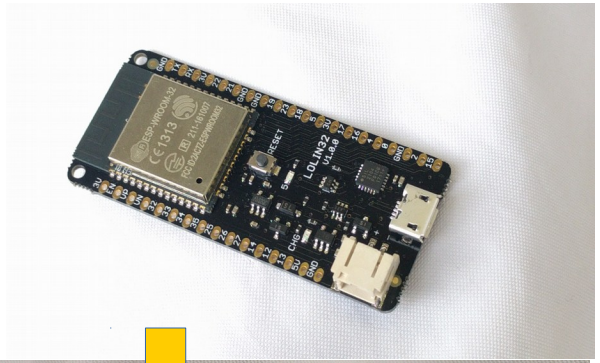
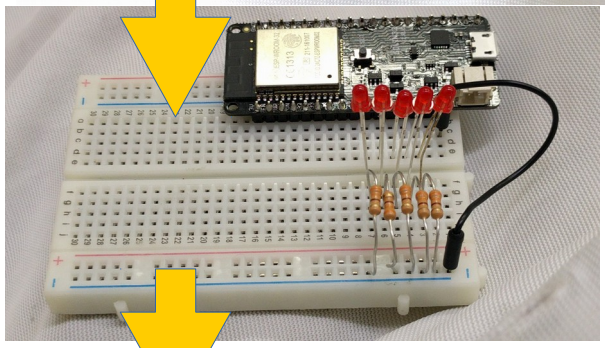
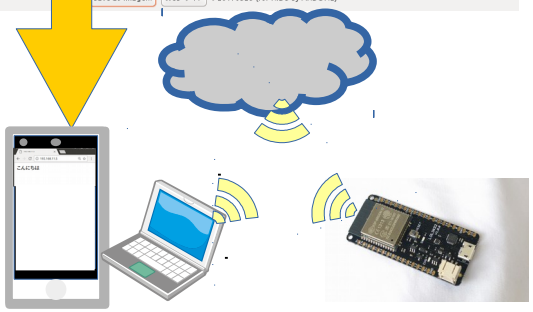
Let's start electronics making with ESP32 Tranning kit

ESP32ではじめる
はじめてのでんしこうさく



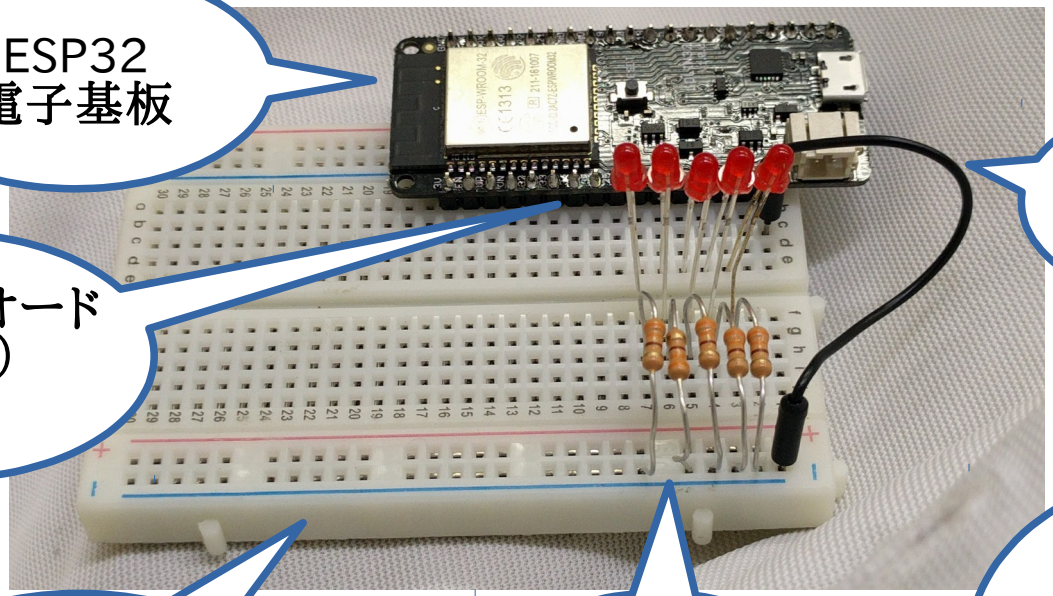




AKBROBOT
日本Androidの会秋葉原支部ロボット部･ジャパン

もくじ

実験キット
「AKBONE ESP32」



ESP32
電子基板

発光ダイオード
（LED）
数本

ブレッドボード

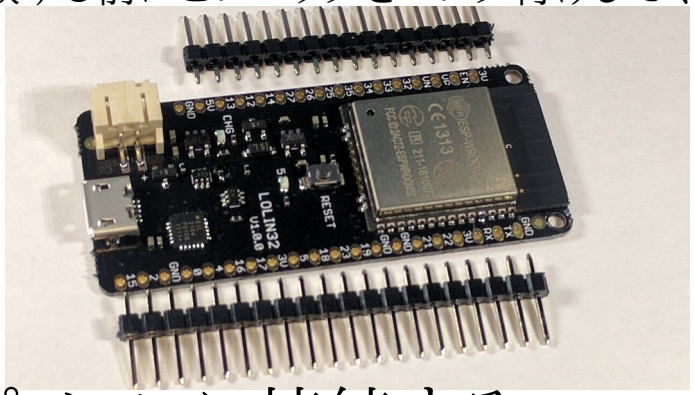
抵抗
数本

電線
数本

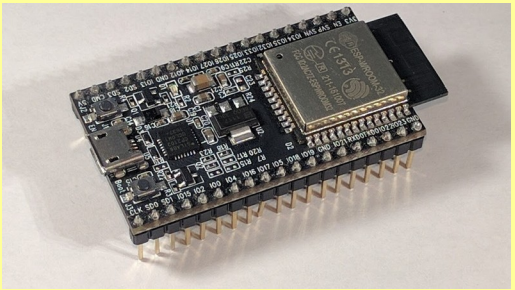
その他
スイッチ、センサー
など

ESP32ボードを組み立てる

キットには、WEMOS LOLIN32が入っています。
実験する前にピンヘッダをハンダ付けしてください



付属のLOLIN32ではなく、
ESP32−DevKitCを代わりに
使うこともできます
（はんだづけは必要ありません）



パソコンに接続する

MicroUSBケーブル(別売)を使ってパソコンに接続して下さい。
パソコンがMac OSXの場合は、以下からドライバをインストールして下さい。
http://jp.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx#mac

Arduino起動、ArduBlock起動

ワークショップ　その1

「ハードウェア」と「ソフトウェア」

「光の流れ」テキスト

日本アンドロイドの会秋葉原支部ロボット部

ESP32を使った小さな電子基板を
Arduino互換基板として使用して、
LEDを光るプログラムを作ります

メーリングリスト：　http://groups.google.com/group/robot-android-group-japan-akb

Webサイト：　https://sites.google.com/site/akbrobot/

「光の流れ」ワークショップ

概要

目的：　「ハードウェア」と「ソフトウェア」に触れる

概要：　光の流れをつくる

対象：　小学生以上

コース時間：　30分～

コース方式：　個別指導

時間

(1)コース説明　3分

(2)ハードウェア工作　10分

(3)課題1(1つ点滅)　5分

(4)課題2(光の流れ)　10分

(5)まとめ、質問　3分

(6)追加の問題(流れを逆にするなど)　15分

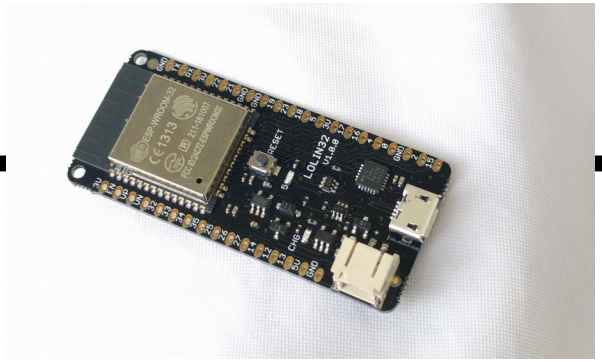
全体のつながり

パソコン

USB接続

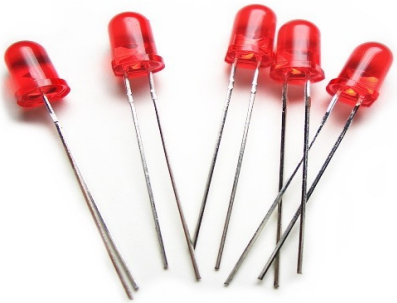
- ソフトウェアを作成する
- ソフトウェアを電子基板に送る
- 電気を供給する

電子基板



- ソフトウェアを保存する
- ソフトウェアを実行する

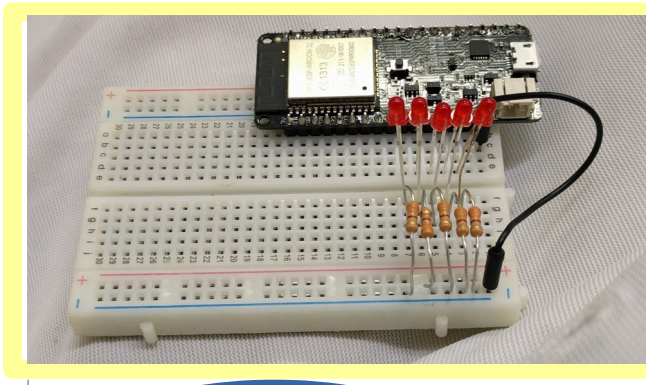
発光ダイオード (LED)

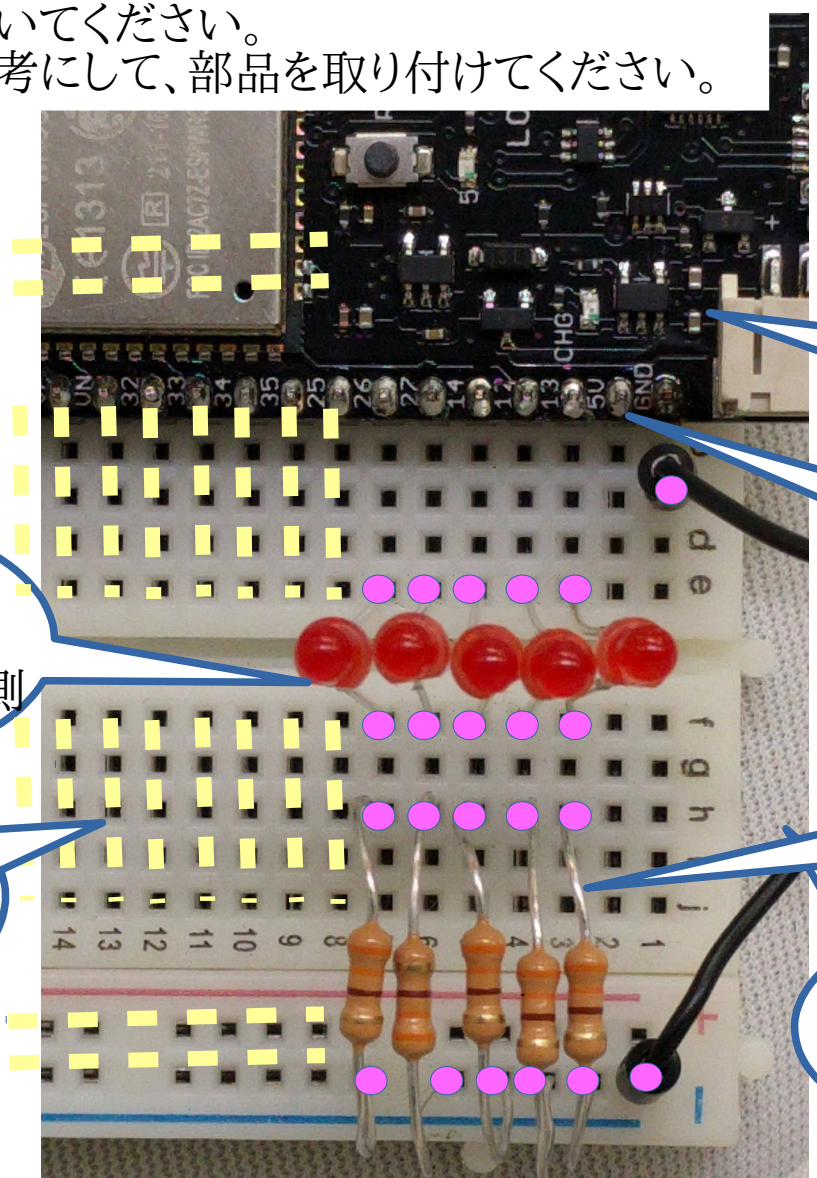


- 光る
- 光の流れをつくる

ハードウェア工作

①最初に先生が部品（抵抗、発光ダイオード、電線）を取り付けます。よく見ていてください。
②つぎに写真を参考にして、部品を取り付けてください。





電子基板
ソフトウェアを保存し、実行する

ピン
ソフトウェアで指定した電気を出力する

抵抗
電気を流れにくくする

電線
電気を通す

発光ダイオード（LED）
長い足がプラス側

ブレッドボード
黄色のようにつながっています

「デモ1」　1つ点滅

ソフトウェア1

実行ボタン
ソフトウェアをパソコンから
電子基板に送り、電子基板
上で実行します。

おまじない
ちょっと長いけど、電子基板の
ピン番号を順番に並べ直すためのもの。
もし、DevKitCを使っている場合は
13,12,14,27,26,25,33,32,35,34に直そう。

「デモ2」光の流れ(発光ダイオード5本)

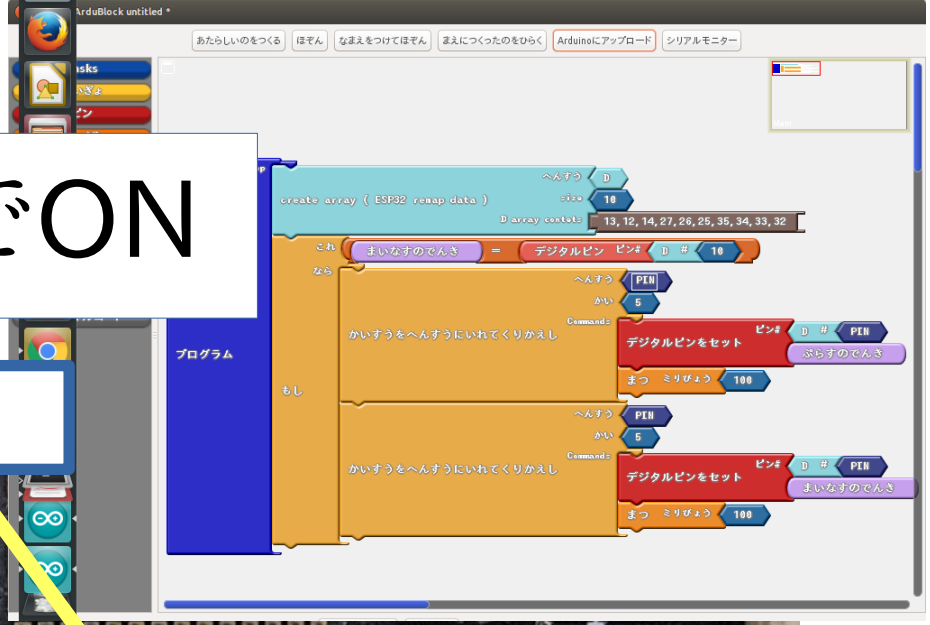
ソフトウェア2

問題に挑戦しよう。

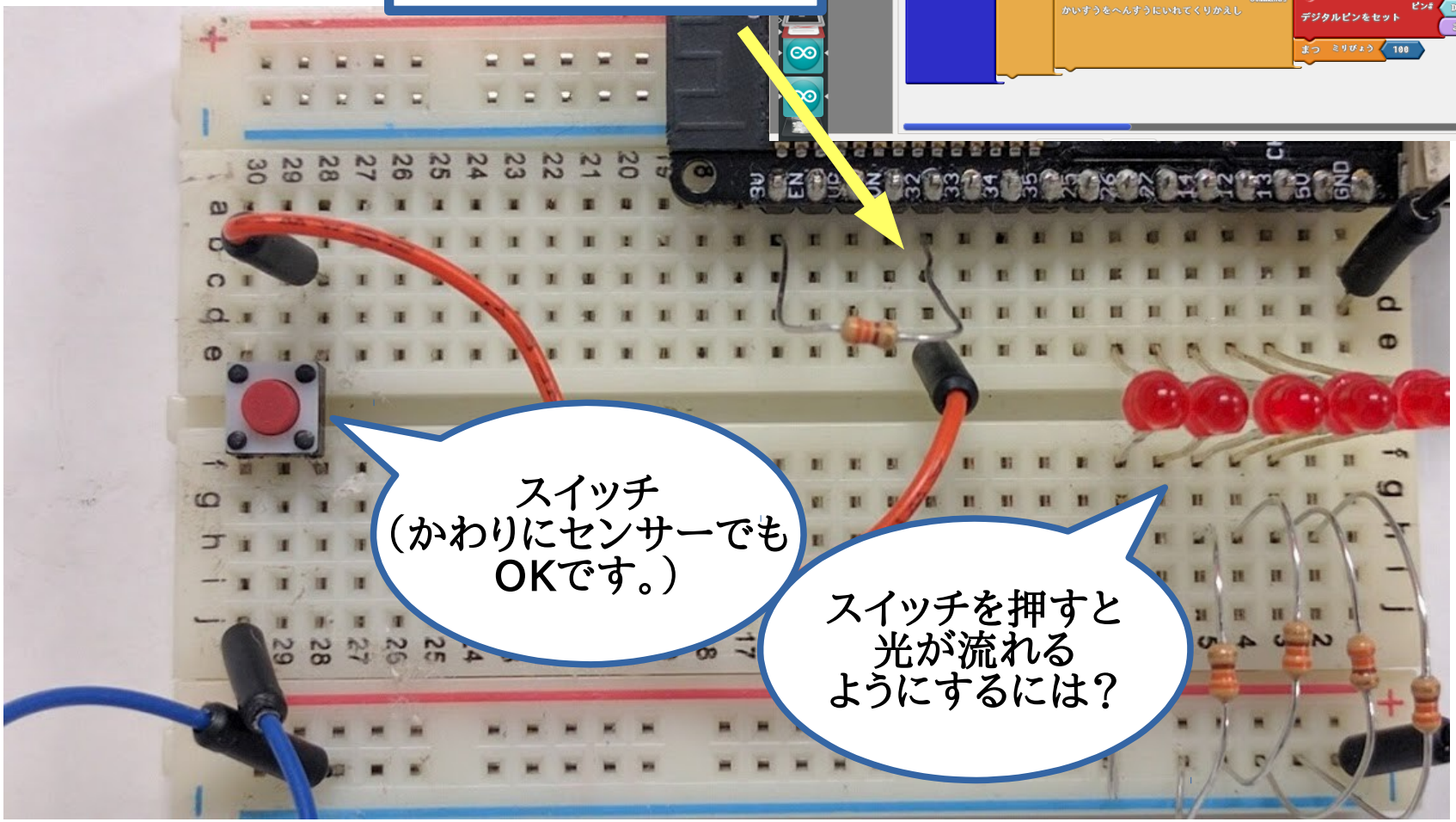
（1）光の流れを逆方向にする
（2）光の流れを折り返す
（3）発光ダイオードを1つ増やす
（4）その他

問題をとくアイデアは？

「おまけ」スイッチでON

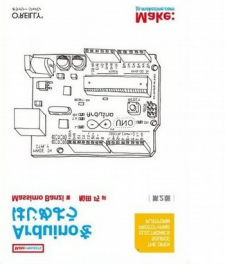


32(D#10)



もっと遊びたい

1.本

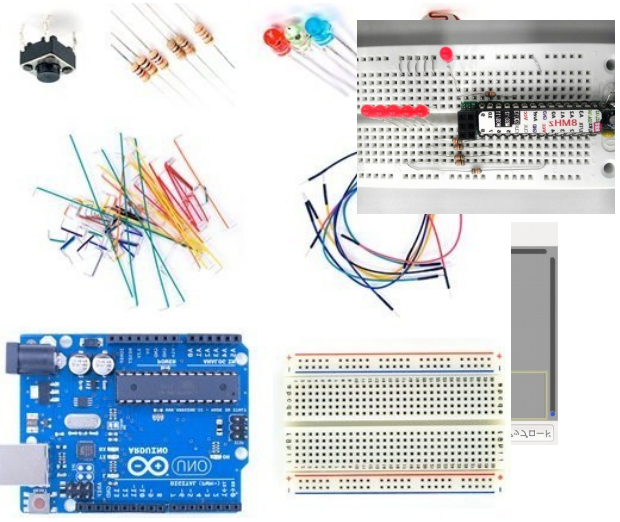


Arduino をはじめよう 第3版 (Make:PROJECTS)　￥2,160
入手先:アマゾン　http://www.amazon.co.jp/　など

2.ハードウェア　（つぎのいづれか）

(1)本テキストで使用した電子基板、部品　「AKBONE」　価格は問い合わせください
入手先:ロボット部　http://groups.google.com/group/robot-android-group-japan-akb

(2)「Arduinoで利口なガジェット」用実験キット　￥1500
入手先:アマゾン、秋葉原　若松通商など



3.ソフトウェア開発ソフト　（本テキストで使用したソフト）

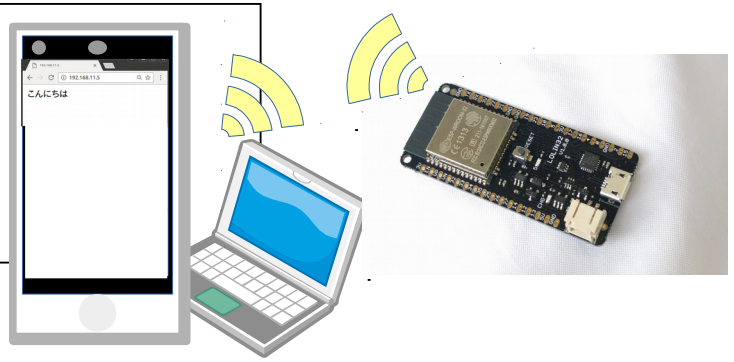
Arduino IDE　無料　開発環境
ArduBlock　無料　ビジュアルプログラミング環境

ダウンロードやインストール方法は、ロボット部の
「プロジェクトAKBONE」　などを参考にしてください。
<http://sourceforge.jp/projects/akbone>

ワークショップ　その2

「ネットワーク」と「インターネット」

IoT の実験



秘密結社オープンフォース

ESP32を使った小さな電子基板を使って、ネットワークや
インターネットと連携するプログラムを作ります。

協力：GEEKLAB.NAGANO

「IoTの実験」ワークショップ

概要

目的：　「ネットワーク」と「インターネット」につなげる

概要：　スマホやパソコンやクラウドと連携

対象：　小学生以上

コース時間：　60分～

コース方式：　個別指導

時間

(1)コース説明　5分

(2)WiFiにつなげる　10分

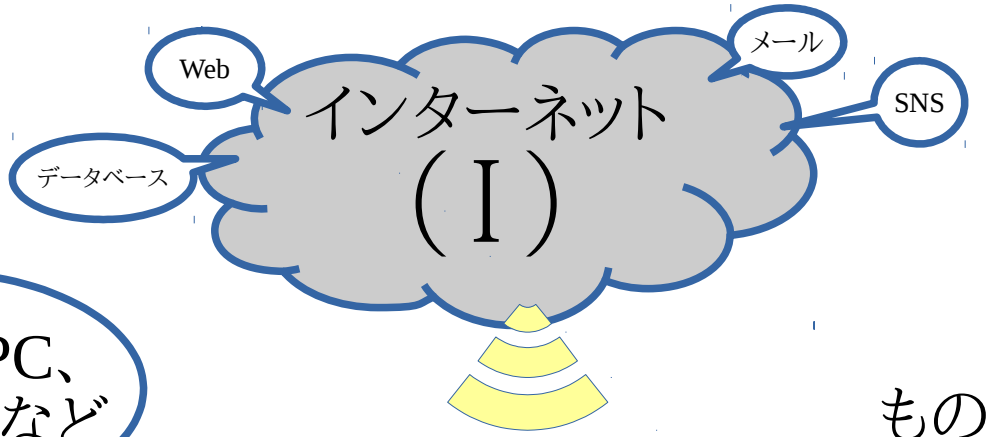
(3)トラブルシューティング(休憩)　10分

(4)事例デモ(Webサーバ、メール、クラウド連携)　10分

(5)Webサーバにする　10分

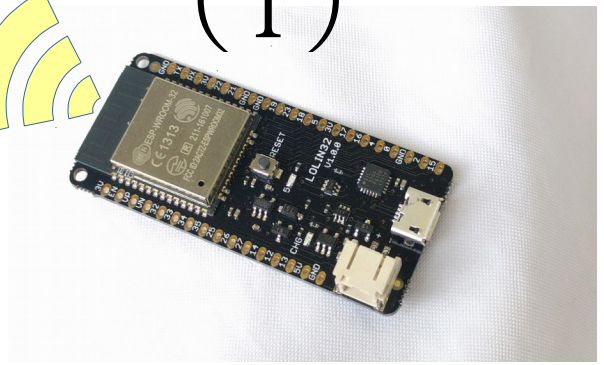
(6)自由課題(インターネットに繋げるなど)　15分

全体のつながり



スマホ、PC、
タブレットなど

もの
（T）



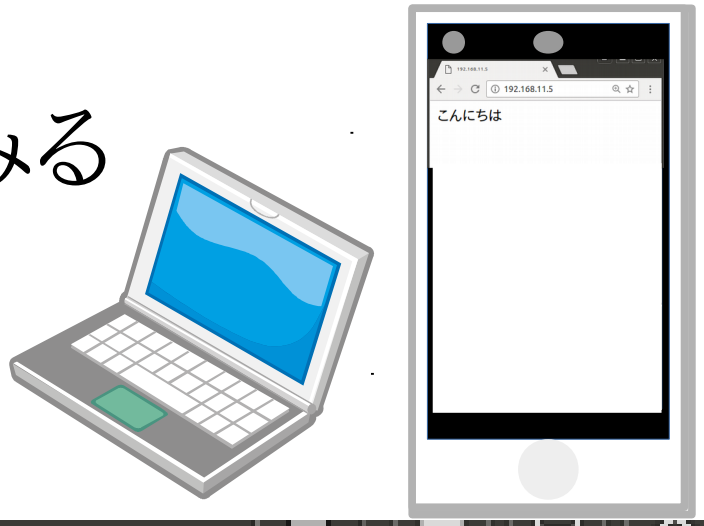
「動作チェック」　WiFiにつなぐ

ソフトウェア1

IPアドレスを調べる


ARDUBLOCK_WEBONOFF | Arduino 1.8.5
ファイル 編集 スケッチ ツール ヘルプ
自動整形
スケッチをアーカイブする
エンコーディングを修正
シリアルモニタ
シリアルプロッタ
ArduBlock
①
シリアルモニタを
起動しておきます
#include <WiFi.h>
WiFiServer server(80
char request[80];
char Get[80];
int charcount=0;
/dev/ttyUSB0
connected
IP address:
192.168.11.5
START
②
ここを9600にします
③
このようになったら
書込OKです
Writing at 0x00044000... (73 %)
Writing at 0x00048000... (78 %)
Writing at 0x0004c000... (84 %)
Writing at 0x00050000... (89 %)
Writing at 0x00054000... (94 %)
Writing at 0x00058000... (100 %)
Wrote 472480 bytes (305915 compr
Hash of data verified.
Compressed 3072 bytes to 122...
Writing at 0x00008000... (100 %)
Wrote 3072 bytes (122 compressed
Hash of data verified.
Leaving...
自動スクロール
改行なし
9600 bps
④
192.168.11.5
が
IPアドレスです

Webブラウザで見てみる



IPアドレスを入力します

192.168.11.5

表示できたらOK

こんにちは

文字化けするときは
下のように変更する

Contents HTML contents こんにちは

Contents HTML contents <meta charset="utf-8"/>こんにちは

トラブルシューティング！！

IPアドレスを決めたい時

PING

OKなとき

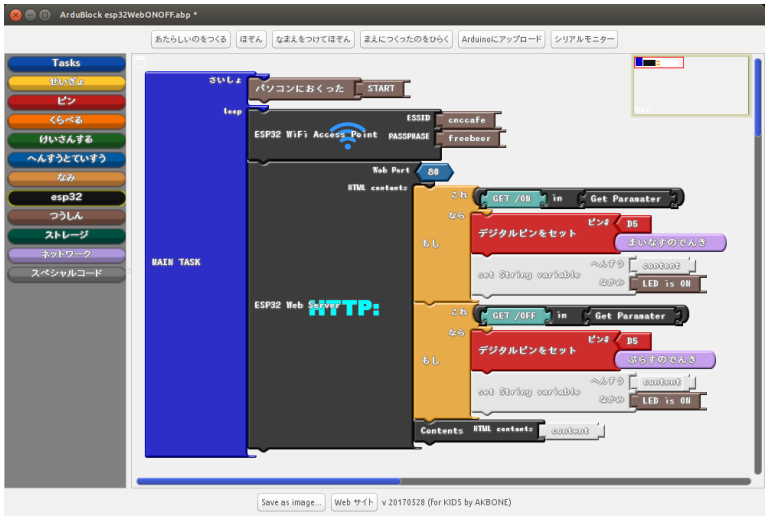
```
nanbuwks@LATITUDE: ~
nanbuwks@LATITUDE:~$ ping 192.168.11.5
PING 192.168.11.5 (192.168.11.5) 56(84) bytes of data.
64 bytes from 192.168.11.5: icmp_seq=1 ttl=255 time=1.54 ms
64 bytes from 192.168.11.5: icmp_seq=2 ttl=255 time=2.84 ms
64 bytes from 192.168.11.5: icmp_seq=3 ttl=255 time=1.82 ms
```

```
nanbuwks@LATITUDE: ~
nanbuwks@LATITUDE:~$ ping 192.168.11.5
PING 192.168.11.5 (192.168.11.5) 56(84) bytes of data.
From 192.168.11.8 icmp_seq=9 Destination Host Unreachable
From 192.168.11.8 icmp_seq=13 Destination Host Unreachable
From 192.168.11.8 icmp_seq=14 Destination Host Unreachable
```

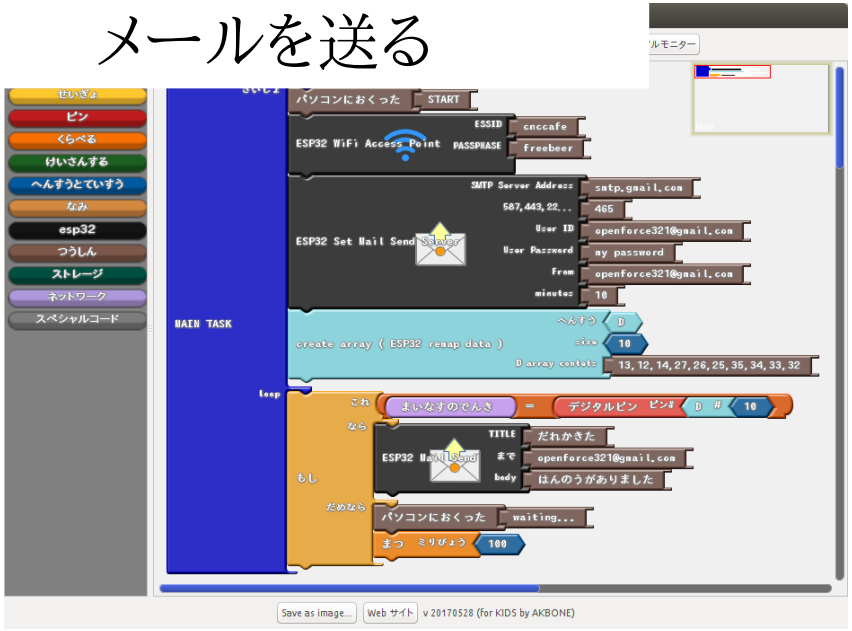
ダメなとき

「デモ」

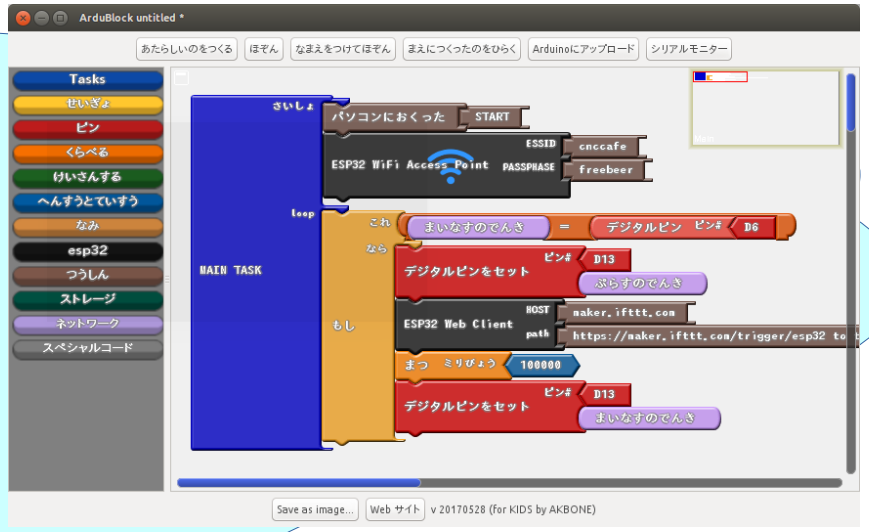
WebでON/OFF



メールを送る



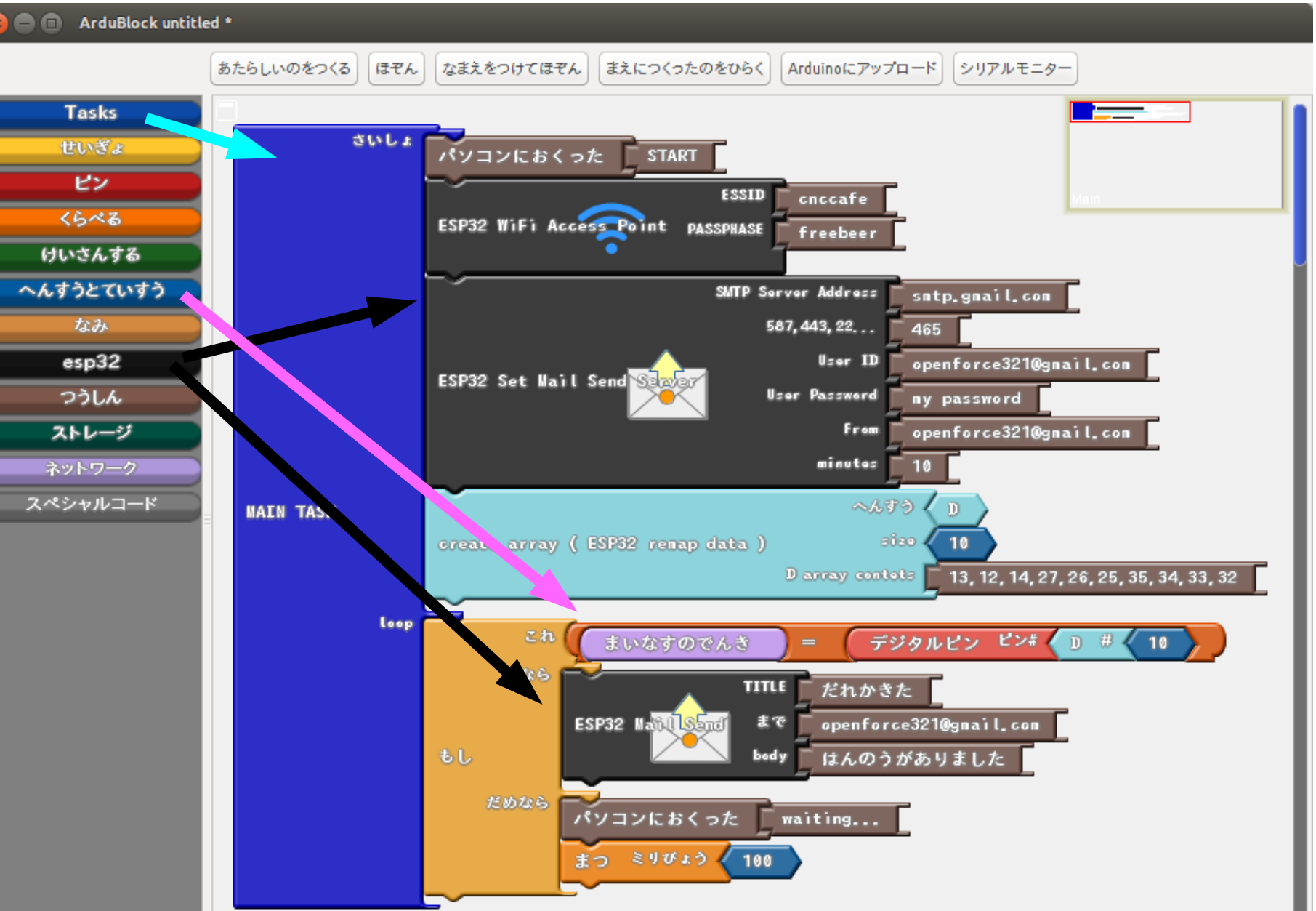
クラウドに接続して
TwitterやLineに送る



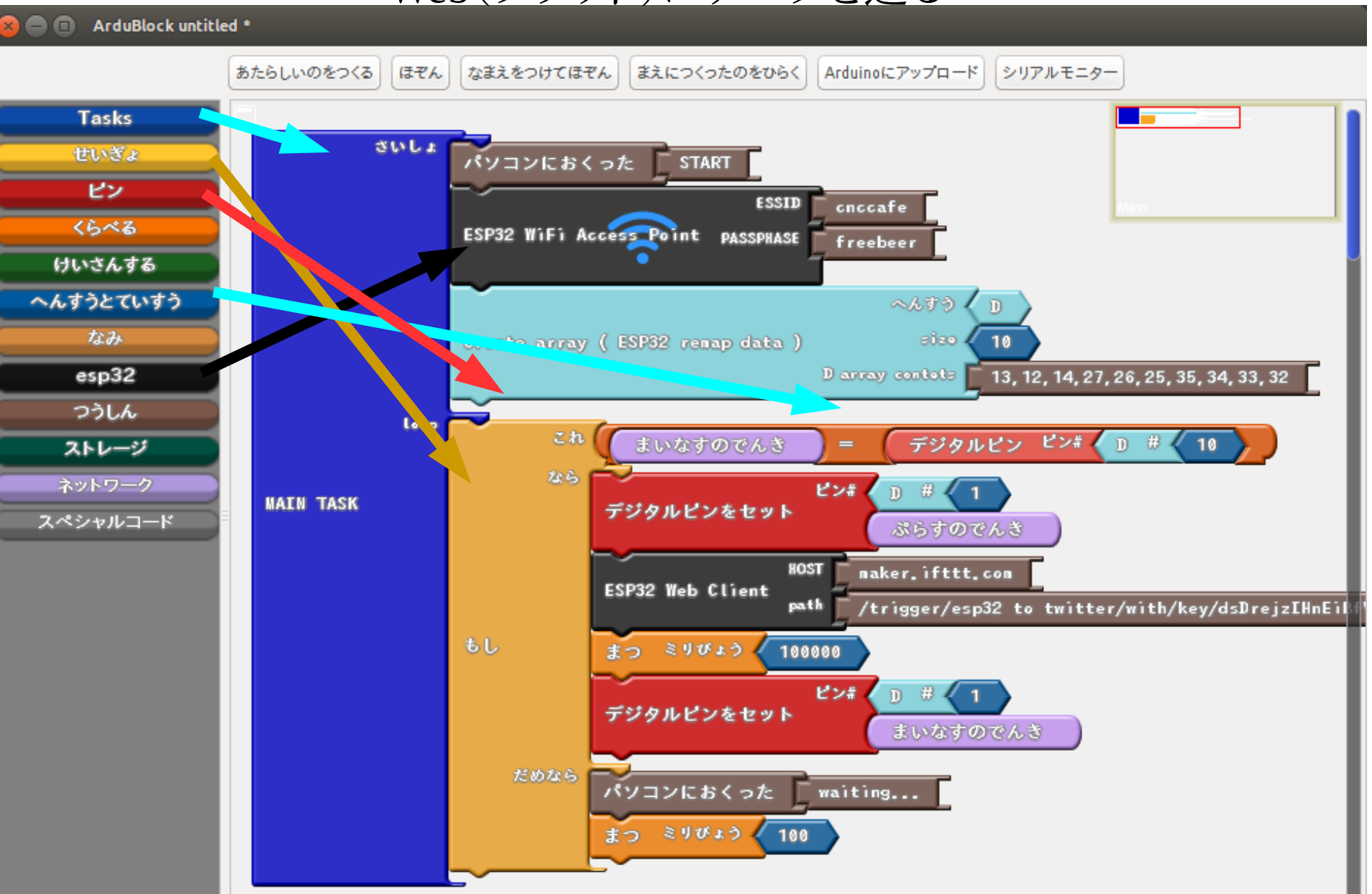
「Webサーバ」ブラウザでOn/Off

ソフトウェア1

だれかきた..はんのうがありました のメールを送る



Web(クラウド)にデータを送る



クラウドから送る

今回は、「IFTTT」という無料のサービスを使います。
ワークショップでは、あらかじめ設定したキーを使います。

IFTTT→Gmail

https://maker.ifttt.com/trigger/esp32_to_gmail/with/key/

IFTTT→twitter

https://maker.ifttt.com/trigger/esp32_to_twitter/with/key/

自分で設定するには、別紙「IFTTTの設定.pdf」を参照してください

考えよう!

（1）どうしてメールは10分に1つなの？

（2）パスワードはどうして隠すの？

通信は相手がある

相手に迷惑をかけないようにするには？

「本格的に実験するには」

1. IFTTTの設定をする方法
2.GMailのメールサーバを使ってメールを送る方法
3.インストールするには（設定済み環境をインストールする）
4.インストールするには（自分で設定する）
5.設定を整える
6.わからないことがあったとき

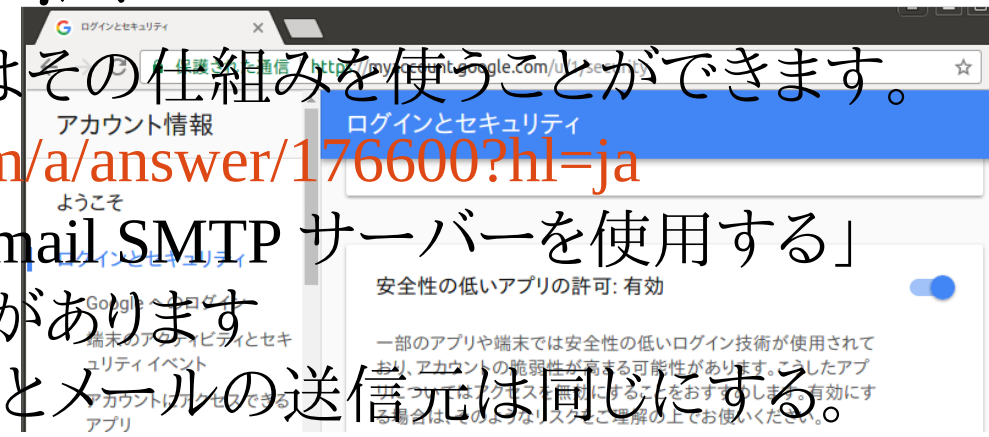
1.IFTTTの設定をする方法

ワークショップでは設定済みIFTTTの仕組みを使います。
自分で設定をするともっと詳しく設定できます。

やってみたい人は別紙資料「IFTTTの設定.pdf」を参照して下さい。

2.Gmailのメールサーバを使ってメールを送る方法

- Gmailを使っている場合はその仕組みを使うことができます。
- https://support.google.com/a/answer/176600?hl=ja
  この資料の「制限付き Gmail SMTP サーバーを使用する」設定を済ませておく必要があります
- メールサーバのユーザIDとメールの送信元は同じにする。
- 自分のPCの置き場所と別の場所からメールを送ろうとしたときは、自分のPCに警告が出るので解除して下さい。



3.インストールするには
（設定済み環境をインストールする）

プロジェクトAKBONEから一式をダウンロードできます
https://ja.osdn.net/projects/akbone/releases/

ダウンロード→解凍→自分の好きなところに置いて、
インストールして下さい
・Linux64環境は install.shを実行して下さい。
・Windows環境／Linux64環境は portable 環境です。
・Mac OSX環境は通常バージョンです

4.インストールするには（自分で設定する）

1.Arduino開発環境をダウンロード、インストールする
http://www.arduino.cc

2.ESP32環境をダウンロード、インストールする
https://github.com/espressif/arduino-esp32
から ZIP ファイルをダウンロード、
展開してできた arduino-esp3-master フォルダの名前を esp32に変更、
hardware フォルダ(*)内に espressifフォルダを作りその中に移動（hardware/espressif/esp32となる）
その中のtoolsフォルダの中のインストーラを実行(Windowsはget.ext Mac/Linuxは python get.py)

3.Ardublock環境をダウンロード、インストールする
https://ja.osdn.net/projects/akbone/downloads
からardublock-all.jarをダウンロード、tools フォルダ(*)の ArduBlockTool/tool フォルダを作って配置

4.Mailライブラリをダウンロード、インストールする
http://kerikeri.top/posts/2017-04-08-esp32-mail/
からmail.zipをダウンロードして展開、libraries フォルダ(*)のMailerフォルダを作りそこにmailer.h配置

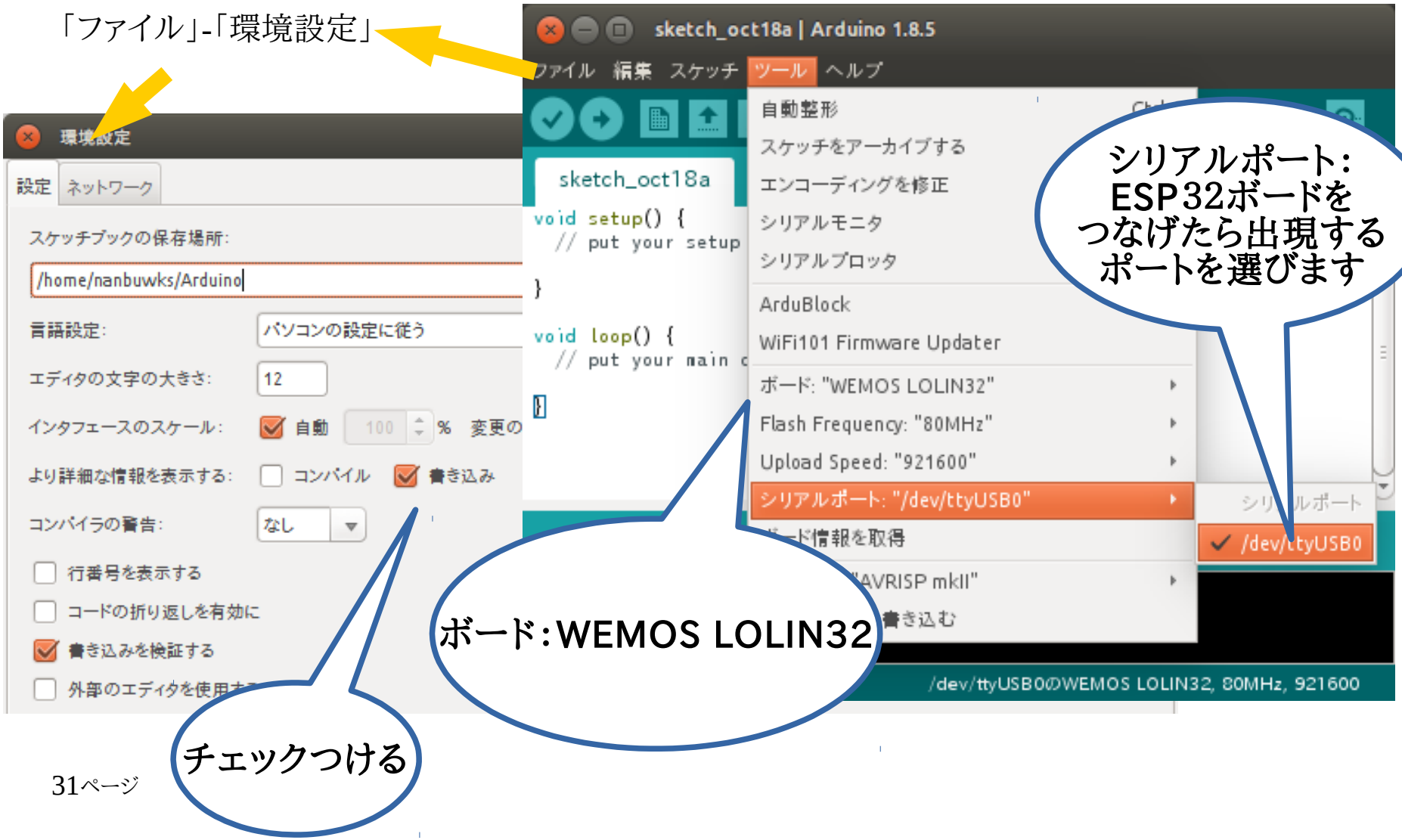
5.BLEライブラリをダウンロード、インストールする（エラーが出るとき）
https://github.com/espressif/arduino-esp32/tree/master/libraries の BLE @ cb1ab4e (名前が変わる場合があります)
のリンク先からZIPファイルをダウンロード、
展開してできたexamplesやsrcなどを 2.の hardware/espressif/esp32/libraries/BLE内にコピー

6.USBシリアルドライバをインストールする
https://jp.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers
「CP210x USB - UART ブリッジ VCP ドライバ」（ Linux の多くはインストールの必要はありません）

hardwareフォルダ、toolsフォルダ、librariesフォルダの場所について
・通常の場合はドキュメントフォルダのArduinoの中です。
・portable環境の場合はportableフォルダを置いているところと同じ所にあります。

5.設定を整える

「ファイル」-「環境設定」



6.もっとよく知りたい時

新版のソフトウェアや基板は https://ja.osdn.net/projects/akbone/ にて公開しています。

勉強会やコミュニティに参加してみよう!

日本Androidの会秋葉原支部ロボット部

月例で勉強会を行なっています(東京都内)。
興味のある方はML(メーリングリスト)にご登録ください。
http://groups.google.com/group/robot-android-group-japan-akb
MLでは勉強会のアナウンスや部員が興味をもった様々な話題がなされています。

秘密結社オープンフォース

ArduBlockのESP32バージョンの開発を行っています。
開発のトピックは https://qiita.com/nanbuwks などで公開しています。

GEEKLAB.NAGANO

長野市で、ITやテクノロジー関係に関する勉強会、イベント、セミナーを開催しています!
平日 9:00～18:00 は無料開放しておりますので、作業場としてなどご利用いただけます。
予約等は不要ですので、お気軽にお越しください!(土日祝日は問い合わせ要)

住所:長野県長野市南県町1003 県都ビル6F