

カスタマイズ可能な MIPS プロセッサ用 ビジュアル・シミュレータの実装

西牧悠二 北道淳司 宮崎敏明

会津大学大学院コンピュータ理工学研究科

1. はじめに

コンピュータアーキテクチャ教育においてマイクロアーキテクチャの内部動作を理解することは重要である。我々は、マイクロアーキテクチャの内部構造および動作を視覚化する教育用 MIPS プロセッサシミュレータ(以下、既存シミュレータ)を開発した[1]。本シミュレータはマイクロアーキテクチャの内部動作の理解には有用であるが、固定的な幾つかのマイクロアーキテクチャにのみ対応し、教育目的にそったカスタマイズが難しい。本稿では、既存シミュレータをカスタマイズ可能とする拡張方法(以下、提案シミュレータ)を提案する。

2. 既存シミュレータ

教育用を目的としたプロセッサシミュレータは数多く提案されている[2, 3]。我々は、プロセッサアーキテクチャ教育のために初学者を対象とした MIPS プロセッサシミュレータを開発した[1]。この既存シミュレータは、複数のプロセッサアーキテクチャ(マルチサイクル、パイプライン処理など)に対応し、アセンブラプログラムの実行に対して内部レジスタの内容やアーキテクチャ上のデータの流れを可視化し、プロセッサの内部動作をわかりやすく表示する。既存シミュレータの構成を図 1(a)に示す。既存シミュレータは、アセンブラ、シミュレーションエンジン、アニメーションエンジンからなる。モジュールライブラリは、論理ゲートや ALU などの基本モジュールの動作定義とモジュールの入出力の接続を定義するワイヤ定義からなり、任意のマイクロアーキテクチャを定義できる。アニメーション定義は、モジュールや配線の描写のためのライブラリであり、モジュールの形状、位置、ワイヤの向きおよび配線位置を定義する。

3. 提案シミュレータ

既存シミュレータのアニメーション定義はモジュールライブラリにより定義されるシミュレーションモデルと独立しているので、シミュレー

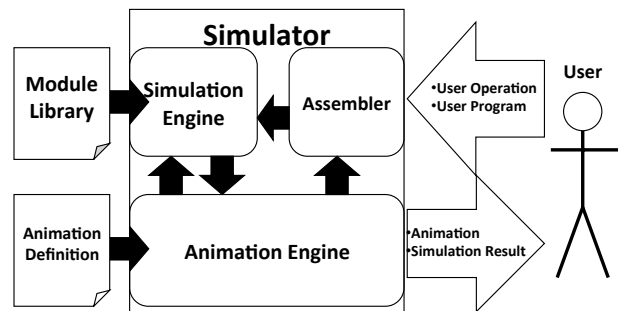


図 1(a). 既存シミュレータの構成

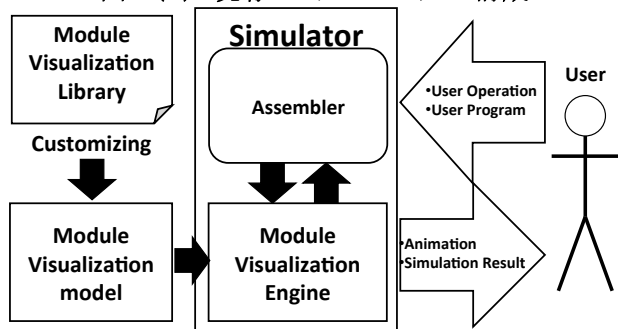


図 1(b). 提案シミュレータの構成

ションモデルの設計者は対応を考慮する必要がある。シミュレーションモデルを変更しやすくするために、本稿では、モジュールライブラリとアニメーション定義を統一する新たなシミュレータを提案する。

図 1(b)に提案シミュレータの構成を示す。提案シミュレータは、モジュールライブラリとアニメーション定義を統合した視覚化モジュールモデルを入力とする。図 2 に提案シミュレータの利用方法を示す。シミュレータの利用者はシングルサイクル、マルチサイクル、および、パイプラインから基本アーキテクチャの選択を行う。これにより、シミュレータが提供する機能の組み合わせが決められる。例えばパイプラインを選択すると、各ステージの動作状況を表示する機能が利用可能になる。次に、視覚化モジュールから必要なものを選択し、GUI ツールを用いてカスタマイズする。モジュールの選択、モジュールの配置、モジュール同士の接続情報の指定、および、配線の配置座標の指定を行い、シミュレーションモデルを完成させる。図 3 にモジュール視覚化モデルの例を示す。視覚化モジュールモデル

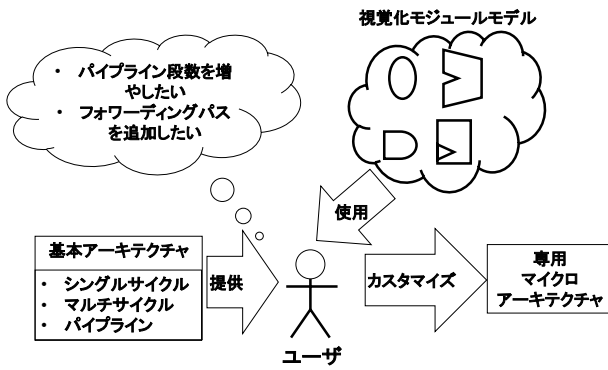


図 2. シミュレータのカスタマイズ方法は、各モジュールに対するアニメーション定義と回路動作定義の対である。視覚化モジュールモデルに従って、シミュレーションエンジンおよびアニメーションエンジンは、レジスタなどの内容を更新するとともに、モジュールやワイヤをアニメーションとして描画する。

アニメーション — 有効な配線 …… 無効な配線		回路動作 マルチプレクサ	
sel = 0	sel = 1	sel = 0	sel = 1
		<p>out の値は in1</p>	<p>out の値は in2</p>

図 3. 視覚化モジュールモデルの例

4. 提案シミュレータの実装

提案機能を実現するために必要な機能は、ユーザが選択したモジュールおよび配線をカスタマイズするためのスキマティックエディタ、カスタマイズされたモジュールにより構成されたアーキテクチャを読み込み、シミュレーションまたは視覚化する機能である。

スキマティックエディタは、ユーザによるモジュールのカスタマイズを支援する。対象モジュールは、ALU や論理ゲートなどのモジュールオブジェクトおよび配線を表すワイヤオブジェクトである。モジュール オブジェクトは、モジュールの入出力、モジュールの形状、動作から定義される。ワイヤオブジェクトは、配線に対応し、配線の座標情報、モジュール同士の接続情報が定義されている。ユーザは基本となるマイクロアーキテクチャを選択すると、スキマティックエディタは、基本の構成を表示する。基本アーキテクチャはモジュールオブジェクトとワイヤオブジェクトによって定義されており、プロセッサ構成要素として必須となるモジュール以外は追加あるいは削除可能である。エディタは、視覚化モジュールモデルにからなるモジュール一覧を表示し、ユーザは一覧に表示されたそれらのモジュールをアーキテクチャに追加し、配

置座標を設定し、次にモジュール間の配線を行う。これは内部的には、ワイヤオブジェクトを生成し、ワイヤオブジェクトに対して配線を形成する座標情報、および、モジュールインスタンスの接続情報を設定することに対応する。また、モジュールオブジェクトに対し、接続されるワイヤオブジェクトの情報を登録する。

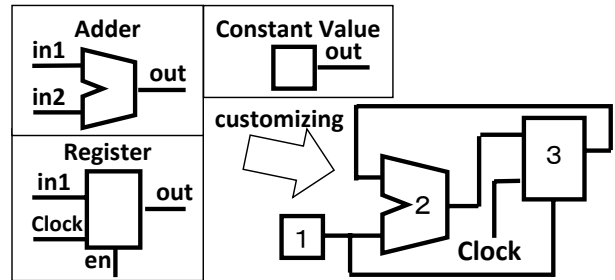


図 4. 視覚化モジュールモデルの生成例

提案シミュレータは、上述の作業で作成された情報を読み込み、シミュレーションと視覚化のための内部情報を生成し、視覚化シミュレーションエンジンに渡す。視覚化シミュレーションエンジンは、モジュールの評価順序を決定する。評価順序は、各モジュールの接続関係を辿ることにより生成する。この評価順序を利用して、モジュールの回路動作(保持する値の更新)およびワイヤ動作(接続されているモジュールの入出力に値を伝達)を実行させ、シミュレーションの実行と視覚化を可能にする。

図 4 に視覚化モジュールモデルの生成例を示す。加算機、レジスタおよび定数発生器を用いて作成されるカウンタである。視覚化シミュレーションエンジンはこの構成から 1→3→2→3 の順序にモジュールを評価し、シミュレーションを実行する。

5. まとめ

本稿では MIPS プロセッサ用ビジュアル・シミュレータのカスタマイズ方法の実装について述べた。今後、提案シミュレータに対してユーザビリティテスト、および評価実験を行いたい。

参考文献

[1] 西牧, 北道, 宮崎, “内部動作を視覚化した教育用 MIPS プロセッサシミュレータシステムの開発,” 電子情報通信学会 和文論文誌 D, Vol. J96-D, No. 10, pp. 2130-2138, 2013.
 [2] James R. Larus, “SPIM A MIPS32 Simulator,” <http://pages.cs.wisc.edu/~larus/spim.html>.
 [3] Kenneth Vollmar, “MARS: an education-oriented MIPS assembly language simulator,” ACM SIGCSE Bulletin, Vol. 38, No. 1, pp. 239-243, March 2006.