**VHDL Ontwikkeljournaal voor blokje**

**Auteur: Jesse Ernste**

**Datum: 04-03-2025 & 06-03-2025**

**Overzicht**

Dit document beschrijft de algemene werking, implementatie en tests van de motorbesturing met FSM.

**Systeemoverzicht**

De FSM bestuurt een motor met meerdere snelheden en wachttijden. De ingangen en uitgangen worden gedefinieerd in de signaaltabel.

**Inleiding**

Dit project betreft de implementatie van een finite state machine (FSM) voor de besturing van een motor of centrifuge. De FSM regelt verschillende snelheden en wachttijden op basis van inputsignalen en systeemstatussen.

**Functionaliteit:**

* + Start en stop van de motor
  + Meerdere snelheidsniveaus (bijv. 200, 800, 1200 tpm)
  + Wachttijden tussen snelheidsveranderingen
  + Verwerking van configuratie-invoer (centrifugekeuze)

**Installatie en Gebruik**

1. Open de VHDL-code in Vivado/Quartus.
2. Compileer en syntheseer het ontwerp.
3. Laad de code op de FPGA of voer een simulatie uit.
4. Controleer of de FSM correct door de toestanden loopt.

**Ontwerpbeslissingen**

* **FSM-gebaseerde besturing**: De motor wordt bestuurd met een toestandsmachine, zoals geïllustreerd in het toestandsdiagram.
* **Snelheidsoverdracht**: De FSM verhoogt de snelheid stapsgewijs en wacht een bepaalde tijd voordat de volgende overgang plaatsvindt.
* **Gebruik van std\_logic en std\_logic\_vector**: Dit zorgt voor compatibiliteit met FPGA-hardware.
* **Asynchrone reset**: De FSM wordt gereset wanneer aan\_uit = '0'.

**Implementatie**

* **Bestandsstructuur:**
  + fsm\_motor.vhd: Hoofdmodule met de FSM
  + fsm\_motor\_tb.vhd: Testbench voor simulatie
* **Belangrijkste signalen:**
  + aan\_uit: Start/stop van het systeem
  + centrifugekeuz: Configuratie van de motor
  + clk: Kloksignaal
  + go\_trommel: Startsignaal voor de trommel
  + motor: 3-bits uitgangssignaal voor snelheid
  + trommel\_ready: Statussignaal voor gereedheid

**Onderhoud en Uitbreiding**

* Voor verdere uitbreiding kan een PWM-regeling worden toegevoegd.
* Extra foutdetectiemechanismen kunnen worden geïmplementeerd voor een robuustere werking.
* Het systeem kan worden geoptimaliseerd voor laag energieverbruik.

**Problemen en Oplossingen**

* **Glitches in snelheidsoverdracht**: Opgelost door extra synchronisatie met klokranden.
* **Resettiming**: Asynchrone reset correct geïmplementeerd om racecondities te voorkomen.

**Testplan**

**Doel van het testplan**

Het testplan beschrijft de methodologie en tests die worden uitgevoerd om te verifiëren dat de FSM correct functioneert en voldoet aan de specificaties.

**Testomgeving**

* **Hardware:** FPGA of simulatieomgeving
* **Software:** ModelSim / Vivado / Quartus Prime
* **Testbench:** fsm\_motor\_tb.vhd

**Testcases**

|  |  |  |
| --- | --- | --- |
| **Testcase** | **Beschrijving** | **Verwachte Uitkomst** |
| TC1 | Startsignaal wordt geactiveerd | FSM schakelt naar eerste snelheid |
| TC2 | FSM doorloopt snelheidstrappen | Motor verhoogt snelheid correct |
| TC3 | Wachttijden worden correct nageleefd | Geen te vroege of te late schakeling |
| TC4 | Resetten van FSM | FSM keert terug naar starttoestand |
| TC5 | Onverwachte invoerwaarden | FSM blijft stabiel of schakelt correct |

**Acceptatiecriteria**

* De FSM moet alle gedefinieerde toestanden correct doorlopen.
* Er mogen geen glitches of onverwachte signaalveranderingen optreden.
* Wachttijden en snelheden moeten overeenkomen met de specificaties.

Afbeelding met schermopname, software, tekst, Multimediasoftware

Door AI gegenereerde inhoud is mogelijk onjuist.