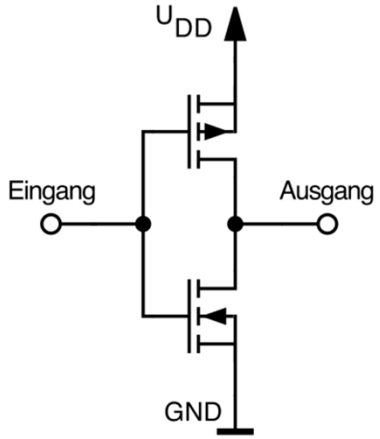
ngspice, an open source mixed signal circuit simulator

Holger Vogt
University Duisburg-Essen
Duisburg, Germany

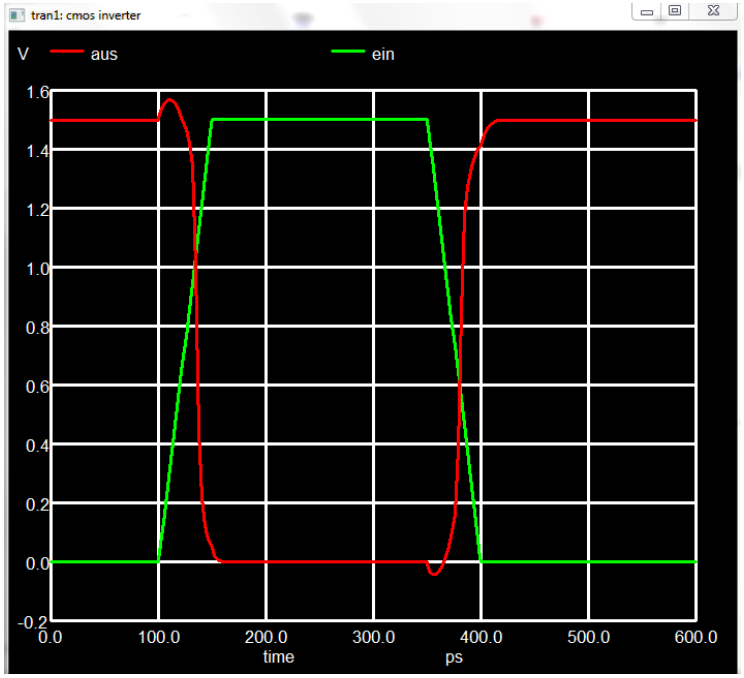
ngspice – what is it ?

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



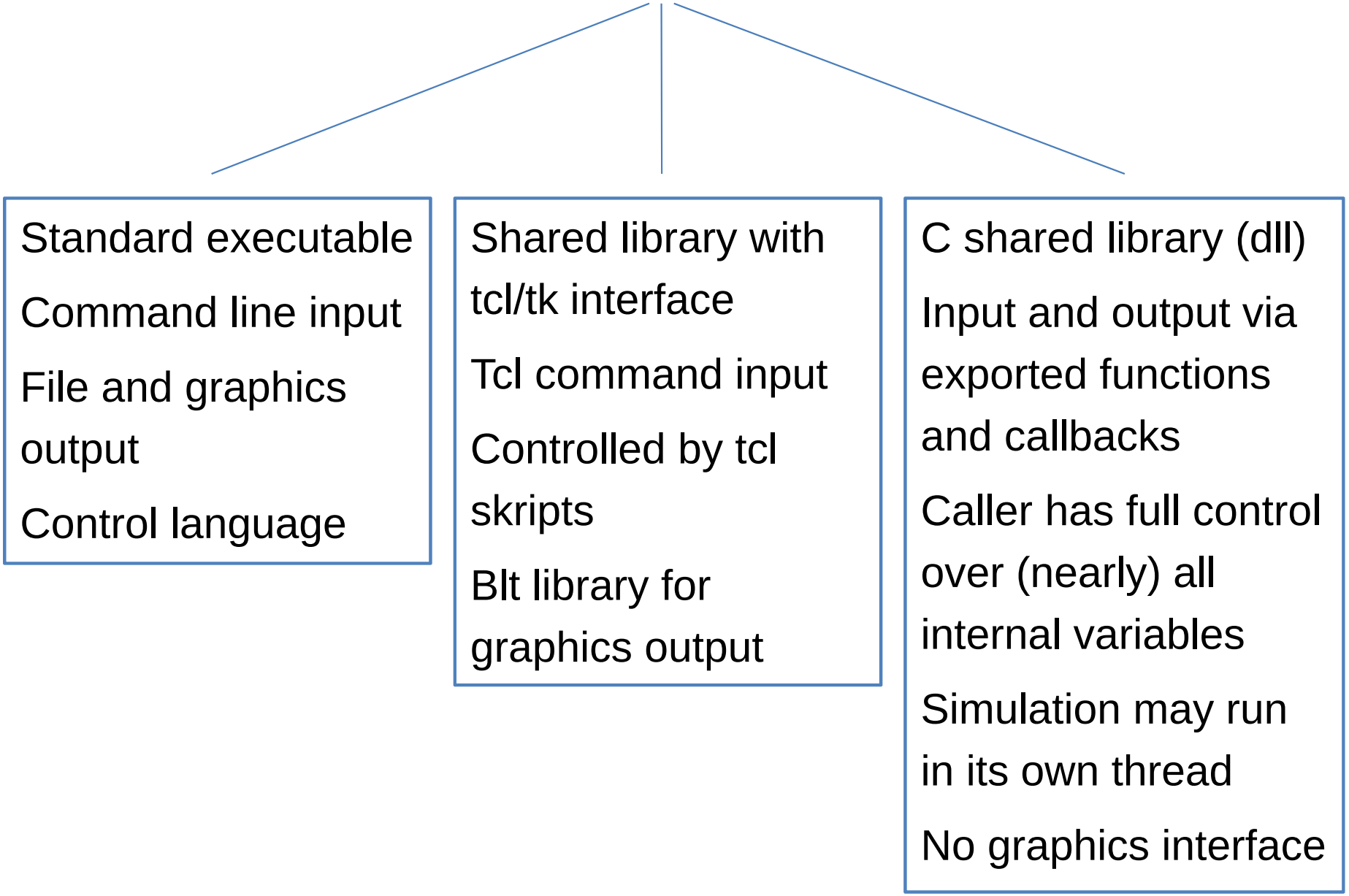

```
CMOS inverter

.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

Three flavors of ngspice



Scripting with control language

Controls

94 commands

62 math functions

91 internally defined user controllable variables

Loops etc.

Applications

To control simulation sequences, including plotting and measuring etc.

Monte Carlo simulation

Example script

```
*ng_script
* Script to run transient sim of adder-digital
.control
source adder-digital.cir
tran 500p 64000n
rusage
display
edisplay
* save data to input directory
cd $inputdir
eprvcd 1 2 3 4 5 6 7 8 s0 s1 s2 s3 c3 > adder_x.vcd
* plotting the vcd file (e.g. with GTKWave)
* For Windows: returns control to ngspice
shell start gtkwave adder_x.vcd --script nggtk.tcl
* Others
*shell gtkwave adder_x.vcd --script nggtk.tcl &
.endc
```

Device modelling

Hard coded models

MOS1-3, BSIM3, 4

BJT, JFET, MESFET

VDMOS, MOS-SOI

Transmission lines

Transient noise

(Verilog A via adms, my constant worry)

Behavioral modelling

B source

with over 30 built-in functions for user-defined models

XSPICE shared library models

C coded analog

C coded digital

Analog-digital interfaces

Macros to ease user made enhancements

Mixed signal capability offered by XSPICE

digital

event simulation

fast

no analog values, but signal strength and delays

23 predefined devices (e.g. nand, flip flops, latches, RAM, state machine, LUTs)

7 hybrid (interface) devices

analog

C coded models

versatile

analog (and frequency domain)

29 predefined devices (e.g. limiter, multiplier, file source, VCOs, table models …)

IC design support

Circuits are made of (huge numbers) of (MOS) transistors and passive components

Requirements:

Standard model support: BSIM 3, 4, (BSIMBULK) models etc.

Large circuit capability

Simulation speed (e.g. multi core support by OpenMP)

HSPICE PDK compatibility (including Monte Carlo simulation)

Digital library for XSPICE (e.g. Isotel/Yosys)

Integration into a design flow (some existing activities: gEDA, Yosys, efabless, Isotel)

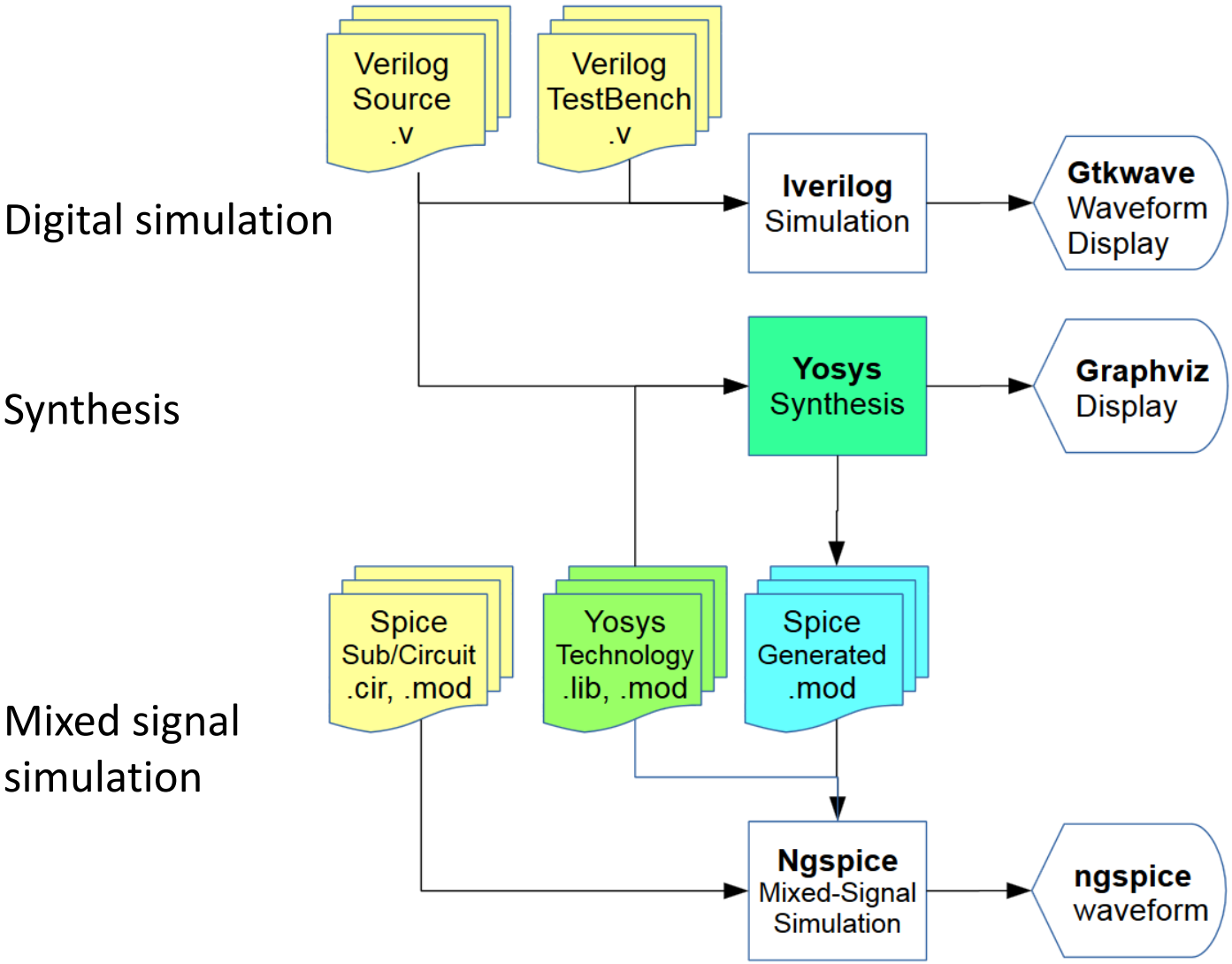
Interfacing ngspice

Standard executable: input netlist file, (script file), output files, interactive plots.

Shared ngspice: netlist, setup and simulation commands, status data, and output data; via pointer exchange over the api by exported and callback functions, multiple shared ngspice libs may be accessed in parallel by the calling program.

Dedicated (user defined) XSPICE code models: Code models are C coded shared libraries. On the ngspice side they are integrated as “devices” into a netlist. On the user side you may do whatever C/C++ coding is offering.

Example 1: An Isotel Concept

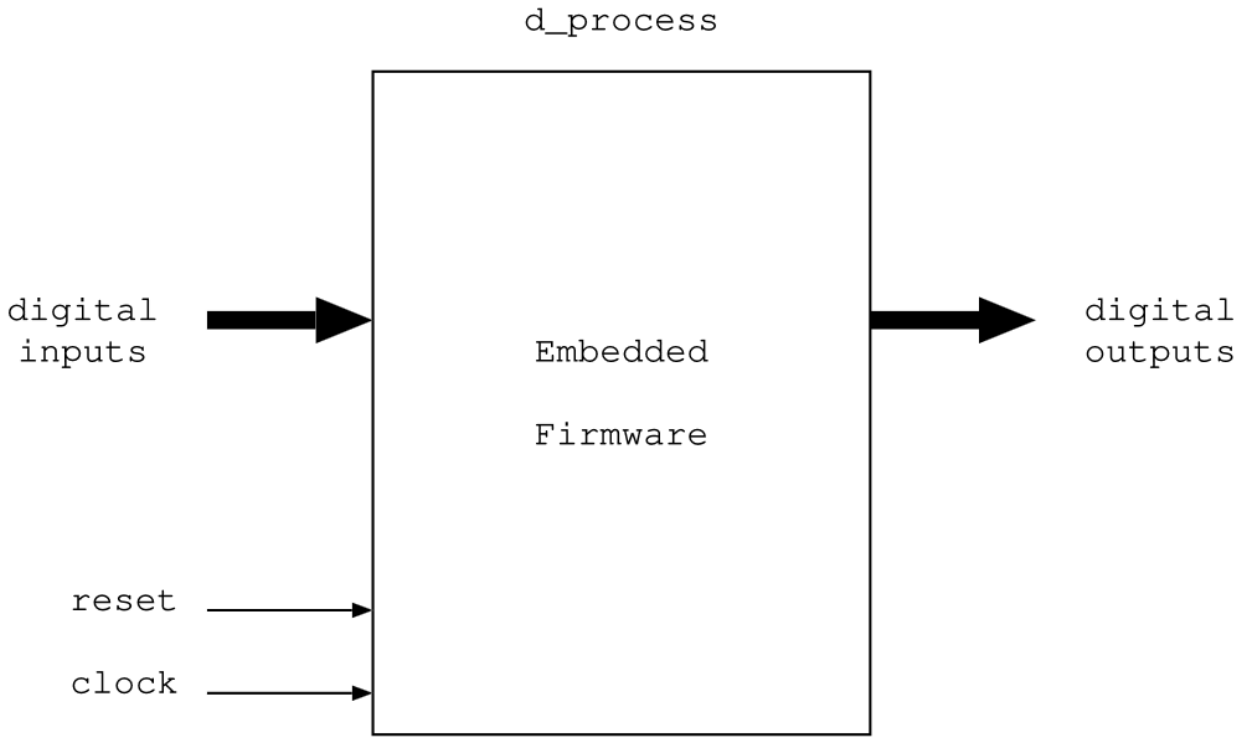


Generate ngspice subcircuits containing digital code models

https://www.isotel.eu/mixedsim/intro/concept.html

Example 2: Another Isotel Concept

Simulating C/C++ Code Together with Analog & Digital ngspice Simulation



Dedicated code model d_process establishes pipes to connect digital I/O to an embedded firmware

Examples on the web page are interacting with Cypress PSoC5 µC or are running a process inside of QEMU ARM emulator

https://www.isotel.eu/mixedsim/embedded/motorforce/index.html

Some future activities

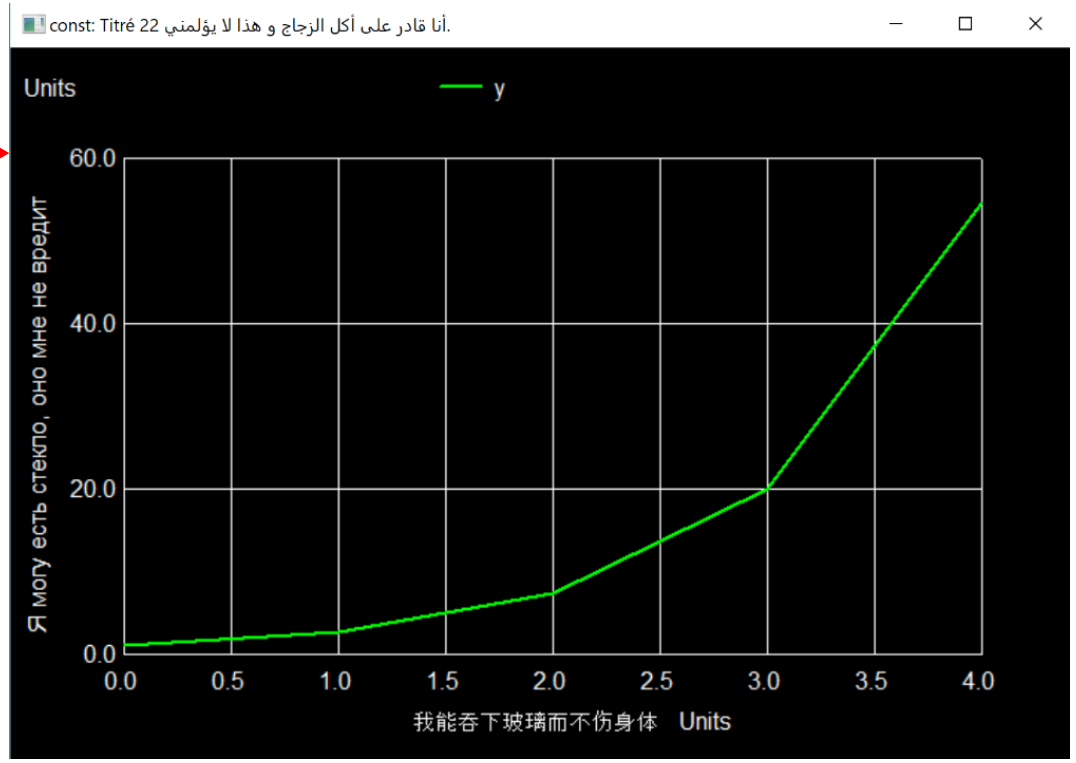
Read Verilog A model files: adms update, or use Berkeley VAPP ?

KLU for XSPICE, add KLU and allow switching between KLU and Sparse Matrix solver

utf-8 encoding for UNICODE support, enhanced graphics

XSPICE improvements (e.g. memory management)

Update and qualitity checks for sens, tf, pz



More on future activities

Task: seamless ngspice integration into mixed signal IC design flows

There are several activities on full blown design flows, like gEDA, coralEDA, Free Silicon Initiative, or company driven Efabless, Isotel

Task: better ngspice integration into KiCad, a PCB design tool, by enhanced Eeschema GUI, circuit diagrams for PCB versus diagrams for simulation, naming conventions, analysis types, device models, …

Summary

Ngspice is a versatile mixed signal simulator. It offers:

- Analog and event based simulation
- Control language for scripting
- Shared library
- XSPICE C code models
- Compatibility to existing commercial simulators

Still a lot to do …

About licenses

Ngspice: New BSD, LGPL (numparam code), and public domain (XSPICE). All are Debian DFSG compatible.

Ngspice manual: Creative Commons Attribution Share-Alike v4.0.

Verilog A models: today: various, soon to come: ECL-2 (generous rules for usage and distribution, not yet settled if DFSG compatible).

However: QA (quality assurance) data are not made available by all model developers.

Device model data: vendors' device models may be used, but perhaps not be re-distributed. Foundry PDKs require a NDA.

Obstacles: The newer and more complex ICs often come along with encrypted model files.