QUCS
Quite Universal Circuit Simulator

Overview and Status

Guilherme Brondani Torri

FOSDEM 2017
EDA Developer room
Brussels, 04 February 2016
Qucs /kjuːks/

• Overview
  – Project background
  – Features

• Status
  – Development
  – Next release

• Final Remarks
Project background

- Created at TU-Berlin
  - Michael Margraf
  - Stefan Jahn
- GPLv2+
- 20+ contributors
- 20 languages
- Cross-platform
- Users
  - Education
  - Research
  - Hobbyists
  - Industry
- 2003 to 2017
- Qucs 0.0.19 - Windows
- Website counter
  - 1,391,567 visitors since 2005/04/27
  - 3,080 downloads from 2017-01-22 to 2017-02-03
Main Features

- Schematic capture
- Simulator
- Data visualization
- Equation system
- Component library
- Design / synthesis tools

- Modeling tools
  - Spice converter (limited)
  - Equation defined device (EDD)
  - Verilog-A model builder

- Post-processing
  - Octave/MATLAB
  - Python

- Dependencies
  - C++ compiler
  - Qt4 (with Qt3Support)
  - Autotools / CMake
  - gperf / flex / bison
  - ADMS
  - LaTeX
Experimental

- **Qucs-S**
  - SPICE support
  - Ngspice, Xyce, SpiceOpus
  - Verilog-A generators
  - XSPICE generators
  - ...
  
  [https://ra3xdh.github.io/](https://ra3xdh.github.io/)

- **Gnucsator**
  - gnucap based qucsator implementation

  [https://github.com/Qucs/gnucsator](https://github.com/Qucs/gnucsator)

- **QUCS →gschem**

  [https://github.com/erichVK5/translate2geda/](https://github.com/erichVK5/translate2geda/)
Support

- Website

- Active maintainers:
  - Guilherme Brondani Torri
  - Claudio Girardi
  - Vadim Kuznetsov
  - Felix Salfelder
  - Andrés Martínez Mera
  - Mike Brinson

- Documentation
  - Help
  - Tutorial Workbook
  - Report Workbook
  - Technical Manual

- SourceForge
  - Binaries
  - Git repository (mirror)
  - Issue tracker
  - Forum / mailing lists

- GitHub
  - Git repository (preferred)
  - Issue tracker
  - Wiki
  - Travis CI
  - AppVeyor
Tools

- **Graphical Interface**
  - Qucs
  - ActiveFilter
  - Attenuator
  - Editor
  - Filter
  - Help
  - Matching
  - Library
  - Rescodes
  - Transcalc

- ~ 170 components

- **Command Line**
  - qucs
  - qucsator
  - qucsconv

- **Third-party and scripts**
  - asco
  - admsXml
  - iverilog
  - freehdl
  - ps2sp
  - octave
  - python
Projects
Contents
Components

- Resistor
- Resistor US
- Capacitor
- Inductor
- Ground
- Subcircuit Port
- Transformer
- symmetric Tran...

 transient simulation

- V1 U=13.5 V
- S1 time=0.5 us
- Pr1 R1=4 Ohm
- L1 L=2 uH
- C1 C=450 pF
- TR1 Type=lin
  - Start=0
  - Stop=3 us
  - Points=1000
  - v(Vt)

Graph:
- x-axis: Time (us)
- y-axis: Voltage (V)
- Waveform showing transient response
Qucs-ActiveFilter

Filter parameters
- Passband attenuation, Ap (dB): 3
- Stopband attenuation, As (dB): 20
- Cutoff frequency, Fc (Hz): 1000
- Stopband frequency, Fs (Hz): 1200
- Passband ripple Rp (dB): 3
- Passband gain, Kv (dB): 0
- Filter order: 5

Transfer function and Topology
- Approximation type: Butterworth
- Filter type: High Pass
- Filter topology: Sallen-Key (S-K)

Filter calculation console
- Filter order = 14
- Poles list: 
  Pk = Re + j*Im
  -0.111964 + j0.993712
  -0.330279 + j0.943883
  -0.532032 + j0.846724
  -0.707107 + j0.707107
  -0.846724 + j0.532032
  -0.943883 + j0.330279
  -0.993712 + j0.111964
  -0.993712 + j-0.111964
Qucs-Attenuator

**Topography:**
- **Tee**

**Input:**
- **Attenuation:** 1 dB
- **Z\text{in}:** 50 Ohm
- **Z\text{out}:** 50 Ohm

**Output:**
- **R1:** -- Ohm
- **R2:** -- Ohm
- **R3:** -- Ohm

**Result:**
Qucs-Help

DEPRECATED

http://qucs-help.readthedocs.io/

Getting Started with Digital Simulations

Qucs is also a graphical user interface for performing digital simulations. This document should give you a short description on how to use it.

For digital simulations Qucs uses the FreeHDL program (http://www.freehdl.seul.org). So the FreeHDL package as well as the GNU C++ compiler must be installed on the computer.

There is no big difference in running an analog or a digital simulation. So having read the Getting Started for analog simulations, it is now easy to get a digital simulation work. Let us compute the truth table of a simple logical AND cell. Select the digital components in the combo box of the components tab on the left-hand side and build the circuit shown in figure 1. The digital simulation block can be found among the other simulation blocks.

The digital sources S1 and S2 are the inputs, the node labeled as Output is the output. After performing the simulation, the data display page opens. Place the diagram truth table on it and insert the variable Output. Now the truth table of a two-port AND cell is shown. Congratulations, the first digital simulation is done!
Qucs-Matching

Create Matching Circuit

- Calculate two-port matching
- Reference Impedance:
  - Port 1: 50 ohms
  - Port 2: 50 ohms
- S Parameter:
  - Input format: real/imag
  - S11: 0.5 +j 0
  - S21: 0.5 +j 0
  - S12: 0 +j 0
  - S22: 0.5 +j 0
- Frequency: 1

Device:
- L1: L=11.254nH
- C1: C=1.50053pF
- L2: L=11.254nH
- C2: C=1.50053pF

Paste into schematic ➔
Qucs-Lib

Diodes

1N4148
1N4148W
1N4148WS
1N4148WT
1N4001
1N4002
1N4003
1N4004
1N4005
1N4006
1N4007
1N5400
1N5401
1N5402
1N5404

Name: 1N4148
Library: Diodes
universal silicon switching diode
75V, 300mA, 4.0ns
Manufacturer: Diodes Inc.

Symbol: ![Diode Symbol]

Drag n'Drop me!
Qucs-Rescodes

Paste into schematic →

R1
R=47 Ohm
Qucs-Transcalc

Transmission Line Type

- Coaxial Line

Substrate Parameters

- $E_r = 2.1$
- $\epsilon_r = 1$
- $\tan \delta = 0.002$
- $\sigma = 4.1 \times 10^7$

Physical Parameters

- $d_{in} = 40$ mil
- $d_{out} = 134$ mil
- $L = 1000$ mil

Component Parameters

- P1
  - Num = 1
  - $Z = 50$ Ohm

- CXTC1
  - $er = 2.1$
  - $D = 3.4036$ mm
  - $d = 1.016$ mm
  - $L = 25.4$ mm

- P2
  - Num = 2
  - $Z = 50$ Ohm

S parameter simulation

- SPTC1
  - Type = log
  - Start = 1 GHz
  - Stop = 100 GHz
  - Points = 51

Paste into schematic
Qucs-Filter

Filter
Realization: LC ladder (pi type)
Filter type: LC ladder (tee type)
Filter class: C-coupled transmission lines
Order: Microstrip end-coupled
Corner frequency: Coupled transmission lines
Stop frequency: Coupled microstrip
Stop band frequency: Stepped-impedance
Pass band ripple: Stepped-impedance microstrip
Stop band attenuation: Equation-defined
Impedance: 50 Ohm

Microstrip Substrate
Relative permittivity: 9.8
Substrate height: 1.0 mm
metal thickness: 12.5 um
minimum width: 0.4 mm
maximum width: 5.0 mm

Calculate and put into Clipboard
Result: --

Paste into schematic
Command Line Tools

• Qucs – schematic
  – schematic to netlist
  – schematic to print
  – dump components data

• Qucsator – simulator
  – DC
  – Transient
  – AC
  – AC Noise
  – **S-Parameter**
  – S-Parameter Noise
  – (Harmonic Balance)

• Qucsconv - converter
  spice - qucs
  spice - qucslib
  vcd - qucsddata
  qucsddata - csv
  qucsddata - touchstone
  citi - qucsddata
  touchstone - qucsddata
  csv - qucsddata
  zvr - qucsddata
  mdl - qucsddata
  qucsddata - matlab

• Custom file formats
  – schematic
  – library
  – netlist
  – data file
Verilog-A

- Includes 53–38 models written in Verilog-A (GPL)

- Compact models
  - BSIM 3, 4, 6 (Berkeley)
  - EKV (EPFL)
  - HICUM L0, L2 (TU-Dresden)
  - FBH-HBT (TU-Berlin)

- ADMS (Automatic Device Model Synthesizer)
  - Verilog-A → XML transformations → “XYZ code”
  - Subset of Verilog-AMS

- QUCS limitations
  - Not supported: $V(n) <+ \ldots$ ;
Demo

• Examples
  – RLC circuit, parameter sweep
  – 555 timer: macro modeling
  – Optimization: Band-pass filter
  – 10 GHz microstrip band-pass filter
  – Verilog counter
  – Verilog-A support / model builder
The voltage overshot strongly depends on the quality of the resonance circuit.
555 macro model
Microstrip band-pass filter

10GHz band pass filter
Created by Toyoyuki ISHIKAWA

[Diagram with component labels and S-parameter simulation graph]
Verilog Counter

```verilog
module counter(out, clk, reset);
    parameter WIDTH = 2;
    output [WIDTH-1 : 0] out;
    input clk, reset;
    reg [WIDTH-1 : 0] out;
    wire clk, reset;

    always @(posedge clk)
        out <= out + 1;

    always @reset
        if (reset)
            assign out = 0;
        else
            deassign out;

endmodule // counter
```

Verilog simulation with Icarus Verilog
Check the 'counter.v' for the HDL model.
Status

• Release 0.0.19 (January 22, 2017)
  – Bug fixing, usability improvements, build system cleanup
  – Ongoing port Qt3Support to Qt4
  – New active-filter synthesis tool
  – Integration of regression tests
  – Removal of non-GPL models
  – Adopted git-flow branching model
  – 168 issues closed

• Develop (Release 0.0.20)
  – RF and Microwave
  – Tuner
  – Removal of qucs-editor and qucs-help
  – … (quick release)
Resources

- Website: http://qucs.sourceforge.net/
- GitHub (preferred): https://github.com/Qucs/qucs/
- SourceForge: http://sourceforge.net/p/qucs/git/
- Mailing lists: http://sourceforge.net/p/qucs/mailman/
- IRC channel: #qucs
- Forum: http://sourceforge.net/p/qucs/discussion/
- Bug trackers:
  - https://github.com/Qucs/qucs/issues
  - http://sourceforge.net/p/qucs/_list/tickets
- Source code documentation:
- Downloads: http://sourceforge.net/projects/qucs/files/
- (NEW) Transifex translations:
Final remarks

• User friendly
• Advanced components and modeling features
• We are open for collaboration
• Help is welcome