Digital VLSI Design Course Project 2

Problem statement- PVT aware Leakage power estimation

Part I: Exploratory Data Analysis

Project 2 is focused on leakage power estimation for larger circuits using machine learning models.
The dataset generation process consists of two key steps: first, creating combinations of Process, Voltage, and Temperature (PVT) by sampling these variables from distributions; second, using these values to simulate circuits. The static leakage power of the circuit will be measured and used as the target variable in the dataset.

Technology nodes: 45nm MGK, 32nm MGK, 32nm HP, 22nm MGK, 22nm HP, 16nm HP.

Target Output: Leakage Power

Standard Cells: NAND2, NOT, AND2, XOR2, OR2, AND3, etc.

Netlist Simulator: NGSPICE

The range of parameter distributions is given below. The values should be varied according to a Monte Carlo distribution, allowing for accurate analysis across PVT conditions. Use NGSPICE to perform the simulations.

1. Generate PVT variation matrix consisting of maximum **5000** samples.

PVT are:

- **Temp**: Temperature range should be from -55 to 125 Celsius (uniform distribution).
- **Pvdd**: ±10% variations from nominal must be considered (uniform distribution).
- **Lmin**: 22nm to 26nm (if tech=22nm MGK) (uniform distribution)
- **Wmin**: 22nm to 440nm (if tech=22nm MGK) (uniform distribution)

For Process, consider the values in the PTM file as nominal standards and vary it with ±3σ variations satisfying Monte Carlo distribution, (i.e., mean = nominal, std = mean/30):

toxe_n: electrical oxide thickness of nmos

toxm_n: physical oxide thickness of nmos

toxref_n: reference oxide thickness of nmos

toxe_p: electrical oxide thickness of pmos

toxm_p: physical oxide thickness of pmos

toxref_p: reference oxide thickness of pmos

toxp_par: parasitic parameter

xj_n: junction depth of nmos

xj_p: junction depth of pmos

ndep_n: doping concentration of nmos

ndep_p: doping concentration of pmos

Ensure dataset quality of standard cells since ML model performance for larger circuit will be dependent on it.

- Include PVT variation in the ngspice netlist. Check the netlist structure properly.
- Launch the simulations in ngspice and save the generated performance matrix in csv/excel.
- Once you have the dataset, you will conduct **Exploratory Data Analysis (EDA)** to analyse the data, identify patterns, and document your observations. Eval 1 will be taken till this part.

Along with Eval 1, you will submit a project proposal related to ML modelling. You will receive feedback on the approach you plan to take for the next stage of the project.

Methodology:

Construct spice netlists for **standard gates required for designing C499 netlist.**

- Generate maximum 5k PVT distributions within the limits specified above and store it in an input file.

CAUTION: For each PVT value, samples will be generated based on the input combinations. For instance, if an input gate has four combinations (00, 01, 10, 11), four samples will be created for each PVT value. Use your constructed SPICE netlist with the generated PVT samples (input file) to create the leakage power dataset.

- **The final dataset for standard cells will contain 5000 PVT* 2^ {no. of inputs} total samples with leakage power and the rest of the columns for PVT input information.**

Part II: Machine Learning

1. Develop regression-based machine learning models to estimate the leakage power of the respective standard cells.
2. Based on your developed approach, you will estimate the leakage power of standard cells while minimizing error. Your method will be evaluated using standard metrics (MAPE, R2 Score, MSE, etc.)
3. The following are example themes. Each team must propose **unique implementation or architecture designs** for their regression models before or during Eval 1:
   a. Meta-Learning: for e.g. ensemble-based approaches of classical machine learning models etc.
   b. Deep-Learning: for e.g. Use of deep neural network etc.
4. You will apply the models developed for standard cells to estimate the leakage power for the C17 and C499 circuit.(Note: You have to design the **ngspice netlist for the C499 circuit** using this example as reference- https://web.eecs.umich.edu/~jhayes/iscas.restore/c499.v)