

CHAT アプリ仕様書

目的：ネットワークを用いた双方向性のあるコンテンツのプログラミングの実現

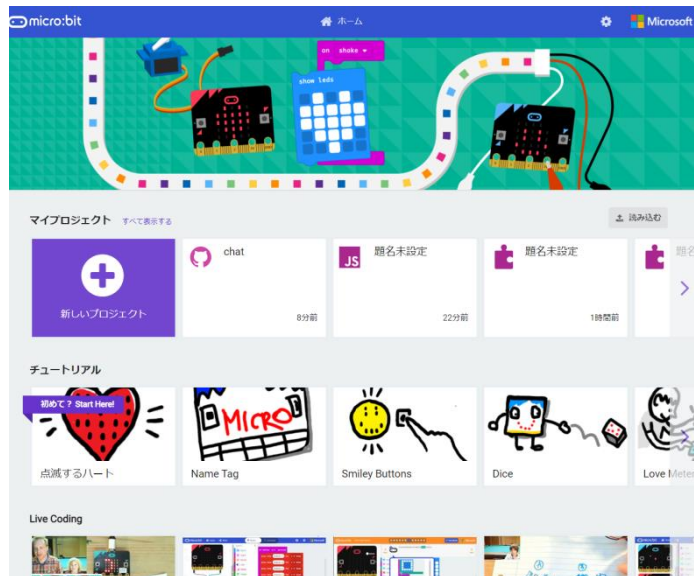
製作理由:ネットワークを用いた双方向的なプログラムの実現には多くの課題があり、実際に校内ネットワーク上に構築することの難しさや、課題、また GIGA スクール構想による多数の Chromebook が採用されたことでオンライン上でプログラム出来て、PC と有線で接続接続すること（無線経由では難しさやペアリングの難しさがある）などからマイコン上で完結し学習できることが有効ではないかと考え作成した。

考えられる用途：中学校技術 D:情報の技術 双方向的なコンテンツを用いたプログラミングなど

次のページ以降使い方の解説を掲載

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

使用方法: <https://makecode.microbit.org/#> の新しいプロジェクトを開き適当な名前を付けたあと作成ボタンを押す。



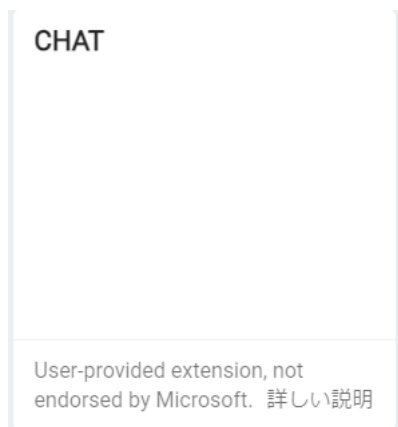
このような画面が出たら、高度なブロックボタンを押して展開、拡張機能ボタンを押す。

そこに

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

と打ち込む（もしくはコピー&ペースト）

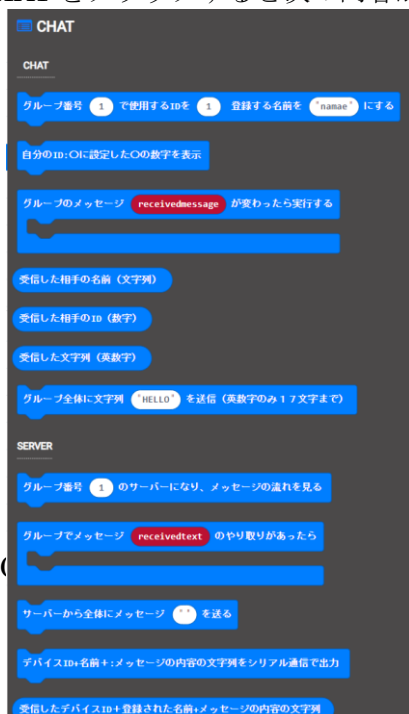
すると



このようなブロックが出現するのでクリック、元の画面に戻り、青いブロックの **CHAT** が出現しているのを確認する。



CHAT をクリックすると次の内容が出現する。



CHAT: 主に使用する機能。メッセージのやりとりを micro:bit 上で行う。

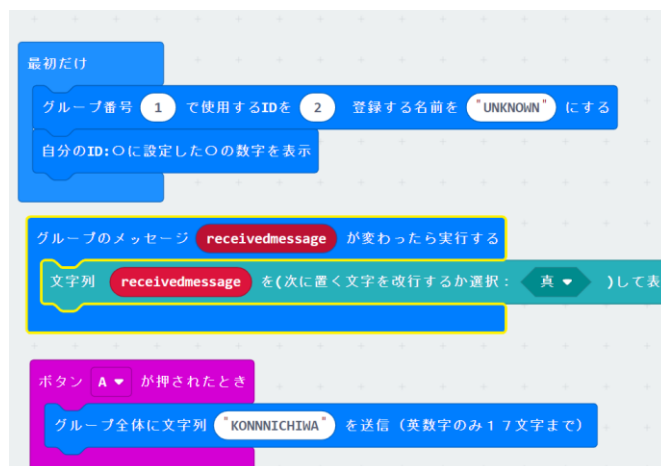
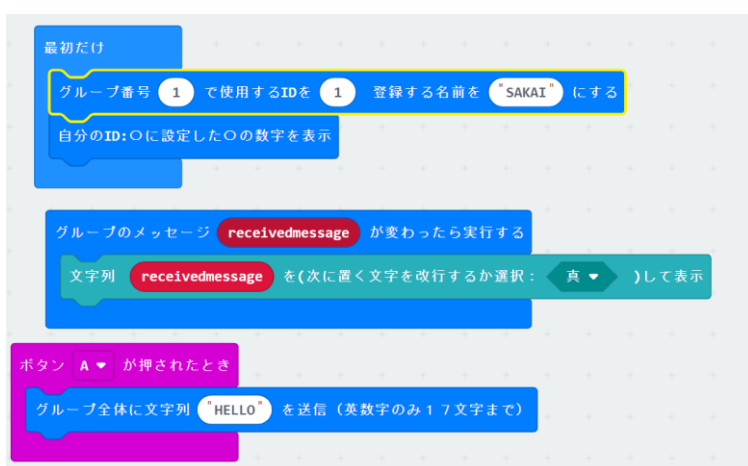
SERVER: なくても機能するが、全体のメッセージを見たい時に使用を推奨。SSD1306 などの OLED を使用しての使用を推奨する。無い場合は **SERVER** モードの端末を用意しメッセージの全容を見やすくするなどをするとうい。

細かい機能説明は次のページより

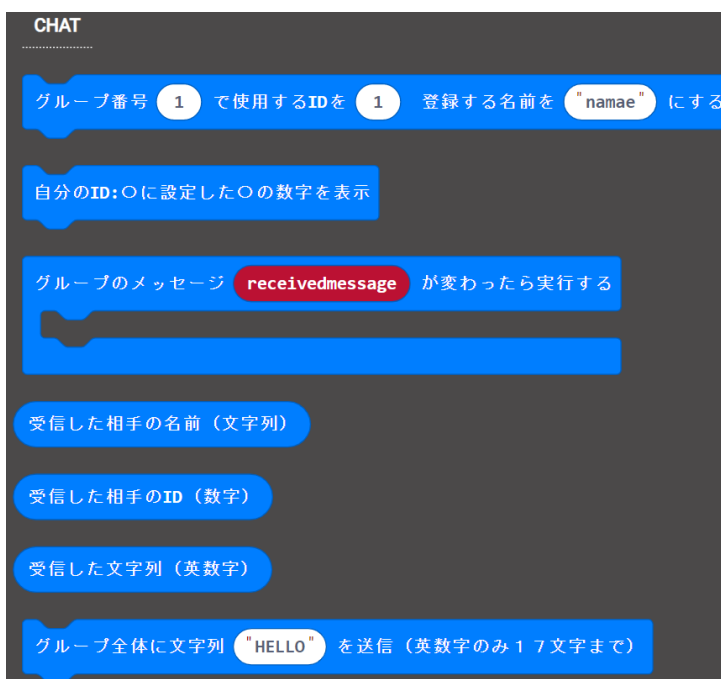
CHAT アプリの基本使用方法 この拡張は相互にメッセージをやり取りするものとなっている。

グループ番号(数字) 例：1 ID (数字) 例：1 登録する名前を任意の文字列例："SAKAI" などに設定する。

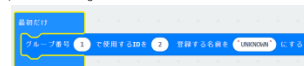
下の画像は左がグループ1, ID1, 名前をSAKAI 右の画像がグループ1 ID 2 名前を UNKNOWN に設定している。このプログラムを書きこむと micro:bit 本体の LED に登録した ID 番号が表示され、A ボタンを押すとグループ全体（ここでは設定した1）にメッセージを送信する。この文字列は通常の micro:bit の LED では見づらいため SSD1306 の拡張 (<https://github.com/sakai-code/oled-ssd1306>) を導入し文字列を表示している。



各機能：



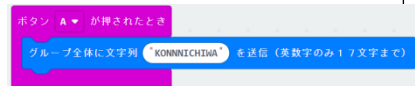
- ・最初に置くブロック、グループ番号、ID、名前を登録する。グループ番号は送る物同士同じにすること。



- ・自分の設定した ID を micro:bit 上の LED に表示

- ・グループ内のメッセージでやり取りがあったらこのブロック内のプログラムが動作する。赤い receivedmessage は受信した文字列本体。

- ・この3つのブロックは受信したメッセージ以外にも相手の ID や名前などを表示したい時に使用する。より使いやすいように改善出来る。



- ・同じグループ内に文字列（例：ABCD）などを送信する。ボタン A が押された時ブロック内などに置く。

サーバーアプリの基本使用方法

同じグループ内にいる人の ID, 名前、メッセージの流れなどを見ることが出来る。無くても動作するが、グループ内のメッセージの流れを見る際に有効。また、シリアル通信出力で Makecode 上のシリアルモニタ欄に表示する仕様にすることが出来る。

基本使用方法： 下の図はグループ 1 のサーバーになり、グループ内のメッセージのやり取りがあったらシリアルモニタに表示するプログラムです。A ボタンを押すとグループ内全体にメッセージ（“HELLO FROM SERVER”）を送るプログラムとなっています。

使用例：



※シリアルモニタの使い方

micro:bit を PC と USB で接続した後

左下のダウンロードボタンの右側・・・ボタンをクリックしデバイスを接続するをクリック。すると左上に下の表示が出るのでクリックする。

メッセージが送信されるとコンソールを表示ボタンが表示されるので、クリックすると表示される。

```
===DEVICEID: 1===|===OOZORS===|===HELLO===|
|=====|=====|=====|
RECEIVED!
|...[DEVICE_ID]...|...[NAME]...|...[MESSAGE]...|
|===DEVICEID: 1===|===OOZORS===|===HELLO===|
|=====|=====|=====|
```

各機能：

SERVER

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

グループでメッセージ receivedtext のやり取りがあったら

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

デバイスID + 名前 + メッセージの内容の文字列をシリアル通信で出力

受信したデバイスID + 登録された名前 + メッセージの内容の文字列

- ・グループを指定しサーバーになる。最初だけブロックに置きます。
- ・メッセージのやり取りがあったら、このブロック内のプログラムが動作します。receivedtext は受け取ったメッセージの中身。
- ・サーバーからグループ内にメッセージを送信する
- ・各種情報を前ページのシリアルコンソールを用いてメッセージの流れを見ることが出来ます。
- ・デバイス ID+名前+メッセージの文字列。拡張の液晶等を使うと見やすくなる。

シリアルモニタの見方

RECEIVED!
...[DEVICE_ID]...	...[NAME]...[MESSAGE].....
===DEVICEID: 2===	===SATOU===	===HELLO===
=====	=====	=====
RECEIVED!		
...[DEVICE_ID]...	...[NAME]...[MESSAGE].....
===DEVICEID: 1===	===YAMADA===	===HELLO===
=====	=====	=====
RECEIVED!

受信したデバイスの ID	受信した名	メッセージ
--------------	-------	-------

改良の仕方の一例：メッセージが来たら接続したモーター、内臓の LED、スピーカーなどが動作するようにする、通知が鳴る。OLED などの液晶にメッセージを表示し、伝言板のようにする。例：

最初だけ

グループ番号 1 で使用するIDを 1 登録する名前を "SAKAI" にする

グループのメッセージ receivedmessage が変わったら実行する

音を鳴らす 高さ (Hz) 真ん中のド 長さ 1 拍

文字列 受信した相手の名前 (文字列) を(次に置く文字を改行するか選択： 偽)として表示

文字列 ":" を(次に置く文字を改行するか選択： 偽)として表示

文字列 receivedmessage を(次に置く文字を改行するか選択： 偽)として表示

ボタン A が押されたとき

グループ全体に文字列 "OK SAKAI READY" を送信 (英数字のみ 17文字まで)

グループ 1 ID 1 名前 SAKAI

メッセージが来たら相手の名前とメッセージを OLED 上に表示する

A ボタンを押すと
OK SAKAI READY を返す。