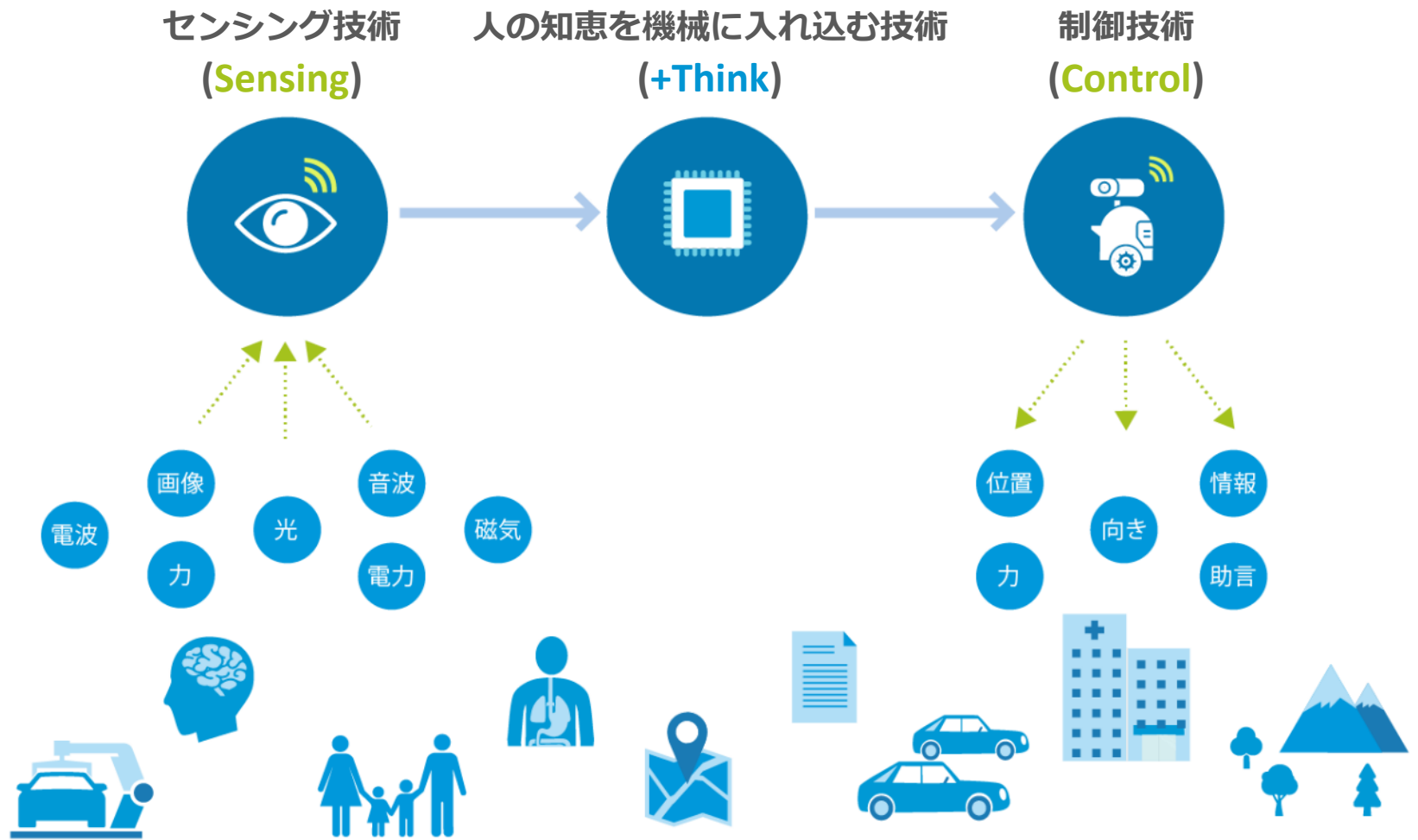


# センサ評価ボードの紹介 形2JCIE-EV01

2020/2/15

オムロン株式会社 事業開発本部  
MEMS開発生産センタ 技術開発部



現場から必要な情報を取り出し、蓄積した現場データと人の知見を用いて、現場に解決策を提供する

IoT社会で欠かせない超小型・高性能なセンサを多数ラインナップ°

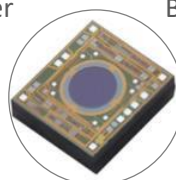


Activity tracker



Blood Pressure Monitor

## 圧カセンサ

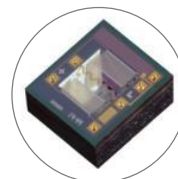


Air conditioner

## IRセンサ



BEMS



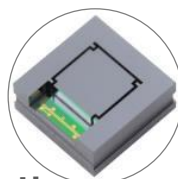
## フローセンサ



Residential fuel cell



Smart gas meter



## 低周波加速度センサ



## センサモジュール



Seismometer



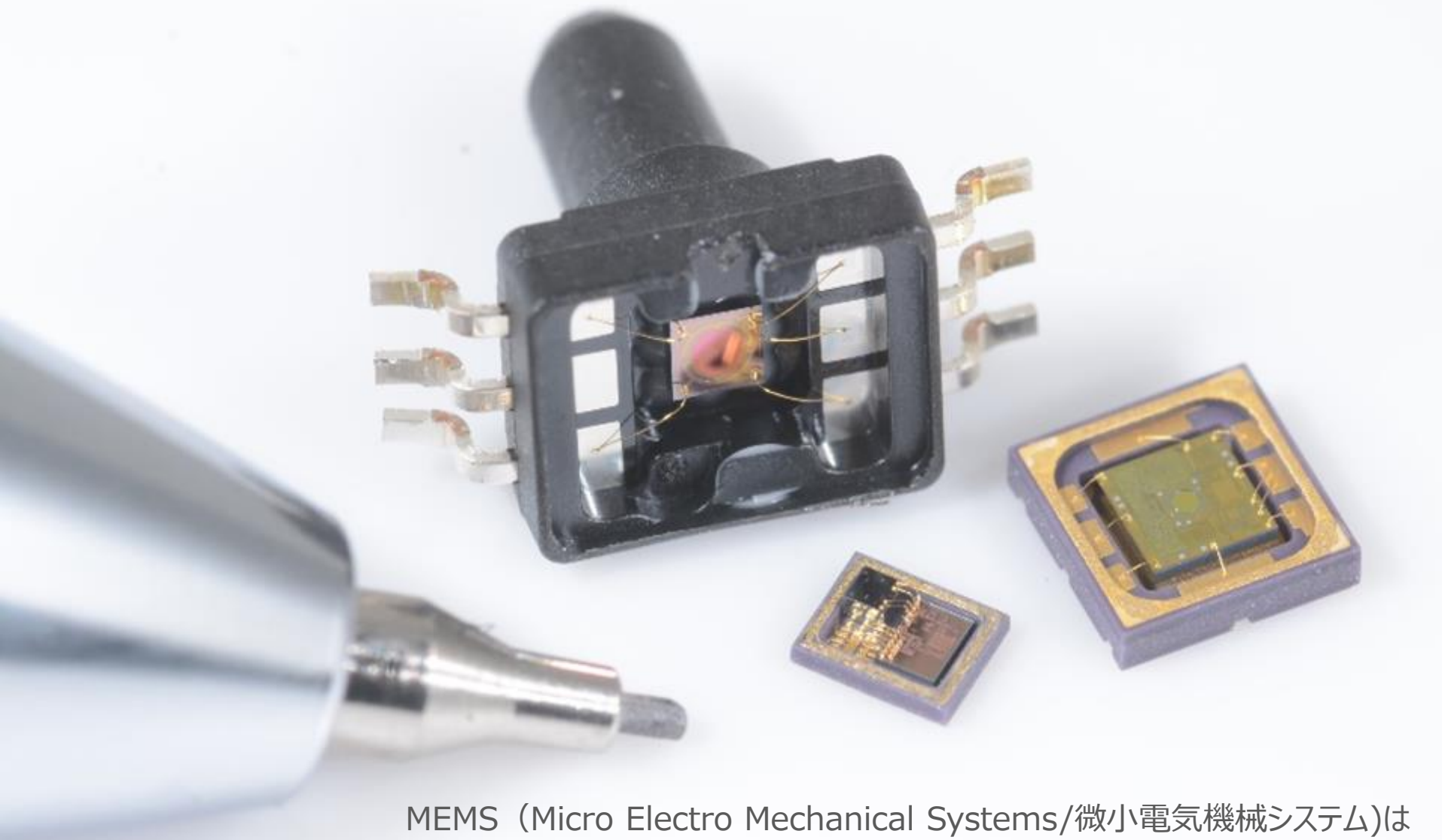
Structural Health Monitoring



Green Environment



Smart Houses, Healthcare



MEMS (Micro Electro Mechanical Systems/微小電気機械システム)は高機能かつ大量生産可能なため、IoTアプリケーション拡大の中で重要な役割を担う

ベースボード、GitHubを組み合わせることで、お客様のPoCを用意に実現

複数のセンサを評価ボードに搭載



温度、湿度、気圧、照度、音、加速度

【センサ評価ボード】

弊社センサと接続可能



サーマルセンサ、差圧センサ、光センサ、ダストセンサ等

Qwiic対応コネクタを搭載



センサデータ取得用の  
参考プログラムをGitHubに配置

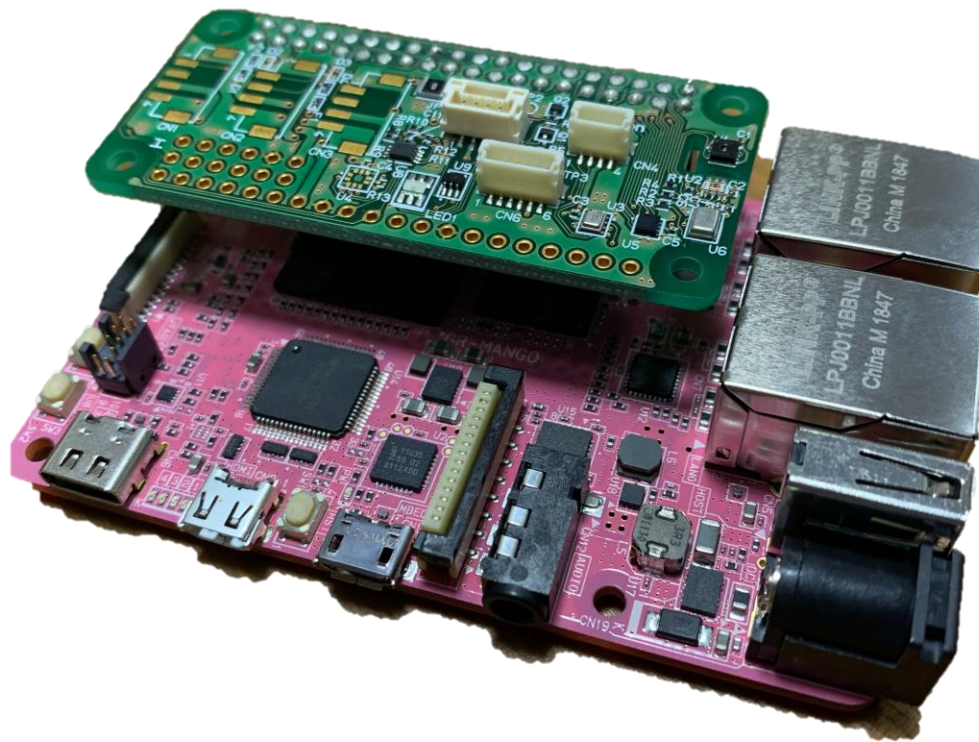
+



他、Arduino, Featherとも接続可能

<https://www.omron.co.jp/ecb/sensor/evaluation-board/2jcie>

GR-MANGOとRaspberryPiはピンコンパチのため、  
ピンソケットを付けることで評価ボードと接続可能





## センサデータ取得プログラムはGitHubで公開

The screenshot shows the GitHub repository page for 'omron-devhub / 2jcieev01-raspberrypi'. The repository is described as 'OMRON sensor evaluation kit 2JCIE-EV01-RP1 with some Raspberry-Pi boards.' It has 22 commits, 1 branch, 0 packages, 0 releases, and 1 contributor. A red box highlights the file '2jcieev01-humi.c' in the file list, which is described as 'debug with actual hardware RasPi-3B' and was committed 9 months ago. A red arrow points from this file to the sample code block on the right.

### サンプルコード

```
24 /* includes */
25 #include "sht30.h"
26
27 #define RASPBERRY_PI_I2C    "/dev/i2c-1"
28 #define I2CDEV              RASPBERRY_PI_I2C
29
30 #define SHT30_STATUSMASK 0xFC1F
31
32 #define conv8s_u16_be(b, n) \
33     ((uint16_t)(((uint16_t)b[n] << 8) | (uint16_t)b[n + 1]))
34
35
36 uint32_t i2c_write_reg16(uint8_t devAddr, uint16_t regAddr,
37                         uint8_t* data, uint8_t length)
38 {
39     uint8_t buf[128];
40     if (length > 127) {
41         fprintf(stderr, "Byte write count (%d) > 127\n", length);
42         return 11;
43     }

```

31.95, 34.2, return code:0

サンプルコードを実行することでセンサデータを取得  
ex) 温度=31.95[°C]、湿度=34.2[%RH]

<https://github.com/omron-devhub>

# GR-MANGOに繋がるセンサ

## 評価ボードに繋がるセンサ

フローセンサ



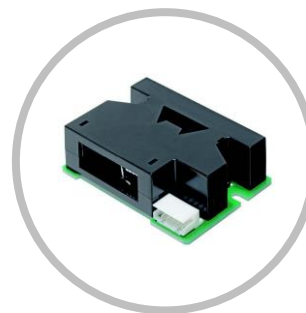
流量・流速検知

限定反射センサ※



物体検知

ダストセンサ※



微粒子検知

IRセンサ

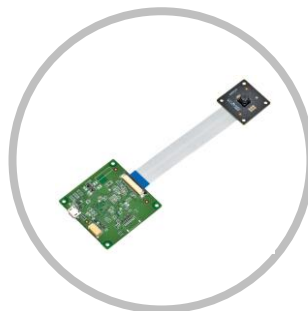


表面温度検知

## GR-MANGOに繋がるセンサ

※Arduino、Featherのみ

人理解画像センサ

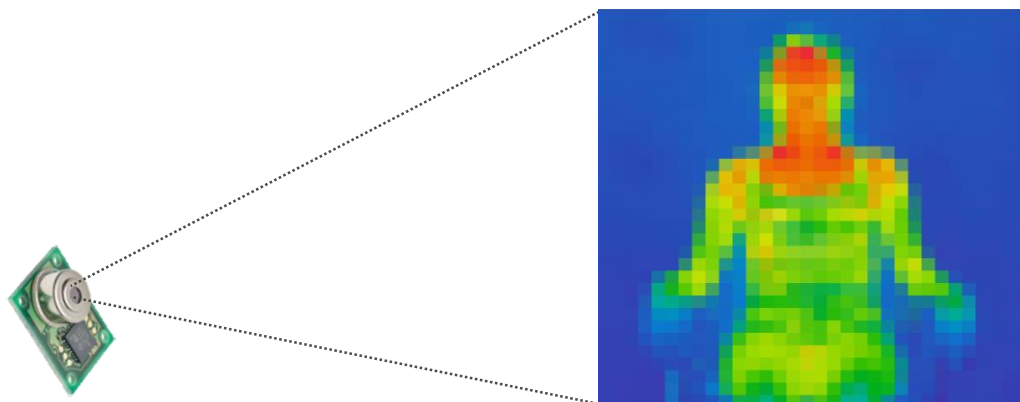


表情・顔検知

<https://www.omron.co.jp/ecb>



IRセンサは、非接触で人やモノの温度検出ができるセンサ



焦電センサは、静止状態の人（モノ）の検知はできないのに対して、  
IRセンサは静止状態でも人（モノ）の検知が可能です。

**焦電センサ** 視野内の「放射熱エネルギーの変化」のみを検出して信号を出力。

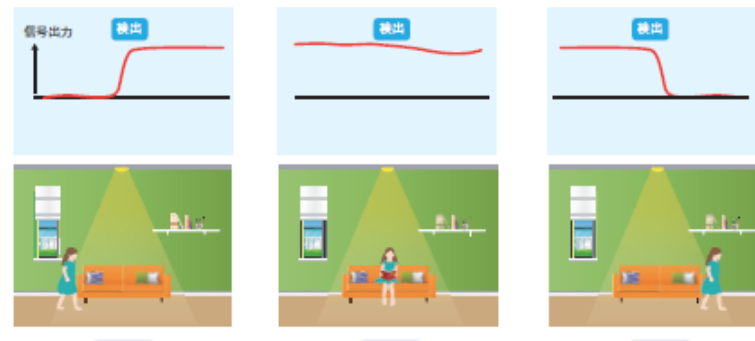


移動状態の人(モノ)の検知は可能

静止状態の人(モノ)の検知は不可

移動状態の人(モノ)の検知は可能

**MEMS非接触温度センサ(サーモパイル)** 視野内の「放射熱エネルギーを常時」検出して信号を出力。



静止・移動状態の人(モノ)の検知でも可能

<https://www.omron.co.jp/ecb/parametric-search?nodeId=804010&nodeParentId=8040>

## 家電



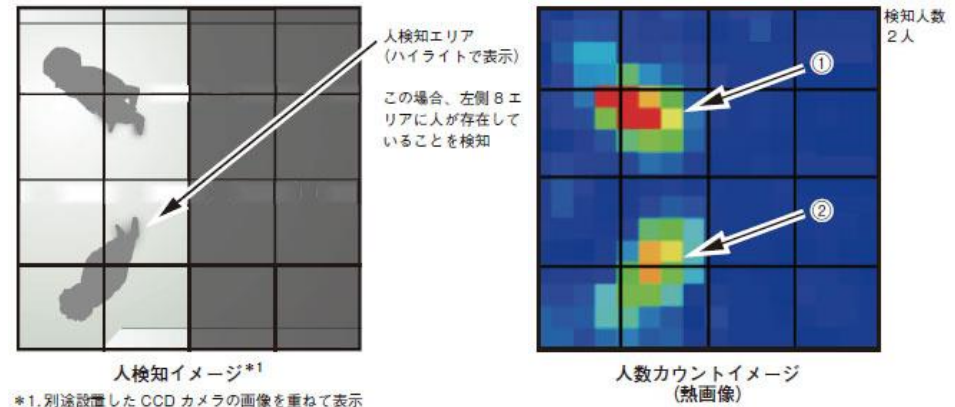
### 対象の温度検知



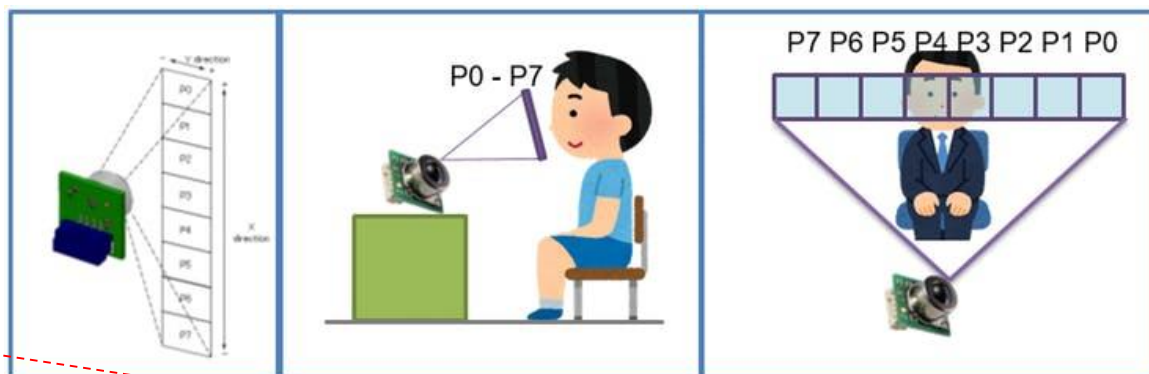
## 人感センサ



### 室内の人数検知



## 在席確認アプリ



```
COM105
21:17:09.707 -> PTAT:27.8, Temperature:22.8, 23.0, 23.7, 23.6, 23.3, 23.1, 22.8, 22.5 [degC], Occupancy:0
21:17:09.954 -> PTAT:27.7, Temperature:22.9, 22.9, 23.8, 23.7, 23.8, 23.8, 22.8, 22.5 [degC], Occupancy:0
21:17:10.207 -> PTAT:27.7, Temperature:22.9, 23.0, 24.0, 26.1, 25.3, 24.0, 22.7, 22.5 [degC], Occupancy:1
21:17:10.454 -> PTAT:27.7, Temperature:23.0, 23.2, 25.6, 28.6, 26.1, 23.6, 22.7, 22.5 [degC], Occupancy:1
21:17:10.707 -> PTAT:27.7, Temperature:23.2, 24.3, 28.7, 29.3, 25.6, 23.2, 22.8, 22.6 [degC], Occupancy:1
21:17:10.955 -> PTAT:27.7, Temperature:23.5, 26.0, 31.2, 28.9, 24.9, 23.1, 22.7, 22.6 [degC], Occupancy:1
21:17:11.208 -> PTAT:27.7, Temperature:23.6, 27.2, 31.9, 28.9, 24.5, 23.0, 22.7, 22.5 [degC], Occupancy:1
21:17:11.454 -> PTAT:27.7, Temperature:23.6, 27.5, 31.9, 28.8, 24.4, 23.0, 22.7, 22.4 [degC], Occupancy:1
21:17:11.708 -> PTAT:27.7, Temperature:23.5, 27.1, 31.7, 28.8, 24.7, 23.0, 22.7, 22.4 [degC], Occupancy:1
21:17:11.954 -> PTAT:27.7, Temperature:23.4, 26.5, 31.4, 28.9, 24.9, 23.0, 22.7, 22.5 [degC], Occupancy:1
21:17:12.208 -> PTAT:27.7, Temperature:23.4, 26.3, 31.1, 29.0, 25.1, 23.0, 22.7, 22.5 [degC], Occupancy:1
21:17:12.455 -> PTAT:27.7, Temperature:23.4, 25.6, 30.5, 30.0, 25.3, 23.0, 22.7, 22.5 [degC], Occupancy:1
21:17:12.755 -> PTAT:27.6, Temperature:23.2, 24.1, 27.3, 29.2, 28.4, 23.1, 22.7, 22.5 [degC], Occupancy:1
21:17:13.009 -> PTAT:27.6, Temperature:22.9, 23.1, 24.2, 25.6, 26.5, 26.7, 23.3, 22.6 [degC], Occupancy:0
21:17:13.256 -> PTAT:27.7, Temperature:22.9, 23.0, 23.7, 23.6, 23.4, 23.8, 24.0, 23.2 [degC], Occupancy:0
21:17:13.509 -> PTAT:27.6, Temperature:22.8, 23.0, 23.7, 23.6, 23.2, 22.8, 22.6, 22.7 [degC], Occupancy:0
```

IRセンサの温度データから  
座席の在不在を確認

閾値を設定し、閾値を  
超える(在席)場合は1、  
超えない場合は0と出力

<https://www.hackster.io/omronelectroniccomponents/seat-occupancy-detect-omron-d6t-thermal-sensor-with-arduino-38e707>

- 評価ボード、IRセンサの無償サンプルをご用意しました(数量限定)
- 評価ボード、IRセンサのレビューをがじえるねの掲示板(がじえつとるねさすコミュニティ)にして頂ける方に差し上げます。
- GR-MANGO×評価ボード×IRセンサを使って、面白いものを作ってください！

**どんなアイデアでも構いません。面白いものを待っています！**



ご清聴ありがとうございました

<https://www.omron.co.jp/>