

ハードの理解で差がつく時代

シミュレータの自作からはじめる CPU動作メカニズム

ご購入はこちら

第3回 RISC-Vシミュレータを動かしてみる

中森章

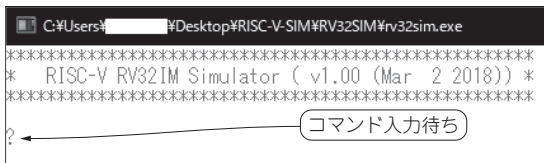


図1 RISC-Vシミュレータの起動画面

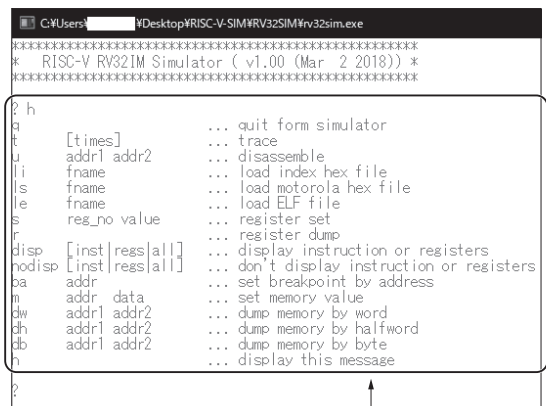


図2 RISC-Vシミュレータのヘルプ・メッセージ

「h」をタイプしてリターン・キーを押せば、ヘルプ・メッセージが表示される

RISC-Vシミュレータの使い方

● ソフトウェア・シミュレータの起動

ソフトウェア・シミュレータ (ISS: Instruction Set Simulator, 以下ISSと記載) 本体は「rv32sim.exe」というファイルです。筆者はWindows環境が好きなので、これはCygwinのgcc (正確には、x86_64-w64-mingw32-gcc) でコンパイルしましたが、しかし、Cygwin上でなくても、Windows上でファイル名をダブル・クリックすることで起動することができます。起動画面を図1に示します。

● ソフトウェア・シミュレータのコマンド

ここで、「h」をタイプしてリターンすればヘルプ・メッセージが表示されます(図2)。このISSは、
コマンド 引き数1 引き数2□
でコマンドを実行します。「引き数」が存在しない場合もあります。□はリターン・キー(エンター・キー)を押すことを示します。表1は各コマンドの解説です。

Dhrystoneベンチマークを 実行してみた

実行の具体例として、図3から図4までに、Dhrystoneベンチマーク(ELFファイルでファイル名はdhry.out)を実行する様子を示します。

これを簡単に説明すると、次の通りです。

(1) le dhry.out

Dhrystoneベンチマークの実行形式(ELF形式)のファイルをシミュレータ内部のメモリにロードし

ます。「start address=0x00002000」はELF形式の内部でエントリが0x2000番地に指定されていることを示します。

(2) s pc 2000

PCに実行開始アドレスである0x2000番地をセットします。ここで、0x2000というのはdhry.outファイルに含まれるmain()関数の開始アドレスです。基本的にはleコマンド実行時の「start address=……」の値と等しいのですが、それと異なる場合もありますから、objdumpコマンドなどでdhry.outを逆アセンブルしてmain()関数の開始アドレスを調べておく必要があります。

(3) u 2000 2010

命令実行を開始する0x2000番地から0x2010番地までを逆アセンブルしています。これは、次にbaコマンドでブレーク・ポイントを設定するアドレスを決定するため、dhry.outのコードが読み込んでいることを示すためです。

(4) ba 2010 stop