

CHAT拡張 仕様書

micro:bitを用いてメッセージのやり取り、及び特定のIDに
メッセージを送るなど双方向的なコンテンツの授業の実現
のための拡張

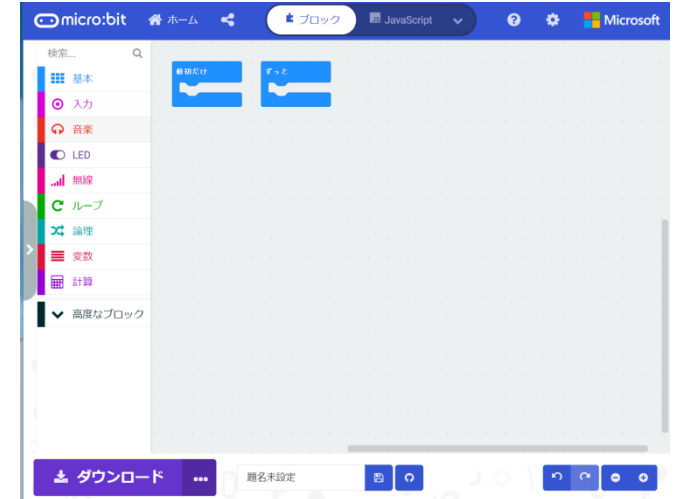
使用方法

Google等の検索エンジンを用いて
「makecode」を検索→micro:bitをクリック

高度なブロック→拡張機能→

「<https://github.com/sakai-code/CHAT>」

と打ち込む



もしくは「https://github.com/sakai-code/micro-bit_extention_package_by-sakai-code」からzipファイルをDL&解凍し中にある(microbit-ip_network.hex)ファイルを、高度なブロック→拡張機能→「ファイルを読み込む」から導入してください。

すると



このようなブロックが出現します。

別途必要(推奨)な物

- Phコネクタを用いた乾電池で動作をさせると持ち運びしやすくなり便利です。
- Micro:bitの標準搭載のLEDを用いて文字列を表示すると動作が遅い&他の動作を受け付けないという癖があるため、SSD1306というi2cのOLEDを用いた表示を推奨しています。その他16x2の液晶もありますがmicro:bitの動作との相性が悪いため非推奨としています。

- 使用しているSSD1306用の拡張:

URL「<https://github.com/sakai-code/oled-ssd1306>」

同じように高度なブロック→拡張機能のURL欄に貼り付けて使用します。

詳しい使い方は(仮)より

もしくは:URL

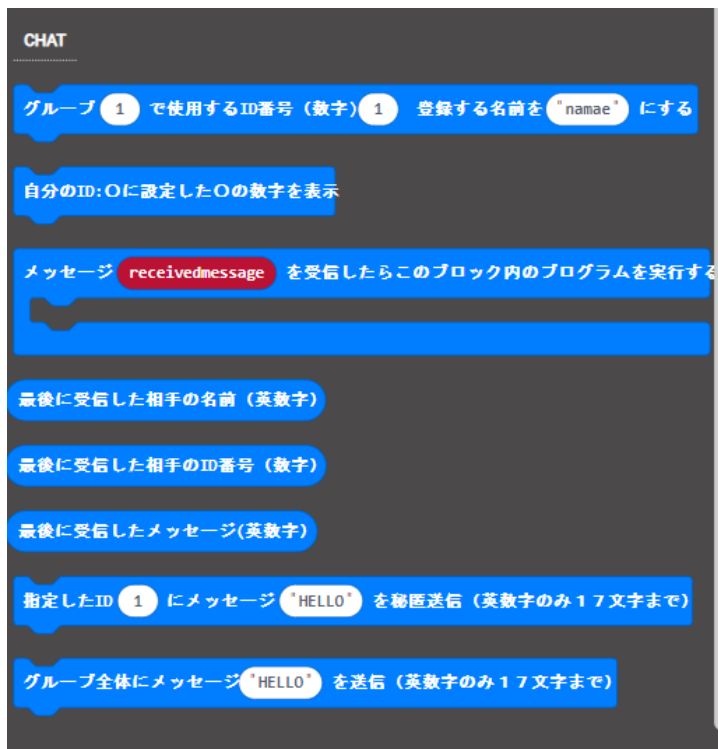


https://github.com/sakai-code/micro-bit_extention_package_by-sakai-code

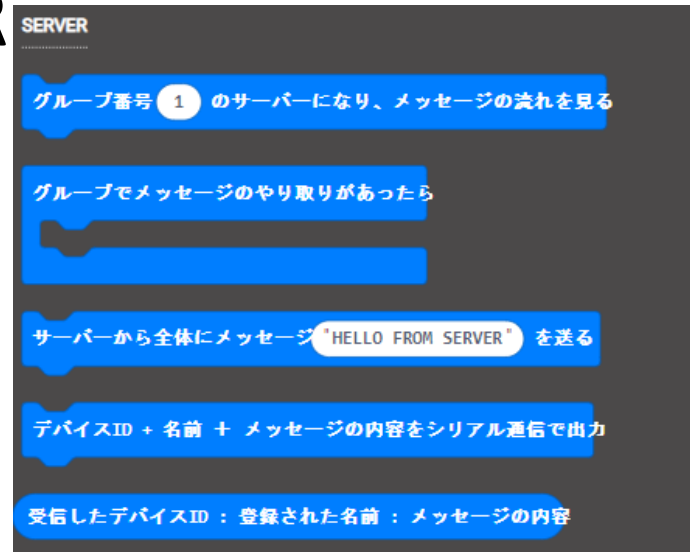
内のmicrobit-OLED-ssd1306.hex を↓ Code→Download ZIP よりダウンロード&解凍し
高度なブロック→拡張機能→ファイルを読み込むより使用してください。

基本仕様

• CHAT



▪ SERVER



この二つの機能に分かれており、CHATが主に使用する機能で、SERVERが先生やグループ内1名がグループ内のメッセージのやり取りを見る際に使用する機能です。

CHAT機能解説(1)

グループ 1 で使用するID番号(数字) 1 登録する名前を "namae" にする

- グループ名番号と登録するID番号(数字) 登録する名前(例: tarou)を設定します。

自分のID: Oに設定したOの数字を表示

- 登録した自分のID番号をmicro:bit上のLEDに表示する

メッセージ message を受信したらこのブロック内のプログラムを実行する

- グループ内でメッセージのやり取りがあった時にこのブロック内のプログラム内を実行します。messageは受信したメッセージの内容(英数字)です

最後に受信した相手の名前(英数字)

- 一番最後に受信した相手の名前(英数字)です。

CHAT機能解説(2)

最後に受信した相手のID番号(数字)

- 最後に受信したID番号(数字)です

最後に受信したメッセージ(英数字)

- 最後に受信したメッセージです

指定したID **1** にメッセージ **"HELLO"** を秘匿送信 (英数字のみ17文字まで)

- 指定した**ID宛**にメッセージを送信します。このブロックで送信したメッセージはサーバー、他の人には見えません。なお、このブロックで送信したメッセージは相手
ブロックのmessage内からのみ取得できます。

グループ全体にメッセージ **"HELLO"** を送信 (英数字のみ17文字まで)

同じグループ内にメッセージを送信します。英数字17文字までが送信できます。なお、このブロックで送信したメッセージはグループ内の
ブロックから取得できます。

メッセージ **message** を受信したらこのブロック内のプログラムを実行する

SERVER機能解説(1)

グループ番号 **1** のサーバーになり、メッセージの流れを見る

- 最初だけブロック内にいれると、入力したグループ番号内でサーバーとなりメッセージの流れをモニタリングすることができます。

グループでメッセージのやり取りがあったら

- グループ内で情報(文字列)のやり取りがあったらこのブロック内のプログラムが実行されます。

サーバーから全体にメッセージ "HELLO FROM SERVER" を送る

- グループ内にサーバーからメッセージ(17文字までの英数字)を送ります

デバイスID + 名前 + メッセージの内容をシリアル通信で出力

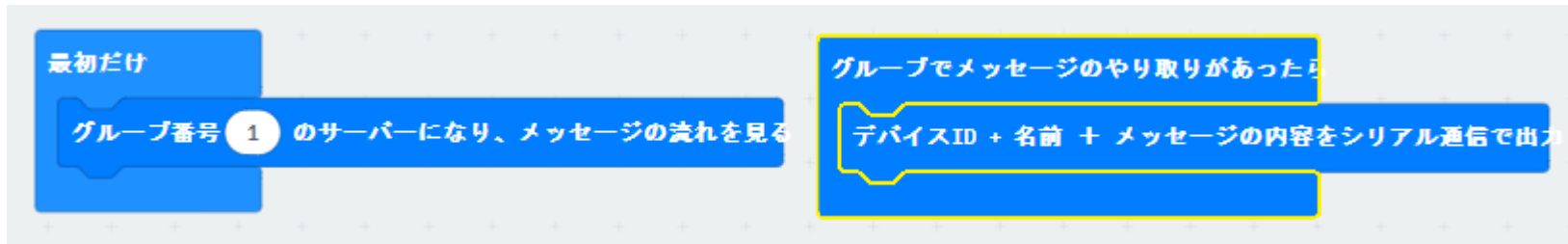
- やりとりの内容をシリアル通信出力で表示します。シリアル通信の使い方は次のページで解説しています。

受信したデバイスID : 登録された名前 : メッセージの内容

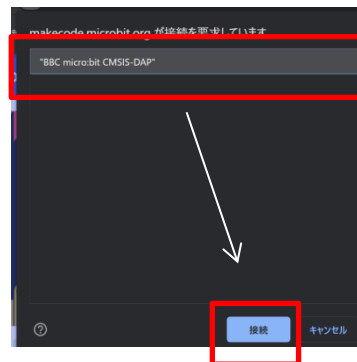
- 受信したID : 受信した名前 : メッセージの文字列を出力します。

SERVER機能解説(2)

- SERVER機能内のメッセージのやりとりをシリアルモニタでモニタリングする方法を解説します。ブロックをこのように配置します。



- 例: グループ1のサーバーとなりメッセージの流れを見ている
- micro:bitをUSBでPCに接続→
- ダウンロードの右の[⋮]ボタンをクリックしデバイスを接続するをクリック→

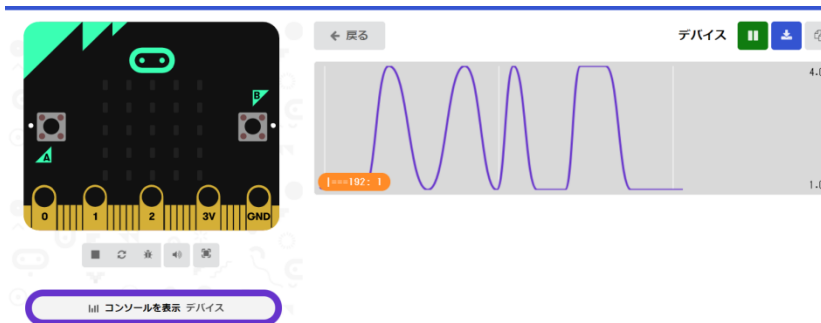


SEVER機能解説(3)

- シリアル通信のモニタリング機能の解説の続き

- メッセージのやり取りがグループ内であると→
が表示される。クリックすると下の画面が表示されます。

||| コンソールを表示 デバイス



RECEIVED!

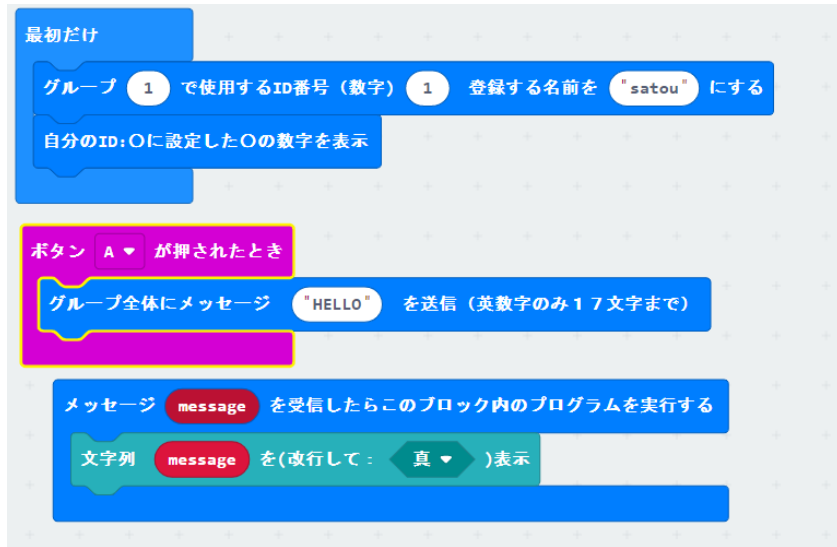
```
|...[DEVICE_ID]...|...[NAME]...|.....[MESSAGE].....|  
|===DEVICEID: 1===|===satou===|===HELLO===|  
|=====|=====|=====|
```

送信したID | 送信した名前 | メッセージ

上記の形式となっています。

```
#####100.0.0.0#####100.0.0.0#####RECEIVED!#####  
|=====|=====|=====|  
RECEIVED!  
|...[FROM]...|...[TO]...|.....[MESSAGE].....|  
|===192.168.0.2===|===192.168.0.1===|===REQUESTDATA:1===|  
|=====|=====|=====|  
RECEIVED!  
|...[FROM]...|...[TO]...|.....[MESSAGE].....|  
|===192.168.0.2===|===192.168.0.1===|===REQUESTDATA:1===|  
|=====|=====|=====|
```

サンプルプログラム 1



- グループ1となりIDを1、名前を“satou”にする
- Aボタンが押されたらグループ全体に“Hello”を送信
- メッセージを受信したらOLED(SSD1306)にmessageを表示

サンプルプログラム2

The diagram shows a sequence of Scratch-like blocks for a program. It starts with a 'When green flag clicked' block (yellow), followed by a 'Say Hello! for 2 secs' block (blue). Then, a 'When button A is clicked' block (purple) triggers a 'Send message to all in group 1' block (blue). This is followed by a 'When I receive a message' block (blue), which triggers a 'Say Hello! for 2 secs' block (blue). Finally, a 'When button B is clicked' block (purple) triggers a 'Send secret message to ID 1' block (blue).

最初だけ

グループ 1 で使用するID番号 (数字) 2 登録する名前を "yamada" にする

自分のID: 0 に設定した0の数字を表示

ボタン A が押されたとき

グループ全体にメッセージ "HELLO" を送信 (英数字のみ17文字まで)

メッセージ message を受信したらこのブロック内のプログラムを実行する

文字列 最後に受信した相手の名前 (英数字) を (改行して: 偽) 表示

文字列 ":" を (改行して: 真) 表示

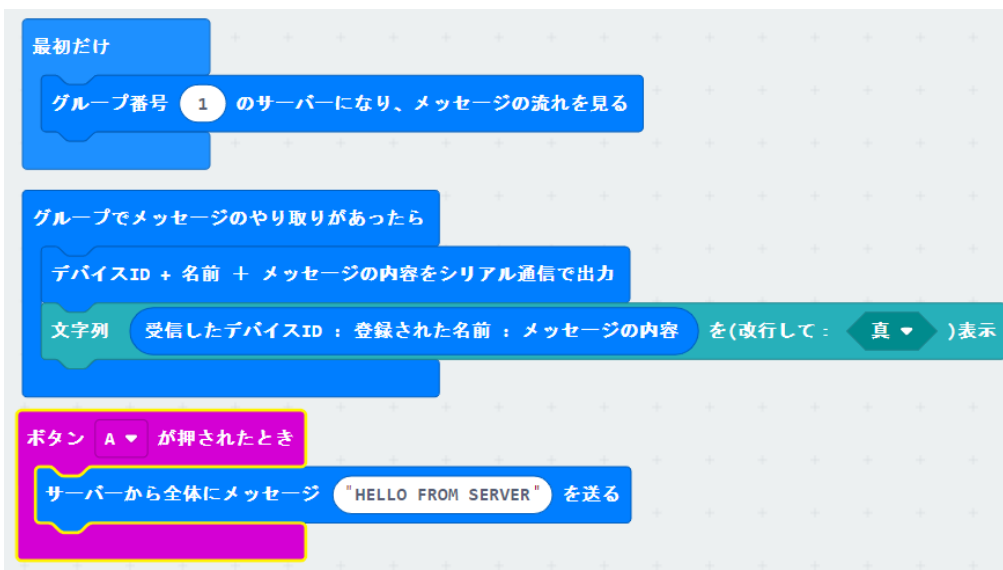
文字列 message を (改行して: 真) 表示

ボタン B が押されたとき

指定したID 1 にメッセージ "SECRET hello" を秘匿送信 (英数字のみ17文字まで)

- グループ1でIDを2、名前をyamada
- ボタンAが押されたらグループ全体に“HELLO”を送信
- メッセージを受信したらOLED (SSD1306) に「相手の名前 : メッセージ」を表示 (誰が何を送ってきたかわかるようにする)
- ボタンBを押すと ID: 1だけに“SECRET hello”を送信する

サンプルプログラム3



- グループ1 のサーバーになる
- グループ内でメッセージのやり取りがあったら シリアルモニタで内容を見る
- OLED上(ssd1306)でも内容を見る
- ボタンAを押すとグループ全体に“HELLO FROM SERVER”を送信

改良のヒント



- 受信したら音を出す
- OLED(SSD1306)に受信した名前とメッセージを表示
- ボタンAを押したときグループ全体に“OK”を送信