

Inlämningsuppgift VI – Tillståndsmaskin eller digital klocka

Mål

- Kunna skapa en konstruktion innefattande multipla moduler, bestående av exempelvis flankdetektering, tillståndsmaskiner och timerkretsar.

Totalt 3p (G = 1p, VG = 2p, VG+ = 3p)

OBS! Endast en av uppgifterna behöver genomföras. Genomför ni båda får ni givetvis poäng för båda (därav VG+).

För full poäng i respektive uppgift ska koden vara lättläslig, väl dokumenterad och indenterad:

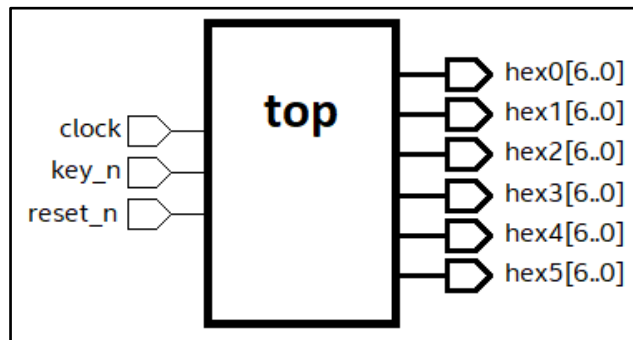
- a) Samtliga moduler, portar, signaler och processer ska namnges med beskrivande namn.*
- b) Syftet med det publika interfacet (samtliga moduler och portar) ska dokumenteras via kommentarsblock (på engelska).*
- c) Indenteringen ska vara konsekvent och ska utgöras av fyra blanka steg (inte ett tab-tecken) per block.*

Alternativ 1 – Tillståndsmaskin (G-uppgift, ger 1p)

- Du ska konstruera en tillståndsmaskin för en räknare i VHDL. Räknaren ska räkna upp från 0 – 99. Värdet ska kontinuerligt skrivas ut hexadecimalt på 7-segmentsdisplayer HEX[1:0] på FPGA-kortet. Vid uppräknings till 100 ska räknaren nollställas.
 - Tillståndsmaskinen ska implementeras via följande tillstånd:
 - **COUNTER_OFF**: Ingen uppräknings sker.
 - **COUNTER_STATE_SLOW**: Räknaren räknas upp en gång i sekunden.
 - **COUNTER_STATE_MEDIUM**: Räknaren räknas upp tio gånger i sekunden.
 - **COUNTER_STATE_FAST**: Räknaren räknas upp hundra gånger i sekunden.
 - För att styra tillståndsmaskinens tillstånd ska tryckknappar KEY[2:0] användas:
 - a) Vid nedtryckning av tryckknapp KEY0 ska tillståndsmaskinens tillstånd uppdateras till nästa.
 - b) Vid nedtryckning av tryckknapp KEY1 ska tillståndsmaskinens tillstånd uppdateras till föregående.
 - c) Vid nedtryckning av tryckknapp KEY2 ska tillståndsmaskinens återställas till startvärdet (lysdioderna släckta).
 - Implementera konstruktionen så att den är generisk i den mån att maxvärdet för uppräknings enkelt kan väljas till ett godtyckligt värde mellan 0 – 255 samt att uppräkningshastigheterna för respektive tillstånd kan ändras efter behov. *Konstruktionen ska också innehålla logik för att förebygga metastabilitet.*
 - Välj FPGA-kort Terasic DE0 (enhet 5CEBA4F23C7). Toppmodulen ska ha samma namn som projektet.
 - Använd följande parametrar och portar:
 - **clock** - 50 MHz systemklocka på FPGA-kortet.
 - **reset_n** - Asynkron inverterande reset-signal ansluten till en tryckknapp, ansluten till KEY2.
 - **button_n[1:0]** – Tryckknappar som används för att uppdatera aktuellt tillstånd framåt eller bakåt, anslutna till KEY[1:0].
 - **hex[1:0]** - 7-segmentsdisplayer som visar aktuellt värde 0 – 99, anslutna till HEX[1:0].
 - Verifiera konstruktionen på ett FPGA-kort. Anslut insignalerna i enlighet med portbeskrivningen ovan.
- ### Examination
- VHDL-koden ska lämnas in via en .qar-fil på Classroom.
 - Vänligen svara på följande frågor i en README-fil (döp filen *README.md*), som lämnas in tillsammans med .qar-filen:
 1. Vad lärde ni er av projektet?
 2. Vad var lätt/svårt?
 3. Vad hade ni velat ha lärt er mer innan projektet?
 4. Övriga kommentarer?

Alternativ 2 – Digital klocka (VG-uppgift, ger 2p)

- Ni ska enskilt konstruera en digital 24-timmars klocka, som ska valideras via ett FPGA-kort. Tiden ska visas på sex 7-segmentsdisplayer. Klockan ska kunna räkna upp från tiden 00:00:00 till 23:59:59. I figuren nedan visas konstruktionens toppmodul *top* i dess grundform:



Figur 1 – Konstruktionens toppmodul *top* i dess grundform.

- Toppmodulen *top* innehar i dess grundform följande in- och utsignaler:
 - Insignal *clock* ska utgöras av en 50 MHz intern klocka på FPGA-kortet.
 - Insignal *key_n* utgör en styrsignal för att starta respektive stoppa klockan.
 - Insignal *reset_n* används som asynkron reset-signal för att återställa klockan till startläget (tiden 00:00:00).
 - Utsignaler *hex[5:0]* utgör 7-segmentsdisplayer som var och en visar en siffra 0 – 9. Tillsammans visar displayerna en tid på formen HH:MM:SS, där *hex[5:4]* visar aktuell timme, *hex[3:2]* visar aktuell minut och *hex[1:0]* visar aktuell sekund.
- Efter slutförd konstruktion ska den digitala klockan kunna genomföra kontinuerlig uppräknig i 24 timmar, mätt från tiden 00:00:00 upp till 23:59:59, där aktuell tid i form av timme, minut och sekund visas på sex 7-segmentsdisplayer. Vid fortsatt uppräknig ska klockan sedan räkna om från tiden 00:00:00 igen. *Konstruktionen ska också inneha logik för att förebygga metastabilitet.*
- Klockan ska vid behov kunna startas och stoppas via en tryckknapp döpt *key_n*. Vid start ska klockan vara avstängd, samtidigt som tiden 00:00:00 skrivs ut på displayerna, och måste då startas via nedtryckning av denna tryckknapp. När klockan är påslagen medför nedtryckning av samma tryckknapp *key_n* att klockan stannar och då ska aktuell tid på 7-segmentsdisplayerna stå kvar. Via nedtryckning av samma tryckknapp ska klockan sedan kunna startas igen och fortsätta där den stannade.
- I konstruktionen ska också en reset-signal implementeras, som vid aktivering medför total systemåterställning, där klockan nollställs och stoppas. Denna reset-signal realiseras via en tryckknapp döpt *reset_n*. Efter systemåterställning måste användaren manuellt starta om klockan via nedtryckning av tryckknappen *key_n* igen.
- Verifiera konstruktionen på ett FPGA-kort. Anslut insignalerna i enlighet med portbeskrivningen ovan.

Examination

- VHDL-koden ska lämnas in via en .qar-fil på Classroom.
- Vänligen svara på följande frågor i en README-fil (döp filen *README.md*), som lämnas in tillsammans med .qar-filen:
 - Vad lärde ni er av projektet?
 - Vad var lätt/svårt?
 - Vad hade ni velat ha lärt er mer innan projektet?
 - Övriga kommentarer?