



# NL5

## Circuit Simulator with Ideal Components

Alexei Smirnov  
Sidelinesoft, LLC

APEC-2018 Exhibitor Seminar  
March 7, 2018

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)




1.



## Circuit simulators



2. There are many circuit simulators available today, here are just few of them. They have different flavors (mostly SPICE-based), platforms, complexity, performance, capabilities, and of course price.



## Circuit simulator tasks

- Modeling integrated circuits (IC)
- Evaluating new circuits on the computer
- Reliability and failure analysis
- System-level simulation
- Design verification
- And more...


[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



### 3. What are they used for? Different tasks:

- Modeling integrated circuits: this is what SPICE algorithm was originally developed for.
- Evaluating new circuit on the computer, rather than building it on a bread-board: it's a huge time and resource saving.
- Perform reliability and failure analysis. Simulate, for example, power supply without burning out and replacing expensive transistors after few seconds of operation.
- And much more.

All those tasks are quite different, and they require different functionality and capabilities of circuit simulation tool. So, it is logical to expect that one simulator can be good for one specific task, and another simulator will perform better for another task.



## “Use the right tool for the right task”

Do engineers follow that rule?  
Not always. Usually, they use:

- what is available
- what is provided/required
- what they are comfortable with

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)




4. Here is a popular expression: "Use the right tool for the right task". The question is: do engineers follow that rule? Do they understand that using the wrong tool for a specific task may easily give non-reliable, or even absolutely wrong results? I have an impression that, unfortunately, not always.

They usually use:

- whatever they have available at the moment,
- or what is provided (or even required) by their employer,
- or what they are comfortable with for any reasons – like “I was using this tool for many years and I have no desire to try anything else”.

Is this the right approach? Is it good for business? Definitely not.




“Use the right tool for the right task”

Using right simulator for the right task, even if it would take some efforts, time, and money, could increase efficiency and quality of your design.

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



5. I believe that using right tool for the right task, even if it would take some efforts, time, and even money, could increase efficiency and quality of your design.



## What is NL5?

### **NL5:** “the right tool” for specific tasks

- what makes NL5 “special”
- what tasks NL5 is good for
- NL5 applications
- NL5 features and capabilities

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



6. What is NL5? Is it just another circuit simulator, doing the same things, in the same way, maybe just a little bit different?

No. NL5 is a tool created for specific tasks. It is much more efficient and reliable for those tasks comparing to other tools. In other words, NL5 is “the right tool” for those tasks.

In this presentation, am going to talk about

- what is “special” about NL5
- what tasks NL5 is good for and other simulators are not
- what applications NL5 can be used for
- maybe a little bit about NL5 features and capabilities



## History

- NL = “Non-Linear”
- started 30 years ago
- used to design electronics for science
- NL5 is 5-th generation
- offered to public in 2007

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



7. A little bit about NL5 history.

The “NL” project, which stands for “non-linear”, was started about 30 years ago. It was developed strictly for personal use of myself, my friends, and colleagues. Majority of that was developing electronics for scientific research.

NL5 is a “fifth” generation, it was offered to public in 2007, more than 10 years ago.

Currently many individual users and Companies are actively using NL5 for quite different tasks. As far as I know, those tasks are exactly the tasks NL5 was designed for. Which just proves once again: smart people are using the right tool for the right task.



## NL5 basics


- Components
- Models
- Algorithm

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



8. Let's talk about what is special about NL5. There are 3 main subjects here: components, models, algorithm.





## NL5 basics

### **Not a SPICE**

Not Simplis, not PSIM, not Plecs, ...

I will refer to all other simulators  
as **SPICE**

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



9. A note: I will compare NL5 with ALL other simulators, but since most of them are SPICE-based, I will use word "SPICE" for all of them. I will also use "SPICE" for Simplis, PSIM, and Plecs, which is not correct, however NL5 is different from those simulators as well.



## Components

### **SPICE**

- close to “real” parts
- “reasonable” limits

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



10. Let’s start from simple “basic” components: such as resistor, capacitor, inductor, switch, voltage and current sources. Most of simulators have restrictions on component parameters, because their algorithms are designed and optimized to deal with “real” parts. That means, the parameters of components should be within some “reasonable” limits.

Most likely, you won’t be allowed to set infinite gain of an amplifier, or negative inductance, or extremely high capacitance. Because such components do not exist in real life, and it does not make sense to use those components.



## Components

### **SPICE:**

- close to “real” parts
- “reasonable” limits

### **NL5:**

- “ideal” components

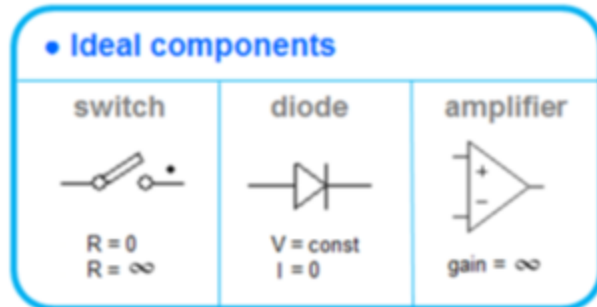
[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



11. NL5 approach is exactly opposite. First, NL5 works with true ideal components. What does it mean?

## Ideal components

- switch/diode with zero/infinite resistance, amplifier with infinite gain



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)

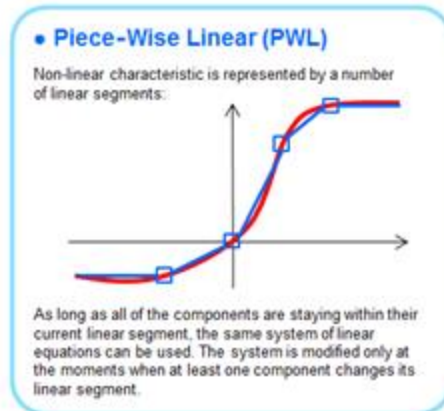


12. True ideal components include:

- switch with zero/infinite resistance
- diode with zero/infinite resistance, or in other words, constant voltage and zero current
- amplifier with infinite gain

## Ideal components

- nonlinear components considered as PWL (piece-wise linear)



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)

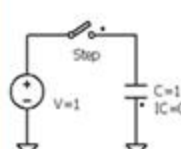


13. Nonlinear components are considered as PWL – piece-wise linear

# Ideal components

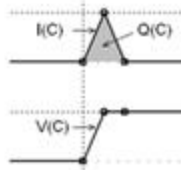
- instantaneous switching

**Instantaneous switching**



An ideal switch in NL5 has instantaneous switching time. NL5 is able to simulate processes where infinitely short current and voltage pulses with infinite amplitude may occur.

Such a pulse is displayed in NL5 as a triangle pulse which exists only at one calculation step, and its area satisfies charge or magnetic flux conservation law.



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



14. Instantaneous switching capability. This is the most problematic part for all simulators, since it may result in infinite voltage or current, and it should be handled properly.



## Components

### SPICE:

- close to “real” parts
- “reasonable” limits

### NL5:

- “ideal” components
- anything allowed

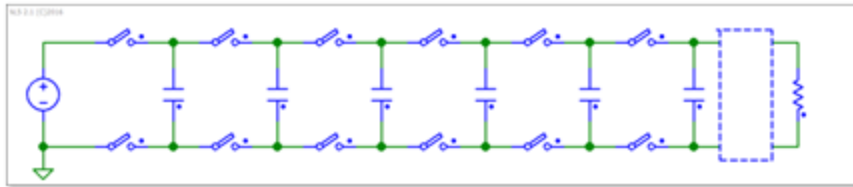
[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



15. Second NL5 specific feature related to components is that practically any component parameters are allowed: zero, infinity, positive, negative – anything. Even if such component does not exist in the real life, it can be used in NL5.

So obviously, NL5 is not a SPICE, since SPICE algorithm cannot handle ideal components and arbitrary parameters.

## Components



Switched capacitors: “charge pump”.

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)

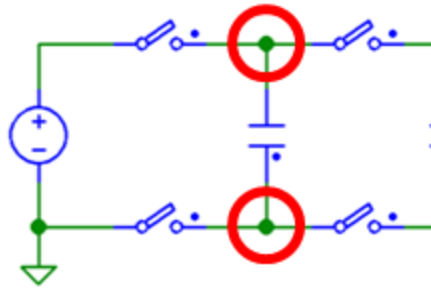


16. This is an example of a very simple circuit: charge pump on switched capacitors. What happens if you use ideal switches in that circuit? A lot of problems.





## Components



Floating nodes and components

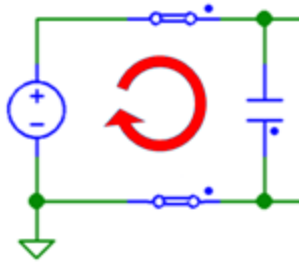
[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



17. Floating nodes (or even floating component)



## Components



Voltage loop

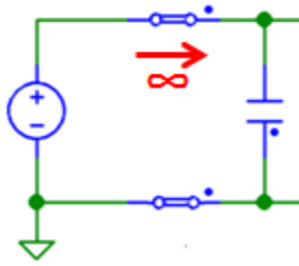
[n15.sidelinesoft.com](http://n15.sidelinesoft.com)



18. Voltage loop (V connected in parallel with C).



## Components




Instantaneous switching: infinite current

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



19. Instantaneous switching will create infinite current.

That circuit is not a big deal for NL5, while other simulators would not like it.



## Models

### **SPICE:**

- as close to “real” part as possible
- may be very complex
- hard or impossible to modify

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)




## 20. Models.

In SPICE simulators, “model” is supposed to be very accurate representation of a “real” component, real part, such as diode, transistor, OpAmp.

Such a model could be very complex with many non-linear components, and many passive components modeling bandwidth and parasitics.

It could be really hard to understand how exactly it works, and practically impossible to modify it. You may ask: why would anybody need to modify the model of standard part? we’ll talk about that later.



## Models

### NL5:

- as simple as possible
- provides basic behavior
- no parasitics
- equivalent circuit with ideal components

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



21. In NL5, models are as simple as possible.

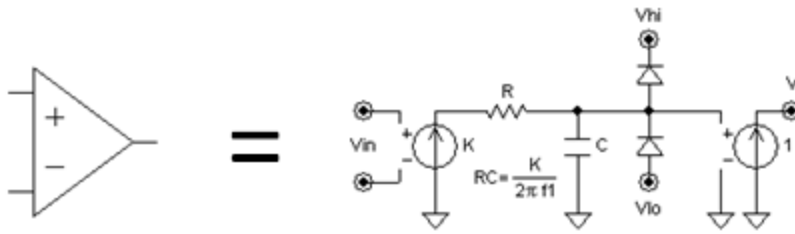
The purpose of the model is to provide just basic, essential behavior of the component.

All specific properties of “real” components, like parasitics, input/output impedance, bandwidth, etc. may not be included into NL5 models.

Most of the models do have a clear and simple equivalent, usually made from ideal components.



# Models



## Differential OpAmp

[n15.sidelinesoft.com](http://n15.sidelinesoft.com)



22. For example, here is equivalent schematic of OpAmp model, so you know exactly what is inside, how it operates, and what to expect from it.

## Algorithm



### 23. Algorithms.

You probably know this person or character: Dr. Gregory House from popular TV show House M.D., and his favorite saying "Everybody lies".

The fact is that all simulation tools lie. LTSpice lies, NL5 lies too. You cannot get rid of that lie completely. What you can do is understand why, where, and when it may occur, and how to minimize it.



## Algorithm: lie #1

### **SPICE:**

- complex numerical algorithms can produce wrong results
- convergence at switching points
- wrong result is hard to detect

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



24. There are two sorts of lie. First, the simulation numerical algorithm lies.

When solving (integrating) differential equations, there is always an error. Usually it depends on simulation step and time constants of the schematic. The error could be estimated, and it could be minimized by using different variation of integration methods.

Situation is much worse with non-linear components. A typical and well known problem is convergence at switching points.

If algorithm is not extremely good and reliable, it could easily produce absolutely wrong results. What is even worse: you may not notice that the result is wrong, unless you know what the right result is.





## Algorithm: lie #1

### NL5:

- simple integration algorithms
- no convergence problems
- results can be easily verified

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)




25. How does NL5 deal with this?

It does not have non-linearities, but it still has integration algorithm. To make it reliable, NL5 is using the simplest second-order trapezoidal integration method. It provides good accuracy, stability, and its behavior is very predictable.

Since switching is instantaneous, integration method is not involved into it, and there is no convergence problem.

If you have any doubts about integration results, just simply looking at data points will allow to discover any possible problems. The math is very simple, everything is transparent, and no surprises expected.



## Algorithm: lie #2

### **SPICE:**

- adds small/large components to make numerical algorithms “happy”
- undocumented features
- may affect circuit operation


[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



26. Lie number two is introduced by simulator itself. Simulator does different tricks to make the numerical algorithm “happy”, and thus avoid the lie number one.

It includes: adding small resistor in series with capacitors or voltage sources, large resistor in parallel with inductors, adding large resistor from some circuit point to the ground, adding small capacitor in parallel with the switch, and much more. If it is done right, it would help numerical algorithm to converge and run smoothly.

What is bad about this lie: most of those tricks are undocumented, or users are not educated enough to be aware of those features. The user may not know what exactly was modified in the schematic, in order to make simulation work, and how it might affect circuit operation. And the effect of such not-authorized simulator “intrusion” could be significant, and sometimes disastrous.



## Algorithm: lie #2

### NL5:

- does not add anything
- with ideal components, no need for very small/large values
- WYSIWYG (“what you see is what you get”)

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



27. In NL5 this does not happen at all. NL5 does not add anything into schematic, because it does not need to.

When ideal components are being used, there will be no very small or very large resistance: instead, there will be zero and infinity, which NL5 can handle easily.

NL5 can easily handle floating nodes, so no resistors to the ground are necessary.

To summarize that, NL5 implements “What You See Is What You Get” principle: nothing is hidden, the result you see belongs exactly to the schematic you entered.



## Algorithm: #3

### SPICE:

- many algorithm related parameters
- set by default for “reasonable” conditions
- not-experienced user may set wrong values

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



28. There is a third algorithm-related problem. Usually, algorithms have some settings, parameters, such as absolute and relative tolerance, minimum conductance, limited rate of change  $dV/dt$ , many of them. By default, those parameters are set up to provide the best performance for “realistic” circuit, at “reasonable” conditions: what is mostly expected from electronics design.

If simulation conditions are out of “reasonable” range, for example very high or very low frequencies, voltages, or currents, those parameters might need to be changed. Unfortunately, not all users are experienced enough to do that right. Changing those parameters without knowledge may easily result in wrong simulation.



## Algorithm: #3

### NL5:

- only simulation step is set up by user

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



29. NL5 does not have algorithm related parameters, except just one: simulation step, so there is nothing to break. The algorithm is supposed to work correctly at any simulation conditions - and it does.



## Applications

- evaluation of design ideas, proof of concept

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



### 30. NL5 applications.

Evaluation of design ideas, proof of concept. This is initial stage of any project, and the goal is to evaluate the principle, basic functionality of the design. “Real” parts and complex models are not needed, ideal components will perfectly do the job.

With NL5, you can iterate design very fast (literally as soon as a new idea comes into your mind), you can get a good understanding of how it is supposed to work, prove its feasibility, discover limitations, and finally get a solution ready for detailed analysis with real parts, using any “real” SPICE simulators.

I want to emphasize again ability of NL5 to simulate any, even “non-realizable” topology, with arbitrary component parameters, at any time/frequency/voltage/current range. In other words, you simulate “schematic that does not exist”, and probably cannot be realized with existing technology.

It may look “crazy” and even “stupid” today, but tomorrow it may become a reality. And you already prepared for it.

## Applications



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



31. This charge pump circuit with ideal switches may be too far from reality: you may set up absolutely “not-realistic” arbitrary capacitance, load, switching diagram. You can run simulation at some “non-realizable” conditions in terms of timing, voltage, current, and NL5 will simulate it. That will give you understanding of its operation, and problems that you may face while implementing this kind of circuit in a real life.



## Applications

- evaluation of design ideas, proof of concept
- analog electronics, with customized complexity

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



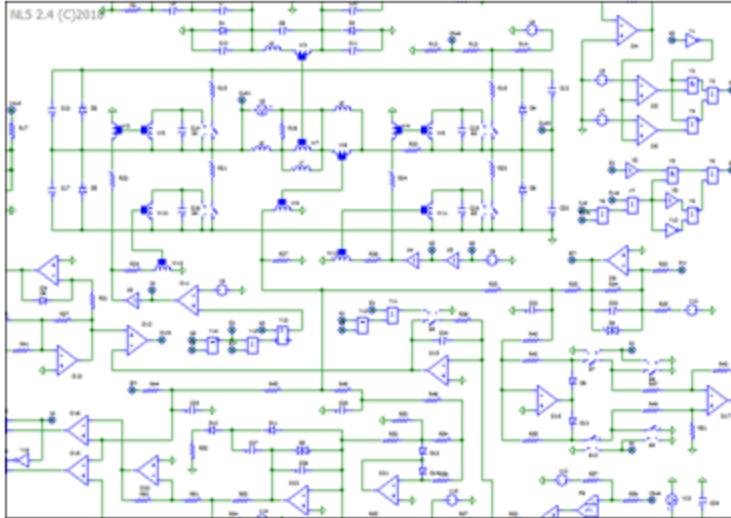
32. Any analog electronics, with custom complexity.

With NL5, you can start simulation with all ideal components, and then add “real properties” step-by-step, one at a time. Doing that you can determine what is critical and what is not, how to deal better with “real” properties, and how to select a right real part for the real design.

Also, the fact is, many engineers do not trust SPICE models provided by manufacturers, especially when the component is being used at the extremes of the spec: for example very high frequency. They prefer to build their own models based on tests, at specific operating conditions, with full understanding how it works. This is exactly what NL5 is capable to do.



## Applications



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



33. This is just a part of a big “ideal” model of a real power supply product. You won’t see any power FETs here: all are modeled by switches and controlled sources. Parasitics are added only where needed. This is very complex circuit with several modes of operation, and NL5 simulation results are very accurate and close to the real test data.



## Applications

- evaluation of design ideas, proof of concept
- analog electronics, with custom complexity
- power electronics, SMPS

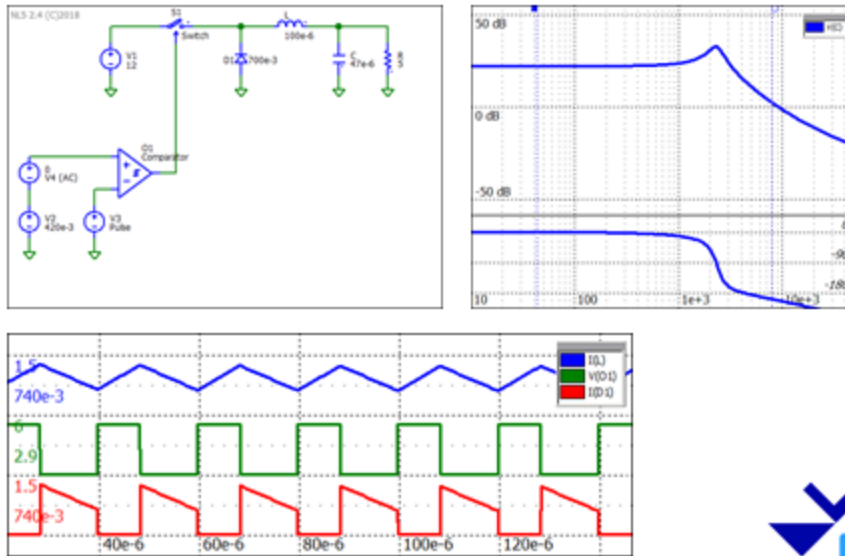
[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



### 34. Power electronics, SMPS.

This is the area where ideal components with instantaneous switching would be a great help. It accelerates simulation, avoids convergence problems, with still reasonable and acceptable results.

# Applications



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



35. Here is example of very simple buck converter, with FET transistor modeled by ideal switch. You can see some transient waveforms, and AC response obtained with Z-transform method.



## Applications

- evaluation of design ideas, proof of concept
- analog electronics, with custom complexity
- power electronics, SMPS
- mixed-signal co-simulation, digital verification

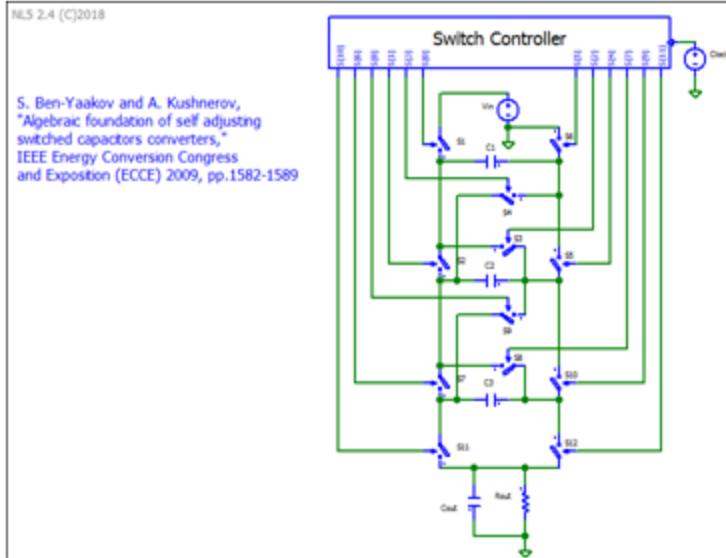
[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



### 36. Mixed-signal co-simulation, digital verification.

When testing mixed-signal digital-analog systems, and the goal is verification of digital part, it would be beneficial to avoid analog simulation problems as much as possible, to concentrate entirely on digital functionality. NL5 can interface with digital counterpart by different ways, and it provides fast and reliable simulation of analog part.

# Applications



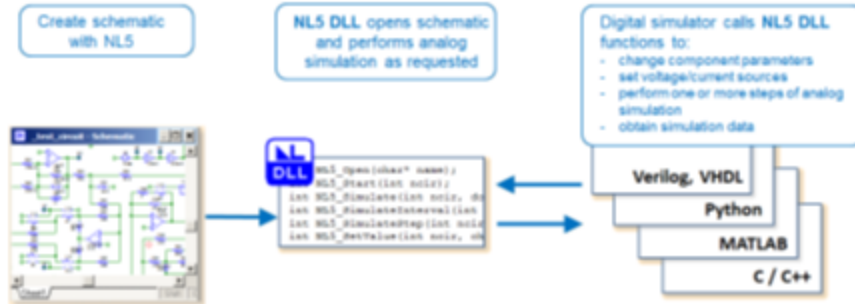
[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



37. Example: “binary capacitors”. The digital “switch controller” generates control signals for ideal switches, and there is a special topology of switched capacitors: it creates requested output voltage from constant voltage source. The controller is a C-code block or DLL, and NL5 executes both digital part of the design, and analog part.

# Applications

Use **NL5 DLL** for co-simulation with digital simulation tools



[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



38. Another example: NL5 DLL is a full-functional NL5 simulation engine. It simulates only analog part of the design.

Digital part is simulated by digital simulation tool, which could be running Verilog or VHDL code, and is calling NL5 DLL to communicate with analog part. This allows design or test engineer to concentrate on digital part, and don't worry about analog simulation, which is going to be fast and reliable thanks to NL5.



## Applications

- system-level design
- education
- any physics discipline described by similar equations (heat, flow, sound, mechanics)

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



39. Just to mention other possible applications: system-level design, education, other physics disciplines (mechanics, sound, heat, flow, etc).



## Some of NL5 features

- 3 types of **AC analysis**: small signal, “sweep AC source”, Z-transform
- subcircuits, C code, and DLL components, **automatic DLL creation** with Microsoft VS
- **script**: C language code to modify parameters, run simulation, and analyze simulation data

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



40. NL5 features: Very quick, just to give an idea:

- 3 types of AC analysis
- automatic DLL creation
- Script





## Some of NL5 features

- **HTTP interface:** built-in HTTP server, capable to receive and execute script commands from popular external tools, such as Matlab, Python, or any custom applications
- **Oscilloscope interface:** control the scope and read back scope traces through NI-VISA interface

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



41.

- HTTP interface
- Oscilloscope interface



## Simple yet powerful

### simple

using simple ideal components

easy to use: 5 minutes to get started and obtain first results

just one small executable file nl5.exe is needed (<6MB), no installation hassle

### powerful

add complexity as needed, get very accurate simulation of real circuits

many advanced features, listed in 600 pages Manual

can be easily integrated with external tools

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



42. And finally, to characterize NL5 in two words, I would say “simple yet powerful”.

Simple: it is using simple ideal components.

Powerful: you can add complexity as needed, get very accurate simulation of real circuits.

Simple: easy to use: 5 minutes to start and get first results.

Powerful: many advanced features, listed in 600 pages Manual.

Simple: just one small executable file nl5.exe is needed (<6MB size), no installation hassle.

Powerful: can be easily integrated with external tools.



Edsger W. Dijkstra  
Computer scientist

*“Simplicity is a great virtue  
but it requires hard work to achieve it  
and education to appreciate it.*

*And to make matters worse:  
complexity sells better”*

[nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)



43. And The last quote today from well-known Dutch computer scientists Edsger Dijkstra:

“Simplicity is a great virtue but it requires hard work to achieve it and education to appreciate it. And to make matters worse: complexity sells better”.