

1113

<今まで>

1文字ずつ送信していた。センサデータは送信していなかった。

<今回>

まとめて送信する。そして、センサデータを追加する。

AA+X+2.12345

AA：最初を表す文字

X：どのセンサの値なのかを表すコマンド

2.12345：センサデータ（適当な値）

【python】

5秒間センサデータを取得し、5秒間の平均を送信する。その際、AAとXを付けて送信する。

```
KYOHEI@KYOHEI:~/Desktop $ python al.py
受信データ：1042.0
電圧：2.084
Traceback (most recent call last):
  File "/home/KYOHEI/Desktop/al.py", line 106, in <module>
    SendSensor(avg_volt)
  File "/home/KYOHEI/Desktop/al.py", line 69, in SendSensor
    uint32ToBytes(dataBytes, data_sensor)
  File "/home/KYOHEI/Desktop/al.py", line 49, in uint32ToBytes
    bytes = decimalValue.to_bytes(4, byteorder="little")
AttributeError: 'float' object has no attribute 'to_bytes'
```

上記の3つの関数でエラーが生じている。センサ値を送信配列に入れてまとめる処理の部分でエラーが起こっている。

入力する型と想定している型が異なるためだと考えている。

【NXP】

Python（ラズパイ）からのデータを受信する。

↓

最初を表すAAとコマンドのXはそれぞれ読み取る。

↓

その次のセンサデータをバッファに格納する。

↓

センサデータを他のMONOSTICKに送信する処理を実行する。

↓

送信

そのデータを受信した他の MONOSTICK 側での処理はまだ構築していない。

```
//ラズパイから受信した 0xAA + X + センサデータ を処理する

uint8_t rxByte = u8AHI_UartReadData (DBG_E_UART_0);

int state;
state = START;

switch(state){
case START:
    if(rxByte == 0xAA)
    {
        state = PRIMITIVE_TYPE;
        DBG_vPrintf(TRUE, "In start state\n");
    }
    break;
case PRIMITIVE_TYPE:
    DBG_vPrintf(TRUE, "In primitive type state \n");
    mlmeReqRsp.u8Type = rxByte;
    SetPrimitiveRxSize(rxByte);
    state = WAIT_DATA;
    break;
case WAIT_DATA:
    tmpString[indexCount] = rxByte;
    indexCount++;
    if(indexCount == primitiveMaxSize){
        processData();
        state = START;
        indexCount = 0;
        memset(tmpString, 0, sizeof(tmpString));
    }
    break;
}
```

アルベルト先生の AA とコマンドを処理する部分を参考にしたが、
AA とコマンドを削除すべきか活用すべきか、どう処理するか検討中である。

- ・ mlmeReqRsp.u8Type = rxByte
- ・ SetPrimitiveRxSize(rxByte)

AA は削除

X はコマンド用配列に入れて、各コマンドに合わせた処理を行う。

センサデータはセンサデータ用配列に入れて、この配列を他の MONOSTICK に送信する。

```
memset(command, 0, sizeof(command));
strncpy(command, ComString, 64);
memset(ComString, 0, sizeof(ComString));
```

command を 0 で埋める。

コマンド X が入っている ComString 配列を command にコピーする。

ComString を 0 に埋める。

今後の日程

11/13～11/20 python → MONOSTICK①

11/20～12/04 MONOSTICK① → MONOSTICK②