



# 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。

# 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

### ぼいす

..... 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！」



# 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

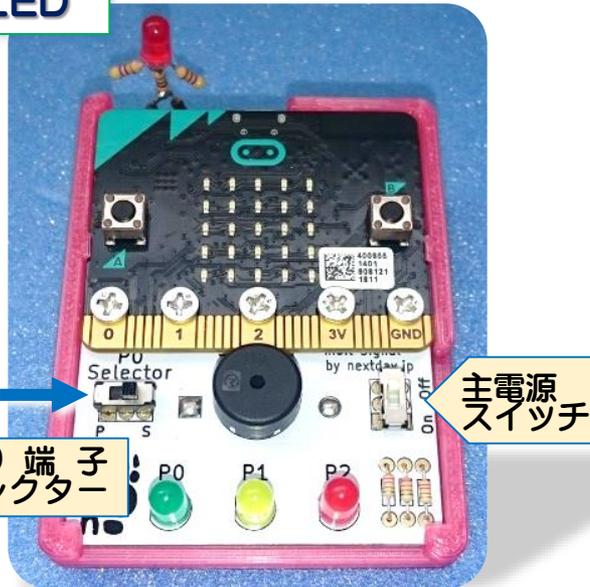
- ◆micro:bit の A ボタン、B ボタンを押すと言葉を話すプログラム  
A ボタン: 「こんにちは」  
B ボタン: 「そだねー」
- ◆センサーで状態を計測して声で状況を話す  
ボードを揺らすと「ゆれてる〜」  
温度センサで計測: 「あたたかいです」  
温度センサの数値を読み上げる  
: 「いまのおんどは 25 ですよ」

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

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

- ・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

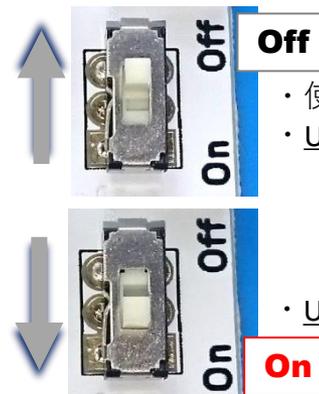
## ・P0端子の入出力を選択するスイッチ



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

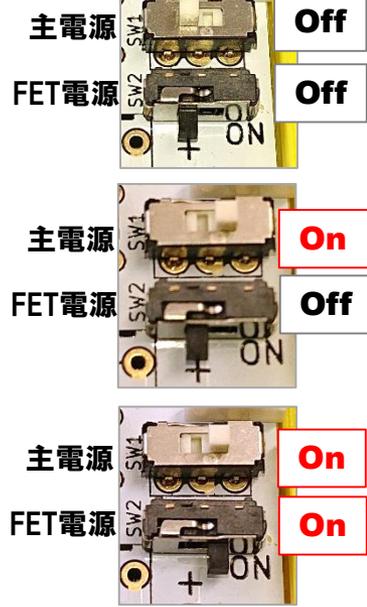


CdSセル  
GL5528

## 光センサータイプ



センサー



# 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

・主電源スイッチ SW1

Off

- ・USBに接続して使う時

On

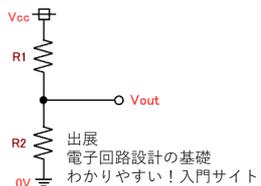
- ・USBに接続しないで使う時

**注意** micro:bit本体側から給電(USB接続)する場合、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。

**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。

・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。



・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

ワニロ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

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

## 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。**

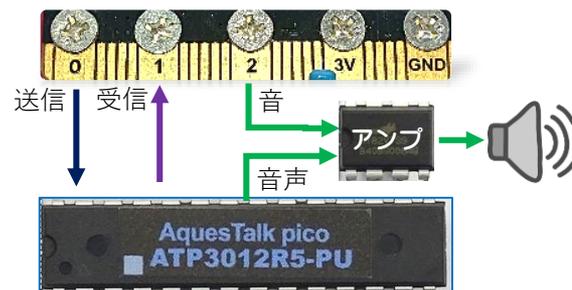
シリアル通信 1行書き出す

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



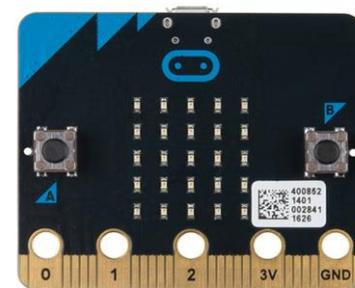
# 1. micro-bitってなに？

## 特徴

### ・ブロック型とスクリプト型（JavaScript/Python）

プログラムを作成する方法は2種類あります。

小学生はブロック型で、中学生はブロック型とJavaScriptの併用、高校生はPythonと同じハードウェアを、学ぶ内容に合わせて適切な言語を選択した活用ができます。



## シュミレーター

作成したプログラムを本体にダウンロード（転送）しなくてもシュミレーターを使ってテストできます。プログラムが思い通りに動作しているのか、正しく動作しているかなどを確認できます。思いついたアイデアを試すことにも使えます。

## ユニークなセンサーを内臓

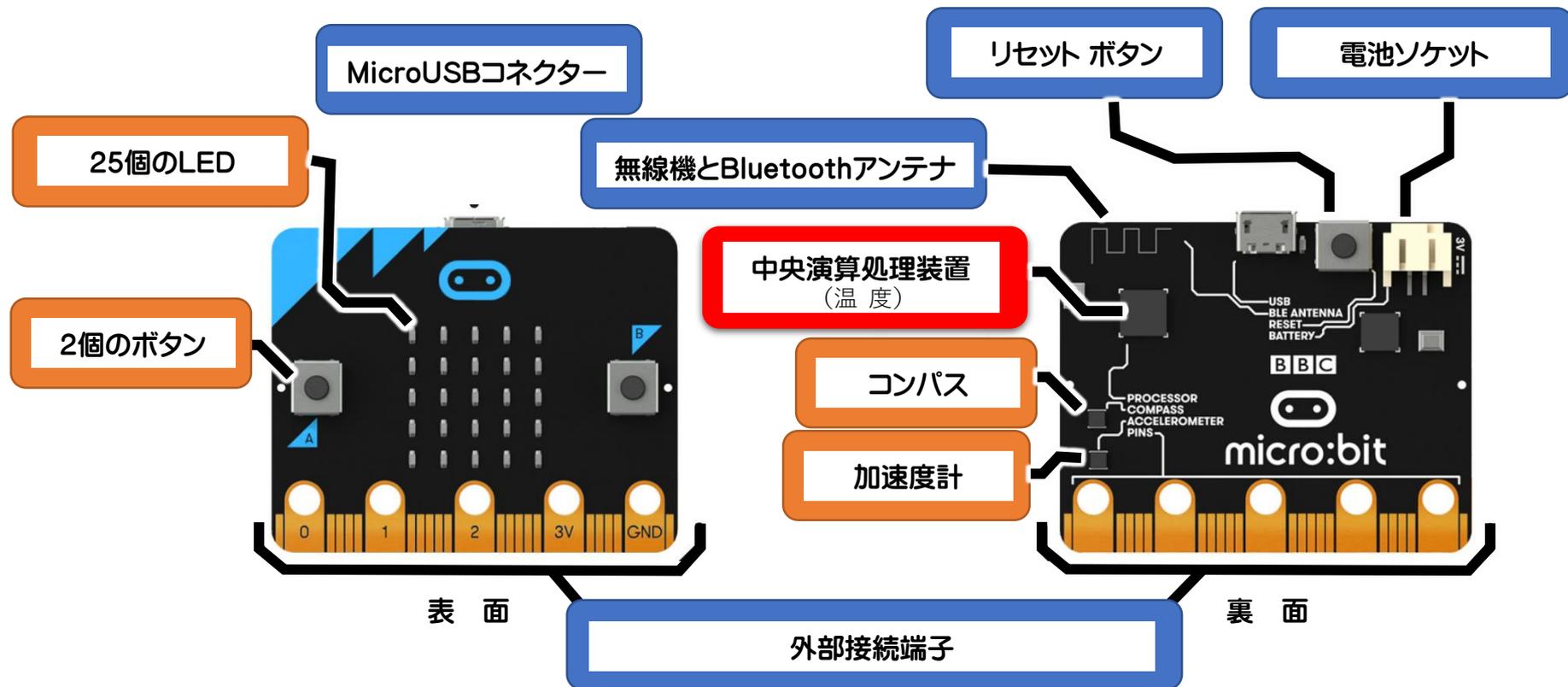
押しボタン、光センサー温度センサー、加速度センサー、コンパス（磁気センサー）などのデータ入力機能を標準で装備しています。それぞれに専用ブロックが用意され簡単に活用することができます。

## 多彩な活用（創造性）

入力と出力の機能がこの1台の中に組み込まれ、外部機器との接続も容易です。micro:bitの特徴を学ぶことで様々なアイデア（発想）をプログラムを使って表現することができます。

## 通信機能

複数のmicro:bitで無線通信ができます。情報（センサーデータ）を送受信でき、コミュニケーションや操作したりできます。



## ◆電源供給

供給方法は3つあり、それぞれの方法によって外部機器が使用できる電流に制限があります。

- ① microUSBコネクタ           : 90mA
- ②電池ソケット                : 3A未満 (供給側の制限)
- ③外部接続端子 (3V/GND)   : 3A未満 (供給側の制限)

## ◆センサ

- ・光センサ
- ・温度センサ
- ・動きセンサ   **加速度計 / コンパス**
- ・ボタン

# はじめてのmicro:bit

## ・プログラムをつくるアプリ

### JavaScriptブロックエディタ

> ブラウザ版

: インターネット接続が必須

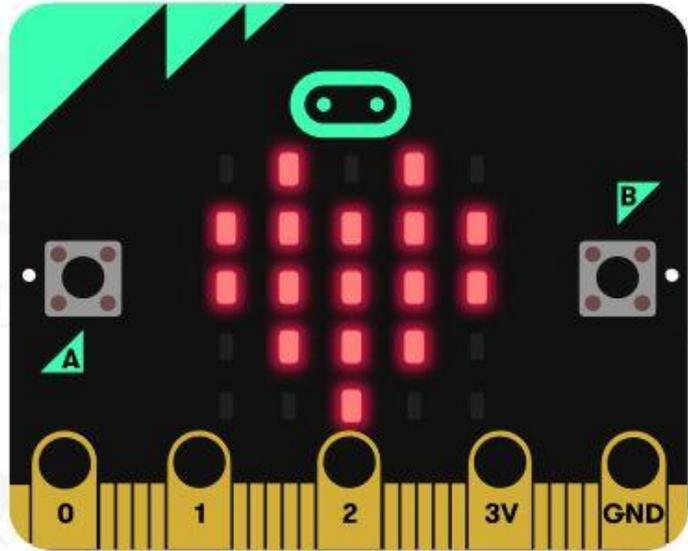
> Windows10アプリ版

: インター接続がなくても使える



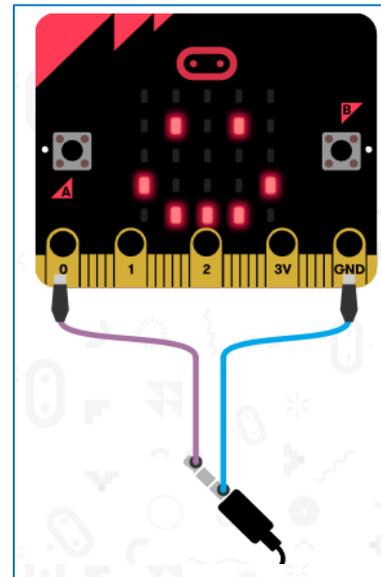
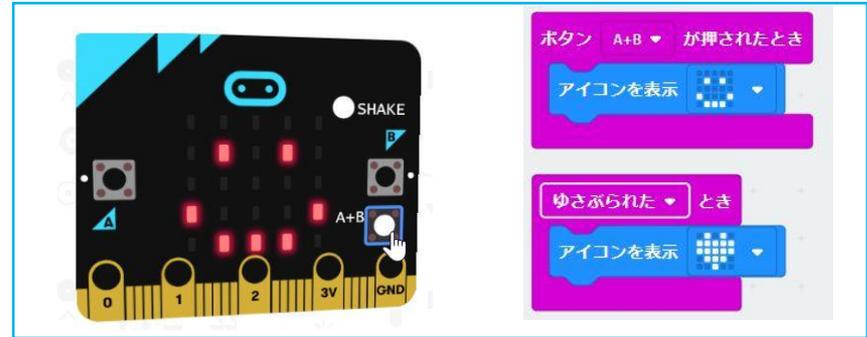
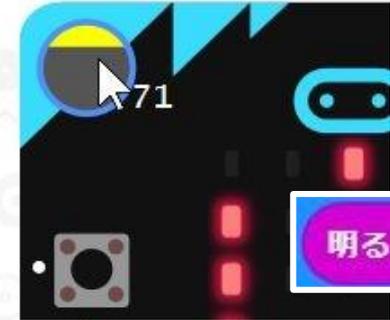
# はじめてのmicro:bit

## ・シミュレーター



micro:bit本体がなくてもプログラムを動作させて確認することができます。使用した命令（ブロック）によって自動的に変化します。マウスで操作したり、センサーの数値を変化させることができます。

マウスで温度や明るさを可変



↑マウスでボタンを押せる

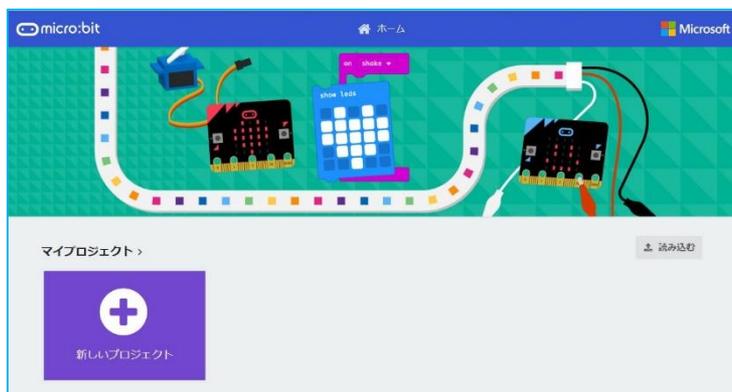


↑マウスでアナログ値を可変

パソコンから音が出る

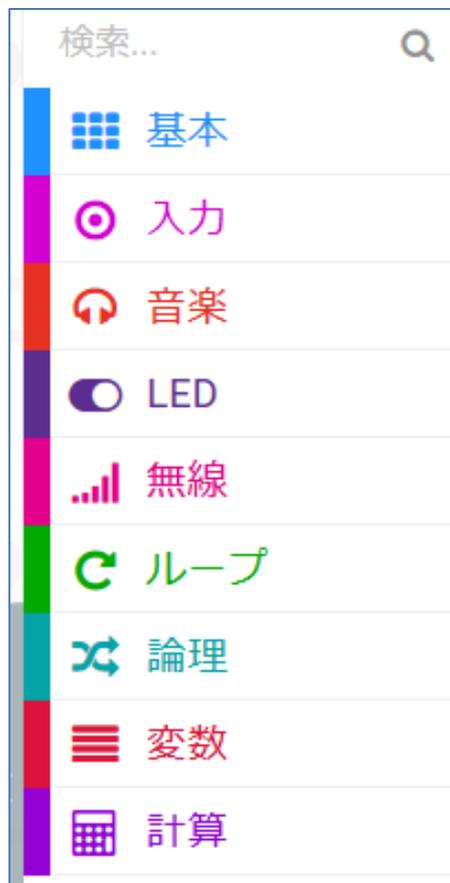
## 2. 使って(動かして)みよう

- ・ 新規にプログラミングする時  
①新しいプロジェクト

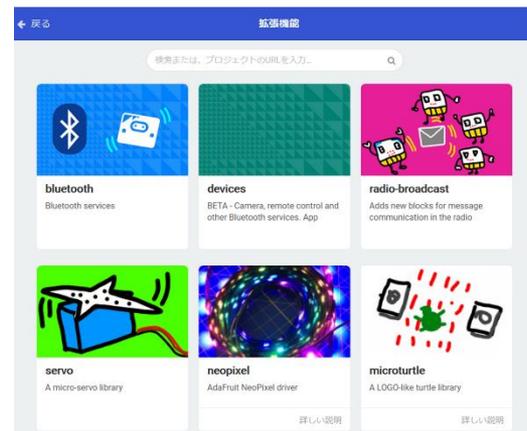


## ・ツールボックス

プログラムブロックが機能別に整理されています。  
それぞれ色でも判別しやすくなっています。



## ◆ 拡張機能



専用の外部機器を使う場合、  
専用のブロックを取り込んで  
使うことができます。

# ◆プログラミングの7ステップ

1. どのように動かす（表示）するかを考える
2. 動き（動作）を細かく分解して、手順を書く
3. 一つひとつの動きに関連するブロックを見つける
4. ブロックを並べる、組み合わせる
5. シミュレーターで動作を確認する
6. プログラムを名前を付けて保存する
7. **micro:bit**に転送する

# Step1

## 文字や数字を表示しよう

- ・25個のLEDに「Hello!」を表示してみよう
- ・0~10までの数字をランダムに表示してみよう

### ツールボックス > 基本



### ツールボックス > 計算



## Step2

# アイコンを表示しよう

起動したら25個のLEDに「ハート」と「小さいハート」アイコンが交互に表示するプログラムを作ろう

ツールボックス > 基本

The screenshot shows a programming environment with a toolbox on the left and a workspace on the right. The toolbox is titled '基本' (Basic) and contains several categories: '基本' (Basic), 'その他' (Other), '入力' (Input), '音楽' (Music), 'LED', '無線' (Wireless), 'ループ' (Loop), '論理' (Logic), and '変数' (Variables). The '基本' category is selected, showing a search bar and a list of blocks. The workspace contains a program with the following blocks: a 'ずっと' (Forever) loop block, a '数を表示' (Show Number) block with the value '0', an 'LED画面に表示' (Show on LED Screen) block, and two 'アイコンを表示' (Show Icon) blocks. The second 'アイコンを表示' block is highlighted, and a dropdown menu is open, showing a grid of icons. The '小さいハート' (Small Heart) icon is selected, and a tooltip with the text '小さいハート' is visible over it.

# Step3

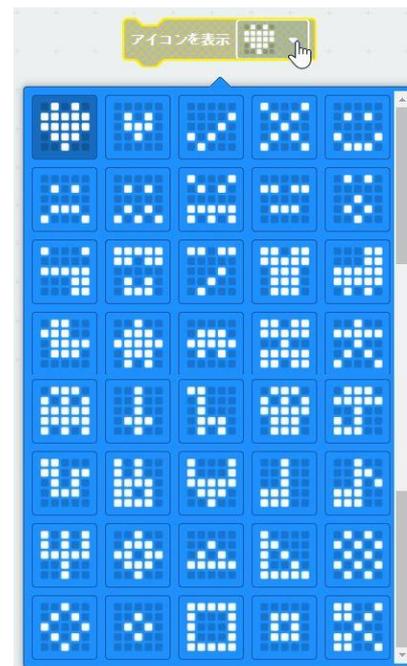
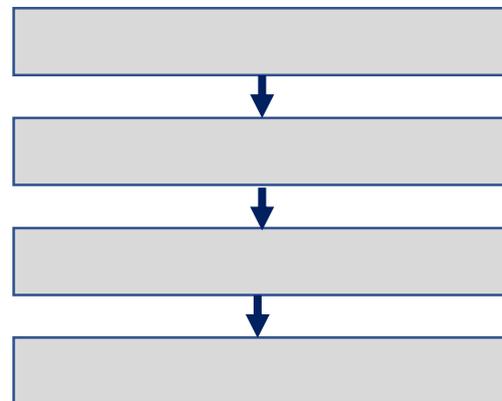
## LED ペースメーカー

25個のLEDにハートマークを表示して、  
1秒ごとに点滅するプログラムを作ろう

ツールボックス > **基本** にあるツールでつくります



動き（動作）を分解



音も鳴るプログラムに改良しよう！

n:bit LEDが必要です



ツールボックス  
> **音楽**



# Step4

## ボタンAB と 文字

ボタンAとボタンBを押したときにそれぞれ異なる文字が表示するプログラムを作ろう

ツールボックス > 入力

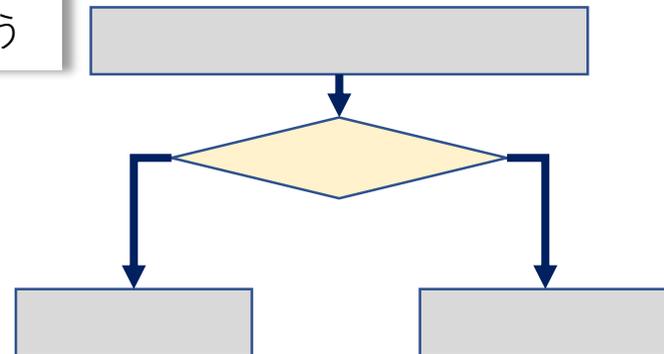


ツールボックス > 基本



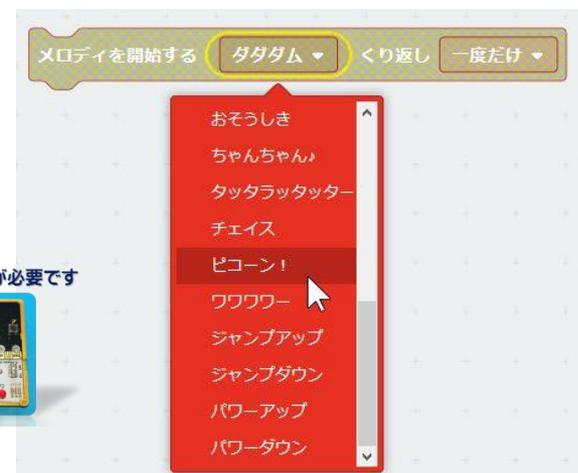
例) A を押したら 「A」と表示  
B を押したら 「B」と表示

動き（動作）を分解



どちらのボタンを押したかが音でも判別できるプログラムに改良しよう！

n:bit LEDが必要です



例) Aを押したら 「ピコーン」と鳴る

# Step5

## 信号機をつくらう

緑・黄・赤のLEDを点滅させて  
信号機プログラムを作ろう

-- 信号機 --

P0  
緑色点灯

P1  
黄色点灯

P2  
赤色点灯

n:bit LEDが必要です



P0 (緑) のLEDを  
使うときは、P側  
にします。

端子 色  
P0 : 緑  
P1 : 黄  
P2 : 赤

高度なブロック

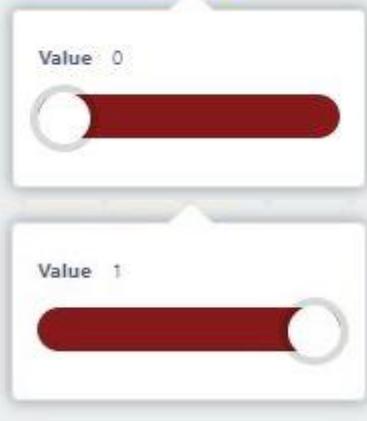
入出力端子

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



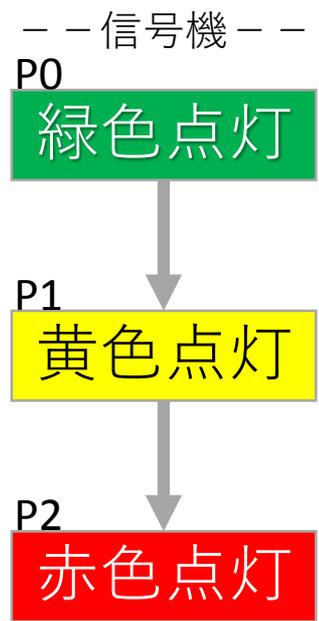
デジタルで出力する 端子 P0 ▼ 値 0



# Step5

## 信号機をつくらう

緑・黄・赤のLEDを点滅させて  
信号機プログラムを作ろう



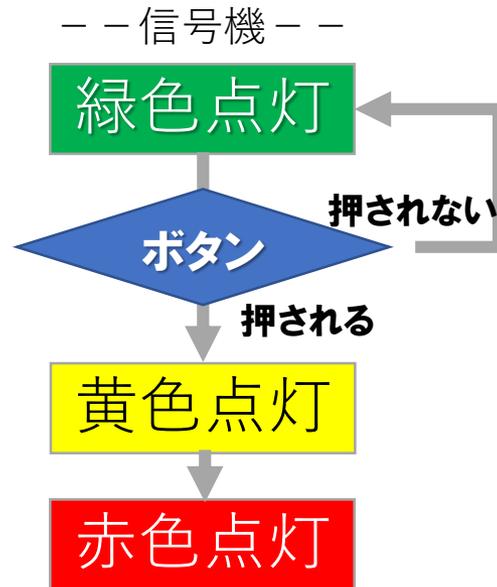


◆機能を組み合わせてみよう

## Step6

### 押しボタン信号機をつくろう

Step-5を改良して、  
押しボタン信号機に改良しよう！！



### ツールボックス > 入力



### ツールボックス > 論理



# Step 7

## 動きセンサー 加速度計



# G

## 加速度の単位

重力と同じ大きさの力が加わる状況のことを 1G  
1秒間にどれだけ速度が増すか

- 1Gの場合「1秒で約4.9メートル」
- 2Gの場合「1秒で約9.8メートル」
- 3Gの場合「1秒で約14.7メートル」

? スペースシャトル

自由落下における距離と時間と速さの関係  
<https://mathwords.net/jiyurakka>

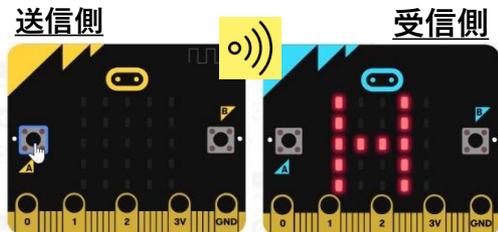
月の重力が地球の1/6になる理由  
<https://mathwords.net/tukijuryoku>

宇宙速度  
<https://mathwords.net/utyusokudo>

# Step8

## micro:bitでコミュニケーション

2台のmicro:bitで情報（データ）をやり取りします。  
文字・センサーなどの情報を双方向で伝えてみましょう。



### 文字

#### 送る側

```

when button A pressed
  send string "Hello!"
  
```

#### 共通

```

when first started
  set wireless group 11
  show number 11
  
```

#### 受ける側

```

when wireless received
  show string receivedString
  
```

グループを  
同じ番号にします。  
グループ番号を表示



receivedString をドラッグして  
文字列を表示に入れる

### 数値

#### 送る側

```

when button A pressed
  send number 99
  
```

#### 共通

```

when first started
  set wireless group 55
  show number 55
  
```

#### 受ける側

```

when wireless received
  show number receivedNumber
  
```

```

show brightness
show temperature
  
```

明るさや温度も  
伝えてみよう！

グループを  
同じ番号にします。  
グループ番号を表示



receivedNumber をドラッグして  
数字を表示に入れる

# じゃんけん

Let's play rock-paper-scissors

## ゲーム



# ゲームの設計

- どうやって \_\_\_\_\_ ?
- \_\_\_\_\_ を決める
- micro:bitの \_\_\_\_\_ をどう使うか?

Step1~7で使ったおもなブロック



# じゃんけんゲームの設計

- どうやってじゃんけんするのか？
- **ルール**を決める
- micro:bitの**機能**をどう使うか？

• 数字でじゃんけん → 多い方が勝ち

• 1, 2, 3 の 数字 を使う

• 数字はランダムに決まる

• 振ってじゃんけんする

• ボタンを押してリセット

ツールボックス > 計算



ツールボックス > 基本



ツールボックス > 入力



サンプル - 1 (数字でじゃんけん)

最初だけ

アイコンを表示 

ずっと

ゆさぶられた  とき

数を表示  から  までの乱数

ボタン  が押されたとき

数を表示 

サンプル - 1b (効果音をつけてみよう)

ゆさぶられた  とき

数を表示  から  までの乱数

メロディを開始する   くり返し 

ボタン  が押されたとき

数を表示 

メロディを開始する   くり返し 

## サンプル - 3 (変数を使ってみよう)

じゃんけんゲーム

最初だけ

アイコンを表示

変数 じゃんけん を 0 にする

ずっと

ゆさぶられたとき

変数 じゃんけん を 1 から 3 までの乱数 にする

数を表示 じゃんけん

ボタン A+B が押されたとき

変数 じゃんけん を 0 にする

数を表示 じゃんけん

## サンプル - 4

ゆさぶられたとき

変数 じゃんけん を 1 から 3 までの乱数 にする

メロディを開始する ピコーン! くり返し 一度だけ

数を表示 じゃんけん

ボタン A+B が押されたとき

変数 じゃんけん を 0 にする

数を表示 じゃんけん

メロディを開始する ジャンプアップ くり返し 一度だけ

## Step-1 文字を表示しよう



## Step-2 アイコンを表示しよう



## Step-3 LED ペースメーカー



## Step-4 ボタンAB と 文字



## Step-5 信号機をつくろう



## Step-6 押しボタン信号機をつくろう



# < ! 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™

— 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](mailto:nextday@ict.skr.jp)

# 未来を創る！

子どもたちに **創る** + Information Technology & Communication Collaboration **楽しさを！**