

SH-2

SH-2

- ・ 今後の研究のためにシンプルなプロセッサ必要
- ・ SH-2でいこう
- ・ 命令セットに実装上困難なところがある

SH-2 実装上の難点 (1)

- 論理演算で直接メモリを書き換えられる
 - OR #imm,@(R0,GBR)
 - $(R0+GBR) \leftarrow (R0+GBR) \text{ or } imm$
- 積和命令のアドレッシングモードが変
 - MAC.L @Rm+,@Rn+
- R0だけ特別扱い
 - MOV.L Rm,@(R0,Rn)
 - R0, Rm, Rnを読む必要がある

- 書き込みが2つあるときがある
 - MOV.L @Rm+,Rn
 - $Rm \leq Rm+4$, $Rn \leq (Rm)$

SH-2の難点

- 即値は8bのみ
 - それ以外はPC相対でテーブルを読もう
- 遅延分岐とPC相対が組み合わさる
 - PC相対はNextPCを見ている
 - 遅延スロットのPC相対は分岐先アドレスを起点

現在までの実装

- IキャッシュIF
 - 途切れることなく命令が供給できるようにした
- DキャッシュIF
 - メモリアクセスノーウェイトを実現
 - OR #imm,@(R0, GBR)はD\$で実装する
- 命令デコード
 - 今のところ簡単にデコードする方法はなさそう
 - とりあえず動きそうな実装をした

残っていること

- 実行ステージを書けば動く
- キャッシュを書く必要がある
 - I\$: アドレスを受け取った次のクロックでデータを出すだけ
 - D\$: 論理演算を組み込む、Test and Setの対応など
 - Pentium方式でそのまま外にLOCK#を出してしまおう