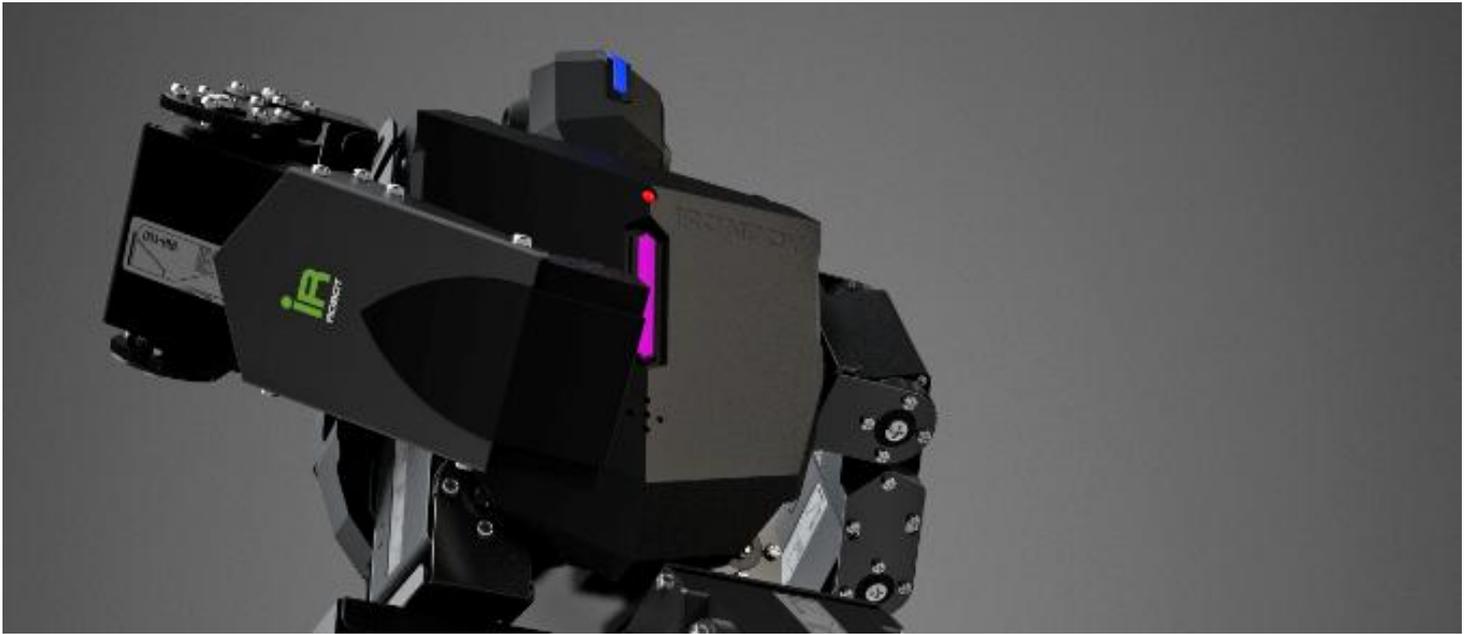


Arduino Compatible Open Source Humanoid -----

アイロンボーイ ユーザーマニュアル



Robotics for everybody。

01 はじめに	01
製品紹介/特徴	01
仕様	03
安全関連	04
02 ロボットの構成	05
構成品目	05
部分別説明	05
関節の構造と名称	08
03 ロボットコントロー	09
主な機能	09
部分別説明	11
04 管理プログラム	12
使用環境	12
インストールする	13
通信設定	14
画面構成/説明	15
ゼロと基本パラメータ	22
基本姿勢の設定	25
モーション編集および	
ライブラリ	26
仮想プロポ	30
リモコンで操縦	34
05 ロボット操縦 APP	36
インストールする	36
設定する	36
通信接続	40
操縦	40
通信の切断	43
06 ユーザープログラミ	44
Arduino と IRduino	44
Shield	46
API とライブラリ	47
07 付録	54
バッテリー管理および充電	54
LED の色制御表	55
Piezo 音番号札	57
モーションデータ関節補正	59
モーションライブラリ	61
ロボット取付けスタンドの	
組み立て	65

1 はじめに

1.1 製品の紹介と特徴

この度はアドゥーイーノ互換ヒューマノイド アイロンボーイ IRH-100 を購入していただきありがとうございます。オープンソースのハードウェア アドゥーイーノ (Arduino) を搭載したヒューマノイドプラットフォームです。安全にご使用頂く為に、本取扱説明書を必ずお読みください。

アイロンボーイ IRH-100 は、16 個のデジタルサーボモータ、専用コントローラ (IRC-24) とメタルブラケットフレームとその他のハードウェアで構成されています。

IRH-100 は、メインコントロールボード (IRC-24) にアドゥーイーノシールドを簡単に装着することができる個別の専用アドゥーイーノボード (IRduino) を標準装備しています。IRduino ボードは、市販されているアドゥーイーノシールドと互換性があり、市販されている様々な種類のアドゥーイーノシールドをユーザーが購入して簡単に装着できます。

IRH-100 では IRduino API とライブラリを提供して、ユーザーの想像に基づいて様々な種類のロボットタスクを実装することができます。IRH-100 は、オープンソーススペースのアルディーノシールドのオプションを使用して、さまざまなロボットタスクを組み合わせることができるメリットがあります。

本製品は、組立完成済みとなっていますので、ご購入直後からソフトウェアプログラミングの学習に集中することができます。

【オールインワンマザーボード】 IRH-100 のメインコントロールボード (IRC-24) にはロボットを駆動するための様々な機能を有しています。ジャイロセンサー (6 軸加速度)、Bluetooth 4.0 モジュールが標準装備されており、また専用 IRduino ボードが装着されており、様々なアドゥーイーノシールドとの接続が容易にできます。

【IRduino ボード】IRduino は ATmega32u4 を基にしたマイクロコントローラであって、IRH-100 のマザーボードに簡単に装着が可能です。20 個のデジタル入力/出力ピンを内蔵しています。(7 つは、PWM 出力、12 個はアナログ入力) また、マイクロ USB、電源ジャック、ICSP ヘッドおよびリセットボタンを標準装備しています。

そしてポートの拡張性をより効率的に運用するために、ハードウェアのシリアルとソフトウェアのシリアル通信の切替をすることができます。



【サーボモータ】 IRH-100 は、プログラム可能なデジタルサーボ（IRS-440）を16個標準装備しています。



【トータルマネージャー】 Total Manager Software は、モーションセッティング、モーションタスクの設定、LED/ピエゾ制御とセンサーの状態を監視することができるユーザーインターフェース専用の管理ソフトウェアです。モーションセッティングでは、逆運動学（Inverse Kinematics）UI と左右の動きの同期機能で、ユーザーだけのモーションを直感的で簡単にプログラミング化が可能で、多くの時間を節約することができます。



【特徴】

- 工場出荷時、組み立て完成済み
- アドゥーイーノ接続用互換ボード付属
- 6軸ジャイロセンサー搭載（3軸ジャイロセンサー + 3軸加速度センサー）
- Bluetooth 4.0 通信モジュール実装済み
- スマートフォンからの無線コントロール可能
- 専用ロボットスタンドと専用キャリングバッグ付属
- 基本モーション 20個+ 80追加モーション インストール済み

[IRH-100 アイロンボーイ]

制御システム	PWM制御 (Pulse Width Modulation) 、1500usec Neutral
自由度	16 自由度 (最大 24 自由度の拡張可能)
動作電圧	5.5V~7.0VDC (専用 Ni-MH 5Cell Battery 推奨)
ロボット操縦方法	Bluetooth 4.0 BLE 通信を介して携帯電話の制御 (アンドロイドのみ対応) 赤外線リモコン (リモコンは付属していません)
プライマリコントローラ (IRC-24)	16bit、ATXmega128、Firmware Upgrade 可能 1MB data memory (基本 100 モーション提供+消費創作モーション 100 保存) IRduino ドッキングポート 6 軸ジャイロ/加速度センサー、Bluetooth 4.0 内蔵
管理ソフトウェア	Total Manager Software : Motion Feedback、Motion Capture and Inverse-Kinematics (逆運動学) 機能など。 モーションと 状態スク設定 LED /ピエゾ 制御、センサのモニタリング
慣性制御センサー	3 軸加速度 + 3 軸ジャイロセンサー (メインコントローラに内蔵)
バッテリー	1,500mA Ni-MH Battery 5 Cells
アドゥーイーノ互換	専用アドゥーイーノボード (IRduino) 組み込み (市販の様々なアドゥーイーノシールドに対応) Arduino API、サンプル、ライブラリ、提供 20 種類のデジタル入力/出カピン (PWM 出力の 7 つの、Analog 入力 12 の) Micro USB コネクタ、電源ジャック、ICSP ヘッド、リセットボタン 32KB Flash memory、2.5KB SRAM、1KB EEPROM と 16MHz Clock Speed
LED 制御	プログラム可能な 3 color LED (胸部)、1x Blue LED (Bluetooth Link 表示 : 頭部)
動作温度	-20°C~+ 60°C
サイズ	170 X 99 X 335mm
重量	1.3kg
標準アクセサリ	専用ロボットスタンド / 専用ロボットキャリングバック

[IRS-440 サーボモータ]

CPU	32ビット マイコン、 4096 解像度、 プログラム可能
動作電圧	4.8～6.5VDC
動作温度	-20°C～+ 60°C
速度	0.2sec at 6.0V
ストールトルク	8kgf.cm at 6.0V
スタンディングトルク	12kgf.cm at 6.0V
IDLE 電流	20mA
動作電流	200mA /無負荷時
デッドバンド	3usec
パルスの各	1°/ 10usec
最大動作角度	-65°+ 65°
サイズ	40 x 20 x 47mm
重量	55g

1.3 安全の為の注意事項

本製品は、組立完成品で、下記の注意に反した使用による、故障や事故等についてはいかなる保証も致しかねます。注意を無視して誤った取扱いをした場合、人的障害や物的損害が生じる危険があります。

- 安全の為に使用前に、ユーザーマニュアルを十分に熟知して、以下の安全注意事項を必ず守ってください。
- 当製品は 15 歳以上を対象としています。
15 歳以上の方でも製品の取り扱いに不慣れな方は販売店や経験者に指導を仰いでください。
- 本商品をお子様には使用させないでください。また、いかなる場合も幼児や子供の触れる可能性のある場所に置いてはいけません。
- 本製品は防水ではありません。湿気の多い所や水のかかる所では絶対に使用 / 保存しないでください。
- 衝撃に弱いので破損等に注意してください。また直接日光に長時間照射すると変形や変色する事があります。
- 製品保証期間はご購入日から 1 ヶ月となります。(各部の消耗品等は除く)
- ニッケル水素電池パックは使用頻度に応じて寿命が流動的な消耗品であり、サーボギアは、一定の時間を使用すると、遊びが徐々に大きくなりますので消耗品となります。
- 各注意・説明に反して誤った設定や不適切な取扱いで起きた結果については、当社は一切責任を持ちません。
- 免責事項：製品の性格上、お客様が当製品をご使用になって起きました結果につきましては責任を負いかねます事を予めご了承ください。

下のマーカーは注意事項を無視して発生する可能性のある潜在的な危険に説明します。



警告：死亡または重傷の危険



注意：けが、または製品の損傷の可能性



警告

-
- ・ナイフ/千枚通し/ドリルなどの不要な工具で任意の分解/組立禁止
 - ・夏場の自動車の中など直射日光や高温でのロボット保管および操作は行なわないでください。
 - ・専用電池パックのショート、物理的な損傷、浸水、加熱などのは行なわないで下さい。
 - ・充電ケーブルは取付、取り外し時に損傷しない様に気を付けてください。ケーブルへの加工・使用時の加熱/過負荷に気を付けてください。火災や感電の恐れがあります。
 - ・メインコントローラ / サーボ の任意の分解 / 改造は火災 / 感電の原因となりますので絶対に行なわないでください。
 - ・湿気の多い場所や結露環境で使用禁止。感電 / 漏電による火災の防止
 - ・ロボット動作時は顔や手を近付けないでください。ロボットの各関節部に指や手を挟み込む危険があります。



注意

-
- ・ご使用時には十分なスペースを確保してください。また机などの高所からの落下や固い床での使用時の転倒は、思わぬ故障や破損につながりますので気を付けてください
 - ・海外での使用時は使用する地域または国により国別許認可や事前の確認が必要となる場合がございます。
 - ・充電器 / バッテリーパックの取り外し時にはコードが破損しないように注意してください。コネクタやコード部はショートしないように気を付けてください。感電の恐れがあります。

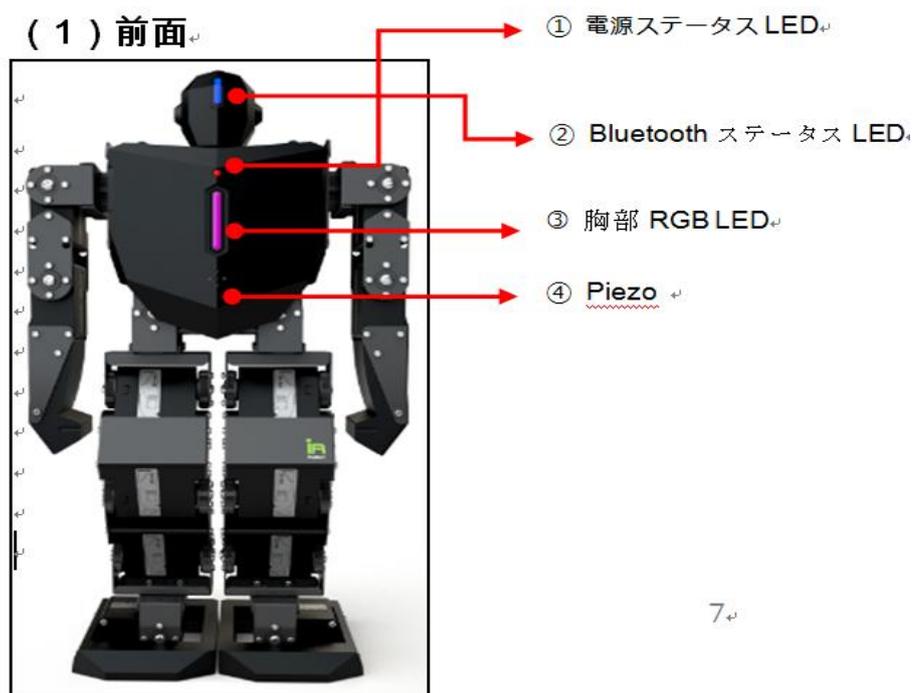
2 ロボット (IRH-100)

2.1. 構成品目

IRH-100 本体	IRduinoボード (ア두ゥーイーノ互換ボード)	Micro B type USBケーブル (コントローラ接続用)	バッテリー (1500mAh、6V NiMH)
			
充電器	ロボットスタンド	キャリングバック	日本語マニュアル
			

*キャリングバック内の内箱は再利用することで、製品の移動時のロボットと部品を保護するのに役立ちます。

2.2. 部分別説明



① 電源ステータス LED

胸部丸型赤色 LED では電源とバッテリーの状態を表示します。

電源の状態	LED の表示
電源スイッチ OFF	OFF
電源スイッチ ON	ON (RED)
バッテリー不足	残量 20%以下 (点滅を繰り返し)

② Bluetooth ステータス LED

コントローラに内蔵された Bluetooth モジュールの電源または接続状態を表示します。

Bluetooth 接続の状態	LED の表示
Bluetooth の電源 OFF	OFF
Bluetooth の電源 ON	ON (WHITE)
Bluetooth の接続	ON (BLUE)

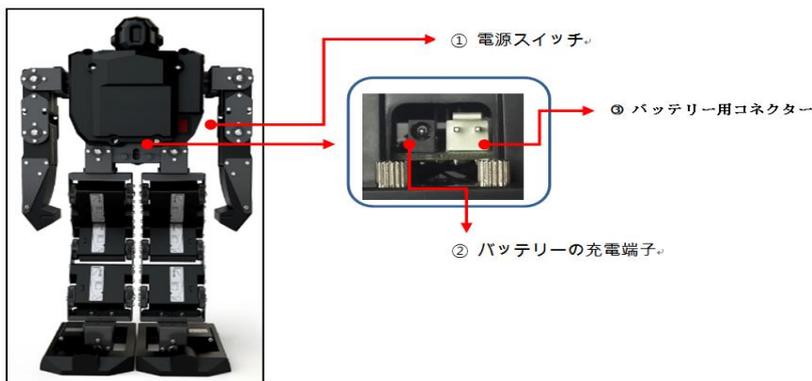
③胸部 RGB LED

様々な色を表示することができる RGB LED です。スマートフォンアプリを利用して、操縦したり、モーション実行中、様々な色表現の指定が可能です。

④ PIEZO

ブザー音が出力され、メロディーを作ることができる Piezo です。モーション実行中、さまざまな音の指定が可能です。

(2) の後面



① 電源スイッチ

タンブラー型スイッチで、上げれば、電源がオンになり、下げると電源がオフになります。

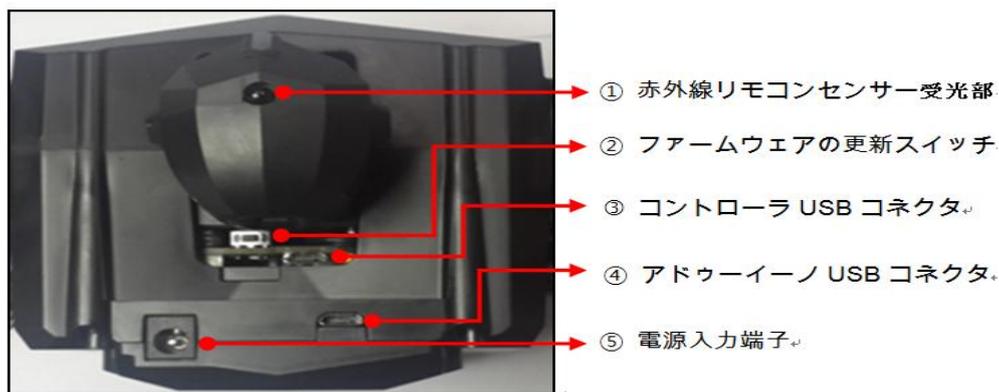
② バッテリーの充電端子

付属の充電器を接続すると、自動的に充電をします。充電時間は、最大 2 時間程度です。

③ バッテリーの接続端子

付属の専用バッテリー (1500mA、6V NiMH) をコネクタに合わせて接続して電源を供給します。

(3) 上面



① 赤外線リモコンセンサー受光部

赤外線リモコンは含まれていません。適合する汎用型赤外線リモートコントロールを介してアイアンボーイを操縦することができます。ただし、事前にリモコンコード表に希望のボタンとモーションプログラムをロボットに保存する必要があります。

② ファームウェアの更新スイッチ

コントローラ駆動プログラムを更新しようとするときは、電源が OFF の状態で、このボタンを押しながら電源を ON にすると、更新モードに入ります。

③ コントローラ USB コネクタ (Micro-B Type)

ロボットのメインコントローラと PC (トータルマネージャーロボット制御用プログラム) を接続するときに使用し、ロボットとトータルマネージャープログラムを接続して、ロボットを設定したり、モーション制作と編集、モーションダウンロードするときに使用します。

④ アドゥーイーノ USB コネクタ (Micro-B Type)

IRduino と PC を接続するときに使用します。

⑤ 電源入力端子

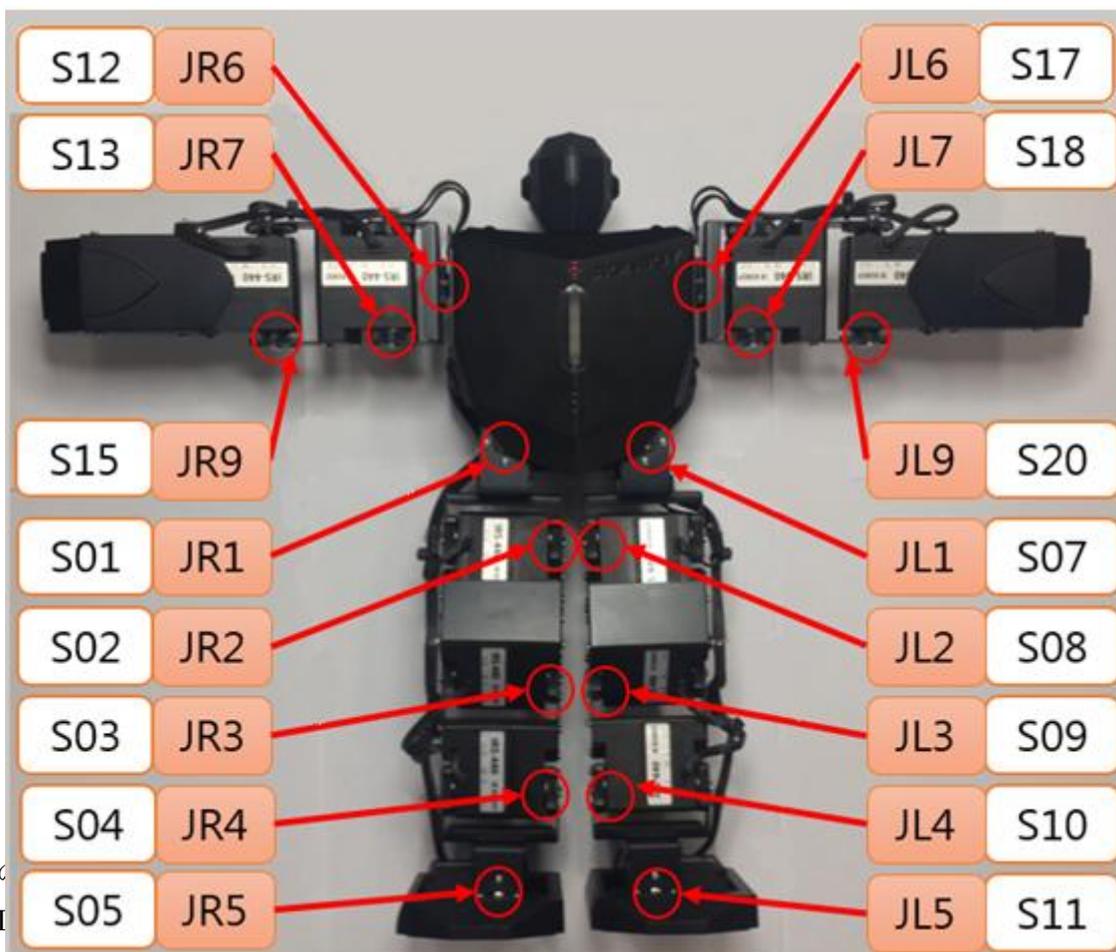
IRduino を独立して使用するときは、電源を供給している端子、

入力電圧：6V～12V

※この端子に電源を印加しても Main コントローラやロボット本体には電源が印加されません。

2.3. 関節各部の構造と名称

IRH-100 は、下図のような関節構造と名称があります。専用コントローラは、サーボをチャンネルごと的一对一に対応するため、各関節の位置にあるサーボは、指定されている位置に合わせてサーボ番号があるコントローラサーボチャンネルを選択して接続が必要です。



(1) 関節の

- JL6n : ロボットの左関節の n 番目の関節
- JR6n : ロボットの右関節の n 番目の関節

(2) 関節数

関節番号 (n)	関節の位置
1	股関節左右回転
2	股関節の前後回転
3	膝関節
4	足首の関節の前後の回転
5	足首の関節左右回転
6	肩関節の前後の回転
7	肩関節左右回転
9	肘関節

(3) サーボ番号

コントローラは、最大合計 24 個のサーボチャンネルが独立して動作可能です。IRH-100 では、関節が 16 ヶ所あり、16 個のサーボチャンネルへ割り当てられています。関節をユーザーが追加して制御しようとするときは、割り当てされていない空きチャンネルを有効にして使用することができます。

[空きチャンネル推奨関節位置]

チャンネル番号	関節の位置
S00	右股関節垂直軸回転
S06	左股関節垂直軸回転
S14	右腕肩関節の回転
S16	右腕の手首の関節やグリップ装置
S19	左腕の肩関節の回転
S21	左腕の手首の関節やグリップ装置
S22	頭左右回転 (垂直軸 Pan)
S23	頭上下 Tilt

3 ロボットコントローラ

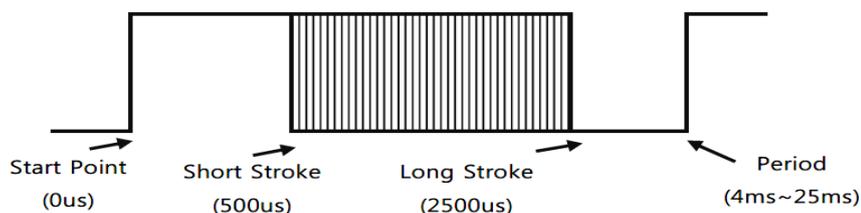
IRH-100 を制御するためのシステムです。サーボモータ制御、6 軸ジャイロセンサ、ブルートゥース 4.0 (BLE) モジュール、大容量のモーションデータメモリ、LED ディスプレイとピエゾの出力は、バッテリーの充電接続、IRduino 専用接続ポート、IR 受信センサーなどで構成されています。

3.1. コントローラの主な機能

コントローラは、ロボットを制御するために、次のような機能を提供します。

(1) サーボ制御 (24ch)

24 個のサーボパルス制御方式を提供しています。RC 用サーボなどに使用する Pulse (PPM)規格を使用します。



PPM (Pulse Position Modulation) Compatible [Radio-Control Servo Pulse Mode]

(2) モーションコントロールと管理

コントローラは、モーションの作成、編集、テスト、実行、停止、トルク制御などを行ないます。ユーザーによる操作プログラム 100、固定動作プログラム 100 など様々な機能の実行が可能であり、モーションデータをロボットに保存して、独立し実行させることが可能です。

ロボットに保存されたモーションファイルは、PC管理プログラム上でインポートが可能で、動作中のモーションを簡単に書き換え変更することができます。

動作番号	プログラム表示	説明
0~99	MOTION_USER_00~_99	ユーザーによる操作プログラム 0~99(外部メモリ領域に格納)
100~199	MOTION_LIBRARY_00~99	固定動作プログラム 0~99 (内部メモリ領域に保存)
200	MOTION_TORQUE_OFF	全サーボのトルクを解除
201	MOTION_USER_TEST	ユーザーの操作の作成中、テストの目的で一時的に実行する動作
255	MOTION_STOP	動作停止 (既存の実行中の動作を停止させる)

(3) ユーティリティサポート

コントローラ駆動プログラムをコンピュータを介してアップデートが可能であり、モーションコントロールなどさまざまな機能をサポートします。

(4) Bluetooth を利用したリモートコントロール

Bluetooth ブルートゥース 4.0 BLE モジュールが内蔵されて、スマートフォンや Bluetooth モジュールを搭載したノートパソコン、PC などでも利用可能です。Bluetooth を利用して操縦可能な Android の専用アプリを提供しています。

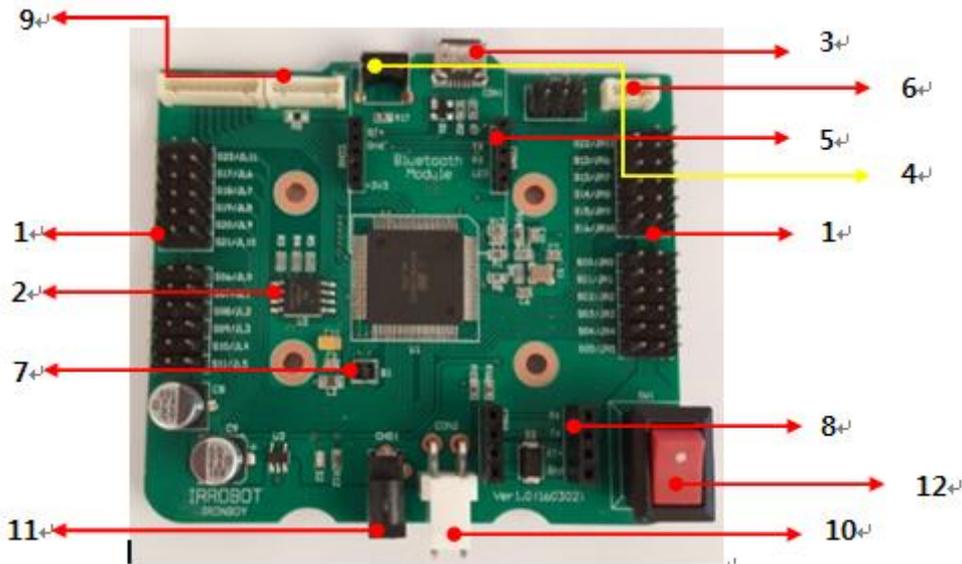
(5) IR (赤外線) 方式を利用したリモートコントロール

汎用型赤外線リモコンを利用して、モーションのリモートコントロールを実行をすることができます。

(6) 慣性センサを用いた姿勢の傾き測定

3 軸ジャイロセンサと 3 軸加速度センサーの 6 軸慣性センサーが搭載されおり、ロボットの姿勢（傾き）を Roll と Pitch の値で測定しています

3.2. コントローラ部の説明



(1) サーボ制御ポート (24ch)

最大 24 個のポートをサーボパルス方式で制御します。

(2) モーションデータ保存メモリ

ユーザーによる操作プログラム 100、固定動作プログラム 100 が格納されているメモリです。

※ 固定ライブラリ 100 個のうち、工場出荷時ゼロ設定がされているのは 20 種類のモーションプログラムであり、残りの 80 個のモーションプログラムは、ユーザーが指定されたモーションにゼロ調整をして使用ください。

(3) Micro-B Type USB ポート (トータルマネージャー接続用)

コントローラ駆動プログラムをコンピュータを介してアップデートが可能であり、トータルマネージャーソフトウェアと連動して、さまざまな機能をサポートします。

(4) コントローラ駆動プログラム更新モード専用スイッチ

通常時電源が OFF の状態で、コントローラ駆動プログラムを更新を行なう場合、このボタンを押しながら電源を ON にすると、更新モードに入ります。

(5) Bluetooth モジュール

ブルートゥース 4.0 BLE モジュールが内蔵されて、スマートフォンや Bluetooth モジュールを搭載したノートパソコン、PC などでも利用可能です。Bluetooth を介して操縦可能な Android の専用アプリを提供しています。

(6) 頭部内蔵センサーボードの接続コネクタ

赤外線リモコンセンサーと Bluetooth 状態表示 LED の制御信号コネクタが接続されています。

(7) 慣性センサー

3 軸ジャイロセンサと 3 軸加速度センサーの 6 軸慣性センサーが搭載されています。

(8) IRduino 装着コネクタ

IRduino 用装着コネクタ。

(9) 胸部ボードの接続コネクタ

電源表示 LED、RGB LED、ピエゾが搭載されている胸部ボードの信号が接続します。

(10) バッテリーコネクタ

ロボットのバッテリーを接続するコネクタです。

(11) 充電器のコネクタ

ロボットにバッテリーが装着された状態で充電する為の接続するコネクタです。

(12) コントローラー用電源スイッチ

タンブラー型スイッチ式 上げれば、電源がオンになり、下に私のために電源がオフになります。

4 ロボットの管理プログラム

(Total Manager Software)

IRH-100 は、コントローラと USB 接続通信に「Total Manager」と呼ばれる PC 用プログラムを使用して、ロボットの動作、モーションデータ管理とテスト、ロボットのセッティングパラメータの変更などを行ないます。

4.1. 使用環境

システム要件に合わせてコンピュータシステムを選択して、「Total Manager」と呼ばれるプログラムをインストールします。

その後、ロボットの電源を接続し、USB ケーブルをコンピュータに接続します。(Micro B type USB ・ 13 ページ 3 番を参照)

適切な COM ポートを確認し、通信の設定がされると、「Total Manager」を実行して、コントローラとの通信を接続します。

ロボットの電源を ON にすると基本姿勢に復帰する過程で、ロボットが動くことがありますので、ご注意ください。

ロボットを ON にすると、ロボットのバッテリーの残量が残っていない場合、USB Com Port 認識ができない場合がありますので、接続前に充電を必ず行なってください。

[Total Manager のシステム要件]

▶OS : Windows XP Service Pack 2 以上/ Vista / 7 (32 / 64bit) / 8 (32 / 64bit)

▶800MHz 以上の 32 ビット (x86) または 64 ビット (x64) プロセッサ

▶512MB 以上のシステムメモリ

▶空き容量が 500MB 以上のハードディスク

※上記の要件を満たしていても、接続する PC の環境により動作しない場合がございます。

4.2. インストールする

Total Manager を以下のようにインストールして使用します。

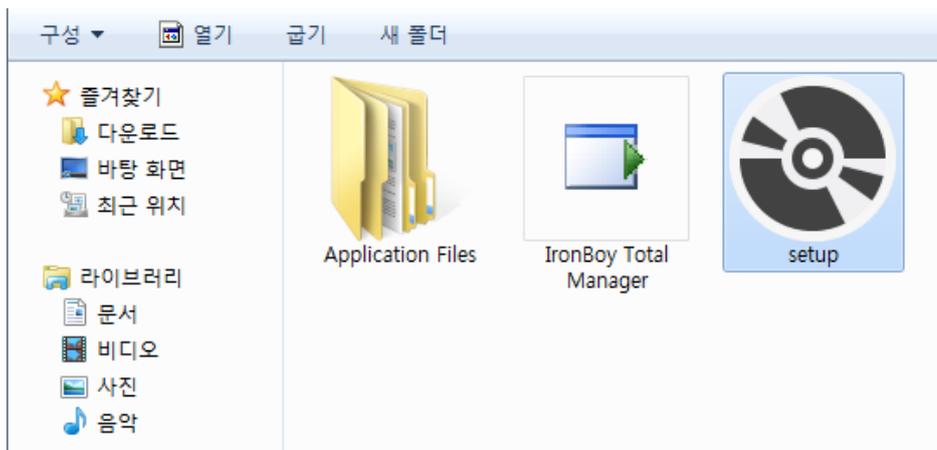
(1) ダウンロード

以下のアドレスより Total Manager のインストールファイルをダウンロードします。

<http://www.irrobot.com/humanoid-data>

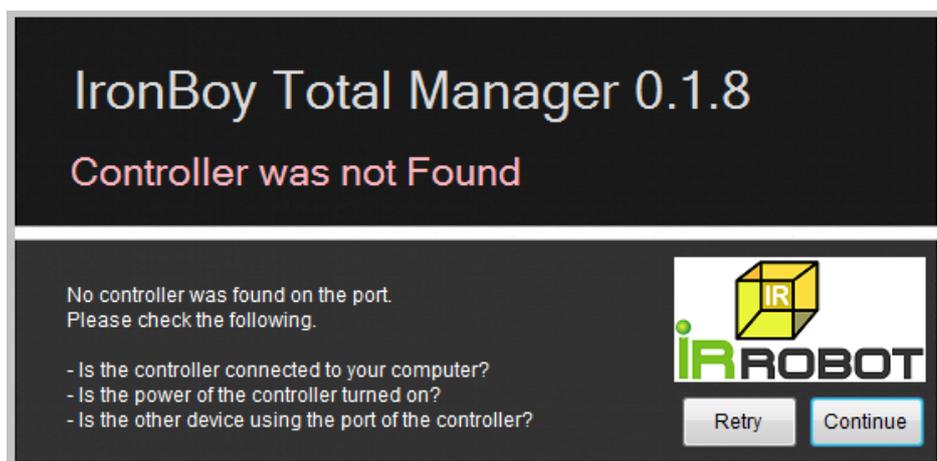
(2) インストールファイルを実行

ダウンロードしたファイルを解凍して開いて次のように **setup.exe** を実行します。



(3) 初めての画面

インストールが正常に実行されると、以下のようにプログラムが起動されます。



※2017年5月現在の画面になります。

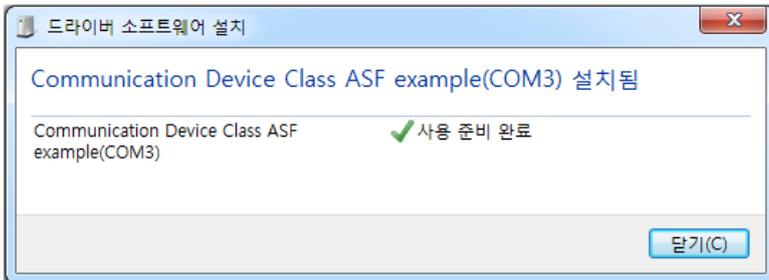
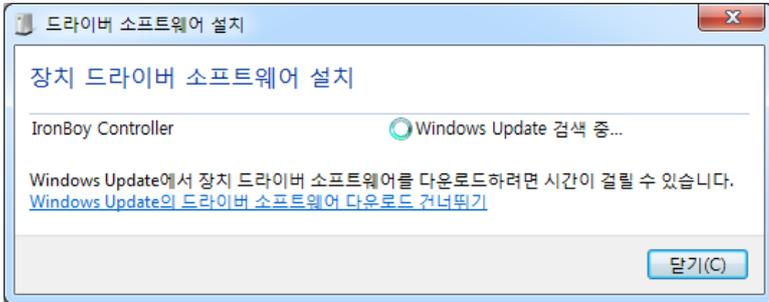
製品の品質向上や改善の為、予告無く変更される場合がございます。

4.3. 통신설정

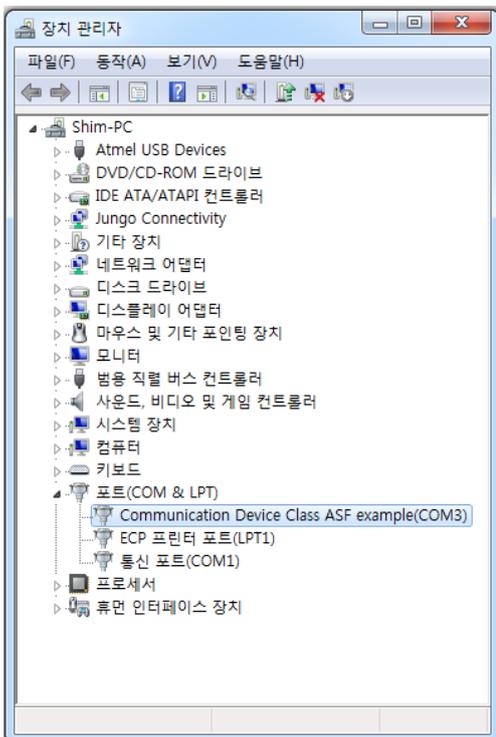
IRH-100을 PC에 연결하여 프로그램을 실행하면, 위의 첫 번째 화면이 나타납니다.

다음에, 로봇의 전원을 ON하고, 다음에 따라 실행합니다.

- (1) IRH-100을 PC에 처음 연결할 때, 인터넷 연결을 통해 자동으로 장치 드라이버를 설치합니다.

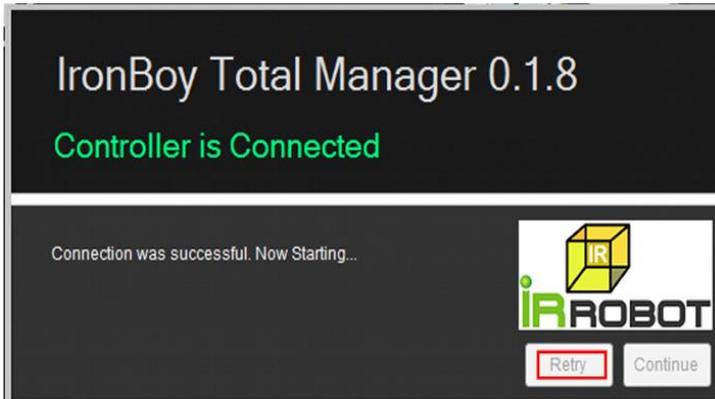


- (2) 하기와 같이, 제어판의 장치 관리자에서 COM 포트를 확인합니다. (화면은 COM3에 할당되어 있습니다.)



(3) プログラムを実行し、上記の最初の画面で、[Retry]ボタンを押します。

(4) IRH-100 がコンピュータ上で認識されている場合、自動的にプログラム上で COM ポートを認識します。

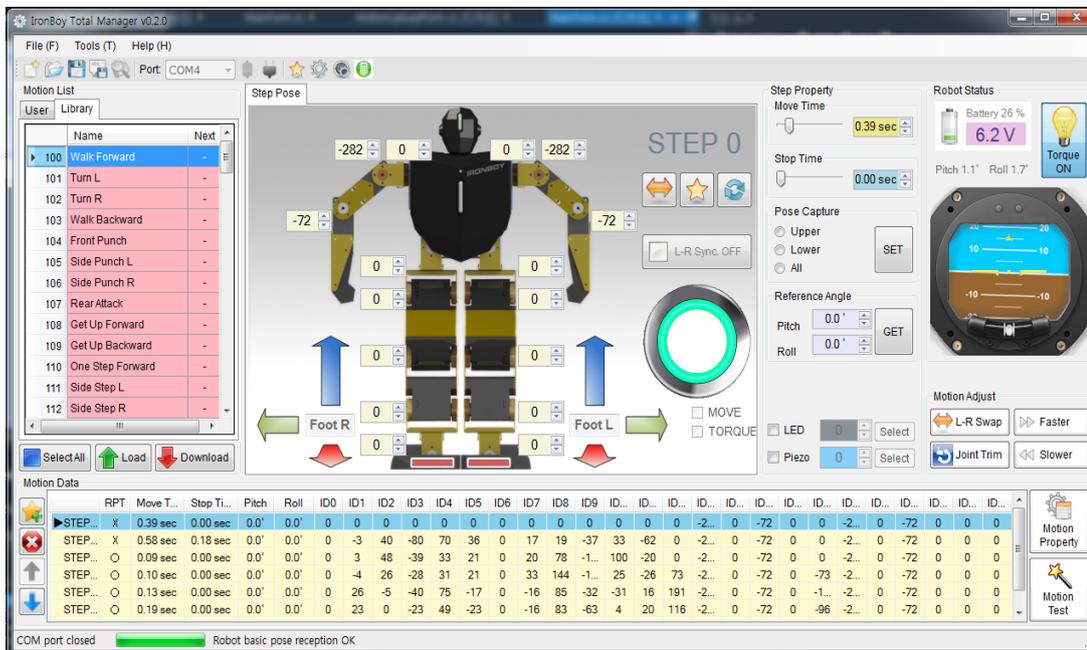


(5) COM ポートが正しく指定された場合は、メニューバーの下のボタンを押すと、PC に接続された COM ポートのリストが表示されます。確認の為、COM ポートを選択します。(例は COM3 です。)



必要に応じて通信接続を分離するアイコンを使用します

4.4. 画面構成と各部の説明



(1) メニュー

① File

- **New** : 新しいモーションデータの生成

新しいモーションプログラムの作業環境を作成します。現在のプログラムの作業環境は削除されます。必要に応じて新しい作業環境を作成する前に、前のモーションの保存を行なってください。

- **Open** : モーションファイルを開く

作成済みのモーションデータファイルを指定して、モーション作業環境をロードします。

- **Save** : モーションファイルの保存

編集中のモーションデータファイルをデータとして保存します。

- **Save As** : 別の名前でのモーションファイルの保存

現在のモーションデータファイルを別のファイルに保存します。

② Tools

- **Robot Basic Pose**

ロボットの基本姿勢編集ウィンドウを起動します。

- **Robot Basic Setting**

ロボットの原点と基本データ設定画面を起動します。

- **Robot Control**

ロボットの仮想動作画面を実行します。

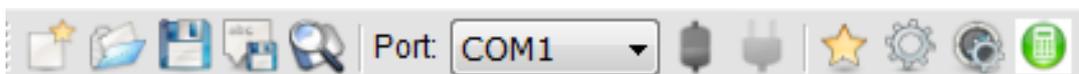
- **IR Remote Setting**

赤外線リモコンの認識コードを実行モーションと連携情報を編集する画面を表示します。

- ③ **Help** : 製品の関連情報とヒントです。

(2) ツールバーおよびCOMポートの接続

簡単に実行することができるツールバーと通信ポート指定して接続/解除を行うことができるメニューです。



① ツールバー



新しいモーションデータの生成



モーションファイルを開く



モーションファイルの保存



モーションファイル名前を付けて保存する



ロボットの基本姿勢の編集ウィンドウを実行する



ロボットのゼロ点調整と基本データの設定画面を実行する



ロボット仮想動作画面を実行する



赤外線リモコンのモーション連携情報編集画面を実行する

② COMポート接続



通信ポートの自動検出と接続

Port: **COM1** ▼

通信ポート検索と指定



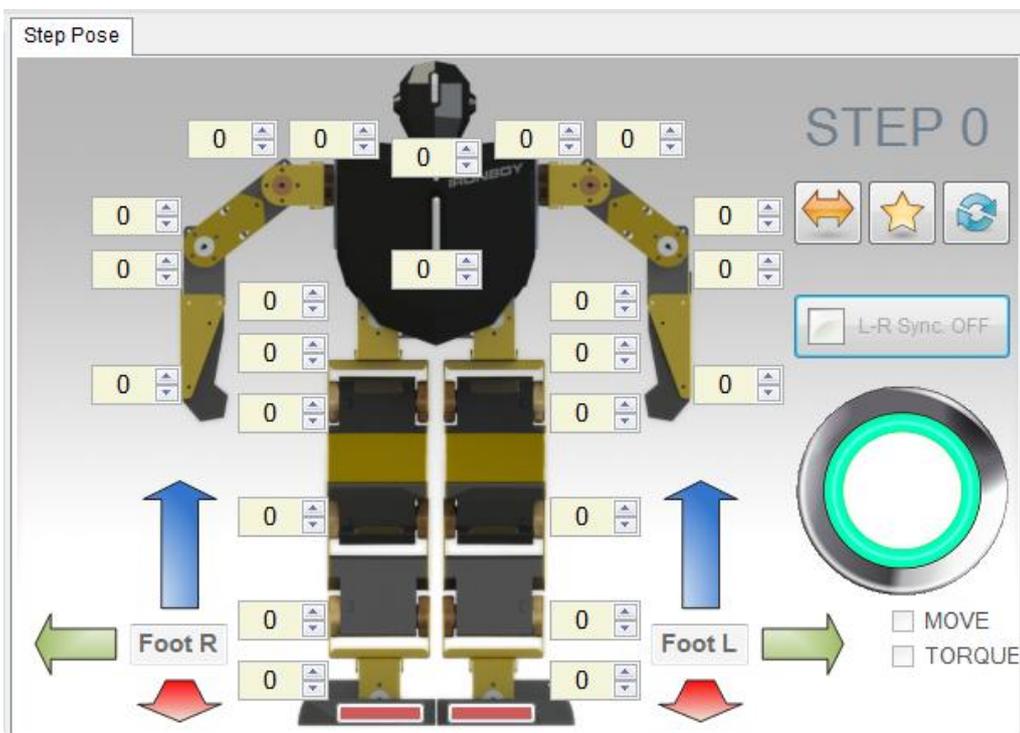
通信ポート接続の接続



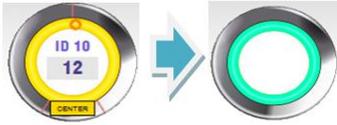
通信ポートの接続解除

(3) ロボットの各関節制御画面

ロボットの各関節のサーボを制御して、状態を表示します。サーボ位置、位置保持、トルク除去、逆運動学的関節制御、個別サーボ位置調整、モーション編集ツールなどで構成されています。この画面は、ロボットの現在の状態または現在のモーションのステップを表現します。



ロボットの各関節のサーボの値を示します。スクロールボタンを押すか、マウスのスクロールを利用して、-511 ~ +511 の値の中で調節可能です。直接数字を入れるときは、[Enter]キーを押したときに適用されます。



各関節のサーボ位置の値をクリックすると、シアンが黄色になります。サーボ位置

の値を変更すると、中央の赤い円形をマウスでのドラッグを使用して値に変換することができます。黄色の線に沿って引くことによる急激な値の変化にロボットがバランスを失うことができます。

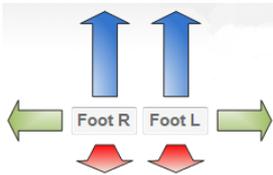
Center ボタンを クリックすると、中央値である「0」にすぐに変更されます。

MOVE

チェックがされているときに表示されたサーボ位置が現在のサーボの位置と同期されます。

TORQUE

チェックがされているときに、サーボのトルクが有効になります。



ロボットの足を基準に、X、Y、Z のビャンヒャンに分け、マウスのドラッグを使用して、逆機構学的に変更が必要な脚サーボの位置の値を全体的に変更します。これを利用すれば、モーション作成時の姿勢の表現は非常に便利です。

STEP 0

ロボットの関節制御画面が表示される現在のモーションステップ番号を表します。



モーター設定値左右切り替え



基本姿勢の読み込みと基本姿勢の修正



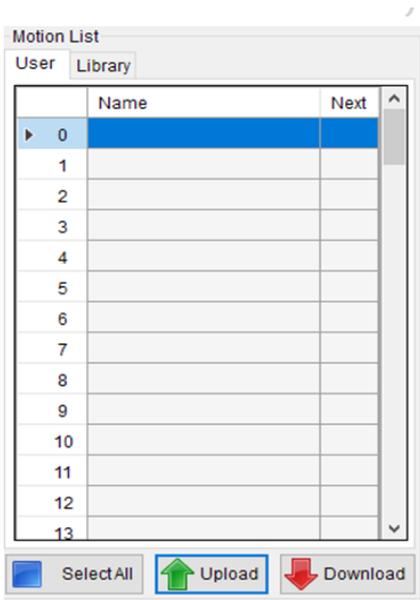
変更前に戻す



Sync On 時、左右対称の同じモーターを同じ値に制御します。

(4) モーションリスト

ユーザー領域 100 の、ライブラリ領域 100 のスペースを持って複数選択、アップロード、ダウンロードが可能です。



※ モーションライブラリ取得する方法：
 購入時は、[ライブラリ]タブにモーションライブラリは保存されていません。ロボットに保存されたモーションのライブラリを読み込みのために、「Select All」ボタンを押した後、「Upload」ボタンを押してください。このとき誤ってSelect AllボタンをクリックしてDownloadボタンを押すと、ロボットに保存されたモーションライブラリが消去されてしまうので注意が必要です。モーションライブラリが削除してしまった場合は、WEBサイトからモーションライブラリをPCに呼び出し編集し、ロボットへの保存が可能です。

Motion List

User Library

ユーザー領域 100 の、ライブラリ領域 100 の領域を選択することができます。

Select All

ライブラリ全体の動きを選択

Load

選択モーションリストにロボットに保存されたモーションをアップロード

Download

選択モーションをロボットにダウンロードして保存

	Name	Next
100	Walk Forward	-

モーション Name 横の Next 欄に次のモーション番号を書いて入れると、選択モーションのステップ完了後、Next に書かれた番号のモーションを実行します。

(5) モーションデータ

選択されたモーションの各ステップのデータを表示および編集して、モーションのプロパティの管理、実行、テストを提供しています。

Motion Data		RPT	Move Time	Stop Time	Pitch	Roll	ID0	ID1	ID2	ID3	ID4	ID5	ID6	ID7	ID8	ID9	ID10	ID11	ID12	ID13	ID14	ID15	ID16	ID17	ID18	ID19	ID20	ID21	ID22	ID23	
▶	STEP 0	X	0.62 sec	0.00 sec	0.0°	0.0°	0	0	0	0	12	0	0	0	0	0	12	0	0	-353	0	-90	0	0	-353	0	-90	0	0	0	0
⊗	STEP 1	X	0.63 sec	0.00 sec	0.0°	0.0°	0	-4	50	-101	62	52	0	22	24	-47	33	-49	0	-353	0	-90	0	0	-353	0	-90	0	0	0	0
⊗	STEP 2	X	0.54 sec	0.00 sec	0.0°	0.0°	0	-30	40	-44	31	69	0	47	30	-29	21	-47	0	-353	0	-90	0	0	-353	0	-90	0	0	0	0
○	STEP 3	○	0.14 sec	0.00 sec	0.0°	0.0°	0	4	60	-49	35	27	0	25	98	-196	119	-25	-1	-353	0	-90	0	1	-308	0	-90	0	0	0	0
○	STEP 4	○	0.15 sec	0.00 sec	0.0°	0.0°	0	-6	33	-35	33	27	0	42	180	-169	26	-33	92	-353	0	-90	0	-92	-353	0	-90	0	0	0	0
○	STEP 5	○	0.20 sec	0.00 sec	0.0°	0.0°	0	33	-7	-51	88	-22	0	-21	107	-41	-45	21	239	-353	0	-90	0	-230	-353	0	-90	0	0	0	0
○	STEP 6	○	0.31 sec	0.00 sec	0.0°	0.0°	0	29	0	-29	56	-29	0	-20	104	-79	-2	25	145	-353	0	-90	0	-120	-349	0	-90	0	0	0	0

▶	STEP 0	X	0.62 sec	0.00 sec	0.0°	0.0°	0	0	0	0	12	0	0	0	0	0	12	0	0	-353	0	-90	0	0	-353	0	-90	0	0	0	0
---	--------	---	----------	----------	------	------	---	---	---	---	----	---	---	---	---	---	----	---	---	------	---	-----	---	---	------	---	-----	---	---	---	---

各ステップのステップ段階、繰り返し区間、動作時間、停止時間、ロボットの傾き、各サーボの位置の値を表示します。



ステップを追加生成します。ステップは、モーションごとに最大 24 個まで作成可能です。



ステップ削除します。選択されたステップを削除します。



ステップの順序を調整。選択されたステップを上下に移動させます。



モーションのプロパティを編集。モーションの繰り返し区間の設定、モーションオプション機能の選択をすることができます。

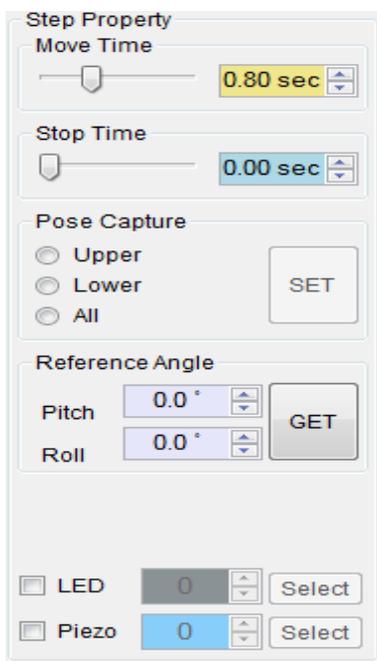


選択されたモーションを実行テストします。作成されたモーションをロボットに一時的に保存させて実行させて動作を確認することができます。

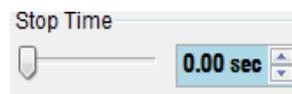
(6) ステップのプロパティのコレクション

選択されたステップの属性で動作時間、停止時間、姿勢キャプチャ、ロボットの傾きの入力、胸部 LED、ピエゾ出力音などを設定することができます。

画面全体



選択されたステップの動作時間を設定します。



選択ステップの実行後停止時間を設定します。



姿勢のキャプチャ機能で、上 (Upper)、下半身 (Lower) を分けて、または全体 (All) を選択すると、モーターの力が解かれながら、自由に各ステップモータの人為的な姿勢の設定が可能

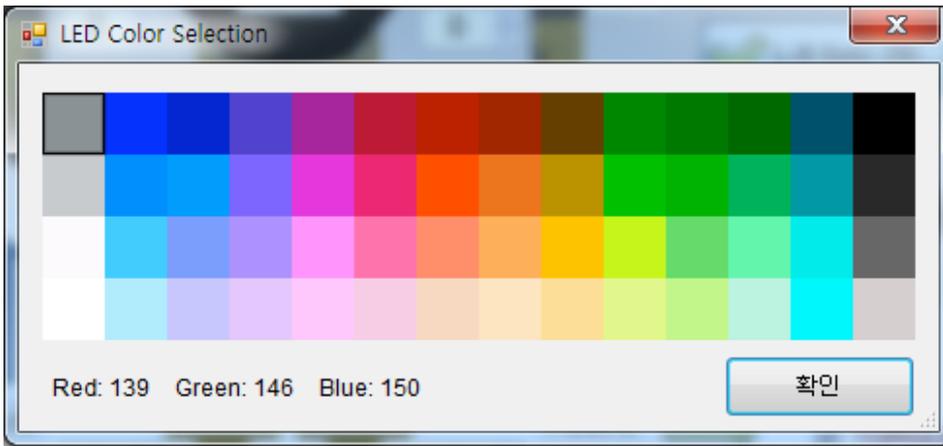


上下左右の座標修正。バランスを維持する角度を設定します。

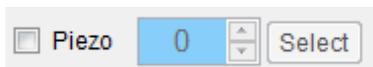


希望モーションステップにロボットの胸の部分にある LED カラーを制御します。

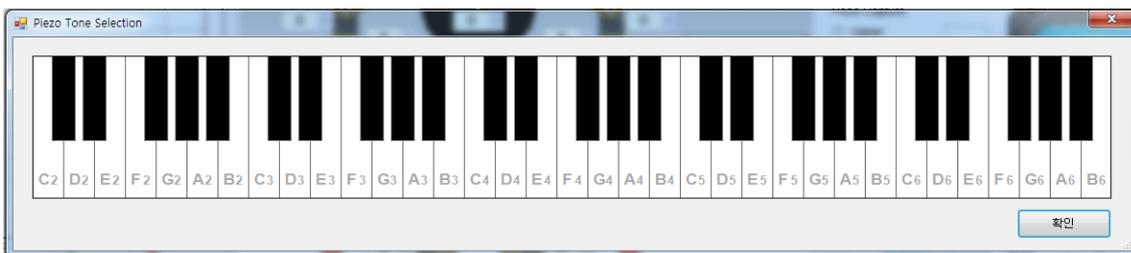
それぞれの数値の設定に応じて、LED の色が変わります。



※ LED 番号星色 List は、付録を参照



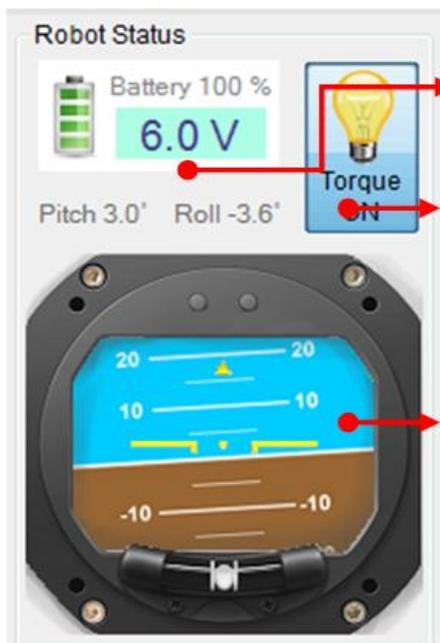
希望モーションステップにロボットの胸部にある Piezo を制御する。それぞれの数値の設定に応じて、Piezo の音が異なります。



※ Piezo 番号星音階 List は、付録を参照

(7) ロボットの状態表示

ロボットのバッテリー電源の状態、全体のトルクスイッチ、ロボットの傾きなどを表示します。

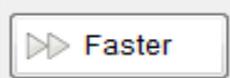
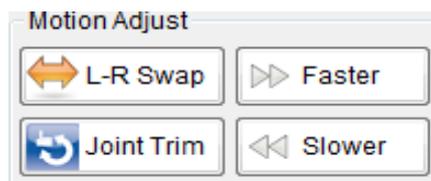


バッテリー電源の状態を電圧とパーセンテージで表示します。

全体のトルク有効/無効。電球形のアイコン Torque On 有効状態でクリックすると Torque OFF に変わり、この時のロボットのすべてのサーボのトルクが解除されます。

ロボットの傾きを表示する機能で、ロボットの上下左右の傾き状態表です。Pitch 値と Roll 値を出力します。

(8) モーション修正ツールバー



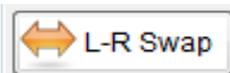
選択されたモーションのすべてのステップの動作時間と停止時間を一定量増加



選択したモーションのすべてのステップの動作時間と停止時間を一定量減少



選択されたモーションのすべてのステップの特定の関節のサーボ位置の値を補正



すべてのステップの選択されたモーションのロボット左右対称であるサーボ間の

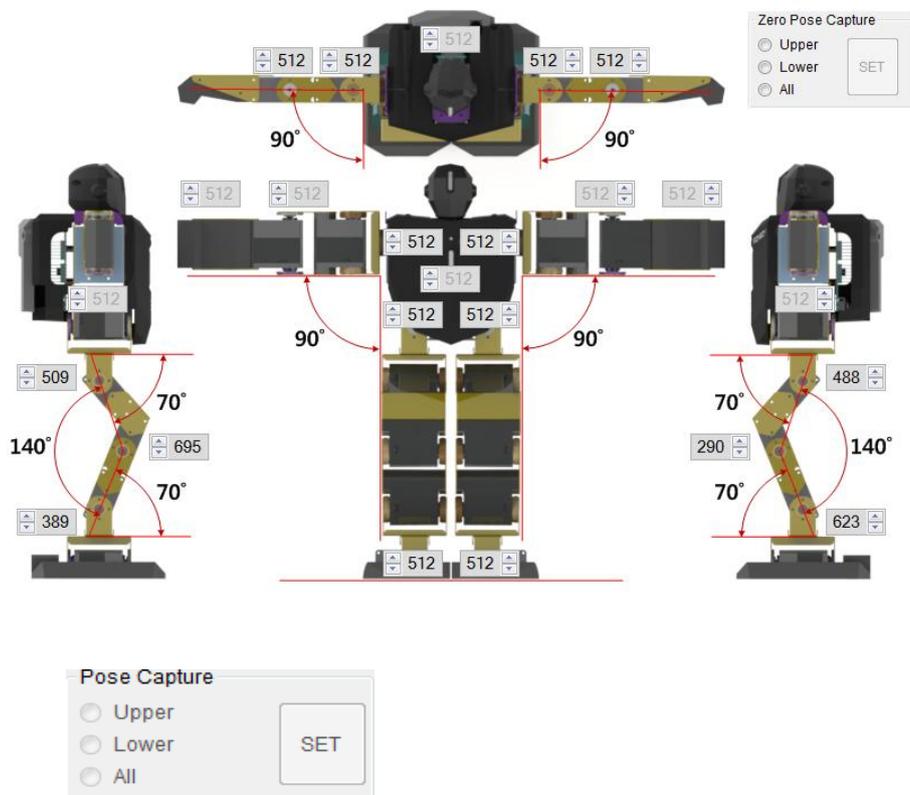
位置の値を交換

4.5. 各関節ゼロ点調整と基本パラメーターの設定

各関節ゼロ点調整、サーボ属性、ロボットの基本属性、自動回復機能などを設定することができます。

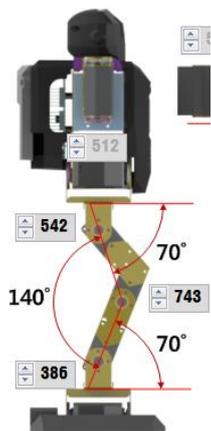
(1) 各関節ゼロ点調整

基本となるゼロ点調整が可能であり、その場所モータの値を調整してゼロ点を調節することができます。

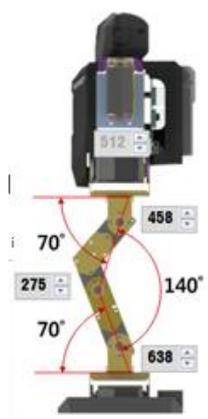


■姿勢をキャプチャして、ゼロ姿勢を設定します。

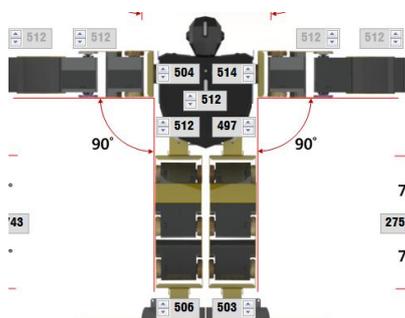
■上 (Upper)、下半身 (Lower) を分けて、または全体 (All) を選択すると、その部位のモーター力が解けて自由に各ステップモータの姿勢を自由に設定することができます。



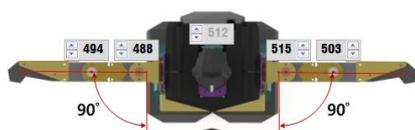
- 左図 2 番と 4 番のモーターの角度が 140 度であり、2 番と 4 番のモーターが一直線になるようにします。（図の黄色の点線を参照）
- のうち、3 回モータを基準として、それぞれ 2 番と 4 番のモーターとの角度をそれぞれ 70 度に保たせます。



- 右図のように、反対側も 8 番と 10 番のモーターの角度が 140 度であり、8 番と 10 番のモーターが一直線になるように維持します。（図の黄色の点線を参照）のうち、9 番モーターを基準にそれぞれ 8 回と 10 回モーターとの角度を 70 度に保たせます



- 正面に見たとき上半身と下半身が一直線になるように、骨盤モーター (1,7) を一直線になるように設定させ、肩モーター (17,12) を設定して、ロボットの腕と胴体を 90 度に設定します。



- 腕の形を手のひらが前を見ることができるようして胸の部分と肘の部分 を 90 度に設定して、肩から手まで一直線に設定します

(2) サーボプロパティの設定

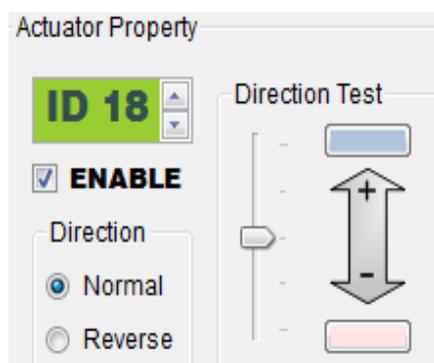
モーターをクリックするモータの数と使用するかどうか、動きのテストが可能です。

選択されたサーボの関節 ID を表示および変更します。

チェックされている状態で、サーボの機能を有効にします。

モータの制御値を、既存の方向または逆方向に選択することができます。

注：現在のライブラリが生成されており、この方向を逆に変えるこれまでに生成されたモータの動きの方向が変更されますので参考にして使用してください。



(3) ロボットの基本属性

ロボットのバッテリー電圧を指定し、各モーションの動きの速度を指定します。

Basic	Balance Control	Auto Recovery
Robot ID	1	
Battery Full Voltage	7.0 V	
Battery Low Voltage	5.5 V	
Motion Speed Rate	1.00 x	

(4) バランス制御

ENABLE がチェックされている状態で有効になります。この機能が実行されている間に、ロボットはジャイロセンサーを使用して、モーション実行中やスタンバイ時のバランスをとるために微細に動くようになります。

Basic	Balance Control	Auto Recovery
<input checked="" type="checkbox"/> ENABLE		

(5) 自動回復 (Auto Recovery) 設定

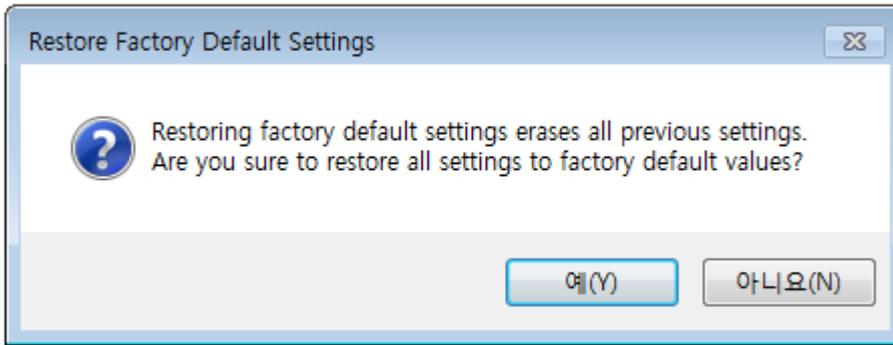
ロボットが倒れたり倒した姿勢になったとき、指定されたモーション番号に基づいて行動をして目を覚ますようにします。

Basic	Auto Recovery
<input checked="" type="checkbox"/> ENABLE	
Trigger Angle	± 45 °
Recovery Motion	
Front Down	109
Back Down	108



(6) 工場初期化

クリックすると、次のメッセージが表示されます。



「はい」をクリックすると、工場出荷時の初期化を実行します。この時、多少時間がかかりますので、中間に電源を遮断せずに待機します。「いいえ」ボタンを押すと、工場出荷時の初期化をせずにキャンセルします。

(7) 保存



すべての設定を終えて、このボタンを押して、ロボットに保存します。

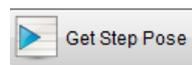
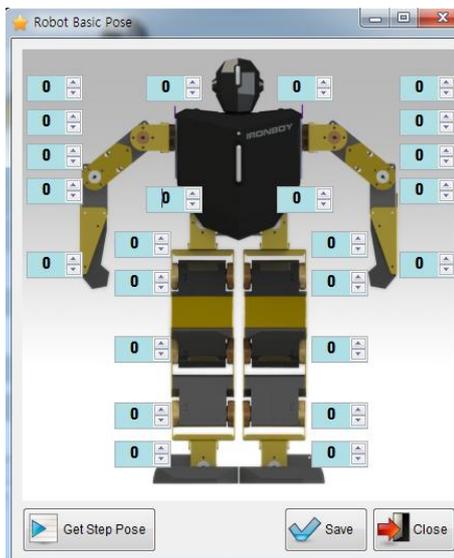
(8) 設定を閉じる



関節と基本パラメータの設定を終了します。

4.6. 基本姿勢の設定

基本姿勢を入力します。各モータ毎に自由に任意に基本姿勢を変更することができます。



ロボットの関節制御画面に入力されているサーボの位置の値をすべて読み込みます。

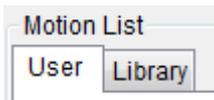


表示されたサーボの位置の値を基本姿勢として保存します。

4.7. モーション編集およびライブラリ

(1) モーション編集

1 モーション生成



編集するモーションリストを選択します。モーションリストがロボットに保存されている

ことを確認して使用します。ユーザー (User) のリストは、ユーザーが直接作成することです。ロボット購入時のモーションが保存されていません。ライブラリ (Library) リストは、ライブラリのファイルを利用して保存します。ロボット購入時の保存されています。

	Name	Next
▶ 0		
1		
2		
3		
4		
5		
6		

生成するモーション番号をクリックして、Motion data

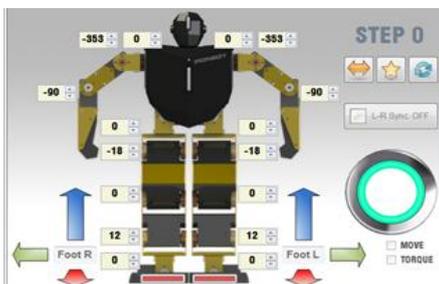
ウィンドウの  アイコンをクリックします。

	Name	Next	RPT	Move Time	Stop Time	
▶ 0	New Motion	-	▶ STEP 0	X	0.80 sec	0.00 sec

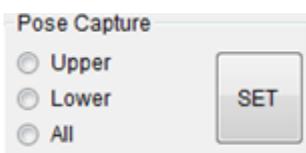
アイコンをクリックする上の図のように New Motion とモーションタイトルと一緒に STEP0 というステップが生成されます。

	Name	Next
✎ 0	Motion New	-

モーションタイトルをダブルクリックすると、モーションの名前を変更することができます。



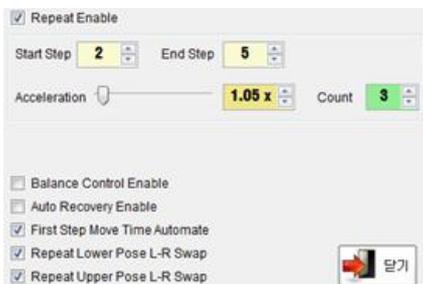
左図のように、ロボットの正面に面した基準に各部分のモーターの値をマウスホイールまたはキーボード入力で制御します。左右を同じ値に動く場合 LR Sync OFF をクリックすると、左右の両軸の同じモーターを同じ値に制御することができます。



個々のモータの値をいちいち入れてくれるのが困難な場合 Pose Capture ウィンドウを使用上半身は Upper、下半身は Lower、サンハチエ全体は All をクリックすると、クリックし、その位置モーターのトルク (力) が解除されます。この時、自由にロボットを動かすことができ、移動中にモーターの値は、ステータスウィンドウに表示されます。ポーズをチュウイハシン[SET]ボタンを押すと、その姿勢でモータの力が入るようにされ、姿勢を維持します。



Motion Data		
	RPT	Move Time
★	X	0.80 sec
▶	X	0.53 sec



	RPT
▶STEP 0	X
STEP 1	X
STEP 2	○
STEP 3	○
STEP 4	○
STEP 5	○
STEP 6	X

Move Time 値に現ステップのモーションの動きの速度を速くまたは遅く設定し、次のステップに進むディレイを Stop Time に設定できます。

現在のステップの機能の中でさらにロボットの胸にかかっている LED と Piezo を使用することができます。使用機能をチェックボックスでチェックすると、各番号ごとに LED の色と Piezo の音が生成されます。



★をもう一度押すと、2 番目のステップが生成されます。2 回目の生成以上を入力された場合も、最初のステップの作成時と同様のプロセスを繰り返します。

上記を行ったモーションに繰り返し区間を設定する場合は、Motion Property アイコンを押して、区間の繰り返しを実行することができます。

繰り返し区間は、ユーザーに応じて繰り返すことができおり、左の図は、2～5 回目のステップ区間を繰り返しました。また、1.05 倍、3 回の速度を増加させ、設定では、左右を交互に繰り返していました。3 回回数が増えれば速度は 1.05 倍になって加速されて速くなります。図に示す残りの設定については、28 ページを参照してください。

繰り返し区間が設定される場合には、左図の RPT 項目の繰り返し区間ステップのみ O 表示に変わり、繰り返し区間がない場合は、X で表示されます。繰り返し区間は、ロボット操縦アプリケーションの場合は、そのボタンを押している間、トータルマネージャーソフトウェアでのモーションテストの場合は、テスト終了ボタンを押すまで繰り返します。左図では繰り返し区間がない STEP6 番の場合終了ボタンを押して STEP5 回に着いた時の動作になります。

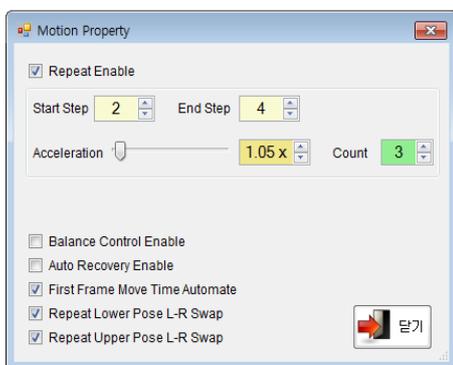


モーションまたはモーションが完成した場合、モーションの選択と **Select All** をクリックして、モーションを選択します。そして **Download** をクリックすると、ロボットライブラリをダウンロードすることができます。



ダウンロードまで終えた後、必ず **File - Save** または **File - Save as** をクリックして、作成されたライブラリを保存してください。保存時にファイルの形式は、左図のように、ファイル名は任意の名前で保存します。

2 モーションのプロパティを設定



動作特性の窓は、特定のモーション部分の区間リピートを可能にすることができ、また、繰り返し区間ごとにモーションの動きの速度を増加させることができます。繰り返し区間はアプリの動作コマンドボタンを押している場合、または、マネージャープログラムのモーションテスト時終了ボタンを押すまで、特定の指定された区間のみを繰り返しており、アプリの動作コマンドボタンを離す場合とマネージャープログラムのテスト終了ボタンを押した場合繰り返し区間が停止され、停止した後選択されているステップに戻ります。

Repeat Enable

繰り返し区間を設定

Start Step 2 End Step 4

繰り返すステップ区間を設定します。(上図はステップ 2~4 を繰り返し)。

Acceleration 1.05 x Count 3

繰り返し区間で繰り返し区間を実行する場合、速度がますます加速します。Count の回数と繰り返し倍速設定に応じて速度が速くなります。(注意：加速度をも過度に設定した場合、サーボモータに悪い影響をかけることになります。テストでは、適切に設定してください。)

Balance Control Enable

モーションを実行中 **Balance Control** を有効にします。

Auto Recovery Enable

モーションの実行中に **Auto Recovery** を有効にします。

First Step Move Time Automate

モーションの最初の起動時に現在の姿勢でスムーズな移行のために適正に速度調節をします。

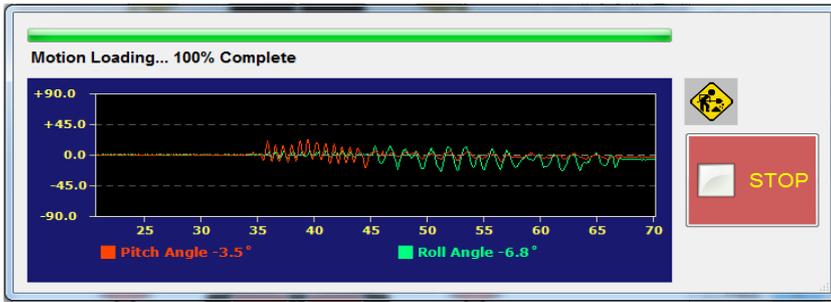
Repeat Lower Pose L-R Swap

下半身のモーター値を左右反転してモーションを繰り返し実行

Repeat Upper Pose L-R Swap

上半身のモーター値を左右反転してモーションを繰り返し実行

3 モーション実行テスト

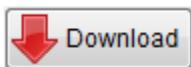


テスト中の画面に Pitch 値と Roll 値を画面に数字とグラフで表現します。 x 軸の動作時間を表し、y 軸は 0 を基準に Pitch と Roll をグラフ化します。



をクリックすると、テストが終了します。

4 モーションをロボットに保存



をクリックすると、進行状況画面が表示され、選択された項目のモーションをロボットに保存します。

5 モーションをロボットで読む



をクリックすると、進行状況画面が表示され、選択された項目のモーションをロボットからロードします。

6 モーションをファイルに保存



をクリックすると、選択されたリストのモーションをファイルに保存します。

7 モーションをファイルから読み込み



を押すと、選択されたリストのモーションをファイルから読み込みます。

8 新しいファイルを作成する

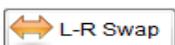


をクリックすると、選択したリストのモーションを削除します。新しいファイルを作成する前に、既存のモーションの保存状態を確認してください。

9 モーション補正



選択したモーションのすべてのステップの値を次の 4 つの方法で補正します。



左右対称なサーボ間の位置の値を交換



特定の関節のサーボ位置の値を補正する[付録]モーションデータ関節補正 (参照)



動作時間と停止時間を一定量増加



動作時間と停止時間を一定量減少

(2) モーションライブラリ

モーション番号 100~123 回まではロボット動作の検査をして出荷されます。

その他のモーションは、動作を参考にして、個々のロボットのモーションを実行に適切で時には、ロボットに合わせて調節して使用します。

4.8. バーチャルコントローラー



(1) ライブラリモーションリスト

	Name	Next
100	Walk Forward	-
101	Turn L	-
102	Turn R	-
103	Walk Backward	-
104	Front Punch	-
105	Side Punch L	-
106	Side Punch R	-
107	Rear Attack	-
108	Get Up Forward	-
109	Get Up Backward	-
110	One Step Forward	-

バーチャルコントローラーの実行がされると、ロボットに格納されているモーションの名前を読んで表示します。モーション番号、モーション名は、次のモーション番号の情報を表示します。

(2) バーチャルコントローラーの操作方法



状態表示ウィンドウです。操縦機を通じたモーション実行状態を表示します。



方向ボタンです。上・下・左・右



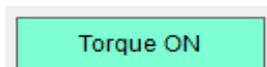
図形ボタンです。○・△・×・□



ジョイスティックです。上、下、左、右と斜め4方向に計8方向になります。左右一個ずつあります。



追加機能のボタンです。追加ボタンは左、右一個ずつあります。

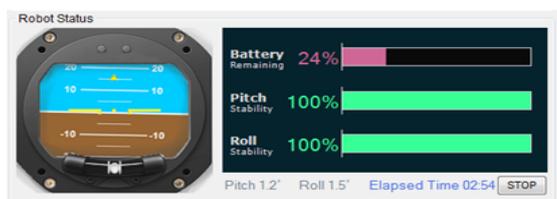


ロボット全体のサーボの力を調整します。「ON」のとき、トルク（力）が入り、「OFF」のとき、解除されます。



送信データの周期を設定します。

(3) ロボットの状態



ロボットのバランス状態を前後（Pitch）、左右

ロボットの現在の状態を示します。バランスの状態、プロボ実行時間、バッテリー残量インジケータを表示します。↵



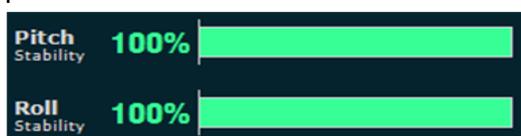
（Roll）で表示します。↵



プロボ実行時間を示し、STOP を押すと一時停止します。↵

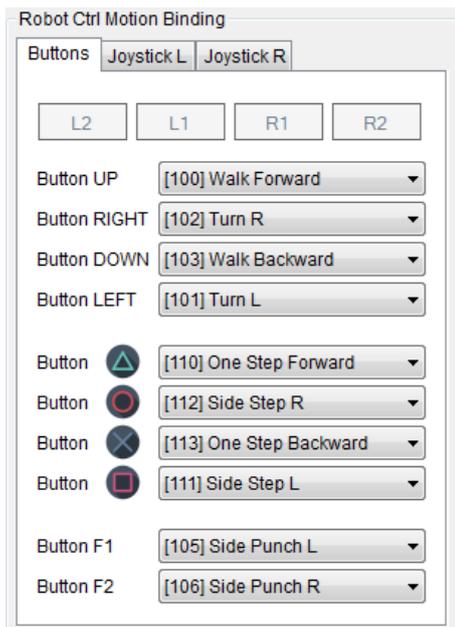


バッテリーの残量を表示します。↵

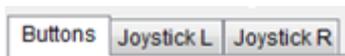


ロボットの Pitch と Roll 方向に動きの安定度を表示します。↵

(4) ロボット制御モーション接続



各ボタンとジョイスティックの動きを選択して接続することができます。



ボタン、ジョイスティックを区切って指定します

次のようにボタンの割り当てを複数に設定して使用可能です。ジョイスティックの値を以下のように5つの異なる組み合わせで使用することができます。



デフォルトのグループボタンの指定値を表示します。



L1 を有効にして L1 グループボタンの割当て、値を保存、使用します。



L2 を有効にして L2 グループボタンの割当て、値を保存、使用します。



R1 を有効にして R1 グループボタンの割当て、値を保存、使用します。



R 2 を有効にして R 2 グループボタンの割当て、値を保存、使用します。



左図の 10 のボタンにそれぞれモーションを指定することができます。

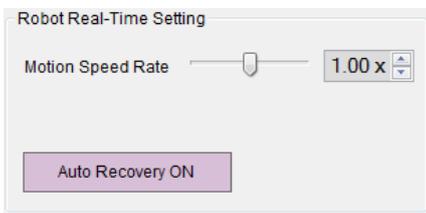


左図に示されている左側ジョイスティック 8 つの方向にそれぞれの動きを指定することができます。



左図に示されている右側ジョイスティック 8 つの方向にそれぞれの動きを指定することができます。

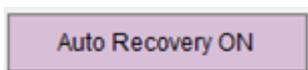
(5) ロボットのリアルタイム設定



モーションを実行するとき動きの速度を調節したり、リアルタイムの動きの機能を有効にすることができます。



モーションを実行するとき動きの速度の比率を調整します。選択されたモーションに関係なく、すべてに適用され、モーションデータが変わるわけではない。



モーション実行すると、自動回復（倒れたとき起こる）機能を選択することができます。変更しても、ロボットの設定値を変えないため、電源ジェイン可視には適用が解除されます。

4.9. 赤外線リモコンで操縦

(1) 使用可能な赤外線リモコン

サムスン電子製TV用リモコンまたはサムスン製に互換性のある万能型リモコンを使用します。
万能型リモコンの場合、サムスン製品のモードに設定して使用します。

(2) リモコンを設定する

サムスン製リモコンを使用時には音量減少ボタンを押して、「Greet with Wave」モーションを実行することを確認します。万能型リモコンを使用時には、サムスンを選択実行して使用します。

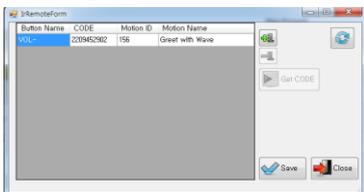
一般的に、TVの場合、音量減少ボタン操作時にTVの画面が出力され設定完了しますがIRH-100の場合は「Greet with Wave」モーションが実行されたのを基準に設定を完了します。

(3) リモコンコード表を編集する



ツールバーで左図のアイコンをクリックします。

以下のような編集画面でリモコンコードを登録して使用します。



[Button Name] リモコンのボタンを区別するための入力を保存します。

[CODE] リモコンのボタンに対応する固有のコードを登録します。

[Motion ID] リモコンのボタンにモーション ID を入力して、リモコンのボタンを押したときに実行するモーションを指定します。

[Motion Name] Motion ID を指定すると、自動的にその動きの名前を検索して表示します。



登録しようとするボタンの CODE 欄を選択し、リモコンの対応するボタンを押した後、このボタンをクリックすると、対応するボタンの固有のコードが自動的に入力されます。



指定ボタンを編集することができる行を追加作成します。



選択されたボタンの行を削除します。



現在アイコンボーイに保存されたリモコンコード表をロードします。



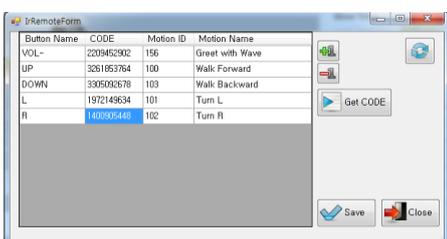
現在編集中的リモコンコード表の情報をアイコンボーイに保存します。



リモコンコードの編集ウィンドウを閉じます。

(4) 操縦する

指定されたとおりのボタンを押して、モーションを実行することができます。



5.1 インストールする

(1) APPインストール仕様

- Android OS 5.1 以上
- Bluetooth 4.0 BLE 以上

※上記の仕様を満たしていても接続できない場合がございます。

(2) APPインストールファイルのダウンロード

アプリ APKは iRRobot ホームページ (<http://www.irrobot.com>) からダウンロードすることができます。
(IRONBOY Controller.apk)

PLAY STORE からのダウンロードも近日公開となります。(2018年2月中旬を予定)

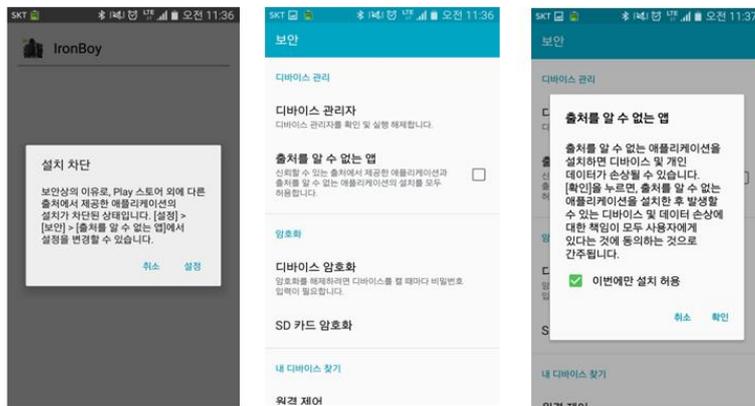


※PCからダウンロードを行なった後、USBケーブルでスマートフォンやタ

ブレットにファイルを移動します。

(3) APPのインストール

インストールファイル (apk file) を実行します。インストール中にインストールブロックのウィンドウが表示されたら設定メニューの「セキュリティ」メニューで「確認できないアプリ」信頼しインストールを行なうかのチェックを求められます。



※APK ファイルは圧縮されています。ご使用のスマートフォンやタブレットにより、ファイルダウンローダーや解凍ソフトが必要になる場合がございます。

インストールが完了するとショートカットアイコンが作成されます。

5.2. 設定する

(1) ボタンの設定方法

インストールされたアプリをショートカットアイコンから起動します。

アプリを実行時、BluetoothがOffの場合、スマートフォン・タブレットの設定からBluetoothをOnにしてからMain画面に移動します。



5.3. IRH-100 との通信リンク

IRH-100の背面にある電源スイッチをONにして待機状態にします。

スマートフォンやタブレットでBluetoothをONにします。

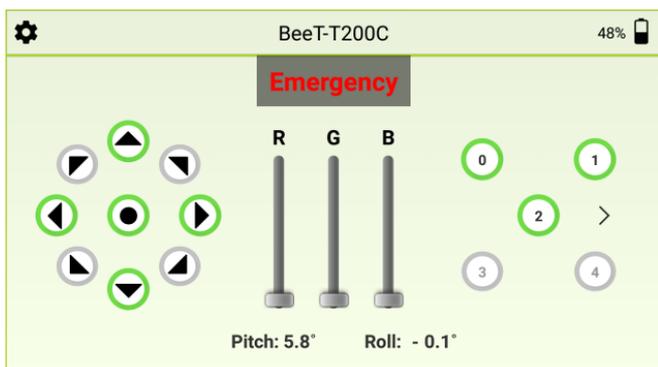
ショートカットアイコンからAPPを起動します。



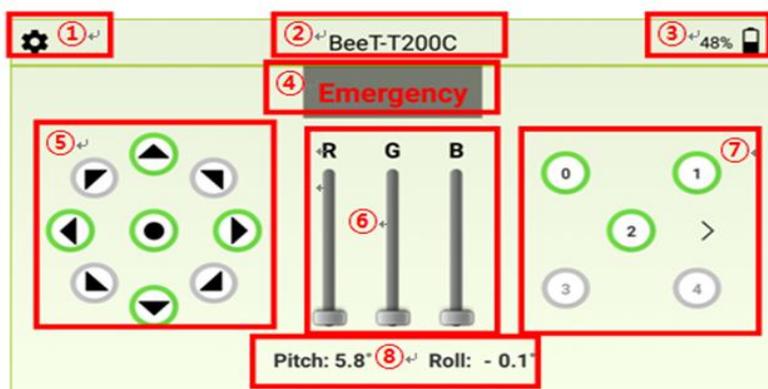
画面の  Bluetoothアイコンをタップして使用するデバイス (IRH-100) を選択します。



スキャンして表示されたBluetoothのうち接続するロボットのBluetoothを選択すると、
下図のControl画面に移動します



5.4 Control 画面について



- ①機能設定メニューボタン
- ②選択された Bluetooth 機器の名称
- ③ロボットのバッテリー残量表示
- ④Emergency 設定
- ⑤左側コントロールパネル
- ⑥胸部LEDコントロール
- ⑦右側コントロールパネル
- ⑧ロボット Pitch 値、Roll 値

- ①機能設定メニューボタン



この設定は、ロボットと接続した後、現在の状態を読み取り、自動的に設定がされます。トータルマネージャープログラムでの設定とは異なり、アプリでの設定は、ロボットの電源が切ってオンにすると消去されます。トータルマネージャープログラムで Auto recovery と Balance control 設定をすると、ロボットの電源を切っても設定値は残ります。



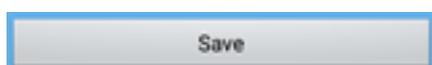
アプリからロボットへ送信されるコマンド伝送速度であり、調節範囲は、毎秒 40~100ms です。



ロボットのモーション実行と待機状態時にジャイロセンサーにより自動バランスコントロールのON/OFFを設定します。



ロボットが倒れたときに自動復旧機能をON/OFFにするボタンです。



すべての設定が保存されます。(ロボットの電源をOFFで設定はクリアされデフォルトに戻ります)

②選択された Bluetooth 機器の名

称 **BeeT-T200C**

③ロボットのバッテリー残量表示



④Emergency 設定



Bluetooth 検索ウィンドウで選択した Bluetooth モジュールの名前

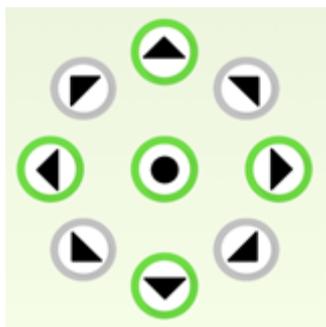
※図は BeeT-T200C デフォルトは IRONBOY

ロボットの電池の残量を表示します。

※図は 48%を示しています。

ロボットテストプロセスや動作の過程で、ロボットのトルクを緊急解除させるときに使用するボタン、再びトーク「On」をさせるためには、ロボットの電源を Off / On する必要があります。

⑤左側コントロールパネル



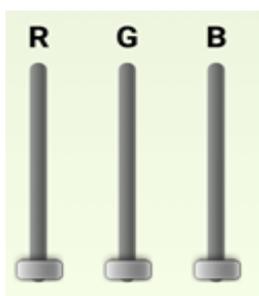
8 方向および中央キーで構成されています。

各ボタンの境界線が緑の場合、モーションが指定されたボタンであり、灰色の場合、モーションが指定されていないボタンです。

ボタンを押し続けると、同じモーションを繰り返し行い、指を離し停止命令を実行すると、停止します。この時、既に入力が行われたモーションは、最後まで実行してから停止します。

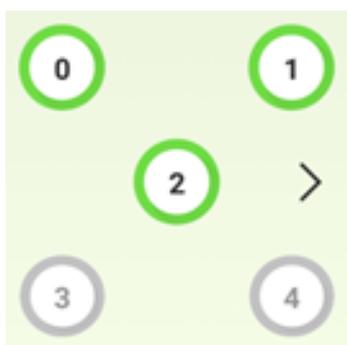
「●」ボタンを「255」に指定した場合は、何らかの理由でモーションが停止せずに継続して移動したときに「●」ボタンを押すと、ロボットを停止させることができます。

⑥胸部 LED コントロール



ロボットの胸側の RGB LED を個別に制御するためのボタンであり、スライド方式で、各 LED の明るさを調節して、色ごとに 255 色を作ることができます。

⑦右側コントロールパネル



5 つのキーで構成されています。

各ボタンの境界線が緑の場合、モーションが指定されたボタンであり、灰色の場合、モーションが指定されていないボタンになります。

ボタンを押し続けると、同じモーションを繰り返し行い、指を離し停止命令を実行すると、停止します。この時、既に入力が行われたモーションは、最後まで実行してから停止します。



ボタンを押すと、ボタンに対応するモーションを実行します。実行時は (Key) 番号とモーション番号 (Value) の値を表示します。



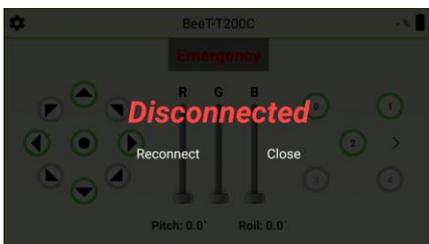
ボタンを押すと、次の追加番号に切り替えて、一度押すごとに 5 個ずつキー (Key) 番号が変わります。200 種類のモーションを保存することができます。(0~199)

⑧ロボット Pitch 値、Roll 値

Pitch: 5.8° Roll: - 0.1°

ロボットの Pitch 値、Roll 値の現在の状態を表示します。

5.5. 通信の切断 (Disconnected)



ロボット Control 途中、何らかの理由ロボットとの通信が切断されると、左図のような画面が表示されます。

Reconnect

ロボットとの再接続を試行します。

Close

アプリの初期画面に移動します。



ロボットとの Bluetooth 接続の間に、何らかの理由で、ロボットとの通信が切断されると、左図のようなメッセージを表示します。

ロボットの再起動もしくは、Bluetooth 接続を再試行してください。

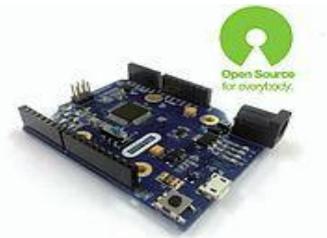
それでも接続ができない場合、ロボットの再起動と、スマートフォンやタブレット端末のアプリも完全に終了した後、アプリの再起動した後、Bluetooth 再接続を試行します。

6.1 Arduino と IRduino

(1) Arduino の特徴

1 オープンソースにより初心者から熟練者まで

アドゥーイーノベースの IRduino は使いやすいハードウェアとソフトウェアに基づくオープンソースのプラットフォームです。初心者だけでなく、開発者が柔軟で自由にプログラムをすることにより、熟練者にも十分に魅力があるプラットフォームとすることができます。



2 豊富な情報

アドゥーイーノコミュニティを通じて、多様で豊富な技術情報をいつでも入手することができます。ウェブに数多くのアドゥーイーノコミュニティが存在しており、初心者でも簡単に関連する情報、ヒントなどを提供することができます。市販されている様々なアドゥーイーノ教材を活用して見ることができます。



3 簡単です。

IRduino を制御するために、ユーザーは該博な電子知識がなくても構わないです。さらにインタラクティブ音楽やビデオアートの分野の芸術家もアドゥーイーノを活用して、彼らの創造的なプロジェクトを容易に作っていきます。

4 C 言語の学習のための最適なソリューションです。

アドゥーイーノまた、C 言語と同様の言語を使用するので、高度なプログラミング言語である C を学ぶための優れた足がかりです。言語は C++ ライブラリを使用して拡張することができ、もっと高難度のプログラミングを学ぼうとする上級ユーザーは AVR C 言語でスキップすることもできます。



一方で、ユーザーは必要に応じて AVR-C コードをアドゥーイーノプログラムに直接追加することができます。また、Flash、Android、MaxMSP、Processing や Object C の経験がある場合は、その知識を活用することもできます。

容易なプログラミングと簡単に接することができる多くの情報を使用して、非専門家でも、ロボット工学を簡単に学び、プロトタイプなどの独自のロボット構造を効率的に作ってみることができます。

様々なシールドボード装着時は、別のはんだ付けが不要ブレッドボード構造です。

5 すぐに使用可能なさまざまなアドゥーイーノシールドボード

すでに市中に様々なアドゥーイーノシールド (センサー等) が流通しており、開発者がさまざまなロボットのタスクを短期間で実装を試みることができます。

市販の他のマイクロコントローラシステムの価格と比較して、合理的な価格で販売されており、開発費を最小限に抑える一助することができます。様々なアドゥーイーノシールドを利用して、自分だけのロボットができることを創作してみてください。



6 便利です。

USB ポートを使用

USB ポートを使用しているおかげで、互換性のあるジェンダーなどが必要ありません。簡単に USB ポートに接続すれば IRduino ボードを制御することができます。

Cross-platform

IRduino は市中に流通されているほとんどの O/S と互換性があります。Windows、Mackintosh OSX、Linux、など自由に使用することができます。



(2) IRduino

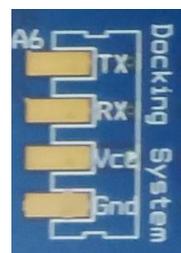
Arduino Leonardo と同様の構成でアドゥーイーノ開発環境が対応するように開発されたボードです。

次のような機能が Arduino Leonardo に比べて追加されました。



1 IRC-24 との通信インターフェース

UART とバッテリーの電力を接触方式で接続する端子です。IRH-100 のメインボードと IRduino ボードは簡単 Docking system を介して接続がされます。



2 UART 通信を使用しているシールドとの同時使用のためのスイッチの通信ポートを使用するシールドとの同時使用のために、通信ポート (D0、D1) と一般的な I/O (D8、D9) を選択的に使用可能です。



6.2. Shield (アドゥーイーノシールド)

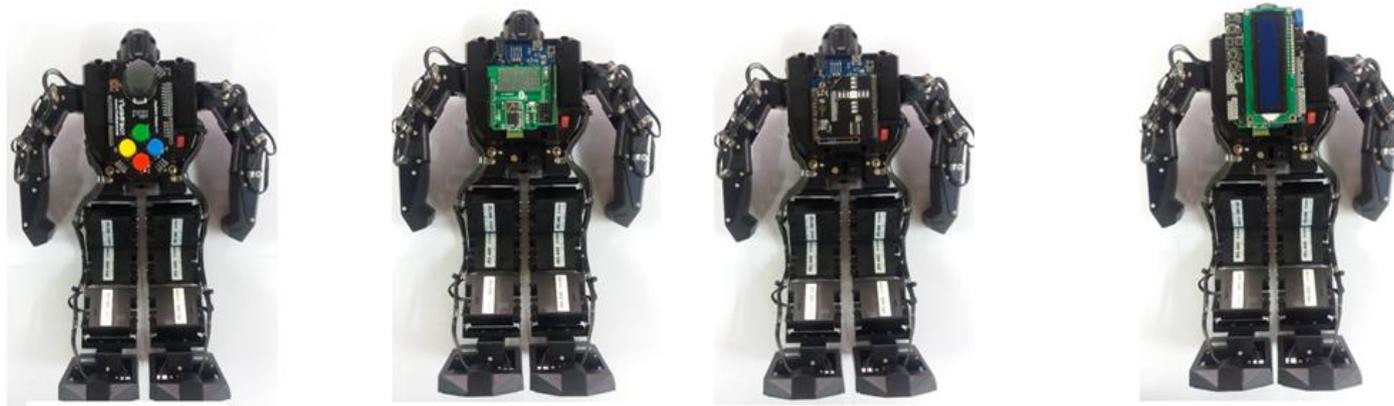
アドゥーイーノシールドは、ボードに接続して 性能を拡張させるためのハードウェアです。

例えば、ブルートゥースシールドをボードに接続すると、ボード上の Bluetooth の使用が可能になります。現時点でシールドの種類は、300 個を超え、ほとんど UNO に合わせて製作されており、他のアドゥーイーノボードと互換性のあることが多いです。頻繁に使用されるものでは LCD ボタンシールド、イーサネットシールド、無線 LAN シールド、SD カードシールドは、Bluetooth シールドなどがあります。

(1) 適用可能な Arduino Shield 人

- LED 制御
- FND 制御
- DOT MATRIX 動作制御
- TEXT LCD の動作制御
- COMMUNICATION 動作制御
- DIP SW の動作制御
- KEYPAD 動作制御
- STEP MOTOR 属性出力
- KEYPAD を利用した STEP MOTOR 制御
- COMPASS SENSOR 制御
- PIR SENSOR 制御
- CDS SENSOR 制御
- SERVO MOTOR 動作制御
- STEP MOTOR 動作制御実習
- KEYPAD を利用した計算機を作る
- KEYPAD を利用した DOT MATRIX 制御
- SERVO MOTOR 属性出力
- KEYPAD を利用した SERVO MOTOR 制御
- ULTRASONIC SENSOR 制御
- VR SENSOR 制御
- オン/湿度 SENSOR 制御
- NFC 制御
- BLUETOOTH 制御

(2) シールドの使用例



6.3. API とライブラリ

1 のライブラリファイルを構成する

Description	API Name	Library Name		
ロボット動作を実行	ExecuteRobotMotion	MotionDriver	Instruction	
アクチュエータの実際の位置の値の移動	writeActuatorRawPosition	ActuatorDriver		
アクチュエータの目標位置の値を移動	writeActuatorPosition	ActuatorDriver		
単一ステップ移動	WriteMonoStep	MotionDriver		
ロボットのリアルタイム機能の設定	RealtimeFunction	ParameterDriver		
ロボット姿勢のキャプチャ設定	MotionCaptureEnable	MotionDriver		
ロボットの基本設定を保存	StoreBaseParameter	ParameterDriver		
ロボットの動作プロパティの保存	StoreActivityProperty	MotionDriver		
ロボットデータ Feedback 要求	FeedBackRobotData	ParameterDriver		
LED の色出力	ChestFullColor	LedDriver		
Piezo 音出力	playSound	PiezoDriver		
コマンド正常受信確認	FeedBackRobotData	ParameterDriver		Feedback
ロボットの現在の状態	FeedBackRobotData	ParameterDriver		
ロボットアクチュエータ基準位置値	FeedBackRobotData	ParameterDriver		
ロボットの基本設定	FeedBackRobotData	ParameterDriver		
に基本姿勢	FeedBackRobotData	ParameterDriver		
ロボットの現在姿勢	FeedBackRobotData	MotionDriver		
ロボット動作の実行応答	FeedBackRobotData	MotionDriver		
ロボット動作属性	FeedBackActivityProperty	MotionDriver		
ロボット動作ステップ	FeedBackMonoStep	MotionDriver		

2 API 説明

- 機能：ロボットの動作を実行
- パラメータ
- MotionNum：モーション番号

※ロボットの動作番号を定義

全体のトルクの 状態	の説明	の定義	動作 番号
) ON	ユーザーの動作 0～99 (外部メモリ領域に格納	MOTION_USER_00～ 99	0～99
	固定ライブラリの動作 0～99 (内部メモリ領域に保存	MOTION_LIBRARY_00 ～99	100～19 9
OFF	全体アクチュエータのトルクを解除	MOTION_TORQUE_O FF) 200
) ON	ユーザーの動作生成のテストのために一時的に実行する動作 (RAM 領域を保存	MOTION_USER_TEST	201
変化なし	動作停止 (既存の実行中の動作を停止させる)	MOTION_STOP	255

- Recv_buf : モーション実行返信データ保管場所

writeActuatorRawPosition (uint8_t _id、int16_t _position)

- 機能 : アクチュエータの実際の位置の値の移動
- パラメータ
- id : 動作するアクチュエータ ID
- _position : 移動、実際の位置の値に 0～1023 を設定

writeActuatorPosition (uint8_t _id、int16_t _position、bool move、bool torque)

- 機能 : アクチュエータの目標位置の値の移動
- パラメータ
- id : 動作するアクチュエータ ID
- _position : 移動目標位置の値に-512～511 を設定
- move : 「0」 のとき、位置の値だけを保存し、「1」 のとき、動作も実行する。
- torque : 「0」 のとき、サーボの力を除去し、「1」 のとき、力を印加する。

WriteMonoStep (uint8_t step_move_time、uint8_t step_stop_time、uint8_t step_led、bool step_led_enable、uint8_t step_piezo、bool step_piezo_enable、int16_t step_reference_pitch、int16_t step_reference_roll、int16_t * position)

- 機能 : 単一のステップ移動
- パラメータ
- step_move_time : ステップ移動時間 (範囲 : 1～255、単位 : 0.01 sec)
- step_stop_time : ステップ停止時間 (範囲 : 1～255、単位 : 0.01 sec)
- step_led : ステップ LED の色番号 (範囲 : 0～63) は、付録の[LED 制御パレット]を参照
- step_led_enable : LED の色出力の有無ビット、「1」のとき、色出力
- step_piezo : ステップ Piezo 音番号 (範囲 : 0～63) は、付録の[ピエゾ音番号札]参照
- step_piezo_enable : Piezo 音出力の有無ビット、「1」のとき、音出力
- step_reference_pitch : ステップ Pitch 参照角度 (範囲 : -900～900、単位 : 0.1°)
- step_reference_roll : ステップ Roll 参照角度 (範囲 : -1800～1799、単位 : 0.1°)
- position : アクチュエータの目標位置の値 (範囲 : -511～511) 低された配列のアドレス

RealtimeFunction (uint8_t motion_speed_rate、uint8_t realtime_function_enable)

- 機能：ロボットのリアルタイム機能の設定
- パラメータ
- motion_speed_rate：動作実行倍速（範囲：50～150、単位：0.01
- x) realtime_function_enable：リアルタイム機能活性バイト

説明	定義	Bit
バランス制御有効ビット (1 : Enable、0 : Disable) Bit	BALANCE_CONTROL	Bit 0
転倒自動回復アクティブビット (1 : Enable、0 : Disable) Bit	AUTO_RECOVERY	1
予備活性ビット	RESERVED	2
予備活性ビット	RESERVED	Bit 3
予備活性ビット	RESERVED	Bit 4
予備有効ビット	RESERVED	Bit 5
予備活性ビット	RESERVED	Bit 6
全体のトルク有効ビット (1 : Enabled、0 : Disabled) MotionC aptureEnable (protocol_servo_bit_t	TOTAL_TORQUE	Bit 7

pose_capture_enable)

- 機能：ロボットの姿勢のキャプチャ設定
- パラメータ
- pose_catpture_enable：姿勢キャプチャ活性バイト
- (3Byte) ※姿勢キャプチャ活性バイト 0

説明	定義の	Bit
アクチュエータ 0 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_0	Bit 0
アクチュエータ 1 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_1	1
アクチュエータ 2 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_2	2
アクチュエータ 3 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_3	3
アクチュエータ 4 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_4	4
アクチュエータ 5 回キャプチャ有効ビット (1 : Enable、0 :	CAPTURE_ENABLE_ACTUATOR_5	5
アクチュエータ 6 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_6	Disable) Bit6
アクチュエータ 7 回キャプチャ有効ビット (1 : Enable、0 : Disable) ※	CAPTURE_ENABLE_ACTUATOR_7	7

姿勢キャプチャ活性バイト 1

説明	定義	Bit
アクチュエータ 8 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_8	Bit 0
アクチュエータ 9 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_9	1
アクチュエータ 10 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_10	2
アクチュエータ 11 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_11	3
アクチュエータ 12 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_12	4
アクチュエータ 13 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_13	Disable) Bit5
アクチュエータ 14 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_14	6
アクチュエータ 15 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_15	7

Enable、0 : Disable) ※姿勢キャプチャ活性バイト 2

説明	定義	Bit
アクチュエータ 16 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_16	Bit 0
アクチュエータ 17 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_17	1
アクチュエータ 18 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_18	2
アクチュエータ 19 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_19	3
アクチュエータ 20 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_20	4
アクチュエータ 21 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_21	5
アクチュエータ 22 回キャプチャ有効ビット (1 : Enable、0 : Disable) Bit	CAPTURE_ENABLE_ACTUATOR_22	6
アクチュエータ 23 回キャプチャ有効ビット (1 : Enable、0 : Disable) StoreBaseParameter (uint8_t	CAPTURE_ENABLE_ACTUATOR_23	7

battery_full_voltage、uint8_t battery_low_warning_voltage、uint8_t
 motion_speed_rate、uint8_t* balance_PID、uint8_t auto_recovery_trigger_angle
 、uint8_t auto_recovery_front_down_motion、uint8_t

auto_recovery_rear_down_motion、uint8_t function_enable_byte) ;

- 機能：ロボットの基本設定を保存
- パラメータ
- battery_full_voltage：バッテリー緩衝電圧（範囲：40～255：単位：0.1
V）
- battery_low_warning_voltage：バッテリー低電圧警告電圧（範囲：40～255：単位：0.1
V）
- motion_speed_rate：動作実行倍速（範囲：50～150、単位：0.01
- x) balance_PID：バランス制御 PID 比例定数
- (RESERVED) auto_recovery_trigger_angle：転倒自動回復反応角度（範囲：1～90、単位
：±1°）
- auto_recovery_front_down_motion：今後転倒した場合、回復動作（範囲：0
～200）
- auto_recovery_rear_down_motion：バック倒れた場合の回復動作（範囲
：0～200）
- function_enable_byte：動作機能活性バイト

説明	定義	Bit
バランス制御アクティブビット (1 : Enable、0 : Disable) Bit	BALANCE_CONTROL	Bit 0
転倒自動回復アクティブビット (1 : Enable、0 : Disable) Bit	AUTO_RECOVERY	1
最初のステップ移動時間自動演算ビット (1 : Enable、0 : Disable) Bit	FIRST_STEP_MOVE_TIME_AUTO_CALCULATE	2
動作を繰り返しアクティブビット (1 : Enable、0 : Disable) Bit	REPEAT_ENABLE	3
繰り返し区間下半身ポーズ左右切り替えビット (1 : Enable、0 : Disable) Bit	REPEAT_LOWER_POSE_EXCHANGE	4
繰り返し区間上半身ポーズ左右切り替えビット (1 : Enable、0 : Disable) Bit	REPEAT_UPPER_POSE_EXCHANGE	5
予備活性ビット	RESERVED	Disable) Bit6
予備活性ビット	RESERVED	Bit 7

StoreActivityProperty (uint8_t motion、uint8_t step_size、uint8_t repeat_start_step、uint8_t repeat_end_step、uint8_t repeat_acceleration_rate、uint8_t repeat_acceleration_count、uint8_t next_motion、uint8_t motion_function_enable_byte、char * motion_name)

- 機能：ロボットの動作プロパティの保存
- パラメータ
- motion：動作番号（範囲：0～199）
- step_size：動作ステップサイズ（範囲：0～23）
- repeat_start_step：繰り返し開始ステップ（範囲：1～22）
- repeat_end_step：繰り返し終了ステップ（範囲：1～22）
- repeat_acceleration_rate：繰り返し区間加速比（範囲：100～250、単位：0.01 x）
- repeat_acceleration_count：繰り返し区間加速回数（範囲：0～15）
- next_motion：次の接続動作（範囲：0～200、動作なし：255）
- motion_function_enable_byte：動作機能活性バイト

動作機能活性バイトの定義

説明	定義	Bit
バランス制御有効ビット (1 : Enable、0 : Disable) Bit	BALANCE_CONTROL	Bit 0
転倒自動回復アクティブビット (1 : Enable、0 : Disable)	AUTO_RECOVERY	1
最初のステップ移動時間自動演算ビット (1 : Enable、0 : Disable) Bit	FIRST_STEP_MOVE_TIME_AUTO_CALCULATE	Bit 2
動作を繰り返しアクティブビット (1 : Enable、0 : Disable) Bit	REPEAT_ENABLE	3
繰り返し区間下半身ポーズ左右切り替えビット (1 : Enable、	REPEAT_LOWER_POSE_EXCHANGE	4
繰り返し区間上半身ポーズ左右切り替えビット (1 : Enable、0 : Disable) Bit	REPEAT_UPPER_POSE_EXCHANGE	0 : Disable) Bit 5
予備活性ビット	RESERVED	6
予備活性ビット	RESERVED	Bit 7

- motion_name：アクション名（60文字）

) FeedBackRobotData (uint8_t feedback_num、protocol_header_t * recv_buf_head)

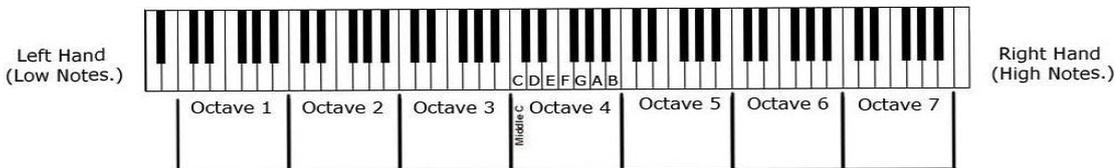
- 機能：ロボットデータ Feedback リクエスト
- パラメータ
- feedback_num：Feedback 要求データ ID

説明	定義	ID
コマンド正常受信確認	PACKET_feedback_ack	1
ロボットの現在の状態	PACKET_feedback_robot_status	2
ロボットアクチュエータ基準位置値	PACKET_feedback_zero_position	3
ロボットの基本設定	PACKET_feedback_base_parameter	4
ロボットの基本姿勢	PACKET_feedback_default_pose	5
ロボット現在姿勢	PACKET_feedback_pose_capture	6
ロボット動作を実行応答	PACKET_feedback_execute_robot_motion	7

- 機能 : LED の色出力
- パラメータ
- `_bright` : LED 全体の明るさ (範囲 : 0~255)
- `_red` : 赤色の明るさ (範囲 : 0~255)
- `_green` : 緑の明るさ (範囲 : 0~255)
- `_blue` : 青の明るさ (範囲 : 0~255)

: 0~255) `playSound (uint8_t_octave, uint8_t_syllable, uint8_t_beat)`

- 機能 : Piezo 音出力
- パラメータ
- `_octave` : オクターブ番号 (範囲 : 1~7)
- `_syllable` : 音番号 (範囲 : 0~11)
- `_beat` : 音の長さ (範囲 : 1~255、単位 : 4 msec) ※ オクターブ番号の定義



※ 番号定義

音	Note	Index
も	C	0
も	C#	1
レ	D	#2
レ	D#	3
米	E	#4
波	F	5
ファ	F#	6
ブラシ	G	#7
ブラシ	G#	8
と	A	#9
と	A#	10
時	B	#11

- #define PACKET_feedback_activity_property 120
- #define PACKET_feedback_motion_step 121

FeedBackActivityProperty (uint8_t motion、 protocol_store_activity_property_t *

recv_buf) protocol_store_activity_property_t 構成パラメータ

- motion : 動作番号 (範囲
- : 0~199) step_size : 動作ステップサイズ (範囲
- : 0~23) repeat_start_step : 繰り返し開始ステップ (範囲
- : 1~22) repeat_end_step : 繰り返し終了ステップ (範囲
- : 1~22) repeat_acceleration_rate : 繰り返し区間加速比 (範囲 : 100~250、単位 : 0.01
- x) repeat_acceleration_count : 繰り返し区間加速回数 (範囲
- : 0~15) next_motion : 次の接続動作 (範囲 : 0~200 、動作なし
- : 255) motion_function_enable_byte : 動作機能活性バイト

動作機能活性バイトの定義

説明	定義	Bit
バランス制御アクティブビット (1 : Enable、 0 : Disable) Bit	BALANCE_CONTROL	Bit 0
転倒自動回復アクティブビット (1 : Enable、 0	AUTO_RECOVERY	1
最初のステップ移動時間自動演算ビット (1 : Enable、 0 : Disable) Bit	FIRST_STEP_MOVE_TIME_AUTO_CALCULATE	: Disab le) Bit2
動作を繰り返しアクティブビット (1 : Ena	REPEAT_ENABLE	3
繰り返し区間下半身ポーズ左右切り替え ビット (1 : Enable、 0 : Disable) Bit	REPEAT_LOWER_POSE_EXCHANGE	ble、 0 : Disab le) Bit4
繰り返し区間上半身ポーズ左右切り替えビ ット (1 : Enable、 0 : Disable) Bit	REPEAT_UPPER_POSE_EXCHANGE	5
予備活性ビット	RESERVED	6
予備活性ビット	RESERVED	Bit 7

- motion_name : アクション名 (60 文字

) FeedBackMonoStep (uint8_t motion、 uint8_t step、 protocol_store_motion_step_t *

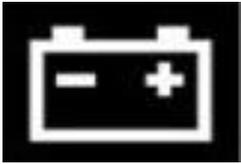
recv_buf) protocol_store_motion_step_t 構成パラメータ

- step_move_time : ステップ移動時間 (範囲 : 1~255、単位 : 0.01
- sec) step_stop_time : ステップ停止時間 (範囲 : 1~255、単位 : 0.01
- sec) step_led : ステップ LED の色番号 (範囲 : 0~63) は、付録の[LED 制御パレット]を参照
- step_led_enable : LED の色出力の有無ビット、「1」のときの色出力
- step_piezo : ステップ Piezo 音番号 (範囲 : 0~63) は、付録の[ピエゾ音番号札]参照
- step_piezo_enable : Piezo 音出力の有無ビット、「1」のとき、音出力

- step_reference_pitch : ステップ Pitch ジョーの角度 (範囲: -900~900、単位
- : 0.1°) step_reference_roll : ステップ Roll 参照角度 (範囲: -1800~1799、単位
- : 0.1°) position : アクチュエータの目標位置の値 (範囲: -511~511) 保存された配列のアドレス

付録

[バッテリー管理および充電]



本ロボットは、主電源に充電が可能な二次電池であるニッケル-メタル水素 (Nickel-metal hydride) バッテリーパックを使用します。

便利ですがメーカーの使用方法を遵守していないと、爆発や火災などの大事故につながることができ、取り扱いに細心の注意が必要です。

■ ニッケル-メタル水素 (Nickel-metal hydride) バッテリーパックの管理方法

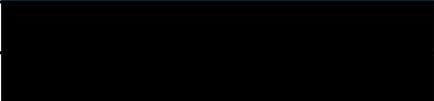
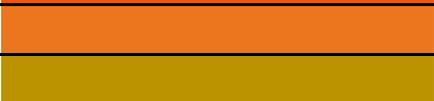
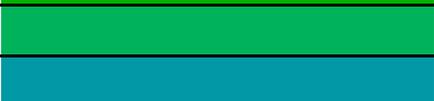
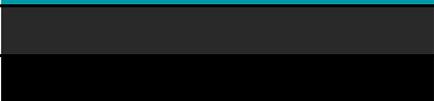
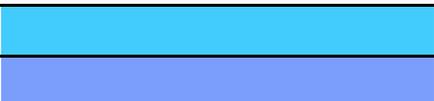
1. を使用した後、長期間保管する場合、50%程度の充電状態でバッテリーをコントロールボードから外して保管してください。
2. 外部列脆弱することで高熱が発生していない場所に保管します。
3. 冬季には、バッテリーが冷えすぎないように保管することをお勧め。常温保存をお勧めします。バッテリーの温度が冷えすぎたら、バッテリーが性能を 100%発揮できません。また、冬には、ランタイムを少し短く使用することをお勧めします。バッテリーが冷えるとカットオフが少し速くなるからです。
4. 本ロボットに含まれているニッケル-メタル水素電池は消耗品ですので、長期間使用性能が低下するため、交換が必要です。

■ 充電方法

1. 付属の充電器を充電ジャックに差し込むだけで充電を開始します。必ず純正の充電器を使用します。
2. 充電時間は約 70 分 (完全に放電されたバッテリー基準) 前後とバッテリー残量に応じて変更されることがあります。
3. バッテリーを使用して再充電をさせるとき、少なくとも 15~20 分程度のバッテリーを休ませてください。15~20 分程度休ませた後、温度が体温よりも低い感じられる場合は、充電をしなければなら効率的な充電が可能です。バッテリーの温度が高すぎる状態で、充電を行うと電圧 (V) のみアップ電流 (A) は、緩衝されていない場合があります。

【LED 制御パレット】

+

番号	色	(R、 G、 B) Value
0		(139, 146, 150)
1		(5, 50, 253)
2		(3, 39, 208)
3		(82, 67, 207)
4		(167, 37, 157)
5		(189, 27, 53)
6		(188, 34, 0)
7		(160, 39, 0)
8		(100, 63, 0)
9		(0, 135, 0)
10		(0, 121, 0)
11		(0, 105, 0)
12		(0, 81, 107)
13		(0, 0, 0)
14		(0, 0, 0)
15		(0, 0, 0)
16		(199, 203, 206)
17		(0, 143, 253)
18		(0, 157, 253)
19		(125, 102, 253)
20		(230, 55, 220)
21		(236, 40, 115)
22		(255, 80, 0)
23		(236, 118, 30)
24		(187, 146, 0)
25		(0, 192, 0)
26		(0, 179, 0)
27		(0, 179, 90)
28		(0, 153, 165)
29		(41, 41, 41)
30		(0, 0, 0)
31		(0, 0, 0)
32		(253, 250, 253)
33		(65, 204, 253)
34		(123, 158, 253)
35		(173, 146, 253)

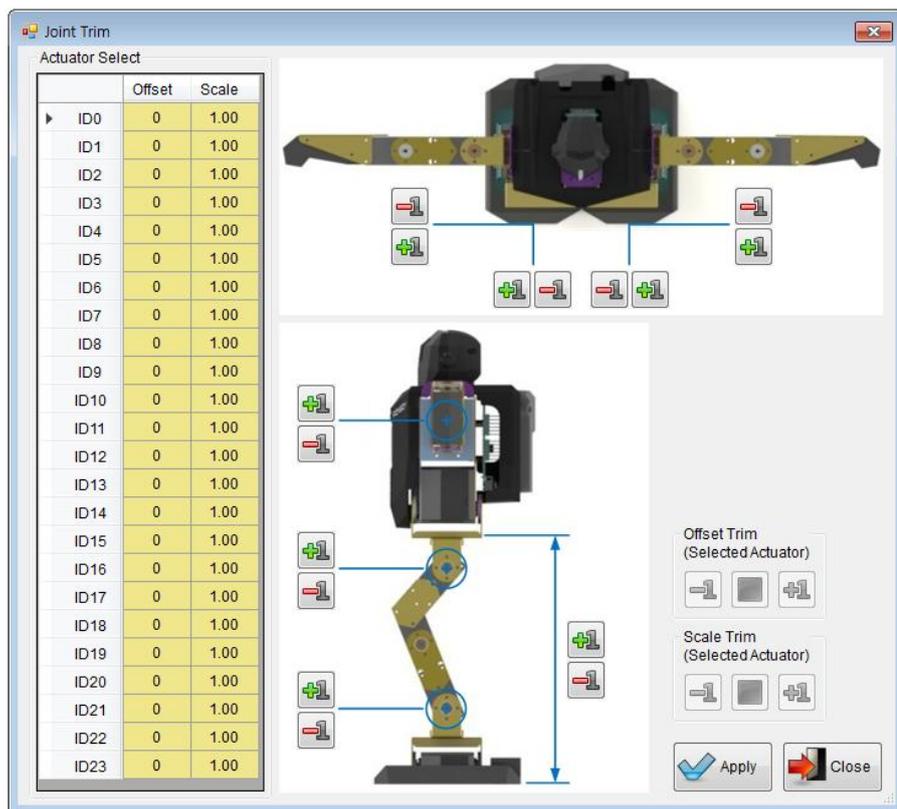
36		(255, 148, 253)
37		(255, 115, 172)
38		(255, 142, 107)
39		(254, 175, 90)
40		(254, 195, 0)
41		(198, 245, 28)
42		(101, 219, 107)
43		(99, 245, 172)
44		(0, 235, 234)
45		(103, 103, 103)
46		(0, 0, 0)
47		(0, 0, 0)
48		(255, 255, 255)
49		(177, 236, 253)
50		(200, 199, 253)
51		(228, 199, 253)
52		(254, 200, 252)
53		(247, 205, 230)
54		(247, 217, 193)
55		(253, 229, 193)
56		(253, 222, 150)
57		(226, 247, 139)
58		(194, 246, 138)
59		(188, 242, 224)
60		(0, 248, 253)
61		(214, 207, 207)
62		(0, 0, 0)
63		(0, 0, 0)

【圧電音番号札】

音番号	Octave (オクターブ)	Note	音
0	Octave 2	C	ド
1	Octave 2	C#	ド#
#2	Octave 2	D	レ
3	Octave 2	D#	レ#
#4	Octave 2	E	ミ
5	Octave 2	F	ファ
6	Octave 2	F#	ファ#
#7	Octave 2	G	ソ
8	Octave 2	G#	ソ#
#9	Octave 2	A	ラ
10	Octave 2	A#	ラ#
#11	Octave 2	B	シ
12	Octave 3	C	ド
13	Octave 3	C#	ド#
#14	Octave 3	D	レ
15	Octave 3	D#	レ#
#16	Octave 3	E	ミ
17	Octave 3	F	ファ
18	Octave 3	F#	ファ#
#19	Octave 3	G	ソ
20	Octave 3	G#	ソ#
#21	Octave 3	A	ラ
22	Octave 3	A#	ラ#
#23	Octave 3	B	シ
24	Octave 4	C	ド
25	Octave 4	C#	ド#
#26	Octave 4	D	レ
27	Octave 4	D#	レ#
#28	Octave 4	E	ミ
29	Octave 4	F	ファ
30	Octave 4	F#	ファ#
#31	Octave 4	G	ソ
32	Octave 4	G#	ソ#
#33	Octave 4	A	ラ
34	Octave 4	A#	ラ#
#35	Octave 4	B	シ
36	Octave 5	C	ド
37	Octave 5	C#	ド#

# 38	Octave 5	D·	レ
· 39	Octave 5	D#	レ#
# 40	Octave 5	E	ミ
41	Octave 5	F	ファ
42	Octave 5	F#	ファ#
# 43	Octave 5	G	ソ
44	Octave 5	G#	ソ#
# 45	Octave 5	A	ラ
46	Octave 5	A#	ラ#
# 47	Octave 5	B	シ
48	Octave 6	C	ド
49	Octave 6	C#	ド#
# 50	Octave 6	D·	レ
· 51	Octave 6	D#	レ#
# 52	Octave 6	E	ミ
53	Octave 6	F	ファ
54	Octave 6	F#	ファ#
# 55	Octave 6	G	ソ
56	Octave 6	G#	ソ#
# 57	Octave 6	A	ラ
58	Octave 6	A#	ラ#
# 59	Octave 6	B	シ
60	Octave 7	C	ド
61	Octave 7	C#	ド#
# 62	Octave 7	D·	レ
· 63	Octave 7	D#	レ#

[モーションデータ関節補正・ Joint Trim Setting]



[関節補正ウィンドウ]

Actuator Select	Offset	Scale
ID0	0	1.00
ID1	0	1.00
ID2	0	1.00

アクチュエータ（サーボ）の選択ウィンドウで、補正しようとする関節を選択します。



選択されたアクチュエータの Offset 値を調整します。選択されたモーションデータのすべてのステップの関節値を Offset 値だけ増減します。



Offset 値を「1」増加させます。「511」まで変更されます。



Offset 値を「1」減少させます。「-511」まで変更されます。



Offset 値を「0」に変更します。



選択されたアクチュエータの Scale 値を調整します。 Offset 値に Scale 値だけの割合で適用します。



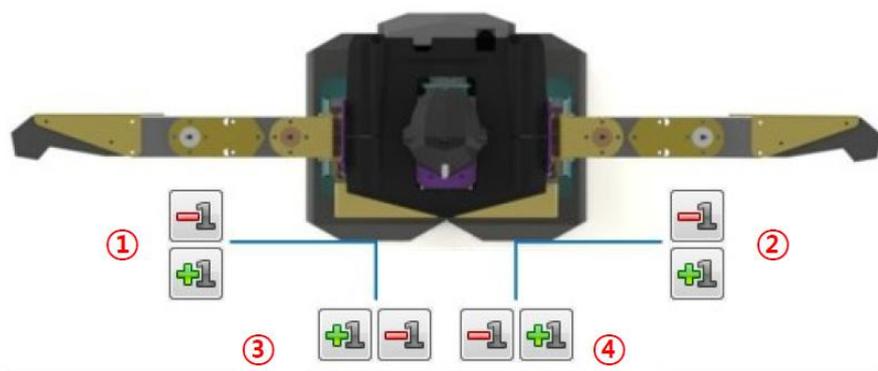
Scale の値を「0.01」に増加させます。「1.50」まで調節されます。



Scale の値を「0.01」減少させます。「0.50」まで調節されます。



Scale の値を「1.00」に変更します。



1 重心前後調節（右）



ID2 の Offset 値を「1」の増加、ID4 の Offset 値を「1」減少させます。



ID2 の Offset 値を「1」減少、ID4 の Offset 値を「1」増加させます。

2 重心前後調節（左）



ID8 の Offset 値を「1」の増加、ID10 の Offset 値を「1」減少させます。



ID8 の Offset 値を「1」減少、ID10 の Offset 値を「1」増加させます。

3 重心左右調節（右）



ID1 の Offset 値を「1」の増加、ID5 の Offset 値を「1」減少させます。



ID1 の Offset 値を「1」減少、ID5 の Offset 値を「1」増加させます。

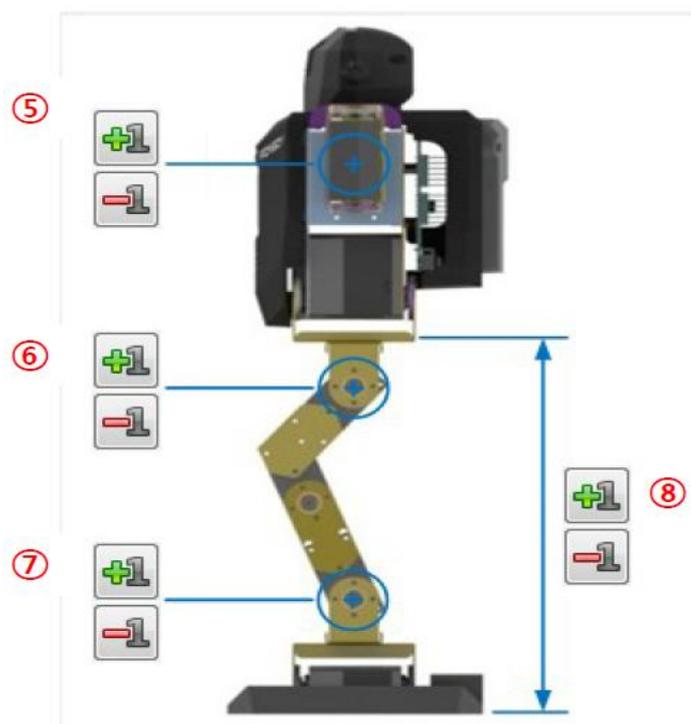
4 重心左右調節（左）



ID7 の Offset 値を「1」の増加、ID11 の Offset 値を「1」減少させます。



ID7 の Offset 値を「1」減少、ID11 の Offset 値を「1」増加させます。



7 足首前後調節



ID4 の Offset 値を「1」の増加、ID10 の Offset 値を「1」減少



ID4 の Offset 値を「1」減少、ID10 の Offset 値を「1」増加

8 足の長さの調節



ID3、ID9 の Offset 値を「2」の増加、ID2、ID4、ID8、ID10 の Offset 値を「1」減少



ID3、ID9 の Offset 値を「2」の減少、ID2、ID4、ID8、ID10 の Offset 値を「1」増加

5 両腕の前後調節



ID12 の Offset 値を「1」の増加、ID17 の Offset 値を「1」減少



ID12 の Offset 値を「1」減少、ID17 の Offset 値を「1」増加

6 上体の前後調節



ID2 の Offset 値を「1」の増加、ID8 の Offset 値を「1」減少



ID2 の Offset 値を「1」減少、ID8 の Offset 値を「1」増加

[モーションライブラリ表

】※100～121 回のモーションは、工場出荷前のバランスセッティングまでチェックして出荷します。残りのモーションは、各ロボットの組立状態に合わせて、お客様が適切に設定して使用します。

モーション番号	モーション名前	モーション説明
100	Walk Forward	前進
101	Turn L	左方向へターン
102	Turn R	右方向へターン
103	Walk Backward	後進
104	Side Step L	立っている状態で、左横移動
105	Side Step R	立っている状態で、右横移動
106	Stand	立ち上がる
107	Sit Down	座る
108	Get Up Forward	バック倒れた状態で体を前に動かして起こる
109	Get Up Backward	今後倒れた状態で体を後ろに動かして起こる
110	Body Side Tilt	両腕を広げ、左右に上体を傾け
111	Provoke	怒ってたくましく待機
112	Pushup	腕立て伏せ
113	Side Kick L	左にサイドキック
114	Side Kick R	右にサイドキック
115	Celebrate the Victory	勝利セレモニー
116	Bow Down	腰を下げて丁寧に挨拶する
117	Arm Whirl	両腕前後を振る
118	Clap	拍手打撃
119	Greet with Wave	右手含ま振る
120	Wing Flap L	左足のみ支持して新しいような姿勢で羽ばたき
121	Wing Flap R	右足のみ支持し、新しいのような姿勢で羽ばたき
122	Side Tumbling L	立っている状態で、左方向に転がる
123	Side Tumbling R	立っている状態で、右方向に転がる
124	Front Punch	一歩今後ながらパンチ攻撃
125	Side Punch L	左に横にパンチ攻撃
126	Side Punch R	右に横にパンチ攻撃
127	Rear Attack	戻る攻撃
128	One Step Forward	一歩前に歩く
129	One Step Backward	一歩後ろに歩く
130	Step Forward Punch	これから大きくしたクロスの移動後、腕を交互にジルギ
131	Front Tumbling	今後転がり
132	Back Tumbling	戻る転がり
133	Arm Wave	両腕を広げウェーブ追記
134	Butterfly	両腕を広げ漕ぎ
135	Side Move Punch L	左に移動した後のパンチ攻撃

136	Side Move Punch R	右に移動した後のパンチ攻撃
137	Large Step Turn L	左に大きく回り
138	Large Step Turn R	右に大きく回り
139	Front Lift Attack	今後抱い高揚
140	Side Lift Attack L	左相手を横に入って第キギ
141	Side Lift Attack R	右相手を横に入って第キギ
142	Back Drop Attack	保持戻るマツチ機
143	Sit Side Move L	座っている状態で、左に移動し
、144	Sit Side Move R	座っている状態で、右に移動
145	Sit Front Attack	座っ今後パンチ
146	Sit Side Attack L	座っ左に横ジルギ
147	Sit Side Attack R	座っ右に横予防する
148	Sit Around Attack	座って両腕でフィ返し打撃
149	Stomping Move Forward	早送り歩く
150	Stomping Turn L	左に素早く回り
151	Stomping Turn R	右に素早く回り
152	Stomping Move Backward	戻るすぐに歩くこと
153	Ball Tackle Front	ボールの前で、今後取り組む
154	Ball Front Kick L	ボールを左の足先に次期
155	Ball Front Kick R	を右足今後、次期
156	Ball Heel Kick	ボールを右足でかかるとに戻る次期
157	Ball Side Kick L	ボールを左足で左にサイドキック
158	Ball Side Kick R	ボールを右足で右にサイドキック
159	Stomping Move Attack	早送り移動しながら腕交互に追い越し
160	Jump	所定の位置にジャンプする
161	Lie Face Down	今後うつぶせ
162	Lie Face Up	戻る横になる
163	Hold and Topple Attack L	キャッチ左に進みツリーた
164	Hold and Topple Attack R	キャッチ右に進み、ツリーた
165	Rise Up From Face Down	今後うつ伏せの状態を起こる
166	Crawl Forward	這わに前進
167	Crawl Side L	這わに左に回り
168	Crawl Side R	這いで右に回り
169	Crawl Backward	這いで後進
170	Crawl and Rise Up	今後伏せ這い進む後起こる
171	Keeper Stance	ゴールキーパーがボールを防ぐ姿勢
172	Keeper Front Block	ゴールキーパーが両腕着ボールを防ぐ
173	Keeper Side Block L	キーパーが両腕広げて左脚伸びてくるボールを防ぐ
174	Keeper Side Block R	キーパーが両腕広げて右足伸びてくるボールを防ぐ
175	Keeper Ball Kick	ゴールキーパーがボールを拾って発売し、次期

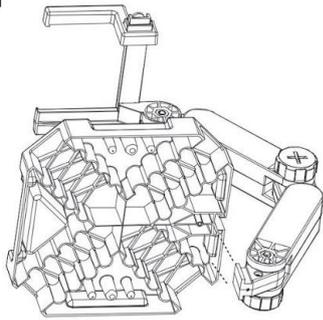
176	Keeper Low Block	ゴールキーパーが手足を両側に伸ばし、下のボールを防ぐ
177	Breathe Action	息を動作
178	One Leg Squat L	右足を持って左足で座った起こる
179	One Leg Squat R	左足を持って右足で座った起こる
180	Dance KPOP 1	韓国歌謡ダンス 1
181	Dance KPOP 2	韓国歌謡ダンス 2
182	Dance KPOP 3	韓国歌謡ダンス 3
183	Dance KPOP 4	韓国歌謡ダンス 4
184	Dance KPOP 5	韓国歌謡ダンス 5
185	Dance KPOP 6	韓国歌謡ダンス 6
186	Dance KPOP 7	韓国歌謡ダンス 7
187	Dance KPOP 8	韓国歌謡ダンス 8
188	Dance KPOP 9	韓国歌謡ダンス 9
189	Heart	両腕でハート形の作成上下に動く
190	Tiptoe Move Forward	つま先で前で歩く (ちょこちょこ
) 191	Tiptoe Turn L	つま先で左回りに回転
192	Tiptoe Turn R	つま先で右回りに回転
193	Tiptoe Move Backward	つま先に戻る歩く
194	Flying Stand Up	羽ばたきで起きる
195	Large Side Step L	左に大きく横移動
196	Large Side Step R	右に大きく横移動
197	Flying Sit Down	羽ばたきながらしゃがむ
198	Large Step Forward	大きく前進
199	Large Step Backward	大きく後進

IR-RS01 [IRONBOY ROBOT STAND] Assembly



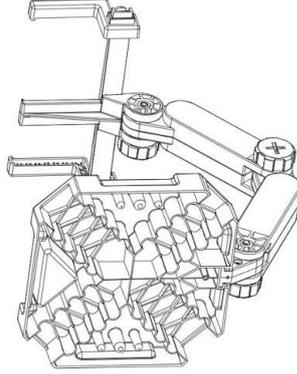
<p>1</p> <p>Assemble left and right arm</p>	<p>2</p> <p>Join two arms together.</p>	<p>3</p> <p>Use screw and nut to fasten the arms.</p>	<p>4</p>
<p>5</p> <p>Assemble the hinge as shown in the picture.</p>	<p>6</p>	<p>7</p> <p>Insert hooks to the hinge</p>	
<p>8</p>	<p>9</p> <p>Join the arms to the hinge</p>	<p>10</p>	
<p>11</p> <p>Put these parts together.</p>	<p>12</p> <p>Assemble #11 to the bottom part of #10</p>	<p>13</p> <p>Use screw and nut to fasten the arms.</p>	

14

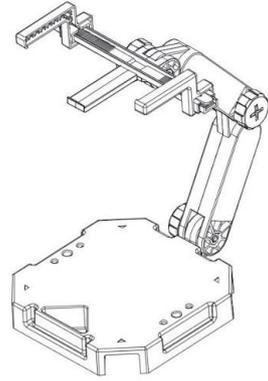


Fasten #13 to the base of the robot stand.

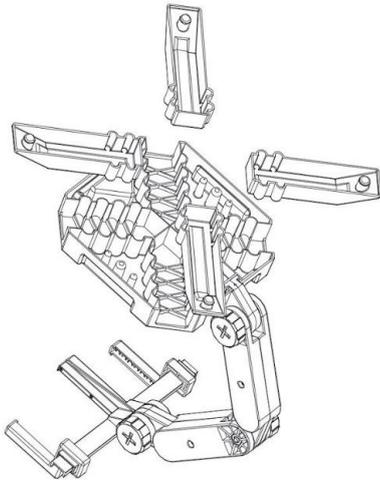
15



16

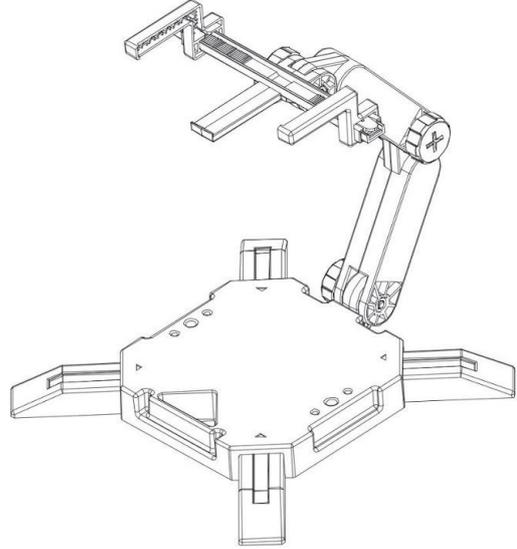


17



Place the legs to the bottom of the base properly. The length of legs is variable.

18



Place IRONBOY on the hinge

Adjust the hooks to secure the robot

