

計算機科学実験 3 A

アーキテクチャ拡張仕様書

チーム 4 : 竹田原俊介 Chung Mung Tim

2022-06-02

第1章 概要

この実験の目的は、SIMPLE (SIxteen-bit MicroProcessor for Laboratory Experiment) と呼ばれるコンピュータを設計することである。このレポートでは、SIMPLE/B 基本アーキテクチャからの拡張仕様とその利点を説明する。

第2章 即値オペランドの強化

基本アーキテクチャでは、演算命令のオペランドはいずれも汎用レジスタであり、レジスタに即値を加えることは少なくとも2命令を要する。 $r[Rd] + \text{sign_ext}(d)$ を実行できるようにし、そして $r[Rd] + r[Rs]$ と $r[Rd] + \text{sign_ext}(d)$ を切り替えられると望ましい。

そこで演算命令形式を以下のようにする。元々の d の中の1ビットをフラグ f に用いるようになった。

15	14	13	11	10	8	7	4	3	2	0
11		Rs		Rd		op3		f	d	

- f が0の場合、 $r[Rd] + r[Rs]$ を計算する。
- f が1の場合、 $r[Rd] + \text{sign_ext}(\text{conc}(Rs, d))$ を計算する。ここで、 $\text{conc}(Rs, d)$ は Rs フィールドの3ビットと d フィールドの3ビットを結合した値を表す。

このようにして得られるオペランドを $\text{sw}(f, Rs, d)$ と表記すると、SIMPLE の演算命令は表2.1のようになる。こうすることで、即値オペランドを使った計算に必要な命令数を減らすことができる。

表 2.1 即値オペランドが強化された SIMPLE の演算命令

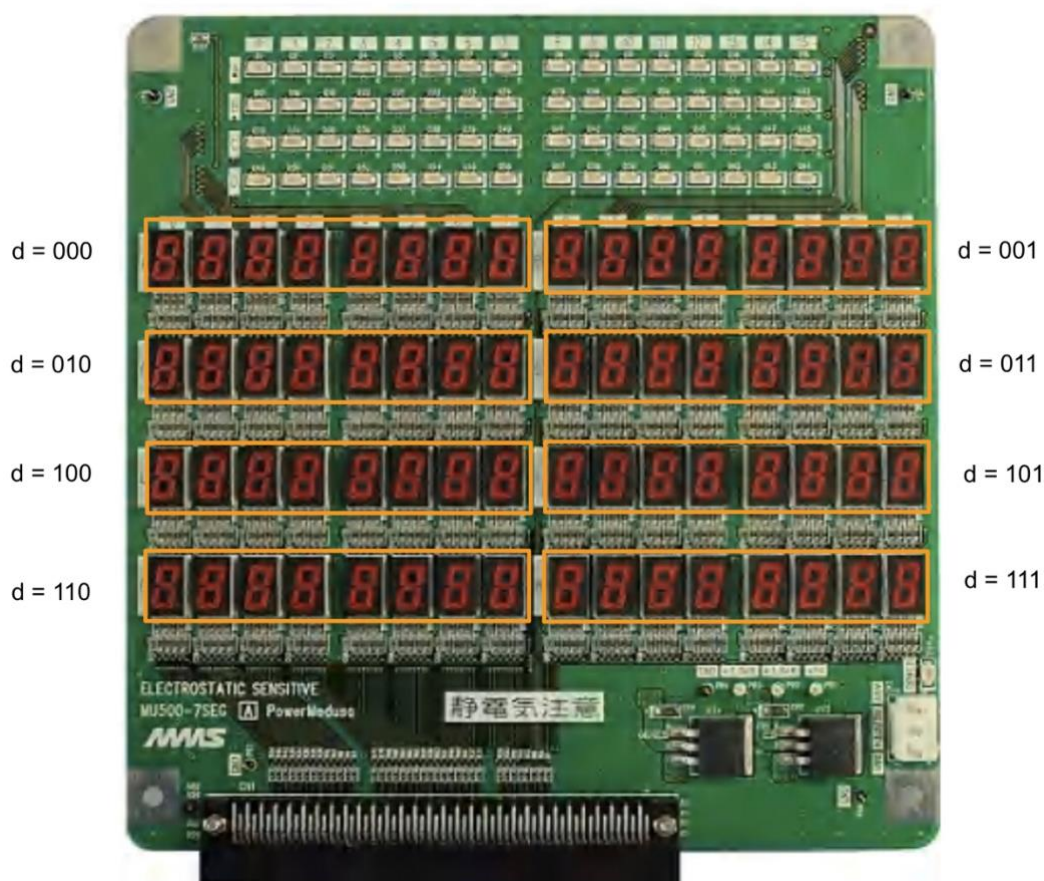
mnemonic		op3	function
ADD	Rd, Rs	0000	$r[Rd] = r[Rd] + \text{sw}(f, Rs, d)$
SUB	Rd, Rs	0001	$r[Rd] = r[Rd] - \text{sw}(f, Rs, d)$

AND	Rd, Rs	0010	$r[Rd] = r[Rd] \& sw(f, Rs, d)$
OR	Rd, Rs	0011	$r[Rd] = r[Rd] sw(f, Rs, d)$
XOR	Rd, Rs	0100	$r[Rd] = r[Rd] \wedge sw(f, Rs, d)$
CMP	Rd, Rs	0101	$r[Rd] - sw(f, Rs, d)$
MOV	Rd, Rs	0110	$r[Rd] = r[Rs]$
(reserved)		0111	/
SLL	Rd, d	1000	$r[Rd] = \text{shift_left_logical} (r[Rd], d)$
SLR	Rd, d	1001	$r[Rd] = \text{shift_left_rotate} (r[Rd], d)$
SRL	Rd, d	1010	$r[Rd] = \text{shift_right_logical} (r[Rd], d)$
SRA	Rd, d	1011	$r[Rd] = \text{shift_right_arithmetic} (r[Rd], d)$
mnemonic		op3	function
ADD	Rd, Rs	0000	$r[Rd] = r[Rd] + sw(f, Rs, d)$
SUB	Rd, Rs	0001	$r[Rd] = r[Rd] - sw(f, Rs, d)$
AND	Rd, Rs	0010	$r[Rd] = r[Rd] \& sw(f, Rs, d)$
OR	Rd, Rs	0011	$r[Rd] = r[Rd] sw(f, Rs, d)$
XOR	Rd, Rs	0100	$r[Rd] = r[Rd] \wedge sw(f, Rs, d)$
CMP	Rd, Rs	0101	$r[Rd] - sw(f, Rs, d)$
MOV	Rd, Rs	0110	$r[Rd] = sw(f, Rs, d)$
(reserved)		0111	/
SLL	Rd, d	1000	$r[Rd] = \text{shift_left_logical} (r[Rd], d)$
SLR	Rd, d	1001	$r[Rd] = \text{shift_left_rotate} (r[Rd], d)$
SRL	Rd, d	1010	$r[Rd] = \text{shift_right_logical} (r[Rd], d)$
SRA	Rd, d	1011	$r[Rd] = \text{shift_right_arithmetic} (r[Rd], d)$

第3章 出力命令（OUT 命令）の強化

レジスタに格納された 16 ビットの値を二つボードに LED などとして出力する OUT 命令も提供されている。私が設計した SIMPLE では、その二つの 16 ビットの値をそれぞれ 4 桁 16 進数として Power Medusa EC6S ボードの 8 桁 7SEG LED に映ることができる。また、命令中の 3 ビットフィールド（d）を使って、出力の場所を指定できるように設計した（図 3.1 参照）。

図 3.1 出力 OUT 命令



こうすることで、ユーザーは数値を出力する場所の選択肢を増やすことができます。例えば、足し算の結果をある場所で出力し、シフトの結果を別の場所で出力する、といったように、出力する場所を利用して演算を表現することが考えられるだろう。