教科書には載っていない！

現場で役立つ プログラミングのちょい技

第9回　CPU演算量や配列メモリ容量を節約するちょい技

邑中 雅樹

前回は，プリプロセッサ・マクロと使ってデバッグしやすくするちょい技を紹介しました．今回はプリプロセッサ・マクロやインライン関数を使って，CPU演算量や配列で使うメモリ容量を節約するちょい技を紹介します．

（編集部）

組み込みシステムと呼ばれる携帯電話，テレビ，洗濯機，自動車，エレベータなどの製品は，その規模の大小に関わらず，使用できるメモリ容量が少ないものです．GHzで動作するデュアルコアCPUに膨大なメモリを搭載したAndroidの世界や，本誌付属基板でさえ富豪的であるとだと思えるような例外は一定の規模で存在しています．同時に8ビットCPUに少ないメモリ容量でのシステムも多く存在しています．

特に極小規模の場合には，まさに爪に火を灯すようなメモリ容量の削減が行われます（**図1**）．まず，データベースやファイル・システムのようなミドルウェアは真っ先に削られ，さらに制約がある場合は，RTOS（Real-Time Operating System）の採用も見送られます．同時に，このような極小規模のシステムでは命令の処理能力も貧弱なので，実行時の計算量を減らすことが求められます．

図1　必要最低限だけにしぼり込み，動作時の負荷を減らす

CPUの計算量を減らすテクニック

具体例として，自動車の制御ユニット（ECU）の場合を考えてみましょう．例えば，エンジンの回転数と負荷により燃料の噴射量を制御する場合，理想的な条件であれば，（燃料の噴射量）== f（エンジンの回転数，負荷）は事前に計算でき，誤差は実際のエンジンを回しながら調整できます．生産時の個体誤差は自動車に組み込むときにフィードバック学習をさせる必要がありますが，基礎となる数値は数表として用意できます．

実際，ECUでは事前に制御用の数値を計算したテーブルをROMに持たせて，CPUの計算量を削減しています．

ちょい技1　事前にできる計算は配列としてROMに持たせる

実行前に行える処理は済ませて実行時の計算負荷を減らすという考え方は，前号で示したプリプロセッサ・マクロを駆使することと同じです．

配列サイズを削減するテクニック

● ほぼすべてのcharは8ビット

上記のようなテーブル・データを扱う時に，C言語では**リスト1**（a）のように配列を使用するのが王道でしょう．しかし，組み込みソフトウェア技術者なら，ここで満足してはいけません．この配列のサイズは半分にできます．

C言語の配列に使えるデータ型の最小サイズは`char`です．ほぼ全てと言って良いくらいの処理系では，`char`のサイズは8ビットです．