



## 第2章 C言語でプログラミングするのが 当たり前の時代

# 進化するPICの開発環境

PICマイコンを動かすためには、プログラムを書き込まなければなりません。最近ではC言語で書いたプログラムを機械語に直してくれるCコンパイラが無償で提供されるようになり、プログラムをC言語で開発する人が増えました。ここではPICマイコンの開発環境、Cコンパイラ、書き込み器、評価基板の種類と選び方を紹介します。

### C言語が当たり前

#### ● すべてのPICマイコンでCコンパイラを提供できる

PICマイコンの開発の主流はC言語へと移りつつあります。2009年3月にPICマイコン用のCコンパイラで定評のあるHI-TECHソフトウェア社を買収しました。これにより、もちあわせていなかった小規模PIC(10F/12F/16Fシリーズ)向けのCコンパイラを手に入れました。マイクロチップ社は小規模PICから大規模PICまで全PICマイコンに対して、Cコンパイラを提供しています。

#### ● アセンブリ言語はほかのPICへの移植性が悪い

小規模向けのPIC10/12/16シリーズではプログラム・メモリ容量の制約があるため、アセンブリ言語を用いての開発が主流でした。アセンブリ言語であれば、個人のスキルで最適化でき、少ないプログラム・メモリに工夫次第で収めることができました。

しかしアセンブリ言語での開発は初心者には手を付けにくいものでした。命令を一つずつ理解しておく必要があったからです。命令体系に依存したものとなるため、移植性が悪いという欠点もありました。

PIC10/12/16シリーズでは12ビット命令体系(33命令)と14ビット命令体系(35命令)が混在しており、似たような命令体系ですが、互換性はありません。

筆者もかつてアセンブリ言語で開発したソースの移植を断念した経験があります。PICにはページという概念があり、ページをまたぐジャンプ命令、コール命令をすべて書き換えなければならず、テスト工数も膨大になりました。

#### ● C言語は移植性に優れる

プログラムを徐々に改良、拡張していくと、プログラム・メモリの容量制限やピン数制限などを始め

ます。高機能の上位PICシリーズを利用したくなりますが、移植性が大きな問題です。

そこでC言語の登場です。もともとUNIXというOSをアーキテクチャの異なるCPU上で移植しやすくなるように考えられました。

もちろん移植を100%完全に保証するものではありませんが、最小限の変更で移植が可能です。少なくともアセンブリ言語のように、移植のためにプログラム全体を書き換えなければならないといった手間を省くことができます。

つまり、はじめからC言語で記述しておけば、移植性に優れたプログラムとなります。仮に現在利用しているPICが生産中止になったとしても、最小限のプログラム修正で同等あるいは上位のPICへ簡単に移行できます。

例えばPIC16シリーズで性能不足や機能不足を感じた場合、PIC24シリーズへ移行しても良いでしょう。C言語ではPIC特有のページという概念を意識する必要もありません。そうしたことはすべてCコンパイラが解決してくれます。

#### ● 生成されるコード数もアセンブリ言語に近くなった

入門用プログラム言語としてC言語が利用されることが多く、初心者でもはじめられやすいでしょう。C言語はANSI規格で統一されており、独特のルールが少ない言語です。

一方で生成されるコード・サイズはCコンパイラの性能に左右されるため、一般的にアセンブリ言語での記述よりもサイズが大きくなる傾向があります。しかしCコンパイラの最適化技術向上により、以前よりも改善されています。

命令実行速度もCコンパイラ任せとなり、アセンブリ言語のような細かい実行命令数制御はできません。しかし、C言語中にアセンブリ言語を埋め込むインライン・アセンブル機能を搭載しているため、必要に応