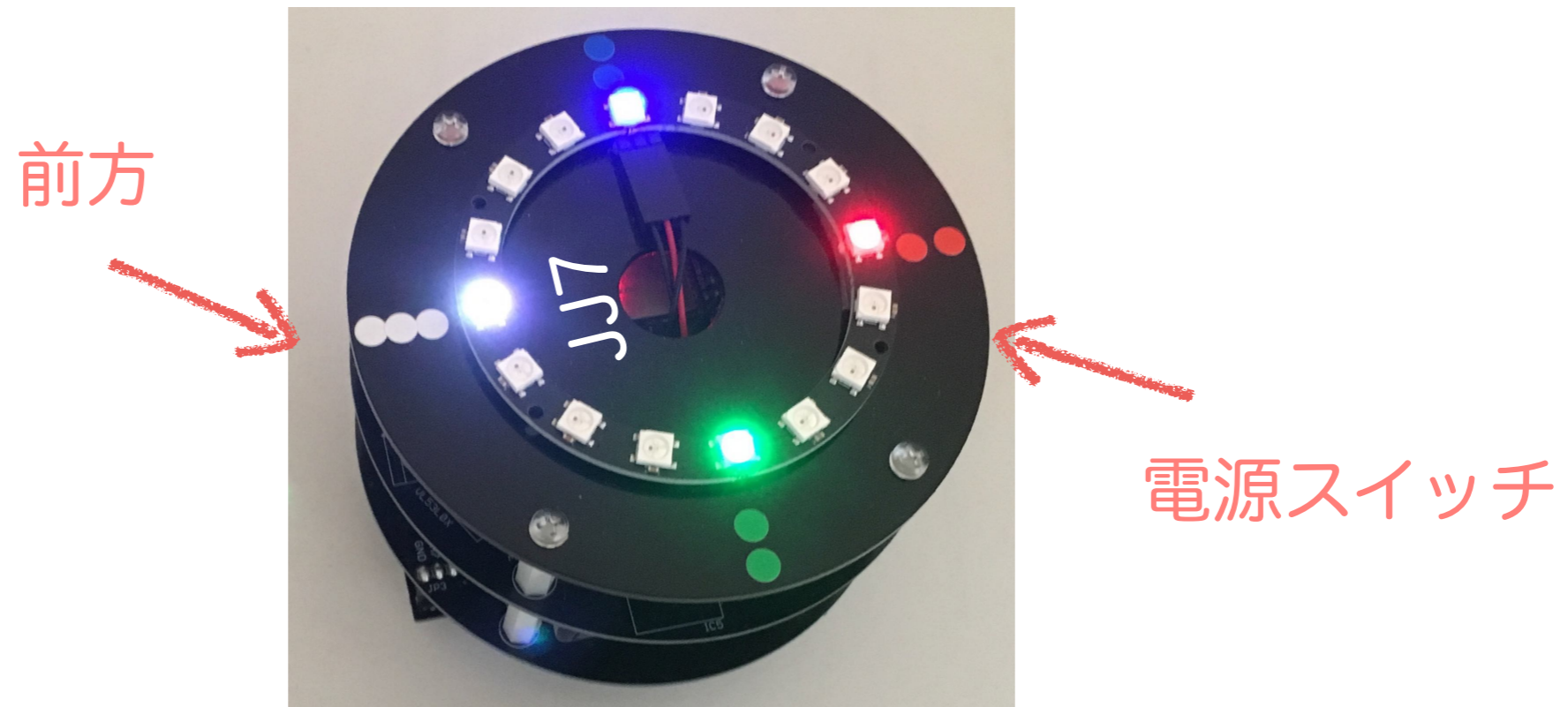


令和1年度 人と地域共創センター公開講座（秋・冬）

## AI/IoTセンサのしくみを知ろう（応用編）



### 第3回 ロボットのモーター1（基本動作）

川上 博

2019/10/26

## 講座内容

---

- ・ 講師：辻 明典（徳島大学技術支援部）  
桑折範彦（徳島大学名誉教授）  
川上 博（徳島大学名誉教授）

- ・ 土曜日：10:00～11:30

- ・ 日程：

- ① 10/5 概要，環境設定，配布部品の確認
- ② 10/19 復習
- ③ 10/26 ロボットのモーター 1
- ④ 11/9 ロボットのモーター 2
- ⑤ 11/16 ロボットのセンサー 1
- ⑥ 11/30 ロボットのセンサー 2
- ⑦ 12/7 ロボットの制御 1
- ⑧ 12/14 ロボットの制御 2
- ⑨ 12/21 ロボットの制御 3

## 今日のテーマ：JJ7ロボットを動かす(1)

---

- JJ7の運動：速度( $v_R$ ,  $v_L$ )と時間 $\tau$ を与えてJJ7を動かす
- スケッチ で与えるプログラムは？  
`motor(spL, spR, oL, oR);`  
`delay( $\tau$ );`
- 4つの基本的な動き：前進・後退, 左回転・右回転
- Turtle Graphics：4つの動きを組み合わせて図形を描く

Ex301：とりあえず動くスケッチをつくる

Ex302：Ex301のスケッチをファイルを分割する

Ex303：リングLEDで前後・左右を表示する

Ex304：Ex302にリングLEDを付加する(turtle graphics)

## JJ7 ロボットを動かす前に

---

- ・ 車輪やタイヤの取り付け等は良好か？ 走行中・旋回中に  
変な音はしないか？
- ・ キャスターの取り付け・遊びは適切か？
- ・ 電池の装着やスイッチ類の操作に注意して動かそう

---

### 目 標

センサーなどからのフィードバックなしに、JJ7自身の力で  
できる限り良好な走行（直進，旋回（回転））をさせたい

## Ex301 を実行する：スケッチの基本構造

---

使用する定数などの定義

```
void setup() {
```

使用するポートや変数等の初期設定

```
}
```

```
void loop() {
```

```
  while (digitalRead(SW_PIN) == HIGH);
```

```
  testFwd(tau);
```

SWを押す毎に1回実行される

```
}
```

```
int sp=330, tau=1000;
```

spとtauを色々変えて、前進運動を実測する

```
void testFwd(int tau){
```

```
  left_offset = 0; right_offset = 0;
```

```
  motor(sp, sp, HIGH, HIGH); //forward
```

```
  delay(tau);
```

```
  motor(0, 0, LOW, LOW); //stop
```

```
  delay(500);
```

```
}
```

```
void motor(int left, int right, int left_c, int right_c) {  
    digitalWrite(MOTOR_L_CWCCW, left_c);  
    digitalWrite(MOTOR_R_CWCCW, right_c);  
    ledcWrite(CH_0, left);  
    ledcWrite(CH_1, right);  
}
```

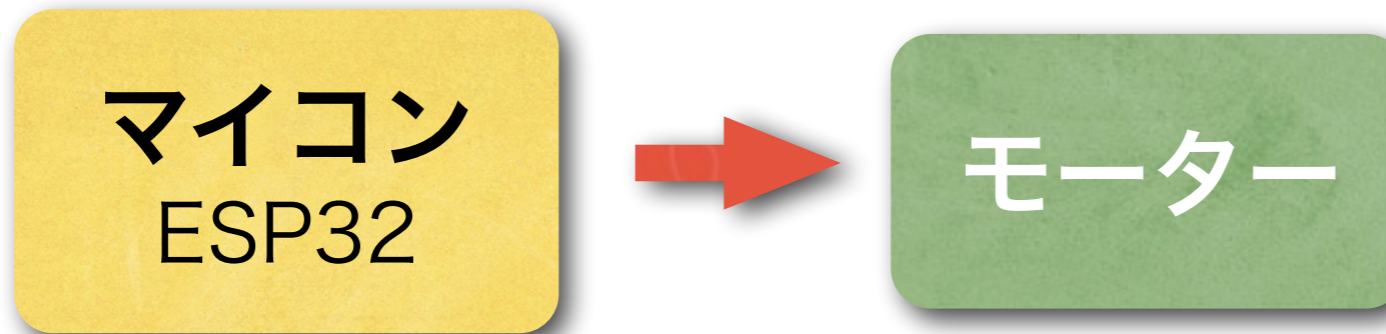
距離 = 速度 × 時間

```
int sp=300; // モーターの速度
```

```
void fwd(int tau){  
    int offset_l=0, offset_r=0;  
    motor(sp+offset_l, sp+offset_r, HIGH, HIGH);  
    delay(tau);  
}
```

```
void rotL(int tau){  
    int offset_l=0, offset_r=0;  
    motor(sp+offset_l, sp+offset_r, LOW, HIGH);  
    delay(tau);  
}
```

# モーターを回す：回転方向・回転速度



整数：HIGH または LOW  
回転方向

```
digitalWrite(MOTOR_L_CWCCW, HIGH);  
ledcWrite(CH_0, sp);
```

ピン番号  
整数：0~1023  
モーター速度(100 <= sp <= 450)

ピン番号

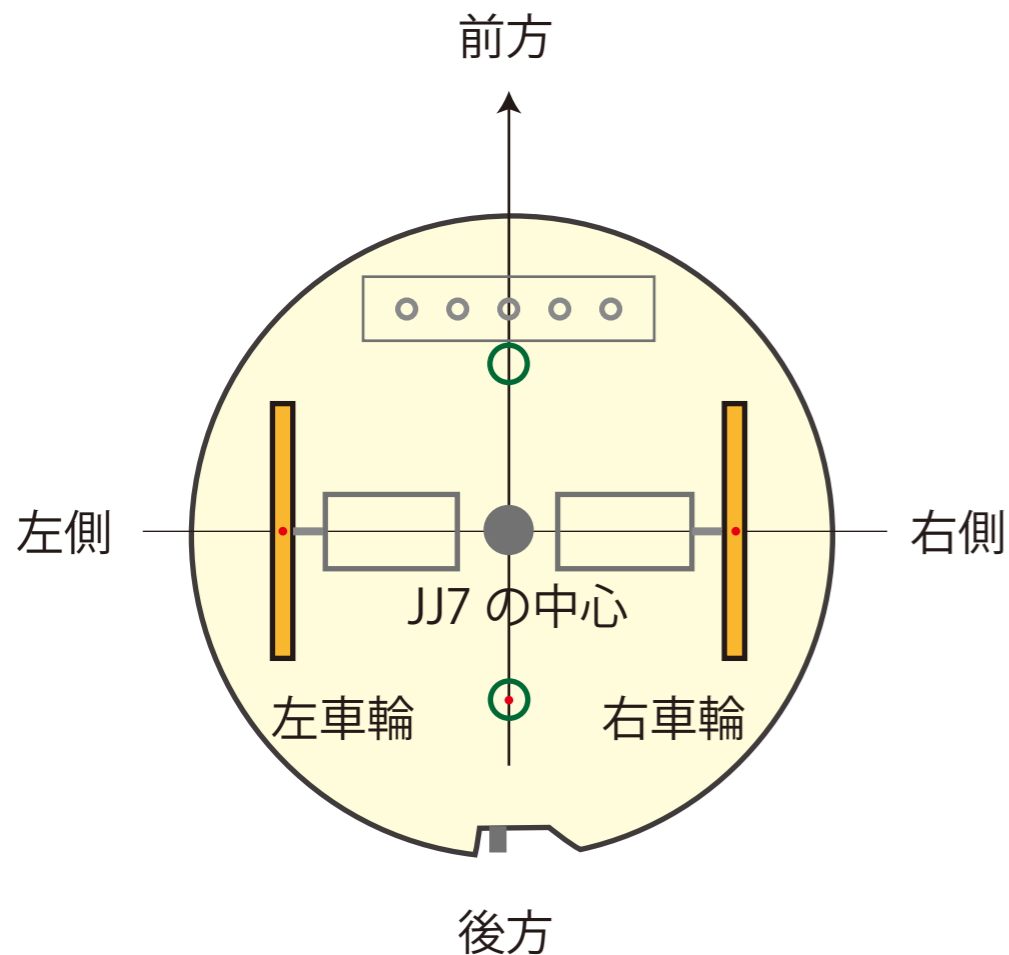
# JJ7のピン配置

辻先生の配布資料より引用

ポート	接続先	型番	ライブラリ	注釈
2	青色 LED(ボード上)	表面実装	pinMode, digitalWrite	
*26	モーターPWM(左)	DRV8835	ledcSetup, ledcAttachPin, ledcWrite	CH_0, 10bit
*4	モーターPWM(右)	DRV8835	ledcSetup, ledcAttachPin, ledcWrite	CH_1, 10bit
*16	モーター方向(左)	DRV8835	pinMode, digitalWrite	
*17	モーター方向(右)	DRV8835	pinMode, digitalWrite	
12	タッチセンサ	T5	touchAttachInterrupt	
13	サーボモーター	SG90	ledcSetup, ledcAttachPin, ledcWrite, ESP32_Servo	CH_2, 5.0V
14	タッチセンサ	T6	touchAttachInterrupt	
15	圧電スピーカー	Piezo	ledcSetup, ledcAttachPin, ledcWriteTone ledcWriteNote	CH_3
18	緑色 LED	OST1313G	pinMode, digitalWrite	
*19	リング LED	WS2812B	FastLED	5.0V
22 (SCL)	地磁気センサ	QMC5883L	FRobot_QMC5883.	I <sup>2</sup> C
21 (SDA)	温度・湿度センサ	HTU21D	SparkFunHTU21D	
	照度センサ	BH1750	Wire	
25	スイッチ	タクト	pinMode, digitalRead	
27	温度センサ(防水)	ds18b20	DallasTemperature	1 wire
*A5 (33)	フォトインタラプタ	LBR127HLD	analogRead	3.3V ADC1 12bit
*A4 (32)	フォトインタラプタ	LBR127HLD	analogRead	3.3V ADC1 12bit
*A7 (35)	フォトインタラプタ	LBR127HLD	analogRead	3.3V ADC1 12bit
*A6 (34)	フォトインタラプタ	LBR127HLD	analogRead	3.3V ADC1 12bit
*A3 (39)	フォトインタラプタ	LBR127HLD	analogRead	3.3V ADC1 12bit
A0 (36)	距離センサ	GP2Y0A21YK	analogRead	5.0V ADC1 12bit



# JJ7 の操作：車輪の回転方向と回転速度



pin番号	回転方向	モーター速度
左車輪	16 HIGH/LOW	26 0 ~ 1023
右車輪	17 HIGH/LOW	4 0 ~ 1023

digitalWrite      ledcWrite

共通操作：持続時間 tau

## JJ7の運動

2つの車輪の速度を操作してロボットを動かす

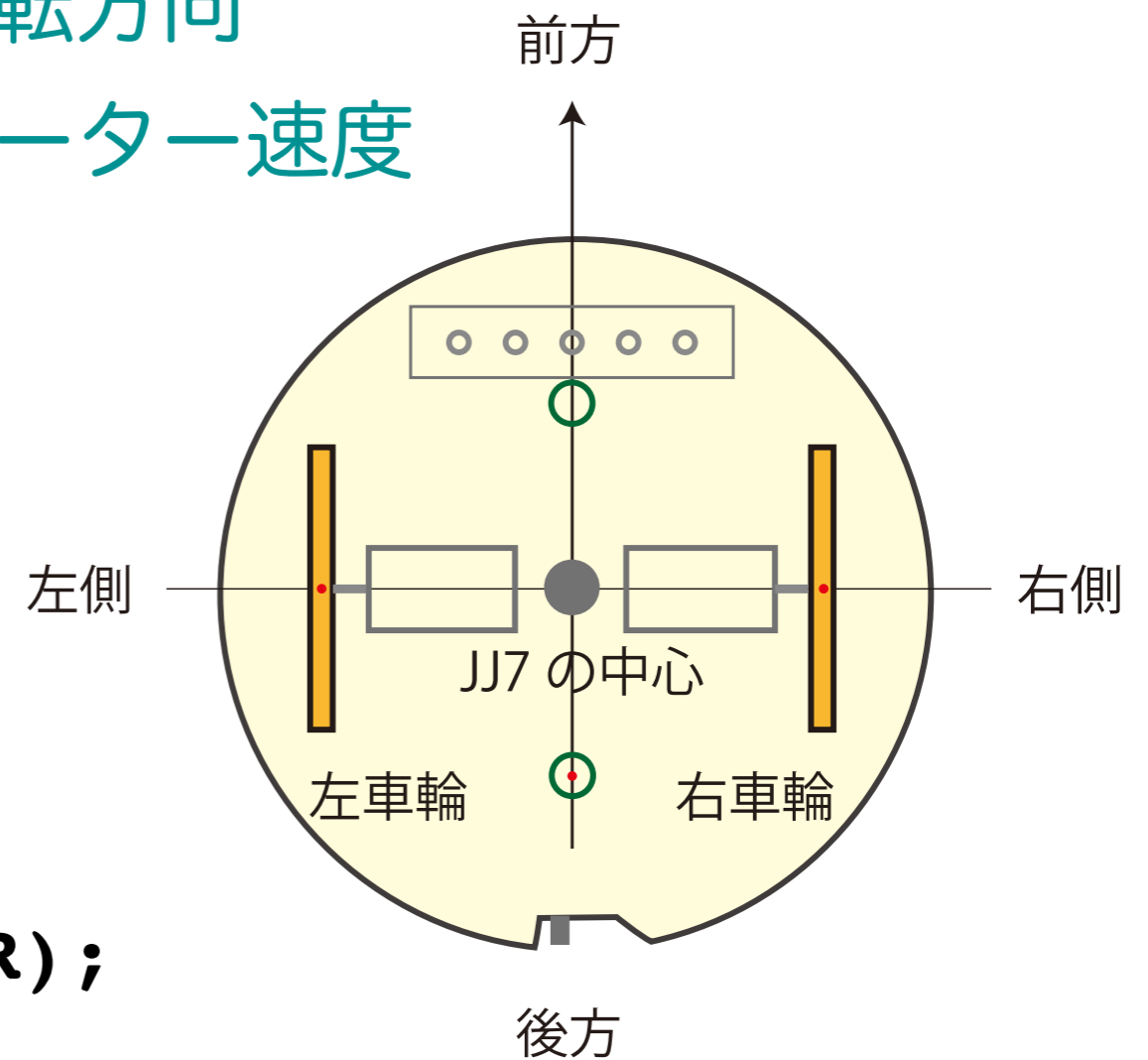
操作量①：左右にある車輪の回転方向

操作量②：左右にある車輪のモーター速度

操作量③：操作を加える時間

距離 = 線速度 × 時間

線速度 = 半径 × 角速度

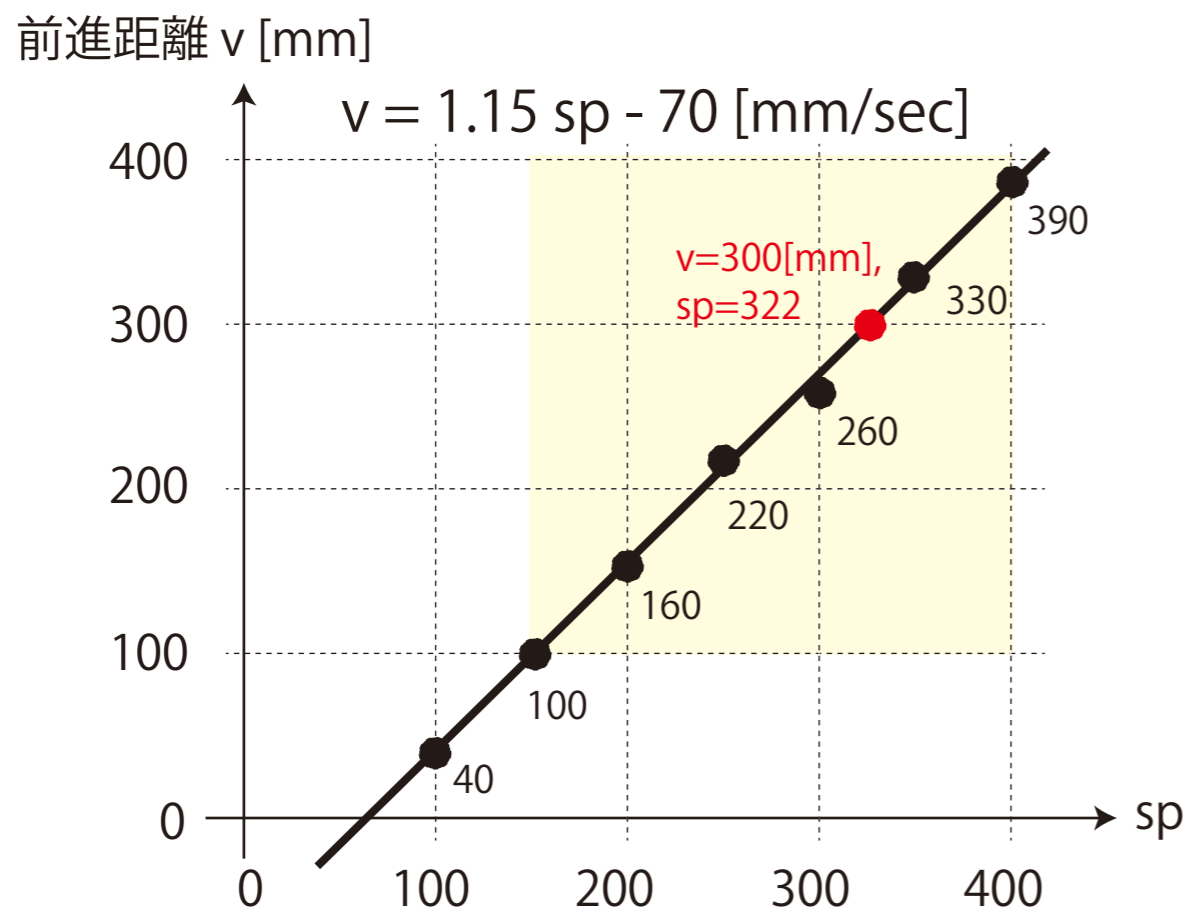


```
motor(spL, spR, oriL, oriR);
delay(tau);
```

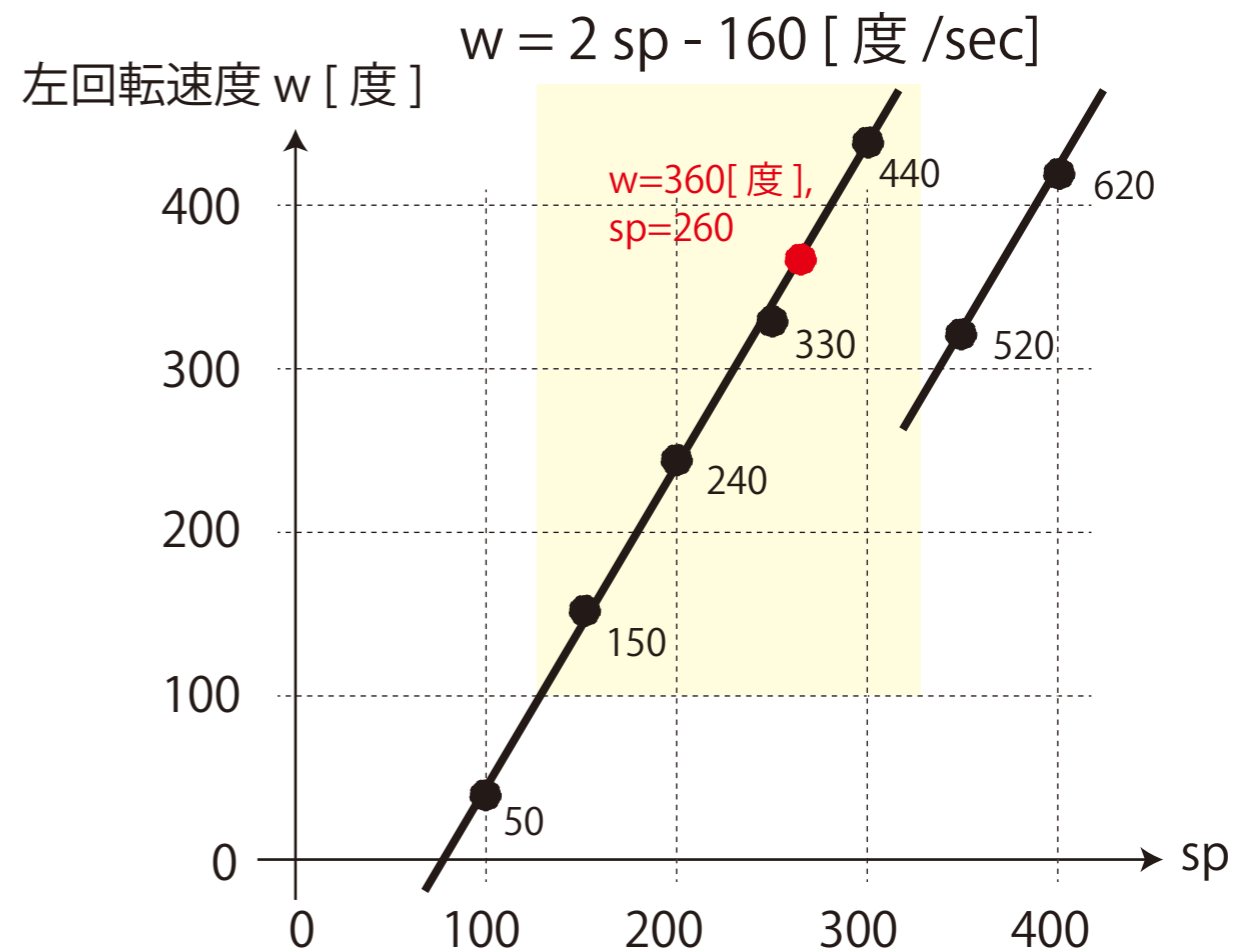
# Ex302を実行する：前進・回転運動の実測例

Ex302：Ex301のスケッチをファイルを分割する

## 1秒間の直進の実測例

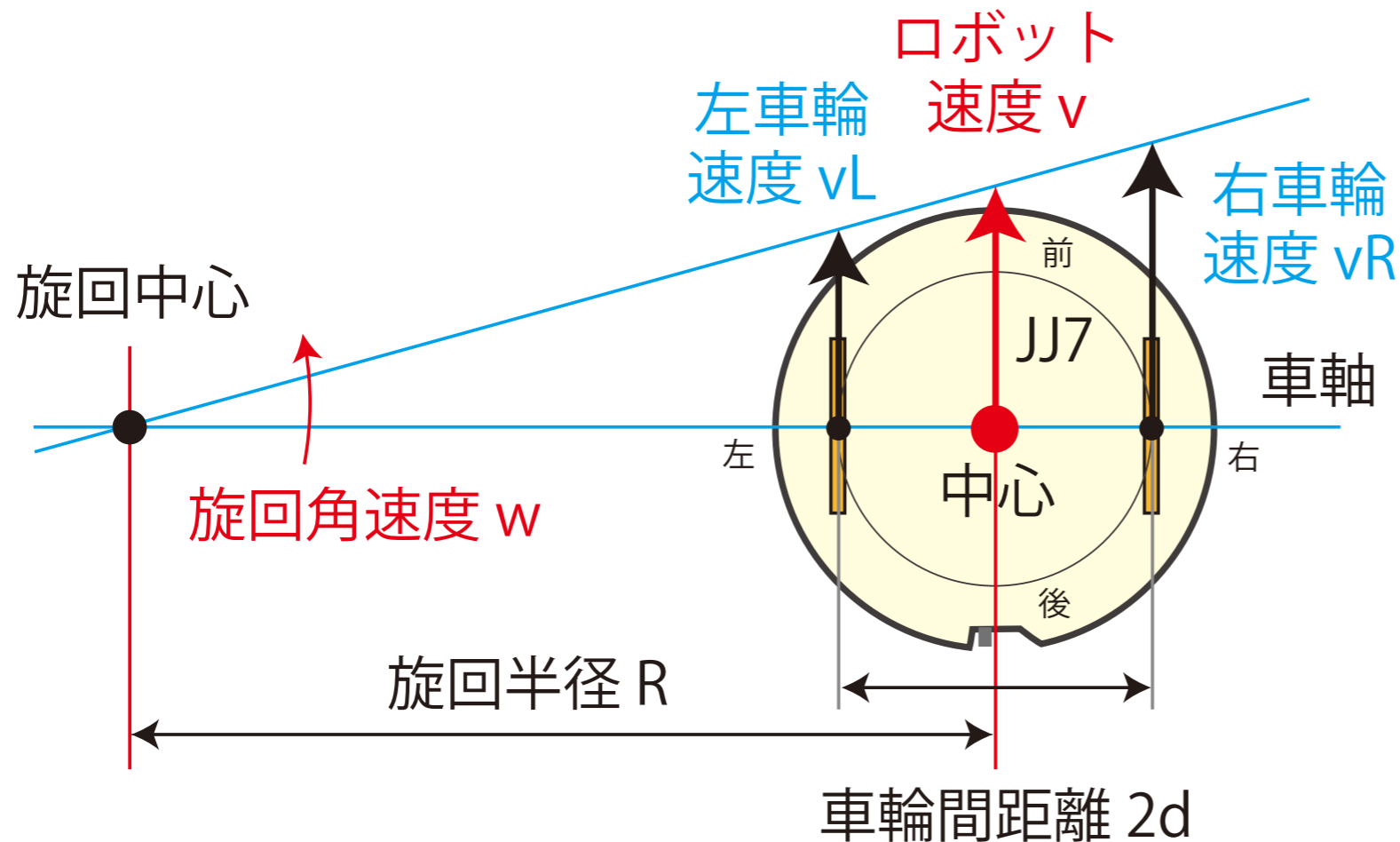


## 1秒間の左回転実測例



## JJ7 ロボットの運動

- JJ7の運動：車輪速度( $v_R$ ,  $v_L$ )と中心速度( $v$ ,  $\omega$ )



$$(R + d)\omega = v_R$$

$$(R - d)\omega = v_L$$

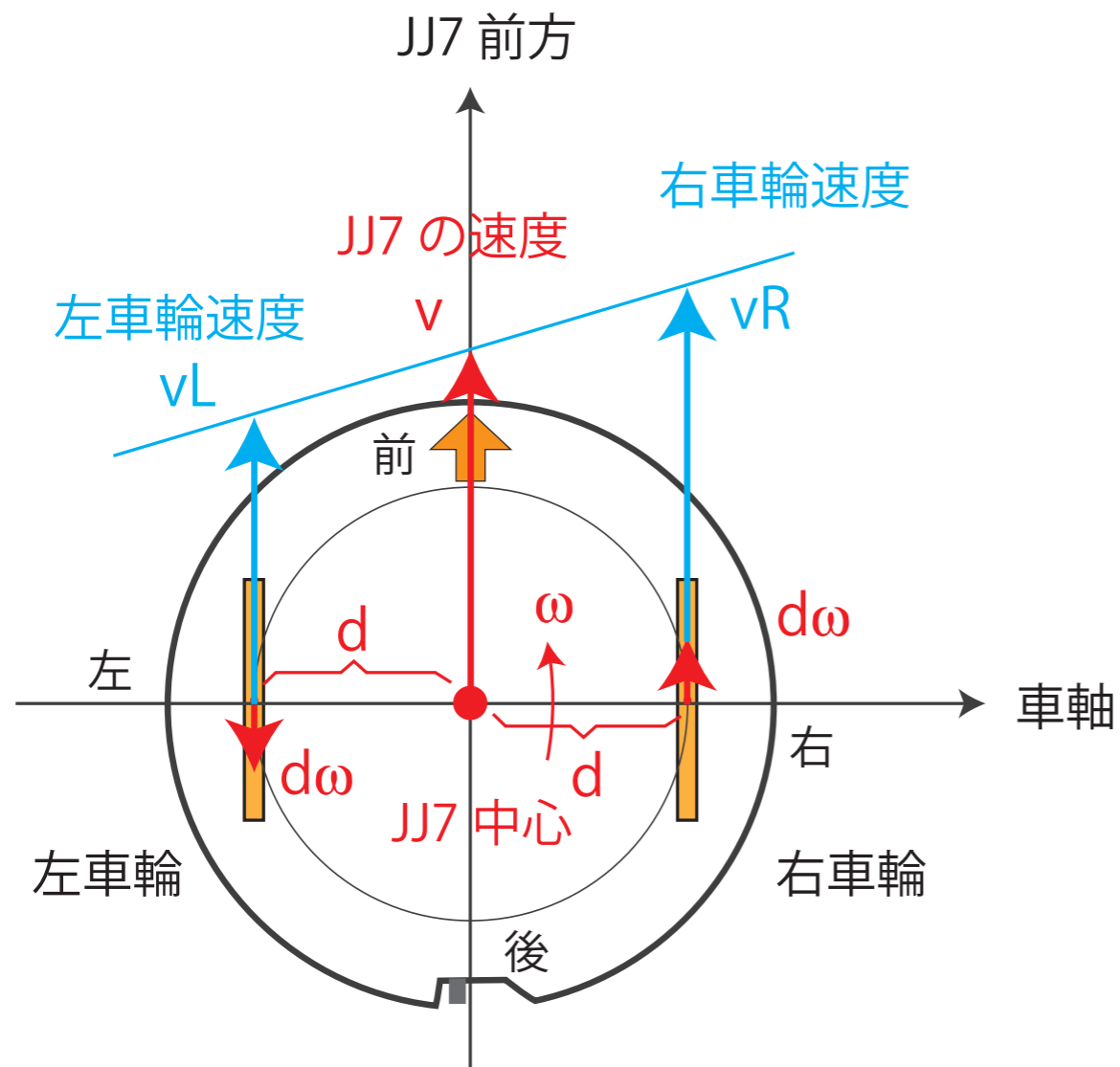
$$R\omega = v$$

$$v_R = v + \omega d$$

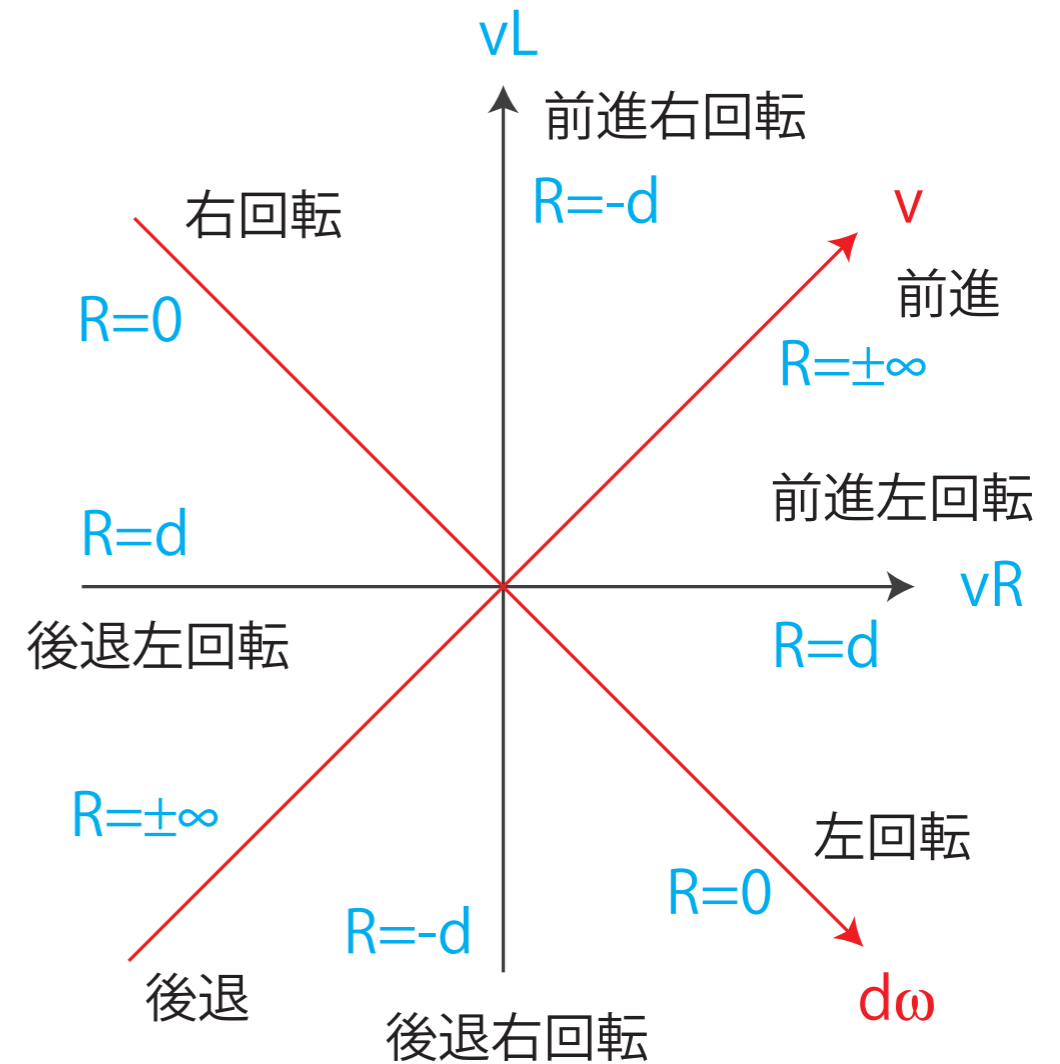
$$v_L = v - \omega d$$

$$v = \frac{1}{2}(v_R + v_L), \quad \omega d = \frac{1}{2}(v_R - v_L), \quad R = \frac{d(v_R + v_L)}{(v_R - v_L)}$$

## 速度平面(vR, vL)と速度平面(v, wd)



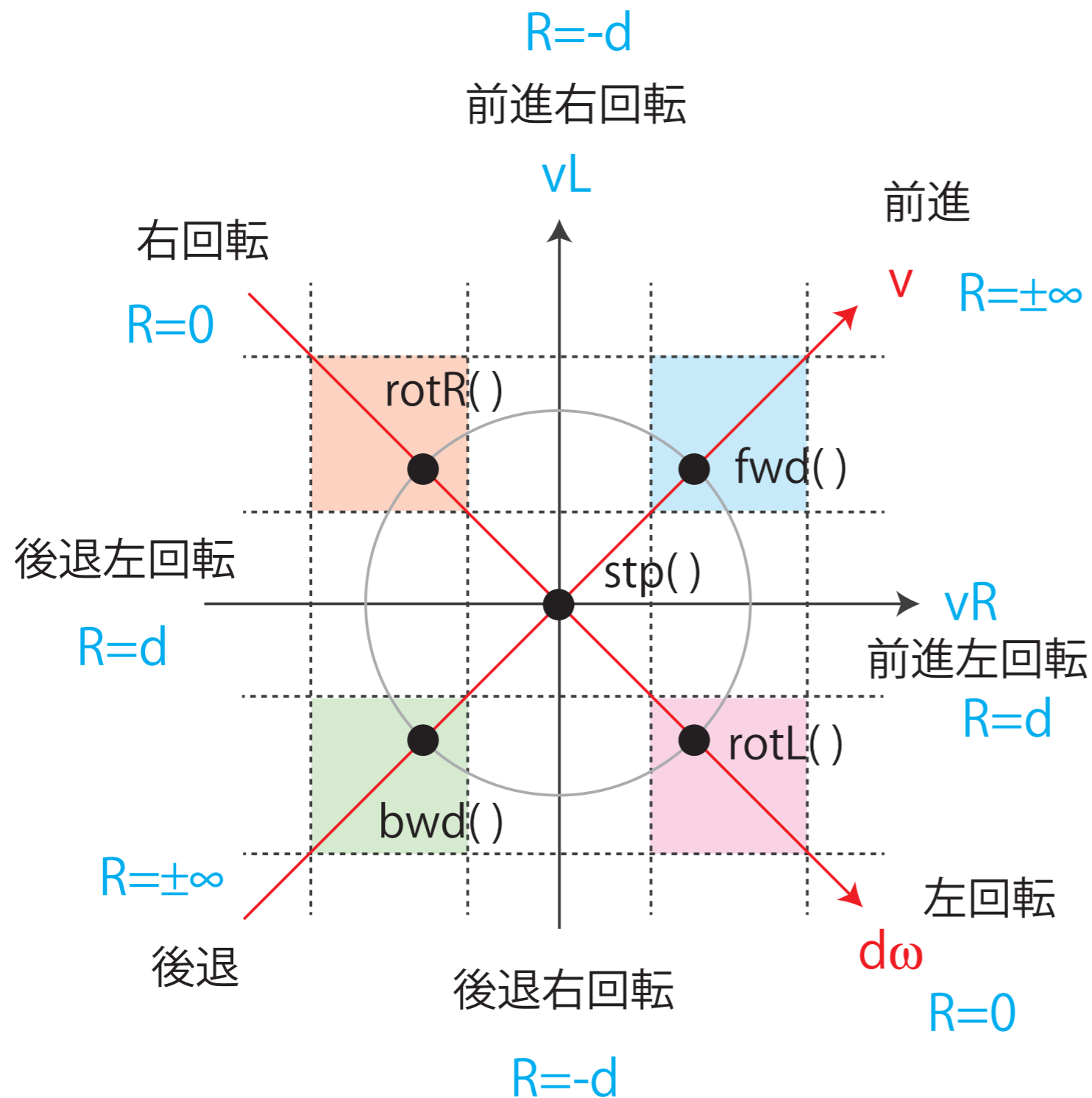
(a)



(b)

$$v = \frac{1}{2}(v_R + v_L), \quad \omega d = \frac{1}{2}(v_R - v_L), \quad R = \frac{d(v_R + v_L)}{(v_R - v_L)}$$

# 速度平面の使用可能領域



$$v_R = v + \omega d$$

$$v_L = v - \omega d$$

$$v = \frac{1}{2}(v_R + v_L)$$

$$\omega d = \frac{1}{2}(v_R - v_L)$$

# Ex303を実行する：リングLEDを付加する

---

## Ex303：リングLEDで前後・左右を表示する

```
#include "myFastLed.h"
```

```
void setup() { // 定数と初期設定をヘッダーファイルにした  
  myFastLedSetup();  
}
```

白：前方, 赤：後方, 左：緑, 右：青

```
//leds[12]:white:forward, leds[4]:red: backward  
// leds[8]:green:left, leds[0]:blue:right
```

```
void loop() { // 白・青と赤・緑の交互点滅：blink スケッチに同じ  
  leds[8] = CRGB(0, 0, 0); leds[0] = CRGB(0, 0, 0);  
  leds[4] = CRGB(50, 0, 0); leds[12] = CRGB(50, 50, 50);  
  FastLED.show(); delay(1000);  
  leds[4] = CRGB(0, 0, 0); leds[12] = CRGB(0, 0, 0);  
  leds[8] = CRGB(0, 50, 0); leds[0] = CRGB(0, 0, 150);  
  FastLED.show(); delay(1000);  
}
```

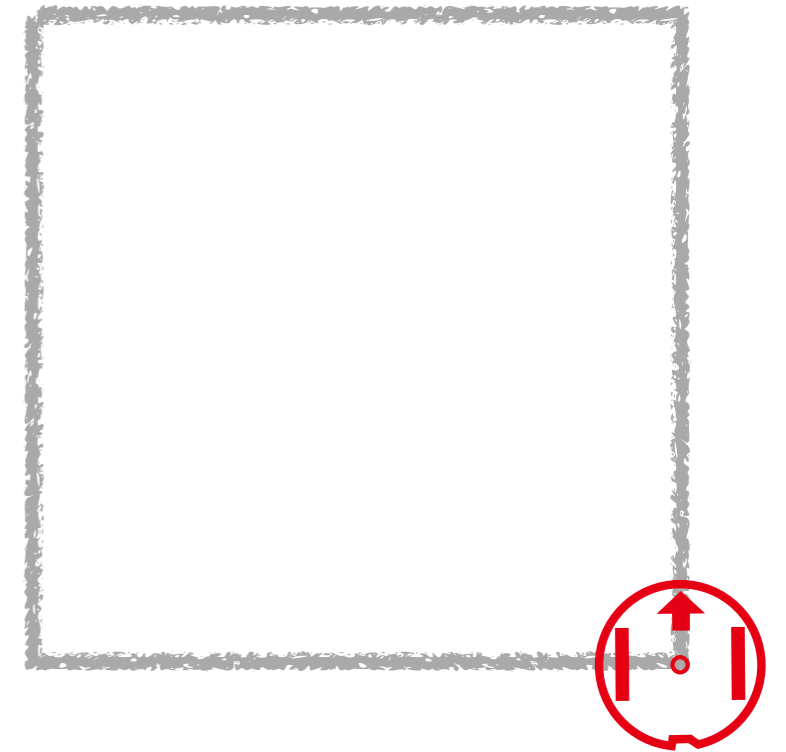
# Logo風 線図形を描く : turtle graphics

例1 : 「30cm進んで90度左回転する」を4回繰り返す

Ex304 : Ex302にリングLEDを付加する

```
void loop() {  
  while (digitalRead(SW_PIN) == HIGH);  
  delay(1000);    SWを押す毎に1回実行される  
  testSquare();  
}
```

```
void testSquare(){  
  for(int i=0; i<4;i++){  
    fwd(30);      「30cm進んで90度左回転する」を4回繰り返す  
    rotL(90);  
  }  
}
```



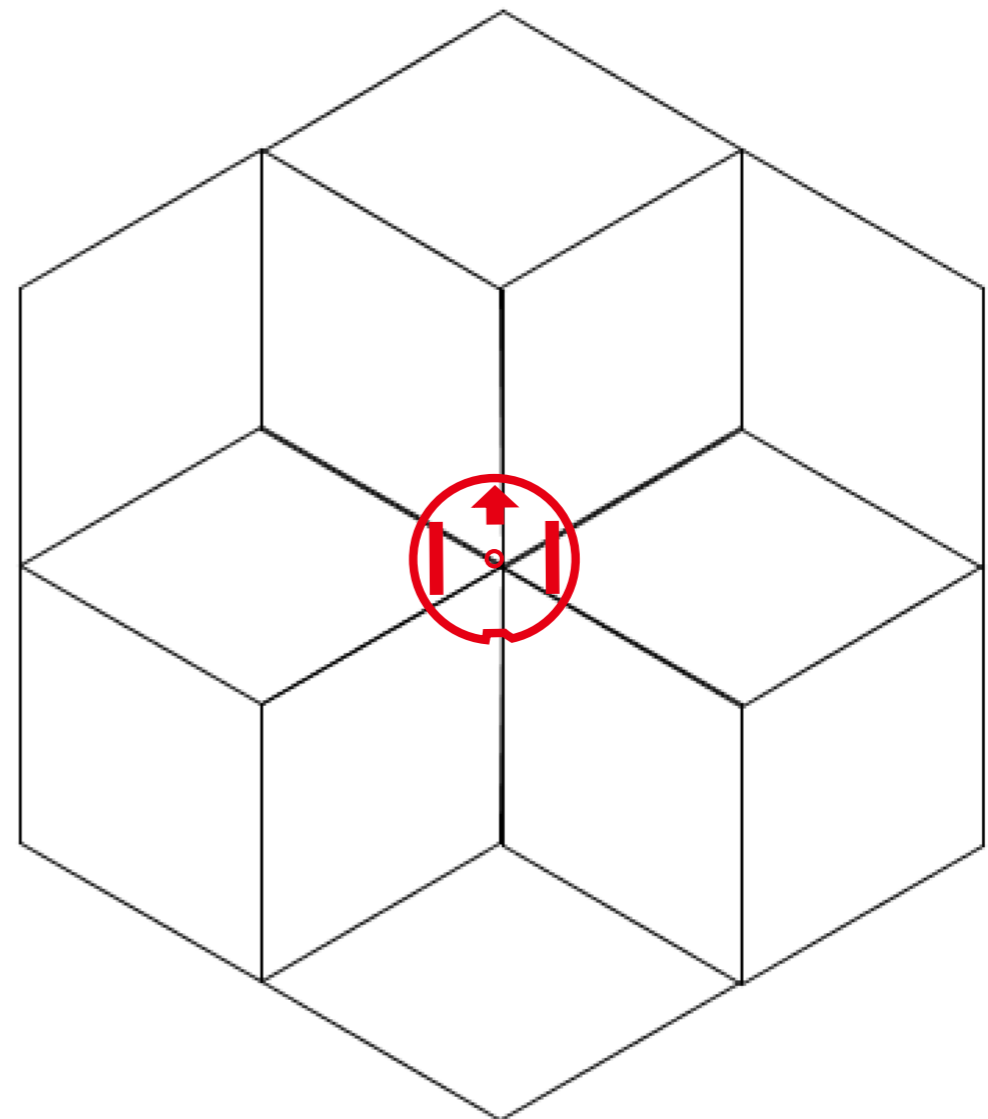
Ex304は、fwd(進行距離), rotL(回転角度) に仕様を変更しました



## turtle graphics の例 2

---

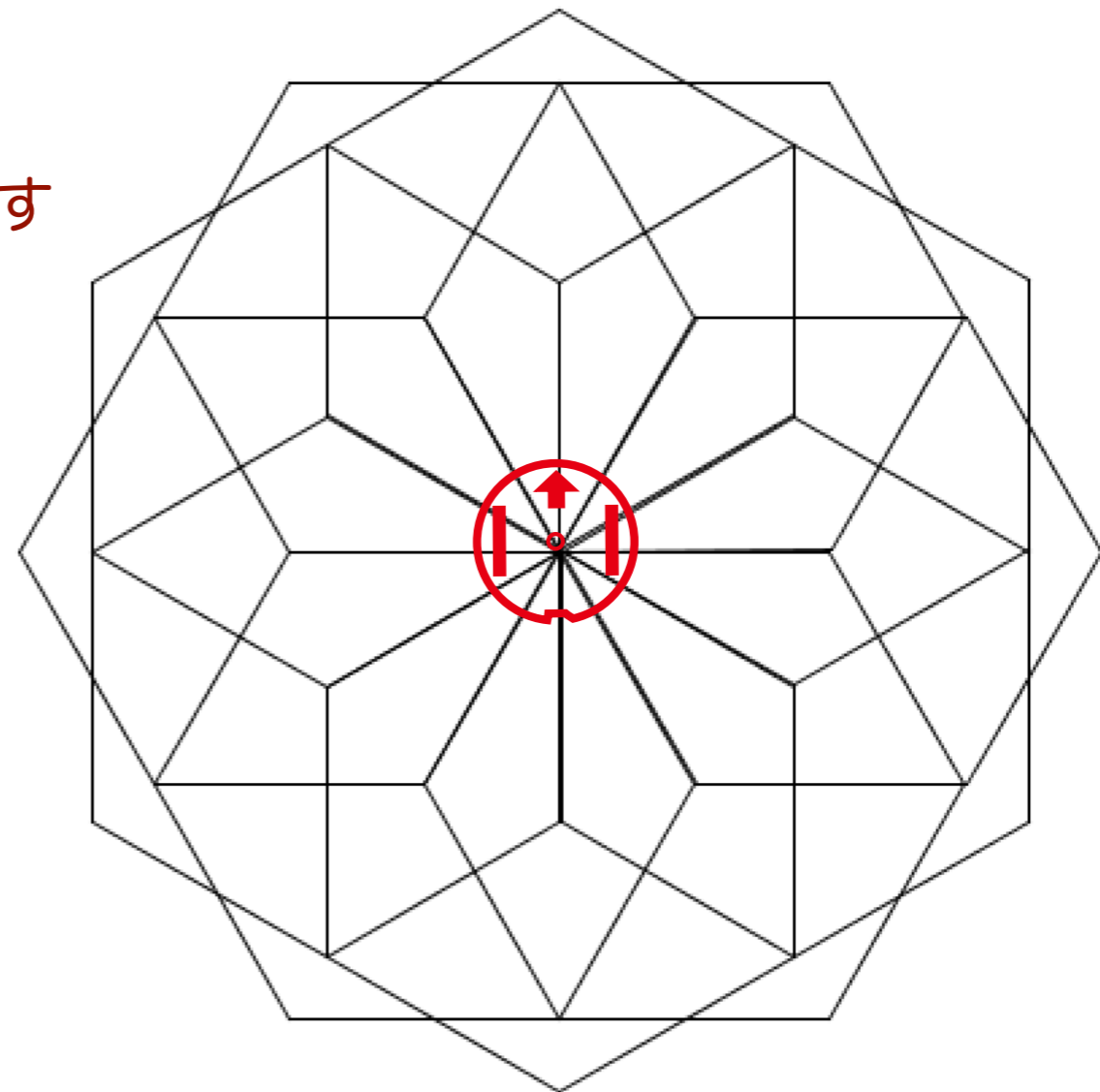
```
void turtle6060(){  
  for(int j=0; j<6; j++){  
    for(int i=0; i<6; i++){  
      fwd(200); 「20cm進んで60度左回転する」を6回繰り返す  
      rotL(60);  
    }  
    rotL(60);  
  } 「60度左回転する」を6回繰り返す  
}
```



## turtle graphics の例 3

---

```
void turtle6030(){  
  for(int j=0; j<12; j++){  
    for(int i=0; i<6; i++){  
      fwd(200); 「20cm進んで60度左回転する」を6回繰り返す  
      rotL(60);  
    }  
    rotL(30);  
  } 「30度左回転する」を12回繰り返す  
}
```



# 力学(Mechanics)と運動学(Kinematics)

分野	与える量	法則	求める量
力学	$m, F$	$m \frac{d^2 x}{dt^2} = F$	$x(t), v = \frac{dx}{dt}$
運動学	$v, \omega$	$\frac{dx}{dt} = v, \frac{d\theta}{dt} = \omega$	$x(t), \theta(t)$

$$\dot{v} = \frac{F}{m} \simeq 0 \quad \text{運動学で力学が近似できる}$$

## 来週の課題は？

---

速度が時間的に変化する場合のJJ7の運動を考えます

目 標

JJ7を踊らせたい