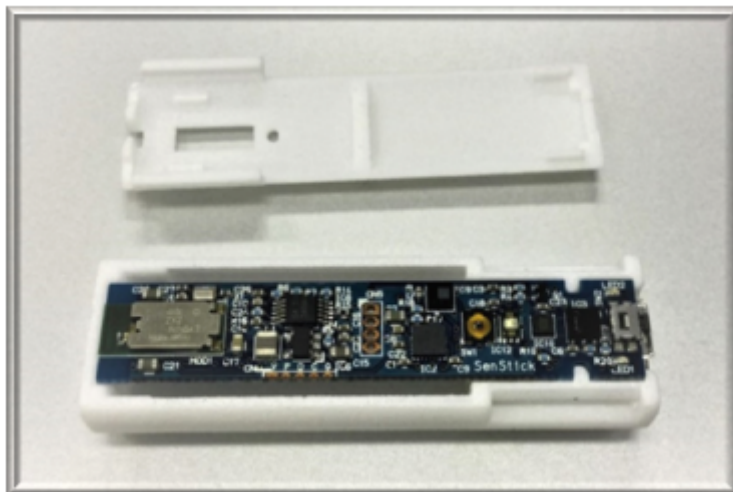


# SenStick4

## 取扱説明書



作成日 : 2022年1月14日(第2版)    **Revision 1.1**



一般社団法人

**Rubyビジネス推進協議会**

# 目 次

<b>1. はじめに</b>	<b>2</b>
1.1. SenStick4について	2
1.2. 初期動作確認	2
1.3. サンプル	3
<b>2. 機能概要</b>	<b>4</b>
2.1. SenStick4の部品構成と各部名称	4
2.2. 加速度及びジャイロセンサと磁気センサの座標系	6
(1) 加速度およびジャイロセンサの座標系	6
(2) 磁気センサの座標系	6
2.3. 動作モードと操作スイッチ	7
2.4. 動作時の消費電流	7
2.5. バッテリーについて	8
(1) 添付のバッテリーについて	8
(2) バッテリーを購入する場合の目安	8
(3) バッテリー充電方法	8
<b>3. SenStick4の仕様</b>	<b>9</b>
3.1. SenStick4搭載のBLE GATTサービスの仕様	9
(1) mruby/cサービス	9
3.2. mrubyセンサクラス リファレンス	11
3.3. サンプルコード	12
<b>4. SenStick4ケースの作り方について</b>	<b>13</b>
4.1. ケース設計データの入手	13
4.2. ケースの発注の方法 (DMM.make)	14
<b>5. その他</b>	<b>15</b>
5.1. 問合せについて	15
5.2. 取扱安全上のご注意	15
<b>6. 改訂履歴</b>	<b>16</b>

## 1. はじめに

### 1.1. SenStick4について

◇ SenStick4は、次の7種類のセンシングデバイスを搭載しています。

- ① 加速度センサ
- ② 角速度センサ
- ③ 磁界(磁気)センサ
- ④ 照度センサ
- ⑤ 紫外線(UV)センサ
- ⑥ 湿度・温度センサ
- ⑦ 気圧センサ

◇ Bluetooth® Low Energy(以下BLEと表記)でPCやスマートフォンとの接続が可能です。

接続にはBLE4.2以上が搭載されている必要があります。接続してSenStick4のデータを活用できます。

◇ 各センシングデバイスは、mruby/cを使いプログラミング制御が可能です。

作成したプログラムは、BLEでSenStick4へ転送・書込を行い実行が可能です。

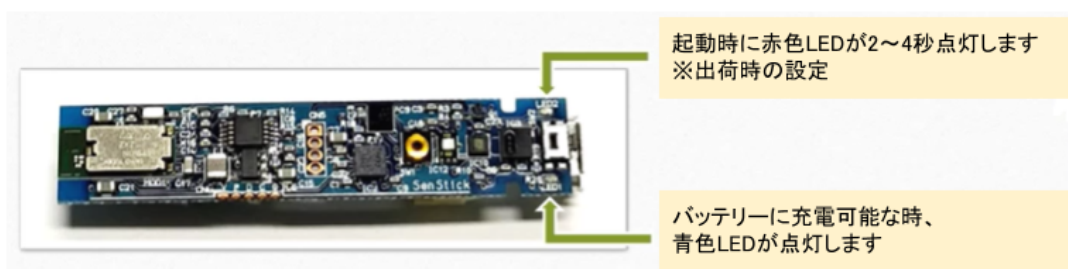
### 1.2. 初期動作確認

SenStick4を入手後すぐに、次の初期動作を確認ください。

なお、接続に必要なマイクロUSBケーブルは、各自でご用意下さい。

①USBから電源を供給して青色LEDが点灯、赤色LEDが2～4秒点灯することを確認

②USBを外してバッテリーを接続し、赤色LEDのみが2～4秒点灯することを確認



### ③チュートリアルを実施してソフト稼働を確認

[https://ruby-b-senstick.github.io/senstick\\_check/](https://ruby-b-senstick.github.io/senstick_check/)

チュートリアルの「1.SenStick4の接続と動作確認」を実行してください。



※動作不良の場合は、購入元にご確認ください。

ただし、ご購入から7日を超えると対応が出来ない場合があります。ご注意下さい。

### 1.3.サンプル

稼働確認後は、チュートリアルを用意していますのでご活用下さい。

2. SenStick4のセンサデータ取得
3. mruby/cのプログラム実行
4. mruby/cプログラムの作成

## 2. 機能概要

### 2.1. SenStick4の部品構成と各部名称

図1) SenStick4の部品構成

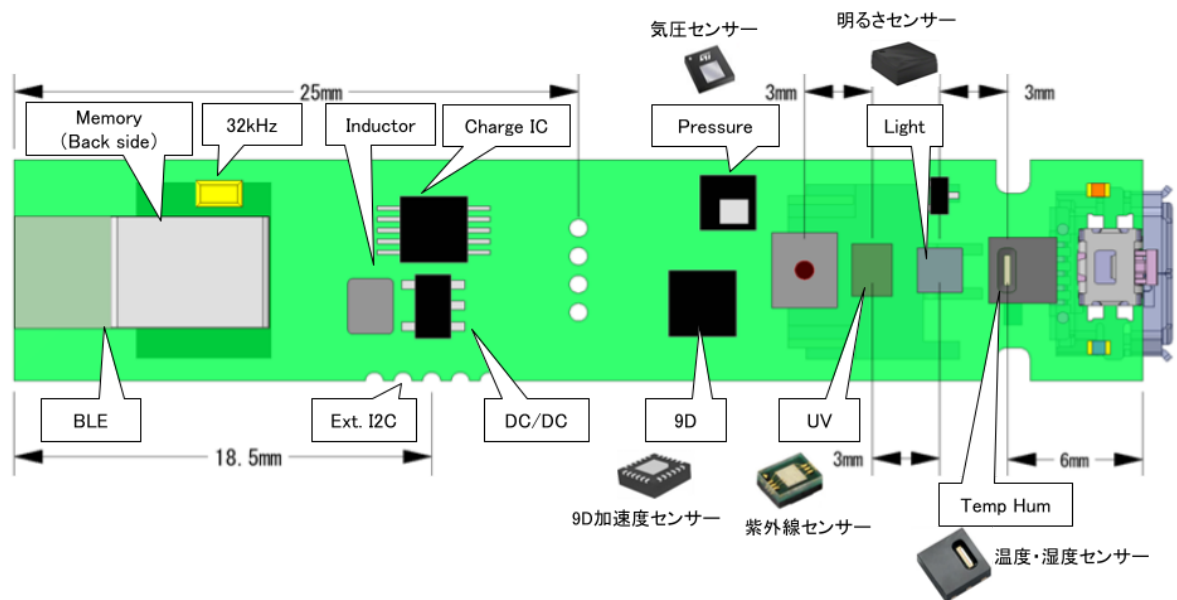


図2) SenStick4の各部名称

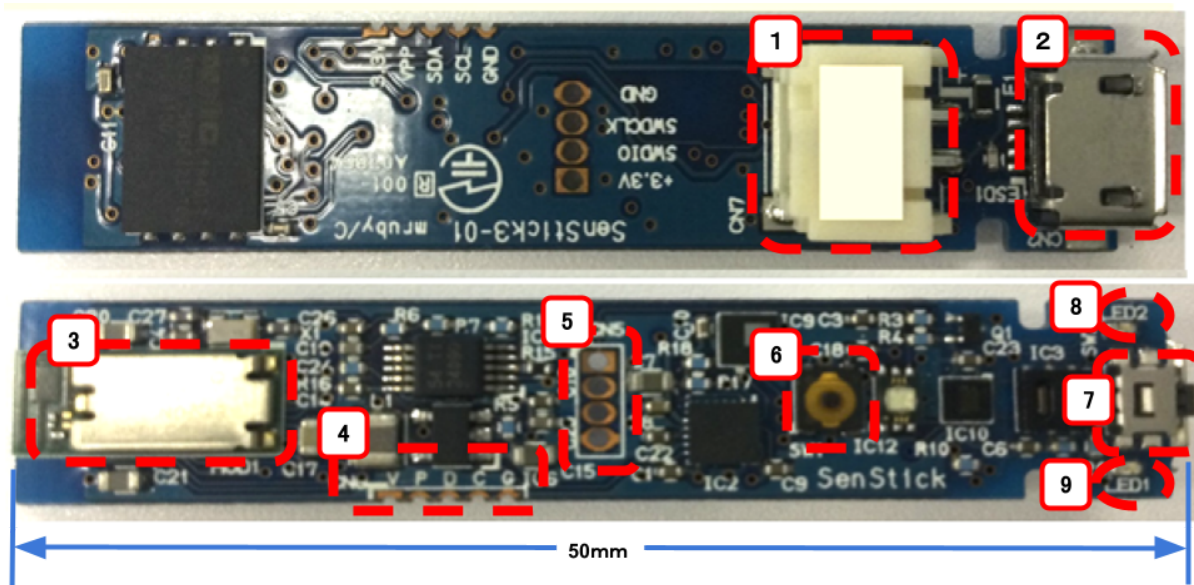


表1) SenStick4各部の機能説明

番号	名 称	概 要
1	バッテリーコネクタ	リチウムイオンポリマーバッテリーの接続コネクタです。 バッテリーの推奨スペックは「2.5.バッテリーについて」に記載しています。
2	マイクロUSBコネクタ	本体電源とバッテリー充電の電力供給にのみ使われます。 USB端子に接続してもUSB機器としては認識されません。
3	BLEモジュール	Nordic Semiconductor社のnRF52832を採用した Bluetooth Low Energy(BLE)モジュールです。 太陽誘電 EYSHJNZXZです。
4	I2C拡張端子	外部回路を接続するためのI2Cと電源端子です。 この端子を利用するにはファームウェアの拡張が必要です。
5	プログラム端子	BLEモジュールのデバッグとファームウェア書き込みのための 端子です。
6	リセットスイッチ	電源を落とす強制リセットスイッチです。
7	操作スイッチ	本体操作のためのスイッチです。
8	青色LED	バッテリー充電時に点灯します。充電が完了すれば消灯します。
9	赤色LED	SenStick4起動時に2～4秒程度点灯します。その後はSenStick4内の mruby/cプログラムで制御されます(mruby/cプログラムの内容によって 点灯・消灯が変わります)。

## 2.2.加速度及びジャイロセンサと磁気センサの座標系

加速度及びジャイロセンサと磁気センサの座標系はいずれも右手系で、XY座標が基板と同じ面内にあり、Z軸が基板を貫く方向にあります。

※加速度／ジャイロセンサと磁気センサでは、XY軸の取り方が異なるため注意が必要です。

### (1)加速度およびジャイロセンサの座標系

加速度の座標系は、SenStick4本体のバッテリーコネクタ面を手前に向けてマイクロUSBコネクタを右側にして本体を置いた時に、基板の上方向がX座標のプラス方向、右方向がY座標のプラス方向、基板の手前から奥に向かう側がZ座標のプラス方向になります。

ジャイロの座標系は、加速度と同じです。右手の方向がプラスになります。

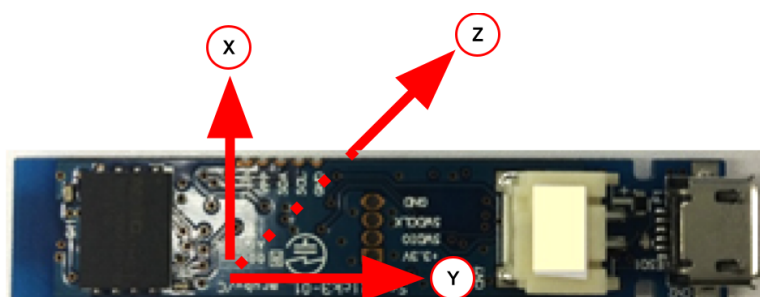


図3)ジャイロセンサの座標

### (2)磁気センサの座標系

磁気の座標系は、基板の上方向がY座標のプラス方向、右方向がX座標のプラス方向、奥から手前に向かう側がZ座標のプラス方向になります。

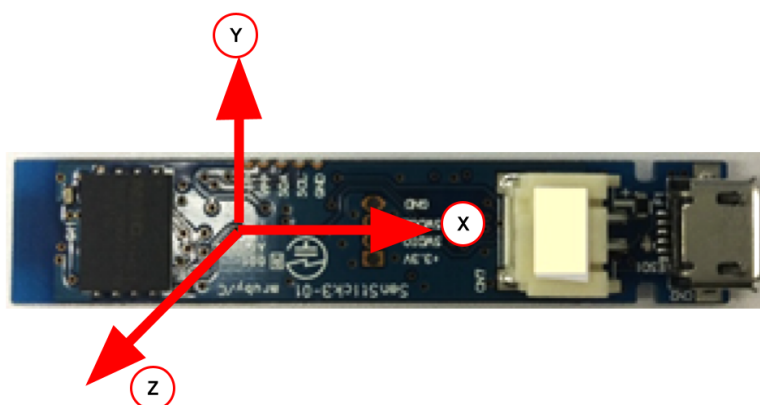


図4)磁気センサの座標

## 2.3. 動作モードと操作スイッチ

SenStick4に電源が供給され起動すると、内蔵のmruby/cプログラムの実行を開始します。  
SenStick4の電源OFFやmruby/cプログラムの初期化を行う場合に動作モードを切り替えます。

SenStick4の操作スイッチを押すと赤色LEDが点滅します。  
操作スイッチを押したまま(長押し)にすると、赤色LEDが【遅い点滅】→【中程度の点滅】→  
【速い点滅】→【遅い点滅】→・・・のように、約3秒ごとに変化します。  
各点滅の段階で操作スイッチから手を離すと以下の処理が行われます。

- 遅い点滅 : 電源をOFFにする
- 中程度の点滅 : 何も動作をせずに、元の状態に戻る
- 速い点滅 : SenStick4を工場出荷時の設定に初期化する

## 2.4. 動作時の消費電流

動作	消費電流
スリープ時(電源off時)	17 $\mu$ A
動作時 9軸 10ミリ秒 その他 200ミリ秒	9.1mA
動作時 9軸 30ミリ秒 その他 200ミリ秒	8.0mA
動作時 9軸 オフ その他 200ミリ秒	7.4mA
動作時 9軸 200ミリ秒 その他 200ミリ秒	7.5mA

※あくまで参考値で、実際には環境により変動があります



## 2.5. バッテリーについて

SenStick4の充電性能は**100mA/h**(1時間で、**100mA**の充電が可能)です。

### (1)添付のバッテリーについて

(仕様)

- ・出力電圧 : 3.7V
- ・容量 : 110mA
- ・キャパシティ : 0.5C・・・※1時間で50mAの充電が可能です。  
充電時間は2時間程度必要です。

※注意)付属のバッテリーは初期動作確認用です。

お早めに目的に合う最適なバッテリーをご購入下さい。

### (2)バッテリーを購入する場合の目安

#### ①バッテリー容量(キャパシティは、**1.0C**を推奨)

充電時間

容量	(1.0C)	／	(0.5C)	C:キャパシティ
・100mA・・・	1時間	／	2時間	
・200mA・・・	2時間	／	4時間	
・300mA・・・	3時間	／	6時間	

#### ②バッテリーのサイズ

・本説明書でご紹介しているケースに収納する場合は、次のサイズ内のバッテリーを選択して下さい。

(横)30mm × (縦)12mm × (高さ)6mm 以内

・バッテリーの電源ケーブルの長さは、ケースに収まるよう出来るだけ短いものを推奨します。



### (3)バッテリー充電方法

バッテリーの充電にはUSBケーブル(マイクロB)が必要です。各自でご用意下さい。

#### ※注意事項

##### ①充電する際は電源をOFFの状態にすることが望ましい

※電源ONの状態でも充電はできますが、充電完了までの時間が長くなります。

##### ②電源OFFの方法は、「2.3.動作モードと操作スイッチ」を参照

### 3. SenStick4の仕様

#### 3.1. SenStick4搭載のBLE GATTサービスの仕様

SenStick4 の Base UUID は F000XXXX-0451-4000-B000-000000000000 です。  
XXXXの部分には、各サービス・キャラクターリスティクスのAlias値が入ります。

表2) サービス一覧

サービス名称	Alias値
mrubycサービス	0x1523
mrubycキャラクターリスティクス	0x1525

##### (1)mruby/cサービス

mrubycキャラクターリスティクス(0x1525)は、バイト列をRWできます。  
書き込みバイト列の先頭の1バイトで各機能を識別します。

表3) mruby/cサービス一覧

1バイト目の値	動 作 説 明	
0x00	赤色LEDを消灯する コマンド投入可能な状態を示す	
0x01	赤色LEDを点灯する	
0x02	バージョン情報を返す 書き込み後に、バイト列を読み出す	
0x04	SenStick4を再起動する	
0x05	SenStick4のフラッシュメモリを消去する	
0x06	mruby/c バイトコード転送 コマンドに続けて以下18バイトのバイト列を書き込むこと	
	バイト位置	内 容
	0	必ず0x06(コマンド)
	1～2	シーケンス番号 16バイトのバイトコード番号で0から開始して、 1ずつ増加させる
	3～18	分割された16バイトのバイトコード

0x07	Flashを消去、コマンド0x06で転送されたバイトコードをFlashに書き込み、mrubycプログラムを実行する												
0x10	<p>SenStick4内のmrubycプログラムの変数値を読み出す(get)  2バイト目以降に変数名(終端文字は'\0')を指定する  (変数名の最大長は8バイト)  書き込みを行うと、以下の形式で変数が結果が得られます</p> <ul style="list-style-type: none"> <li>1バイト目が 0x00 の場合 変数に取得できない値が格納されていたため、読み出せなかった</li> <li>1バイト目が 0x01 の場合 2バイト目～5バイト目に整数値が格納されている</li> <li>1バイト目が 0x02 の場合 2バイト目～9バイト目に浮動小数点値が格納されている</li> </ul>												
0x11	<p>SenStick4内のmrubycプログラムの変数に値を書き込む(set)  2バイト目以降に変数名(終端文字は'\0')と値を指定する  (変数名の最大長は8バイト)  指定できる値は、整数値または浮動小数点値で、変数名の終端文字 '\0' に続けて、以下のように指定する</p> <ul style="list-style-type: none"> <li>整数値を書き込む場合 0x01 に続けて 4バイトの整数値を指定する</li> <li>浮動小数点値を書き込む場合 0x02 に続けて 8バイトの浮動小数点値を指定する</li> <li>0x01 または 0x02 以外を指定した場合、変数に nil が書き込まれる</li> </ul>												
0x20	<p>SenStick4のセンサデータ(RAWデータ・前半)を読み出す  11バイトが以下のセンサデータとなる</p> <table border="1"> <thead> <tr> <th>データ</th><th>センサ値の計算方法</th></tr> </thead> <tbody> <tr> <td>d[0] d[1]</td><td>湿度(%)  <math>= 125.0 * (d[0] + d[1] * 256.0) / 65536.0 - 6.0</math></td></tr> <tr> <td>d[2] d[3]</td><td>温度(°C)  <math>= 175.72 * (d[2] + d[3] * 256.0) / 65536.0 - 46.85</math></td></tr> <tr> <td>d[4] d[5]</td><td>明るさ(lx) = <math>d[4] + d[5] * 256.0</math></td></tr> <tr> <td>d[6] d[7]</td><td>UV強度(mW/cm<sup>2</sup>)  <math>= d[6] + d[7] * 256.0</math></td></tr> <tr> <td>d[8] d[9] d[10]</td><td>気圧(hPa)  <math>= (d[8] + (d[9] + d[10] * 256.0) * 256.0) / 4096.0</math></td></tr> </tbody> </table>	データ	センサ値の計算方法	d[0] d[1]	湿度(%) $= 125.0 * (d[0] + d[1] * 256.0) / 65536.0 - 6.0$	d[2] d[3]	温度(°C) $= 175.72 * (d[2] + d[3] * 256.0) / 65536.0 - 46.85$	d[4] d[5]	明るさ(lx) = $d[4] + d[5] * 256.0$	d[6] d[7]	UV強度(mW/cm <sup>2</sup> ) $= d[6] + d[7] * 256.0$	d[8] d[9] d[10]	気圧(hPa) $= (d[8] + (d[9] + d[10] * 256.0) * 256.0) / 4096.0$
データ	センサ値の計算方法												
d[0] d[1]	湿度(%) $= 125.0 * (d[0] + d[1] * 256.0) / 65536.0 - 6.0$												
d[2] d[3]	温度(°C) $= 175.72 * (d[2] + d[3] * 256.0) / 65536.0 - 46.85$												
d[4] d[5]	明るさ(lx) = $d[4] + d[5] * 256.0$												
d[6] d[7]	UV強度(mW/cm <sup>2</sup> ) $= d[6] + d[7] * 256.0$												
d[8] d[9] d[10]	気圧(hPa) $= (d[8] + (d[9] + d[10] * 256.0) * 256.0) / 4096.0$												
0x21	<p>SenStick4のセンサデータ(RAWデータ・後半)を読み出す  18バイトが以下のセンサデータとなる</p> <table border="1"> <thead> <tr> <th>データ</th><th>内 容</th></tr> </thead> <tbody> <tr> <td>x: d[0] d[1] y: d[2] d[3] z: d[4] d[5]</td><td>地磁気センサの値(x,y,z) (μT)  <math>x = (d[1] + d[0] * 256) * 4912.0 / 32768.0</math>  y,zについても同様の計算で求めます</td></tr> <tr> <td>x: d[6] d[7]</td><td>加速度(重力加速度)の値(x,y,z) (g)</td></tr> </tbody> </table>	データ	内 容	x: d[0] d[1] y: d[2] d[3] z: d[4] d[5]	地磁気センサの値(x,y,z) (μT) $x = (d[1] + d[0] * 256) * 4912.0 / 32768.0$ y,zについても同様の計算で求めます	x: d[6] d[7]	加速度(重力加速度)の値(x,y,z) (g)						
データ	内 容												
x: d[0] d[1] y: d[2] d[3] z: d[4] d[5]	地磁気センサの値(x,y,z) (μT) $x = (d[1] + d[0] * 256) * 4912.0 / 32768.0$ y,zについても同様の計算で求めます												
x: d[6] d[7]	加速度(重力加速度)の値(x,y,z) (g)												

	y: d[8] d[9] z: d[10] d[11]	$x = (d[7] + d[6] * 256) * 4912.0 / 32768.0$ y,zについても同様の計算で求めます
	x: d[12] d[13] y: d[14] d[15] z: d[16] d[17]	角加速度(回転)の値(x,y,z)(度/sec) $x = (d[13] + d[12] * 256) * 4912.0 / 32768.0$ y,zについても同様の計算で求めます

### 3.2. mrubyセンサクラス リファレンス

SenStick4のセンサは、SenStickIFクラスでアクセスします。

表4) SenStickIFクラス一覧

メソッド名	説 明	
new	センサからデータを取得するためのインスタンスを生成します。使用するセンサをパラメータで指定します。温・湿度センサ、9軸IMUセンサについては、パラメータのいずれを指定しても構わない(重複して指定しても構わない)	
	パラメータの値	意 味
	:temperature :humidity	温湿度センサ
	:brightness	照度センサ
	:uv	UVセンサ
	:airpressure	気圧センサ
	:acc :gyro :magnetic	9軸IMUセンサ
get	センサから値を取得します。	
	センサの種類	説 明
	:temperature	摂氏温度を取得する。FLOAT型で単位は℃。
	:humidity	湿度を取得する。FLOAT型で単位は %。
	:brightness	照度を取得する。INT型で単位は lx。
	:uv	UV強度を取得する。INT型で単位は mW/cm2。
	:airpressure	気圧を取得する。FLOAT型で単位は hPa。
	:acc	加速度を取得する。FLOAT型で単位は g。
	:gyro	角加速度を取得する。FLOAT型で単位は 度/s。
	:magnetic	地磁気を取得する。FLOAT型で単位は μT。

### 3.3. サンプルコード

#### センサクラスの使用例

◇ 温度センサを使い、温度が30度を超えるとLEDを点滅させるプログラム例。

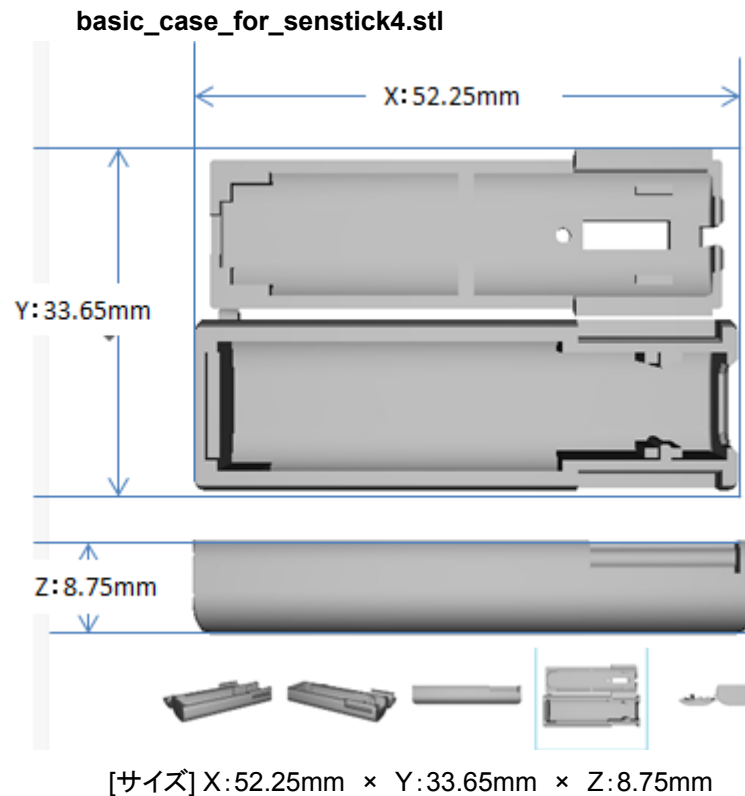
表5) サンプル

```
temp = SenStickIF.new(:temperature)
while ture
  t = temp.get(:temperature)
  if t>=30 then
    led 1
  else
    led 0
  end
  sleep 0.5
end
```

## 4. SenStick4ケースの作り方について

### 4.1. ケース設計データの入手

図5) 基本ケース



○入手方法：

Rubyビジネス推進協議会の“SenStick”関連ページより、ケースの3Dデータを入手できます。  
3Dプリンターがあれば、このデータを使ってケースを作成できます。

ファイル名：basic\_case\_for\_senstick4.stl

ダウンロード先：[https://ruby-b-senstick.github.io/senstick\\_check/](https://ruby-b-senstick.github.io/senstick_check/)



## 4.2. ケースの発注の方法(DMM.make)

○3Dプリンターが無くても、“DMM.make”サイトを利用してケースを作ることができます。

DMM.makeサイトURL : <https://make.dmm.com/>



図6) DMM.make



サイトに接続して **3Dモデルをアップロード** し、  
“basic\_case\_for\_senstick4.stl” データをアップロードします。  
あとは、サイトの手順に従い進めます。



## 5. その他

### 5.1. 問合せについて

問合せなどは、Rubyビジネス推進協議会HP「SenStick4+mruby/c教育キット」サイトの

お申込み・お問い合わせ いたします。

<http://senstick.ruby-b.com/>



### 5.2. 取扱安全上のご注意

1. バッテリーは本体基板に接続した状態で使用してください。バッテリーを外した状態では動作しません。
2. バッテリーが発熱・膨張などの状態になった場合は、バッテリーの劣化や不具合の可能性があります。直ちにバッテリーを本体基板から外し、使用を中止してください。
3. さらに、バッテリーを交換しても同じ状況が続く場合は、本体基板の故障の可能性があります。直ちに使用を中止して、販売元にご相談下さい。
4. 防水機能は搭載していません。本体基板に水滴などがつかないようにご注意ください。
5. ゴミやホコリが付くと故障の原因になります。ケースを別途入手いただき、ケースに入れてのご使用をお勧めします。
6. シンナー・ベンジンなどの有機溶剤で清掃しないでください。
7. 高温での使用はご注意下さい。
8. 無線機器の使用が禁止された区域では使用しないでください。
9. 各センサの出力値は、精度が求められる計測装置としての使用は出来ません。あくまでも参考値としてお取り扱いください。
10. 長期間ご使用しない場合はバッテリーを外して保管して下さい。

## 6. 改訂履歴

2021年11月1日            初版Rev1.0リリース

2022年1月14日           Rev1.1リリース

ファームウェアバージョン「20211222-1」

- ・SenStickIFクラス 地磁気取得用の定数変更
- ・SenStickIFクラス 加速度・角速度・地磁気取得型追記
- ・角速度取得値の単位追記
- ・mruby/cサービス センサ値取得バイト数の記載変更
- ・mruby/cサービス 変数値取得情報の詳細記載