



n:bit & micro:bit

二虎の巻二

Crib Notes



n:bit
ファミリー



NPO NEXTDAY 2019

このドキュメントは、
表示 - 非営利 (CC BY-NC 2.0 JP) に
基づきご利用ください。

- ・「NPO法人NEXTDAY」の
クレジット表示が必要です。
- ・営利目的での利用はできません。

v02-1911_03版

NPO法人NEXTDAY 2019

v02-1911_03版

キーワード

Keyword Index

- 解説編 -

n:bit

LED

7

1. 音をならそう
2. LEDを光らそう
3. センサーを音で聴こう
4. 3色LEDをシグナルメーターで使う
5. 電波を探そう：radio

n:bit

センサー & FET

23

1. 明るさをはかろう
2. 人をみつけよう
3. 音をならそう
4. 電気を制御しよう（モーター）
5. センサーで電気を制御しよう
- ・

n:bit

ばいす

37

1. micro:bitがおしゃべり
 - ・ 音声LSIとコミュニケーション
2. センサーの状態をしゃべらせよう
3. センサーの値を読ませよう

ma'ikurobi'xtuto ki'do_-si'masita.

- 活用編 -

工夫

40

- ・ 磁石をさがせ compass
- ・ 仲間を探せ radio
- ・ 離れてコミュニケーション sensor & radio
- ・ かくれんぼ～micro:bitみ～つけ！～ radio
- ・ 街灯の自動点滅
- ・ サウンドメーカー（音作り）

- 基礎編 -

はじめの一步

49

1. コンピュータってなんだろう
学校IT探検：センサーを探そう
2. プログラムとプログラミング教育

はじめてのmicro:bit

61

1. micro-bitってなに？
 2. 使って（動かして）みよう
 - ・ Step1～Step8
- 演習 「じゃんけんゲームをつくろう！」

TEAM IchigoJam
ほっかいどう

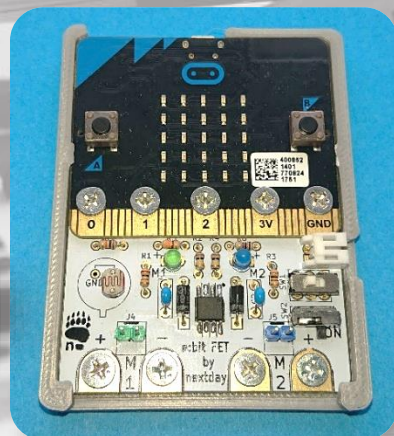
とっても便利で 学びが広がる！

micro:bit
拡張基板

n:bit ファミリー



n:bit
センサー & FET



n:bit
LED/スピーカー



n:bit
ぼいす



- ・基板上的センサー（光センサー or 人感センサー）と micro:bit で - 計測と制御 -
- ・プログラムで出力電流をスイッチのように FET で外部機器を On-Off 制御
モーターや LED を制御
同時に FET の On-Off 状態を基板上的 LED で確認
- ・FET 外部機器の電気を On-Off ができるスイッチ
- ・センサー 明るさを計測する光センサ (Cds) もしくは人などの接近を検知する人感センサ を装備

- みのむしクリップで簡単に接続
- ネジ止めで確実な接続

- ・スピーカー すぐにサウンド機能が使える
- ・電源スイッチ 使う時だけ ON 電池も節約
- ・LED 身近な信号機の動きを簡単に再現
センサー情報を LED で視覚的に表現
- ・ボタン電池 基板裏面に電池ホルダー付き
(入手しやすいボタン電池)
micro:bit に電源供給
持ち運べて、単体で便利に使える
- ・ネジ止め micro:bit と n:bit をネジ止めて
接触不良を防止

- ・ブロックにローマ字で文字を指定して micro:bit にプログラムを書き込むと、音声合成された声でおしゃべりしてくれる

プログラムによる制御例

- ◆micro:bit の A ボタン、B ボタンを押すと言葉を話すプログラム

A ボタン: 「こんにちは」

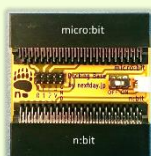
B ボタン: 「そだねー」

- ◆センサーで状態を計測して声で状況を話す
ボードを揺らすと「ゆれてる〜」
温度センサで計測: 「あたたかいです」
温度センサの数値を読み上げる
: 「いまのおんどは 25 ですよ」

シリアル通信 1行書き出す ma'ikurobi'xtuto ki'do-'si'masita.

お問合せ

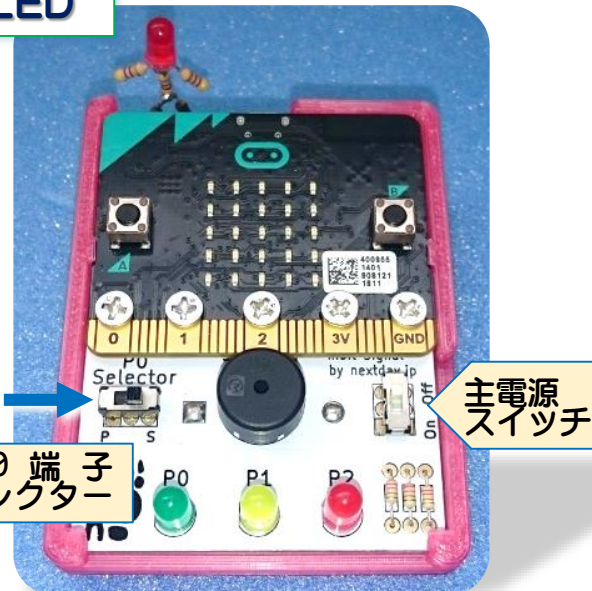
nextday@ict.skr.jp



n:bit Docking Base
- micro:bit 拡張ボード -

micro:bit と n:bit 基板を
ソケットで簡単接続・取り外し

- ・micro:bit の本来のサウンド機能を
n:bit ぼいすのスピーカーから流す



基本仕様

・電源スイッチ	3V電源端子（電池）からの電源を On/Off します。 電源スイッチがOffの場合は、micro:bit本体側から給電されます。
・P0 Selector	P0端子の入出力を選択するスイッチ P側： LED-緑 につながります S側： スピーカーにつながり、音を出せます
・LED	P0-緑 P1-黄 P2-赤 にLEDが接続しています。
・Power	micro:bitの3V/GND端子と3V電源端子に接続して、 外部機器に3V電源を供給します。
・ボタン電池 ソケット	3Vのボタン電池 CR-2032 を取り付けます。



概ね 6～10時間 稼働します。

※使用状況によって増減します

n:bit基板の使い方

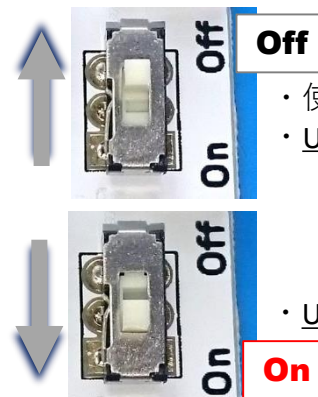
・P0端子の入出力を選択するスイッチ



p側： LED-緑 につながります

s側： 圧電スピーカーにつながり、音を出せます

・主電源スイッチ



- ・使わない時
- ・USBに接続して使う時

- ・USBに接続しないで使う時

注意） micro:bit本体側から給電（USB接続や電池接続）する場合は、必ず n:bit基板の**電源スイッチ**は **Off** にします。

※しまう時は ボタン電池 を外しましょう

センサー & FET

人感センサータイプ

センサー

焦電型
赤外線センサ
EKMC1601111

基本仕様

Micro:bitから2台の機器への電気の制御ができます。
1端子あたり最大500mAです。同時使用では700mA。

外部電池端子

主電源スイッチ

FET電源スイッチ

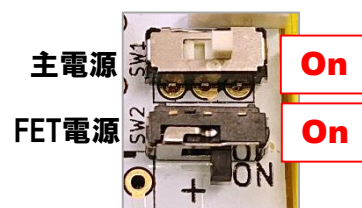
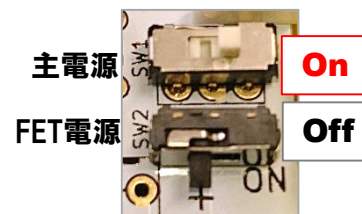
CdSセル
GL5528

光センサータイプ

センサー

M1 端子

M2 端子



基本的な使い方

・FET電源スイッチ SW2

Off

LED M1/2端子
点滅 切

- ・プログラムを書き込む時
- ・センサーとLEDのみ使う時

On

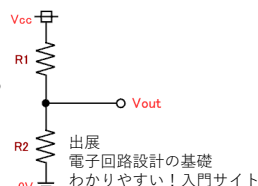
点滅 通電

- ・電池端子に外部電池を接続してモーター等を動かして使う時

注意）micro:bit本体側から給電（USB接続）する場合、M1M2端子のモーターは電圧不足で動かせません。必ず FET電源スイッチは Off にします。

分圧回路

センサーは、R1(右図)に取り付けられています。抵抗R1とR2によって電圧が分圧される仕組みを利用して、Vout(P0)の電圧を測定してセンサーの値を取得します。



・主電源スイッチ SW1

Off

・USBに接続して使う時

On

・USBに接続しないで使う時

注意）micro:bit本体側から給電（USB接続）する場合は、必ず n:bit基板の電源スイッチは Off にします。

・P1/P2 LED

P1およびP2をON-1-（通電）にした時にLEDが点灯します。USB接続時に、プログラムの動作確認（M1/M2端子への給電）やセンサー等のシグナル（合図・状態）用に利用できます。

・M1/M2 端子

ワニ口クリップやリード線で接続します。極性（±）があります。音の出力ポートをP1やP2に設定すると圧電スピーカーをつないで音（音楽）を鳴らすことができます

基本仕様

- ・音声合成LSI（ATP3012）を搭載
- ・micro:bitとシリアル通信で音声合成LSIを制御
ローマ字表記の文字列を送信すると、おしゃべりします。
- ・音（音楽）の出力を P2 に変更すると、
micro:bitのサウンド機能でスピーカーから音を鳴らせる。

起動時の設定

- ・通信先の設定 送信端子：P0
受信端子：P1
- ・通信速度 9600bps
- ・音を鳴らす端子：P2

音を鳴らす端子を P2 にする

設定後は約1秒（1000）のウェイトが必要

シリアル通信

通信先を変更する

送信端子 P0

受信端子 P1

通信速度 9600

スピーカー
ジャック

外部電池端子

主電源スイッチ

音声合成LSI
(ATP3012)

基本的な使い方

100円Shop
スピーカー

スピーカーを搭載していないため
別途、外部スピーカーが必要です。
100円ショップ等でご用意ください。

スピーカーの大きさ（直径）が大きくなると音量も増加します。
ただし電池の消耗も早くなります。
音量の調整はできません。

・主電源スイッチ

SW1

Off

・USBに接続して使う時

On

・USBに接続しないで使う時

注意）micro:bit本体側から給電（USB接続）する場合は、必ず n:bit基板の電源スイッチは Off にします。

シリアル通信 1行書き出す

ma'ikurobi'xtuto ki'do-_si'masita.

ma'ikurobi'xtuto ki'do-_si'masita.

マイクロビット 起動しました

データシート ATP3012XX

https://www.a-quest.com/archive/manual/atp3012_datasheet.pdf


1. 音をならそう Step1

1. 動き（動作）を考え **細かく分解**

プログラミングの手順

S側 にします

P0
Selector圧電サウンダー
から音が出ます・ **起動したら、一度だけ 音（音楽）** なる。

① 起動する



② 音（音楽）を選ぶ



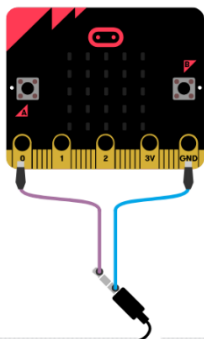
③ 1 回だけなる

2. ブロックを **探す**

ツールボックス

> **音楽**

メロディを開始する タタタム ▼ くり返し 一度だけ ▼

ほかの音楽や
音のブロックでも試してみよう！3. 順番に **並べる**

最初だけ

メロディを開始する パワーアップ くり返し 一度だけ

・ シミュレーターで確認 （再起動）



1. 音をならそう Step2

1. 動き（動作）を考え **細かく分解**

- ① 起動する
- ② 一度だけ、音（音楽）が鳴る
- ③ アイコンを表示

Step 1 を改良

「音楽が鳴ったら、**アイコンを表示する**」

2. ブロックを **探す**



ツールボックス

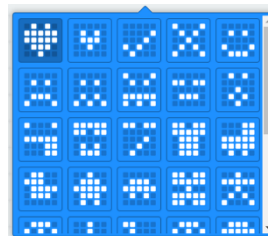
> **音楽**

メロディを開始する ダダダム ▾ くり返し 一度だけ ▾

ツールボックス

> **基本**

アイコンを表示 ▾



3. 順番に **並べる**



いろいろと改良してみよう！

- ・音とアイコンを逆にしてみる
- ・アイコンを変えてみる
- ・何度も繰り返して動作させる

2. LEDを光らそう

n:bit LEDタイプは、micro:bitの端子と次のように接続されています。

端子
色

P0
緑

P1
黄

P2
赤

それぞれの端子に電気を流すと点灯します。

※端子の値を 1 にする



P0
Selector



P0（緑）のLEDを使うときは、**P側**にします。

P0 P1 P2 端子のブロックは、高度なブロック > 入出力端子 にあります。



練習 緑のLEDを点けて消してみようみよう！

①ブロックを選ぶ



デジタルで出力するとは、端子に

- ・ 値 1 で 3.3v が流れ、
- ・ 値 0 で 電気が止まる (0V) ことです。

②端子を選択



③値を選択



●サンプルプログラム

ずっと

P0 に 電気が流れ LEDが点く

デジタルで出力する 端子 P0 値 1

デジタルで出力する 端子 P0 値 0

P0 に 電気が止まり LEDが消える

※キーボードで数値を直接入力することもできます

LEDがうまく点滅しなかった場合、もう一度初めから考えてみましょう。

??? LEDは思いどおりに点滅しましたか？

ブロック(命令)の順番は問題がないようです。

どのように工夫すると点いたり消えたりするのか考えてみよう！



LED(緑)の状態をじっくりと観察してみましょう。

重要な一ブロック が足りないようです。

練習 緑のLEDを点けて消してみようみよう！

1. 動き(動作)をもう一度 **細かく分解**



① 起動する



② LED(緑) に 電気が流れて点く



③



④ LED(緑) の 電気が止まって消える



⑤





一時停止

が足りなかったようです。



なぜ 必要なのか
理由を考えて説明してみよう！

ツールボックス

> 基本

一時停止 (ミリ秒)

100 ▼

100 : _____ 秒
_____ : 0.5秒
1000 : 1秒

5000 : _____ 秒
_____ : 1分
_____ : 30分

・ どこに 一時停止 ブロックを加えるといいか考えてみよう！

ずっと

デジタルで出力する 端子 P0 値 1

デジタルで出力する 端子 P0 値 0

チャレンジ



a. 1秒ごとに点滅させる

b. 1秒ごとに点滅させ、音も鳴らす

c. 信号機プログラムに挑戦！！

3. センサーを音で聴こう

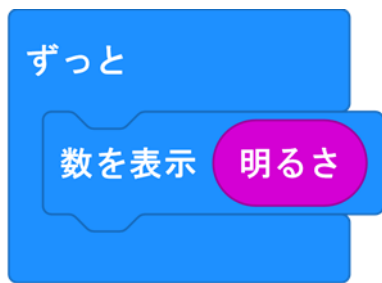
「明るさ」を音に変換！

micro:bitのLED(25個)は『明るさセンサー』として使えます。



S側 にします

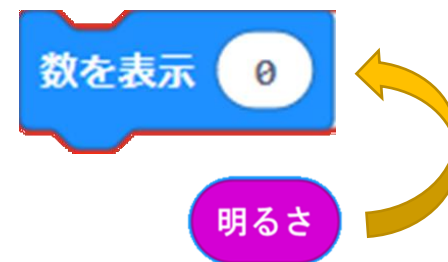
Step-1 明るさセンサーの値を見る（表示）



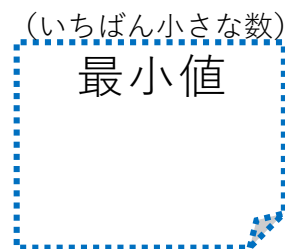
ツールボックス

> **基本**

> **入力**



Step-2 数値の変化を調べる

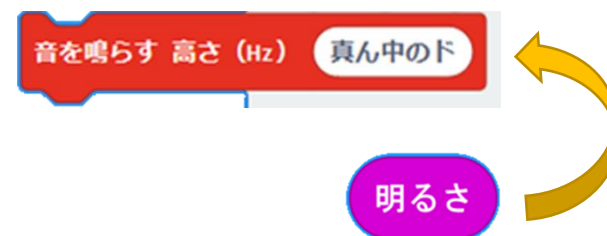


Step-3 明るさを音に変換



ツールボックス

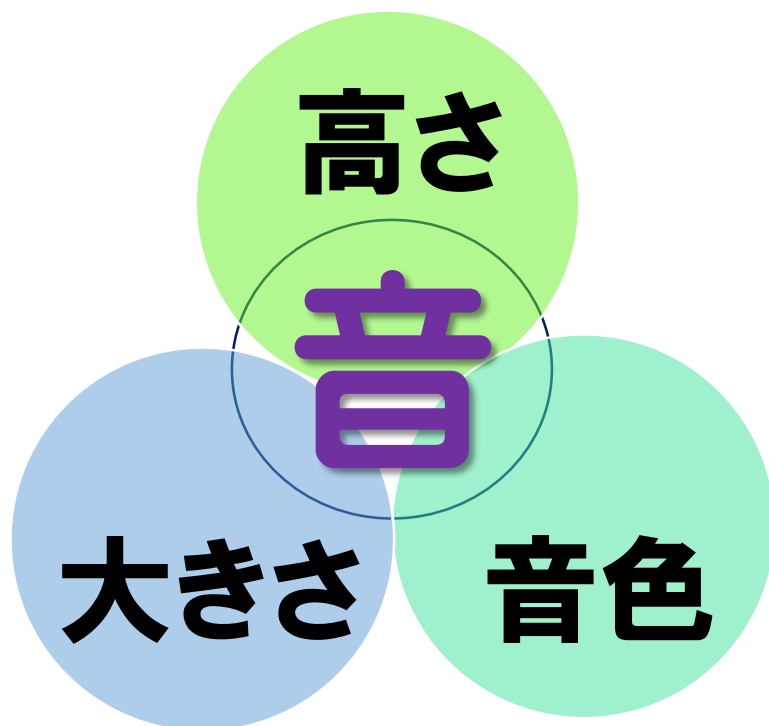
> **音楽**



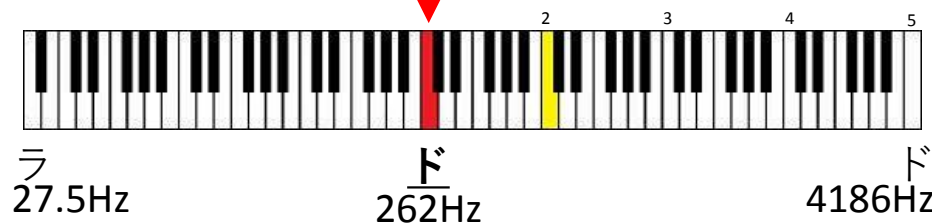
音と数値の関係

音の高さ = 周波数

(1秒間に振動する数)



- ・ 1秒間に約50回振動 = 50 Hz(ヘルツ)
- ・ ピアノの ドの音 = 262 Hz(ヘルツ)
[1秒間に262回の振動]



人間が聞こえる周波数

20 Hz ~ 20000 Hz (20kHz)



音の高さは数値で表すことができます。
センサーで測定した数値を、
人が聞こえる周波数（数値）の範囲にすることで、
音でセンサーの状態を聴くことができます。

Step1

磁力を数値で見よう ～数値で計測～

1. 動き（動作）を **細かく分解**
2. ブロックを **探す**
3. 順番に **並べる**

ずっと

磁力の変化を検知する

数値で表示

数値で表示

磁力の変化を検知する

ずっと

数を表示

磁力 (μT)

絶対値 ▼

ツールボックス

> 入力

磁力 (μT)

X ▼

✓ 絶対値

X

Y

Z

✓ 絶対値

> 基本

数を表示

0

Step2

磁力を音で聴いてみよう ～数値を音に変換～

1. 磁力を micro:bit で 聞くには？

2. 動き（動作）を
細かく分解する

3. 一つひとつの動きに
必要なブロックを探す

数字で見る

(数字で表示)



音で聴く



ツールボックス

> 音楽



4. ブロックを動きの **順番に** 並べる(組み合わせる)

①



②



③



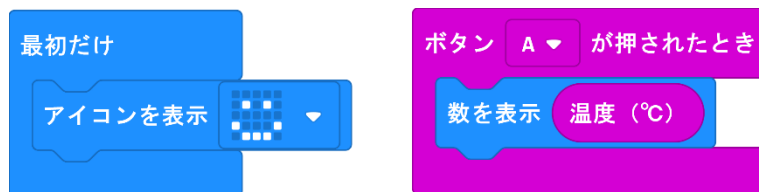
4. 3色LED を シグナルメーター で使おう

緑・黄・赤のLEDを、
温度センサーの値（大小）で点滅させて、
視覚的に現わすメーターとして利用してみよう。

設定条件 ・ **25℃未満**：緑
・ **27℃以下**：黄
・ **30℃以上**：赤

※設定温度は教室の環境に合わせて
変更しましょう

- ・ サンプルプログラム （温度計レベルメーター）



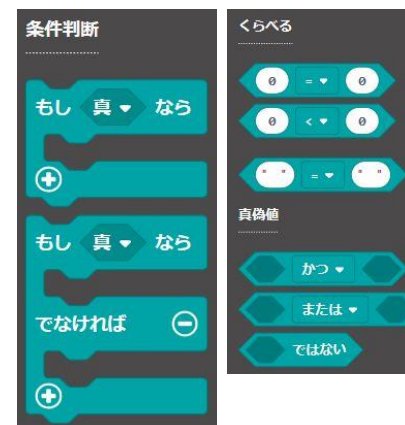
25℃未満なら緑LEDが点灯

27℃以下な黄LEDが点灯

30℃以上な赤LEDが点灯

ツールボックス

> 論理



**設定した温度で
指定のLEDが点灯しましたか？**
LEDの状態をじっくりと観察
してみましょう。 ヒント p12

5. 電波を探そう：radio

micro:bitには2つの無線通信機能があります。

無線通信 -radio-

CPUに組み込まれた無線通信-radio-を使います。
複数のmicro:bitと相互にデータの送受信や制御ができます。

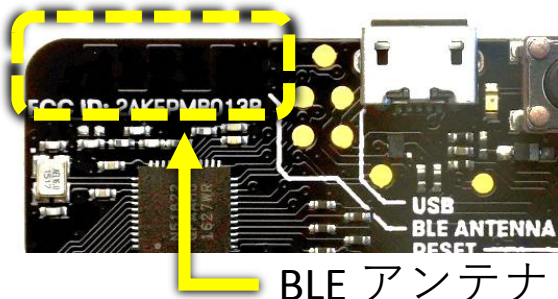


BLE

(Bluetooth Low Energy)

Bluetooth通信機能を使います。
低消費電力の特長を活かしIoT（モノのインターネット）機器
でとても多く使われています。

Web Bluetooth機能を利用して、micro:bitのセンサー値（温度
や加速度等）データをブラウザ上に表示することもできます。



※Web Bluetooth対応ブラウザ



<https://github.com/WebBluetoothCG/web-bluetooth/blob/gh-pages/implementation-status.md>

- ・送信側、受信側ともに、同じグループIDにします。
起動時に設定するだけで相互に通信ができます。



グループIDは、1～255 まで設定できます。

- ・サンプルプログラム



グループ番号を設定

Micro:bitの設定番号を確認できるようにします

起動をアイコン確認

起動を音で確認

実験ー 1 文字を送る : ボタンAを押すと、指定した文字を相手に送る

無線通信 -radio-

送信側

1



2



送信する
文字（英数字）を入力

チャレンジ



- ・ ボタンが押されたかどうかを、アイコンや音で確認できるように工夫してみよう！

ツールボックス

> 無線

受信側



recieveStringをマウスでドラッグするとブロックとして取り出せます



ツールボックス

> 基本



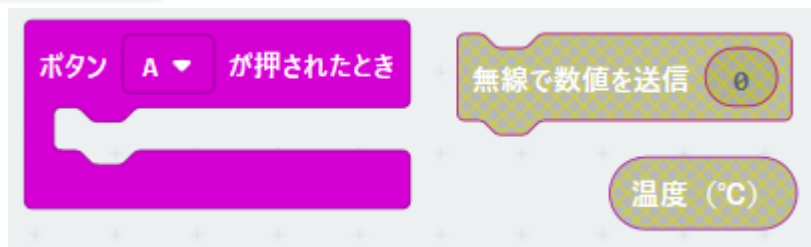
実験－２ 数値を送る : ボタンAを押すと、センサーの値を相手に送る

無線通信 -radio-

送信側

受信側

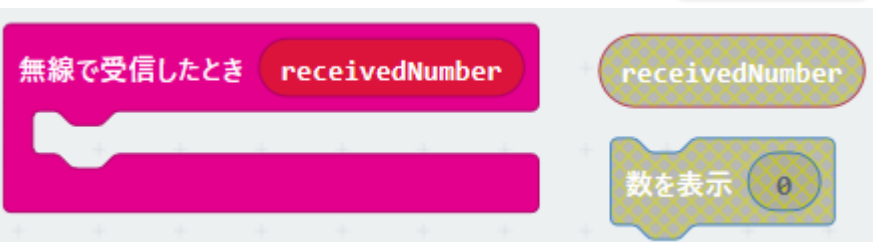
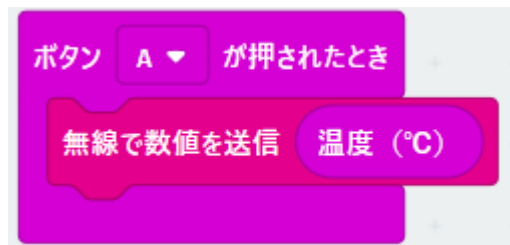
①



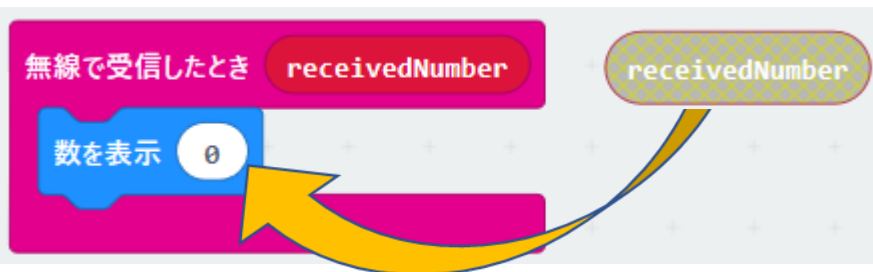
②



③



recieveNumberもマウスでドラッグして取り出します



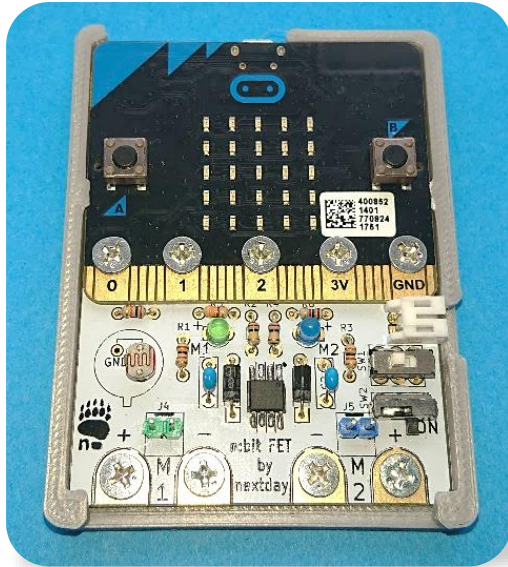
チャレンジ



- ・ ボタンが押されたかどうかを、アイコンや音で確認できるように工夫してみよう！

- ・ **Bボタン**で、送信側の**温度**も伝えてみよう！

センサー & FET



micro:bitで電池ボックスからの電気を制御 (On/Off)
 センサー (光・人感) で電気の流れを制御
 ・モーターカー ・照明

仕様

- ◆2chFET P1 と P2 に接続 一方向 (±逆転せず)
- ◆センサー P0 に接続 (分圧回路)
 - ・CdSセンサー ・人感センサー



CdSセル
GL5528

光センサー(CdS)版

センサーの値を アナログ数値 (0~1023) と
 デジタル数値 (0か1) で測定できます
 その測定値の変化に応じて、電気の制御に利用します



焦電型赤外線センサ
EKMC1601111

人感センサー版

センサーの値を デジタル数値 (0か1) で測定できます
 電気の入 (On) ・切 (Off) など、電気の制御に利用します

n:bit センサー & FET 1. 明るさをはかろう Step1

光センサーは、明るさによって抵抗の値が変化します。
特徴を利用して、センサー回路の電圧をmicro:bitが計測して表示します。

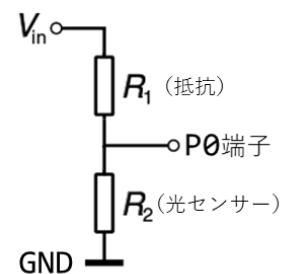
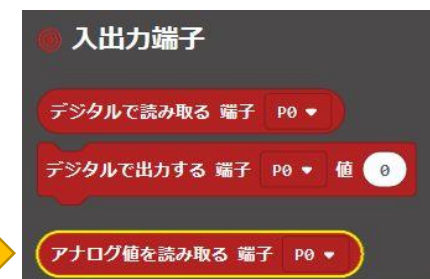


図 センサー回路

光センサーは、図のようにP0端子に接続されています。
P0端子からセンサーの値（アナログ/デジタル）を入力するブロックを使います。

P0端子のブロックは、
高度なブロック > 入出力端子 にあります。

ツールボックス
> 基本



光センサー（P0）の値を表示（読み取る）してみましょう。

約4秒に1回、LEDに3桁の数値で表示されます。

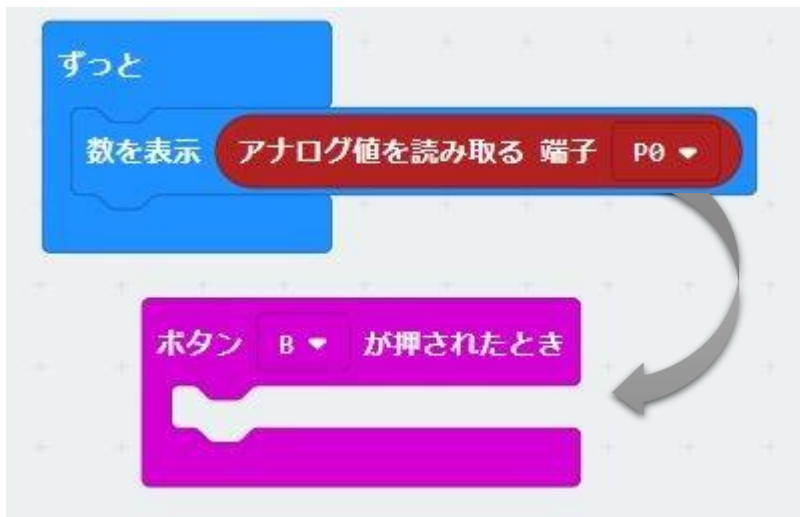
センサーを手でおおったり、部屋を暗くしたりして
表示される数値が変化することを確認してみましょう。

▲最大値（一番大きな数）：

▼最小値（一番小さな数）：

思いどおりに動作しない時など、センサーの状態を確認することにも利用できます。

ボタンを押した時だけ光センサーの値を表示します。



Bボタンを押したときだけ
光センサーの数値が表示されるように改良しましょう。

使うブロックは、
入力> ボタンAが押されたとき

便利アイテム

センサーを手で覆っても大きく変化しません。

そこで黒のマーカーキャップがオススメ！



チャレンジ



micro:bitの温度や傾きセンサー数値も表示してみよう。

ボタン(A・B・AB)によって表示するセンサーを変えるのも便利かも！

いろいろと工夫してみよう！！

暗くなるとLEDが点灯するプログラムをつくってみましょう。

- ・「暗くなる」と判断するセンサーの値を決める。
- ・センサー値400未満の場合に、右（P1）のLED（青）を点灯させる。

最初だけ

アイコンを表示

ボタン B が押されたとき

数を表示 アナログ値を読み取る 端子 P0



絶えずセンサー値を表示しているとP2端子への出力（0/1）が遅れます。そこで数値を確認したいときだけ、ボタンBを押して、センサー値を表示させるように工夫します。

ずっと

もし アナログ値を読み取る 端子 P0 < 400 なら

デジタルで出力する 端子 P2 値 1

でなければ

デジタルで出力する 端子 P2 値 0

アイコンを表示

400 「暗くなる」と判断するセンサー値の設定

400未満の場合

P2 = 1 >>> M2 (3V)

400以上

P2 = 0 >>> M2 (0V)

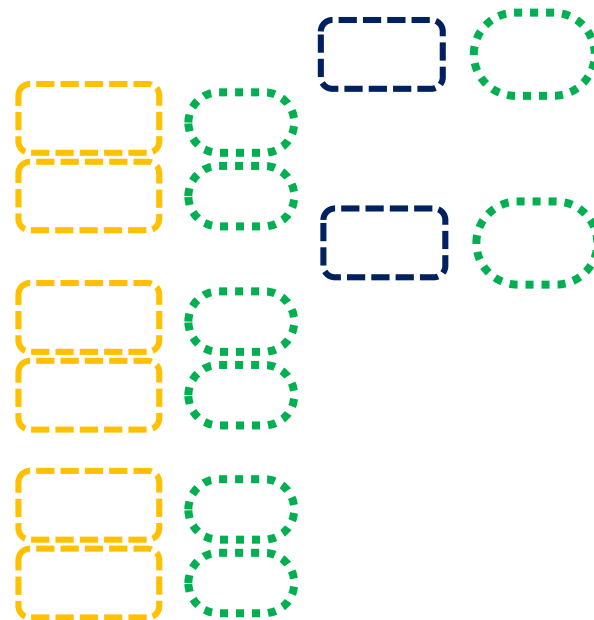
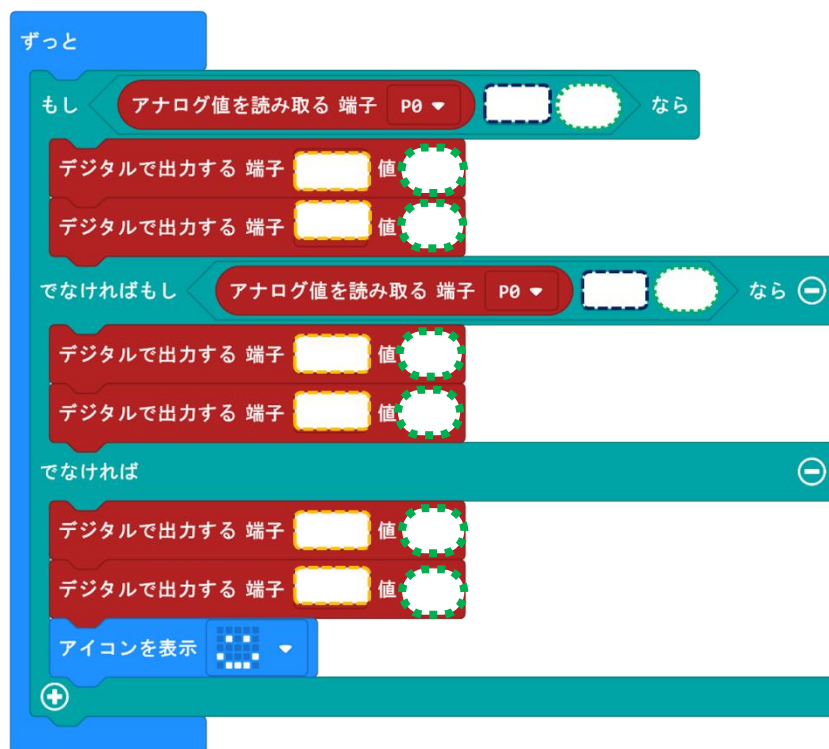
消えているときは、アイコン表示

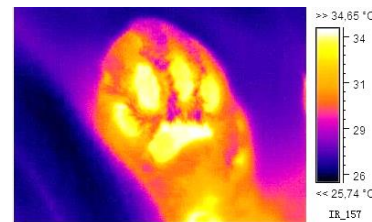
チャレンジ



明るさの変化で、左右のLEDを点滅させるプログラムをつくろう！

- ・ 明るさが400未満の場合、緑LED（P1）と青LED（P2）の両方が点灯
- ・ 明るさが700未満の場合、青LED（P2）のみが点灯
- ・ 明るさが上記以外の場合、両方のLEDは消灯





2. 人を見つけよう Step1

人感センサーは、人間が発する赤外線を検知します。
熱を発するモノは、すべて赤外線が出ているため、
人間以外の小動物・空調機器・太陽光などでも動作（検知）する場合があります。

ツールボックス>

高度なブロック> 入出力端子> その他



・人感センサーを使うための初期設定

最初だけ

アイコンを表示

< 起動確認用アイコン（無くても可）

端子 P0 のプルアップ・プルダウンを設定する なし

プルアップ
プルダウン
✓ なし



n:bitの人感センサーは、

・検出距離 **最大5m** ・検出範囲：約**90～130度** です。
そのため、近くに人がいると絶えず反応してしまうため、
プログラムの動作確認には注意が必要です。

例) ・動作前に紙コップ等でセンサーを覆う

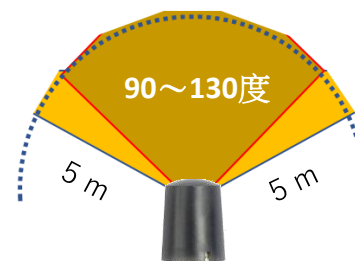
・センサーの視野を狭くするキャップ等を取り付ける

また、静止している人は検知しません。



ぶにゅグリップ

100円Shop 4 個入
半分にカットして
使えます



人感センサー

2. 人をみつけよう Step2

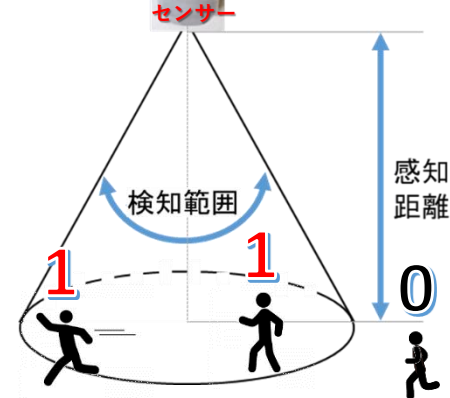
赤外線センサーは
次のように動作します

検 知

1

非検知

0



人を検知する

デジタルで読み取る 端子

P0

=

1

人を検知しない

デジタルで読み取る 端子

P0

=

0

ツールボックス

> 論理

くらべる

条件判断

サンプルプログラム

- センサーの値を表示
- もし 人を検知したなら -1-
緑のLED(P1)を点灯する。
(電気を流す)
- でなければ -0-、
緑のLED(P1)を消す。
(電気を流さない)

ずっと

数を表示

デジタルで読み取る 端子

P0

もし

デジタルで読み取る 端子

P0

=

1

なら

デジタルで出力する 端子

P1

値

1

一時停止 (ミリ秒)

500

でなければ

デジタルで出力する 端子

P1

値

0

一時停止 (ミリ秒)

500

+



一時停止ブロックは

なぜ必要なのでしょう？

理由を考えて説明してみよう！

3. 音をならそう

圧電スピーカー（ブザー）を、M1やM2端子に接続して、micro:bitの音を鳴らすこともできます。

・ サンプルプログラム

※圧電スピーカーは、M2端子につながります。



圧電スピーカー

振動板に直接電圧をかけて音を鳴らす仕組みです。
一般的なスピーカーと違って磁石やコーンはありません。

音の出力端子の指定



標準ではP0が出力端子です。
P2に変更する場合

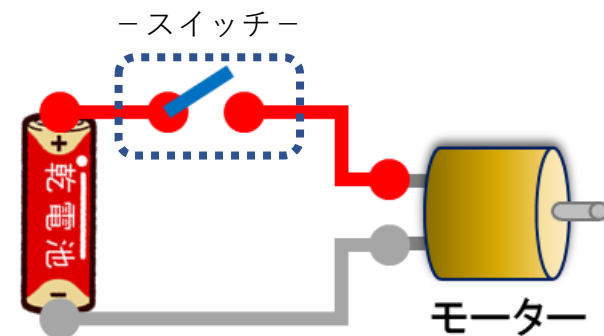


n:bitでは3つの端子しか使えません。
P0をセンサー入力、P1をモーターなどの電気制御、P2を音の出力として使うなどの工夫ができます。

4. 電気を制御しよう（モーター）

モーターへの電気を制御（動く/止める）するには、スイッチを使います。

そのスイッチの役目をn:bitのトランジスタが行い、micro:bitのプログラムで命令します。



電 子 的 な
ー ス イ ッ チ ー



主電源

Off

Off

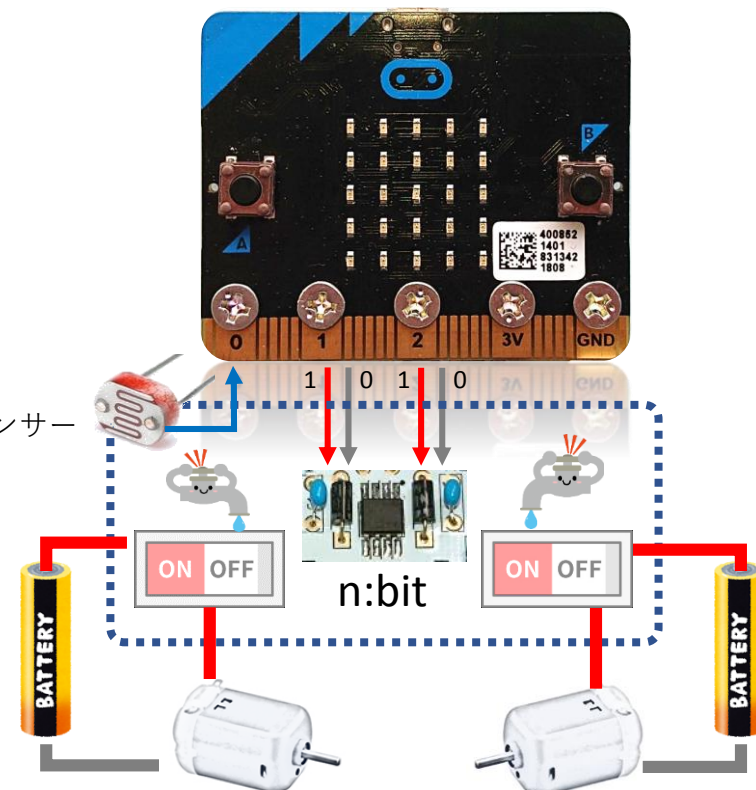
FET電源

注意

USBでパソコンに接続しているときは、
・主電源スイッチと、
・FET電源スイッチは、
切ります。

M1端子にみのむしクリップ等でモーターを接続しておきましょう。

光センサー



次のように動作するプログラムをつくります。

M1端子にモーターを接続

- ・ **ボタンA**を押したらモーターが**回転**します。
- ・ **ボタンB**を押したらモーターが**停止**します。

起動の確認



ボタンAを押したらモーターが回転

ボタンBを押したらモーターが停止

チャレンジ



M1端子とM2端子にそれぞれモーターを接続
・ボタンAを押したらM1のモーターが回転し、
・ボタンBを押したらM2のモーターが回転します。

次のようなプログラムをつくりましたが、
回転したモーターが止まらず困ってしまいました。



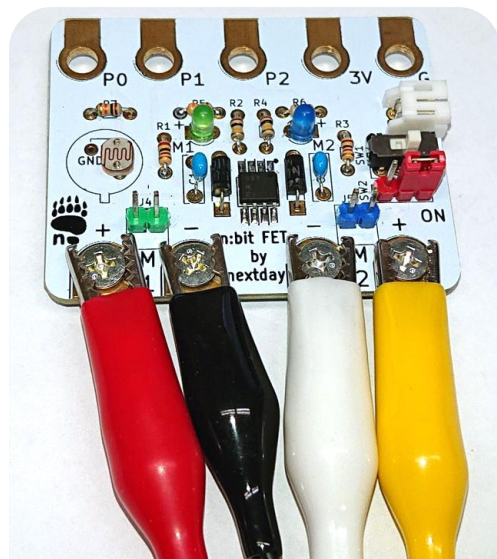
どのようにして止めたら良いか考えてプログラムを改良しましょう。

- ・ ボタン B を押したときに、M 1 側のモーターを止めるには？
その逆の場合は？
- ・ 両方のモーターを同時に止めるには？

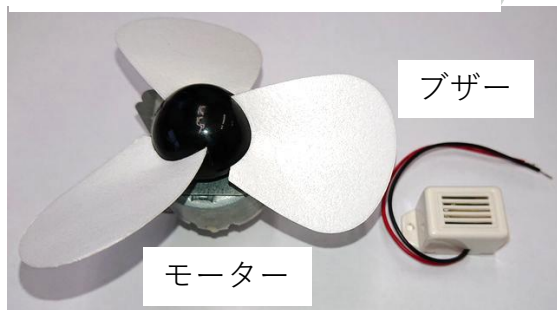
5. センサーで電気を制御しよう

1. 明るさをはかろう Step3 のサンプルプログラムを改良して

明るさによってモーターを動かしてみよう。



ワニ口クリップを使った接続例



モーターを動かすほかに、
ブザーも鳴らすことができます。
(注意：3Vで動作する機器)

P1端子やP2端子からのデジタル信号で
n:bit (トランジスタ) がスイッチの役目をして、
電池 (3V) からの電気をON/OFFします。

たとえば、
P1端子が 1 (ON) になると
M1端子に3Vの電気が流れます。

P1端子が 0 (OFF) になると
M1端子の電気が止まります。

M1端子たM2端子に 3Vで動作する
機器を接続して、プログラムで制御できます。

身の回りで、明るさによって動作 (点滅・動く・止まる)
する機器をさがしてみよう。



注意 USBでパソコンに接続
しているときは、
主電源スイッチと、
FET電源スイッチは、
切ります。

街灯を模したサンプルプログラム

暗くなったらM1/M2両方に電気が流れ、薄暗いとM1のみ、
明るいとは両方の電気が止まります。



ボタン A を押したら明るさを表示

暗い（300未満）で両端子に電気が流れる

薄暗い（600未満）で
M1端子に電気が流れる

明るい（600以上）で
両端子の電気が止まる

暗い・薄暗い・明るいの数値設定は
必要に応じて変更しましょう

2. 人をみつけよう Step2 のサンプルプログラムを改良して

人を検知したらモーターが動くプログラムに改良しよう。



P0端子の数値を表示 -----

人を検知した場合 P0=1 -----

- ・ M1端子に電気を流す P1=1
- ・ アイコン（ハート）を表示
- ・

人を検知しない場合 P0=0 P0<>1 --

- ・ M1端子の電気を止める P1=0
- ・ アイコン（スマイル）を表示

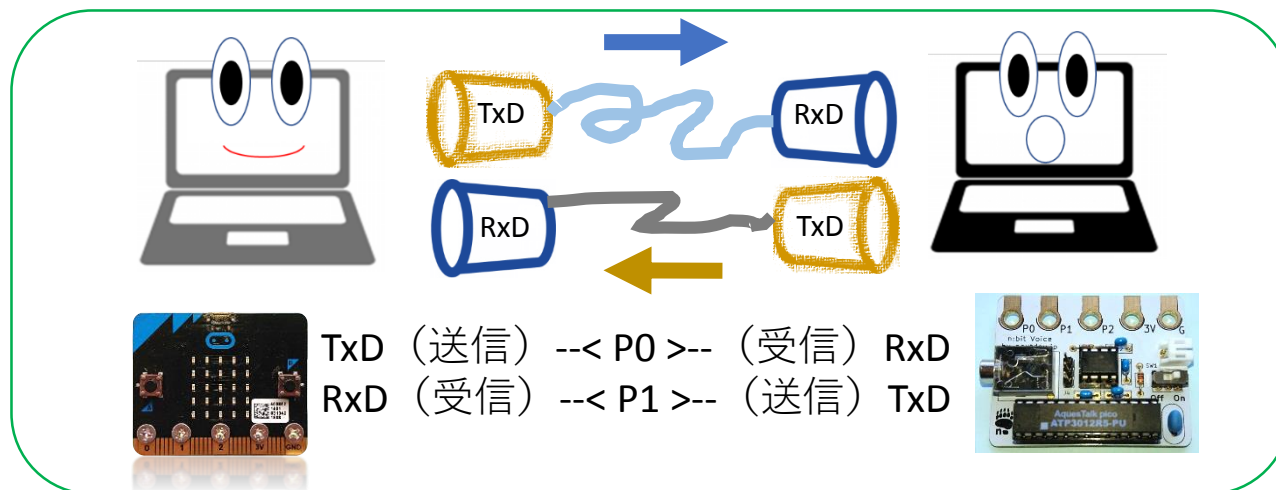
1. micro:bitがおしゃべり ・ 音声LSIとコミュニケーション

micro:bitとn:bit **ぼ・い・す**（音声LSI）と相互に通信して、micro:bitが言葉を話すようにプログラムできます。

その通信制御には、シリアル通信という仕組みを利用します。

TxD : Transmitted Data

RxD : Received Data



micro:bitがn:bit **ぼ・い・す**と通信（コミュニケーション）するには設定が必要です。

ツールボックス

高度なブロック > シリアル通信

- ・ P0端子でmicro:bitから音声LSIにデータを送る
- ・ P1端子で音声LSIからデータを受ける
- ・ 通信速度 9600bps（1秒間に送受信するデータ量）

※通信速度が早すぎたり違うと、データの送受信ができません

最初だけ

シリアル通信

通信先を変更する

送信端子 P0

受信端子 P1

通信速度 9600

一時停止（ミリ秒）

micro:bitからn:bit **ぼ・い・す**に設定情報を伝えます。

伝えるのに約1秒が必要です。

1000

しゃべる言葉は、このブロックの中に記入します。

ツールボックス

高度なブロック > シリアル通信

シリアル通信 1行書き出す

" "

" "

に、記述方法を参考に入力します。

最初だけ

・ 起動したときにしゃべるサンプルプログラム

シリアル通信

通信先を変更する

送信端子

P0

受信端子

P1

通信速度

9600

一時停止（ミリ秒）

1000 ▼

アイコンを表示



シリアル通信 1行書き出す

ma'ikurobi'xtuto ki'do-_si'masita.

マイクロビット 起動しました
ma'ikurobi'xtuto ki'do-_si'masita.

◆ 音声記号列の記述方法

音声記号列とは、生成する音声を文字コードで表現したものです。音声記号列はATP3012の入力データとなります。
ATP3012は単語辞書を持っていませんので、漢字を読むことができません。
したがって、基本的にはローマ字の文字列で表現することになります。

抜粋 ATP3012XX データシートp18

これは、音声記号 です。 * * は、
korewa onse-ki'go-de_su. * * wa

音声	記号	言う	さっぽろ	っ
おんせい	きごう	いう	Sa'xtuporo	xtu
onse-	ki'go-	yuu		

そだねー。
sodane-. * * を、
so'dane-. soda'ne- * * o

アクセント記号 ' ,

アクセント記号は、アクセント核の読み記号の後ろに付与すると定義していますが、音の高さが「高→低」に変化する部分にアクセント記号をつけると考えるとよいと思います。最初の例の「これわ」の部分に見られるように、アクセントが平板で「高→低」に変化しない場合にはアクセント記号はつけません。

抜粋 ATP3012XX データシートp18

クラブ
ku'rabu

準備
じゅんび
jyu'nbi

今度は
こんどは
ko'ndow

もう少し複雑な音声記号です。
そうすこし ふくざつな おんせいきごう です。
mo-suko'si/fukuzatuna/onse-ki'go-de_su.

句切記号

- ・ ポーズ(無音区間)が入る。文の終わりを示す。
- ? ポーズ(無音区間)が入る。文の終わりを示す。文末の 声が高めになる。
- (半角空白) ポーズ(無音区間)が入る。文中の息継ぎの部分に指定。一般に次の音が高くなる。
- , 短いポーズ(無音区間)が入る。半角空白 と機能は同じだが、無音区間の長さが短くなる。
- ; 次のアクセント句が比較的高い音で始まる。ポーズは入らない。
- / 通常のアクセント句の句切に指定。ポーズは入らない。
- + 前後のアクセント句の句切があいまいな場合に指定。ポーズは入らない。
- 無声化を表す。直後の文字が声帯の振動を伴わずに発生される音となる。「su」が無声化

・ サンプルプログラム A



・ サンプルプログラム B



4. 3色LEDでレベルメーター

・サンプルプログラム

最初だけ

アイコンを表示



ボタン A が押されたとき

数を表示 温度 (°C)

・温度を区分する設定条件の変更

・25°C未満 ・25°C以上30°C未満 ・30°C以上

・それぞれの条件ごとに命令の追加

設定した条件のときだけ、指定のLEDを点灯させ、他は消す命令が必要です。

ずっと

もし 温度 (°C) < 25 なら

デジタルで出力する 端子 P0 値 1

P0 (緑LED) に電気を流す

デジタルで出力する 端子 P1 値 0

P1 (黄LED) の電気を止める

デジタルで出力する 端子 P2 値 0

P2 (赤LED) の電気を止める

25°C未満なら緑LEDが点灯

でなければもし 温度 (°C) ≥ 25 かつ 温度 (°C) < 30 なら

デジタルで出力する 端子 P0 値 0

P0 (緑LED) の電気を止める

デジタルで出力する 端子 P1 値 1

P1 (黄LED) に電気を流す

デジタルで出力する 端子 P2 値 0

P2 (赤LED) の電気を止める

25°C以上 30°C未満なら
黄LEDが点灯

でなければもし 温度 (°C) ≥ 30 なら

デジタルで出力する 端子 P0 値 0

P0 (緑LED) の電気を止める

デジタルで出力する 端子 P1 値 0

P1 (黄LED) の電気を止める

デジタルで出力する 端子 P2 値 1

P2 (赤LED) に電気を流す

30°C以上なら赤LEDが点灯

・これ以外の温度設定も
考えてみよう！

・ サンプルプログラム



最初だけ

アイコンを表示



ボタン B が押されたとき

数を表示

アナログ値を読み取る 端子 P0

ずっと

もし

アナログ値を読み取る 端子 P0

<

400

なら

デジタルで出力する 端子 P1

値

1

デジタルで出力する 端子 P2

値

1

でなければもし

アナログ値を読み取る 端子 P0

<

700

なら

−

デジタルで出力する 端子 P1

値

0

デジタルで出力する 端子 P2

値

1

でなければ

−

デジタルで出力する 端子 P1

値

0

デジタルで出力する 端子 P2

値

0

アイコンを表示



+

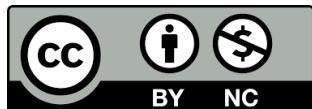
- ・ 明るさが400未満：緑LED（P1）と青LED（P2）の両方が点灯
- ・ 明るさが700未満：青LED（P2）のみが点灯
- ・ 明るさが上記以外：両方のLEDは消灯、アイコンの表示

< ! 注 意 ! >

- ・この教材は、保護者等の適切な指導のもとでの利用が必要です。
- ・部品の取り付けや配線を間違えると、部品等の破損・発熱・発煙・発火等の恐れがあります。
- ・金属や配線がむき出しです。

取り扱いを誤るとショートして、発火・感電・ケガの恐れがあります。

- ・水に濡らさないでください。
- ・部品が熱を持ったり、異臭がしたら、直ちに使用をやめてください。
- ・部品の取り付けや配線は、必ず電源を切って行ってください。
- ・教材を使用しない時は、必ず電池を取り外して保管してください。



このドキュメントは、
表示 - 非営利 (CC BY-NC 2.0 JP) に基づきご利用ください。

- ・「NPO法人NEXTDAY」のクレジット表示が必要です。
- ・営利目的での利用はできません。

詳細は以下をご覧ください。

<https://creativecommons.org/licenses/by-nc/2.1/jp/>

いまをつくる！



make
think
create

n:bit TM

— micro:bit拡張ボード —

NPO法人 **NEXTDAY** は
子供たちの学びを支援しています

お問い合わせは nextday@ict.skr.jp

未来を創る！

子どもたちに **創る** + **I**nformation **T**echnology & **C**ommunication **Collaboration** **楽しさを！**