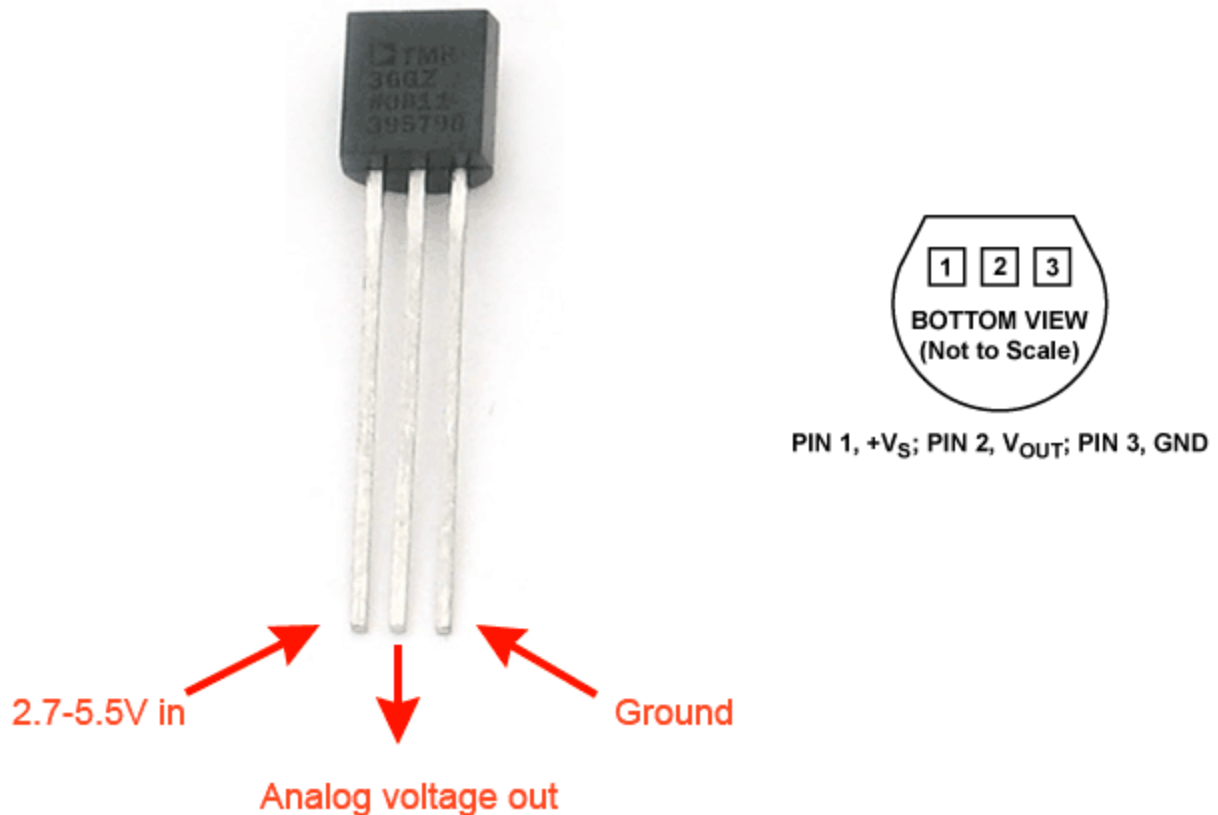


Pomiar temperatury



Czujnik temperatury TMP36 przekazuje na wyjście napięcie proporcjonalne do temperatury otoczenia, zatem określenie bieżącej temperatury wymaga tylko prostej konwersji odczytywanej wartości. Na przykład temperatura 25°C jest reprezentowana przez napięcie 750 mV, a każda zmiana temperatury o 1 stopień odpowiada zmianie napięcia o 10 mV. Czujnik TMP36 może służyć do mierzenia temperatur z przedziału od -40°C do +125°C.

Funkcja `analogRead()` zwraca wartość z przedziału od 0 do nieco poniżej 5000 mV (5V). Wystarczy pomnożyć wartość zwracaną przez funkcję `analogRead()` przez $(5000/1024)$, aby uzyskać rzeczywiste napięcie przekazywane przez czujnik. Należy jeszcze odjąć wartość 500 (czyli zastosowane w czujniku TMP36 przesunięcie niezbędne do reprezentowania temperatur poniżej 0 °C) i podzielić otrzymaną wartość przez 10 – uzyskana w ten sposób liczba będzie reprezentowała temperaturę w stopniach Celsjusza.

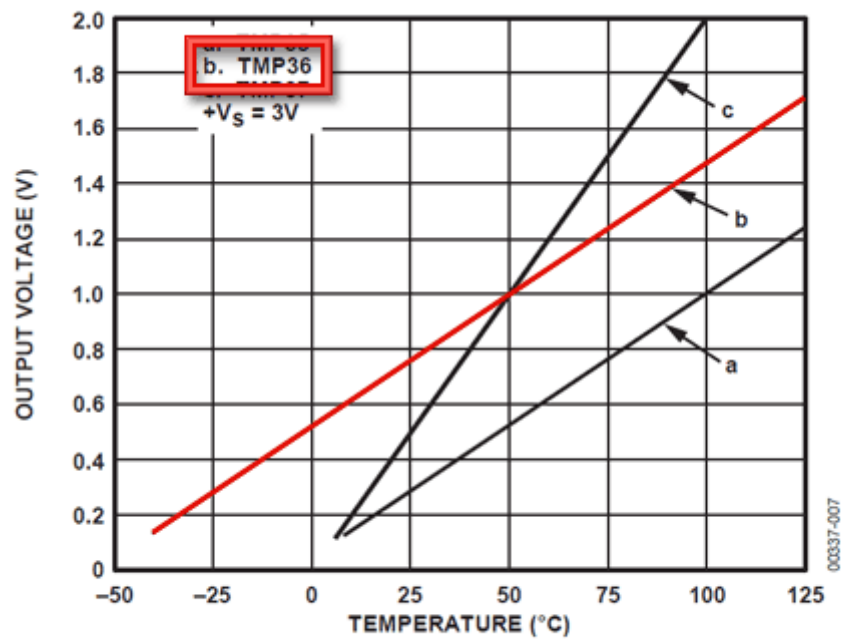
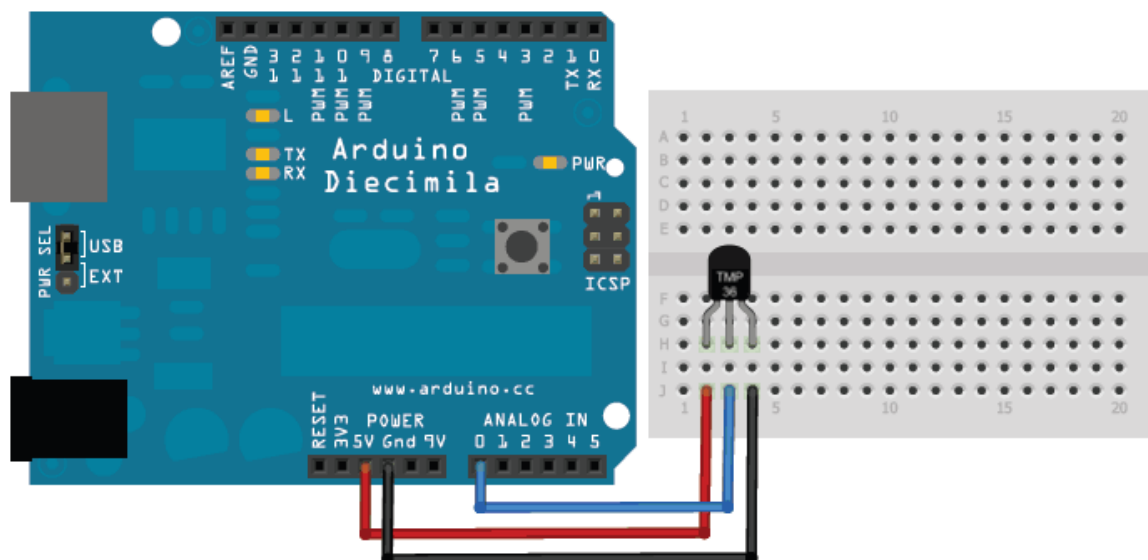


Figure 6. Output Voltage vs. Temperature



//Projekt - wyświetlanie temperatury w oknie monitora portu szeregowego (TMP36)

```
float celsius = 0;
```

```
float fahrenheit = 0;
```

```
void setup()
```

```
{
```

```
  Serial.begin(9600);
```

```
}
```

```

void findTemps()
{
    float voltage = 0;
    float sensor = 0;
    //odczytuje wartość z czujnika temperatury i konwertuje ją
    //na stopnie Celsjusza i Fahrenheita
    sensor = analogRead(0);
    voltage = (sensor*5000)/1024;    //konwertuje wartość z czujnika na mV
    voltage = voltage - 500;        //odejmuje przesunięcie napięcia
    celsius = voltage / 10;         // zamiana mV na stopnie Celsjusza
    fahrenheit = (1.8 * celsius) + 32; //konwertuje stopnie Celsjusza na Fahrenheita
}

void displayTemps()
{
    Serial.print(" Temperatura wynosi ");
    Serial.print(celsius, 0);
    Serial.print(" st. C / ");
    Serial.print(fahrenheit, 0);
    Serial.println(" st. F");
}

void loop()
{
    findTemps();
    displayTemps();
    delay(1000);
}

```

Pomiar odległości



Ultradźwiękowy czujnik odległości HC-SR04

HC-SR04 to czujnik ultradźwiękowy działający w zakresie od 2 do 200 cm, zasilany napięciem 5V. Mierzony za jego pomocą czas trwania sygnału wejściowego, jest proporcjonalny do mierzonej odległości. Typowy pobór prądu podczas pomiaru wynosi około 15mA. Posiada cztery piny, dwa zasilające oraz dwa do wyzwolenia i odczytania pomiaru.

Specyfikacja

- Napięcie zasilania: 5 V
- Średni pobór prądu: 15 mA
- Zakres pomiarowy: 2 - 200 cm
- Wyjście: sygnał częstotliwościowy
- Częstotliwość pracy: 40 kHz
- Kąt pomiaru 15°
- Wymiary: 45 x 20 x 15 mm

Aby rozpocząć pomiar odległości należy podać na pin **TRIG** impuls napięciowy w stanie wysokim **5V** przez czas **10μs**, następnie dokonywany jest pomiar odległości przy pomocy fali ultradźwiękowej o częstotliwości **40kHz**, która odbijając się od napotkanej przeszkody, wraca do czujnika. Kiedy pomiar zostanie wykonany, na pin **ECHO** wystawiany jest sygnał, którego czas trwania zależy od odległości do przeszkody.

Pierwszą metodę pomiaru opisuje prosta zależność, w której mierzony jest czas (**T1**) w sekundach od momentu wyzwolenia sygnału na wejściu **TRIG** do momentu pojawienia się odpowiedzi na pinie **ECHO**:

$$\text{odległość} = (T1 * \text{prędkość rozchodzenia się dźwięku}) / 2$$

Interesuje nas prędkość rozchodzenia się dźwięku w powietrzu, a więc 340 m/s:

$$\text{odległość [m]} = (T1 [s] * 340 [m/s]) / 2$$

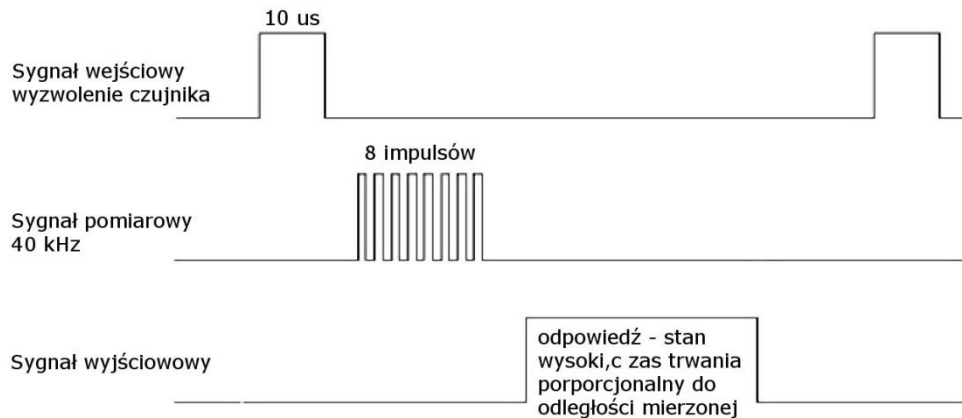
HC-SR04 posiada jednak opcję, która pozwoli nam na określenie odległości od przeszkody na podstawie czasu trwania sygnału **ECHO (T2)** mierzoną w mikrosekundach. Łatwo policzyć więc, że fala dźwiękowa pokonuje jeden centymetr w czasie **29 μs**, a więc ułatwiając sobie powyższą zależność, otrzymamy wzór i wynik w centymetrach:

$$\text{odległość [cm]} = T2 [\mu s] / 58$$

Jeśli interesuje nas wynik w calach, powyższy wzór będzie wyglądał następująco:

$$\text{odległość [inch]} = T2 [\mu s] / 148$$

Dokładność pomiaru może osiągać wartość ± 3 mm. Zaleca się ponawiać pomiar po czasie nie krótszym niż 60 ms.



Przykład 1

```
#include <LiquidCrystal.h>
```

```
int TX = 14; //port nadajnika Ultrasonic
int RX = 15; //port odbiornika Ultrasonic
int CM; //odległość w cm
long TIME; //długość powrotnego impulsu w μs
```

```
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);
```

```
void setup()
{
  lcd.begin(16,2);
  pinMode(TX, OUTPUT);
  pinMode(RX, INPUT);
}
```

```
void loop()
{
  pomiar_odleglosci();
  lcd.clear();
  lcd.setCursor(0,0); //w górnej linii LCD - czas
  lcd.print(TIME);
  lcd.setCursor(0,1); // w dolnej linii LCD - cm
```

```

// szerokość odbitego impulsu w  $\mu$ s podzielone przez 58 to odległość w cm
lcd.print(TIME / 58);

delay(1000);
}

void pomiar_odleglosci ()
{
  // impuls 10  $\mu$ s inicjalizujący
  digitalWrite(TX, HIGH);
  delayMicroseconds(10);
  digitalWrite(TX, LOW);
  TIME = pulseIn(RX, HIGH);
}

```

Przykład 2

//pomiar odległości za pomocą ultradźwięków – wykorzystanie biblioteki Ultrasonic

```

#include <Ultrasonic.h>
#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7); //RS,EN, D0, D1, D2, D3
Ultrasonic ultrasonic(14,15); // TRIG, ECHO

void setup() {
  lcd.begin(16, 2);
  lcd.print("testing...");
}

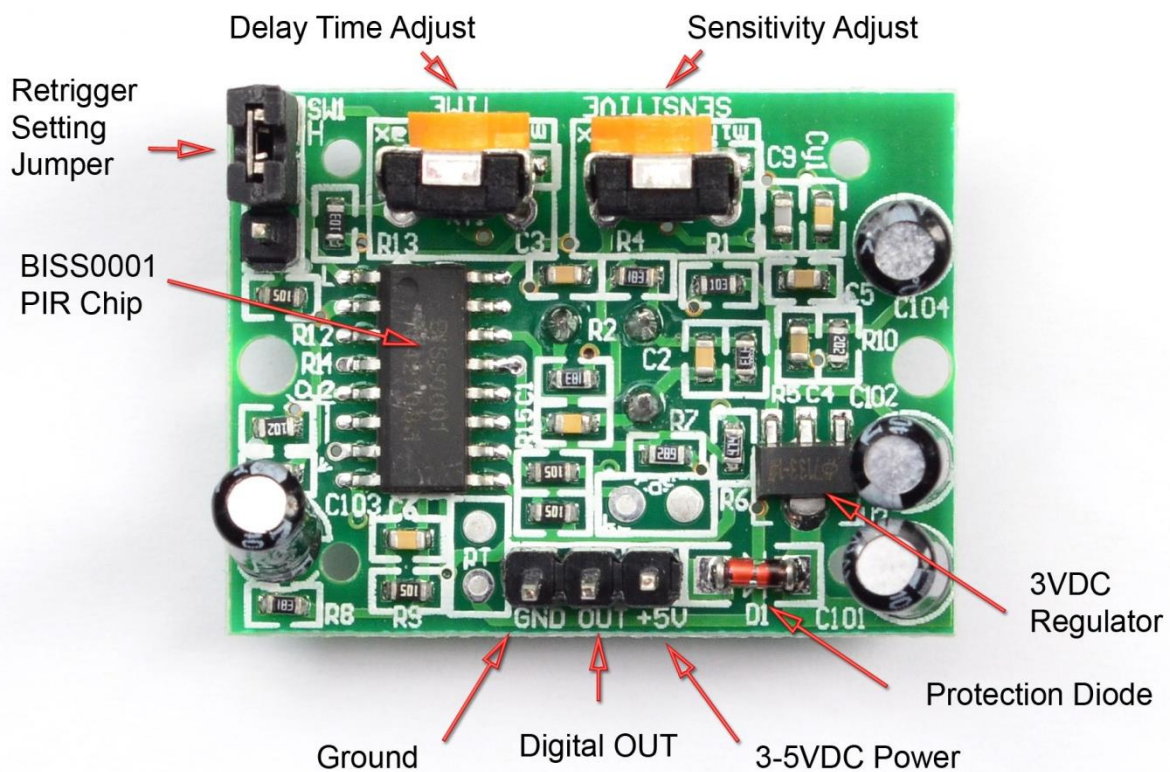
void loop()
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(ultrasonic.read());
  lcd.print("cm");

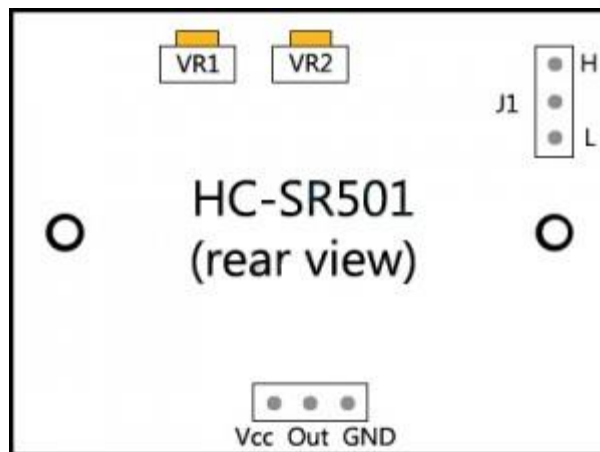
  delay(200);
}

```

Czujnik ruchu

Sensory PIR pozwalają wykryć ruch, reagują na pojawienie się lub zniknięcie człowieka z pola widoczności czujnika. Są małe, tanie, łatwe w użyciu. Znajdują zastosowanie w gospodarstwach domowych oraz biurach. Zwykle nazywane są czujnikami PIR, "Passive Infrared", "Pyroelectric" lub "IR motion"





Widok od strony elementów

Moduł HC-SR501 (DYP-ME003x1) jest to czujnik ruchu oparty na czujniku podczerwieni. Czujka posiada:

- Regulowaną czułość i czas
- Zakres detekcji maks. 7 m
- Kąt wykrywania: do 100 stopni
- Zakres napięcia zasilającego: 5V-20V (DC)
- Pobór prądu (czuwanie): mniej niż 50uA
- Poziom wyjściowy napięcia: wysoki 3V / nisko 0V
- Temperatura pracy: -15 do +70 stopni Celsjusza
- Sposób wyzwalania: L niepowtarzalne / H powtarzalne
- Rozmiar płytki PCB: 32mm x 24mm

Przykład:

`void setup()`

```
{
  pinMode(13,OUTPUT);
  pinMode(15,INPUT);
}
```

`void loop()`

```
{
  digitalWrite(13,digitalRead(15));
}
```