

PROJET COMMUNICATION ET FONCTIONNEMENT BAS NIVEAU

R204



23 mars 2024

ARTEAGA MERCHAN JA. Juan – Muhammad hakeem bhatoo

IUT ORSAY

# Table des matières

[Table des matières 1](#_Toc162206794)

[1 CHAMPS 3](#_Toc162206795)

[1.1 ALU 3](#_Toc162206796)

[1.2 DECODEUR 4](#_Toc162206797)

[1.3 MUX2 5](#_Toc162206798)

[1.4 MUX0 6](#_Toc162206799)

[1.5 MUX1 6](#_Toc162206800)

[1.6 JMPPZ 7](#_Toc162206801)

[1.7 JMPN 8](#_Toc162206802)

[1.8 JMPNZ 8](#_Toc162206803)

[1.9 JMPZ 9](#_Toc162206804)

[1.10 JMP 9](#_Toc162206805)

[1.11 DATA 10](#_Toc162206806)

[2 Instructions 11](#_Toc162206807)

[2.1 LOADA 11](#_Toc162206808)

[2.2 LOADSI 11](#_Toc162206809)

[2.3 LOADADI 12](#_Toc162206810)

[2.4 LOADAADRSI 12](#_Toc162206811)

[2.5 LOADBADRDI 13](#_Toc162206812)

[2.6 LOADDIADRSI 13](#_Toc162206813)

[2.7 INCSI 14](#_Toc162206814)

[2.8 DECDI 14](#_Toc162206815)

[2.9 CMPSIA 15](#_Toc162206816)

[2.10 CMPBA 15](#_Toc162206817)

[2.11 JMP- JMPNZ- JMPPZ 16](#_Toc162206818)

[2.12 LOADASI 17](#_Toc162206819)

[2.13 LOADBADRSI 17](#_Toc162206820)

[2.14 LOADADRSIB 18](#_Toc162206821)

[2.15 CMPDIA- CMPB 18](#_Toc162206822)

[2.16 SUBB 19](#_Toc162206823)

[2.17 JMPN 19](#_Toc162206824)

# CHAMPS

**Pour la définition des champs nous avons décidé d’inclure dans les instructions les champs qui ont pour valeur zéro bien que leur valeur par défaut soit également zéro, nous rencontrions des problèmes sur LOGISIM.**

## ALU

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

L’ALU reçoit des instructions de l'unité de commande (Control Unit) et des données depuis les registres. Ensuite, effectue les opérations demandées et stocke les résultats dans les registres appropriés.

## DECODEUR

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

Lorsqu'une instruction est chargée depuis la mémoire dans le registre d'instruction, le décodeur analyse cette instruction pour déterminer quelle opération doit être effectuée (arithmétique, logique, transfert de données, etc.) et sur quels registres ou emplacements mémoire elle doit être appliquée.

## MUX2

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

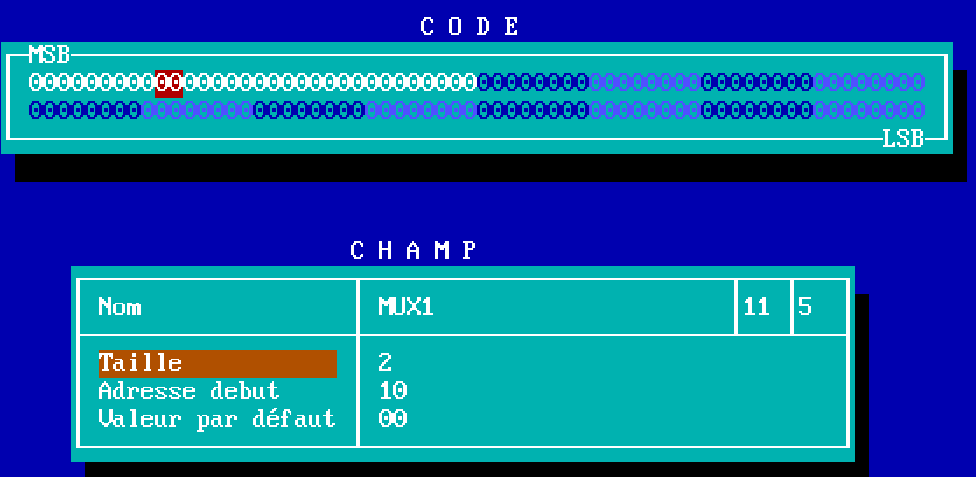
Au lieu d'avoir plusieurs fils pour chaque entrée de données, on peut utiliser un multiplexeur pour sélectionner laquelle des entrées doit être transmise via un seul fil.

## MUX0

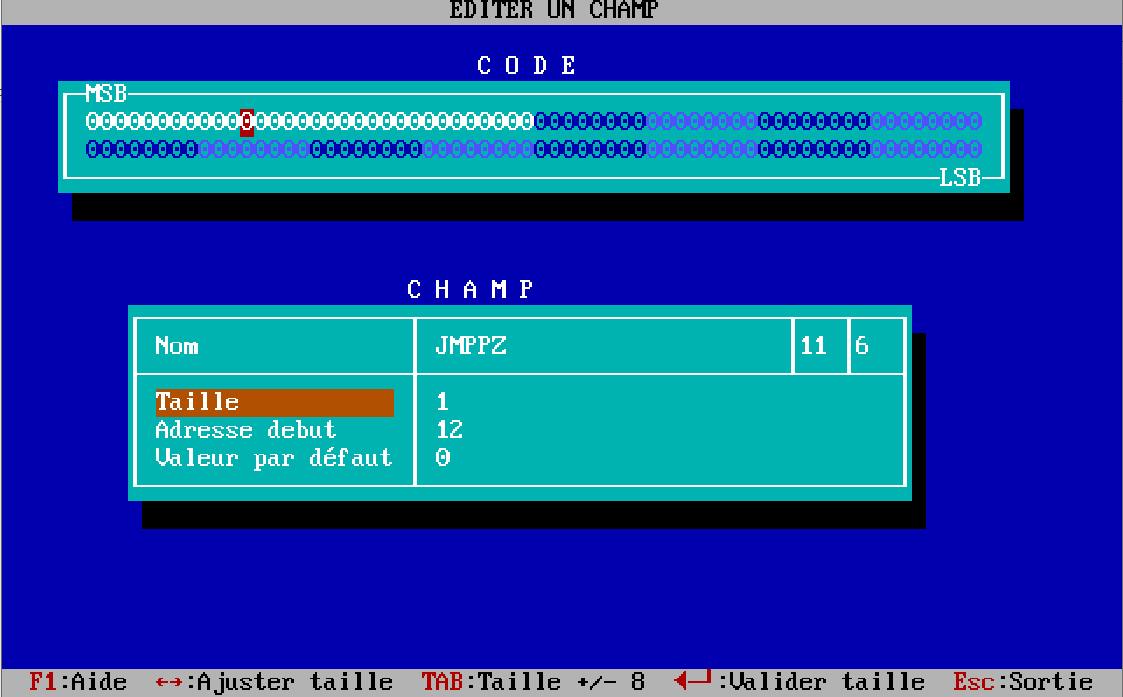
Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

## MUX1



## JMPPZ



"JMPPZ" suggère que le saut se produira si le résultat précédent de l'opération est soit positif, soit nul.

## JMPN

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

Permet de sauter à une adresse mémoire spécifique si le résultat d'une opération précédente est négatif.

## JMPNZ

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

## JMPZ

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

## JMP

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

## DATA

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

# Instructions

**Pour la définition des champs nous avons décidé d’inclure dans les instructions les champs qui ont pour valeur zéro bien que leur valeur par défaut soit également zéro, nous rencontrions des problèmes sur LOGISIM.**

## LOADA

Une image contenant texte, capture d’écran, affichage, logiciel

Description générée automatiquement

La valeur du champ dédié à l’alu est à 0001 ce qui effectue comme opération I0.

Le champ du décodeur est à 001 et on a comme opérande la lecture de la data pour charger dans le registre A la valeur immédiate contenue dans la data.

## LOADSI

Une image contenant texte, capture d’écran, logiciel, nombre

Description générée automatiquement

La valeur du champ dédié à l’ALU est à 0001. Le décodeur est à 011 et on à comme opérande la lecture de la data pour charger dans le registre SI la valeur immédiate contenue dans la data.

## LOADADI

Le champ de L’ALU est à 0010 correspondant à l’opération I1.

Le champ du décodeur toujours à 001.

Le champ du MUX1 est à 10.

LOADADI Copie dans le registre A, le contenu du registre DI

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

## LOADAADRSI

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Le champ de l’ALU à 0010.

Le champ du décodeur est à 001.

Le champ du MUX1 est à 11. L’instruction charge dans le registre A, la donnée de la RAM des données dont l’adresse se trouve dans le registre SI

## LOADBADRDI

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Le même principe que pour l’instruction précédente avec toutefois quelques variations. Charge dans le registre B, la donnée de la RAM des données dont l’adresse se trouve dans le registre DI.

## LOADDIADRSI

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Charge dans le registre DI, la donnée de la RAM des données dont l’adresse se trouve dans le registre SI.

## INCSI

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Incrémente la valeur du registre SI de 1.

## DECDI

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Décrémente de 1 la valeur du registre DI.

## CMPSIA

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Compare SI et A et met à jour les indicateurs en conséquence (l’UAL fait en interne la soustraction SI-A pour mettre à jour les indicateurs).

## CMPBA

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Compare B et A et met à jour les indicateurs en conséquence (l’UAL fait en interne la soustraction B-A pour mettre à jour les indicateurs).

## JMP- JMPNZ- JMPPZ

Une image contenant texte, capture d’écran, ligne, Police

Description générée automatiquement

Effectue un saut inconditionnel à l’instruction donnée.

Une image contenant texte, capture d’écran, ligne, Police

Description générée automatiquement

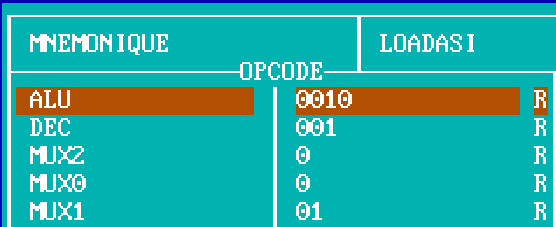
Effectue un saut à l’instruction donnée si le résultat de l’instruction précédente n’est pas nul.

Une image contenant texte, capture d’écran, ligne, Police

Description générée automatiquement

Effectue un saut à l’instruction donnée si le résultat de l’instruction précédente n’est pas négatif (positif ou nul).

## LOADASI



Copie dans le registre A, le contenu du registre SI.

## LOADBADRSI

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Copie le contenu du registre B, dans la RAM des données à l’adresse indiquée par le registre SI.

## LOADADRSIB

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

Copie le contenu du registre B, dans la RAM des données à l’adresse indiquée par le registre SI.

## CMPDIA- CMPB

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Compare DI et A et met à jour les indicateurs en conséquence (l’UAL fait en interne la soustraction DI-A pour mettre à jour les indicateurs)



Compare B et la valeur immédiate qui suit et met à jour les indicateurs en conséquence (l’UAL fait en interne la soustraction B-valeur pour mettre à jour les indicateurs)

## SUBB

Une image contenant texte, capture d’écran, ligne, Police

Description générée automatiquement

Effectue la soustraction B – valeur et stocke le résultat dans B.

## JMPN



Effectue un saut à l’instruction donnée si le résultat de l’instruction précédente est négatif.

# Programmes

Explication des programmes réalisés