

# Datapath

**Attention !**, cette synthèse est encore incomplète. Je vais faire une vidéo et de meilleures explications plus tard.

PC contient une adresse (qui va dans la mémoire d'instructions) Add4 calcule l'instruction suivante L'autre add et ajoute encore pour un branchement Register pour lire et faire des opérations ou écrire l'ALU réalise une opération (comparaison et arithmétique) Sign-extend pour les constantes Mémoire de données (load et store)

Toujours utilisé : PC, mémoire d'instructions, add4

Banque de registre quand l'instruction fait référence à un registre (R dans la greencard) ALU est utilisée si opération (+ - \* / > < << >>>) Mémoire de données (M dans la greencard) Sign-extend utilisée si SignExtImm dans la greencard Shift left 2 et additionneur de droite si branchement (BranchAddr dans greencard)

## Signaux actifs ou inactifs

- RegWrite si R à gauche du =
- RegRead si R à droite du =
- MemWrite si M à gauche du =
- MemRead si M à droite du =

## Multiplexeurs

- premier : On regarde à gauche du = (si rd alors bas, si rt alors haut)
- deuxiem : si opération de deux registres alors haut. Sinon bas pour constante
- troisie : si pas de branchement: haut. Si branchement: bas. Attention il faut connaitre le résultat de la comparaison
- quatre : si on écrit dans un registre : si ALU → bas. sinon (si vient de mémoire) → haut

---

Revision #1

Created 27 April 2023 06:15:13 by SnowCode

Updated 27 April 2023 06:27:26 by SnowCode