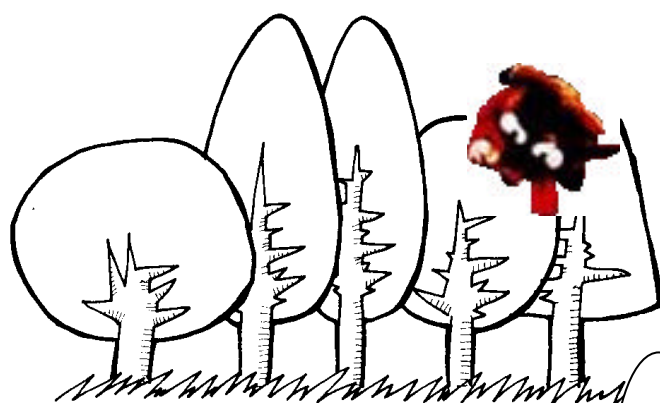


プログラム作成手引き書

ROBOLAB



たすけてー



1 ROBOLAB を起動しよう

(1)「スタート」「プログラム」から、ROBOLAB を選択すると右のような画面になります。

(2) 下のプログラムを選んでクリックしましょう。下の図のような画面になります。

(3) 次に左の inventor4 をダブルクリックします。ダイアグラムウィンドウ、ファンクションズウィンドウが表示されます。

おもに使うのはダイアグラムウィンドウ(図1) ファンクションズウィンドウ(図2) ツールパレット(図3)です。

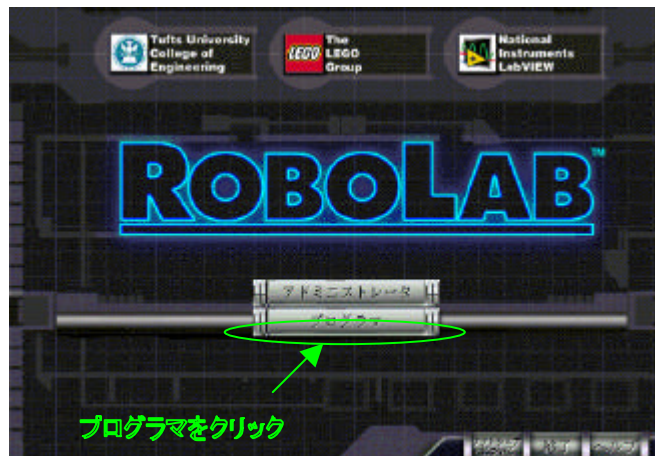
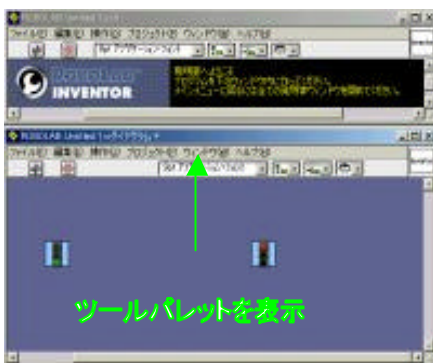
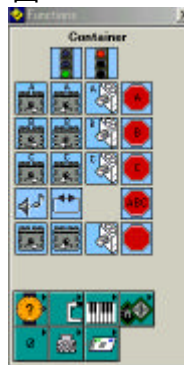


図 1



【ダイアグラムウィンドウ】

図 2



【ファンクションズウィンドウ】

図 3



【ツールパレット】

* ツールパレットはダイアグラムウィンドウの「ウィンドウ」から「ツールパレットを表示」を選んで表示させます。プログラムを作る時はいつも表示させておくようにしましょう。

(4) 必要に応じてウィンドウを出す

← フังก์ション ウィンドウ

ファンクション ウィンドウの下にある7つの緑色のアイコンをクリックするとそれぞれに関連したウィンドウが出てきます。必要に応じてこれらのウィンドウを利用できるようにしましょう。

クリックする

【Modifiers】

修飾という意味
・出力、入力、場所、モーターのパワーレベルなどを制御する時に使う。

【 Wait for 】

待つという意味
・時間をこのアイコンで制御する。
・モーターを動かす時間などを制御する時に使う。

【 Structures 】

構造という意味
・センサーを利用した条件による動きの変化、同じ動きの繰り返し、などの処理を制御する時に使う。

2 ROBO LAB でロボットを動かすプログラムを作成しよう

ROBO LAB は、必要なアイコンをダイヤグラムウィンドウに貼り付けて線でつなげていくという手順でプログラムを作成していきます。たくさんの動作をさせるためには、その中に小さな動作が多数存在することになりますが、1つ1つの動作を分けて考えて、それらの動作を組み合わせることによってプログラムを作成します。まずは、小さなプログラムの作成を練習して、最終的に大きなプログラムを完成しましょう。

日常生活の例

目標 カップラーメンを食べる

- プログラムの手順
- お湯をわかす
 - カップに注ぐ
 - 3分待つ
 - 食べる

大きなプログラムは小さなプログラムが組み合わさってできている

- 小さなプログラム
- お湯をわかす
 - ・やかんに水を入れる
 - ・コンロの火をつける
 - ・お湯がわいたら火を消す

小さなプログラムの中にもさらに小さなプログラムがある

Mission 1

「ロボットを4秒間、前進させて停止させよ！」

目標 ロボットを前進させて停止させる

プログラムの流れ



ロボットを前進させる小さなプログラム

- ・ モーターの制御(A,B,C) どのモーター？
- ・ 回転方向の制御(順回転、逆回転) 回転方向は？
- ・ 強弱の制御(ここでは省略しているが1～5まで設定)
- ・ 時間の制御 どのくらいの時間？

プログラムを作る時は、このようなプログラムの流れを考えることがとても大切！

プログラムを作ろう

(1) アイコンを貼り付ける

【ファンクションズ】からアイコンを選び、ダイアグラムウィンドウにドラッグして貼り付けます。

青信号(はじめ)

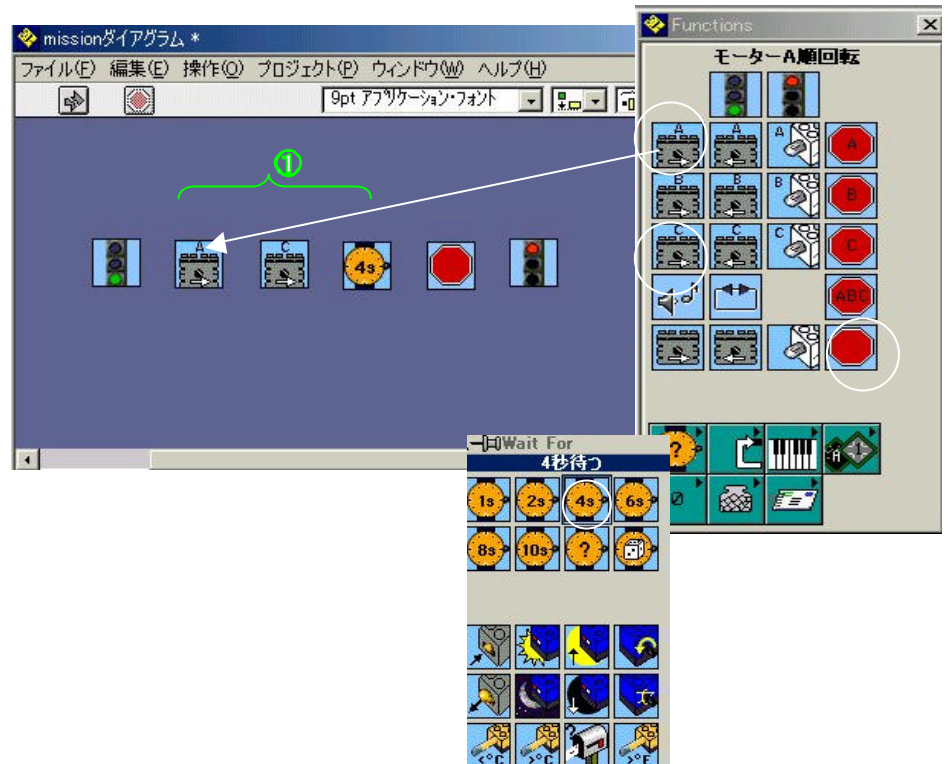
ロボットを前進させる

- ・ モーターAの順回転
- ・ モーターCの順回転
- ・ 4秒待つ

【Wait for】から4Sを貼り付ける

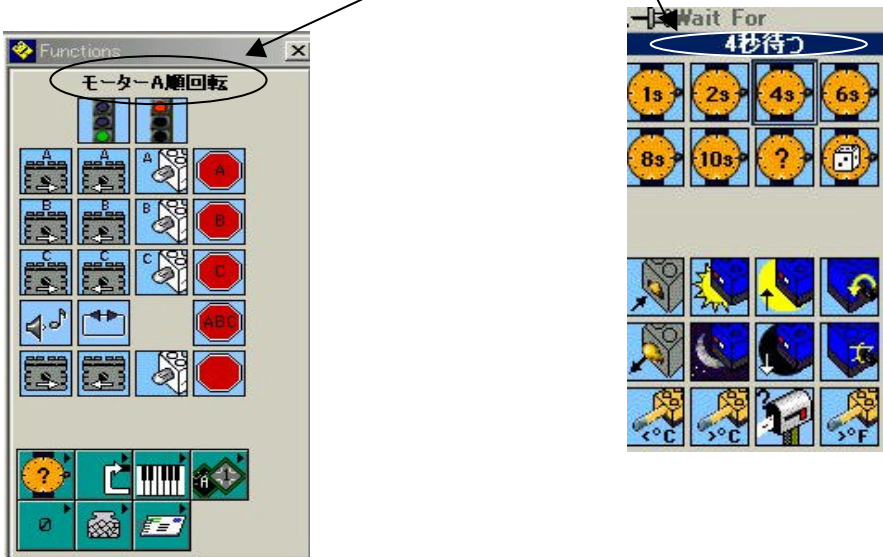
ロボットを停止させる

赤信号(おわり)



【ちょっと便利！】

マウスでクリックすると、その命令の意味がこの部分に表示されるので、これを参考にしながらやっていくとわかりやすい。



(2) アイコンを線でつなぐ

ツールレットからストリングツールを選んでおきます。

青信号(プログラムの始まり)から、モーターA、モーターC、Wait for、停止、と赤信号(プログラムのおわり)まで線でつなぎます。つなげていく時にアイコンの角の部分



部分が点滅して接続するところを教えてくれるので、それにしたがっていくと簡単につながります。


ダウンロードのボタンを見て、うまくつながったかどうかを見ます。うまくつながっていれば、いよいよダウンロード(転送)です。

*うまくつながっていない場合は、編集(E)から「不良ワイヤの削除(R)」を選び、うまくつながっていない線を削除して、つなぎなおして下さい。

*アイコンや線を間違えてつないだりして消したい場合は、の矢印を選択した後、消したいアイコンや線を選び、Delete キーを押せば消すことができます。

3 プログラムをダウンロードする

(1) RCX の電源を入れる。

(2) RCX を赤外線トランスミッタの前に置いて、センサ部分の向きを合わせて実行の  ボタンをクリックします。プログラムの転送は3～4秒程度で終わります。プログラムがうまく転送できると画面に100%転送された表示がでて、“ピロロロー”とRCXから音がします。

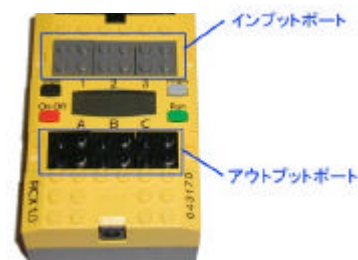


初めて実行する際には、ファームウェアをダウンロードする必要があり、4分程度の時間がかかります。また、線がきちんとつながっていないなど、プログラムに不備があると、上の矢印がきちんと表示されません。

画面の中央にはRCXに番号が表示された画面がでてきますが、これは、プログラムがダウンロードされたところを示しています。RCXには5つのプログラムを保存できるようになっていて、それぞれプログラムスロット1～5と呼ばれています。ここでは、3のところにダウンロードされたとして進めます。*うまくダウンロードできなかった場合は、右のような表示されるのでどこが悪かったのか考えてやり直しましょう。センサ部分がちゃんと向き合っていなかったり、コードがきちんとつながっていなかったり、電池が入っていなかったり、電源が入っていなかったりすることが主な原因です。



4 プログラム実行



(1) モーター、センサとRCXをケーブルで接続する。このとき、モーターはアウトポート側に、センサはインポート側に接続します。電極の部分をしっかり合わせて接続しましょう。また、モーターとアウトポートの接続では、接続するケーブルの向きによってモーターの回転方向が変わります。

(2) RCX側の液晶もプログラムが正しくダウンロードされると図1のように表示されます。ここでは3のところにダウンロードされたとして考えていますので、もし表示が3になっていなければ、Prgmのボタンを押して3が表示されるようにしましょう。

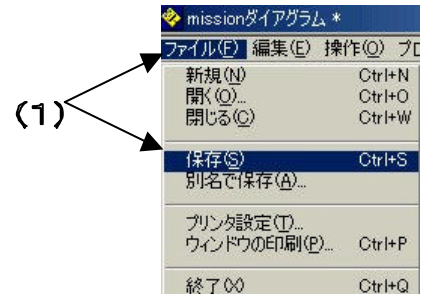
(3) 図1の緑色のボタンのRunを押すとロボットが動作します。



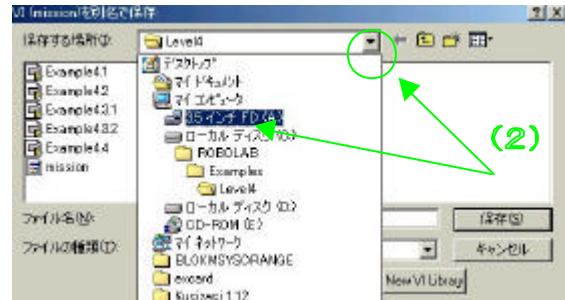
5 プログラムをフロッピーディスクに保存する

プログラムが正しく作成できたらプログラムをフロッピーディスクに保存しておきましょう。

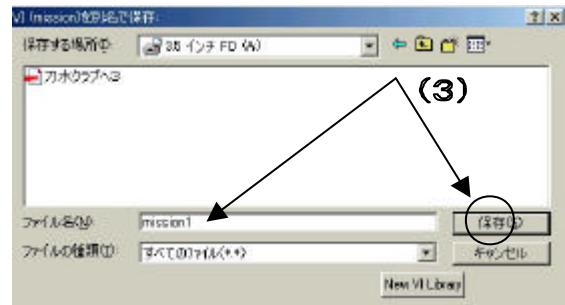
(1)「ファイル」から「保存」を選択します。



(2)保存する場所の をクリックし 3.5 インチ FD を選択します。



(3)ファイル名に名前(ここでは mission 1 と入力してみましょう)を入力し、「保存」をクリックします。

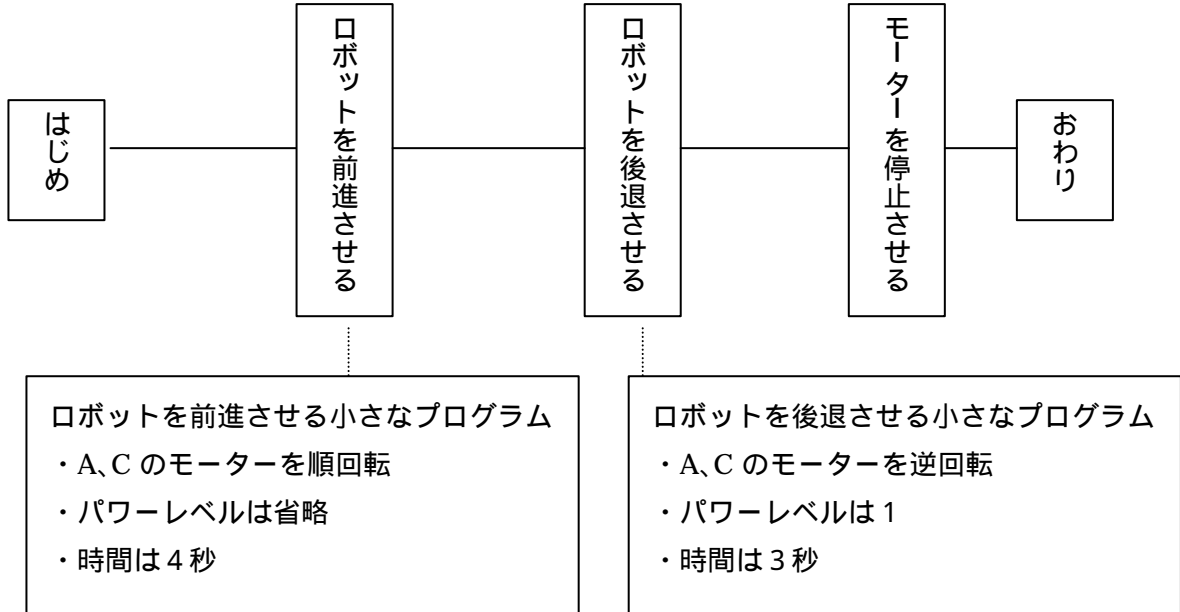


データの保存までの練習をしました。できた人は次の課題に進んでいきましょう。

Mission 2

「ロボットを4秒間前進させ、その後パワーレベル1で3秒間後退して停止させよ！」

目標 ロボットを前進させ、その後、後退させて停止させる
プログラムの流れ



Mission 1 のプログラムに3秒間パワーレベル1で後進するプログラムを付けたします。まず、余分なアイコンと線を消しましょう。

Mission 1 の停止のアイコンとそこにつながる線を消して、赤信号の位置を右へずらします。ツールパレットの「オブジェクトの位置決め/サイズ変更/選択」を選び、停止のアイコンや隣



の線を選択して Delete キーを押すとアイコンや線を消せます。そして、赤信号を選択して右へドラッグすれば位置が変わります。

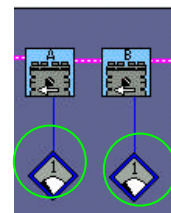
プログラムの解説

(1) パワーレベル1の後進


後進はモーターの回転方向を逆にします。パワーレベル1にするには、ファンクションズ ウィンドウの【Modifiers】から

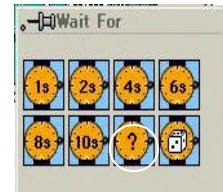


を選択してモーターの下に貼り付け、線でつなぎます。パワーレベルは1～5まで設定できますが、設定しない場合は自動的にレベル5に設定されます。



(2) 3秒間の待ち時間の設定


3秒間というアイコンはないので、ファンクションウィンドウの【Wait For】から  のアイコン「待つ」を選択して貼り付けます。このアイコンは待つ時間を自分で設定する時に使います。そしてこの場合は、その時間を数値として入力し、アイコンの下に入力します。



【Modifiers】の  「数値定数」を選択し

「待つ」のアイコンの下に貼り付け、線でつなぎます



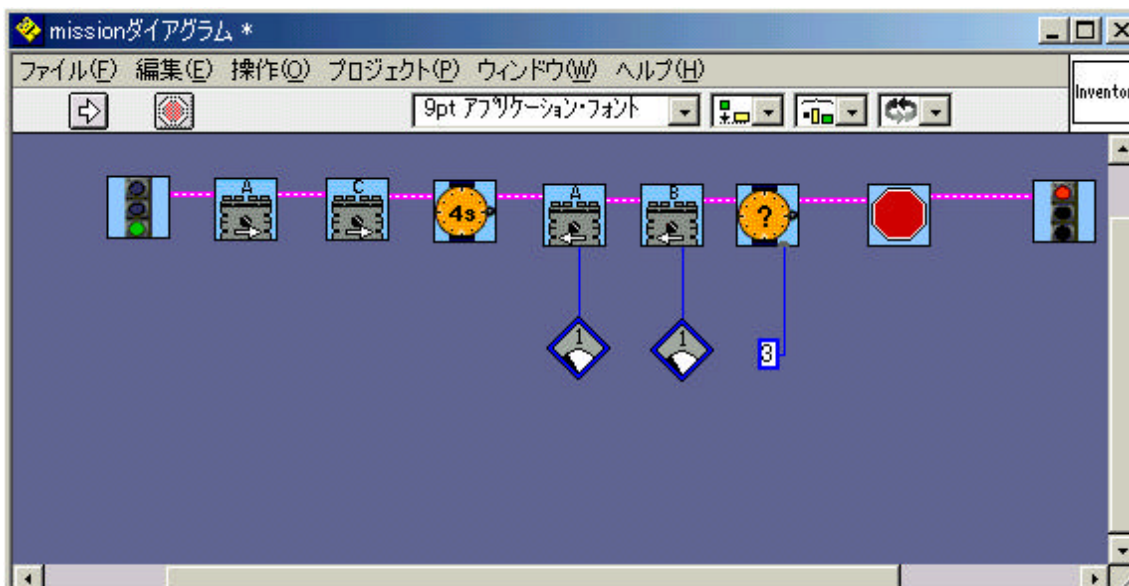
ツールパレットから  を選択し、「数値定数」の四角の中をクリックします。すると数字が入力できるようになるので、ここでは四角の中に3を入力します。(少数点以下の入力もできます)



(3) 停止は Mission 1 の時と同じアイコンを貼り付け、線でつなぎます。

(4) プログラムを実行させ、目的とする動作をしたら「ファイル」「別名で保存」を選択し「mission 2」という名前で保存しておきましょう。目的とする動作をしなかった場合は、原因を分析してやり直してみてください。

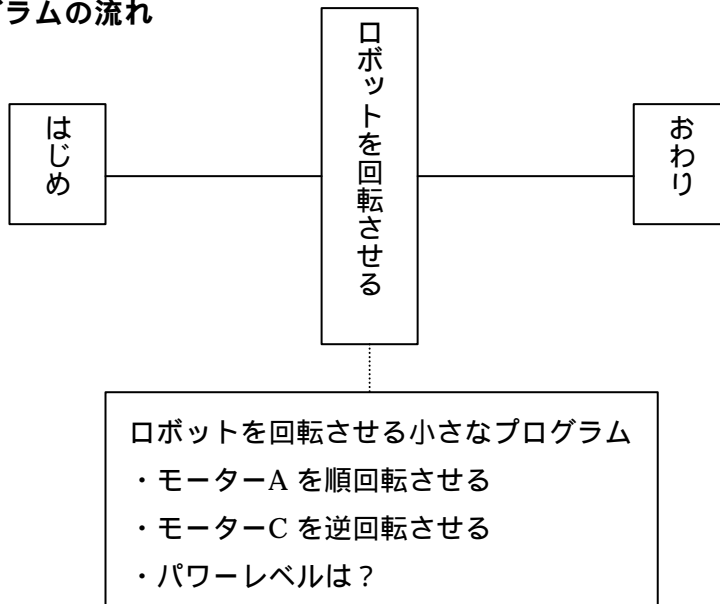
Mission 2 の参考例



Mission 3

「ロボットを回転させよ！」

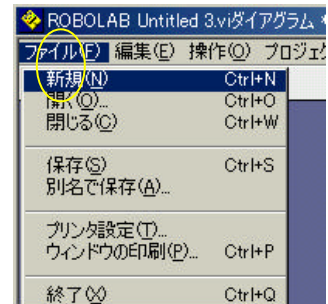
目標 ロボットを回転させる
プログラムの流れ



プログラムの解説

ロボットを回転させるには2つのモーターの回転する方向を逆にする。どちらか1つのモーターだけを回転させる。などが考えられます。

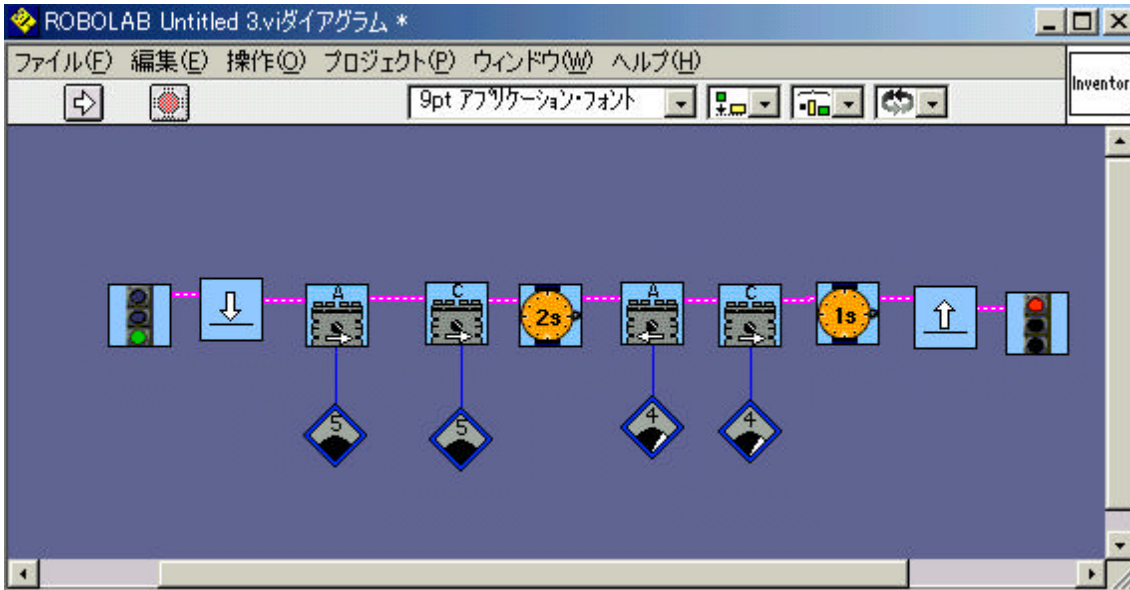
- (1) 「ファイル」「新規」を選択し、新しい画面を表示します。
- (2) 回転方向が逆のモーターを貼り付け、線でつなぎます。
- (3) プログラムを実行させ、目的とする動作をしたら「別名で保存」で「mission3」の名前で保存しましょう。



Mission3 の参考例



Mission4 の参考例

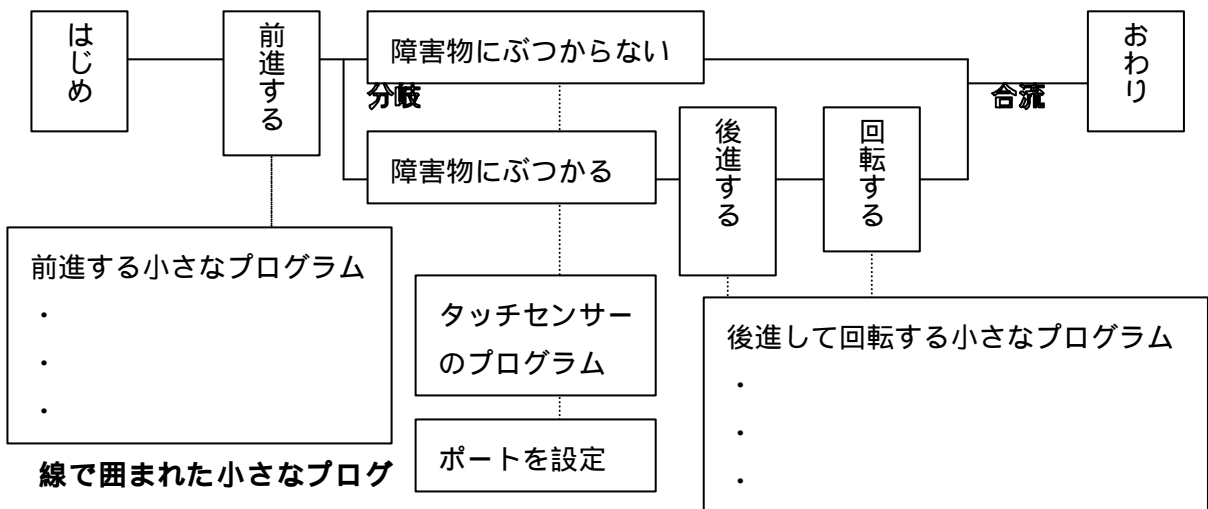


Mission5

「障害物にぶつくと、いったん後退した後、回転！」

目標 ロボットを前進させて障害物にぶつけ、いったん後退した後停止する。

プログラムの流れ



ラムを自分で書いてみよう！

プログラムの解説

(1) 前進する (Mission 1)

(2) 障害物にぶつかる

【ストラクチャズ】の「タッチセンサー分岐」を使う。



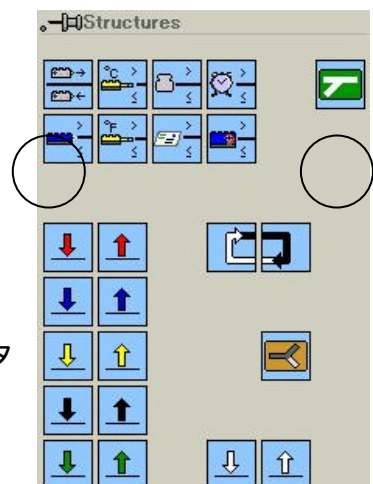
タッチセンサーが押されない状態



タッチセンサーが押された状態

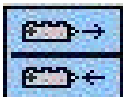


「分岐合流」タッチセンサー分岐した場合は必ず使う。タッチセンサーで分岐した流れを1つに合流させます。



ここでは、障害物にぶつかったかどうかをタッチセンサーで判断させます。

障害物にぶつかる = タッチセンサーが押される



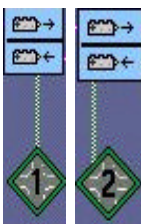
タッチセンサーが押されなければそのまま前進させる

タッチセンサーが押されたら後退して回転させる



合流させる

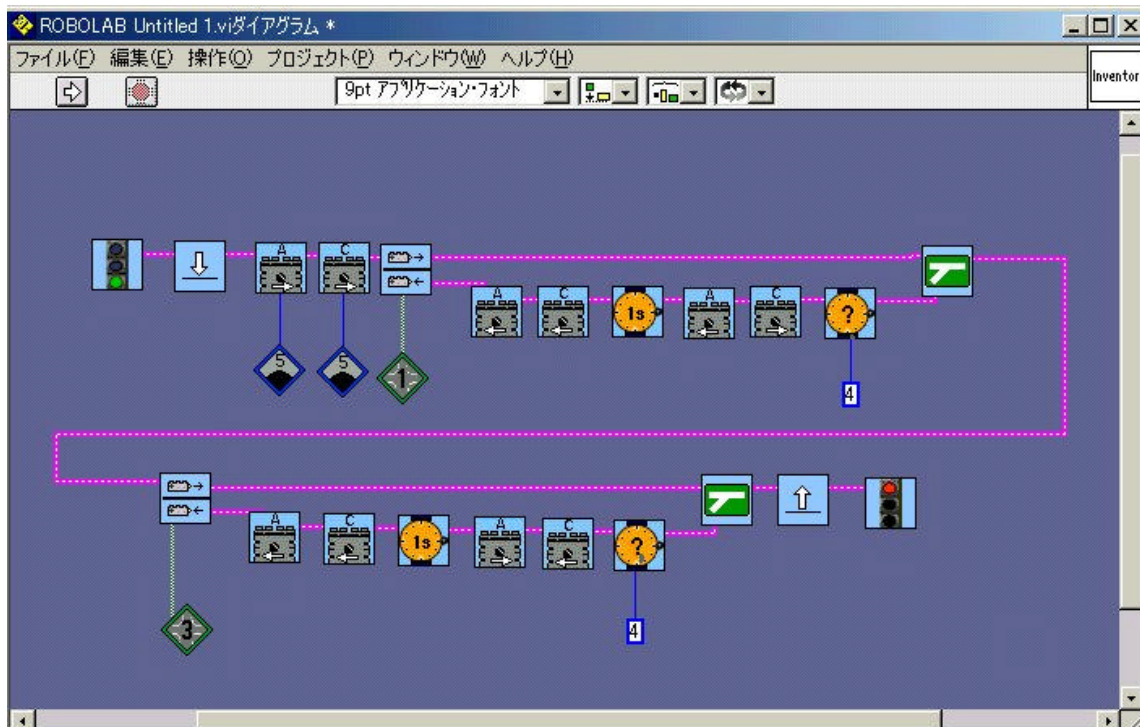
* 2つのタッチセンサーを使う場合は、それぞれのタッチセンサーをインポートポートの1・2・3のどれかに接続し、どのポートのタッチセンサーかを設定します。1と2に接続した場合は、【Modifiers】から入力1と入力2を選択し、タッチセンサー分岐の下に貼り付けることになります。



今回は左右のタッチセンサーを使用するので、左のタッチセンサー、右のタッチセンサーを接続してあるポートを確認してプログラムを作りましょう。そして、左のタッチセンサーが押されたら・・・右のタッチセンサーが押されたら・・・をいう条件を考えていくことになります。

- (3) 後退する (Mission2)
- (4) 回転する (Mission3 を参考)
- (5) アイコンを貼り付け、線でつなぎます。
- (6) プログラムを実行させ、目的とする動作をしたら別名で保存しましょう。

Mission5 の参考例



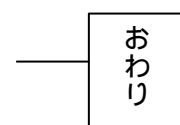
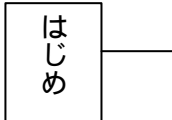
Mission 6

「赤ずきんちゃんを救出せよ！」

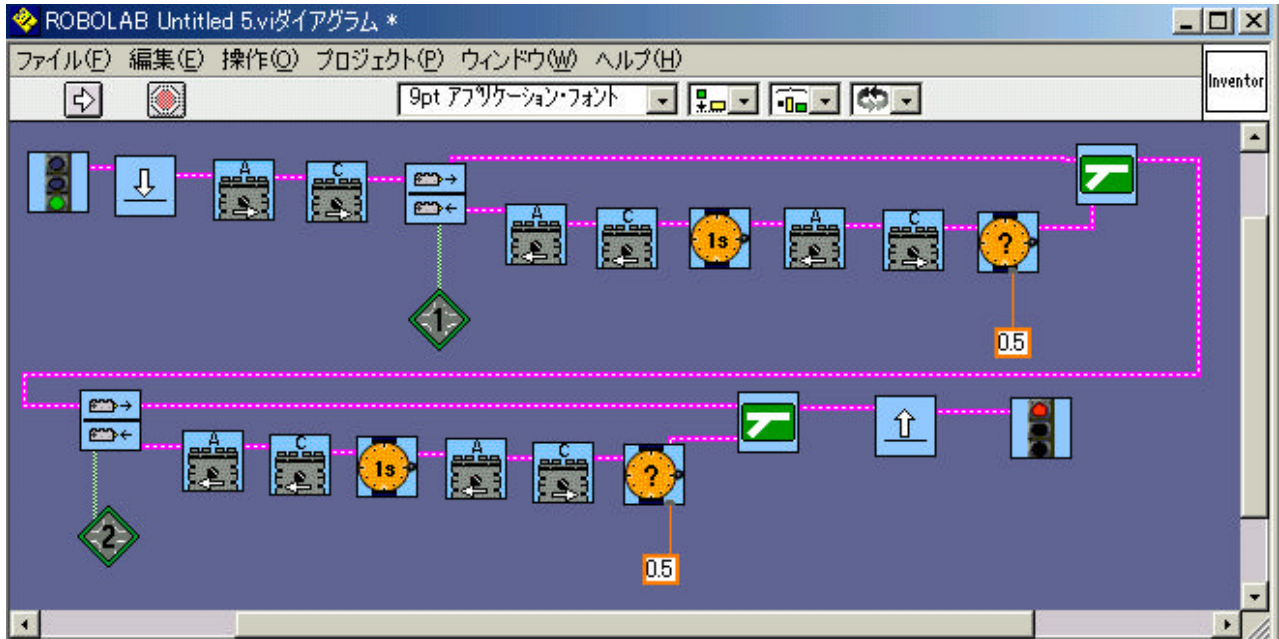
Mission1 ~ Mission5までで学習したことを総合させて赤ずきんちゃんを救出するためのプログラムを作成しましょう。ここまでの学習をもとに目標、プログラムの流れを自分で考え、その手順をプログラムにしていきましょう。

目標

プログラムの流れ



参考例



これはあくまで参考ですので、ここまでの学習をもとに自分なりにプログラムを工夫してください。また、光センサーを使った制御もできますので、興味のある人は下の説明を参考に挑戦してみてください。

ライトセンサー分岐を使う

光センサーを使って条件を判断させる場合は【ストラクチャーズ】の「ライトセンサー分岐」を使用します。



指定した明るさより明るい場合

指定した明るさより暗い場合

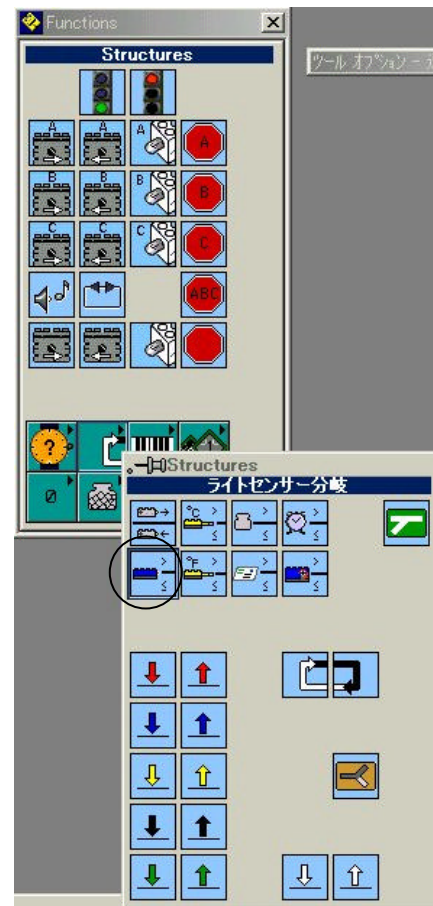
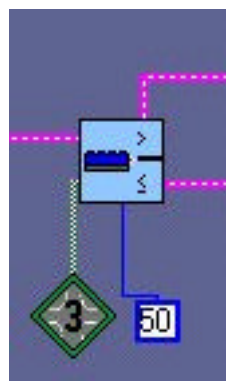
使い方はタッチセンサーの時とほぼ同じですが、どのくらいの明るさにするかを【Modifiers】の「数値定数」で設定します。

例

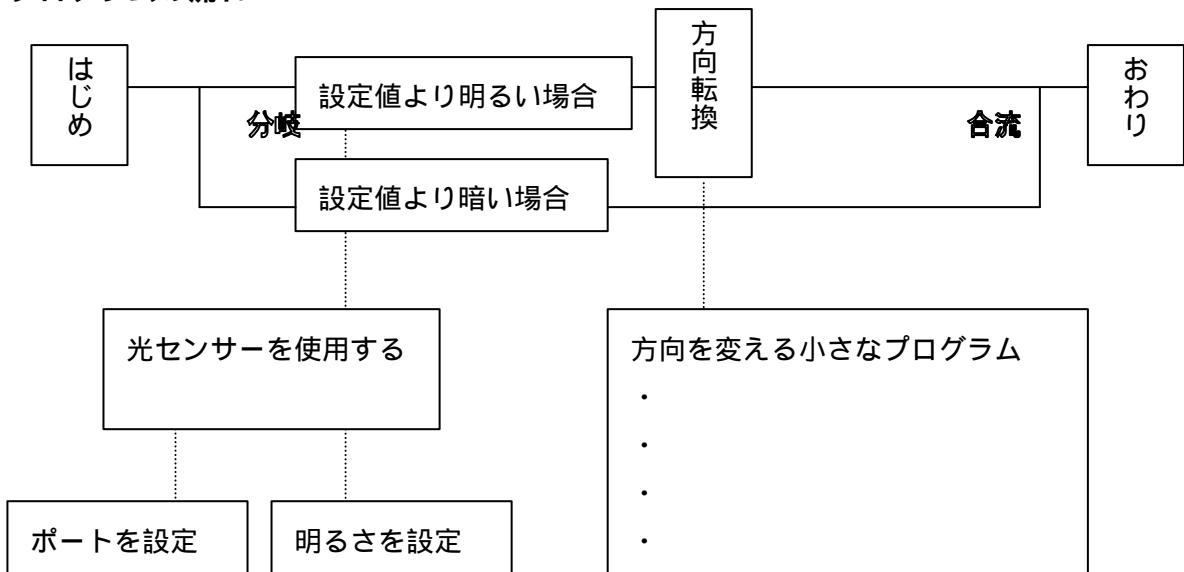
ライトセンサー分岐の「比較%」の所と「数値定数」を線でつなぎ、定数を設定しましょう。(ここでは50を設定しています。)

ライトセンサーが感知する明るさの数値は、実際にプログラムを実行させ、RCXのViewのボタンを2回押すと画面に表示されます。その数値を参考に明るさを設定していきましょう。

光センサーを使って、ロボット制御するプログラムを加えても楽しいかも知れません。



目標 ライトセンサーが 30 より大きい明るさを感知すると方向を変える
プログラムの流れ



参考例

