実践ロボットプログラミング

LEGO Mindstorms EV3 で目指せロボコン！

CU-Robocon講習会

WEB：http://robot-programming.jp/
担当：藤吉弘亘（中部大学工学部ロボット理工学科）
E-mail：hf@cs.chubu.ac.jp

■ ロボットとは

ロボットとは？

- その定義を考えてみよう

国語大辞典によると：

「人間に類似した形態をもち、自動的に作業を行う機械装置」

おすすめ：

「感じる、判断する、動くの三つがそなわっている人工物」

（東嶋和子）

ロボットにおける要素技術

制御技術

状況判断に合わせた制御

（制御理論）

感じる、**判断する**、**動く**がそなわっている**人工物**

センシング技術

外界の情報をセンサを介して取り込む

（カメラ，レーザーセンサ等）

人工知能技術

センサ情報を基に状況を判断

（機械学習）

■ LEGO Mindstormsについて

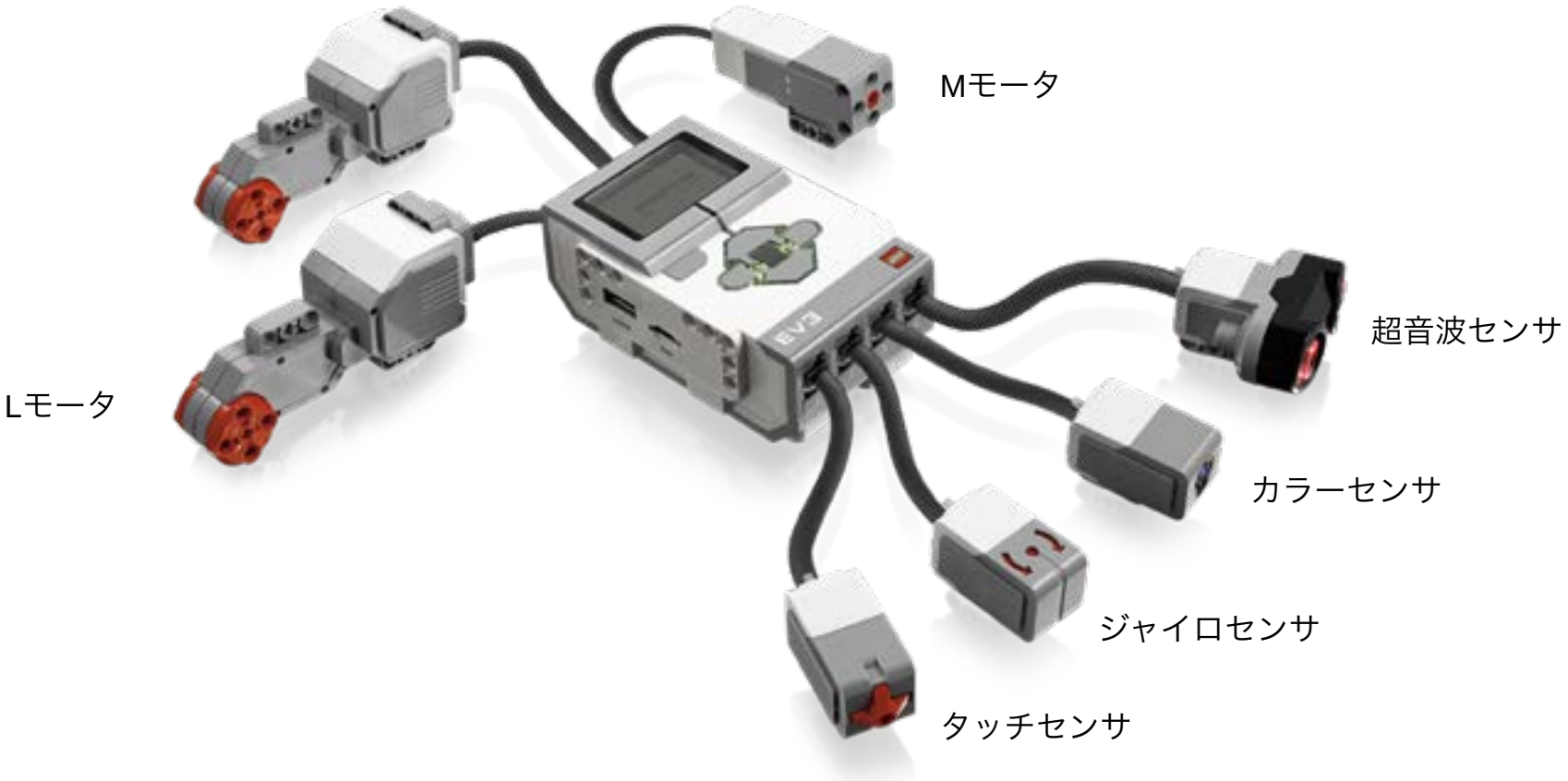
第一世代RCX　第二世代：NXT　第三世代：EV3



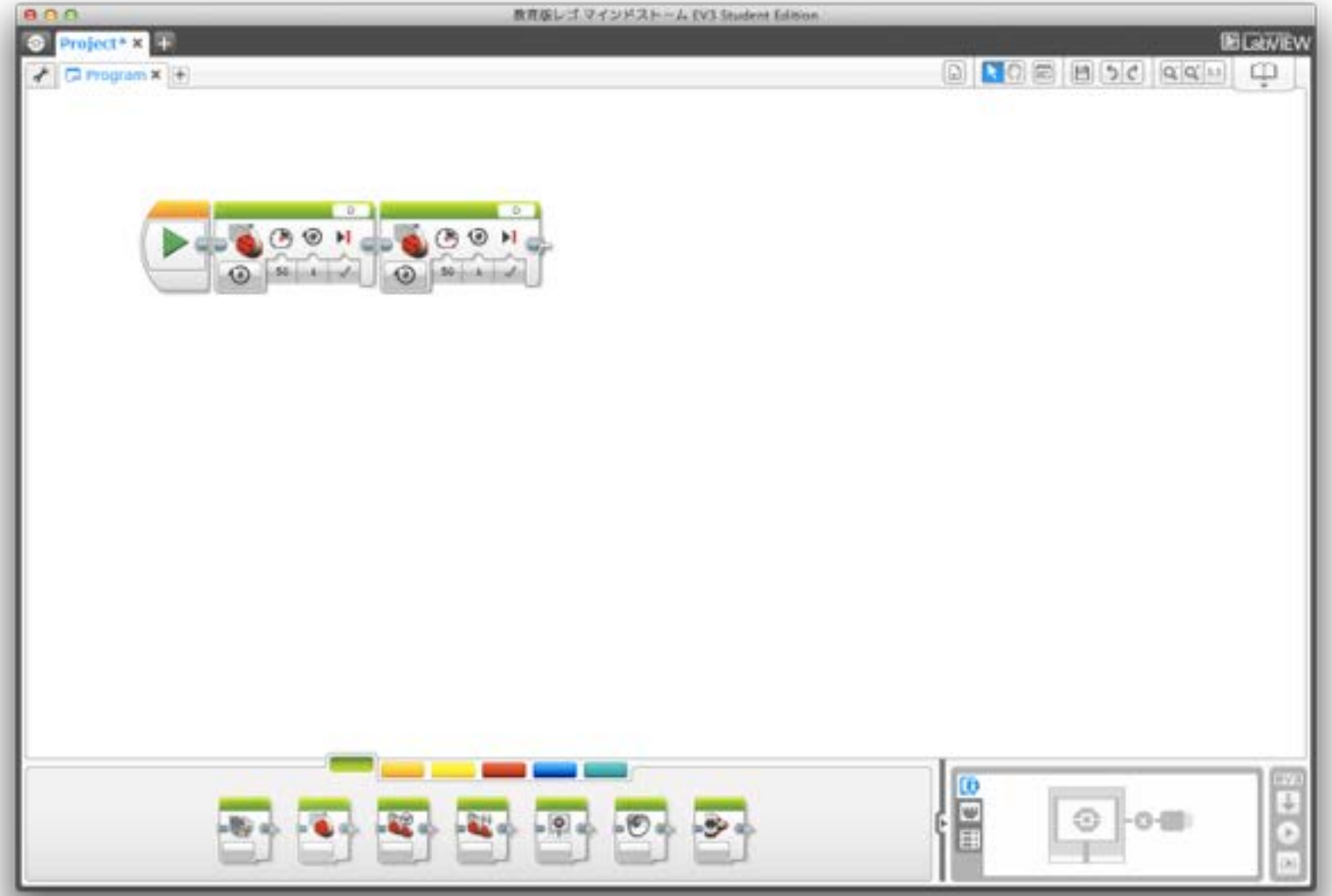
	RIS	NXT	EV3
発売時期	1998年	2006年	2013年
CPU	H8（8 bit）	ARM7（32 bit）	ARM9（32 bit）
クロック周波数	16MHz	48MHz	300MHz
RAM	32KB	64KB	64MB
フラッシュメモリ	なし	256KB	16MB
転送方法	赤外線通信	USB/Bluetooth	U/B + WiFi
ポート数	入力:3 出力:3	入力:4 出力:3	入力:4 出力:4
駆動	電池	電池／バッテリーパック	電池／バッテリーパック

LEGOロボットEV3の構成

- 入力：超音波センサ、カラーセンサ、ジャイロセンサ、タッチセンサ
- 出力：Lモータ、Mモータ



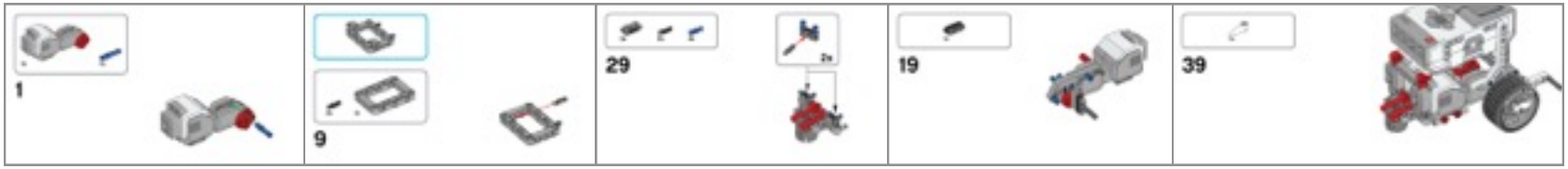
プログラミング環境：EV3ソフトウェア



■ロボットの組み立て

ロボットの組み立て

ロボットエデュケーション⇒組み立てガイド⇒**トレーニングボット**

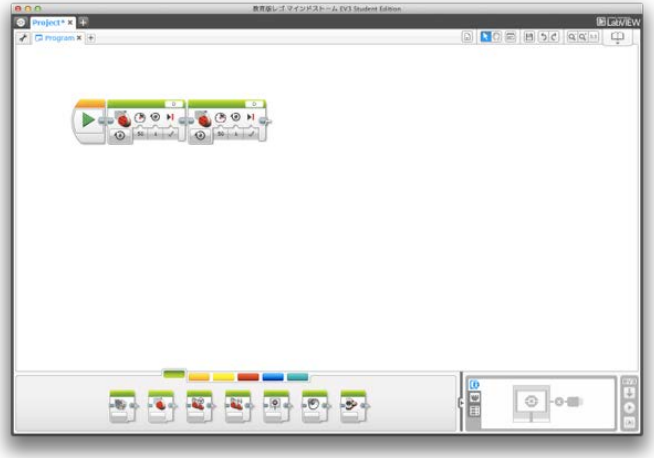


説明書を参考にトレーニングボットを組み立てましょう！

■プログラムを作成するには

プログラム実行までの流れ

1. PC上でソフトウェア(EV3-SW)を用いてプログラムを作成
2. USB/Bluetoothでロボットへダウンロード
3. ロボット上でプログラムを実行



USB

Bluetooth

EV3ソフトウェア

1. EV3ソフトウェアを起動
2. 新規プロジェクトの作成
   - 新規プロジェクト⇒プログラム⇒開く

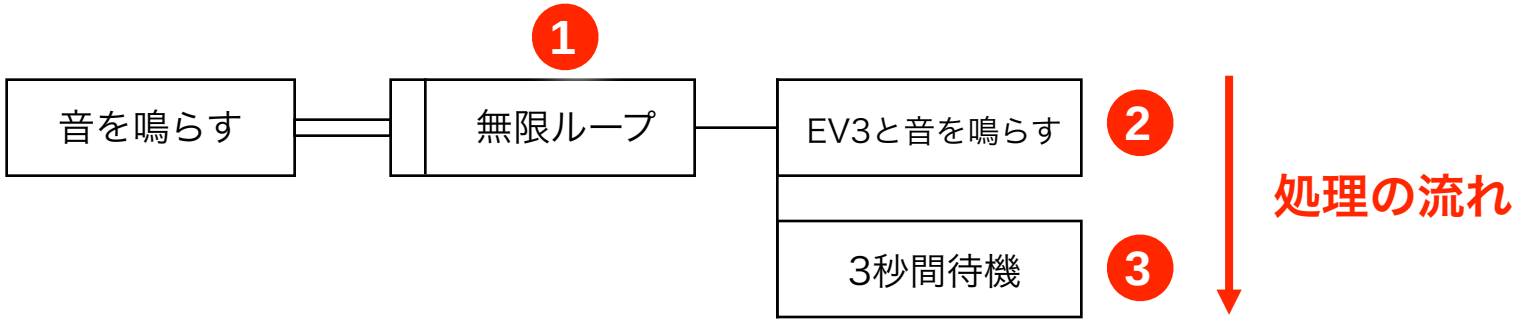
EV3ソフトウェアの画面構成

プログラミングパレット

■音をならしてみよう

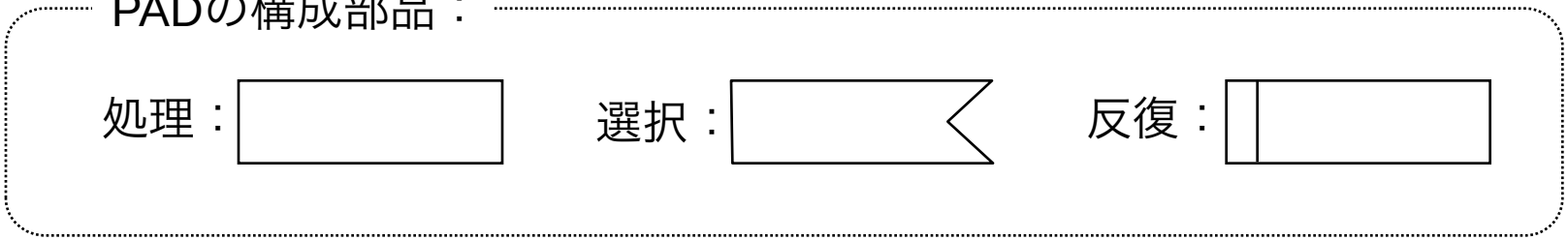
音を鳴らすプログラムの設計図(PAD)

・ 指定したサウンドファイルを無限ループで再生



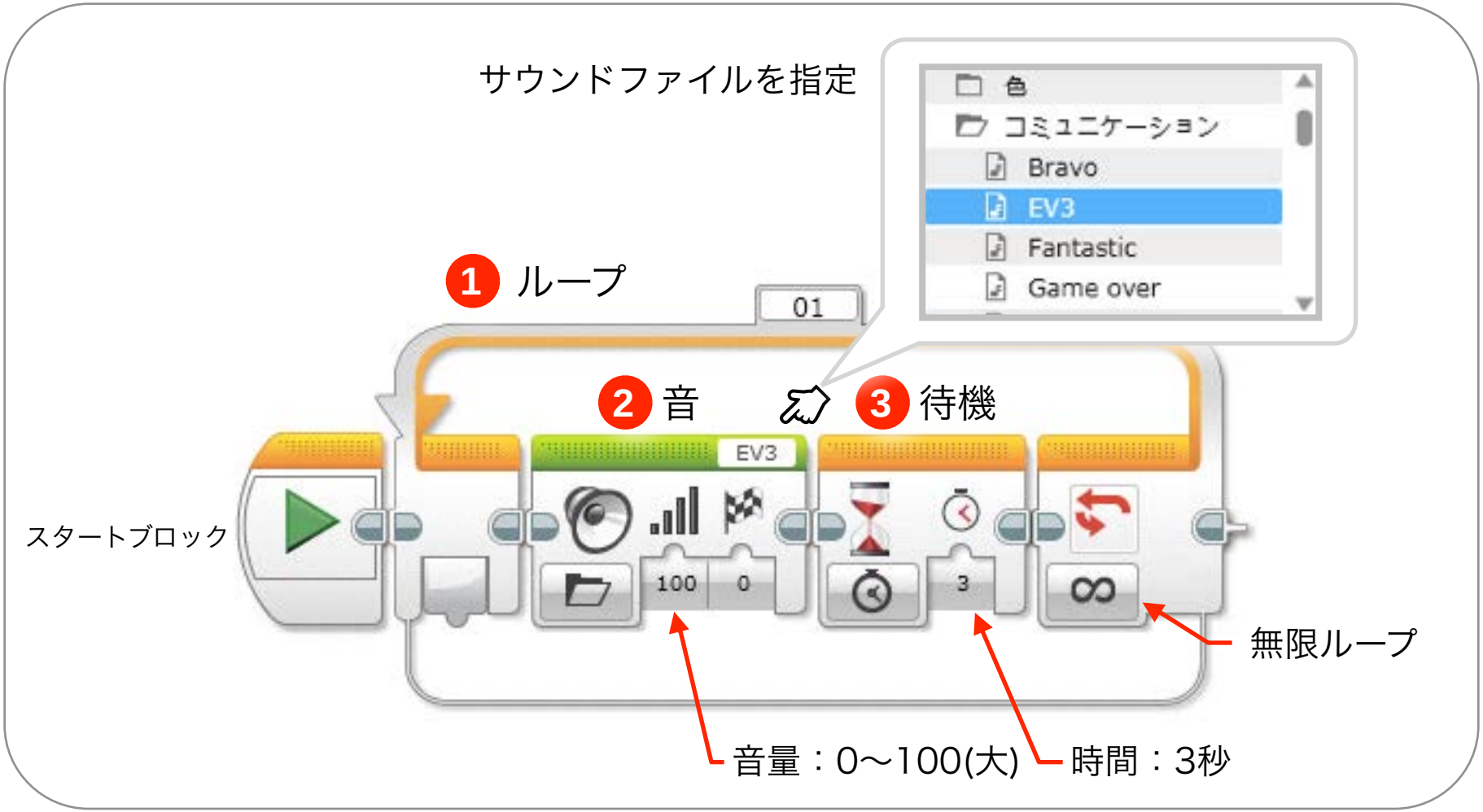
PAD(Problem Analysis Diagram)はプログラムの設計図

PADの構成部品：



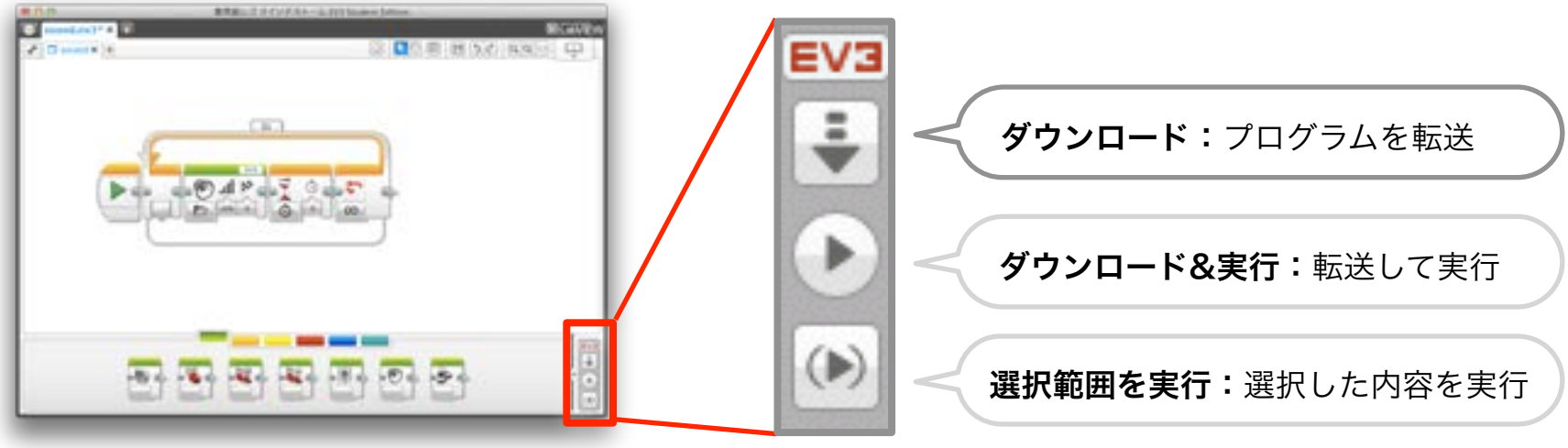
音を鳴らすプログラム

・ 指定したサウンドファイルを無限ループで再生

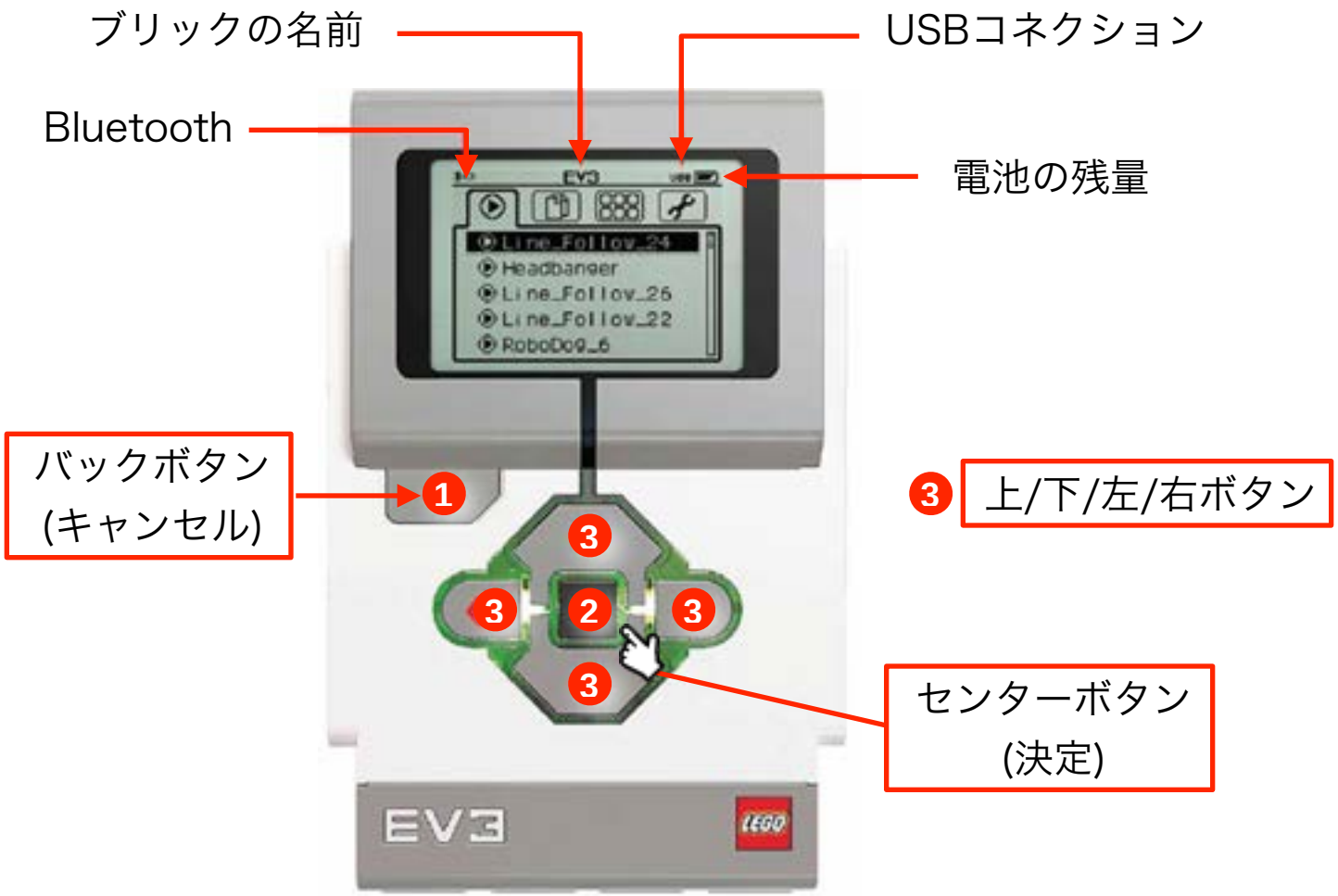


プログラムの転送

1. プログラム名を”sound”に変更
   - プログラム名は半角の英数字のみ
2. ロボットが接続されているか確認 ⇒ ×: ○:
3. ダウンロードボタンによりプログラムを転送 ⇒
4. ロボット上でプログラムを実行



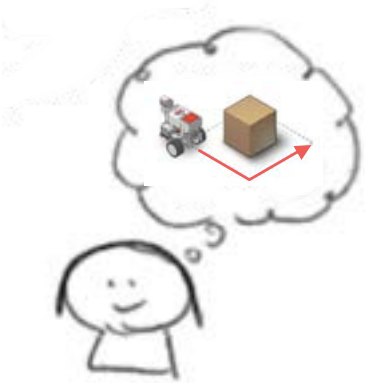
プログラムの実行



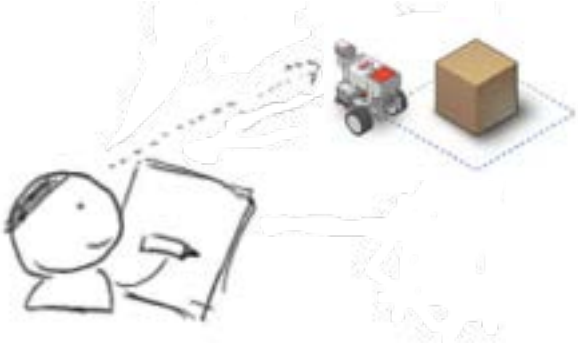
プログラムを選択してセンターボタンで実行
（プログラム終了はバックボタン）

プログラムの実行

- 実行前にすべきこと
  - ケーブルを外して、安全確認
  - プログラムのアルゴリズムを頭の中で実行

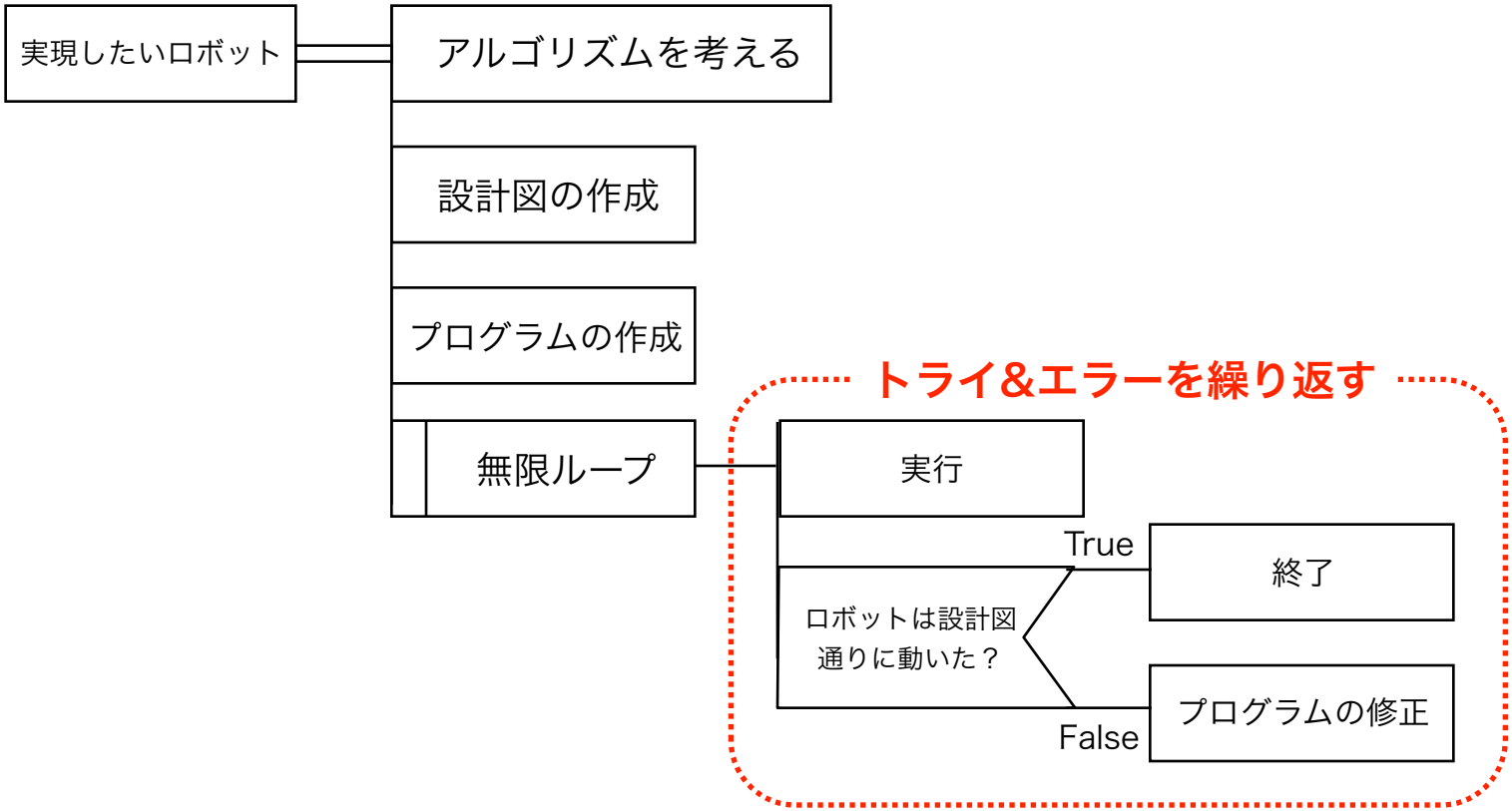


- 実行時の注意
  - ロボットの動作をよく観察し、設計図(PAD)通りにアルゴリズムが実現できているかを確認
  - ロボットが思い通りに動かないときは、ロボットがどこまで設計図通りに動いたかを調べ、プログラムを修正（デバッグ）する



トライ＆エラーによる問題解決

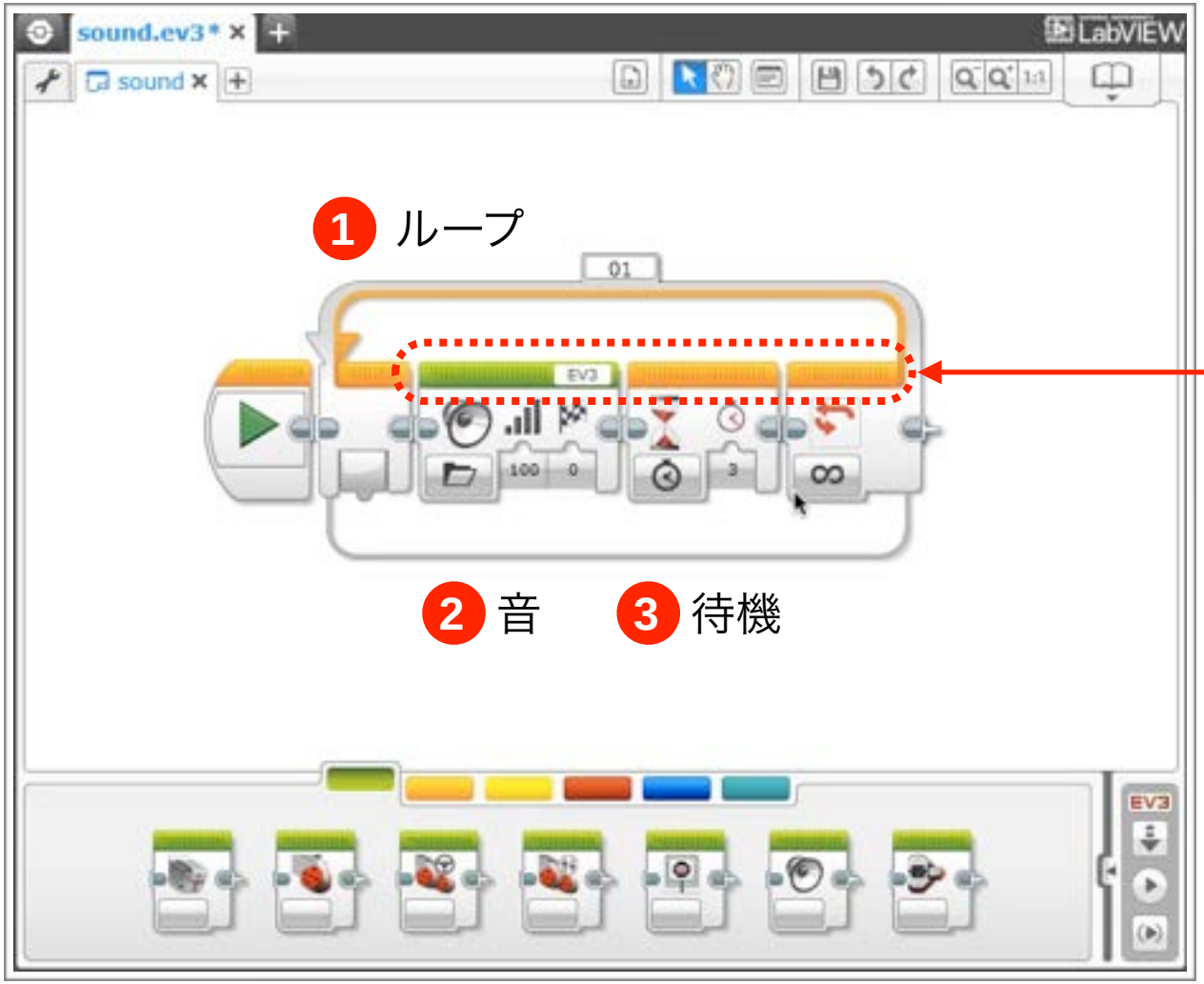
・ ロボットを思い通りに動かすには



→設計図通りにロボットが動くまでトライ＆エラーを繰り返して問題解決

デバッグ方法

- USBケーブルをつなげたままプログラムを実行



ロボットにおける要素技術

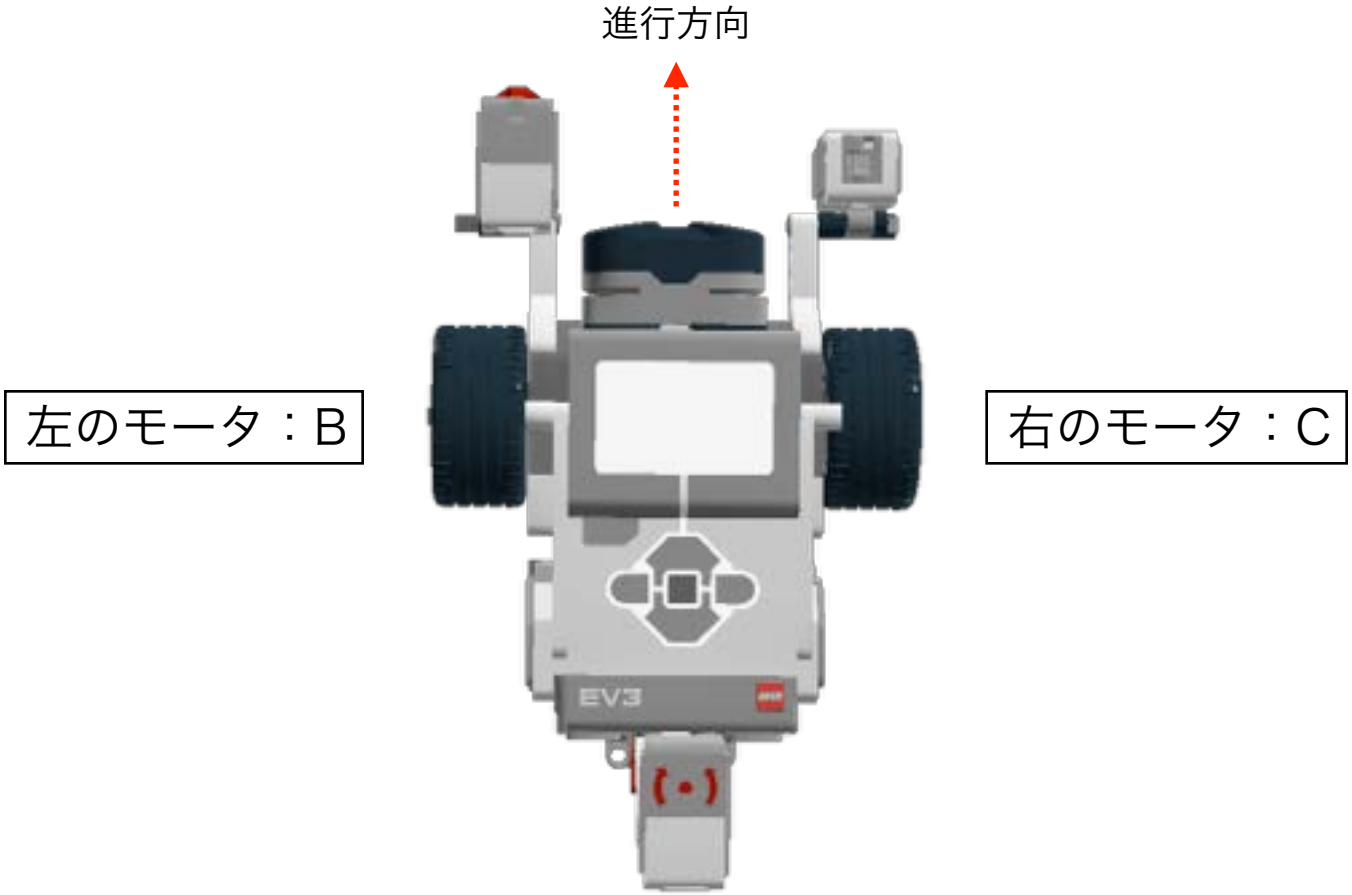
感じる、判断する、動くがそなわっている人工物

■ロボットを前進させるには(モータ制御１)
■ロボットを回転させるには(モータ制御2)
■演算と変数

■ロボットを前進させるには(モータ制御１)

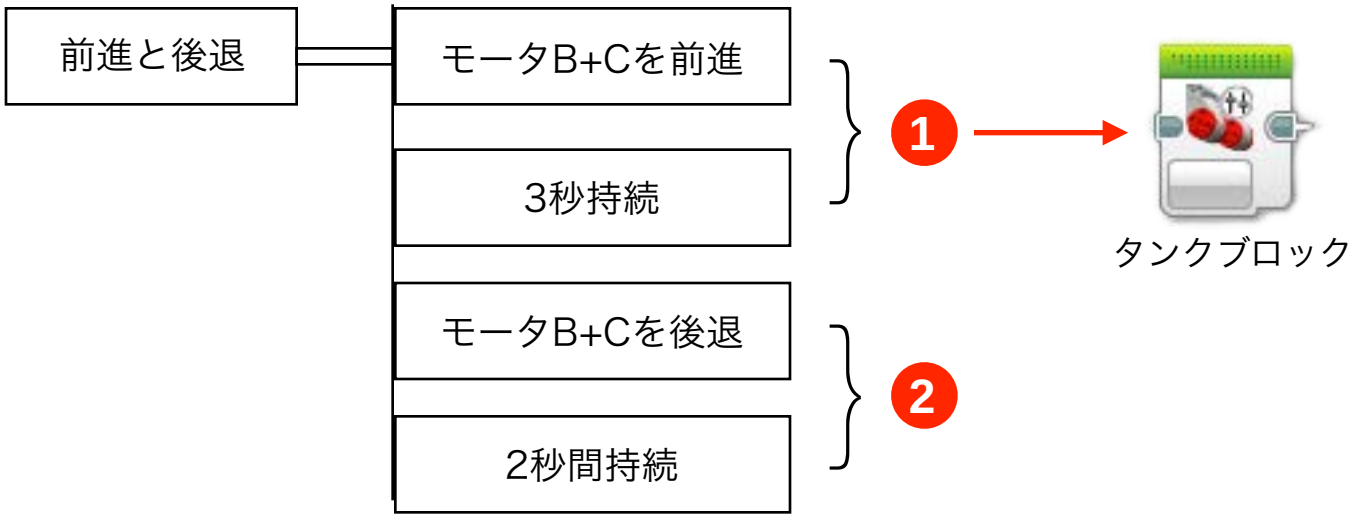
モータの接続

・ EV3のどの出力ポートにモータが接続されているか確認



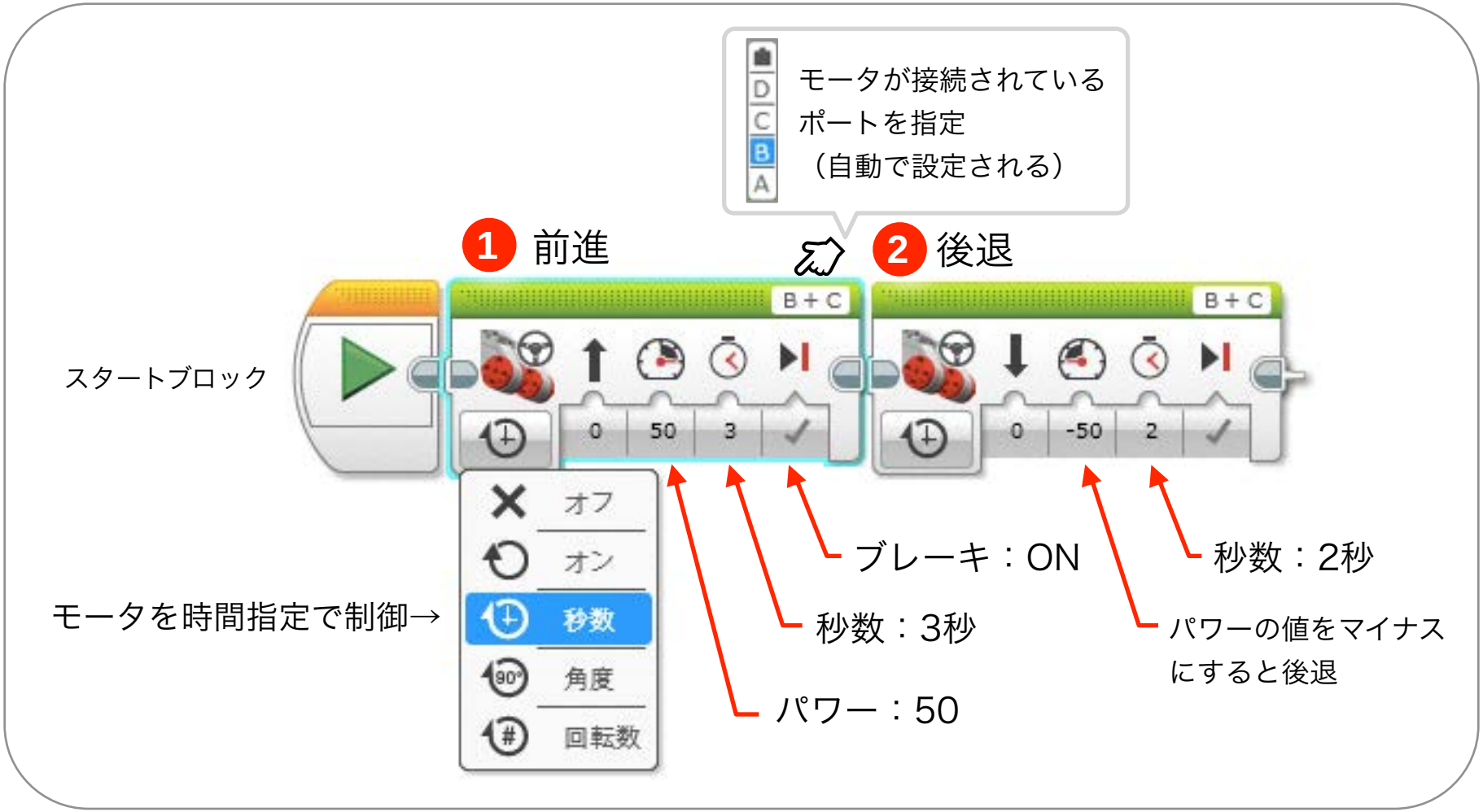
前進プログラムのPAD

・ ロボット(モータB+C)を3秒前進、その後2秒後退



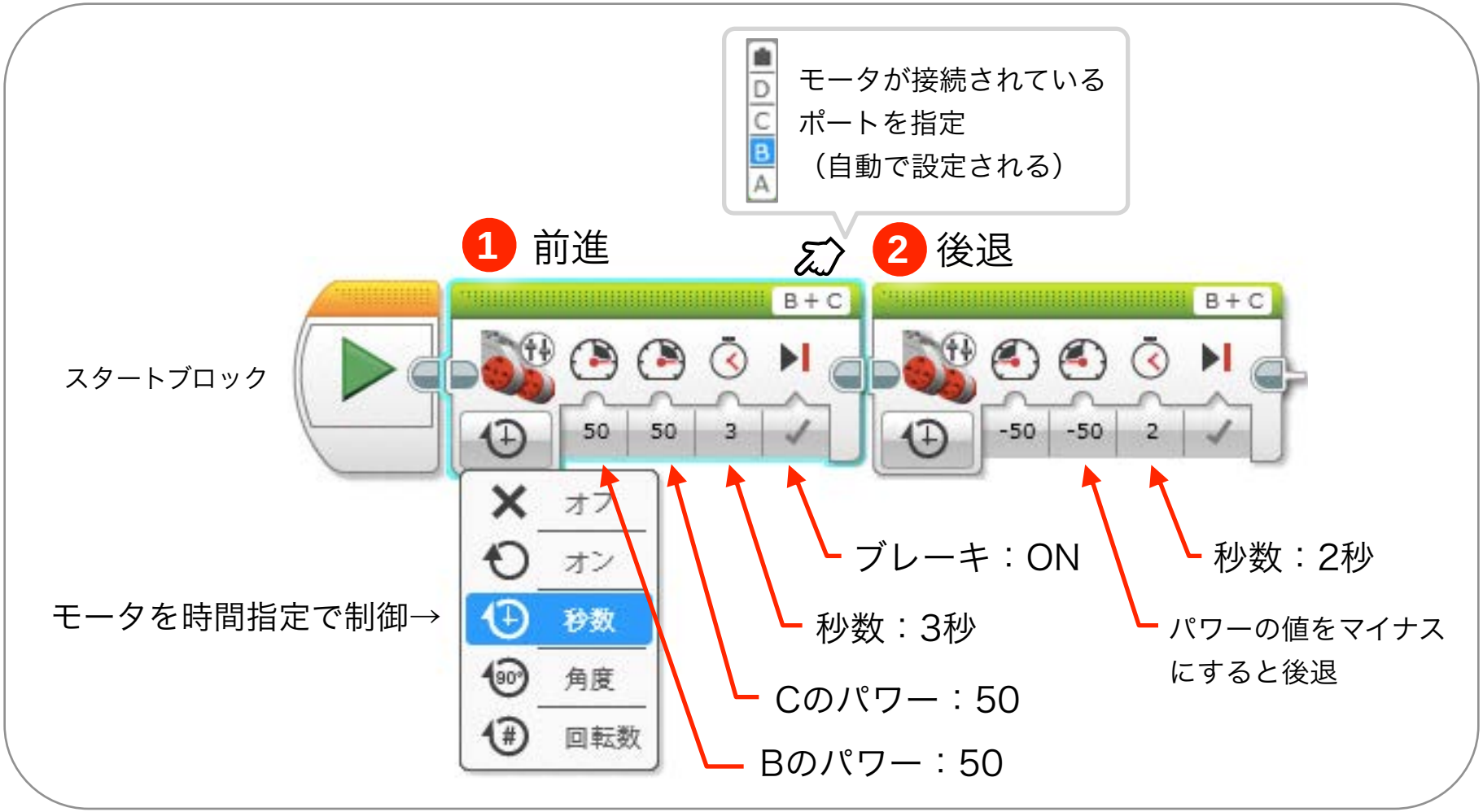
モータ制御によるロボットの前進 (ステアリングブロック)

・ ロボット(モータB+C)を3秒前進、その後2秒後退

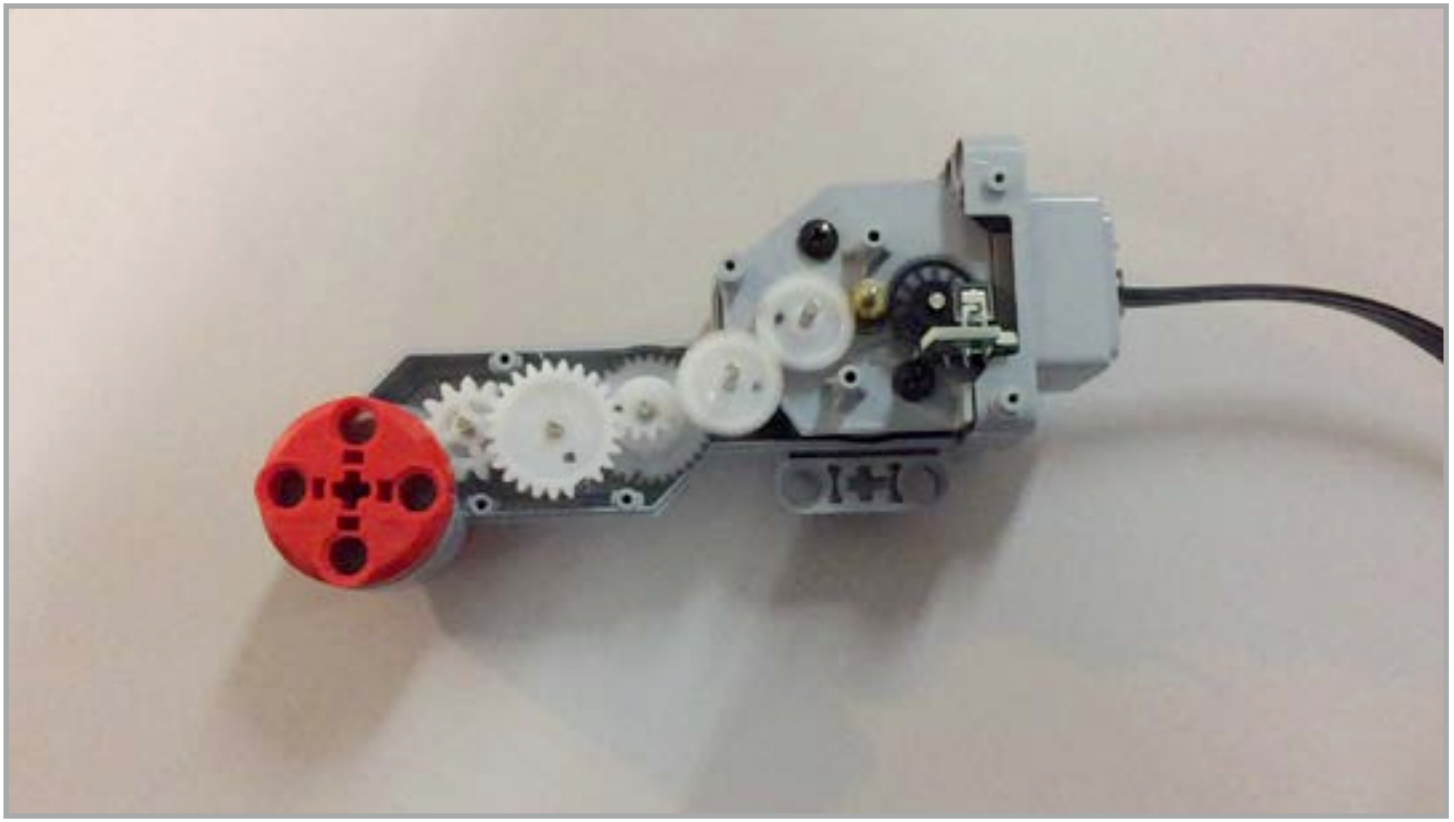


モータ制御によるロボットの前進 (タンクブロック)

・ ロボット(モータB+C)を3秒前進、その後2秒後退



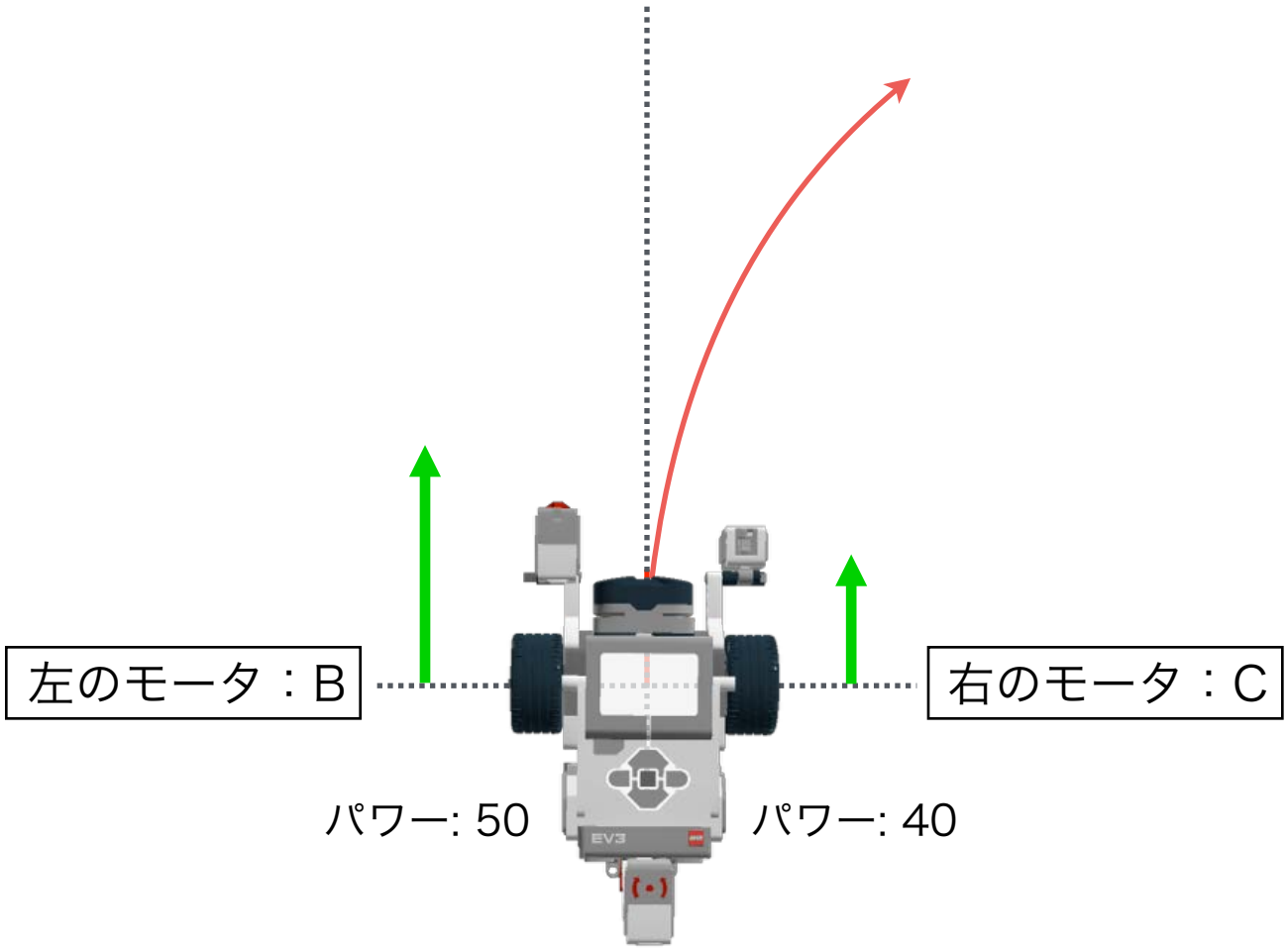
Lモータの原理



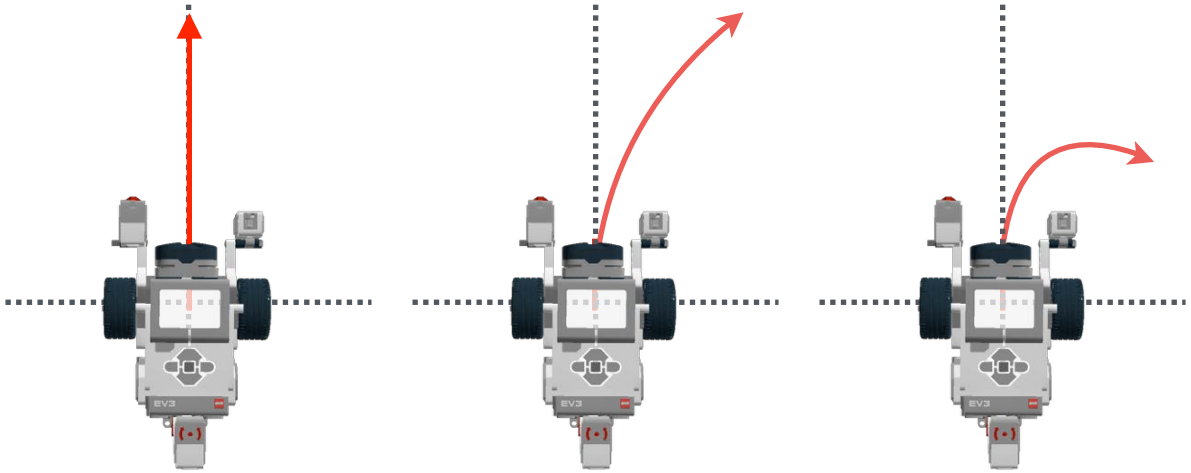
段数：7段　トルク比率：45:1　エンコーダの回転角：2度

曲線の動きを実現するには

- モータBとCのパワーの値のバランスを変えてみよう



曲線の動きを実現するには

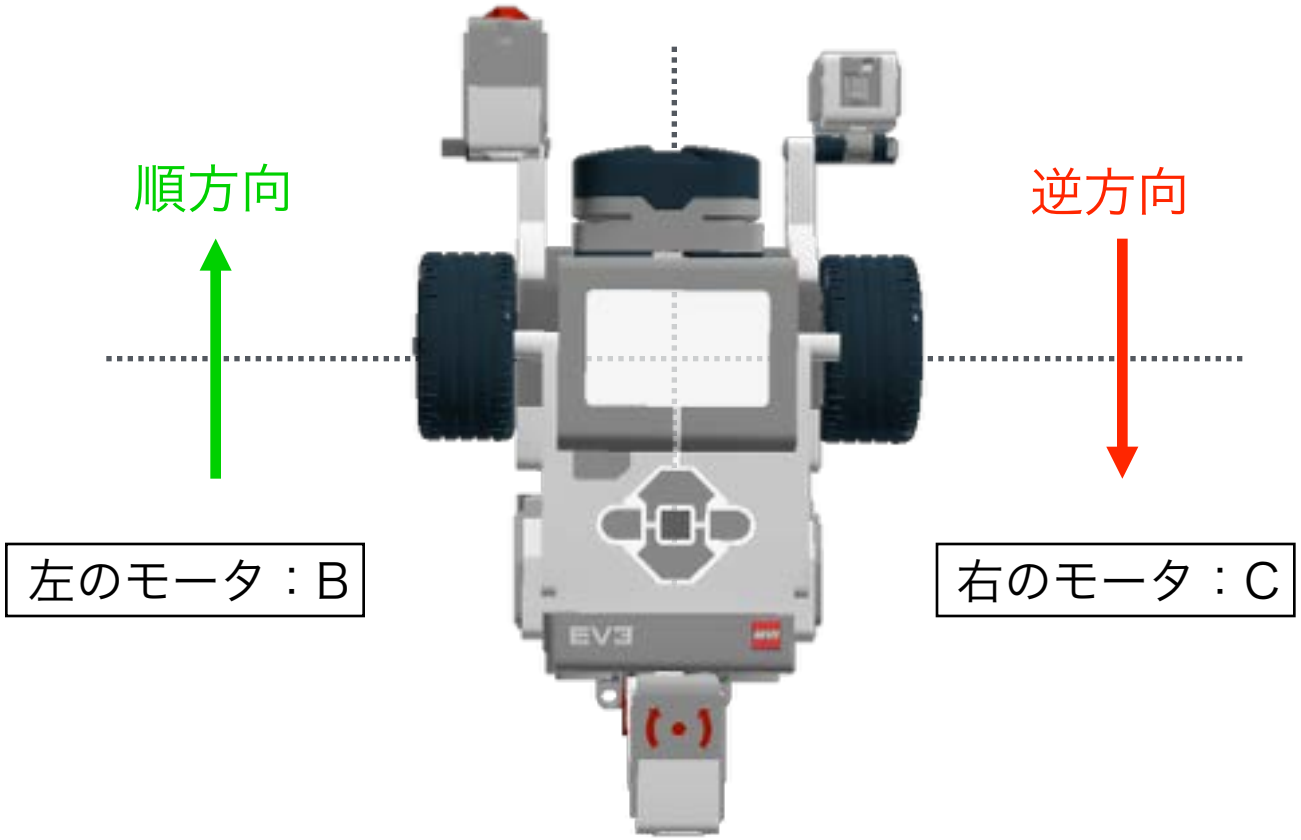


タンクブロック 	B: 50　C: 50	B: 50　C: 40	B: 50　C: 30
ステアリングブロック 	ステアリング: 0	ステアリング: 10	ステアリング: 20

■ロボットを回転させるには(モータ制御2)

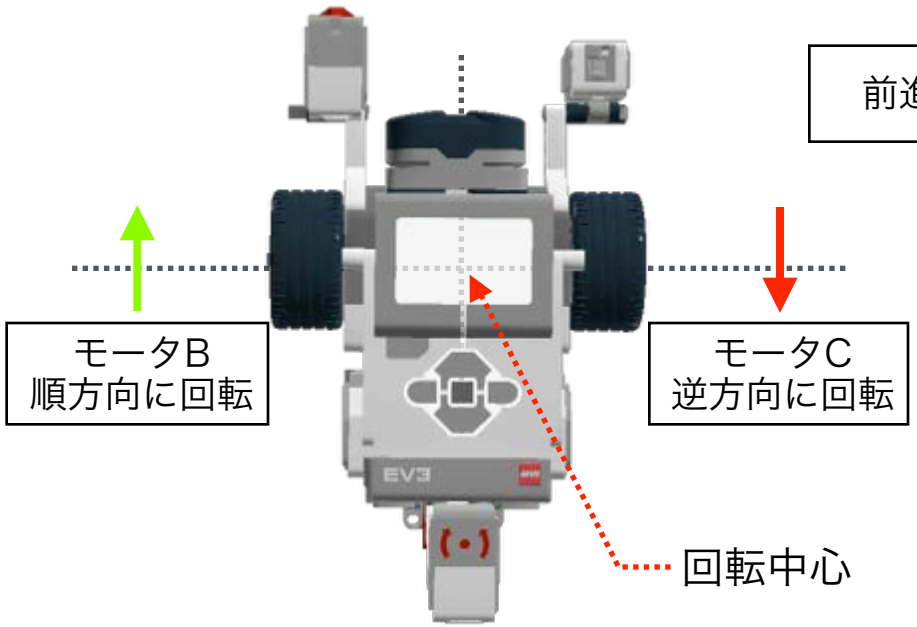
ロボットの回転

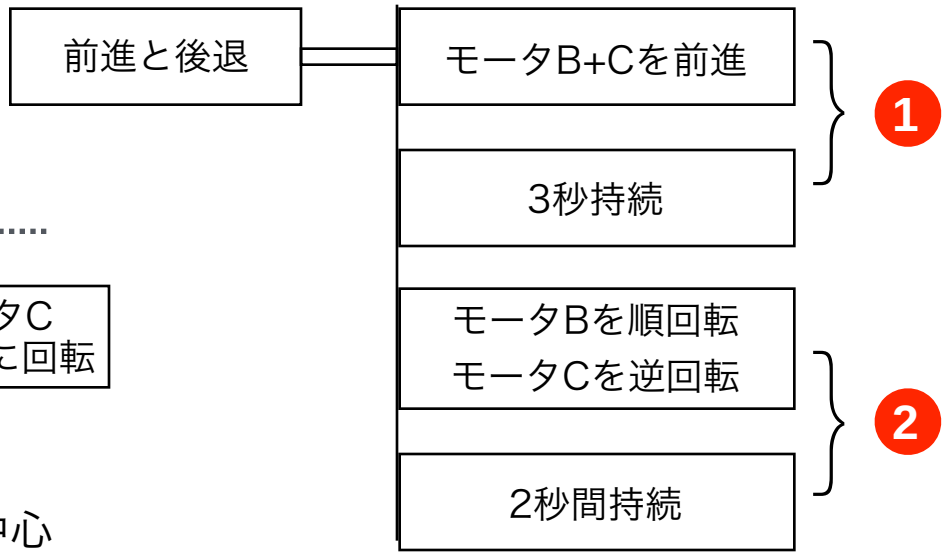
・ ロボットを右回転(その場で旋回)させるには



右回転プログラムのPAD

・ 左のモータBを順回転、右のモータCを逆回転



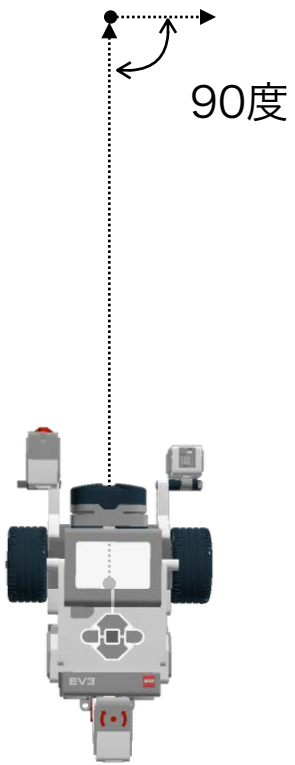


ロボットの右回転

・ 左のモータBを順回転、右のモータCを逆回転

ロボットを90度右回転させるには？

・ どのように実現するか考えてみよう！



ヒント

1. 時間制御
モータのパワーと持続時間を調整

2. 回転制御
モータの回転数もしくは角度を調整

→複数の問題解決方法があるので、いろいろと試してみよう！(試行錯誤しよう)

ロボットを90度右回転させるには

1. 時間制御

モータのパワーと持続時間を調整

持続時間：1.5秒

パワーを変更すると
→持続時間の再調整が必要

2. 回転制御

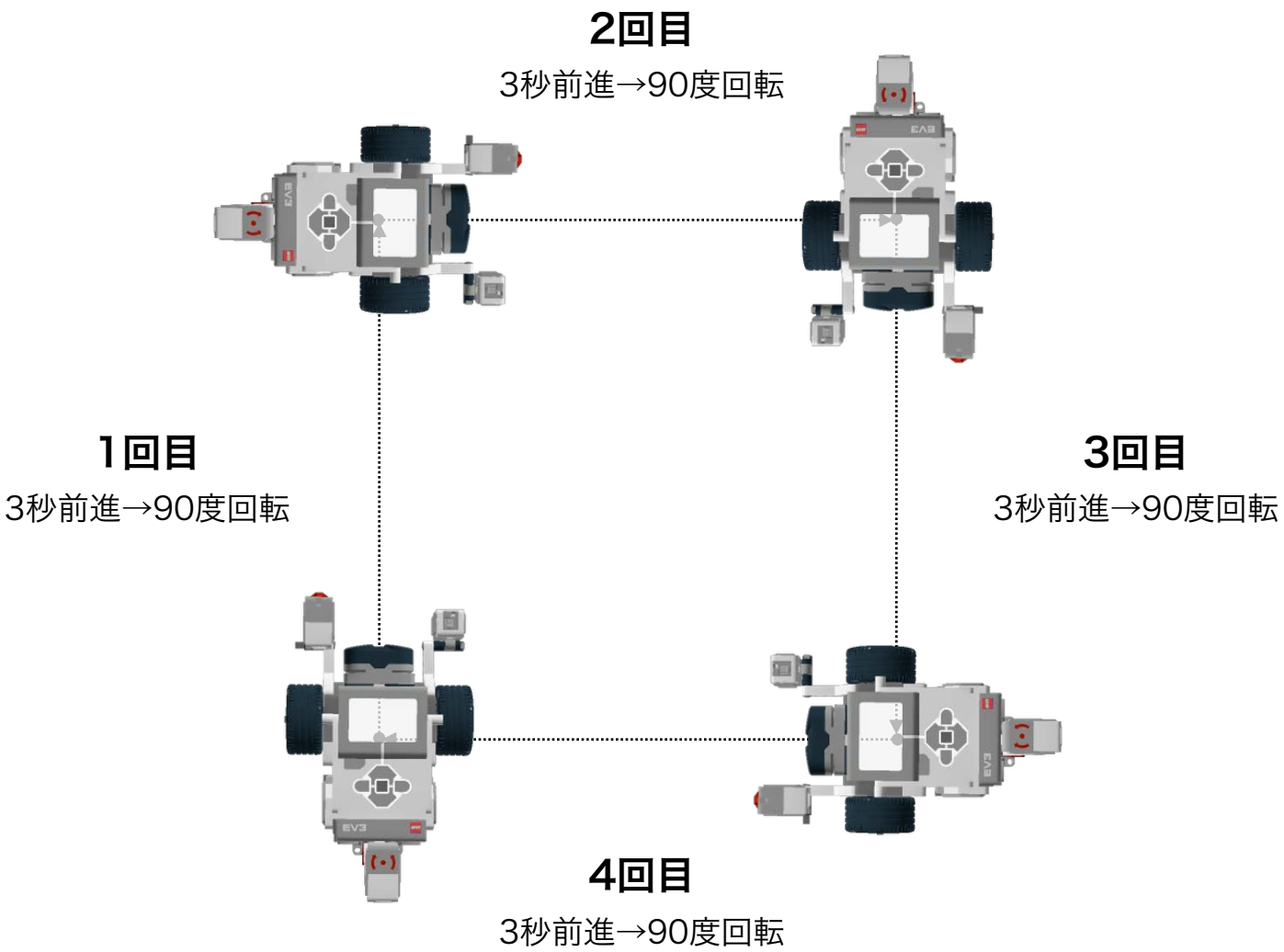
モータの回転数もしくは角度を調整

回転角：180度

パワーを変更しても
→回転角の再調整は必要なし

→回転制御の方が便利(同じ動きでロボットを早くしたいとき等)

ロボットを一周させるには？



ロボットを一周させるには？

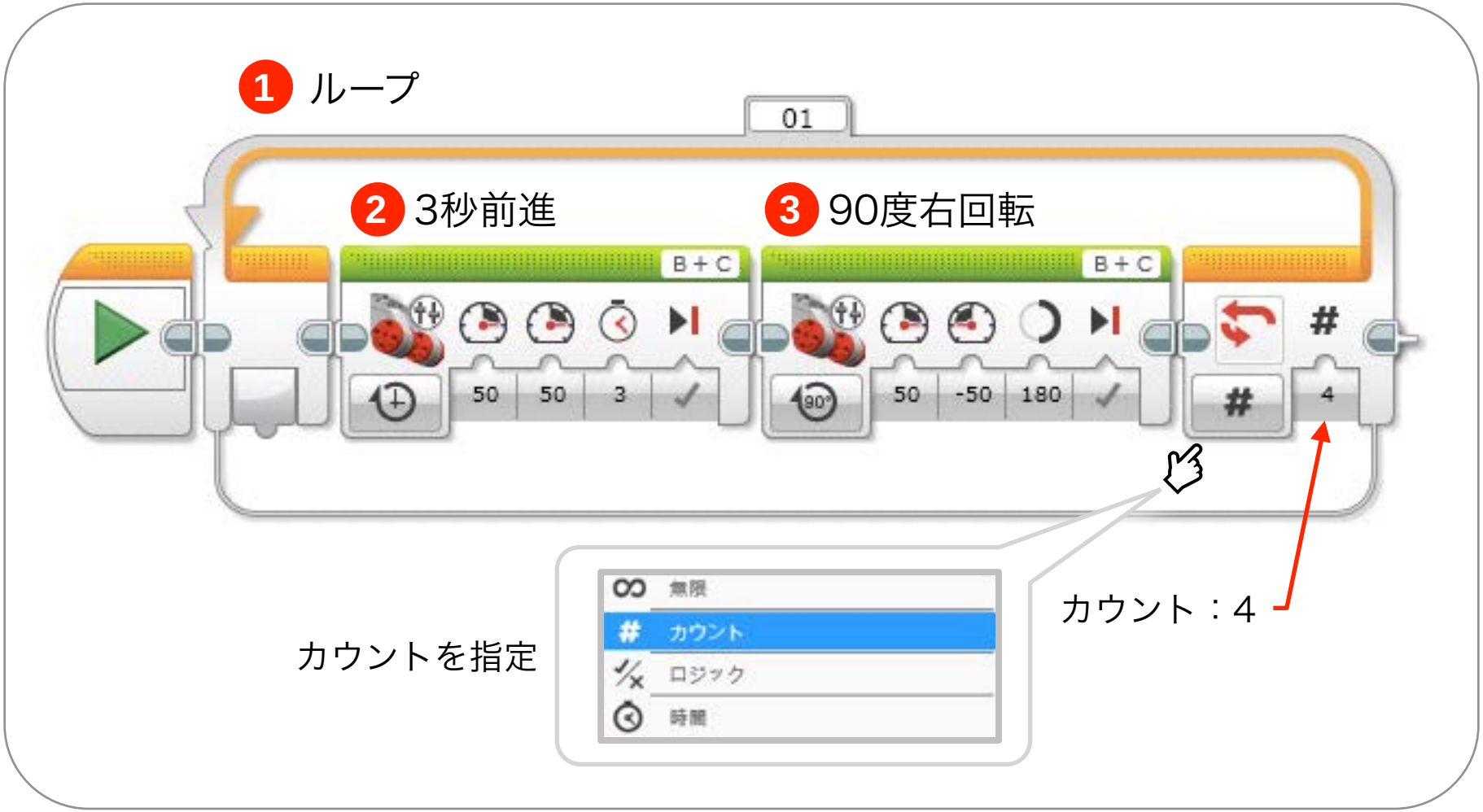
- 3秒前進と90度右回転を4回繰り返せばよい

100周するには？→800個のブロックを並べる必要があり！

- 繰り返し処理(ループブロック)を利用

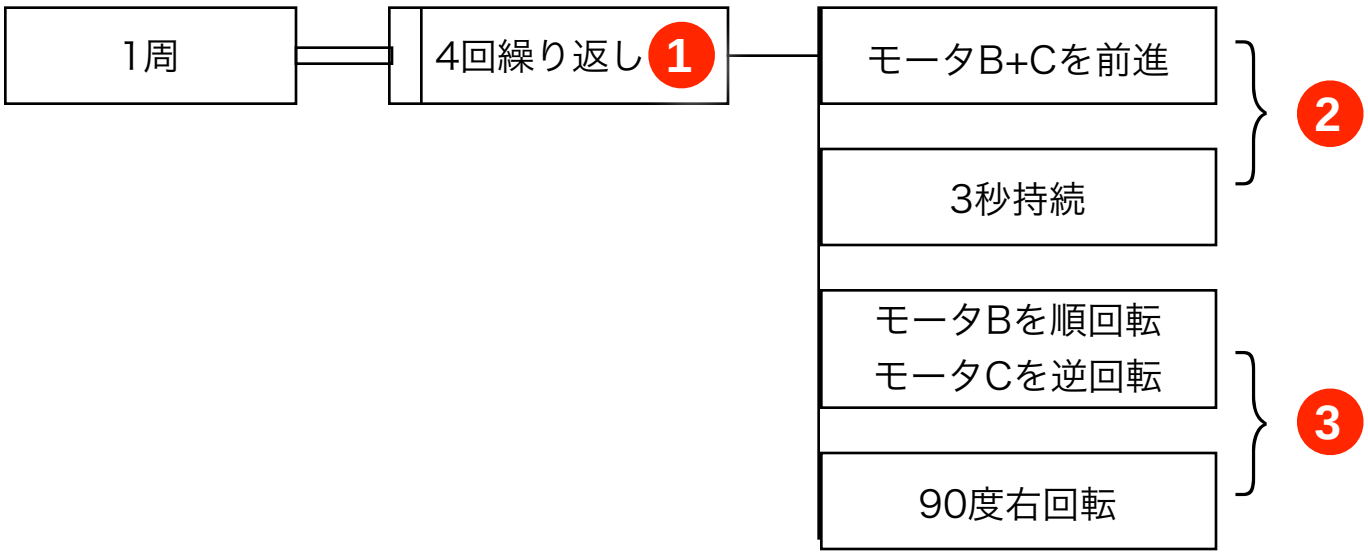
ロボットを一周させるには

- 3秒前進と90度右回転を4回繰り返す



一周するプログラムのPAD

- 3秒前進と90度右回転を4回繰り返す



PADの構成部品：

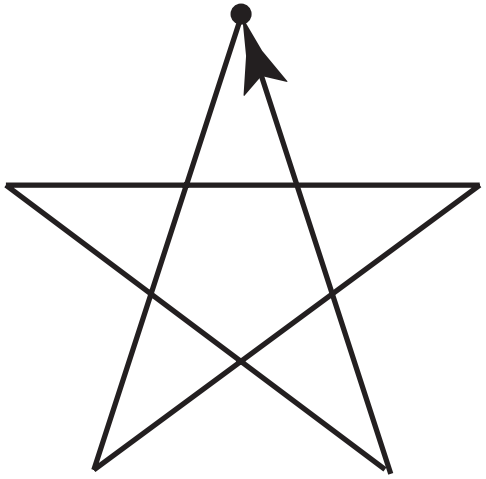
処理：

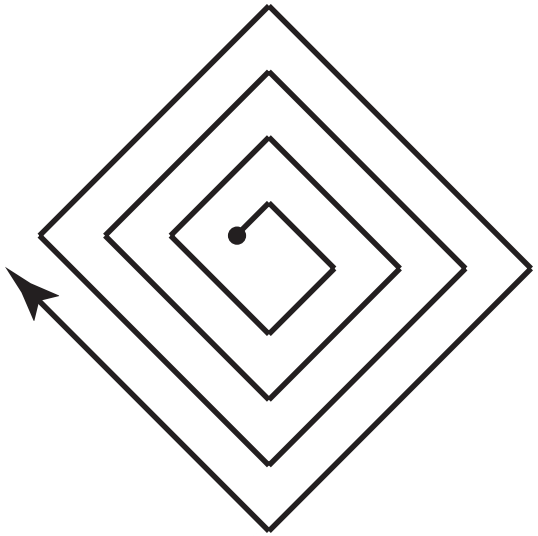
選択：

反復：

■■　チャレンジ！　■■

・　星形やスパイラルの軌跡を描くロボットを実現してみよう！





■演算と変数

50cm前進するには？

・ モータの回転角を変化させたときの直進距離を測定し法則をみつける

角度：
180～1440度

角度[°]	距離[cm]	1cmあたりの回転角度
180	9.0	20.0
360	17.5	20.6
720	34.5	20.8
1080	52.5	20.5
1440	69.5	20.4

中央値

※1cmあたりの回転角度 ＝回転角度[°] / 距離[cm]

回転角度 = 直進したい距離 × 1cmあたりの回転角度

20.5度

50cm前進するには？

- 四則演算結果をモータの回転角度に反映して前進制御

a□b=

演算子を選択

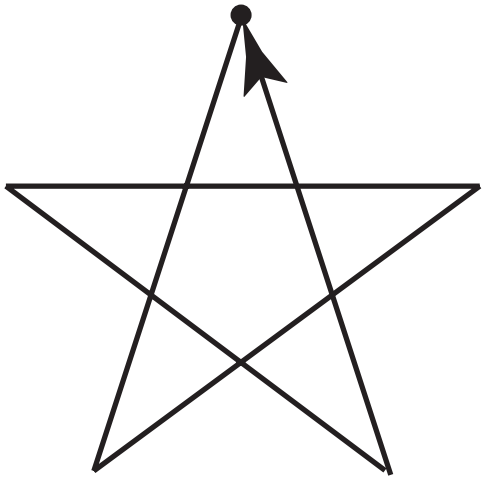
a×b=の結果を引き渡す(データワイヤ)

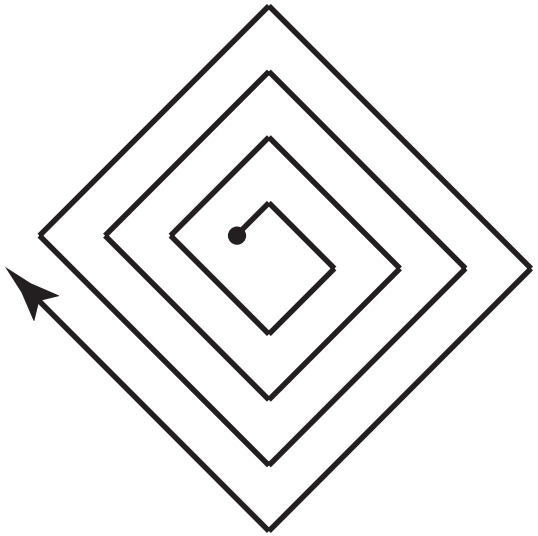
1 cmあたりの回転角度：20.5

直進したい距離：50cm

スパイラル軌跡に挑戦しよう

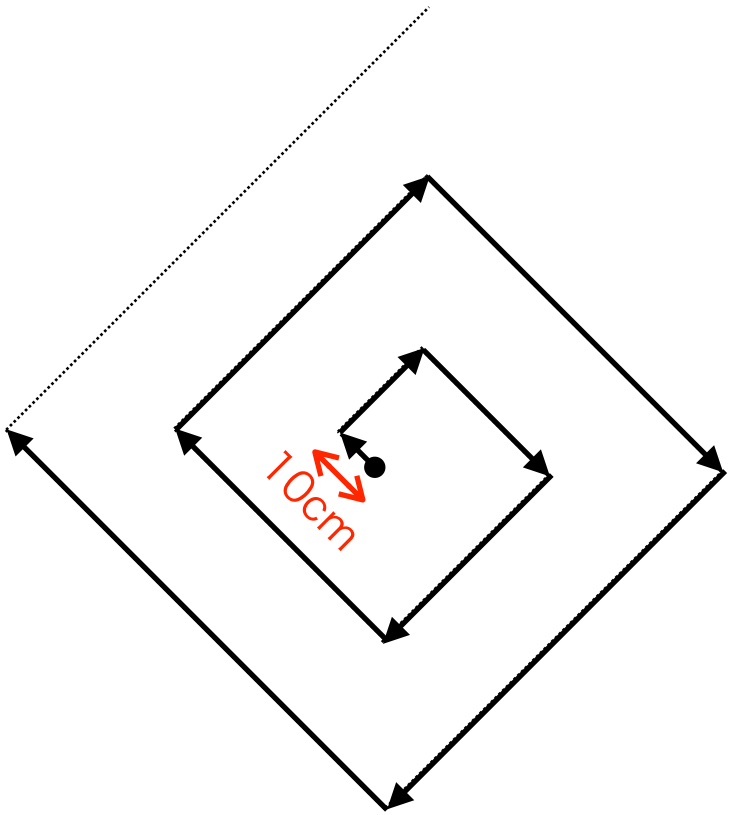
- 星形やスパイラルの軌跡を描くロボットの動きを実現してみよう！





実現したい動きの規則性

・ 繰り返し回数ごとのロボットの動きを表にしてみよう



回数	前進	右回転
1	10cm	90度
2	20cm	90度
3	30cm	90度
4	40cm	90度
5	50cm	90度
6	60cm	90度
7	70cm	90度
8	80cm	90度
9	90cm	90度

実現したい動きの規則性

- 規則性を数式で表現

回数	前進	右回転
1	10cm	90度
2	20cm	90度
3	30cm	90度
4	40cm	90度
5	50cm	90度
6	60cm	90度
7	70cm	90度
8	80cm	90度
9	90cm	90度

1

10

回数←1

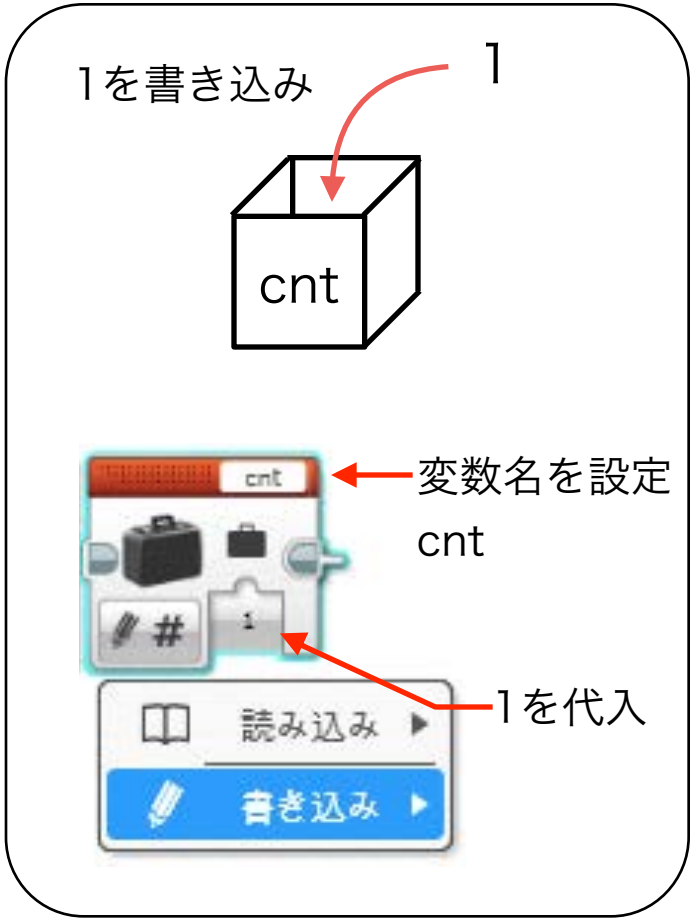
前進する距離 = 10cm×回数

回数←回数+1

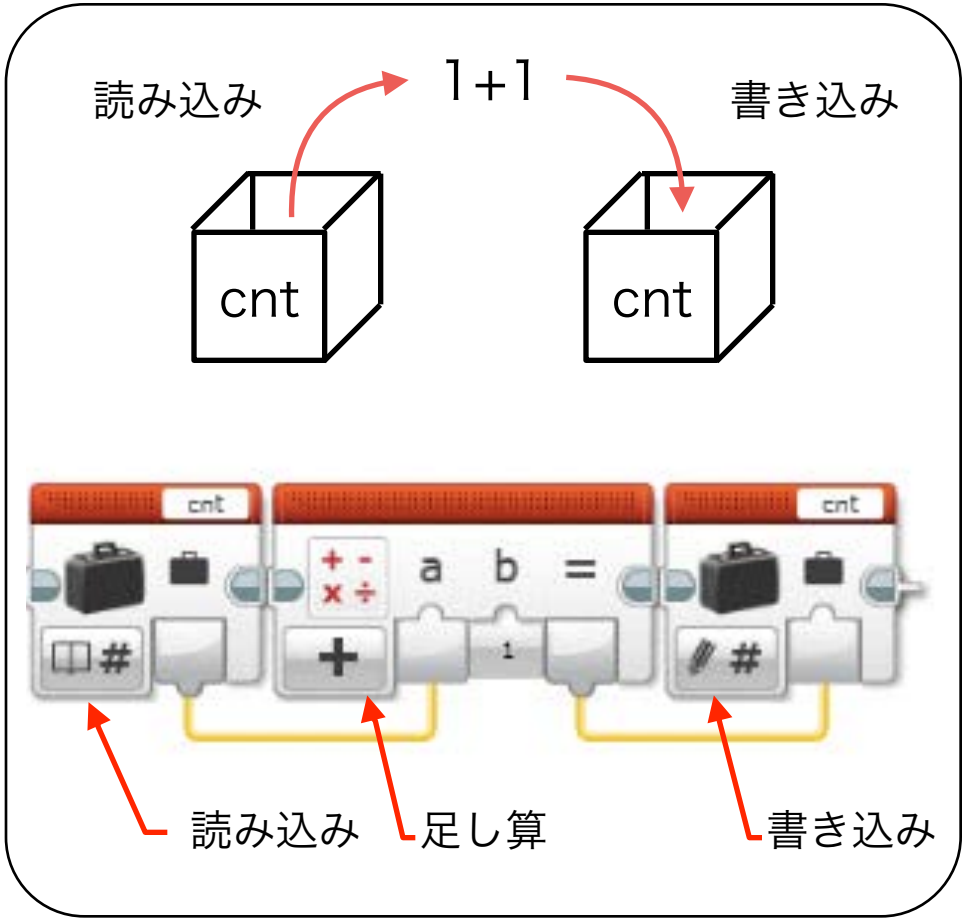
変数の導入

- 変数：値を代入できる箱（変数ブロック）

回数←1

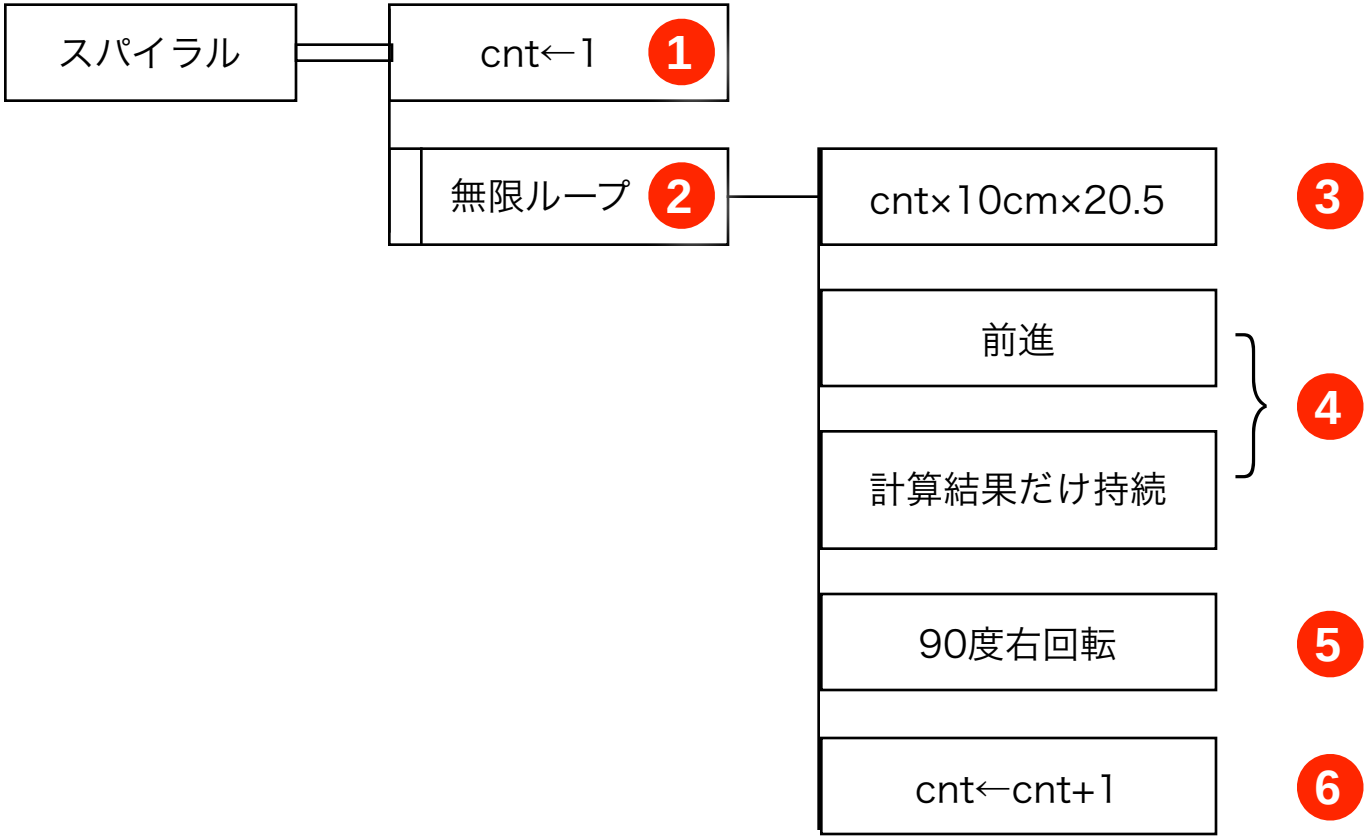


回数←回数+1



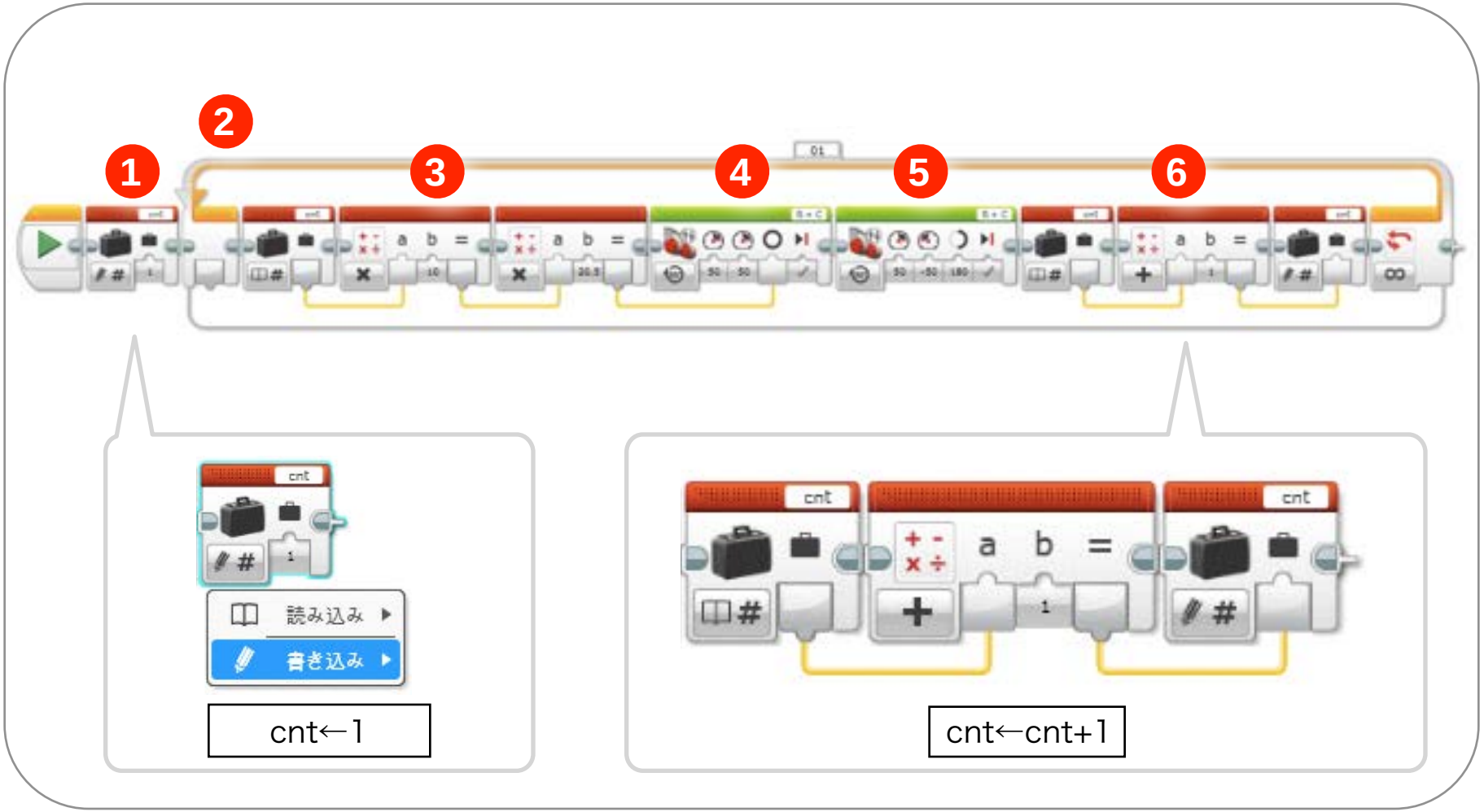
スパイラル状の軌跡のPAD

- 繰り返し回数cntと演算によりスパイラル軌跡を実現



スパイラル状の軌跡

・ 繰り返し回数cntと演算によりスパイラル軌跡を実現



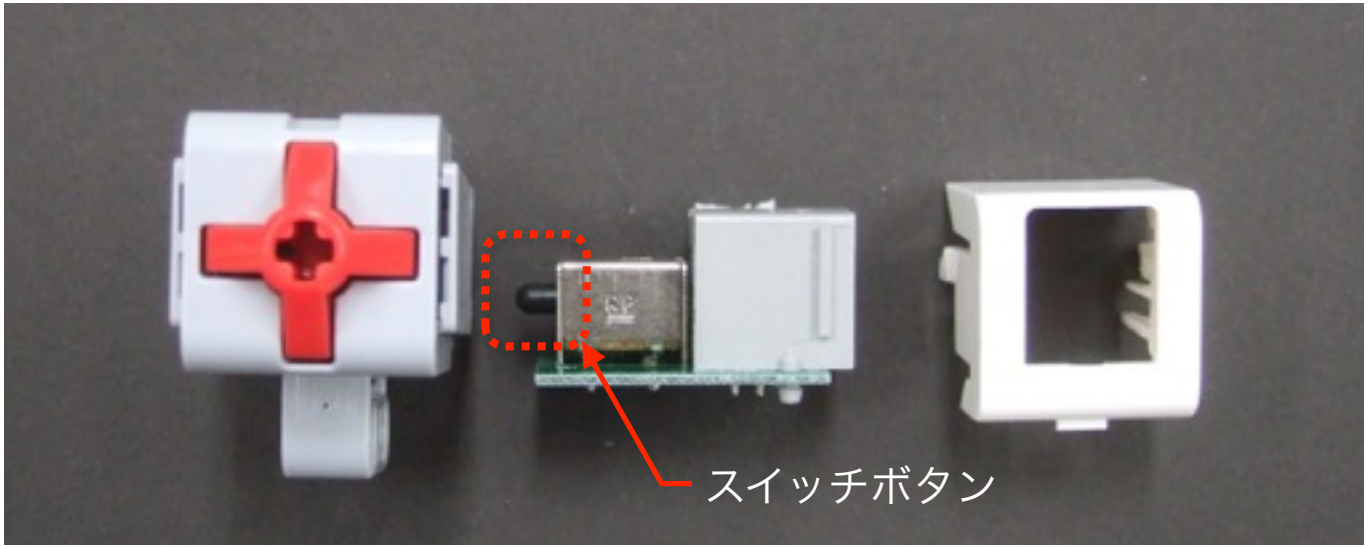
ロボットにおける要素技術

感じる、判断する、動くがそなわっている人工物

■障害物回避(タッチセンサ)
■障害物回避(超音波センサ)
■カラーセンサによるライントレース
■ジャイロセンサを用いたモータ制御

■障害物回避(タッチセンサ)

タッチセンサの原理

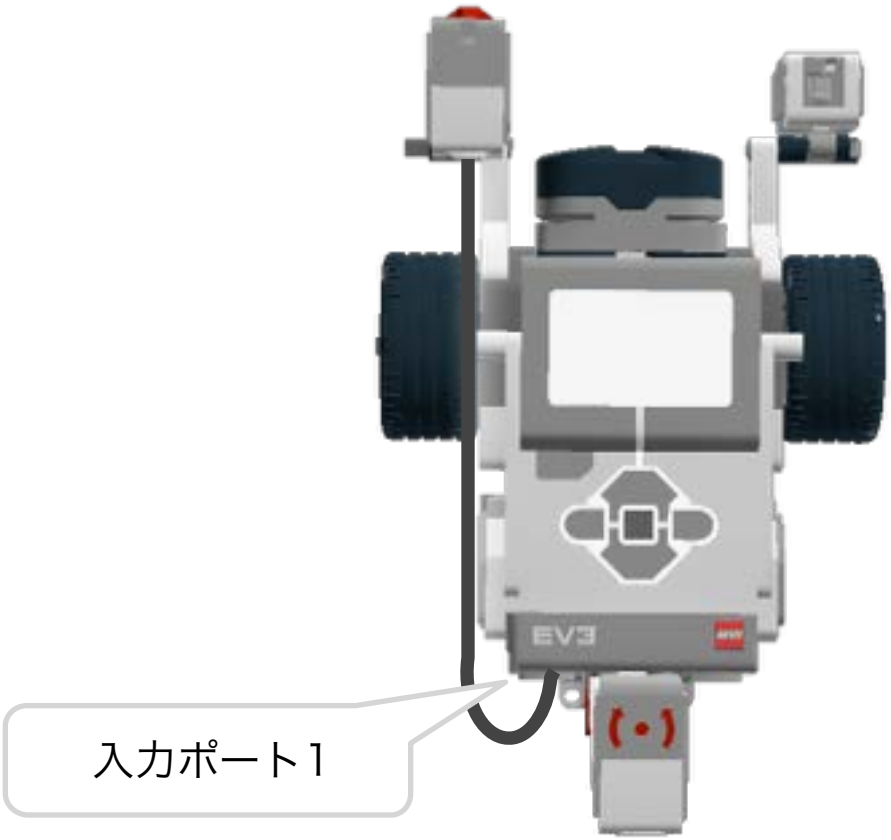


タッチセンサの測定原理

スイッチボタンの状態を、押されていないときは”0”、押されたら”1”,
離したときに”2”を出力する

タッチセンサの接続

・ EV3の入力ポート１にタッセンサを接続



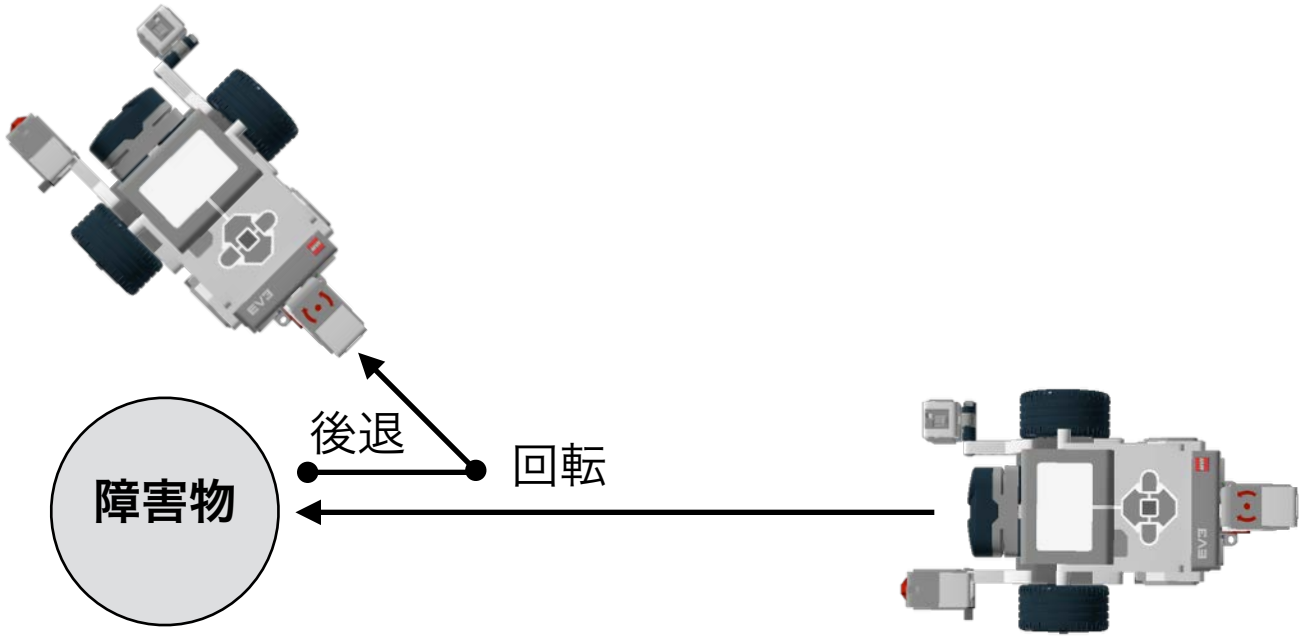
※EV3では入力ポートにどのセンサを接続しているか判定しているため、必ず入力ポート1である必要はない

タッチセンサによる障害物回避

1. 常にロボットを前進　→無限ループの利用

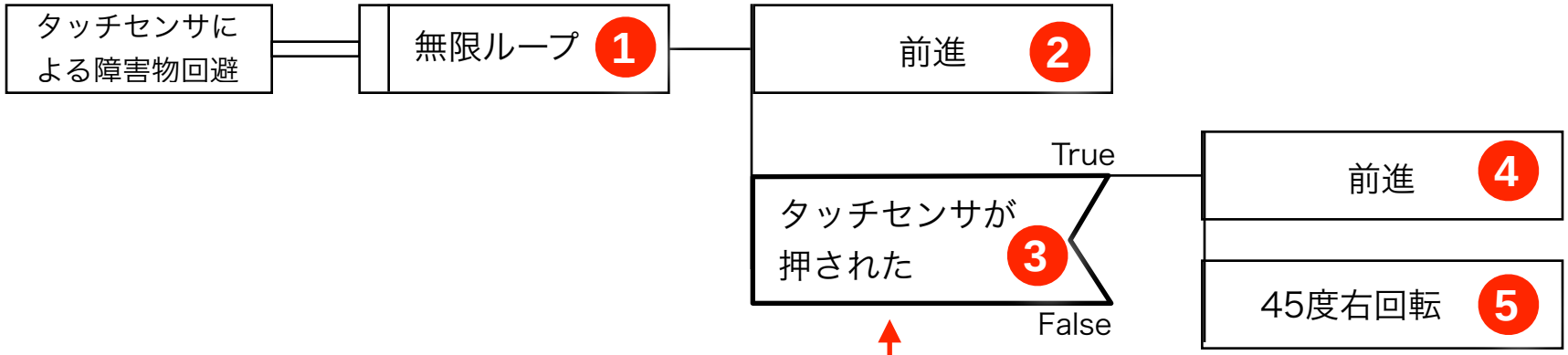
2. タッチセンサによる障害物の衝突を判定　→条件分

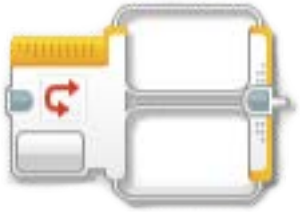
3. 衝突あり：一定時間後退し右回転．その後1. に戻る
   衝突なし：1.に戻る



タッチセンサによる障害物回避のPAD

- タッチセンサによる条件分岐





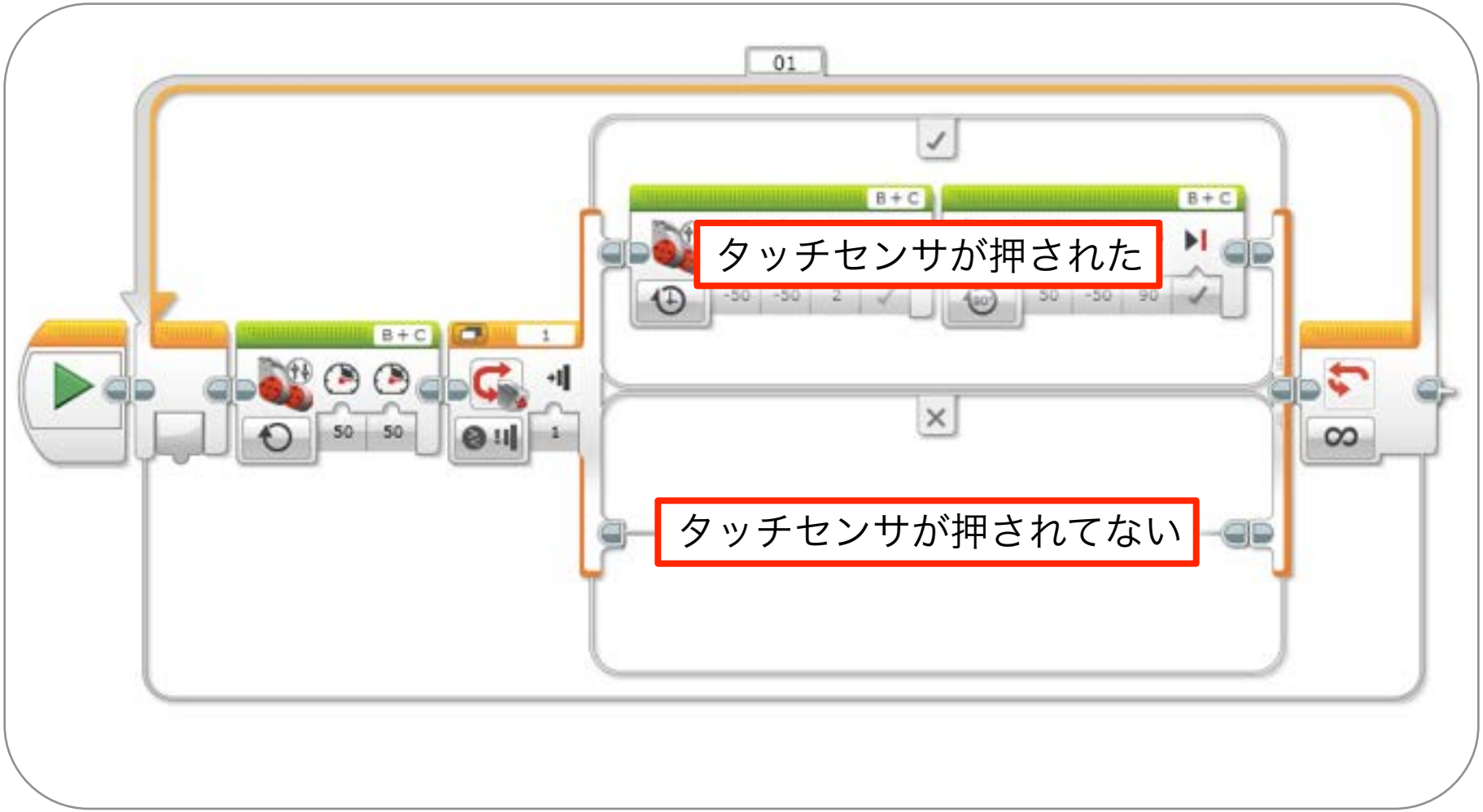
条件分岐：スイッチブロック

PADの構成部品：

処理：　選択：　反復：

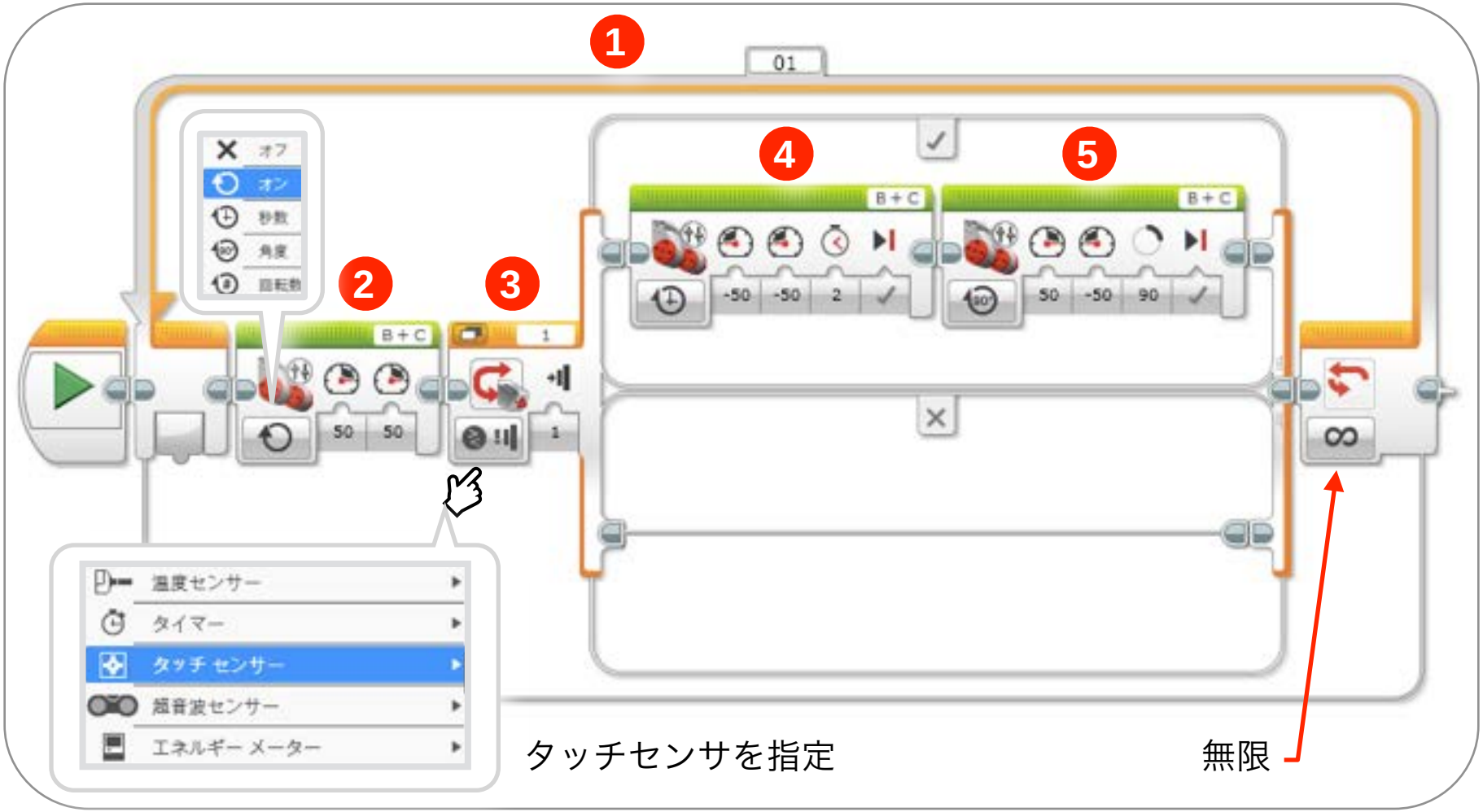
タッチセンサによる障害物回避

- タッチセンサによる条件分岐



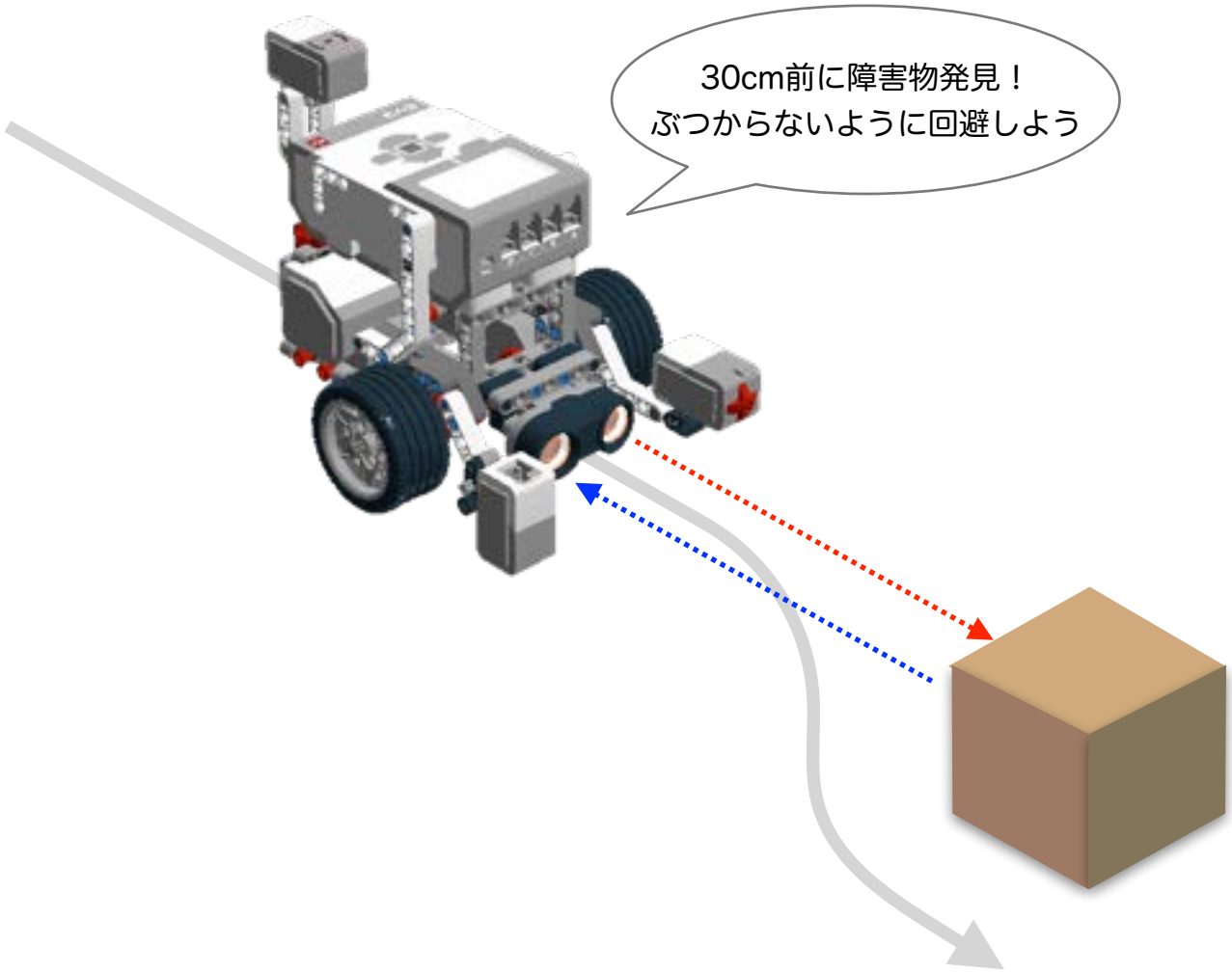
タッチセンサによる障害物回避

- タッチセンサによる条件分岐



■障害物回避(超音波センサ)

超音波センサによる障害物回避



超音波センサ

超音波センサの測定原理

超音波を発信し、対象物で反射した超音波を受信し、この音波の発信から受信までの時間を計測することで対象物までの距離を計測

超音波センサの接続

・ EV3の入力ポート4に超音波センサを接続

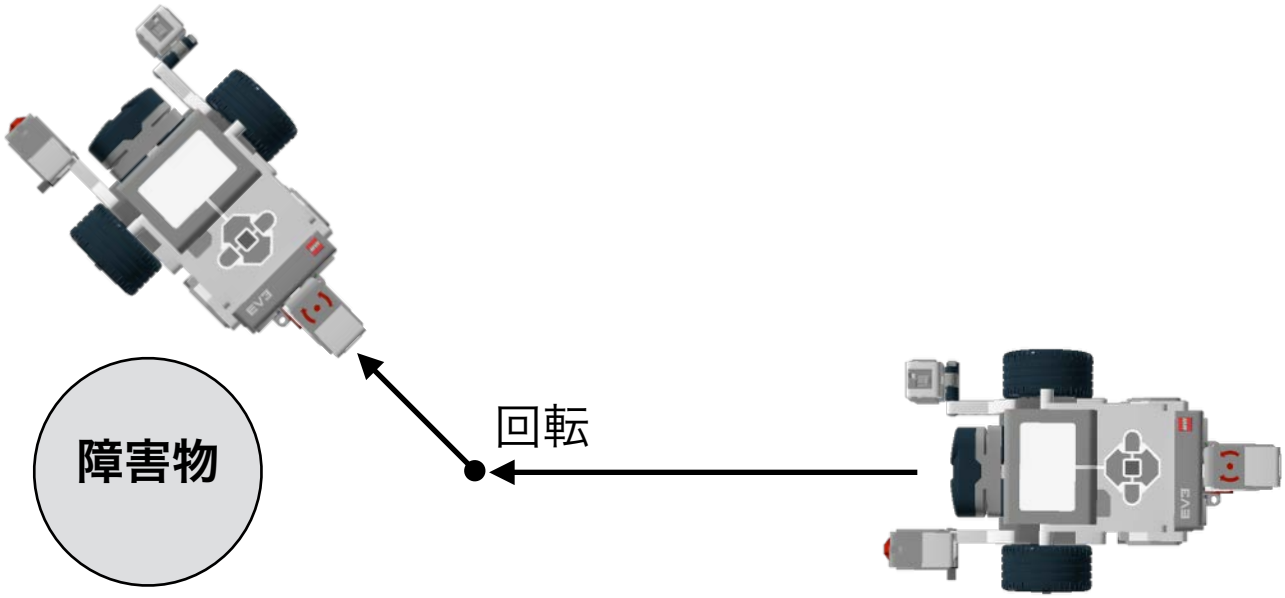
※EV3では入力ポートにどのセンサを接続しているか判定しているため、必ず入力ポート4である必要はない

超音波センサによる障害物回避

1. 常にロボットを前進　→無限ループの利用

2. 障害物までの距離を計測

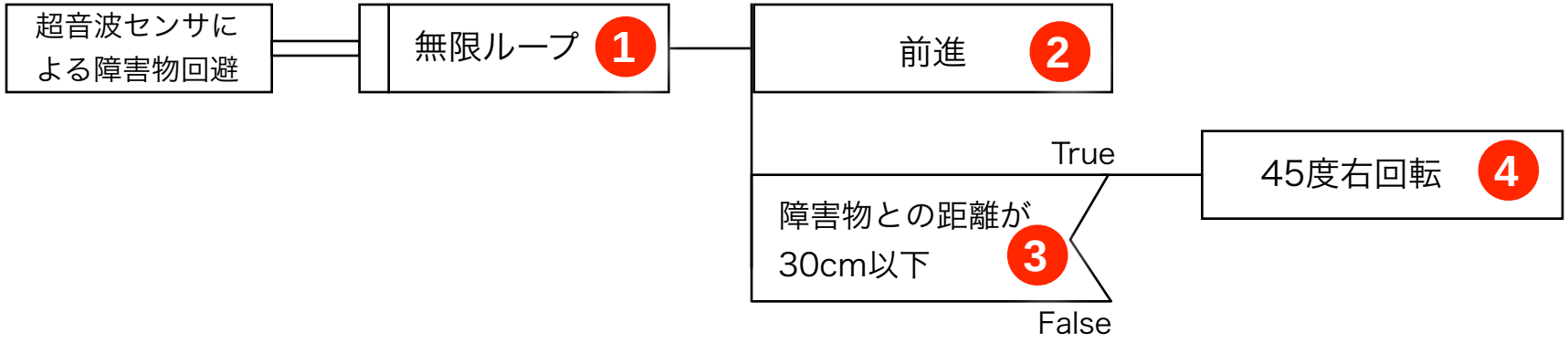
3. 30cm以下(障害物あり)：右回転．その後1. に戻る
   30cm以上(障害物なし)：1.に戻る



→後退する動作を必要としない

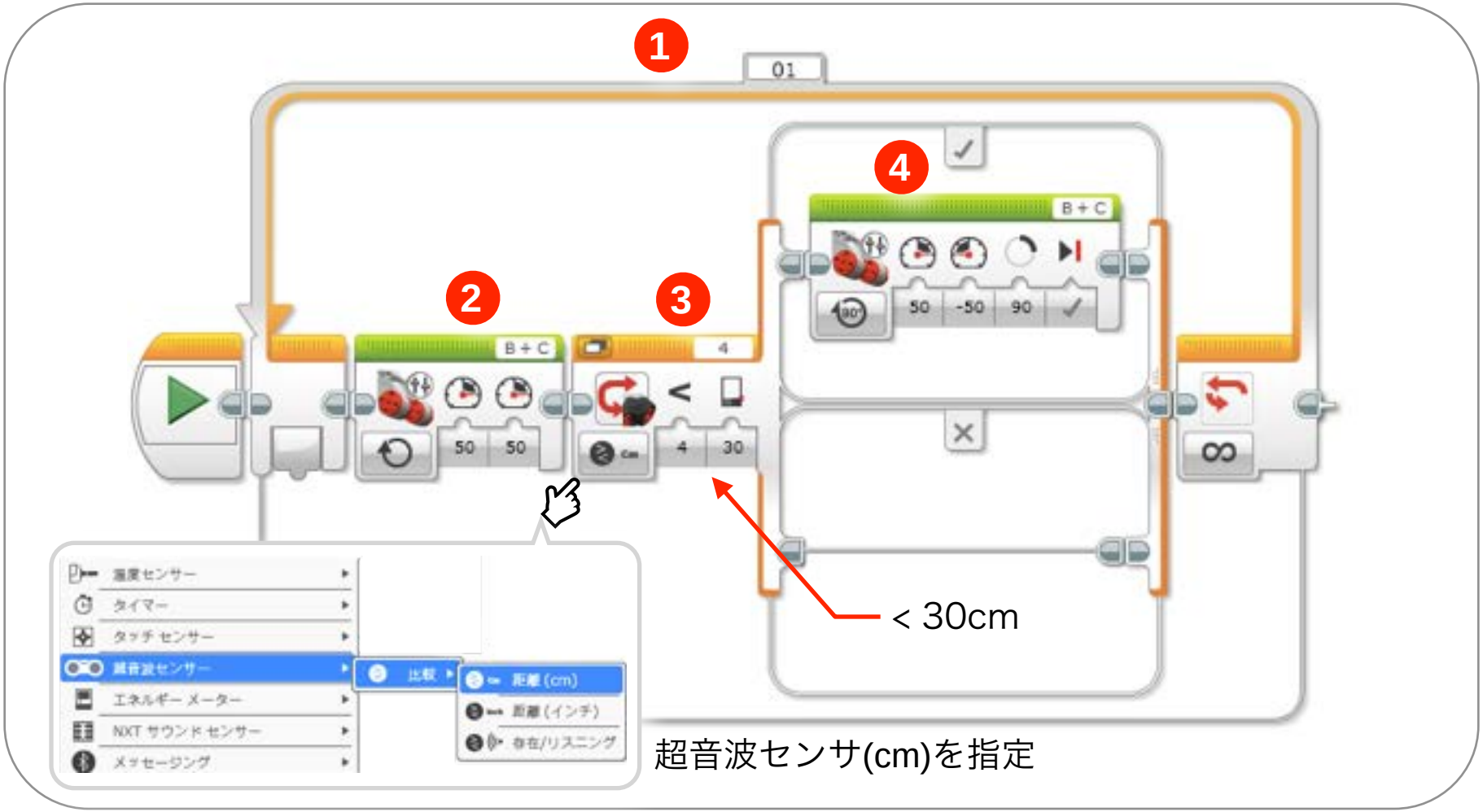
超音波センサによる障害物回避のPAD

・超音波センサの値により条件分岐して障害物回避



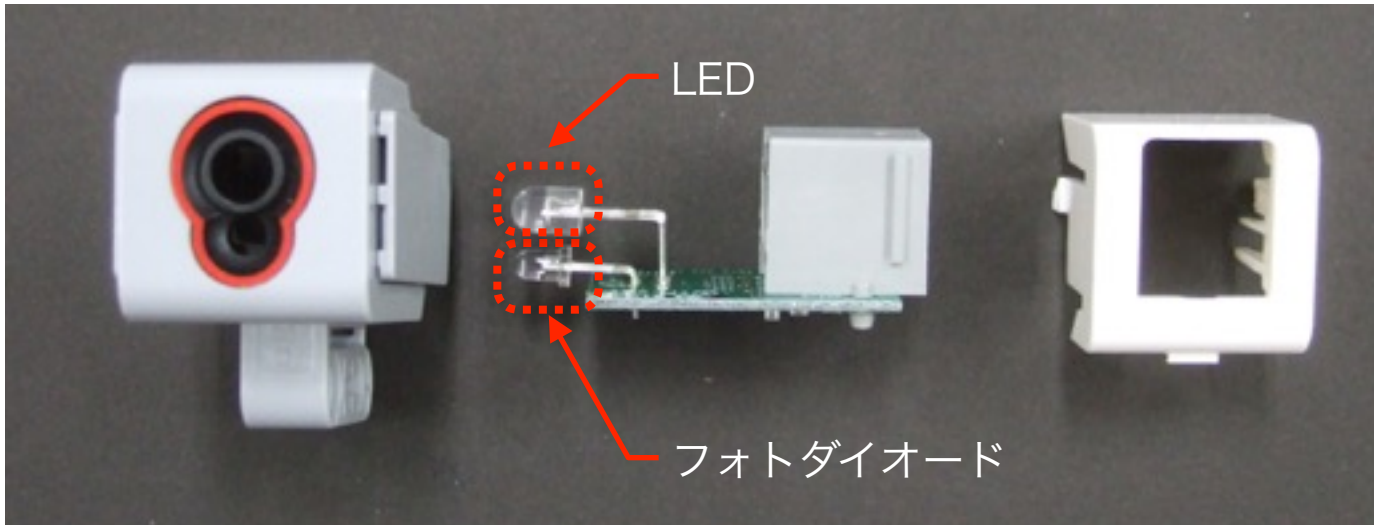
超音波センサによる障害物回避

・超音波センサの値により条件分岐して障害物回避



■カラーセンサによるライントレース

カラーセンサ

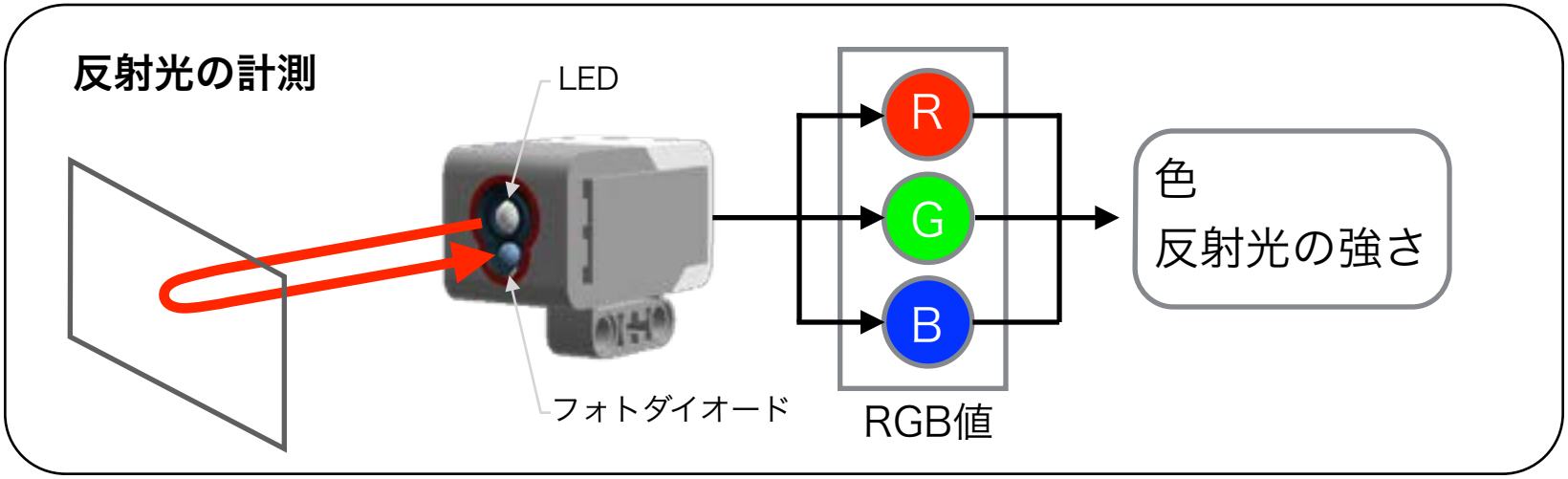


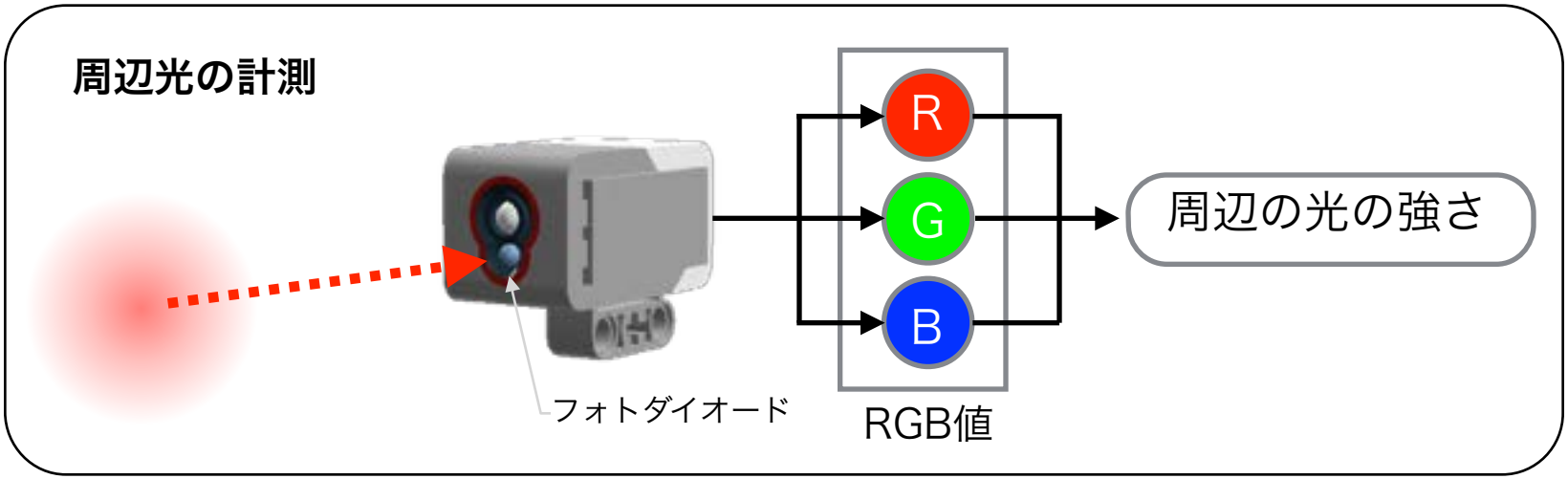
カラーセンサの測定原理

LEDにより赤、緑、青の光を照射して、それぞれの反射量(R, G, B成分)をフォトダイオードで計測して，カラーに変換

カラーセンサ

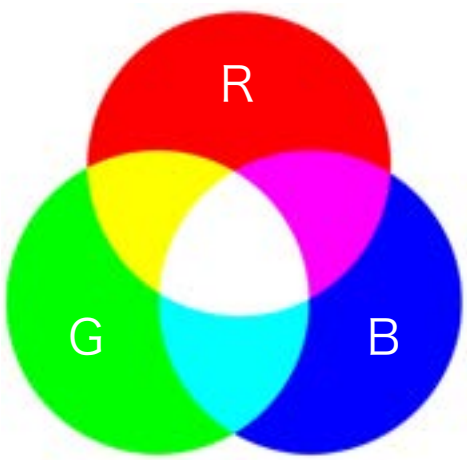
- 二種類のしくみで色や光の強さを計測





光の三原色とカラー（加法混色）

- 赤(R)，緑(G)，青(B)の3色を組み合わせて色彩を表現

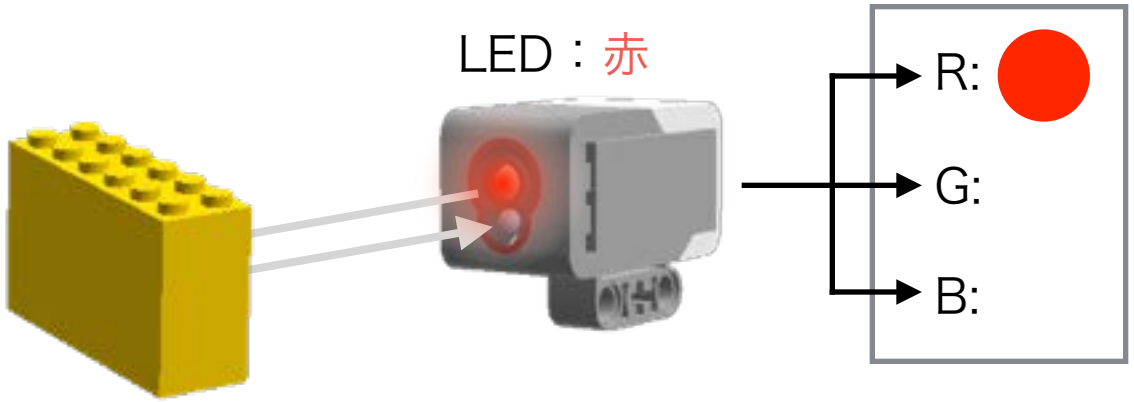


加法混色

色番号	R	G	B
1: 黒	0	0	0
2: 青	0	0	大
3: 緑	大	0	0
4: 黄	大	大	0
5: 赤	大	0	0
6: 白	大	大	大

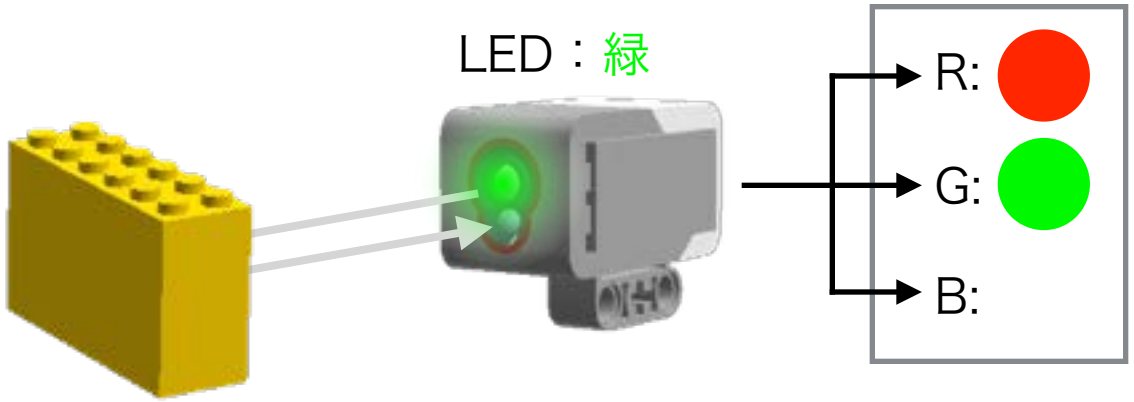
カラーセンサの原理

- LEDを切り替えて発光し、そのタイミングでの反射光を計測して、カラーを計測



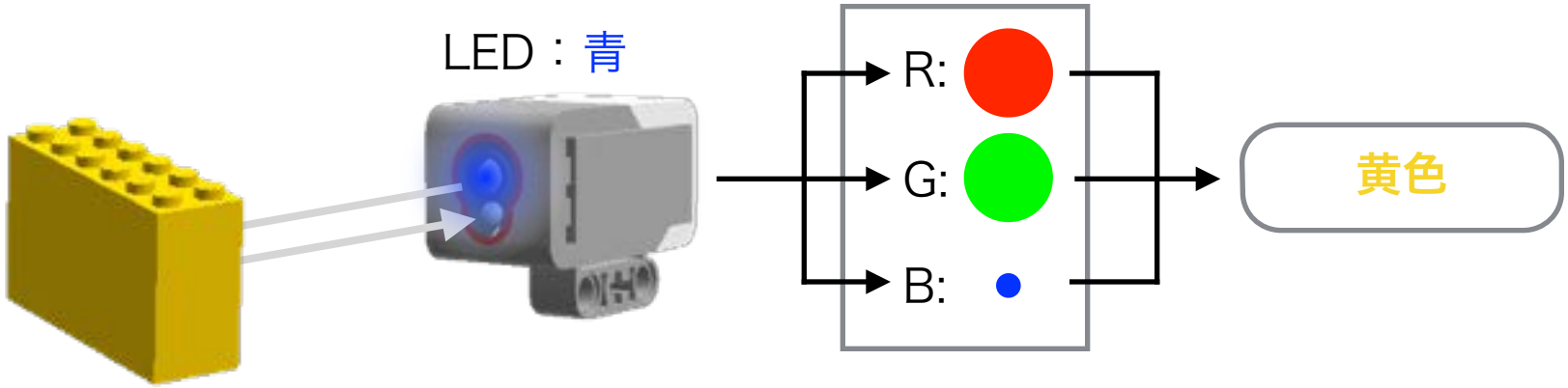
カラーセンサの原理

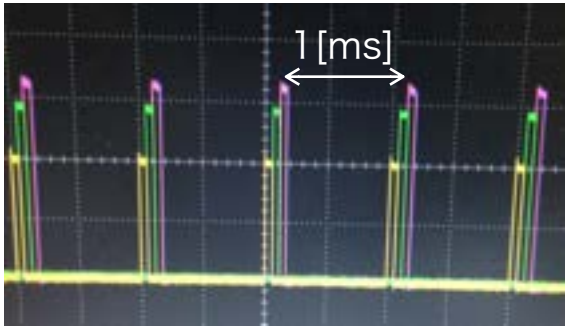
- LEDを切り替えて発光し、そのタイミングでの反射光を計測して、カラーを計測



カラーセンサの原理

- LEDを切り替えて発光し、そのタイミングでの反射光を計測して、カラーを計測

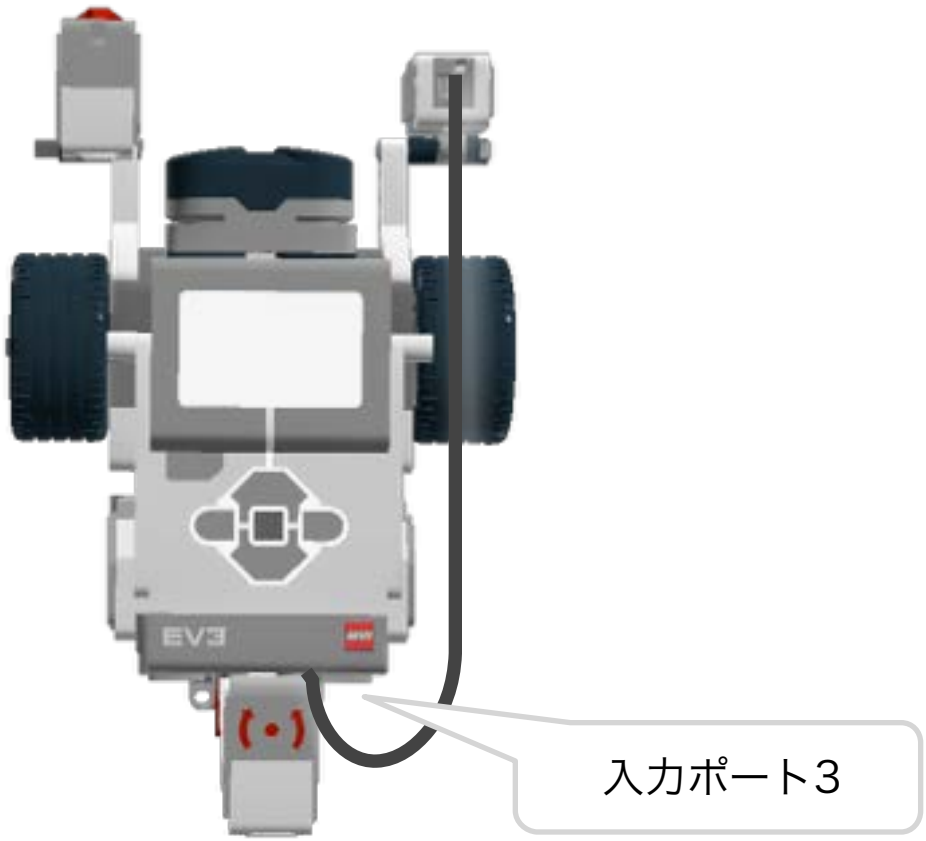




オシロスコープで計測してみたところ：
1秒間に1,000回、R,G,Bと切り替えて発光し、
同期してフォトダイオードで反射光量を計測

カラーセンサの接続

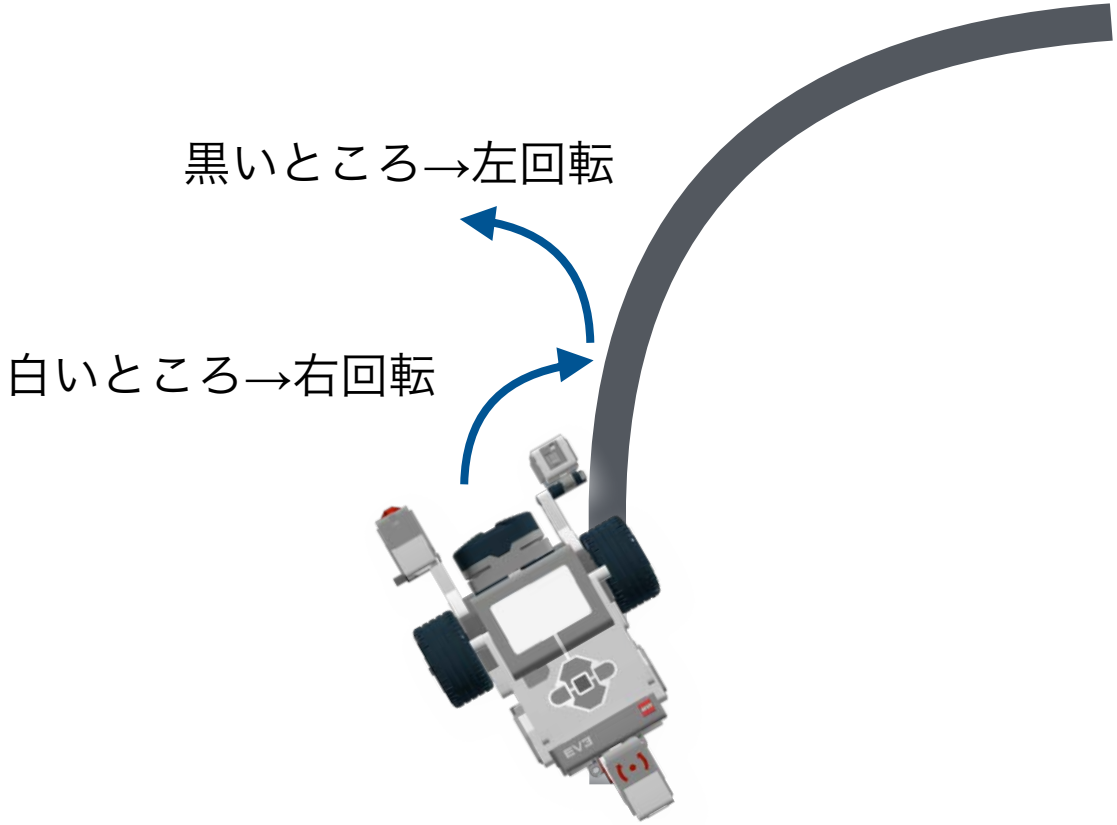
・ EV3の入力ポート3にカラーセンサを接続



※EV3では入力ポートにどのセンサを接続しているか判定しているため、必ず入力ポート3である必要はない

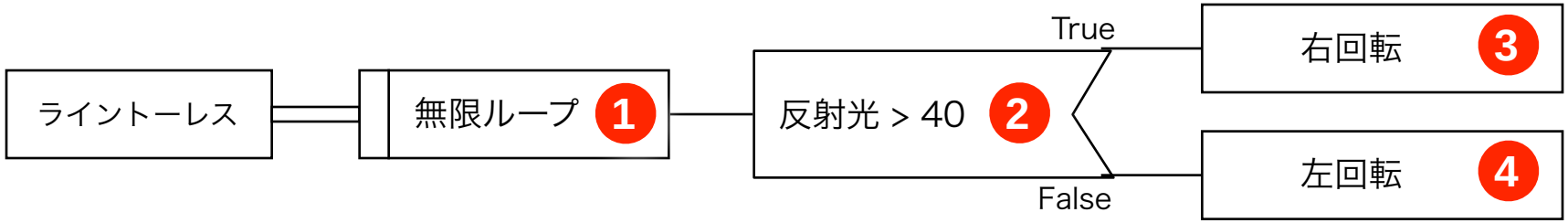
ライントレースの考え方

- カラーセンサの”反射光の強さ”を利用して白と黒に判別
- 白(明るい)ところでは右回転，黒(暗い)ところでは左回転

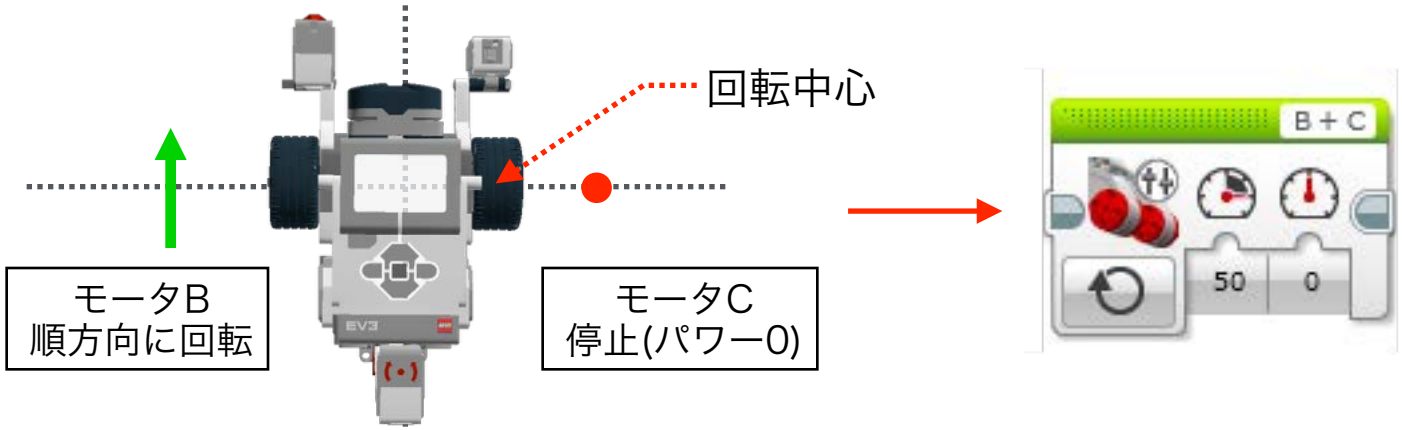


ライントレースのPAD

・ 白(明るい)ところでは右回転，黒(暗い)ところでは左回転

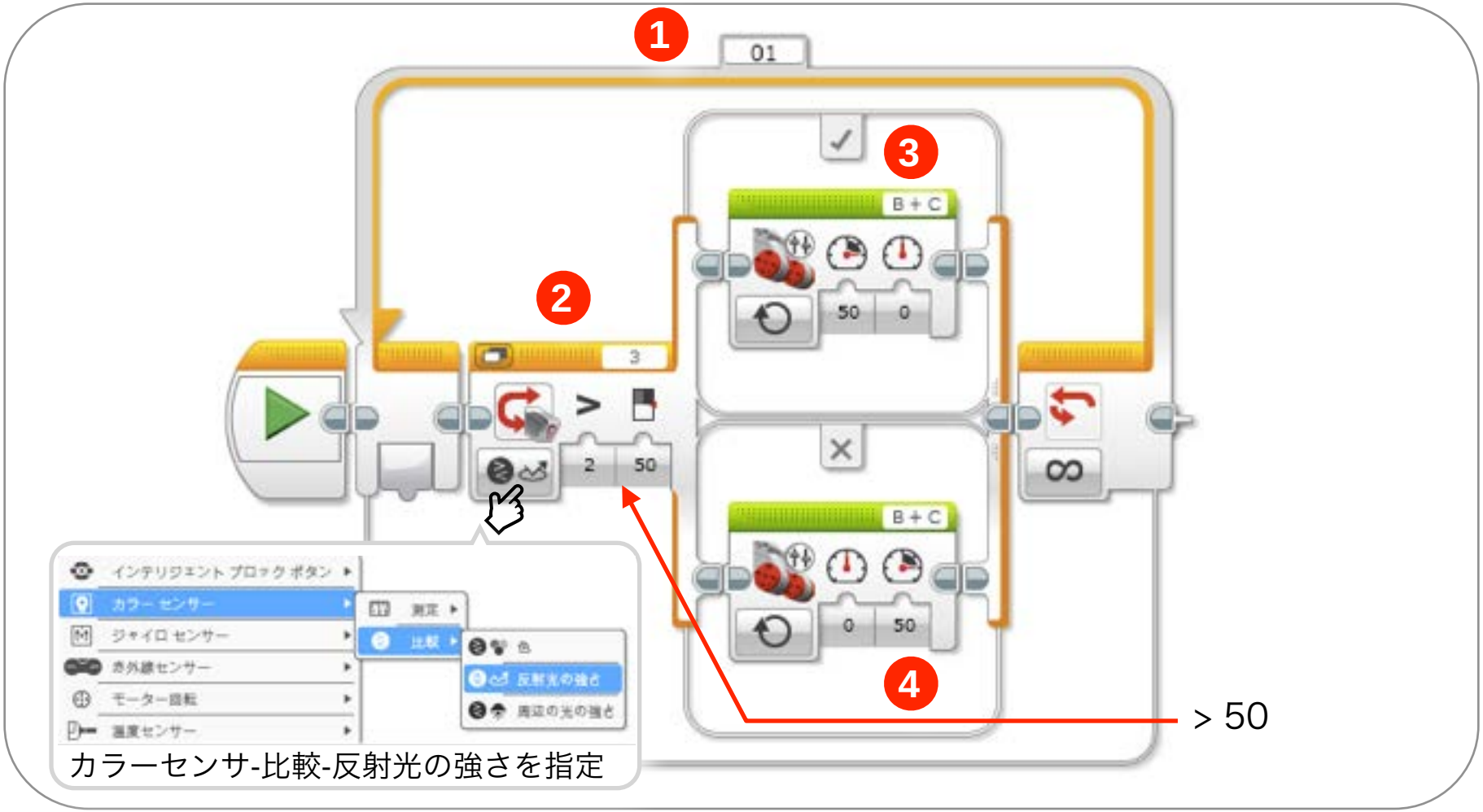


右回転しながら少し前に進むには：



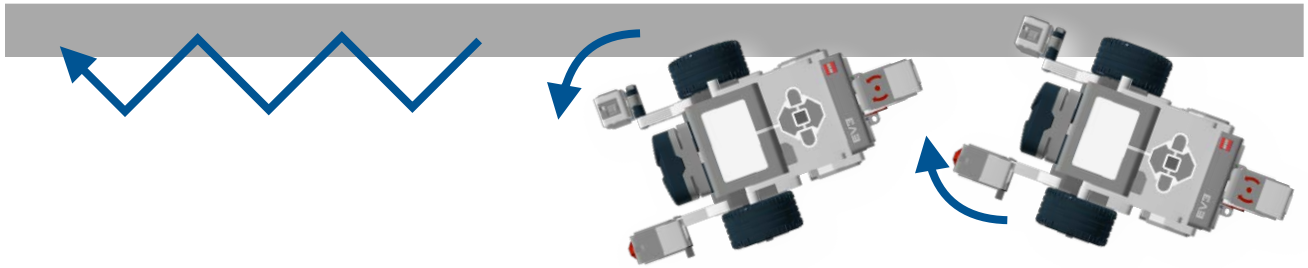
ライントレース

・ 白(明るい)ところでは右回転，黒(暗い)ところでは左回転

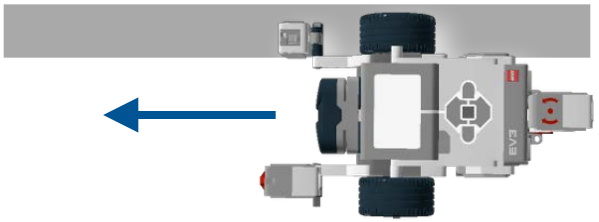


ライントレースの改良

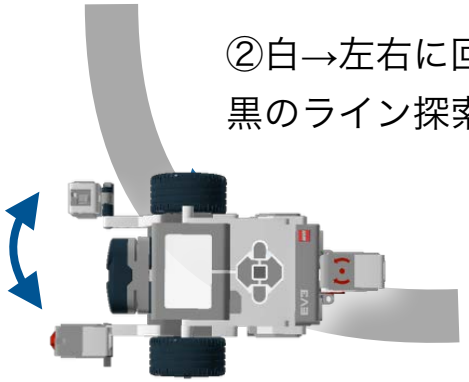
ジグザグ走行なので遅い！



①黒のライン上→前進

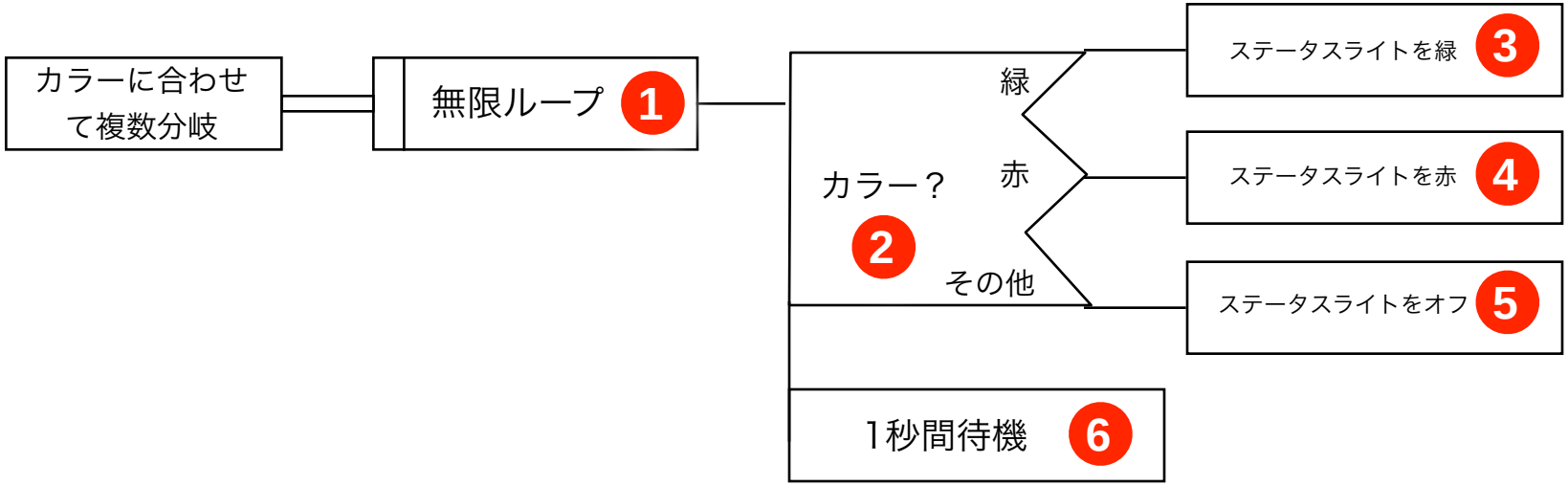


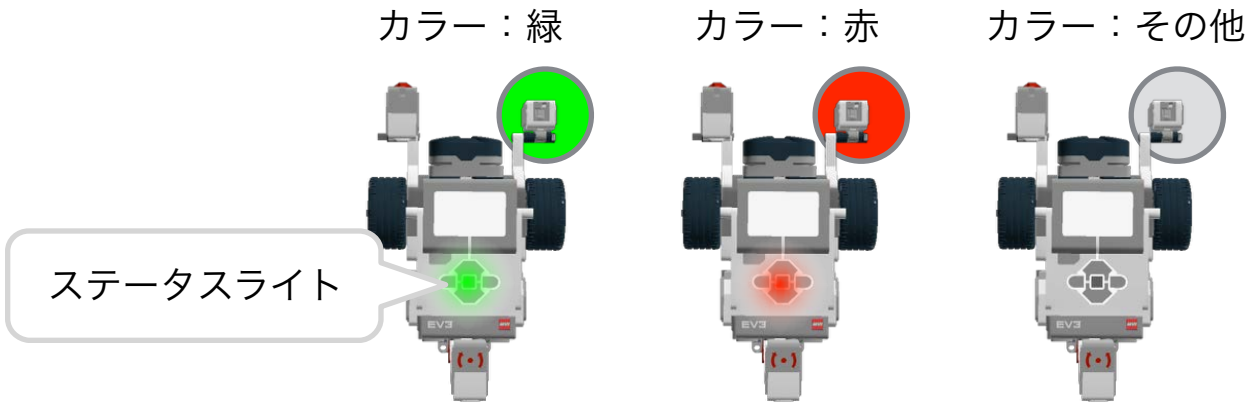
②白→左右に回転して
黒のライン探索



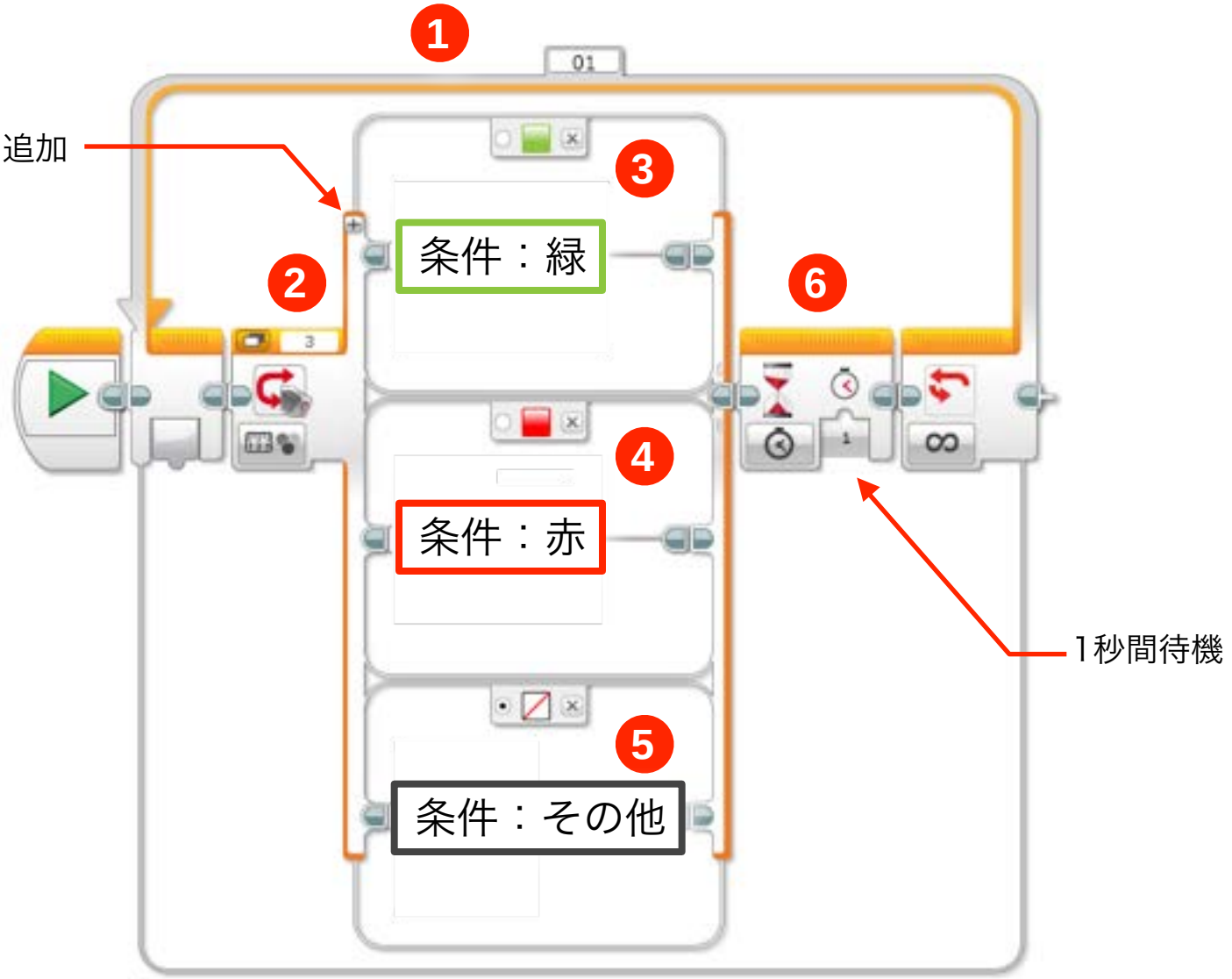
カラーの利用(PAD)

・ カラーに合わせて複数に分岐

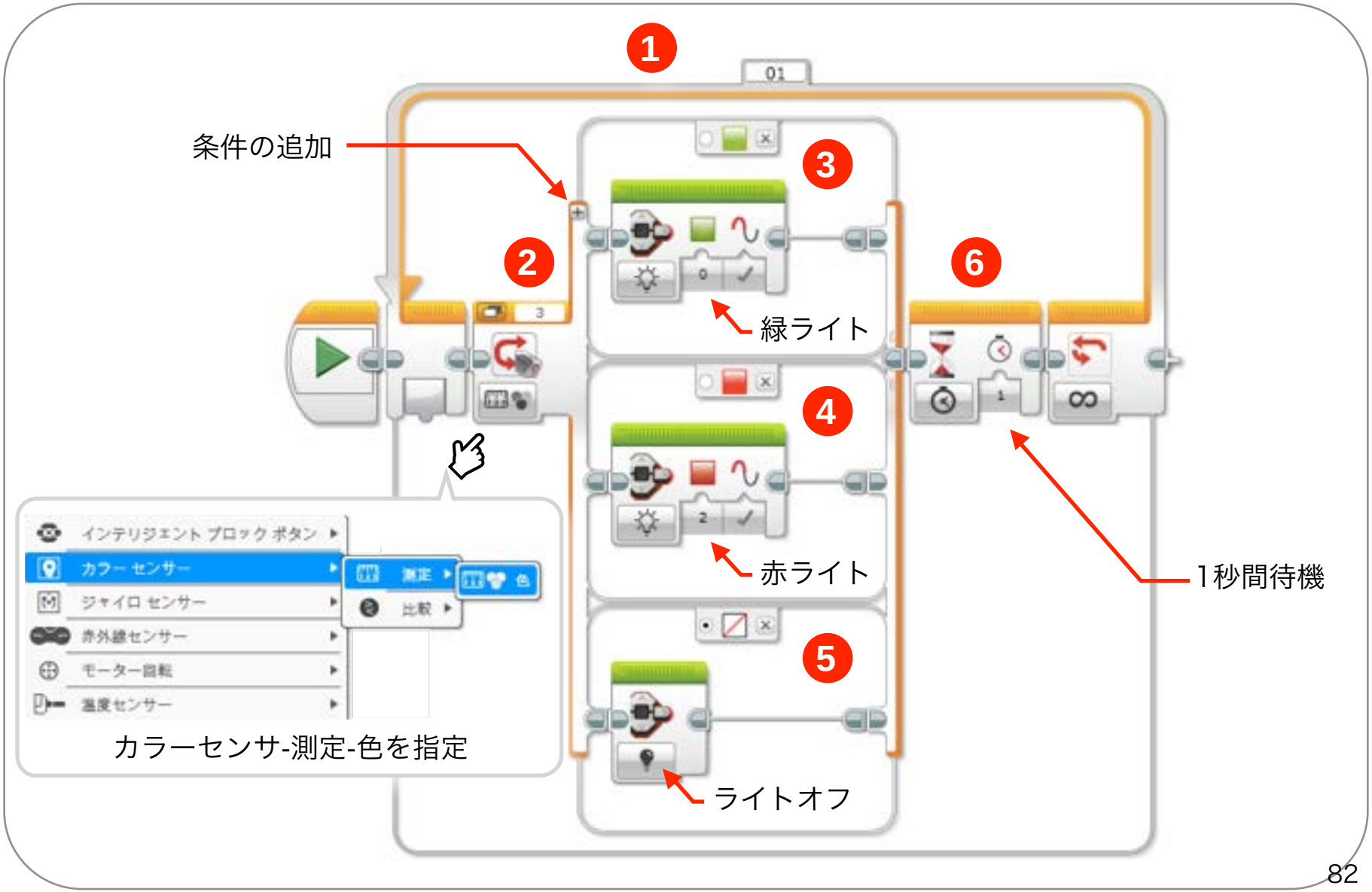




カラーの利用

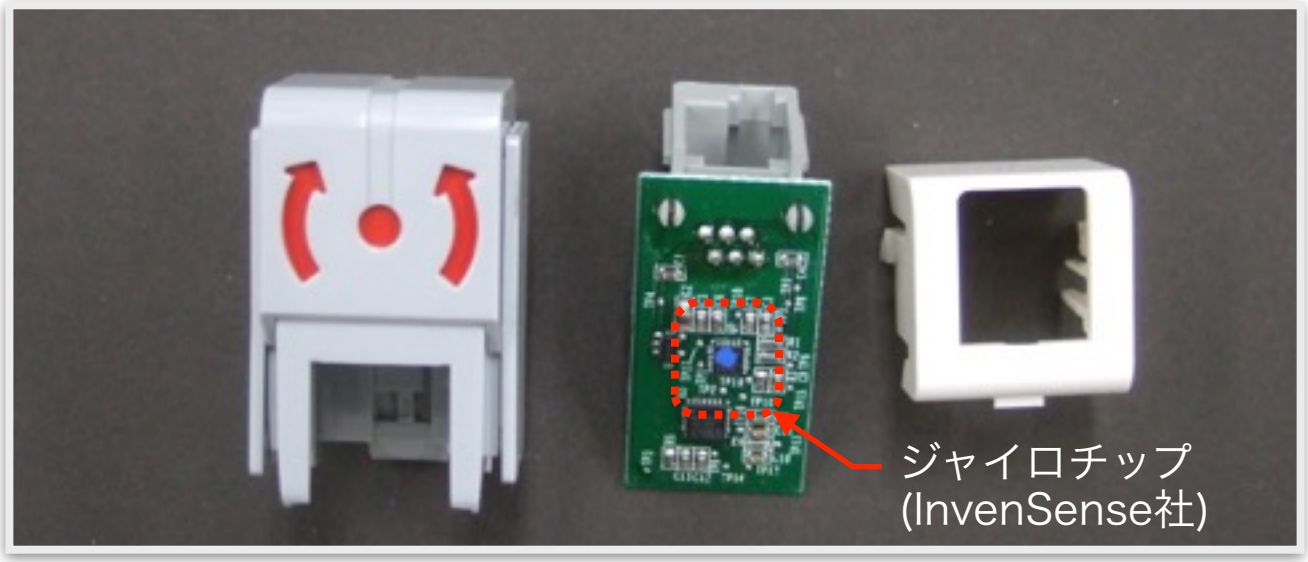


カラーの利用


1
2
3
4
5
6
01
条件の追加
緑ライト
赤ライト
ライトオフ
1秒間待機
インテリジェント ブロック ボタン
カラー センサー
ジャイロ センサー
赤外線センサー
モーター回転
温度センサー
測定
比較
色
カラーセンサ-測定-色を指定

■ジャイロセンサを用いたモータ制御

ジャイロセンサの原理

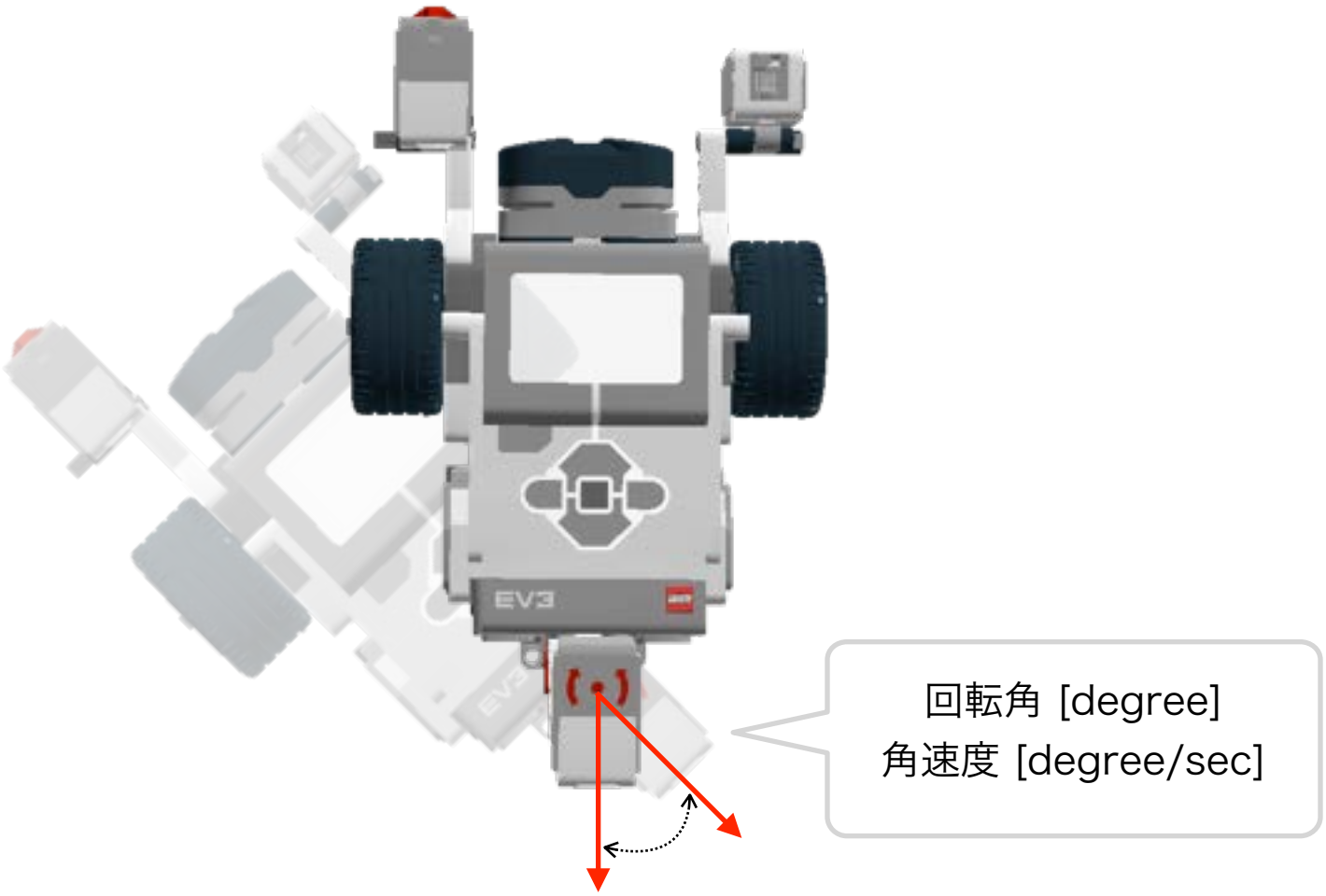


ジャイロセンサの測定原理

ジャイロセンサは角速度センサとも呼ばれ、単位時間あたりの回転角である角速度[degree/s]を計測

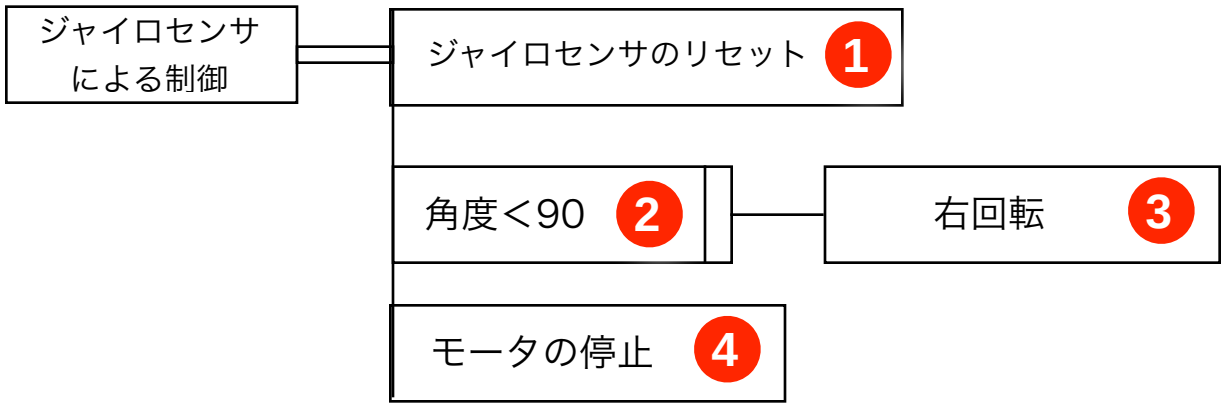
ジャイロセンサ

- EV3のジャイロセンサは一軸の回転角/角速度を計測



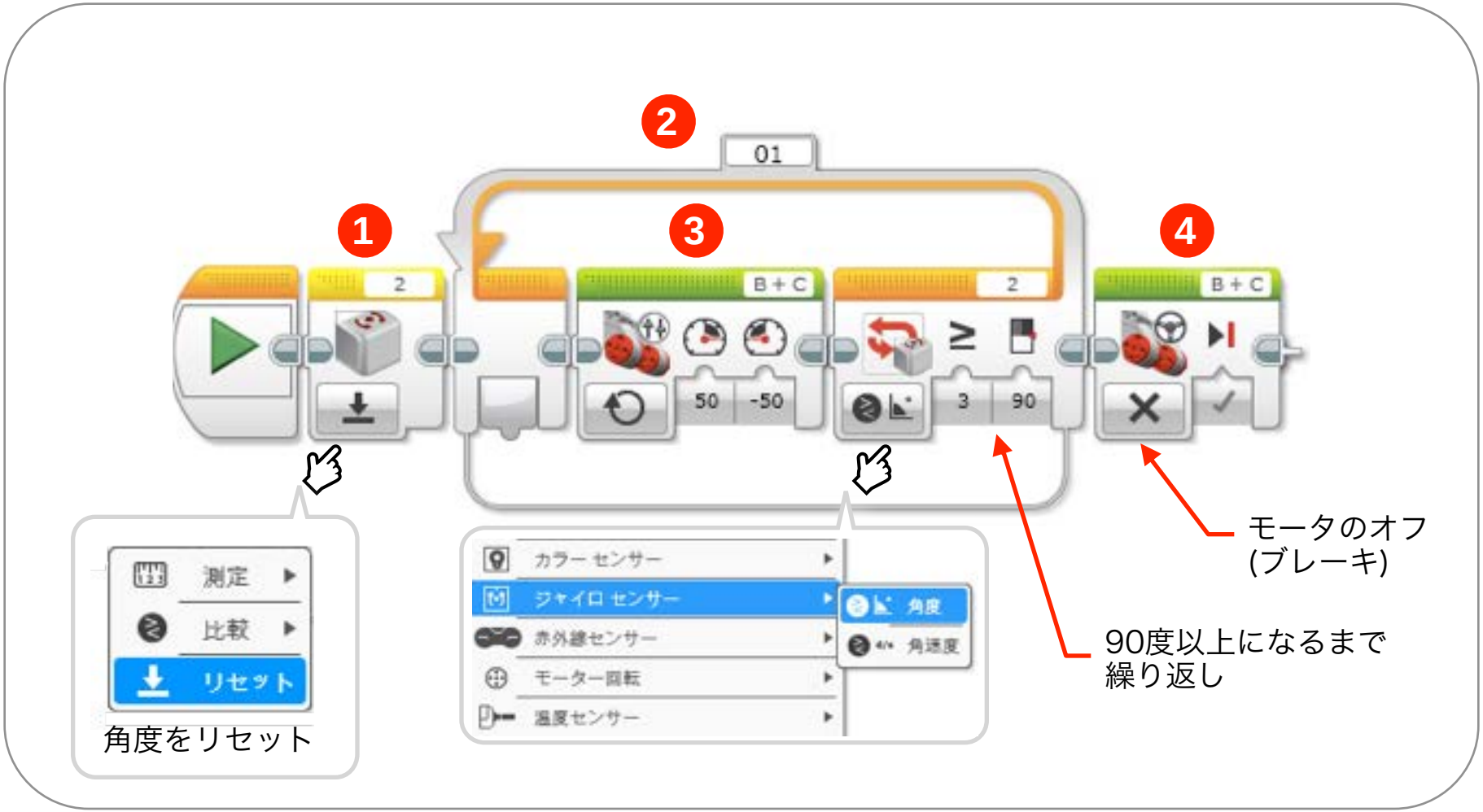
ジャイロセンサによる旋回角度制御(PAD)

- ジャイロセンサの角度が90度以上になるまで右回転



ジャイロセンサによる旋回角度制御

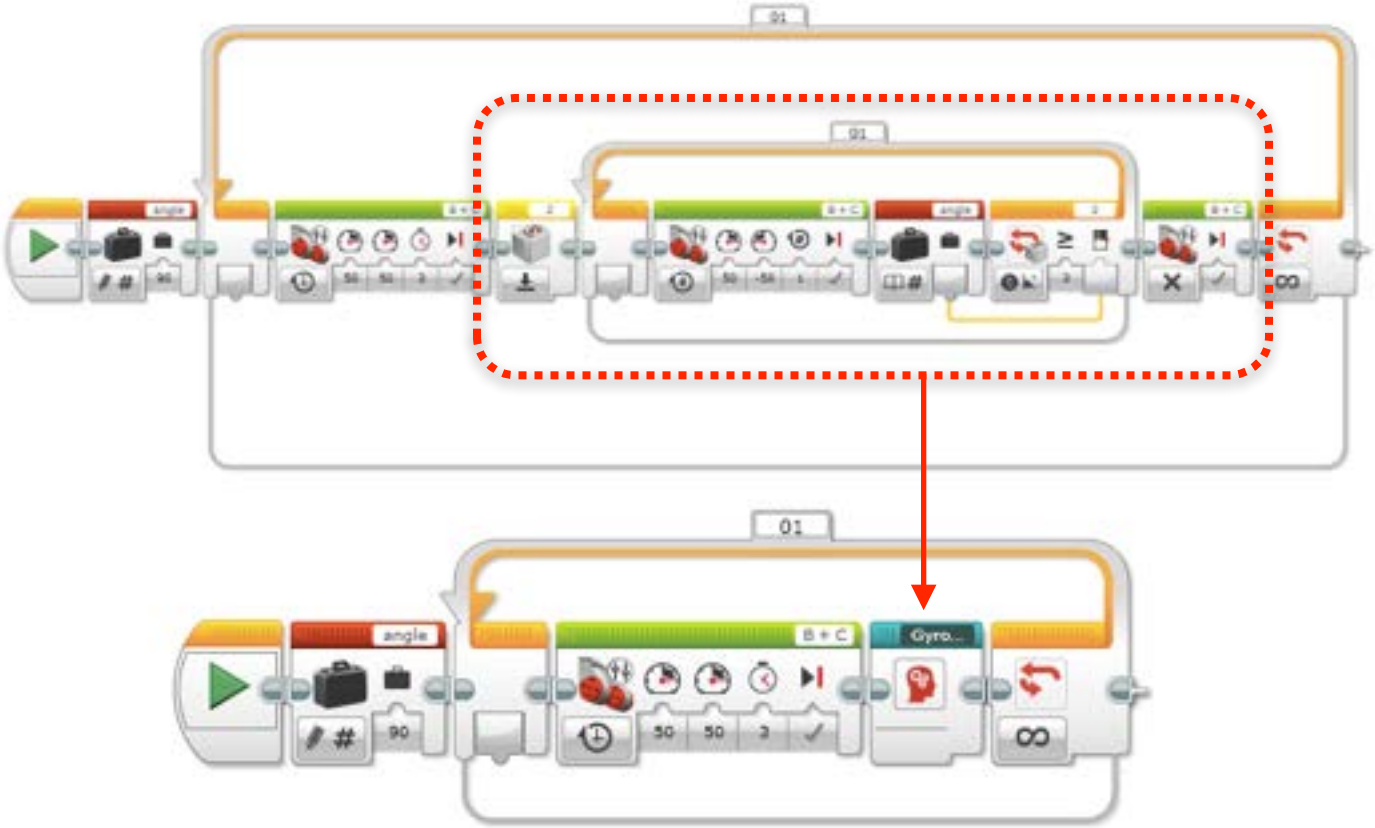
・ ジャイロセンサの角度が90度以上になるまで右回転



■マイブロック

マイブロック化

①マイブロック化したい複数のブロックをドラッグにより選択
②”ツール”⇒”マイブロックビルダー”を起動
③マイブロックの名前やアイコンを決定



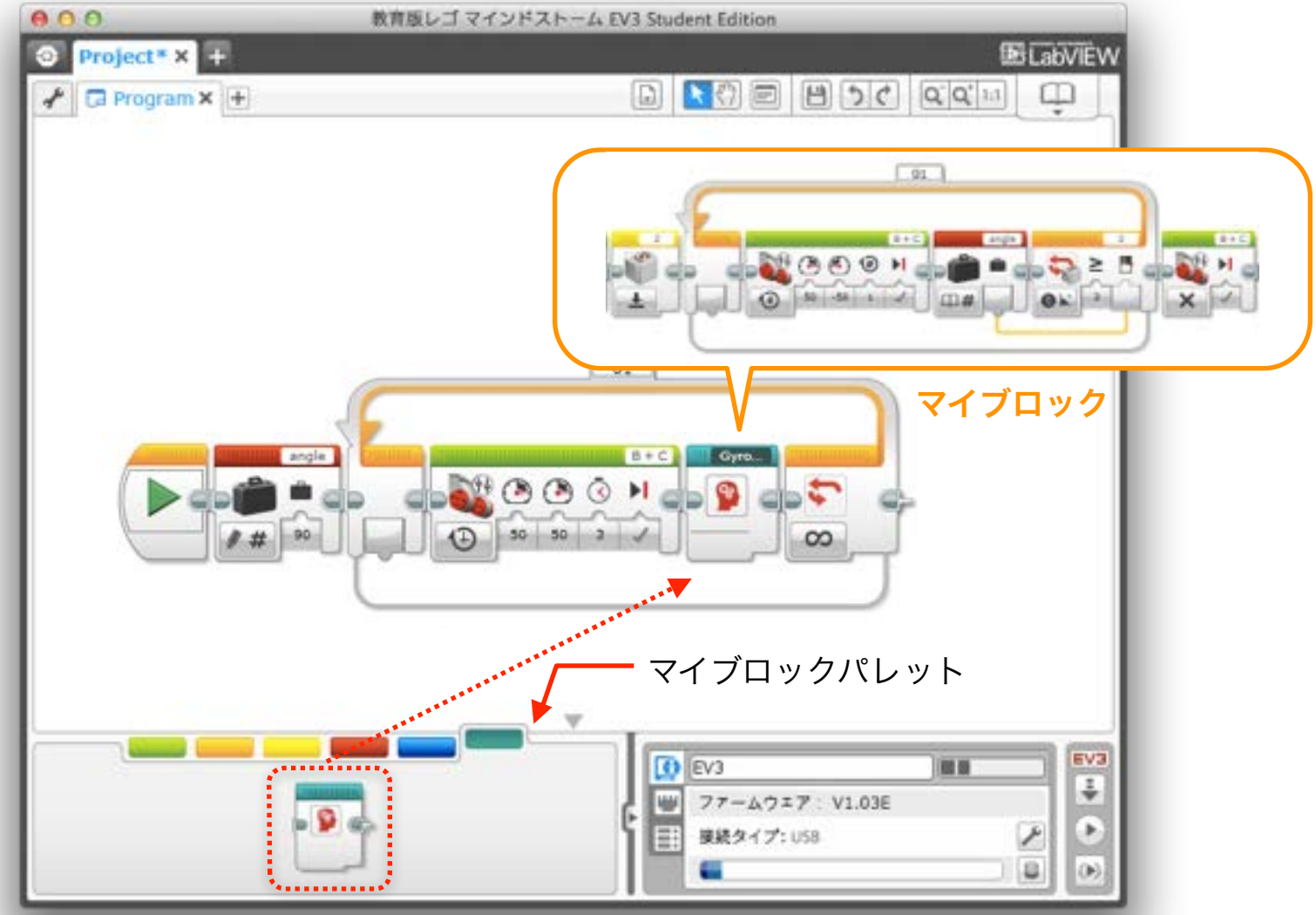
マイブロックの設定

マイブロックの名前

アイコンの選択

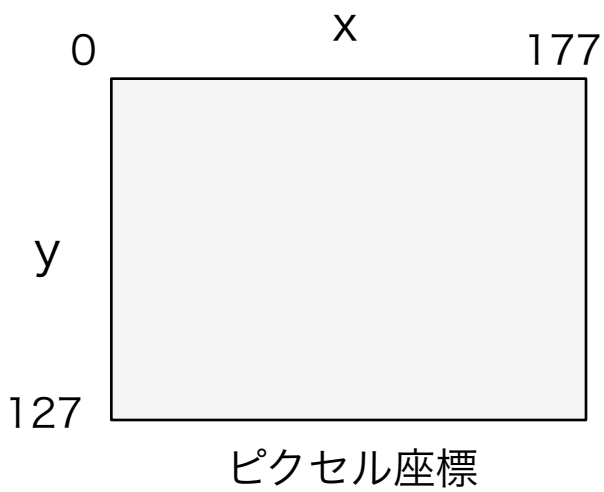
内容を記入

マイブロックの利用

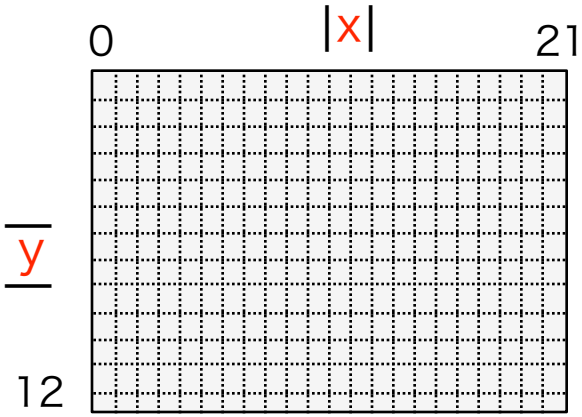

教育版レゴ マインドストーム EV3 Student Edition
Project*
Program
LabVIEW
マイブロック
マイブロックパレット
EV3
ファームウェア：V1.03E
接続タイプ: USB

■ディスプレイへの表示

ディスプレイの座標

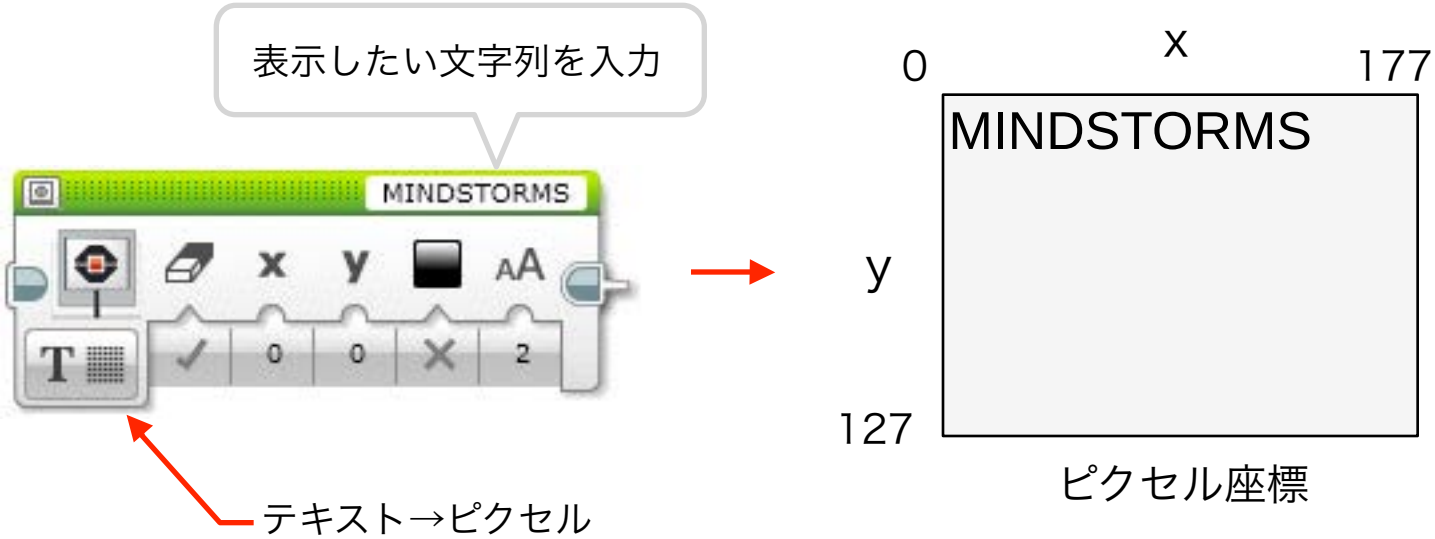


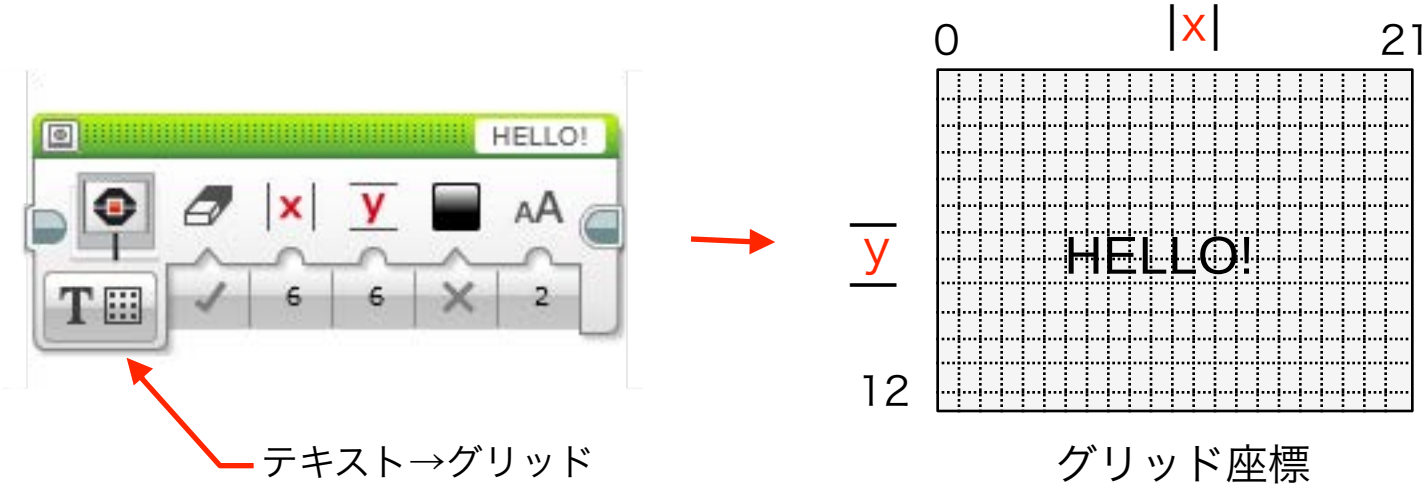
ピクセル座標



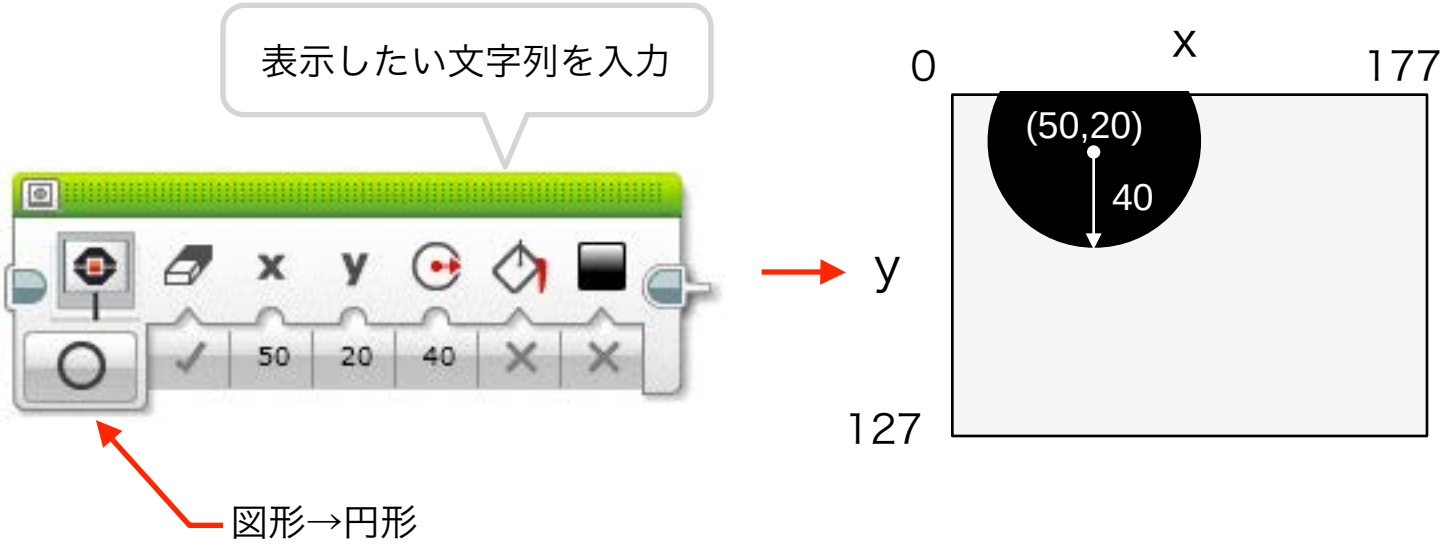
グリッド座標

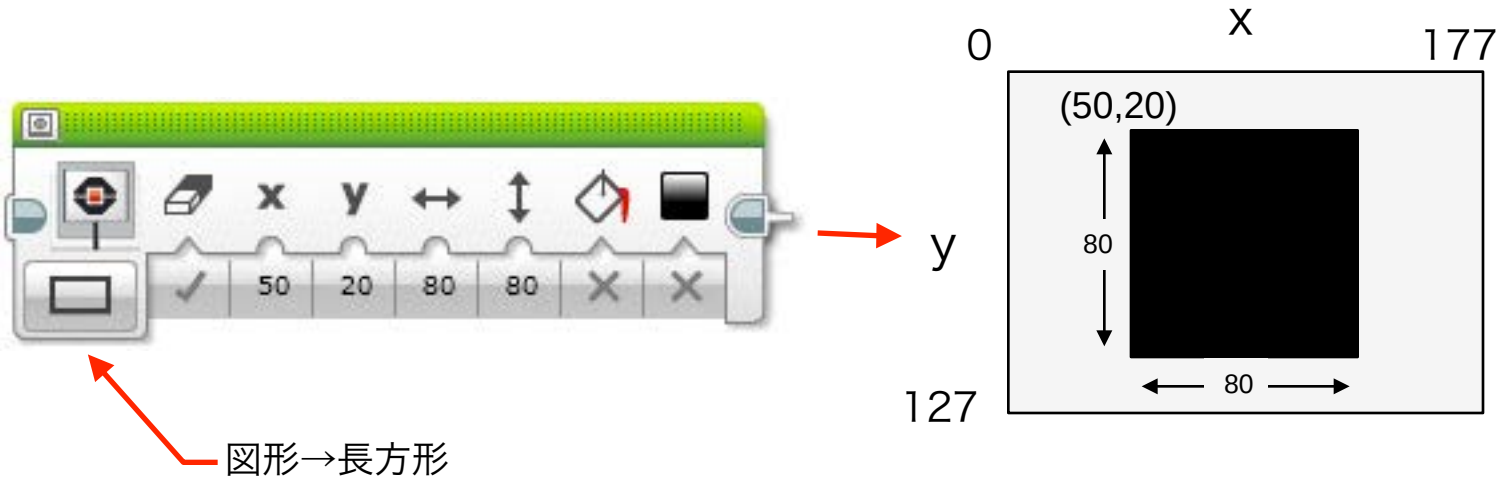
テキストの表示


表示したい文字列を入力
MINDSTORMS
x
y
AA
0
0
2
テキスト→ピクセル
0
x
177
MINDSTORMS
y
127
ピクセル座標


HELLO!
|x|
y
AA
6
6
2
テキスト→グリッド
0
|x|
21
y
HELLO!
12
グリッド座標

図形の表示


表示したい文字列を入力
x
y
50
20
40
図形→円形
0
x
177
(50,20)
40
y
127


x
y
50
20
80
80
図形→長方形
0
x
177
(50,20)
80
80
y
127

超音波センサの値を表示

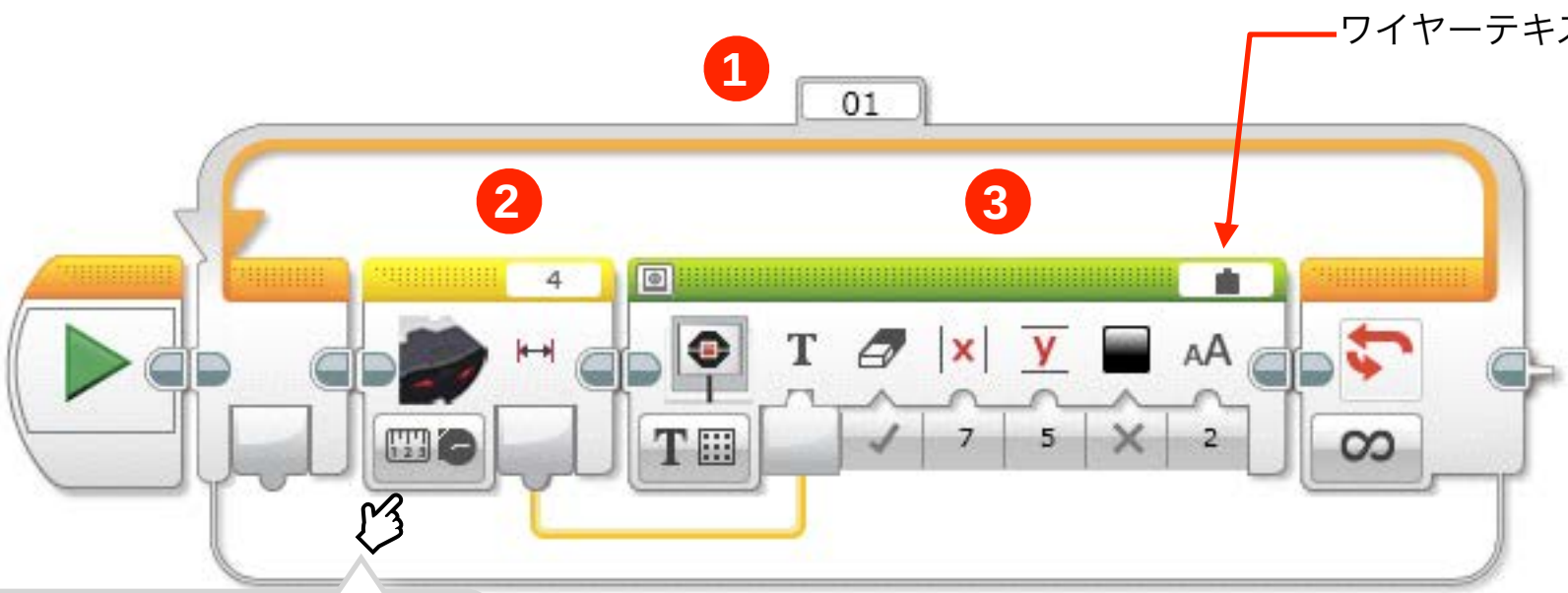
- 超音波センサの値を液晶ディスプレイに表示
  - デバッグに重要

超音波センサの値を表示(PAD)

・超音波センサの値を読み取り、テキストをディスプレイに表示

超音波センサの値を表示

・ 超音波センサの値を読み取り、テキストをディスプレイに表示



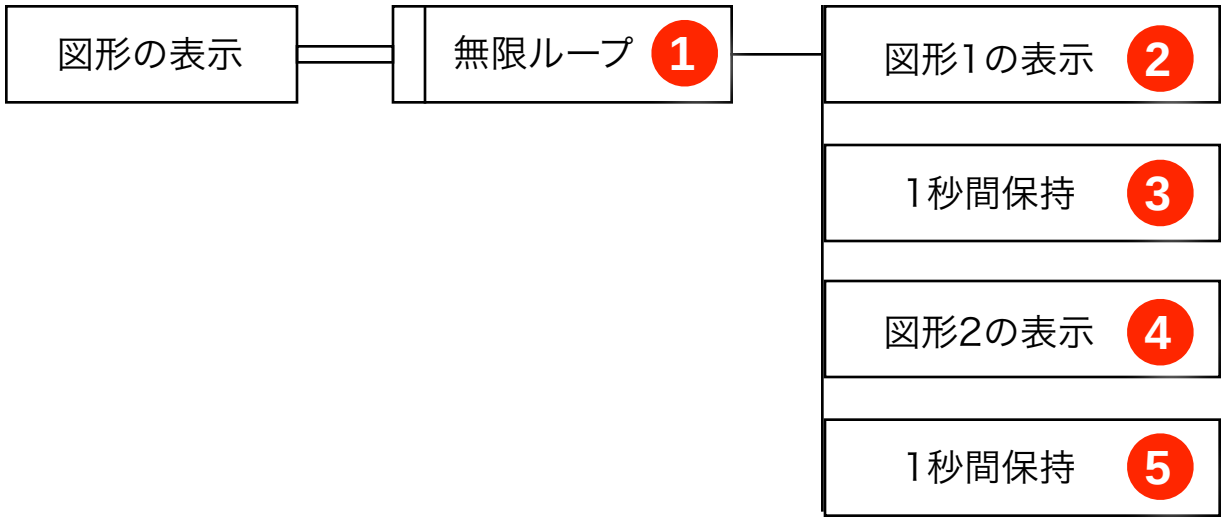
測定-接近度を指定

データワイヤの色と意味

黄	：数値
緑	：ロジック（正偽値）
オレンジ	：テキスト（文字）

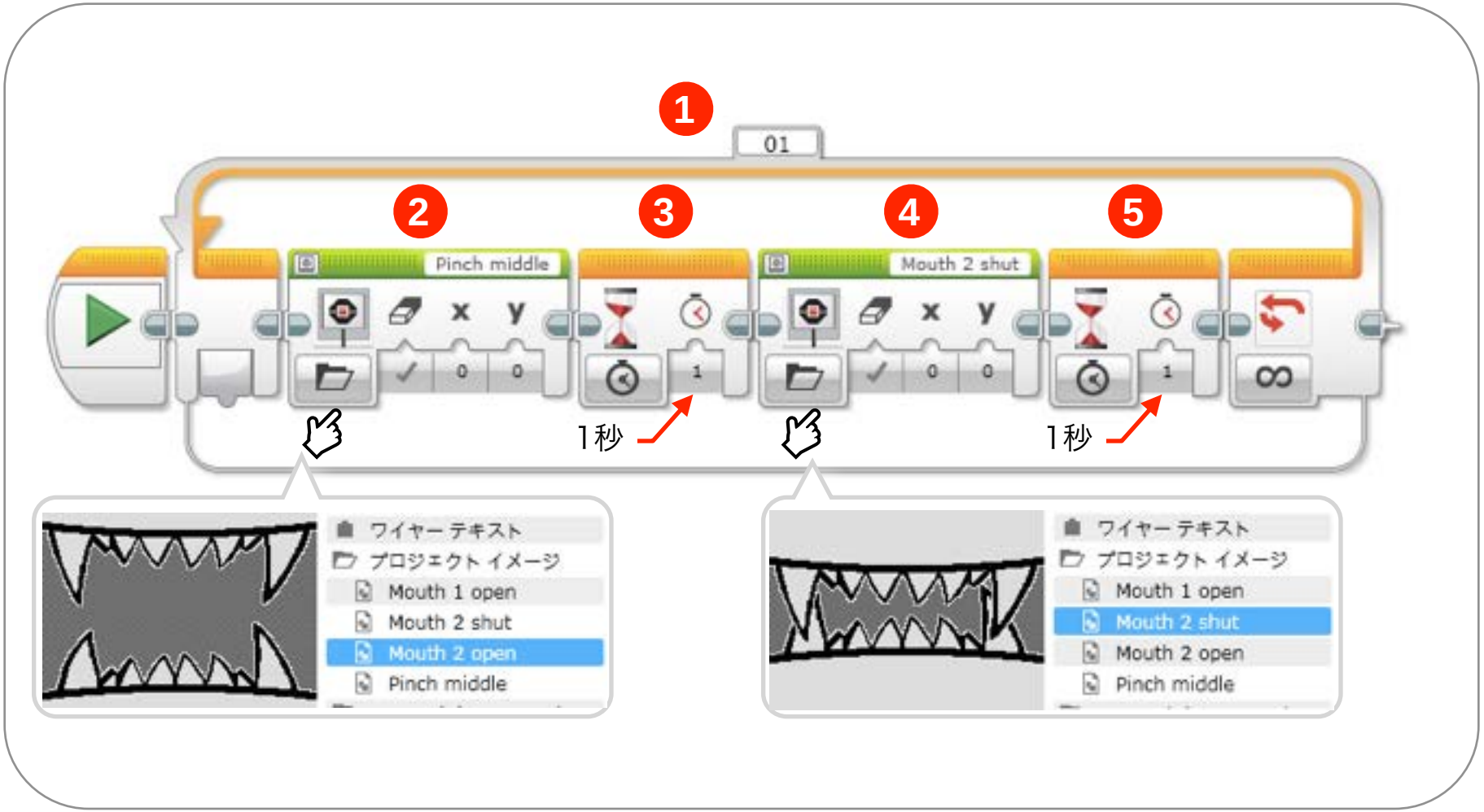
図形の繰り返し表示(PAD)

・図形1と図形2を繰り返しにディスプレイに表示



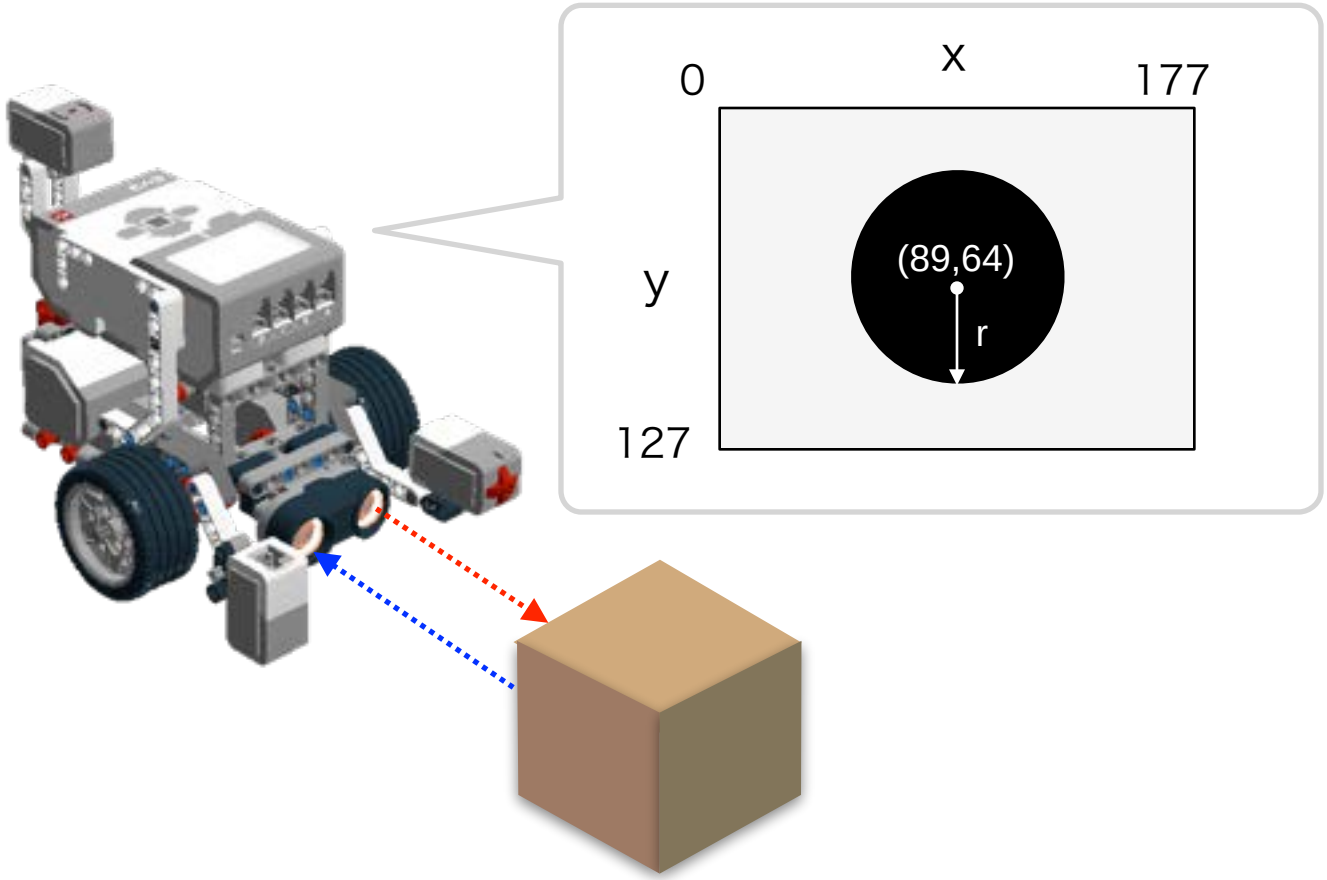
並列タスク

・ 図形1と図形2を繰り返しにディスプレイに表示



■課題

・ 障害物までの距離が小さくなるほど大きくなる円を表示



■並列タスク

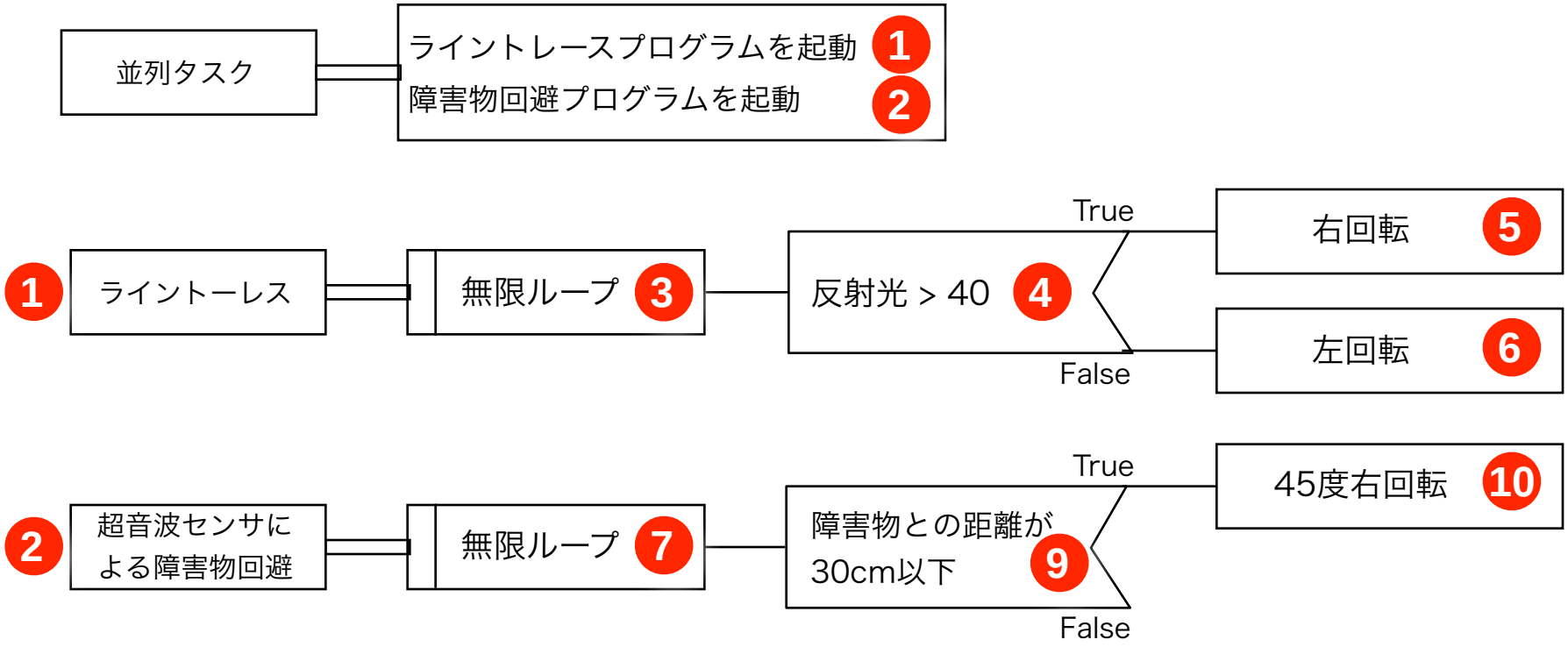
シングルタスク

- シングルタスクと並列タスク

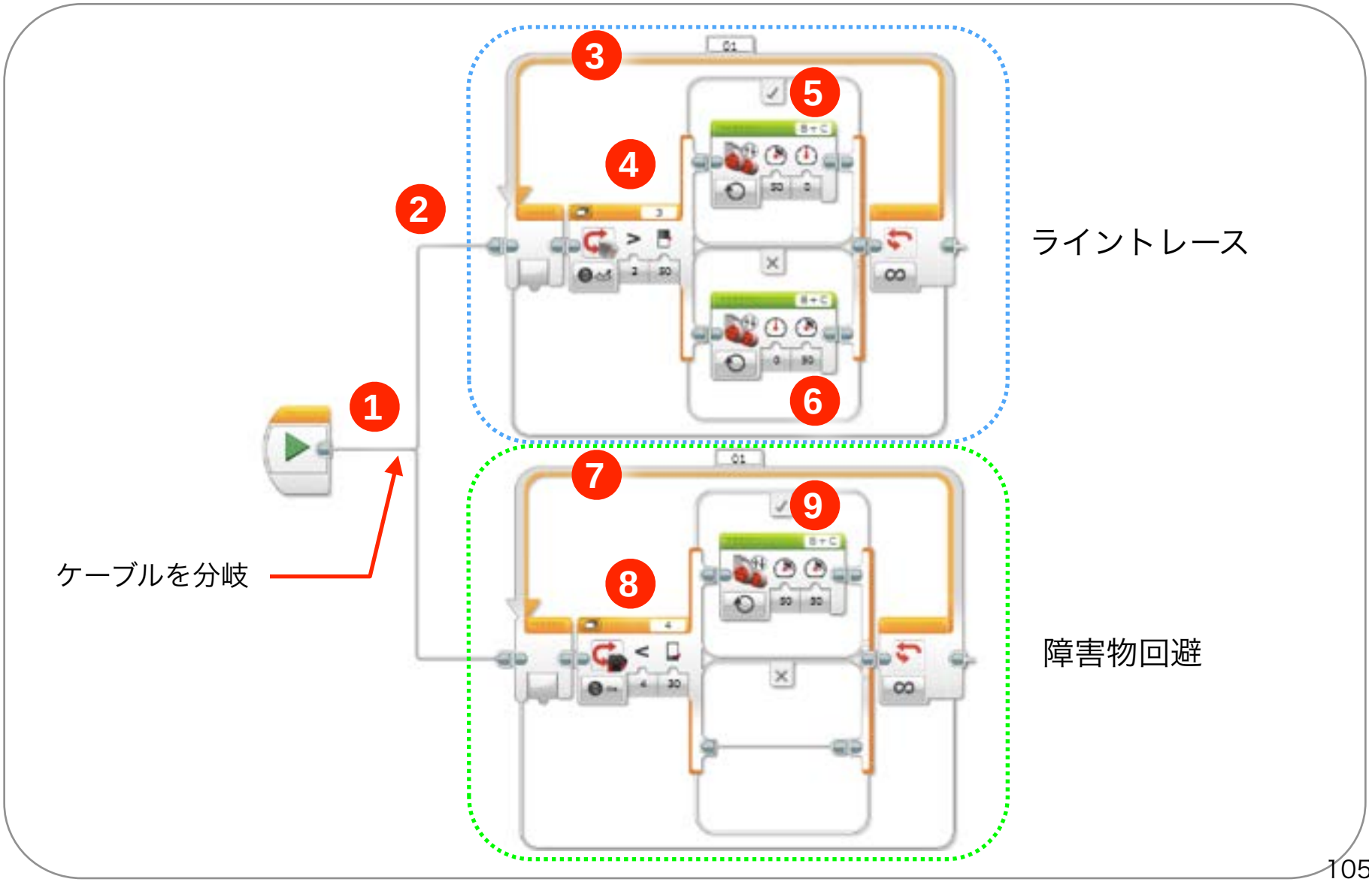
(a) シングルタスク　(b) 並列タスク

→ライントレースと障害物回避を同時に行うには並列タスクを利用

PADによる並列タスクの図示



並列タスク



並列タスクにおける問題

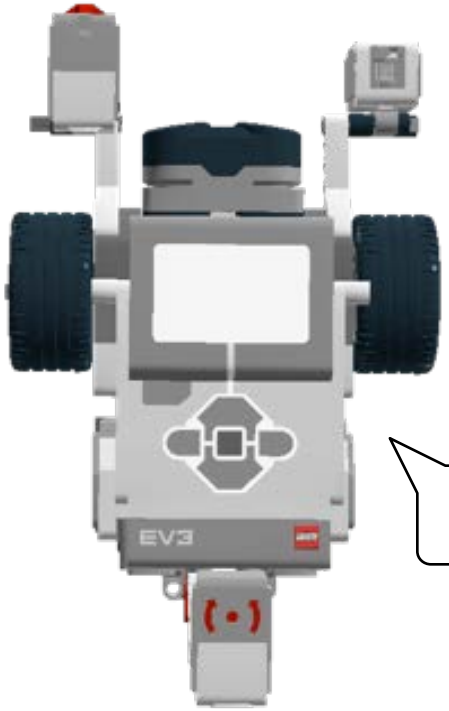
- 命令の衝突（コンフリクト）
  - タスクAがタスクBの実行を妨害
  - 同時にモータを制御

タスクA：前進

モータBCを順方向に回転

タスクB：後退

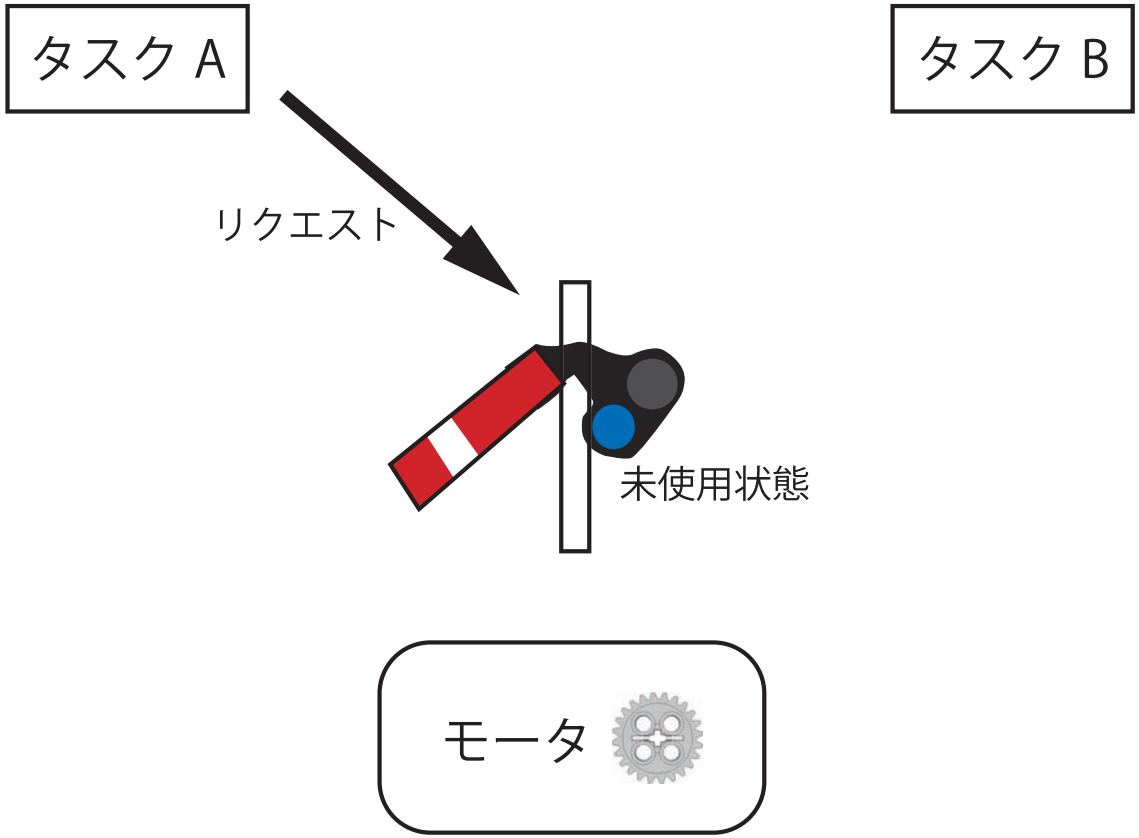
モータBCを逆方向に回転



どっちを実行すればいいの？

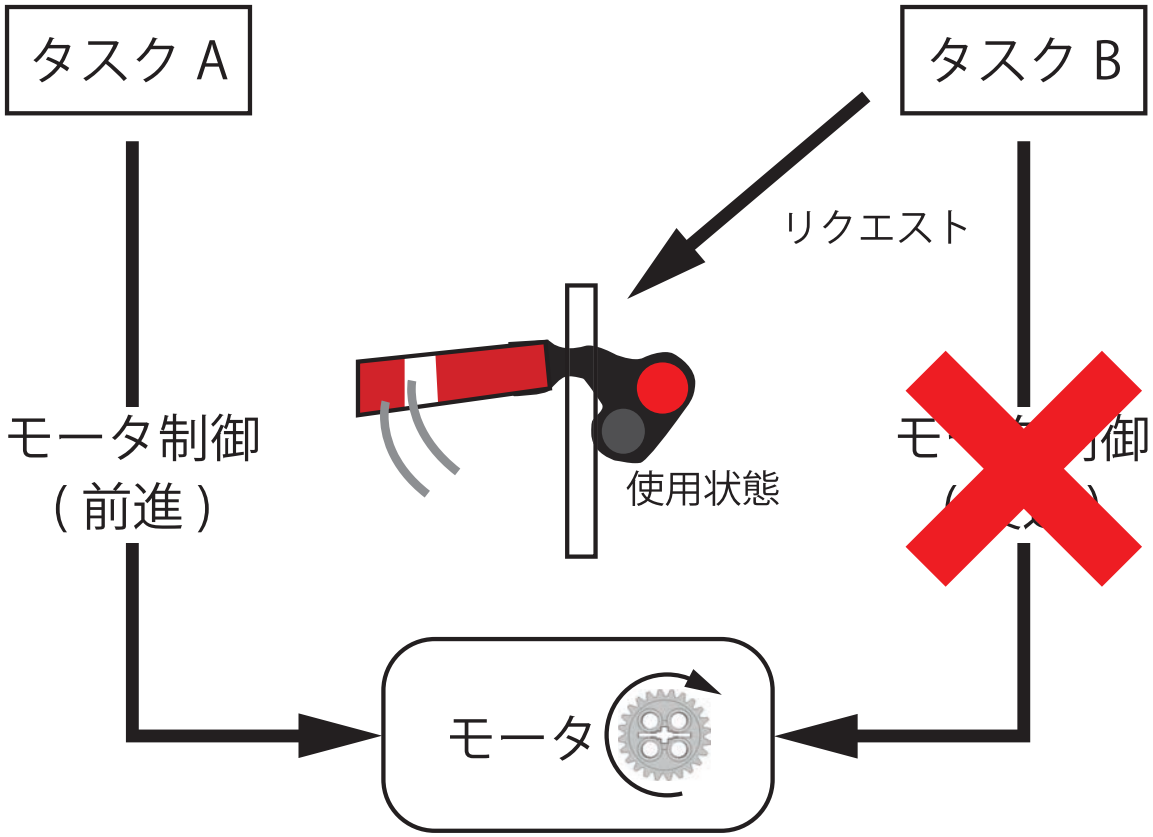
セマフォ（信号灯）

① モータの未使用状態

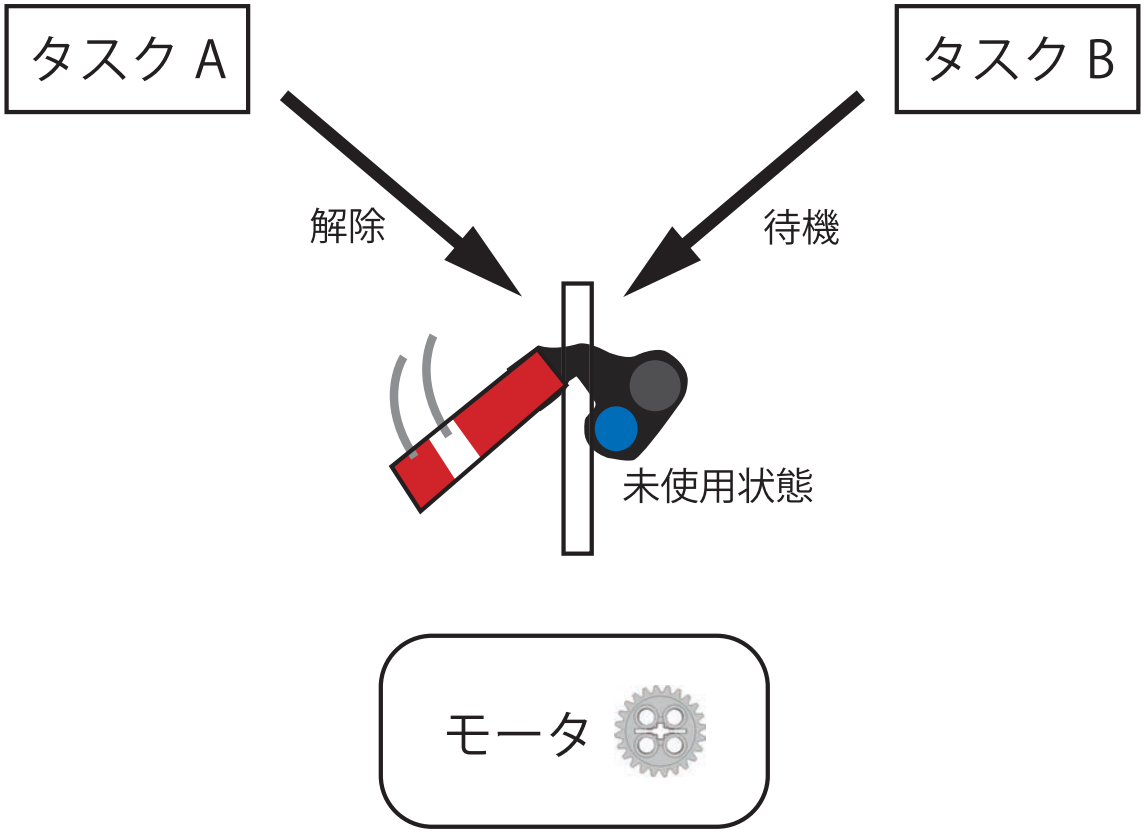


セマフォ（信号灯）

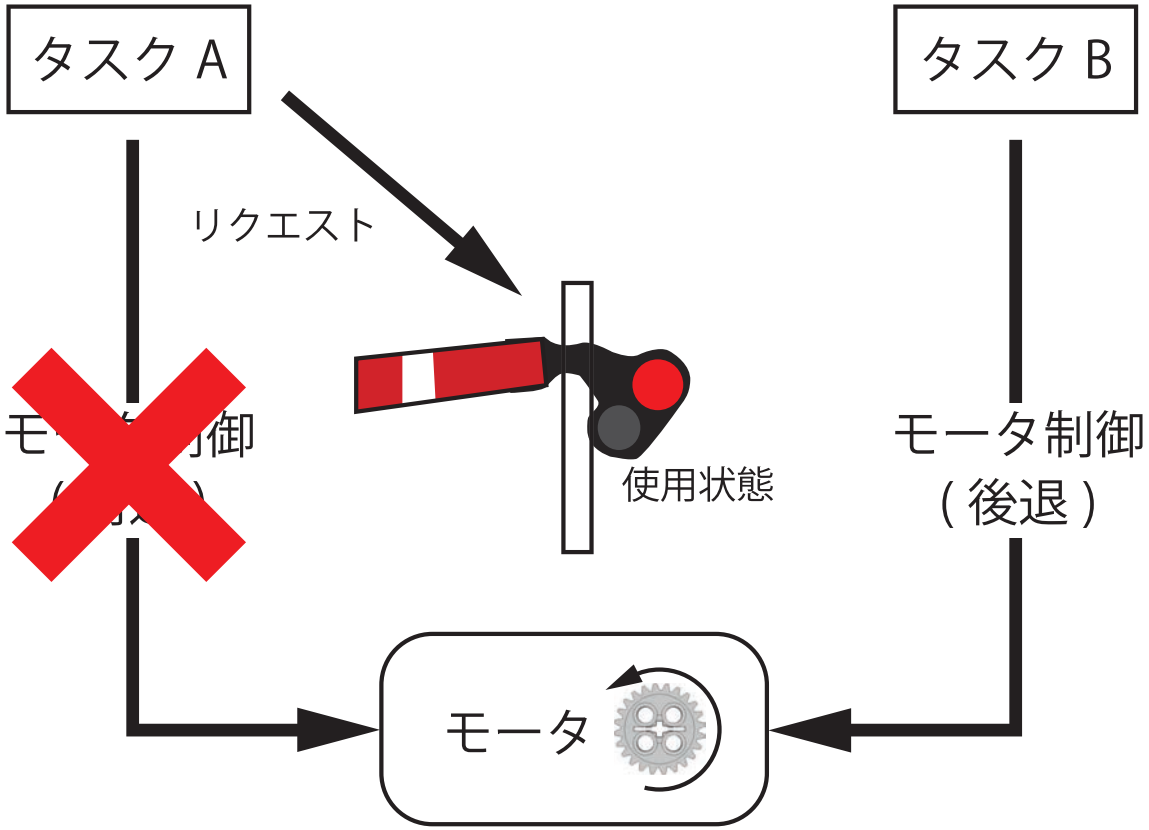
② タスク A によるモータ制御



③ タスク A によるモータ制御終了

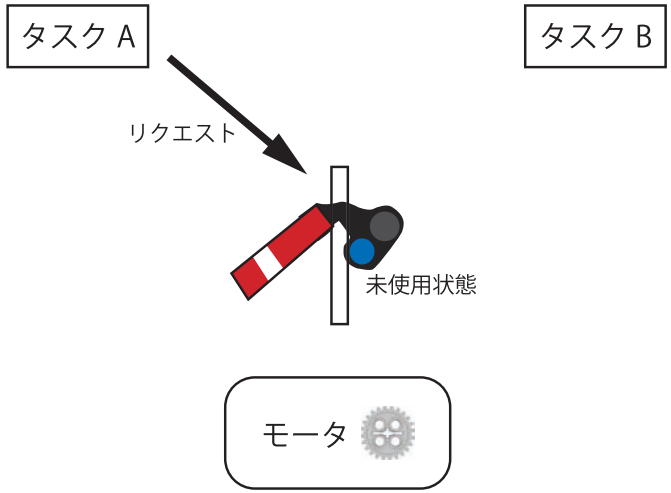


④ タスク B によるモータ制御

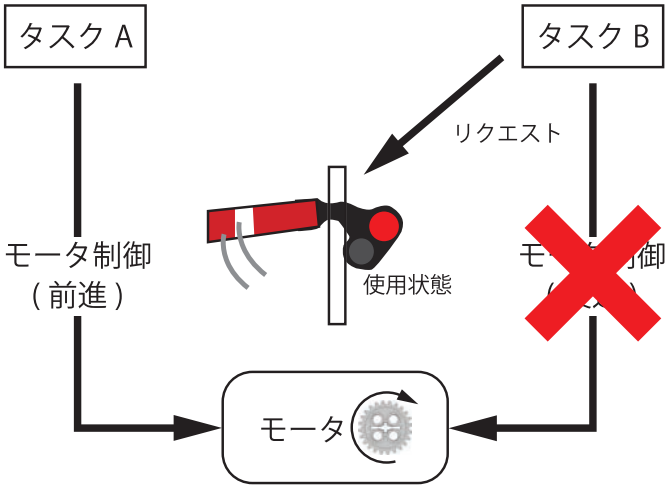


セマフォによるコンフリクトの回避

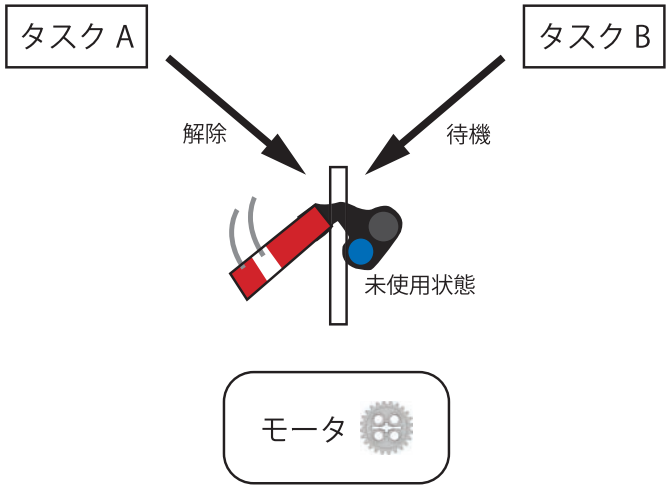
① モータの未使用状態



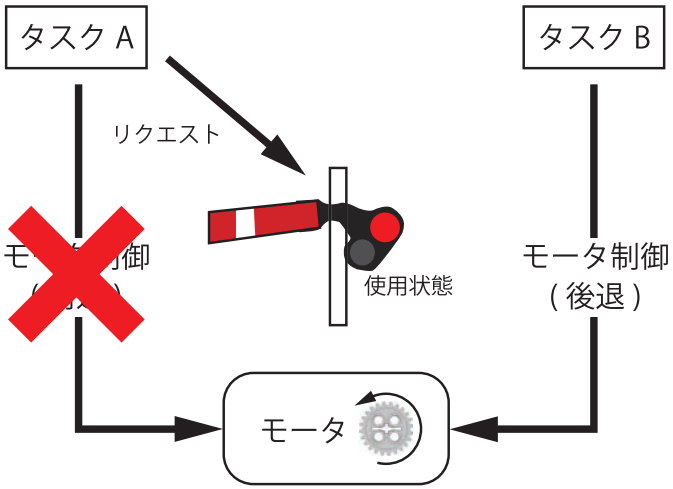
② タスク A によるモータ制御



③ タスク A によるモータ制御終了

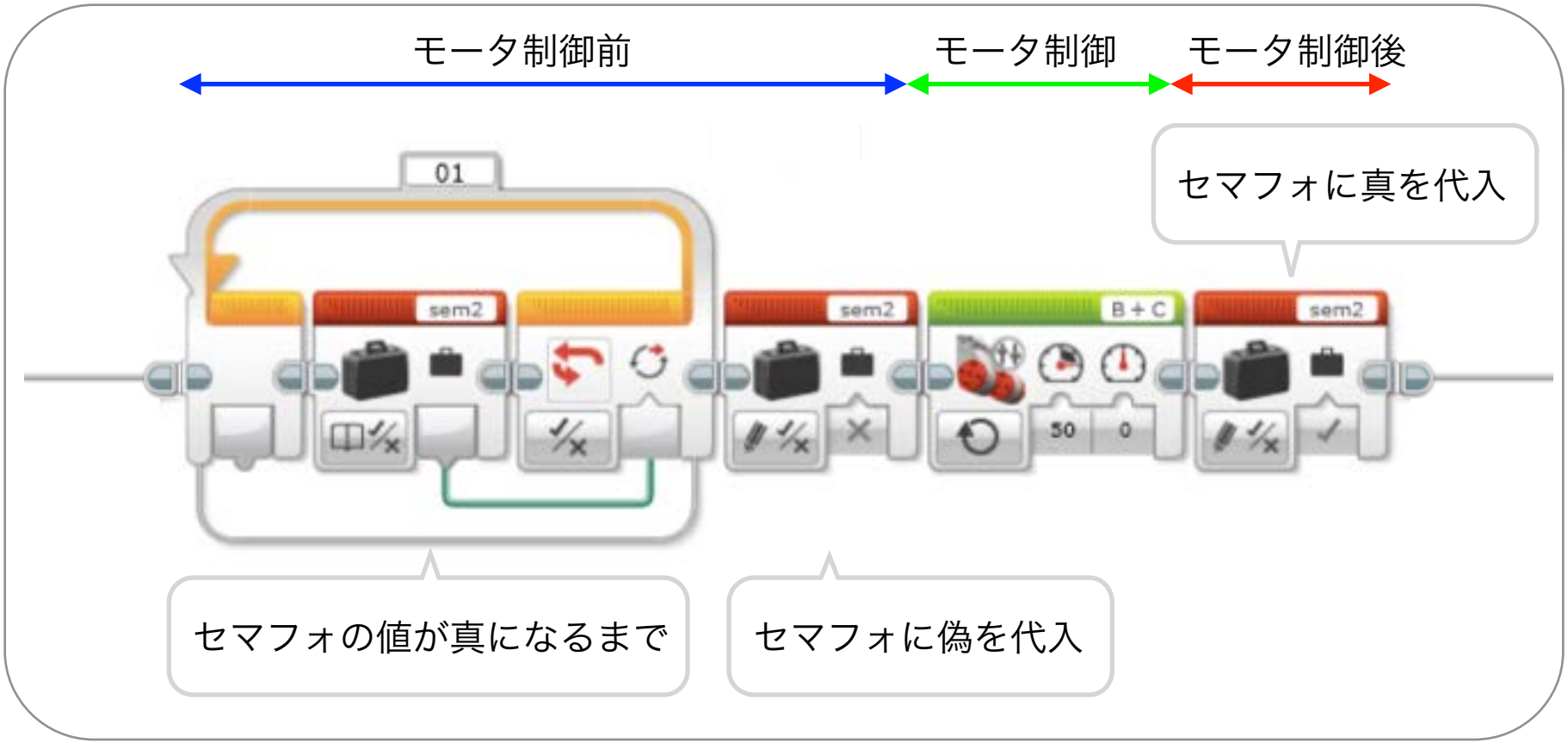


④ タスク B によるモータ制御

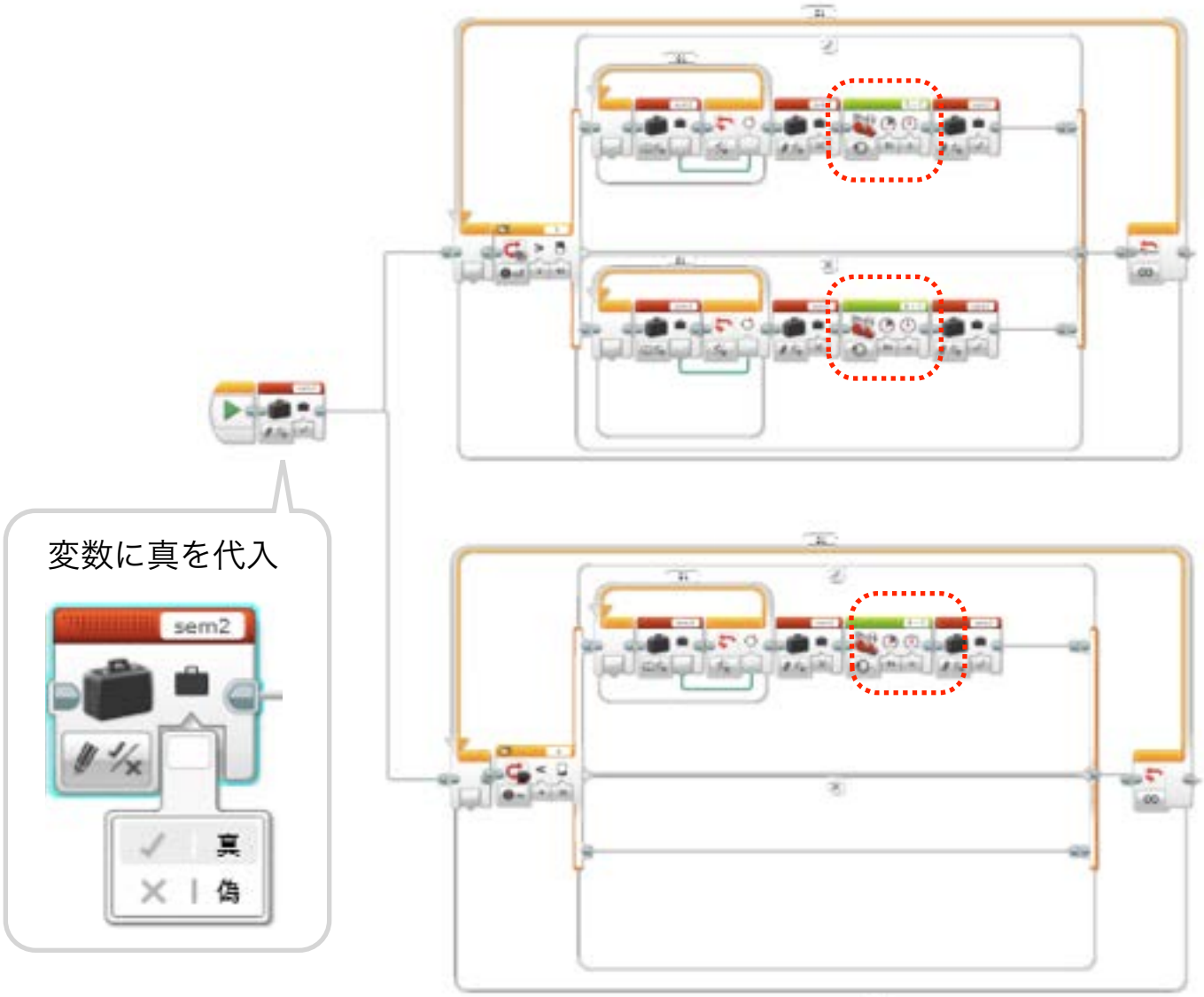


セマフォによるコンフリクト回避

・モータ制御の前に、セマフォによる待機とセマフォに偽を代入
・モータ制御の後には、セマフォに真を代入



セマフォによるコンフリクト回避



藤吉 弘亘 (Hironobu Fujiyoshi)
中部大学 工学部情報工学科（ロボット理工学科），大学院工学研究科情報工学専攻　教授

1997年　中部大学大学院 博士後期課程修了
1997～2000年　米国カーネギーメロン大学 ロボット工学研究所 Postdoctoral Fellow
2000年　中部大学 講師
2004年　中部大学 准教授
2006年　米国カーネギーメロン大学 ロボット工学研究所 客員研究員
2010年～中部大学 教授
2014年～名古屋大学 客員教授

博士（工学）計算機視覚、動画像処理、パターン認識・理解の研究に従事
ロボカップ研究賞(2005年)，山下記念研究賞(2009年)，情報処理学会論文賞(2009年)，画像センシングシンポジウム優秀学術賞(2011年,2013年)，電子情報通信学会ISS論文賞(2013年)

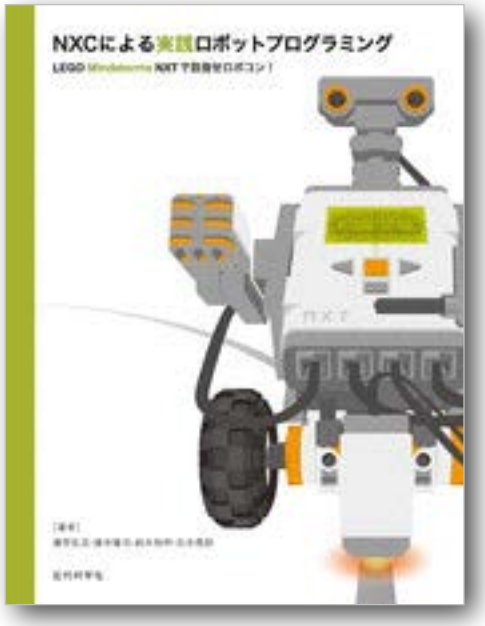
E-mail: hf@cs.chubu.ac.jp
WEB: http://www.vision.cs.chubu.ac.jp/
Facebook: facebook.com/hironobu.fujiyoshi
Twitter: @hf149

[単行本]
実践ロボットプログラミング
LEGO Mindstorms NXTで目指せロボコン!
近代科学社
ISBN-13: 978-4764903784

[WEBサイト]

http://www.robot-programming.jp/

[電子書籍iBook]
NXT-SWによる実践ロボットプログラミング
近代科学社
https://itunes.apple.com/jp/book/id902846245



[電子書籍iBook]
NXCによる実践ロボットプログラミング
近代科学社
https://itunes.apple.com/jp/book/id902356211