A whole new world – SPICE

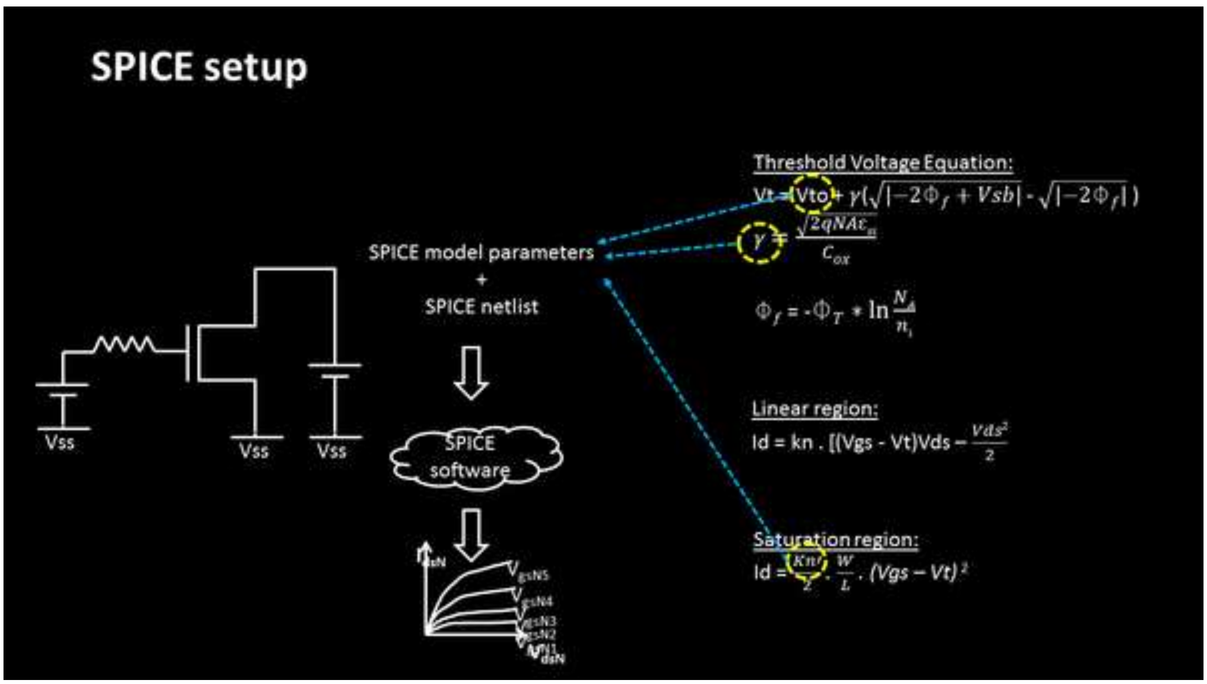
Kunal Ghosh

This was exactly the feeling when I was told to do SPICE …. but you know what, after practically applying the concepts on real design, I felt, it wasn't a new world. It was an extension towards accuracy, that guarantees your chip will not fail, in timing.

And, there it all started. I will be announcing my 5th course – **"VLSI Academy – Circuit Design and SPICE simulations"**, along with my existing 4 courses on Udemy, **which covers right from Physical Design flow to clock tree synthesis to crosstalk to parasitic to timing.**

SPICE comes exactly after these topics. ***Please follow this link to get details of my existing courses***

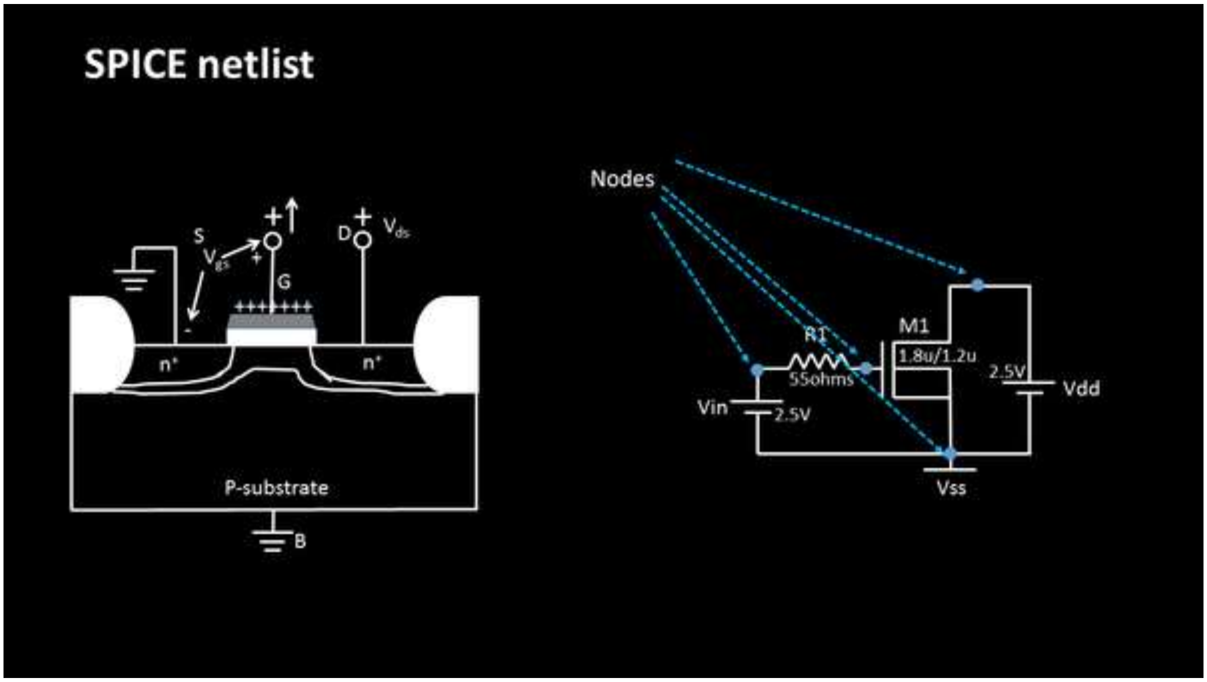
So, let me brief you what exactly we are going to do in SPICE. To start with, below image clearly shows the basic flow of SPICE



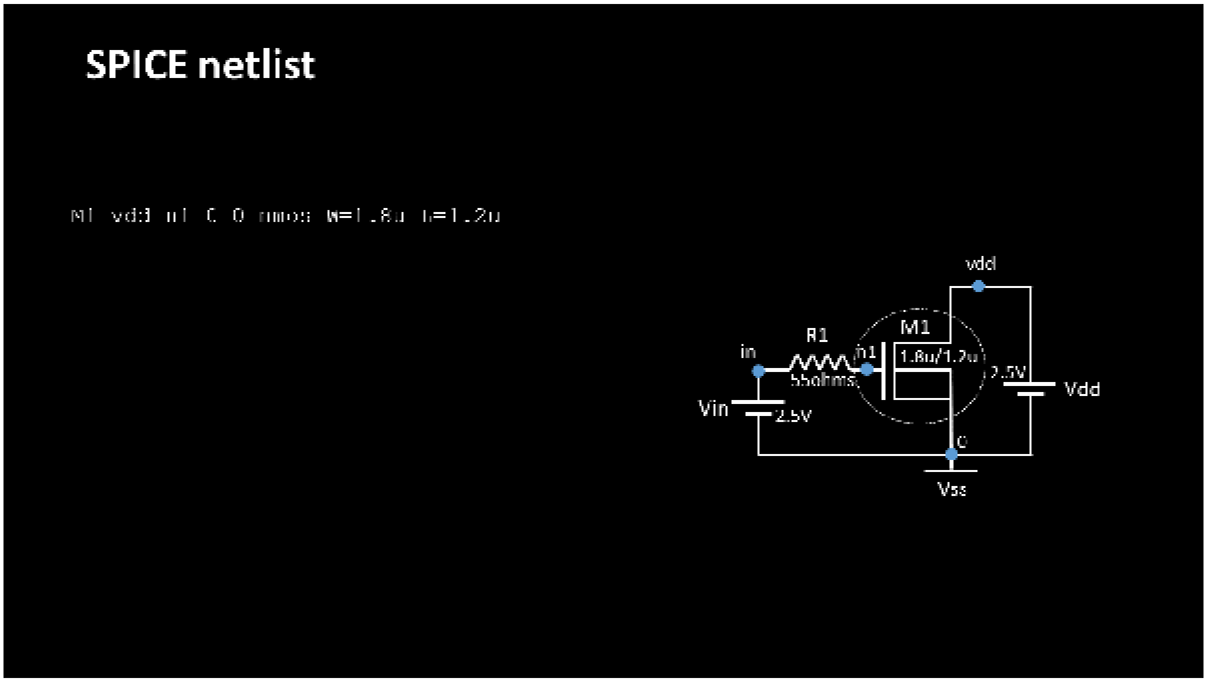
The complex equations on the **right side of the image, are not so complex**, once I go through them in my way in the courses. These equations model a MOSFET, the **left had side is one basic SPICE netlist** (In the course, we will be considering more complex one's, but I will make sure, you go through the basic one's first :))

When we provide both to SPICE engine, the output are some current waveforms (which eventually converts to a timing waveform, I will show you 'how' in the course, when I publish)

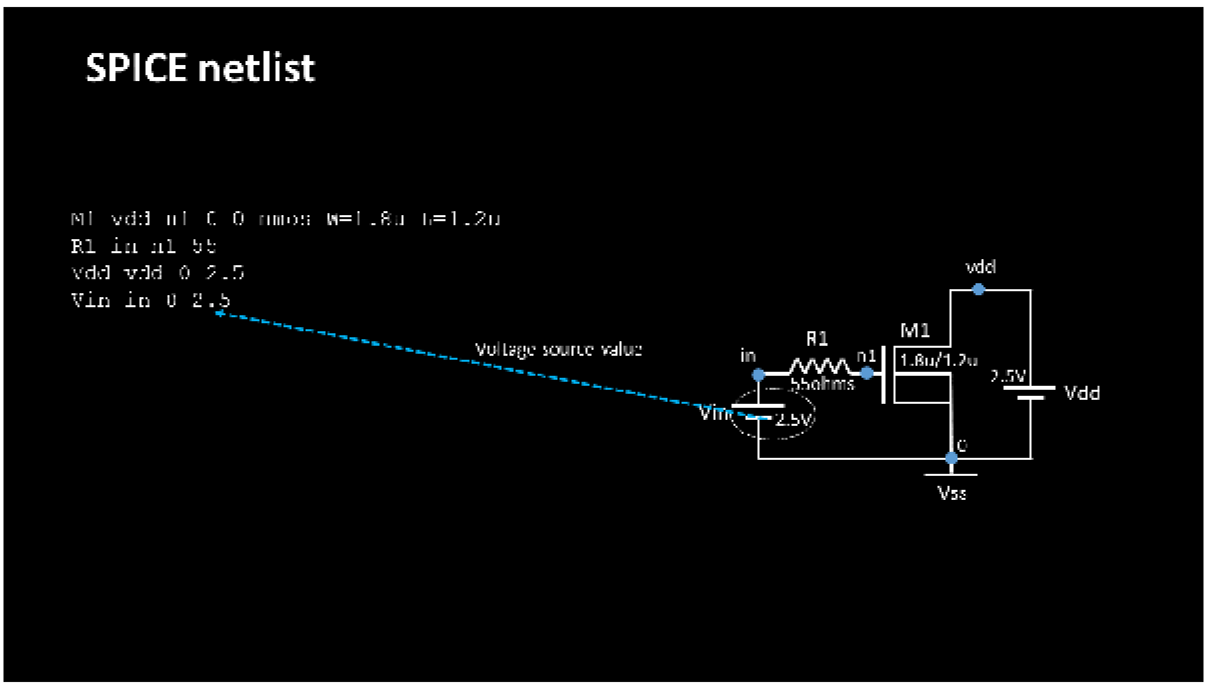
The below MOSFET device, with potential applied to drain terminal and source/substrate being grounded, can be represented by a circuit shown in right side of image. **The nodes help us to create the SPICE deck or SPICE netlist.** Stay with me to see 'how'



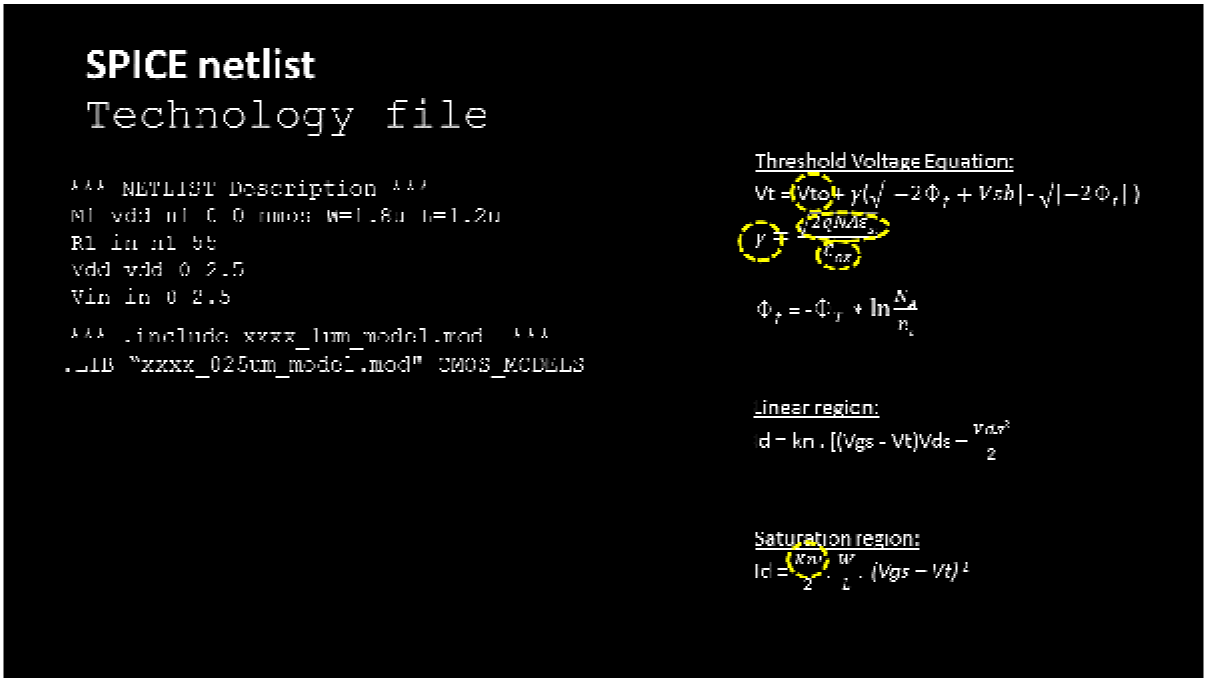
Let's write the SPICE deck for below MOSFET. The name is M1. The nodes are Vdd, n1, 0 (follow the blue dots in the image) and the syntax is "**mosfet_name drain gate source substrate** ". When we write M1 Vdd n1 0 0, it means drain is connected node Vdd (blue dot), gate is connected to node n1, source and substrate is connected to node '0'. (since its ground, I named it '0')



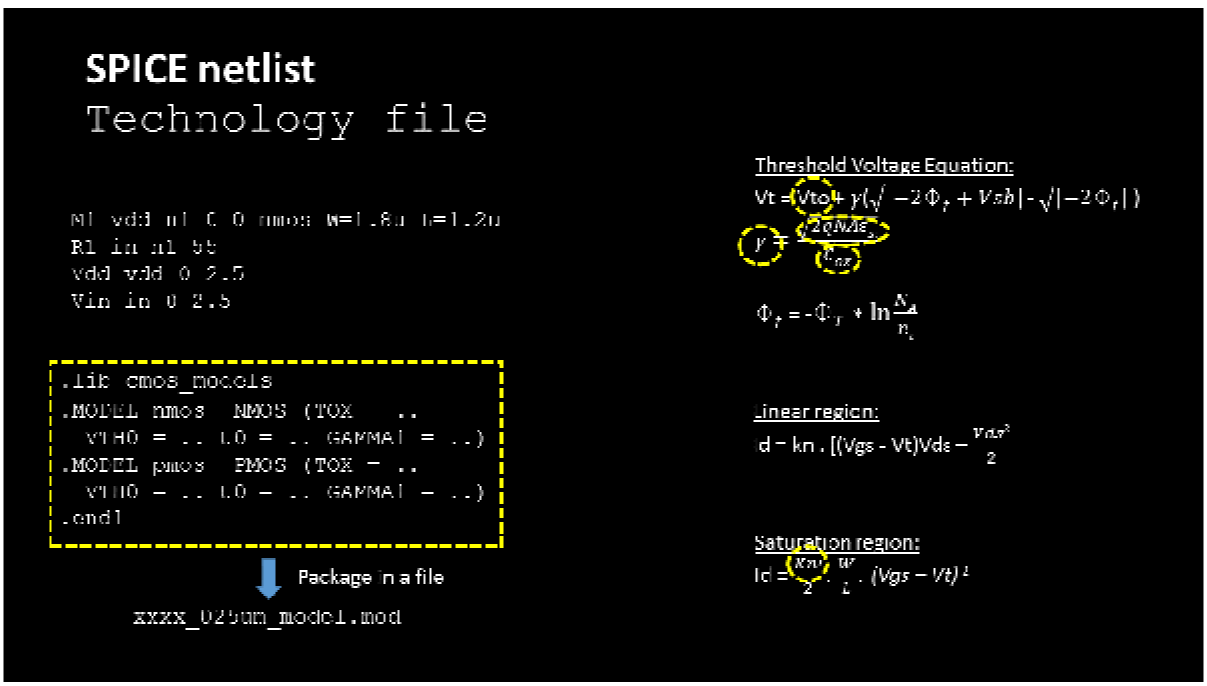
And, we do similar thing for resistor R1, input supply Vin, and main supply Vdd, like below



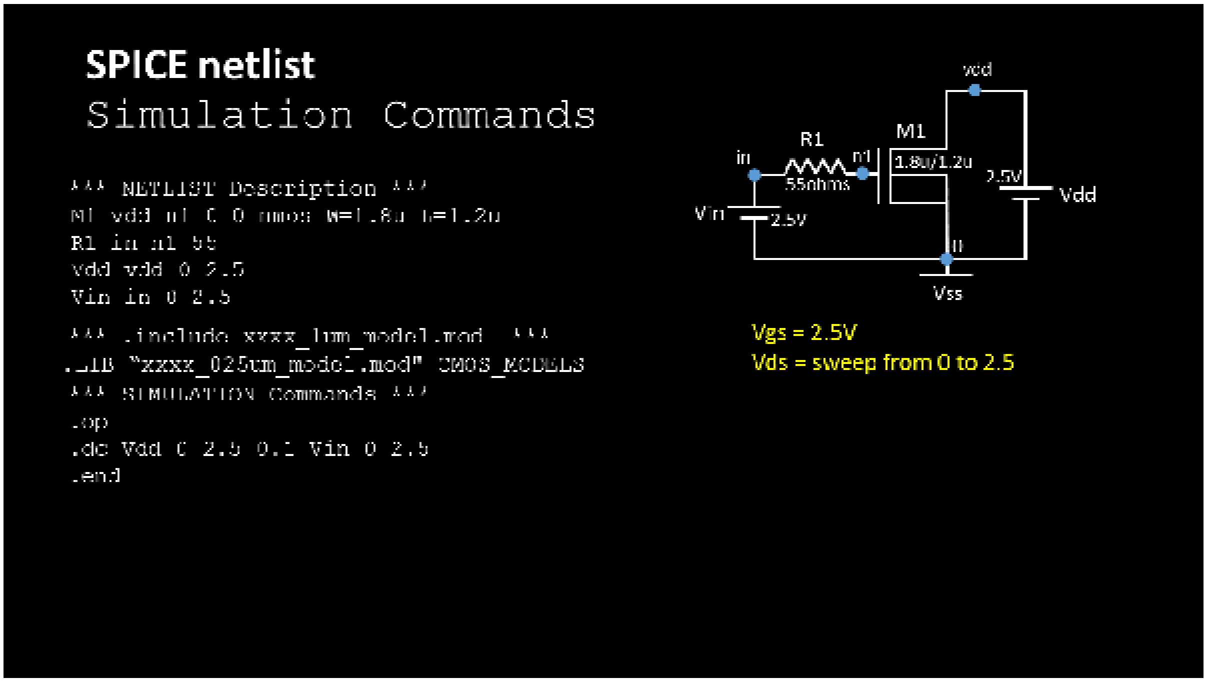
Netlist is ready. Now, we need **to supply the model parameters**, basically, the one's encircled in yellow. Once we give that, **SPICE engine internally has the equations** or **models that evaluates to give the output current values**. The list is huge. Maybe, I will introduce them to you in the course



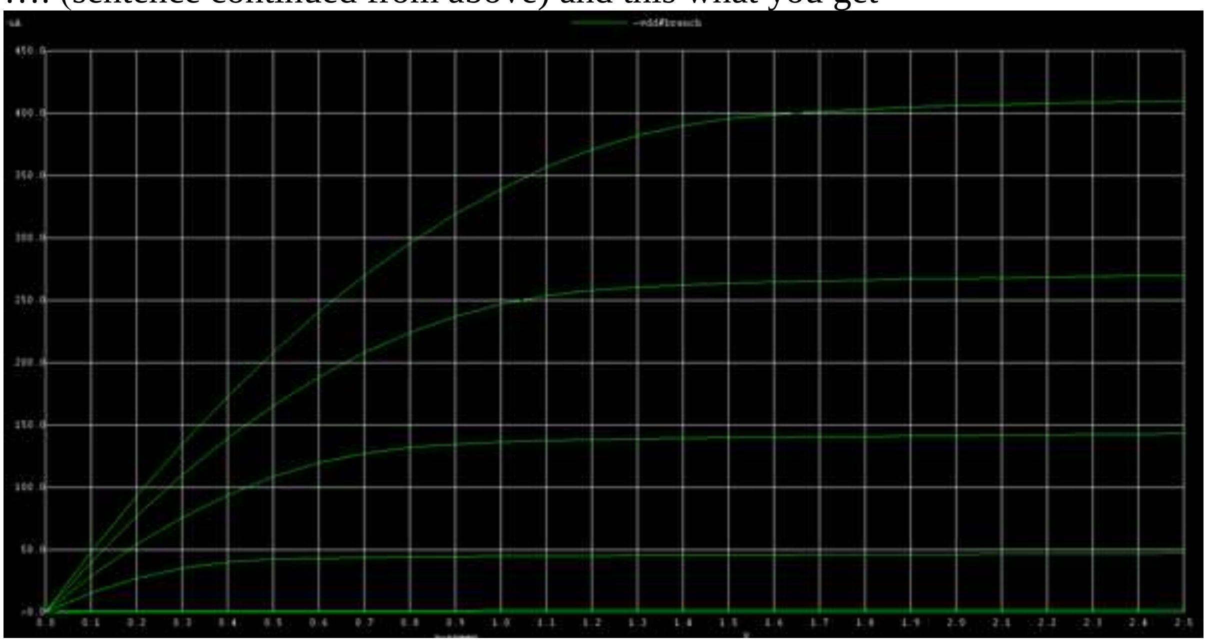
Once, you have them, **pack all the technology related things, into a file, say, .mod file and point to that file, something like below.** This helps maintenance of the files, as, we write SPICE netlist for complex gates.



Now, we provide **circuit + technology file to SPICE engine along with below simulation commands** …. (sentence continued after this image)



…. (sentence continued from above) and this what you get

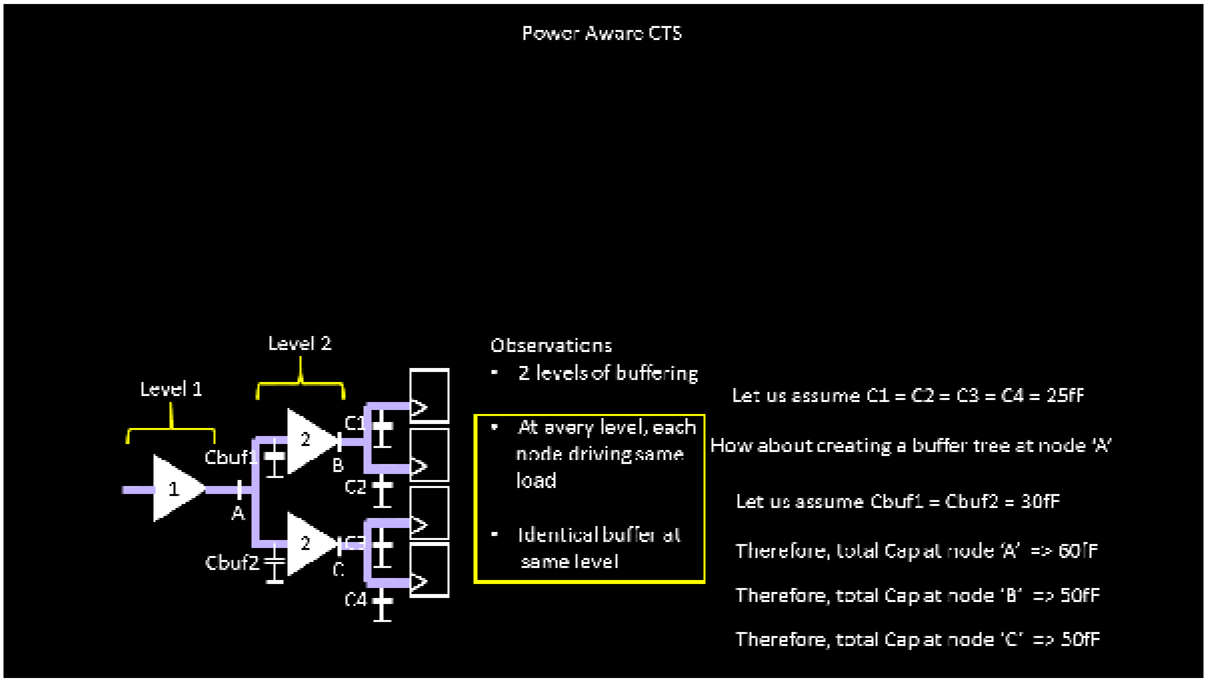


Does that interest you? :). If yes, I will send you the link to my new course very soon. Be ready for another adventure.

I was overwhelmed by the very common question being asked to me after my previous post, and these were from all levels of people (even people from verification and embedded background). "*Why do we need to learn SPICE?* "

And, that was exactly what I was expecting. "For the things, *we must learn before we* can do *them*, *we learn* by *doing them"*. Though, this will be discussed in detail in my **upcoming course of SPICE**, let me try to give a glimpse of it here itself

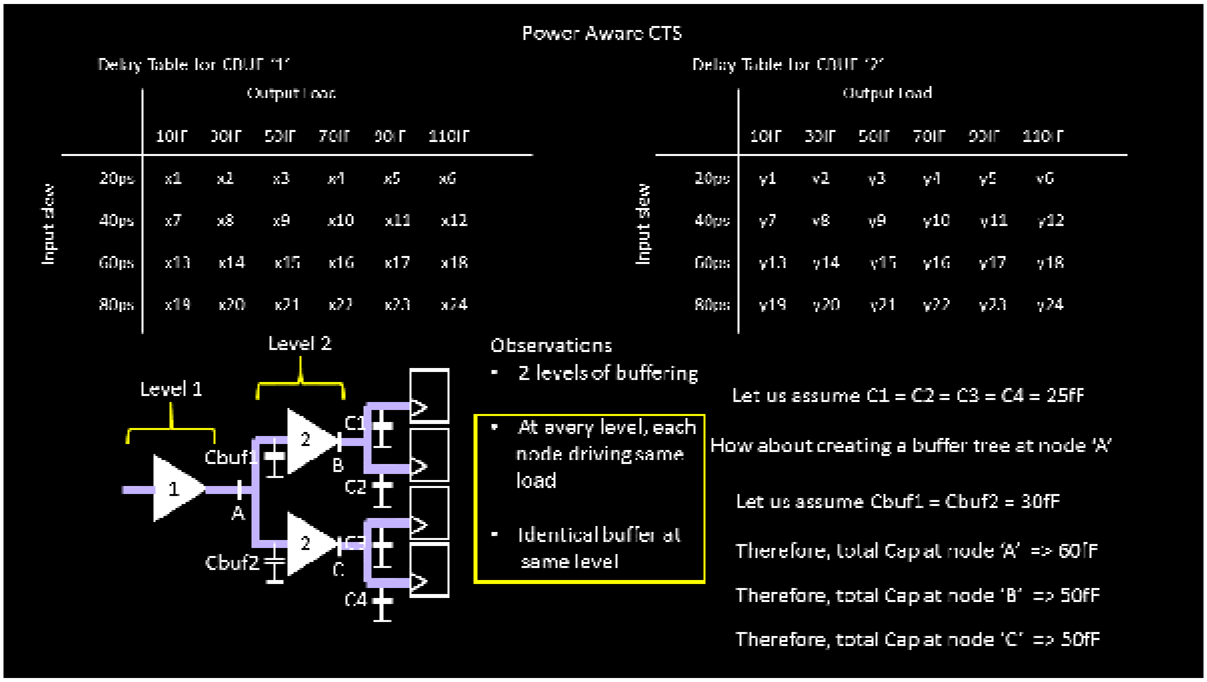
Below is a snippet from my existing course on ***Clock Tree Synthesis***



It has 2 kinds of clock buffers, say '1' and '2'. Now, to calculate the delay of these buffers, the only input, we have been delay models. **These could be "non-linear delay models" or "constant-current source models".**

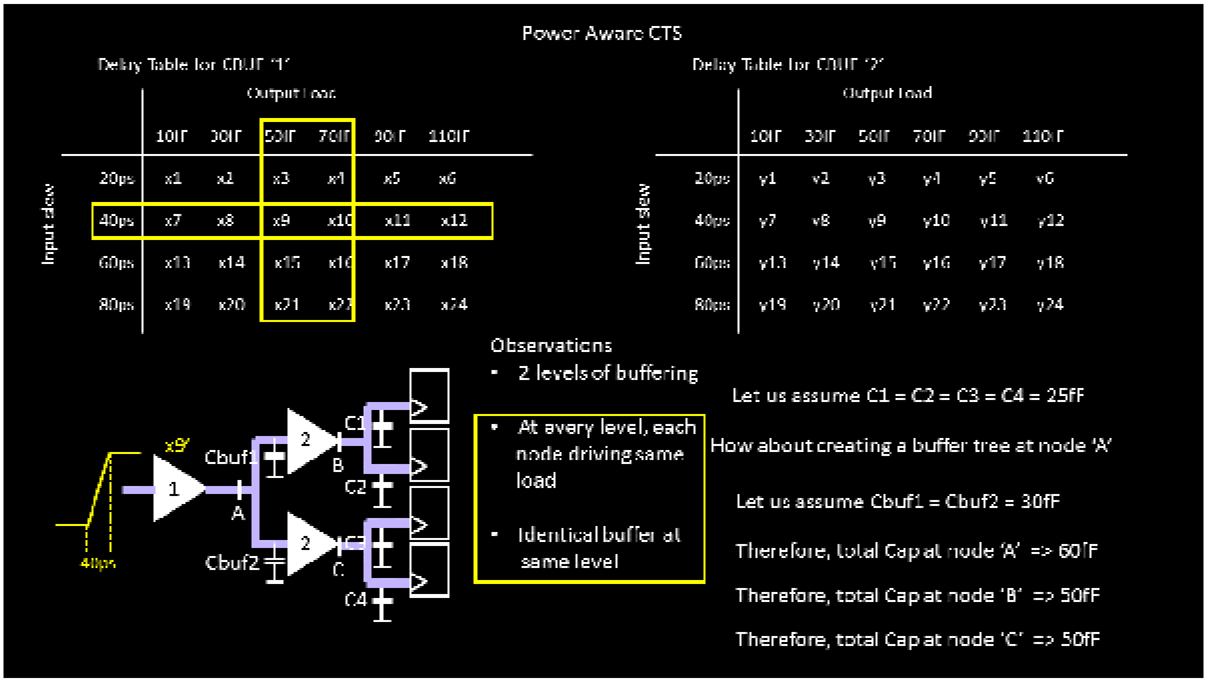
Let's take a simple example of below "**non-linear delay model**." This is how it looks. It has a "**input slew**" one side and output capacitance on other side.

A more complex model (to be discussed separately), and time units on one side and normalized voltage on other side.

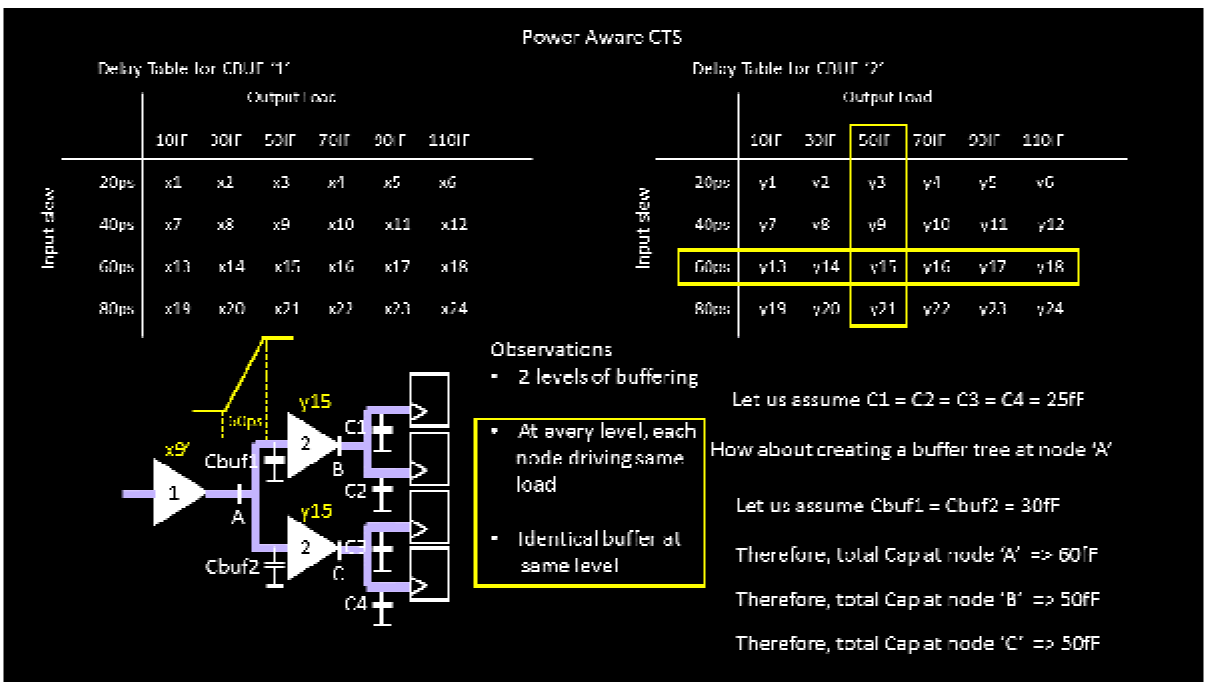


Both buffers can be of different type (or different sizes and drive strength), so each buffer '1' and '2' has their own table.

Now, say, output capacitance of buffer '1' is 60fF and input slew is 'say' 40ps, below is the section of the table that we need to consider computing its delay. **This technique is called 'interpolation'.** We need to interpolate the delay value between 50fF and 70fF, for an output cap of 60fF

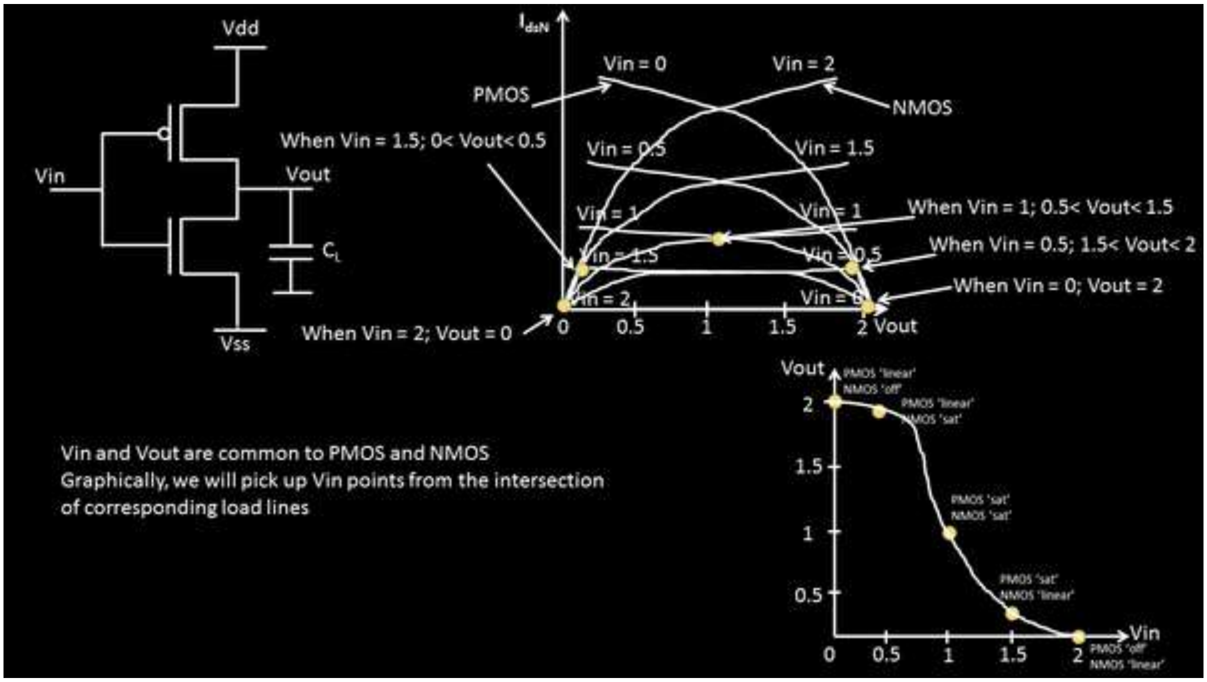


Similarly, for an **input slew of 'say' 60ps and output load of '50fF'**, the delay value of buffer '2' will be about 'y15', as per below table



I haven't yet answered your question. **Why do we need to learn SPICE**? I will answer it now. **Where, do you think, the delay values in the table come from?**

Let’s say we have this below inverter (simple buffer is 2 inverters connected back to back), with the below IO characteristics.



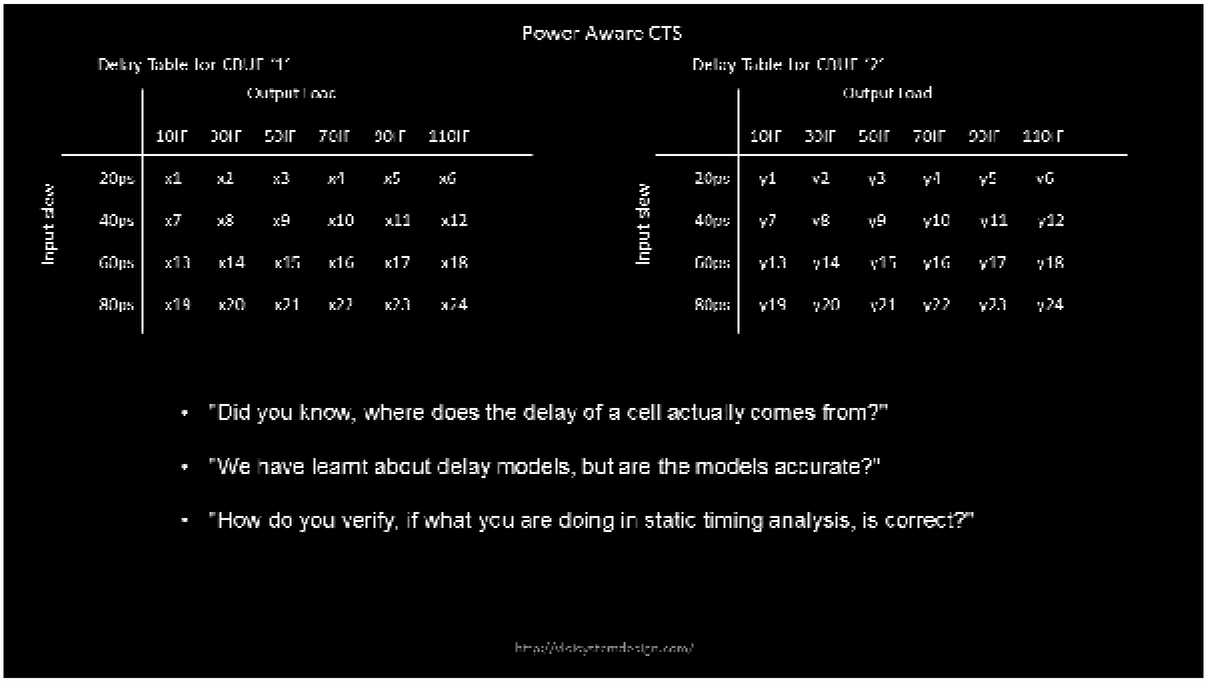
Now, observe very carefully to all above waveforms (specially the last one present in bottom right). Do they remind you of look-alike waveforms, for which we just calculated the delay?

Does the waveform at the top-middle, remind you of the waveforms we derived in last post using a **SPICE simulator**? Do they look like **Id-Vds curves for NMOS**?

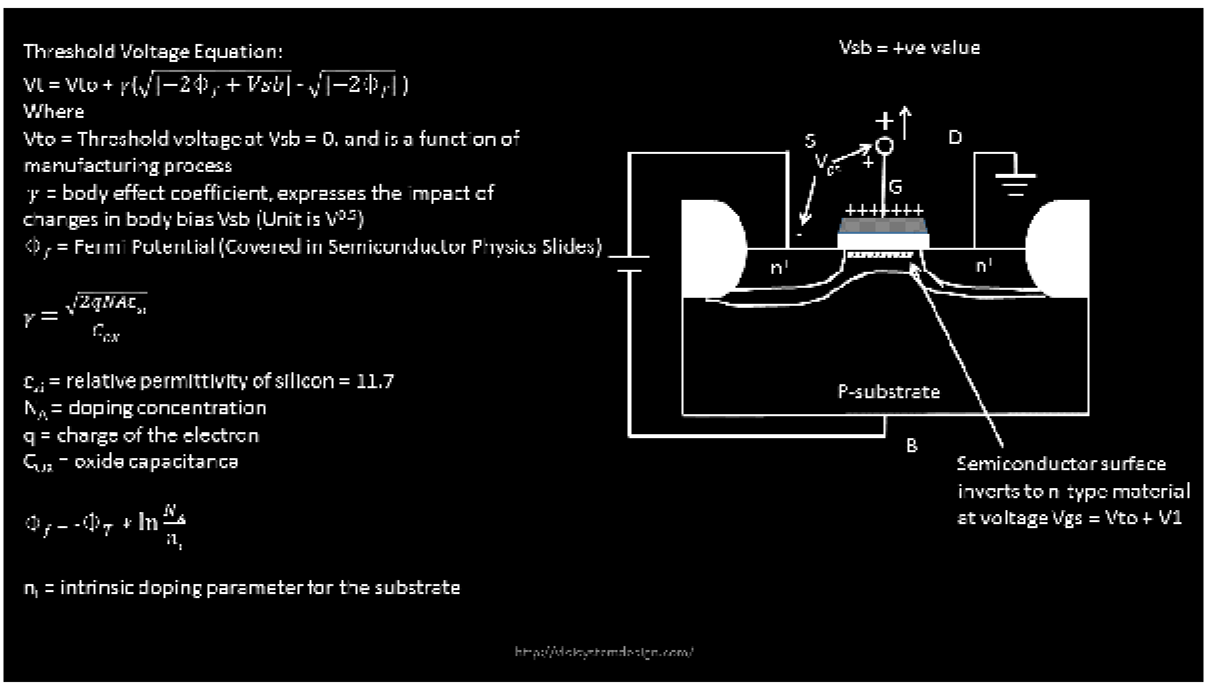
Do you relate how delay of cells are derived from NMOS-PMOS-CMOS transfer characteristics?

I feel, learning or working in VLSI domain without transistor model knowledge is like flying a helicopter with low fuel in a deserted region, not knowing where the hell you will land up.

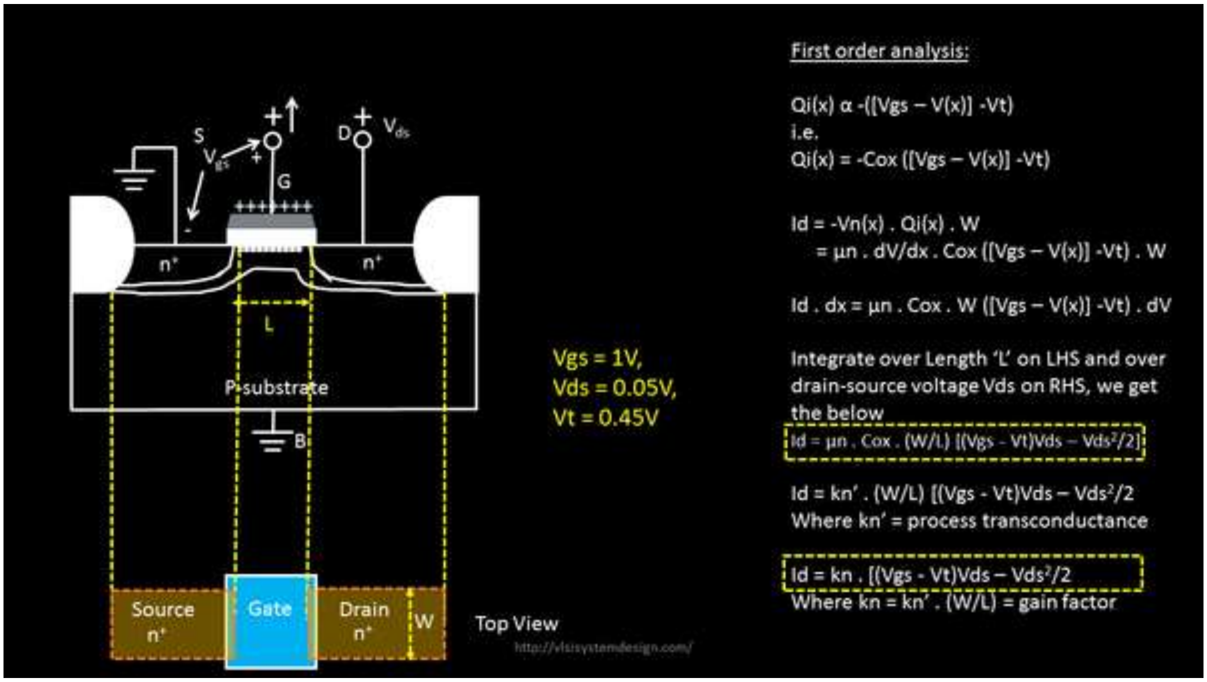
Here how the course goes. I start with asking few questions, on why do we need SPICE

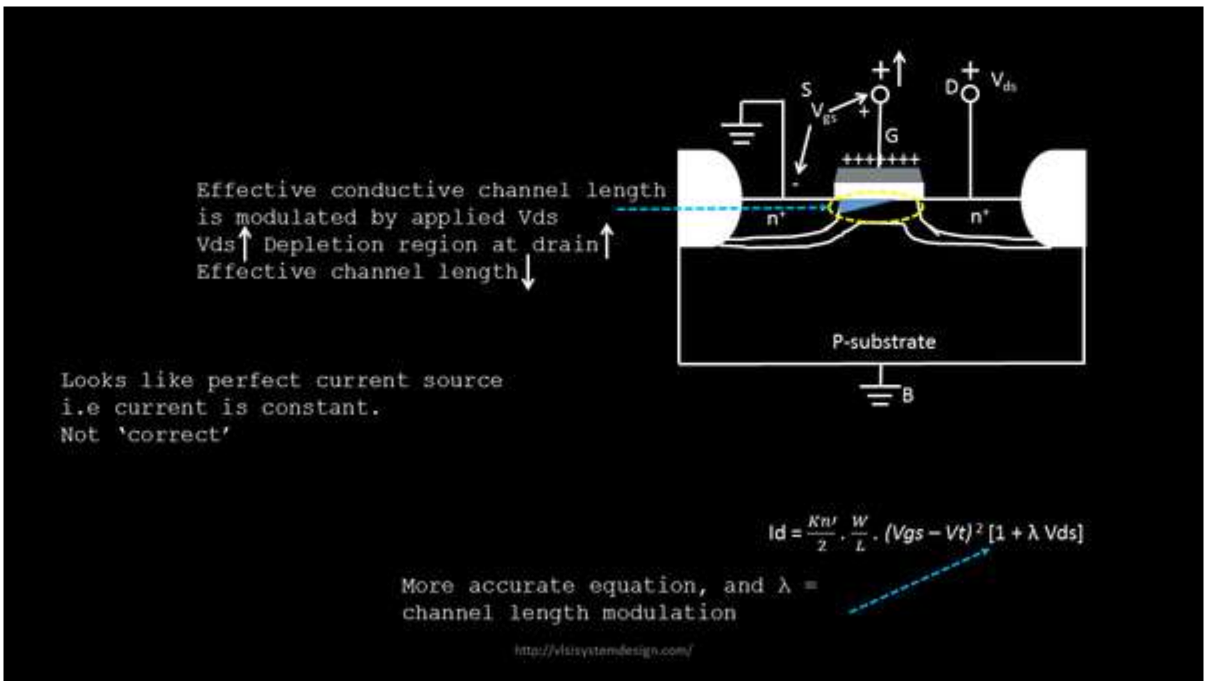


Then, I introduce and (literally) derive models for threshold voltage and …. (sentence continued after below image)

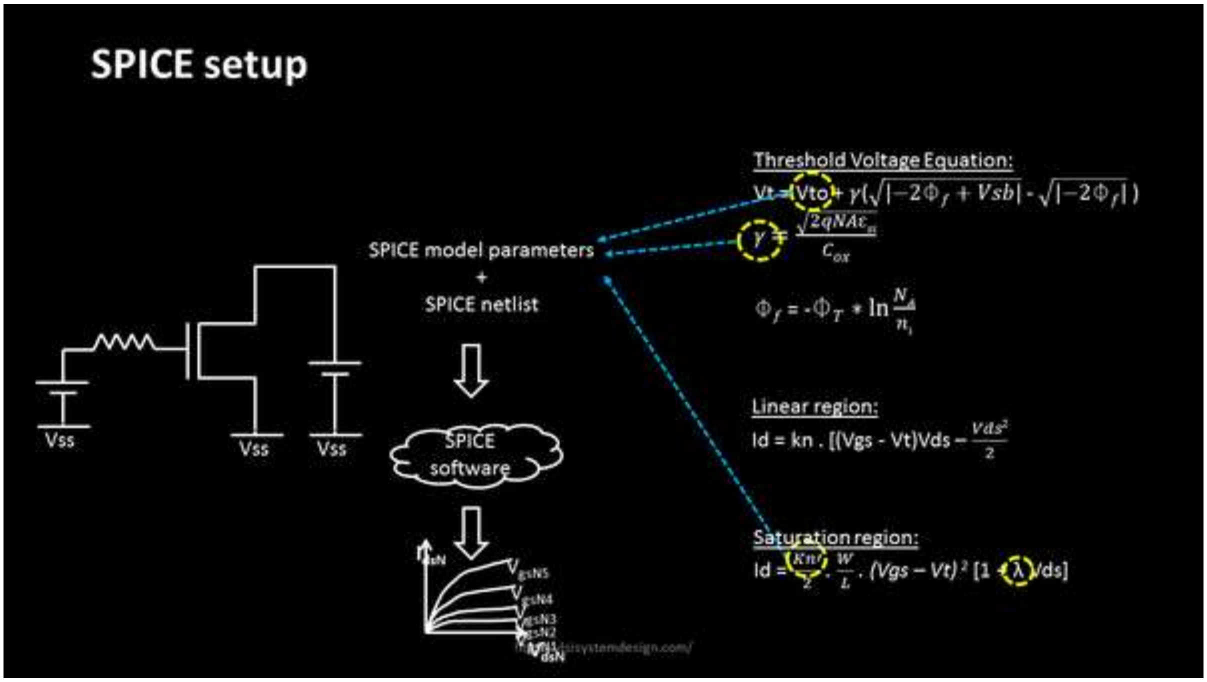


…… (sentence continued from above) drain current models for resistive and saturation region of operation





Eventually, I end up, developing the SPICE setup, and from there on-wards, there is no stopping us from reaching the final goal, where we derive delays for cells.



There's difference in **'Online SPICE simulat*__or'__*** and '**Online SPICE simulati*__ons__*** '. I can happily say that, I was able to achieve the latter, through my **online videos on Circuit design**.

Let me think, and get back on how do I do the 'former'.

Don't you think, it would be great, that while you learn basics on **Circuit Design & SPICE simulations**, parallel you should even be able to simulate it 'Online', without needing to download any simulator on your PC?

That's the 'VISION' I have, and let's see, if I will be able to achieve it.

For now, the first step towards this vision, has been achieved. Look for yourself in the below demo video, how am I doing this

https://youtu.be/KIaAEhpC-ak

__Happy Learning!!__