ngspice circuit simulator - stand-alone and embedded into KiCad

Holger Vogt
Duisburg, Germany

Contents

Why Circuit Simulation?

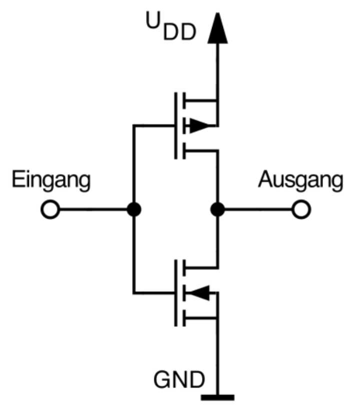
Emulate electronic circuits per software,
cost efficient and time saving:

- Check functionality without hardware
- Assess influence of parasitic elements
- Evaluate circuit variants easily
- Evaluate new concepts without too large effort
- Cross check against automatic circuit generators
- Anticipate reliability (degradation, radiation damage, attacks)
- Learning experience (e.g. peer into the interior of a circuit)

ngspice – what is it ?

Circuit simulator that numerically solves equations describing (electronic) circuits made of passive and active devices for (time varying) currents and voltages.

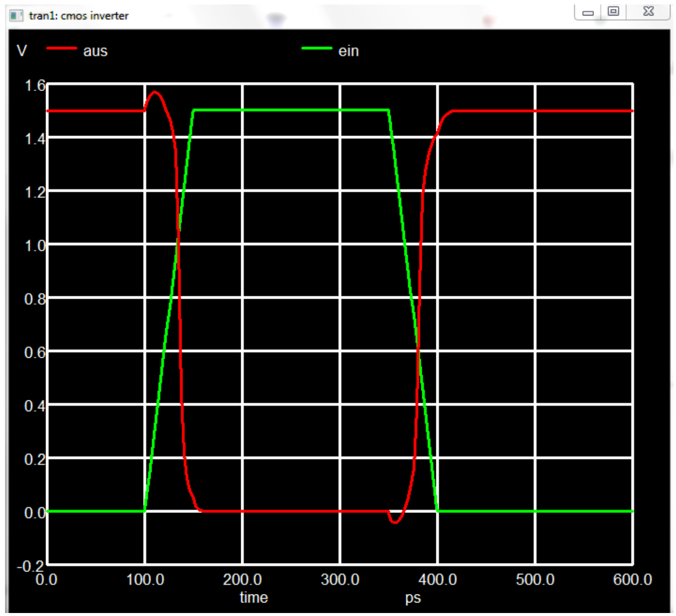
Open source successor of venerable spice3f5 from Berkeley



the circuit

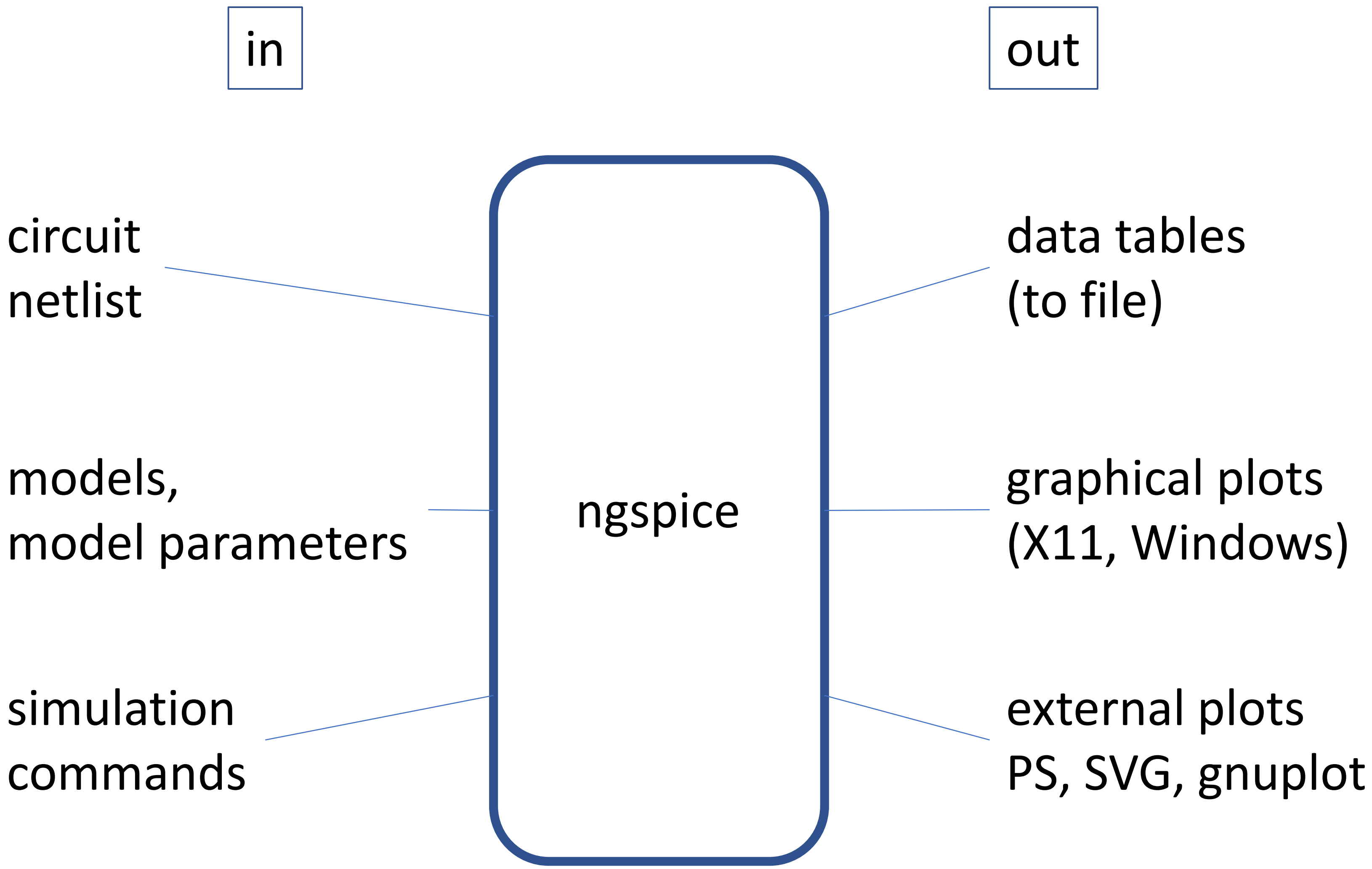
```
CMOS inverter

.include ./bsim4soi/nmos4p0.mod
.include ./bsim4soi/pmos4p0.mod
.option TEMP=27C

Vpower VD 0 1.5
Vgnd VS 0 0

Vgate Ein VS PULSE(0 1.5 100p 50p 50p 200p 500p)

MN0 Aus Ein VS VS N1 W=10u L=0.18u
MP0 Aus Ein VD VS P1 W=20u L=0.18u

.tran 3p 600ps

.control
  run
  plot Ein Aus
.endc

.END
```

the input



the output

ngspice user interface



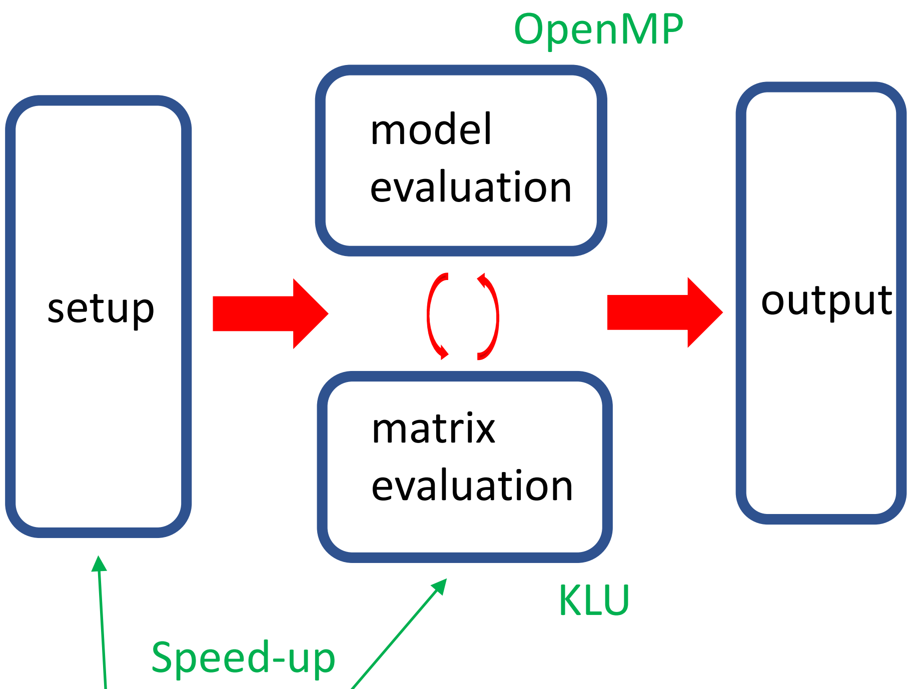
What’s new in ngspice?

- Current release ngspice-42 from Dec. 27th, 2023
- KLU matrix solver (in addition to the venerable Sparse 1.3)
- Support for Verilog-A compact device models
- Co-simulation ngspice mixed-signal – Verilog digital
- Co-simulation ngspice mixed-signal – C-coded digital
- Vastly improved GUI interface KiCad - ngspice

https://sourceforge.net/projects/ngspice/files/ng-spice-rework/42/ https://ngspice.sourceforge.io/

KLU matrix solver

Inside ngspice



Sparse 1.3 solver (K. Kundert) since 1986

KLU matrix solver (T.A. Davis) since ngspice-42

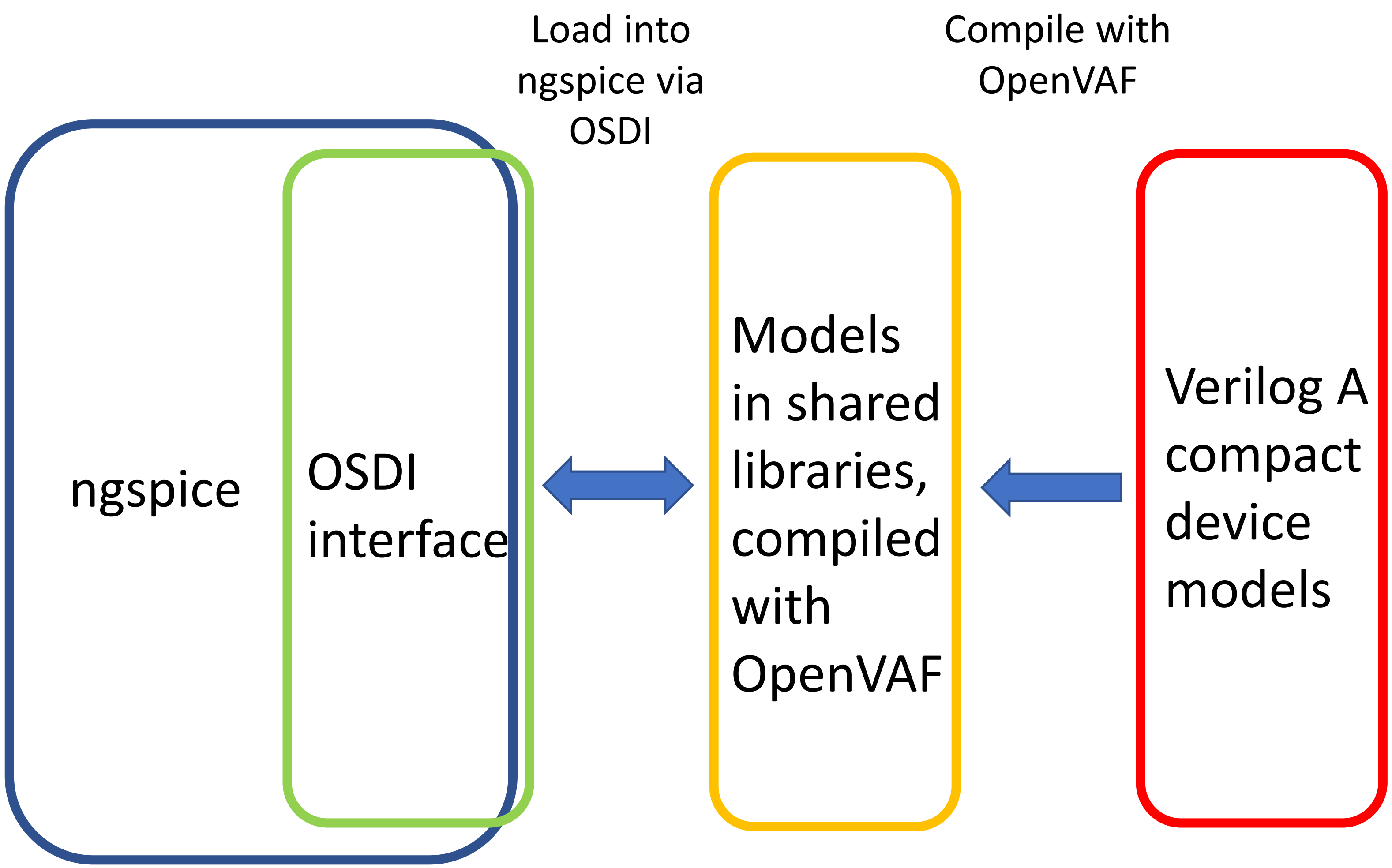
With KLU simulation speed-up by 1.5 to 3 for large circuits

https://sparse.sourceforge.net/description.html https://github.com/DrTimothyAldenDavis/SuiteSparse

Verilog-A compact device models in ngspice

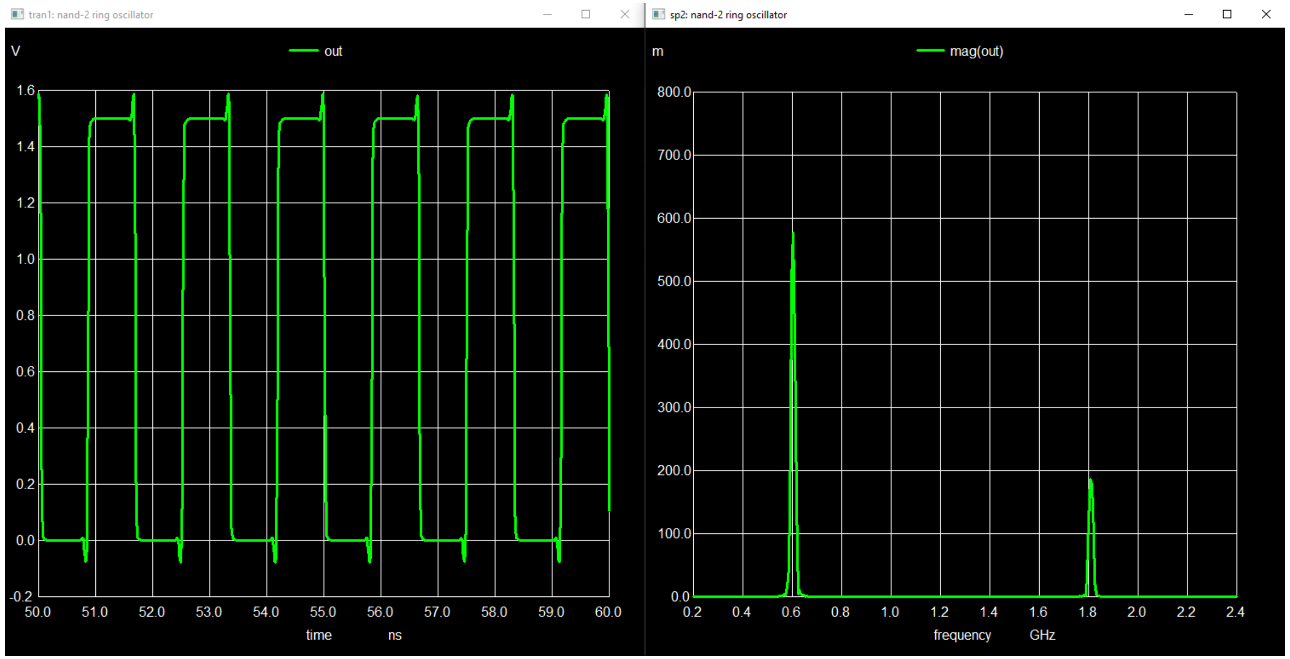
- Compact model development today only with Verilog-A
- Provide simulation access to modern devices
- Examples are:
  BSIMBULK: short channel MOS, BSIMCMG: FinFet,
  ASM-HEMT: GaN devices, HiSIM-HV: Power devices,
  HICUM: 500 GHz HBT bipolar transistors …

Setup for Verilog-A compact device models



For models see https://github.com/dwarning/VA-Models
for OpenVAF see https://openvaf.semimod.de/

Verilog-A use case
Analog simulation with IHP Open Source PDK



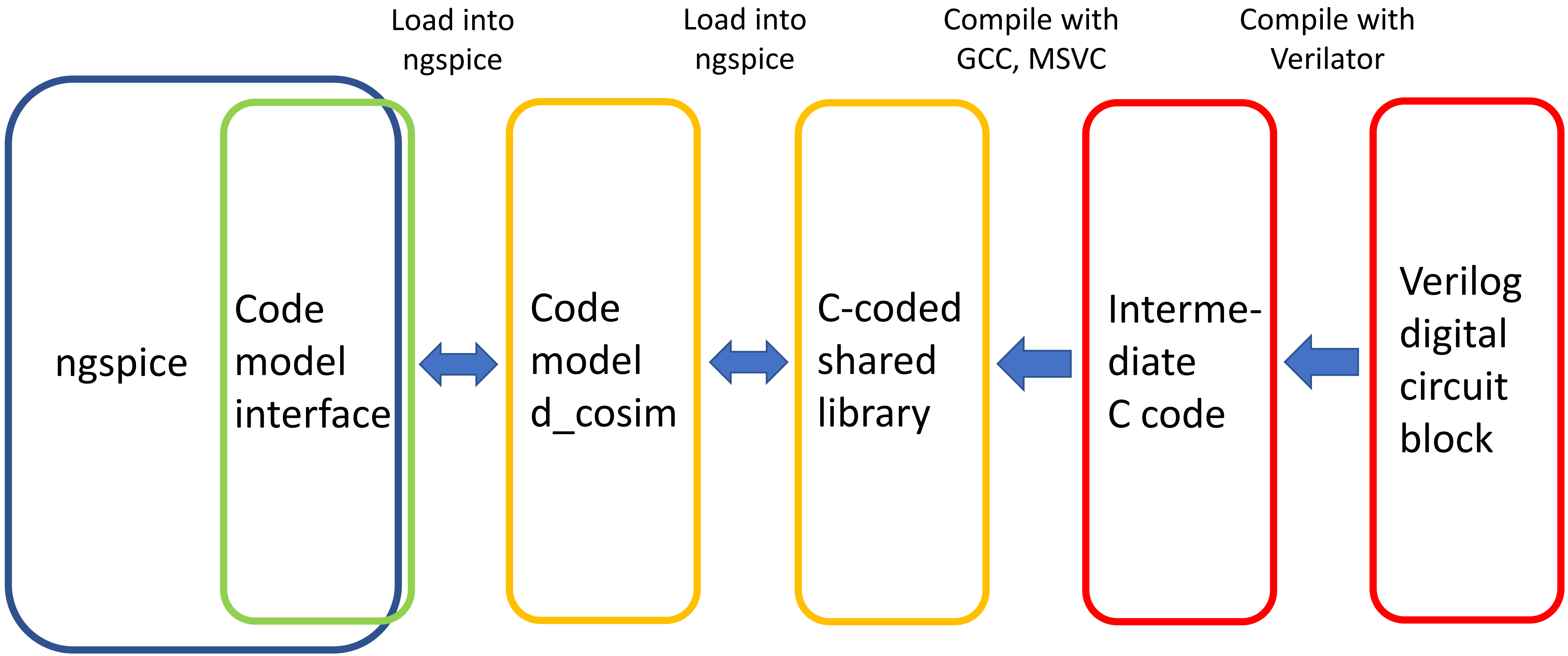
19-stage NAND-gate ring oscillator, 600 MHz,
Inverter delay 280 ps

IHP SG13, 130 nm SiGe BiCMOS

Models used:
Bipolar: VBIC (ngspice intrinsic)
MOS: PSP103.6 NQS (Verilog-A)

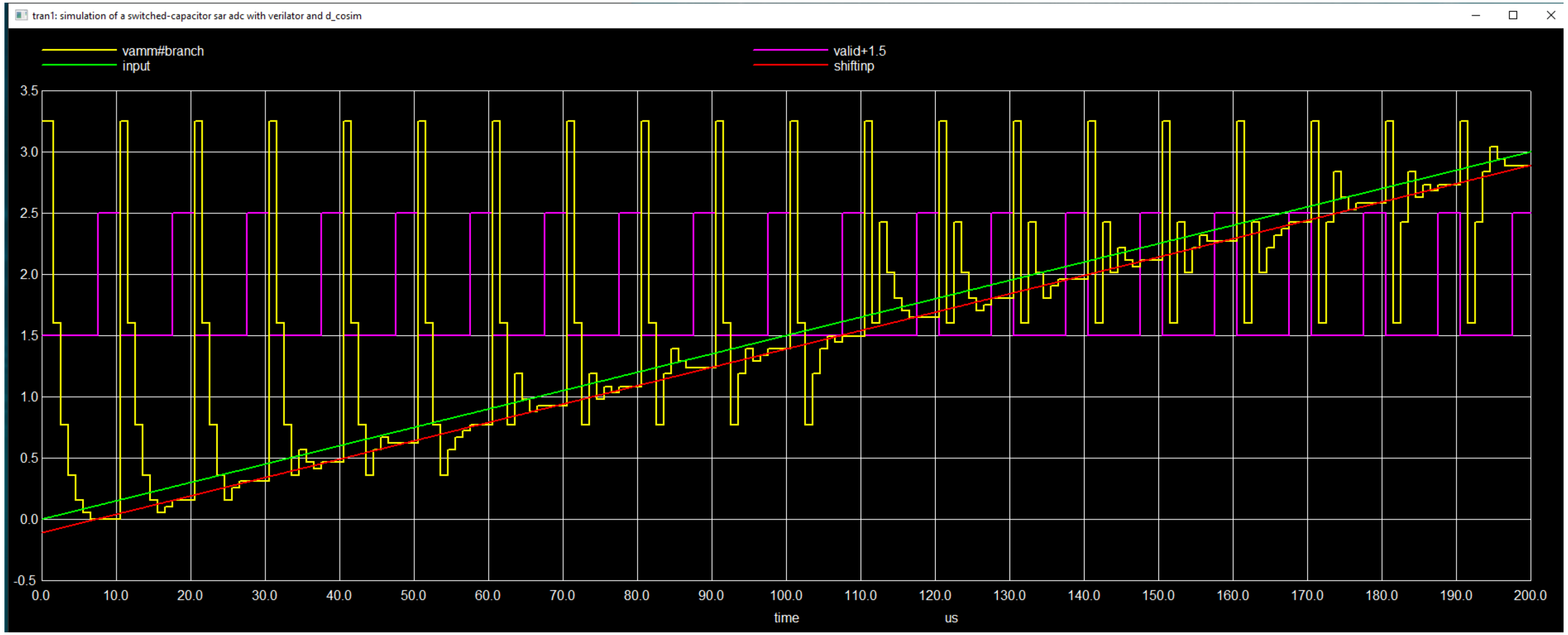
For PDK see https://www.ihp-microelectronics.com/services/research-and-prototyping-service/fast-design-enablement/open-source-pdk
For models see https://github.com/dwarning/VA-Models

Mixed-signal simulation with digital Verilog circuit block and ngspice analog and digital event based circuit netlist



For details see https://ngspice.sourceforge.io/docs/others/Verilog-CoSim.pdf
for Verilator see https://www.veripool.org/verilator/

Verilog digital, ngspice mixed signal use case
Successive approximation register (SAR) analog-digital converter demo



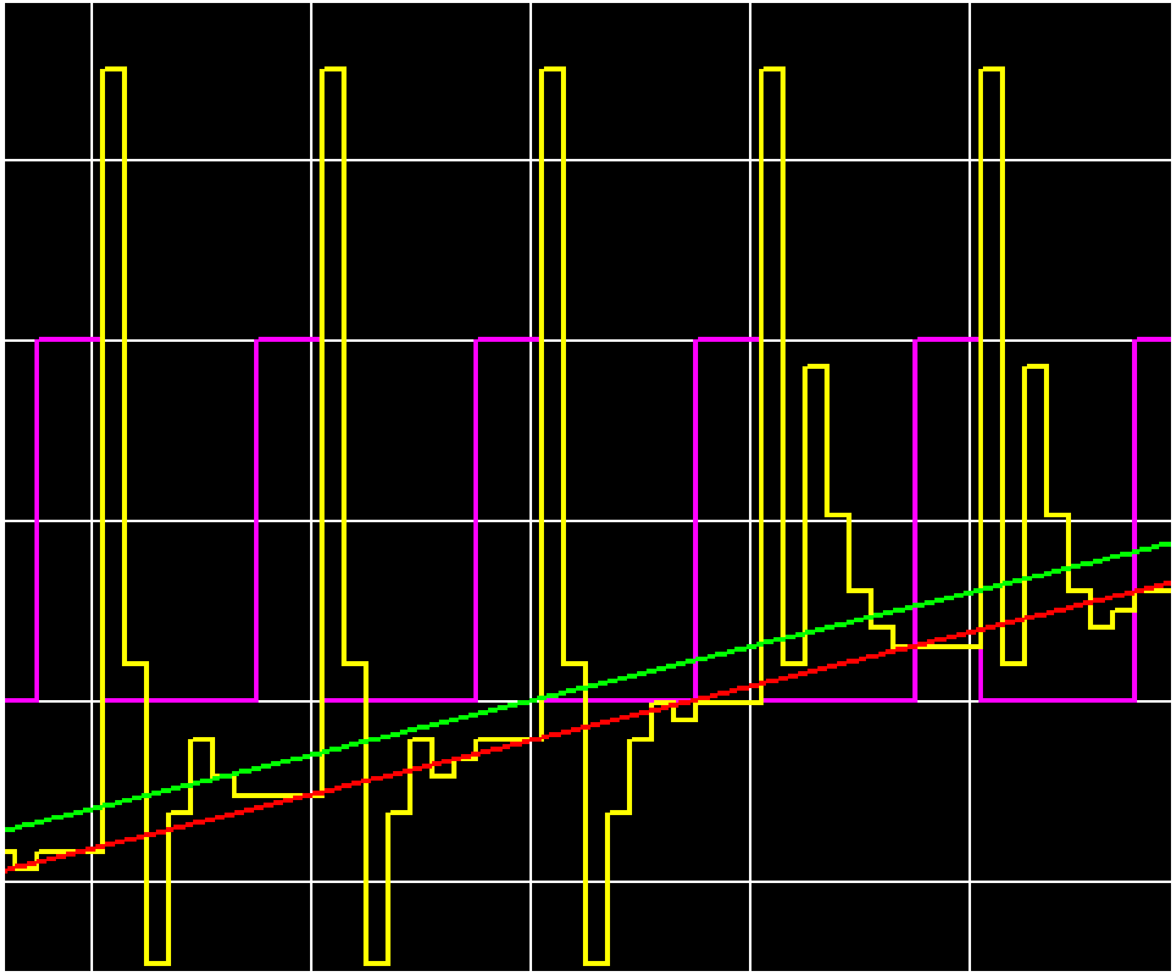
6 bit SAR ADC demo

Analog:
Binary weighted capacitor array, switches, comparator

Digital (Verilog):
SAR control

- Intermediate files and scripts provided with ngspice distribution
- Only two commands required: **ngspice vlnggen adc.v** compile with Verilator, gcc — **ngspice adc.cir** run simulation

SAR output, enlarged



6 bit SAR ADC demo

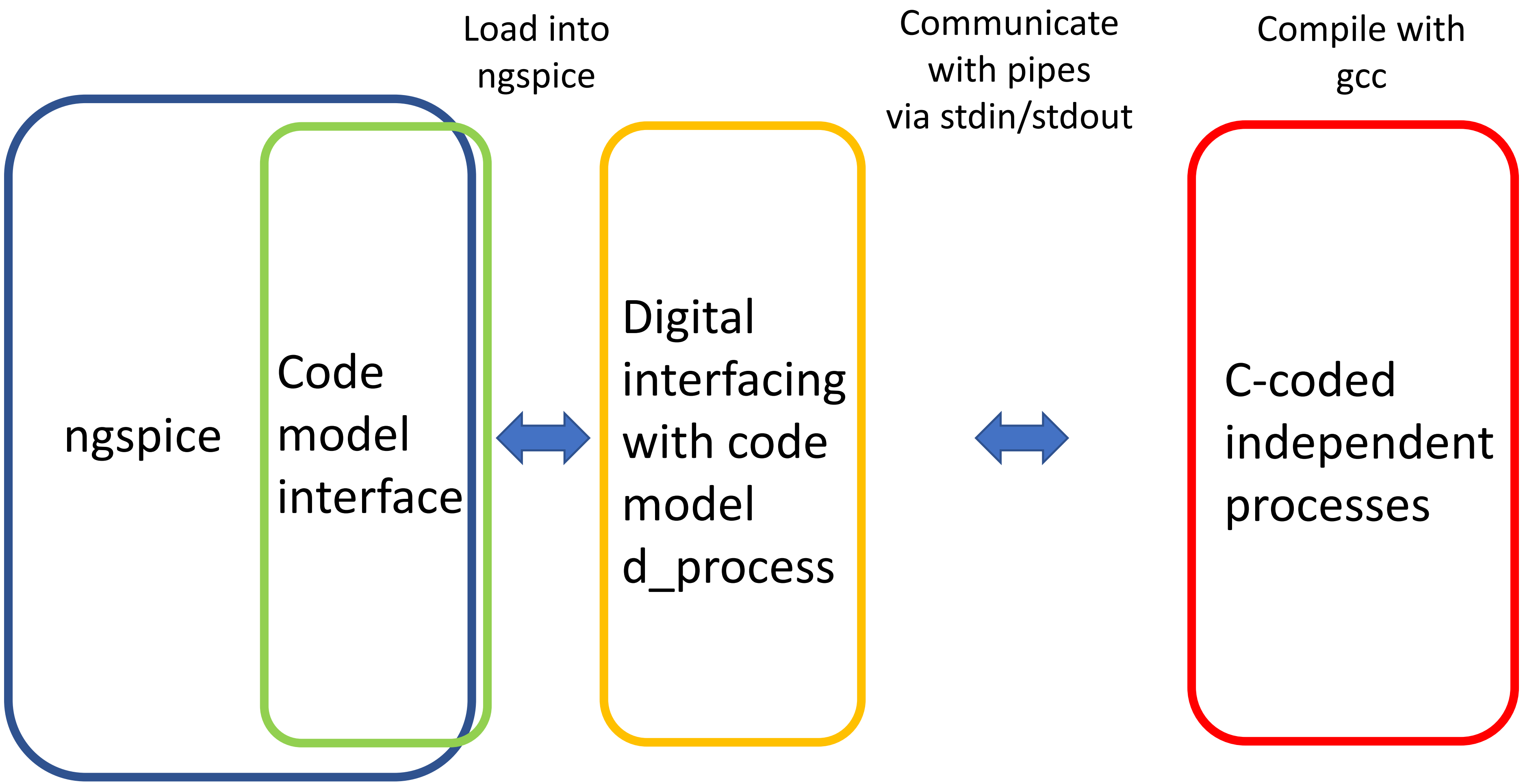
Yellow: SAR action

Magenta: Ready signal

Green: Input

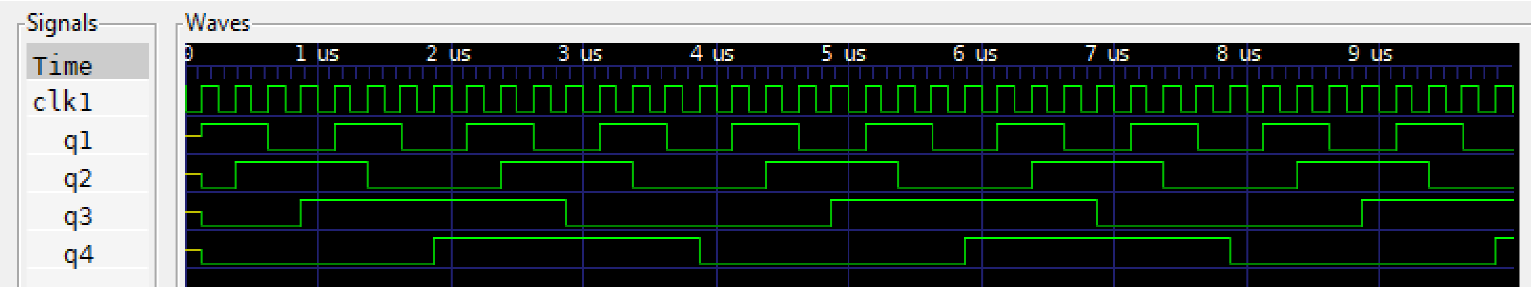
Red: Input shifted by 8.5μs, crosses the Ready period

ngspice analog and digital event based circuit netlist plus C-coded process



Idea and basic code: Uros Platise, Isotel https://www.isotel.eu/mixedsim/embedded/motorforce/index.html

Example: ngspice digital processing calling 4 external processes (e.g. graycode generator)



4 bit graycode, plotted by gtkwave

```
…
static int compute(uint8_t *dataout, int outsz,
double time)
{
    static uint8_t count = 0;
    if (count < 15) {
        count++;
    } else {
        count = 0;
    }
    dataout[0] = (count ^ (count >> 1)) & 0x0F;
    return 1;
}
…
```

- Available on Linux, Windows, macOS (not yet tested)
- Code model d_process offers digital interface
- Processes started from ngspice via execv or spawnvp
- Communication by pipes (stdin/stdout), named pipes on Linux available
- Processes controlled by clock signal

Example files: https://sourceforge.net/p/ngspice/ngspice/ci/master/tree/examples/xspice/d_process/

Schematic entry for ngspice

Why GUI?

- Netlist as input quickly becomes confusing: need for schematic entry.
- Better documentation by grouping input and output.
- ngspice as part of a larger system.
- All the others have one.

Open Source GUIs (under active development, in contact/cooperation with ngspice)

XSCHEM: main focus is IC design

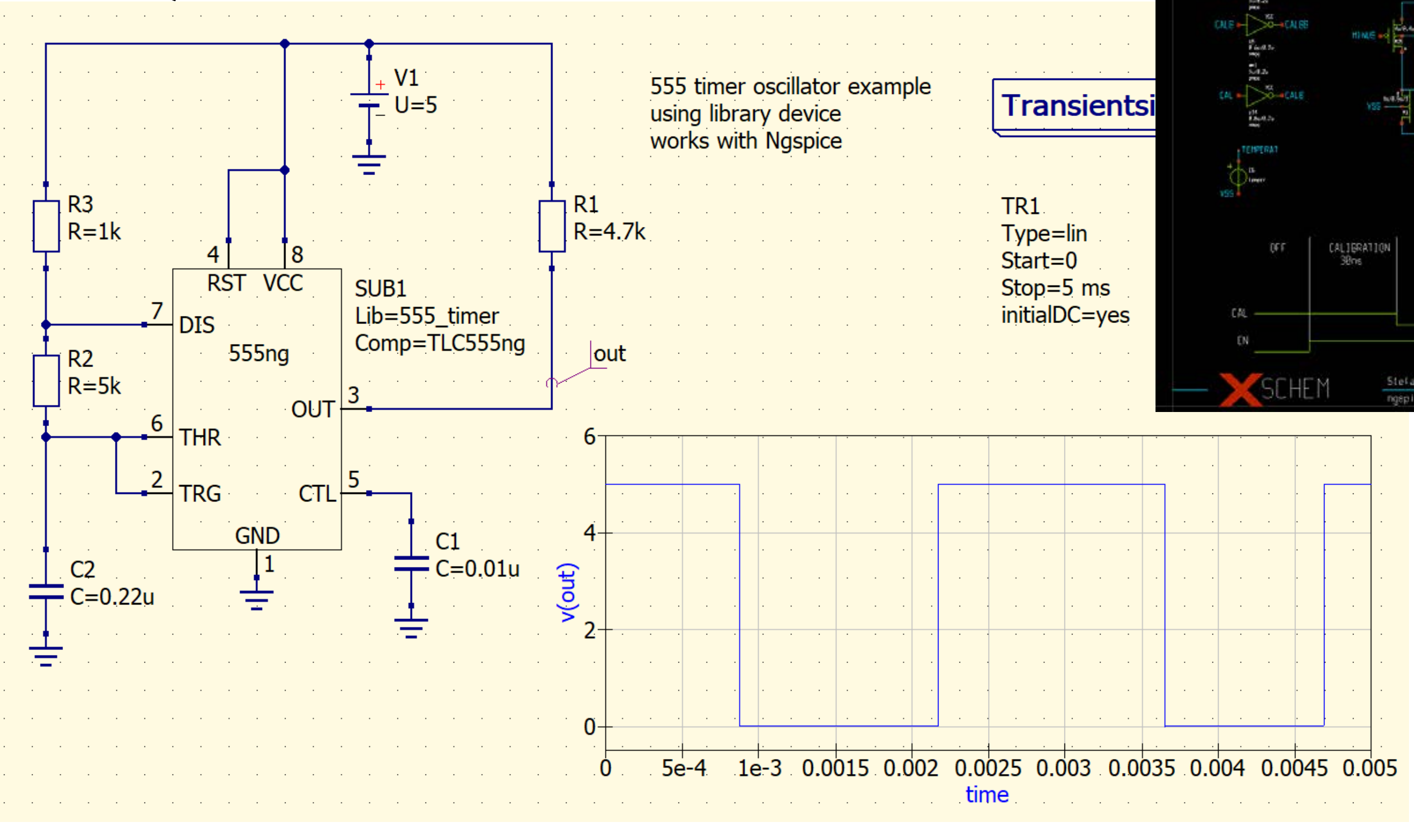
Qucs-S: universal interface, RF simulation

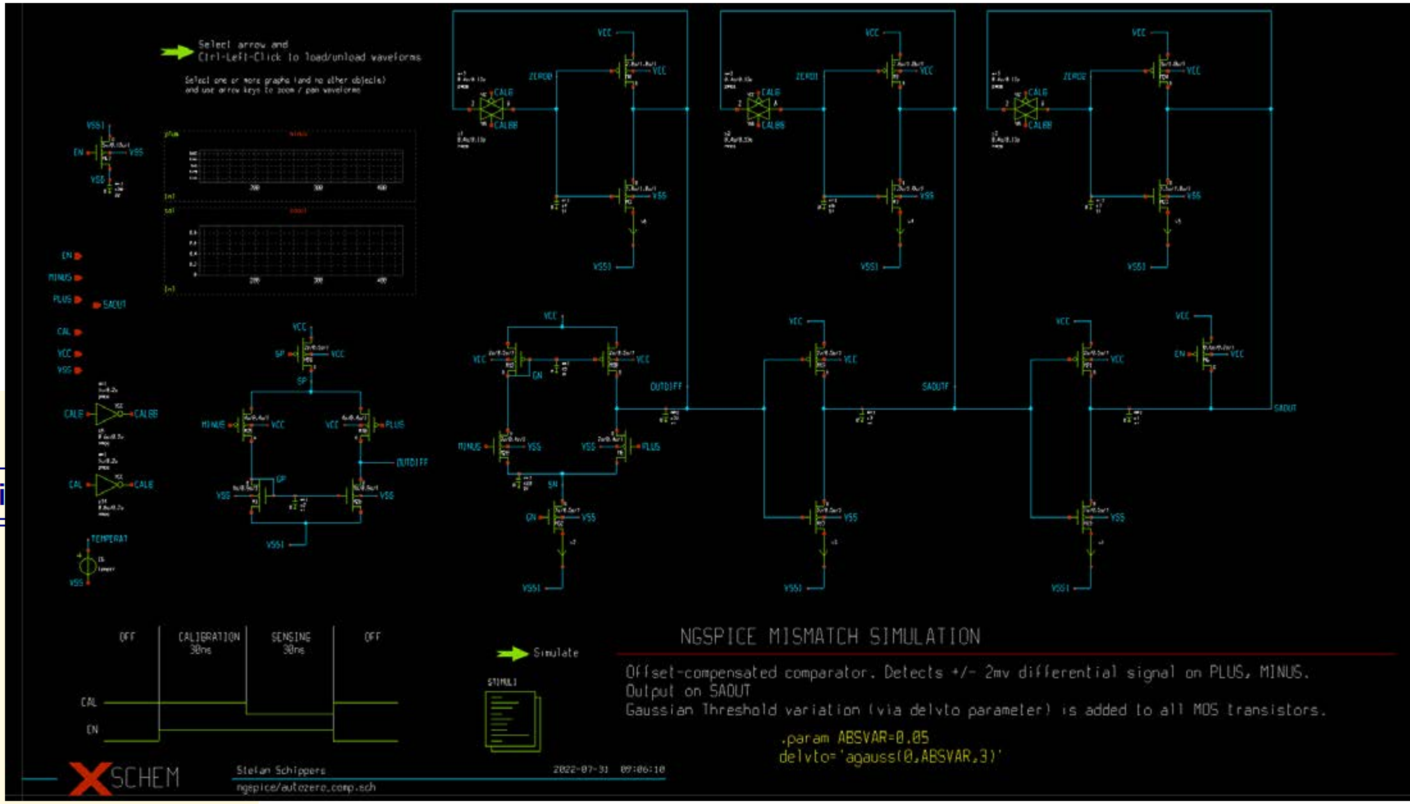
KiCad: PCB design and layout, offers embedded ngspice to support the designer

XSCHEM: https://xschem.sourceforge.io/stefan/index.html Qucs-S: https://ra3xdh.github.io/ KiCad: https://www.kicad.org/

ngspice in XSCHEM and Qucs-S

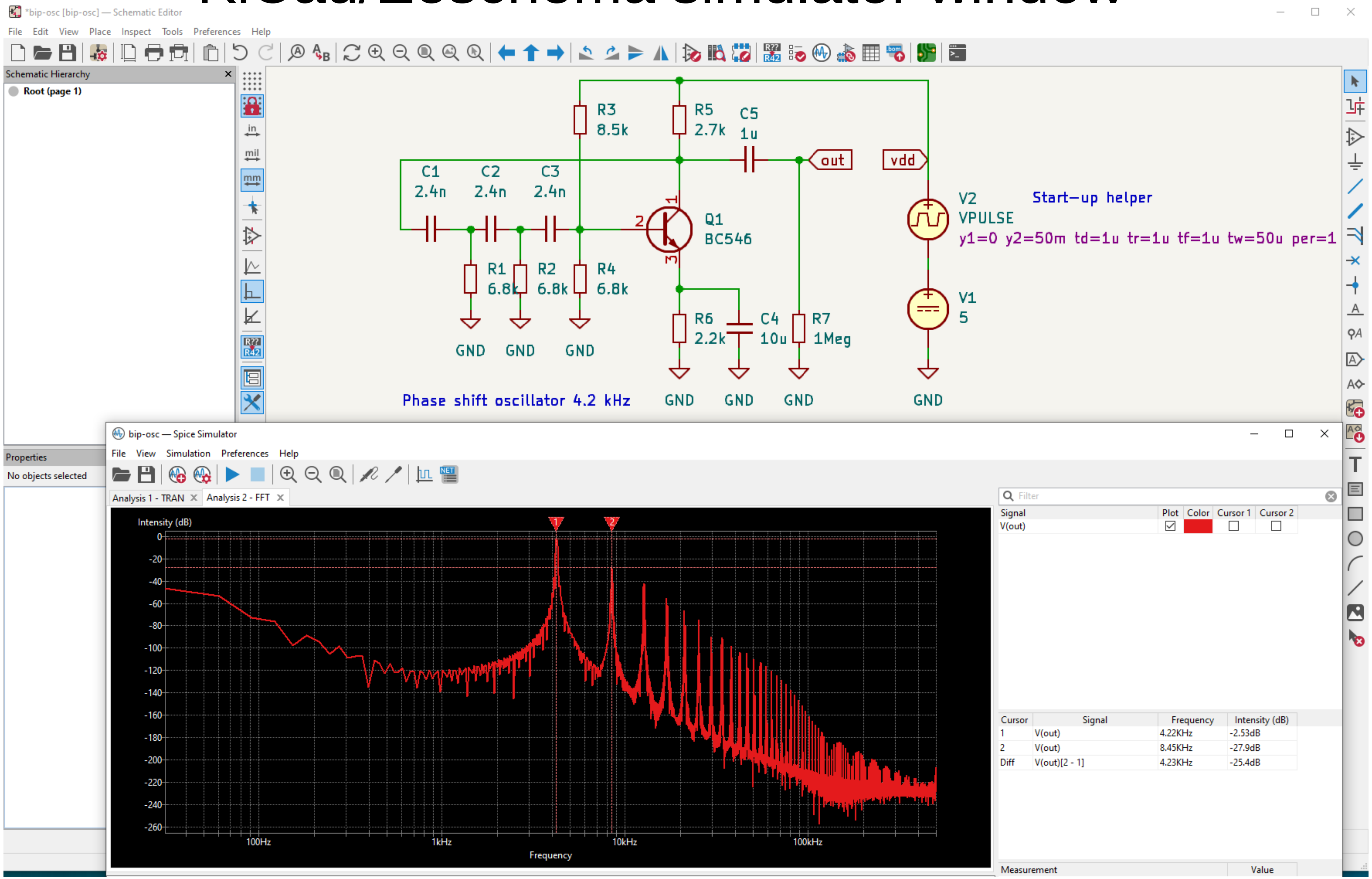
Qucs-S



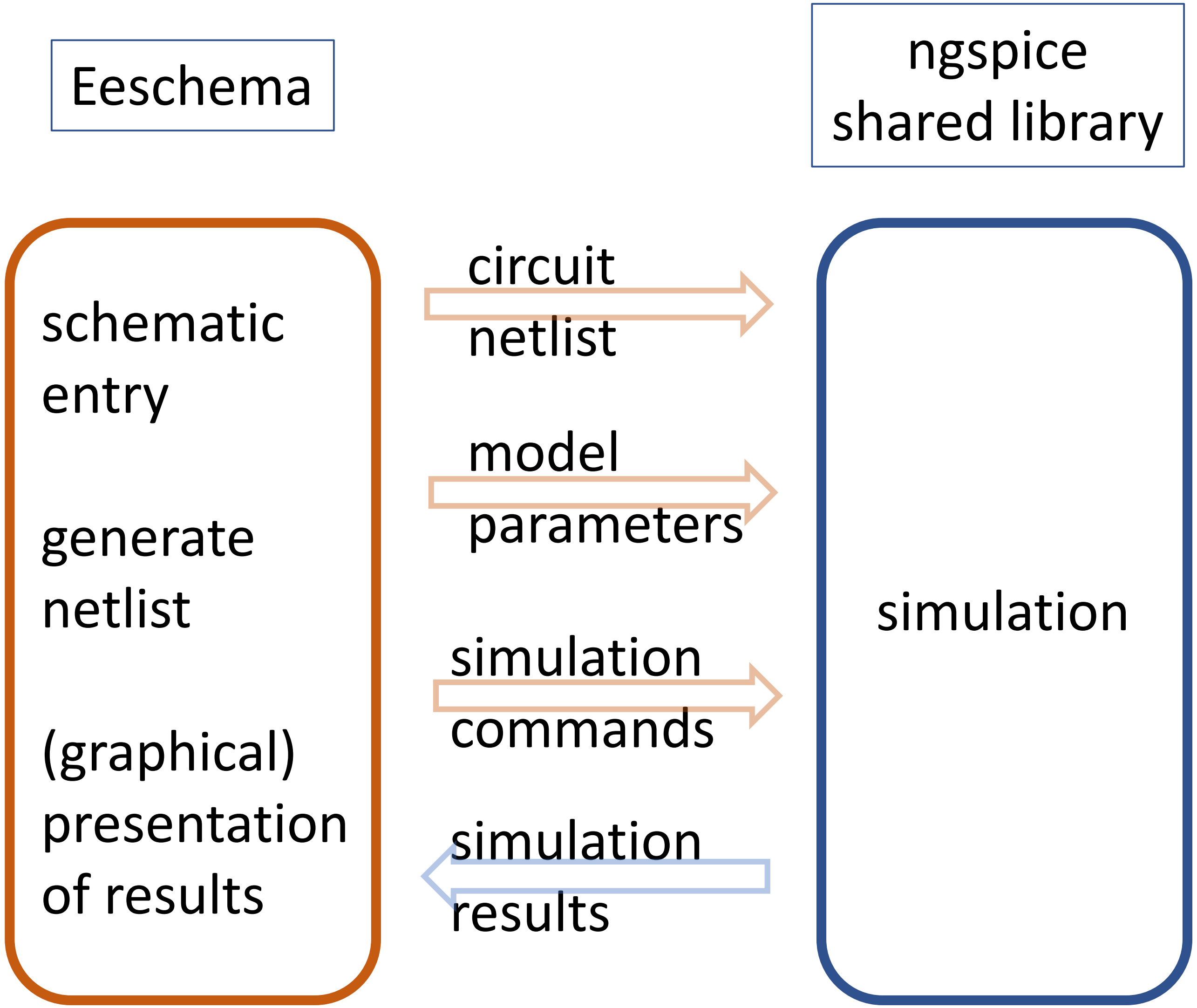


XSCHEM

KiCad/Eeschema simulator window



KiCad/ngspice Interface



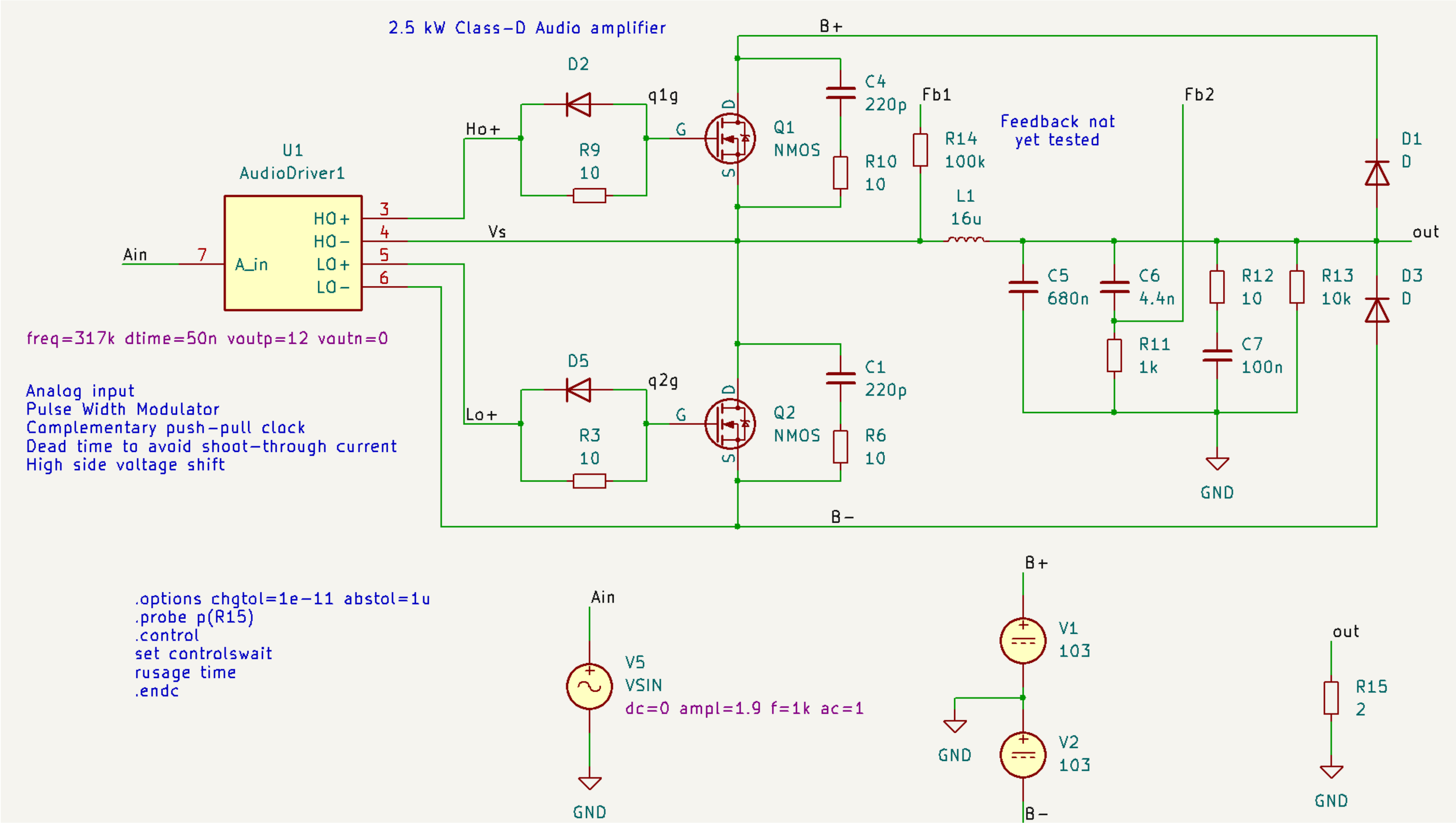
Live demo

- Simulation of a simple opamp circuit
- Transient (TRAN) and AC simulation

More examples

- 2.5 kW class D amp
- mixed signal rotary encoder
- full analog amp
- digital up-down-counter

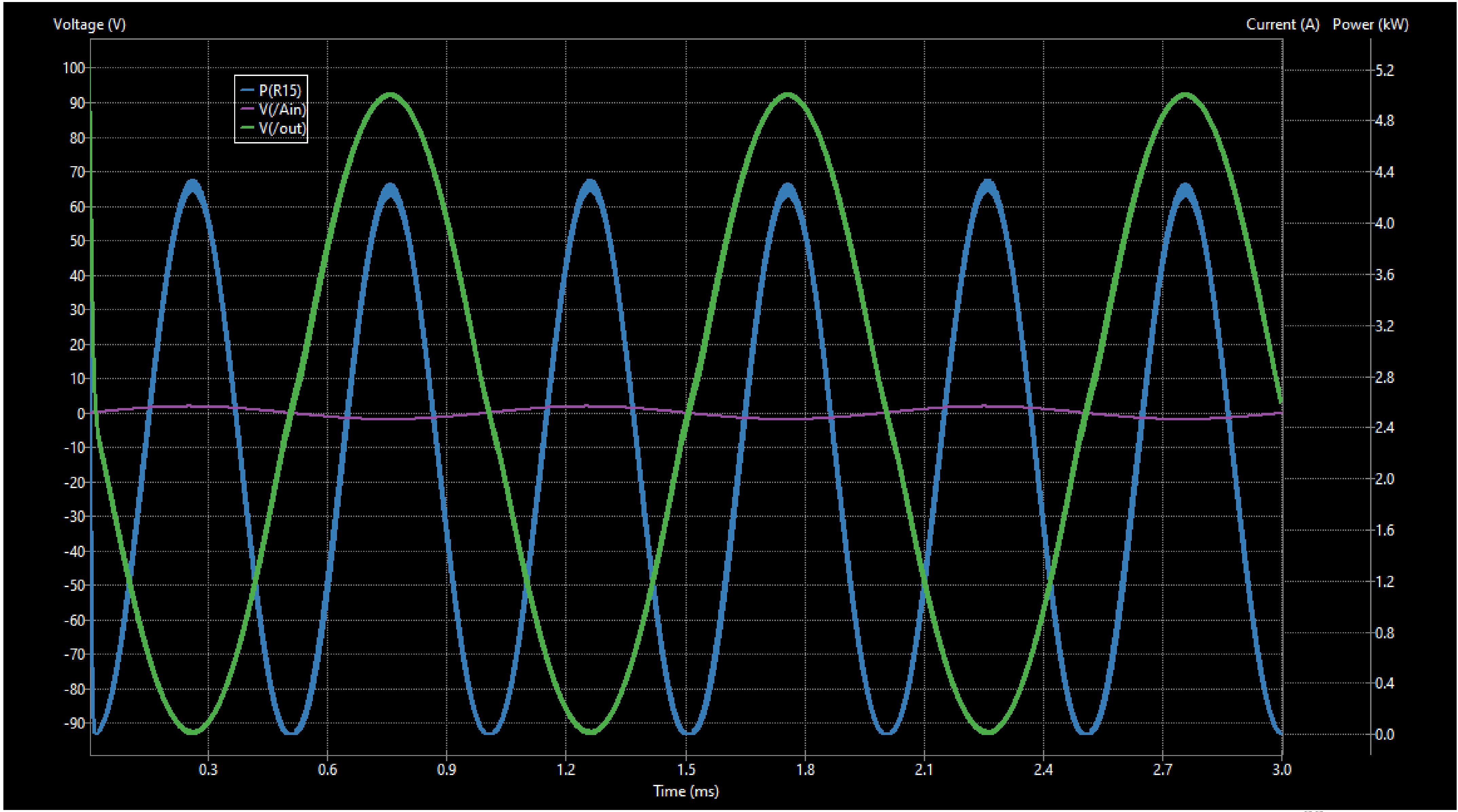
2.5 kW class D audio amplifier



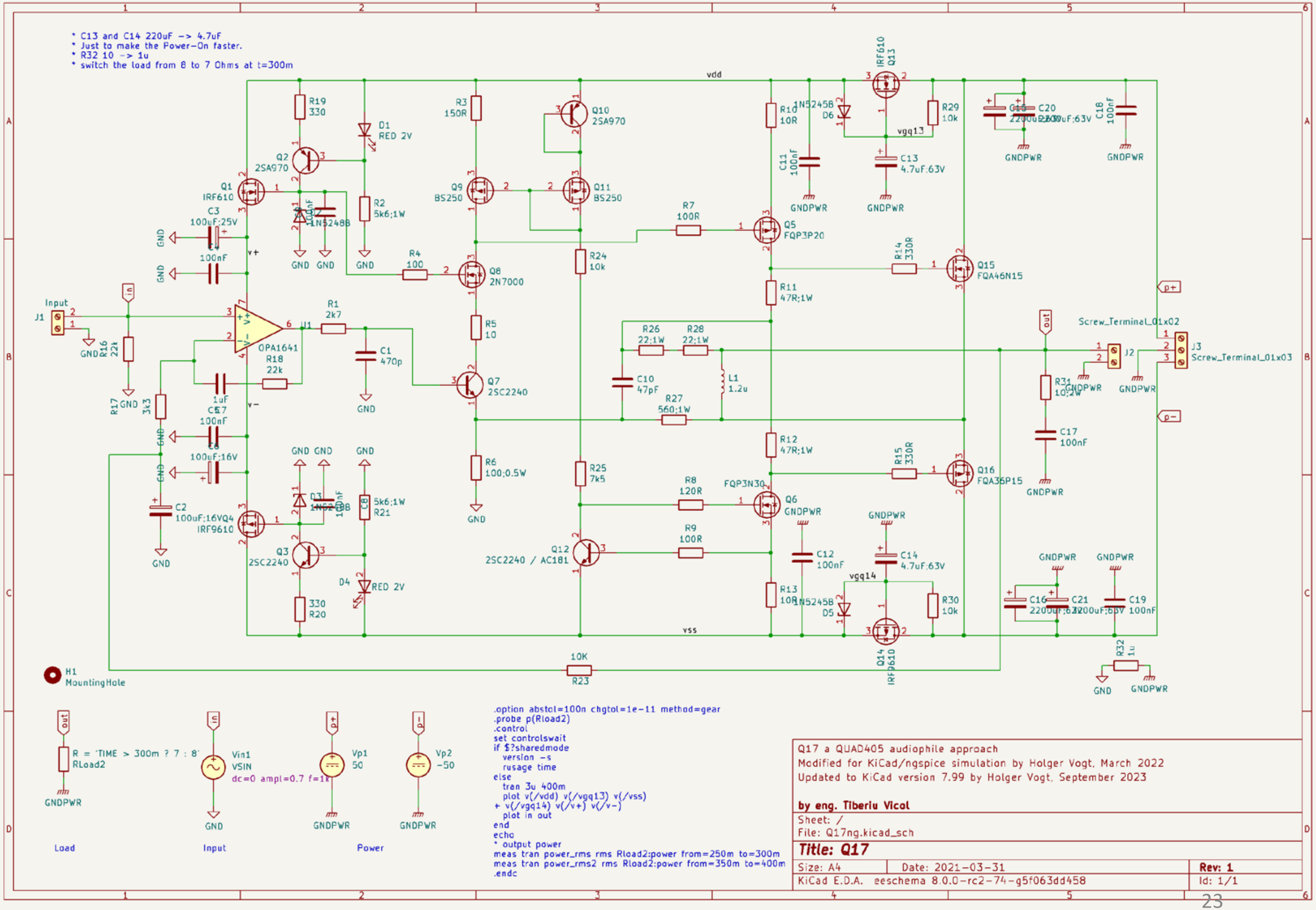
2.5 kW class D audio amplifier

Simulation time 13 s

Output power 2.59 kW



Q17, a Quad 405 audiophile amplifier

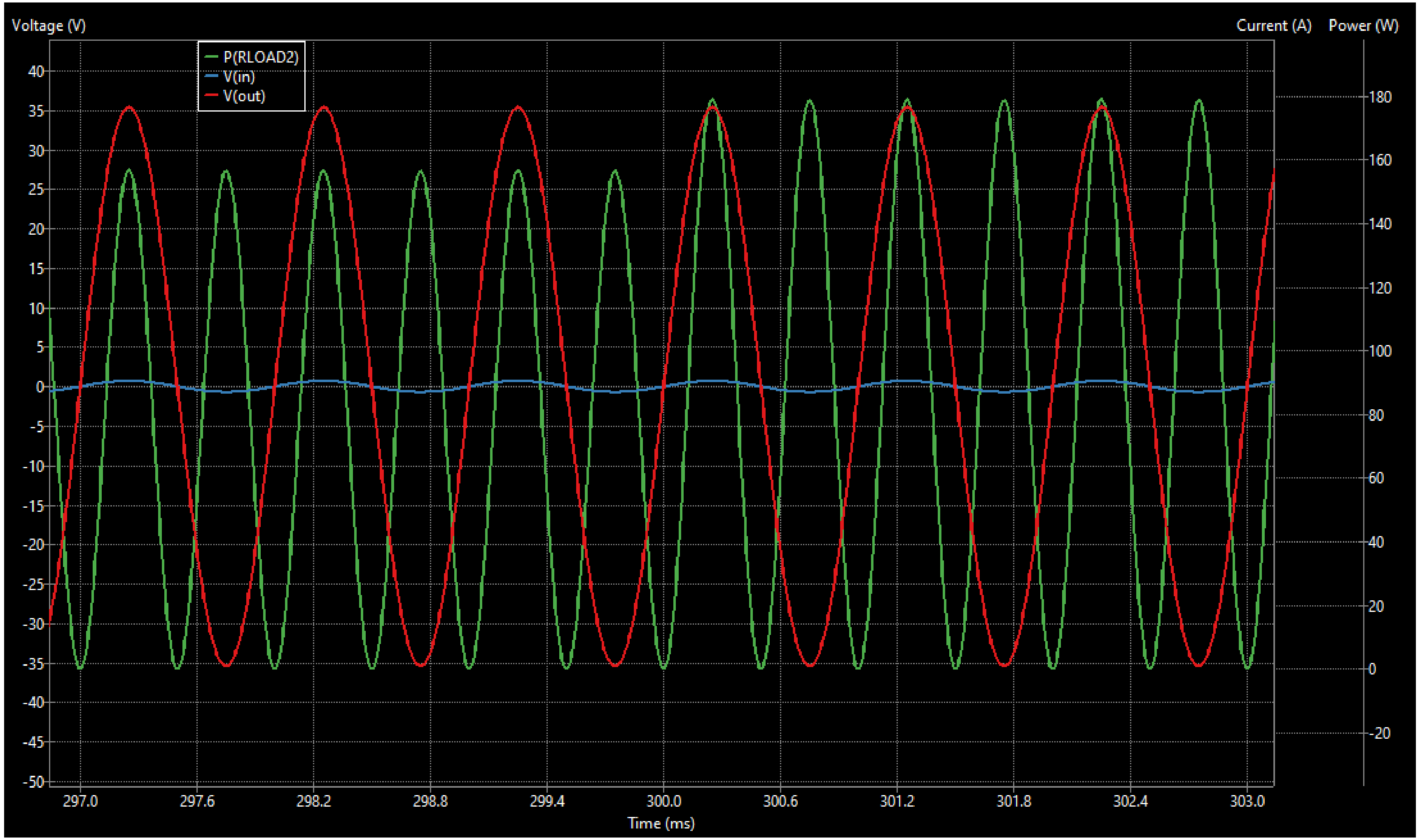


Q17, Quad 405 audiophile amplifier

Input 0.7 V, 1 kHz

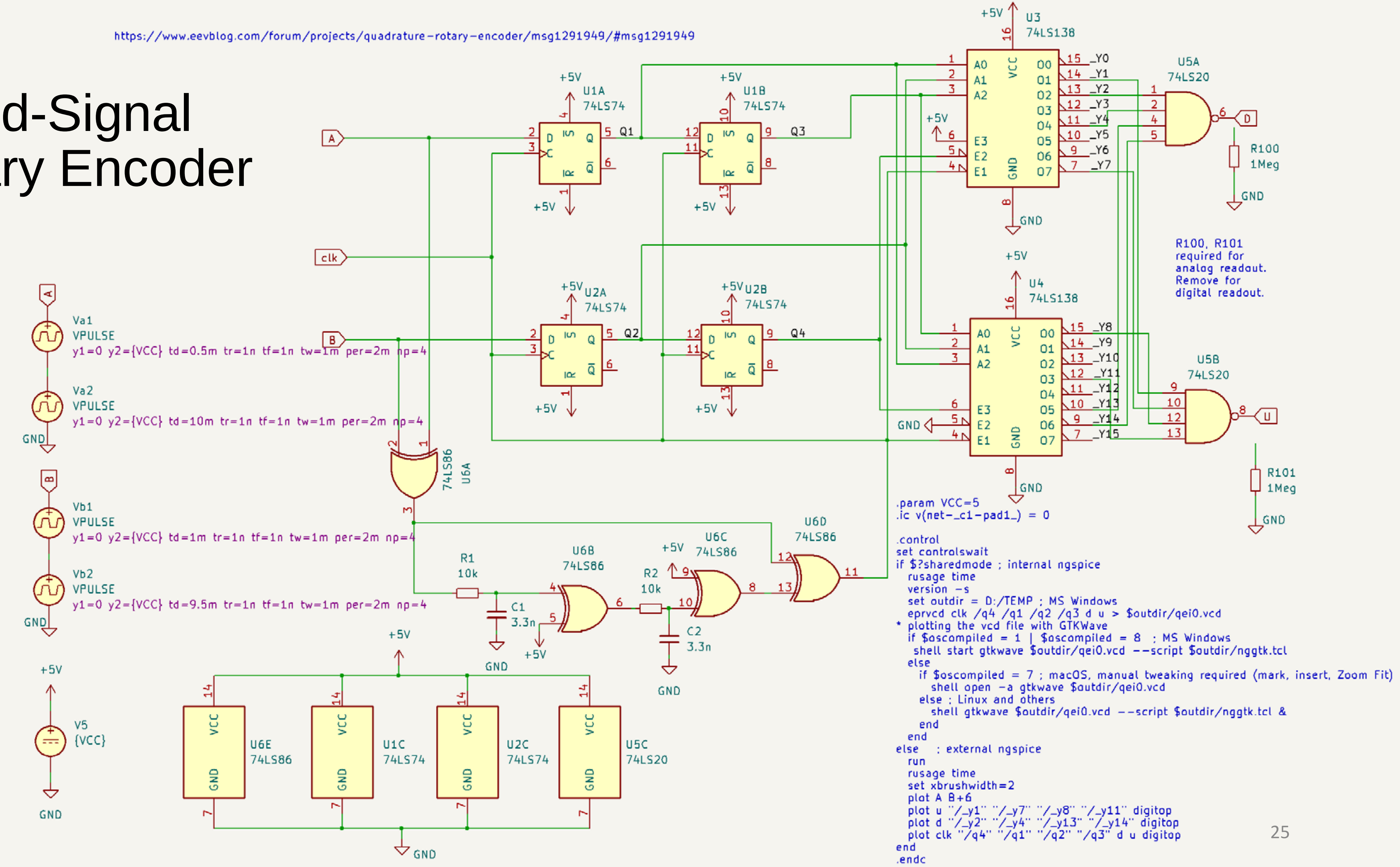
Output ca. 100 W

Load switches at 300 ms from 8 to 7 Ohms.



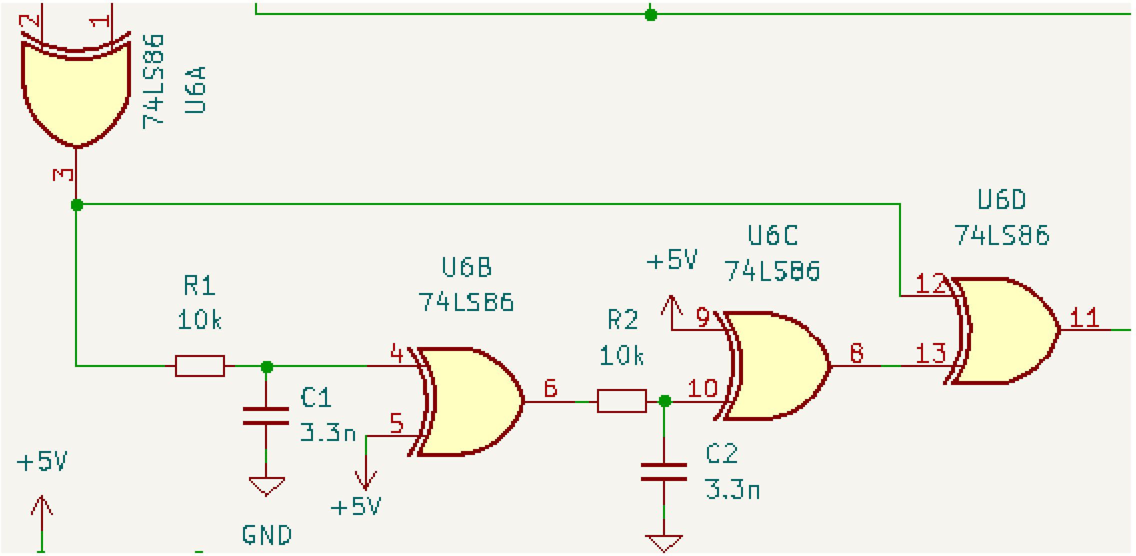
Input and output signal, output power

Mixed-Signal Rotary Encoder



Mixed-Signal Circuit Detail

In from
optical encoder



clk

Clock generator

Digital XOR

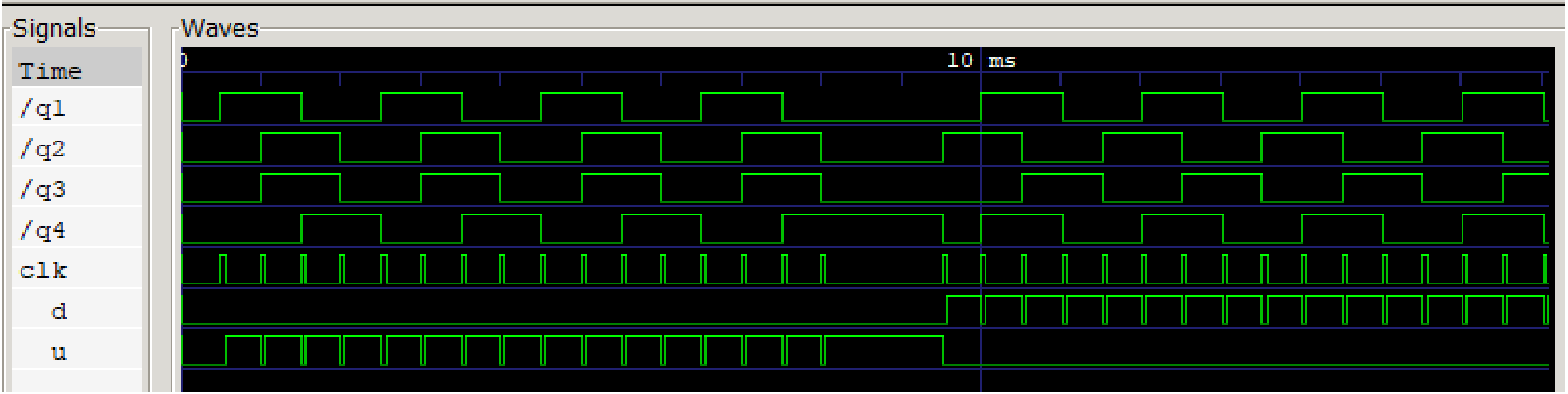
Analog RC delay

Generate clk signal of constant width,
derived from digital encoder inputs

Mixed-Signal Rotary Encoder

Simulation time 60 ms on this laptop, 25 ms on a large machine

Rotating left, rotating right

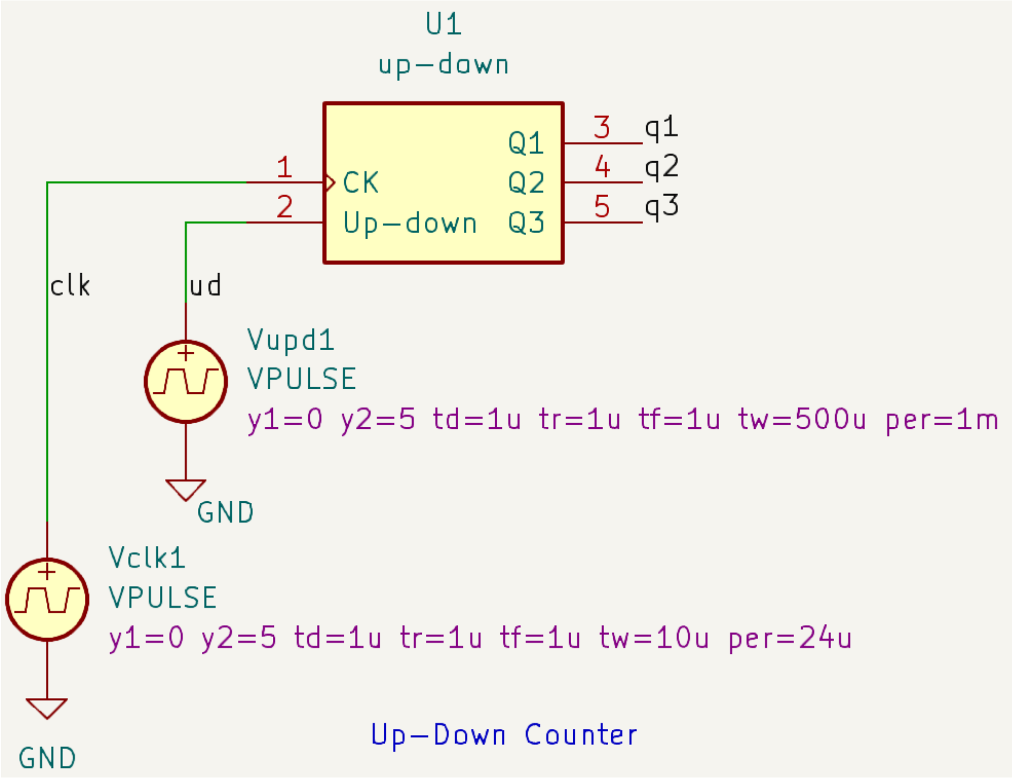


Output signal, plotted by gtkwave

Up-down counter

Digital simulation with state-machine

User-made symbol



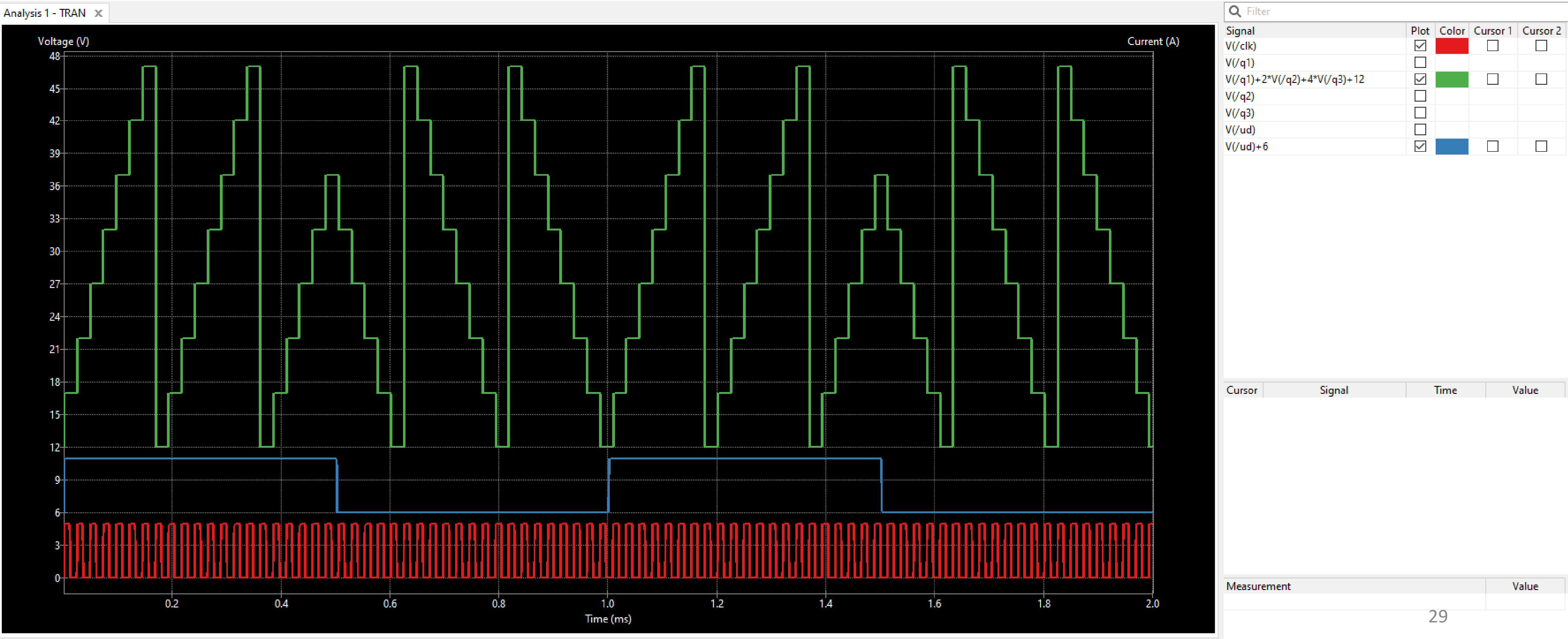
State machine input

```
*state
* output1
* |   output2
* |   |   output3
* |   |   |   input
* |   |   |   |     next state
* |   |   |   |     |
0 0s 0s 0s 0 -> 7
            1 -> 1
1 0s 0s 1z 0 -> 0
            1 -> 2
2 0s 1z 0s 0 -> 1
            1 -> 3
3 0s 1z 1z 0 -> 2
            1 -> 4
4 1z 0s 0s 0 -> 3
            1 -> 5
5 1z 0s 1z 0 -> 4
            1 -> 6
6 1z 1z 0s 0 -> 5
            1 -> 7
7 1z 1z 1z 0 -> 6
            1 -> 0
```

Up-down counter

Digital, event based simulation,
simulation time 37 ms

Plot with user-defined signals



What’s next in ngspice?

Some ideas, some more or less fixed plans, some actual activities:

- More tests with Open Source PDKs (Skywater, IHP)
- Improve RF capability by adding Harmonic Balance
- Support for reliability and degradation simulation
- Simulation of transient noise
- Improve usability of KiCad/ngspice graphics interface
- Support for ngspice digital building blocks in Eeschema
- Enhance compatibility with LTSPICE models (A devices?)

Current development branch at git: https://sourceforge.net/p/ngspice/ngspice/ci/pre-master-43/tree/

Support

Ngspice discussion forums https://sourceforge.net/p/ngspice/discussion/

Ngspice Manual https://ngspice.sourceforge.io/docs.html

KiCad-Info forum https://forum.kicad.info/

KiCad Manual (V7) https://docs.kicad.org/7.0/en/eeschema/eeschema.html#simulator

Tutorials https://ngspice.sourceforge.io/tutorials.html

Models, model parameters https://ngspice.sourceforge.io/modelparams.html

Simulation examples
https://sourceforge.net/p/ngspice/ngspice/ci/master/tree/examples/
https://forum.kicad.info/t/simulation-examples-for-kicad-eeschema-ngspice/34443
https://forum.kicad.info/t/more-simulation-examples-for-kicad-eeschema-ngspice/45546