

PS3

PS3コントローラーを使うためのクラス

struct PS3_data

PS3コントローラの受信状態を格納するための構造体

ボタンは押されているときに1、押されていないときに0を返す

スティックは、右をX軸の正、上をY軸の正として64~-64の値を返す

[要素内容]

↑ボタン

↓ボタン

→ボタン

←ボタン

△ボタン

×ボタン

○ボタン

□ボタン

L1ボタン

L2ボタン

R1ボタン

R2ボタン

Startボタン

Selectボタン

ジョイスティック（左x軸）

ジョイスティック（左y軸）

ジョイスティック（右x軸）

ジョイスティック（右y軸）

受信データ

[要素一覧]

```
struct PS3_data{
    Up,Down,Right,Left,Triangle,Cross,Circle,Square,L1,L2,R1,R2,Start,Select;
    int LxPad,LyPad,RxPad,RyPad;
    uint8_t rx_data[8];
};
```

class PS3

```
void PS3::StartRecive(Pin tx_pin = A9,Pin rx_pin = A10,UartNumber uart_num = SERIAL1);
```

PS3コントローラからの受信を開始する

[パラメータ]

送信ピン番号

受信ピン番号

シリアル番号

(デフォルトで、部室にあるシールド基盤のピン番号、シリアル番号を設定している)

[戻り値]

なし

[サンプルコード]

ps3コントローラからの受信を開始

```
#include "stm32f4xx.h"
#include "stm32f4xx_nucleo.h"
#include "sken_library/include.h"

PS3 ps3;

int main(void){
    sken_system.init();

    ps3.StartRecive();

    while(true){

    }
}
```

bool Getdata(PS3_data* ps3_data);

PS3コントローラのデータを取得する

[パラメータ]

PS3_deta構造体のアドレス

[戻り値]

受信データを取得できればtrue、できなければfalse

[サンプルコード]

ps3_dataにPS3コントローラのデータを代入

```
#include "stm32f4xx.h"
#include "stm32f4xx_nucleo.h"
#include "sken_library/include.h"

PS3 ps3;
PS3_data ps3_data;

int main(void){
    sken_system.init();

    ps3.StartRecive();

    while(true){
        ps3.Getdata(&ps3_data);
    }
}
```

```
}  
}
```