

# MicJack (マイク・ジャック)

CC BY Michio Ono (Micono Utilities)

IchigoJamとつないで、インターネットにつなげて使えるMixJuice互換のESP-WROOM-02（以降、**ESP8266**）用のプログラムです。また最新バージョンでは、ESP-WROOM-32（以降、**ESP32**）にも対応しました。ESP8266を搭載したモジュールの他、ESP32を搭載したM5Stackなどに入れて使うことができます。またMicJackは、トイドローンのTelloの操作を行うためのコマンドにも対応しています。

注）なお、MicJackのご利用は自己責任においてご利用ください。

イチゴジャムシピさんのサイトでMicJackに関しての説明のページができました。

<https://15jamrecipe.jimdo.com/mixjuice/micjack/>

## [更新内容]

### v1.2.0b1

(新規) MJ SETRTC (M5StickCのみ：RTCの時間を設定) コマンドを追加

(新規) MJ GETRTC (日時を取得) コマンドを追加

(新規) M5Stack / M5StickC版で、CardKBユニットまたはFACESによる文字入力に対応

(修正) UDPのSoftAP接続に関する不具合を修正した

### v1.2.0b1

(新規) ESP32系 (ESP-WROOM-32モジュール、M5Stack, M5StickC) 版を作成

## 注意：

1、SPIFFSを使用しています。SPIFFSにデーターを転送するためのセットアップを行う必要があります。以下のURLの**ESP8266**系：「[Arduino IDE に ESP8266 SPIFFS ファイルシステムアップローダーをインストールする方法](#)」、**ESP32**系：「[ESP32-WROOM-32 SPIFFS アップローダープラグインの使い方](#)」などを参考にしてください。

CC BY IchigoJam <http://ichigoJam.net>

CC BY IchigoJam-for-Display-1.2 font <http://ijutilities.micutil.com>

CC BY IchigoLatte-for-Display font <http://il.micutil.com>

# ESPWriterを使ったインストール

MicJackのプログラムを簡単に転送できるように、ESPWriterを作りました。同梱の「Installer」という名前のフォルダにはいっています。

1、コンピュータをUSBシリアル変換器などで繋いで、ダウンロードモードにする。

ESP8266のTXとUSBシリアルのRX

ESP8266のRXとUSBシリアルのTX

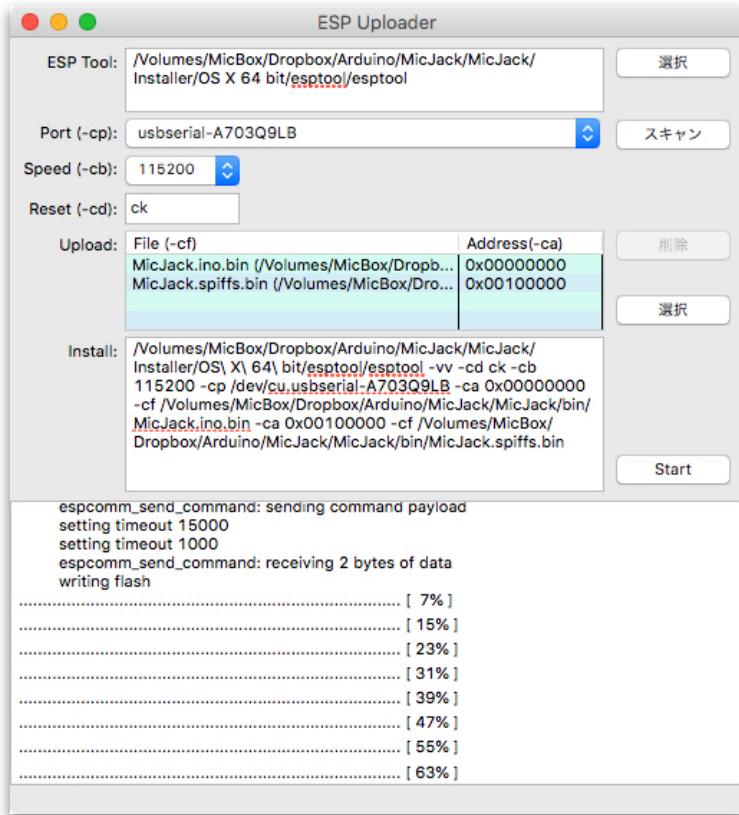
ESP8266のGNDとUSBシリアルのGND

ESP8266のIO0とGNDを接続

をそれぞれ接続した後に電源を入れて下さい。



2、Installerのフォルダの中に、Linx64bit版、OS X 64bit版、Windows版のESPWriterが同梱してあります。お使いのコンピュータのESPWriterを起動してください。



3、ESP8266と繋げているUSBシリアル変換器のPortを選択します。

4、「Start」をクリックすると、MicJackのプログラムの転送が開始されます。

- ・転送には1～2分かかります。

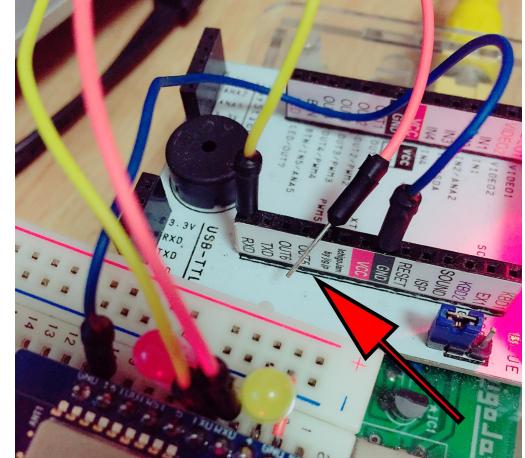
重要：ESP8266のリビジョンによっては、この設定だとSoft WDT Resetがかかってうまく動かない場合があります。そのような場合は、Uploadの所の選択で、binフォルダに入ってる **installsetting.csv** というファイルを選択したのちに、転送してみてください。

# Arduino IDEを使ったインストール ESP8266編 1

## 『MicJackのプログラムの転送』

ArduinoIDEを使ってESP8266にMicJack.inoを書き込んでください。もし、MicJack.inoをMixJuiceに入れる場合は、各自でMixJuiceの復元ができるかをチェックした後に行ってください。

MicJuckをインストールの際は、MixJuiceの場合は、IchigoJamなどから抜いた状態で転送を行って下さい。また、スイッチサイエンスで扱ってるESP8266開発ボードなどを使っている場合は、RXD (ESP側)・TX (IchigoJam側) を繋げないで、ソケットから外して下さい、転送終了後に繋げてください。



### 【接続】

ESP8266のTXとIchigoJamのRX

ESP8266のRXとIchigoJamのTX

ESP8266のGNDとIchigoJamのGND

をそれぞれ接続して下さい。

接続およびアクセス状況を示すLEDを使用する場合は

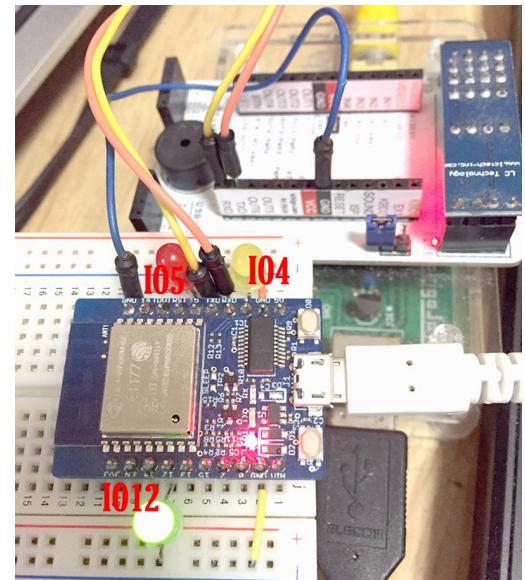
ESP8266の

IO12 (WiFiの接続状況)

IO4 (データ送信時)

IO5 (データ受信)

にLEDを取り付けて、MicJack.inoの中のコメントになって  
いる「#define useMJLED」のコメントを外して下さい。



# Arduino IDEを使ったインストール ESP8266編 2

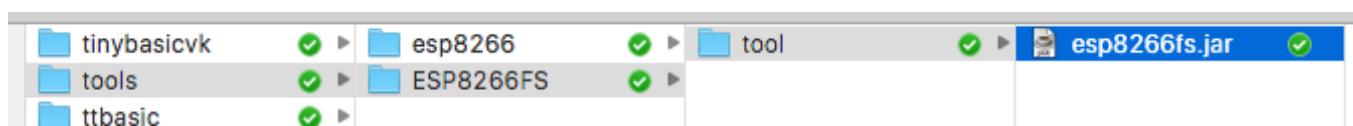
## 『スマフォを使ったキー入力のために』

スマフォなどからキーボード入力するためのファイルをESP8266に転送します。  
以下の手順に従って行って下さい。

### 1、SPIFFSにデータを転送できるようにする

dataフォルダに入れたキーボード入力のためのデータはESP8266のSPIFFSに保存します。そこでSPIFFSにデータを転送するためのセットアップを行う必要があります。以下のURLの「Arduino IDE にESP8266 SPIFFS ファイルシステムアップローダーをインストールする方法」などを参考にしてください。

<https://www.mgo-tec.com/spiffs-filesystem-uploader01-html>



### 2、データをSPIFFSに転送する

MicJackのスケッチをフロントウインドウになっている状態で、「ツール」メニューの中の「ESP8266 Sketch Data Upload」を選択します。この際に以下の点に注意してください。

- ①Flash Size のSPIFFSはdataフォルダ内のファイルの合計より大きなサイズを選択してください。大きい設定ほど転送に時間がかかります。例えば最低2M(1M SPIFFS)くらいのサイズが必要です。
- ②Upload Speedは115200bpsを選んでください。
- ③シリアルモニターウィンドウが開いてると転送に失敗しますので、必ず閉じて下さい。
- ④以上のこと注意して「ESP8266 Sketch Data Upload」でSPIFFSにデータを転送して下さい。

ESP8266のリビジョンによっては4M(1M SPIFFS)でないとSoft WDT Resetにより動かない場合があります。



ボード: "Generic ESP8266 Module"  
Flash Mode: "QIO"  
Flash Size: "4M (1M SPIFFS)"  
Debug port: "Disabled"  
Debug Level: "なし"  
IwIP Variant: "v2 Lower Memory"  
Reset Method: "ck"  
Crystal Frequency: "26 MHz"  
VTables: "Flash"  
Flash Frequency: "80MHz"  
CPU Frequency: "80 MHz"  
Built-in Led: "2"  
Upload Speed: "115200"

# 使い方

## 1、起動時モニターに表示されるもの

MicJackが起動した時に、シリアルモニターをみると、右のような情報が表示されます。内容は、①MicJackのバージョン、②SPIFFSに入っているファイルとサイズ（最新版では表示しない仕様になってます）、③SoftAPのSSID名とIPアドレス、④WiFiルーターのSSIDとローカルのIPアドレスおよびその場合のDNS名などになります。

注意：

- IchigoJamのモニターにはこれら全てちゃんと表示されることは限りませんので注意ください。
- WiFi Stationは最初は何も表示されないです、外部サイトとインターネットで通信しない場合は設定は必用ありません。
- 本来、MJ APC ssid名 Passwordのコマンドで接続設定を行いますが、もしうまく行かない場合は  
1、MJ SSID ssid名  
2、MJ PWD Password  
3、MJ APC  
で接続してみて下さい。

## 2、スマフォのブラウザでコントロールしてみる

SPIFFSにデータが転送してあれば、スマフォのgoogle chrome, firefox, safariなどでMicJackのHTTPサーバーにアクセスして、キー入力などが行えます。iOS10でズームできなくなる制限が効かなくなっていますので、chromeやfirefoxの方が使いやすいかもしれません。

スマフォのWiFi設定で接続を、上記の場合だとmicnetにします。切り替えができたら、ブラウザを開き、**WiFi無線ルータにつないでいる場合は**（上の図のようにされている場合だと）<http://192.168.0.16> にアクセスして下さい。OSによっては <http://micjack.local/> でアクセスできる場合があります。

**無線ルータのない環境**ではSoftAPの右にある名称、図の場合、micjackという名前のアクセスポイントをスマフォのWiFiの環境設定から探し選択し、パスワードの要請がきたらパスワードを入力してください。デフォルトではアクセスポイント名が**micjack**、パスワードが**abcd1234**になっています。正常にコネクトできたら、図のSoftAPの下に表示されてるIPアドレスにブラウザでアクセスします。上の図の場合だと、<http://192.168.4.1>になります。

```
'MixJuice-1.2.1
'MicJack-1.0.0
'CC BY Michio Ono

'SPIFFS files...
'/IchigoJam-for-Display-1.2.ttf:44.30KB
'/IchigoLatte-for-Display.ttf:59.88KB
'/ijkb.html:13.42KB
'/ijkbl.html:13.42KB
'/ijkbl2.html:13.41KB
'/ijkbs.html:13.38KB
'/ilkb.html:13.43KB
'/ilkbl.html:13.42KB
'/ilkbl2.html:13.41KB
'/ilkbs.html:13.38KB
'/index.html:9.69KB
'/quest.html:5.36KB

'Soft AP: MicJack
'IP: 192.168.4.1

'HTTP Server started...

'Connecting.....
'WiFi Station: micnet
'IP: 192.168.0.16
'http://micjack.local/

'Keyboard Mode

'Ready to go!
```



### 3、トップページ画面について

ブラウザでアクセスすると左の図のような画面が表示されます。

## IchigoJam Keyboard

- IchigoJam用の仮想キーボードのボタンが4つあります。S, M, L, LLはキーボードの高さが少しずつ違います。スマフォに合わせて使い分けください。

The screenshot shows the IchigoJam Command interface. On the left, a code editor window displays the following assembly-like code:

```
NEW
10 LED1
20 WAIT30
30 LED0
40 WAIT30
50 GOTO10
```

On the right, there are two buttons: "Clear" and "SEND".

- ESP8266 IO13とIO16をIchigoJamなどのKBD1とKBD2に繋げて、**KBD**を選択してあれば、これらのページから送信したデータは、キーボードとして入力として認識します。IchigoJamの場合は**IJKBD**を選択して下さい。

MixJuice/MicJack Command

- IchigoJamのコマンドが送信できます。改行にも対応していますので、プログラムを打ち込んで送信するのにも便利です。

●MixJuice, MicJackコマンドが送信できます。コマンドの書式は、MJを付ける必要はありません。例えば、"GKP 3"とするだけで、Kidspodから3番のプログラムをダウンロードすることができます。

Get Quest journey from kidspod

539

GET

- KidspodからIchigoQuestのプログラムをダウンロードできます。番号を入れてGETボタンを押すだけです。

Post Quest journey to kidspod

553:hr3dtg

POST

- KidspodにIchigoQuestのプログラムをアップロードします。投稿先の番号とそのパスワードを入力してPOSTボタンを押すだけです。

Send Quest hex journey to IchigonQuest

12000000000000000018000000 000000000900010010000000 080003000000000000B040100 000000000800040000000000 0A0401000000000015010100	<b>Clear</b>
	<b>SEND</b>

- IchigonQuestのHEXプログラムを入力してIchigonQuestに送信することができます。プログラムの終端がFFになっていればFF000…で埋める必要はありません。

## Controller

●大きめのコントローラを表示します。

IchigoLatte Keyboard

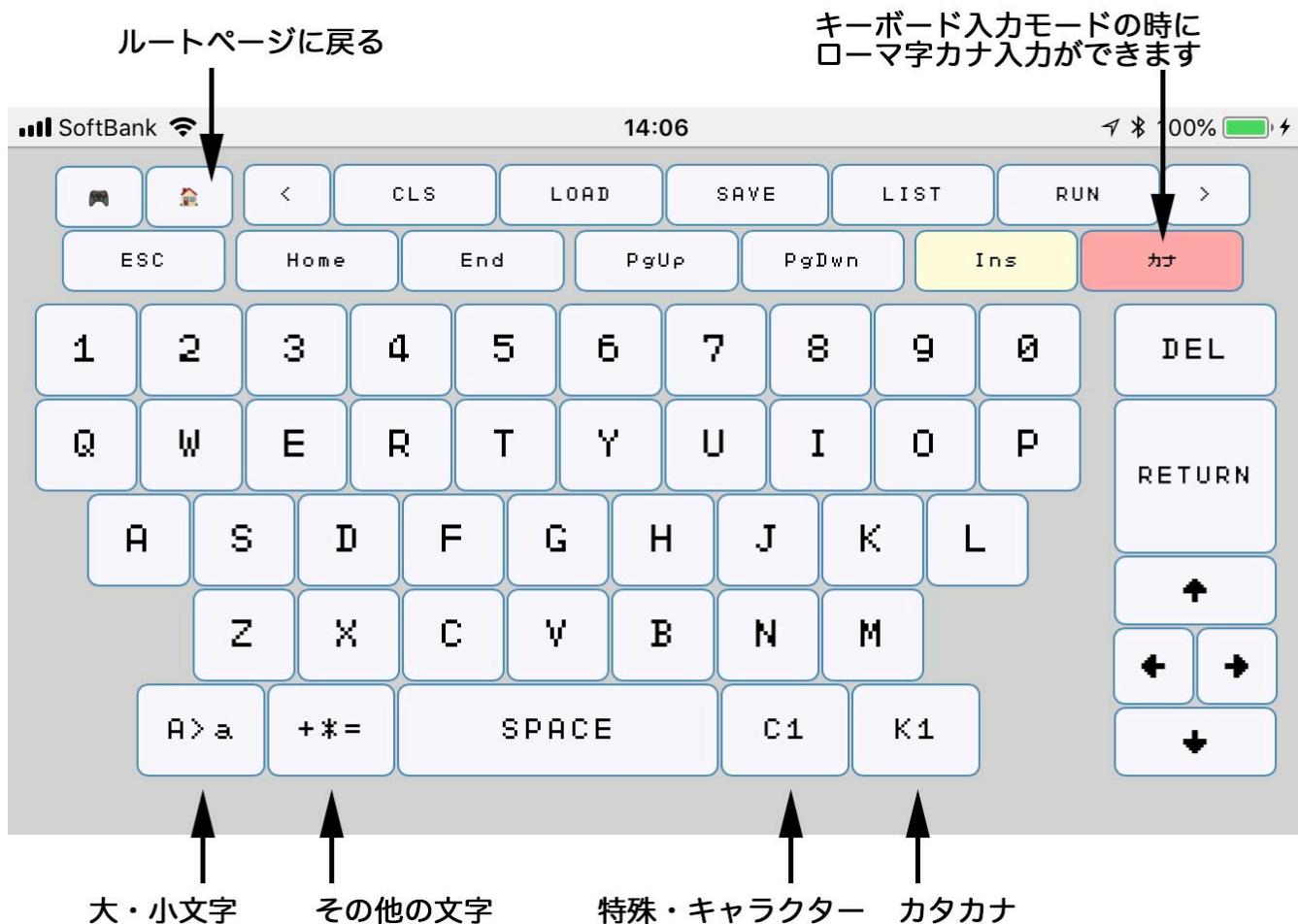
● IchigoJamのコマンドでLED1,LED0を送信します。

S M L LL

●IchigoLatte用の仮想キーボードです。

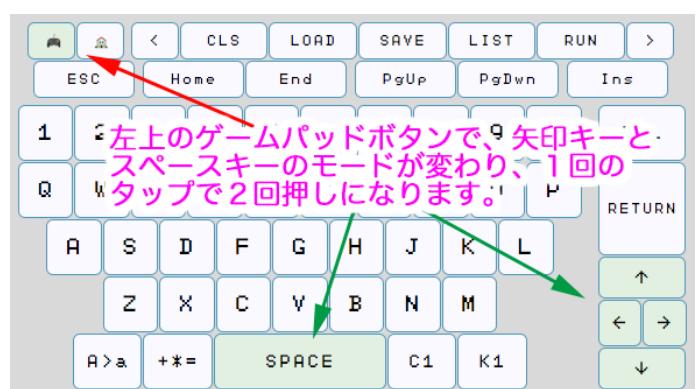
#### 4、キーボード画面について

キーボードを表示させると以下の図のような画面になります。文字の種類は切り替えのボタンで切り替えて表示させられます。キャラクターやカタカナのボタンは、1つの画面ですべてを表示させきれないのと、何回か押すことによってさらに切り替わります。



キーボードの左上にゲームパッドのアイコンのボタンがあります。このボタンをタップすると矢印キーとスペースキーのモードが変わります。1回のタップで2回押しになります。これはUART送信の場合の、BTN(SPACE), BTN(LEFT), BTN(UP), BTN(DOWN), BTN(RIGHT)に対応した機能になります。

KBD1/KBD2に接続している場合は、大部分のキーはPS/2キーボード信号として送信されます。従って、カナは「カナ」キーを押していただくと、ローマ字カナ変換で入力が可能です。



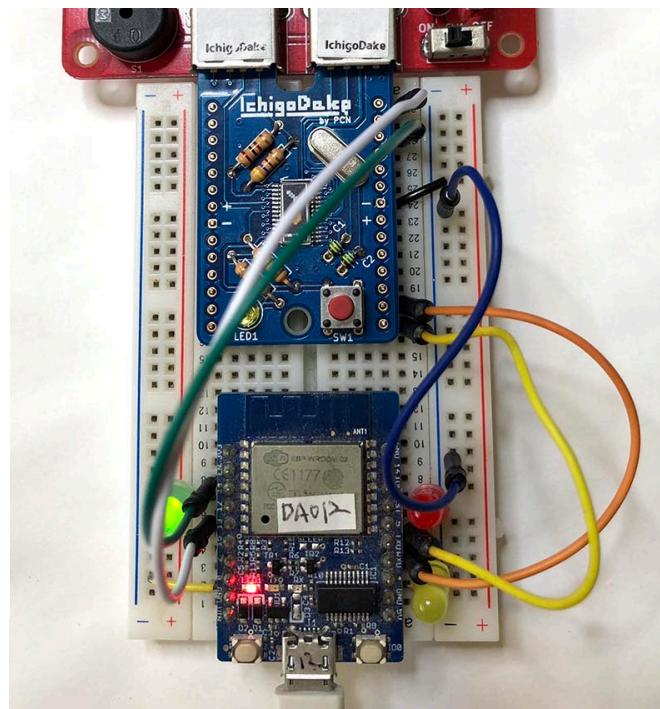
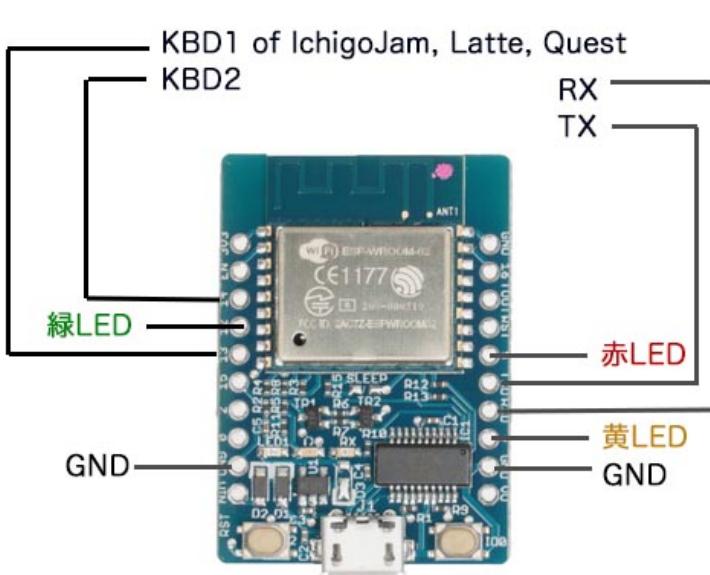
## 5、キーボード入力を使う場合の接続

現在のプログラムでは、KB\_CLK (KBD1)をIO13番ピン、KB\_DATA (KBD2)をIO16番ピンを使う設定になっています。MixJuiceでは、あとIO14番ピンが空いているので、それを使う場合はプログラムの方を書き換えて下さい。

```
121 #include <ps2dev.h>
122 #define KB_CLK      13 // 0 // A4 // PS/2 CLK IchigoJamのKBD1に接続 //21//
123 #define KB_DATA     16 // 15 // A5 // PS/2 DATA IchigoJamのKBD2に接続 //22//
124 uint8_t enabled =0;                                // PS/2 ホスト送信可能状態
```

なお、MixJuiceの場合は、IO13, 14, 16番は引き出されてないため、直接半田付けが必要になりますので、スキルのある方は挑戦して下さい。また、MicJackからキーボード入力する場合は、本物のキーボードをさしたままにもできません。またその逆に、本物のキーボードを使う場合は、13,14または16番ピンは抜いて使う必要がありますので、半田付けする場合は、KBD1/2に抜き差しできるように半田付けする必要があります。以下は13と14を繋げた場合の説明図です。

以下はESPr DeveloperボードのIO13番をKBD1に、IO14番にKBD2を繋いだ場合の写真になります。



## 6、SoftAPを独自のものに変更する。

SoftAPの設定は、デフォルトではアクセスポイント名が**micjack**、パスワードが**abcd1234**になっています。SoftAPの名称とパスワードを変更する場合は、

**??"MJ SOFTAP SSID名 パスワード"**

で変更できます。この場合、**パスワードは8文字以上**でないと設定できませんのでご注意ください。なお、設定状態は、**??"MJ SOFTAP"**とだけ入力することでSSID名とIPを表示させられます（パスワードは表示しません）。

# KidspodにIchigonQuestのプログラムをPOST/GETする場合

1、IchigonQuestはMicJackの情報を画面表示できませんので、WiFiアクセスポイントの設定を画面表示なしでできないわけではありませんが、あらかじめIchigoJamなどでデフォルトの接続先を設定しておくといいです。

2、MixJuiceを使っている場合は、そのまま積み重ねるだけですが、その他のESP8266モジュールを使っている場合は、IchigonQuestのTX/RXと、MicJack(ESP8266)のTX/RXをクロスで繋ぐ（前ページの写真を参考にして下さい）。

IchigonQuestのTX と MicJack(ESP8266)のRX  
IchigonQuestのRX と MicJack(ESP8266)のTX

3、起動後、もし、緑LEDがついてなければ、

1、WiFi設定で、MicJackを選択

2、スマフォ等のブラウザでSoftAPのIPにアクセス

例：192.168.4.1

3、MixJuice/MicJack CommandのところでAPCと入力して送信

4、うまくいけば、緑LEDが点灯する

4、WiFi設定をWiFiステーションのSSIDに変える

例：micnet

5、ブラウザでアクセスする

例：192.168.0.16

6、POSTする場合

① Post Quest journey to kidspodに、POST先の番号とパスワードを入力して、「POST」ボタンを押す。

Post Quest journey to kidspod

554:HO1GE3

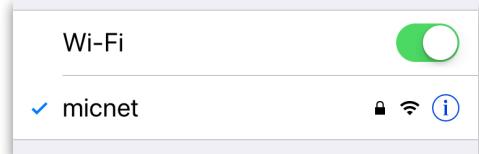
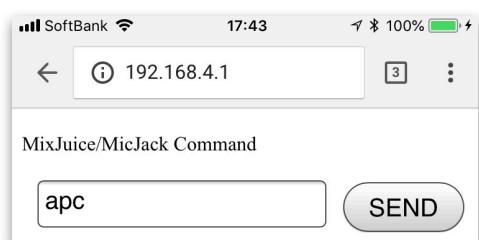
POST

② IchigonQuestで、送る操作をする。



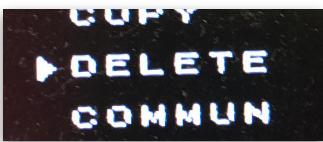
512バイト送信後、自動的にPOST ENDされます。

ポストされるプログラムは、英語版のプログラムに変換したものとHEXに変換したものがアップロードされます。



## 8、GETする場合

- ① 受信する前にいちおうプログラムを消して下さい。



- ② IchigonQuestで、受信操作を行う。



- ③ Get Quest journey from kidspodに、GETするプログラムの番号を入力して、「GET」ボタンを押す。

Get Quest journey from kidspod

行の最初からHEXのプログラムがある場合は、先頭から1024文字（512バイト）分を読み込んで、バイナリーに変換して転送しますが、最初がHEXプログラムでない場合は、”[“がマークです。”[“のある次の行からをHEXのプログラムとしてバイナリーに変換してIchigonQuestに転送します。

HEXデータは、改行なしの連續データで、最後がFFになっていれば、その後1024文字分まで埋まってなくても大丈夫なようにしてあります。

SOURCE CODE

1.	Turn on LED
2.	Wait 60 times
3.	Turn off LED
4.	Wait 1 sec
5.	Go to 1
6.	
7.	[HEX]
8.	01000100000000004003C00010000001000000000000000400E803000000000500000000000000FF 0000

# ESP32版の簡単な説明

現在、ボード設定で、ESP32 Dev Module、M5Stack-Core-ESP32、M5Stick-Cに対応したESP32機器対応のソースおよびbinファイルを同梱しています。

## ●Arduino IDEを使ってご自分でインストールされる場合

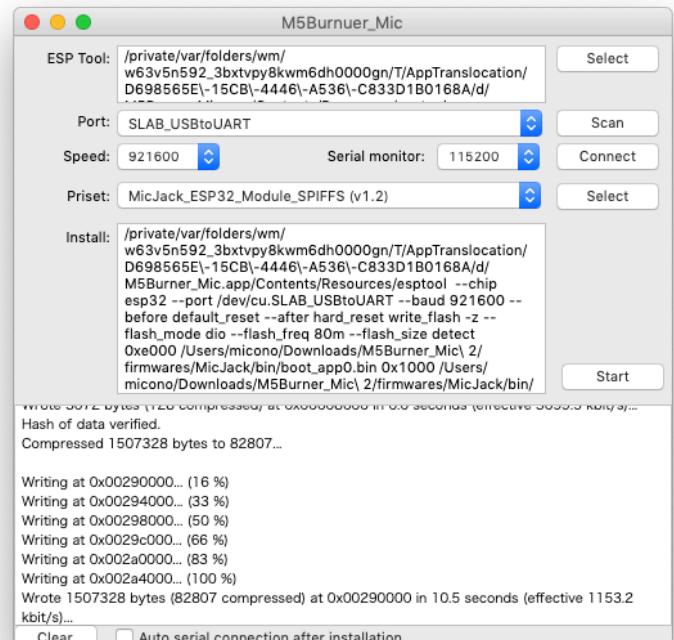
Arduino IDEを使ってご自分でインストールされる場合は、MicJackフォルダの中のBoard Settingフォルダの中に、それぞれの設定のスクショが入っていますので参考にして下さい。また、microSDがないものでは、Dataフォルダに入っているふあいるをSPIFFSに入れる必要があるので、「[ESP32-WROOM-32 SPIFFSアップローダープラグインの使い方](#)」を参考にやって下さい。

## ●M5Burner\_Micを使ってインストールする場合

ESP32用のインストーラーは同梱していません。M5Burner\_MicというESP32用のインストーラーを作っています。M5Burner\_Micは[https://github.com/micutil/M5Burner\\_Mic](https://github.com/micutil/M5Burner_Mic)からダウンロードできます。「Download here (20xx/xx/xx)」と書かれたところがあるので、そこをクリックしてダウンロードして下さい。なお、直接のダウンロードのリンクは[http://micutil.com/download/M5Burner\\_Mic.zip](http://micutil.com/download/M5Burner_Mic.zip)になります。

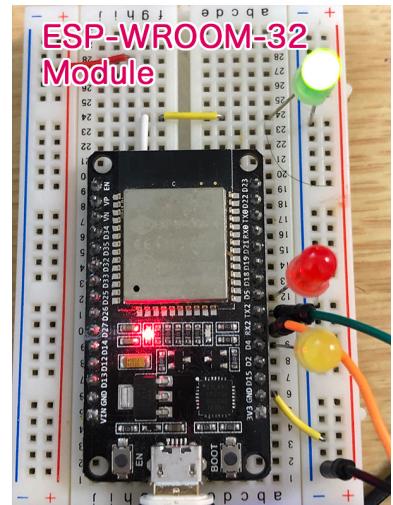
### M5Burner\_Micの使い方とESP32系MicJackのインストール手順

1. MicJackのbinフォルダの中の**ESP32**フォルダの中の**MicJack**フォルダを、M5Burner\_Micの中の**firmwares**フォルダの中に入れます。
2. M5Burner\_Micを起動します。Macの場合は、**M5Burner\_Mic**を、Windowsの場合は、**M5Burner\_Mic.exe**をダブルクリックで起動します。起動の際、セキュリティの関係で引っかかりメッセージが出ますので、そのような場合の、Macの場合、Windowsの場合、それぞれの動かし方にしたがって起動させて下さい。
3. Macの場合、アプリの実態は全然別の場所に存在した状態で起動するため、firmwaresフォルダを見つけることができないので、firmwaresフォルダを選択するダイアログがでますので、M5Burner\_Micフォルダの中に入っているfirmwaresフォルダを選択して下さい。
4. ESP Toolは自動的に選択済みです。
5. ポートは、インストール先のESP32機器のUSBシリアルを選択して下さい。
6. 転送スピード (Speed) は、デフォルトの921600bpsで大丈夫です。
7. プリセットで、MicJackを選択します。現在、4タイプを選ぶことができます。M5StackはmicroSD版とSPIFFS版と選ぶことができます。microSD版の場合、microSDにDataフォルダの中のファイルをmicroSDにコピーして下さい。
8. Startボタンをクリックすると、インストールがはじまります。  
以上です。



### ●ESP32機器とIchigoJamとの接続

**TX/RX** : ESP32版の場合、TX/RXは、**TX2/RX2**でIchigoJamと通信するように設定されています。ESP32モジュールやM5Stackでは、**G16 (RX2)**と**G17 (TX2)**をIchigoJamのそれぞれTXとRXと繋げて下さい。  
M5Stack fireやM5GOの場合はGroveのUARTが**G16 (RX2)**と**G17 (TX2)**となっています。また、M5StickCは、**G0**と**G26**を**RX2**と**TX2**に指定してます。



**GND** : ESP32機器とIchigoJamのGNDを繋げるのを忘れないようにして下さい。

**LED**：状態を示すLEDは、ESP32モジュールでは、**接続LED（緑）**：G12、**ポストLED（黄）**：G4、**ゲットLED（赤）**：G5、に接続しています。M5Stack、M5StickCは、GPIOピンには出力していません。M5Stack、M5StickCは画面上で表示するようにプログラムしてあります。

### ●M5StackとM5Stickのボタン

**Aボタン (M5Stack・M5StickC) :** StationモードでWi-Fiルーターに接続している場合、スマホ、パソコンなどから入力できるような、QRコードを表示します。



**Bボタン (M5Stack)** : ボタンを押すたびに、8ピクセル・16ピクセルと文字サイズが交互に切り替わります。

**Bボタン (M5StickC)** : Aボタンで表示したQRコードを消します。



## ●M5Stackの文字表示

M5Stackで表示される文字はIchigoJam v1.2に対応する文字です。なお、表示される文字は、IchigoJamの画面と同じ状態を表示しているものではなく、ターミナル風な画面になっていますので、単に文字が下から上に流れてい感じになります。

## **[リファレンス]**

MJRefer.pdfファイルをご覧下さい。

## **[リンク]**

IchigoJam

<http://ichigojam.net>

MixJuice

<http://mixjuice.shizentai.jp>

## **[使用と免責]**

現在、CC BY MicJack / Michio Ono として公開しました。

自己責任においてご利用ください。

改良した場合は、改良したスケッチを下さい。

このスケッチを利用したことによるいかなるトラブルに関して

一切の責任を負いません。

## **[お問い合わせ]**

micono@mac.com

<http://ijutilities.micutil.com>

## **[履歴]**

v1.1.0b2

(修正) いくつかのTelloドローンのコマンドで上手く動かないものを修正しました。

(変更) REGXXXのコマンド名を、RGA, RGL, RGC, RGDに変更した。

(更新) ESPWriter マック版を更新

v1.1.0b1

(新規) Telloドローンのコントロールに対応しました（現行でEdu未対応）。

(新規) 8個のアクセス先を登録できるようにしました。

(変更) UDPの仕様をMixJuiceに合わせて少し変更しました。

(変更) アクセスポイントSSID名をMixJuiceに合わせてMicJackからMJ-XXXXXXXXXXXXXに変更しました (XX..には MixJuice の Mac アドレスが入ります)。

v0.10.1b2

(新規) MJ GETSLASTの不具合修正

v0.10.1b1

(新規) MJ IJKBD追加 : IchigoJamでキーボードモードを使う場合

(新規) MJ GETHOME追加 : GETHOMEのGETS対応版

(新規) MJ GETSLAST追加 : GETLASTのGETS対応版

## v0.10.0b5

(新規) MJ QSEND追加：HTMLからHEXプログラムをQuestに送信できるようにした

(新規) ESPWriterを同梱した。MicJackにプログラムを入れるためのアプリです。

## v0.10.0b3, b4 (未公開)

(新規) HTMLから複数行のIchigoJamコマンドを送信できるようにした

## v0.10.0b2

(回避) キーボード入力モードでSoft WDT Resetが起きる不具合を回避

## v0.10.0b1

(新規) キーボード入力信号機能追加 (MJ KBD / MJ UART)

(新規) IchigoQuestのプログラムをKidspodへアップロード機能追加 (MJ QGKP)

(新規) IchigoQuestのプログラムをKidspodへダウンロード機能追加 (MJ QPKP)

(新規) スマフォによる入力のプログラムの各種改良

## v0.9.0

(新規) IchigoLatte-for-Display.ttf (IchigoLatte文字用フォント) を同梱し、IchigoLatteのキーボードとして使える機能を追加した。

(新規) MJ GKP, MJ PKP の2つのコマンドを追加した。kidspod.clubにプログラムをアップロードダウンロードするコマンド入力を簡単にできます。

## v0.8.1

IchigoJam-for-Display-1.2.ttf (IchigoJam文字用フォント) を同梱した。

## v0.8.0

(新規)HTMLキーボード文字入力機能に関して、BTNコマンド（スペース、上下左右の矢印キー）に対応した2回押し機能を追加した。

(新規)PCT (POST CONTENT-TYPE)に対応しました。?"MJ PCT "でデフォルトのContent-Typeなしに相当します。

(修正)OTAでプログラムの書き換えにに関して不具合を修正しました。

(変更)GET, POSTでのヘッダーをMixJuiceにできるだけ合わせるように変更した。

(変更)GETでのデータの読み込みの際の文字転送速度に関して、デフォルト値を文字遅延20ms/改行遅延30msから文字遅延18ms/改行遅延18msに変更した。

(変更)コマンドエラーの場合に表示されるバージョンをMixJuiceに対応しているバージョンを表示するように変更しました。また?"MJ MJVer"でMicJackのバージョンを表示するように変更した。

## v0.6.0 (4)

(新規)SoftAPに対応しました。デフォルトの名称とパスワードは、MicJackとabcd1234になります。

(新規)MJ SOFTAP: SoftAPのSSID名とIPを表示、および、SSIDとパスワードを設定するコマンドを追加しました。

(新規)HTMLキーボード文字入力機能を追加しました（現在UARTによる通信となります）。スマフォでIchigoJamに文字入力ができます。

(更新)MJ SVR: サーバーを停止しました。起動時に立ち上ります。

## v0.5.0 (4)

(新規) MJ SPW: 転送速度設定対応

(新規) MJ GETS対応

(新規) MJ POSTS対応（未検証）

(更新) MJ VER 非対応に変更

ボクニモワカルMixJuiceの8>4>6、8>4>7の

画像転送対応のため

(更新) MJ SVR: サーバー対応 (仕様検討中)  
LEDオン・オフのデモページ

#### v0.4.0

(新規) MJ SVR: サーバー対応 (仕様検討中)  
(新規) MJ PMODE: pinModeを実行  
(新規) MJ DWRT: digitalWriteを実行  
(新規) MJ DREAD: digitalReadを実行  
(新規) MJ AWRT: analogWriteを実行  
(新規) MJ AREAD: analogReadを実行

#### v0.3.1

(新規) SSIDなどが設定されている場合、起動時に自動接続するようにした。

#### v0.3.0

(新規) MJ GETHOMEを追加  
(新規) MJ GETLASTを追加  
(変更) CC BY-SAをCC BYに変更  
(修正) MJ VERなどで表示されるバージョン番号の間違いを修正

#### v0.2.0

(新規) 接続およびアクセス状況表示のLEDに対応  
(新規) MJ SSID, MJ PWD, MJ MACADDRを追加  
(新規) SSID, パスワードの不揮発メモリーに記録するようにした  
(変更) MJ APLでSSID, パスワードを省略可能にした  
(変更) SetupのWifi.disconnectを削除  
(変更) MJ ???の場合のエラー表示  
(修正) MJ GETでのヘッダーの読み飛ばしの不具合の修正