

## レビュー会

### 1. 目的

データパスの設計を確認し、問題点が存在しないか確認する。問題点が訂正された後に、次の工程に進む。「後工程で問題が発見され手戻りが発生することを防ぐ」ことを目的とする。設計のリファインはあくまでも「おまけ」とする。

### 2. 方法

#### (1) 役割分担

- |          |             |
|----------|-------------|
| a. 司会    | (重村)        |
| b. 資料作成者 | (レビューを受ける班) |
| c. レビュアー | (その他の班)     |

#### (2) チェックするポイント

- a. 組合せ回路と順序回路は区別できている。  
(順序回路にクロック入力があることが分かる目印などを付ける)
- b. 命令セットアーキテクチャに必要な全てのレジスタとフラグが存在する。  
(G0, G1, G2, SP, PC, C, S, Z)
- c. メモリとのインタフェースが適切である。  
(アドレス、データ(、制御)) バス
- d. 命令フェッチが可能である。  
(1バイト命令、2バイト命令、PCの更新、フェッチした命令の置き場所)
- e. JMP/CALL/RET 命令の実行が可能である。  
(JMP、CALLにもアドレッシングモードがあるので要注意)  
(SPの操作に関してはjが担当する。)
- f. LD命令、計算命令(ADD ~ XOR)の実行が可能である。  
(SPのことを忘れていないか?)
- g. 全アドレッシングモードで e. f. が可能である。
- h. ST命令の実行が可能である。  
(アドレッシングモードがあるのでその点にも注意)  
(アドレスとデータが同時に指定できる必要がある)  
(SPのことを忘れていないか?)
- i. シフト命令(特殊な演算命令)の実行が可能である。

- j. PUSH/POP 命令の実行が可能である。
- k. 組み合わせ回路だけからなるループが存在しない。
- l. 長い組合せ回路のパスが存在しない。

ALU をデータ・アドレス兼用にした場合のアドレスデータは、以下のパスを通過する。ここまでに 10ns 以内に収まりそうか？

(IR または ステート)→制御回路→読み出しレジスタ変更→ALU→メモリ

命令フェッチ, PC 周辺

PC++,

【PC→メモリアドレス, メモリデータ→IR】,

【PC→メモリアドレス, メモリデータ→命令第2バイト】,

JMP, CALL, RET 命令

【SP→メモリアドレス, PC→メモリデータ】(CALL),

【SP→メモリアドレス, メモリデータ→PC】(RET)

LD 命令, 計算命令 (ADD, SUB, ...), ALU 周辺

実効アドレスの計算 (全アドレッシングモード),

【実効アドレス→メモリアドレス, メモリ→GR】(LD),

【実効アドレス→メモリアドレス, メモリデータ→ALU】(計算),

GR→ALU, ALU→GR, ALU の機能 (シフトなど) SP 注意

ST 命令, スタック操作

SP++, SP--,

【実効アドレス→メモリアドレス, GR→メモリデータ】(ST),

【SP→メモリアドレス, GR→メモリデータ】(PUSH),

【SP→メモリアドレス, メモリデータ→GR】(POP), SP 注意

### (3) 手順

- a. 資料の配布、資料の特徴、資料を読むための注意事項説明  
(ディスプレイ上で説明)
- b. レビューアが担当の観点 (チェックポイント) から資料を確認
- c. 資料作成者への質問 (問題点の指摘、反論)
- d. 修正が必要な点の確定

以上