



# IoTデバイス活用マニュアル

**センサ種類 : Led Button**


**型番 : Grove - LED Button**

ストーリー : ボタンを押すたびにLEDをオン・オフし、その点灯状況を把握する。



改訂記録：

2020/11/25 初版作成 豊住





# 1.Arduinoのプログラム

url:xxxxxxxxxxxxxxxxxxxxxxxxxxxx



## 補足1 LEDの状態の送信頻度について

```
pinMode(FUNC_BTN, INPUT);  
pinMode(BLUE_LED, OUTPUT);  
pinMode(PORT_POWER, OUTPUT);  
digitalWrite(PORT_POWER, HIGH);  
pinMode(buttonPin, INPUT);  
pinMode(ledPin, OUTPUT);  
digitalWrite(ledPin, ledState);  
delay(5);  
prev = 0;  
interval = 5000; //データ送信頻度
```

delay ();で設定をするとその間ボタンの挙動が止まるため、左記のintervalで割り込み制御する。  
初期設定では5秒に一回Azureへ送信する。



## 2. Stream Analyticsのクエリ変更

電流計と同じクエリで可能



# クエリの説明

SELECT

```
Dev as device,  
System.Timestamp As utctime,  
DATEADD(hour, 9, System.Timestamp) as time,  
params.espvalue as value,  
params.duration as duration
```

INTO

outputpowerbi

FROM

inputiothub TIMESTAMP BY EventEnqueuedUtcTime

SELECT

```
Dev as device,  
System.Timestamp As utctime,  
DATEADD(hour, 9, System.Timestamp) as time,  
params.espvalue as value,  
params.duration as duration
```

INTO

outputcosmosdb

FROM

inputiothub TIMESTAMP BY EventEnqueuedUtcTime

## クエリの説明

params	Dev	Id
{ "LED": "condition", "EspValue": 0, "duration": 5 }	toyozam_button	2030045
{ "LED": "condition", "EspValue": 0, "duration": 5 }	"toy	
{ "LED": "condition", "EspValue": 0, "duration": 5 }	"toy	
{ "LED": "condition", "EspValue": 1, "duration": 5 }	"toy	
{ "LED": "condition", "EspValue": 1, "duration": 5 }	"toy	
{ "LED": "condition", "EspValue": 1, "duration": 5 }	"toy	
{ "LED": "condition", "EspValue": 1, "duration": 5 }	"toy	
{ "LED": "condition", "EspValue": 1, "duration": 5 }	"toy	

LEDのオンオフ状態は、  
【1/0】で入手している

点灯時 : 1  
消灯時 : 0



# 3. PowerBIでのレポートの作成



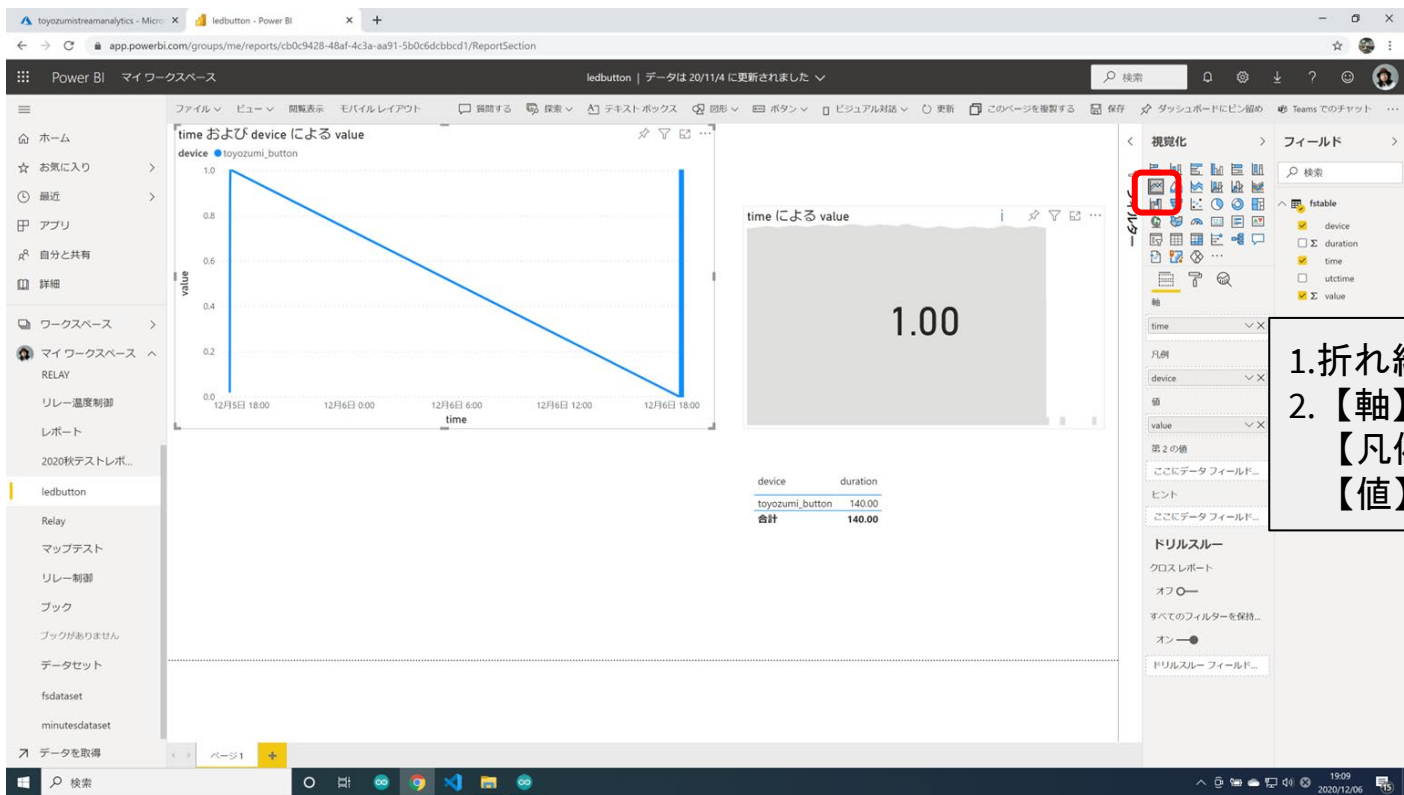


# データセットを開く

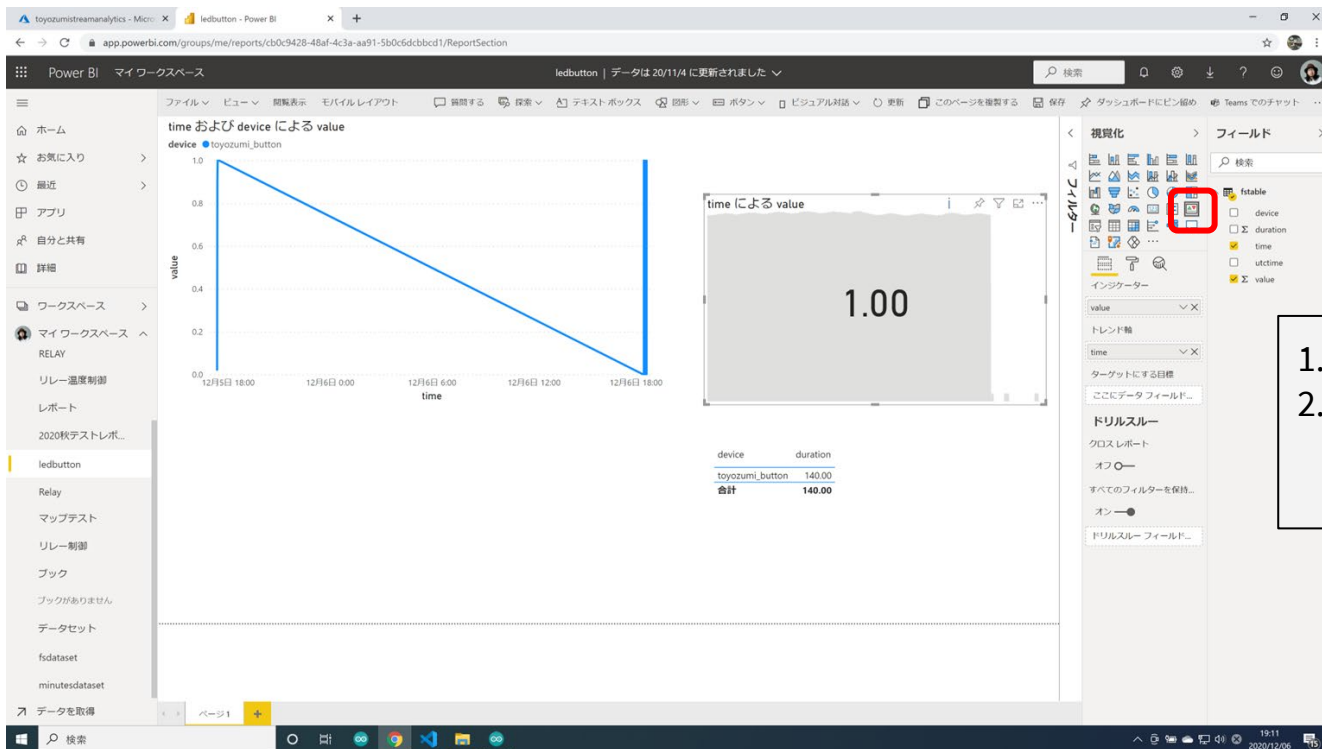
The screenshot shows the Microsoft Power BI web application interface. The left sidebar contains a navigation menu with the following items: ホーム, お気に入り, 最近, アプリ, 自分と共有, 詳細, ワークスペース, and マイワークスペース. Under 'マイワークスペース', there is a list of items: ダッシュボード, 2020秋テスト, リレー温度制御, レポート, 2020秋テストレポ..., マップテスト, リレー制御, ブック, and データセット. The 'データセット' item is highlighted with a red rectangle. Below it, two dataset names are visible: 'fsdataset' and 'minutesdataset'. The main content area displays a message in Japanese: 'データを使用して視覚エフェクトをビルドする' and 'レポートキャンバスにフィールドペインからフィールドを選択またはドラッグします.' Below this message is a diagram showing a dashed box representing a report canvas and a list of fields with a checkmark indicating selection. The right sidebar contains three panels: 'フィルター' (Filters), '視覚化' (Visualizations), and 'フィールド' (Fields). The 'フィールド' panel shows a list of fields: 'fstable', 'device', 'time', 'value', and 'value3'. The '視覚化' panel shows various visualization icons. The 'フィルター' panel shows a search bar and two sections: 'このページでのフィルター' and 'すべてのページでのフィルター', each with a 'ここにデータ フィールド...' link. The bottom status bar shows 'データを取得' and 'ページ 1'.

任意のデータセットを開く  
\* サンプルではfsdataset

# LEDの点灯状態の推移グラフを作成

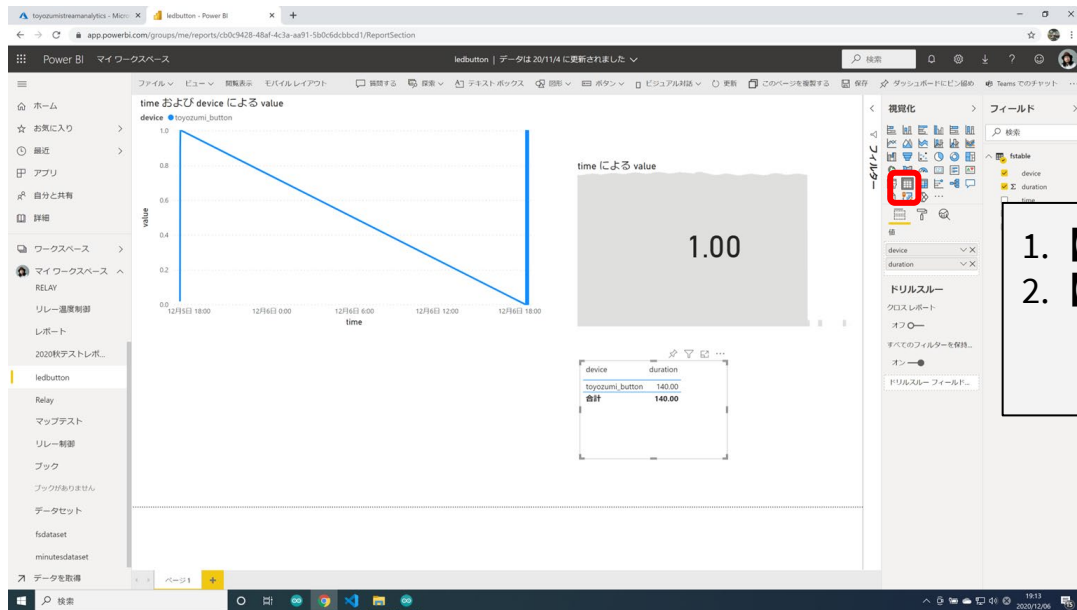


# LEDの現在の状況を把握する



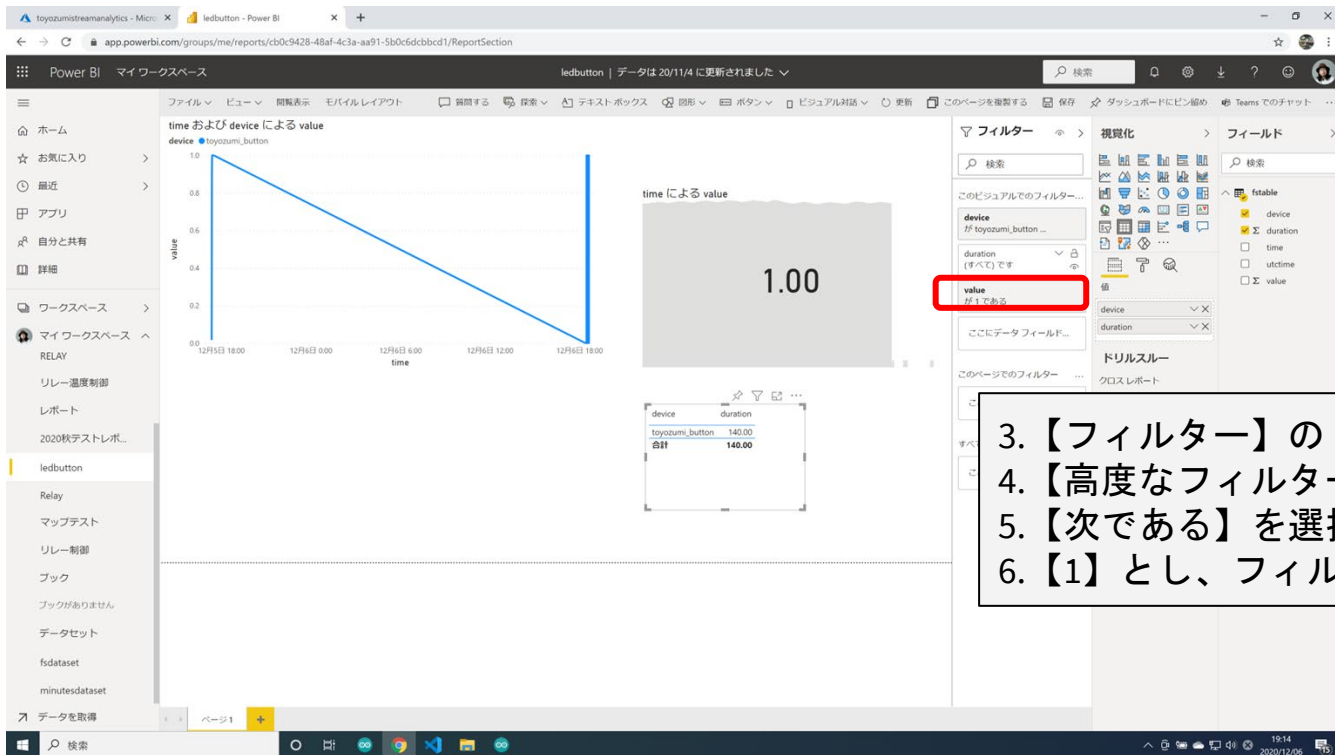
1. 【KPI】をクリック
2. 【インジケータ】にvalue  
【トレンド値】にtime

# LEDの点灯時間を表示する



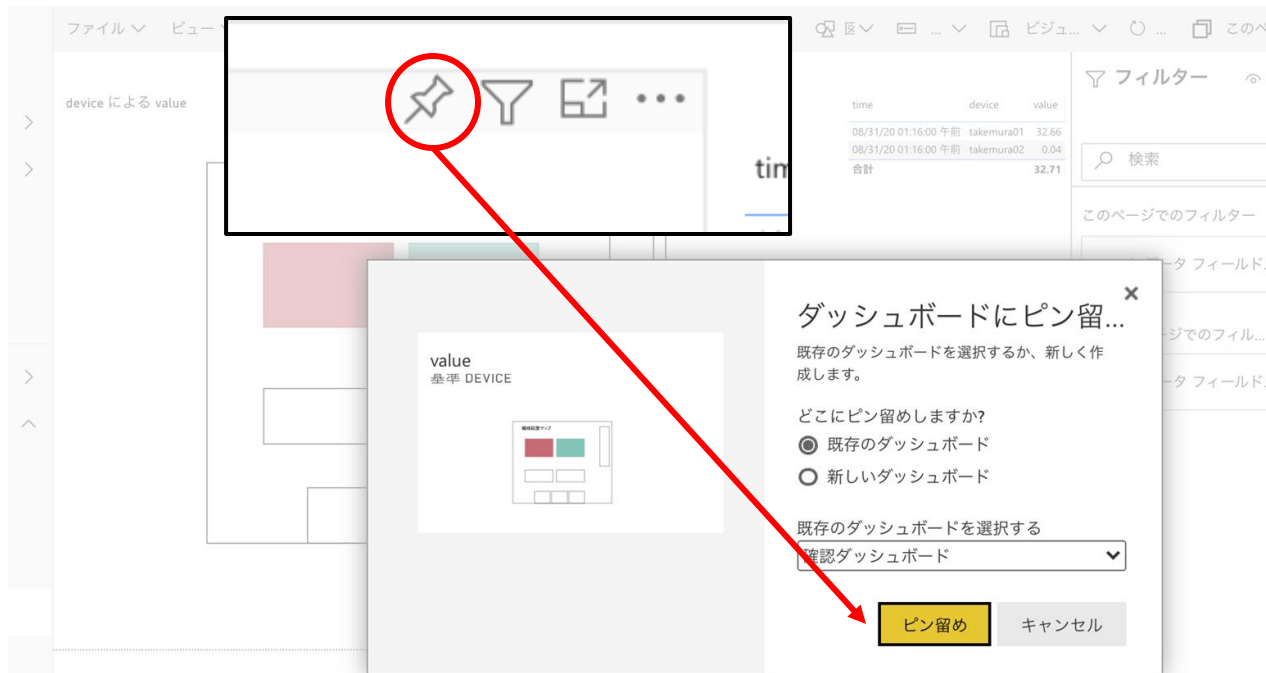
1. 【テーブル】をクリック
2. 【値】にdevice、durationの順番で入れる

# LEDの点灯時間を表示する



3. 【フィルター】の【value】を開く
4. 【高度なフィルター処理】を選択
5. 【次である】を選択
6. 【1】とし、フィルターを適用する

## ダッシュボードにピン留めしてリアルタイムに更新させる



# LEDの状態をダッシュボードに表示

