≫ 文法の曖昧さを理解して確実性と再利用性を高める

マイコンC言語 転ばぬ先のつえ

第20回　静的変数領域の削減③…静的変数のコンパイル結果を見てみる

鹿取 祐二

表1　C言語の変数の分類…静的変数と動的変数（再掲）
局所変数と大域変数とは異なり，記憶クラスと密接に関わっている

宣言場所 / 記憶クラス	局所変数（関数，複文の先頭）	大域変数（関数外部）
なし	動的変数①	静的変数②
auto	動的変数⑤	文法違反
static	静的変数③	静的変数④
register	動的変数⑥	文法違反

リスト1　静的変数を記述したプログラム

```
int a;
int b=1;          ← 表1の②

static int m;
static int n=2;   ← 表1の④

void func(void)
{
static int x;
static int y=3;   ← 表1の③
}
```

（a）ソースコード

```
シンボル名
_a:                 .PUBLIC  _a
_m@1:               .PUBLIC  _b
_x@3@func:          .SECTION .bss,BSS
                    .DS      (2)
_b:                 .DS      (2)
_n@2:               .DS      (2)
_y@4@func:          .SECTION .data,DATA
                    .DB2     0x0001
                    .DB2     0x0002
                    .DB2     0x0003
```

（b）RL78の場合のコンパイル結果

```
シンボル名
_a:                 .glb     _a
__$m:               .glb     _b
__$x$1:             .SECTION B,DATA,ALIGN=4
                    .blkl    1
_b:                 .blkl    1
__$n:               .blkl    1
__$y$2:             .SECTION D,ROMDATA,ALIGN=4
                    .lword   00000001H
                    .lword   00000002H
                    .lword   00000003H
```

（c）RXの場合のコンパイル結果

本連載では，C言語の言語仕様（文法）の曖昧な部分と，それにより起こる問題を解説します．再利用性と効率が高く，安全かつ安心して使えるソフトウェアが開発できるようになることを目指します．

本誌2022年12月号から，静的変数領域の削減について紹介しています．「静的変数」の一言で，どの変数であるかが理解できていればよいのですが，宣言場所や記憶クラスも絡んでおり，多少複雑です．そこで，今後数回にわたり，静的変数の文法的な説明も含めて，その削減方法を紹介します．（編集部）

初期値の有無で変わる静的変数の取り扱い

今回は，静的変数の領域削減の予備知識として，静的変数のコンパイル結果について解説します．静的変数のコンパイル結果はCPUの実行命令ではなく，アセンブラ制御命令です．従って，コンパイル結果（制御命令）の表記はCPUや処理系によって異なりますが，やっていることは同じです．ただし，初期値の有無で制御命令が変わります．

今回は静的変数のコンパイル結果の特徴を，RL78とRX（ルネサス エレクトロニクス）を例に紹介し，静的変数の初期値の有無で異なる取り扱いが必要なことを説明します．細かい点まで覚える必要はありませんから，その特徴を把握してください．

■ 異なる処理系でもやってることは同じ

第18回，第19回（本誌2022年12月号，2023年1月号）から引き続き，C言語の変数の分類を**表1**に示します．

● RL78の場合

表1の②，③，④の静的変数を記述したプログラムを**リスト1**（a）に示し，これをRL78でコンパイルした結果を**リスト1**（b）に示します．

▶特徴1…シンボル名は変数名が基本

リスト1（b）で，シンボル_aが変数a，_m@1が変数m，_x@3@funcが変数xを表します．また，_b

第11回　演算子②…[ ]と*の意味，++aとa++の違い（2022年1月号）
第12回　演算子③…実は本当に必要なケースは多くない！キャスト演算子（2022年2月号）
第13回　演算子④…算術演算子の正しい使い方（2022年3月号）