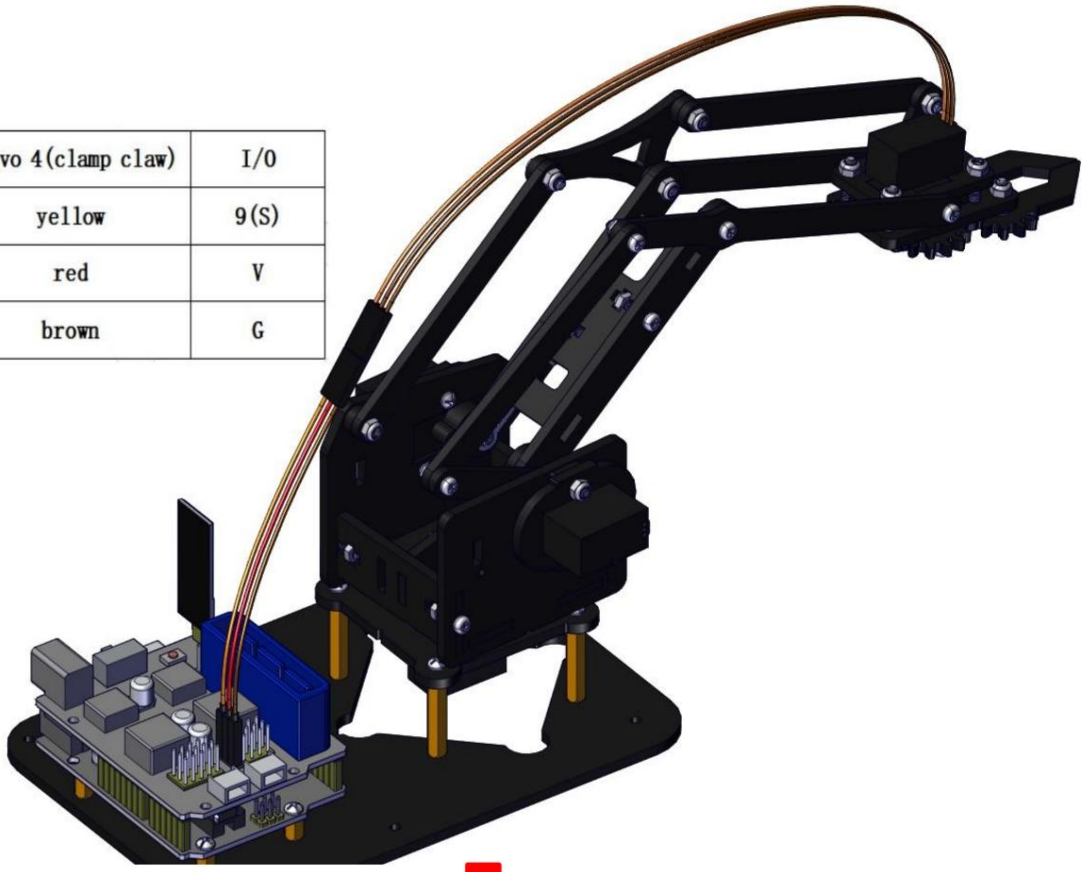
Machine Translated by Google

Servo 2(right side)	I/O
yellow	A0(S)
red	V
brown	G



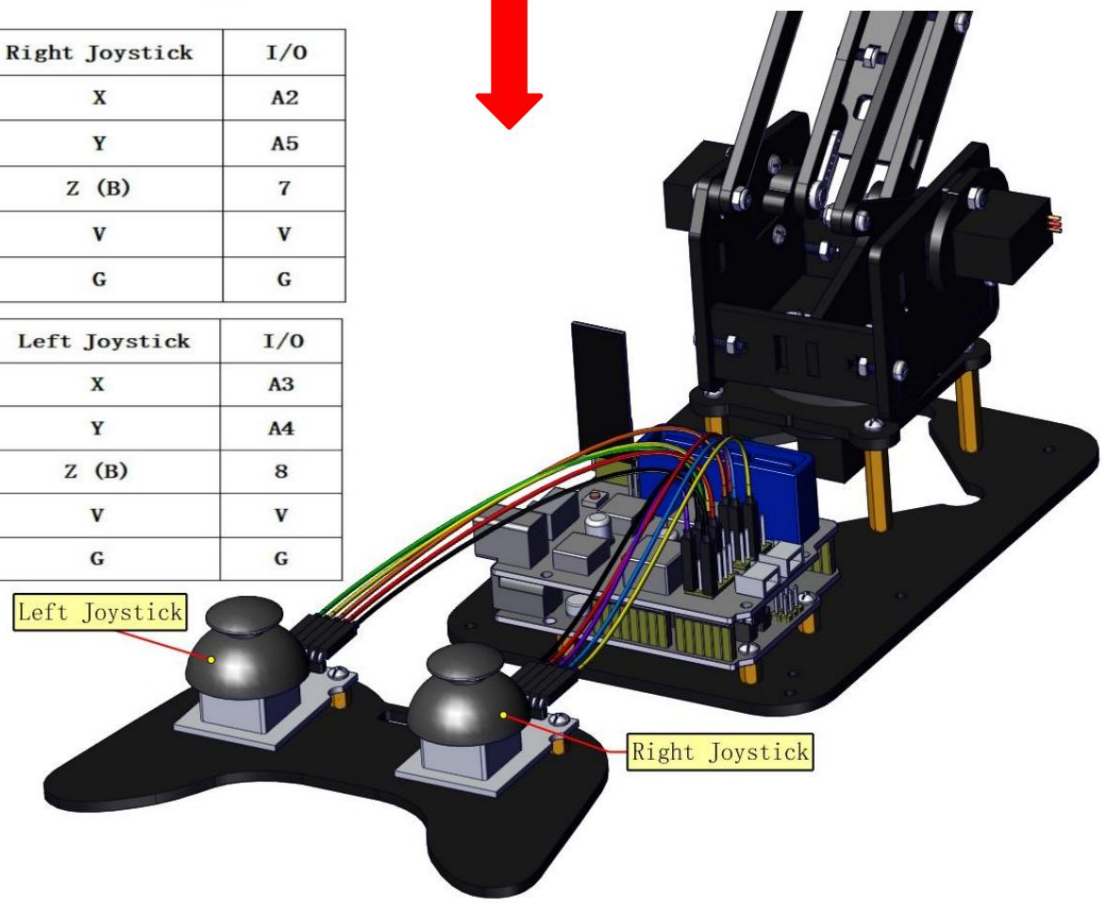
Machine Translated by Google

Servo 4(clamp claw)	I/O
yellow	9(S)
red	V
brown	G

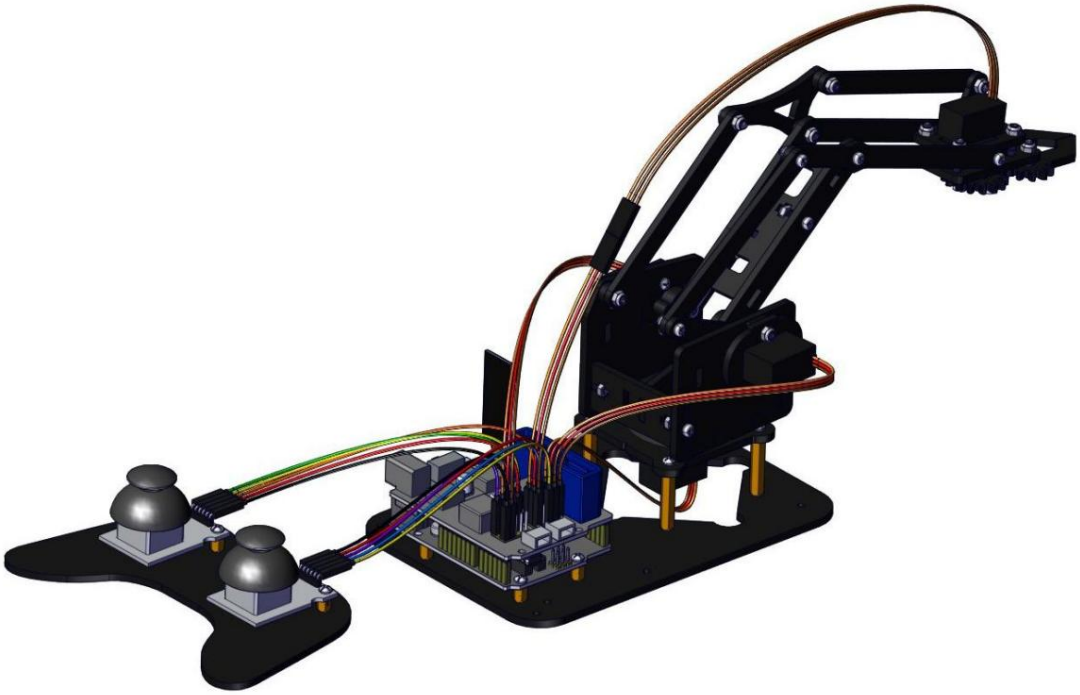


Right Joystick	I/O
X	A2
Y	A5
Z (B)	7
V	V
G	G

Left Joystick	I/O
X	A3
Y	A4
Z (B)	8
V	V
G	G



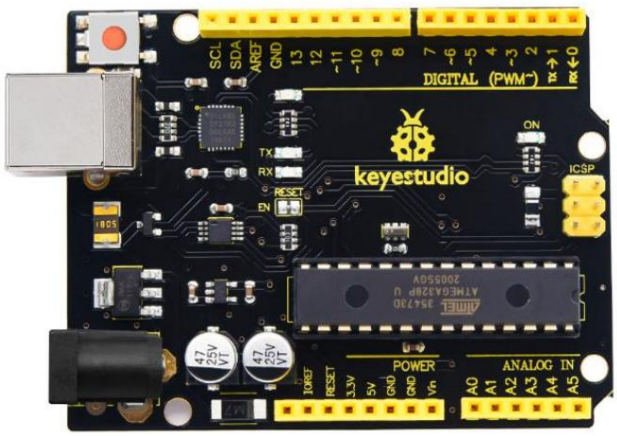
Machine Translated by Google



5. Robotarmprojecten

Project 1: Arduino IDE en stuurprogramma installeren

Keyestudio V4.0 ontwikkelbord

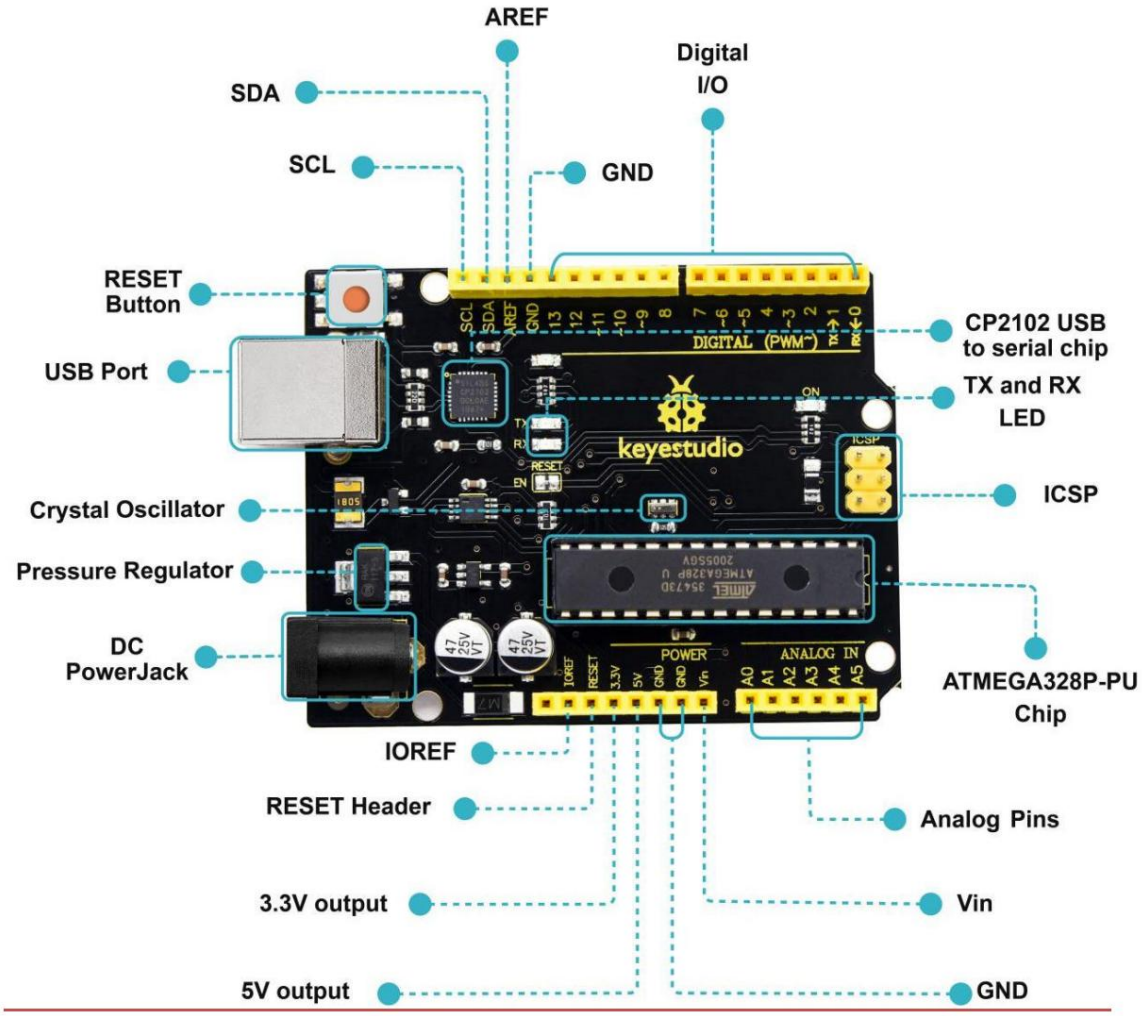


Machine Translated by Google

Keyestudio V4.0 ontwikkelbord is een Arduino uno-comp

atible board, dat is gebaseerd op ATmega328P MCU, en met een

cp2102-chip als een UART-naar-USB-converter.



Machine Translated by Google

1	**ICSP (In circuit Serieel Programma ng) Koptekst**	de AVR, een Arduino-microprogrammakop bestaande uit MOSI, MISO, SCK, RESET, VCC en GND. Het wordt vaak de SPI (seriële perifere interface) genoemd en kan worden beschouwd als een "uitbreiding" van de uitvoer. Slave de uitvoerapparaten in feite naar de SPI-bushost. Programmeer de firmware naar ATMEGA328P-PU.
2	**Activatie lampje Indicator**	De Arduino aanzetten, LED aan betekent dat uw printplaat is correct ingeschakeld. Als de LED uit is, is de verbinding verkeerd.

Machine Translated by Google

3	**digitale I/O**	Arduino MEGA heeft 14 digitale input/output pinnen (waarvan 6 gebruikt kunnen worden als PWM outputs). Deze pinnen kunnen geconfigureerd worden als digitale input pin om de logische waarde (0 of 1) te lezen. Of gebruikt als digitale uitgangspin om verschillende modules aan te sturen, zoals LED, relais, enz. Met behulp van pinMode(), digitaalschrijven(), En digitaal lezen() functies.
4	**GND**	GND
5	**AREF**	Referentiespanning (0-5V) voor analoge ingangen. Gebruikt met analogReference(). Configureert de gebruikte referentiespanning analoge ingang (dwz de waarde die wordt gebruikt als de top van het invoerbereik).
6	**SDA**	IIC-communicatiepen
7	**SCL**	IIC-communicatiepen
8	**RESETTEN Knop**	Je kunt je Arduino-bord resetten,
9	**D13-led**	Er is een ingebouwde LED aangedreven door digitale pin 13. Wanneer de pin een HOGE waarde heeft, is de LED dat ook aan, wanneer de pin LAAG is, is hij uit.

Machine Translated by Google

10	**USB Verbinding**	Arduino-bord kan worden gevoed via USB verbindingsstuk. Het enige dat u hoeft te doen, is de USB aansluiten poort naar pc met behulp van een USB-kabel.
11	**CP2102**	USB seriële chip, vertaal het USB-signaal van computer in serieel signaal
12	**TX-led**	Aan boord vind je het label: TX (verzenden) Wanneer het Arduino-bord communiceert via seriële poort, stuur het bericht, TX led flitst.
13	**RX-led**	Aan boord vindt u het label: RX(ontvangen) Wanneer het Arduino-bord communiceert via seriële poort, ontvang het bericht, RX led flitst.
14	**Kristal Oscillator**	Hoe berekent Arduino de tijd? door een kristaloscillator te gebruiken. Het nummer dat op de bovenkant van de Arduino-kristal is 16.000H9H. Dat vertelt het ons de frequentie is 16.000.000 Hertz of 16 MHz.

Machine Translated by Google

15	**Spanning Regelaar**	Om de spanning te regelen die aan het Arduino-bord wordt geleverd, en om de gelijkspanning die door de processor en andere componenten wordt gebruikt, te stabiliseren. Converteer een externe input DC7-12V spanning in DC 5V, schakel dan DC 5V naar de bewerker en andere componenten.
16	**gelijkstroom Jack**	Arduino board kan geleverd worden met een externe voeding DC7-12V van de gelijkstroom krik.
17	**microcontroller**	Elk Arduino-bord heeft zijn eigen microcontroller. Je kunt het beschouwen als het brein van je board. De belangrijkste IC (geïntegreerde schakeling) op de Arduino is iets anders dan het paneel paar. Microcontrollers zijn meestal van ATMEL. Voordat u een nieuw programma laadt de Arduino IDE, moet je weten wat IC is op je bord. Deze informatie kan zijn gecontroleerd bovenaan IC.

Machine Translated by Google

18	**IOREF**	Deze pin op het bord zorgt voor de spanning referentie waarmee de microcontroller werkt. Een correct geconfigureerd schild kan lees de IOREF-pinspanning en selecteer de geschikte stroombron of activeringsspanning vertalers op de uitvoer om mee te werken de 5V of 3.3V.
19	**RESETTEN** **Koptekst**	Sluit een externe knop aan om de bord. De functie is hetzelfde als resetten knop.
20	**Powerpin** **3V3**	Een 3,3 volt voeding gegenereerd door de boordcomputer regelaar. Maximale stroomafname is 50 mA.
21	**Powerpin** **5V**	Biedt 5V uitgangsspanning
22	**Vin**	U kunt een externe voedingsingang leveren DC7-12V via deze pin naar Arduino-bord.
23	**analoge pinnen**	Onboard heeft 6 analoge ingangen, gelabeld A0 to A5.

Machine Translated by Google

(1) Arduino IDE installeren

Wanneer we de besturingskaart krijgen, moeten we Arduino IDE downloaden en chauffeur voorop.

U kunt Arduino IDE downloaden van de officiële website:

https://www.arduino.cc/, klik op de **SOFTWARE** tijdens het bladeren klik op "DOWNLOADS" om de downloadpagina te openen, zoals weergegeven onderstaand:



Machine Translated by Google

Er zijn twee versies van IDE voor WINDOWS-systemen, dat kan
kies tussen de Installer (.exe) en de Zip-pakketten. Wij
stel voor dat u de eerste gebruikt die alles direct installeert
je moet de Arduino Software (IDE) gebruiken, inclusief de
chauffeurs. Bij het Zip-pakket moet u de stuurprogramma's installeren
handmatig. Het zip-bestand is ook handig als u een
draagbare installatie.

(2) Stuurprogramma van V4.0-kaart installeren

Laten we de driver van keyestudio V4.0 board installeren. De USB-TTL
chip op V4.0-kaart gebruikt seriële chip CP2102. De bestuurder
programma van deze chip is opgenomen in Arduino 1.8 versie en
hierboven, wat handig is. Inpluggen op USB-poort van board,
de computer kan de hardware automatisch herkennen

Installeer het stuurprogramma van CP2102.

als de versie van je Arduino niet hoger is dan versie 1.8, dan kan dat
download de driver van CP2102:

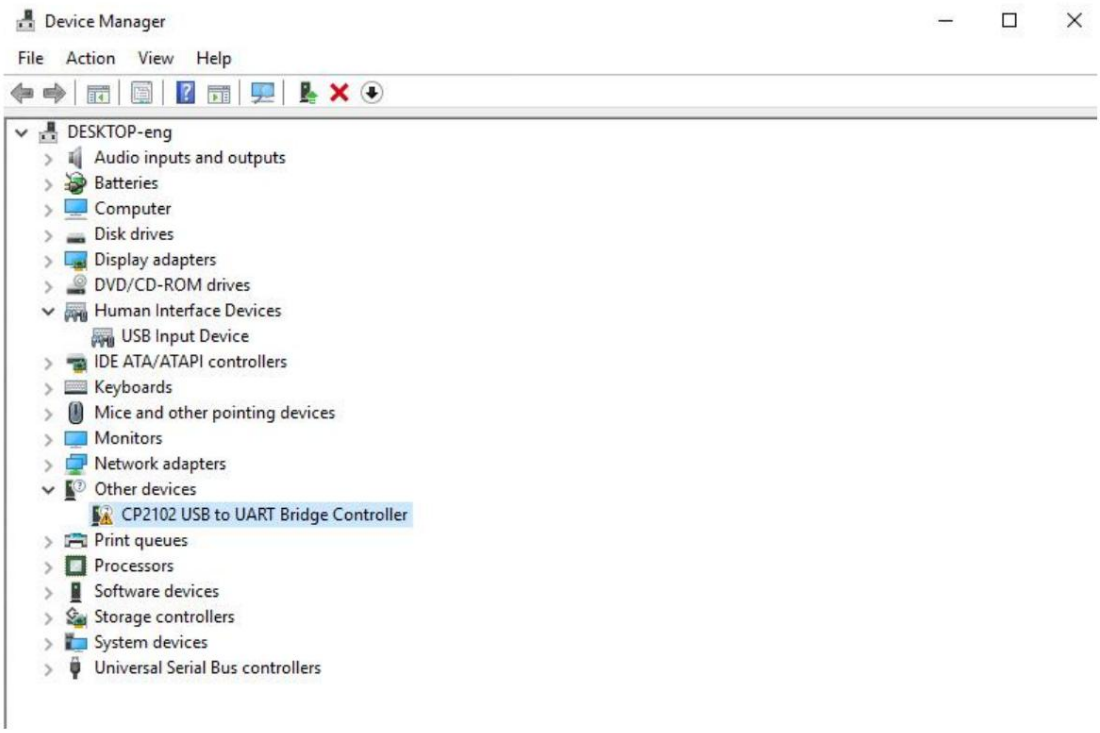
https://www.silabs.com/products/development-tools/softwar
e/usb-naar-uart-bridge-vcp-stuurprogramma's

Machine Translated by Google

Als u het niet succesvol installeert, open dan Apparaatbeheer

van computers. Klik met de rechtermuisknop op Computer ----- Eigenschappen ----- Apparaat

Manager

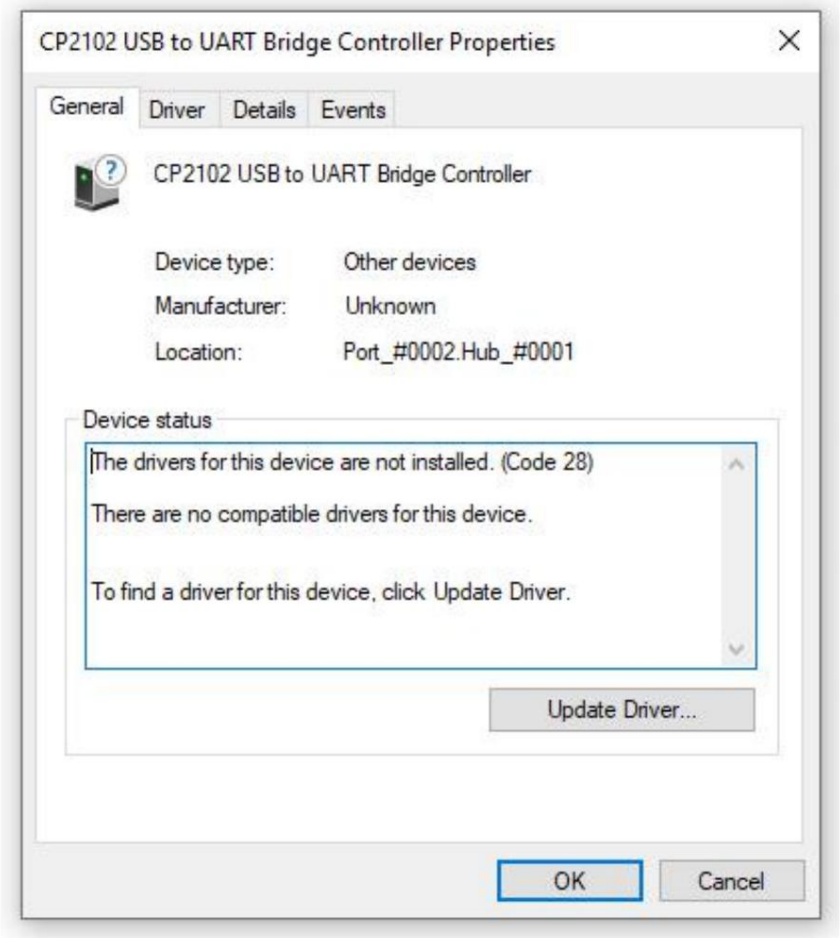


Het gele uitroepteken op de pagina duidt op een

mislukte installatie en u moet dubbelklikken op het

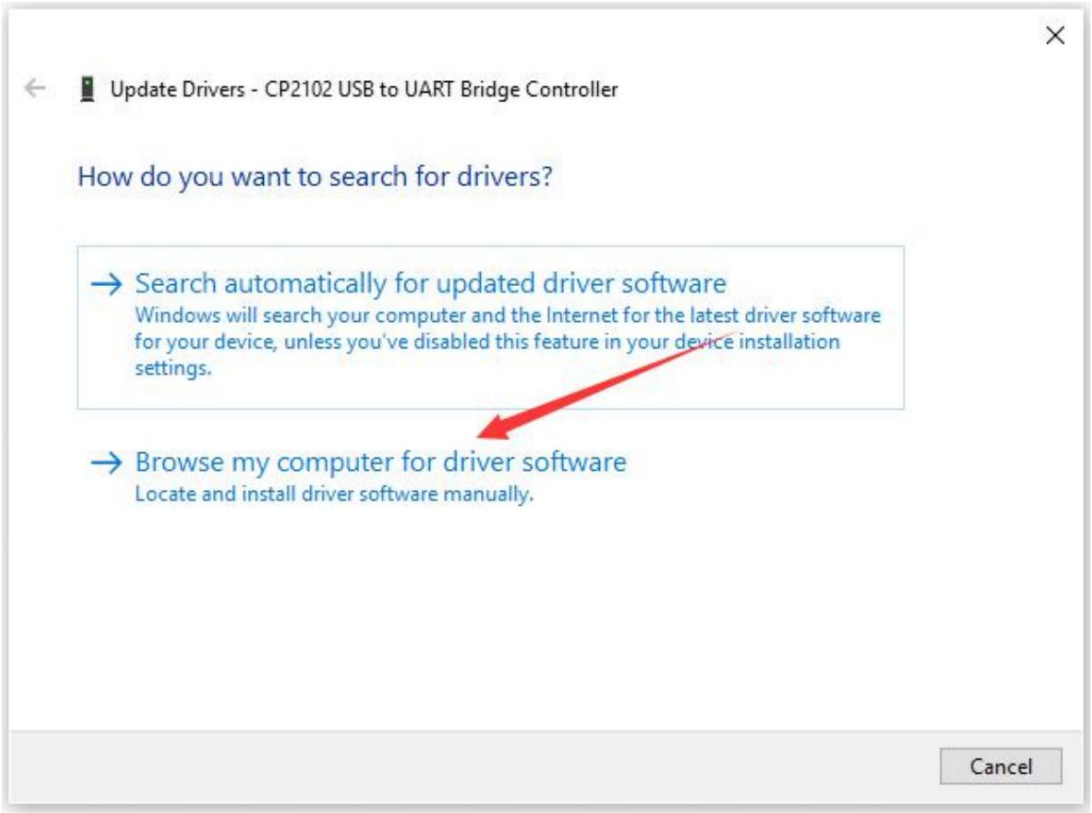
hardware en werk het stuurprogramma bij.

Machine Translated by Google



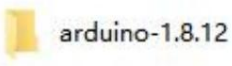
Klik op "OK" om naar de volgende pagina te gaan. Klik op "bladeren in mijn computer voor bijgewerkte stuurprogrammasoftware”

Machine Translated by Google



Klik op "Bladeren", zoek het stuurprogramma van de CP2102 en klik op "Volgende".

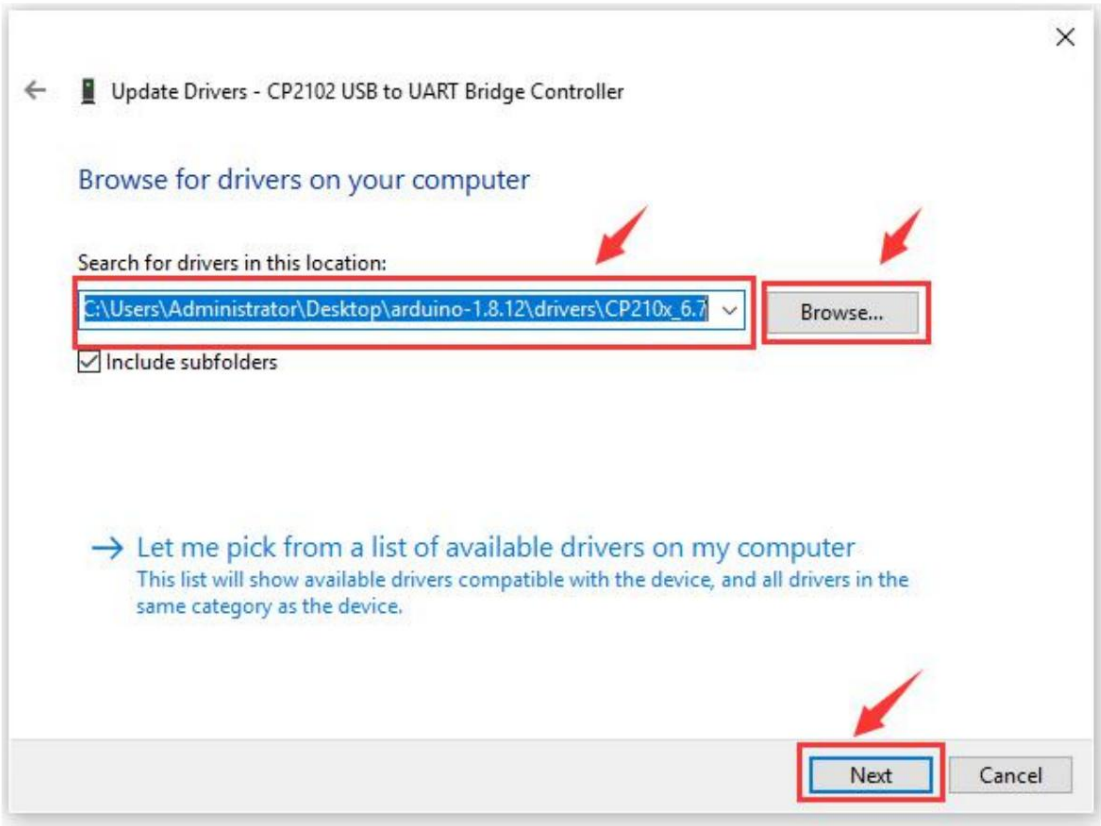
Er is een map DRIVERS in het geïnstalleerde Arduino-softwarepakket

ÿ 

ÿ, open de drivermap en controleer de driver van

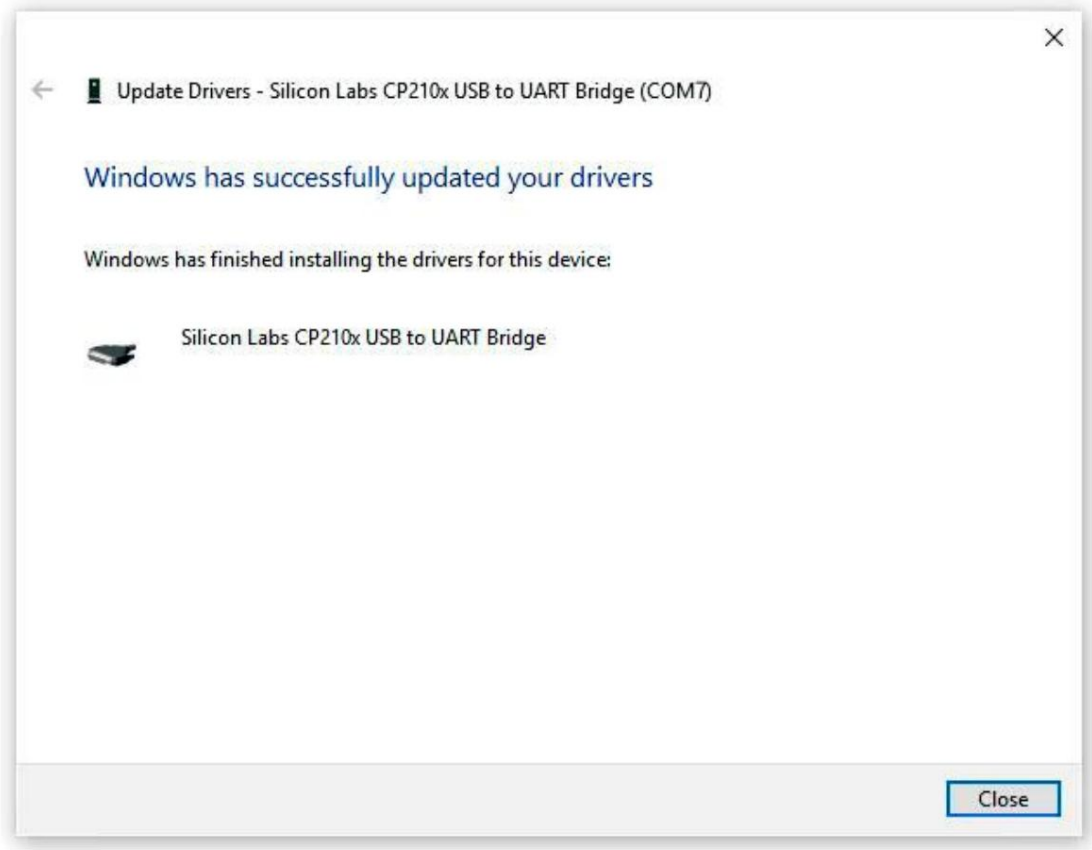
Chips uit de CP210X-serie.

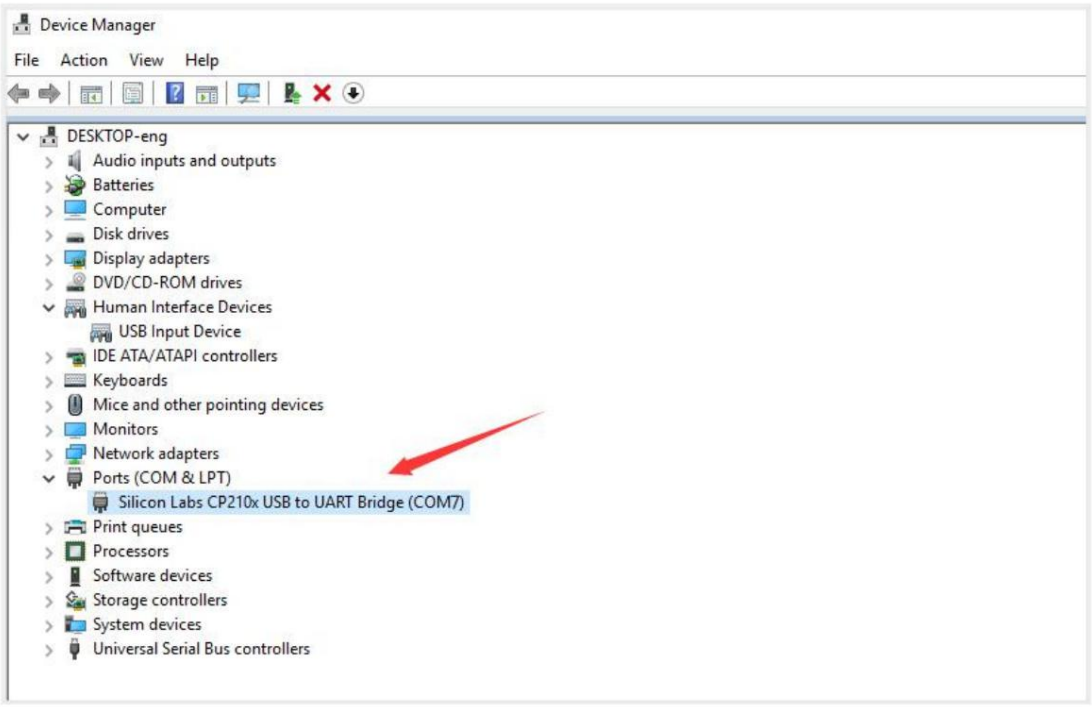
Machine Translated by Google



Bij het openen van Apparaatbeheer vinden we het geel uitroepteken verdwijnen. Het stuurprogramma van CP2102 is geïnstalleerd met succes.

Machine Translated by Google



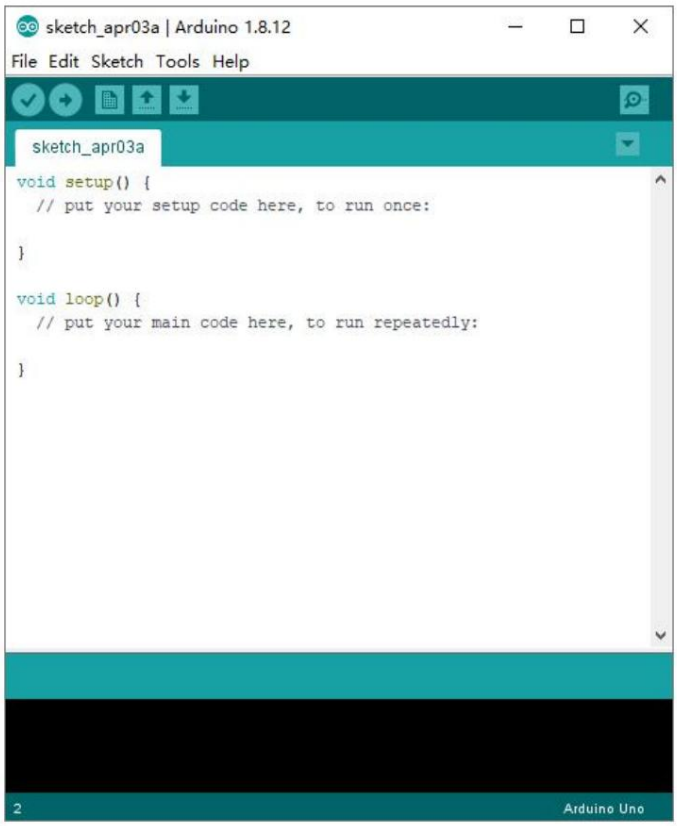


(3) Arduino IDE-instelling

Machine Translated by Google

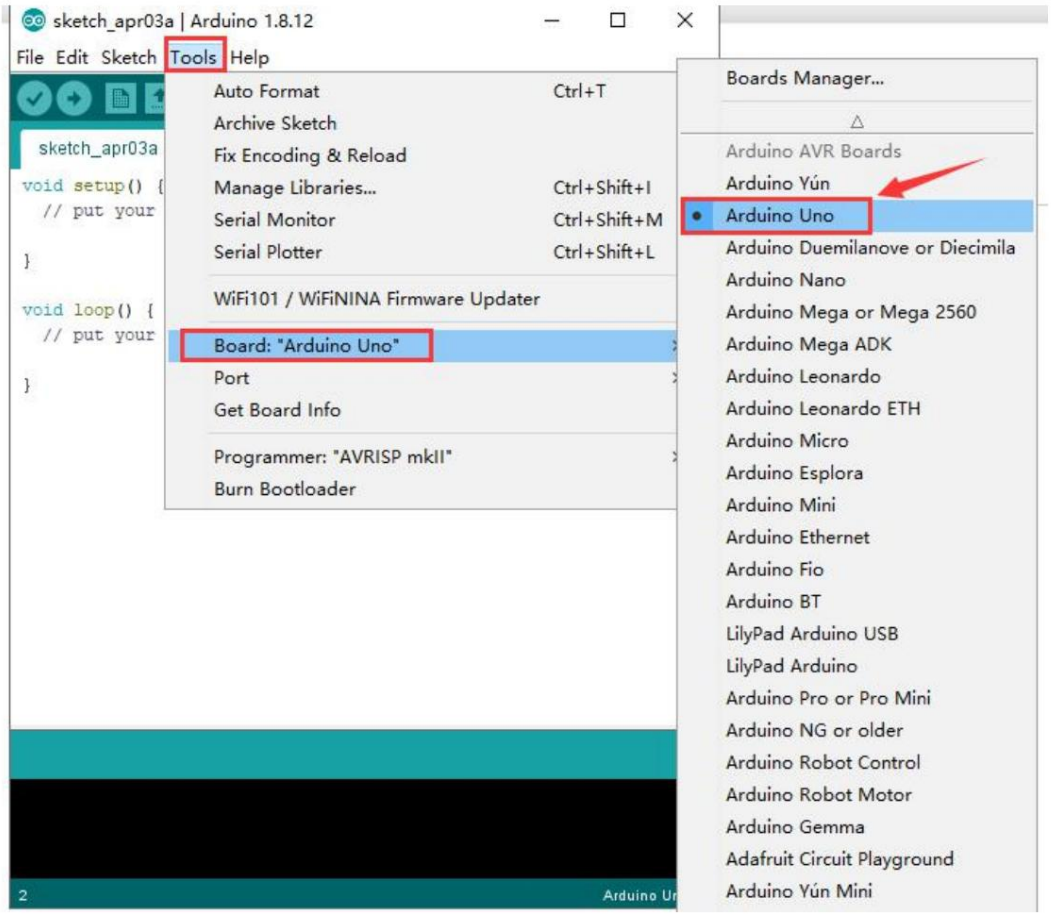
Klik 

icoon, en open Arduino IDE.



Wanneer u de schets naar het bord downloadt, moet u selecteren de juiste naam van het Arduino-bord dat overeenkomt met het bord aangesloten op uw computer. Zoals hieronder getoond;

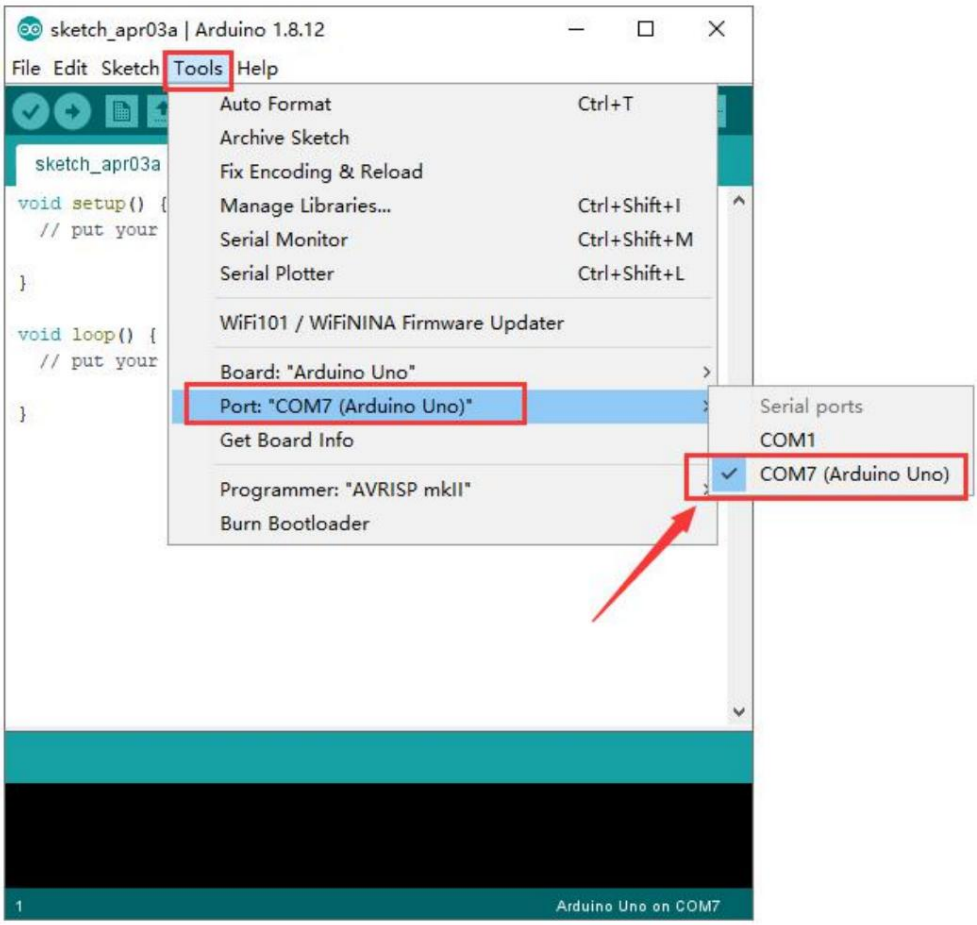
Machine Translated by Google



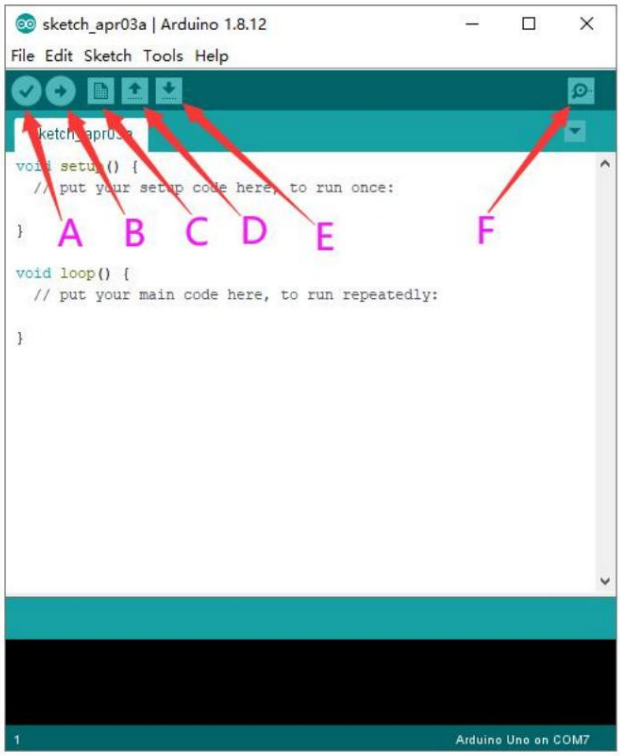
Selecteer vervolgens de juiste COM-poort (u ziet de bijbehorende

COM-poort nadat het stuurprogramma met succes is geïnstalleerd)

Machine Translated by Google



Machine Translated by Google



A- Gebruikt om te controleren of er compileerfouten zijn of

niet.

B- Wordt gebruikt om de schets naar uw Arduino-bord te uploaden.

C- Wordt gebruikt om een snelkoppelingsvenster van een nieuwe schets te maken.

D- Wordt gebruikt om direct een voorbeeldschets te openen.

E- Wordt gebruikt om de schets op te slaan.

F- Gebruikt om de seriële gegevens die van het bord zijn ontvangen naar de te sturen

seriële monitor.

Machine Translated by Google

Naam	0°	180°
Servo 1 (grondplaat)	Draaien naar de uiterst rechts	Draai naar de meest linkse
servo 2 (rechter zijde)	Tuimelaar verbonden met servo 2 trekt zich terug	uitrekken
servo 3 (linkerkant)	Tuimelaar verbonden met servo 3 rekt uit	nadeel
servo 4 (klem klauw)	dichtbij	open

Machine Translated by Google

Project 2: 4DOF-rotatie en pincontrole

1) Gewrichtsrotatie en servohoekinstellingen

2) Pincontrole

Naam	IO-pin
Servo 1 (grondplaat)	A1
Servo 2 (rechterkant)	A0
Servo 3 (linkerkant)	6
Servo 4 (klemklauw)	9
Rechterjoystick X	A2
Rechterjoystick Y	A5
Rechterjoystick Z (B)	7
Linkerjoystick X	A3
Linkerjoystick Y	A4

Machine Translated by Google

Linkerjoystick Z(B)	8
D1/DAT van PS2	12
D0/CMD van PS2	11
CE/SEL van PS2	10
CLK van PS2	13

Project 3: Bedien de robotarm met joysticks

3.1 Servobesturing

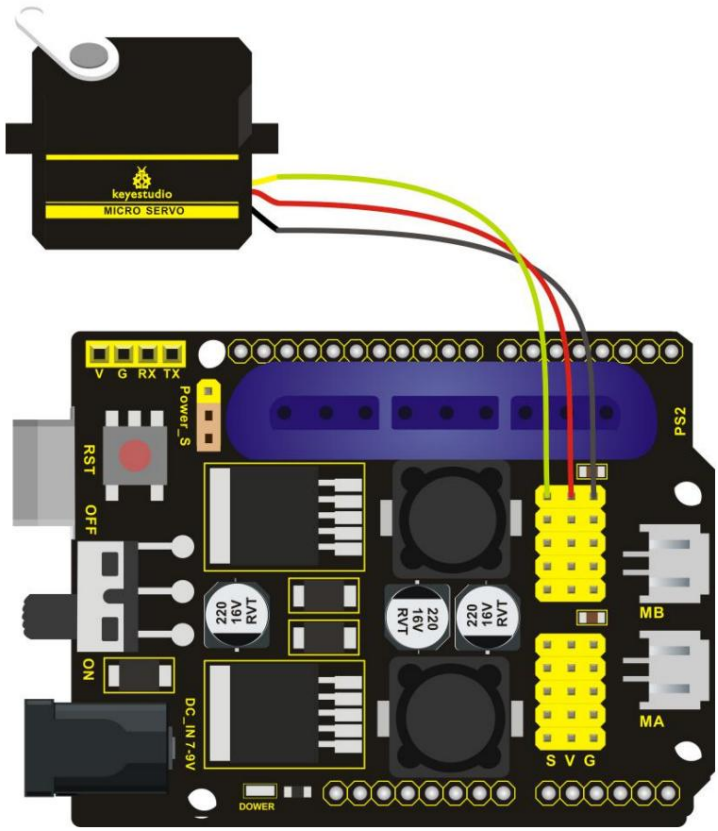
Beschrijving

In de vorige projecten hebben we de blokgolf en hoeken van ingesteld servo's.

Nu gebruiken we bibliotheken met servo's om de hoek van een servo te regelen. We hoeven alleen de servo-map in de bibliotheekmap te plaatsen waar de Arduino IDE-locatie is geïnstalleerd, open dan het Arduino IDE, het bibliotheekbestand wordt van kracht.

Machine Translated by Google

Verbindingsdiagram



Testcode 1:

********** **********

****

```
#include <Servo.h>

Servo mijnservo; // maak een servo-object om een servo te besturen



ongeldige instellingen ()

{

  Serieel.begin(9600);

  vertraging(1000);

}
```

Machine Translated by Google


```
ongeldige lus()
{
    mijnservo.attach(A0); // wijzig elke pin om aan te passen
    mijnservo.write(0); // hoekwaarde
    vertraging(1000);
}
```

****

****

****

****

Testresultaat:

Stapel het aandrijfschild op het V4.0-bord en sluit de servo aan motor, upload de code, steek de stekker in het stopcontact en druk op de resetknop. Dan draait de servo automatisch naar 0°.

ÿ Automatische beweging

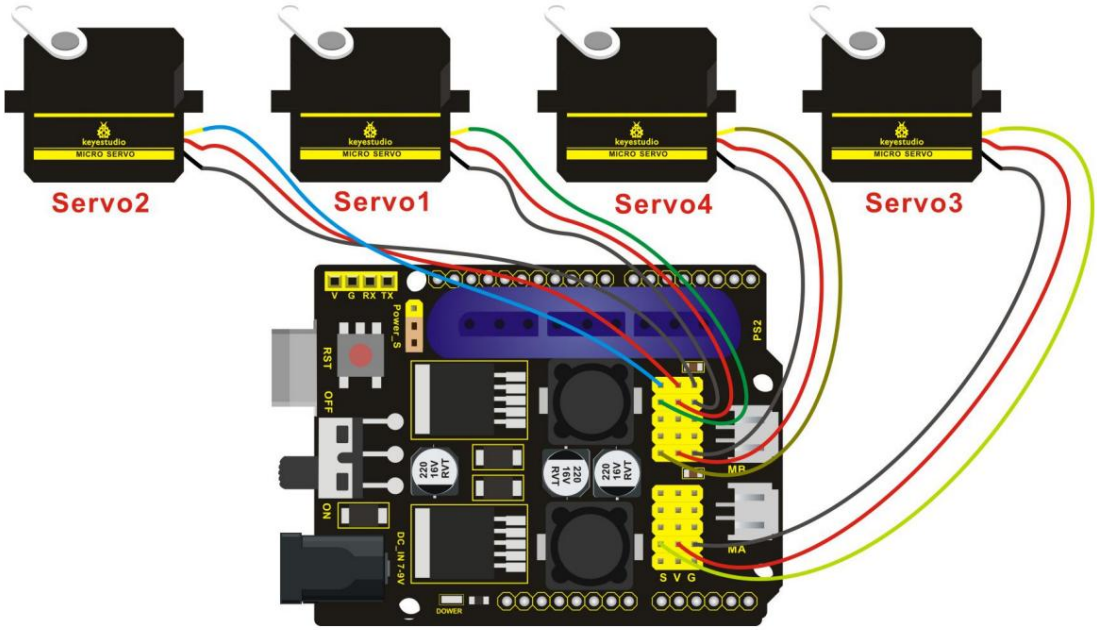
Beschrijving:

In het vorige gedeelte heb je geleerd hoe je de servohoek instelt. In feite hoeven we alleen maar continu de hoeken van 4 servo's te veranderen, waardoor de 4DOF-robotarm anders werkt

Machine Translated by Google

bewegingen.

aansluiting gids:



Testcode 2:

********** **********

****

```
#include <Servo.h>

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;
```

Machine Translated by Google


```
int pos1=80, pos2=60, pos3=130, pos4=0;



ongeldige instellingen ()

{

    mijnservo1.attach(A1); // bevestigt de servo op pin 9 aan het servo-object

    mijnservo2.attach(A0);

    mijnservo3.attach(6);

    mijnservo4.attach(9);



    mijnservo1.schrijven(pos1);

    vertraging(1000);

    mijnservo2.schrijven(pos2);

    mijnservo3.schrijven(pos3);

    mijnservo4.schrijven(pos4);

    vertraging(1500);

}



ongeldige lus()

{

    // sla rechtsaf

    voor(pos1;pos1>0;pos1--)

    {
```

Machine Translated by Google


```
      mijnservo1.schrijven(pos1);
      vertraging(5);              // vertraging 5ms (gebruikt om de servosnelheid aan te passen)
  }
  vertraging(1000);

  // open de klauw
voor(pos4;pos4<100;pos4++)
{
    mijnservo4.schrijven(pos4);
}
  vertraging(1000);

  // rechterservo draait naar 100 graden
  voor(pos2;pos2>50;pos2--)
  {
      mijnservo2.schrijven(pos2);
      vertraging(5);
  }
  // linkerservo draait naar 5 graden
  voor(pos3;pos3>50;pos3--)
  {
      mijnservo3.schrijven(pos3);
```

Machine Translated by Google


```
    vertraging(5);
}
vertraging(1500);

// sluit de klauw
voor(pos4;pos4>0;pos4--)
{
    mijnservo4.schrijven(pos4);
}
vertraging(1000);

// linkerservo draait naar 100 graden, tuimelaar gaat omhoog.
voor(pos3;pos3<120;pos3++)
{
    mijnservo3.schrijven(pos3);
    vertraging(5);
}
vertraging(1000);

// Draai naar links
voor(pos1;pos1<180;pos1++)
{
```

Machine Translated by Google


```
    mijnservo1.schrijven(pos1);
    vertraging(5);
}
vertraging(1000);

// Laat de arm zakken
voor(pos3;pos3>50;pos3--)
{
    mijnservo3.schrijven(pos3);
    vertraging(5);
}
vertraging(1000);

// open de klauw
voor(pos4;pos4<100;pos4++)
{
    mijnservo4.schrijven(pos4);
}
vertraging(1000);

// til de arm op
voor(pos3;pos3<120;pos3++)
```

Machine Translated by Google


```
        {
            mijnservo3.schrijven(pos3);
            vertraging(5);
        }
        vertraging(1000);

        // sluit de klauw
        voor(pos4;pos4>0;pos4--)
        {
            mijnservo4.schrijven(pos4);
        }
        vertraging(1000);
    }
```

****

****

Testresultaat:

Stapel het driver-schild op het V4.0-bord en sluit de servo aan

motor, upload de code goed, steek de stekker in het stopcontact en druk op de

reset knop. Dan zal de robotarm naar rechts draaien, strekken

uit de arm, laat zakken en schakel de klauw in; dan zal het zich terugtrekken, optillen, ,

draai naar links, strek uit, laat zakken en maak de klauw open.

Deze reeks acties zal continu zijn.

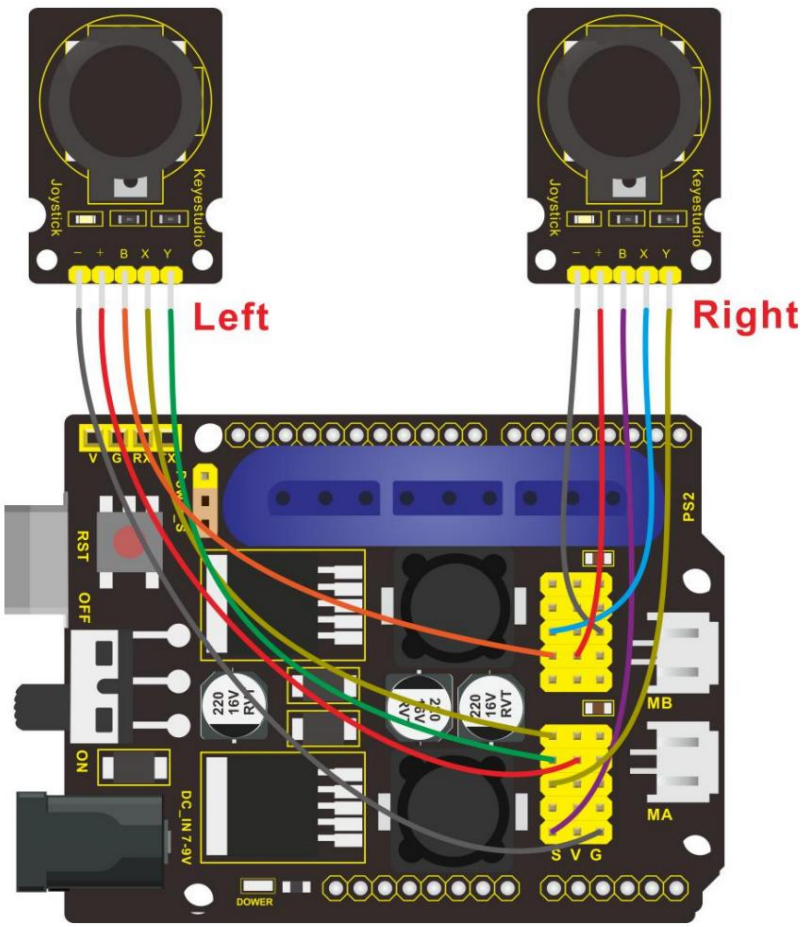
Machine Translated by Google

3.2 Lees de joystickwaarde af

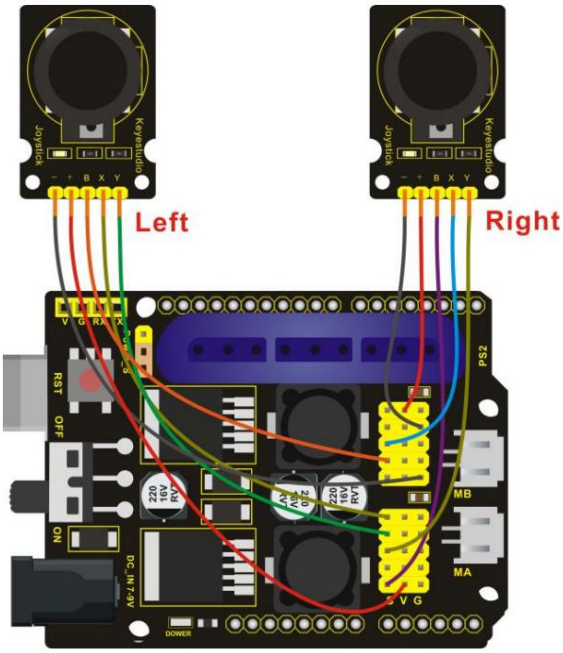
Beschrijving:

De pin X, Y van de sensor is voor analoge sensor, dus lees direct de gemeten analoge waarde af. Pin Z is een digitale knop, eerst moet de pin in de status Input worden gezet en vervolgens de gemeten waarde 1 (ingedrukt) of 0 (niet indrukken) lezen. Bekijk de waarde die op de seriële monitor is afgedrukt.

Verbindingsdiagram:



Machine Translated by Google



Testcode 3:

********** **********

****

const int rechts_X = A2; // definieer de juiste X-pin naar A2

const int rechts_Y = A5; // definieer de juiste Y-pin naar A5

const int right_key = 7; // definieer de juiste sleutelpin tot 7 ÿdat is de waarde Zÿ

const int linker_X = A3; // definieer de linker X-pin naar A3

const int links_Y = A4; // definieer de linker Y-pin naar A4

const int linker_sleutel = 8; // definieer de linkersleutelpin tot 8 ÿdat is de waarde Zÿ

ongeldige instellingen ()

{

pinMode (rechter_toets, INPUT); // stel de toets rechts/links in op INPUT

Machine Translated by Google


```
    pinMode (linker_key, INPUT);
    Serieel.begin(9600); // stel de baudrate in op 9600
}

ongeldige lus()
{
    int x1,y1,z1; // definieer de variabele, gebruikt om de joystickwaarde op te slaan die hij leest
    int x2,y2,z2;

    x1 = analoogRead(right_X); // lees de waarde van rechts X
    y1 = analoogRead(right_Y); // lees de waarde van rechts Y
    z1 = digitalRead(rechter_sleutel); //// lees de waarde van rechts Z

    x2 = analoogLezen(linker_X); // lees de waarde van linker X
    y2 = analoogLezen(linker_Y); // lees de waarde van linker Y
    z2 = digitalRead(linker_key); // lees de waarde van linker Z

    Serial.print("right_X = "); // print op de seriële monitor right_X =
    Serieel.println(x1, DEC); // print de waarde van rechts X en regelomloop uit
    Serial.print("right_Y = ");
    Serieel.println(y1, DEC);
    //Serial.print("right_key = ");
```

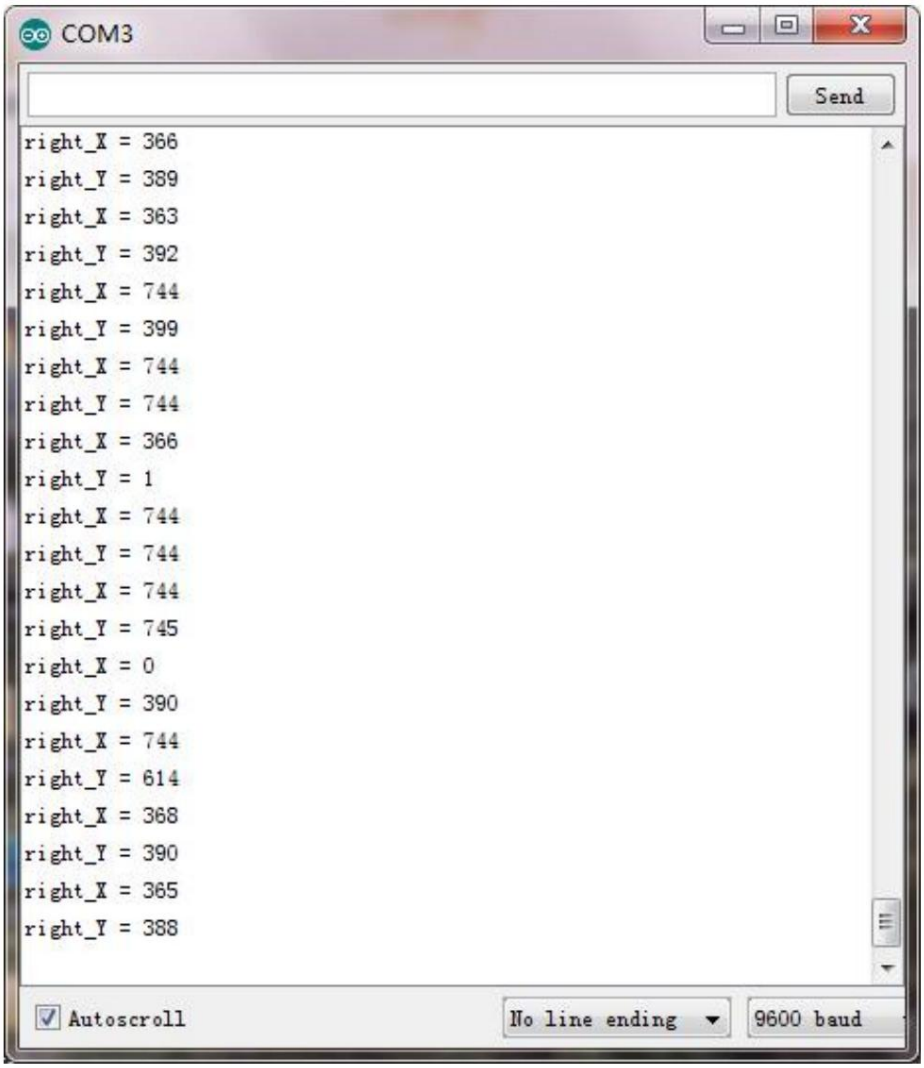
Machine Translated by Google


```
        //Serial.println(z1,DEC);
      // Serial.println("**********right**********");
        /*Serial.print("left_X = ");
        Serieel.println(x2, DEC);
        Serial.print("left_Y = ");
        Serieel.println(y2, DEC);
        Serial.print("left_key = ");
        Serieel.println(z2 ,DEC);
        Serial.println("**********links**********");*/
        vertraging(200);
    }
********************************************** ******************************
****************************************
```

Testresultaat:

Sluit het aan en upload de code goed. Sluit de V4.0 aan op de computer met behulp van een USB-kabel, open vervolgens de seriële monitor en stel de baudsnelheid in op 9600. U zou de analoge waarde van de rechter joystickpen X, Y moeten zien.

Machine Translated by Google



Machine Translated by Google

3.3 Bediening met dubbele joystick

Beschrijving: In het vorige gedeelte hebben we geïntroduceerd hoe u 4 servo's kunt gebruiken om de robotarm te besturen. Combineer vervolgens die twee experimenten. Gebruik twee joystickmodules om de 4DOF-robotarm te besturen en verschillende bewegingen te realiseren.

Stel eerst de opstarthouding in. De joystickbediening wordt weergegeven zoals in onderstaande tabel.

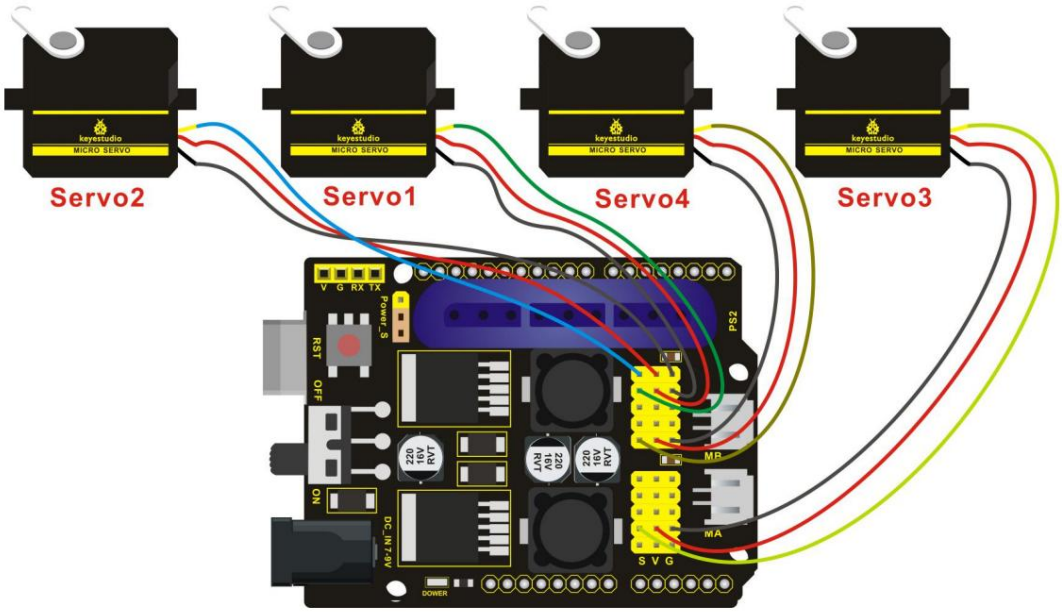
Rechts Joystick	**Servo**	**Links Joystick**	**Servo**
X1<50	Servo 1 geleidelijk reduceert tot 0° (druk rechts joystick naar de juist, de servo dat regelt de armrotatie draait rechts, en stopt bij 0°)	X2<50	Servo 4 vermindert geleidelijk tot 0° (duw de linkerjoystick naar rechts, de klauw is gesloten)

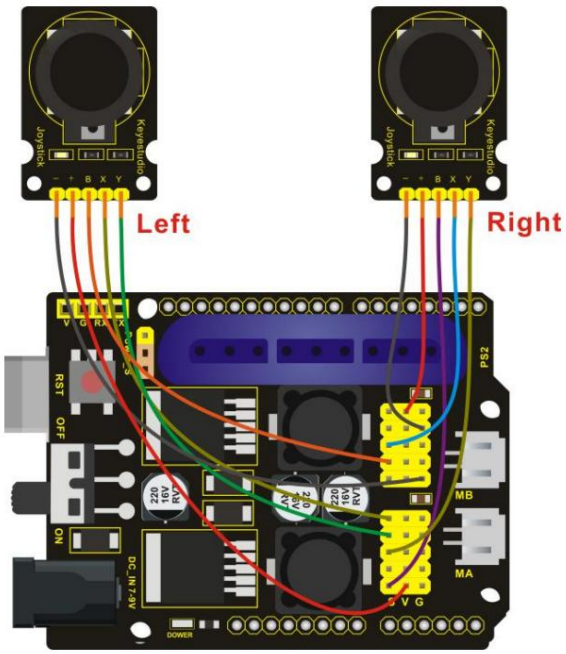
Machine Translated by Google

X1>1000	Servo 1 geleidelijk neemt toe tot 180° (druk rechts joystick naar links, de servo dat controleert de arm rotatie draait naar links, en stopt bij 180°)	X2>1000	Servo 4 neemt geleidelijk toe tot 180° (duw de linkerjoystick naar links, de klauw gaat open)
Y1>1000	Servo 2 geleidelijk reduceert tot 0° (dat wil zeggen, til de robot bovenarm)	Y2>1000	Servo 3 vermindert geleidelijk tot 35° (dat wil zeggen, strek de robot lager uit arm)
J1<50	Servo 2 geleidelijk vermindert tot 180 ° (dat wil zeggen, verlaag de robot bovenarm)	J2<50	Servo 3 neemt geleidelijk toe tot 180° (dat wil zeggen, trek de robot lager terug arm)

aansluiting gids:

Machine Translated by Google





Testcode 4:

Machine Translated by Google


```
**********
**********
```

```
#include <Servo.h> // voeg de servobibliotheken toe

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;



int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoeken en wijs de

initiële waarde (dat is de waarde van de houdingshoek van de laars)



const int rechts_X = A2; // definieer de juiste X-pin naar A2

const int rechts_Y = A5; // definieer de juiste Y-pin naar A5

const int right_key = 7; // definieer de juiste sleutelpin tot 7 (dat is de waarde van Z)



const int linker_X = A3; // definieer de linker X-pin naar A3

const int links_Y = A4; // definieer de linker X-pin naar A4

const int linker_sleutel = 8; // definieer de linkersleutelpin tot 8 ÿdat is de waarde van Zÿ



int x1,y1,z1; // definieer de variabele die wordt gebruikt om de gelezen joystickwaarde op te slaan.

int x2,y2,z2;
```

Machine Translated by Google


```
ongeldige instellingen ()

{

    // laars houding

    mijnservo1.schrijven(pos1);

    vertraging(1000);

    mijnservo2.schrijven(pos2);

    mijnservo3.schrijven(pos3);

    mijnservo4.schrijven(pos4);

    vertraging(1500);

    pinMode (rechter_toets, INPUT); // stel de toets rechts/links in op INPUT

    pinMode (linker_key, INPUT);

    Serieel.begin(9600); // stel de baudrate in op 9600

}

ongeldige lus()

{

    mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1

    mijnservo2.attach(A0); // stel de stuurpen van servo 2 in op A0

    mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
```

Machine Translated by Google




```
    mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9



    x1 = analoogRead(right_X); // lees de juiste X-waarde

    y1 = analoogRead(right_Y); // lees de juiste Y-waarde

    z1 = digitalRead(rechter_sleutel); //// lees de juiste Z-waarde



    x2 = analoogLezen(linker_X); // lees de linker X-waarde

    y2 = analoogLezen(linker_Y); // lees de linker Y-waarde

    z2 = digitalRead(linker_key); // lees de linker Z-waarde

    // vertraging(5); // verlaag de snelheid in het algemeen



    // klauw

    zhuazi();

    // draaien

    zhuandong();

    // bovenarm

    xiaobi();

    //onderarm

    dat();

}



//klauw
```

Machine Translated by Google


```
leegte zhuazi()
{
      //klauw
   if(x2<50) // if duw de linker joystick naar rechts
   {
         pos4=pos4-2; // huidige hoek van servo 4 trekt 2 af ÿwijzig de waarde die u aftrekt, dus
verander de gesloten snelheid van klauwÿ
         //Serial.println(pos4);
         mijnservo4.schrijven(pos4); // servo 4 bedient de actie, klauw wordt geleidelijk gesloten.
         vertraging(5);
         if(pos4<2) // als de waarde van pos4 wordt afgetrokken tot 2, is de klauw in 37 graden die we hebben getest gesloten.
         {                    // (zou de waarde moeten veranderen op basis van het feit)
            pos4=2; // stop met aftrekken bij reduceren tot 2
         }
    }
   if(x2>1000) //// if duw de linker joystick naar links
   {
         pos4=pos4+8; // huidige hoek van servo 4 plus 8ÿverander de waarde die u plus, dus verander
de open snelheid van klauwÿ
         //Serial.println(pos4);
         mijnservo4.schrijven(pos4); // servo 4 bedient de beweging, de klauw gaat geleidelijk open.
         vertraging(5);
```

Machine Translated by Google


```
            if(pos4>108) // beperk de grootste hoek bij het openen van de klauw
            {
                pos4=108;
            }
    }
}
//************************************************** ******
  // draai
leegte zhuandong()
{
    if(x1<50) // if duw de rechter joystick naar rechts
    {
        pos1=pos1-1; //pos1 trekt 1 af
        mijnservo1.schrijven(pos1); // servo 1 bedient de beweging, de arm draait naar rechts.
        vertraging(5);
        if(pos1<1) // beperk de hoek wanneer u naar rechts draait
        {
            pos1=1;
        }
    }
    if(x1>1000) // if duw de rechter joystick naar let
    {
```

Machine Translated by Google




```
        pos1=pos1+1; // pos1 plus 1
        mijnservo1.schrijven(pos1); // arm draait naar links
        vertraging(5);
        if(pos1>180) // beperk de hoek wanneer u naar links draait
        {
            pos1=180;
        }
    }
}

//************************************************** **********/
//bovenarm
leegte xiaobi()
{
        if(y1>1000) // if duw de rechter joystick omhoog
    {
        pos2=pos2-1;
        mijnservo2.schrijven(pos2); // de bovenarm gaat omhoog
        vertraging(5);
        if(pos2<0) // beperk de hefhoek
        {
            pos2=0;
```

Machine Translated by Google




```
        }
    }
    if(y1<50) // if duw de rechter joystick naar beneden
    {
        pos2=pos2+1;
        mijnservo2.schrijven(pos2); // bovenarm gaat naar beneden
        vertraging(5);
        if(pos2>180) // beperk de hoek bij het naar beneden gaan
        {
            pos2=180;
        }
    }
}

//************************************************** *************/
// onderarm
leegte dabi()
{
        if(y2<50) // if duw de linker joystick omhoog
    {
        pos3=pos3+1;
        mijnservo3.schrijven(pos3); // onderarm strekt zich uit
```

Machine Translated by Google


```
        vertraging(5);
        if(pos3>180) // beperk de uitgerekte hoek
        {
            pos3=180;
        }
    }
    if(y2>1000) // if duw de linker joystick naar beneden
    {
        pos3=pos3-1;
        mijnservo3.schrijven(pos3); // onderarm zal terugtrekken
        vertraging(5);
        if(pos3<35) // beperk de ingetrokken hoek
        {
            pos3=35;
        }
    }

}********************************************** ********************************************************* **
**********************
```

Machine Translated by Google

Testresultaat:

Upload de code naar het hoofdbord en stapel het schild erop en sluit ze aan, waarna de 4DOF-robotarm de beginpositie behoudt. Je kunt de robotarm besturen met joysticks

3.4 Geheugenfunctie toevoegen

ÿ Onthoud één houding

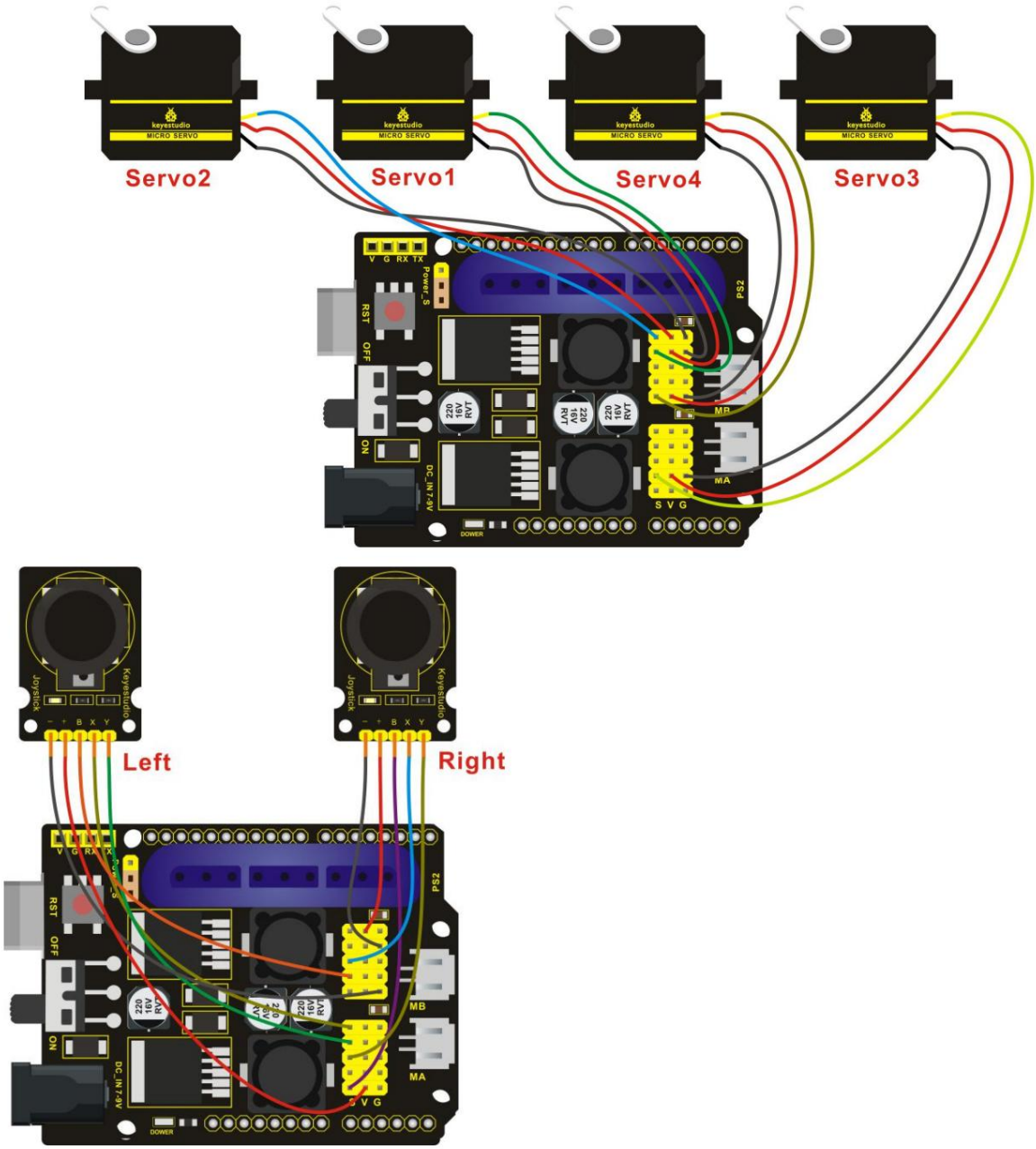
Beschrijving:

Gebruik in het vorige gedeelte de analoge waarde van pin X,Y van 2 joystickmodules om de robotarm te besturen.

In het volgende experiment voegen we een geheugenfunctie toe voor de robotarm, waardoor deze een houding onthoudt en vervolgens handelt. Stel 4 variabelen in voor het opslaan van de hoekwaarde van 4 servo's, gebruik de joystick om een houding te besturen. Druk op de toets Z1 van de rechterjoystick om de hoekwaarde van 4 servo's op te slaan; druk op de Z2-toets van de linkerjoystick om de servo een houding te laten bedienen die is opgeslagen in de variabele.

Machine Translated by Google

Verbindingsdiagram



Testcode 5:

********** ********** ****

****

```
#include <Servo.h> // voeg servobibliotheken toe

Servo mijnservo1; // maak een servo-object om een servo te besturen
```

Machine Translated by Google


```
Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;

int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoek en wijs de
      initiële waarde (dat is de waarde van de houdingshoek van de laars)



const int rechts_X = A2; // definieer de juiste X-pin naar A2

const int rechts_Y = A5; // definieer de juiste Y-pin naar A3

const int right_key = 7; // definieer de juiste sleutelpin tot 7 ÿdat is de Z-waardeÿ



const int linker_X = A3; // definieer de linker X-pin naar A3

const int links_Y = A4; // definieer de linker Y-pin naar A4

const int linker_sleutel = 8; // definieer de linkersleutelpin tot 8 ÿdat is de Z-waardeÿ



int x1,y1,z1; // definieer de variabele die wordt gebruikt om de joystickwaarde op te slaan.

int x2,y2,z2;

int s1,s2,s3,s4;



ongeldige instellingen ()

{

    // laars houding

     mijnservo1.schrijven(pos1);
```

Machine Translated by Google


```
    vertraging(1000);
    mijnservo2.schrijven(pos2);
    mijnservo3.schrijven(pos3);
    mijnservo4.schrijven(pos4);
    vertraging(1500);



    pinMode (rechter_toets, INPUT); // stel de toets rechts/links in op INPUT
    pinMode (linker_key, INPUT);
    Serieel.begin(9600); // stel de baudrate in op 9600
}

ongeldige lus()
{
    mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1
    mijnservo2.attach(A0); // stel de stuurpen van servo 2 in op A0
    mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
    mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9

    x1 = analoogRead(right_X); // lees de juiste X-waarde
    y1 = analoogRead(right_Y); // lees de juiste Y-waarde
    z1 = digitalRead(rechter_sleutel); //// lees de Z-waarde van de juiste sleutel
```

Machine Translated by Google


```
x2 = analoogLezen(linker_X); // lees de linker X-waarde
y2 = analoogLezen(linker_Y); // lees de linker Y-waarde
z2 = digitalRead(linker_key); // lees de Z-waarde van de linkersleutel
// vertraging(5); // verlaag de snelheid in het algemeen
if(z1==1) // als de rechter joysticktoets is ingedrukt
{
    vertraging(10); // vertraging voor het elimineren van schudden
    als(z1==1)          // beoordeel opnieuw of de juiste toets is ingedrukt
    {
        s1=mijnservo1.lezen(); // lees de hoekwaarde van elke servo
        s2=mijnservo2.lezen();
        s3=mijnservo3.lezen();
        s4=mijnservo4.lezen();
    }
}
if(z2==1) // als de linkertoets is ingedrukt
{
    vertraging(10);
    als(z2==1)
    {
        pos1=mijnservo1.lezen(); // noteer de hoekwaarde van 4 servo's in de huidige houding
```

Machine Translated by Google


```
pos2=mijnservo2.lezen();

pos3=mijnservo3.lezen();

pos4=mijnservo4.lezen();



if(pos1<s1) // als de hoek van servo 1 kleiner is dan de waarde van variabele s1

{

  while(pos1<s1) //while loop, draai de servo naar de positie van de opgeslagen waarde
```

de reeks.

```
  {

    mijnservo1.schrijven(pos1); // servo 1 bestuurt de beweging

    pos1++; // pos1 plus 1

    vertraging(5); // vertraging voor 5ms, regeling van de rotatiesnelheid van de servo.

  }



}

else // als de hoek van servo 1 groter is dan de waarde die is opgeslagen in array 1.

{

  terwijl(pos1>s1)          //while loop, draai de servo naar de positie van de opgeslagen waarde
```

de reeks.

```
  {

    mijnservo1.schrijven(pos1); // servo 1 bestuurt de beweging

    pos1--;          //pos1 trekt 1 af
```

Machine Translated by Google




```
            vertraging(5); // vertraging voor 5ms, regeling van de rotatiesnelheid van de servo.
        }
    }
//************************************************** *
// de uitleg is hetzelfde als servo 1
    als(pos2<s2)
    {
        terwijl(pos2<s2)
        {
            mijnservo2.schrijven(pos2);
            pos2++;
            vertraging(5);
        }
    }
    anders
    {
        terwijl(pos2>s2)
        {
            mijnservo2.schrijven(pos2);
            pos2--;
            vertraging(5);
        }
```

Machine Translated by Google




```
    }
//************************************************** *
 // de uitleg is hetzelfde als servo 1
    als(pos3<s3)
    {
      terwijl(pos3<s3)
      {
        mijnservo3.schrijven(pos3);
        pos3++;
        vertraging(5);
      }
    }
    anders
    {
      terwijl(pos3>s3)
      {
        mijnservo3.schrijven(pos3);
        pos3--;
        vertraging(5);
      }
    }
 //************************************************** *
```

Machine Translated by Google




```
        // de uitleg is hetzelfde als servo 1
            als(pos4<s4)
            {
                terwijl(pos4<s4)
                {
                    mijnservo4.schrijven(pos4);
                    pos4++;
                    vertraging(5);
                }
            }
            anders
            {
                terwijl(pos4>s4)
                {
                    mijnservo4.schrijven(pos4);
                    pos4--;
                    vertraging(5);
                }
            }

    }
}
```

Machine Translated by Google


```
    //klauw
    zhuazi();
    //draai
    zhuandong();
    // bovenarm
    xiaobi();
    // onderarm
    dat();
}



//klauw
leegte zhuazi()
{
      //klauw
    if(x2<50) // if duw de linker joystick naar rechts
    {
          pos4=pos4-2; // huidige hoek van servo 4 trekt 2 af ÿwijzig de waarde die u aftrekt, dus
verander de gesloten snelheid van klauwÿ
          //Serial.println(pos4);
          mijnservo4.schrijven(pos4); //servo 4 bedient de actie, klauw wordt geleidelijk gesloten
```

Machine Translated by Google




```
            vertraging(5);

            if(pos4<2) // als de waarde van pos4 wordt afgetrokken tot 2, is de klauw in 37 graden die we hebben getest gesloten.ÿ
            {                       // (zou de waarde moeten veranderen op basis van het feit)
                pos4=2; // stop met aftrekken bij reduceren tot 2
            }
        }
    if(x2>1000) //// if duw de linker joystick naar links
    {
            pos4=pos4+8; // huidige hoek van servo 4 plus 8ÿverander de waarde die u plus, dus verander
de open snelheid van klauwÿ
            //Serial.println(pos4);
            mijnservo4.schrijven(pos4); // servo 4 bedient de beweging, de klauw gaat geleidelijk open.
            vertraging(5);
            if(pos4>90) // beperk de grootste hoek wanneer open
            {
                pos4=90;
            }
    }
}
//************************************************** ******
  // draai
leegte zhuandong()
```

Machine Translated by Google




```
{
    if(x1<50) // if duw de rechter joystick naar rechts
    {
        pos1=pos1-1; //pos1 trekt 1 af
        mijnservo1.schrijven(pos1); // servo 1 bedient de beweging, de arm draait naar rechts.
        vertraging(5);
        if(pos1<1) // beperk de hoek wanneer u naar rechts draait
        {
            pos1=1;
        }
    }
    if(x1>1000) // if duw de rechter joystick naar links
    {
        pos1=pos1+1; // pos1 plus 1
        mijnservo1.schrijven(pos1); // robotarm draait naar links
        vertraging(5);
        if(pos1>180) // beperk de hoek wanneer u naar links draait
        {
            pos1=180;
        }
    }
}
```

Machine Translated by Google


```
//************************************************** **********/
// bovenarm
leegte xiaobi()
{
    if(y1>1000) // if duw de rechter joystick omhoog
  {
    pos2=pos2-1;
    mijnservo2.schrijven(pos2); // de bovenarm gaat omhoog
    vertraging(5);
    if(pos2<0) // beperk de hefhoek
    {
      pos2=0;
    }
  }
  if(y1<50) // if duw de rechter joystick naar beneden
  {
    pos2=pos2+1;
    mijnservo2.schrijven(pos2); // de bovenarm gaat naar beneden
    vertraging(5);
    if(pos2>180) // beperk de hoek bij het naar beneden gaan
    {
```

Machine Translated by Google




```
            pos2=180;
        }
    }
}

//************************************************ *************/
// onderarm
leegte dabi()
{
    if(y2>1000) // if duw de linker joystick omhoog
    {
        pos3=pos3-1;
        mijnservo3.schrijven(pos3); // de onderarm strekt zich uit
        vertraging(5);
        if(pos3<35) // beperk de uitgerekte hoek
        {
            pos3=35;
        }
    }

    if(y2<50) // if duw de linker joystick naar beneden
    {
```

Machine Translated by Google


```
            pos3=pos3+1;
            mijnservo3.schrijven(pos3); // de onderarm trekt zich terug
            vertraging(5);
            if(pos3>180) // beperk de teruggetrokken hoek
            {
               pos3=180;
            }
        }
    }
```

********** ********** ****

****

Testresultaat:

Stapel het schild op V4.0, sluit ze aan, upload de code, steek de stekker in het stopcontact en druk op de toets Z1 van de rechterjoystick om de hoekwaarde van 4 servobesturing op te slaan en druk op de toets Z2 van de linkerjoystick om een opgeslagen servohouding te bedienen in de variabele.

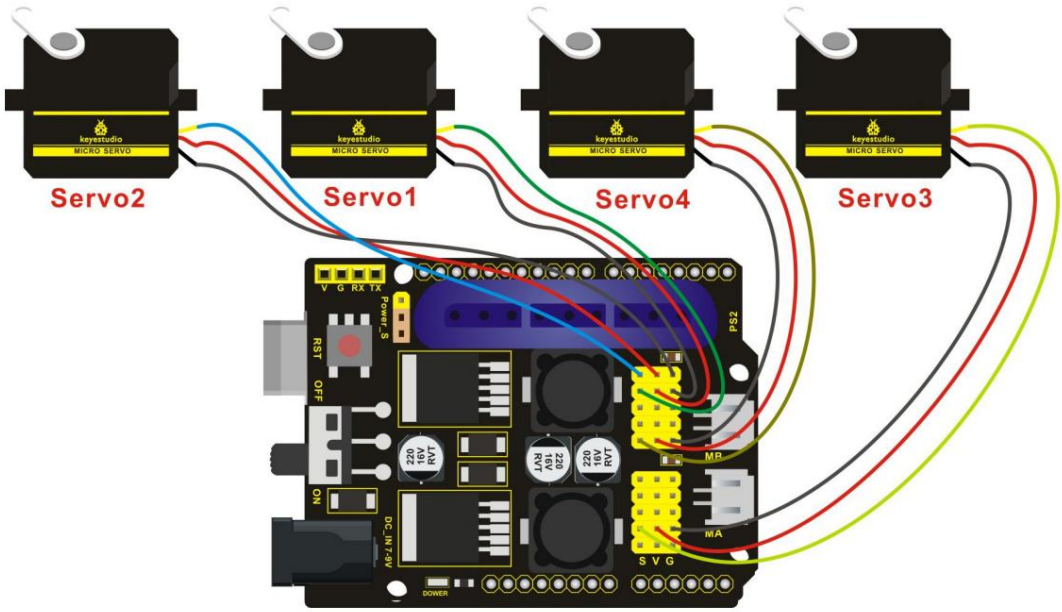
Machine Translated by Google

ÿ Onthoud meerdere houdingen

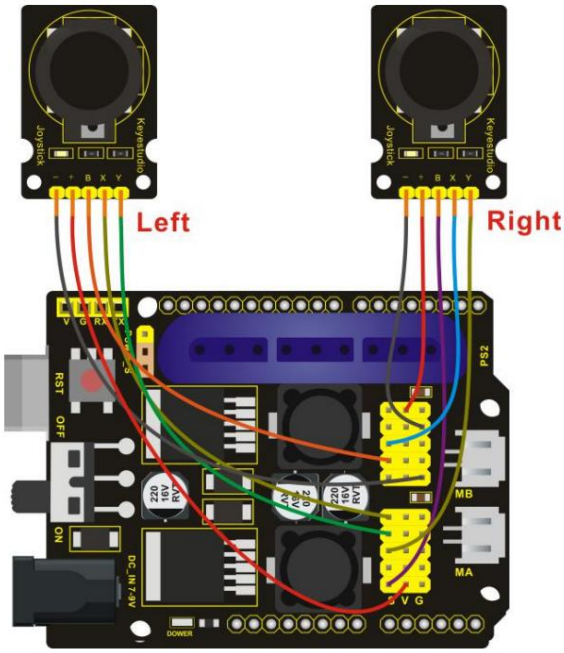
Beschrijving:

In het vorige gedeelte hebben we de hoek van 4 servo's ingesteld om de robotarm een houding te laten onthouden en te laten uitvoeren. Om het experiment uit te breiden, laat u het vervolgens verschillende houdingen onthouden, maximaal 10 (u kunt dit in de code instellen), en laat vervolgens de 4DOF-robotarm de houding in het geheugen continu bedienen. Dat wil zeggen, laat de robotarm een groep acties onthouden en u kunt de geheugensnelheid in de code instellen.

Verbindingsdiagram



Machine Translated by Google



Testcode 6:

********** ********** ****

****

```
#include <Servo.h> // voeg de servobibliotheken toe

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;

int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoek en wijs toe

de initiële waarde (dat is de waarde van de houdingshoek van de laars)



const int rechts_X = A2; // definieer de juiste X-pin naar A2

const int rechts_Y = A5; // definieer de juiste Y-pin naar A5
```

Machine Translated by Google


```
const int right_key = 7; // definieer de juiste sleutelpin tot 7 ÿdat is de Z-waardeÿ



const int linker_X = A3; // definieer de linker X-pin naar A3

const int links_Y = A4; // definieer de linker Y-pin naar A4

const int linker_sleutel = 8; // definieer de linkersleutelpin tot 8 ÿdat is de Z-waardeÿ



int x1,y1,z1; // definieer de variabele, gebruikt om de joystickwaarde op te slaan.

int x2,y2,z2;

int s1,s2,s3,s4;

int jiyi1[10]; // definieer 4 array, afzonderlijk gebruikt om de hoek van vier servo's op te slaan.

int jiyi2[10]; // (array-lengte is 10, kan namelijk hoekgegevens van 0 ~ 10 servo opslaan)

int jiyi3[10]; // als u meer gegevens wilt opslaan, wijzigt u gewoon het getal 10 in een groter getal.

int jiyi4[10];

int i=0; // for loop

int j=0; // bewaar de laatste waarde van i



ongeldige instellingen ()

{

    // laars houding

    mijnservo1.schrijven(pos1);

    vertraging(1000);

    mijnservo2.schrijven(pos2);
```

Machine Translated by Google


```
    mijnservo3.schrijven(pos3);
    mijnservo4.schrijven(pos4);
    vertraging(1500);



    pinMode (rechter_toets, INPUT); // stel de toets rechts/links in op INPUT
    pinMode (linker_key, INPUT);
    Serieel.begin(9600); // stel de baudrate in op 9600
}

ongeldige lus()
{
    mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1
    mijnservo2.attach(A0); // stel de stuurpen van servo 2 in op A0
    mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
    mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9

    x1 = analoogRead(right_X); // lees de juiste X-waarde
    y1 = analoogRead(right_Y); // lees de juiste Y-waarde
    z1 = digitalRead(rechter_sleutel); // lees de juiste Z-waarde

    x2 = analoogLezen(linker_X); // lees de linker X-waarde
```

Machine Translated by Google


```
y2 = analoogLezen(linker_Y); // lees de linker Y-waarde

z2 = digitalRead(linker_key); // lees de linker Z-waarde

// vertraging(5); // verlaag de snelheid in het algemeen

if(z1==1) // als de rechter joysticktoets is ingedrukt

{

    vertraging(10); // vertraging voor het elimineren van schudden

    if(z1==1) // beoordeel opnieuw of de juiste toets is ingedrukt

    {

        s1=mijnservo1.lezen();              // lees de hoekwaarde van elke servo

        vertraging(100);

        Serieel.println(s1);

        s2=mijnservo2.lezen();

        vertraging(100);

        Serieel.println(s2);

        s3=mijnservo3.lezen();

        vertraging(100);

        Serieel.println(s3);

        s4=mijnservo4.lezen();

        vertraging(100);

        Serieel.println(s4);



       jiyi1[i]=s1; // Sla de gelezen servowaarde achtereenvolgens op in de array
```

Machine Translated by Google




```
        jiyi2[i]=s2;
        jiyi3[i]=s3;
        jiyi4[i]=s4;
        ik++;          // i-waarde plus 1
        j=ik;          // wijs de laatste waarde van i toe aan j
        vertraging(100);
        Serieel.println(i); // druk op de seriële monitor de waarde i af
    }
}

if(z2==1) // als de linker joysticktoets wordt ingedrukt
{
    vertraging(10);
    if(z2==1) // beoordeel opnieuw of de linkertoets is ingedrukt
    {
        ik=0; // wijs i toe aan 0, bereid je voor op de volgende herinnering
        pos1 = mijnservo1.lezen();              // onthoud de hoekwaarde van 4 servohoudingen
        pos2 = mijnservo2.lezen();
        pos3 = mijnservo3.lezen();
        pos4 = mijnservo4.lezen();

        for(int k=0;k<j;k++) // lus voor j keer, voer alle opgeslagen acties uit.
```

Machine Translated by Google




```
{
    als(pos1<jiyi1[k])          // als de huidige hoek van servo 1 kleiner is dan de waarde die is opgeslagen in array 1.
    {
      while(pos1<jiyi1[k]) //while lus, laat de servo draaien naar de positie van de waarde die is opgeslagen in
de reeks.
      {
        mijnservo1.schrijven(pos1); // servo 1 voert de actie uit
        vertraging(5);        // vertraging 5ms, regeling van de rotatiesnelheid van de servo
        pos1++; // pos1 plus 1
       //Serial.println(pos1);
      }
    }
    anders        // als de huidige hoek van servo 1 groter is dan de waarde die is opgeslagen in array 1.
    {
     terwijl(pos1>jiyi1[k])              //while lus, laat de servo draaien naar de positie van de opgeslagen waarde
de reeks.
      {
        mijnservo1.schrijven(pos1); // servo 1 voert de actie uit
        vertraging(5);        // vertraging 5 ms, regeling van de rotatiesnelheid van de servo
        pos1--;           //pos1 trekt 1 af
       //Serial.println(pos1);
      }
```

Machine Translated by Google




```
            }
//************************************************ ***************
  // de uitleg is hetzelfde als de vorige servo
            als(pos2<jiyi2[k])
            {
              terwijl(pos2<jiyi2[k])
              {
                mijnservo2.schrijven(pos2);
                vertraging(5);
                pos2++;
                //Serial.println(pos1);
              }
            }
            anders
            {
             terwijl(pos2>jiyi2[k])
              {
                mijnservo2.schrijven(pos2);
                vertraging(5);
                pos2--;
                //Serial.println(pos1);
              }
```

Machine Translated by Google




```
        }
//************************************************** ***************
  // de uitleg is hetzelfde als de vorige servo


        als(pos3<jiyi3[k])
        {
          terwijl(pos3<jiyi3[k])
          {
            mijnservo3.schrijven(pos3);
            vertraging(5);
            pos3++;
           //Serial.println(pos1);
          }
        }
        anders
        {
         terwijl(pos3>jiyi3[k])
          {
            mijnservo3.schrijven(pos3);
            vertraging(5);
            pos3--;
           //Serial.println(pos1);
```

Machine Translated by Google




```
        }
      }
//************************************************ ***************
  // de uitleg is hetzelfde als de vorige servo

      als(pos4<jiyi4[k])
      {
        terwijl(pos4<jiyi4[k])
        {
          mijnservo4.schrijven(pos4);
          vertraging(5);
          pos4++;
          //Serial.println(pos1);
        }
      }
      anders
      {
       terwijl(pos4>jiyi4[k])
        {
          mijnservo4.schrijven(pos4);
          vertraging(5);
          pos4--;
```

Machine Translated by Google




```
                    //Serial.println(pos1);
                }
            }
        }
    }
}

//klauw
zhuazi();
//draai
zhuandong();
//bovenarm
xiaobi();
// onderarm
dat();
}



//klauw
leegte zhuazi()
{
    //klauw
```

Machine Translated by Google


```
    if(x2<50) // if duw de linker joystick naar rechts
    {
          pos4=pos4-2; // hoek van servo 4, trek 2 af (wijzig de waarde die je aftrekt, verander dus
de gesloten snelheid van klauw)
          //Serial.println(pos4);
          mijnservo4.schrijven(pos4); // servo 4 bestuurt de beweging en de klauw wordt geleidelijk gesloten.
          vertraging(5);
          if(pos4<2) // als de waarde van pos4 wordt afgetrokken tot 2, is de klauw in 37 graden die we hebben getest gesloten.ÿ
          {                    // (zou de waarde moeten veranderen op basis van het feit)
              pos4=2; // stop met aftrekken bij reduceren tot 2
          }
     }
    if(x2>1000) //// if duw de linker joystick naar links
    {
          pos4=pos4+8; // huidige hoek van servo 4 plus 8ÿverander de waarde die u plus, dus verander
de open snelheid van klauwÿ
          //Serial.println(pos4);
          mijnservo4.schrijven(pos4); // servo 4 bedient de actie, klauw gaat geleidelijk open.
          vertraging(5);
          if(pos4>90) // beperk de grootste geopende hoek
          {
              pos4=90;
```

Machine Translated by Google




```
            }
        }
    }
//************************************************ ******
  // draai
leegte zhuandong()
{
    if(x1<50) // if duw de rechter joystick naar rechts
    {
        pos1=pos1-1; //pos1 trekt 1 af
        mijnservo1.schrijven(pos1); // servo 1 bedient de beweging en de robotarm draait naar rechts
        vertraging(5);
        if(pos1<1) // beperk de hoek wanneer u naar rechts draait
        {
            pos1=1;
        }
    }
    if(x1>1000) // if duw de rechter joystick naar links
    {
        pos1=pos1+1; // pos1 plus 1
        mijnservo1.schrijven(pos1); // robotarm draait naar links
        vertraging(5);
```

Machine Translated by Google


```
        if(pos1>180) // beperk de hoek wanneer u naar links draait
        {
            pos1=180;
        }
    }
}

//************************************************** **********/
// bovenarm
leegte xiaobi()
{
        if(y1>1000) // if duw de rechter joystick omhoog
    {
        pos2=pos2-1;
        mijnservo2.schrijven(pos2); // de bovenarm gaat omhoog
        vertraging(5);
        if(pos2<0) // beperk de hefhoek
        {
            pos2=0;
        }
    }
    if(y1<50) // if duw de rechter joystick naar beneden
```

Machine Translated by Google




```
    {
        pos2=pos2+1;
        mijnservo2.schrijven(pos2); // de bovenarm gaat naar beneden
        vertraging(5);
        if(pos2>180) // beperk de neerwaartse hoek
        {
            pos2=180;
        }
    }
}

//************************************************** **********/
// onderarm
leegte dabi()
{
    if(y2>1000) // if duw de linker joystick omhoog
    {
        pos3=pos3-1;
        mijnservo3.schrijven(pos3); // de onderarm strekt zich uit
        vertraging(5);
        if(pos3<35) // beperk de uitgerekte hoek
        {
```

Machine Translated by Google




```
            pos3=35;
          }
        }

        if(y2<50) // if duw de linker joystick naar beneden
        {
          pos3=pos3+1;
          mijnservo3.schrijven(pos3); // de onderarm trekt zich terug
          vertraging(5);
          if(pos3>180) // beperk de teruggetrokken hoek
          {
            pos3=180;
          }
        }
    }
```

****

****

Testresultaat:

Sluit het aan, stapel het schild op V4.0, upload de code. Aangeschakeld, druk op de Z1-toets van de rechterjoystick om de hoekwaarde van 4 servo's op te slaan. Druk op toets Z1 om verschillende houdingen te onthouden, maximaal 10 houdingen in de code. Als u meer houdingen wilt onthouden, kunt u dit in de code instellen. Wanneer

Machine Translated by Google

succesvol onthouden, druk op de toets Z2 van de linker joystick om de robotarm verschillende achtereenvolgens opgeslagen houdingen te laten uitvoeren.

Beweeg de thumbsticks en druk dan op de knop van de rechter thumbstick verplaats ze en druk nogmaals op de rechterknop. Dat geeft aan dat acties van de arm worden onthouden. Vervolgens kunt u op de linkerknop drukken om uit te voeren acties opgeslagen.

ÿ Onthoud verschillende houdingen en lus

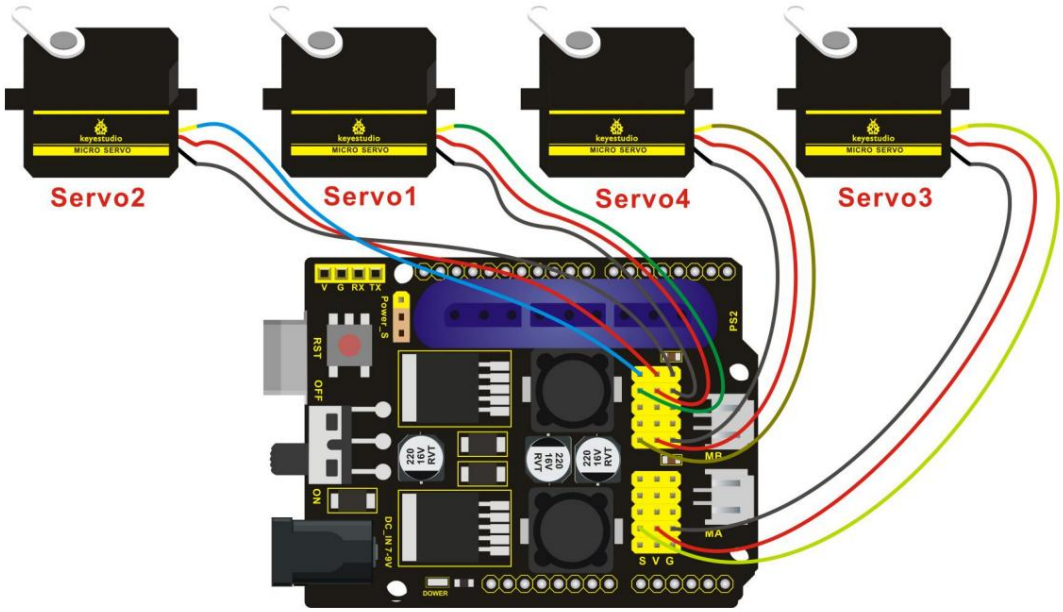
Beschrijving:

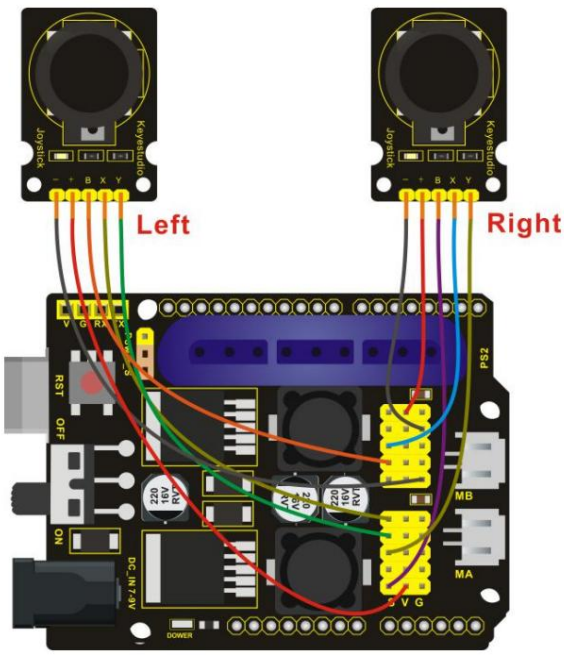
In het vorige gedeelte hebben we geïntroduceerd hoe u een 4DOF-robotarm kunt maken om een groep houdingen te onthouden en uit te voeren. Laten we verder verlengen nog een lusfunctie. Wanneer de robotarm alle opgeslagen taken uitvoert acties, zal het niet stoppen en deze acties blijven herhalen.

Druk in het volgende experiment op de toets Z1, de 4DOF-robotarm verlaat de lus actie. Druk nogmaals op de toets Z1, begin daarna de houding te onthouden dat, druk op de toets Z2 om de opgeslagen acties te herhalen.

aansluiting gids:

Machine Translated by Google





Testcode 7:

********** ********** ****

****

```
#include <Servo.h> // voeg de servobibliotheken toe

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;
```

Machine Translated by Google


```
Servo mijnservo3;

Servo mijnservo4;

int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoek en wijs de

initiële waarde (dat is de waarde van de houdingshoek van de laars)

const int rechts_X = A2; // definieer de juiste X-pin naar A2

const int rechts_Y = A5; // definieer de juiste Y-pin naar A5

const int right_key = 7; // definieer de juiste sleutelpin tot 7 ÿdat is de Z-waardeÿ



const int linker_X = A3; // definieer de linker X-pin naar A3

const int links_Y = A4; // definieer de linker Y-pin naar A4

const int linker_sleutel = 8; // definieer de linkersleutelpin tot 8 ÿdat is de Z-waardeÿ



int x1,y1,z1; // definieer de variabele die wordt gebruikt om de joystickwaarde op te slaan.

int x2,y2,z2;

int s1,s2,s3,s4;

int jiyi1[20]; // definieer 4 array, afzonderlijk gebruikt om de hoek van vier servo's op te slaan.

int jiyi2[20]; // (array-lengte is 20, kan namelijk hoekgegevens van 0 ~ 20 servo opslaan)

int jiyi3[20]; // als u meer gegevens wilt opslaan, wijzigt u gewoon het getal 20 in een groter getal.

int jiyi4[20];

int i=0; // for loop

int j=0; // bewaar de laatste waarde van i
```

Machine Translated by Google




```
ongeldige instellingen ()
{
    // laars houding
    mijnservo1.schrijven(pos1); // draai servo 1 naar 90 graden
    vertraging(1000);
    mijnservo2.schrijven(pos2); // draai servo 2 naar 90 graden
    mijnservo3.schrijven(pos3); // draai servo 3 naar 120 graden
    mijnservo4.schrijven(pos4); // draai servo 4 naar 35 graden
    vertraging(1500);



    pinMode (rechter_toets, INPUT); // stel de toets rechts/links in op INOUT
    pinMode (linker_key, INPUT);
    Serieel.begin(9600); // stel de baudrate in op 9600
}

ongeldige lus()
{
    mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1
    mijnservo2.attach(A0); // stel de stuurpen van servo 2 in op A0
    mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
    mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9
```

Machine Translated by Google




```
x1 = analoogRead(right_X); // lees de juiste X-waarde
y1 = analoogRead(right_Y); // lees de juiste Y-waarde
z1 = digitalRead(rechter_sleutel); // lees de juiste Z-waarde

x2 = analoogLezen(linker_X); // lees de linker X-waarde
y2 = analoogLezen(linker_Y); // lees de linker Y-waarde
z2 = digitalRead(linker_key); // lees de linker Z-waarde
// vertraging(5); // vertraging, gebruikt om de gelezen waarde van de joystick te verminderen, dat wil zeggen de hele snelheid te verlagen.
if(z1==1) // als de joystick naar rechts is ingedrukt
{
    vertraging(10); // vertraging voor het elimineren van schudden
    if(z1==1) // beoordeel opnieuw of de juiste toets is ingedrukt
    {
        s1=mijnservo1.lezen(); // lees de hoekwaarde van servo 1 en wijs deze toe aan s1
        vertraging(100);
        Serieel.println(s1); // druk de hoekwaarde van servo 1 af op de seriële monitor
        s2=mijnservo2.lezen(); // lees de hoekwaarde van servo 2 en wijs deze toe aan s2
        vertraging(100);
        Serieel.println(s2);
        s3=mijnservo3.lezen(); // lees de hoekwaarde van servo 3 en wijs deze toe aan s3
        vertraging(100);
```

Machine Translated by Google




```
        Serieel.println(s3);
        s4=mijnservo4.lezen(); // lees de hoekwaarde van servo 4 en wijs deze toe aan s4
        vertraging(100);
        Serieel.println(s4);


       jiyi1[i]=s1; // Sla de gelezen servowaarde achtereenvolgens op in de array
       jiyi2[i]=s2;
       jiyi3[i]=s3;
       jiyi4[i]=s4;
        ik++; // ik plus 1
       j=ik; // wijs de laatste waarde van i toe aan j
        vertraging(100); // vertraging 100ms
        Serieel.println(i); // print de waarde i uit
    }
}
if(z2==1) // als de linker joysticktoets wordt ingedrukt
{
    vertraging(10); // vertraging voor het elimineren van schudden
    if(z2==1) //oordeel opnieuw als de linkertoets wordt ingedrukt
    {
        pos1 = mijnservo1.lezen(); // onthoud de hoekwaarde van 4 servohoudingen
        pos2 = mijnservo2.lezen();
```

Machine Translated by Google




```
        pos3 = mijnservo3.lezen();
        pos4 = mijnservo4.lezen();


        while(z2==1) // loop, laat de arm de actie herhalen.
        {
          for(int k=1;k<j;k++) //for lus, voer alle opgeslagen acties uit.
          {
            if(pos1<jiyi1[k]) // als de huidige hoek van servo 1 kleiner is dan de waarde die is opgeslagen in array 1.
            {
              while(pos1<jiyi1[k]) //while lus, laat de servo draaien naar de positie van de waarde die is opgeslagen in
de reeks.
              {
                mijnservo1.schrijven(pos1); //servo 1 voert de actie uit
                vertraging(5); // vertraging 5 ms, regeling van de rotatiesnelheid van de servo.
                pos1++; // pos1 plus 1
                //Serial.println(pos1);
              }
            }
            else // als de huidige hoek van servo 1 groter is dan de waarde die is opgeslagen in array 1.
            {
              terwijl(pos1>jiyi1[k])          //while loop, laat de servo draaien naar de positie van de opgeslagen waarde
in de reeks.
```

Machine Translated by Google




```
                {
                    mijnservo1.schrijven(pos1); //servo 1 voert de actie uit
                    vertraging(5); // vertraging 5 ms, regeling van de rotatiesnelheid van de servo.
                    pos1--; //pos1 trekt 1 af
                   //Serial.println(pos1);
                }
            }
//************************************************** ***************
// de uitleg is hetzelfde als de vorige servo.
            als(pos2<jiyi2[k])
            {
               terwijl(pos2<jiyi2[k])
               {
                   mijnservo2.schrijven(pos2);
                   vertraging(5);
                   pos2++;
                  //Serial.println(pos1);
               }
            }
            anders
            {
              terwijl(pos2>jiyi2[k])
```

Machine Translated by Google




```
                {
                    mijnservo2.schrijven(pos2);
                    vertraging(5);
                    pos2--;
                    //Serial.println(pos1);
                }
            }
//*********************************************
// de uitleg is hetzelfde als de vorige servo.
            als(pos3<jiyi3[k])
            {
                terwijl(pos3<jiyi3[k])
                {
                    mijnservo3.schrijven(pos3);
                    vertraging(5);
                    pos3++;
                    //Serial.println(pos1);
                }
            }
            anders
            {
              terwijl(pos3>jiyi3[k])
```

Machine Translated by Google




```
                {
                    mijnservo3.schrijven(pos3);
                    vertraging(5);
                    pos3--;
                    //Serial.println(pos1);
                }
            }
//*****************************************
// de uitleg is hetzelfde als de vorige servo.
            als(pos4<jiyi4[k])
            {
                terwijl(pos4<jiyi4[k])
                {
                    mijnservo4.schrijven(pos4);
                    vertraging(5);
                    pos4++;
                    //Serial.println(pos1);
                }
            }
            anders
            {
              terwijl(pos4>jiyi4[k])
```

Machine Translated by Google


```
                    {
                        mijnservo4.schrijven(pos4);
                        vertraging(5);
                        pos4--;
                        //Serial.println(pos1);
                    }
                }
            }
//************************************************** ************
// voor het verlaten van de lus
                    z1 = digitalRead(rechter_sleutel); // lees de juiste Z-waarde
                    if(z1==1) // als de rechtertoets is ingedrukt
                    {
                        vertraging(10); // Elimineer de shake
                        if(z1==1) // als de toets z1 is ingedrukt
                        {
                            pos1=jiyi1[(j-1)]; // wijs de laatste in de array opgeslagen hoekwaarde toe aan pos
                            pos2=jiyi2[(j-1)]; // voor het verlaten van de lus, nog steeds toegang tot joystickbesturing.
                            pos3=jiyi3[(j-1)];
                            pos4=jiyi4[(j-1)];
                            ik=0; // wijs i toe als 0, bereid u voor op het opslaan van de hoekwaarde met behulp van array
                            z2=0; // wijs z2 toe als 0, om de while-lus te verlaten
```

Machine Translated by Google




```
                        pauze; // verlaat de huidige lus
                    }
                }
//************************************************ ********
            }
        }
    }

    //klauw
    zhuazi();
    //draai
    zhuandong();
    //bovenarm
    xiaobi();
    //onderarm
    dat();
}



//klauw
leegte zhuazi()
{
```

Machine Translated by Google


```
        //klauw
    if(x2<50) // if duw de linker joystick naar rechts
    {
          pos4=pos4-2; // hoek van servo 4, trek 2 af (wijzig de waarde die je aftrekt, verander dus
de gesloten snelheid van klauw)
          //Serial.println(pos4);
          mijnservo4.schrijven(pos4); // servo 4 bestuurt de beweging en de klauw wordt geleidelijk gesloten.
          vertraging(5);
          if(pos4<2) // als de waarde van pos4 wordt afgetrokken tot 2, is de klauw in 37 graden die we hebben getest gesloten.ÿ
          {                    // (zou de waarde moeten veranderen op basis van het feit)
            pos4=2; // stop aftrekken bij verminderen tot 2
          }
     }
    if(x2>1000) ////if duw de linker joystick naar links
    {
          pos4=pos4+8; // huidige hoek van servo 4 plus 8 ÿverander de waarde die u plus, dus verander
de open snelheid van klauwÿ
          //Serial.println(pos4);
          mijnservo4.schrijven(pos4); // servo 4 bedient de actie, klauw gaat geleidelijk open.
          vertraging(5);
          if(pos4>90) //beperk de grootste geopende hoek
          {
```

Machine Translated by Google




```
            pos4=90;
          }
    }
}
//************************************************ ******
 //draai
leegte zhuandong()
{
    if(x1<50) //if duw de rechter joystick naar rechts
    {
        pos1=pos1-1; //pos1 trekt 1 af
        mijnservo1.schrijven(pos1); // servo 1 voert de actie uit, de robotarm draait naar rechts.
        vertraging(5);
        if(pos1<1) // beperk de juiste draaihoek
        {
            pos1=1;
        }
    }
    if(x1>1000) // if duw de rechter joystick naar links
    {
        pos1=pos1+1; // pos1 plus 1
        mijnservo1.schrijven(pos1); //de robotarm draait naar links.
```

Machine Translated by Google




```
        vertraging(5);
        if(pos1>180) //beperk de draaihoek naar links
        {
            pos1=180;
        }
    }
}

//************************************************** **********/
// bovenarm
leegte xiaobi()
{
        if(y1>1000) // if duw de rechter joystick omhoog
    {
        pos2=pos2-1;
        mijnservo2.schrijven(pos2); // de robotarm gaat omhoog
        vertraging(5);
        if(pos2<0) // beperk de hefhoek
        {
            pos2=0;
        }
    }
```

Machine Translated by Google




```
    if(y1<50) // if duw de rechter joystick naar beneden
    {
        pos2=pos2+1;
        mijnservo2.schrijven(pos2); // de robotarm gaat naar beneden
        vertraging(5);
        if(pos2>180) // beperk de neerwaartse hoek
        {
            pos2=180;
        }
    }
}

//************************************************** *************/
// onderarm
leegte dabi()
{
    if(y2>1000) // if duw de linker joystick omhoog
    {
        pos3=pos3-1;
        mijnservo3.schrijven(pos3); // de onderarm strekt zich uit
        vertraging(5);
        if(pos3<35) // beperk de uitgerekte hoek
```

Machine Translated by Google


```
        {
            pos3=35;
        }
    }

    if(y2<50) // if duw de rechter joystick naar beneden
    {
        pos3=pos3+1;
        mijnservo3.schrijven(pos3); // de onderarm trekt zich terug
        vertraging(5);
        if(pos3>180) // beperk de terugtrekhoek
        {
            pos3=180;
        }
    }
}
```

Testresultaat:

Sluit het aan, stapel het schild op V4.0, upload de code. Aangeschakeld, druk op de Z1-toets van de rechterjoystick om de hoekwaarde van 4 servo's op te slaan. Druk op toets Z1 om verschillende houdingen te onthouden, maximaal 10 houdingen in de code.
Als u meer houdingen wilt onthouden, kunt u dit in de code instellen.

Machine Translated by Google

Wanneer het onthouden is gelukt, drukt u op de Z2-toets van de linkerjoystick om de robotarm verschillende achtereenvolgens opgeslagen houdingen in een lus te laten uitvoeren.

Druk lang op de toets Z1, de 4DOF-robotarm zal de lusactie verlaten. Druk nogmaals op toets Z1, begin met het onthouden van de houding, druk daarna op toets Z2 om de opgeslagen acties te herhalen.

****

Beweeg de thumbsticks en druk dan op de knop van de rechter thumbstick verplaats ze en druk nogmaals op de rechterknop. Dat geeft aan dat acties van de arm worden onthouden. Vervolgens kunt u op de linkerknop drukken om uit te voeren acties opgeslagen.

Project 4: BT-gestuurde robotarm

ÿ **Principe van BT-besturing**

Bluetooth-technologie is een draadloze standaardtechnologie die kan worden gerealiseerd gegevensuitwisseling op korte afstand tussen vaste apparatuur, mobiele apparatuur en personal area network (UHF-radiogolven in de ISM-band van 2.4-2.485 GHz). In de kit rusten we uit met de HM-10 Bluetooth-module, een master-slaaf geïntegreerde machine. Wanneer het als meester actief commando kan verzenden s aan de slaaf; als het een slaaf is, kan het alleen het commando van de ma ontvangen ster. Tegelijkertijd ondersteunt de HM-10 Bluetooth-module de Bluetooth 4. 0 en Android /IOS-systeem.

Machine Translated by Google

In het experiment gebruiken we standaard de HM-10 Bluetooth-module als de slave en de mobiele telefoon als de meester. We ontwerpen speciaal APP om robotica te besturen wapen (Android /IOS-systeem)



Specificatie

Bluetooth-protocol: Bluetooth-specificatie V4.0 BLE

Geen bytelimiet in seriële poort Transceiving

Realiseer in een open omgeving 100 m ultraafstandscommunicatie met iphone4s

USB-protocol: USB V2.0

Werkfrequentie: 2,4 GHz ISM-band

Modulatiemethode: GFSK (Gaussian Frequency Shift Keying)

Zendvermogen: -23dbm, -6dbm, 0dbm, 6dbm, kan worden gewijzigd met AT-opdracht.

Gevoeligheid: ÿ-84dBm bij 0,1% BER

Overdrachtssnelheid: Asynchroon: 6K bytes; Synchroon: 6k Bytes

Machine Translated by Google

Beveiligingsfunctie: authenticatie en codering

Ondersteunende service: Central & Peripheral UUID FFE0, FFE1

Stroomverbruik: automatische slaapstand, stand-by stroom 400uA~800uA, 8,5mA tijdens verzending.

Voeding: 5V DC

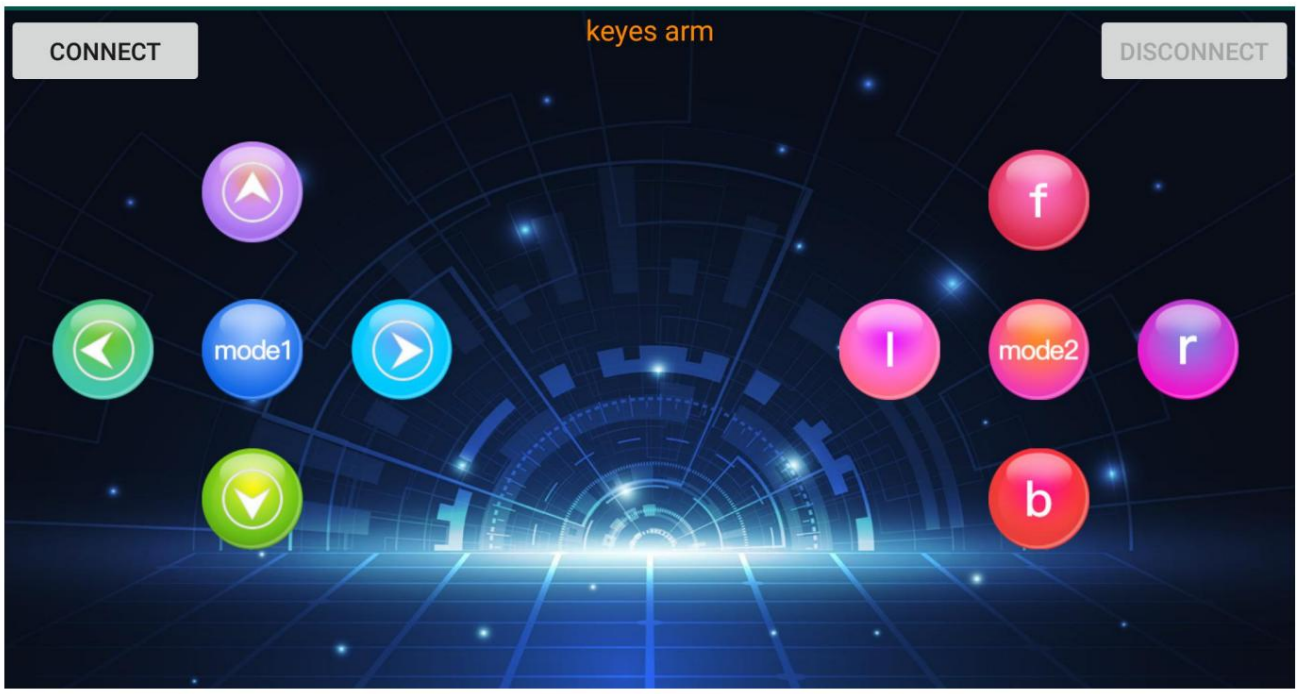
Werktemperatuur: -5 tot +65 Celsius

ÿ **Bluetooth-controlesleuteltest**

Beschrijving

Vervolgens introduceren we de gebruiksmethode voor de HC-06 Bluetooth-module. Naar gebruik eenvoudig de HC-06 Bluetooth-module om vooral de robotarm te besturen

ontwerp de APP-besturing. Hieronder weergegeven.



Machine Translated by Google

Er zijn 10 bedieningstoetsen op de App. Wanneer u de HC-06 Bluetooth-module goed aansluit op de Android-telefoon met behulp van onze APP, drukt u op de bedieningstoets, de Android-telefoon ontvangt een overeenkomstige waarde.
Bij het programmeren kunt u de functie voor de bijbehorende waarde instellen. Dus in het experiment testen we elke sleutel om de overeenkomstige waarde te krijgen.

Installatiestappen voor Android-systeem:

APP voor Android mobiel:

Voer google play in, zoek "keyes arm".

https://play.google.com/store/apps/details?id=com.keyestudio.keyes_arm

123

Opmerking: geef APP toegang tot "locatie" in de instellingen van uw mobiele telefoon wanneer u verbinding maakt met de Bluetooth-module, anders werkt Bluetooth mogelijk niet verbonden zijn.

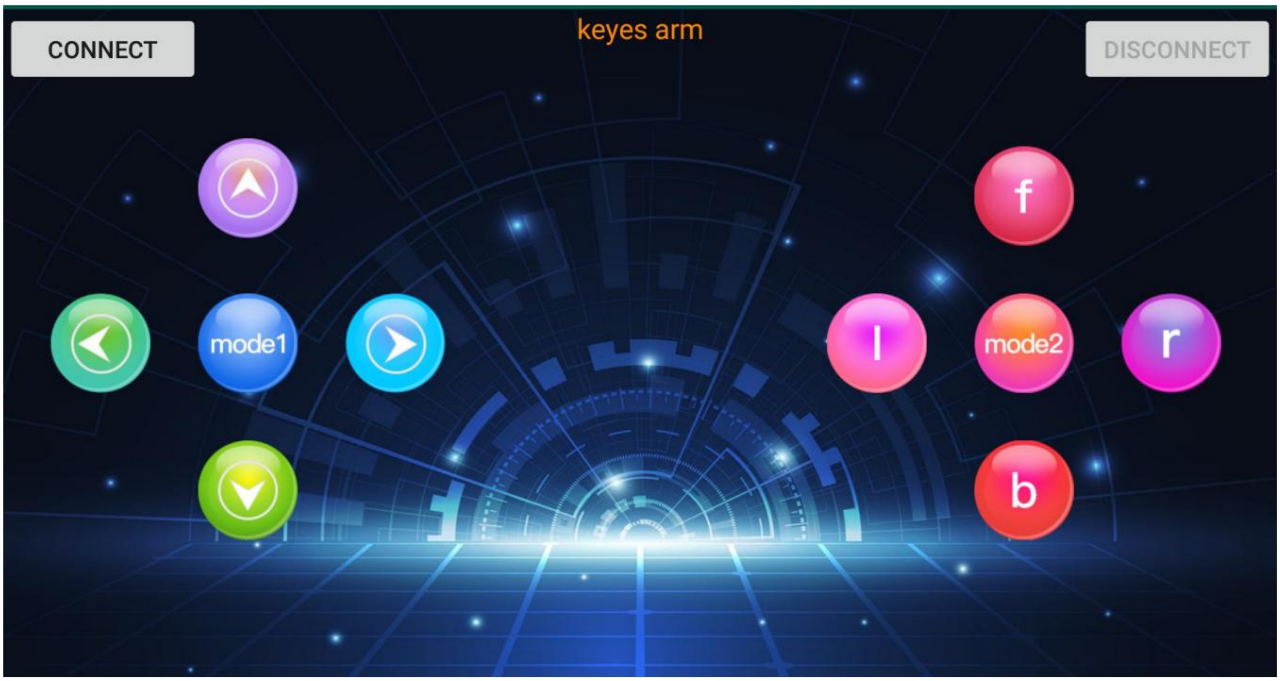
Dit zijn bedieningsstappen zoals hieronder, de interface voor Android en iOS-systeem is dezelfde.

Android-systeem:

Machine Translated by Google

1. Downloaden en installeren 

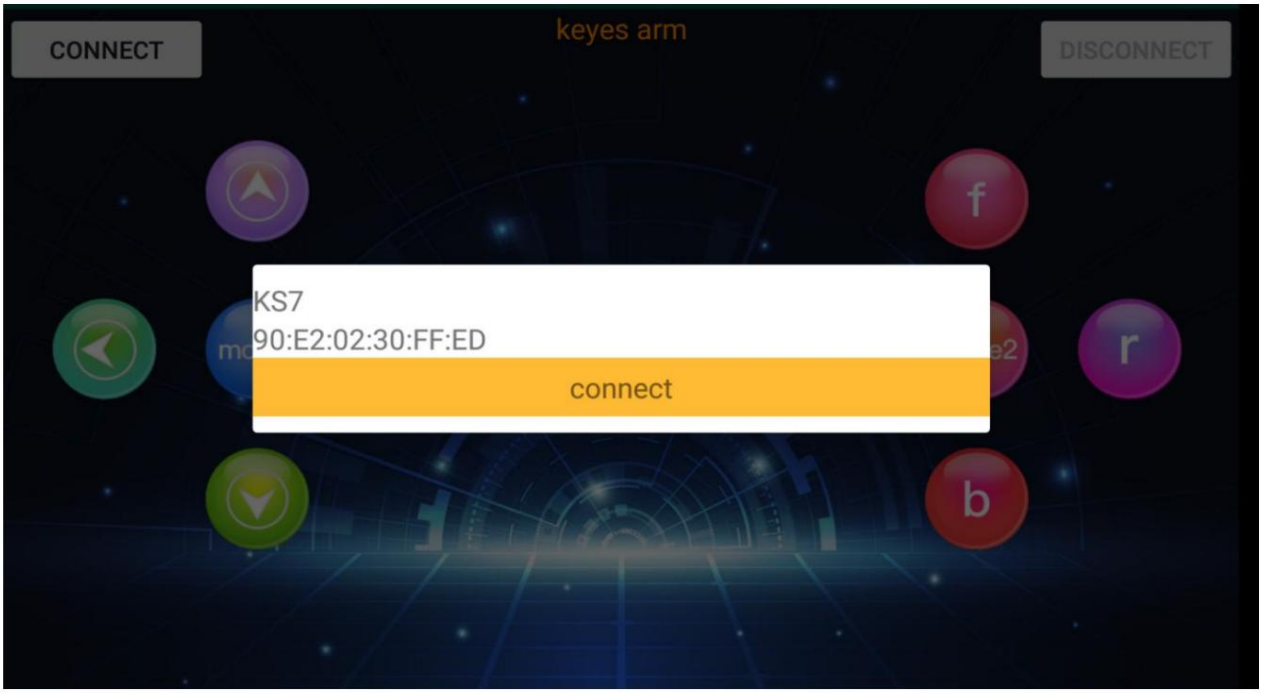
ýde onderstaande interface:



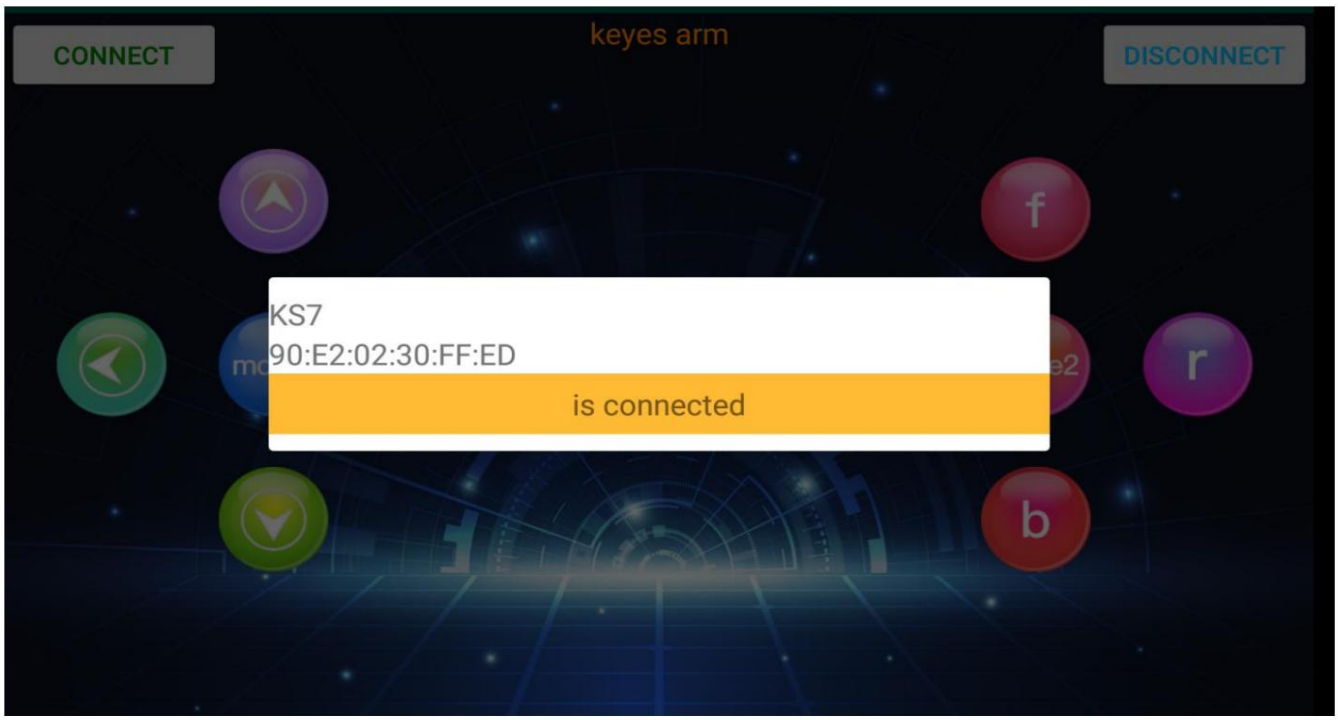
2. Upload code en schakel in, Led van Bluetooth-module knippert. Bluetooth starten

en open de app om op "VERBINDEN" te klikken om verbinding te maken.

Machine Translated by Google

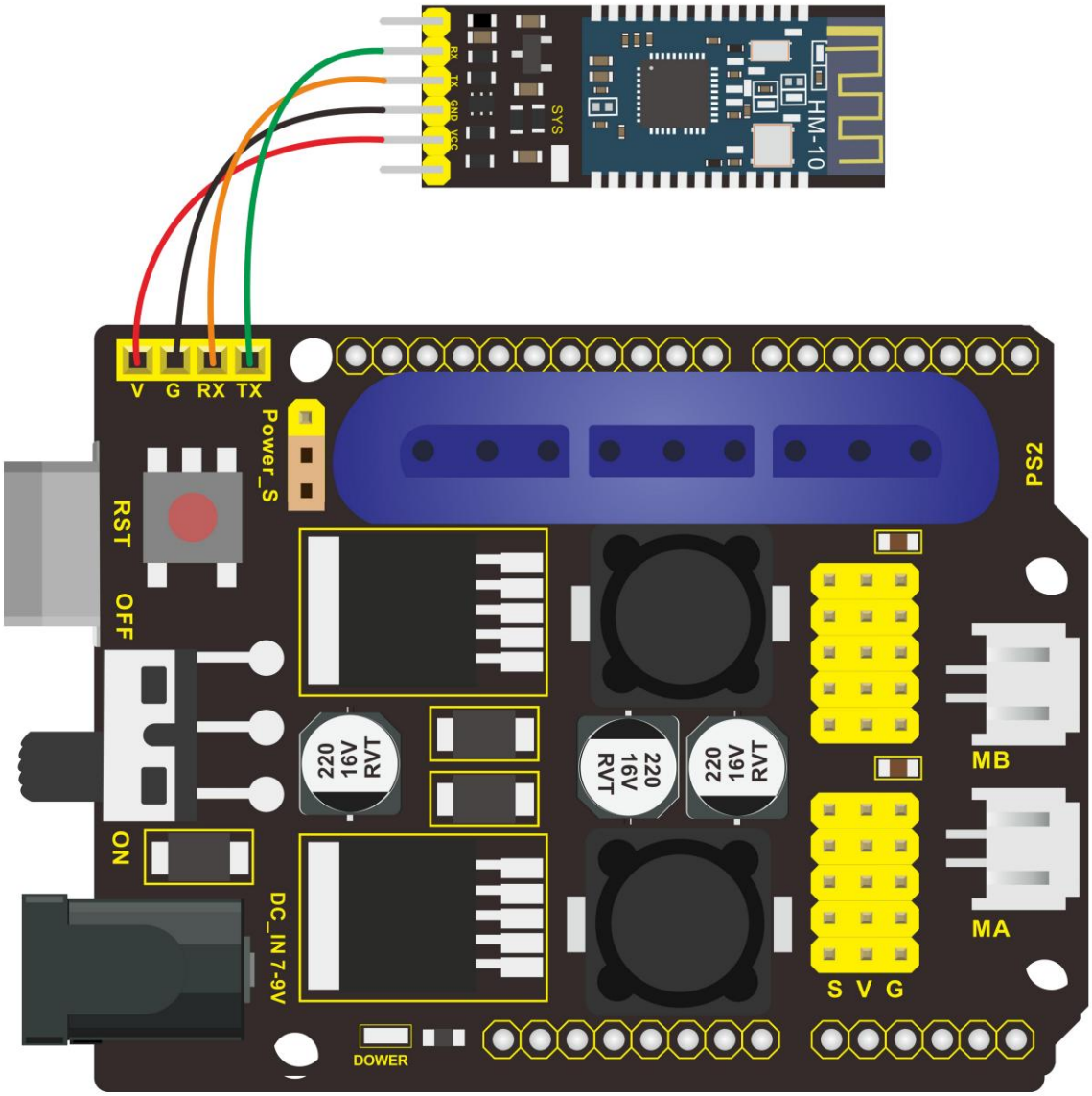


3. Upload code op de besturingskaart, na het inschakelen knippert de LED op Bluetooth moduul. Start Bluetooth en open de app om op "verbinden" te klikken, Bluetooth is verbonden.



Machine Translated by Google

Verbindingsdiagram



Testcode

```
ongeldige instellingen ()

{

   Serieel.begin(9600);          // stel de baudrate in op 9600

}
```

Machine Translated by Google




```
ongeldige lus()
{
  char val;       // definieer een variabele om de gegevens te ontvangen die door Bluetooth zijn gelezen
  if(Serial.available())       // bepalen of gegevens worden ontvangen
  {
val = Serieel.lezen();                    // de gelezen gegevens worden toegewezen aan "val"
Serieel.println(val);
  }
}
```

Nadat u de Bluetooth-module hebt aangesloten, opent u de seriële poortmonitor om de baudsnelheid in te stellen 9600. Druk op de bedieningstoetsen op de app, de seriële poort drukt het bijbehorende af controle karakter. Zoals hieronder getoond:

Machine Translated by Google

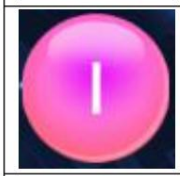
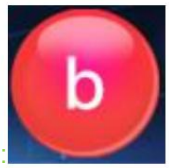


Testresultaat:

De functies van bedieningstoetsen:

CONNECT	Sluit de APP aan op de HM-10 Bluetooth-module	
DISCONNECT	Schakel Bluetooth uit	
	Druk om "F" te verzenden Laat los om "S" te verzenden	Linkerservo gaat naar voren Linkerservo stopt de beweging
	Druk om "L" te verzenden Laat los om "S" te verzenden	Klemklauw gaat open Klemklauw stopt
	Modus 1	

Machine Translated by Google

	Druk om "R" te verzenden Laat los om "S" te verzenden	Klemklauw sluit Klemklauw stopt
	Druk op om "B" te verzenden Laat los om "S" te verzenden	Linker Servo trekt zich terug Linkerservo stopt de beweging
	Druk om "f" te verzenden Laat los om "S" te verzenden	Rechterservo strekt zich uit Rechterservo stopt de beweging
	Druk om "l" te verzenden Laat los om "S" te verzenden	De basisservo draait naar links Basisservo stopt
	Modus 2	----------
	Druk om "r" te verzenden Laat los om "S" te verzenden	Base Servo draait naar rechts Basisservo stopt
	Druk om "b" te verzenden Laat los om "S" te verzenden	Rechter Servo trekt zich terug Rechterservo stopt

ÿ **Bluetooth bestuurt de robotarm**

Beschrijving

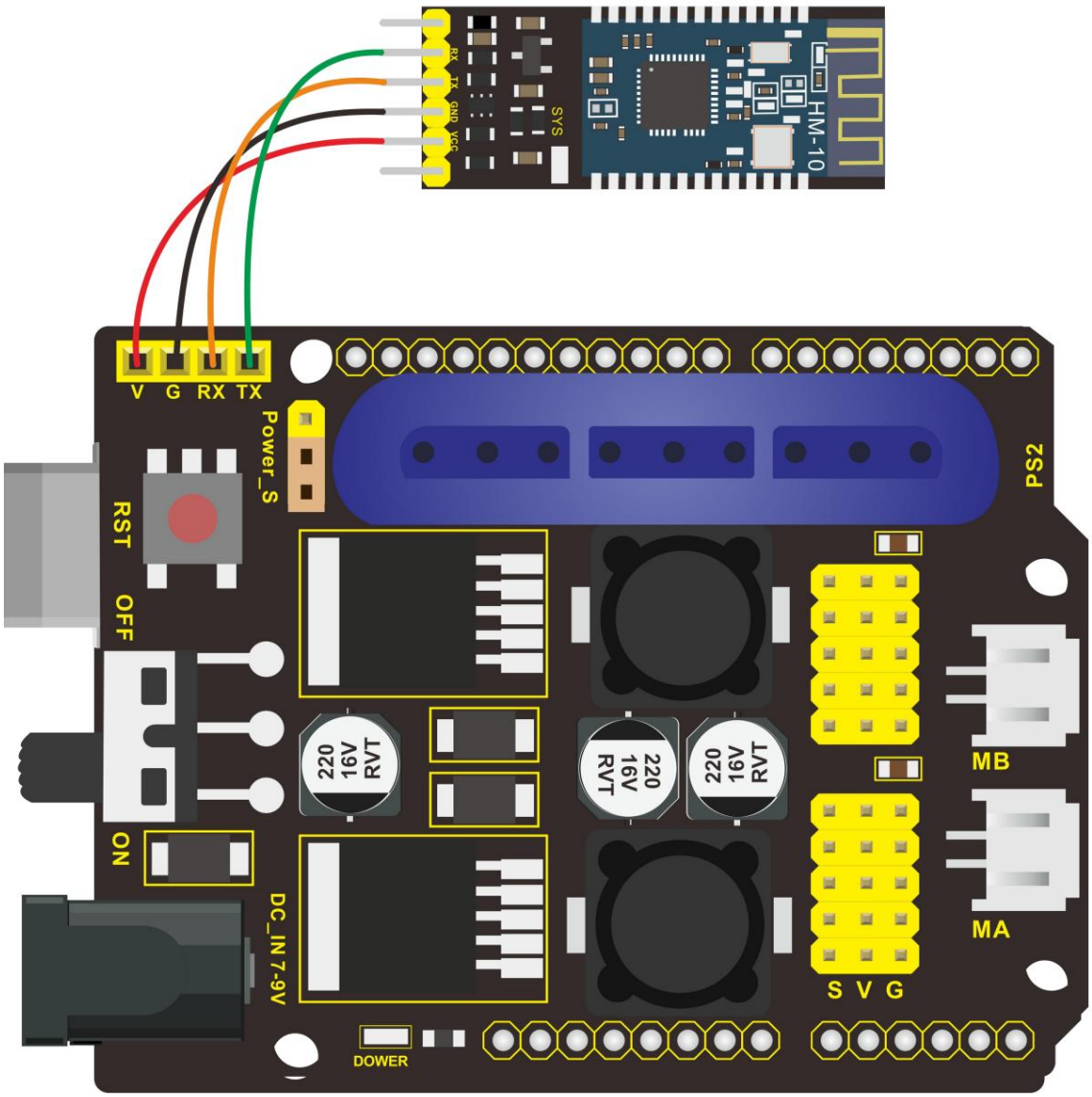
We introduceerden de besturingsmethode van de 4-DOF robotarm en HM-10 Bluetooth

Machine Translated by Google

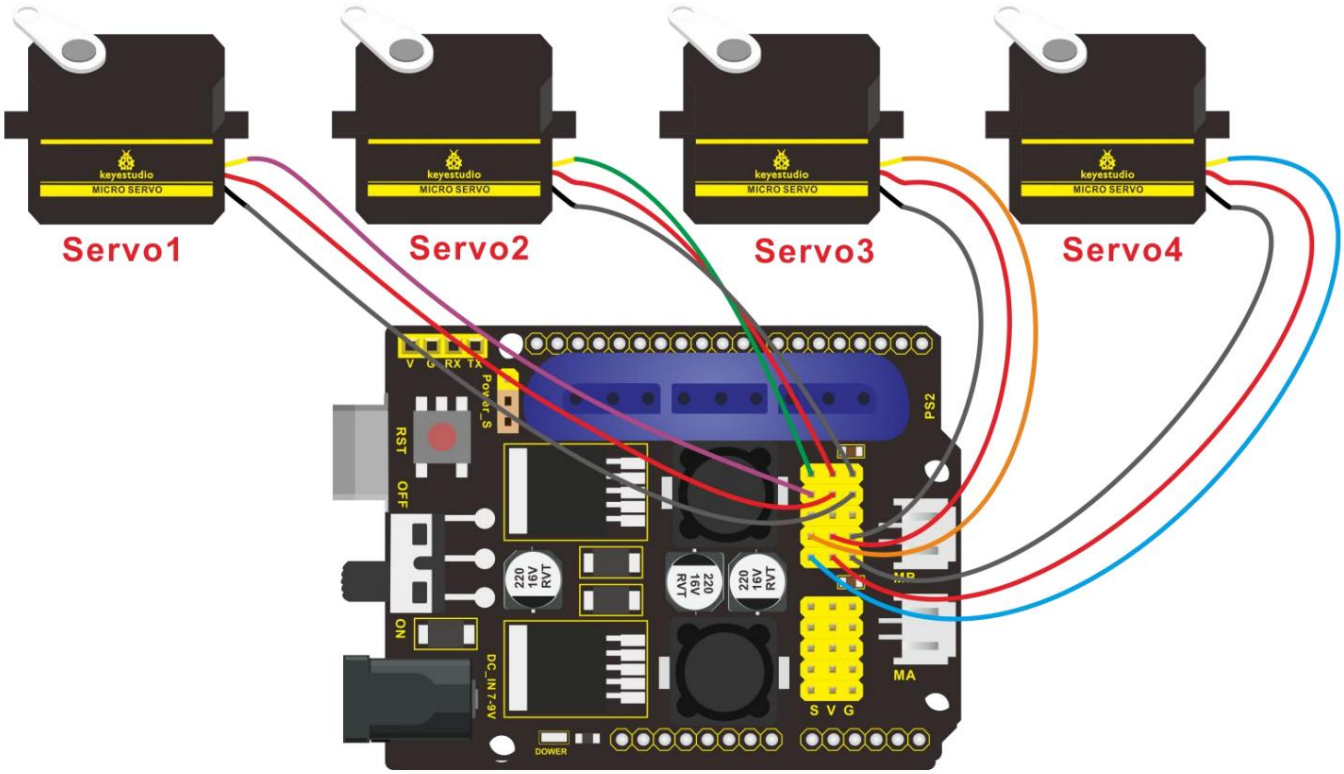
moduul. In dit experiment besturen we 4DOF robotarmbewegingen via APP.

Opmerking: Nadat het uploaden van de testcode is gelukt, koppelt u de USB-gegevenskabel los en opstarten via externe voeding en 4 DOF robotarmbewegingen besturen via APP.

Verbindingsdiagram



Machine Translated by Google



Testcode

```
#include <Servo.h> //voeg Servo-bibliotheekbestanden toe

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;

int pos1=90, pos2=60, pos3=130, pos4=0; // definieer de variabelen van 4 Servo

hoek en wijs initiële waarden toe (houdingshoekwaarden van opstarten)

char val;
```

ongeldige instellingen ()

Machine Translated by Google




```
{
    // houding van opstarten
  mijnservo1.schrijven(pos1);
  mijnservo2.schrijven(pos2);
  mijnservo3.schrijven(pos3);
  mijnservo4.schrijven(pos4);
  vertraging(1500);

  Serieel.begin(9600); // stel de baudrate in op 9600
}

ongeldige lus()
{
  mijnservo1.attach(A1); // stel de stuurpen van servo1 in op A1
  mijnservo2.attach(A0); // stel de stuurpen van servo2 in op A0
  mijnservo3.attach(8); // stel de stuurpen van servo3 in op D8
  mijnservo4.attach(9); // stel de stuurpen van servo4 in op D9

  if(Serial.available()) // bepaal of gegevens worden ontvangen
  {
    val=Serieel.lezen();          // lees de ontvangen gegevens
    Serieel.println(val);
```

Machine Translated by Google




```
    }
    schakelaar(val)
    {
        case 'L': arm_open(); pauze; // klauw gaat open
        case 'R': arm_close(); pauze; // klauw sluit
        geval 'F': left_f(); pauze; // linkerservo zwaait naar voren
        geval 'B': left_b(); pauze; //linker servo zwaait naar achteren
        geval 'f': right_f(); pauze; // rechter servo strekt zich uit
        geval 'b': right_b(); pauze; // rechter servo trekt zich terug
        geval 'l': T_links(); pauze; // basisservo draait naar links
        geval 'r': T_right(); pauze; // basisservo draait naar rechts
    }
}
//********************************************** **
//sla linksaf
leegte T_left()
{
    pos1=pos1+1;
    mijnservo1.schrijven(pos1);
    vertraging(5);
    als(pos1>180)
    {
```

Machine Translated by Google




```
            pos1=180;
        }
    }
    // rechtsaf
    leegte T_right()
    {
        pos1=pos1-1;
        mijnservo1.schrijven(pos1);
        vertraging(5);
        als(pos1<1)
        {
            pos1=1;
        }
    }
    //*****************************************
    // klauw sluit
    leegte arm_close()
    {
            pos4=pos4-1;
            Serieel.println(pos4);
            mijnservo4.schrijven(pos4);
            vertraging(5);
```

Machine Translated by Google




```
        als(pos4<0)
        {
          pos4=0;
        }
}
// klauw gaat open
leegte arm_open()
{
      pos4=pos4+1;
        Serieel.println(pos4);
        mijnservo4.schrijven(pos4);
        vertraging(5);
        als(pos4>108)
        {
          pos4=108;
        }
}

//*********************************************
// linker servo vooruit
leegte left_f()
{
```

Machine Translated by Google




```
        pos2=pos2-1;
        mijnservo2.schrijven(pos2);
        vertraging(5);
        als(pos2<0)
        {
            pos2=0;
        }
}
// linkerservo achteruit
leegte left_b()
{
        pos2=pos2+1;
        mijnservo2.schrijven(pos2);
        vertraging(5);
        als(pos2>180)
        {
            pos2=180;
        }
}

//**************************************
// rechter servo vooruit
```

Machine Translated by Google




```
nietig recht_f()
{
    pos3=pos3+1;
        mijnservo3.schrijven(pos3);
        vertraging(5);
        als(pos3>180)
        {
            pos3=180;
        }

}//rechter servo achteruit
nietig recht_b()
{
    pos3=pos3-1;
        mijnservo3.schrijven(pos3);
        vertraging(5);
        als(positie3<35)
        {
            pos3=35;
        }
}
```

Machine Translated by Google

Testresultaat:

Upload de code, sluit hem aan en zet hem aan, na het aansluiten van de Bluetooth APP, druk op de toets om de robotarm te besturen bewegingen.

Project 5: PS2-gestuurde robotarm (uitbreiding)

5.1 PS2 Joypad-toetstest

Beschrijving:

Op het aandrijfschild bevindt zich een PS2-joystickconnector, waarmee u de 4DOF-robotarm eenvoudig kunt bedienen met de PS2-joypad. Maar je moet het zelf kopen omdat de PS2 Joypad niet in de kit zit.
Wanneer u de PS2 Joypad gebruikt om de robotarm te besturen, moet u eerst het overeenkomstige karakter van elke toets op de PS2 Joypad verkrijgen.
Dit experiment helpt je dus om het karakter van elke toets op de PS2 Joypad te testen.

Na het aansluiten van de Joypad, moet het testprogramma op Arduino IDE worden geüpload. Maar voordat u gaat testen, moet u de map **PS2X_lib** in de bibliotheekmap van de Arduino IDE-directory plaatsen .
Upload de code, open de seriële monitor, sluit de PS2 Joypad aan.
Wanneer u op de toets drukt, zou u het overeenkomstige teken op de monitor moeten zien.

Machine Translated by Google

Testcode 10:

```
********************************************** **********************************

**********
```

```
#include <PS2X_lib.h> //voor v1.6


/****************************************** ****************
 * set pinnen aangesloten op PS2-controller:
 *      - 1e kolom: origineel
 *      - 2e colmun: Stef?
 * vervang pincodes door degene die u gebruikt
 ****************************************** ***************/
#define PS2_DAT        13 //14
#define PS2_CMD        11 //15
#define PS2_SEL        10 //16
#define PS2_CLK        12 //17


/****************************************** ****************
 * selecteer modi van PS2-controller:
 *      - drukken = analoge uitlezing van drukknoppen
 *      - gerommel      = geronk van de motor
 * verwijder commentaar op 1 van de regels voor elke modusselectie
```

Machine Translated by Google


```
  ********************************************************** **********/

//#define drukken waar

#definieer druk onjuist

//#define gerommel WAAR

#definieer gerommel vals


 PS2X ps2x; // maak een PS2-controllerklasse


// op dit moment ondersteunt de bibliotheek GEEN hot-pluggable controllers, wat betekent

// je moet je Arduino altijd opnieuw opstarten nadat je de controller hebt aangesloten,

//of bel config_gamepad(pins) opnieuw nadat je de controller hebt aangesloten.


int-fout = 0;

bytetype = 0;

byte trillen = 0;


ongeldige instelling(){


   Serieel.begin(57600);


   vertraging(300); // vertraging toegevoegd om de draadloze ps2-module eerder wat tijd te geven om op te starten

configureren
```

Machine Translated by Google




```
    // VERANDERINGEN voor v1.6 HIER!!! **************LET OP**************


    // stel pinnen en instellingen in: GamePad (klok, commando, aandacht, gegevens, druk?, Rumble?)
controleer op fouten
    error = ps2x.config_gamepad(PS2_CLK, PS2_CMD, PS2_SEL, PS2_DAT, druk, gerommel);


    als(fout == 0){
      Serial.print("Controller gevonden, succesvol geconfigureerd");
      Serial.print("druk = ");
  als (druk)
     Serieel.println("true");
  anders
     Serieel.println("false");
  Serial.print("rumble = ");
  als (gerommel)
     Serieel.println("waar)");
  anders
     Serieel.println("false");
      Serial.println ("Probeer alle knoppen uit, X zal de controller laten trillen, sneller als je drukt
moeilijker;");
      Serial.println("Als u L1 of R1 vasthoudt, worden de waarden van de analoge stick afgedrukt.");
```

Machine Translated by Google


```
        Serial.println ("Opmerking: ga naar www.billporter.info voor updates en om bugs te melden.");
    }
    anders als(fout == 1)
        Serial.println ("Geen controller gevonden, controleer de bedrading, zie readme.txt om foutopsporing in te schakelen. bezoek
www.billporter.info voor tips voor het oplossen van problemen");

    anders als(fout == 2)
        Serial.println("Controller gevonden maar accepteert geen commando's. zie readme.txt om in te schakelen
debuggen. Bezoek www.billporter.info voor tips voor het oplossen van problemen");

    anders als(fout == 3)
        Serial.println("Controller die weigert naar de Pressures-modus te gaan, ondersteunt dit mogelijk niet. ");

// Serieel.print(ps2x.Analog(1), HEX);

    type = ps2x.readType();
    schakelaar(type) {
        geval 0:
            Serial.print ("Onbekend controllertype gevonden");
            pauze;
        zaak 1:
            Serial.print ("DualShock-controller gevonden");
```

Machine Translated by Google




```
            pauze;
        geval 2:
            Serial.print ("GuitarHero-controller gevonden");
            pauze;
    geval 3:
            Serial.print ("Draadloze Sony DualShock-controller gevonden");
            pauze;
      }
}

ongeldige lus() {
  /* Je moet Gamepad lezen om nieuwe waarden te krijgen en trillingswaarden in te stellen
       ps2x.read_gamepad (kleine motor aan/uit, grotere motorkracht van 0-255)
       als je de rumble niet inschakelt, gebruik dan ps2x.read_gamepad(); zonder waarden
       U moet dit minstens één keer per seconde bellen
    */
  if(error == 1) //lus overslaan als er geen controller is gevonden
      opbrengst;

  if(type == 2){ //Guitar Hero-controller
      ps2x.read_gamepad();                    // lees controleur
```

Machine Translated by Google




```
if(ps2x.ButtonPressed(GREEN_FRET))
    Serial.println ("Groene Fret ingedrukt");
if(ps2x.ButtonPressed(RED_FRET))
    Serial.println ("Rode Fret ingedrukt");
if(ps2x.ButtonPressed(YELLOW_FRET))
    Serial.println ("Gele fret ingedrukt");
if(ps2x.ButtonPressed(BLUE_FRET))
    Serial.println ("Blauwe fret ingedrukt");
if(ps2x.ButtonPressed(ORANGE_FRET))
    Serial.println ("Oranje Fret geperst");


if(ps2x.ButtonPressed(STAR_POWER))
    Serial.println ("Star Power-commando");


if(ps2x.Button(UP_STRUM))                    // zal WAAR zijn zolang de knop wordt ingedrukt
    Serial.println ("Up Strum");
if(ps2x.Button(DOWN_STRUM))
    Serial.println ("DOWN Strum");


als(ps2x.Button(PSB_START))                  // zal WAAR zijn zolang de knop wordt ingedrukt
    Serial.println ("Start wordt vastgehouden");
als(ps2x.Button(PSB_SELECT))
```

Machine Translated by Google




```
            Serial.println ("Select wordt vastgehouden");



        als(ps2x.Knop(ORANGE_FRET)) {                    // print stick-waarde INDIEN WAAR
            Serial.print ("Wammy-balkpositie:");
            Serial.println(ps2x.Analog(WHAMMY_BAR), DEC);
        }
    }
    else {//DualShock-controller
        ps2x.read_gamepad(false, trillen); // lees de controller en stel de grote motor in op 'trillen'
snelheid



        als(ps2x.Button(PSB_START))                      // zal WAAR zijn zolang de knop wordt ingedrukt
            Serial.println ("Start wordt vastgehouden");
        als(ps2x.Button(PSB_SELECT))
            Serial.println ("Select wordt vastgehouden");



        als(ps2x.Button(PSB_PAD_UP)) {                   // zal WAAR zijn zolang de knop wordt ingedrukt
            Serial.print ("Boven hield dit hard vast: ");
            Serieel.println(ps2x.Analog(PSAB_PAD_UP), DEC);
        }
        als(ps2x.Button(PSB_PAD_RIGHT)){
            Serial.print ("Rechts hield dit hard vast: ");
```

Machine Translated by Google


```
        Serieel.println(ps2x.Analog(PSAB_PAD_RIGHT), DEC);
      }
      if(ps2x.Button(PSB_PAD_LEFT)){
        Serial.print("LEFT hield dit hard vast: ");
        Serieel.println(ps2x.Analog(PSAB_PAD_LEFT), DEC);
      }
      if(ps2x.Button(PSB_PAD_DOWN)){
        Serial.print ("DOWN hield dit hard vast: ");
        Serieel.println(ps2x.Analog(PSAB_PAD_DOWN), DEC);
      }

      trillen = ps2x.Analog(PSAB_CROSS); // hiermee wordt de trilsnelheid van de grote motor ingesteld op basis van
hoe hard u op de blauwe (X) knop drukt
      als (ps2x.NewButtonState()) {                    //zal WAAR zijn als een knop van status verandert (aan naar uit, of
uit naar aan)
        if(ps2x.Button(PSB_L3))
          Serial.println ("L3 ingedrukt");
        als(ps2x.Button(PSB_R3))
          Serial.println ("R3 ingedrukt");
        if(ps2x.Button(PSB_L2))
          Serial.println ("L2 ingedrukt");
        if(ps2x.Button(PSB_R2))
```

Machine Translated by Google




```
            Serial.println ("R2 ingedrukt");
        if(ps2x.Button(PSB_TRIANGLE))
            Serial.println ("Driehoek ingedrukt");
    }

    if(ps2x.ButtonPressed(PSB_CIRCLE))                  // zal WAAR zijn als de knop ENKEL is ingedrukt
        Serial.println ("Cirkel zojuist ingedrukt");
    if(ps2x.NewButtonState(PSB_CROSS))                  // zal TRUE zijn als de knop JUST was
ingedrukt OF losgelaten
        Serial.println ("X is zojuist gewijzigd");
    if(ps2x.ButtonReleased(PSB_SQUARE))                 // zal TRUE zijn als de knop JUST was
uitgegeven
        Serial.println ("Vierkant is zojuist vrijgegeven");

    if(ps2x.Button(PSB_L1) || ps2x.Button(PSB_R1)) {//print stick-waarden als een van beide TRUE is
        Serial.print ("Stick-waarden:");
        Serial.print(ps2x.Analog(PSS_LY), DEC); // Linker joystick, Y-as. Andere opties: LX, RY, RX
        Serieel.print(",");
        Serial.print(ps2x.Analog(PSS_LX), DEC);
        Serieel.print(",");
        Serial.print(ps2x.Analog(PSS_RY), DEC);
        Serieel.print(",");
```

Machine Translated by Google




```
            Serieel.println(ps2x.Analog(PSS_RX), DEC);
          }
        }
        vertraging(50);
    }
```

********** **********

****

Testresultaat:

Stapel het schijfschild op V4.0 en upload de code. Sluit de PS2 Joypad aan, open de seriële monitor en stel de baudsnelheid in op 57600. Wanneer u de toets indrukt of op de rocker drukt, zou u het overeenkomstige teken op de monitor moeten zien.

5.2 PS2 Joypad-bediening

Beschrijving:

In het vorige gedeelte hebben we laten zien hoe de joystickmodule kan worden gebruikt om de robotarm te besturen. Het is voor jou bijna hetzelfde om de 4DOF robotarm te besturen met de PS2 Joypad.

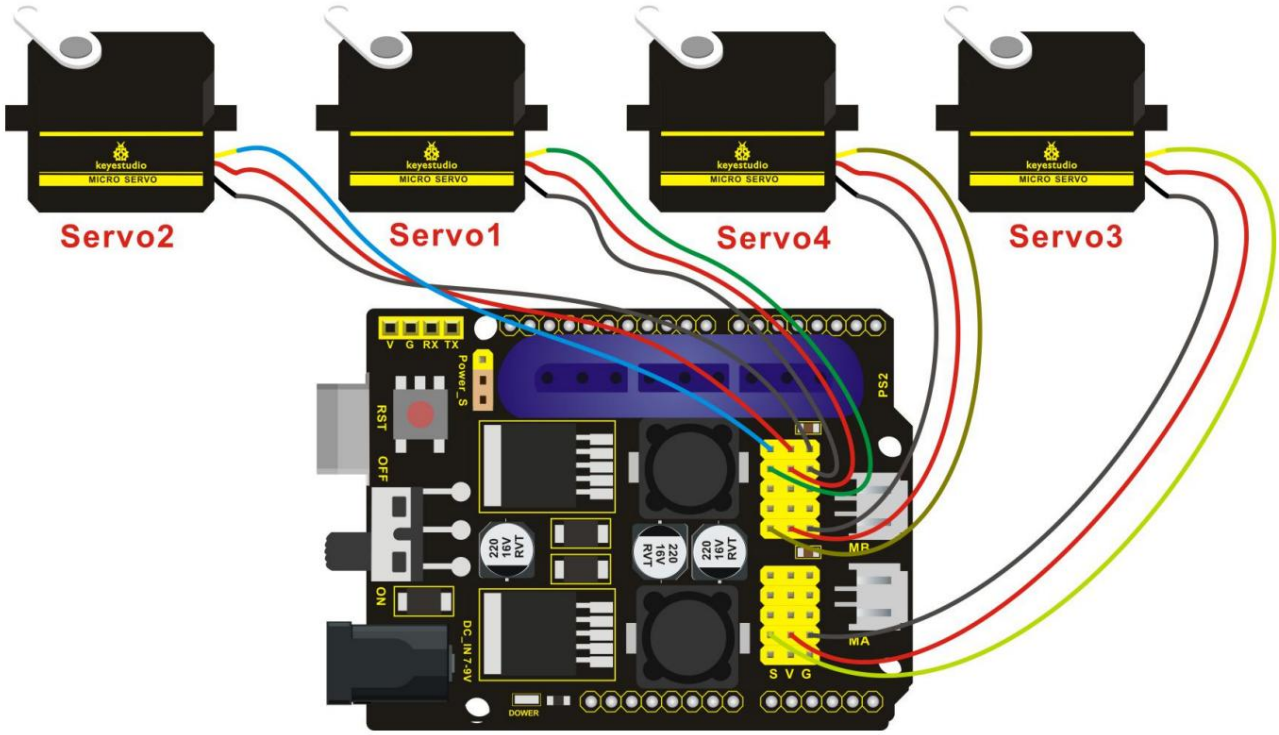
PS2-joystickbediening

Machine Translated by Google

Rechter joystick Robotarm Linker joystick			**Robotachtige klauw**
Duw naar de rechter zijde	Draai naar rechts	Duw naar de rechter zijde	dichtbij
Duw naar de linkerkant	Draai naar links	Duw naar de linkerkant	open
Naar voren te duwen	De kleine arm gaat omhoog	Naar voren te duwen	De grote arm zal vooruit gaan
Terugduwen	De kleine arm zal zakken	Terugduwen	De grote arm zal ga terug

Machine Translated by Google

Verbindingsdiagram



Testcode

```
#include <PS2X_lib.h>


PS2X ps2x; // maak een PS2-controllerklasse




// op dit moment ondersteunt de bibliotheek GEEN hot-pluggable controllers, wat betekent

// je moet je Arduino altijd opnieuw opstarten nadat je de controller hebt aangesloten,

//of bel config_gamepad(pins) opnieuw nadat je de controller hebt aangesloten.

int-fout = 0;
```

Machine Translated by Google


```
byte trillen = 0;



#include <Servo.h> // voeg de servobibliotheken toe

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;



int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoek en wijs de

initiële waarde (dat is de waarde van de houdingshoek van de laars)



ongeldige instelling(){

  Serieel.begin(57600);



// laars houding

    mijnservo1.schrijven(pos1);

    vertraging(1000);

    mijnservo2.schrijven(pos2);

    mijnservo3.schrijven(pos3);

    mijnservo4.schrijven(pos4);

    vertraging(1500);
```

Machine Translated by Google


```
  fout = ps2x.config_gamepad(13,11,10,12);                    // stel GamePad in (klok, commando, aandacht,
data) pinnen, controleer op fouten

  als(fout == 0){
    Serial.println("Controller gevonden, succesvol geconfigureerd");
    Serial.println ("Probeer alle knoppen uit, X zal de controller laten trillen, sneller als je harder drukt;");
   Serial.println("Als u L1 of R1 vasthoudt, worden de waarden van de analoge stick afgedrukt.");
   Serial.println ("Ga naar www.billporter.info voor updates en om bugs te melden.");
  }

   anders als(fout == 1)
    Serial.println ("Geen controller gevonden, controleer de bedrading, zie readme.txt om foutopsporing in te schakelen. bezoek
www.billporter.info voor tips voor het oplossen van problemen");

   anders als(fout == 2)
    Serial.println("Controller gevonden maar accepteert geen commando's. zie readme.txt om foutopsporing in te schakelen.
Bezoek www.billporter.info voor tips voor het oplossen van problemen");

    //Serial.print(ps2x.Analog(1), HEX);

  ps2x.enableRumble();                          // schakel triltrillingsmotoren in
```

Machine Translated by Google


```
  ps2x.inschakelenDruk();                    // schakel het lezen van de drukwaarden van de knoppen in.




}


ongeldige lus(){
    /* Je moet Gamepad lezen om nieuwe waarden te krijgen
     Lees GamePad en stel trillingswaarden in
     ps2x.read_gamepad (kleine motor aan/uit, grotere motorkracht van 0-255)
     als je de rumble niet inschakelt, gebruik dan ps2x.read_gamepad(); zonder waarden

     u moet dit minstens één keer per seconde bellen
     */

   mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1
   mijnservo2.attach(A0); // stel de stuurpen van servo 2 in op A0
   mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
   mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9

  als(fout != 0)
   opbrengst;
```

Machine Translated by Google


```
    ps2x.read_gamepad(false, trillen);                    // lees de controller en stel de grote motor in om te draaien
snelheid 'trillen'


   als(ps2x.Button(PSB_START))                            // zal WAAR zijn zolang de knop wordt ingedrukt
         Serial.println ("Start wordt vastgehouden");
   als(ps2x.Button(PSB_SELECT))
         Serial.println ("Select wordt vastgehouden");



    als(ps2x.Button(PSB_PAD_UP)) {                        // zal WAAR zijn zolang de knop wordt ingedrukt
      Serial.print ("Boven hield dit hard vast: ");
      Serieel.println(ps2x.Analog(PSAB_PAD_UP), DEC);
    }
     als(ps2x.Button(PSB_PAD_RIGHT)){
      Serial.print ("Rechts hield dit hard vast: ");
        Serieel.println(ps2x.Analog(PSAB_PAD_RIGHT), DEC);
    }
     if(ps2x.Button(PSB_PAD_LEFT)){
      Serial.print("LEFT hield dit hard vast: ");
        Serieel.println(ps2x.Analog(PSAB_PAD_LEFT), DEC);
    }
```

Machine Translated by Google


```
        if(ps2x.Button(PSB_PAD_DOWN)){
          Serial.print ("DOWN hield dit hard vast: ");
       Serieel.println(ps2x.Analog(PSAB_PAD_DOWN), DEC);
        }



        trillen = ps2x.Analog(PSAB_BLUE);                    // dit stelt de trilsnelheid van de grote motor in
gebaseerd op
                                                              //hoe hard je op de blauwe (X) knop drukt

    als (ps2x.NewButtonState())                          // zal WAAR zijn als een knop van status verandert (aan naar uit,
of uit naar aan)
    {
          als(ps2x.Button(PSB_R3))

            Serial.println ("R3 ingedrukt");



          if(ps2x.Button(PSB_L3))
            Serial.println ("L3 ingedrukt");

          if(ps2x.Button(PSB_L2))
```

Machine Translated by Google




```
            Serial.println ("L2 ingedrukt");
          if(ps2x.Button(PSB_R2))
            Serial.println ("R2 ingedrukt");
          if(ps2x.Button(PSB_GREEN))
            Serial.println ("Driehoek ingedrukt");

    }

    if(ps2x.ButtonPressed(PSB_RED))                    // zal WAAR zijn als de knop ENKEL is ingedrukt
            Serial.println ("Cirkel zojuist ingedrukt");

    if(ps2x.ButtonReleased(PSB_PINK))                  // zal WAAR zijn als de knop NET is losgelaten
            Serial.println ("Vierkant is zojuist vrijgegeven");

    if(ps2x.NewButtonState(PSB_BLUE))                  // zal WAAR zijn als de knop ENKEL is ingedrukt OF
uitgegeven
            Serial.println ("X is zojuist gewijzigd");

      // rotatie
      zhuandong();
      //klauw
```

Machine Translated by Google




```
        zhuazi();
        //de grotere arm
        dat();
        //de kleinere arm
        xiaobi();

    if(ps2x.Button(PSB_L1) || ps2x.Button(PSB_R1)) // print stick-waarden als een van beide TRUE is
    {
        Serial.print ("Stick-waarden:");
        Serial.print(ps2x.Analog(PSS_LY), DEC); // Linker joystick, Y-as. Andere opties: LX, RY, RX
        Serieel.print(",");
        Serial.print(ps2x.Analog(PSS_LX), DEC);
        Serieel.print(",");
        Serial.print(ps2x.Analog(PSS_RY), DEC);
        Serieel.print(",");
        Serieel.println(ps2x.Analog(PSS_RX), DEC);
    }
    vertraging(5);
}
//********************************************** ********************
// draai
leegte zhuandong()
```

Machine Translated by Google




```
{
// rechtsaf
    if(ps2x.Analog (PSS_RX) > 200) // if duw de rechter joystick naar rechts
     {
        //Serial.println(ps2x.Analog(PSS_RX), DEC);
        pos1=pos1-1; //pos1 trekt 1 af
        mijnservo1.schrijven(pos1); // servo 1 voert de actie uit, de arm zal naar rechts draaien.
       // vertraging(5);
        if(pos1<1) // beperk de juiste draaihoek
        {
            pos1=1;
        }
      }
       // sla linksaf
      if(ps2x.Analoog (PSS_RX) < 50)              // als duw de rechter joystick naar links
      {
         //Serial.println(ps2x.Analog(PSS_RX), DEC);
         pos1=pos1+1; // pos1 plus 1
         mijnservo1.schrijven(pos1);          // de arm draait naar links
        // vertraging(5);
         als(pos1>180)             // beperk de linker draaihoek
         {
```

Machine Translated by Google




```
                pos1=180;
            }
        }
    }
    //************************************************** **********************
    // bovenarm
leegte xiaobi()
{
    // bovenarm voorzijde
      if(ps2x.Analog(PSS_RY)<50) // if duw de rechter joystick omhoog
      {
          pos2=pos2-1;
          mijnservo2.schrijven(pos2);        // de bovenarm gaat omhoog
          vertraging(5);
          als(pos2<0)          // beperk de hefhoek
          {
              pos2=0;
          }
      }
      // bovenarm naar achteren
       if(ps2x.Analog(PSS_RY)>200)               // als duw de rechter joystick naar beneden
       {
```

Machine Translated by Google


```
            pos2=pos2+1;
            mijnservo2.schrijven(pos2); // de bovenarm gaat naar beneden
            vertraging(5);
            if(pos2>180) // beperk de neerwaartse hoek
            {
                pos2=180;
            }
        }
}
//************************************************** **********
leegte zhuazi()
{
// sluit de klauw
      if(ps2x.Analog(PSS_LX)>220) // if duw de linker joystick naar rechts
     {
            pos4=pos4-1;
            Serieel.println(pos4);
            mijnservo4.schrijven(pos4); // servo 4 voert de actie uit en de klauw wordt geleidelijk gesloten.
            vertraging(5);
            als(pos4<0)         // als de pos4-waarde wordt afgetrokken tot 37, is de klauw in 37 graden die we hebben getest
gesloten.)
            {
```

Machine Translated by Google




```
            pos4=0;
          }
        }
        // open de klauw
        if(ps2x.Analog(PSS_LX)<10) // if duw de linker joystick naar links
        {
          pos4=pos4+8;
          Serieel.println(pos4);
          mijnservo4.schrijven(pos4); // servo 4 voert de actie uit en de klauw wordt geleidelijk geopend
          vertraging(5);
          als(pos4>108)        // beperk de maximale openingshoek
          {
            pos4=108;
          }
        }
}
//************************************************** *********
leegte dabi()
{
   // onderarm voorzijde
    if(ps2x.Analog(PSS_LY)>200) // if duw de linker joystick omhoog
    {
```

Machine Translated by Google


```
        pos3=pos3+1;
        mijnservo3.schrijven(pos3); // de onderarm strekt zich uit
        vertraging(5);
        if(pos3>180) // beperk de uitgerekte hoek
        {
            pos3=180;
        }
      }
    als(ps2x.Analoog(PSS_LY)<10)        // als duw de linker joystick naar beneden
    {
        pos3=pos3-1;
        mijnservo3.schrijven(pos3); // de onderarm trekt zich terug
        vertraging(5);
        als(positie3<35)        // beperk de ingetrokken hoek
        {
            pos3=35;
        }
      }
  }
```

****

Machine Translated by Google

Testresultaat

Bestuur de robotarm met de PS2-joypad.

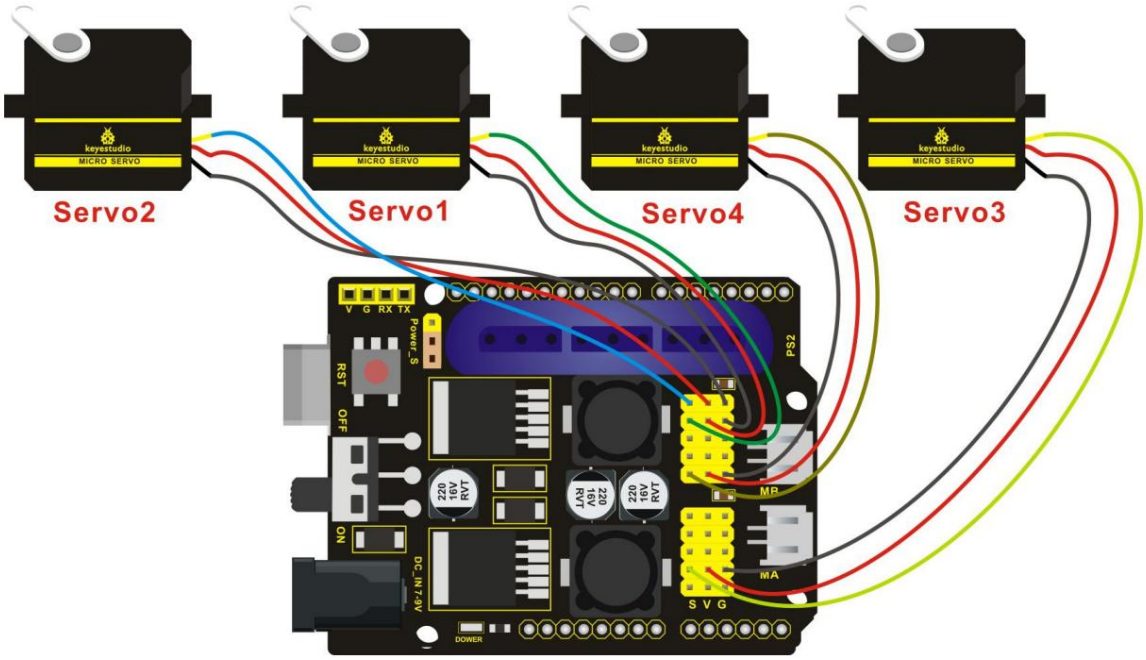
5.3 PS2 Controle van het houdingsgeheugen

Beschrijving

In het vorige experiment hebben we laten zien hoe je de joystickmodule kunt gebruiken om verschillende houdingen te onthouden. Nu vervangen we de joystickmodule door PS2 Joypad. Druk op de Z1-knop om de houdingen van de arm te onthouden. Als u meer wilt onthouden, kunt u de code invoeren. Druk daarna op de Z2-knop om de opgeslagen acties van de arm uit te voeren

Verbindingsdiagram

Machine Translated by Google



Testcode12

```
#include <PS2X_lib.h>

#include <Servo.h> // voeg de servobibliotheken toe

Servo mijnservo1; // maak een servo-object om een servo te besturen

Servo mijnservo2;

Servo mijnservo3;

Servo mijnservo4;



int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoek en wijs de

initiële waarde (dat is de waarde van de houdingshoek van de laars)



PS2X ps2x; // maak een PS2-controllerklasse



// op dit moment ondersteunt de bibliotheek GEEN hot-pluggable controllers, wat betekent
```

Machine Translated by Google




```
// je moet je Arduino altijd opnieuw opstarten nadat je de controller hebt aangesloten,
//of bel config_gamepad(pins) opnieuw nadat je de controller hebt aangesloten.
int-fout = 0;
byte trillen = 0;

int s1,s2,s3,s4;
int jiyi1[20]; // definieer vier arrays, afzonderlijk gebruikt om de hoek van 4 servo's op te slaan.
int jiyi2[20];
int jiyi3[20];
int jiyi4[20];
int i=0;
int j=0;

ongeldige instellingen ()
{
  Serieel.begin(57600);

  // laars houding
    mijnservo1.schrijven(pos1);
    vertraging(1000);
    mijnservo2.schrijven(pos2);
    mijnservo3.schrijven(pos3);
```

Machine Translated by Google


```
    mijnservo4.schrijven(pos4);
    vertraging(1500);



  fout = ps2x.config_gamepad(13,11,10,12);                    // stel GamePad in (klok, commando, aandacht,
data) pinnen, controleer op fouten

  als(fout == 0){
    Serial.println("Controller gevonden, succesvol geconfigureerd");
    Serial.println ("Probeer alle knoppen uit, X zal de controller laten trillen, sneller als je harder drukt;");
    Serial.println("Als u L1 of R1 vasthoudt, worden de waarden van de analoge stick afgedrukt.");
    Serial.println ("Ga naar www.billporter.info voor updates en om bugs te melden.");
  }

    anders als(fout == 1)
    Serial.println ("Geen controller gevonden, controleer de bedrading, zie readme.txt om foutopsporing in te schakelen. bezoek
www.billporter.info voor tips voor het oplossen van problemen");

    anders als(fout == 2)
    Serial.println("Controller gevonden maar accepteert geen commando's. zie readme.txt om foutopsporing in te schakelen.
Bezoek www.billporter.info voor tips voor het oplossen van problemen");
```

Machine Translated by Google




```
      //Serial.print(ps2x.Analog(1), HEX);



  ps2x.enableRumble();                    // schakel triltrillingsmotoren in
  ps2x.inschakelenDruk();                 // schakel het lezen van de drukwaarden van de knoppen in.




}


ongeldige lus()
{
    mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1
    mijnservo2.attach(A0); // stel de stuurpen van servo 2 in op A0
    mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
    mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9

  als(fout != 0)
    opbrengst;

    ps2x.read_gamepad(false, trillen);                    // lees de controller en stel de grote motor in om te draaien
snelheid 'trillen'
```

Machine Translated by Google


```
als(ps2x.Button(PSB_START))                    // zal WAAR zijn zolang de knop wordt ingedrukt
      Serial.println ("Start wordt vastgehouden");
als(ps2x.Button(PSB_SELECT))
      Serial.println ("Select wordt vastgehouden");



 als(ps2x.Button(PSB_PAD_UP)) {              // zal WAAR zijn zolang de knop wordt ingedrukt
   Serial.print ("Boven hield dit hard vast: ");
   Serieel.println(ps2x.Analog(PSAB_PAD_UP), DEC);
  }
  als(ps2x.Button(PSB_PAD_RIGHT)){
   Serial.print ("Rechts hield dit hard vast: ");
     Serieel.println(ps2x.Analog(PSAB_PAD_RIGHT), DEC);
  }
  if(ps2x.Button(PSB_PAD_LEFT)){
   Serial.print("LEFT hield dit hard vast: ");
     Serieel.println(ps2x.Analog(PSAB_PAD_LEFT), DEC);
  }
  if(ps2x.Button(PSB_PAD_DOWN)){
   Serial.print ("DOWN hield dit hard vast: ");
Serieel.println(ps2x.Analog(PSAB_PAD_DOWN), DEC);
```

Machine Translated by Google


```
        }



        trillen = ps2x.Analog(PSAB_BLUE);                    // dit stelt de trilsnelheid van de grote motor in
gebaseerd op
                                                              //hoe hard je op de blauwe (X) knop drukt


    als (ps2x.NewButtonState())                              // zal WAAR zijn als een knop van status verandert (aan naar uit,
of uit naar aan)
    {
            als(ps2x.Button(PSB_R3))
            {
                //Serial.println ("R3 ingedrukt");
                // dossier
                s1=mijnservo1.lezen();
                vertraging(100);
                Serieel.println(s1);
                s2=mijnservo2.lezen();
                vertraging(100);
                Serieel.println(s2);
                s3=mijnservo3.lezen();
                vertraging(100);
```

Machine Translated by Google




```
        Serieel.println(s3);
        s4=mijnservo4.lezen();
        vertraging(100);
        Serieel.println(s4);


        jiyi1[i]=s1; // sla de servowaarde opeenvolgend in de array op
        jiyi2[i]=s2;
        jiyi3[i]=s3;
        jiyi4[i]=s4;
        ik++;
        j=ik;
      // vertraging(100);
       Serieel.println(i);
     }


     if(ps2x.Button(PSB_L3))
     {
       //Serial.println("L3 ingedrukt");
       ik=0;
// ÿÿ
      pos1 = mijnservo1.lezen();
      pos2 = mijnservo2.lezen();
```

Machine Translated by Google


```
      pos3 = mijnservo3.lezen();
      pos4 = mijnservo4.lezen();

    for(int k=0;k<j;k++) //for lus, om alle opgeslagen acties uit te voeren
    {
      if(pos1<jiyi1[k]) //als de huidige hoek van servo 1 kleiner is dan de waarde die is opgeslagen in array 1.
      {
        while(pos1<jiyi1[k]) //while lus, laat de servo draaien naar de positie van de waarde die is opgeslagen in
de reeks.
        {
          mijnservo1.schrijven(pos1); // servo 1 voert de actie uit
          vertraging(5); // vertraging 5ms, regeling van de rotatiesnelheid van de servo
          pos1++;
          //Serial.println(pos1);
        }
      }
      else // als de huidige hoek van servo 1 groter is dan de waarde die is opgeslagen in array 1.
      {
        while(pos1>jiyi1[k]) //while lus, laat de servo draaien naar de positie van de waarde die is opgeslagen in
de reeks.
        {
          mijnservo1.schrijven(pos1); // servo 1 voert de actie uit
```

Machine Translated by Google


```
                vertraging(5);        // vertraging 5 ms, regeling van de rotatiesnelheid van de servo
                pos1--;
                //Serial.println(pos1);
            }

//*************************************************
// dezelfde analyse als de vorige servo
            als(pos2<jiyi2[k])
            {
              terwijl(pos2<jiyi2[k])
              {
                mijnservo2.schrijven(pos2);
                vertraging(5);
                pos2++;
                //Serial.println(pos1);
              }
            }
            anders
            {
             terwijl(pos2>jiyi2[k])
              {
                mijnservo2.schrijven(pos2);
```

Machine Translated by Google




```
                vertraging(5);
                pos2--;
                //Serial.println(pos1);
            }
        }
//************************************************** *****
//dezelfde analyse
        als(pos3<jiyi3[k])
        {
            terwijl(pos3<jiyi3[k])
            {
                mijnservo3.schrijven(pos3);
                vertraging(5);
                pos3++;
                //Serial.println(pos1);
            }
        }
        anders
        {
          terwijl(pos3>jiyi3[k])
            {
                mijnservo3.schrijven(pos3);
```

Machine Translated by Google




```
                vertraging(5);
                pos3--;
                //Serial.println(pos1);
            }
        }
//************************************************** *****
//dezelfde analyse
        als(pos4<jiyi4[k])
        {
            terwijl(pos4<jiyi4[k])
            {
                mijnservo4.schrijven(pos4);
                vertraging(5);
                pos4++;
                //Serial.println(pos1);
            }
        }
        anders
        {
          terwijl(pos4>jiyi4[k])
            {
                mijnservo4.schrijven(pos4);
```

Machine Translated by Google


```
                vertraging(5);
                pos4--;
                //Serial.println(pos1);
            }
          }
        }
      }
    }
```

```
    if(ps2x.Button(PSB_L2))
      Serial.println ("L2 ingedrukt");
    if(ps2x.Button(PSB_R2))
      Serial.println ("R2 ingedrukt");
    if(ps2x.Button(PSB_GREEN))
      Serial.println ("Driehoek ingedrukt");

}
```

```
 if(ps2x.ButtonPressed(PSB_RED))                    // zal WAAR zijn als de knop ENKEL is ingedrukt
      Serial.println ("Cirkel zojuist ingedrukt");
```

Machine Translated by Google


```
    if(ps2x.ButtonReleased(PSB_PINK))                    // zal WAAR zijn als de knop NET is losgelaten
          Serial.println ("Vierkant is zojuist vrijgegeven");

    if(ps2x.NewButtonState(PSB_BLUE))                    // zal WAAR zijn als de knop ENKEL is ingedrukt OF
uitgegeven
          Serial.println ("X is zojuist gewijzigd");

      // draai
      zhuandong();
      // klauw
      zhuazi();
      // onderarm
      dat();
      // bovenarm
      xiaobi();

    if(ps2x.Button(PSB_L1) || ps2x.Button(PSB_R1)) // print stick-waarden als een van beide TRUE is
    {
          Serial.print ("Stick-waarden:");
          Serial.print(ps2x.Analog(PSS_LY), DEC); // Linker joystick, Y-as. Andere opties: LX, RY, RX
          Serieel.print(",");
```

Machine Translated by Google


```
            Serial.print(ps2x.Analog(PSS_LX), DEC);
            Serieel.print(",");
            Serial.print(ps2x.Analog(PSS_RY), DEC);
            Serieel.print(",");
            Serieel.println(ps2x.Analog(PSS_RX), DEC);
    }

  vertraging(5);
}

//********************************************** ********************
// draai
leegte zhuandong()
{
// rechtsaf
     if(ps2x.Analog (PSS_RX) > 200) // if duw de rechter joystick naar rechts
      {
         //Serial.println(ps2x.Analog(PSS_RX), DEC);
          pos1=pos1-1; //pos1 trekt 1 af
          mijnservo1.schrijven(pos1); // servo 1 voert de actie uit en de arm zal naar rechts draaien
         // vertraging(5);
          if(pos1<1) // beperk de juiste draaihoek
```

Machine Translated by Google




```
            {
                pos1=1;
            }
        }
        //sla linksaf
        if(ps2x.Analoog (PSS_RX) < 50)          // als duw de rechter joystick naar links
        {
            //Serial.println(ps2x.Analog(PSS_RX), DEC);
            pos1=pos1+1; // pos1 plus 1
            mijnservo1.schrijven(pos1);          // de arm zal naar links draaien
            // vertraging(5);
            als(pos1>180)          // beperk de linker draaihoek
            {
                pos1=180;
            }
        }
    }
    //********************************************** **********************
    // bovenarm
leegte xiaobi()
{
    // bovenarm voorzijde
```

Machine Translated by Google




```
        if(ps2x.Analog(PSS_RY)<50) // if duw de rechter joystick omhoog
        {
            pos2=pos2-1;
            mijnservo2.schrijven(pos2);        // de bovenarm gaat omhoog
            vertraging(5);
            als(pos2<0)          // beperk de hefhoek
            {
                pos2=0;
            }
        }
        // bovenarm naar achteren
        if(ps2x.Analog(PSS_RY)>200)                // als duw de rechter joystick naar beneden
        {
            pos2=pos2+1;
            mijnservo2.schrijven(pos2); // de bovenarm gaat naar beneden
            vertraging(5);
            if(pos2>180) // beperk de neerwaartse hoek
            {
                pos2=180;
            }
        }
}
```

Machine Translated by Google


```
//************************************************ ***************

leegte zhuazi()

{

// sluit de klauw

    if(ps2x.Analog(PSS_LX)>220) // if duw de linker joystick naar rechts

    {

        pos4=pos4-1;

        Serieel.println(pos4);

        mijnservo4.schrijven(pos4); // servo 4 voert de actie uit en de klauw wordt geleidelijk gesloten.

        vertraging(5);

        if(pos4<0) // if pos4 waarde reduceert tot 37ÿde klauw die we testen in 37graden is gesloteny

        {

            pos4=0;

        }

    }

    // open de klauw

    if(ps2x.Analog(PSS_LX)<10) // if duw de linker joystick naar links

    {

        pos4=pos4+8;

        Serieel.println(pos4);

        mijnservo4.schrijven(pos4); // servo 4 voert de actie uit en de klauw wordt geleidelijk geopend

        vertraging(5);
```

Machine Translated by Google




```
            als(pos4>108)        // beperk de maximale geopende hoek
            {
                pos4=108;
            }
        }
}
//************************************************** *********
leegte dabi()
{
   // onderarm voorzijde
    if(ps2x.Analog(PSS_LY)>200) // if duw de linker joystick omhoog
    {
        pos3=pos3+1;
        mijnservo3.schrijven(pos3); // de onderarm strekt zich uit
        vertraging(5);
        if(pos3>180) // beperk de uitgerekte hoek
        {
            pos3=180;
        }
     }
    als(ps2x.Analoog(PSS_LY)<10)        // als duw de linker joystick naar beneden
    {
```

Machine Translated by Google


```
            pos3=pos3-1;
            mijnservo3.schrijven(pos3); // de onderarm zal intrekken
            vertraging(5);
            als(positie3<35)        // beperk de ingetrokken hoek
            {
               pos3=35;
            }
          }
    }
```

****

****

Testresultaat

Beweeg de PS2-tuimelschakelaar om de robotarm te besturen, druk op de knop van de PS2-tuimelschakelaar om noteer zijn houding, beweeg de joystick opnieuw en druk op de knop van de PS2 rocker om de houding te onthouden. Druk vervolgens op de linkerknop van de PS2-joypad om houdingen uitvoeren.

Machine Translated by Google

5.4 PS2 regelt houdingsgeheugen en lus

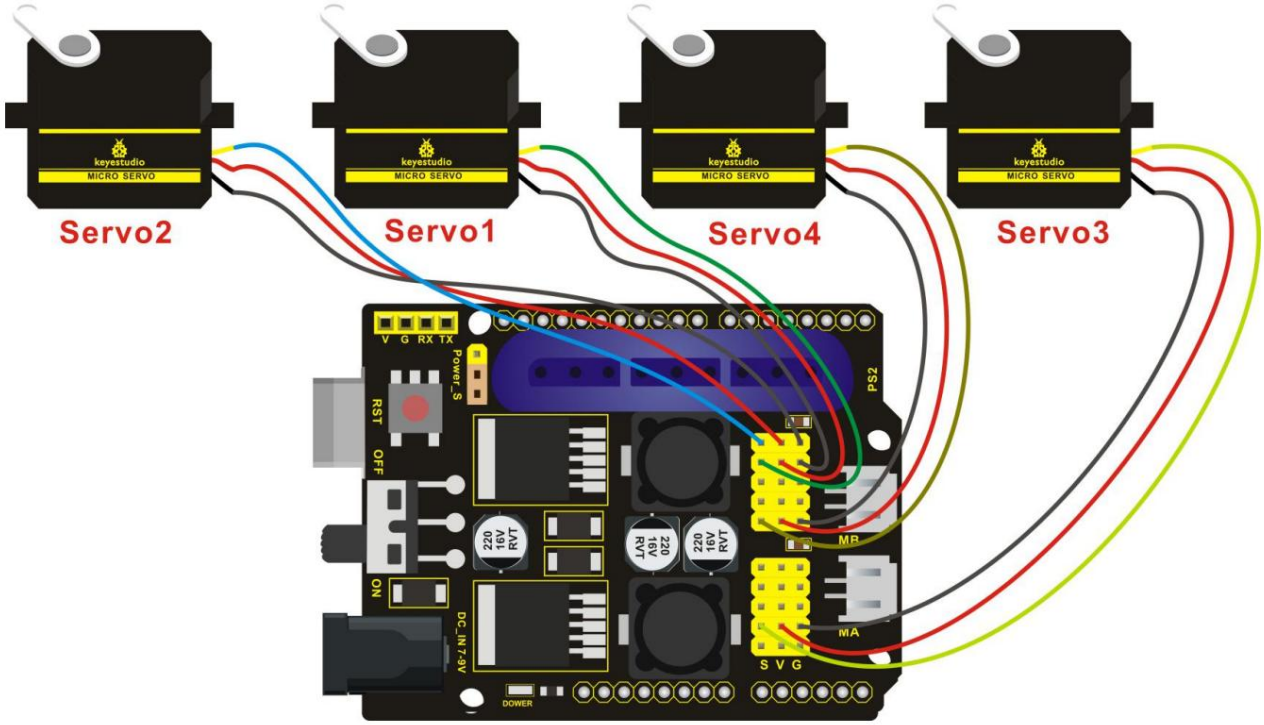
Beschrijving:

In het vorige experiment hebben we laten zien hoe de joystickmodule kan worden gebruikt om de robotarm te besturen en verschillende houdingen en loops te onthouden. Nu vervangen we de Joystick-module door de PS2 Joypad. Het programma is bijna hetzelfde.

Beweeg de thumbsticks en druk dan op de knop van de rechter thumbstick verplaats ze en druk nogmaals op de rechterknop. Dat geeft aan dat acties van de arm worden onthouden. Vervolgens kunt u op de linkerknop drukken om uit te voeren acties opgeslagen.

Verbindingsdiagram:

Machine Translated by Google



Testcode13

********** **********

****

```
#include <PS2X_lib.h>


#include <Servo.h> // voeg de servobibliotheken toe
Servo mijnservo1; // maak een servo-object om een servo te besturen
Servo mijnservo2;
Servo mijnservo3;
Servo mijnservo4;


int pos1=80, pos2=60, pos3=130, pos4=0; // definieer de variabele van 4 servohoek en wijs de
```

Machine Translated by Google




```
initiële waarde (dat is de waarde van de houdingshoek van de laars)


PS2X ps2x; // maak een PS2-controllerklasse


// op dit moment ondersteunt de bibliotheek GEEN hot-pluggable controllers, wat betekent
// je moet je Arduino altijd opnieuw opstarten nadat je de controller hebt aangesloten,
//of bel config_gamepad(pins) opnieuw nadat je de controller hebt aangesloten.
int-fout = 0;
byte trillen = 0;


int s1,s2,s3,s4;
int jiyi1[30]; // definieer vier arrays, afzonderlijk gebruikt om de hoek van 4 servo's op te slaan.
int jiyi2[30];
int jiyi3[30];
int jiyi4[30];
int i=0;
int j=0,tt=0;
ongeldige instellingen ()
{
  Serieel.begin(57600);


 // laars houding
```

Machine Translated by Google


```
    mijnservo1.schrijven(pos1);
    vertraging(1000);
    mijnservo2.schrijven(pos2);
    mijnservo3.schrijven(pos3);
    mijnservo4.schrijven(pos4);
    vertraging(1500);



  fout = ps2x.config_gamepad(13,11,10,12); // stel GamePad in (klok, commando, aandacht,
data) pinnen, controleer op fouten


  als(fout == 0){
     Serial.println("Controller gevonden, succesvol geconfigureerd");
     Serial.println ("Probeer alle knoppen uit, X zal de controller laten trillen, sneller als je harder drukt;");
    Serial.println("Als u L1 of R1 vasthoudt, worden de waarden van de analoge stick afgedrukt.");
    Serial.println ("Ga naar www.billporter.info voor updates en om bugs te melden.");
  }


    anders als(fout == 1)
     Serial.println ("Geen controller gevonden, controleer de bedrading, zie readme.txt om foutopsporing in te schakelen. bezoek
www.billporter.info voor tips voor het oplossen van problemen");
```

Machine Translated by Google


```
    anders als(fout == 2)
      Serial.println("Controller gevonden maar accepteert geen commando's. zie readme.txt om foutopsporing in te schakelen.
Bezoek www.billporter.info voor tips voor het oplossen van problemen");

     //Serial.print(ps2x.Analog(1), HEX);

  ps2x.enableRumble();                    // schakel triltrillingsmotoren in
  ps2x.inschakelenDruk();                 // schakel het lezen van de drukwaarden van de knoppen in.
}

ongeldige lus()
{
    mijnservo1.attach(A1); // stel de stuurpen van servo 1 in op A1
    mijnservo2.attach(A0); // zet de stuurpen van servo 2 op A0
    mijnservo3.attach(6); // stel de stuurpen van servo 3 in op D6
    mijnservo4.attach(9); // stel de stuurpen van servo 4 in op D9

  als(fout != 0)
    opbrengst;

    ps2x.read_gamepad(false, trillen);          // lees de controller en stel de grote motor in om te draaien
```

Machine Translated by Google




```
    snelheid 'trillen'



    als(ps2x.Button(PSB_START))                    // zal WAAR zijn zolang de knop wordt ingedrukt
          Serial.println ("Start wordt vastgehouden");
    als(ps2x.Button(PSB_SELECT))
          Serial.println ("Select wordt vastgehouden");



     als(ps2x.Button(PSB_PAD_UP)) {          // zal WAAR zijn zolang de knop wordt ingedrukt
       Serial.print ("Boven hield dit hard vast: ");
       Serieel.println(ps2x.Analog(PSAB_PAD_UP), DEC);
     }
      als(ps2x.Button(PSB_PAD_RIGHT)){
       Serial.print ("Rechts hield dit hard vast: ");
         Serieel.println(ps2x.Analog(PSAB_PAD_RIGHT), DEC);
     }
      if(ps2x.Button(PSB_PAD_LEFT)){
       Serial.print("LEFT hield dit hard vast: ");
         Serieel.println(ps2x.Analog(PSAB_PAD_LEFT), DEC);
     }
      if(ps2x.Button(PSB_PAD_DOWN)){
       Serial.print ("DOWN hield dit hard vast: ");
```

Machine Translated by Google


```
      Serieel.println(ps2x.Analog(PSAB_PAD_DOWN), DEC);
       }



       trillen = ps2x.Analog(PSAB_BLUE);                    // dit stelt de trilsnelheid van de grote motor in
gebaseerd op
                                                            //hoe hard je op de blauwe (X) knop drukt


   als (ps2x.NewButtonState())                              // zal WAAR zijn als een knop van status verandert (aan naar uit,
of uit naar aan)
   {
          als(ps2x.Button(PSB_R3))
          {
             //Serial.println ("R3 ingedrukt");
             //dossier
             s1=mijnservo1.lezen();
             vertraging(100);
             Serieel.println(s1);
             s2=mijnservo2.lezen();
             vertraging(100);
             Serieel.println(s2);
             s3=mijnservo3.lezen();
```

Machine Translated by Google




```
    vertraging(100);
    Serieel.println(s3);
    s4=mijnservo4.lezen();
    vertraging(100);
    Serieel.println(s4);

    jiyi1[i]=s1; // sla de servowaarde opeenvolgend in de array op
    jiyi2[i]=s2;
    jiyi3[i]=s3;
    jiyi4[i]=s4;
     ik++;
    j=ik;
  // vertraging(100);
  Serieel.println(i);
}

  // uitvoeren
if(ps2x.Button(PSB_L3))
{
    //Serial.println("L3 ingedrukt");
     ik=0;
    tt=1;
```

Machine Translated by Google


```
      pos1 = mijnservo1.lezen(); // noteer de hoekwaarde van 4 servohoudingen
      pos2 = mijnservo2.lezen();
      pos3 = mijnservo3.lezen();
      pos4 = mijnservo4.lezen();

    while(tt==1) // herhaal de acties
    {
      for(int k=0;k<j;k++) //for lus, om alle opgeslagen acties uit te voeren.
     {
       if(pos1<jiyi1[k]) // als de huidige hoek van servo 1 kleiner is dan de waarde die is opgeslagen in array 1.
       {
         terwijl(pos1<jiyi1[k])          //while lus, laat de servo draaien naar de positie van de opgeslagen waarde
de reeks.
         {
           mijnservo1.schrijven(pos1); //servo 1 voert de actie uit
           vertraging(5);          // vertraging 5 ms, regeling van de rotatiesnelheid van de servo.
           pos1++;
          //Serial.println(pos1);
         }
       }
       anders          // als de huidige hoek van servo 1 groter is dan de waarde die is opgeslagen in array 1.
```

Machine Translated by Google




```
        {
          terwijl(pos1>jiyi1[k])              //while lus, laat de servo draaien naar de positie van de opgeslagen waarde
de reeks.
          {
            mijnservo1.schrijven(pos1);       //servo 1 voert de actie uit
            vertraging(5);                    // vertraging 5 ms, regeling van de rotatiesnelheid van de servo.
            pos1--;
            //Serial.println(pos1);
          }

//**************************************************
// dezelfde analyse als de vorige servo
        als(pos2<jiyi2[k])
        {
          terwijl(pos2<jiyi2[k])
          {
            mijnservo2.schrijven(pos2);
            vertraging(5);
            pos2++;
            //Serial.println(pos1);
          }
        }
```

Machine Translated by Google




```
anders
{
  terwijl(pos2>jiyi2[k])
    {
      mijnservo2.schrijven(pos2);
      vertraging(5);
      pos2--;
      //Serial.println(pos1);
    }
}
//************************************************** *****
// dezelfde analyse als de vorige servo
als(pos3<jiyi3[k])
{
    terwijl(pos3<jiyi3[k])
    {
      mijnservo3.schrijven(pos3);
      vertraging(5);
      pos3++;
      //Serial.println(pos1);
    }
}
```

Machine Translated by Google




```
anders
{
  terwijl(pos3>jiyi3[k])
    {
      mijnservo3.schrijven(pos3);
      vertraging(5);
      pos3--;
      //Serial.println(pos1);
    }
}
//************************************************** *****
// dezelfde analyse als de vorige servo
als(pos4<jiyi4[k])
{
    terwijl(pos4<jiyi4[k])
    {
      mijnservo4.schrijven(pos4);
      vertraging(5);
      pos4++;
      //Serial.println(pos1);
    }
}
```

Machine Translated by Google


```
                anders

                {

                  terwijl(pos4>jiyi4[k])

                    {

                      mijnservo4.schrijven(pos4);

                      vertraging(5);

                      pos4--;

                      //Serial.println(pos1);

                    }

                }

            }

        }

        //************************************************** *******

        // verlaat de lus

          ps2x.enableRumble();                    // schakel triltrillingsmotoren in

          ps2x.inschakelenDruk();

          ps2x.read_gamepad(false, trillen);

         trillen = ps2x.Analog(PSAB_BLUE);

           als (ps2x.NewButtonState())                    //zal TRUE zijn als een knop van status verandert (op naar
uit, of uit naar aan)

           {

              als(ps2x.Button(PSB_R3))
```

Machine Translated by Google


```
            {
                tt=0;
                ik=0;
                pauze;
            }
        }
  //**************************************************** *********
  }
}

        if(ps2x.Button(PSB_L2))
          Serial.println ("L2 ingedrukt");
        if(ps2x.Button(PSB_R2))
          Serial.println ("R2 ingedrukt");
        if(ps2x.Button(PSB_GREEN))
          Serial.println ("Driehoek ingedrukt");

}

 if(ps2x.ButtonPressed(PSB_RED))                    // zal WAAR zijn als de knop ENKEL is ingedrukt
        Serial.println ("Cirkel zojuist ingedrukt");
```

Machine Translated by Google


```
    if(ps2x.ButtonReleased(PSB_PINK))                    // zal WAAR zijn als de knop NET is losgelaten
          Serial.println ("Vierkant is zojuist vrijgegeven");

    if(ps2x.NewButtonState(PSB_BLUE))                    // zal WAAR zijn als de knop ENKEL is ingedrukt OF
uitgegeven
          Serial.println ("X is zojuist gewijzigd");

      //draai
      zhuandong();
      //klauw
      zhuazi();
      //onderarm
      dat();
      //bovenarm
      xiaobi();

    if(ps2x.Button(PSB_L1) || ps2x.Button(PSB_R1)) // print stick-waarden als een van beide TRUE is
    {
          Serial.print ("Stick-waarden:");
          Serial.print(ps2x.Analog(PSS_LY), DEC); // Linker joystick, Y-as. Andere opties: LX, RY, RX
          Serieel.print(",");
```

Machine Translated by Google


```
            Serial.print(ps2x.Analog(PSS_LX), DEC);
            Serieel.print(",");
            Serial.print(ps2x.Analog(PSS_RY), DEC);
            Serieel.print(",");
            Serieel.println(ps2x.Analog(PSS_RX), DEC);
    }

  vertraging(5);
}

//********************************************** ********************
// draai
leegte zhuandong()
{
// sla rechtsaf
      if(ps2x.Analog (PSS_RX) > 200) // if duw de rechter joystick naar rechts
        {
           //Serial.println(ps2x.Analog(PSS_RX), DEC);
            pos1=pos1-1; //pos1 trekt 1 af
            mijnservo1.schrijven(pos1); // servo 1 voert de actie uit, de robotarm draait naar rechts.
           // vertraging(5);
            if(pos1<1) // beperk de juiste draaihoek
```

Machine Translated by Google


```
            {
                pos1=1;
            }
          }
           // sla linksaf
          if(ps2x.Analoog (PSS_RX) < 50)              // als duw de rechter joystick naar links
          {
             //Serial.println(ps2x.Analog(PSS_RX), DEC);
             pos1=pos1+1; // pos1 plus 1
             mijnservo1.schrijven(pos1);          // de robotarm draait naar links
            // vertraging(5);
             als(pos1>180)             // beperk de linker draaihoek
             {
                pos1=180;
             }
          }
    }
    //********************************************** **********************
    // de bovenarm
  leegte xiaobi()
  {
     // bovenarm voorzijde
```

Machine Translated by Google




```
        if(ps2x.Analog(PSS_RY)<50) // if duw de rechter joystick omhoog
        {
            pos2=pos2-1;
            mijnservo2.schrijven(pos2);        // de bovenarm gaat omhoog
            vertraging(5);
            als(pos2<0)          // beperk de hefhoek
            {
                pos2=0;
            }
        }
        // bovenarm naar achteren
        if(ps2x.Analog(PSS_RY)>200)                // druk de rechterjoystick naar beneden
        {
            pos2=pos2+1;
            mijnservo2.schrijven(pos2); // de robotarm gaat naar beneden
            vertraging(5);
            if(pos2>180) // beperk de neerwaartse hoek
            {
                pos2=180;
            }
        }
}
```

Machine Translated by Google


```
//************************************************** ***************

leegte zhuazi()

{

// sluit de klauw

    if(ps2x.Analog(PSS_LX)>220) // if duw de linker joystick naar rechts

    {

        pos4=pos4-1;

        Serieel.println(pos4);

        mijnservo4.schrijven(pos4); // servo 4 voert de actie uit en de klauw wordt geleidelijk gesloten

        vertraging(5);

        if(pos4<0) // als de waarde van pos4 wordt afgetrokken tot 37, is de klauw in 37 graden die we hebben getest

gesloten.)

        {

            pos4=0;

        }

    }

    // open de klauw

    if(ps2x.Analog(PSS_LX)<10) // if duw de linker joystick naar links

    {

        pos4=pos4+8;

        Serieel.println(pos4);

        mijnservo4.schrijven(pos4); // servo 4 voert de actie uit en de klauw wordt geleidelijk geopend
```

Machine Translated by Google




```
            vertraging(5);
            als(pos4>108)        // beperk de maximale geopende hoek
            {
                pos4=108;
            }
        }
}
//************************************************** *********
leegte dabi()
{
   // onderarm voorzijde
    if(ps2x.Analog(PSS_LY)>200) // if duw de linker joystick omhoog
    {
        pos3=pos3+1;
        mijnservo3.schrijven(pos3); // de onderarm strekt zich uit
        vertraging(5);
        if(pos3>180) // beperk de uitgerekte hoek
        {
            pos3=180;
        }
     }
    als(ps2x.Analoog(PSS_LY)<10)        // als duw de linker joystick naar beneden
```

Machine Translated by Google


```
        {
            pos3=pos3-1;
            mijnservo3.schrijven(pos3); // de onderarm trekt zich terug
            vertraging(5);
            als(positie3<35)         // beperk de ingetrokken hoek
            {
                pos3=35;
            }
        }
    }
```

********** **********

****

Testresultaat

Bedien de PS2-joypad en druk op de rechter thumbstick om de houding van de robotarm te onthouden, en druk vervolgens op de linker thumbstick om de opgeslagen houdingen uit te voeren. Houd de rechter thumbstick ingedrukt om de lus van houdingen te verlaten.

Testresultaat:

Stapel het schild op V4.0 en upload de code. Als de PS2 Joypad is ingeschakeld en aangesloten, kunt u de PS2 Joypad gebruiken om de robotarm te besturen en verschillende houdingen te onthouden, in een lus.

6. **Middelen**

Machine Translated by Google

Testcode, bibliotheken en gedetailleerde projecten:

https://fs.keyestudio.com/KS0198

KEYESTUDIO WIKI: http://wiki.keyestudio.com/ ARDUINO-software:

https://www.arduino.cc/en/Main/OldSoftwareReleases#1.5.x

Montagevideolink: http://www.keyestudio.com/wp/ks0198/