Guide to Automating PWL Source Generation

A Cadence EDA Tools Help Document

Created by Casey Wallace, Spring 2006
Edited by Patrick O'Hara, Spring 2013

This document explains how to create piecewise-linear voltage waveforms in *spectre*-syntax by using a special program that automates source generation based on a table of input values to be simulated.

1. Create a text file containing the names of your input pins. The names should be separated by a single space, and should all be placed on a single line. For instance, if your input pins are named A and B, your text file would look like:

   ```
   A B
   ```

2. In the same text file containing the input bit vectors you want to simulate. Every row corresponds to a new input, and every column corresponds to a specific input pin, and needs to appear in the same order used in the previous step. For instance, if you want to test a 10 → 01 → 11 set of transitions for pins A and B, respectively, your text file would be:

   ```
   10
   01
   11
   ```

   Save this file to a name of your choice, e.g., *nand_pins.txt*.

3. In the same directory you used to create the text file, type the following command:

   ```
   python /egr/courses/unix/ECE/410/Shared/pwlgen.py <name of vector
   file> <width of pulse> <rise/fall time> <delay>
   <supply voltage>
   ```

<name of vector file>	is the name of the text file in step 1 and 2
<width of pulse>	is the width of the pulse between transitions in nanoseconds
<rise/fall time>	is the rise/fall time of the transition in nanoseconds
<delay>	is the amount of time to delay all the signals
<supply voltage>	is the value of vdd and the input voltage corresponding to logic '1'

   For example, if your pin vector file is named *nand_pins.txt* and you want to test transitions every 10ns where the input transition time (rise and fall) is 50ps with a supply voltage of 3V, you would type:

   ```
   python /egr/courses/unix/ECE/410/Shared/pwlgen.py
   nand_pins.txt 10 0.05 0 3
   ```

   The spectre-formatted PWL voltage sources will be printed to the screen. Include this text in the body of your stimulus file. For the example given here, the output would look like this:

```
V1 (A 0) vsource type=pwl wave=\[
+ 0n 3 9.95n 3
+ 10n 0 19.95n 0
+ 20n 3 29.95n 3
+ ]
V2 (B 0) vsource type=pwl wave=\[
+ 0n 0 9.95n 0
+ 10n 3 29.95n 3
+ ]
```

4. If you want to save the text output of this program to a text file, use the redirection symbol '>', followed by the name of the output file.

```
python /egr/courses/unix/ECE/410/Shared/pwlgen.py
nand_pins.txt 10 0.05 0 3 > name_of_file.txt
```

5. The output of the program can be used to start a new stimulus file. However, you will still need to manually add all other appropriate voltage sources, output capacitances, and simulator commands.

Notes:

1. The PWL generation program was created by an ECE410 TA and is not part of the Cadence software. It is meant as a simple tool for the student to use for generating stimulus files for a specific lab assignment. It is not guaranteed to work for other applications nor provide any functionality beyond what is presented in this guide. However, you may use it for any purpose as long as you understand how it works and the output that the program provides.
2. If you understand the Python scripting language, feel free to copy and modify the program for your own use.