Code Review

Methods of Hardware Obfuscation

Kaixin Yang

December 2, 2019

1. Importing the Original Desgin
   - Constructing graph from verilog
   - Identifying the state FFs
2. Obfuscating the Original Design
   - Inserting Gates

How to parse the verilog file?

xxxx reads the original verilog file line by line, and parses each line by some key words.
They define new structures called **'graph', 'vertex' and 'edge'**, which are corresponding to different elements in the netlist:

$$\text{netlist} \xRightarrow{\text{converted to}} \text{graph}$$

$$\text{gates/FFs} \xRightarrow{\text{converted to}} \text{vertices}$$

$$\text{inputs \& outputs of each vertex} \xRightarrow{\text{converted to}} \text{edges}$$

Reference:

Two steps to identify all state FFs (sequential vertices):

1. Identifying all the FFs, ...
2. Searching for cycles in the graph, ...

Example

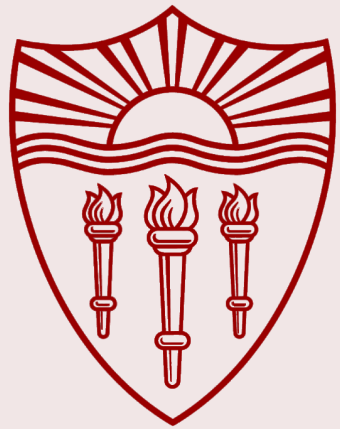

Figure: abc

Q1: How many gates inserted?
- Decided by users (variable in the configuration file)

Q2: Where to insert the gates?
- The outputs of combinational gates

Conclusion: xxxxxxxxxxx

Thank you!