Introduction to free and open source silicon

Olof Kindgren
Qamcom Research & Technology, FOSSi Foundation

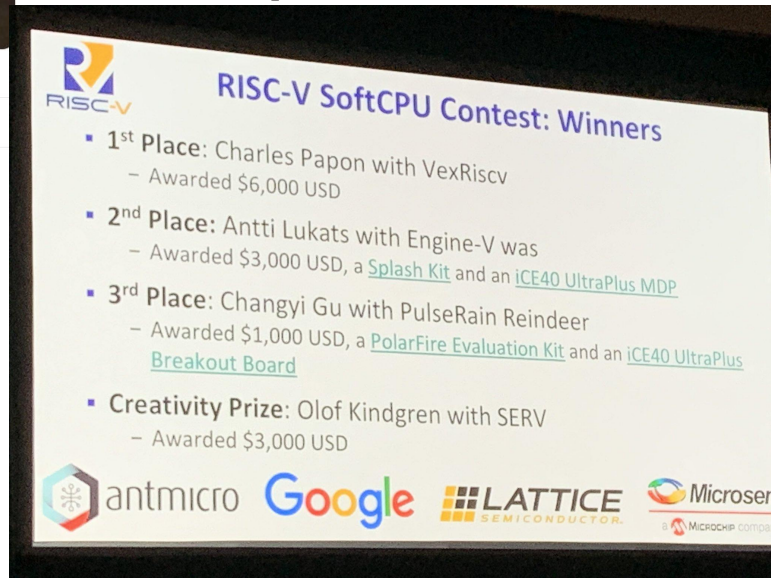
Who am I?

ORConf

Portland Oregon

LATCH-UP

May 4-5 2019



FOSSi Foundation

RISC-V® AMBASSADOR

Award-winning

qamcom

(Free and) open source silicon

is about...

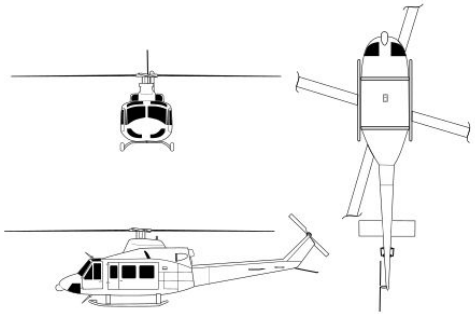
- Programming code to describe chip functionality (commonly called RTL code, written in an HDL)
- Other design sources (netlists, layouts)
- Tools for working with the above (EDA tools)
- Information needed to produce physical chips (e.g. PDK)

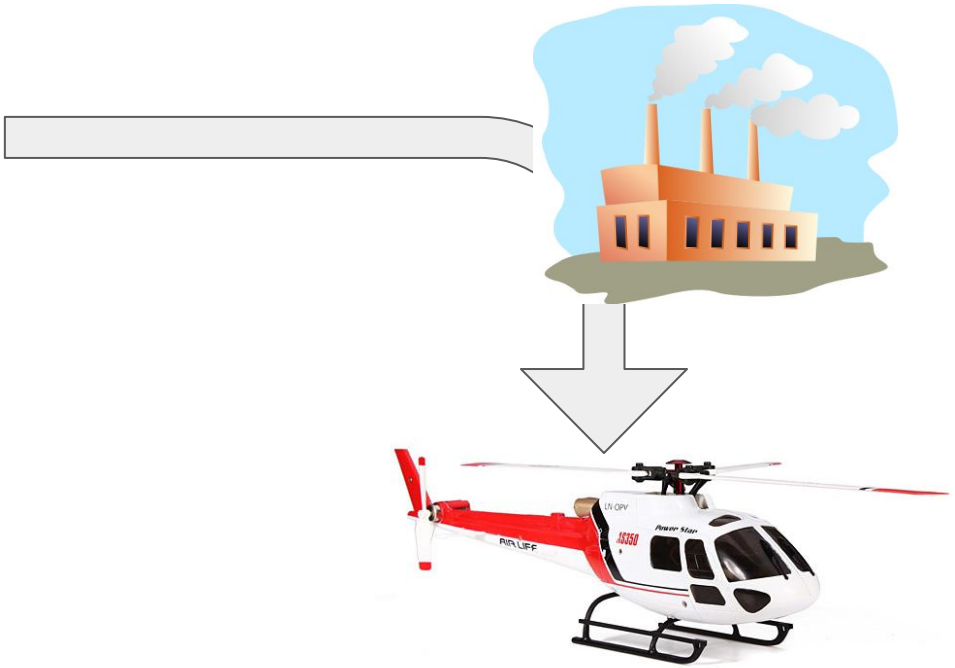
is not…

- open source software (well… actually…)
- open source hardware (well… actually…)

ASIC and FPGA

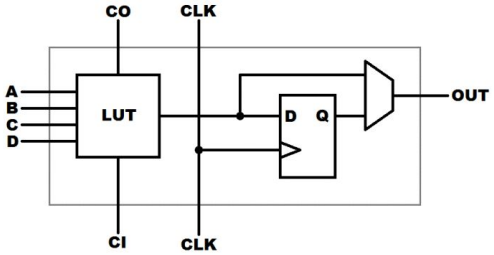
An **A**pplication **S**pecific **I**ntegrated **C**ircuit is a chip created with a specific purpose

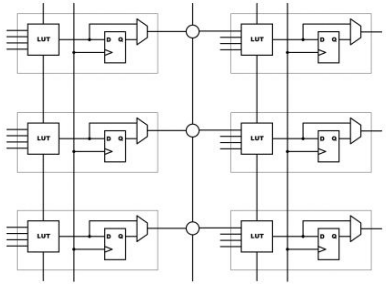


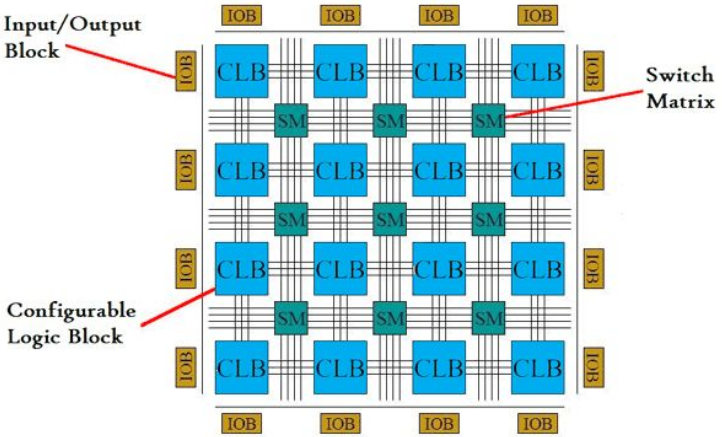


ASIC and FPGA

A **F**ield **P**rogrammable **G**ate **A**rray contains building blocks to create different circuits



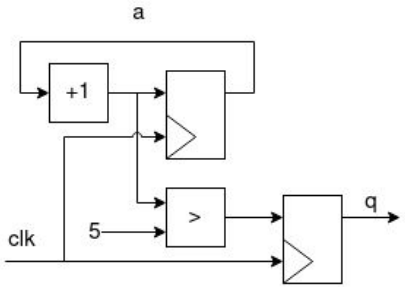




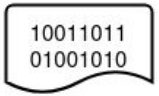
From code to chip

```
module m(
  input wire clk,
  output reg q);

  reg [3:0] a = 4'd0;

  always @(posedge clk) begin
    a <= a + 1;
    q <= (a > 5);
  end
endmodule
```

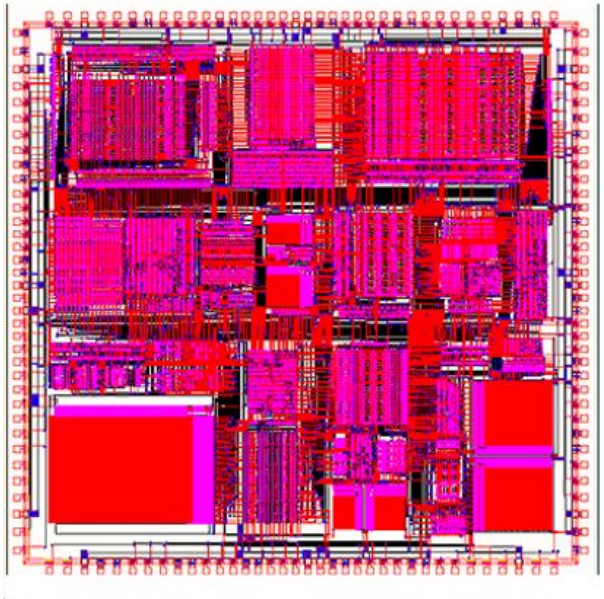
synthesis



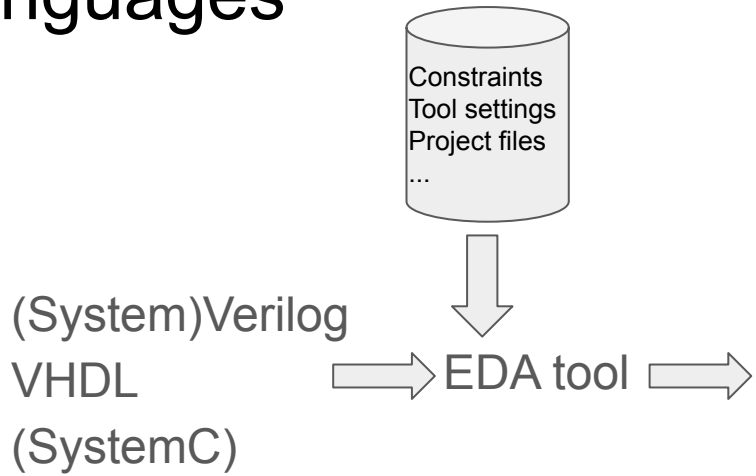
FPGA



ASIC

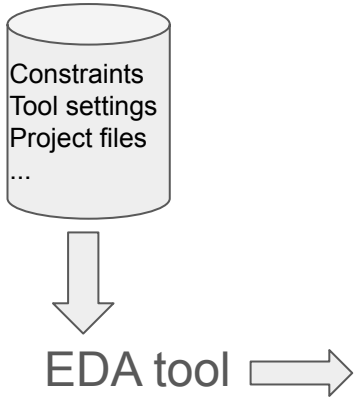


Hardware description languages



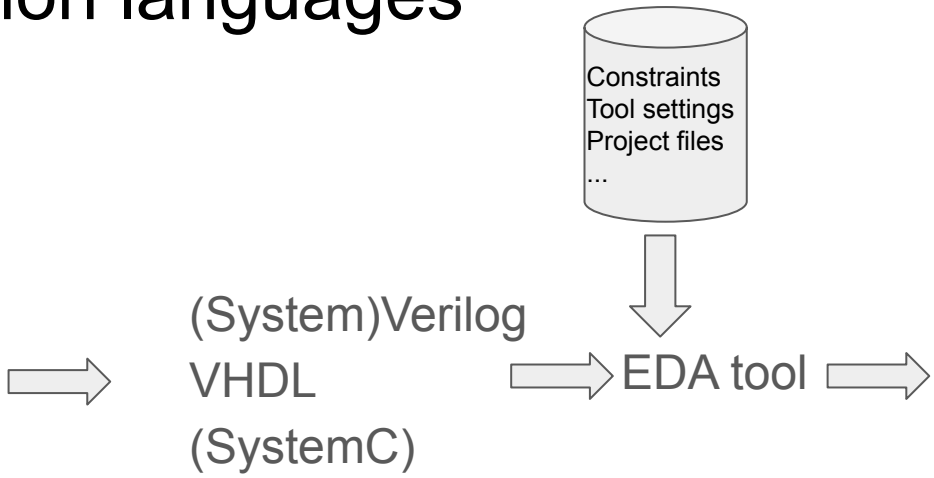
Hardware description languages

Language	Based on
Bluespec	
Chisel	Scala
HardCaml	OCaml
Lava	Haskell
(n)migen	Python
MyHDL	Python
Quokka	C#
PyGears	Python
Silice	
SpinalHDL	Scala
tl-verilog	Verilog+m4
+ many more	



Hardware description languages

Language	Based on
Bluespec	
Chisel	Scala
HardCaml	OCaml
Lava	Haskell
(n)migen	Python
MyHDL	Python
Quokka	C#
PyGears	Python
Silice	
SpinalHDL	Scala
tl-verilog	Verilog+m4
+ many more	

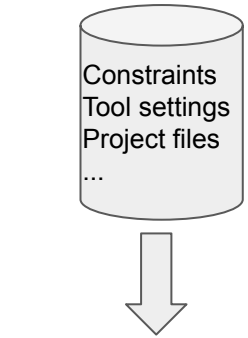


Hardware description languages

Language	Based on
Bluespec	
Chisel	Scala
HardCaml	OCaml
Lava	Haskell
(n)migen	Python
MyHDL	Python
Quokka	C#
PyGears	Python
Silice	
SpinalHDL	Scala
tl-verilog	Verilog+m4
+ many more	

Verilog	
VHDL	

Language	Used by
RTLIL	Yosys
FIRRTL	Chisel
lgraph	
CoreIR	
llhd	
ahir	
...many more	



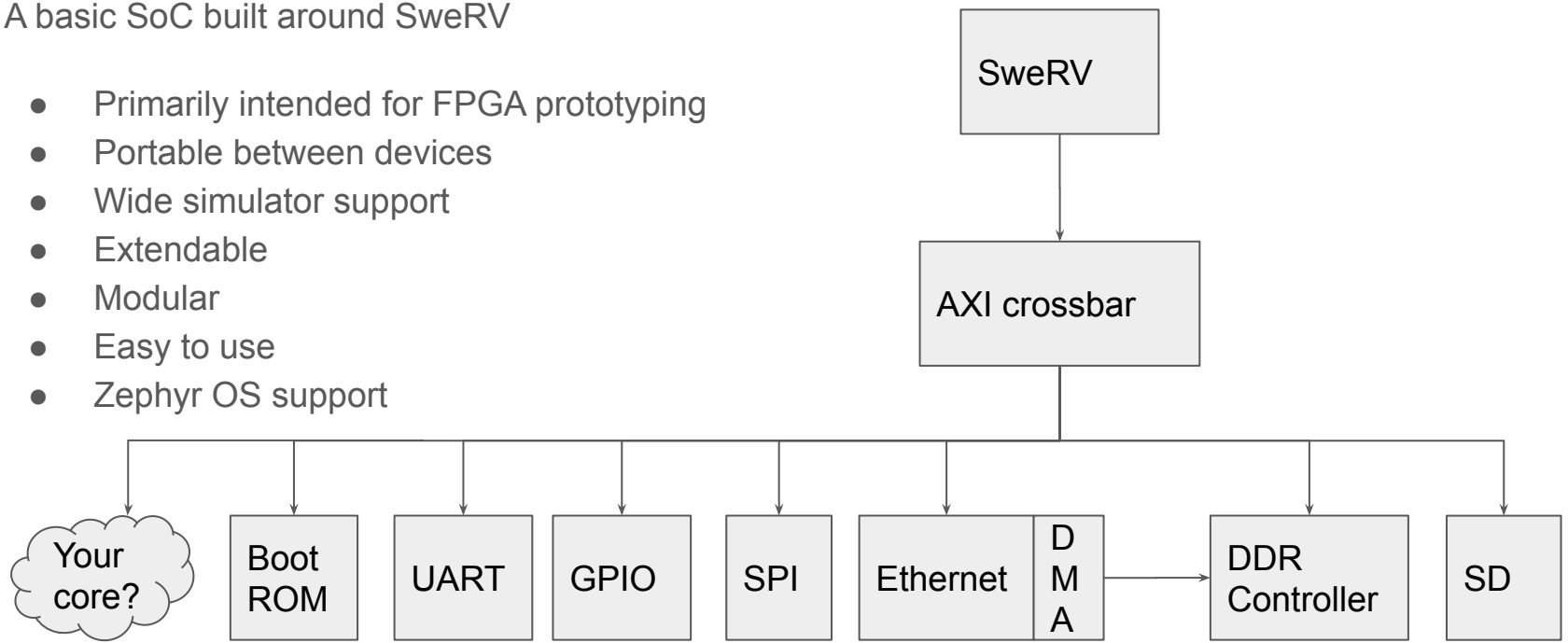
IP cores

- Most digital functionality is created from Verilog (or to a lesser extent VHDL). Analog design is different. Let’s not go there today
- Multiple source files get combined into a larger functional unit called an IP (intellectual property) core. Compare to a library of package in software
- IP cores are combined to create a larger design
- EDA tools are used to create something usable from the source code described in the IP cores.
- Two main types of EDA tool chains: Verification and implementation

SweRVolf

A basic SoC built around SweRV

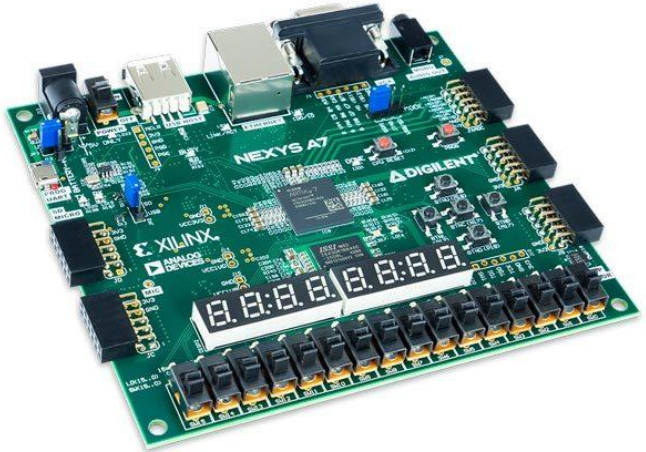
- Primarily intended for FPGA prototyping
- Portable between devices
- Wide simulator support
- Extendable
- Modular
- Easy to use
- Zephyr OS support

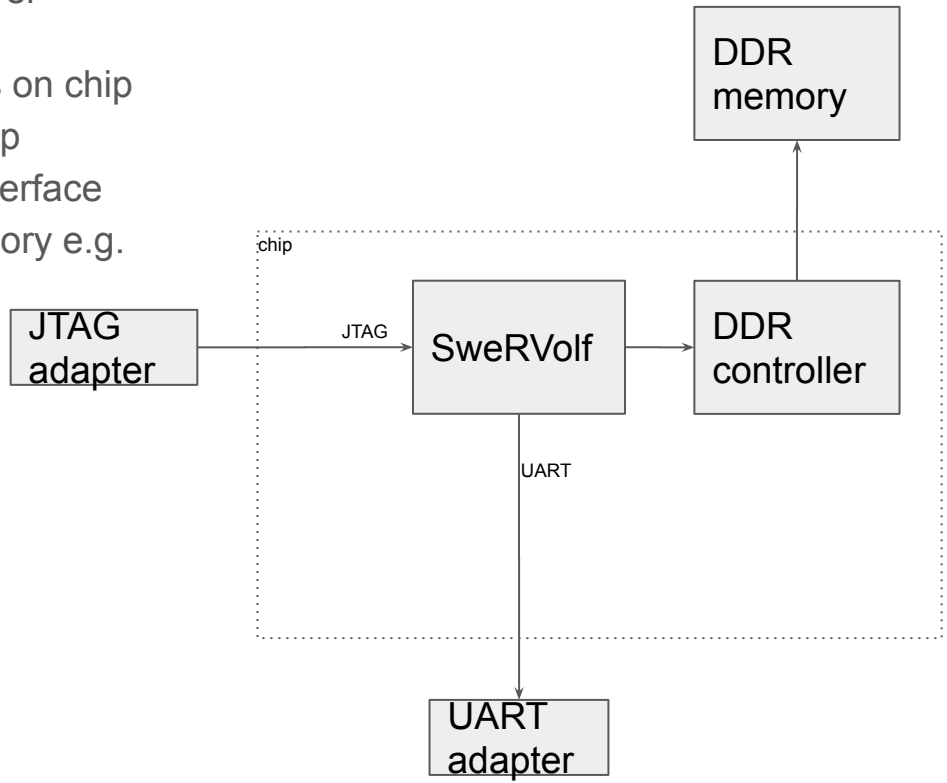


SweRVolf implementation

SweRVolf wrapped in a hardware-specific top-level

- Debug connection goes through JTAG pins on chip
- UART output goes to UART pins on the chip
- Programs can be loaded through debug interface into memory (or stored in non-volatile memory e.g. SPI Flash, not shown here)

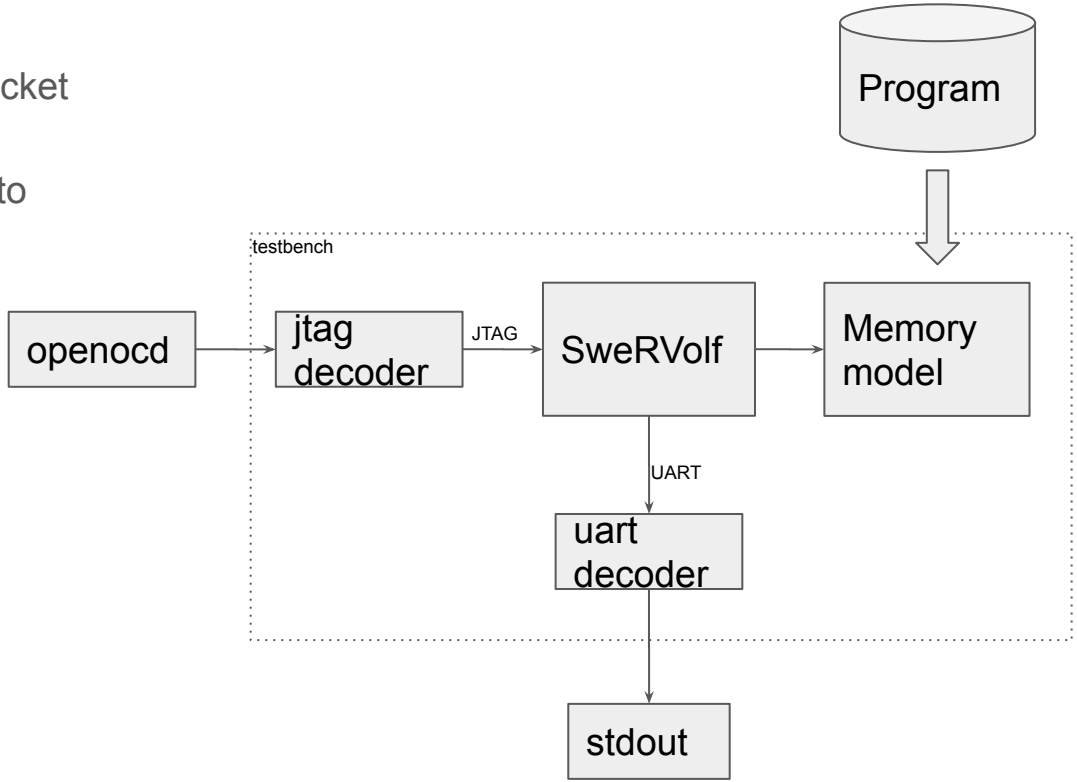




SweRVolf simulation

SweRVolf wrapped in a testbench

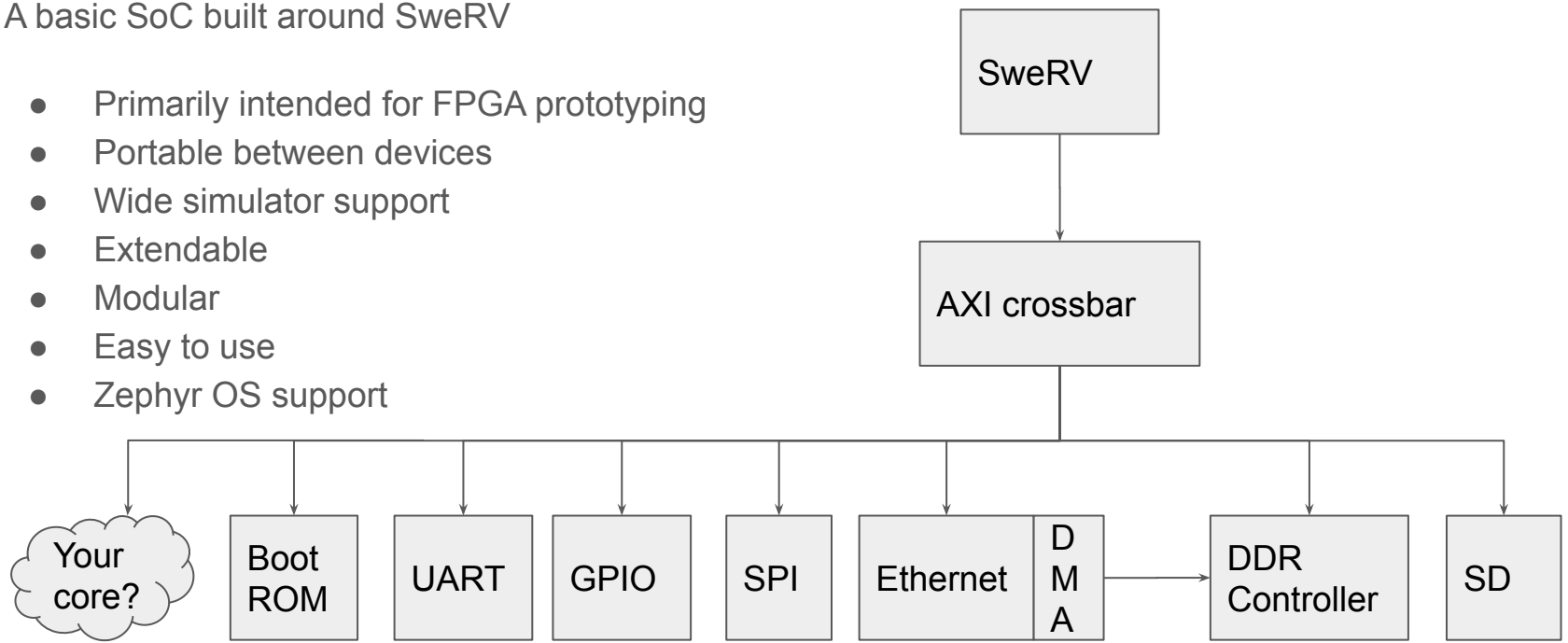
- Debug connection goes through socket
- UART output goes to stdout
- Programs can be loaded directly into memory



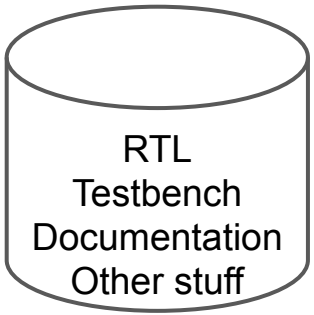
SweRVolf

A basic SoC built around SweRV

- Primarily intended for FPGA prototyping
- Portable between devices
- Wide simulator support
- Extendable
- Modular
- Easy to use
- Zephyr OS support



Core description files



Core description files

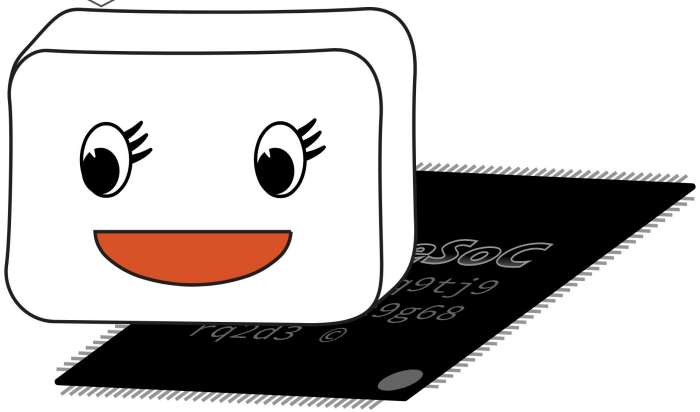
RTL
Testbench
Documentation
Other stuff

Looks like you're making an IP core

Would you like help?

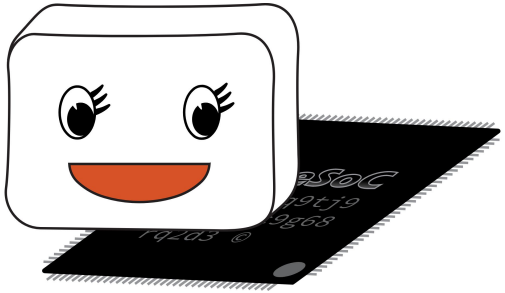
- Get help with making the IP core
- Just make the IP core without help

☐ Don't show me this tip again



Core description files

Core description files describe properties of the core that the EDA tools need (e.g. files, parameters, tool options)

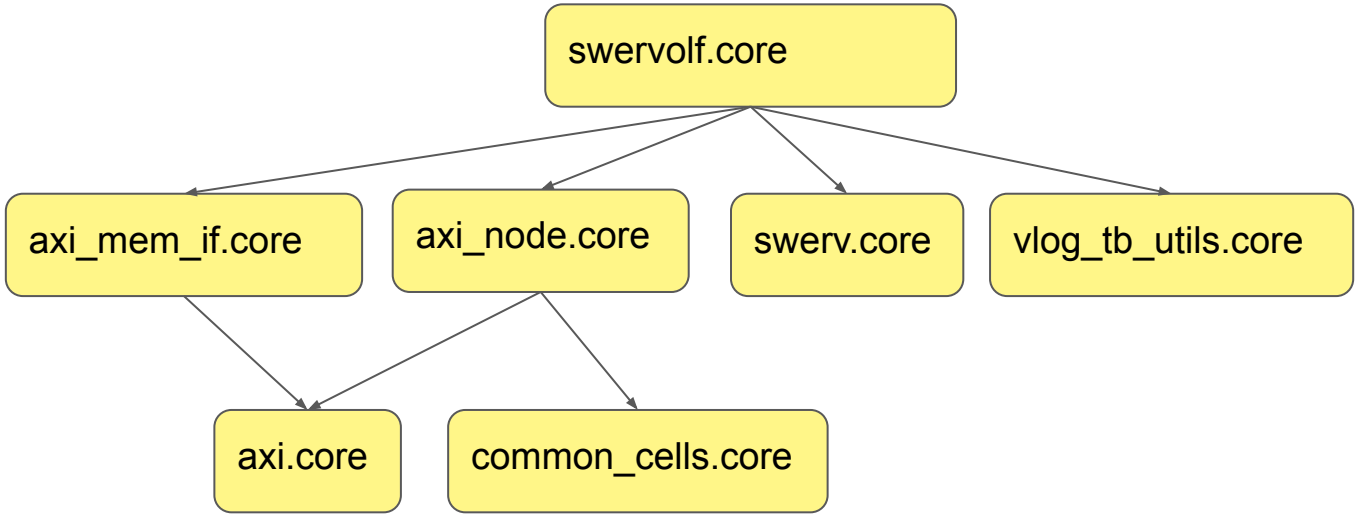


swerv.core

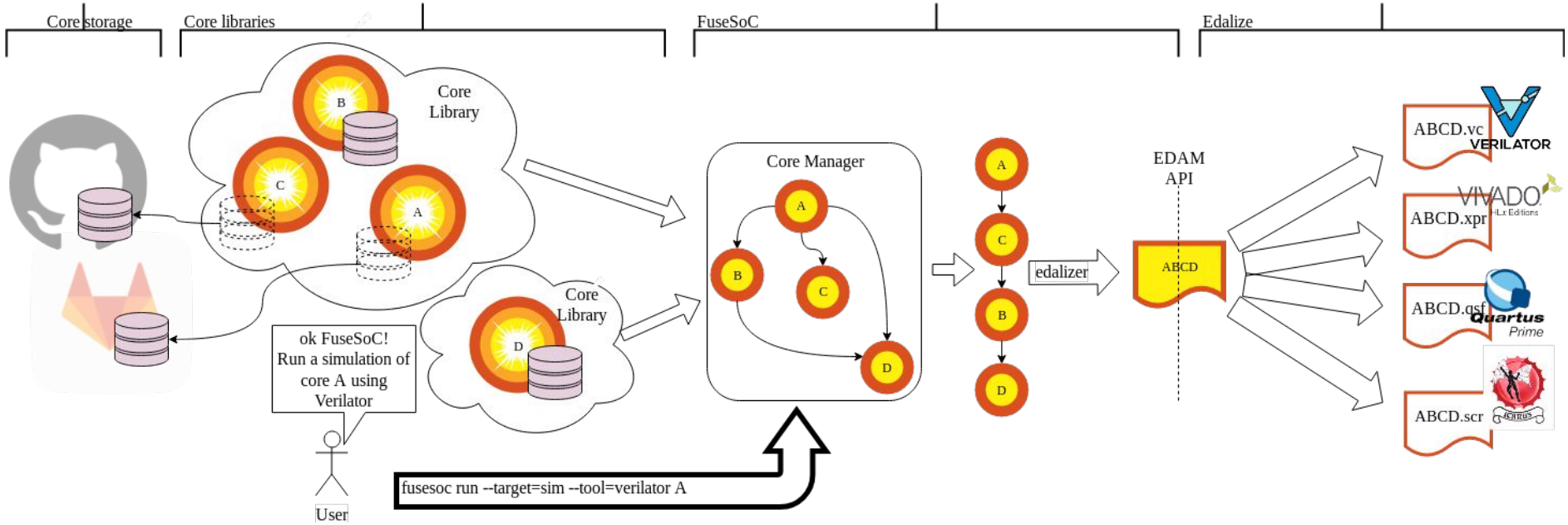
RTL
Testbench
Documentation
Other stuff

```
CAPI=2:

name : chipsalliance:cores:swerv_eh1:1.8

filesets:
  rtl:
    files: [rtl/swerv_top.v, rtl/otherfile.v]
    file_type : verilogSource
  tb:
    files:
      - tb/swerv_wrapper.v : {file_type : verilogSource}
      - tb/swerv_tb.h : {is_include_file : true}
      - tb/swerv_tb.cpp
    file_type : cppSource
```

SweRVolf dependencies



What is FuseSoc?



FuseSoC is a package manager…

...and a build tool for HDL

Existing libraries

fusesoc-cores	77
orpsoc-cores	100
openpiton	66
opentitan	129
optimsoc	102

+ tons of smaller libraries (e.g. picorv32, serv, microwatt, SweRV)

- An ISA, just like ARM, x86, PPC, MIPS
- A modern RISC ISA with open specifications
- Design started 2010
- Governed by RISC-V International with 700+ member companies and institutions
- Highly active development in all areas
- Expected to become a default choice for many areas in 5-10 years

RISC-V details

- Small core instruction set called RV32I (40 instructions) to allow very small implementations
- Optional features are implemented as standard extensions e.g.
  - M - Multiply/Divide
  - A - Atomic operations
  - F - Floating point
  - G - General purpose (combination of extensions I, M, A, F, D, Zicsr, Zifencei)
- ...or custom extension
  - Xpulp
  - + other open and proprietary
- 32- and 64-bit already in products.

Available RISC-V cores and products

- RISC-V exchange lists available cores, SoC, boards and software for those who want to get started with RISC-V
- https://riscv.org/exchange/

Capabilities

RV32I (with an option to include enough of privilege spec to run compliance tests and Zephyr OS)

Formally verified with RISCV-Formal

BSD-licensed

Available at https://github.com/olofk/serv

Wishbone interface

Bit-serial

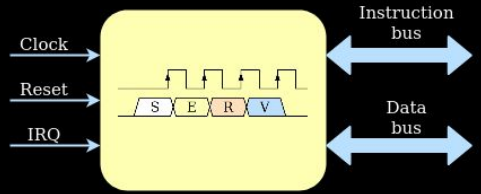
World's smallest RISC-V CPU

Hardware support

Ready-made FuseSoC targets currently exists for

1BitSquared iCEBreaker
Alhambra II
Digilent Arty A7
Digilent Nexys A7
GnarlyGrey Upduino2
RadionaOrg ulx3s
TinyFPGA BX
Trenz Electronics cyc1000
Xilinx ZCU106

More to come



Servant SoC

Reference platform for SERV

Contains memory, GPIO and timer

Runs Zephyr OS

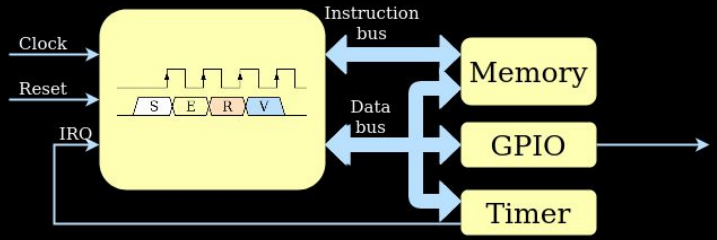
Easy to port to new targets

Hardware support

Ready-made FuseSoC targets currently exists for

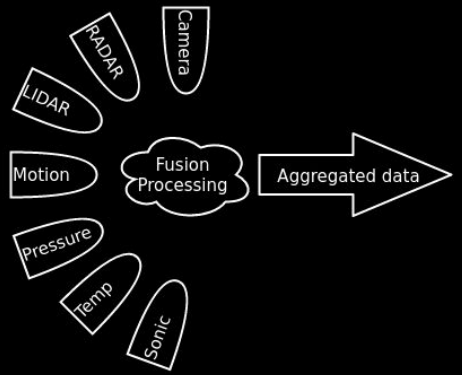
1BitSquared iCEBreaker
Alhambra II
Digilent Arty A7
Digilent Nexys A7
GnarlyGrey Upduino2
RadionaOrg ulx3s
TinyFPGA BX
Trenz Electronics cyc1000
Xilinx ZCU106

More to come



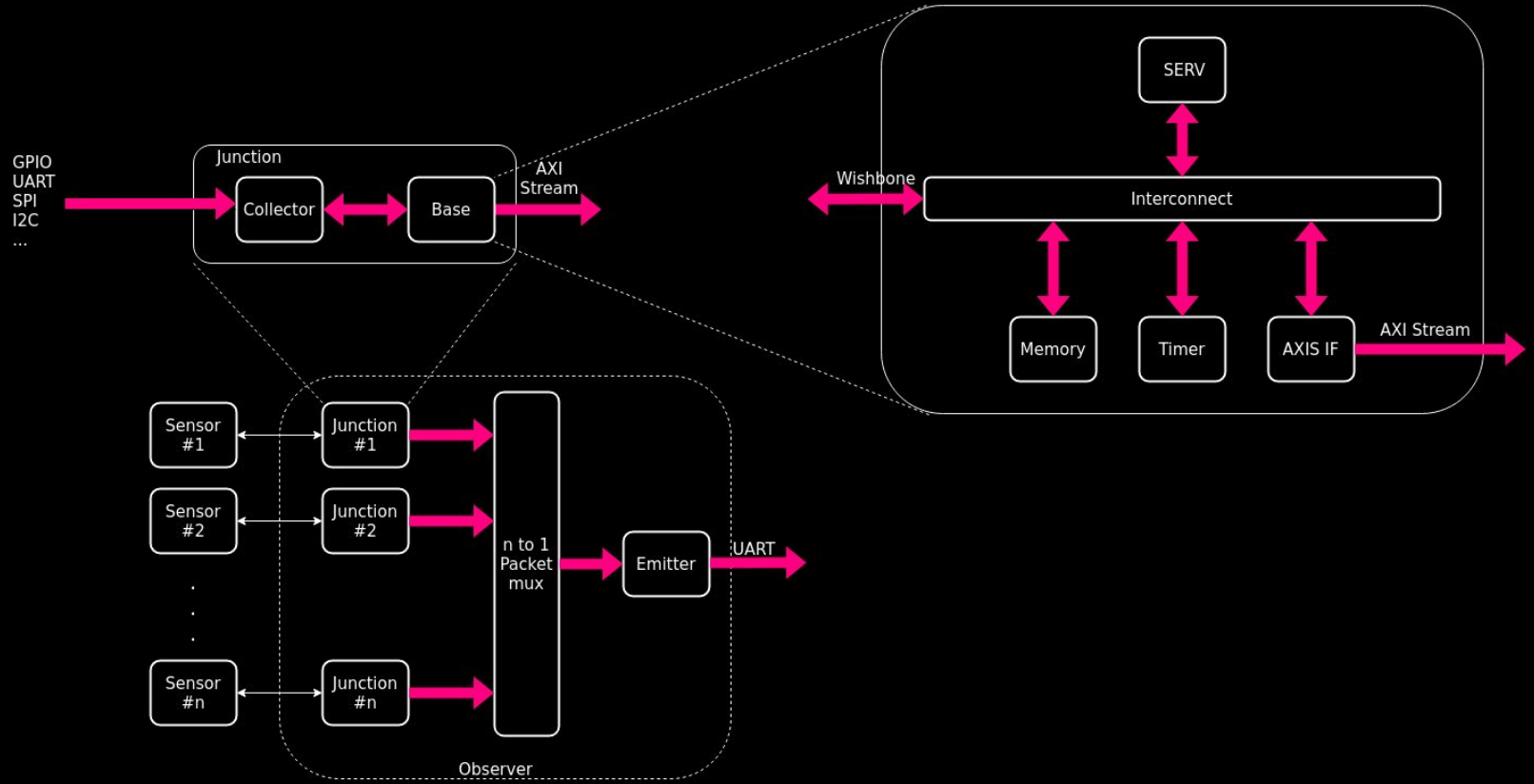
Observer

Observer is a configurable and software-programmable sensor aggregation platform for heterogenous sensors



Wishlist for a heterogenous sensor platform

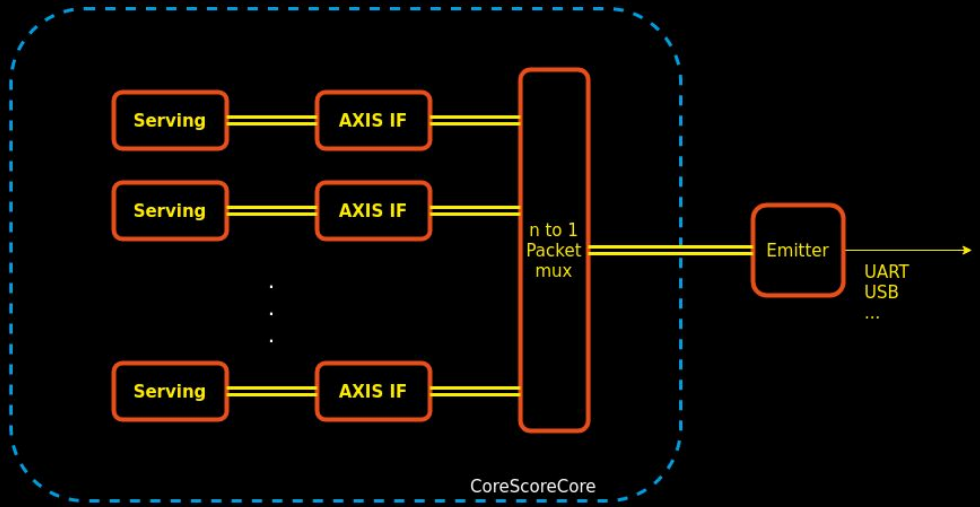
Parallel data collection	FPGA
Flexible I/O	FPGA
Custom data processing	FPGA /CPU
Easy programmability	CPU
Custom control logic	CPU
Status generation	CPU



https://github.com/olofk/observer

CORE SCORE

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



https://github.com/olofk/corescore

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

Board	CoreScore	SERV/$
CYC1000	60	1.54
Nexys A7	268	1.01
Upduino2	11	0.57
TinyFPGA BX	16	0.42
ZCU106	940	0.34

https://github.com/olofk/corescore

What about FPGA?

- Traditionally very closed
  - Most FPGA vendors supply crippled versions of their toolchains + simulators free of charge which still require acquiring a no-cost license, set up license servers etc.
- ...but
  - For a growing subset of FPGAs, open source toolchains are being worked on
  - This year, the first FPGA vendor (QuickLogic) released a fully open source toolchain with their devices
  - Xilinx released most of their simulation models under a proper open source license
  - IEEE agreed to move VHDL standard libraries to an open source license

What about ASIC?

- Building an ASIC is out of reach for most people because
  - Tools are extremely expensive
  - Manufacturing costs are high
  - Required IP cores are expensive
  - Fabs don’t give you access to their knowledge unless there are NDAs in place

What about ASIC?

- Building an ASIC is suddenly possible for most people because
  - We have a free ASIC tool chain (OpenLane, a subset of OpenRoad)
  - Google sponsors four production runs in 2020-2021
  - All required IP cores, including SRAM (OpenRAM) is available as open source
  - The SkyWater foundry has released their PDK (Process Development Kit) as open source

Learn more about this from the FOSSi Foundation Dial-Up video series

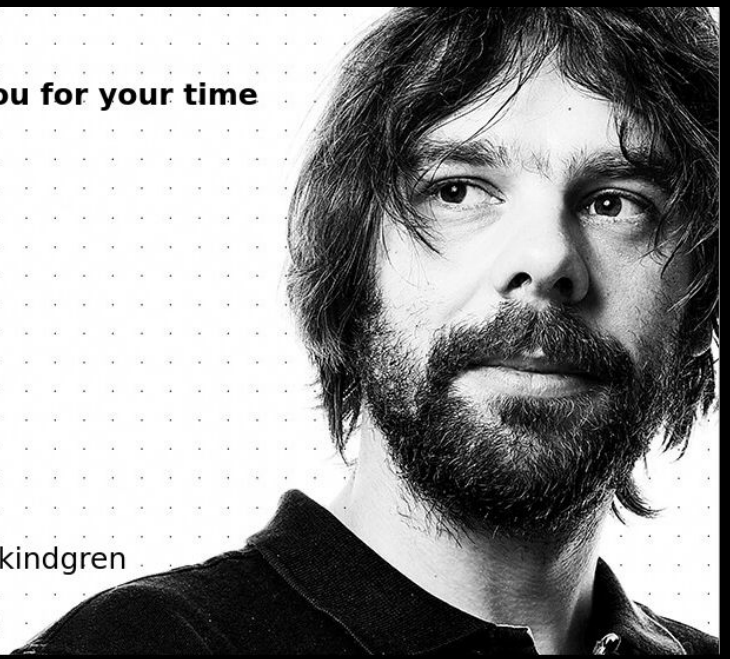
https://fossi-foundation.org/dial-up/

SERV

Observer

CORE SCORE

Thank you for your time



https://github.com/olofk/serv
https://github.com/olofk/observer
https://github.com/olofk/corescore

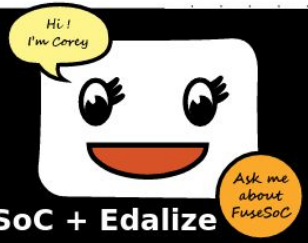
https://www.linkedin.com/in/olofkindgren

https://twitter.com/OlofKindgren

qamcom

RISC-V® AMBASSADOR

https://riscv.org/risc-v-ambassadors
https://www.qamcom.com



FuseSoC + Edalize

https://github.com/olofk/fusesoc
https://github.com/olofk/edalize

FOSSi Foundation

CHIPS ALLIANCE

https://fossi-foundation.org
https://chipsalliance.org