本研究のUIにおけるjavaのクラス一覧

|  |  |
| --- | --- |
| Connection | Wi-FiやMQTTの接続の設定を保持するクラス |
| Description | Pub,PIf,Sub,SIfのUIを表示した時に同時に出る説明用UIクラス |
| MakeCode | Pub,PIf,Sub,SIfなどで設定したLogをまとめ、一つのArduinoファイルにするクラス |
| mqtt\_ui | メインのUIクラス、解説は後述 |
| PIf | Publishする条件を設定するUIクラス |
| Pub | Publishするセンサー、内容を設定するUIクラス |
| PubConfig | PIf,Pubの選択をするUIクラス |
| SIf | Subscribeする条件を設定するUIクラス |
| Sub | Subscribeするアクチュエータ、動作内容を設定するUIクラス |
| SubConfig | SIf,Subの選択をするUIクラス |
| Test\_Display | Processingを用いた図を表示するUIクラス |
| Test | Test\_Displayで表示するために必要な情報、TopicやDisplayする内容を決めるクラス |
| Topic | Topic名とPIf,Pub,SIf,Subで記述したLogを保持するクラス |

insフォルダにおける文法紹介

本UIにおいて、新しいセンサーやアクチュエータなどを追加した時にinsフォルダに特定の文法で記述されたテキストファイル(.txt)を追加するとUIに反映される。テキストファイルのタイトルの最初の部分に指定された文字を入力しないと反映されないので注意。

|  |  |
| --- | --- |
| PIf | Publishする条件 |
| Pub | Publishする内容 |
| SIf | Subscribeする条件 |
| Sub | Subscribeする内容 |
| Con | Arduinoのボードに関する内容 |

行の一列目はInput,Log,Descriptionの3種類に分類される。列はコロンで区切られる。行の最後にも必ずコロンを入れるように。

|  |  |  |  |
| --- | --- | --- | --- |
| 種類 | 引数1 | 引数2 | 引数3 |
| Input | UIで入力したい内容※1 | なし | なし |
| Log | Arduinoのコード※1 | コードを入れる位置情報※2 | コードを入れる位置情報※3 |
| Description | 説明用のテキスト | なし | なし |

※1　Logの引数1において、円マーク\は特別な意味を示している。Input引数1ではUIにおいて入力する部分であるが、その内容をLogの引数1に反映させる。例えばPub\_test.txt

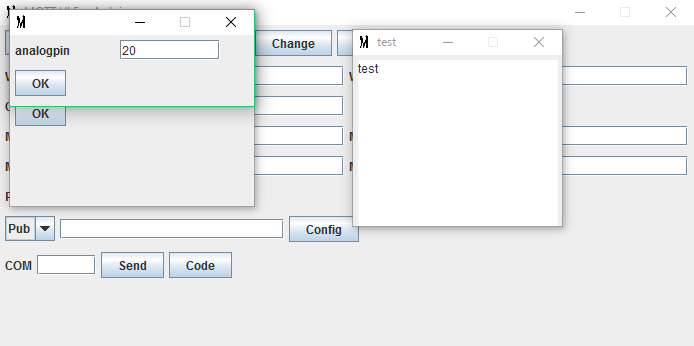
Input:analogpin:

Log:\analogpin\:pub:ins:

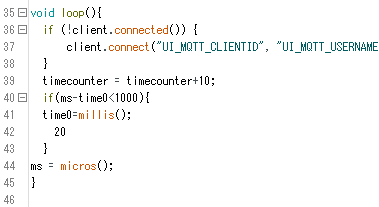
Description:test:

という記述があったとする。

これをinsフォルダに入れ、UIを開き、Configボタンを押すとコンボボックスにtestという選択肢が追加される。testを選択した後、analogpinの中身を以下のように設定する



そして実際に生成されるコードは以下のようになる。



また、上記以外にも特別な文字を用いることによってコードに反映させる変数がある。以下の表が特殊な変数である

|  |  |
| --- | --- |
| \topic\ | mqtt\_uiで入力したTopicを代入する |
| \num\ | Topicの番号を代入する  mqtt\_uiにおいて複数のTopicを使うとき、変数名が重複しないために番号付けを行う |

※2　MakeCodeクラスにinsert関数というのがある。insert関数の引数はLogの引数2で使われる内容を参照し，Logの引数1のコードを入れ込む。

Logの引数2の種類は以下の通り

|  |  |
| --- | --- |
| inc | ライブラリなどをincludeする位置に挿入 |
| Def | defineなどの定数、変数を宣言する位置に挿入 |
| setup1 | setup関数のWi-Fiの接続する前の位置に挿入 |
| setup2 | setup関数のWi-Fiの接続する後の位置に挿入。Subscribeの設定をする時に使用 |
| pub | loop関数に挿入 |
| sub | Subscribeした時に呼ばれるコールバック関数に挿入 |

※3

Log引数2のpubとsub以外は宣言の順番が重要視されないが、pubとsubの位置に関してはIf文の中にセンサーの値をPublishしたり、Subscribeした内容によってActuatorを動かすコードを記述しなくてはいけない。Log引数2ではその設定を行う。start→ins→endの順番でコードが記述される。

|  |  |
| --- | --- |
| start | Log引数1のpub,subにおいて1番目に記述される。主にPIf,SIfで使用する。  if(){ |
| Ins | Log引数2のpub,subにおいて1番目に記述される。主にPub,Subで使用する。  センサーやアクチュエータ独自のプログラミングを挿入する |
| End | Log引数3のpub,subにおいて1番目に記述される。主にPIf,SIfで使用する。  } |
| 空白 | Log引数1のpub,sub以外を入力した時は何も入力しない |

例）

PIf

Input:time:

Log:long time\num\;:def::

Log:if(ms-time\num\<\time\){:pub:start:

Log:time\num\=millis();:pub:ins:

Log:}:pub:end:

Description:Publisher publishes constantly time-delimited.1->1ms,1000->1s:

Pub

Input:analogpin:

Log:char buf[10];:pub:ins:

Log:sprintf(buf,"%d",analogRead(A\analogpin\));:pub:ins:

Log:client.publish(\topic\,buf);:pub:ins:

Description:Grove Moisture is used for judge if there is water around the sensor.:

SIf

Input:From:

Input:To:

Log: if(atof(json)>\From\&&atof(json)<\To\&&!strcmp(topic, "\topic\")){:sub:start:

Log:}:sub:end:

Description:If Subscriber subscribed the topic and value is more than From and value is less than To, Actuator will act.:

Sub

Input:digitalpin:

Input:delay:

Log:pinMode(\digitalpin\,OUTPUT);:setup1:ins:

Log:digitalWrite(\digitalpin\,HIGH);:sub:ins:

Log:delay(\delay\);:sub:ins:

Log:digitalWrite(\digitalpin\,LOW);:sub:ins:

Log:client.loop();:pub::

Description:GroveBuzzer will sound during the delay .:

Conについて

mqtt\_uiクラスのsendボタンでArduinoに直接コードを送るようになっているが、実際はArduinoのボードの情報が必要となる。また、ボードによってWi-Fiに接続するclientの方法なども違うのでConファイルでその内容を記述する。この内容はmqtt\_uiの左上のコンボボックスでボードを選択することで反映させる。

|  |  |
| --- | --- |
| Package | パッケージ名※1 |
| Arch | アーキテクチャ名※1 |
| Board | ボード名※1 |
| Log | Arduinoのコード※2 |

※1(Windows版)

EsprDeveloperやEdisonなどのArduinoの派生種はArduinoのIDEのボード一覧にはなく、インストールしなければいけない。この時のPackage,Arch,Boardの確認方法は以下の通りである。

Package：

ユーザー名\AppData(隠しフォルダ)\AppData\Local\Arduino15\packages

に記述されている該当するフォルダ名

Arch:

ユーザー名\AppData(隠しフォルダ)\AppData\Local\Arduino15\packages\パッケージ名\hardware

に記述されているフォルダ名

Board:

ユーザー名\AppData(隠しフォルダ)\AppData\Local\Arduino15\packages\パッケージ名\hardware\Arch名\バージョン\board.txt

board.txtはボードの内容を記述しているが、

##############################################################

の区切りの下に〇〇.name = ××

と記述されている。

××はArduinoのIDEでボードを選択する時に出てくる文字であり、Boardで入力するのは〇〇の部分となる。

また、Arduinoに標準で選択できるボードは

Package:arduino

Arch:ave

Board:

ProgramFiles(x86)\Arduino\hardware\arduino\avr\board.txt

board.txtはボードの内容を記述しているが、

##############################################################

の区切りの下に〇〇.name = ××

と記述されている。

××はArduinoのIDEでボードを選択する時に出てくる文字であり、Boardで入力するのは〇〇の部分となる。

※2　先述したLogと同等の内容となっている。ボードの設定や接続の設定を記述している。

ボードによってはClientの宣言の仕方やWi-Fiの接続が異なるのでこのLogでボードに依存するコード部分を記述する。