

Rapport architecture microprocesseur

HORNERO Baptiste
HERZLICH Raphaël
HELARY Axel

Architecture du processeur

L'architecture du processeur est une architecture proche de l'architecture Risc-V, les instructions sont codées sur 32 bits. On a 16 registres accessibles, numérotés de 0 à 15, chacun de taille 32. On ajoute un register de 32 bits correspondant à la position du pointeur sur la ROM, qu'on appelle P. On se munit de trois flags : ZF (dernière opération a renvoyé 0 ou non), SF (positivité de la dernière opération), OF (overflow de la dernière opération arithmétique) tous de taille 1.

Instructions

Le set d'instructions est le suivant:

Instruction	Encodage	Description	Arguments
NOP	0000 0000	No opération	
ADD	0001 0001	Addition	rs1 rs2
SUB	0001 0010	Soustraction	rs1 rs2
MUL	0001 0011	Multiplication	rs1 rs2
AND	0010 0001	Et logique	rs1 rs2
OR	0010 0010	Ou logique	rs1 rs2
XOR	0010 0011	Xor logique	rs1 rs2
NOT	0011 0001	Non	rs1
SLL	0011 0010	Décalage gauche logique	rs1
SRL	0011 0011	Décalage droite logique	rs1

et :

Instruction	Encodage	Description	Arguments	Description formelle
MOV	0110 0001	Met un registre à la valeur d'un autre registre	rs1 rs2	$rs1 \leftarrow rs2$
MOVI	0111 0001	Met une valeur immédiate dans un registre	rs1 immediate	$rs1 \leftarrow immediate$
CMP	1000 0001	Compare deux valeurs	rs1 rs2	Mise à jour des flags
JMP	0100 0001	Jump à une valeur	immediate	$P \leftarrow immediate$
JNE	0100 0010	Jump si non égal ($ZF = 0$)	immediate	Si $ZF = 0$ alors $P \leftarrow immediate$
JE	0100 0011	Jump si égal ($ZF = 1$)	immediate	Si $ZF = 1$ alors $P \leftarrow immediate$
JGE	0100 0100	Jump si plus grand ou égal	immediate	Si $OF=SF$ alors $P \leftarrow immediate$
LOAD	0101 0001	Lit dans la ram	rs1 rs2	$rs1 \leftarrow R[rs2]$
STORE	0101 0010	Stoque dans la ram	rs1 rs2	$R[rs2] \leftarrow rs1$
LOADFIX	0101 0011	Lit dans la ram	rs1 immediate	$rs1 \leftarrow R[immediate]$
STOREFIX	0101 0100	Stoque dans la ram	rs1 immediate	$R[immediate] \leftarrow rs1$

Si le temps nous le permet

Interface graphique

Implémentation d'une petite interface graphique et des instructions suivantes pour gérer l'interface

Instruction	Encodage	Description	Arguments
PIX	0010 0001	Change la valeur du pixel	rs1 rs2
CLEAR	0010 0010	Met toute l'interface graphique à 0	