

CPUのしくみ

(3)分岐

人類文明継続装置
輪廻 ヒロ

監修: 矢口 裕明
(博士(情報理工学))

はじめに

前回はプログラム内蔵方式の説明でした。
プログラム内蔵方式では、
プログラムカウンタに書き込まれたアドレスを
1命令の大きさ分加算しながら
順番に実行していました。

分岐命令とは？

順番に実行以外の実行がしたい。

- 条件によって実行内容を変える
- 条件を満たすまで繰り返す
- プログラム部品を使いまわす

->プログラムカウンタそのものを
書き換えることができれば
実現できるのでは？

->この操作をする命令=分岐命令

条件分岐と無条件分岐

条件分岐:

特定のレジスタの値を見て分岐する

無条件分岐:

強制的に分岐する

どちらの場合も以下のような操作をする。

- プログラムカウンタを書き換える。
- 現在のプログラムカウンタの
次の命令のアドレスをレジスタに退避
(JAL: Jump And Link)
 - そのアドレスをリターンアドレスと呼ぶ

ループとサブルーチン

ループの実現方法

今のプログラムカウンタよりも
前のアドレスに分岐する。

サブルーチンの実現方法

JALで呼び出す前の
プログラムカウンタを保存、
呼び出した後に復元。

実際の分岐命令について

CPUによって違う。なんなら呼び方も違う。

BranchのBから始まる命令と
JumpのJから始まる命令がある。
Bが条件分岐かと思いきや
Bの無条件分岐命令、
Jの条件分岐命令もある。

条件を判定するために
フラグレジスタを専用に持つ場合がある。

サブルーチン呼び出し用に
スタックポインタを専用に持つ場合がある。

分岐を構成する命令について

この資料ではMIPS[2]とRISC-V[4]を参考に
以下のように構成する。

条件分岐命令(B: Branch)

- 二つのレジスタの値を比較して真なら分岐

無条件分岐命令(J: Jump)

- 無条件で分岐

どちらの場合も、以下のオプションあり。

- リターンアドレスを保存(AL: And Link)
- レジスタに保存されたアドレスへ(R: Register)

参考文献

- [1] David Patterson, John L. Hennessy, 成田光彰(訳).
「コンピュータの構成と設計 MIPS Edition 第6版 [上]
～ハードウェアとソフトウェアのインタフェース～」, 日経BP, 2021.
- [2] David Patterson, John L. Hennessy, 成田光彰(訳).
「コンピュータの構成と設計 MIPS Edition 第6版 [下]
～ハードウェアとソフトウェアのインタフェース～」, 日経BP, 2021.
- [3] 渡波 郁. 「CPUの創り方」, 毎日コミュニケーションズ, 2003.
- [4] 西山 悠太郎, 井田 健太.
「RISC-VとChiselで学ぶ はじめてのCPU自作
オープンソース命令セットによるカスタムCPU実装への第一歩」, 技術評論社, 2021.
- [5] Takenobu Tani. 「プログラマーのためのCPU入門
CPUは如何にしてソフトウェアを高速に実行するか」,
ラムダノート株式会社, 2023.
- [6] 矢沢 久雄. 「コンピュータはなぜ動くのか 第2版
知っておきたいハードウェア&ソフトウェアの基礎知識」, 日経BP, 2022.
- [7] 馬場 敬信. 「算数で読み解くコンピュータの仕組み」, 技術評論社, 2022.

ライセンスについて

本文書のライセンスはクリエイティブ・コモンズ表示4.0 CC BY 4.0です。

© 2023 クシナダ機巧株式会社

フォントライセンス

- Rounded M+: M+ FONTS LICENSE