

Robotica club Werenfridus



Handleiding MER-rupsvoertuig-ar

Ton Schuckman
Amanuensis/instructeur natuurkunde



Inhoudsopgave

De rechthoekige grondplaat.....	03
Zaagsleuven voor de servo.....	03
Plaatsing middelste wielletjes.....	03
Montage van de motortjes.....	04
Rupsband met voorste wielletje.....	04
Het solderen van de draden op de motoren.....	04
Montage van de zijpanelen.....	04
Het plaatsen van de microcontroller.....	05
De batterijhouder.....	05
Het printplaatje.....	05
Werking van de schakeling.....	06
De diodes en condensatoren.....	06
Draden op de microcontroller.....	06
Het elektrisch schema.....	07
Polariteit.....	08
Batterijen en opslag.....	08
Assemblage van de printplaat.....	08
Tabel 1 afkortingen van de insteek printplaat.....	09
Tabel 2 aansluitingen op de insteek printplaat.....	09
Plaatsing van de servo.....	10
De ultrasoon sensor.....	10
Aansluiting ultrasoon sensor.....	10
Installeren van het Arduino programma.....	11
Software rondje acht.....	12
Software rechtuit, bij obstakel linksaf.....	13
Software met ultrasoon sensor en servo.....	14
Tabel 3 materiaallijst.....	16

Handleiding

De bouw is gesplitst in drie delen:

- **Het mechanisch gedeelte**
- **De elektronische schakeling met aansluitingen**
- **Het programmeren van de microcontroller**

Het mechanisch gedeelte

De rechthoekige grondplaat

Gebruik bij het boren altijd een veiligheidsbril voor mogelijk rondvliegende metaalsplinters en bind lange haren achter je hoofd in een knot om te voorkomen dat ze door de ronddraaiende boorkop worden meegetrokken.

De zwarte pijl (zie figuur 1) geeft de boven en voorzijde aan. Het ronde gat met een diameter van 15 mm is de rechterzijde. Plaats de zijkanten zodanig en in het midden gecentreerd dat de motoren aan de achterzijde komen. Draai het geheel nu om en teken de vier plaatsen af waar de zijkanten aan de grondplaat vastgezet moeten worden. Boor nu deze gaatjes met een boortje van 3,5 mm. De scherpe splinters kun je wegdraaien met een grote 10 mm boor. Monteer de zijkanten nog niet, dit komt later.

Zaagsleuven voor de servo

We maken nu al met de ijzerzaag twee sleufjes aan de voorkant voor als je later de servo gaat monteren. Als de robot klaar is dan is dit niet meer mogelijk. Zaag hiervoor 30 mm vanaf de linker voorkant gemeten een sleufje van 15 mm diep. Doe hetzelfde 40 mm vanaf de rechter voorkant.

Plaatsing middelste wielletjes

Steek hiervoor een lange bout door het wielletje en draai daarna een moertje erop tegen het wielletje aan. Als je het moertje nu een halve slag losdraait kan het wielletje vrij draaien en zet je het met een ander moertje vast op de middelste plek van de zijplaat. Draai de moertjes met twee steeksleuteltjes tegen elkaar in vast.

Montage van de motortjes

Draai eerst met de hand drie korte boutjes hand vast en zet ze daarna pas vast met een sleuteltje. Zo voorkom je dat ze er scheef ingedraaid worden en de schroefdraad beschadigd. Soms moet het aandrijfwieltje nog vastgezet worden op het aluminium hulpstukje. Controleer hier eerst met een boutje of de schroefdraad goed is. Voordat je het aandrijfwieltje op het asje schuift controleer je ook hier eerst of het boutje er goed ingedraaid kan worden. In sommige gevallen is de schroefdraad niet helemaal doorlopend en moet dat alsnog gebeuren met een m3 tapeindje. Het boutje komt aan de vlakke kant van het motorasje.

Rupsband met voorste wiel

Het kan voorkomen dat de rupsband nog op maat gemaakt moet worden. De rupsband leg je om het nog niet gemonteerde voorste wiel en daarna om de andere wieltjes. Zet daarna het voorste wiel vast zoals je met het middelste wiel gedaan hebt.

Het solderen van de draden op de motoren

Adem de soldeerdamp niet in, deze bestaat uit zware metalen. Let op dat je niet met de soldeerbout tegen de plastic kap van het motortje aankomt. Algemene werkwijze: voor de + aansluitingen gebruiken we rode draden en voor de - kant zwarte draden. Knip eerst de draden op lengte (20 cm) vanaf de motor tot de insteekprintplaat en soldeer op de linker motor een rode draad op het + oogje en zwart op het – oogje. Op de rechter motor is het andersom, rood op het – oogje en zwart op het + oogje want deze motor zit gespiegeld ten opzichte van de linker motor. Voor wie nog nooit gesoldeerd heeft: Hou de punt van de warme soldeerbout twee volle seconden tegen het oogje waar het draadje al doorheen zit, nu het soldeert in ertegenaan houden en je ziet dat het soldeert in volledig vervloeit met het draadje en oogje. Bout nu weghalen en even wachten tot het afgekoeld is. Ter controle voel je even of het draadje stevig vast zit. Aan het andere eind van de draden komen header pinnen.

Montage van de zijkanten

Na het solderen ga je de twee complete zijkanten monteren onder de grondplaat. Plaats ze zo langs de langste zijden van de grondplaat zodat de motoren aan de achterzijde komen. Gebruik hiervoor de vier 75 mm lange tapeinden, deze zijn aan één kant afgezaagd. Zorg ervoor dat die kant aan de onderkant komt en dat ze alle vier even hoog boven de grondplaat uitsteken. Hierdoor wordt de elektronica beschermd en kun je de robot later verder uitbreiden of voorzien van zonnepanelen.

De elektronische schakeling

Het plaatsen van de microcontroller

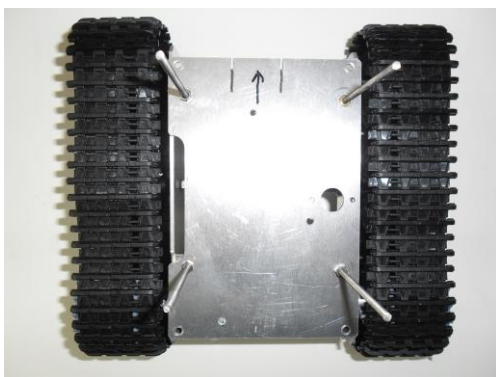
Indien nodig met een isolatiestuk aan de onderzijde om kortsluiting via de aluminium grondplaat te voorkomen. Boor gaatjes door het isolatiestuk op de plaatsen van de gaatjes in de Arduino. Plaats hem zodanig dat de aansluitingen naar achter wijzen. Steek nu bij de gaatjes van pin A5, het drukknopje en bij de voeding plug touwtjes door de grondplaat en microcontroller en knoop ze vast.

De batterijhouder

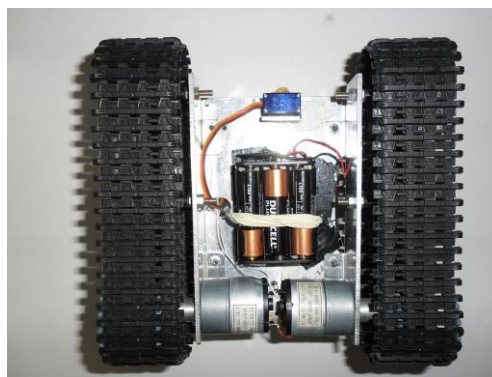
Knip een grote paperclip open en maak daarvan twee s-haakjes waar een elastiekje aan vast komt zodat de batterijhouder aan de onderzijde in de twee gaatjes op de grondplaat gehaakt kan worden. Plak op deze plek twee lagen duc-tape om kortsluiting van de batterijhouder te voorkomen. De uiteinden van de rode en zwarte draden van de batterijhouder stekker soldeer je op aparte dubbele male headers.

Het printplaatje

Het insteek printplaatje plaats je boven op de grondplaat met de langste zijde tegen de microcontroller. Het heeft een zelfklevende laag aan de onderzijde. Snij met een stanleymes een strookje van 5 mm in en verwijder dit strookje plastic zodat de insteekprintplaat maar op 5 mm zelfklevend materiaal vast komt te zitten. Je kunt hem anders nooit meer verwijderen of verplaatsen zonder hem stuk te trekken.



Figuur 1 bovenzijde MER



Figuur 2 onderzijde MER

Werking van de schakeling - transistoren

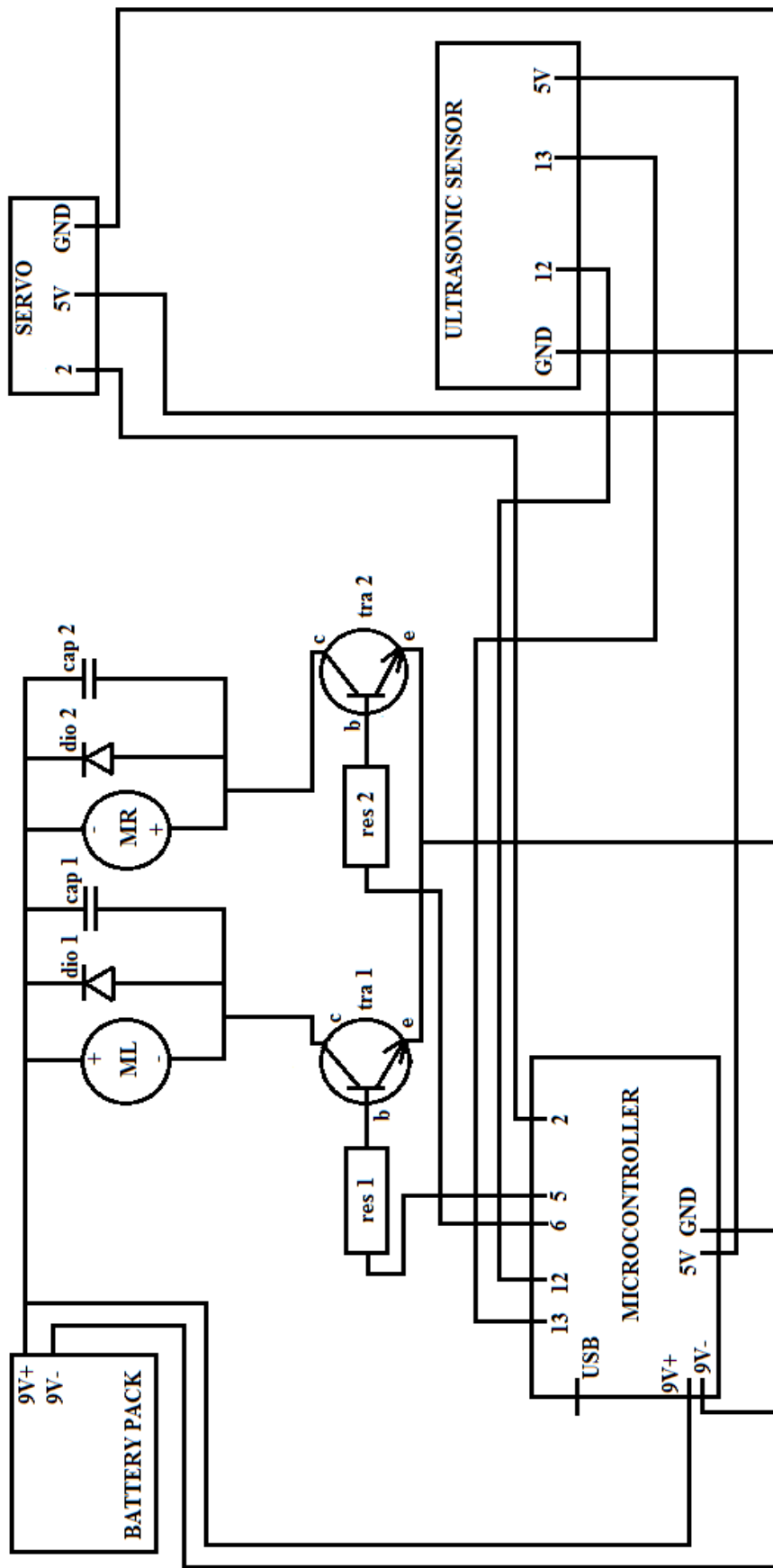
De microcontroller werkt zoals de meeste elektronica op een spanning van 5 volt die wordt geleverd door de zes oplaadbare AA batterijen. Gebruik alleen oplaadbare batterijen om een te hoge spanning te voorkomen. Deze geven weliswaar samen maximaal 9 volt maar de microcontroller heeft zelf een spanningsstabilisator aan boord die daar 5 volt van maakt. De motoren doen het hier prima op, die kunnen maximaal 12 volt aan. Als nu via het programma dat je gaat schrijven een spanning van 5 volt op pin 5 gezet wordt, gaat deze spanning, verlaagd door de weerstand naar de basis B van de transistor. Een transistor heeft een collector, emitter en een basis. Normaal als de basis geen spanning krijgt blijft de transistor 'gesloten'. Dit kan omdat er in de transistor een halfgeleider materiaal zit dat pas geleidend wordt als er een klein stroompje via de basis naar de emitter loopt. Pas dan wordt de transistor geleidend en kan de hoofdstroom via de collector C naar de emitter E stromen. De motor gaat dan draaien want er is een stroomkring tot stand gebracht. Volg het elektrisch schema (figuur 3) van + naar - maar eens. Kruisende lijnen zijn niet met elkaar verbonden.

De diodes en condensatoren

De diode is een onderdeel dat maar in één richting stroom doorlaat en is er als beveiliging. Als je namelijk met de hand een motor laat draaien verandert de motor in een dynamo en zou de microcontroller kunnen beschadigen. De diode zorgt er dan voor dat deze inductie stroom alleen naar de batterij terug kan stromen. De condensatoren dienen ter ontstoring van de motor.

Draden op de microcontroller

Let goed op dat je de draadjes (plus en min) van de voedingsplug op de microcontroller niet verwisseld op het insteek printplaatje. Hierdoor kan de microcontroller beschadigen. Dit geldt ook voor de pinnen 5V en GND.



Ton Schuckman

MEER electrical diagram

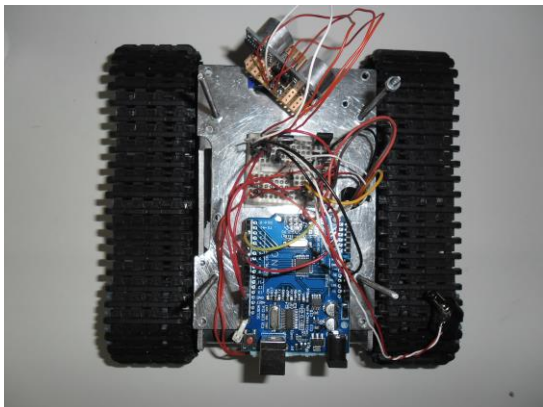
Figuur 3 het elektrisch schema

Polariteit

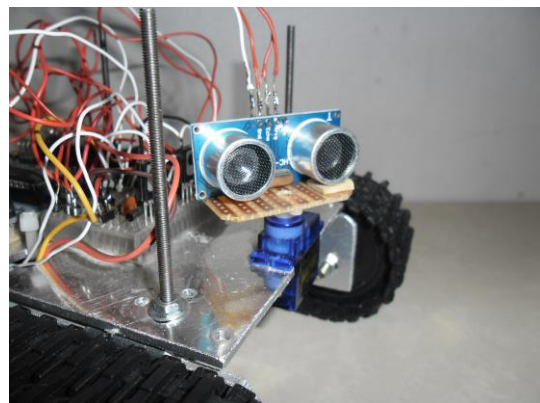
Voor de weerstand en condensator maakt het niet uit hoe om ze gemonteerd worden, voor de diode echter moet het witte randje (de sperzijde) op de + markering komen. De transistoren worden met de metalen zijde naar voren gemonteerd zodat e/emitter, c/collector en b/basis, goed uitkomen. (de aansluiting van de servo wordt in tabel 2 aangegeven, de gele aanstuurdraad komt in pin 2 van de microcontroller). De insteekpinnen van de microcontroller en de insteekprintplaat hebben een gestandaardiseerde maat.

Batterijen en opslag

Als je ze nog niet hebt dan moet je nu gaan zorgen voor de batterijen: 6 maal 1,5 volt AA oplaadbare batterijen. Als je het apparaat opbergt leg dan de plug die je uit de microcontroller trekt op een rupsband, hier blijft immers spanning op staan. Langdurige opslag: verwijder de batterijen om lekkage en kortsluiting te voorkomen.



Figuur 4 componenten configuratie



Figuur 5 sensor en servo configuratie

Assemblage van de printplaat

Maak de draden niet te lang voor een goed overzicht maar je kunt ook kant en klare jumper-wires gebruiken. Kort de uiteinden van de weerstanden, diodes en condensators in tot een lengte van 12 mm. Nu kun je het insteekprintplaat schema gebruiken (tabel 2) om eerst de elektronische componenten en daarna de draden van boven naar beneden op de goede manier aan te sluiten. Ieder rechthoekje van het schema komt overeen met één insteekgaatje op de insteekprintplaat. Als je goed kijkt dan zie je dat de insteekgaatjes aan de onderzijde per vijftal met elkaar verbonden zijn. Zo worden de componenten met elkaar verbonden.

bas	basis (transistor)	m.l.	motor links
bat	batterijhouder	m.r	motor rechts
bru	bruine draad (servo)	ora	oranje draad (servo)
cap	capacitor/condensator	pin	aansluiting op de mic
col	collector (transistor)	res	resistor/weerstand
dio	diode	ser	servo
ech	echo (u.s sensor)	tra	transistor
emi	emitter (transistor)	tri	trigger (u.s sensor)
gnd	ground/massa (-)	u.s.	ultrasoon sensor
mic	microcontroller	wir	wire/draad

Tabel 1 afkortingen van de insteekprintplaat

					tra emi	tra col	tra bas					tra emi	tra col	tra bas		
u.s. 5V	u.s. tri	u.s. ech	u.s. gnd			dio 1 -			dio 1 +				dio 2 -			dio 2 +
					wir e -	cap 1			cap 1				cap 2			cap 2
wir f +	pin 13	pin 12	wir e -									bat -				
					wir c -	m.l. -	res 10 ³		wir a +			wir d -	m.r .	res 10 ³		wir b +
## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##	## ##
bat 9V	bat 9V				wir c -						wir f +	wir d -				
wir a +	wir b +				wir g -							wir g -				
	mic +						res 10 ³			ser ora	ser 5V	ser bru		res 10 ³		
					mic -											
m.l. +	m.r .				pin gnd		pin 5			pin 2	pin 5V			pin 6		

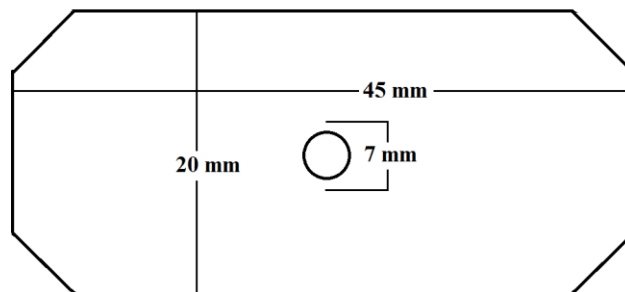
Tabel 2 aansluitingen op de insteek printplaat

Plaatsing van de servo

Hier volgt de omschrijving om de ultrasoon sensor op een servo te monteren aan de voorzijde zodat de robot om zich heen kan kijken (eigenlijk luisteren) en er op obstakels gereageerd kan worden. Hiervoor had je in het begin al de twee sleufjes uitgezaagd op de grondplaat. Buig het stukje aluminium tussen de sleufjes nu 90 graden naar beneden zodat de servo met een rijgstrip vastgezet kan worden. De servo moet immers ook nog losgemaakt kunnen worden als hij stuk is. Zie verder het schema van de insteekprintplaat.

De ultrasoon sensor

Maak uit een oud stukje printplaat of iets anders het voetstukje voor de ultrasoon sensor. Zaag een stuk van 45 mm bij 20 mm uit en boor dan in het midden een gat van 7 mm. De scherpe hoekjes knip je af met een kniptang. Nu lijm je wederom met twee minuscule drupjes twee seconden lijm het voetje van de servo met de platte kant aan de onderkant van het stukje printplaat. Lijm nu de ultrasoon sensor met de pinnen naar boven gericht op de bovenkant van het grondplaatje zo horizontaal mogelijk met een stukje hout van 2 mm dik ter ondersteuning onder één van de twee aluminium buisjes. Je kunt hiervoor het uiteinde van een wasknijper gebruiken.



Figuur 6 grondplaatje ultrasoon sensor

Aansluiting ultrasoon sensor

De vier aansluit draden (20 cm lang vanwege het ronddraaien) soldeer je op de pennen van de sensor. Het aansluit schema van de draden is al vermeld in tabel 2. Soldeer de draden voor GND (zwart), echo, trig en vcc (rood) op een viervoudige male header. Maar je kunt ook kant en klare male/female draden gebruiken. Het geheel van printplaatje met ultrasoon sensor schroef je nu voorzichtig vast op de servo. Hieronder de benodigde software, te beginnen met het rondje acht.

Het programmeren van de microcontroller

Installeren van het Arduino programma

Je kunt de Arduino IDE (integrated development environment) programma gratis downloaden van de Arduino website. Met onderstaand programma rijdt de robot een rondje acht. Achter twee // kun je uitleg plaatsen die niet in het programma wordt meegenomen. Als je nu het programma overtypt dan heb je je eerste programma geschreven. Als de microcontroller is aangesloten op de pc en geïnstalleerd, dan kun je dit programma uploaden. Mogelijk moet er nog wel een poort op de pc worden toegewezen. Hiervoor klik je bovenin het Arduino programma op → hulpmiddelen → poorten en kijk welke poort er gebruikt wordt. Nu klik je op de pc op → starten → rechtermuisknop computer → beheren → apparaat beheer en controleer of hier de juiste poort gebruikt wordt. Pas dit eventueel aan. De onderstaande software is met opzet in het Engels omdat dit gebruikelijk is.

Software rondje acht

```
const byte motorLeft = 5; const byte motorRight = 6; //motors are assigned to pin 5 and 6.
```

```
void setup() //put your setup code here, to run once:
```

```
{ pinMode(motorLeft, OUTPUT); pinMode(motorRight, OUTPUT); } //pins set as output.
```

```
void loop() //put your main code here, to run repeatedly:
```

```
{ delay(5000); //time to put the MER on the ground. (5000) = 5 seconden.  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); delay(1000);  
digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); delay(2000);  
digitalWrite(motorRight, HIGH); delay(1000);
```

```
digitalWrite(motorRight, LOW); delay(2000); }
```

Software rechtuit, bij obstakel linksaf

```
const byte echoPin = 12; const byte trigPin = 13;
const byte motorLeft = 5; const byte motorRight = 6;
int duration; int cm;
```

```
void setup()
```

```
{ Serial.begin(9600); //initialize serial communication.
pinMode(trigPin, OUTPUT); pinMode(echoPin, INPUT);
pinMode(motorLeft, OUTPUT); pinMode(motorRight, OUTPUT); forward(); }
```

```
int microsecondsToCentimeters(int microseconds)
```

```
//speed of sound = 340 m/s or 29 microseconds per cm; the sound wave travels out and
//back, so the distance of the object is half of the distance travelled.
{ return microseconds / 29 / 2; }
```

```
void forward()
```

```
{ digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, HIGH); }
```

```
void turn()
```

```
{ digitalWrite(motorLeft, LOW); digitalWrite(motorRight, HIGH); delay (300); }
```

```
void loop()
```

```
{ digitalWrite(trigPin, LOW); delayMicroseconds(70); digitalWrite(trigPin, HIGH);
delayMicroseconds(10); digitalWrite(trigPin, LOW); duration = pulseIn(echoPin, HIGH);
cm = microsecondsToCentimeters(duration);
```

```
//this line prints the value of the ultrasonic sensor on the serial monitor.
```

```
Serial.print(cm); Serial.print("cm"); Serial.println();
```

```
if (cm > 50) { forward(); } else { turn(); } }
```

Als de robot goed functioneert met de ultrasoon sensor dan kan de laatste stap gezet worden, het toepassen van het laatste programma dat ook de servo implementeert.

Software met ultrasoon sensor en servo

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

//pin configuration.
const byte servoPin = 2; const byte motorLeft = 5; const byte motorRight = 6;
const byte echoPin = 12; const byte trigPin = 13; //send = trigger pin, receive = echo pin.

//delay determines how long the MER turns, increase to make the turning angle larger.
const int turnDelay = 700;
//standard angle for looking straight ahead, compensate for construction errors.
const int standardAngle = 90;
//angle in degrees that controls the direction in which the sensor looks.
const int viewAngle = 45;
//delay to let servo turn, not optimized; for smaller angles smaller delay could be used.
const int servoDelay = 250;
//object distance in cm at which the MER decides to look around and turn another way.
const int lineOfSight = 40;
//pwm duty cycle of the motor, use a value from 0 (no movement) to 255 (full speed).
const int motorSpeedLeft = 100;
const int motorSpeedRight = 100;

void setup() //setup pin configuration and initialize the sensor position.
{ delay(3000); //delay to give us some time to place the MER on the ground.
myservo.attach(servoPin); //initialize the servo and look straight ahead.
turn_servo(standardAngle);
pinMode(motorLeft, OUTPUT); pinMode(motorRight, OUTPUT);
pinMode(trigPin, OUTPUT); pinMode(echoPin, INPUT); drive_mer(); } //start the mer.

void drive_mer() //this function lets the MER drive forward.
{ analogWrite(motorLeft, motorSpeedLeft); analogWrite(motorRight, motorSpeedRight); }

void stop_mer() //this function stops the mer.
{ digitalWrite(motorLeft, LOW); digitalWrite(motorRight, LOW); }

void turn_servo(int angle)
//turn servo to given angle, do not return until angle is reached.
//if the servo is already at the correct angle, we don't have to do anything.
```

```

{ if (angle == myservo.read()) { return; }
else { myservo.write(angle); delay(servoDelay); } }

```

```

int look() //looking function, returns distance in the observed direction as an integer.
{ int duration, distance;
digitalWrite(trigPin, LOW); delayMicroseconds(20); digitalWrite(trigPin, HIGH);
delayMicroseconds(10); digitalWrite(trigPin, LOW); duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1; return distance; }

```

```

//turning function: makes MER turn in preferred direction. 0 for left and 1 for right.
void turn_mer(int direction)
{ if (direction == 0) { digitalWrite(motorLeft, LOW); digitalWrite(motorRight, HIGH); }
else { digitalWrite(motorLeft, HIGH); digitalWrite(motorRight, LOW); }
delay(turnDelay); stop_mer(); } //stop the engines when the MER has turned far enough.

```

```

//this function turns the MER depending on the observed distances.
void handle_measurements(int distanceLeft, int distanceStraight, int distanceRight)

```

```

//if objects in each direction, stop the mer.
{ if (distanceLeft < lineOfSight and distanceStraight < lineOfSight and
distanceRight < lineOfSight) { stop_mer(); }

```

```

//if we see something on the left side, turn right.
else if (distanceLeft < lineOfSight) { turn_mer(1); drive_mer(); }

```

```

//if something straight ahead go to best direction.
else if (distanceStraight < lineOfSight) { if (distanceRight > distanceLeft)
{ turn_mer(1); } else { turn_mer(0); } drive_mer(); }

```

```

//if we see something on the right side, turn left.
else if (distanceRight < lineOfSight) { turn_mer(0); drive_mer(); } }

```

```

void loop()
{ int distanceLeft, distanceStraight, distanceRight;
//look each direction starting left.
turn_servo(standardAngle + viewAngle); distanceLeft = look();
turn_servo(standardAngle); distanceStraight = look();
turn_servo(standardAngle - viewAngle); distanceRight = look();
handle_measurements(distanceLeft, distanceStraight, distanceRight); }

```

Onderdeel	Type	Aantal
Tank chassis	HE0102-460	1
Microcontroller	Arduino uno	1
Insteek printplaat	HE0109-002	1
Batterijhouder	615587-89	1
Boutjes en moertjes	M3	< >
Diodes	SB160	2
Transistors	NPN BD139	2
Weerstand	1000 ohm	2
Condensatoren	1 microfarad	2
Draden male/male	HE0108-002	< >
Male headers	HE0236-600	1
Ultrasoon sensor	HC-SR04	1
Servo	SG90	1
Voeding plug	393468-89	1
Batterij stekker	624691-89	1
Draad bundel	HE0167-001	1
Batterij 1,5 volt AA	2100 mAh	6
USB kabel	Type A naar B	1

Tabel 3 materiaallijst

Ton Schuckman
Amanuensis/instructeur natuurkunde
Tabor College Werenfridus

