How to use vcdParser

Nicolas BONFANTE

June 16, 2015

Contents

Part I
Prepare animation

1 In your VHDL files

The only think you need to do for this kind of files is to note somewhere (in your head or on a paper) the name and the scope of the variables you want to animate. I found it harder to do this with the VHDL files, so I strongly recommend you to do this with VCD files. (VHDL is too much High-Level).

2 In your VCD files

The only think you need to do for this kind of files is to note somewhere (in your head or on a paper) the name and the scope of the variables you want to animate. Most of cases, it corresponds to the VHDL name in lower case.

3 In your SVG files

vcdParser will only take into account the part of the file which is between `<g></g>` tags. The inner part (id est the part between opening and closing tags) follows a very strict architecture describe in the following.

3.1 1 bit variable

The inner part MUST contain a `path` and a `text`.

- The `path` contains all the attributs you want. But in fact it is useless to fill them because I will overwrite them.
- The `text` contains only a `tspan`. Same remarks concerning to the attribut. The data in the `tspan` follows a specific format describes just below.

The format for the `tspan` data is the following :

`$<drawing_name>:<vcd_name>$`

drawing_name is the name that will be displayed on the drawing and *vcd_name* is the name of the variable in the VCD file. Most of the cases, it corresponds to the VHDL name in lower case.

3.2 n bits variable

The inner `part` MUST contain a `path` and 2 `text`.

- The `path` contains all the attributs you want. But in fact it is useless to fill them because I will overwrite them.
- The first `text` contains only a `tspan`. Same remarks concerning to the attribut. The data in the `tspan` follows a specific format describes just below.
- The second `text` has the same structure as the first one. But the data is just "#value#".

The format for the `tspan` data is the following :

```
$<drawing_name>:<vcd_name>$
```

drawing_name is the name that will be displayed on the drawing and *vcd_name* is the scope plus the name of the variable in the VCD file. Most of the cases, it corresponds to the VHDL name in lower case. For example, if you want to animate the variable called "ce" in the scope "proc.mem", the *vcd_name* will be "proc.mem.ce".

4 Example

You can find an example of such files on github.

Part II
Prepare animation

5 1 bit variables

There is true possible values for a 1-bit variable : 0,1 or U. On the drawing, 0 is represented by the color red, 1 by green and U by green.

6 n bits variables