

計算機アーキテクチャの教育を支援する可視化レベル変更可能な PDP-11 シミュレータ

林 毅

法政大学 大学院

理工学研究科 応用情報工学専攻

和田 幸一

法政大学

理工学部 応用情報工学科

計算機アーキテクチャの学習を支援する PDP-11 シミュレータを考える。提案するシミュレータは PDP-11 のマシンサイクルの各ステージの動作を提示するだけでなく、ステージ毎に内部動作を詳しく提示し、バスの制御のレベルを可視化することができる。このシミュレータは、コンピュータの内部動作をより深いレベルまで学習できることを目的としている。

1.はじめに

法政大学では、計算機アーキテクチャとアセンブリ言語の講義がある。現在アセンブリ言語の講義では、COMET2 に対応したアセンブリ言語 CASL2 のシミュレータを用いて学生がアセンブリ言語を学んでいる。COMET2 はアセンブリ言語の学習用に開発された仮想計算機であるため、アセンブリのプログラムとしてアドレッシングモードなどの機能があるが、実用化されている CPU に比べると種類が少なく、単純であり、計算機アーキテクチャを学ぶにはもの足りないところがある。そのため計算機アーキテクチャの講義では PDP-11 を用いて講義が行われている。そこで、今回はその PDP-11 のシミュレータの提案、実装を行い、計算機アーキテクチャの教育を支援する。また、アセンブリ言語の教育支援に関しては「アセンブリ言語教育支援教材 Sim AI の設計と実装(小林 晴紀)」を参照する。

2. PDP-11

PDP-11 は DEC 社が開発した 16 ビットミニコンピュータである。命令は 1 語 16 ビットで、命令数は 60 種類あるため、オペレーションコードは可変長になっている。番地はバイト毎についているバイトマシンで、1 バイト(8 ビット)のデータに対する演算も用意されている。汎用レジスタ (general register) は R0 から R7 までの 8 本あり、これは 3 ビットで指定する。そのうち R7 はプログラムカウンタ (program counter) PC を兼ねており、また、R6 はスタックポインタ (stack pointer) として使用される。メモリの番地 (address) は 16 ビットで指定するので、主記憶は最大でも 32K 語(64kB)までとなる。そのうち、実記憶は 28K 語で、最上位番地の 4k 語は特別な用途に割り当てられている。(参考文献[1])

3.PDP-11 シミュレータ

シミュレータの基本的な機能を述べる。提案するシミュレータではアセンブリ言語の記述が可能で、ユーザーは自由にプログラムの編集、実行が可能である。また、エディターウインドウではレジスタの内容とメモリの内容が確認できる。本シミュレータでは、プログラムを実行したあと、ユーザーはプログラムの実行に対応したマシンサイクルの内部動作をマシンサイクルウインドウで確認できる。

3.1. PDP-11 シミュレータの特徴

このシミュレータではマシンサイクルを動作の抽象度によって、レベルに分けたシミュレーションを可能としている。そのレベルを 3 段階に分け、レベル 1~3 とする。レベル 1 では、アドレスの内容やデコード結果、演算結果を表示させて、マシンサイクルの大まかな流れを見せる。レベル 2 では、フェッチされたアドレスの内容やデコードされた命令がどのように処理されているのかを表示させて、各ステージの細かい流れを見せる。レベル 3 では、メモリへのアクセス方法やデコード方法、演算方法をバスレベルでより細かく表示させて、各ステージの処理をより細かく見せる。各レベルで提示するものを表 1 に示す。また、このシミュレータで学習できる計算機アーキテクチャの機能を表 2 に示す。

表 1. 各レベルで提示するもの

レベル 1	レジスタ, 主記憶, 各ステージの入出力を提示させてマシンサイクルの大まかな流れを理解させる.
レベル 2(フェッチステージ)	プログラムカウンタが指すアドレスがどのように保持されて, デコードステージへ渡されるのかを提示する.
レベル 2(デコードステージ)	フェッチステージからもらったデータがどのようなデータであるのかを提示する. 2語命令以上の場合は, 再びフェッチステージに移行することがわかるようにもする.
レベル 2(実行ステージ)	各命令がどのようにして, 実行されるのかを図示する. 計算式や条件の分岐を見せる.
レベル 3(フェッチステージ)	割り込み処理の動作を提示する. バスの同期の取り方を理解させる.
レベル 3(デコードステージ)	命令がデコード内でどのようにして決められているのかを, セレクタ回路などを提示させて理解させる.
レベル 3(実行ステージ)	データがどのようにして演算されているのかを, 論理回路レベルで図示し, ALU の構成, 動作を理解させる.

表 2. 各レベルで学習できる計算機アーキテクチャ

レベル 1	CPU の構造 命令サイクル
レベル 2	命令形式, 演算コード, アドレッシングモード(直接、即値) オペランド アドレッシングモードの実現 サブルーチンの実現 命令の記号化 バスの構造 論理演算とシフト 算術論理演算命令 加減乗除命令、ロード・ストア命令 分岐命令 機械語命令形式 具体的な計算機の機械語命令 スタックの概念 スタックポインタ スタック操作命令 サブルーチンの基本 サブルーチンの概念
レベル 3	バスの制御(Data in, Data out) ALU の構成 割り込みの概念 割り込み要因・動作

3.1.1. シミュレートレベル 1

シミュレートレベル 1 では以下のものを図示する。このレベルで表示させるマシンサイクルウィンドウを図 3.1 に示す。

- ・アドレスバス(AB)
- ・データバス(DB)
- ・CPU の内部バス
- ・汎用レジスタ R0~R7 の内容
- ・メモリの内容
- ・フェッチステージに入力されるメモリのアドレスと内容
- ・デコードステージで解析された命令のニモニックと機械語
- ・実行ステージで実行された演算名と演算結果と条件フラグ

シミュレートレベル 1 では、レジスタの内容の変化や各ステージの入出力の結果を表示させて、マシンサイクルの大きな流れを理解する。

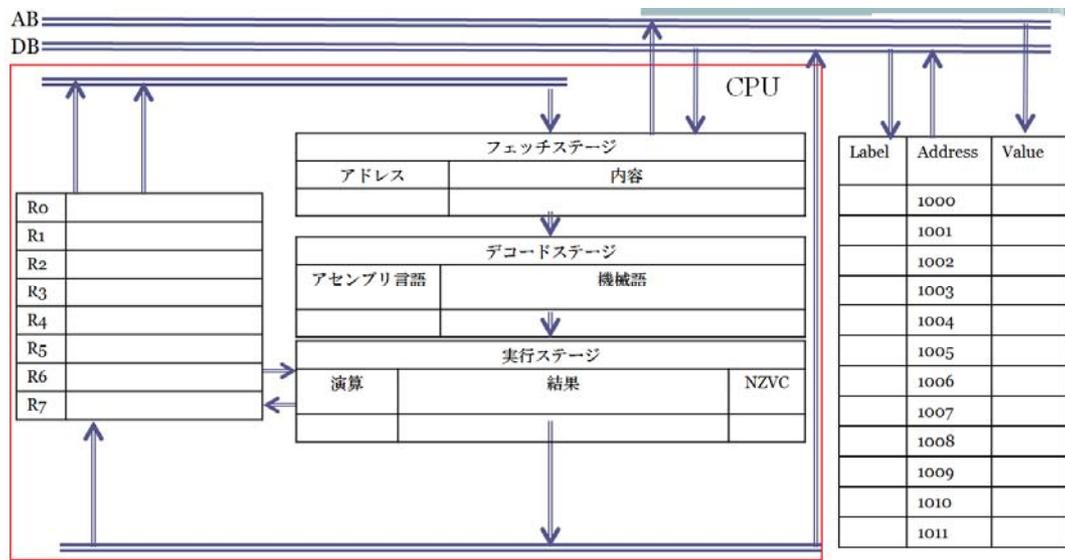


図 3.1.提案するマシンサイクルウィンドウ

3.1.2. シミュレートレベル 2

より細かく見たいステージの内部動作をレベル 1 より掘り下げてみる事ができる。

フェッチステージでは、PC(Program Counter)の内容が内部バスをとって BAR(Bus Address Register)に格納される。その値が指すアドレスの内容が BDR(Bus Data Register)に格納され、IR(Instruction Register)に送られる。(図 3.2)この一連の動作がフェッチステージのシミュレートレベル 2 となる。

フェッチステージのシミュレートレベル 2 で表示するものをいかに示す。

- ・アドレスバス(AB)
- ・データバス(DB)
- ・CPU の内部バス
- ・メモリの内容
- ・汎用レジスタ R7(PC)
- ・バスアドレスレジスタ(BAR)
- ・バスデータレジスタ(BDR)

・インストラクションレジスタ(IR)

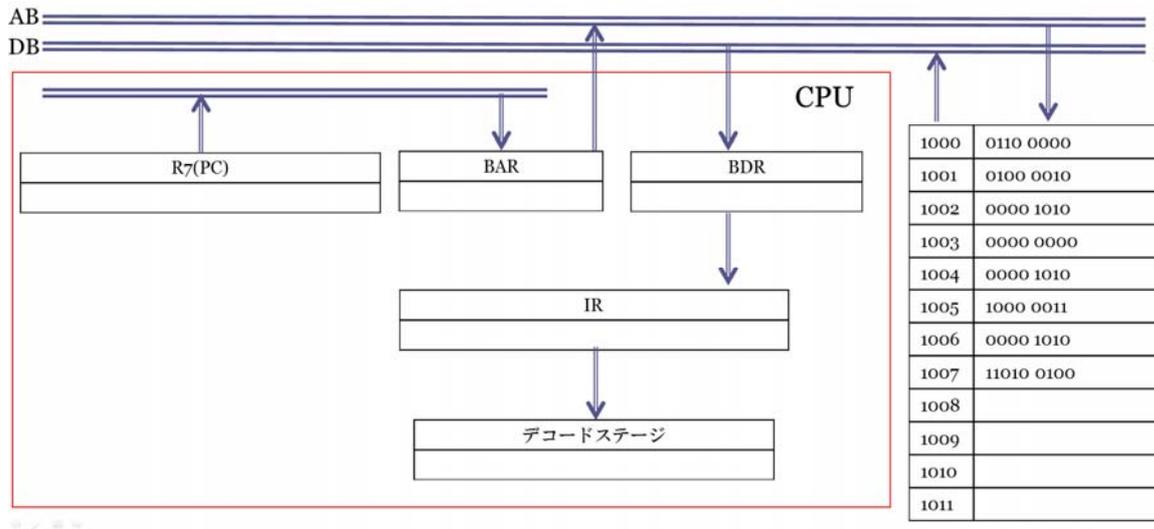


図 3.2.フェッチステージのシミュレートレベル 2

デコードステージでは、IR に格納されている内容の命令の種類、その命令の説明、機能を表示させる。その命令の構成、opc(operation code), NZVC(条件フラグ), dst(destination operand),src(source operand)などの情報も加えて表示させる。(図 3.3)命令が 2 語命令以上の場合、フェッチステージに移行することを理解させるために、2word の命令の場合は、図中の fetch の部分が変色する。

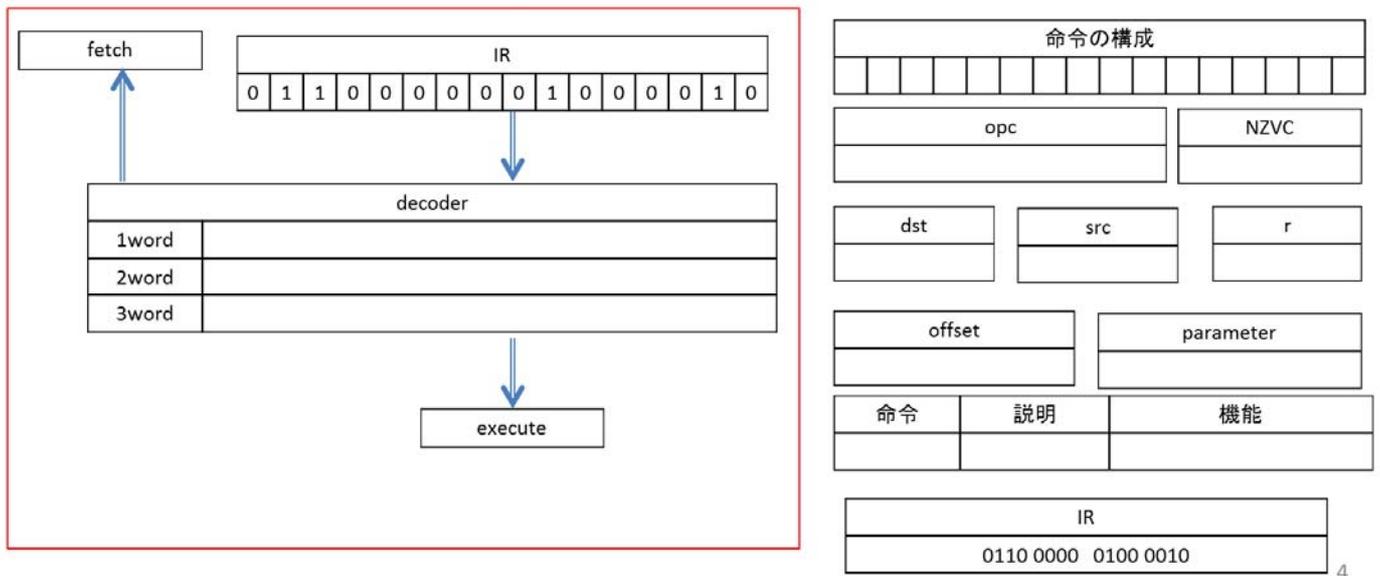


図 3.3.デコードステージのシミュレートレベル 2

実行ステージでは、各命令に対応した形式で図示し、実行時の CPU の内部動作を可視化する。

算術演算命令

16 ビットの 2 の補数表現として演算を行い、結果によって条件コード NZVC を設定する。

表 3. 実行ステージで可視化される算術演算命令

算術演算命令	比較 加算 減算 クリア 1 加算 1 減算 2 の補数 キャリー加算 キャリー減算 テスト
--------	---

算術演算命令では、表 3 に示す演算命令の動作が可視化される。ウインドウには、IBA, IBB, レジスタ, dst の値, src の値, 演算内容, 演算結果, PSW が図示されており, これらの値とデータの移動が確認できる(図 3.3)。また, クリア, 1 加算命令など, src が使われず決まった

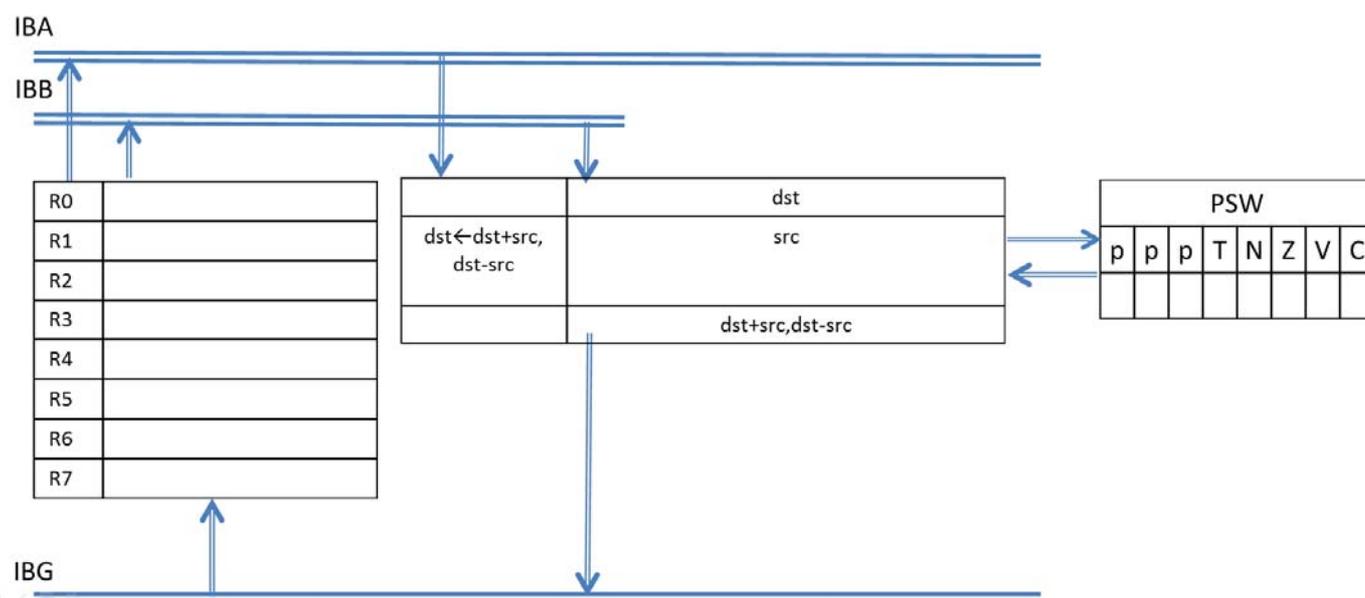


図 3.3. 実行ステージ算術命令(例 1 加算, 減算)

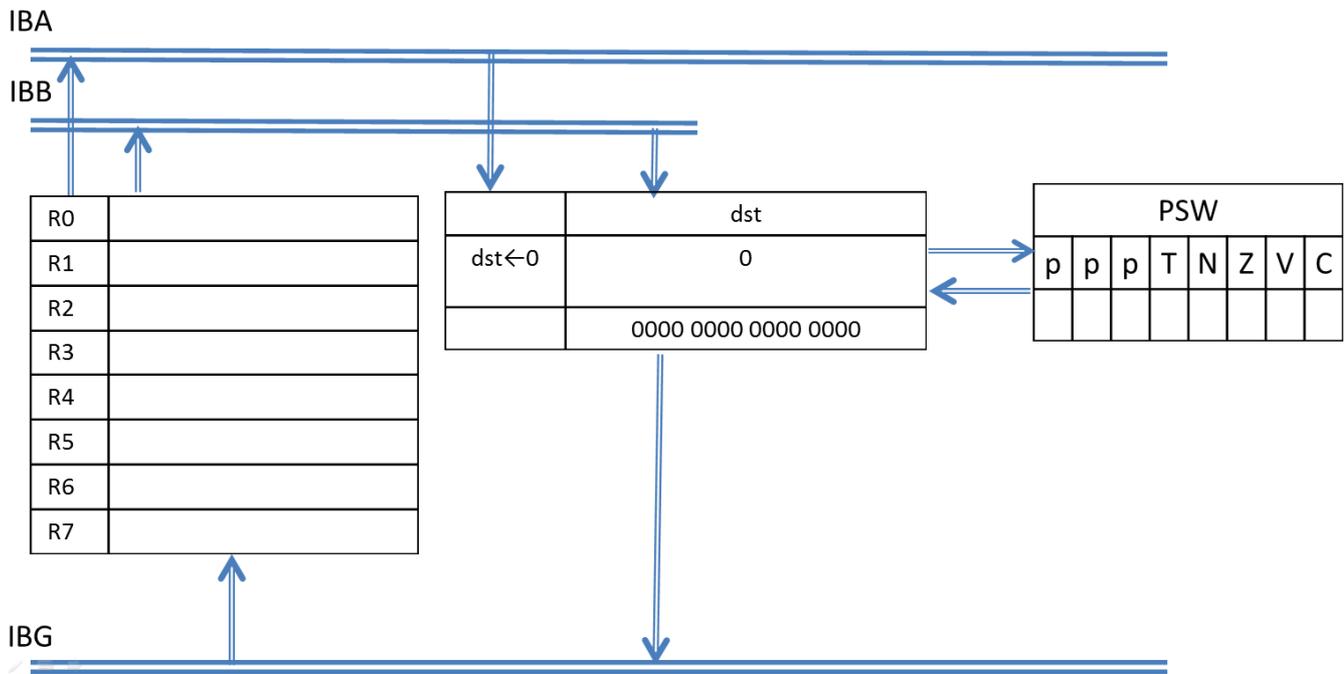


図 3.4. 実行ステージ算術命令(例 2 クリア)

論理演算命令

論理演算命令では、表 4 に示す演算命令の動作が可視化される。

表 4. 実行ステージで可視化される論理演算命令

論理演算命令	転送 ビットテスト ビットクリア ビットセット 反転 バイト交換
--------	---

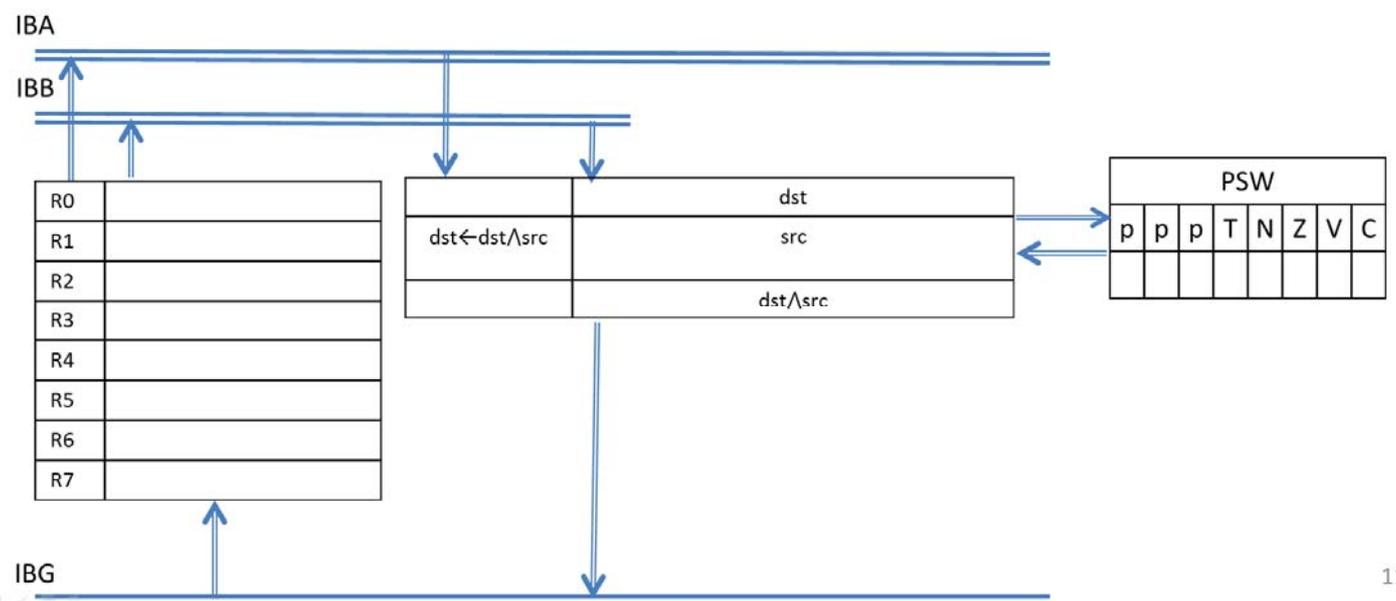


図 3.5. ビットテスト

シフト命令

シフト命令は、データを1ビット、右か左に動かすもので、右か左かと、算術データか、論理データかで4通りのシフト法がある(表5)。シフト命令のウインドウ画面では、IBA、IBB、IBG、レジスタ、dst、C、PSWを表示させる。シフト前後のデータの変化をユーザーに理解してもらい、その時の条件コードもわかるようにする。

表5. 実行ステージで可視化されるシフト命令

シフト命令	右巡回シフト 左巡回シフト 算術右シフト 算術左シフト
-------	--------------------------------------

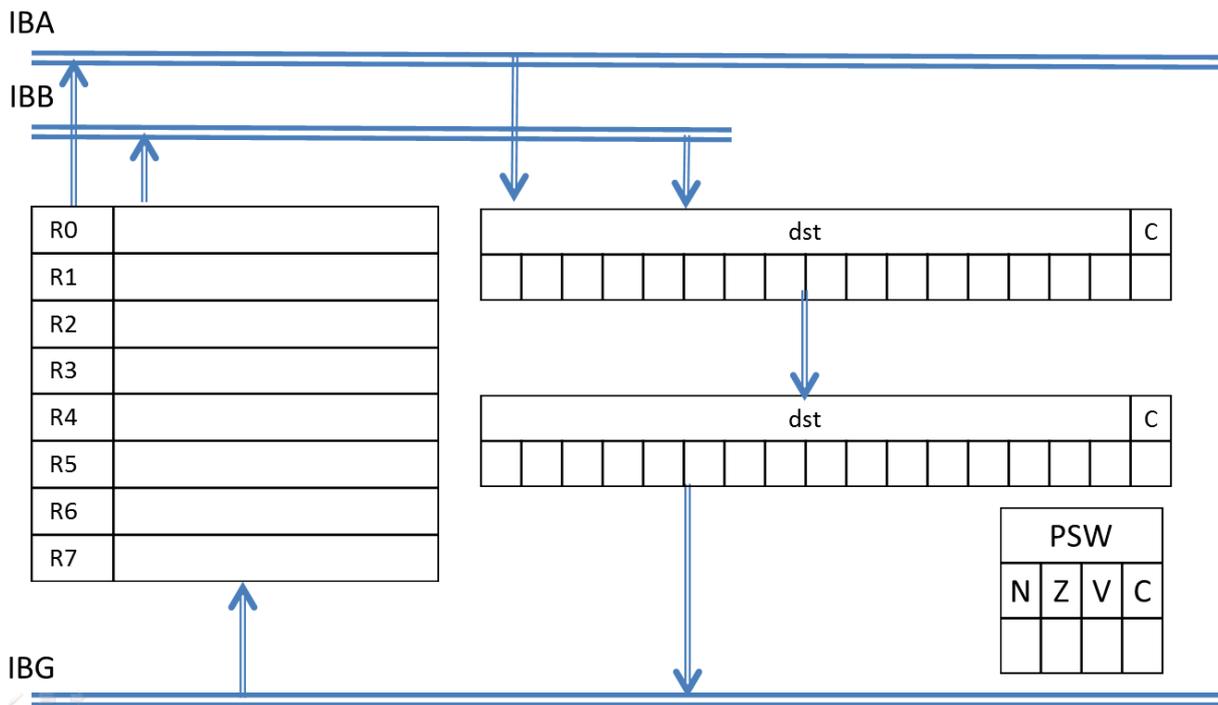


図3.6. シフト演算

バイト命令

いくつかの命令は対応したバイトデータに対する演算をする命令がある(表6)。

バイト命令は基本的に、算術演算、論理演算、シフト命令と同じ動作をするためウインドウは上記の図と同じように図示する。しかし、アドレッシングモードのバイト命令の場合、処理するアドレスは下位バイトのみである。

表6. 実行ステージで可視化されるバイト命令

バイト命令	転送 比較 ビットテスト ビットクリア ビットセット クリア 反転 1 加算 1 減算
-------	---

	2 の補数 キャリー加算 キャリー減算 テスト 右巡回シフト 左巡回シフト 算術右シフト 算術左シフト
--	---

分岐命令

分岐命令は、条件コードの値(表 7)によって、PC の値をそのまま使うか、あるいは、オフセットを加算して使うかを切り替える。

表 7. 実行ステージで可視化される分岐命令

分岐命令	分岐条件
	True
	Z=0
	Z=1
	N=0
	N=1
	V=0
	V=1
	C=0
	C=1
符号付分岐	
N=V	
N≠V	
Z=1∧N=V	
Z=1∨N≠V	
符号なし分岐	
C∨Z=0	
C∨Z=1	
C=0	
C=1	

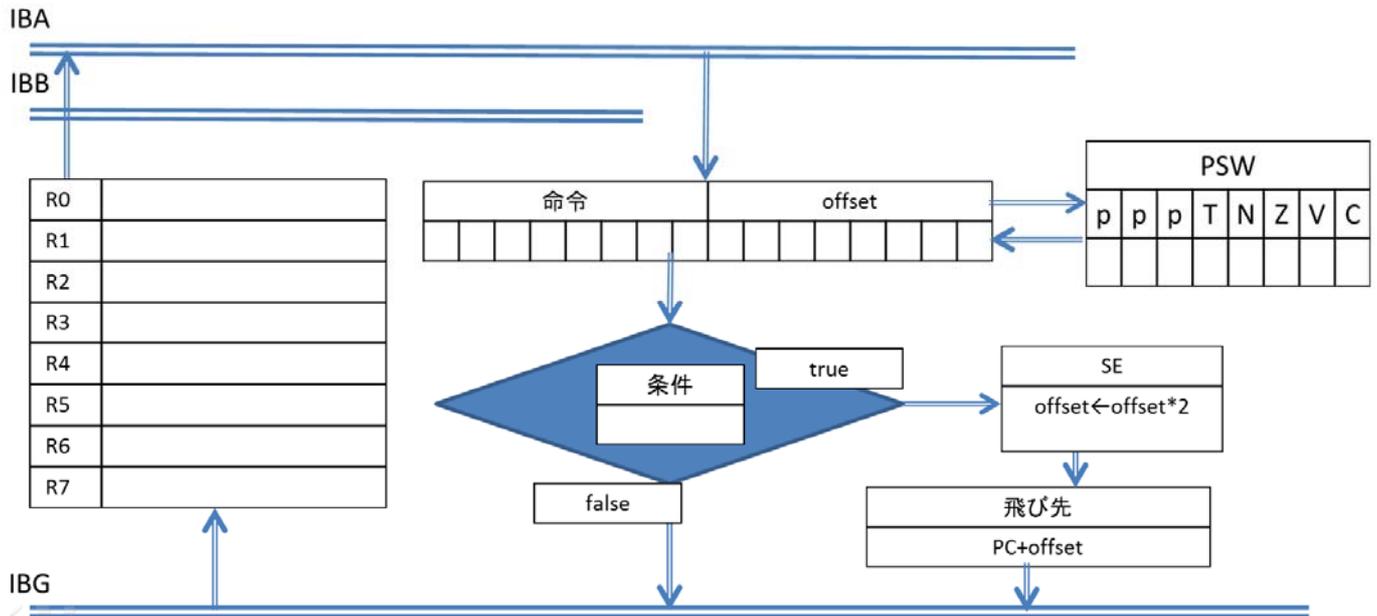


図 3.7. 分岐命令

その他の実行制御命令

その他の実行制御命令では、表 4 に示す演算命令の動作が可視化される。

表 8. 実行ステージで可視化されるその他の実行制御命令

その他の実行制御命令	ジャンプ命令 サブルーチン呼び出し サブルーチンからの戻り ソフトウェア割り込み 割り込みからの戻り リセット, 待ち 停止 ソフトウェア割り込み トラップ
------------	--

アドレッシングモード

PDP-11 の特色のひとつはアドレッシングモードである。オペランドの指定では、コンピュータの中のどの場所のデータにアクセスするかを決める。そのデータは、大きく分けると汎用レジスタ、主記憶、命令中に書き込まれているデータである。主記憶はバイト毎に番地が与えられており、16 ビットの番地で指定される。レジスタは 8 個あり、3 ビットで指定できる。R7 はプログラムカウンタ(program counter), R6 はスタックポインタ(stack pointer)として使用する。したがって、アプリケーションプログラムで汎用レジスタとして自由に使えるのは R0 から R5 の 6 つである。ただ、PDP-11 では、PC やスタックポインタを特別扱いせず、汎用レジスタとして平等に扱うことを選択したために、豊富なアドレッシングモード (addressing mode) が可能となった。8 つのアドレッシングモードがあり、やや複雑に見えるが、実際によく使うのはモード 0, 2, 4, 6 で、モード 3, 5, 7 はあまり使うことはない。あと、レジスタとして、プログラムカウンタ PC を使うとき、直値、絶対番地、相対番地など重要なアドレッシングモードになる。ソースオペランド、デスティネーションオペランドは、機械語では 6 ビットであるが、それはつぎのように、上位 3 ビットがモード、下位 3 ビットがレジスタの指定に使われる。アドレッシングモードの種類を表 7 に示す。

表 9. 実行ステージで可視化されるアドレッシングモード

アドレッシングモード	
直接アドレッシング	レジスタ 自動加算 自動減算 インデックス
間接アドレッシング	レジスタ間接 自動加算間接 自動減算間接 インデックス間接
PC を使う	直値 絶対 相対 相対間接

3.1.3. シミュレートレベル 3

シミュレートレベル 3 では割り込み処理, デコーダーの内部動作, ALU の内部動作を提示する.

割り込み処理

割り込みはコンピュータ内外の状況によって, 通常の処理を一時中断し, その状況に応じた処理を優先して実行する仕組みである. シミュレートレベル 3 の割込命令では,新たに MSYN(master synchronization),SSYN(slave synchronization),read,write を表示させる(図 3.8).これらは,MSYN と SSYN で同期をとり,read と write の値を書き換えて割り込みの制御を行う.

シミュレートレベル 3 では,より複雑な計算機アーキテクチャの構造となるので,さらに詳しくみることもできる.割込み命令では,図 3.9 のように,AB,DB,CB(control bus)に乗っているデータを表示させ,CPU とメモリ間でどの用にデータがやりとりされているのか細かくみることができる.また,Data in の図では,割込み制御の同期の取り方の順序を追って確認できる.



図 3.9. 割り込み命令

デコーダー内部動作

セレクタ回路で命令の解析を行っており、その解析の方法を回路図で提示し、デコーダーの原理を理解してもらう。

ALU 内部動作

算術演算，論理演算，シフト演算などすべての命令での ALU 内部動作を提示し、命令に対して ALU の内部ではどのように演算されているのかを理解してもらう。

4. 計算機アーキテクチャの各項目の実現，今後の課題

現在は提案シミュレータの構成を考えており，レベル 1，レベル 2 のフェッチステージ，デコードステージの提案は実装できる環境にある．レベル 2 の実行ステージでは，各命令に対応したシミュレートが必要なため，今後すべての命令に対応したシミュレータの提案を進める．レベル 3 のフェッチステージでは，割り込み処理の動作の提案は完了しているが，デコーダーの内部動作，ALU の内部動作は，バスレベルでの考案が必要なため，現段階では実装はできない環境である．

5. おわりに

本研究では，計算機アーキテクチャとアセンブリ言語の教育の現状を調査し，シミュレータの要件定義を行った．今後の課題として，可視化できるレベルをより掘り下げていく．そして，実装を行い学生の講義で使用してもらう．

6. 参考文献

- [1] PDP-11 Handbook by Digital Equipment Corporation(1979)
- [2] 法政大学 理工学部 応用情報工学科 計算機アーキテクチャ 講義資料
- [3] 法政大学 理工学部 応用情報工学科 計算機ハードウェア 講義資料