Documentație PWMGEN

Brăduț-Mihai Iordache(Lawenda700)
Eduard Simion(eddy-24)
Mario Timoc(MarioTimoc21)

December 14, 2025

Cuprins

1 SPI BRIDGE

Modulul SPI bridge folosește 2 blocuri `always`, unul pentru frontul crescător al ceasului și celălalt pentru frontul descrescător. Ambele verifică activarea funcției de reset și, în caz că aceasta este activată, resetează valorile output-urilor la valorile default + valoarea celor 2 countere interne.

Pe frontul crescător (blocul `always posedge`) se realizează recepția datelor de la Master. Se citește bit cu bit din linia `miso` în registrul de shift intern, de la MSB la LSB. Odată ce se citește valoarea ultimului bit (LSB), counter-ul se resetează, iar octetul complet format este salvat în `data_latch` și se inversează starea semnalului intern `flag_toggle`.

Pe frontul descrescător (blocul `always negedge`) se realizează transmisia datelor către Master. Se scrie pe linia `mosi` (Output din Slave în această configurație), shiftând biții din `data_out` de la MSB la LSB.

Sincronizarea cu domeniul de ceas al sistemului (`clk`) se face printr-un bloc separat care urmărește schimbarea semnalului `flag_toggle`. Când se detectează o tranziție, se generează pulsul `byte_sync` și se actualizează `data_in`. Citirea și scrierea se realizează doar atunci când chip select-ul `cs_n` este setat (activ pe 0).

2 INSTRUCTION DECODER

Instruction Decoder-ul funcționează asemenea unui AFD cu 3 stări. Trecerea dintr-o stare în alta se realizează pe frontul crescător al ceasului, iar în caz de reset se revine în starea S0. Stările automatului:

S0: Se verifică primirea semnalului `byte_sync` care semnalează primirea datelor de input. Se scrie în `data_write` pe poziția 0 valoarea bitului de pe poziția 6 din `data_write` (ne ajută la scrierea în MSB/LSB din registri), se preia valoarea bitului de pe poziția 7 și, în caz că este 1, `write` este setat ca 1 și se trece în starea S1 a automatului, iar în caz că este 0, `read` este setat ca 1 și se trece în starea S2 a automatului. Totodată, biții [5:0] din `data_in` sunt scriși în `addr`. În caz că `byte_sync` este 0, nu se întâmplă nimic.

S1: Se verifică primirea semnalului `byte_sync` care semnalează primirea datelor de input. Datele din `data_in` sunt transferate în totalitate în `data_write` și se trece în starea S0 a automatului.

În caz că `byte_sync` este 0, nu se întâmplă nimic.

S2: Se trec datele din `data_out` în `data_read` și se trece în starea S0 a automatului.

3 REGISTERS

Se stochează adresele registrelor pe 6 biți în parametrii locali ai modulului. Se declară un registru `sb` care va indica dacă scrierea/citirea se va face în MSB sau în LSB pentru registrele care sunt pe 16 biți. Funcționarea modulului va avea forma unui AFD cu 3 stări (S3, S4, S5). La activarea funcției de reset toți registrii sunt trecuți în valoarea default și se revine la starea S3. Trecerea dintr-o stare în alta se realizează pe frontul crescător al ceasului. Stările automatului:

S3: `sb` ia valoarea bitului 0 din `data_write`. Dacă `write` este 1 se trece la starea S4, alternativ se trece la S5.

S4: `counter_reset` revine la valoarea 0 (acesta trebuie, conform documentației, să fie 1 doar un singur ciclu de ceas după ce este setat). Se verifică, cu un bloc `case`, adresa la care trebuie scrisă informația din `data_write` și implicit registrul. Dacă registrul este pe 16 biți, se verifică `sb` pentru a vedea dacă `data_write` se scrie în MSB sau LSB. Se trece în starea S3.

S5: Se verifică, cu un bloc `case`, adresa la care se află registrul care trebuie citit în `data_read` (dacă adresa nu este una în care se află unul dintre registre, se citește valoarea 0). Dacă registrul este pe 16 biți, se verifică `sb` pentru a vedea dacă `data_read` se citește din MSB sau LSB. Se trece în starea S3.

4 COUNTER

Se creează un counter intern și un *period* intern (numit `ss`), care este un `wire` setat ca 1 pe 16 biți, shiftat la stânga cu `prescale` număr de biți. Se declară un bloc `always` activ pe frontul crescător al ceasului, care, la activarea funcției de reset, resetează `count_val` la valoarea 0, indiferent de direcția de numărare. Registrul intern este mapat direct la ieșirea `count_val`. Numărătoarea începe atunci când `en` are valoarea 1.

Când `upnotdown` este 1, numărătoarea este crescătoare de la 0 la `period`. Când este 0, numărătoarea este descrescătoare; dacă valoarea curentă este 0, următorul pas va încărca valoarea `period`, altfel se decrementează.

Incrementarea/Decrementarea lui `count_value` se realizează atunci când counter-ul intern ajunge la valoarea `ss - 1` (deoarece începe de la 0). Când se incrementează/decrementează `count_value`, counter-ul intern se resetează și el la 0.

5 PWM GENERATOR

Se utilizează un bloc `always` activ pe frontul crescător al ceasului. La primirea semnalului de reset, valoarea lui `pwm_out` se resetează la 0.

Când `pwm_en` este 0, `pwm_out` este și el 0. Când `pwm_en` este 1, se aplică o verificare prioritară: dacă `compare1` este egal cu `compare2`, ieșirea `pwm_out` este forțată la 0. Altfel, se verifică în `functions` tipul de aliniere al semnalului, valoarea `count_val` din momentul respectiv al ceasului și valoarea/valorile cu care aceasta din urmă trebuie comparată. În funcție de aceste condiții se determină semnalul `pwm_out` la momentul dat.

6 Modificări top

Fișierul top.v a fost modificat astfel încât la apelul modulului `spi_bridge.v` să se utilizeze wire-rurile `byte_sync,data_in și data_out, iar în apelul modulului instr_dcd intrarea byte_sync primește valoarea wire-ului byte_sync din top.`