

# Samojezdny / Zdalnie sterowany samochodzik – instrukcja budowy i konfiguracji

---

Wersje pdf można pobrać [tutaj](#)

Ten plik README opisuje, jak zbudować i skonfigurować samojezdny lub zdalnie sterowany samochodzik na bazie podwozia 4WD z Raspberry Pi Zero W jako kontrolerem. Projekt umożliwia autonomię (unikanie przeszkód za pomocą czujnika HC-SR04, śledzenie linii z czujnikami IR) oraz zdalne sterowanie przez Wi-Fi. Poniżej znajdziesz listę komponentów, instrukcje montażu, konfigurację oprogramowania, przykładowy kod oraz wskazówki dotyczące bezpieczeństwa.

---

## Spis treści

- [Samojezdny / Zdalnie sterowany samochodzik – instrukcja budowy i konfiguracji](#)
    - [Spis treści](#)
    - [Lista komponentów](#)
    - [Wymagania oprogramowania](#)
      - [Biblioteki Python](#)
      - [Instalacja](#)
    - [Budowa i montaż](#)
    - [Konfiguracja Raspberry Pi](#)
    - [Przykładowy kod](#)
      - [Sterowanie silnikami](#)
      - [Unikanie przeszkód](#)
      - [Śledzenie linii](#)
      - [Zdalne sterowanie przez Wi-Fi](#)
    - [Uwagi dotyczące bezpieczeństwa](#)
    - [Rozwiązywanie problemów](#)
- 

## Lista komponentów

- **Chassis Rectangle 4WD (ROB-07289)** – Podwozie z 4 kołami i napędem (54,90 zł, 1 szt.).
- **2× L298N (MOD-08227)** – Sterownik silników (19,90 zł/szt., 2 szt., 39,80 zł).
- **HC-SR04 (MOD-01420)** – Czujnik ultradźwiękowy (2–200 cm, 8,90 zł, 1 szt.).
- **Zestaw przewodów żeńsko-męskich 20 cm, 40 szt. (JUS-19621)** – Połączenia (6,50 zł, 1 szt.).
- **Płytki stykowe 400 otworów (JUS-19942)** – Montaż układu (6,90 zł, 1 szt.).
- **Raspberry Pi Zero W 512MB RAM (RPI-08330)** – Kontroler z Wi-Fi (69,90 zł, 1 szt.).
- **2× Cytron Maker Reflect (CTN-16056)** – Czujniki IR do linii (9,50 zł/szt., 2 szt., 19,00 zł).
- **Konwerter poziomów logicznych 3.3 V ↔ 5 V (MOD-01513)** – Bezpieczne połączenie (12,90 zł, 1 szt.).
- **Przetwornica LM2596 3.2V–35V 3A (MOD-02967)** – Obniżanie napięcia (9,90 zł, 1 szt.).
- **Taśma dwustronna piankowa 5 mm × 3 m (MOD-05551)** – Mocowanie (9,50 zł, 1 szt.).
- **Kamera OdSeven OV5647 5Mpx (ODS-05619)** – Wizja maszynowa (24,90 zł, 1 szt.).
- **Karta microSD 16 GB Goodram (KAP-02123)** – System operacyjny (11,39 zł, 1 szt.).

- **Koszyk na 2× 18650 (AKC-05240)** – Zasilanie (3,99 zł, 1 szt.).
  - **Ładowarka XTAR MC2 (AKC-08543)** – Ładowanie akumulatorów (26,90 zł, 1 szt.).
  - **2× Ogniwo 18650 XTAR 2600 mAh (ZAS-09067)** – Zasilanie (27,90 zł/szt., 2 szt., 55,80 zł).
  - **Przełącznik On-Off (UCC-08382)** – Włączanie/wyłączanie (2,90 zł, 1 szt.).
  - **Zestaw radiatorów (MOD-09871)** – Chłodzenie RPi (2,30 zł, 1 szt.).
  - **Bezpiecznik Midi 5 A (KAB-08810)** – Ochrona przed zwarcie (1,65 zł, 1 szt.).
  - **Kondensator elektrolityczny 470 µF (PAS-04384)** – Stabilizacja napięcia (2,50 zł, 1 szt.).
  - **Przewody męsko-męskie (JUS-19950)** – Połączenia pomocnicze (7,90 zł, 1 szt.).
  - **Przetwornik A/C MCP3008 (UCC-02358)** – Dla czujników IR (15,90 zł, 1 szt.).
- 

## Wymagania oprogramowania

### Biblioteki Python

- **RPi.GPIO** – Sterowanie GPIO.
- **opencv-python** – Obsługa kamery.
- **Flask** – Zdalne sterowanie przez Wi-Fi.
- **spidev** – Obsługa MCP3008 dla czujników IR.

### Instalacja

```
sudo apt update
sudo apt install python3-pip python3-rpi.gpio python3-spidev
pip3 install opencv-python flask
```

---

## Budowa i montaż

### 1. Złożenie podwozia:

- Złóż podwozie 4WD (ROB-07289) według instrukcji. Przykręć silniki i koła.
- Sprawdź, czy koła obracają się swobodnie.

### 2. Montaż komponentów:

- Użyj taśmy piankowej (MOD-05551) do zamocowania Raspberry Pi, płytki stykowej, przetwornicy i koszyka na akumulatory.
- Zamocuj czujnik HC-SR04 z przodu (dla unikania przeszkód).
- Czujniki IR (CTN-16056) umieść pod spodem, 1–2 cm od podłoża, w linii kół.

### 3. Zasilanie:

- Włóż dwa akumulatory 18650 (ZAS-09067) do koszyka (AKC-05240).
- Podłącz koszyk do przetwornicy LM2596 przez bezpiecznik 5 A (KAB-08810) i przełącznik (UCC-08382).
- Ustaw przetwornicę na 5.0 V (sprawdź multimetrem).
- Dodaj kondensator 470 µF (PAS-04384) na wyjściu przetwornicy.

#### 4. Kamera:

- Podłącz kamerę OV5647 (ODS-05619) do gniazda CSI na Raspberry Pi.
- Upewnij się, że taśma jest dobrze włożona.

#### 5. Radiatory:

- Przyklej radiatory (MOD-09871) na Raspberry Pi, aby zapobiec przegrzaniu.

#### 6. Połączenia:

- Użyj przewodów żeńsko-męskich i męsko-męskich do połączenia komponentów.
- Podłącz moduły L298N do silników i Raspberry Pi (użyj GPIO do sterowania).
- Czujnik HC-SR04 podłącz przez konwerter poziomów logicznych (MOD-01513).
- Czujniki IR podłącz do MCP3008, a ten do SPI na Raspberry Pi.

---

## Konfiguracja Raspberry Pi

### 1. Przygotowanie karty microSD:

- Pobierz Raspberry Pi Imager.
- Wgraj **Raspberry Pi OS Lite** na kartę microSD (KAP-02123).
- W katalogu `/boot` utwórz plik `ssh` (bez rozszerzenia).
- Skonfiguruj Wi-Fi w `/boot/wpa_supplicant.conf`:

```
country=PL
network={
    ssid="TwojaSiecWiFi"
    psk="TwojeHaslo"
}
```

### 2. Uruchomienie:

- Włóż kartę microSD, podłącz zasilanie (5 V z przetwornicy).
- Połącz się przez SSH: `ssh pi@raspberrypi.local` (hasło: `raspberry`).
- Zaktualizuj system:

```
sudo apt update && sudo apt upgrade
```

### 3. Włączenie interfejsów:

- Uruchom `sudo raspi-config`:
  - Włącz kamerę: **Interfacing Options** → **Camera** → Enable.
  - Włącz SPI: **Interfacing Options** → **SPI** → Enable.

- Ustaw strefę czasową i język w **Localisation Options**.

#### 4. Instalacja bibliotek:

```
sudo apt install python3-pip python3-rpi.gpio python3-spidev
pip3 install opencv-python flask
```

#### 5. Test kamery:

- Wykonaj: `raspistill -o test.jpg`.
- Jeśli plik się utworzy, kamera działa.

---

## Przykładowy kod

### Sterowanie silnikami

Podstawowy kod do sterowania ruchem samochodziku.

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
pins = [17, 18, 27, 22] # Przykładowe piny dla L298N
for pin in pins:
    GPIO.setup(pin, GPIO.OUT)

def forward():
    GPIO.output(17, GPIO.HIGH)
    GPIO.output(18, GPIO.LOW)
    GPIO.output(27, GPIO.HIGH)
    GPIO.output(22, GPIO.LOW)

def stop():
    for pin in pins:
        GPIO.output(pin, GPIO.LOW)

try:
    forward()
    time.sleep(2)
    stop()
except KeyboardInterrupt:
    GPIO.cleanup()
```

### Unikanie przeszkód

Kod do wykrywania przeszkód za pomocą HC-SR04.

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(6, GPIO.OUT) # TRIG
GPIO.setup(13, GPIO.IN) # ECHO

def get_distance():
    GPIO.output(6, True)
    time.sleep(0.00001)
    GPIO.output(6, False)
    start = time.time()
    while GPIO.input(13) == 0:
        start = time.time()
    while GPIO.input(13) == 1:
        end = time.time()
    return ((end - start) * 17150)

try:
    while True:
        dist = get_distance()
        print(f"Odległość: {dist:.1f} cm")
        if dist < 20:
            stop()
        else:
            forward()
            time.sleep(0.5)
except KeyboardInterrupt:
    GPIO.cleanup()
```

## Śledzenie linii

Kod do śledzenia linii za pomocą czujników IR i MCP3008.

```
import spidev
import RPi.GPIO as GPIO

spi = spidev.SpiDev()
spi.open(0, 0)

def read_adc(channel):
    adc = spi.xfer2([1, (8 + channel) << 4, 0])
    return ((adc[1] & 3) << 8) + adc[2]

try:
    while True:
        left = read_adc(0)
        right = read_adc(1)
        if left > 500 and right > 500:
            forward()
        elif left > 500:
```

```
        # Skręć w prawo
        GPIO.output(27, GPIO.LOW)
    elif right > 500:
        # Skręć w lewo
        GPIO.output(17, GPIO.LOW)
    else:
        stop()
        time.sleep(0.1)
except KeyboardInterrupt:
    GPIO.cleanup()
    spi.close()
```

## Zdalne sterowanie przez Wi-Fi

Prosty serwer Flask do sterowania przez przeglądarkę.

```
from flask import Flask
import RPi.GPIO as GPIO

app = Flask(__name__)
GPIO.setmode(GPIO.BCM)

@app.route('/control/<cmd>')
def control(cmd):
    if cmd == 'forward':
        boron()
    elif cmd == 'stop':
        stop()
    return "OK"

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000)
```

**Uruchomienie:** Zapisz jako `app.py`, uruchom: `python3 app.py`. W przeglądarce wpisz:

`http://<IP_Raspberry_Pi>:5000/control/forward`.

---

## Uwagi dotyczące bezpieczeństwa

### ⚠ Ostrzeżenia:

- **Akumulatory 18650:** Nie zwarzaj, nie przeładuj, nie rozładuj poniżej 3 V. Używaj ładowarki XTAR MC2.
- **Zasilanie:** Ustaw przetwornicę na 5.0 V (sprawdź multimetrem).
- **Bezpiecznik 5 A:** Obowiązkowy dla ochrony przed zwarcieniem.
- **Kondensator 470 µF:** Stabilizuje napięcie na przetwornicy.
- **Testowanie:** Sprawdzaj każdy moduł osobno przed pełnym montażem.
- **Konwerter poziomów:** Używaj dla HC-SR04 (pin ECHO) i innych modułów 5 V.

💡 **Uwaga:** Sprawdź napięcia (3.3 V/5 V) przed podłączeniem.

---

## Rozwiązywanie problemów

- **Raspberry Pi nie startuje:**
  - Sprawdź napięcie przetwornicy (5.0 V).
  - Upewnij się, że karta microSD ma system.
- **Silniki nie działają:**
  - Sprawdź jumper 5 V na L298N.
  - Testuj silniki osobno.
- **HC-SR04 daje błędne odczyty:**
  - Upewnij się, że używasz konwertera poziomów logicznych.
  - Sprawdź połączenia TRIG/ECHO.
- **Czujniki IR nie działają:**
  - Włącz SPI w `raspi-config`.
  - Sprawdź odległość czujników od podłoża (1–2 cm).
- **Kamera nie działa:**
  - Sprawdź taśmę CSI.
  - Włącz kamerę w `raspi-config`.
- **Wi-Fi nie działa:**
  - Sprawdź plik `wpa_supplicant.conf`.
  - Upewnij się, że Raspberry Pi jest w zasięgu sieci.