

AC_TEMA2: Hiding secrets - README

Mărgăritescu Vlad - 333AC

În rezolvarea temei 2 am urmărit indicațiile din cerința și am implementat un automat cu stări finite care să modeleze ceea ce se cere. Acest automat comunică cu modulele deja implementate.

Pentru început, am instantiat modulul `base2_to_base3` așa cum era precizat în TODO, iar apoi am început realizarea FSM-ului propriu-zis.

Componentele și funcționarea automatului:

În prima parte am construit partea secvențială, iar apoi am continuat și cu cea combinatională în care am inclus toate stările și tranzițiile posibile ale automatului.

Stările automatului (pentru grayscale):

0: START

1: FIND_YELLOW

2: END_GRAYSCALE

Variabile de stare:

`state` & `next_state` (pentru starea curentă și următoarea stare)

Variabile auxiliare:

`row_reg` & `next_row_reg` (linia curentă și următoarea linie)

`col_reg` & `next_col_reg` (coloana curentă și următoarea coloană)

Explicații stări:

START: `out_we_reg` este inițial 0, deoarece încă nu este îndeplinită condiția de a activa scrierea pentru imaginea de ieșire. Tot aici sunt declarate și valorile minim și maxim, folosite în starea următoare pentru a găsi minimul și maximum pentru cele 3 canale din fiecare pixel. Apoi se merge la starea următoare.

FIND_YELLOW: Aici se iau cele 3 canale în parte (pe rând biții [23:16], [15:8], [7:0] și se caută valoarea minimă și maximă). La final, valoarea din yellow [15:8] este media aritmetică dintre minim și maxim, iar ceilalți biți devin 0. Semnalul `out_we_reg` devine 1 și acum se poate trece la ultima etapă din Greyscale.

END_GRAYSCALE: Aici se verifică dacă toți pixelii au fost testați. Dacă linia și coloana sunt 63 (adică 64 ținând cont că se numără de la 0) înseamnă că întreaga imagine a fost parcursă și se aserțiază semnalul `grey_done_reg`. Altfel, se trece din nou în starea inițială și se incrementează, după caz, linia și/sau coloana până când toată matricea de 64*64 a fost testată.