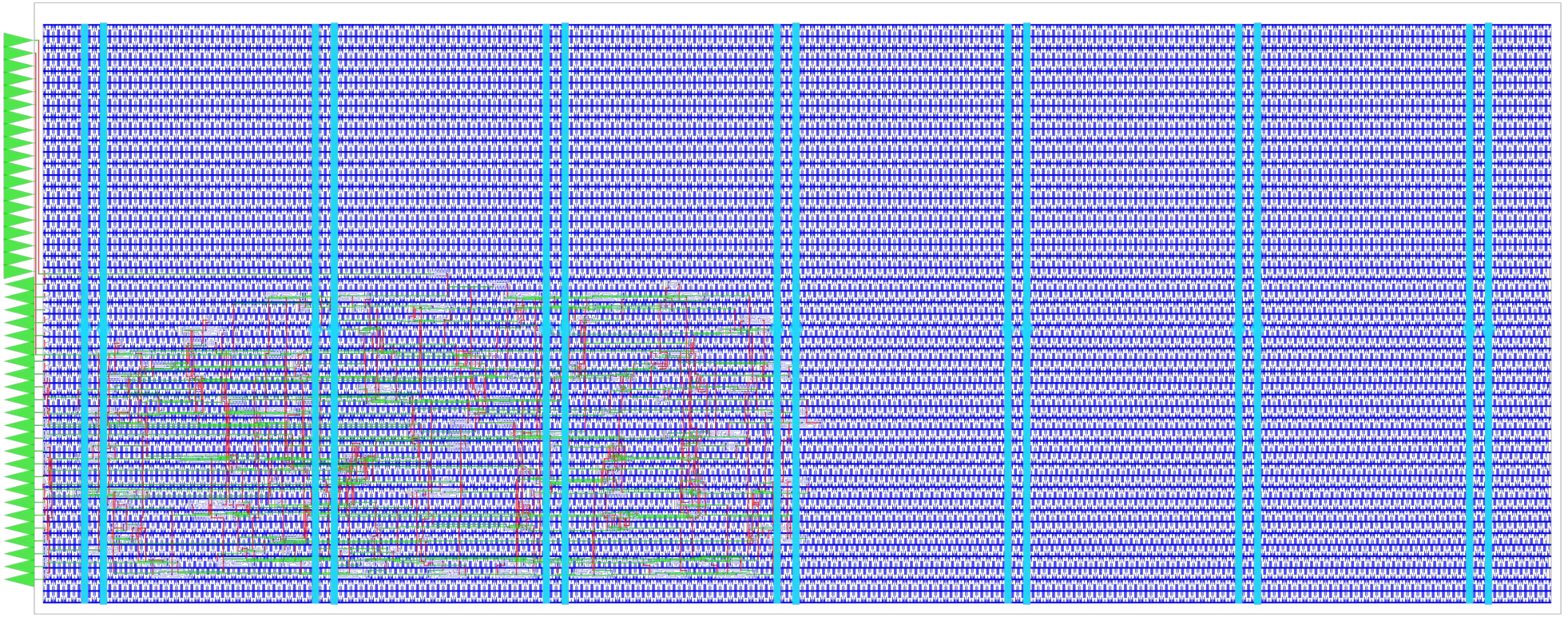
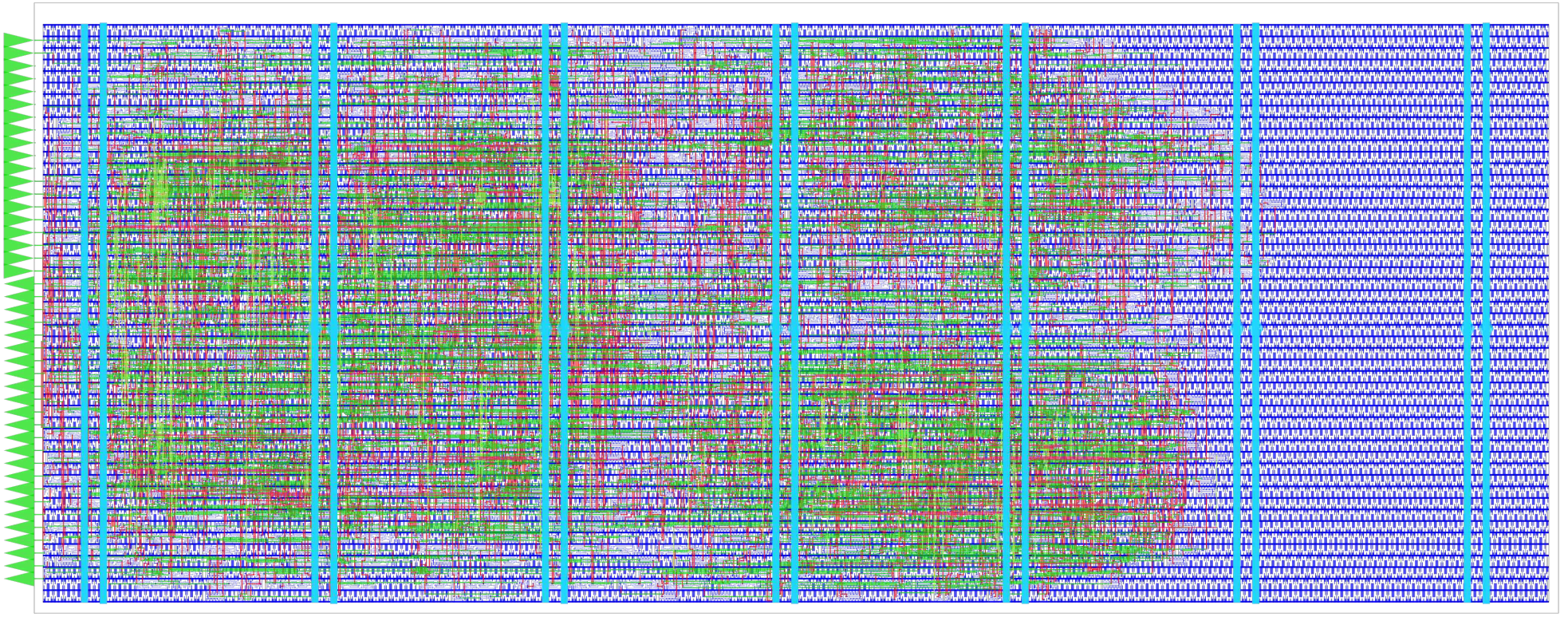
8 bit CPU + eFPGA/sorter

- Members
  - Philippos, Deepak, Moritz, Malte, Enrica
- Components
  - SAP-3 processor (stand-alone slot)
  - Sorter accelerator (stand-alone slot)
  - eFPGA part

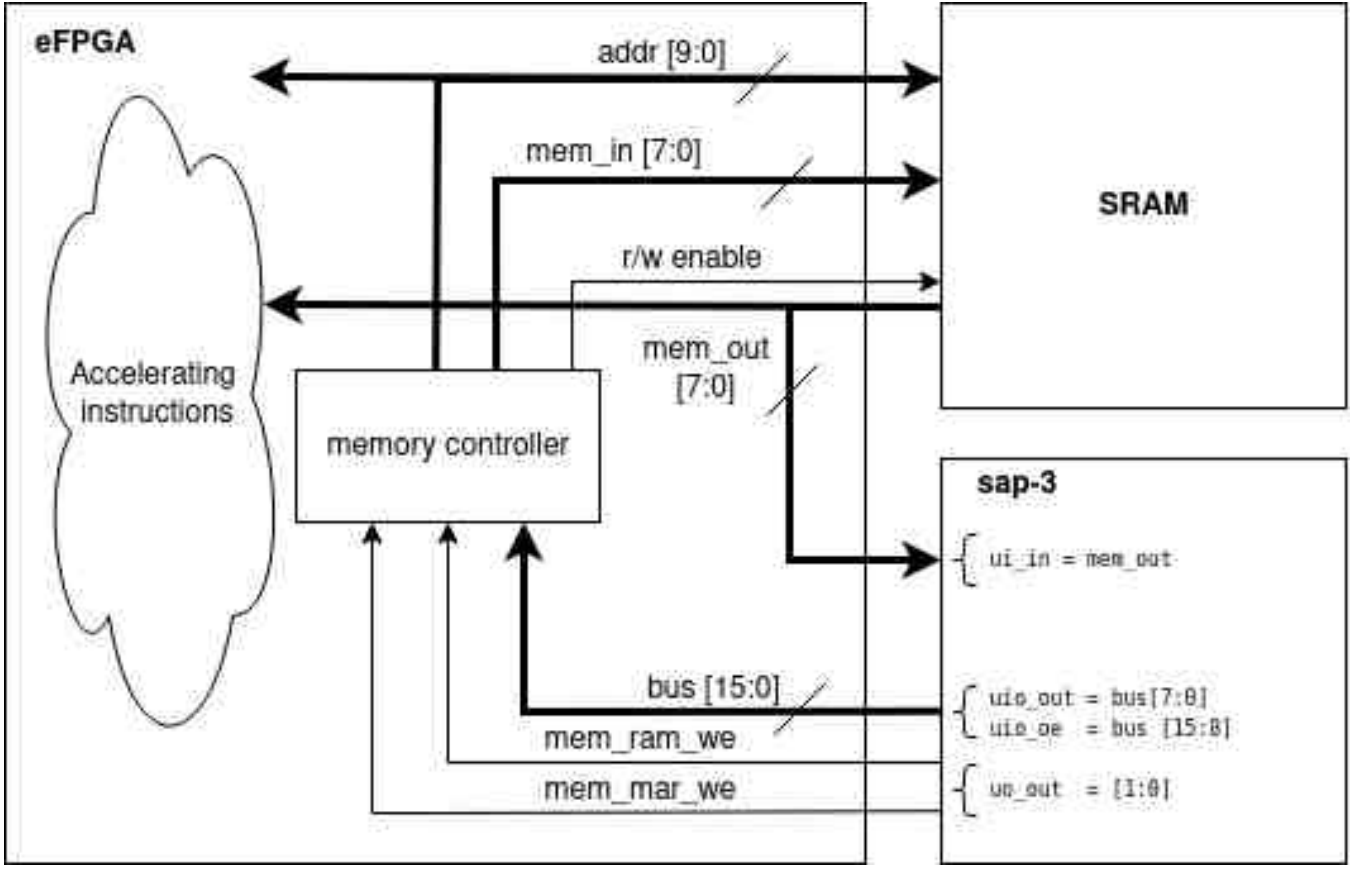
Physical implementation SAP-1



Physical implementation SAP-3



SAP-3


eFPGA
addr [9:0]
mem_in [7:0]
r/w enable
mem_out
[7:0]
Accelerating
instructions
memory controller
SRAM
sap-3
ui_in = mem_out
bus [15:0]
uio_out = bus[7:0]
uio_oe = bus [15:8]
mem_ram_we
mem_mar_we
uo_out = [1:0]

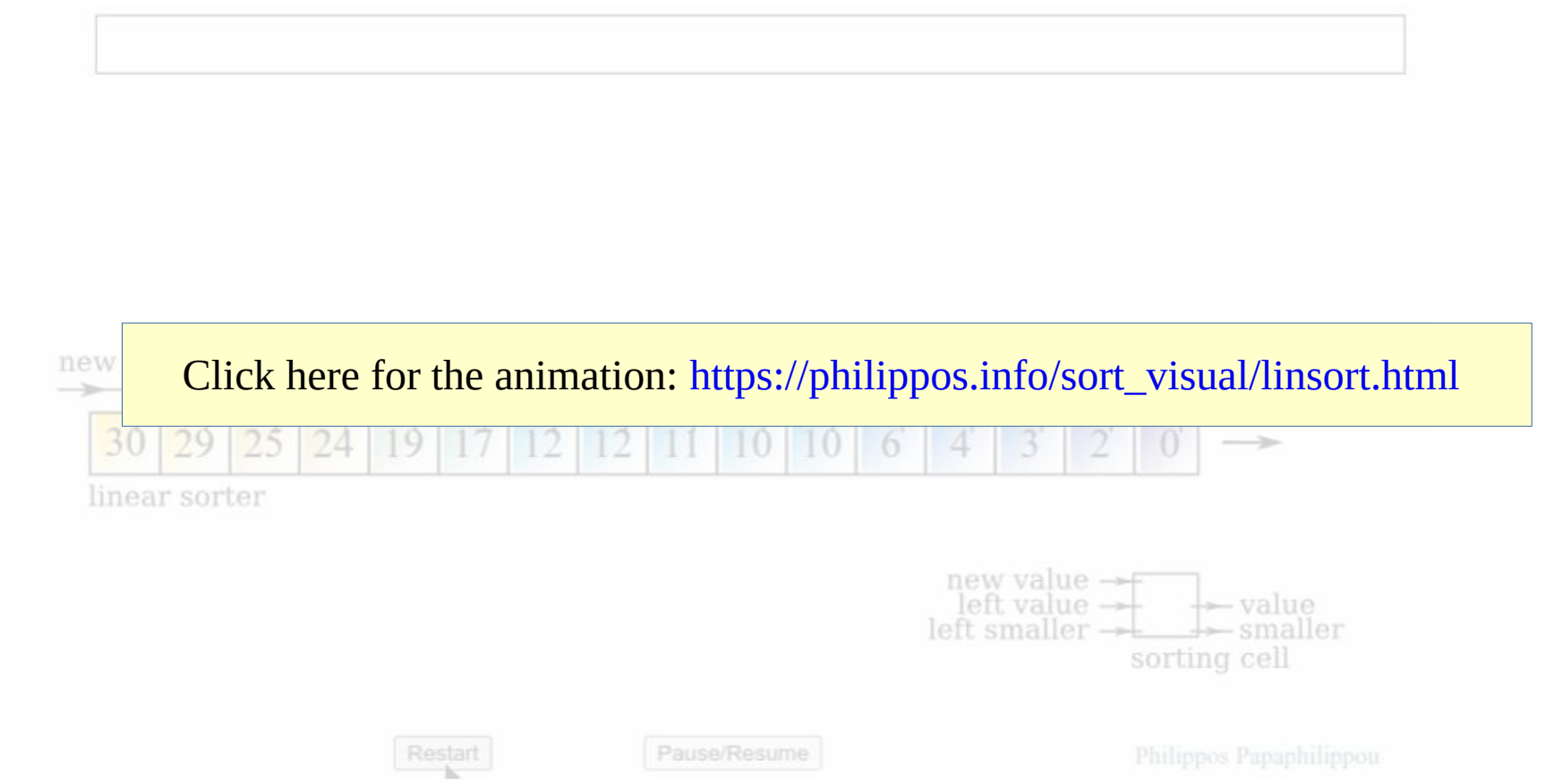
“Top Sorter”

- Initially a backup project for “8-bit CPU + custom acc.”
  - Works stand-alone
  - Extension of “8-bit CPU + custom acc.”

The task

- Linear sorter
  - “Parallel” version of insertion sort

Animation from philippos.info

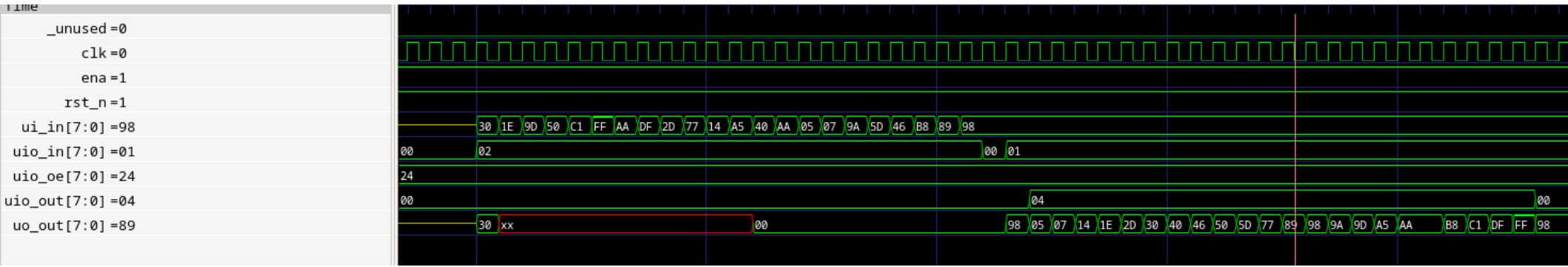


Click here for the animation: https://philippos.info/sort_visual/linsort.html

Testbench

```
FST info: dumpfile /home/philippos/Documents/Git/heichips25_top_sorter/tb/sim_build/heichips25_top_sorter.fst opened for output.
Inserting number 166
Inserting number 69
Inserting number 97
Inserting number 1
Inserting number 84
Inserting number 94
Inserting number 208
Inserting number 3
Inserting number 251
Inserting number 130
Inserting number 185
Inserting number 181
Inserting number 89
Inserting number 74
Inserting number 249
Inserting number 131
Inserting number 86
Inserting number 85
Inserting number 155
Inserting number 56
Inserting number 220
Inserting number 201
Exporting 1
Exporting 3
Exporting 56
Exporting 69
Exporting 74
Exporting 84
Exporting 85
Exporting 86
Exporting 89
Exporting 94
Exporting 97
Exporting 130
Exporting 131
```

Testbench waveform

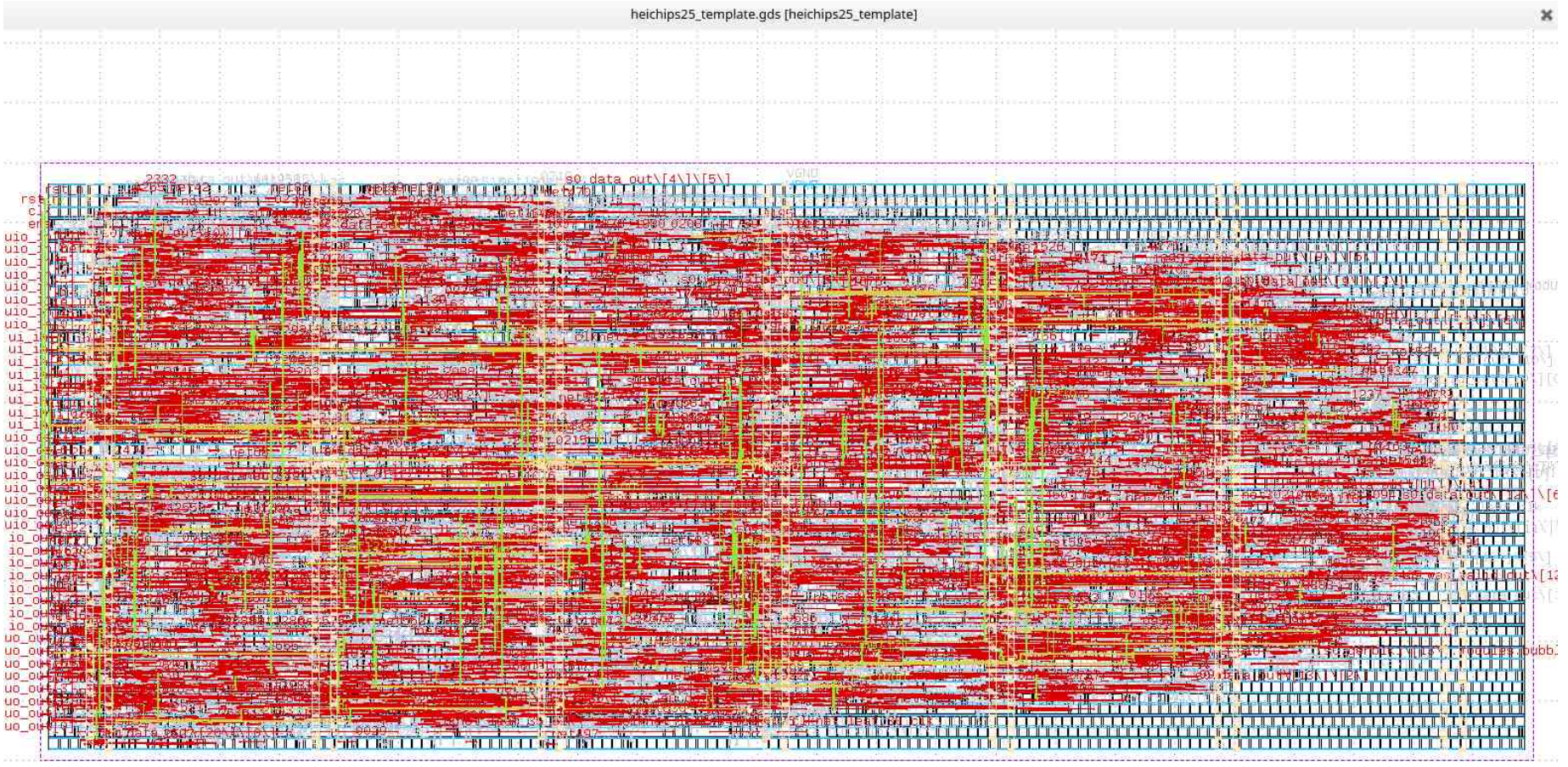


Final design specs

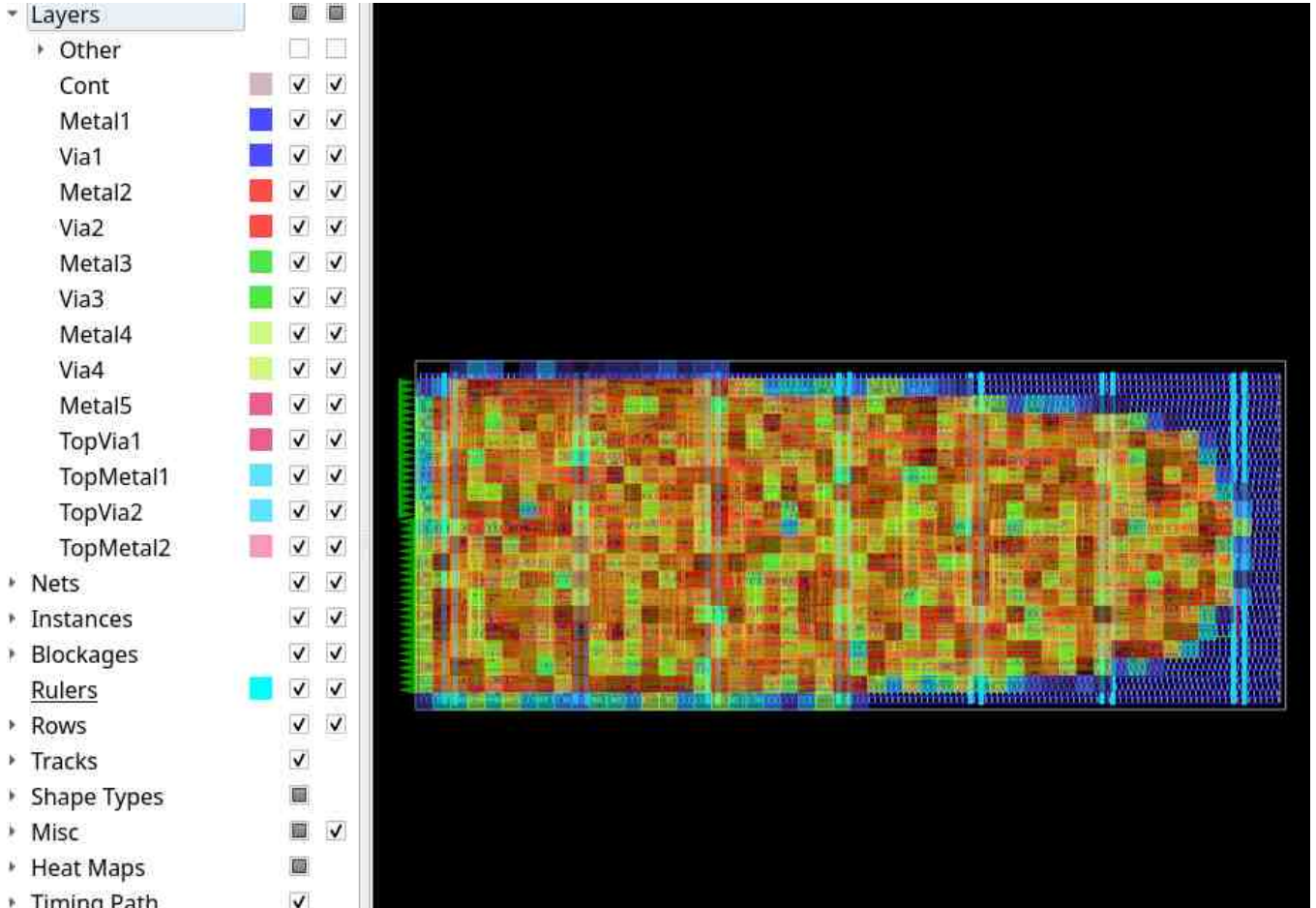
- N = 22 (sorted list size/ top N)
- Data width = 8 bits
- 100 MHz

    → Throughput: 100 MB/s

Klayout



OpenRoad


Layers
Other
Cont
Metal1
Via1
Metal2
Via2
Metal3
Via3
Metal4
Via4
Metal5
TopVia1
TopMetal1
TopVia2
TopMetal2
Nets
Instances
Blockages
Rulers
Rows
Tracks
Shape Types
Misc
Heat Maps
Timing Path

Ongoing work

- SAP-3
  - eFPGA custom instr. verification
- Sort
  - Optimisation (increase N at 100 MHz+)
    - Systolic version
    - No need to have the entire list sorted on next cycle
  - More functionality
    - Top N
    - Merge-version for sorting the entire SRAM with CPU