

—— ロボットを動かしてみよう ——

1. 準備
2. プログラムの基本
3. プログラムの応用
4. まとめ

ミュウロボエディタ画面の説明

- プログラムを作る・送る
- 計測や制御(手動操作)
- そのほか(プログラム操作など)

The screenshot shows the MYUUSB.EXE V1.0 (2011/6/26) application window. It is divided into several sections:

- 編集ボックス (Edit Box):** A large empty area for editing programs, with a yellow box highlighting it and the text "プログラムを作る領域" (Area for creating programs).
- 命令ボックス (Command Box):** A list of commands including DEFINE, パワーオンスタート, 前進, 後退, 右回り, 左回り, 停止, 左前, 左後, 右前.
- 解説ボックス (Explanation Box):** A text area for explanations.
- 計測/R実行 (Measurement/R Execution) Panel:** A detailed control panel with the following sections:
 - レジスタ値(USBSTAT):** Displays system information like A= 112(70), AH= 0(00), AX= 112, ARG1= 0(00), ARG2= 0(00), ARG= 0, IX= 0(00), IXP= 0(00), IXM= 237(ED), 実行Address: FB32, 命令Code: 226(E2), 命令: USBSTAT, FLAG= 4(04), BLKP= 176(B0), STAKP= 255(FF), and a hex dump.
 - 基板DATA:** A hex dump of board data with a "ステップイン" (Step In) button.
 - 出力ポート (Output Port):** Includes checkboxes for RC7-RC0 and a "PORTC = 8(08)" field.
 - 入力ポート (Input Port):** Includes checkboxes for In2-In4, "デジタル計測" (Digital Measurement), "AN10計測" (AN10 Measurement), "PORTB = 112(70)", and "SERVO-1" settings.
 - Buttons:** Includes "0.1秒ごと実行" (Execute every 0.1 seconds), "R実行" (Execute), and "R逐次実行" (Execute sequentially).
- Remote Control Panel:** Includes buttons for "開く" (Open), "保存" (Save), "転送" (Transfer), "クリア" (Clear), and "前進" (Forward). A "リモート 前進" (Remote Forward) dropdown menu is also present.

Annotations include:

- A thought bubble pointing to the "計測/R実行" button with the text: "ミュウロボがつながっていると USB-O → USB-● になる。" (When Myurobo is connected, it becomes USB-O → USB-●).
- Red arrows pointing from the "計測/R実行" button to the "リモート 前進" dropdown menu.

ミュウロボでプログラム 実行させる手順

①ミュウロボへ転送



②ケーブルをはずす



③リセットスイッチを押す



④プログラムを保存したり
消したりする

PCに保存して
あるプログラ
ムを呼び出す

USB-○

開く

プログラムを
ミュウロボに
送る

転送

ミュウロボで
計測する

計測/R実行

リモート操作
する

リモート

前進

読み込

ミュウロボ上
のプログラ
ムを呼び出す

保存

プログラ
ムを
PCに保存する

クリア

編集画面
の消去

編集ボックスクリア

編集ボックスをクリアしますか？

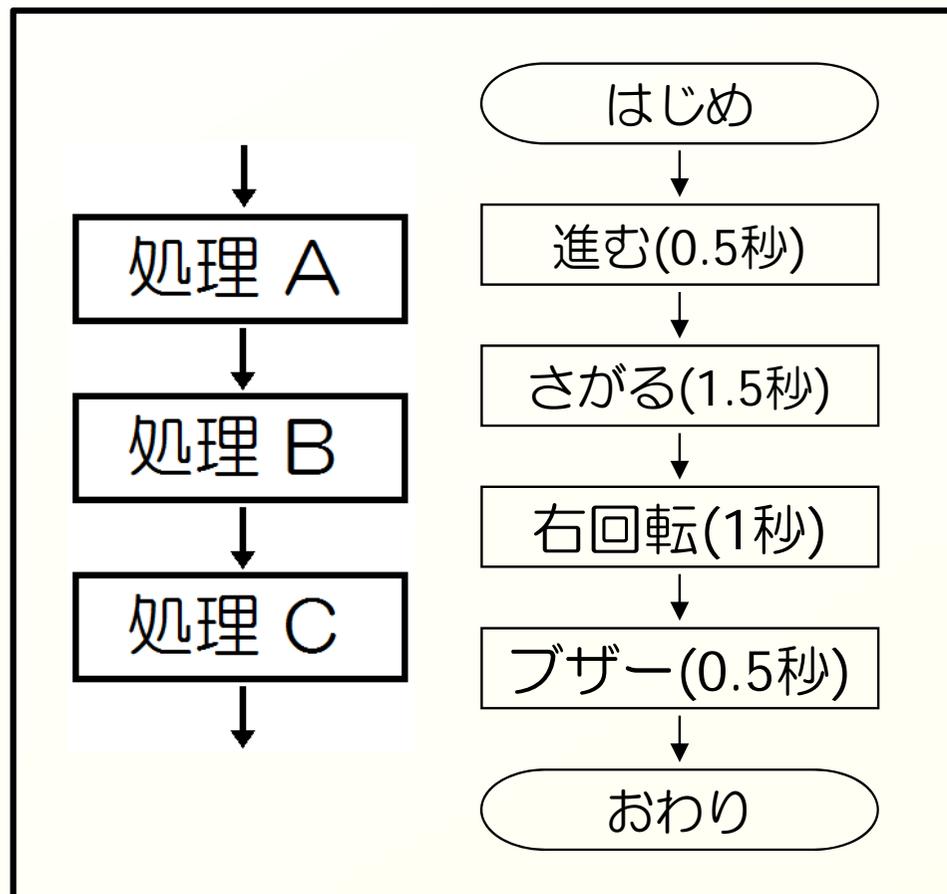
はい(Y)

いいえ(N)

2. プログラムの基本

【順次構造】

動作に必要な「命令の言葉」を順番に上からならべる



命令の言葉, XX (, XX)

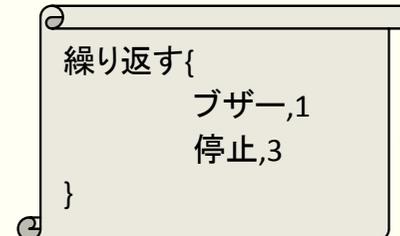
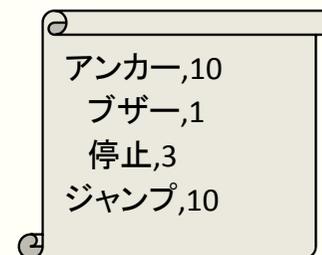
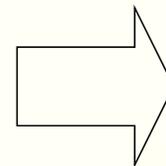
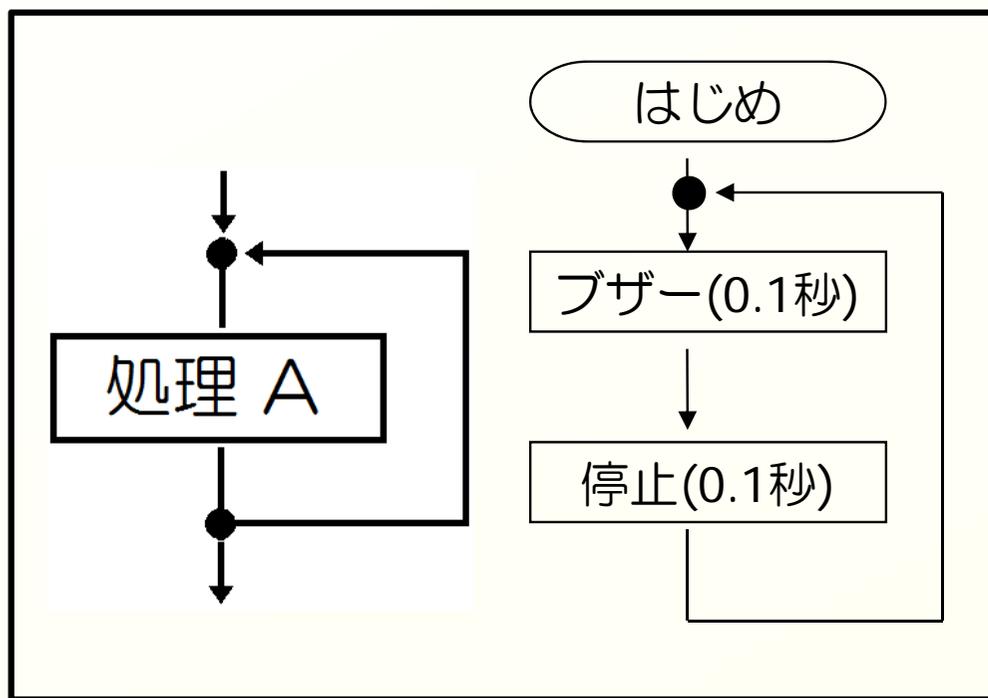
動作をつづける時間

編集ボックス

```
前進, 5  
後退, 15  
右回り, 10  
ブザー, 5
```

※数字は0から255まで

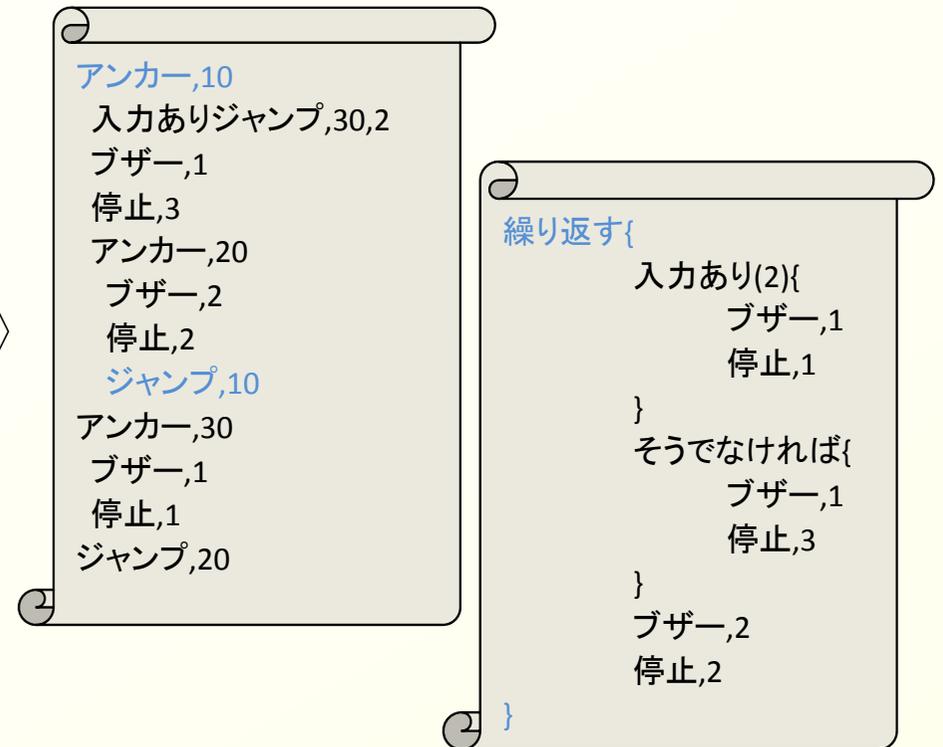
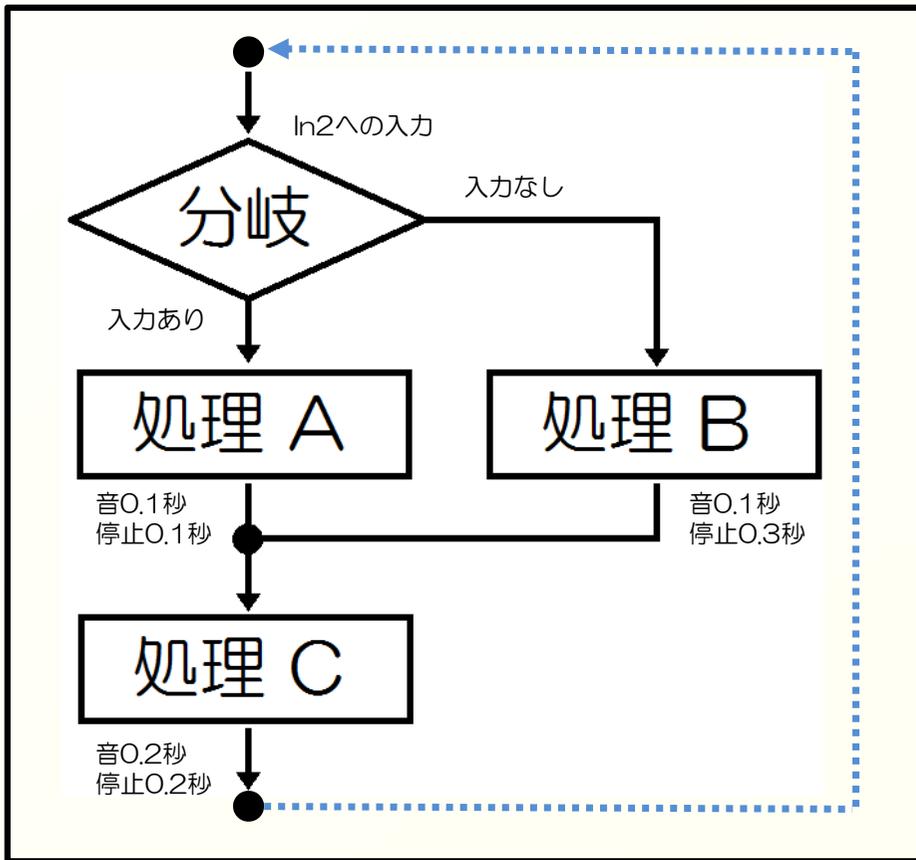
【繰り返し構造】 (ある条件の下で) 同じ処理を繰り返す



※同じ動作がずっと繰り返される

【選択構造】

2つ以上用意された処理の中から、
ある条件によって実行する処理を選ぶ

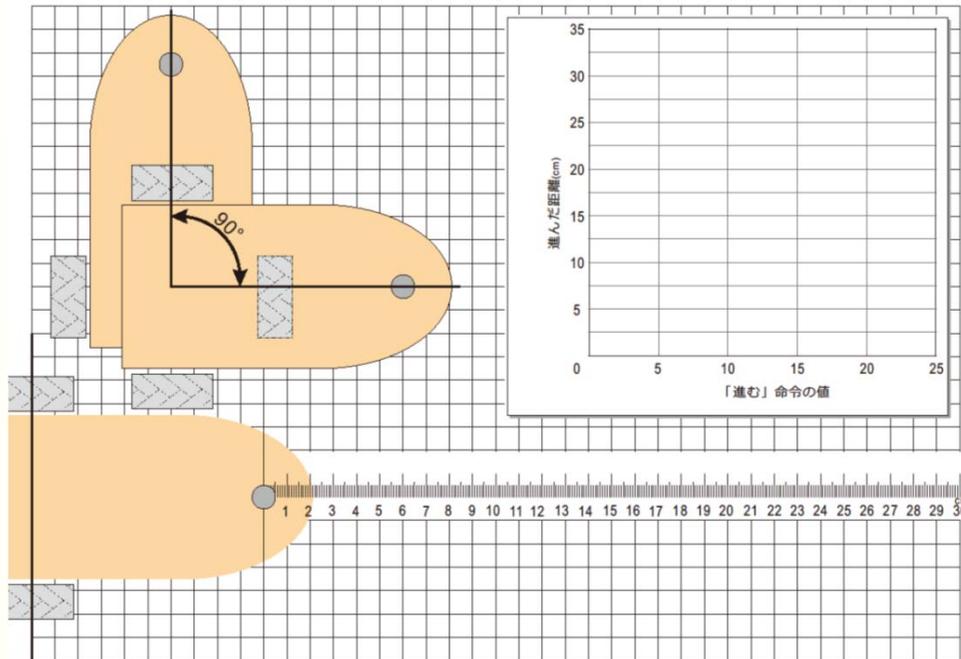


※信号の有無によって違った動作

3. プログラムの応用

【コース走行】

プログラムの命令は時間を指定（動作の大きさではない）
一定の距離を進めさせる，角度を回転させるためには？



① ミュウロボを
28cm進める
時間XX

編集ボックス

前進, XX

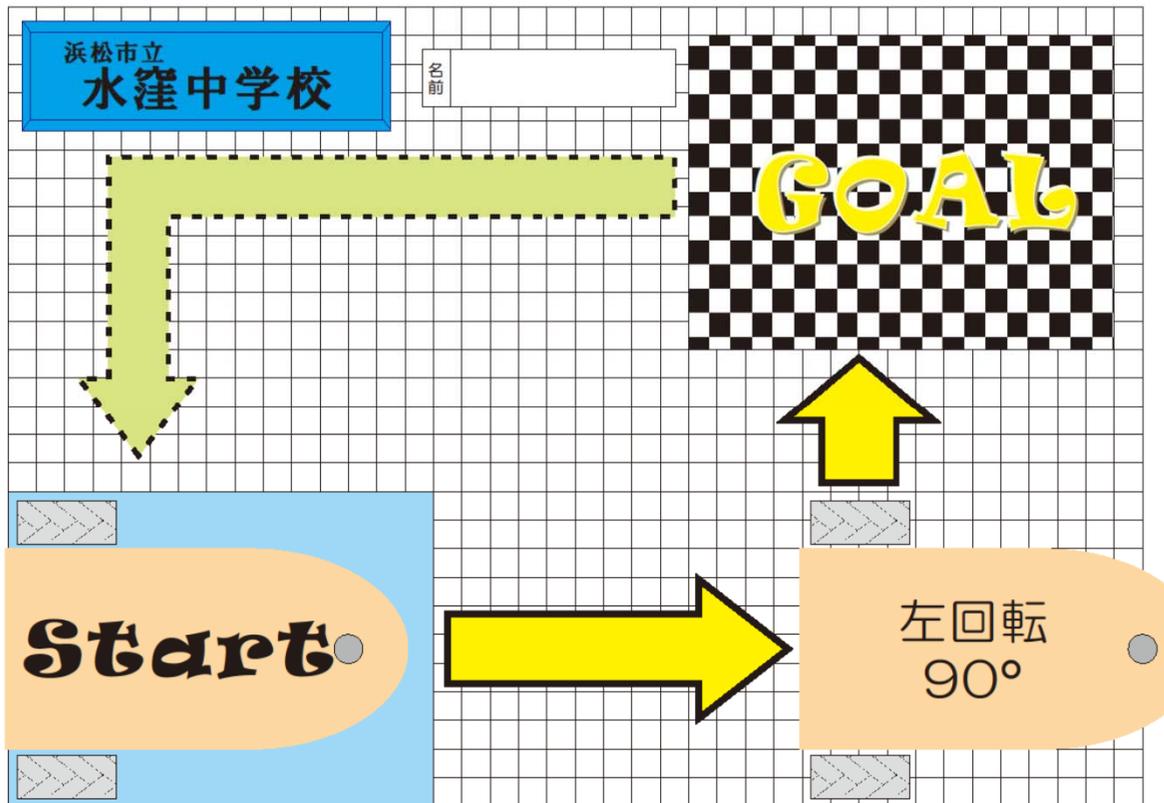
② ミュウロボを
90° 左回転さ
せる時間XX

編集ボックス

左回り, XX

※電池の状態，左右のタイヤ，ギヤボック
スの状態によって時間は変化します！

コース



ミュウロボの部品の性質をよく知り、自分の考えた通りに動くように工夫する

ミュウロボのおかしな動きの原因は、プログラムミスがほとんどである

他のロボット、身の回りの製品のうごきについても考えてみる

【音を鳴らそう】

表 「電子音」命令の音程データ

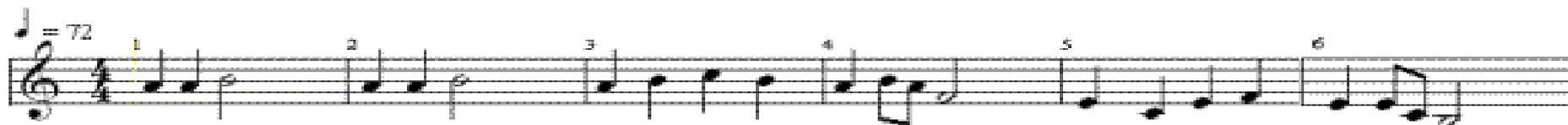
音程	C	C [#]	D	E ^b	E	F	F [#]	G	G [#]	A	B ^b	B
	ド	ド [#]	レ	ミ ^b	ミ	ファ	ファ [#]	ソ	ソ [#]	ラ	シ ^b	シ
周波数(Hz)	1046	1108	1174	1244	1318	1396	1479	1567	1661	1760	1864	1975
音程データ	59	55	52	49	46	44	41	39	37	34	32	31

命令の書き方（“ら”の音を0.8秒ならす場合）

電子音, 8, 34

音の高さ(音程)

音の長さ(電子音命令をつづける時間)



音程 ー り
 数字1 ー 8
 数字2 ー 34

※ 音の高さ(音程)、音の長さの2つがキーポイント

【音だしプログラムの考え方】

4分音符を1拍(下の4の意味)とし4拍で1小節(上の4)

1分で72拍(音) → 1拍は60/72秒 → 1拍 約0.8秒

4分音符
0.8秒

4分音符(1小節を4分割した長さ)

2分音符(4分音符の2倍長)

8分音符(4分音符の1/2長)

1小節

音程	ラ	ラ	シ	ラ	ラ	シ	ラ	シ	ド	シ	ラ	シ	ラ	ファ	ミ	ド	ミ	ファ	ミ	ミ	ド	シ
数字1	8	8	16	8	8	16	8	8	8	8	8	4	4	16	8	8	8	8	8	4	4	16
数字2	34	34	31	34	34	31	34	31	29	31	34	31	34	44	46	59	46	44	46	46	59	62

プログラムの書き方

- ① (音を出す長さ, 音程) の数字をひとまとめにする
- ② 命令の言葉をつくる
- ③ くりかえす

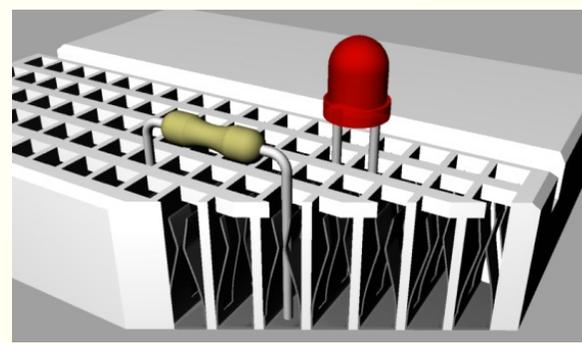
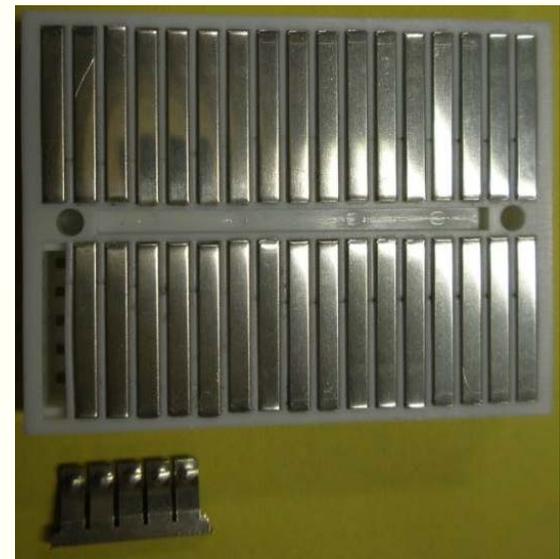
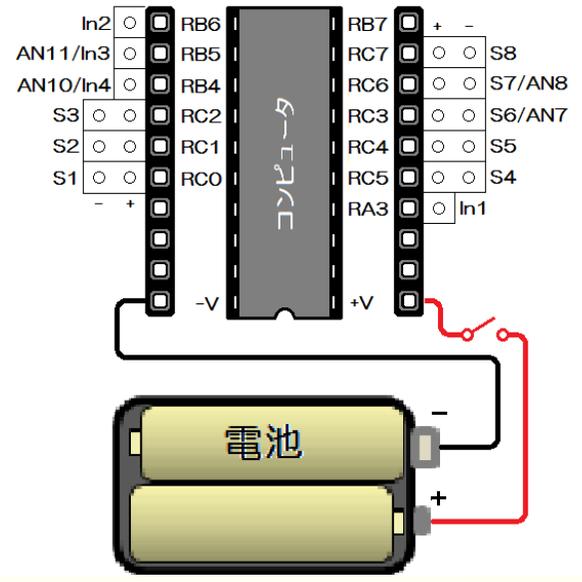
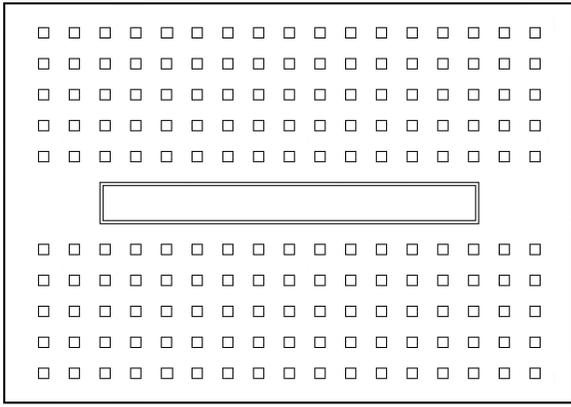
ラ	↓	電子音, 8, 34
ラ	↓	電子音, 8, 34
シ	↓	電子音, 16, 31
.	
.	

——計測制御へと応用してみよう——

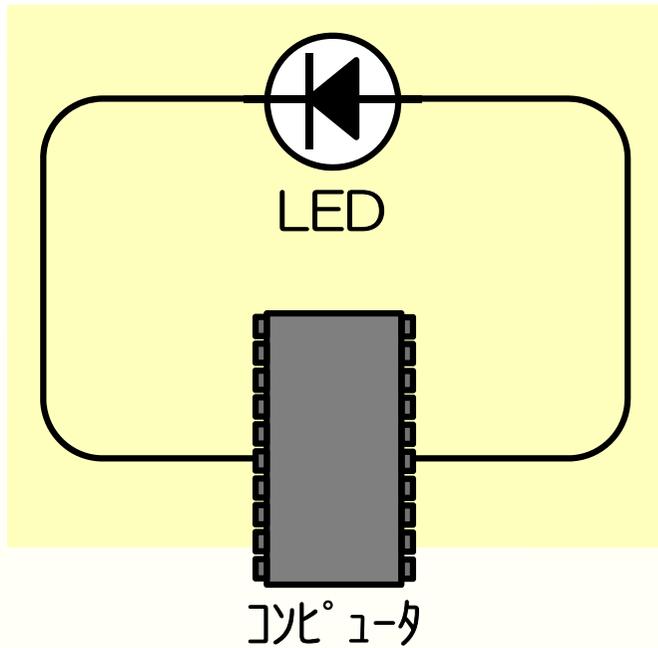
1. 電気配線について
2. 制御技術の基本
3. 計測技術の基本
4. 計測・制御への応用

内部の電気配線 ブレッドボード

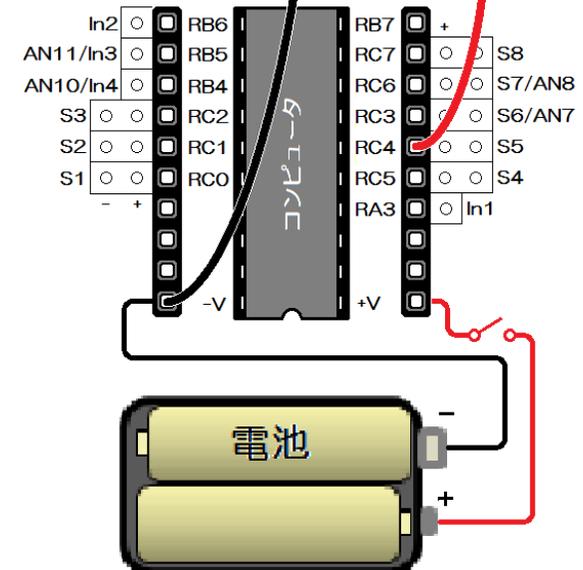
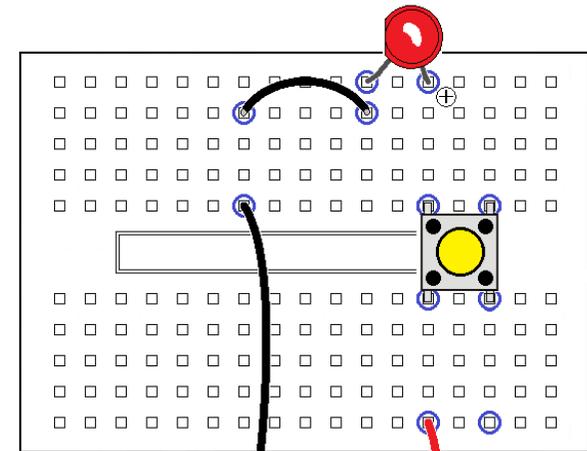
ブレッド
ボード



LEDを制御して点灯



※回路にはLEDやコンピュータポートの保護のため抵抗(100Ω程度)を入れた方がよい



ミュウロボエディタ画面

USB ●

開く

保存

転送

計測/R実行

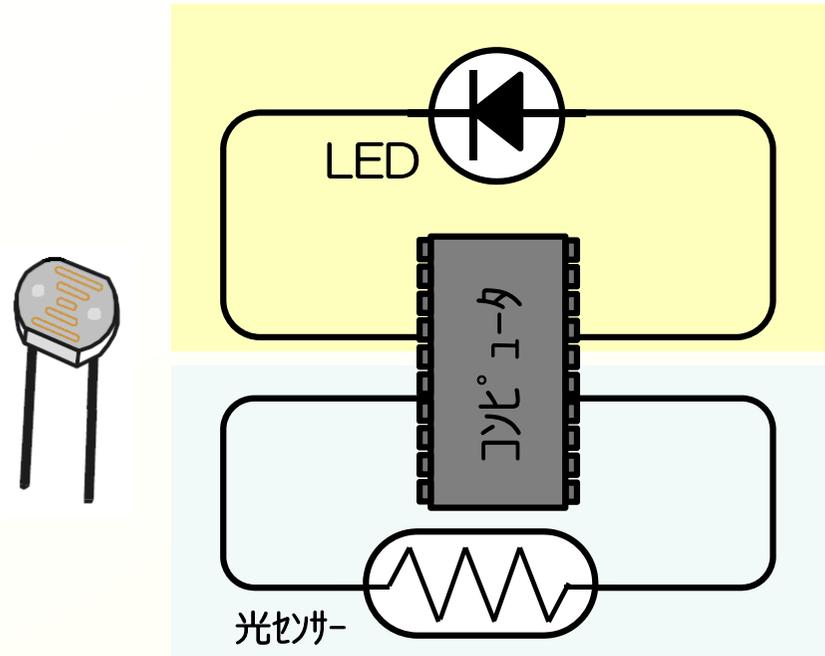
読み込

出力ポート

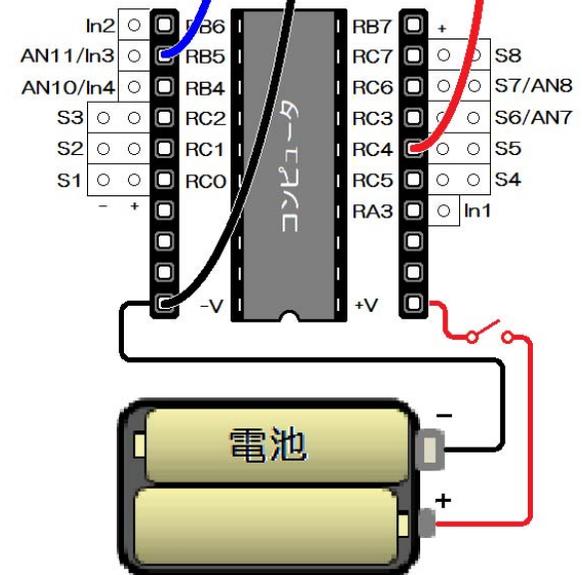
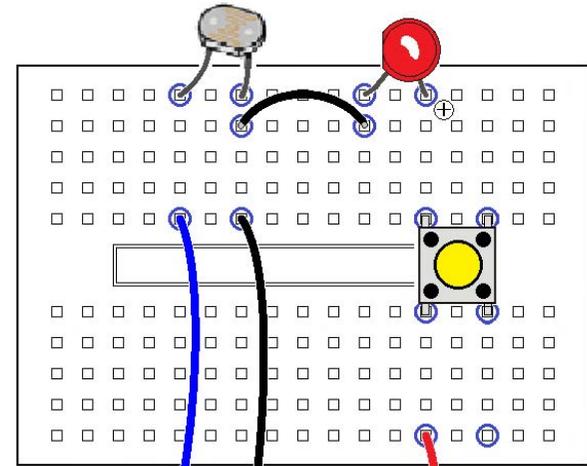
RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

PORTC = 16(10)

光の量で違った動作をさせよう



※回路にはLEDやコンピュータポートの保護のため抵抗(100Ω程度)を入れた方がいい



ミュウロボエディタ画面

USB- ● 開ク 転送 計測/R実行

読込

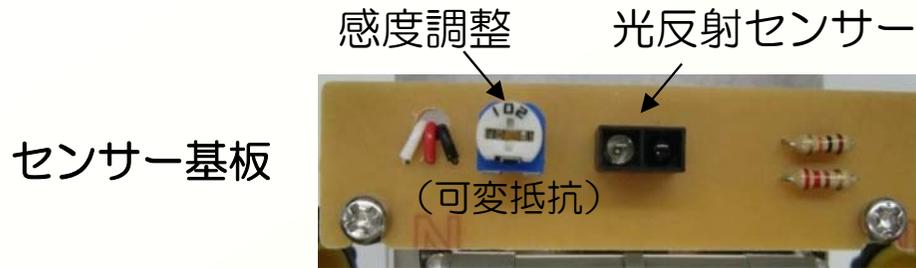
入力ポート
In2-● In3-○ In4-●

AN11 10BIT

デジタル計測 AN11計測

PORTB = 80(50)
AN11 = 70(0046)

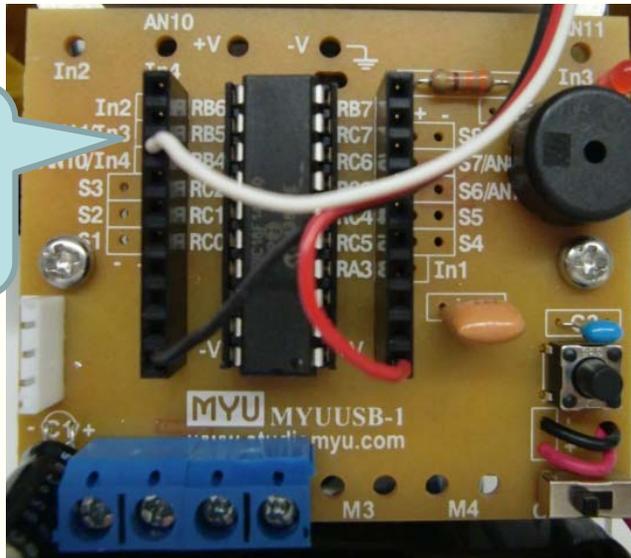
ライトレースロボットを作ってみよう



配線

(白線) → In4
(赤線) → +V
(黒線) → -V

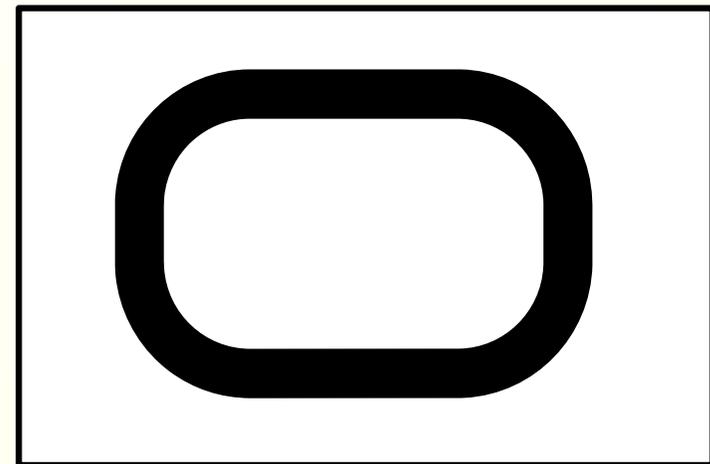
※白線を他の入力ポートに繋がせた場合、右のプログラム中の青字の数字を対応するポートの番号に変える(例. In3に繋がせた時は3にする)



(枠からはみ出ないロボットのプログラム例)

```
アンカー,10  
入力なしジャンプ,20,4  
右回り,1  
停止,1  
ジャンプ,10  
アンカー,20  
前進,0  
ジャンプ,10
```

```
繰り返す{  
    入力なし繰り返す(4){  
        前進,0  
    }  
    右回り,1  
    停止,1  
}
```



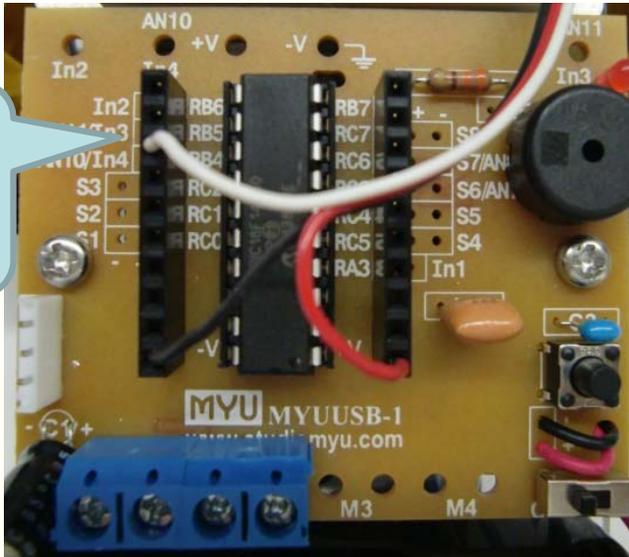
枠からはみ出ないロボットを作ってみよう



配線

(白線) → In4
(赤線) → +V
(黒線) → -V

※白線を他の入力ポートに繋げた場合、右のプログラム中の青字の数字を対応するポートの番号に変える(例. In3に繋げた時は3にする)



(枠からはみ出ないロボットのプログラム例)

```
アンカー,10  
  入力なしジャンプ,20,4  
  前進,0  
  ジャンプ,10  
アンカー,20  
  右回り,1  
  ジャンプ,10
```

```
繰り返す{  
  入力なし繰り返す(4){  
    右回り,1  
  }  
  前進,0  
}
```

