

Obiectiv

Tema are ca scop exersarea lucrului cu noțiunile de Verilog folosite pentru proiectarea circuitelor secvențiale.

Descriere și cerințe

Implementați în Verilog un circuit secvențial sincron care parcurge un labirint de dimensiune 64×64.

Inițial fiecare punct din labirint poate fi descris de stări notate cu 0 și respectiv 1. Culoarul labirintului este descris de punctele marcate cu 0, iar pereții labirintului sunt marcați cu 1. Fiecare punct parcurs prin labirint va fi marcat cu 2, descriind astfel traseul dintre punctul de start și ieșirea din labirint. Regula de parcurgere a labirintului este dată de algoritmul Wall Follower cu urmărirea peretelui drept.

Implementare

Automatul cu stări finite care va modela comportamentul general trebuie implementat în modulul maze, având interfața:

```
module maze(  
    input          clk,  
    input [maze_width - 1:0] starting_col, starting_row, // indicii punctului de start  
    input          maze_in,                               // oferă informații despre punctul de coordonate [r  
    output [maze_width - 1:0] row, col,                   // selectează un rând și o coloană din labirint  
    output         maze_oe,                               // output enable (activează citirea din labirint la  
    output         maze_we,                               // write enable (activează scrierea în labirint la  
    output         done);                                 // ieșirea din labirint a fost găsită; semnalul răm
```

Descrierea semnalelor folosite de acest modul este următoarea:

- `clk` - semnal de sincronizare
- `starting_row`, `starting_col` - coordonatele punctului de start pentru labirint
- `row` - indexul rândului punctului care se dorește a fie citit/scriș
- `col` - indexul coloanei punctului care se dorește a fie citit/scriș
- `maze_in` - semnal care conține informația punctului de coordonate [row,col]
- `maze_oe` - când este activ returnează pe firul `maze_in` informația de la coordonatele [row,col]
- `maze_we` - când este activ marchează cu 2 în labirint poziția desemnata de indicii [row,col]
- `done` - indică faptul că s-a terminat parcurgerea labirintului; trebuie să fie activ după găsirea ieșirii din labirint



Este permisă modificarea declarării ieșirilor în output `reg`.

Algoritmul Wall Follower

Algoritmul Wall Follower este unul dintre cei mai cunoscuți algoritmi ce propun soluționarea unui labirint. Este recunoscut și după regula de dreapta sau de stânga.

În cazul în care labirintul este simplu conectat, urmărirea peretelui din dreapta, de exemplu, asigură că se ajunge la o ieșire diferită dacă există una, în caz contrar o să se întoarcă la intrarea de la care a pornit. Dacă labirintul nu este pur și simplu conectat, această metodă nu va atinge obiectivul. Mai multe despre algoritm în resurse.

Exemplu input pentru un labirint 8x8

```
starting_row = 1
starting_col = 1

  0 1 2 3 4 5 6 7
  ───────────
0 | 1 1 1 1 1 1 1 1
1 | 1 0 1 0 0 0 0 0
2 | 1 0 1 0 1 1 1 1
3 | 1 0 0 0 1 1 0 1
4 | 1 1 0 1 1 0 0 1
5 | 1 1 1 1 1 0 1 1
6 | 1 1 0 0 0 0 1 1
7 | 1 1 1 1 1 1 1 1
```

Exemplu output pentru un labirint 8x8

```
  0 1 2 3 4 5 6 7
  ───────────
0 | 1 1 1 1 1 1 1 1
1 | 1 2 1 2 2 2 2 2
2 | 1 2 1 2 1 1 1 1
3 | 1 2 2 2 1 1 0 1
4 | 1 1 2 1 1 0 0 1
5 | 1 1 1 1 1 0 1 1
6 | 1 1 0 0 0 0 1 1
7 | 1 1 1 1 1 1 1 1
```

Ruta folosind algoritmul Wall Follower:

(1,1) → (2,1) → (3,1) → (3,2) → (4,2) → (3,2) → (3,3) → (2,3) → (1,3) → (1,4) → (1,5) → (1,6) → **(1,7)**