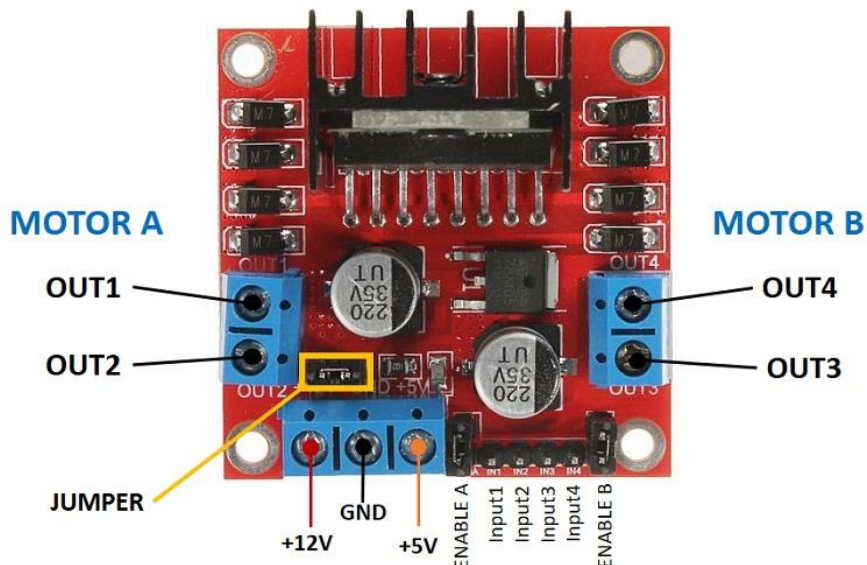
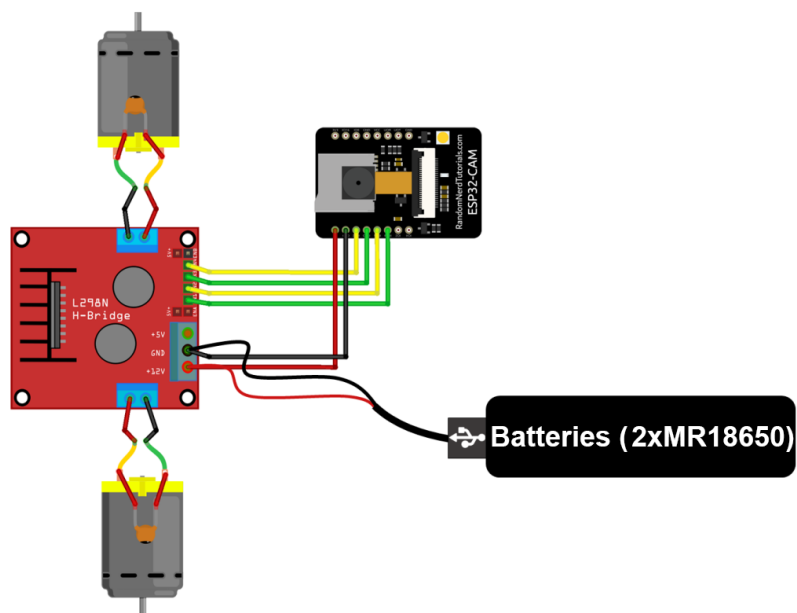


Motorvezérlő

Az L298N egy kétcsatornás, teljes hídból álló motorvezérlő, amelyet széles körben alkalmaznak robotika, járművek, és más vezérelt mozgatórendszerekben. A chip lehetővé teszi két DC motor vagy egy léptető motor egyidejű vezérlését. Képes magas áramokat (akár 2A) kezelni minden csatornán, így alkalmas nagyobb teljesítményű motorokhoz is. Az L298N beépített védelemmel rendelkezik a túláram, túlmelegedés és túlfeszültség ellen, ami megbízható és biztonságos működést biztosít. A modul könnyen használható és széles körben elérhető, mivel számos mikrovezérlővel és más vezérlő elektronikával kompatibilis. Általában az L298N-t tápegységként és jelvezérlőként egyaránt használják, ami lehetővé teszi a motorok precíz és hatékony irányítását a kívánt mozgások eléréséhez.



Bekötés



Így működik az ESP kód

Nézzük meg azokat a releváns részeket, amelyek irányítják a robotot. Definiáljuk a GPIO-kat, amelyek vezérlik a motorokat. Minden motort két tű irányít.

```
#define MOTOR_1_PIN_1 14
#define MOTOR_1_PIN_2 15
#define MOTOR_2_PIN_1 13
#define MOTOR_2_PIN_2 12
```

Amikor megnyomod a gombokat, különböző URL-en keresztül indítasz kéréseket.

```
<table>
  <tr><td colspan="3" align="center"><button class="button" onmousedown="toggleCheckbox('forward');"
  ontouchstart="toggleCheckbox('forward');" onmouseup="toggleCheckbox('stop');"
  ontouchend="toggleCheckbox('stop');">Forward</button></td></tr>
  <tr><td align="center"><button class="button" onmousedown="toggleCheckbox('left');"
  ontouchstart="toggleCheckbox('left');" onmouseup="toggleCheckbox('stop');"
  ontouchend="toggleCheckbox('stop');">Left</button></td><td align="center"><button class="button"
  onmousedown="toggleCheckbox('stop');" onmouseup="toggleCheckbox('stop');">Stop</button></td><td align="center"><button
  class="button" onmousedown="toggleCheckbox('right');" ontouchstart="toggleCheckbox('right');"
  onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">Right</button></td></tr>
  <tr><td colspan="3" align="center"><button class="button" onmousedown="toggleCheckbox('backward');"
  ontouchstart="toggleCheckbox('backward');" onmouseup="toggleCheckbox('stop');"
  ontouchend="toggleCheckbox('stop');">Backward</button></td></tr>
</table>
<script>
  function toggleCheckbox(x) {
    var xhr = new XMLHttpRequest();
    xhr.open("GET", "/action?go=" + x, true);
    xhr.send();
  }
  window.onload = document.getElementById("photo").src = window.location.href.slice(0, -1) + ":81/stream";
</script>
```

Vezérlő kérések:

Előre:

<ESP_IP_ADDRESS>/action?go=forward

Hátra:

/action?go=backward

Balra:

/action?go=left

Jobbra:

/action?go=right

Stop:

/action?go=stop

Kérések kezelése:

```
if(!strcmp(variable, "forward")) {
    Serial.println("Forward");
    digitalWrite(MOTOR_1_PIN_1, 1);
    digitalWrite(MOTOR_1_PIN_2, 0);
    digitalWrite(MOTOR_2_PIN_1, 1);
    digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "left")) {
    Serial.println("Left");
    digitalWrite(MOTOR_1_PIN_1, 0);
    digitalWrite(MOTOR_1_PIN_2, 1);
    digitalWrite(MOTOR_2_PIN_1, 1);
    digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "right")) {
    Serial.println("Right");
    digitalWrite(MOTOR_1_PIN_1, 1);
    digitalWrite(MOTOR_1_PIN_2, 0);
    digitalWrite(MOTOR_2_PIN_1, 0);
    digitalWrite(MOTOR_2_PIN_2, 1);
}
else if(!strcmp(variable, "backward")) {
    Serial.println("Backward");
    digitalWrite(MOTOR_1_PIN_1, 0);
    digitalWrite(MOTOR_1_PIN_2, 1);
    digitalWrite(MOTOR_2_PIN_1, 0);
    digitalWrite(MOTOR_2_PIN_2, 1);
}
else if(!strcmp(variable, "stop")) {
    Serial.println("Stop");
    digitalWrite(MOTOR_1_PIN_1, 0);
    digitalWrite(MOTOR_1_PIN_2, 0);
    digitalWrite(MOTOR_2_PIN_1, 0);
    digitalWrite(MOTOR_2_PIN_2, 0);
}
```