

[第4章]

ハードの挙動を把握しながらデバッグするときに最適な

ICEを使ったソフトウェアの開発例

荻山 正生

本章ではプログラムの開発を行う上で有効なツールであるICE(アイス：In Circuit Emulator)の解説を行います。

ICEを使うことで飛躍的に短時間で確実にプログラムの動作確認ができる。

ICEは高価だがプログラムの動作確認(デバッグ)に悩む多くの時間を考慮すると、導入を検討する価値がある。

世の中にICEの種類は数多くありますが、基本的な機能は同じです。

4-1 ICEとは

ICEはマイクロコンピュータ(PIC)の開発を行うツールの一つで、ソフトウェアとハードウェアの動作確認を行うことができます。

ここで紹介するICEはData Action L.L.C.社のRICE3000(写真4-1)で、動作対象(ターゲットという)はマイクロチップ社のPICです。

利用するときは、基板からPICを外して、その代わりにICEを接続して使用します。

ICE内には外部から制御可能なPICと同じ機能をもっているエミュレーション・チップや制御用コントローラ、トレース・メモリなどがあり、PICの内部データ(レジスタやメモリ)などを表示、変更することができます。

図4-1にICEの内部構成を示します。

プログラムの開発に使われるツールには、ICEのほかに次の2種類があります。

(1) パソコン上で検証するソフト・シミュレータ

基板(ハード)と接続せずにパソコン上の画面の中で動作の確認が行えます。スイッチの入力やLED点灯などは画面のビット表示で代用します。

欠点 ハード回路の動作確認はできない。

(2) ターゲットの特殊端子から制御するデバッガ

ターゲット(PIC)の外部制御用ピンにデバッガなる機器を接続し、ターゲットと通信を行いながら動作の確認が行えます。ターゲットは実際の基板に実装された状態で使用するの、スイッチ入力やLED点灯

このアイコンは、章末に用語解説があります

など実際に目視できます。

欠点 入出力ピンの少ないPICの場合、外部制御用(ICSP)に入出力ピンが割り当てられるため全ピン使用することができない。

4-2 ICEで何ができるか

先ほど紹介したソフト・シミュレータやデバッガにもある機能を含めて、次のことができます。

(1) ステップ動作

プログラムを1ステップずつ順次実行する機能で、実行後は各レジスタの値などが確認できます。

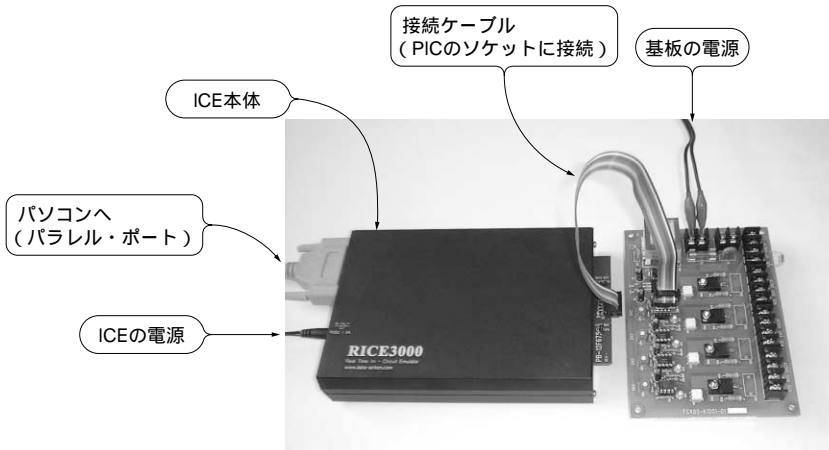


写真4-1 ICEと基板を接続したところ

RICE3000とPIC12F675を使用した基板。

接続ケーブルは10ピンで8ピンはPICソケットに接続し、残りの2ピンはジャンパ・ピンとして使用している。

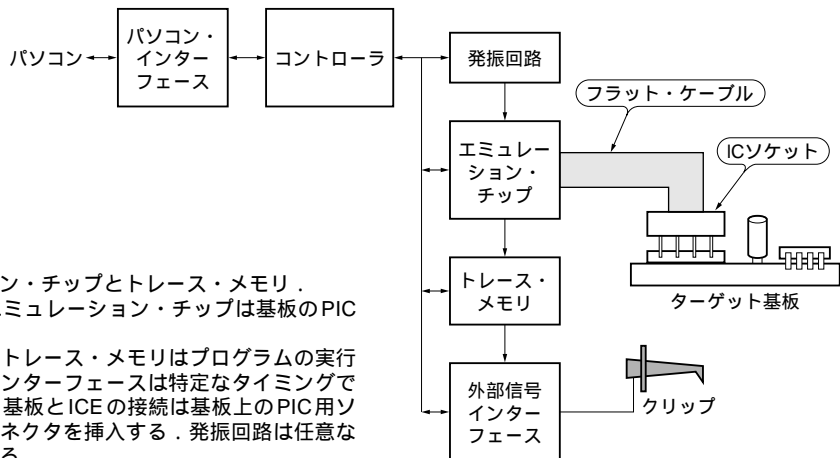


図4-1 ICEの内部構成

主要な部分はエミュレーション・チップとトレース・メモリ。コントローラに制御されたエミュレーション・チップは基板のPICの代わりに動作を行う。

外部信号も同時に記憶できるトレース・メモリはプログラムの実行軌跡を記憶する。外部信号インターフェースは特定なタイミングで信号を出力する機能もある。基板とICEの接続は基板上のPIC用ソケットにICEのケーブル・コネクタを挿入する。発振回路は任意な周波数を発振することができる。

(2) データ編集

レジスタ・ファイル，RAM，EEPROM などを表示したり任意に変更することができます。

(3) ブレーク・ポイント設定

プログラムの任意のアドレスや特定条件で強制的に停止させ，各レジスタなどの値を表示，変更することができます。

(4) トレース機能

プログラムの動作を記録，表示する機能でプログラムの流れを解析することができます。外部入力端子を接続すると外部データも同時に記録できます。

(5) 実基板との接続

実際の基板にICEを接続するのでPICの各ピンはハード回路と直結され，ハード回路の動作確認が容易にできます。

(6) 実時間での動作

一般にターゲット(実際の基板上のPIC)よりも高性能なPICがICEに内蔵されていて，エミュレーション(代理動作)するので，動作時間はターゲットとまったく同じになります。

4-3 ICEの機能詳細

● 環境関係

(1) 環境設定 (図4-2)

PICの種類，発振器，ウォッチ・ドッグ，発振周波数などの設定を行います。

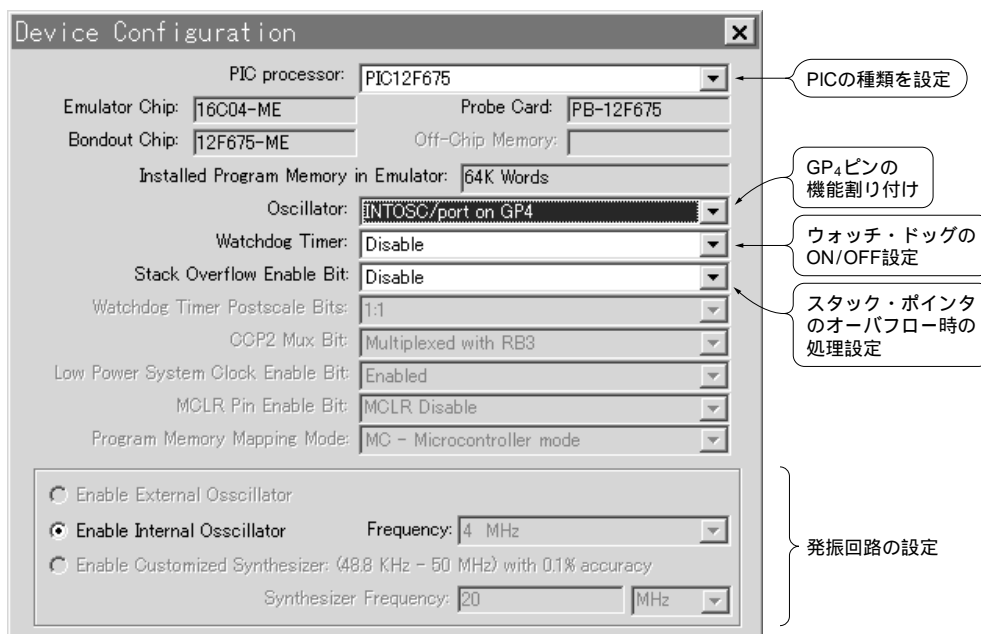


図4-2 環境設定画面

PICの種類の設定を行う。PICの種類によって設定の内容は異なる。

発振器は内部、外部などPICの仕様に合わせて選択可能です。発振周波数は数kHzから数10MHzまで設定できます。

● ファイル関係

(2) ソース・ファイルのオープン

ソース・ファイルを開いて編集することができます。



図4-3 ソース・プログラムの画面
ソース・プログラムの編集画面で、アセンブラを実行するとメモリ・アドレスが表示される。

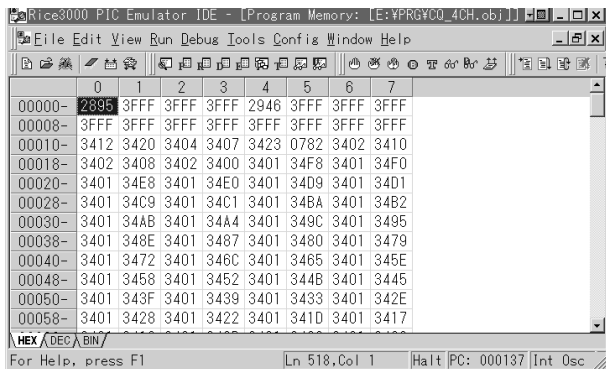


図4-4 プログラム・メモリの画面
プログラム・エリアのデータでHEX, DEC, BINで表示, 変更することができる。

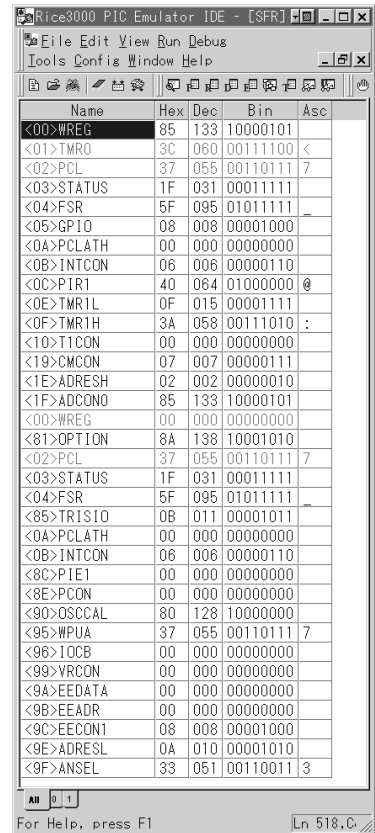


図4-5 SFRの画面
Special Function RegisterのデータでHEX, DEC, BIN, ASCで表示, 変更することができる。

エディタの機能は一通りの作業はできますが、使いやすさなどは専用のテキスト・エディタには劣ります。

(3) ソース・ファイルのアセンブル

PICの実行ファイル(機械語)を生成することができます。ICE上で実行できるプログラムはアセンブラMPASMWINのほかMPC/MPLABC, CC5Xなども可能です。

(4) 実行ファイルのダウンロード

実行ファイルをICEに転送(ダウンロード)できます。

図4-3にダウンロード後のソース・プログラム画面を示します。

● 表示関係

(5) プログラム・メモリの編集

プログラム・メモリのデータを表示、変更、比較ができます(図4-4)。

表示は16進, 10進, 2進の3種類が選択できます。「(11)逆アセンブル・コードの表示」でも編集できます。

(6) SFR(Special Function Register)の編集

スペシャル・ファンクション・レジスタの内容を表示、変更ができます(図4-5)。

プログラム・カウンタ(PCL)の値を変更する場合は、「(11)逆アセンブル・コードの表示」のSet PCで行います。

(7) データ・メモリの編集

データ・メモリのデータを表示、変更ができます(図4-6)。

表示は16進, 10進, 2進, アスキの4種類が選択できます。

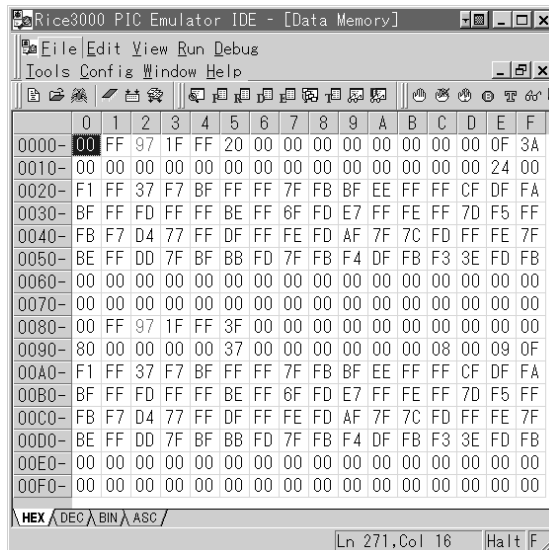


図4-6 データ・メモリの画面
データ・メモリ・エリアのデータでHEX, DEC, BIN, ASC
で表示、変更することができる。

(8) EEPROM データ・メモリの編集

EEPROMのデータを表示, 変更できます(図4-7)。

変更した場合はEEPROMデータのダウンロードが必要です。

表示は16進, 10進, 2進の3種類が選択できます。

(9) スタック・ポインタの表示

8個のスタック・ポインタのデータを表示します(図4-8)。

表示のみで変更はできません。

(10) トレース・メモリの表示

プログラムの実行軌跡を1クロックごとに表示します(図4-9)。

プログラム・アドレス, 実行コマンド, タイム・スタンプ, 外部入力信号(8本), ソース・コードが表示されます。

トレースの条件は(21)を参照願います。

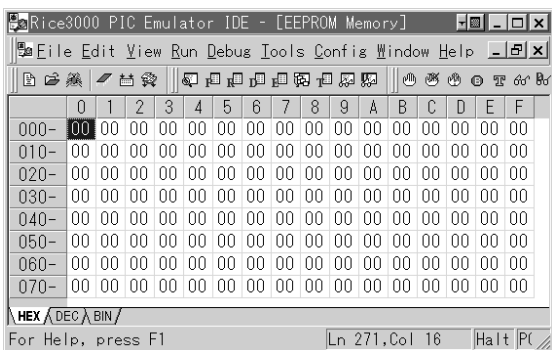


図4-7 EEPROMの画面

EEPROMエリアのデータでHEX, DEC, BINで表示, 変更することができる。

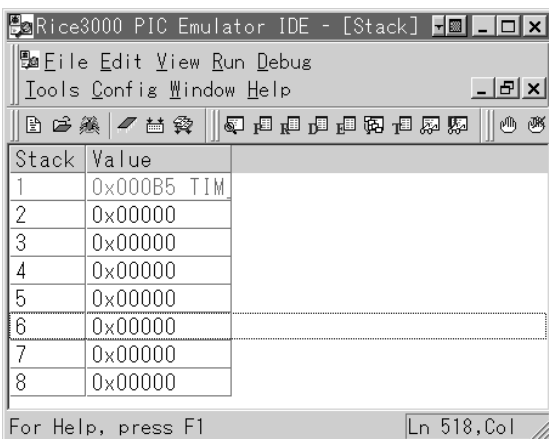


図4-8 スタック・ポインタの画面

現在のスタック・ポインタを示す。

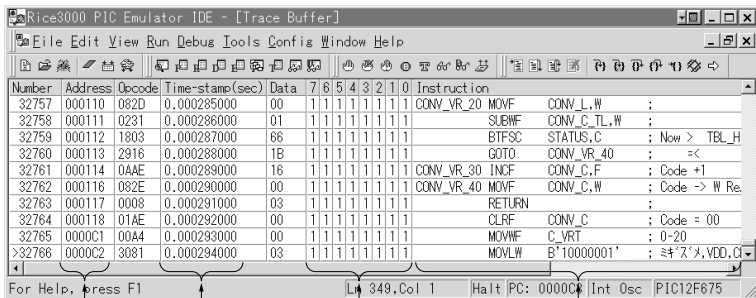


図4-9 トレース・メモリの画面

プログラムの実行軌跡を本器の場合は32766ステップ分表示する。このメモリはエンドレス動作(古いステップから上書き)する。

(11) 逆アセンブル・コードの表示

プログラム・メモリの内容を逆アセンブル表示できます(図4-10)。

「PC:」項にアドレスを設定し、「Set PC」ボタンを押すと、任意のアドレスにプログラム・カウンタを設定できます。また、「Goto」ボタンを押すと、任意のアドレスから実行できます。

● 実行関係

(12) リセット(図4-11)

PICのリセットを行います。

プログラム・カウンタ、スタック・ポインタはゼロになります。

(13) 停止(図4-12)

PICを停止させます。

停止した状態での各データが各ウィンドウに表示されます。各データは以前実行したときの値と異なる場合、データの色は赤にハイライトされます。次に実行するソース行がソース・プログラム画面に矢印で表示されます。

(14) 無条件実行

現在のプログラム・カウンタからプログラムを実行します。

(15) シングル・ステップ実行(図4-13)

プログラムを1ステップ実行し、停止します。

ソース・コードがC言語の場合は1行ずつの実行と、アセンブラ・レベルの1コマンド単位の実行が選択できます。

(16) 1サブルーチン実行(図4-14)

ソース・コードの1行を実行し、停止します。

サブルーチン行の場合も1回の実行で処理が終了します。

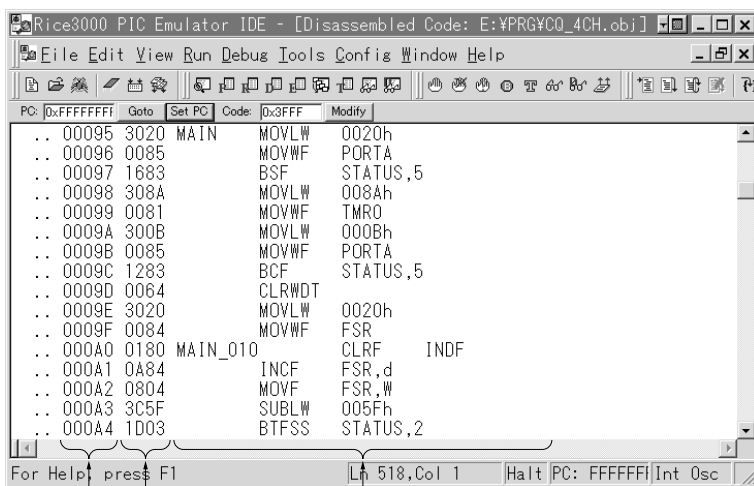
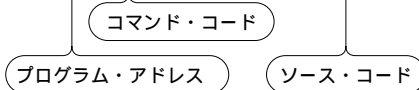


図4-10 逆アセンブル・コードの画面
プログラム・メモリのデータから逆アセ



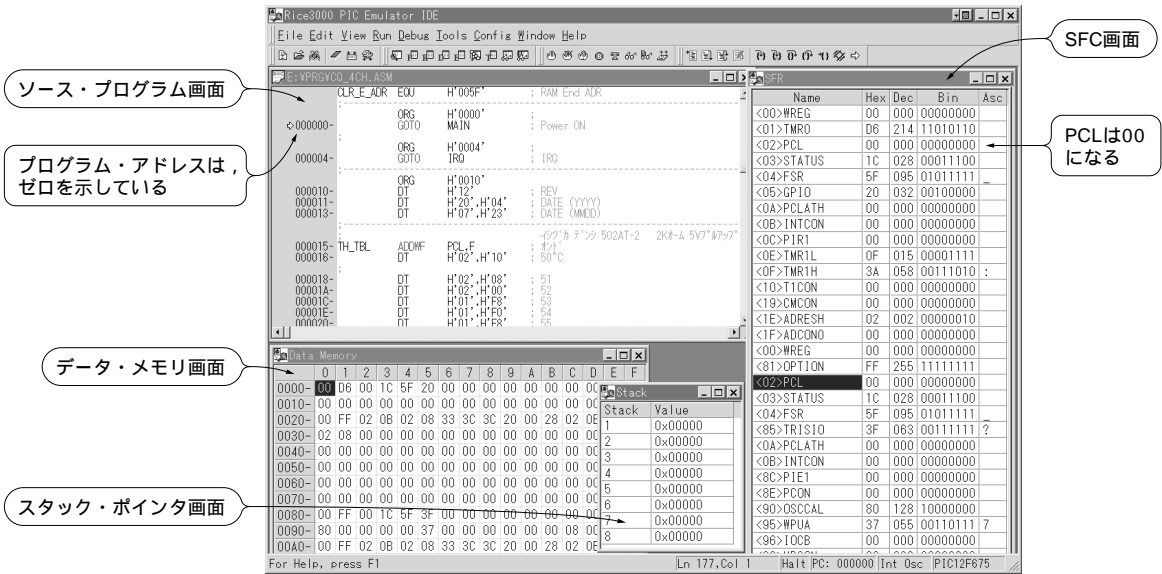


図4-11 リセット直後の表示
通常使用するソース・プログラム、SFR、データ・メモリ、スタック・ポインタの画面。



図4-12 任意に停止した場合の表示
ソース・プログラム画面には停止位置のプログラム・アドレスが表示され、SFR、データ・メモリ、スタック・ポインタ画面には現在の値が表示される。