

## 第1章 マイコン・ボードの不良原因と対策の実際

# マイコン回路のトラブル対策

渡辺 明禎  
Akiyoshi Watanabe

### トラブルシューティングの手順

H8/300H(ルネサステクノロジ)を使ったマイコン・ボードを設計・試作したのにうまく動作しないという想定で、どこに原因があるかをチャート形式で探っていきます。そして、原因となっている事柄をピックアップし、その原因と解決法を説明します。逆に、これらの原因を知り、設計段階でそれらに注意すれば、トラブルの少ないワンボード・マイコンを設計できるはずです。

### ■ ワンボード・マイコンとして動作させるまでの三つの工程

開発環境により異なりますが、H8/300Hの場合、ワンボード・マイコンとして実際に動作させるまでの工程には、大きく分けて三つのステップがあります。

- (1)内蔵フラッシュROMへモニタ・プログラムを書き込む
- (2)モニタ・プログラムを使ってアプリケーション・プログラムを制作する
- (3)実行プログラムをROMに書き込んで、スタンド・アロンで動作させる

ここでは、モニタ・プログラムのもとでプログラムを開発することにします。

**モニタ・プログラム**とは、メモリやレジスタのデータを読み書きしたり、プログラムやデータをホスト・マシン(パソコン)とやりとりするような小さなプログラムのことです。図1-1はその画面の例です。

かつてはワンボード・マイコンを製作したら、最初にモニタ・プログラムを自作したのですが、最近のボード・マイコンにはモニタ・プログラムが標準装備

されていて、パソコンのシリアル・ポートと接続するだけで、上述したような最低限の操作ができるものが多いようです。以下の説明では単に「モニタ」と呼ぶことにします。

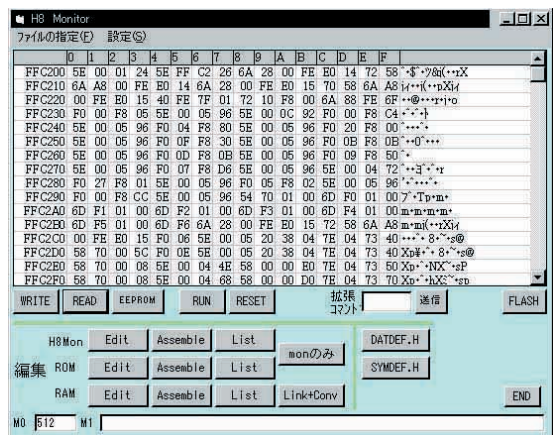
### ■ 内蔵フラッシュROMへモニタを書き込む

フローチャートを図1-2に示します。実際の原因や解決法は、チャート上に記した丸付き数字に該当する本文中の説明を読んでください。

ボードが完成しても、**すぐに電源を投入してはいけません**。はやる気もちを抑え、**電源回りをチェック**しましょう。できれば、電源は電流制限ができて消費電流を監視できる定電圧定電流(CVCC)電源を使うのが望ましいのですが、図中の①部分に注意すれば、通常の電源でも問題はなりません。

フラッシュROMの書き換えは、パソコン上の専用

〈図1-1〉モニタ・プログラムの例(H8Mon1.exe)



### Keywords

モニタ・プログラム, スタンド・アロン, フラッシュROM, ブート・モード, H8/3048F, H8/3067F, H8/3069F, H8/3052F, リセット・スタート, メモリ・アクセス速度, ウェイト挿入, デバッグ・ルーチン, ステップ実行, ボリスイッチ, PTCサーミスタ.

プログラムを使います。通信機能のほかにH8をブート・モードにするための制御回路を内蔵したライター・ケーブルを使うと、ブート・モードによる自動起動などができます。書き込みが正常終了したら、パソコン上でモニタ・プログラムを起動し、実際にモニタがうまく動作しているかを確認します。

## ■ モニタを使ったプログラム開発

フローチャートを図1-3に示します。ここでは、ハードウェア的に問題があり、モニタが正常に動作しないケースも考えます。また、外部RAMが接続されている場合のトラブルを含めた状態でのプログラム開発の流れを示しました。

プログラム開発の方法は、開発環境によって大幅に違うと思います。主な方法を以下にあげます。

- (a) ICEを使った方法
- (b) シミュレータによる方法
- (c) モニタによる方法
- (d) その他

(a)と(b)は開発環境が高価ですし、環境によっては特殊なので、ここでは(c)の方法を紹介します。

モニタを使ったプログラム開発とは、ROMにモニタを書き込み、その力を借りてプログラムを制作する方法です。制作したプログラムをRAMに転送してRAM上で実行することにより、プログラムを効率よく開発できます。また、実機上で実行しながら開発するので、割り込み動作などの検証もでき、プログラムの開発を終えた時点で、すぐ実用的に使えるというメリットがあります。

開発したプログラムは、最終的に内蔵フラッシュROMに書き込み、スタンド・アロンで動作させることができます。

## ■ 実行プログラムのROM化によるスタンド・アロン動作

フローチャートを図1-4に示します。スタンド・アロン動作とは、ホスト・マシン(パソコンなど)に頼らずに、ワンボード・マイコンに電源さえ投入すれば

＜図1-2＞ フラッシュROMへモニタを書き込むフローチャート

