第3章 機械語，アセンブリ言語，コンパイラ，C言語，デバッガ

コンパイルとデバッグに必須の環境を中心に見る プログラミング言語と開発環境の重要キーワード

組み込み機器の開発にソフトウェアは欠かせない．パソコン向けのプログラムであれば，ソース・コードを書いてビルドすれば実行可能なプログラムを作成できたが，組み込み機器開発においては，x86以外のCPUで動く実行プログラムを作成したり，実機でデバッグを行う必要があるなど，さまざまな知識が必要になる．

そのような環境でプログラムを作成するために必要なプログラミング言語と開発環境に関する重要キーワードを見ていこう．（編集部）

邑中 雅樹

1. ソフトウェアの記述手法

コンピュータの世界に足を踏み入れるときに最初に読む本には，大抵「ディジタル・コンピュータは0と1しか解釈できず，それを人間が理解するのは至難の業であり，云々」というくだりがあります．実際のところ，それらの本のとおりです．

人間の脳の働きとコンピュータの働きは，似たような仕事はできるものの，構造が違います．そのため，コンピュータと人間とを比べると，細かいところでさまざまな得手不得手があります．このことは，漠然としながらも多くの人々が一般教養として身に付けているでしょう．

非常に限られた計算機資源しか存在しないとするならば，エンジニアは機械に近いプログラミングを行う必要があります．計算機は「空気を読む」ことができませんから，空気を読める人間（エンジニア）が頑張る以外にお互いの距離を近づける手段がありません．そのような壁を超えて人間とコンピュータの橋渡しをすることに，コンピュータ・エンジニアの存在意義があるのかもしれません．

コンピュータ・エンジニア達は，二つのアプローチで，コンピュータを人間にとって身近なものに近づけました．一つは，ソフトウェアの記述手法を改良して，なるべく多くの人々がソフトウェアを作成できるようにしたことです．もう一つは，コンピュータの処理速度や効率の向上です．前者は主にソフトウェアの進歩によるものですが，実現にはハードウェアの性能向上が不可欠でした．後者は主に，ハードウェアの性能向上により実現されましたが，ソフトウェアで実現されたものもあります．

このように，ハードウェアとソフトウェアは切り離して考えることが難しいのですが，本稿では，ソフトウェアに焦点を当てて話を進めていきます．

2. 機械語とアセンブラ

keyword 命令，命令セット，機械語，バイナリ，ニーモニック，疑似命令，アセンブリ言語，アセンブラ，ハンド・アセンブル，逆アセンブル，逆アセンブラ，実行可能オブジェクト，バイナリ・パッチ

● 機械語――コンピュータの黎明期からある言語

現存するコンピュータが行っている処理を突き詰めていくと，メモリにある数値を読み込んでそれに対応する計算を行い，結果をメモリに記録する，という単純な機構の繰り返しに行き当たります．ここで注目したいのは，計算対象となる数値だけでなく，どのような計算をするのかといった指示もまた，番号（数値）で表されるということです．

CPUの設計者は，ソフトウェアの処理を支援するためには，どのような計算処理をハードウェアで提供すればよいのかということを考え，実装し，数値表に割り当てます．おのおのの計算処理のことを**命令**，数値表のことを**命令セット**，これらの概念は総称して**機械語**と呼ばれます．

機械語と機械語が命令処理のために用いる数値がひとまとまりになったものを**バイナリ**（Binary）と呼ぶことがあります．英語でのBinaryの本来の意味は，2進数です．機械語の設計において2進数が重要な概念であることがうかがえます．