

# Workshop

## WLAN-Controlled Space Mining Rover

(controlled by your mobile phone)



CREATIVE-LAB.LU



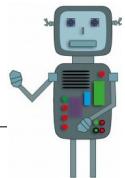
Lycée des  
Arts et Métiers





## Inhaltsverzeichnis / Table of contents

1. Montageanleitung / Assembly instructions.....	3
1.1. Materialliste / Bill of materials.....	3
1.2. Mechanische Teile montieren.....	3
1.3. Platine bestücken und löten PCB assembling and soldering.....	6
1.4. Motoren anschließen / Connecting the motors.....	7
1.5. Batterie und Schalter anschließen / Batteries and switch.....	8
1.6. Fertigstellung / Finish.....	8
2. Erster Test / First Test.....	8
3. Software.....	10



# 1. Montageanleitung / Assembly instructions

## 1.1. Materialliste / Bill of materials

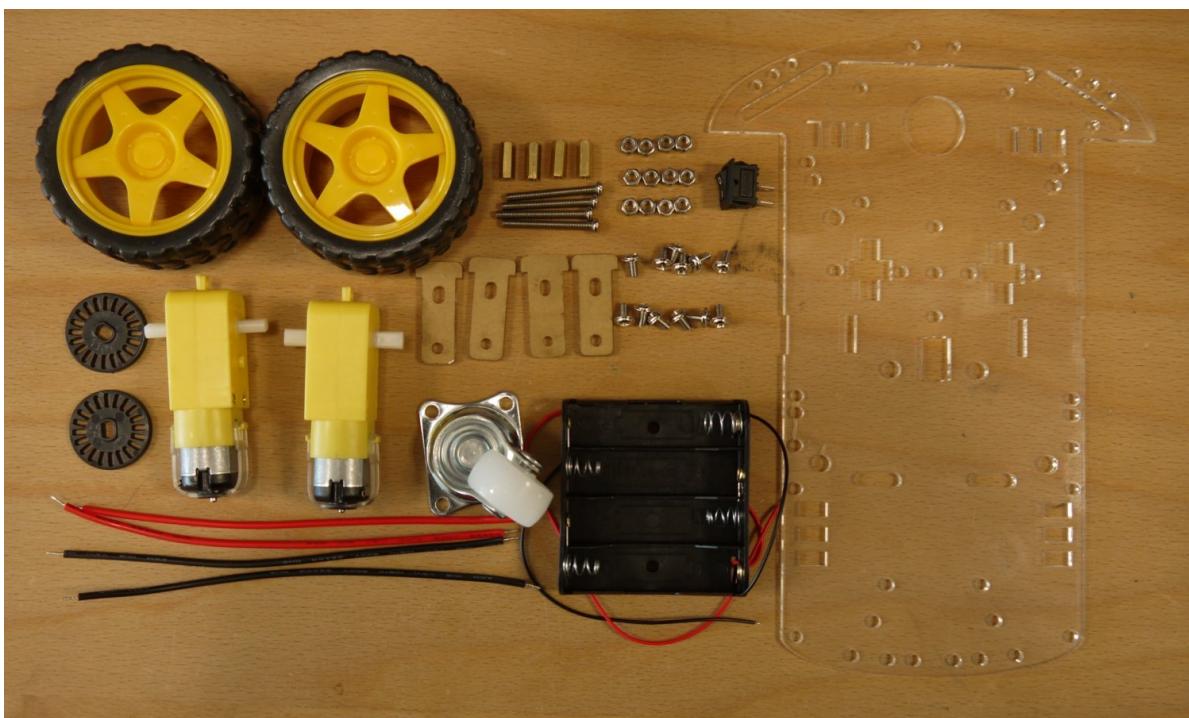
- Roboterchassis (Bausatz) / robot car chassis (kit)
- Mikrocontrollerplatine: Wemos D1 mini pro / microcontroller-board: Wemos D1 mini pro
- Creative-lab Platine / PCB from creative-lab
- IC L293D Motortreiber / motor-driver IC L293D
- IC Fassung 16 polig / IC socket 16-pins
- Diode 1N4004 oder 1N4007 / diode 1N4004 or 1N4007
- Roter Plexiglasstreifen 60x10 / red bar 60x10 with 2 holes (washer for the battery holder)
- 2 Schrauben M2\*10 mit Muttern / 2 screws M2x10 + nuts
- 2 Schrauben M3\*10 mit Abstandshaltern und Muttern / 2 screws M3x10 + nuts + 5mm spacer (to fix the PCB)
- 4 Batterien AA 1,5V / 4 batteries AA alkaline 1,5V

**Achtung:** Bei den meisten Teilen ist es wichtig, in welche Richtung man sie dreht! Um frustrierendes Wieder-Auseinandernehmen zu vermeiden, sollte man jeden Schritt genau überlegen.

**Most parts have to be mounted in the right direction.** It is frustrating to unmount parts that are not mounted the right way, so stay concentrated and look twice at the pictures.

## 1.2. Mechanische Teile montieren

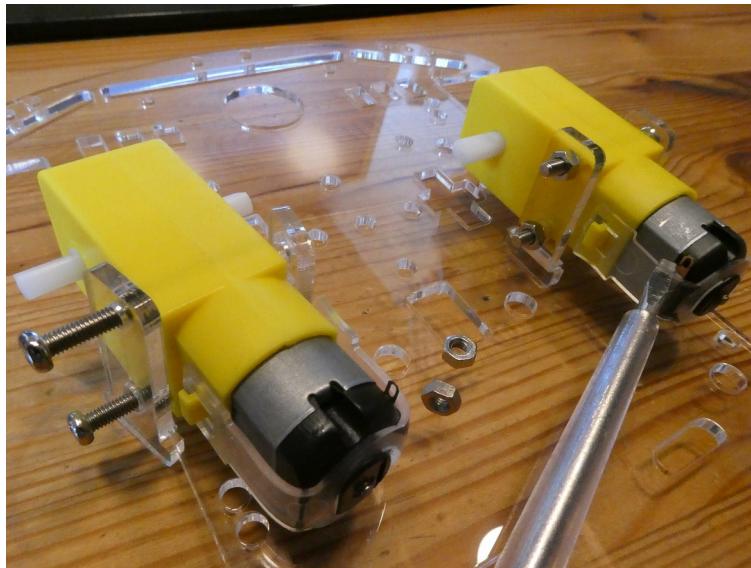
Dein Bausatz sollte diese Teile enthalten (parts from your kit):





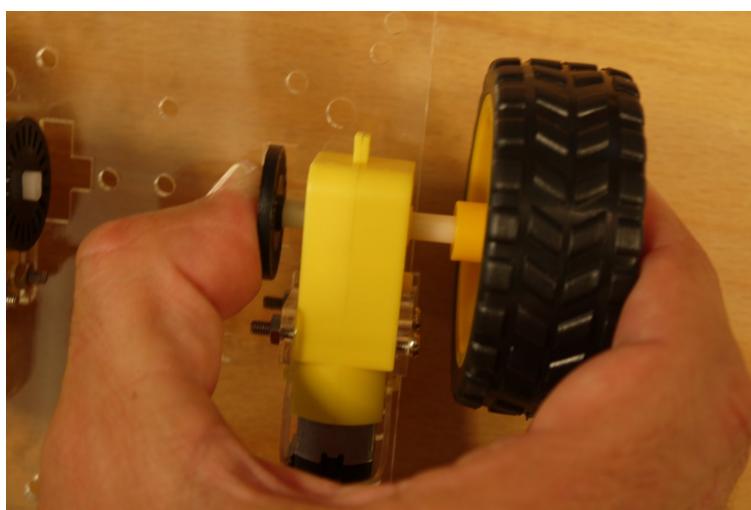
- Entferne die Schutzfolie bzw. Papier von den Plexiglasteilen.  
*Remove the protective film or paper from the acrylic glass plates.*
- Montiere die Motoren. Du benötigst 4 M3x30 Schrauben und 4 M3 Muttern.  
**Achte darauf, dass die Kontakte beider Motoren (siehe Schraubendreher-Spitze) nach innen zeigen!**

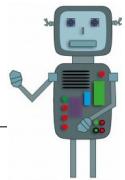
*Mount the 2 motors. You need 4 M3x30 screws an 4 M3 nuts.  
Pay attention to turn the motor contacts to the insight (see screwdriver tip).*



- Stecke die Encoderscheiben (schwarze Schlitzscheiben) sowie die Räder auf den Motor.  
**Vorsicht! Die Plexiglashalter brechen leicht ab! Am besten den Motor gut dabei festhalten, so dass keine Kraft auf die Halterung wirkt.**

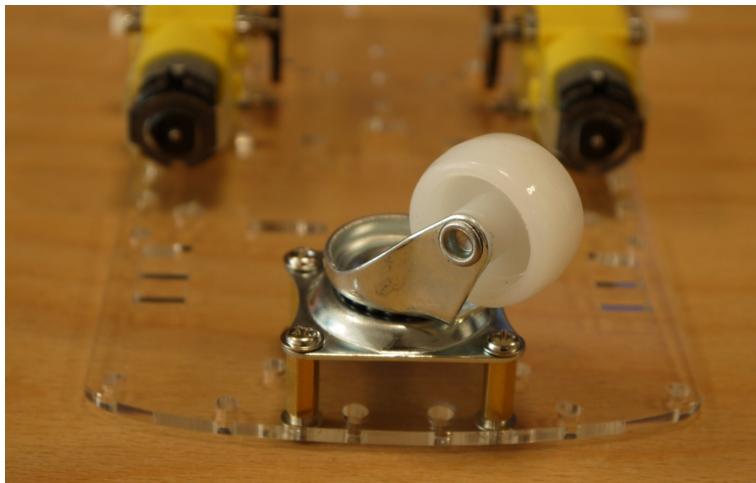
*Fix the wheels and encoder disks.  
The acrylic holders can break very easily so be gentle :).*





- Montiere das Vorderrad mit 4 Abstandshaltern M3x12 und 8 Schrauben M3x6.

*Mount the front wheel (4 spacer M3x12, 8 screws M3x6).*

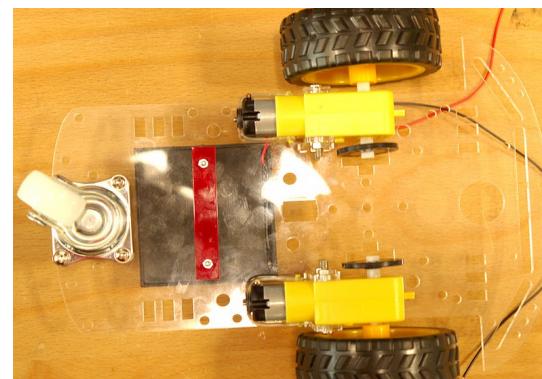
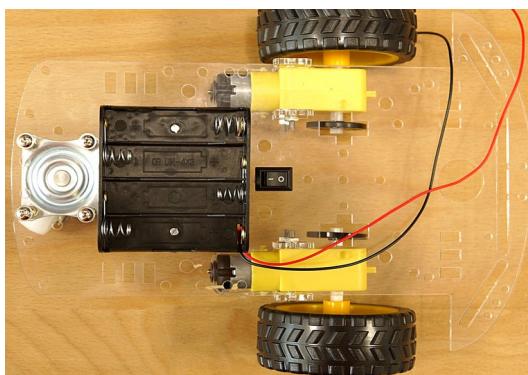


- Drehe die Platte um und montiere den Batteriehalter. Dieser wird mit 2 M2-Schrauben befestigt. Der kleine rechteckige Streifen 60x10 mit 2 Bohrungen dient dabei als Unterlegscheibe für die Muttern (unten).

*Turn the chassis to mount the battery holder. It is fixed with 2 M2 screws. The red bar (60x10) with 2 holes is used as washer for the 2 nuts (mounted under the chassis).*

- Drücke den Schalter in die rechteckige Öffnung neben dem Batteriehalter.

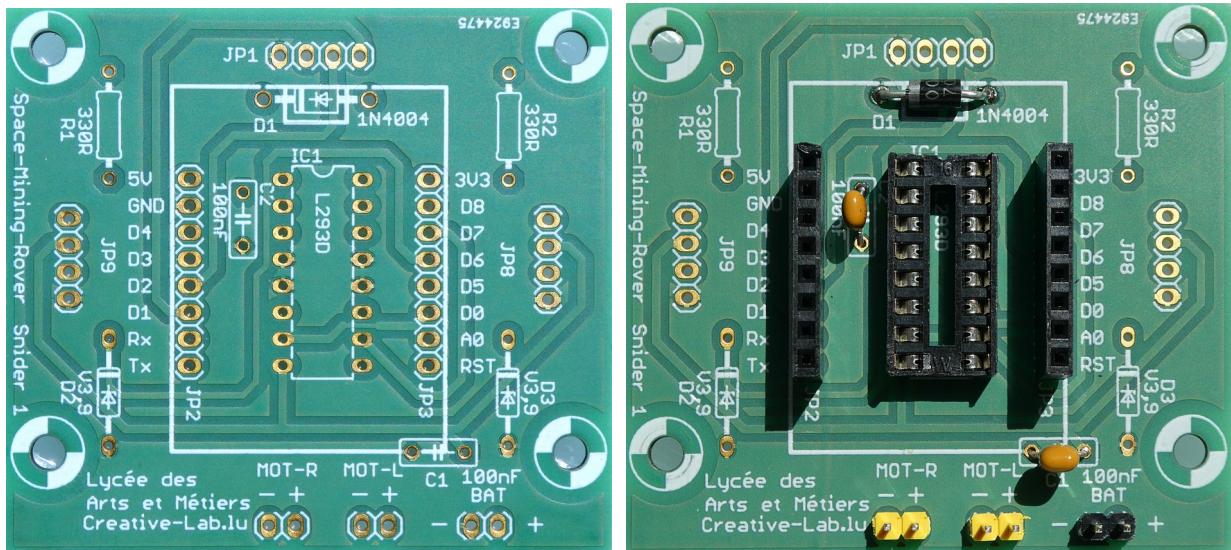
*Push the switch in the square hole (click) near the battery holder.*





## 1.3. Platine bestücken und löten

### PCB assembling and soldering



- Löte die Diode 1N4004 (oder ähnlich) ein. **Achte auf die Polung, der silberne Ring ist auf der Platine markiert.**

*Solder the diode 1N4004 (or 1N4007). Pay attention, the diode has a polarity. The cathode (silver ring) is on the left (white ring on the PCB).*

- Löte den IC-Sockel ein. **Achtung: die Kerbe muß zur Diode hin zeigen.** Am besten fängt man mit 2 Pins an, die einander diagonal gegenüber liegen. Dann kontrolliert man ob der Sockel flach aufliegt. Wenn nicht, kann man einen Pin nochmal erhitzen und die Position korrigieren. Dann verlötet man die restlichen Pins.

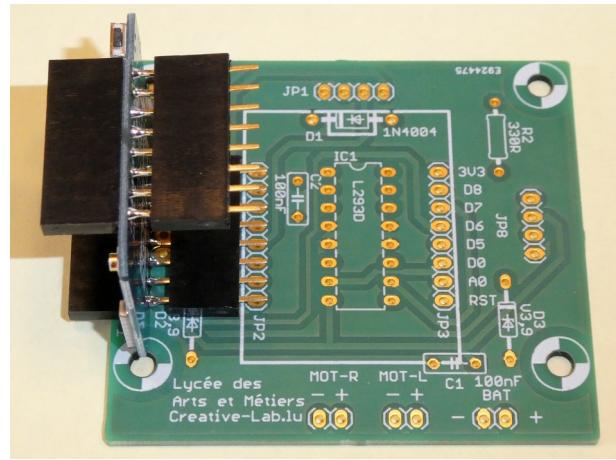
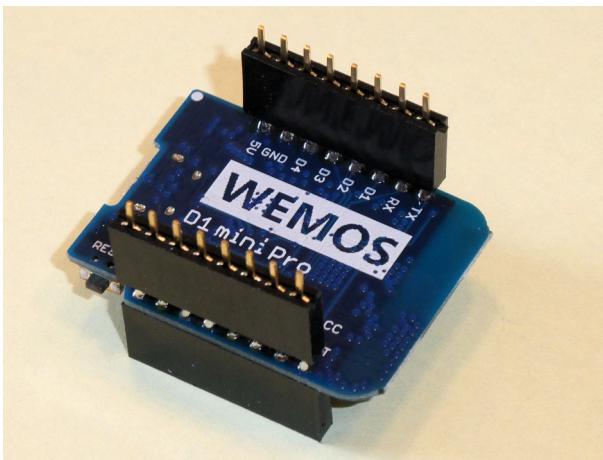
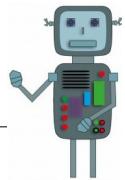
*Next is the IC socket. It has to be turned the right way. It has a little notch on one side. The notch can also be seen on the PCB (top under the diode). First solder 2 pins at the edges. We control that the socket sits flat on the surface. If not we correct this by reheating the pins. Afterwards we solder the remaining pins.*

- Die zwei Kondensatoren im Bild (100nF) und die 3 Stecker sind optional und nicht im Bausatz enthalten.

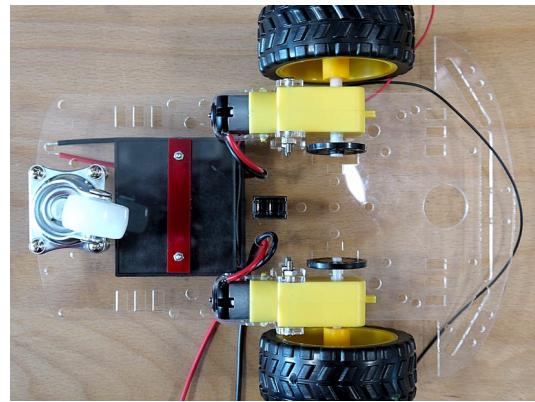
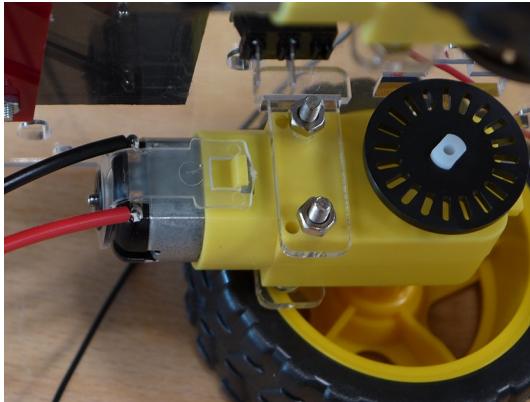
*The two capacitors and the 3 header are optional and not included in the kit.*

- Stecke die 2 Buchsenleisten auf die Pins der WEMOS-Platine auf. Stecke die WEMOS-Platine auf und verlöte sie (siehe Bilder auf der nächsten Seite). Am besten lötet man auch hier zuerst 2 diagonal gegenüberliegende Pins, kontrolliert noch einmal und verlötet dann den Rest.

*After this we solder the remaining 2 8-pin female headers to our PCB. For this we use the microcontroller board as help.*



## 1.4. Motoren anschließen / Connecting the motors



- Löte bei jedem Motor ein rotes und ein schwarzes Stück Draht an. **Schwarz zur Plexiglasplatte hin, rot auf der anderen Seite.** Am besten verzinnt man zuerst die Kontakte und den Draht, danach lötet man beides zusammen.

[Quizfrage: Wie viel Hände möchte ein Elektroniker eigentlich haben (zumindest beim Löten). Spätestens nach diesem Schritt weißt du die Antwort]

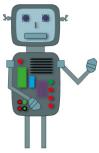
*Solder a red and a black wire to the motor contacts. **The black wire is nearest to the acrylic plate!***

- Die Drähte werden durch die Bohrungen links und rechts vom Schalter geführt.

*The wires pass the holes left and right from the switch.*

- Die Drähte werden nun mit der Platine verlötet, auf der Seite gegenüber von der Diode. Die 4 Lötaugen rechts sind die Motoranschlüsse (links ist die Spannungsversorgung). **Achte auf die Beschriftung.**

*The wires are soldered to the PCB. **Pay attention to the markings on the PCB.***



## 1.5. Batterie und Schalter anschließen / Batteries and switch

- Zuerst den roten Draht des Batteriehalters (Pluspol) durch eine Bohrung in der Platte zum Schalter führen.  
Dort den Draht abknipsen und abisolieren und am Schalter verlöten.  
Das Reststück Draht am anderen Pol des Schalters verlöten und durch die Platte zur Platine führen und dort verlöten. **Achte auf die Beschriftung!**

*Push the red wire of the battery holder (plus) through a hole in the acrylic plate. Shorten the wire and strip the insulation, so we can solder it to the switch. Take the second piece of the wire strip the insulation on two sides and solder it to the second pole of the switch. Get the wire back through the plate to the PCB and solder it to the PCB. Pay attention to the markings on the PCB.*

- Nun den Minuspol des Batteriehalters (schwarz) am zweiten Lötauge verlöten.  
*The black wire of the battery holder (minus) is soldered to the second pin on the PCB. Pay attention to the markings on the PCB.*

## 1.6. Fertigstellung / Finish

- Die Platine mit der Lupe noch einmal kontrollieren ob alle Verbindungen korrekt und nirgends ein Kurzschluss ist.

*Control all soldering with a magnifying glass. Control all connections and make sure that there is no short circuit.*

- Kontrollieren ob der Schalter auf "Aus" (0) steht.  
*The switch has to be on "Off" (0).*
- Setze das IC L293D in den Sockel ein. Achtung: die Kerbe muß zur Seite mit der Diode zeigen!

*Place the motor-driver IC (L293D) in the socket. The notch has to be on the same side than on the socket (direction of the diode).*

- Stecke die Microcontroller Platine auf. **Die USB-Buchse zeigt nach aussen! (zur Diode).**

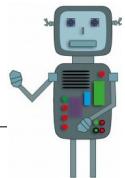
*Mount the microcontroller board. The USB connector faces the diode!*

## 2. Erster Test / First Test

- Schalte den Rover ein. Wenn die blaue LED auf dem WEMOS-Chip leuchtet, und der Rover sich kurz nach links dreht, kannst du schon ein wenig optimistisch sein.

*Switch the rover on.*

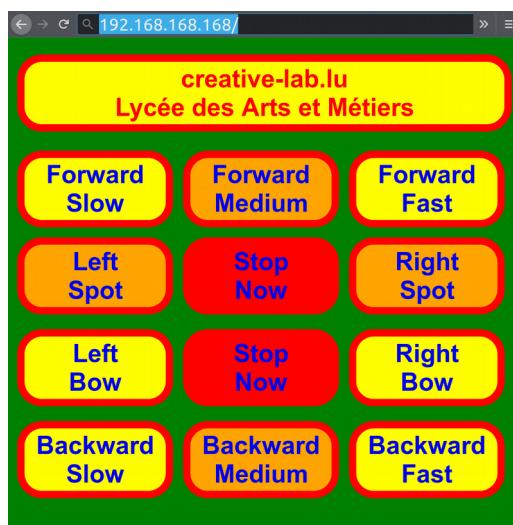
- Suche nun auf deinem Handy nach WiFi-Netzen. Jeder Rover stellt einen eigenen Hotspot zur Verfügung. Er erzeugt also sein eigenes lokales WiFi-Netz, mit seiner eigenen SSID. Die ssid heisst "creativeXX", wobei XX die zweistellige Nummer deines Rovers ist (Etikette auf der WEMOS-Platine). "creative12" wäre also z.B. das WiFi-Netz von Rover 12. Wähle dieses Netz aus und gib als Passwort die SSID ein (z.B. "creative02").



The rover creates his own hot spot (you don't need another existing WiFi network) with his own SSID. The name of the SSID is "creativeXX". XX is the 2 digit number on your WEMOS-board. "creative12" for rover number 12. Choose this network and log in with "creative12"; the password being the same ass the SSID.

- Wenn alles gut geht, meldet dein Handy "Connected". Starte nun einen Internet-Browser und gib als Adresse "192.168.168.168/" ein. Achte auf den Slash "/" am Ende. Im Handy-Display erscheint nun ein grafisches Menü mit einigen Knöpfen, die den Roboter steuern können. Probier es aus!

*After your phone is showing "Connected", you start an Internet browser and type the following URL: "192.168.168.168/". Don't forget the slash (""). Now you see some buttons to control your rover.*



### 3. Software

Hier das Arduino Programm , das natürlich nach Belieben angepasst werden soll.

*Here the basic arduino sketch. Change it at will.*

```
/*
 * Space Mining Rover aka snider1
 * (wemos Webserver with AP and L293D)
 * creative-lab.lu
 */

#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>

const char *ssid = "creative23";           // AP settings
const char *password = "creative23";        // password must have more than 7 characters!!
IPAddress IP_AP(192, 168, 168, 168);
IPAddress MASK_AP(255, 255, 255, 0);

ESP8266WebServer server(80);              // create object with name "server" port 80
const int slow = 500;                      // speed (500 to get min 2,5V)
const int medium = 750;
```



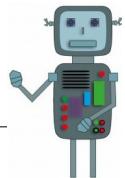
```
const int fast = 1023;           // max. 1023)
const int bowdiff = 150;         // if bigger, lesser radius

const byte LED = 2;             // D4
const byte MotorR1_pin = 16;    // D0 config Wemos pins to motor driver
const byte MotorR2_pin = 0;      // D3
const byte MotorL1_pin = 15;    // D8
const byte MotorL2_pin = 13;    // D7

// here comes the html and css magic
String myhtml = R"=====(
<!DOCTYPE html>
<html>
  <head>
    <title>creative-lab.lu</title>
  </head>
  <style>
    body {font-family: Arial, Verdana, sans-serif;
          font-size: 12px;
          color: red;
          background-color: green;}
    div {margin: 0.2em auto;
         width: 100%;
         clear: both;}
    p {margin: 0.2em;
       padding: 0.1em;
       border-radius: 1em;
       background-color: yellow;
       border: 0.3em red solid;
       font-size: 4em;
       font-weight: bolder;
       text-align: center;}
    p.center {margin: 0.2em auto;
              width: 27%;
              clear: both;}
    p.lam {clear: both;}
    p.main {background-color: orange;
             width: 27%;
             float: left;}
    p.side {width: 27%;
             float: left;}
    p.stop {background-color: red;
             width: 27%;
             float: left;
             margin-bottom: 0.4em;}
    a {display: block;
        text-decoration: none;}
  </style>
  <body>
    <br><p class="lam">creative-lab.lu
      <br/>Lyc&eacute;e des Arts et M&eacute;tier
    <div>
      <p class="side"><a href=/forwards>Forward<br/>Slow</a></p>
      <p class="main"><a href=/forwardm>Forward<br/>Medium</a></p>
      <p class="side"><a href=/forwardf>Forward<br/>Fast</a></p>
    </div><div>
      <p class="main"><a href=/lefts>Left<br/>Spot</a></p>
      <p class="stop"><a href=/halt>Stop<br/>Now</a></p>
      <p class="main"><a href=/rights>Right<br/>Spot</a></p>
    </div><div>
      <p class="side"><a href=/leftb>Left<br/>Bow</a></p>
      <p class="stop"><a href=/halt>Stop<br/>Now</a></p>
      <p class="side"><a href=/rightb>Right<br/>Bow</a></p>
    </div><div>
      <p class="side"><a href=/backwards>Backward<br/>Slow</a></p>
      <p class="main"><a href=/backwardm>Backward<br/>Medium</a></p>
      <p class="side"><a href=/backwardf>Backward<br/>Fast</a></p>
    </div>
  </body>
</html>
)=====;

void setup() {
  pinMode(MotorR1_pin, OUTPUT); // 4 pins to motor driver as output
  pinMode(MotorR2_pin, OUTPUT);
  pinMode(MotorL1_pin, OUTPUT);
  pinMode(MotorL2_pin, OUTPUT);
  pinMode(D4, OUTPUT); // blue LED Pin (D4) as output

  digitalWrite(D4, LOW); // LED on (negative logic)
```



```

action(0,0,0,0);                                // stop the motors
/*digitalWrite(Motor_R_PWM_1, LOW);
digitalWrite(Motor_R_PWM_2, LOW);
digitalWrite(Motor_L_PWM_1, LOW);
digitalWrite(Motor_L_PWM_2, LOW);*/

WiFi.softAP(ssid, password);
WiFi.softAPConfig(IP_AP, IP_AP, MASK_AP);
IPAddress myIP = WiFi.softAPIP();

// handle http requests; root must be called! type: http://192.168.168.168/
server.on("/", handleRoot);                      // (don't forget the "/")
server.on("/forwards", forwards);
server.on("/forwardm", forwardm);
server.on("/forwardf", forwardf);
server.on("/lefts", lefts);
server.on("/rights", rights);
server.on("/halt", halt);
server.on("/leftb", leftb);
server.on("/rightb", rightb);
server.on("/backwards", backwards);
server.on("/backwardm", backwardm);
server.on("/backwardf", backwardf);

server.begin();
server.setContentLength(myhtml.length()); // if not given we get
}                                         // net:err_content_length_mismatch

void loop() {
    server.handleClient();                  // wait for http requests
}

void handleRoot() {
    digitalWrite(LED, LOW);
    server.send(200, "text/html", myhtml); // HTTP response code 200 (alt. 404)
    digitalWrite(LED, HIGH);
}

void forwards() { action(slow,0,slow,0); }
void forwardm() { action(medium,0,medium,0); }
void forwardf() { action(fast,0,fast,0); }
void lefts() { action(slow,0,1023-slow,1023); }
void rights() { action(1023-slow,1023,slow,0); }
void halt() { action(0,0,0,0); }
void rightb() { action(slow,0,slow+bowdiff,0); }
void leftb() { action(slow + bowdiff,0,slow,0); }
void backwards() { action(1023-slow,1023,1023-slow,1023); }
void backwardm() { action(1023-medium,1023,1023-medium,1023); }
void backwardf() { action(1023-fast,1023,1023-fast,1023); }

void action(int MotorR1, int MotorR2, int MotorL1, int MotorL2) {
    digitalWrite(LED, LOW);
    analogWrite(MotorR1_pin, MotorR1);
    analogWrite(MotorR2_pin, MotorR2);
    analogWrite(MotorL1_pin, MotorL1);
    analogWrite(MotorL2_pin, MotorL2);
    server.send(200, "text/html", myhtml);
    digitalWrite(LED, HIGH);
}

```