

yosys を使ってみる

Ubuntu 20.04 だと , apt でインストールできる .

```
$ sudo apt install yosys
```

バージョンは ,

```
$ yosys -V  
Yosys 0.9 (git sha1 1979e0b)
```

とのこと . 起動するとプロンプトが表示されて対話的に利用できる

```
$ yosys  
...  
yosys>  
yosys> exit
```

試しに , 1-bit の加算を合成してみる .

```
module add (  
    input wire a,  
    input wire b,  
    output wire c  
);  
assign c = a + b;  
endmodule
```

対話的に処理してもいいけど ,
処理したいコマンド列をファイルに保存しておいて与えるのが便利 .
たとえば , のような add.yo とか作って

```
read_verilog add.v  
  
# elaborate design hierarchy  
hierarchy -check -top add  
  
# the high-level stuff  
proc; opt; fsm; opt; memory; opt  
  
show -format svg -prefix techmap_add  
write_verilog techmap_add.v  
write_json techmap_add.js  
  
# mapping to internal cell library  
techmap; opt  
  
# mapping flip-flops to mycells.lib  
dfflibmap -liberty cmos_cells.lib  
  
# mapping logic to mycells.lib  
abc -liberty cmos_cells.lib  
  
show -format svg -prefix cellmap_add  
write_verilog cellmap_add.v  
write_json cellmap_add.js  
  
clean
```

yosys を

```
yosys add.yo
```

などとして実行 . cmos_cells.lib は

<https://github.com/YosysHQ/yosys/tree/master/examples/cmos>

から持ってきた .

apt で yosys 入れたとしても , git リポジトリをクローンしておく方が便利かも .

生成されるファイルは

- techmap_add.v - '+' が xor になるレベルの途中経過
- techmap_add.js - JSON でも出力可能
- techmap_add.dot - Graphviz 用 dot ファイル
- techmap_add.svg - svg 形式の
- cellmap_add.v - cmos_cells.lib 使ってマッピングした後
- cellmap_add.js - 同じく JSON で出力可能
- cellmap_add.dot - Graphviz 用 dot ファイル
- cellmap_add.svg - svg 形式で出力

show に引数を与えなければ実行中に , Graphviz のウィンドウが表示される .

グラフは svg じゃなくて png でも出力できる . 出力はこんな感じ .

techmap_add.png

cellmap_add.png

続・雑用

各方面の状況の整理 .