

www.M5Stack.com



Shenzhen Mingzhan Information Technology Co., Ltd
RM106, Block A, MingLiang Technology Park, Zhuguangbei Road, Nanshan,
Shenzhen, China.
E-mail: Support@m5stack.com
add: (日本語訳: 秋田統一)



M5StickC

初心者でも簡単にESP32でIoT
アプリケーションを開発

革新的なAll-in-OneのコンパクトなIoTソリューション



www.M5Stack.com

M5STICK

目次

はじめに

UIFlow IDE	05
UIFlow入門	07
操作メニュー	09
プログラミングの手順	10
プログラムの保存と読み出し	11

カリキュラム

HelloWorld	15
時計をつくる	21
グラフィック描画	27
加速度センサを使ってみる	31
デジタル名刺	35
玉よけゲーム	38
デジタル音楽	44
デジタル温度計	52

付録

Wi-Fiの設定	59
ファームウェアの更新	63

さいごに・・・

BEFORE WE START:

OPERATIONS MENU

UIFLOW MENU BAR

「始める前に…」

アイデアを素早く形に

UIFLOW PROGRAMMER

BEFORE WE START

INTRODUCTION

M5StickC



はじめに



UIFlow IDEについて



UIFlowを使ってみよう



できること

M5StickC

M5StickC

M5StickC

M5StickC

はじめに

1、M5StickCとは?

M5StickCは、親指サイズでプログラミングができる小さなコンピュータ(マイコン)です。Wi-FiとBluetooth通信機能があるESP32マイコンと、6軸の加速度センサ、液晶ディスプレイ、LEDと赤外線送信機がついています。本体には2つのボタンと電源ボタン、それにセンサなどをつなぐGrove互換コネクタがあります。簡単なリモコンデバイスからIoTシステムまで、簡単にいろいろ作ることができます。プログラミングの経験がなくても大丈夫、この小さなデバイスではじめてみましょう。

ハードウェアの概要



機能の詳細

CPU

コンピュータには計算や処理を行う、コンピュータの頭脳とでも呼ぶべきCPU(中央演算装置)があります。コンピュータには2種類のメモリがあります。

RAM:RAMは「揮発性」、つまり電源をオフにすると記憶内容が消えてしまうメモリです。プログラムからデータを高速に読み書きできます。

フラッシュ(RAM):フラッシュはプログラムを保存しておくメモリで、電源をオフにしても内容は消えません。

入力

コンピュータが、私たちの物理世界からの情報を受け取るのが入力です。例えばマウス、キーボード、タッチパネル、マイクなどがあります。

出力

コンピュータが、私たちの物理世界に情報を送り出すのが出力です。例えばスピーカー、液晶ディスプレイ、プリンタなどです。

2、M5StickCとパソコンの比較

	パソコン	M5StickC
CPU	✓	✓
メモリ	✓	✓
入力	キーボード、マウス	ボタン、加速度センサ、マイク
出力	ディスプレイ、スピーカー	LED、液晶ディスプレイ、赤外線送信機

UIFlow IDE

UIFlowは、M5Stackシリーズをプログラミングするための環境で、ブロックを使ったビジュアル・プログラミングと、MicroPythonでのプログラミングができます。プログラミングの初心者でもプログラミングの基本を学んだり、複雑なプログラムを簡単に書くことができます。ここでは、UIFlowの設定と、LEDを点滅させる簡単なプログラム(いわゆる「Lチカ」)をやってみましょう。

★注:M5StickCは出荷時のテストプログラムを読み込みます。UIFlowIDEでプログラミングおよび開発する前に、M5StickCのUIFlowファームウェアを書き込みます。詳細については、付録の「ファームウェアの更新」の章を参照してください。

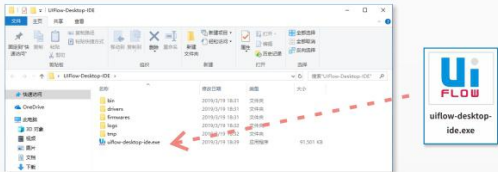
01 ダウンロード

https://m5stack.com へアクセスし、上部のSoftware→Downloadへ進みます。自分のパソコンにあわせてUIFlow-Desktop-IDEをダウンロードします。



02 解凍して実行

ダウンロードしたzipファイルを解凍し、中にあるuiflow-desktop-ideアイコンをダブルクリックして起動します。



03 ドライバのインストール

プログラムが起動すると、ドライバがインストール済みかチェックします。インストールされていない場合は、「Install Now」の手順によってインストールします。



注意
M5StickCは、Windows10、Linux、Mac (HighSierra以降)であればドライバは不要です。その他のOSではFTDIドライバとCP210xドライバのインストールが必要です。

04 デバイスを接続する

UIFlowのインストールが済んだら、付属しているUSB-CケーブルでM5StickCをパソコンに接続します。



05 USB Programming mode

本体左側にあるボタンを2秒間長押しして電源をオンにします。UIFlowのロゴが表示されたら、すぐに正面のAボタンを押し、設定メニューに入ります。設定メニューの項目は本体右側のBボタンで移動し、正面のAボタンで選択します。ここでは、プログラミング・モードとWi-Fiの設定ができます。ここでは「USBモード」を選択しておきます。パソコンのUIFlowでは、M5StickCが接続されている「COMポート」を選択します。MacやLinuxでは、「tty'+英数字の名前になるはず。



06 はじめてのプログラミング

ブロック一覧から「Hardware」(ハードウェア)を選び、LEDの中の「LED ON」を選び、画面右側のプログラミング画面にドラッグして持っていく。「Setup」の下にはめます。そして、右上の三角マークの「再生」ボタンを押して、M5StickCにプログラムを転送して実行させます。M5StickCの赤LEDが点灯するはず。



07 ファームウェアの更新

M5StickCには、UIFlow用のファームウェア(内部プログラム)が書き込まれています。ファームウェアは随時、機能追加やバグ修正で更新されます。UIFlowプログラムとファームウェアは最新のものに更新しておきましょう。更新の手順は最後の「付録」に載っています。

UIFlow入門

Blockly・Python

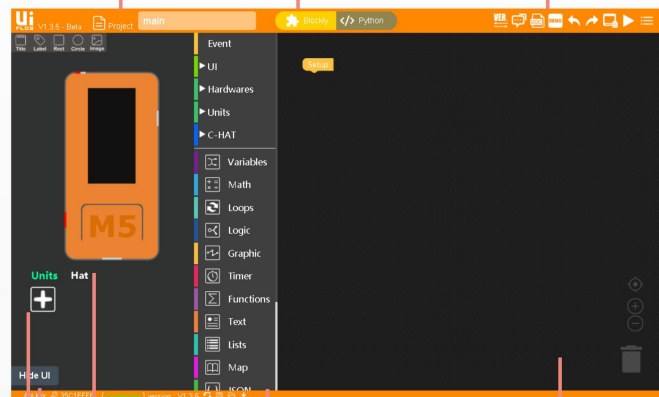
Blockly(ブロック型言語)で書いたプログラムをPython言語のプログラムに切り替えて確認したり編集したりできます。

メニュー

ここから、ユーザフォーラム、各種ドキュメント、例題、Undo(取り消し)やRedo(やり直し)、M5StickCへの書き込みと実行、その他各種設定ができます。

プロジェクト名

自分のプログラムに名前をつけてここに書けます。パソコンに保存したりM5StickCで実行されるときには、その名前が使われます。



UIの確認

Text(文字列)や図形などをM5StickCのディスプレイに表示させ、UI(User Interface、コンピュータの操作に関わる部分)をつくります。Python言語のプログラムが自動的に生成されます。置いたテキストなどは自由に位置を変えたり大きさを変えたりできます。

プログラミング・ブロック

プログラミングに使うブロックのリストです。

プログラミング画面

ここにブロックをドラッグで移動し、組み合わせてプログラミングします。

UIを隠す

UI編集画面を隠して、プログラミング画面を広くできます。

Unit(ユニット)

各種センサなどのUnitをここから追加して、接続するポートを指定します。

プログラミング・ブロック

ここには、いろいろなプログラミング・ブロックがあって、プログラミングに使えます。M5StickCに載っているLEDなどを直接制御するものから、数値演算、プログラムの流れの制御などのためのブロックがあります。まずは簡単なブロックから順に使っていきましょう。

イベント

繰り返しや、ボタンを押したときの動作などを設定できます。

ハードウェア

M5StickCに載っているLED、加速度センサ、電源制御などの要素(ハードウェア)を制御します。

Unit

Unitを追加すると、そのUnitに応じた動作のブロックが現れます。

Math(数学)

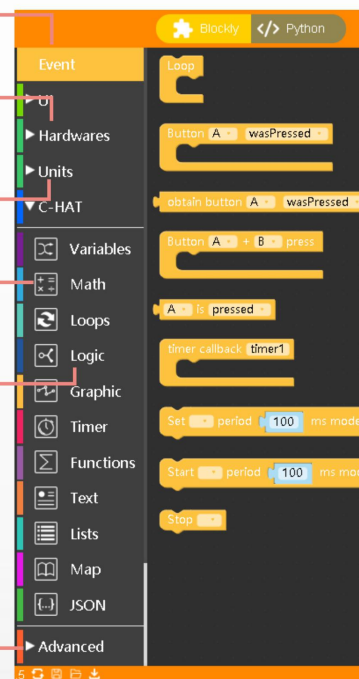
プログラミングでは数学は大切です。簡単な数値演算から複雑な計算まで、いろいろな数学処理のブロックがあります。

Logic(論理)

プログラミングに必要な、条件に応じた動作の制御などの論理を設定できます。

Advanced(上級者向け)

上級者向けのブロック。ネットワーク通信、デジタル・アナログ入出力などがあります。



メニュー

メニューの詳細

メインメニューには、フォーラム、各種ドキュメント、プログラム例、Undo (取り消し)、Redo (やりなおし)、プログラムの書き込みや実行、各種設定ができます。



プログラミング・ブロックの使い方

Setup (セットアップ)

Setupブロックは、プログラムの実行が始まる時に最初に実行されます。



Loop (ループ)

Loopブロックは、中に置いたブロックをずっと繰り返し、電源を切りセットするまで実行します。

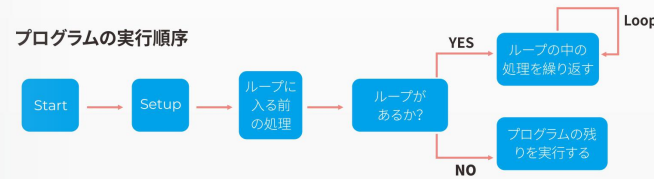


Wait (待機)

Waitブロックは、指定した時間だけブロックの実行を止めて待ちます。これがないと、表示されるデータがすぐに消えてしまっ見えなといった事が起こってしまいます。



プログラムの実行順序



プログラミング・ブロックの接続

UIFlowではBlocklyというブロック型のプログラミング言語を使います。Blocklyはジグソーパズルに似ていて、はまるところに移動させると色が変わって、かちとはまり、正しく動作するようになります。色が変わらなかったら、ちゃんとはまっていないか、その動作ができないかのどちらかです。

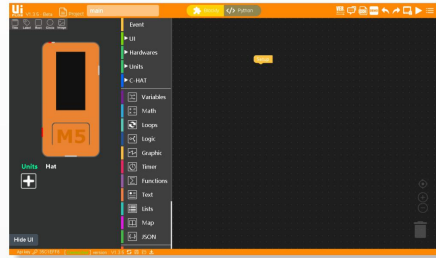


UIFlowのちょっとしたテクニック

ブロックをコピーするには、マウスの左ボタンをダブルクリックするか、右クリックから複製(Duplicate)を選びます。ブロックを消すには、ブロックが並んでいるリストに戻るか、プログラミング画面の右下のゴミ箱に移動させます。

プログラムの保存と転送

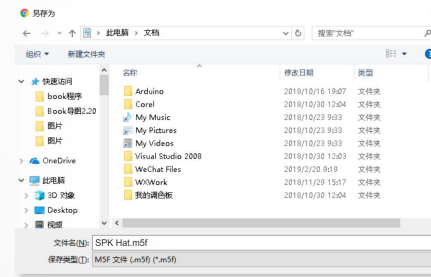
いよいよプログラミングのレッスが始まりますが、これからはいろいろなプログラムを作っていくことになります。複雑ですぐには終わらないこともあるでしょう。そういうときは、作ったプログラムをパソコンに保存しておいて、日を改めて続きをやったり実行させたりできます。突然パソコンがフリーズしてしまっても困らないように、こまめに保存しましょう。メニューの保存(Save)から保存でき、開く(Open)から保存されたプログラム(.m5fファイル)を読み込めます。



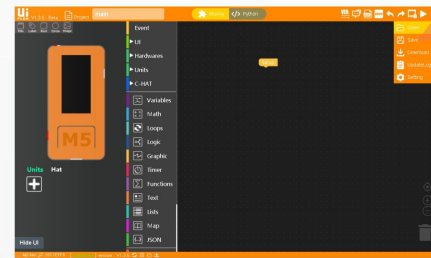
01 UIFlowのWebページを開きます
(flow.m5stack.com)



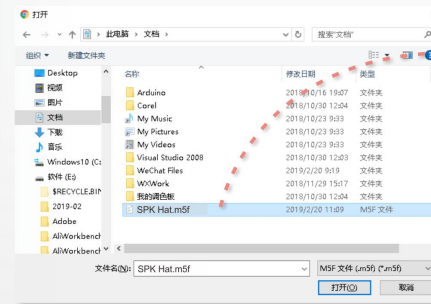
02 プログラムを書いたら、保存(Save)から保存しましょう。



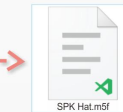
03 m5fファイル(拡張子がm5f)として保存されます。



04 m5fファイルは、メニューの開く(Open)から読み込めます。



05 前に保存した.m5fファイルを指定します。



CURRICULUM:

DIGITAL MUSIC

FLASHY GRAPHICS

「カリキュラム」

アイデアを素早く形に

TRAFFIC LIGHT

DIGITAL NAMECARD

DODGING GAME

M5StickC

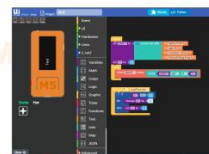
M5StickC



デジタル音楽



グラフィック描画



デジタル名刺



ドッジボールゲーム

M5StickC

M5StickC

M5StickC

M5StickC

1 HELLO WORLD

UIFlowの画面の左側には「UI Manger (UIマネージャ)」が表示されていて、ここに文字列やグラフィックが表示されます。

このレッスンでは、M5StickCのプログラムを書いて実行できるようになります。簡単な文字列として「Hello World」と表示させてみます。

1-1 超簡単プログラミング

“Hello World”は、昔からある「はじめてのプログラミング」の定番で、画面に「Hello World」と表示させるものです。簡単そうに見えますが、初めて使うプログラミング言語やコンピュータ(M5StickC)を使うのには最適です。これができれば、これを応用してさらに高度なプログラミングへ進めます。



1、ラベルの追加

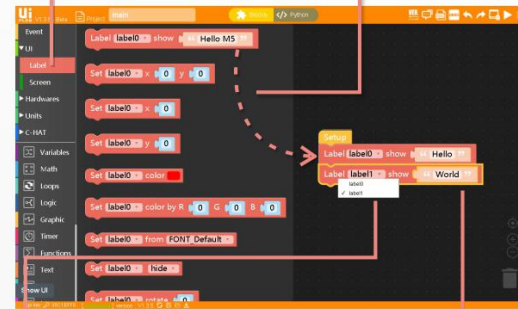
UIマネージャを使うと、画面に文字を簡単に表示することができます。“Label”アイコンを画面上にドラッグするだけです。

- 01 上にある“Label”をUIマネージャの画面の適当な場所にドラッグします。



ブロッカー一覧の“UI”をクリックすると、その中に“Label”というグループがあり、テキスト(文字列)に関するいろいろなブロックがあります。“Label[ラベル名] Show [表示する文字列]”というブロックをプログラミング画面へ移動させて、Setupブロックにはめます。これでできあがりです。

- 02 「Label」ブロックをクリックし、「Label Show」ブロックを選ぶ
- 03 それをプログラミング画面へドラッグし、Setupブロックにはめる



- 04 ラベルが複数あるときは、ラベルごとに名前(番号)がつくので、ドロップダウンリストから、表示させたいLabelの名前(番号)を選びます。
- 05 表示させる文字列を“Hello World”に変更します。

これで最初のプログラムが完成です。簡単でしたね。早速M5StickCで実行させてみましょう。パソコンとM5StickCをUSBケーブルで接続し、前の章で説明した手順で実行させてみましょう。



- 07 右上にある実行ボタンを押してプログラムを実行させます。

- 06 M5StickCがちゃんと接続されていれば、個々の表示が緑の“connected”になります。そうならないときは、更新マークを押してみましょう。

2、UI要素のプロパティ(属性)

UIマネージャにLabelなどのUI要素を追加したら、それをクリックすればその要素のプロパティ(属性)メニューが表示されます。ここで、大きさや位置、色などを変更できます。

名前
LabelなどのUI要素の名前を変更できます(画面に表示される文字列ではありません)

x/y
画面上的位置(x座標とy座標)を変更できます。M5StickCの画面サイズは80×160画素です。

色
ここをクリックすれば、カラーバーかパレットから色を設定できます。

テキスト(文字列)
画面に最初に表示される文字列を設定します。

フォント
ここから、表示する文字のフォント(文字の形)を選択して設定できます。

回転(rotation)
指定した角度だけ回転させられます。

レイヤー(layer)
レイヤーを使うと、画面上的文字や図形の重なり具合(手前や奥)を指定できます。

3、UI要素の追加と削除

UIマネージャの上には、Label以外にもいくつかの要素があって画面を構成できます。

UI要素をUIマネージャの画面に置くと、そのUI要素に関するブロックがブロックリストの"UI"グループに現れます。

UI要素を間違えて追加してしまったり、もう使わなくなったら、UIマネージャの上にあるゴミ箱にドラッグすれば削除できます。

4、UI要素のプログラミング

UI要素にあわせて現れるブロックを使えば、UI要素に関するいろいろなプログラムを書けます。

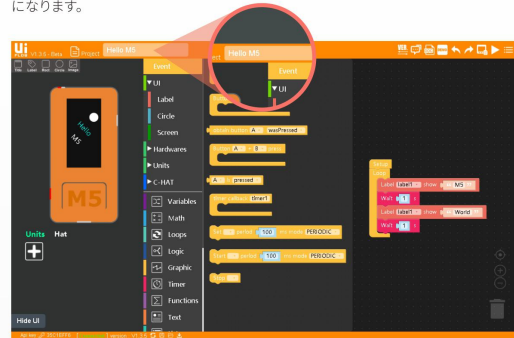
01 "Label Show"ブロックを2個、プログラミング画面に置き、それぞれ別の文字列を表示するようにしておきます。

02 Labelブロックの間には、Wait(待機)ブロックを置きましょう。これで画面に文字列が表示された状態で少しプログラムの実行が止まり、文字列を見ることができます。

03 繰り返し実行させたい部分は、"Event"の中にあるLoopブロックの中に置きます。

5. プログラムの名前

ここにプログラムの名前を入力できます。パソコンやM5StickCにプログラムをダウンロードしたとき、この名前になります。



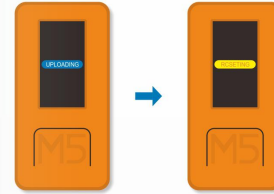
6. プログラムの実行とダウンロード

プログラミングについて勉強していく過程では、プログラムをしょっちゅう実行させることになります。再生ボタン(三角ボタン)を押すと、プログラムを実行できます。ただしM5StickCの電源を切ると、プログラムは消えてしまいます。M5StickCにプログラムを保存して、パソコンにつながなくても実行させたい場合は(友達に見せるなど)、ダウンロードを使います。

右上の設定(歯車のアイコン)をクリックし、メニューの中から"Download(ダウンロード)"を選びます。



ダウンロードを選んでダウンロードが始まると、M5StickCの画面には"Uploading" (アップロード中)と表示され、終わると表示が"Resetting" (リセット中)となってM5StickCはリセットされ、プログラムの実行が始まります。



7. App List (プログラム一覧)

M5StickCにダウンロードしたプログラムは、一覧から選んで実行することができます。M5StickCの起動時(リセット直後)に正面のM5ボタンを押すと、メニューが現れ、本体右側のボタンで項目を選べます。そこで"AppList"を選び、M5ボタンを押すと、ダウンロード済みのプログラムの一覧が表示されるので、実行したいプログラムを選び、M5ボタンを押せばそのプログラムが実行されます。



注意: 右ボタンを長押し(3秒間)すると、そのプログラムが消去されます。またダウンロードするプログラムと同じ名前のプログラムがある場合は、新しいものの上書きされます。電源オナまたはリセット後にメニューから選択しない場合は、最後にダウンロードしたプログラムが実行されます。

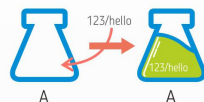


2 時計をつくる

ここでは、変数と論理(条件)、ボタン操作イベントについて学び、それらを使って時計をつくってみます。
変数の作り方と使い方、それと条件の入れ子構造について学びます。

2-1 変数について

(1) 変数は、ビンのようなものです。ビンに、例えば"A"という名前をつけて、そこに数値や文字を入れておきます。



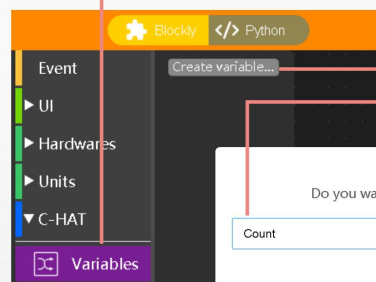
(2) ビンに別の値を入れると、もとの値は消えて、その値に書き換わります。



ある変数(ビン)の値を、別の変数に入れることもできます。この場合、元の変数の値が、入れた先の変数の値にコピーされます。

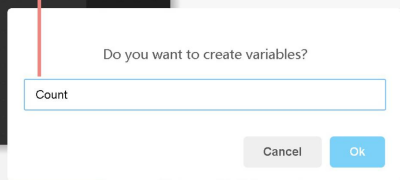


01 ブロックメニューの"Variable"
(変数)をクリック

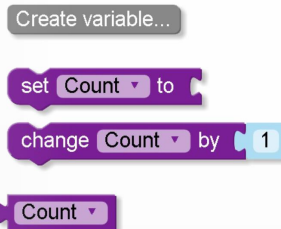


02 "Create Variable" (変数を作る)
をクリックして、新しい変数
を作ります

03 自分で変数に名前をつけて
OKをクリック

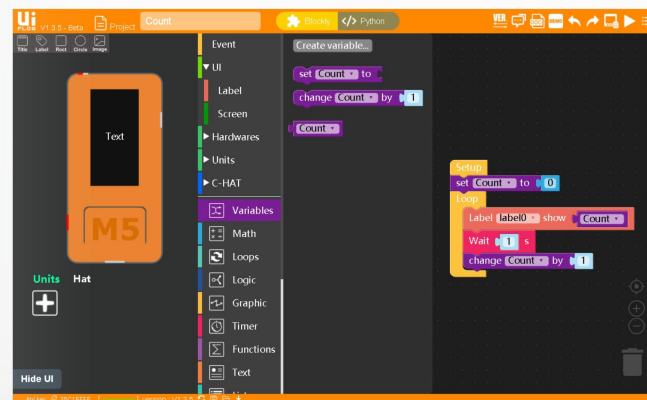


新しい変数をつくったら、その変数のブロックが現れます。



1. 数をカウントするプログラム

"count" (カウント) という名前の変数をつくって、その値を0にする(代入する)ブロックをSetupにはめます。
Labelを追加して、そこに変数countの値を表示させて、これを1秒毎に値を増やしていきます。



これでcountの値は1秒ごとに1ずつ増えていくので、秒で数える時計(ストップウォッチ)になります。60秒=1分、60分=1時間、24時間=1日、ですね。

2-2 時計をつくる

1、画面の構成

Labelを3個(秒、分、時)をつくります。とりあえずすべてのLabelの表示は"0"にしておきましょう。好きなフォントや色を設定しておきましょう。わかりやすいように、それぞれのLabelの名前を、"h"(hour=時間)、“m”(minute=分)、“s”(second=秒)に変えておきます。それぞれの数値の間には、":"(コロン)を表示するLabelを置いておきましょう。



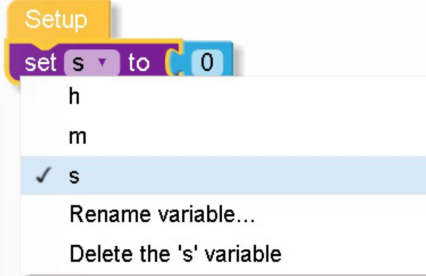
2、プログラムを書く

01 "Create Variable" (変数をつくる) をクリックします

02 時、分、秒の値を入れておく3つの変数(名前はh、m、s)をつくります。



Setupの下にLoopブロックをはめ、その中に、変数の値を設定するブロックをはめて変数"h"を0に設定しておきます。同じように変数"m"と"s"も0を入れておきます。

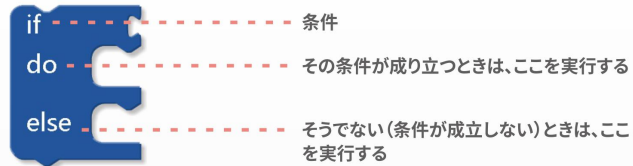


- (1) 3つの変数の値を0にするブロックの下に、もう1つLoopブロックをはめます。
- (2) 時・分・秒を表示するLabelを追加しておきます。
- (3) 変数"s"の値を1増やすブロックをはめ、その後で1秒間待つようにします。
- (4) Loopブロックや値を増やすブロック、Labelを変える部分の処理にかかる時間はほぼゼロなので、これでほぼ1秒ごとに変数"s"の値が1ずつ増えていきます。



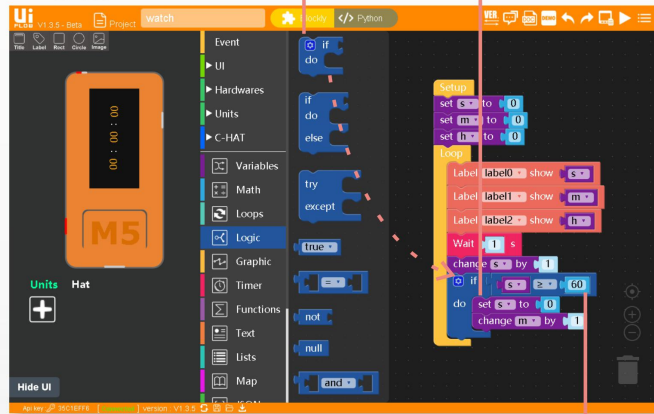
時、分、秒の値を表示するのにLabelを使っています。これで1秒ごとに「秒」は増えていきますが、秒が60になったら、分を1進めなければなりません、これはどのようにやるのでしょうか?これには、新しく「if」ブロックを使います。

“if”ブロックは、条件に応じてプログラムの実行を変えることができます。ifで指定した条件が成立したときには、“do”のところにはまっているブロックが実行され、成り立たないときは“else”のところのブロックが実行されます。



現在の時刻にあわせて表示を更新するためには、簡単な論理(仕掛け)のできる、この“if/do”ブロックで十分です。

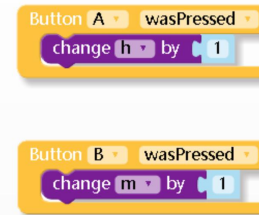
秒の変数“s”が60になったら、秒を0に戻して分を1進めます



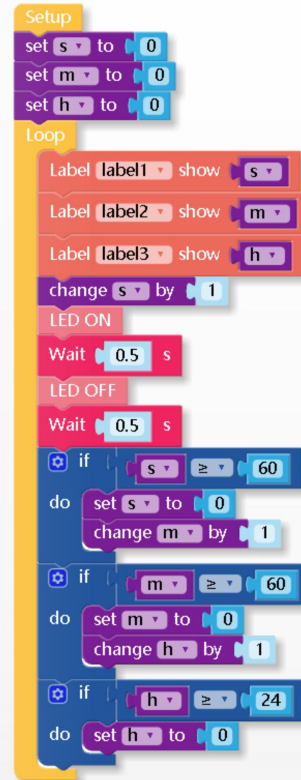
秒の変数“s”が60を超えていないかも、大小比較(不等号)のブロックでチェックしておきます。

3. IFの条件

それぞれの変数を1ずつ増やすごとに、60や24になっていないかをチェックし、超えていたら0に戻し、上の桁を1増やします。これに加えて、時間のセットのためのボタンを追加しておきましょう。



2つの“buttn pressed”(ボタンが押されたら)ブロックを“Event”グループからもってきて、その中に“change”(変える)ブロックを置いて“m”と“h”の値を1ずつ増やすようにしておきます。



3 グラフィック描画

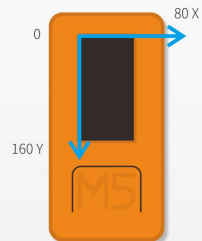
ここでは、液晶ディスプレイにグラフィック模様を描いて、座標について理解します。X/Y座標系に慣れましょう。また Random (乱数) ブロックについて学び、それを使って液晶ディスプレイ上のランダムな場所に模様を描いてみます。

3-1 座標を理解する

座標は、画面上の決まった場所に図形を描くのにとっても便利です。また、画面上の2つの場所の間の距離を求めることもできます。座標の考え方は、私たちの普段の生活でもよく使います。例えば地図では、座標を使って正確な場所を表現することができます。正しくその場所に行くことができます。液晶ディスプレイにも、地図のような座標のための格子があり、また画面自体が、画素と呼ばれる小さな四角が集まってできています。



M5StickCの液晶ディスプレイは、80×160個の画素で成り立っています。X軸(横方向)に80個、Y軸(縦方向)に160個です。画面上に画素の点や図形を描くときには、その場所はX/Y座標で指定します。その座標が示す場所が画面の外にあったら、画面上には表示されません。



3-2 円を描く

ここでは、UIマネージャを使って画面上に円を描いて、その位置を示すX/Y座標を属性(プロパティ)から設定してみます。



01 円(Circle)を画面上にドラッグして置きます。

03 属性(プロパティ)で、図形の名前、座標、半径、色、レイヤを変更できます。

02 図形をクリックすると、属性(プロパティ)が表示されます。ダブルクリックすると図形を複製できます。

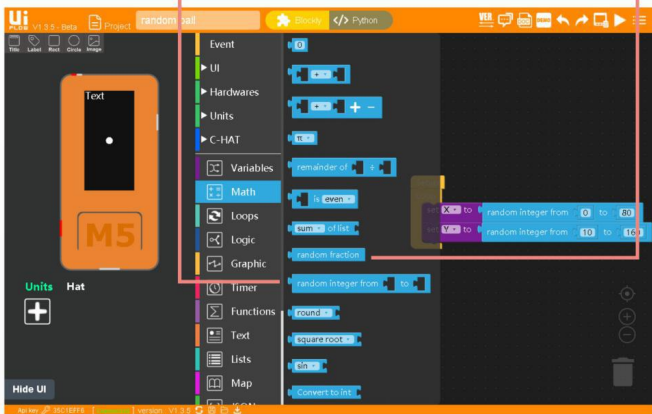
04 Labelを画面上に置きます
このラベルは、座標を理解するために、円の座標を表示するのに使います。

乱数

"Math" (数学) グループの中に、乱数を作るブロックが2種類あります。

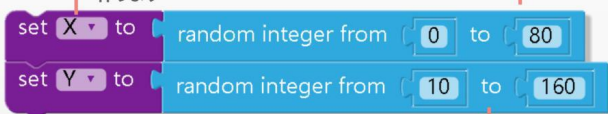
"random draction"は、0~1の範囲の乱数(例えば0.2)を生成します。

"random integer from a to b"は、指定された範囲(a~b)の整数の乱数を生成します。



01 変数を入れておくためのXとYの2つの変数を作ります

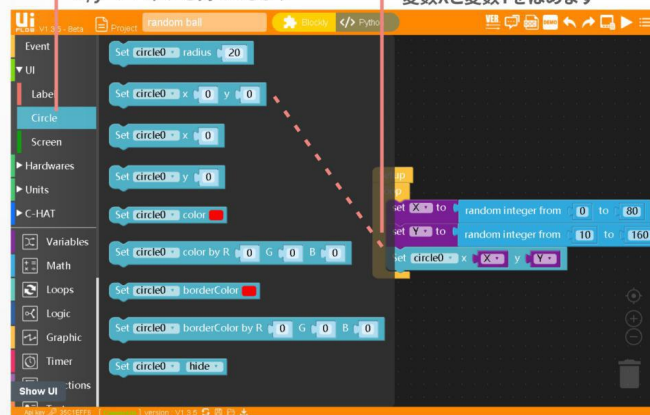
02 変数Xの値を、範囲を0~80に指定した"random integer"で決めます。



03 変数Yの値を、範囲を10~160に指定した"random integer"で決めます。

04 "UI"をクリックし、その中の"circle x/y"ブロックを持ってきます

05 その円のX/Y座標のところに、変数Xと変数Yをはめます



このプログラムを実行させると、画面全体のいろいろな場所に円が現れます。次に、その円のX/Y座標をLabelに表示させてみましょう。1つのLabelに、X座標とY座標の2つの数値を一緒に表示するのは、どうやればいいでしょうか？

ここで、"Text" (テキスト) の中にある新しいブロックを使います。

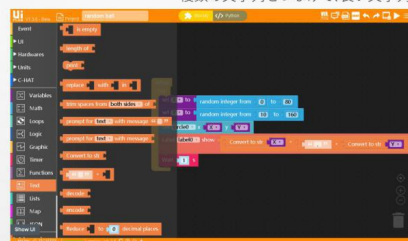
Convert to str

"Cnver to string"ブロック

数値を文字列(string)に変換します。文字列に変換したあとは、例えば1を足すような計算はできません。

"Connect srings"ブロック

複数の文字列をつなげて、長い文字列をつくります。



"connect string"ブロックを、別の"connect string"ブロックの中に置き、"convert to string"をXとYの変数の値の前に置きます。これらを、最初においた"connect string"ブロックの最初と際その場所にはめます。こうして作った長い文字列を、"labe show"ブロックにはめます。

座標の表示がよく見えるように、1秒分の"wait"ブロックを置いておきましょう。

4 加速度センサを使ってみる

ここでは、加速度センサを使って、画面の色を変えてみます。加速度センサの働きと使い方や、論理とプログラムの処理の制御についても学びます。

加速度センサとは？

加速度センサはものが動くときの加速の度合いを計測します。加速度は数ミリ秒、数マイクロ秒の短い時間の間に速さがどれくらい変化するかを、という察です。

例えば、まっすぐな道で、車が1秒間に時速30kmから時速50kmまで加速したとしましょう。このときの加速度は、1秒あたり時速20kmとなります。

ちょっと問題:2台の車がスタートラインに止まっていて、時速60kmまで加速したとします。その速度になるまでに、片方の車は5秒、もう片方は10秒かかったとします。2台の車の加速度は同じでしょうか？

もちろん答えは、違う、です。5秒かかった車のほうが、10秒かかった方よりも、加速が急、つまり加速度が大きいわけです。



加速度は、私たちの身の回り、例えば手をちょっと動かすときでも現れます。動かす向きが変わるのも、加速度が生じているわけです。



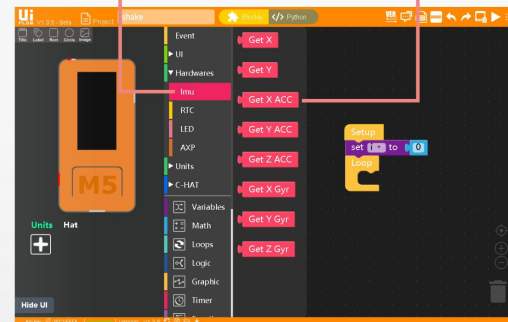
次に、加速度にあわせて画面の色を変えてみましょう。

- (1) 色を表す変数"i"をつくります
- (2) Setupのところで、"i"の値を0にしておきます



01 "Hardware" (ハードウェア) の中にある "IMU" を選びます (IMUとは、慣性計測装置 inertial measurement unitの略です)

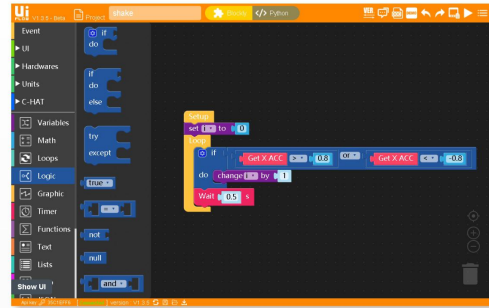
02 X方向の加速度を計測する "Get X ACC" ブロックを使います。



4-2 制御の流れ

加速度センサの値をみれば、M5StickCを振ったことがわかるので、それにあわせて変数の値を変えて、画面の色を変えてみましょう。

振った回数を保存する変数をつくります。最初は値を0にしておいて、降るごとに1ずつ増やすことにします。

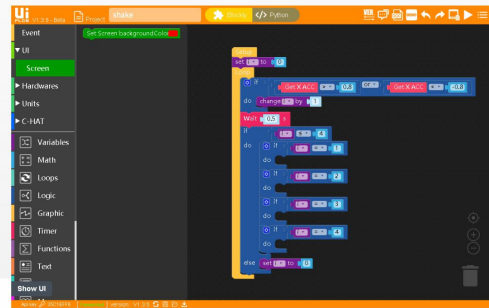


加速度センサのX方向の値のしきい値(これより大きかったら「振った」と判定する境界の値)を0.8としておきます。

論理の比較ブロックには、AND、OR、NOTの3つの設定ができます。

- (1) AND: 左と右の両方の条件が成立するときに「真」となり、動作します。どちらか一方でも成立しなければ「偽」となって実行されません。
- (2) OR: どれか1つでも条件が成立するときに「真」となって動作します。両方とも成立しないときだけ「偽」となって動作しません。
- (3) NOT: 条件が成立しないときに「真」となり、成立するときに「偽」となります。

振った回数にあわせて画面の色を変えます。4回を超えたら0に戻すことにします。



```

Setup
set i to 0
Loop
if Get X ACC > 0.8 or Get X ACC
do change i by 1
Wait 0.5 s
if i ≤ 4
do
if i = 1
do Set Screen backgroundColor red
if i = 2
do Set Screen backgroundColor yellow
if i = 3
do Set Screen backgroundColor cyan
if i = 4
do Set Screen backgroundColor green
else set i to 0
    
```

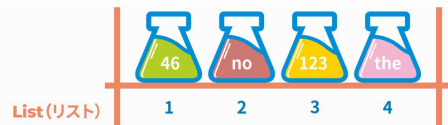
画面の色が、変数*i*の値にあわせたif文の条件が成り立つところの色に変わります。

5 デジタル名刺

ここでは、ボタンを押すとその人の詳しい情報が表示されるデジタル名刺をつくってみたい。
リストの作り方と使い方を学びます。

1、リストとは?

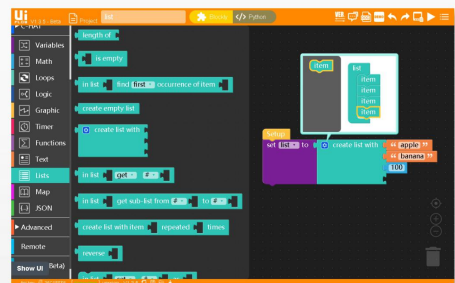
リストとは、いくつかの変数を1つにまとめて扱えるようにしたものです。変数は箱(またはビン)のようなものですが、リストは箱(ビン)を入れておく箱(コンテナ)のようなものです。コンテナの中のビンには番号がついていて、その番号で指定するビン(変数)の中身を読み出したり書き換えたりできます。



リストをつくるには、変数にリストを割り当てて、そのリストの中の番号で、使いたい変数を指定します。

2、リストをつくる

まずリストを入れる変数をつくり、その変数の"set list to"ブロックをSetupにはめます。ブロック一覧の"List"の中の"create list with"ブロックを、先ほどの"set list to"ブロックにはめます。それから、リストの中の変数に、"Math"の中にある値や、"Text"の中にある文字列(text)をはめたり、必要なければなにもはめないままにしておきます。ここにはめた値が、リストの中の変数(番号は1から始まる)に代入されます。最初はリストの中の変数は4個ですが、歯車マークをクリックして、そこでitemをドラッグすれば増やすこともできます。



3、リストの使い方

リストをつくって、中の変数に値を代入したら、それをどう使うのか、次の例でみてみましょう。

01 "Lists" (リスト)のタブをクリックします。

02 "In List Get" (リストの値を得る)を持ってきて、変数につなぎます。

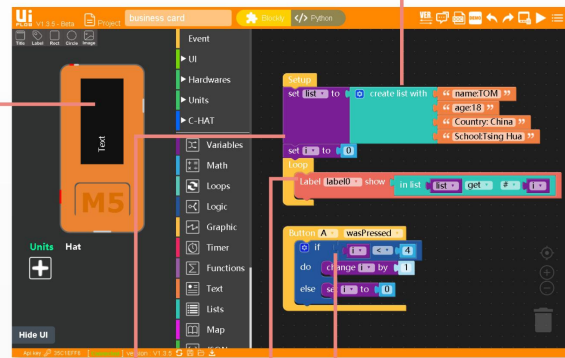
03 ドロップダウンメニューから、リストの中の番号を指定して、変数の値を得たり消したりできます。

UIFlowではリストの変数の番号は1から始まりますが、Pythonでは0から始まります。この例で"M5"という文字を取得するには、番号(index)に1を指定します。

ちょっと考えてみよう; リストの中の番号を巡回して(最後の番号の次は最初に戻る)使うことはできますか? まさにそれがリストが便利な理由の一つです。

5-2 自分のデジタル名刺を作ってみよう

- 01 名刺の内容を表示するLabelを追加します。M5StickCの画面の幅は狭いので、Labelを270度(左向きに90度)回転して、縦長の画面にあわせて表示させることにします。
- 02 名刺に載せる情報を入れておくリストを作ります



- 03 リストの中の変数の場所を示す変数をつくり、最初に値を1にしておきます。
- 04 リストの中身を表示する"Label show"ブロックを置きます。
- 05 "A button pressed"ブロックを置き、その中に変数"i"(リストの中の表示すべき項目の番号)を設定します。この例ではリストの中の項目は4個ですから、ifブロックを使って、4の次は0にできるようにしておきます。



6 玉よけゲーム

画面に四角を表示し、加速度センサで動かします。繰り返しループとその使い方、それをゲームを作るのにどう使いか学びます。

6-1 繰り返し処理とは?

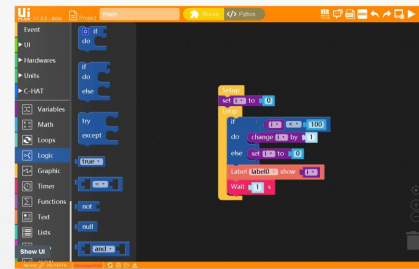
ちょっと考えてみよう: プログラムで繰り返し処理を使うと、どんないいことがあるのでしょうか?



簡単なプログラムで具体的にみてみましょう。

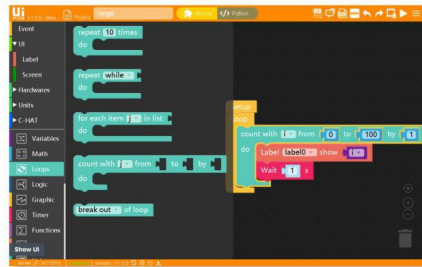
指定した回数を繰り返す処理を、2つの方法でプログラムを描いてみて、どちらがよいか、比べてみましょう。カウンタの値の変数を0から100まで1ずつ増やしていく場合を考えます。

Lets have a comparison of two ways to iterate numbers and see which one is more efficient. The goal is to count up from 0 - 100 in increments of 1.



方法1

変数"i"が100より小さい間、"i"を1ずつ増やしていき、"i"が100になったら0に戻します。数が増えていく様子を確認できるように、0.1~1秒のwaitを入れておきます。これらを、1つのLoopブロックの中に入れます。



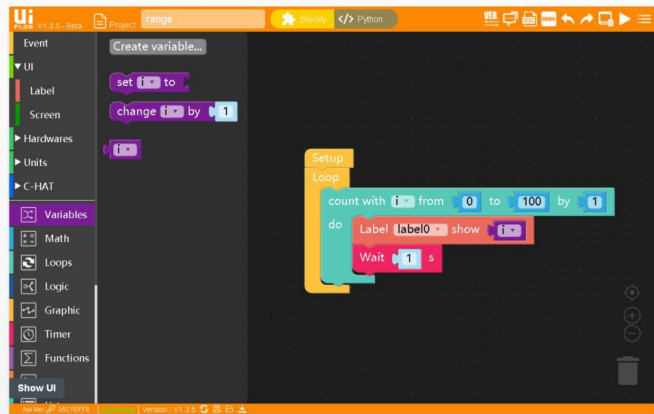
方法2

"count with i from"ブロックを使って、最初の値を0,max(最大値)を100、1回あたりの増えず数を1にしています。確認のためにループの中に1秒の待ち時間を入れておきます。これで繰り返し処理が終わるまで100秒かかることとなります。"i"の値をLabelで表示しておけば、0から1ずつ増えていって100まで行ったら0に戻るのが確認できるはずです。

方法2のほうが簡単ですね。このようにcountブロックを使ったほうが、繰り返し処理のプログラムは簡単になることが多いです。

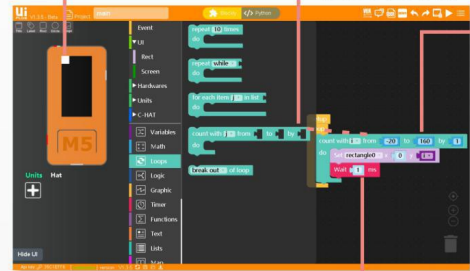
補足

UIFlowでは、countブロックを置くと、自動的に変数"i"が作られますが、別の変数を作成して、それをドロップダウンリストから選ぶこともできます。



6-2 countブロックで座標を設定

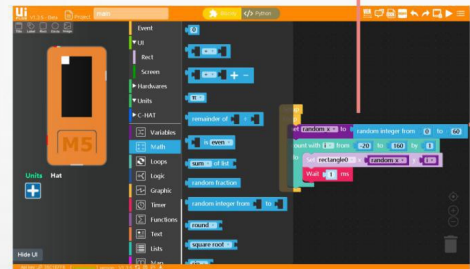
- 01 rect (長方形) を画面におきます。
- 02 countブロックを使って、rect (長方形)の座標を少しずつ変えるようにします。



- 03 長方形が移動する速度を決めるために、waitブロックを置いて待ち時間を入れます。

- 04 rect (長方形)の場所は、左上の点の座標で指定します。ここでは、長方形が画面の上から現れて、下まで行ったら元の位置に戻すようにしてみましょう。そのためにはcountブロックの値の範囲は20~160となります。

- 05 繰り返しの最初のところで、乱数をつくるRandomブロックで、長方形が落下し始める前にX軸(横)方向の位置を決めます。



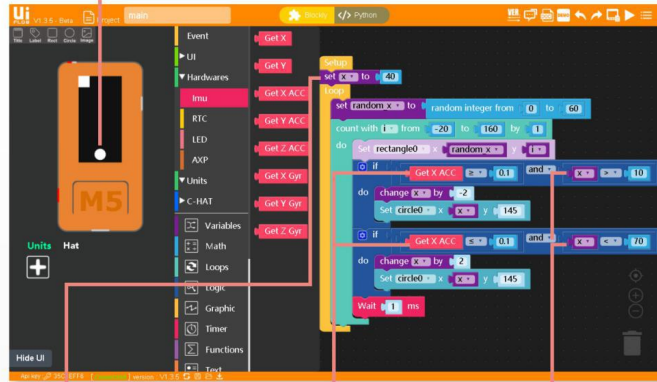
- 06 長方形が画面内に収まるように、乱数の範囲は0~60にしておきます。(長方形のサイズは20×20画素で、画面の横方向は80画素)

- 07 これで、countループの最初で、長方形が現れる場所が決まります。

6-3 ボールの制御

次に、プレーヤーの役割のボールをつくら、加速度センサの値に応じて左右に動かしてみます。

circle (円) を、画面の下の端に置きます(これがプレーヤーのボール)



ボールのX座標を入れておく変数"X"をつくり、最初の値を中央の40にしておきます

加速度センサの値を読み、それに応じてボールを左右に動かすためにX座標を増減させます

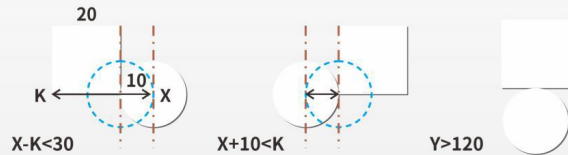
"less" (〜より小さい) 比較ブロックを使って、ボールが画面からはみ出さないようにします

6-4 当たり判定

ゲームの設計上、得点のための当たり判定をつくっておきます。

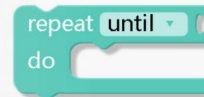
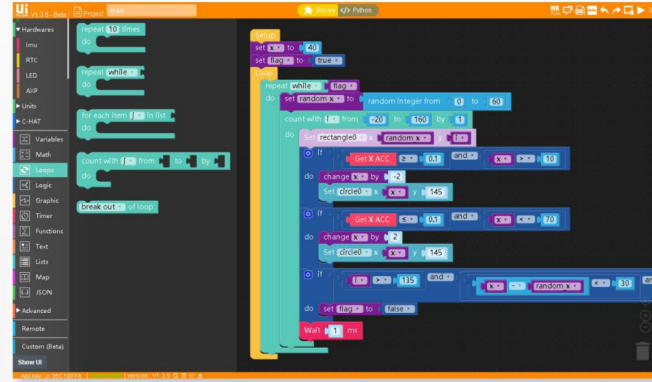
当たり判定は、次の3通りが考えられます。

- (1) 四角の右側が円の左側に接触する、
- (2) 四角の左側が円の右側に接する、
- (3) 四角の下側が円の上面に接する、の3通りです。このどれかが成り立ったら、ゲームオーバーとしましょう。

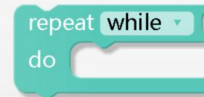


ゲームの実行

スタートボタンを押したらゲームが始まるようにプログラムを描いていないので、プログラムの実行が始まると、すぐにゲームが始まってしまいます。またゲーム中、四角があたってゲームオーバーになったら、いったんボールを止めて、再スタートできるようにする必要があります。そこでゲーム中かゲーム終了かの状態を表す変数をつくりましょう。次のプログラムをみてください。



"repeat until"ブロックは、指定した条件が成り立つまで、中に置いたプログラムを実行します。条件が成り立ったらループは終わりです。



"repeat while"ブロックは、指定した条件が成り立っている間は、中に置いたプログラムを実行します。その条件が成り立たなくなったら、ループは終わりです。

6-7 プログラムの最適化

これでゲームのプログラムの基本骨格はできました。次に、点数とゲームのリセットについてのプログラムを追加しましょう。

The code block shows a game loop. It starts with 'set score to 0' and 'set flag to true'. A 'do' loop contains 'set score to 0', 'set label to hide', 'set rectangle to show', and 'set circle to show'. A 'repeat while' block with 'flag' contains a 'do' loop with 'set random-x to random integer from 0 to 60', 'count with from 20 to 160 by 1', 'do' loop with 'set rectangle to random-x * 10', 'if' block with 'get x acc < 0.01 and < 0.01' containing 'change x by -2' and 'set circle to x * 145', and another 'if' block with 'get x acc < 0.01 and < 0.01' containing 'change x by 2' and 'set circle to x * 145'. A final 'if' block with 'x < 100 and < 135' contains 'set flag to false', 'wait 1 ms', 'change score by 1', 'set rectangle to hide', 'set circle to hide', and 'repeat until A pressed' containing 'set label to show' and 'set score to score + 1'. The code ends with 'set flag to true'.

Annotations:

- 点数を入れておく変数"score" (スコア、点数)です
- ループの始まりの段階では、Labelは消しておいて、四角と円を表示させておきます。
- プレイヤーが四角をよけられたら、点数を加算します
- ゲームオーバーになったら、四角と円を消します

Aボタンを押すまでは、前のゲームの点数が表示されています。ゲームオーバーになったときに、変数"x"が160未満なら、ループは抜けません。ループ内のプログラムは、ループを抜ける前に1回実行されます。つまり点数は正確ではないので1減らして調整しなければなりません。Aボタンを押したら、"flag"を真(true)にします。

7 デジタル音楽

スピーカーのHAT (拡張ユニット)を使って、M5StickCで音を鳴らして、簡単な音楽を演奏してみましょう。音の原理について学び、"repeat"ブロックの使い方を学びます

HATを追加する

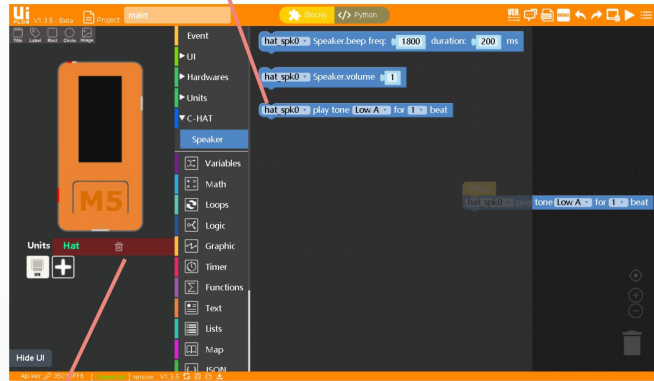
The screenshot shows the AT menu with various HAT options. Callouts 01 through 04 point to specific elements:

- 01: HATのプラスマークのボタンを押して、ATメニューを表示させます
- 02: HATメニューから、いろいろなHATやセンサーを使えるように追加できます
- 03: "SPK" (スピーカ) のHATを選びます
- 04: Okを押して保存します



HATの使い方

新しくHATを追加すると、ブロック一覧に「C-HAT」というグループが現れます。HATにはそれぞれの機能にあわせたブロックがあり、HATからセンサなどの情報を読み取ったり、何かを出力したりできます。プログラムを実行する前に、使うHATを、M5StickCの頭の部分の拡張コネクタにしっかりと接続します。



HATと取り除く

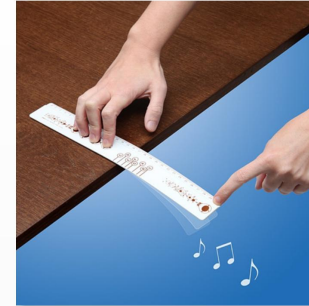
HATを間違えて追加してしまったり、もう使わなくなったら、そのHATを、UIマネージャの上の方に現れるゴミ箱にドラッグすれば、取り除くことができます。

7-1 音楽をつくる

ここでは、M5StickCにつないだSPK HATのスピーカから、簡単な音をならせてみましょう。ところで音って、なんでしょう？

音は、実は振動です。定規を机の端に置いて片方を固定し、もう片方を手で押し込んで離してみましょう。どんな音が聞こえて、持っている手にどんな振動を感じますか？

定規の場所を変えたり、押さえる場所を変えてみましょう。いろんな音が出るはずですよ。



この実験から、速く振動するほど音が高く、遅いほど低くなるのがわかります。振動の速さは「周波数」と呼ばれて、「Hz(ヘルツ)」という単位で表します。人間の耳に聞こえるのは、およそ20Hz~20kHz(20,000Hz)の音です。それより低かったり高い音は聞こえません。

音程と周波数

もう少し進んで、音から音楽へいってみましょう。「Hardware」の中の「Speaker」ブロックを持ってきて、Setupにつなぎます。このブロックの1つ目は周波数(freq)で、2つ目はその音になる時間(ms=ミリ秒、1000分の1秒)です。これらの数値を変えてどんな音が聞こえるか試してみましょう。

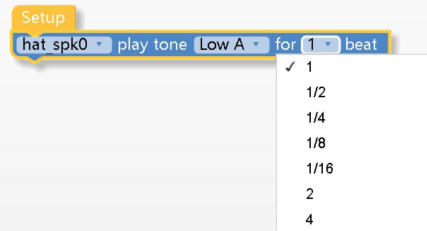
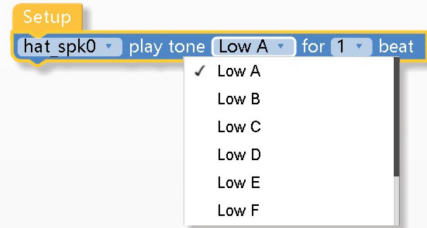
Setup

hat_spk0 | Speaker.beep freq: 1800 duration: 200 ms

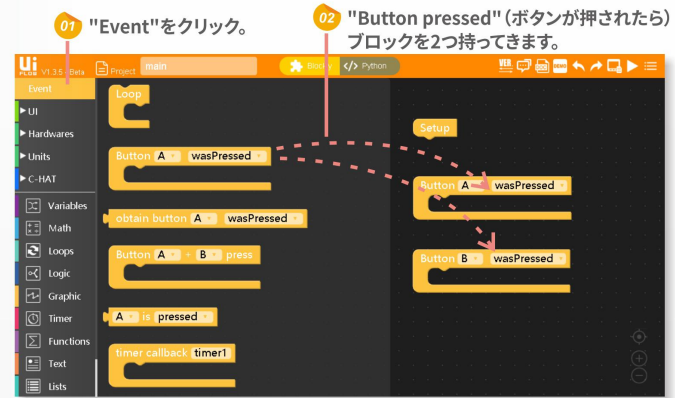
音楽をつくるのに、いちいち周波数を入力するのは面倒です。そこで使われるのが「音程」です。C, D, E, F, G, A, Bの記号が、順にド、レ、ミ、ファ、ソ、ラ、シを表し、それぞれの周波数が決まっています。ピアノなどの楽器では、さらに高い音や低い音も、このC~Bを繰り返します（オクターブ）。それぞれの音程と周波数の関係は次の表のとおりです。

	C	D	E	F	G	A	B
Low	131	147	165	175	196	220	247
Middle	262	294	330	349	392	440	494
High	523	587	659	698	784	880	988

スピーカのブロックには、音程を直接指定できるブロックもあります。1つ目が音程、2つ目が拍、つまり音の長さです。



下の画面のように周波数と音の関係を確認するプログラムを書いてみましょう。



05 プログラムを実行し、AボタンとBボタンを順に押してみよう。同じ音が鳴るのがわかります。

簡単な楽譜の音楽をつくってみましょう。

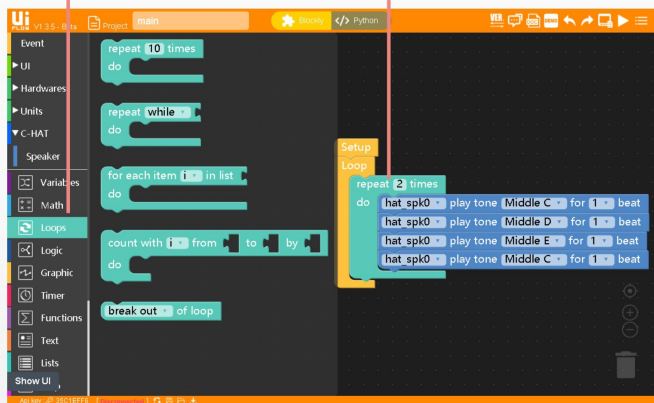


これは簡単な音楽の楽譜です。よく見ると、同じフレーズが繰り返されているところがあります。プログラミングでは、このような繰り返しは簡単に書くことができます。"Repeat" (繰り返し) ブロックは、Loopブロックに似ていますが、指定した回数だけ繰り返すものです。



01 "Loop"をクリック

02 "Repeat"の回数を2に設定して、その中に、楽譜の最初の小説を書いておきます。



"Repeat"を使わないと、左のように、ずいぶん長いプログラムになります。動作の結果は同じですが、Repeatブロックを使うと、ずいぶん短くて、すっきりしたプログラムになりますね。



プログラミングでは、このように「楽をする」ことは、けっこう大切です。

7-2 M5オーケストラ

みんなで合奏(合唱)してみましょう。それぞれが自分のパートを、ボタンを押して演奏し、誰か一人が指揮者になって、ボタンを押すタイミングを指示しましょう。

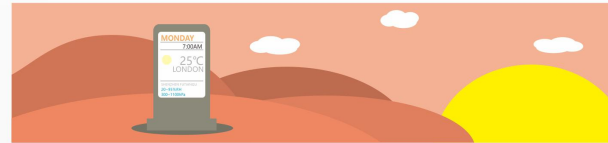


8 デジタル温度計

ここでは、ENV(環境センサ) HATを使って、デジタル温度計をつくってみましょう。ENV HATの使い方と、それでセンサ値を読み取る方法について学びます。

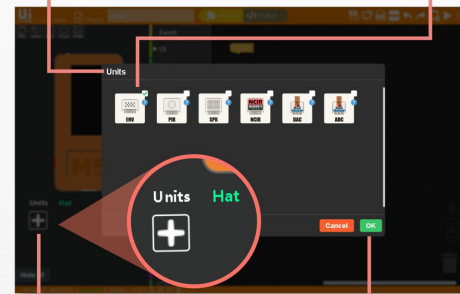
8-1 ENVとは?

ENV (Environment:環境) HATとは、温度、湿度、気圧、磁気方位を計測できるセンサです。植物栽培などのために現在の温度を調べるときなどに便利です。ENV HATを使えば、天気予報もできそうです。温度は-20~60°C、湿度は20~95%RH(相対湿度)、気圧は300~1100Pa(パスカル)の範囲で計測できます。



02 HATメニューの中に、使えるいろいろなセンサが並んでいます

03 ENV HATを選択します



01 "HAT"を選び、+ボタンを押します

04 Okボタンを押して追加します

8-2 センサの値を読み取る

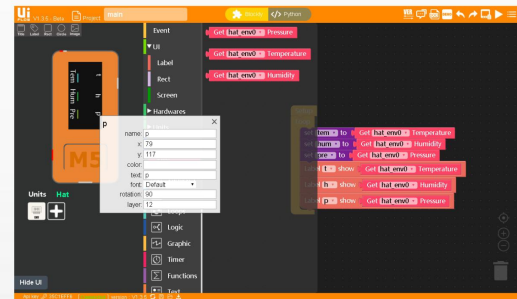
Labelを使って、センサの値を画面に表示してみましょう。



次から、センサの値を表示する画面をつくっていきます。

8-3 センサの値の表示

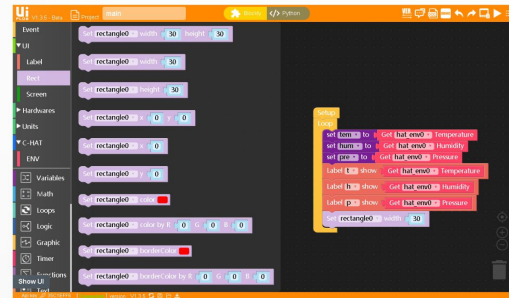
UIマネージャで、Labelを6個置きます。そのうち3個はセンサ値の名前を、残り3つはセンサ値を表示させます。その他に見栄えをよくなるためにいくつか四角を置いておきましょう。



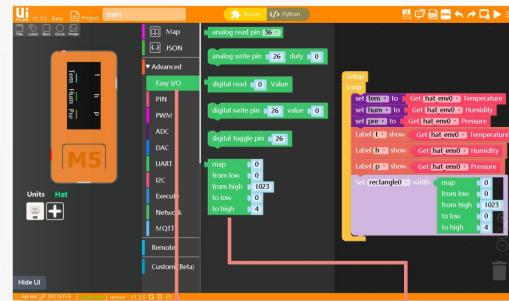
四角を3個置きます。枠の太さは1にして、色は好きな色にしてください。

センサ値の名前を表示するLabelを3個置きます。

センサ値を表示するためのLabelを3個置きます。



四角は温度・湿度・気圧の値の大きさを大きさとして表示するのに使います。しかし画面の解像度は80×160画素しかないので、センサの値をそのまま四角の大きさにすると画面をはみ出してしまいます。そこでMapブロックを使って、センサ値から画面サイズにおさまる四角のサイズに変換して調整します。



- 01 "Advanced"の中の"EasyIO"を選択します
- 02 "Map"ブロックをLoopの中に置きます

Mapブロックを使って、温度センサの値の範囲である-20~60を、0~30の範囲に変換します。これで温度の値に応じて四角の大きさが0~30で変化して、温度を確認しやすくなります。

```

map
  Get hat_env0 Temperature
  from low -20
  from high 60
  to low 0
  to high 30
  
```

Map (割り当て) とは?

Mapブロックは、ある数値を、別の数値に変換する働きがあります。例えばドルから円へ両替するとき、お金の価値は変わりませんが、為替レートの比率で金額を変換します。



温度と同じように、湿度と気圧も、次のようにMapブロックで値を変換しておきます。ENV HATをつないでプログラムを実行すれば、センサの値が表示されるはずです。

```

Setup
Loop
  set tem to Get hat_env0 Temperature
  set hum to Get hat_env0 Humidity
  set pre to Get hat_env0 Pressure
  Label t show Get hat_env0 Temperature
  Label h show Get hat_env0 Humidity
  Label p show Get hat_env0 Pressure

  Set rectangle0 width map
    Get hat_env0 Temperature
    from low -20
    from high 60
    to low 0
    to high 30

  Set rectangle1 width map
    Get hat_env0 Humidity
    from low 20
    from high 95
    to low 0
    to high 30

  Set rectangle2 width map
    Get hat_env0 Pressure
    from low 300
    from high 1100
    to low 0
    to high 30
  
```

APPENDIX:

FIRMWARE UPGRADE

APPENDIX

「付録」

アイデアを素早く形に

WIFI CONFIGURATION

LEARN MORE:

FIRMWARE UPGRADE

M5StickC



ファームウェアの更新



Wi-Fiの設定



付録



より深く学ぶ

M5StickC

M5StickC

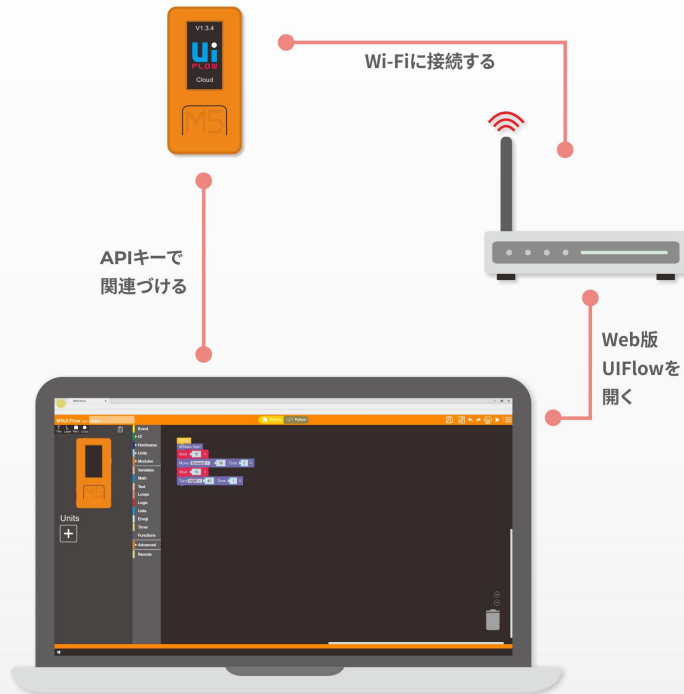
M5StickC

M5StickC

Wi-Fiの設定

M5StickCでプログラムを実行させるには、USB経由以外に、Wi-Fi経由でもできます。Web版のUIFlowでは、無線通信やIoT関連の機能が使えます。

Web版とIDE版のUIFlowの機能はほとんど同じですが、Web版ではM5StickCをWi-Fiに接続してから使う必要があります。そのためにはM5StickCの「APIキー」で、プログラムを実行させるM5StickCを関連づけます。



Wi-Fiの設定

01 本体左側の電源ボタンを2秒間長押しして電源をオンにします。画面にロゴが表示されたら、すぐに本体正面のM5ボタンを押します。



02 本体右側のボタンを押してメニューの項目を選択します

03 M5ボタンを押してメニューの選択を確定します

04 Cloud (クラウド) モードを選択します



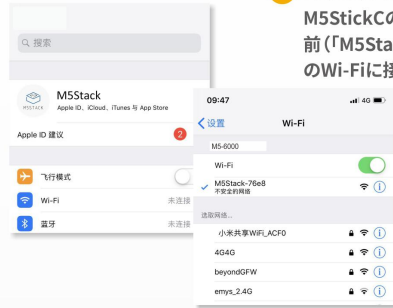
"Wi-Fi Setting" (Wi-Fi設定) から、接続するWi-Fiを選択したり、接続したことがあるWi-Fiの設定を削除できます。

注意：
使う場面にあわせて、Wi-FiモードとUSBモードを選択します

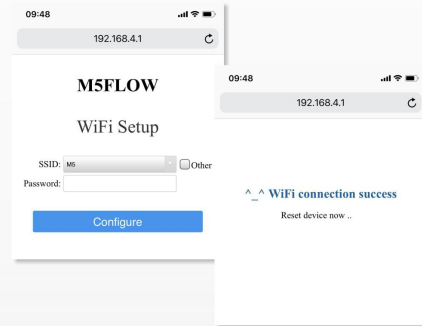
05 "change Wi-Fi connect" (Wi-Fi接続を変更する) を選ぶと、M5StickCのWi-Fi名が表示されます。



05 スマホやパソコンのWi-Fi設定から、M5StickCの画面に表示されている名前(「M5Stack-????」という名前のはず)のWi-Fiに接続します。



06 接続できたら、Webブラウザを開き、アドレス(URL)に"192.168.4.1"を入力するとWi-Fi設定画面が表示されます。



07 M5StickCが接続したいWi-Fiを選び、パスワードを入力してかConfigureボタンを押します。そのWi-Fiに接続できたら、M5StickCは自動的にリセットがかかります。

APIキーの入力

01 M5StickCが起動すると、Cloudモードの画面が表示されます。地球マークが赤から緑になったら準備OKです。



02 パソコンのWebブラウザでflow.m5stack.com を開き、UIFlowを開きます。APIキーの入力画面が表示されるはずですが。(表示されなかったら、右上の歯車マークをクリックすれば表示されます)

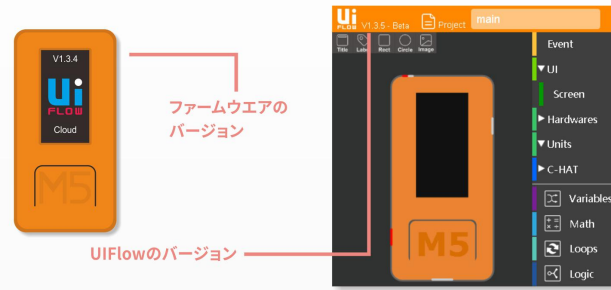


03 "Device"のときは、"Stick-C"を選んでおきます

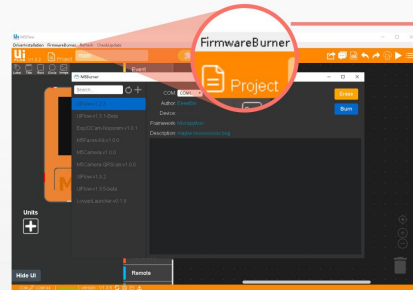
04 M5StickCの画面に表示されているAPIキー(8桁の英数字、この例では"16573A08")を入力してOKを押して、接続されたら、プログラムの実行ができます。

📱 ファームウェアの更新

UIFlowの最新の機能を使うためには、定期的にM5StickCのファームウェア(内部のプログラム)を更新しましょう。M5StickCのファームウェアは、UIFlowと同じバージョンのものを使います。M5StickC起動時に画面の右上にバージョン番号が表示されています。UIFlowのバージョンの方がM5StickCよりも新しいならば、UIFlowのバージョンが更新されているということなので、M5StickCのファームウェアも更新すべきということになります。UIFlow(オフライン・IDE版)にはファームウェア書き込みツールもついています。



注意:最近のパソコンでは、M5StickCを使うのに特別なドライバーは必要ありませんが、やや古いパソコンでは、ftdichip.comからVCP (virtual com port)ドライバーをインストールする必要があります。



"Firmware Burner"をクリックして、ファームウェア書き込みツールを開きます。

1.COMポートの番号

M5StickCをパソコンにつなぐと、COMポートの番号が割り当てられます。ファームウェアの書き込みの時には、M5StickCに割り当てられたCOMポート番号をドロップダウンリストから選びます。Windows/パソコンでは、COMポート番号は、COMのあとに数字がついた名前 (COM32など) です。Macでは"/dev/tty.SLAB_USBtoUART"のような名前になります。

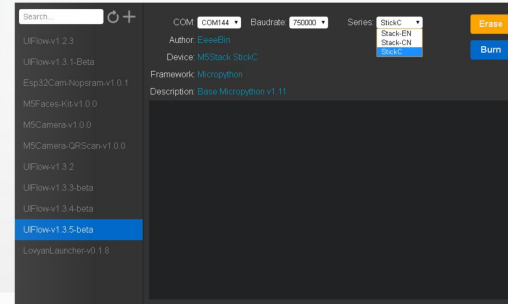


2. Baud (通信速度)

Baud rate (通信速度) は、ファームウェアを書き込むときのM5StickCとパソコンの通信速度です。通常は自動的に115200か921600になります。

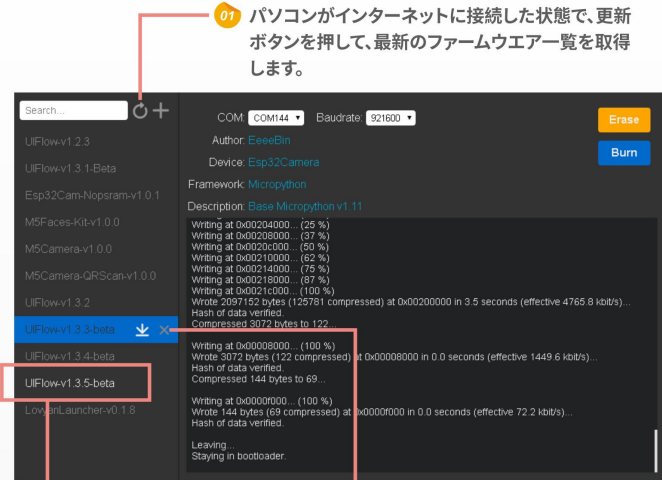
3. ファームウェアのシリーズ

左のリストから書き込みたいバージョンのファームウェアを選びます。"Series"はM5StickCを選び、"Burn"ボタンで書き込みます。



4. ファームウェアのバージョン

左のリストから、ダウンロードしたいバージョンのファームウェアを選びます。古いバージョンと最新のバージョンのものがあり、用途にあわせて選びますが、通常は、UIFlowと同じバージョンのファームウェアをエラに併。



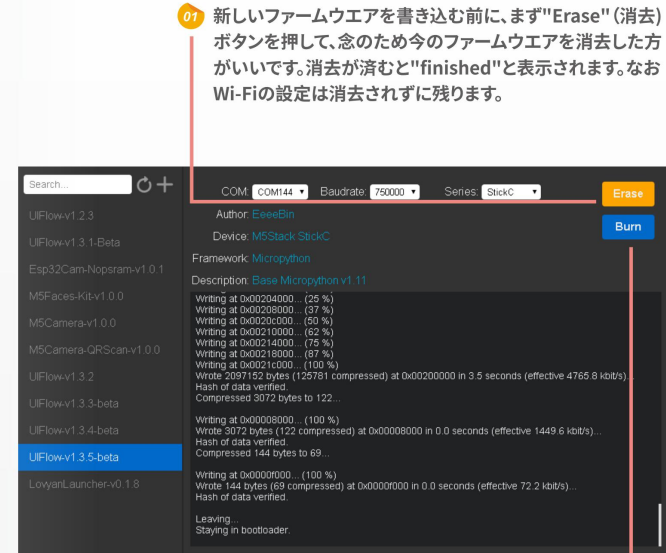
01 パソコンがインターネットに接続した状態で、更新ボタンを押して、最新のファームウェア一覧を取得します。

02 色が白っぽいものはダウンロード済みで、書き込みができます。

03 グレーのものはまだダウンロードされていないので、右側のダウンロードのアイコンをクリックしてダウンロードします。

5. ファームウェアの書き込み

書き込みをしたいバージョンのファームウェアを選んで書き込みをします。



01 新しいファームウェアを書き込む前に、まず"Eraser" (消去) ボタンを押して、念のため今のファームウェアを消去した方がいいです。消去が済むと"finished"と表示されます。なおWi-Fiの設定は消去されずに残ります。

02 "Burn" (書き込み) ボタンを押して書き込みを開始します。"Finished"が表示されたら、書き込み完了です。

M5StickC



より詳しい情報やM5Stackについては、<https://m5stack.com> を参照してください。

M5StickC

この本が、UIFlowでM5StickCのプログラミングをする助けになることを祈っています。そしてプログラミングの楽しさを知って、自分のプログラムを書ききっかけになればと思います。プログラミングは始まったばかりです、もっといろいろ勉強して、いろいろなものを作ってみましょう。

M5StickCの機能を拡張するパーツがいろいろあるおんで、ぜひ探してみてください。
日本ではスイッチサイエンス www.switch-science.com から購入できます。



facebook



twitter



shopify

M5StickC

