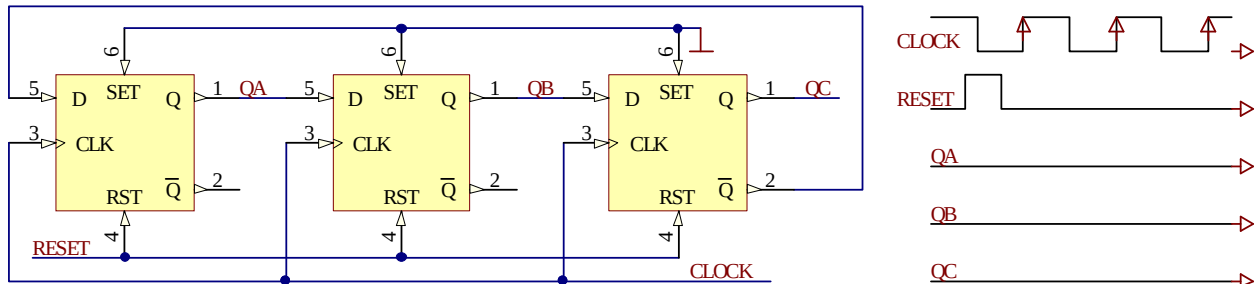


Név:

Index szám:

Pontszám:
(max. 50)

1. Rajtolja meg az $F_1=f(A,B,C,D,E)$ függvénynek megfelelő logikai hálózatot minimális számú logikai kapuval, amely csak az 10010 bemeneti kombinációra ad logikai egyes kimenetet! Valósítsa meg egy külön hálózattal a másik függvényt (F_2), amely a fenti esetben nullát ad, minden más esetben pedig egyest! (10 pont)
2. A megadott kapcsolásra rajzolja meg a QA, QB és QC jelek idődiagramját a CLOCK jel 15 periódusára, figyelembe véve a CLOCK és a RESET jel megadott lefolyását! Határozza meg a QA, QB és QC jelek ismétlődésének periódusát a CLOCK jel periódusában kifejezve! (15 pont)



3. Határozza meg a Verilog HDL leírásban definiált számláló modulusát! Alakítsa át a leírást úgy, hogy a számláló modulusa 19 legyen és a megadottal ellenkező irányban számláljon! (10 pont)

```

module exam(q, d, increment, load_data,
global_reset, clock);
output [3:0] q;
input [3:0] d;
input load_data, global_reset, clock,
increment;
reg [3:0] q;
always @(posedge clock)
if (global_reset)
q = 4'b0000;

```

```

else if (load_data)
q = d;
if (increment) begin
if (q == 12)
q = 0;
else
q = q + 1;
end
endmodule

```

4. Rajzolja le a lenti Verilog HDL leírásból szintetizálható hardver rajzát és adja meg az in1, in2, sel és out jelek idődiagramját úgy, hogy a viselkedés megfeleljen az alábbi hardver-leírásnak! A diagramot húsz időegységnyi időtartamra rajzolja! (15 pont)

```

module exam ();
reg in1, in2, sel;
wire out;
assign out=(sel)?in1:in2;
initial
begin
in1=0;
in2=0;
sel=0;
end

```

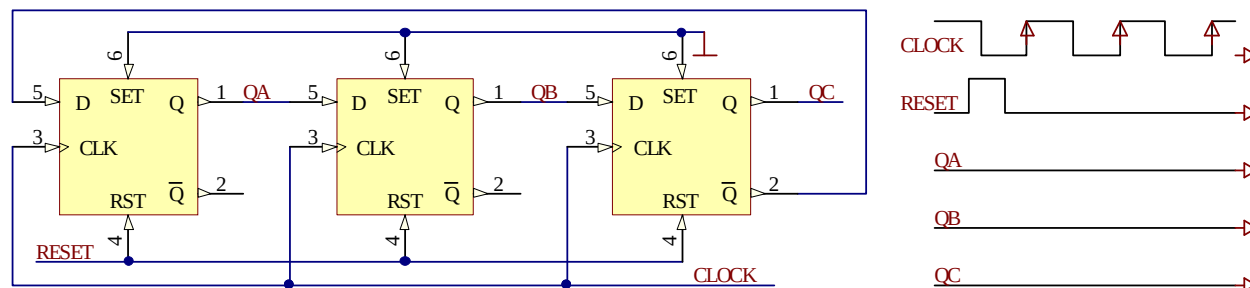
```

always
#1 in1=~in1;
always
#3 in2=~in2;
always
#2 sel=~sel;
endmodule

```

Ime:	Broj indeksa:	Broj poena: (max. 50)
------	---------------	--------------------------

- Nacrtati logičku šemu koja realizuje funkciju $F_1=f(A,B,C,D,E)$ sa minimalnim brojem logičkih kapija koja daje logičku jedinicu na izlazu samo za ulaznu kombinaciju 10010! Pomoću druge, odvojene mreže realizovati drugu funkciju, F_2 , koja u prethodnom slučaju daje logičku nulu a za sve druge kombinacije daje jedinicu! (10 poena)
- Nacrtati vremenske dijagrame signala QA, QB i QC u datom kolu, uzimajući u obzir zadate signale CLOCK i RESET, za 15 perioda CLOCK signala! Odrediti dužinu periode ponavljanja signala QA, QB i QC izraženo u dužini periode CLOCK signala! (15 poena)



- Odrediti modul brojača definisanog u datom Verilog HDL opisu! Izvršiti potrebne promene u opisu tako da modul brojanja bude 19 i da brojač broji u suprotnom smeru! (10 poena)

```

module exam(q, d, increment, load_data,
global_reset, clock);
output [3:0] q;
input [3:0] d;
input load_data, global_reset, clock,
increment;
reg [3:0] q;
always @(posedge clock)
if (global_reset)
q = 4'b0000;

```

```

else if (load_data)
q = d;
if (increment) begin
if (q == 12)
q = 0;
else
q = q + 1;
end
endmodule

```

- Nacrtati hardver koji će se sintetizovati na bazi dole navedenog Verilog HDL opisa i dati vremenske dijagrame za signale *in1*, *in2*, *sel*, *out* tako da ponašanje hardvera odgovara datom opisu! Dijagrame crtati za interval od dvadeset vremenskih jedinica! (15 poena)

```

module exam ();
reg in1, in2, sel;
wire out;
assign out=(sel)?in1:in2;
initial
begin
in1=0;
in2=0;
sel=0;
end

```

```

always
#1 in1=~in1;
always
#3 in2=~in2;
always
#2 sel=~sel;
endmodule

```