

実践IoTハウス

Home automation & Security

<http://www.pepolinux.com>

Twitter@kujiranodanna

山内

2017/8/5 in OSC2017Kyoto

IOTハウスやっていますか？

- ◆ IOTハウスとは・・・インターネット経由でなんか家の中の物を動かす
- ◆ インターネットに繋がってたら家の物が勝手に動く？少し気持ち悪いかも
- ◆ スマートハウスとは・・・スマートフォンでなんか家の中の物を動かす？
- ◆ 個人的な解釈ですが、さて皆さんは？

IOTハウスやっています

- ◆ 出先から部屋の温度を確認してクーラー運転する
- ◆ 夕方に外が暗くなると玄関や部屋の明かりを点ける
- ◆ 外の湿度を元に家庭菜園の水やりをする
- ◆ 防犯センサーがはたらくと写真をメールする
- ◆ 火災感知器でメールと電話（ワンギリ）で知らせる
- ◆ 8~19時に2階の温度が27度を超で屋根裏の換気扇を回す
- ◆ 迷惑な鳥を鷹の鳴き声で追い払う
- ◆ などRaspberry PiでIoTハウスを実践しています。
- ◆ 引用：<https://ja.osdn.net/projects/pepolinux/>

実践IoTハウス

Home automation & Security



実践IoTハウス

Raspberry PiでIoTハウスを実践しています。



三都物語＋北海道＋東京＋名古屋

kof2002→osc2011kobe→osc2011kyoto→kof2011→kof2012→osc2013Kyoto

→kof2013→osc2014hokkaido→kof2014→osc2015Tokyo→osc2015Nagoya→osc2015kyoto→kof2015→osc2016nago

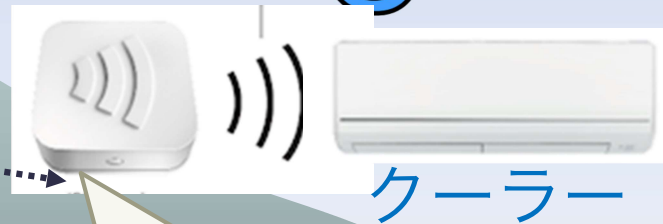
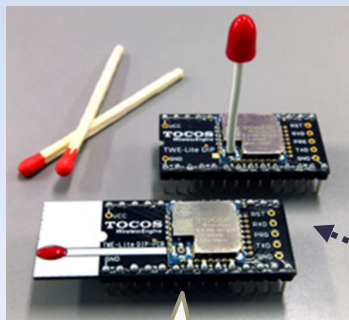
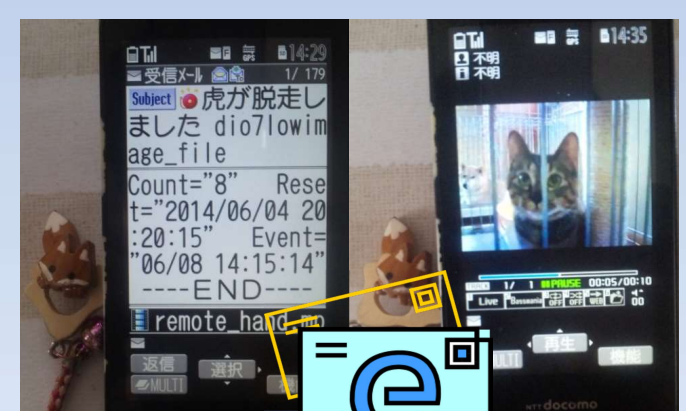
ya→kof2016→osc2017Osaka→osc2017Hamanako

→osc2017Nagoya→osc2017Kyoto



実践IoTハウス

Home automation & Security



クーラー

Tocos無線モジュール
離れた物を監視制御

火災報知器発報で動画メール
と電話同時通知

Raspberry Pi + Piface
Gmail連携で動物監視
外部機器の監視制御

IRKit
学習型赤外線リモコン



リモートハンド

古いPC (LinuxBeans) でIoTハウス

開発中



実践IOTハウス

Remote-Hand Raspberry_pi_xx.zipで公開中

- ◆ 最新Ver0.37は2017.7.29リリース
- ◆ リモートDIO・音声ファイル再生機能など追加
- ◆ Remote-Hand Raspberry_piは全てオープンソース下記からダウンロード
- ◆ <https://osdn.jp/projects/pepolinux/releases/p14620>

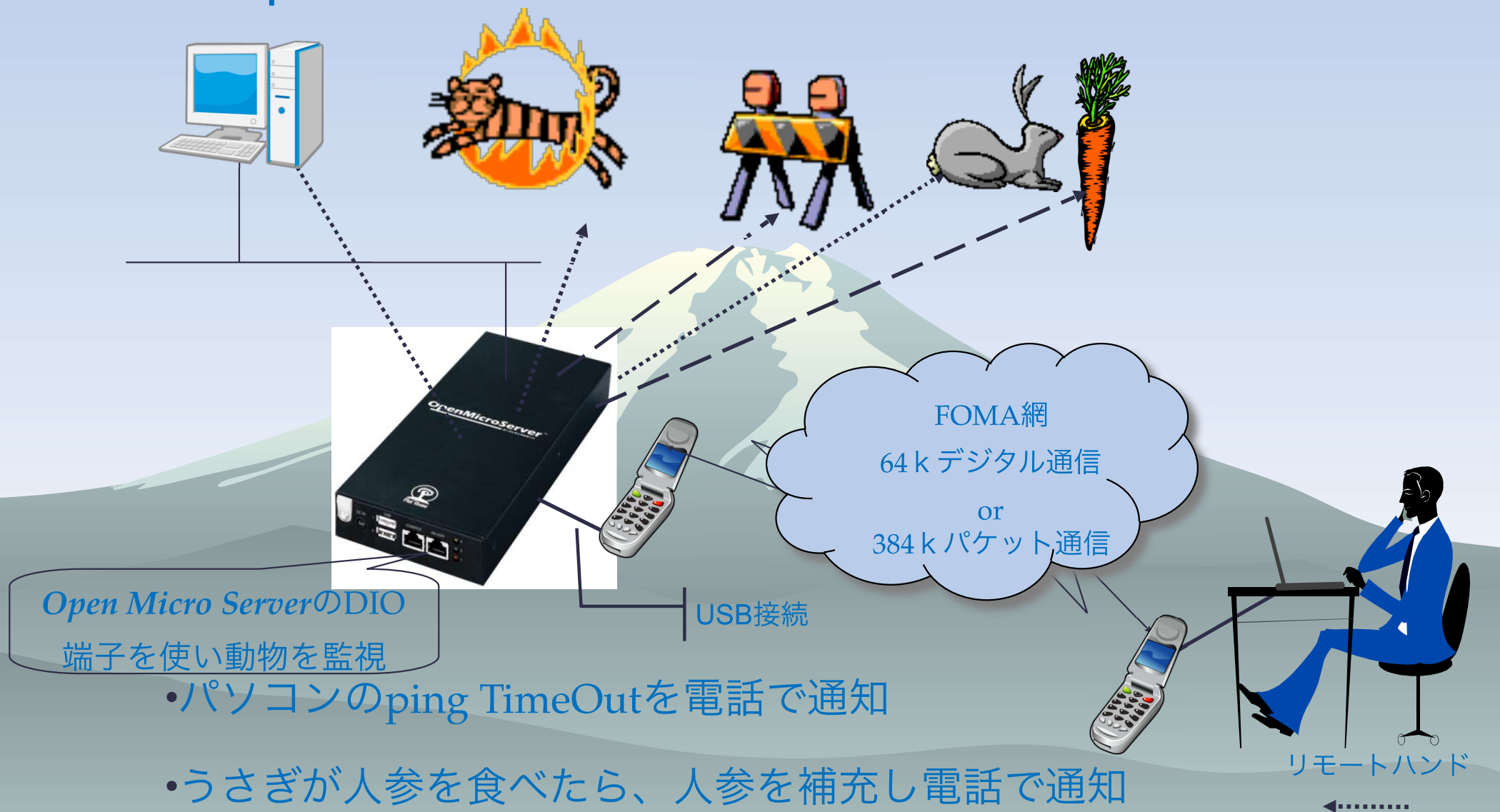
プログラミング楽しんでますか

Remote-Hand Raspberry_piの開発環境

- ◆ 元Windows2000にLinuxBeans、 wine上で
TeraTerm、 Notepad++、 vim、 最後に
Windows10
- ◆ WebUI、 javascript、 jQueryとCGI、 ヘルパーAPは全
てシェルスクリプト(Bash)、 一部C言語
- ◆ デバックは実践IOTハウスで日々
- ◆ 還暦を超えてもプログラミングと電子工作を楽しんで
ます

PepoLinux

OpenMicroServer + FOMA Remote-Hand



- パソコンのping TimeOutを電話で通知
- うさぎが人参を食べたら、人参を補充し電話で通知
- トラが檻を破って脱走したら、パトライトを点灯
- うさぎやトラをどうやって認識させる

PepoLinux

Vmplayer + PepoLinux + DIO + FOMA = Remote-Hand



Vmplayer + PepoLinux +
シリアルDIOボードで動物
を監視

シリアル接続

USB接続

FOMA網

64 k デジタル通信

or

384 k パケット通信

•トラが檻を破って脱走したらパトライトを点灯させ電話通知

•うさぎが人参を食べたら、人参を補充しMailで通知

•Serverのping TimeOutをMailと電話で同時通知

•うさぎやトラをどうやって認識させる？

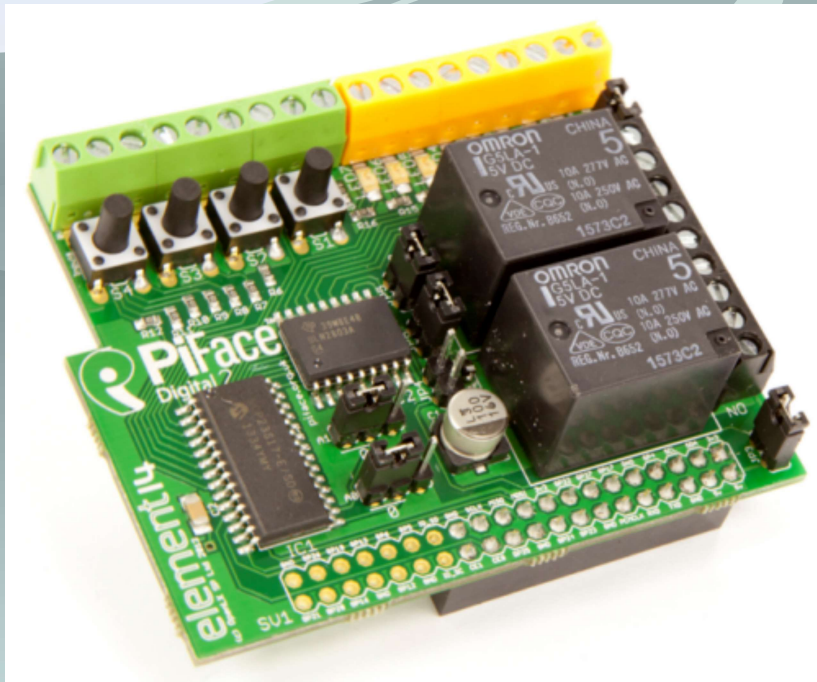
ところでRaspberry Pi

◆ raspberry piとは . . .



Raspberry Piの拡張ボードPiface

- ◆ Pifaceとはデジタル I / O 拡張ボード
Raspberry Pi用の入出力 8 ビットの拡張ボード
- ◆ 5 V リレー × 2 の出力で AC100V / DC24V 5 A 迄
外部スイッチング



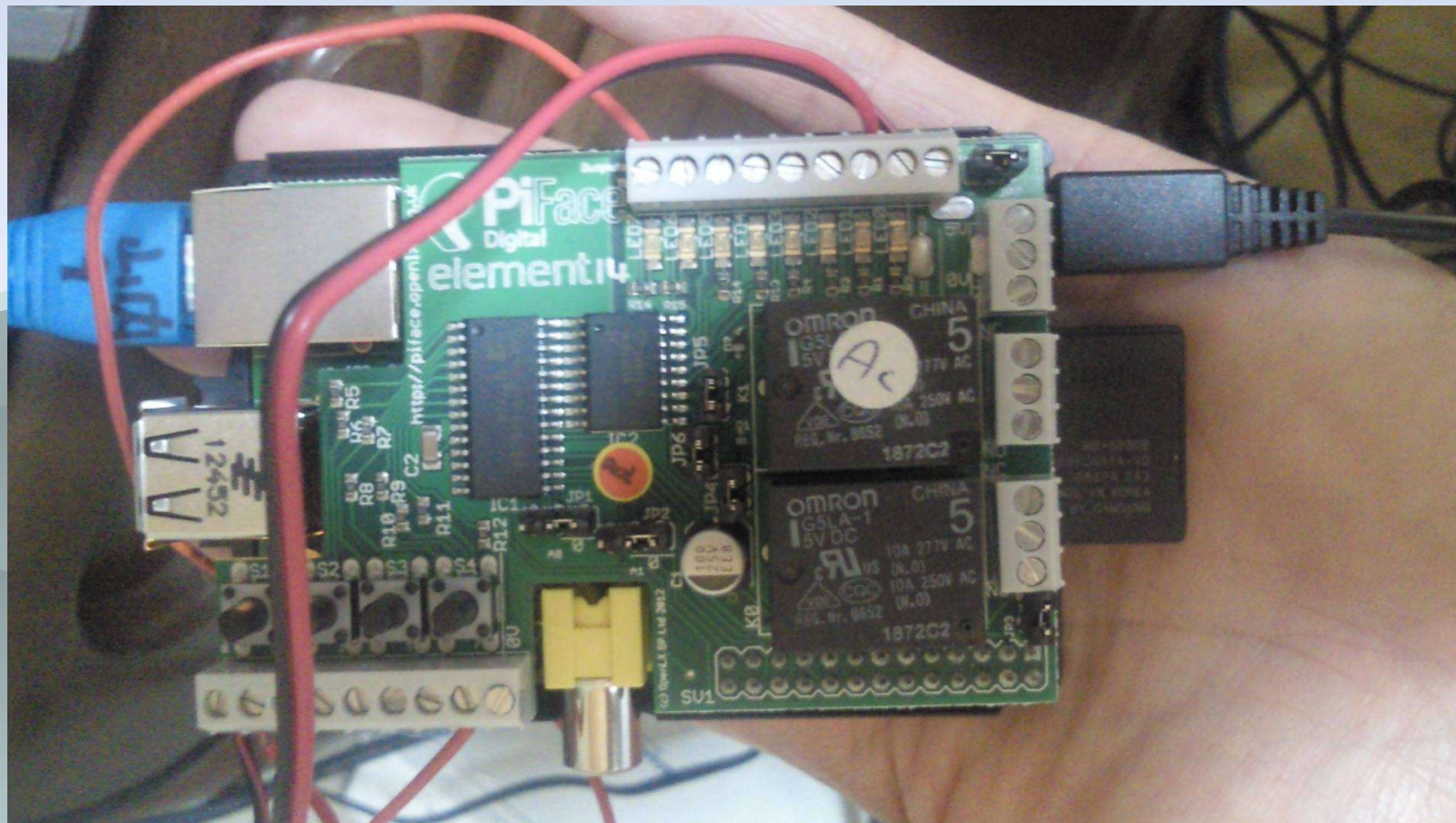
実践IOTハウス

Home automation & Security

コアプログラム

- pepopiface : DIOのR/Wをマルチユーザーで動作
- pepopifaced : DIのHI/LOWに合わせメール、DO、カメラほか制御
- pepogmail4dio : gmailでDIO操作ほか情報を返信
- pepogmail4pic : gmail経由でModulesカメラ動画を数秒添付
- pepotocosctl : 2.4GHz無線モジュール TWE-Lite DIPで離れた物を監視制御する
- pepoirkitpost : 学習型赤外線リモコンIRkitの操作

pepopifaceはpiface(Gpio)を制御



pepopifaceはpiface(Gpio)を制御

** Welcome to pepopiface Version-0.6 Copyright
Yamauchi.Isamu compiled:Sep 20 2015 **

usage:pepopiface port:0-8 [0|1] [timer:0-300000ms]

pepopiface 0 1 → 1 port0へwrite

pepopiface 0 1 1500 → port0へ1.5秒 1 write

pepopiface 0 → 1

pepopiface 8 → 00ff 上位8 : 出力・下位8 : 入力

pepopifaceはlibpiface使用pfio.cにpatch

<https://docs.google.com/folder/d/0B-UAZ9CyJCLGQjJ3RDlqa2pqaDg/edit?pli=1>

pepopifaceコマンドはpiface(Gpio)を制御する
libpifaceに難あり

pfio_init()を最初にcallが必要、しかし毎回出力ポートが初期化

patchを作成 **../piface/c/src/piface/pfio.c**

```
// spi_write(GPIOA, 0x00); // turn on port A
spi_write(IODIRA, 0); // set port A as an output
spi_write(IODIRB, 0xFF); // set port B as an input
spi_write(GPPUB, 0xFF); // turn on port B pullups

// initialise all outputs to 0
// int i;
// for (i = 1; i <= 8; i++)
//     pfio_digital_write(i, 0);
//
```

pifaceを使い込むと不具合が

- ◆ マルチプロセス環境ではpifaceで競合
- ◆ 点けたはずの灯りが点かない
- ◆ 消したはずの灯りが消えてない
- ◆ 排他制御が必要
- ◆ 排他制御セマフォ

排他制御セマフォ

- ◆ semaphoreセマフォはプロセス間でリソースの排他制御
- ◆ Raspbianとpepopifaceで実装
- ◆ セマフォはflagの加算 / 減算処理
- ◆ flagがマイナスでロック
- ◆ flagがプラスでアンロック
- ◆ 詳しくはhttps://ja.osdn.net/projects/pepolinux/wiki/chobit_prog%20semaphore

pepopifacedはdaemon

- ◆ 定周期でpepopiface 8を投げ入力 8ポートを監視
- ◆ 例入力ポート 0 の変化によりコマンドを起動

High→Low : dio0lowコマンドを実行

Low→High : dio0highコマンドを実行

Management DI(Digital Input)-1

Setting first action to the digital input

火事です	high	Action:low→high	パトライトlow	Alt	none	ms	none	Log display
Phone		Email	パトライトhigh		Count:	8	2017/07/18 14:44:48 ~ 2017/07/18 14:50:47	
			パトライトlow					
防犯センサー	high	Action:low→high	Output8high	Alt	none	ms	none	
Phone		Email	Output8low					
			Output9					
Input3	high	Action:low→high	Output10	Alt	none	ms	none	
Phone		Email	Output11					
			Output12					
			Output13					
Input4	high	Action:low→high	Output14	Alt	none	ms	none	
Phone		Email	玄関電灯high					
			玄関電灯low					
Input5	high	Action:low→high	Output16high	Alt	none	ms	none	
Phone		Email	Output16low					
			Output17high					
			Output17low					
Input6	high	Action:low→high	Phone	Alt	none	ms	none	
Phone		Email	Email					
			Email_messageage					
Input7	high	Action:low→high	Web_camera Video	Alt	none	ms	none	
Phone		Email	Mod_camera Still					
			Mod_camera Video					
			Sound_1					

防災システム

大阪ガス・ピコピコ火災報知器（無電圧A接点）



火災報知器が発報するとメール
と電話で通知

防犯システム

人感センサーが働くと動画
メールと電話で通知



人感センサー
無線受信器



人感センサー
無線送信器



センサーライト
+ 無線送信器

動物監視（ペット）



Gmail経由カメラで動画
子供達の顔を確認

ソリッドステートスイッチ AC100Vコンセント

2Aのリセッタブル
ヒューズ内蔵



DIO出力0-5VでAC100V/5Aをオフ/オンさせるソリッド
ステートスイッチ内蔵コンセントBOX 2,000円程度

実践IoTハウス

Home automation & Security

Remote-Hand raspberry_pi ver:0.24.2016.11.8

Setting automated process

URL:

Start: * 00-59 * 00-23 * 1-31 * 1-12 * *:Every time

Folder name: Podcasts Entry

Se	URL	Condition	Action	Time	Repeat
1	CPU_Temphigh	CPUファンhigh	1800(ms)	*	n
2	屋外照度low	窓の電気high	3000(ms)	*	n
3	屋外照度low	窓の電気low	3000(ms)	*	n
4	屋外照度high	窓の電気low	ms	*	n
5	屋外照度low	ななちゃん電灯low	3000(ms)	*	n
6	屋外照度high	ななちゃん電灯low	ms	*	n
7	屋外照度low	玄関電灯low	3000(ms)	*	n
8	屋外照度low	玄関電灯low	3000(ms)	*	n
9	屋外照度high	玄関電灯low	ms	*	n
10	Enable	ななろく天井電灯	ms	30	n
11	Enable	ななろく天井電灯	3000(ms)	02	n
12	USB_Temphigh	扇風機high	9000(ms)	*	n
13	I2C_Humlow	散水ポンプhigh	6000(ms)	10	min
14	Enable	none	ms	*	min

おまけ
podcastを定時取得
デバイスに保存

定時処理＋条件
定時に電灯や散水ポンプを
オン/オフ
照度・温度・湿度の条件で
オン/オフ
定時に音楽ファイルを再生

家庭菜園の自動散水は湿度の条件 オン・オフ



Setting TWE-LITE Wireless DI & AI

TI1	low	散水スイッチ	none	▼
TI2	high	Input10	none	▼
TI3	high	Input11	none	▼
AI1	high	1568 屋外照度	Slice 1300	none ▼ Analog input-1 Graph
AI2	high	4080 Input14	Slice 3000	none ▼ Analog input-2 Graph
AI3	low	4080 Input15	Slice 4500	none ▼ Analog input-3 Graph
AI4	high	4080 Input16	Slice 2000	none ▼ Analog input-4 Graph

CPU_Temperature	low	45.4°C	Slice 45	none ▼	CPU Temperature Graph
USB_Temperature	low	20.8°C	Slice 25	none ▼	USB Temperature Graph
I2C_Temperature	low	15.5°C	Slice 30	none ▼	I2C Temperature Graph
I2C_Humidity	high	92.1%	Slice 90	none ▼	I2C Humidity Graph

2	屋外照度low ▼	窓の電気high ▼	3000(ms	*	min	5 ▼	18-2hour *	
3	屋外照度low ▼	窓の電気low ▼	3000(ms	*	min	5 ▼	00-0hour *	
4	屋外照度high ▼	窓の電気low ▼		ms	*	min	5 ▼	05-0hour *
5	屋外照度low ▼	ななちゃん電灯low ▼	3000(ms	*	min	5 ▼	00-0hour *	
6	屋外照度high ▼	ななちゃん電灯low ▼		ms	*	min	5 ▼	05-0hour *
7	屋外照度low ▼	玄関電灯low ▼	3000(ms	*	min	5 ▼	17-2hour *	
8	屋外照度low ▼	玄関電灯low ▼	3000(ms	*	min	5 ▼	00-0hour *	
9	屋外照度high ▼	玄関電灯low ▼		ms	*	min	5 ▼	05-0hour *
10	Enable ▼	ななろく天井電灯 ▼		ms	30	min	* ▼	17 hour
11	Enable ▼	ななろく天井電灯 ▼	3000(ms	02	min	* ▼	00 hour	
12	USB_Temphigh ▼	扇風機high ▼	9000(ms	*	min	2 ▼	* hour	
13	I2C_Humlow ▼	散水ポンプhigh ▼	6000(ms	10	min	* ▼	20 hour	
14	Enable ▼	ななろく天井電灯 ▼	3000(ms	02	min	* ▼	00 hour	

玄関電灯は簡易照度計（100均電卓）の条件 でオン・オフ

17～24時屋外照度Lowで点灯
5～7時屋外照度High消灯



Setting TWE-LITE Wireless DI & AI

TI1	low	散水スイッチ	none	▼
TI2	high	Input10	none	▼
TI3	high	Input11	none	▼
AI1	high	1568 屋外照度	Slice 1300	none ▼ Analog input-1 Graph
AI2	high	4080 Input14	Slice 3000	none ▼ Analog input-2 Graph
AI3	low	4080 Input15	Slice 4500	none ▼ Analog input-3 Graph
AI4	high	4080 Input16	Slice 2000	none ▼ Analog input-4 Graph

CPU_Temperature	low	45.4°C	Slice 45	none ▼	CPU Temperature Graph
USB_Temperature	low	20.8°C	Slice 25	none ▼	USB Temperature Graph
I2C_Temperature	low	15.5°C	Slice 30	none ▼	I2C Temperature Graph
I2C_Humidity	high	92.1%	Slice 90	none ▼	I2C Humidity Graph



実践IOTハウス

害鳥を検出したら鷹の声で追い払う

Management DI(Digital Input)-1

Setting first action to the digital input

火事です	high	Action:low→high	パトライトlow	Alt	none	ms	none	Log display
Phone		Email		Message		Count:	8	2017/07/18 14:44:48 ~ 2017/07/18 14:50:47
防犯センサー	high	Action:low→high	none	Alt	none	ms	none	
Phone		Email		Message				
ドップラーセン	high	Action:low→high	Sound_3	Alt	none	ms	none	Log display
Phone		Email		Message		Count:	2	2017/07/18 16:16:39 ~ 2017/07/18 16:39:47

ドップラーセンサーで検出

Setting Sound

Sound Upload

Sound1	ファイルを選択	選択されていません	Upload	police_car.mp3	none
Sound2	ファイルを選択	選択されていません	Upload	パトカーサイレン.mp3	none
Sound3	ファイルを選択	選択されていません	Upload	鷹の雄叫び.mp3	none
Sound4	ファイルを選択	選択されていません	Upload	none	none
Sound5	ファイルを選択	選択されていません	Upload	none	none

Gmail経由で制御 ユーザーとキーワード登録

Setting system Email

Setting operation in Gmail

Gmail User:

Gmail Password:

Mail Address:

Key Word:

Subject:

Mail Check Timer: New Email check

Jitter: EmailArrival correction(Sec)

Entry ▼

Run

Clear

Gmail経由でコマンド
例pifaceの1番出力を
5秒間オンにする
Subject: dio2 1 1 5000

実践IOTハウス

Gmail経由監視・とらが脱走したらメール

Management DI(Digital Input)-2

Setting second action to the digital input

火事です : high Action: low→high NONE Timer: ms none

Phone: Email: 回数: 0 2015/05/24 18:00:09 ~

防犯センサー : high Action: low→high NONE Timer: ms none

Phone: Email: ななちゃん電気low 1673 2015/05/24 18:00:10 ~ 2015/07/27 14:07:07

出力1と接続 : high Action: low→high NONE Timer: ms none

Phone: Email: 16 2015/07/03 05:02:16 ~ 2015/07/25 12:32:49

Input4 クーラー運転send Timer: -

Phone: クーラー停止send Timer: -

Input5 スイングsend Timer: -

Phone: 首を振ってsend Timer: -

Input6 戦闘機send Timer: -

Phone: 2階天井換気扇high Timer: -

Input7 2階天井換気扇low Timer: -

Phone: Output16high Timer: -

Input8 Output16low Timer: -

Phone: Output17high Timer: -

2階が火事で Output17low Timer: -

Phone: Phone Email Image Email

Input10 NONE Count: -



Gmail経由で制御

キーワードでコマンドリスト送付

==input list==

1:Input1=high
2:Input2=high
3:Input3=high
4:Input4=high
5:Input5=high
6:Input6=high
7:Input7=high
8:防犯センサー=high
9:ドップラーセンサー=high
10:Input10=high
11:Input11=high
13:簡易照度計=high
14:Input14=high
15:Input15=low
16:Input16=low
17:CPU_Temp=low
18:USB_Temp=none
19:I2C_Temp=high
20:I2C_Hum=low

==output list==

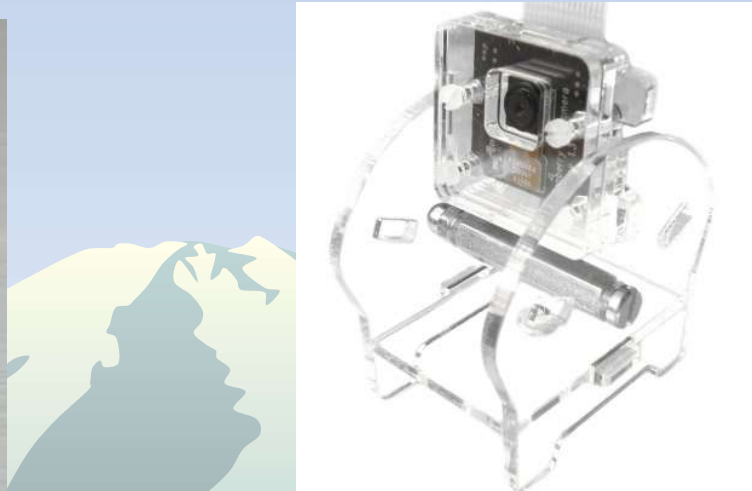
1:CPUファン=high
2:Outnput2=low
3:Outnput3=low
4:Outnput4=low
5:Outnput5=low
6:Outnput6=low
7:Outnput7=low
8:Outnput8=low
9:Outnput9
10:Outnput10
11:Outnput11
12:Outnput12
13:Outnput13
14:Outnput14
15:髭剃り充電=low
16:天井換気扇=low
17:Outnput17=none

==Other Command==

21:CPU Temperature
22:USB Temperature
23:No1 Camera Movie
24:No1 Camera Picture
25:No2 Camera Movie
26:No2 Camera Picture
27:No3 Camera Movie
28:No3 Camera Picture
29:Module Camera Movie
30:Module Camera Picture
31:I2C Temperature&Humidity
32:CPU Temperature&Graph
33:USB Temperature&Graph
34:I2C Temperature&Graph
35:I2C Humidity&Graph

モジュールカメラで写真メール添付

`raspistill -t 1000 -vf -hf -w 2592 -h 1944 -o ${IMAGE}`



高画質な写真や動画が撮れます

IRKit

IRKitとはネットワーク対応学習型赤外線リモコン



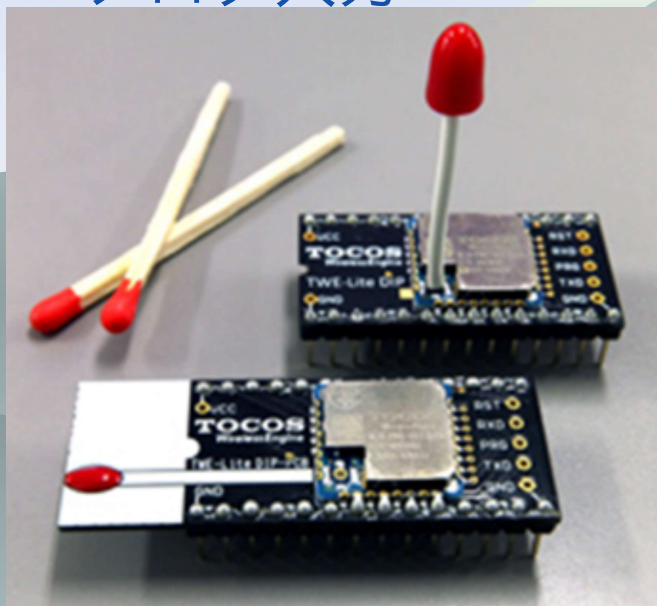
IRKitとコラボ

Raspberry piでIRデータ登録・送信
ペット部屋の天井電灯を定時オン/オフ
外出先から室温確認してクーラーオン/オフ

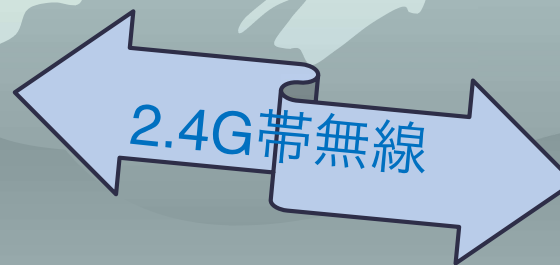


離れたものを監視制御する Tocos無線モジュール

- Tocosからmono-wirelessに変わったけど
- 簡単にすぐに使える無線モジュール
- TWE-Lite DIP（トワイライト・ディップ）は AI・PWM・I2C・DIO
- DIO（デジタルイン・アウト）I2C温湿度センサーAM2321やアナログ入力



TWE-Lite DIP

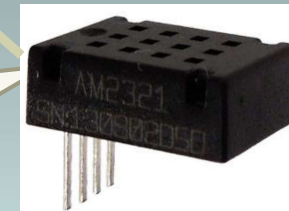
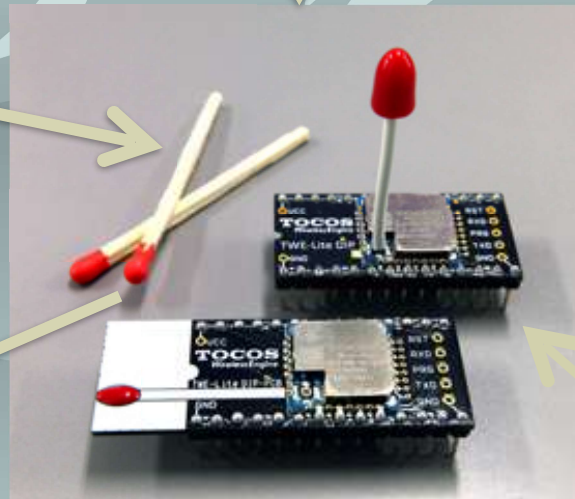


USB Dongle
ToCoStick

離れたものを監視制御する Tocos無線モジュール

wirelessでDIOが使える

2階の電気・火災報知器や
庭の自動散水が無線で制御



AM231温度湿度センサー
小指の爪サイズ

離れたものを監視制御する

Tocos無線モジュール

- USB Dongle ToCoStickはシリアル制御
- シリアル制御コマンドは稚拙のepiconで
- `/usr/local/bin/epicon -s 115200 -d 5 -ql /dev/ttyUSB0 -c ${CMD}`
- 無線は水もの？・・・時々不通状態→リトライ
- `:7881150175810000380026C9000C04220000FFFFFFFFFFFFA7`なデータが1秒毎に定期受信→どういうアプリを組むの？
- DOの状態確認が出来ない→動作モードを変更（定期送信を行わない）、DI-DOを1ビット削りループ接続、確認用へ
- 送信時にチェックSUMを付加、安直なscriptでは計算2秒程→Cで・・・TWE-Lite Ver1.6からは送信は自動付加・受信のみチェック
- `pepotocosctl`コマンド完成

離れたものを監視制御する

Tocos無線モジュールの設定

```
# epicon -ql /dev/ttyUSB0 -s 115200
```

+++ ←エコー無し

```
--- CONFIG/MONO WIRELESS TWELITE DIP APP
```

```
SID=0x81006e14/LID=0x78 ---
```

```
a: set Application ID (0x67720102)
```

```
i: set Device ID (--)
```

```
c: set Channels (18)
```

```
x: set Tx Power (03)
```

```
t: set mode4 sleep dur (1000ms)
```

```
y: set mode7 sleep dur (10s)
```

```
f: set mode3 fps (32)
```

```
z: set PWM HZ (1000)
```

```
o: set Option Bits (0x00000012)
```

```
b: set UART baud (38400)
```

```
p: set UART parity (N)
```

```
---
```

```
S: save Configuration
```

```
R: reset to Defaults
```

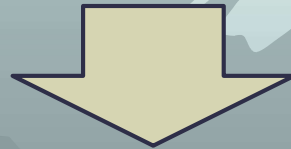
DO4とDI4をループ接続してAPでDO4をLOW/HIへ変化させ情報を読み取ります

動作モードを変更
DIの変化のみ送信、AI変化・定期送信なし

離れたものを監視制御する

Tocos無線モジュール

- 無線は水もの？・・・時々不通でデジタル出力が raspberry pi側と不一致
- 点けた筈の電灯が点かない
- 消した筈の電灯が点きっぱなし
- 2.4G帯が混雑？



- 2.4G帯 3 チャンネルを 1 CHに固定→混雑緩和
- 出力 3 ビット 1 回のコマンドで終わらせる→混雑緩和
- ポーリングタイマーを 2 秒→4 秒→混雑緩和
- raspberry pi側でDOのオンオフ情報を保持させて、数秒間隔で同期させるdaemonで改善→無限リトライ

離れたものを監視制御する Tocos無線モジュール

ToCoStickとraspberry piはシリアル通信epiconを使う

```
/usr/local/bin/epicon -s 115200 -ql /dev/ttyUSB0 -c command_file
```

```
#!/bin/sh
echo -en ":7888AA015C0000X"¥r¥n
msleep 20
read -s -t 1 RD || RD="-1"
#echo $RD >/dev/stderr
echo -en ":7888AA015C03020004X"¥r¥n
msleep 20
read -s -t 1 RD || RD="-1"
msleep 1500
echo -en ":7888AA025C0006X"¥r¥n
```

5CはAM2321の
I2Cアドレス

command_fileの内
容shellスクリプト

TOCOS TWE-LiteとToCoStick（トコスティック） 温湿度計AM2321センサー制御 入力データフォーマット(外部マイコン→TWE)

引用：https://mono-wireless.com/jp/products/TWE-APPS/App_Twelite/step3-I2C.html

:DB88010123230055

^1^2^3^4^5^6^7^9

1: 1バイト: 無線デバイスのアドレス (0xDB: 自分自身, 0x00 ~ 0x7F, 0x00: 親機, 0x78: 全子機)

2: 1バイト: 0x88

3: 1バイト: 要求番号(応答メッセージにそのまま出力される識別子)

4: 1バイト: コマンド (0x1: 書き込み, 0x2: 読み出し, 0x4: 書込み&読み出し)

5: 1バイト: I2Cアドレス (1010100 なら 0x54)

6: 1バイト: I2Cコマンド (最初のコマンドバイト)

7: 1バイト: データサイズ (無い時は 0)

8: Nバイト: データ (データサイズが0のときは、本フィールドは無し)

9: 1バイト: チェックサム

TOCOS TWE-LiteとToCoStick（トコスティック） 温 湿度計AM2321センサー制御 出力データフォーマット(TWE→外部マイコン)

引用：[https://mono-wireless.com/jp/products/TWE-APPS/App_Twelite/
step3-I2C.html](https://mono-wireless.com/jp/products/TWE-APPS/App_Twelite/step3-I2C.html)

:7889AA0201060304017300933E

^1^2^3^4^5^6^7^9

1: 1バイト: 無線デバイスのアドレス

2: 1バイト: 0x89

3: 1バイト: 要求番号

4: 1バイト: コマンド (0x1: Write, 0x2: Read, 0x4: Write&Read)

5: 1バイト: 0:FAIL, 1:SUCCESS

6: 1バイト: データサイズ (無い時は 0)

7: Nバイト: データ (データサイズが0のときは、本フィールドは無し)

TOCOS TWE-LiteとToCoStick（トコスティック）で 温湿度計AM2321センサーのデバック中

センサーのWakeUP送信

: 7888AA015C0000X

温湿度レジスタから4byte読み取り送信

:7888AA015C03020004X

応答

:7889AA010000054

1500ms経過後に読み取り送信・・・4byteと思って読み取りCountを4にする
とダメ

:7888AA025C0006X

応答

:7889AA0201060304022A00F029

何処に温度と湿度があるんやろか

022A湿度⇒ $02 \times 256 + 2 \times 16 + 10/10 \Rightarrow 55.4\%$

00F0温度⇒1バイト目80以上でマイナス、2バイト目は下位

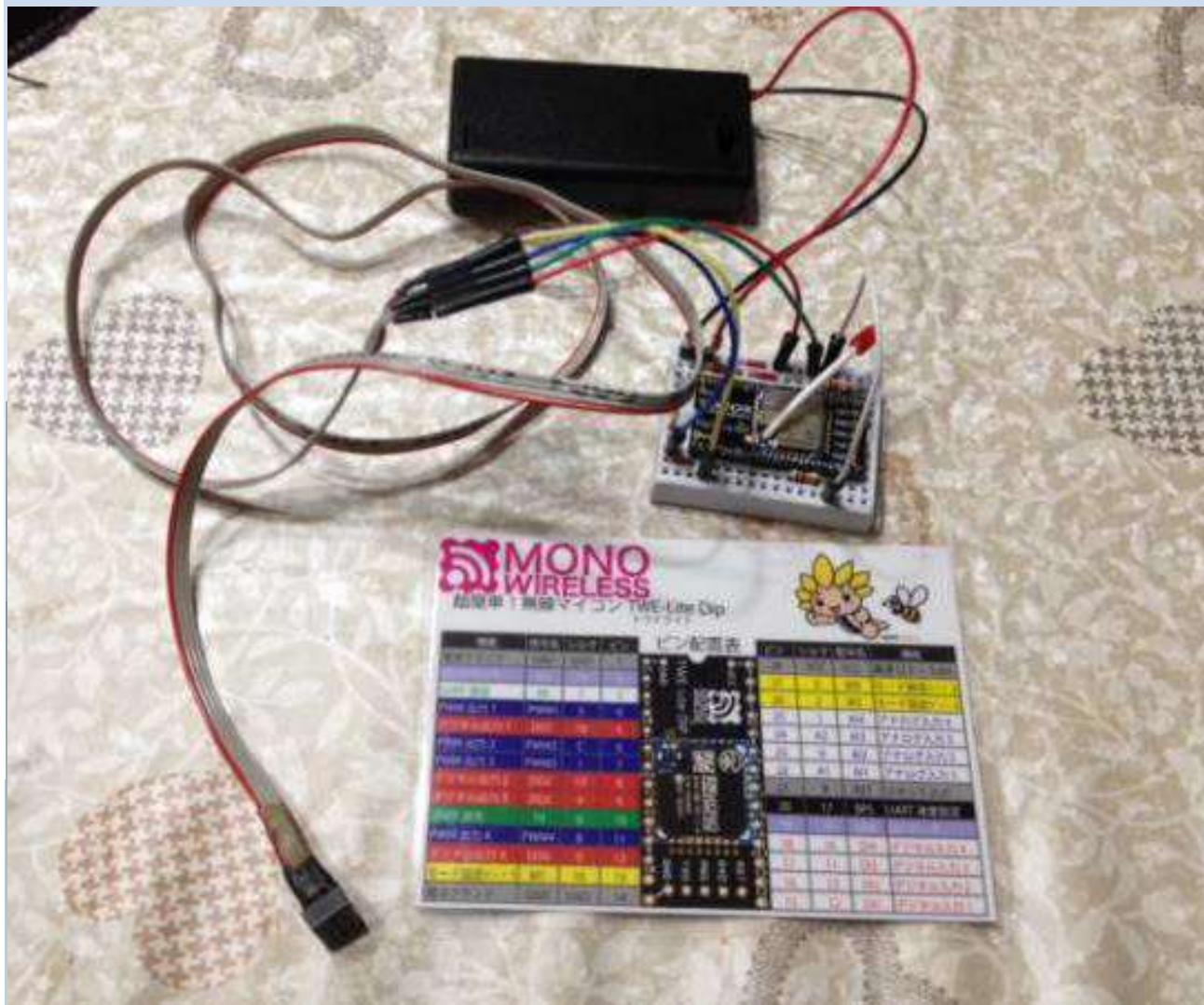
今回はプラスF⇒15

$00 \times 256 + 15 \times 16 + 0/10 = 24.0^{\circ}\text{C}$

TOCOS TWE-LiteとToCoStick（トコスティック）

で温湿度計AM2321センサー

TWE-Lite内蔵ワイヤレス制御電源コンセント



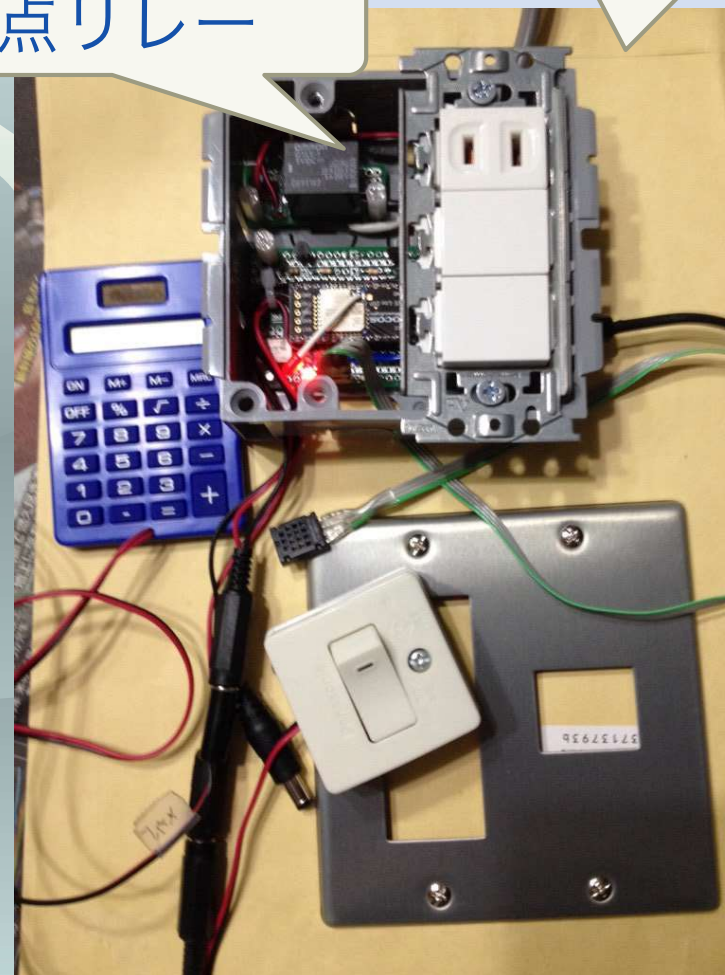
TOCOS TWE-LiteとToCoStick（トコスティック）で無線制御電源コンセント

ジャンパー線だらけ再現性なし

再現性ありKUGADEN
KK-005IIキット改造

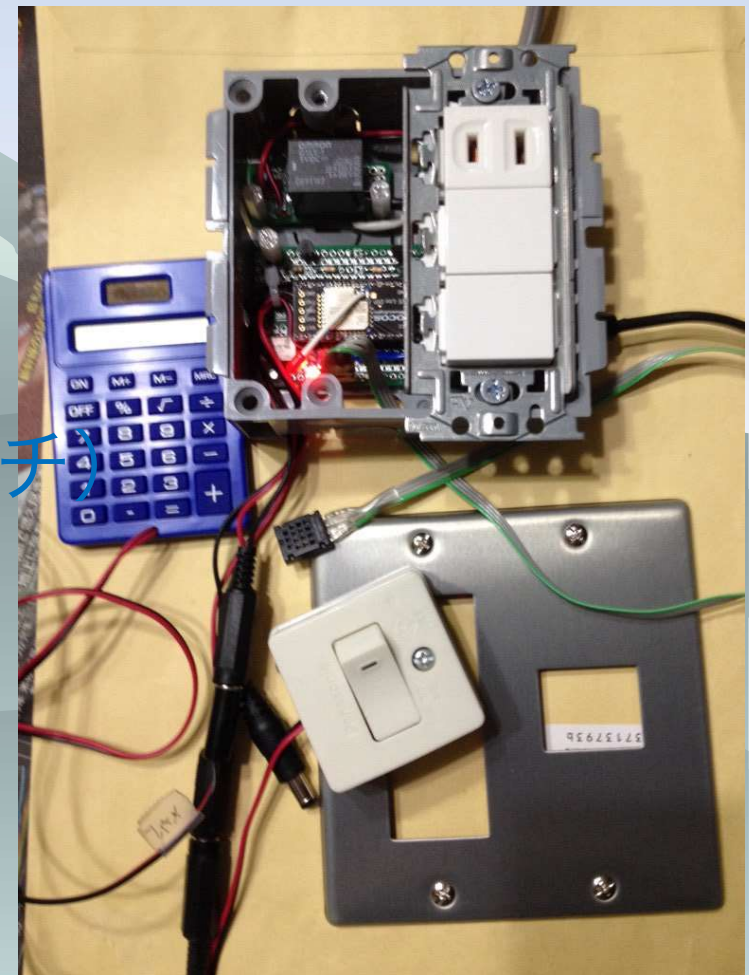
有接点リレー

無接点ソリッドステートスイッチ



TOCOS TWE-LiteとToCoStick（トコスティック）で 無線制御電源コンセント KUGADEN KK-005IIキット改造

- ◆ DO1～3→DI1～3ループ接続カット
- ◆ DO4→DI4
- ◆ AI 1 →電卓ソーラー
- ◆ DI1→スイッチ
- ◆ DO1→5Aリレー（AC100Vスイッチ）
- ◆ DO2→0.3Aリレー（元のまま）
- ◆ I2C→AM2320（温湿度センサー）



TOCOS TWE-LiteとToCoStick（トコスティック）で簡易照度センサー（100均電卓）デバック中

:788115017E8100979C003C67000D291B00085CFFFFFFFC6E

^1^2^3^4^5^6^7^8^9^a^b^c^d^e1e2e3e4ef^g

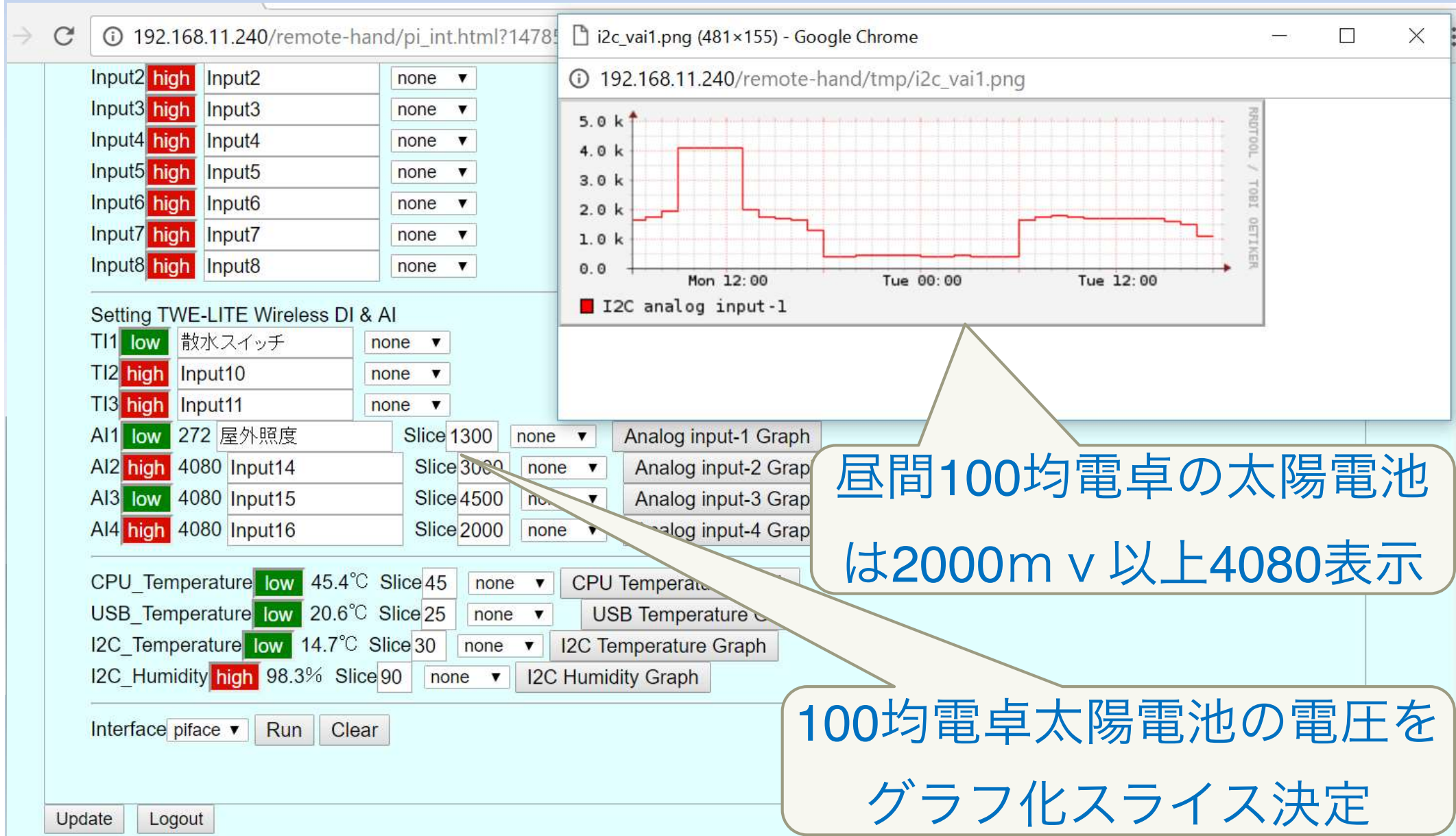
e1~e4: AD1~AD4の変換値。ef: AD1~AD4の補正值

AD値の復元には以下のように計算 $AD1[mV] = (5C * 4 + 0) * 4$ FCの下位2ビットが補正值 (0) $\Rightarrow 1472$



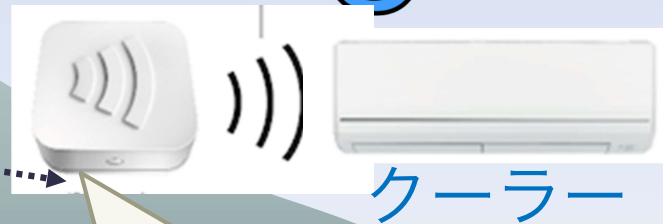
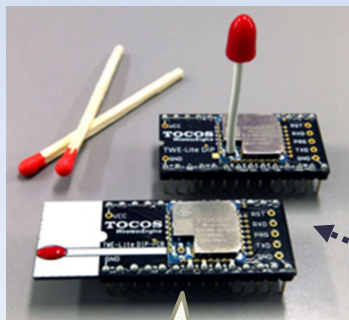
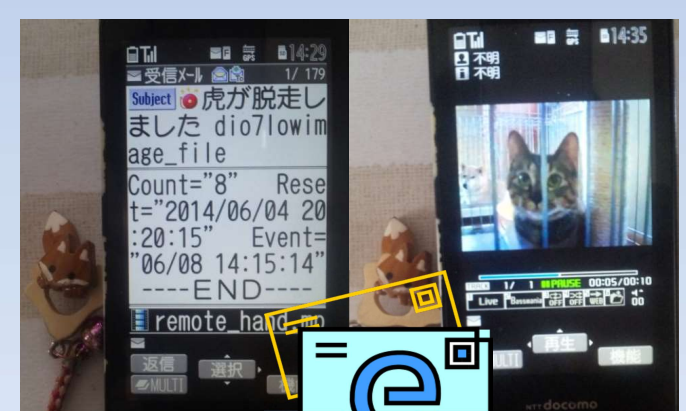
100均電卓のソーラーパネルを
Tocosアナログ入力1に接続

TOCOS TWE-LiteとToCoStick（トコスティック） で簡易照度センサー（100均電卓）デバック中



実践IoTハウス

Home automation & Security



Tocos無線モジュール
離れた物を監視制御

火災報知器発報で動画メール
と電話同時通知

Raspberry Pi + Piface
Gmail連携で動物監視
外部機器の監視制御

IRKit
学習型赤外線リモコン



リモートハンド

実践IoTハウス

Home automation & Security

Management DI(Digital Input)-1

Setting first action to the digital input

火事です **high** Action:low→high パトライトlow Alt none ms none Log
Phone Email Message Count: 8 2017/07/18 14:44:48

防犯センサー **high** Action:low→high none Alt none ms none
Phone Email Message

Input3 **high** Action:low→high none Alt none ms none
Phone Email Message

Update 2015/07/25 08:39:52 #1528
Update 2015/07/25 08:39:37 #1528
Update 2015/07/25 08:38:56 #1527
Update 2015/07/25 08:38:44 #1526
Update 2015/07/25 08:38:31 #1525
Update 2015/07/25 08:37:11 #1524
Update 2015/07/25 08:36:59 #1523

入力high→low→high
22イベントとログ表示
33アクション×2設定

USB or modules
カメラで動画

温度・湿度・アナログ入力の
グラフとスライスでデジタル化

Setting digital output terminal name

Server-Synchronized at 2016/11/08 15:20:36

Output1	low	none	Timer	ms	CPUファン	none
Output2	low	none	Timer	ms	火事テスト	none
Output3	low	none	Timer	ms	窓の電気	none
Output4	low	none	Timer	ms	Output4	none
Output5	low	none	Timer	ms	Output5	none
Output6	low	none	Timer	ms	カ	none
Output7	high	none	Timer	ms	扇	none
Output8	low	none	Timer	ms	C	none

入出力端子へ別名設定

Setting IRKit

IR1 none Timer ms なるべく天井電灯 none IR_dataRea
IR2 none Timer ms クーラー運転 none IR_dataRea

Web Camera1 30 Sec Web Camera2 30 Sec 30 Sec Module Camera 30 Sec
Streaming start Streaming stop Server 1

Setting digital input terminal name

Input1	high	火事です	none
Input2	high	Input2	none
Input3	high	Input3	none
Input4	high	Input4	none
Input5	high	Input5	none
Input6	high	Input6	none
Input7	high	Input7	none
Input8	high	Input8	none

IRKit赤外線データ
登録&出力

Setting TWE-LITE Wireless DI & A

T11	low	散水スイッチ	none
T12	high	Input10	none
T13	high	Input11	none
A11	high	1568 屋外照度	Slice 1300 none Analog input-1 Graph
A12	high	4080 Input14	Slice 3000 none Analog input-2 Graph
A13	low	4080 Input15	Slice 4500 none Analog input-3 Graph
A14	high	4080 Input16	Slice 2000 none Analog input-4 Graph

Tocos無線モジュール
DIO & AI

CPU_Temperature	low	45.4°C	Slice 45 none CPU Temperature Graph
USB_Temperature	low	20.8°C	Slice 25 none USB Temperature Graph
I2C_Temperature	low	15.5°C	Slice 30 none I2C Temperature Graph

実践IoTハウス イベントログを表示

The screenshot displays the diolhigh control interface. On the left, a list of inputs and their corresponding actions is shown. Each input has a 'Phone' and 'Email' field, a status indicator (red 'high' box), and an 'Action: low→high' label. The inputs are: 防犯センサー, 出力1と接続, Input4, Input5, Input6, Input7, and Input8. A dropdown menu is open, showing a list of actions: NONE, 入力3と接続high, 入力3と接続low, Output2high, Output2low, 窓の電気high (highlighted in blue), 窓の電気low, Output4high, Output4low, 散水ポンプhigh, 散水ポンプlow, ななちゃん電気high, ななちゃん電気low, 扇風機high, 扇風機low, 玄関あかりhigh, 玄関あかりlow, あかりsend, クーラー運転, and クーラー停止. On the right, a log window titled 'diolhigh Update Log - Google Chrome' is open, showing a list of update events with timestamps and IDs. A callout box with a red arrow points to the log window, containing the text: 入力毎に2つのイベントhigh、lowのカウンターとログを表示.

Input	Status	Action
防犯センサー	high	Action: low→high
出力1と接続	high	Action: low→high
Input4	high	Action: low→high
Input5	high	Action: low→high
Input6	high	Action: low→high
Input7	high	Action: low→high
Input8	high	Action: low→high

入力毎に2つのイベントhigh、lowのカウンターとログを表示

実践IoTハウス

グラフ表示

RRDtoolでCPU温度
などグラフ表示

cpu_temp.png (481×155) - Google Chrome

192.168.11.240/remote-hand/tmp/cpu_temp.png



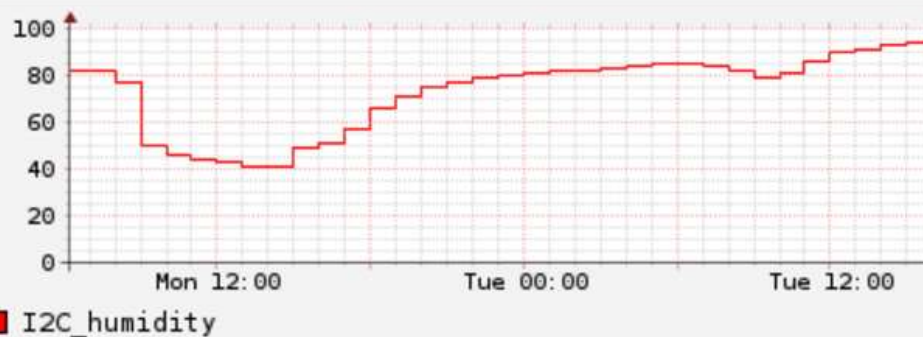
temper.png (481×155) - Google Chrome

192.168.11.240/remote-hand/tmp/temper.png



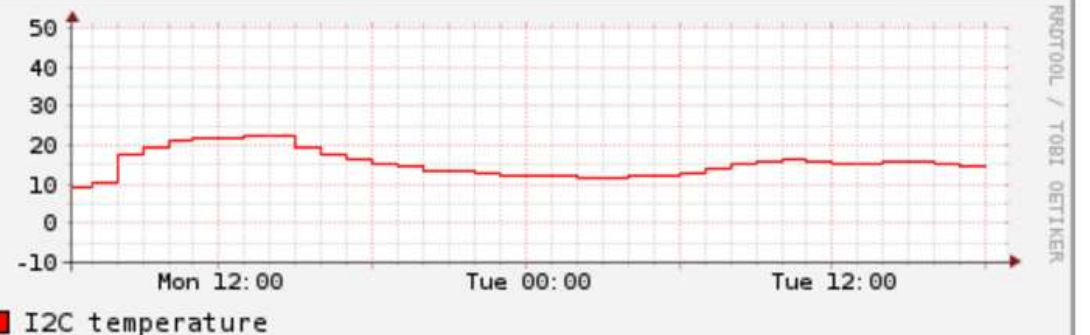
i2c_hum.png (481×155) - Google Chrome

192.168.11.240/remote-hand/tmp/i2c_hum.png



i2c_temp.png (481×155) - Google Chrome

192.168.11.240/remote-hand/tmp/i2c_temp.png



スマホde実践IoTハウス

Home automation & Security

Remote-Hand raspberry_pi ver:0.31 2016.11.8

Control digital output

lo	CPUファン	High/Low
lo	火事テスト	High/Low
hi	窓の電気	High/Low
lo	Output4	High/Low
lo	Output5	High/Low
hi	ななちゃん	High/Low
hi	扇風機	High/Low
lo	Output8	High/Low
Ready	ななろく天	Send
Ready	クーラー運	Send
Ready	クーラー停	Send
none	Output12	Send
none	Output13	Send
none	Output14	Send
lo	散水ポンプ	High/Low
no	Output16	High/Low
hi	玄関電灯	High/Low

Digital input information

hi 火事です
hi Input2
hi Input3
hi Input4
hi Input5
hi Input6
hi Input7
hi Input8
lo 散水スイッ
hi Input10
hi Input11
lo 屋外照度

Ready	クーラー停	Send
none	Output12	Send
none	Output13	Send
none	Output14	Send
lo	散水ポンプ	High/Low
no	Output16	High/Low
hi	玄関電灯	High/Low

Digital input information

hi 火事です
hi Input2
hi Input3
hi Input4
hi Input5
hi Input6
hi Input7
hi Input8
lo 散水スイッ
hi Input10
hi Input11
lo 屋外照度
hi Input14
lo Input15
hi Input16
lo CPU_Tempe
lo USB Tempe
lo I2C Temper
lo I2C Humidit

43.3°C CPU Temperature Graph

USB Temperature 18.8°C USB Temperature Graph

I2C Temperatur 11.5°C I2C Temperature Graph

I2C Humidity 90.3% I2C Humidity Graph

Update

Logout

Javascript+c+shell Script

- ◆ フルスクラッチ
- ◆ cgi bash Script 13000行
- ◆ Helper bash Script 3100行
- ◆ Javascript 3000行
- ◆ c 2000行
- ◆ 気がつけば約20000行余り

Raspberry Piはflashを酷使

/var/logや/tmpをメモリファイルシステムtmpfsへ

```
[root@tmp]# df
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
------------	-----------	------	-----------	------	------------

/dev/root	7254872	2724852	4138444	40%	/
-----------	---------	---------	---------	-----	---

devtmpfs	185960	0	185960	0%	/dev
----------	--------	---	--------	----	------

tmpfs	38048	288	37760	1%	/run
-------	-------	-----	-------	----	------

tmpfs	5120	0	5120	0%	/run/lock
-------	------	---	------	----	-----------

tmpfs	76080	0	76080	0%	/run/shm
-------	-------	---	-------	----	----------

/dev/mmcblk0p1	57214	23118	34096	41%	/boot
----------------	-------	-------	-------	-----	-------

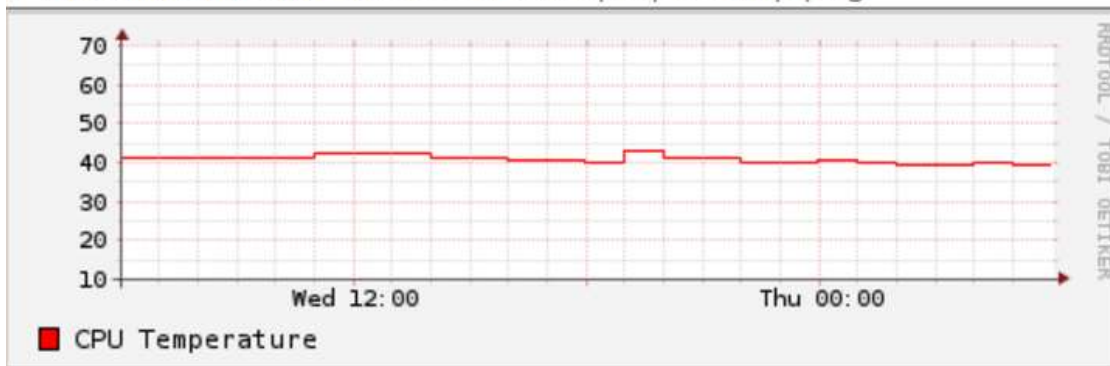
tmpfs	102400	3248	99152	4%	/www
-------	--------	------	-------	----	------

Raspberry piハングアップ

- ◆ CPU温度をリアル表示させたところ、何と58°C付近
- ◆ 急遽清音CPUファンを取り付け、50°C前後で推移
- ◆ 無線LANチップが高温になるので効果絶大
- ◆ CPUファンを回しっぱなしでは芸が無いので、pifaceのリレーでオンオフ
- ◆ CPU温度条件でファンを運転してからは50°Cを超えることは無い
- ◆ ファンを回す

cpu_temp.png (481×155) - Google Chrome

192.168.11.240/remote-hand/tmp/cpu_temp.png



Raspberry piのCPUはスライス温度45°Cでファン制御

Setting digital input terminal name

Input1	high	火事です	none
Input2	high	Input2	none
Input3	high	Input3	none
Input4	high	Input4	none
Input5	high	Input5	none
Input6	high	Input6	none
Input7	high	Input7	none
Input8	high	Input8	none

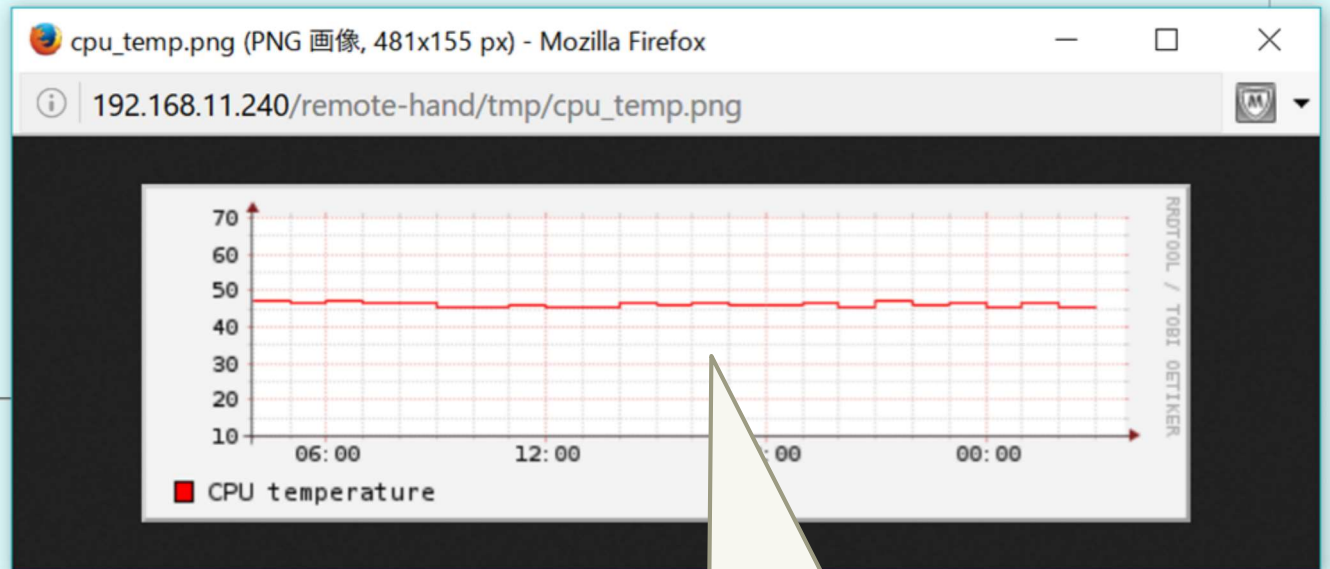
Setting TWE-LITE Wireless DI & AI

TI1	low	散水スイッチ	none
TI2	high	Input10	none
TI3	high	Input11	none

AI1	low	320 屋外照度	Slice 1300	none
AI2	high	4080 Input14	Slice 3000	none
AI3	low	4080 Input15	Slice 4500	none
AI4	high	4080 Input16	Slice 2000	none

CPU_Temperature	low	45.4°C	Slice 45	none	CPU Temperature Graph
USB_Temperature	low	19.7°C	Slice 25	none	USB Temperature Graph
I2C_Temperature	low	11.1°C	Slice 30	none	I2C Temperature Graph
I2C_Humidity	low	73.9%	Slice 90	none	I2C Humidity Graph

Interface



CPU_Temperature low 45.4°C Slice 45 none

Analog input-3 Graph

Analog input-4 Graph

CPU温度45°C一定

Raspberry pi無線LANダウン

- ◆ 無応答時にUSB無線LANチップを抜き差しで復旧
- ◆ 無線LANチップの高温が原因では無かった
- ◆ 定期的にgatewayへping、failしたら無線LANをrestart

```
if ! ping -c $GATEWAY 2>&1 > $PING_LOG;then
```

```
ifdown wlan0
```

```
ifup wlan0
```

```
Fi
```

- ◆ これでもダメ、2.4GHZ帯は限界か
- ◆ やっぱりRaspberry piは有線で使おう
- ◆ 2017/5/22現在CPU冷却用ファンで無線LANチップも冷やしているので安定しています。

実践IoTハウス

Home automation & Security

Setting system Email

Setting operation in Gmail

Gmail User: dokodemodo@gmail.com

Gmail Password:

Mail Address: all@pepolinux.local

Allow Email address

Key Word: dio Subject(keyword)

Mail Check Timer: 30 New Email check interval(Sec)

Jitter: 30 EmailArrival correction(Sec)

Entry ▼

Run

Clear

メールアドレス・キーワード・チェック間隔を設定

実践IoTハウス

Home automation & Security

Server configuration and control

Date: Time:

Host name:

domain:

IP address eth0: / ▼

default gateway:

nameserver1:

nameserver2:

root password:

web user:

web password:

wireless ssid:

key:

IP address wlan0: / ▼

☒ Stop ☐ Restart ☐ Initial setting&Restart

IPアドレスなどネットワーク設定と現在のデータ保存

IPアドレス・メール設定の後リスタートで反映

実践IoTハウス

Home automation & Security

Setting DIO & IRKit & Tocos

Setting digital output terminal name

Server-Synchronized at 2016/11/19 06:17:47

Output1	low	none ▼	ms	CPUファン			none ▼
Output2	low	none ▼	ms	火事テスト			none ▼
Output3	high	none ▼	ms	窓の電気			none ▼
Output4	low	none ▼	ms	Output4			
Output5	low	none ▼	ms	Output5			
Output6	high	none ▼	ms	ななちゃん電灯			
Output7	high	none ▼	ms	扇風機			
Output8	low	none ▼	ms	Output8			none ▼

出力端子の名前設定と現在
データ表示

Setting IRKit

IR1	none ▼	ms	ななろく天井電灯		none ▼	IR_data	Ready	none ▼
IR2	none ▼	ms	クーラー運転		none ▼	IR_data	Ready	none ▼
IR3	none ▼	ms	クーラー停止		none ▼	IR_data	Ready	none ▼
IR4	none ▼	ms	Output12		none ▼	IR_data	none	none ▼
IR5	none ▼	ms	Output13		none ▼	IR_data	none	none ▼
IR6	none ▼	ms	Output14		none ▼	IR_data	none	none ▼

IRKit_IP192.168.11.16 none

Search_Set

Setting Tocos wireless ttyUSB1 ▼ I2C_Temperature&Humidity 06:17 14.6°C 94.9%

TO1	low	none ▼	ms	散水ポンプ		none ▼
TO2	none	none ▼	ms	Output16		none ▼
TO3	high	none ▼	ms	玄関電灯		none ▼

Setting Modem Device ttyUSB0 ▼

実践IoTハウス

Home automation & Security

Setting Modem Device **ttyUSB0** ▼

Voice control **Command** Japanese ▼ Mail:

Recognition start Recognition stop StateStop

Web Camera1 30 Sec Web Camera2 30 Sec Web Camera3 30 Sec Module Camera 30 Sec

Streaming start Streaming stop Server 192.168.11.240

Setting digital input terminal name

Input1	high	火事です	none ▼
Input2	high	Input2	none ▼
Input3	high	Input3	none ▼
Input4	high	Input4	none ▼
Input5	high	Input5	none ▼
Input6	high	Input6	none ▼
Input7	high	Input7	none ▼
Input8	high	Input8	none ▼

入力端子の名前設定と現在
データ表示

Setting TWE-LITE Wireless DI & AI

TI1	low	散水スイッチ	none ▼
TI2	high	Input10	none ▼
TI3	high	Input11	none ▼

AI1	low	320mv	屋外照度	Slice 1300	none ▼	Analog input-1 Graph	Analog input-1 Day Graph
AI2	high	4080mv	Input14	Slice 3000	none ▼	Analog input-2 Graph	Analog input-2 Day Graph
AI3	low	4080mv	Input15	Slice 4500	none ▼	Analog input-3 Graph	Analog input-3 Day Graph
AI4	high	4080mv	Input16	Slice 2000	none ▼	Analog input-4 Graph	Analog input-4 Day Graph

CPU Temperature **low** 45.4°C Slice 45 none ▼ CPU Temperature Graph

USB Temperature **low** 20.7°C Slice 25 none ▼ USB Temperature Graph

I2C Temperature **low** 14.6°C Slice 30 none ▼ I2C Temperature Graph I2C Temperature Day Graph

I2C Humidity **high** 94.9% Slice 90 none ▼ I2C Humidity Graph I2C Humidity Day Graph

実践IoTハウス

Home automation & Security

Setting Sound

Sound Upload

Sound1	ファイルを選択	police_car.mp3	Upload	police_car.mp3	none
Sound2	ファイルを選択	パトカーサ...ン.mp3	Upload	パトカーサイレン.mp3	none
Sound3	ファイルを選択	選択されていません	Upload	none	none
Sound4	ファイルを選択	選択されていません	Upload	none	none
Sound5	ファイルを選択	選択されていません	Upload	none	none

入カイベントで鳴らす音楽
ファイルのUpload

実践IoTハウス

Home automation & Security

Management DI(Digital Input)-1

Setting first action to the digital input

火事です	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input2	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input3	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input4	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input5	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input6	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input7	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	
Input8	high	Action:low→high	NONE	Alt	none	Timer:	ms	none	Log display
Phone:		Email:		Message:		Count:		-	

カイベントに対する処理
の設定ー1とログ表示

実践IoTハウス

Home automation & Security

火事です	high	Action:high→low	Phone	Alt	none	Timer:	ms	none	
Phone:090123456789	Email:	Message:	Count:7	2016/02/29 17:13:03 ~ 2016/02/29 18:26:39					
Input2	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:						
Input3	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:						
Input4	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:-						
Input5	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:-						
Input6	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:-						
Input7	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:-						
Input8	high	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:-						
散水スイッチ	low	Action:high→low	NONE	Alt	none	Timer:	ms	none	
Phone:	Email:	Message:	Count:46	2016/04/19 06:45:11 ~ 2016/05/12 04:44:15					

入カイベントに対する処理
の設定 - 2

実践IoTハウス

Home automation & Security

Setting ping monitoring and digital output action

Setting IP address and the digital Output for monitoring

IP1:	192.168.11.1	DO:	窓の電気high	Timer:	50000	ms	Entry
IP2:		DO:	none	Timer:		ms	Entry
IP3:		DO:	none	Timer:		ms	Entry
IP4:		DO:	none	Timer:		ms	Entry

Ping monitoring interval: 5 Min ▼

Run Clear

Update Logout

PING監視設定・タイムアウトするとPifaceの出力端子をON/OFF

実践IoTハウス

Home automation & Security

Setting ping monitoring and e-mail

Setting IP address and FAIL at which monitoring

IP1: 192.168.11.1	Email1: dokokanodare@gmail.com	Entry ▼
IP2:	Email2:	Entry ▼
IP3:	Email3:	Entry ▼
IP4:	Email4:	Entry ▼

Ping monitoring interval: 5 Min ▼

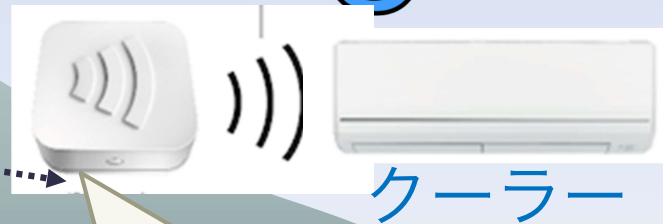
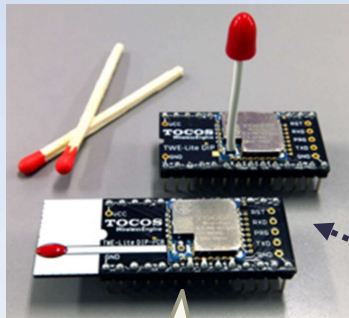
Run

Clear

PING監視設定・タイムアウトするとメールを送る

実践IoTハウス

Home automation & Security



Tocos無線モジュール
離れた物を監視制御

火災報知器発報で動画メール
と電話同時通知

Raspberry Pi + Piface
Gmail連携で動物監視
外部機器の監視制御

IRKit
学習型赤外線リモコン



リモートハンド

古いPC (LinuxBeans) でIoTハウス

Home automation & Security



Raspberry pi vs NotePC

何と5～6倍スピード

raspberry pi

```
# time pepotocosctl 7
```

```
7;1664,4080,4080,4080
```

```
real 0m2.063s
```

```
user 0m0.150s
```

```
sys 0m0.220s
```

```
# time pepotocosctl 5
```

```
2017/04/20,16:07:31,20.3°C,59.6%
```

```
real 0m2.422s
```

```
user 0m0.210s
```

```
sys 0m0.290s
```

元Windows2000、今LinuxBeans

(Ubuntu 12.04.5 LTS)

```
# time pepotocosctl 7
```

```
7;1168,4080,4080,4080
```

```
real 0m0.350s
```

```
user 0m0.048s
```

```
sys 0m0.096s
```

```
# time pepotocosctl 5
```

```
2017/04/20,15:51:35,21.2°C,49.7%
```

```
real 0m0.460s
```

```
user 0m0.088s
```

```
sys 0m0.144s
```

Raspberry pi vs NotePC

何と6～7倍ものの電流値

◆ 消費電流を測定

- ❖ Raspberry Pi 1 Model B 3.5W（実測：24mA）
- ❖ Note PC 20W（実測：160mA）

◆ 電力料金（IOTハウスは24時間365日稼働）

- ❖ 関電従量電灯 A 300kWh超過分 1kWh単価33.32円
- ❖ Raspberry Pi : $3.5 \times 24 \times 365 \times 0.001 \times 33.32 = 1,022$ 円
- ❖ NotePC : $20 \times 24 \times 365 \times 0.001 \times 33.32 = 5,838$ 円

おしまい
ご清聴有難う御座いました

<http://www.pepolinux.com>
Twitter@kujiranodanna
山内