



# Projet PARM

## Équipe gens-bons-2

Projet de réalisation d'un micro-processeur  
ARM7 simplifié

# SOMMAIRE

Vue d'ensemble du projet

Fonctionnement de l'ARM7

- Zoom sur l'ALU

- Partie Assembleur

- Partie Contrôleur

- Partie Banc de Registre

Répartition du travail

Analyse des problèmes

Bilan final



# Vue d'ensemble du projet

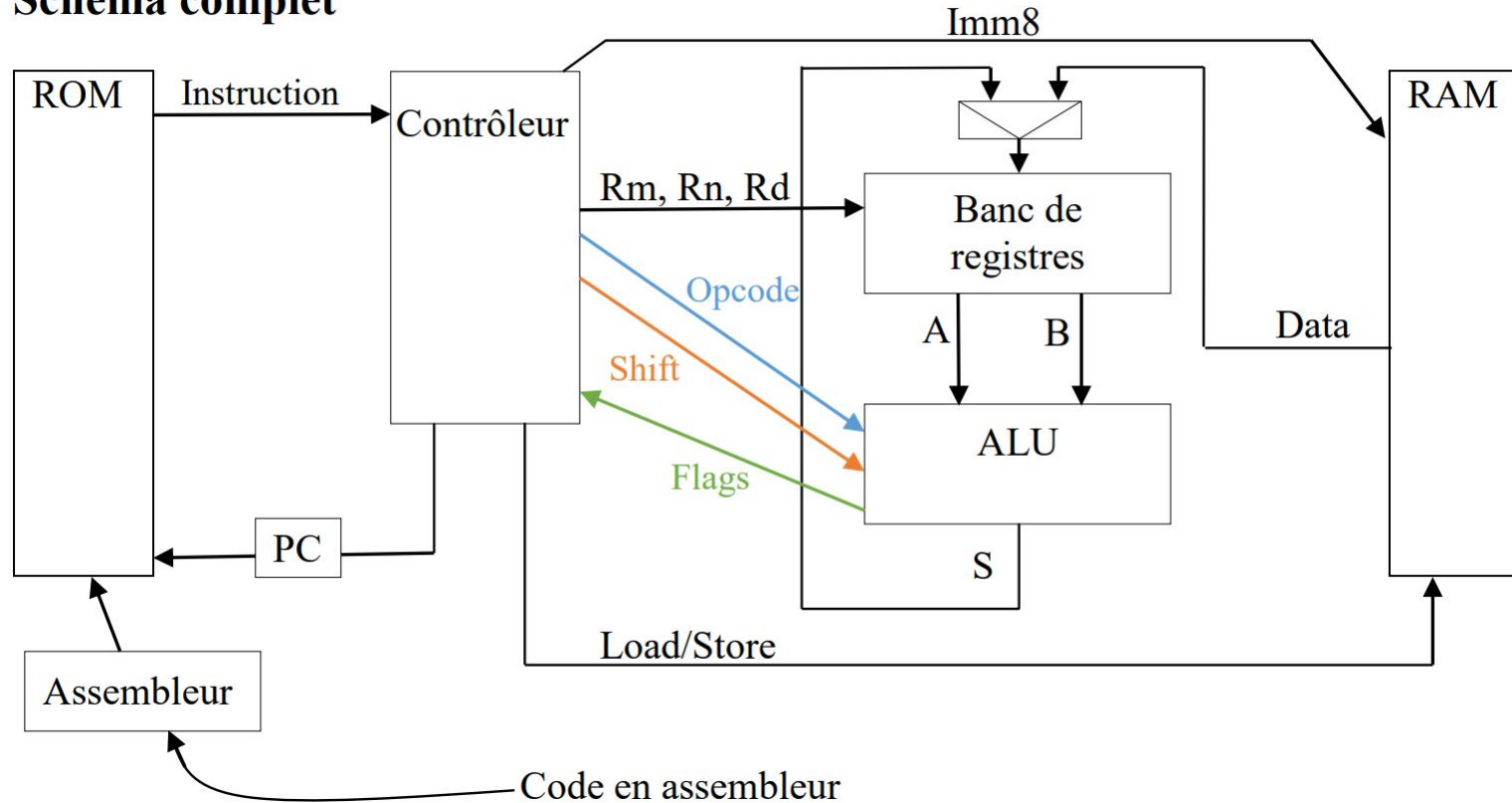
Recréer micro-processeur

- Répartition :

Tâche	Responsable	Participant
ALU	PERES	COLOMBAN
Banc de registre	COLOMBAN	-
Contrôleur	LEFEBVRE	COLOMBAN
Assembleur	LONGORDO	PERES

# Fonctionnement de l'ARM7

## Schéma complet





# Assembleur

But : Transformer un code humain assembleur en un code binaire machine, compiler.

Lecture du fichier

Gestion du segment data

Gestion des labels

Traduction des instructions



# Gestion du segment data

Création d'une liste de donnée contenant le nom de la variable, sa taille et son adresse

Pour chaque besoin on recherche l'adresse dans la liste



# Gestion des labels

Pré-lecture du fichier pour déterminer chaque labels

HashMap

Key : Nom du label

Value : Adresse de l'instruction



# Traduction des instructions

HashMap key value

Key : Nom de l'instruction

Value : Fonction qui traduit

Initialisation de la hashmap de chaque instruction

Exemple : MOV R1 6

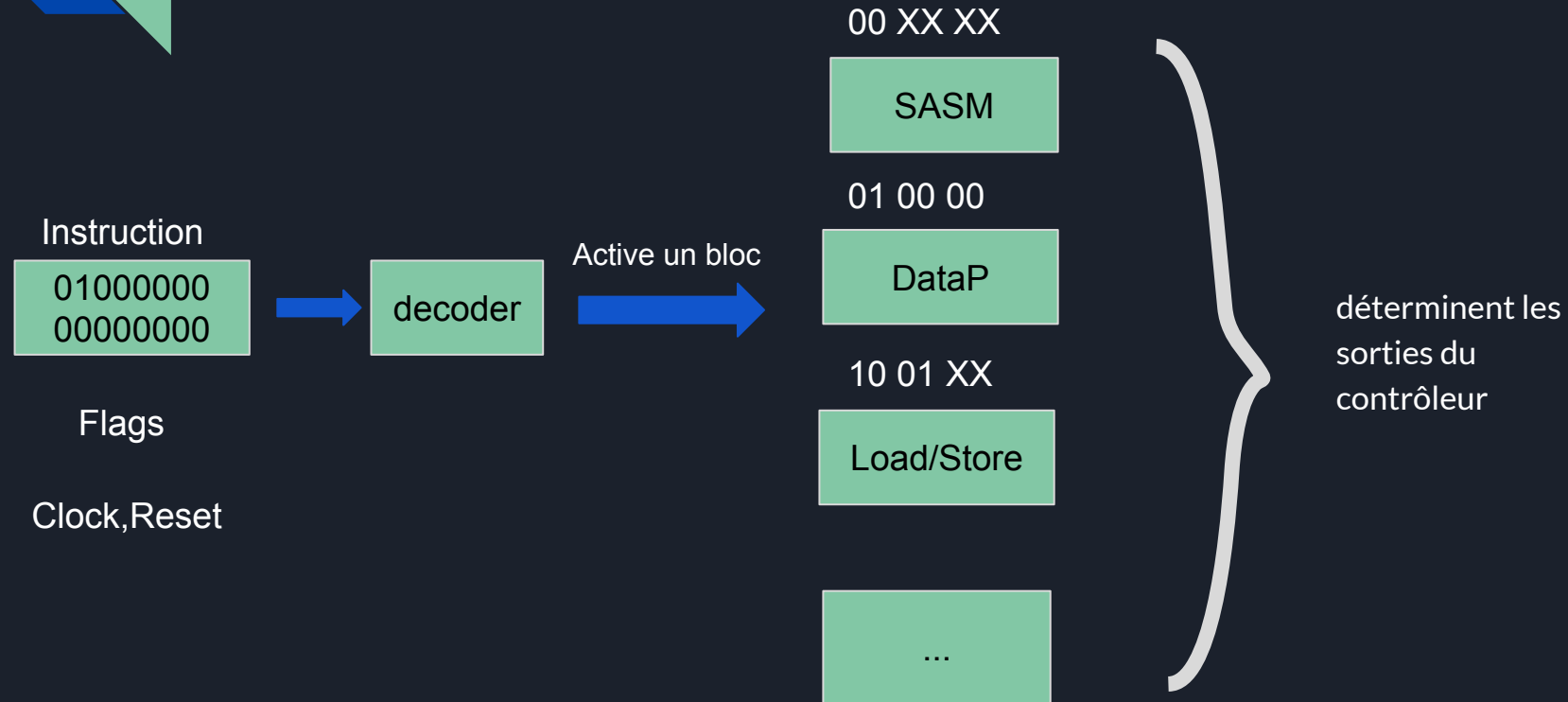
On récupère les valeur qu'on calcule avec ou arithmétique pour former la valeur de retour

Exemple : B else

On récupère dans la hashmap l'adresse du label else



# Contrôleur : Vue d'ensemble



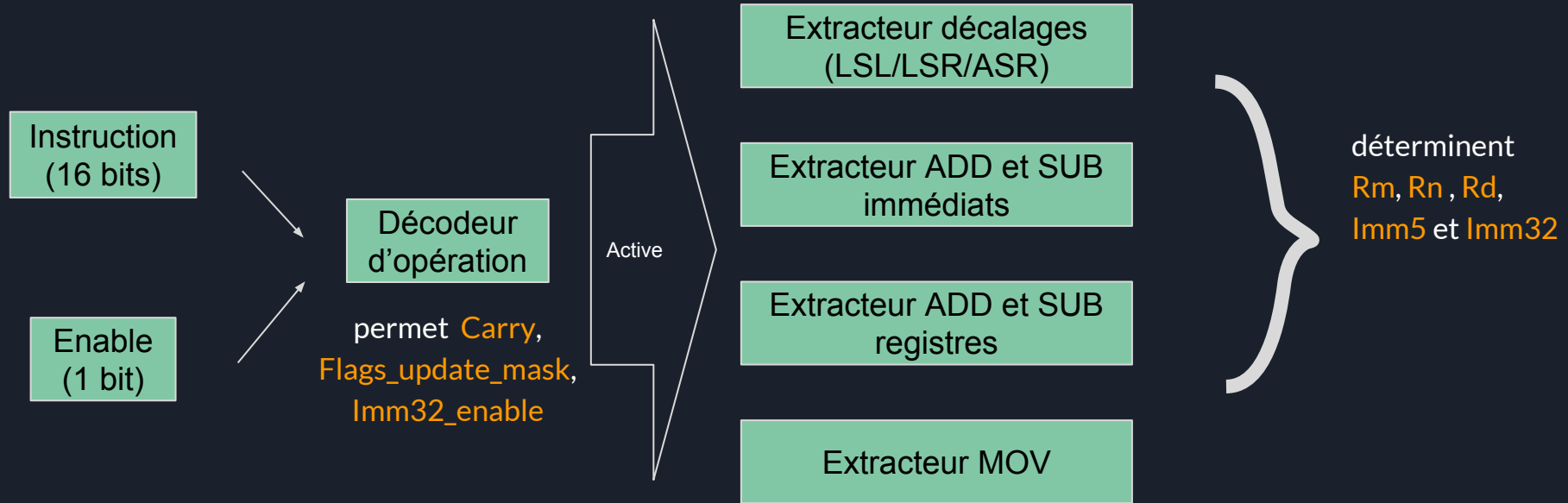
# Contrôleur : Décodeur d'instruction

01000000  
00000000



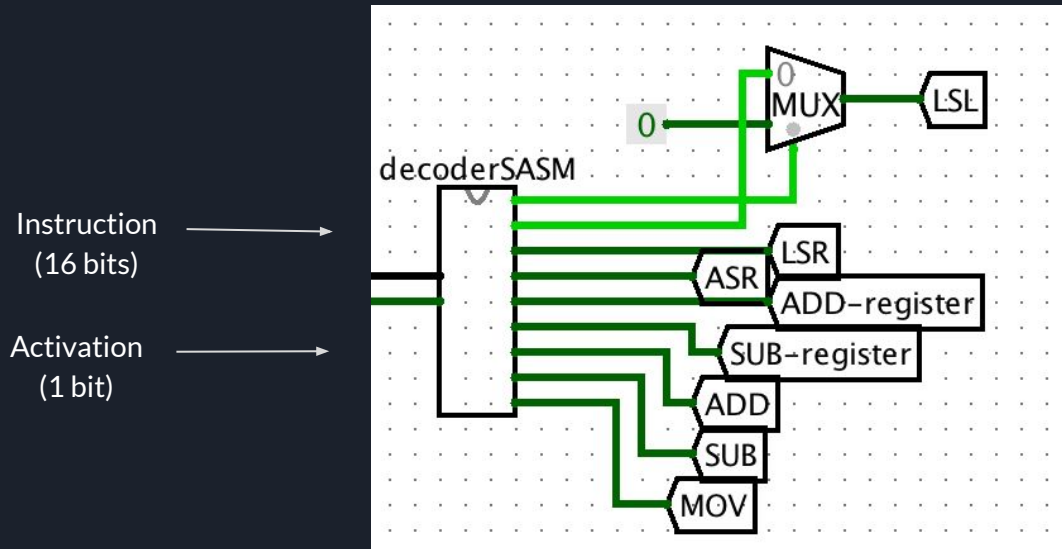
# Contrôleur : Shift, add, sub, mov

## Vue d'ensemble



# Contrôleur : Shift, add, sub, mov

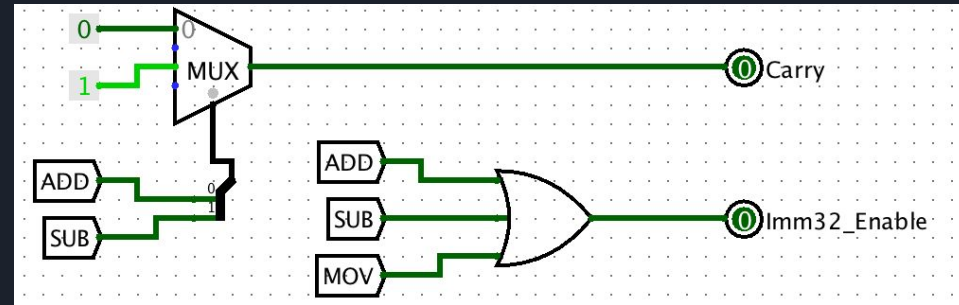
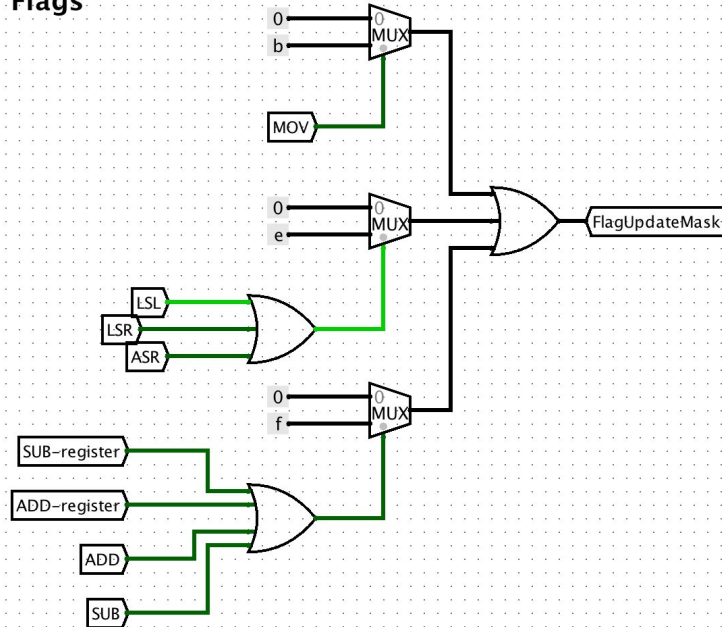
## 01 Déterminer l'opération de l'instruction



La première sortie indique si le composant est désactivé.

# Contrôleur : Shift, add, sub, mov

## Flags



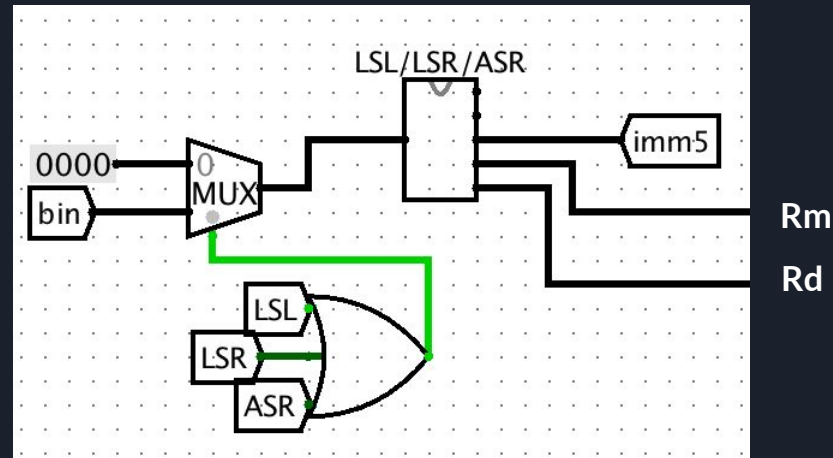
# Contrôleur : Shift, add, sub, mov

## 02 Extraire les données de l'instruction

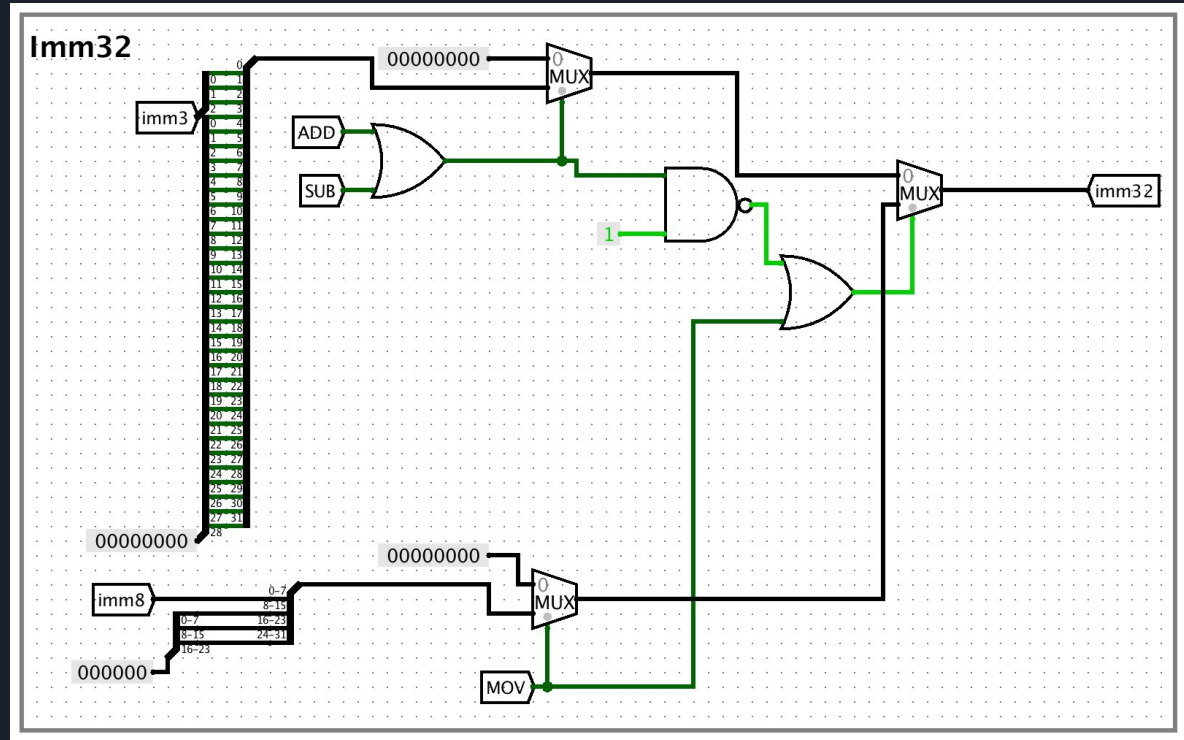
Décalage

Binaire :

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	imm5					Rm			Rd		

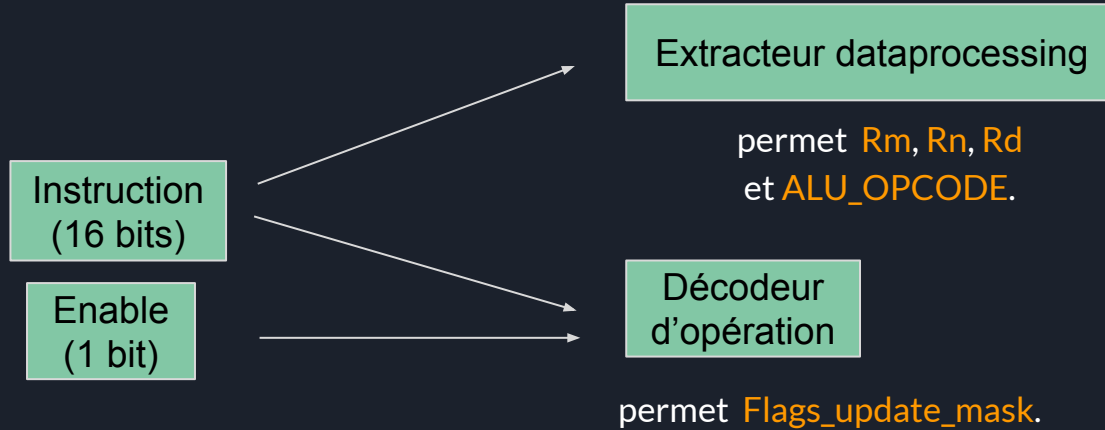


# Contrôleur : Shift, add, sub, mov



# Contrôleur : Data Processing

## Vue d'ensemble



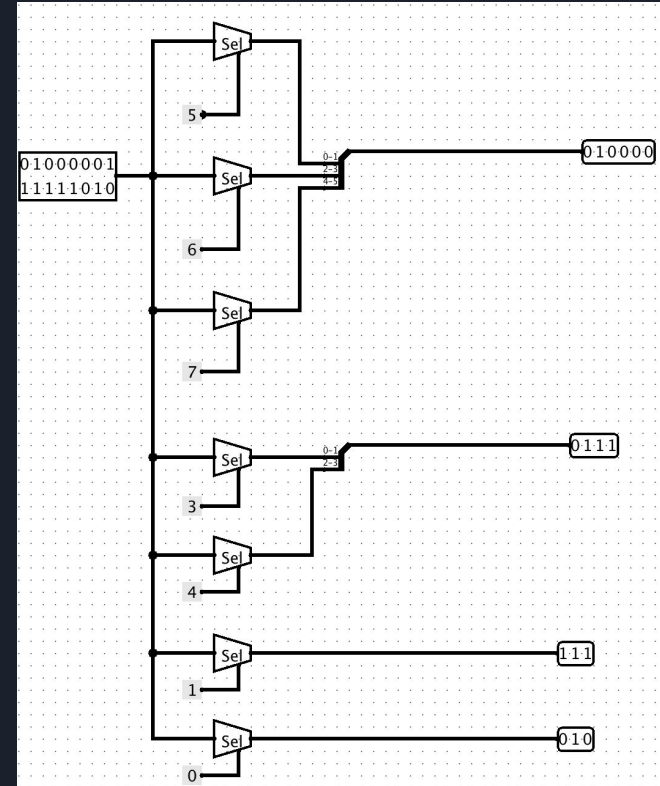


# Contrôleur : Data Processing

## Zoom sur l'extracteur

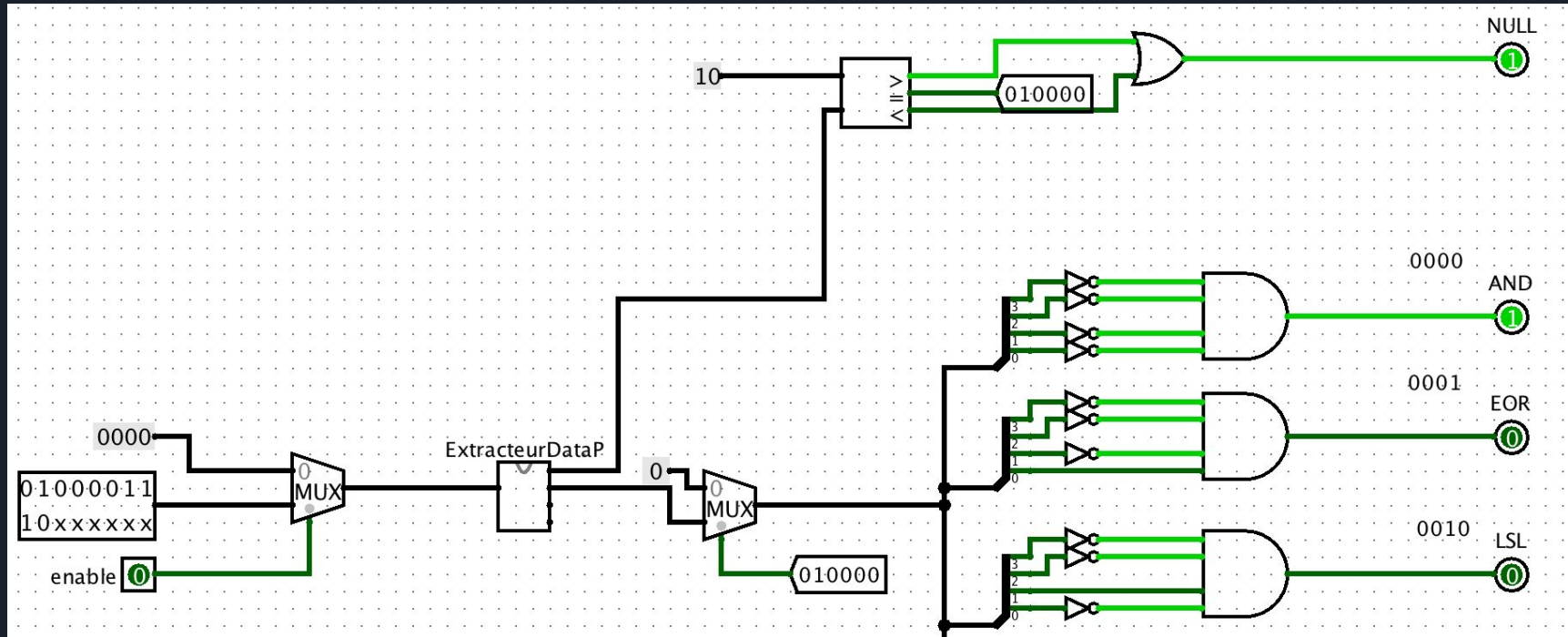
Binaire :

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	0	0	0	0	0	Rm			Rdn		



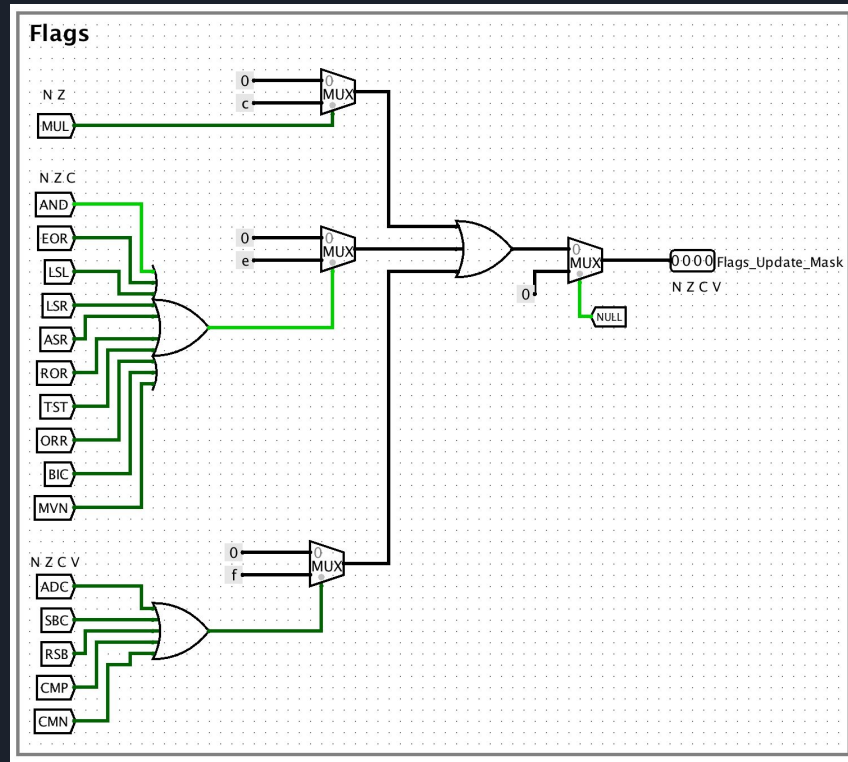
# Contrôleur : Data Processing

## Zoom sur le décodeur

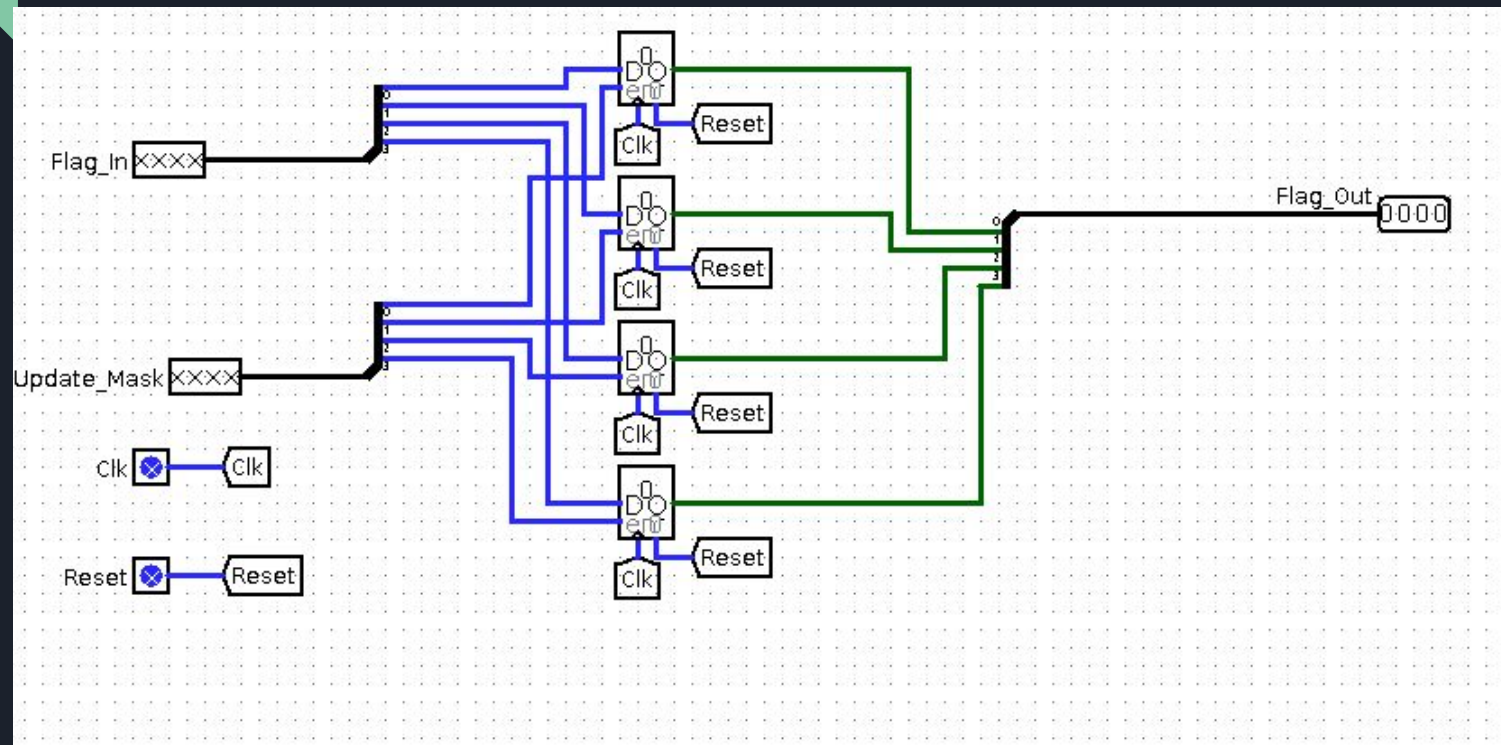


# Contrôleur : Data Processing

## Zoom sur flag update mask



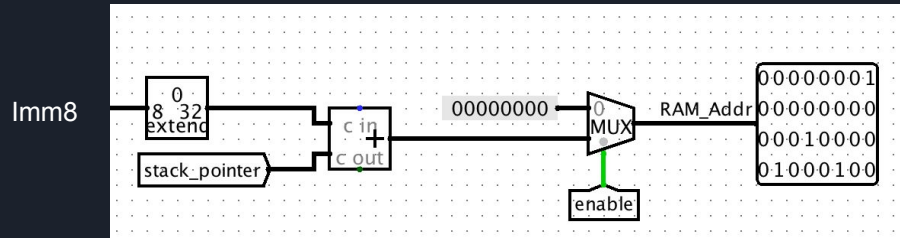
# Contrôleur : Flags APSR



# Contrôleur : Load/Store

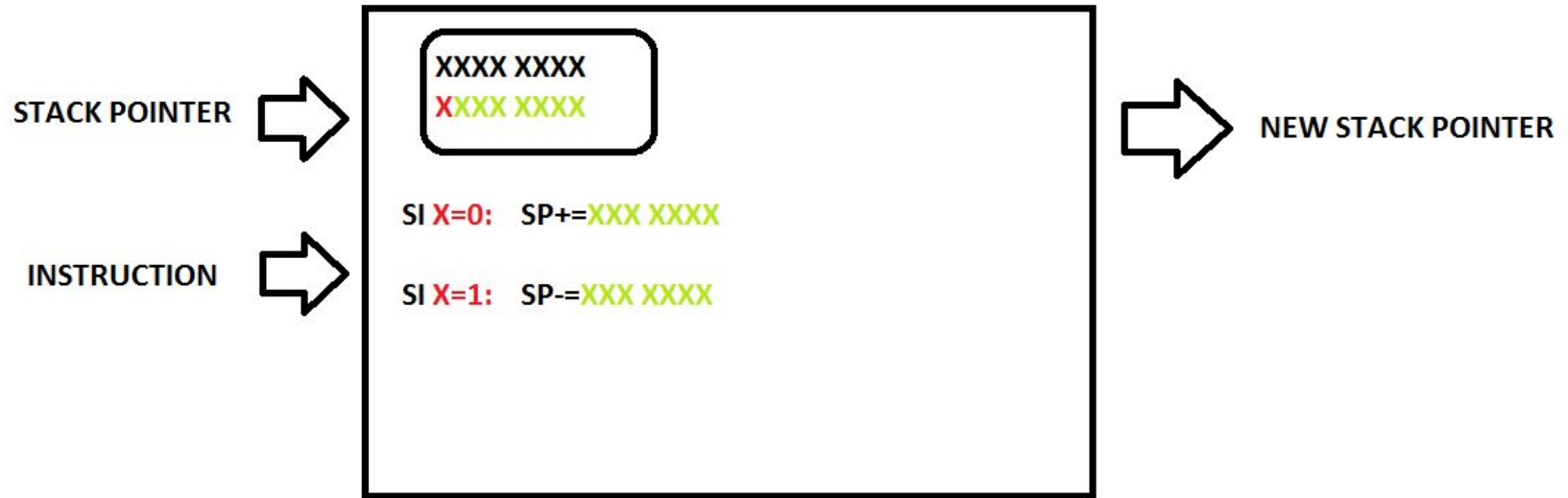
Binaire :

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	1	0	Rt			imm8							

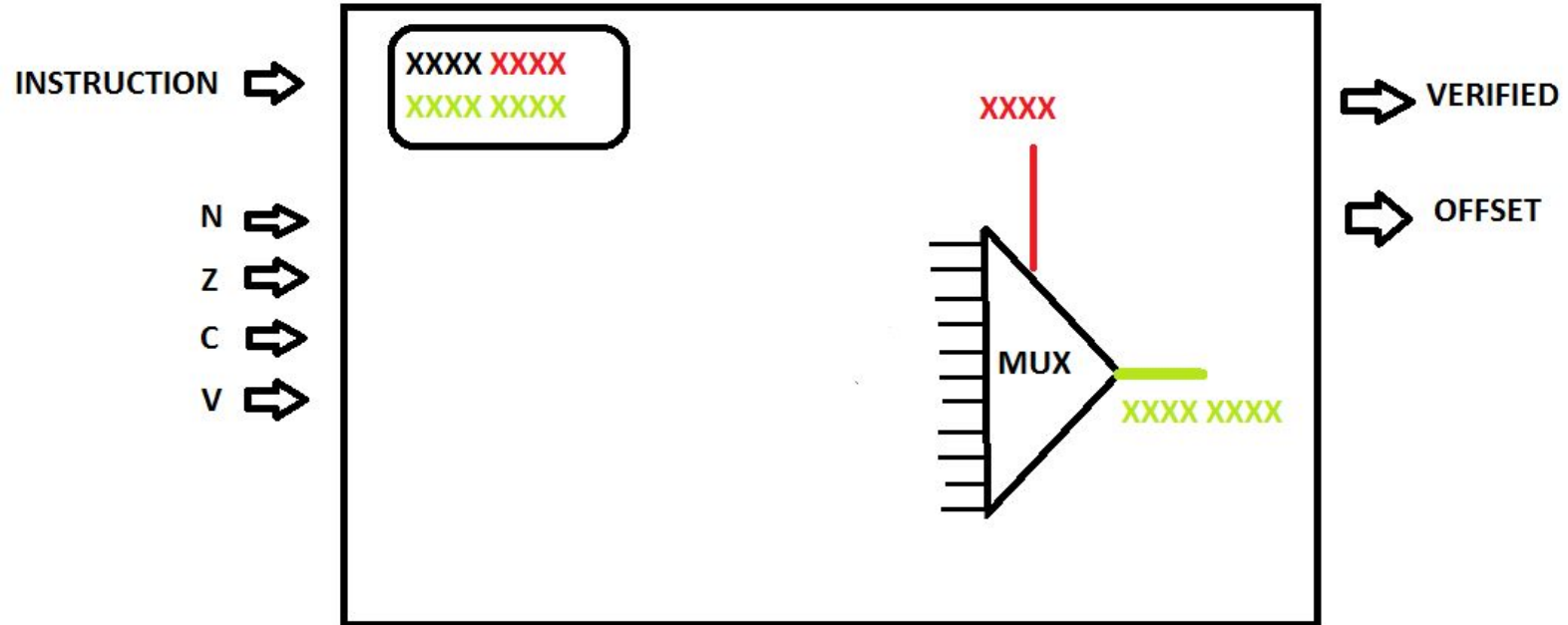


Adresse mémoire d'  
écriture/chargement

# Contrôleur : SP Address



# Contrôleur : Conditional

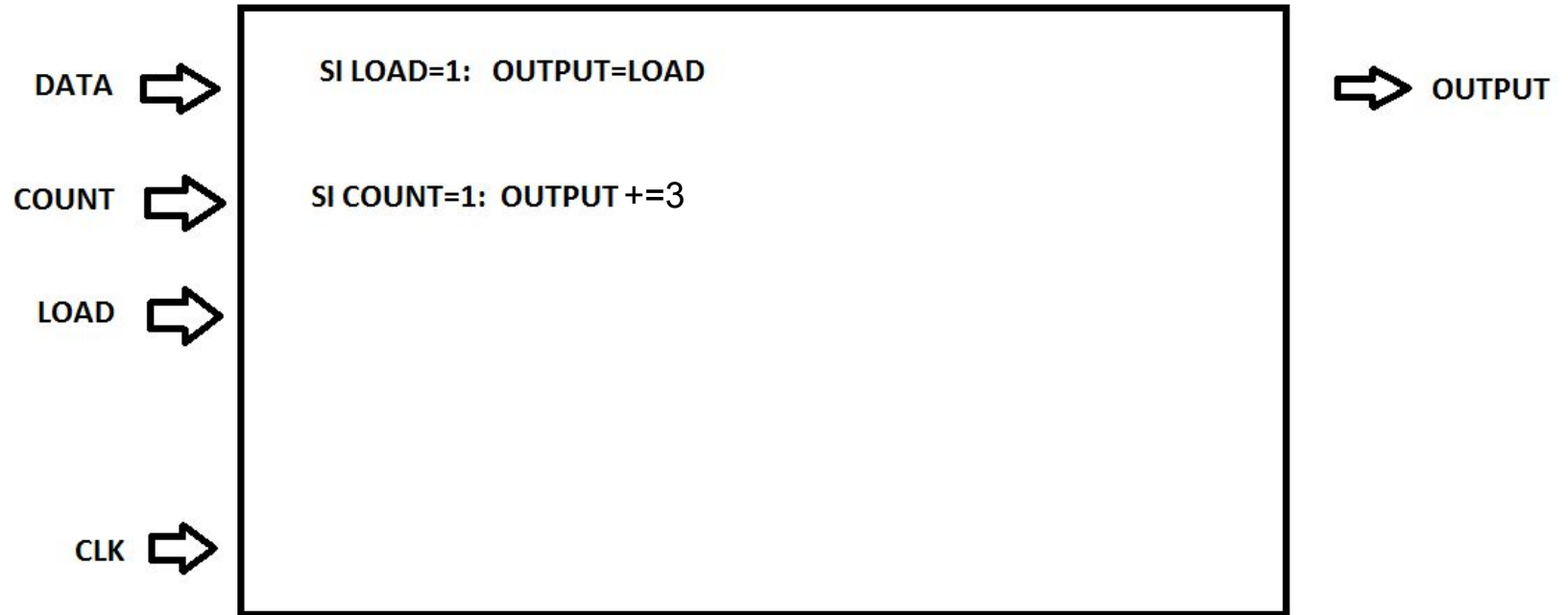


# Contrôleur : Conditional

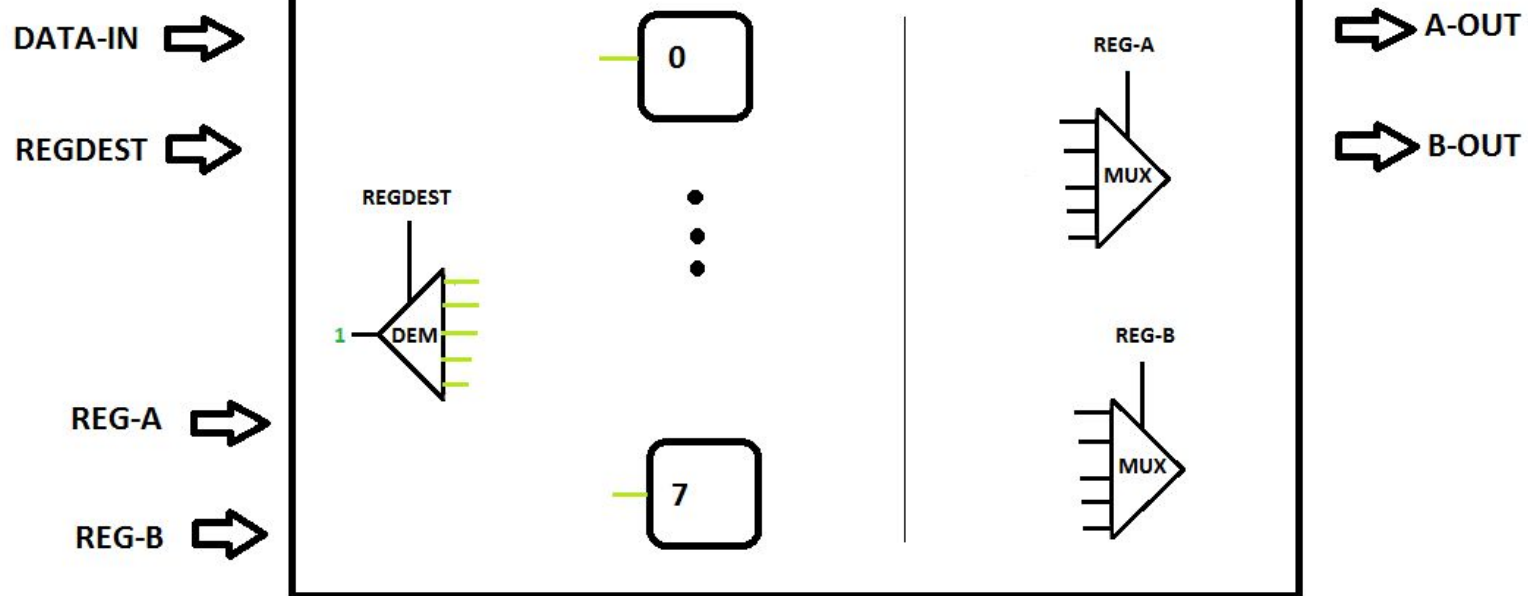
cond	Mnemonic extension	Meaning, integer arithmetic	Meaning, floating-point arithmetic <sup>a</sup>	Condition flags
0000	EQ	Equal	Equal	$Z = 1$
0001	NE	Not equal	Not equal, or unordered	$Z = 0$
0010	CS <sup>b</sup>	Carry set	Greater than, equal, or unordered	$C = 1$
0011	CC <sup>c</sup>	Carry clear	Less than	$C = 0$
0100	MI	Minus, negative	Less than	$N = 1$
0101	PL	Plus, positive or zero	Greater than, equal, or unordered	$N = 0$
0110	VS	Overflow	Unordered	$V = 1$
0111	VC	No overflow	Not unordered	$V = 0$
1000	HI	Unsigned higher	Greater than, or unordered	$C = 1$ and $Z = 0$
1001	LS	Unsigned lower or same	Less than or equal	$C = 0$ or $Z = 1$
1010	GE	Signed greater than or equal	Greater than or equal	$N = V$
1011	LT	Signed less than	Less than, or unordered	$N \neq V$
1100	GT	Signed greater than	Greater than	$Z = 0$ and $N = V$
1101	LE	Signed less than or equal	Less than, equal, or unordered	$Z = 1$ or $N \neq V$
1110	None (AL) <sup>d</sup>	Always (unconditional)	Always (unconditional)	Any



# Contrôleur : Program Counter



# Banc de registre





# Analyse des problèmes

- 01 Impossibilité d'implémenter CPU/RAM/ROM à cause d'un problème de version
- 02 Partie assembleur réalisée en C
- 03 Difficulté de compréhension de l'agencement global


# Répartition du travail





# Conclusion

- Découverte d'un nouveau domaine
- Meilleure compréhension d'un ordinateur (micro-processeur)
- Projet assez complexe à réaliser
- Mais bonne dynamique de groupe



Merci de nous avoir écouté  
Avez-vous des questions  
?

