



Valentin-Ioan Maranescu, Catalin Căleanu, Mocoșan Mugur

Stimulus Generator for Spice Simulations Using Graphical Programming Language

*The paper presents a virtual instrument designed in graphical programming language LabView. This virtual instrument is a spice stimulus generator used for mixed analog and digital circuit simulations. Innovatory for this tool is the time scaling and rescaling possibility also for digital stimulus not just for analog stimulus. Transition level editing is avoided, sequences in stimulus being generated byte by byte. Digital sequences can be synchronized with analog signals if needed. Editing the stimulus is faster just filling a table with hex codes. The stimulus can be generated serially, and/or parallel on 1, 2, 3 or 4 bytes. For implementation, spice stimulus format *.stl under Orcad 10.3 was analyzed.*

1. Introduction

Integrated circuits design with mixed analog and digital circuits involve complex simulations, where we need both digital values and analog signals. Usually all design and simulations tools have a stimulus editor to help user to built complex shape signals. These kinds of complex signals are needed when a digital core is simulated in the same time with analog part. The simulations will take long time, and user want to be sure that stimulus file is correct edited from the first simulation. A small numbers of errors in stimulus can easy take hours or days to correct them. Manual edit will take long time to build the stimulus file, without having confidence that everything was correct.

The possibility to corrupt stimulus appears also when we want to change the signal, to scale in time without changing the shape of the signals or to insert something in signals. In addition, almost all the times we want to change back the signals without to edit again all the modifications or to save for each step a new stimulus file.

2. Stimulus format

Stimulus files have a typical format for each tool, but it is easy to deduce the correlation between stimulus and corresponding text file. Orcad .stl files were analyzed. Basically these files have the following format, where a line with first character * is a comment line, ignored by simulator, and text with *italic* is a variable:

```
*File Header: name and file path, time stamp.
;!Stimulus Get
;! DSignal1 Digital ...SignalN Digital ASignal1 Analog ASignalM Analog
;!OK
*Editor View settings (example):
;!Plot Axis_Settings
;!Xrange 0s 5us
;!Yrange 0V 10V
;!ManualUniverse
;!Xuniverse 1000ns
;!Yuniverse -1V 10V
;!XminRes 0.1ns
;!YminRes 50mV
;!Ok

.STIMULUS DSignal1 STIM (1, 1)
+ 0s 0
+ transition times, transition bit(s) value(s)
+ .....

.STIMULUS DSignalN STIM (8, 11111111)
+ 0s 00000000
+ transition times, transition bit(s) value(s)
+ .....

.STIMULUS ASignal1 PWL
+ TIME_SCALE_FACTOR = 1
+ VALUE_SCALE_FACTOR = 1.00
+ (0, 0)
+ transition times, transition values
.....
```

Watching to any digital stimulus header versus to an analog stimulus header we will observe that time scaling is possible just for analog stimulus.

This means that if we want to keep a correlation between all signals we have to manually rescale signals changing every transition time.

In stimulus used to simulate mixed circuits, we can have thousand of transitions/signal, and even watching at the signals can be a difficult thing. The solution to rescale the entire stimulus in time is to recalculate and edit all time transitions in every stimul. This can be done easy if the stimulus is generated not edited. This paper proposes a tool that will generate the stimulus file.

3. Stimulus Generator - Virtual Instrument - CAD tool

Computer Aided Design tools are built to help designing and simulating schematics.

Instead of spending hours and days with editing manual complex signals, we build a virtual instrument (VI) using LabView graphical programming language. After we have identified the format of the stimulus file and all the rules, we captured them in this VI. Because of modular realization with subVI's the stimulus generator can be easy modified or enhanced to generate signals for any application. The stimulus generator is provided with a front panel (figure 1) where the most important parameters of the signals can be edited.



Figure 1. Stimulus generator – front panel.

The most important feature is time scaling for both stimulus types analog and digital. It is also possible to scale with a coefficient just some periods from signals, just checking some buttons.

Front panel aloud to edit the name of the signals, common parameters of the signals, i.e. time step or clock period, time unit, individual amplitude scalar for each analog signal. For digital signals, hex mode using bytes can be used for description of stimulus, suitable for testing serial/parallel interfaces. It is very easy to insert bytes in one signal, and maintain correlation with rest of the signals without having a skew.

In Figure 2. we show generated signals with our tool that takes less than 2 minutes of editing in front panel, which using Stimulus editor from Orcad usually will take more than 30 minutes. From this short example we can realize that our tool can compress weeks of work in one day.

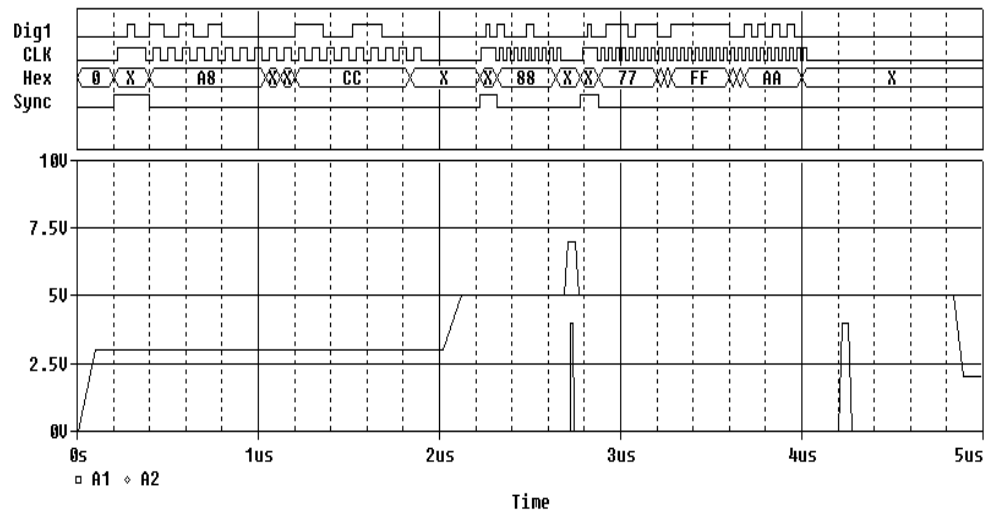


Figure 2. Example of stimulus file – generated by virtual instrument.

4. Conclusion

The stimulus generator virtual instrument is suitable for application where user need to simulate a circuit that need different complex stimulus. Is known that manual work can be source of many errors. Having an automated tool with the rules implemented, stimulus will be always generated without errors. This instrument was successful used for testing mixed analog and digital circuits. A digital

core with its instruction set was simulated in the same time with analog part using stimulus generated by this tool. Estimating the stimulus manual editing time for all the simulations, and designing time of the automatic stimulus generator, results in saving days of work. Having this stimulus generator, designer can rapidly edit/change complex stimulus, remaining focused on his design. Stimulus generator can be easily modified in the future for similar applications.

References

- [1]***** *PSpice User's Guide*, Product Version from Orcad10.3
- [2]***** *LabView Help*, March 2004 edition
- [3]Andrei Vladimirescu, *Spice*, Editura Tehnică, București 1999.

Addresses:

- Assist. Drd. Valentin-Ioan Maranescu, "Politehnica" University of Timișoara, Bd. V.Pârvan, nr. 2, Timișoara, 300223, valentin.maranescu@etc.utt.ro
- Lecturer. Dr. Catalin Căleanu, "Politehnica" University of Timișoara, Bd. V.Pârvan, nr. 2, Timișoara, 300223, catalin.căleanu@etc.utt.ro
- Lecturer. Dr. Mugur Mocofan, "Politehnica" University of Timișoara, Bd. V.Pârvan, nr. 2, Timișoara, 300223.