

uart

UART通信を制御するクラス

ポーリングモードでの送受信とDMA(Direct Memory Access)を使用した受信を行うことができる

class Uart

**void Uart::init(Pin tx_pin,Pin rx_pin,UartNumber
uart_num,int baudrate,uint32_t parity =
UART_PARITY_NONE)**

シリアル通信の初期化を行う関数

[パラメータ]

送信ピン番号

受信ピン番号

シリアル番号

転送レート. 単位は[bps]

パリティ.デフォルトではなし

[戻り値]

なし

[サンプルコード]

UART1をボーレート9600で設定する

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

Uart serial;

int main(void)
{
    sken_system.init();
    serial.init(A9,A10,SERIAL1,9600);
    while(1)
    {

    }
}
```

**void Uart::startDmaRead(uint8_t* data_p,int
data_size)**

データ受信をDMAで行うための設定を行う関数. 配列の先頭アドレスと要素数を指定することでリング状にデータを格納する.

[パラメータ]

格納先配列の先頭アドレス

格納先配列の大きさ

[戻り値]

なし

[サンプルコード]

2バイトのデータをDMAで受信する

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

Uart serial;

uint8_t data[2];

int main(void)
{
    sken_system.init();
    serial.init(A9,A10,SERIAL1,9600);
    serial.startDmaRead(data,2);
    while(1)
    {

    }
}
```

uint8_t Uart::read(unsigned int dead_time)

ポーリングモードでデータを受信する関数。デッドタイム以上の時間、データが受信できなかった場合タイムアウトフラグがtrueになる。

[パラメータ]

受信待ち時間

[戻り値]

受信データ

[サンプルコード]

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

Uart serial;

uint8_t data;

int main(void)
{
    sken_system.init();
    serial.init(A9,A10,SERIAL1,9600);
    while(1)
    {
        data = serial.read(1000);
    }
}
```

void Uart::write(uint8_t* data_p,int data_size,int dead_time = 100)

ポーリングモードでデータを送信する関数.

[パラメータ]

送信データ配列の先頭アドレス

送信データ配列の大きさ

デッドタイム. デフォルトでは100[ms]

[戻り値]

なし

[サンプルコード]

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

Uart serial;

uint8_t data[2] = {1,2};

int main(void)
{
    sken_system.init();
    serial.init(A9,A10,SERIAL1,9600);
    while(1)
    {
        serial.write(data,2);
    }
}
```

bool Uart::checkTimeOut(void)

ポーリングモードでデータを受信した際の通信切断を確認できる

[パラメータ]

なし

[戻り値]

タイムアウトフラグ

[サンプルコード]

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

Uart serial;

uint8_t data;

int main(void)
{
    sken_system.init();
    serial.init(A9,A10,SERIAL1,9600);
    while(1)
    {
        data = serial.read(1000);
        if(serial.checkTimeOut){
```

```
    }  
    else{  
        }  
    }  
}
```