Notes Archi

Louis Thevenet

1. LEDs

```
PILE = 0x200 // fond de pile à l'adresse 0x200
set PILE, %sp //initialisation du pointeur de pile : ABSOLUMENT NECESSAIRE

LEDS = 0xB0000000
N = 10
set N, %r1
call Afficher_LEDs
Fin: ba Fin



Afficher_LEDs:
    // affiche le contenu de %r1 sur les LEDs
    push %r1
    push %r20
    set LEDS, %r20
    st  %r1, [%r20]
    pop %r20
    pop %r1
    ret
```

2. Nombre d'occurences

```
PILE=0x200

Prog_Principal:
set PILE, %sp
set 2, %r2
set Tab, %r3
set 12, %r4
call Nb_Occ
FinProg: ba FinProg

Nb_Occ:
// entrées : %r2 : le nombre à chercher
//           %r3 : début du tableau
//           %r4 : taille du tableau
// sortie :  %5  : le nb d'occurrences de %r2 dans Tab
push %r6
push %r7

set 0,%r5
set 0,%r6
Boucle:
cmp %r6,%r4
bge FinBoucle
ld [%r3+%r6],%r7
```

```
cmp %r7,%r2
bne FinSi
inc %r5
FinSi:
inc %r6
ba Boucle
FinBoucle:
pop %r7
pop %r6
ret // IMPORTANT

Tab:.word 2,5,4,2,5,4,1,3,2,5,8,2
```

3. Interruptions

```
PILE = 0x200
LEDS = 0xB0000000
N=6

ba Prog_Principal

Handler_IT:
// affiche le contenu de %r2 sur les LEDs
push %r20
set LEDS, %r20
st %r2, [%r20]
pop %r20
reti // IMPORTANT

Prog_Principal:
set PILE,%sp
set 0, %r2
set N, %r3
Boucle: cmp %r2, %r3
bl Suite
set 0, %r2
Suite:
inc %r2
ba Boucle
```

4. MiniCraps

Sure, here's a brief summary of your CPU:

The CPU is a 4-stage pipeline that executes instructions in a loop starting from Fetch and ending with PCPlus1, then repeating the cycle. It has a total of 16 possible instructions, some of which can perform arithmetic operations on memory operands. The CPU fetches an instruction from memory in the « Fetch » stage, decodes it in the « Decode2PCPlus1 » stage, increments the PC value by adding it to the contents of the Memory module at the address determined by the PCPlus1 signal in the « PCPlus1 » stage, and finally sets the flags N.