25.06.2023

Sipos Bogdan

grupa 30215

Îndrumător:ing.DIana Pop

Aplicatie FPGA si mouse

Proiect PSN

Cuprins

[1 Specificații 2](#_Toc130249632)

[2 Proiectare 5](#_Toc130249633)

[2.1 Schema Bloc 5](#_Toc130249634)

[2.2 Unitatea de Control și Unitatea de Execuție 5](#_Toc130249635)

[1.2.1 Maparea intrărilor și ieșirilor cutiei mari pe cele două componente UC și UE. 5](#_Toc130249636)

[1.2.2 Determinarea resurselor (UE) 6](#_Toc130249637)

[2.2.3 Schema bloc a primei descompuneri 7](#_Toc130249640)

[2.2.4 Reprezentarea UC prin diagrama de stări (organigrama) 8](#_Toc130249641)

[2.2.5 Schema de detaliu a proiectului 9](#_Toc130249642)

[3 Manual de utilizare și întreținere 9](#_Toc130249643)

[4 Justificarea soluției alese 9](#_Toc130249644)

[5 Posibilități de dezvoltări ulterioare 10](#_Toc130249645)

[6 Bibliografie 10](#_Toc130249646)

**Aplicatie FPGA si mouse**

**1.1. Specificatii**

Fac parte dintr-o echipa care are de proiectat o aplicatie care permite utilizatorului contorizarea numarului de click-uri ale mouse-ului.

Mouse-ul ar trebui sa functioneze dupa cum urmeaza:

* starea initiala este una de asteptare, afisajul SSD va fi gol (starea ”000’’)si se va aprinde ledul ”IS\_LEFT”
* la actionarea butonului stanga al mouse-ului starea curenta se va incrementa si va fi afisata pe afisorul SSD
* la actionarea butonului dreapta al mouse-ului starea curenta se va derementa si va fi afisata pe afisorul SSD
* la actionarea switch-ului ”REVERSE” se vor inversa functiile mouse-ului (la actionarea butonului dreapta al mouse-ului starea curenta se va incrementa iar la actionarea butonului stanga al mouse-ului starea curenta se va derementa) si se va stinge ledul ”IS\_LEFT”
* butonul ”RESET” are rolul de a goli afisajul SSD (starea ”000”)

Mouse-ul exista deja.

**1.2. Protocol PS2**

După pornire, microcontrolerul este în modul de configurare, fie descarcă un flux de biți în FPGA, fie așteaptă programarea acestuia din alte surse. Odată ce FPGA este programat, microcontrolerul trece în modul aplicație, care, în acest caz, este modul USB HID Host. Firmware-ul din microcontroler poate conduce un mouse sau o tastatură atașată la conectorul USB de tip A la J2 etichetat „USB”. Suportul pentru hub nu este disponibil momentan, așa că poate fi folosit doar un singur mouse sau o singură tastatură. PIC24 conduce mai multe semnale în FPGA – două sunt folosite pentru a implementa o interfață standard PS/2 pentru comunicarea cu un mouse sau o tastatură, iar celelalte sunt conectate la portul de programare serial cu două fire al FPGA, astfel încât FPGA poate fi programat de la un fișier stocat pe un pen drive USB.

Microcontrolerul cu funcție auxiliară ascunde protocolul USB HID de FPGA și emulează o magistrală PS/2 de stil vechi. Microcontrolerul se comportă exact ca o tastatură sau un mouse PS/2. Aceasta înseamnă că noile modele pot reutiliza nucleele PS/2 IP existente. Mouse-urile și tastaturile care folosesc protocolul PS/2 folosesc o magistrală serială cu două fire (ceas și date) pentru a comunica cu o gazdă. Pe Basys 3, microcontrolerul emulează un dispozitiv PS/2, în timp ce FPGA joacă rolul gazdei. Atât mouse-ul, cât și tastatura folosesc cuvinte de 11 biți care includ un bit de pornire, un octet de date (în primul rând LSB), paritate impară și un bit de oprire, dar pachetele de date sunt organizate diferit, iar interfața tastaturii permite transferuri de date bidirecționale ( astfel încât dispozitivul gazdă să poată aprinde LED-urile de stare de pe tastatură). Orarul autobuzului este prezentat în figura de mai jos.O imagine care conține text, Font, diagramă, captură de ecran

Descriere generată automat

Ceasul și semnalele de date sunt conduse numai atunci când au loc transferuri de date; în caz contrar, acestea sunt menținute în starea inactivă la logica „1”. Acest lucru necesită ca atunci când semnalele PS/2 sunt utilizate într-un proiect, pull-up-urile interne trebuie să fie activate în FPGA pe pinii de date și de ceas. Semnalul ceasului este în mod normal condus de dispozitiv, dar poate fi ținut la un nivel scăzut de gazdă în cazuri speciale. Timingurile definesc cerințele de semnal pentru comunicațiile mouse-la-gazdă și comunicațiile bidirecționale de la tastatură. Un circuit de interfață PS/2 poate fi implementat în FPGA pentru a crea o interfață de tastatură sau mouse.

Când o tastatură sau un mouse este conectat la Basys 3, o comandă „autotest trecut” (0xAA) este trimisă gazdei. După aceasta, pot fi emise comenzi către dispozitiv. Deoarece atât tastatura, cât și mouse-ul folosesc același port PS/2, se poate spune tipul de dispozitiv conectat folosind ID-ul dispozitivului. Acest ID poate fi citit prin lansarea unei comenzi Read ID (0xF2). De asemenea, un mouse își trimite ID-ul (0x00) imediat după comanda „autotest trecut”, care îl deosebește de o tastatură.

O imagine care conține text, Font, linie, alb

Descriere generată automatOdată intrat în modul flux și raportarea datelor a fost activată, mouse-ul emite un ceas și un semnal de date atunci când este mutat: în caz contrar, aceste semnale rămân la „1” logic. De fiecare dată când mouse-ul este mutat, sunt trimise trei cuvinte de 11 biți. de la mouse la dispozitivul gazdă, așa cum se arată în fig. 10. Fiecare dintre cuvintele de 11 biți conține un bit de început „0”, urmat de 8 biți de date (LSB mai întâi), urmat de un bit de paritate impar și terminat cu un bit de oprire „1”. Astfel, fiecare transmisie de date conține 33 de biți, unde biții 0, 11 și 22 sunt biți de pornire „0”, iar biții 11, 21 și „33 sunt biți de oprire „1”. Cele trei câmpuri de date pe 8 biți conțin date de mișcare, așa cum se arată în figura de mai jos. Datele sunt valabile la marginea descendentă a ceasului, iar perioada de ceas este de la 20 la 30 KHz.

Mouse-ul presupune un sistem de coordonate relativ în care deplasarea mouse-ului spre dreapta generează un număr pozitiv în câmpul X, iar deplasarea spre stânga generează un număr negativ. De asemenea, mutarea mouse-ului în sus generează un număr pozitiv în câmpul Y, iar deplasarea în jos reprezintă un număr negativ (biții XS și YS din octetul de stare sunt biții de semn – un „1” indică un număr negativ). Mărimea numerelor X și Y reprezintă rata de mișcare a mouse-ului - cu cât numărul este mai mare, cu atât mouse-ul se mișcă mai rapid (biții XV și YV din octetul de stare sunt indicatori de depășire a mișcării - un „1” înseamnă că a avut loc depășirea) . Dacă mouse-ul se mișcă continuu, transmisiile pe 33 de biți sunt repetate la fiecare 50 de ms sau cam asa ceva. Câmpurile L și R din octetul de stare indică apăsarea butoanelor Stânga și Dreapta (un „1” indică faptul că butonul este apăsat).

O extensie populară pentru mouse-ul standard PS/2 este Microsoft Intellimouse. Aceasta include suport pentru un total de cinci butoane de mouse și trei axe de mișcare (dreapta-stânga, sus-jos și o roată de scroll). Aceste caracteristici suplimentare necesită utilizarea unui pachet de date de mișcare de 4 octeți în loc de pachetul standard de 3 octeți.

**2. Proiectare**

**2.1 Schema bloc**

CLK

RESET

REVERSE

IS\_LEFT

Valoare SSD

PC

PD

**2.2 Unitatea de Control si Unitatea de Executie**

Cutia neagră a sistemului trebuie descompusă mai departe pentru a putea găsi componente implementabile. Vom face o descompunere top-down a problemei pana cand ajungem la circuite cunoscute, iar apoi vom implementa bottom-up.

Prima descompunere a oricărui sistem este una în care vom diferenția între logica de control din sistem și resursele sistemului. Logica de control este reprezentată de Unitatea de Control iar resursele sunt reprezentate de Unitatea de Execuție. Orice algoritm poate fi descompus în acest fel (reprezentarea abstractă a unui algoritm se face printr-un flow-chart).

**1.2.1 Maparea intrarilor si iesirilor cutiei pe cele doua componente UC si UE**

Afisajul valoarii de pe ssd

IS\_LEFTEFT

REVERSE

CLOCK

RESET

PC

PD

UE

UC

Putem împărți atât intrările cât și ieșirile în 2 categorii: de date si de control. Aceasta separare este esențială la început.

* Intrări de date: clock\_1.0 HZ, clock\_mouse(PC), data\_mouse(PD)
* Intrari de control: switch-urile REVERSE si RESET
* Ieșiri de date: afisajul valorii de pe SSD
* Iesiri de control: led-ul ”IS\_LEFT”

**1.2.2 Determinarea resurselor (UE)**

Pentru a stabili mai departe legăturile dintre UC și UE trebuie mai întâi să identificăm resursele pe baza cărora luăm decizii sau resursele care devin informații pentru utilizator. Aceste resurse pot sa genereze semnale către unitatea de control și pot fi controlate de UC prin semnale de Enable sau Reset. Orice informație pe baza careia se ia decizii trebuie sa vină de la o resursă care generează acea informație și o transmite mai departe UC.

Resursele pot fi circuite simple, care pot fi implementate direct (numarator, registru etc) sau resurse complexe (algoritm de dat rest, algoritm de inmultire, etc). Aceste resurse complexe pot apărea în prima descompunere cu cutii negre cărora trebuie sa le stabilim intrari si iesiri, dar ulterior trebuie descompuse mai departe (de obicei tot în UC și UE) pana cand ajungem la circuite cunoscute.

**RESURSE:**

1. Numarator reversibil

counterout1

counterout2

counterout3

clock\_1.0 HZ

RESET

up

down

Acesta este un numarator binar sincron reversibil. Acest circuit va avea 3 iesiri pe 4 biti care reprezinta fiecare cifra a numarului de trei cifre (contorul numarului de click-uri) numite counterout1, counterout2 si counterout3. Avem nevoie de intrarea reset data de UC pentru UE. In plus, avem nevoie de un semnal de tact dat de placuta si doua semnale care ne spun cum sa fie numararea, directa (UP) sau inversa (Down).

1. PS2\_READER

buton\_dreapta

buton\_stanga

clock\_1.0 HZ

RESET

PDD

PC

Aceasta componenta are rolul de a prelua pachetele de biti de la mouse cu ajutorul semnalului de tact de la placuta, semnalului de tact de la mouse (PC) si a datelor date (PD) de la mouse. Dupa aceea, verifica daca butoanele de click ale mouse-ului au fost apasate (bitul 40, pentru butonul din dreapta, respectiv 41, pentru butonul de stanga, are valorea 1) si transmitem la unitatea de executie valorile celor doi biti, 40 si 41.

<https://www.fpga4student.com/2017/12/how-to-interface-mouse-with-FPGA.html\>

1. Componenta de afisare

anozi

catozi

clk\_200HZ

n2

n3

n1

Aceasta componenta ne permite afisarea numarului de click-uri. Intrarile in aceasta resursa sunt 3 cifre pe 4 biti numite n1, n2 si n3. La aceasta componenta intra si un clk cu o frecventa de 200HZ pentru a creea o iluzie optica la afisarea numarului complet, deoarece SSD-ul are 4 anozi comuni si 8 catozi independenti. Iesirea este valoarea codificata a numerelor transmise pe anozi si catozi.

1. Divizor de frecventa

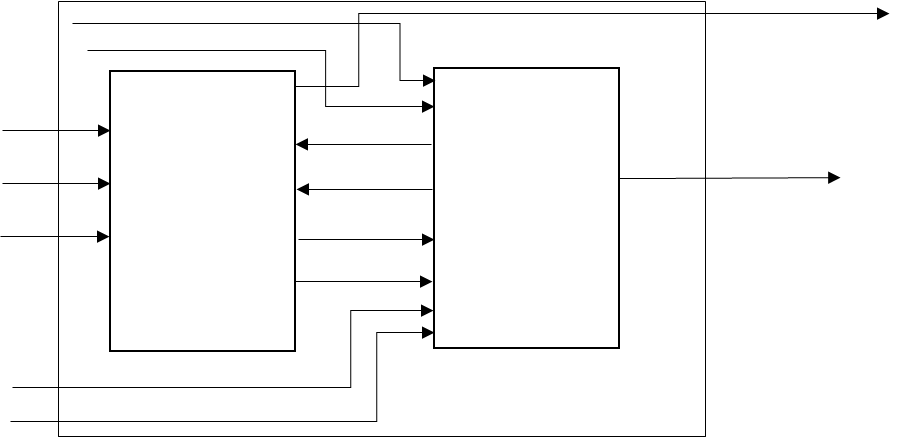
clk\_in

clk\_out

Acesta componenta returnează un clock cu frecventa de 200Hz si cu factorul de umplere de 50%. Acesta ne va trebui la SSD.

**2.2.3 Schema bloc a primei descompuneri**

IS\_LEFTT



UE

UC

Buton\_dreapta

Buton\_stanga

INC

DEC

PD

PC

RESET

CLOCK

REVERSE

Afisajul valorii de pe SSD

**2.2.4 Reprezentarea UC prin diagrama de stări (organigrama)**

Reverse

Left-Click

IS\_LEFT

Right-Click

Afisare SSD

RESET

1

0

1

0

Left-Click

1

0

Right-Click

1

1

0

0

DOWN

1

0

UP

UP

DOWN

O imagine care conține diagramă, linie, text, captură de ecran

Descriere generată automat**2.2.5 Schema de detaliu a proiectului**

**3. Manual de utilizare si intretinere**

Prima data trebuie ca mouse-ul sa fie legat de placuta prin intermediul portului USB.

Daca aparatul se afla in modul normal, indicat prin faptul ca ledul IS\_LEFT este aprins, la apasarea butonului de click stanga se va incrementa valoarea de pe afisor iar la apasarea butonului de click dreapta se va decrementa.

Prin actionarea switch-ului REVERSE, led-ul IS\_LEFT se stinge, iar acum butonul de click stanga va decrementa valoarea de pe afisor si butonul de click dreapta o va incrementa.

Prin actionarea switch-ului RESET afisorul isi reseteaza valoarea, afisand ”000”. Valoarea de ”000” se mentine cat timp switch-ul RESET este actionat.

Daca valoarea de pe afisor ajunge la ”999” valoarea nu se mai poate incrementa atata timp cat este pe ”999” si poate fi numai decrementata.

Daca valoarea de pe afisor ajunge la ”000” valoarea nu se mai poate decrementa atata timp cat este pe ”000” si poate fi numai incrementata.

1. **Justificarea solutiei alese**

Motivul pentru acesta soluție este simplitatea modului de funcționare și implementare. Acest numărător de click-uri are o funcționalitate simplă și o interfață clară.

Am ales reprezentarea pe 3 biti a valorii de pe SSD in loc de cea pe 4 biti din cauza simplitatii si a improbabilitati ca cineva sa doreasca sa ajunga la o valoare de ordinul miilor.

**5. Posibilitati de dezvoltare ulterioare**

-afisarea pe un ecran a valorii de pe ssd

-calcularea vitezei de deplasare a mouse-ului pe axele de coordonate

-verificarea stream-ului de date transmis de mouse, pentru evitarea eventualelor erori

-se poate salva doar bitii 41 si 40 primiti de la mouse nu tot pachetul de biti pentru o eficienta mai sporita

**6. Bibliografie**

* [**https://digilent.com/reference/programmable-logic/basys-3/reference-manual**](https://digilent.com/reference/programmable-logic/basys-3/reference-manual)
* [**https://www.fpga4student.com/2017/12/how-to-interface-mouse-with-FPGA.html**](https://www.fpga4student.com/2017/12/how-to-interface-mouse-with-FPGA.html)
* [**https://www.fpgakey.com/tutorial/section927**](https://www.fpgakey.com/tutorial/section927)
* [**https://www.fpgakey.com/tutorial/section929**](https://www.fpgakey.com/tutorial/section929)
* [**https://www.youtube.com/watch?v=A1YSbLnm4\_o&ab\_channel=LBEbooks**](https://www.youtube.com/watch?v=A1YSbLnm4_o&ab_channel=LBEbooks)