

Projet VLSI

Processeur ARM v3 pipeline 5 étages

Sommaire

1) Présentation des étages

2) DECOD

3) EXEC

4) Résultats

Présentation des étages

1) **FETCH**

-> Aller chercher en mémoire une instruction

2) **DECOD**

-> Décoder l'instruction et donner aux étages suivants quoi faire

3) **EXEC**

-> S'occuper de tous les calculs à effectuer

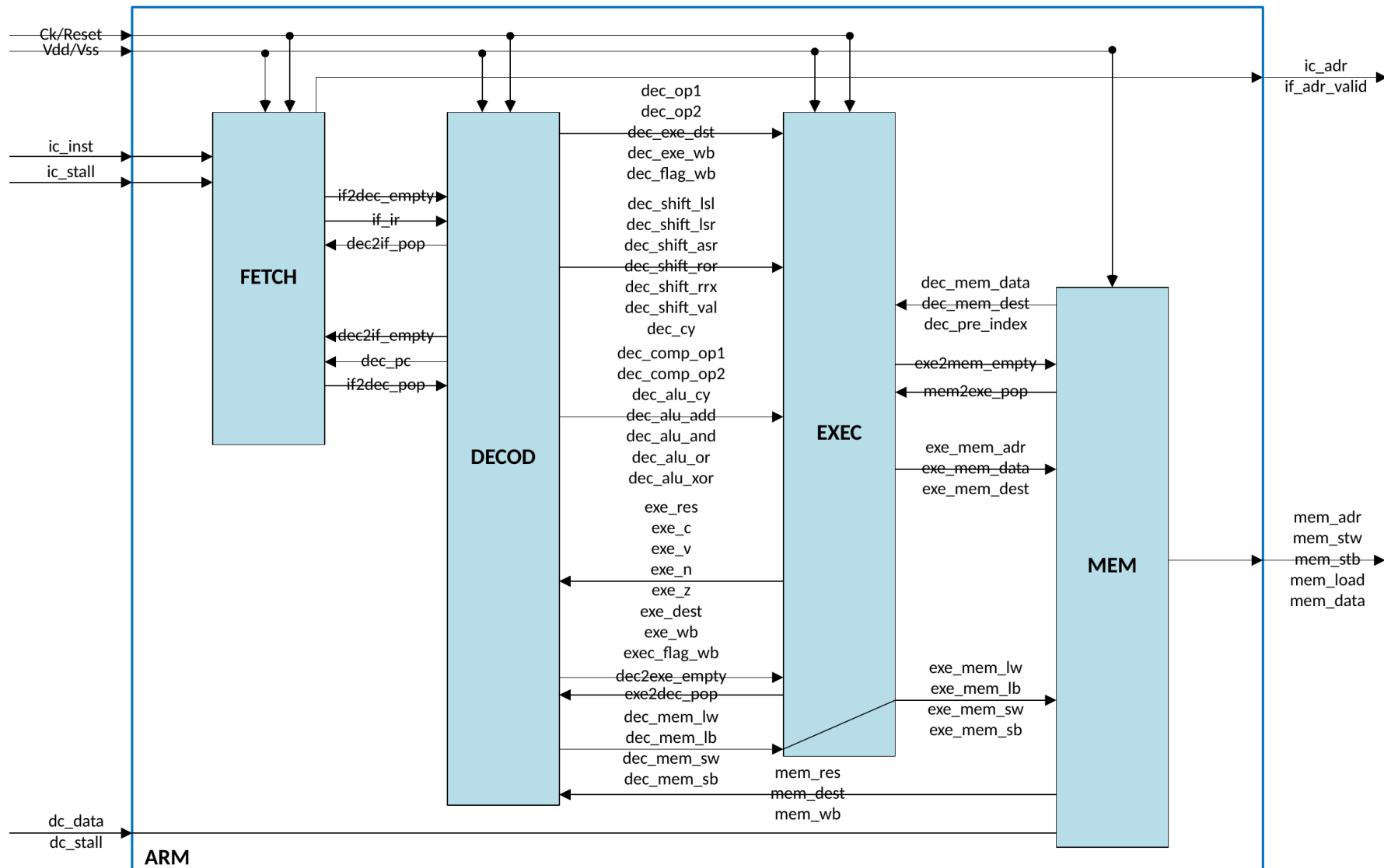
4) **MEM**

-> Aller écrire ou lire en mémoire

5) **WB**

-> Modifier les valeurs contenues dans les registres

Structure complète



Banc de Registre

- **16 registres accessible utilisateur**

- > PC (r15) lr (r14) accessibles comme n'importe quel registre

- **2 ports d'écriture, 4 de lecture**

- > lecture instantanée

- > écriture différée demande un cycle pour invalider la valeur du registre

- **4 flags (C, Z, N, O)**

- > écriture et lecture comme dans un registre

- **Registre PC accessible à tout moment**

State Machine

- **FSM (state machine)**

- > machine à état pour le decodage et le process des instructions

- **FSM_ST**

- > machine à état de type Mealy pour le stockage de l'état de la FSM pour le cycle suivant

FIFO

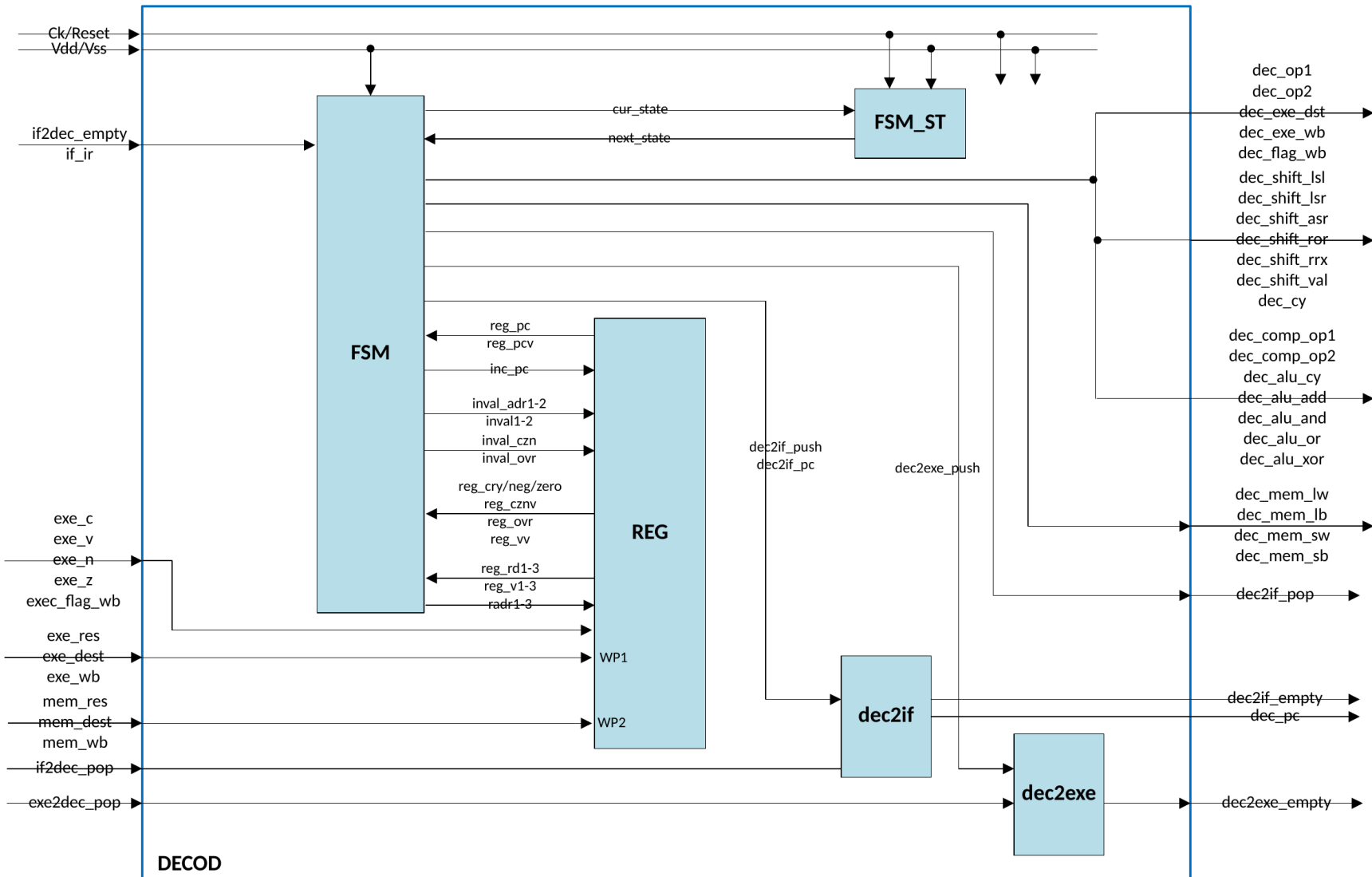
dec2if

-> une fifo de 32 bits contenant l'adresse d'une instruction à transmettre à l'étage IFETCH

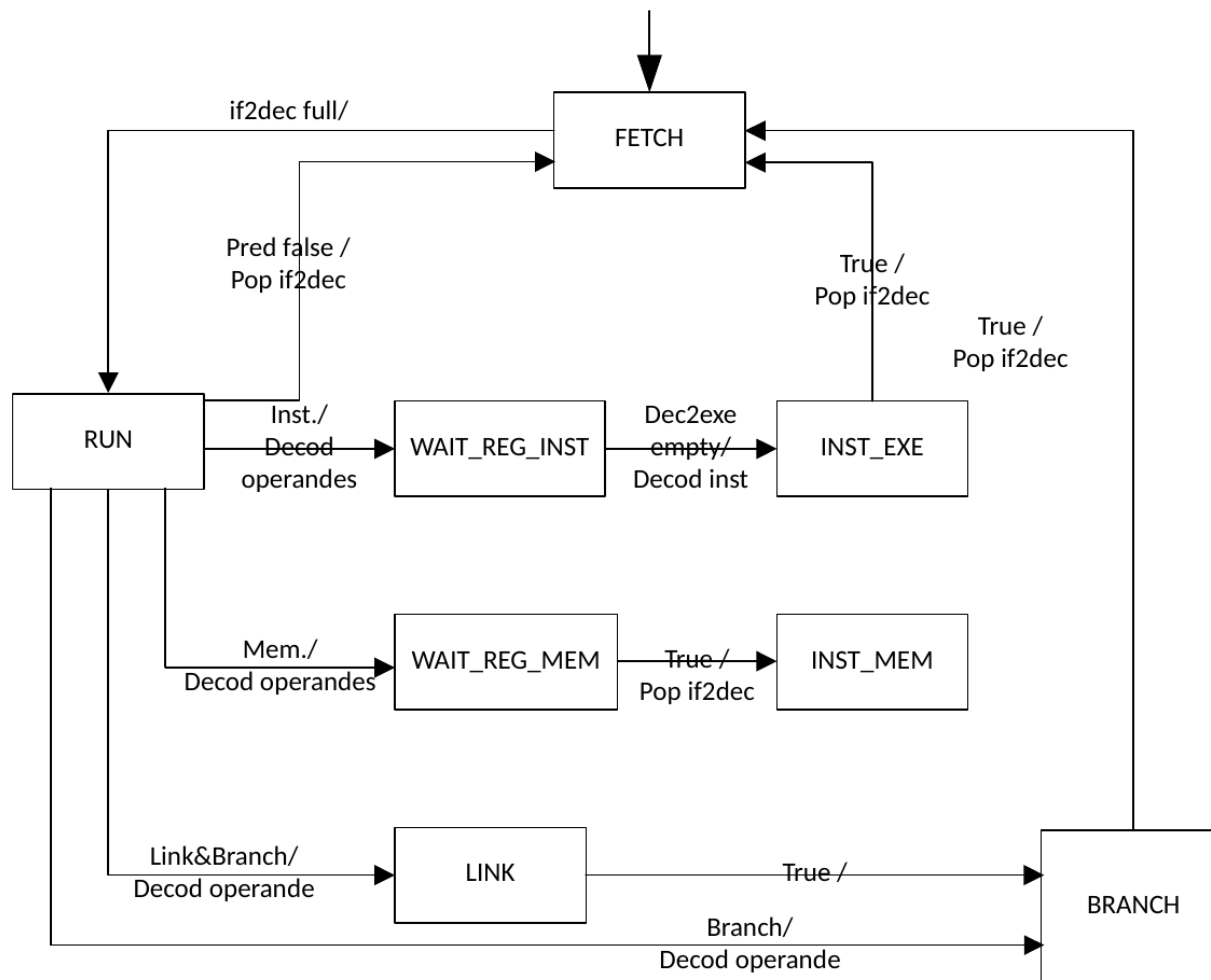
dec2exe

-> une fifo 128 bits contenant RD, RS, op1, op2.

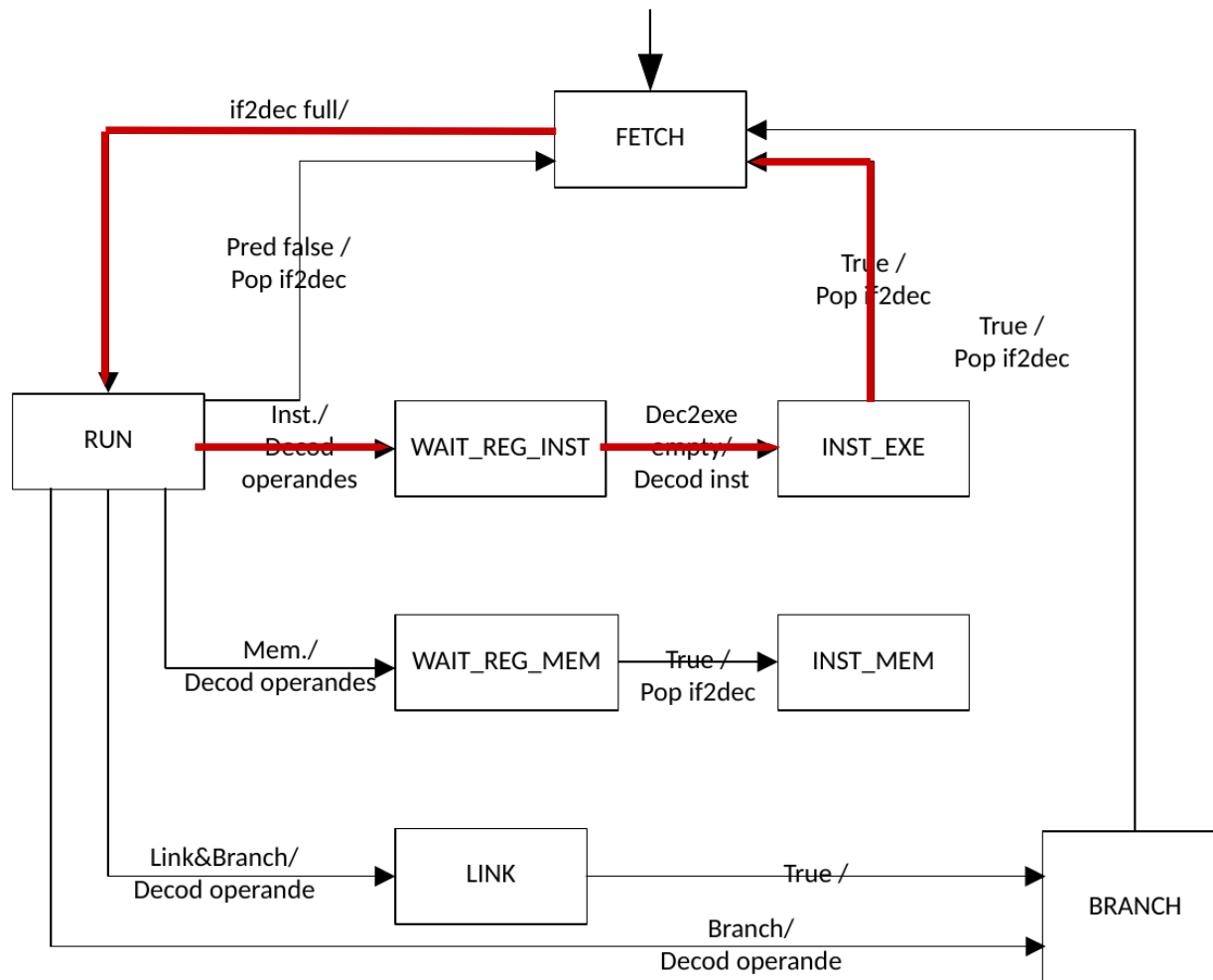
DECOD



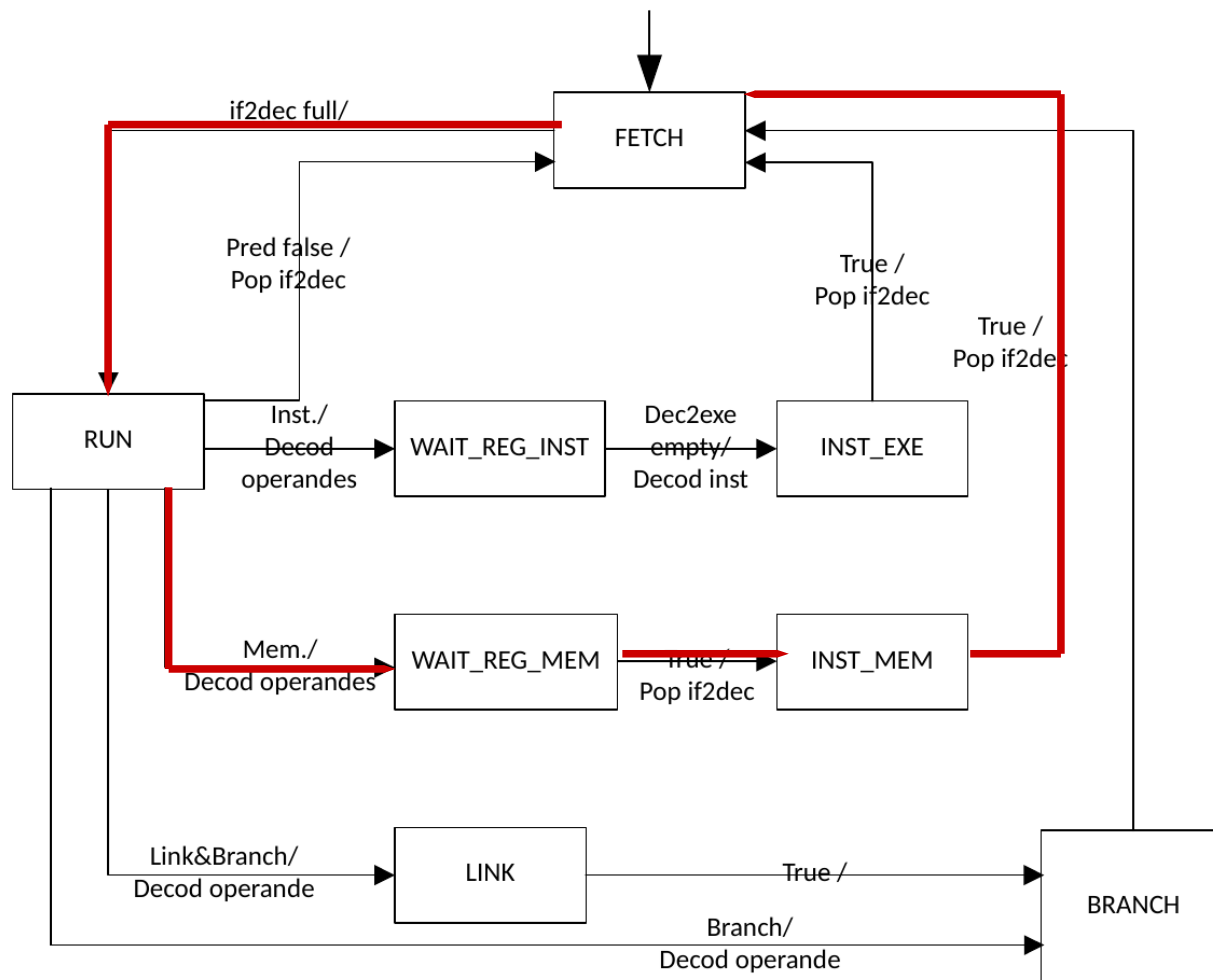
STATE MACHINE



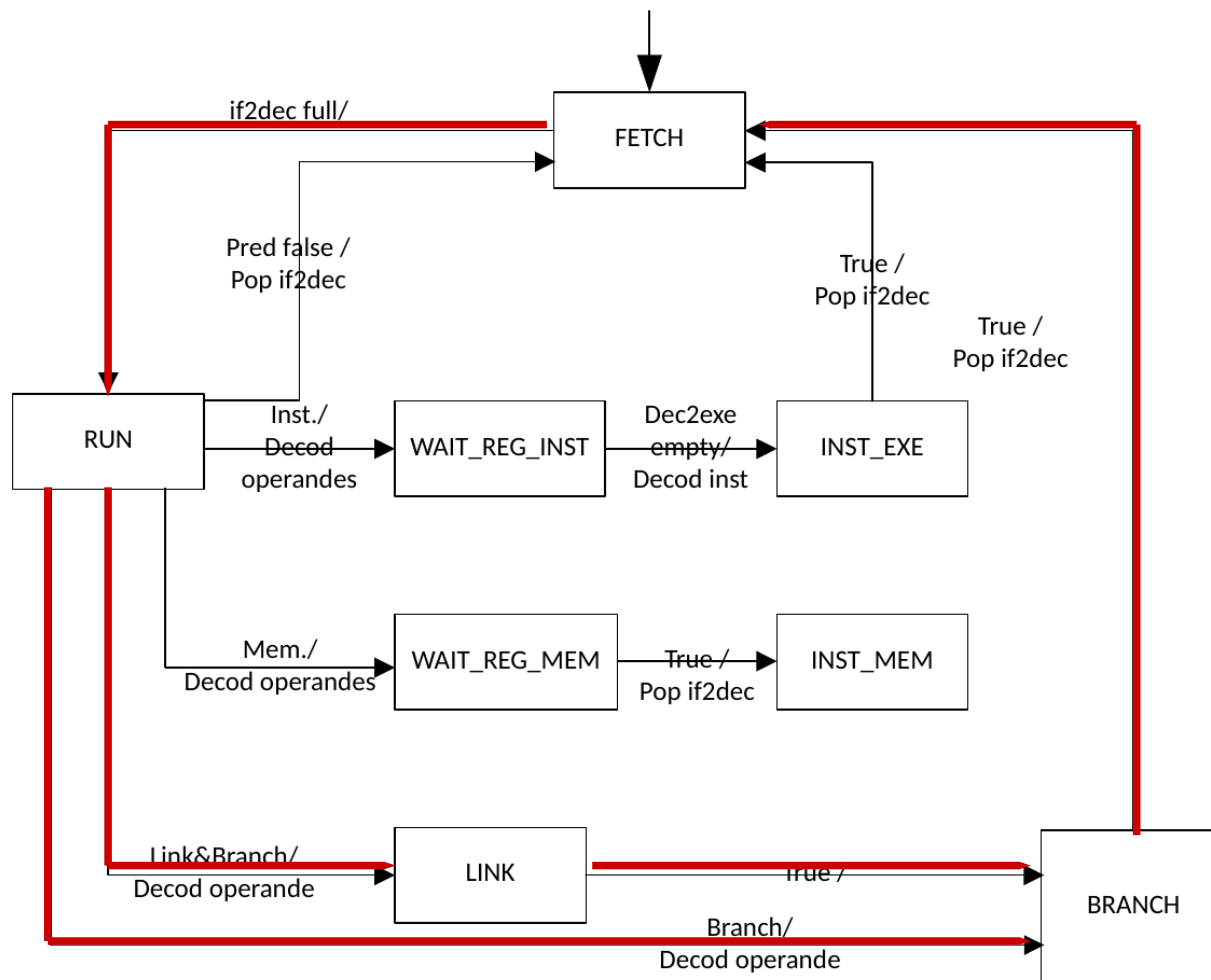
STATE MACHINE INST



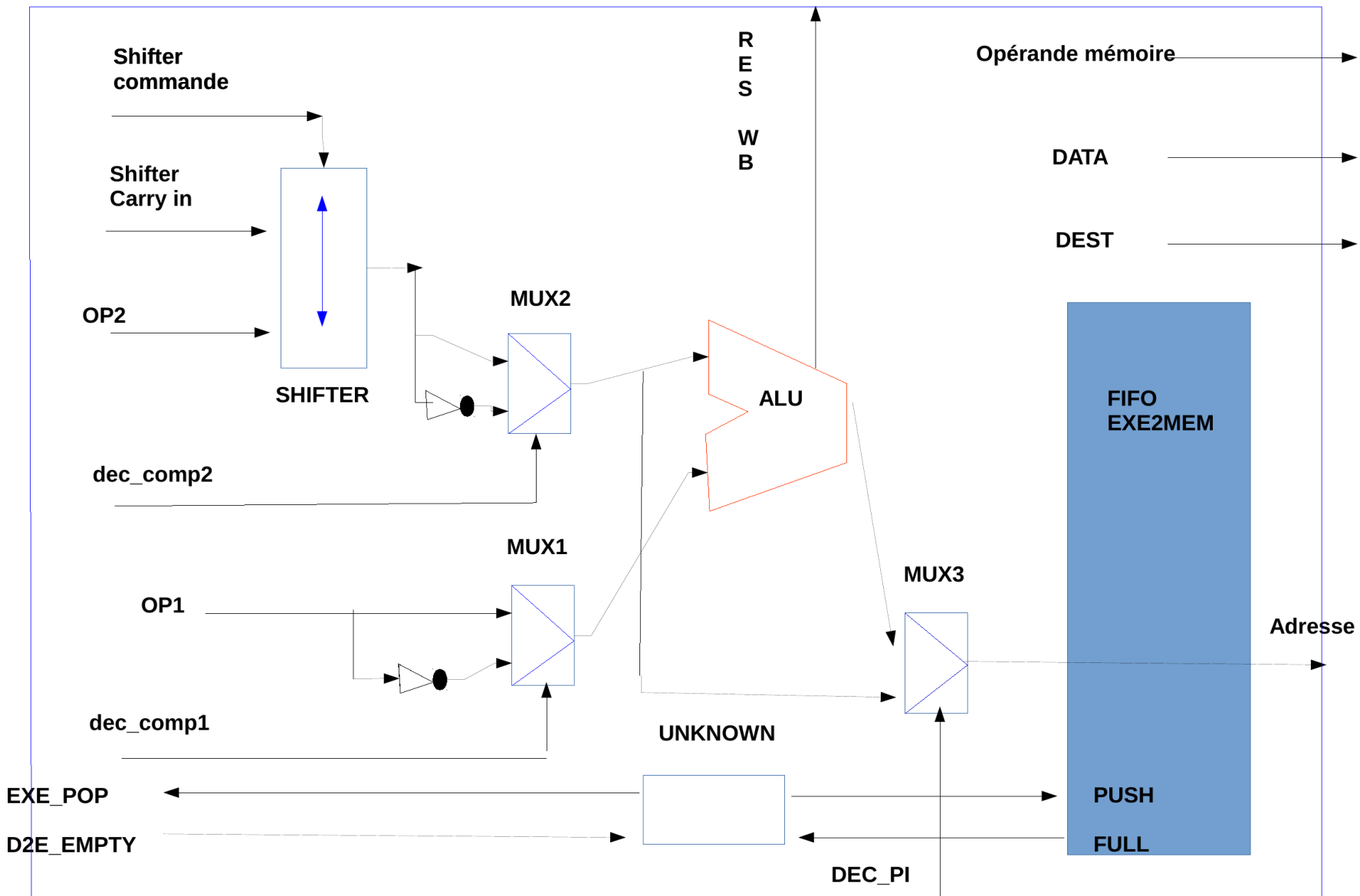
STATE MACHINE MEM



STATE MACHINE Branch/ Branch&Link



EXEC



Résultat

Tous les composants réalisés

REG, EXE implémenté et entièrement testés

DECOD implementation machine à état (*sauf multiples transferts*)

Ne synthétise pas entièrement

Résultat

Tous les composants réalisés

REG, EXE implémenté et entièrement testés

DECOD implementation machine à état (*sauf multiples transferts*)

Ne synthétise pas entièrement