

Proiectarea si Implementarea unui Periferic PWM Controlat prin Interfata SPI

Avarvarei Andrei-Alexandru
Ilie Eduard Cristian
Echipa 3

14 decembrie 2025

Cuprins

1 Introducere	2
2 Arhitectura generala a sistemului	2
3 Analiza detaliata a implementarii	2
3.1 Interfata fizica si sincronizarea semnalelor (spi_bridge.v)	2
3.2 Decodificarea instructiunilor (instr_dcd.v)	2
3.3 Banca de registre si logica de auto-clear (regs.v)	2
3.4 Numaratorul si prescaler-ul (counter.v)	3
3.5 Generatorul PWM (pwm_gen.v)	3
4 Corectarea modulului top.v	3
5 Simulare si validare	3
6 Concluzii	4

1 Introducere

Acet document descrie soluțiile adoptate pentru proiectarea și implementarea unui periferic hardware PWM (Pulse Width Modulation), care poate fi configurat printr-o interfață SPI. Proiectul a fost realizat conform specificațiilor temei, având ca scop principal crearea unui sistem modular, configurabil și fiabil.

Perifericul permite controlul complet al semnalului PWM, inclusiv perioada, factorul de umplere, modul de aliniere și direcția de numărare, prin scrierea în registre interne accesibile unui master SPI extern. Documentația se concentrează pe detaliile implementării și pe provocările apărute în timpul integrării sistemului.

2 Arhitectura generală a sistemului

Sistemul este organizat modular și este compus din mai multe blocuri funktionale, fiecare având o responsabilitate bine definită. Aceasta abordare a permis dezvoltarea și testarea individuală a modulelor, urmata de integrarea acestora în modulul de top.

Fluxul de date este urmatorul: semnalele seriale sunt receptionate prin interfața SPI, sunt decodificate în comenzi interne, stocate în banca de registre și utilizate ulterior de numarator și de generatorul PWM pentru sinteza semnalului de ieșire.

3 Analiza detaliată a implementării

3.1 Interfața fizică și sincronizarea semnalelor (spi_bridge.v)

Modulul spi_bridge reprezintă interfața directă cu exteriorul și implementează protocolel SPI în modul CPOL=0, CPHA=0. Datele sunt esantionate pe frontul crescător al semnalului sclk, iar ieșirea miso este actualizată pe frontul descrescător.

Semnalele sclk și cs_n sunt asincrone față de ceasul intern al sistemului, motiv pentru care sunt tratate cu atenție pentru a evita probleme de metastabilitate. Recepția unui byte complet este semnalizată prin impulsul byte_sync, utilizat de etajele superioare pentru delimitarea comenziilor SPI.

3.2 Decodificarea instrucțiunilor (instr_dcd.v)

Modulul instr_dcd este responsabil de interpretarea fluxului de date provenit de la spi_bridge. Decodificarea se face folosind o structură secvențială cu două etape: recepționarea header-ului, care conține tipul operației și adresa, urmata de recepționarea datelor în cazul unei operații de scris.

Semnalele de control read și write sunt generate sub forma de impulsuri de un singur ciclu de ceas, pentru a preveni scrieri multiple neintenționate în banca de registre.

3.3 Banca de registre și logica de auto-clear (regs.v)

Modulul regs gestionează stocarea tuturor parametrilor de configurație ai perifericului. Registrele de 16 biți sunt accesate prin două operații succesive pe magistrala SPI, una pentru octetul inferior și una pentru octetul superior.

Un registru special este cel de resetare software a numaratorului. Acesta este implementat cu o logică de tip auto-clear, astfel încât, după activarea resetului printr-o scrisoare

SPI, semnalul este dezactivat automat de hardware dupa un numar redus de cicluri de ceas. Aceasta solutie reduce incarcarea magistralei si simplifica controlul din software.

3.4 Numaratorul si prescaler-ul (counter.v)

Modulul counter reprezinta baza de timp a sistemului. Acesta suporta numarare crescatoare sau descrescatoare, resetare la atingerea valorii period si utilizarea unui prescaler configurabil.

Prescaler-ul este implementat eficient folosind deplasari logice pe biti, conform relatiei:

$$Limit = 2^{prescale}$$

Aceasta metoda este optima din punct de vedere al resurselor hardware utilizeze.

3.5 Generatorul PWM (pwm_gen.v)

Modulul pwm_gen utilizeaza valoarea curenta a numaratorului si registrele de comparatie pentru a genera semnalul PWM. Sunt implementate mai multe moduri de functionare, incluzand alinierea la stanga, alinierea la dreapta si modul nealiniat intre doua valori de comparatie.

Pentru siguranta sistemului, iesirea PWM este conditionata de semnalul pwm_en. Atunci cand acest semnal este dezactivat, iesirea este fortata la nivel logic zero, prevenind aparitia de impulsuri nedorite in timpul configurarii.

4 Corectarea modulului top.v

Fisierul top.v primit initial continea mai multe probleme structurale care impiedicau functionarea corecta a sistemului.

Prima problema a fost redeclararea porturilor de intrare si iesire ca semnale interne de tip wire. In Verilog, porturile sunt implicit de tip wire, iar redeclararea lor poate conduce la erori de compilare sau la conectari incorecte intre module.

A doua problema majora a fost conectarea incompleta a modulului spi_bridge. Semnalele esentiale pentru comunicarea SPI, respectiv byte_sync, data_in si data_out, nu erau conectate catre modulele instr_dcd si regs. Din aceasta cauza, comenziile SPI nu erau decodificate, registrele nu erau programate, iar perifericul ramanea in stare inactiva.

In implementarea finala, aceste probleme au fost corectate prin eliminarea redeclararilor ilegale ale porturilor si prin conectarea completa a interfetei SPI intre module, fara a modifica interfata externa a modulului top. Astfel, compatibilitatea cu testbench-ul primit a fost pastrata.

5 Simulare si validare

Validarea functionala a fost realizata prin simulare, folosind testbench-ul primit. Valorile necunoscute observate la inceputul simularii sunt normale si apar din cauza initializarii semnalelor inainte de aplicarea resetului global.

Dupa configurarea corecta a registrelor prin SPI, numaratorul incepe sa functioneze conform parametrilor setati, iar semnalul PWM generat respecta caracteristicile asteptate.

6 Concluzii

Proiectul a demonstrat implementarea completa a unui periferic PWM configurabil prin interfața SPI. Principalele dificultăți întampinate au fost legate de integrarea corectă a modulelor și de conectivitatea semnalelor, nu de logica internă a acestora.

Soluțiile adoptate au permis obținerea unui sistem funcțional, robust și extensibil, care respectă specificațiile temei.