

## 基板を使って基礎から分かる! ARM組み込み機器開発の手順

待谷 君次

今月号は毎年恒例の付属基板企画である。今年は富士通の最新マイコンFM3が搭載された基板が付属する。この基板を使う前に、組み込み機器開発とは何か?を見ていこう。

(編集部)

### 1. 組み込み機器の特徴

#### ● ありとあらゆるものが組み込み機器

現代のほとんどの電気製品にはCPUが内蔵されています。製品に組み込まれたCPUは、センサやスイッチから情報を取得して演算し、結果をモータやLED、液晶画面やネットワークへ出力します。例えば炊飯器であれば、スイッチが押されたらヒータの電源を入れ、お湯が沸騰したらヒータを弱め、指定時間が経過したら蒸らしモードへと移行し、しばらくしたら炊きあがりの音を鳴らすでしょう。このような一連の動作を行うためにはCPUのプログラムによる制御がうってつけです。炊飯器や洗濯機などのいわゆる白物家電から、ハード・ディスク・レコーダやBlu-rayレコーダ、地デジ対応テレビ、携帯電話などの家電から、車載機器や航空機、工場のライン制御からプラント制御まで、ありとあらゆるものが組み込み機器に分類されます。

ここで図1のように、組み込み機器向けのCPUを規模によって三つに分けてみました。

#### ● メモリ32バイト(!)の4~8ビットCPU

一番小さいのは4~8ビットCPUクラスの製品です。例えばリモコンなどのように超低コストが要求され、数10円

のCPUと32バイト程度のRAM(32Kバイトでなく!)と256バイトのROMで作るような世界です。プログラム言語はアセンブリで、C言語を使うことすら厳しい世界です。

#### ● 主流の1チップ32ビットCPU

##### ：今回のFM3マイコン

その上はいきなり32ビットCPUです<sup>注1</sup>。1チップ型の32ビットCPUは、「CPU本体」であるCPUコアとROM、RAMといったメモリを全て内蔵して1チップにしたものです。1チップにすることにより低コストになり、基板面積が小さくて済むために小型化が可能で、全体として低消費電力になります。数百Kバイト~1Mバイト程度のROMと数十Kバイト~数百Kバイト程度のRAMが内蔵され、μITRONのような組み込み向けOSが動作するようになります。このレベルであれば、C言語と統合開発環境が整備され、いわゆる「組み込みプログラミング」が可能になります。組み込みプログラミングはパソコン向けのプログラミングとは少し違った手順と考え方が必要になるため、プログラマが少なく、引く手あまたです。このクラス以下のCPUをマイコン(MCU, Micro Controller Unit)ということもあります。今回搭載されたFM3マイコンはこのクラスのCPUで、本誌がメインで扱っている組み込み向けのCPUとなります。

注1: 8ビットと32ビットの間にハイエンド8ビットCPUと16ビットCPUもあるが、32ビットCPUであるARMの低価格化により、8ビットCPUと同程度のコストの32ビットCPUが生産されるようになり(ARM Cortex-M0など)、市場が浸食されてきている。

#### 4~8ビットCPU

ROM 数Kバイト~32Kバイト  
RAM 数バイト~32Kバイト  
OSなし  
極小組み込み機器

小←

#### 32ビットCPU(1チップ)

ROM 64Kバイト~1Mバイト  
RAM 32Kバイト~256Kバイト  
μITRONなど  
組み込み機器

規模

#### 32ビットCPU(メモリ外付け)

ROM 1Mバイト~  
RAM 256Mバイト~4Gバイト  
Linux, Windows CEなど  
ほとんどパソコンと同じ

→大

図1 組み込み機器向けのCPU

## ● Linuxも動くメモリ外付け32ビットCPU

その上がメモリ外付けの32ビットCPUです。メモリを外付けにすることにより、メモリ容量をいくらでも増やすことができるようになり、LinuxやWindows CEのような大規模OSが動くようになります。さらにCPUをAtomのようなx86アーキテクチャにすれば、デスクトップ向けのWindows 7などがそのまま動くようになります。また、パソコンをそのまま内蔵した組み込み機器のような製品も出てきました。

今回のFM3マイコンはメモリ外付け構成も可能で、このような用途でも使うことができます。フルセットのLinuxは動きませんが、MMUを使用しないuClinuxであれば、メモリを外付けすれば十分動作します。

## 2. 組み込み向け開発とパソコン向け開発の違い

### ● ハードウェアも開発する

パソコン向けの開発といった場合、通常はソフトウェアのみの開発となります。ハードウェアはPC/AT互換アーキテクチャといわれるパソコンなので、ハードウェアから開発することはまずありません。

これに対し、組み込み向けの開発では、ハードウェアも含めて開発を行います。パソコン向けソフトウェアでは、図2(a)のようにハードウェアの違いはデバイス・ドライバが吸収し、ソフトウェアからはハードウェアの違いを意識する必要はありません。例えば、ネットワーク経由でデータを送受信するソフトウェアを開発するときに、ネットワーク・インターフェース・カード(NIC)のメーカーやEthernetコントローラのチップ名を気にする必要がありません。デバイス・ドライバはデバイスを提供するメーカー(ここでは

NICのメーカー)が提供するので、アプリケーション開発者はデバイス・ドライバを作る必要がありません(デバイス・ドライバの開発者以外)。

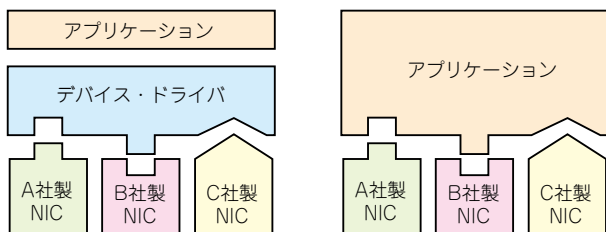
ここで(デバイス・ドライバの開発者以外)と書きましたが、組み込み機器開発では、まさにそのデバイス・ドライバ部分を開発することが仕事になります。組み込み向けでは、製品ごとにより違うハードウェア向けのデバイス・ドライバの開発が重要になります。パソコン向けソフトウェア開発でも、デバイス・ドライバを開発する人は、ある意味組み込み開発者といえなくもありませんが。

### ● 組み込み向けではデバイス・ドライバがない？

パソコン向けOSでは、ハードウェアにアクセスする部分はデバイス・ドライバが行い、アプリケーション・プログラムはデバイス・ドライバへハードウェアへのアクセスを依頼するという構造になっています。

しかし組み込み向けソフトウェアではそうではない場合があります。図2(b)のように、 $\mu$ ITRONなどの組み込み向けOSでは、デバイス・ドライバを介さずにアプリケーションが直接ハードウェアにアクセスすることがあります。これはデバイス・ドライバを介することによるオーバヘッド(メモリ使用量の増加や速度の低下)を嫌ったためであったり、わざわざデバイス・ドライバという抽象的な概念を使わなくても簡単に制御できるハードウェアであるためです(単に手抜きということもあるが…)。本特集で取り上げるプログラムでは、説明を簡単にするために、デバイスをアクセスするときにデバイス・ドライバを使わずにアプリケーションから直接ハードウェアを制御します。

とはいえ、デバイス・ドライバがないと、別のデバイスを使用しようとしたとき、アプリケーションも手直しする必要があります。これは面倒なため、現在では組み込み機器向けOSでもデバイス・ドライバを使用する方向に進んでいます<sup>注2</sup>。



(a) デバイス・ドライバがある場合 (b) デバイス・ドライバがない場合

図2 パソコン向けと組み込み向けのネットワーク・ソフトウェアの違い

## 3. 組み込みソフトウェア開発の手順

### ● パソコン向けとほぼ同じ、しかし一部違う

4ビット～8ビットCPU向けのソフトウェア開発ではメ

注2:  $\mu$ ITRON デバイスドライバ設計ガイドラインなど。

<http://www.ert1.jp/ITRON/GUIDE/device-j.html>