

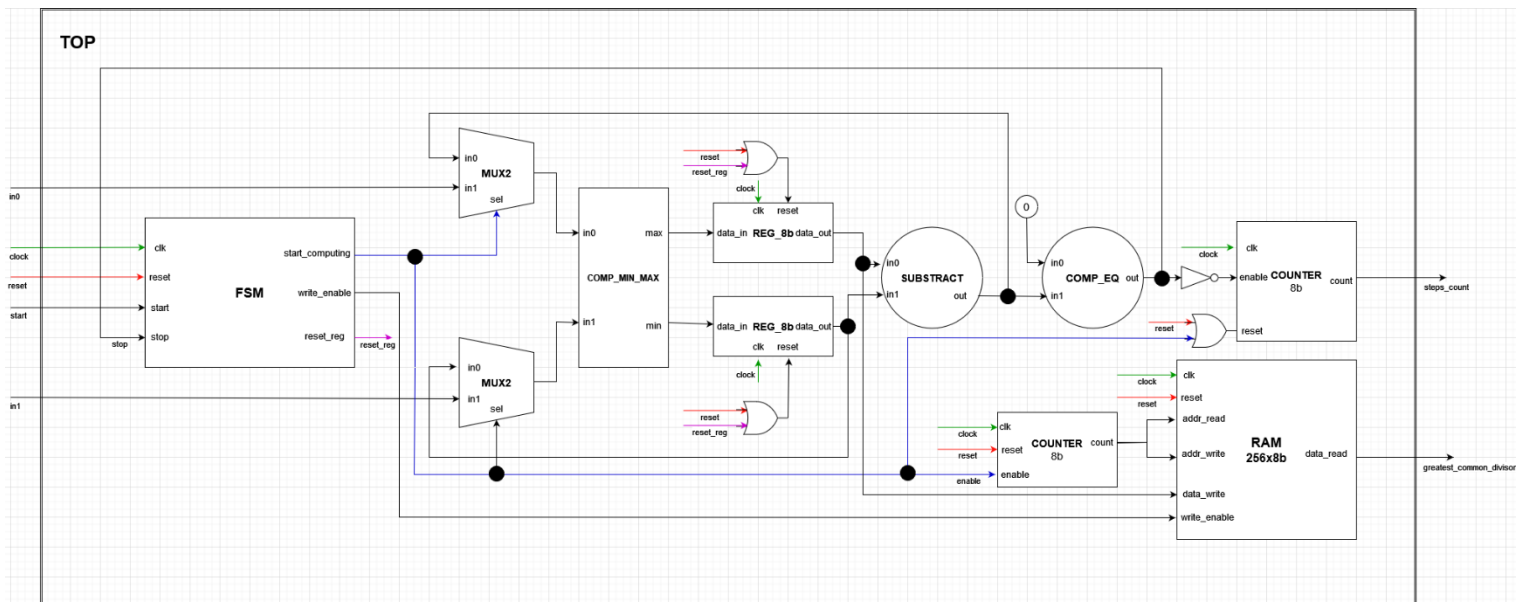
Grupa: 423Aa

Timp de lucru: 1h 30min

### Cerința:

Implementați schema de mai jos in Verilog si efectuați simulări pentru verificarea funcționalității.

### Schema bloc:



### Descrierea schemei + cerințe speciale:

Circuitul implementează un sistem de găsim al celui mai mare divizor comun între două numere prin metoda scăderilor repetate.

Modulul de scădere (SUBTRACT) execută scăderea lui in1 din in0.

Registrii au intrarea enable blocată pe valoarea 1 (nu a mai fost desenată în schema bloc).

Modulul de memorie RAM are citire asincronă.

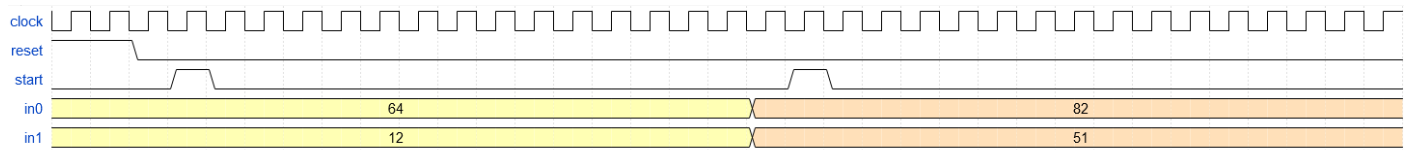
Automatul (FSM) are trei stări: IDLE, COMPUTING și END. Automatul își schimbă starea pe frontul pozitiv (posedge). Ieșirile sunt modificate pe frontul negativ (negedge).

La primirea semnalului de start, automatul trece din starea IDLE în starea COMPUTING și transmite către circuit „mesajul” **start\_computing**. Automatul funcționează în starea COMPUTING până când primește semnalul de stop. În momentul în care automatul primește semnalul de stop, da voie să se execute scrierea prin semnalul **write\_enable**. Din starea END, automatul trece direct în starea inițială (IDLE), provocând resetarea registrilor prin semnalul **reset\_reg**. În orice alt moment în afara celor menționate ieșirile automatului sunt 0.

## Simulare:

Pentru testarea circuitului se va lăsa sa funcționeze 70 ns. Un ciclu de ceas are 2 ns.  
După fiecare start, se așteaptă 30 ns pentru vizualizarea rezultatelor la ieșire.

Formele de unda de intrare trebuie sa corespunda cu figura de mai jos:



După realizarea acestor forme de unda se va adaugă un **exemplu personal** pentru a demonstra înțelegerea funcționării circuitului. Pentru explicarea valorilor rezultate se vor scrie comentarii in fișierul de simulare. **Explicația va fi scurta si la obiect!**

## Programare placa:

**Atenție la ordinea biților!**

clock - Clock 100MHz (F14)

reset – BTN2 (H2)

start - BTN3 (J1)

in0 – SW[7:0]

in1 – SW[15:8]

greatest\_common\_divisor – LED[7:0]

steps\_count – LED[15:8]

**Barem:** Total - 30 puncte

**Design:** 22 puncte

- fsm – 4p
- counter – 3p
- ram – 3p
- reg\_8b – 3p
- mux – 1p
- comp\_min\_max 1p
- subtract – 1p
- comp\_eq – 1p
- top – 5p  
(interfața 1p,  
instanțiere 3p,  
fire + conexiuni 1p)

**Simulare:** 5p + 1p

- Testbench 3p + 1p  
(1.5p instanțiere +  
1.5p generarea semnalelor +  
**1p exemplu personal si  
explicații**)
- Demonstrație simulare
- (îmi arăți) – 2p

**Programare Placa:** 3p

- Fișierul de constrângeri  
(pin planner) - 1p
- Demonstrație funcționare  
(imi arati) – 2p