

SONY®

4-659-404-01(1)



AIBO Master Studio

ERF-PC03 日本語版

チュートリアルガイド

目次

はじめに

チュートリアルを始める前に	8
マニュアルについて	8
オンラインマニュアルについて	8
準備しよう	8
チュートリアルの構成	9
ビヘイビアを作ろう！ 編	9
アクションを作ろう！ 編	10
チュートリアルで使うファイル	12
アクションコンポーザーのファイル	12

ビヘイビアを作ろう！

Lesson 1 AIBOのセンサーを押すと、座って 立って 伏せる	14
クイックビヘイビアアレンジャーを起動しよう	14
順次実行タイプでプログラムを作成しよう	15
プログラムを“メモリースティック”に保存しよう	16
AIBOで実行しよう	17
Lesson 2 頭を押すと座り、「すたーと」で歩く	18
クイックビヘイビアアレンジャーを起動しよう	19
イベント実行タイプでプログラムを作成しよう	19
プログラムを“メモリースティック”に保存しよう	21
AIBOで実行しよう	21
Lesson 3 歩いてから おじぎ	22
準備しよう	22
ビヘイビアを新規作成しよう	23
アクションボックスを作成しよう	24
「立つ」アクションボックスを作成する	24
「歩いて、おじぎをする」アクションボックス作成	26
ボックスを接続しよう	28
プログラムを保存し、AIBOで実行しよう	30
ワイヤレスLANを使ってみよう	30

Lesson 4 頭を押されたら おじぎ	33
Lesson3で作成したファイルを表示しよう	34
ボックスを作成しよう	35
ボックスをつないでプログラムをループさせよう	38
プログラムを保存し、AIBOで実行しよう	41
Lesson 5 「あいぼ」と呼ぶとおじぎ、「こんにちは」であいさつ	42
アクションライブラリを準備しよう	43
ボックスをまとめて、グループを作成しよう	45
「頭タッチ?」と「1秒待ち」をグループに移動する	45
ブランチボックス・アクションボックスを編集する	46
アクションファイルをプログラムに盛り込む	51
グループを組み込んでプログラムを完成させよう	53
プログラムを保存し、AIBOで実行しよう	55
Lesson 6 同時に複数のアクションを実行する！	56
現在のプログラムを確認しよう	57
複数のアクションを同時に実行させよう	58
プログラムを保存し、AIBOで実行しよう	61
Lesson 7 高度なワザをマスターしよう！	63
ファイルを準備する	63
メインプログラムを確認しよう	63
「ボール追跡」グループボックスのプログラムを確認しよう	64
実行中のコマンドの数によって、アクションを変える	66
「キック」グループプログラムを確認しよう	67
端子の順序を変更しよう	68

アクションを作ろう！

ウィンドウ構成	70
Lesson 1 アクションライブラリを作る！	72
準備しよう	73
ライブラリを作成しよう	73
アクションを作成しよう	74
既存のアクションをライブラリに追加しよう	76
“メモリースティック”に保存しよう	77
ワイヤレスLANを使って保存してみよう	78
Lesson 2 作成したアクションを確認しよう！	80
PC上で、アクションと素材を確認しよう	81
モーションを確認	82
サウンドを確認	83
LEDを確認	84
AIBOで確認しよう	85
ライブラリに未登録のアクションや素材を確認しよう	86
アクションを確認	86
素材を確認	87
Lesson 3 素材をちょっと直してみる！	90
素材を準備しよう	91
モーションを編集しよう	92
ポーズを編集	92
キーフレームを追加	93
サウンドを編集しよう	94
サウンドファイルの編集	94
サウンドの開始位置の調整	96
LEDを編集しよう	96
LEDデータを追加する	96
LEDの点滅を繰り返す	98
アクションファイルを保存しよう	100

Lesson 4 オリジナルな素材を作ろう！	101
アクションを作成しよう	102
モーションを作成しよう	102
サウンドを作成しよう	105
素材を組み合わせよう	107
Lesson 5 曲に合わせて、LEDを点灯させる！	108
Lesson 6 これがアクション作りのコツだ！	112
全体として	112
モーション	112
モーションの機種別情報	116
サウンド WAVE編	117
サウンド MIDI編	117
LED	118

本書を使って、より効果的な楽しいAIBOのプログラムを作ってください！

チュートリアルを始める前に

AIBOマスタースタジオで用意されているマニュアルの種類と、チュートリアルを始める前の準備について説明します。

マニュアルについて

AIBOマスタースタジオには、2種類のマニュアルがあります。

チュートリアルガイド(本書)

AIBOマスタースタジオの使いかたを、例を用いて説明しています。AIBOマスタースタジオを初めてお使いになる場合は、このマニュアルに従って一通り操作し、AIBOマスタースタジオに慣れてください。

ユーザーガイド

AIBOマスタースタジオの概要をはじめ、インストールや設定、またAIBOで動作させるために必要な情報、AIBOマスタースタジオを構成する各プログラムの使いかたを説明しています。

ご注意

本文中のイラストや画面は、主にAIBO ERS-210を使用しています。必要に応じてERS-220やERS-310シリーズの画面を使用しています。

オンラインマニュアルについて

チュートリアルガイドとユーザーガイドは、PDF形式のオンラインマニュアルとしてもご覧いただけます。オンラインマニュアルの見かたは、ユーザーガイド(3ページ)をご覧ください。

準備しよう

AIBOマスタースタジオのインストールや、“メモリースティック”の準備はできていますか？

また、ワイヤレスLANをお使いの方は、あらかじめ設定が必要です。準備が終了していない方は、ユーザーガイドの「準備」(17ページ)をご覧ください。

チュートリアル構成

チュートリアルは、次の2部構成になっています。

ビヘイビアを作ろう! 編

クイックビヘイビアアレンジャーとビヘイビアアレンジャーを使って、プログラムを作成する方法を学習します。

初級コース

Lesson1と2では、ウィザード形式でプログラムが作成できる「クイックビヘイビアアレンジャー」を使います。

Lesson3では、ビヘイビアアレンジャーの基礎的な操作を学べます。

Lesson 1 AIBOのセンサーを押すと、座って 立って 伏せる

クイックビヘイビアアレンジャーで、「AIBOのセンサーを押すと、AIBOが設定したアクションを順番に実行する」というプログラムを簡単に作ってみましょう!

Lesson 2 頭を押すと座り、「すたーと」で歩く

クイックビヘイビアアレンジャーで、「AIBOのセンサーを押したり、設定した言葉を話しかけたりすると、言葉やセンサーによって違ったアクションをする」というプログラムを作成してみましょう。ウィザード形式なので、Lesson1と同様に簡単です!

Lesson 3 歩いてから おじぎ

いよいよビヘイビアアレンジャーを使ってAIBOのプログラムを作ってみます。ビヘイビアアレンジャーでの基本的なプログラムの作成方法と、ワイヤレスLANを使ってAIBOにプログラムを実行させる方法を学習しましょう。

中級コース

ビヘイビアアレンジャーの基礎的な操作を習得している方向けです。音声認識や独自のアクションを使ったプログラムをすることで、プログラムの分岐やグループ(サブルーチン)、変数の使い方を学べます。

Lesson 4 頭を押されたら おじぎ

ブランチボックスの使い方とセンサー情報を使ったプログラムを学習します。プログラムをループさせて、繰り返し実行する方法も覚えましょう。Lesson3で作成したプログラムを編集してプログラムを作成します。

Lesson 5 「あいぼ」と呼ぶとおじぎ、「こんにちは」であいさつ

グループの作り方、音声認識を使ったプログラム、アクションコンポーザーのアクションをプログラムに組み込む方法を学習しましょう。少し複雑に見えますが、ひとつずつテクニックを学んでいくことができます！

上級コース

上級でのテクニックをマスターすることで、ペットのように行動するアプリケーションやゲームなどを作ることも可能です！

Lesson 6 同時に複数のアクションを実行する！

ロボット的なプログラミング手法として、同時に複数の命令を実行する方法を学習しましょう。歩きながらセンサー情報を取得して前進から後進にAIBOの動きが変わるプログラムを作成していきます。

Lesson 7 高度なワザをマスターしよう！

ビヘイビアアレンジャーでプログラムを作成するときの、高度なテクニックを紹介します。「ピンクボールを探して追跡する」プログラムの解説に従い、プログラミング手法を学んでいきましょう。

アクションを作ろう！ 編

アクションコンポーザーを使って、ビヘイビアアレンジャーで使うアクションライブラリや、アクションファイル、素材を作成する方法を学習します。

初級コース

すでにある素材をそのまま使ったりアレンジして、アクションコンポーザーの基礎となる操作フローを学習します。

Lesson 1 アクションライブラリを作る！

ビヘイビアプログラムで使うために必要な操作を覚えます。既存のファイルを組み合わせ、アクションライブラリとアクションを作ってみましょう。

Lesson 2 作成したアクションを確認しよう！

作成したアクションファイルのアクションをPCの画面上や、実際のAIBOで動かしてみましょう。サウンドや動きを目や耳で確認することは、いい作品を作る上でも大切です！

Lesson 3 素材をちょっと直してみる！

既存のアクションファイルの素材を手直ししながら、素材やアクションファイルを編集する方法を覚えましょう！

中級コース

素材作りに入りこんでいきます。ここで素材作りの操作を学習します。

Lesson 4 オリジナルな素材を作ろう！

いよいよ、新規に素材を作る方法を学習します。

素材を作れるようになったら、もう立派なAIBOクリエイターです。

Lesson 5 曲に合わせてLEDを点灯させよう！

さらに一歩踏み込んで、素材を同期させてみましょう。

例えば、ウインクさせたり、おしゃべりに合わせて口が動くリップシンクをさせてみたり。音楽に合わせて踊らせるのも、楽しそう！

上級コース

効率よく効果的なアクションを作るテクニックを身に付けましょう！

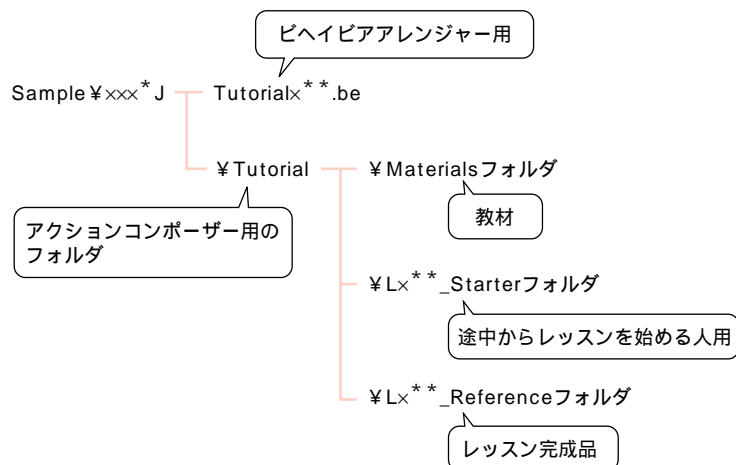
Lesson 6 これがアクション作りのコツだ！

あなたのオリジナルのAIBOのアクションを、さらにグレードアップさせましょう！「いかにAIBOで効果的に見える作品を作るか」をテーマに、アクション作成のコツを紹介します。

チュートリアルで使うファイル

チュートリアルで使うファイルはすべて「Sample」フォルダの中にあります。

¥ AIBOマスタースタジオのインストールフォルダ ¥ Sample



チュートリアルで使うファイルは、いつでもレッスンで使えるよう、上書きしないようにしましょう。

* AIBOの機種名(機種ごとにフォルダが分かれています)。チュートリアルでは音声認識は日本語を使いますので「J」のついたフォルダを使ってください。
** レッスン番号

アクションコンポーザーのファイル

レッスンを始める前に、あらかじめ「Materials」フォルダごと、マイドキュメントなど自分のワークスペースにコピーしてからお使いください。

途中のレッスンから学習を始める方は

「Lx**_Starter」フォルダごと自分のワークスペースにコピーして、レッスンを始めてください。

例えば、Lesson2から始める場合は、「L2_Starter」をコピーしてお使いください(ただし、ファイル名が、レッスン内の説明文と一部異なります)。

作ったものを確認する

各レッスンで作成するファイルと同じ内容の完成ファイルは、「Lx**_Reference」フォルダの中にあります。作ったものが正しくできているか確認するときに、参照してください。

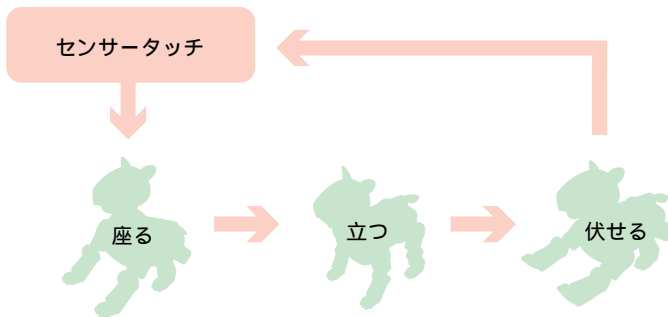
では、さっそくレッスンを始めましょう！

AIBOのセンサーを押すと、座って 立って 伏せる

目標：クイックビヘイビアアレンジャー*の「順次実行タイプ」を学習する

*クイックビヘイビアアレンジャーを使うと、ウィザード形式で簡単にプログラムを作成できます。ユーザーガイド(25ページ)を参照してください。

プログラム概要：
センサーを押すと、設定したアクションを順番に再生する



プログラム作成・実行手順：

- クイックビヘイビアアレンジャー起動(14ページ)
- 順次実行タイプでプログラム作成(15ページ)
- プログラムを“メモリースティック”に保存(16ページ)
- AIBOで実行(17ページ)

クイックビヘイビアアレンジャーを起動しよう

- 1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[クイックビヘイビアアレンジャー]を選ぶ

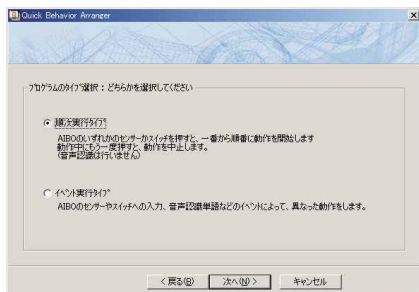
クイックビヘイビアアレンジャーが起動します。
AIBOの機種を選択する画面が表示されます。



2 お使いのAIBOを選び、[次へ]をクリックする

プログラムを実行させたいAIBOの「機種名(J)」(日本語版)を選択してください。

プログラムのタイプを選択する画面が表示されます。



日本語版と英語版が選択できます。日本語版では、日本語の音声認識を使ったプログラムを、英語版では英語を使ったプログラムを作成できます。選択するコマンドの説明を日本語表示するためにも、ここでは日本語を選んでください。

順次実行タイプでプログラムを作成しよう

1 順次実行タイプを選び、[次へ]をクリックする

AIBOのアクションを設定する画面が表示されます。

イベント実行タイプは、Lesson2で学習します。

2 「1番目」で「SIT[座る]」を選ぶ



センサーを押した後にAIBOが最初に行うアクションを設定します。

3 「2番目」に「STAND[立つ]」を、「3番目」に「LIE[伏せる(1)]」を選ぶ



手順2で設定した「座る」に続けて行うアクションを設定します。

4 [次へ]をクリックする

「プログラムの完了」画面が表示されます。

プログラムを“メモリースティック”に保存しよう

1 付属の“メモリースティック”を“メモリースティック”ドライブに挿入し、[保存]をクリックする

「名前を付けて保存」ダイアログボックスが表示されます。

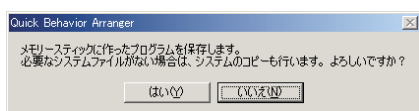
2 ファイル名を付けて保存する

「Lesson1.be」というファイル名で、ご自分のワークスペースに保存してください。

保存したファイルは、ビヘイビアアレンジャーで編集できます。

3 続いて、“メモリースティック”に保存する

確認のダイアログで[はい]をクリックしてください。



プログラムを保存する前に、付属の“メモリースティック”に必要なシステムをコピーしておく習慣をつけましょう。(クイックビヘイビアアレンジャーでは、システムをコピーしていない場合は、作ったプログラムと一緒にコピーできます。)

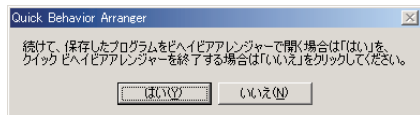
ご注意

“メモリースティック”ドライブを設定(ユーザーガイド20ページ)していない場合は、ドライブを指定する画面が表示されます。ドライブ名を入力してください。

設定したプログラムの内容を“メモリースティック”に保存して、AIBOで実行できるようにします。

“メモリースティック”に保存したファイルは、ビヘイビアアレンジャーでは編集できません。

書き込みが終了すると次のダイアログが表示されます。



4 [いいえ]をクリックする

プログラムの作成・保存が終了します。

このレッスンではビヘイビアランジャーでプログラムを編集しませんので、[いいえ]を選択してください。

AIBOで実行しよう

作成したプログラムをAIBOで実行します。

1 “メモリースティック”をAIBOに挿入し、ポーズボタンを押してAIBOを起動する

AIBOが完全に起動して、のび運動をし終わるまでお待ちください。

2 AIBOのタッチセンサーを押す

「すわる 立つ ふせる」を順番に実行します。
最後までプログラムを実行すると、AIBOの動きが止まります。

どのタッチセンサーを押しても良いです。

3 もう一度、タッチセンサーを押す

AIBOは再び最初からプログラムを実行します。

AIBOが動いている最中にセンサーを押すと、AIBOは動きを止めます。再度センサーを押すと最初から実行します。

さあ、うまくできたかな？

クイックビヘイビアランジャーを使っのプログラミングは、意外と簡単だったでしょう？ このように順次実行タイプでは、センサーを押すと設定したアクションを順番に実行するプログラムを作ることができます。

続いて、クイックビヘイビアランジャーのもうひとつの機能、イベント実行タイプを使ってプログラミングしてみましょう！

頭を押すと座り、「すたーと」で歩く

目標：クイックビヘイビアアレンジャーの「イベント実行タイプ」
を学習する

プログラム概要：

センサーや音声認識によって違うアクションを再生する

タッチセンサー

頭タッチ

座る

音声認識

「すたーと」

50cm
歩く

「ひだりきック」

キック

その他の条件

ピンクボール見せる

立つ

「ふせて」

伏せる

プログラム作成・実行手順

- クイックビヘイビアアレンジャー起動(19ページ)
- イベント実行タイプでプログラム作成(19ページ)
- プログラムを“メモリースティック”に保存(21ページ)
- AIBOで実行(21ページ)

クイックビヘイビアアレンジャーを起動しよう

クイックビヘイビアアレンジャーを起動し、AIBOの機種を選択します。必ず、「日本語版(J)」を選んでください。
プログラムのタイプを選択する画面が表示されます。

詳しくは、Lesson1(14ページ)
参照。



イベント実行タイプでプログラムを作成しよう

1 イベント実行タイプを選び、[次へ]をクリックする

センサー・スイッチへの入力とアクションを設定する画面が表示されます。

2 以下の項目を選び、[次へ]をクリックする

	センサー・スイッチへの入力	動作
ERS-210	頭センサーが押された	SIT[座る]
ERS-220	ヘッドタッチセンサーが押された	SIT[座る]
ERS-310シリーズ	顔が下を向くように押す	SIT[座る]

AIBOが左のボックスで選択した
入力を検知すると、右のボックス
で設定したアクションを実行しま
す。



3 「音声認識単語」と「動作」に以下の項目を選び、[次へ] をクリックする

音声認識単語	動作
すたーと	WALK:0:500[50cm歩く]
ひだりきック	KICK:30:1000[左キックする]
ふせて	LIE[伏せる]



左のボックスで選択した単語を話しかけると、右のボックスで選択したアクションをAIBOが実行します。ここでは、以下の設定をしました。

- ・「すたーと」 50cm歩く
- ・「ひだりきック」 キックする
- ・「ふせて」 伏せる

4 1段目の「その他のイベント」に「ピンクボールを見つけたら」を、「動作」に「STAND[立つ]」を選び、[次へ] をクリックする



左のボックスで選択した条件に合致すると、右のボックスで選択したアクションをAIBOが実行します。ここでは、「ピンクボールを見つけたら立ち上がる」というコマンド(命令)を設定したことになります。

プログラムを“メモリースティック”に保存しよう

Lesson1と同様に、プログラムを“メモリースティック”に保存しましょう。「Lesson2.be」というファイル名で保存してください。

AIBOで実行しよう

AIBOでプログラムを実行し、設定したとおりにAIBOがアクションするか確認してみましょう！

頭を押す* 座る

「すたーと」と話しかける 50cm歩く

「ひだりきっく」と話しかける キックする

「ふせて」と話しかける 伏せる

ピンクボールを見せる 立つ

このようにイベント実行タイプでは、入力イベント(センサー・音声認識・その他の入力)によって異なるアクションをするプログラムを作ることができます。

これで、クイックビヘイビアアレンジャーの使い方はマスターできました！

つぎは、いよいよ、ビヘイビアアレンジャーでプログラムを作り、使い方の基礎を学んでください。

Lesson1の「AIBOで実行しよう」(17ページ)参照。

*入力の方法は19ページの手順2参照。

AIBOが音を出している間は音声認識しません。また、動いているときは、言葉を聞きとりにくい場合があります。

歩いてから おじぎ

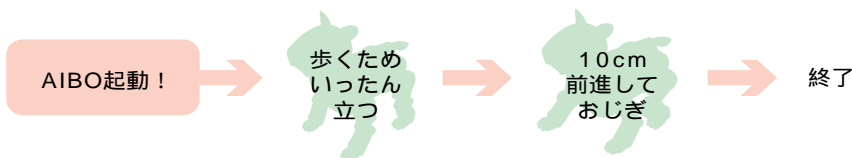
目標：次の2点を習得する

- ビヘイピアアレンジャー*の基本操作と、連続したアクションを再生するプログラムの作成方法
- ワイヤレスLANの使い方

* コマンドを入力したボックスを並べて、AIBOの行動プログラムを作るためのツールです。詳しくはユーザーガイド(29ページ)を参照。

プログラム概要：

AIBOを起動すると、立ち上がり、10cm前進しておじぎをする



プログラムの作成・実行手順：

- 準備(22ページ)
- ビヘイピアの新規作成(23ページ)
- アクションボックス**でコマンド(命令)入力(24ページ)
- 1「立つ」コマンド
- 2「歩いておじぎをする」コマンド
- 個々のアクション接続(28ページ)
- プログラム保存・AIBOで実行(30ページ)
- ワイヤレスLANで実行(30ページ)

** アクションボックスについては、ユーザーガイド(32ページ)参照。

準備しよう

- Lesson1と2で使用した“メモリースティック”をご用意ください。
- ワイヤレスLANをご利用の方は、設定を行ってください。設定方法について詳しくは、ユーザーガイド(23ページ)をご覧ください。

ワイヤレスLANは、Lesson1と2で使ったクイックビヘイピアアレンジャーでは使用できません。

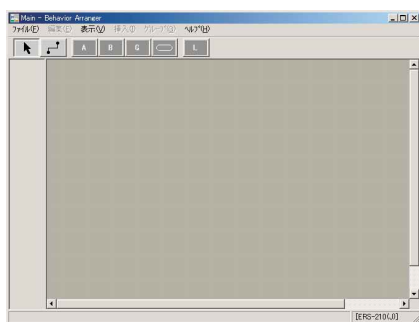
Lesson3から始める方は、AIBOの機種に適したシステム(日本語版)を“メモリースティック”にコピーしておいてください。

ビヘイビアを新規作成しよう

ビヘイビアを新規に作成して、プログラムを作っていきます。

- 1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[ビヘイビア アレンジャー]を選ぶ

ビヘイビアアレンジャーが起動し、編集ウィンドウが表示されます。



- 2 メニューの[ファイル]-[新規作成]を選ぶ

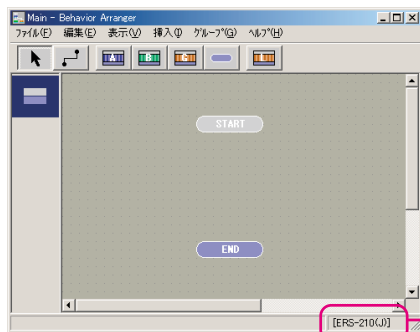
AIBOの機種を選択する画面が表示されます。

ビヘイビアを新規作成します。

- 3 お使いのAIBOを選ぶ

「START」と「END」ターミナルボックス*が表示されます。

ここでは日本語版(J)を選択してください。日本語版・英語版について詳しくは、15ページをご覧ください。



ここに選んだ機種名(J)が表示されます。

* ターミナルボックスについては、ユーザーガイド(33ページ)参照。

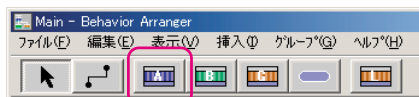
アクションボックスを作成しよう

アクションボックスにコマンドを入力していきます。

はじめに「立つ」アクションボックスを、次に「歩いておじぎをする」アクションボックスを作成します。

「立つ」アクションボックスを作成する

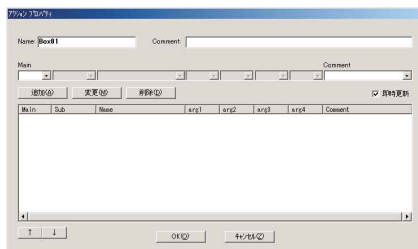
1 [アクションボックス]ボタン をクリックする



AIBOを歩かせるためには、最初に立たせる必要があります。このため、まず「立つ」アクションボックスを作成します。

2 「START」と「END」ターミナルボックスの間でクリックする

クリックした位置に、アクションボックスが置かれます。
コマンドを入力するアクションプロパティが表示されます。



3 アクションボックスの名前と内容を入力する



Name : 立つ

Comment : AIBOを立ち上げらせる

Nameに入力した名前は、アクションボックスに表示されます。
(全角で7文字を超えると、省略して表示)

ボックス名で表現しきれないようなプログラムの内容は、Comment欄に入力しておくとい良いでしょう。

4 「STAND(立つ)」というシステムアクションを再生するコマンドを入力する

Sub: ACTION(システムアクション)

Main: PLAY
(動作)

Name: STAND
(立つ)

システムアクションとは、AIBO マスタースタジオで用意されているアクションです。

PLAY-ACTIONを選ぶと、システムアクションが選択できるようになります。

5 [追加]をクリックしてコマンドを確定する

下の空欄に手順4で入力した、PLAY ACTION STANDが表示されます。

[追加]をクリックした後にコマンドを訂正する場合は、下の欄のコマンドを選んで入力しなおします。

「立つ」というアクションを設定しました。

6 続いて以下のコマンドを入力し、[追加]をクリックする

Main	Time#[ms]
------	-----------

WAIT 空欄

下の欄に、「WAIT」が追加されます。

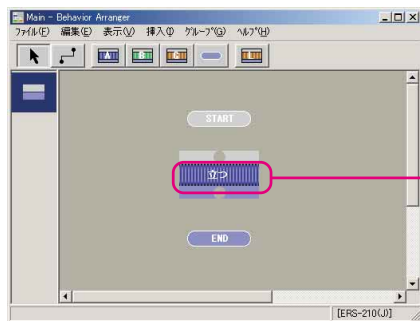
これは、手順4と5で設定したPLAY - ACTION - STAND(立つ)をAIBOが実行し終わるのを待つ(WAIT)というコマンドです。つまり、「AIBOが立つというアクションを終えるまで次のコマンドを実行させない」ことを意味しています。

AIBOは、複数のコマンドを同時に実行することがあります。その結果、途中でコマンドがキャンセルされたり、予期しない動きになってしまう場合があります。ひとつのコマンドを終了するまで次のコマンドを実行させたくない場合は、WAIT命令の時間(Time#[ms])を空欄にしておきます。

これは、Time#[ms]に「0」を設定するのと同じ意味を持ちます。プログラミングの世界では、「WAIT 0」は「待ち時間0」(=すぐに次を実行)ではなく、「コマンドが終了するまで、次のコマンドをスタートさせない」ことを意味します。

7 [OK]をクリックする

プロパティが閉じて、「立つ」アクションボックスが「START」と「END」の間に表示されます。



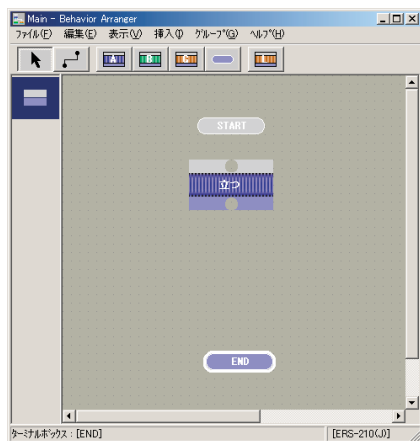
手順3で入力した
ボックス名

「立つ」アクションボックスが完成しました！

「歩いて、おじぎをする」アクションボックス作成

もうひとつアクションボックスを作成します。

1 「END」ターミナルボックスをクリックして選び、ドラッグして下に移動する



「立つ」と「END」の間にもうひとつアクションボックスを追加するために、「END」を下に移動します。

2 をクリックし、「立つ」と「END」の間でクリックする

アクションプロパティが表示されます。

3 アクションボックスの名前と内容を入力する

NAME : 歩くおじぎ

Comment : 10cm歩いてからおじぎをする

4 10cmまっすぐ歩くコマンドを入力し、[追加]をクリックする

Main	Sub	Name	水平角	距離
PLAY	ACTION	WALK	0	100

水平角0°方向(まっすぐ前方)に、距離100mm「歩く」というコマンドです。

5 WAITコマンドを入力し、[追加]をクリックする

Main	Time#[ms]
WAIT	空欄

「10cm歩き終わるまで次のコマンドを実行しない」ようにします。

6 (おじぎをするために)頭を(まっすぐ下に50°)下げるコマンドを入力し、[追加]をクリックする

Main	Sub	Name	水平角	垂直角
PLAY	ACTION	MOVE_HEAD	0	-50*

* ドロップダウンリストには表示されないので、直接入力します。

頭を水平角0°(まっすぐに)、垂直角(上下方向に)-50°へ動かす、つまり「頭を50度下げる」コマンドです。

7 手順5と同様にWAITコマンドを入力し、頭を下げるまで次のコマンドを実行しないようにする

8 (下げた)頭を元の位置に戻すコマンドを入力し、[追加]をクリックする

Main	Sub	Name	水平角	垂直角
PLAY	ACTION	MOVE_HEAD	0	0

頭を水平角0°(まっすぐに)、垂直角(上下方向)0°に動かす、つまり「頭を元に戻す」コマンドです。

- 9 手順5と同様にWAITコマンドを入力し、頭を上げるまで次のコマンドを実行しないようにする



システムアクションでおじぎをさせるには、頭の位置を下げ、元に戻すという動作を指定しますが、アクションコンポーザーでおじぎモーションを作って利用する方法もあります。

- 10 [OK]をクリックする


これで、「歩くおじぎ」アクションボックスが完成しました。

2つのアクションボックスが完成しました。

ボックスを接続しよう

すべてのプログラムをつないで、「立ち上がり、10cm歩いて、おじぎをする」という一連のアクションにしましょう。

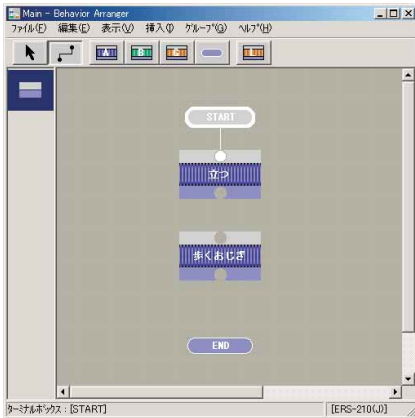
コマンドを入力した各ボックスをラインでつなぐと、一連のアクションになります。

- 1 [マウス接続モード]ボタン  をクリックする
マウスがボックスを接続するモードになります。

ここまで、マウスは「通常モード」でした。ボックスを接続するときには「接続モード」に、それ以外は「通常モード」にしましょう。



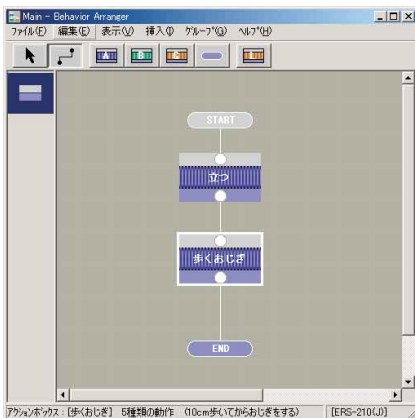
- 2 「START」をクリックし、「立つ」までドラッグする
ふたつのボックスがラインでつながります。



- 3 同様に「立つ」と「歩くおじぎ」をつなぐ

- 4 「歩くおじぎ」と「END」をつなぐ

これですべてのボックスが接続され、個々の動きが「連続したアクション」になりました。



Lesson3ではふたつのアクションボックスを使いましたが、ひとつのアクションボックスにコマンドをまとめても同じプログラムが作成できます。

プログラムを保存し、AIBOで実行しよう

作成したプログラムをハードディスクに保存した後、“メモリースティック”に保存します。

続いてAIBOでプログラムを実行しましょう。

プログラムをハードディスクに保存すると、他のビヘイビアを作成するときに再利用できたり、後で開いて編集できて便利です。

1 メニューから[ファイル]-[名前を付けて保存]を選ぶ

「Lesson3.be」というファイル名で保存してください。

2 “メモリースティック”を“メモリースティック”ドライブに挿入する

3 メニューの[ファイル]-[メモリースティックに保存]を選ぶ

書き込みが終了したら確認ダイアログが表示されますので、[OK]をクリックします。

プログラムを保存した“メモリースティック”をAIBOに挿入して起動すると、プログラムが実行されて、AIBOが立ち上がります。
もう一度プログラムを実行するには、AIBOを起動しなおします。

ワイヤレスLANを使わない方は、
ここでLesson3は終了です。

ワイヤレスLANを使ってみよう

ワイヤレスLANをお使いの場合は、PCからAIBOにプログラムを転送し、実行できます。

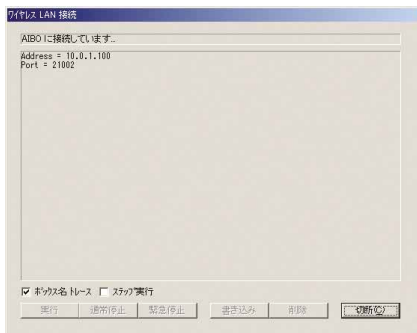
1 “メモリースティック”をAIBOに挿入して起動し、PCから通信できる範囲に置く

ワイヤレスLANを使うと、
Lesson3で作ったプログラムを再実行するときAIBOの電源を入れなおす必要がなくなり、何度でもプログラムを再実行することができます。

また、プログラムを作成・変更したときも、“メモリースティック”をAIBOに挿入したまま、PCからAIBO内の“メモリースティック”のプログラムを書き換えられるので便利です。

2 メニューの[ファイル]-[ワイヤレスLAN接続]を選ぶ

ワイヤレスLAN接続ダイアログが表示されます。



3 [書き込み]をクリックする

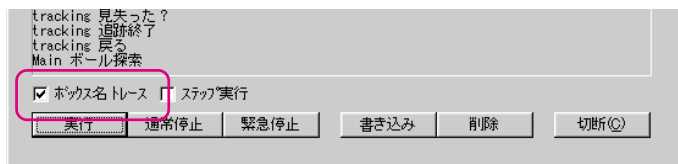
作成したプログラムが“メモリースティック”に保存されます。

すでに保存済みですが、ここでは勉強のため、ワイヤレスLAN経由で再度保存しましょう。

4 [実行]をクリックする

AIBOがプログラムに従って行動します。

「ボックス名トレース」にチェックを入れておくと、実行中のボックスが画面に表示されます。



AIBOが行動している途中で[切断]ボタンをクリックしても、プログラムはすでに転送されているので、AIBOはプログラムを最後まで実行します。

途中でAIBOの動きを止めるには、[通常停止]ボタンをクリックしてから、[切断]ボタンをクリックしてください。AIBOは安定した姿勢になってから動きを止めます。

すぐに動きを止めたいときは、[緊急停止]ボタンをクリックしてください。ただし、AIBOは不安定な姿勢のまま止まりますので、倒れたりすることがあります。

5 AIBOの動きが止まったら、[切断]をクリックする

編集ウィンドウに戻ります。

このレッスンで作成したプログラムは

「Sample ¥ 機種名」フォルダ内の「Tutorial3.be」ファイルと同じプログラムです。プログラムが正しくできたか確かめてみてください。

ここまで、うまくできましたか？

アクションボックスを使っのプログラミングは、意外と簡単だったでしょう？

これが基本のプログラミングです。

項目や数値を変えて、いろいろ試してみてください。

次は、中級コース。とはいっても、Lesson3の基本のプログラムを応用してみるだけですので、難しく考えずにLet's try !

12ページ参照

ビヘイビアアレンジャーで

「Tutorial3.be」を開き、各ボックスをダブルクリックするとプロパティが表示され、コマンドを確認できます。

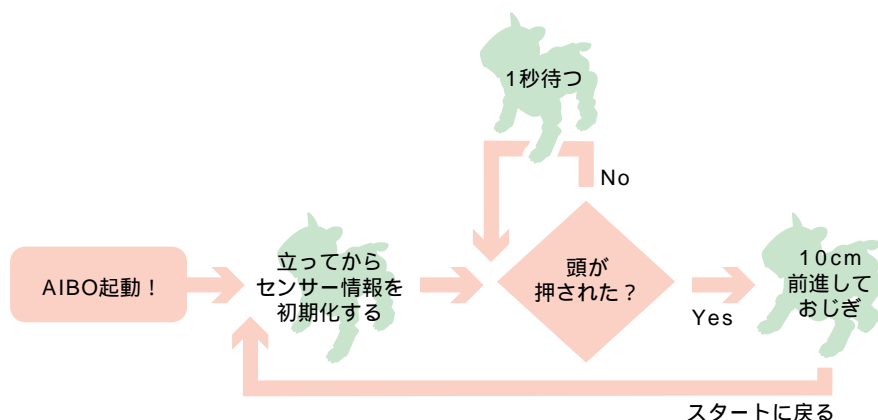
頭を押されたら おじぎ

目標：次の4点を習得する

- 既存のプログラムの編集方法
- 条件により分岐するプログラムの作成方法
- センサー情報の取得方法
- ループを使ったプログラムの作成方法

プログラム概要：

立った後、頭が押されるたびに前進しておじぎする



プログラムの作成・実行手順：

編集元のファイル表示 (34ページ)

ボックス作成 (35ページ)

1 センサー変数を初期化するコマンド追加 (既存ボックス編集)

2 分岐条件を含むランチボックス*作成

3 頭を押されたかを確認するボックス作成

ボックスを接続してプログラムをループ (38ページ)

プログラム保存・AIBOで実行 (41ページ)

* ランチボックスについては、
ユーザーガイド (32ページ) 参
照。

Lesson3と同様に、“メモリースティック”を用意し、ワイヤレスLANを設定しておきます。

Lesson3で作成したファイルを表示しよう

1 ビヘイビアアレンジャーを起動する

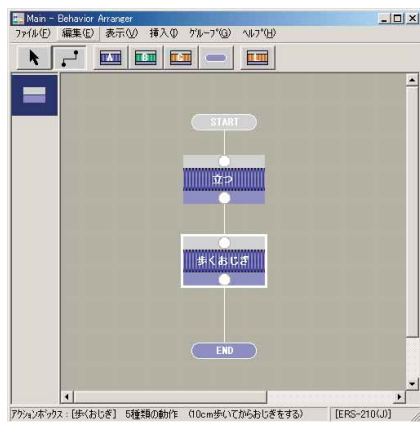
2 Lesson3で作成したプログラムを表示する

メニューの[ファイル]-[開く]で「Lesson3.be」を開きます。
Lesson3から続いてLesson4に進む方は、[マウス通常モード]ボタンをクリックしておきます。

Lesson4から始める方は、以下のファイルをコピーしてお使いください。

「Sample ¥ 機種名」フォルダ内の「Tutorial3.be」ファイル

(12ページ参照)



ボックスを作成しよう

3つのボックスを編集・作成します。

「立つ」ボックスにセンサー変数を初期化するコマンドを追加

「頭を押されたら...、押されないと...」という分岐条件を含むブランチボックスを作成

「1秒間待つ」(センサー情報取得)ボックスを作成

センサー変数を初期化するコマンドを追加する

「立つ」アクションボックスにセンサー変数を初期化するコマンドを追加します。

センサー変数は、一度「1」(センサー情報あり=タッチされた)がセットされると、その値を保持し続けます。そのため、ループするプログラムを作成するときや、以前に他のボックスでセンサーを使った場合などは、センサー変数を「0」に戻す必要があります。「0」に(元)に戻すことを「初期化」といいます。

1 「立つ」と「歩くおじぎ」の間のラインをクリックして選び、Deleteキーで削除する

「直ちに削除しますか?」というダイアログボックスが表示されますので[はい]をクリックしてください。

2 「立つ」アクションボックスをダブルクリックする

「立つ」アクションの次に「頭センサー」の変数を初期化するコマンドを追加します。

3 頭センサーを初期化するコマンドを追加し、プロパティを閉じる

Main	Variable	Var/Value
------	----------	-----------

SET Head_ON* 0**



Head_ONに「0」を代入するというコマンドを入力することで、頭センサーを初期化します。

* AIBOの頭部のセンサーは、ERS-210では頭センサー、ERS-220ではヘッドタッチセンサー、ERS-310シリーズでは頭スイッチ(頭内部のスイッチ)という名称です。いずれも頭センサーへの入力の変数名は「Head_ON」です。

** 頭センサーの変数値

1: 押された

0: 押されていない

4 「歩くおじぎ」と「END」を下に移動する

ブランチボックスで分岐を設定する

「頭を押されたら歩いておじぎをし、押されなかったら別のボックスにジャンプする」というボックスを作成します。

1 [ブランチボックス] ボタン をクリックし、「立つ」と「歩くおじぎ」の間に置く

2 ボックス名とコマンドを入力し、プロパティを閉じる

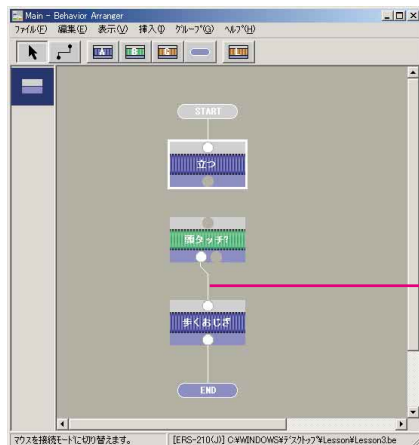
Name : 頭タッチ ?

頭を押されたら、歩くおじぎボックスへコマンドを入力する

Type	Variable	Op	Var/Value	JumpTo
IF	Head_ON	=	1	歩くおじぎ

続いて、頭が押されなかった場合のコマンドを入力する

Type	JumpTo
ELSE	null*



でジャンプ先を指定したので、自動的にラインでつながれる

ふたつのボックスを一度に移動するには、マウスでふたつのボックスを囲むようにドラッグして選択します。ドラッグして移動し、「立つ」と「歩くおじぎ」の間にスペースを設けます。

頭が押されたかどうかを判断して次の行動へ移るプログラムを作成します。

もし (IF) 頭を押された (Head_ON = 1) ら、「歩くおじぎ」ボックスへジャンプ


それ以外 (ELSE) なら、nullへジャンプ

* ジャンプ先のボックスはこれから作るので、ここでは「未定」を意味する「null」を選択しておきます。あとでジャンプ先のボックスを作ってラインでつなげると、「null」は自動的にジャンプ先のボックス名に書き変わります。

ブランチボックスには、出力端子が2つ表示されます。これは、分岐条件が2つあるためです。

「1秒間待つ」アクションボックスを作成する

頭が押されたかどうかを1秒間隔で確認するため、1秒間待つアクションボックスを作ります。

- 1  をクリックし、新規アクションボックスを「頭タッチ？」の横に置く

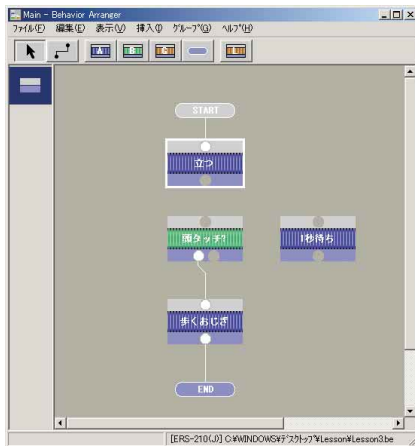
- 2 1秒待つコマンドを入力し、プロパティを閉じる

Name : 1秒待ち

Main	Time#[ms]
------	-----------

WAIT 1000

1000ms (1秒) 待つ



ボックスをつないでプログラムをループさせよう

ボックスをつないでプログラムの流れをつくります。

1秒ごとに頭センサーの入力を確認する

- 1 マウスを接続モードにしてから、「立つ」「頭タッチ？」ボックスを接続する

「マウス通常モード」でシフトキーを押しながらマウスをクリックしても、接続することができます。

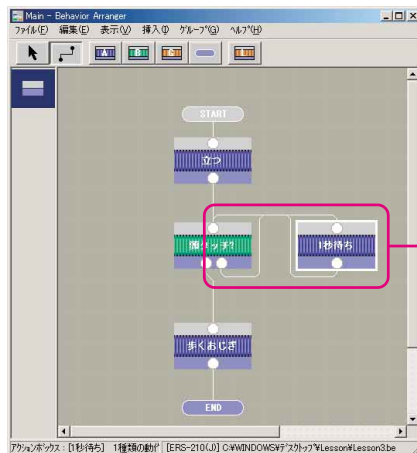
- 2 ループするようにラインを接続する

「頭タッチ？」の空いている接続端子 「1秒待ち」

「1秒待ち」の下接続端子 「頭タッチ？」の上の接続端子

1秒待つアクションボックスがループされ、AIBOは頭センサーへの入力が無かったら、1秒後に再度センサーの入力を調べるようになります。これで、頭センサーを押すとAIBOが歩いておじぎをするようになります。

ブランチボックスとジャンプ先のボックスを接続するときは、接続端子の条件を確認してから、正しいジャンプ先に接続してください。接続端子にマウスを置くと、分岐条件が表示されます。「頭タッチ？」の空いている接続端子(ELSE)を「1秒待ち」につなげることで、「頭を押されなかったら」「1秒待ちする(1秒待ちボックスへジャンプ)」という分岐が設定されました。



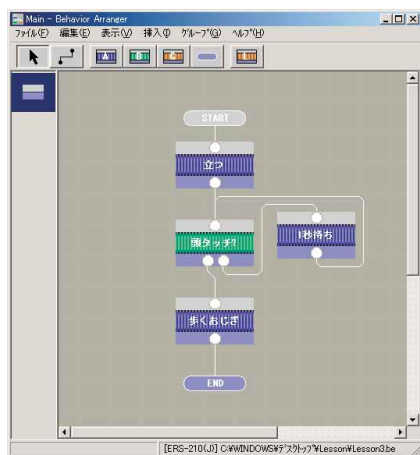
1秒間隔で頭センサーへの入力を確認する

ブランチボックスで「ELSE - JumpTo: - null」と入力しましたが、空いていた出力端子を「1秒待ち」につないでジャンプ先を設定したので、「頭タッチ?」のプロパティを開くと「null」が「1秒待ち」に書き換えられます。

Type	Variable	Op	Value	JumpTo
IF	Head_ON	=	1	先くおにぎ
ELSE				1秒待ち

3 交差しているラインをドラッグして整理する

ラインが交差して見にくい場合は、マウスを「通常モード」にしてから、ラインをドラッグして移動します。



ボックスの数が増えて画面に入らなくなった場合は、[表示] [編集領域のサイズ変更] で編集ウィンドウの幅と高さを変えてください。全体のボックス配置を確認したい場合は、[表示] [縮小表示 (50%)] や [縮小表示 (25%)] で全体を確認できます。

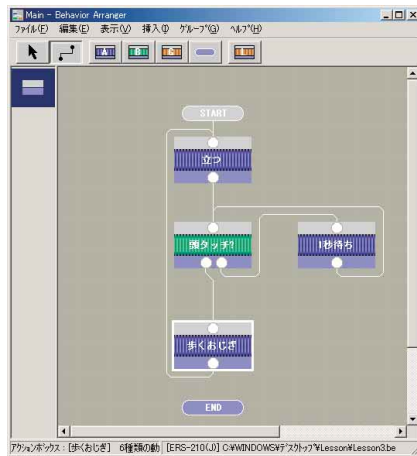
プログラム全体をループさせよう

AIBOが歩いておじぎをしたら、最初に戻って、頭センサーへの入力があるとおじぎをするように、ループさせます。

1 「歩くおじぎ」と「END」の接続を切断する

35ページ参照

2 「歩くおじぎ」と「立つ」をつなぐ



プログラム全体をループさせて終了しないようにするので、「END」ボックスはどこからも接続せずに、そのままにしておきます。

おじぎが終わったあと「立つ」ボックスに戻ることでセンサー値が初期化され、再度頭センサーへの入力を待つ状態になります。

プログラムを保存し、AIBOで実行しよう

作成したプログラムを「Lesson4.be」というファイル名で保存してから「メモリースティック」に書き込み、AIBOで実行してみましよう。

プログラムが実行されると、AIBOは立ち上がった後、頭が押されるのを待ちます。頭を押すと、10cm前進して、おじぎをします。その後、立ち姿勢のまま再び頭が押されるのを待ちます。

プログラムを終了するには、AIBOが倒れないように手で支えてからAIBOのポーズボタンを押して電源を切ってください。

ワイヤレスLANをお使いの場合は、[通常停止] ボタンでAIBOの動きを止めてから、[切断] ボタンで接続を切断します。

30ページ参照

すぐに動きを止めたいときは、31ページ参照。

このレッスンで作成したプログラムは

「Sample ¥ 機種名」フォルダ内の「Tutorial4.be」ファイルと同じプログラムです。プログラムが正しくできたか確かめてください。

12ページ参照

さあ、分岐条件を含む、すこし複雑なプログラムが完成しました。ランチボックスの項目を変えて、いろいろな条件を設定してみてください。

AIBOにアクションをさせるのが、どんどん楽しくなっていきます!

「あいぼ」と呼ぶとおじぎ、「こんにちは」であいさつ

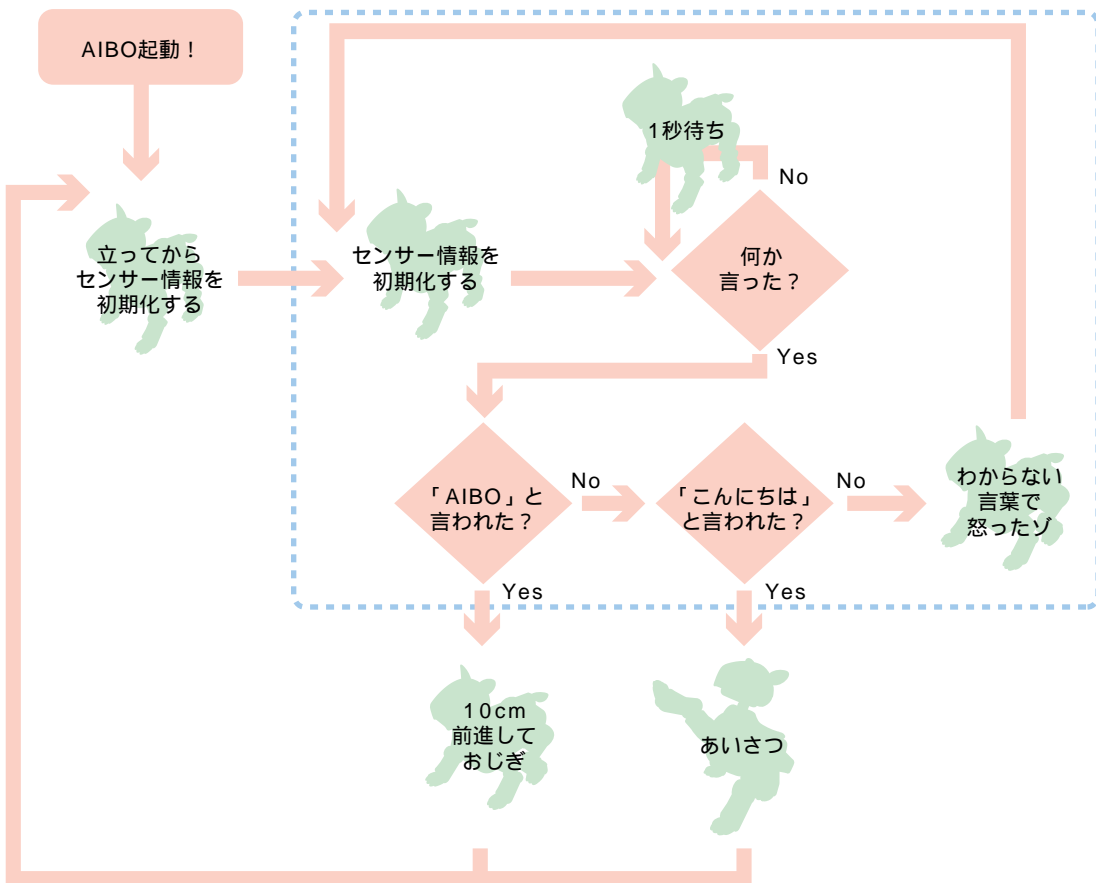
目標：次の3点を習得する

- グループの作成方法の習得
- 音声認識を使ったプログラムの作成方法
- アクションコンポーザーのアクションをプログラムに組み込む方法の習得

プログラム概要：

立った後、言われた言葉によって違うアクションをする

ひとつのグループ



プログラムの作成・実行手順

アクションコンポーザーのアクションライブラリ準備(43ページ)

グループ作成(45ページ)

- 既存のボックスをグループ化
- アクションボックス・プランチボックス編集
- アクションファイル追加

グループをメインプログラムに組み込む(53ページ)

プログラム保存・AIBOで実行(55ページ)

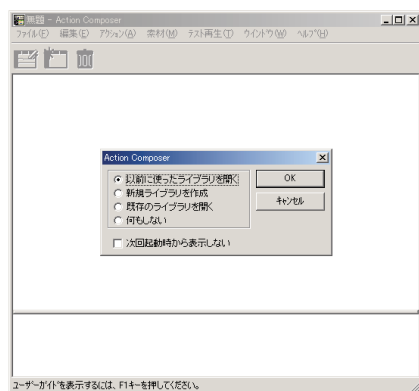
アクションライブラリを準備しよう

アクションコンポーザーのアクションライブラリをビヘイピアアレンジャーで使うには、あらかじめアクションライブラリを“メモリースティック”に保存しておく必要があります。このレッスンでは既存のアクションライブラリを利用します。

アクションコンポーザーで新規に作成した素材やアクションファイルを使う場合は、まず素材とアクションファイルをアクションライブラリに登録してから、“メモリースティック”に保存します。詳しくは、ユーザーガイド(38ページ)をご覧ください。

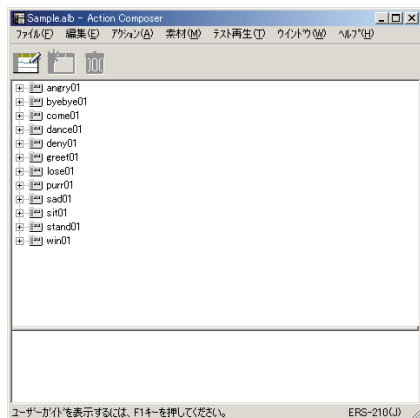
1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[アクションコンポーザー]を選ぶ

アクションコンポーザーが起動します。



2 「既存のライブラリを開く」を選び、[OK]をクリックする

3 ビヘイビアと同じ機種名のフォルダから、「Sample.alb」を開く



アクションライブラリは、「Sample ¥ 機種名」フォルダ内にあります(12ページ参照)。

ビヘイビアとアクションライブラリの機種名・言語は一致している必要があります。

アクションライブラリは、“メモリースティック”に保存すると同時に、ハードディスクの決められたディレクトリにもコピーされ、ビヘイビアアレンジャーで参照できるようになります。

ビヘイビアアレンジャーが参照できるのは、1ライブラリだけで、“メモリースティック”にも1ライブラリだけ保存できます。“メモリースティック”にすでにアクションライブラリが保存されている場合は、上書き保存されます。

ご注意

ビヘイビアアレンジャーでは、最後に保存したアクションライブラリを参照します。複数の“メモリースティック”を使っている場合は、使用する“メモリースティック”とハードディスク内のアクションライブラリが異ならないよう、ご注意ください。

Lesson5から学習を始める方は、以下のファイルをコピーしてお使いください。

「Sample ¥ 機種名」フォルダ内の「Tutorial4.be」ファイル(12ページ参照)

4 “メモリースティック”をドライブに挿入し、メニューの[ファイル]-[メモリースティックにライブラリを保存]を選ぶ

アクションライブラリが“メモリースティック”に書き込まれます。

5 アクションコンポーザーを閉じる

6 ビヘイビアアレンジャーを起動し、Lesson4で作成したビヘイビアファイル(Lesson4.be)を開く

ボックスをまとめて、グループを作成しよう

グループボックス*の使い方を学習します。「頭タッチ?」と「1秒待ち」アクションボックスをグループ化してまとめます。

* グループボックスについては、ユーザーガイド(43ページ参照)。

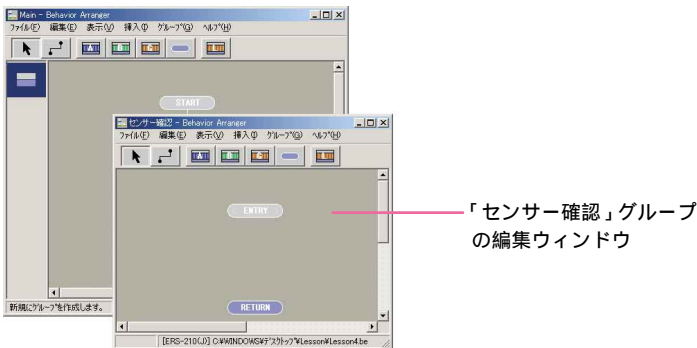
「頭タッチ?」と「1秒待ち」をグループに移動する

複数のボックスをグループにまとめておくと、プログラム内でグループ単位の再利用ができて便利です。

1 メニューの[グループ]-[新規グループの作成]を選ぶ

2 「Name」に「センサー確認」と入力し、[OK]をクリックする

「センサー確認」というグループが作成され、「センサー確認」グループの編集ウィンドウが表示されます。



3 「RETURN」をドラッグして下へ移動する

スペースを設けます。

4 メインウィンドウで、Ctrlキーを押しながら「頭タッチ?」ボックス、「1秒待ち」ボックスと、ふたつのボックスを結び2本のラインをクリックして、選択状態にする

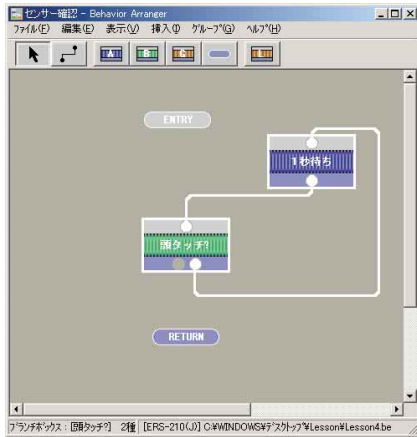
5 メニューの[編集]-[切り取り]を選ぶ

選択したボックスとラインが切り取られます。

6 「センサー確認」グループのウィンドウで、[編集]-[貼り付け]を選ぶ

選択したボックスとラインが、「センサー確認」グループに移動します。

選択したボックスとラインをマウスでドラッグして移動することはできません。



ブランチボックス・アクションボックスを編集する

移動したブランチボックスとアクションボックスを、以下のように編集しましょう。

「頭タッチ？」ブランチボックスのコマンドを変更

音声認識を初期化するコマンドを追加

音声認識によってふたつのボックスへ分岐するブランチボックスを作成

「頭タッチ？」ブランチボックスを書き換える

「頭タッチ？」ブランチボックスを、音声を認識したらループから抜けるボックスに変更します。

日本語版のシステムを使うことで、AIBOは登録されている53語の日本語を認識できるようになります。単語に対する反応を設定することで、音声認識を利用したプログラムが作れます。単語一覧はユーザーガイド(59ページ)をご覧ください。

1 「頭タッチ？」を開く

2 プロパティで、ボックス名とコマンドを変更する

ボックス名を「音声認識?」に変更する

リストに表示されている「IF Head_ON = 1」をクリックして選び、コマンドを変更する

Variable	Op	Var/Value	JumpTo
AU_Voice*	=	1	null

[変更]をクリックする

「IF Head_ON = 1 歩くおじぎ」が「IF AU_Voice = 1」に書き換わります。

*「AU_Voice」は音声認識の変数で、ビヘイビアで使用できる53の単語の入力を表します。

「言葉(53語のどれか)を認識したら 未定ボックスへジャンプ」に変わります(nullについては36ページ参照)。

「即時更新」ボックスをチェックしている場合は、手順 は不要です。

3 ブランチプロパティダイアログを閉じる

音声認識を検出する変数を初期化する

「音声認識?」ブランチボックスで音声を確認したかどうかを調べる前に、システム変数を初期化するボックスを作成します。

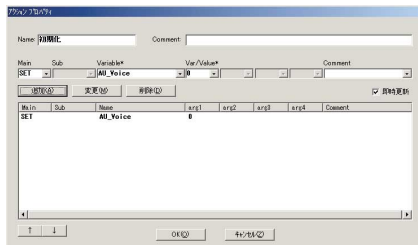
一度変数を「1」と認識すると、「音声認識 = 1 (音声認識された)」という情報をAIBOが持ち続けてしまうからです。

1 「音声認識?」の上に新規アクションボックスを置く

2 初期化するコマンドを入力し、プロパティを閉じる

Name : 初期化

Main	Variable	Var/Value
SET	AU_Voice	0



初期化については35ページ参照。

音声認識を初期化するボックスが完成しました。

認識した音声によって、2つのターミナルボックスへ分岐させる

「あいぼ」と音声認識したら進む「AIBO」ターミナルボックスと、「こんにちは」と音声認識したら進む「Hello」ターミナルボックスを作成します。

さらに、AIBOが認識できる他の言葉だった場合は、グループボックスの最初に戻ってループさせます。

以下のようなプログラムになります。

1秒ごとに音声認識をしているかをチェック

- 「あいぼ」の音声进行認識
「AIBO」ボックスへ
- 「こんにちは」の音声进行認識
「Hello」ボックスへ
- それ以外のAIBOが認識できる
言葉
ループ(初期化ボックスへ)

1 「RETURN」をダブルクリックする

ターミナルプロパティが表示されます。

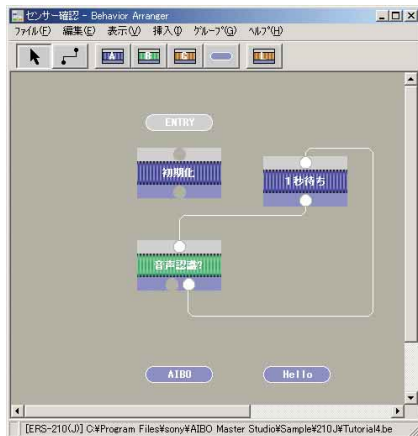
2 「Name」を「AIBO」に変更し、プロパティを閉じる

「AIBO」ターミナルボックスになります。

3 「ターミナルボックス」ボタン をクリックし、 「AIBO」の横に置く

4 「Name」に「Hello」と入力し、プロパティを閉じる

「Hello」ターミナルボックスができます。

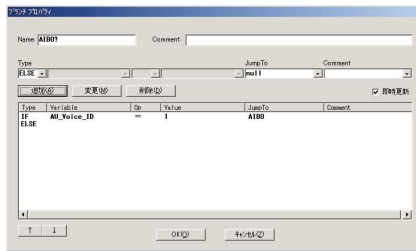


5 「音声認識？」の下に新規ブランチボックスを置く

6 ボックス名とコマンドを入力し、プロパティを閉じる

Name : AIBO?

Type	Variable	Op	Var/Value	JumpTo
IF	AU_Voice_ID	=	1	AIBO
ELSE				null



このボックスで「あいぼ」と認識したら「AIBO」ターミナルボックスへジャンプするコマンドを作成します。

「あいぼ」と認識 「AIBO」ターミナルボックスへジャンプ

「AU_Voice_ID=1」は「音声認識単語ID=1(あいぼ)」「あいぼという言葉を知った」を表します。

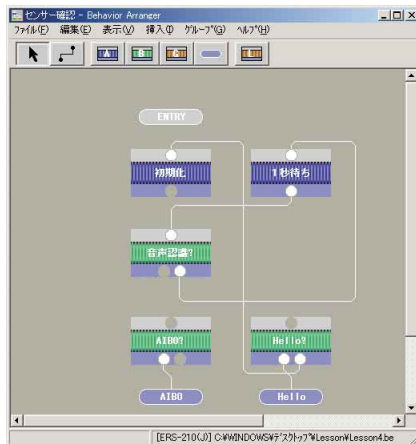
それ以外なら 未定ボックスへジャンプ

7 「AIBO?」の横に新規ブランチボックスを置く

8 ボックス名とコマンドを入力し、プロパティを閉じる

Name : Hello?

Type	Variable	Op	Var/Value	JumpTo
IF	AU_Voice_ID	=	6	Hello
ELSE				初期化



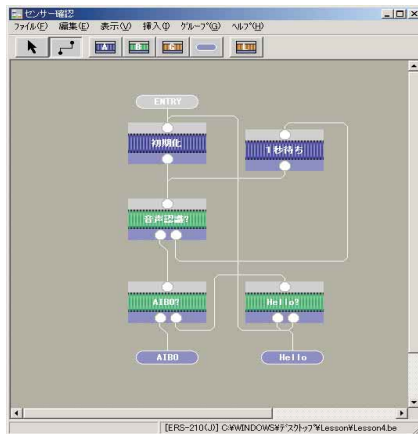
このボックスで「こんにちは」と認識したら「Hello」ターミナルボックスへジャンプするプログラムを作成します。

「AU_Voice_ID=6」は「音声認識単語ID=6(こんにちは)」(こんにちはという音声を認識した)を表します。

「あいぼ」「こんにちは」以外の単語を知ったとき、プログラムの先頭にループさせるように、「それ以外 初期化」ボックスにジャンプするように設定します。

9 ボックスをつなぐ

「ENTRY」「初期化」「音声認識?」の空き端子「AIBO?」「AIBO?」の空き出力端子「Hello?」とつながめます。



これで「あいぼ」と「こんにちは」の言葉を認識したら、それぞれ「AIBO」と「Hello」に進むグループボックスが作成できました。

「AIBO?」と「Hello?」を接続するときは、「AIBO?」の下の空いている端子と「Hello?」の上の空いている端子を接続します。ボックスを選択して接続しようとする、ブランチ設定ウィンドウが表示されますので、[キャンセル]をクリックしてください。

ボックスではなく、端子同士を接続するよう、マウスを「AIBO?」の下の空いている端子部分でクリックし、そのままドラッグして、「Hello?」と接続してください。

認識した言葉がどちらでもない場合は、再び「AU_Voice」を初期化して、新しい音声認識されるまで1秒間隔で音声認識をしたかをチェックし続けます。

アクションファイルをプログラムに盛り込む

保存したアクションライブラリ(43ページ)を使って、アクションファイルをプログラムに盛り込み、AIBOが表現豊かに行動するようにしましょう。

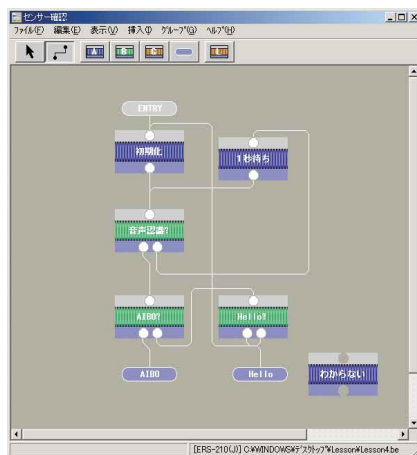
ここでは、AIBOが「あいぼ」「こんにちは」以外の言葉を認識したことを、怒ったポーズで表現させます。

1 「Hello?」の右下に新規アクションボックスを置く

2 ボックス名とコマンドを入力し、プロパティを閉じる

Name : わからない

Main	Sub	Name
PLAY	ACTION+ *	angry01
WAIT	空欄 **	



*「ACTION+」はアクションライブラリのアクションファイルを表します。PLAY-ACTION+でアクションライブラリのアクションが使えます。

ここでは、「angry01」というアクションファイルを組み込みます。

** アクションを再生し終わるまで待たせます(25ページ参照)。

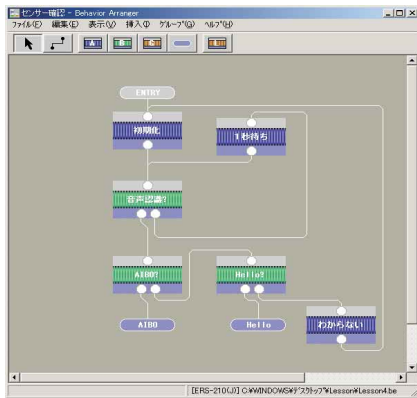
3 ボックスをつなぎ直し、ラインを整理する

「Hello?」と「初期化」の接続を切断します。

続いて「Hello?」の右下の端子 「わからない」「わからない」「初期化」とつなぎます。

ELSEの場合 わからないボックスへジャンプ

わからない場合 angry01を実行
初期化ボックスへ戻る



4 グループウィンドウを閉じる


「センサー確認」グループが完成しました。

ウィンドウを閉じて、グループは消えません。

編集ウィンドウには、作ったグループは表示されません。次の項でグループボックスを使って、作成した「センサー確認」グループをプログラムに組み込みます。

グループを組み込んでプログラムを完成させよう

最初にプログラムを作っていた編集ウィンドウに戻り、プログラムを完成させます。作成したグループプログラムを組み込みます。

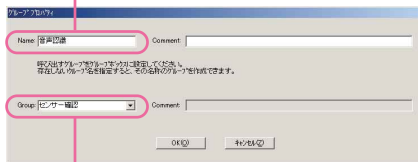
- 1 [グループボックス]ボタン  をクリックし、「立つ」と「歩くおじぎ」の間に置く

グループプロパティが表示されます。

- 2 グループボックス名を入力し、ボックスが呼び出すグループ名を選んでプロパティを閉じる

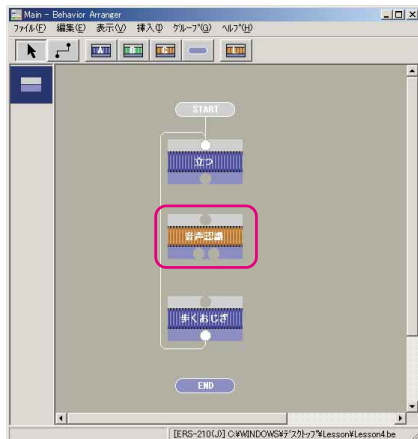
「Name」に「音声認識」と入力し、「Group」で「センサー確認」(作成したグループ名)をリストから選択します。

グループボックス名



呼び出すグループ名

グループボックスが置かれます。



最初にプログラムを作っていたのもひとつのグループで、Mainグループといいます。Mainグループに作ったグループを組み込んで、以下のアクションを完成させます。

「あいぼ」 おじぎ

「こんにちは」 右手を上げて挨拶
どちらでもない わからないから
怒ったぞポーズ

1秒間隔で言葉チェック

「グループボックス名」と「グループ名」は異なります。ボックス名はボックスに表示される名前、グループ名はそのボックスが呼び出すグループの名前を表します。

このグループボックスは、ふたつのターミナルを持つグループを呼び出すので、出力端子がふたつあります。

ひとつは、「AIBO」ターミナルボックスからの出力、もうひとつは「Hello」ターミナルボックスからの出力端子です。

マウスを端子部分に置くと、「AIBO」「Hello」とターミナル名が表示されることを確認してください。

3 「音声認識」の「AIBO」端子と「歩くおじぎ」ボックスをつなぐ

これで、「あいぼ」という言葉を認識すると「歩いておじぎをする」というプログラムになりました。

4 新規アクションボックスを「歩くおじぎ」の右側に置く

「こんにちは」という音声を実行したときに実行する「あいさつ」のアクションコマンドを入力します。「あいさつ」には、アクションライブラリ(43ページ参照)内の「greet01」アクションファイルを使います。

5 ボックス名とコマンドを入力し、プロパティを閉じる

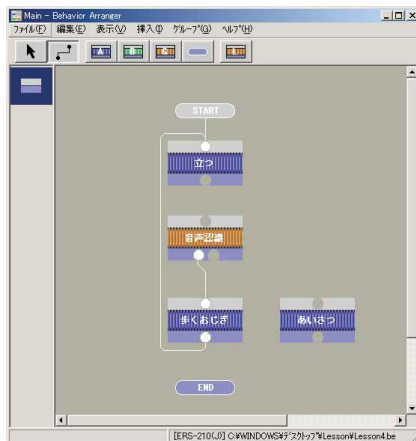
Name : あいさつ

1行目にあいさつするコマンドを入力する

Main	Sub	Name
PLAY	ACTION+	greet01

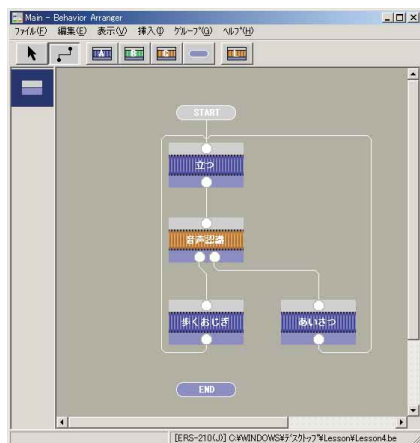
2行目にWAIT 0コマンドを入力する

25ページ参照



6 ボックスをつなぎ、ラインを整理する

「立つ」「音声認識」「音声認識」の「Hello」端子「あいさつ」
「あいさつ」「立つ」とつなげます。



「こんにちは」という言葉を認識するとあいさつし、あいさつすると最初に戻って立ち、言葉を待つようになります。

プログラムが見にくかったらラインやボックスを移動させましょう。

プログラムを保存し、AIBOで実行しよう

作成したプログラムを「Lesson5.be」というファイル名で保存してから“メモリースティック”に書き込み、AIBOで実行してみましょう。

30ページ参照

プログラムが実行されると、以下のアクションを繰り返します。

立つ	「あいぼ」を認識	10cm歩いておじぎ
	「こんにちは」を認識	右手をあげてあいさつ
	上記以外の言葉を認識	怒ったぞ!ポーズ

このレッスンで作成したプログラムは

「Sample¥機種名フォルダ」内の「Tutorial5.be」ファイルと同じプログラムです。プログラムが正しくできたか確かめてみてください。

12ページ参照

複雑なプログラムが完成しました!

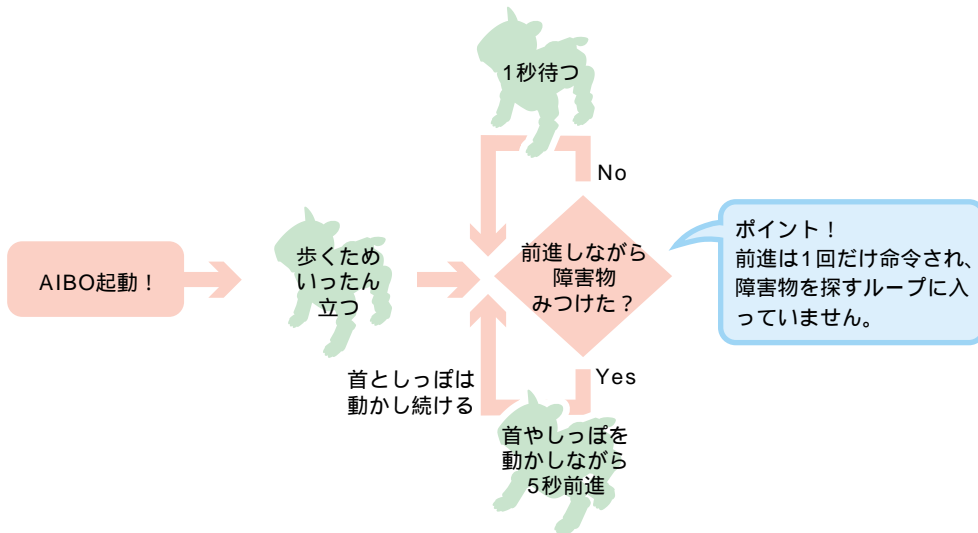
手順が多くて大変でしたが、いろいろなテクニックを習得できていますので、もうビヘイビアアレンジャーを使いこなせるはずです。組み込むアクションを変えてみたり、歩く距離を変えてみたりして、いろいろ応用してください!

同時に複数のアクションを実行する！

目標：同時に複数のアクションを実行するプログラムを作る

プログラム概要：

障害物を見つけたら、首やしっぽを動かしながら5秒間後進し、
また前進する



プログラム作成・実行手順

現在のプログラムを確認 (57ページ)

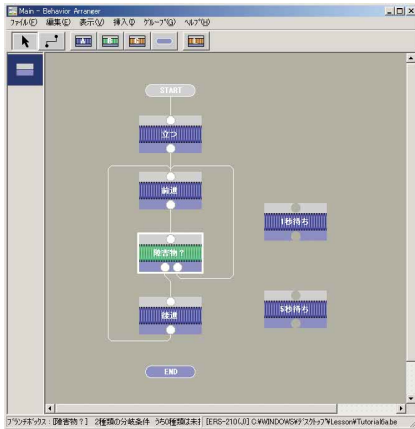
複数のアクションを同時に実行するプログラムを作成 (58ページ)

- ・ 前進プログラム 前進しながら障害物を探索プログラムに変更
 - ・ プログラムをループ
 - ・ しっぽを回し (ERS-210のみ) 首を横に向けるアクション追加
- プログラム保存・実行 (61ページ)

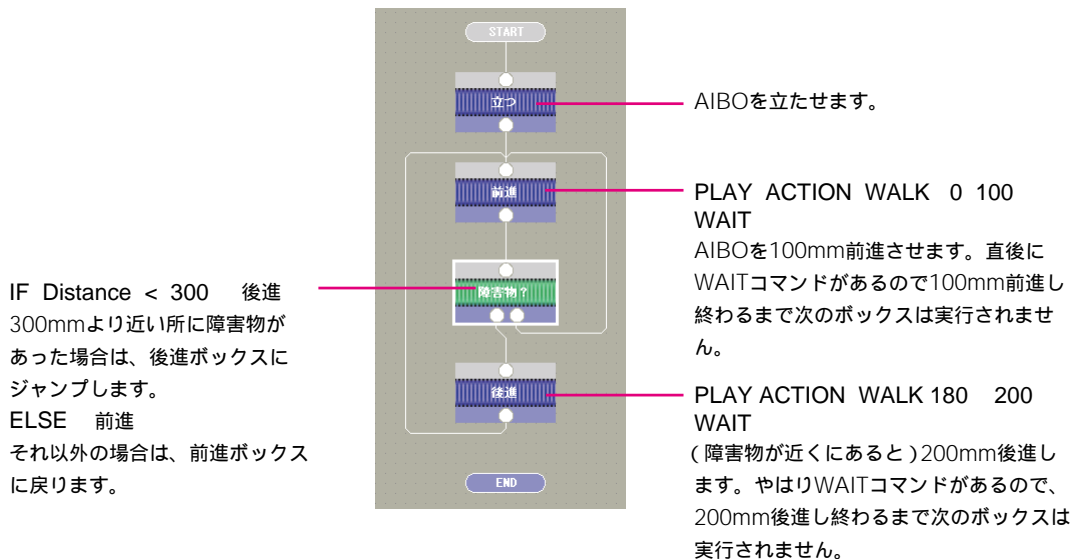
現在のプログラムを確認しよう

「Tutorial6a.be」ファイルを編集します。
ファイルをコピーし、ビヘイビアレンジャーで開いておきます。

「Tutorial6a.be」ファイルは
「Sample ¥機種名」フォルダ内に
あります(12ページ参照)。



このプログラムは、ボックスの中のコマンドを(複数を同時にではなく)「ひとつずつ」実行していきます。



このように、ひとつの命令が終了するまで、次のボックスを実行しないようプログラムされています。

複数のアクションを同時に実行させよう

「歩きながら障害物を検出する」という、複数のアクションを同時に実行するプログラムに書き換えましょう。

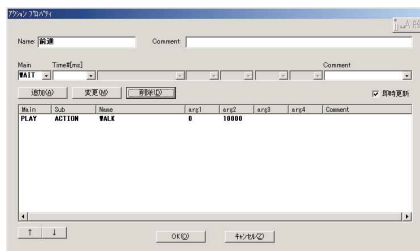
1 「前進」ボックスのプロパティを開く

2 歩く距離を変更する

PLAY – ACTION – WALK – 0 – 100 コマンドの距離「100」を「10000」に変更します。

10000mmという長い距離を設定することで、歩き続けることができます。

3 WAITコマンドを削除し、プロパティを閉じる



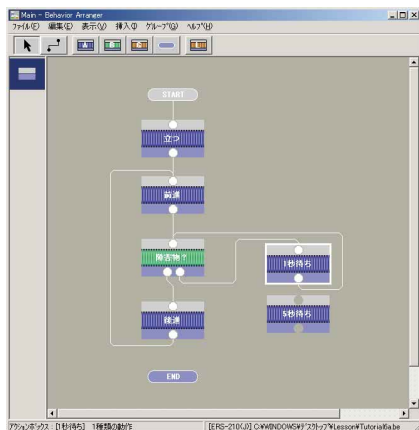
続いて、障害物がなかったら、（再び前進ボックスに戻らずに）障害物が見つかったかどうか1秒おきに確認し続けるよう、ループさせます。

WAIT命令が削除されたので、前進し続ける命令を出した後すぐに「障害物？」ボックスが実行されます。これでAIBOは前進しながら障害物があるかを調べるようになります。

4 「障害物？」のプロパティで、「ELSE 前進」を「ELSE 1秒待ち」に変更し、プロパティを閉じる

ジャンプ先を「前進」「1秒待ち」ボックスに変更

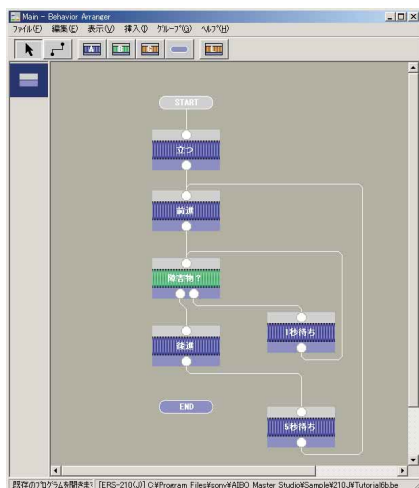
5 「1秒待ち」の出力端子と「障害物?」の入力端子をつなぎ、ラインを整理する



最後に、「後進」した後に次の命令を実行する条件を、「距離」ではなく「時間」に変更しましょう。

6 「後進」ボックスのプロパティで、後進するの移動距離「200」を「10000」に変更し、WAITコマンドを削除してプロパティを閉じる

7 「後進」と「前進」の接続を切断し、「後進」と「5秒待ち」と「前進」を順につなぐ



これで前進命令を実行しながら1秒間隔で障害物(距離センサー)を調べることができます。

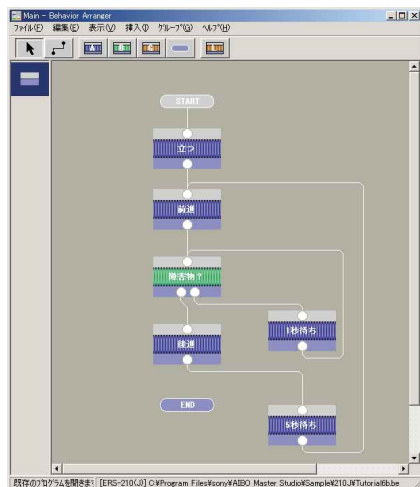
「障害物?」と「1秒待ち」をループで実行中に「前進」が再度命令されていないことがポイントです。

WAIT命令が削除されたので、後進し続ける命令を出した後すぐに「5秒待ち」ボックスが実行されます。これでAIBOは5秒間後進ようになります。その後前進します。

さらに、しっぽを回す(ERS-210のみ)アクションと首を横に向けるアクションを追加しましょう。

8 「後進」ボックスのプロパティでコマンドを追加し、プロパティを閉じる

```
PLAY ACTION PALONE.AUTO.TAILROT (ERS-210のみ)  
PLAY ACTION MOVE_HEAD 50 0
```



「しっぽを回す」「頭を50度かたむける」コマンドが追加されました。

これで、プログラムは完成です。

プログラムを保存し、AIBOで実行しよう

プログラムを「Lesson6.be」というファイル名で保存し、続いて“メモリースティック”に保存して、AIBOで実行してみましょう。

コマンドを書き換えた部分のアクションに注目！

「5秒待ち」ボックスを実行しているとき、4つのアクションが同時に実行されています。

- 5秒待ち
- 後進
- しっぽ回し(ERS-210のみ)
- 首を左に50度かたむける

その後「前進」ボックスを実行しますが、(しっぽ回しと) 首の向きを変える命令は前進命令後も有効となり、そのまま実行され続けます。

このように、今までのレッスンとは逆に、引き数なしのWAITコマンド(WAIT 0)を使わないことによって、複数のアクションを同時に実行することができます。場面、場面でWAITコマンドをうまく使ってください。

このレッスンで作成したプログラムは

「Sample¥機種名」フォルダ内の「Tutorial6b.be」ファイルと同じプログラムです。プログラムが正しくできたか確かめましょう。

12ページ参照

HINT

- システムアクションのうち、歩行と首の動き(WALK, MOVE...)、探索(SEARCH...)、耳としっぽの動き(PALONE...)、LED表示(SMESS...)は引き数なしのWAITコマンドがないと、次の命令をすぐに実行します(コマンドの上書き実行)。

次の命令が実行中の命令を中断する必要がある場合は、前の命令を中断して新しい命令を実行します(前進から後進に、後進から前進に切り替える場合など)。

実行中の命令を中断しなくてもよい場合は、継続して今の命令を実行します(前進と後進が切り替わっても、(しっぽ回しと)首を横へ向ける命令は有効になっている)。

また、上記以外のシステムアクションとアクションコンポーザーで作成したユーザーアクションは引き数なしのWAITコマンドがないと、次の命令をキューに溜め、実行中の命令が終わってから、キューに溜まった命令を実行します。ただし、すぐに実行できない命令が続くと、次々と命令がキューに溜められ、キューがあふれてプログラムが正常に動作しなくなります。

ユーザーガイド(55ページ)参照

- 障害物を検出したらずぐに前進から後進に動きを変えるには、「1秒待ち」ボックスを削除して、「障害物？」ボックスのELSE側端子を「障害物？」の入力端子に接続します(自ボックス内でのループ)。これで障害物検出を(1秒間隔ではなく)可能な限り最短時間で行います。
- 「1秒待ち」ボックスのWAITコマンドに0 msを指定した場合、引き数なしのWAITコマンドと同じになり、前進ボックスの

PLAY ACTION WALK 0 10000

の終了を待ちます。つまり、AIBOが10m前進を終えるまで障害物を検出しないので注意してください。

- 10m 前進する間に障害物を発見しないと、前進ボックスの

PLAY ACTION WALK 0 10000

は終了するので、それ以上AIBOは前進しません。

10m前進したあと、止まっているAIBOに障害物を見せると、「障害物？」と「1秒待ち」のループから抜けて、「後進」ボックスを実行しますので、5秒間後進したあと、前進します。

高度なワザをマスターしよう！

ビヘイピアレンジャーでプログラムを作成するときの、高度なテクニックを紹介します。「ピンクボールを追跡してキックする」プログラムの内容を確認ながら、プログラミング手法を学んでいきましょう。ここでは、プログラムの編集は行いません。

ファイルを準備する

「Soccer.be」ファイルをビヘイピアレンジャーで開いておきましょう。

「Soccer.be」ファイルは、「Sample ¥機種名」フォルダ内にあります(12ページ参照)。

メインプログラムを確認しよう

立ち上がる

立ち姿勢のコマンドを出し、WAITコマンドで立ち姿勢になるまで待たせています。

ピンクボールを探索する

PLAY ACTION MOVE_HEAD 0 0

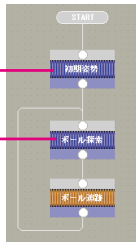
WAIT

上記コマンドで正面を向かせたあと、下記コマンドでボールを探します。

PLAY ACTION SEARCH.HEAD.LOWCENT PINK_BALL

WAIT

SEARCH コマンドはピンクのボールを探し続けますが、SEARCHコマンドは上書き実行が可能のため、次のコマンドがあると探すのをやめてしまう可能性があります。WAIT コマンドを使って次のコマンドを実行しないようにすることで、AIBO はボールを探しつづけます。



「ボール追跡」グループボックスのプログラムを確認しよう

ボールの追跡を始める

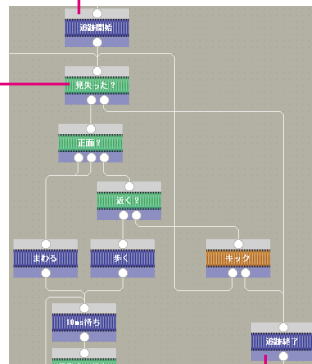
PLAY ACTION TRACK_HEAD PINK_BALL
が入力されていますが、WAIT コマンドはありません。ボールを追跡しながら次の命令が実行されることに注目しましょう。

見失ったかどうか、判断する

IF Pink_Ball = 1 正面？

ELSE 追跡終了

で、ボールが見えているかどうかを確認しています。ボールが見えていると AIBO は、Pink_Ballシステム変数で「1」を返し、「正面？」ボックスへジャンプします。ピンクボールが見失うと、「追跡終了」へジャンプします。



追跡を終了する

先にボールを見失ったときの処理を確認しましょう。

「追跡終了」ボックスでは、WAITコマンドのみが入力されています。引き数なしのWAITコマンドは実行中のすべての命令が終了するまで待ちます。

ここでは、「追跡開始」ボックスの

PLAY ACTION TRACK_HEAD PINK_BALL

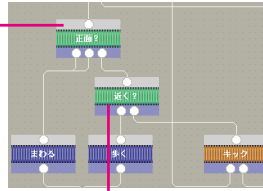
の終了を待ちます。Pink_Ballシステム変数が「1」ではないため、ボールを見失っている状態ですが、すぐに TRACK_HEAD命令が終了しません。TRACK_HEADを終了するまで時間差があるため、このWAITコマンドで TRACK_HEADの終了を待っているのです。

ピンクボールの位置によって、体の向きを変える

ボールが見えているときのジャンプ先の「正面?」ボックスでは

```
IF Head_Pan > 35
AND Wait = 1 まわる
IF Head_Pan < -35
AND Wait = 1 まわる
ELSE 近く?
```

となっています。Head_Panシステム変数はAIBOの横方向の首の角度で、この角度が35より大きいか - 35より小さく、かつWait変数が「1」の場合、AIBOの向きを変える「まわる」ボックスにジャンプします。Waitシステム変数は、実行中の命令の数を表し、ボールを追跡中はPLAY ACTION TRACK_HEAD PINK BALLを実行中なので、「1」になります。それ以外の場合は、ボールは正面にありますので、ボールまでの距離が近いか遠いかを調べるため「近く?」ブランチボックスにジャンプします。



ボールまでの距離を調べる

```
IF Head_Tile > -60
AND Wait = 1 歩く
ELSE キック
```

AIBOの首の角度を使ってボールが近いか遠いかを判定しています。「- 60°」より大きい場合は、首が上を向いていますので、ボールが遠いことになります。

Waitシステム変数が「1」でボールの追跡が続いている場合は「歩く」ボックスにジャンプします。

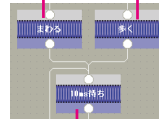
首の角度が「- 60°」より小さい場合は、AIBOは下を向いているので、ボールが足元にあることになります。このときは「キック」グループボックスにジャンプします。

頭を向けている方角へ旋回する/歩く

「まわる」ボックスでは、
PLAY ACTION TURN Head_Pan
となっており、

「歩く」ボックスでは
PLAY ACTION WALK Head_Pan 30
となっています。

「Head_Pan」は首の水平方向の角度なので、ここでは、ボールが見えている方向を表します。ここで、WAITコマンドが直後でないことに注目してください。このコマンドの後では、「追跡開始」ボックスのPLAY ACTION TRACK_HEAD PINK BALLと「PLAY ACTION TURN」または「PLAY ACTION WALK」のどちらかが同時に実行されます。つまり、ふたつのコマンドが同時に実行されます。



0.01秒待つ

WAIT 10

ここで 10ms(0.01秒)待ちます。

プロパティの変数が見えにくい場合は、ドラッグしてセル幅を調整します。

実行中のコマンドの数によって、アクションを変える

IF Wait 1 10ms待ち
ELSE 見失った?

Waitシステム変数は実行中のコマンドの数です。

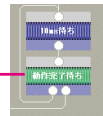
「まわる」または「歩く」ボックスのプログラムを終了したところで、

「追跡開始」ボックスの「PLAY ACTION TRACK_HEAD PINK BALL」と「まわる」ボックスの「PLAY ACTION TURN Head_Pan」または「歩く」ボックスの「PLAY ACTION WALK Head_Pan」のどちらか一方が同時実行されているので、Waitシステム変数は「2」になります。

つまり、「TURN」または「WALK」が終わるまでは「Wait 1」なので、10ms間隔で「TURN」または「WALK」が終わったかループして確認しています。

実行中のコマンドが

「PLAY ACTION TRACK_HEAD PINK BALL」だけになるとWaitシステム変数が「1」になるので「見失った?」ブランチボックスに戻ります。



このようなプログラムをすることで、Waitシステム変数から実行中のコマンドの数を知ることができます。

「キック」グループプログラムを確認しよう

システム変数の値を保存する

SET pan Head_Pan

WAIT 500

「pan」はユーザー変数で、「Head_Pan」システム変数の値を一時保存して、500ms(0.5秒)待ちます。

ボールが止まっているかどうかを確認する

IF Head_Pan = pan キック!

ELSE 追跡継続

0.5秒前に記憶した「Head_Pan」の値と今の

「Head_Pan」の値を比較して、同じ場合は「キック!」

に、異なる場合は「追跡継続」にジャンプします。

「Head_Pan」の値が0.5秒前と変わらない場合は、ボールは静止していると判断してその場所でキックをします。場所が違っていている場合は、ボールが移動したと判断して追跡を継続します。

ボールの追跡を終了し、ボールをキックする

STOP

PLAY ACTION KICK pan 1000

WAIT

STOPコマンドは実行中のコマンドを終了させます。

ここまで「追跡開始」ボックスにあった

PLAY ACTION TRACK_HEAD PINK BALL

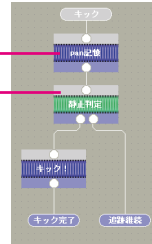
が継続して実行されていましたが、このSTOPコマンドでようやくボールの追跡を終えます。

PLAY ACTION KICK pan 1000

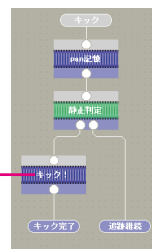
で「pan」ユーザー変数の角度へ距離100cmのキックをしています。ボールの上部をキックして遠くに転がしたいので、100cm先のボールをキックするようにしています。

WAITコマンドでキックを終えるまで待ちます。

「距離」は、AIBOの体の中心からボールと床面の接地点までを表します。



小文字で始まるユーザー変数を必要ときに定義して使うことができます。



端子の順序を変更しよう

「キック」グループボックスの端子を確認します。

「キック」グループボックスには、左側が「追跡継続」、右側が「キック完了」という2つの端子があります。

一方、「キック」グループボックスが呼び出す「kick」グループ内には、左側に「キック完了」、右側に「追跡継続」ターミナルボックスが並んでいます。

グループボックスに表示される端子は、ターミナルリストの順番に依存しています。メニューの[グループ]-[端子の順序設定]でターミナルリストを表示して、確認しましょう。上から「追跡継続」「キック完了」と表示されていますので、「キック」グループボックスでは、この順序で左から右に端子が並んでいるのです。

ラインが交差してしまう場合は、端子の順序を変えて見やすくする工夫をしてください。

これで、「Soccer.be」にあるすべてのボックスを確認しました。

以上で、ビヘイビアのレッスンは終了です！チュートリアルテクニックを活かしてオリジナルなプログラムを作り、AIBOを楽しんでください。

端子の順序の変更は、ユーザーガイド(49ページ)参照。

左側のヒストリーバーを利用して、上下階層のグループに移動して確認しましょう。

では、さっそくレッスンを始めましょう！

ウィンドウ構成

アクションコンポーザーは、以下のようなウィンドウから構成されています。
各ウィンドウの使い方は、記載されている各レッスンで学ぶことができます。

Motion Viewer

モーションのキーフレームが表示される。キーフレームをコピーしたいときなどに表示する。

Lesson 2 キーフレーム確認
Lesson 6 キーフレームコピー



Pose Windowへ

スタート



メインウィンドウ
素材やアクションは
ライブラリで管理

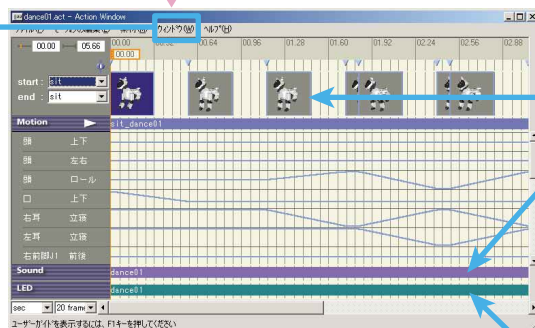
Lesson 1
アクション
ファイル作成

MIDI
Windowへ

Action Window*

モーションのタイミングを調整したり、アクションを構成するすべての素材を確認できます。

Lesson 1 アクションファイル作成
Lesson 4 素材作成・組み合わせ
Lesson 5 同期
Lesson 6 キーフレームコピー



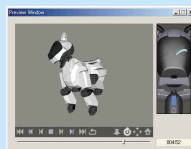
WAVE
Windowへ

LED
Windowへ

Preview Window

作成したAIBOのアクションをPCの画面上で確認できる。

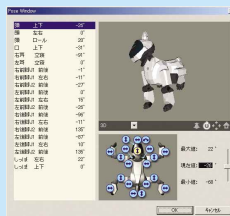
Lesson 2
モーション・サウンド・LED確認



Pose Window

AIBOのポーズ(モーションの中継点となるキーフレームの姿勢)を編集する。

Lesson 3 ポーズ編集
Lesson 4 素材作成



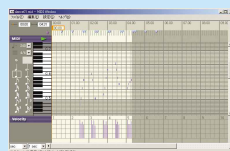
備考

* 各ウィンドウは、メインウィンドウ上のファイル名をダブルクリックしても表示できます。また、[スタート]メニューからの表示も可能です。

MIDI Window*

AIBOが鳴らすMIDI音を作成する。ピアノロール形式により、音の動きや音符の長さを視覚的に確認できる。

Lesson 2 MIDIサウンド確認
Lesson 4 素材作成
Lesson 5 同期



WAVE Window*

AIBO で鳴らすWAVE音を編集する。

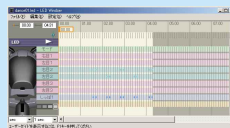
Lesson 2 WAVEサウンド確認
Lesson 3 ポーズ編集
Lesson 4 素材作成
Lesson 5 同期



LED Window*

AIBOのランプのLEDの点滅パターンを作成する。

Lesson 2 LED確認
Lesson 3 ポーズ編集
Lesson 4 素材作成
Lesson 5 同期

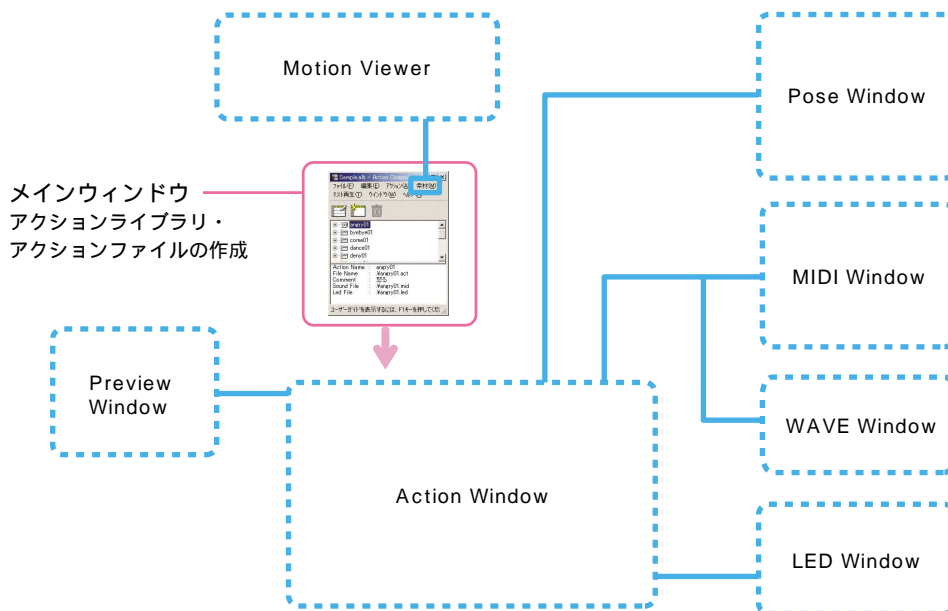


アクションライブラリを作る！

目標：アクションライブラリとアクションファイルの作成方法を学習する

アクションコンポーザー*では、モーション、音、ランプ(LED)の素材と、素材を組み合わせたアクションファイル、アクションファイルを登録してまとめたアクションライブラリを作ることができます。作成したアクションライブラリを使い、クイックビヘビアアレンジャーやビヘビアアレンジャーでアクションをビヘビアプログラムに組み込みます。

Lesson1で使うウィンドウ



作業手順：

- 準備 (73ページ)
- アクションライブラリ作成 (73ページ)
- アクション作成 (74ページ)
- アクションをアクションライブラリに追加 (76ページ)
- アクションライブラリ保存 (77ページ)

ひとつのビヘビアで使えるアクションライブラリはひとつです。ビヘビアごとにアクションライブラリを作っても良いですし、(機種ごとに)作ったすべてのアクションをひとつのアクションライブラリにまとめても良いです。

*オリジナルのアクションを作るためのツール。詳しくはユーザーガイド(72ページ)参照。

ビヘビアアレンジャーでアクションをプログラムに組み込む方法は、43ページ参照。

アクションやアクションライブラリ、素材は好きな順番で作ることができますが、ここでは、「アクションライブラリ アクションファイル」の順に作成します。

準備しよう

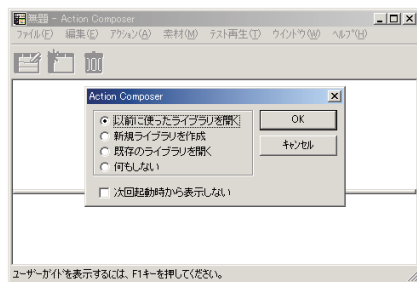
- AIBOの機種に適したシステム(日本語版)をコピーした“メモリースティック”をご用意ください。
- ワイヤレスLANをご利用の方は、設定を行ってください。設定方法について詳しくは、ユーザーガイド(23ページ)をご覧ください。
- レッスンで使うファイルは以下の「Materials」フォルダ内にあります。あらかじめMaterialsフォルダごとマイドキュメントなどの作業用フォルダにコピーしてからお使いください。
¥AIBOマスタースタジオのインストールフォルダ ¥Sample ¥機種名(J) ¥Tutorial ¥Materials
Tutorialフォルダ内のファイルについては、「チュートリアルで使うファイル」(12ページ)を参照してください。

チュートリアルでは、日本語版(J)を選んでください。

ライブラリを作成しよう

1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[アクション コンポーザー]を選ぶ

アクションコンポーザーが起動し、ダイアログ*が表示されます。



* このダイアログを「ナビゲーションダイアログ」といいます。表示しないように設定したときは、メニューの[ウィンドウ]-[オプション]で表示できます。

2 「新規ライブラリを作成」を選び、[OK] をクリックする

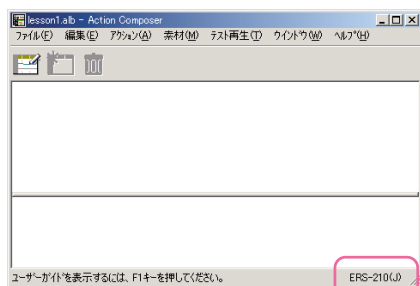
ナビゲーションダイアログを非表示にしている場合は、メニューの[ファイル]-[新規作成]を選びます。

3 お使いのAIBOを選び、[OK] をクリックする



日本語版が英語版を選択できますが、日本語版のシステムの“メモリースティック”を使いますので、日本語版(J)を選んでください。(機種名・言語は“メモリースティック”のシステムとビヘイビアとアクションライブラリであわせておく必要があります。)

4 ファイル名に「lesson1.alb」と入力し、コピーしたMaterialsフォルダに保存する メインウィンドウが表示されます。

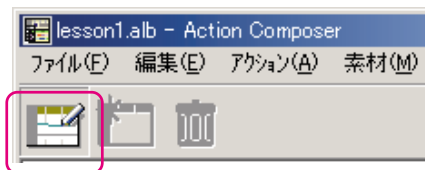


ここに手順3で選んだ機種名(J)が表示されます。

アクションを作成しよう

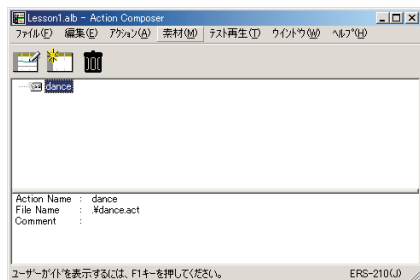
既存の素材を使って、アクションライブラリ中にアクションファイルを作成します。

1 新規アクションボタン をクリックする



2 「dance.act」というファイル名で保存する

アクションファイルが作成され、メインウィンドウに「dance」アクションが表示されます。



3 手順2のdanceをクリックして選び、コピーしたMaterialsフォルダから素材を追加する

モーションを追加

[素材] - [モーションの追加] で sit_dance01.mtn を選択し、[開く] をクリックする

サウンドを追加

[素材] - [サウンドの追加] で dance01.mid を選択し、[開く] をクリックする

LEDを追加

[素材] - [LEDの追加] で dance01.led を選択し、[開く] をクリックする

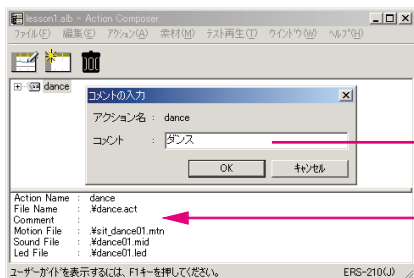
あらかじめ用意されている素材を組み合わせたアクションファイルが完成しました。

選択されたアクションファイルに素材が追加されます。

「アクションファイルに素材を追加する」とは、素材がライブラリ（アクション）にコピーされるのではなく、素材ファイルを参照しているだけです。ですから、追加した素材を編集する（Lesson3以降）と、参照先のファイルが直接編集されてしまうので、注意が必要です。素材ファイルをコピーしてから作業する習慣をつけてください。

4 アクションにコメントをつける

danceを選択した状態で、メニューの[アクション]-[コメントの入力]を選び、「ダンス」と入力し、[OK]をクリックします。



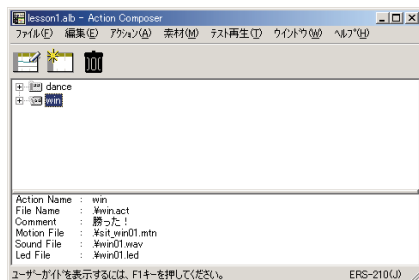
ここに表示される

既存のアクションをライブラリに追加しよう

サンプルなど、すでにあるアクションファイルをアクションライブラリに追加します。

1 メニューの[アクション]-[アクションの追加]でコピーしたMaterialsフォルダの「win.act」を選び、[開く]をクリックする

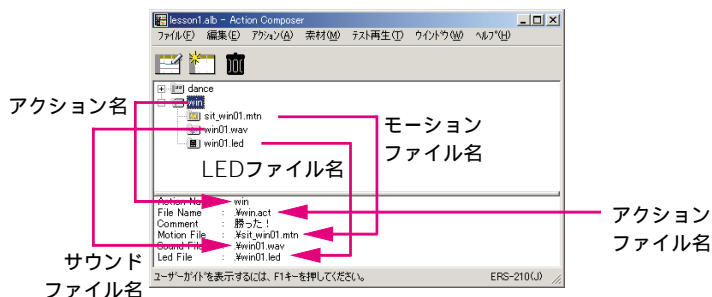
メインウィンドウにアクションファイルが追加されます。



2 winの先頭にある **+** をクリックする

アクションファイルに追加されている素材が確認できます。

winを選ぶと、下にwinアクションファイルのコメントなどの詳細が表示されます。



(ERS-210の場合)

メインウィンドウでは、アクションや素材ファイルをツリー構造で見ることができます。

ERS-220/310ではwinにLEDはついていません。

3 メニューの[ファイル]-[ライブラリを上書き保存]でアクションライブラリを保存する

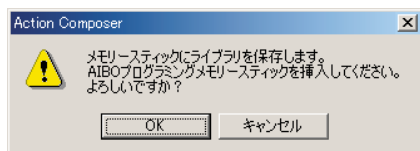
ライブラリの内容が更新されます。

上書き保存をしないと、アクションライブラリやライブラリ内に含まれるアクションの内容が更新されません。メインウィンドウでの操作を更新する場合は、必ず上書き保存をしてください。

“メモリースティック”に保存しよう

システムがコピーされている“メモリースティック”を用意し、作成したアクションライブラリを“メモリースティック”に保存します。保存することで、ビヘイビアで、アクションライブラリのアクションが使えるようになります。

1 “メモリースティック”をPCに挿入し、メニューの[ファイル]-[メモリースティックにライブラリを保存]を選ぶ 確認のダイアログが表示されます。

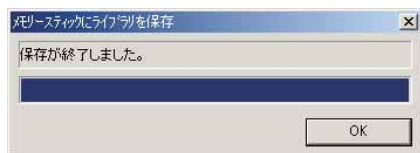


システムをコピーする方法は、ユーザーガイド(21ページ)を参照してください。アクションライブラリとシステムの機種・言語が異なると、アクションライブラリは保存できません。

“メモリースティック”に保存できるアクションライブラリのは数はひとつです(44ページ参照)。

2 [OK]をクリックする

アクションライブラリが保存され、保存終了後にダイアログが表示されますので、[OK]をクリックしてください。



同時に、PCのハードディスクにもアクションライブラリがコピーされます。

このコピーされた内容をビヘイビアアレンジャーやクイックビヘイビアアレンジャーで参照することができ、ビヘイビアにライブラリのアクションが使えるようになります(51ページ参照)。

ワイヤレスLANを使わない方は、ここでLesson1は終了です。

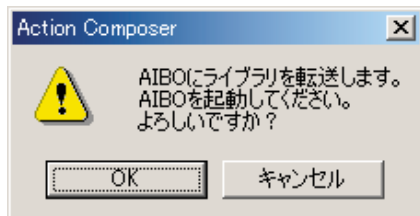
ワイヤレスLANを使って保存してみよう

ワイヤレスLANをお使いの場合は、PCからAIBOにアクションライブラリを転送できます。

1 “メモリースティック”をAIBOに挿入して、AIBOを起動したままPCから通信できる範囲に置く

2 メニューの[ファイル]-[AIBOにライブラリを転送]を選ぶ

確認ダイアログが表示されます。

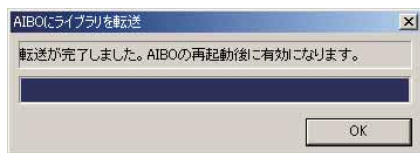


すでに作成したアクションライブラリは“メモリースティック”に保存済みですが、ここでは勉強のため、ワイヤレスLAN経由で再度保存しましょう。

3 [OK]をクリックする

アクションライブラリをAIBOに転送します。

転送が終了すると、ダイアログボックスが表示されますので、[OK] をクリックします。



アクションライブラリを保存しただけでは、AIBOは動きません。ここまですでにアクションコンポーザーの役目は終了です。次は(クイック)ビヘイビアアレンジャーにボタンタッチなのですが、続けてもっとアクションコンポーザーについて学びましょう。

AIBOを再起動するとアクションライブラリが有効になります。再起動しないで、ビヘイビアプログラムを保存しプログラムを実行しても、アクションライブラリのアクションは再生されません。

このレッスンで作成したアクションライブラリは「L1_Reference」フォルダ内の「lesson1.alb」と同じです。アクションライブラリが正しくできたか確かめてください。

「L1_Reference」フォルダの場所は、「チュートリアルで使うファイル」(12ページ)を参照。

既存のアクションや素材を組み合わせるアクションライブラリを作る方法は、わかりましたか？

Lesson1では、素材やアクションの内容を確認せずにライブラリを作りました。Lesson2では、追加した内容を目と耳で確認しましょう！

作成したアクションを確認しよう！

目標：アクションの確認方法を学習する

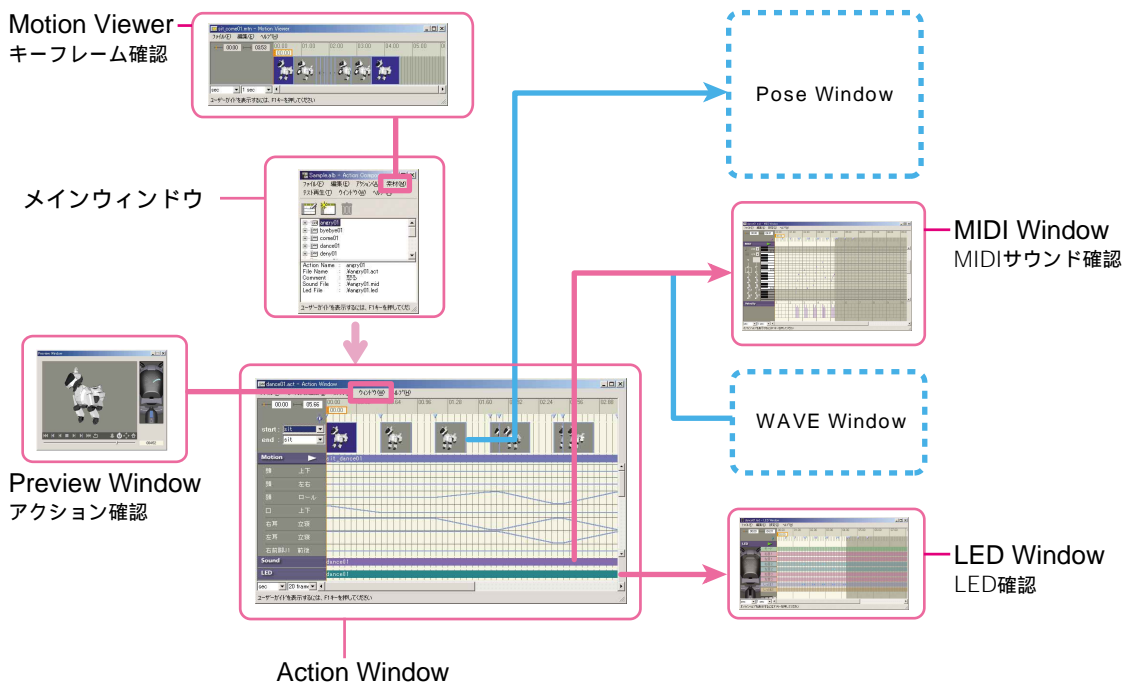
作成したアクションファイルや既存のファイルの動作を実際に目で確認しましょう。

動作を確認するには、次の2つの方法があります。

- PCの画面上で確認(81ページ)
- AIBOで実際に動かす(ワイヤレスLANが必要)(85ページ)

アクションライブラリに未登録のアクションファイル・素材の確認方法も学習しましょう。


Lesson2で使うウィンドウ

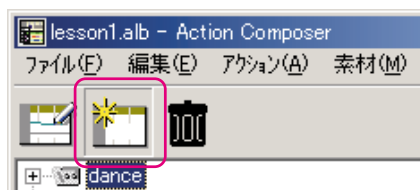


PC上で、アクションと素材を確認しよう

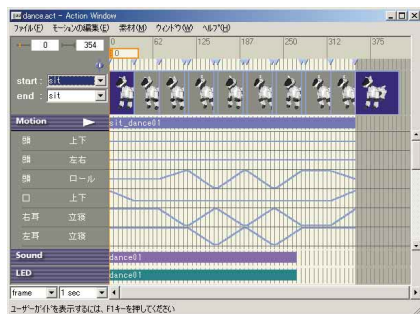
Action Windowでモーション・サウンド・LEDを確認していきます。

1 Lesson1で使ったアクションライブラリをメインウィンドウで開き、「dance」アクションファイルを選ぶ

2 アクション編集ボタン  をクリックする



Action Windowが起動し、選んだアクションのデータが表示されます。



Lesson2から始める方は「L2_Starter」フォルダの「tutorial2.alb」を開いてください（12ページ参照）。編集しませんので、コピーして使う必要はありません。

手順1でアクションファイルをダブルクリックしても同じ操作になります。

Action Windowでは、組み合わせた素材を一覧で見ることができます。モーションのキーフレームや音・LEDのファイル名や長さが表示されます。メインウィンドウでファイルの詳細を文字で見るとよりも、一步深くアクションを理解することができます。

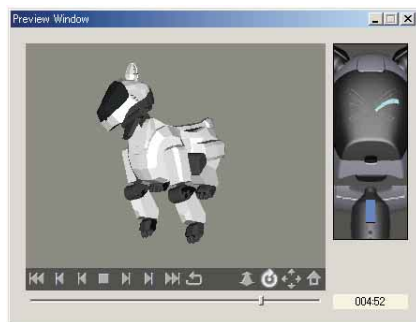
モーションを確認

キーフレームやサウンドとLEDのファイル名だけではアクションが想像しにくいので、PCで再生してみます。

1 Action Windowの 再生ボタン▶をクリックする



Preview Windowが表示され、アクションの再生が始まります。



ここでは、アクションに含まれる素材(モーション・音・LED)と一緒に再生されます。

もう一度再生するには、Preview Window内の▶をクリックします。

ご注意

お使いのPCの性能によっては、フレームが間引かれて再生されます。キーフレーム、サウンド、LEDの個々のデータも確認することをお奨めします。

2 ✕をクリックしてウィンドウを閉じる

サウンドを確認

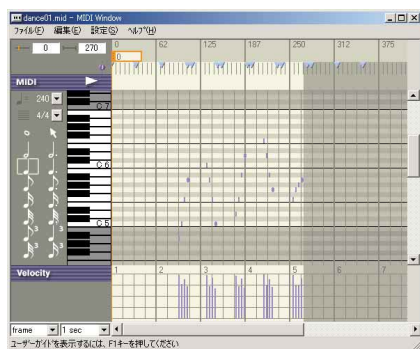
Preview Windowでは耳でサウンドを確認できますが、ここでは目で確認してみましょう。

- 1 Action Windowのサウンドチャンネルに表示されている「dance01」をダブルクリックする



メインウィンドウ内のサウンドファイルやLEDファイルをダブルクリックしても各ウィンドウが表示されます。

MIDI Windowが表示され、MIDIデータの配置を見ることができます。



- 2 MIDI Windowの再生ボタン▶をクリックする



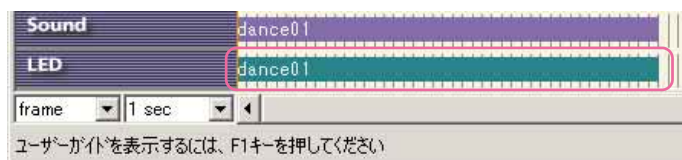
音の再生が始まります。

- 3 ウィンドウを閉じる

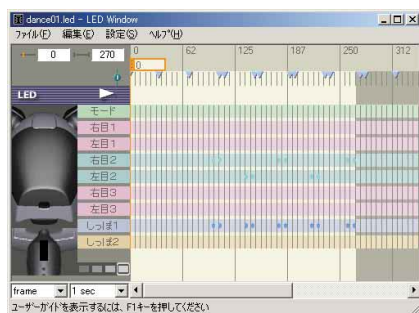
LEDを確認

サウンドと同様に、目でLEDのデータを確認します。

- 1 Action WindowのLEDチャンネルに表示されている「dance01」をダブルクリックする



LED Windowが表示され、LEDデータの配置を見ることができます。



- 2 ▷をクリックする

左側のAIBOでLEDの光り方を確認できます。



- 3 ウィンドウを閉じる

Action Windowを閉じます。

ワイヤレスLANを使わない方は、86ページへ進んでください。アクションライブラリに未登録の素材・アクションを確認します。

AIBOで確認しよう

ワイヤレスLANをお使いの場合は、AIBOを実際に動かしてアクションを確認していきます。PC上で3Dモデルを使って仮に確認するのは違い、AIBOをリアルワールドで動作させた結果を見ることができません。

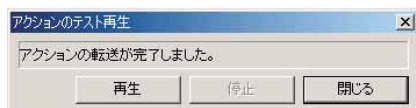
ワイヤレスLANをお使いにならない場合は、テスト的な実行をするためにも、アクションをビヘイビアに組み込んでからAIBOで実行する必要があります。

- 1 “メモリースティック”をAIBOに挿入して起動し、PCから通信できる範囲に置く

- 2 メインウィンドウでアクションや素材を選んでから、メニューの[テスト再生]を使って再生する

アクションの確認

アクションファイルを選び、[テスト再生]-[アクションのテスト再生]を選ぶ



ワイヤレスLANでAIBOの動作をテスト実行するには、確認するアクションファイルや素材ファイルを、あらかじめアクションライブラリに登録しておく必要があります。

テスト実行の場合は、ファイルは“メモリースティック”に保存されません。

モーションの確認

モーションファイルを選び、[テスト再生]-[モーションのテスト再生]を選ぶ

サウンドの確認

サウンドファイルを選び、[テスト再生]-[サウンドのテスト再生]を選ぶ

LEDの確認

LEDファイルを選び、[テスト再生]-[LEDのテスト再生]を選ぶ

- 3 データが転送されたら[再生]をクリックする

AIBOで再生されるアクションを確認してください。

すぐに再生を止めたいときは、[停止]をクリックしてください。ただし、AIBOは不安定な姿勢のまま止まりますので、アクションやモーションを再生しているときは、倒れたりすることがあります。

-
- 4 AIBOの動きが止まったら、[閉じる]をクリックし、すべてのウィンドウを閉じる
-

では、「アクションライブラリに登録していないと、アクションファイルや素材は確認できないの?」と疑問に思われるでしょうが、大丈夫です。次の方法で、確認しましょう。

ライブラリに未登録のアクションや素材を確認しよう

メインウィンドウからではなく、[アクション コンポーザーツール] からアクションコンポーザーを構成する各ウィンドウを起動して確認します。

アクションを確認

-
- 1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[アクション コンポーザーツール]-[アクション ウィンドウ]を選ぶ
-

- 2 お使いのAIBOを選び、[OK]をクリックする



-
- 3 メニューの[ファイル]-[開く]を選び、Lesson1で作った「dance.act」を選び、[開く]をクリックする

あとは、82ページと同様です。
確認を終えたら、ウィンドウを閉じてください。

アクションライブラリでは、日本語版か英語版かを指定しなければいけませんが、アクションファイルや素材には、日本語版・英語版の区別がありません。機種名だけを選択してください。

Lesson2から始める方は「L2_Starter」フォルダをコピーして「tutorial2.act」を開いてください(12ページ参照)。

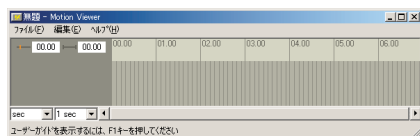
素材を確認

モーションの確認

モーションは、アクションと同様にAction Windowで素材としてモーションを追加して確認する方法もありますが、ここでは勉強のため、別のウィンドウで確認します。

- 1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[アクション コンポーザーツール]-[モーション ビューワー]を選ぶ

モーションビューワーが表示されます。



- 2 メニューの[ファイル]-[開く]を選び、Lesson1で使った「sit_dance01.mtn」ファイルを開く

キーフレーム*が表示されます。キーフレームでモーションを確認しましょう。



モーションビューワーは、キーフレーム表示のみで、再生したりキーフレームの編集はできません。再生や、キーフレームの編集は、Action Windowで行います。モーションビューワーは、キーフレームをコピーして再利用するときに、フレームのコピー元として使うと便利です。

メインウィンドウでモーションファイルをダブルクリックしてもこのウィンドウが起動します。

* キーフレームについては、ユーザーガイド(72ページ)参照。

サウンドの確認

- 1 Windowsの[スタート]ボタンをクリックし、[プログラム] - [AIBOマスタースタジオ] - [アクション コンポーザーツール] - [MIDI ウィンドウ]を選ぶ

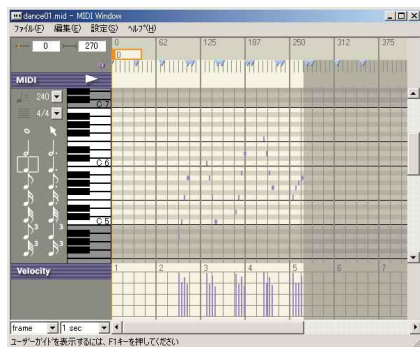
MIDI Windowが表示されます。

[WAVE ウィンドウ]をクリックすると、WAVE Windowが表示されます。

音のフォーマットはAIBOの3機種とも共通なので、機種名は選択しません。つまり、サンプルで用意されている別の機種のサウンドファイルを使っても良いのです。

- 2 メニューの[ファイル]-[開く]を選び、Lesson1で使った「dance01.mid」ファイルを開く

あとは、83ページの手順2～3と同様です。



LEDの確認

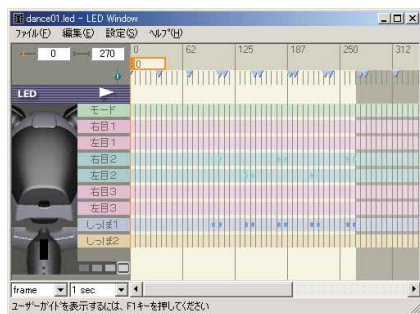
- 1 Windowsの[スタート]ボタンをクリックし、[プログラム] - [AIBOマスタースタジオ] - [アクション コンポーザーツール] - [LED ウィンドウ]を選ぶ

- 2 お使いのAIBOを選ぶ

LED Windowが表示されます。

3 メニューの[ファイル]-[開く]を選び、Lesson1で使った「dance01.led」ファイルを開く

あとは、84ページの手順2～3と同様です。



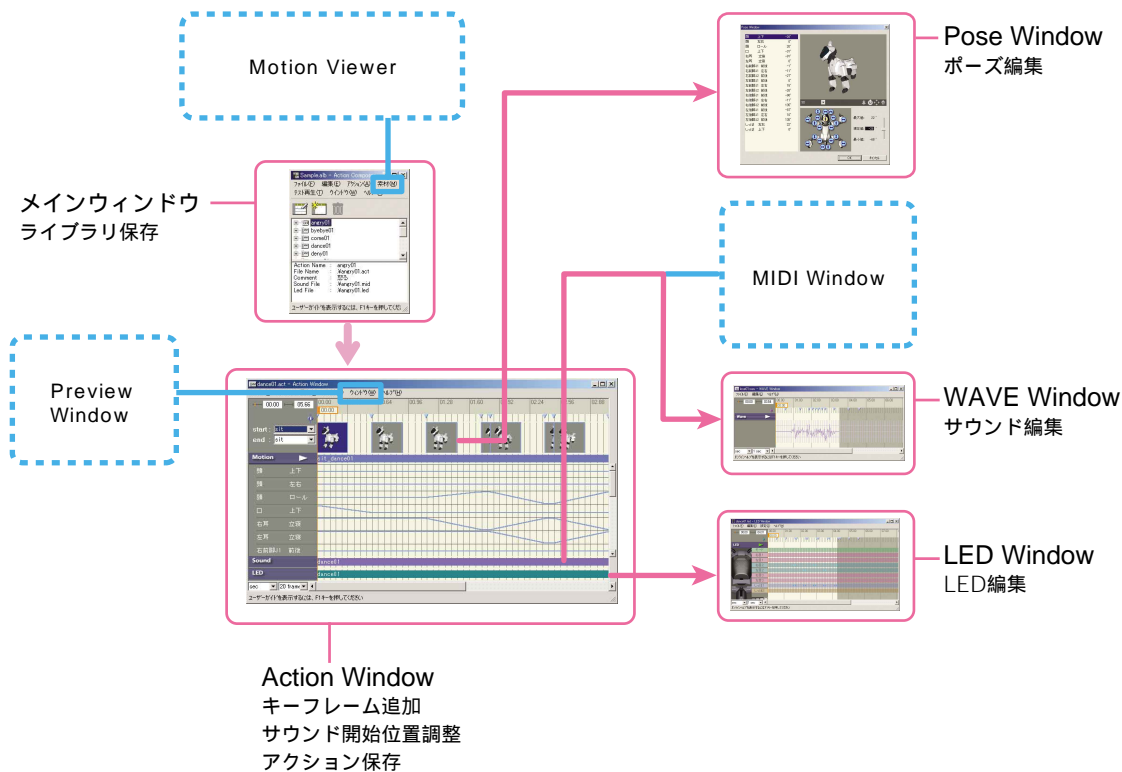
これで確認の勉強は終わりです。

ここまで理解したら、既存の素材やアクションを組み合わせでアクションライブラリが作れます！

次は、ファイルや素材を編集して、あなたオリジナルのアクションを作りましょう。

目標：素材の編集方法を理解する

Lesson3で使うウィンドウ



作業手順

素材準備(91ページ)

モーショ編集(92ページ)

「ゆっくり頭を上げ、少し速く下げる」を「すばやく頭を上げ、下げる」モーションにする

サウンド編集(94ページ)

無音部分を削除し、音の開始位置を調整する

LED編集(96ページ)

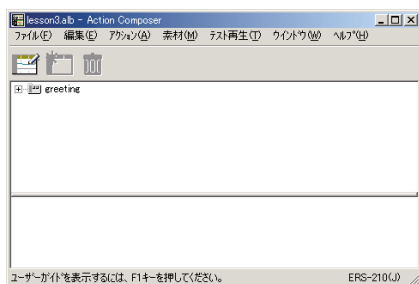
モーションに合わせて点滅パターンにする

アクションファイル保存(100ページ)

素材を準備しよう

1 アクションコンポーザーを起動する

2 ナビゲーションダイアログで[既存のライブラリを開く]を選び、コピーしたMaterialsフォルダから「lesson3.alb」ライブラリを開く



Lesson3から始める方は、Materialsフォルダをコピーしてから、レッスンを始めてください(73ページ)。

ナビゲーションダイアログを非表示にしている場合は、メニューの[ファイル]-[ライブラリを開く]を選んでください。

3 「greeting」アクションファイルを選ぶ

「greeting」アクションファイルをダブルクリックしてAction Windowを開くこともできます。

4 アクション編集ボタン をクリックする

Action Windowが開きます。

再生して、ゆっくり頭を上げ、「はじめまして」と言い、少し速く頭を下げるモーションを確認します。

このAction Windowから素材を編集していきます。

モーションを編集しよう

頭をすばやく上げるモーションを作りましょう。

ポーズを編集

1 2番目のキーフレームをダブルクリックする

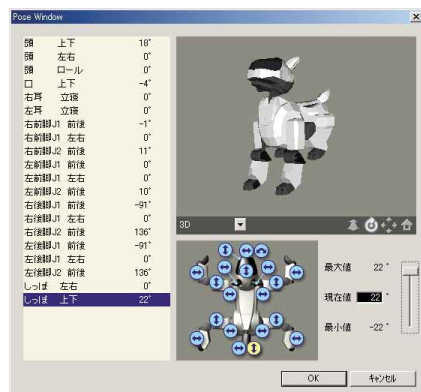
Pose Windowが開きます。

ここで、キーフレームのポーズを編集します。

2 部位リストで角度を変更する部位を選び、「現在値」に角度を入力し、Enterキーを押す

AIBO平面図(下中央の展開図)の「↑」「↓」「←」「→」からも選択できます。

角度は、直接入力か、右のスライダーを上下して入力します。



ERS-210/220	頭上下	18°
	しっぽ上下	22°
ERS-310シリーズ	頭上下	20°

3DのAIBOが動きました！

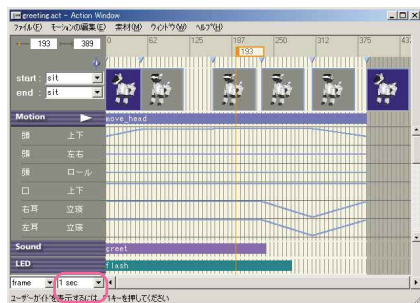
3 [OK]をクリックして、Pose Windowを閉じる

Action Windowの2番目のキーフレームが変わったことを確認してください。

キーフレームを追加

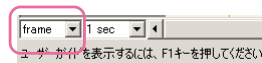
1 タイム/フレームチャンネルの「193フレーム(3.10秒)」をクリックする

カレントバーが移動し、現在位置が変わります。



(時間幅* : 1 sec時)

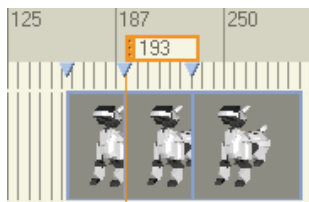
タイム/フレームチャンネルの表示単位(秒/フレーム)は、Action Window左下のsec/frameボックスで切り換えます。



* 時間幅とは、タイム/フレームチャンネルの数値から数値までの幅を意味します。データを入力したり、フレームを選ぶときは、ドロップダウンリストで時間幅を大きく設定すると作業をしやすくなります。

2 メニューの[モーションの編集]-[キーフレーム化]を選ぶ

現在位置に、その時点でのポーズのキーフレームが作成されます。



希望の位置にキーフレームを作り、ポーズを編集することでモーションを作り上げていきます。

3 キーフレームのポーズを設定する

92ページの手順1～3と同様に、各部位を以下のとおりに設定してください。

ERS-210	頭 ロール	19°
ERS-220	頭 左右	30°
ERS-310シリーズ	頭 左右	30°

これでモーションの変更は終了です。

メニューの[素材]-[モーションに名前を付けて保存]で、「move_head_2.mtn」という名前で保存します。モーションチャンネルのモーション名も変更されます。

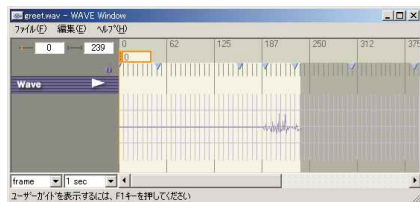
サウンドを編集しよう

サウンドファイルの編集

無音部分を削除してみましょう。

1 Action Windowのサウンドチャンネルバーの「greet」(ファイル名)をダブルクリックする

WAVE Windowが表示されます。



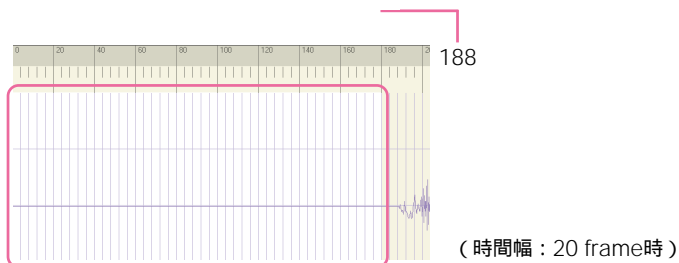
「はじめまして」という声が入っていますが、先頭に無音部分があります。

WAVEファイルは容量が大きいので無音部分をできるだけ削除し、「ここぞ!」というアクションに使うように、工夫してください。

AIBOで再生できるWAVEファイルは、「8KHz、8bit、Mono、PCM」形式です。Windowsのサウンドレコーダーで音を録音したり、フリーの音楽ファイルを変換して使ってください。

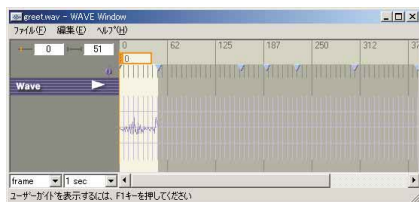
2 0フレーム(0秒)をクリックし、Shiftキーを押しながら188フレーム(3秒)をクリックする

無音部分のサウンドチャンネルが白くなり、選択されます。



3 メニューの[編集]-[切り取り]を選ぶ

選択した無音部分が切り取られ、データが前に移動します。



[削除]の場合は、音は無音になりますが、ファイルの長さは変わりません。

4 メニューの[ファイル]-[名前を付けて保存]を選び、「greet_2.wav」というファイル名で保存する

無音部分を削除したサウンドファイルができました。

ウィンドウを閉じましょう。

5 Action Windowでいったんサウンドファイルを削除してから、再度「greet_2.wav」をアクションに組み込む

メニューの[素材]-[サウンドの削除]で削除してから、[素材]-[サウンドの追加]で追加します。

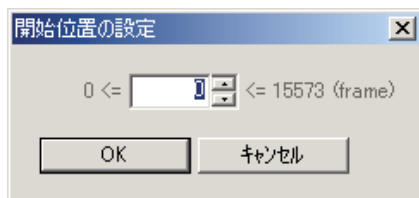
組み込み済みのサウンド・LED
ファイルを編集したら、再度組み込み直してアクションを手動で更新する必要があります。

サウンドの開始位置の調整

先頭の無音部分を切り取ったので、音が前に移動してしまいました。
開始位置を調整しましょう。

- 1 Action Windowで[素材]-[サウンドの開始位置設定]
を選ぶ

[開始位置設定]ダイアログボックスが表示されます。



- 2 削除したフレーム数(95ページの手順2)の「188」を入力し、[OK]をクリックする

ファイルの位置が移動し、音の開始位置がもとの位置に戻ります。

「188」が開始位置になります。
WAVEのように容量が大きいファイルは、開始位置を調整して、できるだけ効果的なところにサウンドを置く工夫が必要です。

LEDを編集しよう

モーションに比べLEDの長さが短いので、LEDを編集していきましょう。

LEDデータを追加する

LEDを足して点滅パターンを作っていきます。

- 1 Action WindowのLEDチャンネルバーの「flash」(ファイル名)をダブルクリックする


LED Windowが表示されます。

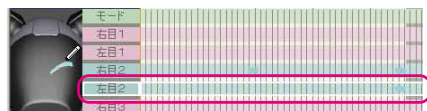
2 [輝度選択]ボタンの一番右をクリックする

一番明るい輝度を選びます。



3 AIBOのLEDをクリックする

マウスカーソルが  になります。

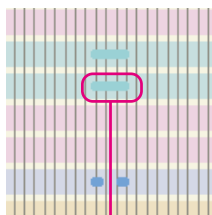


LEDデータを配置するモードになります。クリックすると、選んだLEDが、LEDチャンネルのどの位置に相当するのかがわかります。

4 LEDデータを下図のとおり配置する

ERS-210

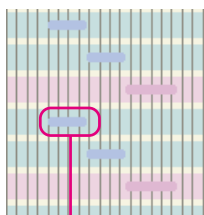
最初は片目しか光らないので「左目2」に追加して両目を光らせよう



ここを入力

ERS-220

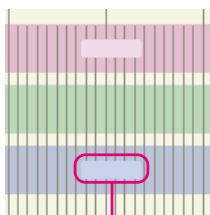
順番に点灯するように、「フェイスL1」に一番明るい輝度を追加しよう



ここを入力

ERS-310シリーズ

青を混ぜてピンクにしよう



ここを入力

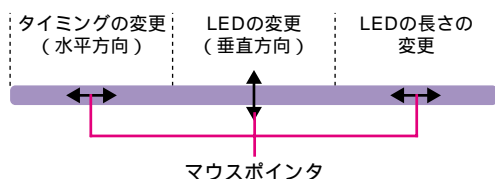
(時間幅: 1/2 sec時)

データを移動したり長さを変えるには

LEDやサウンドデータの上にマウスカーソルを置いてドラッグすると、データを移動したり長さを変えることができます。カーソルの位置によってできることが異なります。

データを置き間違えたら

[編集]-[元に戻す]で間違えたデータをクリアして置きなおします。



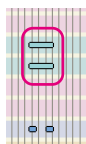
LEDデータがすでにある位置に新しいLEDデータを作成しようとしても、上書きできません。新しいデータを作成する際は、まず空きエリアでLEDデータの長さを変更してから、希望の位置に配置する方法が良いでしょう。

LEDの点滅を繰り返す

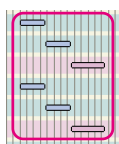
点滅を繰り返して、LEDの長さをモーションに合わせましょう。

- 1 メニューの[設定]-[選択モード]を選び、マウスをドラッグしてLEDデータを選ぶ

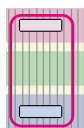
ERS-210




ERS-220



ERS-310
シリーズ



「入力モード」はデータをチャンネルに配置するモードです。LEDではAIBOのLEDを、MIDIでは音符をクリックすると入力モードになります。

「選択モード」はデータを編集するモードです。チャンネル上のデータを選んだり移動したりできます。(MIDIでは  をクリックすると選択モードになります。)

- 2 メニューの[編集]-[コピー]を選ぶ

- 3 タイム/フレームチャンネルの「160フレーム(2.56秒)」フレームをクリックする

カレントバーが移動します。

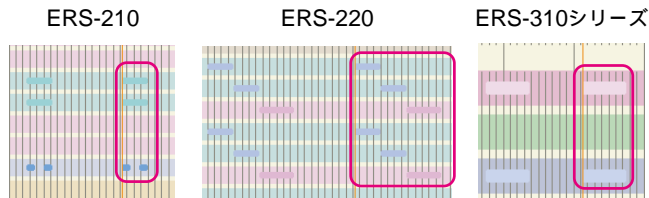


(時間幅: 1/2 sec時)

ここから再度点灯させます。

4 メニューの[編集]-[上書き貼り付け]を選ぶ

最初の点灯と同じ点灯パターンがコピーされます。



(時間幅: 1/2 sec時)

▷で点滅を確認しましょう。

5 「flash_2」という名前で保存する

6 Action WindowでLEDファイルをアクションに組み込み直し、▷をクリックする

95ページ手順5参照

Preview Windowでモーションと一緒に点滅を確認しましょう。

アクションファイルを保存しよう

-
- 1 Action Windowの[ファイル]-[上書き保存]でアクションを保存し、ウィンドウを閉じる

アクションを構成する素材が、新しく作った素材に置きかわります。

-
- 2 メインウィンドウの[ファイル]-[上書き保存]でアクションライブラリを保存する
-

このレッスンで作成したファイルは「L3_Reference」フォルダ内のファイルと同じです(12ページ参照)。ファイルが正しくできているか確かめてください。

これで、アクションコンポーザーの基本的な使い方と(クイック)ビヘイピアレンジャーで使うためのアクションライブラリの作り方は、マスターしました！

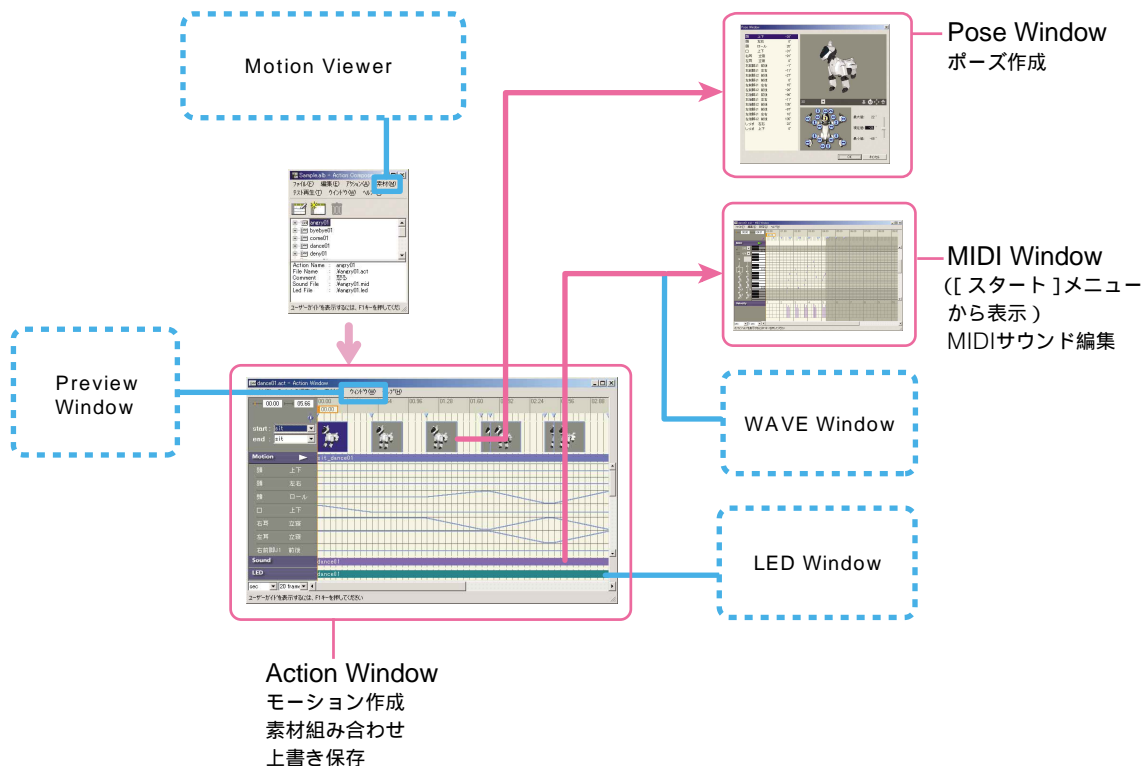
ここまでのレッスン内容を少しずつ自分なりに変えると、あなたオリジナルのアクションを作ることができます！

オリジナルな素材を作ろう！

目標：素材の作成方法を学習する

アクションライブラリの使い方を覚えたら、あとは作品作りに入っていきます。

Lesson4で使うウィンドウ



作業手順：

アクション作成 (102ページ)

Lesson4では、初級とは異なり、アクションから作ります。

モーション作成 (102ページ)

簡単なモーションを新規作成しましょう。

サウンド編集 (105ページ)

MIDIの入力方法を覚えましょう。

素材を組み合わせる (107ページ)

アクションを作成しよう

アクションライブラリを作ってからその中に含まれるアクションを作る方法もありますが、逆にアクションを作って、後でアクションライブラリに追加する方法もあります。

アクションライブラリからではなく、アクションファイルを単独で作成します。

- 1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[アクションコンポーザーツール]-[アクション ウィンドウ]を選ぶ

アクションファイルを作るときは、日本語版・英語版の区別が必要ありません。

- 2 お使いのAIBOを選ぶ

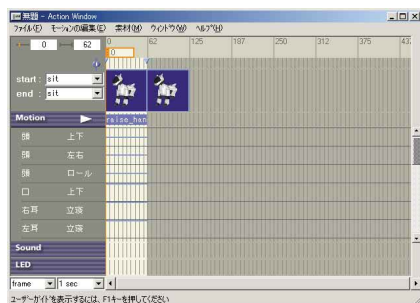
Action Windowが表示されます。

モーションを作成しよう

モーション作りの勉強として、とても簡単な「座って前脚と一緒に頭を上げる」モーションを作ります。

- 1 [素材]-[モーションの新規作成]を選ぶ
名前を付けて保存ダイアログが表示されますので、「raise_hands」というファイル名で保存します。
開始と終了のポーズが表示されます。

中身を作りこむ前に、最初にファイルを作成します。



2 [start] [end]ともに「sleep」(伏せる)に変更する

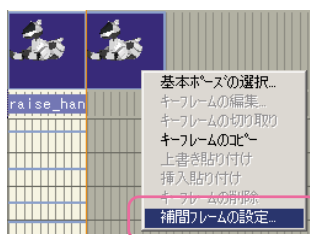
開始と終了のポーズが変わります。



開始と終了のポーズを基本姿勢から選ぶことで、ビヘイビアを作るときにアクション間で姿勢をそろえずにすみずみ。アクション間はAIBOが自動的に基本姿勢をつなげてくれます。

3 「end」キーフレームを右クリックし、「補間フレームの設定」を選ぶ

補間フレームを長くして、モーション全体を長くします。



4 「450」と入力し、[OK]をクリックする

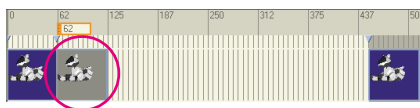
補間部分が長くなり、全体で451フレームのモーションになります。



5 タイム/フレームチャンネルで「62フレーム(1秒)」をクリックし、キーフレーム化する

キーフレームが作成されます。

93ページ参照



6 キーフレームをダブルクリックし、Pose Windowで下図と同じポーズを作成する

ERS-210

[頭 上下]	23°
[右前脚J1 前後]	107°
[右前脚J2 前後]	80°
[左前脚J1 前後]	107°
[右前脚J2 前後]	80°



ERS-220

[頭 上下]	23°
[右前脚J1 前後]	107°
[右前脚J2 前後]	80°
[左前脚J1 前後]	107°
[右前脚J2 前後]	80°

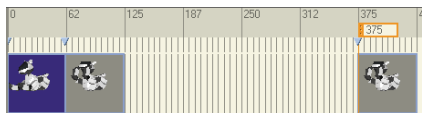


ERS-310シリーズ

[頭 上下]	20°
[頭 上下2]	20°
[右前脚J1 前後]	107°
[右前脚J2 前後]	80°
[左前脚J1 前後]	107°
[左前脚J2 前後]	80°



- 7 手順6で作成したキーフレームをコピーし、「375フレーム (6.00秒)」に上書き貼り付けする



- 8 「230フレーム (3.68秒)」にキーフレームを作り、ポーズを設定する

各機種共通

[右前脚J1 前後] 60 °
[左前脚J1 前後] 60 °

- 9 メニューの[素材]-[モーションに名前を付けて保存]で「raise_hands.mtn」を上書き保存する

アクションを保存しないでAction Windowを閉じます。

Action Windowからモーションを新規に作成したときは、[素材]-[モーションに名前を付けて保存]で手順1のファイル名を選ぶと、アクションファイルを作らずにモーションファイルのみを作ることができます。

サウンドを作成しよう

サウンドファイルとLEDファイルは、アクションウィンドウでは新規作成できません。アクションコンポーザーツールから、各編集ウィンドウを起動して作成します。

作成した後、アクションファイルに組み込みます。

ここでは、用意されたファイルを使って、MIDIで音を入力する方法を学習しましょう。入力する方法を覚えたら、楽譜を見て主旋律を打ち込んでいくだけで、MIDIファイルが作れます。

市販のMIDIエディターで作った音を使うこともできますが、アクションコンポーザーのMIDIウィンドウで作ると、AIBOに適したサウンドを作りやすく、おすすめです。

- 1 Windowsの[スタート]ボタンをクリックし、[プログラム]-[AIBOマスタースタジオ]-[アクション コンポーザーツール]-[MIDI ウィンドウ]を選ぶ

MIDI Windowが表示されます。

2 [ファイル]-[開く]で「jinglebells.mid」ファイルを開く

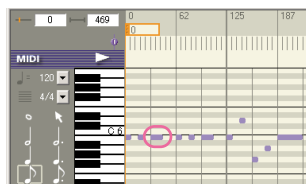
コピーしたMaterialsフォルダ内にあります。

3 ▷で再生する

ジングルベルに似ていますが、ちょっと変です。
足りないところに音を追加し、ずれている音程をなおしましょう。

4 音符をクリックし、下図のように1音を入れる

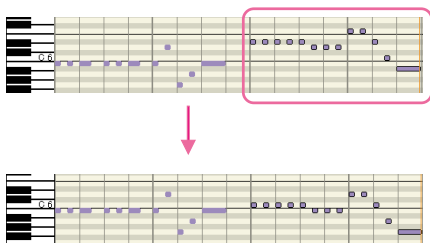
抜けている部分に音を入力します。
位置がずれたら、ドラッグして直しましょう。



音符をクリックすると、サウンドチャンネルに音を入力できる「入力」モードになります。
位置を調整する方法はLEDデータの調整を参照(98ページ)。

5 をクリックして下図のように音を選び、下にドラッグして移動する

マウスカーソルが↓になる位置におくと、ドラッグできます。



6 ▷をクリックする

正しく入力できたか、試聴してみましょう！

7 サウンドファイルを「jinglebells_2」で保存する

ウィンドウを閉じます。

素材を組み合わせよう

作成したファイルをAction Windowに追加して、素材を組み合わせましょう。

1 Action Windowを起動し、作った素材を追加する

モーション : raise_hands

サウンド : jinglebells_2

2 アクションファイルを「sing_jinglebells.act」という名前で保存する

すべてのウィンドウを閉じます。

このレッスンで作成したファイルは「L4_Reference」フォルダ内のファイルと同じです(12ページ参照)。ファイルが正しくできたか確かめてください。

アクションや素材の作り方は、これで理解できたと思います。
作り方さえ覚えてしまえば、あとはあなたのセンス次第です！

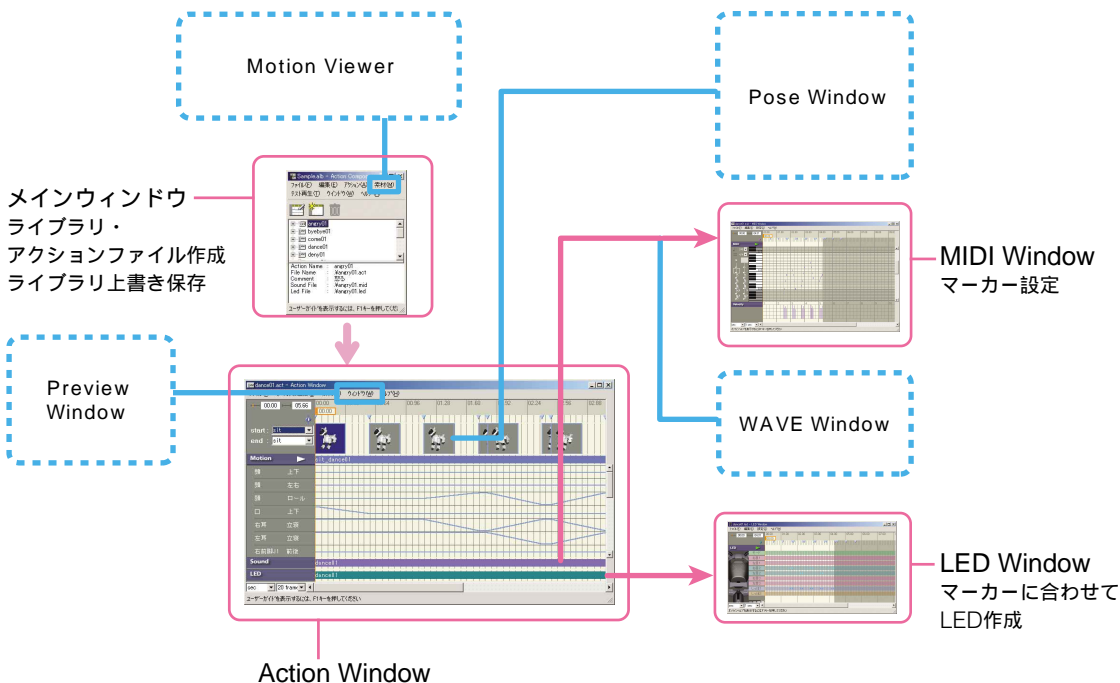
曲に合わせて、LEDを点灯させる！

目標：モーション・サウンド・LEDを同期させる方法を学習する

自分なりの素材を作れるようになったら、さらに一步踏み込んで、素材を同期させてみましょう。

例えば、ウィンクさせたり、おしゃべりに合わせてリップシンクさせたり。音楽に合わせて踊らせるのも楽しそうです。

Lesson5で使うウィンドウ



作業手順：

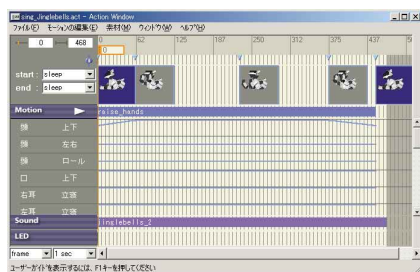
Action Windowから起動したMIDI Windowでマーカーを設定。設定したMIDIのマーカーに合わせてLEDを作成し、アクションファイルを保存。

素材を同期させるには、アクションのマーカ機能便利です。
ここでは、曲に合わせてLEDが光るようにしてみましょう。

1 Action Windowで、Lesson4で作った 「sing_jinglebells」アクションファイルを開く

2 メニューの[素材]-[LEDの追加]を選び、 「sync_sound.led」ファイルを選ぶ

このLEDファイルにはデータがないため、LEDチャンネルには何も表示されません。



3 サウンドとLEDの各編集ウィンドウを起動する

LEDの編集ウィンドウは、[素材]-[LEDの編集]で開きます。

Lesson5から始める方は
「L5_Starter」フォルダをコピーして
「tutorial5.act」を開いてください(12ページ参照)

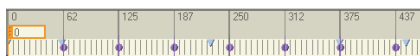
LEDとサウンドファイルは、Action Windowからは作れません。Materialsフォルダ内のあらかじめ用意されたファイルを使用してください。メインウィンドウから直接起動したMIDI、WAVE、LED Windowではマーカを設定できません。アクションウィンドウを起動してから、各素材の編集ウィンドウを起動してください。

4 MIDI Windowでマーカーを設定する

Altキーを押しながらマーカーチャンネルをクリックして、以下のフレームにマーカーを設定します。

各機種共通

62、125、187、250、312、375、437フレーム



MIDI Windowでつけた紫色のマーカーがAction WindowとLED Windowにも付いているのを確認しましょう。

間違えたときは...

Altキーを押しながら、マーカーをクリックすると削除できます。

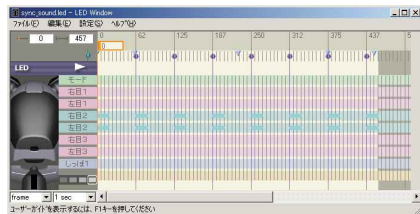
5 マーカーにあわせてLEDを作る

下の図のように20フレームの長さの点灯を繰り返すデータを作成します。

96ページ参照

ERS-210

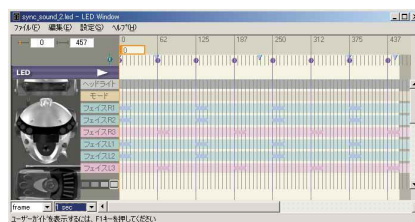
右目2と左目2を明るい輝度で点灯



ERS-220

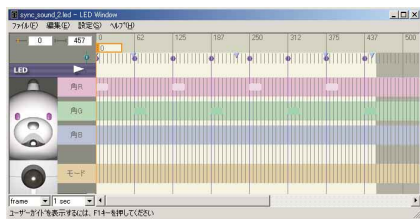
青と赤を交互に点灯

青：フェイスRL1 赤：フェイスCR3
フェイスRL2 テール中
テール左右



ERS-310シリーズ

赤と緑を交互に点灯



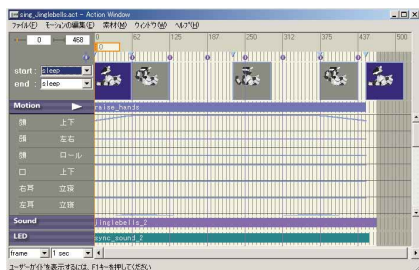
(時間幅：1 sec時)

6 LEDファイルを「sync_sound_2」として保存する

Action WindowでLEDファイルを「sync_sound_2」に置きかえます。

95ページ参照

ここで、▶でプレビューしましょう。



7 アクションファイルを「sing_jinglebells_2」という名前で保存する

次回、このアクションファイルを見ると、各ウィンドウにマーカー情報がついています。

同様に、LED Windowでマーカーを設定すると、LEDに合わせてキーフレームやサウンドデータの位置を調整するのに便利です。

このレッスンで作成したファイルは「L5_Reference」フォルダ内のファイルと同じです(12ページ参照)。ファイルが正しくできたか確かめてください。

これで、モーション・サウンド・LEDの同期がとれたアクションファイルが完成！ライブラリ機能を使うと、簡単に同期がはかれることが、お分かりいただけましたか？

次のレッスンでは、良い作品を作るコツをお教えしましょう。

これがアクション作りのコツだ！

さらに魅力的なAIBOのアクションを作るコツ、教えましょう！

よい作品を作るには、アクションコンポーザーを使いこなすことももちろんですが、最終的にはAIBOの特長を知ることと、個人のセンスが一番大切なのです。

ここでは、「いかにAIBOで効果的に見える作品を作るか」をテーマに、アクション作成のコツを紹介します。

全体として

モーション・サウンド・LEDは三位一体で！

- ・ モーション・LED・サウンドが一体になってこそ、味がでます。そっけないモーションにLEDとサウンドをつけただけで、きつとずっと味のあるアクションに見えるはずです。

リアルワールドで確認を！

- ・ AIBOはリアルワールドに存在するロボット。だから、PC内ではなく、AIBOで実際にアクションを確認するのが一番です。

モーション

重心をとるのがミソ！

- ・ PCでプレビューしながら作っても、PCの中ではAIBOの重心は計算されません。ですから、実際にはAIBOが転倒してしまうことも多いでしょう。例えば、右前脚を上げる前に体を左側に傾けておくと、安定します。サンプルから一部コピーしたり、ワイヤレスLANを使ってAIBOで試しながら作っていくことをお勧めします。
- ・ 伏せる姿勢(sleep)が一番安定しているので、モーションを作りやすいです。

例) 伏せる姿勢を使っのモーション

後ろに脚をもっていき、ばた脚で泳ぐモーションなども楽しいでしょう。

この場合、いったん後ろ脚を浮かせて肉球を地面から離してから、後ろに伸ばしましょう。後ろ脚をそのまま伸ばそうとしても、肉球が地面に引っかかって、うまく伸びません。

サンプルとして、「Materials」フォルダに「lesson6.mtn」が用意されていますので、活用してください。



- 3D上のAIBOの中心は、ボディの真中にあるので、実際のAIBOの動きとは少し違います。この違いをうまく捕らえるのが大切です。例えば、立った状態で脚先(肉球)の位置を固定させ、体をゆするようなモーションを作るときは、脚のJ1を左右対称の角度に設定し、スピードスケートのように脚を滑らせたモーションを作ります。すると実際のAIBOでは、体をゆするモーションになります。肉球の4点を動かさないよう、AIBOの3D表示を下から見ながら作るなど、工夫してください。

地面がない？

- 3D内には地面がありません。このため、3D内と実世界では体の向きが異なる場合があります。

また、3D内でも地面があることを想定してモーションを作しましょう。接地できないようなモーションでは、AIBOが転んでしまいます。

例)あお向けのモーションを作っても...

背中を下にして、4本の脚を上にも伸ばしたあお向けのモーションを作っても、背中側が地面とは限りません。実際には、あお向けではなく、4本脚で接地したモーションになるでしょう。

ぶつからない？

- パーツ同士がぶつからないかチェックする機能はありません。充分注意した上でモーションを作り、もし実際のAIBOでぶつかるようなことがあったら、すぐに動きを止め、AIBOをいたわりましょう。

姿勢やポーズは！

- 「これは」という決めのポーズを作ってから間のキーフレームを作っていくことが、大きなコツです。
耳やしっぽなどの細かい部分から作っていくと、あとが大変です。
- キーフレーム間はAIBOが自動的に線形補間しますので、思い通りの軌跡で動かすには、キーフレームで軌跡を指定していきます。

例)前脚をぐるっと回す

「右前脚をぐるっと上から回して後ろに移動する」というモーションの場合、途中にキーフレームを入れないと、AIBOは下から脚を回そうとするかもしれません。

3D上では動作可能に見えますが、実際には、地面やAIBOの他のパーツにぶつかってしまい、故障の原因になります。

使えるものは再利用しよう！

- 作ったポーズは流用しましょう。よく使うポーズやモーションをひとつのモーションファイルにまとめて「キーフレーム集」を作ると便利です。これをコピーして使いましょう。
- 立ったり、座ったりと、姿勢を変えるモーションを作るのは、難易度が高いです。姿勢変位は、AIBOにまかせましょう。どうしても作りたいときは、SampleライブラリのSit01やStand01のモーションをコピーして使ったり、参考にしてください。「座る」よりも重力に逆らう「立つ」方が難しいです。

キーフレームのコピーの方法は、ユーザーガイド(84ページ)参照。

動きに緩急を！

- 手や頭などを同じスピードで動かしたり、緩急をつけたりすると違う印象をうけます。同様に、モーション全体をゆっくりさせたり、速くさせるだけでも違う印象になります。サンプルモーションの速さを変えて使うのも、ちょっと違った雰囲気でもいいのでは？
- モーションをずっと動かしているのではなく、同じポーズのキーフレームを2枚置いて動きをいったん止めるのも緩急がでて面白いです。

スピードにご用心！

- AIBOの関節の稼動スピードは思ったより遅いです。ですから、すばやいモーションは作ることができませんので、ご注意ください。AIBOが対応できるスピードの限界を超えると、モーションチャンネルのグラフが赤くなって、あなたに「ムリムリ」と教えてくれます。その時は、補間フレーム数を変更したり、角度を小さくしたりして速度を調整してください。



モーションが速すぎる範囲

- 着地する直前にキーフレームを1つ作ると、モーションのスピードを変更でき、ゆっくり着地させることができます。速いスピードのまま着地させようとする、慣性も影響してAIBOのパーツが床などにぶつかってしまいます。AIBOをいたわるモーションを作りましょう。

首をいたわって！

- 首はAIBOのパーツの中でもデリケートな部分です。頭をぶんぶん振りまわすモーションは避けて、AIBOをいたわりましょう。

肉球は滑り止め！

- 肉球は滑り止めになるので、転ばないために肉球を地面に接地させるのが大切です。

しかし逆に考えると、地面にひっかかるということです。肉球が接地していると、地面を滑らせることができず、脚の付け根の関節に負担がかかってしまいます。脚を移動するときは、いったん脚を持ち上げて、肉球を地面から離しましょう。

AIBOを滑らせたいときは、前脚の前腕部分の裏を使うのもよいでしょう。

あなたの視線！

- モーションを作る視線と、実際にAIBOを動かして見るときの人間の視線は違います。イメージ通りのモーションかどうか、AIBOで確認することが大切です。

モーションの機種別情報

機種ごとのプラス点・マイナス点を活かして、あなたなりの味つけでAIBOの動きの世界観を出しましょう。

ERS-210/220

- 座っているときに、片手をあげるのは比較的簡単ですが、両手を上げるのは少し難しいかもしれません。後ろ脚を浮かせて開き、肉球を接地しないと難しいでしょう。
- ERS-210は、耳や尻尾をうまく使うと表情を出しやすいです。
- ERS-220は「垂直・水平の軌跡を多用する」「同じスピードでゆっくり動かす」などのモーションにすると、ロボットっぽくみえるでしょう。
- 頭の「ロール」は、かわいらしさや愛嬌を表現できます。ロボットっぽくモーションさせたい場合は、ロールを使わない方がいいでしょう。

ERS-310シリーズ

- 頭部が重いので、ERS-210/220よりも重心をとるのが難しいです。

例)前脚を上げる

立っている状態で右前脚を上げるには、左側の脚を深くまげるなど、重心を後ろに置きます。その際、頭を下げないなどの工夫も必要です。

座っている状態で前脚を上げる場合も、後ろ脚を開いて重心を移動してから、上げましょう。

両手を上げるのは、かなり難しいです。

- 首の動きがERS-210/220とは異なります。ERS-310シリーズの首の位置を変えるのは「頭 上下2」です。座っているときの首の位置もERS-210/220とは異なります。
- のんびりしたモーションを作ると、310シリーズらしく見えます。
- 脚や胴体が太いので、お互いのパーツがぶつかりやすいです。

サウンド WAVE編

ノーマライズを使おう

- 音が小さいと思ったらノーマライズ機能を使いましょう。録音した音をひずませずに、1番大きな音にできます。

サウンド MIDI編

テンポを変えて！

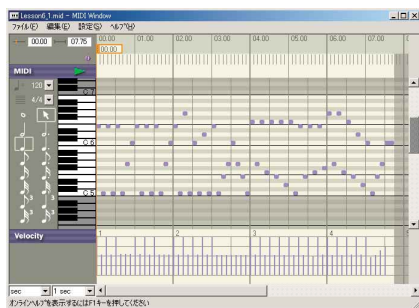
- 同じ曲でもテンポを変えるだけで、ゆっくりしたり、せかせかしたりという違った印象に変わります。

上級テクニック

ここでは、MIDIデータ作成の上級テクニックを公開。例を使って説明しましょう。

- 例1 主旋律の音を短くし、間に伴奏になる音を配置する

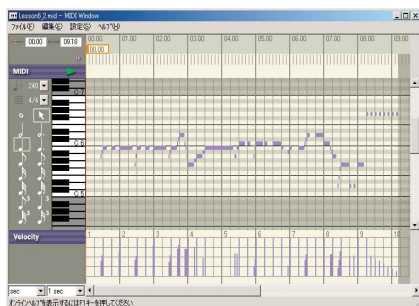
主旋律の音を短くして伴奏の音を入れると、ちょっと複雑な音楽に聞こえます。音を短くする代わりに、主旋律の音のペロシティを大きくし、伴奏音を小さくすると、主旋律がはっきり聞こえます。AIBOは単音しか出せないので、伴奏部分には和音をばらばらにした音を配置するといっそう雰囲気がでるかもしれません(分散和音)。



(Materials ¥ lesson6_1.mid)

• 例2 音の前に「飾り音」を入れると、違った印象に

近い音を使って飾り音を入れてみよう。飾り音が多いとうるさい・元気、少ないとのんびりした印象になるでしょう。



(Materials¥lesson6_2.mid)

LED

色を混ぜる

- ERS-210のしっぽランプ、ERS-310シリーズの角ランプなどは、多色LEDが使われています。それぞれのLEDを点灯させると、色がまざり、LEDそのものの色以外も表現できます。

輝度を利用する

- 4段階の輝度があるLEDは、「輝度を等間隔で明るくする」「徐々に明るくする」「急激に明るくする」など、明るくする方法によって、違った印象の光を作ることができます。ぜひ試してみてください！

輝度 4 4 4 4 3 3 3 2 2 2 1 1 1 2 2 2 3 3 3 4 4 4 規則的な印象

輝度 4 4 4 4 4 3 3 3 2 2 1 2 2 3 3 3 4 4 4 4 4 4 ぼわーんとした印象

輝度 4 3 3 2 2 2 1 1 1 1 2 2 2 3 3 4 激しい印象

音と一緒に！

- 音と同期させると効果的です。

たくさん光らせる

- たくさん光らせると、激しさを表現できます。ただし、逆にうっとおしくなる場合もあるので、ご用心！

以上で、アクションコンポーザーのレッスンは終了です。

素材やモーション作成のテクニックは学習できましたか？

オリジナルな素材やモーションをたくさん作って、AIBOと一緒に楽しく遊んでください。

プログラム作りが苦手な方は、アクションライブラリを作って“メモリースティック”に保存してからクイックビヘイビアアレンジャーを使うと、簡単にプログラムが作れます。アクション作りに力を注ぐのも、AIBOマスタースタジオのひとつの遊び方です！

Sampleフォルダに作品例としてExtraアクションライブラリが入っています。是非ご活用ください！

このチュートリアルガイドは再生紙を使用しています。

Printed in Japan



* 4 6 5 9 4 0 4 0 1 * (1)