

プロジェクト実習III 論理設計

第3,4週「プロセッサ」

今回（と次回）の実習

1. プロセッサの設計（前半）

- レジスタ部
- ALU部

2. プロセッサの設計（後半）

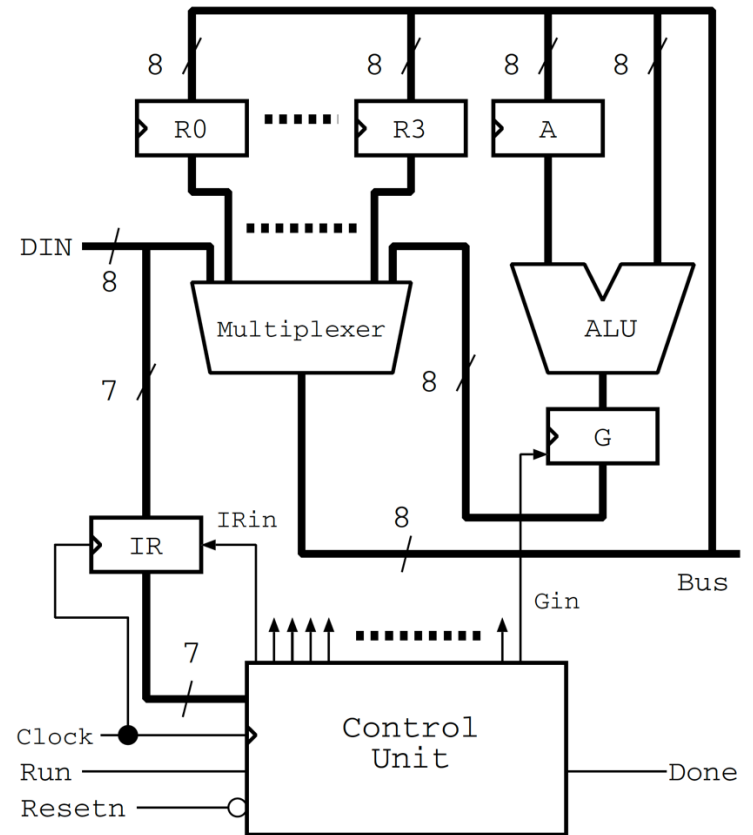
- 制御部

これまでの実習の総まとめを行う

プロセッサの構成

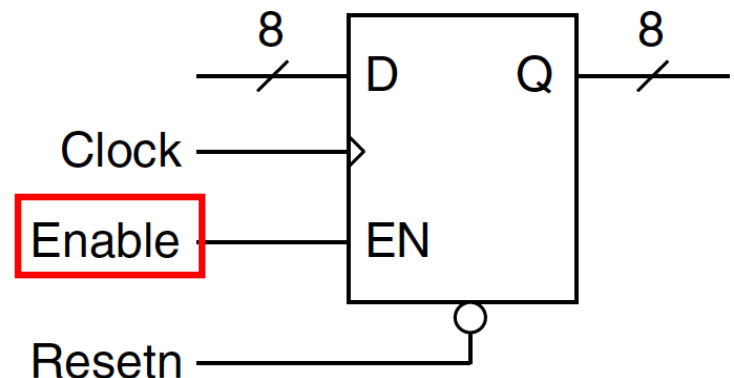
- 8bitレジスタR0～R3
- 算術論理演算ユニット
ALU
- 命令レジスタIR
- マルチプレクサ

これまでの実習で作成したものと基本的に同じ



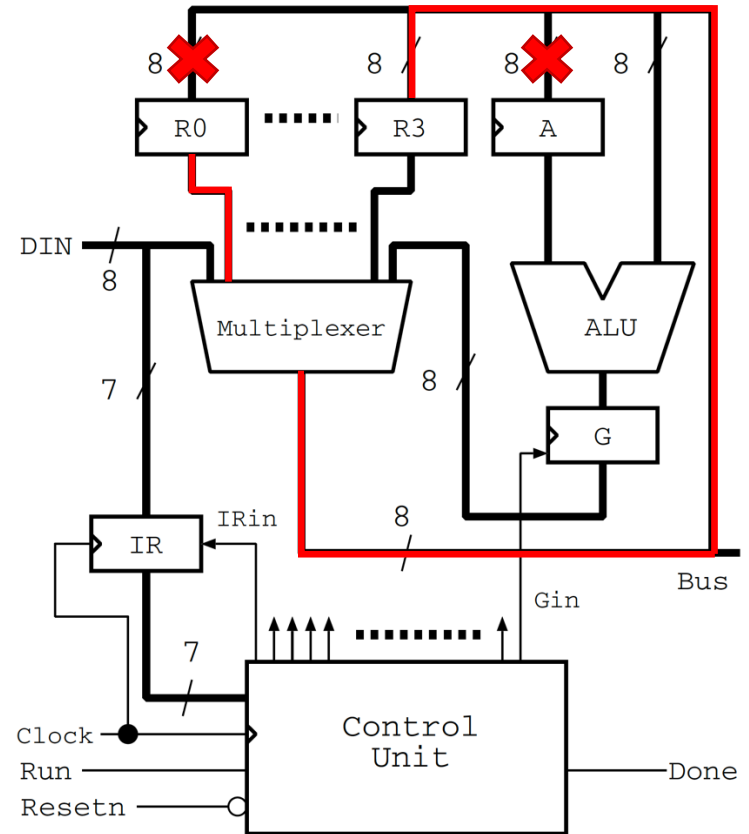
入力制御信号をもつレジスタ

- 入力制御信号**Enable**が1のときのみ、クロックの立ち上がり時に**D**を取り込む
- **Enable==0**ならば、クロックの立ち上がりでも**D**を取り込まない（入力を無視する）



なぜ必要か

- 例えばR0からR3へ値をコピーする場合
 1. R0の出力をマルチプレクサが選択する
 2. R0の内容がバスに流れる
 3. **R3だけ**がバスから信号を取り込むように、入力制御信号を設定 (R3のEnableはOn. R0～R2とAのEnableはOff)



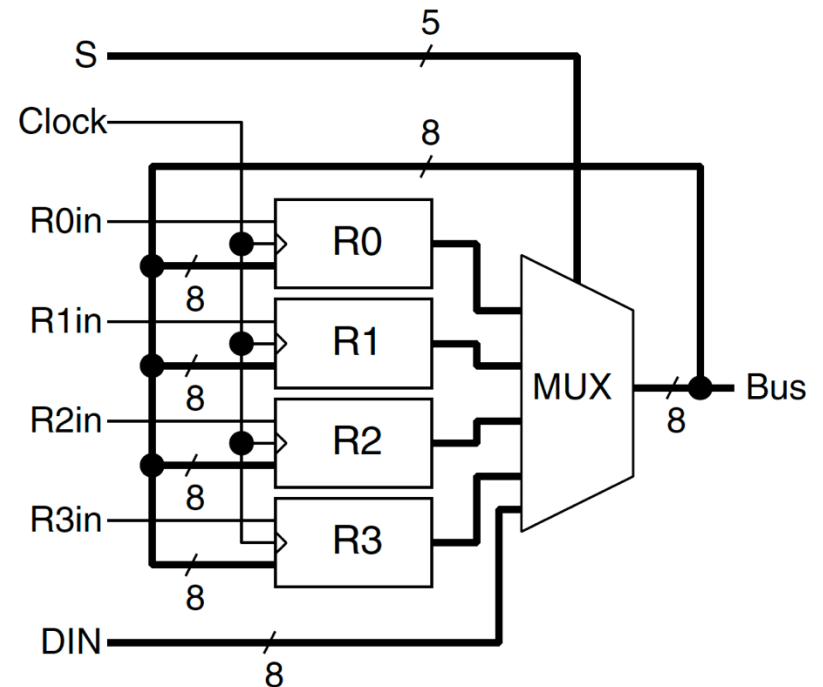
実習の手順

- 実習1
 - レジスタとMUX
- 実習2
 - 実習1 + ALU
 - 前半（第3週）はここまでが目標
- 実習3
 - 実習2 + 制御ユニット

各段階でのコードと動作確認結果を示し、レポートで報告する

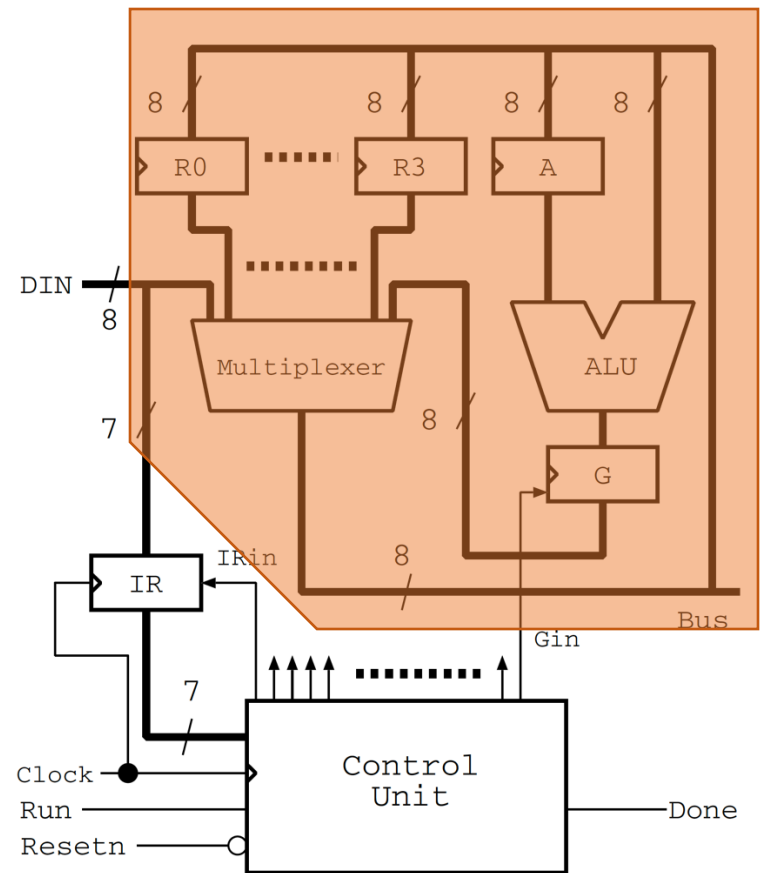
実習1

- レジスタ間でデータをコピーできるか確認
 - DIN→レジスタ
 - レジスタ→レジスタ
- 目標のレジスタにデータが入っているか
- 目標以外のレジスタにデータが入っていないか



実習2

- 実習1の結果に「ALUとA, Gレジスタ」を加える
 - $R^* \rightarrow A$ レジスタ
 - 演算結果 $\rightarrow G$ レジスタ
 - G レジスタ $\rightarrow R^*$
- ALUは第2週「演算回路」の結果を再利用して構わない
- ここまでで全体の上半分が完成



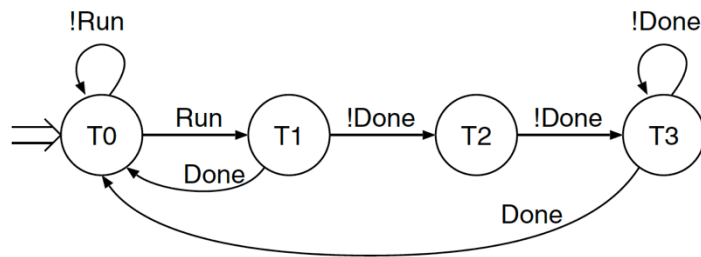
実習3

- マシン命令の動作を実現する
 1. MUXでどの信号を選択するか
 2. どのレジスタの「入口」を開けるか
 3. 上記1,2の操作を「いつ」行うか
- 制御ユニットで上記1～3の制御を行う

命令コード	命令	動作
000	mv Rx, Ry	$R_x \leftarrow [R_y]$
001	mvi Rx, #D	$R_x \leftarrow D$
010	add Rx, Ry	$R_x \leftarrow [R_x] + [R_y]$
011	sub Rx, Ry	$R_x \leftarrow [R_x] - [R_y]$

制御ユニット

- MUXの選択信号
 - 各レジスタの入力制御信号
- を各タイムステップ ($T_0 \sim T_3$) でコントロールする



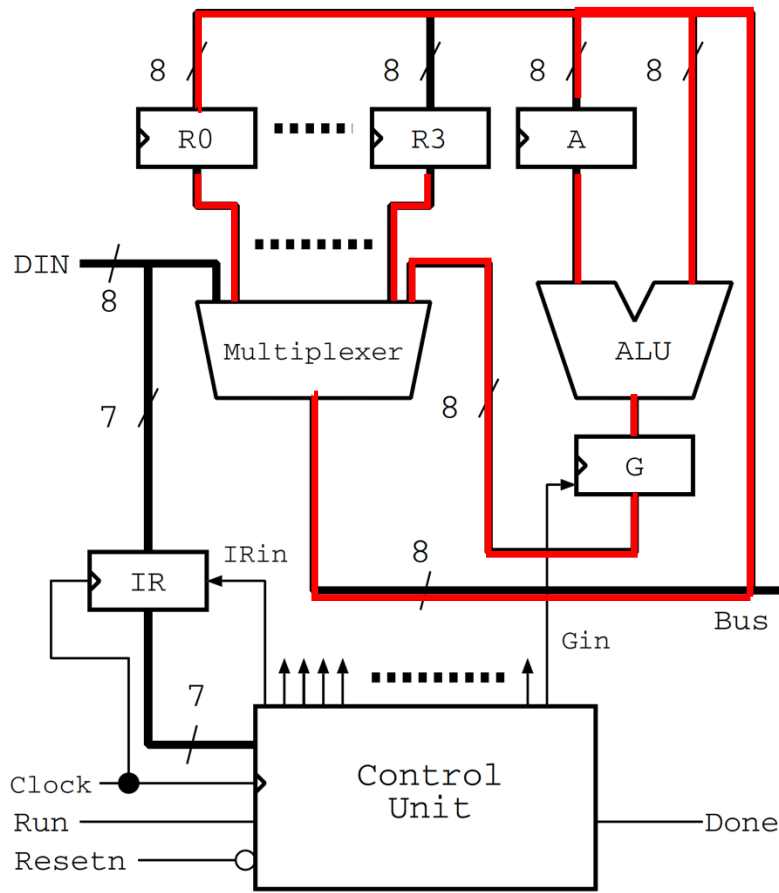
現在のタイムステップ

命令	T_0	T_1	T_2	T_3
mv	IR_{in}	$RY_{out},$ $RX_{in},$ $Done$	-	-
mvi		$DIN_{out},$ $RX_{in},$ $Done$	-	-
add		RX_{out}, A_{in}	RY_{out}, G_{in}	$G_{out},$ $RX_{in},$ $Done$
sub		RX_{out}, A_{in}	$RY_{out},$ $G_{in},$ $Mode$	$G_{out},$ $RX_{in},$ $Done$

現在実行中の命令

ここに載っていない信号は0にする

【例】 add命令の動作



add R0, R3

命令	T_0	T_1	T_2	T_3
mv	IR_{in}	$RY_{out},$ $RX_{in},$ $Done$	-	-
mvi		$DIN_{out},$ $RX_{in},$ $Done$	-	-
add		RX_{out}, A_{in}	RY_{out}, G_{in}	$G_{out},$ $RX_{in},$ $Done$
sub		RX_{out}, A_{in}	$RY_{out},$ $G_{in},$ $Mode$	$G_{out},$ $RX_{in},$ $Done$

制御ユニットの構成方法

- ① 有限状態機械でタイムステップの遷移を管理 ⇒ **順序回路**
- ② 「現在のタイムステップ」と「現在実行中の命令」
(入力) が分かれば, 1に設定すべき信号 (出力) が一意に決まる ⇒ **組み合わせ回路**

①と②は分離して記述するとよい

【注意】 ②で「1にしない信号」には明示的に0を設定すること（そうしないと前回の信号値を記憶しておく必要が生じるため、組み合わせ回路にならない）

レポートについて

- 「プロセッサ」2回目の実習実施後に提出
- 実習1～3：必須
- 演習1, 2：解答必須
- 選択課題1～3：余力があれば実施する．加点あり