



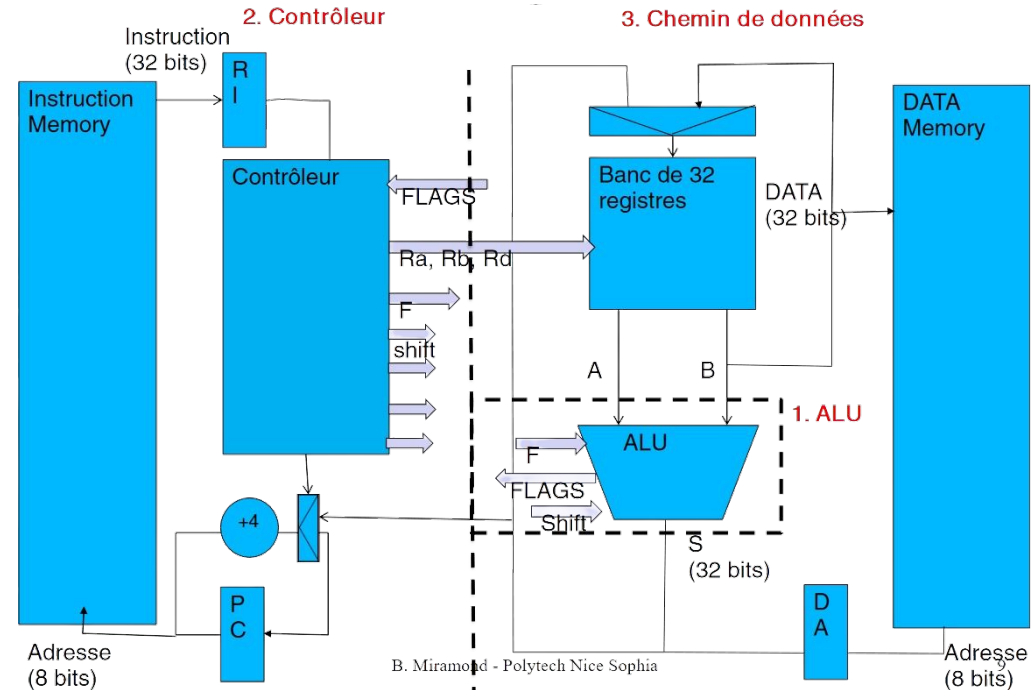
Projet PARM

Hajar El Gholabzouri
Gwendolyne Bouchard
Camille Antonios
Amélie Muller

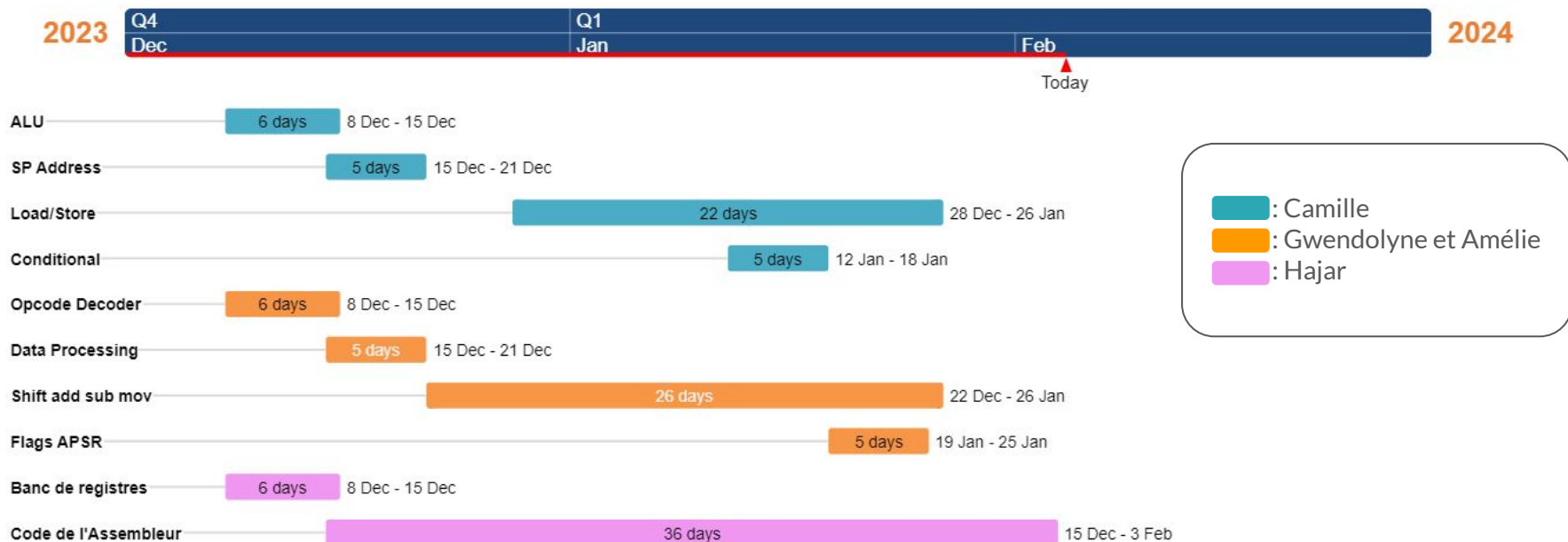
SI3 Février 2024

Architecture générale

- **But** : Reconstruire un microprocesseur ARM capable d'exécuter une liste d'instruction
- Fonctionnement du Processeur



Répartition du travail





Difficultés rencontrées

- Compréhension globale du projet
- Manque de connaissances globale en Électronique au début du projet
- Code de l'assembleur : écriture du parseur fastidieuse
- Debuggage
- Modélisation de certains composants :
 - Mise en place du Load/Store
 - Mise en place du Shift, add, sub, mov

Qualité du projet

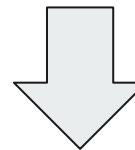


Bonne qualité	Mauvaise qualité
<ul style="list-style-type: none">• Projet complet : toutes les instructions de la doc sont gérées• Tous les tests unitaires passent• Tous les tests d'intégration passent• Tableau Excel regroupant toutes les instructions en binaire très clair	<ul style="list-style-type: none">• Doute sur l'efficacité de l'architecture des branchements (peu de connaissances en électronique avant le projet)• Code de l'assembleur long, complexe et peu lisible

Démonstration Assembleur

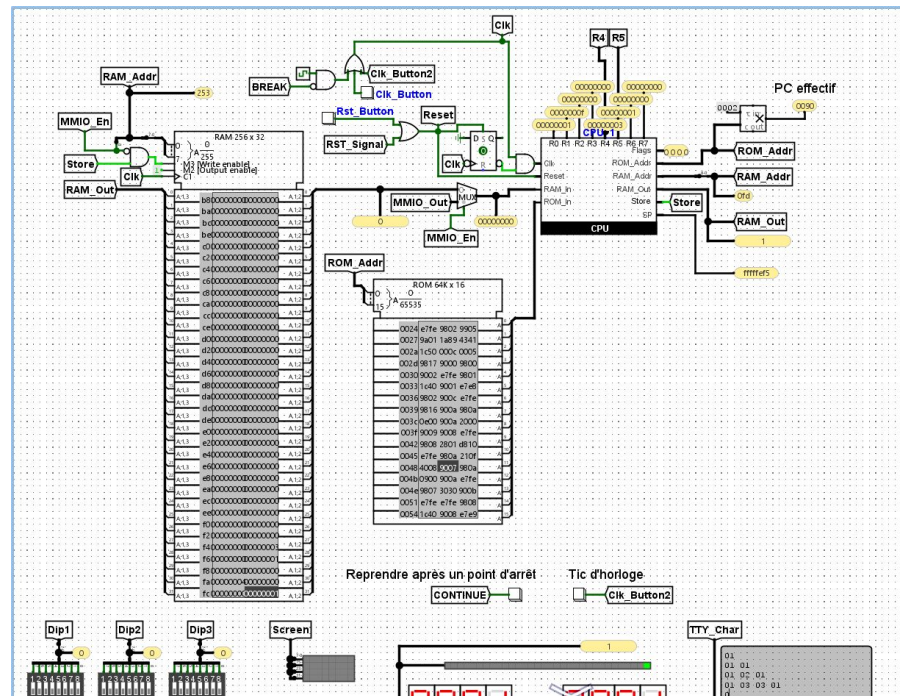
- Rôle du code assembleur : Transformer des instructions assembleur en hexadécimal

```
run:
    .fnstart
    .pad    #96
    sub     sp, #96
    @APP
    sub     sp, #508
    @NO_APP
    @APP
    sub     sp, #452
    @NO_APP
    movs    r0, #1
    str     r0, [sp, #8]
    movs    r0, #2
    str     r0, [sp, #4]
    ldr     r0, [sp, #8]
    ldr     r1, [sp, #4]
    adds    r0, r0, r1
    str     r0, [sp]
    ldr     r0, [sp]
    str     r0, [sp, #36]
    b       .LBB0_1
.LBB0_1:
    b       .LBB0_2
.LBB0_2:
    b       .LBB0_2
.Lfunc_end0:
    .size   run, .Lfunc_end0-run
    .cantunwind
    .fncend
```



```
b098 b0ff b0f1 2001 9002 2002 9001 9802 9901 1840 9000 9800 9009 e7fe e7fe e7fd
|
```

Démonstration Processeur



Merci pour votre écoute