

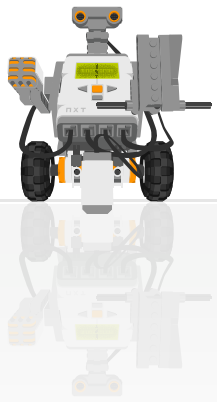
# 実践ロボットプログラミング

LEGO Mindstorms NXT で目指せロボコン!

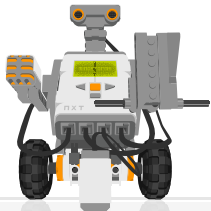
WEB : <http://robot-programming.jp/>

著者 : 藤吉弘亘, 藤井隆司, 鈴木裕利, 石井成郎

E-mail : [support@robot-programming.jp](mailto:support@robot-programming.jp)



## ■関数化



## 関数化1 (p.72)

関数化

```
task main()
{
  OnFwd(OUT_AC, 75);
  Wait(10000);

  OnFwd(OUT_C, 75);
  OnRev(OUT_A, 75);
  Wait(2000);

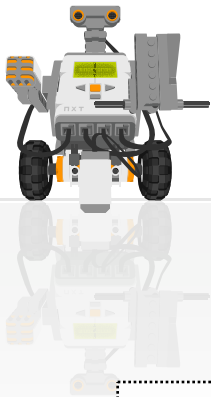
  Off(OUT_AC);
}
```

```
void forward() {
  OnFwd(OUT_AC, 75);
  Wait(10000);
}
```

```
void turn_right() {
  OnFwd(OUT_C, 75);
  OnRev(OUT_A, 75);
  Wait(2000);
}
```

```
task main()
{
  forward();
  turn_right();
  Off(OUT_AC);
}
```

呼び出し

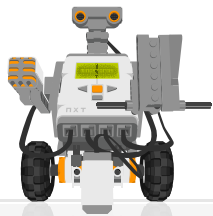


## 関数化2

```
void forward() {  
    OnFwd(OUT_AC, 75);  
    Wait(5000);  
}  
void turn_right() {  
    OnFwd(OUT_C, 75);  
    OnRev(OUT_A, 75);  
    Wait(2000);  
}  
task main()  
{  
    forward();  
    turn_right();  
    Off(OUT_AC);  
}
```

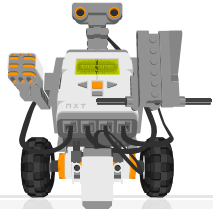


```
void forward(int time) {  
    OnFwd(OUT_AC, 75);  
    Wait(time);  
}  
void turn_right(int time) {  
    OnFwd(OUT_C, 75);  
    OnRev(OUT_A, 75);  
    Wait(time);  
}  
task main()  
{  
    forward(5000);  
    turn_right(2000);  
    Off(OUT_AC);  
}
```



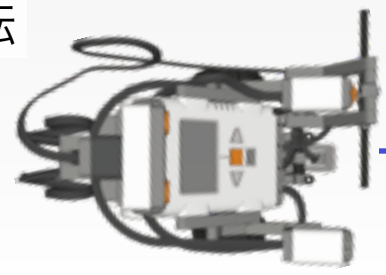
## 引数を持つ関数

- ・ 関数の書式：void 関数名(引数){ }
- ・ 引数は関数内のみ有効



# 一周するには？

回転



前進



回転

前進



回転

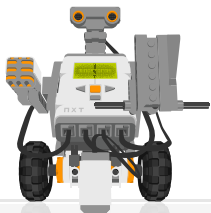
前進



前進



回転



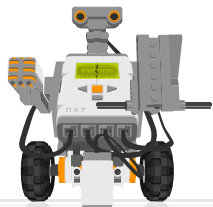
## 1周するプログラム

```
void forward(int time) {
    OnFwd(OUT_AC, 75);
    Wait(time);
}

void turn_right(int time) {
    OnFwd(OUT_A, 75);
    OnRev(OUT_C, 75);
    Wait(time);
}

task main()
{
    int i;
    for(i=1; i<=4; i++){
        forward(5000);
        turn_right(500);
    }
}
```

関数を利用すると記述するプログラムが読みやすくなる



## #defineによる定義

```
#define FWD 5000
#define TURN90 500

void forward(int time) {
    OnFwd(OUT_AC, 75);
    Wait(time);
}

void turn_right(int time) {
    OnFwd(OUT_A, 75);
    OnRev(OUT_C, 75);
    Wait(time);
}
```

```
task main()
{
    int i;
    for(i=1; i<=4; i++){
        forward(FWD);
        turn_right(TURN90);
    }
}
```

時間を調整するには、TURN90の値を変えるだけで全てに反映