

## e01 G 命令の使い方

ー モーター2ケの動作とその時間を指定する

G①①①②②② : ③③

- ① : 左モーターパワー + で正回転・逆回転
- ② : 右モーターパワー
- ③ : 動作時間 (Wと同じ)

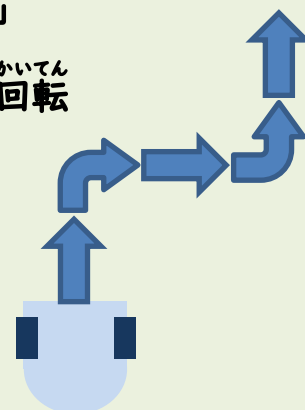
### 01 G命令

```
01G+30+30:10
02G+30-30:04
03G+30+30:10
04G-30+30:04
05G+30+30:10
```

### -クランク前進プログラム

左右同じスピードは「前進」

左が前進、右が後退で右回転



### 練習問題

1. コースを作って走らせてみよう
  - 動作時間とモーターパワーを色々変えて試してみよう

## e02 M命令の使い方

— モーター2ケの動作

M①①①②②②

①：左モーターパワー 十で正回転・逆回転

②：右モーターパワー

— W命令 プログラムの動作ストップ

W：①①

①：停止時間 1/10秒単位 (例:15=1.5秒)

※ MとWの組合せで G命令 と同じ動きになる。単純に動かすときはG命令の方が簡単になる

### 02 M命令

01M+30+30

02W:10

03M+30-30

04W:04

05M+30+30

06W:10

07M-30+30

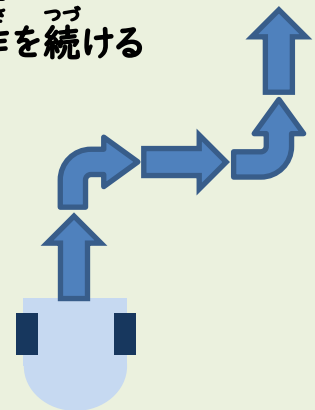
08W:04

09M+30+30

10W:10

### -クランク前進プログラム

左右同じスピードは「前進」  
W:10で、1秒間上の動作を続ける



### 練習問題

1. e01と同じように、コースを作って走らせてみよう  
- 動作時間とモーターパワーを色々変えて試してみよう

## e03 L命令の使い方

— LEDの点灯消灯、LCDへの数字表示を行う。

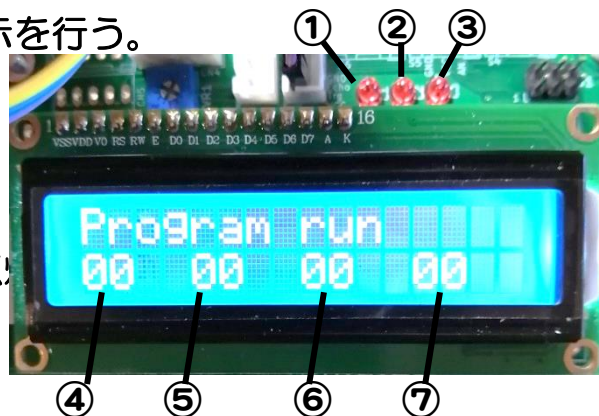
L①：②②

①：1～3はLED、

4～7はLCD左から4ヶ所

②：1-3 00で消灯、それ以外で点灯

4-7 2桁数字を表示



数字のみでなく変数も可(S・C・X・T・I)

### 03 L命令

01 L1 : 01

02 W : 10

03 L1 : 00

04 L2 : 01

05 W : 10

06 L2 : 00

### -LEDの連続点滅

L1を点灯

1秒待つ

L1を消して

L2を点灯

1秒待つ

L2を消す

### 練習問題

1. 点滅スピードを変えてみよう

- W命令の時間を変えて、変化を確認しよう

## e04 J命令（繰り返し動作）

- LEDの点灯消灯、LCDへの数字表示を行う。
  - JLS: ループの最初（無条件で繰り返し）
  - JLE: ループの終わり
    - JLQ: ループの最初(条件付き) 判定正の間繰り返す(式は?と同じ)
    - JLB: ループの中断 判定正の時ループから出る(式は?と同じ)

04 J命令	繰り返し-流れるLED
01 JLS :	繰り返しループの始まり
02 L1 : 01	L1を点灯
03 W : 10	1秒待つ
04 L1 : 00	L1を消して
05 L2 : 01	L2を点灯
06 W : 10	1秒待つ
07 L2 : 00	L2を消す
08 L3 : 01	L3を点灯
09 W : 10	1秒待つ
10 L3 : 00	L3を消す
11 JLE :	繰り返し ループ JLS:へ

### 練習問題

1. 点滅スピードを変えてみよう
  - W命令の時間を変えて、変化を確認しよう

## e05 判定命令 -フロアセンサー-

ー 判定命令（条件分岐）センサー等の値でプログラムを変える

書式：？S□○△□□      △は+かー

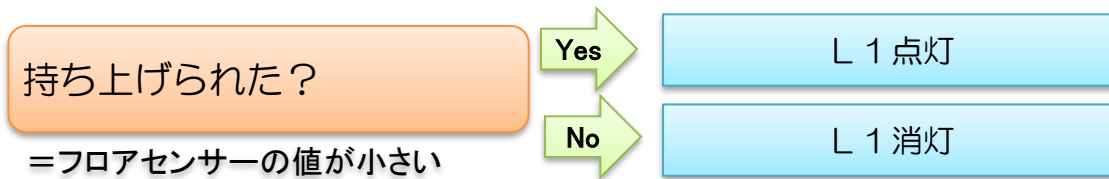
例： ?S1>+50 = 1番センサーが50%以上か？

①：センサー番号、変数、タイマー番号 (S・C・X・T)

②：判定記号 (<、>、=、!)

①：判定する値 数字のみでなく変数も可(S・C・X・T)J①：②②

判定が正しいときはこの行以下の[+]の行、正しくないときは[-]の行を実行



### 05 判定命令      ロボットが持ち上げられたら LED を点灯

01 JLS :	繰返しループの始まり
02 ?S2<+15	S2(フロアセンサー)が15以下なら
03 +L1 : 01	<span style="background-color: #ADD8E6;">はい</span> ハイなら L1点灯
04 -L1 : 00	<span style="background-color: #FF6347;">いいえ</span> イイエなら L1消灯
05 JLE :	繰返しループ

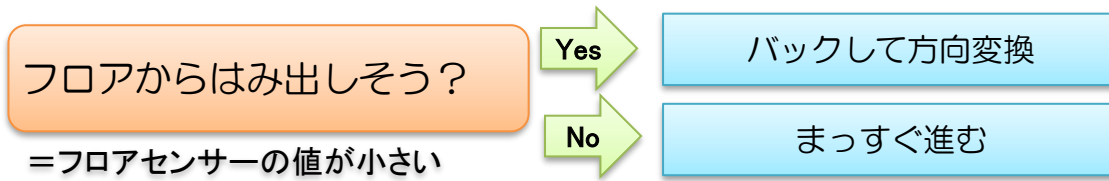
#### 練習問題

ロボットがテーブルの上に置かれているときは左に回転。持ち上げられるとモーターが停止するプログラムを作ってみよう

ヒント- 上のプログラムの03,04行にモーターの動作を書く

## e06 判定命令 -フロアセンサー2-

ー フロアセンサーではみだし防止プログラムを作ってみる



### 06 判定命令 はみだし防止

01 JLS : 繰返しループの始まり

02 ? S2 < + 20 S2(フロアセンサー)が 20 以下なら

03 + G - 30 - 30 : 10 はい バックする

04 + G - 30 + 30 : 10 はい 方向変換

05 - M + 30 + 30 いいえ まっすぐ進む

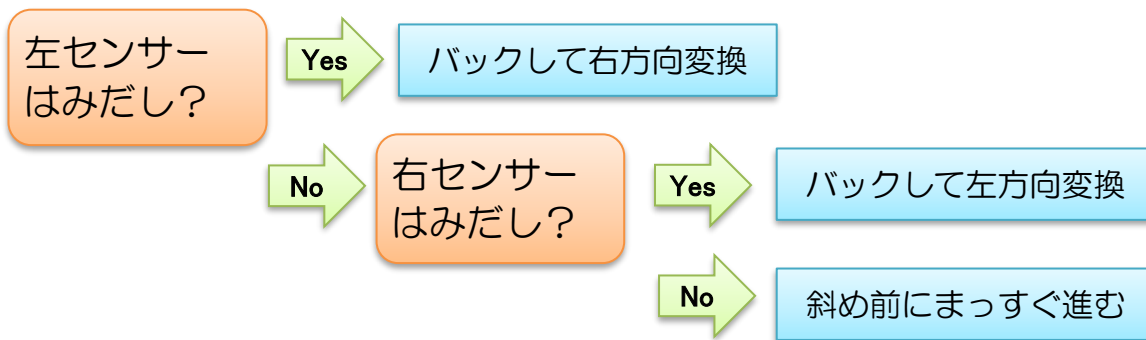
06 JLE : 繰返しループ

### 練習問題

S1 のセンサーも使って、それぞれ違う方向に回るようにしてみよう

## e07 判定命令 -フロアセンサー-3- 迷路探索 1

ー はみだし防止プログラムの応用で下のような迷路探索をさせてみる



### 07 判定命令 迷路探索

01 JLS :	繰返しループの始まり
02 ? S1 < +30	SI (左) が 30 以下なら
03 +G -20 -20 : 10	<span style="background-color: #ADD8E6;">はい</span> バックする
04 +G +30 -30 : 08	右方向変換
05 - ? S2 < +30	<span style="background-color: #FF6347;">いいえ</span> (左はみだしてない時) S2(右)確認
06 - +G -20 -20 : 10	<span style="background-color: #ADD8E6;">はい</span> (右はみだし) バック
07 - +G -30 +30 : 08	左方向変換
08 --M +30 +30	<span style="background-color: #FF6347;">いいえ</span> (両方はみだしなし) まっすぐ進む
09 JLE :	繰返しループ

## e08 判定命令 -超音波センサー-

### ー 超音波センサーのテスト

超音波センサ(S7)は、対象物との距離を cm で測定しています。

これを使って 20cm より近づいたら LED を点灯するプログラムを作ります

### 08 判定命令 超音波センサーのテスト

01 JLS :	繰返しループの始まり
02 ? S7 < + 20	S7(超音波センサ)が 20cm 以下?
03 + L1 : 01	<span style="background-color: #0070C0; color: white; padding: 2px;">はい</span> LED点灯
04 - L1 : 00	<span style="background-color: #FF0000; color: white; padding: 2px;">いいえ</span> LED消灯
05 L4 : S7	S7の値モニタ
06 JLE :	繰返しループ

### 練習問題

壁に近づいたら方向を変えて進むプログラムを作ってみよう



## e09 判定命令 -タイマーの使い方-

### - タイマーのテスト

NRob では T0~T2 まで 3 ケのタイマーがあります。これらは 0.1 秒のカウントアップタイマーで、代入命令 (#T) で値をセット、判定命令 (?S) で判定をする形で使います。

例として、スタート 1 秒後に L1 を点灯、3 秒後に L2 を点灯、5 秒後に 2 つを消してタイマーをリセット (0 にする) するプログラムを作ってみます

### 09 判定命令 超音波センサーのテスト

プログラム	説明
01 JLS :	繰返しループの始まり
02 L4 : T1	タイマー T1 の値を L4 に表示
03 ? T1 > +10	T1 が 1.0 秒を超えたら
04 + L1 : 01	L1 を点灯
05 ? T1 > +30	T1 が 3.0 秒を超えたら
06 + L2 : 01	L2 を点灯
07 ? T1 > +50	T1 が 5.0 秒を超えたら
08 + L1 : 00	L1 を消灯
09 + L2 : 00	L2 を消灯
10 + # T1 = +00	T1 をリセット
11 JLE :	

### 練習問題

上のプログラムを書き換えて、5.0 秒で L3 を点灯  
8.0 秒で 3 つとも消して、タイマーリセットするようにしてみよう

## e 10 変数の使い方 -カウンターの場合-

### — タイマーのテスト

NRob では X1~X0 まで 10 変数があります。

変数はプログラムで使う一時記憶のようなもので、いろいろな使い方があります。ここでは、09のタイマーより早くカウントするカウンターとしての使い方の例を示します。

### 10 変数      カウンターの例

01 #X1=00+00	変数 X1 に初期値をセット
02 JLS :	ループの始まり
03 L4 : T1	T1 を L4 に表示
04 #X1=X1+01	変数 X1 をカウントアップ
05 L5 : X1	X1 を L5 に表示
06 ?T1>+10	T1 が 1.0 秒を超えたら
07 +L1 : 01	L1 を点灯
08 ?T1>+30	T1 が 3.0 秒を超えたら
09 +L2 : 01	L2 を点灯
10 ?T1>+50	T1 が 5.0 秒を超えたら
11 +L1 : 00	L1 を消灯
12 +L2 : 00	L2 を消灯
13 #T1=+00	T1 をリセット
14 JLE :	JLS から繰り返し

### 練習問題

X1 は 99 を超えると” —— ” 表示になります。99 を超えたら、また 0 に戻ってカウントアップするようにしてみよう。

## e 1 1 変数の使い方 -その2-

— SW を読み込んでカウンターを止める

NRob では S9 で SW を調べることができます。

カウンターを SW で止めるプログラムを作ってみます。

S9 の値が (1=Set 2=Down 3=Up 4=Right 5=Left) で SW を確認できます。

### 1 1 変数      カウンターストップ

01 # X1 = 00 + 00	変数 X1 に初期値 0 をセット
02 # X2 = 01 + 00	変数 X2 に初期値 1 をセット(カウント動作フラグ)
03 JLS :	ループの始まり
04 L4 : X1	変数 X1 を L4 に表示
05 ? X2 = + 01	X2 が 1 なら
06 + # X1 = X1 + 01	X1 をカウントアップ
07 + ? X1 > + 99	X1 が 99 を超えたら
08 ++ # X1 = 00 + 00	X1 をリセット
09 ? S9 = + 02	S9 が 2 (下 SW が押された)
10 + ? X2 = + 01	X2 が 1 の時は(カウント中)
11 ++ # X2 = 00 + 00	X2 を 0 に(カウント停止)
12 +- # X2 = 01 + 00	違うときは X2 を 1 に(カウント開始)
13 JLE :	JLS から繰り返し

### 練習問題

X1 が '77' になったら L1 を点灯させるプログラムを追加しよう。

また、カウンターを1つつアップさせると難しいため '11' ずつカウントするように変更しよう