

picoca workshop

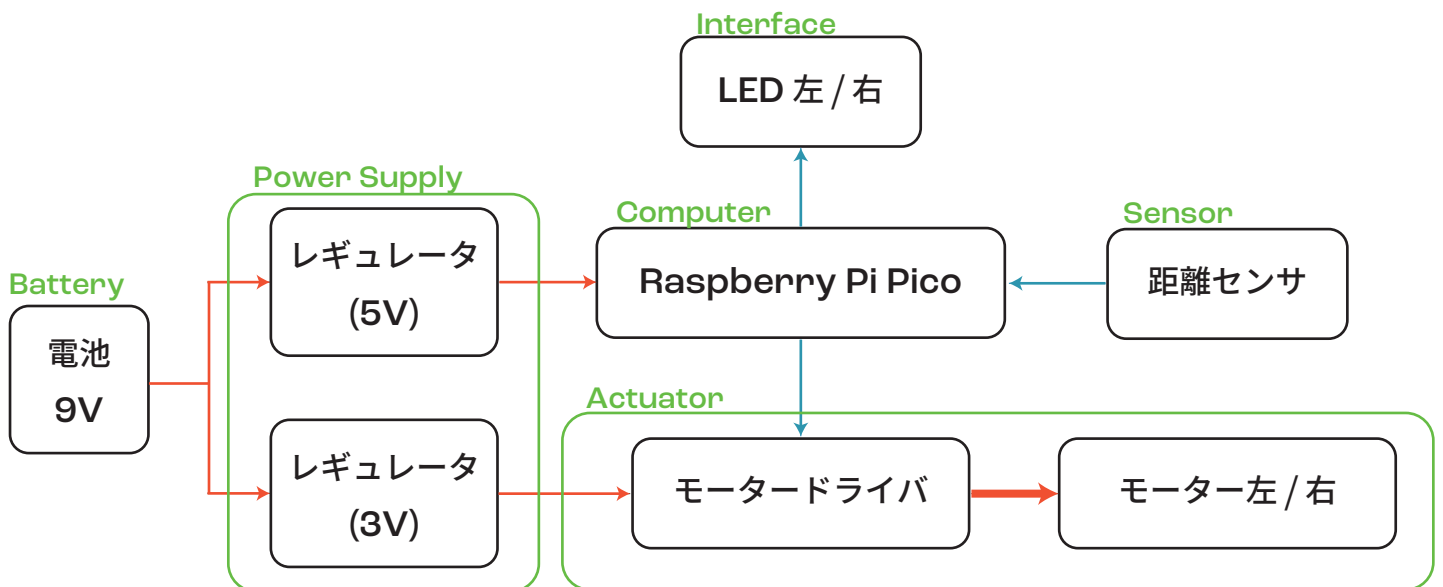
氏名 _____

✓ 道具のチェック

- ☐ picoca 本体
- ☐ USB ケーブル
- ☐ 006P 電池 (9V)

- ☐ PC(Arduino IDE インストール済)
- ☐ PC 充電ケーブル
- ☒ ワークシート

🔌 picoca の構成



🔌 Raspberry Pi Pico に書き込むための準備

URL にアクセス <https://michica.jp/picoca-ws/>

- 必要なファイルをダウンロードしましょう。
- Arduino IDE のセットアップを行いましょう。
- 道具が全て揃っているかチェックしましょう

🔌 LED を点滅させる (for 文について学ぼう)

複数回、同じ命令を繰り返したい場合には「for 文」を使うことができます。
for 文で囲った部分を繰り返すことができます。

```
for(①i=0; ②i<5; ③i++){  
    繰り返したい命令();  
    繰り返したい命令();  
}
```

- ① 繰り返しをカウントするための変数を初期化する。例：変数 *i* を 0 にセット
- ② 繰り返しの条件式を決める。例：変数 *i* が 5 より小さい場合に繰り返す
- ③ 繰り返されるたびに実行しカウントを行う。例：変数 *i* の値を +1 する

led.ino

```
void loop() {  
    // LEDを5回点滅させる  
    for(int i=0; i<5; i++){  
        // LEDを点灯させる  
        digitalWrite(LeftLED, HIGH);  
        digitalWrite(RightLED, HIGH);  
        delay(500);  
        // LEDを消灯  
        digitalWrite(LeftLED, LOW);  
        digitalWrite(RightLED, LOW);  
        delay(500);  
    }  
    // 2秒待つ  
    delay(2000);  
}
```

5 回繰り返す

変数 *i* が 5 以下なら
for 文の中を繰り返す

for 文が
繰り返されるたびに一度実行される
変数 *i* を +1 する

距離を測る (if 文について学ぼう)

条件によって実行したい命令を分けたい場合には「if 文」が使えます。

条件が当てはまれば if 文で囲った部分を実行することができます。

else 文と組み合わせることで複数の条件を追加して分岐処理が可能になります。

```
if(depth <= 20){  
    命令 1();  
}  
else if(depth > 50){  
    命令 2();  
}  
else{  
    命令 3();  
}
```

① if 文では () の中に条件を記入する。
例：depth が 20 以下の場合

② else if(条件式) とすることで、
他の条件を追加することができる。
例：depth が 50 より大きい場合

③ else 文では他の条件に
一致しなかった場合に実行される。
例：①と②の条件以外

depth.ino

```
void loop() {  
    int depth = 0;  
    depth = getDepth();  
    Serial.print(depth);  
    Serial.println("cm");  
  
    // もし距離センサが20cmより小さかったらLEDを点灯させる  
    if(depth < 20){  
        digitalWrite(LeftLED, HIGH);  
        digitalWrite(RightLED, HIGH);  
    }else{  
        digitalWrite(LeftLED, LOW);  
        digitalWrite(RightLED, LOW);  
    }  
    delay(500);  
}
```

距離を取得する

if 文
depth が 20 より
小さかったら実行

else 文
if 文の条件式以外
だったら実行

モーターを回転させる

モーターの回転させるためには、モーターを基板の A と B のピンに接続します。接続するピンの向きを反対に指すとどうなるのかを実験してみましょう。

```
drive_A(50);  
drive_B(50);
```

`drive_A()`; `drive_B()`; はそれぞれモーターを回転させるための命令です。それぞれの () の中には -255~255 までの数値を入力することができ、その数値で、モーターの回転方向や回転速度を変更することができます。

motor.ino

```
void loop() {  
  // モーターを動かす  
  drive_A(50);  
  drive_B(50);  
  delay(1000);  
  drive_A(-50);  
  drive_B(-50);  
  delay(1000);  
}
```

picoca で遊んでみる

picoca を使って、遊んでみましょう。

以下のような課題に取り組んでみましょう！

- ○○cm のところで停止するプログラム
- まえの picoca についていくプログラム
- 簡単な迷路を移動するプログラム
- 曲がる時にウィンカーを出すプログラム



michica
STEAM Education Hub



<https://michica.jp/sns/>