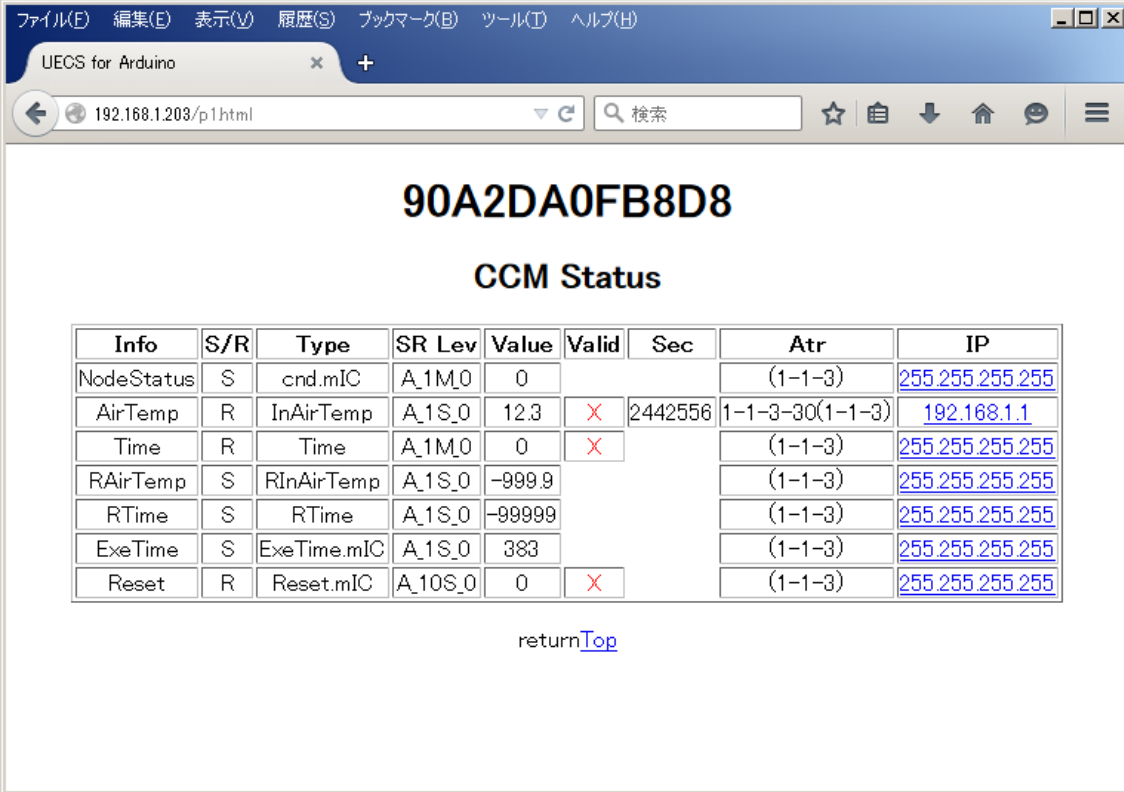


黒崎です。

今まで送信側のノードばかり作っていたので気づかなかったのですが、受信ノードを作るとき **Validity** の値が変になるので困っております。症状は例えば **A_1S_0** のノードに 1 秒間隔でデータを送ると、たまに **Sec** の値が異常に大きくなって本来 **true** のはずの **Validity** が **false** になります。

自分のノードの問題なのか、他でも起こるのかちょっと切り分けがまだできていない段階ですが、一応報告しておきます。



90A2DA0FB8D8

CCM Status

Info	S/R	Type	SR Lev	Value	Valid	Sec	Atr	IP
NodeStatus	S	cnd.mIC	A_1M_0	0			(1-1-3)	255.255.255.255
AirTemp	R	InAirTemp	A_1S_0	12.3	X	2442556	1-1-3-30(1-1-3)	192.168.1.1
Time	R	Time	A_1M_0	0	X		(1-1-3)	255.255.255.255
RAirTemp	S	RInAirTemp	A_1S_0	-999.9			(1-1-3)	255.255.255.255
RTime	S	RTime	A_1S_0	-999999			(1-1-3)	255.255.255.255
ExeTime	S	ExeTime.mIC	A_1S_0	383			(1-1-3)	255.255.255.255
Reset	R	Reset.mIC	A_10S_0	0	X		(1-1-3)	255.255.255.255

[returnTop](#)

黒崎です

このバグを発生させる条件が大体わかりました。

受信用 CCM が 1 つだけの場合、このバグは発生しません。

受信用 CCM を複数登録すると 3 秒に 1 回ぐらいの頻度で発生します。

On 2015/03/02 20:13, Yasuba Ken-ichiro wrote:

> 黒崎様

>

> 安場です.

> レポートありがとうございます.

- > なんとなく配列などのデータ書き込みで、受信時間のメモリー領域を書き込んで
- > しまって、たまにエラーになるようなパターンのような気がするのですが、ライ
- > ブラリーを精査してみないとわかりません。(今まであまり気になったことが
- > 無かったので)
- > もし可能でしたら、使用する文字列の長さを全体的に短くして試してもらえます
- > でしょうか？

安場様

デバッグコードを入れてみてわかったことですが、
p1.htmlを見ながら F5 を連打して短時間に何度も更新すると
HTMLを送信する瞬間に 1/5 ぐらいの確率で発生するようです。
受信時間のデータが破壊されているようです。

逆に HTML を送信していない場合異常動作はしないようです。
受信 CCM が複数あると発生するというのは純粋にメモリー上の配置場所の問題で、
時間の計算部分のバグというより、実際には HTML を生成するときに生じる
メモリーリークが悪さをしている可能性のほうが高いです。

なお、時間データが破壊された時、大体同じ数値

`_time=1852794415`

になることに気づきましたので、これを復号してみました。

6E n

6F o

66 f

2F /

逆さまになりますので文字列にすると"/fon"です。font タグの一部分？

やはり html の生成に使われたメモリーがはみ出してきているようです。

こちらでもさらに追跡してみます。

ただ、html の生成部分はかなりの難所なので解読には時間がかかるかもしれません。

黒崎様

安場です。

html のタグは ROM から直接読みだすようにしているので、例の PROGMEM をうまく使ってやる必要があります、

ARDUINO のもっとめんどくさいところだと思います。たぶん font タグの文字列を RAM に転送するところで文字数を指定しているのですが、 その文字数に誤りがあるのだと思います。

font タグ関係のバグは依然修正したような記憶がかすかにあるのですが、まだ問題が残っていたようです。

Serial.print を使うと、同じような問題が発生した記憶もあります。

出張中で状況がよくわからずすいません。

黒崎様

安場です。さきほどソースを見てみたのですが

EthernetManeger.h ファイルの 95 行目くらいに

```
const char PROGMEM UECSpeke[] = "<font color=¥"red¥">X</font>";
```

とあるのを

```
const char PROGMEM UECSpeke[] = "X";
```

に修正してもバグは解消しませんでしょうか？

黒崎です。

例のメモリーリークらしきバグは安場さんの修正でとりあえず出なくなりましたのでしばらく様子を見えます。

黒崎様

安場です。

バグの件、確認ありがとうございます。メモリーリークというより私の記述ミスでした。Arduino は文字列の取り扱いが面倒ですので、20 文字以上と以下で取扱い方法を変えていたのですが、該当箇所は取扱いにミスがありました。すいません。

Arduino 用のライブラリでノード名を付けるとき、20 文字以上入力すると末端にゴミが付きます。

原因は 20 文字ぴったりになると末端の null が EPROM に保存されないのが原因かと思いますが、

応急処置として以下の関数の中で

```
void UECScheckHttpRequest(char* _buffer, char* _tempStr20, UECSCCM
```

```
_ccmList[], UECSOriginalAttribute* _orgAttribute, UECSTEMPCCM* _tempCCM)
```

以下の場所を

```
for(int i = 0 ; i < 20; i++){
    if(_buffer[pointer] != '&'){
        U_nodename[i] = _buffer[pointer];
    }else{
        if(_buffer[pointer+1] == 'S'){
            U_nodename[i] = 0;
            break;
        }
    }
}
```

↓

```
for(int i = 0 ; i < 19; i++){
    if(_buffer[pointer] != '&'){
        U_nodename[i] = _buffer[pointer];
    }else{
        if(_buffer[pointer+1] == 'S'){
            U_nodename[i] = 0;
            break;
        }
    }
}
```

に変更して 19 文字まで受け入れるようにすれば何とかなりそうです。

しかし、Arduino のようなメモリの少ないマイコンで可変長の html を処理するのは負担が大きいですね。

黒崎様

安場です。

確かに¥n があるので 19 文字まででした。

Arduino は ROM の使い方に癖があり、今回メモリーをかなり節約しなければ、ライブラリーとして使いものにならないので、html はかなりプログラム上の難所（集中力が必要という意味で）です。細かいバグは色々とおもいますが、手加減して使ってみてください。

