

# Tri à bulle

## Architecture ARM

Le 7 janvier 2025

Par Corentin RICHARD

## Avant-Propos

Ce document sert de support à la mise en compatibilité avec les opérations du tri à bulle du processeur d'architecture ARM mis en place au cours des TP. Nous allons décrypter les commandes pour ensuite décider des éléments à rajouter.

## Sommaire

Avant-Propos .....	1
Sommaire .....	1
Analyse du tri .....	2
Registres Supplémentaires.....	4
R0 .....	4
R1 .....	4
R4 .....	4
Opérations supplémentaires.....	5
Comparaison .....	5

# Analyse du tri

Les différentes opérations effectuées pour l'algorithme du tri à bulle sont listées dans le tableau ci-dessous et une opération ainsi que des registres lui sont associés. Cela nous permettra de déterminer quels modules sont à ajouter dans notre architecture.

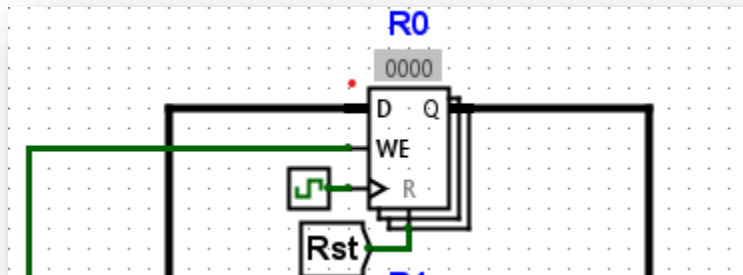
Instruction	Type	Registres utilisés	Forme binaire	Traduction
<b>e007</b>	18	Aucun	1110 0000 0000 0111	b.n #0x1C
<b>0008</b>	-	Aucun	0000 0000 0000 1000	Donnée
<b>000c</b>	-	Aucun	0000 0000 0000 1100	Donnée
<b>000a</b>	-	Aucun	0000 0000 0000 1010	Donnée
<b>0009</b>	-	Aucun	0000 0000 0000 1001	Donnée
<b>000b</b>	-	Aucun	0000 0000 0000 1011	Donnée
<b>0003</b>	-	Aucun	0000 0000 0000 0011	Donnée
<b>0007</b>	-	Aucun	0000 0000 0000 0111	Donnée
<b>0005</b>	-	Aucun	0000 0000 0000 0101	Donnée
<b>2000</b>	13	R0	0010 0000 0000 0000	mov r0, #0
<b>2100</b>	13	R1	0010 0001 0000 0000	mov r1, #0
<b>2202</b>	13	R2	0010 0010 0000 0010	mov r2, #2
<b>6813</b>	7	R3, R1	0110 1000 0001 0011	ldr r3, [r1]
<b>6854</b>	7	R4, R2	0110 1000 0101 0100	ldr r4, [r2]
<b>429c</b>	4	R3, R4	0100 0010 1001 1100	cmp r3, r4
<b>da02</b>	16	Aucun	1101 1010 0000 0010	bge #0x8
<b>6014</b>	7	R4, R1	0110 0000 0001 0100	str r4, [r1]
<b>6053</b>	7	R3, R2	0110 0000 0101 0011	str r3, [r2]
<b>2101</b>	13	R1	0010 0001 0000 0001	mov r1, #1
<b>3202</b>	3	R2	0011 0010 0000 0010	sub r2, r2, #2

<b>3001</b>	3	R1	0011 0000 0000 0001	add r1, r1, #1
<b>2807</b>	2	R0	0010 1000 0000 0111	cmp r0, #7
<b>d1f4</b>	16	Aucun	1101 0001 1111 0100	bne #-0xC
<b>2901</b>	2	R1	0010 1001 0000 0001	cmp r1, #1
<b>d0ef</b>	16	Aucun	1101 0000 1110 1111	beq #-0x22
<b>46c0</b>	5	Aucun	0100 0110 1100 0000	nop

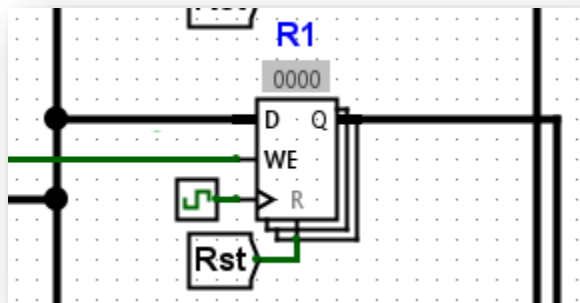
Cette analyse nous permet alors d'en ressortir les registres et opérations à rajouter dans notre processeur.

# Registres Supplémentaires

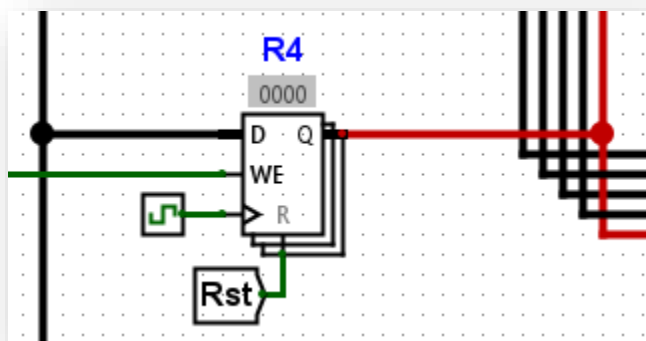
R0



R1



R4



# Opérations supplémentaires

## Comparaison

