**Universitatea Tehnică Gh. Asachi Iași**

**Facultatea de electronică ,telecomunicații și tehnologia informației**

**Specializarea: Microelectronică, optoelectronică şi nanotehnologii**

## Proiect LDH

* **Numele disciplinei:** Limbaje de Descriere Hardware
* **Numărul temei de proiect:** 4

**Partea I** – Proiectarea şi analiza dinamică în SPICE a circuitului EDFFTR

* **Tehnologia CMOS: CMOS\_EKV26\_05**
* **Anul de studiu:** III
* **Numele studentului:** Alexa Alin-Vasile
* **Grupa:** 5314
* **An universitar: 2024-2025**

Tema nr. 4

**Partea I – Proiectarea şi analiza dinamică în SPICE a circuitului EDFFTR**

1. Să se dimensioneze tranzistoarele MOS corespunzător tehnologiei date şi să se scrie netlist-ul SPICE al circuitului. (Netlist-ul se va scrie ierarhic folosind subcircuite).
2. Să se efectueze analiza tranzitorie a circuitului pentru a pune în evidenţă funcţionarea sa. La fiecare ieşire a circuitului se va considera cîte o capacitate de sarcină. Valorile pentru tensiunea de alimentare VDD, timpii de tranziţie TR ai formelor de undă de la intrări şi, respectiv, capacităţile de sarcină CL se vor adopta în funcţie de tehnologia impusă conform tabelului următor:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tehnologie  CMOS | 0.13 μm | 0.18μm | 0.25μm | 0.35μm | 0.50μm |
| VDD | 1.2V | 1.8V | 2.5V | 3.3V | 5V |
| TR | 0.1ns | 0.15ns | 0.2ns | 0.3ns | 0.4n |
| CL | 0.06pF | 0.07pF | 0.08pF | 0.09pF | 0.1pF |

1. Determinaţi prin simulare parametrică şi folosind funcţii ţintă în *Probe* dependenţa timpilor de propagare a circuitului (de la intrarea de clock la ieşiri) în funcţie de capacitatea de sarcină (CL), respectiv în funcţie de timpul de tranziţie (TR) al formei de undă al intrării de clock. În ambele analize parametrice variaţia parametrilor se va face în intervalul (0.5 1.5)\* val. adoptată conform tehnologiei. Determinaţi şi notaţi pe

graficele obţinute valorile întârzierilor de propagare corespunzătoare valorilor adoptate pentru CL şi TR.

Considerând că dependenţa timpilor de propagare în funcţie de CL este de forma:

tptotal = tpintrinsec + Kload\*CL

determinaţi pe baza graficelor obţinute valorile pentru tpintrinsec şi Kload şi completaţi un tabel de felul următor (tpintrinsec corespunde valorii CL=0pF):

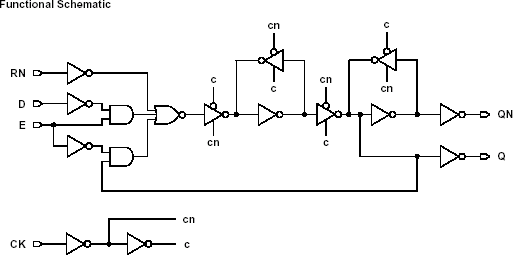
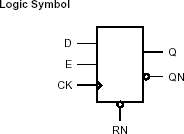
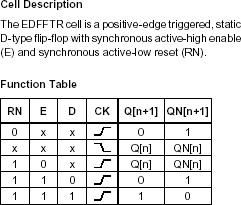
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Descriere | tpLH | | tpHL | |
| tpLHintrinsec | Kload | tpHLintrinsec | Kload |
| CK → Q |  |  |  |  |
| CK → QN |  |  |  |  |

1. Determinaţi prin simulare parametrii de constrângere SETUP\_TIME, HOLD\_TIME şi durata minimă a pulsului MINPW pentru pinii de intrare specificaţi în tabelul următor:

|  |  |  |
| --- | --- | --- |
| Pin intrare | Parametru constrângere | Durata interval  [ns] |
| D | SETUP\_TIME\_LH → CK |  |
| SETUP\_TIME\_HL → CK |  |
| HOLD\_TIME\_LH → CK |  |
| HOLD\_TIME\_HL → CK |  |
| CK | MINPW\_HIGH |  |
| MINPW\_LOW |  |

Pentru simulările efectuate la fiecare din punctele 2, 3 şi 4 se vor prezenta schema de test, fişierul SPICE (.cir) şi formele de undă sau caracteristicile reprezentative pe baza cărora s-au determinat parametrii ceruţi în tabele.

Mai jos sunt prezentate descrierea, schema bloc şi tabelul de funcţionare al circuitului.



O imagine care conține text, schiță, desen, tablă albă de scris

Descriere generată automatO imagine care conține text, scris de mână, papetărie, produse de papetărie

Descriere generată automatO imagine care conține text, scris de mână, papetărie, caiet

Descriere generată automatO imagine care conține text, scris de mână, tablă albă de scris, cerneală

Descriere generată automat

\*Proiect LDH Tema4

\*SUBCIRCUIT INVERSOR

\* I O Vdd

.subckt INV 1 2 3

MN1 2 1 0 0 NMOS L=0.5U W=1U

\* D G S Substr

MP1 2 1 3 3 PMOS L=0.5U W=4.28U

.ends

\*SUBCIRCUIT NAND2

\* I I O Vdd

.subckt NAND2 1 2 4 5

MN1 4 1 3 0 NMOS L=0.5U W=1U

MN2 3 2 0 0 NMOS L=0.5U W=1U

MP1 4 1 5 5 PMOS L=0.5U W=2.14U

MP2 4 2 5 5 PMOS L=0.5U W=2.14U

.ends

\*SUBCIRCUIT AND2

\* I I O Vdd

.subckt AND2 1 2 4 5

Xnand2 1 2 3 5 NAND2

Xinv 3 4 5 INV

.ends

\*SUBCIRCUIT NOR3

\* I I I O Vdd

.subckt NOR3 1 2 3 5 4

MN1 5 1 0 0 NMOS L=0.5U W=1U

MN2 5 2 0 0 NMOS L=0.5U W=1U

MN3 5 3 0 0 NMOS L=0.5U W=1U

MP1 7 1 4 4 PMOS L=0.5U W=12.86U

MP2 6 2 7 4 PMOS L=0.5U W=12.86U

MP3 5 3 6 4 PMOS L=0.5U W=12.86U

.ends

\*SUBCIRCUIT INVERSOR COMANDAT

\* I C Cn O Vdd

.subckt INVC 1 2 3 4 5

MN1 4 2 7 0 NMOS L=0.5U W=1U

MN2 7 1 0 0 NMOS L=0.5U W=1U

MP1 6 1 5 5 PMOS L=0.5U W=4.28U

MP2 4 3 6 5 PMOS L=0.5U W=4.28U

.ends

\*CIRCUIT DE LUCRU

\* RN D E CK QN Q Vdd

.subckt EDFFTR 1 2 3 13 15 16 17

X1 1 4 17 INV

X2 2 5 17 INV

X3 3 6 17 INV

X4 5 3 8 17 AND2

X5 6 7 9 17 AND2

X6 4 8 9 10 17 NOR3

X7 10 20 21 11 17 INVC

X8 11 12 17 INV

X9 12 21 20 11 17 INVC

X10 12 21 20 7 17 INVC

X11 7 14 17 INV

X12 14 20 21 7 17 INVC

X13 14 15 17 INV

X14 7 16 17 INV

X15 13 20 17 INV

X16 20 21 17 INV

.ends

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0.4p

.TRAN 1n 100n

.PROBE

.END

\*Circuit main

\*Vin 1 0 DC=0

\*+PULSE (0 3.3 0 {TR} {TR} {10n-TR} 20n)

\*.Param TR=1N

\*Vdd 2 0 5

\*X1 1 3 2 INV

\*CL= 3 0 {CVAL}

\*.Param CVAL=0.1p

\*.INC CMOS\_EKV26\_05.txt

\*.Step Param CVAL 0.05p 0.15p 0.01p

\*.TRAN 0.1n 20n 0 0.1n

\*.DC Vin 0 5 30m

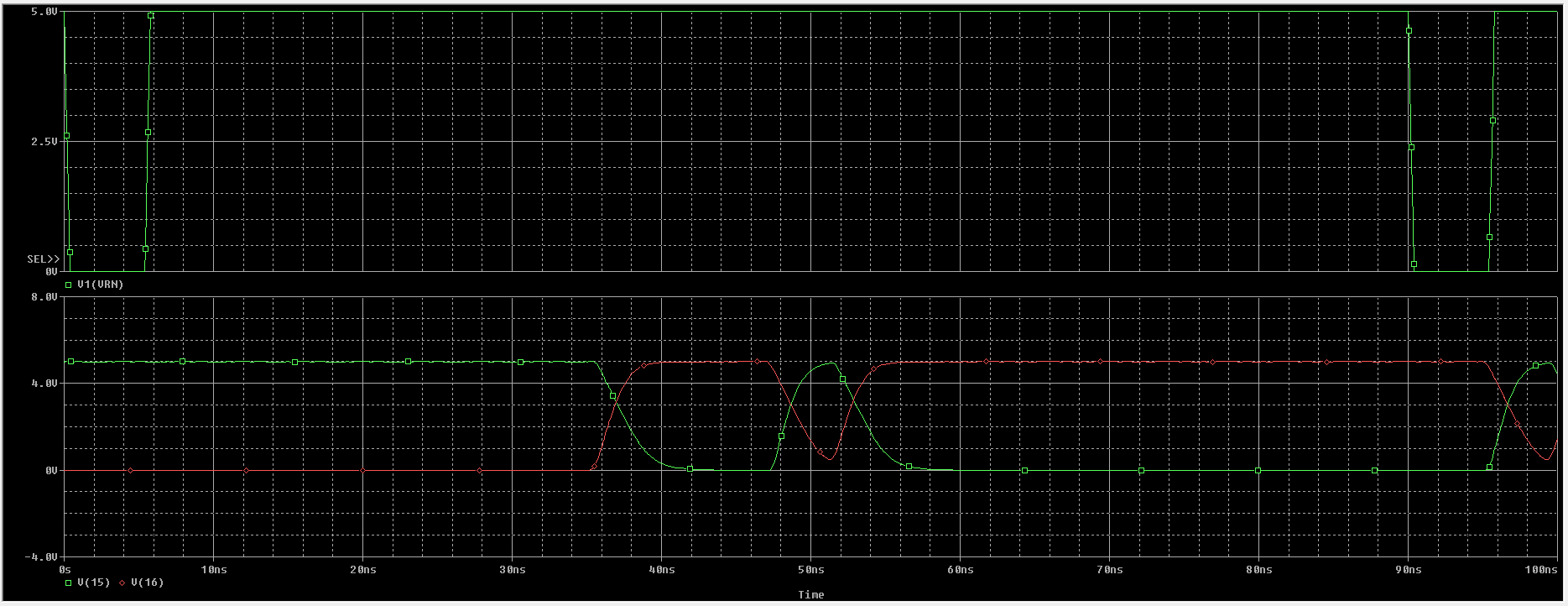
\*.Probe

\*.End

O imagine care conține captură de ecran, electronice, afișaj

Descriere generată automat

C=0.4p



C=0.0pO imagine care conține captură de ecran

Descriere generată automat

Timpi de propagare la tranzitia din LOW in HIGH la o analiza parametrica in care se variaza parametru CL al iesirii Q.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_intrinsec=0.4989ns

O imagine care conține captură de ecran, text, linie

Descriere generată automat

T\_total=1.3905ns

Timpi de propagare la tranzitia din LOW in HIGH la o analiza parametrica in care se variaza parametru CL al iesirii QN.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_intrinsec=0.80851ns

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_total=1.2146ns

Timpi de propagare la tranzitia din HIGH in LOW la o analiza parametrica in care se variaza parametru CL al iesirii Q.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran, linie, Interval

Descriere generată automat

T\_intrinsec=0.66008ns

O imagine care conține captură de ecran, linie, Interval, text

Descriere generată automat

T\_total=1.3902ns

Timpi de propagare la tranzitia din HIGH in LOW la o analiza parametrica in care se variaza parametru CL al iesirii QN.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_intrinsec=0.644025ns

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_total=1.3666ns

Timpi de propagare la tranzitia din LOW in HIGH la o analiza parametrica in care se variaza parametrul TR al iesirii Q.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

\*.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.STEP param tr list 0.2n 0.3n 0.4n 0.5n 0.6n

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran

Descriere generată automat

T\_intrinsec=0.82016ns

O imagine care conține captură de ecran, text

Descriere generată automat

T\_total=1.4901ns

Timpi de propagare la tranzitia din LOW in HIGH la o analiza parametrica in care se variaza parametrul TR al iesirii QN.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

\*.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.STEP param tr list 0.2n 0.3n 0.4n 0.5n 0.6n

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_intrinsec=0.78058ns

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_total=1.4929ns

Timpi de propagare la tranzitia din HIGH in LOW la o analiza parametrica in care se variaza parametrul TR al iesirii Q.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

\*.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.STEP param tr list 0.2n 0.3n 0.4n 0.5n 0.6n

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran, linie

Descriere generată automat

T\_intrinsec=0.631308ns

O imagine care conține captură de ecran

Descriere generată automat

T\_total=1.2851ns

Timpi de propagare la tranzitia din HIGH in LOW la o analiza parametrica in care se variaza parametrul TR al iesirii QN.

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 0.4N 0.4N 5N 90N)

VD 2 0 PULSE(0 5 0 0.4N 0.4N 12N 16N)

VE 3 0 PULSE(5 0 0 0.4n 0.4N 30N 60n)

.param vdd=5

.param tr=0.4n

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0p

\*.STEP param CL LIST 0 0.05p 0.06p 0.09p 0.11p 0.15p

.STEP param tr list 0.2n 0.3n 0.4n 0.5n 0.6n

.TRAN 1n 100n

.PROBE

.END

O imagine care conține captură de ecran

Descriere generată automat

T\_intrinsec=1.0568ns

O imagine care conține captură de ecran

Descriere generată automat

T\_total=1.7203ns

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Descriere | tpLH | | tpHL | |
| tpLHintrinsec | Kload | tpHLintrinsec | Kload |
| CK → Q | 0.4989ns | 2.229 | 0.66008ns | 1.3863 |
| CK → QN | 0.80851ns | 1.3863 | 0.644025ns | 1.8064 |

tpTotal = tpIntrinsec + Kload\*CL

Kload = (tpTotal - tpIntrinsec)/CL

TP LH CL Q

(1.3905-0.4989)e-9/0.4e-9 = 0.8916/0.4 = 2.229 s/F

TP LH CL QN

(1.2146-0.80851)e-9/0.4e-9 = 0.554452/0.4 = 1.3863 s/F

TP HL CL Q

(1.3902-0.66008)e-9/0.4e-9 = 0.73012/0.4 = 1.3863 s/F

TP HL CL QN

(1.3666-0.644025)e-9/0.4e-9 = 0.722575/0.4 = 1.8064 s/F

**Determinarea timpului SETUP\_TIME\_LH**

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(0 5 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 {tr} {tr} 5N 90N)

VD 2 0 PULSE(0 5 {7.495n-DL} {tr} {tr} 12N 16N)

VE 3 0 PULSE(5 0 0 {tr} {tr} 30N 60n)

.param vdd=5

.param tr=0.4p

CL1 16 0 {CL}

CL2 15 0 {CL}

.param CL=0.1p

.param DL=10p

\*.step param dl 0n 0.1n 10p

.TRAN 0.0001N 14n 1n

.PROBE

.END

O imagine care conține captură de ecran

Descriere generată automat

O imagine care conține captură de ecran

Descriere generată automat

Setup\_time LH = 8.015ns-7.5ns = 0.515 ns

**Determinarea timpului SETUP\_TIME\_HL**

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(0 5 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 {tr} {tr} 5N 90N)

VD 2 0 PULSE(0 5 {7.495n-DL} {tr} {tr} 12N 16N)

VE 3 0 PULSE(5 0 0 {tr} {tr} 30N 60n)

.param vdd=5

.param tr=0.4p

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0.1p

.param DL=10p

.step param dl 0n 0.1n 10p

.TRAN 0.0001N 50n 1n

.PROBE

.END

O imagine care conține captură de ecran

Descriere generată automat

O imagine care conține captură de ecran, text

Descriere generată automat

Setup\_time\_HL = 20.015ns – 19.53ns = 0.485 ns

**Determinarea timpului HOLD\_TIME\_LH:**

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(0 5 0 {tr} {tr} 2n 4n)

VRN 1 0 PULSE(5 0 0 {tr} {tr} 5N 90N)

VD 2 0 PULSE(0 5 {7.495n-DL} {tr} {tr} 12N 16N)

VE 3 0 PULSE(5 0 0 {tr} {tr} 30N 60n)

.param vdd=5

.param tr=0.4p

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0.1p

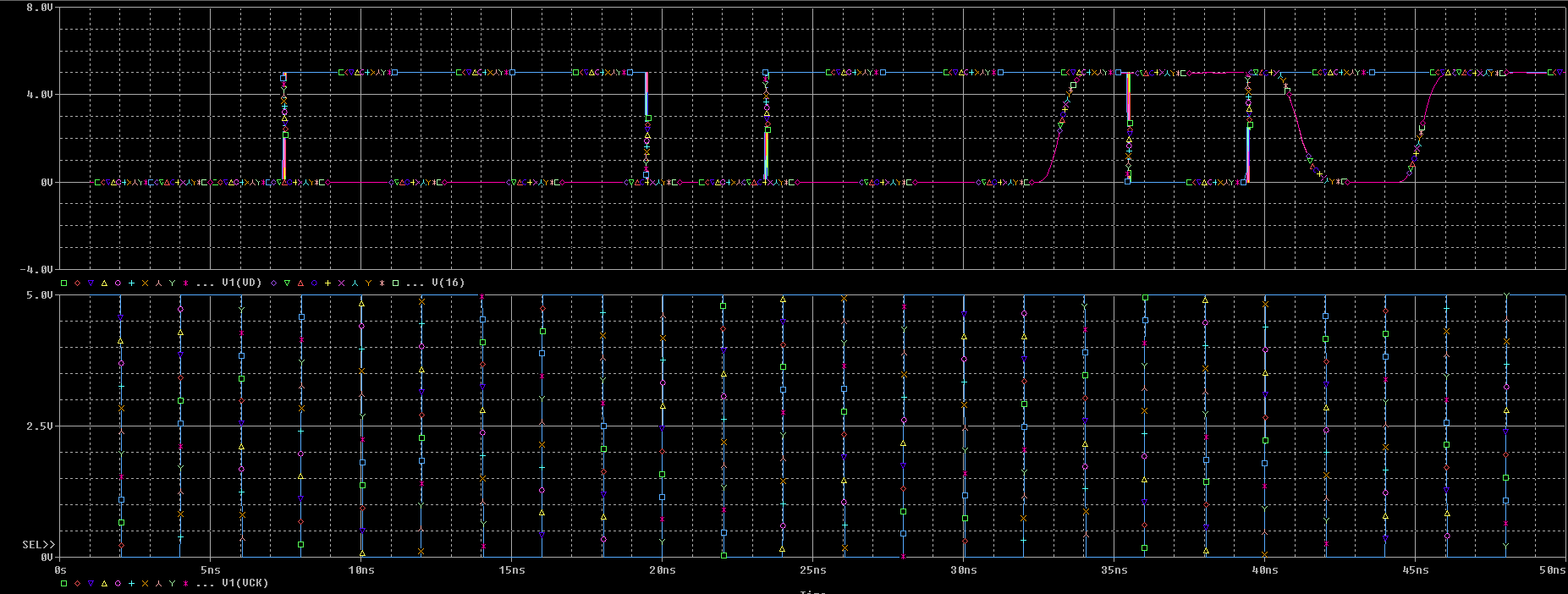
.param DL=10p

.step param dl 0n 0.1n 10p

.TRAN 0.0001N 50n 1n

.PROBE

.END



**Calculul MINPW\_HIGH**

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(0 5 0 {tr} {tr} {pw} 4n)

VRN 1 0 PULSE(5 0 0 {tr} {tr} 5N 90N)

VD 2 0 PULSE(0 5 0 30p 30p 2n 6n)

VE 3 0 PULSE(5 0 0 {tr} {tr} 30N 60n)

.param vdd=5

.param tr=0.4p

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0.1p

.param pw=0.1n

.STEP PARAM pw 0.05N 0.1N 0.01N

.TRAN 0.001N 44n 1n

.PROBE

.END

O imagine care conține captură de ecran, spațiu, linie

Descriere generată automat

**Calculul MINPW\_LOW**

\*Testarea circuitului

.INC CMOS\_EKV26\_05.txt

XEDFFTR 1 2 3 13 15 16 17 EDFFTR

VDD 17 0 {vdd}

VCK 13 0 PULSE(5 0 0 {tr} {tr} {pw} 4n)

VRN 1 0 PULSE(5 0 0 {tr} {tr} 5N 90N)

VD 2 0 PULSE(5 0 0 30p 30p 2n 6n)

VE 3 0 PULSE(5 0 0 {tr} {tr} 30N 60n)

.param vdd=5

.param tr=0.4p

CL1 15 0 {CL}

CL2 16 0 {CL}

.param CL=0.1p

.param pw=0.1n

.STEP PARAM pw 0.05N 0.1N 0.01N

.TRAN 0.001N 44n 1n

.PROBE

.END

O imagine care conține captură de ecran

Descriere generată automat

O imagine care conține captură de ecran, linie

Descriere generată automat

|  |  |  |  |
| --- | --- | --- | --- |
| Pin intrare | Parametru constrângere | Durata interval |  |
| [ns] |  |
|  |  |  |
|  | SETUP\_TIME\_LH → CK | 0.515 |  |
| D | SETUP\_TIME\_HL → CK | 0.485 |  |
| HOLD\_TIME\_LH → CK | 0.886 |  |
|  |  |  |
|  | HOLD\_TIME\_HL → CK | 0.891 |  |
| CK | MINPW\_HIGH |  |  |
| MINPW\_LOW |  |  |
|  |  |  |

***VHDL***

Codul pentru DFF(EDFFTR):

library IEEE; use IEEE.STD\_LOGIC\_1164.ALL;

entity EDFFTR is Port ( RN : in STD\_LOGIC;

E : in STD\_LOGIC;

D : in STD\_LOGIC;

CK : in STD\_LOGIC;

Q : out STD\_LOGIC;

QN : out STD\_LOGIC

); end EDFFTR;

architecture Comportament of EDFFTR is

-- Parametrii dinamici (în ps pentru simulare precisă)

constant tpLH\_CK\_Q : time := 498 ps; -- Timp de propagare CK -> Q

constant tpHL\_CK\_Q : time := 660 ps; -- Timp de propagare CK -> Q

constant tpLH\_CK\_QN : time := 808 ps; -- Timp de propagare CK -> QN

constant tpHL\_CK\_QN : time := 644 ps; -- Timp de propagare CK -> QN

constant tSETUP : time := 515 ps; -- Setup time pentru D înainte de CK

constant tHOLD : time := 886 ps; -- Hold time pentru D după CK

begin

process(CK)

begin if rising\_edge(CK) then

-- Verificarea timpului de setup

assert D'stable(tSETUP)

report "Setup time violated" severity error;

-- Verificarea timpului de hold  
 assert D'stable(tHOLD)  
 report "Hold time violated" severity error;  
  
 -- Comportamentul bistabilului  
 if RN = '0' then  
 Q <= '0' after tpHL\_CK\_Q;  
 QN <= '1' after tpLH\_CK\_QN;  
 elsif E = '1' then  
 Q <= D after tpLH\_CK\_Q;  
 QN <= not D after tpHL\_CK\_QN;  
 end if;  
 end if;  
end process;

end Comportament;

Testbench EDFFTR

library IEEE; use IEEE.STD\_LOGIC\_1164.ALL;

entity TB\_EDFFTR is -- Testbench-ul nu are porturi end TB\_EDFFTR;

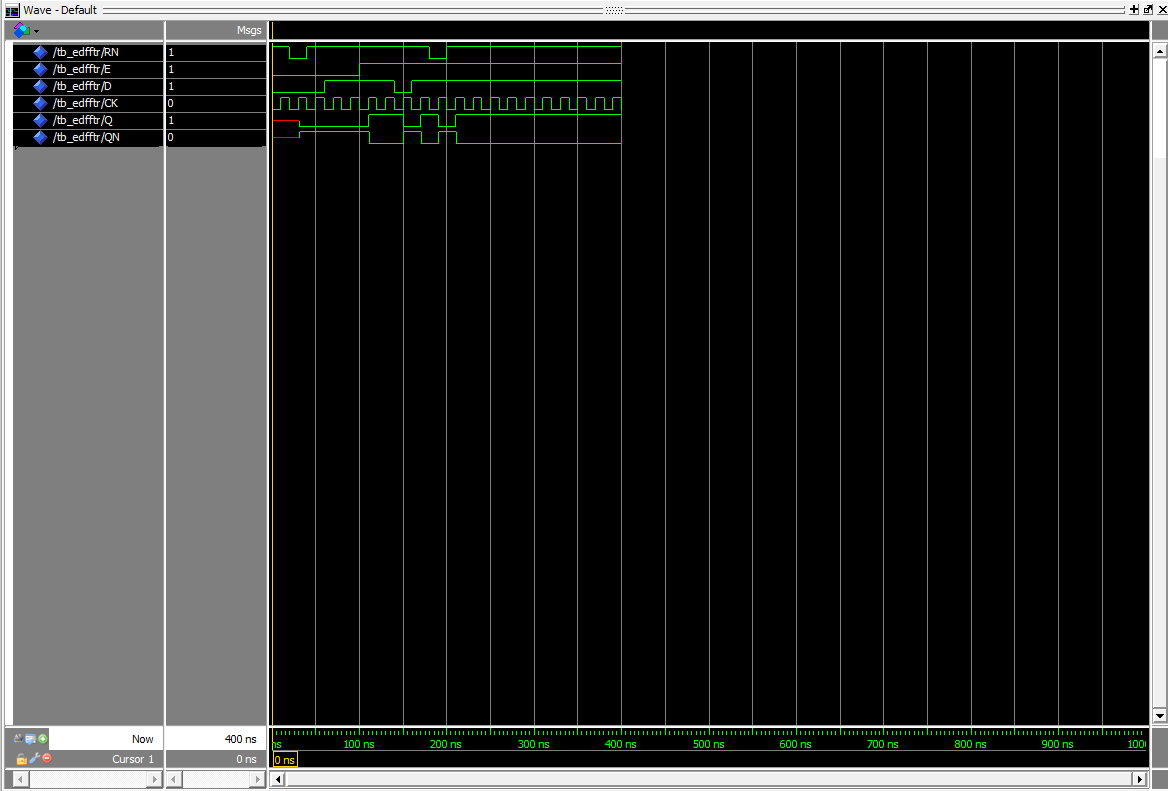
architecture Comportament of TB\_EDFFTR is

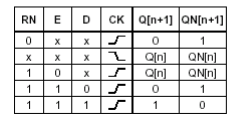
-- Declararea semnalelor de test  
signal RN : STD\_LOGIC := '1'; -- R initial   
signal E : STD\_LOGIC := '0'; -- E initial   
signal D : STD\_LOGIC := '0'; -- D initial  
signal CK : STD\_LOGIC := '0'; -- Ceas ini?ial  
signal Q : STD\_LOGIC;   
signal QN : STD\_LOGIC;

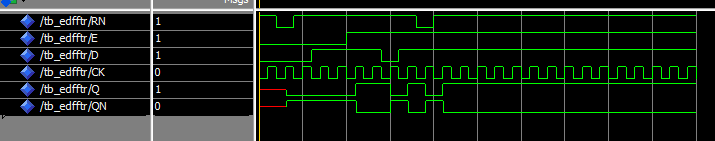
begin

-- Instantierea DFF  
uut: entity work.EDFFTR  
 port map (  
 RN => RN,  
 E => E,  
 D => D,  
 CK => CK,  
 Q => Q,  
 QN => QN  
 );  
  
-- Generarea semnalului de ceas (50% duty cycle)  
clk\_process: process  
begin  
 while true loop  
 CK <= '0';  
 wait for 10 ns;  
 CK <= '1';  
 wait for 10 ns;  
 end loop;  
end process;  
  
-- Proces de testare  
test\_process: process  
begin  
 -- Ini?ializare  
 wait for 20 ns; -- A?teapt? dou? cicluri de ceas pentru stabilizare  
  
 -- Test 1: Reset activ (RN = 0)  
 RN <= '0'; wait for 20 ns; -- Activeaz? resetul  
 RN <= '1'; wait for 20 ns; -- Dezactiveaz? resetul  
  
 -- Test 2: Enable dezactivat (E = 0)  
 D <= '1'; E <= '0'; wait for 40 ns; -- Schimb? valoarea lui D, dar ie?irea nu trebuie s? se modifice  
  
 -- Test 3: Enable activ (E = 1)  
 E <= '1'; wait for 40 ns; -- Ie?irea Q trebuie s? preia valoarea lui D  
  
 -- Test 4: Schimbarea valorii lui D cu enable activ  
 D <= '0'; wait for 20 ns;  
 D <= '1'; wait for 20 ns;  
  
 -- Test 5: Reset activ cu enable activ  
 RN <= '0'; wait for 20 ns; -- Ie?irile trebuie resetate  
 RN <= '1'; wait for 20 ns;  
  
 -- Încheierea simul?rii  
 wait;  
end process;

end Comportament;







***VERILOG***

`timescale 1ps / 1ps

module EDFFTR (

input wire RN, // Reset asincron (activ pe '0')

input wire E, // Enable

input wire D, // Intrare de date

input wire CK, // Semnal de ceas

output reg Q, // Iesirea

output wire QN // Iesirea negata

);

// Iesirea complementara este negarea lui Q  
assign QN = ~Q;  
  
// Parametri generici pentru timpi dinamici  
parameter tpLH\_CK\_Q = 498;

parameter tpHL\_CK\_Q = 660;

parameter tpLH\_CK\_QN = 808;

parameter tpHL\_CK\_QN = 644;

parameter tSETUP = 515;

parameter tHOLD = 886;

always @(posedge CK or negedge RN) begin  
 if (!RN) begin  
 Q <= #tpHL\_CK\_Q 1'b0; // Reset pe Q  
 end else if (E) begin  
 Q <= #tpLH\_CK\_Q D; // Transfer? D la Q  
 end  
end  
  
specify  
 // Specifica?ii pentru timpii de setup ?i hold  
 $setup(D, posedge CK, tSETUP);  
 $hold(posedge CK, D, tHOLD);  
endspecify

endmodule

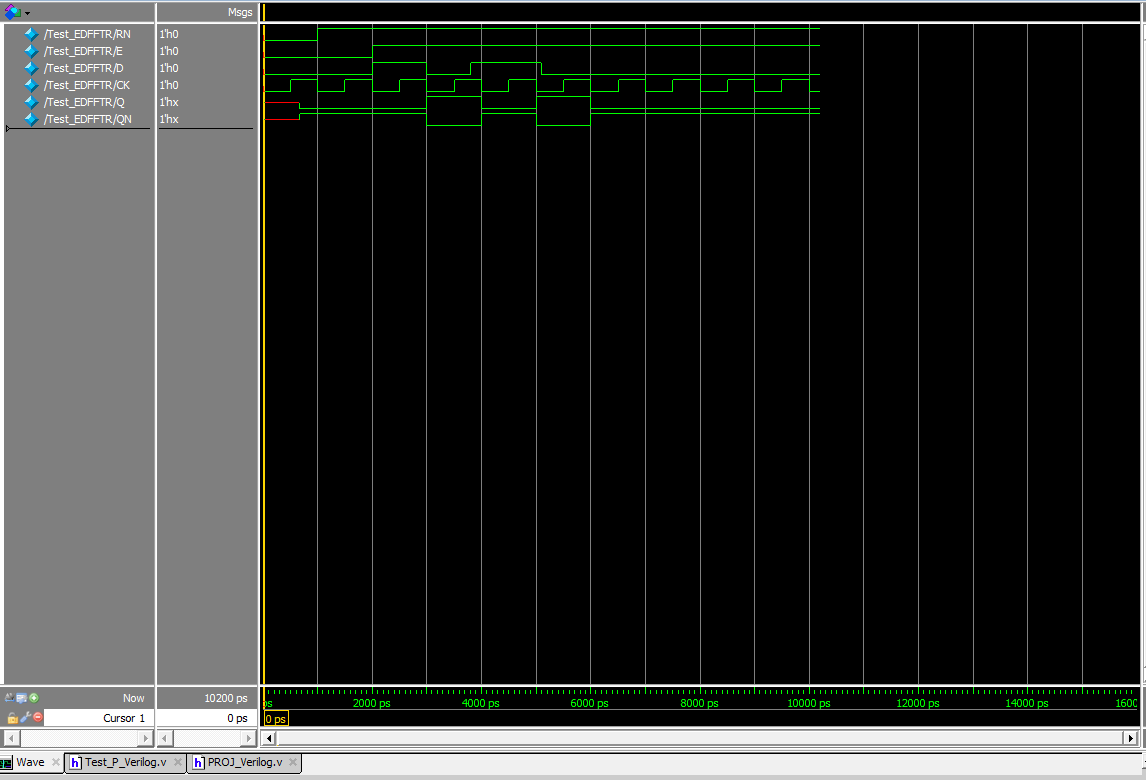
Simulare EDFFTR Verilog

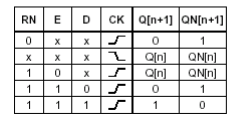
`timescale 1ps / 1ps

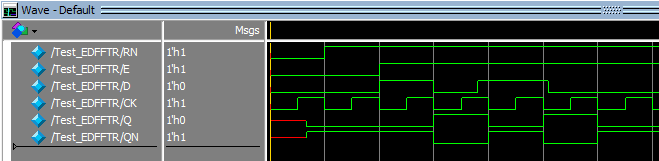
module Test\_EDFFTR;

reg RN, E, D, CK; // Semnale de intrare  
wire Q, QN; // Semnale de ie?ire  
  
// Instan?ierea modulului EDFFTR  
EDFFTR #(  
 .tpLH\_CK\_Q(498),  
 .tpHL\_CK\_Q(660),  
 .tpLH\_CK\_QN(808),  
 .tpHL\_CK\_QN(644),  
 .tSETUP(515),  
 .tHOLD(886)  
) uut (  
 .RN(RN),  
 .E(E),  
 .D(D),  
 .CK(CK),  
 .Q(Q),  
 .QN(QN)  
);  
  
// Generarea semnalului de ceas  
initial begin  
 CK = 0;  
 forever #500 CK = ~CK; // Perioada ceasului: 1 ns (500 ps pentru fiecare tranzi?ie)  
end  
  
// Testare comportament  
initial begin  
 // Resetare ini?ial?  
 RN = 0; E = 0; D = 0;  
 #1000 RN = 1; // Activare reset dup? 1 ns  
  
 // Test cu Enable activ  
 #1000 D = 1; E = 1; // D = 1, E = 1  
 #1000 D = 0; // Schimbare pe D  
   
 // Test pentru timpi de setup ?i hold  
 #800 D = 1; #300; // Setup time = 515 ps  
 #1000 D = 0; #100; // Hold time = 886 ps  
  
 // Finalizare simulare  
 #5000 $stop;  
end

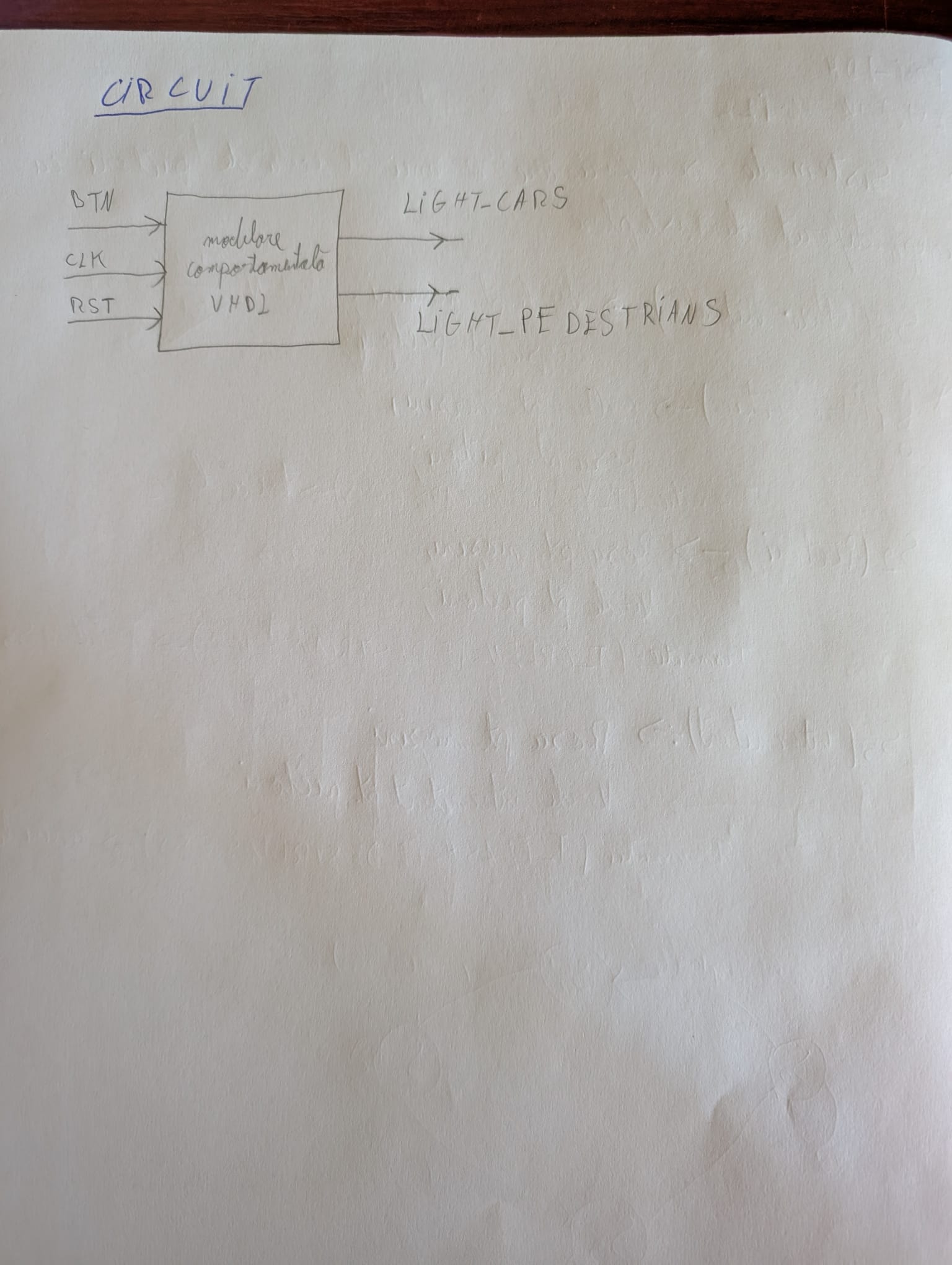
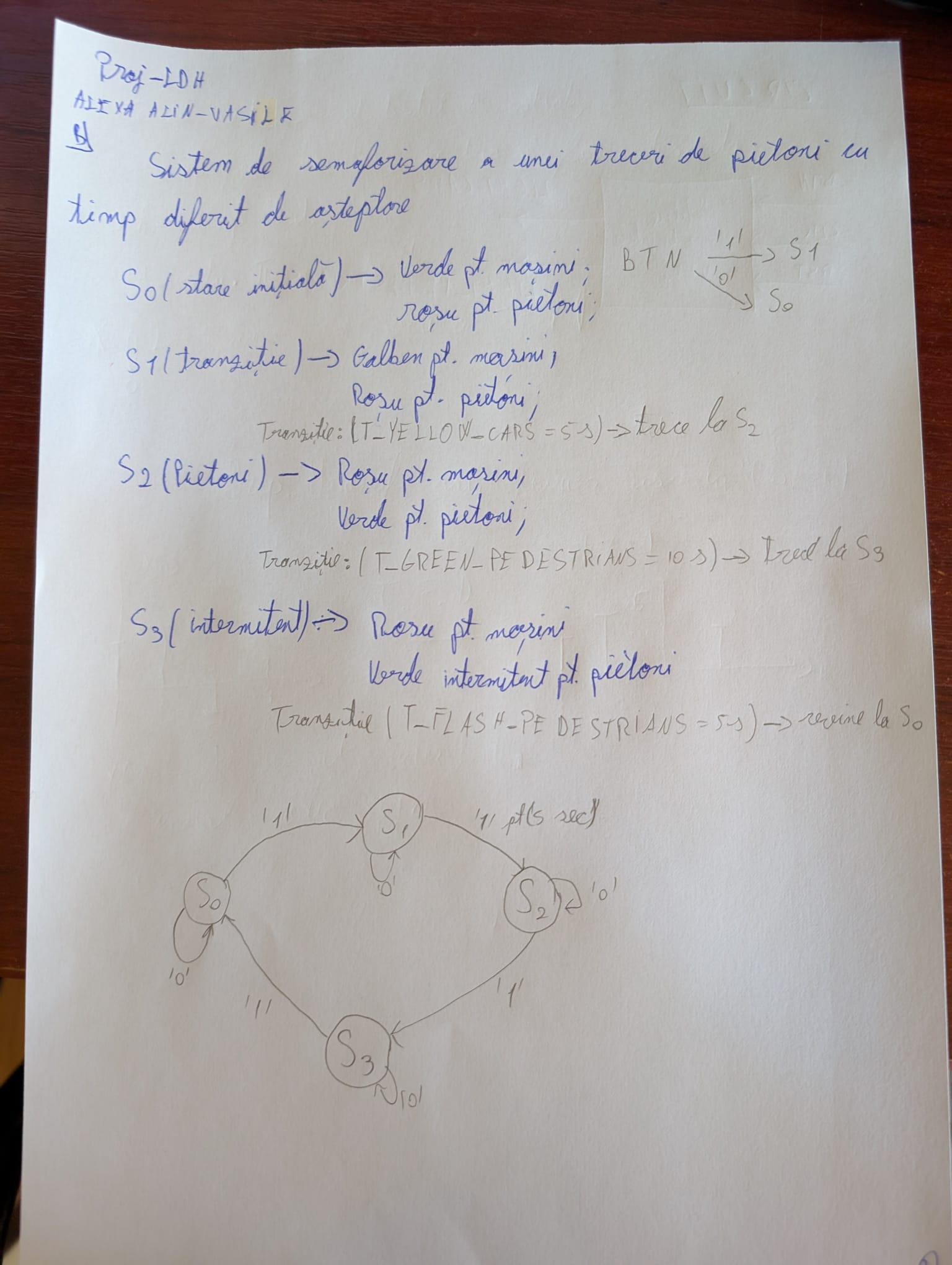
endmodule







***PARTEA B***



***Descrierea Circuitului***

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.NUMERIC\_STD.ALL;

entity TrafficLightSystem is

Port ( CLK : in STD\_LOGIC; -- Semnal de ceas

RST : in STD\_LOGIC; -- Reset asincron

BTN : in STD\_LOGIC; -- Buton pentru pietoni

LIGHT\_CARS : out STD\_LOGIC\_VECTOR(2 downto 0) ; -- Semafor ma?ini (3 LED-uri) LIGHT\_PEDESTRIANS : out STD\_LOGIC\_VECTOR(1 downto 0) -- Semafor pietoni (2 LED-uri)

);

end TrafficLightSystem;

architecture Behavioral of TrafficLightSystem is

-- Definirea st?rilor FSM

type state\_type is (S0, S1, S2, S3);

signal current\_state, next\_state : state\_type;

-- Contor pentru timp  
signal timer : integer := 0;  
  
-- Constante pentru timpi  
constant T\_GREEN\_CARS : integer := 30; -- 30 secunde pentru verde ma?ini  
constant T\_YELLOW\_CARS : integer := 5; -- 5 secunde pentru galben ma?ini  
constant T\_GREEN\_PEDESTRIANS : integer := 10; -- 10 secunde pentru verde pietoni  
constant T\_FLASH\_PEDESTRIANS : integer := 5; -- 5 secunde pentru verde intermitent

Begin

-- Proces pentru FSM

process(CLK, RST)

Begin

if RST = '1' then

current\_state <= S0; -- Reset la starea ini?ial?

timer <= 0;

elsif rising\_edge(CLK) then

current\_state <= next\_state;

-- Contor pentru timp  
 if current\_state = next\_state then  
 timer <= timer + 1;  
 else  
 timer <= 0;  
 end if;  
 end if;  
end process;  
  
-- Proces pentru tranzi?iile FSM  
process(current\_state, BTN, timer)  
begin  
 case current\_state is  
 when S0 =>  
 -- Verde pentru ma?ini, ro?u pentru pietoni  
 LIGHT\_CARS <= "100"; -- Verde  
 LIGHT\_PEDESTRIANS <= "01"; -- Ro?u  
 if BTN = '1' then  
 next\_state <= S1;  
 else  
 next\_state <= S0;  
 end if;  
  
 when S1 =>  
 -- Galben pentru ma?ini  
 LIGHT\_CARS <= "010"; -- Galben  
 LIGHT\_PEDESTRIANS <= "01"; -- Ro?u  
 if timer >= T\_YELLOW\_CARS then  
 next\_state <= S2;  
 else  
 next\_state <= S1;  
 end if;  
  
 when S2 =>  
 -- Ro?u pentru ma?ini, verde pentru pietoni  
 LIGHT\_CARS <= "001"; -- Ro?u  
 LIGHT\_PEDESTRIANS <= "10"; -- Verde  
 if timer >= T\_GREEN\_PEDESTRIANS then  
 next\_state <= S3;  
 else  
 next\_state <= S2;  
 end if;  
  
 when S3 =>  
 -- Ro?u pentru ma?ini, verde intermitent pentru pietoni  
 LIGHT\_CARS <= "001"; -- Ro?u  
 if (timer mod 2) = 0 then  
 LIGHT\_PEDESTRIANS <= "10"; -- Verde  
 else  
 LIGHT\_PEDESTRIANS <= "00"; -- Stins  
 end if;  
 if timer >= T\_FLASH\_PEDESTRIANS then  
 next\_state <= S0;  
 else  
 next\_state <= S3;  
 end if;  
  
 when others =>  
 next\_state <= S0;  
 end case;  
end process;

end Behavioral;

**TEST\_BANCH**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.NUMERIC\_STD.ALL;

entity TrafficLightSystem\_tb is

-- Entitatea testbench nu are porturi

end TrafficLightSystem\_tb;

architecture Behavioral of TrafficLightSystem\_tb is

-- Semnale interne pentru conectare

signal CLK : STD\_LOGIC := '0';

signal RST : STD\_LOGIC := '0';

signal BTN : STD\_LOGIC := '0';

signal LIGHT\_CARS : STD\_LOGIC\_VECTOR(2 downto 0);

signal LIGHT\_PEDESTRIANS : STD\_LOGIC\_VECTOR(1 downto 0);

-- Perioada semnalului de ceas (1s)

constant CLK\_PERIOD : time := 1 sec;

Begin

uut: entity work.TrafficLightSystem

port map

( CLK => CLK,

RST => RST,

BTN => BTN,

LIGHT\_CARS => LIGHT\_CARS,

LIGHT\_PEDESTRIANS => LIGHT\_PEDESTRIANS

);

-- Generarea semnalului de ceas  
CLK\_process: process  
begin  
 while true loop  
 CLK <= '0';  
 wait for CLK\_PERIOD / 2;  
 CLK <= '1';  
 wait for CLK\_PERIOD / 2;  
 end loop;  
end process;  
  
-- Stimuli pentru test  
Stimuli\_process: process  
begin  
 -- Reset activ timp de 2s  
 RST <= '1';  
 wait for 2 sec;  
 RST <= '0';  
  
 -- Ap?sare buton la 10s  
 wait for 10 sec;  
 BTN <= '1';  
 wait for 1 sec;  
 BTN <= '0'; -- Eliberare buton  
  
 -- Ap?sare buton la 50s  
 wait for 50 sec;  
 BTN <= '1';  
 wait for 1 sec;  
 BTN <= '0'; -- Eliberare buton  
  
 -- Finalizarea simul?rii dup? 100s  
 wait for 100 sec;  
 wait;  
end process;

end Behavioral;