

≫ 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29回 動的スタック領域の削減①…ネスト・レベルを減らす

鹿取 祐二

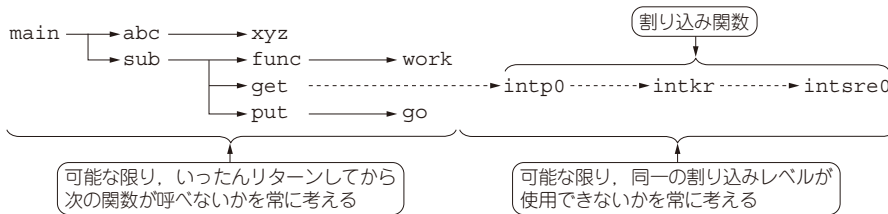


図1 関数呼び出しのネスト・レベルが深くなるほどスタック使用量も増大する

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。

(編集部)

今回からは、動的領域であるスタック領域の削減について紹介します。スタックの使用量は、プログラミングの経験値がはっきりと出る項目です。素人だとスタック領域は肥大化し、玄人だとスタック領域はそれほど大きくなりません。

素人と言われないためにも、動的領域であるスタック領域の削減方法を理解しましょう。

47 スタック領域を減らすなら…

ネスト・レベルを減らすのが最優先!

● 方法①…関数呼び出しのネスト・レベルを減らす

図1に示す通り、スタック使用量は関数の呼び出しネスト・レベルが深くなると増大します。削減するためには、ネスト・レベルを減らすのが1番です。

例えば、必要以上に関数を細分化して、スタック使用量のことを考えず、ネスト・レベルの深いプログラムを設計する人が居たとします。それは、関数の再利用性の観点では良いかもしれませんが、スタック使用量の観点だと誤りです。動的領域の削減を優先するのであれば、関数は常にいったん戻してから次の関数が呼ばないかを考えるべきです。関数呼び出しのネ

スト・レベルが深くなればなるほどスタック使用量が増加すると考えましょう。

● 方法②…割り込みのネスト・レベルを減らす

最近では、多重割り込みのレベル数が15程度あるマイコンが多くなってきました。本連載で取り上げているルネサス エレクトロニクスのマイコンでは、SHとRXが15段階の多重割り込みが可能です。割り込みレベル数が豊富ですが、これも関数の呼び出しネスト・レベルと同様に使用する割り込みレベル数を減らすことでスタック使用量を削減できます。

▶ 多重割り込みを使うとスタック使用量が増大する

15段階あるからといって、15段階全てを使うと、図1に示す通り、使用した割り込みレベルは、その中で最もスタック領域を消費する割り込み関数を全て加算しなければなりません。図1の場合はわずか3レベルですが、これが15レベルとなったときを想像してみてください。図1の4倍程度のスタック使用量となることは容易に想像できます。可能な限り同一の割り込みレベルで処理できないかを常に考えるべきです。

▶ 同一割り込みレベルでは遅延を考慮しよう

もちろん、同一の割り込みレベルを使った場合、考慮しなければならないのは遅延です。同一である他の割り込みは受け付けられません。今実行中の割り込みが終了しなければ受け付けられないので、その間は処理の遅延が発生することになります。その遅延時間がシステムに悪影響を与えないか検討する必要があります。