

Questão 1 - Memória Principal

Implementamos o código seguinte para a primeira questão

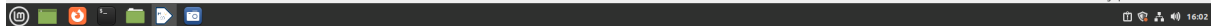
```
.text
addi a7, x0, 4           # a7 = 4: Print string
la a1, teste             # a1: endereço que aponta para string
ecall                   # call para saída: string

.data                    # memória de dados (0x10000000)
teste: .string "HELLO WORLD!" # string
```

Observamos o seguinte resultado no programa Ripes após a execução para os endereços:

The screenshot shows the Ripes debugger interface. The main window is divided into several panes. On the left, there is a sidebar with icons for 'Enter', 'Processor', 'Cache', and 'Memory'. The central pane is split into three sections: 'Source code' on the left, 'Assembly' in the middle, and 'GPR' on the right. The 'Source code' pane shows the assembly code from the previous block. The 'Assembly' pane shows the disassembled instructions with their addresses and hex values. The 'GPR' pane shows a list of registers (x0 to x26) with their names, aliases, and values. The register x11 is highlighted in orange, showing its alias 'a1' and value '0x10000000'. The bottom status bar indicates the processor is '5-stage processor' and the ISA is 'RV32IM'.

Name	Alias	Value
x0	zero	0x00000000
x1	ra	0x00000000
x2	sp	0x7ffffff0
x3	gp	0x10000000
x4	tp	0x00000000
x5	t0	0x00000000
x6	t1	0x00000000
x7	t2	0x00000000
x8	s0	0x00000000
x9	s1	0x00000000
x10	a0	0x00000000
x11	a1	0x10000000
x12	a2	0x00000000
x13	a3	0x00000000
x14	a4	0x00000000
x15	a5	0x00000000
x16	a6	0x00000000
x17	a7	0x00000004
x18	s2	0x00000000
x19	s3	0x00000000
x20	s4	0x00000000
x21	s5	0x00000000
x22	s6	0x00000000
x23	s7	0x00000000
x24	s8	0x00000000
x25	s9	0x00000000
x26	s10	0x00000000



The image shows a screenshot of the Ripes debugger interface. At the top, the title bar reads "Ripes" with standard window controls. Below the title bar is a menu bar with "File", "Edit", "View", and "Help". A toolbar contains icons for file operations and a status indicator showing "100 ms". On the left side, there is a vertical sidebar with icons for "CPU", "Memory", "I/O", and "CPU". The main window is divided into two panes. The left pane, titled "Memory viewer", displays a table of memory addresses and their corresponding data. The right pane, titled "Memory map", shows a summary of memory segments. The "Memory viewer" table has columns for "Address", "Word", "Byte 0", "Byte 1", "Byte 2", and "Byte 3". It lists memory addresses from 0x00000036 to 0x00000010, followed by a section of memory starting at 0x0000000c. The "Memory map" table has columns for "Name", "Size", and "Range". It lists segments for ".text" (16 bytes, range 0x00000000 - 0x00000010), ".data" (12 bytes, range 0x10000000 - 0x1000000c), and ".bss" (0 bytes, range 0x10000000 - 0x10000000). At the bottom of the "Memory viewer" pane, there are controls for "Display type" (set to "Hex"), "Go to register:" (empty), and "Go to section:" (empty). The bottom status bar shows "Processor: 5-stage processor ISA: RV32IM" and a system tray with icons for network, volume, and time (16:00).

The image shows a screenshot of the Ripes debugger interface. At the top, the title bar reads "Ripes" with standard window controls. Below the title bar is a menu bar with "File", "Edit", "View", and "Help". A toolbar contains icons for file operations and a status bar showing "100 ms". On the left side, there is a vertical toolbar with icons for CPU, Memory, and I/O. The main window is divided into two panes. The left pane, titled "Memory viewer", displays a table of memory addresses, words, and bytes. The right pane, titled "Memory map", displays a table of memory segments with names, sizes, and ranges. At the bottom, there is a status bar showing "Display type: Hex", "Go to register:", "Go to section:", and "Processor: 5-stage processor ISA: RV32IM".

The screenshot displays the Ripes debugger interface. At the top, the title bar reads "Ripes". The menu bar includes "File", "Edit", "View", and "Help". Below the menu bar, there is a toolbar with icons for file operations, a refresh button, and a timing display showing "100 ms".

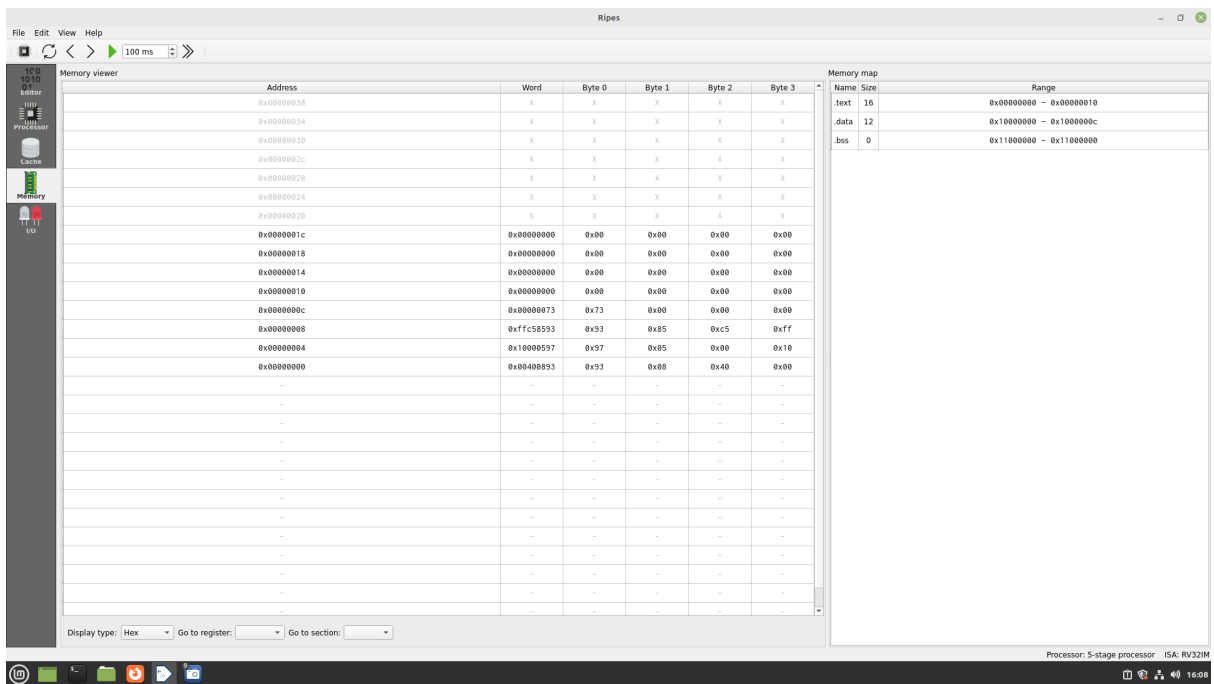
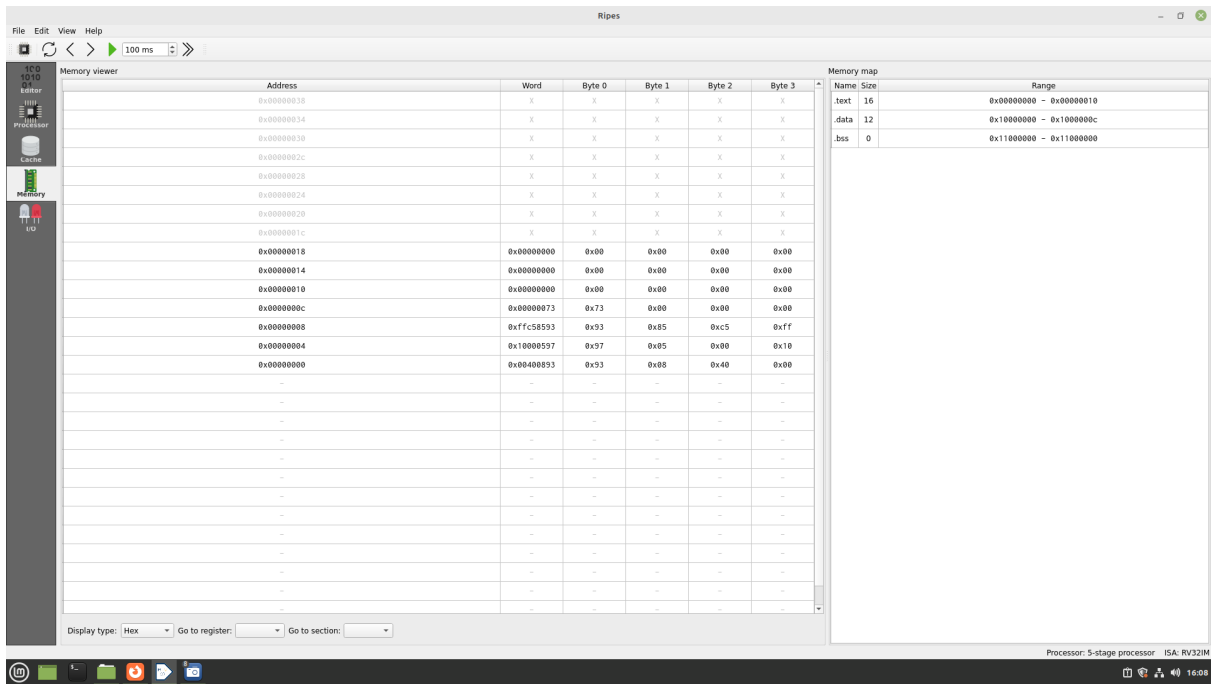
The main window is divided into two primary sections: "Memory viewer" on the left and "Memory map" on the right.

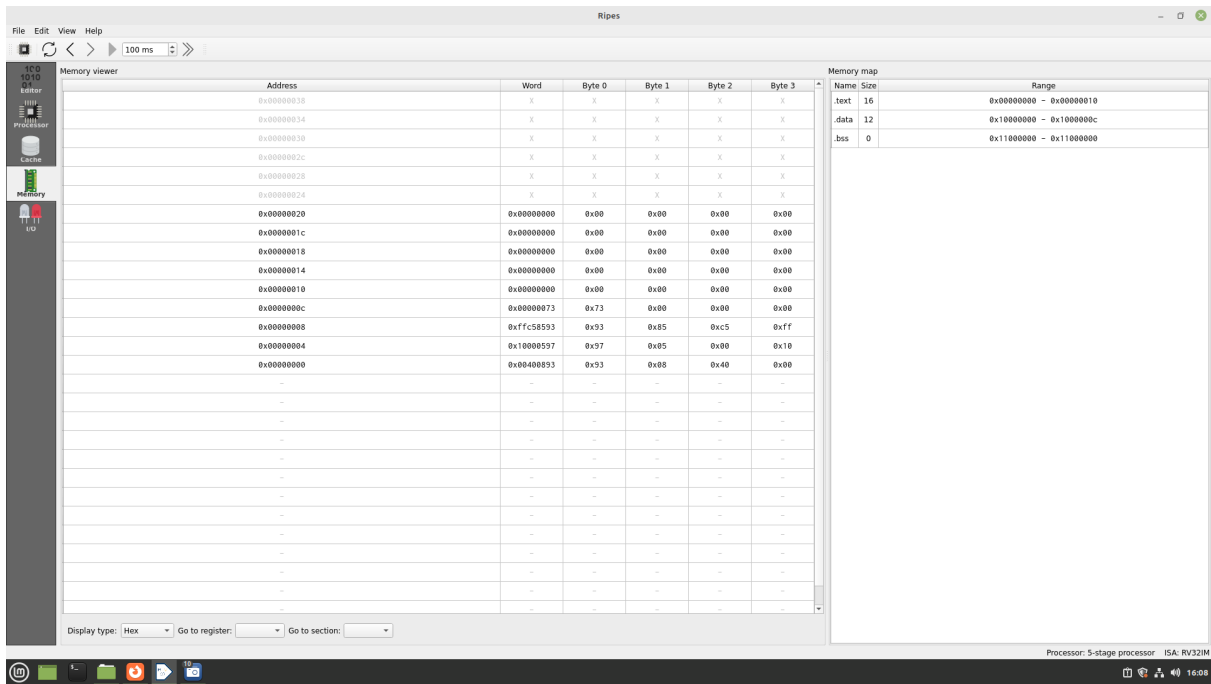
Memory viewer: This section contains a table with the following columns: "Address", "Word", "Byte 0", "Byte 1", "Byte 2", and "Byte 3". The table lists memory addresses from 0x00000030 to 0x00000010, with corresponding data values. For example, at address 0x00000030, the word is 0x00000000, and the bytes are 0x00, 0x00, 0x00, 0x00. The table continues down to address 0x00000010, where the word is 0x00000000 and the bytes are 0x00, 0x00, 0x00, 0x00.

Memory map: This section contains a table with the following columns: "Name", "Size", and "Range". It lists memory segments: ".text" (Size 16, Range 0x00000000 - 0x00000010), ".data" (Size 12, Range 0x10000000 - 0x1000000c), and ".bss" (Size 0, Range 0x10000000 - 0x10000000).

At the bottom of the window, there is a status bar with the text "Processor: S-stage processor ISA: RV32IM".

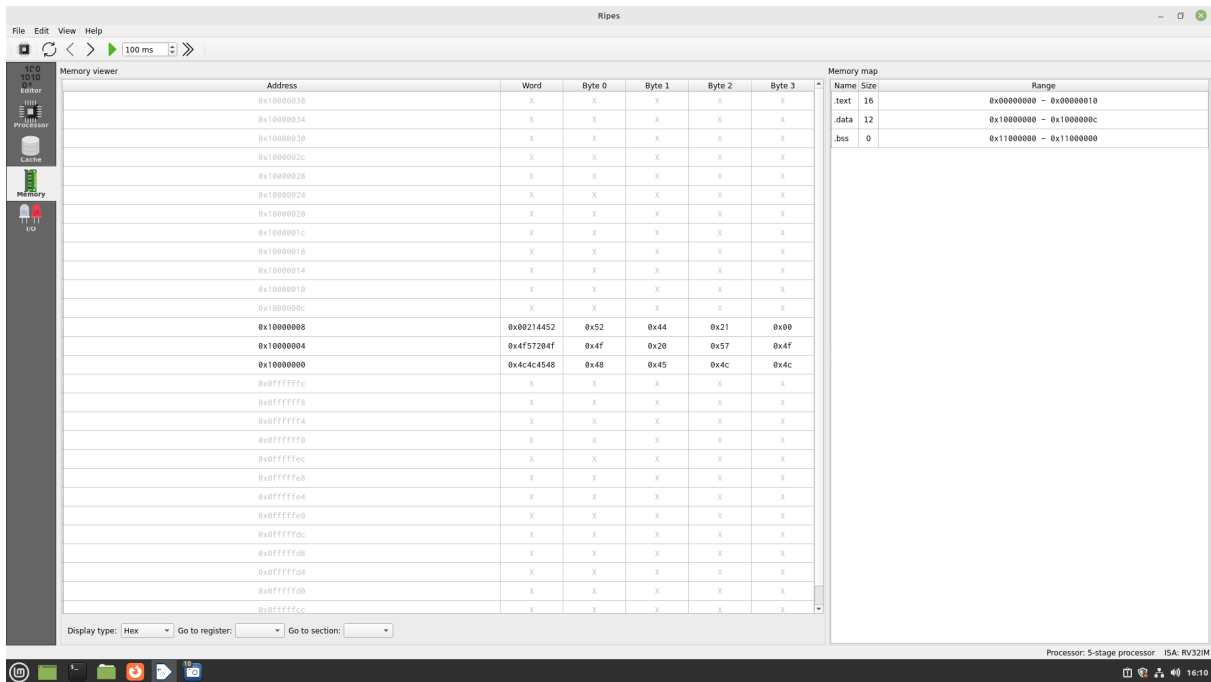
The screenshot displays the Ripes debugger interface. At the top, a menu bar includes 'File', 'Edit', 'View', and 'Help'. Below the menu, a toolbar contains navigation icons and a '100 ms' timer. The left sidebar features icons for 'TOOLBOX', 'EDITOR', 'HW', 'PROFILER', 'CACHE', 'MEMORY', and 'I/O'. The main window is divided into two primary sections: 'Memory viewer' on the left and 'Memory map' on the right. The 'Memory viewer' section contains a table with columns for 'Address', 'Word', 'Byte 0', 'Byte 1', 'Byte 2', and 'Byte 3'. It lists memory addresses from 0x00000030 to 0x00000040, showing data values and their byte-level breakdown. The 'Memory map' section contains a table with columns for 'Name', 'Size', and 'Range'. It lists memory regions such as 'text' (0x00000000 - 0x00000010), 'data' (0x10000000 - 0x1000000c), and 'bss' (0x10000000 - 0x10000000). At the bottom of the main window, there are controls for 'Display type' (set to Hex), 'Go to register', and 'Go to section'. The bottom status bar indicates the processor is '5-stage processor' and the ISA is 'RV32IM'. The system tray at the very bottom shows various icons and the time '16:04'.

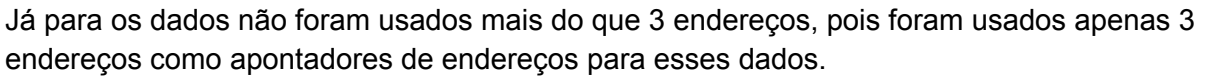




Observe que o programa só começa a usar os endereços de memória nos momentos de escrita na memória usando os resultados para guardar.

Dados da memória RAM no início da execução das instruções para os dados:



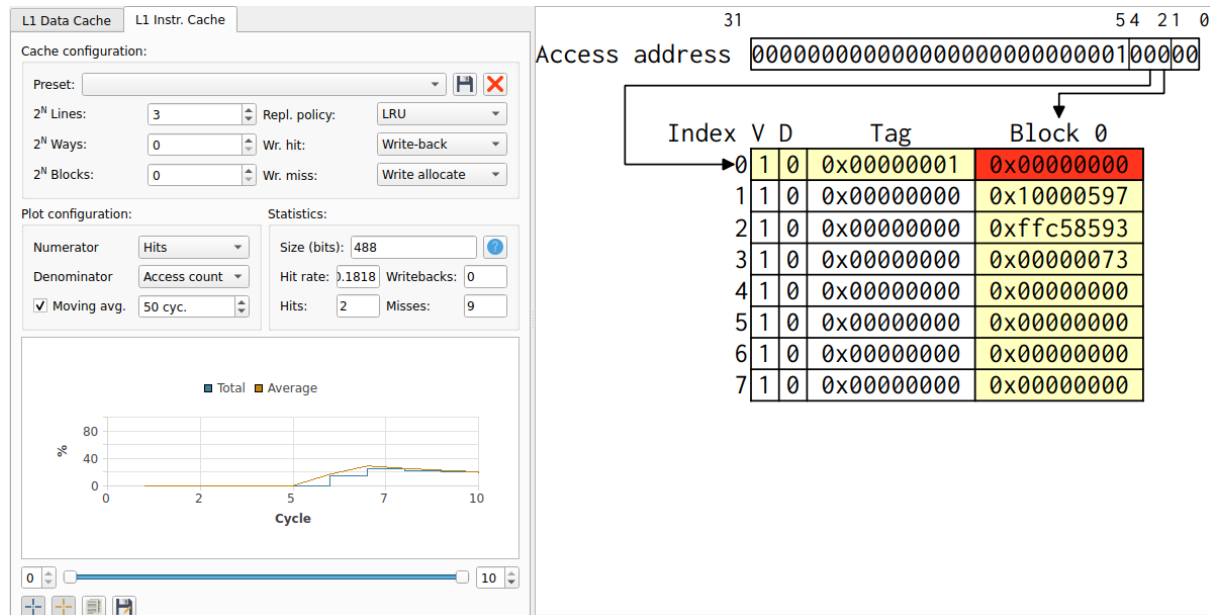


Já para os dados não foram usados mais do que 3 endereços, pois foram usados apenas 3 endereços como apontadores de endereços para esses dados.

Questão 2.1:

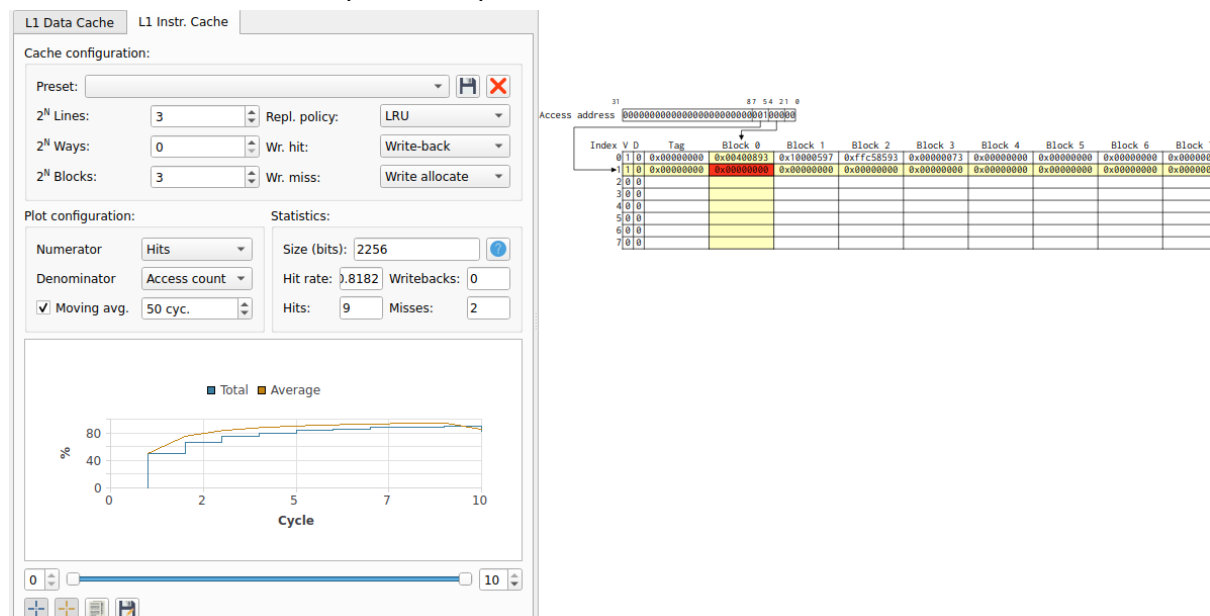
Para a segunda parte iremos usar a memória cache no Ripes para as instruções e dados para o mapeamento direto e observar o número de hits e miss para a memória cache.

Para 2^3 linhas e 2^0 blocos para o mapeamento direto.



É possível observar que houveram 2 hits(acertos) e 9 misses(falhas) para 2^0 blocos e 2^3 linhas no mapeamento direto.

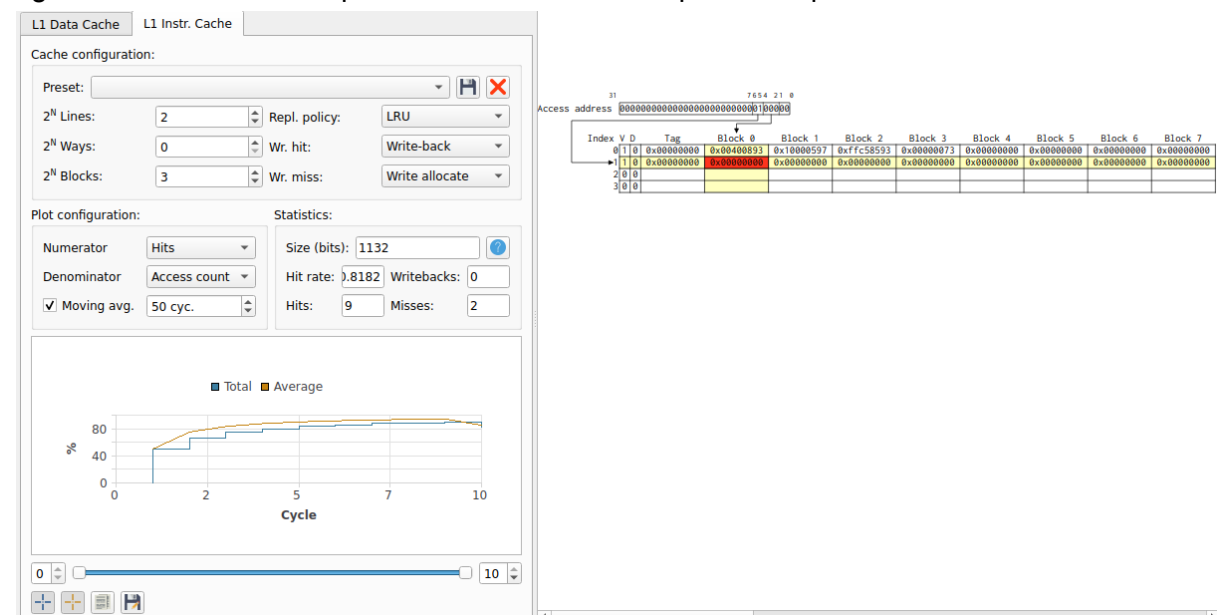
Para 2^3 linhas e 2^3 blocos para o mapeamento direto.



Percebe-se que houve uma diferença entre o número de hits(acertos) e miss(falhas). Dessa vez foram 9 hits(acertos) e 2 misses(falhas) para o mapeamento direto.

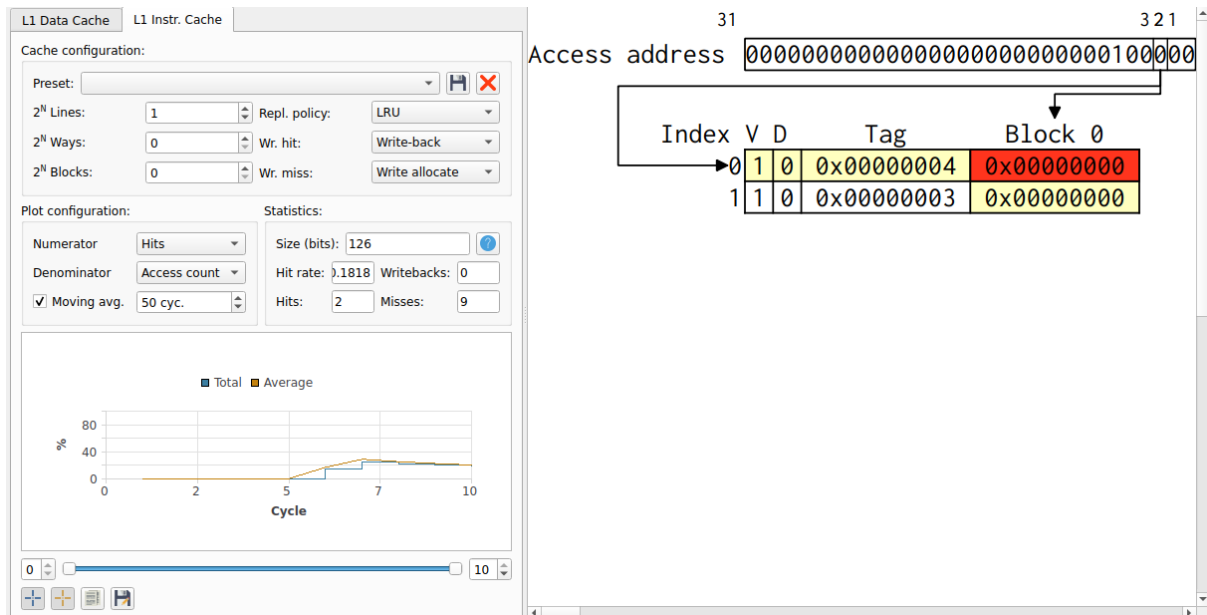
[illegible]

Agora faremos o mesmo para 2^2 linhas e 2^3 blocos para o mapeamento direto.



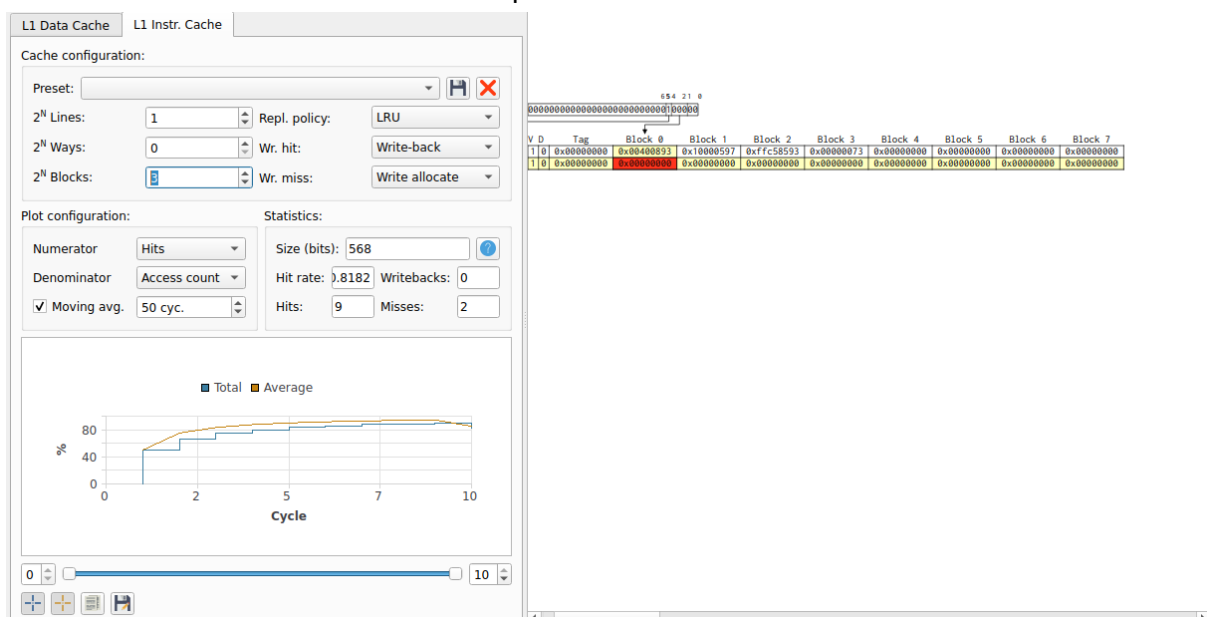
Foram 9 hits(acertos) e 2 misses(falhas) para o mapeamento direto.

Agora para 2¹ linhas 2⁰ blocos para o mapeamento direto.



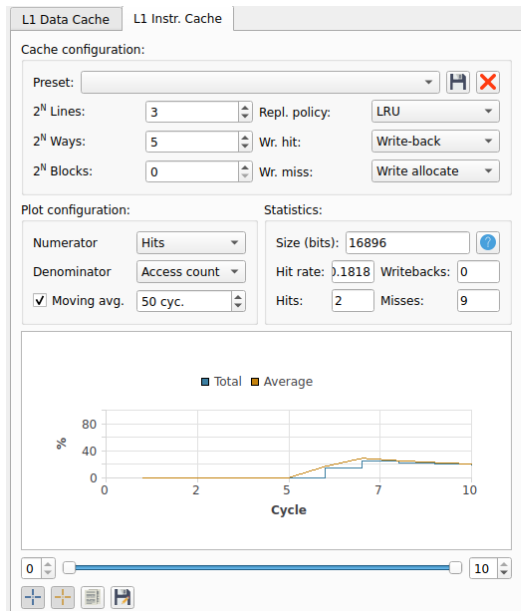
Desta vez foram 2 hits(acertos) e 9 misses(falhas) para 2¹ linhas e 2⁰ blocos. para o mapeamento direto.

Para 2^1 linhas e 2^3 blocos usando o mapeamento direto.



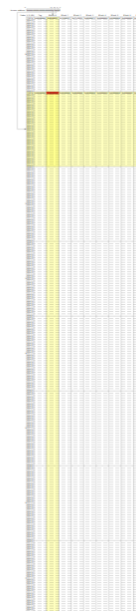
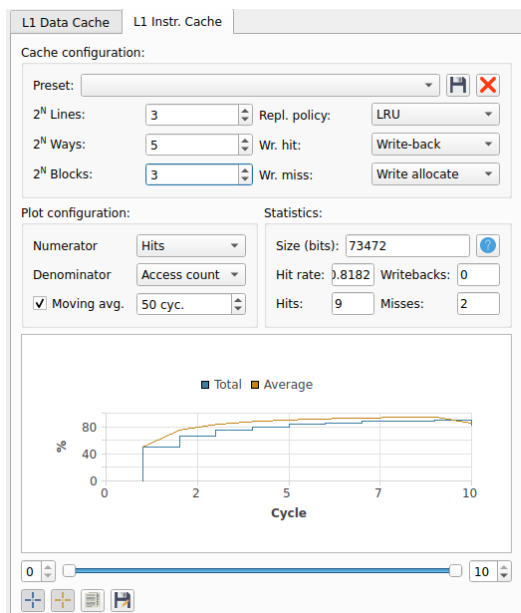
Para 2¹ linhas e 2³ blocos foram 9 hits(acertos) e 2 misses(falhas) para o mapeamento direto.

Agora faremos o mesmo para o mapeamento associativo.
Para 2³ linhas e 2⁰ blocos para o mapeamento associativo.



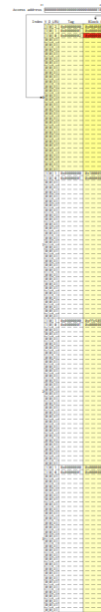
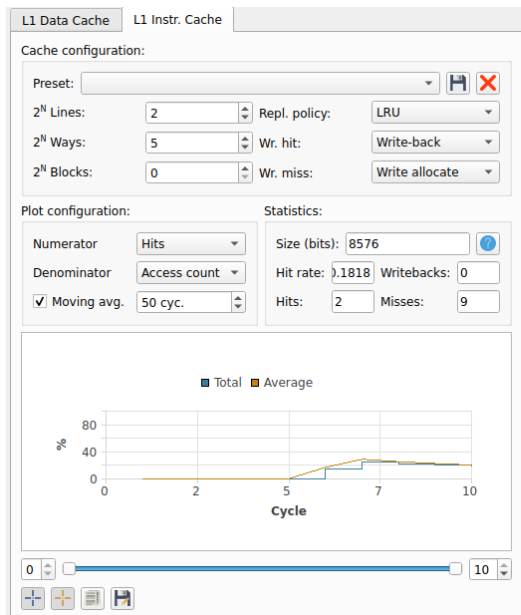
Foram 2 hits(acertos) e 9 misses(falhas) para o mapeamento associativo.

Para 2³ linhas e 2³ blocos para o mapeamento associativo.



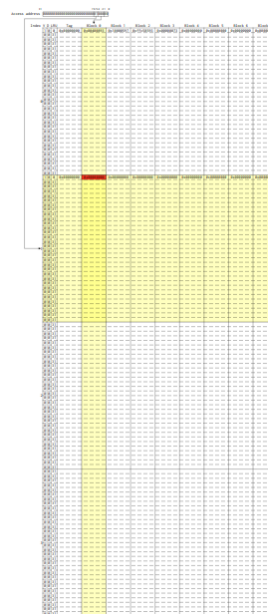
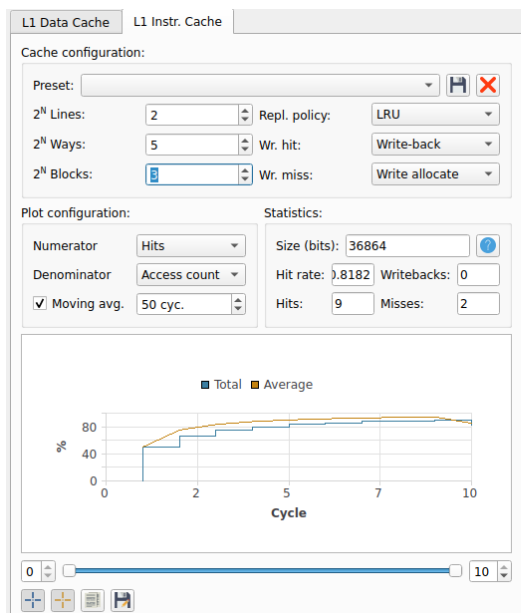
Foram 9 hits(acertos) e 2 misses(falhas) para o mapeamento associativo.

Para 2^2 linhas e 2^0 blocos usando mapeamento associativo.



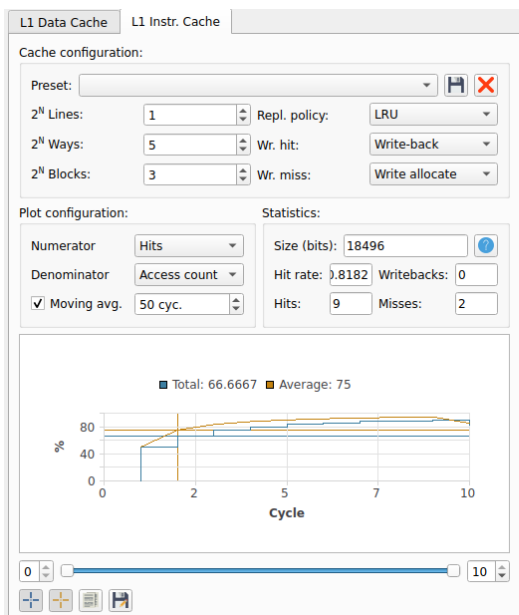
Foram 2 hits(acertos) e 9 misses(falhas) para o mapeamento associativo.

Para 2^2 linhas e 2^3 colunas para o mapeamento associativo.



Foram 9 hits(acertos) e 2 misses(falhas) para o mapeamento associativo.

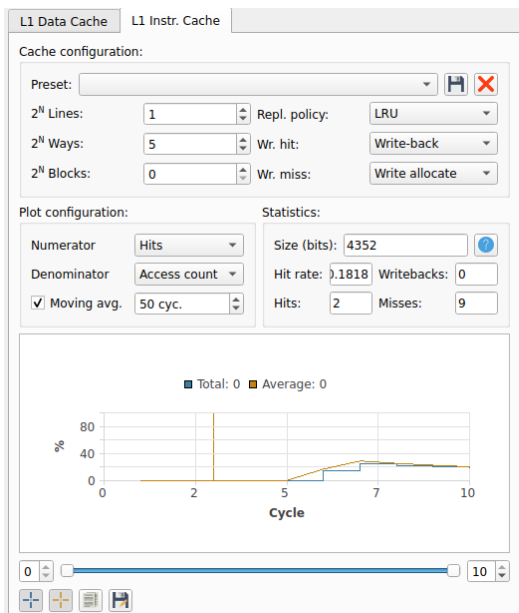
Para 2¹ linhas e 2³ blocos usando mapeamento associativo.



Index	Y	D	LRU	Tag	Block 0	Block 1	Block 2	Block 3	Block 4	Block 5	Block 6	Block 7
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0

Foram 9 hits(acertos) e 2 misses(falhas) para o mapeamento associativo.

Para 2¹ linhas e 2⁰ blocos para o mapeamento associativo.



Index	Y	D	LRU	Tag	Block 0
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	0	0	0	0	0
18	0	0	0	0	0
19	0	0	0	0	0
20	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0
24	0	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	0	0
28	0	0	0	0	0
29	0	0	0	0	0
30	0	0	0	0	0
31	0	0	0	0	0

Foram 2 hits(acertos) e 9 misses(falhas) para o mapeamento associativo.

Mapeamento Direto:

- 2^3 linhas e 2^3 blocos: 9 misses (acertos) e 2 misses(falhas).
- 2^3 linhas e 2^0 blocos: 2 misses (acertos) e 9 misses(falhas).
- 2^2 linhas e 2^3 blocos: 9 misses (acertos) e 2 misses(falhas).
- 2^2 linhas e 2^0 blocos: 2 misses (acertos) e 9 misses(falhas).
- 2^1 linhas e 2^3 blocos: 9 misses (acertos) e 2 misses(falhas).
- 2^1 linhas e 2^0 blocos: 2 misses (acertos) e 9 misses(falhas).

Mapeamento Associativo:

- 2^3 linhas e 2^3 blocos: 9 misses (acertos) e 2 misses(falhas).
- 2^3 linhas e 2^0 blocos: 2 misses (acertos) e 9 misses(falhas).
- 2^2 linhas e 2^3 blocos: 9 misses (acertos) e 2 misses(falhas).
- 2^2 linhas e 2^0 blocos: 2 misses (acertos) e 9 misses(falhas).
- 2^1 linhas e 2^3 blocos: 9 misses (acertos) e 2 misses(falhas).
- 2^1 linhas e 2^0 blocos: 2 misses (acertos) e 9 misses(falhas).

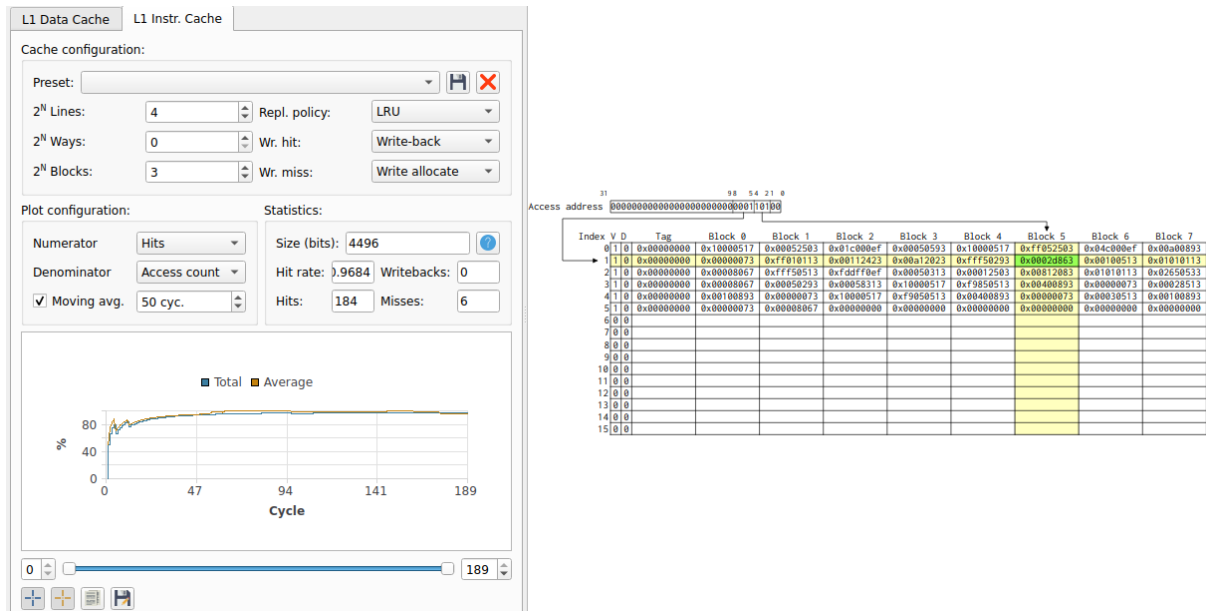
Mesmo que o mapeamento associativo seja melhor que o direto, os resultados obtidos foram os mesmos para o mapeamento direto e mapeamento associativo.

Onde no mapeamento direto, cada bloco da memória principal é mapeado para uma linha da memória cache.

E o mapeamento associativo, cada bloco é carregado em qualquer linha da memória cache.

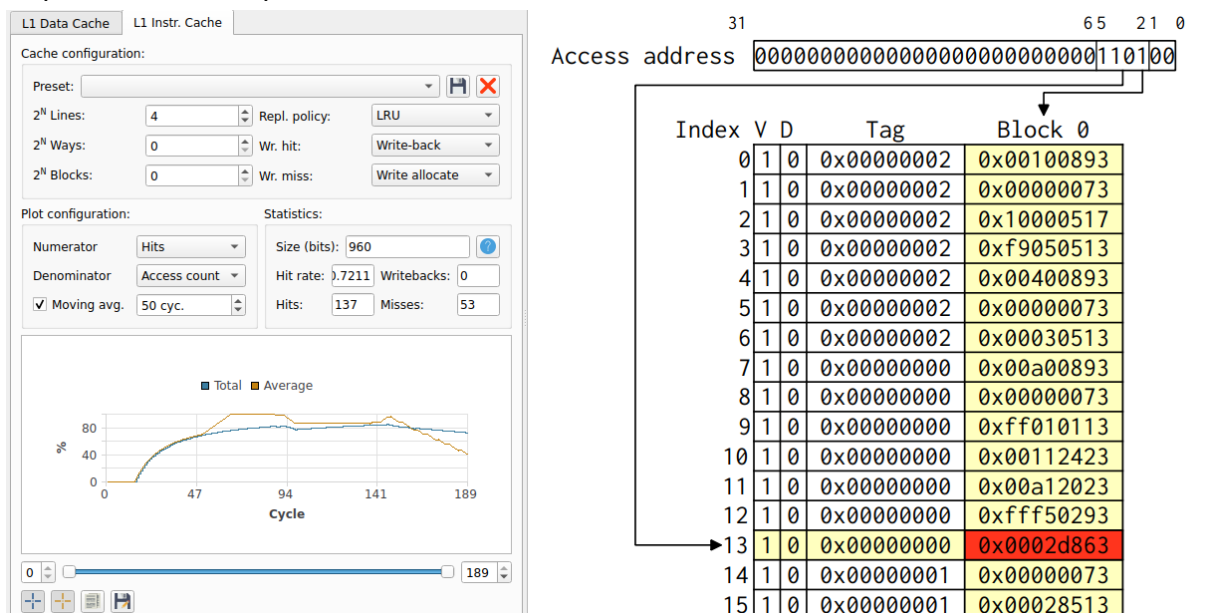
Questão 2.2:

Mapeamento Direto para 2^4 linhas e 2^3 blocos. :



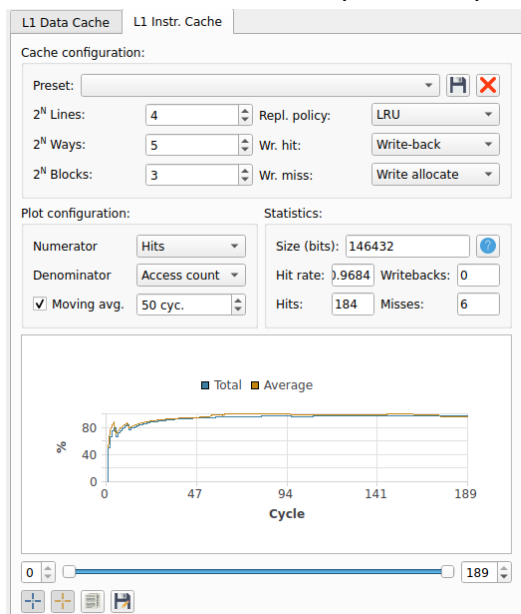
Foram 184 hits(acertos) e 6 misses(falhas) para o mapeamento direto.

Mapeamento direto para 2^4 linhas e 2^0 blocos.



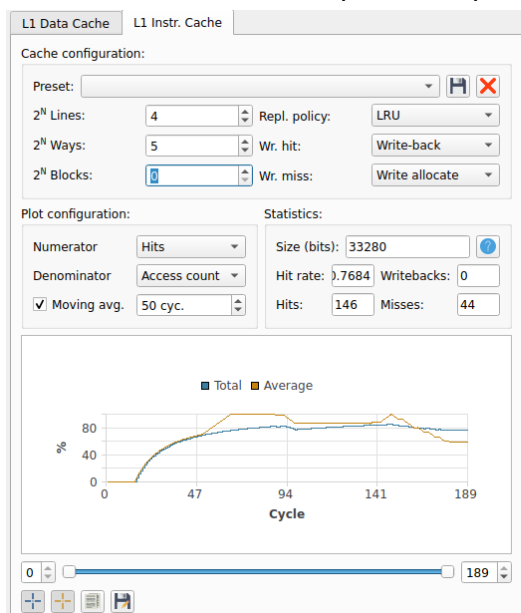
Foram 137 hits(acertos) e 53 misses(falhas) para o mapeamento direto.

Para 2^4 linhas e 2^3 blocos para o mapeamento associativo.



Foram 184 hits(acertos) e 6 misses(falhas) para o mapeamento associativo.

Para 2^4 linhas e 2^0 blocos para o mapeamento associativo.



Foram 146 hits(acertos) e 44 misses(falhas) para o mapeamento associativo.

Mapeamento Direto:

Para 2^4 linhas e 2^3 blocos: 184 hits (acertos) e 6 misses(falhas)

Para 2^4 linhas e 2^0 blocos: 137 hits (acertos) e 53 misses(falhas)

Mapeamento Associativo:

Para 2^4 linhas e 2^3 blocos: 184 hits (acertos) e 6 misses(falhas)

Para 2^4 linhas e 2^0 blocos: 146 hits (acertos) e 44 misses(falhas)

É possível observar que quanto maior for o número de blocos, maior será a chance de dar falha ao tentar acessar um dado na cache, fazendo com que ela busque na memória principal tanto para o Mapeamento Direto quanto para o Mapeamento Associativo.

Observando apenas os dois dados para o Mapeamento Direto de 2^4 linhas e 2^3 blocos e o Mapeamento Direto de 2^4 linhas e 2^0 blocos, quanto menores forem a quantidades de blocos, conseqüentemente as chances de acertos são maiores.

O mesmo acontece para o os dados do Mapeamento Associativo de 2^4 linhas e 2^3 blocos e o Mapeamento Associativo de 2^4 linhas e 2^0 blocos.

Nota-se que para as mesmas configurações para os dados do mapeamento direto e o mapeamento associativo, os resultados foram distintos.

Como o número de blocos inicialmente para os dois eram maiores, a chance de ocorrer falhas também era grande.

Já quando o número de blocos caiu para um, a diferença entre o número de acertos e falhas para o mapeamento direto e associativo foi perceptível.