Politecnico di Torino

DLX Presentation

Microelectronic Systems 2021-2022

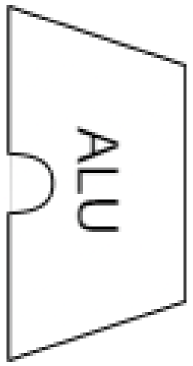
Alfredo Paolino, Vincenzo Petrolo, Diamante Simone Crescenzo

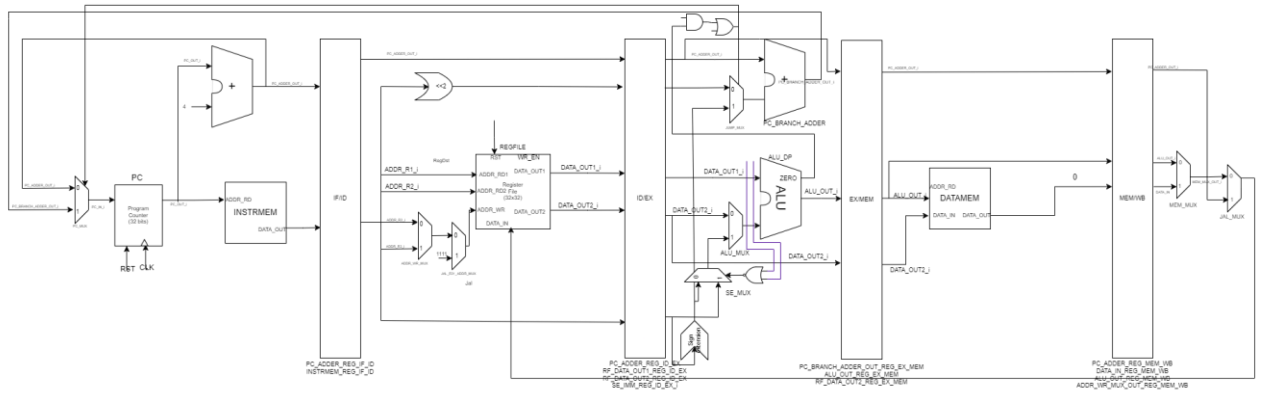
Outline

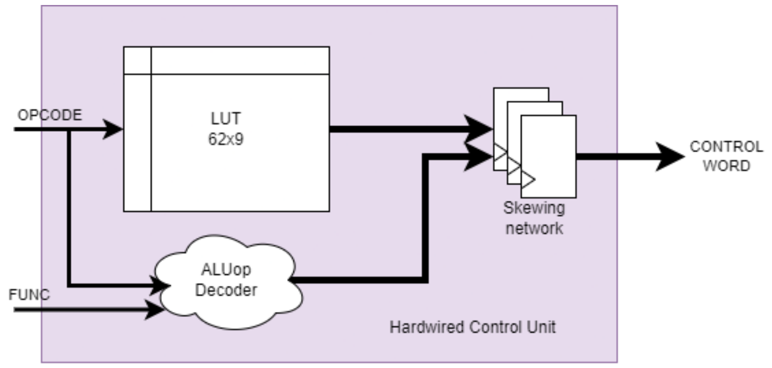
- VHDL Design
  - Datapath
  - Instruction memory
  - Data memory
  - Control Unit
- Simulation
- Synthesis
- Physical layout
- Technical report
- Possible improvements

VHDL Design

- Chosen approach: Bottom-up



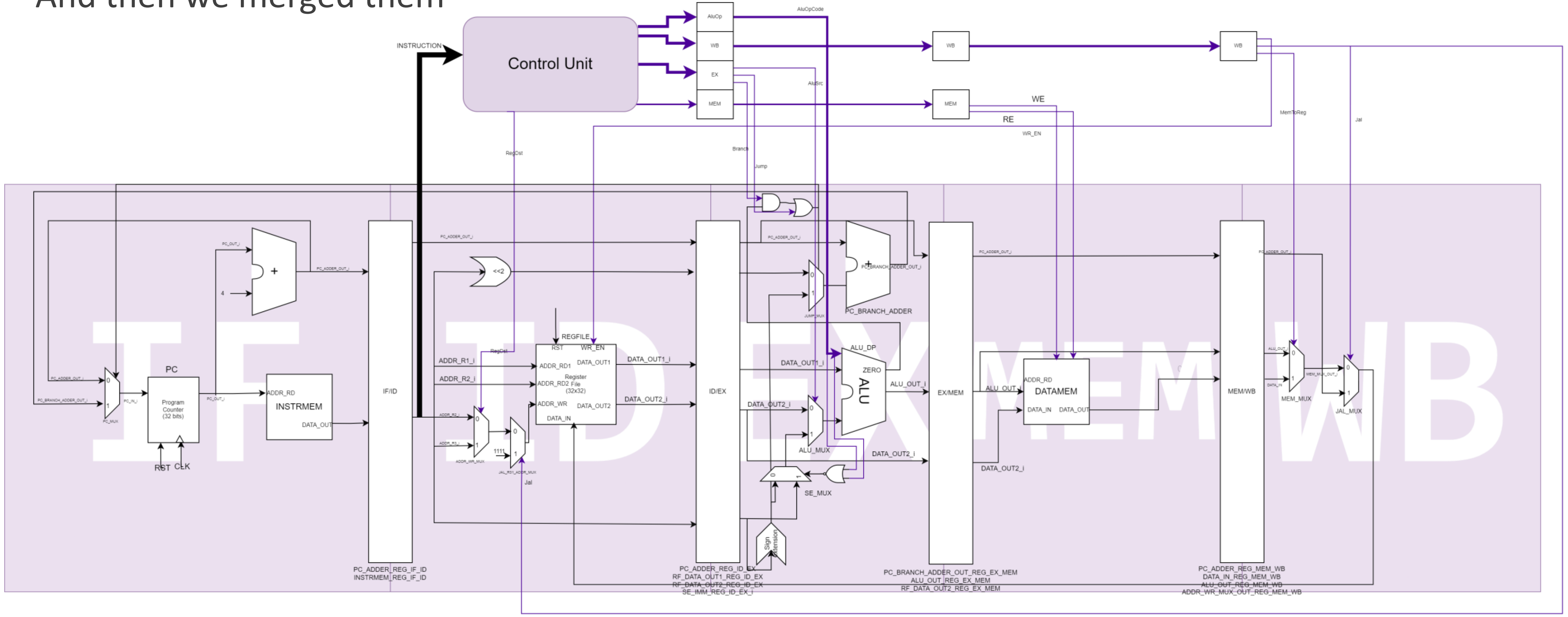




1 2 3

VHDL Design (cont.)

- And then we merged them

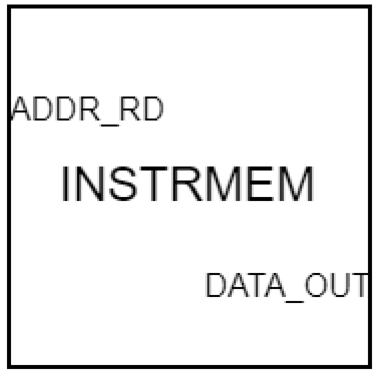


Datapath

- ALU : It is described, using a mixed behavioral-structural description in order to use the Pentium-4 adder in case of additions or subtractions.
- Register File : It is composed of 32 registers and has 2 read ports and 1 write port.
- Adders : Ripple carry adders used for computing next Program Counter.
- Multiplexers : Are used to choose between input sources in the Register File, Program Counter and ALU.
- Pipeline registers : Are used to store the results from previous stage.

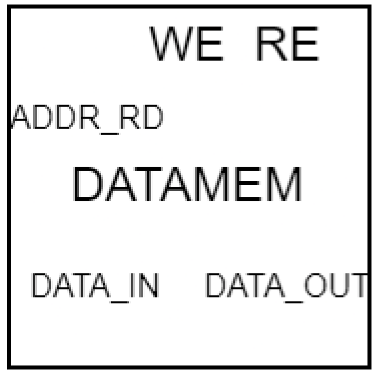
Instruction Memory

- Size : 2 kB
- Word : 32 bits
- It contains the firmware that is loaded into the microprocessor.
- Asynchronous memory.



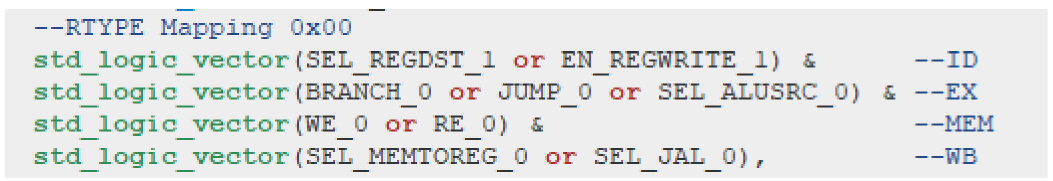
Data memory

- Size : 2 kB
- Word size: 32 bits
- It stores data coming from registers.
- Asynchronous memory.

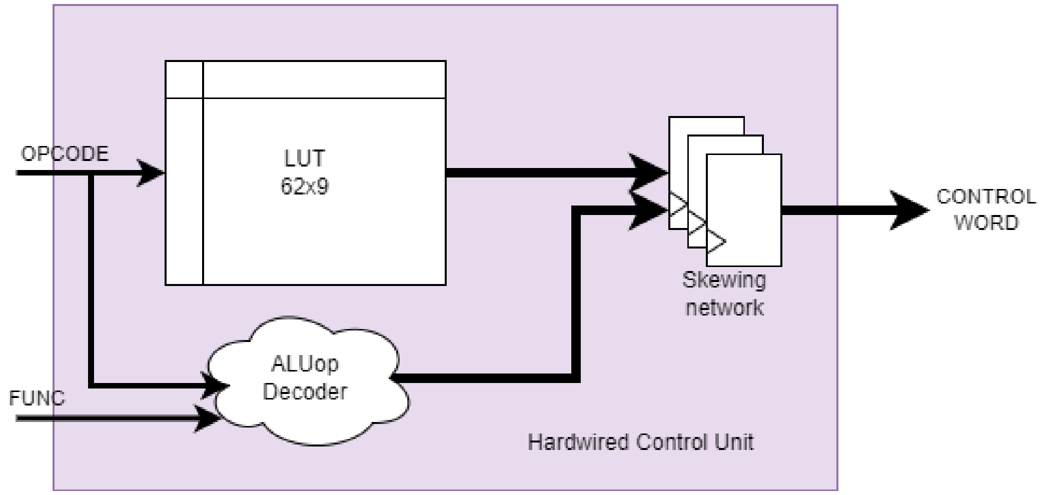


Control Unit

- LUT size : 62 lines
- Control Word: 9 bits
- Hardwired implementation
- Modular control word generation through **std_logic_vector** concatenation to improve readability during debugging phase.



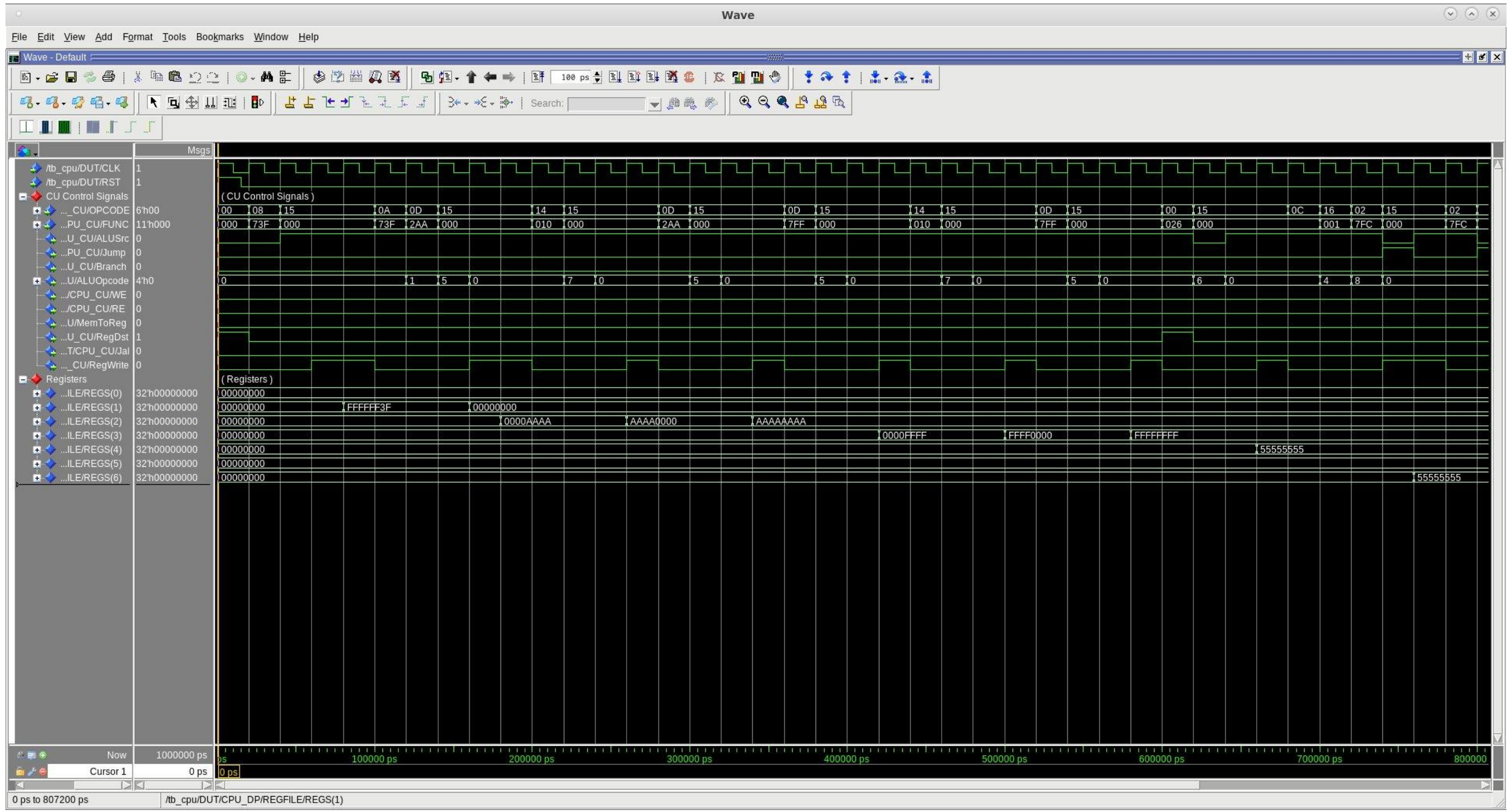
```
--RTYPE Mapping 0x00
std_logic_vector(SEL_REGDST_1 or EN_REGWRITE_1) &       --ID
std_logic_vector(BRANCH_0 or JUMP_0 or SEL_ALUSRC_0) & --EX
std_logic_vector(WE_0 or RE_0) &                        --MEM
std_logic_vector(SEL_MEMTOREG_0 or SEL_JAL_0),          --WB
```



Simulation – I-type tests

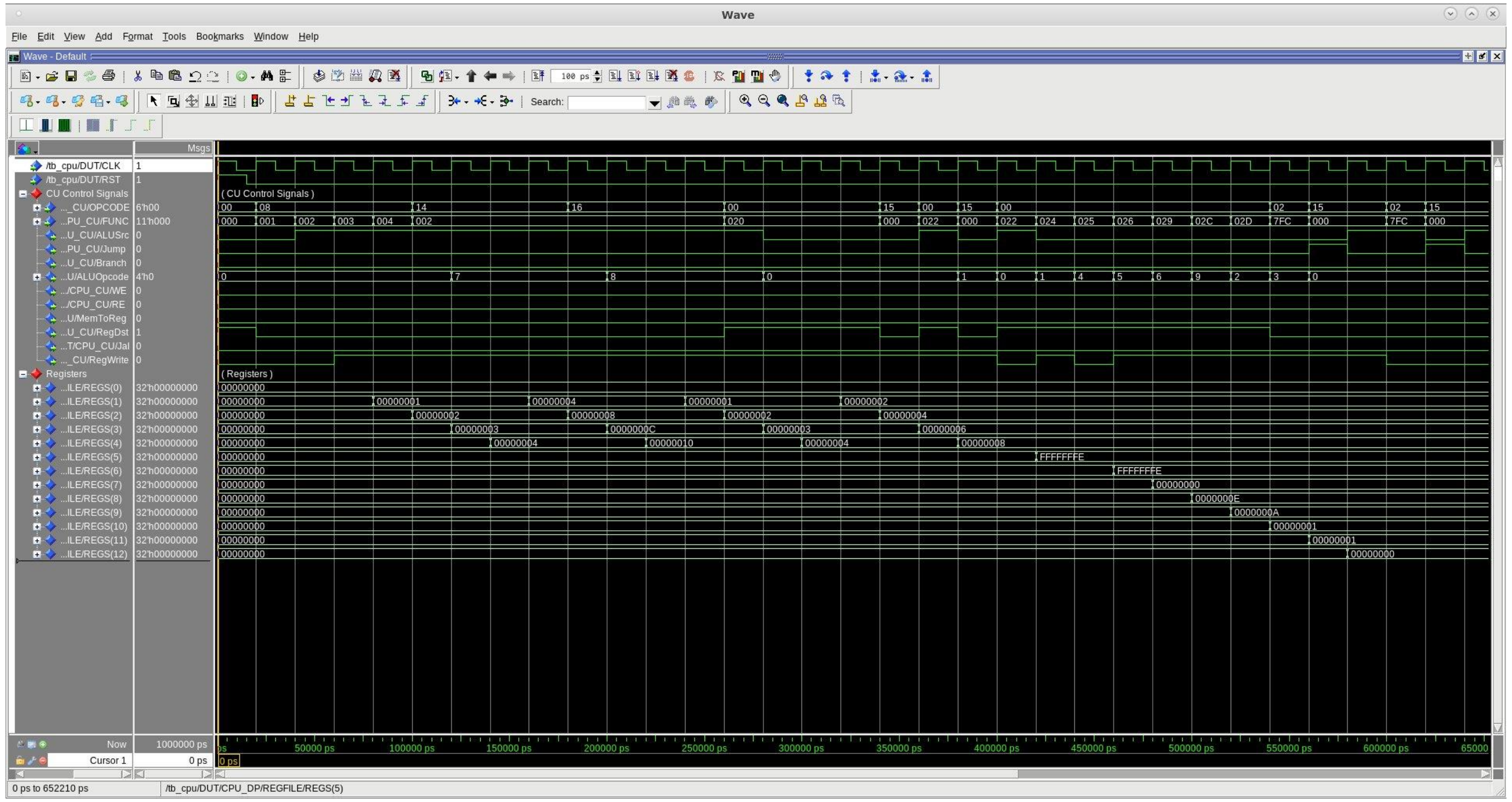
- In order to verify that the instruction set for I-type instructions works, we wrote a simple assembly program that performs:
  - Additions;
  - Subtractions;
  - Mask operations through logical instructions (i.e. AND/OR/XOR etc..).

- Eventually the program halts.



Simulation – R-type tests

- In order to verify that the instruction set for R-type instructions works, we wrote a simple assembly program that performs:
  - Additions;
  - Subtractions;
  - Shift operations.
- Eventually the program halts.



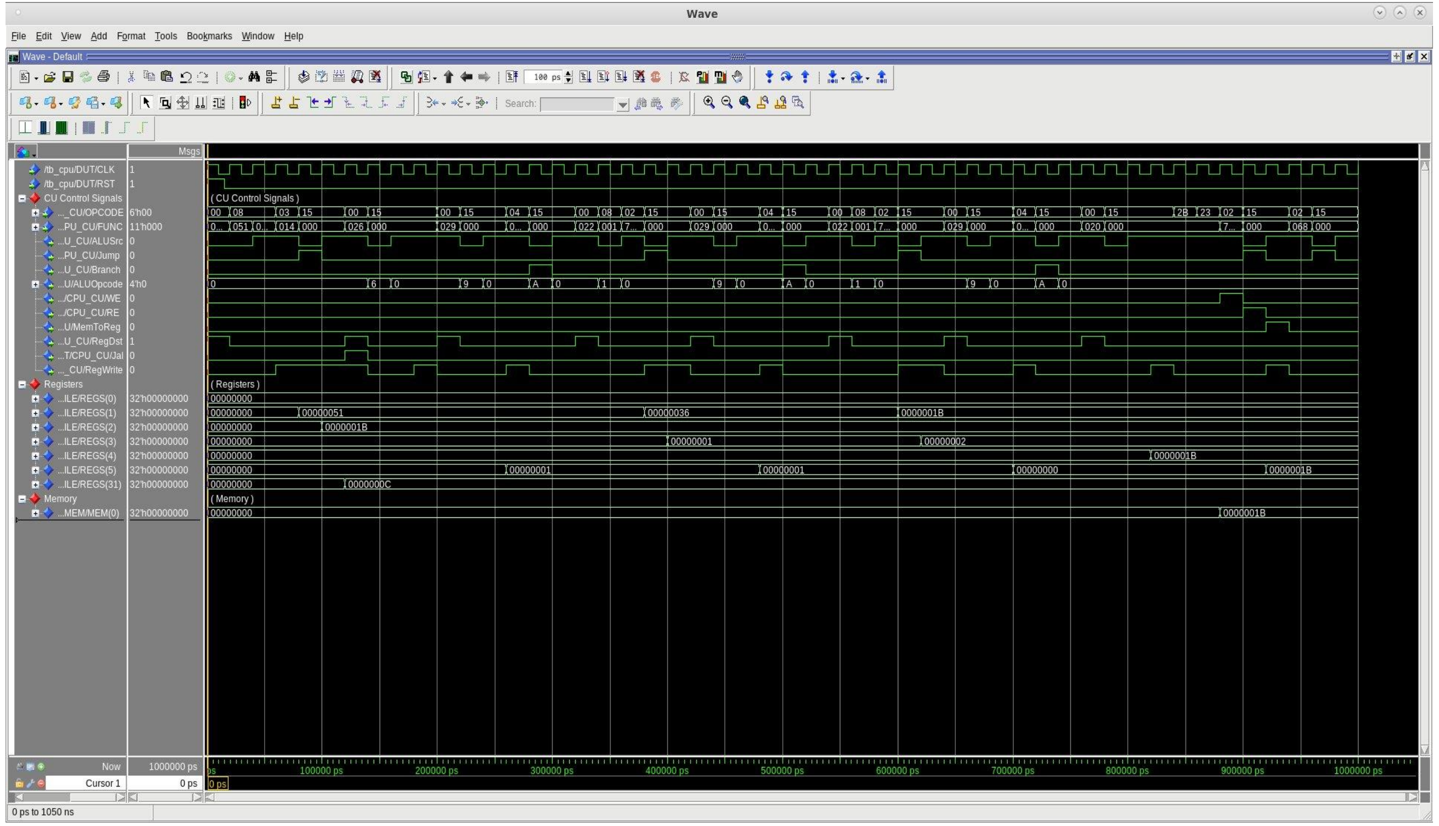
```
/tb_cpu/DUT/CLK        1
/tb_cpu/DUT/RST        1
CU Control Signals
..._CU/OPCODE          6'h00
...PU_CU/FUNC          11'h000
...U_CU/ALUSrc         0
...PU_CU/Jump          0
...U_CU/Branch         0
...U/ALUOpcode         4'h0
.../CPU_CU/WE          0
.../CPU_CU/RE          0
...U/MemToReg          0
...U_CU/RegDst         1
...T/CPU_CU/Jal        0
..._CU/RegWrite        0
Registers
...ILE/REGS(0)         32'h00000000
...ILE/REGS(1)         32'h00000000
...ILE/REGS(2)         32'h00000000
...ILE/REGS(3)         32'h00000000
...ILE/REGS(4)         32'h00000000
...ILE/REGS(5)         32'h00000000
...ILE/REGS(6)         32'h00000000
...ILE/REGS(7)         32'h00000000
...ILE/REGS(8)         32'h00000000
...ILE/REGS(9)         32'h00000000
...ILE/REGS(10)        32'h00000000
...ILE/REGS(11)        32'h00000000
...ILE/REGS(12)        32'h00000000

Now        1000000 ps
Cursor 1   0 ps

0 ps to 652210 ps    /tb_cpu/DUT/CPU_DP/REGFILE/REGS(5)
```

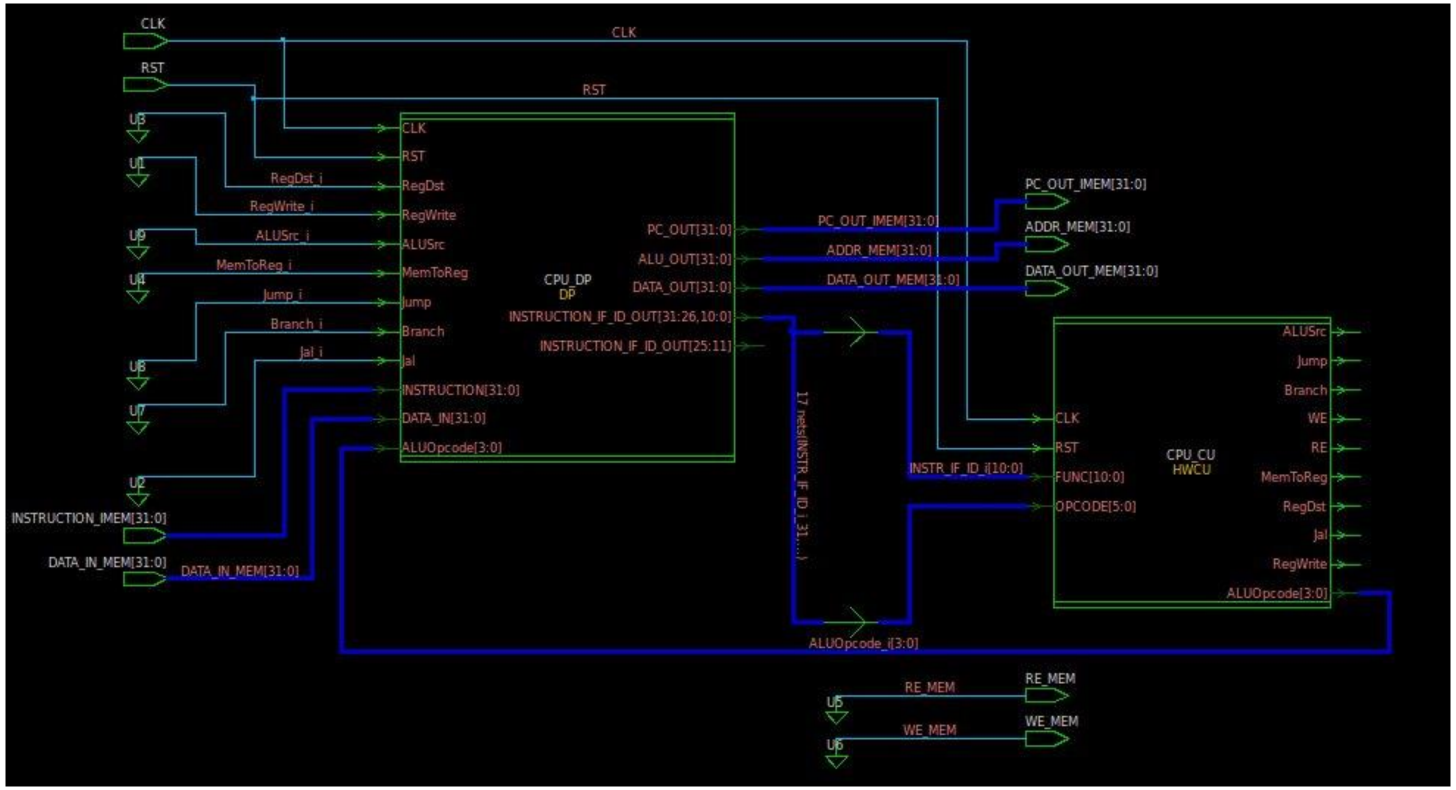
Simulation – Iterative Division

- In conclusion, to create a more complete program we went through the iterative division simulation.
- The algorithm is executed after a call to a procedure (JAL) and performs 81/27.
- When the procedure ends it stores the result into memory and load again into another register. Then it returns to the caller.
- Eventually the program halts.



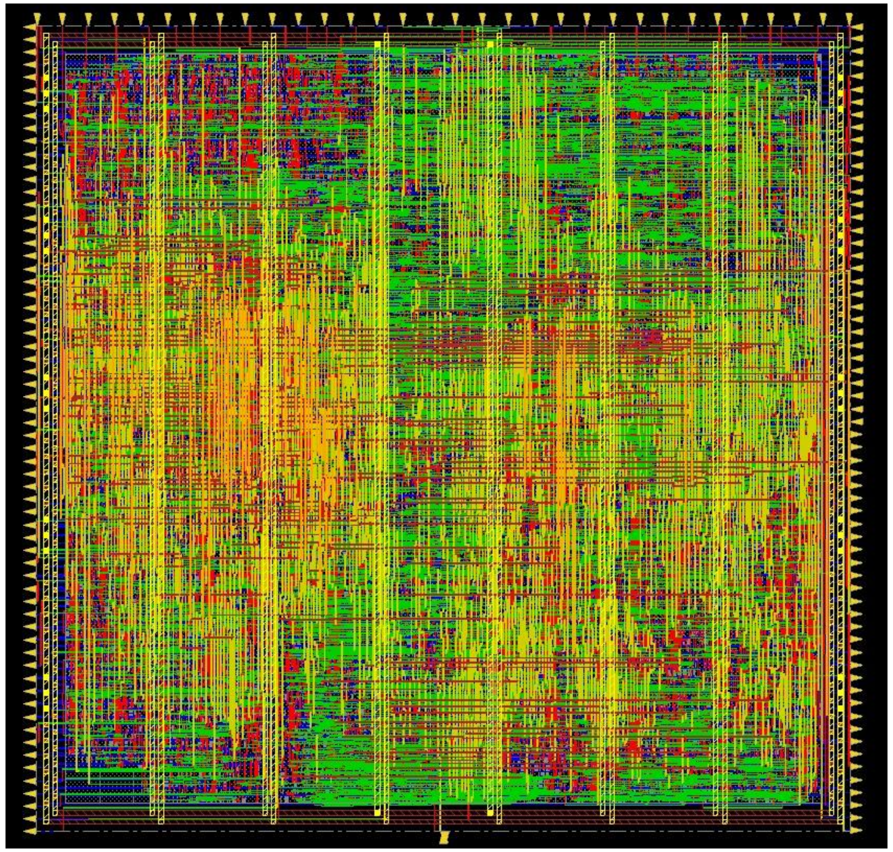
Synthesis

- Starting from a clock frequency of 50 MHz we reached up to 1 GHz, without major synthesis optimizations.
- Eventually 2 GHz clock frequency goal was achieved using aggressive optimization flow with the usage of **compile_ultra, set_dont_touch** to avoid removal of skewing registers and **set_max_delay** to constrain the quasi-critical paths.



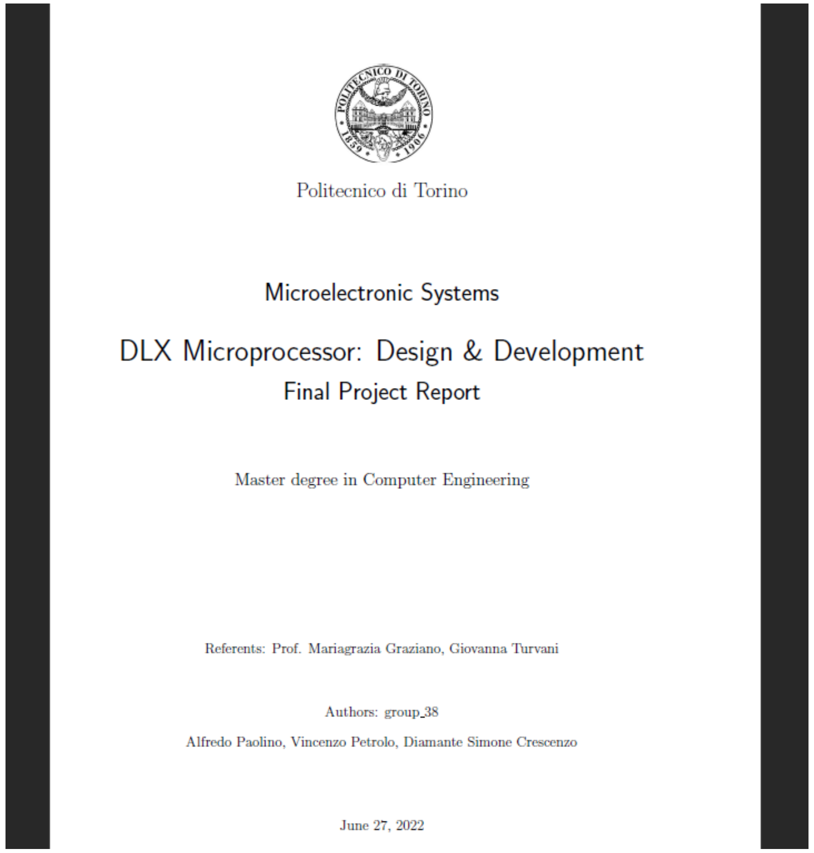
Physical layout

- We followed the flow for the place & route phase.
- After the post-routing optimization phase, we run a static power analysis and got an estimated power of **500 mW** running at 1 GHz.
- The total power estimated from PrimeTime (1 GHz) was: **426 mW**



Technical report

- The final step was to produce a report describing more in detail what is shown in this presentation.


Politecnico di Torino

Microelectronic Systems

DLX Microprocessor: Design & Development

Final Project Report

Master degree in Computer Engineering

Referents: Prof. Mariagrazia Graziano, Giovanna Turvani

Authors: group_38

Alfredo Paolino, Vincenzo Petrolo, Diamante Simone Crescenzo

June 27, 2022

Possible improvements

For a future PRO version, some of the possible improvements:

- Data Hazard Unit
- Branch Prediction Unit
- Floating Point Unit
- Instruction/Data cache
- Extended Instruction set
- Reducing to 1 the Branch delay slot
- Verification using UVM

Thanks for your attention!