# いいいい 「サンプル プログラミング集」 (第4.0版) 【ブロック・microPython・Pythonプログラミング】

大分県立芸術文化短期大学 情報コミュニケーション学科 非常勤講師 望月陽一郎 mochizuki@oita-pjc.ac.jp 作品のクレジットを表示 すること・営利目的での 利用をしないこと・元の 作品を改変しないこと (作者の許可を得ている 場合は除きます)

Created by ©Yoichiro.Mochizuki 2018-2021



1





<u>目次</u>

001 温度を表示する 026 ひらがな表示 027 傾き(ピッチ)角度を表示 002 明るさを表示する 028 明るさにより発信音の高さを変える 003 ボタンで温度を表示する 004 ボタンで明るさを表示する 029 太陽高度計 005 ボタンで明るさ・温度を表示す 030 ふりこ送信 る 031 ふりこ送信 段階1 006 暗くなるとLEDが点く 032 ふりこ送信 段階2 007 Happy Birthdayを鳴らす 033 文字列を送信 008 電子オルゴールを鳴らす 034 接近したら点灯(距離センサー) 009 テスターで音を鳴らす 035 スイッチを制御(制御スイッチ) 010 明るさで音が変わるテルミン 036 リモコン・スイッチで点灯 (制御スイッ 011 ストップウォッチ チ) 012 明るさで秒数計測 037 人感センサー (距離センサー・制御ス イッチ) 013 コンパス (Compass) 038 距離を表示(距離センサー) 014 2台で送受信 039 明るさグラフ 015 水準器 040 明るさグラフリアルタイム 016 傾きメーター 041 距離を測定(距離センサー) 017 方位磁針 042 micro:bitファームウェアアップデート 018 方位磁針16方位 043 人感センサー 019 方位磁針16方位program分 044 ランダム数値表示 割 020 暗くなったことを検知 045 じゃんけん 021 相互にストップウォッチ 046 じゃんけん無線判定 022 通信で合奏 047 磁気表示 023 明るさ変化を送信 048 磁力計 049 検流(ペルチェ素子) 024 揺れたら送信 025 明るさ数値送信 050 検流グラフ(ペルチェ素子)

Created by ©Yoichiro.Mochizuki 2018-2021



# <u>目 次</u>

05 I	計算機
052	計算機(関数)
053	気温を小数表示
054	湿度を小数表示
055	気圧を小数表示
056	気温・湿度・気圧を小数表示
057	簡易熱中症計
058	気温数値を送信
059	気温数値を受信
060	気温・湿度を同時送信
061	気温・湿度を同時受信
[v2 #	編】
001	重ねて音を鳴らす
002	音がしたらビックリ
003	タッチしたらビックリ
004	音のしきい値
005	静かになったら
006	音量グラフ
007	音量を送信
008	音量を受信
009	タッチする楽器
010	メロディを打楽器に

【microPython编】 001 「HelloWorld」と「ハートマー ク」を表示する 002 LED上に「O」を表示する 003 傾きを表示する 004 0から9までを表示する 【Pythonプログラミング編】 関数など 001 温度を表示する 002 明るさを表示する 003 ボタンで温度を表示する 004 ボタンで明るさを表示する 条件(if文) 006 暗くなるとLEDが点く 音 007 Happy Birthdayを鳴らす 複数の条件 013 コンパス (Compass) 無線 014 2台で送受信 傾き 015 水準器 016 傾きメーター 乱数と繰り返し 044 ランダム数値表示 045 じゃんけん



# micro:bit プログラミングサイト

https://makecode.microbit.org/#editor





【説明】 micro:bit プログラミングサイト <u>https://makecode.microbit.org/#editor</u> にアクセスしてプログラミングします。 ①ブロックを組み合わせてプログラミング。 ②確認用シミュレータで確認する。 ③USB接続したmicro:bit にプログラムをコピー。 ④micro:bit 本体で動作させる。 (プログラムをPCにダウンロードして保存すること もできます。)

※ダウンロード保存しておいたプログラムを読み込 んで修正することもできます。



ボ	セ	L	出
通		オ	

## 001 温度を表示する



ボ	セ	L	出
通		4	

## 002 明るさを表示する





# 001 温度を表示する

micro:bit が持っているセンサー(温度・光・傾き・磁気)を使うベースサンプルとして作りました。 温度センサーの数値をLEDに表示します。他のセン サーの数値ブロックに「置き換え」てみてもよいです ね。



- ・・・使用するのは、センサー・LED
- ・・・分類をわかりやすくするための表記

2018/4/18作成

## 002 明るさを表示する

LEDにグラフ表示する方法を調べて作りました。 センサーの数値→グラフ化。



# ・・・使用するのは、センサー・LED

2018/4/18作成



ボ	セ	L	出
通		オ	

# 003 ボタンで温度を表示する





# 004 ボタンで明るさを表示する



# 003 ボタンで温度を表示する

「001 温度を表示する」をボタンで動作するよう に「置き換え」たものです。

「ずっと」→「ボタンAが押されたとき」。ボタンの使 い方のベースです。



・・・ボタン・センサー・LED

2018/4/18作成

# <u>004 ボタンで明るさを表示する</u>

「002 明るさを表示する」のボタン表示パターン です。このように、いろいろな組み合わせを考えるこ とができますね。1000ミリ秒(1秒)で表示が消え るようにしています。



・・・ボタン・センサー・LED

2018/4/25作成



## 005 ボタンで明るさ・温度を表示する







## 005 ボタンで明るさ・温度を表示する

「003 ボタンで温度を表示する」と「004 ボタ ンで明るさを表示する」を組み合わせたものです。

複数のプログラムを組み合わせることができることを示したものです。



・・・ボタン・センサー・LED

## 2018/4/18作成



006 暗くなるとLEDが点く

ボ	セ	L	出
通		*	





# 006 暗くなるとLEDが点く

ボタンではなくセンサーの数値をきっかけにLED の点き方を変えています。(条件分岐)

設定している「センサーの数値」を変えることで、 変化する明るさ(感度)を変えることができます。



・・・センサー・LED

#### 2018/4/22作成



ボ	セ	L	出
通		オ	

Î	検索 Q	ボタン A ▼ が押されたとき
	■ 基本	音を鳴らす 高さ (Hz) <u>真ん中のソ</u> 長さ 750
	⊙ 入力	音を鳴らす 高さ (Hz) <u>夷ん中のソ</u> 長さ 1/4 ▼ 拍
	○ 音楽	☆ <u> </u>
$\circ$ $\circ$ $\circ$ $\circ$ $\circ$ $\circ$ $\circ$	C LED ·	$\dot{\mathbf{x}}$
	.al 無線	
	<b>C</b> ループ	
	☆ 論理	音を鳴らす 高さ(Hz) <u><u></u> <mark> </mark></u>
	■ 変数	音を鳴らす 高さ(Hz) <u>真ん中のソ</u> 長さ 750
	■ 計算	音を鳴らす 高さ(Hz) <mark>夷ん中のソ</mark> 長さ 1/4 ▼ 拍
	✓ 高度なブロック	音を鳴らす 高さ(Hz) <mark>真ん中のラ</mark> 長さ 1 ▼ 拍
		音を鳴らす 高さ(Hz) <u>真ん中のソ</u> 長さ 1 ▼ 拍
	-L 13	音を鳴らす 高さ(Hz) 上のレ 長さ 1 ▼ 拍
	- カーや	音を鳴らす 高さ(Hz) 上のド 長さ 2 ▼ 拍
イヤホン	/をつなぐ	

008 電子オルゴールを鳴らす

ボ	セ	L	出
通		オ	





#### <u>007 Happy Birthday を鳴らす</u>

micro:bit が音を出力できることを示しているサンプル です。楽譜を見ながらブロックをつなげていきます。四分 音符=1000なので、選択肢にない符点八分音符を 750で入力(「計算」の数値入力ブロックで)しています。 micro:bit 単体にはスピーカーがないので、PO端子と GND端子にスピーカーやイヤホンをつないで確認します。

ボ	セ	L	出	2018/4/23作成
通		オ		2019/1/3修正

・・・出力(この場合は音声として)

#### 008 電子オルゴールを鳴らす

PI端子とGND端子に導線をつなぎ、その間に「電流 が流れたら」音を鳴らすように置き換えたものです。

PO端子とGND端子にスピーカーをつないでいても、 PI端子とGND端子のタッチが別に動作するか確認した ものです。



・・・センサー・出力

2018/4/23作成



ボ	セ	L	出
通		オ	

009 テスターで音を鳴らす



ド	セ	L	出
通		オ	

## 010 明るさで音が変わるテルミン





## 009 テスターで音を鳴らす

「008 電子オルゴールを鳴らす」をシンプル化したものです。PI端子とGND端子に導線をつなぎ、その間に「電流がながれたら」音が鳴ります。



・・・センサー・出力

導線から明るさセンサーに置き換えたものです。 初期の電子楽器「テルミン」のように手を明るさセ ンサーにかざすことで、音の高さが変わるようにし ています。

明るさの数値を変えることで、変化の度合い(感度)を変えることができます。



・・・センサー・出力

2018/4/23作成



ボ	セ	L	氏
通		*	

011 ストップウォッチ





<u>011 ストップウォッチ</u>

ボタン操作で時間を測るようにしています。

変数「停止」「秒数」を作り、「停止」が1のときは 秒数に加算、「停止」が0のときは加算しないように しています。

・・・ボタン・LED

ボ	セ	L	Æ
通		オ	

2018/5/6作成



## 012 明るさで秒数計測



21

# <u>012 明るさで秒数計測</u>

懐中電灯の灯りをセンサーにあてておき変数「最 初の明るさ」にしています。ボタンでスタートし、何か が灯りをさえぎる(前を横切る)までを計測できな いかを考えたプログラムです。



・・・ボタン・センサー・LED

2018/5/6作成



# 013 コンパス(Compass)

ボ	セ	L	出
通		オ	







# 013 コンパス(Compass)

micro:bit の磁気センサーでコンパスを作りました。

NEWS (東西南北)をアルファベットで表示します。

注意・・・最初に磁気センサーの補正のため、LED が全部点くようにぐるぐるまわします。 TILT TO FILL SCREEN→うまくいくと下のマークが出ます。



・・センサー・LED

2018/5/14作成



ボ	セ	L	出
通		オ	

# 014 2台で送受信



ボ セ L 出 通 オ	水準器
傾きセンサーで計測	
	0    1     2     3V    GND 変数 x * を 2 にする 変数 y * を 2 にする Tっと 消払I x x * y Y * 変数 x * を -1 - * 加速度 x * + * 1000 ÷ * 400 にする 変数 y * を -1 - * 加速度 y * + * 1000 ÷ * 400 にする
<ul> <li>□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □</li></ul>	

Created by ©Yoichiro.Mochizuki 2018-2021



# 014 2台で送受信

micro:bit が持つBluetoothの通信機能をテストしたものです。通信を受け取ったら文字を表示します。(「receivedNumber」は、無線を受け取ったらのブロックからとり出します)



・・・ボタン・LED・通信

2018/5/17作成

# 015 水準器

micro:bit の傾きセンサーをテストしたものです。

最初x,yが2,2なのは、LEDの表示位置が0~4 のためです(2,2が真ん中)。傾きが変わると、それ にあわせて点灯する位置を変えるサンプルです。





016 傾きメーター







【説明】 <u>016 傾きメーター</u>

micro:bit の傾きセンサーをテストしたものです。 左右の傾きを検出してバーとして表示するようにし たものです。



・・・ボタン・LED・通信

2018/5/21作成



# 017 方位磁針





017 方位磁針

micro:bitの磁気センサーを使って、LED表示します。文字ではなくLEDのドットです。

5×5のLEDで、図形を表示する方法を模索した プログラムです。

注意・・・最初に磁気センサーの補正のため、LED が全部点くようにぐるぐるまわします。 TILT TO FILL SCREEN →うまくいくと下のマークが出ます。

ボ	セ	L	出
通		オ	

・・・センサー・LED



2018/5/23作成



ボ	セ	L	出
通		*	

018 方位磁針16方位-1





018 方位磁針16方位-1

micro:bitの磁気センサーを使って、LED表示します。文字ではなくLEDのドットです。

5×5のLEDで、図形を表示する方法を模索した プログラムですが、8方位版と違い、より矢印の図 形を表すのが難しくなっています。

注意・・・最初に磁気センサーの補正のため、LED が全部点くようにぐるぐるまわします。 TILT TO FILL SCREEN →うまくいくと下のマークが出ます。



・・・センサー・LED

(プログラムの続きがあります 018-2)

2018/5/24作成



018 方位磁針16方位-2



Created by ©Yoichiro.Mochizuki 2018-2021



33

【説明】 <u>018 方位磁針16方位-2</u> 018-1の続きになります。

2018/5/24作成





019 方位磁針16方位program分割-1

ボ	セ	L	出
通		*	





019 方位磁針16方位program分割-1

micro:bitの磁気センサーを使って、LED表示します。文字ではなくLEDのドットです。

5×5のLEDで、図形を表示する方法を模索した プログラムですが、8方位版と違い、より矢印の図 形を表すのが難しくなっています。

プログラムが長くなったため、分割ができないか 工夫したものです。

「ずっと」を2つ作り、022「方位磁針16方位」を 分割して入れました。「でなければ」に何も入れない ことで、2つが成り立っているようです。

注意・・・最初に磁気センサーの補正のため、LED が全部点くようにぐるぐるまわします。 TILT TO FILL SCREEN →うまくいくと下のマークが出ます。



・・・センサー・LED 2018/5/24作成 (プログラムの続きがあります 019-2)


#### ボ セ L 出 通 才 検索... Q 157 方角 (°) かっ・ 方角(<sup>。</sup>) < - 202 **a6** Θ でなければも し 基本 LED画面に表示 〇 入力 長いプログラムを分割 C LED $\odot$ ... 無線 C ループ ずっと ŧι 202 方角(°) 方角 (°) 247 **なら** かつ・ < -24 論理 LED画面に表示 変数 ■ 計算 高度なブロック 247 方角 (°) 292 **46** Θ 方角 (°) でなければもし 10--









【説明】 <u>019 方位磁針16方位program分割-2</u> 019-1の続きになります。

2018/5/24作成



38

020 暗くなったことを検知

ボ	セ	L	出
通		4	





# 020 暗くなったことを検知

micro:bit の明るさセンサーを使って、明るさの 変化によりLED表示を変化させます。

設定している明るさの値を変化させると、変化の 度合い(感度)が変わります。

ボ	セ	L	出
通		オ	

・・・センサー・LED

#### 2018/5/28作成 2019/1/6修正



# 021 相互にストップウォッチ





# <u>021 相互にストップウォッチ</u>

micro:bit のBluetooth通信機能を使って相互 通信させるプログラムです。

ストップウォッチのプログラムを転用して試しています。



・・・ボタン・LED・通信

2018/6/14作成



# 022 通信で合奏

ボ	セ	L	出
通		オ	



ボ	セ	L	出
通		¥	

# 023 明るさ変化を送信

検索	Q	最初だけ		無線で受信したとき recei	vedNumber			
■■ 基本		無線のグループを設定 1		もし receivedNumber		0	> <b>&amp;</b> 6	
○ 入力		LED画面に表示		LED画面に表示	+ +		-	
C LED								
無線				<u>i</u> eeei	+ +			
<b>C</b> ループ				<b>→時停止(ミリ秒)</b> 1	90 💌			
≭ 論理		Job						
■ 変数		毎週で数値を送信	C.P.					
■ 計算								
					+ +			
▶ ◆ 局度なフロッ	ク							



【説明】 <u>022 通信で合奏</u>

micro:bit のBluetooth通信機能が、一対多で 動作するか確認するために作ったプログラムです。

無線のグループを同じにしたmicro:bit を多数配置しておき、一台をコンダクター(信号を送信する) とすることで、合奏もできるのではないかと考えました。スピーカーをPO端子とGND端子に接続しておきます。



・・・ボタン・出力・通信

2018/6/14作成

# 023 明るさ変化を通信

micro:bit のBluetooth通信機能を使って、一 台の明るさセンサーの変化をもう一台に送信するテ ストプログラムです。



・・・センサー・LED・通信

2018/6/15作成





# 024 揺れたら送信

検索 Q	最初だけ		無線で受信したとき receivedNumber	
■ 基本	無線のグループを設定 1		ŧ L receivedNumber ■ ▼	0 <b>4</b> 6
◎ 入力	LED画面に表示		LED画面に表示	
♀ 音楽				
C LED				
			→時待止(ミリ秒) 100 ▼	
• 🖸 🕺 🗛 ax:980	ずっと		LED画面に表示	
	もし 加速度 X - > ·	· 200 &5		
$\Box$	無線で数値を送信 0 ◆	-		
•0			€	

۴	セ	L	出
邇		*	

# 025 明るさ数値送信





# 【説明】 <u>024 揺れたら送信</u>

micro:bit のBluetooth通信機能を使って、一 台の明るさセンサーの変化をもう一台に送信するテ ストプログラムです。



・・・センサー・LED・通信

2018/6/18作成

# 025 明るさ数値送信

micro:bit のBluetooth通信機能を使って、一 台のセンサー数値自体をもう一台に送信できるか テストしたものです。

この場合は、ボタンを押したときの明るさセンサーの数値を送信しています。



・・・ボタン・センサー・LED・通信

# 2018/6/18作成



026 ひらがな表示





<u>026 ひらがな表示</u>

micro:bit は、日本語が表示できない、ということ で、なんとか表示したように工夫できないかと考え たものです。

ドットで疑似的に表示しているだけですが、「できないことをなんとかできないかと『エ夫』した例」として作ってみました。



•••LED

2018/7/29作成



# 027 傾き(ピッチ)角度を表示





# 027 傾き(ピッチ)角度を表示

ボタンを押したときのmicro:bit 基板の角度 (ピッチのほう。もう一つの角度がロール)を使って 表示するプログラムです。

これは太陽の地面からの角度(南中高度)を測る ことができないかというアイデアを実現するための プログラムの一部です。



2018/7/29作成



・・・ボタン・センサー・LED



# 028 明るさにより発信音の高さを変える





028 明るさにより発信音の高さを変える

ボタンを押したときの明るさセンサーの値により、 音の高さを変えて出力するものです。

これは太陽の地面からの角度(南中高度)を測る ことができないかというアイデアを実現するための プログラムの一部です。

ボ	セ	L	出
通		オ	

・・・ボタン・センサー・出力

#### 2018/7/29作成



029 太陽高度計

ボ	セ	L	出
通		4	





029 太陽高度計

これは太陽の地面からの角度(南中高度)を測る ことができないかというアイデアを実現するための プログラムの一部です。

リアルタイムで明るさセンサーの値から音の高さを 変えて出力するプログラムと、ボタンを押したときの micro:bit 基板の角度 (ピッチのほう。もう一つの 角度がロール)から太陽高度を表示するプログラム を組み合わせています。

太陽に向けて、一番明るい時を「太陽に向かった 時」と考えて、そのときにボタンを押すことで測ろう としています。

2つのプログラムが連動していないのもポイントで す。



- ・・・ボタン・センサー
  - ・LED・出力





# 030 ふりこ送信

ボ	セ	L	出
通		オ	



最下点に来た時をカウント





030 ふりこ送信

ふりこの周期を測るためのプログラムの一部です。 「ふりこの最下点に来た時、ロールが0(水平)になる」ことを使っています。

水平になった回数をBluetoothで、もう一台に 「0」を送信する部分です。



・・・ボタン・センサー・LED・通信

2018/8/15作成



56

031 ふりこ受信 段階1

ボ	セ	L	出
通		4	





031 ふりこ受信 段階1

ふりこの周期を測るためのプログラムの一部です。 受信プログラムの原型で、受信したら秒数カウン トが始まるようにしています。



・・・ボタン・センサー・LED (段階2があります)







Created by ©Yoichiro.Mochizuki 2018-2021

032 ふりこ受信 段階2

ボ	セ	L	出
通		4	





表示を消す

でなければ

 $\odot$ 

 $\odot$ 

100

Θ

表示を消す

数を表示

でなければ

Ð

文字列を表示 " STOP/"

時間 🔻 🛛 🕂 🔻

Θ

# 032 ふりこ受信 段階2

ふりこの周期を測るためのプログラムの一部です。 030「ふりこ送信」と組み合わせます。

受信プログラムは、送られてきた「最下点に来た 回数」が「」」から秒数をスタートして「3」になった らストップします。

(ふりこを離して、最下点 | 回目~3回目の間が周期になる)

ボ	セ	L	出
通		4	

・・・ボタン・センサー・LED・通信





# 033 文字列を送信

ボ	セ	L	出
通		¥	



61

**033 文字列を送信** 

「文字列そのものを送信できませんか」という質問 からつくってみたものです。

一台のmicro:bit から「文字列」を送信して、受信側のmicro:bit では、「送られてきた文字列」を表示します。

送受信部が両方入っていますが、分割して2台に それぞれプログラムを入れても動作します。



・・・ボタン・LED・通信

2018/8/27作成



034 接近したら点灯(距離センサー)

ボ	セ	L	出
通		オ	





034 接近したら点灯(距離センサー)

<u>https://tfabworks.com/product/tfw-</u> dsI/

端子にワンタッチではめこむタイプのオプションを 使ってみました。

<u>https://makecode.microbit.org/45600-</u> 49587-10845-13475

(距離センサー(pxt-uds版))を参照して変更しま した。距離の数値を変更すると、反応する距離を変 更することができます。

ボ	セ	L	出
通		オ	

・・・センサー・LED・オプション

2018/9/24作成





ボ	セ	L	出
通		オ	



<u>035 スイッチを制御(制御スイッチ)</u>

<u>https://tfabworks.com/product/mb-set-</u> sw2/ を使用

端子にワンタッチではめこむタイプのオプションを 使ってみました。

フィルムターミナルを懐中電灯の乾電池の接触部 分のすきまにはさみこむことで、PI端子から「オン オフ」を制御することができます。懐中電灯のスイッ チを入れても、ターミナルが「オフ」なら点きません。 micro:bitのAボタンを押すと出力がターミナルを 「オン」にするので、懐中電灯が点くわけです。

乾電池で動くものならいろいろ制御できそうですね。

ボ	セ	L	出
通		オ	

・・・ボタン・出力・オプション

2018/9/27作成 2021/1/ 4修正



# 036 リモコン・スイッチで点灯(制御スイッチ)





micro:bit用プログラム 制御スイッチセット 電磁石版 (TFabWorks)



036 リモコン・スイッチで点灯(制御スイッチ)

<u>https://tfabworks.com/product/mb-set-</u> sw2/

端子にワンタッチではめこむタイプのオプションを 使ってみました。

035「スイッチを制御」を通信機能で、もう一台の micro:bit から「オンオフ」するように変えたもので す。こうすることで離れた場所にあるものも制御する ことができます。



・・・ボタン・出力・通信・オプション

2018/9/27作成 2021/1/ 4修正



## 037 人感センサー(距離センサー・制御スイッチ)-1

ボ	セ	L	出
通		オ	

	検索 Q	đ - k
	■ 基本	* J.  S#本別学すみ trie PB * sela D1 *  スキ 50 たら
Break and the second	● 入力	
	♀ 音楽	LED画面に表示 最初だけ
	C LED	無線のグループを設定 1
	F WhaleySans Font	
	.』 無線	
0 1 2 3V GND	□ 超音波距離センサ	無線で数値を送信 1 ちゅうちゅうちょうちょうちょうちょうちょうちょう
	<b>C</b> ループ	eathul O
	☆ 論理	LED直面に表示
B	■ 変数	
·n n.	■ 計算	
	<sup>_</sup> 榆索 0	
	■ 基本	
	○ 入力	
· · · · ·	♀ 音楽	Crathalf $\Theta$
	C LED	LED画面に表示
and the second se	F WhaleySans Font	
	·····································	
	▶ 超音波距離センサ	
Contraction of the second seco	<b>C</b> ループ	──時停止(ミリ秒) 100
	☎ 論理	無線で数値を送信 0 ちゅうちゅうちゅうちゅうちゅうちゅうちゅうちょう
	■ 変数	





<u>037 人感センサー(距離センサー・制御スイッチ) -1</u>

<u>https://tfabworks.com/product/tfw-dsl/</u>

ー台のmicro:bit の端子にワンタッチではめこむ タイプの距離センサーを使います。

(<u>https://makecode.microbit.org/45600-49587-10845-13475</u>(距離センサー(pxt-uds版))を参照して変更)

もう一台のmicro:bit の端子にワンタッチでは めこむタイプの制御スイッチ経由で懐中電灯をつな ぎます。

036「リモコン・スイッチを制御」と距離センサー を組み合わせることで近づいたら懐中電灯が点くよ うになります。数値を変えると、感じる距離を変える ことができます。



・・・ボタン・センサー・LED・出力

・通信・オプション

(プログラムの続きがあります 043-2)



# 037 人感センサー(距離センサー・制御スイッチ)-2





<u>037 人感センサー(距離センサー・制御スイッチ)</u> <u>-2</u>

037-1の続きになります。

https://tfabworks.com/product/mb-setsw2/

端子にワンタッチではめこむタイプのオプションを 使っています。

> 2018/10/1作成 2021/01/4修正


## 038 距離を表示(距離センサー)

ボ	セ	L	出
通		オ	

	検索	Q Î	· +												
	■■ 基本		ずっと												
	⊙ 入力														
			数を	表示	小数:	急以下四	舍五入,	2	距離を	測定す	a trif	P0 -	echo	P1 -	
·O 🚺 O.	C LED		85	停止(	ミリ秒)	1000	•								
	F WhaleySans F	ont													
	無線														
0 1 1 2 3V GND	□ 超音波距離セ	ンサ													
	C ループ														
	☆ 論理														
	■ 変数														
	■ 計算		+ +	+	+ +	+	+	+	+	+	+	+ +	+	+	+ · ·







# 038 距離を表示(距離センサー)

<u>https://tfabworks.com/product/tfw-</u> dsI/

ー台のmicro:bit の端子にワンタッチではめこむ タイプの距離センサーを使います。

距離センサーの数値自体を表示するようにしてみました。

(<u>https://makecode.microbit.org/45600-49587-10845-13475</u>(距離センサー(pxt-uds版))を参照して変更)

ボ	セ	L	出
通		オ	

・・・センサー・LED・オプション

2018/10/3作成 2020/1/10修正



# 039 明るさグラフ

ボ	セ	L	出
通		¥	



75

【説明】 039 明るさグラフ

センサーの値をバーの長さ(グラフ)表示(一番左の列に)させるテストとして作ったものです。値を他のセンサーに置き換えることもできますね。



・・・センサー・LED

2018/10/3作成 2020/1/10修正



## 040 明るさグラフリアルタイム

ボ	セ	L	出
通		オ	

リアルタイムでバーの

長さが変化







040 明るさグラフリアルタイム

039「明るさグラフ」をリアルタイムで更新するようにしてみたものです。

別なセンサーの値にすれば、そのグラフを作ること もできます。



・・・センサー・LED

2018/10/3作成 2020/1/10修正



#### 041 距離を測定(距離センサー)



79

# 041 距離を測定(距離センサー)

<u>https://tfabworks.com/product/tfw-</u> dsI/

ー台のmicro:bit の端子にワンタッチではめこむ タイプの距離センサーを使います。

Aボタンを押してから1秒間に近づいた距離を測るようにしています。

(<u>https://makecode.microbit.org/45600-49587-10845-13475</u>(距離センサー(pxt-uds版))を参照して変更)

ボ	セ	L	出
通		オ	

・・・センサー・LED・オプション

2018/10/4作成 2020/1/11修正



042 micro:bit ファームウェアアップデート -1

### USB接続して、DETAILS.TXTを確認



#### この場合 version 0241

	_		$\times$
ファイル( <u>F)</u> 編集( <u>E</u> ) 書式( <u>O</u> ) 表示( <u>V</u> ) ヘルプ( <u>H</u> )			
# DAPLink Firmware - see https://mbed.com/daplink			^
Unique ID: 9900000041414e45002f000f0000002000000009	79699	01	
HIC ID: 97969901			
Auto Reset: 1			
Automation allowed: 0			
Daplink Mode: Interface			
Interface Version: 0241			
Git SHA: fa4132987102c51c110751d8bdb8a74aeb7b071b			
Local Mods: 1			
USB Interfaces: MSD, CDC, HID			
Interface CRC: 0xe369fb16			
			$\sim$
<			>

# <u>042 micro:bit ファームウェアアップデート</u>

当時のベータサイトで作ったサンプルの表示が場 合によって違っていたため、micro:bit本体の ファームウェアがアップデートされていないためか と考えて、アップデートした手順を記録したものです。

> 2018/10/8作成 2020/1/11修正



# 042 micro:bit ファームウェアアップデート -2

#### <u>https://github.com/ARMmbed/DAPLink/releases</u> にアクセス



# この場合 version 0254 のデータがあるので 0254\_release\_package\_f499eb6e.zip をダウンロード



# 【メモ】



### ZIPファイルを開いて、hexファイルを探す

0254_k20dx_rblnano_0x5000.bin	BIN ファイル	107 KB	
0254_k20dx_twrke18f_0x8000.bin	BIN ファイル	95 KB	無
0254_k20dx_twrkl28z72m_0x8000.bin	BIN ファイル	95 KB	無
0254_k20dx_xdot_l151_0x8000.bin	BIN ファイル	95 KB	無
0254_k26f0x0000.bin	BIN ファイル	188 KB	無
0254_k26f_frdmk32w042_0x8000.bin	BIN ファイル	188 KB	無
0254_kl26z_0x0000.bin	BIN ファイル	95 KB	無
0254_kl26z_microbit_0x8000.hex	Microsoft MakeCode bina	268 KB	無
0254_kl26z_nina_b1_0x8000.bin	BIN ファイル	95 KB	無
0254_lpc11u350x0000.bin	BIN ファイル	60 KB	無
📄 0254_lpc11u35_6lowpan_borderrou	BIN ファイル	60 KB	無
📄 0254_lpc11u35_6lowpan_borderrou	BIN ファイル	60 KB	無
📄 0254_lpc11u35_6lowpan_borderrou	BIN ファイル	60 KB	無

#### リセットボタンを押しながら基板をUSB接続すると メンテナンスモードが開く





# 【メモ】





## 042 micro:bit ファームウェアアップデート -4

#### hexファイルをドロップすると再起動される



## DETAILS.TXT でversionが0254になったことを確認

DETAILS.TXT - 火モ帳	_		$\times$
ファイル(E) 編集(E) 書式(Q) 表示(V) ヘルプ(H)			
<pre># DAPLink Firmware - see https://mbed.com/daplink Unique ID: 9900360141404e450036901600000074000000097969901 HIC ID: 97969901 Auto Reset: 1 Automation allowed: 0 Overflow detection: 1 Daplink Mode: Interface Interface Version: 0254 Git SHA: f499eb6ec4a847a2b78831fe1acc856fd8eb2f28 Local Mods: 0 USB Interfaces: MSD, CDC, HID, WebUSB Interface CRC: 0x18740bef Remount count: 0 URL: https://microbit.org/device/?id=9900&amp;v=0254</pre>			<
<			>
1 行、1 列 100% Windows (CRLF)	UTF-	8	

# 【メモ】



043 人感センサー

ボ	セ	L	出
通		オ	







043 人感センサー

<u>https://tfabworks.com/product/tfw-</u> <u>rk2/</u>

を使用

<u>http://tfab.jp/rk2p</u> を編集してつくりました。

人が近づいたら、チェックを表示 遠ざかったら(いなかったら)×を表示します。 Aボタン、Bボタンをチェック用にしています。



・・・ボタン・センサー・LED・オプション



044 ランダム数値表示

ボ	セ	L	出
通		オ	





044 ランダム数値表示

ここまで「乱数」を使ってこなかったので、初めて使ってみたものです。

ボタンを押すことで0~9の中から一つ表示します。



2019/1/17作成 2020/8/25修正

・・・ボタン・LED



#### 045 じゃんけん

ボ	セ	L	出
通		オ	





【説明】 <u>045 じゃんけん</u>

乱数を使ってじゃんけんをするものです。 Aボタンを押したとき乱数で発生した0~2を使って、 LEDで「パー」「チョキ」「グー」を表示します。



・・・ボタン・LED

2019/1/17作成 2020/1/11修正 2020/8/25修正





#### 046 じゃんけん無線判定

ボ	セ	L	出
通		オ	





### 046 じゃんけん無線判定

・・・ボタン・LED

出

「045じゃんけん」を、相手の出した手を無線で読み取り、勝敗判定するようにしたものです。

乱数を発生させるときに、その数をBluetoothで送 信するため、相手はその数値と自分が発生した数 値(パー、チョキ、グーに該当)と比較して勝敗を判 定します。

ニ重の条件(自分の数値に対して送信されてきた 相手の数値3パターンで判定)を設定することで、 プログラムを短くしています。

> 2019/1/18作成 2020/1/12修正 2020/8/26修正



ボ

セ

Created by ©Yoichiro.Mochizuki 2018-2021

047 磁気表示

ボ	セ	L	出
通		オ	





## 047 磁気表示

理科の実験で作る電磁石の強さを測りたいと考え たものです。

磁気センサーでとらえた磁力の強さを数値表示しま す。

注意・・・最初に磁気センサーの補正のため、LED が全部点くようにぐるぐるまわします TILT TO FILL SCREEN →下のマークが出るようにします。



2019/1/23作成

・・・ボタン・センサー・LED



## 048 磁力計

ボ	セ	L	出
通		オ	





99

<u>048 磁力計</u>

「047磁気表示」を「002明るさを表示する」のグ ラフ表示のやり方を使って、作り変えたものです。

磁力が強くなるとLED上のグラフが明るくなるので、 磁石を見つけたり、磁力の強さを測ることができる とかんがえました。

注意・・・最初に磁気センサーの補正のため、LED が全部点くようにぐるぐるまわします TILT TO FILL SCREEN →下のマークが出るようにします。



2019/2/19作成

・・・ボタン・センサー・LED



## 049 検流 (ペルチェ素子)







### 049 検流(ペルチェ素子)

端子POに電流が入力されたかどうかを測るサンプ ルを作ってみました。

入力のためには、「ペルチェ素子 (熱と電流を相互 変換する)」を使いました。

入力されない間は、×を表示し、ペルチェ素子の片 面を指で温めると電流が発生し入力されるため、 ✓に表示が変わります。温度が下がると入力がな くなるため、×に戻ります。

※写真のようなペルチェ素子は数百円からあります。 ペルチェ素子は、電流が流れることで片面から熱を もう片面に移動することで冷却加熱するものです。 逆に温めたり冷やすことで電流を発生させます。



2019/3/2作成

・・・LED・入力



## 050 検流グラフ(ペルチェ素子)











050 検流グラフ(ペルチェ素子)

端子P0に電流が入力されたかどうかを測るサンプ ル049をグラフ表示にしたものです。感度調整は 「最大値」で設定しています。

入力のためには、「ペルチェ素子 (熱と電流を相互 変換する)」を使いました。

入力されない間は、×を表示し、ペルチェ素子の片 面を指で温めると電流が発生し入力されるため、 ✓に表示が変わります。温度が下がると入力がな くなるため、×に戻ります。

※写真のようなペルチェ素子は数百円からあります。 ペルチェ素子は、電流が流れることで片面から熱を もう片面に移動することで冷却加熱するものです。 逆に温めたり冷やすことで電流を発生させます。



2019/3/4作成 2020/1/12修正

・・・LED・入力



ボ	セ	L	出
通	ኦ	オ	

#### 051 計算機-1





051 計算機-1

micro:bitで計算機をつくることができないか試したものです。

Aボタンの役割・・・

数字を増やす、四則を変える

Bボタンの役割・・・

今計算のどの段階か区切りをつける

A+Bボタン(同時押し)の役割・・・

リセット

計算の進み方

・~される数を決める。Aボタンを押すと数が増える。 ・Bボタンを押して、~される数を決定。



# ・・・ボタン・LED

※051 計算機-2 に続きます。



ボ	セ	L	出
通	ኦ	オ	

# 051 計算機-2







## 051 計算機-2

※051 計算機-1 の続きです。

- ・Aボタンを押して、四則(+-×÷)を変化させます。
  ・Bボタンを押して、四則を決定。
- ・~する数を決める。Aボタンを押すと数が増える。
- ・Bボタンを押して、~する数を決定。

・Aボタンを押すと、=が1秒間表示され、計算結果 が表示されます。



・・・ボタン・LED

2019/3/4作成



108
ボ	セ	L	出
通	አ	¥	

## ※051 計算機と比較した変更部分(関数)のみ掲載しています。





052 計算機(関数)

「051 計算機」の計算結果を表示する部分を、関数(処理をまとめたもの)に置き換えています。 関数は、「高度なブロック」→「f(x)関数」から作成 します。

051の計算結果表示部分を、関数に入れて、表示 する際は、関数を「呼び出し」て表示させるようにし ました。



・・・ボタン・LED

2019/3/4作成



053 気温を小数表示

ボ	セ	L	出
通		大	







111

Created by ©Yoichiro.Mochizuki 2018-2021

**053 気温を小数表示** 

https://tfabworks.com/product/tfw-enl/

気温・湿度・気圧センサーを使用

以前拡張パーツ(Weather:bit)を使っていました が、TFabWorksの新しいパーツに変えてプログラ ムしたものです。

https://tfab.jp/stem

専用ブロックを読み込んで編集しました。

(℃)で表示しますが、小数表示のほかに整数表示 もできます。



・・・ボタン・センサー・LED・オプション

2021/1/1作成



054 湿度を小数表示

ボ	セ	L	出
通		大	









**054 湿度を小数表示** 

https://tfabworks.com/product/tfw-enl/

気温・湿度・気圧センサーを使用

以前拡張パーツ(Weather:bit)を使っていました が、TFabWorksの新しいパーツに変えてプログラ ムしたものです。

https://tfab.jp/stem

専用ブロックを読み込んで編集しました。

(%)で表示しますが、小数表示のほかに整数表示 もできます。



・・・ボタン・センサー・LED・オプション

2021/1/1作成



055 気圧を小数表示

ボ	セ	L	出
通		大	







**055 気圧を小数表示** 

https://tfabworks.com/product/tfw-enl/

気温・湿度・気圧センサーを使用

以前拡張パーツ(Weather:bit)を使っていました が、TFabWorksの新しいパーツに変えてプログラ ムしたものです。

https://tfab.jp/stem

専用ブロックを読み込んで編集しました。

(hPa)で表示しますが、小数表示のほかに整数表 示もできます。



・・・ボタン・センサー・LED・オプション

2021/1/1作成



056 気温・湿度・気圧を小数表示

ボ	セ	L	出
通		才	







117

# 056 気温・湿度・気圧を小数表示

https://tfabworks.com/product/tfw-enl/

気温・湿度・気圧センサーを使用

以前拡張パーツ(Weather:bit)を使っていました が、TFabWorksの新しいパーツに変えてプログラ ムしたものです。

https://tfab.jp/stem

専用ブロックを読み込んで編集しました。 ボタンに割り当てて、それぞれの数値を表示させま す。



・・・ボタン・センサー・LED・オプション

2021/1/1作成



#### 057 簡易熱中症計

ボ	セ	L	出
通		才	





119

057 簡易熱中症計

<u>https://tfabworks.com/product/tfw-enl/</u>

気温・湿度・気圧センサーを使用

以前拡張パーツ(Weather:bit)を使っていました が、TFabWorksの新しいパーツに変えてプログラ ムしたものです。簡単な熱中症を予防する機械がで きないかと考えたものです。気温が高くなる、湿度 が高くなると、熱中症になりやすいと考え(あくまで 簡易の考え方です)、表のように条件を考えました。

		70%		50%		30%	10%
35°C	×		Δ		0		-
30℃	Δ		Δ		0		-
25℃	0		0		0		-
20°C	-		1		-		-

https://tfab.jp/stem

専用ブロックを読み込んで編集しました。

条件分岐を組み合わせて条件にあったら表示を出 すようにしています。



・・・ボタン・センサー・LED・オプション



## 058 気温数値を送信







#### ※「059 気温数値を受信」とセットになります。



058 温度数値を送信

<u>https://tfabworks.com/product/tfw-enl/</u>

オプションパーツ(拡張基板)である温度・湿度・ 気圧センサーをmicro:bit の端子部に差し込んで 使います。

https://tfab.jp/stem

にアクセスすることで、専用の「STEM」ブロック から「温度」ブロックを使い、数値を送信します。

※「059 気温数値を受信」とセットになります。



・・・センサー・LED・通信・オプション

#### 2021/1/2作成



## 059 気温数値を受信

ボ	セ	L	出
通	ኦ	オ	





## ※「058 気温数値を送信」とセットになります。



**059 温度数値を受信** 

<u>https://tfabworks.com/product/tfw-enl/</u>

オプションパーツ(拡張基板)である温度・湿度・ 気圧センサーをmicro:bit の端子部に差し込んで 使います。

受け取った数値を温度として表示します。

※「058 気温数値を送信」とセットになります。

ボ	セ	L	出
通	メ	¥	

・・・LED・出力・通信

#### 2021/1/2作成



#### 060 温度・湿度を同時送信







## ※「061気温・湿度を同時送信」とセットになります。

Created by ©Yoichiro.Mochizuki 2018-2021



# 060 温度・湿度を同時送信

<u>https://tfabworks.com/product/tfw-</u> enl/

オプションパーツ(拡張基板)である温度・湿度・ 気圧センサーをmicro:bit の端子部に差し込んで 使います。

## https://tfab.jp/stem

にアクセスすることで、専用の「STEM」ブロック から「温度」「湿度」ブロックを使い、name(文字 列)とvalue(数値)をセットで送信しています。

- ・temp2=温度
- ・humi2=湿度

※「061気温・湿度を同時受信」とセットになります。



・・・センサー・LED・通信・オプション

## 2020/1/13作成



061 気温・湿度を同時受信



## ※「060 気温・湿度を同時送信」とセットになります。



# 061 温度・湿度を同時受信

<u>https://tfabworks.com/product/tfw-</u> enl/

オプションパーツ(拡張基板)である温度・湿度・ 気圧センサーをセットしたmicro:bit 送信側から送 られている、文字列temp2(気温)とvalue(数値) と文字列humi2(湿度)とvalue(数値)のセットを 受信しています。 変数temp3(気温用)とhumi3(湿度用)を追加し て、セットで表示しています。

※論理→くらべるの、文字列比較で、

name=temp2 かどうか調べています。

※受信と表示のタイミングをはかるため、変数を一 度0にして調整しています。

※「060 気温・湿度を同時送信」とセットになります。



・・・LED・通信

2020/1/13作成



# いいい、「サンプル プログラミング集」 v2 編

大分県立芸術文化短期大学 情報コミュニケーション学科 非常勤講師 望月陽一郎 mochizuki@oita-pjc.ac.jp 作品のクレジットを表示 すること・営利目的での 利用をしないこと・元の 作品を改変しないこと (作者の許可を得ている 場合は除きます)

Created by ©Yoichiro.Mochizuki 2018-2021



# <u>目次</u>

- 001 重ねて音を鳴らす
- 002 音がしたらビックリ
- 003 タッチしたらビックリ
- 004 音のしきい値
- 005 静かになったら
- 006 音量グラフ
- 007 音量を送信
- 008 音量を受信
- 009 タッチする楽器
- 010 メロディを打楽器に



## 001 重ねて音を出す

ボ	セ	L	出
通		オ	v2

Comicro:bit 🖀 ホーム 🔩 共有	■ ブロック ● Python マ
・    ・	ボタン A ▼ が押されたとき
	ボタン B ▼ が押されたとき
C ループ X 論理	play sound giggle V until done
· · · · · · · · · · · · · · · · · · ·	
◆ 高度なブロック	

micro:bit v2 のみで動作







001 重ねて音を出す

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

Aボタンには従来の音を鳴らすブロック Bボタンには、v2用の音を鳴らすブロック (音楽ブロックの下のほうにあります) を設定しました。

Aのメロディが流れながら、Bのメロディも流れ ます。本体にスピーカーがついているのでオプ ションが不要になりました。



・・・使用するのは、ボタン・LED・スピーカー

2020/12/04作成



# 002 音がしたらビックリ

ボ	セ	L	出
通		オ	v2



micro:bit v2 のみで動作





## <u>002 音がしたらビックリ</u>

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2には、マイクがついているため、音センサー として使うことができます。 (入力ブロックの下のほうにあります) 大きな音をきっかけとして、LEDにビックリした 顔を表示します。

ボ	セ	L	出
通		オ	v2

・・・使用するのは、センサー (マイク)・LED

2020/12/04作成



003 タッチしたらビックリ











## <u>003 タッチしたらビックリ</u>

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2は、ロゴがタッチセンサーとなっています。 入力ブロックの下のほうにある「ロゴをタッチ」 ブロックを使います。 ロゴタッチをきっかけとして、LEDにビックリした 顔を表示します。

ボ	セ	L	出
通		オ	v2

・・・使用するのは、センサー (タッチ)・LED

2020/12/04作成

136

004 音のしきい値

ボ	セ	L	出
通		オ	v2



## micro:bit v2 のみで動作





<u>004 音のしきい値</u>

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2には、マイクがついているため、音センサー として使うことができます。 (入力ブロックの下のほうにあります) 大きな音をきっかけとして、LEDにハートマー クを表示します。

最初に「大きな音のしきい値」を大きくしておく ことで反応する音の大きさを変えることができ ます。

ボ	セ	L	出
通		オ	v2

・・・使用するのは、センサー(マイク)・LED

2020/12/08作成



138

005 静かになったら

ボ	セ	L	出
通		オ	v2



## micro:bit v2 のみで動作





005 静かになったら

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2には、マイクがついているため、音センサー として使うことができます。 (入力ブロックの下のほうにあります) 004「音のしきい値」の「小さな音」の使用例 です。

最初に「小さな音のしきい値」を設定し、まわり の音がそれより静かになったらニッコリした顔 が表示されます。



・・・使用するのは、センサー(マイク)・LED

2020/12/11作成







## 006 音量グラフ



micro:bit v2 のみで動作





<u>006 音量グラフ</u>

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2には、マイクがついているため、音センサー として使うことができます。 (入力ブロックの下のほうにあります) 入力された数値は「音量」として扱うことがで きるので、それをグラフで表示します。

ボ	セ	L	出
通		オ	v2

・・・使用するのは、センサー(マイク)・LED

2020/12/11作成







## 007 音量を送信



micro:bit v2 のみで動作





#### 007 音量を送信

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2には、マイクがついているため、音センサー として使うことができます。 (入力ブロックの下のほうにあります) 入力された数値は「音量」として扱うことがで きるので、それを無線で送信します。



・・・使用するのは、センサー(マイク)・通信 ※008 音量を受信とセットです。

2020/12/18作成



144
ボ	セ	L	出
通		才	v2

### 008 音量を受信



micro:bit v2 「007 音量を送信」とセットです。



### <u>008 音量を受信</u>

micro:bit v2のみで動作するサンプルでは ありませんが、007 音量を送信するサンプル がv2のみで動作するので、このサンプルもv2 対応です。

送られてきた数値(データとしては音量)をリ アルタイムでグラフ化します。



・・・使用するのは、LED・通信

※007 音量を送信 とセットです。

2020/12/18作成



ボ	セ	L	Ħ
通		オ	v2

### 009 タッチする楽器



### micro:bit v2 のみで動作







009 タッチする楽器

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2は、ロゴがタッチセンサーとなっています。 入力ブロックの下のほうにある「ロゴをタッチ」 ブロックを使います。 ロゴタッチをきっかけとして、音を出します。 タッチを離すと音が切れるので、打楽器的に使 うことができます。

ボ	セ	L	出
通		才	v2

・・・使用するのは、センサー (タッチ)・スピーカー

### 2020/12/22作成



ボ セ L 出 通 オ v2	010 メロディを打楽器に			
	黄色のロゴが タッチセンサー	■ JavaScript V		
		Q 最初だけ set P0 ◆ to touch mode capacitive ◆          ずっと         もし logo is pressed なら         play sound giggle ◆         でなければもし、端子 P0 ◆ がタッチされている なら ○         play sound happy ◆         でなければ         マスではければ		
		ボタン A ▼ が押されたとき すべての音を停止		

micro:bit v2 のみで動作





010 メロディを打楽器に

micro:bit v2のみで動作するサンプルです。 ・・・v1.5 ではエラーが出ます。

v2は、ロゴがタッチセンサーとなっています。 入力ブロックの下のほうにある「ロゴをタッチ」 ブロックを使います。

また、PO端子をタッチセンサー(これまではPO とGNDを両方さわる必要があった)にするオプ ションが、

・高度なブロック→入出力端子→その他

にある「set PO to touch mode capacitive」 にあります。

ロゴタッチとPO端子で、メロディを鳴らします。 Aボタンを押すと、音が切れるようにして打楽器 的に使えるようにしました。

ボ	セ	L	出
通		オ	v2

・・・使用するのは、センサー (タッチ)・スピーカー

2020/12/28作成

# いいい、「サンプル プログラミング集」 microPython編

大分県立芸術文化短期大学 情報コミュニケーション学科 非常勤講師 望月陽一郎 mochizuki@oita-pjc.ac.jp 作品のクレジットを表示 すること・営利目的での 利用をしないこと・元の 作品を改変しないこと (作者の許可を得ている 場合は除きます)

Created by ©Yoichiro.Mochizuki 2018-2021



## <u>目次</u>

- 001 「HelloWorld」と「ハートマー ク」を表示する
- 002 LED上に「O」を表示する
- 003 傾きを表示する
- 004 0から9まで表示する



## micro:bit microPythonプログラミングサイト

## https://microbit.org/ja/code/

⊖micro:bit	プログラムしましょう フ	アイデア はじめよう 教える 購入 日本	語 🖲	
	プログラムで想像	力を発揮する		
Ģ	BBC micro:bit は、ブロック、JavaScript、P か? もしBBC micro:bitを使ったことがなければ、 単にはじめる方法).	ythonを使ってプログラムできるって知ってました 試してみてくださいクイックスタートガイド(簡		
	ere (1.000 (1.000)) O O Marce Perto Q Perto Q	MakeCodeエディター Microsoft提供のMakeCodeエディターを使え	エアィッテへのアッセスに同趣があったり、     エディターをオフラインで使いたい場合は、 FAOをご確認ください。	
	C Longe C L		マログラムしましょう リファレンス Lessons	
		Pythonエディター Pythonエディターは自分のプログラミングの スキルをさらに向上させたい人に最適です。 違ばれたプログラム例とあらかじめ用意され た様々な画像や音楽は、プログラミングの手 助けになります。グローバしなPythonコミュ ニティーの成果物によって動いています。	Comicrobil	
		Find out more about the left of the left version 2 of the near Editor. プログラムしましょう リファレンス・		
	オンラインエラ	ディター ディアー <sup>Prevention</sup> が App Store の詳細に ウェア)	エクマ アプリでは、Bluetoothを使用してワイヤレスであなたのmicro:bitに ムを送り込みます。ケーブルは必要ありません!使用方法について ついては、このページを見てください。 アプリケーション(ソフト ・	

micro:bit プログラミングサイト

https://microbit.org/ja/code/

のPythonエディターでオンラインプログラミングします。

・Download→プログラムをダウンロードして、 micro:bitへ送る

・Load/Save→\*.py (Mu編集用ファイル)やプロ グラムを読み込み・保存できる



### micro:bit microPythonプログラミングツール Mu

## https://codewith.mu/





micro:bit プログラミングツール Mu サイト <u>https://codewith.mu/</u> からダウンロード・インストールします。

※オンラインエディターとの違い

- ・転送→プログラムを接続されているmicro:bitへ
- ・保存→\*.py (Mu編集用ファイル)



## 001 「HelloWorld」と「ハートマーク」を表示する

ボ セ L 出 通 入 オ	https://python.microbi1	<u>.org/v/2.0</u>
🖸 micro:bit		
Download Connect	Load/Save Open Serial Open Serial Help @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @	
<pre>3 4 5 while True: 6 display.scrol1 7 display.show(1 8 sleep(2000)</pre>	l('Hello, World!') Image.HEART)	
9		

#・・・プログラムに関係しないメモ from microbit import \*・・・micro:bit用モジュールの読み込み while True・・・ずっと繰り返す display.scroll('')・・・LEDに文字列を表示 display.show(Image.HEART)・・・LEDに「ハート」を表示 sleep(2000)・・・2000ミリ秒=2秒停止

ダウンロード→micro:bitに転送



λ

才

出

ボセ

通

IMA	GEO	)種類
-----	-----	-----

Image.HEART ハート Image.HEART\_SMALL 小ハート Image.HAPPY 笑顔 Image.SMILE 笑ったロ Image.SAD 悲しい顔 Image.CONFUSED 困り顔 Image.ANGRY 怒った顔 **Image.ASLEEP 眠っている顔** Image.SURPRISED びっくり顔 Image.SILLY 変顔 Image.FABULOUS 素晴らしい顔 Image.YES はい(チェック) Image.NO いいえ(×) Image.CLOCK12, Image.CLOCK11, Image.CLOCKI0, Image.CLOCK9, Image.CLOCK8, Image.CLOCK7, Image.CLOCK6, Image.CLOCK5, Image.CLOCK4, Image.CLOCK3, Image.CLOCK2, Image.CLOCKI 〇時(短針を表示) Image.ARROW\_N, Image.ARROW\_NE, Image.ARROW\_E, Image.ARROW\_SE, Image.ARROW\_S, Image.ARROW\_SW, Image.ARROW\_W, Image.ARROW\_NW 矢印表示(上を北として) Image.TRIANGLE 三角 Image.TRIANGLE\_LEFT 左三角 (RIGHTはない)

Image.CHESSBOARD チェス盤 Image.DIAMOND ダイヤ Image.DIAMOND\_SMALL 小ダイヤ Image.SQUARE 四角 Image.SQUARE\_SMALL 小四角 Image.RABBIT うさぎ Image.COW 牛 Image.MUSIC\_CROTCHET 四分音符 Image.MUSIC\_QUAVER 八分音符 Image.MUSIC\_QUAVERS つないだハ 分音符 Image.PITCHFORK フォーク Image.XMAS クリスマス Image.PACMAN パックマン Image.TARGET ターゲット Image.TSHIRT Tシャツ Image.ROLLERSKATE ローラースケー  $\mathbf{F}$ Image.DUCK アヒル Image.HOUSE 家 Image.TORTOISE カメ Image.BUTTERFLY チョウ Image.GHOST おばけ Image.SWORD 剣 Image.GIRAFFE キリン Image.SKULL どくろ Image.UMBRELLA 傘 Image.SNAKE ヘビ

文部科学省 高等学校情報科「情報 I 」教員研修用教材より ※オプション例は2019/9 別途調べました。



### 002 LED上に「O」を表示する



OO=Image("09990:" ・・・OがLED消灯、9が最も明るい ・・・ "09990:")・・・変数として設定 display.show(OO)・・・変数を表示 display.clear()・・・LED表示を消す



## 【説明】 002 LED上に「O」を表示する

## ・・・順次の例として、LEDで〇を1秒間点灯したあ と消灯するプログラムです。

ボ	セ	L	出
邇	く	¥	

文部科学省 高等学校情報科「情報 I」教員研修用教材より



### 003 傾きを表示する



reading=accelerometer.get\_x()

- ・・・変数に加速度センサーの値を入力
- ・・・>100 右に傾き
  - (<100 左に傾き =100 左右水平)

※accerometer.get\_y()
・・・>100 手前に傾き <100 向こうに傾き</p>



## 【説明】 003 傾きを表示する

・・・ 分岐の例として、micro:bitの加速度センサ が右に傾いているデータを検出すれば、LEDに 「R」が点灯し、そうでなければLEDに「-」が点灯 するプログラムです。

文部科学省 高等学校情報科「情報 I 」教員研修用教材より

### ※温度センサーの例



2020/	/	13	作成
-------	---	----	----

ボ	セ	L	出
通	く	¥	



### 004 0から9までを表示する



### for i in range(10)・・・i を0~9までカウントしながら繰り返し display.show(i)・・・i に入っている値を表示



## 【説明】 004 0から9までを表示する

・・・反復の例として、Oから9までの数字を順に表 示するプログラムです。

ボ	セ	L	出
通	く	オ	

文部科学省 高等学校情報科「情報 I」教員研修用教材より



# いいい、「サンプル プログラミング集」 Pythonプログラミング編

大分県立芸術文化短期大学 情報コミュニケーション学科 非常勤講師 望月陽一郎 mochizuki@oita-pjc.ac.jp 作品のクレジットを表示 すること・営利目的での 利用をしないこと・元の 作品を改変しないこと (作者の許可を得ている 場合は除きます)

Created by ©Yoichiro.Mochizuki 2018-2021



## <u>目次</u>

関数など

- 001 温度を表示する 002 明るさを表示する 003 ボタンで温度を表示する
- 004 ボタンで明るさを表示する

条件(if文)

006 暗くなるとLEDが点く

### 音

007 Happy Birthdayを鳴らす

### 複数の条件

017 コンパス (Compass)

### 無線

0182台で送受信

### 傾き

019 水準器

020 傾きメーター

#### 乱数と繰り返し

052 ランダム数値表示

053 じゃんけん



## micro:bit プログラミングサイト

https://makecode.microbit.org/#editor





micro:bit プログラミングサイト

https://makecode.microbit.org/#editor

にアクセスしてプログラミングします。

- ・Python画面でプログラミング。
- ・確認用シミュレータで確認する。
- ・USB接続したmicro:bit にプログラムをコピー。
- micro:bit 本体で動作させる。

(・micro:bitを接続せずにプログラムをダウン ロードすることもできます。)

※ダウンロード保存しておいたプログラムを読み込 んで修正することもできます。



```
001 温度を表示する
【コード】
def on_forever():
basic.show_number(input.temperature())
basic.forever(on_forever)
```

#### 【解説】

- ・・・def on\_forever(): ・・・関数「ずっと実行」 実行内容
- ・・・basic.show\_number ・・・数値を表示
- ・・・input.temperatire() ・・・温度の値
  - ・明るさの値 input.light\_level()
  - ・磁力の値 input.magnetic\_force(Dimension.X)
- ・・・basic.forever(on\_forever) ・・・関数on\_forever を呼び出し

```
002 明るさを表示する
```

【コード】

```
def on_forever():
```

```
led.plot_bar_graph(input.light_level(), 255)
```

basic.pause(1000)

```
basic.forever(on_forever)
```

#### 【解説】

```
・・・led.plot_bar_graph ・・・LEDにグラフを表示 255は最大値(変更可)
・・・basic.pause() ・・・一時停止 1000は停止時間(ミリ秒)
```

```
003 ボタンで温度を表示する
【コード】
def on_button_pressed_b():
basic.show_number(input.temperature())
input.on_button_pressed(Button.B, on_button_pressed_b)
```

【解説】

```
・・・def on_button_pressed_b(): ・・・関数「Bボタンを押すと実行」
```

...input.on\_button\_pressed(Button.B, on\_button\_pressed\_b)

・・・呼び出し

・・・A+Bボタンは、Button.AB on\_button\_pressed\_ab

```
004 ボタンで明るさを表示する
```

【コード】

def on\_button\_pressed\_a():

led.plot\_bar\_graph(input.light\_level(), 255)

basic.pause(1000)

```
basic.clear_screen()
```

input.on\_button\_pressed(Button.A, on\_button\_pressed\_a)

### 【解説】

・・・basic.clear\_screen() ・・・画面を消去



006 暗くなるとLEDが点く 【コード】 def on\_forever(): if input.light\_level() < 15: basic.show\_leds(""" # """) elif input.light\_level() < 30: basic.show\_leds(""" . . . . . . # # # . . # # # . . # # # . . . . . . """) else: basic.show\_leds(""" . . . . . . . . . . . . . . . . . . . """) basic.forever(on\_forever)

#### 

•••. 消灯 # 点灯



```
007 Happy Birthdayを鳴らす
【コード】
def on_button_pressed_a():
  music.play_tone(392,750)
  music.play_tone(392,music.beat(BeatFraction.QUARTER))
  music.play_tone(440,music.beat(BeatFraction.WHOLE))
  music.play_tone(392,music.beat(BeatFraction.WHOLE))
  music.play_tone(523,music.beat(BeatFraction.WHOLE))
  music.play_tone(494,music.beat(BeatFraction.DOUBLE))
  music.play_tone(392,750)
  music.play_tone(392,music.beat(BeatFraction.QUARTER))
  music.play_tone(440,music.beat(BeatFraction.WHOLE))
  music.play_tone(392,music.beat(BeatFraction.WHOLE))
  music.play_tone(587,music.beat(BeatFraction.WHOLE))
  music.play_tone(523,music.beat(BeatFraction.DOUBLE))
input.on_button_pressed(Button.A, on_button_pressed_a)
```

### 【解説】

```
    ・・・music.play_tone(440,music.beat(BeatFraction.WHOLE))
    ・・・440 音の高さ(ラ)
    392 ソ 440 ラ 494 シ 523 上のド 587 上のレ など
    ・・・WHOLE 音の長さ一拍(4分音符)
    DOUBLE 二拍(2分音符)
```

```
QUARTER I/4拍(16分音符)
```

```
HALF I/2拍(8分音符)
```

```
750 数值(付点4分音符)
```



### 013 コンパス (Compass)

【コード゜】

def on\_forever():

```
if 315 < input.compass_heading() or input.compass_heading() < 45:
    basic.show_string("N")
```

```
elif 45 < input.compass_heading() and input.compass_heading() < 135:
basic.show_string("E")
```

```
elif 135 < input.compass_heading() and input.compass_heading() < 225:
    basic.show_string("S")
```

```
elif 225 < input.compass_heading() and input.compass_heading() < 315:
    basic.show_string("W")
```

else:

```
basic.clear_screen()
```

```
basic.forever(on_forever)
```

【解説】

- ・・・or・・・または
- ・・・and ・・・かつ
- ・・・basic.show\_string() ・・・文字列の表示



### 014 2台で送受信

#### 【コード】

def on\_received\_number\_deprecated(receivedNumber):

if receivedNumber == 0:

basic.show\_string("Receive!")

basic.pause(1000)

basic.clear\_screen()

radio.on\_received\_number\_deprecated(on\_received\_number\_deprecated)

def on\_button\_pressed\_a():

radio.send\_number(0)

input.on\_button\_pressed(Button.A, on\_button\_pressed\_a)

radio.set\_group(1)

【解説】

- ・・・radio.set\_group()・・・無線グループの設定
- ・・・radio.send\_number()・・・数値を送信

・・・radio.on\_received\_number\_deprecated

- (on\_received\_number\_deprecated) ・・・数値を受信したら
- $\cdots on\_received\_number\_deprecated(receivedNumber):$ 
  - ・・・受信した数値の処理関数



174





```
044 ランダム数値表示
【コード】
def on_button_pressed_a():
  basic.show_number(randint(0, 9))
input.on_button_pressed(Button.A, on_button_pressed_a)
                             【解説】
045 じゃんけん
                               ・・・randint(0,9) ・・・0~9の間で乱数(ランダムに)
【コード】
乱数 = 0
def on_button_pressed_a():
  global 乱数
 乱数 = randint(0, 2)
                             【解説】
  basic.show_number(乱数)

    ・・・乱数 = 0 ・・・変数に日本語が使える。

  if 乱数 == 0:
                              ・・・if 乱数 == 0: ・・・== 等しい (= は右から左への代入)
    basic.show leds("""
      ####.
      ####.
      #####
      ####.
      .###.
      """)
  elif 乱数 == 1:
    basic.show_leds("""
      . # . # .
      . # # . .
      ####.
      ####.
      . # # # .
      """)
  else:
    basic.show_leds("""
      . . . . .
      ####.
      # # # # #
      ####.
      .###.
      """)
input.on_button_pressed(Button.A, on_button_pressed_a)
```



コードを打つ時に気をつけること

・縦の列をそろえます。
 def OO():

basic.… ↑

ここのタブは、「半角スペース4つ」。Tabキーは使いません。

・「,」の次は、見やすくするため半角スペースを入れます。

・「=」「+」などの両側には、半角スペースを入れます。

・問題があるときは、画面の下に

「Problems 数字」のように表示されます。









### 

作品のクレジットを表示 すること・営利目的での 利用をしないこと・元の 作品を改変しないこと (作者の許可を得ている 場合は除きます)

```
2018/10/19 第1.0版
2018/10/24 修正
2018/11/02 第1.1版
2018/11/03 修正
2019/03/12 第1.98版
2019/05/05 第2.0版
2019/11/12 第2.1版
2019/12/30 第2.11版
2020/01/13 第3.0版
2020/08/25 第3.5版
2020/08/30 修正
2020/01/04 第4.0版
```

