How much score could a CoreScore score if a CoreScore could score cores?

Spring 2022 RISC-V week - Paris 2022.05.04

Who are we?

Klas Nordmark

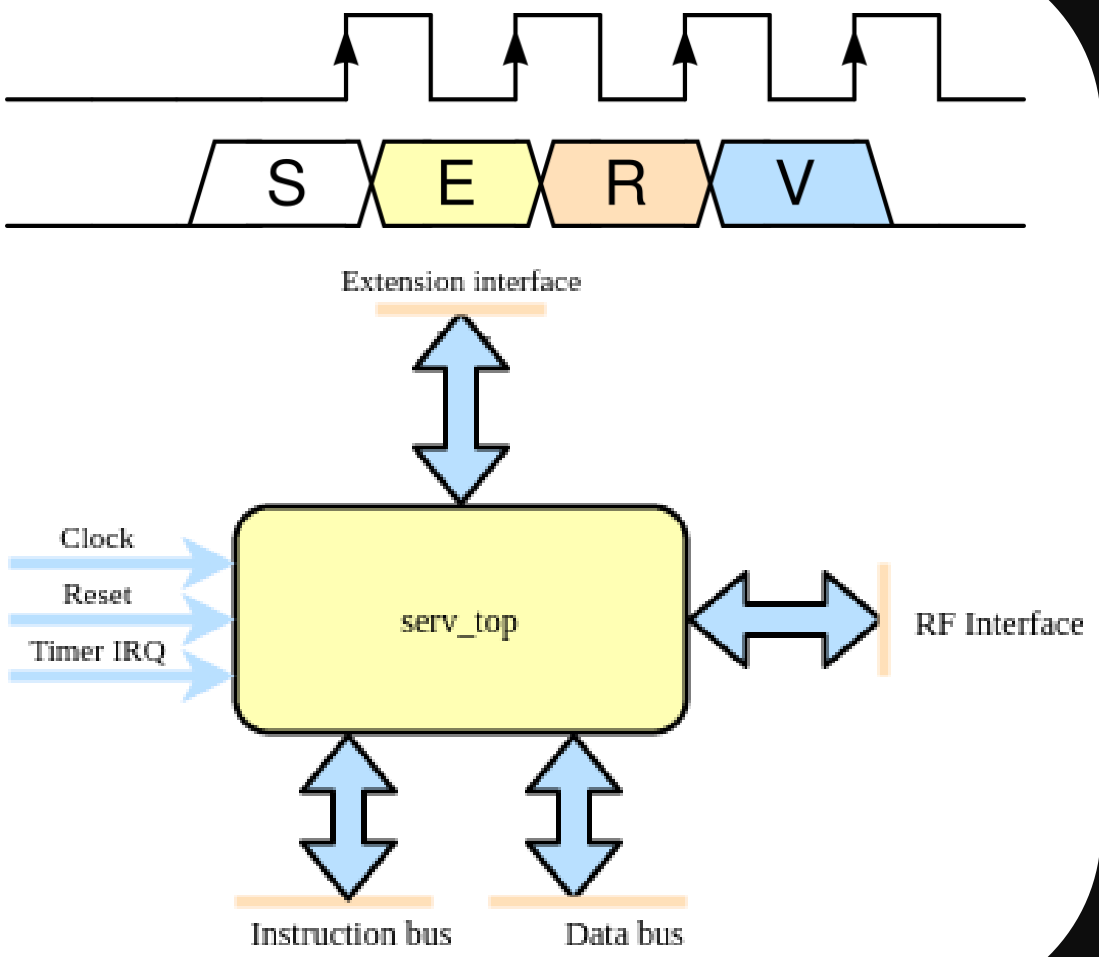
FPGA & Embedded SW Consultant at 2550 Engineering

Skriv text här

Olof Kindgren

FPGA & Embedded SW Consultant at Qamcom Research & Technology

Features



- RV32I
- Wishbone data bus
- Wishbone instruction bus
- CFU extension interface
- Formally verified with riscv-formal
- BSD licensed
- + optionally
  - CSR extension
  - A bit of the privilege spec (enough to run Zephyr)
  - M extension
  - C extension

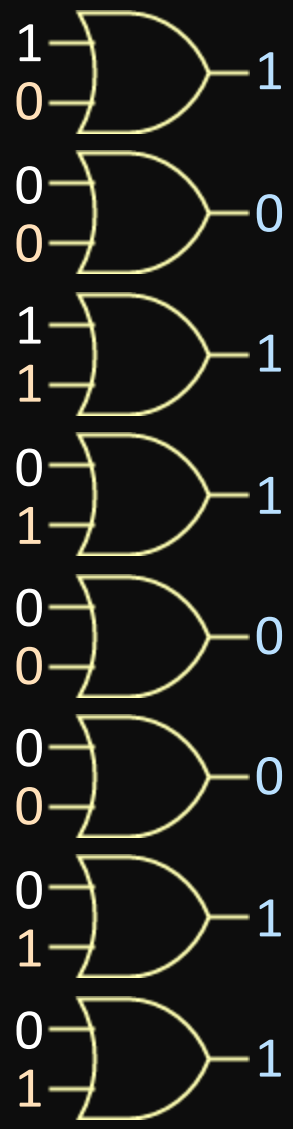
Zeeshan Rafique
M extension, GSoC 2021

Abdul Wadood
C extension LFX 2022

What's a bit-serial CPU?

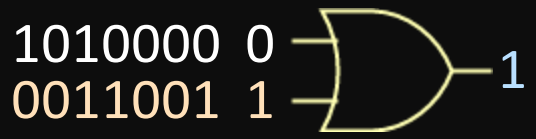
Bit-serial

Skriv 0xA0 | 0x33 = 0xB3



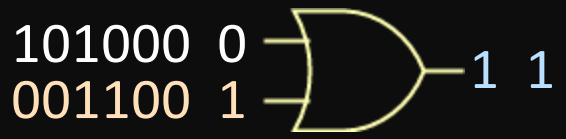
Bit-serial

Skriv 0xA0 | 0x33 = 0xB3



Bit-serial

Skriv 0xA0 | 0x33 = 0xB3



Bit-serial

Skriv 0xA0 | 0x33 = 0xB3

Bit-serial

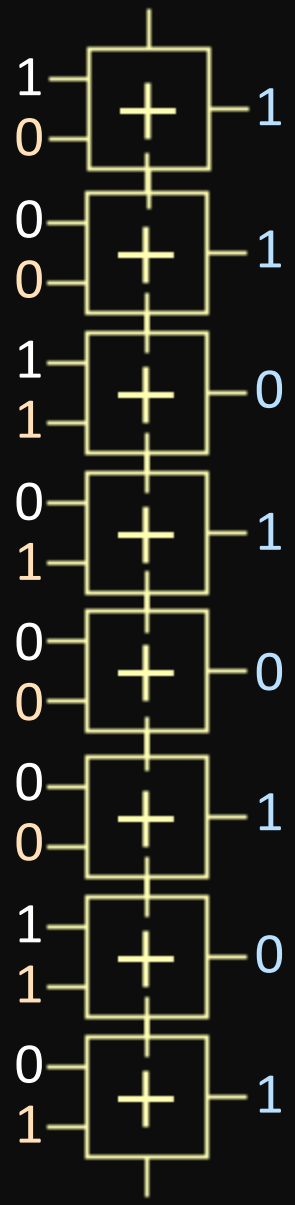
Skriv 0xA0 | 0x33 = 0xB3

Bit-serial

Skriv 0xA0 | 0x33 = 0xB3

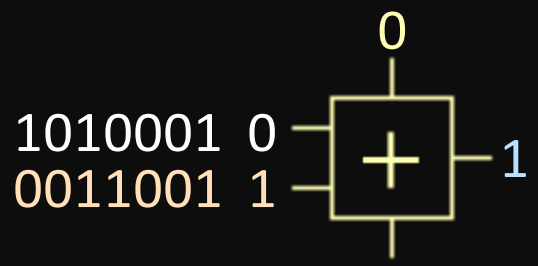
Bit-serial

Skriv 0xA2 + 0x33 = 0xD5



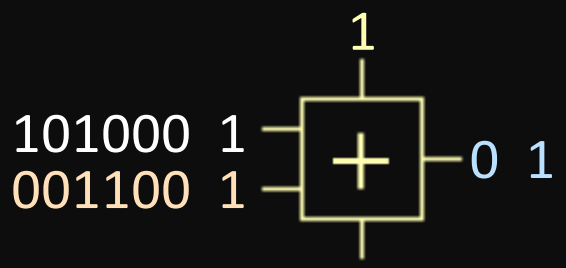
Bit-serial

Skriv 0xA0 | 0x33 = 0xB3



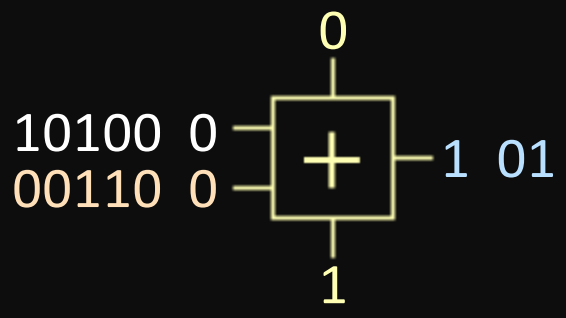
Bit-serial

Skriv 0xA0 | 0x33 = 0xB3

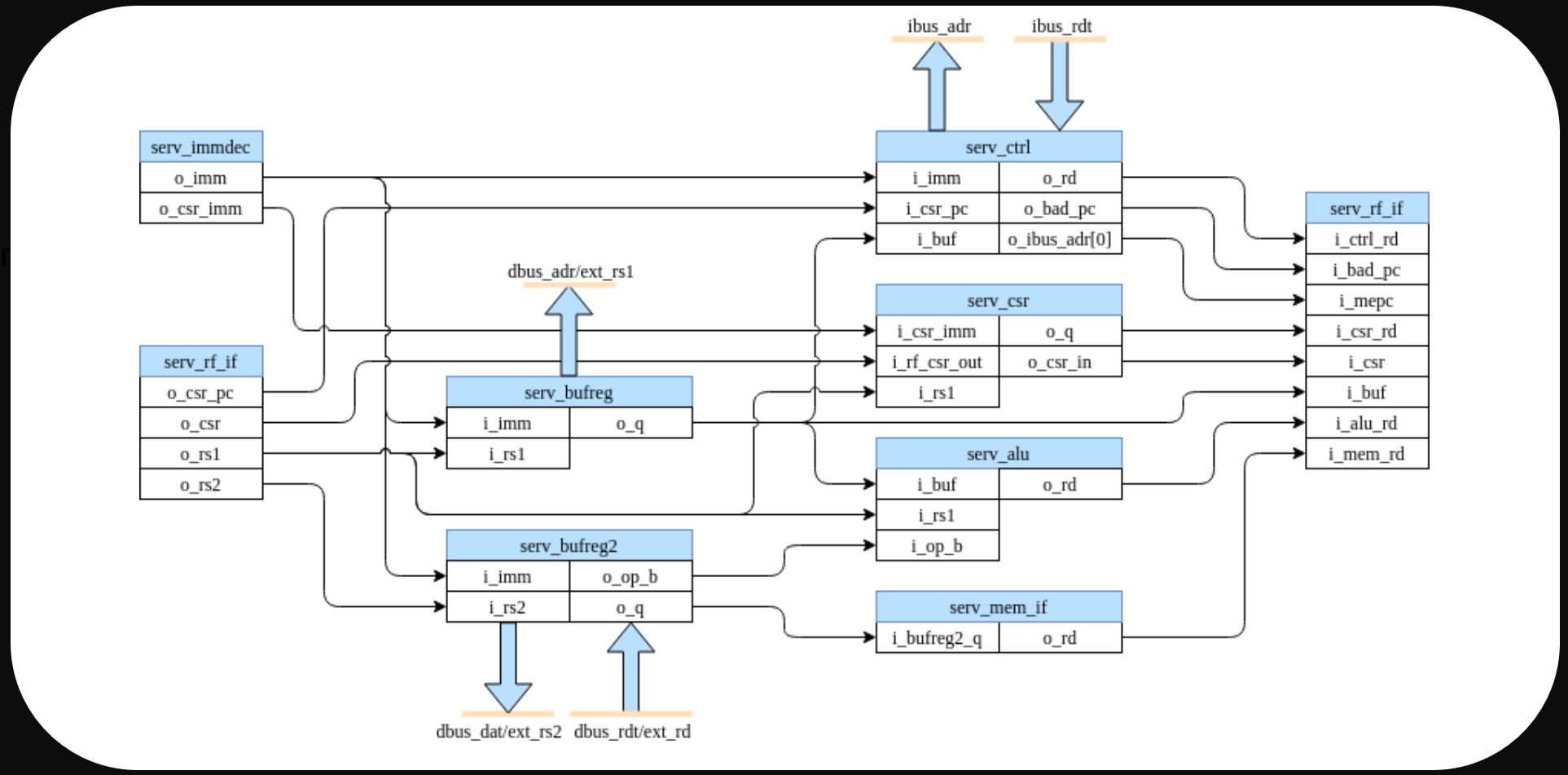


Bit-serial

Skriv 0xA0 | 0x33 = 0xB3

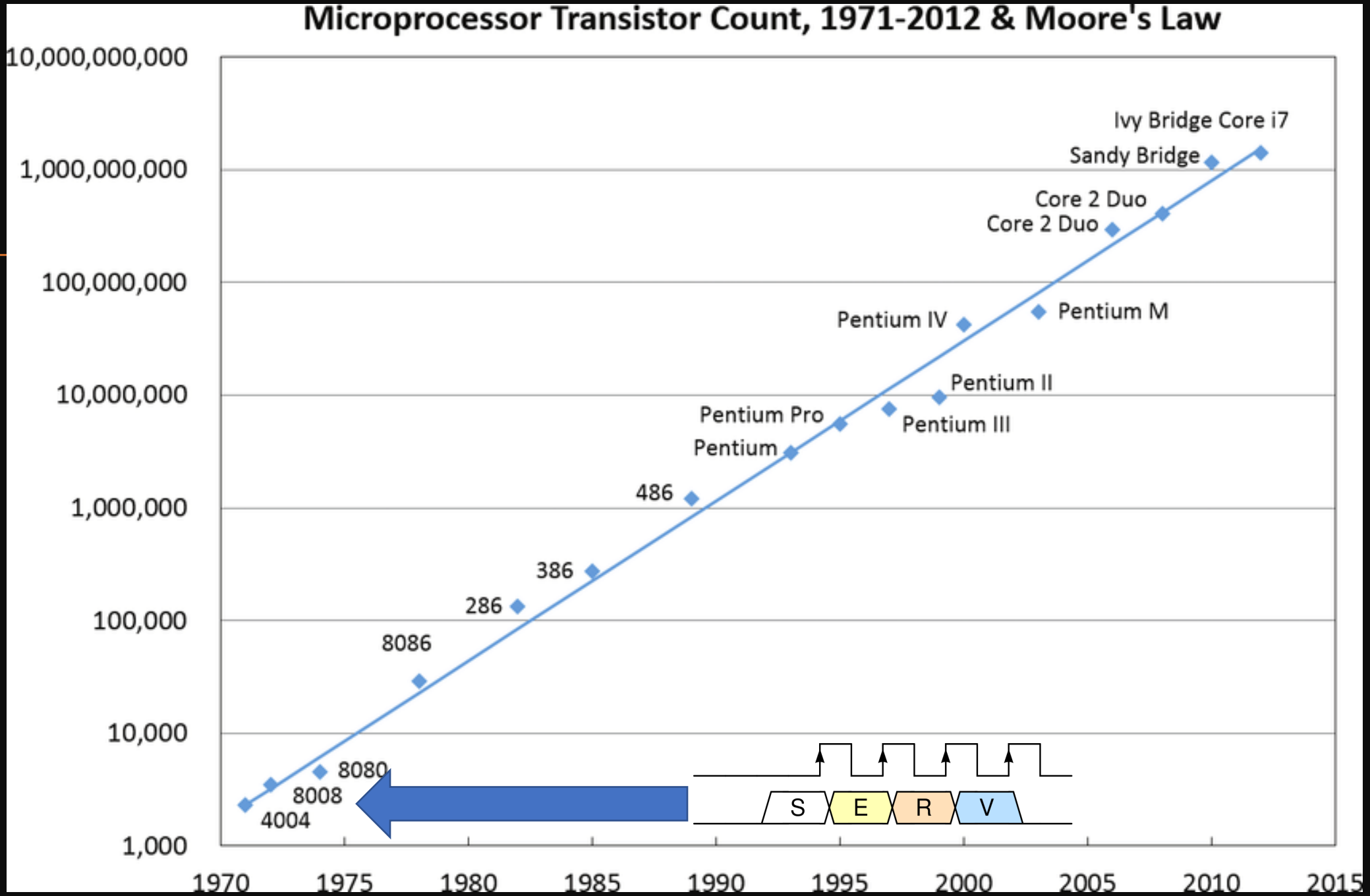


Bit-serial


ibus_adr
ibus_rdt
serv_immdec
o_imm
o_csr_imm
serv_ctrl
i_imm
o_rd
i_csr_pc
o_bad_pc
i_buf
o_ibus_adr[0]
serv_rf_if
i_ctrl_rd
i_bad_pc
i_mepc
i_csr_rd
i_csr
i_buf
i_alu_rd
i_mem_rd
dbus_adr/ext_rs1
serv_csr
i_csr_imm
o_q
i_rf_csr_out
o_csr_in
i_rs1
serv_rf_if
o_csr_pc
o_csr
o_rs1
o_rs2
serv_bufreg
i_imm
o_q
i_rs1
serv_alu
i_buf
o_rd
i_rs1
i_op_b
serv_bufreg2
i_imm
o_op_b
i_rs2
o_q
serv_mem_if
i_bufreg2_q
o_rd
dbus_dat/ext_rs2
dbus_rdt/ext_rd

Size

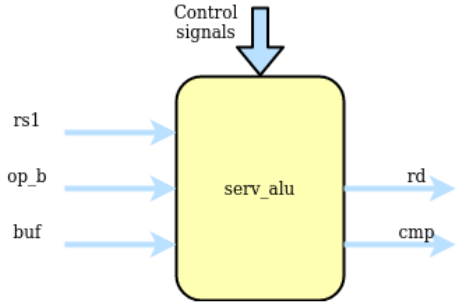
	Minimal		**Standard**	
	LUT	FF	LUT	FF
Lattice iCE40	198	164	261	182
AMD Artix-7	125	164	170	182
Intel Cyclone 10	239	164	297	182


Microprocessor Transistor Count, 1971-2012 & Moore's Law
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
1970
1975
1980
1985
1990
1995
2000
2005
2010
2015
Ivy Bridge Core i7
Sandy Bridge
Core 2 Duo
Core 2 Duo
Pentium IV
Pentium M
Pentium II
Pentium III
Pentium Pro
Pentium
486
386
286
8086
8080
8008
4004
S
E
R
V

Documentation

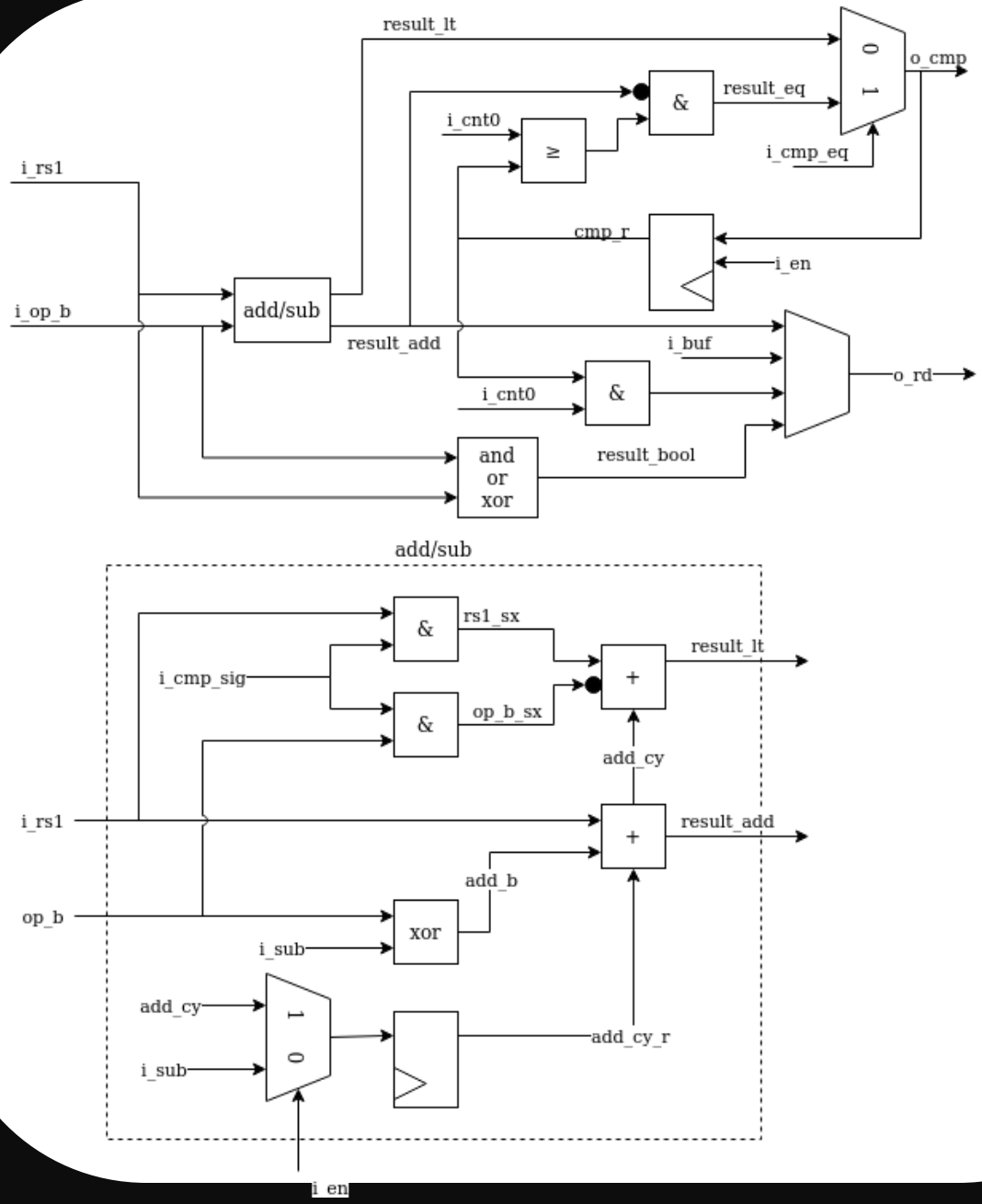
Documentation

serv_alu

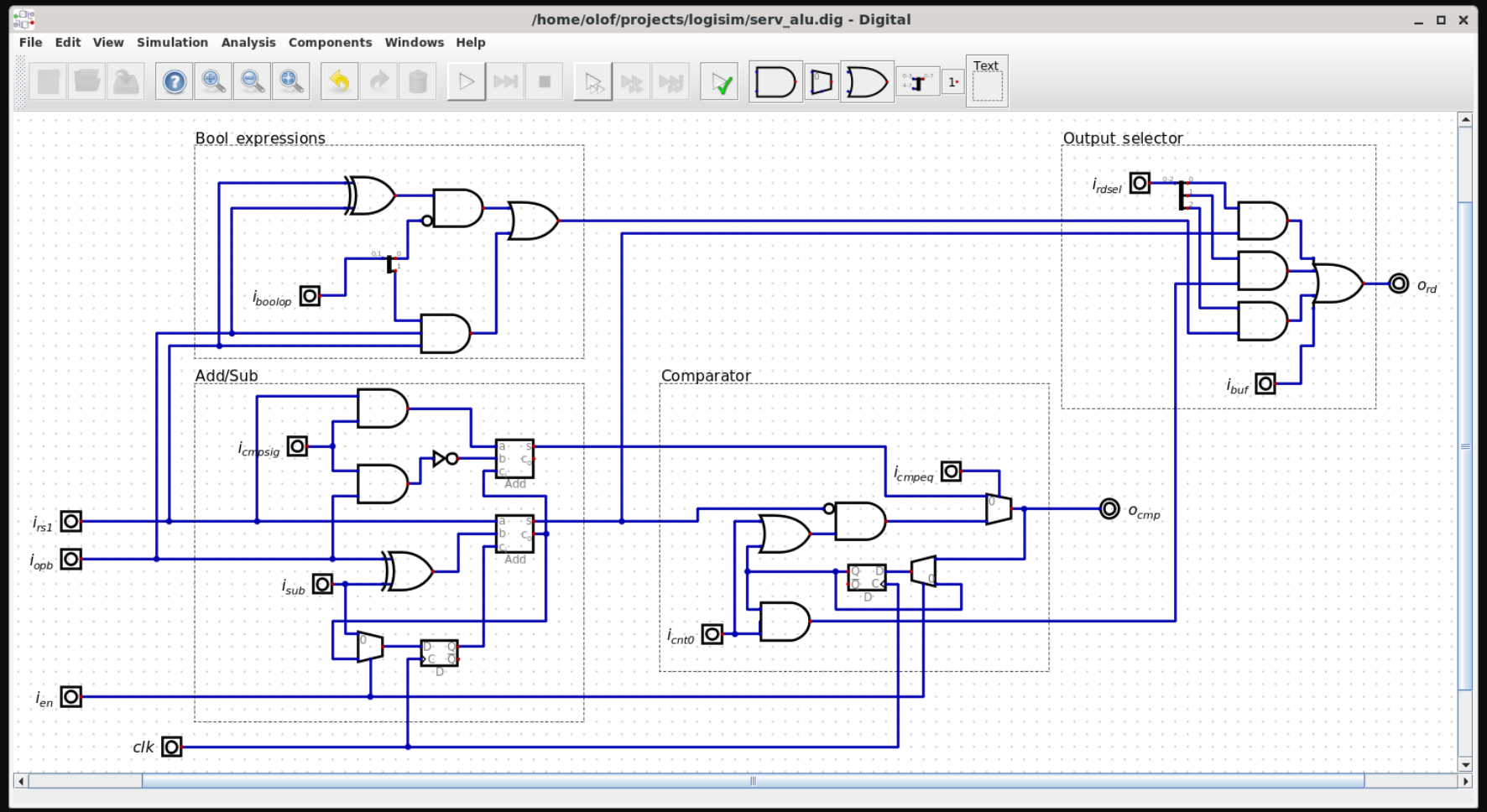


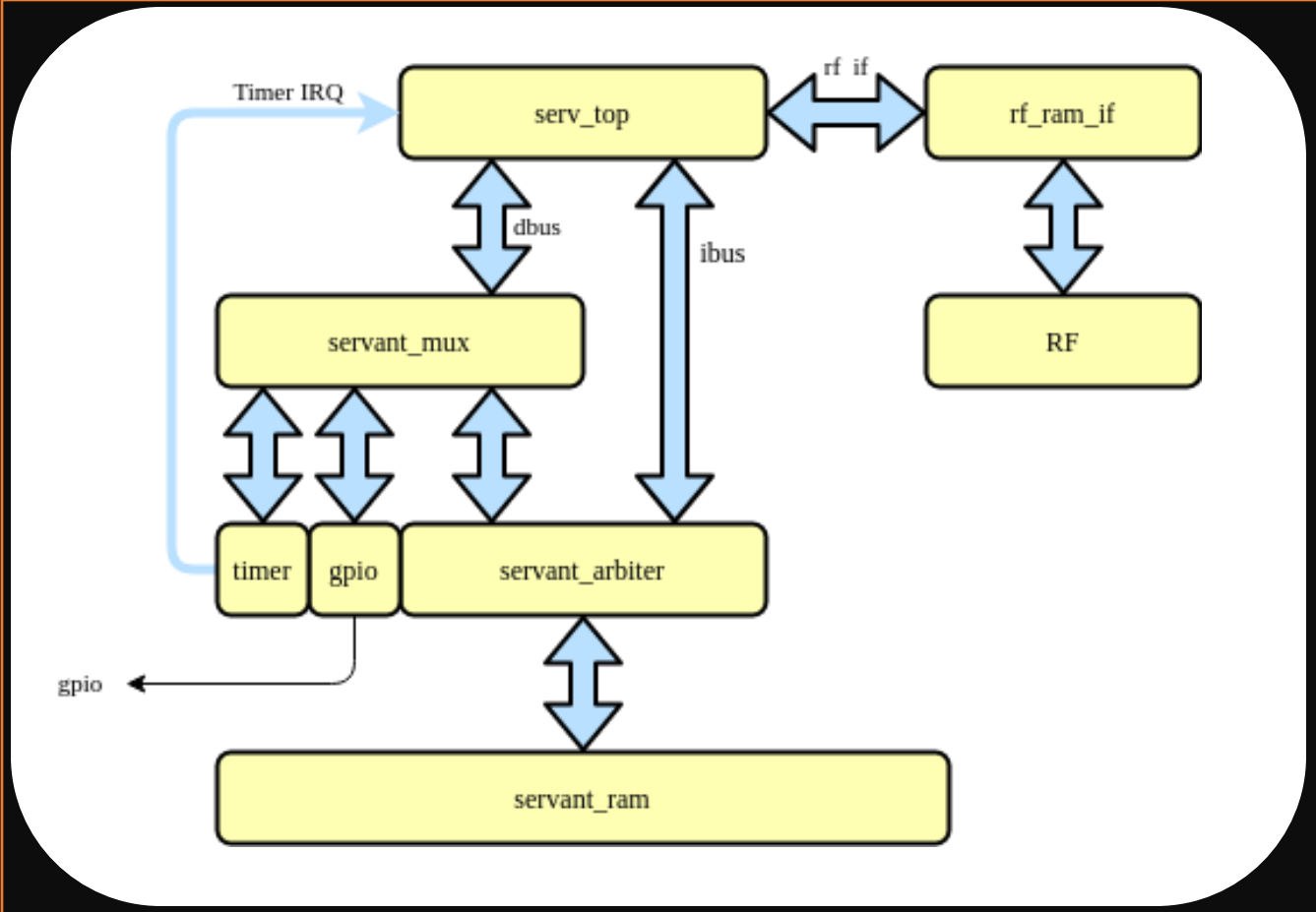
serv_alu handles alu operations. The first input operand (A) comes from i_rs1 and the second operand (B) comes from i_rs2 or i_imm depending on the type of operation. The data passes through the add/sub or bool logic unit and finally ends up in o_rd to be written to the destination register. The output o_cmp is used for conditional branches to decide whether or not to take the branch.

The add/sub unit can do additions A+B or subtractions A-B by converting it to A+B+1. Subtraction mode (i_sub = 1) is also used for the comparisions in the slt* and conditional branch instructions. The +1 used in subtraction mode is done by preloading the carry input with 1. Less-than comparisons are handled by converting the expression A<B to A-B<0 and checking the MSB, which will be set when the result is less than 0. This however requires sign-extending the operands to 33-bit inputs. For signed operands (when i_cmp_sig is set), the extra bit is the same as the MSB. For unsigned, the extra bit is always 0. Because the ALU is only active for 32 cycles, the 33rd bit must be calculated in parallel to the ordinary addition. The result from this operations is available in result_lt. For equality checks, result_eq checks that all bits are 0 from the subtraction.



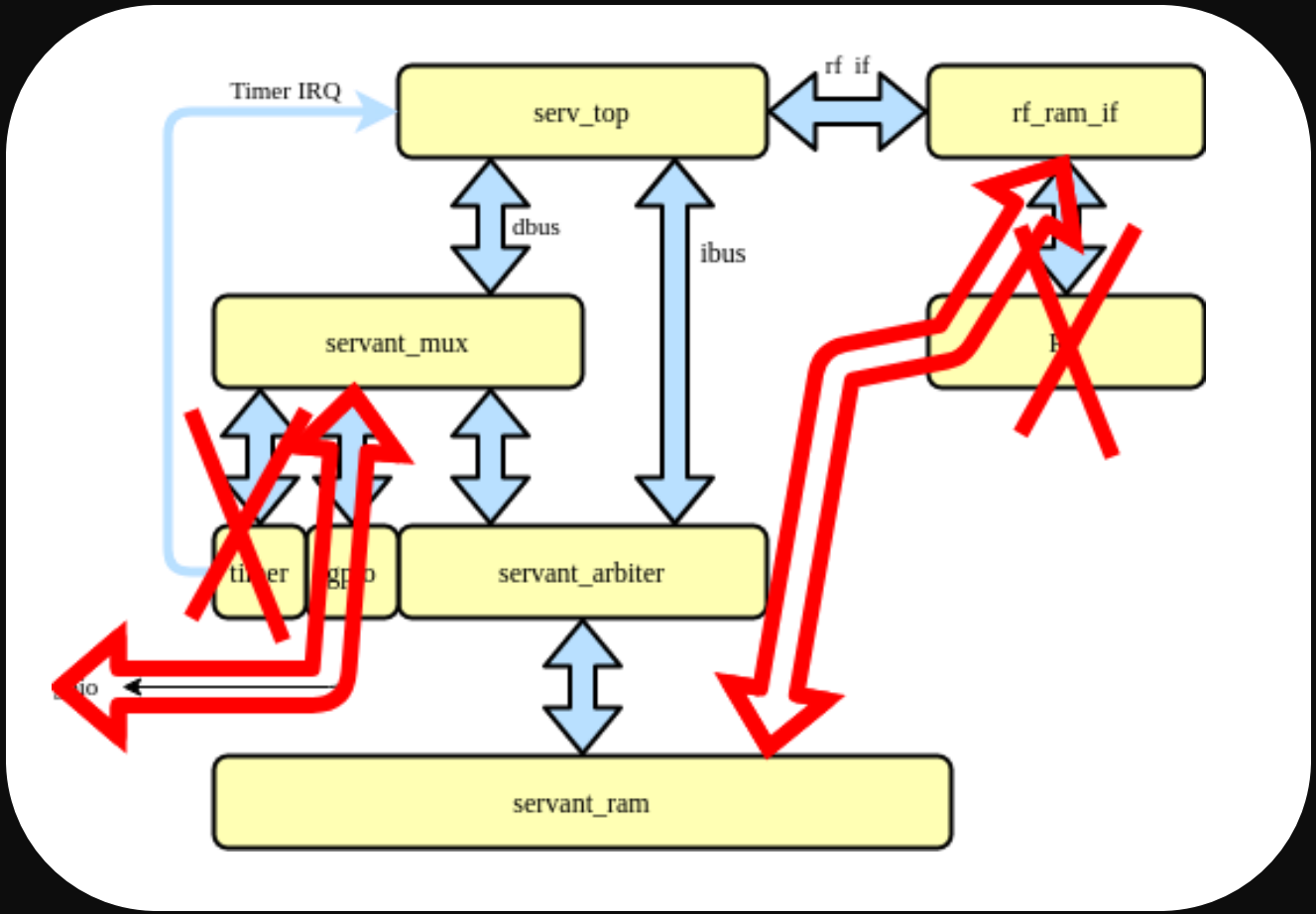
Documentation





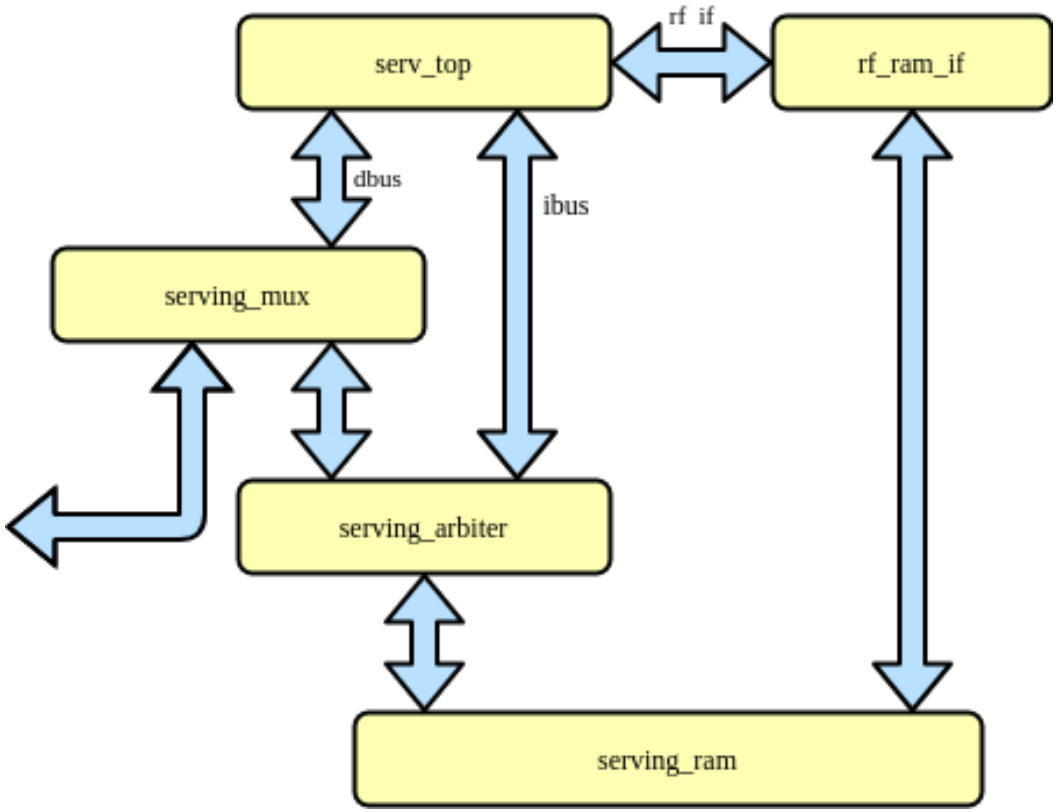
Servant

- Minimal reference platform
- Available for 30+ FPGA boards
- Runs Zephyr OS



Servant

- Minimal reference platform
- Available for 30+ FPGA boards
- Runs Zephyr OS



Serving

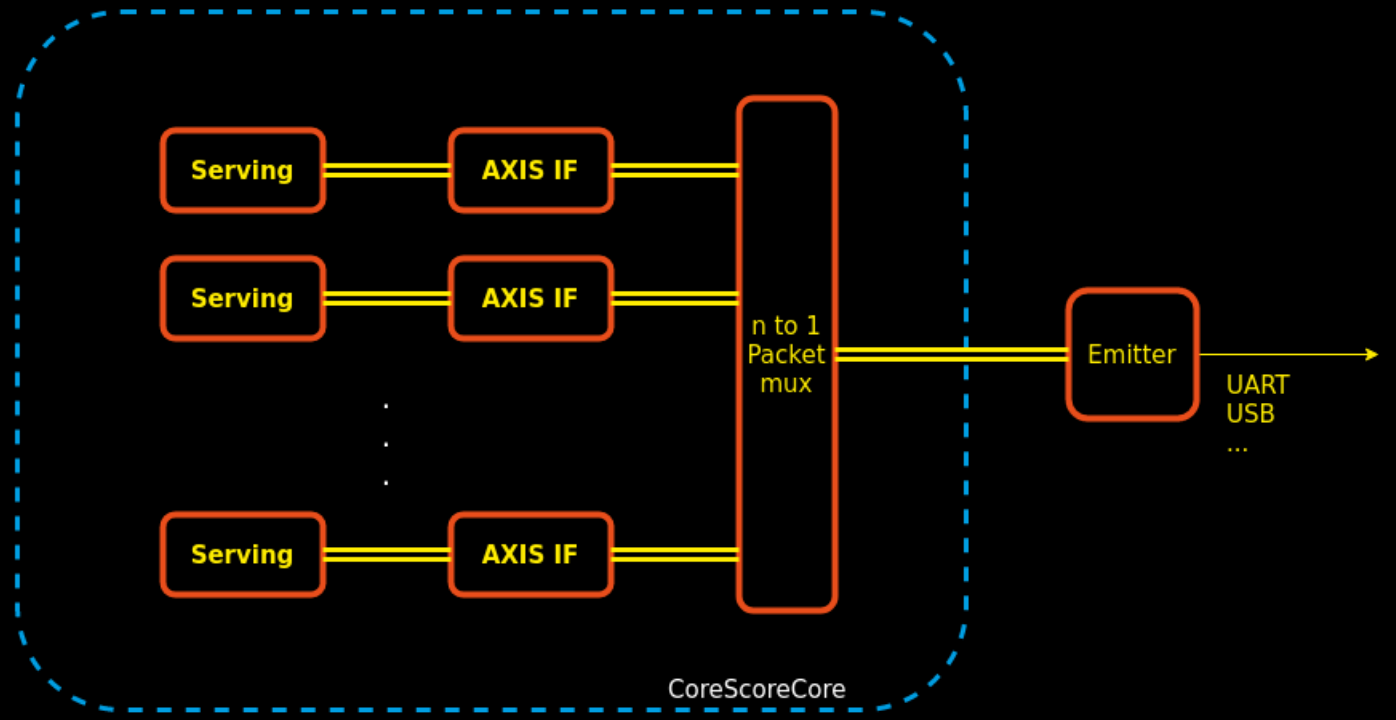
- SoClet
- RF in RAM

Use cases

- Replace complex FSM
  (e.g. DDR init, logging, status...)
- Sensor data aggregation
- Replace 8-bit MCUs
- Replace threaded apps
- Benchmarking

CORE SCORE

an award-giving benchmark for FPGAs
and their synthesis/P&R tools



https://github.com/olofk/corescore

CORE SCORE

an award-giving benchmark for FPGAs
and their synthesis and P&R tools

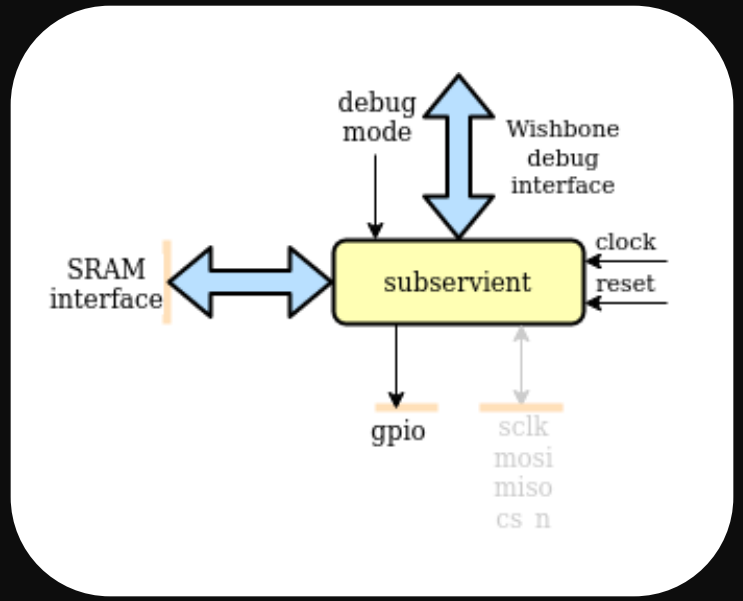
Board	CoreScore
vcu128	6000
intel_s10gx_devkit	5600
vcu118	5087
haps_dx7	3040
intel_a10gx_devkit	2600
de5_net	1568
storeypeak	1152
hpc_k7	1024
hpc_ku	1024
genesys2	967
kc705	960
zcu106	940
polarfireeval	882

Subservient – SERV in OpenMPW

OpenMPW is a Google-financed initiative by Efabless for doing free prototype runs of open source ASICs in the Skywater 130nm process.

Submissions are built using the FOSS Openlane flow and are required to be surrounded by the Caravel harness, including a picorv32 with a Wishbone interface that the user design can connect to.

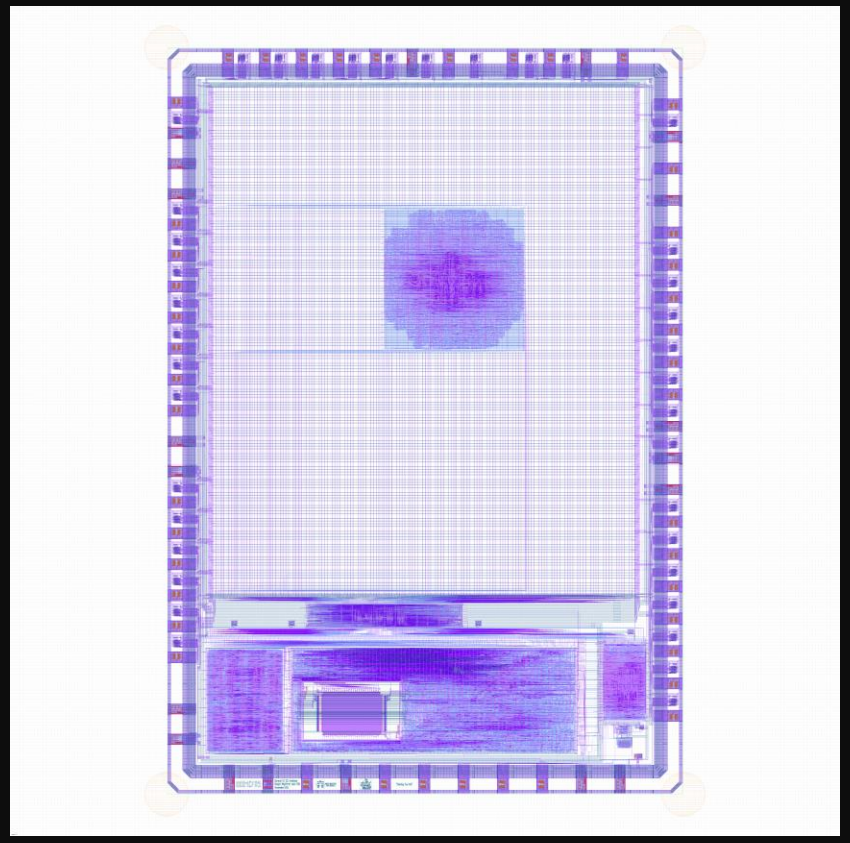
Subservient – SERV in OpenMPW



Subservient is a variant of Serving exposing an SRAM interface and a Wishbone debug interface that can read or write the SRAM when Subservient is kept in debug mode with a separate pin.

We have sent in two slightly different variants, in order to meet timing with later versions of Openlane after the presumed failure of MPW-2

Subservient – SERV in OpenMPW

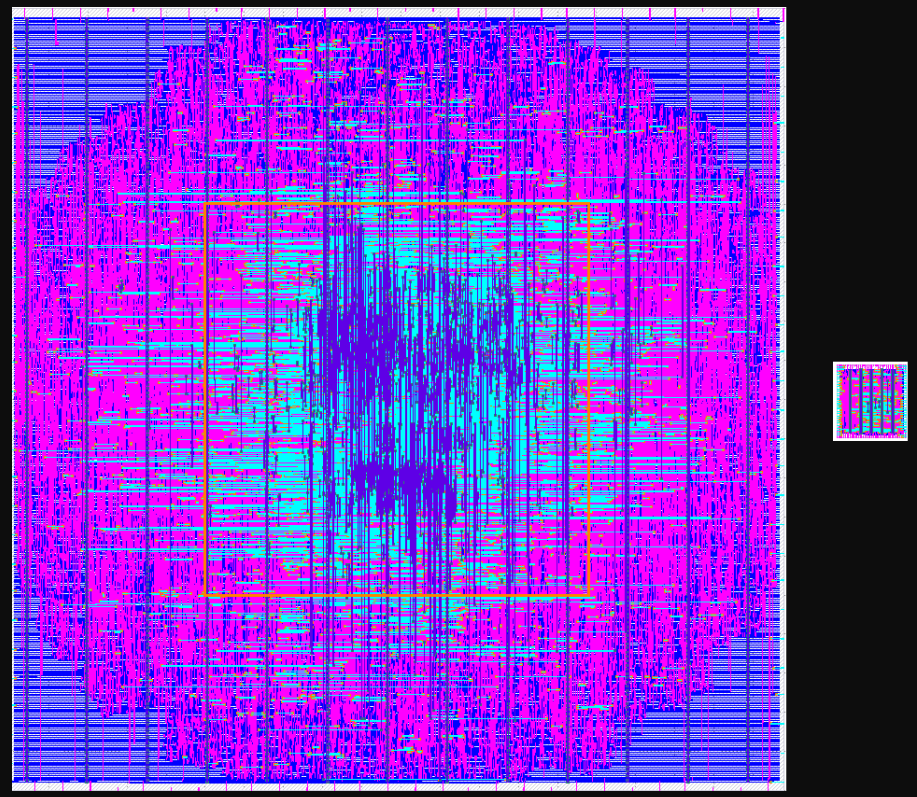


User project area of 2920 by 3520 micrometers

Subservient synthesized together with 512 bytes of DFFRAM about 1000 by 1000 micrometers, 25 MHz clock

Implies a corescore of six or so. Not very good...

Subservient – SERV in OpenMPW

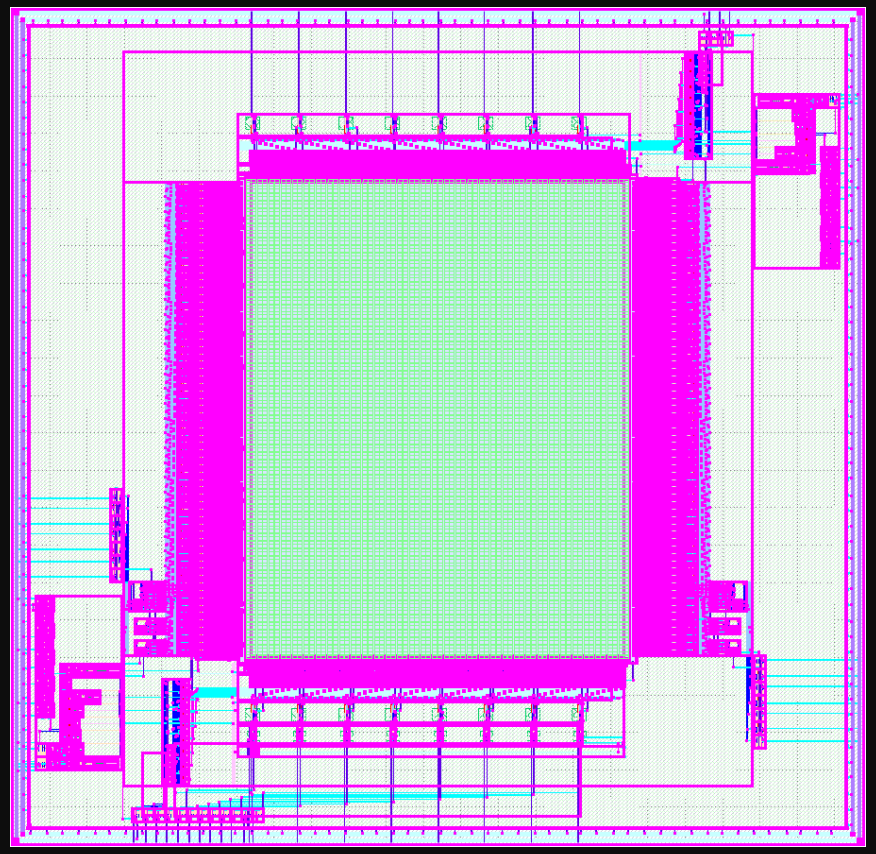


Removing the memory both significantly decreases gate count and eases up the place and route problem, allowing higher density.

Subservient without memory here about 160 by 160 micrometers

If memory was out of the picture completely, we'd have a Corescore of nearly 400!

Subservient – SERV in OpenMPW



Using OpenRAM has been problematic for us

1kbyte, 1 byte wide ready-made macro for sky130A is about 460 micrometers wide

With one such macro for each smaller Subservient, we'd look at about 20 cores in the user project area...

Smaller memories? Shared memories?

Subservient – SERV in OpenMPW

We have not yet received the first batch of chips and can't yet confirm if SERV has been succesfuly ASIC-proven.

Future work

DSRV, QERV

More extensions

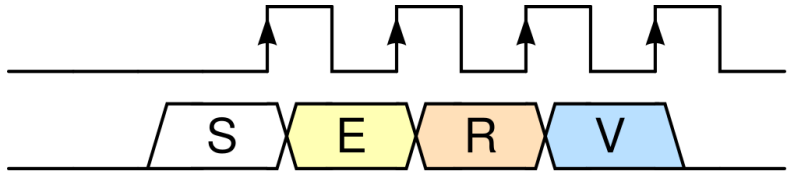
FPGA with hard SERV for each SRAM

Recreate SERV in Logisim, Minecraft, 7400 chips, tubes...

Thank you for your time

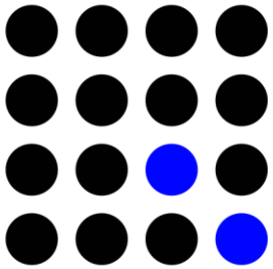
Subservient
https://github.com/klasnordmark/subservient_wrapped

https://corescore.store



https://github.com/olofk/serv

Klas Nordmark
linkedin.com/in/klas-nordmark-33a97244
@knordmark89
https://2550.engineering



Olof Kindgren
linkedin.com/in/olofkindgren
@OlofKindgren
https://qamcom.com