SCHEMATIC PCELLS, FUTURE OF DEEP SUBMICRON CUSTOM IC DESIGN

PRANAV BHUSHAN
CADENCE DESIGN SYSTEMS INC.
+91-120-2562842
pbhushan@cadence.com

RAJ ARUMUGAM
QUALCOMM INC.
+1-858-845-7877
rarumuga@qualcomm.com

CDN LIVE! SILICON VALLEY
September 12-15, 2005
Santa Clara, CA

ABSTRACT

As we make inroads into deep sub-micron processes (DSM)s, digital and analog designers alike see a clear need for accurate pre-layout modelling of second order effects for front-end simulations.

In this paper, we explore and improve upon the capabilities offered by schematic pcells to achieve more accurate representation of the physical effects of primitive devices in the netlist without the need for cumbersome netlist post-processing. In DSM, the physical effects like Shallow Trench Isolation (STI) and Nwell proximity effects play a significant role and needs to be accounted for during front-end simulations to closely match post-layout simulation results. Due to the limitations of the spice simulators, it becomes necessary that multi-fingered mos devices are represented as individual discrete transistors in the netlist to accurately estimate some of these effects.

In this paper, we demonstrate how one could take advantage of the pcell concept and virtual schematics to build custom logic cells and control how they are represented in the netlist. For a logic gate like a nand gate, one could also have the primitive transistors in a folded or stacked configuration yielding different parasitics. In our proof of concept library, based on user input for fingers, shared configuration and other parameters of interest, and with the help of a intelligent pcell code, we create schematics in virtual memory on the fly with the required connectivity information. Each finger is represented as a discrete transistor in the netlist with the associated parasitics and the implicit pcell connections are established using database calls. The current approach can also be ehnanced to build complex schematics to illustrate the default structure of the gate with explicit wires. The OSS (Open Simulation System) would reference the corresponding schematic in virtual memory when it hits an instance of a gate built as a schematic pcell during netlisting. There are also advantages to building top level transistors (tnmos/tpmos) using schematic pcells which in turn reference the primitive nmos/pmos devices built based on foundry specifications. When foundry changes their design specification which affects the way parasitics are calculated, legacy designs using nmos or pmos would have to go through an translation to update any inter-dependent instance parameters. If one uses tnmos/tpmos instead, since the underlying schematic is created on the fly every time a netlist is generated, the legacy designs are now immune to any changes that affect primitive nmos/pmos. We believe that the schematic pcell approach is superior to the conventional wisdom to write netlist post-processors to address such issues since this solution is very clean and requires very little maintenance. Also, one could extend this approach to say add inter-node capacitances and resistances at shared nodes in multi-fingered devices.

Using schematic pcells, one could come up with highly customizable cells that offer tremendous flexibility to analog designers with accurate modeling of layout parasitics at the front-end. Some future work related to this field will also be described in the paper.

Keywords: SKILL, analogLib, pcell, virtual schematics.