

計算機科学実験3HW 最終レポート

橘大佑

1029-31-6811

2019 年度入学

2021/05/27

1 設計を担当したコンポーネントの機能設計仕様

中間レポートで設計した SIMPLE に加えて即値オペランドの強化を行った。具体的にはレジスタに数値 d を加算する際に SIMPLE では 2 命令必要としたが、演算命令のうち、 d フィールドを使わないシフト以外の演算命令 (ここでは即値ロード／無条件分岐命令の $op2$ が 001 の、ドントケアの命令) に対し、レジスタ Rd と d フィールドで指定した即値を演算させることで、1 命令で実行できるようになった。そのために、controlunit の改良を行った。以下で controlunit の仕様を説明する。ほとんど中間レポートと同様であるため、変更点 (内部仕様) のみ記す。

1.1 controlunit の変更点

16 ビットの入力 $irin$ の上位 5 ビットが 10001 のとき (SIMPLE では reserved となっていた) に、ALU(演算装置) で 2 つの入力の加算を行うように case 文に追加した。また、その場合の出力も加算命令 (ADD) の時の出力とすべて同じである。

2 controlunit の性能

controlunit の論理要素、レジスタ、ピンはそれぞれ 59,15,41 個であった。図 1 に controlunit の回路図を示す。

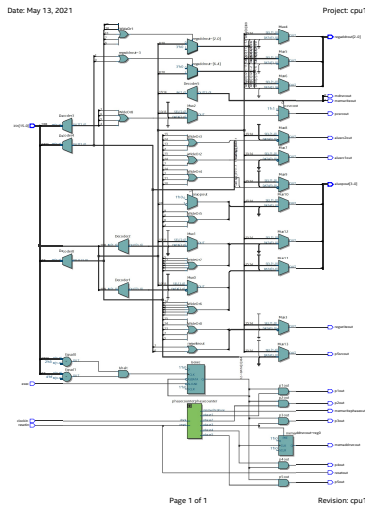


図 1: controlunit の回路図

3 考察

プロセッサの拡張としてはグループで p1 と p5 の並列実行と即値オペランドの強化を行った。p1 と p5 の並列実行は、拡張前は p4 を dr の値が素通りしていたので、pc の値を分岐で更新するときは mdr ではなく dr の値を入れるので、p4 が開始する時点で pc の次の値は確定している。したがって、現在 p5 信号で更新している pc を p4 信号で更新するようにすることで実現した。レジスタの更新は mdr の値を使用する時もあるので、考慮が必要であった。即値オペランドの強化は case 文で場合分けを行うだけであったので、実装にそれほど手間はかからず、レジスタに数値を加算する命令に必要であった命令が 2 から 1 になったのでかなり実装の簡単な改良であった。

4 感想

Verilog HDL を用いたプロセッサの設計を通して 2 回後期の授業「計算機の構成」で学んだプロセッサの仕組みを体系的に学ぶことができた。LED に 4 ビット 10 進カウンタの段階でかなり苦労したので、プロセッサの設計をできるか不安だったが、ペアに助けてもらい設計を進めることができた。反省点としては自分は演算装置や符号拡張、フェーズカウンタなどの小さいパーツの設計を主に担当したので、それらのパーツを用いた各 1 から 5 のフェーズやマルチプレクサ、controlunit はすべてペアが設計してくれ、メモリやレジスタへの書き込みをどの段階で許可するかどうか、などのタイミングも重要となる設計にあまり関われなかったことである。最終的にペアの実装をトレースしてとりあえず理解はしたが、少しでも実装に協力できればよかったと思う。