HORLOGE

NOMME LE TA MERE

ENS

10 janvier 2025

Instruction set

Proposition 1.1 : Conventions

On choisit une architecture la plus proche possible de celle du cours. La taille d'un mot est de 32 bits. On représente les addresses sur 16 bits.

Instruction	*Encoding*	*Description*
ADD	01 rs1 rs2 00	rs1 <- $rs1 + $rs2
SUB	02 rs1 rs2 00	rs1 <- $rs1 - $rs2
XOR	03 rs1 rs2 00	rs1 <- $rs1 $\hat{}$rs2
OR	04 rs1 rs2 00	rs1 <- $rs1 | $rs2
AND	05 rs1 rs2 00	rs1 <- $rs1 $rs2
ADDIMM	45 rs1 rs2 00	rs1 <- imm | $rs1
NOT	06 rs1 rs2 00	rs1 <- $rs2
LSHIFT	07 rs1 rs2 00	rs1 <- $rs1 « $rs2
RSHIFT	08 rs1 rs2 00	rs1 <- $rs1 » $rs2
LOADROM	09 rs1 rs2 00	rs1 <- M[$rs2]
LOADRAM	14 rs1 rs2 00	rs1 <- M[$rs2]

Instruction	*Encoding*	*Description*
MOVIMM	49 rs1 rs2 00	rs1 <- $imm
STORE	2A rs1 rs2 00	M[$rs2] <- $rs1
MOV	0B rs1 rs2 00	rs1 <- $rs2
NONZERO	82 rs1 rs2 00	rs1 <- $rs2
JMP	80 rs1 00 00	PC += $rs1
JMPIMM	81 00 imm	PC += imm

Structure

Le projet est structuré en plusieurs fichiers. alu.py contient les fonctions relatives à l'ALU. registers.py permet de créer et connecter les registres. Le main gère la lecture du code dans la ROM. Enfin, utils.py contient des fonctions telles que gigamux (permettant de faire la sélection de registres, d'instructions etc)

oaaa

foutre de l'assembleur pr faire une demo

oaaa

dessin du circuit (jen ai un dans mes notes de cours ; faut juste que je le redessine au propre)

oaaa

image carry lookahead wikipedia

fonctionnement clock

Conclusion

- sysnum c a chier
- iris c une grosse pute