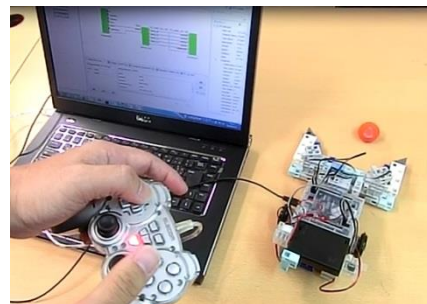


StarTno_01_基本マニュアル(Ver.1.0)

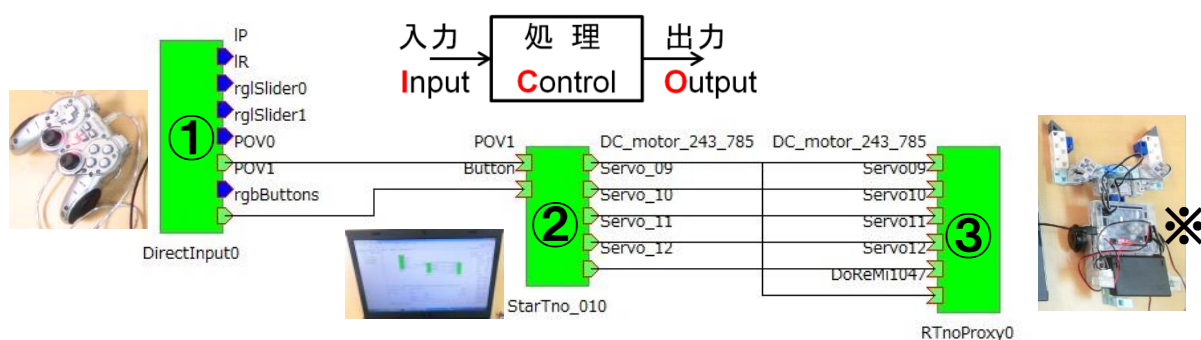
StarTno (スター・ティーノ) は, RTM(RT ミドルウェア)入門用として, RTM スタートの環境整備を目的としています. StarTno_01 では, ゲームコントローラを使い, ハンド付の移動ロボットの操作する課題に取り組みます. ロボットは, アーテック社製の Robotist を使います. 基本マニュアルでは, StarTno_01 コンポーネント群の使い方を説明します. RTM を活用した, 入力, 処理, 出力の基本システムを実現することができます.



1. StarTno_01 コンポーネント群

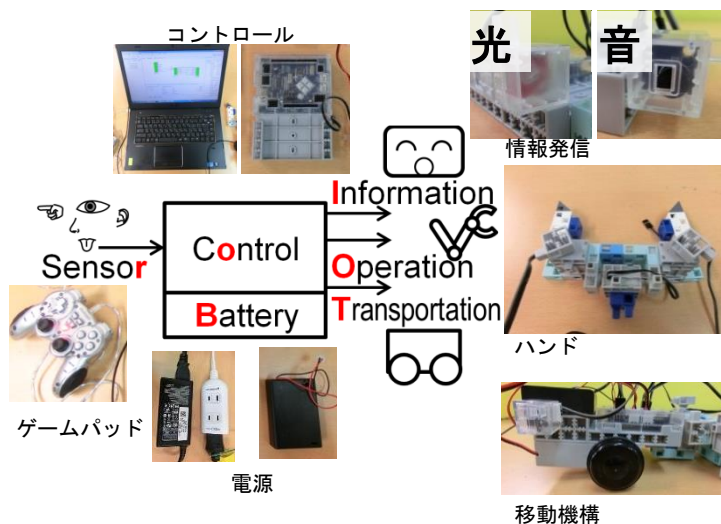
※ _00_StarTno_01.ino : StarTno 用スケッチ(Arduino 言語)[ロボット本体に書き込むスケッチ]

- ① DirectInputComp : ゲームパッド用 RTC(菅氏作)
- ② StarTno_01Comp : directinputcomp と RTnoProxyComp をつなぐ RTC
- ③ RTnoProxyComp : Arduino 用 RTC(菅氏作)



StarTno_01 コンポーネント群は, ①②③の 3 つのコンポーネントと, ※の Arduino 用スケッチから構成されています. それぞれ, 入力, 処理, 出力に対応します.

本システムで使用する「ハンド付き移動ロボット」は, ロボットの 6 大要素(SI2015_2K1-5)を持つロボットとして設計しました. ロボットの 6 大要素は, 入力, 処理, 出力のモデルを拡張し, ロボットの各要素をわかりやすく示したものです. 入力
はセンサ(Sensor), 処理は, コントロール部
(Control)と, エネルギー源のバッテリー
(Battery)です. 出力は, 情報発信
(Information), 作業 (Operation), 輸送
(Transportation)です. 各キーワード
(Information, sensoR, cOntrol Battery,
Operation, Transportation)の一部をつなぎ
合わせると, 「I, ROBOT.」となります.



2. 手順の概要

基本的な手順を説明します。手順の詳細は、後で説明します。

1. 素材を集める

- ①環境整備と StarTno_01 のダウンロード
- ②StarTno_01_組立マニュアルに従い、ロボットを組み立てる。
- ③ゲームパッド、ロボットを PC に接続する

2. 形を整える

- ①00_StarTno_01.ino(StarTno 用スケッチ)を Studuino に書き込む
- ②使用している COM ポートを調べる。(Arduino IDE 又は デバイスマネージャ)
- ③RTnoProxy.conf 中の COM ポートの指定を書き換える。

3. 組み立てる(RTM)

- ①RTM ソフトウェアの起動

Start naming service(RTC 管理ソフトウェア)の起動

RT システムエディタ(システムの構築環境)の起動

RTC(システム構築の部品)の起動(3つの RTC)

- ②システム構築

新しいシステムエディタを開く(Open New System Editor)

RTC をシステムエディタ上に配置

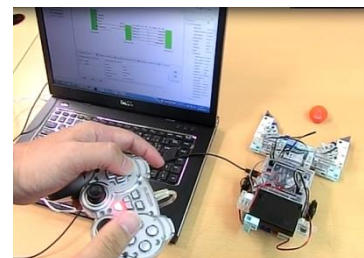
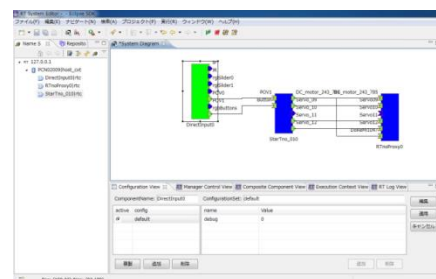
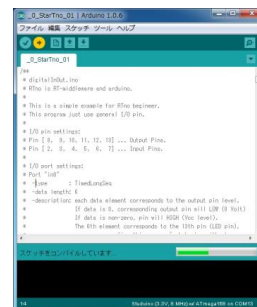
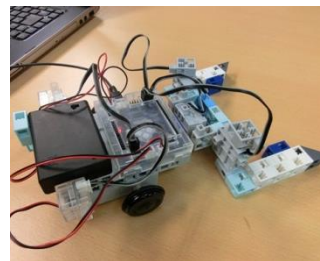
RTC を接続

- ③アクティベート(アクティブな状態にする。スイッチ ON)

左から順にアクティベート(All で行うと真ん中の RTC がエラー状態になります)

4. 使う:ロボットの操作

- ①ロボットの電源のスイッチを入れ、ゲームパッドで操作する。



基本的な手順を、「1. 素材を集める」、「2. 形を整える」、「3. 組み立てる(RTM)」、「4. 使う」の4つの過程に分けました。DIY でもものを作る手順と同じです。部品を買ってきて、部品の形を整えて、組み立て、使う。また、料理の手順と同様です。食材を買い、下ごしらえをして、火にかけて、食べる。基本マニュアルは、「すぐに使う」、「すぐに食べる」ことができるように手順を整理しました。「1. 素材を集める」、「2. 形を整える」までが、少し大変です。「3. 組み立てる(RTM)」、「4. 使う」は簡単です。次に説明する手順の詳細に従って、進めてください。詳細内容は、「03_StarTno_01 応用マニュアル」で説明します。



スター・ティノー, エピソード_1のはじまりです。

3. 手順の詳細





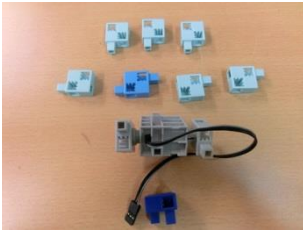

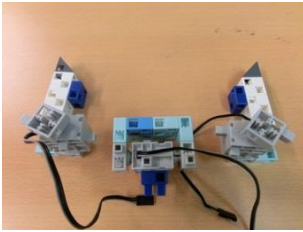
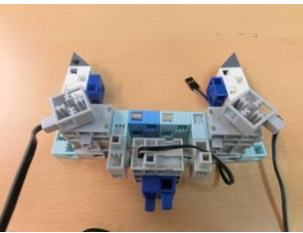

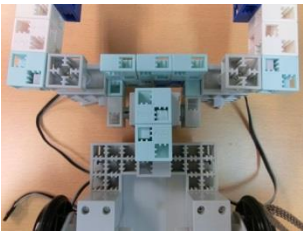
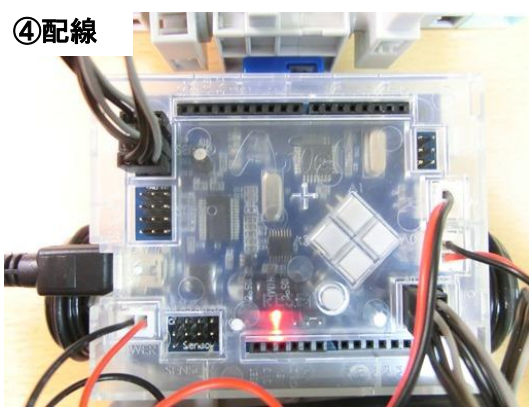
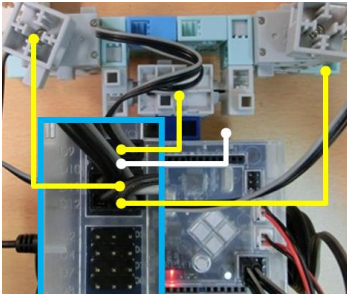
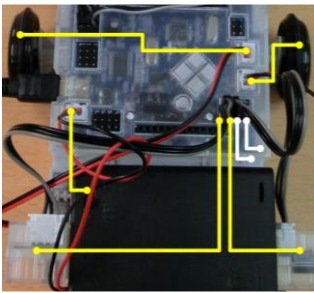
1. 素材を集める

1-①環境整備と StarTno_01 のダウンロード

それぞれの単独の環境で、動作確認を行ってください。

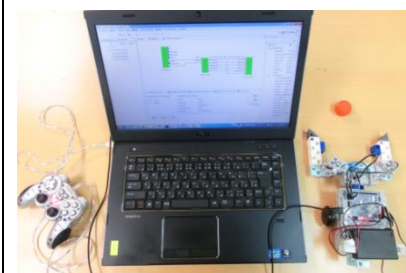
	<p>RTM 環境(http://www.openrtm.org)</p> <ol style="list-style-type: none"> ①インストール ②PC の再起動 ③OpenRTM-aist を 10 分ではじめよう(動作確認)
	<p>ArduinoIDE(https://www.arduino.cc/)</p> <ol style="list-style-type: none"> ①ダウンロード: Ver1.0.6 をダウンロードしてください ②Arduino のドライバをインストール ③サンプルプログラム Blink で動作確認
	<p>RTno(http://ysuga.net/?p=124)</p> <ol style="list-style-type: none"> ①インストールのタブを選択 ②RTno のダウンロードと展開 ③GitHUB ④RTno の準備
	<p>Studuino 環境(http://www.artec-kk.co.jp/studuino/Studuino_dl.html)</p> <ol style="list-style-type: none"> ①デバイスドライバのインストール ②Studuino ライブラリセット環境設定 ③サンプルプログラム Blink で動作確認
	<p>StarTno_01</p> <ol style="list-style-type: none"> ①_0_StarTno_01_easy_solution の中のファイルを使います。 詳細は、後で説明します。

1-②. ロボットの組立 02_StarTno_01_組立マニュアルを参照して下さい

①移動機構		ハーフ A(灰色)×1 タイヤ付き DC モータ×2 スタディーノ LED(赤) 電子ブザー ハーフ C(水色)×2 電池ボックス(単 3×3)	
②ハンドの組立 ②-1 左右ハンド		ハーフ B(青色)×2 ハーフ C(薄水色)×2 サーボモータ×2	
②-2 上下動機構		ハーフ C(薄水色)×6 ハーフ D(水色)×1 サーボモータ×1 ハーフ B(青色)×1	
②-3 ハンドの組立		ハンド左部品 ハンド左部品 ハンド上下動部品	
③ハンド+移動機構		ハンド ハーフ C(薄水色)×2 移動機構	
④配線		 PC(USB miniB)	

Arduino で代用する場合, 09, 10, 11, 12 番からサーボモータ用の信号を出力します.
 左モータは 2, 4, 3, 右モータは 7, 8, 5 でコントロールします, (02,04),(07,08)をモータドライブ IC につなぎモータの正転・逆転の制御, 3,5 の信号でスピードのコントロールができます.

1-③. ゲームパッド, ロボットを PC に接続する



ゲームパッドを PC に接続する

ロボットを PC に接続する. (USB miniB)

2. 形を整える

2-①. 00_StarTno_01.ino (StarTno 用スケッチ) を Studuino に書き込む

00_StarTno_01 ファイルを開く

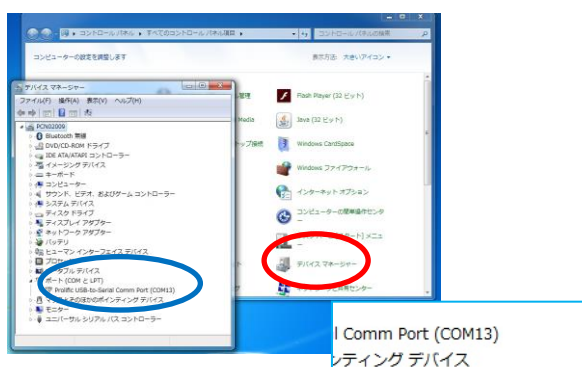
マイコンボードは Studuino を選択す

マイコンに書き込む

マイコンボードへの書き込みが完了しました。

書き込み完了

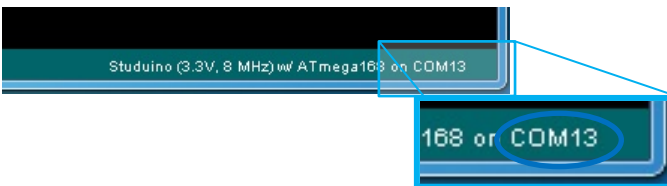
2-②. 使用している COM ポートを調べる.



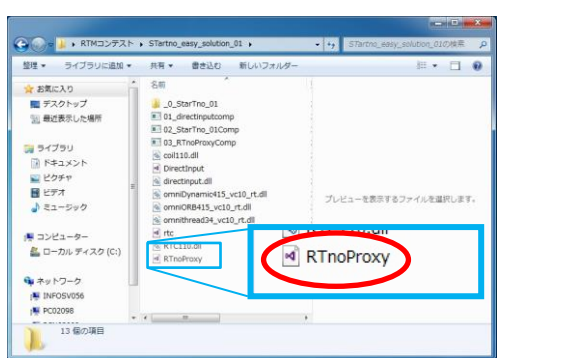
COM ポートを調べる

コントロールパネルからデバイスマネージャを開き、ポートの番号を調べる.

Arduio IDE の下段からも確認できる.



2-③. RTnoProxy.conf の中の COM ポートの指定



VC で開いた場合

メモ帳で開いた場合

COM15

COM15

上記の場合 COM15 を COM13 に変更する

3-②. システム構築

クリックして展開する

RTC を配置する

RTC をドラッグ&ドロップ

3 つの RTC を配置

ポートにポイントしドラッグすると配線が出る
接続するポートにドロップする。
内容を示す Window が開く。

RTC を接続する。

1 本 接続完了

接続確認
OK

2 本

4 本

2 本

5 本

3-③. アクティベート

Activate されると緑色になる。

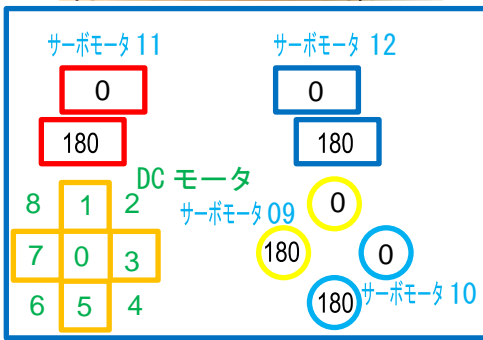
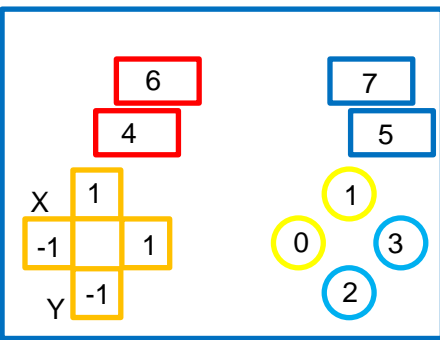
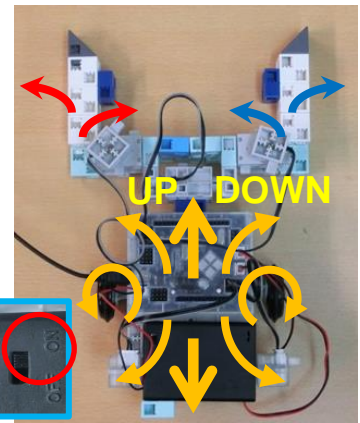
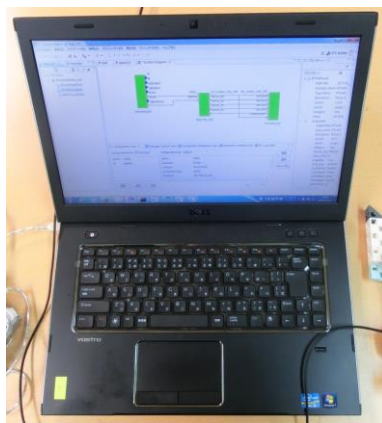
RTC を選択し、右クリックから Activate

DirectInput を最初に Activate する必要があり、
All Activate を使うと、
StarTno がエラー状態になる

入力：ポート名 データ型 → Inactive(青色 OFF) ↔ Active(緑色 ON) → 出力：ポート名 データ型
Error(赤色) [内部状態]

4. 使う：ロボットの操作

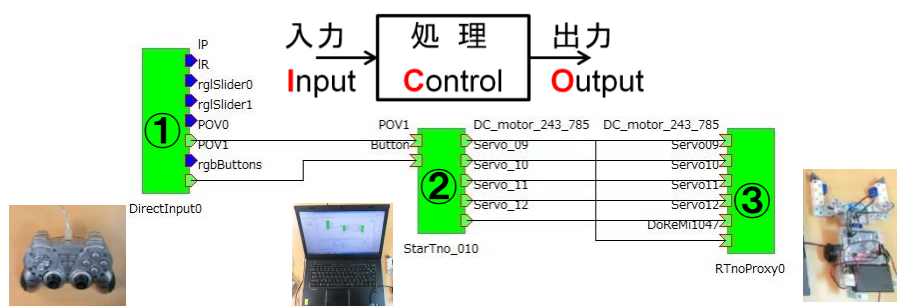
ロボット電源をオンにしてください。DC モータ, サーボモータに, 電池からの電源供給が必要です。(Studuino の仕様)



DC	2	4	3	7	8	5
1	H	L	255	H	L	255
2	H	L	255	H	L	128
3	H	L	255	L	H	255
4	L	H	255	L	H	128
5	L	H	255	L	H	255
6	L	H	128	L	H	255
7	L	H	255	H	L	255
8	H	L	128	H	L	255
0	-	-	0	-	-	0

Ser	In	Map
09	0 90 180	0 90 180
10	0 90 180	0 90 180
11	0 90 180	45 90 150
12	0 90 180	135 90 30

4. 各コンポーネントの詳細



DoReMi	Hz	Do
1. 180	1047	ド
2	1175	レ
3	1319	ミ
4	1397	ファ
5	1568	ソ
6	1760	ラ
7	1976	シ
8	2093	ド
0	No	No

① DirectInput : ゲームコントローラ用の RTC

- POV1 : 十字ボタンからの出力 (−1,0,1)
- rgbButton : ゲームパッドの各ボタンに対応 (Boolean, 0,1)

② StarTno01 : DirectInputComp と RtnoProxyComp をつなぐ RTC

- POV1 からの入力で, 2 個の DC モータをコントロールするため 0~8 の数字を出力
- rgbButton からの入力で, サーボモータをコントロールするため, 0, 90, 180 の数字を出力

③ RtnoProxyComp : Arduino 用 RTC

- DC_Motor_243_785 は, 0~8 数字を受け取る。
前進, 旋回, 後退等, ゲームコントローラの十字キーに対応したコントロールを行う。
Arduino のピン番号(2,4,3)が M1 左モータ, (7,8,5)M2 右モータに対応。(前, 後, スピード)
- Servo09~12 は, 0, 90, 180 の数字を受け取る。(0°, 90°, 180°)
各サーボモータをコントロールする。Arduino スケッチを変更して角度の調整ができる。
- DoReMi1047 は, 数字(0~8, 180)を受け, LED から光, 電子ブザーからドレミの音階を出力。
0 または 180 を受け取ったとき, 音は「ド(1047Hz)」を出力する。

注 1 : DC モータ, サーボモータは, 電池からの電源供給が必要です。音を確認する場合, 電池のスイッチを切ると, 音が聞き取りやすくなります。

注 2 : 音を出すと, ピン番号 3 からの PWM 出力ができないため, 左車輪のスピードコントロールができなくなります。(Arduino の仕様)

注 3 : 電池からの電流容量が足りない場合, サーボモータが振動する等, 動作が不安定になります。パワフルな電池を使用してください。

→各コンポーネントの詳細については, 「03_StarTno_01 応用マニュアル」に書きます。