

MICS で SH プログラミング

MICS 上の SH で動作させるサンプルプログラムとして、
とりあえずビットマップ画像のローダーでも書いてみようかと思った。
これに伴い、まずビットマップが表示できるデバイスを MICS に追加。
で、普通に C でプログラムを書いて走らせてみると、
アクセスしないはずのメモリエリアにがんがんアクセスする ...
で、原因

1. リトルエンディアンとビッグエンディアンの違いによる値の違い
- 2.bmp フォーマットの解析に使っていた構造体で align されて予期しないアドレスの値を使用

1 番目の方は、やれやれと思いながら、ntoh みたいな関数を用意。解決。

2 番目の方にはなかなか気づかなかった。

そして、MICS でプログラムを走らせる場合のデバッグの難しさを痛感。

これどうにか解決しないとなあ。

ちなみに、解決方法は、gcc の場合、構造体に attribute つけて

```
struct bitMapFileHeader{
    unsigned short bfType __attribute__((packed));
    unsigned int bfSize __attribute__((packed));
    unsigned short bfReserved1 __attribute__((packed));
    unsigned short bfReserved2 __attribute__((packed));
    unsigned int offBits __attribute__((packed));
};
```

やれやれ。