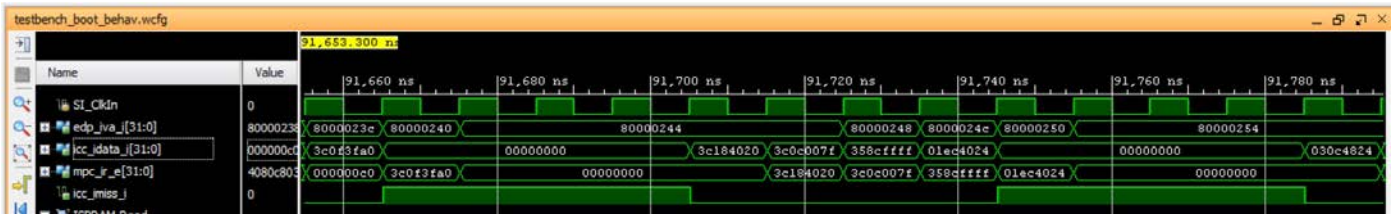
Simulation:

1) Simulation without ISPRAM:

We next show the execution of the initial instructions of the FP algorithm:

```
8000023c:   3c0f3fa0    lui   t7,0x3fa0
80000240:   3c184020    lui   t8,0x4020
80000244 <flpadd>:
80000244:   3c0c007f    lui   t4,0x7f
80000248:   358cffff    ori   t4,t4,0xffff
8000024c:   01ec4024    and   t0,t7,t4
80000250:   030c4824    and   t1,t8,t4
…
```

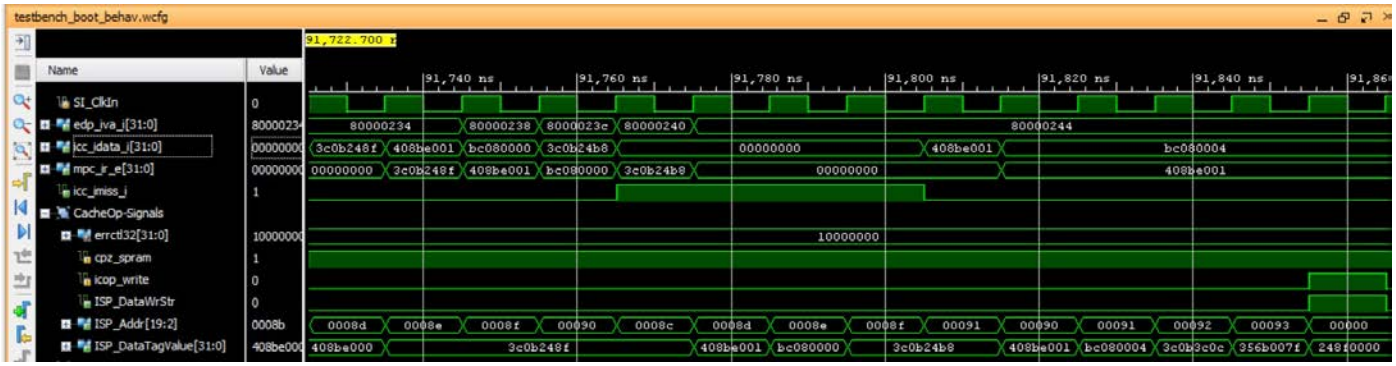


You can see that every 4 instructions there is an I$ miss.

2) Simulation with ISPRAM:

We first show the simulation of the cache instruction that initializes the first instruction in the ISPRAM (instruction `addiu $t7,$a0,0` (0x248f0000)):

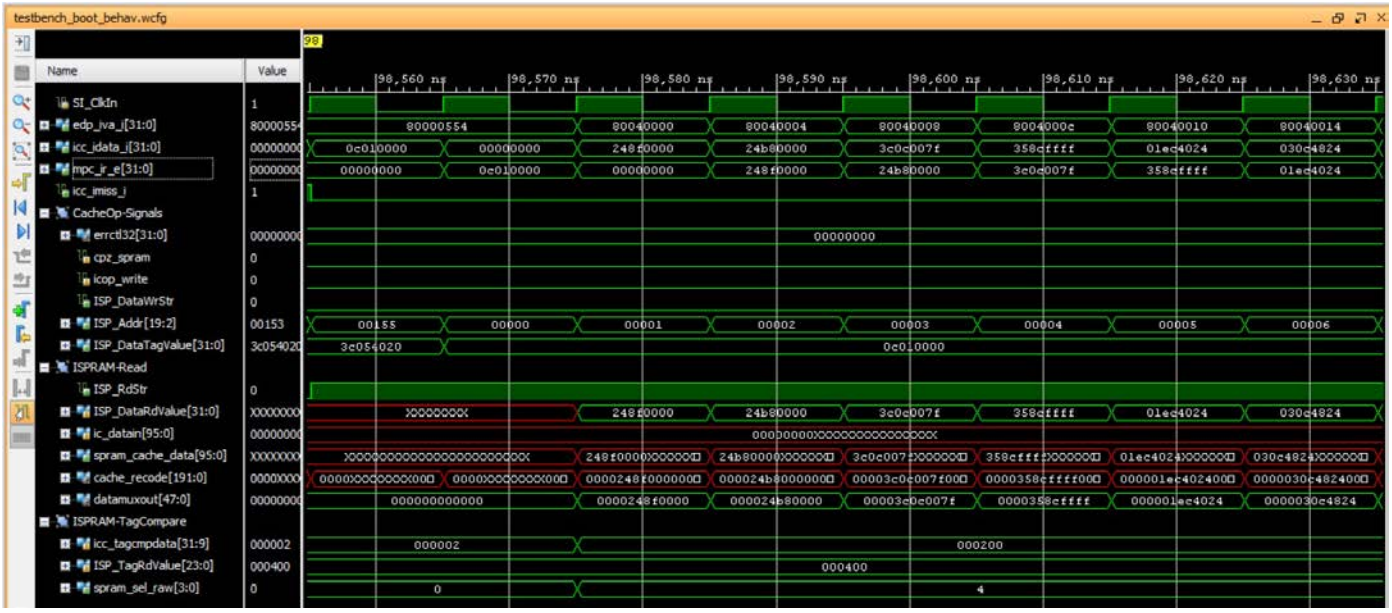
```
80000230:   3c0b248f    lui   t3,0x248f
80000234:   408be001    mtc0  t3,c0_datalo
80000238:   bc080000    cache 0x8,0(zero)
```



Observe that the first cache instruction (0xbc080000) is fetched in the third cycle. Then, in the last cycle, the ISPRAM is written with the first instruction (`addiu $t7,$a0,0` (0x248f0000)).

We next show the execution of the initial instructions of the FP algorithm:

```
80040000:   248f0000    addiu $t7,$a0,0
80040004:   24b80000    addiu $t8,$a1,0
80040008 <flpadd>:
80040008:   3c0c007f    lui   t4,0x7f
8004000c:   358cffff    ori   t4,t4,0xffff
80040010:   01ec4024    and   t0,t7,t4
80040014:   030c4824    and   t1,t8,t4
…
```



In this case there are no misses at all, as the algorithm is being fetched from the ISPRAM.

Execution on the board:

- *SimulationSources_WithoutISPRAM*:
  - Switches=0 → 91 cycles
  - Switches=1 → 43 instructions
    CPI = 91/43 ≈ 2.12
  - Switches=2 → Result: $t9=0x40700000, which corresponds to the result of adding 1.25+2.5

- *SimulationSources_ISPRAM*:
    - Switches=0 → 65 cycles
    - Switches=1 → 56 instructions
      CPI = 65/56 ≈ 1.16
    - Switches=2 → Result: $t9=0x40700000, which corresponds to the result of adding 1.25+2.5

The difference is that instructions are always found in the ISPRAM, whereas 1 every 4 instructions fetched from the I$ miss (note that if the program was not sequential, it would be even worse), adding a miss penalty of around 4 cycles.