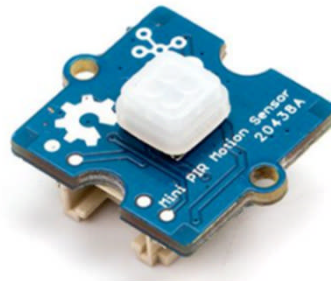


IoTデバイス活用マニュアル

センサ種類：PIR(Passive Infrared Ray)人感モーションセンサー
型番：次項参照
ストーリー：人感センサーの所定条件を満たしたときの出力



対応センサーはいずれもSeeed社のGrove仕様です。

出力は、検出無しでLow、検出有りHIGH

人などが発する赤外線に反応します。モーションセンサーなので赤外線を発していても動きがないと所定時間経過後に検出無しになりますのでご注意ください。



- 電源電圧: 2.7~3.3 V
- 検出角度: 水平80度/垂直55度
- 検出距離: 最大2 m (25℃)

GROVE - ミニPIRモーションセンサ
SKU 101020353

<https://www.switch-science.com/catalog/3584/>



- 電源電圧: 3~5V
- 検出角度: 120度
- 検出距離: 最大6m
- 検出距離、ホールド時間調整可能(外付け部品必要)

GROVE - PIRモーションセンサ
SKU 101020020

<https://www.switch-science.com/catalog/1049/>



3V-5V
最大距離: 3-12m
角度: <100°

ミニサイズ
シンプルで直感的

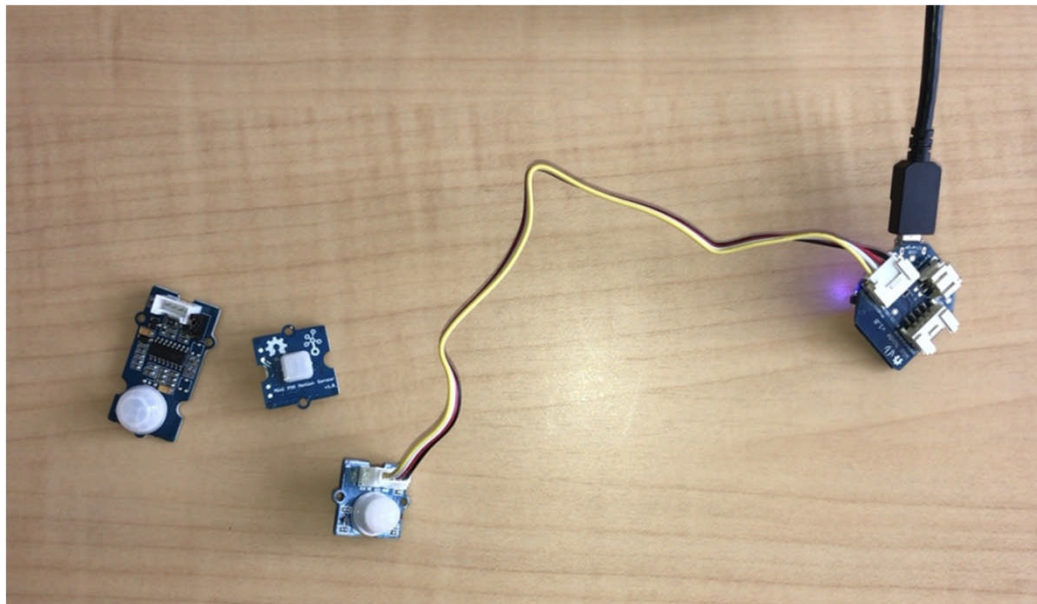
Grove - Digital PIR Motion Sensor(12m)
SKU 101020793

<https://www.seeedstudio.com/Grove-Digital-PIR-Motion-Sensor-p-4524.html>

改訂記録：

2021/9/4 初版作成 門奈

2021/9/18 改訂 門奈 millis()はオーバーフロー時に誤動作の可能性があるため使用しないように修正



接続例

プログラムの概要説明

センサーはモーションセンサのため、検出していても動きがなくなると検出なしとの出力となります。この特性を考慮し、次の通り2つのプログラムを提供します

◆プログラム（1）

検出があれば速やかに"1"を出力、約5秒間検出がなければ"0"を出力
（センサーからの出力が検出になっていると0.1秒毎に"1"を出力し続けます）

◆プログラム（2）

約10秒間に1回でも検出があると"1"を出力、無ければ"0"を出力
検出の有無にかかわらず約10秒単位で"1"か"0"が出力されます

・いずれも検出したときには、WioNodeの青LEDが点灯します

WioNodeプログラム (1) 検出が有れば速やかに”1”を出力、約5秒間検出がなければ”0”を出力
前半と後半のすべてをIDEエディターへコピーして、Wifi (SSID,PASS) およびiothub接続文字列を入力してください。

前
半

```
#include <AzureIoTHub.h>

int fail_count = 0;
const uint8_t PORT0A = 1;
const uint8_t PORT0B = 3;
const uint8_t PORT1A = 4;
const uint8_t PORT1B = 5;
const uint8_t PORT_POWER = 15; // (common with RED_LED)
//set PORT_POWER as HIGH for power supply. Low means no power.

const uint8_t FUNC_BTN = 0;
const uint8_t BLUE_LED = 2;
const uint8_t RED_LED = PORT_POWER;

const uint8_t UART_TX = PORT0A;
const uint8_t UART_RX = PORT0B;
const uint8_t PIR = PORT1B;

int pretime = 0;

void setup()
{
  Serial.begin(115200);
  Serial.println("");
  Serial.println("*** Wake up WioNode...");

  // WiFi及びAzure周りの初期化
  WiFi.begin("SSID", "PASS"); // 2.4GHz帯のWiFiを指定すること
  Azure.begin("KEY"); //YourKey
  Azure.setCallback(azureCallback);

  // IO周りの初期化
  pinMode(PIR, INPUT);
  pinMode(FUNC_BTN, INPUT);
  pinMode(BLUE_LED, OUTPUT);
  pinMode(PORT_POWER, OUTPUT);
  digitalWrite(PORT_POWER, HIGH);
  digitalWrite(BLUE_LED, HIGH);
  delay(5);
  Serial.println("WioNode With PIR Motion Sensor started!");
}

void azureCallback(String s) {
  Serial.print("azure Message arrived [");
  Serial.print(s);
  Serial.println("] ");
}
```

以下50は5秒を指します。

後
半

```
//検出が有れば速やかに”1”を出力、5秒間検出がなければ”0”を出力
void loop()
{
  if (WiFi.status() == WL_CONNECTED) {
    Azure.connect();

    int detect = 0;
    detect = digitalRead(PIR);

    if((detect > 0 )||(pretime <= 0)){
      pretime = 50; // 50*100ms -> 5sec
      digitalWrite(BLUE_LED, LOW); // LED on

      Serial.print("PIR Motion: ");
      Serial.print(detect);
      Serial.println("");

      DataElement a = DataElement();
      // Azureへ送るデータの用意
      a.setValue("sensor","PIR");
      a.setValue("EspValue", detect);

      // Azure IoT Hub ヘブッシュ
      Azure.push(&a);
    }

    delay(100); // 100mSec wait
    // wait中は青LEDを消灯する
    digitalWrite(BLUE_LED, HIGH); // LED off
    pretime = pretime - 1;

  } else {
    Serial.print("Not connected to the Internet:");
    Serial.println(WiFi.status());
    delay(250);
  }
}
```

WioNodeプログラム (2) 10秒間に1回でも検出があると“1”を出力、無ければ“0”を出力

前半と後半のすべてをIDEエディターへコピーして、Wifi (**SSID,PASS**) および **iothub接続文字列**を入力してください。

前半

```
#include <AzureIoTHub.h>

int fail_count = 0;
const uint8_t PORT0A = 1;
const uint8_t PORT0B = 3;
const uint8_t PORT1A = 4;
const uint8_t PORT1B = 5;
const uint8_t PORT_POWER = 15; // (common with RED_LED)
//set PORT_POWER as HIGH for power supply. Low means no power.

const uint8_t FUNC_BTN = 0;
const uint8_t BLUE_LED = 2;
const uint8_t RED_LED = PORT_POWER;

const uint8_t UART_TX = PORT0A;
const uint8_t UART_RX = PORT0B;
const uint8_t PIR = PORT1B;

int pretime;

void setup()
{
  Serial.begin(115200);
  Serial.println("");
  Serial.println("*** Wake up WioNode...");

  WiFi.begin("SSID", "PASS"); // 2.4GHz帯のWiFiを指定すること
  Azure.begin("KEY"); //YourKey
  Azure.setCallback(azureCallback);

  // -----
  // IO周りの初期化
  pinMode(PIR, INPUT);
  pinMode(FUNC_BTN, INPUT);
  pinMode(BLUE_LED, OUTPUT);
  pinMode(PORT_POWER, OUTPUT);
  digitalWrite(PORT_POWER, HIGH);
  digitalWrite(BLUE_LED, HIGH);
  delay(5);

  Serial.println("WioNode With PIR Motion Sensor started!");
}

void azureCallback(String s) {
  Serial.print("azure Message arrived [");
  Serial.print(s);
  Serial.println("] ");
}
```

後半

以下**100**は10秒を指します。

```
//10秒間に1回でも検出があると"1"を送信する。
void loop()
{
  if (WiFi.status() == WL_CONNECTED) {
    Azure.connect();

    unsigned char detect = 0;
    digitalWrite(BLUE_LED, HIGH); // LED off
    pretime = 100; //100*100ms-> 10sec

    for(int i = 0; i < pretime; i++){
      if(digitalRead(PIR)>0){
        detect=1;
        digitalWrite(BLUE_LED, LOW); // LED on
      }
      delay(100); //0.1s Wait
    }

    Serial.print("PIR Motion: ");
    Serial.print(detect);
    Serial.println("");
    DataElement a = DataElement();
    // Azureへ送るデータの用意
    a.setValue("sensor","PIR");
    a.setValue("EspValue", detect);

    // Azure IoT Hub へプッシュ
    Azure.push(&a);

  } else {
    Serial.print("Not connected to the Internet:");
    Serial.println(WiFi.status());
    delay(250);
  }
}
```

Stream Analyticsのクエリ

WioNodeのプログラムの種類にかかわらず共通です

```
SELECT
    Dev as device,
    DATEADD(hour, 9, EventEnqueuedUtcTime) as time,
    EventEnqueuedUtcTime as utctime,
    params.sensor as sensor,
    params.espvalue as value,
    params.duration as duration
INTO
    outputpowerbi
FROM
    inputiothub
```

```
SELECT
    Dev as device,
    DATEADD(hour, 9, EventEnqueuedUtcTime) as time,
    EventEnqueuedUtcTime as utctime,
    params.sensor as sensor,
    params.espvalue as value,
    params.duration as duration
INTO
    outputcosmosdb
FROM
    inputiothub
```

本PIRでは、
“duration”の部分は使っていません。

PowerBI

データセットを開く



PowerBI レポートの編集



PowerBI レポートの編集

Power BI マイワークスペース

PIR | データは 21/8/22 に更新されました

検索

ファイル ビュー 閲覧表示 モバイルレイアウト 質問... 探 テキストボ... 図 ボ... ビジュー... 更. このページを複... 保存 ダッシュボードに... Teams でのチ...

人感モーションセンサー稼働状況 (直近12時間)

device 95monna03

人感モーションセンサー稼働状況 (直近10分間)

device 95monna03

表示する時間条件を設定してください

フィルター

検索

device
が 95monna03 で...

sensor
が PIR である

time
(すべて) です

utctime
3:18:27 - 3:28:27

フィルターの種類

相対時間

次の値のときに項目を表示:

最近の

10

分

フィルターを適用

value
(すべて) です

ここにデータ フィールド...

視覚化

フィールド

検索

fsdatatable2021summer

- ☒ device
- ☐ Σ duration
- ☐ sensor
- ☒ time
- ☐ utctime
- ☒ Σ value

軸

time

凡例

device

value

第 2 の値

ここにデータ フィールド...

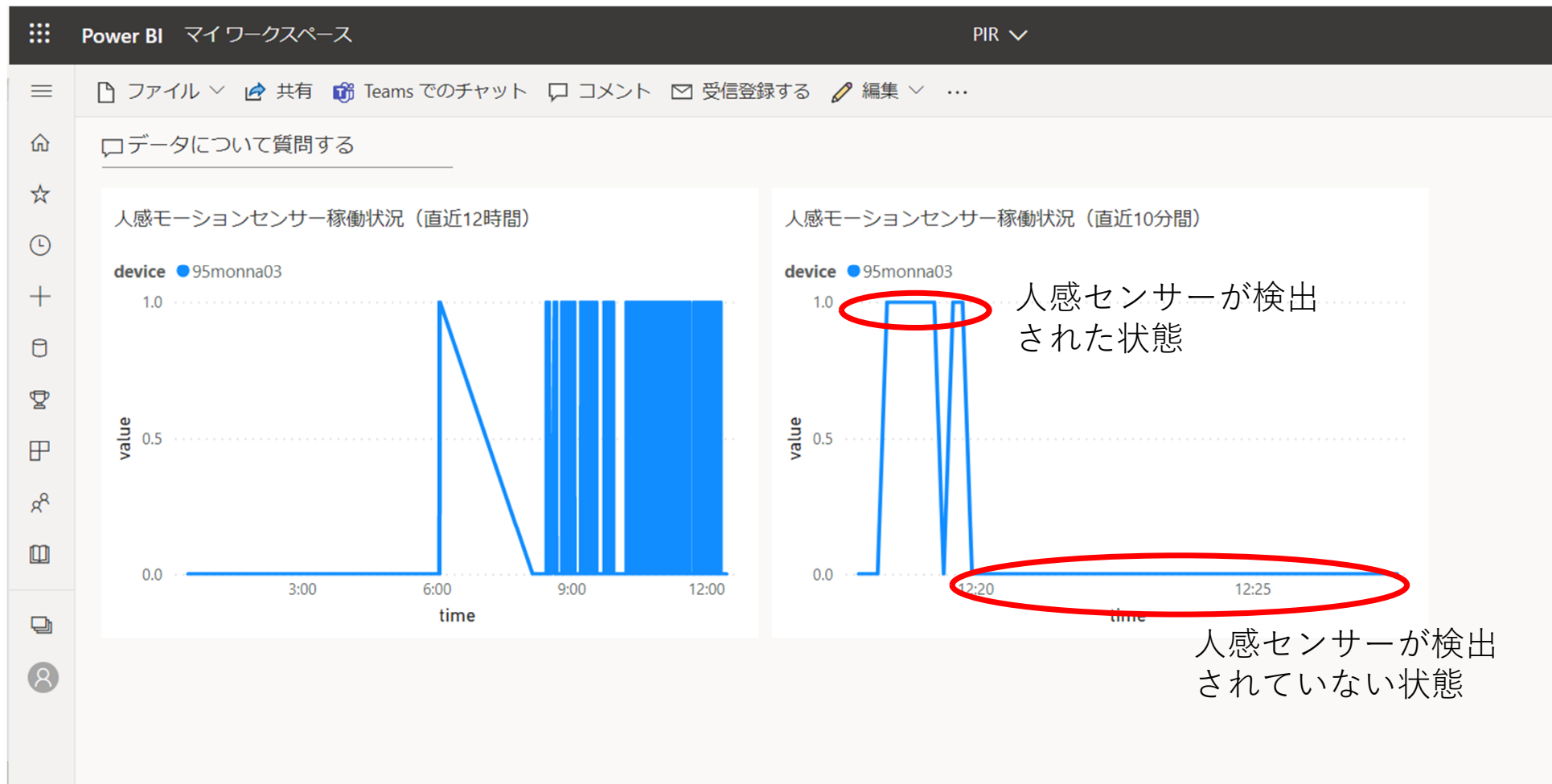
スモール マルチプル

ここにデータ フィールド...

ヒント

ページ 1

PowerBI ダッシュボードの表示



PIRセンサー反応時にアラーム発報する手順

PowerBI レポートの編集 KPIを設定する

Power BI マイワークスペース

PIR | データは 21/8/22 に更新されました

検索

ファイル ビュー 閲覧表示 モバイルレイアウト 質... 挿... テキスト... 図... ボ... ビジュ... 更... このページを... 保存 ダッシュボードに... Teams での...

人感モーションセンサー稼働状況 (直近12時間)

device 95monna03

value

time

人感モーションセンサー稼働状況 (直近10分間)

device 95monna03

value

time

最後に、KPIを任意のダッシュボードへピン留めする

表示する時間は任意に設定ください

フィールド

検索

fsdatatable2021summer

- ☐ device
- ☐ duration
- ☐ sensor
- ☒ time
- ☐ utctime
- ☒ value

device

が 95monna03 で...

フィルターの種類

基本フィルター

検索

☒ すべて選択

☒ 95monna03 1013

sensor

が PIR である

フィルターの種類

基本フィルター

検索

☒ すべて選択

☒ PIR 1013

utctime

2021/10/24 13:58:...

フィルターの種類

相対時間

次の値のときに項目を表示:

最近の

12

時間

フィルターを適用

インジケータ

value

トレンド軸

time

ターゲットにする目標

ここにデータ フィールド...

ドリルスルー

クロス レポート

オフ

すべてのフィルターを保持...

オン

ドリルスルー フィールド...

ダッシュボードを確認、アラートの設定

The screenshot shows the Power BI 'マイワークスペース' (My Workspace) interface. The top navigation bar includes 'Power BI', 'マイワークスペース', a user profile 'PIR', and a search bar. Below the navigation bar, there's a toolbar with options like 'ファイル', '共有', 'Teamsでのチャット', 'コメント', '受信登録する', and '編集'. The main content area displays two charts. The top chart, titled '人感モーションセンサー稼働状況 (直近10分間)', shows a line graph with blue vertical bars representing sensor activity over time. A context menu is open over this chart, listing various actions. The bottom chart, titled '人感モーションセンサー稼働状況 (直近12時間)', shows a similar line graph but for a longer period. A red arrow points from the '...' menu icon on the bottom chart to the first option in the context menu. Another red arrow points from the 'アラートを管理' option in the context menu to the text '②「アラートを管理」をクリック'.

Power BI マイワークスペース PIR

検索

ファイル 共有 Teamsでのチャット コメント 受信登録する 編集

データについて質問する

人感モーションセンサー稼働状況 (直近12時間)

device 95monna03

value

time

人感モーションセンサー稼働状況 (直近10分間)

アラートを管理

②「アラートを管理」をクリック

① 「・・・」をクリック

ダッシュボードでアラートの設定

①「アラートルールの追加」をクリック

The screenshot displays the Power BI 'マイワークスペース' (My Workspace) interface. The main area contains three charts related to '人感モーションセンサー稼働状況' (Human Motion Sensor Operation Status). The top-left chart shows data for device '95monna03' over the last 12 hours. The top-right chart shows data for device '95monna03' over the last 10 minutes. The bottom-left chart shows a summary bar chart with a large '0.00' value. On the right side, the 'アラートを管理' (Manage Alerts) panel is open. It features a yellow button '+ アラートルールの追加' (Add Alert Rule) at the top. Below it, the configuration for an alert rule is shown, including a title, a condition set to 'value' greater than or equal to '0', and notification frequency options. A red arrow points from the text '①「アラートルールの追加」をクリック' to the yellow button. Another red arrow points from the text '②条件を設定' (Set condition) to the condition configuration section.

Power BI マイワークスペース

PIR

検索

データについて質問する

人感モーションセンサー稼働状況 (直近12時間)

device ● 95monna03

value

time

人感モーションセンサー稼働状況 (直近10分間)

device ● 95monna03

value

time

人感モーションセンサー稼働状況 (直近12時間)

0.00

アラートを管理

+ アラートルールの追加

人感モーションセンサー稼働状況 (直近12時間)

アクティブ

オン

アラートタイトル

人感モーションセンサー稼働状況 (直近12時間)

次のアラートルールを設定

value

条件

上

しきい値

0

通知の最大頻度

☐ 最大で 24 時間に 1 回

☒ 最大で 1 時間に 1 回

[Microsoft Power Automate を使用して追加のアクションをトリガーする](#)

保存して閉じる

キャンセル

②条件を設定