

# 計算機科学実験及演習 3

## 機能設計仕様書

提出期限:2019/6/14

提出日: 2019 年 6 月 10 日

グループ番号: 9

1029293806 大山 偉永

## 1 設計を担当したコンポーネントの機能設計仕様 (中間報告からの追加分)

### 1.1 即値加算命令 ADDI

即値加算命令実行のためには既存のコンポーネントに変更を与えた。LD 命令と同様に ADDI 命令の時は  $\text{sign\_ext}(d)$  と  $r[Rb]$  の値を足す必要があるのでロードの時に用いたマルチプレクサ (ロードの時はレジスタから来た  $r[Ra]$  の値ではなく符号拡張した値  $\text{sign\_ext}(d)$  を出力するマルチプレクサ) を流用し  $\text{addi}$  の時にもこれの制御線を 1 に設定するように変更する。そうすることで  $\text{addi}$  のときにも即値と  $r[Rb]$  の和を計算することができるようになる。

### 1.2 ジャンプ命令 JMP

ジャンプ命令実行のためにいくつかのコンポーネントを追加した。図 1 のように、PC に 1 足した値と分岐命令による符号拡張したアドレスが入力のマルチプレクサの出力を入力の一つとし、もう一方は符号拡張した値  $\text{sign\_ext}(d)$  を入力とした  $\text{jump}$  という制御線を持つマルチプレクサを PC の前段に置く。こうすることでジャンプ命令の時は符号拡張した値がそのままアドレスとして PC に入力される。

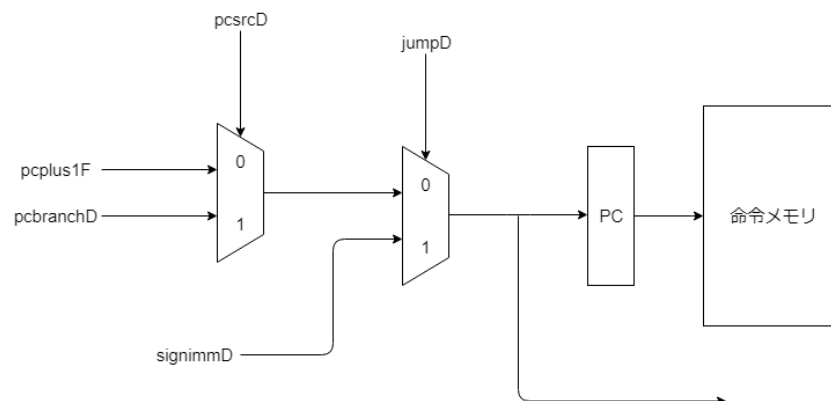


図 1 プログラムカウンタ周辺の回路図

## 2 設計を担当したコンポーネント単体の性能評価

自分はマルチプレクサから ALU 等の細かい部品を作成し、フォワーディングユニット、ハザード検出、制御部、全てのコンポーネントを繋ぐ作業はペアの分担である。このレポートでは自分の担当の中で主要なものである ALU の性能評価を記述する。

## 2.1 ALU

LE 数 389

クリティカルパス 22.3ns

## 3 考察・感想

中間デモからはハードはペアが大目に担当し僕がソートコンテスト用のアルゴリズムの作成とアセンブラの作成を担当した。ハードのデバッグは比較的やりやすかったものの、ソートのアセンブリのデバッグは本当に困難だった。コンパイルエラーのようなデバッグの手がかりが一切ないためシュミレーションで一つずつプログラムカウンタと命令列を見比べてその内容が正しいか確認する作業は本当に骨の折れる作業だった。またソートコンテストについて random.mif をソートしたときのサイクル数は自分たちの一つ上の順位の班のサイクル数より自分たちの班のほうが多く実装できていたのに、昇順済みデータと降順ソート済みデータについては大きくサイクル数を離されていた。あとから昇順済みデータ降順ソート済みデータについてなぜそんなにサイクル数を小さくすることができたのか本人たちに聞いたところ、初めに昇順データ、降順ソート済みデータで場合分けするようにしていたと聞いてとても悔しかった。今回のソートコンテスト専用にチューニングしたアルゴリズムに負けたということである。現に random をソートする場合は自分たちの班の CPU のほうがサイクル数は少なく済んでいたのでもっと悔しかった。ハード面ではそもそもそこまで自分たちの班のものは優れていないので高い順位を目指すことはできなかったが、ほかの班と同様にソートコンテスト専用に対応したアルゴリズムで挑戦していたおそらく 1ms はきれたと思われる。(結果は 6 位で 1.8ms だった) せめてほかの人たちがそのようなずるともいえるアルゴリズムでしていることを先に知っていればもう少し改善できたのにととても悔やまれる。自分たちの班は fmax 値が 74Mhz で実験を終えたがこれでもタイミング制約をぎりぎりまで詰めた結果であり最高周波数が 170Mhz とかの班は本当にどうやったらそんな値になるのか知りたい限りである。そもそも自分たちの班の全体のクリティカルパスは WB ステージからレジスタに書き込むまでの部分であり、その WB ステージの値のレジスタ書き込みのタイミングも PLL を用いてクロックの周期を 202.5 度ずらしたものを使用した(このタイミングでのレジスタ書き込みが一番クリティカルパスとしての遅延時間を減らすことができた)これ以上の改善はしようもなかった。なぜならレジスタへの書き込みはどうしても同じフェーズで行うしかないからである。ゆえに 170MHz のような値を出している班は本当にどのようなになっているのか不思議である。

この cpu 実験(特にソートコンテストで)でハード面ではないもののアルゴリズムのすごさが再確認された。また計算量も今まで深く意識したことがなかったがこれを機にコーディングで計算量を意識するきっかけとなった。もちろんハード面でも今回は静的分岐予測を実装した CPU だったのでそのようなハードの制約も意識したコーディングも身につくきっかけになったと思う。

もう少し時間があれば投機実行などさらなる改善を施したかったが時間の制約上これ以上は厳し

かった。また自分はペアに恵まれたものの本当にペアに依存する実験で友人はやる気のない人と組まされておりもう少し別の実験のやり方があるように思える。グループの人数を東大のように増やすなど。