

Practicum 5.1 Stoplicht

1. teken het state machine diagram voor een stoplicht. Het stoplicht heeft vier toestanden die gekenmerkt worden door de lampen die branden: groen, oranje, rood, rood-oranje (het is een buitenlands stoplicht, waarbij vlak voor het groen wordt, de lampen rood en oranje kort tegelijkertijd branden). Elke opgaande clock geeft een overgang naar een nieuwe toestand. Maak gebruik van een asynchrone reset;
2. start een nieuw project en schrijf in VHDL state machine met twee processen. De entity is:

```
entity stoplicht is
    port(clkin, reset, vraaggroen : in std_logic;
          lights : out std_logic_vector(2 downto 0));
end entity delay;
```
3. test het stoplicht m.b.v. de waveform monitor of testbench;
4. verander het stoplicht zodat wanneer het signaal 'vraaggroen' hoog is, het stoplicht bij de volgende clockovergang meteen op groen springt (dit is duidelijk een geïdealiseerd stoplicht).
5. test het stoplicht m.b.v. de waveform monitor (of testbench);
6. wat gebeurt er als je input-signalen weghaalt of toevoegt in de sensitivity list? Is het resultaat hetzelfde? En als je de reset synchroon maakt? Maakt het dan uit welke input-signalen er in de sensitivity list staan?
7. maak een nieuwe VHDL file en koppel de drie lampen aan de leds (LEDG 0..2). Koppel de drukknoppen (KEY2 t/m KEY0) aan de signalen clock, reset en vraaggroen. Let op dat de drukknoppen LAAG zijn wanneer ingedrukt. De drukknoppen zijn ontdenderd;
8. compileer, programmeer en test op het DE0 board;
9. bekijk de gesynthetiseerde hardware m.b.v. de RTL viewer en de State Machine Viewer. Komt het schema overeen met je verwachtingen?

Practicum 5.2 Cijferslot

10. teken het state machine diagram voor een cijferslot. Ingang van het cijferslot zijn 4 bits, die de binaire code van het cijfer 0..9 representeren. Bij elke opgaande clock wordt gekeken welk cijfer wordt ingevoerd. Wanneer de juiste code van 4 cijfers in de juiste volgorde wordt ingevoerd, gaat het slot open (een LED branden). Elke verkeerde invoer geeft een reset naar de begintoestand. Neem als juiste code 3371. Om het geheel makkelijk te kunnen testen wordt voor elk cijfer dat correct is ingevoerd (elke andere toestand in de state machine) een andere LED opgelicht.
11. Maak een nieuw project en maak een architecture en entity voor het cijferslot. De entity is:

```
entity cijferslot is
    port(clkin, reset: in std_logic;
          digitsin: in std_logic_vector(3 downto 0);
          lights : out std_logic_vector(4 downto 0));
end entity cijferslot;
```

12. test het cijferslot m.b.v. de waveform monitor (of testbench).
13. maak een nieuwe VHDL file en koppel de vijf leds (LEDG 0..4) aan *lights*. Koppel de drukknoppen (KEY2 en KEY1) aan de signalen *clock* en *reset*. Koppel de schakelaars (SW3 t/m SW0) aan *digitsin*;
14. compileer, programmeer en test op het DE0 board;